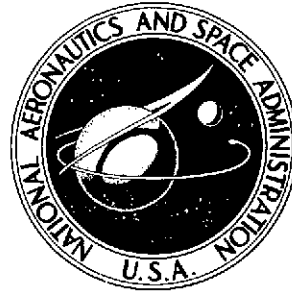


NASA TECHNICAL NOTE



NASA TN D-7970

NASA TN D-7970

(NASA-TN-D-7970) REAL-TIME COMPUTER DATA SYSTEM FOR THE 40- BY 80-FOOT WIND TUNNEL FACILITY AT AMES RESEARCH CENTER (NASA) 39 p HC \$3.75

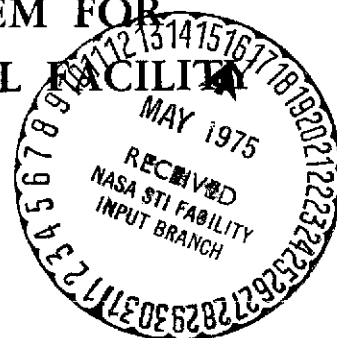
N75-22337

CSSL 14B

Unclas 20252

H1/09

REAL-TIME COMPUTER DATA SYSTEM FOR THE 40- BY 80-FOOT WIND-TUNNEL FACILITY AT AMES RESEARCH CENTER



Joseph M. Cambra and Geno P. Tolari
Ames Research Center
Moffett Field, Calif. 94035



1. Report No. NASA TN D-7970	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle REAL-TIME COMPUTER DATA SYSTEM FOR THE 40- BY 80-FOOT WIND-TUNNEL FACILITY AT AMES RESEARCH CENTER		5. Report Date May 1975	6. Performing Organization Code
		8. Performing Organization Report No. A-5751	10. Work Unit No. 021-06-16-05-00
7. Author(s) Joseph M. Cambra and Geno P. Tolari		11. Contract or Grant No.	13. Type of Report and Period Covered Technical Note
9. Performing Organization Name and Address NASA Ames Research Center and Informatics, Inc. Moffett Field, Calif. 94035 Palo Alto, Calif. 94303		14. Sponsoring Agency Code	
		12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D. C. 20546	
15. Supplementary Notes			
16. Abstract <p>An on-line real-time computer system is installed in the largest wind tunnel in the United States, the 40- by 80-Foot Wind Tunnel facility at Ames Research Center. The system gathers static and dynamic data during aerodynamic tests of the various vertical and short takeoff and landing systems. The computer system monitors aerodynamic forces and moments of periodic and quasi-periodic functions, and it displays and plots computed results in real time. Comprising several off-the-shelf subsystems that are all interconnected, this large interactive real-time-distributed computer system is linked to a large data processing center.</p> <p>The heart of the system is a central processor unit with 32k 24-bit words of core memory, a number of standard peripherals, and several special processors: namely, a dynamic analysis subsystem, a 256-channel PCM-data subsystem and ground station, a 60-channel high-speed data acquisition subsystem, a communication link, and static force and pressure subsystems. This paper presents the background material and operational concepts of the total system. It describes a computer-based system for an operating wind tunnel and also describes, as one of its design goals, the role of the test engineer, who is a vital link in the total system.</p>			
17. Key Words (Suggested by Author(s)) Distributed computer system High speed data gathering Dynamic analysis		18. Distribution Statement Unclassified - Unlimited STAR Category - 09	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 40	22. Price* \$3.75

LIST OF ABBREVIATIONS

ACE	analog conditioning element
ACK	positive acknowledgement
ADC	analog to digital converter
ADCS	Ames Data Communication System
ASCII	American Standard Code for Information Interchange
BCD	binary coded decimal
CRT	cathode ray tube
DAC	digital to analog converter
DAS	Dynamic Analysis System
DMA	direct memory access
DRS	Dynamic Recording System
FIFO	first-in-first-out
FM	frequency modulation
HSDAS	High Speed Data Acquisition System
I/O	input/output
MAP	modal analysis package
NAK	negative acknowledgement
OBS	Onboard System
PC	printed circuit
PCM	pulse code modulation
RTM	real time monitor
TSL	time series language
VCS	Vibration Control System
V/STOL	vertical/short-take-off and landing

REAL-TIME COMPUTER DATA SYSTEM FOR THE 40- BY 80-FOOT WIND-TUNNEL FACILITY AT AMES RESEARCH CENTER

Joseph M. Cambra
Ames Research Center

and

Geno P. Tolari
Informatics, Inc.

SUMMARY

An on-line real-time computer system is installed in the largest wind tunnel in the United States, the 40- by 80-Foot Wind Tunnel facility at Ames Research Center. The system gathers static and dynamic data during aerodynamic tests of the various vertical and short takeoff and landing systems. The computer system monitors aerodynamic forces and moments of periodic and quasi-periodic functions, and it displays and plots computed results in real time. Comprising several off-the-shelf subsystems that are all interconnected, this large interactive real-time-distributed computer system is linked to a large data processing center.

The heart of the system is a central processor unit with 32k 24-bit words of core memory, a number of standard peripherals, and several special processors; namely, a dynamic analysis subsystem, a 256-channel PCM-data subsystem and ground station, a 60-channel high-speed data acquisition subsystem, a communication link, and static force and pressure subsystems. This paper presents the background material and operational concepts of the total system. It describes a computer-based system for an operating wind tunnel and also describes, as one of its design goals, the role of the test engineer, who is a vital link in the total system.

INTRODUCTION

The Ames 40- by 80-Foot Wind Tunnel and its staff are devoted to research and developmental testing which is in the national interest. Frequently the tunnel is used to test powered flight and model vehicles. Figure 1 shows the relative sizes of the tunnel and a model, as illustrated by the person standing next to the model support strut. The size of the model makes it possible to instrument them with complete systems which include various types of transducers. Thus, the complexity of the instrumentation requires an equally complex computer system for gathering and processing the transducer signals. This report discusses a distributed computer system for use in these wind-tunnel investigations.

Instrumentation setup is performed at a model preparation site near the wind tunnel. While the models are being instrumented at the model preparation site, the instrumentation is checked and made ready for use in the wind tunnel. After the model preparation is complete, the instrumented

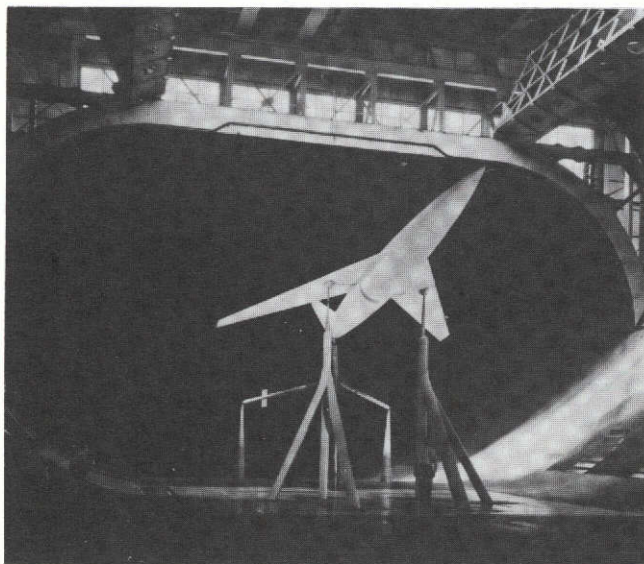


Figure 1.— Looking downstream in the test section (overhead doors partially open).

model is brought into the tunnel and placed on a balance frame that is electrically isolated from commercial electrical ground. The tunnel balance frame is coupled by a set of lever arms to a scale system that forms the basic force-measuring system.

Until 1970, this force system, along with a pressure scanning system and an FM analog recording system, was the only instrumentation that this wind tunnel had for recording static and dynamic data. Since that time, a real-time dynamic data gathering and analysis system for studying stability problems in vertical and short takeoff and landing (V/STOL) systems was installed. This system can: (1) accept and record data from 60 analog channels for bandwidths varying from 1 to 300 Hz; (2) perform real-time harmonic

analysis of two or more channels; (3) determine transfer and cross-correlation functions; (4) have a 256-channel pulse-coded modulation (PCM) system that mounts on the model; (5) compute and plot aerodynamic coefficients using the scale data; and (6) record as many as 50 analog channels continuously during a 2-hour test.

SYSTEM DESCRIPTION

Figure 2 is a simplified block diagram that shows a model under test with various types of transducer outputs. These outputs are connected by twisted, shielded pair cable to a patch board for distribution. The block diagram shows a master computer system, force system, scale system, pressure system, a special instruments system, high-speed data acquisition system (HSDAS), a dynamic analysis system (DAS), a dynamic recording system (DRS), a ground station with an airborne PCM data system, and an operator's console — all interfaced to the master computer. In addition, it shows the master computer connected to a communication subsystem, which is connected to a central data processing center through telephone lines. Also connected to the central data processing center is a remote job entry terminal which is used to communicate with the central data processing center in a time-sharing mode in competition with other remote job entry terminals. The purpose of the communication is to provide the user with greater computing power so that he can solve more complex problems.

SUBSYSTEM SPECIAL CHARACTERISTICS

All the subsystems (fig. 2) have special functional characteristics which overlap and complement each other. The master computer provides control, mass storage, and processing for the system. The HSDAS provides signal conditioning for the DAS and the DRS. In addition, the HSDAS

simultaneously samples and holds 60 analog sources and converts these samples to corresponding digital values. The DAS is the fallback system for the HSDAS and provides the quick-look function during dynamic data gathering. The DRS is an analog recording and playback system. It provides time-code information that is used for data recovery and correlation. The onboard system (OBS) is a combination PCM and ground-station system overlapping the functions of the aging pressure system which it will replace in the future. The pressure system is a scanivalve control system that is capable of monitoring 480 pressures and recording the data on punched cards. The scale system connected to the floating frame is the force system that measures aerodynamic forces exerted on the model. The special instrumentation system is a digital multiplexing subsystem which interfaces digital panel meters, switches, etc., to the master computer. The transducer conditioning system provides power condition to individual gauges. All of these units are connected together through interfaces and twisted, shielded cables that make up a secondary subsystem. The Ames data communications system (ADCS) ties the 40- by 80-foot wind tunnel to the central data processing center at Ames Research Center. This latter tie provides the test engineer with greater computing capability.

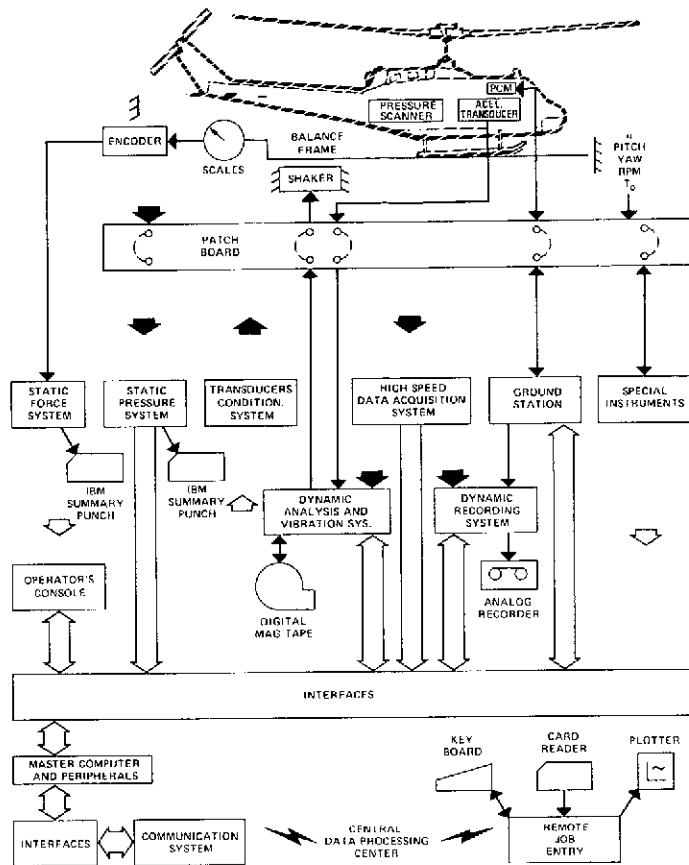


Figure 2.— System hardware, simplified block diagram.

By overlapping the function of several subsystems, we obtained a reliable data gathering system providing for system degradation. Any one or two of the major subsystems, including the master computer, may fail, but data can still be gathered with a reduction in capability. This can be considered an optimal form of system degradation.

DATA SYSTEM OPERATION

To describe the operational aims of this system in a single statement, we can say the system was designed to gather data in a wind-tunnel environment efficiently and accurately by increasing the reliability of data gathering and reducing the operating cost of the wind tunnel. Although the choices of system architecture to achieve these criteria are numerous, we decided in a distributed system because each subsystem could have overlapping features that would act as a backup to some other subsystem. In this way, the tunnel could continue to run and collect data if one of the

subsystems should fail. If the master computer should fail, the remaining subsystems could operate as stand-alone units. This architecture provides a system with two major operating modes:

1. Stand-alone operation (each subsystem operates off-line and in real time as an independent data system).
2. Community operation (all subsystems are connected to a single master computer through a system of interfaces).

In this latter mode, the master computer is the heart of the systems; it performs control functions, raw data storage, data processing, and interactive communications with the user (fig. 3). All subsystems work together on-line to function as a single distributed real-time system.

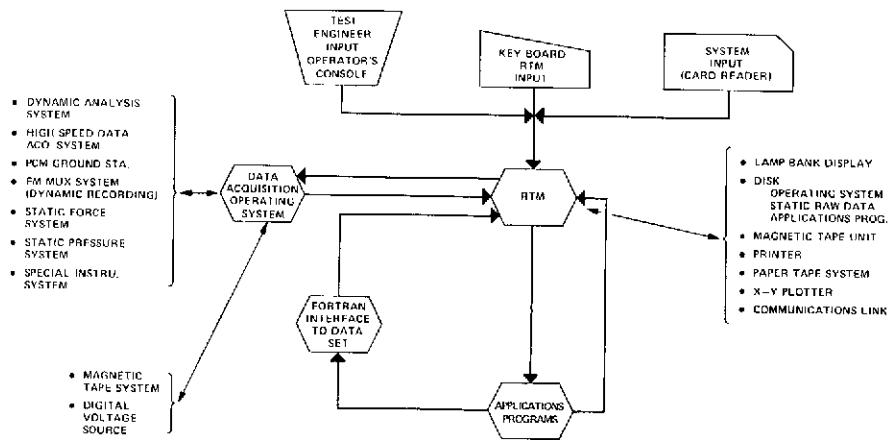


Figure 3.— System software, simplified block diagram.

There are two basic data gathering modes in the 40- by 80-foot wind-tunnel system. These are the real-time and playback modes. In the real-time mode, the system gathers and processes the data as it occurs. During the playback mode (fig. 4), the system recovers data from analog magnetic tape that was previously recorded by the dynamic recording subsystem.

On-Line Operation

On-line operation begins when the test engineer initializes the system (fig. 5). He does this by mounting a set of parameter initialization cards in the card reader hopper and entering the on-line codes into the computer through a set of program selector switches at the operator's control console (fig. 6). After accepting the on-line code, the master computer causes the cards to be read and verifies that the required on-line systems are connected, ready to go, and calibrated under computer control. With the initialization mode complete, the system resumes an idle state waiting for further inputs from either the card reader or control console. (A detailed description of the initialization is given in the system software section.)

After the test engineer loads the FORTRAN applications programs, he can initiate the real-time mode of operation. He does this by actuating the RECORD pushbutton on the operator's console master control panel (fig. 7) to start the first record session (zero test conditions). Upon

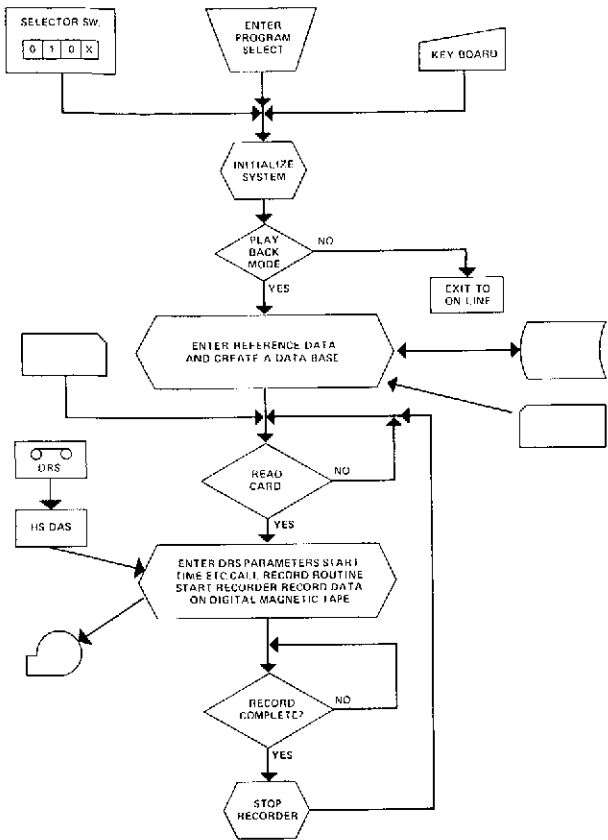


Figure 4.-- Playback mode flow diagram.

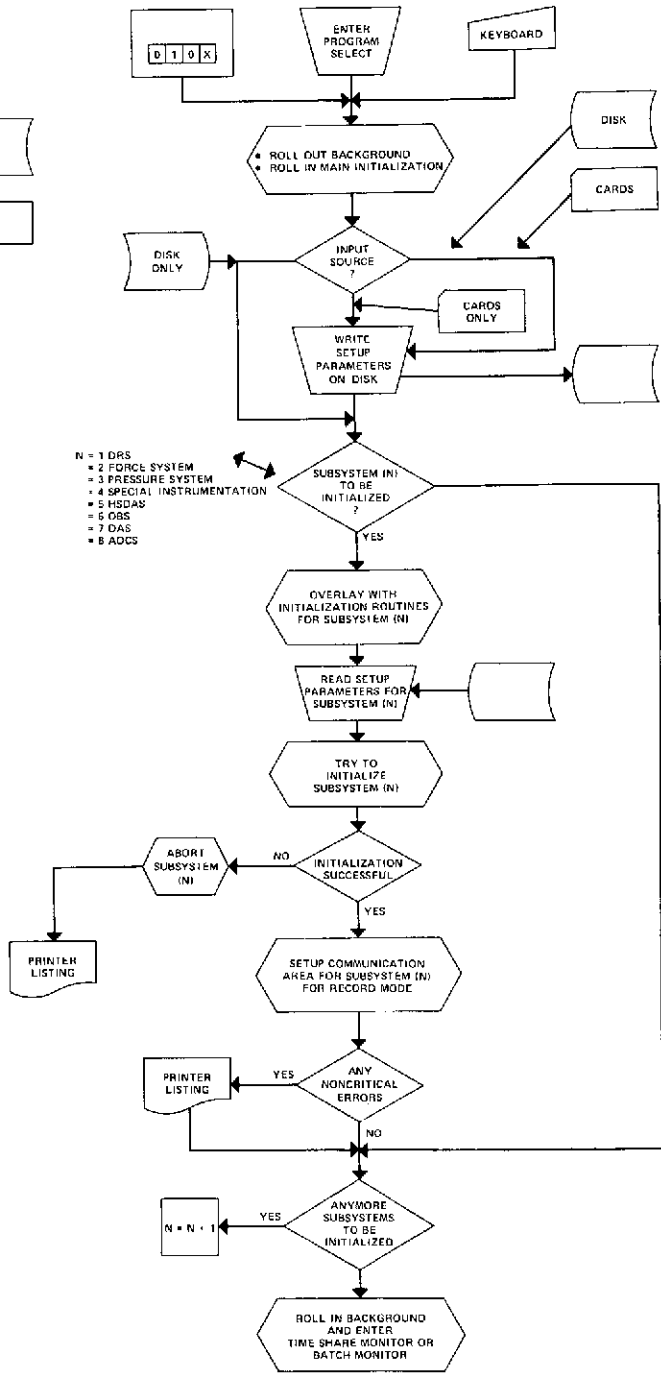


Figure 5.-- Initialization flow diagram.

ORIGINAL PAGE IS OF POOR QUALITY

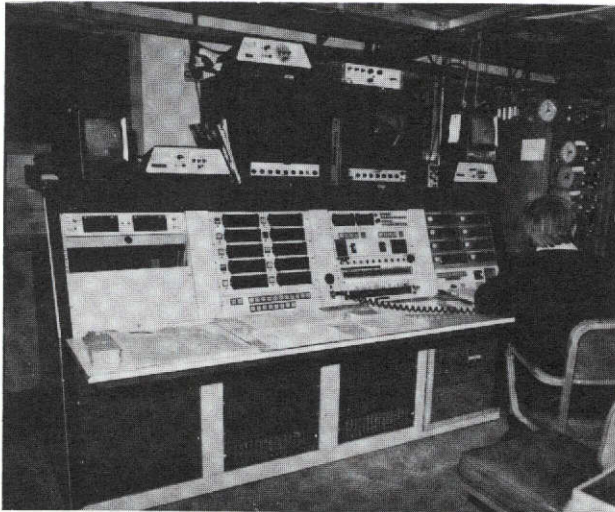


Figure 6.— Operator's system control console.

receiving the record interrupt (fig. 8), the system records the test zeros on magnetic tape and also stores these static-data zeros on disk for later use by the user applications programs. When the record session is completed, the system loads the monitor routines and the user applications programs into core from disk (fig. 9). This done, the system begins real-time on-line operation by gathering data from the scales and the special instrument system and displaying the processed results on a lamp bank for monitoring. The system repeats the process of gathering scale and instrument data and displaying results until it receives another record request.

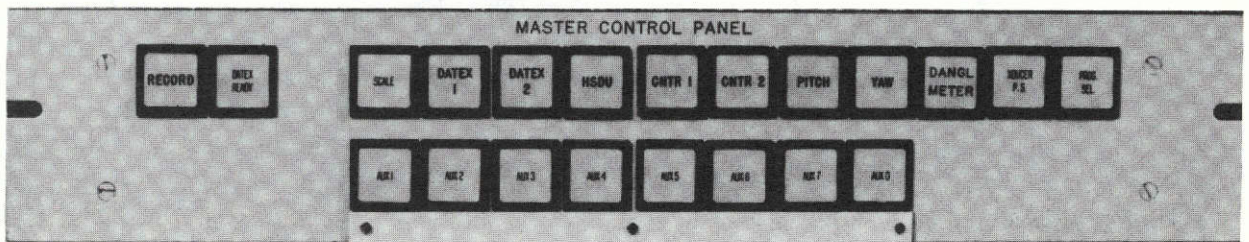


Figure.7.— Operator's console master control panel.

During the monitoring period tunnel conditions are changed, and, after the tunnel conditions stabilize, the test engineer initiates a new record request. When this request is acknowledged, the current data from the special instruments is frozen in memory. When the other sub-systems have completed the record session, the master computer repeats the mass storage cycle and reloads the monitor routines and the user applications programs. This sequence of events, going from the monitor to the record and back to the monitor mode, is repeated each time a record request is initiated.

In the monitor mode, the DAS and OBS are also displaying selected data. In the case of the DAS, the test engineer can interact with it independent of the master computer. For example, the test engineer can: select any 1 of 32 signal sources; perform a Fourier transform, an autospectrum analysis, an average; and display the results — all in real time. In addition, he can perform a cross-spectrum analysis between any 2 of the 32 channels, or a modal analysis on any of the channels. The modal analysis software (MAP) for the DAS is a special package (ref. 1) that is used in determining resonances and model stability in the dynamic system.

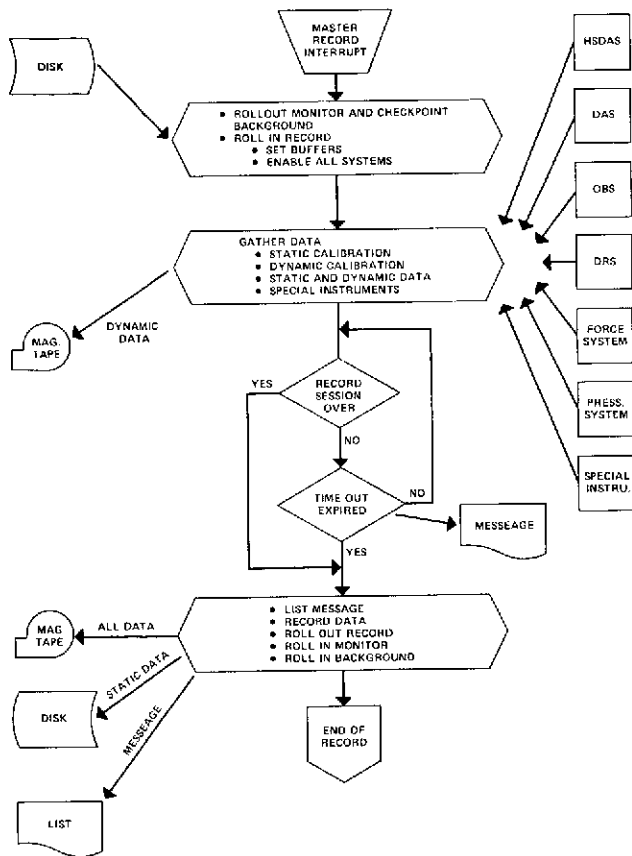


Figure 8.— Record mode flow diagram.

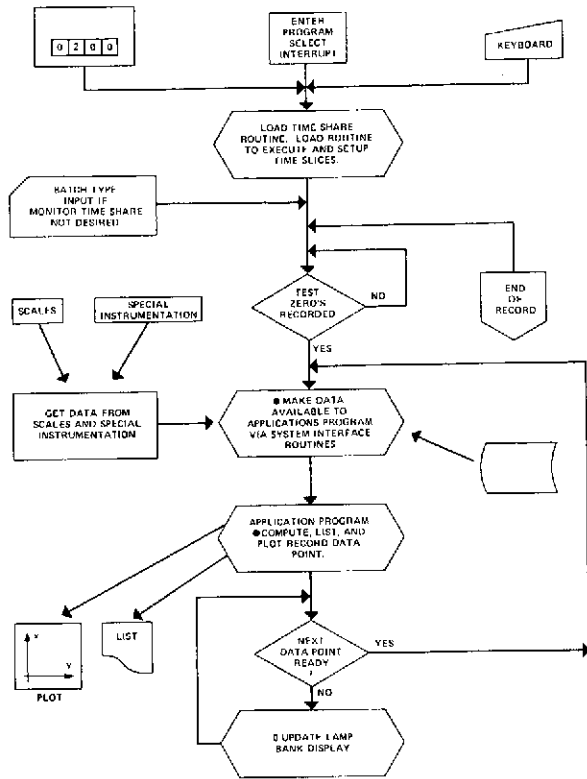


Figure 9.— Monitor mode flow diagram.

SOFTWARE

Figure 3 is a simplified block diagram of the 40- by 80-foot wind-tunnel software. This software provides the interface (via setup cards) between the wind-tunnel engineer and data-acquisition hardware. The setup cards contain parameters that specify his test configuration and the characteristics of the hardware being used. The software verifies that all hardware components are operating correctly and then, under command of the user, it controls the data-acquisition process by logically connecting and disconnecting the hardware as required.

To provide a complete system, the software is partitioned into three operational modes: (1) for system setup, (2) for wind-tunnel data acquisition, and (3) for monitoring of wind-tunnel conditions while they are changing and for interfacing users' programs and the data. The third mode includes various computer software modifications that enable users to simultaneously gather the data, compute aerodynamic forces and coefficients, and plot the results of the computations. The remainder of this section describes the software concerned with each of the operational modes in more detail.

In general, our philosophy has been to design an interactive software system for the test engineer. The dialogue between the computer system and the engineer is simple because the

language design uses key words familiar to him. It provides an easy method for setting up the various hardware subsystems. It generates meaningful messages whenever setup errors occur, and it is considerate. It recognizes that the test engineer is interested in aerodynamic phenomena rather than hardware and software sophistication. The software is interactive, and it guides the engineer during subsystem setup. It forgives him for mistakes except those that would prohibit him from properly interpreting data and, in these cases, the initialization process aborts the subsystem setup procedure.

Setup Software

The setup software (fig. 5) comprises a group of subroutines which collectively control the setting up of the various hardware components. All of the setup subroutines are coded in FORTRAN except those interrupt handlers communicating directly with the hardware. The setup subroutines bring the various hardware subsystems online so that they may participate in monitoring, recording, and/or data analysis. The system is brought on-line by: (1) specifying the characteristics of the system, (2) verifying the operation of the system, and (3) verifying that the data from the various hardware subsystems are within a specified tolerance.

Specifying system characteristics— The system characteristics must be specified whenever the system can be configured in more than one way. For example, the test engineer may select from 1 to 60 channels over which to input data from the HSDAS. He may also select any 1 of 8 gains and any 1 of 11 bandwidths for each channel, as well as whether the HSDAS hardware will be controlled by the computer clock or some external clock.

The test engineer must initially specify the system parameter through setup cards. The setup cards provide the user with a keyword input capability. This technique allows the test engineer to use default (preset) parameters where applicable, and it provides meaningful parameter names whenever a default value must be reset. The keyword input also provides an easy way of identifying errors and generating meaningful error messages. The input parameters are used to select active subsystems, data channels, and electrical characteristics. The setup cards can be prepared in advance to reduce the demands made upon the wind-tunnel engineer during the actual test.

Once the system setup parameters have been initially specified, the user may reinitialize as often as he wishes from the disk. The software provides for copying the setup cards onto disk, and then reading them from the disk for initialization purposes. The software also provides for initializing from both disk and cards, where the cards act as an override to the disk parameters. In this manner, the wind-tunnel engineer will have to read in only parameters that he desires to change.

The basic philosophy incorporated here is to allow the wind-tunnel engineer to have complete control for changing the hardware setup and to do so with as little effort as possible after the initial setup.

Verifying system operation— The second action taken by the initialization routine is to verify the subsystem operation. This typically consists of verifying that a subsystem is powered up, and then verifying that data can be received from the subsystem. In some cases, such as the special-instrument subsystem, this is accomplished by simply waiting for a specific interrupt to occur. At the other extreme, the HSDAS subsystem must be setup by the computer, and then the data

gathered must be examined to confirm that proper communication exists between the HSDAS and the computer.

Whenever a subsystem is not operating properly, the test engineer is alerted to it by an error message. For example, if the setup cards requested 10 channels of data per data scan for the HSDAS, but the last channel indicator indicates that there are 20 channels per scan, the following message will be printed: "10 CHANNELS PER DATA SCAN REQUESTED, BUT ACTUAL HARDWARE SET TO 20 CHANNELS PER SCAN. PLEASE SET CORRECTLY." The software will repeat this message every 30 sec for 2 min before aborting the HSDAS initialization. If, during the 2 min, the software determines that the error has been corrected, it will continue to verify proper system operation.

The software will not abort the initialization of a subsystem unless a catastrophic error exists. For example, if the setup card specified a gain of 10 for the first HSDAS channel and the actual gain setting was 2, the test engineer would be informed of the error but initialization would continue.

That the system handles the test engineer's needs with prime consideration is ably demonstrated in the case of error detection in sampling rates: It is assumed that the test engineer is not a student of sampling theory but does know how many channels of data he would like to study, the maximum signal bandwidth, and the percentage of foldover error he considers to be acceptable. Given these characteristics, the software computes what the required sampling rate should be. In the event the required sampling rate exceeds the total system throughput, the software will suggest an alternate setup. It should be noted that, although an alternate setup is suggested, the initialization process continues with the current setup. As a specific example, if the test engineer requested 60 channels at 1-percent foldover error and the maximum frequency to be studied was 300 Hz, the software would proceed as follows: First, all bandwidths would be examined to determine that at least one channel had a bandwidth setting of 300 Hz or more. If the maximum bandwidth setting was less than 300 Hz, say 200 Hz, the HSDAS initialization would be aborted, since it would be impossible to study a 300-Hz signal through a 200-Hz filter setting. If at least one channel had a bandwidth setting of 300 Hz or more, the following message would appear:

```
***ERROR CANNOT SAMPLE 60 CHANNELS WITH ONLY 1% ERROR, WILL SAMPLE BUT  
ERROR IS 48%
```

```
IF YOU WISH TO SAMPLE WITH FOLDOVER ERROR OF 1% YOU MUST REINITIALIZE  
  
THE NUMBER OF CHANNELS, BANDWIDTH SETTINGS AND MAXIMUM FREQUENCY  
PARAMETERS MUST THEN BE RESPECIFIED IN ONE OF THE FOLLOWING WAYS.
```

```
THE MAXIMUM FREQUENCY THAT CAN BE STUDIED WITH 60 CHANNELS AND FOLD-  
OVER ERROR 1% IS 100 HZ
```

```
26 CHANNELS CAN BE STUDIED AT BANDWIDTHS SETTING OF 300 HZ AND FOLD-  
OVER ERROR OF 1%
```

The general philosophy adopted here is to identify all errors that will affect system operation and to notify the wind-tunnel engineer of these. In the cases where the errors are so catastrophic

that proper operation is impossible, the wind-tunnel engineer will be informed and given some reasonable time to correct the errors. If he has not corrected the errors during this time, the initialization is aborted. Whenever he does correct the problem in the required time, initialization continues. In the cases where errors do not seriously affect system operation, the test engineer is informed and the initialization process continues.

Verifying hardware data input tolerances— The third action verifies that the data gathered by the various hardware subsystems are within specific tolerances. This typically consists of examining data gathered against known or expected values. In the case of force and pressure data, it consists of examining the transmitted card image, column by column, against the backup punched card. For the HSDAS, the digital-to-analog converter (DAC) is used to input a known dc signal to each HSDAS channel. Data are then input from each channel and compared with the output digital value, taking into account the effect of the channel and ADC gain settings. The compared results are listed on a channel-by-channel basis, indicating the percentage full-scale digital input value and whether or not each channel is within tolerance. For the HSDAS, there are typically three standardization signals for each channel, one for minus 80-percent full scale, one for zero offset, and one for plus 80-percent full scale.

Data Acquisition Software

After system setup is complete and when the experimenter decides that the test conditions in the tunnel are suitable for data acquisition, he initiates a record interrupt by activating the master record switch on the master control panel (fig. 7). Whenever the software (fig. 8) receives this interrupt, all background processing is halted and check-pointed to disk. The record mode software is then loaded so that the data acquisition process (record session) can begin.

The data acquisition process consists of gathering both static and dynamic data. Dynamic data may be gathered by either the HSDAS or OBS. Static data may be gathered by the OBS, force system, pressure system, and special instrumentation system. In addition, the DAS may transmit results of its analysis of dynamic data.

The events which occur as a result of the master record interrupt are summarized below:

1. The applications program in the background is halted, and check-pointed to disk.
2. After the record mode software is loaded, the systems that were requested during initialization are examined to verify that they are on-line. Once verified, all necessary interrupts are enabled.
3. The amount of calibration data and dynamic data specified at initialization time is gathered from the dynamic data source and written onto digital magnetic tape.
4. The onboard system, when active as a static data source, is commanded to take and store in its core memory the amount of calibration and static data determined during initialization.
5. A number of samples of static data, as determined at initialization, is gathered from the other static sources and retained in computer memory.

6. When the data acquisition process is complete, the information from the onboard system is recorded on digital magnetic tape.

7. The static data from the other active sources are recorded on both disk and magnetic tape.

8. The experimenter is informed that the data acquisition process for that test point is complete.

9. The check-pointed user application program is copied into computer memory and execution continues from the point of interrupt.

Since the data acquisition process places a greater demand on the computer's resources than any other on-line activity, some restrictions are imposed during this phase of operation. The restrictions are identified below:

- There can be only one source of dynamic data.
- One magnetic tape unit is unavailable to the background user.
- The maximum system throughput rate is approximately 40 kHz, and it is dictated by the digital magnetic tape recording speed.
- Timing considerations dictate that most of the data gathering subsystem and the digital magnetic tape interrupts be at a higher priority than the standard real-time monitor (RTM) software.

At the beginning of the record session, all subsystems which are to participate in the test are examined to determine that they are truly on-line. Whenever a subsystem that was marked as active by the initialization process is off-line, the data point proceeds, but with that subsystem excluded from the test. The experimenter is notified of the condition by means of an explanatory message. If the experimenter wishes to bring that subsystem on-line, he must, in general, reinitialize it. The only exception is the special instrumentation subsystem, in which case, each channel can be turned on without reinitialization.

The data acquisition software comprises a number of routines each of which is directly or indirectly entered as a result of an interrupt. Rather than detailing each of these routines and its specific function, the remainder of this section discusses what happens functionally for each hardware subsystem during the data acquisition process.

High-speed data acquisition process— Whenever the HSDAS is in the test, the gains and bandwidths for each of the active channels are input and written to digital magnetic tape. The computer settable switches are then set, and those that are not settable but are computer readable are examined to verify that they have not changed from the initial settings. Whenever the remote/local or internal/external clock (located on the computer interface chassis) switch settings have changed, the HSDAS is not allowed to participate in the test until they are set properly. When the system is operating with an external scan, the rpm value input from the special instrumentation system is

examined. If it exceeds the value determined as the maximum allowable rpm at initialization, the HSDAS is not allowed to participate in the test. When the maximum rpm is exceeded, the data input rate will exceed the maximum system throughput rate. Whenever these or other error conditions occur, the experimenter is notified via an explanatory message.

Assuming that the HSDAS is properly set up, the high-speed data acquisition process will proceed. The software continues the data gathering process by switching three dc-calibration signals into the HSDAS auxiliary inputs and then recording the calibration data on digital magnetic tape. The three dc signals provide an up-to-date static calibration of the input to the HSDAS as wind-tunnel conditions change. The HSDAS is dynamically calibrated under software control by switching in both a signal from a function generator as the transducer excitation and a shunt calibration resistor as an off-set. The number of calibration blocks of dynamic calibration determined at initialization is then input and written to digital magnetic tape. Finally, the actual dynamic data acquisition process for the HSDAS is initiated. This is accomplished by inputting (from the HSDAS) the number of dynamic data blocks determined at initialization and outputting these blocks to digital magnetic tape. The process of inputting dynamic data and writing it to tape requires chaining and buffer address swapping to accomplish maximum system throughput rate. These techniques allow for extending the maximum throughput rate whenever a faster recording device can be made available.

Onboard system process— The OBS may participate in the test as either a dynamic data source or a static data source, or both. For convenience of discussion, assume that the OBS is participating as both a dynamic and a static data source. The software first verifies that proper communication exists between the two systems. In the event that communication is impossible or erroneous, the OBS is not allowed to participate in the test, and the experimenter is notified of the problem by an informative message. When proper communication is established, the OBS is commanded to acquire and transmit two static calibration scans of data. Each scan contains a static calibration for all channels (dynamic and static) that are active in the test. These two scans of data are written on the digital magnetic tape. When the static calibration is complete, the dynamic channels are dynamically calibrated. This is accomplished in the same manner as the HSDAS dynamic calibration. The number of scans of dynamic calibration data determined at system initialization are input and written on digital magnetic tape. After this is complete, the OBS is commanded to transmit dynamic data. These data are also recorded on magnetic tape using the data-chaining and buffer-address-swapping techniques. Finally, the OBS is commanded to acquire the number of static scans of data at the time interval determined during system initialization. Each of these scans contains only those channels specified to be active static data channels. At the proper time, the OBS is commanded to transmit the static scans to the master computer so that they also can be written on digital magnetic tape. This ends the data acquisition process for the OBS for that data point.

Force, pressure, special instrumentation processes— The data acquisition process for the remaining static data sources is essentially the same. The force-system and pressure-system hardware are examined to verify that they are on-line. In the event that either system's hardware is not, that subsystem is not allowed to participate in the test, and the experimenter is notified of the condition via an error message. If both systems are active, the software enables each of the interrupts. When an interrupt for either one occurs, the data from that system are input and stored in the computer memory. If the required number of card images for each subsystem has been input and all other data acquisition is complete, the data from each subsystem are written on digital magnetic tape. However, if the number of interrupts received from either subsystem is less than the number of card

images specified by subsystem initialization, the experimenter will be notified. The error message states the number of card images actually input and the number of card images expected. The record session is complete after the most recent special instrumentation readings are written to digital magnetic tape.

Monitor Software

The software for this phase of the system operation is divided into two sections. The first section monitors wind-tunnel conditions while they are changing and formats the data so that the application program can compute the results in real time. The second section handles additions to, and modifications of, the existing computer system software (RTM) to expand the computer capabilities.

Monitoring— The monitor mode software (fig. 9) provides the experimenter with the means of observing selected parameters on lamp banks. He may observe computed aerodynamic coefficients, forces, and various temperatures while the tunnel conditions are changing. Data from the scales digitizing system automatically updates the scales lamp-bank displays every 1.25 sec. Other lamp banks are updated by a background application program which is resident during the monitor mode. Data which the program needs for computing aerodynamic coefficients are strobed into the computer from the scales digitizing system and the special instrumentation system. Monitoring allows the experimenter to determine when to record data and whether any hazardous static data conditions are being generated. Most importantly, it allows him to conduct the test more efficiently, recording only significant results, and recognizing when interesting results occur.

The majority of the time that the tunnel system is on-line, the master computer operates in the monitor mode. In the monitor mode, system routines input data from the following sources: (1) disk, (2) scales digitizing system, and (3) special instrumentation system.

A different method is used to input data from each of these sources. Input data are available to the application program for display, plotting, or listing on the lamp-bank display, the X-Y plotter, or the line printer, respectively.

The monitor mode is initiated either by entering a batch job through the card reader, which calls the monitor routines, or by entering the monitor mode according to the setting of the selector switches which load the monitor time-sharing routines. The monitor routines are as follows: a scales-monitoring routine, a special-instrumentation-monitoring routine, and an input-interrupt routine. Each of the monitor routines is executed whenever its corresponding interrupt is generated and received by the master computer. These routines strobe data into specific computer memory buffers in the communication area. The memory buffers have been preallocated to specific devices and hold only the most current data value. The memory buffers associated with the special instrumentation are accessed by an application program through a special interface routine. Whenever the monitor time-sharing routine is active, the memory buffer associated with the selector switches is accessed to determine which application program the experimenter wishes to execute. After loading the monitor routines and the associated interrupts are enabled, the monitor mode is initiated, either as the result of an interrupt or a call by a FORTRAN program.

Scales monitoring routine: During monitor mode, scales data are displayed on a remote lamp bank display unit. At the time an update occurs, an interrupt is generated and the scales digitizing

monitor is entered. Digital data from the scales are strobed into memory. The data are sent over a set of 10 isolated lines which represent the digits 0 through 9. Forty decade lines are used, one for each of the first 40 columns on a standard punched card. The digital I/O interface unit selects, under program control, each of the 40 decade lines independently. The routine receives decimal data from each of the 40 decade lines. After the data are gathered, an error detection routine examines the data for no-punch and multiple-punch errors. If any of the data are bad, a correction is attempted and the user is informed of the error. Finally, data are converted from decimal to binary values and stored in memory. Another FORTRAN-compatible interface routine passes the data to the application program. Only the most recent input of scales data is ever retained in memory.

Special instrumentation monitoring system: The special instrumentation system includes a lamp-bank display, an integrating digital voltmeter (IDVM), special instruments, and the selector switch.

The lamp-bank display is updated by the lamp-bank-update routine upon request from the FORTRAN application program. The caller provides the routine with the output data, the identity of the lamp bank to be updated, and whether a force or coefficient is to be displayed. From this information, the update routine transcribes all data and properly locates the decimal point.

The IDVM, the instruments, and the selector switches are all input devices. The instruments measure pitch, yaw, temperature, pressure, rpm, etc., and constantly monitor tunnel conditions. The selector switches are a group of four decades of BCD thumbwheel switches that are used to request a specific system function; that is, they can select which application program is to be executed. The selector switches are also used as a means of communicating exceptional conditions. For example, using preassigned selector switch positions, the experimenter may specify the software to: (1) enter subsystem initialization, (2) flag last data point recorded as being in error, (3) enter subsystem diagnostics, and (4) enter other types of coded information.

A separate interrupt switch is associated with the selector switches. It generates an interrupt when activated, and it causes the selected routine to be entered.

The instrument interrupt routine gathers data from various types of meters and stores the contents in dedicated computer memory locations in the communication area. When a FORTRAN application program wishes to access the data, it calls a special instrument interface routine. This routine converts the BCD number to binary, strips out code bits, and returns both the code bits and the instrument reading to the caller.

Disk monitor routine: The disk monitor routine is invoked in response to a call by an application program. Its function is to locate specific data stored on disk during a previous data recording, read that data into memory, and pass it to the application program. Parameters passed by the calling routine include:

- Point number
- Data identifier (i.e., whether the user wishes special instrument data, force data, pressure data, or any combination of these)

- A flag indicating whether or not to wait if any or all of these are not yet available (i.e., whether background processing is ahead of real-time data acquisition)
- A flag indicating whether this is an on-line test or a debug run.

After executing the request, the disk monitor routine returns to the caller with the following parameters:

- Request satisfied/not satisfied indicators for each type data
- Number of data cards for each system
- The number of active special instruments
- Data stored in card-image form on a disk file (which can be read by FORTRAN using a formatted read just as if it were being input through the card reader)
- Special instrument data in the array that was passed to the interface routine.

If the request was an on-line operation, the interface routine updates status information in the system software tables which are used by FORTRAN to indicate that future reads are to be from the disk. Further, the interface software copies the static data for the next sequential data point onto a dedicated portion of the disk so that it can be read directly by a FORTRAN application program (i.e., certain control words at the beginning of each disk sector must be properly initialized). After setting the proper disk seek address in other system tables which FORTRAN uses, and indicating the last sector for which data was recorded, the interface routine returns to the application program. This program can then continue its normal operation regardless of whether the static data being read are from card or from disk, or whether it is executing in a real-time environment or an off-line environment. The interface routine provides one additional benefit for the application program. In the event that the next sequential point has not yet been recorded (i.e., the application program has caught up with the data acquisition process), the interface program will, upon request, sit in an idle loop until the next sequential point is ready. The application program has a parameter by which it can indicate this wait request. This feature provides a means of accommodating the existing programs by exactly simulating the existing conditions of the reader (i.e., whenever you wish to read a card that is available), but it does not restrict future programs from providing other computations or updates during this dead time.

Several other interface routines were provided to allow the applications programmer to access the most recent scale data and to display either forces or coefficients on a lamp-bank display. These interface programs convert all input BCD information into the proper integer or floating-point value for the application program. In the case of the lamp bank, the application program indicates whether the number to be displayed is an aerodynamic force or a coefficient. The interface program then converts the integer or floating-point number to the proper BCD number, and handles the problem of properly placing the decimal point and all other display characteristics. All of these interface programs can be called directly from a FORTRAN program by using the standard FORTRAN call statement. These interface routines have been very helpful since they allow the applications programmer to concentrate only on the particular application he is programming and

relieves him from hardware considerations such as, is the lamp bank on, does it work, etc. These interface programs also provide him with a means of debugging a program off-line, and having it ready to run on-line with absolutely no program changes.

Software modifications-- Provision for software modifications in the computer system were brought about because of inadequacies in the computer system software. Perhaps with a newer computer system this software would not be necessary. In any event, the computer system software is dynamically modified whenever the monitor time-sharing routine is loaded. This routine is loaded by dialing the correct setting on the selector switches. The system software is then modified such that no job control cards will be recognized. The monitor mode software supervises all task loading and task scheduling functions. This software requires that the application programs to be executed must be previously stored on disk under preassigned names. The monitor mode time-sharing software is loaded and executes those jobs the experimenter requests via the selector switches. The monitor mode time-sharing software provides for time sharing up to 10 jobs in the background. It has the added capability of either allotting each job a time slice, or allowing each application program to determine its own time slice.

The computer system software recognizes only one overlay boundary; namely, at the beginning of the background area. Ordinarily, this condition would not present any major problem, since only one program at a time could run in the background. However, with the time-sharing monitor software, this presents a major problem, since one background program could overlay a second background program. To avert this problem, the overlay system software was also modified to recognize multiple program boundaries.

In summary, these modifications provide for a time-sharing environment that can be controlled by either the user or the system. The only requirement for running in this mode is that load modules must be stored on disk with preassigned names. All routines that could previously run in batch mode can also run in the time-sharing environment with no program modifications.

OPERATOR'S CONSOLE

Theory of Operation

The operator's console (fig. 6) is the control center of the total system. It houses the control panels for the static force subsystem, the static pressure subsystem, and the Toledo scales. In addition, it controls the HSDAS, OBS, and the DRS through the master computer. Thus, it initiates the gathering of both static and dynamic data. The philosophy of this subsystem is to provide pushbutton control (fig. 7) of the complete system once card deck initialization is completed, thereby making the computer operation imperceptible to the test engineer. In theory, each major subsystem, such as the force and pressure subsystems, appears to the operator as a stand-alone system with the computer in the background acting as an observer which records only the requested data. When a master record cycle is initiated, the record cycle provides the test engineer with backup data on IBM cards, magnetic tape, and real-time data recorded on disk for real-time processing. Initiating a data gathering session is simple and occurs only when the subsystem has been enabled and initialized.

A record cycle has the following sequence of events:

1. When the master record pushbutton is actuated the data lines of the enabled static force and pressure subsystem are switched momentarily to the computer (for about 20 msec after an initial delay of 5 sec). The data lines from the other systems such as the HSDAS, DAS, etc., are hardwired to the computer system through special interfaces and are not switched.

2. A record interrupt is generated to the computer.

3. The data are gathered and then recorded on magnetic tape (all data) and disk (static data only — force, pressure, and special instruments). The record cycle takes a maximum of about 40 sec to complete.

4. During the record cycle, the static force and pressure data lines are repeatedly switched to the computer for periods of 20 msec and then returned to the card punch so that card punching ensues. It should be noted that, for disabled force and pressure systems, switching will not occur. If the other major subsystems, such as HSDAS, DAS, OBS, etc., are disabled after card-deck initialization, they are logically disconnected at the computer interface and data is not passed to the computer.

The system design concept, therefore, allows the test engineer to connect and disconnect the various subsystems by pressing the appropriate pushbutton and selecting the appropriate program selector switch code.

The control console layout (figs. 6 and 7) has the following interactive control and display panels:

- Master control panel
- Computer display panel
- Force and pressure system control panels
- Data correlator
- Scale display panel
- Scales and program selector switch
- Factor board switches

MASTER COMPUTER SYSTEM

The master computer is the heart of the real-time computer system, and it is used to service all other subsystems. The services include input, output, and analysis of data in real time; verifying that other hardware subsystems are setup properly; and updating various displays. In any

event, it controls all hardware subsystems which are connected to it by software. The master computer system (figs. 10 and 11) comprises a central processor with 32k of 24-bit words, memory, 61 interrupt levels, a real-time clock, an interval timer, and 8 direct memory-access (DMA) channels. Its peripherals are: keyboard printer with a paper-tape punch and reader; 2 moving-head disks; 2 magnetic tape units; a high-speed paper-tape punch and reader; a line printer; a card reader; a fan fold X-Y plotter; and 20 digital input-output units. The system software is described in the software section.

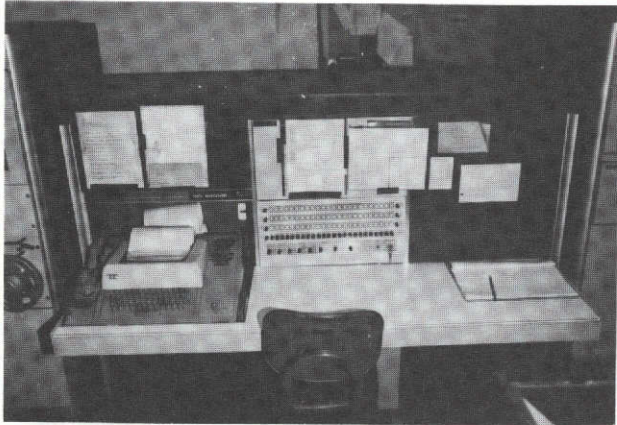


Figure 10.— Master computer console.

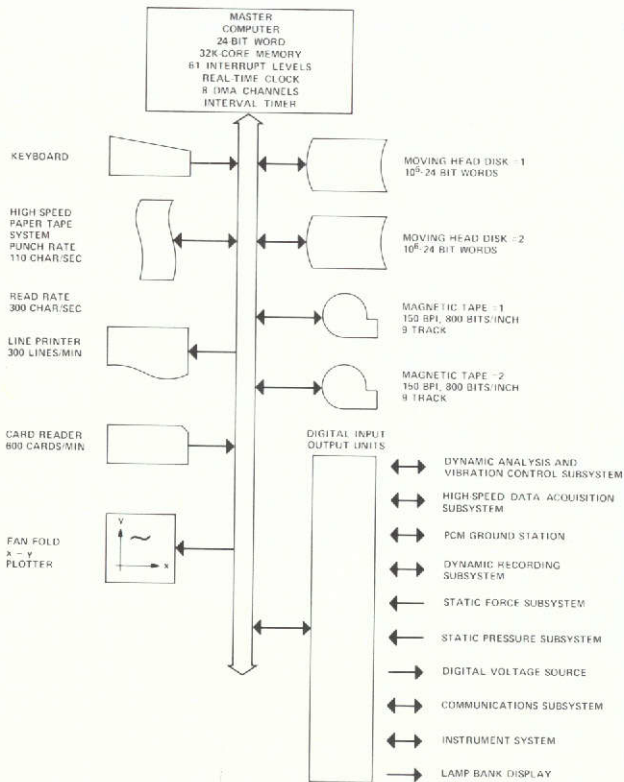


Figure 11.— Master computer block diagram.

HIGH-SPEED DATA ACQUISITION SYSTEM

The HSDAS (fig. 12) is a data-gathering computer front end. It gathers dynamic data from as many as 60 channels of guarded low-level voltage sources and four single ended high level sources. It is a high resolution and accuracy subsystem that simultaneously conditions, samples, and holds voltages from each source. It multiplexes the samples onto a single analog-to-digital converter (ADC) that measures and converts the analog voltage into a corresponding digital value. The subsystem transmits these digital values to the master computer for recording on magnetic tape.

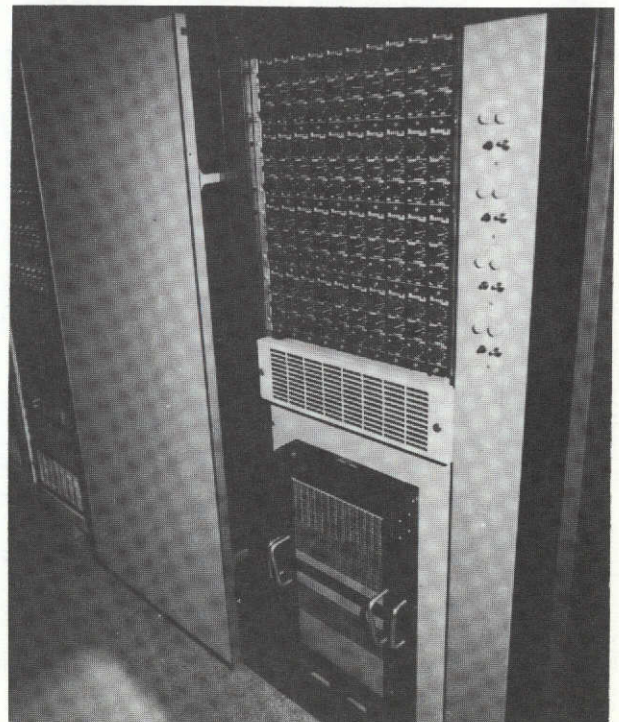


Figure 12.— High-speed data acquisition system.

Figure 13 shows the three major subdivisions of the subsystem. During the initial planning stage of this subsystem, a survey of manufacturers was made. It was found that no manufacturer of data gathering equipment made a system that meets the 40- by 80-foot wind tunnel requirements. However, there were several manufacturers that made high-speed multiplexer/ADC computer-compatible front ends. Other manufacturers made high quality signal-conditioning amplifiers. Therefore, dividing this subsystem into three subdivisions and procuring these independently as a signal conditioner unit, a high-speed digital front end, and an interface was an obvious approach to implementing this subsystem.

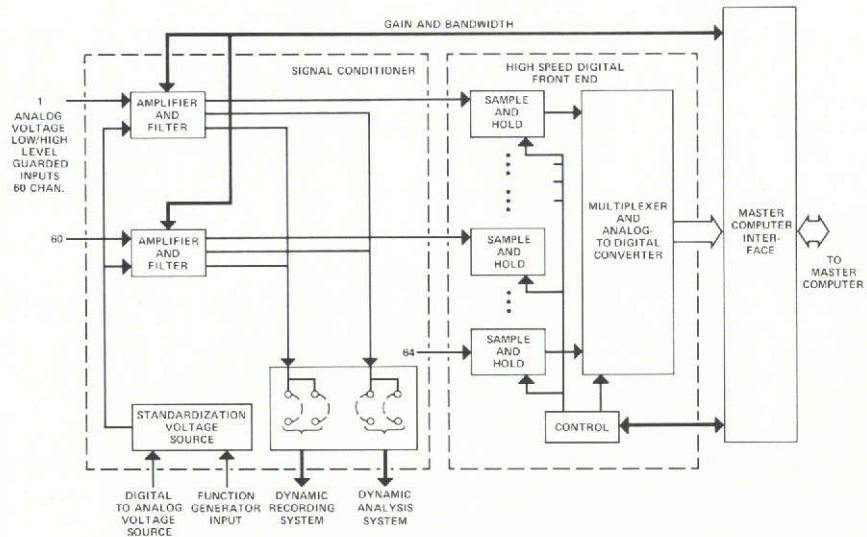


Figure 13.— High-speed data acquisition system block diagram.

The signal conditioner (figs. 14 and 15) is a low/high-level guarded-input unit with two inputs, three outputs, and a gain/bandwidth interrogation circuit that is micrologic compatible. The conditioner features high common-mode rejection (120 dB at 60 Hz) with matched rolloff characteristics, selectable gains and bandwidths, and an overload indicator. One of the inputs is an auxiliary input for instrument standardization.

The high-speed front end contains 64 sample-and-hold amplifiers and a 64-channel multiplexer/ADC unit. It features a resolution of 1 part in 16,383, worst case accuracy of 0.04 percent of full scale, and a sample rate of 100k samples/sec. It is capable of random and sequential sampling. The computer interface is an in-house design and has two separate chassis, one at each end of the connecting cables. The interfaces at the HSDAS end consist of differential line drivers and receivers for data and control lines. It has special control logic that checks data-word dropouts and scan synchronization. This interfacing system is self-synchronizing. The interface at the computer end contains data line drivers and receivers in addition to a more sophisticated control logic. Its control logic provides for variable scanning rate, external scan control, and external triggering of a recording session. These last two functions provide the capability of synchronizing data sampling over each single period of periodic functions. Some of the advantages of synchronized sampling are that it: (1) optimizes data

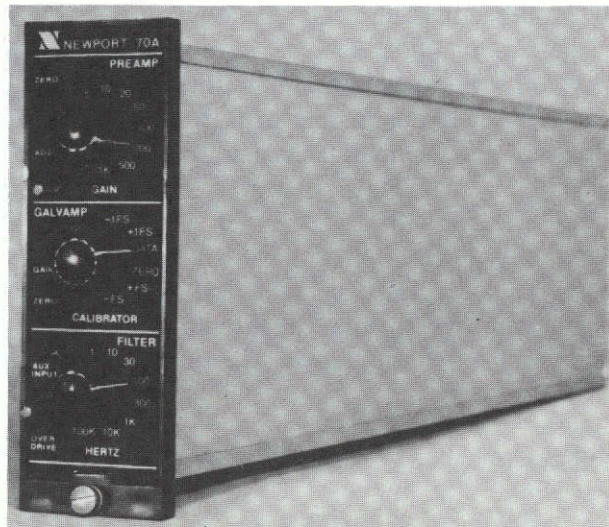


Figure 14.— Typical signal conditioner.

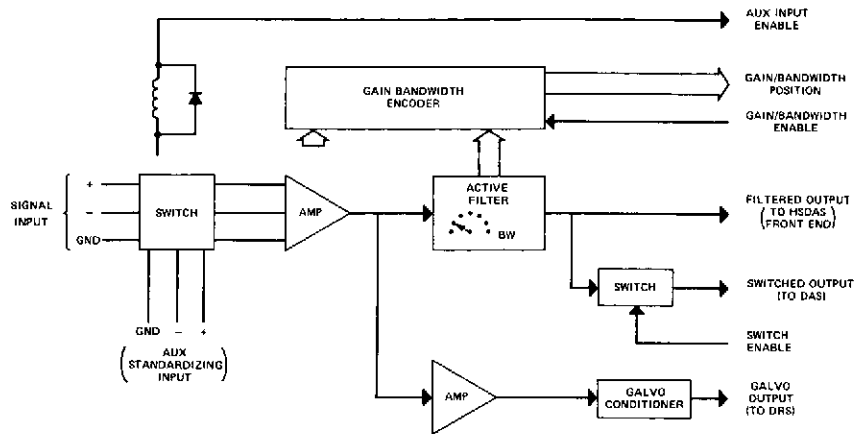


Figure 15.— Signal conditioner block diagram.

gathering, (2) reduces aliasing error due to truncating a sample for harmonic analysis, and (3) provides accurate phase correlation.

The following is a summary of the system characteristics:

- 60 low-/high-level channels
- 60 simultaneous sample-and-hold channels
- 0.1 percent of full scale is worst case end-to-end accuracy
- 10 V full scale equals 16,384 counts
- Scan rate is variable from 1 to 1.1k scans/sec in binary steps of 1, 2, 3, etc.
- 100k samples/sec, maximum sample rate
- Selectable gains from 1 to 2000 in steps of 5, 10, 20, 50, 100, 200, 500, 1k, and 2k
- Selectable bandwidths (in Hz): 1, 10, 30, 100, 300, 1k, 10k, and 100k
- 4-pole Bessel filter
- Programmable ADC gains of 1, 2, 4, 8
- Guarded input circuiting capable of withstanding 300 V (ac or dc) peak
- Gain and bandwidth identification data for computer.

DYNAMIC ANALYSIS SYSTEM (DAS) AND VIBRATION CONTROL SYSTEM (VCS)

The DAS and the VCS (figs. 16 and 17) are two software systems with the same hardware. When either software system resides in the minicomputer memory, it excludes the other software system.

In the DAS configuration, the system can operate on-line or off-line as a stand-alone data gathering and analysis system. In the on-line mode, the DAS performs as a quick-look system while the master computer gathers data from the other subsystems as shown in figure 2. While operating on-line, it can gather data from all or any two of 32 signal sources and perform any one of a number of time series analyses (ref. 2) in real time. In particular, it can perform histograms, autocorrelation, cross-correlation, impulse responses, characteristics functions, Fourier transforms, autospectrums, cross

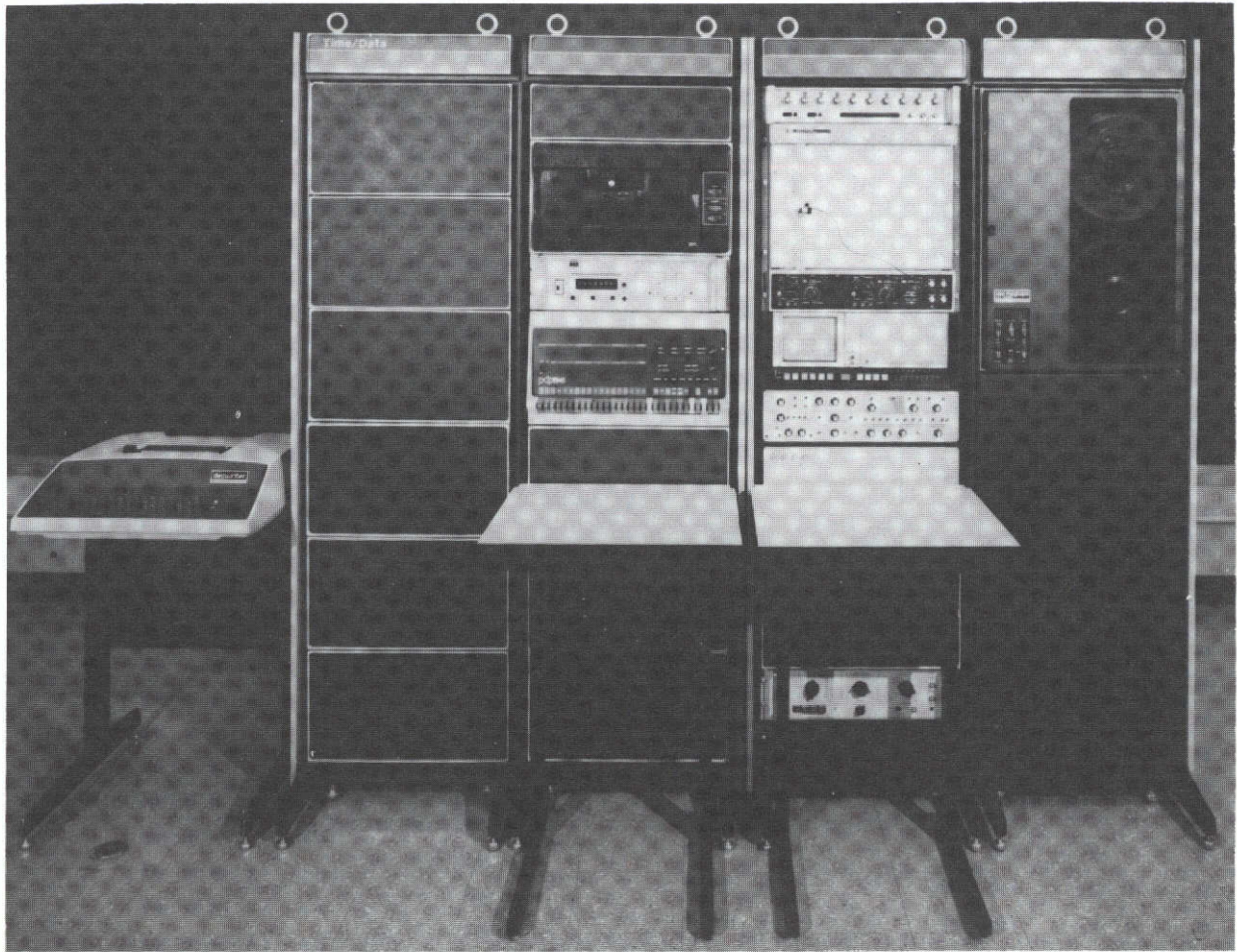


Figure 16.— Dynamic analysis subsystem.

spectrums and transfer functions. It can perform linear, continuous, or exponential averaging of up to 51,200 samples. It can display the results on command on a cathode ray tube (CRT), X-Y plotter, or printer. The system can vary the analysis range and resolution on input data. In addition, there is a modal analysis software package (MAP) (ref. 1) that numerically evaluates the pole locations and their residues from empirically derived functions.

In this configuration, the system can also operate in an interactive mode using a time series language (TSL) and the input keyboard printer.

TSL (ref. 3) is an interpretive language similar to BASIC. Its mnemonic instructions represent time series mathematical functions. For example, a mnemonic instruction such as 'C SPEC B0, B1, B2' is interpreted by TSL as an instruction to perform a cross-spectrum analysis between the data stored in buffer B0 and B1 and to add the results to the contents of B2. TSL is divided into five basic groups: input-output (I/O) statements; block arithmetic statements; branch, looping, and test statements; non-block arithmetic statements; and compiler-editor statements.

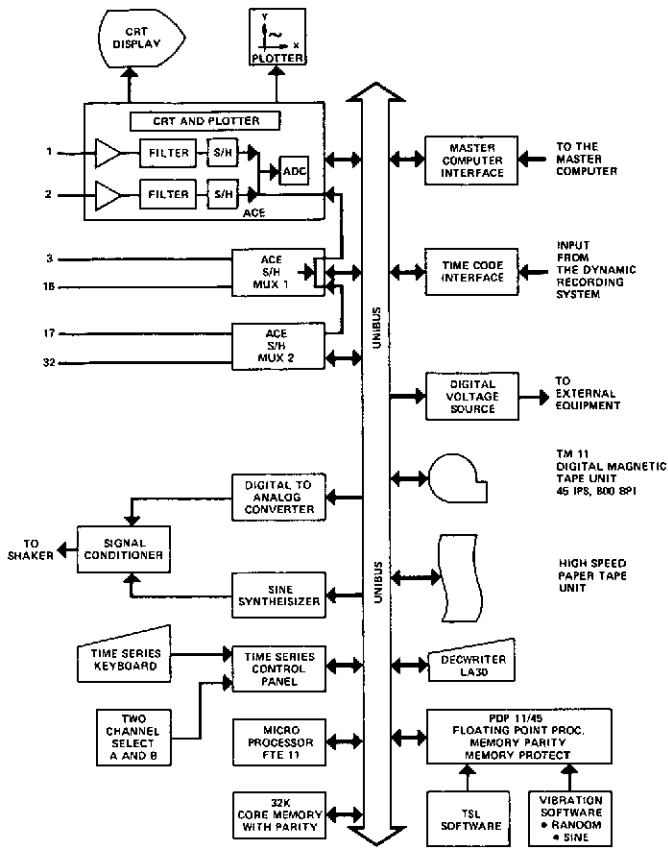


Figure 17.— Dynamic analysis subsystem block diagram.

form an average autospectrum or an average cross-spectrum. Some of the other block instructions include taking the square root, integrating, differentiating, computing logs, antilogs, cosines, sines, etc.

The combined five statement groups make up a powerful language that is attractive to the researcher. Some of the features that make this language attractive are:

- TSL is easy to learn because it is similar to the widespread BASIC and FORTRAN languages.
- TSL provides for a fully and automatically calibrated system.
- It has a complete set of block arithmetic statements.
- It provides for instant program modification and execution because it does not require multistep editing and compiling. It is an interactive system and has full execution-time diagnostic capability, such as error trace and messages that allow the user to correct his errors quickly.
- It has full subroutine capability.
- It uses core economically.

The I/O statements include the standard instructions such as I/O to peripherals, read in or output a character, format, etc. There are some CRT output instructions that position the CRT beam, erase the beam, and draw straight line traces between specified coordinates of X and Y. There are other special I/O instructions that set up the data-gathering front end, which is also called analog conditioning element (ACE). The ACE I/O instructions set the sampling rate, number of channels, and modes. These instructions start acquisition and coordinate acquisition with processing loops.

The block arithmetic statements are a group of powerful instructions providing Fourier transform operations and supporting block operations. This group of instructions provides for defining and undefining block sizes; setting or retrieving individual parameters; adding, subtracting, multiplying, and dividing two real or complex arrays. Additionally, the block instructions provide for multiplying an array with its complex conjugate or two arrays to

- It executes fast.
- It provides alphanumerics on the CRT.
- It has a display routine that makes (via keyboard commands) a calibrated and annotated plot of any real or complex block in storage.
- It is capable of inputting analog data on a block basis, and it can set input attenuators, sample rate, multichannel, anti-aliasing analog filters, and triggering. Further, it allows buffering to be controlled by the user.

In the VCS configuration, an independent software system exercises control over the hardware shown in figure 2 and drives a shaker in either a random or sine mode. The software inputs a signal from an accelerometer mounted on a model and initiates the digital conversion cycle in the ACE. With the aid of the fast Fourier hardware, it computes a line spectrum and compares the spectrum with a reference spectrum generated during the test setup. From the results of this comparison, the software generates a new shaker drive signal. This closed-loop operation continues in real time, generating the drive signal several times per second.

The VCS software provides a conversational mode that allows the operator to change the reference spectrum at will during a real-time operation. In the test setup mode, it guides the test engineer, via a parameter entry routine from the keyboard/printer, by asking operator-oriented questions and accepting only credible answers, then reporting on the system action through messages after each answer.

Additionally, the VCS software has an editing capability which allows the test engineer to load, modify, or prepare new test programs. It provides the system with a starting capability from a front panel, and it initiates the test program automatically at outputs that are at 1/8 full level. The software responds to an increase control which allows manual increase of the output level to a maximum specified by the test program.

ONBOARD PCM SYSTEM (OBS)

The OBS is a high-speed airborne data-gathering system modified to meet the wind-tunnel throughput requirements. The system (fig. 18) is modular and is divided into two major subsystems. These are the airborne PCM hardware (fig. 19) and a computer-based ground station (fig. 20). The prototype of the airborne hardware was developed for NASA Edwards Flight Research Center (refs. 4-5). The wind-tunnel airborne hardware is a production version. The ground station is a developmental project designed for the wind tunnel.

The airborne hardware comprises several presample filter and transducer conditioning modules connected to a multichannel analog and digital multiplexer with an 11-bit encoder for digitizing analog data. The airborne hardware can operate as a standard PCM system or as a random-address sampling system. The hardware features a programmable gain amplifier with a single downrange step from the present gain setting. There are a number of special input printed circuit (PC) cards for the airborne hardware that allows the OBS to accept thermocouples, strain gauge bridges, and low level

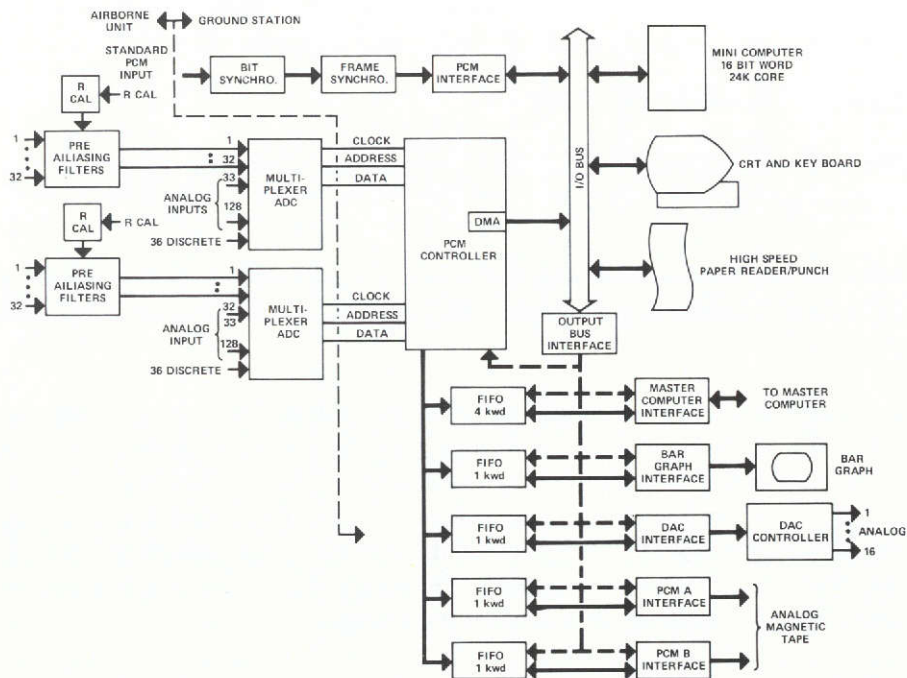


Figure 18.— Onboard system block diagram.

signals. In addition, there are PC control cards that provide for discrete outputs, relay drivers, and scanivalve controllers.

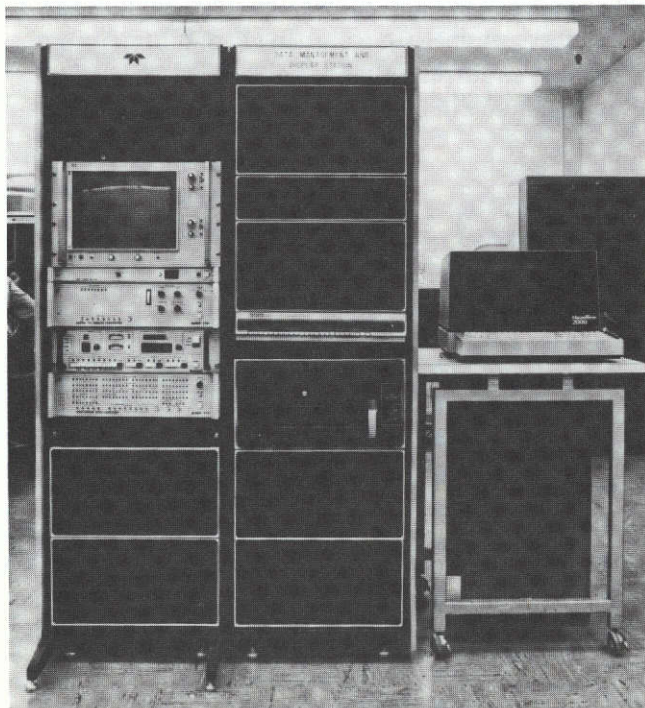


Figure 19.— Onboard system computer base ground station.

The ground station features a PCM and display controller that is programmable, a minicomputer with 16k 16-bit words of core, a CRT display, a high-speed paper-tape reader and punch, a PCM bit synchronizer, a PCM frame synchronizer, a PCM frame synchronizer, a PCM frame synchronizer, a bar graph, 16 channels of digital-to-analog conversion, a magnetic tape unit, a master computer interface, and several first-in-first-out (FIFO) memory modules.

The OBS is a computer-based system which can operate on-line with the master computer. When on-line with the master computer, computer-to-computer conversation is maintained under control of "hand-shaking" (ACK/NAK, etc.) protocol software through a bidirectional hardware interface. This same communications software provides for programming the OBS system from the master computer. The master computer programs the OBS by

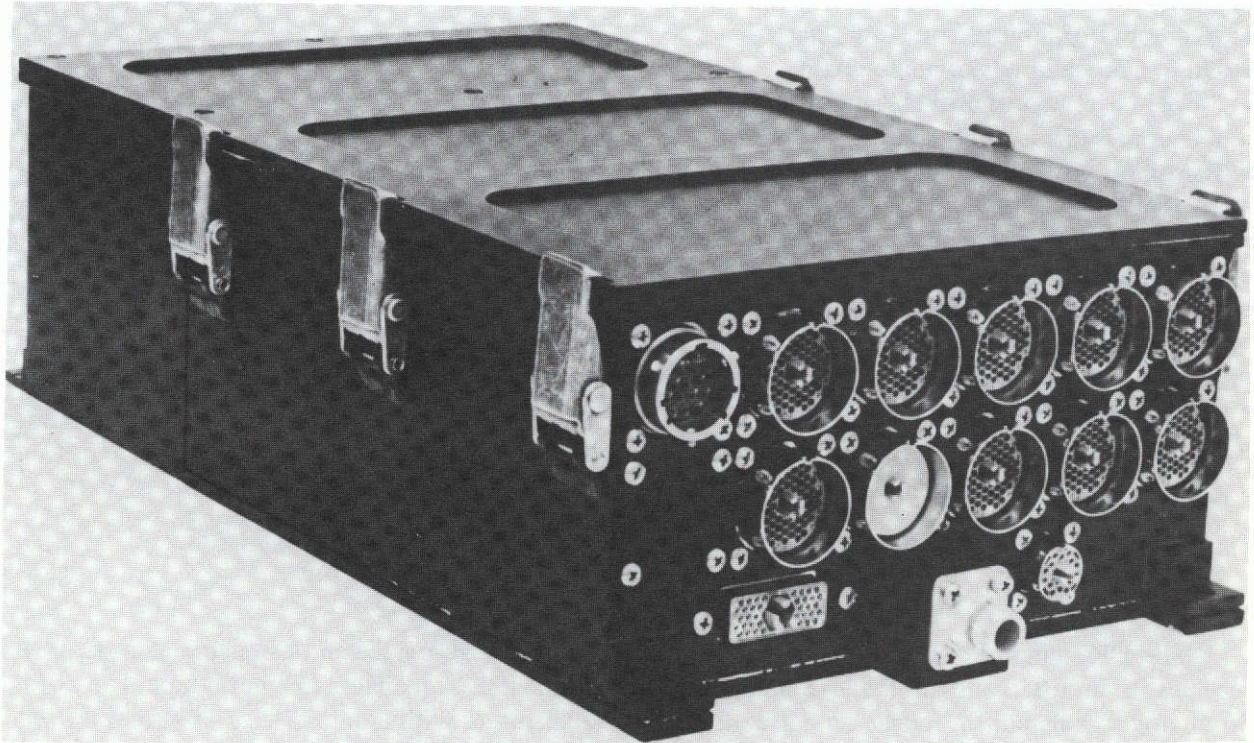


Figure 20.— Onboard PCM data gathering front end.

transmitting a table of parameters, which determines: the type of data (whether static or dynamic), the sampling rate, the type of calibration, the number of data samples, and a bad channel list. The minicomputer responds by retransmitting the same table back to the master computer for verification. Data are acquired by the OBS and transmitted to the master computer under commands from the master computer.

The other peripherals of the OBS, such as the bar graph, bit synchronizer, frame synchronizer, and DAC, are set up in the OBS stand-alone mode.

Programming is done in conversational mode; the user is prompted with setup questions. His answers to these questions result in an operational system. Once the OBS is set up, the PCM and controller simultaneously distribute the data to the various displays and output units. The input and output of data proceed independent of the minicomputer through the hardware controller and its I/O data bus. This system architecture frees the minicomputer so that it can perform other functions such as signal averaging, peak detection, etc.

This system features:

- Stand-alone or on-line operation
- Voltage range: ± 10 mV to ± 5 V
- 11-bit resolution
- Fully guarded low-/high-level system
- Programmable gain

- Automatic down ranging
- Accuracy: 0.5 percent of full scale
- Up to 256 channels of analog data
- 3-pole butterworth anti-aliasing filters
- Random address or standard PCM
- Bar-graph display
- 16 DACs
- Standard PCM interface
- PCM output
- Conversational setup software
- Portable address generator for checking out the airborne hardware
- Concurrent data I/O and processing
- FIFO memory
- Scanivalve control

DYNAMIC RECORDING SYSTEM (DRS)

The DRS (figs. 21 and 22) is the system's raw-data storage device and time-code generator. It receives its inputs from the HSDAS and OBS and records these on analog magnetic tape. It records 56 analog signals continuously on 8 tracks and 2 PCM sources with a data rate of 10k samples/sec. It has playback hardware for playing back 28 analog channels and 2 PCM channels.

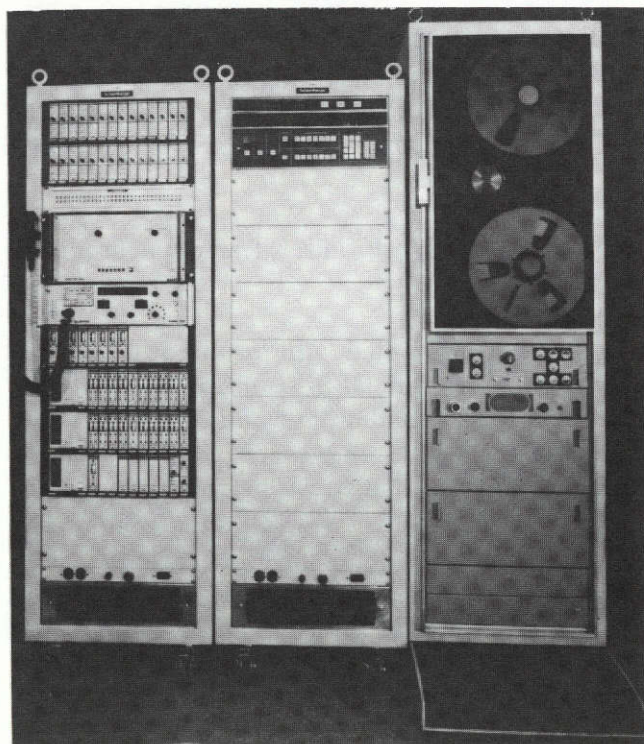


Figure 21.— Dynamic recording system.

The DRS is comprised of an FM multiplex unit, a demultiplex unit, a tape search and control unit, and a 14-track magnetic-tape record and playback unit. All these units are interconnected to perform as the backup data recovery system. The multiplex unit multiplexes 7 data channels and an encode signal on each of 8 tracks for a total of 56 record channels. The demultiplex unit is comprised of 4 groups of 7 discriminators which demultiplex 4 tracks into 28 channels of analog data output, with a scan synchronizing signal.

The tape search and control unit controls the magnetic tape unit in the playback mode. In addition, it generates a time code that is recorded on analog magnetic tape and transmitted to the master computer and the DAS for recording. The tape search and control unit is programmable from the master computer.

In the record mode, the DRS functions as either a continuous or burst recording

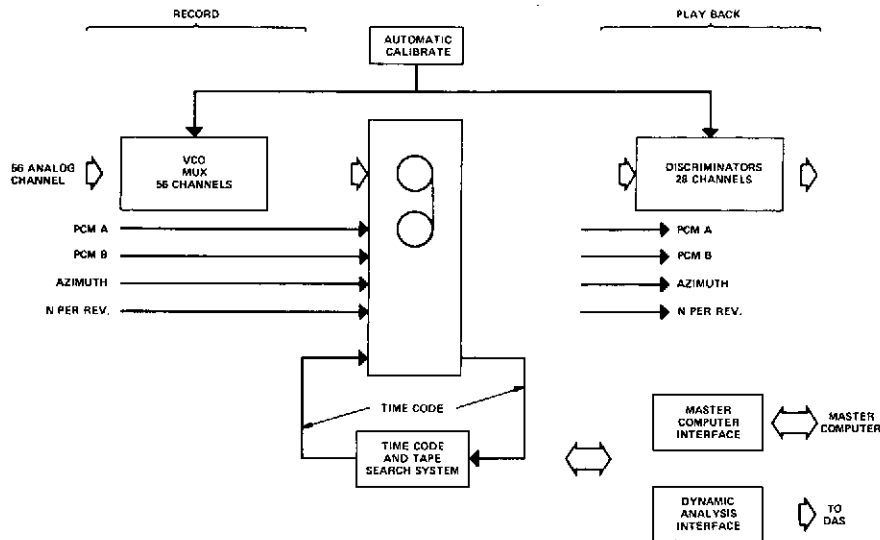


Figure 22.— Dynamic recording subsystem block diagram.

system. The continuous recording mode consists of starting the DRS at the beginning of the run when zeros are taken and allowing it to record for the duration of the run. A run comprises up to 100 points and a point is the interval during which test conditions are held constant. Burst recording is initiated each time a master record session is actuated. The following are some DRS characteristics:

- Frequency response: 1 kHz, with a 5-pole Bessel filter
- Amplitude correlation: ± 0.5 dB
- Interchannel time correlation: 30 μ sec
- Harmonic distortion: 0.75 percent
- Signal-to-noise ratio: 40 dB
- Independent linearity: ± 0.05 percent
- Zero drift (± 0.01 percent of center)
- 56 analog record channels
- 28 analog playback channels
- 2 PCM record/playback channels
- Maximum output: ± 10 V, 50 mA
- Recording time: 2 hr at 15 ips

STATIC PRESSURE SYSTEM

The primary function of the pressure system (fig. 23) is to digitize analog signals from two mechanical pressure scanning transducers. Its secondary function is to scan signals from a number of discrete analog transducers, for example, strain gauges, thermocouples, potentiometers, etc. It should be noted that many of these functions can be performed by the OBS.

Scanivalves are the pressure scanning transducers. Each scanivalve consists of one to five 24-/48-port scanning modules connected to a single drive motor. Each module has a single pressure

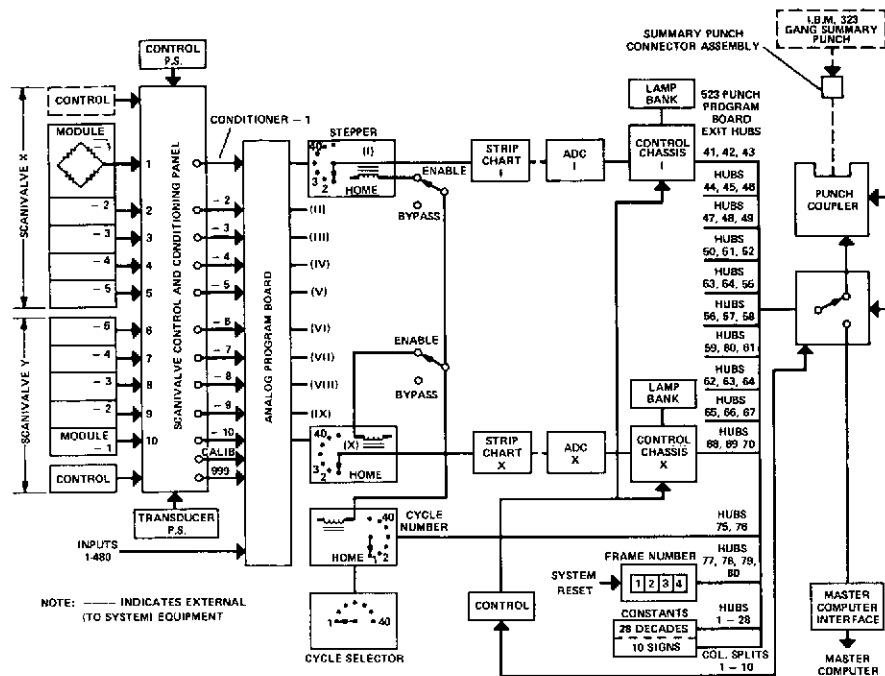


Figure 23.— Static pressure subsystem block diagram.

cell that is vented to each port sequentially. The sequential output from the module is connected to a strip-chart recorder. There are 10 strip-chart recorders, one for each module. These recorders have shaft-angle gray-code encoders that digitize the slide-wire position which represents the measured signal. The output of the gray encoder is translated by the control chassis and recorded on punched cards by a summary punch.

The secondary mode of operation is accomplished by the use of 10 stepping switches. The stepping switches are 48-point steppers that commutate the output signals of 48 discrete transducers on to one output line equivalent to the output of one module. Since the stepping switch operation is in lieu of the scanvalve operation, controls for each are identical.

In addition to recording the data on punched cards, the static pressure system is hardwired to the master computer through the special computer interface. The outputs of the recorder-ADC combination are switched to the computer interface for a fraction of a second. During this time, the computer records the static pressure data in an open loop mode. At the end of the switching interval, the data are switched to the card punch. The data switching occurs independent of the computer. As a result, the computer operation is imperceptible to the static pressure system.

STATIC FORCE SYSTEM

The static force system (fig. 24) comprises a scale-averaging system, up to four channels of strain-gauge data, factoring and correlation inputs from the operator's console, and a summary card punch. The scale-averaging system is a unit designed to digitize the values measured by the scales.

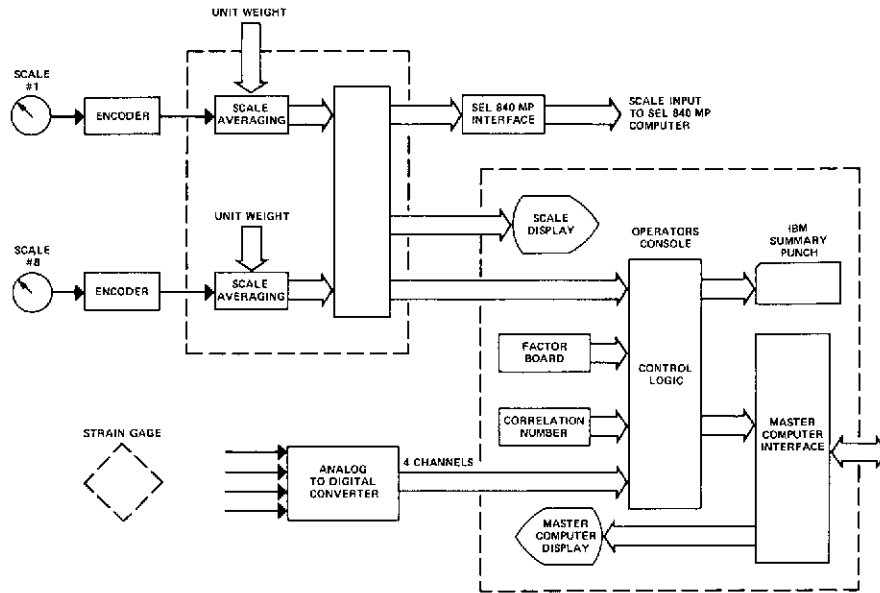


Figure 24.— Static force subsystem block diagram.

The strain-gauge data are processed by up to four analog-to-digital converters (ADCs) as inputs to the operator's console.

The averaging subsystem has two modes of operation, the record mode and the monitor mode. In the record mode, the averaging subsystem provides averaged digital scale data to the system operator's console. In the monitor mode, the averaged data are automatically updated and displayed in counts on a scale display, and input to the master computer. When a recording session is started, data from a correlation counter, the 28 thumbwheel switches, up to 4 analog-to-digital converters (ADCs), are combined with scale data and punched on cards. In addition, the card image is transmitted from the computer interface to the computer under program control. The number of characters that are available for recording exceeds the card capacity. As a consequence, up to four card formats are generated depending on the number of ADCs used on the test. When the ADCs are used, the number of thumbwheel switches corresponding to the number of decimal characters in the ADCs are deleted from the card format.

SPECIAL INSTRUMENT SYSTEM

The special instrument system (figs. 25 and 26) expands the computer input capability by time sharing a single computer input unit with as many as 15 special purpose instruments. The special instrument system has the following characteristics:

- All instruments have the same unit address.
- Each instrument is assigned a unique priority code for identification.
- Each instrument has a hardware priority that is determined by its physical position in the multiplexing system.

- The instrument hardware priority is at the same level as the level assigned to the instrument for identification.
- Data can be input to the computer at a variable rate. The rate is dependent upon computer utilization and varies from approximately 0.1 Hz to 10 kHz.
- The computer can only service one instrument at a time.

The special instrument system comprises an instrument interface, a control bus, a data bus, and a system clock. All of these components are connected together and operate as a system.

The system operation can best be described by assuming an instrument is ready to update. Whenever the instrument updates, the instrument interface connects the instrument to the control bus and the data bus. The instrument interface then waits to be interrogated by the system clock. Whenever a clock pulse is generated (rises from a logic zero to a logic one state), it is propagated along the control bus, interrogating each instrument. The instruments having data ready for input are queued with the highest priority code first when the clock pulse returns to logic zero. The next clock pulse generates an interrupt to the master computer for the highest priority instrument in the queued state, causes the data to be placed on the data bus, and freezes the system clock in the logic one state. Whenever the interrupt is processed by the computer, the clock pulse signal changes from the

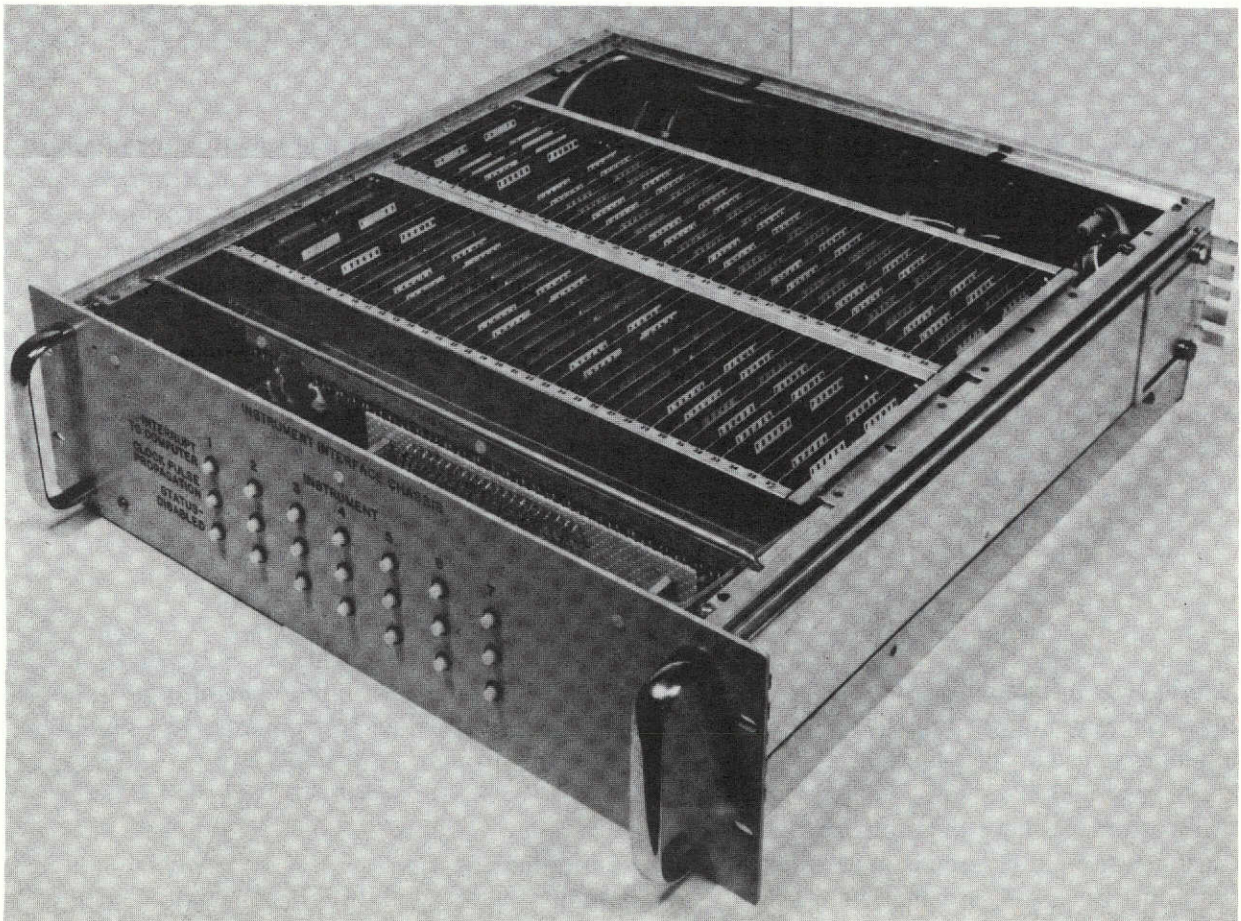


Figure 25.— Special instruments interface.

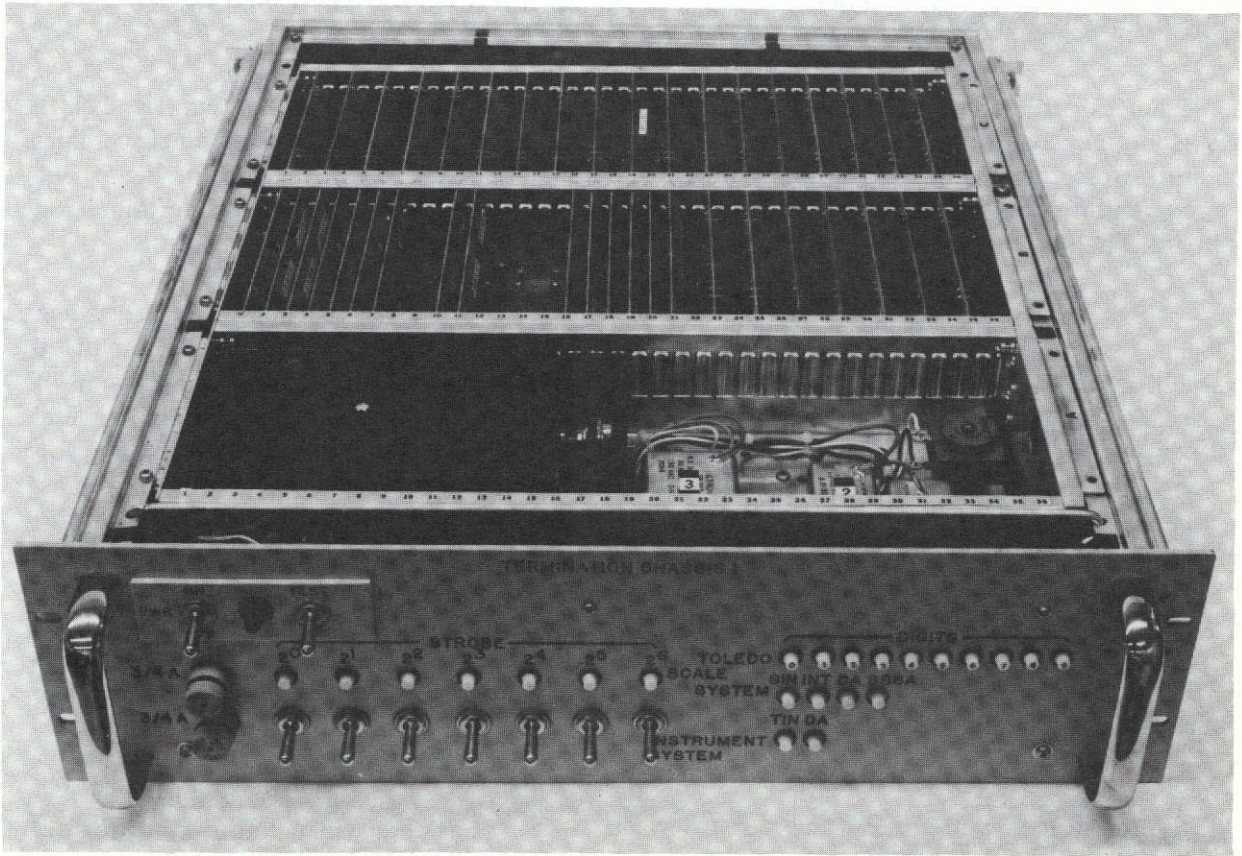


Figure 26.— Special instruments to master computer interface.

logic one state to the logic zero state. This change of state from logic one to zero connects the next queued higher priority instrument, and unlocks the system clock. The next clock pulse strobes data out onto the data bus and generates an interrupt when changing from logic zero to logic one and connects and quees the current highest priority instrument when changing from logic one to logic zero. The next and all succeeding clock pulses generated perform the same function.

The amount of time that it takes the master computer to service the interrupts determines the clock-pulse rate. The clock will nominally operate at 10 kHz, and there will be an average of four interrupts per instrument per second.

The following instruments are currently on the instrumentation multiplexing bus:

- Counters I and II
- Pitch and yaw instruments
- Dangleometer (measures the pitch angle of the model)
- Barometric pressure

SYSTEM PROGRAM BOARD AND TRANSDUCER CONDITIONING SYSTEM

A signal distribution center and transducer conditioning are part of every system (fig. 27). In this system, a system program board (fig. 28) is the signal distribution center and all electrical lines emanating from the test model terminate on this board. In addition, outputs from the high-speed data acquisition system signal conditioner and the playback signals from the DRS also terminate on this board. The program board is a removable board and can be reprogrammed for each test.

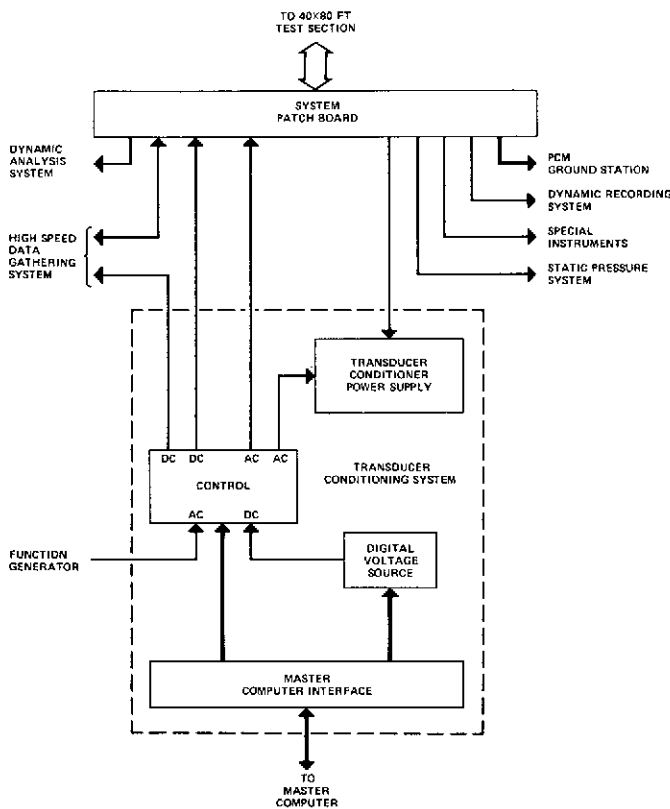


Figure 27.— Transducer conditioner subsystem and system program board block diagram.

reference. This same technique may be used to determine the effects of vibration on the transducer. In this case, the VCS vibrates the model and the gauge output is recorded for off-line processing.

Some of the conditioning system characteristics are:

Transducer conditioner

Common mode rejection at 60 Hz: 120 dB

Channel-to-channel crosstalk: 115 dB

Overload recovery within 0.01 percent of steady state: 50 μ sec

Stability: 0.01 percent of full scale for 30 days

The transducer conditioning system (fig. 29) provides for the excitation of 60 transducer bridges, for the transmission of a digital voltage-source output to several locations, and for the distribution of ac voltages from a function generator. The transducer conditioner unit provides for constant voltage or current power to resistance bridges and for shunt resistance calibration. It is an eight-wire system with two wires for power, two wires for remote voltage or current sensing, two wires for bridge calibration by shunt resistor, and two wires for signal transmission.

It has an auxiliary input that provides for powering the transducers from either the digital voltage source (fig. 30) or from an external function generator. This feature is used to check the dynamic characteristics of the transducers and wiring. This is done by connecting a swept sine-wave signal and engaging the shunt calibration resistance. The gauge output signal is measured by the DAS, and the transfer function is determined, displayed, and recorded for future

Constant voltage characteristics

Output voltage rise with open sense leads: 1 V
Voltage range: 0 to 40 V
Resolution: 0.01 percent of full scale
Ripple: less than 100 μ V peak to peak

Constant current

Current range: 0 to 50 mA
Resolution: 50 mA
Ripple: less than 500 mA

Digital voltage source

Voltage range: ± 8.192 V and ± 50 V
Resolution: ± 0.005 percent of full scale
Accuracy: 0.01 percent of full scale
Current: 5 A

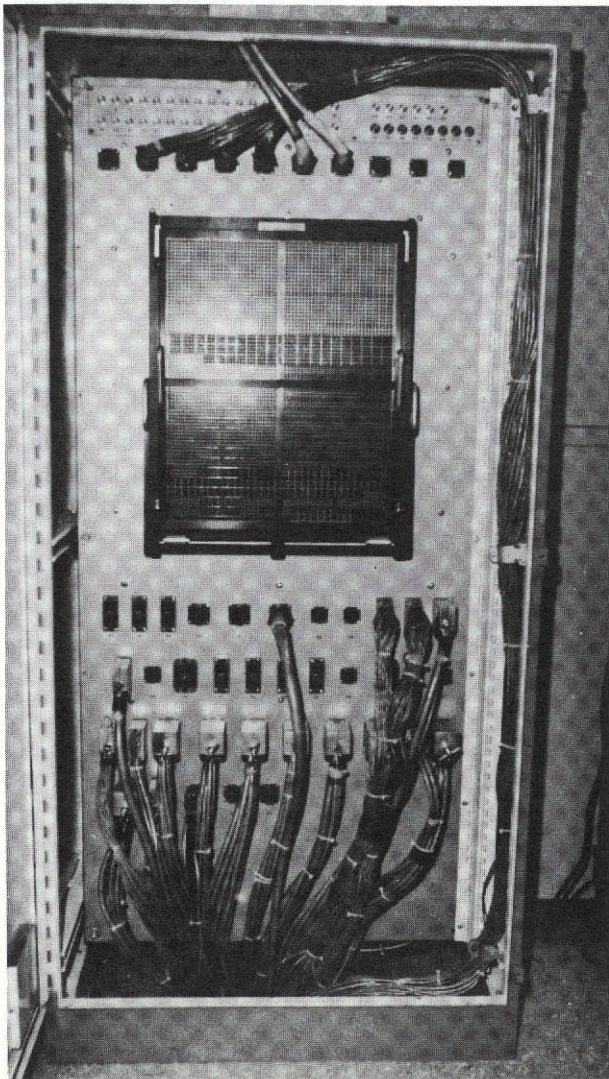


Figure 28.— System patch board.

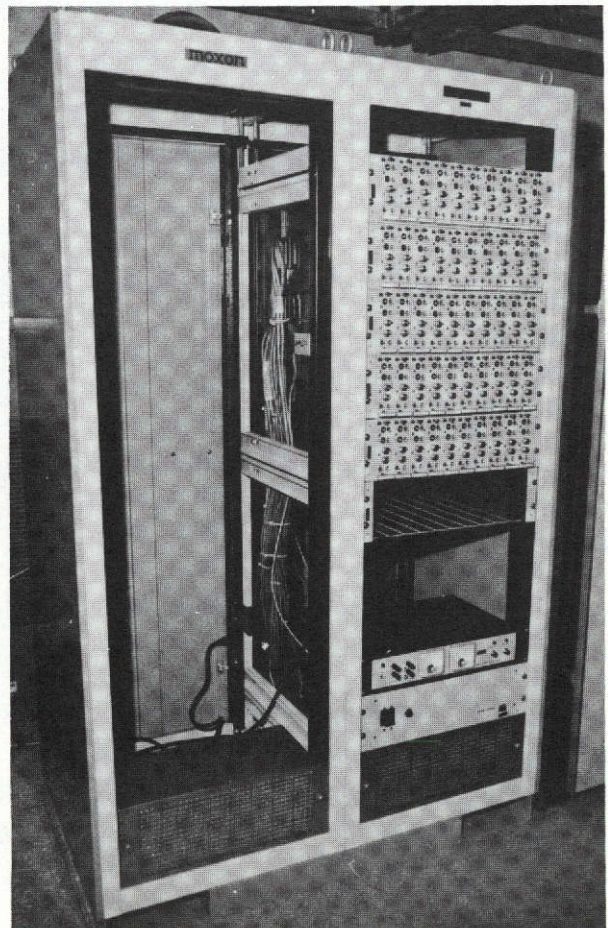


Figure 29.— Transducer conditioner.

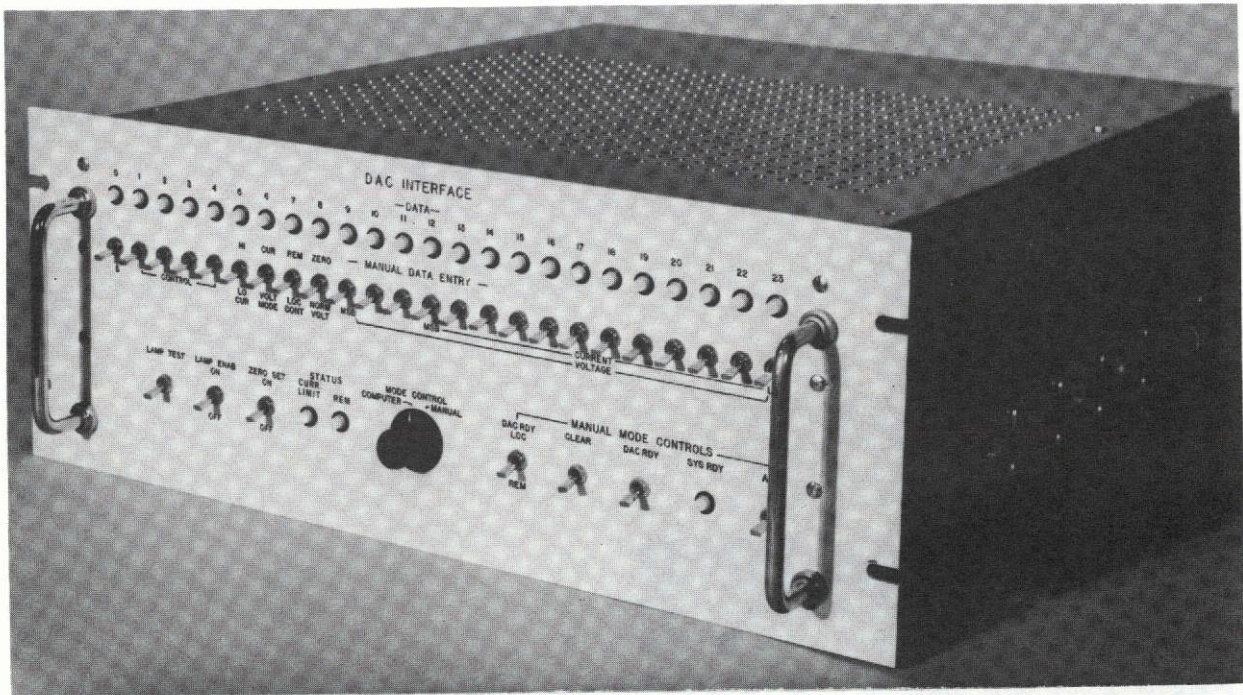


Figure 3Q.— Digital voltage source to computer interface.

AMES DATA COMMUNICATION SYSTEM (ADCS)

The ADCS (figs. 31 and 32) interfaces between the 40- by 80-foot wind tunnel system and the Ames central data processing center. This system comprises a minicomputer, a keyboard, a high-speed paper tape reader, a line printer, an RS232 interface, and a control panel for the selection of various functions.

The minicomputer in the ADCS has 16k bytes. The word size is variable, ranging from 8 to 32 bits in 8-bit byte steps. This computer is basically a byte-oriented machine. Its entire instruction set is microprogrammed so that, although it is byte-oriented, the arithmetic operations have 16-bit precision.

The ADCS has three operational modes: (1) on-line to the central data processing center, (2) local, and (3) test.

In the on-line mode, the ADCS can select any one of three input sources for transmitting data to the Ames central data processing center. These input sources are: (1) master computer, (2) data acquisition systems, and (3) paper tape reader.

When the master computer is selected as the input device, the ADCS appears as a simplex communications peripheral to the master computer. As such, the ADCS connects to a voice grade telephone via a standard modem that has a signaling rate of 2400 bits/sec. Communications to the Ames central data processing center are under control of bisynchronous communications protocol.

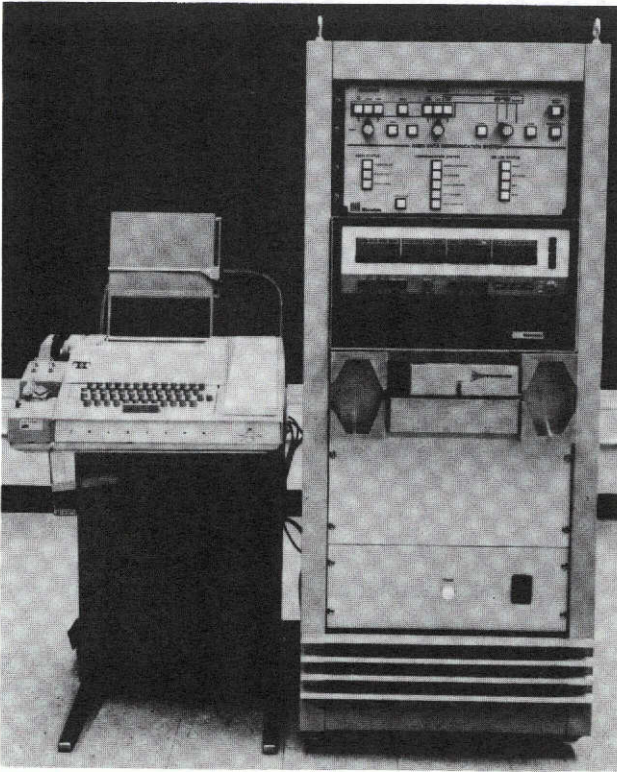


Figure 31.— Ames Data Communications Subsystem.

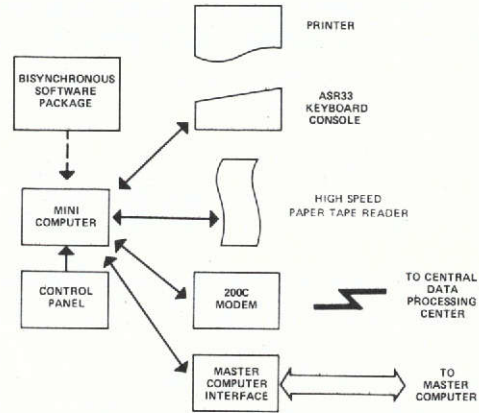


Figure 32.— Ames Data Communications Subsystem.

This communications package contains several data recovery routines that respond to line-loss conditions, negative data acknowledgment, and out-of-synchronization conditions. Further, the software can retransmit blocks of data as many as 64 times, depending on the type of failure. In this mode, the 24-bit word from the master computer is transformed into bytes by the master computer to ADCS interface. The ADCS packages the computer words into 80 ASCII character blocks and prepares them for the bisynchronous transmission.

In the local mode of operation, the ADCS acts as a master-computer line printer and controller. It receives data from the master computer as ASCII (truncated ASCII) characters and prints them on a 120-character line format.

In the test mode, the ADCS transmits a predetermined message to the Ames central data processing center to determine the condition of the transmission lines. The responses during the test mode are displayed on the ADCS control panel. The ADCS can detect the following causes of data transmission errors: (1) power failure, (2) modem fault error conditions, (3) communication errors, and (4) parity errors.

At present, The ADCS has a throughput rate of 100 characters/sec. However, the hardware and software can be tuned to approach 10k bytes/sec.

CONCLUSIONS

In conclusion, the wind-tunnel real-time computer system is a distributed data gathering system that features the following: a master computer subsystem, a high-speed data gathering subsystem, a quick-look dynamic analysis and vibration control subsystem, an analog recording

backup subsystem, a PCM onboard subsystem, a communications subsystem, and a transducer excitation and calibration subsystem. All of these subsystems are off-the-shelf systems which have overlapping characteristics. This overlapping of functions permits a graceful degradation as subsystem failures occur because each subsystem can operate in the stand-alone mode.

The subsystems are united to the master computer through an executive software system and standard hardware and FORTRAN software interfaces.

The executive software system has four basic software routines: the playback, setup, record, and monitor routines. The playback software is used for recovering data from raw analog signals recorded on analog magnetic tape. The setup software guides the user through system setup, calibrates each subsystem and verifies that each subsystem is on-line and operational. The record software collects both dynamic and static data gathered by each of the subsystems. It stores all the data on disk for real-time processing. The monitor software updates a lamp-bank display with current computed aerodynamic forces or coefficients every 1.25 sec and allows several FORTRAN programs to time-share the background.

The standard hardware interfaces along with the software interfaces provide the system with the capability of adapting to new environments. These interfaces allow the system to grow and to replace obsolete equipment with its current counterpart. The FORTRAN software interfaces allow FORTRAN programs generated on other facilities (such as time-sharing terminals) to be automatically adapted to the wind-tunnel real-time environments. The data that were gathered and stored in the real-time format is accessed and reformatted by the software interfaces to conform to the format required by the FORTRAN applications software.

Ames Research Center

National Aeronautics and Space Administration

Moffett Field, California 94035, November 19, 1974

REFERENCES

1. Time Data Staff: MAP, Modal (Laplace) Analysis Package: Operating Note. Time Data Corporation, Palo Alto, California, 1974. (Publication no. 1923-5085)
2. Time Data Staff: System Description Manual, TDA Time Series Analysis Systems. Time Data Corporation, Palo Alto, California, 1973. (Publication no. 1923-5025)
3. Time Data Staff: Time Series Programming Language: TSL™ Programming Manual. Time Data Corporation, Palo Alto, California, 1973. (Publication no. 1923-5030B)
4. Trover, William F.: A High Speed Airborne Data Acquisition and Control System with an Integrated Digital Computer. Proceedings of the International Telemetry Conference, vol. 9, 1973, p. 88.
5. Reynolds, Don. R.: An Integrated PCM Data System for Full Scale Aeronautics Testing. Proceedings of the 1974 International Telemetry Conference, vol. 10, pp. 347-356.
6. Finger, Herb J.; and Cambra, Joseph M.: The Computer-Controlled, On-Board Data Acquisitions System for Wind-Tunnel Testing. Proceedings of the 1974 International Telemetry Conference, vol. 10, pp. 337-346.
7. Trover, William F.: Firmware Controlled, High-Speed, Random Data Acquisition Unit. Proceedings of the 1974 International Telemetry Conference, vol. 10, pp. 312-336.
8. Prichard, Guy D.: A Central Controller/Display System with Real Time Processing for Remote Data Acquisition Units. Proceedings of the 1974 International Telemetry Conference, vol. 10, pp. 357-376.