(2)

PL-TR-92-2262

A NEW DATA COLLECTION SYSTEM FOR IONOSPHERIC MODELLING AND RELATED TOPICS

Robert E. Sheehan

Boston College
Department of Physics
Chestnut Hill, MA 02167-3111

April 15, 1993

Final Report
November 7, 1988 - September 30, 1992

Approved for public release; distribution unlimited

PHILLIPS LABORATORY
Directorate of Geophysics
AIR FORCE MATERIEL COMMAND
HANSCOM AIR FORCE BASE, MA 01731-5000

"This technical report has been reviewed and is approved for publication"

EDWARD J. WEBER
Contract Manager

JOHN E. RASMUSSEN
Branch Chief

WILLIAM K. VICKERY
Division Director

This document has been reviewed by the ESD Public Affairs Office (PA) and is releasable to the National Technical Information Service (NTIS).

Qualified requestors may obtain additional copies from the Defense Technical Information Center. All others should apply to the National Technical Information Service.

If your address has changed, or if you wish to be removed from the mailing list, or if the addressee is no longer employed by your organization, please notify PL/TSI, Hanscom AFB, MA 01731-5000. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document requires that it be returned.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704 0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
April 15, 1993

3. REPORT TYPE AND DATES COVERED
Final Report Nov 7, 1988 - Sept 30, 1992

4. TITLE AND SUBTITLE
A New Data Collection System for Ionospheric Modelling and Related Topics

5. FUNDING NUMBERS
PE 62101F
PR 4643 TA 08 WU AR
Contract F19628-89-K-0001

6. AUTHOR(S)
Robert E. Sheehan

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Boston College
Department of Physics
Chestnut Hill, MA 02167-38111

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Phillips Laboratory
Hanscom AFB, MA 01731-5000

Contract Manager: Edward Weber/GPIS

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
PL-TR-92-2262

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Work performed under this contract can be divided into three areas, (1) development and testing of a PC based Scintillation Data Recording System (SDRS) and related analysis software, (2) updating and extending D. Anderson's global ionospheric model for studies concerning the generation and transport of polar cap patches, and (3) implementation of existing and new software utilities on the PC and SUN workstation in support of Phillips Laboratory.

Beginning in November 1990, SDRS has been installed at 5 field sites from Thule, Greenland to Ascension Island in the South Atlantic. Modified systems have also become TEC receiving stations for a tomography project and a remote access system designed to report scintillation activity at field sites. New software was developed to extract and display SDRS and TEC phase data on the PC. Two important additions were made to the Anderson model; the Hardy statistical model describes electron precipitation patterns, and production caused by precipitating electrons is based on a parameterized version of Strickland's flux

(over)

14. SUBJECT TERMS
rf scintillations
ionospheric tomography

15. NUMBER OF PAGES
100

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	SAR

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

13. ABSTRACT (continued)

transport model provided R. Daniel. An extension of the parameterization for energies typically observed in the dayside cusp (E0 $<$ 1 keV) has been written but not yet tested in the model. As the need arose, key analysis and planning tools were ported to the PC or SUN workstation in response to the shift away from mainframe computers, and special purpose software was written for CRRES and RODEO.

Table of Contents

I. Introduction

This report describes work on two completely different aspects of ionospheric modelling, data collection to drive and verify models, and numerical simulation of ionospheric conditions based on physical approximations. The original intent was to improve a one-dimensional computer program that calculates ionospheric chemistry, O+ diffusion, and thermal effects along a magnetic field line, with an emphasis on ion production by precipitating electrons (Strickland, 1976). Two and three-dimensional models can them be constructed by adding transport by neutral winds and convection electric fields. Such a framework already existed in an ongoing project inspired and supervised by D. Anderson (Phillips Laboratory). Because the Anderson model is already well documented, part 3 of this report discusses in detail only the precipitation effects which were added to the model. Dwight Decker and Cesar Valladares (Boston College), two major contributors to the model development, have worked on global convection, overall program control, and graphic displays.

Early in the contract period I assumed responsibility for the software development of a PC system designed to collect, process, and record radio scintillation data. Traditionally, the data were recorded at a number of field sites by multichannel strip chart recorders on paper or on reel to reel tape and sent back to the laboratory for analysis and archiving. It is much more efficient, however, to take advantage of rather inexpensive PC technology which can collect the data digitally, compute physical values and record the results on a floppy disk, so that much less material has to returned from the field. There is still the option to store a limited amount of raw, full wave form data on bulk medium such as a tape cartridge. The price of this flexibility is the time spent to write and test the software for a fairly large and complicated project.

Programming a real time system has problems and subtleties not encountered in standard scientific computing. For example, nearly all mainframe and PC programs run in a procedure oriented fashion where tasks, subroutines, or functions execute sequentially, uninterrupted except possibly by user input through the keyboard or mouse. The essence of real time program, however, is the ability to react at any time (asynchronously) to external events such as keyboard input or the collection of a data sample. Moreover, for a real time system on a PC, the programmer must be aware of hardware considerations like data storage efficiency, video speed, and interrupts. In particular, because most aspects of input-output (I/O) and hardware are controlled by the computer operating system, the capabilities and limitations of DOS, the standard PC operating system, require constant attention. A more thorough consideration of real time programming and DOS is given in Appendix 1.

After early difficulties, SDRS systems have been installed and are running at four remote field sites, Thule, Greenland (Nov 1990), Ascension Island (April 1991), Sondrestrom, Greenland (Sept 1991), and Svaalbard, Norway (Jan 1992). The stations at

Thule, Sondrestrom, and Svaalbard record one channel of scintillation data only; Ascension Island records three scintillation channels and two polarimeter channels. A station at Hanscom AFB has been used to test system development and program modifications. Currently, the original strip chart recorders run in parallel with the new SDRS systems. SDRS statistics on the floppy disks and printouts are selectively compared with strip chart records. So far only one significant bug has slipped past testing to the field systems, fortunately affecting only the multichannel site at Ascension Island. A correction in the data retrieval program can fix most of the errors caused by the mistake.

SDRS development also prompted several programs that playback and display data on the PC. PLOTSTA reads SDRS data saved on floppy disks and plots the S4 index, spectrum, and cumulative distribution function at 5 minute intervals. FWF displays the full wave form data, when available on tape cartridge, and gives the options to compute the spectrum for each 82 second sample (4096 points). FORMSTA reads SDRS floppy disks and creates files compatible with statistical programs used for scintillation studies. Lastly, ASCIDAT converts packed, full wave form data into text files suitable for spectral programs and other detailed analysis.

During SDRS development and testing two significant applications stemmed from the original concept. A proposed ionospheric tomography study took advantage of the generic data collecting capabilities of the program to record signal phase data from TRANSIT satellite beacons. Analysis programs were also written to extract and deramp desired sections of data for relative slant TEC information. Two of these modified SDRS systems were operated at Goose Bay, Labrador and Atlanta, Georgia in November 1991 along with several other standard TRANSIT receiver sites in a north-south chain. The second application is a remote access SDRS that can be called for a voice message giving current S4 conditions. In addition, a warning option permits the remote system to call any telephone and issue a voice message indicating disturbed S4 conditions or signal loss along the satellite link.

Material in the main body of this report focuses on the overall system logic and data handling. No programming knowledge is necessary to follow the text, but a general knowledge of computers and programming would be useful. Items specific to the C programming language are reserved for the appendix where the source code also appears.

The order of topics reflects a priority in terms of uniqueness. Since this effort is the only source of SDRS software development, it is important to document the system in detail. The Anderson model, on the other hand, is described elsewhere and only my limited contribution to it will be discussed.

2. SDRS Description

a. Logic Overview

SDRS is a PC based data collection and processing system designed to supplement conventional strip chart recordings of radio scintillations at field sites. Scintillations are disturbances in signals received from a satellite caused by density irregularities in the ionosphere. Fluctuations of signal strength and phase are often observed at UHF frequencies in polar regions and along the magnetic equator after sunset. Marked seasonal and solar cycle dependencies in their occurrence have been uncovered by analyzing many years of data, and part of the motivation for SDRS has been to gather more statistical data through the next solar cycle while eliminating tedious archiving and analysis of strip chart records. Besides collecting data at 50 samples/sec, SDRS can routinely calculate power spectra and auto-correlations in the field.

Specifications for a scintillation system based on a Zenith-248 AT clone were proposed in March 1988. An informative outline written by the first programmer, M. Keane of KEO Associates, describes the system philosophy and status of the project when he left (Appendix 1). At that time the program could handle most of the necessary tasks such as real time data collection, calculation of the required statistical parameters, and use of a fast fourier transform (fft) routine to get the power spectrum. Data was recorded to floppy and hard disk files, and a few commands could be executed through the keyboard. A clock function displayed the time and there was a basic but versatile set of routines for displaying text and plots as well as routing text to the printer.

The programming language is Microsoft C version 5.1 and the operating system is MSDOS version 3.30; although later versions of each have appeared, the environment has been fixed so that all system of this generation will be as consistent as possible to simplify maintenance. In addition to the standard C libraries there is a TimeSlicer library by Lifeboat Inc and driver library from System Guild for the Metrabyte analog to digital (A/D) expansion board. The Microsoft Quick C environment was used to edit the source code files. After editing and saving changes in the source code, a MAKE utility program compiles any modified source code and links in the libraries to produce the executable file (SDRSD.EXE).

SDRS itself is launched by typing SDRS at the DOS prompt (C:\SDRS>) and pressing the return key. This runs a batch file (SDRS.BAT) that automatically executes a series of DOS commands ending with the SDRS executable file. Before this a small program (RDDIR.EXE) checks to see if any full wave form files exist on the hard disk and prompts whether they should be saved or deleted.

After two setup screens for system configuration, SDRS runs initialization routines, allocates memory, and allows the operator to calibrate the receivers before starting data collection. Once SDRS begins data collection all processing is automatic; the operator must only perform calibration periodically and change the printer paper, floppy disks, and tape cartridges, if necessary. When power is restored after a power failure or a tape backup is completed, the system restarts automatically, bypasses the user prompts, and begins data collection immediately.

The relationship between the TimeSlicer and the SDRS application is shown schematically in Figure 1. In conventional programming there is a single starting point (PROGRAM unit in FORTRAN or BASIC, main() function in C) and the program runs in a linear fashion until at explicitly stops, encounters a fatal error, or hangs in an infinite loop, intentional or not. MSDOS loads the executable program, usually a .EXE file, into memory and turns over control of the CPU to the program, except for routine updating of the clock and possibly scanning the keyboard for the program abort sequence Cntrl-C. Most programs, however, implicitly rely on DOS to perform hardware tasks involving I/O like keyboard input, printing, and reading or writing disk files.

TimeSlicer becomes an intermediary between DOS and the SDRS program. The start of the SDRS application (ts_main()) spawns the individual tasks in Figure 1 and then ends, its role completed. The individual tasks are deliberately programmed to be in recurring loops so that they are available at all times as the TimeSlicer switches among the tasks in a round robin fashion. Each task has its own memory space to save function arguments and CPU registers; approximately every 1/18 second, the PC hardware clock tickrate, TimeSlicer interrupts the executing task, saves the stack and CPU registers, loads the stack and registers for the next task and starts it running at the point it was interrupted earlier. In this way no single task controls the CPU for a long time, giving the illusion that all of them are running simultaneously. Because of the way DOS works there are complications inherent in such task switching, a consideration that will come up later.

The TimeSlicer scheme implements preemptive multi-tasking, or multi-threading, in the sense that a task, with some exceptions, can be interrupted at any point. Non-preemptive multi-tasking, in contrast, requires each task to explicitly relinquish control and switch to another task. This is simpler conceptually than preemptive switching, but it becomes difficult to manage in practice, both in handling all possible situations as well as running smoothly. With the TimeSlicer controlling the use of the CPU, tasks need not be aware of each other, yet are able to share information and run in concert. Still, a task can deliberately give up control and switch out so that the next task can run, a valuable ability used extensively in the program.

Of the five tasks depicted in Figure 1, only the task labelled acquire and cld are switched exactly in the preemptive manner described above. Two tasks, clock and rtclock, are set to timers and executed at 1 sec and 15 min intervals, respectively. Once awakened these tasks are switched preemptively until there procedure loop is finished, where they wait to be restarted at the next timer alarm. Unpack buffers is triggered by a special condition known as an interrupt request which is sent when a full sample of A/D data is collected. An interrupt service routine unpacks the multiplexed channel data and deposits the data into individual memory locations, or buffers, for each channel.

Depending on the number of channels and operator activities, the principal tasks, acquire and cld are waiting most of the time for something to happen. Acquire, the data processing task, waits for a flag to be set indicating that data from each channel must be processed. Cld waits for a key to be pressed and decides what meaning the key has, if any, for performing some function such as updating the clock, changing a display, or checking storage. The calibration procedure, a particularly intensive user interaction function, is part of the cld task.

Data Transfer

Data sampling by the Metrabyte A/D board is a relatively unobtrusive function controlled by DMA (Dynamic Memory Access) data transfer, which frees the CPU to do other tasks. Figure 2 shows schematically how SDRS collects and stores data in memory. An analog voltage level representing the received signal strength is converted by the A/D board to a 12 bit digital value ranging from 0 (-10 v) to 4095 (+10 v). Generally, about 1/3 to 1/2 of the potential dynamic range corresponds to a 30 db range in signal strength. Every 20 msec (50 Hz) the A/D samples all requested channels and the DMA controller then transfers the values into one of the DMA buffers reserved in memory by the SDRS program. The only time lost is that needed to switch control of the data bus and transfer the data.

Figure 2 also shows how memory is allocated in the system. DOS and other PC system data occupy low memory and areas between 640 Kbyte and 1 Mbyte. All SDRS program code, associated data, along with the DMA buffers, must fit in the less than 600 Kbyte of remaining memory. Extended memory above 1 Mbyte cannot be used directly in the program, but data files can be stored there using a RAMDISK utility that allows ordinary file reads and writes at much faster memory to memory rates compared with memory to disk device. Although the storage is volatile (data disappears when the power is turned off) SDRS uses RAMDISK and extended memory to store temporary calibration and statistics data.

A schematic showing how data is managed in memory appears in Figure 2.3a. Two types of buffers are required to hold raw digital data, a DMA buffer containing multiplexed channel data from the A/D board and individual channel buffers with 4096 data points constituting an 82 sec sample 50 Hz data, the fundamental processing cycle in SDRS. The real time nature of the system also demands double buffering for the DMA and channel data. When one becomes filled it is made available for immediate processing and new incoming data is directed to the other buffer.

An additional complication arises because of the segmented addressing structure of the Intel CPU used in all PCs. The maximum size of a DMA buffer is 64 Kbyte and it must not straddle a 64 Kbyte segment boundary. Because of this restriction, the two DMA buffers are the first to be allocated, but if they were to be the same size as the channel data buffers only a maximum of four channels could be collected (4 Kbyte/buffer x 2 bytes/integer x 2 buffers/channel x 4 channels = 64 Kbyte). However, the DMA buffers are only temporary holding areas for data and are not directly involved in the actual data processing. This means that memory can be saved by filling and unpacking the DMA buffers four times during each processing cycle as shown in Figure 2.3b. Each DMA buffer then only requires 4 Kb/channel and 7 channels would need 28 Kb/buffer.

Once the data collection problems were solved it was necessary to design a user interface, both for running the program and calibrating the system to make the correspondence between digital values and physical power levels. A commercial data entry program library for text was available, but it was felt that menus on the graphics screen were preferable since they allowed the operator to monitor system activity while entering commands and performing calibrations. Additional software libraries also consumed memory already becoming more and more scarce. This led to the development of customized menus and a calibration procedure with the standard Microsoft C libraries for text and graphics. An important saving in complexity and memory was realized by noting that although data must be collected simultaneously for all channels, processing can proceed sequentially so that only a single set of temporary buffers is needed to hold temporary data as each channel is taken care of in turn. There remains the strict real time system requirement that all processing has to be completed before the next set of active data buffers is filled. For SDRS this means that all calculations and recording has to be finished in 82 sec, the time needed to fill the channel data buffers.

The core of the SDRS program is contained in the acquire task; this is the only one that will be discussed in detail here. Descriptions and flow charts for the other tasks appear in the appendix with the source code listing. While "INI" appears in the MONITOR window acquire sets aside memory for various arrays and data structures, calculates a Blackman-Harris window used in computing the power spectrum, creates RAMDISK files to hold temporary statistical data, and initializes the fft routine and plotting parameters. The operator then decides whether to calibrate the receivers or continue on with the current calibrations, during which "CAL" appears in the MONITOR window. Once initialization is finished acquire starts data collection and "AMP" or "POL" appears in the MONITOR window, depending on the channel configuration.

When unpack buffers signals that the channel data buffers are filled, "BUF" appears under all channels in the MONITOR window and acquire's processing loop executes channel by channel starting with channel 1. If all options are selected for a scintillation (AMP) channel, the processing cycle for a channel consists of the following (approximate times on a Z-248 indicated in parentheses):

(1) "PAK" appears in the SDRS MONITOR window and the 4096 A/D integer values in the channel data buffer are packed into 12 bits (6144 bytes total) and appended to an existing full wave form file on the hard disk (1 sec).

(2) The calibration is applied to the raw A/D values yielding the received signal power. The floating point calibrated values are stored in a 16 Kbyte temporary buffer.

(3) The 82 sec signal sample is plotted . All plots referred to below are only for a single selected channel during a processing cycle. The Display menu option allows the operator to choose which channel should be plotted.

(4) "STA" appears in the MONITOR window; the S4 and cumulative distribution function (cdf) are calculated and then accumulated and stored in a temporary file with previous samples in a recording cycle, described below (2 sec).

(5) "FFT" appears in the MONITOR window and the power spectrum is calculated, and then accumulated and stored in a temporary file. The power spectrum is plotted (8 sec).

(6) "COR" appears in the MONITOR window and the auto-correlation is calculated. The lag corresponding to a correlation of .5 is stored and the auto-correlation function is plotted (2 sec).

(7) "OK" appears in the MONITOR window under the channel just completed and the processing begins at (1) for the next scintillation channel. If the last 2 channels are polarimeter channels, the processing continues with (8) instead.

(8) "POL" appears in the monitor window and 1 sec averages at 0 sec and 41 sec are stored for each of the polarimeter channels (1 sec).

Unless this cycle is the fourth in a series "AMP" or "POL" reappears in the MONITOR window and acquire waits for the next processing cycle to begin.

After the fourth 82 sec processing cycle is completed, "REC" appears under each channel in the MONITOR window as the accumulated S4, cdf, spectrum, and auto-correlation values during the previous "5 minutes" (82 sec x 4 = 5 min 28 sec) are appended to the appropriate statistics (.STA) files on the floppy disk. A summary plot showing the 5 minute S4 and correlation times during the past 12 or 24 hours is updated and a record of the 5 min activity is printed out. Acquire then re-initializes all temporary buffers to zero and waits for the first processing cycle of the next recording cycle to begin. The time needed to record to the floppy disk is varies between 3 to 5 seconds per channel, depending on how long it takes to find available space. The crucial time constraint, then, is the longest time it could take to complete a processing cycle and record data to the floppy disk. Assuming 10 to 12 sec per channel to complete a processing cycle and 5 sec to record each channel, a likely maximum time to complete the entire process is 75 sec, leaving some time but coming uncomfortably near the 82 sec deadline in some situations. This illustrates how heavily SDRS could tax the capabilities of a 286 based PC, even with a numeric co-processor.

Calculation of Physical Quantities and Data Storage

As noted above SDRS records the S4 value, cumulative distribution function, power spectrum, and the time lag at which the auto-correlation function is 1/2. Although these quantities are calculated every 82 sec, the values are combined every 5 minutes and recorded on floppy disks. S4 is simply a normalized variance of the signal strength; a 5 minute value is determined by accumulating the square of the average power and the average of the power squared. To produce the cdf, digital values are binned and summed after every 82 sec processing cycle. At the 5 minute recording cycle the calibration is applied and the db levels at 11 percentile breakpoints are determined.

The power spectrum and auto-correlation are the most computationally intensive quantities that SDRS produces. First a Blackman-Harris window, a raised cosine function that emphasizes data in the center of the window, is applied to the 82 sec data record. This window provides a good trade off between frequency resolution and reduction of side lobes. After the ftt routine is done the real and imaginary parts of the frequency spectrum are squared and added to yield the power spectrum. The auto-correlation function is then obtained by running the power spectrum through the fft routine and keeping the real part.

Statistics data files for each channel are created when SDRS starts and periodic calibrations are run. A header containing channel and site information and the calibration points is written, after which the time and data are appended every 5 minutes. Internal binary representation for data is used to conserve storage space. All data files are closed immediately after being written so that data is saved even if a program crash or power failure occurs.

Full wave form files containing raw 50 Hz data are handled the same way as the statistics files except they are packed and written every 82 sec to the hard disk instead of the floppy. Full wave form data accumulates very fast; hence to keep them manageable new files are created when the current file reaches 1 Mbyte in size as well as when a calibration is performed. When the total amount of full wave form files stored exceeds 25 Mbyte, they can be backed up manually or automatically to a tape cartridge. Two backups are used to fill each 60 Mbyte capacity cartridge. An on screen storage function tells the operator the status of the floppy and hard disks along with the times that the disk and/or cartridge should be replaced. The operator informs SDRS that a new disk or cartridge is installed through the Disk and Tape menu functions.

2.b. Data Retrieval

Four PC programs for reading and displaying SDRS data have been developed. FORMSTA (Format Statistics) reads S4 and cdf data from a statistics file on a floppy disk and generates DOS text (ASCII) files in the same format used to archive scintillation data in the past. Three files are produced from each statistics file, a file with 15 minute values of S4 (the time interval currently in effect for archiving), a file containing 5 minute S4 values, and a file with 5 minute values of SI derived from the cdf. SI, the decibel difference between the 2% and 98% levels, has been used to estimate S4 from strip chart records based on comparisons with direct S4 calculations. The program has options to process single files or every file for a desired channel on the disk. A signal strength threshold can also be specified to eliminate S4 and SI values associated with weak reception; a typical threshold is 10 db above the noise floor. A cutoff is necessary because S4 tends to be overestimated by noise in the digital data at weak signal levels.

PLOTSTA (Plot Statistics) plots a time series of S4 values from a single statistics file along with the 5 minute cdf and spectrum. Figure 2.4 displays S4 data collected at Svaalbard, Norway on Jan 3-4, 1992. Scintillation activity was high throughout the period with a quieter interval from 2 to 9 UT on Jan 4. Extremely high activity (S4 => 1.0) like the burst just after 4 UT on Jan 4 should be suspect because it can be caused by interference or a weak but noisy signal. Individual 5 minute periods can be selected to show the spectrum or cdf alone, or both functions side by side. Figure 2.5 shows the cdf and spectrum over two consecutive 5 minute intervals when scintillation activity was relatively stable. The cdf displays a character typically seen with moderate to strong scintillations, an asymmetry about the median power level indicating deeper fades than enhancements. The spectrum shows a broad peak near .1 Hz and decreases in a power law manner to the noise floor approximately 4 orders of magnitude below the peak level.

FFT reads, unpacks, and plots each 82 sec sample of signal data in a full wave form file. Using the receiver calibration stored in the file header, the S4, median signal level, and spectrum are calculated and displayed. Figure 2.6 shows two 82 sec samples of full wave form data recorded at Hanscom AFB. The 50 Hz data in the left hand panels is plotted on a scale linearly inverse to AGC voltage. Signal strength increases toward the top; the nonlinear calibration is reflected in the 3 db tick marks along the vertical axis. The floor in the spectra seen at higher frequencies is an artifact of the routine that restricts the range and reduces the storage requirements. The spectra on the right are calculated by the same routine normally used by SDRS while collecting data; the only difference being that spectra taken in the field are combined and recorded every 5 minutes. Full wave form data allows a more detailed analysis over selected intervals back at the laboratory.

ASCIDAT converts selected sections of full wave files into ASCII files suitable for analysis on mainframes or workstations. Figures 2.7a-b were produced on different computer system at Phillips Laboratory (Cliff Bryant, Boston College) using the same section of full wave form data seen Figure 2.6 which was transferred through ASCIDAT. Independent S4 calculations by the two programs agreed (not shown), while the spectra agree in their overall character but not in detail. The scale difference results from different zero db reference levels.

c. Improvements and Prospects

As noted above the current version of SDRS uses the Z-248 PC nearly at its full potential. The main area that could be improved in the present software is error reporting, handling, and recovery, if possible. Although the field systems have been very reliable, every large program has bugs, and, whether caused by hardware problems or operator mistakes, there have been failures. There is some reporting for data collection and recording errors, but there is limited recovery ability short of re-booting the computer. A hardware solution to this problem is a watchdog timer board which monitors activity and restarts the system if it senses a failure. Even this would not help if critical files are corrupted by hardware or program errors unless a double buffering or fall back file scheme were devised.

Hardware upgrades are the most obvious ways to expand and improve the present system; running on a 25 Mhz 386 PC clone with math co-processor the same SDRS software can compute the statistics and spectrum in less than 1 sec per channel. This leaves much more time for additional processing and graphics. More esoteric features like a GPIB interface could control external equipment and perform automatic calibrations. A more pressing concern, however, is to replace the tape cartridge drives in the present systems with a removable medium such as Bernoulli disk drives made by Iomega. Tape backups of full wave form data in SDRS is extremely awkward; the system must stop, run an archiving program that can last 20 minutes, and restart. Because of hard disk space limitations this has to be done twice per 60 Mbyte tape cartridge, further complicating the program logic and potentially confusing for the operator. A Bernoulli drive, on the other hand, can be read from and written to as if it were another hard drive without stopping SDRS. Also, when tapes are returned from the field they must be read sequentially to restore files to a hard disk. This is time consuming even if only one file out of the 50 stored on a full tape is desired. Any file can be read or copied immediately from a Bernoulli disk.

A more fundamental software upgrade involves shifting to a Microsoft Windows or other graphical environment. Although Windows 3.1 does have multitasking, it is not preemptive in the sense of the TimeSlicer interrupting and switching tasks. There are provisions for timers, but it is not clear how they would be implemented in the present SDRS framework. OS/2 and the proposed Windows NT offer real preemptive multithreading. All these graphical environments are radically different than traditional DOS programming in that they provide standard dialog, graphics, and scheduling utilities which relieve the programmer of many duties and eases the task of modifying and upgrading software. The principal burdens are in learning a new, complex operating environment and following strict programming conventions. This path should be considered carefully and will only apply for a second generation system with modern PCs at all field sites.

3. Other SDRS Applications

a. Ionospheric Tomography System

In a related application SDRS was modified to record phase changes in TRANSIT satellite signals proportional to changes in total electron content (TEC) along the path from the satellite to the receiver. Simultaneous measurements by receivers in a longitudinal chain can be combined to produce a 2-D (latitude-altitude) tomographic reconstruction of electron densities. H. Kuenzler (Phillips Lab.) modified two JMR TRANSIT receivers to make them compatible with the SDRS inputs. Two identical channels 90 degrees apart in phase are necessary to resolve ambiguities in single channel measurements when the phase crosses 180 degrees. Two other AGC voltage channels indicating lock on the 150 Mhz carrier and 400 Mhz reference carrier were also recorded, making a total of four channels recorded in the full wave form mode. No field processing was required but a new display was developed to show the incoming phase and AGC signals in real time so that the operators could tell whether the receiver locked on a satellite expected to be in range of the site.

During the first field test in Nov 1991, sites with the modified SDRS systems at Goose Bay and Atlanta recorded several dozen passes in a four day study period. Once marginal passes were eliminated and data from three other conventional stations were examined, about 10-20 passes were deemed promising for further study. Relative slant TEC measurements from the individual stations are compared to yield absolute slant TEC. To get the slant TEC from the stored full wave form phase data, a program was written to display the phase and AGC data, and then to extract a pass for the deramping step. Figure 3.1 is a single pass from files recorded at Goose Bay on Nov 14, 1991. Two quadrature phase channels are shown as wrapped phase measurements from -180 to 180 degrees in a range of 0 to 5 volts. The phase is unwrapped by following one channel until it approaches +-180 degrees; the other channel is then picked up after taking into account the 90 degree phase difference. The resulting deramped phase is shown superimposed on the phase channel data. Signal lock is seen just after 1943 UT and the phase is successfully deramped until lock is lost at 2002 UT. Relative TEC is found by assuming a reference level at the beginning of the pass and applying a calibration relating change in phase to change in TEC. An ASCII file containing 1 sec slant TEC values is then written for comparison with other stations. Preliminary tomographic analysis of this and other passes are being done by J. Klobuchar, P. Fougere, W. Pakula (Phillips Lab.) and P. Doherty (Boston College).

b. Remote Access System

Scintillation activity at a site can determine how reliable a satellite link at that site is likely to be. Severe scintillations may degrade data transmission to the point where, lacking information about the potential ionospheric effects, maintenance personnel would be making fruitless attempts to locate hardware problems in ground receivers or in the satellite. With SDRS at a site monitoring the satellite to ground link, one more item of crucial information can help diagnose a communications problem.

One way to make this information available to an user anywhere in the world is to add a voice mail system that can be called by an ordinary telephone over a standard telephone line. Version [illegible] of a demonstration remote access system was shown at Offutt AFB in May and August, 1991, and at Peterson AFB in September. It consists of a single channel SDRS system that sends an S4 value every 5 minutes through the serial COM port to second PC (386 BiLink portable) with a voice mail expansion board installed and software written by P. Ning (KEO Associates). The second PC computes updated 15 minute, 1 hour, and 3 hour averages of S4 and has a menu system allowing the caller to access a voice message giving numerical or qualitative values of the current and recent scintillation activity at the site.

An additional feature is an option that initiates a call to any desired telephone and play a voice message concerning some condition observed by SDRS. Currently, the system is programmed to issue warning for active scintillations and loss of satellite signal. The active S4 threshold can be set by the operator at the SDRS site. Similarly, the threshold level indicating signal loss and the duration necessary to trigger an alarm can also be set. SDRS issues an alarm only at the start of a warning condition; a period of normal activity must occur before another alarm is sent. Two other conditions, abrupt change in signal level and signal interference, were also considered, careful study is needed to distinguish between valid signals and these warning situations. The next stage of development is to combine the SDRS and voice systems into a single portable PC and to conduct field tests at sites with a history of scintillation problems.

4. Anderson Global Ionospheric Model

A global ionospheric model currently under development at Phillips Laboratory requires many inputs for the calculation of a density profile along a single flux tube (Anderson et al., 1988). To follow many convecting flux tubes over long periods of time, inputs must be computed quickly, even on a fast mainframe computer or workstation. One critical input to the model is the production rate of O+ ions by precipitating electrons in the auroral oval, polar cap, and dayside cusp region. O+ is the dominant species at F region altitudes where convection is dominant. Older versions of the ionospheric model employed a simple Rees type calculation that assumed a continuous slowing down approximation. More exact models like that developed by D. Strickland are now available, but, although ideal for case studies, the routines are too slow for a global study where many repetitive calculations are required, and relatively large uncertainties in the inputs do not demand such detailed procedures. Fast, parameterized routines that summarize model results are more appropriate for the task.

The principal model improvement accomplished during this effort was the incorporation of the Hardy statistical precipitation model (Hardy et al., 1987) into the overall convection pattern. Instead of rings of precipitation, the Hardy model in Figures 4.1a and b gives a more realistic specification of the precipitation effects. Yet, it is a statistical distribution and it represents a smeared out version of the real, instantaneous pattern. Also, the average electron energy in Figure 4.1a is questionable in the polar cap and toward the dayside, where the radiation belts may contaminate the DMSP detectors with high energy counts. Even so, the flux pattern in Figure 4.1b indicates that the total production is not likely to be very large in these anomalous regions. A new production rate routine calculates O+ production rate profiles based on fits of Chapman functions to Strickland model results obtained with maxwellian input fluxes (Rob Daniel, Computational Physics Inc). The fast, parameterized routine, designed primarily for E region applications, is valid for characteristic electron energies > 500 eV, which, for maxwellians corresponds to average energies > 1 keV. In the nightside auroral zone region of Figure 4.1a the parameterized routine does a reasonably good job; where average energies fall below 1 keV, such as in the dayside cusp, a different approach, described below, should do better.

Figures 4.2 and 4.3 compare O+ production rate profiles at two neutral atmosphere temperatures derived from the full Strickland code (dashed) with profiles generated by the parameterized Strickland routine. The maxwellian with low energy tail (Figure 4.4) describes more closely soft electron flux distributions typically observed near the cusp. All results here are for isotropic fluxes over the downgoing hemisphere.

Although the fitted and full model profiles diverge somewhat above 200 km altitude, the fast routine is adequate for global modelling purposes over its range of characteristic energies given the likely accuracy of the precipitation inputs. The goal here is to show how to construct a comparably fast method for O+ rate profiles produced by softer precipitating electron fluxes.

The global ionospheric model has three inputs to the routine that calculates production by precipitating electrons: average electron energy, total integrated electron energy flux, and neutral atmosphere density profiles (N2, O2, and O). Since production rates are proportional to the energy flux for a fixed differential flux shape, the essential computation depends only on the average energy and the neutral atmosphere, which can be specified very nearly by the exospheric neutral temperature.

Figures 4.5 and 4.6 illustrate separately the dependence of O+ rate profiles on the characteristic energy E0 and the neutral atmosphere. All profiles were calculated by the full Strickland flux transport and production rate codes using maxwellian plus low energy tail fluxes as inputs. Figure 4.5 shows the expected increase in the altitude of maximum production as E0 decreases, while Figure 4.6 exhibits effects of increased scale height as exospheric temperature increases.

The data in Figures 4.5 and 4.6 could be interpolated to get a profile at any characteristic energy and exospheric temperature in the ranges shown; however, much of the temperature dependence can be eliminated by plotting the profiles in Figure 4.6 against total integrated neutral column density instead of altitude (Figure 4.7, the column density at a given altitude is the number of neutrals in a column 1 cm2 extending from that altitude to infinity). Because the relation between altitude and column density is already available in the ionospheric model, it is a simple matter to convert the column density profiles in Figure 4.7 back to altitude profiles.

Finally, Figure 4.8 shows rate profiles corresponding to Figure 4.7 over a range of E0 from 150 to 700 ev. Each profile was computed for an exospheric temperature of 1000 K but they closely approximate profiles over a 500 K range of exospheric temperature. This set of profiles and similar sets for generic flux spectra will provide the data base for obtaining production rate profiles generated by precipitating electron fluxes. This data base will be integrated into the global model and tested in a future model development.

5. Miscellaneous Utility Software

Several programs used to analyze data and plan experiments were either written or modified to run on different machines. This was particularly important because of the shift away from mainframe computers like the Cyber to workstations and PCs. Cyber FORTRAN code for the ubiquitous LOKANGLE program was altered to compile and run using Microsoft FORTRAN v. 4.0 on a Zenith-248 PC (8 Mhz, 286 CPU). LOKANGLE takes satellite ephemeris information and computes the orbit and look angles from a ground station. Since this change, a revised version of LOOKANGLE for the PC was published by Hein et al. (1991). Three other major programs, OBSWIND (observation window based on sun and/or moon visibility criteria), BMAGTRC (magnetic field line trace up or down from a starting altitude), and CGTRANS (geographic to corrected geomagnetic/local time coordinate transformation) were also ported either to the PC or SUN workstation.

Part of the CRRES mission in July, 1992 had the Airborne Ionospheric Observatory in position to record phase and amplitude disturbances caused by an ionized cloud released from a rocket at 250 km altitude off the north coast of Puerto Rico. It required the aircraft to be in position that placed the cloud in the raypath from a geostationary satellite beacon. A geometry program was written to take LOKANGLE results and the cloud location to give the cloud's "shadow" location at the aircraft altitude. The aircraft also carried a modified SDRS system to record full wave form records of the amplitude scintillations.

Another small utility program for ground based optics stations was written to give the look angles to the cloud based on the cloud location and altitude.

Finally, programs were developed to read Sondrestrom incoherent scatter radar tape files in support of several RODEO experiments. Tape records were copied to MSDOS files and then decoded using the format information provided by M. McCready. The relevent velocity, density, and temperature data was read and formatted for additional analysis.

References

Hardy, D.A., M.S. Gussenhoven, R.R. Raistrick, and W.J. McNeil, Statistical and functional representations of the pattern of auroral energy flux, number flux, and conductivity, J. Geophys. Res., 92, 12275, 1987.

Hein, C.A., K.H. Bhavnani, and E.C. Robinson, LOKANGL PC user's guide, Phillips Laboratory Technical Memorandum No. 191, 1991.

Strickland, D.J., D.L. Book, T.P. Coffey, and J.A. Fedder, Transport equation techniques for the deposition of auroral electrons, J. Geophys. Res., 81, 2755, 1976.

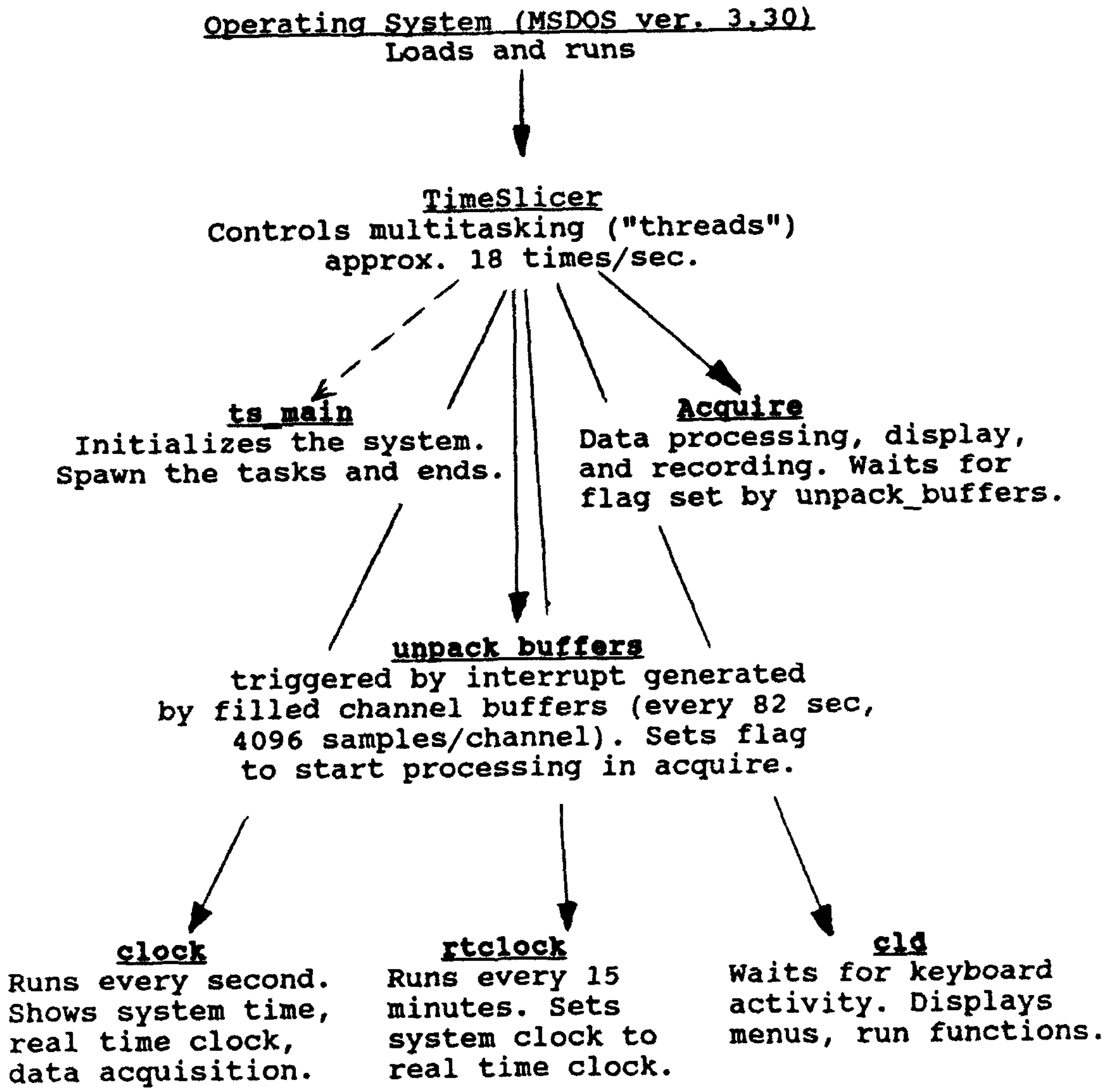

Operating System (MSDOS ver. 3.30)
Loads and runs
TimeSlicer
Controls multitasking ("threads")
approx. 18 times/sec.
ts_main
Initializes the system.
Spawn the tasks and ends.
Acquire
Data processing, display,
and recording. Waits for
flag set by unpack_buffers.
unpack buffers
triggered by interrupt generated
by filled channel buffers (every 82 sec,
4096 samples/channel). Sets flag
to start processing in acquire.
clock
Runs every second.
Shows system time,
real time clock,
data acquisition.
rtclock
Runs every 15
minutes. Sets
system clock to
real time clock.
cld
Waits for keyboard
activity. Displays
menus, run functions.

Figure 1

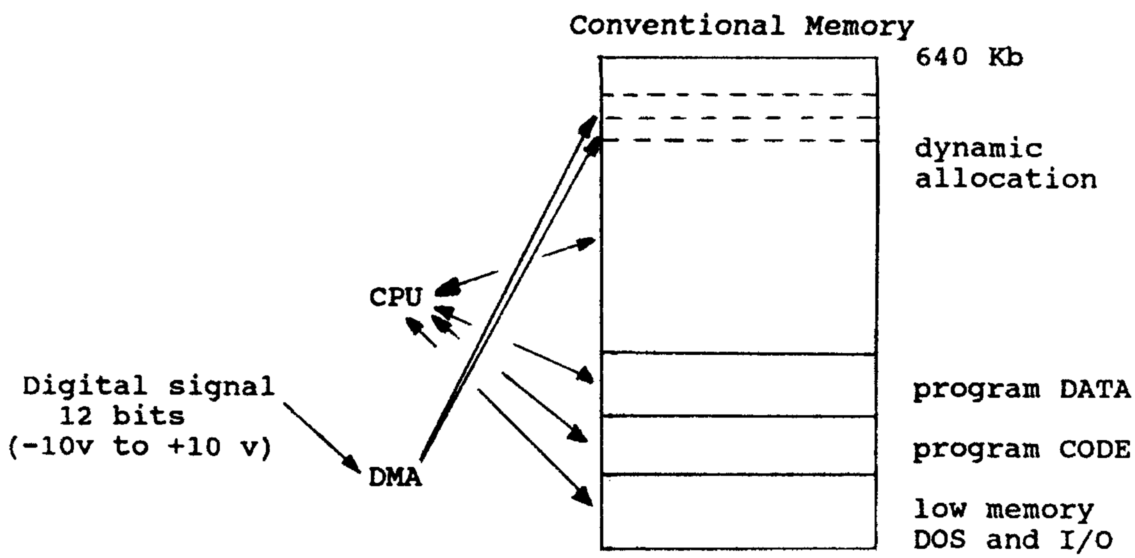

Conventional Memory
640 Kb
dynamic
allocation
CPU
Digital signal
12 bits
(-10v to +10 v)
DMA
program DATA
program CODE
low memory
DOS and I/O

Figure 2

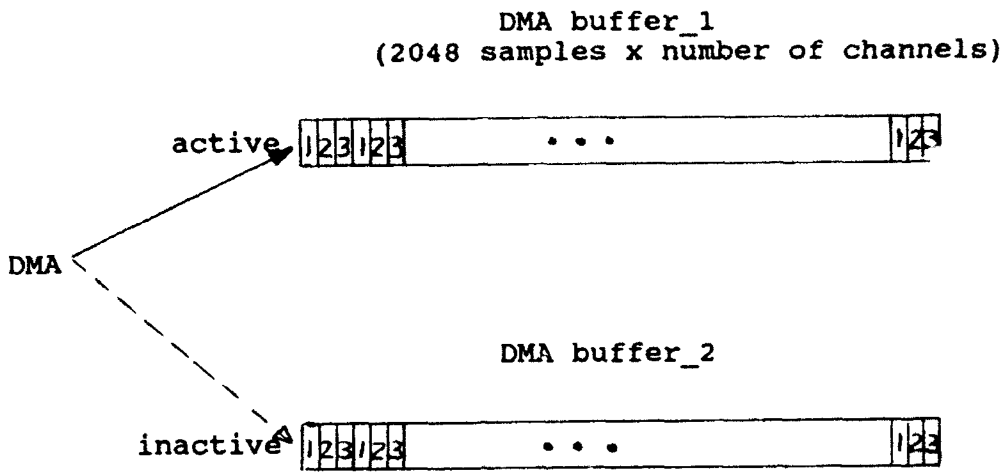


Figure 3a

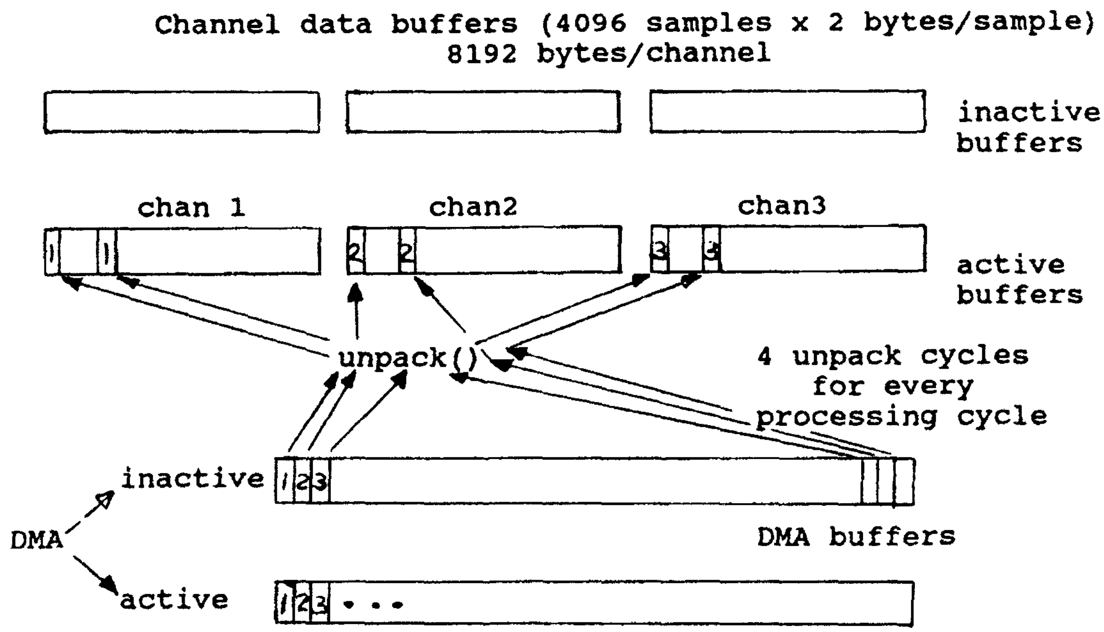


Figure 3b

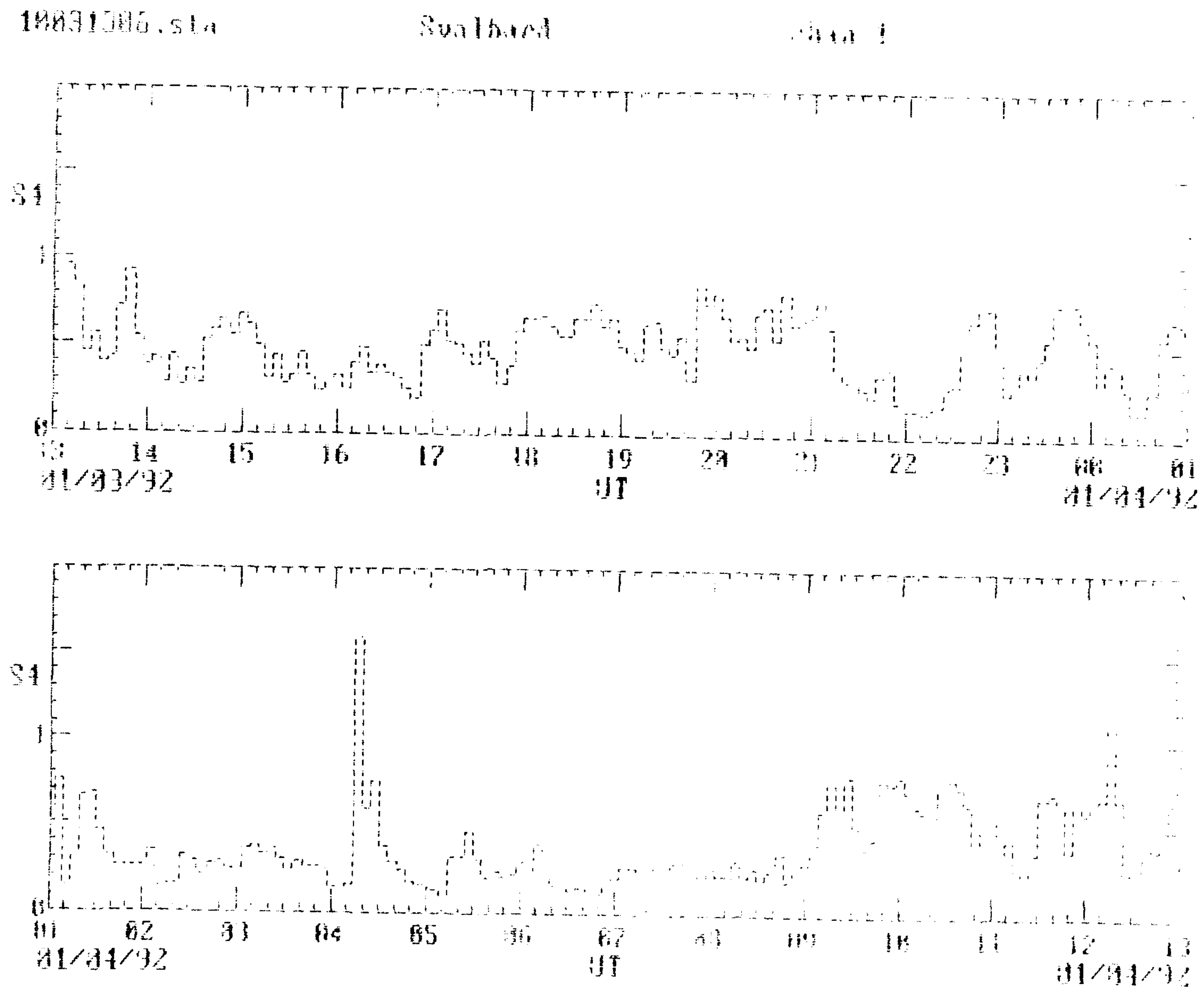
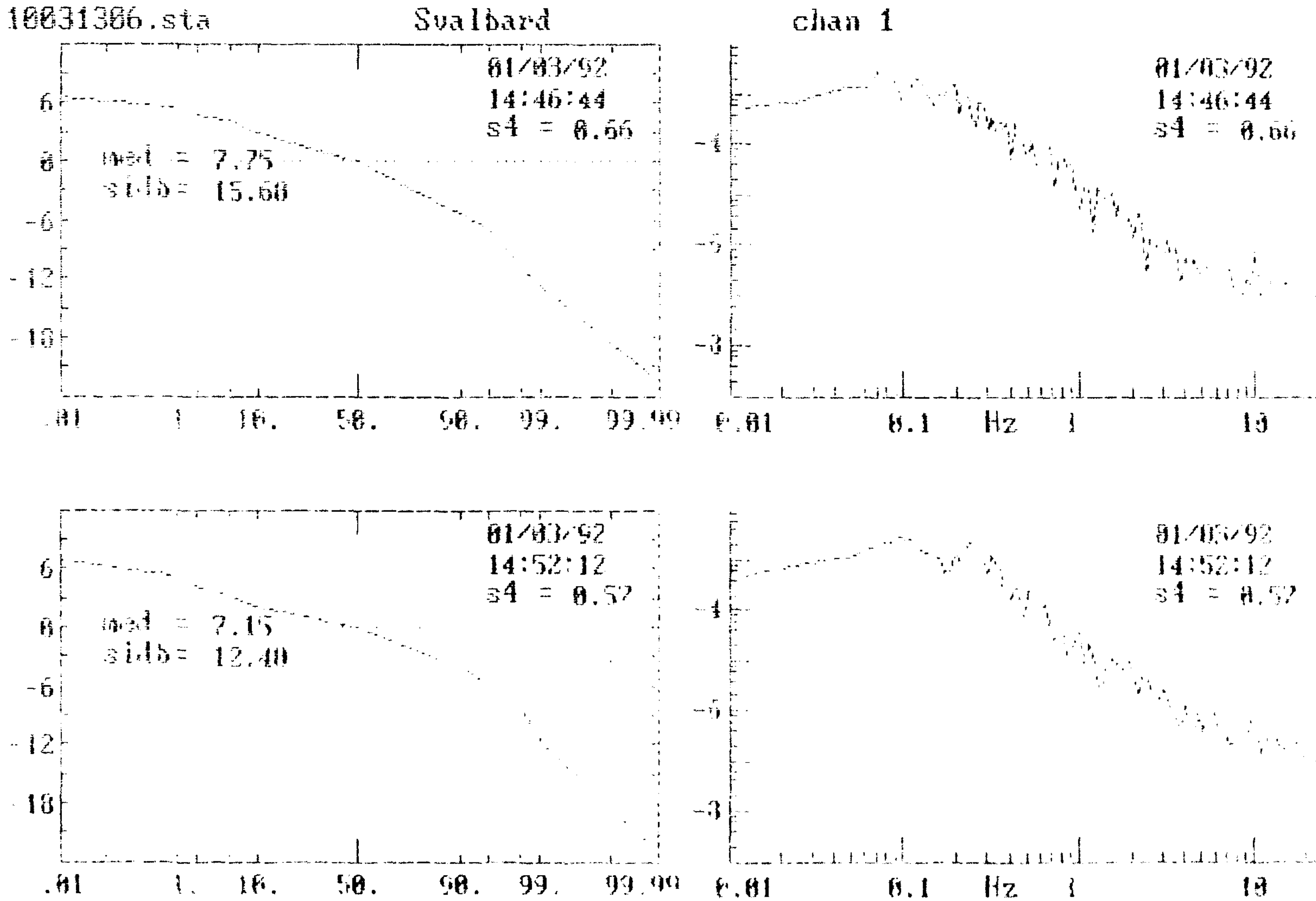

Svalbard
S4
1
0
14
15
16
17
19
20
22
23
01
01/03/92
UT
01/04/92
S4
1
0
02
03
04
05
07
10
12
13
01/04/92
UT
01/04/92

Figure 4



10031306.sta
Svalbard
chan 1
01/03/92
14:46:44
s4 = 0.66
med = 7.75
sidb= 15.60
01/03/92
14:52:12
s4 = 0.57
med = 7.15
sidb= 12.40
.01
1
16.
50.
90.
99.
99.99
0.01
0.1
Hz
1
10

Figure 5

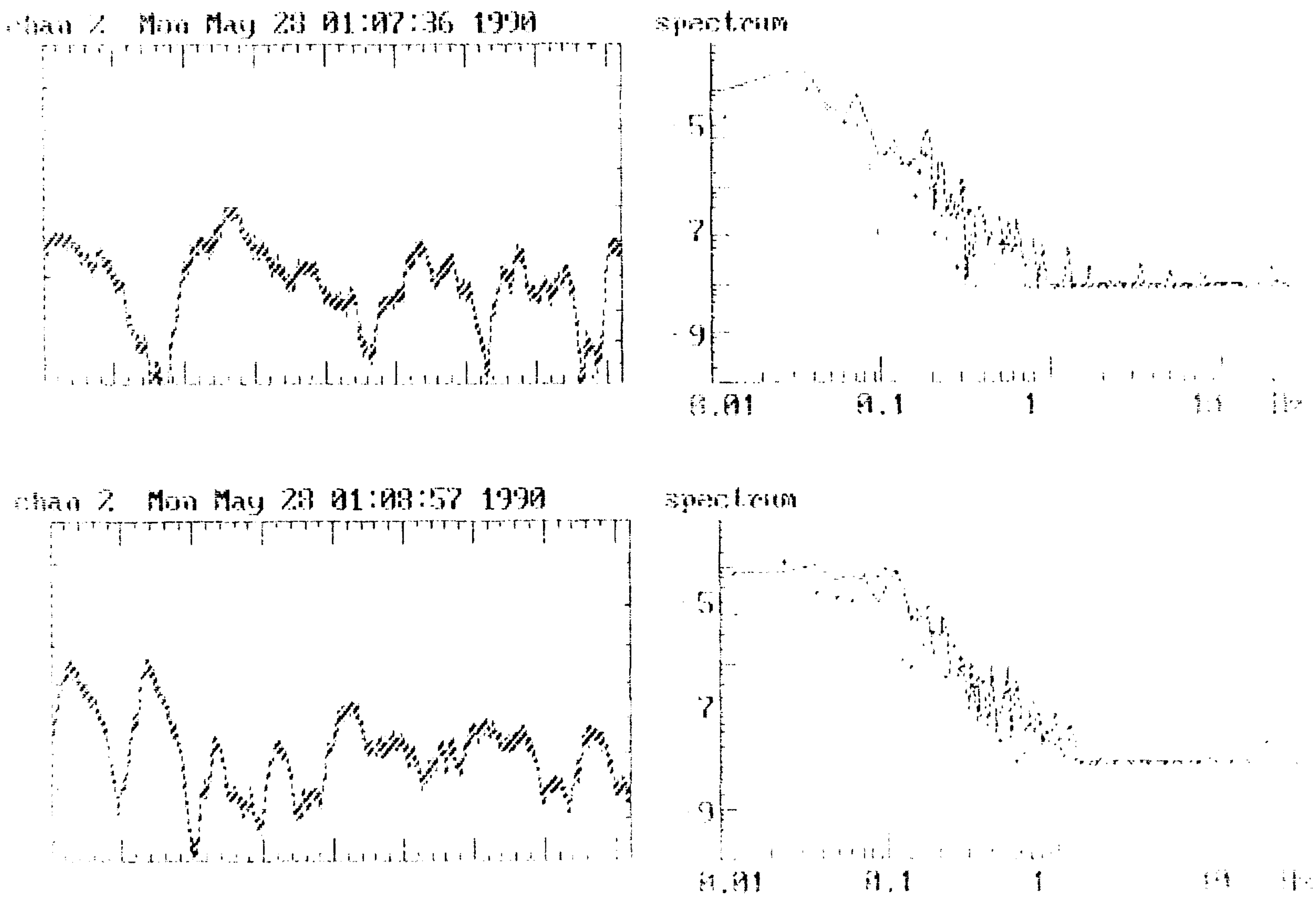

chan 2 Mon May 28 01:07:36 1990
spectrum
5
7
9
0.01
0.1
1
10
chan 2 Mon May 28 01:08:57 1990
spectrum
5
7
9
0.01
0.1
1

Figure 6

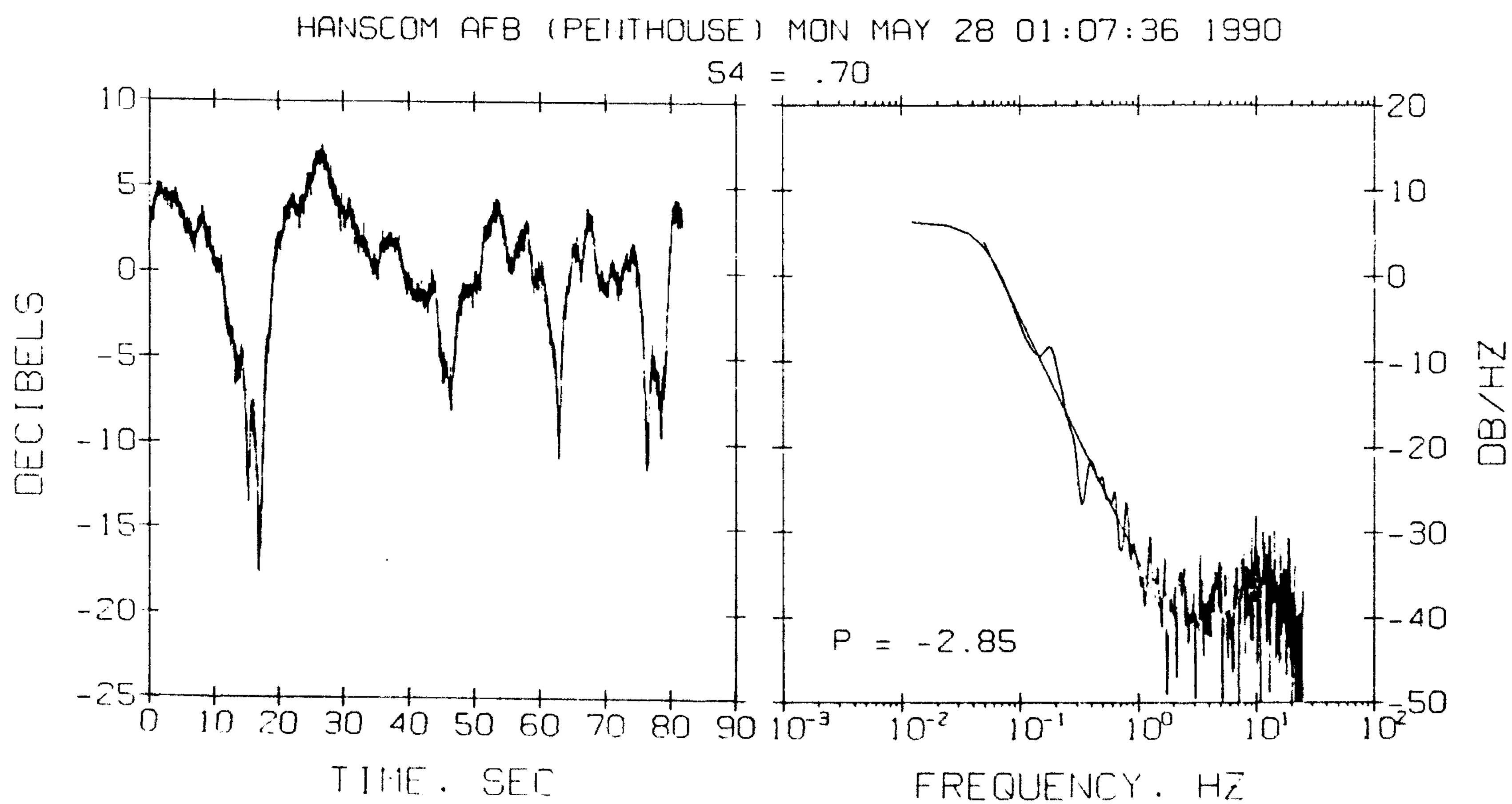

HANSCOM AFB (PENTHOUSE) MON MAY 28 01:07:36 1990
S4 = .70
DECIBELS
TIME. SEC
P = -2.85
DB/HZ
FREQUENCY. HZ

Figure 7a

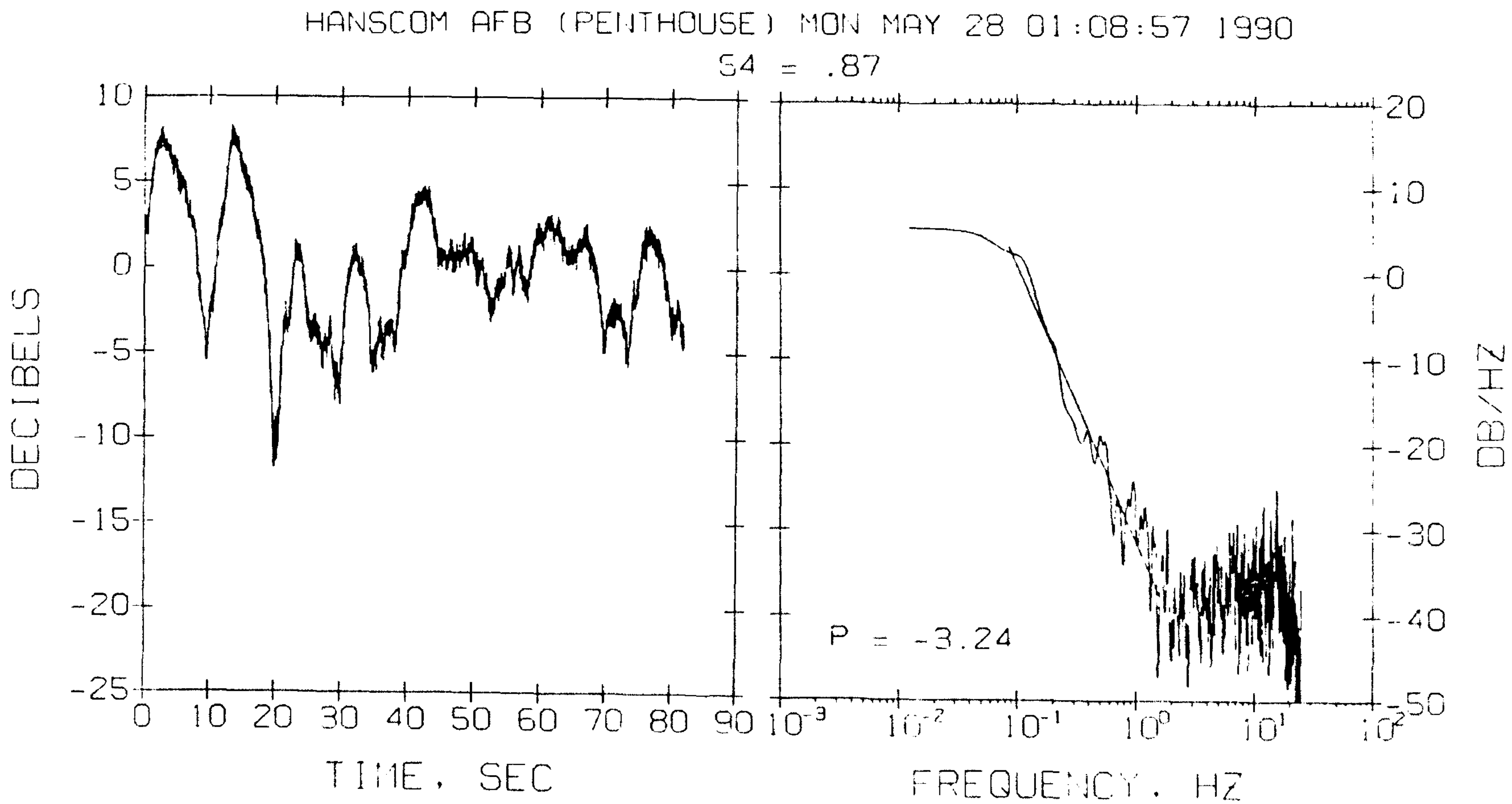
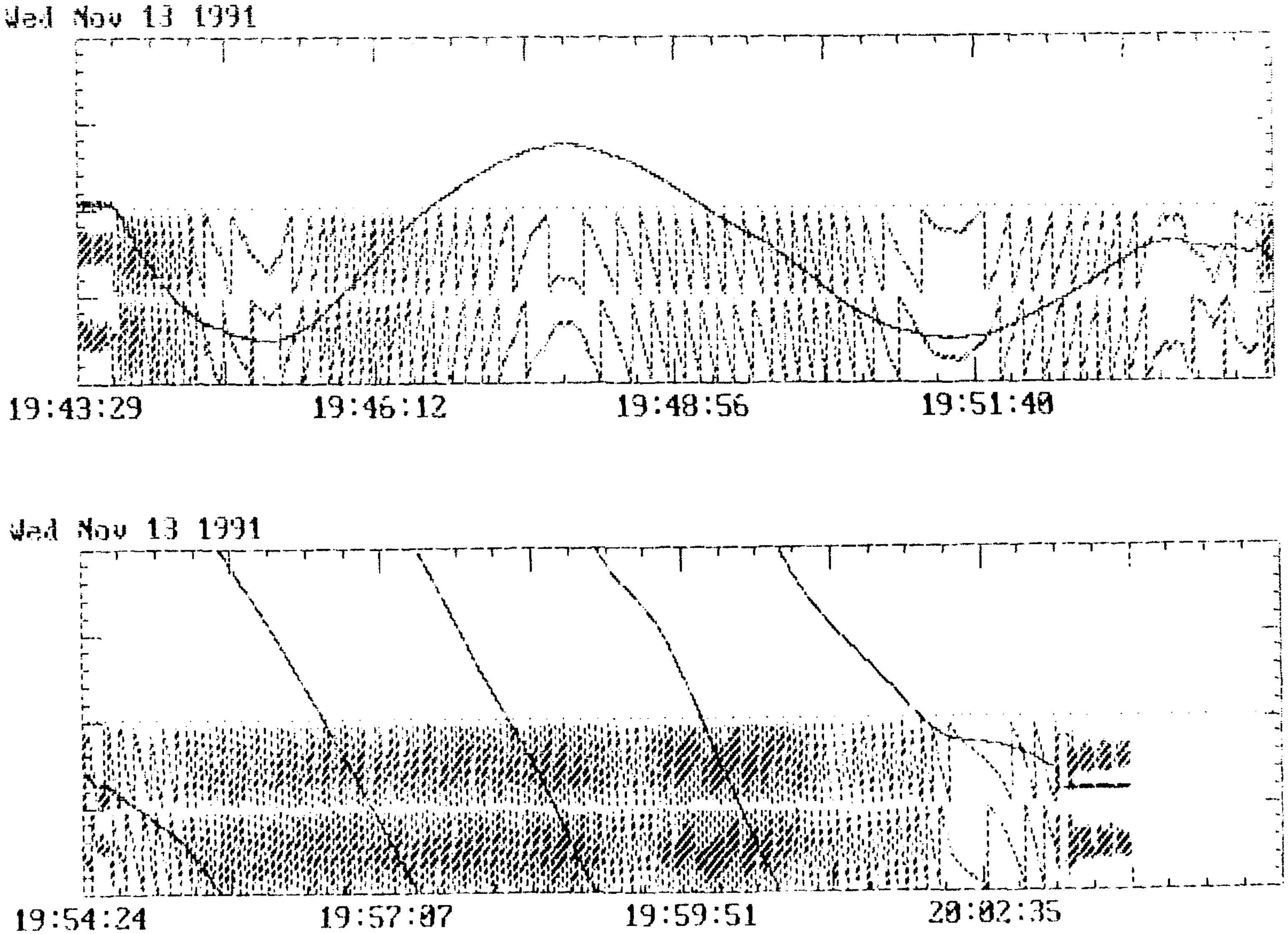

HANSCOM AFB (PENTHOUSE) MON MAY 28 01:08:57 1990
S4 = .87
DECIBELS
TIME, SEC
P = -3.24
DB/HZ
FREQUENCY, HZ

Figure 7b



Wed Nov 13 1991
19:43:29
19:46:12
19:48:56
19:51:40
Wed Nov 13 1991
19:54:24
19:57:07
19:59:51
20:02:35

Figure 8

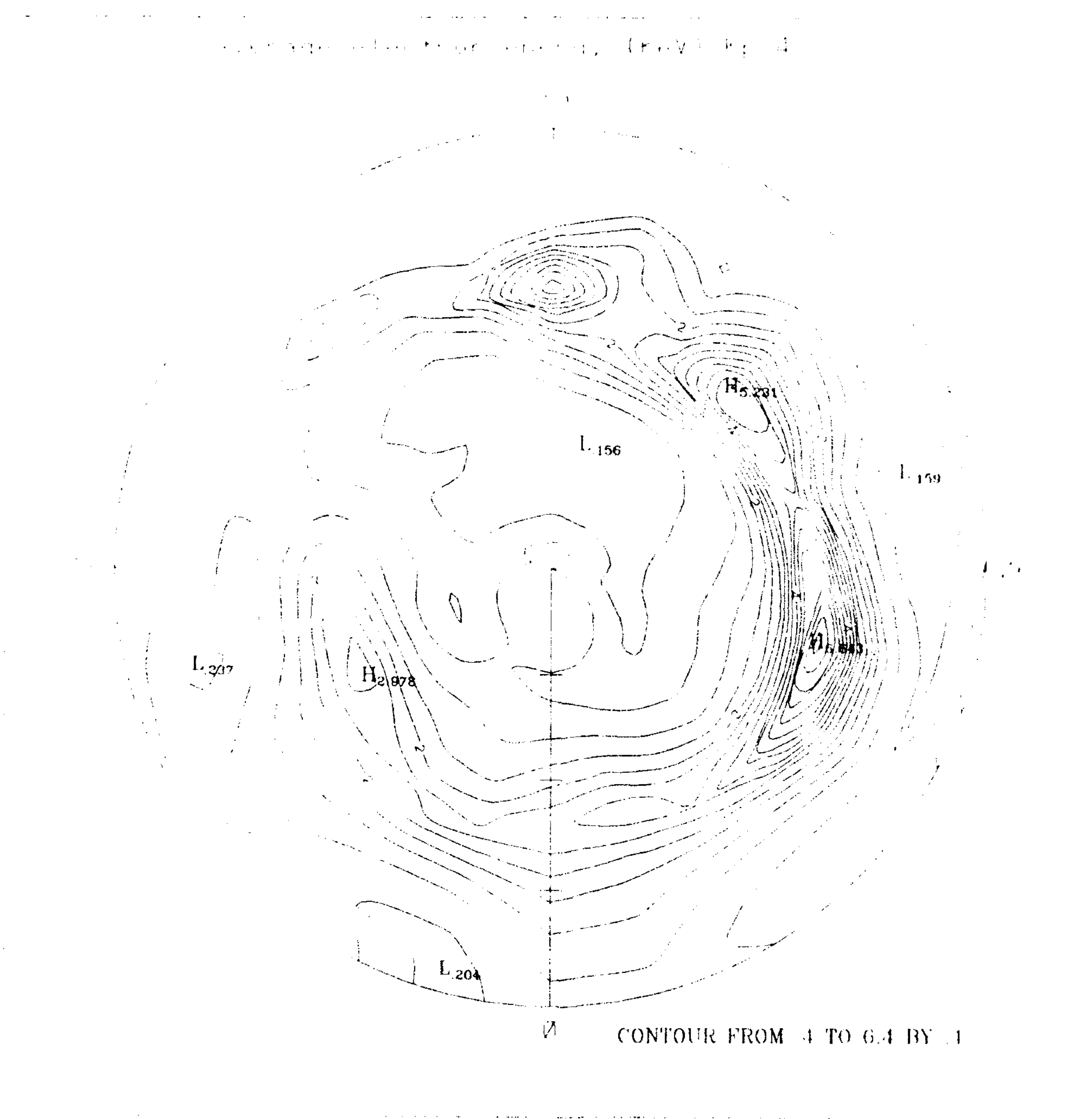


Figure 9a

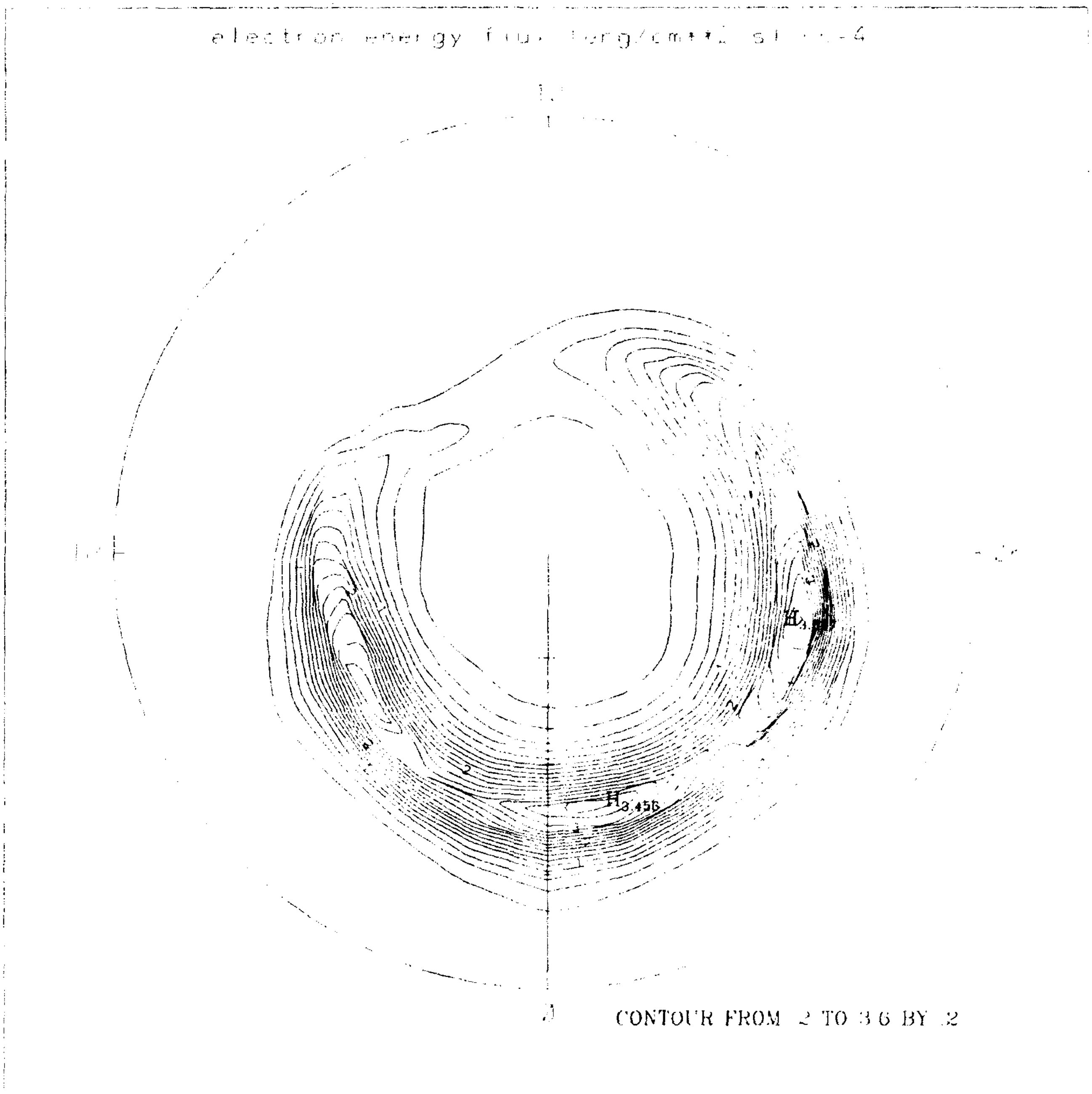


Figure 9b

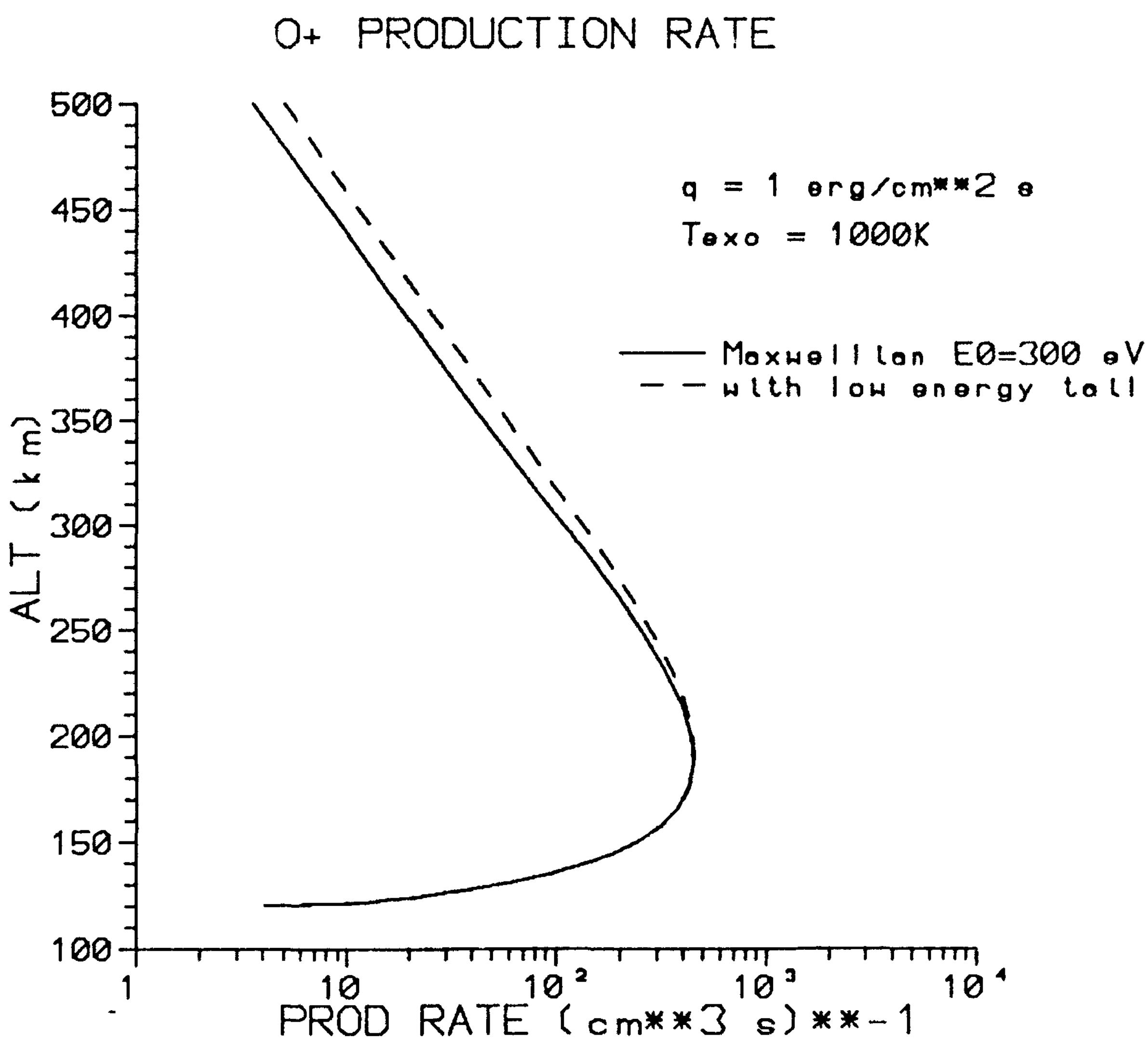

O+ PRODUCTION RATE
q = 1 erg/cm**2 s
Texo = 1000K
Maxwellian E0=300 eV
with low energy tail
ALT (km)
500
450
400
350
300
250
200
150
100
1
10
10²
10³
10⁴
PROD RATE (cm**3 s)**-1

Figure 10

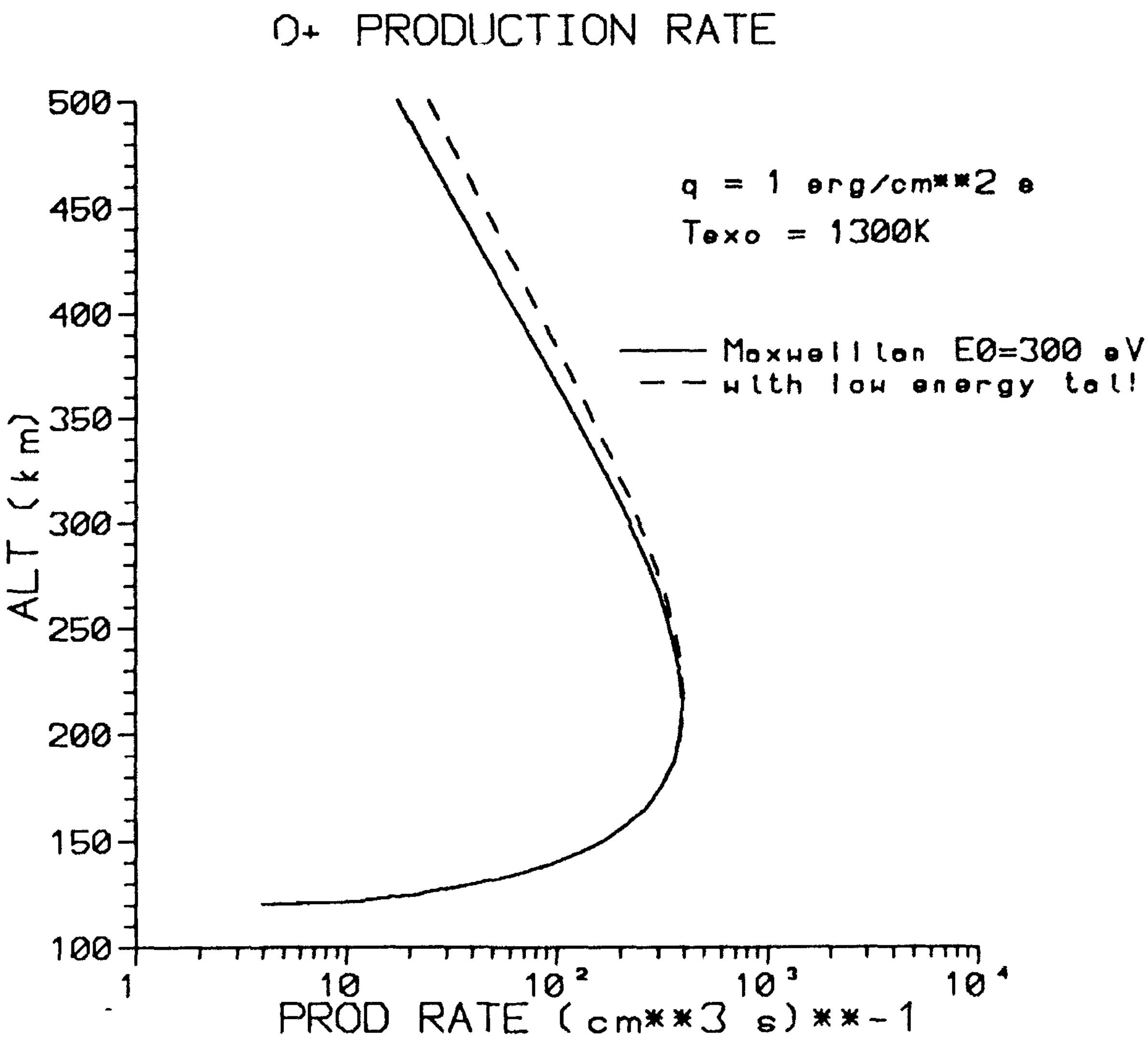

O+ PRODUCTION RATE
q = 1 erg/cm**2 s
Texo = 1300K
Maxwellian E0=300 eV
with low energy tail
ALT (km)
500
450
400
350
300
250
200
150
100
1
10
10²
10³
10⁴
PROD RATE (cm**3 s)**-1

Figure 11

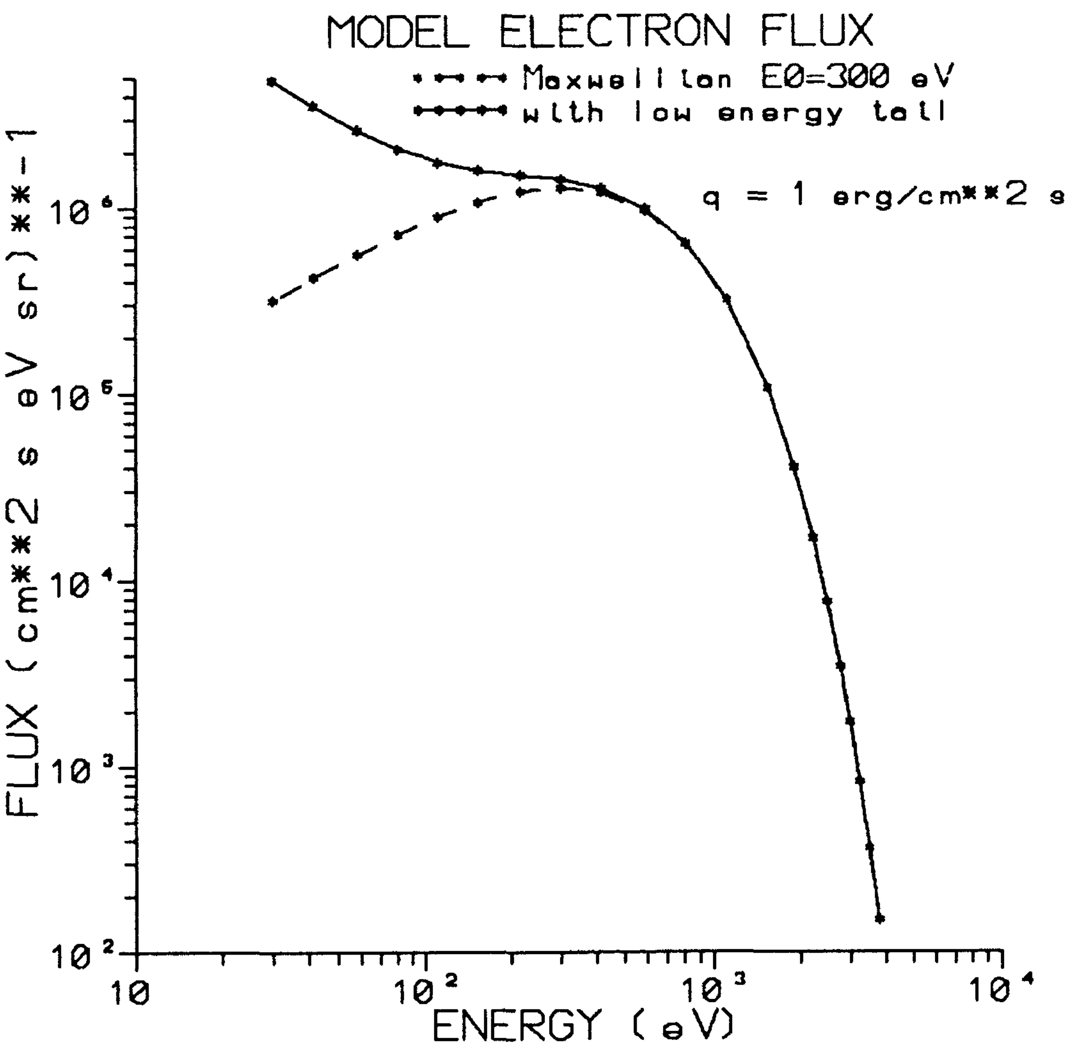

MODEL ELECTRON FLUX
Maxwellian E0=300 eV
with low energy tail
q = 1 erg/cm**2 s
FLUX (cm**2 s eV sr)**-1
10⁶
10⁵
10⁴
10³
10²
10
10²
10³
10⁴
ENERGY (eV)

Figure 12

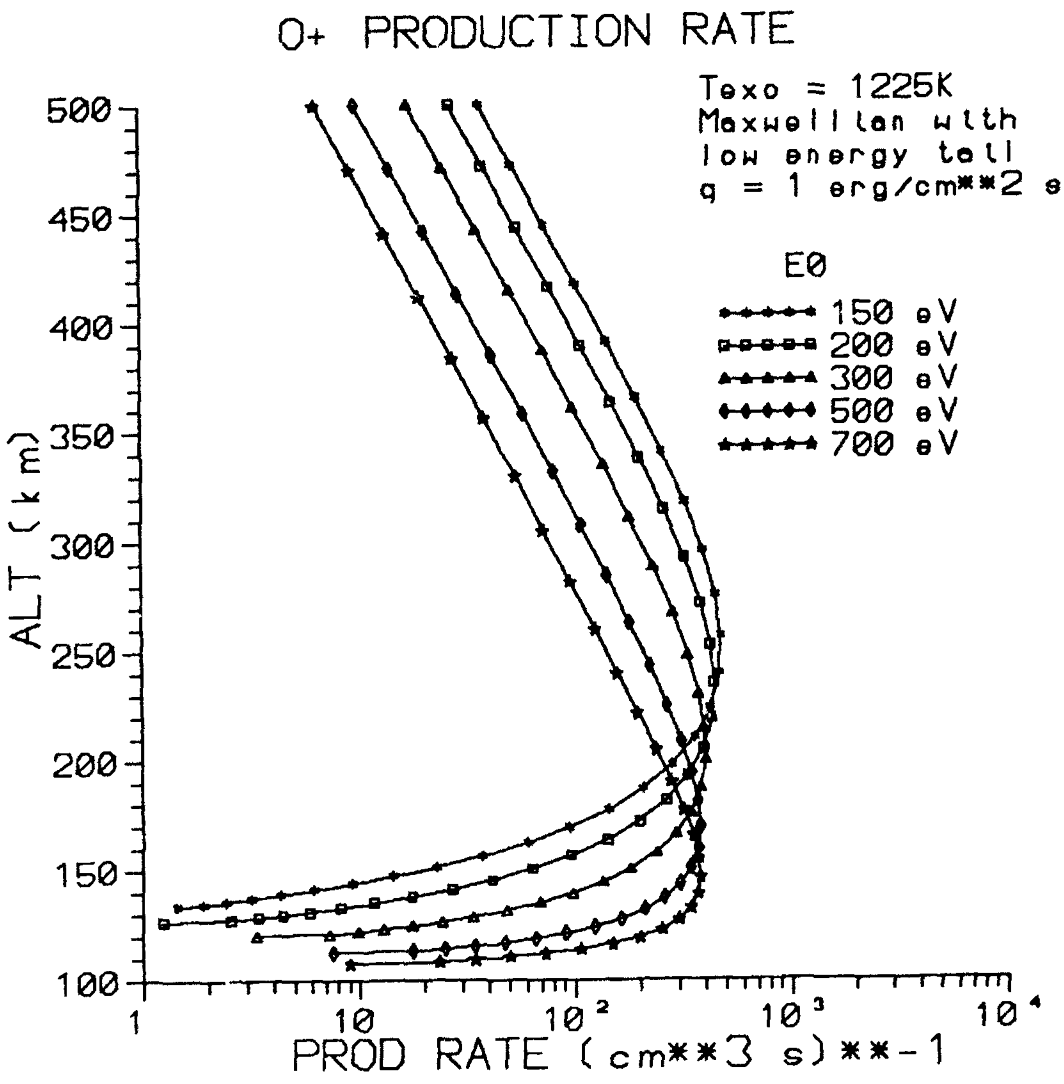

O+ PRODUCTION RATE
Texo = 1225K
Maxwellian with
low energy tail
q = 1 erg/cm**2 s
E0
150 eV
200 eV
300 eV
500 eV
700 eV
ALT (km)
500
450
400
350
300
250
200
150
100
1
10
10 2
10 3
10 4
PROD RATE (cm**3 s)**-1

Figure 13

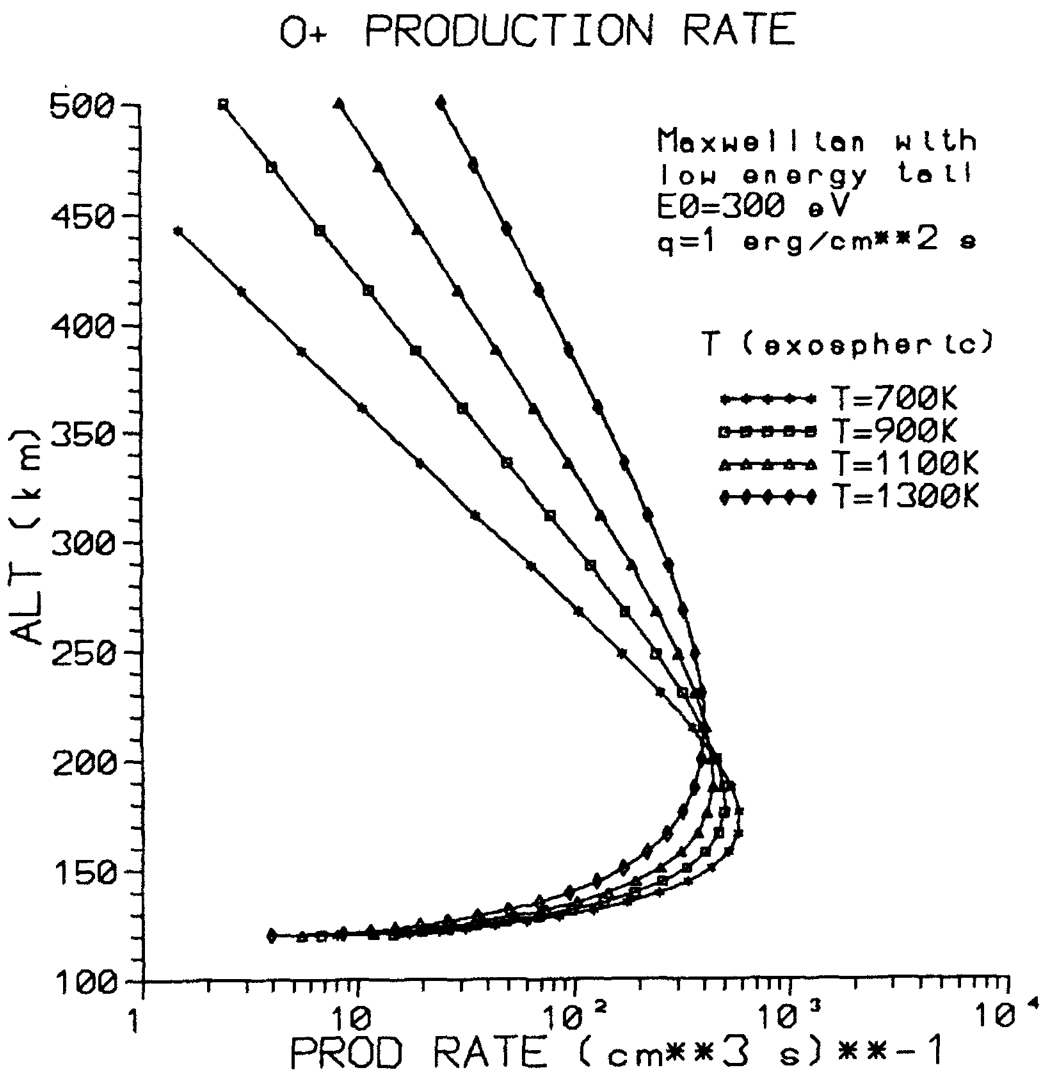
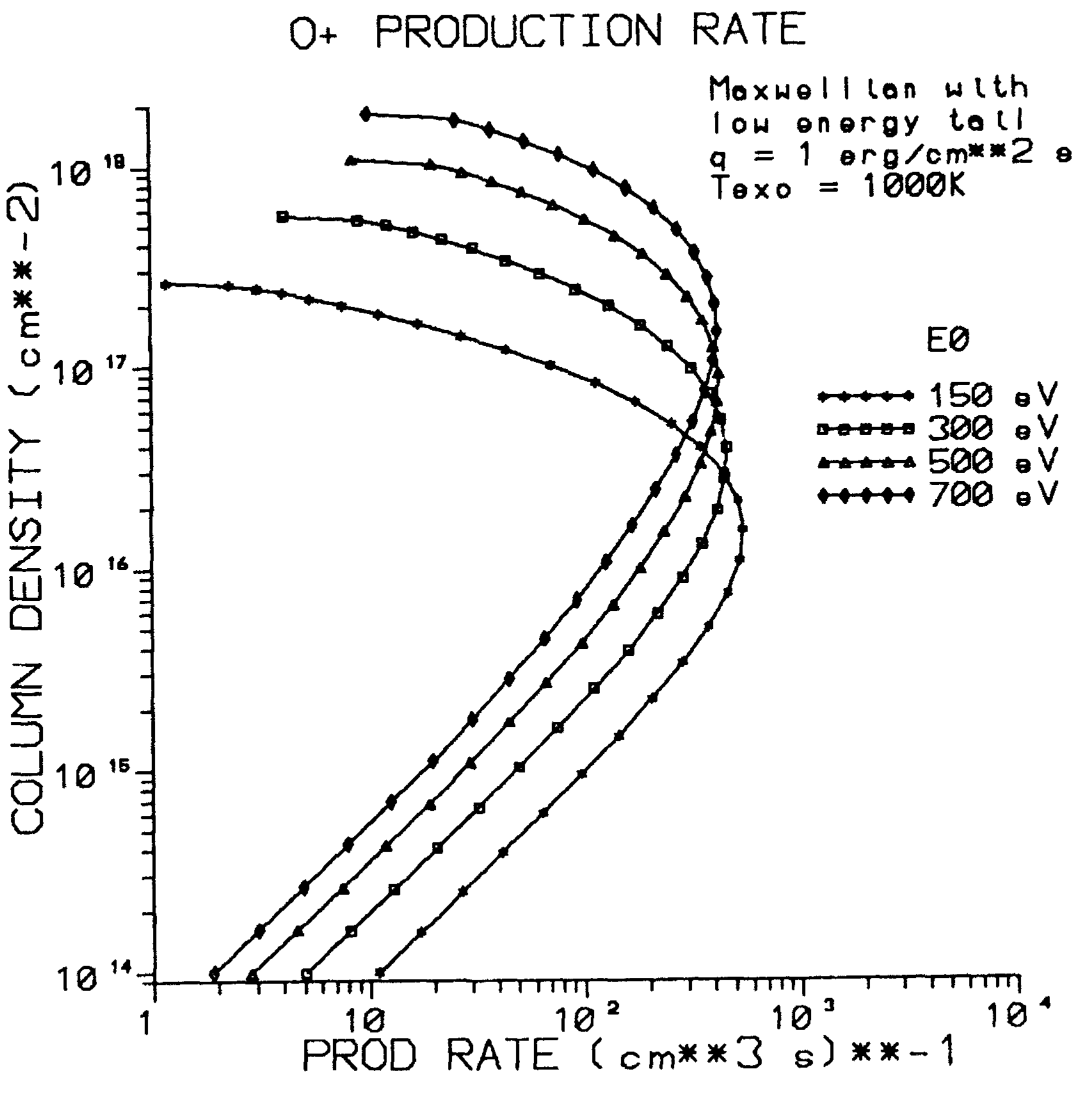

O+ PRODUCTION RATE
Maxwellian with
low energy tail
E0=300 eV
q=1 erg/cm**2 s
T (exospheric)
T=700K
T=900K
T=1100K
T=1300K
ALT (km)
500
450
400
350
300
250
200
150
100
1
10
10²
10³
10⁴
PROD RATE (cm**3 s)**-1

Figure 14



O+ PRODUCTION RATE
Maxwellian with
low energy tail
q = 1 erg/cm**2 s
Texo = 1000K
E0
150 eV
300 eV
500 eV
700 eV
COLUMN DENSITY (cm**-2)
10 18
10 17
10 16
10 15
10 14
1
10
10 2
10 3
10 4
PROD RATE (cm**3 s)**-1

Figure 15

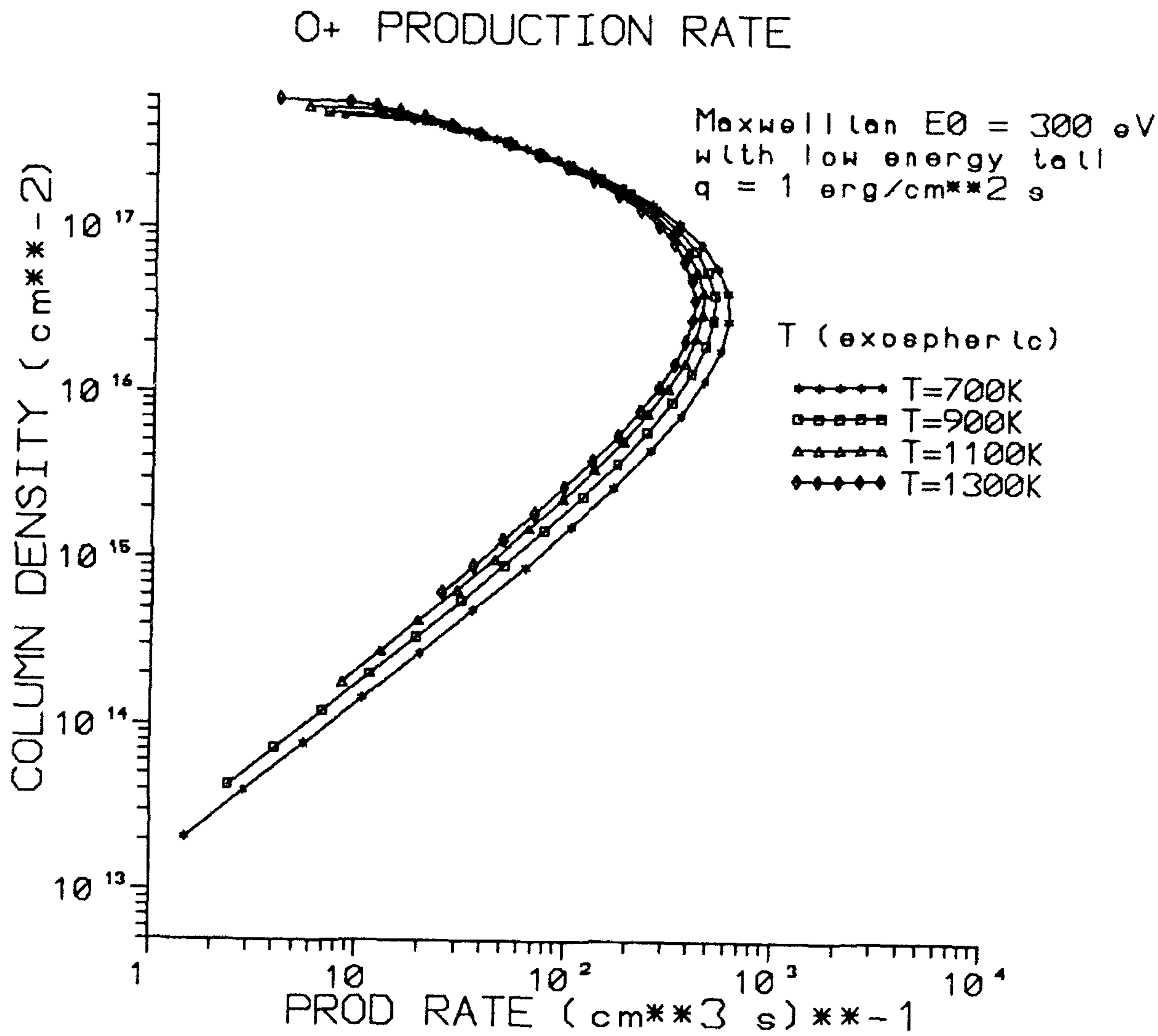

O+ PRODUCTION RATE
Maxwellian E0 = 300 eV
with low energy tail
q = 1 erg/cm**2 s
T (exospheric)
T=700K
T=900K
T=1100K
T=1300K
COLUMN DENSITY (cm**-2)
10 17
10 16
10 15
10 14
10 13
1
10
10 2
10 3
10 4
PROD RATE (cm**3 s)**-1

Figure 16

Appendix 1. SDRS Overview (M. Keane, edited by R. Sheehan)

The principal objective of this project is to implement a system to replace multi-channel strip chart recorders currently used to record scintillation data at remote field sites. In addition to acting simply as a direct replacement for these recorders, it was decided that a microcomputer based system could make significant improvement by making use of the ability to process the data as well as to record the data. Online processing of the data stream in real time eliminates the need to transport and archive a large volume of raw data. Only the much smaller volume of derived statistical quantities need be stored for return and analysis.

The hardware platform was determined to be a Zenith Z-248, an 8 Mhz AT clone. The design requirements as they have evolved:

i. - Operate at 50 Hz sampling rate with 12 bits of digitization.
ii. - A maximum of 7 scintillation channels.
iii. - Provide the following statistics.
   a. S4 (Normalized variance).
   b. The cumulative distribution function (sampled at ~10 points).
   c. Power spectrum from several mHz to 25 Hz (sampled at ~50 points).
   d. Decorrelation time (estimated via the mean zero crossing interval).
iv. - Capability for full waveform recording.
v. - Maximum autonomous operation with a minimal need for interaction with an operator.
vi. - One channel of polarization acquisition and processing.

2. Choice of Operating System

The Operation of a data acquisition and processing system would ideally take place under a real-time, multitasking operating environment. While the benefits of such environment for data acquisition and control systems are well understood in the minicomputer world, they seem to have been lost on those reared in the microcomputer culture. Let me first provide a brief introduction to real-time and multitasking and why they apply to the project at hand.

Real-time means that the system has a predictable response to an external event. The concern is both with the rapidity of response (minimum interrupt latency) as well as with the variations in response time. The concern is that the event is time critical and something detrimental to the operation of the system will occur if response is not sufficiently rapid.

The obvious analogue in a typical PC system is serial communications: if you do not respond to the serial interrupt quickly enough you will lose data. You should note that designers of communication packages for PCs will universally replace the DOS and BIOS serial interrupt handlers with ones of their own design for reasons of efficiency. The object lesson should be that simply because DOS has some support for some device or function at some level does not imply efficient or complete support. This is rarely the case.

Multitasking means the ability to switch the execution stream between distinct paths or "threads" through the program. Ideally these switches are driven by the occurrence of events both external and internal; hence there is an intimate connection with real-time capabilities.

My definition of multitasking does not imply the ability to execute multiple DOS programs simultaneously (although this is the common (mis)usage of this term in the PC world) and the multitaking requirement is not satisfied by this capability. Multitasking is the attempt to make full utilization of every available CPU cycle by overlapping CPU activity with slower asynchronous activities, especially I/O operations.

Unfortunately there are few choices for real-time operating systems for PCs. The common alternate operating environments such as DeskView, MS Windows or OS/2 don't directly address the issue of real-time/multitasking.

More appropriate to the problem at hand are systems like QNX by Quantum Software Systems Ltd. and AMX by Kadak Products Ltd. which were both examined at an early stage of this project. It was not at all obvious that the advantages of these systems warranted the abandoning of the well supported world of MS-DOS. The tradeoff was between the advantage of a cleaner runtime environment versus the need for more development time in a sparse development environment.

A compromise was to use the Timeslicer library from Lattice. This library allows for a limited capability of task switching and inter-task communications. Note that this approach does not address a fundamental weakness of MS-DOS as a pseudo-multitasking environment: its non-reentrancy.

3. Choice of Implementation Language

General language requirements:

- i. - High level language to minimize development time.
- ii. - Compilers ability to generate small, efficient code.
- iii. - Commercial availability of support libraries.
- iv. - Separate compilation units and block structure are desirable for maintainability.

FORTRAN and BASIC lack language elements such as bit primitives necessary if we need to do any system programming directly in the implementation language.

PASCAL and MODULA-2 are orphans without extensive support by third party developers of support libraries.

Assembler is not sufficiently high level.

We are left with C as the choice for implementation language. It's extremely difficult to justify any other languages other than C or Assembler for the development of any serious application for the PC.

4. Program Structure

To avoid causing more complications than MS-DOS provides, all code written for use in the Scintillation Data Recording System (SDRS) should be fully reentrant. This means that all parameters to a buffer space should be allocated from the heap not declared and static variables should not used inside functions. Variables declared with external scope should only be used as semaphores or to represent truly static information. External variables must never be used to pass parameters between functions.

The SDRS is a single executable "sdrs.exe". On startup the program enters an initialization phase during which the DMA board is configured, memory buffers are allocated, the various operational tasks are started, and finally ends with the triggering of the first DMA transfer. Currently the system configuration is static, being specified by the C header file "scint.h"; however, the initialization sequence was intended to allow dynamic configuration by an operator. This requires the improvement of the command line interpreter (see below).

The operational tasks are:

i. - DMA interrupt handler.
ii. - Statistics processor (one per active channel).
iii. - Command line interpreter.
iv. - Clock.

The DMA interrupt handler responds to the end of transfer interrupt from the MetraByte DMA board. It immediately swaps the DMA buffer and starts the next block transfer. The handler then unpacks the recently arrived block into individual double buffers for each processing channel. This step is make necessary because:

i. - Individual DMA transfers are limited to 64 Kbytes by the segmented architecture of the Intel 80x86.
ii. - The data received from the MetraByte board are in channel order and encoded with their channel number (see MetraByte manual).

The DMA handler makes use of the analog I/O and DMA support functions supplied by MetraByte.

The processing module is where all the action is. After receiving a full data buffer (actually a pointer) from the DMA handler, the processing module loads the raw data numbers into a histogram for computing the distribution function. Use of raw data numbers in this step is valid since the computation of the cumulative distribution function in calibrated units involves only a monotonic transformation of the raw values (as long as we don't try to interpolate values).

Next comes the application of the channel calibration which takes the short (16 bit integer) data numbers into calibrated floats (32 bit floating point) by the application of the predetermined lookup table. The type conversion is required since the calibrated values have dynamic range greater than 65,536 even though they have only 12 bits of precision.

The calibrated values are used to compute the S4 statistic (normalized variance), power spectrum (via fft), and mean zero (median) level crossing interval. Because of limitations of the segmented memory architecture and processing time constraints, the nominal 5 minute statistics interval (16,384 samples at 50 Hz) may be broken into several smaller buffers for the actual processing. The summary statistics are written to disk as a structure defined in "stats.h" consisting of the statistical data and a time hack.

When full waveform recording is enabled on a channel the computation and recording of statistical data is suspended. The (12 bit) data numbers are packed 2 samples per 3 bytes and written to disk. Filenames for both the statistical data and the full waveform data are simple combination of day number, UT start time, and channel number.

At present, the SDRS has only a very primitive command line interpreter which accepts single letter command from the keyboard. One major addition that must be made before a version of SDRS may begin to be shipped to the field is the conversion of the operator interface into a full system of windowed menus.

The SDRS was designed to take advantage of the full range of text and graphical display capabilities of the EGA and enhanced color monitor combination. The EGA text pages are intended for an operator interface based on the character windows defined the "Windows for C/Data" libraries. The EGA graphics pages are used to provide operational and status information such as plots and summary statistics. Access to the graphics screens is via a simple set of windowing functions defined in "gwindows.c" that were written using the Microsoft C graphics library.

To provide a UT clock display on screen, SDRS used a task which reads the system clock and updates the time window on screen once per second. A second task is used to keep the DOS time function within one second of the time provided by the battery-backed crystal controlled time/date clock available on the AT. This is accomplished by setting the system time to the clock time at regular intervals. In normal operation the time/date chip is only read at bootup to set the DOS time and date. DOS time and date are kept using a separate time base which, though more precise, is found to be less accurate than the time kept by the time/date chip.

5. Limitations

The fundamental limit to overall system performance is the ability of the Z-248 to perform the fft. By a large margin (>85%), the computation of the transform in order to estimate the power spectrum is the major contributor to time required to process a single channel. Currently we have chosen to limit the resolution and number of points in individual transforms (nb.

this is a completely separate issue from the number of points we save from the final power spectrum) as well as the number of channels on which transforms are performed.

There is no obvious solution given the hardware constraints. Investigating a more efficient algorithm for estimating power spectrum is probably a prudent investment of resources. Machine language fft routines, which are available either in the public domain or from commercial vendors (e.g., Alligator Technologies), may offer some improvement.

The alternative method of maximum entropy, while in the general case taking exponentially longer than transforms, may be worthwhile investigating if a small number of poles can satisfactorily reproduce the smooth scintillation spectra. One caveat, MEM is an iterative algorithm, and you must insure convergence within a given (real) time constraint. This is important when testing an iterative algorithm since it is not the median time that is relevant but the maximum possible (worst-case) time.

A second limitation which has more to do with the inability of MS-DOS to act as a true multitasking environment. The two faults are:

i. - MS-DOS is non-reentrant, which means that system calls cannot be overlapped. In particular, DOS INT 21h and all its various functions must be treated as a "critical resource" by using a locking mechanism to serialize all system activity including disk I/O.

ii. - In the AT architecture, hard disk I/O operations are done using programmed I/O via the CPU, rather than by DMA, so that disk access is compute bound. One must replace the BIOS routines for the hard disk (INT 13h) to change this.

This is not a major efficiency concern as long as the SDRS remains compute limited. If one were to appeal to a hardware solution to obtain the power spectrum, this becomes a much greater concern.

The non-reentrancy of MS-DOS is also a problem when adding any software or device drivers that make DOS system calls. The most immediate concern in this area is a tape drive. Whatever method is chosen to incorporate the drive into the system, any data which is spooled from the hard disk to the tape using the MS-DOS file system must involve issuing DOS system calls.

If tape activity is to occur simultaneously with the acquisition and processing of data, the tape drive software must adhere to the DOS access conventions of SDRS. Any software supplied with the tape drive is guaranteed to not adhere to these conventions since the specifics of task locking are unique to the SDRS. This choice is then to either:

i. - Stop data/statistics recording operations while the tape drive is active.

ii. - Write a complete set of software drivers for the tape that conform to the DOS protocol of SDRS.

Appendix 2. SDRS Operation

1. SDRS startup

After the SDRS hardware is set up and all cables are connected, turn on the printer, monitor, and computer. The automatic bootup procedure loads MSDOS and finishes with the MSDOS prompt, **C:\SDRS>** . Make sure that a formatted high density (1.2 Mbyte) floppy disk is in the floppy drive A:, and that a blank tape cartridge is available if full waveform data will be recorded. At the prompt type **SDRS** and press the Enter key. In the following descriptions, special keys such as Enter and Esc are designated by <Enter> and <Esc>. Type only the key indicated inside the angled brackets, not the brackets. Key combinations such as <Shift F1> indicate that the Shift and F1 keys should be pressed simultaneously.

A. Options for exisiting full wave files

When <Enter> is pressed and the screen clears and one of the following messages appears on the monitor:

No full wave form (.DAT) files found.
Loading main SDRS program...

or,

x full wave form (.DAT) files (bbbbbbb bytes) found.
Delete .DAT files?(y/n) or "dir" to see directory:

The second message indicates that there are x number of full wave files totaling bbbbbbb bytes from a previous SDRS session not backed up from the hard disk to a tape cartridge. A **dir** response lists a MSDOS like directory of the full wave form files. The user must then respond **y** (for yes=delete) or **n** (for no=do not delete) whether to delete or keep the files before continuing. If the files are retained they can be backed up at any time once SDRS is running. Depending on the response one of the following message appears:

All exisisting full wave form files will be deleted.
<Enter> to confirm, <Esc> to cancel:

or,

All existing full wave form files will not be deleted.
<Enter to confirm, <Esc> to cancel:

<Enter> confirms the message; <Esc> cancels the message and allows another chance to delete or keep the files. Once confirmed the response is acknowledged by one of the messages:

All full wave files deleted.
Loading main SDRS program...

or,

Full wave form files not deleted.
Loading main SDRS program...

B. Setup screens

The first screen to appear (Figure 17 a) identifies the site and basic system configuration. To change an item use the up and down cursor (arrow) keys on the numeric key pad or press one of the highlighted keys to select its current value, which will be highlighted in a bright white color. Change the value by typing a valid entry for that item (text for the **Site**; an integer from 1 to 5 for the total number of **Scint Chans** (Scintillation channels); and **0** (= no) or **1** (= yes) for two channel **Polar**(imeter) data. When an item is modified the value or text changes to yellow color. Press <Enter> to confirm the change. Use <Back Space> to edit a character or press <Esc> to cancel the change entirely. Invalid entries are not accepted. When the setup configuration is correct select **OK: Proceed** and press <Enter>.

The **PROCESSING OPTIONS** window should then appear like Figure 17 b if there is no polarimeter present (Polar = 0 in the previous screen) or like Figure 17 c if there is (Polar = 1).
This window displays the default channel configuration and processing options for the current setup. The first channel(s) are always scintillation (amplitude) channels and the polarimeter channels, if present, occupy the two channels immediately after the last scintillation channel. A **1** under a **CHANNEL** number opposite a particular process indicates that the process is turned on for that channel; a **0** indicates that it is turned off. First use the up and down arrow keys or highlighted keys to select an item. Then use the left and right arrow keys to select a desired channel by highlighting the current value in bright white and type in a new value (0 = off; **1** = on). The default setup is: **FWF Rec**, full wave form recording, is off (0) for all scintillation channels; **Stat**, statistics (S4 and cumulative distribution function), is on (1) for all scintillation channels; **Spectrum**, power spectrum and autocorrelation, is on (1) for all scintillation channels; and **POL**, polarimeter recording, is on (1) for the two polarimeter channels, if present. When the Processing Options are correct, select **OK: Proceed** and press enter to continue with SDRS initialization.

The starup procedure can be aborted at any point by pressing <Esc> once or twice. This will return the system to the MSDOS prompt **C:\SDRS>**. This startup procedure must be used whenever the physical channel arrangements is changed, generally when SDRS AMP or POLAR AMP cards are added.

C. Start of main SDRS Program

After the screen clears, the printer will print a short sign on message, and the **SDRS** screen will appear (Figure 18). Section 2.B describes the screen windows in detail. The clock should be advancing in the **CLOCK** window, and, after about 10 seconds, the the **Calibrate** option should be highlighted (bright white color) in the window labeled **INITIAL OPTIONS**. Use the up or down cursor (arrow) keys on the numeric key pad to select (highlight) one of the three initial options. Press <Enter> to carry out the function selected. The **Calibrate** option immediately clears the **SDRS** screen and puts up the **CALIBRATION PROCEDURE** screen (Figure 33). This screen is used to enter new calibration curves for scintillation (SDRS AMP) channels and either create or modify calibration (.CAL) files on the hard disk. Calibration is described in Section 2.E. The **Use Current Calibration** option assumes that up to date calibration files exist and SDRS initialization proceeds. The last option, **Quit to DOS**, immediately exits SDRS, clears the screen , and returns to MSDOS with the MSDOS prompt (**C:\SDRS>**). The **Quit to DOS** option must be used to return safely to MSDOS. The standard MSDOS abort combination <Ctrl c> should not be used to quit the SDRS program. It may appear to work but the proper cleanup procedures to exit the program are not carried out and MSDOS will not behave properly. Turning the power off and then on, or pressing <Ctrl Alt Del> (warm reboot) begins the power failure restart procedure for SDRS that bypasses the **INITIAL OPTIONS** selections (equivalent to the **Use Current Calibration** option).

SDRS initialization continues after selecting **Use Current Calibration** and pressing <Enter> , or, if **Calibrate** is chosen instead, after exiting the calibration procedure . As calibration files are processed a CAL status will come up under each scintillation channel in the **MONITOR** window and messages will be displayed in the **CALIBRATION FILES** and **MESSAGES** windows (Figure 19). Each channel takes approximately 5 seconds to process. The time when a channel was last calibrated is also given in the **CALIBRATION PROCEDURE** screen (Figure 33). When all initial processing is completed the screen will appear similar to Figure 20. In this example the **MONITOR** window indicates that channels 1 and 2 are receiving scintillation data, designated by **AMP**, and channels 3 and 4 are receiving polarimeter data, designated by **POL**. At this time the system is transferring channel data to the memory buffers as shown by the information under **DMA** (Direct Memory Access) in the **CLOCK** window. After noting the storage data in the **STORAGE** window press <Esc> to clear the data and display the operator functions in the **MAIN**

OPTIONS window (Figure 21; the bottom half of this figure contains the plots selected by the **Display** option, Section 2.D.2). From this time on it is the responsibility of the operator to (1) perform periodic calibrations, described in Section 2.E, (2) change the recording media, disk and/or tape, and (3) replace the printer paper.

The startup procedure described above should not need to be repeated except when the channel configuration is changed; in practice, the system may have to be restarted when problems develop. The most likely event is a program error that causes the system to behave erratically or stop. If the **SDRS** screen is still readable the situation usually will be apparent in the **CLOCK** window (Figure 18) by a stopped clock and/or DMA percentage. Please note the data in all the windows because the information could be useful for corrections in future versions of the program. To restart the system turn the computer power off, wait a few seconds, and turn the power on. Since the program was not exited normally, the system will execute the automatic restart procedure and immediately go through initial processing and begin receiving data as shown in Figures 19 and 20.

2. SDRS operation

This chapter begins with general descriptions of the main SDRS screen and windows (Sections A and B). Section C explains in greater detail how the system works and what the operator should see as data is collected and processed. Section D describes how the operator executes all user options from the Options menu, except for calibration which is treated separately in Section E.

A. Main SDRS screen

While SDRS is running the main SDRS screen will appear as shown in Figure 21. The top half of the screen consists of four windows (**CLOCK, MAIN OPTIONS, MONITOR,** and **MESSAGES**) where the operator can view system performance and execute commands. The lower right corner of the screen displays the amplified AGC signal, spectrum, and correlation during the previous 82 seconds for a selected scintillation channel. To the left is a summary plot of the most recent 12 or 24 hours of S4 and correlation time for the same scintillation channel. Alternatively, summary plots of the two channel polarimeter data can be displayed.

From left to right, the top line displays the station entered in the setup screen, the SDRS version number, and the keyboard status, **READY** or **WAIT**. In the second line under the station is the date and time when the current SDRS session was started, whether from the setup screen, full waveform backup, or restart after a power failure. Under the keyboard status is the maximum time used by a processing cycle during the current session.

B. Window descriptions

1. **CLOCK** window (Figure 22)

The top half of the window displays the current Universal (UT) date and time. The time shown in red is the system clock; the one in black is the battery operated real time clock, which should agree with the system time to within one second. The real time clock is normally used only to set the system date and time when the computer is turned on or after a power failure. However, because the real time clock is more accurate than the system clock, SDRS synchronizes the two clocks every 15 minutes. The real time clock can be reset while SDRS is running. The portion of the window below **DMA** (Dynamic Memory Access) gives current information about the status of the channel data buffers (% filled) and A/D byte number transferred to memory by DMA. Also shown are which DMA buffer is currently being filled , **buffer_1** or **buffer_2**, and the data sample being collected, 1 through 4. On the last line are shown the status word, ST, and the control word, CT, of the A/D transfer. When the sytem is working correctly, this line should be, ST=2x CT=B7, where x changes as the clock is updated and can be any number from 0 to one less than the number of channels being collected.

2. **MONITOR** window (Figure 23)

Along with the **CLOCK** window, this window indicates the current status and activity of the system. Under each channel are shown its current status, the seconds elapsed during the last processing cycle, and the data sample number awaiting to be or currently being processed. Activity in this window is described in the next section.

3. **MESSAGES** window (Figure 23)

From time to time system messages will appear and scroll by in this two line window. Some of these messages are warnings or acknowlegments to operator, others are programmer messages that are often cryptic but can be useful to diagnose a problem or monitor a particular function. Programmer messages may change from one version of SDRS to the next and are not crucial for system performance.

4. **MAIN OPTIONS** window (Figure 24)

The operator executes all options, enters data, and confirms all input in this window. If the window does not appear as in Figure 24, press <Esc> to display the options menu. As indicated by the prompt, the options menus is activated by pressing <Shift F1>, which causes the first item, **Processing**, to be selected (highlighted in bright white color). Use the cursor keys on the numeric key pad or the highlighted key in each option to select a desired item. BE SURE THAT THE Num Lock KEY IS OFF!! Otherwise, the number keys on the key pad are active instead of the cursor keys. To run an option press the <Enter> key while an option is selected. When finished with the option press <Esc> to return to the unactivated options menu.

C. Data collection and processing

While data is being collected and no processing is taking place the **status** line in the **MONITOR** window will appear like Figure 23. In this example **AMP** indicates that channels 1 and 2 are collecting amplitude data for scintillation measurements and **POL** indicates that channels 3 and 4 are collecting polarimeter or phase data for total electron content measurements. When the percentage under **DMA** in the **CLOCK** window cycles back to 0% the channel buffers are full and the channel status changes to **BUF**, in red, for all active channels. This initiates data processing starting with the first active channel. Depending on what is selected, each step appears in bright on the **status** line under the channel being processed (**PAK** for full wave form packing and recording, **STA** for S4 and distribution statistics, **FFT** for power spectrum by fast fourier transform, **COR** for autocorrelation by inverse transform, and **POL** for 1 sec polarimeter averages). When processing completed **Ok** is displayed on the the **status** line and processing begins for the next channel. When the last active channel is completed the **status** line reverts to Figure 23.

While processing the previous buffer, SDRS acquires data in the current buffer as shown by the **DMA** percentage in the **CLOCK** window. It takes approximately 82 seconds to collect a sample of 4096 points for all channels at a 50 samples/sec rate. This 82 second collection cycle is a basic period of the SDRS system. Data processing and recording must be completed before the active buffer is filled at the end of the collection cycle when the **DMA** percentage resets to 0%. In the **CLOCK** and **MONITOR** windows, sample refers to the current collection cycle in a recording cycle that consists of four samples and takes about 5 minutes and 28 seconds. After the fourth sample is processed, data is recorded to the floppy disk (A drive) and summary information is displayed by the printer, if it is turned on. As data is recorded, **REC** appears under each channel in the **status** line. Note: the keyboard is locked out while SDRS records to the floppy disk. After a recording cycle is completed the **Storage Info** option is automatically executed (section D.8, Figure 31); press <Esc> to return to the options menu.

D. User options

To run any user option: (1) start from the options menu in the **MAIN OPTIONS** window (press <Esc>, if necessary), (2) activate the options menu by pressing <Shift F1>, (3) select an option by using the cursor keys or highlighted key in each option so that the desired option is highlighted in bright white, and (4) press <Enter> to start the option. Figure references below show the appearance of the window accompanying each option.

1. **Processing** (Figure 25)

Purpose: To show or change current processing options.

Options for processing each active channel are indicated by a **1** for yes, perform the operation, or 0, for no, do not perform the operation. The appropriate operations for amplitude (scintillation) channels are **FWF Rec** (full wave form record), **Stats** (S4 and cumulative distribution function), and **Spectrum**, (power spectrum and autocorrelation). Use the up and down cursor keys or highlighted keys to select an option line by highlighting the operation in red; pressing <Enter> selects the next option line. **Exit**, however, is highlighted in bright white when selected (see below). Use the left and right cursor keys to select a channel by hightlighting in bright white the current option (1 or 0) under the channel number. Type 1 or 0 to change an option. Changes do not become effective until <Enter> is pressed while **Exit** is selected. If the **Spectrum** option is on for a channel then **Stats** will be automatically turned on for that channel even if it is not explicitly chosen. The two polarimeter channels, if present, can only be chosen to be **1** (yes, record two channel polarimeter data), or **0** (no, do not record) and are selected and modified as a pair.

NOTE: Changes made in the processing directions do not take effect until the **Exit** line is selected and <Enter> is pressed. This permits the operator to correct mistakes or return to the options menu without making any changes. If <Enter> is pressed while the **Exit** line is selected and an option has been turned off (changed from **1** to **0**) the action takes effect immediately with the next 82 second processing cycle. However, if an option has been turned on the action does not take effect until the beginning of the next five and one half minute recording cycle.

2. **Display** (Figure 26 a-b)

Purpose: To show or change current display options.

This option controls the plot displays seen in the bottom half of the main SDRS screen. During every 82 second processing cycle the raw AGC signal, spectrum, and autocorrelation of a selected scintillation channel are displayed in the lower right plot. If there are no active polarization channels, the summary plot on the left shows the S4 and a measure of the autocorrelation during the past 12 or 24 hours for the selected scintillation channel. If there are polarization channels the summary plot may be chosen to show the two polarization channels instead. As in the **Processing** option above, any changes in this window do not become effective until <Enter> is pressed while the **Exit** line selected or highlighted in bright white.

If there are no polarization channels, only the scintillation channel to be displayed, indicated on the AMP line, and the summary plot time scale, shown on the **Time** line, can be changed. To change the channel, use the up and down cursor (arrow) keys or "a" until **AMP** is highlighted in red and use the left and right cursor keys to select the desired channel by highlighting the **1** (on = display) or **0** (off = do not display) under the channel number. Then type 1 or 0 to change the channel selection. Only one channel can plotted during any processing cycle; turning a channel on automatically turns off the channel that was formerly on. To change the time scale of the summary plot, select the **Time** line and use the left or right cursor keys to toggle between **12 hr** or **24 hr**. Finally, select **Exit** and press <Enter> to confirm the changes or press <Esc> at any time to return to the options menu without performing any changes, if any. The summary plot reflects changes immediately after confirmation; the scintillation plot changes with the next processing cycle.

If polarization channels are present, they can be displayed in the summary plot by selecting the **POL** line and typing 1 to turn on the polarization channels as a pair (Figure 26 a). When selected, **POL** channels take display priority over a selected **AMP** channel. If the polarization channels are displayed, the voltage range in the summary plot can be changed by selecting the **Scale** line and pressing the left or right cursor keys to toggle among four ranges (**(-10v,+10v)**, **(-5v,+5v)**, **(-10v,0v)**, **(0v,+10v)**). To turn off the polarization plot and return to the scintillation summary plot, select the **POL** line and type 0 (Figure 26 b). Remember to confirm any changes by selecting **Exit** and pressing <Enter>.

3. **Calibrate** (see Section E)

Purpose: To view and update calibration curves for scintillation channels; to monitor real time data collection from scintillation or polarization channels.

4. **Screen** (Figure 27 a-b)

Purpose: To view and change screen blanking time.

To prevent a fixed image from burning in the video screen, the screen automatically blanks 20 minutes after the last keystroke by default. Press any key to unblank and revert back to the normal video screen. While the screen is blanked the clock and DMA percentages will still be visible in dark red. To change the blanking time enter a number from 1 to 60 minutes and press <Enter> to confirm the change or <Esc> to cancel the change. Press <Esc> to return to the options menu without making a change ,or after confirming or canceling a change.

5. **Clock** (Figure 28)

Purpose: To set the real time clock.

The time can be set by entering a universal (UT) in hh:mm:ss format and pressing <Enter> to synchronize the computer's real time clock with a reliable clock. Press <Esc> to return to the options menu without making a change or after setting the clock.

NOTE: Do not attempt to set the clock while recording to the floppy disk (A drive light on). It is also better not set the clock while processing is taking place. Response may be slow and the elapsed time in the **MONITOR** window may be inaccurate or show 000 for that processing cycle.

The date cannot be set while SDRS is running. Under normal circumstances the time may drift several minutes over an extended period and cause the date to be off by one day near 00 UT. Wait until the date shown in the SDRS **CLOCK** window is correct before setting the clock. Otherwise, quit SDRS (below), set the calendar clock in the Zenith setup screen, reboot the computer, and start SDRS (Section 1 A).

6. **Tape** (Figures 29 a-c)

Purpose: To show scheduled tape backup times and to notify SDRS that a new tape has been installed.

NOTE: This option will run only if full wave form recording is on for at least one scintillation channel. Make processing changes from the **SDRS SETUP** screen or **Processing** option (above).

Because the tape cartridge drive can only archive files, the current system is programmed to (1) exit to DOS when 25 Mbytes of full waveform files (.DAT) have been recorded to the hard disk D drive, (2) run the backup program, and (3) restart SDRS automatically. Two backups are made to each 60 Mbyte tape cartridge. To work properly the system must know whether a tape is present and if it is new, half full, or full; for example, if the system has recorded 25 Mbytes and is not aware that a suitable tape is present, full wave form recording will be suspended. The time to replace a tape is displayed after performing the **Tape** option as described in this section, as well as in the line immediately below the **OPTIONS** window and by performing the **Storage Info** option (below).

When 25 Mbytes of full wave form files have been recorded, SDRS exits to DOS and starts the SYTOS backup program. After a few moments the SYTOS screen appears and the program searches for the end of information to start appending files if it is backup 2. After all full waveform files are dumped the tape is rewound. Then the backup log is printed, the files that were backed up are deleted from the hard disk, and SDRS automatically restarts with the same settings prior to backup. The entire backup process takes approximately 5 to 10 minutes.

After **Tape** is selected in the options menu and <Enter> is pressed, the **OPTIONS** window will list two tape options, **New Tape** and **Start backup**, and appear like Figure 29 a; in this example one backup has already been performed on the current tape. **New Tape** and **Start backup** are options that can be selected (highlighted in bright white) by using the up and down cursor keys or highlighted keys. **New Tape** is the option generally used from this window.

6.1 **New Tape**

Purpose: To notify SDRS that a new blank tape has been installed.

1. Remove the old filled or partially filled tape from the tape drive.

2. Select **New Tape** and press <Enter>.

3. Window Figure 29 c will appear. It warns that the next scheduled backup will start at the beginning of the new tape. NOTE: Any data on the new tape will be erased. Press <Enter> to continue with the new tape procedure. Press <Esc> to return to the tape options without making any changes.

4. If <Enter> is pressed in step 3, window Figure 29 d will appear. The **Tape name** assigned by SDRS is not written to the tape. It is important, then, that the operator writes the name on the tape's paper label to prevent unintended use later on. The name contains the year, day, and UT it was generated (day 043, 1991 at 1431 UT in the example). Insert the new labelled tape into the tape drive, close the latch, and press <Enter> to continue.

5. The tape options window appears as in Figure 29 e. The bottom four lines give the current tape name and the approximate times of backup 1, backup 2, and the replacement deadline for a new tape. The tape can be replaced any time after backup 2 and before the replacement time (Sat Feb 16 11:00 - Mon Feb 18 09:00 in the example). This is the period required to record 25 Mbytes of full waveform files to the hard disk. Full waveform recording will be suspended if a new tape is not installed before the replacement deadline. After the operator performs the **New Tape** option, backup is started automatically after the next record cycle, if 25 Mbytes are already on the hard disk. A backup can be started immediately using the **Start backup** option (below).

6. Press <Esc> to return to the options menu.

6.2 **Start backup** (Figure 29 a or e)

Purpose: To exit to DOS and start full waveform backup immediately.

This option permits the operator to manually start backup without waiting for a full 25 Mbytes to be recorded to the hard disk. Select **Start backup** and press <Enter>; Figure 29 b appears, and gives the option to actually start the backup. Press <Esc> to cancel and return to the options menu.

7. **Disk** (Figure 30)

Purpose: To notify SDRS that a new floppy disk has been installed.

If <Enter> is pressed while the **New Disk** option is highlighted in Figure 20, SDRS writes a short file to the floppy disk identifying it as a SDRS data disk and resets to zero the number of bytes written to the A drive (see **Storage Info** below). If SDRS finds the identifying file already on the disk a warning will appear. Pressing <Enter> only writes a new identifying file and resets the byte counter; no data files will be deleted. The **Storage Info** option automatically runs immediately after **New Disk** is performed. Press <Esc> at any time to return to the options menu.

The only time **New Disk** <u>must</u> be run is when the floppy disk in the drive is full and the message **floppy**

disk full flashes above the **MONITOR** window with an accompanying beep. Statistical and polarimeter data recording is suspended at this time. The only way to silence the annoying beep is to replace the full disk and run **New Disk**.

8. **Storage Info** (Figure 31)

Purpose: To show remaining disk storage capacity.

This option, which is automatically run after each recording cycle, displays the number of bytes free and written to both the floppy disk A drive (statistics) and hard disk D drive (full waveform), as well as the replacement time for the floppy and next backup time from the hard disk to tape. Running **New Disk** from the **Disk** option (above) resets the bytes written to A; the bytes written to D is reset after each successful tape backup. Press <Esc> to return to the options menu.

9. **Quit to DOS** (Figure 32)

Purpose: To quit SDRS and return to the MS-DOS operating system.

When Figure 32 appears, press <Enter> to quit SDRS and return to the DOS prompt **SDRS>**. Press <Esc> to return to the options menu. NOTE: <u>Do not</u> use the normal DOS abort key combination <Ctrl c> to quit SDRS. If it works DOS will not run properly and you will have to reboot the computer.

<u>E. Calibration Instructions</u>

Step 1. From the SDRS screen (Figure 21) select **Calibrate** from the options menu (press <Shift F1> and use the cursor (arrow) keys or "c" to highlight **Calibrate**) and press <Enter>.

Step 2. The **CALIBRATION PROCEDURE** screen (Figure 33) appears. If a calibration file for channel 1 (CHAN1.CAL) exists, the current calibration curve is displayed in the lower left plot; otherwise, the calibration for this channel will use the **LEARN** mode. The real time AGC voltage from channel 1 of the SDRS amplifier is displayed in the lower right plot. Every 10 seconds (**time(sec)** value) the trace wraps around and plots alternately in red and blue. Enter a valid **Channel** number and press <Enter> to display and calibrate a different channel; only the real time display is active when a polarimeter channel is chosen.

All calibrations start at 0 db attenuation applied to a fixed reference rf signal (e.g., -80 db) substituted for the antenna input. If a calibration file exists the **FILE** mode is automatically selected for the calibration run (below) and the attenuation settings from the file are used. During a calibration run the operator has the option to change to the **LEARN** mode and set the remaining attenuation levels manually.

<u>Routine Calibration (**FILE** mode)</u>

To update a calibration curve using the attenuation settings of the current curve, follow the steps below:

Step 1. Press the right cursor (arrow) key or "r" to select **Start run** (Figure 34) and press <Enter>. The **Step Mode** (Figure 35) indicates that the **FILE** mode applies. The **Step size** shows the next attenuation step in db relative to the current **db attenuation**. In the example shown in Figure 35, the first calibration point is at 0 db attenuation (e.g., -80 db absolute rf signal level) and the second point will be at 1 db attenuation (-81 db absolute). The current curve is replotted in black for reference.

Step 2. Disconnect the antenna and connect the signal generator. The white dot in the calibration plot (lower left) shows the signal strength at the minimum attenuation (0 db). The values under **db atten.** and **A/D value** indicate the attenuation level and signal strength that will be recorded when <Enter> is pressed. Figure 36 shows how the screen appears after the first point is accepted. The second point is pending at 81 db attenuation.

Step 3. When the signal stabilizes, press <Enter> to accept the displayed values of **db atten.** and **A/D value**. Because the calibration curve must be a single valued (monotonic) relation of AGC voltage vs. signal strength, a calibration point is valid only if the AGC voltage, or gain, is greater than all preceding points at smaller attenuations. The current point is valid if the signal level dot and **A/D value** are white in color. Otherwise an invalid point is shown in red and will not be accepted; check that the attenuation of the rf signal is correct if this occurs. An accepted point is plotted in blue and **db atten.** increases by the indicated **Step size**. The signal dot also moves toward the right to the new attenuation level.

Step 4. Attenuate the signal strength by the value shown under **db atten.**

Step 5. Repeat step 3 above, or press <Esc> to abandon a calibration run and start over from step 1. To exit a run before the last attenuation level is accepted, press "q" and proceed to step 6.

Step 6. Figure 37 shows the screen before the last point is accepted. When the last attenuation level is accepted the updated calibration curve will be replotted if it is valid (Figure 38).

Step 7. The **Save** option will be highlighted and a **SAVE CALIBRATION** window will appear with the options to press <Enter> if the new curve is acceptable, <Esc> if not. A few seconds after <Enter> is pressed the **Last Calibrated** date and time will be updated indicating that the new curve has become the current calibration. If <Esc> is pressed **Save** is not executed and the calibration file is not updated; the current calibration file can be viewed again entering the approprate **Channel** number, and pressing <Enter>.

Step 8. Enter the next **Channel** to be calibrated. Press <Enter> and repeat from step 1 above.

Step 9. To exit **CALIBRATION PROCEDURE** and return to the **SDRS** screen, use the cursor keys, <Esc>, or "x" to select **Exit calibration** and press <Enter>.

New Calibration (**LEARN** mode)

If there is no existing calibration file (CHANn.CAL, where n is the channel number) the screen will appear like Figure 39 and the calibration will automatically start in the **LEARN** mode when **Start run** is executed. However, even if there is an existing calibration file, the operator can switch to the **LEARN** mode after **Start run** is executed by pressing "m" and toggling between **FILE** and **LEARN** by pressing the space bar (Figure 40). Press <Enter> while **LEARN** is displayed; once in the **LEARN** mode it is not possible to switch back to the **FILE** mode. Next, enter the desired db **Step Size** from 1 to 9 db (Figure 41). Press <Enter> again to confirm the db **Step Size** and continue from step 2 or 3 above, depending on whether the first calibration point has been accepted (Figure 42). Press "q" at any time to exit the calibration run and save the newly entered curve, if desired.

While in the **LEARN** mode, the attenuation for the pending calibration point can be changed by pressing "s", entering a new **Step Size**, and pressing <Enter> again to return to the calibration run. For example, if the last accepted point was at 18 db and the current **Step Size** is 2 db, the pending point will be at 20 db. To change the **Step Size** to 3 db, press "s", type "3", and press <Enter>. The pending calibration point will move to 21 db and the remaining points will be at 24 db, 27 db, 30 db, etc., unless the **Step Size** is changed again.

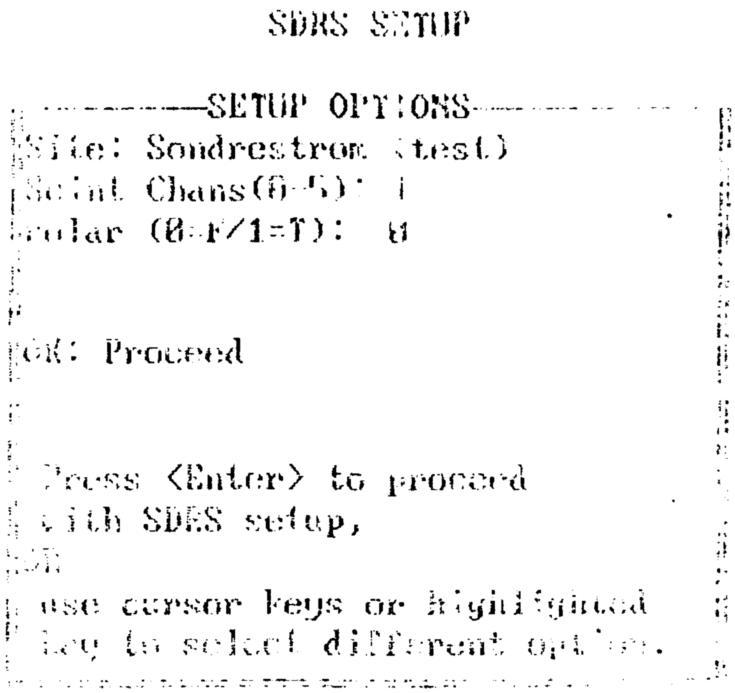


Figure 17a

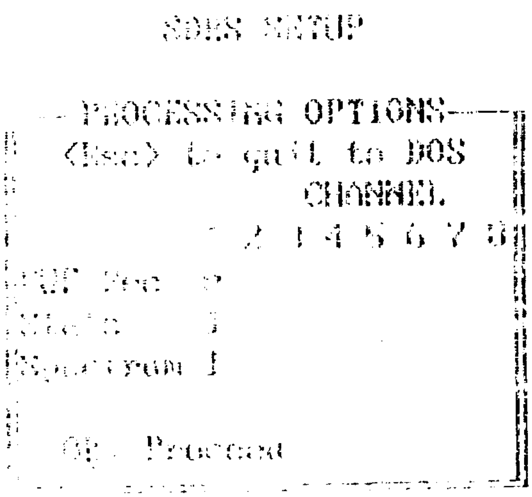


Figure 17 b

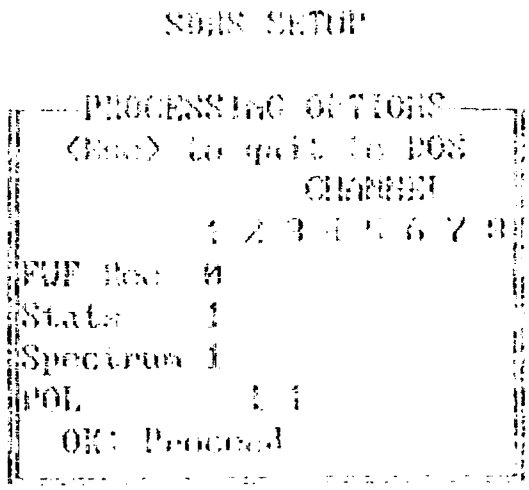


Figure 17c

```
Hanscom AFB                       SDRS Ver. 1.10              Keyboard: READY
Started: Feb 13 1991 20:15:01 UT                              max time:
—CLOCK——  ———INITIAL OPTIONS————  ——————————MONITOR————————————
Feb 13 1991                        chan.   1   2   3   4   5   6   7   8
20:15:12 UT                        status  INI INI INI INI
20:15:13   Calibrate                 sec.
   DMA     Use Current Calibrations sample
           Quit to DOS
                                   ——————————MESSAGES———————————
       W                           sine table computed

ST=FF CT=FF
```

Figure 18

```
Hanscom AFB                       SDRS Ver. 1.10              Keyboard: WAIT
Started: Feb 13 1991 20:15:01 UT                              max time:
—CLOCK——  ———CALIBRATION FILES———  ——————————MONITOR————————————
Feb 13 1991                        chan.   1   2   3   4   5   6   7   8
20:16:09 UT Chan 1 last calibrated: status  CAL CAL INI INI
20:16:09    01/11/1991 19:55 UT      sec.
   DMA      Chan 2 last calibrated: sample
            01/11/1991 20:01 UT
                                   ——————————MESSAGES———————————
       W                           Creating A:\10442016.STA
                                   Creating A:\20442016.STA
ST=FF CT=FF
```

Figure 19

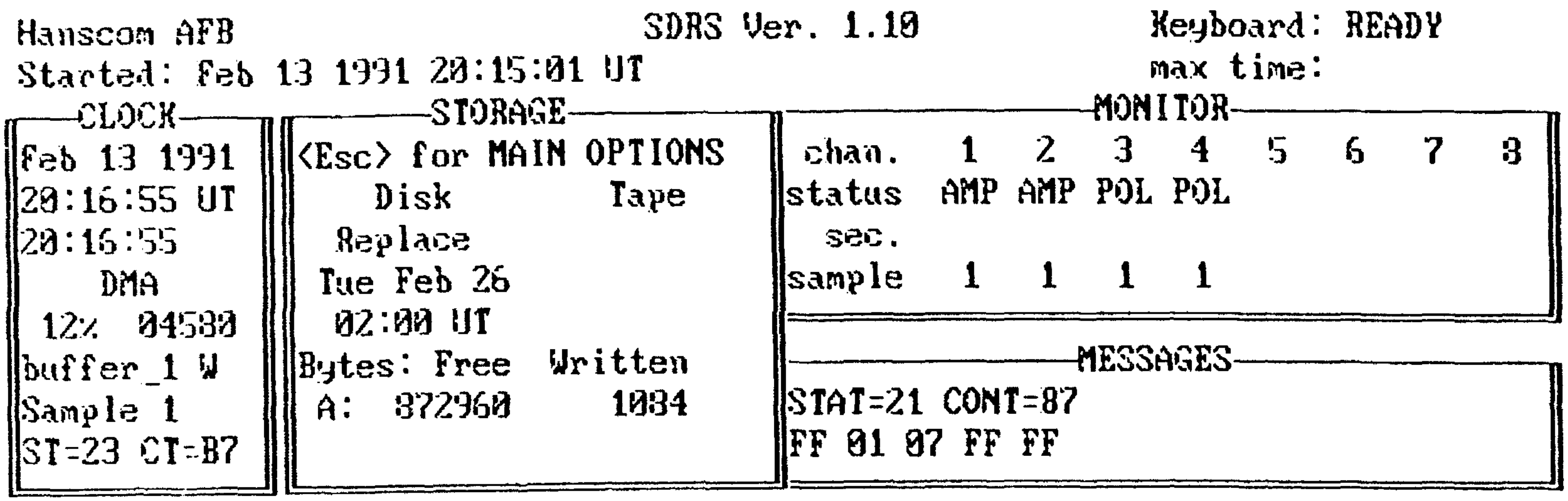

Hanscom AFB
SDRS Ver. 1.10
Keyboard: READY
Started: Feb 13 1991 20:15:01 UT
max time:
CLOCK
Feb 13 1991
20:16:55 UT
20:16:55
DMA
12% 04580
buffer_1 W
Sample 1
ST=23 CT=B7
STORAGE
<Esc> for MAIN OPTIONS
Disk Tape
Replace
Tue Feb 26
02:00 UT
Bytes: Free Written
A: 872960 1084
MONITOR
chan. 1 2 3 4 5 6 7 8
status AMP AMP POL POL
sec.
sample 1 1 1 1
MESSAGES
STAT=21 CONT=87
FF 01 07 FF FF

Figure 20

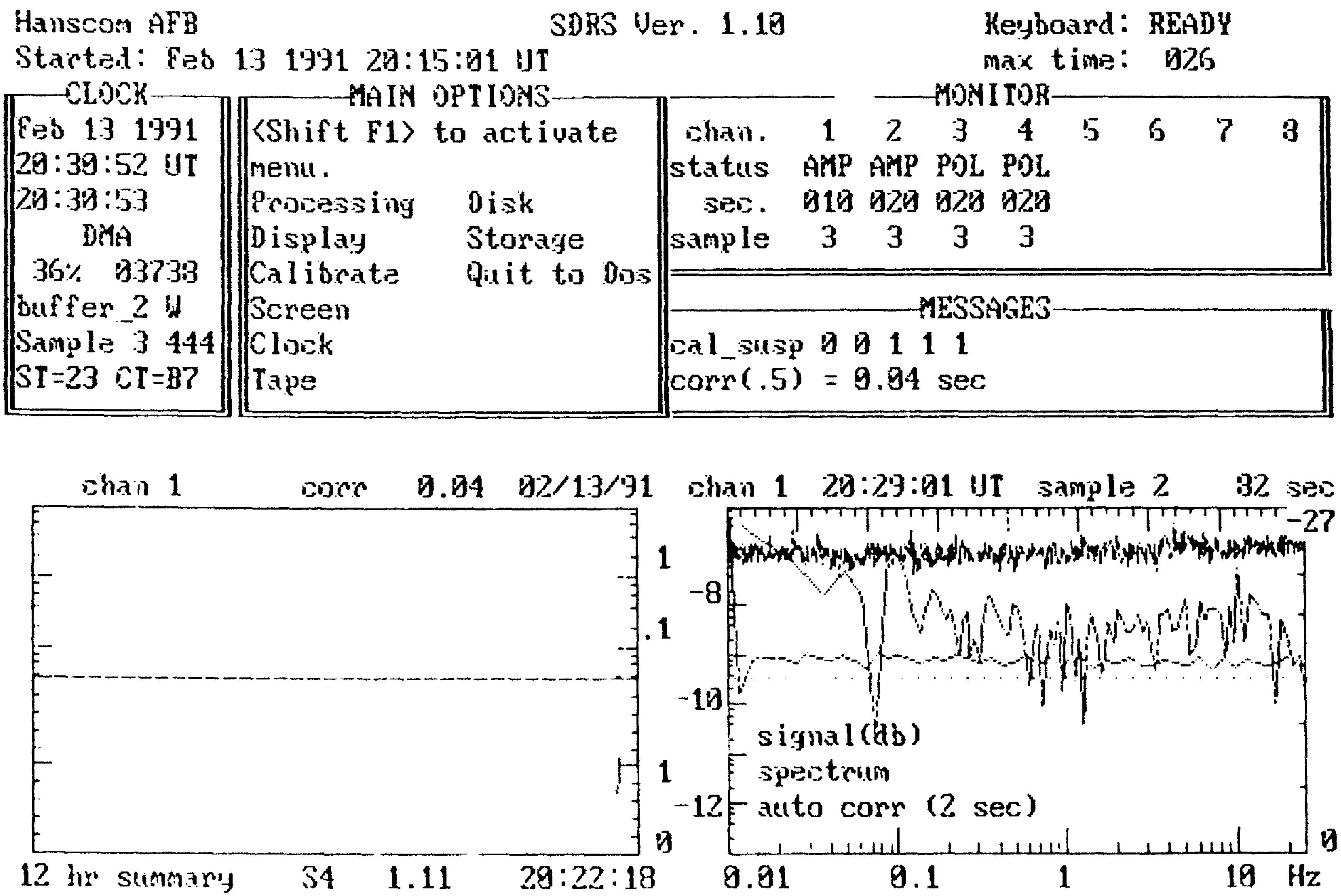

Hanscom AFB
SDRS Ver. 1.10
Keyboard: READY
Started: Feb 13 1991 20:15:01 UT
max time: 026
CLOCK
Feb 13 1991
20:30:52 UT
20:30:53
DMA
36% 03738
buffer_2 W
Sample 3 444
ST=23 CT=B7
MAIN OPTIONS
<Shift F1> to activate
menu.
Processing
Disk
Display
Storage
Calibrate
Quit to Dos
Screen
Clock
Tape
MONITOR
chan. 1 2 3 4 5 6 7 8
status AMP AMP POL POL
sec. 010 020 020 020
sample 3 3 3 3
MESSAGES
cal_susp 0 0 1 1 1
corr(.5) = 0.04 sec
chan 1
corr 0.04 02/13/91
chan 1 20:29:01 UT sample 2 32 sec
signal(db)
spectrum
auto corr (2 sec)
12 hr summary S4 1.11 20:22:18
0.01
0.1
1
10
Hz

Figure 21

```
CLOCK
Feb 13 1991
20:29:26 UT
20:29:27
   DMA
 30%  02036
buffer_2 W
Sample 2 444
ST=22 CT=B7
```

Figure 22

```
                          MONITOR
 chan.    1   2   3   4   5   6   7   8
status   AMP AMP POL POL
  sec.   010 020 020 020
sample    2   2   2   2

                         MESSAGES
cal_susp 0 0 1 1 1
corr(.5) = 0.04 sec
```

Figure 23

```
MAIN OPTIONS
<Esc> to de-activate
menu.
Processing   Disk
Display      Storage
Calibrate    Quit to Dos
Screen
Clock
Tape
```

Figure 24

```
  PROCESSING OPTIONS
 <Esc> for MAIN OPTIONS
             CHANNEL
         1 2 3 4 5 6 7 8
FWF Rec  0 0
Stats    1 1
Spectrum 1 1
POL          1 1
  Exit
```

Figure 25

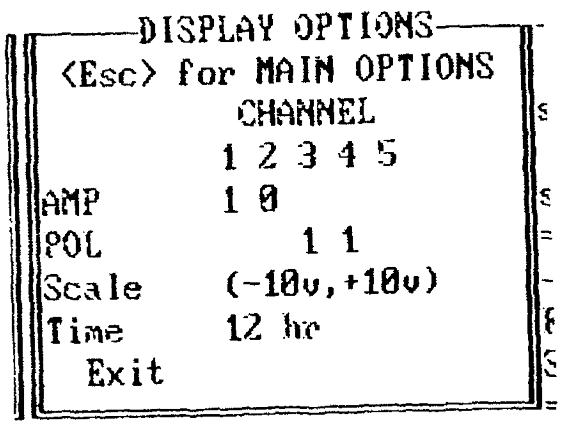


Figure 26a

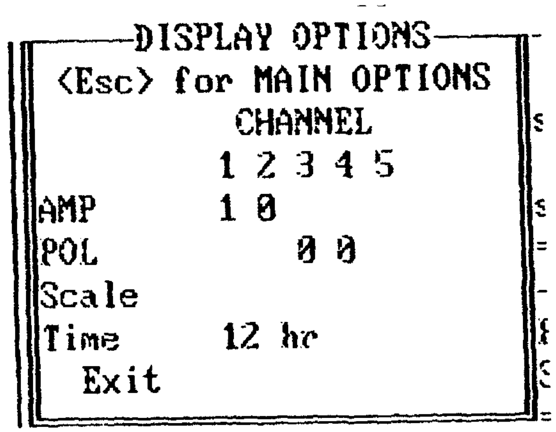


Figure 26b

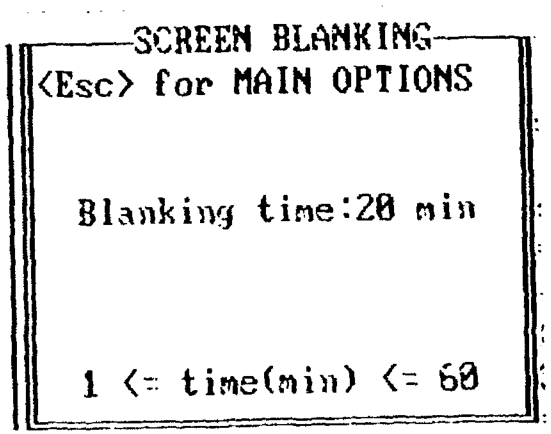


Figure 27a

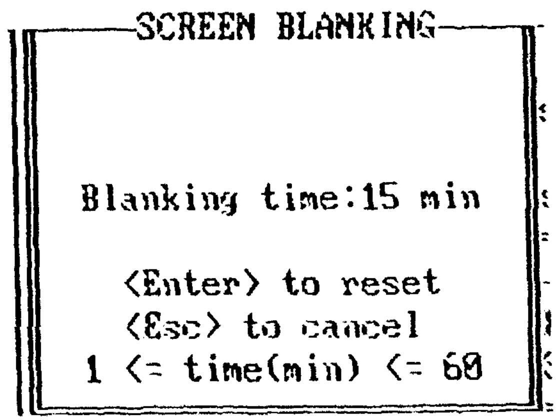


Figure 27b

```
RESET CLOCK
<Esc> for MAIN OPTIONS

00:00:00

<Enter> to reset
```

Figure 28

```
TAPE
<Esc> for MAIN OPTIONS

New Tape
Start backup
Tape: sdrs910241924 (2)
1st. Done
2nd. Thu Feb 14 12:00 UT
Rep. Sat Feb 16 11:00 UT
```

Figure 29a

```
TAPE
***START BACKUP***
WARNING!
<Enter> to exit SDRS
and start tape backup

<Esc> to cancel
```

Figure 29b

```
TAPE
****New Tape****
Next backup will start
at beginning of tape.

<Enter> to Confirm
<Esc> to Cancel
```

Figure 29c

```
TAPE

<Enter> to Continue

Tape name:sdrs910431431
Write tape name on
label and insert tape.
```

Figure 29d

```
TAPE
<Esc> for MAIN OPTIONS

New Tape
Start backup
Tape: sdrs910431431 (1)
1st. Thu Feb 14 12:00 UT
2nd. Sat Feb 16 11:00 UT
Rep. Mon Feb 18 09:00 UT
```

Figure 29e

```
DISK
<Esc> for MAIN OPTIONS

Replace floppy disk
and press <Enter>
New Disk
```

Figure 30

```
STORAGE
<Esc> for MAIN OPTIONS
   Disk          Tape
 Replace       Backup 1
 Sun Feb 24   Thu Feb 14
  21:00 UT     12:00 UT
Bytes: Free   Written
 A:  374496      1084
 D: 2838528         0
```

Figure 31

```
QUIT SDRS

<Enter> to Quit Program
<Esc> for MAIN OPTIONS
```

Figure 32

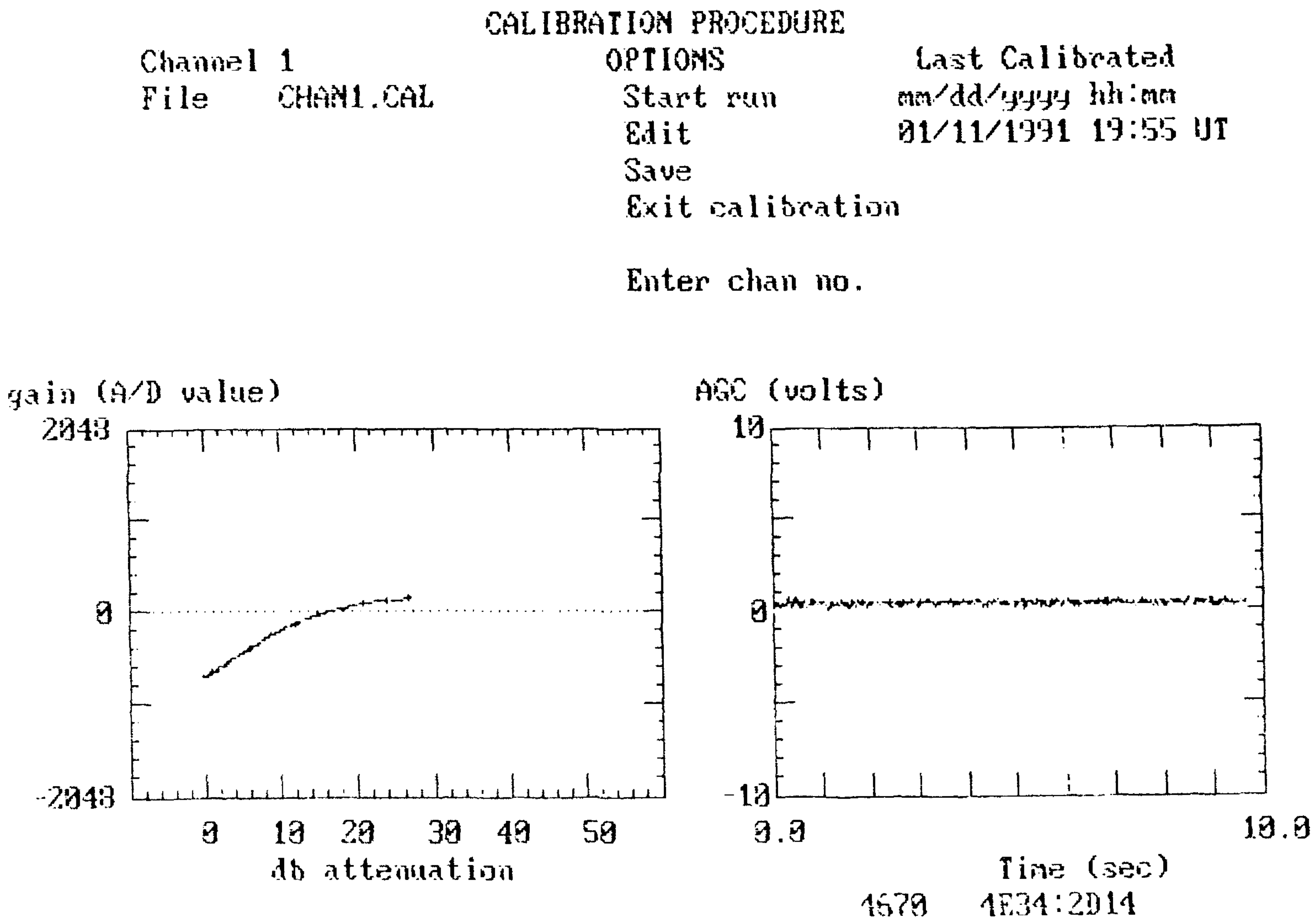

CALIBRATION PROCEDURE
Channel 1
File CHAN1.CAL
OPTIONS
Start run
Edit
Save
Exit calibration
Last Calibrated
mm/dd/yyyy hh:mm
01/11/1991 19:55 UT
Enter chan no.
gain (A/D value)
2048
0
-2048
0 10 20 30 40 50
db attenuation
AGC (volts)
10
0
-10
0.0
10.0
Time (sec)
4670 4E34:2D14

Figure 33

CALIBRATION PROCEDURE

Channel 1
File CHAN1.CAL

OPTIONS
Start run
Edit
Save
Exit calibration

Last Calibrated
mm/dd/yyyy hh:mm
01/11/1991 19:55 UT

Use cursor keys to change option
<Enter> to perform option

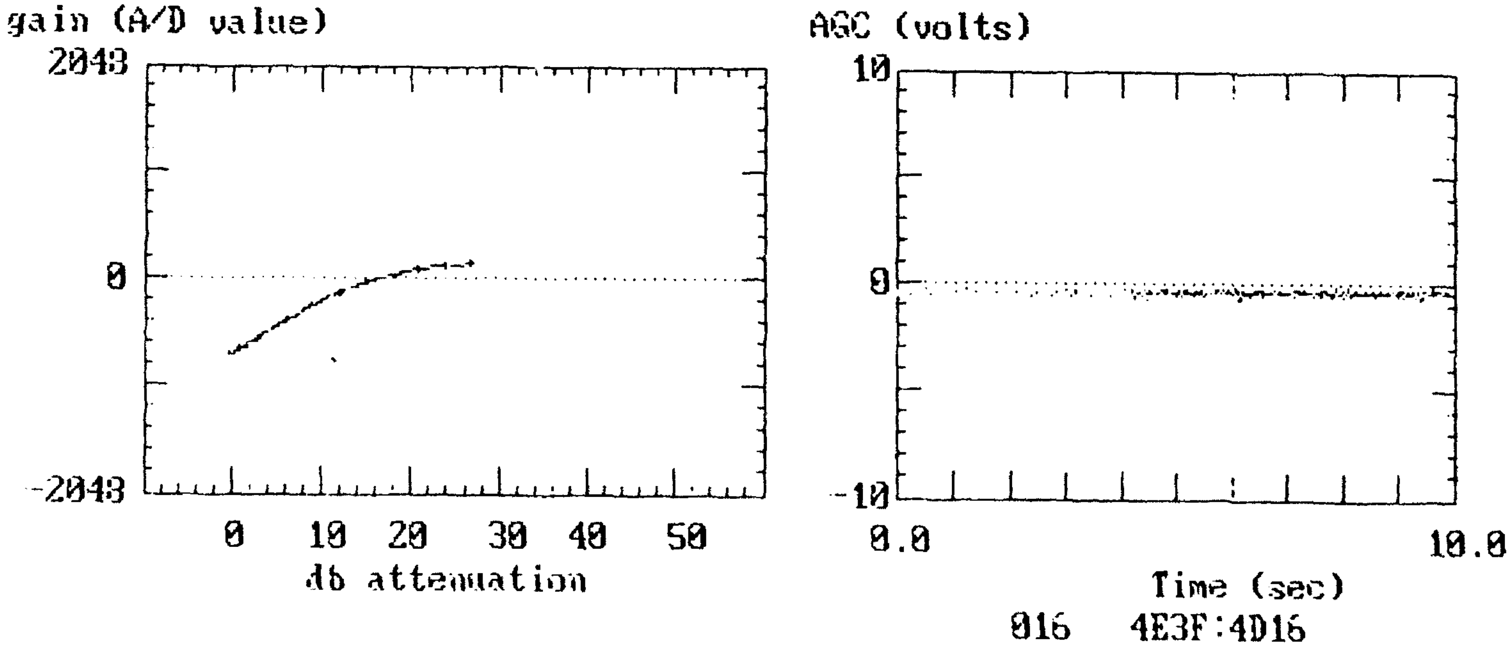


Figure 34

CALIBRATION PROCEDURE

Channel 1
File CHAN1.CAL

OPTIONS
Start run
Edit
Save
Exit calibration

Last Calibrated
mm/dd/yyyy hh:mm
01/11/1991 19:55 UT

Step Mode:FILE
Step size:1 db

db atten.	A/D value	level
0.0	-621	1 of 12

Check atten. level: <Enter>=accept point
q=exit run and plot; <Esc>=abandon run

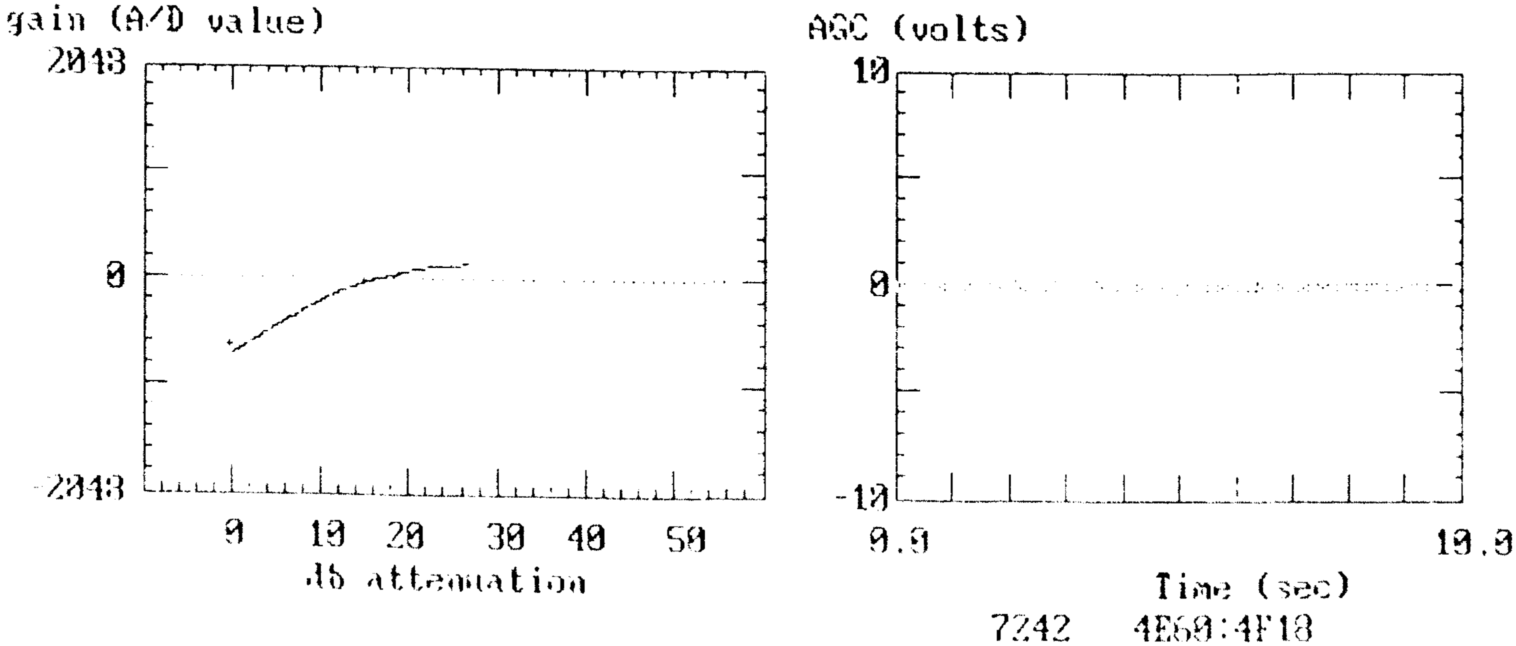


Figure 35

CALIBRATION PROCEDURE

Channel 1
File CHAN1.CAL

OPTIONS
Start run
Edit
Save
Exit calibration

Last Calibrated
mm/dd/yyyy hh:mm
01/11/1991 19:55 UT

Step Mode:FILE
Step size:1 db

db atten.	A/D value	level
1.0	-581	2 of 12

Check atten. level: <Enter>=accept point
q=exit run and plot; <Esc>=abandon run

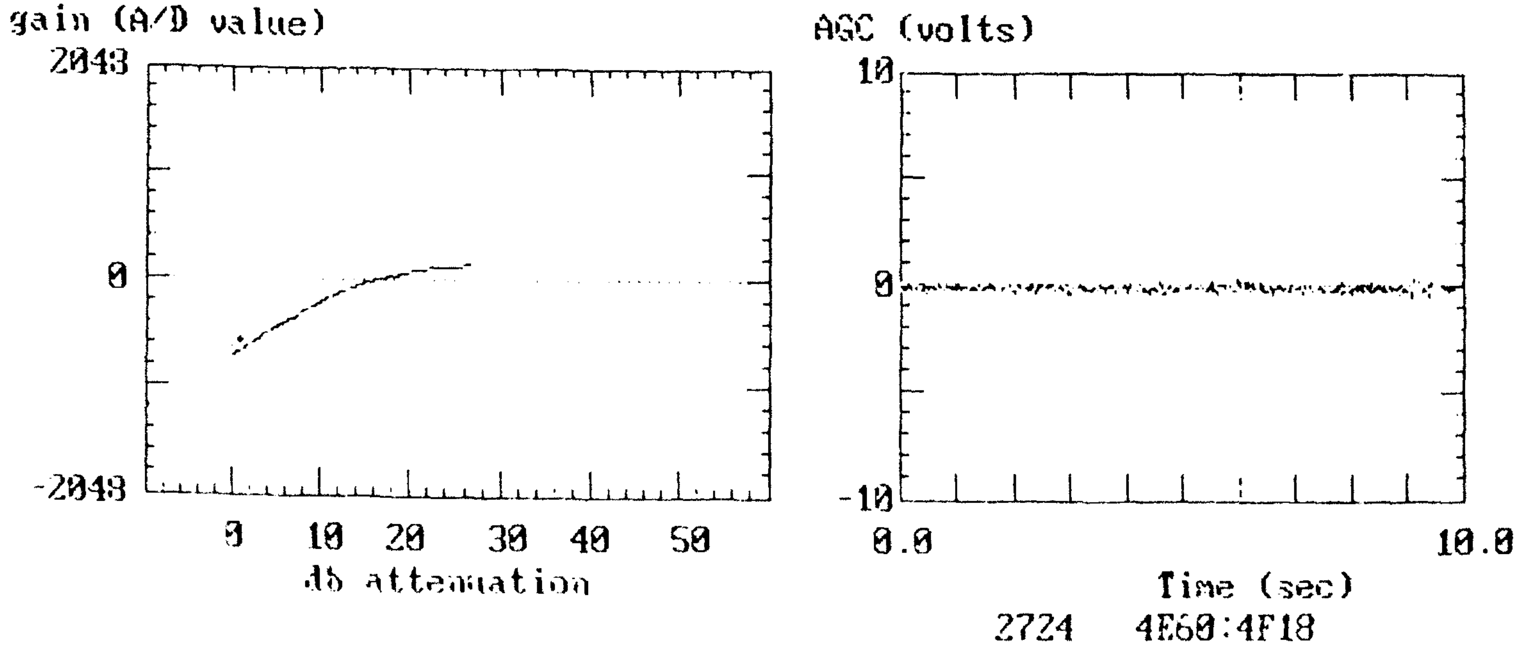


Figure 36

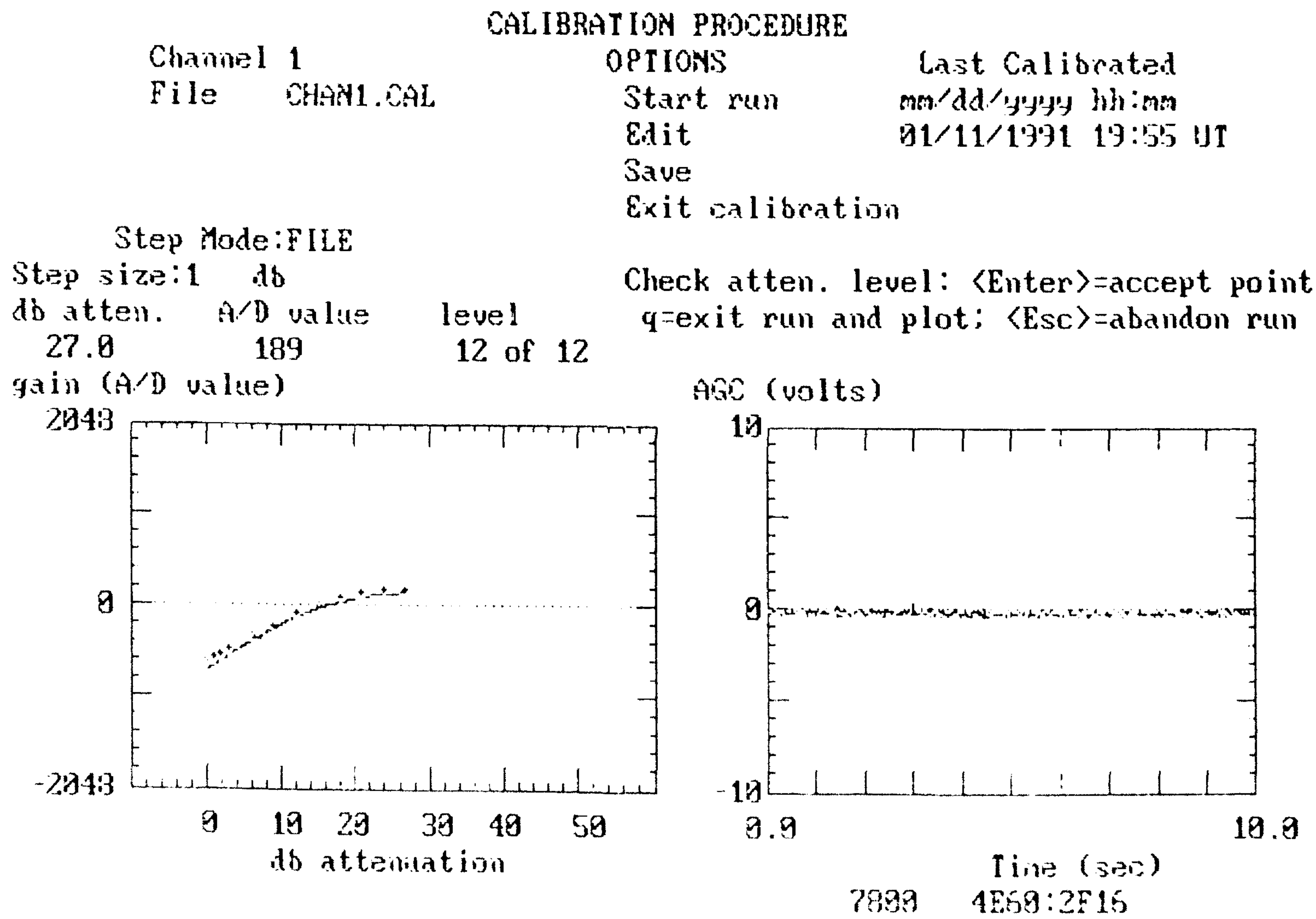

CALIBRATION PROCEDURE
Channel 1
File CHAN1.CAL
OPTIONS
Start run
Edit
Save
Exit calibration
Last Calibrated
mm/dd/yyyy hh:mm
01/11/1991 19:55 UT
Step Mode:FILE
Step size:1 db
db atten. A/D value level
27.0 189 12 of 12
Check atten. level: <Enter>=accept point
q=exit run and plot; <Esc>=abandon run
gain (A/D value)
2048
0
-2048
0 10 20 30 40 50
db attenuation
AGC (volts)
10
0
-10
0.0
10.0
Time (sec)
7800 4E60:2F16

Figure 37

CALIBRATION PROCEDURE

Channel 1
File CHAN1.CAL

OPTIONS
Start run
Edit
Save
Exit calibration

Last Calibrated
mm/dd/yyyy hh:mm
01/11/1991 19:55 UT

Use cursor keys to change option
<Enter> to perform option

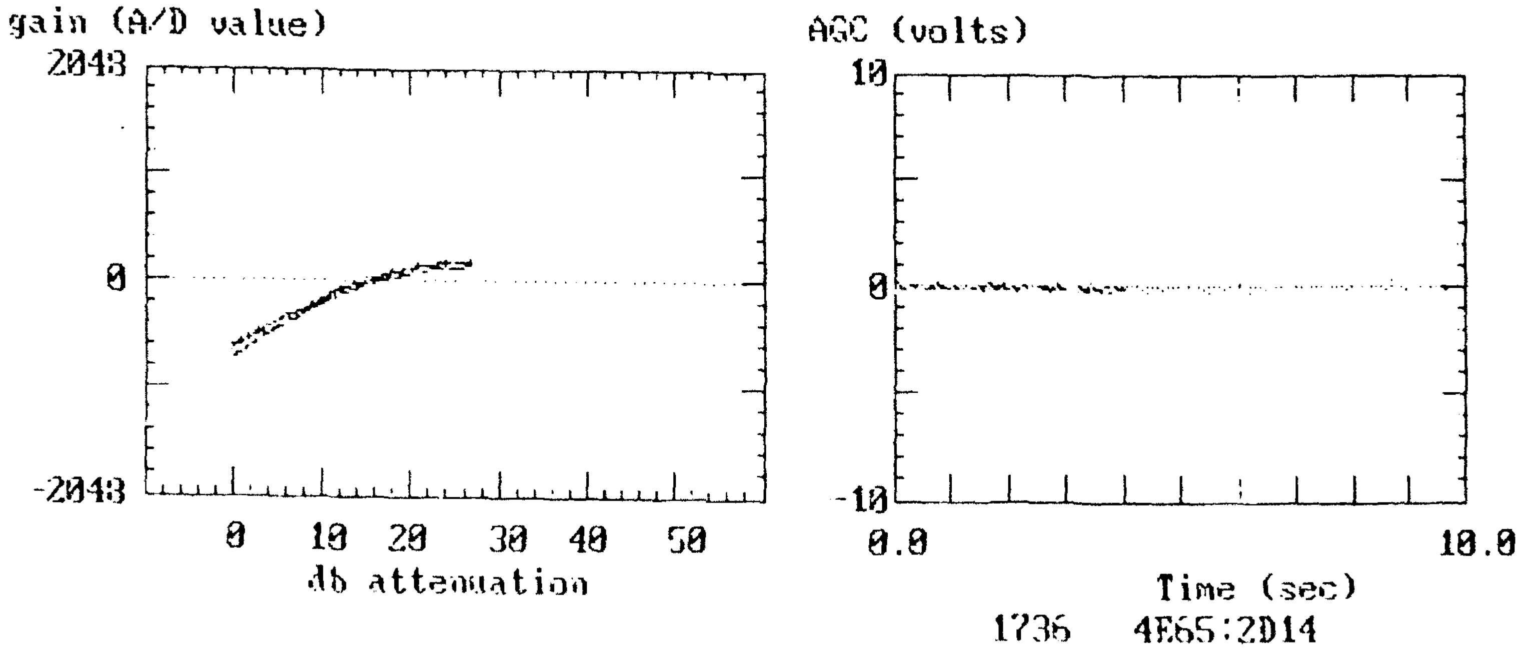


Figure 38

CALIBRATION PROCEDURE

Channel 1 — OPTIONS — Last Calibrated

File LEARN — Start run — mm/dd/yyyy hh:mm

Edit

Save

Exit calibration

Enter chan no.

CHAN1.CAL not found

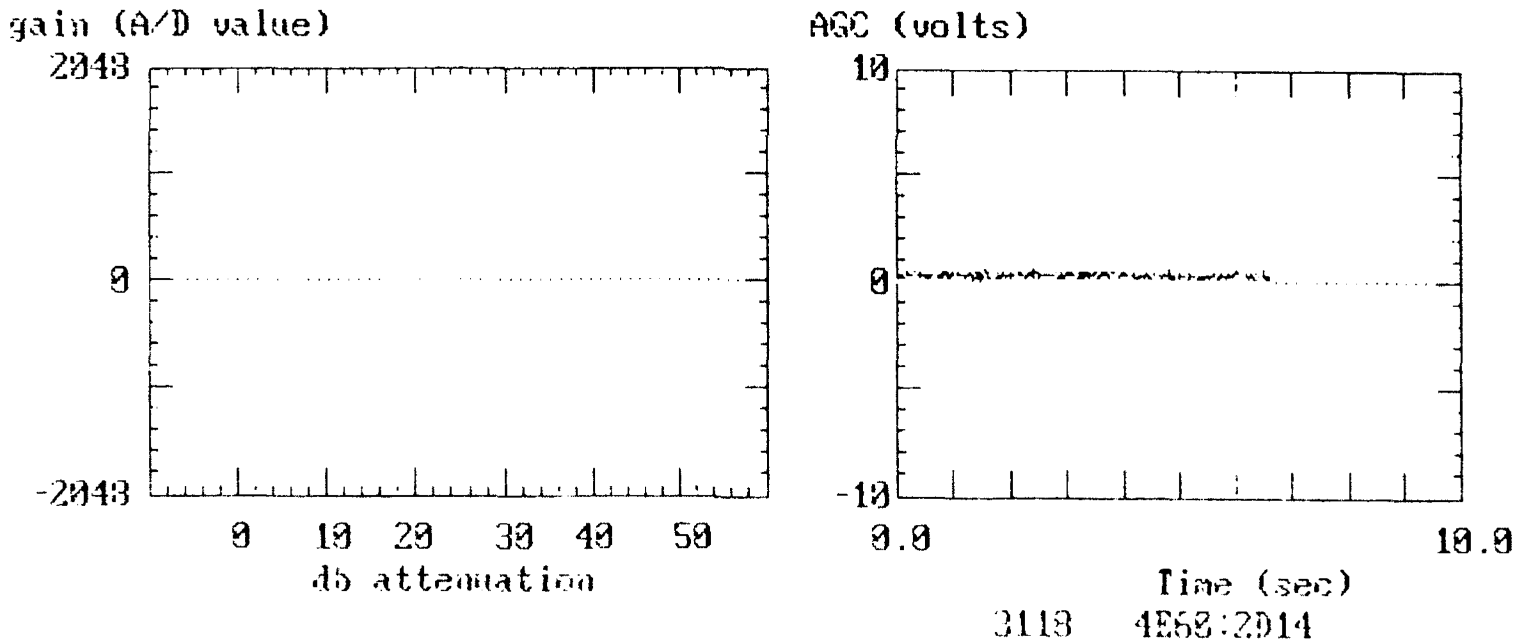


Figure 39

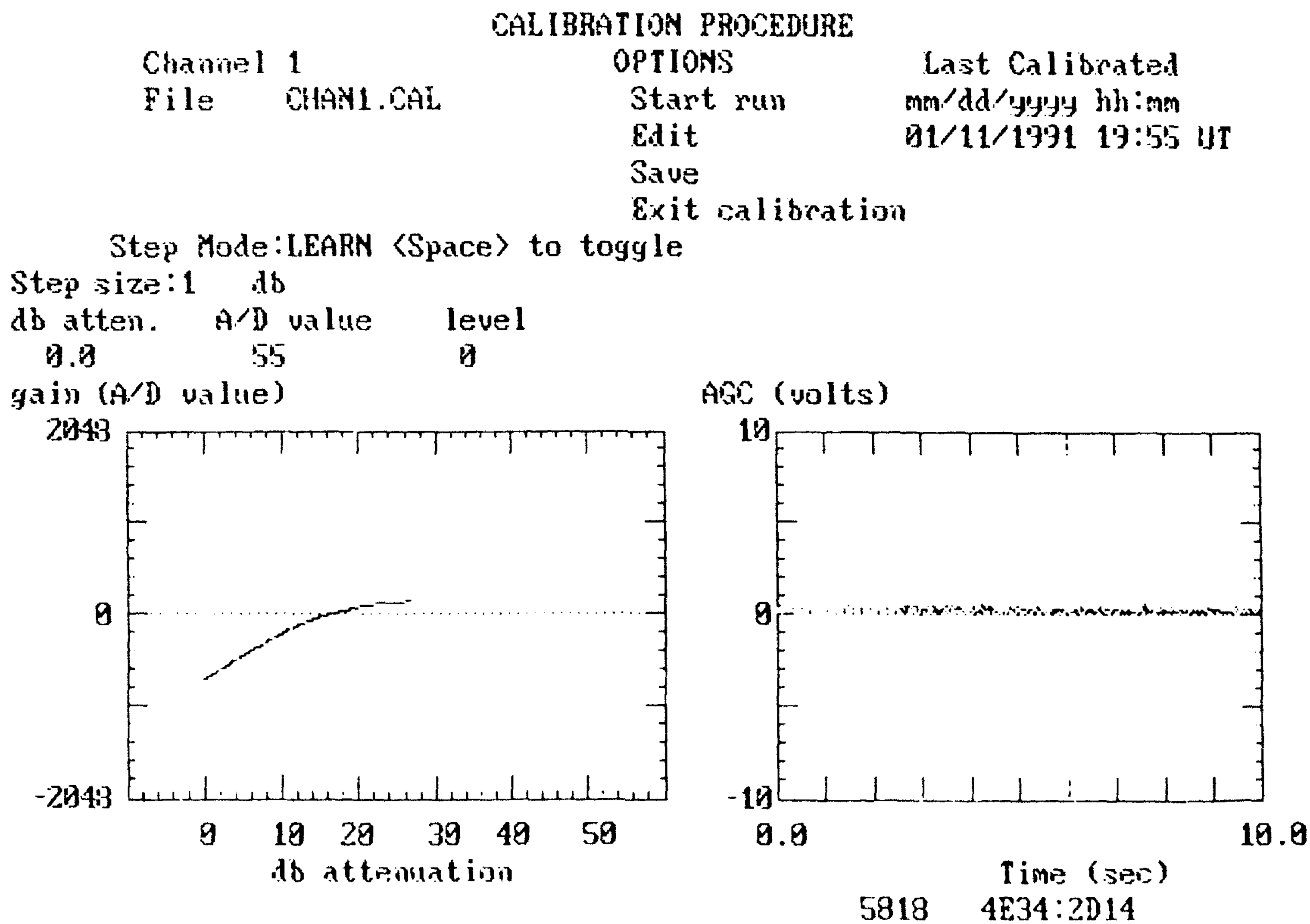

CALIBRATION PROCEDURE
Channel 1
File CHAN1.CAL
OPTIONS
Start run
Edit
Save
Exit calibration
Last Calibrated
mm/dd/yyyy hh:mm
01/11/1991 19:55 UT
Step Mode:LEARN <Space> to toggle
Step size:1 db
db atten. A/D value level
0.0 55 0
gain (A/D value)
2048
0
-2048
0 10 20 30 40 50
db attenuation
AGC (volts)
10
0
-10
0.0
10.0
Time (sec)
5818 4E34:2D14

Figure 40

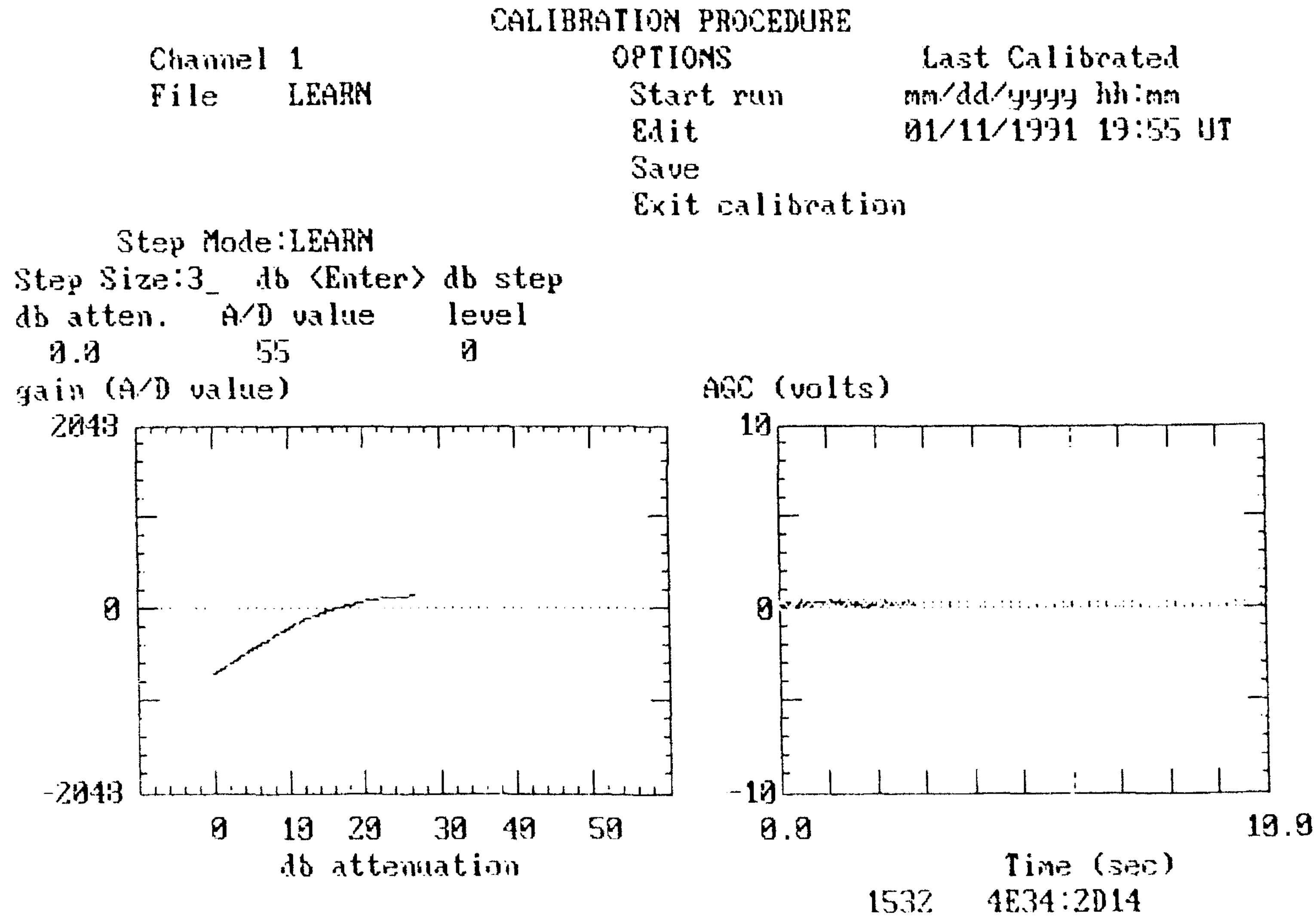

CALIBRATION PROCEDURE
Channel 1
File LEARN
OPTIONS
Start run
Edit
Save
Exit calibration
Last Calibrated
mm/dd/yyyy hh:mm
01/11/1991 19:55 UT
Step Mode:LEARN
Step Size:3_ db <Enter> db step
db atten. A/D value level
0.0 55 0
gain (A/D value)
2048
0
-2048
0 10 20 30 40 50
db attenuation
AGC (volts)
10
0
-10
0.0
10.0
Time (sec)
1532 4E34:2D14

Figure 41

CALIBRATION PROCEDURE

Channel 1
File LEARN

OPTIONS
Start run
Edit
Save
Exit calibration

Last Calibrated
mm/dd/yyyy hh:mm
01/11/1991 19:55 UT

Step Mode:LEARN
Step Size:3 db

db atten.	A/D value	level
0.0	-798	1

Check atten. level: <Enter>=accept point
q=exit run and plot; <Esc>=abandon run

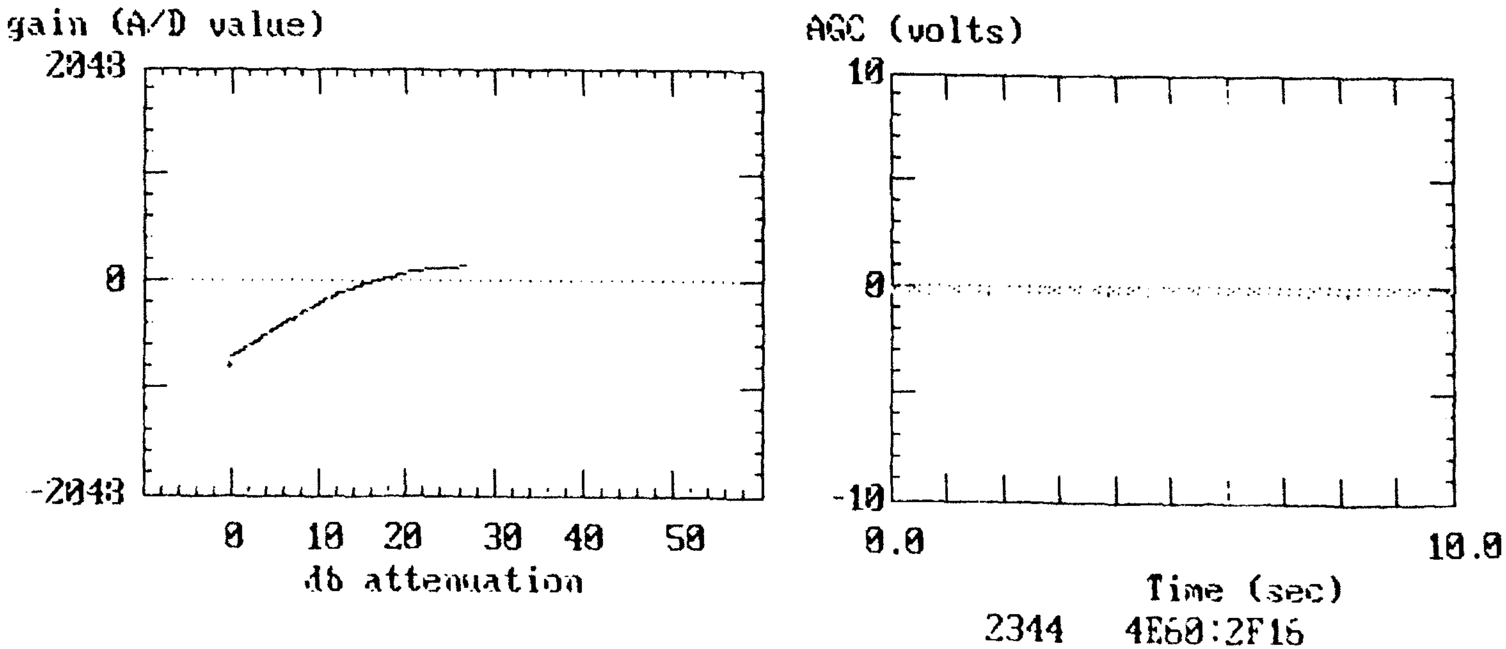


Figure 42

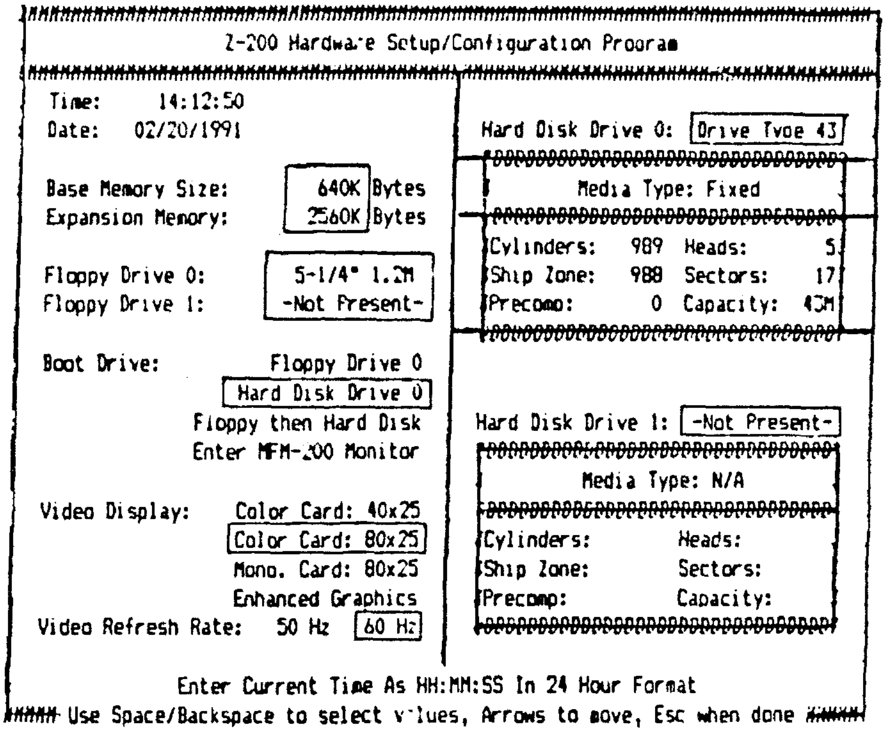
When SDRS does not run properly:

1. Please note the problem and the state of the video screen, if visible; this could help the programmer diagnose errors in the code.
2. Reboot the computer by turning off the power switch on the back of the Z-248 chassis.
3. Wait five to ten seconds.
4. Switch the power back on.
5. The MSDOS operating system will reload and SDRS will restart in the power failure mode. The system should begin collecting and processing data using the current calibration files. It is important to do the cold restart described above; a warm reboot with the <Alt Ctrl Del> key combination may still leave important sections of memory corrupted. Check that the calibration file(s) and processing options are the ones you want.

Indications that SDRS is not working properly:

1. SDRS screen or **CALIBRATION PROCEDURE** screen is corrupted or will not unblank when a key is pressed.
   Action: Reboot.
2. Keyboard does not respond.
   Action: (1) Remember to press <Shift F1> to activate options menu in the **MAIN OPTIONS** window. (2) SDRS is programmed to ignore the keyboard while recording to the floppy disk (**REC** status in **MONITOR** window). Wait until recording is finished. (3) Reboot if (1) and (2) do not apply.
3. Clock or DMA % not advancing in **CLOCK** window.
   Action: Reboot.
4. Processing does not commence (**MONITOR** window) when DMA reaches 100% and starts counting again from 0%.
   Exception: After performing calibrations and returning to the SDRS screen, processing will not begin for a recently calibrated channel until the current 328 second recording cycle is completed and sample 1 of the next recording cycle is collected.
   Action: (1) Run **Processing** options. (2) Reboot if processing options indicated as on are not performed during each 82 second processing cycle.

If floppy drive full flashes on the SDRS screen and a beep is heard every few seconds, (1) replace the floppy disk, (2) run the **Disk** option from the **MAIN OPTIONS** window, and (3) run the **New Disk** option. The beep should cease and bytes written to the A: drive should be reset to 0 in the **STORAGE** window.



To run Hardware Setup/Configuration:

1. Press <Alt Ctrl Enter>.
2. -> appears (Monitor prompt).
3. Type "setup".
4. Press <Enter>.
5. Setup screen (above) appears.
6. Make changes.
7. Press <Esc> when done.
8. Press <Enter> to save changes.
9. Wait for system to reboot.

To boot from floppy disk:

1. Enter Hardware Setup/Configuration and change Boot Drive to floppy drive 0.
2. Put MSDOS system disk 1 into floppy drive before step 8 above.
3. Press <Enter>.
4. Check date and time and press <Enter> until DOS prompt appears (A:\).

```
 Volume in drive C has no label
 Directory of  C:\

COMMAND  COM    25532  10-31-88   2:54p
CONFIG   SYS      134   1-24-91   4:45p
AUTOEXEC BAT       79  11-30-90   6:38p
BIN          <DIR>     11-27-90  10:23p
STARTUP  BAT       79  11-30-90   6:38p
RESTART  BAT       87   2-08-91   2:11p
LOGPRINT EXE     9847   4-12-90   7:05p
LOADSDRS BAT      353   5-07-90   7:42p
SDRS         <DIR>     11-28-90   3:14p
        9 File(s)   5943296 bytes free
```

```
 Volume in drive C has no label
 Directory of  C:\SDRS

.            <DIR>     11-28-90   3:14p
..           <DIR>     11-28-90   3:14p
BIN          <DIR>     11-28-90   3:14p
ADEL     BAT       19   4-03-90   3:03p
SDRS     BAT      655   2-19-91   8:27p
GO       BAT      108  11-07-90   8:33p
        6 File(s)   5943296 bytes free
```

```
 Volume in drive C has no label
 Directory of  C:\SDRS\BIN

.            <DIR>     11-28-90   3:14p
..           <DIR>     11-28-90   3:14p
SDRSD    EXE   211685   2-19-91   8:40p
SETUPJ   EXE   122654   1-24-91   5:13p
EXITPRNT EXE     9057   5-29-90   5:04p
RDDIR    EXE    77026   9-25-90   [illegible]
CHAN1    CAL      134   2-07-91   2:45p
CHAN2    CAL      150   2-07-91   2:47p
CHAN3    CAL      134   1-15-91   7:03p
CHAN4    CAL      134   1-16-91   8:58p
CHAN5    CAL      134   5-07-90   9:19p
UPDATE   INP       27   2-20-91   1:51p
SETUP    INP      129   2-20-91   1:58p
SDRSEXIT BAT       43   2-20-91   1:58p
S4TEMP1           946   2-20-91   1:51p
S4TEMP2           946   2-20-91   1:51p
S4TEMP3           946   2-20-91   1:51p
S4TEMP4           946   2-20-91   1:51p
S4TEMP1  BAK      946  12-05-90   2:13p
SETUP    HLP     3713   9-10-90   6:19p
POLTEMP          1574   2-20-91   1:51p
S4TEMP2  BAK      946  12-05-90   2:13p
S4TEMP3  BAK      946  12-05-90   2:13p
S4TEMP4  BAK      946  12-05-90   2:17p
S4TEMP5  BAK      946  12-03-90   8:13p
POLTEMP  BAK     1574  12-05-90   2:13p
       26 File(s)   5943296 bytes free
```

```
Volume in drive C has no label
Directory of  C:\BIN

.            <DIR>      11-27-90  10:23p
..           <DIR>      11-27-90  10:23p
COMMAND  COM     25532  10-31-88   2:54p
SYTOS    EXE    140523   7-01-88  12:00p
INSTALL  EXE     38211   8-03-88   3:29p
APPEND   EXE      5794   6-08-88   9:52a
ASSIGN   COM      1530   5-11-88   2:10p
CHKDSK   COM     11003   6-09-88   9:34a
DEBUG    COM     15928   4-26-88  10:15a
CONFIGUR COM     38854   8-04-88  10:06a
DSKSETUP COM     33506   8-03-88   4:17p
SELECT   COM      4132   4-06-88   2:37p
EXE2BIN  EXE      3050   4-07-88   7:43a
FIND     EXE      6483   6-07-88   3:15p
PRINT    COM      9491   6-08-88   9:44a
RECOVER  COM      5262   5-06-88   4:01p
SYS      COM      5045   6-14-88   3:45p
FORMAT   COM     12206   6-19-88   4:02p
SHARE    EXE      8608   4-06-88   2:25p
SORT     EXE      1946   4-07-88   7:42a
GRAFTABL COM      6136   4-07-88   7:43a
LABEL    COM      2346   4-07-88   7:43a
JOIN     EXE      9612   4-06-88   2:59p
MORE     COM       282   4-07-88   7:43a
REPLACE  EXE     13234   4-07-88   7:44a
XCOPY    EXE     11216   4-07-88   7:44a
DISKCOMP COM      6236   6-23-88  10:02a
DISKCOPY COM      6762   6-23-88  10:02a
DRIVER   SYS      1215   4-22-88   8:10a
ZCACHE   SYS      5140   9-08-88   3:52p
SHIP     COM       914   4-06-88   3:15p
NOSTACK  COM       170   6-07-88   3:28p
MODE     COM     15440  11-16-88   3:50p
SEARCH   COM      4343   4-06-88   3:15p
APPLY    COM      2165   4-06-88   3:14p
EMM      SYS      7373  11-17-88   2:17p
ATTRIB   EXE     10655   4-07-88   7:42a
EDLIN    COM      7495   6-02-88   2:06p
FC       EXE     15974   4-06-88   2:56p
GRAPHICS COM     13953   6-17-88  10:40a
VDISK    SYS      3121  10-27-88   4:18p
PART     EXE     55807  11-17-88   8:46a
PREP     EXE     24255  11-17-88  11:28a
DETECT   COM     17026   9-14-88  10:19a
BOOTF    COM       174   6-10-88  11:11a
ZSPOOL   COM      3004   4-06-88   3:16p
BACKUP   COM     29114  11-15-88   2:44p
SUBST    EXE     10552   4-06-88   3:01p
TREE     COM      2083   4-06-88   3:15p
COMP     COM      3759   9-10-87   8:40p
MACHINE  COM     24885   6-29-88   3:44p
RESTORE  COM     26480  11-15-88   2:45p
FASTOPEN EXE      3088   4-06-88   2:36p
ANSI     SYS      1647   4-07-88   9:58a
LINK     EXE     65539   3-07-88   5:10a
COUNTRY  SYS     11254   6-21-88  12:52p
KEYB     COM      9209  10-13-88   3:31p
KEYBOARD SYS     19735   6-06-88  10:50a
NLSFUNC  EXE      3029   4-06-88   2:37p
DISPLAY  SYS     11258  11-10-88   3:20p
EGA      CPI     49065   7-24-87  12:00a
LCD      CPI     10752   7-24-87  12:00a
PRINTER  SYS     13559   4-07-88   9:59a
4201     CPI     17089   7-24-87  12:00a
5202     CPI       459   7-24-87  12:00a
ZCOM     EXE     24133  11-03-88  11:16a
GDU      EXE     84699  10-24-88  11:21a
GDUTSR   COM     11407   9-15-88   3:08p
COMPACT  EXE     51005  11-04-88   8:56a
STCONFIG EXE     39719   7-01-88  12:00p
STBATCH  EXE     39239   7-01-88  12:00p
LOGPRINT EXE      9847   4-12-90   7:05p
README   BAT      2788   7-01-88  12:00p
TOSUS    HLP     61820   7-01-88  12:00p
TOSUS    LNG     20603   7-01-88  12:00p
SYCLOCK  EXE     33415   7-01-88  12:00p
CLKUS    HLF      8776   7-01-88  12:00p
CLKUS    LNG      2092   7-01-88  12:00p
QC       EXE    326656   3-07-88   5:10a
MANIFEST     <DIR>      12-07-90   4:16p
BACKUP   LOG       692   2-11-91   4:15p
       81 File(s)   5943296 bytes free
```

Appendix 3. SDRS Source Code

The SDRS program is contained in 26 source code files (.c) and 5 additional files of header information (.h). A list of the file names is given below.

main.c	buffersc.c	cld.c	clock.c	rtclock.c
abort.c	calibd.c	fft.c	gwindowb.c	histo.c
initd.c	mkname.c	pack.c	plotd.c	pwrspect.c
read_inp.c	s4.c	sort_cal.c	sumplot.c	video.c
zero_cro.c	printer.c	interpt.c		

scint.h	config.h	gwindowb.h	video.h	fft.h
histo.h	plotc.h	setup.h	mt.h	s4.h

The source code for main.c begins on the next page.

```
/* Scintillation Data Recording System */
/* main.c */

#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#include <string.h>
#include <graph.h>
#include <math.h>
#include <time.h>
#include <ts.h>
#include <malloc.h>
#include <process.h>
#include <dos.h>

#include "config.h"
#include "fft.h"
#include "gwindowb.h"
#include "histo.h"
#include "plotc.h"
#include "scint.h"
#include "s4.h"
#include "setup.h"
#include "mt.h"
#include "video.."

#define DONE 1
#define FAIL 0
extern GWINDOW *status_window, *statistics_window,
*acqstatus_window,
             *clock_window, *banner_window, *options_window,
             *storage_window, *spec_window, *display_window;
extern CHAN *ipc, *ready;

/* Fuction Prototypes */

extern int _slint;
extern void break_point(char *);

extern int _printer_status( void );
extern float calibrate_scalar( int );
extern void gclock(int, int, int), cld(int, int, int),
sync_clock( void );
extern void warning();
extern void acquire(int, int, int, int, int);
extern void sds_exit(int);
extern void unpack_buffers(int, int, int);
extern int pack12(int *, unsigned int *, int), rdlog();
extern void init(), update();
extern long unsigned *read_up();
extern void start_dma(int);
extern void cosft(float *, int, int);
```

```
extern void grab_screen();
extern char *site, stat_drive;
extern int startdma_flag, startdma_data, new_cal, fwf_cycle,
*data_rec;
extern int restart, bak_prev, blanked, init_done, setup,
fwf_delete;
extern int sample_mod, keyhit, first_chan, nchans, ndma,
ndma_chan, tec_chan;
int sample_mod, setup=FALSE;
CHAN *ready;
extern PROCESS *clock_process, *rtsync_process, *acq_process;
extern PROCESS *dma_process, *cl_process, *err_process,
*init_process;

PROCESS *clock_process=NULL, *rtsync_process=NULL,
*acq_process=NULL;
PROCESS *dma_process=NULL, *cl_process=NULL, *err_process,
*init_process;
extern char *break_test;
extern char *sdrs_dir, *fwf_log;
char *sdrs_dir="c:\\sdrs";
char *fwf_log="c:\\sdrs\\fwf.log";

extern time_t blank_time;
char *break_test;
int key, startdma_flag=FALSE, startdma_data=FALSE, new_cal=TRUE,
fwf_cycle;
int restart=FALSE, bak_prev=DONE, blanked=TRUE, init_done=FALSE;
extern long unsigned *tape_up;
extern long unsigned int dwrit, data_writ[], awrit, stat_len,
spec_len, a_min;
extern long unsigned tec_len, data_len, backup_len, fwf_len;
long unsigned stat_len=16L, spec_len=102L;
long unsigned tec_len=28L, data_len=6148L,
backup_len=25000000L;/*25000000L*/
long unsigned awrit=0L, dwrit=0L, data_writ[5], fwf_len=1000000L;
long unsigned *tape_up, a_min=10000L;
time_t blank_time=1200L;

ts_main(argc, argv) /* called by Time Slicer main() */
 int argc;
 char *argv[];
{
  char start[2], string[40];
  char *blank_str="      ";
  int chan;
  int *display;
  int nsamples = 4096, nmod = 4;
  int ndats, n;
  struct find_t fwf_file;

  tape_up = read_up();
```

```
  awrit = tape_up[0];
  fwf_cycle = (int)tape_up[2];


  if(*argv[1] == 'y')
   setup = TRUE;

  if(*argv[4] == 'y')
   new_cal = TRUE;
  else
   new_cal = FALSE;

  break_test = argv[2];

  if(*argv[6] == 'y')
   {
    restart = TRUE;
   }

  critstart(DOS_CRCLASS);
  if(fwf_cycle > 1 && fwf_cycle < 4)
   {
    if(_dos_findfirst(fwf_log, _A_NORMAL, &fwf_file) == 0)
     {
      if((bak_prev = rdlog()) == FAIL)
       {
     fwf_cycle = fwf_cycle - 1;
       }
     }
   }
  if(_dos_findfirst("d:\\*.dat", _A_NORMAL, &fwf_file) == 0)
   {
    dwrit = fwf_file.size;
    while(_dos_findnext(&fwf_file) == 0)
     dwrit += fwf_file.size;
   }
  critend(DOS_CRCLASS);

    sample_mod = nmod;
/***********************************************************************
*************/
                      /* initialize hardware & software */

  ready = copen("READY", 2*sizeof(char));
  init();
  ndats = nchans * nsamples;
  ghome(banner_window);
  _settextposition(1,2);
  _outtext(site);

  if(*break_test == 'y')
   break_point("dma");
```

```
/* Start multi-threads (tasks) */
/* unpack thread */
  dma_process = spawn("DMA", 0x400, unpack_buffers, ndma,
first_chan, nchans); /* Buffer manager */

  if(*break_test == 'y')
   break_point("clock");

/* Screen clock thread */
  if(*argv[3] == 'y')
   {
    clock_process = spawn("CLOCK", 0x400, gclock, ndma, nsamples,
ndats);
    ptimer((long)(1*SECOND), clock_process);
   }
  if(*break_test == 'y')
   break_point("rtsync");

/* Real time clock synchronization thread */
  if(*argv[3] == 'y')
   {
    rtsync_process = spawn("RTSYNC", 0x400, sync_clock);
    ptimer((long)(30*MINUTE), rtsync_process);
   }

     if(fwf_cycle < 1 || fwf_cycle > 3)  /* nfwf_cycle = 3 in
mkname.c */
      fwf_cycle = 0;

  if(*break_test == 'y')
   break_point("break 1a");

  unblank_screen();


  for (chan=first_chan; chan  < first_chan + nchans; chan++)
  {
  if(*break_test == 'y')
   break_point("acquire");

    allocate_buffer(chan, nsamples);

  stat_text(WHITE, chan, " ACQ", acqstatus_window);
  }

/* acquire thread (processing loop) */
    acq_process = spawn("ACQUIRE", 0xa00, acquire,
        first_chan, nchans, nsamples, nmod, ndma);

/* Wait to recieve message from acquire indicating that system
   is ready to recieve data
*/
```

```
    creadw(ready, (char *)start, sizeof(char));
    cclose(ready);


  if(new_cal == FALSE)
   {
    start_dma(ndma);       /* Begin DMA operation */
    startdma_flag = TRUE;
    startdma_data = TRUE;
   }

  if(*break_test == 'y')
   break_point("cl");
/* cl thread (user interface) */
  cl_process = spawn("CL", 0x800, cld, ndma, first_chan, nchans);

}

void break_point(char *string)
{
    gtext(WHITE,status_window, string);
    while(!kbdscan())
     { guc(); }

    if(kbdget() == 'q')
     sds_exit(FALSE);
    gtext(WHITE,status_window, "        ");
}
```

SUPPLEMENTARY

INFORMATION

PL-TR-92-2262

A New Data Collection System for Ionospheric Modelling and Related Topics

Robert E. Sheehan

Boston College
Department of Physics
Chestnut Hill, MA 02167

15 April 1993

Final Report
7 November 1988 - 30 September 1992

ERRATA

THE TR NUMBER ON THE COVER AND IN BLOCK 10 OF THE REPORT DOCUMENTATION PAGE FOR SUBJECT REPORT SHOULD BE CHANGED FROM PL-TR-92-2262 AND REPLACED WITH PL-TR-92-2263

PHILLIPS LABORATORY
Directorate of Geophysics
AIR FORCE MATERIEL COMMAND
HANSCOM AIR FORCE BASE, MA 01731-3010