NASA Technical Memorandum 81306

A Time Lag Study of the Vertical Motion Simulator Computer System

William B. Cleveland

August 1981

National Aeronautics and
Space Administration

NASA Technical Memorandum 81306

A Time Lag Study of the Vertical Motion Simulator Computer System

William B. Cleveland, Ames Research Center, Moffett Field, California

NASA
National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 9403

TABLE OF CONTENTS

SYMBOLS

f	frequency, Hz
j	imaginary operator, $\sqrt{-1}$
k	ratio of sample period to frequency analyzer step period
s	Laplace variable
T	computer step size and sample period, sec
T_{IO}	time lag of input/output operation, sec
X(s)	Laplace transform of continuous variable
X*(s)	Laplace transform of sampled variable
T	time constant of first-order system, sec^{-1}
ϕ	phase, radian
ϕ_e	difference between predicted and observed phase, deg
ω	frequency = $2\pi f$, rad/sec

Subscripts

FA	frequency analyzer
F1	low pass filter
F2	pure time lag "filter"
DAC	digital-to-analog converter
Pre	analog prefilter of ADC

A TIME LAG STUDY OF THE VERTICAL MOTION SIMULATOR COMPUTER SYSTEM

William B. Cleveland

Ames Research Center

SUMMARY

A study was performed to evaluate an experimental method to determine time lags in real-time computer systems as the one associated with the Vertical Motion Simulator at Ames Research Center. The approach was to use an ordinary frequency analyzer to measure the phase difference between inputs and outputs of the computer system. The various elements of the program and computational architecture were modeled. Various factors, such as computer frame time and input frequency, were varied so that they were representative of the operational use of the simulator facilities. Experimentally determined results were compared with predictions derived from the simulation models. The results indicate that the frequency analyzer can be readily used to evaluate time lags in systems of this type. Differences between predicted and measured phase values indicate that the hardware and software imparts a time lag of about 5 msec to this facility.

INTRODUCTION

The effect of the computer system on the results of simulations has many facets and has been the subject of many investigations. The results can be affected by the quality of the modeling of the system, by the method of discrete implementation of continuous models, by the quality of the programming technique, and by the hardware configuration of the computer system together with its input/output (IO) system. This investigation addresses itself to an analysis of the computer and IO system associated with the Vertical Motion Simulator at Ames Research Center. This simulator, because of its large vertical travel of ±10 m, is attractive to researchers and program managers interested in the performance of helicopters and modern vertical and short takeoff and landing (V/STOL) aircraft. Simulation of these vehicles is more difficult than that of other aircraft, especially transport aircraft, because of the size of the aircraft model and, more importantly, the higher natural frequencies of both the aircraft and their associated control systems. The combination of size and frequency requires that a large amount of computation be performed within a short time period. Therefore, it becomes important to rid the simulation of unnecessary time lags since time lags have a destabilizing effect on simulations of aircraft often already burdened with stability problems.

The VMS computer system consists of two computers and an IO system. Because of the number and location of the flight simulators (and computer facilities) at Ames, a policy of associating a small computer with the simulator is evolving. The "host" or large computer that solves the aircraft's

(a) What is the minimum turnaround time in the system? In other words, if the ADC samples a signal that is then used in the digital computer somehow and then is output via a DAC, what is the minimum time to perform this?

(b) Are there any unknown time lags in the system? When one models the system and the contents of digital simulation, one has the means to predict the time lags. Do theoretical and actual results agree?

(c) Are there simple means to measure the time lags in the digital simulation? The answer to this question, in general, is "no" if one is interested in measuring the time lags of the blocks such as those in figure 1. These timings are merely incidental information for the systems analysis of interest anyway. However, one can use a frequency analyzer to measure the total phase difference between the ADC's input and the DAC's output. This phase can be interpreted as total system time lag.

These questions form the background for the general objective of the investigation which was to determine if a frequency analyzer can be used experimentally to provide verification or validation for a simulation computer configuration.

Approach

The method of satisfying the objectives consisted of comparing experimental and theoretical results. Figure 2 illustrates the major elements for which linear models were derived. In the case of the "computer program," simple linear systems were selected for ease of operation as well as for clarity of interpreting results; in particular, a one-T delay was employed as one of the cases.

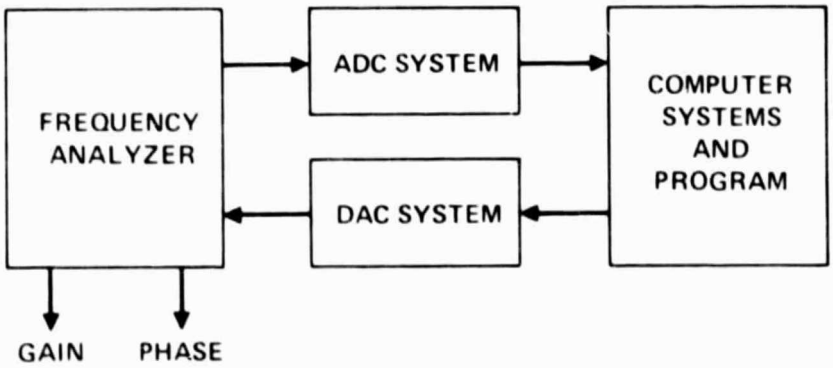


Figure 2.- Frequency analyzer technique for determination of system time lags.

DESCRIPTION OF FACILITY

The facility of interest is the computer and IO associated with the Vertical Motion Simulator (VMS) at Ames. The capabilities of the simulator are described in reference 2. Within this section, the computer and IO hardware and software are described to the extent that the models needed for the analysis may be derived.

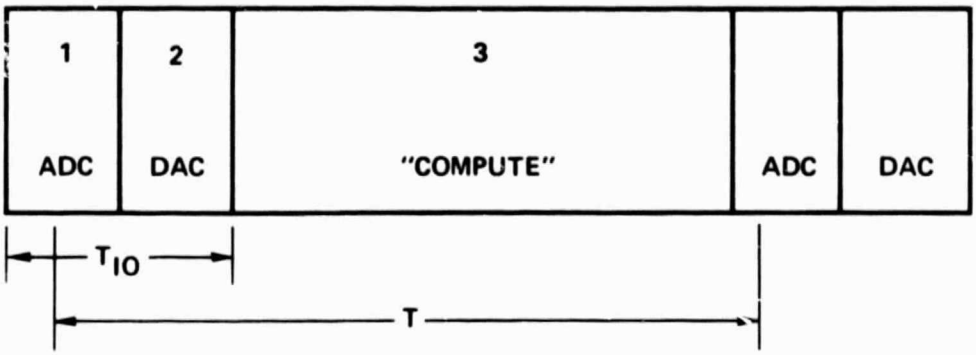
Computer and IO Operation

The relevant aspects of the computers and IO systems are the method of timing and control of the various processes. The ADCs of the IO section run asynchronously while the two computers operate synchronously. While the details of the IO are given in reference 3, it is useful to outline its operation here. The IO system consists of individual circuit boards for the ADC function (as well as for the DACs and discrete functions). Each board has a microcomputer to control 16 channels of ADC. The controller and ADCs are all self-contained on the one board. If more than 16 ADCs are needed for the simulation, they are added board by board to the IO system in increments of 16 ADCs.

The microcomputer runs an ADC and a 16-channel multiplexer to effectively obtain 16 channels of ADC. The operation consists of the microcomputer running the function at 50 μsec per ADC conversion or 0.8 msec for the 16 ADC channels. The results of the conversion are stored in a buffer on the circuit card. The buffer is updated or has fresh information every 0.8 msec. Since each ADC board operates independently, each set of 16 ADCs is refreshed at this rate also. Consequently, no matter how many blocks of ADCs are used, there is no more than 0.8 msec of time skew between any two channels.

Timing of the overall computer system is controlled by a clock associated with the PDP computer. Figure 3 illustrates the process in which the ADCs, DACs, and the host computer are coordinated by the PDP. Several important points are to be considered. The first is the order of the ADC-DAC-compute schedule. The implication is that, in a given period, the DAC cannot output data that are a function of the ADC input from that same period. An alternate schedule sometimes used in process control applications is the ADC-compute-DAC-compute schedule. The control laws that contain lead are calculated in the first computer period and are output via the DACs as soon as possible to minimize phase lag. Any remaining controls are calculated in the second computer period for an operation much the same as the ADC-DAC-compute schedule. This latter sequence of IO and compute is not used in the simulation system at Ames.

Another observation is that the individual timing associated with sending data from the ADC to the PDP, from the PDP to the host, etc., is not important from a systems point of view. So long as the input is sampled by the ADC at the first clock pulse and the output is converted by the DAC at the subsequent clock pulse, what went on in the computer system to make it all happen is not



1. ADC VALUES ARE OBTAINED WITHIN 0.8 msec IN ADVANCE OF THE CLOCK PULSE. THE LOCAL COMPUTER OBTAINS THE ADC DATA FROM THE I/O SYSTEM AND SENDS IT TO THE HOST COMPUTER AS INPUTS TO THE SIMULATION EQUATIONS.

2. THE LOCAL COMPUTER LOADS AND INITIATES THE DAC CONVERSION IN THE I/O SYSTEM.

3. THE HOST COMPUTER SOLVES THE EQUATIONS THEN OUTPUTS TO THE LOCAL COMPUTER WHICH READIES THE DAC DATA FOR THE NEXT CYCLE.

Figure 3.- Timing and functional description of control cycle.

important. Naturally, minimizing the "overhead" time required by all the systems to operate is important so that the simulation computing time will be sufficient; it is advantageous to minimize T to maximize accuracy of the simulation.

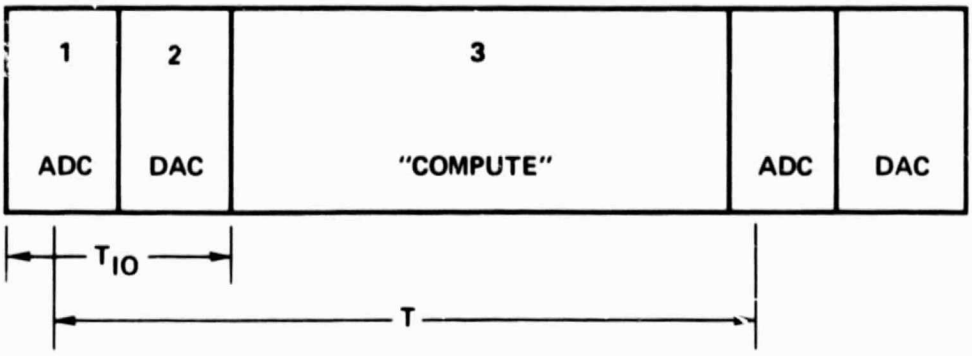
System Models

The system may be broken into the four components shown in figure 2, that is, the ADCs, the computer, the DACs, and the frequency analyzer. The model of each component contributes to the understanding of the system.

ADC model- The ADC model has two parts of interest: the ADC prefilter and the ADC itself. The prefilter was included in the VMS IO system to filter out 60-Hz noise; its corner frequency is set to 20 Hz for this purpose. The filter may be switched in or out of the system as desired. The phase of the filter for input frequency ω below the corner frequency ω_o is

$$\phi_{pre} = -\omega/40\pi \ , \qquad \omega < \omega_o$$

From this, the equivalent pure time lag may be calculated as



1. ADC VALUES ARE OBTAINED WITHIN 0.8 msec IN ADVANCE OF THE CLOCK PULSE. THE LOCAL COMPUTER OBTAINS THE ADC DATA FROM THE I/O SYSTEM AND SENDS IT TO THE HOST COMPUTER AS INPUTS TO THE SIMULATION EQUATIONS.

2. THE LOCAL COMPUTER LOADS AND INITIATES THE DAC CONVERSION IN THE I/O SYSTEM.

3. THE HOST COMPUTER SOLVES THE EQUATIONS THEN OUTPUTS TO THE LOCAL COMPUTER WHICH READIES THE DAC DATA FOR THE NEXT CYCLE.

Figure 3.- Timing and functional description of control cycle.

important. Naturally, minimizing the "overhead" time required by all the systems to operate is important so that the simulation computing time will be sufficient; it is advantageous to minimize T to maximize accuracy of the simulation.

System Models

The system may be broken into the four components shown in figure 2, that is, the ADCs, the computer, the DACs, and the frequency analyzer. The model of each component contributes to the understanding of the system.

ADC model- The ADC model has two parts of interest: the ADC prefilter and the ADC itself. The prefilter was included in the VMS IO system to filter out 60-Hz noise; its corner frequency is set to 20 Hz for this purpose. The filter may be switched in or out of the system as desired. The phase of the filter for input frequency ω below the corner frequency ω_o is

$$\phi_{pre} = -\omega/40\pi \ , \qquad \omega < \omega_o$$

From this, the equivalent pure time lag may be calculated as

$$T_{pre} = -\phi_{pre}/\omega = 1/40\pi = 0.008 \text{ sec}$$

Thus filtering the high frequencies on the input results in a pure time lag of 8 msec.

The ADC converter model shown in figure 4 is made up of time delay plus a sampler operating at the rate of one sample for every period of T seconds. The sampler is the standard representation of an ADC, but the addition of the time lag of T_{IO} seconds is not. The lag is introduced for two reasons. One reason is that the 0.8-msec period for sampling 16 ADCs is extended by the actual data transfer process from the ADC microcomputer to the PDP computer. If the sampling rate T is 40 msec, 50 conversions of the 16 ADCs have occurred every time the PDP computer "samples" the ADC unit. Since the PDP may sample at any point in the ADC's operation, it is possible that the converted ADC data in the microcomputer may be as much as 0.8 msec old, exactly in synchronization, or somewhere in between. An average time lag of 0.4 msec in the ADC is a reasonable assumption.

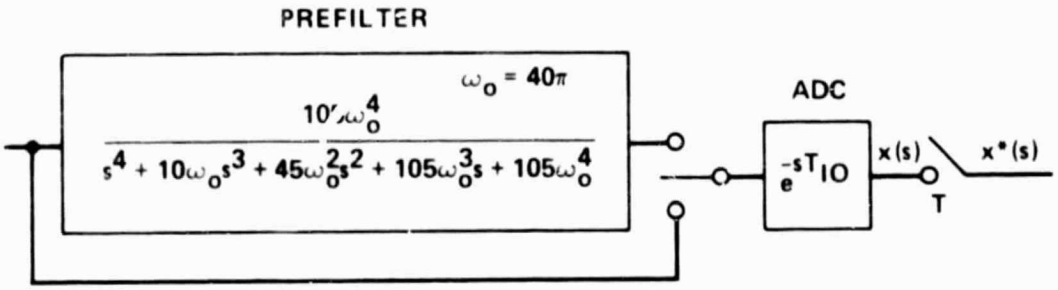


Figure 4.- ADC system model containing a fourth-order Bessel filter and conversion model.

The second portion of the IO time lag is due to the time it takes the PDP to receive the ADC data from the RIOU and send it on to the Sigma computer in addition to the output data transfer from the PDP to the RIOU. This time has been estimated to be on the order of 3 to 4 msec but no timing has been performed. Since timing data are not available, a pure time delay of T_{IO} is assumed to be present with its magnitude to be determined from the differences between experimental and theoretical timing.

The model of the sampler itself may be obtained from reference 4:

$$X^*(s) = \frac{1}{T} \sum_{n=\infty}^{\infty} X(s - jn2\pi/T)$$

In this experiment, the repeating spectra generated by the sampler have no effect on the experimental results. Although they are the source of the "staircase" effect in a DAC's output, it is shown in the appendix that, given a frequency analyzer that operates on the basis of forming Fourier series coefficients, the repeating spectra have no effect on the coefficients when

the ratio of the sampling frequency to the input frequency being analyzed divides evenly into 200. This condition was observed in the experimental setup. Thus the ADC model may be simplified to

$$X^*(s) = \frac{1}{T} X(s)$$

Computer model- From the considerations discussed in the section on computer and IO operation, no model is required of the computer hardware itself. What is needed is the model of the computations performed. Two different dynamical models were simulated for the test — one a first-order system of unity gain and time constant T, and the second, a pure time lag of T seconds.

The first-order system was mechanized using the zero-order hold method resulting in the theoretical phase:

$$\phi_{F1} = \tan^{-1} \frac{-\sin \omega T}{\cos \omega T - \exp(T/T)}$$

The pure lag system produces the phase:

$$\phi_{F2} = -\omega T$$

Both filters were run under the Ames software (described in refs. 4 and 5), in which portions of the calculations may run at speeds that are fractions of the basic or nominal speed. For this test, both filters were simulated using periods T and 2T. The effect of operating the software at the slower speed, equivalent to a period of 2T, is to place a sampler in the "slow" path which, in effect, performs the function of an ADC running at period 2T. The computations produce a new result every 2T seconds which effectively makes the DACs operate at a period of 2T also.

When the experimental tests are conducted, representations of both filters are operating at both fast and slow speeds; thus, there are four computer model cases.

DAC model- The digital-to-analog converters are conventional zero-order hold devices. Like the ADCs, the DACs are packed 16 to a circuit board. Figure 3 illustrates the timing in which the PDP computer loads buffers of the DAC circuit and initiates the actual conversion. Since the IO lag was discussed in the ADC section, the DAC operation may be modeled in the conventional manner. The phase relationship for the zero-order hold DAC (taken from ref. 6) is

$$\phi_{DAC} = -\omega T/2$$

This phase is interesting since it is half the phase of a pure time delay of T seconds, giving rise to a rule of thumb that a DAC is equivalent to a T/2 delay. This is always true for the phase characteristics. The gain characteristics, however, depend on input frequency, but this is an appropriate approximation for small T.

Frequency analyzer- The frequency analyzer used was a Schlumberger model 1170. Its sine wave frequency generator provided the input to the ADC. In addition, the analyzer used this input as well as the DAC's output to determine the gain and phase differences between the two. The appendix gives some insight into how the analyzer works in theory, but the important thing from the system test point of view is the method by which the input sine wave is represented and its interaction with the ADC sampling. Basically, the Schlumberger unit generates a sine wave in a discrete formulation. Two hundred discrete levels are used for every cycle. Thus the frequency generator effectively operates as though a continuous sine wave had been sampled, then as output through a zero-order hold device.

The ADC sampling the "discrete" sine wave is not synchronized in any way with the sampling rate of the frequency analyzer. This brings up two important considerations: the ratio of the sample-hold frequency of the analyzer to that of the ADC and the observation that the ADC is, in effect, sampling a step input at each sample point.

It was originally thought that the frequency ratio should be an integer such as 10 or 20 merely for convenience and to ensure that the sampling frequency did not violate the Nyquist criterion of being less than twice that of the input. However, the appendix shows that any spectrum generated by the ADC has no effect on the frequency analyzer operation when the frequency ratio divides evenly into 200. This leads to an operational requirement to change the input frequency when the sample rate of the computing system is varied.

We let $k = T/T_{FA}$ and note that $T_{FA} = 1/(200f)$, then $f = k/(200T)$. This relationship specifies the input test frequency when k and T have been specified. The parameter k has the requirements that it divide integrally into 200 and that it be smaller than 100 to avoid violation of the Nyquist criterion.

The fact that the sine wave is composed of 200 steps leads to a statistical phase model of the interaction of the ADC and frequency generator. Figure 5 illustrates a typical ADC sample of the input at some random point in the hold period, T_{FA}, of the wave. Since the ADC sample may occur at any time in the period, a uniform distribution of pure time delays between 0 and T_{FA} is expected. The mean delay is $T_{FA}/2$. In terms of phase, the mean is

$$\bar{\phi}_{FA} = \pi/200 \text{ radians}$$

$$= 0.9°$$

EXPERIMENTAL MATRIX

Using the elements of the facility and program as described here, the experimentation was designed to be run in a manner akin to normal operational practices, that is, the values of T were representative, the software was that which is normally used, etc. The factors of the study were the step

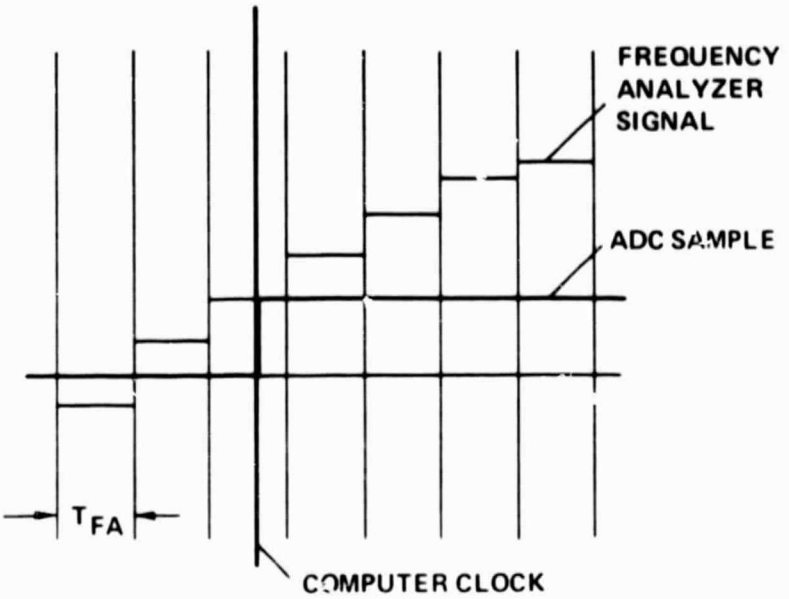


Figure 5.- Time delay due to frequency analyzer and ADC interaction.

size T, input frequency f, the particular digital filter (pure delay or first-order lag), and the presence or absence of the ADC prefilter. Table 1 illustrates these factors.

TABLE 1.- EXPERIMENTAL FACTOPS

Step size, T	40, 50 msec
Frequency, f	1, 1.25, 2, 2.5 Hz
Filter type	e^{-sT}, 1/(TS + 1)
ADC prefilter	In, Out

The step size T was specified to be 40 and 50 msec in the faster calculation loop (80 and 100 msec in the slower loop). By letting k = 10 and 20, the frequency test values were obtained. One might argue that k and not f is the factor; however, if the frequency had been selected first, the values of k would follow. It appears that either will suffice as the variable factor.

Care was taken with the first-order filter to ensure some consistency within the test cases. The corner frequencies of the filters were set to 1/10 the Nyquist frequency or

$$1/T = 2\pi(1/2T)(0.1)$$

so that

$$T = 10T/\pi$$

While the value of T varies with T, it is not a critical value in the sense that results depend on it; therefore, it is not a factor in the experimental sense. The important aspect is that both classical first-order, low-pass filters and pure time delays were used to determine effects arising from time step size and time lags.

RESULTS

Data for the 32 cases resulting from the matrix were obtained and compared with theoretical or predicted results (table 2). In acquiring the phase data, it was immediately obvious that the results were statistical. The frequency analyzer technique yielded consistent run-to-run magnitude results, but the phase data varied considerably. Data for maximum and minimum phase values for each case were collected as a result; both are listed in the table, except four cases for which data were not collected. The predicted values are the sums of the theoretical phases for the individual components in the hardware-software loop. The models for the DAC, digital filters, and prefilter were used with the values of T and f to obtain the prediction values.

Comparing the predicted values with the experimental results shows that the experimental results follow fairly well with the deterministically predicted values. From the models of the subsystems discussed here, only the frequency analyzer-ADC interaction and the possibility of time lags in the ADC-computer interaction are not included in the predicted values. The former is a phase-related item and the latter is a time lag item. This idea leads to a model for the difference or error between observed and calculated. Letting

$$\phi_e = \phi_o + af$$

where ϕ_e is the observed difference and ϕ_o is the phase offset that would represent phase-related effects. The product af is the portion of the error correlated with frequency or relative to a delay time since, for a pure time lag, the phase equals the product of frequency and the time lag. Figure 6 shows phase errors plotted against frequency along with the linear regression fit to each of the two sets of data. Consider

$$\phi_{e_{Max}} = 0.12 - 2.307f$$

The constant indicates a phase offset of only 0.12°; this is not a large value and will be ignored as being within the error of data observation. However, the part correlated with frequency is interpreted to result from a time lag T_{IO} as

TABLE 2.- THEORETICAL AND EXPERIMENTAL PHASE RESULTS

Case	T, msec	f, Hz	G(z) case	Pre-filter	System phase, deg		
					Predicted	Measured	
						Max.	Min.
1	50	1	1	0	-27.0	-29.4	-28.6
2	50	1	2	0	-63.5	-65.7	-65.1
3	100	1	1	0	-54.0	-56.3	-55.6
4	100	1	2	0	-100.4	-.02.8	-102.0
5	50	1	1	I	-29.9	-32.4	-31.5
6	50	1	2	I	-66.4	-68.7	-67.9
7	100	1	1	I	-56.9	-59.2	-58.5
8	100	1	2	I	-103.3	-105.5	-104.8
9	50	2	1	0	-54.0	-58.5	-57.1
10	50	2	2	0	-100.4	-104.8	-103.2
11	100	2	1	0	-108.0	-112.9	-110.8
12	100	2	2	0	-149.9	-154.7	-152.6
13	50	2	1	I	-59.8	-64.4	-62.8
14	50	2	2	I	-106.3	-110.6	-108.9
15	100	2	1	I	-113.8	-118.4	-116.3
16	100	2	2	I	-155.7	-160.6	-158.4
17	40	1.25	1	0	-27.0	-29.8	-29.2
18	40	1.25	2	0	-63.5	-65.8	--
19	80	1.25	1	0	-54.0	-56.8	-56.2
20	80	1.25	2	0	-100.4	-102.7	--
21	40	1.25	1	I	-30.6	-33.5	--
22	40	1.25	2	I	-67.3	-69.6	-69.2
23	80	1.25	1	I	-57.6	-60.0	--
24	80	1.25	2	I	-104.0	-106.6	-106.2
25	40	2.5	1	0	-54.0	-59.5	-58.3
26	40	2.5	2	0	-100.4	-105.7	-104.4
27	80	2.5	1	0	-108.0	-113.8	-112.2
28	80	2.5	2	0	-149.9	-155.4	-154.0
29	40	2.5	1	I	-61.2	-66.9	-65.7
30	40	2.5	2	I	-107.6	-113.1	-112.0
31	80	2.5	1	I	-115.2	-121.3	-119.4
32	80	2.5	2	I	-157.1	-162.5	-161.0

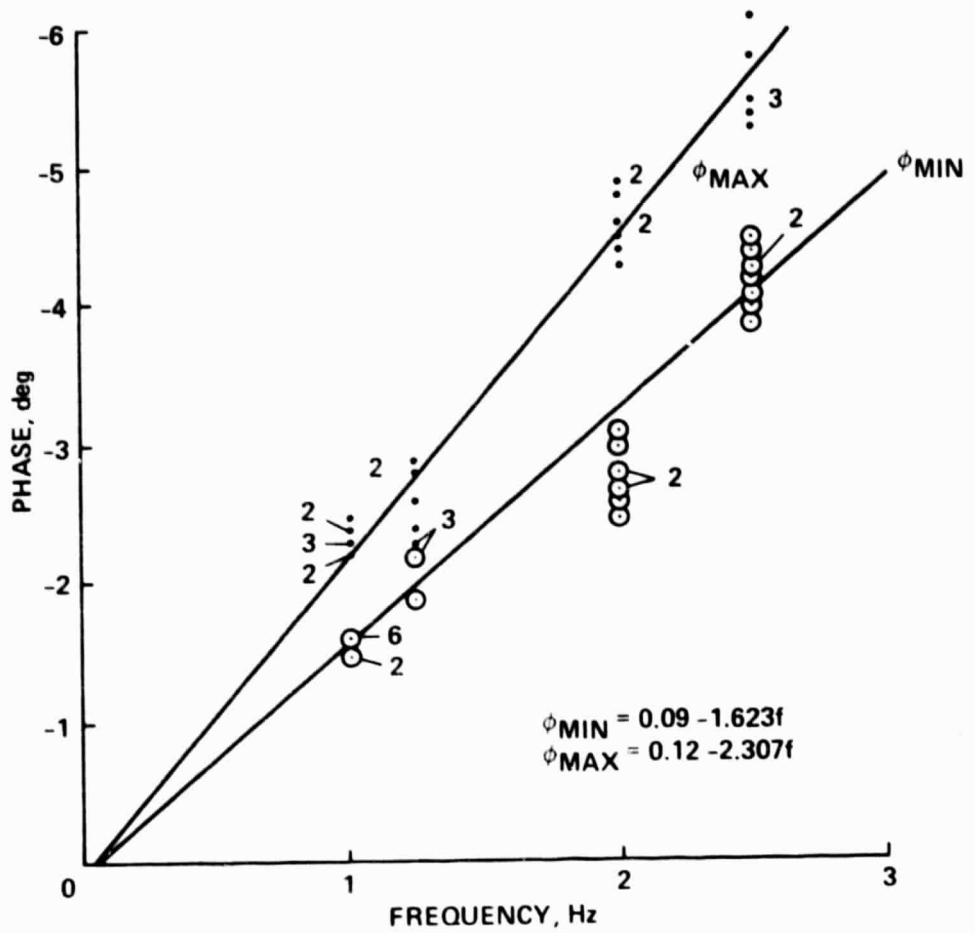


Figure 6.- Linear regression fits to error data.

$$-2.307f = 360°T_{IO}f$$

or

$$T_{IO} = 6.4 \text{ msec}$$

Similarly, for $\phi_{min} = 0.09 - 1.623f$, the lag is obtained as $T_{IO} = 4.5$ msec. Thus a lag of 4 to 6 msec apparently exists within the simulation loop, but no appreciable phase offset is apparent from the frequency analyzer-ADC interaction.

CONCLUSIONS

The goal of this study was to determine whether the VMS simulation computer system could be verified using an experimental approach with a frequency analyzer. By carefully structuring the experiment and modeling the components,

the overall phase of the simulation loop is both predicted and determined experimentally. The results indicate that the technique can be used successfully.

The differences between the predicted and measured values lend themselves to a statistical analysis which indicates that the computer hardware and software configuration imparts a time lag of approximately 5 msec to simulations on the facility. This value is relatively small since naive use of the facility can produce 8 msec by unintentional use of an ADC prefilter or as much as 40 to 50 msec with poor programming techniques. Such errors would be detected easily through the use of this technique.

APPENDIX

EFFECTS OF ADC ON FREQUENCY ANALYZER

The phase and gains produced by the frequency analyzer used in this test were derived on the basis that a sine wave input to a system will yield a sine wave output that may be represented as a sum of sine and cosine waves. Thus,

$$\begin{aligned} v_o &= M \sin(\omega\tau + \phi) \\ &= A \cos \omega\tau + B \sin \omega\tau \end{aligned} \tag{A1}$$

where

$$M = (A^2 + B^2)^{1/2}$$

$$\phi = \arctan A/B$$

The frequency analyzer actually determines A and B first, then calculates M and ϕ. The procedure is directly from Fourier series analysis which yields

$$A = \frac{\omega}{\Pi} \int_o^{2\pi/\omega} v_o \sin \omega\tau \tag{A2}$$

and

$$B = \frac{\omega}{\Pi} \int_o^{2\pi/\omega} v_o \cos \omega\tau \tag{A3}$$

The frequency analyzer does not perform continuous integration but uses a simple quadrature formulation. The period of integration (summations) is one full cycle of the frequency.

The ADC has the effect of creating copies of the signal input to it as described earlier in the text. For a sine wave input, its sampled version is

$$v = \frac{1}{T} \sum_{n=0}^{\infty} \sin\left(\omega + \frac{2\pi n}{T}\right) \tau$$

The DAC effect on the spectra is to attempt to filter out all components except n = 0. However, the output contains traces of the other components as

$$v_o = \sum_{n=0}^{\infty} M_n \sin\left[\left(\omega + \frac{2\pi n}{T}\right) \tau + \phi_n\right]$$

which may be written as

$$v_o = \sum_{n=0}^{\infty} \left[A_n \sin\left(\omega + \frac{2\pi n}{T}\right) + b_n \cos\left(\omega + \frac{2\pi n}{T}\right) \right] \tag{A4}$$

This signal, v_o, is the one applied to the frequency analyzer equations. Before performing the Fourier series integrations, it is useful to consider the particular frequencies used in the experiment. The computer frame time T was always chosen so that its frequency was a multiple of ω. It was formulated as $T = k\pi/100\omega$. Thus the spectral locations are determined as

$$\omega + \frac{2\pi n}{T} = \omega\left(1 + \frac{200n}{k}\right) \tag{A5}$$

The parameter k represents the number of computer sampling periods occurring for each period of the input frequency ω. Values of either 10 or 20 used in this test make the frequencies of the spectra odd multiples of ω so that

$$\omega\left(1 + \frac{200n}{k}\right) = i_n\omega \tag{A6}$$

where the i_n are all odd integers. When v_o from equation (A4) is substituted into (A2), the fact that the integral of odd functions over the interval is zero leaves only

$$A = \frac{\omega}{\pi} \int_0^{2\pi/\omega} \sin \omega\tau \sum_{n=0}^{\infty} a_i \sin i_n\omega\tau \, dt$$

Using the fact that i_n is odd leaves

$$A = \frac{\omega}{\pi} \int_0^{2\pi/\omega} a_o \sin^2\omega\tau = a_o$$

Similarly, $B = b_o$.

Thus the frequency analyzer theoretically finds the amplitudes of the sine and cosine of the primary frequency only. None of the repeated spectra are used to compromise the calculation of the frequency response of the system analyzed at frequency ω. It must be stressed that this is not a general result, but is true when the particular input sine wave frequencies and computer frame times are selected carefully.

REFERENCES

1. Baron, Sheldon; Lancraft, Rog; and Zacharias, Greg: Pilot/Vehicle Model Analysis of Visual and Motion Cue Requirements in Flight Simulation. NASA CR-3312, 1980.

2. Jones, A. David: Operations Manual: Vertical Motion Simulator (VMS) S.08. NASA TM-81180, 1980.

3. Fetter, John L.: Real-Time Computation System. Paper to be presented at Summer Computer Simulation Conference (Washington, D.C.), July 1981.

4. Sinacori, John B.; and Stapleford, Robert L.: Researcher's Guide to the NASA Ames Flight Simulator for Advanced Aircraft (FSAA). NASA CR-2875, 1977.

5. Cleveland, William B.: The Handling Qualities Program for the Augmentor Wing Jet STOL Research Aircraft. AFIPS Fall Joint Computer Conference Proceedings, 1971.

6. Cleveland, William B.: First-Order-Hold Interpolation Digital-to-Analog Converter with Application to Aircraft Simulation. NASA TN D-8331, 1976.