NASA Technical Memorandum 89040

Aircraft Noise Synthesis System

David A. McCurdy and Robert E. Grandle

FEBRUARY 1987

NASA

NASA Technical Memorandum 89040

Aircraft Noise Synthesis System

David A. McCurdy and Robert E. Grandle
Langley Research Center
Hampton, Virginia

National Aeronautics
and Space Administration

Scientific and Technical
Information Branch

1987

Use of trademarks or names of manufacturers in this report does not constitute an official endorsement of such products or manufacturers, either expressed or implied, by the National Aeronautics and Space Administration.

CONTENTS

SUMMARY

A second-generation Aircraft Noise Synthesis System has been developed to provide test stimuli for studies of community annoyance to aircraft flyover noise. The computer-based system generates realistic, time-varying, audio simulations of aircraft flyover noise at a specified observer location on the ground. The synthesis takes into account the time-varying aircraft position relative to the observer; specified reference spectra consisting of broadband, narrowband, and pure-tone components; directivity patterns; Doppler shift; atmospheric effects; and ground effects. These parameters can be specified and controlled in such a way as to generate stimuli in which certain noise characteristics, such as duration or tonal content, are independently varied, while the remaining characteristics, such as broadband content, are held constant. The system can also generate simulations of the predicted noise characteristics of future aircraft, such as the proposed advanced turboprop or propfan aircraft. A description of the synthesis system and a discussion of the algorithms and methods used to generate the simulations are provided. An appendix describing the input data and providing user instructions is also included.

INTRODUCTION

Understanding and quantifying the annoyance to aircraft flyover noise experienced by an observer on the ground has been the goal of numerous laboratory and field studies over the past 50 years. Achieving this goal has proven to be difficult because of the complexity of the noise. Noise level, broadband spectral content, tonal content, noise duration, Doppler shift, temporal fluctuations, and rise time are among the interrelated characteristics which potentially affect human response to flyover noise. Additional problems are encountered when trying to predict the annoyance to flyover noise of new aircraft which are being designed and have not yet been built or flown. This report describes a research tool which was developed to aid in the study of the annoyance effects of different noise characteristics and the annoyance response to future aircraft.

Historically, psychoacoustic studies of aircraft flyover noise have employed three sources of noise stimuli in their experiments: actual aircraft flyovers, recordings of aircraft flyovers, and simulated aircraft flyover noises ranging from octave bands of broadband noise to recordings of jet-engine test-stand noise. Studies intended to quantify the effect on annoyance of an individual noise parameter often yield widely varying results because of the confounding of noise parameters, which occurs in actual and recorded aircraft flyover noise. Noise duration is an excellent example of a parameter that is subject to this problem. To obtain actual or recorded aircraft flyover noises with a wide range of durations, it is necessary to vary the distance from the takeoff/landing threshold at which the observer or microphone is located. The effects of this change in distance on duration, spectral content, and Doppler shift are illustrated in figure 1. For a given aircraft noise source, figure 1 shows the relative values of duration, spectra, and Doppler shift under the flight path at three distances from the threshold. When using actual or recorded aircraft in duration studies, long-duration stimuli have less high-frequency noise and a lower rate of change in frequency than stimuli of shorter duration. As a consequence of this inability to vary duration and other noise parameters independently, any effects on annoyance attributed to duration may, in fact, be caused by changes in spectral content or Doppler shift. The use of simulated aircraft flyover noises to avoid this confounding of parameters or to represent future aircraft can also yield inaccurate results, because these stimuli often have time histories and spectra that are unrealistic and not representative of real aircraft. Also, the simulated flyover noises often do not include important noise characteristics such as Doppler shift.

To avoid these drawbacks, an Aircraft Noise Synthesis System (ANOSS) was developed to provide test stimuli for studies of community annoyance response to aircraft flyover noise. The computer-based system generates realistic, time-varying audio simulations of aircraft flyover noise at a specified observer location on the ground. The synthesis takes into account the time-varying aircraft position relative to the observer; specified reference spectra consisting of broadband, narrowband, and pure-tone components; directivity patterns; Doppler shift; atmospheric effects; and ground effects. These parameters can be specified and controlled in such a way as to generate stimuli in which individual noise characteristics, such as duration or tonal content, are independently varied, while the remaining characteristics, such as broadband content, are held constant. The system can also use predictions of noise characteristics to generate simulations of future aircraft.

The original version of the Aircraft Noise Synthesis System was developed in 1976 by Time/Data Corporation under contract to NASA and was implemented on a Digital Equipment Corporation PDP-11/15 computer using Time Series Language (TSL) and assembly language. (See appendix A.) This first-generation synthesizer was used to generate test stimuli for three studies of aircraft flyover noise. The first study (ref. 1) compared the annoyance response to aircraft recordings with the annoyance response to recordings of synthesized simulations of the same aircraft. The effects of noise duration, tonal content, and Doppler shift were independently examined using synthesized recordings in the second study. (See ref. 2.) In addition to again studying duration and tonal content, the third study (ref. 3) used synthesized recordings with electronically induced fluctuations to determine the effects of variations in the rate and magnitude of sound-level fluctuations on the annoyance caused by aircraft flyover noise. Only synthesized stimuli were used in the second and third studies.

A number of lessons were learned from these studies about the original synthesis system. The first study led to the conclusion that the annoyance responses to the real recordings and the synthesized recordings were comparable; that is, people perceived and responded to both in the same manner. Informal discussions with test subjects immediately following their participation in the second and third studies revealed that only two out of eighty subjects realized that they were not listening to recordings of real aircraft noise. These studies confirmed the usefulness of the synthesis system. The most obvious deficiency in the synthesized flyovers was their relatively smooth time histories when compared with real aircraft flyover noise. Time histories of real aircraft flyover noise contain many sound-level fluctuations caused by atmospheric and ground effects. The models of atmospheric and ground effects in the original synthesizer did not adequately reproduce these fluctuations. The first study did not find this deficiency to be a serious shortcoming. This result from the first study was confirmed by the third study, which found that the rate and magnitude of the fluctuations did not affect annoyance.

Despite the successes achieved with the original synthesis system, a number of problems and limitations were identified. In addition to the failure of the models of atmospheric and ground effects to produce realistic sound-level fluctuations, there were some questions as to whether the broadband spectra and directivity input data were being accurately modeled and translated into the output signal. Also, an unwanted, low-level, background, crackle sound was present in the synthesized noise. The number of narrowband and pure-tone components allowed was limited, and the flight profile was restricted to one segment at constant speed. Computer capacity and speed dictated a limited noise duration and a long noise generation time of approximately 1 minute of computer time per second of flyover noise. Individually, these problems and limitations were minor, but collectively they justified revising the original synthesis system.

Modifying the original system was difficult for four reasons: users were unfamiliar with the system structure and software; source code was unavailable for many routines; limited documentation was available; and the software was written in assembly language and TSL, a language which is not commonly used. Also, the computer system was to be upgraded to a Digital Equipment Corporation PDP-11/70 computer, on which the TSL software would not run without extensive modification. For these reasons it was decided to start from the beginning and develop a second-generation synthesis system written in the more common FORTRAN language and a minimum of assembly language. The structure and modeling techniques of the original system were used as a general guide where appropriate, but the second-generation software was written from scratch with no attempt to directly translate the TSL coding of the original version.

This paper presents a general description of the second-generation Aircraft Noise Synthesis System as presently implemented at the Langley Aircraft Noise Reduction Laboratory. The computer algorithms, modeling techniques, and methodology used in the system are also discussed. Of particular interest are the fast Fourier techniques required to produce natural-sounding, listening-quality audio simulations with the specified noise characteristics. Appendix B provides user instructions for the system.

SYMBOLS AND ABBREVIATIONS

ANOSS	Aircraft Noise Synthesis System
BPI	bits per inch
D/A	digital-to-analog
DMA	direct memory access
DOS	disk operating system
F( )	function of parameters inside parentheses
FIFO	first in/first out
f	frequency, Hz
f_s	source frequency, Hz
h	distance of closest approach of aircraft to observer, m

IO	input/output
ips	inches per second
SPL	sound pressure level, dB
TSL	Time Series Language
v	aircraft velocity, m/sec
α_{peak}	peak directivity angle, deg

GENERAL DESCRIPTION

Hardware

The second-generation ANOSS hardware configuration is illustrated in figure 2. The system processor is a Digital Equipment Corporation PDP-11/70 multiuser, general-purpose computer with 1.28 megabytes of semiconductor memory running the Digital RSX-11M operating system. The system includes 176 megabytes of removable disc media and 625 megabytes of fixed disc media. Interactive terminals are used to access the system. A 45-ips, 1600-BPI digital tape drive is used to store the final digital time histories. The digital-to-analog (D/A) converter and the attenuator are used to generate analog time-history signals from the digital time histories. Time/Data designed this special-purpose D/A converter as part of the original synthesis system. It includes a 12-bit resolution converter (1 in 4096 counts) and an attenuator. The attenuator was designed into the system because of the resolution limitation of the D/A converter. With the attenuator, the range of the D/A converter could be expanded to cover a range of audio signal levels of about 90 dB without encountering the step-like output effects present in the low end of a D/A converter. Time/Data designed this device as a direct-memory-access (DMA) device for the UNIBUS of the PDP-11. This DMA interface is capable of outputting a block of up to 4096 points at a maximum rate of 100 000 points per second. The changing of the attenuator levels is synchronized with the outputting of the data points and can be changed anytime during the DMA transfer. To aid in the outputting of larger data blocks, the device also contains a four-word first-in/first-out (FIFO) buffer. This buffer allows a greater setup time at the end of the data-block output. The computer-controlled Adret frequency generator is used to control the rate at which the data points are output from the D/A converter. The VOTRAX voice synthesizer was added to the system to allow a voice identification of the noise signal to be produced during the D/A conversion of the digital time history. Although the analog signal from the D/A converter can be played through a sound reproduction system in real time, the signal is typically recorded on an analog tape recorder for later use.

Software

To make the second-generation system software easier to understand, use, and modify, it was written with four programming guidelines in mind. First, the code was written in the commonly used (and then standard) FORTRAN IV programming language. Second, a modular approach was used in the coding of the new software. Third, the code was well commented. And lastly, except for areas where speed or direct manipulation of hardware was involved, no assembly language was used. The final software package has only two modules written in assembly language. One is a fast Fourier transform subroutine and the other is the driver for the D/A converter.

Organization

The synthesis system is organized into four sections: input, spectra generation, time-history generation, and digital-to-analog conversion. Figure 3 illustrates this organization and the functions of each section. Data files are created after each of the first three sections for use in the next sections. The fourth section provides the audio signal output. Although two or more of these sections could be combined, experience indicates that this arrangement is more convenient when producing large numbers of stimuli.

The input section allows the user to enter and manipulate the 57 to 338 data values required to describe the desired aircraft flyover noise. The data can then be stored for future use and/or immediately transferred to the next ANOSS section. The spectra generation section calculates and stores the spectrum (maximum frequency of 12 500 Hz) at the observer for each time increment considered as the program steps through the simulated flyover. In calculating the spectra at the observer, this section takes into account the specified noise source characteristics, the time-varying aircraft position, and the noise propagation effects. Next, the time-history generation section of ANOSS performs a fast Fourier transform on each of the spectra to obtain a time-history segment for each time increment. The end points of adjacent time-history segments are then matched to provide a continuous time history representing the desired aircraft flyover noise. The time history is stored in digital form for further use. The digital-to-analog conversion section generates an analog signal from the stored digital time history, which can be played directly to test subjects or, as is usually the case, recorded on analog magnetic tape for later presentation.

Capabilities

The goal in developing the synthesis system was to provide as accurate and flexible a simulation of aircraft flyover noise as possible using input data that were relatively easy to determine. For this reason, it was decided to describe the aircraft by specifying noise characteristics instead of physical characteristics of the aircraft and its engines. The source noise of the aircraft is defined in terms of a broadband random-noise spectrum of one-third-octave bands, a set of narrowband random-noise spikes, and a set of pure tones. Each of these is defined at a user-specified reference distance from the aircraft. This allows noise measurements made at different distances to be used as input. Also, these reference distances allow the noise spectrum at the observer to be specified by setting the reference distances equal to the distance of closest approach of the aircraft to the observer. Directivity patterns for the broadband spectrum and each of the narrowband spikes and pure tones are also defined in the input data.

The observer can be located anywhere on the ground, and his height can be specified in the input. The flight path can be defined as one or two linear segments, and the aircraft can be accelerated or decelerated. Propagation of the noise from the aircraft to the observer takes atmospheric effects and ground effects into account.

ALGORITHMS, MODELS, AND METHODS

Input Section

The input section is designed to provide easy entry and manipulation of the input data and to detect errors in the data prior to further execution of the program. This program section is operated interactively by the user from a terminal. The input data may be entered from the terminal or from a mass storage device or from both. Error checks are performed on the data to insure that the input values are within acceptable ranges and that the assumptions and constraints of the system are not violated. The user may list the data, correct or modify the data, and store the data. When an error-free data set is ready, the input section initiates execution of the spectra generation section. The models used to define the flight path, broadband noise, narrowband noise, pure tones, and directivity are described in appendix B.

Spectra Generation Section

The spectra generation section calculates the spectra at the observer as the aircraft is flown along the specified flight path. These spectra are then stored for later conversion to the time domain. The system places three constraints on the aircraft. The aircraft is restricted to subsonic flight, the aircraft cannot fly at altitudes lower than ground level, and the noise produced by the aircraft is restricted to a frequency range from 9 Hz to 12 500 Hz. The third restriction also applies to the noise at the observer. If the Doppler effect shifts some of the noise to frequencies outside this range, that noise is not reproduced. Intermediate calculations take into account frequencies outside this range, but the final stored spectra are limited to this range. For example, when the 12 500-Hz one-third-octave band is broken into discrete frequency bands, the energy is distributed across the entire band from 11 180 Hz to 14 142 Hz.

At the beginning of the spectra generation section, a number of position and distance parameters and initial values are calculated from the input data. To simplify later calculations, the broadband random, narrowband random, and pure-tone noise levels are propagated from the specified reference distances back to a distance of 1 m from the aircraft (which is assumed to be a point source). The simulation start and stop (aircraft) times, which represent aircraft positions, are translated into observer time (i.e., the corresponding time at which the noise reaches the observer). The time increment between spectra and the discrete-frequency bandwidth are also calculated (e.g., 0.315 sec and 3.05 Hz, respectively, for a 26 000-Hz sample rate and 4096 frequency intervals). Spectra are calculated at the observer start time and at every time increment thereafter until the observer stop time is reached.

At each observer time increment, the corresponding aircraft time is calculated. Based on the aircraft time, the slant range, directivity angle, and Doppler shift factor are computed. To calculate the broadband random, narrowband random, and pure-tone noise levels at the observer, corrections for directivity, atmospheric propagation, and ground effects are applied to the noise levels defined at 1 m from the aircraft. The atmospheric-propagation corrections are based on reference 4. The ground-effects corrections are based on reference 5. The broadband and narrowband random-noise information is combined at the observer to obtain a noise spectrum calculated in terms of the discrete frequency bands. This combined noise spectrum and the pure-tone values at the observer are then stored. This process is repeated for each time increment.

Time-History Generation Section

The two preceding sections describe a method of obtaining a series of noise spectra and pure-tone descriptions to give a pseudo time history of a flyover. This section and the next section describe the methods that were used to convert this pseudo, stepped, frequency time history into a smooth, continuous, analog time history. The basic procedure followed in this conversion is to read the discrete-frequency random-noise levels and the pure-tone descriptions from the data file on disk and convert them to a digital time history on digital magnetic tape. The digital magnetic tape is then processed through the special D/A interface on the computer, and the analog output is recorded on a standard analog tape recorder. This reproduction process is split into two parts, digital time-history generation and D/A conversion, because the conversion from the frequency domain to the time domain still cannot be done in real time. The present ratio for the translation is about 10 to 1 or 10 sec of computer time for each 1 sec of analog time history. It was decided to place the digital time history on digital tape instead of the storage discs, so that large numbers of the time-history data files could be stored for long periods of time.

Although the time-history generation section of the reproduction process may appear to be a simple task, it is neither straightforward nor simple. To understand the complexity of the task, it is necessary to discuss some of the problem areas encountered in converting from the frequency domain to the time domain using the methods in this synthesis system.

First, there is insufficient phase information provided in the input to this section. The data produced by the spectra generation section for input to this section consist of random-noise spectra and pure-tone descriptions. The random-noise data are in the form of a series of discrete-frequency-band levels at fixed time intervals of approximately 0.3 sec (assuming a sampling rate of 26 000 Hz and 4096 frequency intervals) and has no phase information. This is essentially a series of small samples of the time history that is to be generated. This process would seem to be acceptable, because it is the inverse of the process of getting spectral levels from a real time history. The difference is that the phase information that is absent from the input data is necessary to reproduce a connected, nonperiodic, time-history signal. Therefore, to synthesize analog time histories from the spectral data, a procedure must be devised to generate the phase information that is not present. The constraints placed on this generation process are that the time history must not have any step discontinuities in either phase or amplitude at the end points of each segment, and that no new periodic components be added to the final time history. The pure-tone descriptions, which are specified separately, consist of both amplitude information and phase information. However, they are only specified at the start of the same discrete time interval as the random-noise data. In the final output signal, the pure-tone components must change smoothly over time in both frequency and amplitude.

The second problem area is that the human ear, though insensitive to variations in phase, is very sensitive to repetitive or periodic sounds. This means that the synthesized noise must be sufficiently random over a long time period so that the ear does not detect a pattern to the sound that might indicate to the listener that its source was artificial.

The third problem area is that, for a flyover to sound natural, the levels during the beginning and end of the flyover must not be constant. This is the problem, mentioned in the Introduction, of adequately modeling the unpredictable fluctuations in acoustic level at great distances from the source caused by atmospheric and ground effects. With all these problem areas in mind, some of the approaches that were tried and their results are discussed in the sections that follow.

Random-noise generation.- The first part of the digital time-history generation procedure is to generate the random-noise component of the acoustic waveform. This proved to be the most difficult problem to solve in the synthesis of an aircraft flyover from individual spectral time histories. As indicated previously, the levels lack the necessary phasing information to produce a continuous analog time history. Several methods were suggested to achieve this transformation, all of which used a fast Fourier transform as their basis with a random-number generator to supply the necessary phase information.

The method used in the original system software was to generate random phase angles for the discrete-frequency-band levels and then perform an inverse Fourier transform to change the resulting data into a time history. The resulting time-history segment generally does not match the preceding time-history segment in either value or slope at the end points. This difference is due to the fact that the phase information was generated in a random manner. Therefore, a procedure must be developed to match or mask these discontinuities. The original system software masked the end-point discontinuities by using the following procedure. First, during the transformation process, a cosine window would be applied to the data. This would allow a gradual change from the old signal to the new one. The transformed signal would then be summed into the output signal at approximately one-third of the transform length. To preserve the length of the time history, each transform would be summed into the time history three times. This method also had an added advantage in that the total level of the signal could be increased at a rate equal to one-third of the length of each of the segments. However, this method, though fast, has several disadvantages. First, and most important, the summing of time histories has a major problem in that what one is essentially doing is summing a series of sine and cosine waves. This has the effect of not only producing the sums of the two waveforms, but their differences and the original waves as well. Since all the sine waves are harmonics of each other, the effect of this summation is to produce unwanted levels at other lines in the spectra. Therefore, the procedure can and does occasionally produce spectra that are markedly different from those requested by the input data. Second, unless the phasing of the components of the inverse transform is chosen correctly, the change of the level of the time-history segment is unpredictable. Therefore, the only way to generate the necessary level change in the flyover using this method is to multiply each of the points in a segment by a constant factor, which is different for each segment. This generally produces a good approximation to the changing level of a flyover. However, if the rate of change of the level is great enough, the flyover root-mean-square level appears to have an almost stairstep quality. For these and several other minor reasons, it was decided that another method must be developed to simulate the random-noise time history.

The approach that was developed is outlined in the following paragraphs and is presented graphically in figure 4. Where possible, an explanation is given as to why certain choices were made from several alternatives. The starting point of the new method was the same as the original, except that no window was placed on the inverse transformed data. The original procedure showed that no method using any summation of the various time histories to mask the end-point discontinuity problem would produce a final output time history whose shape could be closely controlled. In addition, the Fourier transform, even with all of its problems of end-point matching, offered a solution for mass production of time histories from spectral data in a reasonable amount of computer time. As in the original procedure, to be able to transform the spectral data into a time history, some form of phase information must be generated for the input data. The method for generating this phase information is to first generate a series of uniformly distributed random numbers between -0.5 and +0.5. There is no real significance to the range of the numbers except that they must be uniformly distributed about zero. This random series of numbers is then transformed into the frequency domain and the desired spectrum is used as a weighting factor to shape the spectrum of the uniformly distributed random numbers. This is essentially a convolution of the desired spectral shape with a uniformly distributed random signal. This weighted spectrum is then transformed back into a time history. Despite the fact that it is important to minimize the time required to generate a time history, it is necessary to transform the signal twice for two reasons. The first reason is to get the phase information that is necessary to produce a time history. The second is that it solves most of the end-point matching problem, because the random-number generators on most machines tend to have a short-term correlation, which produces end points that are close in both magnitude and slope. This process can be compared with taking a white noise signal and passing it through a set of continuously varying narrowband filters.

This approach of generating a random time history and transforming it twice was chosen over performing a single transform with random phase information because the single transform method produces, after the digital-to-analog conversion, a time history containing a low-frequency, beating tone. The two-transform method does not produce this beating-tone phenomenon. The beating tone produced by the single transform is mainly a result of the short-term correlation that digital random-number generators tend to produce and is a result of the nature of a Fourier transform. A Fourier transform is an attempt to decompose an infinite time history into a series of pure sine waves. The fast Fourier transform can exactly represent a time history only if that time history is exactly periodic within the length of the data taken for the transform and if the waveform repeats itself with that period forever. The important point is that the inverse of any limited Fourier series is a periodic function and contains only the frequencies of the original series. Frequencies that are not periodic within the time frame of the transform have their energy spread out among all the lines of the transform, with the greatest magnitude of energy at the Fourier frequencies just less than and greater than the non-Fourier frequency. This is important in the synthesis of noise, because the sum and differences are produced when a summed pair of sine waves of different frequencies are generated. When a time-history waveform produced by taking a shaped amplitude spectrum with random phase is generated, enough of the lines have the same phase relationship to produce a pseudo tone at the difference in frequency between a pair of lines. These pseudo tones are produced at each of the multiples of the frequency differences between the lines. The most noticeable pseudo tone is at the minimum frequency difference, and this signal cannot be filtered out because it is the result of the summation process. By using the process of transforming a random time history, weighting it, and transforming again, the correlation in the phase information is avoided and the pseudo tones do not occur.

The method of transforming a random time history twice produces a reasonable end-point matching, but not an exact one. The program uses the following method to produce exact end-point matches. The program stores the last point of each output segment and then modifies the next spectrum so that the first line of the next segment has the same value. This procedure produces a small flat spot in the random noise, but it is not noticeable in the final time history. The program calculates the value of the first point of the next segment by summing all the real components of the weighted spectrum. It then randomly chooses lines from the spectrum to be used to correct the value of that first point. These chosen lines have all their energy shifted to their real component to adjust the value of the first point. In general, several lines must be used to perform this correction, because the magnitudes of these lines must not vary from their original weighted values. Changes in these magnitudes would cause a variation from the desired output spectrum. Also, during the weighting process to shape the spectrum, the first three lines of the raw random-noise spectrum are not weighted, because these lines have the greatest effect on the value of the end point. This method of shifting energy in the weighted spectra produces good end-point matching with none of the noticeable periodic noise which had been caused by other methods, such as the windows and interpolation schemes.

Up to this point in the time-history generation, only the desired shape of the random-noise spectrum has been attained. Now the amplitude of the signal must be modified to produce the predicted change in the overall noise level with time. The algorithm used to control the pattern of the overall level of the random noise is basically a linear interpolation of the overall levels from the beginning to the end of the segment. Each point in the time-history segment is multiplied by a value slightly larger or slightly smaller than the preceding value to achieve a smooth increase or decrease in the overall level of the time history. This method eliminates the step-like increases that occurred using the original approach of changing the amplitude with each of the time-history segments. This stepping phenomenon was noticeable in flyovers with very short durations and rapid level changes.

Pure-tone generation.- The second part of the digital time-history-generation procedure is to generate the pure-tone components of the acoustic waveform. The method chosen to generate these series of waveforms was a sine calculation method. The phase angle for each data point is calculated and the sine of the phase angle is taken and multiplied by the amplitude of the tone at that time. Several other methods of generating the pure-tone values were tried, but all were eliminated because of limitations on machine address space or because of the quality of the tones generated. Of the rejected methods, sine lookup tables showed the most promise of increasing the speed of tone generation, but this required too much address space in the machine. One of the most difficult problems with the pure tones is that their frequency must vary continuously during the flyover to simulate the Doppler shift experienced with a moving sound source. This continuously shifting frequency is also generally accompanied by a continuously changing amplitude. Both of these requirements were met by using a linear interpolation scheme for the amplitude and the frequency. This method necessitates calculating the value of the sine at each time point for each tone. The output of this calculation is then added to the output of the random-noise data. The resulting digital time history is then recorded on digital magnetic tape for use in the digital-to-analog conversion section of the system.

Level-fluctuation generation.- The final part of the digital time-history generation procedure is to modulate the time history to simulate the level fluctuations that occur at the beginning and end of aircraft flyover noise. To produce these fluctuations, which are caused by atmospheric and ground effects, the time history is modulated by a slowly varying function. (See fig. 5.) The amplitude of the modulation is inversely proportional to the ratio of the present root-mean-square value of the time history to the peak root-mean-square value of the time history. The modulation function is created by multiplying this amplitude function times a function derived from two sets of random numbers. The first set of random numbers is used to determine the lengths of a series of ramps of linearly varying amplitude. The maximum amplitude of each ramp is determined by a second series of random numbers between zero and one. The mean length for the ramps was chosen to be 0.3 sec. The time-history modulation is achieved by multiplying the time history by the modulation function. This form of amplitude modulation produced the desired random level fluctuations at the beginning and end of the flyover noise.

Digital-to-Analog Conversion Section

The digital-to-analog conversion process involves sending the digital time history to the specially designed digital-to-analog converter. Since the data points on tape represent evenly spaced samples of a pseudo analog time history, they must be reproduced at a constant rate with no breaks or gaps. The rate of reproduction is governed by the maximum frequency of interest in the spectrum and must be at least twice that frequency. The method used in this conversion was a double buffering scheme, where two buffers are allocated in the computer memory. Data are read into one of the buffers from the digital magnetic tape unit while the data from the other buffer are output to the D/A converter. When the data from the output buffer are exhausted, the filled input buffer is swapped with the output buffer and the process is repeated until the data on tape are exhausted. There are two limiting factors on this method of reproduction.

First, all the program steps necessary to swap the buffers must be completed between the time the D/A converter signals the completion of the emptying of the output buffer in memory and the time that the FIFO buffer in the D/A converter becomes empty. The FIFO buffer is a first-in/first-out buffer that can be loaded by the computer at a very high, but variable, rate on the input end, while delivering data to the digital-to-analog end at a constant rate. Also, the depth of the FIFO buffer (i.e., the number of data values that it will store) increases the amount of time available for setting up the next buffer for transfer to the D/A converter. Without the FIFO buffer in the system, the swapping of the buffers in memory would have to be done within the time period between the output of two data points.

The second limiting factor is that the setup time and fill time for the input buffer must be less than the setup and empty time for the output buffer. At present, the second factor is the limiting factor. The input fill time from the 45-ips, 1600-BPI tape drive restricts the reproduction rate to 26 000 Hz. The limiting rate caused by the output buffer swap time is about 80 000 Hz. As mentioned previously, this section of software is one of two pieces of software written predominantly in machine language. This was necessary because this was the only point in which hardware that was not supported by the operating system was used. Also, the operating system is a multiprogram system, and its overhead for starting an input/output (IO) operation can vary from 1 millisecond to as high as 20 milliseconds on a very busy system. If a higher level language at the program level were used to swap the buffers or to start the individual block IO, the rate of analog output would be severely restricted. Therefore, all the code necessary to perform the IO operations and buffer swaps is included as part of the code for the device driver that controls the D/A converter. This decision to include the code in the device driver allows a much higher rate of point production, but severely limits the transportability of the software to other machines. Since both the tape drive and the D/A converter are DMA devices, the impact on other computer users of placing the double buffering code in the device driver is very small.

DISCUSSION

The second-generation synthesis system is operational, and numerous simulations have been produced. Two laboratory studies have been conducted using this improved system. (See refs. 6 and 7.) In both studies, the system was used to produce simulations of advanced turboprop aircraft. The advanced turboprop or "propfan" aircraft, which is expected to enter commercial service in the 1990's, has a propeller that is vastly different from conventional propellers in shape and number of blades. In the first study, the synthesis system produced 45 simulations in which three noise characteristics were systematically and independently varied. This allowed the effects on annoyance of blade-passage frequency, blade-tip Mach number, and tone-to-broadband noise ratio to be assessed. In the second study, annoyance responses to 18 advanced turboprop simulations and 10 recordings of conventional jets and turboprops were compared. Informal discussions with the subjects in the second study indicated that none of the subjects realized that some of the noises were artificial.

The second-generation synthesis system has also been used to generate stimuli for an aircraft interior noise study. By making minor modifications to the software to delete the Doppler shift and ground-effects correction, it was possible to produce interior noise stimuli for use in a study of the effects on annoyance of multiengine propeller beating phenomena.

CONCLUDING REMARKS

A second-generation Aircraft Noise Synthesis System has been developed to provide test stimuli for studies of community annoyance to aircraft flyover noise. The computer-based system generates realistic, time-varying, audio simulations of aircraft flyover noise at a specified observer location on the ground. The synthesis takes into account the time-varying aircraft position relative to the observer; specified reference spectra consisting of broadband, narrowband, and pure-tone components; directivity patterns; Doppler shift; atmospheric effects; and ground effects.

Based on the experience gained from experimental studies, the second-generation synthesis system has proven to be very successful. The system is easier to use, easier to modify, and produces a higher quality representation of the aircraft flyover noise than the first-generation system. It has been particularly useful in simulating the noise of future aircraft and in isolating the annoyance effects of individual noise characteristics.

NASA Langley Research Center
Hampton, VA 23665-5225
December 22, 1986

APPENDIX A

DESCRIPTION OF ORIGINAL AIRCRAFT NOISE SYNTHESIS SYSTEM

The original version of the Aircraft Noise Synthesis System program became operational in 1976. It was written for a Time/Data Corporation time-series-analysis computer system under a contract to the NASA Langley Research Center. This time-series-analysis machine was based on a Digital Equipment Corporation PDP-11/15 minicomputer (the 11/15 was an original equipment manufacturer version of the PDP-11/20) in conjunction with a proprietary operating system supplied by Time/Data known as the Time Series Language (TSL).

The PDP-11/15 system consisted of the following peripherals: two 2.5-megabyte removable disc drives, a line printer, a high-speed paper tape punch and reader, and a Tektronix 4010 Graphics terminal. As part of the contract for the noise synthesis work, Time/Data designed a special-purpose digital-to-analog (D/A) converter to output the analog time-history signals. This D/A converter consisted of one 12-bit-resolution converter (1 in 4096 counts) with 1 master attenuator and 10 separate output attenuators. The master attenuator was designed into the system because of the resolution limitation of the D/A converter. With the master attenuator, the range of the D/A converter could be expanded to cover an audio-signal-level range of about 90 dB without encountering the step-like output effects present in the low end of a D/A converter. The 10 separately controlled output attenuators were to be used to provide separate outputs in a listening facility to produce the appearance of motion through control of the directivity of the sound. Time/Data designed this device as a direct-memory-access (DMA) device for the UNIBUS of the PDP-11. This DMA interface was capable of outputting a block of up to 4096 points at a maximum rate of 100 000 Hz. The changing of the attenuator levels was synchronized with the outputting of the data points and could be changed anytime during the DMA transfer. To aid in the outputting of larger data blocks, the device also contained a four-word first-in/first-out (FIFO) buffer. This buffer allowed a greater setup time at the end of the data-block output.

The operating-system software, TSL, was a modified form of the standard Dartmouth Basic offered by Digital Equipment Corporation in 1976 with most of the linear algebraic operations removed and replaced by array functions. Also, the functions necessary to perform time-series analysis, such as fast Fourier transforms and autocorrelations and cross correlations, had been added to the Basic. Most of the routines for the noise synthesis program were written in the TSL language, but some of the proprietary subroutines were written in assembly language. Algebraic operations on scalar variables were done through the use of a stack operation in reverse Polish notation.

APPENDIX B

USER INSTRUCTIONS FOR AIRCRAFT NOISE SYNTHESIS SYSTEM

The synthesis system consists of four sections divided among three programs: ANOSS, MAKES, and MUSIC. As illustrated in figure 6, the input and spectra generation sections are in the ANOSS program. The MAKES program is the time-history generation section and MUSIC is the digital-to-analog conversion section.

Input File Generation

The input section allows the user to enter, manipulate, and store the data required to describe the desired aircraft flyover noise. Figure 7 is a flow chart of the input section. After logging on to the appropriate computer account, the program is initiated with the command:

```
RUN ANOSS
```

The program responds with a series of questions which direct the program to the appropriate subroutines. The first question is:

```
ENTER PARAMETERS (Y/N)?
```

A positive (Y) response calls the ENTER subroutine, which allows the input data to be entered from a storage device or from the keyboard. After the data are entered, the CHECK subroutine is called to perform a number of checks on the data to insure that the specified simulation represents a realistic situation (e.g., the aircraft does not fly into the ground). A negative (N) response sends the program directly to the CHECK subroutine. After the CHECK subroutine, the program asks:

```
LIST INPUT (Y/N)?
```

If answered with a positive (Y) response, subroutine LIST displays the input data on the terminal so that a hard copy may be obtained. The program then asks:

```
CORRECTIONS (Y/N)?
```

If answered with a positive (Y) response or if subroutine CHECK detected errors in the input data, subroutine CORRECT is called. This subroutine allows the user to change individual values in the data set. When the corrections are completed, the CHECK subroutine is called again and the "list" and "corrections" questions are repeated. A negative (N) response sends the program to the next question:

```
COPY INPUT (Y/N)?
```

The COPY subroutine is called when this question is answered with a positive (Y) response. COPY allows the user to save the data on a storage device for later use or modification. The program then asks:

```
SET UP COMPLETE (Y/N)?
```

A positive (Y) response terminates the input section of ANOSS and begins the spectra generation section. A negative (N) response sends the program back to the "enter parameters" question, so that the user can continue to manipulate the input data.

The subroutines are further described in the following sections.

ENTER subroutine.- The input data are divided into nine parts. The ENTER subroutine is used to enter the nine parts as a group or individually, either from the keyboard or a storage device. The option to enter parts individually allows the user to combine parts of different data sets that have been previously stored. The subroutine first asks:

```
WHICH PART OF PARAMETERS DO YOU WISH TO ENTER?

(IF NONE, TYPE IN 0; IF ALL, TYPE IN 10):
```

To enter an entire set of data, the user should respond with a "10". Otherwise, the user types in the number, "1" through "9", of the part he wishes to enter. A "0" response skips the remainder of ENTER and proceeds to the CHECK subroutine. If the response is not "0" the subroutine requests

```
ENTER LOCATION OF DATA FOR THIS PART (XXnn:FILENAME.EXT).

(e.g. TERMINAL= TI: ; SYSTEM DISK= FILENAME.EXT ):
```

To enter data item by item from the keyboard, the response is "TI:". To read stored data the user specifies the storage device (XXnn:) and the file name and extension (filename.ext) of the data file.

The input values required in each of the parts of the data set are described in the following paragraphs.

```
PART 1.  IDENTIFICATION

 1.  HEADING (MAXIMUM OF 40 ALPHANUMERIC CHARACTERS):
```

The heading allows the user to enter a simple description of the data file.

PART 2. OBSERVER LOCATION

1. SIDELINE DISTANCE, METERS:

2. DOWNRANGE DISTANCE, METERS:

3. OBSERVER HEIGHT, METERS (>0.):

Figure 8 illustrates the observer location relative to the flight path of the aircraft. The sideline distance is the perpendicular distance from the observer to the ground track of the aircraft. (The program assumes that the ground track is a straight line.) No distinction is made as to which side of the ground track the observer is on. The downrange distance is the distance along the ground track from the ground contact point to the point at which a perpendicular line from the observer intersects the ground track. Positive downrange distances place the observer under the flight path. Negative values for downrange distances place the observer uprange of the ground contact point. Observer height specifies how far above the ground the listening apparatus (i.e., the ear or microphone) is located. Observer height must be greater than zero.

PART 3. FLIGHT PATH

1. NUMBER OF FLIGHT PATH SEGMENTS (1. OR 2.):

2. AIRCRAFT GROUND SPEED, METERS/SEC (0.<=VO<=340.):

3. ANGLE OF SEGMENT A, DEGREES (0.<=PHIA<=90.):

4. AIRCRAFT ACCELERATION ON SEGMENT A, METERS/SEC**2:

5. TRANSITION DISTANCE, METERS (>0.):

6. ANGLE OF SEGMENT B, DEGREES (-90.<=PHIB<=90.):

7. AIRCRAFT ACCELERATION ON SEGMENT B, METERS/SEC**2:

The aircraft flight path can consist of either one or two linear segments. If one segment is selected, the program requests parameters 2 through 4 of Part 3. If a two-segment path is selected, the program also requests parameters 5 through 7. A two-segment flight path is illustrated in figure 8. (A one-segment path would be illustrated by segment A alone, with the simulation outer point lying on segment A.) The aircraft ground speed is the speed of the aircraft at the ground contact point (i.e., the speed at which the aircraft leaves the ground during takeoffs or touches down during landings) and is limited to the subsonic range, 0 to 340 m/sec. The angles of segments A and B are the angles between the ground track and the segments. Segment A is limited to 0° to 90°, and segment B can range from -90° to 90°. Aircraft accelerations can be specified for each segment; however, the program rejects values which accelerate the aircraft to supersonic speeds (>340 m) or decelerate it to negative speeds during the time period of the simulation. The transition distance

(>0 m) is the distance along the ground track from the ground contact point to the point above which the aircraft moves from one segment to the other.

```
PART 4.  BROADBAND NOISE SPECTRUM

 1.  REFERENCE DISTANCE, METERS (>0.):

 2.  REFERENCE SPL, DB:

 3.  RELATIVE SPL OF    10.0 HZ BAND, DB:

 4.  RELATIVE SPL OF    12.5 HZ BAND, DB:

 5.  RELATIVE SPL OF    16.0 HZ BAND, DB:
     .
     .
     .

32.  RELATIVE SPL OF  8000.0 HZ BAND, DB:

33.  RELATIVE SPL OF 10000.0 HZ BAND, DB:

34.  RELATIVE SPL OF 12500.0 HZ BAND, DB:

35.  DIRECTIVITY PEAK ANGLE, DEGREES (0.<=AP<=180.):

36.  ATTENUATION AT   0 DEGREES TO JET AXIS, DB (>=0.):

37.  ATTENUATION AT  90 DEGREES TO JET AXIS, DB (>=0.):

38.  ATTENUATION AT 180 DEGREES TO JET AXIS, DB (>=0.):
```

The broadband random-noise spectrum is defined at a point 90° from the jet axis (flight path) and is assumed to be symmetric about the jet axis. The reference distance (>0 m) is the distance from the aircraft to the point at which the one-third-octave-band levels are specified. This is illustrated in figure 9. The reference-distance parameter allows flexibility, in that actual noise measurements made at some known distance from the aircraft can be used or the user may specify the actual noise at an observer by setting the reference distance equal to the distance of closest approach of the aircraft to the observer as defined by the flight path. The one-third-octave-band levels are specified in terms of one reference sound pressure level and a relative sound pressure level for each of the bands from 10 Hz to 12 500 Hz. The actual level for a given band is the sum of the reference SPL and the relative SPL for the band. This arrangement allows flexibility, in that the relative SPL's can be set equal to the actual SPL's (Reference SPL = 0), or the relative SPL's can be set relative to a single level given by the reference level (e.g., Reference SPL = 100, relative SPL of 10-Hz band = -10,..., relative SPL of 1000-Hz band = 0,..., relative SPL of 12 500-Hz band = -15). In either case, when modifying a data set, all the band levels can be adjusted by a constant amount by just changing the reference SPL.

A directivity pattern about the pitch axis is specified for the broadband spectrum. Figure 9 defines directivity angle in relation to the jet axis. The directivity peak angle is the angle at which the peak noise level occurs; this angle can range from 0° to 180°. The attenuations at 0°, 90°, and 180° to the jet axis are the differences in level between the levels at 0°, 90°, and 180° and the level at the directivity peak angle. The attenuations are defined to be greater than or equal to zero. If the directivity peak angle is set equal to 0°, 90°, or 180°, the attenuation at that angle must be set equal to zero. A set of sine and cosine equations are used to define the directivity pattern between the four specified points. Figure 10 shows two typical directivity patterns. (The directivity corrections based on these equations are adjusted by a constant, so that the correction applied at 90° to the jet axis equals zero. This adjustment is made to avoid changing the noise levels specified at 90° in the input data.)

```
PART 5.  NARROWBAND NOISE COMPONENTS

 1.  ARE NARROWBAND NOISE COMPONENTS REQUIRED (Y/N)?

 2.  REFERENCE DISTANCE, METERS (>0.):

 3.  REFERENCE SPL, DB:

 4.  NUMBER OF COMPONENTS (1.<=RIC<=20.):

   ENTER THE FOLLOWING PARAMETERS FOR EACH NARROWBAND

   COMPONENT.

   (SEPARATE PARAMETERS WITH COMMAS, i.e. A,B,C,D,E,F,G):

    A.  CENTER FREQUENCY, HZ (9.<=XLST(I,1)<=12500.)

    B.  RELATIVE SPL, DB

    C.  BANDWIDTH, HZ (>0.)

    D.  DIRECTIVITY PEAK ANGLE, DEGREES

         (0.<=XLST(I,4)<=180.)

    E.  ATTENUATION AT   0 DEGREES TO JET AXIS, DB (>=0.)

    F.  ATTENUATION AT  90 DEGREES TO JET AXIS, DB (>=0.)

    G.  ATTENUATION AT 180 DEGREES TO JET AXIS, DB (>=0.)

 5.  COMPONENT NO.  1. :

 6.  COMPONENT NO.  2. :
       .
       .
       .
```

Since the narrowband random-noise components are optional, the user must first indicate whether or not they are to be included. A negative (N) response to line 1 sends the program on to Part 6. A positive (Y) response causes the program to request the parameters describing the narrowband components. The reference distance and reference SPL are defined in the same manner as those for the broadband spectrum in Part 4, but they may have different values. From 1 to 20 narrowband components may be specified. Each component is described in terms of seven parameters. The center frequency defines the center of the component and may range from 9 to 12 500 Hz. The narrowband component is modeled as a triangular power function about the center frequency. The relative SPL is also defined in the same manner as in Part 4 and interacts with the reference SPL in the same way. The sum of the reference SPL and the relative SPL is defined to be the summed level of the entire component (i.e., the overall level if the component were the only noise present). The bandwidth is the width of the narrowband component at 3 dB down from the center frequency; the bandwidth is always greater than zero. In terms of the triangular power model, the bandwidth is the width at which the power is reduced by half. The directivity peak angle and the attenuations at 0°, 90°, and 180° to the jet axis are defined in the same way as in Part 4. A different directivity pattern may be specified for each component.

```
          PART 6.  PURE TONE COMPONENTS

           1.  ARE PURE TONE COMPONENTS REQUIRED (Y/N)?

           2.  REFERENCE DISTANCE, METERS (>0.):

           3.  REFERENCE SPL, DB:

           4.  NUMBER OF COMPONENTS (1.<=RID<=20.):

              ENTER THE FOLLOWING PARAMETERS FOR EACH PURE TONE

              COMPONENT

              (SEPARATE PARAMETERS WITH COMMAS, i.e. A,B,C,D,E,F,G):

               A.  FREQUENCY, HZ (9.<=YLST(I,1)<=12500.)

               B.  RELATIVE SPL, DB

               C.  PHASE, DEGREES

               D.  DIRECTIVITY PEAK ANGLE, DEGREES

                     (0.<=YLST(I,4)<=180.)

               E.  ATTENUATION AT   0 DEGREES TO JET AXIS, DB (>=0.)

               F.  ATTENUATION AT  90 DEGREES TO JET AXIS, DB (>=0.)

               G.  ATTENUATION AT 180 DEGREES TO JET AXIS, DB (>=0.)

           5.  COMPONENT NO.  1. :

           6.  COMPONENT NO.  2. :
                .
                .
                .
```

The pure-tone components are defined in exactly the same manner as the narrowband random-noise components, except that a phase angle is specified instead of a bandwidth. The reference distance, reference SPL, and directivity patterns can be different from those used to define the other noise components.

PART 7. PULSE COMPONENTS

1. ARE PULSE COMPONENTS REQUIRED (Y/N)?

This section is included for future expansion of the synthesis system. Only a negative (N) response is accepted.

PART 8. GROUND ATTENUATION PARAMETERS

1. GROUND FLOW RESISTANCE, KILOGRAMS/SEC*METERS**3

   (100,000<=GFR<=500,000):

2. INCOHERENCE CONSTANT (0.001<=REFCO<=0.1):

3. NOISE SOURCE HEIGHT, METERS (>0.):

The ground flow resistance is the specific flow resistance of the ground. (See refs. 5 and 8.) The value, which varies with the type of surface, ranges from 100 000 to 500 000 kg/sec-m^3; a value of 150 000 is typical. The incoherence constant, which ranges from 0.001 to 0.1, is an empirical quantity which limits cancellation effects. (See ref. 5.) Cancellation effects decrease as the constant increases; a typical value is 0.01. The noise source height (>0 m) is the distance between the ground and the noise source (e.g., the engine) when the aircraft is on the ground. The primary purpose of this parameter is to prevent an error from occurring in the ground-effects subroutine. Therefore, any value greater than zero is acceptable, and a representative value is not necessary (i.e., using a value such as 0.01 m avoids the error and does not impact the flight-path calculations).

PART 9. SYNTHESIS PARAMETERS

1. START TIME, SEC (>=0. FOR TAKEOFFS,

   <0. FOR LANDINGS):

2. LENGTH OF FLIGHT, SEC (>0.):

3. LENGTH OF TAPER, SEC (>=0.):

4. SAMPLE RATE, HZ (>0.):

5. NUMBER OF FREQUENCY INTERVALS (1024., 2048., OR

   4096.):

Time is defined to be zero when the aircraft is at the ground contact point (i.e., the point at which the aircraft leaves the ground during takeoffs or touches down during landings). The start-time parameter is greater than zero for takeoffs and less than zero for landings. Length of flight (>0 sec) is the amount of time the aircraft is flown along the flight path for the simulation. For example, a start time of 10 sec with a 30-sec length of flight simulates a takeoff from a point 10 sec after the aircraft leaves the ground to the point where the aircraft is located 40 sec after it leaves the ground. A start time of -50 sec with a 45-sec length of flight simulates a landing from a point 50 sec prior to touchdown to a point 5 sec prior to touchdown.

To avoid an abrupt start and stop of the noise simulation, the synthesis program allows the noise level to be tapered up from zero to the calculated value at the beginning of the simulation and back down to zero at the end of the simulation. Length of taper (>=0 sec) is the amount of time over which each taper is done. The length of taper must be less than or equal to one-half the duration of the simulated noise. The noise duration may be slightly different from the specified length of flight, since the distance between the aircraft and observer usually varies during the simulation.

The sample rate is the number of points produced per second in the time domain. It should be set to slightly more than twice the maximum frequency to be reproduced. Since the system is programmed for a maximum frequency of 12 500 Hz, the sample rate is usually set to 26 000 Hz.

The number of frequency intervals is the number of intervals into which the frequency range is divided (i.e., 12 500 is divided by 1024, 2048, or 4096). The number of intervals, which must be a power of two, is limited to 1024, 2048, or 4096. The best frequency resolution is obtained by using 4096, which defines a point every 3.05 Hz. However, the higher the resolution, the longer the program takes to execute.

CHECK subroutine.- This subroutine checks for errors that occur from the interaction of different input parameters. An error message is printed for each error that occurs in the data. Each of these messages, along with an explanation, is listed in this section.

```
ERROR 1:  TIME OF SIMULATION END FOR LANDING IS GREATER THAN

          ZERO.
```

The length of flight is greater than the simulation time allowed by the landing start time (e.g., Start time = -30 sec, Length of flight = 40 sec). This results in the aircraft immediately taking off after touching down.

```
ERROR 2:  GROUND VELOCITY AND A-SEGMENT ACCELERATION ARE

          BOTH ZERO.
```

The aircraft ground speed and the aircraft acceleration on segment A are both zero. This results in the aircraft not moving.

ERROR 3: VELOCITY AT POINT 3 IS LESS THAN ZERO.

The aircraft speed at the simulation outer point (i.e., point 3, the simulation end point for takeoffs or the simulation start point for landings) is less than zero. The aircraft cannot fly at negative speeds.

ERROR 4: VELOCITY AT POINT 3 IS GREATER THAN OR EQUAL TO THE SPEED OF SOUND.

The aircraft speed at the simulation outer point (i.e., point 3, the simulation end point for takeoffs or the simulation start point for landings) is supersonic. The program is limited to subsonic speeds.

ERROR 5: SQUARE ROOT RADICAL IN T2 CALCULATIONS IS NEGATIVE. INPUT ERROR CHECK SUBROUTINE TERMINATED EARLY.

When a two-segment flight path is specified, this error check insures that a square root of a negative number does not occur in the quadratic-equation solution used to determine the time at which the aircraft is at the transition point (point 2). This error terminates the subroutine immediately.

ERROR 6: VELOCITY AT POINT 2 IS LESS THAN ZERO.

The aircraft speed at the transition point of a two-segment flight path (i.e., point 2, the point at which segments A and B meet) is less than zero. The aircraft cannot fly at negative speeds.

ERROR 7: VELOCITY AT POINT 2 IS GREATER THAN OR EQUAL TO THE SPEED OF SOUND.

The aircraft speed at the transition point of a two-segment flight path (i.e., point 2, the point at which segments A and B meet) is supersonic. The program is limited to subsonic speeds.

ERROR 8: VELOCITY AT POINT 3 AND B-SEGMENT ACCELERATION ARE BOTH EQUAL TO ZERO.

A two-segment flight path is specified, and the aircraft speed at the simulation outer point (i.e., point 3, the simulation end point for takeoffs or the simulation start point for landings) and the aircraft acceleration on segment B are both zero. This results in the aircraft not moving.

ERROR 9: ALTITUDE AT POINT 3 IS LESS THAN THE NOISE SOURCE HEIGHT.

The aircraft altitude at the simulation outer point (i.e., point 3, the simulation end point for takeoffs or the simulation start point for landings) is less than the noise source height. The aircraft is below ground level.

ERROR 10: SPECIFIED BROADBAND DIRECTIVITY ATTENUATION IS NOT EQUAL TO ZERO AT SPECIFIED PEAK ANGLE OF 0., 90., OR 180. DEGREES.

By definition, the directivity attenuation at the specified directivity peak angle is zero. If the directivity peak angle is specified to be one of the three angles (0°, 90°, or 180° to the jet axis) at which attenuations are specified, the attenuation specified for that angle must be zero.

ERROR 11: NARROWBAND COMPONENT NO. x : SPECIFIED DIRECTIVITY ATTENUATION IS NOT EQUAL TO ZERO AT SPECIFIED PEAK ANGLE OF 0., 90., OR 180. DEGREES.

By definition, the directivity attenuation at the specified directivity peak angle is zero. If the directivity peak angle is specified to be one of the three angles (0°, 90°, or 180° to the jet axis) at which attenuations are specified, the attenuation specified for that angle must be zero.

ERROR 12: PURE TONE COMPONENT NO. x : SPECIFIED DIRECTIVITY ATTENUATION IS NOT EQUAL TO ZERO AT SPECIFIED PEAK ANGLE OF 0., 90., OR 180. DEGREES.

By definition, the directivity attenuation at the specified directivity peak angle is zero. If the directivity peak angle is specified to be one of the three angles (0°, 90°, or 180° to the jet axis) at which attenuations are specified, the attenuation specified for that angle must be zero.

```
ERROR 13:  SPECIFIED LENGTH OF TAPER IS GREATER THAN ONE

             HALF THE TOTAL TIME THAT THE NOISE ARRIVES AT

             THE OBSERVER.
```

The program requires that the noise level be allowed to reach its full value. Therefore, the length of taper must be less than or equal to one-half the duration of the simulated noise. The noise duration may be slightly different from the specified length of flight, because the distance between the aircraft and observer usually varies during the simulation.

When a two-segment flight path is specified, one of the following notes is printed to inform the user as to which flight segments are included in the specified simulation times.

```
NOTE 1:  SPECIFIED TIME LIMITS SIMULATE FLIGHT ON BOTH

           SEGMENTS A AND B.

NOTE 2:  SPECIFIED TIME LIMITS SIMULATE FLIGHT ON

           SEGMENT A ONLY.

NOTE 3:  SPECIFIED TIME LIMITS SIMULATE FLIGHT ON

           SEGMENT B ONLY.
```

LIST subroutine.- The LIST subroutine displays the input data on the terminal in a format similar to that used by the ENTER subroutine. A hard copy of the input data can then be made from the screen.

CORRECT subroutine.- The CORRECT subroutine allows the user to change individual input values by specifying the associated part and line numbers. These changes can be made in any order. The subroutine first asks which part of the input is to be changed.

```
ENTER PART OF PARAMETERS (1-9) YOU WISH TO CORRECT (IF

FINISHED WITH CORRECTIONS, ENTER 0):
```

A response of "0" completes the correction routine, and the CHECK subroutine is called. A response of "1" through "9" causes the subroutine to ask for the line number of the input value to be changed.

```
ENTER NUMBER OF LINE YOU WISH TO CORRECT (IF NONE,

ENTER 0):
```

A zero response causes the program to ask again for a part number. Otherwise, the subroutine prints out the part and line description (as used in subroutine ENTER) and waits for a new value of the parameter to be entered. After the new value is entered, the program requests the part number of the next input value to be changed. This is repeated until a zero response is given to the part-number request.

If the number of flight-path segments is changed, the subroutine automatically deletes the extra input (from two segments to one) or asks for the needed additional input (from one segment to two).

Similarly, if the requirement for narrowband random-noise components or pure-tone components is changed (part 5 or 6, line 1), the subroutine deletes the component descriptions (changed from required to not required) or requests the input values that are needed to describe the components (changed from not required to required).

If the number of narrowband random-noise components or pure-tone components is changed (part 5 or 6, line 4), the subroutine deletes or adds components as required. If the number of components is reduced, the subroutine asks which components (by line number) are to be deleted. If the number of components is increased, the subroutine requests the input values that are needed to describe each additional component.

COPY subroutine.- The COPY subroutine allows the user to save the input data on a storage device for later use or modification. The subroutine first asks where to store the data.

```
WHERE DO YOU WANT DATA COPIED (XXnn:FILENAME.EXT)?

(e.g. SYSTEM DISK= FILENAME.EXT ):
```

The format used to specify the storage location is "XXnn:filename.ext", where "XXnn:" is the device specification and "filename.ext" is the name of the file and extension chosen by the user. Next, the subroutine asks which part of the input parameters is to be stored.

```
WHICH PART OF PARAMETERS (1-9) DO YOU WISH TO COPY?

(IF NONE, TYPE IN 0; IF ALL, TYPE IN 10):
```

The subroutine allows the user to store one or more individual parts of the input parameters at the specified location. A response of "1" through "9" causes the subroutine to store that part and then repeat the question, so that additional parts can be stored. A response of "10" causes the subroutine to store all 9 parts of the input. After a "0" or "10" response, the subroutine asks if the user wants to store the data on another device or file.

```
DO YOU WANT TO COPY DATA TO ANOTHER LOCATION (Y/N)?
```

A negative (N) response ends the subroutine. A positive (Y) response causes the subroutine to cycle and request another storage location.

Spectra File Generation

The spectra generation section calculates and stores a series of spectra representing the noise at the observer over the duration of the flight. Only three user inputs are required in this section. First, the program requests the storage location for the spectra file.

```
ENTER SPECTRA FILE LOCATION (XXnn:FILENAME.EXT)

(e.g. SYSTEM DISK= FILENAME.EXT ):
```

The format used to specify the storage location is "XXnn:filename.ext", where "XXnn:" is the device specification and "filename.ext" is the name of the file and extension chosen by the user. If the same device is used, the filename or extension of the spectra file should be different from that of the input file. Otherwise, the input file could be destroyed. Next, the program asks

```
DO YOU WANT EXTENDED PRINTOUT (0) OR SHORT FORM

PRINTOUT (1)?
```

As the spectra are generated and stored, the program displays the values of parameters describing the initial arrangement and each of the spectra. The answer to this question determines how many parameters are displayed. A "0" response displays all the parameters that are available. A "1" response displays a smaller set of parameters. This output is used primarily when modifying the program, but it does give an indication of the progress of the program.

Also, for the purpose of checking program modifications, the program gives the user the option of displaying the pressure distribution of the final discrete-frequency random-noise spectrum.

```
DO YOU WANT FINAL P(K) PRINTED OUT (Y/N)?
```

Since each spectrum contains a number of pressure values equal to the number of frequency intervals (1024, 2048, or 4096), the user normally answers with a negative (N) response to this question.

When the program is finished generating the spectra file, it displays

```
NORMAL END OF ANOSS PROGRAM.
```

This concludes the spectra generation section and the ANOSS program.

Time-History File Generation

The time-history generation section converts from the frequency domain to the time domain by performing a Fourier transform on the data in the spectra file. A file containing the digital time history of the noise at the observer is created on magnetic tape. After logging on to the appropriate computer account, and prior to starting the program, the user must place a digital, 0.5-in., 9-track, magnetic tape on the tape drive. After physically putting the tape on the drive, the user must mount the drive to the user's account and terminal with the appropriate computer-system command. For the current system, the command is

```
MOU MM:/FOR
```

The digital tape on the drive must be initialized as a Disk Operating System (DOS) formatted tape at 1600 BPI. A previously initialized ANOSS data tape need not be reinitialized. If a previously initialized ANOSS data tape is not used, a new tape should be used and must be initialized. The following commands are used to initialize a new tape on the current system. The first command is

```
FLX
```

The computer responds with the prompt

```
FLX>
```

In reply to the prompt, the second command is given:

```
MM:/DO/DNS:1600/ZE
```

The tape is now ready for use with the program.

The program is initiated with the command

```
RUN MAKES
```

The program responds by asking for the location of the spectra file.

```
Enter spectra file name :
```

This is the file specified by the user in program ANOSS for storing the calculated spectra. The format used is "XXnn:filename.ext", where "XXnn:" is the device specification and "filename.ext" is the name of the file and extension chosen by the user.

After the file name is entered, the generation of the time history begins. As the time history is generated, the program displays the values of several parameters associated with the transformation of each spectrum. This output is used primarily when modifying the program, but does give an indication of the progress of the program. When calculation of the time history is finished, the program displays

```
MOUNT TAPE PLEASE
```

To insure proper operation of the program and tape drive, the tape should be, as indicated previously, mounted and initialized prior to starting the program. Attempting to mount and initialize the tape at this point is usually unsuccessful and results in the loss of the data generated.

When the tape drive is ready, the program displays

```
WRITING HEADER
```

The time history is then written to the tape, and the MAKES program ends. The program can be repeated and additional time histories can be stored on the same tape. A computer-system command file or batch job can be created to automatically make multiple runs of this program using different spectra files. To do this, a privileged user must install the MAKES program as a known task to the system. The command for this installation is:

```
INS MAKES/TASK=...MAK
```

The following line can then be placed in a command procedure or a batch job for each file to be done:

```
MAK XXnn:filename.ext
```

As previously defined, "XXnn:filename.ext" is the file specification. The number of runs in a command file or batch job is limited only by the capacity of the digital magnetic tape.

Analog Signal Generation

The digital-to-analog conversion section uses a digital-to-analog converter to produce an analog noise signal from the digital time history stored on magnetic tape. This analog signal can be played in real time through a sound reproduction system and/or recorded for later use. The magnetic tape on which the time-history file is stored is placed on the tape drive and, after the user logs on to the appropriate computer account, the drive is mounted to the user's account and terminal using the command

```
MOU MM:/FOR
```

Before the program is used, certain devices must be correctly connected and set to the remote mode. The frequency synthesizer must be on, set to remote, and its output connected to the clock input of the digital-to-analog converter. The voice synthesizer input must be connected to a terminal port on the computer, and its output must be connected to a channel of the analog tape recorder. The output of the digital-to-analog converter must be connected to another of the analog tape recorder channels. Once these devices are properly connected and set up and the digital tape is mounted on the tape drive, the program is initiated with the command

```
RUN MUSIC
```

The time histories on the tape are each identified by a number representing their order on the tape. The program asks

```
WHICH FILE # ON TAPE ?
```

The user responds with the file number and the program asks

```
WHAT FREQUENCY ?
```

This question refers to the sample rate (in Hz) specified in the input section (part 9, line 4). For the analog noise signal to have the correct frequency range and duration, the number entered here must be the same as that used in the input data. (Entering a different frequency here causes a shift of all noise frequencies by the ratio of the input sample rate frequency to the frequency entered here. It also results in a change in the length of the flyover by the same ratio.) After the number is entered, the program locates the file on tape and displays

```
READY TO PLAY HIT CR TO START
```

To play the analog noise signal, the user presses the carriage return key. The program then displays some information describing the file and plays the file. When the file is finished playing, the program cycles and again requests a file number. The program is terminated by pressing <control> and <Z> at the same time.

Consecutive flyovers on the digital tape may be played during one program run by using the program MUSIC2 in place of MUSIC. Instead of asking which file number, MUSIC2 asks how many files are to be played. The program starts with the next file found on the digital tape.

REFERENCES

1. Mabry, J. E.; and Sullivan, B. M.: Response to Actual and Simulated Recordings of Conventional Takeoff and Landing Jet Aircraft. NASA CR-145318, 1978.

2. McCurdy, David A.; and Powell, Clemans A.: Effects of Duration and Other Noise Characteristics on the Annoyance Caused by Aircraft-Flyover Noise. NASA TP-1386, 1979.

3. McCurdy, David A.: Effects of Sound Level Fluctuations on Annoyance Caused by Aircraft-Flyover Noise. NASA TP-1576, 1979.

4. American National Standard Method for the Calculation of the Absorption of Sound by the Atmosphere. ANSI S1.26-1978 (ASA 23-1978), American Natl. Standards Inst., Inc., June 23, 1978.

5. Zorumski, William E.: Aircraft Noise Prediction Program Theoretical Manual. NASA TM-83199, 1982.

6. McCurdy, David A.: Quantification of Advanced Turboprop Aircraft Flyover Noise Annoyance. AIAA-84-2293, Oct. 1984.

7. McCurdy, David A.: Comparison of Advanced Turboprop and Conventional Jet and Propeller Aircraft Flyover Noise Annoyance: Preliminary Results. NASA TM-87637, 1985.

8. Delany, M. E.; and Bazley, E. N.: Acoustical Properties of Fibrous Absorbent Materials. Appl. Acoust., vol. 3, no. 2, Apr. 1970, pp. 105-116.

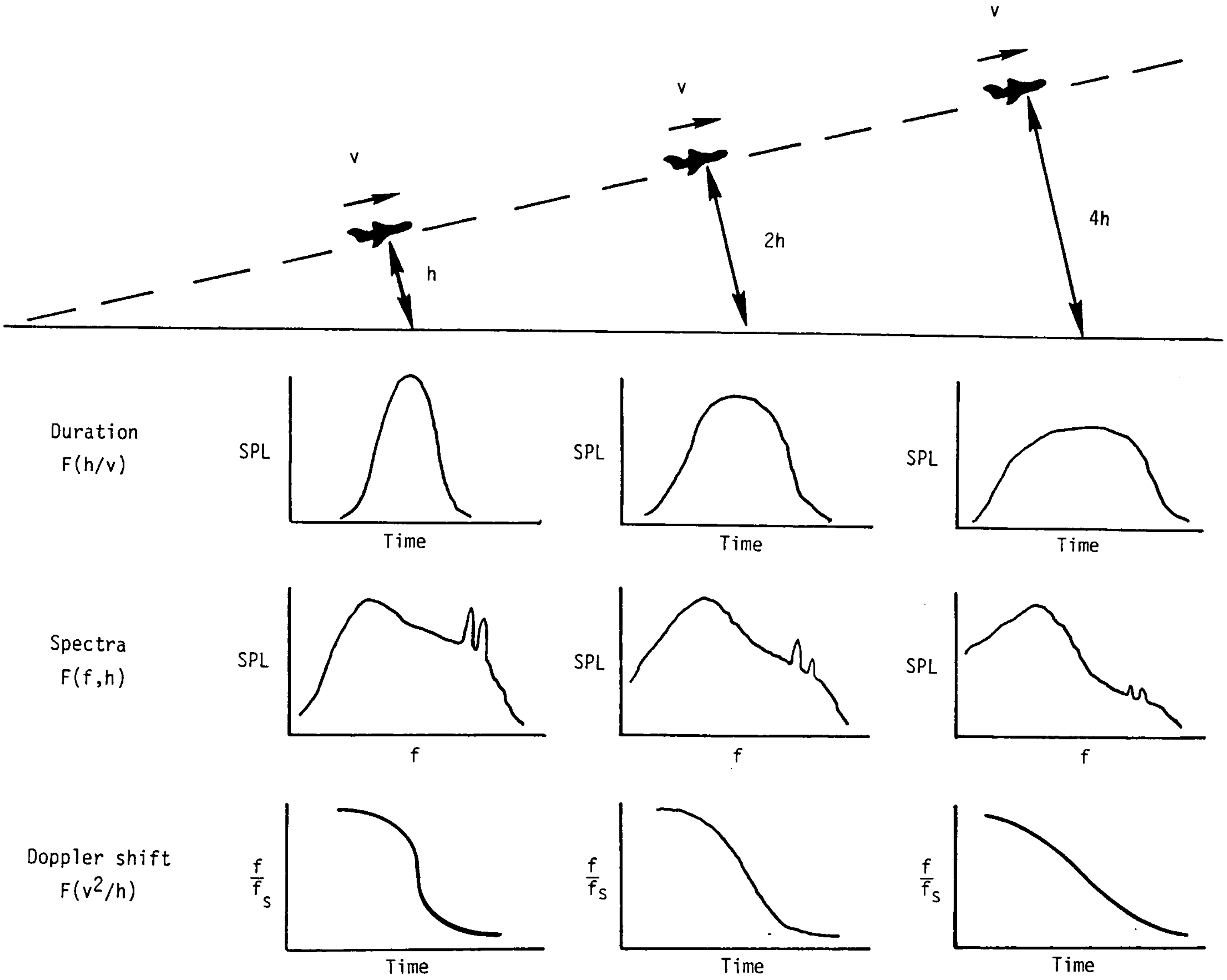


Figure 1.- Confounding of parameters associated with recordings of real aircraft noise.

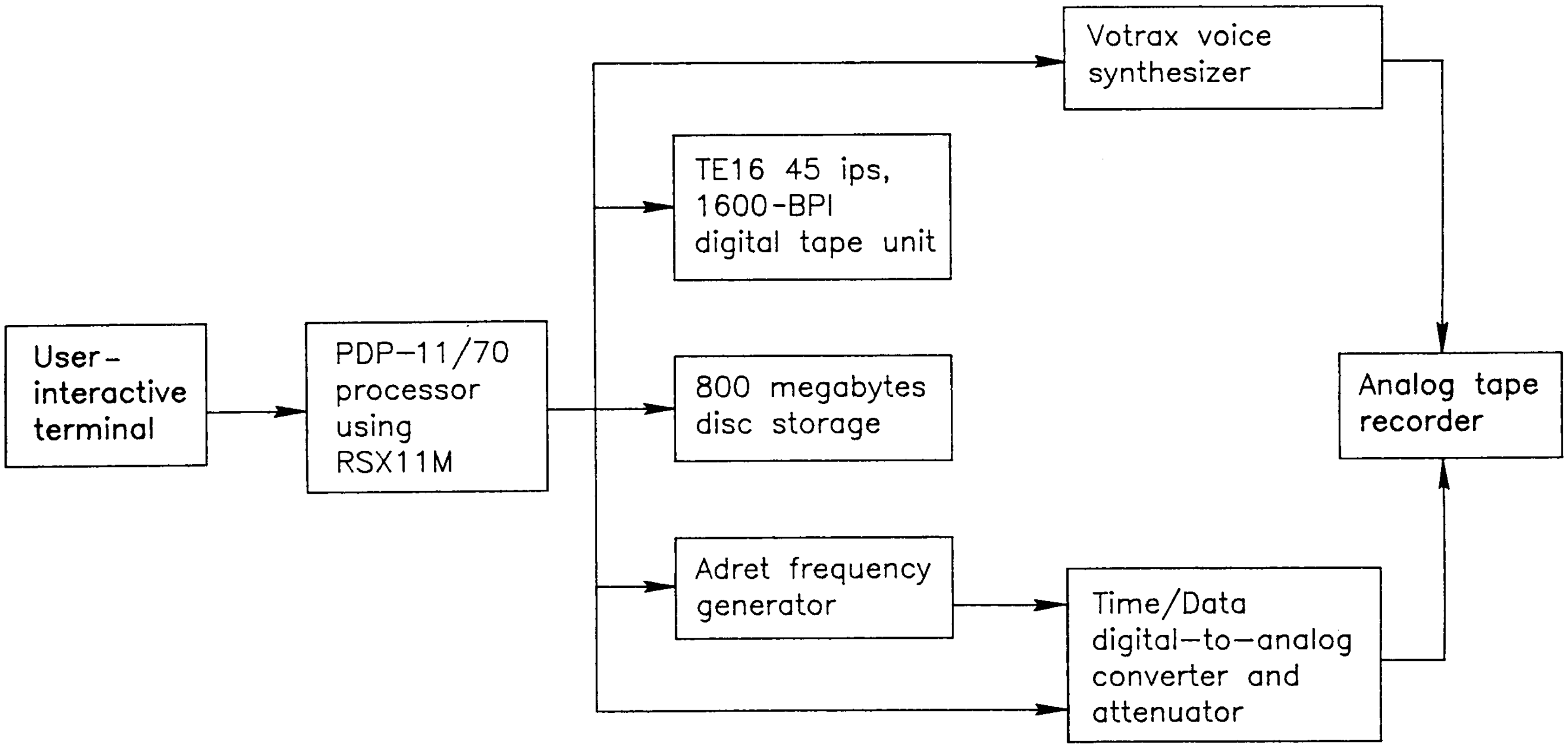


Figure 2.- Aircraft Noise Synthesis System hardware configuration.

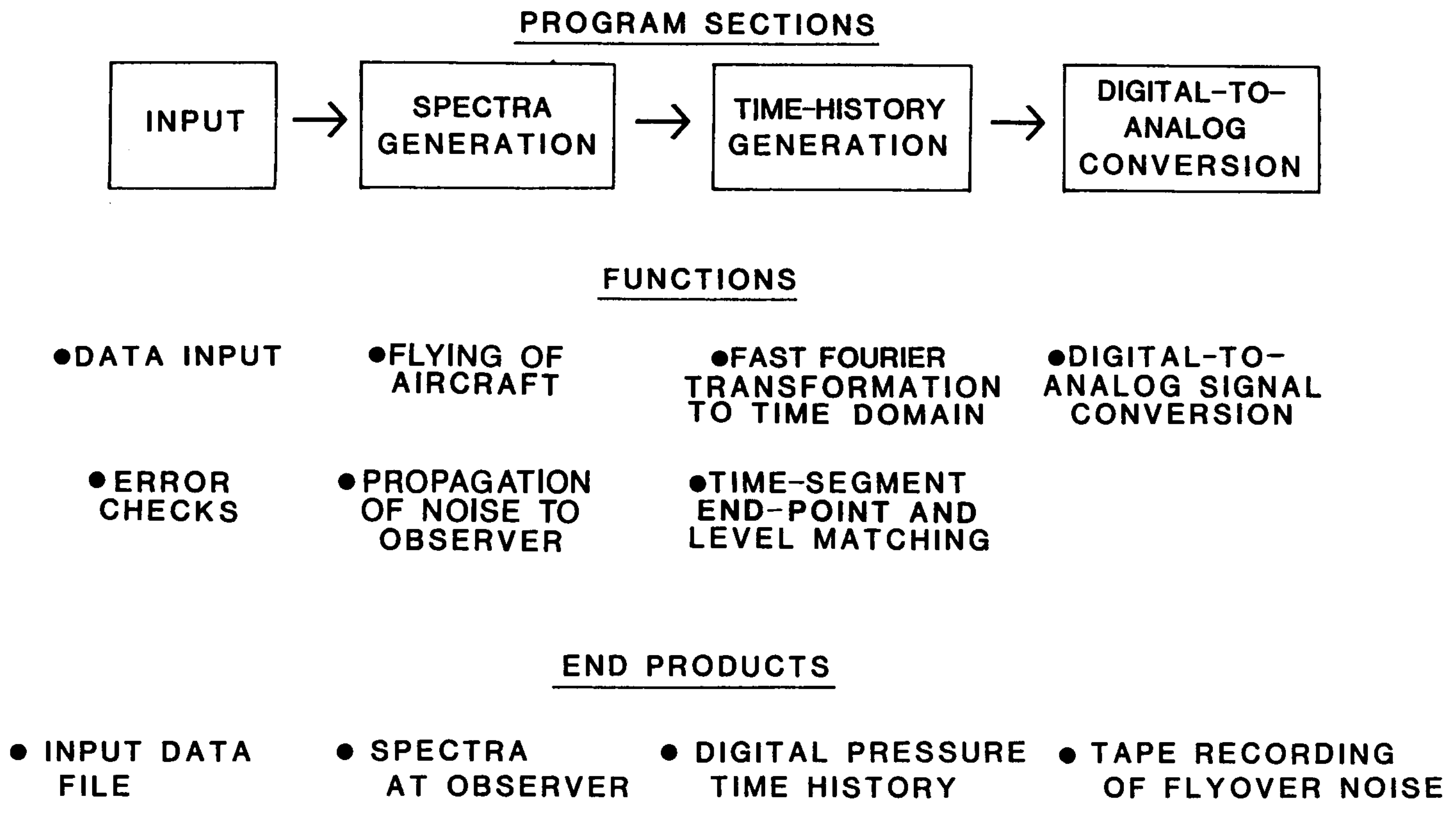

PROGRAM SECTIONS
INPUT
SPECTRA GENERATION
TIME-HISTORY GENERATION
DIGITAL-TO-ANALOG CONVERSION
FUNCTIONS
●DATA INPUT
●ERROR CHECKS
●FLYING OF AIRCRAFT
●PROPAGATION OF NOISE TO OBSERVER
●FAST FOURIER TRANSFORMATION TO TIME DOMAIN
●TIME-SEGMENT END-POINT AND LEVEL MATCHING
●DIGITAL-TO-ANALOG SIGNAL CONVERSION
END PRODUCTS
● INPUT DATA FILE
● SPECTRA AT OBSERVER
● DIGITAL PRESSURE TIME HISTORY
● TAPE RECORDING OF FLYOVER NOISE

Figure 3.- Organization of Aircraft Noise Synthesis System.

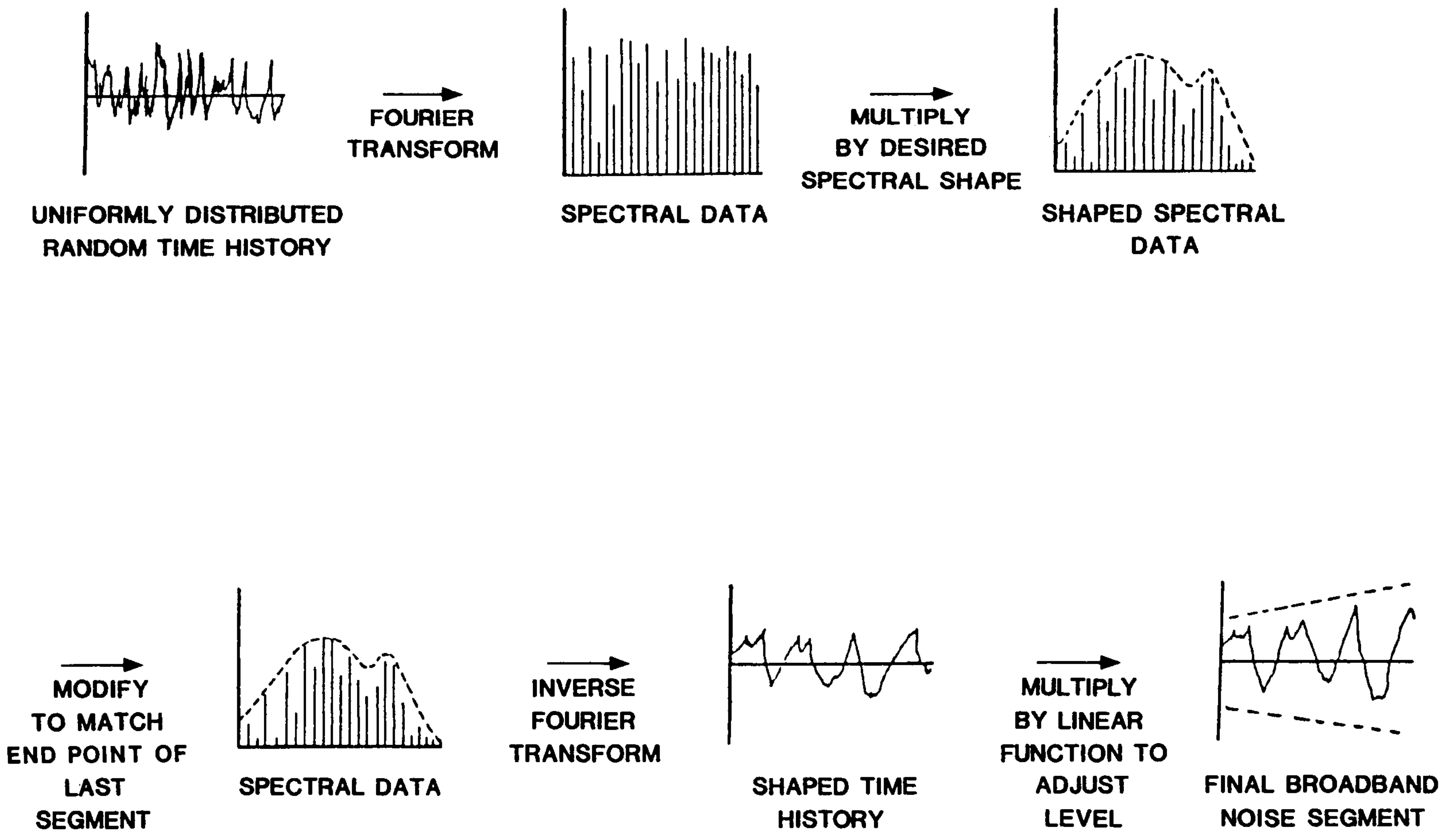


Figure 4.- Generation of random-noise time history.

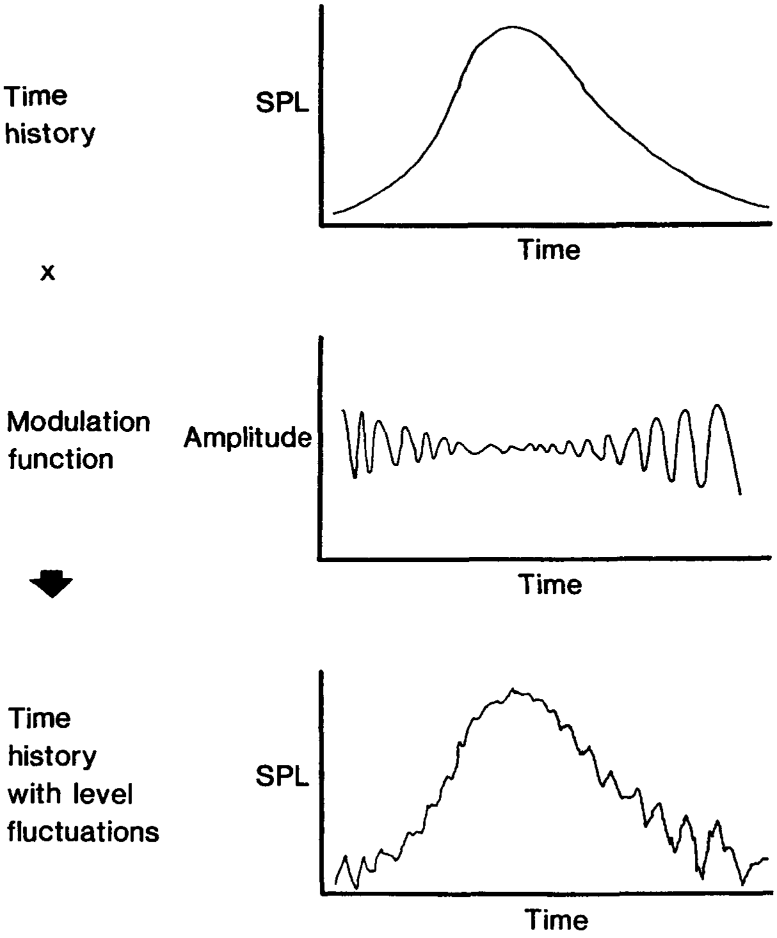


Figure 5.- Generation of level fluctuations.

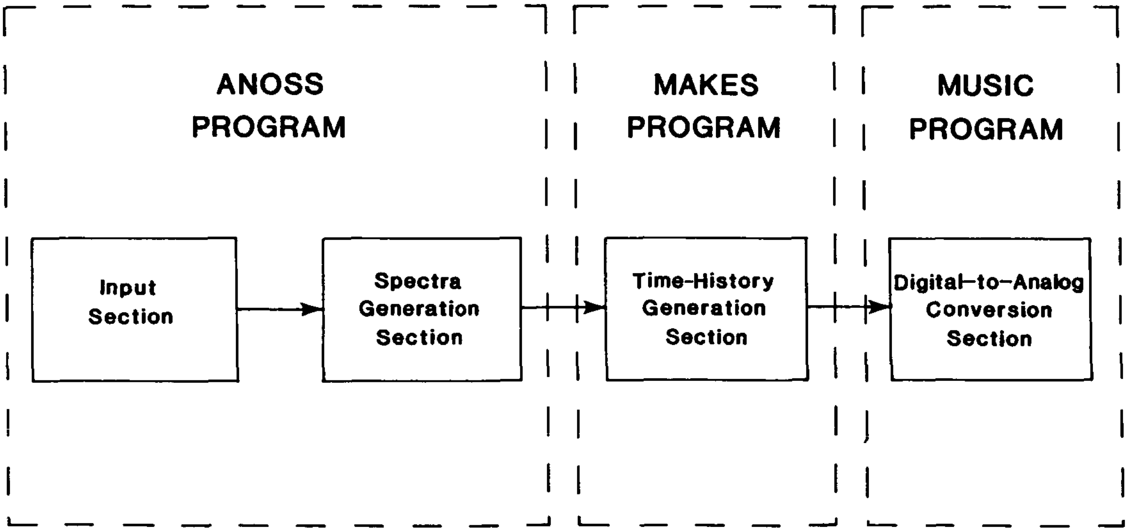

ANOSS PROGRAM
MAKES PROGRAM
MUSIC PROGRAM
Input Section
Spectra Generation Section
Time-History Generation Section
Digital-to-Analog Conversion Section

Figure 6.- Organization of Aircraft Noise Synthesis System software.

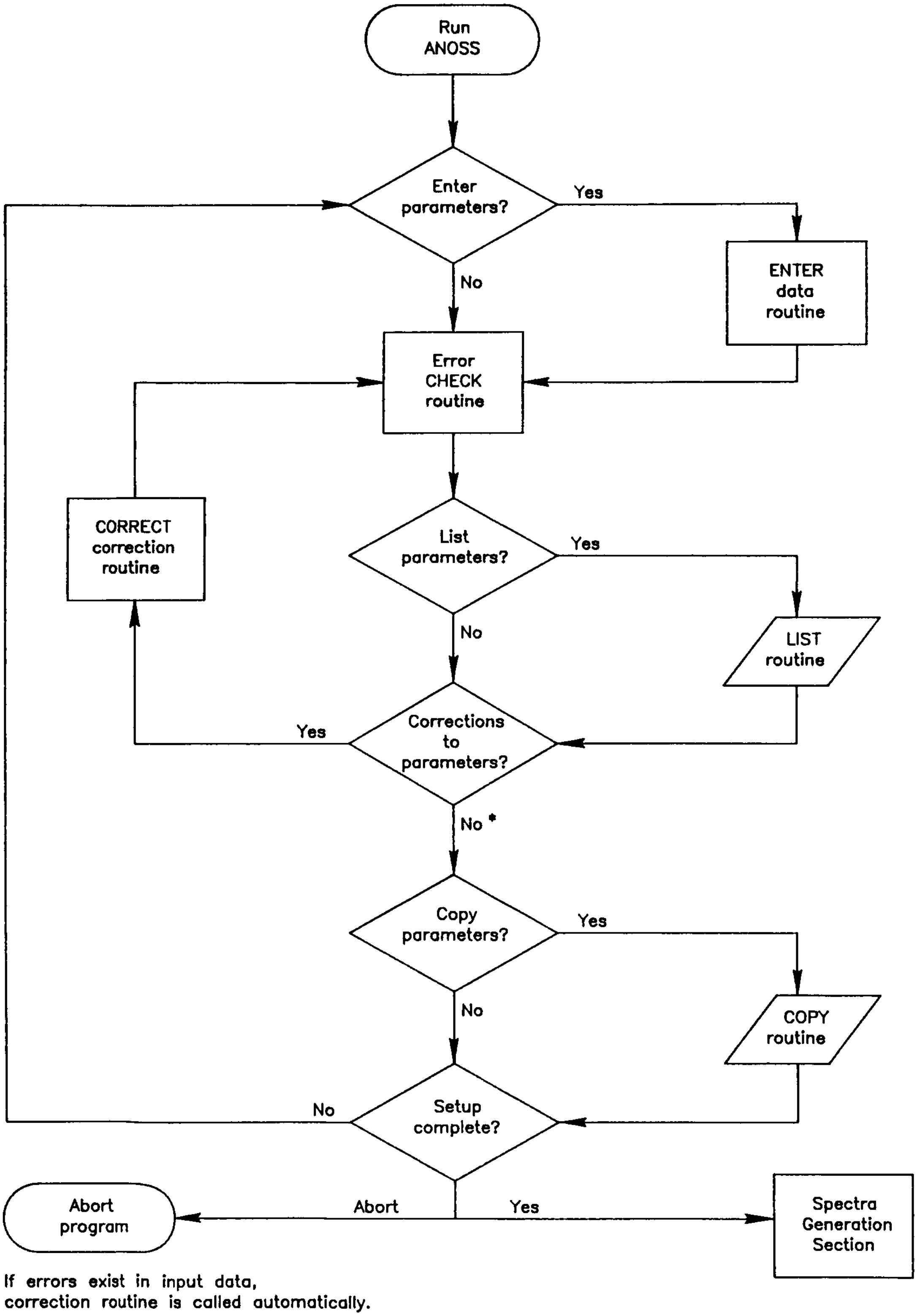


Figure 7.- Aircraft Noise Synthesis System input section flow chart.

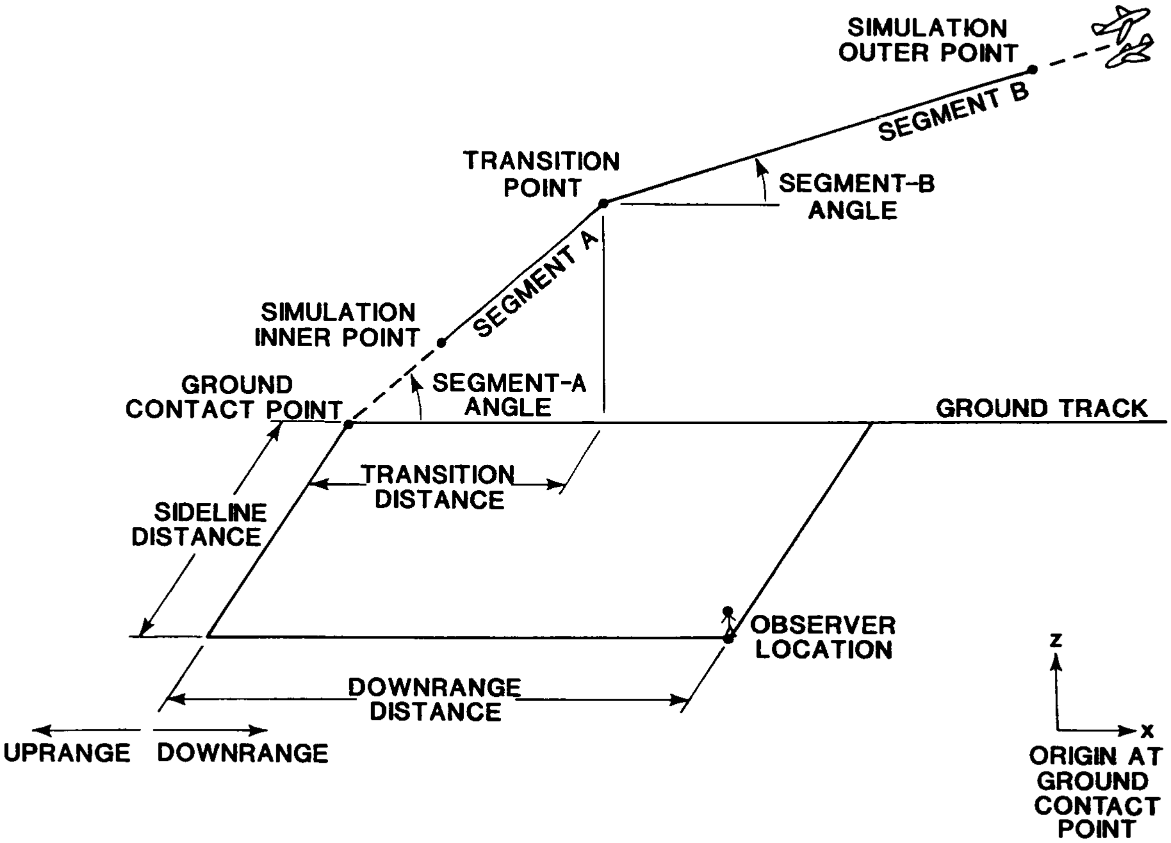


Figure 8.- Observer location relative to aircraft flight path.

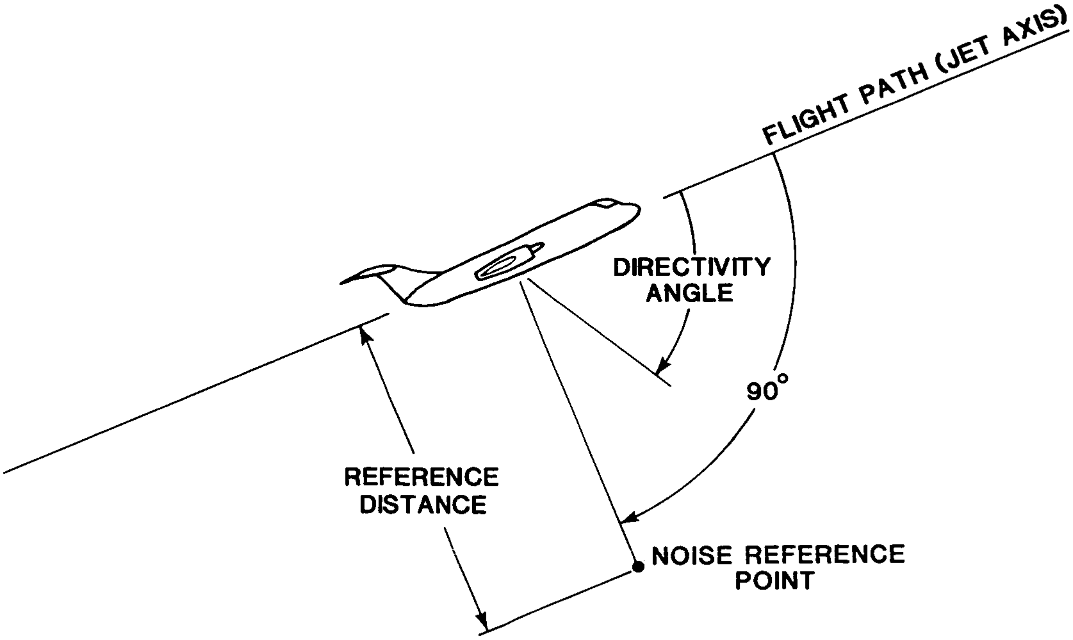
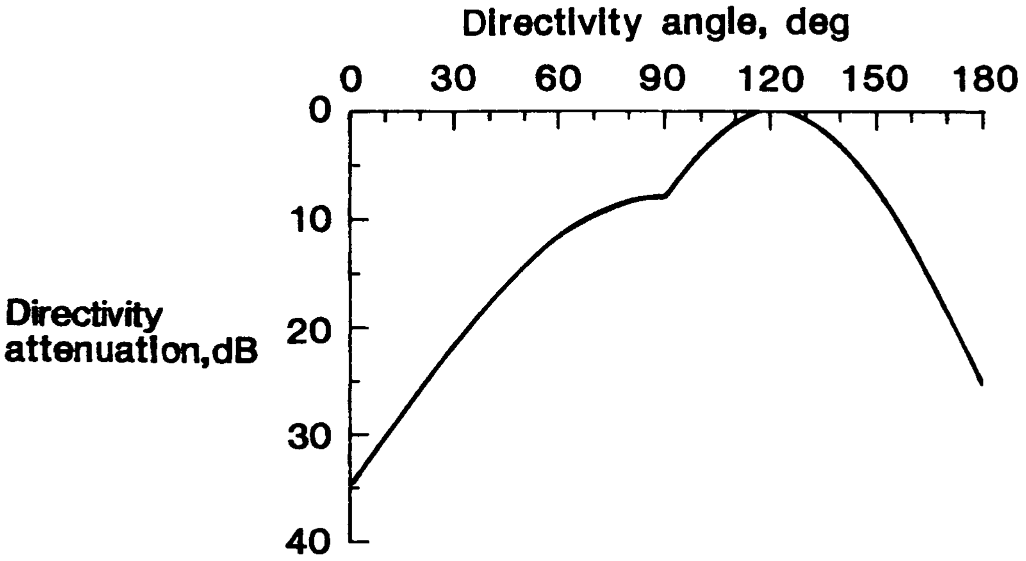
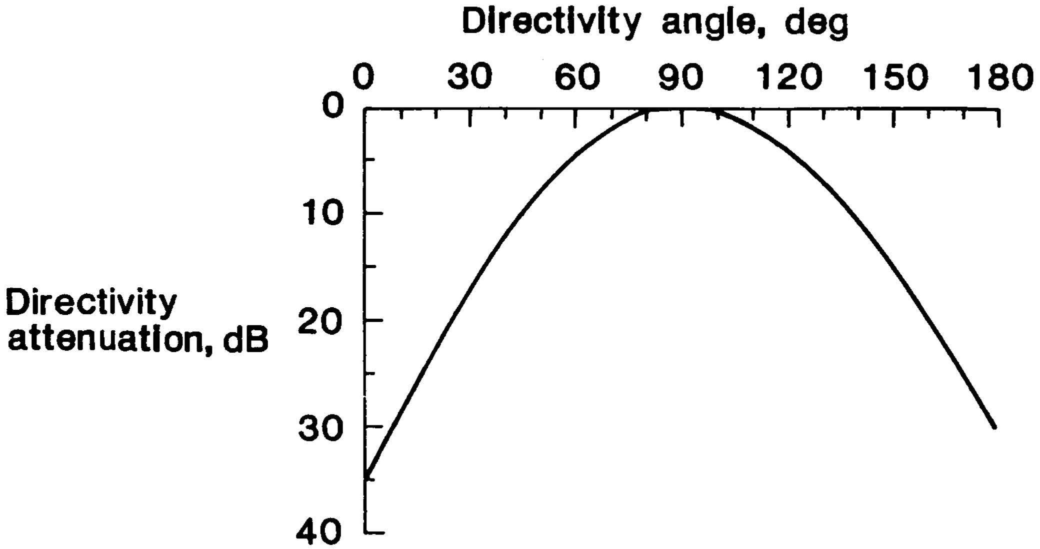


Figure 9.- Aircraft noise description reference frame.



(a) Directivity pattern for $\alpha_{peak} = 120°$.
$A_0 = 35$ dB; $A_{90} = 10$ dB; $A_{180} = 25$ dB.



(b) Directivity pattern for $\alpha_{peak} = 90°$.
$A_0 = 35$ dB; $A_{90} = 0$ dB; $A_{180} = 30$ dB.

Figure 10.- Examples of directivity patterns.

Standard Bibliographic Page

1. Report No.
NASA TM-89040

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
Aircraft Noise Synthesis System

5. Report Date
February 1987

6. Performing Organization Code
535-03-11-03

7. Author(s)
David A. McCurdy and Robert E. Grandle

8. Performing Organization Report No.
L-16140

9. Performing Organization Name and Address
NASA Langley Research Center
Hampton, VA 23665-5225

10. Work Unit No.

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, DC 20546-0001

13. Type of Report and Period Covered
Technical Memorandum

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A second-generation Aircraft Noise Synthesis System has been developed to provide test stimuli for studies of community annoyance to aircraft flyover noise. The computer-based system generates realistic, time-varying, audio simulations of aircraft flyover noise at a specified observer location on the ground. The synthesis takes into account the time-varying aircraft position relative to the observer; specified reference spectra consisting of broadband, narrowband, and pure-tone components; directivity patterns; Doppler shift; atmospheric effects; and ground effects. These parameters can be specified and controlled in such a way as to generate stimuli in which certain noise characteristics, such as duration or tonal content, are independently varied, while the remaining characteristics, such as broadband content, are held constant. The system can also generate simulations of the predicted noise characteristics of future aircraft. A description of the synthesis system and a discussion of the algorithms and methods used to generate the simulations are provided. An appendix describing the input data and providing user instructions is also included.

17. Key Words (Suggested by Authors(s))
Aircraft noise
Subjective acoustics
Psychoacoustics
Synthesizer
Noise simulation

18. Distribution Statement
Unclassified - Unlimited

Subject Category 71

19. Security Classif.(of this report)
Unclassified

20. Security Classif.(of this page)
Unclassified

21. No. of Pages
42

22. Price
A03

For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1987