A COMPARISON OF A COAXIAL FOCUSED LASER DOPPLER SYSTEM IN ATMOSPHERIC MEASUREMENTS

by

S. Karaki

Prepared under

National Aeronautic and Space Administration
Contract No. NAS8-26234
Marshall Space Flight Center
Huntsville, Alabama

June 1973

CER71-72SK35

Contractor:	Civil Engineering Department Engineering Research Center Foothills Campus Colorado State University Fort Collins, Colorado 80521
Title:	Final Report
Author:	S. Karaki
Type of Report:	Final Report
Contract Number:	NAS8-26234
Control Number:	DCN 1-0-75-00101(1F)&S1(1F) DCN 1-0-75-00101 S1 (1F)
Date:	June 25, 1973

Prepared for George C. Marshall Space Flight Center
NASA, Huntsville, Alabama 35812

TABLE OF CONTENTS

TABLE OF CONTENTS - Continued

LIST OF FIGURES

LIST OF FIGURES - continued

LIST OF FIGURES - continued

LIST OF TABLES

INTRODUCTION

Measurements of fluid flow speed may be made by utilizing the Doppler shift of laser light scattered from small particles suspended in the flowing fluid. The principle of the Doppler shift is of course well known, but only recently was a technique introduced by Yeh and Cummins (1964) to utilize the Doppler shift of a laser radiation to successfully measure fluid flow speeds. Since that time there have been a number of separate investigations reported in the literature (see references). The instrument utilized in this investigation was developed by a team of scientists at NASA/MSFC (Huntsville, Alabama), Raytheon Company (Sudbury, Massachusetts) and Lockheed Missiles and Space Company (Huntsville, Alabama). Much of the technology used was originally developed in assembling a system to be used in subsonic and supersonic gas flows with large quantities of particle entrainment [Rolfe et al. (1968)]. The system used in this study involved only aerosols and particulate matter suspended naturally in the atmosphere.

Interest in application of the instrument has broadened currently (1972) to a variety of practical situations where a remote-sensing instrument has particular advantages over conventional velocimeters. Two applications currently under research is for use as an airport warning system for wake vortex detection and as an air-borne system for clear-air turbulence detection. A potentially important use of the instrument is in meteorological investigations of the atmospheric boundary layer. Further uses of the instrument could be for remote air-pollution detection and for measurement of mass and momentum fluxes in a variety of fluid flow fields.

In principle it is possible to measure "point" velocities in the flow field with complete vector directional resolution. A laboratory three-dimensional instrument is presently being investigated at NASA/MSFC (Huntsville, Alabama), where also an atmospheric three-dimensional arrangement is under research and development. The instrument used in this investigation was a one-dimensional co-axial system, using a 25-watt CO_2 laser and back-scattered radiation. The direction of wind velocity was resolved by utilizing an ordinary wind-vane direction sensor.

The purpose of this research project was to obtain measurements of atmospheric velocities and turbulence with the laser Doppler system and to compare the results with cup anemometer and hot-wire measurements in the same wind field.

BASIC PRINCIPLES

The frequency of laser light scattered by moving particles in a flow field is shifted by the Doppler effect. The Doppler shift is detected by optical mixing of the emitted or incident and scattered beams. A variety of optical configurations is possible to accomplish the optical mixing. In the present arrangement the back-scattered radiation along the axis of the incident beam was redirected into the laser to combine with the original laser beam. The resultant heterodyne or "beat" frequency is equal to the difference in frequencies of the emitted and scattered frequencies, and is directly proportional to the particle speed. If the scatterers are small, and no relative velocity exists between the particle and the fluid, then fluid velocity is measured. An infrared detector was used to convert the Doppler-shifted frequency to a measurable electrical signal. The arrangement of the system is shown schematically in Figure 1.

The laser Doppler velocity measurement system (hereinafter referred to as the laser Doppler velocimeter and mnemonically denoted LDV) is almost instantaneous and has the advantage that no prior calibration is required as with other velocity instruments. The range of detectable velocities is very large. There is minimal perturbation of the fluid flow field by the laser radiation. The spatial resolution which is fixed ultimately by diffraction limitations can be controlled to a large degree by size and optical quality of the lenses and mirrors.

A nonrelativistic derivation of velocity determination from the Doppler shift frequency follows. A definition diagram relative to the derivation is shown in Figure 2. For purpose of clarity, the scattered beam is shown at an arbitrary angle θ from the direction of particle motion. In the case of a coaxial system, $\theta = \alpha$.

The emitted monochromatic laser radiation of wave length λ_o and speed c illuminates a particle having a velocity $\hat{V}$. The direction of the incident beam is defined by the unit vector $\hat{r}_o$. If the particle is motionless, the number of waves incident on the particle per unit of time is $f_o = c/\lambda_o$, where c is the speed of the laser radiation and λ_o is the wave length.

If the particle is in motion at an angle α with respect to the incident beam, the frequency of the waves per unit of time relative to the moving particle is

$$f_p = \frac{c + V\cos\alpha}{\lambda_o}$$

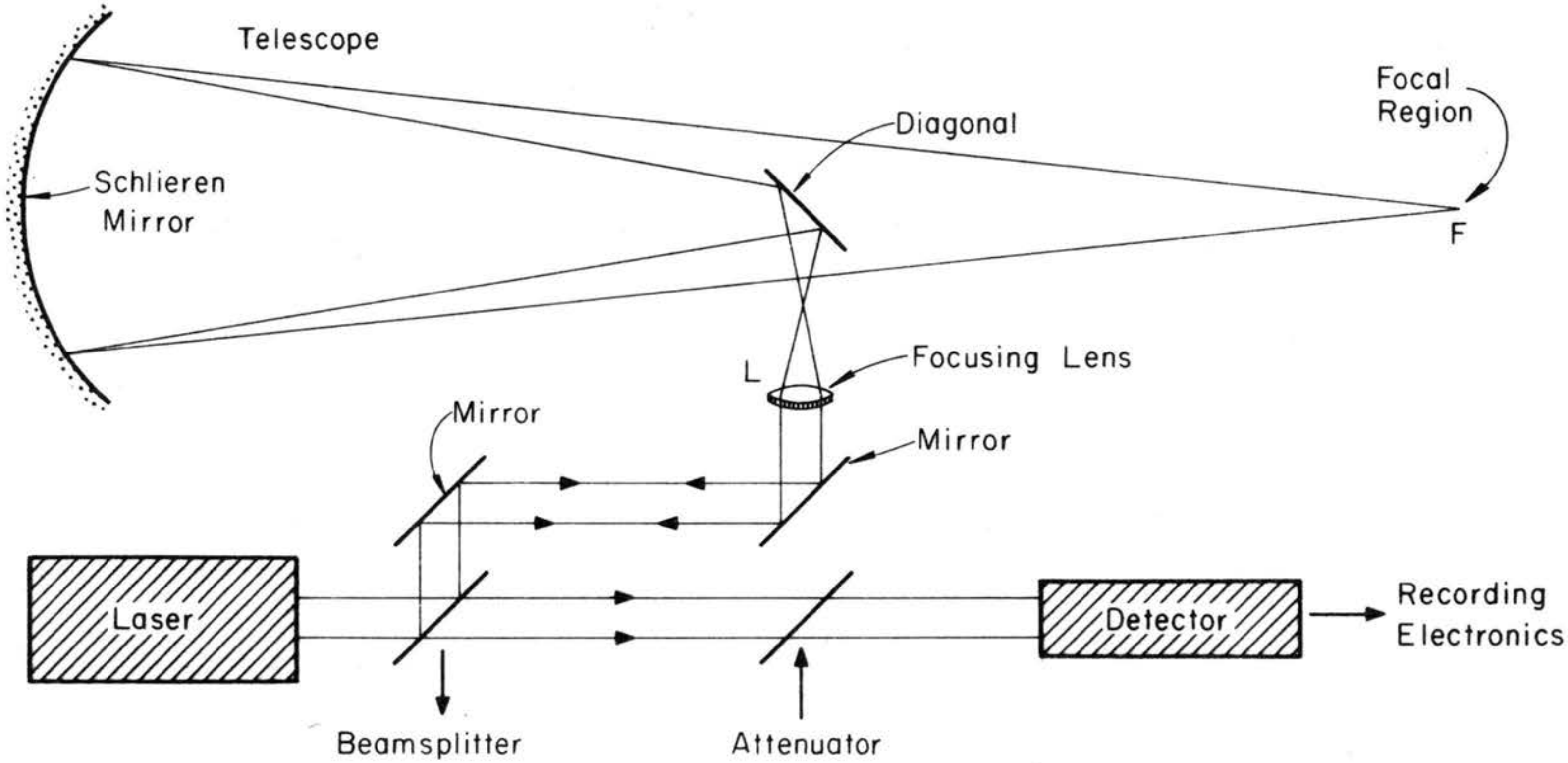


Figure 1. Schematic arrangement of the laser Doppler velocimeter.

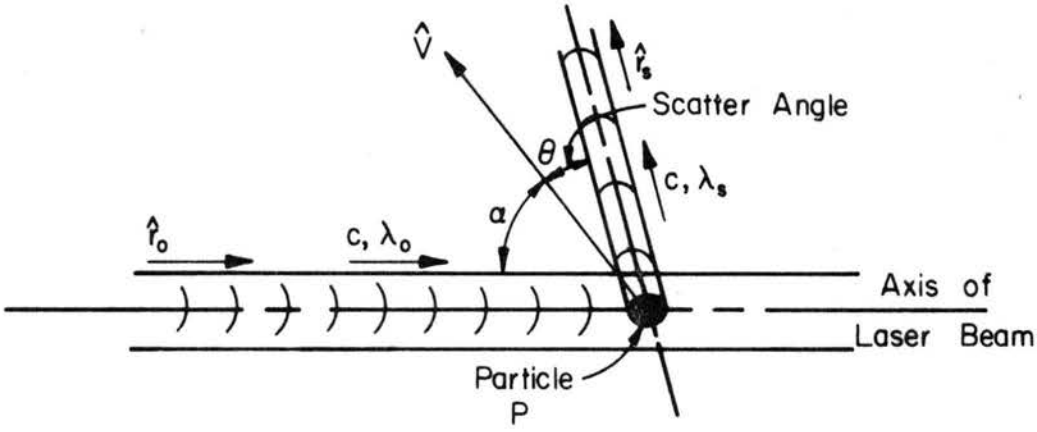


Figure 2. Definition diagram for Doppler shift frequency.

which is also the frequency of the scattered waves relative to the particle. The scattered radiation is directed toward a fixed point along a direction $\hat{r}_s$ from point P. The frequency of the scattered radiation relative to the particle is f_p, but to a fixed observer along $\hat{r}_s$, the wave length appears to be

$$\lambda_s = \frac{c - V\cos\theta}{f_p} = \frac{c - V\cos\theta}{c + V\cos\alpha}\lambda_o$$

and the frequency of the scattered radiation appears to be

$$f_s = \frac{c}{\lambda_s} = \frac{c}{\lambda_o}\left(\frac{c + V\cos\alpha}{c - V\cos\theta}\right)$$

which is rearranged to give

$$f_s = \frac{c}{\lambda_o}\left(\frac{1 + \frac{V\cos\alpha}{c}}{1 - \frac{V\cos\theta}{c}}\right) \quad .$$

The apparent shift in frequency, the Doppler shift, is

$$f_D = f_s - f_o$$

or,

$$f_D = \frac{1}{\lambda_o}[V(\cos\alpha + \cos\theta)]$$

using the approximation that $\frac{|V|}{c} << 1$.

For backscatter along the incident laser beam, $\theta = \alpha$, thus

$$f_D = \frac{2V\cos\alpha}{\lambda_o}$$

and

$$V = \frac{\lambda_o f_D}{2\cos\alpha} = \frac{c}{2\cos\alpha}\frac{f_D}{f_o} \quad .$$

In particular the component of the particle velocity along the laser beam axis V_o is always determinable from

$$V_o = V\cos\alpha = \frac{\lambda_o f_D}{2} = \frac{cf_D}{2f_o} \quad .$$

The wavelength of the CO_2 laser was 10.6 microns, thus the velocity is given by

$$V_o = 5.3 \times 10^{-6} f_D \text{ m/sec}$$

or,

$$V_o = .53 \text{ cm/sec/KHz Doppler shift.}$$

DESCRIPTION OF THE LASER DOPPLER VELOCIMETER

The optical configuration of the LDV is shown schematically in Figure 1. It consists of a 25-watt, 10.6μ, CO_2 laser, beam splitters, mirrors and attenuators, an f8 12-inch Newtonian telescope and a liquid-helium cooled Ge-Hg infrared detector.

Based on relative power of 100 percent of the laser output (nominal 25 watts), the power at the focal region F was about 60 percent. The focal region is the sample space or volume from where the scattered signal is effectively heterodyned. The relative power at the detector was about 1 percent.

The laser radiation is focused at the desired range by a 2-in. focusing lense L. A diagonal, 1-7/8 by 2-21/32 inches mounted on a spider within the 15-in. diameter tube of the telescope, directs the beam to a 12-in. diameter schlieren mirror mounted at the end. The mirror is adjustable on a 3-point mount. Physical limitation of the focusing lense movement limited the near range of the telescope focus to about 60 feet from the mirror. The other limit of the telescope focusing range is limited to about 250 feet by the size of the diagonal. Of course if the power loss from beam "spill over" at the diagonal is not of concern the range can be extended. A curve of focal distance as a function of lense movement is shown in Figure 3. The reference position of the lense is arbitrary and made relative to 60 feet in the figure. The range of the telescope relative to "performance" is also diffraction limited [cf. Lockheed Missiles and Space Company (LMSC) progress report D162417, July 23, 1970].

Spatial Resolution

The spatial resolution of the system is specified in terms of the signal-to-noise, S/N, ratio. A calculation of S/N was made by LMSC (cf. Appendix A, Interim Report D225028, June 1971). It has been shown [Thomson and Dorian (1967)] that only radiation scattered from the region near the focus of the telescope contributes most significantly

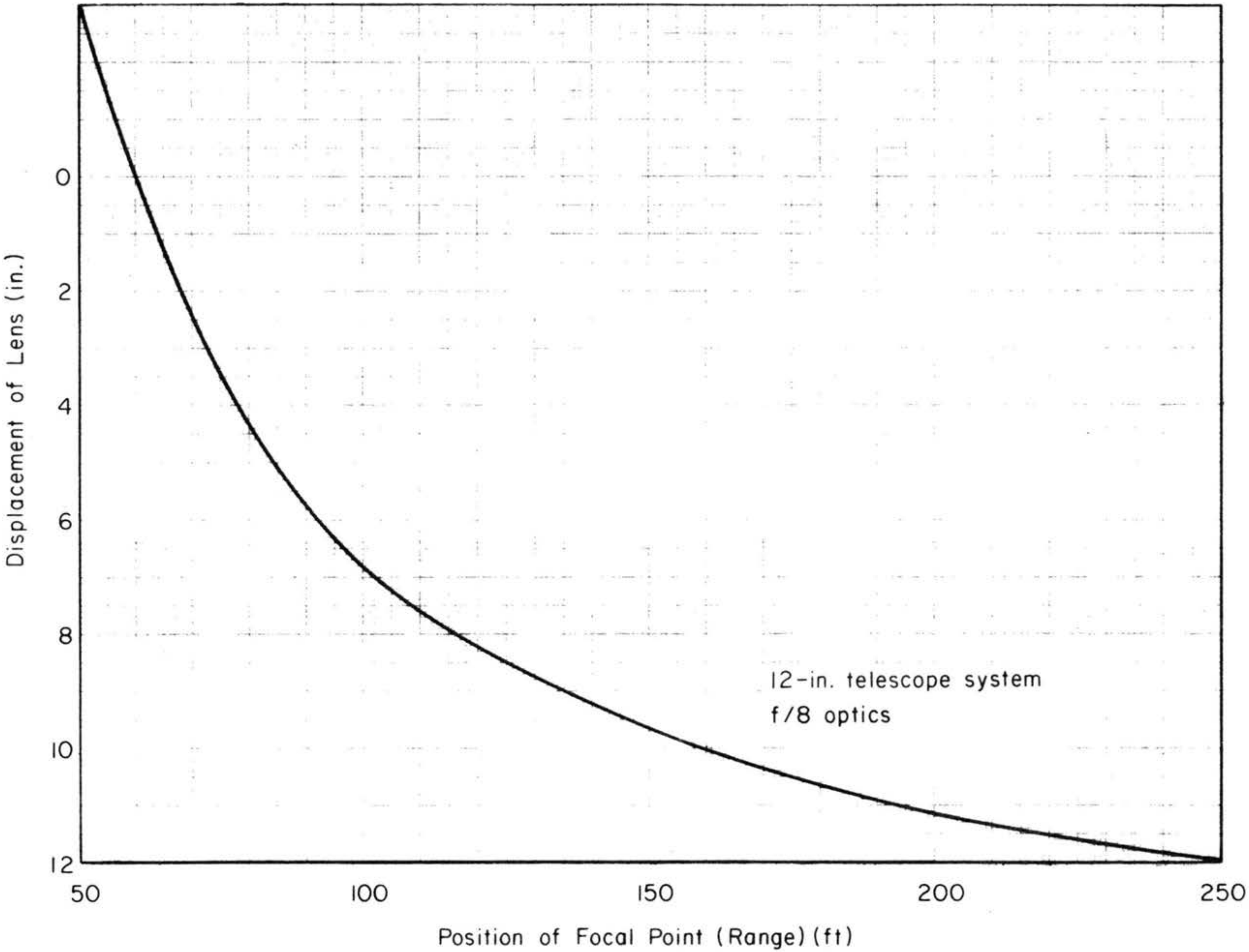


Figure 3. Range positioning as a function of lens position.

to the Doppler signal. Nevertheless, there is some amount of heterodyned signal attributable to scattered returns in the whole space of illumination. The ratio of S/N from the focal region in comparison to the total S/N, then, is a method of defining the spatial resolution. A curve of spatial resolution (axial dimension) as a function of focal range is reproduced from the LMSC calculations as Figure 4.

Signal Processing

There are several options for discriminating the Doppler shift in frequency from the detector. These are:

1. Spectrum analyzer
2. Wide-band frequency discriminator
3. Filter bank
4. Doppler frequency tracker
5. Phase-locked receiver.

The merits, advantages and disadvantages are discussed by Rolfe et al. (1968). In this system principal use was made of a spectrum analyzer and to a limited extent of a frequency tracker.

Spectrum Analyzer - The Hewlett-Packard Model 8553B/8552A spectrum analyzer used in this investigation is a swept superhetrodyne receiver. A simplified block diagram is shown in Figure 5. Essentially the signal frequency is compared with a harmonic of the local oscillator frequency and the analyzer displays the signal directly in the frequency domain as a carrier with its side bands. The center frequency is tuneable, and a scan of the total band is selectable. The spectrum analyzer resolution is determined by a selectable IF bandwidth. The scan time can vary from 1 millisecond to 100 seconds for the selected scan width.

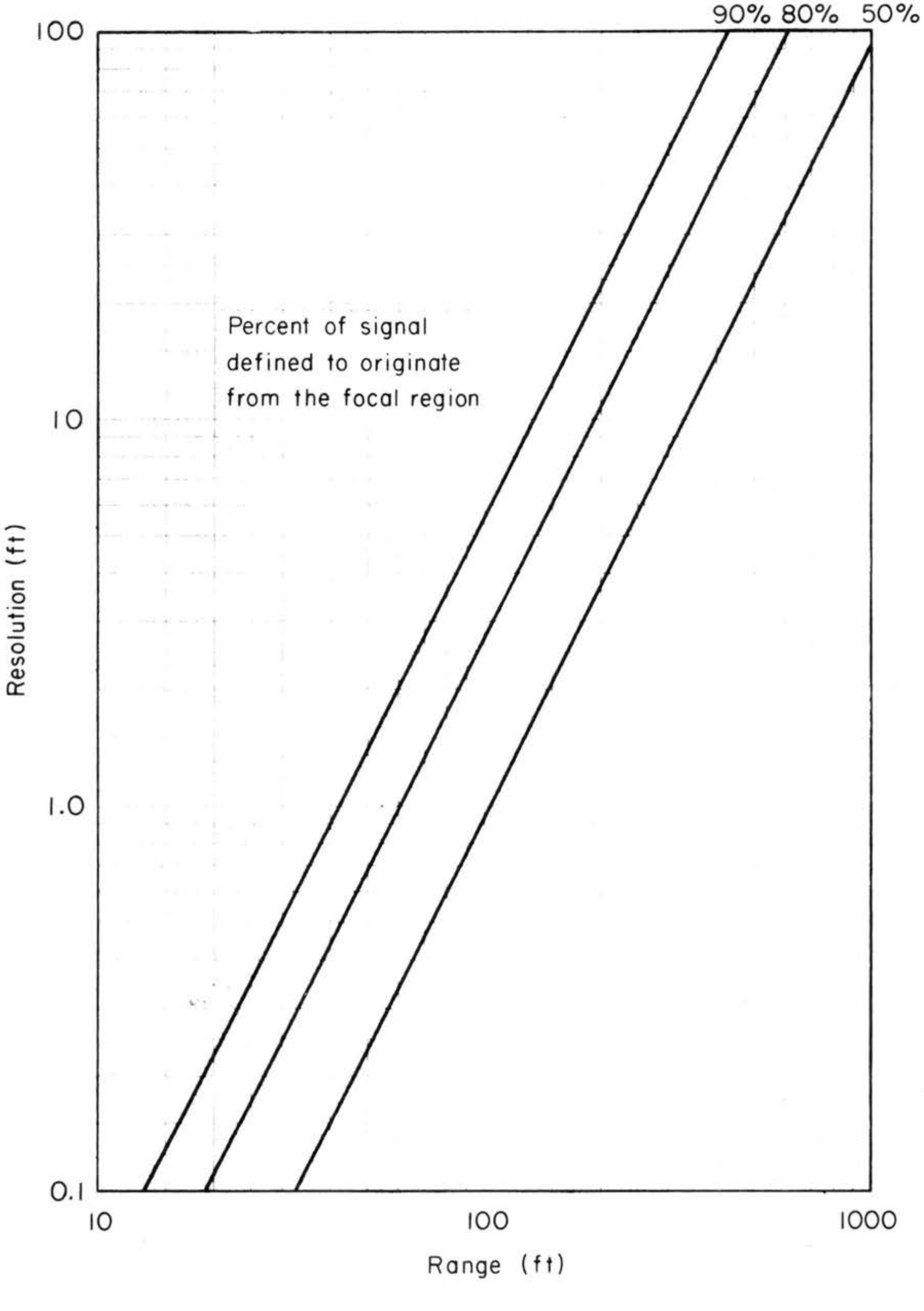


Figure 4. Spatial resolution of the 12-in. telescope.

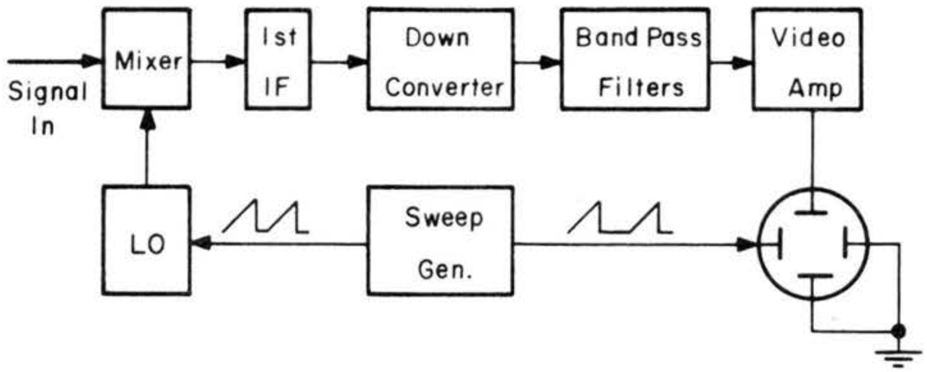


Figure 5. Spectrum analyzer block diagram.

If time intervals are too small, power output of the signal may be too small to measure. On the other hand, for large time intervals the output reflects the spectrum of particle speeds passing through the resolution focal volume of the beam, and can give therefore only a spectrum of velocities (Doppler frequencies) and not an "instantaneous" velocity as a function of time. Clearly, for "instantaneous" velocities the time interval should be consistent with the focal resolution volume, convected particle speed and S/N ratio of the spectrum analyzer.

In order to convert spectral information in frequency space to velocity, use is made of the linear variation of velocity with Doppler frequency shift. The frequency bandwidth of the spectrum analyzer is "swept" at a rate consistent with resolution of the analyzer and the power contained in the bandwidth is recorded on a conventional FM recorder in time space. Conversion from time to frequency hence to velocity in principle is simple, requiring only a reference zero frequency and known bandwidth or alternatively a calibrated external frequency. The rate at which the spectral bandwidth is swept is controlled externally to the spectrum analyzer. A schematic arrangement of the process including preconditioning of the detector signal is shown in Figure 6. A typical time, frequency trace of the power output is shown in Figure 7.

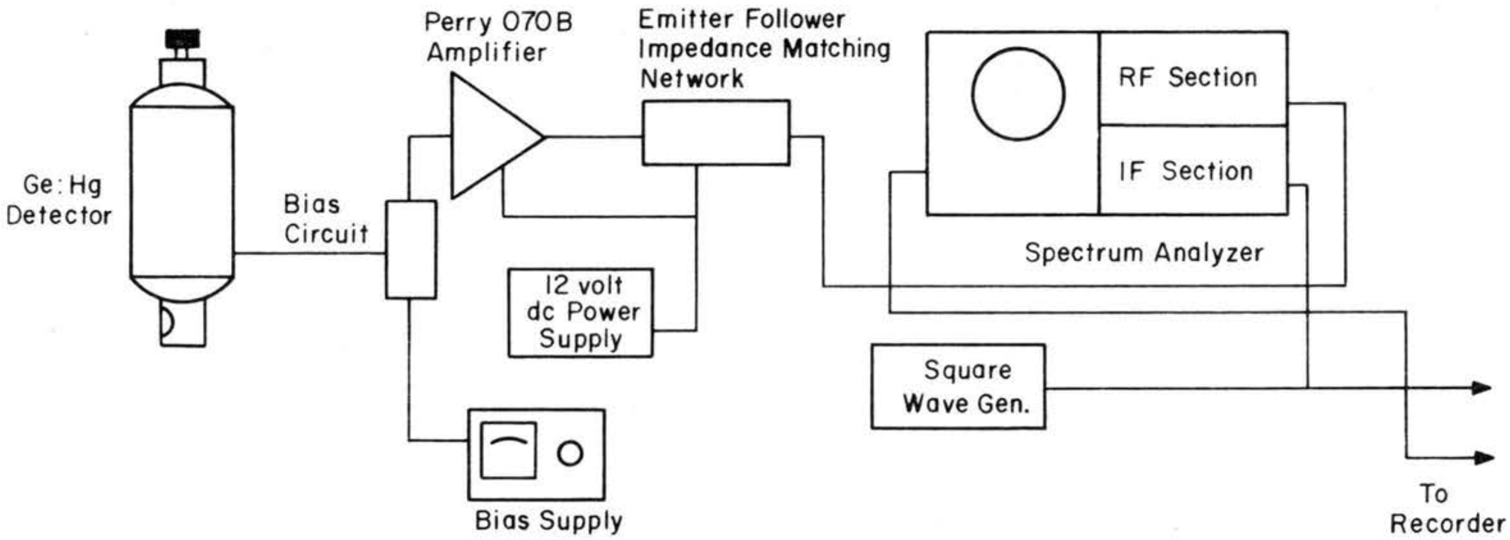


Figure 6. Block diagram of signal detection circuitry.

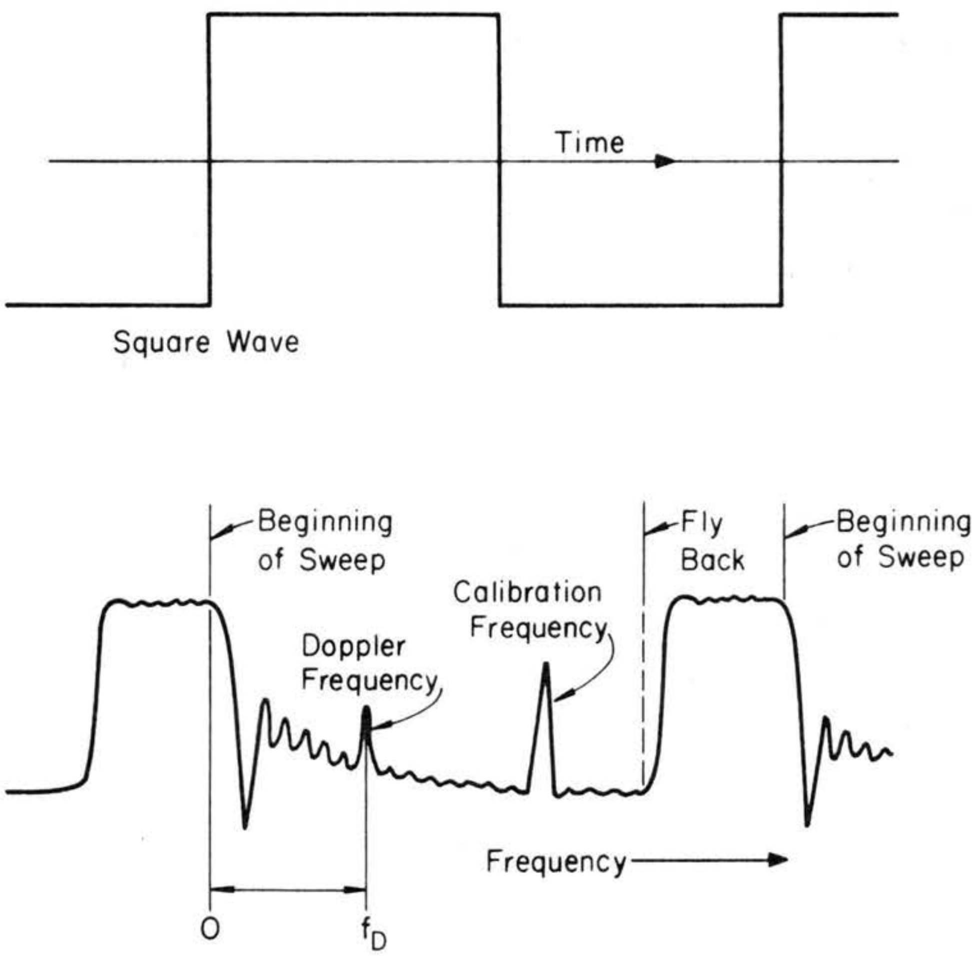


Figure 7. Typical spectrum analyzer output for calibration and Doppler frequencies.

Doppler frequency tracker - A device which provides an output voltage proportional to a given Doppler frequency is termed a Doppler frequency tracker, or simply frequency tracker. The technique is also known as "frequency compressive feedback" or "frequency-locked loop" [cf. Rolfe et al. (1968)]. The Doppler frequency, $f_D(t)$, is heterodyned with a local oscillator frequency. The local oscillator frequency, f_{LO}, is varied so that the difference $f_{LO} - f_D$ is constant and equal to the center frequency of a discriminator. The driving voltage of the local oscillator is then proportional to f_D, hence to the velocity. A schematic representation of the tracker is shown in Figure 8.

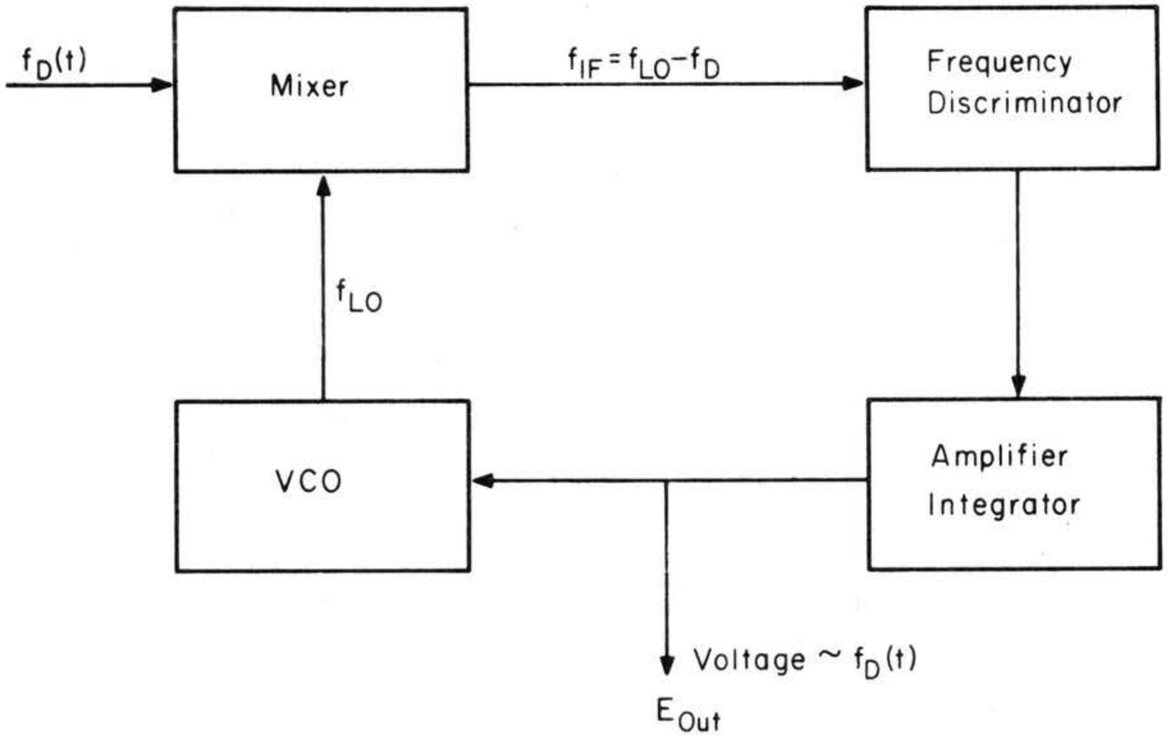


Figure 8. Block diagram of frequency tracker.

TEST FACILITIES

The field site for the experiments was selected at the Colorado State University airport (Christman field) located approximately three miles west of the city of Fort Collins, Colorado (see Figure 9). The test site has a clear field from northwest to northeast, and from south to southwest. There are buildings and trees in the range from south to east, although the nearest building is some 1100 feet away. To the west is the foothills of the Rocky Mountains about a mile distant. The site was selected on the basis of land and power availability and proximity to the research center about 1/2 mile away. The dominant wind directions in the area are north-south, as evident from the alignment of the runway, although strong winds also blow over the foothills directly from the west.

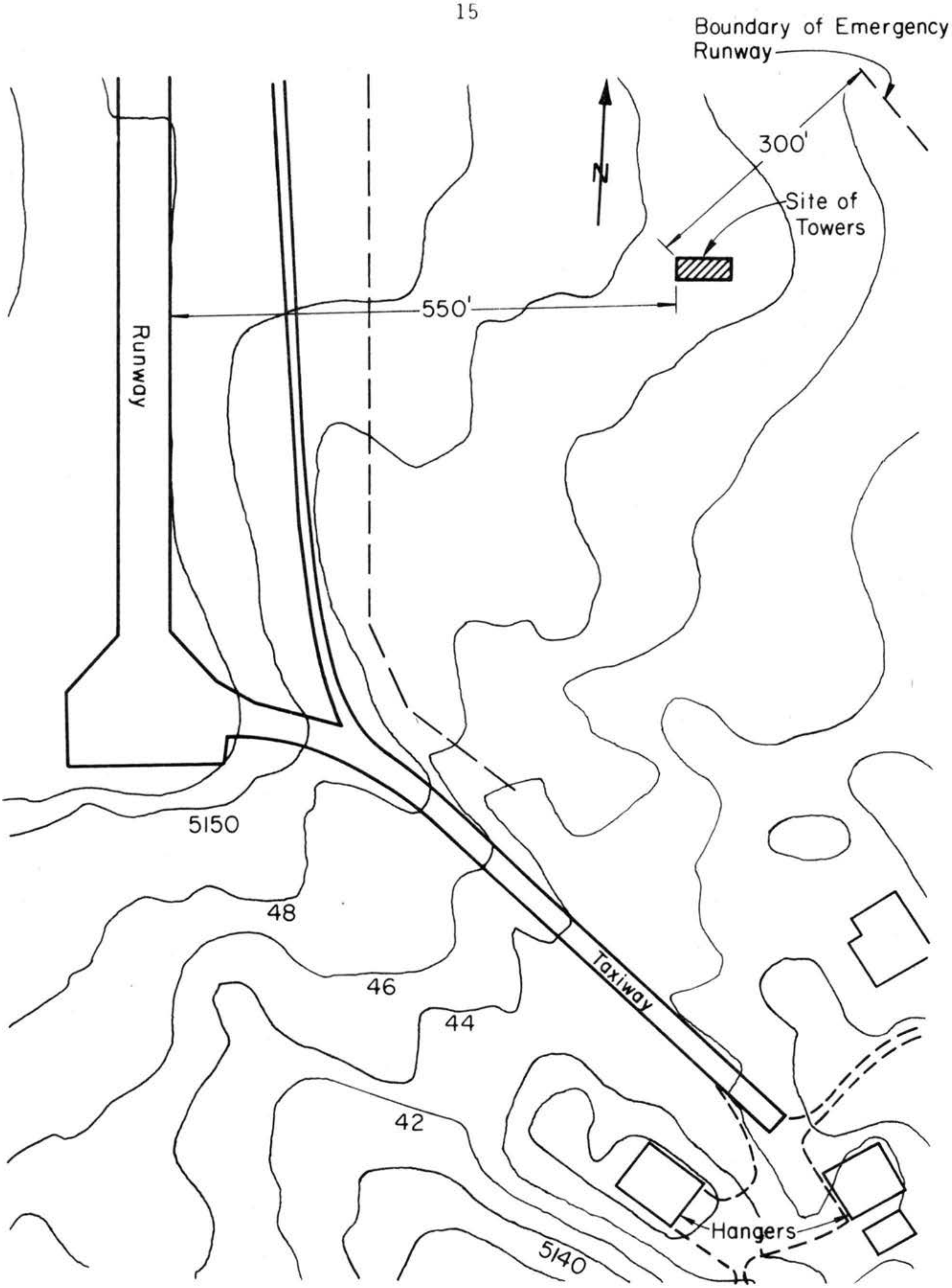


Figure 9. Field site at Christman Field.

The site facilities included two towers and two trailer vans to house the instruments and the LDV system. The arrangement shown in Figure 10 was to provide clear wind fields to the north and south. As winds seldom blow from the east, the instrument vans were located so as to cause as little interference as possible to the wind field.

The 60-ft high tower was used to mount the wind profiling anemometers. The 40-ft tower was used to mount mirrors to direct the laser beam and also to mount the comparison instruments, a climet anemometer and wind vane, and a hot wire for turbulence measurements. Photographs of the established arrangement are shown in Figures 11 and 12.

INSTRUMENTATION

The arrangement of the various instruments in the laser instrument van is shown in the photograph of Figure 13. The total instrumentation for data taking and recording included the following:

Spectrum analyzer - The function and description of the spectrum analyzer was given in a previous section.

Frequency tracker - This instrument was also discussed in the earlier section.

Wide band frequency generator - A frequency generator of MHz range was used to establish a calibration point for the spectrum analyzer. Depending upon the prevailing wind speed, a calibration frequency was selected near the extreme of the wind speed range and the scan width of the spectrum analyzer was selected to contain this calibration frequency.

Frequency counter - A frequency counter was used to determine the calibration frequency.

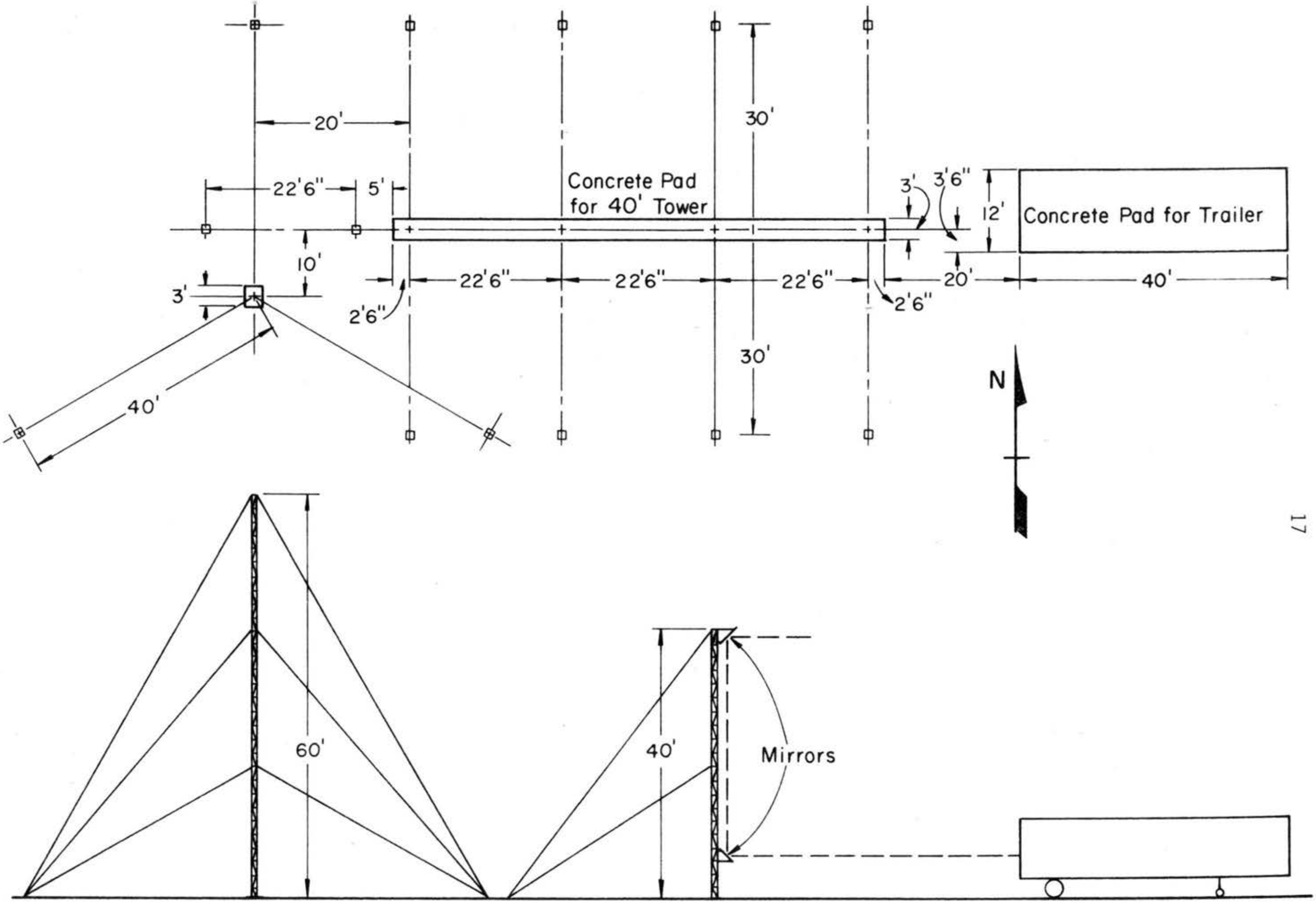


Figure 10. Site arrangement for towers and instrument van.

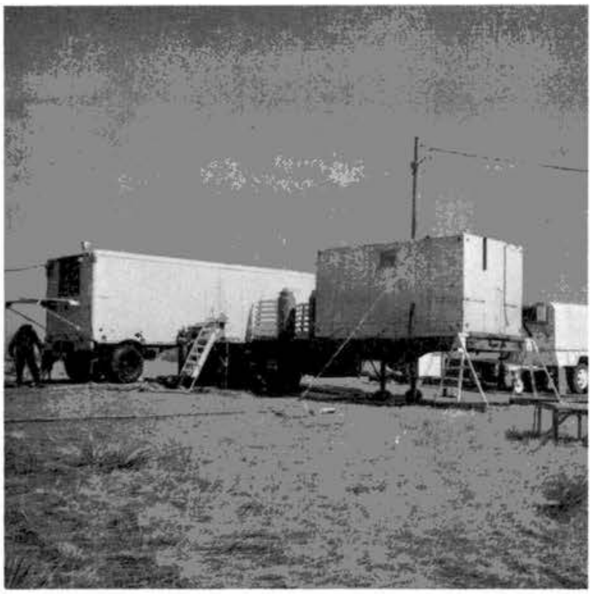


Figure 11. Instrument vans at test facility.

Figure 12. Towers at test facility. Profile tower is at left.

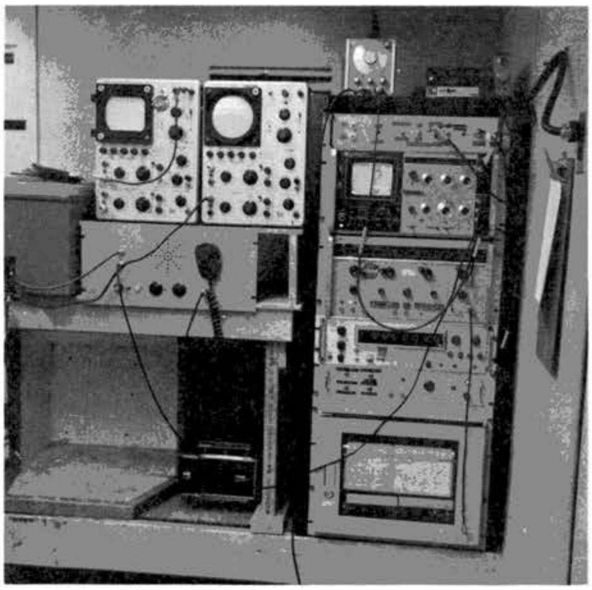


Figure 13. Instrument arrangement inside laser van.

Function generator - A stable function generator was used to drive the sweep of the spectrum analyzer IF at a rate consistent with the spectrum analyzer scan time. A finite sweep time and "flyback" is involved. A given combination of sweep duration and scan width has its optimum IF filter bandwidth. A table of sample rates for various scan time settings is given in Table 1, and bandwidths as a function of scan width and scan time is given in Table 2. These tables were reproduced from the LMSC report No. D162840 describing the operating procedures of the LDV system.

Mirror position indicator and drive - The upper mirror on the 40-ft tower had a motor drive to rotate the mirror about its vertical axis. This permitted orientation of the laser beam into nominal alignment with the wind direction. The position of the mirror was indicated by a 357 degree potentiometer. There were 3 degrees of ambiguity from 357 degrees to 360 (zero) degrees. The position pot of the mirror was oriented so that zero was due east.

Climet wind translator - The translator presented wind direction and speed as sensed by the cup anemometer and wind direction sensor into recordable analog signals. The wind direction sensor was oriented so that zero output coincided with due east. The analog signals were then monitored on a dual channel strip chart recorder.

FM tape recorders - Two 14 channel FM tape recorders were used to record the analog signals, one a CP-100 Ampex unit and the second an FR-1300 Ampex recorder.

Temperature sensor - A standard bridge and amplifier circuity was constructed for this study to measure the deviations in temperature of the various thermistors from a reference unit.

TABLE 1. MAXIMUM SAMPLE RATES FOR SELECTED SCAN TIMES

Spectrum Analyzer Scan Time (Millisec/cm)	Maximum Sample Rate (Hz)	External SYNC Period (sec)
0.5	165	0.006
1.0	69	0.0145
2.0	40	0.025
5.0	18.2	0.055
10.0	5.0	0.200
20.0	3.3	0.303

TABLE 2. MINIMUM BANDWIDTHS IN kHz FOR COMBINATIONS OF SCANWIDTH AND SCAN TIME

Scan	Scan Time, Millisec/Division					
Width/cm	1.0	2.0	5.0	10.0	20.0	50.0
0.02 kHz	0.3	0.3	0.1	0.1	0.1	0.1
0.05 kHz			0.3			
0.1 kHz	1.0			0.3		
0.2 kHz		1.0			0.3	
0.5 kHz	3.0		1.0			0.3
1.0 kHz		3.0		1.0		
2.0 kHz			3.0		1.0	
5.0 kHz	10.0			3.0		1.0
10.0 kHz		10.0			3.0	
20.0 kHz	30.0		10.0			3.0
0.05 MHz		30.0		10.0		
0.1 MHz	100.0		30.0		10.0	
0.2 MHz		100.0		30.0		10.0
0.5 MHz	300.0		100.0		30.0	
1.0 MHz		300.0		100.0		30.0
2.0 MHz	---		300.0		100.0	
5.0 MHz	---	---		300.0		100.0
10.0 MHz	---	---			300.0	

Hot-wire anemometer - A constant temperature hot-wire anemometer was used to measure the atmospheric turbulence. A 100-ft long cable was used for the probe and a cable capacitance compensator was used for the long-length cable. The hot wires were calibrated with the extra cable and compensator.

Time code generator - A time code generator in IRIG B format was used to synchronize the two tape recorders. Usually the times were synchronized with the National Bureau of Standards time broadcasts.

RECORDING OF TEST DATA

There were in all 26 separate pieces of continuous information desired for each test. Two analog 14 channel FM recorders were needed. However, two recorders were not available for all tests and some information was therefore sacrificed. The sample data recording sheet shown in Figure 14 indicates the data recorded on each channel of each recorder. They were arranged in such a way that temperature and humidity data were sacrificed when the second recorder was unavailable.

The data can be grouped into the following sets. On the 60-ft tower, six levels of wind speed were obtained to establish the vertical profile of the wind field in which comparison data were taken. These were grouped in the CP-100 Ampex recorder. Also, on the same tower, there were six levels of temperature measurements to determine the temperature profile and four levels of wet bulb temperatures to establish the humidity profile. These were grouped on the FR-1300 Ampex recorder. On the 40-ft tower the comparison instruments, the cup anemometer, the wind vane, and the hot wire were mounted. These data together with the

Test No. ______

ATMOSPHERIC LASER DOPPLER VELOCIMETER PERFORMANCE VERIFICATION

☐ NASA-MSFC Field Test Site, Huntsville, Alabama
☐ Airport Field Test Site, Foothills Campus, Colorado State Univ., Ft. Collins
☐ Other

Test Conducted Between ___ a.m./p.m. and ___ a.m./p.m. on ______________ (date)

METEOROLOGICAL DATA
Air Pollution Index: __________. Visibility: ☐ Good; ☐ Fair; ☐ Poor
Sky Condition: ☐ Clear; ☐ Light Clouds; ☐ Medium Clouds; ☐ Heavy Overcast
Temperature ______ °F; Relative Humidity _____ % or Dew Point _____ °F;
Barometric Pressure _____ mb; Anemometer(s) Locale ______________________

Time into Test (min)	0	15	30	45	60	
Mean Wind Speed (knots, mph, ft/sec)						
Mean Wind Direction (deg wrt north)						
Laser Coolant Temperature (°F)						

General Weather Conditions (frontal presence, rain in past 12 hours, etc.):

____

____

OPTICAL CONFIGURATION
Mirror Orientation ____ deg (wrt north)
Telescope mirror to lower tower mirror distance: ____ ft ____ in.
Distance between top and bottom mirrors on tower: ____ ft ____ in.
Total distance from telescope mirror to focus vol: ____ ft
Homodyne configuration: ☐ Mach Zehnder; ☐ internal cavity
Laser power into telescope: ____ watts; Power at focal volume: ____ watts; He dewar check ☐ .
Telescope mirror size: ____ in. diam.; Lens focal length: ____ in; Detector type: _____ .

Comments: ____

____

____

SPECTRUM ANALYZER/AVERAGER DATA
Sweep Rate: ____ ms/cm; Sample Rate: ____ samples/sec.
Number of sweeps averaged per sample: _____ .
Frequency dispersion: _____ MHz/cm. Filter bandwidth: _____ kHz; Bandwidth: ____ kHz.

Other: ____

____

____

FM RECORDER DATA - MODEL CP-100 AMPEX

Label Tape Reel with Test No. __
Tape No. _____; Tape Speed ____ ips; Response ____ Hz.

Cha. No.	Contents
1	Voice: Test ident. etc.
2	Spect. analy. sync. pulse:
3	Spect. analy. Y out. Freq. disp.
4	Wind dir. Climet volts ⟶ °wrt
5	Wind speed Climet volts ⟶ fps
6	Mirror azimuth volts ⟶ wrt
7	Hot wire anemometer
8	Fixed tower data - Wind sp. level 1
9	Wind sp. level 2
10	Wind sp. level 3
11	Wind sp. level 4
12	Wind sp. level 5
13	Wind sp. level 6
14	Time code ident. IRIG B

FM RECORDER DATA - MODEL FR-1300 AMPEX

Label Tape Reel with Test No. __
Tape No. ____; Tape Speed ____ ips; Response ____ Hz.

Cha. No.	Contents (Fixed Tower Data)
1	Therm., ref. amb. temp. level 2
2	Therm., diff. dry air level 3
3	Therm., diff. dry air level 1
4	Therm., diff. wet bulb level 1
5	Therm., diff. dry air level 4
6	Therm., diff. wet bulb level 4
7	Therm., diff. dry air level 5
8	Therm., diff. wet bulb level 5
9	Therm., diff. dry air level 6
10	Therm., diff. wet bulb level 6
11	Wind direction level 5
12	
13	
14	Time code ident. IRIG B
Aux	Voice: Test ident. etc.

AUDIO RECORD

☐ Test Identification Number; ☐ Spectrum Analyzer settings (sweep rate, number samples averaged, etc.); ☐ Mean Wind; ☐ Distance from telescope mirror to focal volume; ☐ Visual quality of signal; and ☐ Problems and other comments.

____
(Signed) Test Engineer

Figure 14. Sample data sheet.

spectrum analyzer signal and appurtenant data were grouped into the CP-100 recorder. On one channel of each recorder there was an IRIG B time code for referencing the two sets of data to corresponding times. A voice channel (direct record) was reserved for verbal description of conditions and problems which occurred during a test.

Data with a frequency tracker were taken during a period when the second tape recorder was unavailable. Since two additional channels were required to record the signals from the tracker, two levels of wind speed data were sacrificed on the CP-100. These were levels 2 and 4.

TEST PROCEDURE

Pre-Test Preparation

Preparations for recording one-hour of continuous wind data and associated documentation was elaborate and time-consuming. For any given test, or attempted test, the following routine was necessary.

Cooling the Ge-Hg detector - The Ge-Hg detector was pre-cooled with liquid nitrogen for a period of about one hour before filling with liquid helium. This procedure was followed primarily to conserve liquid helium, which is comparatively many times more expensive than liquid nitrogen. Just prior to data-taking, after all preparations were completed, the dewar of the detector was filled with liquid helium.

Optics alignment - Before each test, alignment of the optics was necessary. A specific alignment procedure progressing outward from the laser to the tower was necessary. Although the beam splitters and mirrors did not require frequent adjustment, the optical beam on which the focusing mirror was mounted required frequent adjustment. As the scattered radiation was redirected back into the laser, slight misalignment of the optical axis caused poor to no heterodyning, hence weak or no Doppler detection. Since alignment of the focusing lense mount is coupled with the diagonal and the schlieren mirror, a sequence of trial and readjustment was usually necessary.

After the optical beam was adjusted, the diagonal required minute adjustment to center the diverging radiation on the schlieren mirror. The schlieren mirror in turn required adjustment to direct the converging beam through the end of the 9-ft long tube. Thereafter, the entire mounting table required movement to center the beam on the lower external mirror near the base of the 40-ft tower. If the optics were bumped out of alignment during this process, then the entire procedure was restarted.

Once the laser beam was centered on the lower mirror, then the lower mirror was adjusted to center the beam on the upper one, and finally the upper mirror was rotated to direct the beam as closely as possible either directly into the prevailing wind direction or downwind along the wind direction, checking also to see that the beam was parallel with the ground. To establish the latter, an identification mark on the adjacent 60-ft tower was used to place the line of sight parallel to the ground, hence the axis of the laser beam was in the horizontal plane of the mean wind.

Profile tower - The thermistors on the 60-ft tower were arranged in a radiation shield, with a suction pump arranged to draw 2 ft/sec air velocity over the "dry bulb" thermistor and 30 ft/sec over the "wet bulb" thermistors. Distilled water was forced up the tower by air pressure into water wells with wicks leading to the "wet bulb" thermistors.

These thermistors were checked before each test and wicks were prewetted to insure that the distilled water would be drawn up from the wells.

Hot-wire anemometer - The hot-wire anemometer which was dismounted during a non-test period was remounted. The wire was placed in a vertical axis and the probe was oriented toward the wind and in a location such that there was no interference from the mirror, the cup anemometer or the tower itself.

Pre-Test Calibration

Tape recorder - The FM record amplifiers of the tape recorder are subject to slight deviations from calibrated conditions from day to day. To account for these deviations, a five-level DC signal was provided as a calibration of tape-recorded (and playback) voltage against a "true" voltage registered by a calibrated digital voltmeter (DVM). Since in the data set, a continuous square-wave signal was recorded, the calibration set did not include a sinusoidal signal of known rms value.

Climet anemometers - Both climet anemometer translators were calibrated for zero and full scale 1 volt outputs and recorded on the tape recorder. Prior to mounting the anemometers in the towers, all cup anemometers as well as the hot wires were calibrated in the Colorado State University wind tunnel against a pitot probe of known performance. Calibrations were performed twice, in February and June 1971.

Mirror position - The mirror position, with zero oriented directly east for convenience, was calibrated for zero and full scale ouput, with the assumption of linearity with angular position. Since the position was indicated by a potentiometer, the assumption seems justified.

Spectrum analyzer - Proper settings of the spectrum analyzer controls were established consistent with the prevailing wind speeds.

The sweep of the spectrum analyzer was triggered by an external square wave from a stable function generator by a change from negative to positive voltage. A known deviation frequency was input to the spectrum analyzer and the resultant signal from the IF output was recorded as the frequency band was swept. This calibration thus provided the reference for determining velocity from the Doppler shifted frequency of the backscattered radiation.

Noise calibration - The final pre-test calibration was made of the background noise emitted from the detector. With the detector dewar charged with liquid helium, and the main laser beam to the telescope blocked, the output signal from the detector which consisted only of noise was recorded.

Data Recording

After completing the pre-test preparations and calibrations, data were recorded on the tape recorders for nominal periods of one hour duration. Constant monitoring of the data was provided, and instrument adjustments when necessary were properly recorded as to time and nature.

The turbulence range of interest extended only to a maximum of 5 Hz, thus the CP-100 tape recorder was operated at 7½ inches per second (ips) and the FR-1300 recorder at 1-7/8 ips. The higher speed of the CP-100 recorder was necessary to record the Doppler signals from the spectrum analyzer. At 7½ ips the recorder amplifiers are responsive to 2.5 KHz.

Anomalies in the data noticed were recorded on a voice channel (direct record) of the tape recorder, as well as on the data record sheet (Figure 14).

DATA REDUCTION PROCEDURE

All data for this investigation were analyzed digitally, the digitizing being done in prescribed sets in simultaneous sample and hold mode at the NASA-MSFC computer center. The digitized data were analyzed at the Colorado State University computer center.

Selection of Digitizing Rates

The turbulent frequencies of interest in this study are less than 5 Hz, thus the digitizing frequency should be at 10 samples per second, and also, because in general the recorded information should be related to the same instants of time, a simultaneous sample and hold mode was used in digitizing. The analog signals were filtered at 5 Hz (real-time base).

The scan rate of the spectrum analyzer for Doppler frequencies was 16 Hz. Since the Nyquist frequency is equal to one-half sampling frequency,

$$f_N = \frac{f_D}{2}$$

the highest frequency information contained in the recorded signal is 8 Hz. However, the usual criterion of digitizing rate to obtain this frequency information does not apply. The objective in data reduction was to determine the location (time base) of the Doppler signal with reference to zero frequency, hence of Doppler frequency and of wind velocity. The bandwidth and resolution of the spectrum analyzer determines the nature of the Doppler signal. If we view the peak signal in the bandwidth as depicting the mean velocity in the prescribed resolution interval, then the digitizing rate of the Doppler signal is independent of the spectrum analyzer settings. Thus with a view to maximizing the frequency resolution (of the peak) in a given sweep, a choice of 250 points per sweep was made. The choice of this digitizing rate does however affect the total quantity of digitized data. Two channels of information, the external function generator and the IF output of the spectrum analyzer, were digitized at this higher rate, multiplexed on digital magnetic tape in binary format. The sampling rate for these channels was thus 4 KHz/channel and the data were filtered at 2 KHz.

Multiplexed Data Groups

The 26 channels of analog information were digitized in three separate groups.

Group 1 - The sweep signal (square wave) and the spectrum analyzer IF(y) output were multiplexed and digitized at a rate of 4 KHz/channel.

Group 2 - The climet anemometer and wind direction sensor, the mirror position, the hot wire output and six levels of wind speeds on the profile tower were multiplexed and digitized at a rate of 10 Hz/channel.

Group 3 - The ten channels of thermistor data were multiplexed and digitized at a rate of 10 Hz/channel.

The remaining four channels of voice, time codes and wind direction on the profile tower were not digitized and were retained for reference. The time code information was of course used to identify the regions of the analog tape which were digitized.

Data Format

The A/D converter used at NASA/MSFC generated data words of 10 bits plus sign. The packed format of the multiplexed data therefore were written in groups of 11 bits. The CDC 6400 at Colorado State University is a 60-bit word machine, thus some tape reading problems were presented with the original format of the generated data tape. In order to reduce the reading problem, the original data tapes were reformated to give data words which were 11 bits plus sign, or 12 bit words where a zero was inserted into the most significant bit. The packed 12-bit data words were thus conveniently separated and sorted from the 60-bit computer word.

The data included a record of header information at the beginning of each data set, and a 24-bit time word at the beginning of each data record. This time word is a reference digitizing time, and relates to real time in accordance with the ratio of record to playback tape speed. However, for records of the order of 60 minutes real time duration, the time word (expressed in milliseconds) becomes excessively large. Thus the digitizing clock which recycles after 100 seconds requires accounting of the cycles to convert digitizing time to real time as well as the ratio of record to playback speeds.

Data Reduction

Laser Doppler signals - The bulk of data reduction involved the Doppler signals. The view adopted in computer program formulation was to devise a general, automatic program. This was successful to a degree, however sufficient problems with data anomalies were encountered that some initialization is necessary. Considerable time was spent in developing this feature of a data reduction program. In retrospect, perhaps less automatic, sequential programs would be more economical in terms of total effort. The flow chart for the program is shown in Figure 15 and a listing is given in Appendix A.

The essential technique is as follows: Data from Group 1 (identified above), and the first channel of the multiplexed data of Group 2, are necessary to convert the spectrum analyzer data to wind speed. If the mirror direction varies in the data period, that information is also required.

The cup anemometer wind speed, the hot wire data and the profile information can be processed separately, but because the two groups of data were arranged on different tapes and had to be read in "simultaneously" to analyze the Doppler signal, the program included processing of these data at the same time. It should be noted here that several alternative methods were recognized from the outset, and a one-pass automatic program seemed feasible and most desirable. Ultimately the profile data program was separated from the others and analyzed in a separate pass. The flow chart in Figure 15 reflects this variation to the original technique.

The program first determines the input-output calibration of DC voltage. This calibration enables conversion of such data as wind and mirror directions, cup anemometer speeds and hot wire turbulence velocities from tape output voltage to true voltage hence to the physical quantities. The next step in the analysis is to determine the calibration Doppler frequency. That is, the known frequency input is identified in the time space (number of points) from zero frequency, and since velocity is linear with Doppler frequency, then calibration is obtained for the velocity component along the laser axis. In order to

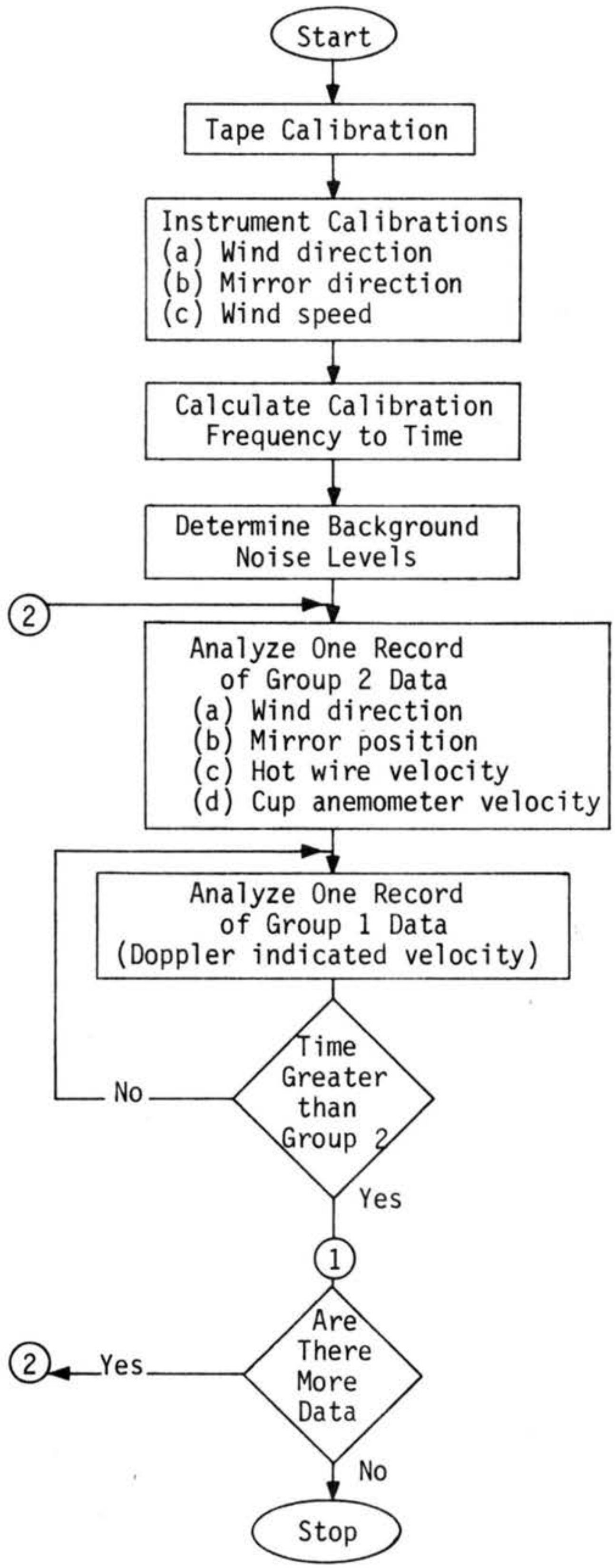


Figure 15. Simplified Flow Chart of Doppler Data Reduction.

distinguish the Doppler "peak" from the background noise, the noise calibration established the noise level across the entire frequency band of the spectrum analyzer. In the program the S/N ratio is a variable and may be set at any level compatible with the recorded Doppler signal.

The first step in the data analysis is to read in one record from the multiplexed Group 2 data. Each digital value is converted to velocity, and reference times for each value are calculated. The velocities and reference times are stored. The cup and hot wire data are digitized at identical times, thus one reference time serves both channels of information. Means and variances are calculated. Wind direction voltages are averaged for 10 seconds (one record) and converted to angle with respect to the laser beam. The value is temporarily stored. The mirror azimuth (direction) is averaged and checked. If no change occurred (i.e. the mirror was not rotated) the information is redundant and discarded.

The first record of Group 1 is then read in. Each spectrum analyzer scan, approximately 250 points, is searched for zero frequency (the change in voltage of the square wave from negative to positive identifies the beginning of the sweep) and the Doppler signal. The reference time for Doppler-converted velocity is referenced to the beginning of the sweep. Successive sweeps and time words at the beginning of each record references the true time of the calculated Doppler-measured velocity. The first identifiable Doppler peak is accepted as the measured velocity. To determine the peak value, comparison is made to successive points, and if the signal level (voltage) drops, the previous point is accepted as the Doppler frequency. It is possible that in a given sweep there is no

Doppler signal (signal dropout), in that event, the velocity determined in the previous sweep is recorded. The Doppler-indicated velocity is then converted to wind velocity by the 10-second average angle of the wind direction with respect to the laser beam axis (mirror direction).

There are 3000 data points (2 channels) in each record of the Group 1 data. This corresponds to 6 sweeps of the spectrum analyzer and 0.375 second in terms of real time. Successive records of Group 1 are read in and analyzed until the real time reference period exceeds the real time period of the data read in from the Group 2 data. Additional Group 2 data are then read and reduced, and the process repeated.

The stored values of velocities and reference times are periodically purged from storage and written on a magnetic tape. Thus the entire test record is converted to velocity-time history with the same reference times for the cup anemometer and hot-wire data, but a different reference time for the Doppler-indicated velocities.

The generated velocity-time history tape is then reprocessed to obtain the statistical characteristics of the turbulent wind data. These characteristics are the mean, variance (standard deviation), probability density and spectral densities (power spectrum).

Velocity profiles - The velocity profiles are calculated in a straight forward manner, using the other six channels of data in Group 2. Only the mean values are of concern, and ten-minute average velocities are calculated for each anemometer. The calibration data for voltages, and the prior wind-tunnel calibrations, are all that are required. A program listing is given in Appendix A.

Temperature profiles - Temperature and humidity profiles likewise, are relatively straightforward requiring manufacturer's calibration data for the thermistors and conversion of average tape voltage to true voltage. The resistances are calculated from a standard bridge equation, hence temperatures are determined. The program listing is given in Appendix A.

EXPERIMENTAL RESULTS AND DISCUSSION

Calibrations

Climet anemometers - Calibration curves of the climet anemometer, Series No. 828, are shown in Figure 16. The calibration was performed in a wind tunnel with the translator set for 1 volt output at 1896 Hz input (signal frequency generated by the cup) for the 60 scale setting on the translator. Ordinarily, the translator is adjusted to output 1 volt for specific input frequencies on each scale. However, for purpose of this calibration, adjustment was made for 1 volt output on the 60 scale only (any frequency would have served as well) and outputs read from both 30 and 60 scales. In setting the translator during an experiment, therefore, adjustment was always made only for the 60 scale. The output is linear with velocity as seen in the figure.

The CP-100 tape recorder has a low input impedance, causing a loading problem with virtually all the instruments connected to it. Thus the cup anemometers and hot wires were calibrated with the outputs connected to the tape recorder.

Hot-wire anemometer - A typical calibration curve for the hot-wire anemometer is shown in Figure 17. For purpose of this investigation, the King's law relationship is shown, and it is seen that in the region

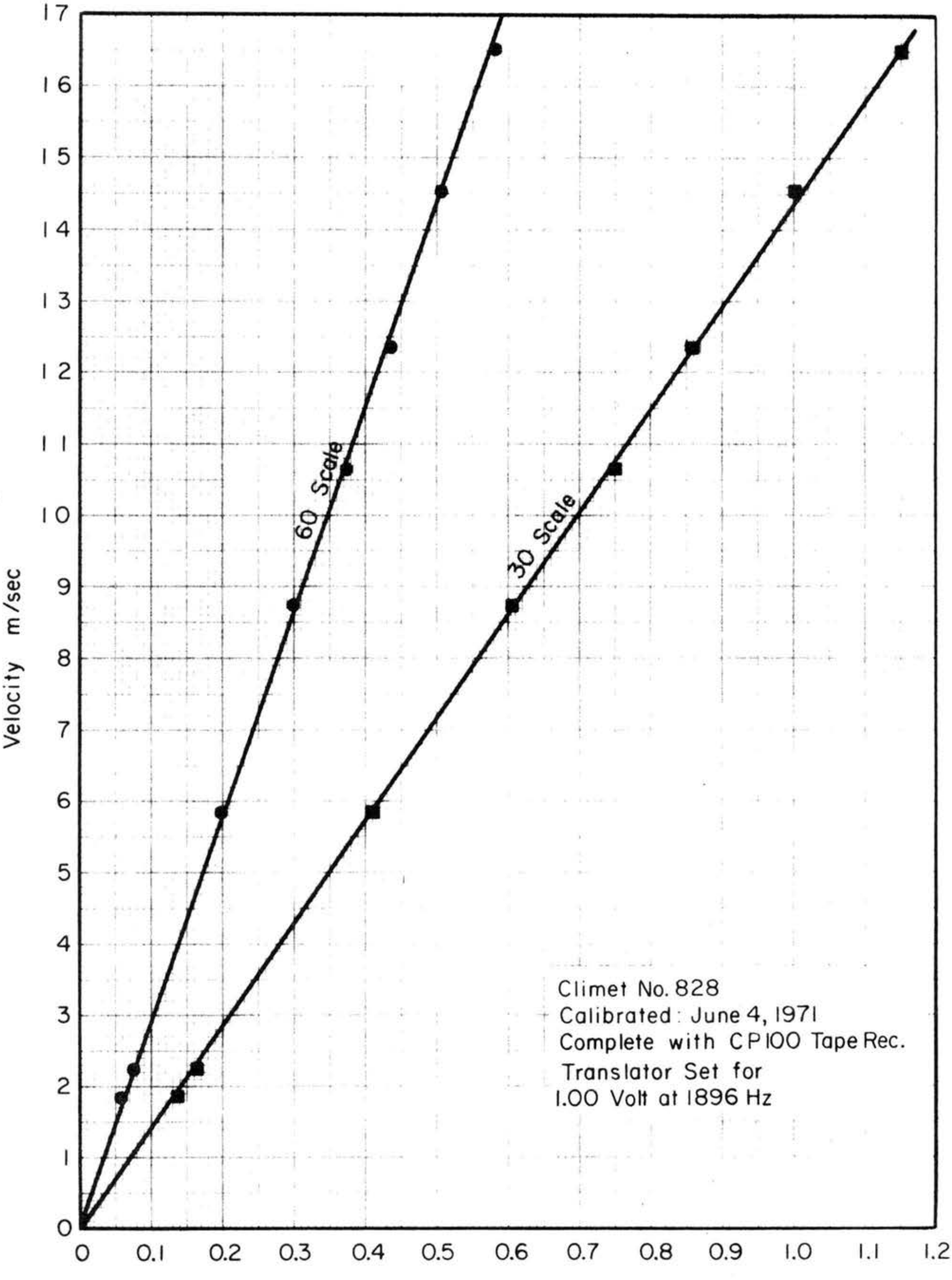


Figure 16. Calibration curves for climet anemometer.

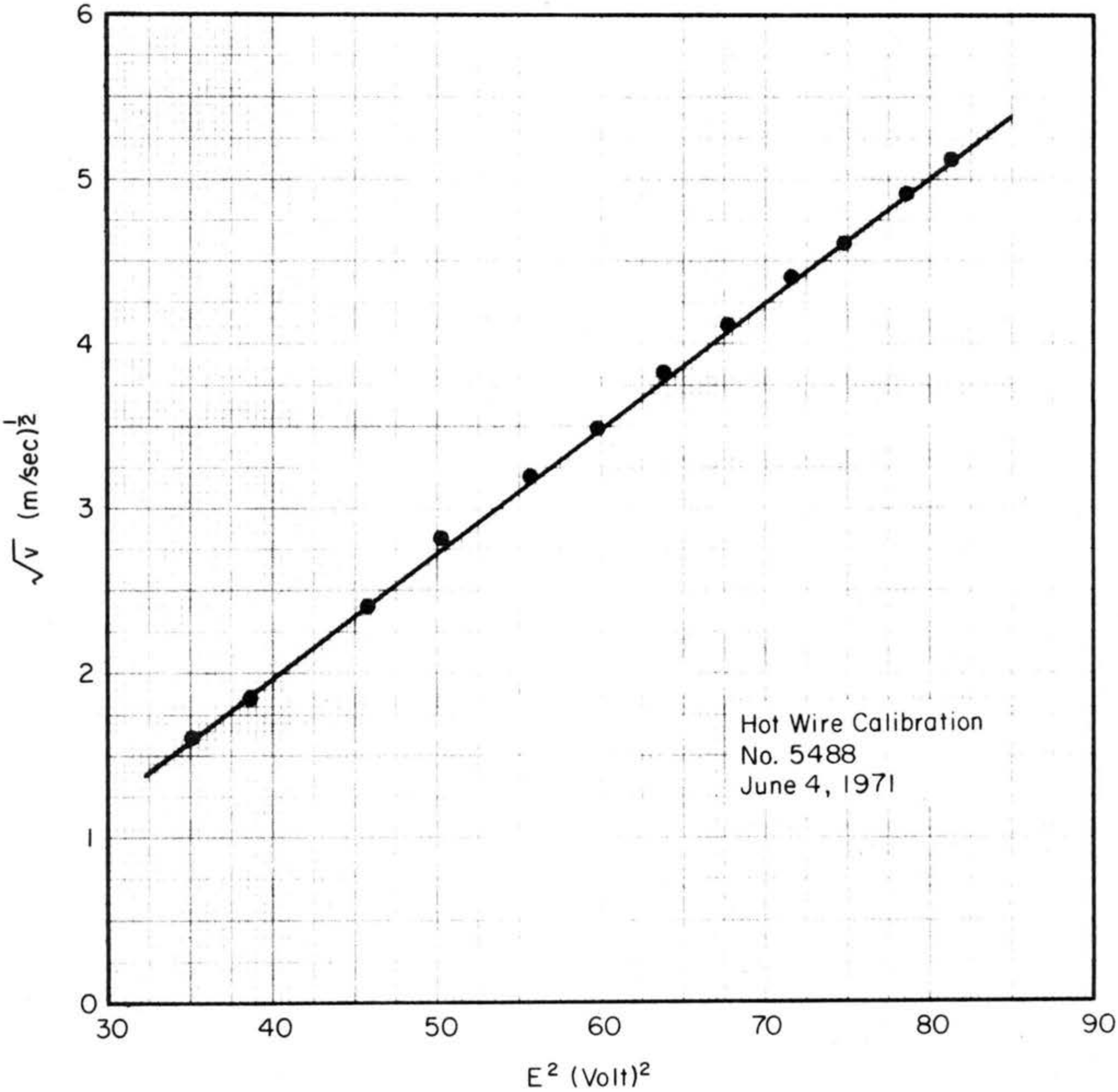


Figure 17. Hot-wire calibration curve

of interest, the curve was linear. A linearizer was not used with the anemometer. Instead, each digitized data point was converted to actual voltage and velocity calculated from the calibration.

Measurements of Run 50801 (May 8, 1971)

The data for this test were taken from 1:48 pm to 2:45 pm, covering a period of approximately one hour. At the beginning of the test the wind was blowing from the south-southeast (30 degrees east from south) which gradually changed to south-southwest (15 degrees west from south) by the end of the test period. The wind speed was reasonably constant at about 4 m/sec (9 mph) throughout the test period. Particle counts in the atmosphere were not available for this test; however, with the prevailing south wind, the pollution from Denver was evident as a blue haze along the horizon. This was also reflected in the strength of the Doppler signals on the spectrum analyzer.

Velocity profiles - The velocity profiles for successive 10-minute periods throughout the test are shown on Figure 18. The velocity profiles were logarithmic as expected; however, the slope of the profiles differ, indicating that the effects of accelerating and decelerating winds (gusts) are reflected in the profiles. It will be seen in the time traces of velocities that the fluctuations are of the same order of magnitude as the means, and the mean values change with time. The analysis to establish the profiles assumes piece-wise stationarity.

Spectrum analyzer settings - The following settings were made on the spectrum analyzer:

Sweep rate:	5 ms/cm
Sample rate:	16 sweeps/sec

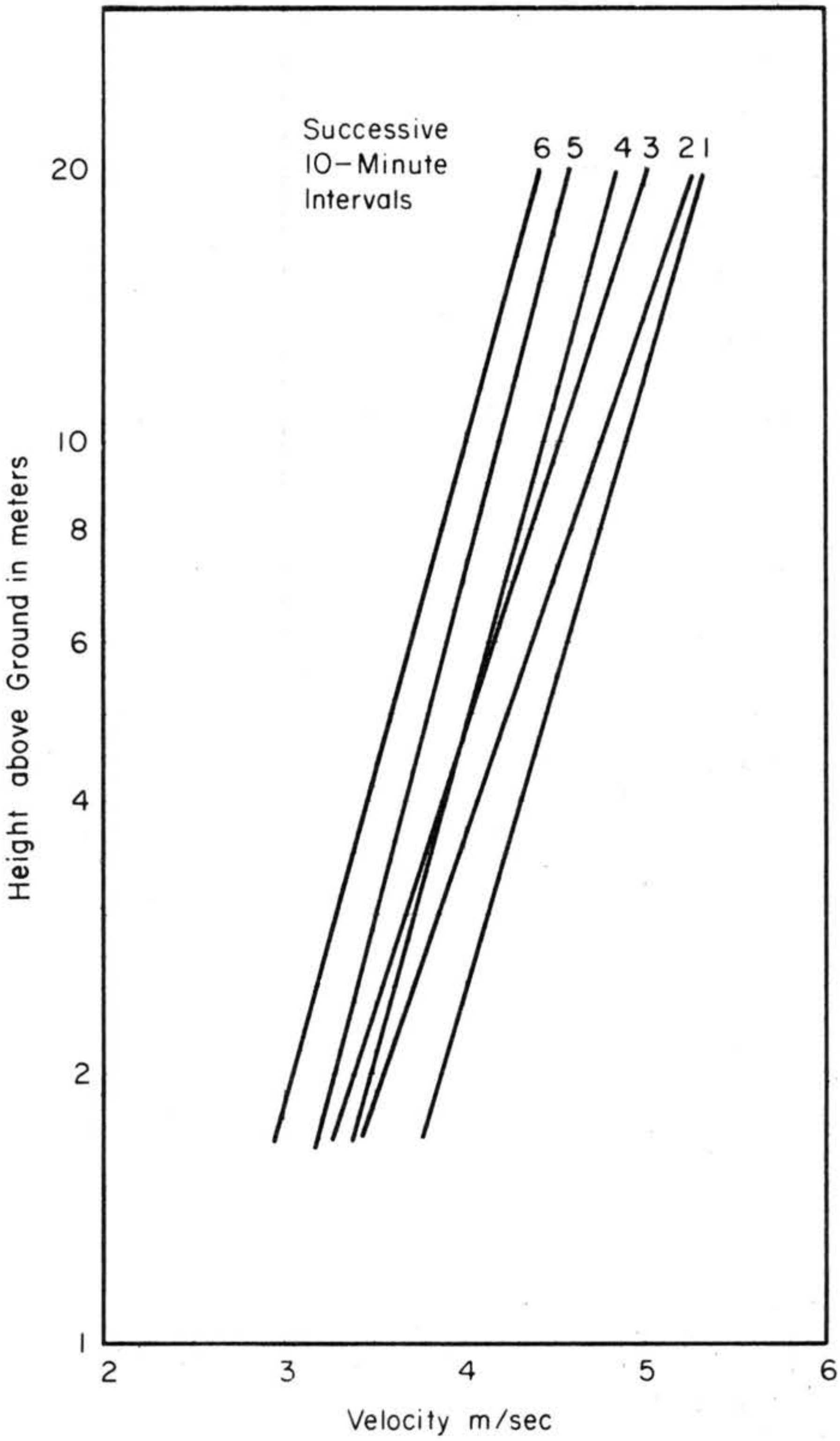


Figure 18. Velocity profiles for test period 50801.

Frequency Dispersion: 0.2 MHz/cm

Filter Bandwidth: 10 KHz

Bandwidth: 30 KHz

The calibration frequency was 1.007 MHz (5.34 m/sec) which is pictured in Figure 19. The noise level from the detector is shown in Figure 20. The photograph is the oscilloscope trace from playback (at record time) of the recorded signal on the CP-100. The signal is inverted to avoid confusion with the square wave shown at the top part of the picture. The vertical scale is 200 mv/cm.

Typical Doppler signals are shown in Figures 21 and 22. As noted, the S/N ratio is large, but the spectral bandwidth is also large. Peaks in the signal of the kind shown in Figure 21 are relatively easy to determine; however, multiple peaks are evident in Figure 22. In these instances, the first largest peak is detected, and the others ignored. There were undoubtedly particles of different sizes in the focal region with different angularity with respect to the laser beam axis which cause the multiple peaks in a given sweep.

Velocity time traces - Time traces of velocity from the cup anemometer, hot wire and the LDV, for two consecutive 4.26-minute periods are shown in Figures 23 and 24. Mean velocities for each 4.26-minute interval have been subtracted; the fluctuations thus are referenced to zero for each plot.

As seen in these traces, there is reasonable conformance between the cup anemometer, hot wire and LDV outputs. It should be noted here that the cup anemometer was at a level 11.3 meters above ground, the hot wire was 0.3 meters below the cup level and the laser beam axis was at the same level as the hot wire although the focal region was 3 meters

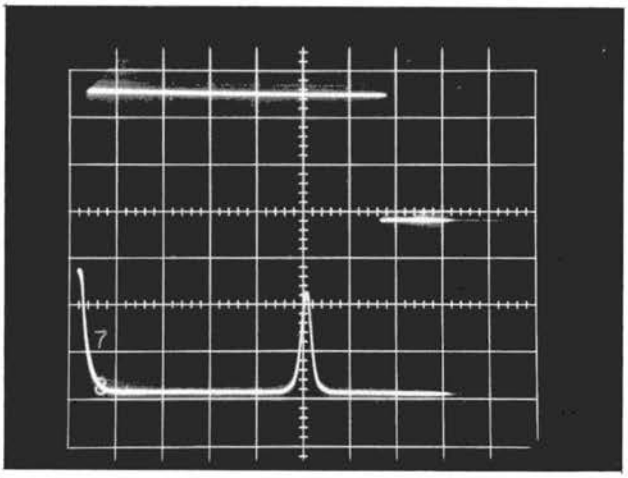


Figure 19. Calibration frequency 1.007 MHz. Test 50801

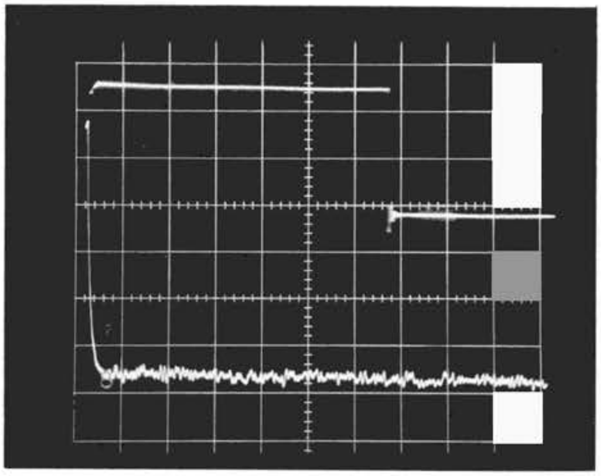


Figure 20. Detector noise calibration. Vertical scale is 200 mv/cm. Test 50801

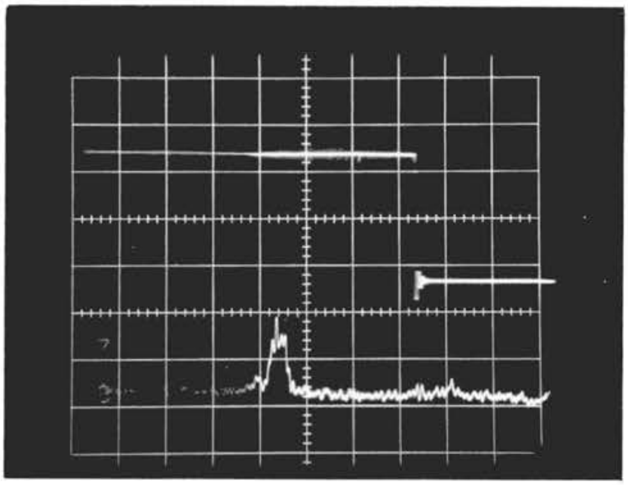


Figure 21. Sample Doppler signal. Test 50801

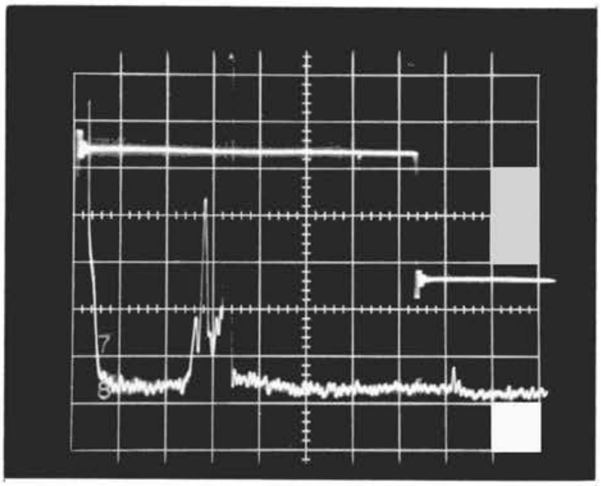


Figure 22. Sample Doppler signal. Test 50801

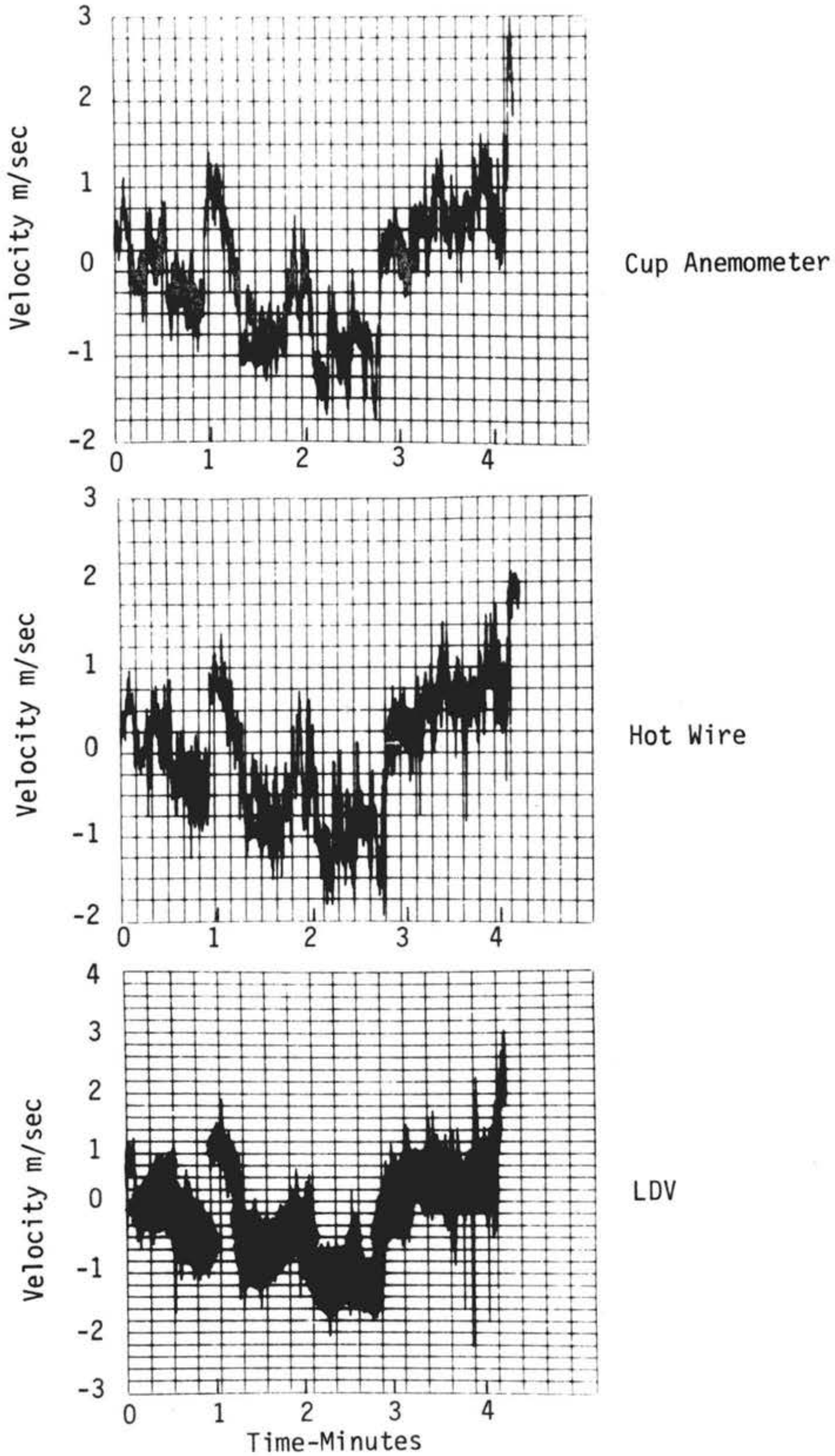


Figure 23. Time traces of wind velocity.
Test 50801, Interval 1
(For means and variances see Table 3)

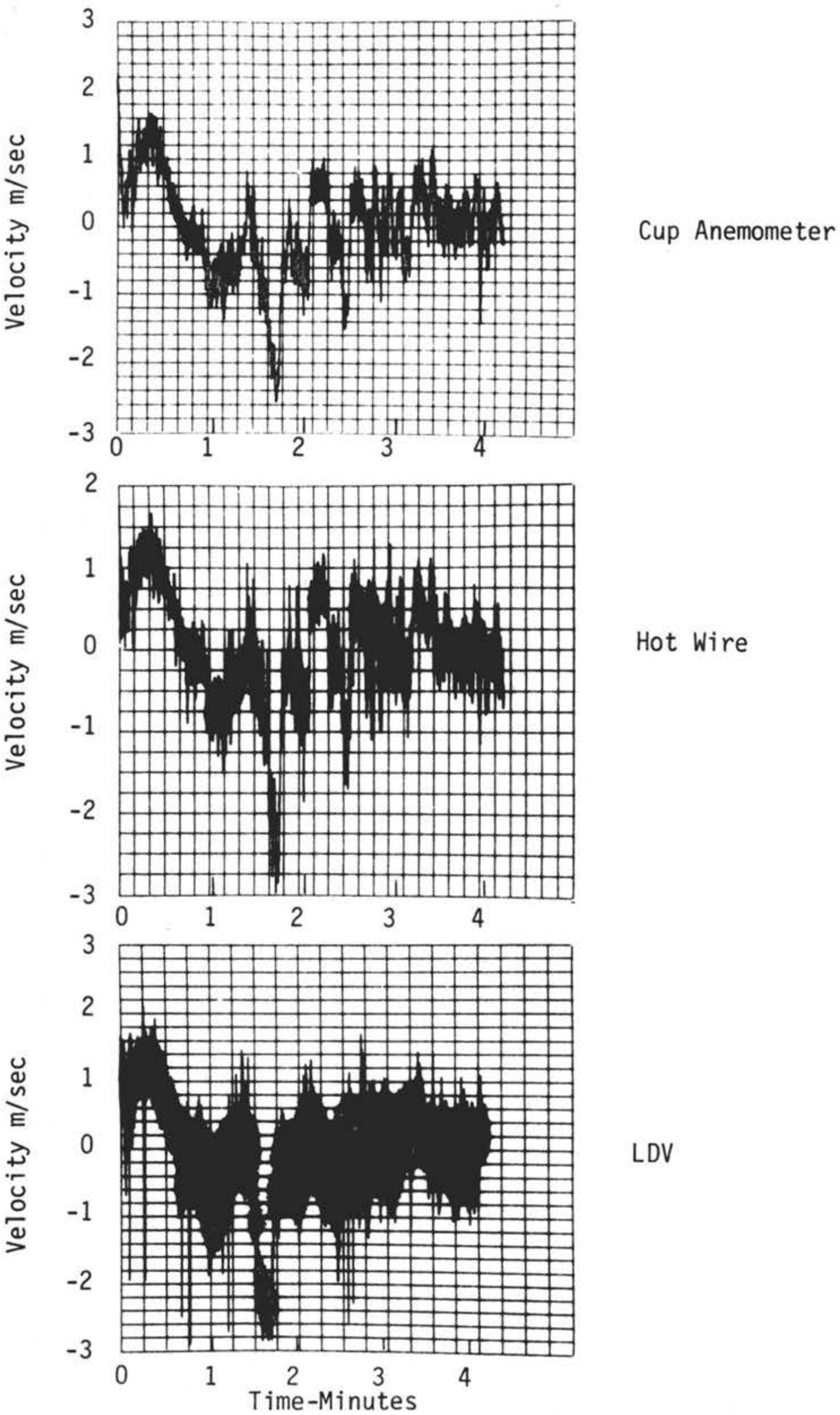


Figure 24. Time traces of wind velocity
Test 50801, Interval 2
(For means and variances see Table 3)

farther upwind. It should be noticed in making visual comparisons that the vertical scales are different for the traces.

Means and variances - The means and variances from a 34-minute interval of the total record were analyzed and are shown in Table 3. The choice of a 34-minute period was based largely on the limitations of the spectral analysis program. This was also a sufficiently large period to reflect a reasonable confidence interval for the spectral densities.

TABLE 3. MEANS AND VARIANCES FOR TEST 50801

4.26-Minute Intervals	Mean Velocities m/sec			Variances $(m/sec)^2$		
	Cup	Hot Wire	LDV	Cup	Hot Wire	LDV
1	4.203	4.232	4.044	.612	.604	.689
2	4.486	4.488	4.253	.539	.524	.672
3	3.762	3.799	3.585	.340	.258	.348
4	4.245	4.270	4.247	.458	.355	.596
5	3.976	4.000	3.953	.444	.340	.526
6	3.823	3.847	3.693	.342	.342	.503
7	3.618	3.642	3.489	.623	.598	.573
8	4.212	4.235	4.073	.461	.361	.674
Averages	4.041	4.064	3.917	.472	.413	.567

The mean wind speeds detected by the LDV is in overall 3 percent agreement with the cup anemometer, and within 5 percent for any given 4.26-minute interval. The greater spread for smaller time intervals is to be expected because of the spatial spread of sampling points for the two instruments.

The variances for LDV are larger than those detected by either the hot-wire or the cup anemometer. It is surprising to note also that the variances for the hot wire are less than that for both the cup anemometer and LDV measurements. The greater variances for the LDV results are due in part to the fact that only mean horizontal angularity of the particle motion with respect to the laser axis is included in the correction. Thus there are greater variations of velocities from the mean. This is observed also in comparing the mean speeds for the three data sets. The mean is lower for the LDV as compared to cup speeds.

Probability distributions - The distributions of velocities about the means for the three instruments are shown in Figure 25. These data are in terms of standard deviations, and are not normalized so that straight lines are drawn from one data point to another. The distributions are skewed to the right. This skewness is governed by the nature of the turbulence in the atmosphere rather than by instrument response, as it can be seen that all three instruments respond similarly. The percentage of data near the mean is greater for the cup anemometer than for the other instruments, as was suggested in the preceding paragraph, the percent of low velocities appear to be greater for the LDV than for either cup or hot wire measurements.

Spectral densities - The spectral densities for measured turbulence in the atmosphere are shown in Figures 26, 27 and 28 for the cup, hot wire and LDV instruments, respectively, and a comparison of the three are shown on Figure 29.

There are apparent energy concentrations in the spectra for the cup anemometer and hot wires at 5 Hz which are also noted at 2.5 and

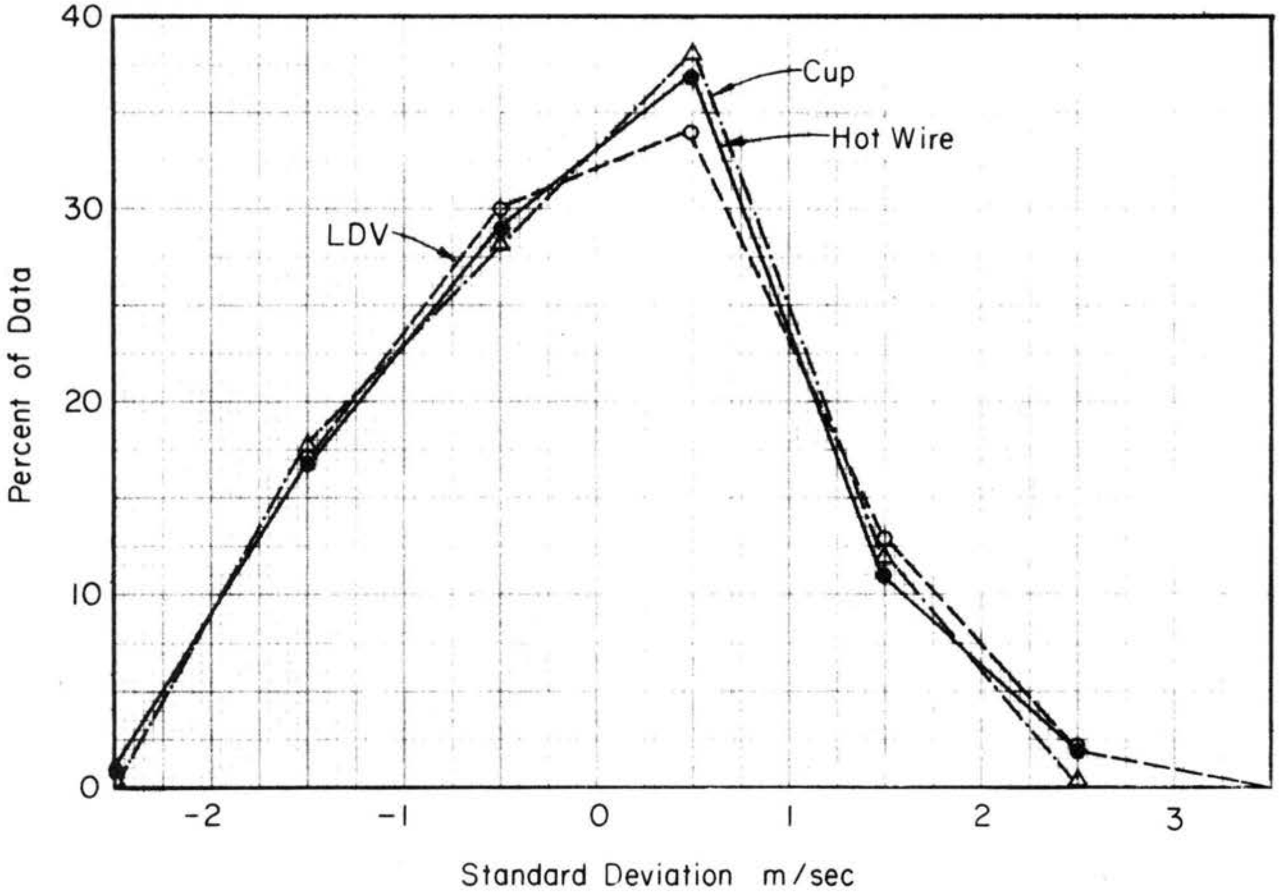


Figure 25. Distributions of velocities about the mean.
Test 50801

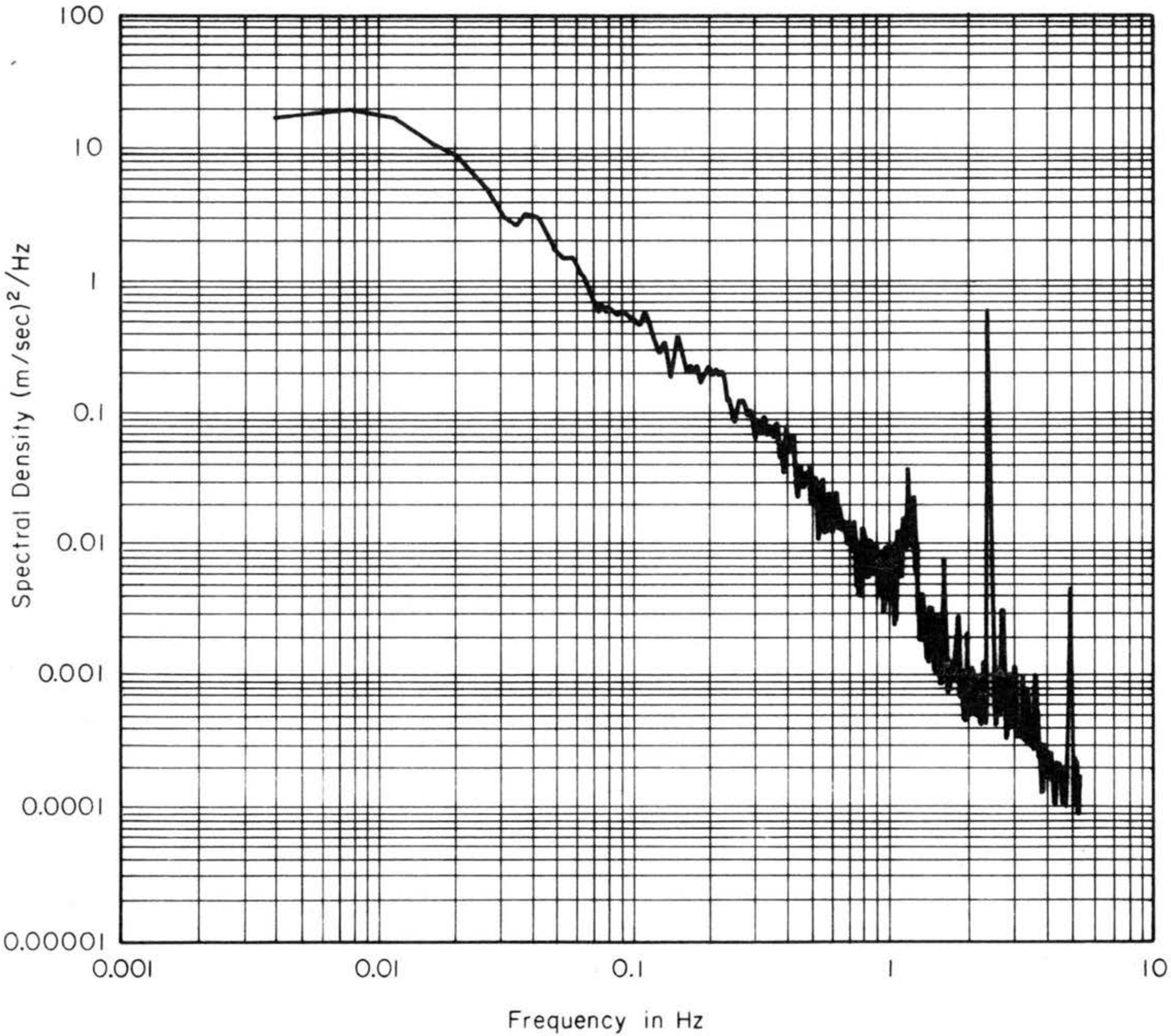


Figure 26. Spectral density distributions for cup anemometer.
Test 50801

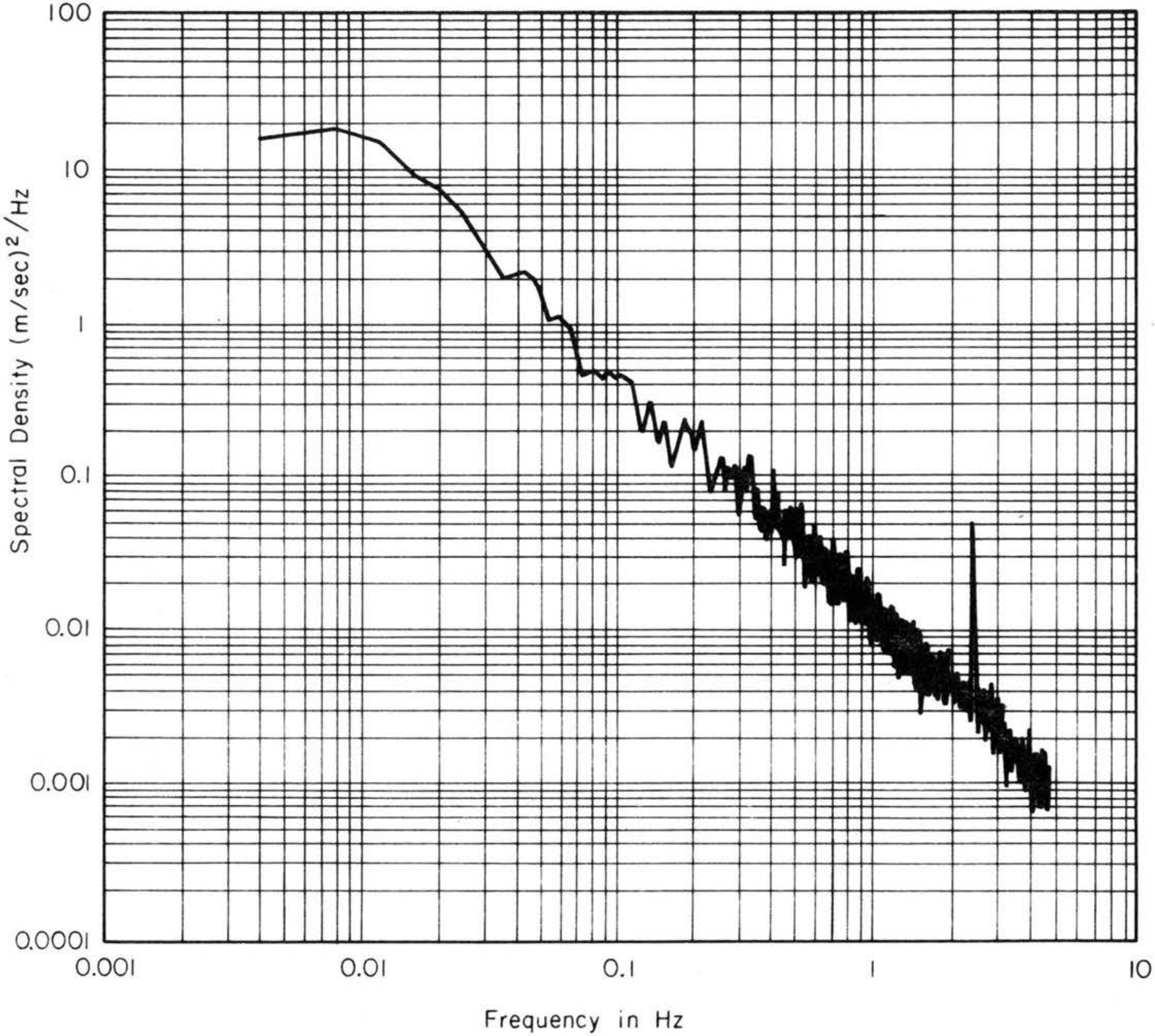


Figure 27. Spectral density distributions for hot-wire anemometer. Test 50801

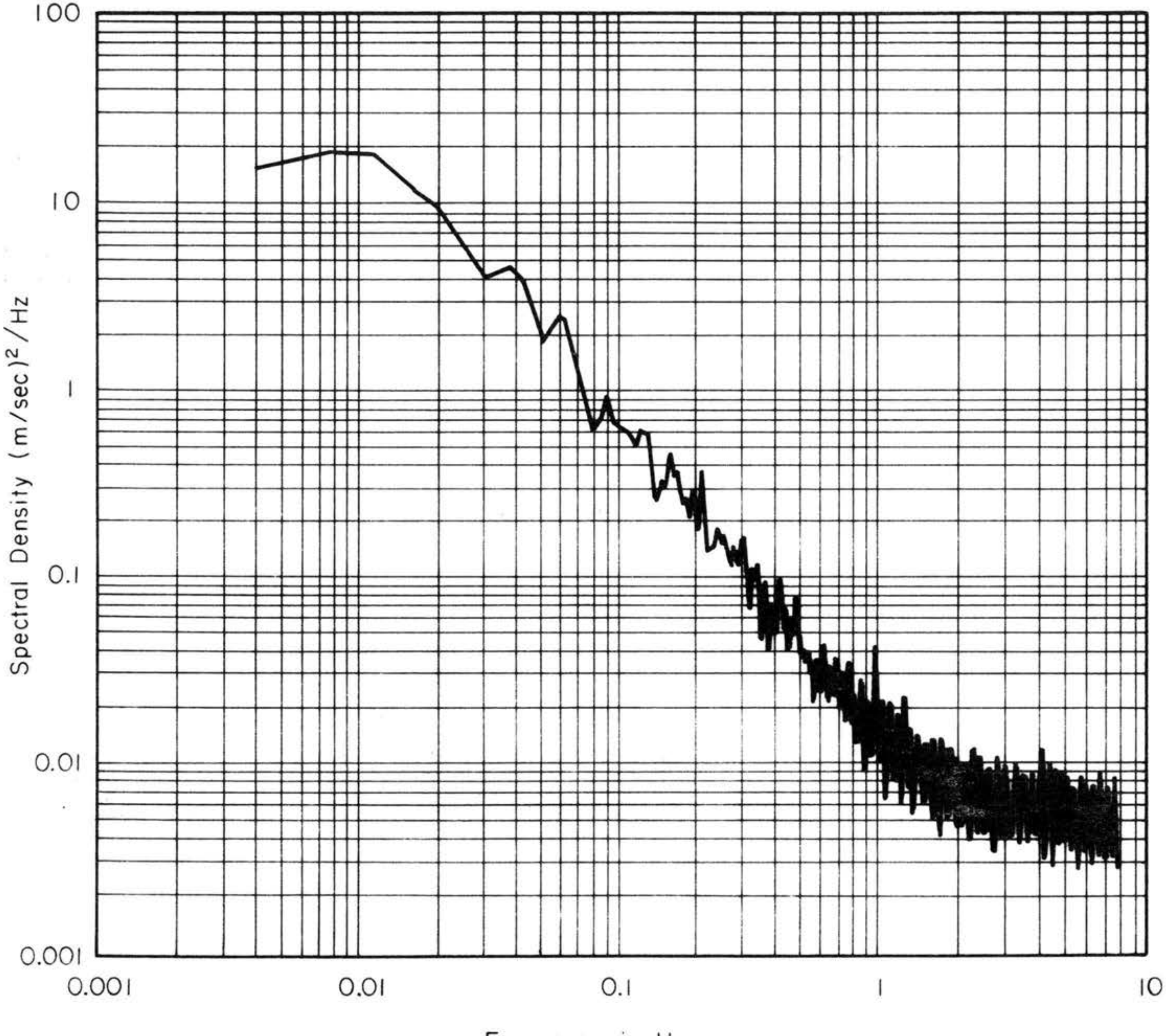


Figure 28. Spectral density distributions for LDV data.
Test 50801

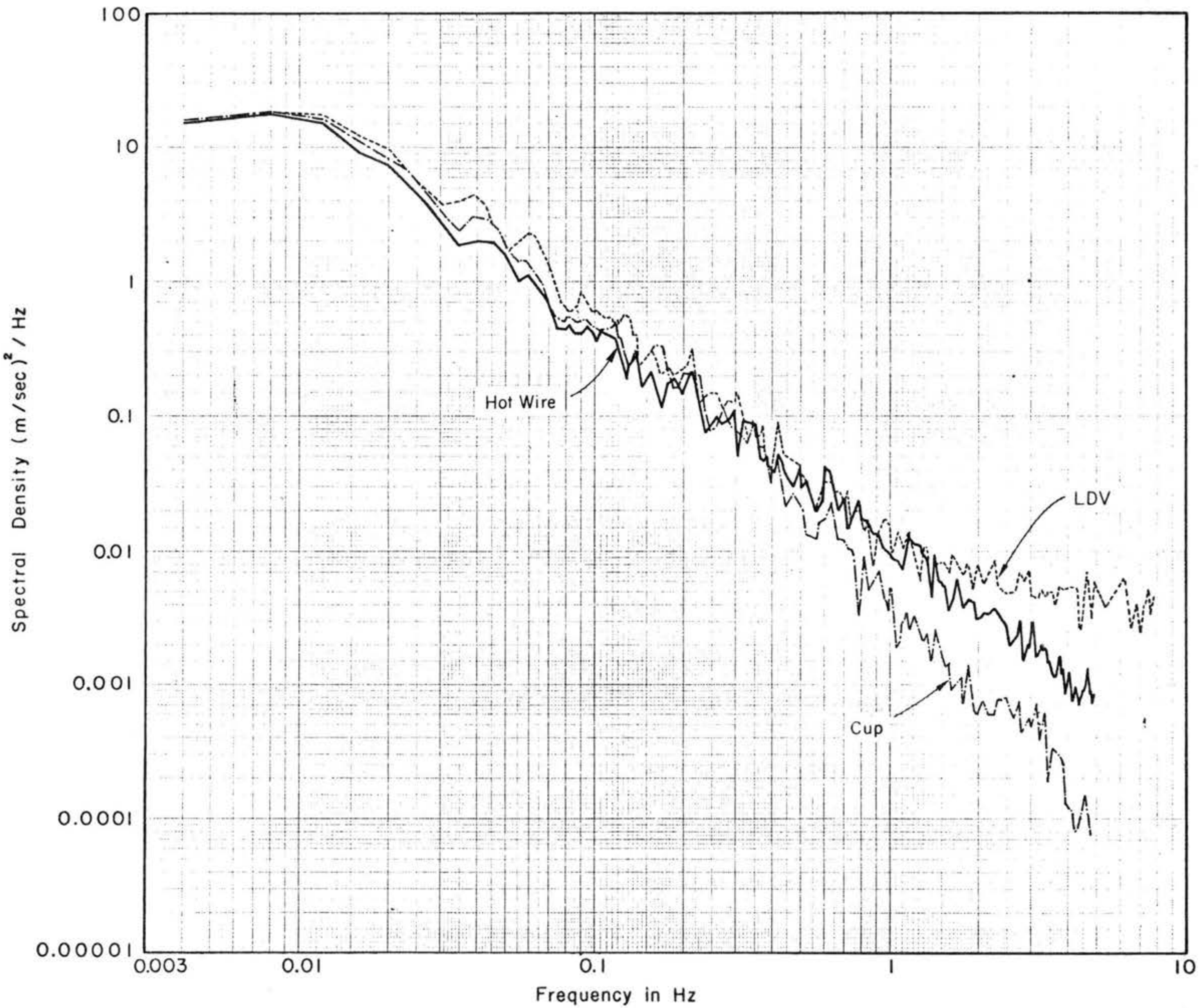


Figure 29. Comparison of spectral density distributions for Test 50801.

1.25 Hz. These must be due to mechanical aliased frequencies from the tape recorder, for they appear in the hot-wire and cup anemometer data but not in the LDV data. Mechanical aliasing does not appear in the LDV data because of the manner in which the velocity-time history is generated (see section on data reduction).

If the aliased spectral densities are ignored, it can be seen that the hot-wire and cup anemometer have identical spectra up to 0.4 Hz. Beyond that frequency, the spectrum decreases because of the limited frequency response of the cup anemometer. The cup anemometer data may in principle be corrected by a frequency response function (see Camp 1965), but in this study the correction was not made, as the comparison spectrum for higher frequency is given by the hot-wire anemometer data. The response of the constant temperature hot wire used here is up to at least 1 KHz and the data were filtered at 5 Hz before digitizing.

As it is seen on Figure 29, the spectral densities for the LDV-measured turbulence is slightly greater for frequencies less than 1 Hz, but essentially parallel to the hot-wire data. For higher frequencies, there appears to be more energy contained in the LDV-measured turbulence. This must be aliased information because the hot-wire data do not show this trend.

The aliasing must arise from the technique used in data reduction. While the spectrum analyzer is being swept (sampled) at a rate of 16 Hz, thereby effectively establishing the Nyquist frequency, the velocity time data cannot be filtered at 8 Hz before the sampling is done. That is, turbulence of higher frequency transporting aerosol and solid particles in the atmosphere are sensed in the resolution volume of the LDV. Thus in calculating the velocity from the sampled spectrum, the aliasing from higher frequency cannot be avoided. What is surprising, however, is to note the magnitude of the aliased spectrum in the LDV-measured turbulence indicated by the deviation beyond 1 Hz.

<u>Measurements of Run 32701</u> (March 27, 1971)

The data for this test were taken from 3:30 pm to 4:18 pm, a period of 48 minutes. The wind was essentially steady from the north-east (60 degrees east from north) at around 12 m/sec (27 mph). Particle counts in the atmosphere were not available for this test. There was an arctic front moving in from the north and the air was "clean." Visibility was virtually unlimited. The laser beam axis was directed downwind in this test because the direction of the wind was such that the laser beam axis would have been close to a vertical leg of the tower.

Velocity profiles - The velocity profiles for successive 10-minute intervals are shown on Figure 30. The profiles are logarithmic and the mean velocities increased in the first 20 minutes of the 50-minute period and decreased thereafter. The spread of mean velocities for the total period varied from about 10.7 to 13 m/sec at the level of the focal region of the laser beam.

Spectrum analyzer settings - The settings of the spectrum analyzer were as follows:

Sweep rate:	5 ms/cm
Sample rate:	16 sweeps/sec
Frequency dispersion:	0.5 MHz/cm
Filter Bandwidth:	off
Bandwidth:	30 KHz

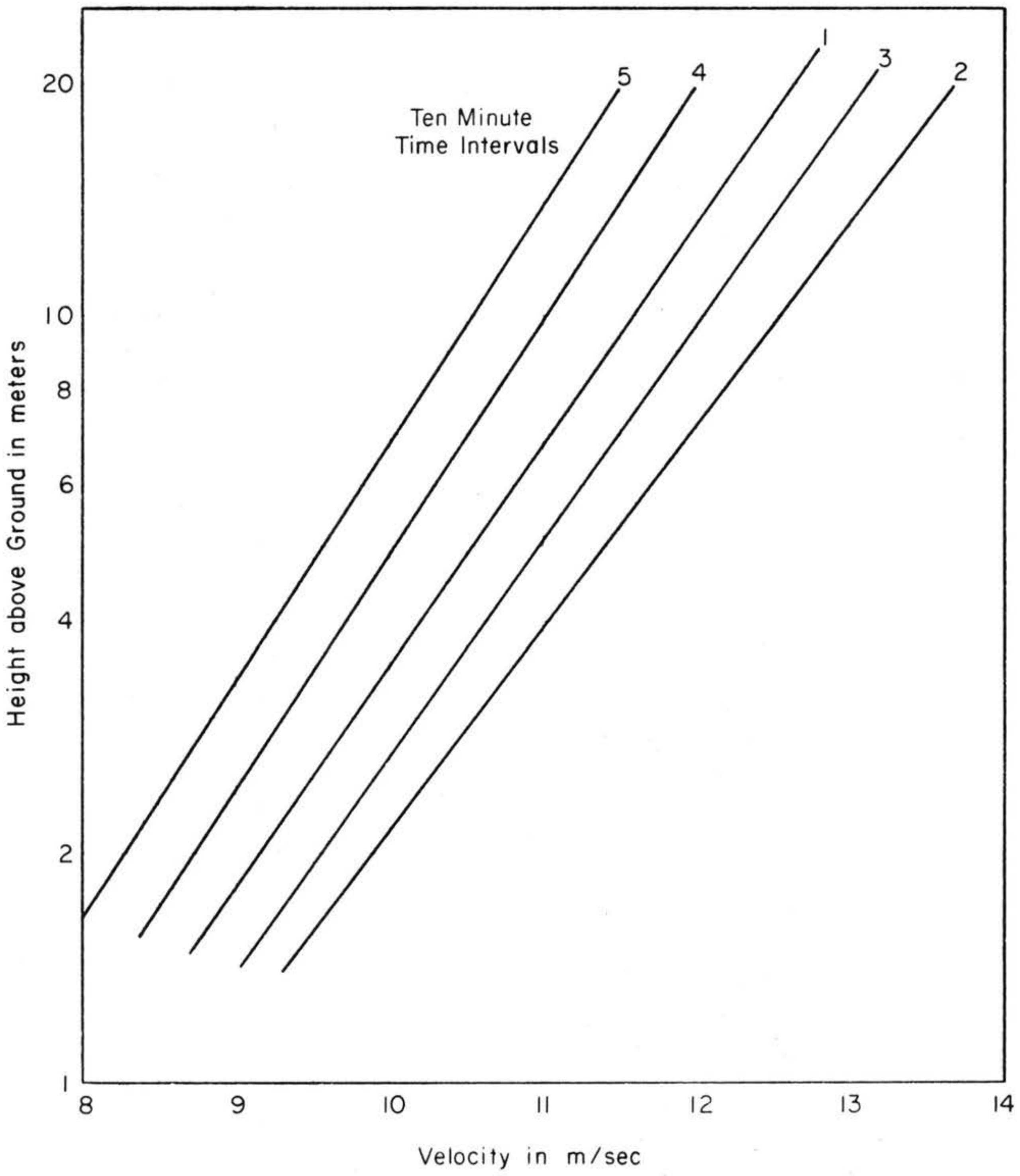


Figure 30. Velocity profiles. Test 32701

The calibration frequency was 4.009 MHz (21.2 m/sec) which is shown in Figure 31. The noise level from the detector is shown in Figure 32. The vertical scale in the oscilloscope trace is 100 mv/cm.

Typical Doppler signals are shown in Figures 33 and 34. As noted, the S/N ratio is small and the spectral dispersion is also small. There were larger periods of signal dropout, that is sweeps when there were no detectable signals. In these instances the analysis was made assuming that the velocity indicated in the current sweep was equal to that of the previously detected velocity.

Velocity time traces - Time traces of velocity from the three instruments are shown in Figures 35 and 36 for two representative 4-minute time intervals.

There is reasonable agreement between the cup anemometer and hot-wire traces in general trend of mean velocities. However, the turbulent fluctuations in the hot-wire signals are greater than that indicated by the cup anemometer traces. The LDV signals have several peculiarities. The fluctuations are clipped at both the upper and lower limits. These clipped signals are results of the low S/N ratio and the computer program. As indicated previously, the low particle concentration in the atmosphere often caused no detectable signal in a given sweep of the spectrum analyzer. In such instances the velocity was set equal to the immediately-previous calculated velocity. At the lower end, the signal was lost in the noise (see the noise calibration trace of the oscilloscope) and a previously higher value was then identified as the velocity for that sweep. There are noticeable high peaks in the LDV trace. It is believed that these signals are spurious, resulting from identification of high noise peaks as Doppler signals. The trend of mean

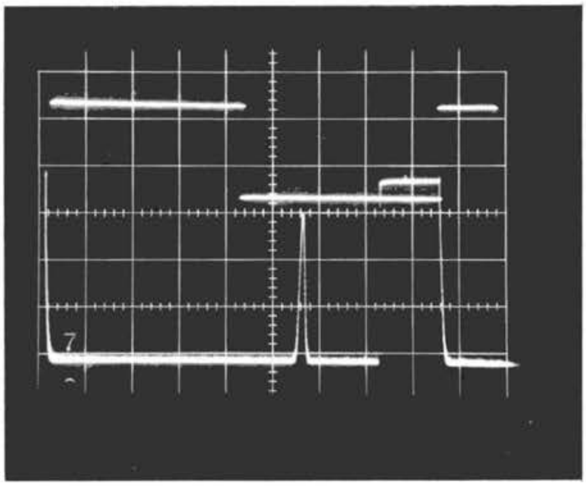


Figure 31. Calibration frequency 4.009 MHz.
Test 32701

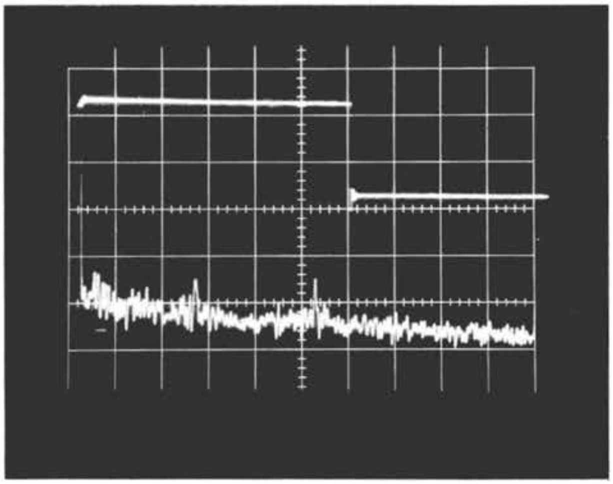


Figure 32. Noise calibration.
Vertical scale is 100 mv/cm
Test 32701

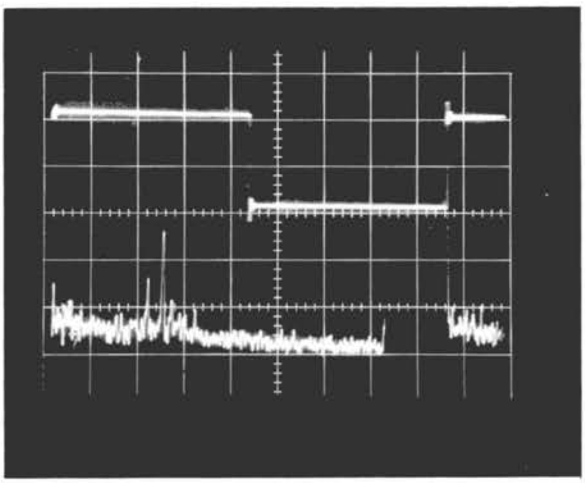


Figure 33. Typical Doppler signal.
Test 32701

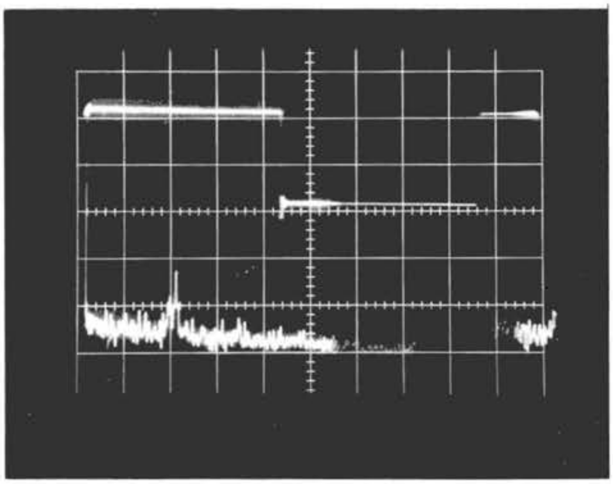


Figure 34. Typical Doppler signal.
Test 32701

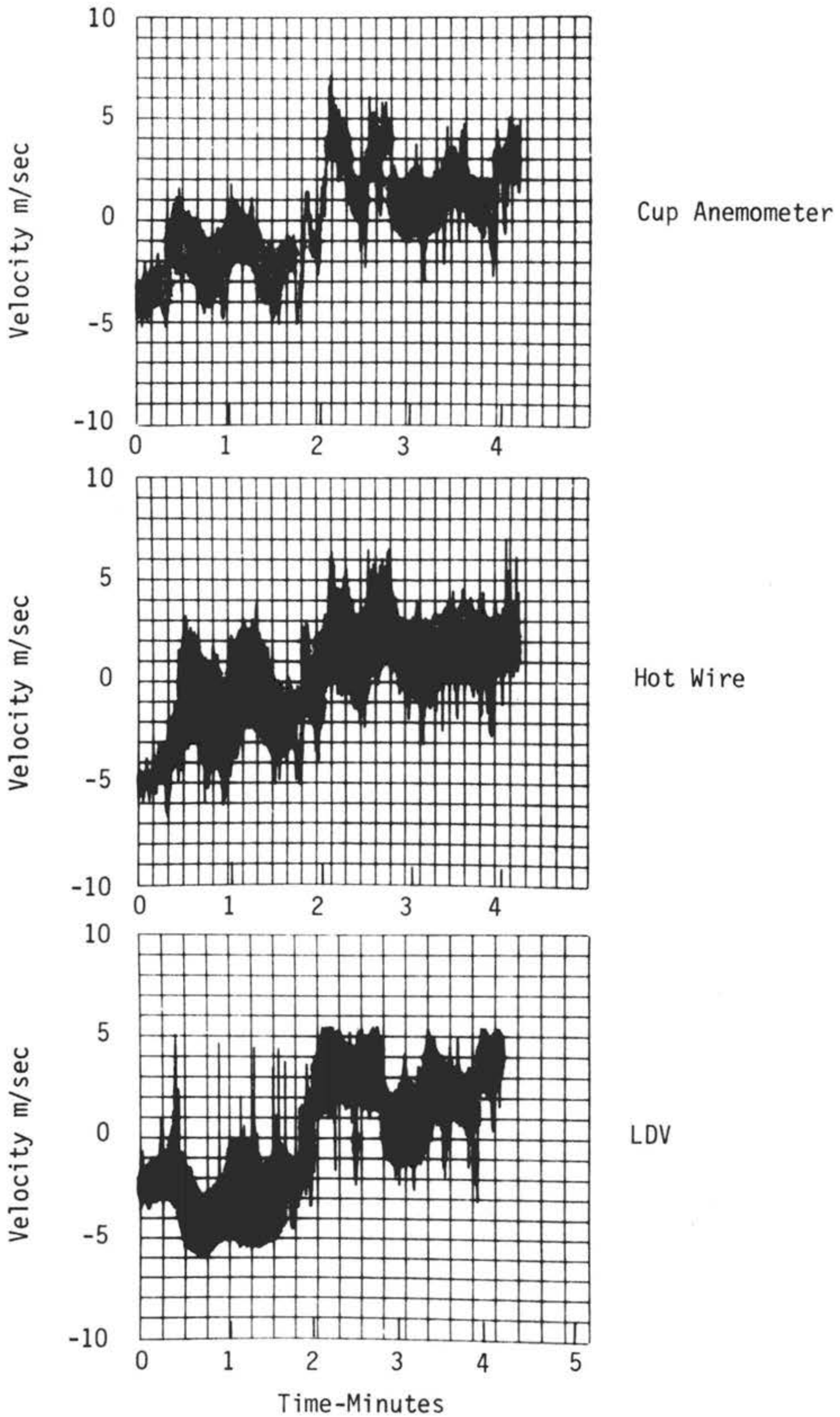


Figure 35. Time traces of wind velocity
Test 32701, Interval 3
(For means and variances see Table 4)

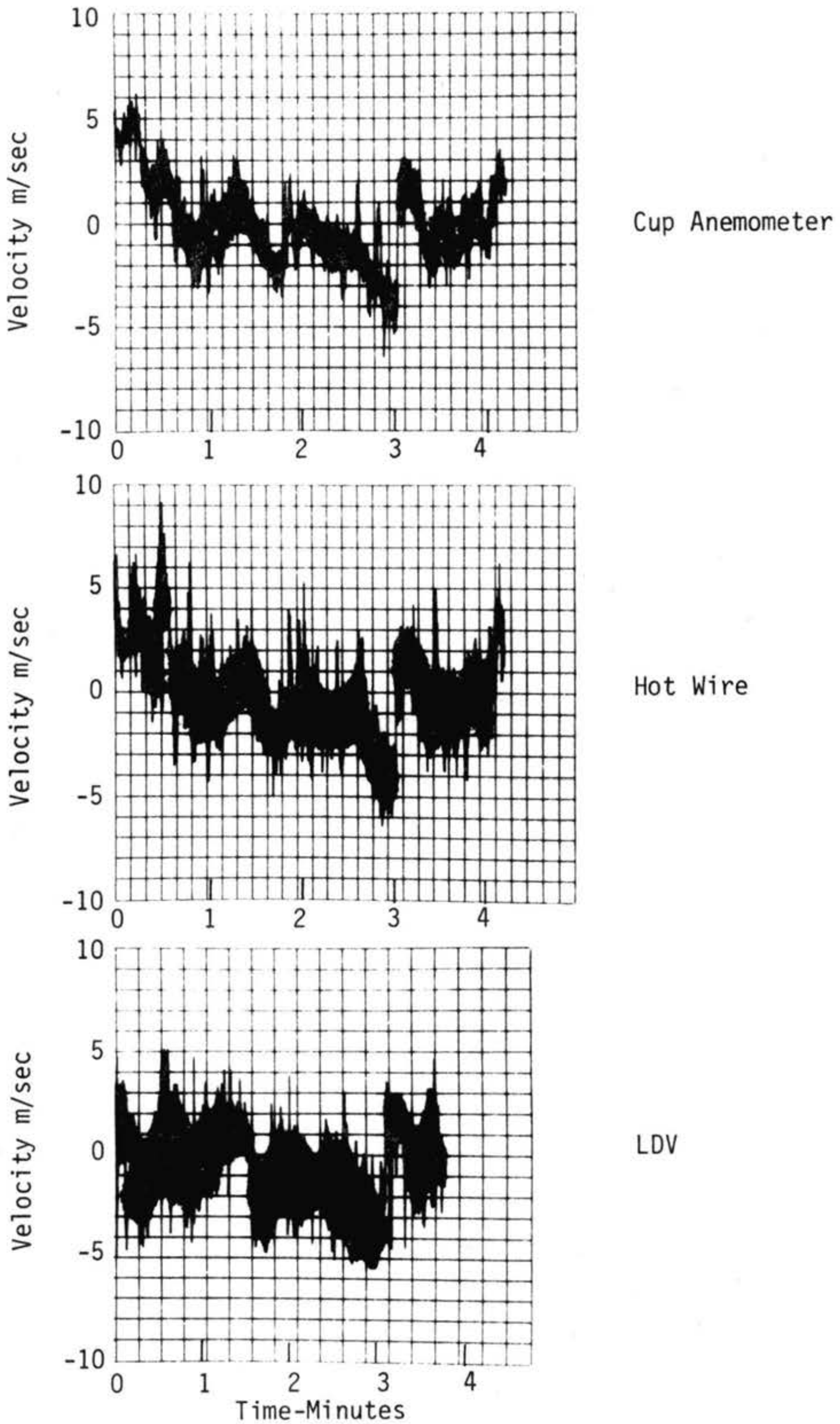


Figure 36. Time traces of wind velocity
Test 32701, Interval 5
(For means and variances see Table 4)

velocities is generally identifiable, but the comparison is not as favorable as for test 50801.

Means and variances - The means and variances from a 34-minute interval of the total record are given in Table 4.

TABLE 4. MEANS AND VARIANCES FOR TEST 32701

4.26-Minute Intervals	Mean Velocities m/sec			Variances $(m/sec)^2$		
	Cup	Hot Wire	LDV	Cup	Hot Wire	LDV
1	12.041	12.152	11.697	2.686	5.067	4.326
2	13.659	13.835	13.460	4.951	4.281	6.111
3	13.164	13.203	13.990	6.497	7.258	9.897
4	13.973	14.094	14.226	4.117	5.382	5.415
5	13.486	13.575	14.557	5.167	5.429	6.833
6	12.658	12.697	12.441	2.812	3.349	6.620
7	12.417	12.578	11.570	4.000	6.290	3.193
8	11.453	11.551	10.071	2.934	3.826	2.802
Averages	12.856	12.961	12.751	4.093	5.040	5.448

The average wind speed detected by the LDV in the 34-minute period is within 1 percent of the cup and hot wire averages. There are larger variations however for the shorter 4.26-minute intervals, and as the time traces would suggest, variations become greater for even shorter periods. As noted in the preceding section, these are undoubtedly caused by the spurious signals in the velocity calculations. The mean velocities measured by the hot wire were generally larger than the cup anemometer, and the variances as expected are definitely greater because the frequency response of the cup anemometer is limited.

Over a 34-minute period, the fluctuations (variances) detected by the LDV are larger than those of the hot wire. This was also true for Test 50801 which had considerably lower mean wind speeds. Again, the spurious signals in the LDV velocities contribute significantly to variances.

Probability distributions - The distributions of velocities about the means for the three instruments are shown in Figure 37. Turbulence velocities are skewed to the left for all three instruments. The LDV data indicated difficulty in tracing the larger velocities. As explained previously, this could be due in part to the three dimensional nature of turbulence and only the horizontal angularity was corrected (in the mean) in these measurements. This feature of the LDV traces was noted also for test 50801.

Spectral densities - The spectra for the cup anemometer, hot wire and LDV data are shown in Figures 38, 39 and 40, respectively. For comparison, the three are replotted in Figure 41. Spikes of high frequency are again noted at 2.5 and 5 Hz in the cup anemometer spectra. It was noted that the time traces of the LDV data included spurious spikes of high velocity. These spikes are transformed into the spectra and are noted particularly as spikes of power near 1 and 3 Hz. These spikes in the spectra were ignored in replotting on Figure 41.

The spectra of turbulence measured by the LDV and hot wire compare favorably. This is also indicated by the comparison of variances in Table 4. The cup spectra however drops off at around 0.2 Hz because of the limited frequency response. Response corrections for the cup anemometer were not made.

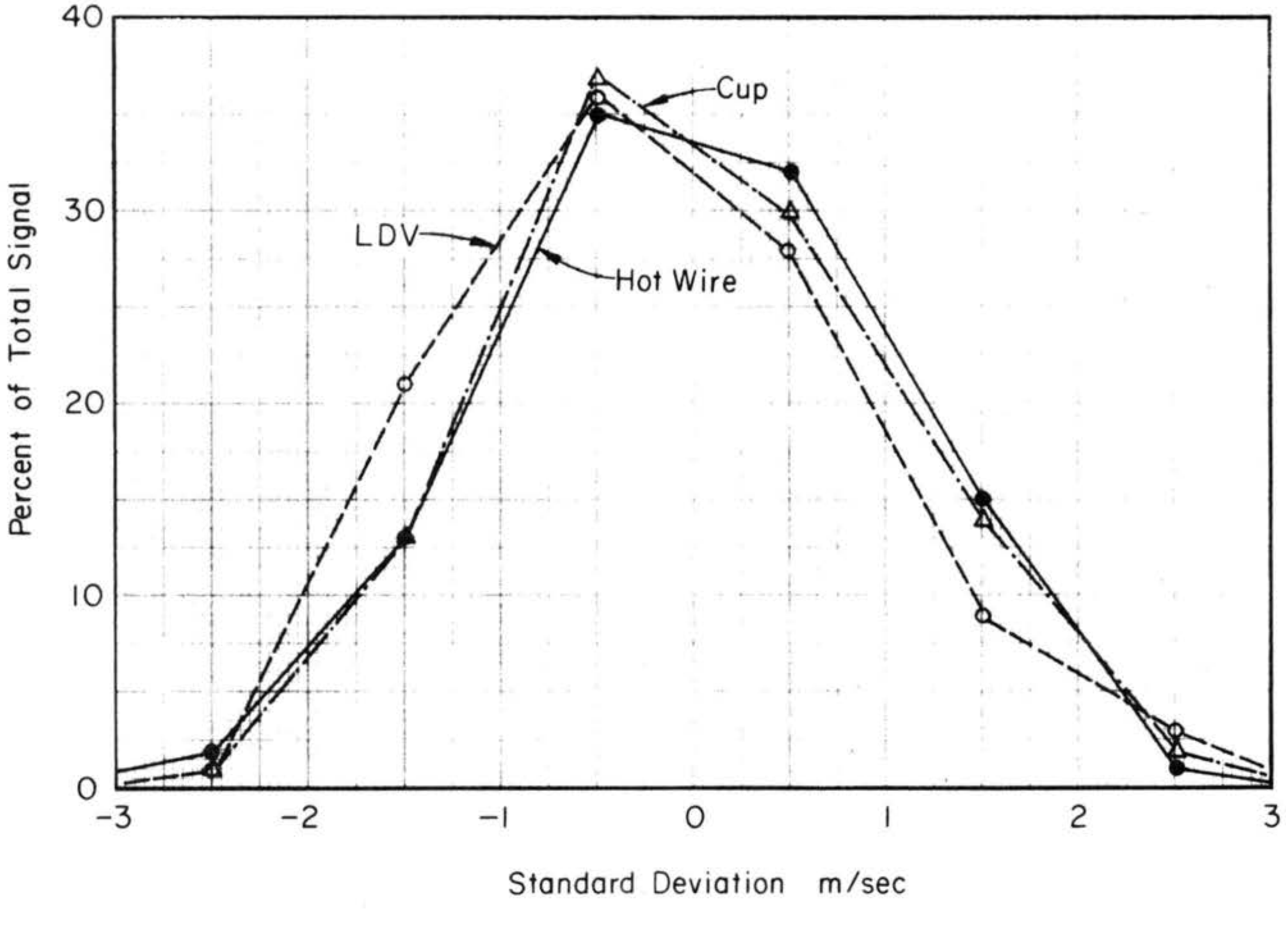


Figure 37. Distributions of velocities about the mean.
Test 32701

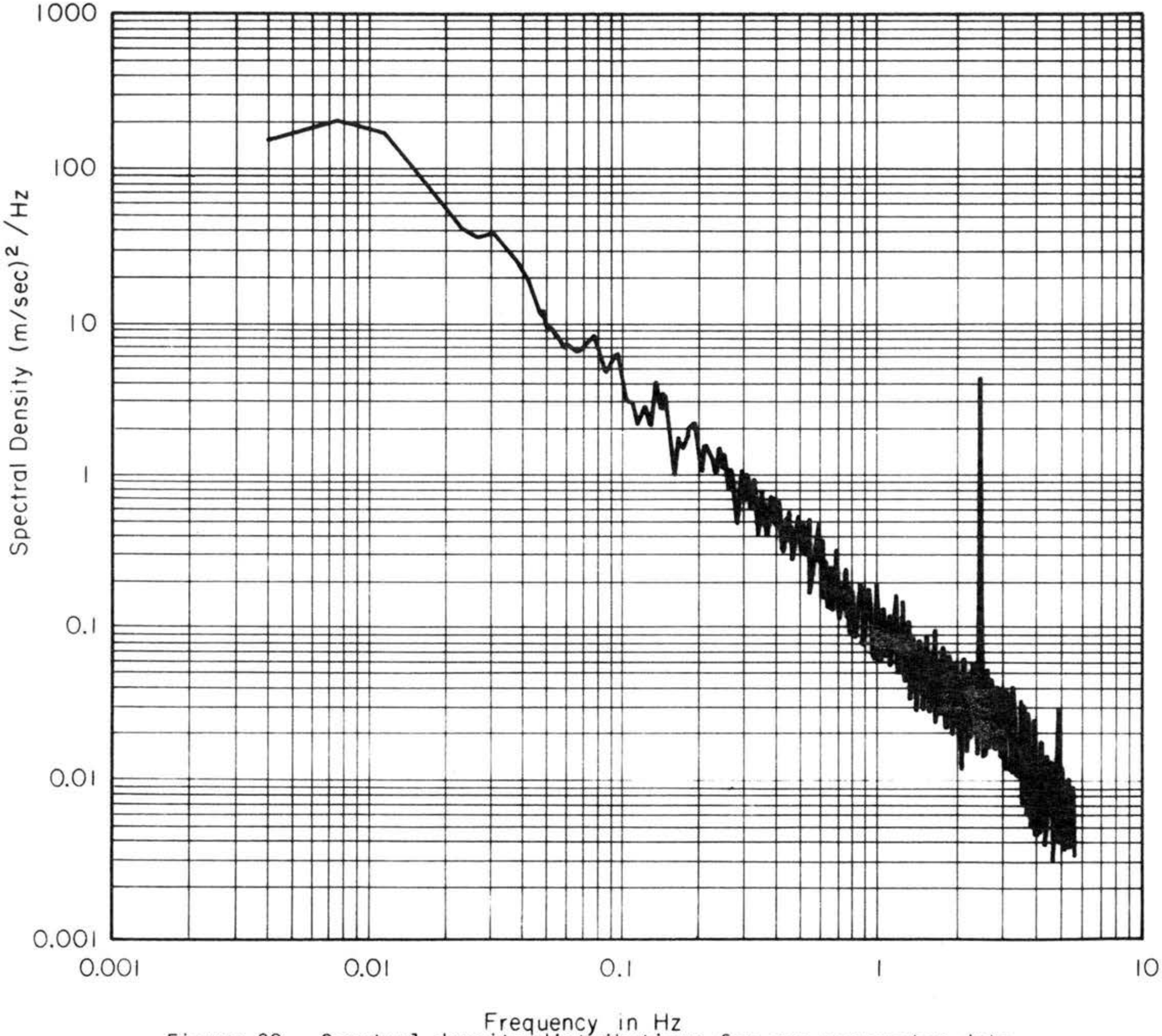


Figure 38. Spectral density distributions for cup anemometer data.
Test 32701

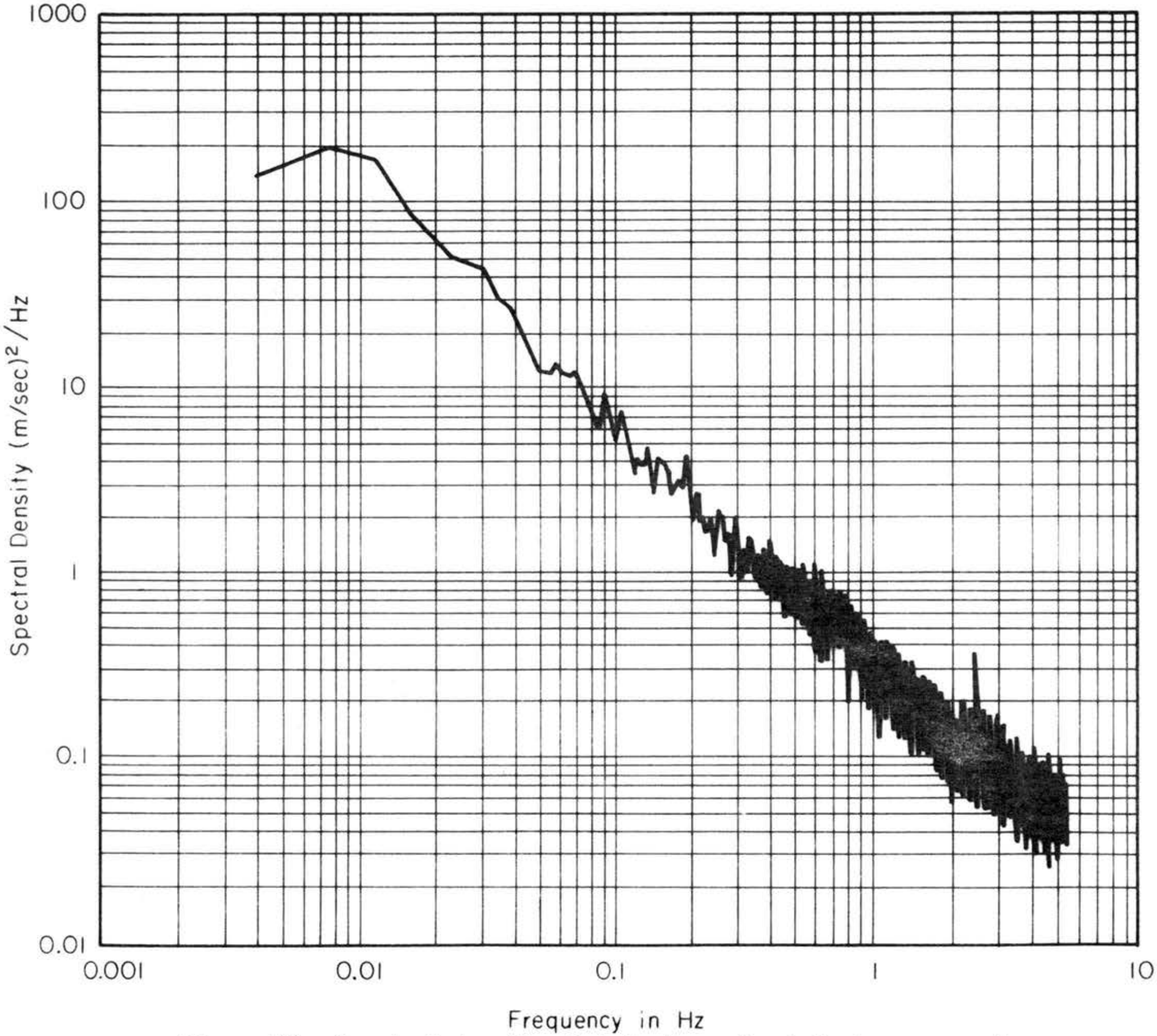
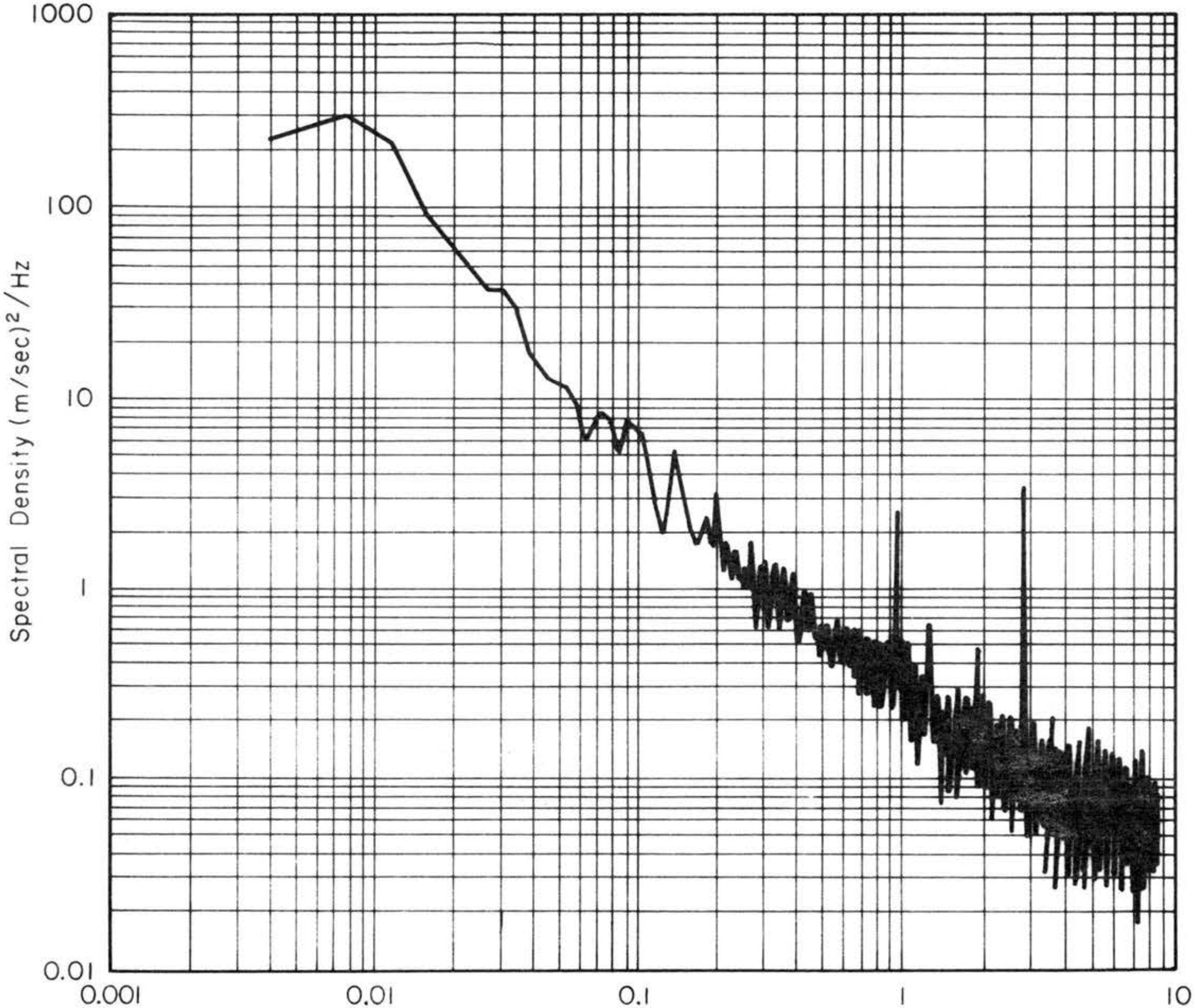


Figure 39. Spectral density distributions for hot-wire anemometer. Test 32701



Frequency in Hz

Figure 40. Spectral density distributions for LDV data.
Test 32701

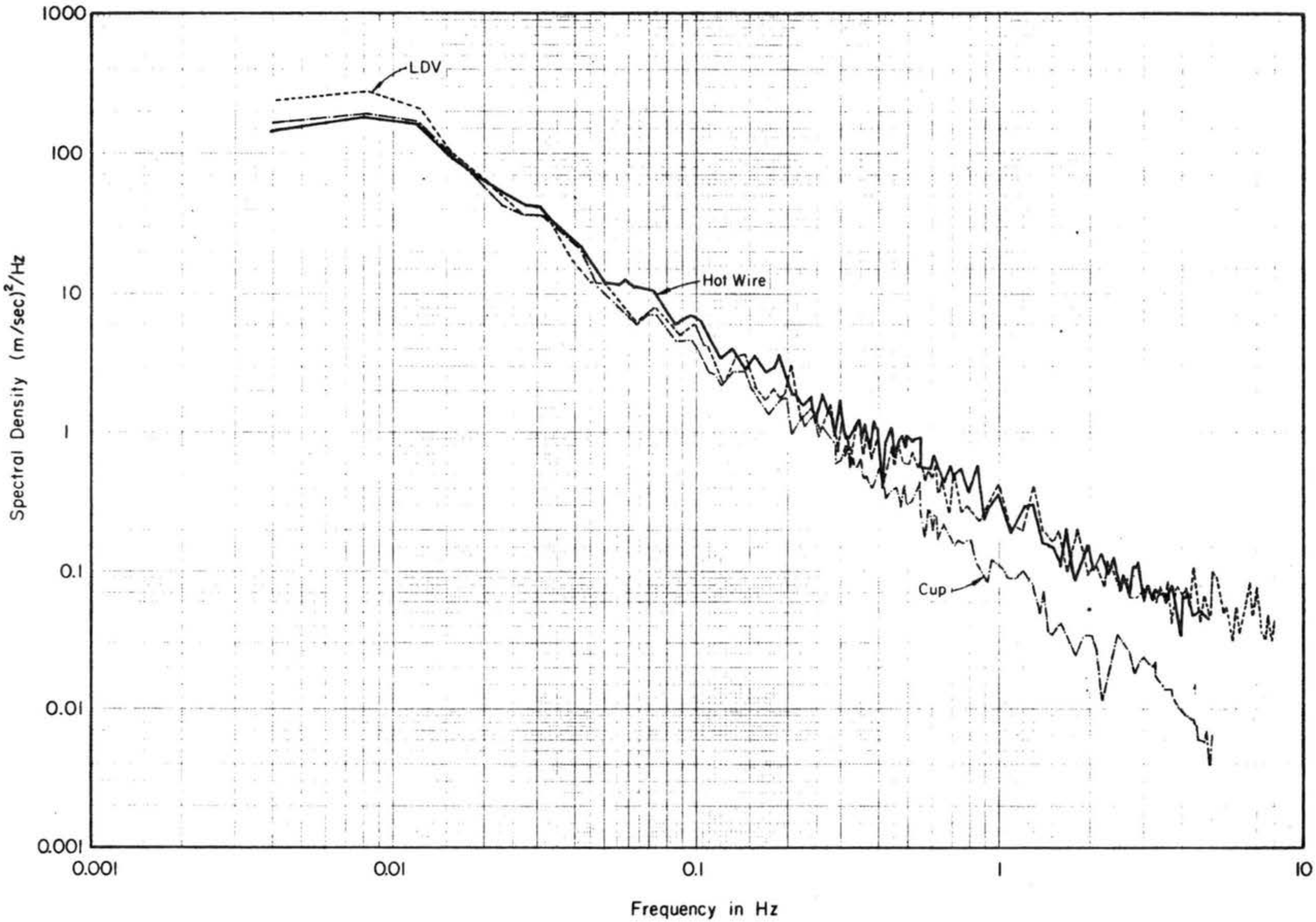


Figure 41. Comparison of spectral density distributions for Test 32701.

Measurements of Run 101401 (October 14, 1971)

The data for this test were taken from 9:16 pm to 9:55 pm, a period of 39 minutes. The wind was from the north-northwest across the clear grassland. The mean wind speed varied from about 4 m/sec at the start of the test to about 5.7 m/sec at the end. The wind direction remained constant. With a northern weather front moving in, the air was clear, (little pollution), and visibility was good.

Spectrum analyzer settings - The settings of the spectrum analyzer were as follows:

Sweep rate:	5 ms/cm
Sample rate:	16 sweeps/sec
Frequency Dispersion:	0.2 MHz/cm
Filter Bandwidth:	10 KHz
Bandwidth:	30 KHz

The calibration frequency was 1.691 MHz, which is shown in Figure 42. The noise level is shown in Figure 43. It will be noted that reference zero frequency is shifted slightly from the pulse rise of the square wave, resulting from a horizontal axis shift of the spectrum analyzer. An accounting of this shift was made in data analysis.

A sample trace of one sweep of the spectrum analyzer is depicted in Figure 44. The S/N of the Doppler trace is small but was sufficient to discriminate from noise. There were drop outs in Doppler signature as indicated by the time traces of wind speeds.

Velocity time traces - Time traces of wind speeds from the cup and hot wire anemometers and the LDV are shown for representative 4-minute intervals in Figures 45 and 46. As with the two previous tests, the mean trends correspond with apparent differences in turbulence

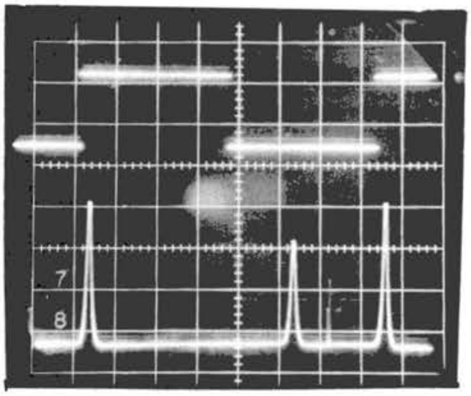


Figure 42. Calibration frequency 1.691 MHz.
Test 101401

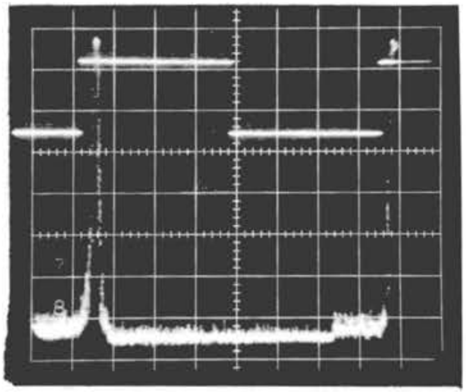


Figure 43. Noise Calibration.
Test 101401

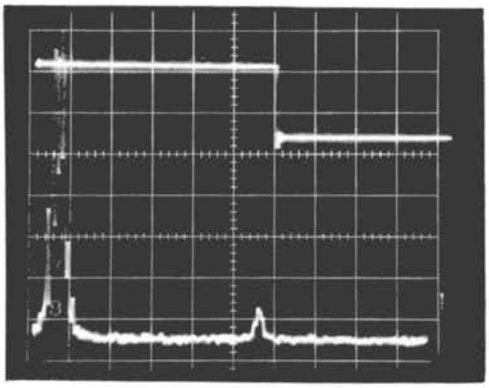


Figure 44. Sample Doppler signal.
Test 101401

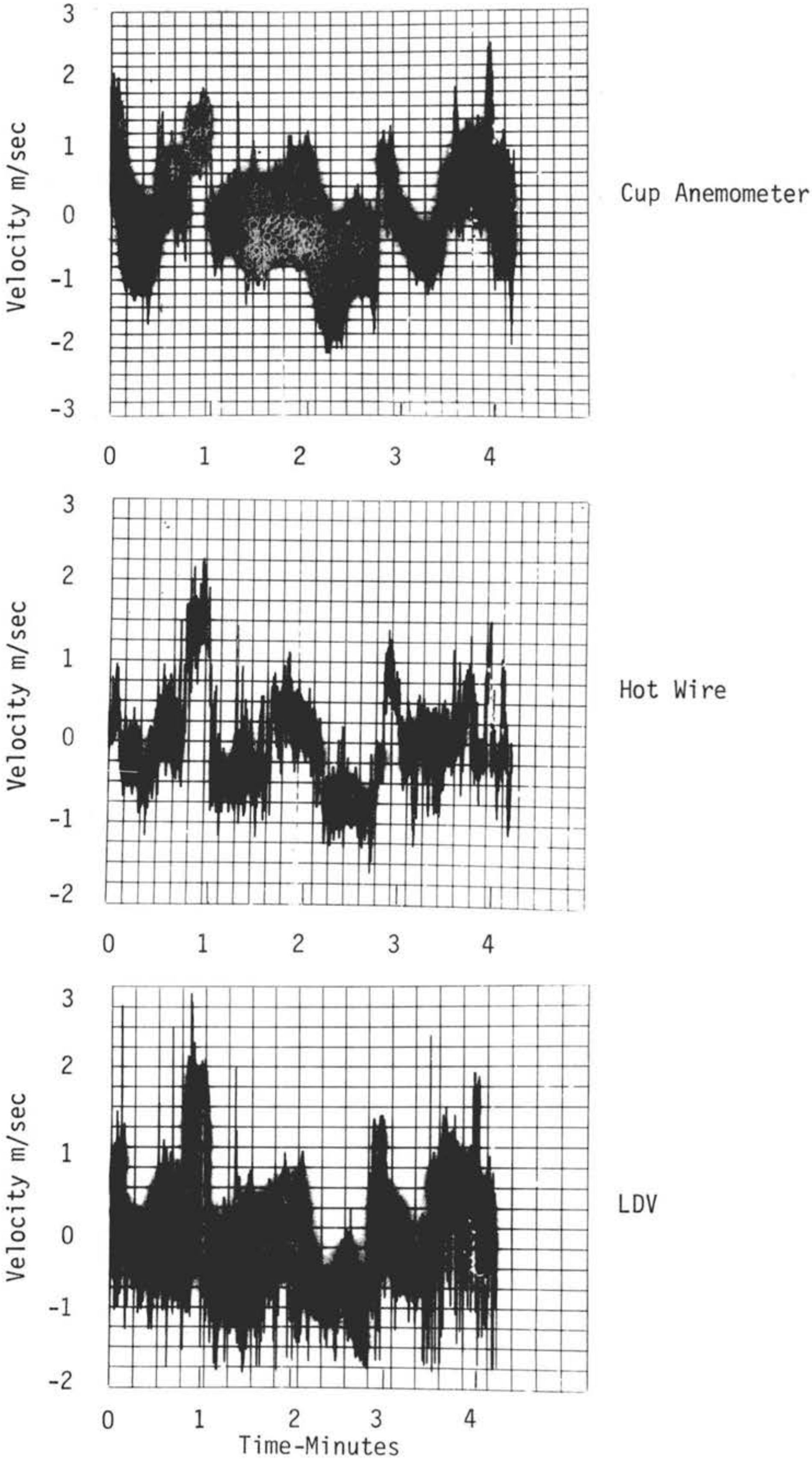


Figure 45. Time traces of wind velocity.
Test 101401, Interval 1
(For means and variances see Table 5)

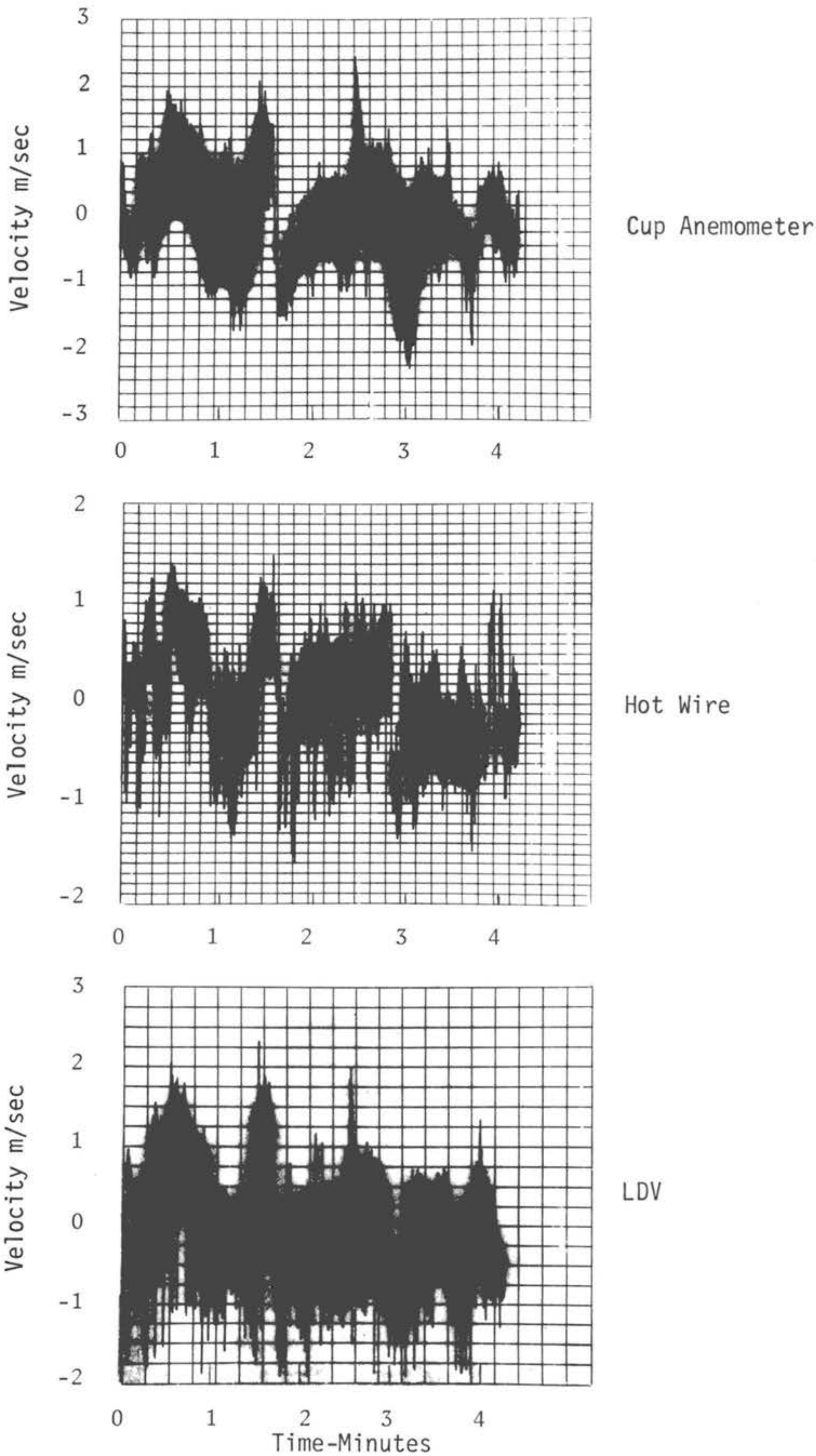


Figure 46. Time traces of wind velocity.
Test 101401, Interval 5
(For means and variances see Table 5)

fluctuations. The large number of low points in the LDV signature resulted from the low S/N ratio; particularly by having to set a low level trigger in the computer program. The spurrious high peaks are believed to be caused by extraneous signal in the Doppler sweep. There are not enough of these to cause difficulty with the statistical anslysis.

Means and variances - Means and variances for the entire 34-minute test period are given in Table 5 for each 4.26-minute segment.

TABLE 5. MEANS AND VARIANCES FOR TEST 101401

4.26-Minute	Mean Velocities m/sec			Variances $(m/sec)^2$		
Intervals	Cup	Hot Wire	LDV	Cup	Hot Wire	LDV
1	5.150	5.154	5.451	.760	.654	.770
2	5.535	5.543	5.677	.736	.600	.847
3	5.425	5.479	5.722	.940	.760	.977
4	6.052	6.092	6.463	.813	.744	.856
5	5.381	5.406	5.742	.714	.586	.692
6	6.417	6.426	6.698	.822	.809	.879
7	6.417	6.457	6.821	.702	.659	.707
8	5.958	5.996	6.218	.745	.614	.675
Averages	5.792	5.819	6.099	.799	.678	.800

The average wind speed indicated by the LDV measurements is about 5 percent greater than that indicated by the cup anemometer. This is comparably about the same as for Test 50801. The variance for the LDV is greater than for the anemometers. Also, the variance for the hot wire is less than that for the cup anemometer as was the case also for Test 50801.

Probability distributions - The distributions of velocities about the means for the three instruments are shown in Figure 47. The turbulent fluctuations are more normally distributed about the mean than was the case for the previous two tests. As before, the probability distributions compare favorably one instrument to another.

Spectral densities - A comparison of the spectral density distributions with frequency for the three instruments is shown in Figure 48. The spectral distribution for the cup anemometer drops off slightly at about 0.5 Hz, the hot wire spectrum decreases on a constant slope and the LDV spectrum tends to level off for higher frequencies. The 2.5 and 5 hertz spikes were not included in drawing these spectra. The comparisons are reasonable to about 1 Hz frequency.

Frequency tracker - Considerable difficulty was experienced in tracking the LDV output with the frequency tracker. The tracker required frequent adjustments during the test, and tracking was often lost. Consequently the tape recorded output was too intermittent and analysis was difficult.

From observations during the test, it was noted that when tracking was achieved, the D.C. output (although slightly nonlinear) corresponded with the mean Doppler frequency, hence with the indicated wind speed. The A.C. output however did not correspond very well with the turbulent fluctuations. For example, in Figure 49, is shown a simultaneous trace of the hot wire and the A.C. output from the tracker for Test 101401. The hot wire leads the laser focal volume by about 3 meters and the average wind speed was about 6 meters per second. The horizontal sweep on the oscilloscope was 0.2 sec/cm.

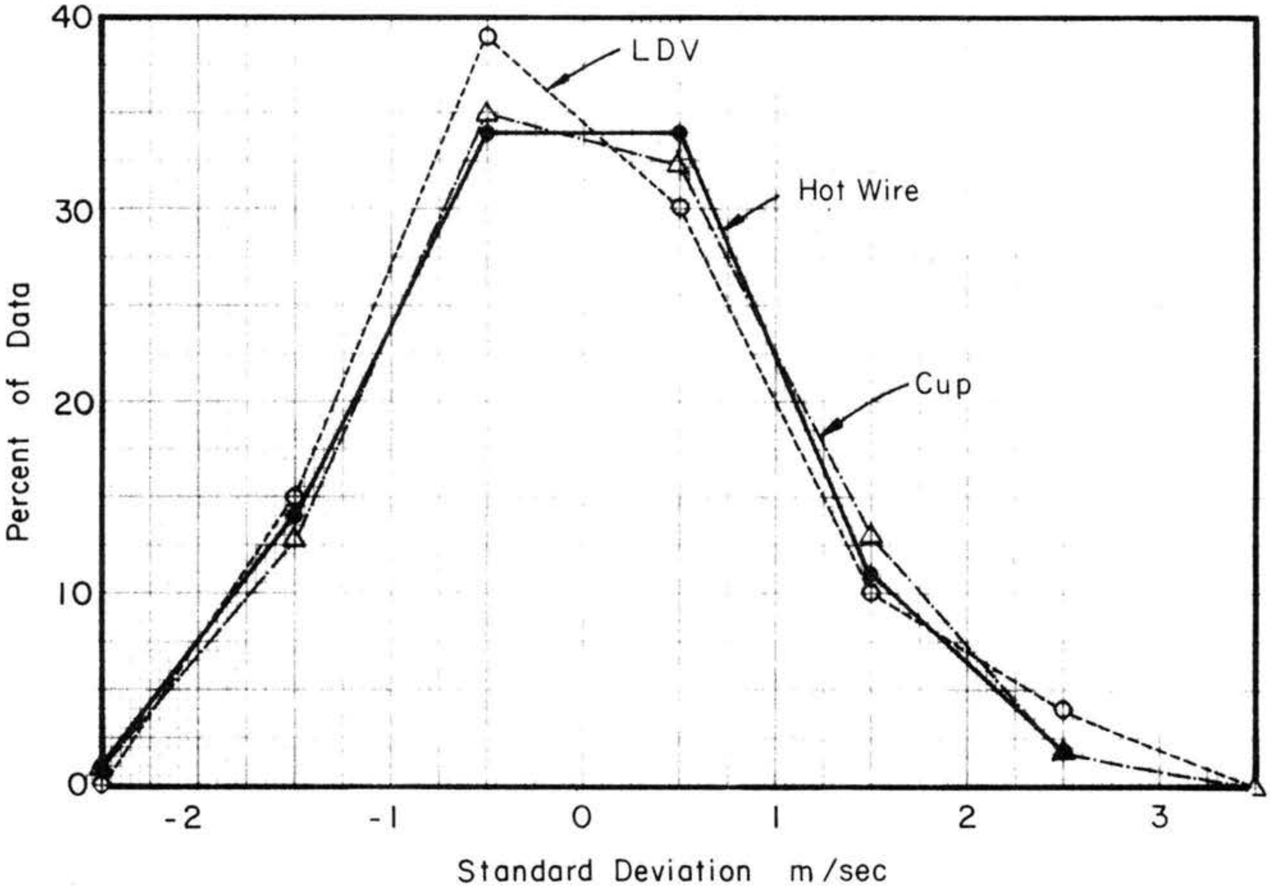


Figure 47. Distribution of velocities about the mean.
Test 101401

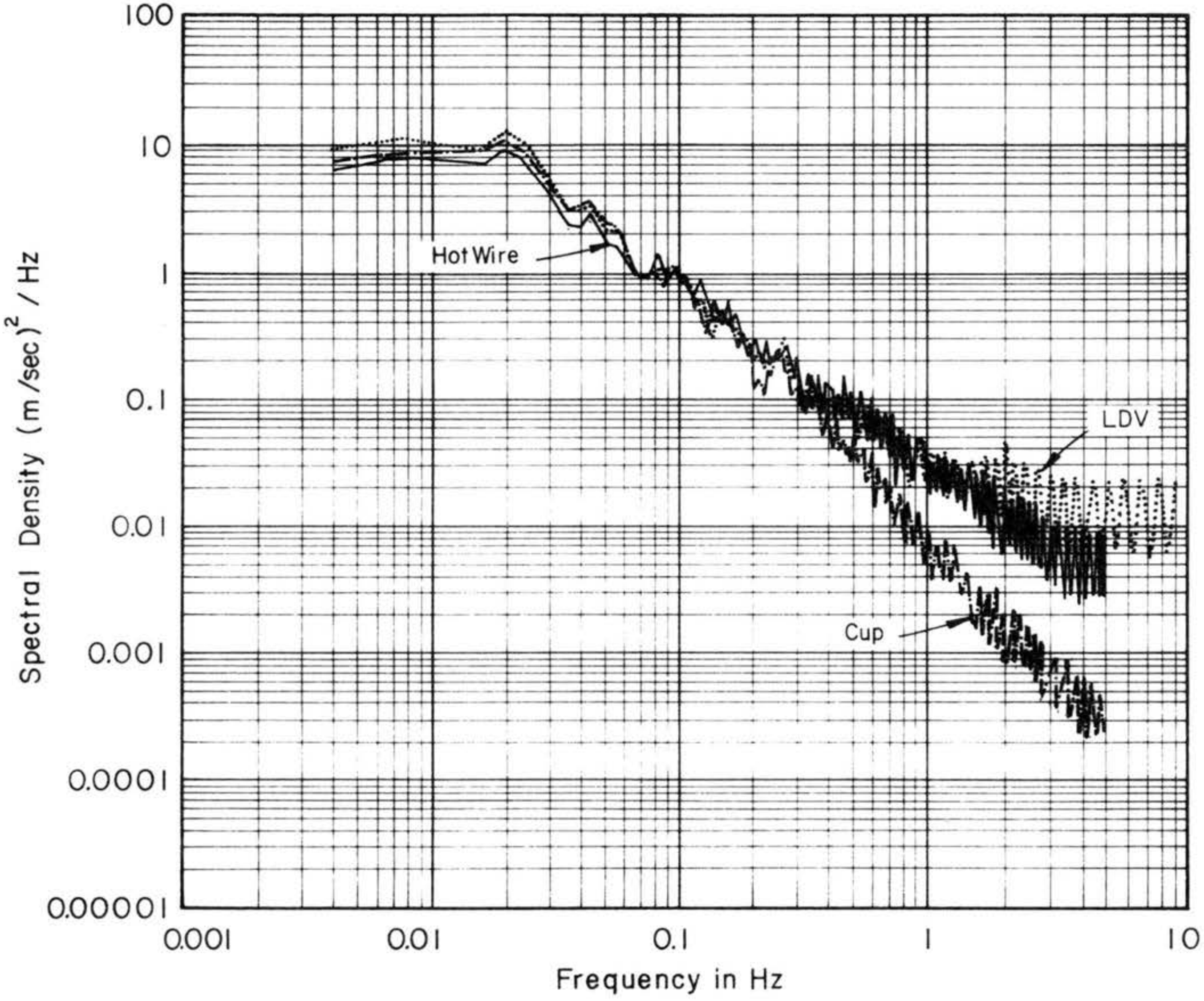


Figure 48. Comparison of spectral density distributions.
Test 101401

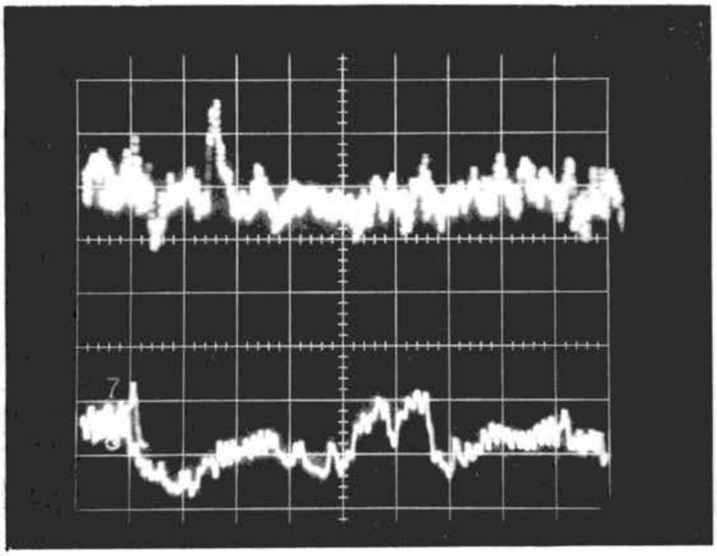


Figure 49. A.C. Tracker and Hot Wire Traces.
Test 101401

The A.C. output (top trace) resembles noise rather than turbulence, while the hot wire output is clearly that which traces the turbulence. The intermittency of the tracker signal created considerable difficulty with digital data analysis. After considerable effort, this part of the data analysis was abandoned. The particular frequency tracker used in these tests (1971) should be modified to provide long-term uninterrupted velocity-time histories. This of course is related to Doppler S/N ratio and to the concentration of aerosols which provide the Doppler shifted signals. With no Doppler signature (signal drop out) there can be no tracking regardless of the quality and design of the frequency tracker.

Measurements of Run 102501 (October 25, 1971)

Test time was from 2:04 pm to 2:45 pm. The wind was from the south-southeast at about 5 m/sec. There were no active weather fronts in the vicinity and the sky had been clear for the day. Some pollution was evident in the air, but visibility was good.

Spectrum analyzer settings - The settings were as follows:

Sweep rate:	5 ms/cm
Sample rate:	16 sweeps/sec
Frequency Dispersion:	0.2 MHz/cm
Filter Bandwidth	10 KHz
Bandwidth:	30 KHz

The calibration frequency was 1.678 MHz as shown in Figure 50. The noise level from the detector is shown in Figure 51. The vertical scale is 200 mv/cm. A sample Doppler trace of one sweep is shown in Figure 52. As is observable, the S/N ratio is small which made data analysis difficult.

Velocity time traces - Time traces of velocity from the cup and hot wire anemometers and the LDV are shown in Figures 53 and 54. There was much more variability of wind speeds during this test than in previous tests. The smaller scale turbulence is superimposed on larger scale variations. Thus, it should be expected, as will be seen later, that the power spectra would indicate greater power at the lower frequencies. Some amount of dropout in signals is indicated for the LDV. In general comparisons of the time traces appear satisfactory.

Means and variances - The means and variances for 8 segments of a 34-minute time period are given in Table 6.

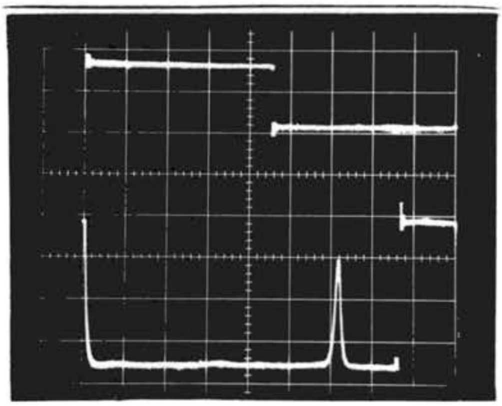


Figure 50. Calibration frequency 1.678 MHz. Test 102501

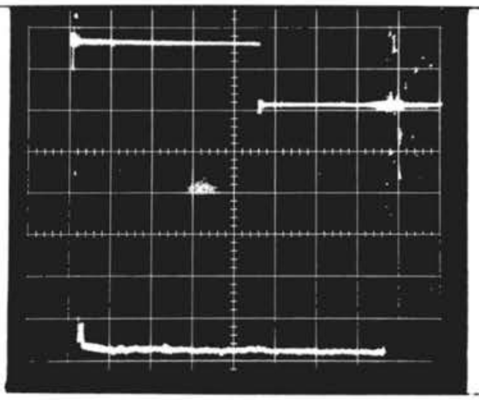


Figure 51. Noise Calibration. Vertical scale is 200 ms/cm. Test 102501

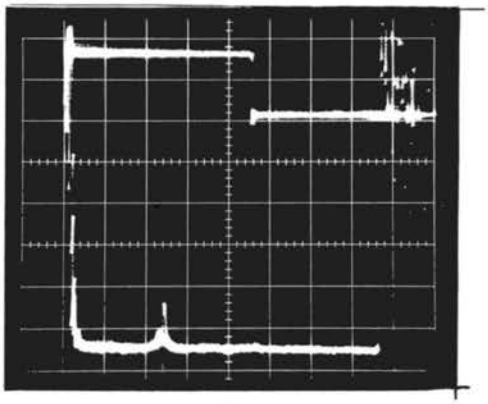


Figure 52. Sample Doppler signal. Test 102501

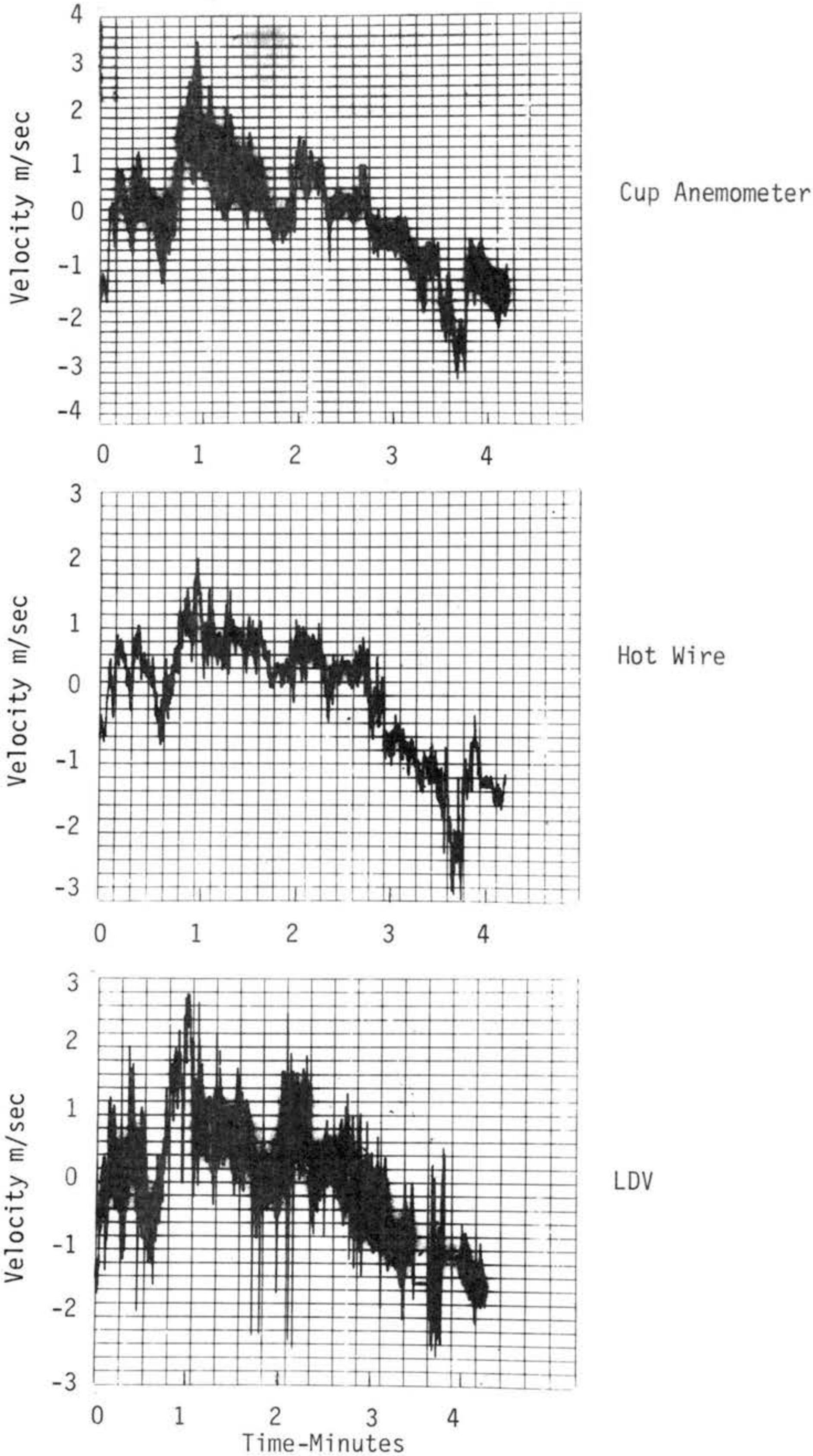


Figure 53. Time traces of wind velocity.
Test 102501, Interval 1
(For means and variances see Table 6)

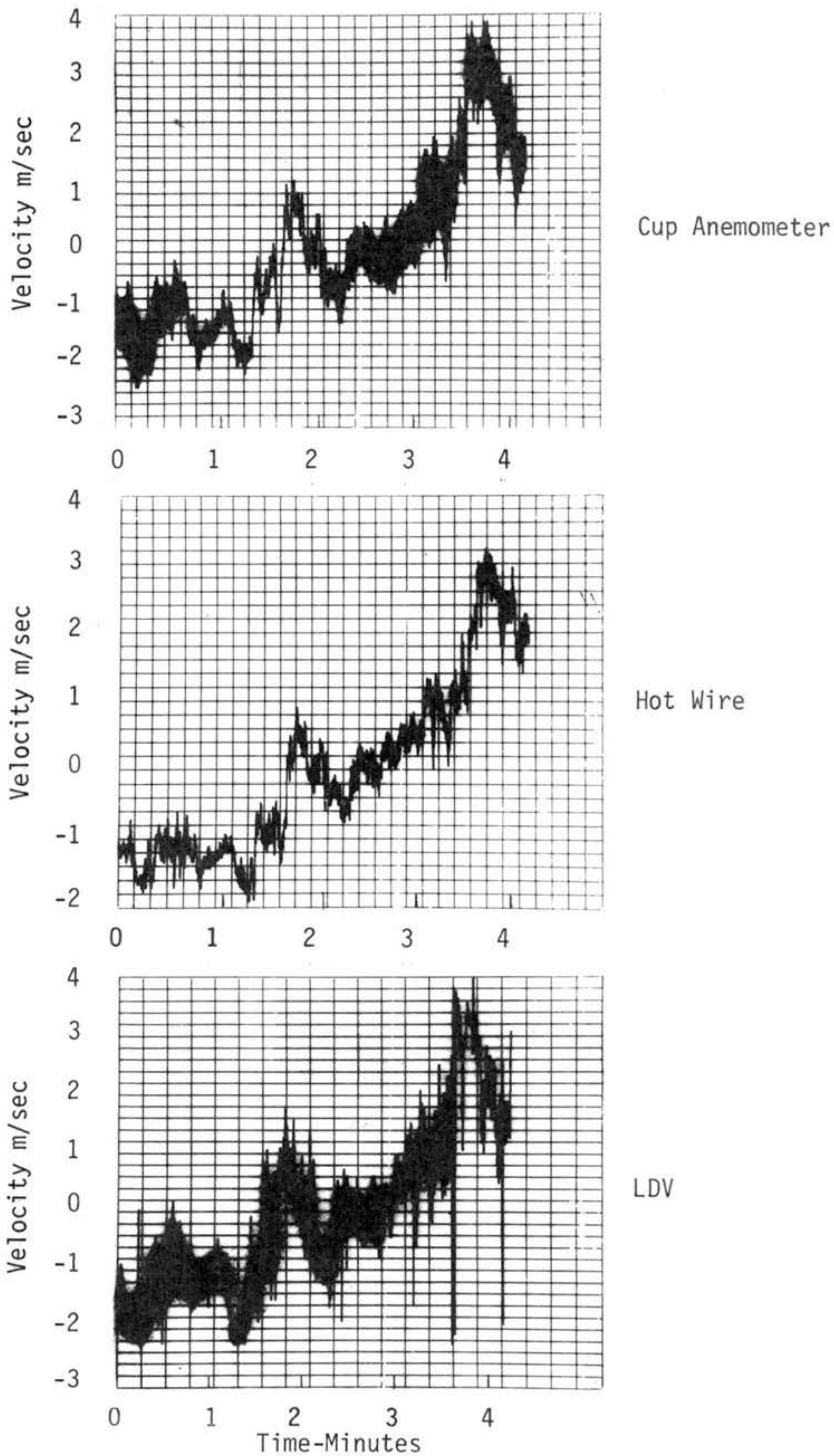


Figure 54. Time traces of wind velocity.
Test 102501, Interval 2
(For means and variances see Table 6)

TABLE 6. MEANS AND VARIANCES FOR TEST 102501

4.26-Minute Intervals	Mean Velocities m/sec			Variances $(m/sec)^2$		
	Cup	Hot Wire	LDV	Cup	Hot Wire	LDV
1	4.397	4.444	4.298	1.077	.900	.984
2	4.154	4.169	3.946	1.372	1.273	1.418
3	6.025	6.010	5.805	.762	.482	.762
4	4.943	5.000	4.683	1.450	1.162	1.436
5	5.307	5.315	4.989	.992	.717	.921
6	4.713	4.748	4.252	.873	.710	.953
7	5.082	5.102	4.878	.933	.702	.968
8	5.278	5.284	5.004	.628	.385	.666
Averages	4.987	5.009	4.732	1.011	.792	1.014

The average wind speed indicated by the LDV is within 5 percent of the cup and hot wire averages. The comparison is reasonably good.

Probability distributions - The distributions of velocities about the means for the three instruments are shown in Figure 55. Turbulence velocities are skewed to the right. The distributions are about the same as for the other tests.

Spectral densities - The spectral distributions of turbulence are shown in Figure 56. As was noted earlier the lower frequency variations of velocities produced greater power spectral densities at the lower frequencies. The cup anemometer response drops off at about 0.5 Hz, and the LDV tends to level off for frequencies greater than about 2 Hz. The comparison of spectral distributions is reasonably good.

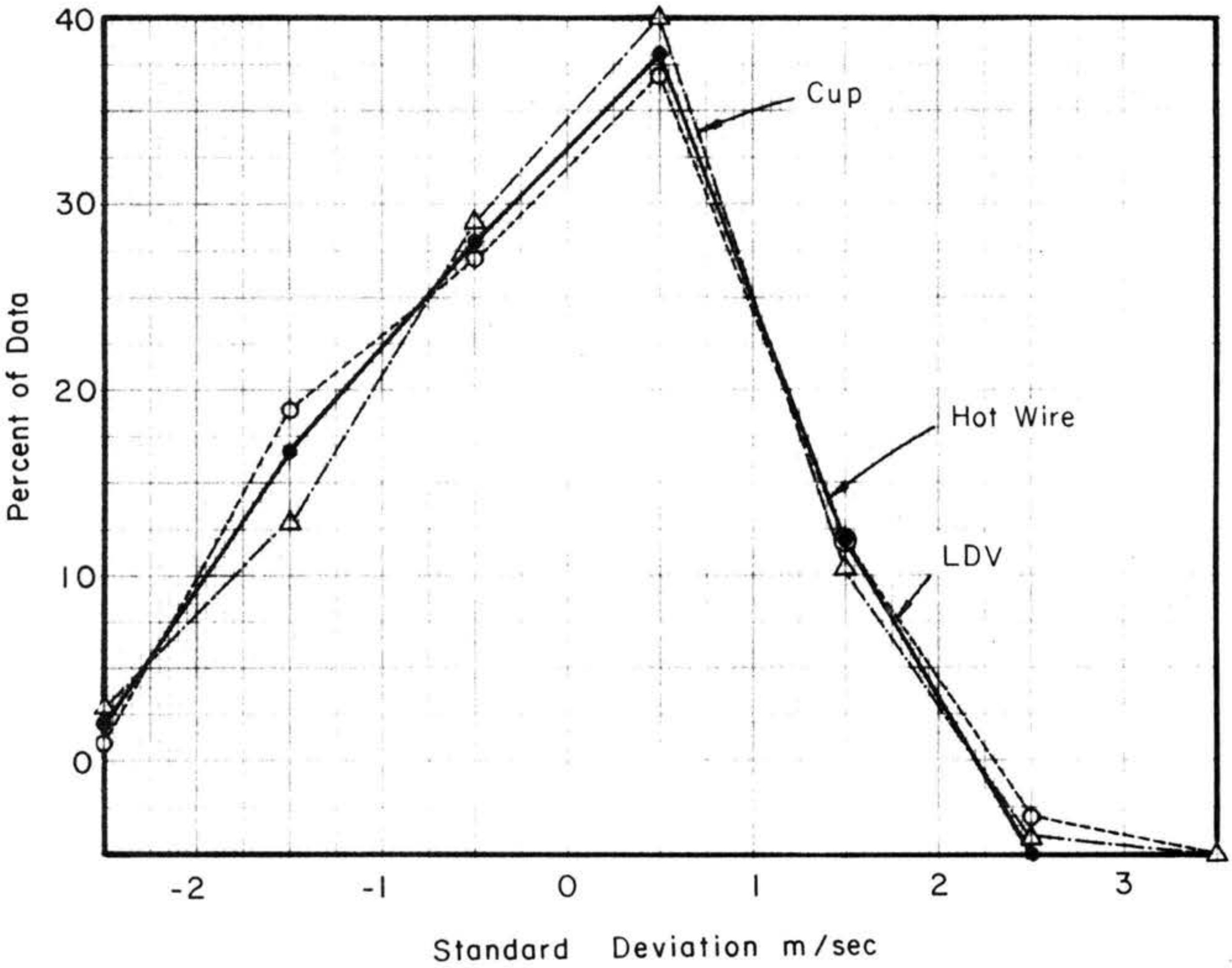


Figure 55. Distribution of velocities about the mean.
Test 102501

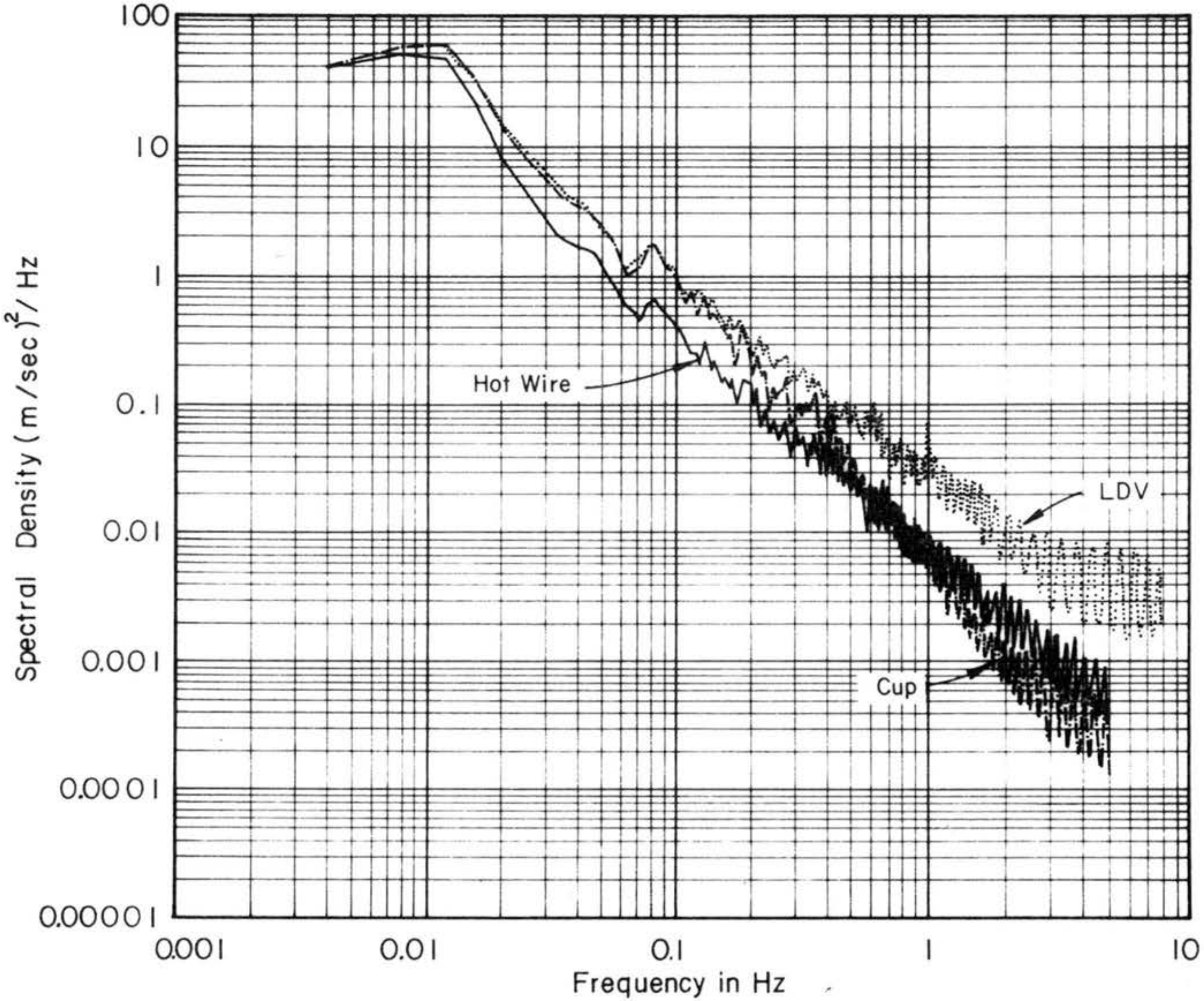


Figure 56. **Comparison of spectral density distributions.**
Test 102501

OBSERVATIONS AND CONCLUSIONS

As a consequence of the comparisons presented, the following observations can be made regarding the one-dimensional LDV system.

1. The gross features of atmospheric phenomena in the boundary layer are measured by the LDV system. The time traces show reproduction of these gross features and comparison with other anemometers are favorable.
2. Mean values determined from the LDV data are in general within 5% of other anemometer data for long (34-minute) time periods. The variations are larger for shorter time periods, chiefly because of larger variations in measured velocities. That the LDV measures larger velocities is also indicated by the probability (percent) distributions of the data and by the spectral distributions with frequency.
3. The confidence of measuring high frequency turbulence (greater than 2 Hz in atmosphere) is not yet established.
4. The technique for data reduction of the LDV data is cumbersome in its present form. Immediate improvements can be made by including on-line analog to digital equipment including a special purpose minicomputer to calculate the velocities from the digitized data. Alternatively an analog system to detect Doppler signals such as an improved frequency tracker could be used. The frequency tracker used in this study required very fine tuning, and dependable frequency lock was not achieved.

REFERENCES

Angus, J. C., D. L. Morrow, J. W. Dunning, Jr. and M. J. French (1969) Motion measurements by laser Doppler techniques. Industrial and Engineering Chemistry, Vol. 61, No. 2, pp. 9-20.

Camp, D. W. (1965) Analysis of wind tunnel data for several Beckman and Whitley series 50 and Climet C1-14 anemometers. NASA TM X-53271.

Foreman, J. W., Jr., E. W. George and R. D. Lewis (1965) Measurement of localized flow velocities in gases with a laser Doppler flowmeter. Applied Physics Letters, Vol. 7, No. 4, pp. 77-78.

Foreman, J. W., Jr., R. D. Lewis, J. R. Thornton and H. J. Watson (1966) Laser Doppler velocimeter for measurement of localized flow velocities in liquids. Proceedings of the IEEE, pp. 424-425, March.

Goldstein, R. J. and W. F. Hagen (1967) Turbulent flow measurements utilizing the Doppler shift of scattered laser radiation. Physics of Fluids, Vol. 10, pp. 1349-1352.

Goldstein, R. J. and D. K. Kreid (1967) Measurement of laminar flow development in a square duct using a laser-Doppler flowmeter. Journal of Applied Mechanics, Vol. 34, pp. 813-818.

Greated, C. A. (1969) An improved method of flow measurement in water. La Houille Blanche, No. 6, pp. 631-633.

Lockheed Missiles and Space Company (1970) Progress Report No. D162417, July.

Lockheed Missiles and Space Company (1971) NASA-MSFC field carbon dioxide laser Doppler system operating procedures. Report No. LMSC-HREC D162840, January.

Lockheed Missiles and Space Company (1971) Application of laser Doppler velocity systems. Interim Report, June.

Rolfe, E., J. K. Silk, S. Booth, K. Meister and R. M. Young (1968) Laser Doppler velocity instrument. NASA Contractor Report CR1199, Prepared by Raytheon Company.

Thomson, A. and M. F. Dorian (1967) Heterodyne detection of monochromatic light scattered from a cloud of moving particles. CDC-ERR-AN-1090, General Dynamics/Convair, San Diego, California, June.

Watson, R. C., Jr., R. D. Lewis and H. J. Watson (1969) Instruments for motion measurements using laser Doppler heterodyning techniques. ISA Transactions, Vol. 8, No. 1, pp. 20-28.

Yeh, Y. and H. Z. Cummins (1964) Localized fluid flow measurements with an He-Ne laser spectrometer. Applied Phsyics Letters, Vol. 4, No. 10, pp. 176-178.

APPENDIX A

A-1 Computer Program for Analysis of Doppler Signals

A-2 Computer Program for Determination of Velocity Profiles

A-3 Computer Program for Determination of Temperature and Humidity Profiles

APPENDIX A-1

Computer Program for Analysis of Doppler Signals

```
      PROGRAM LASDOP(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE2,TAPE1,
     .TAPE3)
      COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
     .              ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
      COMMON/BLOCK2/LENARR2,SYNC(1500),YLASER(1500),NRECOR2,NFILE2,
     .              ZEROTM2,WRIDAT2,NTAPE2
      COMMON/BCALIBR/SLOPE(2),ZEROTAP(2),SLOPEAN,ANINTER,SLOPEHW,
     .               SLOPEWD,WDINTER,SLOPEMD,DMINTER
      COMMON/BTAPECA/NCHANTP,NCALVAL,VARITP,ACTVOLT(5)
      COMMON/BINSTCA/NINSCAL,VARIIN,FULSCAW,ZEROWD,FULSCAM,ZEROMD,
     .               FRSTINT
      COMMON/BLASER/NFLYBAC,NPTSWP
      COMMON/BLASCAL/NCALREC,NSWPREC,IBEGCHK,CALEVEL,CALVELO,WAVLEN,
     .               DEVFREQ
      COMMON/BNOISCA/FLYBACK,NOISREC,XNLEVEL(275)
      COMMON/BBUFLA1/NTOTFI1,JCLOCK1,IEXIT,TIMADA,NREC
      COMMON/BSOPT1/IBEGSK1,ISKIP1,FACTOR1
      COMMON/BBUFLA2/JCLOCK2,IEXTIME,NREC2,NREC3,NREC4,TIMADJ1,TIMADJ2,
     .               TIMADJ3,TIMADJ4
      COMMON/BSOPT2/IBEGSK2,ISKIP2
      COMMON/BSKPEO1/LPACDA1,         IDENT1,NFLSKP1,NRCSKP1
      COMMON/BSKPEO2/LPACDA2,         NTOTFI2,IDENT2,NFLSKP2,NRCSKP2,
     .               NTOTAPE,IEXIT2,NTOTREC,TIMADJ
      COMMON/BCONSTM/NAVEMIR,DIRECMR,CHGMIR,TIMEMIR
      COMMON/BVOLTAD/ISCALE
      COMMON/BSPEED/SUMVELO,ISAMPLE,IDATAHW,SUMVOLT,TIMRAT1,CHANNE1,
     .              DIGRAT1,TIMECHG,VOLTCHG,MULTIME,TIMEHW,DCSUPRE
     .             ,FRSTSPD,WRITAPE,PRINTOK
      COMMON/BAVEWIN/JSAMPLE,SUMWIND,MULTIM1,TIMAVWD,AVEWD(3000)
     .               ,LASTIME
      COMMON/BBUMPUP/TIME1(702),VELOC2(702)
      COMMON TIME2(200),VELOLAS(200),IPOINT
      COMMON/UNPK1/ITIME1,ICOMWRD(201),IDATWRD(1000)
      COMMON/UNPK2/ITIME2,LCOMWRD(601),LDATWRD(3000)
      READ(5,1) VOLTCHG,WRIDAT2,CALTAPE,CALINST,CALLAS,
     .          CALNOIS,WRIDAT1,IDENT1,IDENT2,LENARR1,LENARR2,IBEGSK1,
     .          ISKIP1,IBEGSK2,ISKIP2,NAVEMIR,NFLYBAC,ISCALE,NTOTFI2,
     .          NTOTAPE,LPACDA2,NCHANTP,NCALVAL,LPACDA1,NTOTFI1,NCALREC,
     .          NSWPREC,IBEGCHK,NOISREC,NINSCAL,NLASREC,DIGRAT1,
     .          TIMECHG,TIMEHW,SLOPEHW,TIMRAT1,CHANNE1,VARITP,(ACTVOLT
     .          (I),I=1,5),TIMAVWD,CALEVEL,FLYBACK,DFLYBAC,VARIIN,
     .          FULSCAW,ZEROWD,FULSCAM,ZEROMD,DCSUPRE,DEVFREQ,WAVLEN,
     .          CHANNE2,DIGRAT2,TIMRAT2
    1 FORMAT( 9A3,11I4,/,11I4,5F6.0,/,2F6.0,5(F5.0),7F6.0,/,3F6.0,2E9.3,
     .       3F6.0)
      READ(5,5) TIMADJ1,TIMADJ2,TIMADJ3,TIMADJ4,TIMADA,NREC2,NREC3,
     .           NREC4,NCALFIL,NCALTAP,NREC
    5 FORMAT(5F10.3,3I5,2I2,I5)
      READ(5,8) FRSTAPE,FRSTINT,FRSTLAS,FRSTNOS,NTAPFI1,NLASFI2,NINSFI1,
     .          NOISFI2,WRITAPE,IRUNNO,MRCONST
    8 FORMAT(4A3,4I3,A3,I6,A3)
      WRITE(6,10) IRUNNO
   10 FORMAT(1H1,5X*INPUT DATA FOR RUN NUMBER*I7)
      WRITE(6,2)VOLTCHG,WRIDAT2,CALTAPE,CALINST,CALLAS,
     .          CALNOIS,WRIDAT1,IDENT1,IDENT2,LENARR1,LENARR2,IBEGSK1,
```

```
     .          ISKIP1,IBEGSK2,ISKIP2,NAVEMIR,NFLYBAC,ISCALE,NTOTFI2,
     .          NTOTAPE,LPACDA2,NCHANTP,NCALVAL,LPACDA1,NTOTFI1,NCALREC,
     .          NSWPREC,IBEGCHK,NOISREC,NINSCAL,NLASREC,DIGRAT1,
     .          TIMECHG,TIMEHW,SLOPEHW,TIMRAT1,CHANNE1,VARITP,(ACTVOLT
     .          (I),I=1,5),TIMAVWD,CALEVEL,FLYBACK,DFLYBAC,VARIIN,
     .          FULSCAW,ZEROWD,FULSCAM,ZEROMD,DCSUPRE,DEVFREQ,WAVLEN,
     .          CHANNE2,DIGRAT2,TIMRAT2
    2 FORMAT(1H0,* VOLTCHG =*A4* WRTDAT2 =*A4* CALTAPE =*A4* CALINST =*A
     .4* CALLAS =*A4* CALNOIS =*A4* WRIDAT1 =*A4* IDENT1 =*A4* IDENT2 =*
     .A4/* LENARR1 =*I4* LENARR2 =*I4* IBEGSK1 =*I3* ISKIP1 =*I3* IBEGSK
     .2 =*I3* ISKIP2 =*I3* NAVEMIR =*I3* NFLYBAC =*I4* ISCALE =*I2/* NTO
     .TFI2 =*I2* NTOTAPE =*I2* LPACDA2 =*I4* NCHANTP =*I3* NCALVAL =*I2
     .* LPACDA1 =*I4* NTOTFI1 =*I2* NCALREC =*I3* NSWPREC =*I2/* IBEGCHK
     . =*I3* NOISREC =*I3* NINSCAL =*I2* NLASREC =*I3/* DIGRAT1 =*F7.1
     .* TIMECHG =*F3.1* TIMEHW =*F9.4* SLOPEHW =*F9.4* TIMRAT1 =*F4.1/
     .* CHANNE1 =*F5.1* VARITP =*F5.3/* ACTVOLT(1 THRU 5) =*5F5.1/* TIMA
     .VWD =*F5.2* CALEVEL =*F9.2* FLYBACK =*F5.1* DFLYBAC =*F5.1* VARIIN
     . =*F6.3/* FULSCAW =*F9.3 * ZEROWD =*F9.3* FULSCAM =*F9.3* ZEROMD =
     .*F9.3* DCSUPRE =*F7.3* DEVFREQ =*E10.3/* WAVLEN =*E13.6* CHANNE2 =
     .*F5.1* DIGRAT2 = *F9.1* TIMRAT2 =*F6.1)
      WRITE(6,6)FRSTAPE,FRSTINT,FRSTLAS,FRSTNOS,NTAPFI1,NLASFI2,NINSFI1,
     .          NOISFI2,WRITAPE,MRCONST
    6 FORMAT(1H ,* FRSTAPE =*A4* FRSTINT =*A4* FRSTLAS =*A4* FRSTNOS =*A
     .4* NTAPFI1 =*I3* NLASFI2 =*I3* NINSFI1 =*I3* NOISFI2 =*I3* WRITAPE
     . = *A3* MRCONST =*A3)
      WRITE(6,9)TIMADJ1,TIMADJ2,TIMADJ3,TIMADJ4,TIMADA,NREC2,NREC3,NREC4
     .          ,NCALFIL,NCALTAP,NREC
    9 FORMAT(1H ,* TIMADJ1 =*F4.1* TIMADJ2 =*F10.1* TIMADJ3 =*F10.1* TIM
     .ADJ4 =*F10.1* TIMADA =*F10.1/* NREC2 =*I5* NREC3 =*I6* NREC4 =*I6
     .* NCALFIL =*I2* NCALTAP =*I2* NREC = *I4)
      REWIND 1
      REWIND 2
      IF (WRITAPE .EQ. 3HYES) REWIND 3
      FRSTPT = 3HYES
      IEXTIME = 0
      MULTIME = 1
      SUMVELO = 0.0
      ISAMPLE = 0
      SUMVOLT = 0.0
      JSAMPLE = 0
      SUMWIND = 0.0
      CALVELO = 0.0
      NSWPS = 0
      NRECOR1= 0
      NRECOR2=0
      NTRIG = 1
      NFILE1 = 1
      NFILE2 = 1
      ZEROTM1= 0.0
      ZEROTM2=0.0
      IEXIT = 3H NO
      IEXIT2 = 3H NO
      SLASTPT = 10.0
      ICHANGE = 0
      FACTOR1 = SQRT(2.)/(2.**9-1.0)
```

```
      FRSTSPD = 3HYES
      NTAPE2 = 1
      NEXTPTS = 0
      NAVEWD = 0
      NDATAPT = 1
      JCLOCK1 = 0
      JCLOCK2 = 0
      IWIND = 1
      IPOINT = 1
      JCOUNT = 0
      IF (IDENT1 .EQ. 3HYES) CALL HEADER1
      IF (IDENT2 .EQ. 3HYES) CALL HEADER2
      IF (FRSTAPE .EQ. 3H OK) GO TO 20
      NFLSKP1 = 0
      NRCSKP1 = 1
      CALL SKPEOF1
   20 IF (CALTAPE .EQ. 3HYES) CALL TAPECAL
      IF (CALTAPE .EQ. 3HNED) READ(5,11) (SLOPE(I),ZEROTAP(I),I=1,2)
11    FORMAT(4F10.3)
      IF(NFILE1 .GT. NTAPFI1) GO TO 21
      NFLSKP1 = 1
      NRCSKP1 = 0
       CALL SKPEOF1
21    IF (CALINST .EQ. 3HYES) CALL INSTCAL
      IF (CALINST .EQ. 3HNED) READ(5,11) SLOPEWD,SLOPEMD,WDINTER,DMINTER
      IF (NFILE1 .GT. NINSFI1) GO TO 23
      NFLSKP1 = 1
      NRCSKP1 = 0
      CALL SKPEOF1
   23 IF(FRSTLAS .EQ. 3H OK) GO TO 24
      NFLSKP2 = 0
      NRCSKP2 = 1
      CALL SKPEOF2
24    IF (CALLAS .EQ. 3HYES) CALL LASCAL
      IF (CALLAS .EQ. 3HNED) READ(5,12) CALVELO,NPTSWP
12    FORMAT(F10.3,I4)
      FLYBACK = FLYBACK - DFLYBAC
      IF (NFILE2 .GT. NLASFI2) GO TO 25
      NFLSKP2 = 1
      NRCSKP2 = 0
      CALL SKPEOF2
   25 IF (FRSTNOS .EQ. 3H OK) GO TO 26
      NFLSKP2 = 0
      NRCSKP2 = 1
      CALL SKPEOF2
26    IF (CALNOIS .EQ. 3HYES) CALL NOISCAL
       IF (CALNOIS .EQ. 3HNED) READ(5,4) (XNLEVEL(I),I=1,202)
    4 FORMAT((13F6.0))
      IF (NFILE2 .GT. NOISFI2) GO TO 27
      NFLSKP2 = 1
      NRCSKP2 = 0
      CALL SKPEOF2
27    CALL VOLTADJ
      IF (MRCONST .EQ. 3HYES) CALL CONSTMR
      IF (MRCONST .NE. 3HNED) GO TO 75
```

```
      READ(5,14) CHGMIR,TIMEMIR
      READ(5,13) DIRECMR
   13 FORMAT(F10.3)
   14 FORMAT(A3,F6.2)
      WRITE(6,15) DIRECMR,CHGMIR,TIMEMIR
   15 FORMAT(1H0,5X*DIRECMR =*F10.3* CHGMR =*A3* TIMEMIR =*F10.3)
   75 JCLOCK1 = 0
      ZEROTM1 = 0.0
      JCLOCK2= 0
      ZEROTM2 = 0.0
      IDATAHW = 1
      MULTIM1 = 1
      IEXTIME = 0
      LASTIME = 1
      MULTIME = 1
      PRINT 16
   16 FORMAT(1H1)
  100 CALL SPEED
      CALL AVEWIND
      IF(PRINTOK .EQ. 3HYES) WRITE(6,7)NFILE2,NTAPE2
    7 FORMAT(1H0,5X*LASER VELOCITIES*5X*FILE*I2,5X*TAPE*I2/10X*TIME,SEC*
     .10X*VELOCITY,M/SEC*10X*RECORD*)
  150 CALL BUFLAS2
      IBEGIN = 1
  175 DO 200 M=IBEGIN,LENARR2
      IF (YLASER(M) .GE. FLYBACK) GO TO 300
  200 CONTINUE
      GO TO 150
  300 JCOUNT = JCOUNT + 1
      M = M + 1
      IF (M .LE. LENARR2) GO TO 500
  400 CALL BUFLAS2
      M= 1
  500 IF (YLASER(M) .GE. FLYBACK) GO TO 300
      IF (JCOUNT .GT. 15) GO TO 600
      JCOUNT = 0
      IBEGIN = M
      IF (M .LE. LENARR2)GO TO 175
      GO TO 150
  600 M = M + IBEGCHK - 1
      IF (M .LE. LENARR2)GO TO 650
      CALL BUFLAS2
      M = M - LENARR2
  650 JCOUNT = 0
      LAST = NPTSWP-NFLYBAC
      DO 800 I= IBEGCHK, LAST
      IF (YLASER(M) .GE. XNLEVEL(I) * 2.)GO TO 900
      M= M + 1
      IF (M .LE. LENARR2) GO TO 800
  700 CALL BUFLAS2
      M = 1
  800 CONTINUE
      IBEGIN = M
      IF (FRSTPT .EQ. 3HYES) GO TO 175
      VELOLAS(IPOINT) = VELOLAS(IPOINT - 1)
```

```
      TIME2(IPOINT)= TIME2(IPOINT-1)+(NPTSWP*CHANNE2*TIMRAT2)/DIGRAT2
      IF (TIME2(IPOINT) .GT. IWIND*TIMAVWD)IWIND =IWIND + 1
      IF (AVEWD(IWIND) .NE.0.0) GO TO 825
      IF (IEXIT .EQ. 3HYES) GO TO 820
      IF (WRITAPE .EQ. 3HYES) CALL LASWRIT
      IPOINT = 1
  810 CALL SPEED
      CALL AVEWIND
      IF (PRINTOK .EQ. 3HYES) WRITE(6,7)NFILE2,NTAPE2
      GO TO 825
  820 IWIND = IWIND - 1
      IF (AVEWD(IWIND) .EQ. 0.0) GO TO 820
  825 IF (PRINTOK .NE. 3HYES) GO TO 875
      WRITE(6,3)TIME2(IPOINT),VELOLAS(IPOINT),NRECOR2
  875 IPOINT = IPOINT + 1
      GO TO 175
  900 IF (YLASER(M+1) .LT. YLASER(M)) GO TO 925
      I = I +1
      M = M + 1
      GO TO 900
  925 TIME2(IPOINT) = TIMRAT2*((ITIME2-ZEROTM2)/10000+((M  )*CHANNE2)/
     .                           DIGRAT2)
      FRSTPT = 3H NO
      IF (TIME2(IPOINT) .GE. TIMEMIR .A. CHGMIR .EQ. 3HYES) CALL CONSTMR
      IF (TIME2(IPOINT) .GT. IWIND * TIMAVWD) IWIND = IWIND + 1
      IF (AVEWD(IWIND) .NE. 0.0)GO TO 935
      IF (IEXIT .EQ. 3HYES) GO TO 930
      IF (WRITAPE .EQ. 3HYES) CALL LASWRIT
      IPOINT = 1
  926 CALL SPEED
      CALL AVEWIND
      IF (PRINTOK .EQ. 3HYES) WRITE(6,7)NFILE2,NTAPE2
      GO TO 935
  930 IWIND = IWIND - 1
      IF (AVEWD(IWIND) .EQ. 0.0) GO TO 930
  935 WDIREC = AVEWD(IWIND) - DIRECMR
      WDIREC = (WDIREC * 2. * 3.14)/ 360.
      VELOLAS(IPOINT) = ((I + 4)*CALVELO)/COS(WDIREC)
      IF (PRINTOK .NE. 3HYES) GO TO 1000
      WRITE(6,3)TIME2(IPOINT),VELOLAS(IPOINT),NRECOR2
    3 FORMAT(1H , 8XF8.3,15XF6.3,14XI4)
 1000 IPOINT = IPOINT + 1
      IBEGIN = M
      IF (M .LE. LENARR2)GO TO 175
      GO TO 150
      END
```

```
      SUBROUTINE TAPECAL
      COMMON/BTAPECA/NCHANTP,NCALVAL,VARITP,ACTVOLT(5)
      COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
     .             ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
      COMMON/BCALIBR/SLOPE(2),ZEROTAP(2),SLOPEAN,ANINTER,SLOPEHW,
     .              SLOPEWD,WDINTER,SLOPEMD,DMINTER
      DIMENSION SUMCAL(2),SUMTAP(2),SQVALUE(2),SUMACT(2),ACT X TAP(2),
     .          SUMSQ(2,5),RECMEAN(2),TOTMEAN(2,5),TEPMEAN(2),SUMEAN(2),
     .          TEMPSUM(2),STANDEV(2,5)
      ICHECK = 0
      NSAMPLE = 0
      LASTCAL = 0
      ICALVAL = 1
      DO 100 I= 1,NCHANTP
      SUMEAN(I) = 0.0
      TEMPSUM(I) = 0.0
      SUMCAL(I) = 0.0
      SUMACT(I) = 0.0
      SUMTAP(I) = 0.0
      SQVALUE(I) = 0.0
      ACT X TAP(I) = 0.0
      TEPMEAN(I) = 0.0
      RECMEAN(I) = 0.0
      DO 100 J=1,NCALVAL
      TOTMEAN(I,J) = 0.0
  100 SUMSQ(I,J) = 0.0
  110 CALL BUFLAS1
      GOTOBUF = 3H NO
      IF (ICALVAL .EQ. NCALVAL) LASTCAL = LASTCAL + 1
      DO 120 I=1,NCHANTP
      DO 120 K=1,LENARR1
  120 SUMCAL(I) = SUMCAL(I) + VOLT(I,K)
      IF (ICHECK .GT. 0) GO TO 151
      NSAMPLE = NSAMPLE + 1
      DO 140 I=1,NCHANTP
      RECMEAN(I) = SUMCAL(I) /LENARR1
      IF (NRECOR1 .EQ. 1) GO TO 125
      IF (RECMEAN(I) .GT. TOTMEAN(I,ICALVAL) + VARITP .O. RECMEAN(I)
     .    .LT. TOTMEAN(I,ICALVAL)- VARITP) GO TO 150
  125 DO 130 K=1,LENARR1
  130 SUMSQ(I,ICALVAL) = SUMSQ(I,ICALVAL) + VOLT(I,K)**2
      SUMEAN(I) = SUMEAN(I) + RECMEAN(I)
      SUMCAL(I) = 0.0
      TOTMEAN(I,ICALVAL) = SUMEAN(I)/NSAMPLE
      IF (I .EQ. 1)
     .WRITE(6,1)NRECOR1,ICALVAL,ACTVOLT(ICALVAL)
    1  FORMAT(1H0,5X*RECORD MEANS*4X*RECORD NUMBER*I4,7X*CALIBRATION*I2,
     .4X*INPUT VALUE*F5.1/11X*CHANNEL*10X*MEAN*13X*CUMULATIVE MEAN* 6X
     .*NUMBER RECORDS FOR CUMULATIVE MEAN*)
  140 WRITE(6,2)I,RECMEAN(I),TOTMEAN(I,ICALVAL),NSAMPLE
    2 FORMAT(1H ,12X,I2,10X,F8.4,14X,F8.4,25X,I3)
      GO TO 110
  150 NSAMPLE = NSAMPLE - 1
      ICALVAL = ICALVAL + 1
  151 IF (ICALVAL .GT. NCALVAL .A. LASTCAL .GT. 3) GO TO 180
```

```
      ICHECK = ICHECK + 1
      WRITE(6,5)NRECOR1,ICALVAL,ACTVOLT(ICALVAL)
    5  FORMAT(1H0,5X*TEMPORARY MEANS*8X*RECORD NUMBER*I4,10X*CALIBRATION
     .*I2,4X*INPUT VALUE *F5.1/11X*CHANNEL*10X*MEAN*)
      DO 170 I=1,NCHANTP
      RECMEAN(I) = SUMCAL(I)/LENARR1
      WRITE(6,6)I,RECMEAN(I)
    6 FORMAT(1H ,12XI2,10XF8.4)
      SUMCAL(I) = 0.0
      IF (ICHECK .EQ. 1) GO TO 160
      DO 155 K=1,LENARR1
  155 SUMSQ(I,ICALVAL) = SUMSQ(I,ICALVAL) + VOLT(I,K)**2
      TEMPSUM(I) = TEMPSUM(I) + RECMEAN(I)
  160 IF (RECMEAN(I) .GT. TOTMEAN(I,ICALVAL-1) + VARITP .O. RECMEAN(I)
     .     .LT. TOTMEAN(I,ICALVAL-1)-VARITP) GOTOBUF = 3HYES
  170 CONTINUE
      IF(ICHECK .GT. 3) GO TO 180
      IF(GOTOBUF .EQ. 3HYES) GO TO 110
      DO 175 I=1,NCHANTP
      TEMPSUM(I) = 0.0
  175 SUMSQ(I,ICALVAL) = 0.0
      ICHECK = 0
      ICALVAL = ICALVAL - 1
      GO TO 110
  180 IEND = ICALVAL - 1
      WRITE(6,8)IEND,ACTVOLT(IEND)
    8 FORMAT(1H0,/,5X*STANDARD DEVIATIONS*10X*CALIBRATION*I2,5X*INPUT VA
     .LUE*F5.1/11X*CHANNEL*10X*RMS*)
      DO 190 I=1,NCHANTP
      STANDEV(I,IEND) = SQRT(SUMSQ(I,IEND)/(NSAMPLE*LENARR1) -
     .                    TOTMEAN(I,IEND)**2)
  190 WRITE(6,9)I,STANDEV(I,IEND)
    9 FORMAT(1H ,12XI2,7XF9.3)
      NSAMPLE = ICHECK - 1
      DO 195 I=1,NCHANTP
      SUMEAN(I) = TEMPSUM(I)
      TOTMEAN(I,ICALVAL) = TEMPSUM(I)/NSAMPLE
  195 TEMPSUM(I) = 0.0
      ICHECK = 0
      IF (ICALVAL .LE. NCALVAL) GO TO 110
      WRITE(6,10)NRECOR1
   10 FORMAT(1H0,5X*ACTUAL VS TAPE VOLTAGE*10X*LEAST SQUARE METHOD*5X*NU
     .MBER RECORDS USED FOR CALCULATIONS*I3)
      DO 210 I=1,NCHANTP
      DO 200 J=1,NCALVAL
      SUMTAP(I) = SUMTAP(I) + TOTMEAN(I,J)
      SQVALUE(I) = SQVALUE(I) + TOTMEAN(I,J)**2
      ACT X TAP(I) = ACT X TAP(I) + TOTMEAN(I,J)*ACTVOLT(J)
  200 SUMACT(I) = SUMACT(I) + ACTVOLT(J)
      SLOPE(I) = (SUMACT(I) *SUMTAP(I) -NCALVAL* ACT X TAP(I))/
     .            (SUMACT(I)**2- NCALVAL*SQVALUE(I))
      ZEROTAP(I) = (SUMACT(I)*ACT X TAP(I) - SUMTAP(I)*SQVALUE(I))/
     .            (SUMACT(I)**2- NCALVAL*SQVALUE(I))
      WRITE(6,11)I,(ACTVOLT(J),TOTMEAN(I,J),J=1,NCALVAL)
   11 FORMAT(1H0,10X*C H A N N E L*I3/15X*VALUES USED FOR LEAST SQUARE C
```

```
         .ALCULATIONS*10X*INPUT VALUE*5X*TAPE VALUE*/(69XF4.1,11XF6.3))
     210 WRITE(6,12)SLOPE(I),ZEROTAP(I)
      12 FORMAT(1H0,15X*VALUES OBTAINED FROM LEAST SQUARE CALCULATIONS* 7X*
         .SLOPE*8X*INTERCEPT*/68XF5.3,11XF5.3)
         PRINT 13
     13  FORMAT(1H1)
         RETURN
         END
```

```
      SUBROUTINE INSTCAL
      COMMON/BINSTCA/NINSCAL,VARIIN,FULSCAW,ZEROWD,FULSCAM,ZEROMD,
     .            FRSTINT
      COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
     .             ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
      COMMON/BCALIBR/SLOPE(2),ZEROTAP(2),SLOPEAN,ANINTER,SLOPEHW,
     .             SLOPEWD,WDINTER,SLOPEMD,DMINTER
      DIMENSION SUMSQWD(3),SUMSQMD(3),TMEANWD(3),TMEANMD(3),STDEVWD(3),
     .           STDEVMD(3)
      LEVEL = 10HZERO INPUT
      NRECOR1 = 0
      IF (FRSTINT .EQ. 3HYES) NRECOR1 = 1
      SUMWD = 0.0
      SUMMD = 0.0
      SUMAVEW = 0.0
      SUMAVEM = 0.0
      NSAMPLE = 0
      INSCAL = 1
      TEMPWD = 0.0
      TEMPMD = 0.0
      ICHECK = 0
      DO 100 I = 1,NINSCAL
      SUMSQWD(I) = 0.0
  100 SUMSQMD(I) = 0.0
      IF (NRECOR1 .EQ. 1) GO TO 175
  150 CALL BUFLAS1
      GOTOBUF = 3H NO
      IF (NFILE1 .GT. 1) GO TO 850
  175 DO 200 K=1,LENARR1
      SUMWD = SUMWD + WINDIRE(K)
  200 SUMMD = SUMMD + DIRMIRR(K)
      AVEWD = SUMWD/LENARR1
      AVEMD = SUMMD/LENARR1
      SUMWD = 0.0
      SUMMD = 0.0
      IF (NRECOR1 .EQ. 1) GO TO 300
      IF (ICHECK .GT. 0) GO TO 600
      IF (AVEWD .GT. TMEANWD(INSCAL) + VARIIN) GO TO 500
  300 SUMAVEW = SUMAVEW + AVEWD
      SUMAVEM = SUMAVEM + AVEMD
      NSAMPLE = NSAMPLE + 1
      TMEANWD(INSCAL) = SUMAVEW/NSAMPLE
      TMEANMD(INSCAL) = SUMAVEM/NSAMPLE
      WRITE(6,1)NRECOR1,LEVEL,AVEMD,TMEANMD(INSCAL),NSAMPLE,AVEWD,
     .          TMEANWD(INSCAL),NSAMPLE
    1 FORMAT(1H0,5X*INSTRUMENT CALIBRATION*5X*RECORD MEANS*5X*RECORD*I3,
     .5X*INPUT *A10/10X*INSTRUMENT*10X*RECORD MEAN*10X*CUMULATIVE MEAN*
     .10X*NUMBER OF RECORDS IN CUMULATIVE MEAN*/7X*MIRROR DIRECTION*9XF6
     ..3,19XF6.3,30XI2/ 8X*WIND DIRECTION*10XF6.3,19XF6.3,30XI2)
      DO 400 K=1,LENARR1
      SUMSQWD(INSCAL) = SUMSQWD(INSCAL) + WINDIRE(K)**2
  400 SUMSQMD(INSCAL) = SUMSQMD(INSCAL) + DIRMIRR(K) **2
      GO TO 150
  500 INSCAL = INSCAL + 1
  600 ICHECK = ICHECK + 1
```

```
      IF (ICHECK .LT. 2) GO TO 150
      TEMPWD = TEMPWD + AVEWD
      TEMPMD = TEMPMD + AVEMD
      IEND = ICHECK - 1
      LEVEL = 10HFULL SCALE
      WRITE(6,2)NRECOR1,LEVEL,AVEMD,TEMPMD,IEND,AVEWD,TEMPWD,IEND
    2 FORMAT(1H0,5X*TEMPORARY SUMS*10X*RECORD*I3,10X*INPUT *A10/ 10X*INS
     .TRUMENT*10X*RECORD MEAN*10X*SUM OF RECORD MEANS*10X*NUMBER OF RECO
     .RDS IN SUM*/ 7X*MIRROR DIRECTION*9XF6.3,20XF6.3,25XI2/8X *WIND DIR
     .ECTION*10XF6.3,20XF6.3,25XI2)
      DO 700 K=1,LENARR1
      SUMSQWD(INSCAL) = SUMSQWD(INSCAL) + WINDIRE(K)**2
  700 SUMSQMD(INSCAL) = SUMSQMD(INSCAL) + DIRMIRR(K)**2
  800 IF (AVEWD .GT. TMEANWD(INSCAL-1) + VARIIN) GOTOBUF = 3HYES
      IF (ICHECK .GT. 3) GO TO 900
      IF(GOTOBUF .EQ. 3HYES) GO TO 150
      SUMSQWD(INSCAL) = 0.0
      SUMSQMD(INSCAL) = 0.0
      ICHECK = 0
      TEMPWD = 0.0
      TEMPMD = 0.0
      INSCAL = INSCAL - 1
      GO TO 150
  850 INSCAL = INSCAL + 1
  900 IEND = INSCAL - 1
      LEVEL = 10HZERO INPUT
      IF (IEND .EQ. 2) LEVEL = 10HFULL SCALE
      STDEVWD(IEND) = SQRT(SUMSQWD(IEND)/(NSAMPLE*LENARR1) -
     .                     TMEANWD(IEND)**2)
      STDEVMD(IEND) = SQRT(SUMSQMD(IEND)/(NSAMPLE*LENARR1) -
     .                     TMEANMD(IEND)**2)
      WRITE(6,3)IEND,LEVEL,STDEVMD(IEND),STDEVWD(IEND)
    3 FORMAT(1H0,5X*STANDARD DEVIATIONS*10X*CALIBRATION*I2,10X*INPUT *
     .A10/10X*INSTRUMENT*10X*RMS*/7X*MIRROR DIRECTION* 6XF5.3/ 8X*WIND D
     .IRECTION*7XF6.3)
      LEVEL = 10HFULL SCALE
      NSAMPLE = ICHECK - 1
      SUMAVEW=TEMPWD
      SUMAVEM = TEMPMD
      TMEANWD(INSCAL) = SUMAVEW/NSAMPLE
      TMEANMD(INSCAL) = SUMAVEM/NSAMPLE
      TEMPWD = 0.0
      TEMPMD = 0.0
      ICHECK = 0
      IF (INSCAL .LE. NINSCAL) GO TO 150
      SLOPEWD = (FULSCAW-ZEROWD)/(TMEANWD(2)-TMEANWD(1))
      SLOPEMD = (FULSCAM-ZEROMD)/(TMEANMD(2)-TMEANMD(1))
      DMINTER = FULSCAM - SLOPEMD*TMEANMD(2)
      WDINTER = FULSCAW - SLOPEWD*TMEANWD(2)
      WRITE(6,4)NRECOR1,TMEANMD(1),ZEROMD,TMEANMD(2),FULSCAM,SLOPEMD,
     .DMINTER,TMEANWD(1),ZEROWD,TMEANWD(2),FULSCAW,SLOPEWD,WDINTER
    4 FORMAT(1H0,5X*INSTRUMENT CALIBRATION*5X*NUMBER OF RECORDS USED*I3/
     . 10X*MIRROR DIRECTION*/15X*VALUES USED FOR CALIBRATION*10X*INPUT*5
     .X *TAPE VALUE*5X*ACTUAL VALUE*/ 53X*ZERO*6XF6.3,10XF7.3/ 49X*FULL
     .SCALE*4XF6.3,10XF7.3/ 15X*VALUES OBTAINED*22X*SLOPE*5X*INTERCEPT*/
```

```
            .51XF7.3,6XF5.3//10X*WIND DIRECTION*/ 15X*VALUES USED FOR CALIBRATI
            .ON*10X*INPUT*5X *TAPE VALUE*5X*ACTUAL VALUE*/53X*ZERO*6XF6.3,10XF7
            ..3/ 49X*FULL SCALE*4XF6.3,10XF7.3/15X*VALUES OBTAINED*22X*SLOPE*5X
            .*INTERCEPT*/51XF7.3,6XF7.3)
115          RETURN
             END
```

```
      SUBROUTINE LASCAL
      COMMON/BLASCAL/NCALREC,NSWPREC,IBEGCHK,CALEVEL,CALVELO,WAVLEN,
     .               DEVFREQ
      COMMON/BLOCK2/LENARR2,SYNC(1500),YLASER(1500),NRECOR2,NFILE2,
     .              ZEROTM2,WRIDAT2,NTAPE2
      COMMON/BLASER/NFLYBAC,NPTSWP
      DIMENSION NPTRIG(20)
      NEXT = 0
      NPTSWP = 0
      SLASTPT = 10.0
      CALVELO = 0.0
      NSAMPLE = 0
      NSWPS = 0
      LASTPT = 0
      GO = 3H NO
      LEFTOVR = 0
  100 CALL BUFLAS2
      IF (NRECOR2 .GT. NCALREC) GO TO 900
  150 NTRIG = 1
      DO 50 I=1,20
   50 NPTRIG(I) = 0
      DO 500 K=1,LENARR2
      IF (K .GT. 1) GO TO 200
      IF (SYNC(K) .GT. 0.0 .A. SLASTPT .LT. 0.0) GO TO 300
      GO TO 500
  200 IF (SYNC(K) .GT. 0.0 .A. SYNC(K-1) .LT. 0.0) GO TO 300
      GO TO 500
  300 IF (GO .EQ. 3H NO) GO TO 400
      NPTSWP = NPTSWP + K - LASTPT + LEFTOVR - 1
      NSWPS = NSWPS + 1
      LEFTOVR = 0
  400 NPTRIG(NTRIG) = K  - 1
      NTRIG = NTRIG + 1
      LASTPT = K - 1
      GO = 3HYES
  500 CONTINUE
      SLASTPT = SYNC(LENARR2)
      LEFTOVR = LENARR2 - NPTRIG(NTRIG-1)
      LASTPT = 0
      NEXT = 0
      NTRIG = 1
      NAVEPTS = NPTSWP / NSWPS
  600 ISTART = NPTRIG(NTRIG) + IBEGCHK
      LAST = NPTRIG(NTRIG) + NAVEPTS - NFLYBAC
      IF (LAST .GT. LENARR2) NEXT = LAST - LENARR2
      IF (LAST .GT. LENARR2)LAST=LENARR2
      DO 700 I = ISTART,LAST
      IF (YLASER(I) .GT. CALEVEL) NEXT = 0
      IF (YLASER(I) .GT. CALEVEL) GO TO 800
  700 CONTINUE
      IF (NEXT .EQ. 0) GO TO 750
      CALL BUFLAS2
      DO 725 J=1,NEXT
      IF (YLASER(J) .LE. CALEVEL) GO TO 725
      I=J+LENARR2
```

```
      GO TO 800
  725 CONTINUE
      IF (NRECOR2 .GT. NCALREC) GO TO 900
      GO TO 150
  750 NTRIG = NTRIG + 1
      IF ( NPTRIG(NTRIG) .GT. 0) GO TO 600
      IF (NRECOR2 .GT. NCALREC) GO TO 900
      GO TO 100
  800 IF (YLASER(I+1) .GT. YLASER(I)) I=I +1
      CALVELO = CALVELO + I - NPTRIG(NTRIG)
      NSAMPLE = NSAMPLE + 1
      NTRIG = NTRIG + 1
      IF (NRECOR2 .GT. NCALREC) GO TO 900
      IF (NEXT .GT. 0) GO TO 150
      IF (NPTRIG(NTRIG) .GT. 0) GO TO 600
      GO TO 100
  900 CALVELO = CALVELO/NSAMPLE
      NPTSWP = NPTSWP / NSWPS
      WRITE(6,5)NRECOR2,CALVELO,NPTSWP
    5 FORMAT(1H1,5X*DEVIATION FREQUENCY CALIBRATION*5X*NUMBER OF RECORDS
     . USED FOR CALIBRATION*I3/95X*WAVELENGTH X DEV. FREQ.*/10X*AVERAGE
     .NUMBER OF POINTS TO DEVIATION FREQUENCY*5X*AVERAGE NUMBER OF POINT
     .S/SWP*5X*--------------------------*/ 95X*2 X POINTS TO DEV. FREQ.*/
     . 32XF5.1,39XI3)
      CALVELO = (WAVLEN*DEVFREQ)/(2*CALVELO)
      WRITE(6,6)CALVELO
    6 FORMAT(1H+,104XF5.4)
      RETURN
      END
```

```
      SUBROUTINE NOISCAL
      COMMON/BNOISCA/FLYBACK,NOISREC,XNLEVEL(275)
      COMMON/BLOCK2/LENARR2,SYNC(1500),YLASER(1500),NRECOR2,NFILE2,
     .              ZEROTM2,WRIDAT2,NTAPE2
      COMMON/BLASER/NFLYBAC,NPTSWP
      DIMENSION SUMPTS(300)
      DO 100 I=1,NPTSWP
  100 SUMPTS(I) = 0.0
  150 CALL BUFLAS2
      K = 1
      JCOUNT = 0
  175 DO 200 I=K,LENARR2
      IF (YLASER(I) .GT. FLYBACK) GO TO 300
  200 CONTINUE
      IF (NRECOR2 .GE. NOISREC) GO TO 800
      GO TO 150
  300 DO 400 J=I,LENARR2
      JCOUNT = JCOUNT + 1
      IF (YLASER(J) .LT. FLYBACK) GO TO 450
  400 CONTINUE
      IF (NRECOR2 .GE. NOISREC) GO TO 800
      CALL BUFLAS2
      I = 1
      GO TO 300
  450 IF (JCOUNT .GT. 15) GO TO 500
      JCOUNT = 0
      K=J
      IF (K.LE. LENARR2) GO TO 175
      IF (NRECOR2 .GE. NOISREC) GO TO 800
      GO TO 150
  500 M=1
  550 ISTART = J - 1
      DO 600 K=ISTART,LENARR2
      SUMPTS(M) = SUMPTS(M) + YLASER(K)
      M=M+1
      IF (M .GT. NPTSWP - NFLYBAC) GO TO 700
  600 CONTINUE
      IF (NRECOR2 .GE. NOISREC) GO TO 800
      CALL BUFLAS2
      J= 1
      GO TO 550
  700 NSAMPLE = NSAMPLE + 1
      JCOUNT = 0
      IF (K .LE. LENARR2) GO TO 175
      IF (NRECOR2 .LT. NOISREC) GO TO 150
  800 LAST = NPTSWP - NFLYBAC
      DO 900 I = 1,LAST
  900 XNLEVEL(I) = SUMPTS(I) / NSAMPLE
      WRITE (6,1) (XNLEVEL(I),I=1,LAST)
    1 FORMAT (1H0,/,* NOISE LEVELS*,/,(1X,15(F8.3)))
      DO 1000 I=9,LAST
 1000 XNLEVEL(I) = 40.0
      RETURN
      END
```

```
      SUBROUTINE BUFLAS1
      COMMON/BBUFLA1/NTOTFI1,JCLOCK1,IEXIT,TIMADA,NREC
      COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
     .               ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
      COMMON/BSKPEO1/LPACDA1,        IDENT1,NFLSKP1,NRCSKP1
      COMMON/UNPK1/ITIME1,ICOMWRD(201),IDATWRD(1000)
      CORTIM1 = 3H NO
  100 BUFFER IN(1,1) (ITIME1,ICOMWRD(LPACDA1))
      IF (UNIT(1)) 500,200,400
  200 WRITE(6,1) NRECOR1,NFILE1
    1 FORMAT(1H0,* THERE ARE*I5* RECORDS ON FILE*I3* UNIT 1.  ENCOUNTERE
     .D IN BUFLAS1*)
      NRECOR1 = 0
      NFILE1 = NFILE1 + 1
      IF (NFILE1 .GT. NTOTFI1) GO TO 300
      IF (IDENT1 .EQ. 3HYES) CALL HEADER1
      GO TO 100
  300 IEXIT= 3HYES
      RETURN
  400 NRECOR1 = NRECOR1 + 1
      LEN = LENGTH(1)
      WRITE(6,2) NRECOR1,NFILE1,LEN
    2 FORMAT (1H0,* PARITY ERROR ON NEXT DATA, RECORD*I5* FILE*I4,5X* NU
     .MBER OF COMPUTER WORDS*I4)
      IF(LEN .NE. LPACDA1) GO TO 100
      CALL UNPAK1
      CALL SORT1
      IF (WRIDAT1 .EQ. 3H NO) CALL DATWRI1
      GO TO 600
  500 NRECOR1 = NRECOR1 + 1
      LEN = LENGTH(1)
      IF (LEN .NE. LPACDA1) GO TO 700
      CALL UNPAK1
      CALL SORT1
  600 IF (NRECOR1 .EQ. 1) ZEROTM1 = ITIME1
      IF (ITIME1-999999 .GT. -12000) CORTIM1=3HYES
      ITIME1 =ITIME1 +JCLOCK1*999999 + TIMADA/(NREC-1)*(NRECOR1-1)
      IF (CORTIM1 .EQ. 3HYES) JCLOCK1 = JCLOCK1 + 1
      RETURN
  700 WRITE(6,3)NRECOR1,NFILE1,LEN
    3 FORMAT(1H0,* RECORD ENCOUNTERED OF IMPROPER LENGTH ON UNIT 1. RECO
     .RD*I4* FILE*I2* NUMBER OF COMPUTER WORDS*I4)
      GO TO 100
  800 RETURN
      END
```

```
                  SUBROUTINE HEADER1
                  COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
                 .                ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
                  COMMON/BHEAD1/ID(9)
              50  BUFFER IN(1,0)(ID(1),ID(9))
                  IF (UNIT(1))300,200,100
             200  PRINT 1,NFILE1
               1  FORMAT(1H0,* EOF READ IN HEADER ON FILE*I2* UNIT 1*)
                  GO TO 50
             100  PRINT 2, NFILE1
              2   FORMAT(1H0* PARITY ERROR IN HEADER ON FILE*I2* UNIT 1*)
             300  LEN = LENGTH(1)
                  PRINT 3, NFILE1,(ID(I),I=1,2),LEN
              3   FORMAT(1H0,* ID ON UNIT 1, FILE*I2* IS *2A10* NUMBER OF COMPUTER W
                 .ORDS*I4)
                  RETURN
                  END
```

```
      SUBROUTINE SORT1
      COMMON/BSORT1/IBEGSK1,ISKIP1,FACTOR1
      COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
     .               ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
      COMMON/UNPK1/ITIME1,ICOMWRD(201),IDATWRD(1000)
      M= IBEGSK1
      DO 100 I=1,LENARR1
      WINDIRE(I) = IDATWRD(M) * FACTOR1
      VOLT(1,I) = IDATWRD(M+1) * FACTOR1
      DIRMIRR(I) = IDATWRD(M+2) * FACTOR1
      VOLT(2,I) = IDATWRD(M+3) * FACTOR1
  100 M= M+ ISKIP1
      IF (WRIDAT1 .EQ. 3HYES) CALL DATWRI1
      RETURN
      END
```

```
      SUBROUTINE DATWRI1
      COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
     .             ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
      COMMON/UNPK1/ITIME1,ICOMWRD(201),IDATWRD(1000)
      WRITE (6,1) NRECOR1,ITIME1
    1 FORMAT (1H1,* RECORD NUMBER *I4,6X* ITIME1*I6)
      WRITE (6,2) (VOLT(1,I),I=1,LENARR1)
    2 FORMAT (1H0,/,(1X,10(F10.5,2X)))
      WRITE (6,2) (VOLT(2,I),I=1,LENARR1)
      WRITE (6,2) (WINDIRE(I),I=1,LENARR1)
      WRITE (6,2) (DIRMIRR(I),I=1,LENARR1)
      RETURN
      END
```

```
      SUBROUTINE BUFLAS2
      COMMON/BBUFLA2/JCLOCK2,IEXTIME,NREC2,NREC3,NREC4,TIMADJ1,TIMADJ2,
     .                TIMADJ3,TIMADJ4
      COMMON/BLOCK2/LENARR2,SYNC(1500),YLASER(1500),NRECOR2,NFILE2,
     .               ZEROTM2,WRIDAT2,NTAPE2
      COMMON/BSKPE02/LPACDA2,          NTOTFI2,IDENT2,NFLSKP2,NRCSKP2,
     .                NTOTAPE,IEXIT2,NTOTREC,TIMADJ
      COMMON/BSPEED/SUMVELO,ISAMPLE,IDATAHW,SUMVOLT,TIMRAT1,CHANNEL,
     .               DIGRAT1,TIMECHG,VOLTCHG,MULTIME,TIMEHW,DCSUPRE
     .              ,FRSTSPD,WRITAPE,PRINTOK
      COMMON/BWRITE/TIME2(1500),VELOLAS(1500),IPOINT
      COMMON/UNPK2/ITIME2,LCOMWRD(601),LDATWRD(3000)
      CORTIM2 = 3H NO
  100 BUFFER IN(2,1)(ITIME2,LCOMWRD(LPACDA2))
      IF (UNIT(2)) 400,200,300
  200 WRITE(6,1) NRECOR2,NFILE2,NTAPE2
    1 FORMAT (1H0,* THERE ARE*I6* RECORDS ON FILE*I3* TAPE*I2)
      NFILE2 = NFILE2 + 1
      NRECOR2 = 0
      IF (NFILE2 .GT. NTOTFI2) GO TO 250
  225 IF (IDENT2 .EQ. 3HYES) CALL HEADER2
      GO TO 100
  250 NTAPE2 = NTAPE2 + 1
      IF (NTAPE2 .GT. NTOTAPE) GO TO 600
      NFILE2 = 1
      NTOTFI2 = 1
      CALL UNLOADW(2)
      JCLOCK2 = 0
      IEXTIME = ISTORTM
  260 GO TO (220,230,240),NTAPE2
  220 NTOTREC = NREC2
      TIMADJ = TIMADJ2
      GO TO 225
  230 NTOTREC = NREC3
      TIMADJ = TIMADJ3
      GO TO 225
  240 NTOTREC = NREC4
      TIMADJ = TIMADJ4
      GO TO 225
  300 NRECOR2 = NRECOR2 +1
      LEN = LENGTH(2)
      WRITE (6,3) NRECOR2,NFILE2,NTAPE2,LEN
    3 FORMAT (1H0,* PARITY ERROR ON RECORD*I6* FILE*I3* TAPE*I2* NUMBER
     .OF COMPUTER WORDS*I4)
      IF (LEN .NE. LPACDA2 ) GO TO 100
      CALL UNPAK2
      CALL SORT2
      IF (WRIDAT2 .EQ. 3H NO) CALL DATWRI2
      GO TO 500
  400 NRECOR2 = NRECOR2 + 1
      LEN = LENGTH(2)
      IF (LEN .EQ. LPACDA2) GO TO 450
      WRITE(6,4)LEN,NRECOR2,NFILE2,NTAPE2
    4 FORMAT(1H0,* ENCOUNTERED RECORD OF IMPROPER LENGTH. LENGTH WAS*I3*
     . COMPUTER WORDS. THIS OCCURRED ON RECORD*I5* FILE*I2* TAPE*I2* ON
```

```
     .UNIT 2*)
      GO TO 100
  450 CALL UNPAK2
      CALL SORT2
  500 IF (NTOTREC .EQ. 0) RETURN
      IF (NRECOR2 .EQ. 1) ZEROTM2 = ITIME2
      IF (ITIME2 - 999999 .GT. -935) CORTIM2 = 3HYES
      ITIME2 = ITIME2 + JCLOCK2*999999+(TIMADJ/(NTOTREC-1))*(NRECOR2-1)
     . + IEXTIME
  550 IF (CORTIM2 .EQ. 3HYES) JCLOCK2 = JCLOCK2 + 1
      ISTORTM = ITIME2
      IF (NRECOR2 .LE. NTOTREC)RETURN
      WRITE(6,2)NRECOR2,NTAPE2
  2   FORMAT(1H0,5X*REACHED RECORD*I5* ON TAPE*I2* WITHOUT EOF*)
      GO TO 200
  600 IF (WRITAPE .NE. 3HYES) CALL EXIT
      LENARR2 = 1
      CALL LASWRIT
  700 ENDFILE 3
      ENDFILE 3
      ENDFILE 3
      ENDFILE 3
      REWIND 3
      CALL EXIT
      RETURN
      END
```

```
      SUBROUTINE HEADER2
      COMMON/BLOCK2/LENARR2,SYNC(1500),YLASER(1500),NRECOR2,NFILE2,
     .                ZEROTM2,WRIDAT2,NTAPE2
      COMMON/BHEAD2/ID(9)
   50 BUFFER IN(2,0) (ID(1),ID(9))
      IF (UNIT(2)) 300,100,200
  100 PRINT 1,NFILE2,NTAPE2
    1 FORMAT(1H0,* EOF IN HEADER ON FILE*I2* TAPE*I2* ON UNIT 2.*)
      GO TO 50
  200 PRINT 2, NFILE2,NTAPE2
    2 FORMAT(1H0,* PARITY ERROR IN HEADER ON FILE*I2* TAPE*I2* UNIT 2*)
  300 LEN = LENGTH(2)
      PRINT 3, NFILE2, NTAPE2,(ID(I),I=1,2),LEN
    3 FORMAT(1H0,* ID ON FILE*I2* TAPE*I2* UNIT 2 IS *2A10* NUMBER OF CO
     .MPUTER WORDS*I4)
      RETURN
      END
```

```
      SUBROUTINE SORT2
      COMMON/BSORT2/IBEGSK2,ISKIP2
      COMMON/BLOCK2/LENARR2,SYNC(1500),YLASER(1500),NRECOR2,NFILE2,
     .              ZEROTM2,WRIDAT2,NTAPE2
      COMMON/UNPK2/ITIME2,LCOMWRD(601),LDATWRD(3000)
      M=IBEGSK2
      DO 100 I=1,LENARR2
      SYNC(I) = LDATWRD(M)
      YLASER(I) = LDATWRD(M+1)*(-1.0)
  100 M=M+ISKIP2
      IF (WRIDAT2 .EQ. 3HYES) CALL DATWRI2
      RETURN
      END
```

```
      SUBROUTINE DATWRI2
      COMMON/BLOCK2/LENARR2,SYNC(1500),YLASER(1500),NRECOR2,NFILE2,
     .                ZEROTM2,WRIDAT2,NTAPE2
      COMMON/UNPK2/ITIME2,LCOMWRD(601),LDATWRD(3000)
      WRITE (6,1) NRECOR2,ITIME2
    1 FORMAT (1H1,* RECORD NUMBER*I4* ITIME2=*I6)
      WRITE (6,2) (SYNC(I),I=1,LENARR2)
    2 FORMAT (1H0,/,(1X,10(F10.5,1X)))
      WRITE (6,2) (YLASER(I),I=1,LENARR2)
      RETURN
      END
```

```
      SUBROUTINE SKPEOF1
      COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
     .             ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
      COMMON/UNPK1/ITIME1,ICOMWRD(201),IDATWRD(1000)
      COMMON/BSKPEO1/LPACDA1,         IDENT1,NFLSKP1,NRCSKP1
      NREC = 0
      IF (NFLSKP1 .LE. 0) GO TO 500
      NFILSKP = 1
  100 BUFFER IN(1,1)(ITIME1,ICOMWRD(LPACDA1))
      IF (UNIT(1)) 300,400,200
  200 LEN = LENGTH(1)
      NREC = NREC + 1
      NRECOR1 = NRECOR1 + 1
      WRITE(6,2) NRECOR1,NFILE1,NFLSKP1,NREC,LEN
    2 FORMAT(1H0,* PARITY ERROR IN RECORD*I4* FILE*I2* ON UNIT 1.  ENCOU
     .NTERED WHILE SKIPPING FILE*I3/     5X* NUMBER RECORDS SKIPPED*I3*.
     .  LENGTH OF RECORD*I4* COMPUTER WORDS*)
      GO TO 100
  300 LEN = LENGTH (1)
      NREC = NREC + 1
      NRECOR 1 = NRECOR1 + 1
      IF (LEN .NE. LPACDA1)      WRITE(6,3)LPACDA1,LEN,NRECOR1,NFILE1,
     .                                         NREC
    3 FORMAT(1H0,* A RECORD WAS ENCOUNTERED WITH LENGTH NOT EQUAL TO*I4*
     . COMPUTER WORDS.  LENGTH WAS*I4*.*/5X* RECORD*I4* FILE*I2* ON UNIT
     .1.  NUMBER OF RECORDS SKIPPED*I4)
      GO TO 100
  400 WRITE(6,4)NRECOR1,NFILE1,NREC,NFILSKP,NFLSKP1
    4 FORMAT(1H0,5X*THERE WERE*I5* RECORDS ON FILE*I2* UNIT 1.*/5XI3* RE
     .CORDS SKIPPED ON THIS FILE.  TOTAL NUMBER OF FILES SKIPPED*I2* TOT
     .AL NUMBER TO BE SKIPPED*I2)
      NFILE 1= NFILE1 + 1
      NFILSKP = NFILSKP + 1
      NREC = 0
      NRECOR1= 0
      IF (IDENT1 .EQ. 3HYES) CALL HEADER1
      IF (NFILSKP .LE. NFLSKP1) GO TO 100
      IF (NRCSKP1 .GT. 0) GO TO 500
      RETURN
  500 DO 900 I=1,NRCSKP1
      BUFFER IN(1,1)(ITIME1,ICOMWRD(LPACDA1))
      IF (UNIT(1))800,700,600
  600 LEN = LENGTH (1)
      NREC = NREC + 1
      NRECOR1 = NRECOR1 + 1
      WRITE(6,5) NRECOR1,NFILE1,NREC,NRCSKP1,LEN
    5 FORMAT(1H0* PARITY ERROR IN RECORD*I4* FILE*I2* ON UNIT 1.*/5X* NU
     .MBER RECORDS SKIPPED*I4* NUMBER RECORDS TO BE SKIPPED*I3*.  LENGTH
     . OF RECORD WAS*I4* COMPUTER WORDS.*)
      GO TO 900
  700 WRITE(6,6) NRCSKP1,NREC,NRECOR1,NFILE1
    6 FORMAT(1H0,* EOF READ WHILE TRYING TO SKIP*I3* RECORDS.*I4* RECORDS
     .S HAVE BEEN SKIPPED.  RECORD NUMBER*I4* FILE*I2* ON UNIT 1*)
      GO TO 900
  800 NREC = NREC + 1
```

```
      NRECOR 1= NRECOR1 + 1
      LEN = LENGTH(1)
      IF (LEN .NE. LPACDA1)     WRITE(6,3)LPACDA1,LEN,NRECOR1,NFILE1,
     .                          NREC
  900 CONTINUE
      WRITE(6,7) NREC,NRCSKP1,NRECOR1,NFILE1
    7 FORMAT(1H0,* COMPLETED SKIPPING*I4* RECORDS.  NUMBER OF RECORDS TO
     . HAVE BEEN SKIPPED*I4/5X* RECORD NUMBER*I5* FILE*I2* ON UNIT 1*)
      RETURN
      END
```

```
      SUBROUTINE SKPEOF2
      COMMON/BSKPEO2/LPACDA2,        NTOTFI2,IDENT2,NFLSKP2,NRCSKP2,
     .                NTOTAPE,IEXIT2,NTOTREC,TIMADJ
      COMMON/BBUFLA2/JCLOCK2,IEXTIME,NREC2,NREC3,NREC4,TIMADJ1,TIMADJ2,
     .                TIMADJ3,TIMADJ4
      COMMON/BLOCK2/LENARR2,SYNC(1500),YLASER(1500),NRECOR2,NFILE2,
     .               ZEROTM2,WRIDAT2,NTAPE2
      COMMON/UNPK2/ITIME2,LCOMWRD(601),LDATWRD(3000)
      NREC = 0
      IF (NFLSKP2 .LE. 0) GO TO 600
      NFILSKP = 1
  100 BUFFER IN(2,1) (ITIME2,LCOMWRD(LPACDA2))
      IF (UNIT(2)) 300,400,200
  200 NREC = NREC + 1
      LEN = LENGTH(2)
      NRECOR2 = NRECOR2 + 1
      WRITE(6,2) NFILE2,NTAPE2,NRECOR2,NREC,LEN
    2 FORMAT(1H0,* PARITY ERROR OCCURRED WHILE SKIPPING RECORDS ON FILE
     .NUMBER*I2* OF TAPE*I2* UNIT 2.*/5X* THE RECORD NUMBER IS*I5,2XI3*
     .RECORDS HAVE BEEN SKIPPED.  THE RECORD LENGTH WAS*I4* COMPUTER WOR
     .DS*)
      GO TO 100
  300 NREC = NREC + 1
      NRECOR2 = NRECOR2 + 1
      LEN = LENGTH(2)
      IF (LEN .NE. LPACDA2) WRITE(6,3)LPACDA2,LEN,NRECOR2,NFILE2,NTAPE2,
     .                                 NREC
    3 FORMAT(1H0,* LENGTH OF A RECORD WAS NOT EQUAL TO*I4* COMPUTER WORD
     .S.  IT CONTAINED*I4* COMPUTER WORDS.*/5X* THIS OCCURRED WHEN RECOR
     .D*I5* WAS SKIPPED ON FILE*       I2* TAPE*I2* UNIT 2.  TOTAL NUMBER
     . OF RECORDS SKIPPED*I3)
      GO TO 100
  400 WRITE(6,4) NRECOR2,NFILE2,NTAPE2,NREC,NFILSKP,NFLSKP2
    4 FORMAT(1H0,5X*THERE WERE*I5* RECORDS ON FILE*I2* TAPE*I2* UNIT 2.*/
     ./5XI3* RECORDS SKIPPED ON THIS FILE .  TOTAL NUMBER OF FILES SKIPP
     .ED*I2* TOTAL NUMBER TO BE SKIPPED*I2)
      NFILE2 = NFILE2 + 1
      NFILSKP = NFILSKP + 1
      IF (NFILE2 .LE. NTOTFI2) GO TO 500
      NTAPE2 = NTAPE2 + 1
      IF (NTAPE2 .LE. NTOTAPE) GO TO 450
      IEXIT2 = 3HYES
      RETURN
  450 NTOTFI2 = 1
      CALL UNLOADW(2)
      NFILE 2= 1
      NRECOR 2= 0
      NREC = 0
  475 GO TO (480,485,490),NTAPE2
  480 NTOTREC = NREC2
      TIMADJ = TIMADJ2
      GO TO 495
  485 NTOTREC = NREC3
      TIMADJ = TIMADJ3
      GO TO 495
```

```
      490 NTOTREC = NREC4
          TIMADJ = TIMADJ4
     495  IF (IDENT2 .EQ. 3HYES) CALL HEADER2
          IF (NFILSKP .LE. NFLSKP2) GO TO 100
          IF (NRCSKP2 .GT. 0) GO TO 600
          RETURN
      500 NRECOR2 = 0
          NREC = 0
          IF(NFILE2 .EQ. NTOTFI2)GO TO 475
          IF (IDENT2 .EQ. 3HYES) CALL HEADER2
          IF (NFILSKP .LE. NFLSKP2) GO TO 100
          IF (NRCSKP2 .GT. 0) GO TO 600
          RETURN
      600 DO 1000 I=1,NRCSKP2
          BUFFER IN(2,1) (ITIME2,LCOMWRD(LPACDA2))
          IF (UNIT(2)) 900,800,700
      700 LEN = LENGTH(2)
          NREC = NREC + 1
          NRECOR2 = NRECOR2 + 1
          WRITE(6,5) NRECOR2,NFILE2,NTAPE2,NREC,LEN
        5 FORMAT(1H0,* PARITY ERROR OCCURRED WHILE SKIPPING RECORDS.  RECORD
         . NUMBER*I5* FILE*I2* TAPE*I2* ON UNIT 2.*/5XI4* RECORDS HAVE BEEN
         .SKIPPED.  LENGTH OF RECORD WAS*I4* COMPUTER WORDS.*)
          GO TO 1000
      800 WRITE(6,6) NREC,NRECOR2,NFILE2,NTAPE2
      6   FORMAT(1H0* AN EOF WAS ENCOUNTERED WHILE SKIPPING RECORDS.* I5* RE
         .CORDS HAVE BEEN SKIPPED.*/5X* RECORD NUMBER*I5* OF FILE*I2* ON TAP
         .E*I2* OF UNIT 2.*)
          GO TO 1000
      900 NREC = NREC + 1
          NRECOR2 = NRECOR2 + 1
          LEN = LENGTH(2)
          IF (LEN .NE. LPACDA2)    WRITE(6,3)LPACDA2,LEN,NREC,NFILE2,NTAPE2,
         .                              NRECOR2
     1000 CONTINUE
          WRITE(6,7) NREC,NRCSKP2,NRECOR2,NFILE2,NTAPE2
        7 FORMAT(1H0,* COMPLETED SKIPPING*I4* RECORDS.  NUMBER OF RECORDS TO
         . HAVE BEEN SKIPPED*I4*   RECORD NUMBER*I5* FILE*I2* TAPE*I2* ON UN
         .IT 2*)
          RETURN
          END
```

```
      SUBROUTINE CONSTMR
      COMMON/BCONSTM/NAVEMIR,DIRECMR,CHGMIR,TIMEMIR
      COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
     .               ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
      COMMON/BCALIBR/SLOPE(2),ZEROTAP(2),SLOPEAN,ANINTER,SLOPEHW,
     .               SLOPEWD,WDINTER,SLOPEMD,DMINTER
      AVEMIR = 0
      NRECOR1 = 0
      CALL BUFLAS1
  100 CALL BUFLAS1
      DO 200 K=1,LENARR1
  200 AVEMIR = AVEMIR + DIRMIPR(K)
      IF (NRECOR1 .LE. NAVEMIR) GO TO 100
      DIRECMR = AVEMIR/(LENARR1*NAVEMIR)
      LAST= NAVEMIR + 1
      DO 300 I=1,LAST
      BACKSPACE 1
  300 CONTINUE
      READ(5,1) CHGMIR,TIMEMIR
    1 FORMAT (A3,F6.2)
      WRITE(6,2)NRECOR1,DIRECMR,SLOPEMD,DMINTER
    2 FORMAT(1H0,5X*MIRROR DIRECTION*5X*NUMBER OF RECORDS USED FOR AVERA
     .GE*I3/ 10X*AVERAGE VOLTAGE*5X*SLOPE*5X*INTERCEPT*5X*DIRECTION,DEGR
     .EES*/14XF7.3,9XF7.3,5XF5.3)
      DIRECMR = SLOPEMD*DIRECMR + DMINTER + 180
      WRITE(6,3)DIRECMR
    3 FORMAT(1H+,59XF7.3)
      NRECOR1 = 0
      RETURN
      END
```

APPENDIX A-2

Computer Program for Determination of Velocity Profiles

```
      PROGRAM ANEVEL(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,FILMPL)
      COMMON TIME(101),VELOC(6,100),NCHANNC,LENARR,NFILE,NRECOR,IEXIT
      COMMON /BCAIBR/ NCALVAL, ACTVOLT(5),SLOPE(6),ZEROTAP(6),
     .                STANDEV(6,5),VARI
      COMMON/BBUFANE/IDENT,IPARITY,LPACDAT,EOFMUL,NTOTFIL,PLOT
      COMMON/BSORT/IBEGSKP,WRITDAT,ISKIP,FACTOR
      COMMON/BVOLTAD/ICHANGE,SCALE(6),ZEROACT(6),VOLTCHG,TIMECHG,
     .               ISCALE(6)
      COMMON/BPLOTVE/ELEV(6),SUMAVE(6),ISAMPLE,LABELX(4),LABELY(4),
     .                LTITLE(4),MULTIME,IDATAPT
      COMMON/BSKIPEO/NFILSKP,NRECSKP
      COMMON/UNPK/ITIME,ICOMWRD(200),IDATWRD(1000)
      DATA LABELX/40H             VELOCITY, M/SEC             /
     .    ,LABELY/40H              ELEVATION, M               /
     .    ,LTITLE/40H            VELOCITY PROFILE             /
      REWIND 1
      READ(5,1)       IDENT,WRITAP,EOFMUL,WRITDAT,WRITPAP,PLOT,VOLTCHG,
     .          LPACDAT,NCHANNC,NCALVAL,ISKIP,IBEGSKP,LENARR,NTOTFIL,
     .          PLOTIME,AVETIME,(ISCALE(I),I=1,6),(ELEV(I),I=1,6),VARI,
     .          DIGRAT,(ACTVOLT(I),I=1,5),TIMERAT,CHANNEL,TIMECHG,
     .          INSTCAL,TAPECAL
    1 FORMAT (3X,7A3,7I4,2F5.3,6(I2),/,6(F6.2),2F5.2,5(F5.1),3F3.1/2A3)
      WRITE(6,3)      IDENT,WRITAP,EOFMUL,WRITDAT,WRITPAP,PLOT,VOLTCHG,
     .          LPACDAT,NCHANNC,NCALVAL,ISKIP,IBEGSKP,LENARR,NTOTFIL,
     .          PLOTIME,AVETIME,(ISCALE(I),I=1,6),(ELEV(I),I=1,6),VARI,
     .          DIGRAT,(ACTVOLT(I),I=1,5),TIMERAT,CHANNEL,TIMECHG,
     .          INSTCAL,TAPECAL
    3 FORMAT(1H0,* IDENT =*A4* WRITAP =*A4* EOFMUL =*A4* WRITDAT =*A4* W
     .RITPAP =*A4* PLOT =*A4* VOLTCHG =*A4/* LPACDAT =*I4* NCHANNC =*I2*
     . NCALVAL =*I2* ISKIP =*I3* IBEGSKP =*I3* LENARR =*I4* NTOTFIL =*I2
     ./* PLOTIME =*F5.1* AVETIME =*F5.1* ISCALE(1 THRU 6) =*6I2/* ELEV(1
     . THRU 6) =*6F6.2* VARI =*F4.2* DIGRAT =*F5.1/* ACTVOLT(1 THRU 5) =
     .*5F5.1* TIMRAT =*F4.1* CHANNEL =*F4.1* TIMECHG =*F3.1/* INSTCAL =*
     .A4* TAPECAL =*A4)
      IF (IDENT.EQ. 3HYES)CALL HEADER
      IEXIT = 3H NO
      CORTIME = 3H NO
      JCLOCK = 0
      ZEROTIM = 0.0
      ICHANGE = 0
      FACTOR = SQRT(2.)/(2.**9 - 1.0)
      MULTIME = 1
      NRECOR = 0
      BADDATA = 3H NO
      NEXTPTS = 0
      NFILE = 1
      IPARITY = 0
      ISAMPLE = 0
      DO 100 I=1,NCHANNC
  100 SUMAVE(I) = 0.0
      IF (TAPECAL .EQ. 3HYES) CALL CALIBRA
      IF(INSTCAL .NE. 3HYES)GOTO 102
      NFILSKP = 1
      NRECSKP = 0
      CALL SKIPEOF
```

```
      102 CALL VOLTADJ
      103 CALL BUFANE
          IF (IEXIT .EQ. 3HYES) TIME(IDATAPT-1) = TIME(IDATAPT-1) + 0.1
          IF (IEXIT .EQ. 3HYES) CALL PLOTVEL
          IF (NRECOR .EQ.1 .AND. ITIME .NE. 0) ZEROTIM = ITIME
          IF ((ITIME - 999999) .GT. -12000) CORTIME = 3HYES
          ITIME = ITIME + JCLOCK * 999999
          IF (CORTIME .EQ. 3HYES) JCLOCK = JCLOCK + 1
          CORTIME = 3H NO
          IDATAPT = 1
      105 DO 110 I=1,NCHANNC
          VELOC(I,IDATAPT)=SLOPE(I)*VELOC(I,IDATAPT) + ZEROTAP(I)
      110 VELOC(I,IDATAPT) = (VELOC(I,IDATAPT)*SCALE(I) +ZEROACT(I))*0.3048
          TIME(IDATAPT)= TIMERAT*((ITIME -ZEROTIM)/10000.+((IDATAPT-1)*
         .              CHANNEL)/DIGRAT)
          IF (TIME(IDATAPT) .GE. TIMECHG .AND. VOLTCHG .EQ. 3HYES)
         .     CALL VOLTADJ
          IDATAPT = IDATAPT + 1
          IF (AVETIME*MULTIME .LE. TIME(IDATAPT-1) .AND. PLOT
         .     .EQ. 3HYES) GO TO 120
          IF ( IDATAPT .LE. LENARR) GO TO 105
          IF (WRITPAP .EQ. 3HYES) GO TO 134
          IF (WRITAP .EQ. 3HYES) GO TO 135
          GO TO 103
      120 DO 130 I= 1,NCHANNC
      130 SUMAVE(I) = SUMAVE(I) + VELOC(I,IDATAPT-1)
          ISAMPLE = ISAMPLE + 1
          IF ( TIME(IDATAPT-1) .GE. PLOTIME*MULTIME) CALL PLOTVEL
          IF (IDATAPT .LE. LENARR) GO TO 105
          IF (WRITPAP .EQ. 3HYES) GO TO 134
          IF (WRITAP .EQ. 3HYES) GO TO 135
          GO TO 103
      134 WRITE(6,2)
      2   FORMAT(1H1,4X* TIME,SECS*10X* VELOCITIES,M/SEC*4X* LEVEL 1*4X
         .       * LEVEL 2*4X* LEVEL 3*4X* LEVEL 4*4X* LEVEL 5*4X* LEVEL 6*
         .       //)
          WRITE (6,4) (TIME(J),(VELOC(I,J),I=1,6),J=1,LENARR)
      4   FORMAT(1H ,100(4X,F10.3,28X,6(F6.3,6X)/))
          GO TO 103
      135 CONTINUE
      140 CONTINUE
          END
```

```
          SUBROUTINE CALIBRA
          COMMON TIME(101),VELOC(6,100),NCHANNC,LENARR,NFILE,NRECOR,IEXIT
          COMMON /BCAIBR/ NCALVAL, ACTVOLT(5),SLOPE(6),ZEROTAP(6),
         .               STANDEV(6,5),VARI
          DIMENSION SUMCAL(6),SUMTAP(6),SQVALUE(6),SUMACT(6),ACT X TAP(6),
         .          SUMSQ(6,5),RECMEAN(6),TOTMEAN(6,5),TEPMEAN(6),SUMEAN(6),
         .          TEMPSUM(6)
          ICHECK = 0
          NSAMPLE = 0
          LASTCAL = 0
          ICALVAL = 1
          DO 100 I=1, NCHANNC
          SUMEAN(I) = 0.0
          TEMPSUM(I) = 0.0
          SUMCAL(I) = 0.0
          SUMACT(I) = 0.0
          SUMTAP(I) = 0.0
          SQVALUE(I) = 0.0
          ACT X TAP(I) = 0.0
          TEPMEAN(I) = 0.0
          RECMEAN(I) = 0.0
          DO 100 J=1, NCALVAL
          TOTMEAN(I,J) = 0.0
      100 SUMSQ(I,J) = 0.0
      105 CALL BUFANE
          GOTOBUF= 3H NO
          IF (ICALVAL .EQ. NCALVAL) LASTCAL = LASTCAL + 1
          DO 110 K=1,LENARR
          DO 110 I=1, NCHANNC
      110 SUMCAL(I)= SUMCAL(I) + VELOC(I,K)
          IF (ICHECK .GT. 0) GO TO 131
          NSAMPLE = NSAMPLE + 1
          DO 125 I=1, NCHANNC
          RECMEAN(I)= SUMCAL(I)/LENARR
          IF (NRECOR .EQ. 1) GO TO 120
          IF (RECMEAN(I) .GT. TOTMEAN(I,ICALVAL) + VARI .OR. RECMEAN(I) .LT.
         .    TOTMEAN(I,ICALVAL) - VARI) GO TO 130
      120 IF (I .EQ. 1) WRITE(6,1)NRECOR,ICALVAL,ACTVOLT(ICALVAL)
        1 FORMAT(1H0,5X*RECORD MEANS*4X*RECORD NUMBER*I4,7X*CALIBRATION*I2,4
         .X*INPUT VALUE*F5.1/11X*CHANNEL*10X*MEAN*13X*CUMULATIVE MEAN*6X*NUM
         .BER RECORDS FOR CUMULATIVE MEAN*)
          DO 123 K=1,LENARR
      123 SUMSQ(I,ICALVAL) = SUMSQ(I,ICALVAL) + VELOC(I,K)**2
          SUMEAN(I) = SUMEAN(I) + RECMEAN(I)
          SUMCAL(I) = 0.0
          TOTMEAN(I,ICALVAL) = SUMEAN(I)/NSAMPLE
      125 WRITE(6,2)I,RECMEAN(I),TOTMEAN(I,ICALVAL),NSAMPLE
        2 FORMAT(1H ,12X,I2,10X,F8.4,14X,F8.4,25XI3)
          GO TO 105
      130 NSAMPLE= NSAMPLE - 1
          ICALVAL = ICALVAL + 1
      131 IF (ICALVAL .GT. NCALVAL .AND. LASTCAL .GT. 3) GO TO 160
          ICHECK = ICHECK + 1
          WRITE(6,3)NRECOR,ICALVAL,ACTVOLT(ICALVAL)
        3 FORMAT(1H0,5X*TEMPORARY MEANS*8X*RECORD NUMBER*I4,10X*CALIBRATION*
```

```
     .I2,4X*INPUT VALUE*F5.1/11X*CHANNEL*10X*MEAN*)
      DO 140 I=1, NCHANNC
      RECMEAN(I) = SUMCAL(I)/LENARR
      WRITE(6,4) I,RECMEAN(I)
4     FORMAT(1H ,12XI2,10XF8.4)
      SUMCAL(I) = 0.0
      IF (ICHECK .EQ. 1) GO TO 135
      DO 137 K=1,LENARR
137   SUMSQ(I,ICALVAL)=SUMSQ(I,ICALVAL) + VELOC(I,K)**2
      TEMPSUM(I) = TEMPSUM(I) + RECMEAN(I)
135   IF (RECMEAN(I) .GT. TOTMEAN(I,ICALVAL-1) + VARI .OR. RECMEAN(I)
     .     .LT. TOTMEAN(I,ICALVAL-1) - VARI) GOTOBUF=3HYES
140   CONTINUE
      IF (ICHECK.GT. 3) GO TO 160
      IF (GOTOBUF .EQ. 3HYES) GO TO 105
      DO 150 I=1, NCHANNC
      TEMPSUM(I) = 0.0
150   SUMSQ(I,ICALVAL)=0.0
      ICHECK = 0
      ICALVAL = ICALVAL - 1
      GO TO 105
160   IEND = ICALVAL - 1
      WRITE(6,5) IEND,ACTVOLT(IEND)
5     FORMAT(1H0,5X*STANDARD DEVIATIONS*10X*CALIBRATION*I2,5X*INPUT VALU
     .E*F5.1/11X*CHANNEL*10X*RMS*)
      DO 170 I=1, NCHANNC
      STANDEV(I,ICALVAL-1)= SQRT(SUMSQ(I,ICALVAL-1)/(NSAMPLE*LENARR) -
     .                     TOTMEAN(I,ICALVAL-1)**2)
170   WRITE(6,6) I,STANDEV(I,IEND)
6     FORMAT(1H ,12XI2,7XF9.3)
      NSAMPLE = ICHECK - 1
      DO 175 I=1,NCHANNC
      SUMEAN(I) = TEMPSUM(I)
      TOTMEAN(I,ICALVAL) = TEMPSUM(I)/NSAMPLE
175   TEMPSUM(I) = 0.0
      ICHECK = 0
      IF (ICALVAL .LE. NCALVAL) GO TO 105
180   WRITE(6,7) NRECOR
7     FORMAT(1H0,5X*ACTUAL VS TAPE VOLTAGE*10X*LEAST SQUARE METHOD*5X*NU
     .MBER RECORDS USED FOR CALCULATIONS*I3)
      DO 200 I=1, NCHANNC
      DO 190 J=1, NCALVAL
      SUMTAP(I)=SUMTAP(I) + TOTMEAN(I,J)
      SQVALUE(I)=SQVALUE(I)+TOTMEAN(I,J)**2
      ACT X TAP(I) = ACT X TAP(I) + TOTMEAN(I,J)*ACTVOLT(J)
190   SUMACT(I) = SUMACT(I) + ACTVOLT(J)
      SLOPE(I) = (SUMACT(I)*SUMTAP(I) -NCALVAL*ACT X TAP(I))/
     .           (SUMTAP(I)**2-NCALVAL* SQVALUE(I))
      ZEROTAP(I) = (SUMTAP(I)*ACT X TAP(I) - SUMACT(I)*SQVALUE(I))/
     .           (SUMTAP(I)**2- NCALVAL*SQVALUE(I))
      WRITE(6,8) I,(ACTVOLT(J),TOTMEAN(I,J),J=1 ,NCALVAL)
8     FORMAT (1H0,10X*C H A N N E L*I3/15X*VALUES USED FOR LEAST SQUARE
     .CALCULATIONS*10X*INPUT VALUE*5X*TAPE VALUE*/(69XF4.1,11XF6.3))
200   WRITE(6,9) SLOPE(I),ZEROTAP(I)
9     FORMAT(1H ,15X*VALUES OBTAINED FROM LEAST SQUARE CALCULATIONS*7X*S
```

```
     .LOPE*8X*INTERCEPT*/68XF5.3,11XF5.3)
      RETURN
      END
```

```
      SUBROUTINE BUFANE
      COMMON TIME(101),VELOC(6,100),NCHANNC,LENARR,NFILE,NRECOR,IEXIT
      COMMON/BRUFANE/IDENT,IPARITY,LPACDAT,EOFMUL,NTOTFIL,PLOT
      COMMON/UNPK/ITIME,ICOMWRD(200),IDATWRD(1000)
  100 DO 105 I=1,LPACDAT
  105 ICOMWRD(I) = 0
      LUNPDAT = LPACDAT * 5
      DO 110 I = 1,LUNPDAT
  110 IDATWRD(I) = 0
  115 NRECOR = NRECOR + 1
  120 BUFFER IN (1,1)(ITIME,ICOMWRD(LPACDAT))
  125 IF (UNIT(1)) 140,130,135
  130 NRECOR = NRECOR - 1
      WRITE (6,1) NRECOR, NFILE
      NRECOR = 0
      NFILE = NFILE +1
      IF (NFILE .GT. NTOTFIL) GO TO 136
      IF (IDENT .EQ. 3HYES)CALL HEADER
      GO TO 100
  135 IPARITY = IPARITY + 1
      WRITE (6,2) NRECOR, NFILE
      NRECOR = NRECOR - 1
      WRITE (6,3) IPARITY
      GO TO 115
  136 IF (PLOT .EQ. 3HYES) IEXIT = 3HYES
      IF (PLOT .EQ. 3HYES) GO TO 150
      CALL EXIT
  140 CALL UNPAK
      CALL SORTANE
    1 FORMAT (1H0,* THERE ARE *I4* RECORDS ON FILE NUMBER*I3)
    2 FORMAT (1H0,* PARITY ERROR OCCURRED ON RECORD NUMBER*I4* FILE NUMB
     .                 ER*I3)
    3 FORMAT (1H0,* THERE HAVE BEEN*I3* PARITY ERRORS*)
  150 RETURN
      END
```

```
      SUBROUTINE HEADER
      COMMON TIME(101),VELOC(6,100),NCHANNC,LENARR,NFILE,NRECOR,IEXIT
      COMMON/BHEADER/ID(2)
      BUFFER IN(1,0)(ID(1),ID(2))
      IF (UNIT (1)) 100,110,110
  110 WRITE(6,2) NFILE
    2 FORMAT(1H0,* PARITY ERROR OR EOF OCCURRED IN HEADER OF FILE NO*
     .          I3)
  100 WRITE(6,1) ID,NFILE
    1 FORMAT(1H1,* HEADER IN BINARY *2A10* ON FILE NUMBER *I2)
  120 RETURN
      END
```

```
      SUBROUTINE SORTANE
      COMMON TIME(101),VELOC(6,100),NCHANNC,LENARR,NFILE,NRECOR,IEXIT
      COMMON/RSORT/IBEGSKP,WRITDAT,ISKIP,FACTOR
      COMMON/UNPK/ITIME,ICOMWRD(200),IDATWRD(1000)
      M= IBEGSKP
      DO 100 I=1,LENARR
      VELOC(1,I) = IDATWRD(M) * FACTOR
      VELOC(2,I) = IDATWRD(M+1) * FACTOR
      VELOC(3,I) = IDATWRD(M+2)*FACTOR
      VELOC(4,I) = IDATWRD(M+3) * FACTOR
      VELOC(5,I) = IDATWRD(M+4) * FACTOR
      VELOC(6,I) = IDATWRD(M+5) * FACTOR
  100 M = M + ISKIP
      IF ( WRITDAT .EQ. 3HYES) CALL DATAWRI
      RETURN
      END
```

```
      SUBROUTINE DATAWRI
      COMMON TIME(101),VELOC(6,100),NCHANNC,LENARR,NFILE,NRECOR,IEXIT
      COMMON/UNPK/ITIME,ICOMWRD(200),IDATWRD(1000)
      DO 10 I=1,NCHANNC
   10 WRITE (6,1) I,(VELOC(I,J),J=1,LENARR)
    1 FORMAT(1H0,10X,* ANEMOMETER VELOCITY DATA, LEVEL NUMBER*I2/101
     .       (10F11.5/))
      PRINT 2,ITIME
    2 FORMAT (1H0,10X,*ITIME AT BEGINNING OF RECORD*I10)
      RETURN
      END
```

```
      SUBROUTINE SKIPEOF
      COMMON TIME(101),VELOC(6,100),NCHANNC,LENARR,NFILE,NRECOR,IEXIT
      COMMON/BBUFANE/IDENT,IPARITY,LPACDAT,EOFMUL,NTOTFIL,PLOT
      COMMON/UNPK/ITIME,ICOMWRD(200),IDATWRD(1000)
      COMMON/BSKIPEO/NFILSKP,NRECSKP
      NREC = 0
      IF (NFILE .GT. NFILSKP) GO TO 125
  100 BUFFER IN(1,1)(ITIME,ICOMWRD(LPACDAT))
      IF (UNIT(1))115,120,110
  110 NREC = NREC + 1
      LEN = LENGTH(1)
      NRECOR = NRECOR + 1
      PRINT 2,NFILE,NRECOR,NREC,LEN
    2 FORMAT(1H0,5X*PARITY ERROR OCCURRED WHILE SKIPPING FILE*I2* RECORD
     .*I4/7XI3* RECORDS HAVE BEEN SKIPPED.  LENGTH WAS*I4)
      GO TO 100
  115 LEN = LENGTH(1)
      NREC = NREC + 1
      NRECOR = NRECOR + 1
      IF (LEN .NE. LPACDAT + 1) WRITE(6,3) LEN,NRECOR,NFILE,NREC
    3 FORMAT(1H0,5X*RECORD SKIPPED OF IMPROPER LENGTH. LENGTH WAS*I4/7X
     .*RECORD*I4* FILE*I2,2XI2* RECORDS SKIPPED*)
      GO TO 100
  120 WRITE(6,4) NREC,NFILE,NRECOR
    4 FORMAT(1H0,5X*SKIPPED*I3* RECORDS ON FILE*I2* THERE WERE*I4* RECOR
     .DS ON THIS FILE*)
      NFILE = NFILE + 1
      IF (IDENT .EQ. 3HYES)  CALL HEADER
      NREC = 0
      NRECOR = 0
      IF (NFILE .LE. NFILSKP)GO TO 100
  125 IF (NRECSKP .EQ. 0) RETURN
      DO 160 I=1,NRECSKP
      BUFFER IN (1,1) (ITIME,ICOMWRD(LPACDAT))
      IF (UNIT(1))130,150,140
  130 NRECOR = NRECOR + 1
      NREC = NREC + 1
      LEN = LENGTH(1)
      IF (LEN .NE. LPACDAT + 1) WRITE(6,3) LEN,NRECOR,NFILE,NREC
      GO TO 160
  140 NREC = NREC + 1
      NRECOR = NRECOR + 1
      LEN = LENGTH (1)
      WRITE(6,5) NRECSKP,NFILE,NRECOR,LEN,NREC
    5 FORMAT(1H0,5X*PARITY ERROR OCCURRED WHILE SKIPPING*I2* RECORDS ON
     .FILE*I2/7X*RECORD*I4* LENGTH*I4* RECORDS SKIPPED*I3)
      GO TO 160
  150 WRITE(6,6)NFILE,NRECOR,NREC
    6 FORMAT(1H0,5X*EOF OCCURRED WHILE SKIPPING RECORDS ON FILE*I2/7X*LA
     .ST RECORD*I4,2XI3* RECORDS HAVE BEEN SKIPPED*)
  160 CONTINUE
      WRITE(6,7) NREC,NFILE
    7 FORMAT(1H0,5X,I3* RECORDS HAVE BEEN SKIPPED ON FILE*I2)
      RETURN
      END
```

```
      SUBROUTINE VOLTADJ
      COMMON TIME(101),VELOC(6,100),NCHANNC,LENARR,NFILE,NRECOR,IEXIT
      COMMON/BVOLTAD/ICHANGE,SCALE(6),ZEROACT(6),VOLTCHG,TIMECHG,
     .               ISCALE(6)
      IF (ICHANGE .GT. 0) READ (5,1)(ISCALE(I),I=1,6),VOLTCHG,TIMECHG
      DO 90 I= 1,6
   20 GO TO (30,40,50,60,70,80),I
   30 GO TO (31,32,33),ISCALE(I)
   31 SCALE(I) = 40.166
      ZEROACT(I) = 2.799
      GO TO 90
   32 SCALE(I) = 78.867
      ZEROACT(I) = 2.413
      GO TO 90
   33 SCALE(I) = 0.0
      ZEROACT(I) = 0.0
      GO TO 90
   40 GO TO (41,42,43),ISCALE(I)
   41 SCALE(I) = 42.161
      ZEROACT(I) = 2.183
      GO TO 90
   42 SCALE(I) = 81.437
      ZEROACT(I) = 2.281
      GO TO 90
   43 SCALE(I) = 0.0
      ZEROACT(I) = 0.0
      GO TO 90
   50 GO TO (51,52,53),ISCALE(I)
   51 SCALE(I) = 42.981
      ZEROACT(I) = 2.057
      GO TO 90
   52 SCALE(I) = 83.606
      ZEROACT(I) = 1.883
      GO TO 90
   53 SCALE(I) = 0.0
      ZEROACT(I) = 0.0
      GO TO 90
   60 GO TO (61,62,63),ISCALE(I)
   61 SCALE(I) = 42.869
      ZEROACT(I) = 3.674
      GO TO 90
   62 SCALE(I) = 83.224
      ZEROACT(I)  =3.065
      GO TO 90
   63 SCALE(I) = 0.0
      ZEROACT(I) = 0.0
      GO TO 90
   70 GO TO (71,72,73),ISCALE(I)
   71 SCALE(I) = 47.070
      ZEROACT(I) = 0.075
      GO TO 90
   72 SCALE(I) = 93.300
      ZEROACT(I) = 0.330
      GO TO 90
   73 SCALE(I) = 0.0
```

```
      ZEROACT(I) = 0.0
      GO TO 90
   80 GO TO (81,82,83),ISCALE(I)
   81 SCALE(I) = 40.217
      ZEROACT(I) = 3.764
      GO TO 90
   82 SCALE(I) = 77.313
      ZEROACT(I) = 3.639
      GO TO 90
   83 SCALE(I) = 0.0
      ZEROACT(I) = 0.0
   90 CONTINUE
      ICHANGE = ICHANGE + 1
    1 FORMAT (6(I2),A3,F5.3)
      WRITE(6,2)
    2 FORMAT(1H0,5X*ACTUAL VOLTAGE VS VELOCITY*5X*REGRESSION VALUES*/10X
     .*LEVEL*5X*SLOPE*5X*INTERCEPT*)
      DO 100 I=1,NCHANNC
  100 WRITE(6,3) I,SCALE(I),ZEROACT(I)
    3 FORMAT(1H ,11XI1,7XF6.3,6XF5.3)
      RETURN
      END
```

```
      SUBROUTINE PLOTVEL
      COMMON TIME(101),VELOC(6,100),NCHANNC,LENARR,NFILE,NRECOR,IEXIT
      COMMON/BPLOTVE/ELEV(6),SUMAVE(6),ISAMPLE,LABELX(4),LABELY(4),
     .                 LTITLE(4),MULTIME,IDATAPT
      DIMENSION AVEVEL(6)
      MULTIME = MULTIME + 1
      WRITE(6,1) TIME(IDATAPT-1),NRECOR
    1 FORMAT(1H0,5X*VELOCITY PROFILE PLOTTED AT TIME*F9.3,5X*RECORD NUMB
     .ER*I4/10X*VALUES USED FOR PLOT*5X*LEVEL*5X*ELEVATION, M.*5X*VELOCI
     .TY, M/SEC*)
      DO 100 I = 1,NCHANNC
  100 AVEVEL(I) = 0.0
      DO 110 I=1,NCHANNC
      AVEVEL(I) = SUMAVE(I)/ISAMPLE
      WRITE(6,2) I,ELEV(I),AVEVEL(I)
    2 FORMAT(1H ,36XI1,10XF6.3,13XF6.3)
  110 SUMAVE(I) = 0.0
      CALL IDIOT (AVEVEL,ELEV,6,2,DUM,LABELX,LABELY,LTITLE,+1)
      ISAMPLE = 0
      IF (IEXIT .EQ. 3HYES) CALL EXIT
      RETURN
      END
```

```
                                             IDENT UNPAK
                                   *            INSERT LENGTHS OF PACKED AND UNPACKED ARRAYS
                        311        LENGTHA   SET   201
                       1750        LENGTHB   SET   1000
                                   *
                                             USE   /UNPK/
     0                             NE        BSS   LENGTHA
   311                             B         BSS   LENGTHB
                                             USE
                                             ENTRY UNPAK
     0                             UNPAK     BSS   1
     1 7170000001                            SX7   1B
                 7100004000                  SX0   4000B
     2 43214                                 MX2   12
            5110000000 C                     SA1   NE        A(1) --- FIRST WORD OF ARRAY (TIME WORD) IS IGNORED
     3 6160000310 C                          SB6   B-1       B(J) BASE
                 6170002260 C                SB7   B+LENGTHB-1     B(LAST)
     4 6150000060                            SB5   48
                 6110000074                  SB1   60
     5 5011000001                  GET60     SA1   A1+1      GET A(I)
     6 6166000001                  GET12     SB6   B6+1
                 11621                       BX6   X2*X1     MASK OUT 12 BITS
                      67515                  SB5   B1-B5           RIGHT SHIFT
     7 22656                                 LX6   B5,X6            BUT
            67515                            SB5   B1-B5           AVOID SIGN EXTENSION
                                   *
                 11760                       BX7   X6*X0           CK FOR SIGN BIT
    10 0307000011 +                          ZR    X7,STORB
                 15660                       BX6   -X0*X6          MASK OUT SIGN BIT
                      14666                  BX6   -X6
                                   *
    11                             STORB     BSS   0
    11 21601                                 AX6   1         DELETE ZERO-BIT RIGHT-FILL
            56660                            SA6   B6        STORE IN B(J)
                 0467000000 +                EQ    B6,B7,DONE
    12 0550000014 +                          NE    B5,B0,INMID
                 6150000060                  SB5   48
    13 43214                                 MX2   12
            0400000005 +                     EQ    GET60
    14 20260                       INMID     LX2   48
            6155777763                       SB5   B5-12
    15 0400000006 +                          EQ    GET12
                          0 +      DONE      EQU   UNPAK
    16                                       END

                      46302        STORAGE USED              44 STATEMENTS       10 SYMBOLS
                                   6400 ASSEMBLY          0.341 SECONDS          23 REFERENCES
```

APPENDIX A-3

Computer Program for Determination of Temperature and Humidity Profiles

```
      PROGRAM TEMPHUM(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,
     .                FILMPL)
CS    DEBUG
CS    ARRAYS
      COMMON NFILE,NRECOR,LENARR,NCHANN,TEMP(10,100),WRITDAT
      COMMON/BSORT/IBEGSKP,FACTOR,ISKIP
      COMMON/BINSTCA/NINSCAL,VARIIN,EXCITVO,RESIS(10,2),CALRES(10,2),
     .               GAIN(10)
      COMMON/BCALIBR/SLOPE(10),ZEROTAP(10)
      COMMON/BBUFTEM/IDENT,MULEOF,IEXIT,NTOTFIL,PLOT,IPARITY,ZEROTIM,
     .               JCLOCK
      COMMON/UNPK/ITIME,ICOMWRD(200),IDATWRD(1000)
      COMMON/BSKPEOF/BADATA,LPACDAT,NRECSKP,NFILSKP
      COMMON/BHUMID/SIGMA,BARPRES,HEATLAT,CP,CPV,HUMIDI(10)
      COMMON/BTAPECA/NCALVAL,VARITP,ACTVOLT(5)
      COMMON/BPLOTEM/AVETEMP(10),TIME,ELEV(6),LABELX(4),LABELY(4),LTITLE
     .               (4)
      DIMENSION RESISR(10),SUMTEMP(10)
      DATA LABELX/40H                 TEMPERATURE,C               /
     .    ,LABELY/40H                  ELEVATION,M                /
     .    ,LTITLE/40H              TEMPERATURE PROFILE            /
      READ(5,1) CALTAPE,CALINST,WRITDAT,IDENT,MULEOF,PLOT,NCHANN,LENARR,
     .          NAVEREC,IBEGSKP,ISKIP,NINSCAL,LPACDAT,NTOTFIL,EXCITVO,
     .          RESISR(1),A,B,C,D,E,TIMRAT,VARIIN,SIGMA,BARPRES,CP,
     .          HEATLAT,CPV,VARITP,(ACTVOLT(I),I=1,5),(RESIS(I,1),I=1,10
     .          ),(RESIS(I,2),I=1,10),(CALRES(I,1),I=1,10),(CALRES(I,2),
     .          I=1,10),(ELEV(I),I=1,6),NCALVAL
    1 FORMAT(6A3,8I4,3F8.3/4F8.3,2F5.2,5F7.3/1F5.2,5(F5.2),10(F5.2)/
     .       10(F5.2),6(F5.2)/4(F5.2),10(F5.2)/6(F6.3),I3)
      REWIND 1
      PRINT 3
    3 FORMAT(1H0,5X*NOTE...CHANNEL 1 IS LEVEL 2, AMBIENT TEMPERATURE*/12
     .X*CHANNEL 2 IS LEVEL 3, DRY*/12X*CHANNEL 3 IS LEVEL 1, DRY*/12X*CH
     .ANNEL 4 IS LEVEL 1, WET*/12X*CHANNEL 5 IS LEVEL 4, DRY*/12X*CHANNE
     .L 6 IS LEVEL 4, WET*/12X*CHANNEL 7 IS LEVEL 5, DRY*/12X*CHANNEL 8
     .IS LEVEL 5, WET*/12X*CHANNEL 9 IS LEVEL 6, DRY*/12X*CHANNEL 10 IS
     .LEVEL 6, WET*)
      PRINT 4,  CALTAPE,CALINST,WRITDAT,IDENT,MULEOF,PLOT,NCHANN,LENARR,
     .          NAVEREC,IBEGSKP,ISKIP,NINSCAL,LPACDAT,NTOTFIL,EXCITVO,
     .          RESISR(1),A,B,C,D,E,TIMRAT,VARIIN,SIGMA,BARPRES,CP,
     .          HEATLAT,CPV,VARITP,(ACTVOLT(I),I=1,5),(RESIS(I,1),I=1,10
     .          ),(RESIS(I,2),I=1,10),(CALRES(I,1),I=1,10),(CALRES(I,2),
     .          I=1,10),(ELEV(I),I=1,6),NCALVAL
    4 FORMAT(1H0,* CALTAPE =*A4* CALINST =*A4* WRITDAT =*A4* IDENT =*A4
     .* MULEOF =*A4* PLOT =*A4/* NCHANN =*I3* LENARR =*I4* NAVEREC =*I5
     .* IBEGSKP =*I2* ISKIP =*I3* NINSCAL =*I2* LPACDAT =*I4* NTOTFIL =*
     .I2/* EXCITVO =*F5.3* RESISR(1) =*F5.2* A =*F8.2* B =*F7.3* C =*F7
     ..3* D=*F5.3* F=*F6.3/* TIMRAT =*F4.1* VARIIN =*F5.2* SIGMA =*F7.5
     .* BARPRES =*F8.2* CP =*F5.3* HEATLAT =*F6.1* CPV =*F5.3* VARITP =*
     .F4.2/*ACTVOLT(1 THRU 5)=*5F5.1/* RESIS(1 THRU 10,1) =*10F5.2/* RES
     .IS(1 THRU 10,2) =*10F5.2/* CALRES(1 THRU 10,1) =*10F5.2/* CALRES (1
     .1 THRU 10,2) =*10F5.2/* ELEV(1 THRU 6) =*6F7.3* NCALVAL =*I2)
      IEXIT = 3H NO
      JCLOCK = 0
      BADATA = 3H NO
```

```
      BADATA = 3H NO
```



```
      FACTOR = SQRT(2.0) /(2.0**9-1.0)
      NFILE = 1
      NRECOR = 0
      ZEROTIM = 0.0
      IPARITY = 0
      MULREC = 1
      DO 100 I=1,NCHANN
  100 SUMTEMP(I) = 0.0
      IF (CALTAPE .EQ. 3HYES)CALL TAPECAL
      IF (CALINST .EQ. 3HYES) CALL INSTCAL
      NFILSKP = 1
      NRECSKP = 0
      IF (NFILE .LE. 1) CALL SKIPEOF
      JCLOCK = 0
      ZEROTIM = 0.0
      NRECOR = 0
      READ(5,5)FRSTREC
    5 FORMAT(A3)
      IF (FRSTREC .NE. 3HBAD) GO TO 200
      NFILSKP = 0
      NRECSKP = 1
      CALL SKIPEOF
  200 CALL BUFTEMP
      IF (IEXIT .EQ. 3HYES) GO TO 400
      DO 300 I=1,NCHANN
      DO 300 K=1,LENARR
  300 SUMTEMP(I) = SUMTEMP(I) + TEMP(I,K)
      IF (NRECOR .LE. NAVEREC*MULREC) GO TO 200
      GO TO 500
  400 NAVEREC = NRECOR-(NAVEREC*(MULREC-1))
  500 DO 700 I=1,NCHANN
      AVETEMP(I) = SUMTEMP(I)/(NAVEREC*LENARR)
      SUMTEMP(I) = 0.0
      AVETEMP(I) = (SLOPE(I)*AVETEMP(I) + ZEROTAP(I))/GAIN(I)
      FACTOR1 = AVETEMP(I)/EXCITVO
      FACTOR2= RESIS(I,1)/(RESISR(I) + RESIS(I,1))
      AVETEMP(I) = RESIS(I,2)/(FACTOR2-FACTOR1)- RESIS(I,2)
      IF (I .GT. 1)GO TO 700
      DO 600J=2,NCHANN
  600 RESISR(J) = AVETEMP(I)
  700 AVETEMP(I) = A+B*AVETEMP(I) + C*AVETEMP(I)**2 +D*AVETEMP(I)**3 +
     .             E*AVETEMP(I)**4
      TIME = (TIMRAT*(ITIME-ZEROTIM)/10000.)
      WRITE(6,2)TIME,NRECOR,(AVETEMP(I),I=1,NCHANN)
    2 FORMAT(1H0,* TEMPERATURES AVERAGED OVER 10 MINUTE INTERVALS.  TIME
     . *F9.3* SECS.  RECORD NUMBER*I4//2X*CHANNEL 1*4X*CHANNEL 2*4X*CHAN
     .NEL 3*4X*CHANNEL 4*4X*CHANNEL 5*4X*CHANNEL 6*4X*CHANNEL 7*4X*CHANN
     .EL 8*4X*CHANNEL 9*4X*CHANNEL 10*/3X*LEVEL 2*6X*LEVEL 3*6X*LEVEL 1*
     .6X*LEVEL 1*6X*LEVEL 4*6X*LEVEL 4*6X*LEVEL 5*6X*LEVEL 5*6X*LEVEL 6*
     .6X*LEVEL 6*/3X*AMBIENT*8X*DRY*10X*DRY*10X*WET*10X*DRY*10X*WET*10X*
     .DRY*10X*WET*10X*DRY*10X*WET*/3XF6.3* C*5XF6.3* C*5XF6.3* C*5XF6.3*
     . C*5XF6.3* C*5XF6.3* C*5XF6.3* C*5XF6.3* C*5XF6.3* C*5XF6.3* C*)
      CALL HUMID
      IF (PLOT .EQ. 3HYES) CALL PLOTEMP
      MULREC = MULREC + 1
```

```
110                 NULREC = NULREC + 1
```



```
                    IF (IEXIT .NE. 3HYES) GO TO 200
                    END
```

```
      SUBROUTINE TAPECAL
      COMMON/BTAPECA/NCALVAL,VARITP,ACTVOLT(5)
      COMMON/BCALIBR/SLOPE(10),ZEROTAP(10)
      COMMON NFILE,NRECOR,LENARR,NCHANN,TEMP(10,100),WRITDAT
      DIMENSION SUMCAL(10),SUMTAP(10),SQVALUE(10),SUMACT(10),ACT X TAP(1
     .          0),SUMSQ(10,5),RECMEAN(10),TOTMEAN(10,5),TEPMEAN(10),SU
     .          MEAN(10),TEMPSUM(10),STANDEV(10,5)
      ICHECK = 0
      NSAMPLE = 0
      LASTCAL = 0
      ICALVAL = 1
      DO 100 I=1,NCHANN
      SUMEAN(I) = 0.0
      TEMPSUM(I) = 0.0
      SUMCAL(I) = 0.0
      SUMACT(I) = 0.0
      SUMTAP(I) = 0.0
      SQVALUE(I) = 0.0
      ACT X TAP(I) = 0.0
      TEPMEAN(I) = 0.0
      RECMEAN(I) = 0.0
      DO 100J=1,NCALVAL
      STANDEV(I,J) = 0.0
      TOTMEAN(I,J) = 0.0
  100 SUMSQ(I,J) = 0.0
  200 CALL BUFTEMP
      GOTOBUF = 3H NO
       IF (ICALVAL .EQ. NCALVAL) LASTCAL = LASTCAL + 1
      DO 300 K=1,LENARR
      DO 300 I=1,NCHANN
  300 SUMCAL(I) = SUMCAL(I) + TEMP(I,K)
      IF (ICHECK .GT. 0) GO TO 800
      NSAMPLE = NSAMPLE + 1
      WRITE(6,1) NRECOR,ICALVAL,ACTVOLT(ICALVAL)
    1 FORMAT(1H0,5X*RECORD MEANS*4X*RECORD NUMBER*I4,7X*CALIBRATION*I2,
     .4X*INPUT VALUE*F5.1/11X*CHANNEL*10X*MEAN*13X*CUMULATIVE MEAN* 6X
     .*NUMBER RECORDS FOR CUMULATIVE MEAN*)
      DO 600 I=1,NCHANN
      RECMEAN(I) = SUMCAL(I) / LENARR
      WRITE(6,2)I,RECMEAN(I)
    2 FORMAT(1H ,12X,I2,10X,F8.4)
      IF (NRECOR .EQ. 1) GO TO 400
      IF (RECMEAN(I) .GT. TOTMEAN(I,ICALVAL)+ VARITP .O. RECMEAN(I)
     .    .LT. TOTMEAN(I,ICALVAL) - VARITP) GO TO 700
  400 DO 500 K=1,LENARR
  500 SUMSQ(I,ICALVAL) = SUMSQ(I,ICALVAL) + TEMP(I,K)**2
      SUMEAN(I) = SUMEAN(I) + RECMEAN(I)
      SUMCAL(I) = 0.0
      TOTMEAN(I,ICALVAL) = SUMEAN(I)/NSAMPLE
  600 WRITE(6,4) TOTMEAN(I,ICALVAL),NSAMPLE
    4 FORMAT(1H+,46X,F8.4,25X,I3)
      GO TO 200
  700 NSAMPLE = NSAMPLE - 1
      ICALVAL = ICALVAL + 1
  800 IF (ICALVAL .GT. NCALVAL .A. LASTCAL .GT. 3) GO TO 1300
```

```
      800 IF (ICALVAL .GT. NCALVAL .A. LASTCAL .GT. 3) GO TO 1300
```



```
          ICHECK = ICHECK + 1
          WRITE(6,5) NRECOR,ICALVAL,ACTVOLT(ICALVAL)
        5 FORMAT(1H0,5X*TEMPORARY MEANS*8X*RECORD NUMBER*I4,10X*CALIBRATION
         .*I2,4X*INPUT VALUE *F5.1/11X*CHANNEL*10X*MEAN*)
          DO 1100 I=1,NCHANN
          RECMEAN(I) = SUMCAL(I)/LENARR
          WRITE(6,6)I,RECMEAN(I)
        6 FORMAT(1H ,12XI2,10XF8.4)
          SUMCAL(I) = 0.0
          IF (ICHECK .EQ. 1) GO TO 1000
          DO 900 K=1,LENARR
      900 SUMSQ(I,ICALVAL) = SUMSQ(I,ICALVAL) + TEMP(I,K)**2
          TEMPSUM(I) = TEMPSUM(I) + RECMEAN(I)
     1000 IF (RECMEAN(I) .GT. TOTMEAN(I,ICALVAL-1) + VARITP .O. RECMEAN(I)
         .  .LT. TOTMEAN(I,ICALVAL-1) -VARITP) GOTOBUF = 3HYES
     1100 CONTINUE
          IF (ICHECK .GT. 3) GO TO 1300
          IF (GOTOBUF .EQ. 3HYES) GO TO 200
          DO 1200 I=1,NCHANN
          TEMPSUM(I) = 0.0
     1200 SUMSQ(I,ICALVAL) = 0.0
          ICHECK = 0
          ICALVAL = ICALVAL - 1
          GO TO 200
     1300 IEND = ICALVAL - 1
          WRITE(6,8)IEND,ACTVOLT(IEND)
        8 FORMAT(1H0,/,5X*STANDARD DEVIATIONS*10X*CALIBRATION*I2,5X*INPUT VA
         .LUE*F5.1/11X*CHANNEL*10X*RMS*)
          DO 1400 I=1,NCHANN
          STANDEV(I,ICALVAL-1) = SQRT(SUMSQ(I,ICALVAL-1)/(NSAMPLE*LENARR) -
         .                       TOTMEAN(I,ICALVAL-1)**2)
     1400 WRITE(6,9)I,STANDEV(I,IEND)
        9 FORMAT(1H ,12XI2,7XF9.3)
          NSAMPLE = ICHECK - 1
          DO 1500 I=1,NCHANN
          SUMEAN(I) = TEMPSUM(I)
          TOTMEAN(I,ICALVAL) = TEMPSUM(I)/NSAMPLE
     1500 TEMPSUM(I) = 0.0
          ICHECK = 0
          IF (ICALVAL .LE. NCALVAL) GO TO 200
     1550 WRITE(6,10) NRECOR
       10 FORMAT(1H0,5X*ACTUAL VS TAPE VOLTAGE*10X*LEAST SQUARE METHOD*5X*NU
         .MBER RECORDS USED FOR CALCULATIONS*I3)
          DO 1700 I =1,NCHANN
          DO 1600 J=1,NCALVAL
          SUMTAP(I) = SUMTAP(I) + TOTMEAN(I,J)
          SQVALUE(I) = SQVALUE(I) + TOTMEAN(I,J)**2
          ACT X TAP(I) = ACT X TAP(I) + TOTMEAN(I,J) * ACTVOLT(J)
     1600 SUMACT(I) = SUMACT(I) + ACTVOLT(J)
          SLOPE(I) =(SUMACT(I) * SUMTAP(I) - NCALVAL*ACT X TAP(I))/
         .             (SUMTAP(I)**2-NCALVAL*SQVALUE(I))
          ZEROTAP(I) = (SUMTAP(I)*ACT X TAP(I) - SUMACT(I)*SQVALUE(I))/
         .             (SUMTAP(I)**2-NCALVAL*SQVALUE(I))
          WRITE(6,11) I,(ACTVOLT(J),TOTMEAN(I,J),J=1,NCALVAL)
       11 FORMAT(1H ,10X*C H A N N E L*I3/15X*VALUES USED FOR LEAST SQUARE C
```

```
    11  FORMAT(1H ,10X*C H A N N E L*I3/15X*VALUES USED FOR LEAST SQUARE C
```



```
       .ALCULATIONS*10X*INPUT VALUE*5X*TAPE VALUE*/(69XF4.1,11XF6.3))
  1700 WRITE(6,12) SLOPE(I),ZEROTAP(I)
    12  FORMAT(1H0,15X*VALUES OBTAINED FROM LEAST SQUARE CALCULATIONS* 7X*
       .SLOPE*8X*INTERCEPT*/68XF5.3,11XF5.3)
        RETURN
        END
```

```
      SUBROUTINE INSTCAL
      COMMON/BINSTCA/NINSCAL,VARIIN,EXCITVO,RESIS(10,2),CALRES(10,2),
     .               GAIN(10)
      COMMON/BCALIBR/SLOPE(10),ZEROTAP(10)
      COMMON NFILE,NRECOR,LENARR,NCHANN,TEMP(10,100),WRITDAT
      COMMON/BSKPEOF/BADATA,LPACDAT,NRECSKP,NFILSKP
      DIMENSION SUMAVE(10),TSUMAVE(10),SQVALUE(10),STANDEV(10,2),
     .          TOTAVE(10,2),SUMTEMP(10),TSUMSQ(10),ACTUAL(10),
     .          AVEREC(10)
      NRECOR = 0
      BADATA = 3HYES
      DO 100 I=1,NCHANN
      SUMAVE(I) = 0.0
      TSUMAVE(I) = 0.0
      TSUMSQ(I) = 0.0
      SQVALUE(I) = 0.0
      SUMTEMP(I) = 0.0
      DO 100 K=1,NINSCAL
      STANDEV(I,K) = 0.0
  100 TOTAVE(I,K) = 0.0
      INSCAL = 1
      ICHECK = 0
      NSAMPLE = 1
      LEVEL = 10HZERO INPUT
  200 CALL BUFTEMP
      IF (NFILE .GT. 1) GO TO 1050
      GOTOBUF = 3H NO
      DO 400 I=1,NCHANN
      DO 300 K=1,LENARR
  300 SUMTEMP(I) = SUMTEMP(I) + TEMP(I,K)
      AVEREC(I) = SUMTEMP(I)/LENARR
  400 SUMTEMP(I) = 0.0
      IF (ICHECK .GT. 0) GO TO 800
      IF (NRECOR .EQ. 1) GO TO 500
      IF (ABS(AVEREC(1)) .GT. ABS(TOTAVE(1,INSCAL)) + VARIIN) GO TO 700
  500 WRITE(6,3) NRECOR,INSCAL,LEVEL
    3 FORMAT(1H0,5X*RECORD MEANS*4X*RECORD NUMBER*I4,7X*CALIBRATION*I2,4
     .X*INPUT *A10/11X*CHANNEL*10X*MEAN*13X*CUMULATIVE MEAN*6X*NUMBER OF
     . RECORDS FOR CUMULATIVE MEAN*)
      DO 600 I=1,NCHANN
      SUMAVE(I) = SUMAVE(I) + AVEREC(I)
      TOTAVE(I,INSCAL) = SUMAVE(I) / NSAMPLE
      WRITE(6,2)I,AVEREC(I),TOTAVE(I,INSCAL),NSAMPLE
    2 FORMAT(1H ,12X,I2,10X,F8.4,14X,F8.4,25X,I3)
      DO 600 K=1,LENARR
  600 SQVALUE(I) = SQVALUE(I) + TEMP(I,K)**2
      NSAMPLE = NSAMPLE + 1
      GO TO 200
  700 INSCAL = INSCAL + 1
      LEVEL = 10HFULL SCALE
  800 ICHECK = ICHECK + 1
      IF (ICHECK .EQ. 1) GO TO 200
      IEND = ICHECK - 1
      WRITE(6,4) NRECOR,INSCAL,LEVEL
    4 FORMAT(1H0,5X*TEMPORARY SUM OF MEANS*4X*RECORD NUMBER*I4,7X*CALIBR
```

```
     .ATION*I2,4X*INPUT *A10/11X*CHANNEL*10X*SUM OF MEANS*13X*NUMBER OF
     .RECORDS IN SUM*)
      DO 900 I=1,NCHANN
      TSUMAVE(I) = TSUMAVE(I) + AVEREC(I)
      WRITE(6,5)I,TSUMAVE(I),IEND
    5 FORMAT(1H ,13X,I2,16XF6.3,27XI2)
      DO 900 K=1,LENARR
  900 TSUMSQ(I) = TSUMSQ(I) + TEMP(I,K)**2
      IF (ABS(AVEREC(1)) .GT. ABS(TOTAVE(1,INSCAL-1)) + VARIIN)
     .   GOTOBUF = 3HYES
      IF (ICHECK .GT. 3) GO TO 1100
      IF (GOTOBUF .EQ. 3HYES) GO TO 200
      DO 1000I=1,NCHANN
      TSUMAVE(I) = 0.0
 1000 TSUMSQ(I) = 0.0
      LEVEL = 10HZERO INPUT
      ICHECK = 0
      INSCAL = INSCAL - 1
      GO TO 200
 1050 INSCAL = INSCAL + 1
      NSAMPLE = NSAMPLE - 1
 1100 IEND = INSCAL - 1
      LEVEL = 10HZERO INPUT
      IF (IEND .EQ. 2) LEVEL = 10HFULL SCALE
      WRITE(6,6) IEND,LEVEL
    6 FORMAT(1H0,5X*STANDARD DEVIATIONS*10X*CALIBRATION*I2,5X*INPUT *A10
     ./11X*CHANNEL*10X*RMS*)
      DO 1200 I=1,NCHANN
      STANDEV(I,INSCAL-1) = SQRT(SQVALUE(I)/(NSAMPLE*LENARR)- TOTAVE(I,
     .                           INSCAL-1)**2)
 1200 WRITE(6,7)I,STANDEV(I,IEND)
    7 FORMAT(1H ,12XI2,7XF9.3)
      NSAMPLE = ICHECK - 1
      DO 1300 I=1,NCHANN
      SUMAVE(I) = TSUMAVE(I)
      TOTAVE(I,INSCAL) = SUMAVE(I)/NSAMPLE
      SQVALUE(I) = TSUMSQ(I)
      TSUMAVE(I) = 0.0
 1300 TSUMSQ(I) = 0.0
      ICHECK = 0
      NSAMPLE = NSAMPLE + 1
      LEVEL = 10HFULL SCALE
      IF (INSCAL .LE. NINSCAL) GO TO 200
 1400 DO 1600 I=1,NCHANN
      DO 1500 K=1,NINSCAL
 1500 TOTAVE(I,K) = SLOPE(I) * TOTAVE(I,K) + ZEROTAP(I)
      ACTUAL(I) = EXCITVO*((RESIS(I,1)/(RESIS(I,1)+CALRES(I,1))) - (RESI
     .           S(I,2)/(RESIS(I,2) + CALRES(I,2))))
      GAIN(I) = (TOTAVE(I,2) - TOTAVE(I,1))/ACTUAL(I)
 1600 WRITE(6,8) I,TOTAVE(I,1),TOTAVE(I,2),ACTUAL(I),GAIN(I)
    8 FORMAT(1H0,10X*C H A N N E L*I3/15X*VALUES USED FOR GAIN CALCULATI
     .ONS*10X*INPUT*5X*TAPE VALUE*5X*ACTUAL VALUE*/59X*ZERO*7XF6.3,12X*0
     ..0*/56X*FULL SCALE*4XF6.3,11XF6.3//15X*GAIN COMPUTED*F10.3)
      RETURN
      END
```

```
      END
```



```
      SUBROUTINE BUFTEMP
      COMMON/BBUFTEM/IDENT,MULEOF,IEXIT,NTOTFIL,PLOT,IPARITY,ZEROTIM,
     .                JCLOCK
      COMMON/BSKPEOF/BADATA,LPACDAT,NRECSKP,NFILSKP
      COMMON NFILE,NRECOR,LENARR,NCHANN,TEMP(10,100),WRITDAT
      COMMON/UNPK/ITIME,ICOMWRD(200),IDATWRD(1000)
      IF (IDENT .EQ. 3HYES .A. BADATA .EQ. 3H NO .A. NRECOR .EQ. 0)
     .    CALL HEADER
      CORTIME = 3H NO
      BADATA = 3H NO
  100 NRECOR = NRECOR + 1
      BUFFER IN(1,1)(ITIME,ICOMWRD(LPACDAT))
      IF(UNIT(1))500,200,400
  200 WRITE(6,1) NRECOR,NFILE
    1 FORMAT(1H0,* THERE ARE*I5* RECORDS ON FILE*I3)
      NRECOR = 0
      IF (MULEOF .NE. 3HYES) GO TO 300
      NFILE = NFILE + 1
      IF (NFILE .GT. NTOTFIL) GO TO 300
      GO TO 100
  300 IF (PLOT .EQ. 3HYES) IEXIT = 3HYES
      IF (PLOT .EQ. 3HYES) RETURN
      CALL EXIT
  400 WRITE(6,2) NRECOR,NFILE
    2 FORMAT(1H0,* PARITY ERROR IN DATA, RECORD*I5* FILE*I3)
      IPARITY = IPARITY + 1
      WRITE (6,3) IPARITY
    3 FORMAT(1H0,* THERE HAVE BEEN*I3* PARITY ERRORS*)
      CALL UNPAK
      CALL SORT
      IF (WRITDAT .EQ. 3H NO) CALL DATWRIT
      GO TO 600
  500 CALL UNPAK
      CALL SORT
  600 IF (NRECOR .EQ. 1 .A. ITIME .NE. 0) ZEROTIM = ITIME
      IF (ITIME-999999 .GT. -12000) CORTIME = 3HYES
      ITIME = ITIME + JCLOCK * 999999
      IF (CORTIME .EQ. 3HYES) JCLOCK = JCLOCK + 1
      RETURN
      END
```

```
          SUBROUTINE HEADER
          COMMON NFILE,NRECOR,LENARR,NCHANN,TEMP(10,100),WRITDAT
          DIMENSION ID(2)
          BUFFER IN(1,0) (ID(1),ID(2))
          IF (UNIT(1)) 200,100,100
      100 WRITE(6,1) NFILE
        1 FORMAT(1H0,* PARITY ERROR OR EOF IN HEADER OF FILE NUMBER*I3)
      200 WRITE(6,2) NFILE,(ID(I),I=1,2)
        2 FORMAT(1H0,* HEADER ON FILE*I3* IS *2I10)
          RETURN
          END
```

```
      SUBROUTINE SORT
      COMMON NFILE,NRECOR,LENARR,NCHANN,TEMP(10,100),WRITDAT
      COMMON/BSORT/IBEGSKP,FACTOR,ISKIP
      COMMON/UNPK/ITIME,ICOMWRD(200),IDATWRD(1000)
      M = IBEGSKP
      L = 0
      DO 200 I= 1,LENARR
      DO 100 K=1,NCHANN
      TEMP(K,I) = IDATWRD(M + L) * FACTOR
  100 L = L + 1
      L = 0
  200 M=M + ISKIP
      IF (WRITDAT .EQ. 3HYES) CALL DATWRIT
      RETURN
      END
```

```
      SUBROUTINE DATWRIT
      COMMON/UNPK/ITIME,ICOMWRD(200),IDATWRD(1000)
      COMMON NFILE,NRECOR,LENARR,NCHANN,TEMP(10,100),WRITDAT
      WRITE(6,1) NRECOR, ITIME
    1 FORMAT(1H0,* RECORD NUMBER*I5* ITIME IS*I10)
      DO 100 I=1,NCHANN
  100 WRITE(6,2) I,(TEMP(I,K),K=1,LENARR)
    2 FORMAT(1H0,* TAPE CHANNEL NUMBER*I3,/,(16F8.3))
      RETURN
      END
```

```
      SUBROUTINE SKIPEOF
      COMMON/BSKPEOF/BADATA,LPACDAT,NRECSKP,NFILSKP
      COMMON/BBUFTEM/IDENT,MULEOF,IEXIT,NTOTFIL,PLOT,IPARITY,ZEROTIM,
     .                JCLOCK
      COMMON/UNPK/ITIME,ICOMWRD(200),IDATWRD(1000)
      COMMON NFILE,NRECOR,LENARR,NCHANN,TEMP(10,100),WRITDAT
      NREC = 0
      IF (NFILE .GT. NFILSKP) GO TO 500
  100 BUFFER IN(1,1)(ITIME,ICOMWRD(LPACDAT))
      IF (UNIT(1))300,400,200
  200 LEN = LENGTH(1)
      NREC = NREC + 1
      NRECOR = NRECOR + 1
      WRITE(6,1) NFILE,NRECOR,NREC,LEN
    1 FORMAT(1H0,5X*PARITY ERROR OCCURRED WHILE SKIPPING FILE*I2* RECORD
     .*I4*,*I3* RECORDS HAVE BEEN SKIPPED.  LENGTH OF RECORD*I4)
      GO TO 100
  300 LEN = LENGTH(1)
      NREC = NREC + 1
      NRECOR = NRECOR + 1
      IF (LEN .NE. LPACDAT + 1) WRITE(6,2)NRECOR,NFILE,LEN,NREC
    2 FORMAT(1H0,5X*RECORD ENCOUNTERED OF IMPROPER LENGTH.  RECORD*I4* FI
     .ILE*I2* LENGTH*I4,2XI3* RECORDS HAVE BEEN SKIPPED*)
      GO TO 100
  400 WRITE(6,3)NREC,NFILE,NRECOR
    3 FORMAT(1H0,5XI3* RECORDS HAVE BEEN SKIPPED ON FILE*I2*.  THERE WER
     .E*I4* RECORDS ON THIS FILE*)
      NFILE = NFILE + 1
      NREC = 0
      NRECOR = 0
      IF (IDENT .EQ. 3HYES) CALL HEADER
      IF (NFILE .LE. NFILSKP) GO TO 100
  500 IF (NRECSKP .EQ. 0) RETURN
      DO 900 I=1,NRECSKP
      BUFFER IN(1,1)(ITIME,ICOMWRD(LPACDAT))
      IF (UNIT(1))800,700,600
  600 NREC = NREC + 1
      NRECOR = NRECOR + 1
      LEN = LENGTH(1)
      WRITE(6,4)NRECOR,NFILE,NREC,LEN
    4 FORMAT(1H0,5X*PARITY ERROR OCCURRED WHILE SKIPPING RECORD*I4* ON F
     .ILE*I2*,*2XI3* RECORDS HAVE BEEN SKIPPED.  LENGTH*I4)
      GO TO 900
  700 WRITE(6,5)NRECOR,NFILE,NREC
    5 FORMAT(1H0,5X*EOF OCCURRED WHILE SKIPPING RECORDS.  LAST RECORD WA
     .S*I4* ON FILE*I2,2XI3* RECORDS HAD BEEN SKIPPED.*)
      GO TO 900
  800 NRECOR = NRECOR + 1
      NREC = NREC + 1
      LEN = LENGTH(1)
      IF (LEN.NE. LPACDAT + 1) WRITE(6,2)NRECOR,NFILE,LEN,NREC
  900 CONTINUE
      WRITE(6,6)NREC,NFILE,NRECOR
    6 FORMAT(1H0,5XI3* RECORD(S) HAVE BEEN SKIPPED ON FILE*I2*.  RECORD
     .NUMBER IS*I4)
```

```
    55            .NUMBER IS*I4)
```



```
                  RETURN
                  END
```

```
      SUBROUTINE HUMID
      COMMON/BHUMID/SIGMA,BARPRES,HEATLAT,CP,CPV,HUMIDI(10)
      COMMON/BPLOTEM/AVETEMP(10),TIME,ELEV(6),LABELX(4),LABELY(4),LTITLE
     .                (4)
      M = 1
      DO 100I=4,10,2
      RHOS = 10.*(10.**((22.5518*(AVETEMP(I)+273.16)-2937.4)/(AVETEMP(I)
     .        +273.16)))/((AVETEMP(I)+273.16)**4.9283)
      WMIXRA = (SIGMA*RHOS*AVETEMP(I))/(BARPRES- RHOS*AVETEMP(I))
      HUMIDI(M) = WMIXRA*HEATLAT-(AVETEMP(I-1) - AVETEMP(I))*CP /
     .            (CPV*(AVETEMP(I-1) - AVETEMP(I)) + HEATLAT)
  100 M = M + 1
      WRITE(6,1) (HUMIDI(I),I=1,4)
    1 FORMAT(1H0,5X*HUMDITIES*/5X* LEVEL 1*5X* LEVEL 4*5X* LEVEL 5*5X* L
     .EVEL 6*/4(5XF8.3))
      RETURN
      END
```

```
              .NUMBER IS*I4)
```



```
      SUBROUTINE PLOTEMP
      COMMON/BPLOTEM/AVETEMP(10),TIME,ELEV(6),LABELX(4),LABELY(4),LTITLE
     .                (4)
      DIMENSION TEMP(6)
      TEMP(1) = AVETEMP(3)
      TEMP(2) = AVETEMP(1)
      TEMP(3) = AVETEMP(2)
      TEMP(4) = AVETEMP(5)
      TEMP(5) = AVETEMP(7)
      TEMP(6) = AVETMEP(9)
      WRITE(6,1) TIME,(ELEV(I),TEMP(I),I=1,6)
    1 FORMAT(1H0,* TEMPERATURE PROFILE VALUES*/3X* AVERAGE OF POINTS OVE
     .R TEN MINUTE INTERVALS.  TIME OF PLOT *F10.3* SECS*/3X* ELEVATION,
     .M       TEMPERATURE,C*/6(6X,F6.3,10XF6.3/))
      CALL IDIOT (TEMP,ELEV,6,1,DUM,LABELX,LABELY,LTITLE,+1)
      RETURN
      END
```

```
                                       IDENT UNPAK
                           *              INSERT LENGTHS OF PACKED AND UNPACKED ARRAYS
                  311      LENGTHA    SET   201
                 1750      LENGTHB    SET   1000
                           *
                                      USE   /UNPK/
        0                  NE         BSS   LENGTHA
      311                  B          BSS   LENGTHB
                                      USE
                                      ENTRY UNPAK
        0                  UNPAK      BSS   1
        1 7170000001                  SX7   1B
                 7100004000           SX0   4000B
        2 43214                       MX2   12
             5110000000 C             SA1   NE         A(1) --- FIRST WORD OF ARRAY (TIME WORD) IS IGNORED
        3 6160000310 C                SB6   B-1        B(J) BASE
                 6170002260 C         SB7   B+LENGTHB-1    B(LAST)
        4 6150000060                  SB5   48
                 6110000074           SB1   60
        5 5011000001       GET60      SA1   A1+1       GET A(I)
        6 6166000001       GET12      SB6   B6+1
                 11621                BX6   X2*X1      MASK OUT 12 BITS
                      67515           SB5   B1-B5            RIGHT SHIFT
        7 22656                       LX6   B5,X6             BUT
             67515                    SB5   B1-B5            AVOID SIGN EXTENSION
                           *
                 11760                BX7   X6*X0            CK FOR SIGN BIT
       10 0307000011 +                ZR    X7,STORB
                 15660                BX6   -X0*X6           MASK OUT SIGN BIT
                      14666           BX6   -X6
                           *
       11                  STORB      BSS   0
       11 21601                       AX6   1          DELETE ZERO-BIT RIGHT-FILL
             56660                    SA6   B6         STORE IN B(J)
                 0467000000 +         EQ    B6,B7,DONE
       12 0550000014 +                NE    B5,B0,INMID
                 6150000060           SB5   48
       13 43214                       MX2   12
             0400000005 +             EQ    GET60
       14 20260            INMID      LX2   48
             6155777763               SB5   B5-12
       15 0400000006 +                EQ    GET12
                      0 +  DONE       EQU   UNPAK
       16                             END

                  46302    STORAGE USED           44 STATEMENTS       10 SYMBOLS
                           6400 ASSEMBLY          0.340 SECONDS       23 REFERENCES
```

```
          SUBROUTINE VOLTADJ
          COMMON/BVOLTAD/ISCALE
          COMMON/BCALIBR/SLOPE(2),ZEROTAP(2),SLOPEAN,ANINTER,SLOPEHW,
         .               SLOPEWD,WDINTER,SLOPEMD,DMINTER
          GO TO (100,200,300),ISCALE
      100 SLOPEAN = 47.736
          ANINTER = -0.411
          WRITE(6,1)ISCALE,SLOPEAN,ANINTER
        1 FORMAT(1H0,5X*ANEMOMETER VALUES*5X* SCALE*I2/10X*SLOPE*5X*INTERCEP
         .T*/10XF6.3,6XF6.3)
          RETURN
      200 SLOPEAN = 93.021
          ANINTER = 0.451
          WRITE(6,1)ISCALE,SLOPEAN,ANINTER
          RETURN
      300 SLOPEAN = 0.0
          ANINTER = 0.0
          WRITE(6,1)ISCALE,SLOPEAN,ANINTER
          RETURN
          END
```

```
      SUBROUTINE SPEED
      COMMON/BSPEED/SUMVELO,ISAMPLE,IDATAHW,SUMVOLT,TIMRAT1,CHANNEL1,
     .               DIGRAT1,TIMECHG,VOLTCHG,MULTIME,TIMEHW,DCSUPRE
     .               ,FRSTSPD,WRITAPE,PRINTOK
      COMMON/BBUMPUP/TIME1(702),VELOC2(702)
      COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
     .               ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
      COMMON/BCALIBR/SLOPE(2),ZEROTAP(2),SLOPEAN,ANINTER,SLOPEHW,
     .                SLOPEWD,WDINTER,SLOPEMD,DMINTER
      COMMON/UNPK1/ITIME1,ICOMWRD(201),IDATWRD(1000)
      COMMON/BBUMP/WRITIME(100),VELOC1(100)
      COMMON/BBUFLA1/NTOTFI1,JCLOCK1,IEXIT,TIMADA,NREC
      PRINTOK = 3H NO
      KFIRST = 1
      IF (FRSTSPD .EQ. 3HYES .A. NRECOR1 .NE. 0) GO TO 25
      CALL BUFLAS1
      IF (IEXIT .EQ. 3HYES) GO TO 200
   25 DO 100 IDATAWS = KFIRST,LENARR1
      VELOC1(IDATAWS) = SLOPE(1)* VOLT(1,IDATAWS) + ZEROTAP(1)
      VELOC1(IDATAWS) = (SLOPEAN* VELOC1(IDATAWS) + ANINTER) *0.3048
      VELOC2(IDATAHW) = SLOPE(2)*VOLT(2,IDATAWS) + ZEROTAP(2)+ DCSUPRE
      IF (FRSTSPD .EQ. 3HYES) GO TO 50
      SUMVELO = SUMVELO + VELOC1(IDATAWS)
      ISAMPLE = ISAMPLE + 1
      SUMVOLT = SUMVOLT + VELOC2(IDATAHW)
   50 TIME1(IDATAHW) = TIMRAT1*((ITIME1-ZEROTM1)/10000 + ((IDATAWS  )
     .*CHANNEL1)/DIGRAT1)
      IF (TIME1(IDATAHW) .GE. TIMECHG .A. VOLTCHG .EQ. 3HYES)
     .    CALL VOLTADJ
      IF (TIME1(IDATAHW) .GE. MULTIME*TIMEHW) GO TO 200
  100 IDATAHW = IDATAHW + 1
      FRSTSPD = 3H NO
      ISTART = IDATAHW - LENARR1+KFIRST-1
      DO 115 I=KFIRST,LENARR1
      WRITIME(I) = TIME1(ISTART)
  115 ISTART = ISTART + 1
  120 IF (NRECOR1 .GE. 2 .A. NRECOR1 .LE. 8) GO TO 125
      I = NRECOR1 - 1
      IF (MOD(I,30) .EQ. 0)GO TO 125
      GO TO 160
  125 PRINT 3,NRECOR1
    3 FORMAT (1H0,* ANEMOMETER VELOCITY    RECORD NUMBER*I4//2X*TIME,SEC
     .S*5X* VELOCITY,M/SEC*10X* TIME,SECS*5X* VELOCITY,M/SEC*10X* TIME,S
     .ECS*5X* VELOCITY,M/SEC*/)
      PRINTOK = 3HYES
      WRITE(6,1)(WRITIME(I),VELOC1(I),I=1,LENARR1)
    1 FORMAT(1H ,1XF8.3,11XF6.3,15XF8.3,11XF6.3,15XF8.3,11XF6.3)
      IF (WRITAPE .NE. 3HYES) RETURN
  160 WRITE(6,12) NRECOR1
   12 FORMAT(1H ,* NRECOR1=*I5)
      BUFFER OUT(3,1) (WRITIME(1),VELOC1(100))
      IF (UNIT(3)) 400,180,190
  180 WRITE(6,6) NRECOR1,NFILE1
    6 FORMAT(1H ,* EOF ON RECORD NUMBER*I10* FILE NUMBER*I3)
      GO TO 400
```

```
  190 WRITE(6,7) NRECOR1,NFILE1
    7 FORMAT(1H ,* PARITY ERROR INPUT ON RECORD NUMBER*I10* FILE NUMBER
     .*I5)
      GO TO 400
  200 AVEVELO = (SUMVELO/ISAMPLE)/0.3048
      AVEVOLT = SUMVOLT/ISAMPLE
      KFIRST = IDATAWS + 1
      DO 250 JK = 1,IDATAWS
  250 WRITIME(JK) = TIME1(IDATAHW-IDATAWS + JK)
      SUMVELO = 0.0
      SUMVOLT = 0.0
      ISAMPLE = 0
      MULTIME = MULTIME + 1
      HWINTER = SQRT(AVEVELO)-SLOPEHW*AVEVOLT**2
      DO 300 I=1,IDATAHW
  300 VELOC2(I) =((SLOPEHW*VELOC2(I)**2+HWINTER)**2)*0.3048
      IF (NRECOR1 .GE. 2 .A. NRECOR1 .LE. 8) GO TO 305
      I = NRECOR1 - 1
      IF (MOD(I,30) .EQ. 0)GO TO 305
       GO TO 310
  305 WRITE(6,2) TIME1(1),TIME1(IDATAHW),NRECOR1,(TIME1(I),VELOC2(I), I=
     .1,IDATAHW)
    2 FORMAT(1H0,* HOT WIRE VELOCITY, CALCULATED FOR TIME PERIOD FROM*
     .F7.2* TO*F7.2* RECORD NUMBER*I3//* TIME,SECS*5X* VELOCITY,M/SEC*10
     .X* TIME,SECS*5X* VELOCITY,M/SEC*10X* TIME,SECS*5X* VELOCITY,M/SEC*
     .//,(1X,F8.3,12X,F6.3,14X,F8.3,12X,F6.3,14X,F8.3,12X,F6.3))
      IF (WRITAPE .EQ. 3HYES) GO TO 310
      IF (IEXIT .EQ. 3HYES)RETURN
  307 IDATAHW = 1
      IF (KFIRST .GT. LENARR1)GO TO 120
      GO TO 25
  310 M = 1
      PRINT 13,NRECOR1,IDATAHW
   13 FORMAT(1H ,* H. W. NRECOR1=*I10* NUMBER OF WORDS = *I10)
      LAST = 2
      IF (IDATAHW .LE. 301)LAST = 1
      DO 370 I=1,LAST
      N = M + 300
      IF (I .EQ. LAST) N = IDATAHW
      BUFFER OUT(3,1) (TIME1(M),TIME1(N))
      IF (UNIT(3))340,330,320
  320 WRITE(6,8) NRECOR1,NFILE1,M,N
    8 FORMAT(1H ,* PARITY ERROR ON HW TIME, RECORD*I10* FILE*I3* M=*I5*
     .N=*I5)
      GO TO 340
  330 WRITE(6,9) NRECOR1,NFILE1,M,N
    9 FORMAT(1H ,* EOF ON HW TIME, RECORD*I10* FILE*I3* M=*I5* N=*I5)
  340 BUFFER OUT(3,1) (VELOC2(M),VELOC2(N))
      IF (UNIT(3)) 370,360,350
  350 WRITE(6,10) NRECOR1,NFILE1,M,N
   10 FORMAT(1H ,* PARITY ERROR ON HW VELOCITY, RECORD*I10* FILE*I3* M=*
     .I5* N=*I5)
      GO TO 370
  360 WRITE(6,11) NRECOR1,NFILE1,M,N
   11 FORMAT(1H ,* EOF ON HW VELOCITY, RECORD* I10* FILE*I3* M=*I5* N=*I
```

```
                  .5)
            370 M = M + 301
                IDATAHW = 1
                IF (IEXIT .EQ. 3HYES) RETURN
                IF (KFIRST .GT. LENARR1)GO TO 120
                GO TO 25
            400 RETURN
                END
```

```
      SUBROUTINE AVEWIND
      COMMON/BAVEWIN/JSAMPLE,SUMWIND,MULTIM1,TIMAVWD,AVEWD(3000)
     .               ,LASTIME
      COMMON/BSPEED/SUMVELO,ISAMPLE,IDATAHW,SUMVOLT,TIMRAT1,CHANNE1,
     .              DIGRAT1,TIMECHG,VOLTCHG,MULTIME,TIMEHW,DCSUPRE
     .             ,FRSTSPD,WRITAPE,PRINTOK
      COMMON/BBUMPUP/TIME1(702),VELOC2(702)
      COMMON/BLOCK1/LENARR1,WINDIRE(100),NRECOR1,NFILE1,
     .              ZEROTM1,DIRMIRR(100),VOLT(2,100),WRIDAT1
      COMMON/BCALIBR/SLOPE(2),ZEROTAP(2),SLOPEAN,ANINTER,SLOPEHW,
     .               SLOPEWD,WDINTER,SLOPEMD,MDINTER
      J= IDATAHW - LENARR1 - 1
      DO 100 K=1,LENARR1
      J = J + 1
      JSAMPLE = JSAMPLE + 1
      SUMWIND = SUMWIND + WINDIRE(K)
      IF (K .LT. LENARR1)GO TO 100
  150 AVEWD(MULTIM1) = SUMWIND/JSAMPLE
      AVEWD(MULTIM1) = SLOPEWD*AVEWD(MULTIM1) + WDINTER
      MULTIM1 = MULTIM1 + 1
      SUMWIND = 0.0
      JSAMPLE = 0
  100 CONTINUE
      IF (WRITAPE .EQ. 3HYES)RETURN
      WRITE (6,1) TIMAVWD,MULTIM1,(AVEWD(I),I=LASTIME,MULTIM1)
    1 FORMAT(1H ,* WIND DIRECTIONS IN DEGREES,  MEANS OF DATA FOR*F5.1*
     .SEC INTERVALS.  THIS IS INTERVAL NUMBER*I4/(10F10.3))
      LASTIME = MULTIM1
      RETURN
      END
```

```
      SUBROUTINE LASWRIT
      COMMON TIME2(200),VELOLAS(200),IPOINT
      COMMON/BLOCK2/LENARR2,SYNC(1500),YLASER(1500),NRECOR2,NFILE2,
     .              ZEROTM2,WRIDAT2,NTAPE2
      WRITE(6,5) NRECOR2,IPOINT
    5 FORMAT(1H ,* NRECOR2=*I10* NUMBER OF WORDS =*I10)
   50 M = 1
      N = IPOINT - 1
      BUFFER OUT(3,1) (TIME2(M),TIME2(N))
      IF (UNIT(3))300,200,100
  100 WRITE(6,1) NRECOR2,NFILE2,M,N
    1 FORMAT(1H ,* PARITY ERROR ON LASER TIME RECORD NUMBER*I10*FILE NUM
     .BER*I3* M=*I5* N=*I5)
      GO TO 300
  200 WRITE(6,2) NRECOR2,NFILE2,M,N
    2 FORMAT(1H ,* EOF ON LASER TIME RECORD*I10* FILE *I3* M=*I5* N=*I5)
  300 BUFFER OUT(3,1) (VELOLAS(M),VELOLAS(N))
      IF (UNIT(3))600,500,400
  400 WRITE(6,3) NRECOR2,NFILE2,M,N
    3 FORMAT(1H ,* PARITY ERROR ON LASER VELOCITY, RECORD*I10* FILE*I3*
     .M=*I5* N=*I5)
      GO TO 600
  500 WRITE(6,4) NRECOR2,NFILE2,M,N
    4 FORMAT(1H ,* EOF ON LASER VELOCITY, RECORD*I10*FILE*I3* M=*I5* N=*
     .I5)
  600 TIME2(1) = TIME2(IPOINT)
      VELOLAS(1) = VELOLAS(IPOINT)
      RETURN
      END
```

```
                                                IDENT UNPAK1
                                  *               INSERT LENGTHS OF PACKED AND UNPACKED ARRAYS
                         312      LENGTHA      SET   202
                        1750      LENGTHB      SET   1000
                                  *
                                               USE   /UNPK1/
         0                        NE           BSS   LENGTHA
       312                        B            BSS   LENGTHB
                                               USE
                                               ENTRY UNPAK1
         0                        UNPAK1       BSS   1
         1  7170000001                         SX7   1B
                      7100004000               SX0   4000B
         2  43214                              MX2   12
                 5110000000 C                  SA1   NE         A(1) --- FIRST WORD OF ARRAY (TIME WORD) IS IGNORED
         3  6160000311 C                       SB6   B-1        B(J) BASE
                      6170002261 C             SB7   B+LENGTHB-1     B(LAST)
         4  6150000060                         SB5   48
                      6110000074               SB1   60
         5  5011000001            GET60        SA1   A1+1       GET A(I)
         6  6166000001            GET12        SB6   B6+1
                      11621                    BX6   X2*X1      MASK OUT 12 BITS
                           67515               SB5   B1-B5           RIGHT SHIFT
         7  22656                              LX6   B5,X6            BUT
                 67515                         SB5   B1-B5           AVOID SIGN EXTENSION
                                  *
                      11760                    BX7   X6*X0           CK FOR SIGN BIT
        10  0307000011 +                       ZR    X7,STORB
                      15660                    BX6   -X0*X6          MASK OUT SIGN BIT
                           14666               BX6   -X6
                                  *
        11                        STORB        BSS   0
        11  21601                              AX6   1          DELETE ZERO-BIT RIGHT-FILL
                 56660                         SA6   B6         STORE IN B(J)
                      0467000000 +             EQ    B6,B7,DONE
        12  0550000014 +                       NE    B5,B0,INMID
                      6150000060               SB5   48
        13  43214                              MX2   12
                 0400000005 +                  EQ    GET60
        14  20260                 INMID        LX2   48
                 6155777763                    SB5   B5-12
        15  0400000006 +                       EQ    GET12
                             0 +  DONE         EQU   UNPAK1
        16                                     END

                       46302      STORAGE USED              44 STATEMENTS        10 SYMBOLS
                                  6400 ASSEMBLY           0.338 SECONDS          23 REFERENCES
```

```
                                      IDENT UNPAK2
                             *          INSERT LENGTHS OF PACKED AND UNPACKED ARRAYS
                     1132    LENGTHA  SET   602
                     5670    LENGTHB  SET   3000
                             *
                                      USE   /UNPK2/
      0                      NE       BSS   LENGTHA
   1132                      B        BSS   LENGTHB
                                      USE
                                      ENTRY UNPAK2
      0                      UNPAK2   BSS   1
      1 7170000001                    SX7   1B
                 7100004000           SX0   4000B
      2 43214                         MX2   12
           5110000000 C               SA1   NE        A(1) --- FIRST WORD OF ARRAY (TIME WORD) IS IGNORED
      3 6160001131 C                  SB6   B-1       B(J) BASE
                 6170007021 C         SB7   B+LENGTHB-1    B(LAST)
      4 6150000060                    SB5   48
                 6110000074           SB1   60
      5 5011000001           GET60    SA1   A1+1      GET A(I)
      6 6166000001           GET12    SB6   B6+1
                 11621                BX6   X2*X1     MASK OUT 12 BITS
                      67515           SB5   B1-B5          RIGHT SHIFT
      7 22656                         LX6   B5,X6          BUT
           67515                      SB5   B1-B5          AVOID SIGN EXTENSION
                             *
                 11760                BX7   X6*X0          CK FOR SIGN BIT
     10 0307000011 +                  ZR    X7,STORB
                 15660                BX6   -X0*X6         MASK OUT SIGN BIT
                      14666           BX6   -X6
                             *
     11                      STORB    BSS   0
     11 21601                         AX6   1         DELETE ZERO-BIT RIGHT-FILL
           56660                      SA6   B6        STORE IN B(J)
                 0467000000 +         EQ    B6,B7,DONE
     12 0550000014 +                  NE    B5,B0,INMID
                 6150000060           SB5   48
     13 43214                         MX2   12
           0400000005 +               EQ    GET60
     14 20260                INMID    LX2   48
           6155777763                 SB5   B5-12
     15 0400000006 +                  EQ    GET12
                        0 +  DONE     EQU   UNPAK2
     16                               END

                    43417    STORAGE USED            44 STATEMENTS       10 SYMBOLS
                             6400 ASSEMBLY           0.339 SECONDS       23 REFERENCES
```

```
                                                 IDENT     UNLOADW
                                                 ENTRY     UNLOADW
                                                 USE       DATA.
      246  03111720000000000000       CIOC       VFD       18/3LCIO,2/1,40/0
      247  15230700000000000000       MSGC       VFD       18/3LMSG,42/0
      250  01022401000000000000       ABTCS      VFD       18/3LABT,6/1,36/0
      251  01022400000000000000       ABTC       VFD       18/3LABT,42/0
      252  15051500000000000000       MEMC       VFD       18/3LMEM,42/0
      253  22031400000000000000       RCLC       VFD       18/3LRCL,42/0
      254  05160400000000000000      ENDC        VFD       18/3LEND,42/0
      255  01030520000000000256 +     CNTCC      VFD       18/3LACE,2/1,22/0,18/CNTCCB
      256  00000000000000000000       CNTCCB     DATA      0         . 10B  READ FORWARD     40B FOR BACKSP
      257  06111405552701235522       MSG1       DATA      C*FILE WAS REWOUND BEFORE RETURN *
      263  55251614170104551716       MSG2       DATA      C* UNLOAD ON NON-TAPE FILE *
      266  55251614170104551716       MSG3       DATA      C* UNLOAD ON UNDEFINED FILE *
      271  55202217072201155503       MSG4       DATA      C* PROGRAM CONTINUED *
      274  55222516550102172224       MSG5       DATA      C* RUN ABORTED WITH SPEC PROCESSING *
      300  55222516550516241122       MSG6       DATA      C* RUN ENTIRELY ABORTED - DUMP *
      304  55222516550102172224       MSG7       DATA      C* RUN ABORTED WITHOUT DUMP *
      307  55270111241116075506       MSG9       DATA      C* WAITING FOR NEXT REEL - GO TO CONTINUE*
      314  55222516550516040504       MSG8       DATA      C* RUN ENDED WITH NO DUMP -NORMAL CC STREAM *
      321  00000000000000000000       SAVEW      DATA      0
                                                 USE       *



                                      CALLPP     MACRO     A
                                                 IFC       NE,**A*,1
                                                 BX7       X.A
                                      +          SA5       1
                                                 NZ        X5,*
                                      +          SA7       A5
                                      +          SA5       A5
                                                 NZ        X5,*
                                                 ENDM



                                      CLOSER     MACRO
                                                 LOCAL     LOPE
                                                 SA3       B2+2
                                                 SA4       A3+1
                                                 IX5       X3-X4
                                                 ZR        X5,LOPE        . IN EQ OUT
                                                 MX0       18
                                                 LX0       18
                                                 SA3       B2
                                                 BX7       -X0*X3
                                                 BX4       X0*X3
                                                 SX1       3B
                                                 BX3       -X1*X4
```

```
               ZR      X3,LOPE      . FILE NOT OPENED
               SX1     4B           . WRITE MASK
               ZR      X3,LOPE      .NOT OPEN FOR WRITE
               SX1     500B
               BX3     X1*X4
               NZ      X3,LOPE      . SPECIAL FUNCTION CODE
               SX0     2B
               BX6     X0*X4
               SX5     24B          . WRITE CODE
               BX6     X6+X5        . ADD PARITY BIT
               BX6     X7+X6        . ADD LOGICAL FILE NAME
               SA6     B2           . RESET FIRST WORD FET
               SX6     B2
               SA5     CIOC
               BX7     X5+X6        . ADD FET ADR TO CALL WORD
               CALLPP
      LOPE     BSS     0           END INSTRUCTION IN MACRO
               ENDM



      REWIND   MACRO
               CLOSER
               SA5     B2
               MX0     18
               LX0     18
               BX6     -X0*X5       .SAVE FILE NAME
               SX0     2B
               BX5     X5*X0
               SX4     50B          .REWIND CODE
               BX5     X4+X5        .ADD PARITY BIT
               BX6     X6+X5        .ADD FILE NAME
               SA6     B2
               SX6     B2
               SA5     CIOC
               BX7     X5+X6
               CALLPP
               SX6     MSG1
               SA5     MSGC
               BX7     X6+X5
               CALLPP
               ENDM



      WAITER   MACRO
               LOCAL   LOP
               SX4     MSG9         .GET WAIT DAYFILE MESSAGE
               SA5     MSGC         .GET PP CALL WORD
               BX7     X4+X5        .ADD ADDRESS TO CALL WORD
               CALLPP
```

```
                                          SX0     10000B
                                          SA5     B0
                                          BX6     X5+X0
                                          SA6     A5
                                          SA2     RCLC
                                 LOP      CALLPP  2
                                          SA5     B0
                                          BX5     X5*X0
                                          NZ      X5,LOP
                                          ENDM


   3                             UNLOAD   MACRO
                                          CLOSER
                                          SA5     B2
                                          MX0     18
                                          LX0     18
                                          BX6     -X0*X5         .SAVE FILE NAME
                                          SX0     2B
                                          BX5     X5*X0
                                          SX4     60B            .UNLOAD CODE
                                          BX5     X4+X5          .ADD PARITY BIT
                                          BX6     X6+X5          .ADD FILE NAME
                                          SA6     B2
                                          SX6     B2
                                          SA5     CIOC
                                          BX7     X5+X6
                                          CALLPP
                                          ENDM


                                 PMSG     MACRO   A
                                          SX6     A
                                          SA5     MSGC
                                          BX7     X6+X5
                                          CALLPP
                                          ENDM


                                          LIST    -R
          0                      UNLOADW  BSSZ    1
          1  5021000001                   SA2     A1+1
          2  0302000003 +        +        ZR      X2,*+1
                        53220    -        SA2     X2
          3  10722               +        BX7     X2
```

```
            5170000321 +            SA7     SAVEW
                       63210        SB2     X1+B0
      4  67202                      SB2     B0-B2
            0100000000 X            RJ      =XGETBA
      5  0720000105 +               LT      B2,B0,NTDEF
                  5152000001        SA5     B2+1
      6  0325000106 +               PL      X5,NTTAPE
                  5132000002        UNLOAD
     35  7140000307 +               WAITER
     51  5132000002                 REWIND
    104  0400000000 +               EQ      UNLOADW
    105  7160000266 +       NTDEF   SX6     MSG3
                  0400000142 +      EQ      CONT.
    106                     NTTAPE  BSS     0
    106  5132000002                 REWIND
    141  7160000263 +               SX6     MSG2
    142  5150000247 +       CONT.   SA5     MSGC
                  12765             BX7     X6+X5
    143  5150000001                 CALLPP
    146  5150000321 +               SA5     SAVEW
                  63250             SB2     X5
    147  6130000005                 SB3     5B
    150  0632000151 +       +       GE      B3,B2,*+1
                  66200     -       SB2     B0
    151  0220000153 +       +       JP      JMP+B2
    152  0400000000 +               EQ      UNLOADW
    153  0400000161 +       JMP     EQ      EXIT1
    154  0400000167 +       +       EQ      EXIT2
    155  0400000201 +       +       EQ      EXIT3
    156  0400000213 +       +       EQ      EXIT4
    157  0400000234 +       +       EQ      EXIT5
    160  0400000000 +       +       EQ      UNLOADW
    161  7160000271 +       EXIT1   PMSG    MSG4
    166  0400000000 +               EQ      UNLOADW
    167  7160000274 +       EXIT2   PMSG    MSG5
    174  5150000251 +               SA5     ABTC
                  10755             CALLPP  5
    200  0000000000                 PS
    201  7160000300 +       EXIT3   PMSG    MSG6
    206  5150000250 +               SA5     ABTCS
                  10755             CALLPP  5
    212  0000000000                 PS
    213  7160000304 +       EXIT4   PMSG    MSG7
    220  5140000255 +               SA4     CNTCC
    221  7160000010         LOOP    SX6     10B
                  5160000256 +      SA6     CNTCCB
    222  10744                      CALLPP  4
    226  5150000070                 SA5     70B
                  0315000221 +      NZ      X5,LOOP
    227  5120000254 +               SA2     ENDC
                  10722             CALLPP  2
    233  0000000000                 PS
    234  7160000314 +       EXIT5   PMSG    MSG8
    241  5120000254 +               SA2     ENDC
                  10722             CALLPP  2
    245  0000000000                 PS
    322                             END
```