# Design Principles for Digital Autopilot Synthesis

Howard Berman* and Richard Gran†
*Grumman Aerospace Corporation, Bethpage, N.Y.*

**This paper presents a control system synthesis technique for direct, computer-aided design of digital control system software with performance improvements in the areas of: sample time determination, control/structural interactions, noise effects (including gusts), dynamic response characteristics, and reduced sensor requirements. The design approach uses a newly developed set of digital control synthesis computer programs (known as DIGISYN). DIGISYN is based on stochastic control and estimation theory. Control requirements are specified as upper bounds for the state vector error and the desired vehicle response to either control commands or external disturbances. Using a system dynamics model that includes parasitic modes, DIGISYN collectively considers system and sensor noise and external stochastic disturbances (e.g., wind gusts) to determine the maximum permissible sample time, the optimal state estimator, and a set of feedback gains to yield the desired response characteristics. The unique feature of this approach is that sample time is determined by propagating the state covariance matrix until the specified error bounds are exceeded. The rationale for using this method to determine sampling time is that corrective action to reduce the system errors is only applied at the end of each sample period. This paper discusses an application of DIGISYN to pitch-plane control of the Grumman V/STOL Design 607A and verifies performance via a digital time history simulation.**

## I. Introduction

EVOLVING aircraft requirements include extended flight regimes for multimission aircraft and advanced control modes (e.g., automatic landing, speed command, load alleviation) to obtain improved performance. This increasing demand on flight control systems plus the trend toward increased use of digital computers in aircraft mandates an organized methodology for "digital autopilot" design. In this paper, digital autopilot is defined as the software link between flight control commands (generated manually or via an automatic guidance law), vehicle motion sensors, and actuators.

This paper will help answer the following often asked digital autopilot design questions[1,2]: 1) What are the sampling requirements? 2) What are the definitions and values of the digital control parameters including feedback control gains, digital filter equations, and filter gains?

In addition to an organized procedure for the design of a digital autopilot, the approach presented solves the digital control problem directly. Historically and seemingly the easiest approach to the control problem was to mimic the control system design using analog techniques. Attempts to duplicate an analog system generally result in a much higher sampling rate requirement than the approach presented here. Converting control system equations from analog to digital form adds the unnecessary trial-and-error step of specifying a sample time that "adequately" models the system dynamics in discrete time.

## II. Control System Synthesis: DIGISYN

A basic premise implied in the use of DIGISYN is the separation of the deterministic linear control problem from the stochastic estimation problem (the Separation Theorem[3]). That is, the optimum set of control gains to satisfy predescribed performance requirements is determined independently from the optimum state estimator.

A discrete time representation of the control loop (Fig. 1) shows the items that DIGISYN determines: sample time $(T)$, control gains $(G_K)$, state estimator weights $(W_K)$, and discrete transition equation coefficients $(\Phi_K, F_K)$.

The following paragraphs summarize the functions provided by DIGISYN and, because of the uniqueness associated with determining sample time, a mathematical basis and an expanded discussion of this function is given.

### Sample Time Determination

The sampling rate requirement is a direct function of how often control law computations (i.e., algorithms that transform desired commands to actuator commands) must be updated. Determination of the sampling time for a digital control system has been one of the many different trial-and-error problems the designer must solve. Past solutions to this problem required as a rule of thumb that the sampling frequency be 10 times faster than the highest frequency contained in the waveform to be sampled. This was loosely justified by the Sampling Theorem due to Shannon and empirical results. However, in practice one can use sampling frequencies that are much lower. To illustrate this point, consider a single input system of first order with a known arbitrary input $u$ described by the differential equation:

$$\dot{x} + ax = au \tag{1}$$

Based upon the rule of thumb procedure, one would design a feedback controller to give the input $u$ with a sampling time of $2\pi/10a$ sec (10 times the "highest" frequency which is $a$ rad/sec). However, one may simply write the solution to Eq. (1) as follows:

$$x(t) = e^{-a(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{-a(t-\tau)}au(\tau)d\tau \tag{2}$$

From Eq. (2), it is quite clear that the only unknown is

Presented as AIAA Paper 73-848 at the AIAA Guidance and Control Conference, Key Biscayne, Fl., Aug. 20–22, 1973; submitted August 31, 1973; revision received December 20, 1973.

Index categories: Navigation, Control, and Guidance Theory; Aircraft Handling, Stability, and Control.

*Control Systems Analyst.

†Head, Dynamics and Control Theory Research Branch.

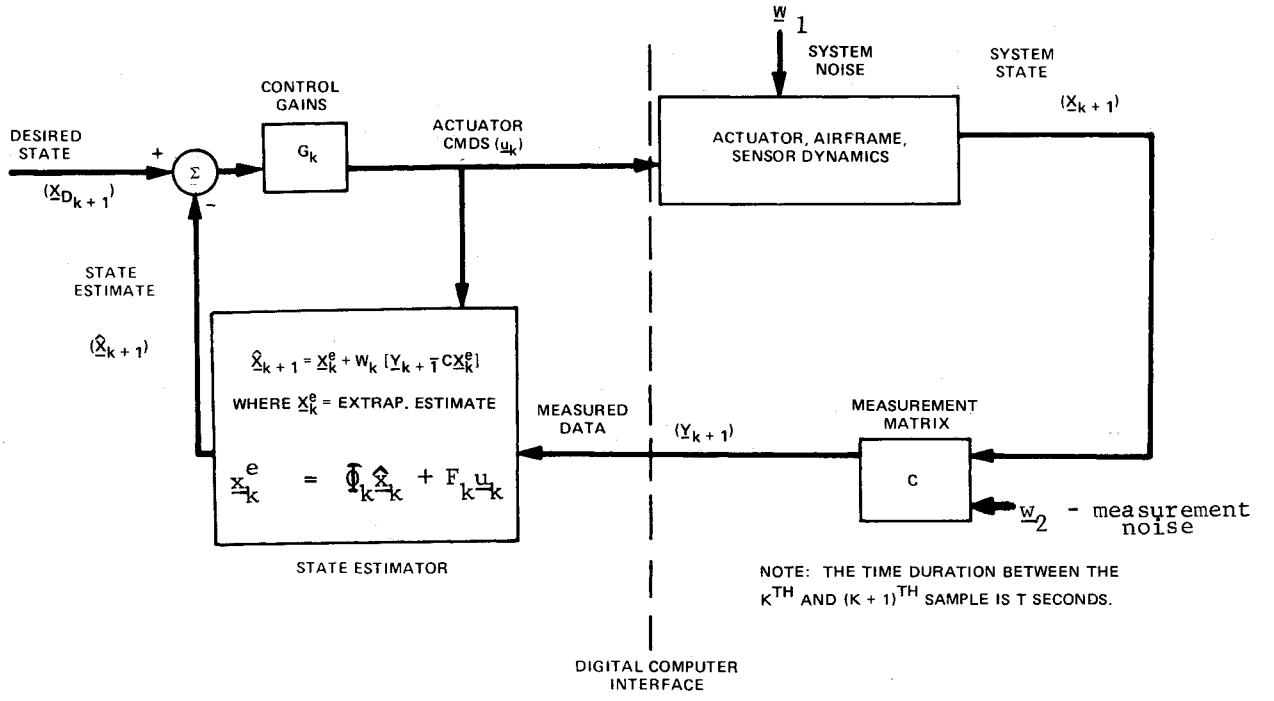


**Fig. 1 Generic discrete control loop, parameters obtained via DIGISYN.**

$x(t_0)$ so that only one sample is needed in order to specify completely the solution $x(t)$, and thereby to specify the control $u(t)$. This result is contrary to the rule of thumb discussed above. According to this result, sampling is not required. This certainly defies our intuition about control of systems, and it quite naturally makes a person skeptical of the conclusion. What is the trick?

The trick is that in writing Eq. (1), it was assumed that this was the actual equation that described the system. In practice, "$a$" for a real design is imprecisely known. In addition, there are various noise sources that enter the system and the measurements. These uncertainties are also not accounted for when using the sampling theorem rule of thumb. The introduction of these uncertainties actually makes the bandwidth of the system very large so that the rule-of-thumb sampling time would also have to be correspondingly large.

The reason for the differences in the approaches to the sampling time lies in the fundamental statement of the sampling theorem. The sampling theorem asks, "What is the least amount of information needed in order to reconstruct a time signal from all of its sample values?" The answer is, of course, that the highest frequency component in the signal must be less than or equal to one-half of the sampling frequency. However, in practice we know much more about the system than its highest frequency component. Also, in controller designs, decisions must be made with only limited numbers of past samples.

Pursuing the system approach described above, one can take into account the effect of the uncertainties in the measurement and the system. Again consider Eq. (1), but add a disturbance noise. Thus

$$\dot{x} + ax = au + bw_1 \tag{3}$$

where $w_1$ = white noise. Also, assume that the measurements of $x$ are made through a noisy channel. That is, $y$ (the measurement) will be

$$y = x + cw_2 \tag{4}$$

where $w_2$ is white noise independent of $w_1$.

The situation now is representative of the "real world," and the selection of a sampling time can proceed in an orderly manner. Unlike the situation that existed in the noiseless case, when we make a measurement at $t = 0$, $y(0)$, we can no longer predict the future behavior of the system with certainty; however, we can predict bounds on the system behavior. To do this, after having made the measurement and knowing the uncertainty introduced by the noise $w_2$, the actual state $x(0)$ can be expected to be within some region around the measurement [e.g., $y(0) \pm 3\sigma_2$, where $\sigma_2$ is the standard derivation of the measured noise]. Using Eq. (3), the upper and lower bound on the measurement may be propagated into the future. As the propagation proceeds (see Fig. 2), the uncertainty in the location of the actual system output tends to increase. Control systems are designed to maintain outputs within specified bounds; inevitably, the uncertainty will exceed these bounds. (Since we are using the propagated response to control the system, too large an error would

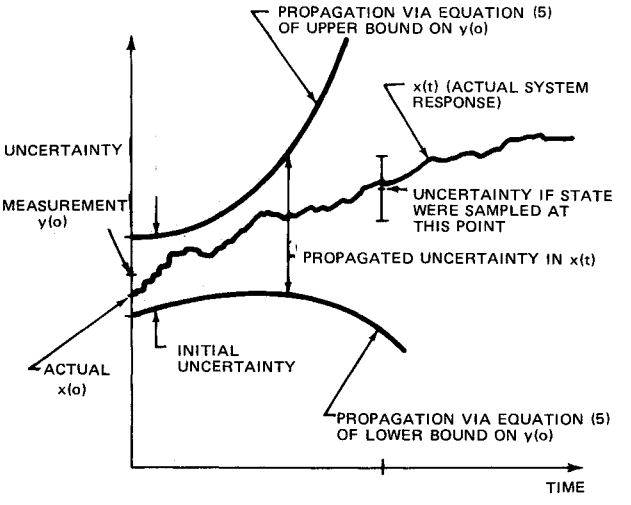


**Fig. 2 Uncertainty vs sampling time.**

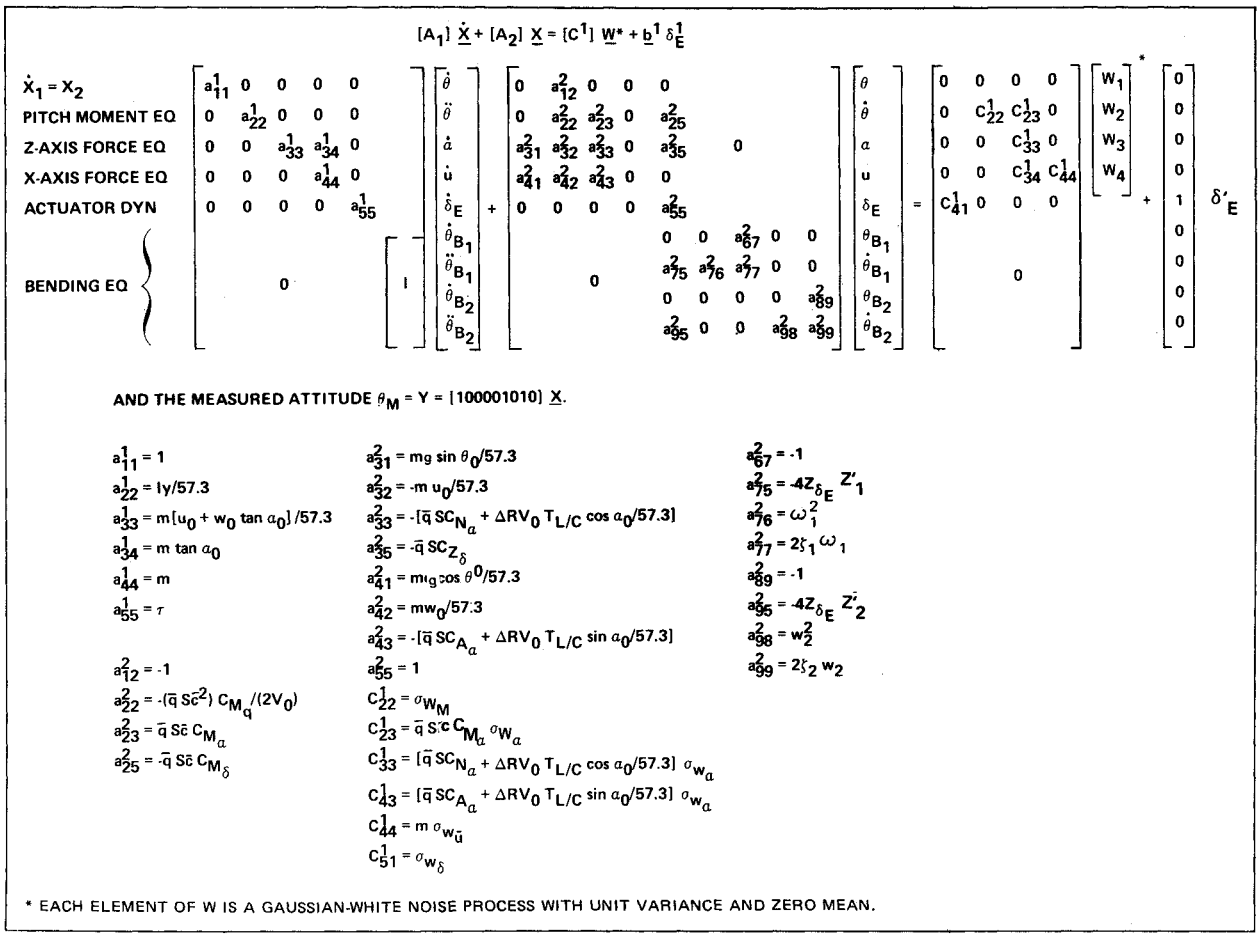


**Fig. 3 Pitch plane perturbation equations with noise.**

cause control degradation.) At this time, one must sample again to provide a tolerable error. This technique is used in DIGISYN to determine the sampling time. Specifically, sample time is determined by propagating the state covariance matrix until the specific state vector error bounds are exceeded; the time duration to exceed the error bounds defines the required sample time—the rationale being that corrective action to reduce the errors is resumed at the end of each sample period. Since the covariance matrix is the solution of the equation

$$\dot{P} = AP + PA^T + Q \tag{5}$$

where $A$ is the system matrix and $Q$ is the covariance matrix of the system excitation noise, the solution of the equation gives the upper and lower bounds of Fig. 2. The procedure is to assume that, at the start of a sample period, all uncertainty is eliminated (i.e., $P(0) = [0]$). Then $P(t)$ is numerically computed within DIGISYN using Eq. (5) and some small iteration time ($\delta t$). After some time $T = K\delta t$, one or more of the variances will exceed boundaries that are imposed by the control system performance criteria. These criteria have been converted into covariance matrix bounds by assuming they are $3\sigma$ values. The elapsed time $(T)$ is the sampling time.

In addition to system noise disturbances, pilot input changes will necessitate computation updates. These inputs can be monitored on an interrupt basis (to avoid missing impulsive type commands) and then stored to await execution. Finally, it is emphasized that a significant improvement in sampling requirement (i.e., longer sampling intervals) can be realized by applying this synthesis procedure as opposed to "mimicking" an analog system. The primary reasons for this improvement is that the sampling constraints imposed by digitizing an analog system are due to the combination of numerical integration problems and additional uncompensated lags inherent to the sampling process. The design approach presented here avoids these difficulties by taking advantage of our a priori knowledge of the system (i.e., the deterministic dynamics as well as the expected stochastic disturbances) to develop the state transition equations. This allows us to extrapolate the state vector in flight from one sample instant to the next as a function of known actuator commands. The existence of this type of state estimator eliminates the need for numerical integration and implies a foreknowledge of future system behavior that is not available with a classically designed analog system. Furthermore, given an estimate of the state, one can derive feedback gains to achieve a desired response while accounting for the delays introduced by sampling.

### Control Gain Determination

The formulation of the state transition equations is required to determine both the state estimator equations and the set of feedback gains. A DIGISYN algorithm initially transforms the set of differential equations describing the open-loop system (including rigid airframe dynamics, body bending modes and any other parasitic modes, and actuator and sensor dynamics) into the following state vector representation:

$$\dot{x} = Ax + Bu \tag{6}$$

where: **x** is the system state and **u** is the control command to the actuators. Given a sample time ($T$ sec), the equivalent discrete system is then formulated, using some assumption about the nature of the control between samples (e.g., O-order hold), as:

$$x_{K+1} = \Phi_K x_K + F_K u_K \tag{7}$$

where: $\mathbf{x}_{K+1}$ and $\mathbf{x}_K$ are the states, one sample ($T$ sec) apart.

DIGISYN will determine gains to provide a desired closed-loop pole configuration. It has been shown in the literature[4] that characteristic roots of a closed-loop system can be forced to a set of any desired locations as long as the system state is completely known (or estimated). Assuming the existence of a complete estimate of the state $\hat{\mathbf{x}}_K$, the control loop is closed by weighting this estimate by the gain matrix, $G_K$, to yield control commands to the actuators. Referring to Eq. (7), the closed-loop discrete transition equation with no command input can now be represented by

$$x_{K+1} = [\Phi_K - F_K G_K] x_K \tag{8}$$

assuming $\mathbf{x}_K = \hat{\mathbf{x}}_K$, a perfect state estimate. DIGISYN will derive a $G_K$ so that the eigenvalues of $[\Phi_K - F_K G_K]$ correspond to the desired closed-loop, $Z$-plane, characteristic roots.

It should be noted that although, in theory, a complete state estimate is required to precisely match a set of desired poles, a part of the design procedure is to perform additional simulation analyses to determine the extent to which the number of feedback terms can be reduced without severely penalizing system performance.

#### State Estimator Determination

The state estimator consists of an extrapolation of the previous estimate that is adjusted by the weighted difference between the measured data and the extrapolated state. Specifically

$$\hat{x}_{K+1} = \hat{x}_K^e + W_K[Y_{K+1} - C\hat{x}_K^e] \tag{9}$$

where $Y_K$ is the measured data, and $\hat{\mathbf{x}}_K^e = \Phi_K \hat{\mathbf{x}}_K + F_K \mathbf{u}_K$ is the extrapolated estimate.

The discrete transition matrix ($\Phi_K$) and the control influence matrix ($F_K$) are formulated using the previously described DIGISYN routine that converts the continuous system to a discrete equivalent system. The filter weights ($W_K$) derived by this synthesis routine, are discrete Kalman filter steady-state weights[5] (i.e., $W_K$ is formulated so that the mean square error between the estimated state and the true state is minimized). The process of deriving these optimal weights uses a system model (including body bending modes or any other parasitic modes) augmented by a noise dynamics model (e.g., wind gusts, computer noise, sensor noise, computer round-off errors, and engine vibration).

Once again additional sensitivity analyses are necessary to determine the extent to which terms can be eliminated from the full-state estimator to arrive at a minimum configuration for onboard digital computer mechanization.

## III. Sample Application: V/STOL Pitch Plane Cruise

#### Equations of Motion and Control Mode Formulation

Solutions of the linear control problem for each critical flight condition provide the control system configuration. The control parameters, which are defined in Fig. 1, are obtained for each flight condition through the use of DIGISYN. The first synthesis task is to define the linear differential equations of the plant. The plant is assumed to be the pitch axis of a V/STOL consisting of the rigid airframe, two bending modes, and a power actuator as shown in Fig. 3. (The symbols are defined in Table 1.) As the equations indicate, the system state is 9th order and is defined by,

$$\begin{bmatrix} \theta \\ \dot{\theta} \\ \alpha \\ u \\ \delta_E \\ \theta_{B_1} \\ \dot{\theta}_{B_1} \\ \theta_{B_2} \\ \dot{\theta}_{B_2} \end{bmatrix} = \begin{bmatrix} \text{attitude} \\ \text{attitude rate} \\ \text{angle-of-attack} \\ x\text{-axis velocity} \\ \text{elevator angle} \\ \text{Mode 1 displacement} \\ \text{Mode 1 rate} \\ \text{Mode 2 displacement} \\ \text{Mode 2 rate} \end{bmatrix}$$

A flight control mode is defined by the instantaneous desired perturbation state ($\mathbf{x}_D$). For demonstration purposes, the desired state vector corresponding to a rate command-attitude hold mode is derived. Before proceeding, note that only desired rigid-body states are formulated. This paper will not address the problem of actively controlling the bending modes although such control would require only a trivial modification of the equations presented here. We include bending modes as part of the state estimator; thus, accurate estimates of rigid body motions from sensed data corrupted by elastic deflections are possible. The objective is to avoid control interactions with these structural modes and in turn to allow all elastic deflections to damp naturally.

In order to realize a rate command mode, the instantaneous desired attitude rate, $\dot{\theta}_D$, is proportionally equated to the stick displacement $\delta_S$, or

$$\dot{\theta}_{D_K} = K_S \delta_{S_K} \tag{10}$$

It then follows that an approximate pitch attitude, $\theta_D$, is given by

$$\theta_{D_K} = \dot{\theta}_{D_{K-1}} T + \theta_{D_{K-1}} \tag{11}$$

where $T$ is the sample time.

Remaining desired state vector terms are necessary to guarantee consistency to the steady-state response (including the phugoid mode and the closed-loop actuator mode). In fact, subsequent time history simulation results demonstrate adequate short-period-response characteristics with just a two-element desired state ($\theta_D$ and $\dot{\theta}_D$). The desired state vector elements derived below are used to provide the relationships necessary to perform subsequent simulation tests comparing performance of a reduced system with that of the complete state vector command system.

The third desired state vector element, angle of attack ($\alpha_D$), is obtained by returning to the equations of motion and solving for the steady-state $\alpha$ corresponding to a steady-state pitch rate (e.g., $\dot{\theta} = \dot{\theta}_D$, $\ddot{\theta} = 0$). The result is

$$a_{D_K} = \frac{\left[\dfrac{mC_{M_{\delta_E}}}{57.3}(u_0 + w_0 \tan a_0) - \bar{q}\dfrac{S\bar{c}}{2V_0} C_{M_q} C_{Z_{\delta_E}}\right]\dot{\theta}_{D_K}}{\bar{q}SC_{M_a}C_{Z_{\delta_E}} - C_{M_{\delta_E}}\left[\bar{q}S(C_{N_a} - C_{A_a}\tan a_0) + \dfrac{\Delta R V_0 T_{L/C}}{57.3}\left(2\cos a_0 - \dfrac{1}{\cos a_0}\right)\right]} \tag{12}$$

The desired $x$-axis velocity, $u_D$, is derived by noting that a steady-state pitch rate requires a steady-state $\ddot{u}$. That is

$$\ddot{u}_{D_K} = \frac{g\cos\theta_0}{57.3}\dot{\theta}_{D_K} \tag{13}$$

It then follows that an approximate $\dot{u}$ is:

$$u_{D_K} = \ddot{u}_{D_{K-1}} T + \dot{u}_{D_{K-1}} \text{ and } u_{D_K} = \frac{1}{2}\ddot{u}_{D_{K-1}} T^2 + \dot{u}_{D_{K-1}} T + u_{D_{K-1}}$$

**Table 1 Symbols for pitch plane equations and representative data**

| Symbol | Definition | V/STOL data for a low-speed flight condition |
|---|---|---|
| $\bar{c}$ | Reference chord | 12.96 ft |
| $C_{M_q}$ | Nondimensional pitching moment due to pitch rate | 0.209 1/deg |
| $C_{M_\alpha}$ | Slope of nondimensional pitching moment as a function of $\alpha$ | 0.00467 1/deg |
| $C_{A_\alpha}$ | Slope of nondimensional axial force as a function of $\alpha$ | 0.012 1/deg |
| $C_{N_\alpha}$ | Slope of nondimensional normal force as a function of $\alpha$ | 0.0575 1/deg |
| $C_{M_{\delta_E}}$ | Nondimensional pitching moment due to tail deflection | 0.00846 1/deg |
| $C_{Z_{\delta_E}}$ | Nondimensional normal force due to tail deflection | 0.0078 1/deg |
| $g$ | Gravity acceleration constant | 32.174 fps$^2$ |
| $I_y$ | Airframe pitch inertia | 73740 slug-ft$^2$ |
| $m$ | Airframe mass | 931.67 slugs |
| $\bar{q}$ | Dynamic pressure | 104.3 lb/ft$^2$ |
| $\Delta R$ | Engine ram drag per lb of thrust | 0.00052 lb/fps-lb |
| $S$ | Reference area | 365 ft$^2$ |
| $T_{L/C}$ | Cruise engine thrust | 7993 lb |
| $u$ | Small perturbation axial velocity | |
| $u_0$ | Axial velocity, reference condition ($x$-axis) | |
| $V_0$ | Total velocity, reference condition | 296 fps |
| $W_0$ | Normal velocity, reference condition ($Z$-axis) | |
| $Z_{\delta_E}$ | Normal force as a function of $\delta_E$ | 0.3182 ft/sec$^2$-deg |
| $Z'_{1,2}$ | Bending displacement weighting factor for Modes 1 and 2 | 0.05, 0.175 (typical) |
| $\alpha$ | Small perturbation angle of attack | |
| $\alpha_0$ | Angle of attack, reference condition | |
| $\theta$ | Small perturbation inertial pitch attitude | |
| $\theta_0$ | Inertial pitch attitude, reference condition | 9.5 deg |
| $\delta_E$ | Small perturbation elevator angle | |
| $\delta'_E$ | Command to elevator actuator | |
| $\tau$ | Elevator actuator time constant | 0.05 sec |
| $\sigma_{wm}$ | Standard deviation of noise on pitch moment | |
| $\sigma_{w\alpha}$ | Standard deviation of noise on $\alpha$ | |
| $\sigma_{w\dot{u}}$ | Standard deviation of noise on $\dot{u}$ | |
| $\sigma_{w\delta_E}$ | Standard deviation of noise on $\delta_E$ | |
| $\omega_{1,2}$ | Bending mode natural frequency | 25, 50 rad/sec |
| $\zeta_{1,2}$ | Bending mode structural damping | 0.01 |

Finally, the desired actuator angle $\delta_{(ED)}$ is obtained by solving for the steady-state elevator angle that is required for maintaining a constant pitch rate of $\theta_D$. Specifically

$$\delta_{E_{D_K}} = \frac{\left\{\frac{C_{M_q}\bar{C}}{2V_0}\left[\bar{q}S(C_{N_\alpha} - C_{A_\alpha}\tan\alpha_0) + \frac{\Delta R V_0 T_{L/C}}{57.3}\left(2\cos\alpha_0 - \frac{1}{\cos\alpha_0}\right)\right]\frac{mC_{M_\alpha}}{57.3}[u_0 - W_0\tan\alpha_0]\right\}\dot{\theta}_{D_K}}{\bar{q}SC_{M_\alpha}C_{Z_{\delta_E}} - C_{M_{\delta_E}}\left[\bar{q}S(C_{N_\alpha} - C_{A_\alpha}\tan\alpha_0) + \frac{\Delta R V_0 T_{L/C}}{57.3}\left(2\cos\alpha_0 - \frac{1}{\cos\alpha_0}\right)\right]} \tag{14}$$

Although Eqs. (12-14) are algebraically cumbersome, they are not intended for "onboard" calculations, but rather to develop command weighting constants as a function of flight conditions. For example, applying the data listed in Table 1, the algorithms define the desired state corresponding to a rigid body state vector command as (note that no bending states are commanded)

$$\left\{\begin{array}{ll}
\dot{\theta}_{D_K} = K_S\delta_{S_K} & \text{-State 2}\\
\theta_{D_K} = \dot{\theta}_{D_{K-1}}T + \theta_{D_{K-1}} & \text{-State 1}\\
a_{D_K} = 2.42\,\dot{\theta}_{D_K} & \text{-State 3}\\
\ddot{u}_{D_K} = 0.5545\,\dot{\theta}_{D_K} & \\
\dot{u}_{D_K} = \ddot{u}_{D_{K-1}}T + \dot{u}_{D_{K-1}} & \\
u_{D_K} = \frac{1}{2}\ddot{u}_{D_{K-1}}T^2 + \dot{u}_{D_{K-1}}T + u_{D_{K-1}} & \text{-State 4}\\
\delta_{E_{D_K}} = -1.88\dot{\theta}_{D_K} & \text{-State 5}
\end{array}\right. \tag{15}$$

**Applying DIGISYN**

Returning to the system of equations presented in Fig. 3, we see that the system is corrupted by white-Gaussian noise processes. The noise sources considered for this sample problem are: 1) Vertical wind gusts that directly affect $u$ and $\alpha$, 2) Computer interface noise and round-off that directly affect actuator commands, 3) Engine vibration that directly affects both $x$-axis and pitch acceleration, and 4) Elevator vibration that directly affects $\delta_E$.

Table 2 lists magnitudes for these disturbances along with the states and state derivatives that are directly affected and their corresponding $1\sigma$ uncertainty levels.

The next major step in the synthesis process is to determine the required sample time. As noted previously, the sample time is based upon a specified "State Vector Bound" (or $3\sigma$ state variation), and for this sample problem is chosen as

a) Attitude = 0.5°
b) Attitude rate = 5°/sec
c) Angle of attack = 2°
d) $x$-axis velocity = 2 fps
e) Elevator angle = 1°
f) Mode 1 displacement = 0.05°
g) Mode 1 rate = 0.5°/sec
h) Mode 2 displacement = 0.05°
i) Mode 2 rate = 0.5°/sec.

The required sample time is determined to be 163 msec. Specifically, in the absence of control updates (i.e., between sampling instants), a system exposed to the disturbances given in Table 2 will be constrained within the above bounds, provided the time between samples does not exceed 163 msec. Note that using the conventional "rule of thumb" a sampling time of 12.5 msec would be required which is *an order of magnitude faster than it need be.*

The next control synthesis step is to convert the continuous system into a discrete system by deriving the matrices needed to formulate the state transition equation

**Table 2 Noise description**

| Noise source | Std deviation of source | Noise influenced state or state derivative | |
|---|---|---|---|
| | | Element | Std dev |
| Wind gusts | 15 fps | Angle-of-attack | $\sigma_{w_\alpha} = 2.9°$ |
| | | $x$-axis velocity | $\sigma_{w_u} = 2.5$ fps |
| Engine vibration | 5% thrust var | $x$-axis accel | $\sigma_{w_{\dot{u}}} = 0.43$ fps$^2$ |
| | 0.5° TV angle | pitch moment | $\sigma_{w_m} = 896$ ft-lb |
| Computer interface | 2 bits of a 16-bit word | Elev. angle | $\sigma_{w_\delta} = 0.1°$ |
| Elev. vibration | 0.1° | | |

$$x_K = \Phi_K x_{K-1} + F_K u_{K-1} \tag{16}$$

For example, the resulting $\Phi_K$ and $F_K$ matrices, corresponding to a 150 msec sample time, are obtained from DIGISYN as:

$$\Phi_K = \begin{bmatrix} 1.0 & 0.1311 & -0.01797 & 0 & -0.01488 & 0 & 0 & 0 & 0 \\ 0 & 0.7525 & 0.226 & 0 & -0.1256 & 0 & 0 & 0 & 0 \\ 0 & 0.1264 & 0.9159 & 0 & -0.01721 & 0 & 0 & 0 & 0 \\ -0.0831 & -0.1125 & 0.08687 & 1.0 & 0.0126 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0497 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.00143 & -0.7962 & -0.02195 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.10677 & 13.72 & -0.7853 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.00248 & 0 & 0 & 0.3363 & 0.01713 \\ 0 & 0 & 0 & 0 & 0.213 & 0 & 0 & -42.826 & 0.3191 \end{bmatrix}$$

$$F_K = \begin{bmatrix} -0.01839 \\ -0.29763 \\ -0.02409 \\ 0.01555 \\ 0.9503 \\ 0.881 \times 10^{-2} \\ 0.02856 \\ 0.313 \times 10^{-2} \\ 0.495 \times 10^{-2} \end{bmatrix}$$

The steady-state Kalman filter weights are formulated next. This sample problem assumes only an attitude measurement; that is, the measurement $Y_K$ is related to the state vector as follows:

$$Y_K = [1\ 0\ 0\ 0\ 0\ 1\ 0\ 1\ 0]X_K = CX_K \tag{17}$$

The resulting Kalman weights are calculated to be

$$W_K = \begin{bmatrix} 0.9951 \\ 7.853 \\ 3.247 \\ -3.272 \\ -0.203 \times 10^{-4} \\ 0.485 \times 10^{-2} \\ -0.09799 \\ 0.1525 \times 10^{-4} \\ 0.05141 \end{bmatrix}$$

The State Estimator has now been completely defined; that is, all coefficients appearing in Eq. (9) have been determined.

The final step in the synthesis process using DIGISYN is the formulation of the control gains via pole matching. The desired characteristic closed-loop roots have been specified in terms of their natural frequency and damping given in Table 3.

The computer gain vector that will realize the above closed-loop characteristics is calculated to be

$$G_K = [0.135\ \ 0.815\ \ 1.60\ \ -0.042\ \ 0.467]$$

This gain vector is the rigid body portion of the full-state gain vector. The command to the elevator actuator, as a function of state errors, can now be expressed as

**Table 3 Desired closed-loop poles**

| Mode | Nat. freq, rad/sec | Damping | Time constant, sec |
|---|---|---|---|
| Short period | 2.86 | 0.7 | 0.5 |
| Phugoid | 0.05 | 0.2 | 100.0 |
| Actuator | 25.0 | 1.0 | 0.04 |

$$\delta_{E_K}' = u_K = -0.135(\theta_{D_K} - \hat{\theta}_K) - 0.815(\dot{\theta}_{D_K} - \hat{\dot{\theta}}_K) - 1.6(\alpha_{D_K} - \hat{\alpha}_K) + 0.042(u_{D_K} - \hat{u}_D) + 0.467(\delta_{E_K} - \hat{\delta}_{E_K}) \tag{18}$$

At this point, all parameters needed to form the discrete control loop as defined in Fig. 1, have been determined. The remaining synthesis tasks include performance verification and attempts at simplification such as reducing the estimator and eliminating state vector command elements. These remaining tasks are performed with the aid of a time history simulation.

### Design Verification and Simplification

The digital control system previously synthesized with the help of DIGISYN is verified via a numerical integration simulation of the small perturbation pitch-plane equations presented in Fig. 3. A rate command-attitude hold mode is mechanized using sensed attitude and deriving the remaining state variables. Time response performance is verified using a sampling time of 150 msec. This sampling requirement was previously established to satisfy error state constraints; once established, the discrete system control gains ($G_K$) were formulated so that the short-period response would have a 0.5-sec time constant.

The first series of tests were performed without structural bending. Vehicle response to a step rate command is shown in Fig. 4. The short-period response is as predicted. Specifically, steady state is reached after approximately 1.5 sec, which is equivalent to three time constants. Although this response time verifies the initial design, there is a significant attitude rate overshoot observed. This overshoot illustrates a pole-matching drawback resulting from no direct control over the placement of closed-loop

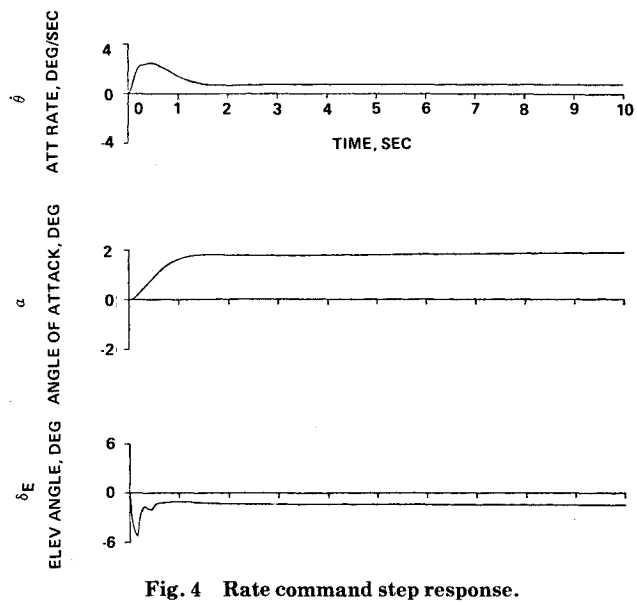


**Fig. 4 Rate command step response.**

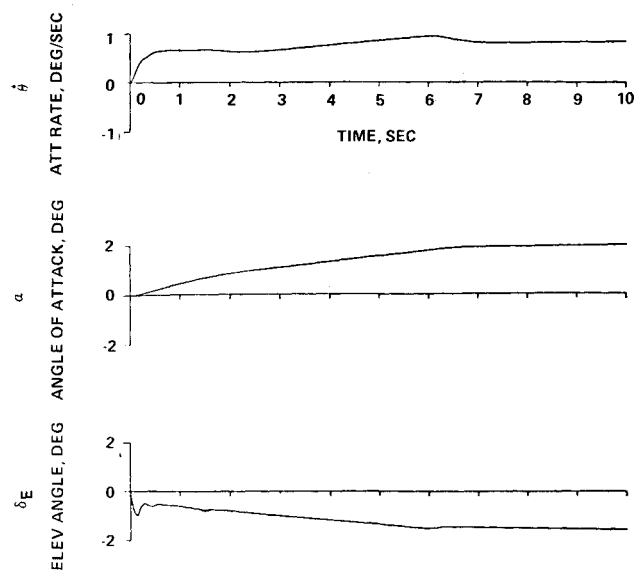


Fig. 5 Rate command step response with command smoothing.

zeros. In summary, response time, and of course stability, can be specified as pole-matching design criteria; however, the percentage overshoot cannot. Fortunately, the above drawback can usually be remedied by smoothing the state commands. For example, by simply changing from a zero-order hold to an approximate first-order hold on the rate and the angle of attack, the vehicle response characteristics can be altered from that shown in Fig. 4 to that shown in Fig. 5. Specifically, the step rate command was identical for each of the two test conditions; however, for the test results shown in Fig. 5, the desired rate and the desired angle of attack were accumulated at increments not exceeding 10% of the commanded values corresponding to the present stick displacement. This additional command logic eliminated the rate overshoot as well as changed the apparent rate response time from 1.5 to 0.75 sec. This is an excellent example of how powerful a digital controller can be; that is, a simple software change can effectively fine tune the time response in minimum time and at negligible cost.

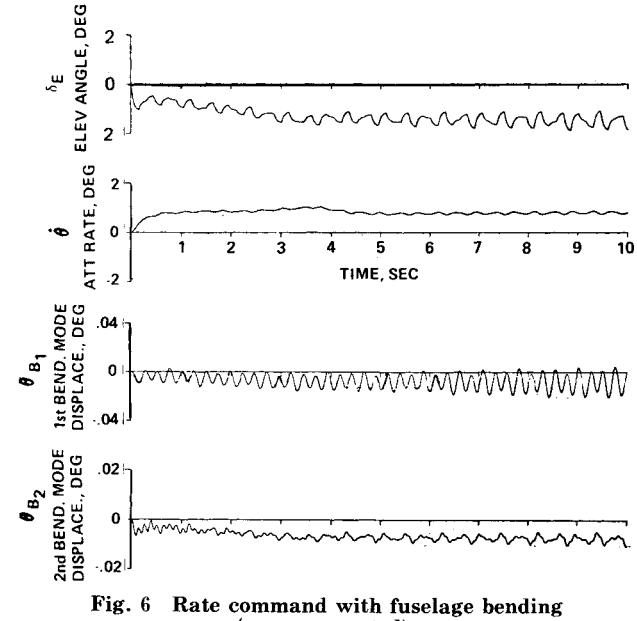


Fig. 6 Rate command with fuselage bending (uncompensated).

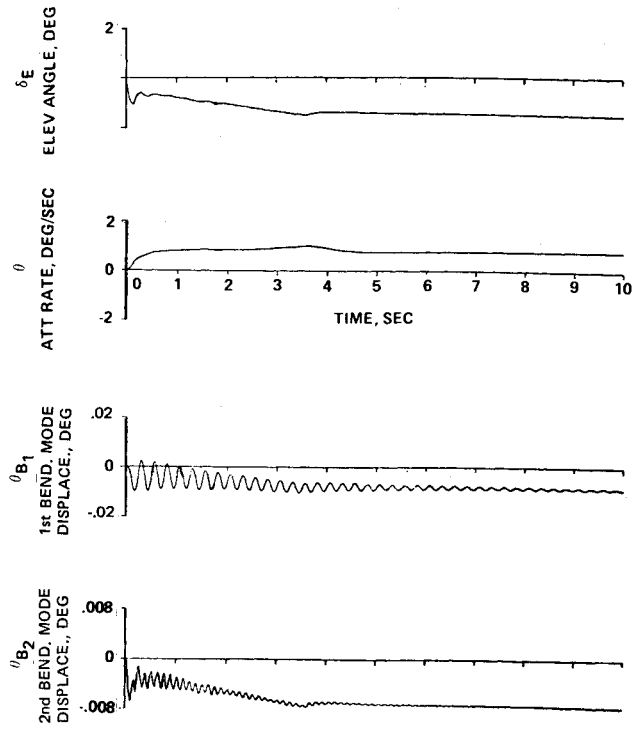


Fig. 7 Rate command with bending model as part of state estimator.

The same test conditions as above were rerun with the first two fuselage bending modes included in the airframe model; the results are shown in Fig. 6. The two modes are characterized with natural frequencies of 25 and 50 rad/sec and 1% structural damping. The time history traces exhibit an obvious control/structural interaction that results in an instability. The conventional remedy would be a digital notch filter whose sampling time requirement would be as low as 12 msec (i.e., 1/10th the period of the filter's natural frequency). The approach used here is to retain the 150 msec sampling time and to include a bending model as part of the state estimator so as to obtain a better estimate of rigid body motion. The success of this compensation approach is demonstrated by the traces presented in Fig. 7. Although the initial elevator response to the step rate command does excite the bending modes, these elastic motions are allowed to damp naturally with no visible interactions with the control system. The significance of these results is that bending compensation was accomplished using a sampling rate less than the bending frequency.

The next series of simulation tests was performed to investigate the effects of simplifying the control algorithms. Until now, the desired conditions for each rigid body state was derived [Eq. (15)]; rate command-attitude hold performance is now examined using just a two-state ($\theta$ and $\dot{\theta}$) command vector. This implies just two feedback elements that in turn reduces the state estimator order and the complexity of the control commands [Eq. (18)]. The time response characteristics of the reduced system to a step rate command is shown in Fig. 8. An apparent steady-state is reached within 1.5 sec with an approximate 20% overshoot. In addition, there are no observable control/structural interactions. Command smoothing (which was not included during these tests) is considered unnecessary since the overshoot was not excessive. There is a slight drift in attitude rate that is attributed to a shift in the phugoid mode. From investigations using the reduced state estimator, the following can be deduced: 1) in order

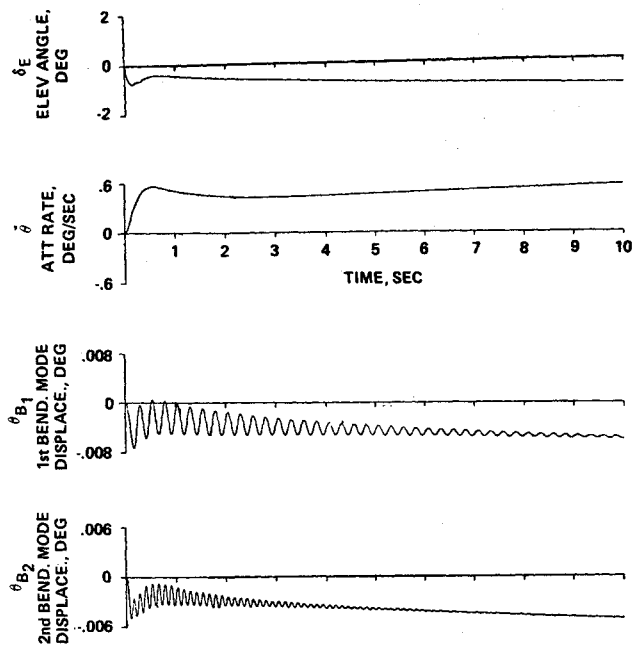


**Fig. 8 Reduced system, rate command.**

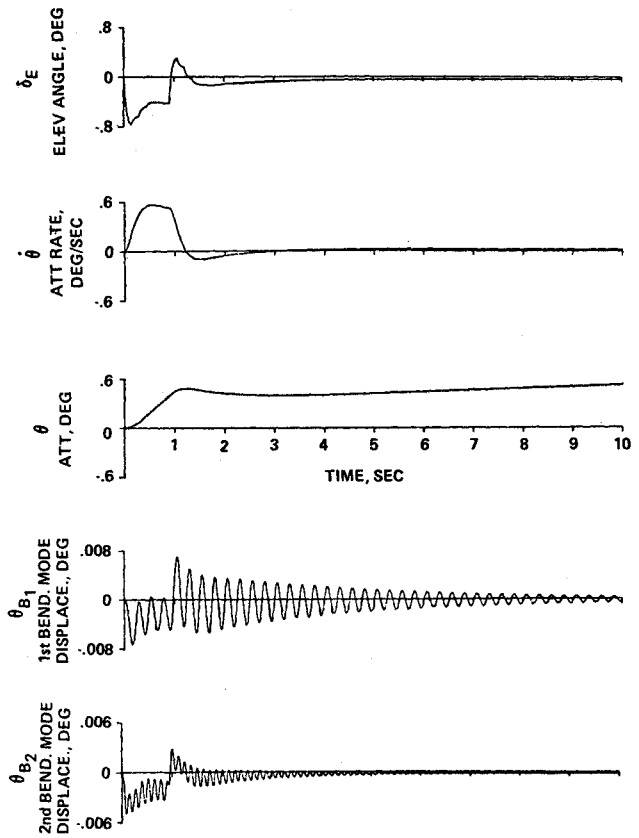


**Fig. 9 Reduced system, rate command attitude hold (stick pulse command).**

to obtain a reasonable bending mode estimate, the system model must include actuator dynamics or any other high-frequency effects of the same order of magnitude as the bending mode, and 2) although our ultimate objective is to derive $\dot{\theta}$ from $\theta$, a derived estimate of $\alpha$ is also needed for good results.

The set of traces presented in Fig. 9 demonstrates reduced system rate command-attitude hold performance in response to a pulse stick command. Once again, all initial elastic deflections are permitted to damp naturally without any observable interactions with the control system. Following the 0.9-sec pulse command, an attitude reference is established and is held within acceptable limits.

Finally, the algorithms associated with the reduced system are not considered overly complex even when compared to an equivalent digitized analog system. An equivalent digitized analog system would include a double-notch filter, and some means for deriving attitude rate from attitude measurement data.

## IV. Conclusions

Conventional digital synthesis techniques, such as digitizing an analog designed system, impose unnecessary restrictions on the resulting digital system. The approach presented in this paper utilizes a state estimate to develop the necessary feedback. This is made possible by taking advantage of both: an a priori knowledge of the system to be controlled (i.e., the disturbances as well as the dynamic characteristics), and the computational capabilities of a digital machine. The general control loop organization presented, where estimator outputs are compared with desired states derived from controller inputs or guidance commands, is not a new approach and in fact was successfully demonstrated by Apollo generation autopilots.[6,7] This mechanization approach lends itself to realizing advanced control modes (e.g., velocity control, or direct flight path control). That is, assuming all the necessary feedbacks are defined by the state estimator, different control modes can be realized by simply changing state vector command algorithms.

Rather than testing sampling requirements as an afterthought following the design of a continuous system, they are systematically determined by considering control accuracy requirements in combination with the aircraft's stochastic environment. Application of this approach demonstrates a potential order of magnitude improvement in sampling requirements.

Stability and response time requirements are achieved by closing the loop through gains that are derived specifically using a discrete time formulation. This paper presents results using $Z$-plane pole matching that although in general yields satisfactory results, it does have a disadvantage of not including percentage overshoot as part of the design criteria. An alternate means of determining the discrete closed-loop gains is Implicit Model Following where the control gains minimize quadratic cost functions that are based upon a desired system model.

The information provided in this paper is basically exploratory and, to date, the results are promising. We have yet to show performance sensitivity as a function of how well we can predict system characteristics; these results are forthcoming. In today's cost-conscious climate, it is easy to rely on old techniques as long as they work. Perhaps, for near-term applications this is the only feasible option. However, it has been shown that there is definite room for improvement, and as flight computer computational loads become more demanding, efficiency will be of concern.

## References

[1]Gran, R., "Design Principles for a Shuttle Digital Autopilot," *Proceedings of the Space Shuttle Integrated Electronics Conference,* NASA/MSC, NASA TMX-58063, Vol. 1, May 1971, pp. 347-364.

[2]Gran, R. and Zetkov, G., "On the Optimal Stochastic Control of Systems with Noisy Measurements," Grumman Rept. ADR

05-06-69.1, 1969, Grumman Aerospace Corp., Bethpage, N.Y.

[3]Joseph, P. D. and Tou, J. T., "On Linear Control Theory," *AIEE Transactions*, Vol. 80, Pt. II, Sept. 1961, pp. 193–196.

[4]Wonham, W. M., "On Pole Assignment in Multi-input Controlable Linear Systems," *IEEE Transactions on Automatic Control*, Vol. AC-12, Dec. 1967, pp. 660–665.

[5]Kalman, R. E., "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME Journal of Basic Engineering*, Vol. 82D, March 1960, pp. 33–45.

[6]Widnal, W. S., "The Minimum-Time Thrust Vector Control Law in the Apollo Lunar Module," *Automatica*, Vol. 6, No. 5, Sept. 1970, pp. 661–672.

[7]Cox, K., "A Case Study of the Apollo Lunar Module Digital Autopilot," *Proceedings of the IEEE Group on Automatic Control, 1969 JACC Case Studies in Systems Control*, University of Colorado, Boulder, Colo., Aug. 1969.

JULY 1974 J. AIRCRAFT VOL. 11, NO. 7

# Turbine Inlet Gas Temperature Measurement System Using a Fluidic Temperature Sensor

William L. Webb*
*Pratt & Whitney, West Palm Beach, Fla.*
and
Paul J. Reukauf†
*NASA Flight Research Center, Edwards, Calif.*

**A fluidic turbine inlet gas temperature measurement and control system was developed for use on a Pratt & Whitney Aircraft J58 engine. This paper includes the criteria used for material selection, system design, and system performance. It was found that the fluidic temperature sensor had the durability for flight test under the conditions existing in the YF-12 airplane. As a result of turbine inlet gas temperature fluctuations, over-all engine-control system performance cannot be adequately evaluated without a multiple gas sampling system.**

## I. Introduction

THE size and weight of a gas turbine are strongly affected by the maximum allowable turbine inlet gas temperature (TIGT). The maximum temperature is limited by the metallurgical properties of the turbine. Once an engine has been qualified and enters service, the effect of increased temperature is to decrease engine life and the effect of lower temperature is to increase thrust specific fuel consumption. Parametrically, the effect of temperature on engine life is as shown in Fig. 1. Thrust specific fuel consumption increases are on the order of 3%/100°F for modern engines. These relationships generate a requirement for a direct measurement of TIGT.

Turbine inlet gas temperature in the 2000°F to 2500°F range can be measured with noble metal thermocouples. However, because of low mechanical strength, thermocouple sensing junctions must be relatively large to achieve the life required in turbine engines, some models of which have overhaul cycles in excess of 10,000 hr. Since a thermocouple is a thermoelectric device, its response to temperature change is directly related to the size of the sensing junction. This phenomenon is illustrated in Fig. 2.

A fluidic temperature sensor can be used to measure TIGT. Fluidic sensors demonstrated the ability to operate at a 4000°F sensed temperature for short periods and provided rapid transient response. Fluidic sensors have a demonstrated extended life potential at 2200°F sensed temperature. Pratt & Whitney Aircraft, with Honeywell, Inc., as the subcontractor, has built a fluidic TIGT measurement and control system which will be flight tested on a YF-12 airplane at the NASA Flight Research Center, Edwards, Calif. References 1 and 2 provide development and capability information for fluidic temperature sensors upon which this program is based.

Presented as Paper 73-1251 at AIAA/SAE 9th Propulsion Conference, Las Vegas, Nev., November 5–7, 1973; submitted November 26, 1973; revision received April 1, 1974.

Index categories: Airbreathing Engine Testing; Airbreathing Propulsion, Subsonic and Supersonic.

*Senior Assistant Project Engineer, Control Engineering Department, Florida Research and Development Center.

†Aerospace Engineer.

## II. Fluidic Temperature Sensor Operation

The fluidic temperature sensor operates as a sustained acoustic oscillation with the frequency of oscillation an exponential function of the gas temperature. The transient response of a fluidic temperature sensor is comprised of two components. The first is a very rapid response and represents the time required to fill the oscillator cavity with a "new" sample of gas. The second is a slow response and represents the time required to bring the "wetted" sensor area up to the sensed gas temperature. The first response is a function of the oscillator cavity volume, the flow area into and out of the cavity, and the pressure ratio across the sensor. Typical flushing times are on the order of 0.002 to 0.01 sec. The second response is a function of the sensor mass that must be brought up to temperature and is on the order of 8 to 15 sec for refined immersed sensors. A more thorough description of this phenomenon is included in Ref. 3 and is shown pictorially in Fig. 3.

The uniqueness of this two-level response can be used to advantage in a high-response temperature measurement system. Since the heat transfer rate of a given metal body can be predicted by standard heat transfer equations and since the initial response has been shown to represent a percentage of the total temperature change, an electrical equivalent of the second response can be constructed and added to the initial response to approximate the final value of input temperature. This "added" input can then be removed at a rate equivalent to the heat transfer into