

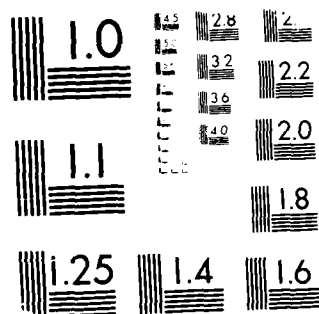
AD-A189 675

F-16 SIMULATOR FOR MAN-IN-THE-LOOP TESTING OF AIRCRAFT
CONTROL SYSTEMS (SINTACS) (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI... M W KASSAM
DEC 87 AFIT/GE/ENG/87D-30 F/G 1/4

1/2

UNCLASSIFIED

NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

DTIC FILE COPY

①

AD-A189 675



F-16 SIMULATOR FOR MAN-IN-THE-LOOP
TESTING OF AIRCRAFT CONTROL SYSTEMS
(SIMTACS)

THESIS

Mark W. Kassan
Second Lieutenant, USAF

AFIT/GE/ENG/87D-30

DTIC
LECTE
MAR 07 1988
S H D

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

88 3 01 0511

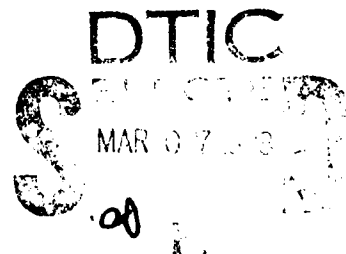
AFIT/GE/ENG/87D-30

F-16 SIMULATOR FOR MAN-IN-THE-LOOP
TESTING OF AIRCRAFT CONTROL SYSTEMS
(SIMTACS)

THESIS

Mark W. Kassan
Second Lieutenant, USAF

AFIT/GE/ENG/87D-30



Approved for public release; distribution unlimited

AFIT/GE/ENG/87D-30

F-16 SIMULATOR FOR MAN-IN-THE-LOOP
TESTING OF AIRCRAFT CONTROL SYSTEMS
(SIMTACS)

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering

Mark W. Kassan, B.S.E.E.
Second Lieutenant, USAF

December 1987

Approved for public release; distribution unlimited

Preface

This report is the culmination of my efforts to design and implement a real-time flight simulator on the two SIMSTAR hybrid computer systems in the School of Engineering. The report is written for readers with some background in aircraft dynamics, control system design and the computer languages of the SIMSTAR computer system.

Through the countless hours of working on this thesis project, the number one difficulty was keeping the SIMSTAR computer systems alive. Without the dedication of Don Smith and Bob Ewing keeping the SIMSTARs operating, mainly for my benefit, this project would have been impossible in the allotted time. I express my thanks to both of them. I would also like to thank my thesis advisor, Lt. Col. Zdzislaw Lewantowicz, for his unyielding help and belief in my ability to accomplish this task.



Accession For	
NTIS SPA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Avail and/or	
Dist	

A-1

Table of Contents

	Page
Preface	ii
List of Figures	v
List of Tables	vii
Notation	viii
Abstract	xv
I. Introduction	1
Overview	1
Background	1
Problem Statement	2
Purpose of the Study	3
Summary of Current Literature	3
Presentation	6
II. Simulation Specifications	7
Overview	7
Simulation Objectives	7
Equipment and Materials	9
Approach	13
Scope and Limitations	14
III. Aircraft and Flight Control System Description ..	18
Overview	18
Aircraft Description	18
Digital Flight Control System Description ..	19
IV. Simulation Modelling	32
Overview	32
Axes and Sign Conventions	32
Equations of Motion	34
Atmospheric Model	41
Aerodynamic Data Format	44
Aerodynamic Interpolation Equations	49
Aerodynamic Coefficient Equations	51
Backward Difference Equation	54
V. Simulation Design	56
Overview	56
Interpolation and Look-up Routine	57

Table of Contents Continued

	Dynamics Model Routine	59
	Control System Design Routine	61
	Cockpit Display Routine	62
	Implementation of Routines	63
VI.	Simulator Testing and Verification	65
	Overview	65
	Validating Aircraft Model	65
	Simulator Performance	103
VII.	Conclusions and Recommendations	104
	Appendix A: SIMTACS User's Guide	107
	Appendix B: Intermachine Cable Hookups	140
	Appendix C: Pilot Control Schematics	144
	Appendix D: Cockpit Graphics Display	147
	Appendix E: Simulator Source Code	152
	Bibliography	155
	Vita	157

List of Figures

Figure	Page
2.1. General Hardware Arrangement	12
3.1. DFCS Longitudinal Control Block Diagram	21
3.2. DFCS Gain Functions for Longitudinal Control ..	22
3.3. DFCS Lateral-Directional Control Block Diagram	23
3.4. DFCS Gain Functions for Lateral-Directional Control	24
3.5. DFCS Control Surface Mixer	25
3.6. Simplified Longitudinal DFCS Block Diagram	27
3.7. Simplified Lateral-Directional DFCS Block Diagram	28
3.8. Pitch Command Gradient	30
3.9. Roll Command Gradient	30
3.10. Rudder Command Gradient	31
4.1. F-16 Sign Conventions	33
4.2. F-16 Axes Systems	34
4.3. Equations of Motion Simulation Flow Diagram ...	40
5.1. Simulator System Block Diagram	56
5.2. Simulator Flow Diagram displaying routine residence	58
5.3. Oscilloscope Cockpit Display	63
6.1. No Input - Trimmed Condition, Coupled Responses	66-69
6.2. One-g Pitch Stick Input, Coupled Responses	71-74
6.3. 45 deg/sec Roll Stick Input, Coupled Responses	76-79
6.4. One deg Rudder Pedal Input, Coupled Responses	81-84

6.5.	No Input - Trimmed Condition, Uncoupled Responses	86-89
6.6.	One-g Pitch Stick Input, Uncoupled Responses	90-93
6.7.	5 deg/sec Roll Stick Input, Uncoupled Responses	94-97
6.8.	One deg Rudder Pedal Input, Uncoupled Responses	98-101
B.1.	SIMSTAR Back Panel	140
C.1.	Hand Control Electrical Schematic	145
C.2.	Hand Control Mechanical Schematic	146

List of Tables

Table		Page
I.	Gains for Simplified DFCS	26
II.	Altitude-Density Relation based on NACA No. 218	42
III.	Mass properties and initial conditions used in time response plots	70

NOTATION

<u>SYMBOL</u>	<u>DEFINITION</u>	<u>UNITS</u>
a	sonic velocity	ft/sec
ADOT	Change in angle of attack	deg/sec
AFIT	Air Force Institute of Technology	
AOA	Angle of attack	deg
AW	Aircraft weight	lb
A _y	Lateral acceleration at the accelerometers	g's
A _{ycg}	Lateral acceleration through the center of gravity	g's
b	Wing span (30.0 ft)	ft
$\bar{c}$	Wing reference chord (11.32 ft)	ft
CD	Drag coefficient with control surfaces in nominal position (F-16)	
CD _{TOTAL}	Total Nondimensional Drag	
CG	Center of gravity given as ratio of the difference of the aerodynamic center and center of gravity to the wing reference chord (CG=.35)	
CL	Lift coefficient with control surfaces in nominal position (F-16)	
CL _q	Lift to pitch rate derivative	per rad
CL _{$\dot{\alpha}$}	Lift to time rate of change in angle of attack derivative	per rad
CL _{δFA}	Lift to flaperon deflection derivative	per deg
CL _{TOTAL}	Total Nondimensional Lift	
Cl	Rolling Moment coefficient with control surfaces in nominal position (F-16)	

Cl_p	Stability axis rolling moment to roll rate derivative	per rad
Cl_r	Stability axis rolling moment to yaw rate derivative	per rad
$Cl_{\delta FA}$	Rolling moment to differential flaperon deflection derivative	per deg
$Cl_{\delta HA}$	Rolling moment to differential horizontal tail deflection derivative	per deg
$Cl_{\delta r}$	Rolling moment to rudder deflection	per deg
Cl_{TOTAL}	Total Nondimensional Rolling Moment	
CM	Pitching Moment coefficient with control surfaces in nominal position (F-16)	
CM_q	Pitching moment to pitch rate derivative	per rad
$CM_{\dot{\alpha}}$	Pitching moment to time rate of change in angle of attack derivative	per rad
CM_{TOTAL}	Total Nondimensional Pitching Moment	
CN	Yawing Moment coefficient with control surfaces in nominal position (F-16)	
CN_p	Stability axis yawing moment to stability roll rate derivative	per rad
CN_r	Stability axis yawing moment to stability yaw rate derivative	per rad
$CN_{\delta FA}$	Yawing moment to differential flaperon deflection derivative	per deg
$CN_{\delta HA}$	Yawing moment to differential horizontal tail deflection derivative	per deg
$CN_{\delta r}$	Yawing moment to rudder deflection	per deg
CN_{TOTAL}	Total Nondimensional Yawing Moment	
CY	Side Force coefficient with control surfaces in nominal position (F-16)	

CY_p	Side force to stability roll rate derivative	per rad
CY_r	Side force to stability yaw rate derivative	per rad
$CY_{\delta FA}$	Side force to differential flaperon deflection derivative	per deg
$CY_{\delta HA}$	Side force to differential horizontal tail deflection derivative	per deg
$CY_{\delta r}$	Side force to rudder deflection derivative	per deg
CY_{TOTAL}	Total Nondimensional Side Force	
DAP	Digital Arithmetic Processor- The SIMSTAR has one for executing the digital region	
DFCS	Digital flight control system	
DH	Symmetrical horizontal tail deflection (also δH)	deg
DLEF	Leading-edge flap deflection (also δLEF)	deg
Fa	Aileron pilot force stick input	lb
Fe	Elevator pilot force stick input	lb
Fr	Rudder pedal pilot input	lb
g	Acceleration of gravity	ft/sec/sec
h	Altitude	ft
H1	Altitude	ft
H2	Altitude	ft
I_e	Aircraft engine moment of inertia	slug-ft ²
I_x	Aircraft mass moment of inertia about the x body axis	slug-ft ²
I_{xy}	Aircraft mass product of inertia about the xy body plane	slug-ft ²
I_{xz}	Aircraft mass product of inertia about the xz body plane	slug-ft ²

I_y	Aircraft mass moment of inertia about the y body axis	slug-ft ²
I_{yz}	Aircraft mass product of inertia about the yz body plane	slug-ft ²
I_z	Aircraft mass moment of inertia about the z body axis	slug-ft ²
MACH	Mach number	
m	Aircraft mass	
N_z	Normal acceleration at the accelerometers	g's
N_{zcg}	Normal acceleration through the center of gravity	g's
OHT	Original horizontal tail	
P_o	Sea level static pressure (2116.216)	lb/ft ²
P_s	Stability axis roll rate	rad/sec
P_S	Static pressure	lb/ft ²
q_s, q	Stability axis and body axis pitch rate	rad/sec
q_w	Wind axis pitch rate	rad/sec
$\bar{q}$	Dynamic pressure	lb/ft ²
q_c	Total pressure (impact)	lb/ft ²
r	Body axis yaw rate	rad/sec
r_s	Stability axis yaw rate	rad/sec
r_w	Wind axis yaw rate	rad/sec
RL_q	Flexible to rigid lift ratio for pitch rate and time rate of change in angle of attack	
$RL_{\delta H}$	Flexible to rigid lift ratio for horizontal tail deflection	
RL_p	Flexible to rigid ratio for the rolling moment to roll rate derivative	

Rl_{β}	Flexible to rigid rolling moment ratio for addition of the vertical tail	
$Rl_{\delta a}$	Flexible to rigid rolling moment ratio for defelction of total roll control for a gearing of 0.25 degree δHA for each 1 degree of δFA	
$Rl_{\delta FA}$	Flexible to rigid rolling moment ratio for deflection of differential flaperons	
$Rl_{\delta r}$	Flexible to rigid rolling moment ratio for rudder deflection	
RM_q	Flexible to rigid pitching moment ratio for pitch rate and time rate of change in angle of attack	
$RM_{\delta H}$	Flexible to rigid pitching moment ratio for addition and deflection of the horizontal tail	
RN_{rVT}	Flexible to rigid yawing moment ratio of the vertical tail for the yaw rate derivative	
RN_{β}	Flexible to rigid yawing moment ratio for the addition of the vertical tail	
$RN_{\delta r}$	Flexible to rigid yawing moment ratio for rudder deflection	
RY_{rVT}	Flexible to rigid side force ratio of the vertical tail for the yaw rate derivative	
RY_{β}	Flexible to rigid side force ratio for the addition of the vertical tail	
$RY_{\delta r}$	Flexible to rigid side force ratio for rudder deflection	
PSP	Parallel Simulation Processors - SIMSTARs can have up to two for running the analog portion of the machines.	
S	Wing area (300 sq ft)	ft ²
TH	Thrust	lb force
T	Sampling period	sec

V_T	Wind X-axis velocity	ft/sec
α	Angle of attack	rad
$\dot{\alpha}$	Time rate of change in angle of attack	rad/sec
β	Sideslip angle	rad
δF	Symmetrical Flaperon deflection	deg
δFA	Differential Flaperon deflection	deg
δFL	Left Flaperon surface deflection	deg
δFR	Right Flaperon surface deflection	deg
δH	Symmetrical horizontal tail deflection (also DH)	deg
δHA	Differential horizontal tail deflection	deg
δHL	Left horizontal tail surface deflection	deg
δHR	Right horizontal tail surface deflection	deg
δLEF	Leading-edge flap deflection (also DLEF)	deg
δr	Rudder deflection	deg
ΔCL_{FLEX}	Incremental lift coefficient due to flexibilizing the wing/body	
ΔCL_{FT}	Incremental lift coefficient due to full scale effects	
ΔCM_{FLEX}	Incremental pitching moment coefficient due to flexibilizing the wing/body	
ΔCM_{FT}	Incremental pitching moment coefficient due to full scale effects	
θ	Body axis pitch Euler angle	rad
ϕ	Body axis roll Euler angle	rad
ψ	Body axis yaw Euler angle	rad

ρ	Air density	slug/ft ³
ρ_0	Air density at sea level	slug/ft ³

Abstract

The focus of this project is the development of a real-time flight simulator for man-in-the-loop evaluation of flight control system designs. The F-16A is the simulated aircraft and a simplified F-16 digital flight control system is used. The simulation models nonlinear aerodynamic data over most of the flight envelope and nonlinear inertially cross-coupled aircraft dynamics.

The simulator is implemented on two Electronic Associates, Inc. SIMSTAR hybrid computer systems and an initial cockpit is designed for human pilot interaction. The simulated aircraft dynamic behavior is validated using AFTI/F-16 closed-loop time responses provided by the Air Force Flight Dynamics Laboratory.

The real-time, man-in-the-loop simulator is designed using the SIMSTAR computer systems. Currently, the simulator performance is limited in the speed of response. This temporary lack of speed is the result of the computational limitations of the current digital processors of the SIMSTAR computer systems. These limitations can be removed with hardware upgrades.

F-16 SIMULATOR FOR MAN-IN-THE-LOOP
TESTING OF AIRCRAFT CONTROL SYSTEMS
(SIMTACS)

I. Introduction

Overview

Aircraft performance simulations are primarily used for two purposes. They are engineering design and evaluation simulation, and training simulation (7:84). This thesis focuses on using simulation as a design and evaluation tool for aircraft flight control laws. Particular emphasis is placed on evaluation of these control laws by a human pilot.

Background

Every year, several theses at the Air Force Institute of Technology (AFIT) develop aircraft flight control designs using different analytic approaches. In addition, students in the Guidance and Control program specializing in the Flight Control application sequence learn these various approaches by actually designing simplified control systems. Presently, only Computer-Aided-Design (CAD) packages are used by students to test their theses or class project designs.

An invaluable additional dimension to the education of the student and to the improvement of his design evaluation is a research tool that "real-world" design engineers use in developing and testing aircraft control systems. Design engineers use simulators as an essential part of the design

process (6:40,510; 12:v). This is necessitated by the complexity of the systems and of the designs. The aircraft flight control system designs are becoming so complex and computationally burdensome that to study the design without simulation is risking both the pilot's life and the technology advancement built into an aircraft prototype (6:40-41,510).

A real-time simulator with man-in-the-loop capability provides students, faculty and engineers a research tool which simulates, with high fidelity, a complex real-world aircraft and allows their control designs to interact dynamically with human pilot inputs. Normally, a dynamic model of a human pilot can be incorporated into a CAD design by modelling the pilot's control using a lead-lag transfer function approximation. However, this is a very crude approximation which cannot replace the actual human-in-loop. Using the human pilot in-the-loop in designing and evaluating flight control systems with the aid of a man-in-the-loop simulator more accurately assesses the design and allows handling quality criteria to be considered directly. Thus the human pilot interaction is an essential iteration during the design of a flight control system.

Problem Statement

The objective of this thesis is to design a real-time simulator that provides the user with a man-in-the-loop testing capability of aircraft flight control system designs. Thorough

user instructions for the simulator must be provided so that he can easily implement an aircraft control system.

Purpose of the Study

The design of a simulator as an interactive tool will aid in the design process of aircraft control systems. The user instructions should be sufficiently easy such that the usefulness of the tool outweighs the extra effort made by the user to use the tool. It is hoped that this project will spark continuing work in the area of man-in-the-loop testing of aircraft control systems at the School of Engineering, Air Force Institute of Technology.

Summary of Current Literature

Current literature for this thesis generally falls into three areas. These areas are aircraft control systems, man-in-the-loop testing, and simulation.

D'Azzo and Houpis state that "The successful operation of space vehicles and the space shuttle depend on the proper functioning of the large number of control systems used in such ventures"(4:1). This view, which is prevailing in engineering, demonstrates the importance of control systems in the continuing advancement of technology.

While every aircraft is a nonlinear analog device, the control system most frequently being implemented on present aircraft is digital and based on linear system designs. As noted by D'Azzo and Houpis, "Recent advances in digital

computers and microprocessors have made their use very attractive as components in control systems"(4:639) such as those in aircraft.

Numerous methods are used to design aircraft control systems, but three methods are generally studied and improved upon through theses each year by graduate students at the Air Force Institute of Technology. The first method, based on Porter's work, is described by D'Azzo and Houpis as "a design for output feedback systems" based on a "high gain proportional plus integral (PI) control law"(5:773). The other method, based on Horowitz' work, is described by Houpis as "a frequency response design method applied to the design of a control system with an uncertain plant"(8:1.13). The third method as explained by Maybeck, is the design of optimal stochastic controllers which take into account dynamic disturbances from the environment and imperfect measurements from sensors (10:2-9).

After designing an aircraft control system, the ultimate test of the system is an evaluation by a pilot flying in the aircraft with the control system implemented. This test considers the true dynamics of the particular pilot and is known as "putting the man in the loop". The importance of man-in-the-loop testing is noted by Etkin:

The care exercised in considering the human element in the closed-loop system made up of pilot and aircraft can determine the success or failure of a given aircraft design to complete its mission in a safe and efficient manner (6:491).

Grodsky refers to man-in-the-loop testing as meaning to gather "Qualitative data on such factors as handling qualities of aerospace systems, which is a necessary component of both configuration and control system development"(5:90). One set of handling quality criteria called the Cooper-Harper pilot rating scale gives pilots guidelines for evaluating a system (13:6.85-6.89). Roskam describes the scale as representing an "attempt to relate pilot comments about the ease or difficulty with which airplanes can be controlled in certain flight situations to a numerical rating"(13:6.88). With an aircraft control system design and a pilot rating system at hand, a simulation will determine the success or failure of the design.

The importance of simulation is summed up by Rodgers and Shapiro. They state, "The story of man's progress in science and technology is actually the story of his success in the use of analogy and his progress in simulation"(12:v).

The question now arises, what is the purpose of simulating flight control systems of aircraft? According to Blech and Arpasi,

Hardware-in-the-loop and man in the loop simulations, which require real-time performance, provide many cost- and time saving benefits. This is particularly true for jet aircraft systems, where ground and flight testing are costly. For example, piloted simulators are used as a convenient, low-cost method to evaluate system design changes and their effect on pilot workload (1:43).

Some current examples of elaborate simulators include the Real-Time MultiProcessor Simulator (RTMPS) project at the NASA

Lewis Research Center, Cleveland (1:43) and the Large Amplitude Multi-mode Aerospace Research Simulator (LAMARS) at the Flight Dynamics Laboratory, Wright-Patterson AFB (15).

The three areas summarized form the basis for the research of this thesis. It is important to understand man-in-the-loop testing and how it pertains to the simulation of aircraft control systems. It is obvious that the design and simulation of aircraft control systems is important to the development of better performing aircraft.

Presentation

This paper is presented in seven chapters. Chapter II explains the simulation objectives, hardware and limitations. Chapter III describes the aircraft and flight control system used. Chapter IV presents the modelling theory used in the simulations in order to model the real-world aircraft and its flight control system. Chapter V describes the simulator design by breaking it up into its separate functions and explains how this design achieves the original objectives. Chapter VI explains how the simulation is tested and validated. And Chapter VII concludes with recommendations for future research and design in this area.

II. Simulation Specifications

Overview

Whenever a system is developed the designer must have specific objectives in mind. Without these objectives, the designer is unable to determine whether or not his system reaches the level of confidence necessary to solve the intended problem. For this thesis effort, the original objective to design a fully operational man-in-the-loop simulator is optimistic, but necessary. The hardware limitations of this ambitious project could not be uncovered without this full system design approach.

Simulation Objectives

This thesis concentrates on developing a real-time fixed-base simulator that provides a user with a tool for testing an aircraft control system design. With this in mind, the objective is to provide an accurate but easy to use man-in-the-loop simulator.

When using a CAD package to test an aircraft control system design, a set of linearized perturbation state equations at a specified nominal flight condition is developed to approximate the aircraft dynamics. These equations consist of stability derivatives developed by evaluating partial derivatives of nonlinear force and moment equations at the operating point in the flight regime. The particular set of stability derivatives are valid only for the given equilibrium

condition. The simulator must not restrict the user to an approximation and a point flight condition. Instead, nonlinear aircraft aerodynamic data from actual flight test data extending over much of the flight envelope should be used in nonlinear dynamic equations of motion. This provides the user flexibility to test his design throughout the flight envelope.

Even though the CAD package is adequate for the design process it is not adequate for testing the design. This thesis simulation must accurately describe the environment of the aircraft and the aircraft's response to the environment through a dynamics model incorporating the nonlinear equations of motion.

Another objective of the real-time simulator is to allow a human pilot to interact with the simulation. This entails developing a cockpit environment complete with nonlinear stick control and cockpit instrument display. The cockpit instruments must update pertinent aircraft output, driving display variables in real-time.

The overall simulator must execute with the speed necessary to provide a near zero closed-loop (from pilot input to aircraft cockpit instrument update) time lag. If there is time lag, it must be sufficiently small so that it is not noticeable to the pilot. It is important for the pilot to believe that he is flying the actual aircraft in order to accurately achieve man-in-the-loop testing goals.

A final objective of this thesis is to design user-friendly program code and step by step instructions for the user to implement and test his own design.

The software of the simulator is divided into separate routines, each with its own function. This is necessary to facilitate future upgrades of the simulator. It also provides a cleaner, versatile working environment for the user. With this separation of the routine functions, the user should find it easy to modify the routines for his specific simulation.

The above objectives are the basis for this thesis project. The first step in achieving these objectives is to bring together the necessary hardware and materials.

Equipment and Materials

The Air Force Institute of Technology has two Electronic Associates, Inc. (EAI) SIMSTAR computer systems. They are referred to by their serial numbers, #114 and #115. The majority of the software for this thesis is written on the two SIMSTARs. The SIMSTAR computer system is a digital and analog (hybrid) computer specifically designed for real-time simulation. Unlike previous generation analog computers, the SIMSTAR requires no patchcords because all analog components are connected through an electronic switch matrix. These connections are made through the software in the compilation process. The scaling of variables is still necessary even though the SIMSTAR performs the actual scaling calculations.

The user must provide the variable ranges and the SIMSTAR translator/compiler performs the actual scaling.

The SIMSTAR consists of a Digital Arithmetic Processor (DAP) and up to two Parallel Simulation Processors (PSP). The AFIT SIMSTARs each have one PSP. Basic setup and control of the user program along with on-line serial computations are provided by the DAP. The PSP performs high-speed parallel (analog) computations as dictated by the user program.

A continuous system simulation language (CSSL) called the Advanced Continuous Simulation Language (ACSL) is used as the basic language for model definition. A basic program consisting of a digital and a parallel (analog) region is written in a structure separating the digital from the parallel with each part essentially written in the same language. A digital translator (D-TRAN) compiles the digital region into a totally discrete FORTRAN model for processing in the DAP, and a parallel translator (P-TRAN) compiles the parallel region. The parallel region is compiled into two programs. One program is a setup routine used by the DAP and the other program is a set of connection statements to the various mathematical computing blocks (MCB) used by the PSP. The MCBs are used to carry out the parallel tasks specified by the user including integration, differentiation, summation and multiplication. A FORTRAN program can be written on the DAP alone or in conjunction with the user's hybrid model. For more specific information on

programming the SIMSTAR, the reader is referred to the P-TRAN (17) and D-TRAN (16) Manuals.

A factor in determining the utilization of the SIMSTARs to their full capability is the computational capacity of the analog portions. SIMSTAR #114 is a half matrix machine which has been striped of MCBs in order to fill the full matrix machine, SIMSTAR #115. The full matrix refers to the fact that the machine can be filled up to its maximum capacity in MCBs, whereas, a half matrix refers to the fact that the machine's full capacity in MCBs is half that of a full matrix machine. The current limited capacity of the analog portions presents an added challenge in the programming, however this limitation is expected to disappear with the planned SIMSTAR expansion.

An Evans and Sutherland PS340 graphics terminal is connected to SIMSTAR #115 through an RS-232 port and can display several cockpit instruments for the pilot. The graphics display was eventually replaced for the purpose of this thesis with a simple oscilloscope display of pitch and roll. The reason for this is explained later in this chapter. The graphics display is still mentioned in this thesis because the author wishes to document some of the work that he has accomplished with the Evans and Sutherland terminal so that future work in this area will be completed (see Appendix D). Some other hardware pieces necessary consist of a hand control (force stick) for aileron and elevator control and an eight-channel strip chart recorder for recording time responses. The

hand control requires a signal generator for a power source. The hand control is connected directly to the analog ports of SIMSTAR #115. Also SIMSTAR #114 is connected directly to SIMSTAR #115 through their analog input and output ports. Variables are instantaneously transferred between the two analog regions of the machines. This is further explained later. Figure 2.1 shows a general arrangement of the hardware. For more specific details on cabling between the machines refer to Appendix B.

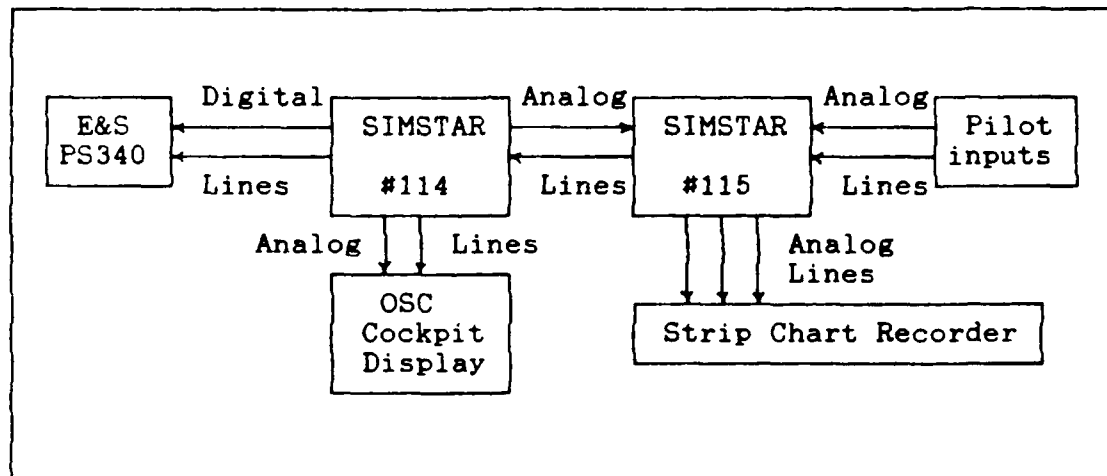


Figure 2.1. General Hardware Arrangement

Also necessary for this project, is access to all of the aerodynamic data for the F-16A (OHT) and a digital flight control system for the F-16 with closed-loop time responses. These are further explained in Chapter III. The F-16A aircraft is chosen because its well-defined specifications and aerodynamic data base are easily accessible.

Approach

The general approach taken for this thesis revolves around using the two SIMSTARs to the maximum advantage and making every effort to reduce the computational speed necessary for real-time output and satisfy the real-time requirement.

The tasks to be programmed on the SIMSTARs include looking-up and interpolating aerodynamic data, computing aircraft dynamics, executing the flight control system and updating the graphics display. The task of updating the PS340 graphics display is not completed in this thesis due to limitations in the SIMSTARs current configuration. This is further explained in the next section.

The task of interpolating and looking-up the aerodynamic data for a specific flight condition is considered to be the computationally most burdensome. SIMSTAR #114's DAP is used entirely for this task. The analog region could not be used for this task because all MCBs, except a few used for passing the variables from the analog region of SIMSTAR #115 to the digital region of SIMSTAR #114, were moved to SIMSTAR #115 for the dynamics model.

The task of computing the aircraft dynamics is done in the analog region of SIMSTAR #115. Because the analog region computes everything instantaneously (no time delay), the only limitation is the number of MCBs available to model the full inertially and aerodynamically coupled aircraft. SIMSTAR #114 was stripped of MCBs to fill SIMSTAR #115 for this purpose.

One of the tasks of SIMSTAR #115's digital region is to execute the aircraft control system computations. This code is written entirely in FORTRAN and easily readable. The user unfamiliar with the detailed operation of the SIMSTAR computer system should be able to implement his or her own flight control system with relative ease if the user already knows FORTRAN. The other task that must be executed along with the control system is the sending of selected output variables to the graphics display.

Each software routine is validated using test data then joined together. Full scale system testing is performed to validate that the simulation closed-loop aircraft responses match those responses of the aircraft from other published sources. The external pilot environment is developed once the major software routines are properly working together. The pilot cockpit incorporates the human pilot inputs and the display of pitch and roll on an oscilloscope. Throughout the development, user-friendliness is stressed in the design.

Scope and Limitations

Limitations on the thesis work are attributable to the capability of the hardware and the aircraft data used. When this thesis started, the full capability of the SIMSTARs was unknown. Since then, many problems and limitations of the systems have been overcome, but at the completion of this thesis, a few still exist. The major limitation not overcome is executing the digital interpolation routines on SIMSTAR #114

with the necessary speed. The speed necessary for one cycle through all of the code, including the aircraft control system code in SIMSTAR #115, is around 15 msec. The best speed possible for the code in SIMSTAR #114 alone is currently about 125 msec whereas the aircraft control system code in SIMSTAR #115 can run in about 20 msec. This limitation is due to the processor (DAP) speed. Two Gould 32/27 processors, which are slowest of this line of Gould processors, serve as the DAPs in both SIMSTARS. There are several possible solutions to this hardware limitation, all of which require the replacing of the current configuration with new hardware. This is further discussed in the conclusions and recommendations in Chapter VII.

Another limitation not overcome on SIMSTAR #115 includes the inability of using the FORTRAN "write" statement in the DERIVATIVE region. The DERIVATIVE region is executed every cycle time. This is the region where the aircraft control system resides in SIMSTAR #115. This limitation greatly hampers the task of sending updates to the graphics terminal and contributes to its incompleteness in this thesis. Due to the manner in which the SIMSTARS partition their jobs, the FORTRAN "write" statement can only be used in the INITIAL, DYNAMIC and TERMINAL regions. The INITIAL and TERMINAL regions are digital regions executed only once every run either at the beginning of the run or the end of the run as is evident in its name. The DYNAMIC region is executed only if the DAP finds it has extra

time between the user-defined interrupts and the DERIVATIVE region execution. By placing the "write" statement in the DYNAMIC region, the "write" statement is updated only when the computer believes it has enough time to execute it. As a result, the computer sends updates to the graphics terminal about seven times every ten seconds. This speed is not acceptable for man-in-the-loop operation. This limitation was not discovered until late in the thesis development.

Other limitations in the aerodynamic data base for the full flight envelope were known from the outset of the thesis. These limitations include many standard simplifications customarily used in defining the scope of the simulation, therefore they are not considered problems for this thesis. The aircraft model accounts for various aero-elastic effects due to angular velocities, control surface positions and sideslip angle.

The model does not yet account for ground effects, speedbrake and landing gear effects, engine gyroscopic effects and variable mass for the aircraft. The model can be easily expanded to include these additional flight conditions. The earth is assumed to be an inertial frame and the air surrounding the aircraft is considered a homogeneous mixture at rest with respect to the earth. The aircraft is also assumed symmetrical with reference to a vertical plane aligned with the longitudinal axis of the aircraft. The flight envelope limit

of the model is dictated by the F-16A (OHT) 'CLEAN' aerodynamic data base described in Chapter IV.

The simulator cockpit is currently limited to an oscilloscope display of pitch and roll. A pilot control stick consisting of a force stick for elevator and aileron control fill out the crude initial cockpit design.

III. Aircraft and Flight Control System Description

Overview

This chapter presents the aircraft and flight control system used in this thesis. The flight control system used is simplified from an F-16 digital flight control system obtained from the F-16 System Program Office (SPO), Wright-Patterson AFB.

Aircraft Description (11:3.1-3.7)

The aircraft is the F-16A with the original horizontal tail (OHT), carrying wingtip AIM-9 missiles. The F-16A is a single-engine, single-seat, multi-role tactical fighter with full air-to-air and air-to-surface combat capabilities. The aircraft is powered by an F100-PW-100 turbofan engine in the 25,000 pound thrust class.

The control surfaces consist of a pair of flaperons, a pair of horizontal stabilizers, a pair of leading-edge flaps, and a rudder. The flaperons, mounted on the trailing edge of the wing, combine the function of flaps and ailerons. The horizontal stabilizers, which are the horizontal tails, provide elevator and aileron functions through symmetrical and differential deflection. The leading-edge flaps are automatically controlled by the flight control system to enhance performance over a wide speed range and the rudder, located on the vertical tail, provides directional stability and control.

This aircraft is chosen because the aerodynamic data package (21) is available from ASD/ENFTC and the F-16 SPO. The aerodynamic data also is in a form that is easier to work with than that of other aircraft.

One concern when choosing an aircraft, is to simulate an aircraft that future AFIT students would be designing aircraft control systems for. It turns out that most thesis students working with aircraft control systems design for the AFTI/F-16 aircraft, a modified F-16A. The external modifications include the addition of twin vertical moveable canards on the forward underside of the engine inlet and a dorsal fairing on the upper fuselage centerline running from the back of the pilot's canopy to midway on the vertical fin root structure. All other external dimensions on the AFTI/F-16 are the same as the F-16A. Internally, the integrated servoactuators for the control surfaces are the same for both aircraft. Therefore, a student can simulate an AFTI/F-16 but would have to disregard any control design utilizing the canards. This is only a current temporary limitation which can be easily overcome.

Digital Flight Control System Description

The digital flight control system (DFCS) is a simplified version of the Block 25 F-16 DFCS Functional Block Diagram (22). The full F-16 DFCS block diagram is divided into the longitudinal and the lateral-directional mode of operation as shown in Figures 3.1 and 3.3. Their respective gain schedules are shown in Figures 3.2 and 3.4 and a control

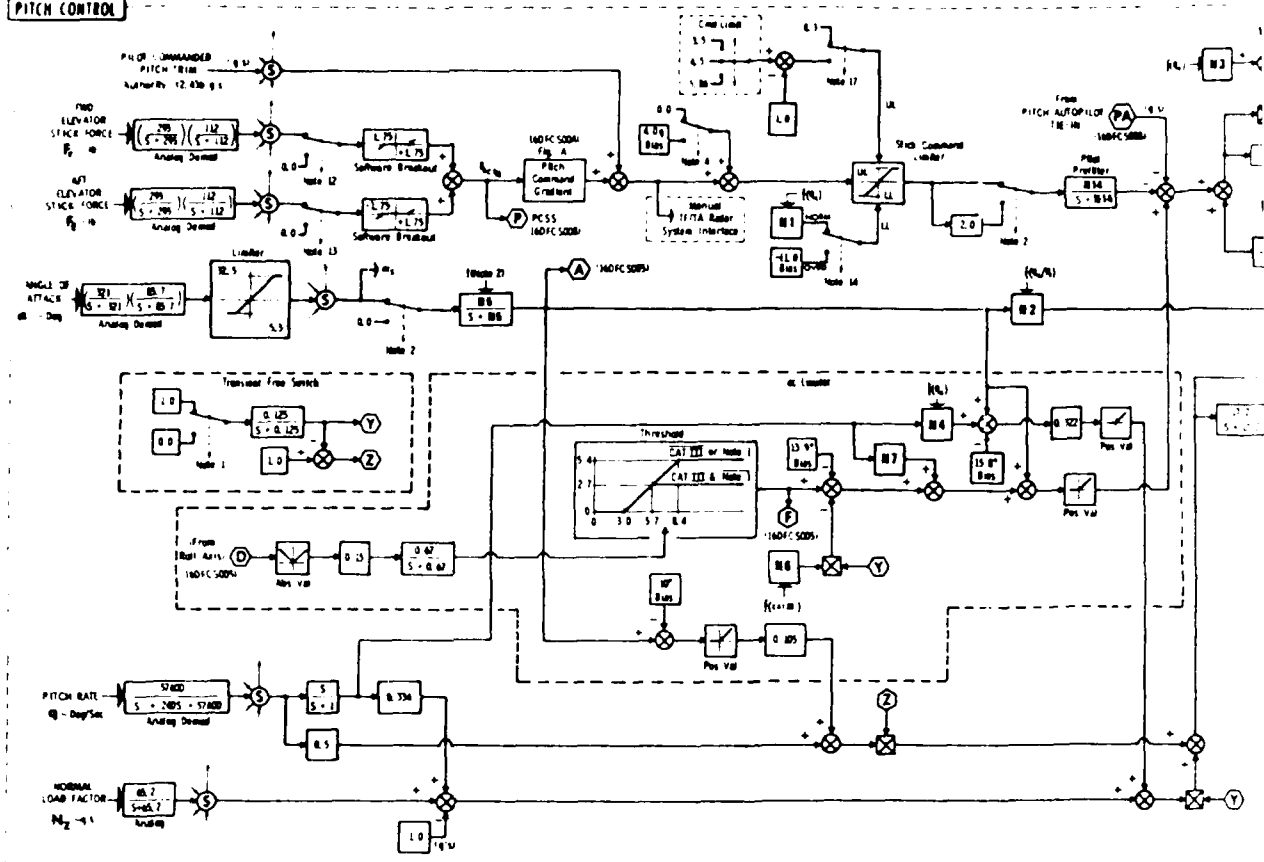
surface mixer is shown in Figure 3.5. The simplification of both the longitudinal and lateral-directional modes are described here and consist mainly of the simplifications in defining the gain values for straight and level flight at a Mach number of 0.9 and altitude of 20,000 feet.

The justification for using a simplified control system is three-fold. First, many of the conditions that the full control system considers such as landing, gunnery, high angle of attack and refueling are not taken into account in the F-16A aerodynamic data used for this thesis, therefore it is unnecessary and wasteful to program the extra code. Second, the design about a single flight condition models directly the concept of testing a student's CAD single point design, demonstrating possible future work using the simulator as a tool. And third, it provides the designer of the simulator a simple controller that is easy to program and debug in the overall scheme of getting a working simulator.

The basic characteristic of the longitudinal DFCS consists of feeding back pitch rate and normal acceleration. These feedback variables operate through a proportional-plus-integral controller in the forward loop providing an elevator deflection. Angle of attack feedback provides static stability of the unstable F-16A aircraft.

The lateral-directional flight control system employs roll rate feedback to the aileron and differential elevator.

PITCH CONTROL



NOTE 1 Switch Activated By ANY of the following Methods:

- LG Handles in DOWN Position
- Standby Cams & Push or Air Ref Switch in EXTING Position & $g_z > 1.00 \text{ g}$
- Standby Cams & Air Ref Switch in DOWN Position & $g_z > 1.00 \text{ g}$

NOTE 2 Switch Activated By ANY of the following Methods:

- Weight on MGC
- Note 18 & Note 14

NOTE 3 Switch Activated By ANY of the following Methods:

- LG Handles in DOWN Position
- Standby Cams & $g_z > 1.00 \text{ g}$

NOTE 5 Switch Activated By ANY of the following Methods:

- LG Handles in DOWN Position
- Push or Air Ref Switch in EXTING Position

NOTE 6 Switch Activated By ANY of the following Methods:

- Weight on MGC
- Standby Cams & $g_z > 1.00 \text{ g}$ OR $g_z > 1.00 \text{ g}$ & $g_z > 1.00 \text{ g}$ OR $g_z > 1.00 \text{ g}$

NOTE 12 Switch Activated When ALL of the following Conditions Are Met:

- Stick Selector Control Switch in ATT Position
- All Stick Possible Switch Depressed

NOTE 4 Switch Activated When ALL of the following Conditions Are Met:

- ATT Arm
- Weather Paddle Switch Push or Air Depressed
- More Than One TT Data Good Discrete Indicators Data Not Good

NOTE 10 Switch Activated When ALL of the following Conditions Are Met:

- Note 1
- Air Ref Switch in DOWN Position
- Note 18
- No Start Warning Cue
- Standby Cams
- No Dual Branch Failure
- Weather Paddle Switch Push or Air Depressed
- All Load 1 or 4 TT Data Good Discrete Indicators Data Not Good

NOTE 11 Switch Activated When ANY of the following Conditions Are Met:

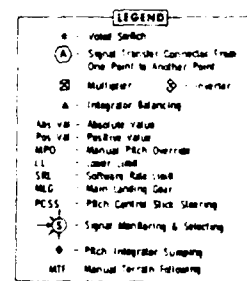
- Note 1
- Air Ref Switch in DOWN Position
- Note 18
- Start Warning Cue
- Standby Cams
- Dual Branch Failure

NOTE 13 Switch Activated When ALL of the following Conditions Are Met:

- Stick Selector Control Switch in ATT Position
- Push Stick Possible Switch Depressed

NOTE 14 Switch Activated By ANY of the following Methods:

- Push MGC Switch in DOWN Position & Note 12
- Air Ref Switch in DOWN Position & Note 12



NOTE 17 Switch Activated When ALL of the following Conditions Are Met:

- Pitch Limiter Switch in the PITCH CMD - ATT Position
- Weather Paddle Switch Push or Air Depressed
- Note 14

NOTE 18 Switch Activated When ALL of the following Conditions Are Met:

- $g_z > 2.0 \text{ g}$
- Weight on MGC

Figure 3.1. DFCS Longitudinal Control Block Diagram

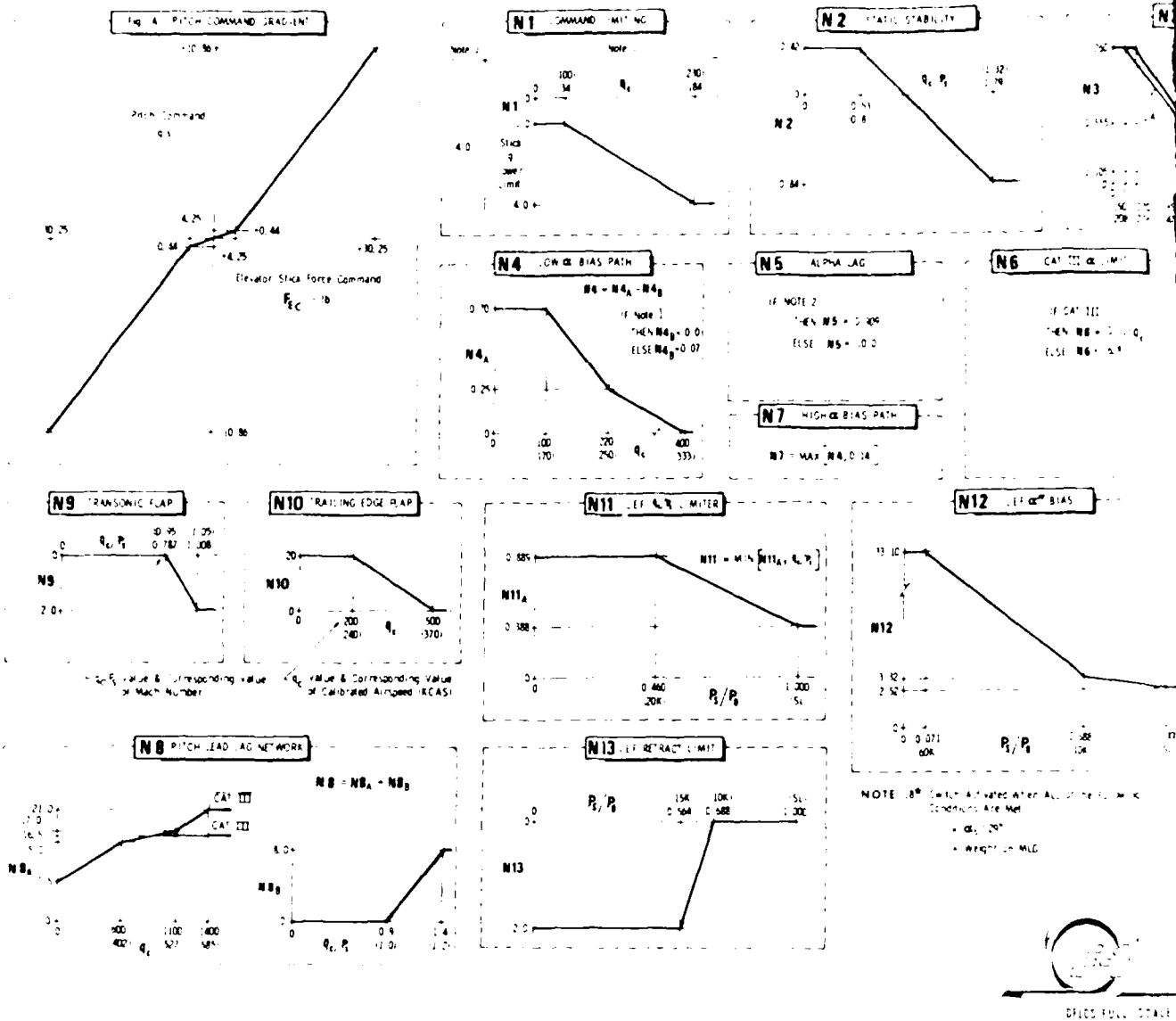
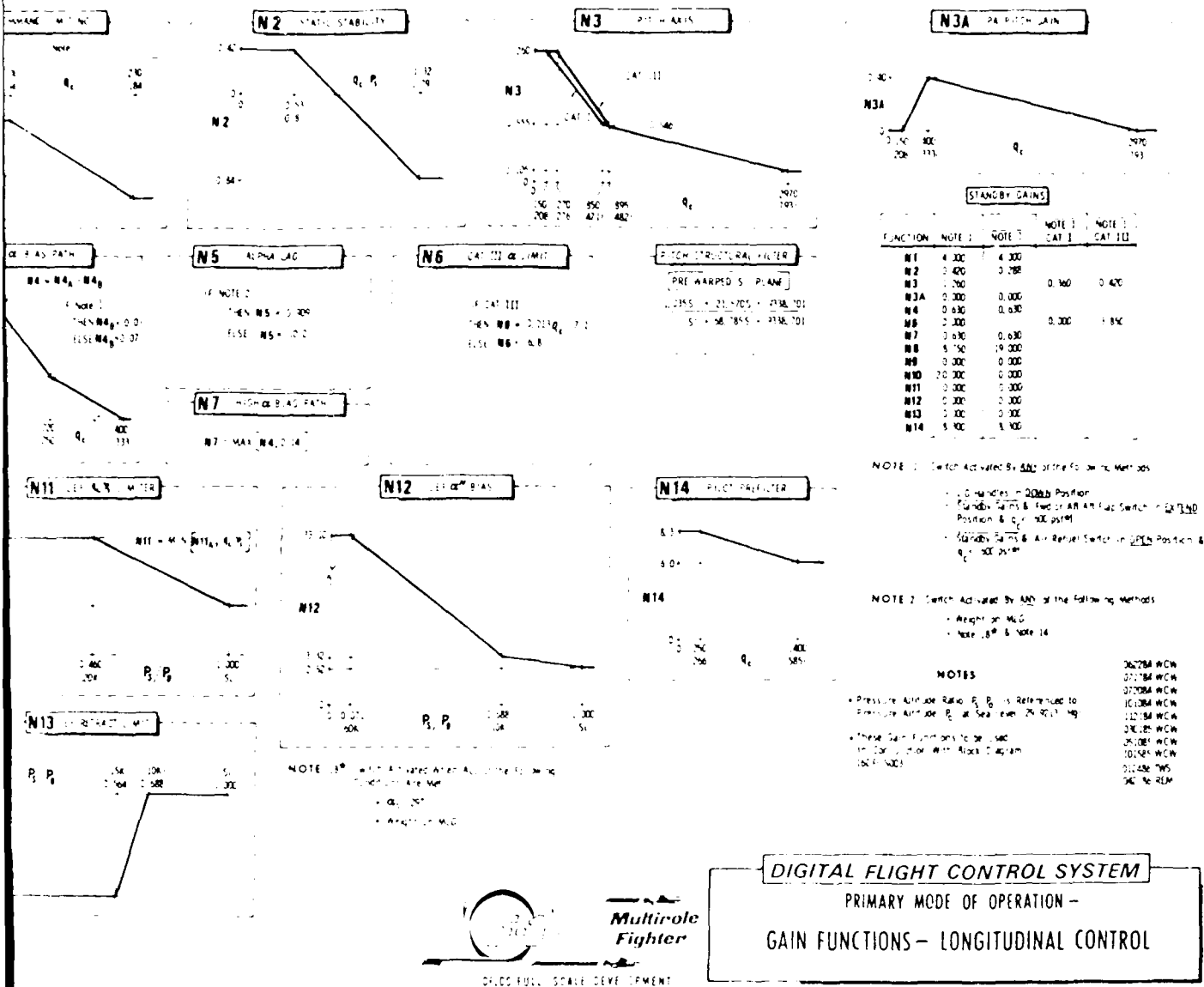


Figure 3.2. DFCS Gain Functions for Longitudinal Control



16DFCS004

Longitudinal Control

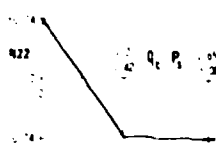


AIN

N21 TURN COORDINATION

N22 TURN COORDINATION

ROLL STRUCTURAL FILTER

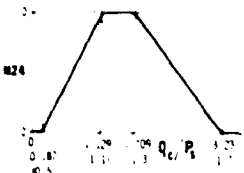


ROLL STRUCTURAL FILTER
PRE-WARPC'S PLANE
4.0005 + 0.0015 + 0.0015
5.0005 + 0.0015 + 0.0015
ROLL RATE NO. 12
2.0005 + 0.0015
5.0005 + 0.0015 + 0.0015

N24 TURN COORDINATION

N25 ROLL NON-NEAR ROLL PREFILTER

YAW STRUCTURAL FILTER



$W = W + \Delta W$
N25 + 1.5
N25 + 1.5
N25 + 1.5

PRE-WARPC'S PLANE
0.0005 + 0.0015 + 0.0015
5.0005 + 0.0015 + 0.0015

N28 ROLL HORIZ. TAIL

N30 YAW 240 AC NETWORK



NOTES

- Pressure Altitude Ratio P/P_0 is Referenced to Pressure Altitude at Sea Level 29.9213 in.
- These Gain Functions to be used in conjunction with Block Diagram 16DFCS005

- 16DFCS005
- 16DFCS006
- 16DFCS007
- 16DFCS008
- 16DFCS009
- 16DFCS010



Multirole Fighter

OF 10 FULL-SCALE DEVELOPMENT

DIGITAL FLIGHT CONTROL SYSTEM

PRIMARY MODE OF OPERATION -

GAIN FUNCTIONS - LATERAL-DIRECTIONAL CONTROL

16DFCS006

for Lateral-Directional

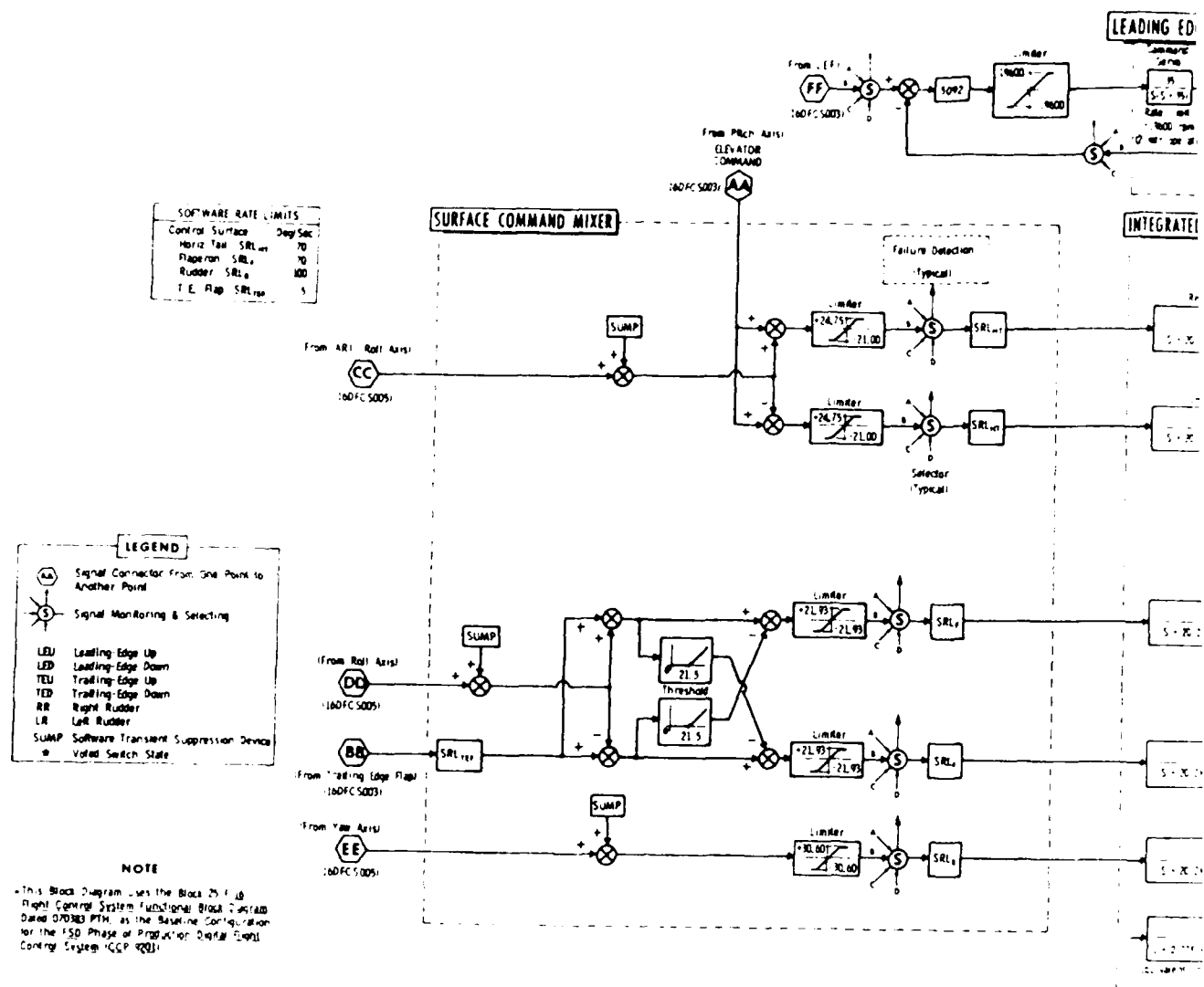
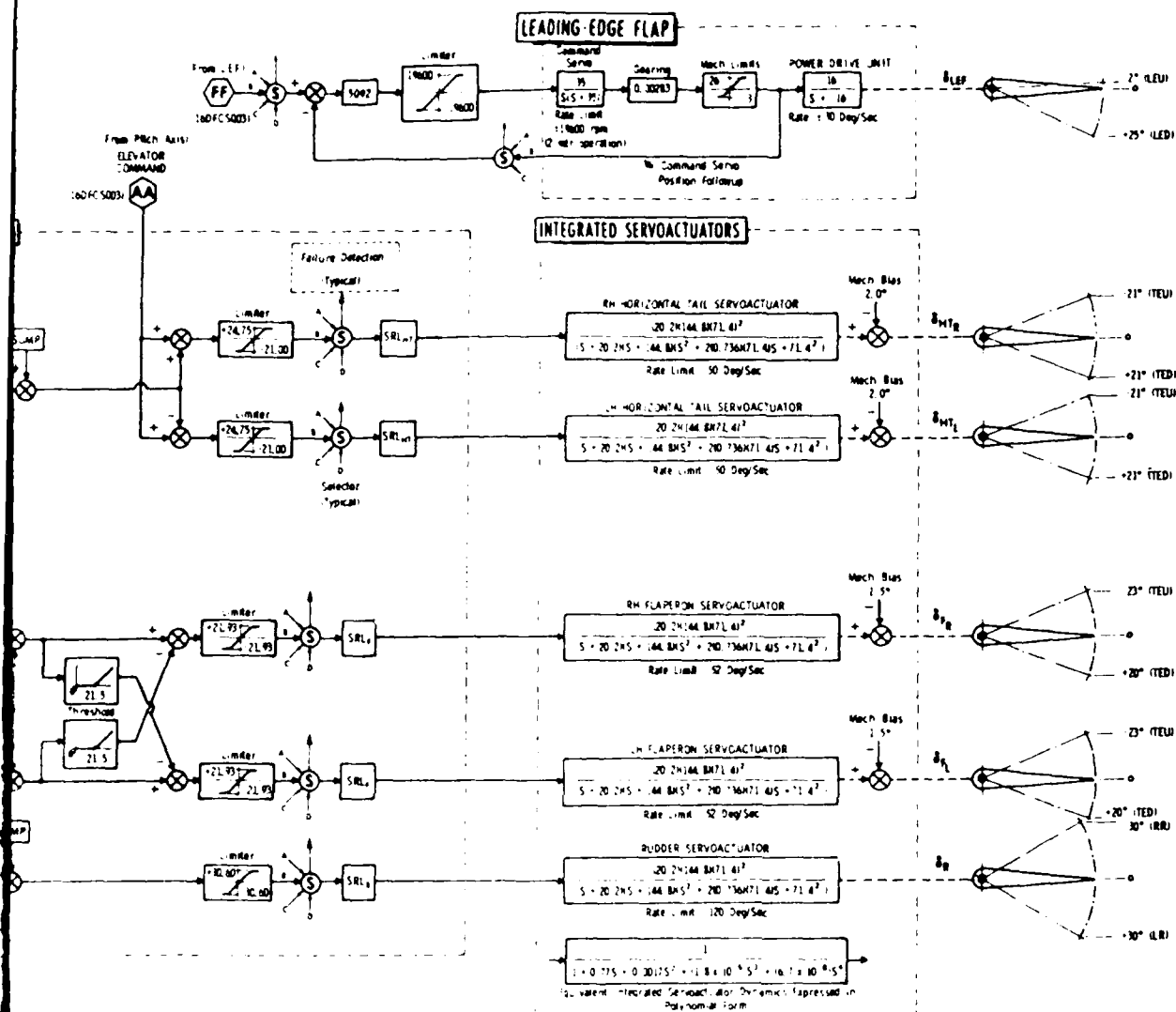


Figure 3.5. DFCS Control Surface Mixer



062204 WCN 062106 TW6
071704 WCN
072004 WCN
101004 WCN
112104 WCN
030105 WCN
051005 WCN
101505 WCN
012806 TW5



DIGITAL FLIGHT CONTROL SYSTEM

PRIMARY MODE OF OPERATION -
CONTROL SURFACE MIXER

16DFC5007

Surface Mixer

A combination of lateral acceleration and yaw rate is fed back to the rudder and an aileron-rudder interconnect is used.

The simplified longitudinal and simplified lateral-directional flight control systems, as shown in Figures 3.6 and 3.7 respectively, have utilized the following simplifications and assumptions.

1. All gains are determined based on the Mach number of 0.9 and altitude of 20,000 feet. These gains are listed in Table I.

Gain Constants	Gain Values
N2	0.38
N3	0.70
N5	10.0
N8	14.4
N14	7.2
N23	0.5
N24	0.67
N25	2.5
N30	20.0

Table I. Gains for Simplified DFCS

2. All mechanical and physical limiters are eliminated. If the aircraft is not overworked with high rates then this simplification holds. Anti-wind up limiters are also assumed unnecessary for this thesis.

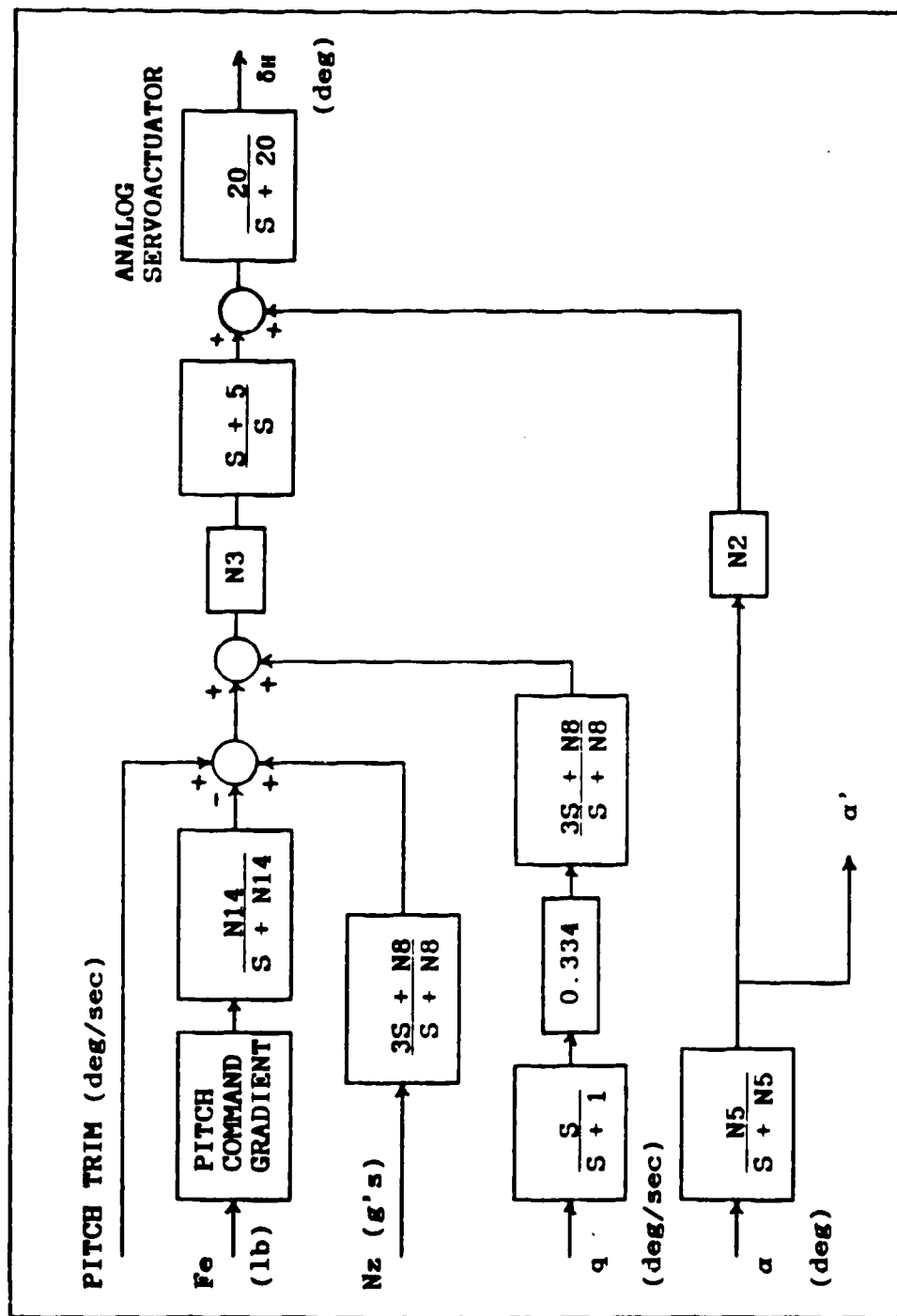


Figure 3.6. Simplified Longitudinal DFCS Block Diagram

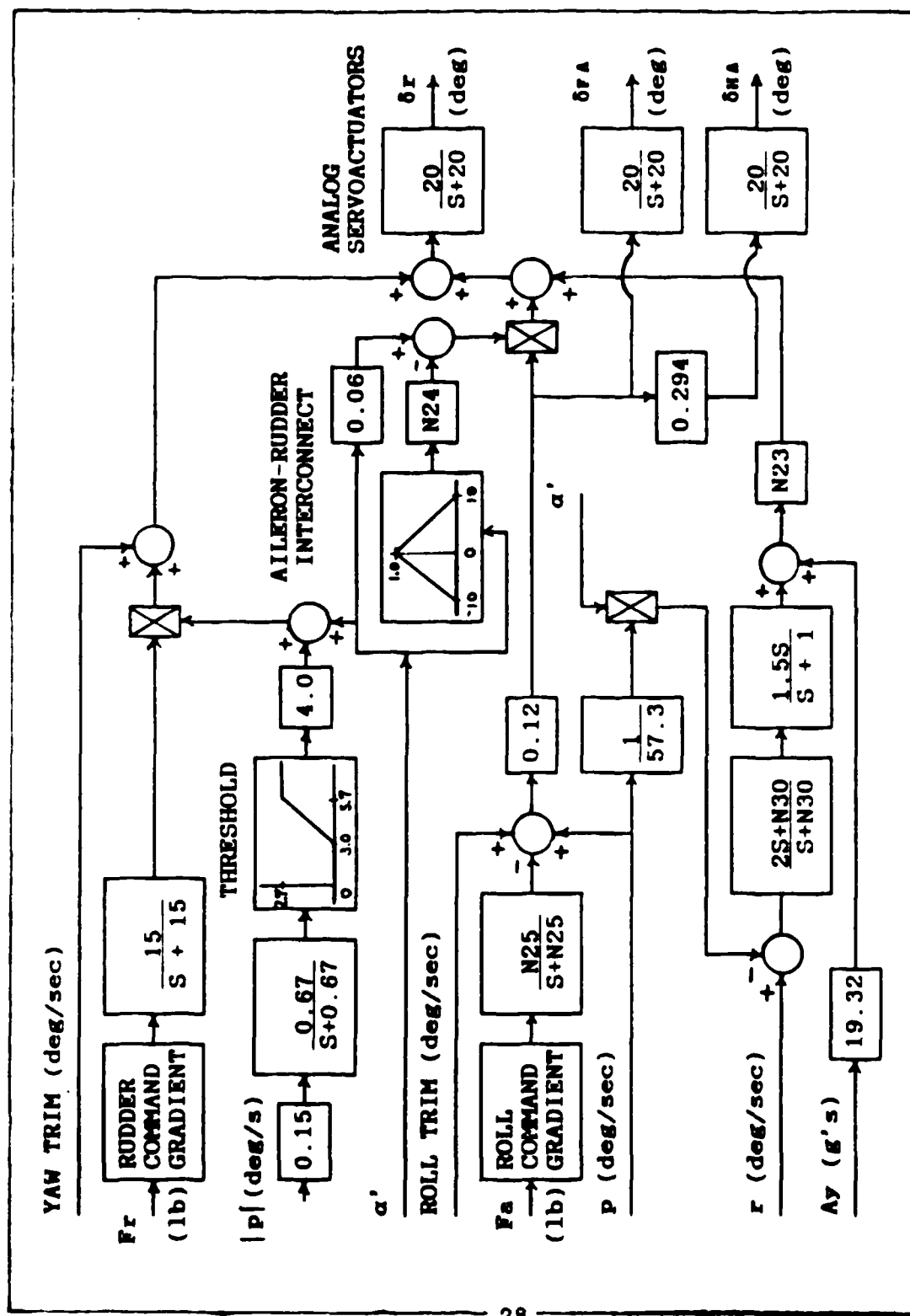


Figure 3.7. Simplified Lateral-Directional DFCS Block Diagram

3. The analog demodulators are removed. These are removed based on the assumption that the frequency content of the various signals will be relatively low.

4. The aircraft is considered to be flying with no extended landing gear or extended flaps and is not in an air refueling mode. This eliminates the possibility of any 'notes' changing the configuration of the control system.

5. The servoactuators are simplified to a simple 20 rad/sec first-order lag filter which corresponds to the lowest frequency filter of the actual servoactuators.

6. For the lateral-directional mode, the gun compensation is ignored.

7. The yaw structural filter and the roll filters are eliminated considering high frequency vibrations that these filters would attenuate as insignificant.

8. The leading-edge flap deflection is considered zero for all time for this thesis. This has little effect on the performance of the overall control system.

9. Any automatic angle of attack limiter loops are eliminated, simplifying the loop nesting of the block diagram.

10. The pitch command gradient for the simplified longitudinal controller is shown in Figure 3.8. The roll command gradient and the rudder command gradient for the simplified lateral-directional controller are shown in Figures 3.9 and 3.10.

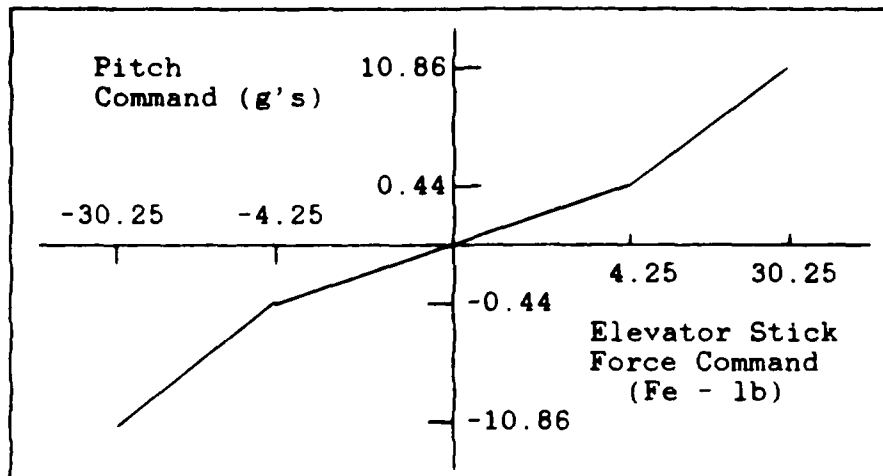


Figure 3.8. Pitch Command Gradient

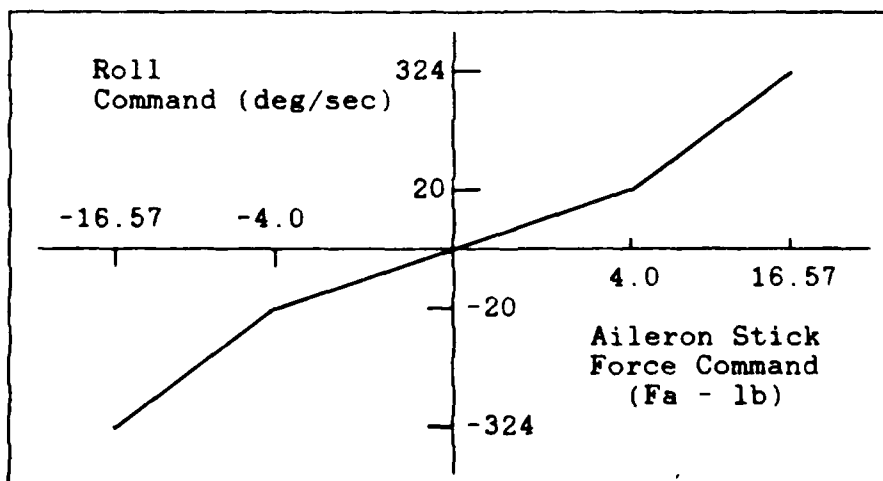


Figure 3.9. Roll Command Gradient

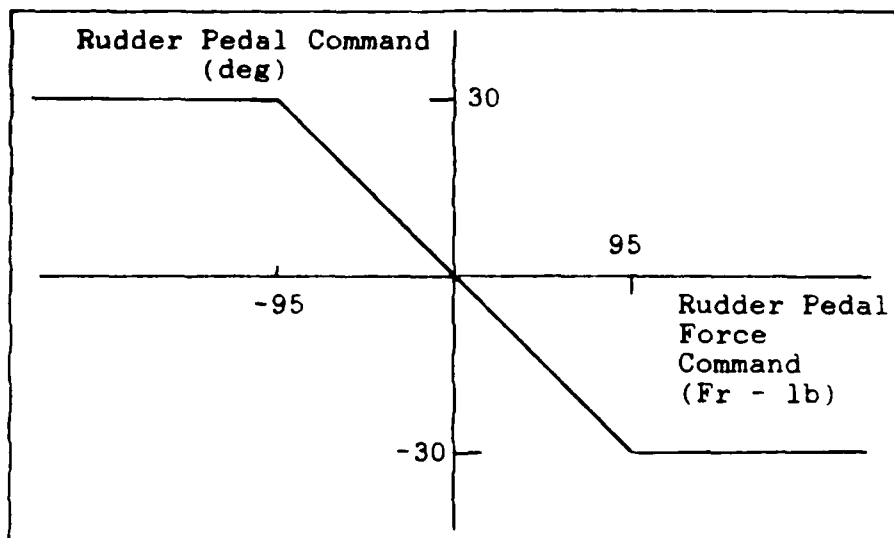


Figure 3.10. Rudder Command Gradient

The simplifications are performed without significant model degradation and provide a rather good controller for the specific flight condition and the surrounding flight envelope.

An aircraft and a simplified flight control system are described in this chapter. The next chapter discusses the techniques used for modelling the aircraft and the flight control system.

IV. Simulation Modelling

Overview

The aircraft mathematical model used for this thesis is based upon the F-16A 'CLEAN' aerodynamic data base obtained from ASD/ENFTC, Wright-Patterson AFB.

The complete simulation model consists of the equations of motion, the atmospheric model, the aerodynamic data format, the aerodynamic interpolation equations, the aerodynamic coefficient equations and the backward difference equation. Also important to the simulation model are the sign and axes conventions.

Axes and Sign Conventions

The standard sign convention and axes systems for the F-16 are used throughout this thesis. This sign convention is shown in Figure 4.1 and is summarized by the following:

$$\begin{aligned}\delta F &= \frac{1}{2} * (\delta FR + \delta FL) \\ \delta FA &= \frac{1}{2} * (\delta FR - \delta FL) \\ \delta H &= \frac{1}{2} * (\delta HR + \delta HL) \\ \delta HA &= \frac{1}{2} * (\delta HR - \delta HL)\end{aligned}$$

δF , δH , δHR and δFR are positive with the trailing edge down and δHL and δFL are positive with the trailing edge up. δr is positive with the trailing edge to the left of the aircraft.

If the pilot stick is pulled back towards the pilot, the resulting aircraft motion is a pitch up of the nose. If the pilot stick is pushed to the pilot's right the aircraft's right

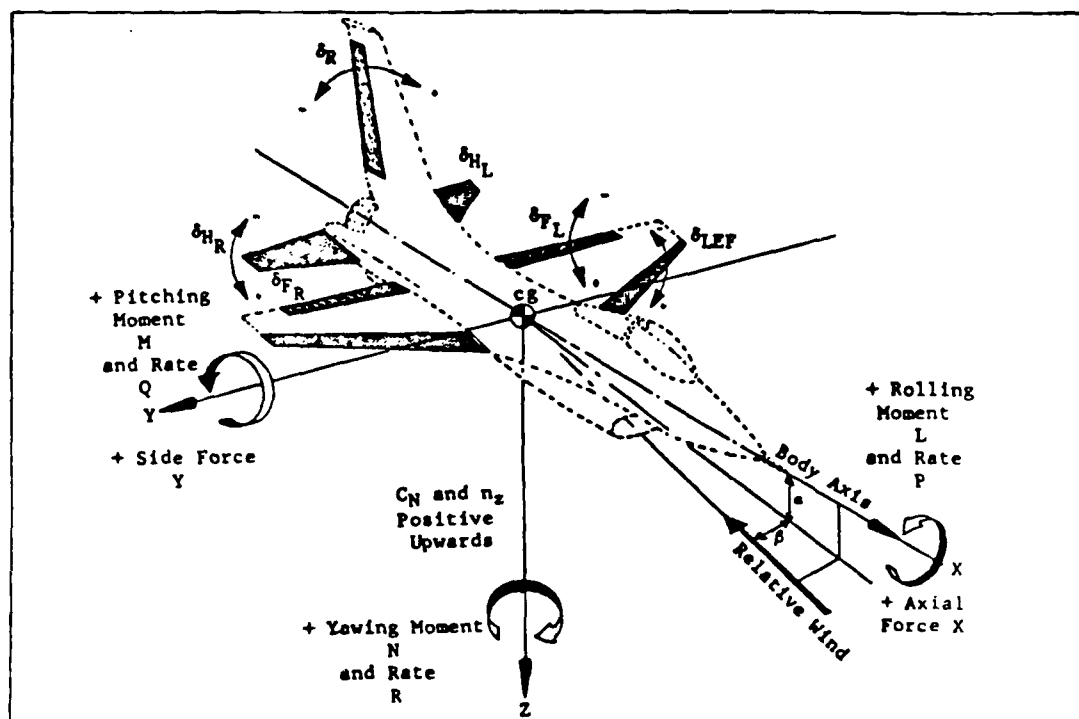


Figure 4.1. F-16 Sign Convention

wing will motion downward while the left wing will motion upward. If the right rudder pedal is pushed forward, the aircraft's nose will point toward the right.

The three axes used throughout this thesis are the stability, body and wind axes. They are shown together in Figure 4.2. Notice the aerodynamic forces and moments are given in the stability axes and the true forward velocity (V_T) is given in the wind axes. All axes have origins at the center of gravity of the aircraft.

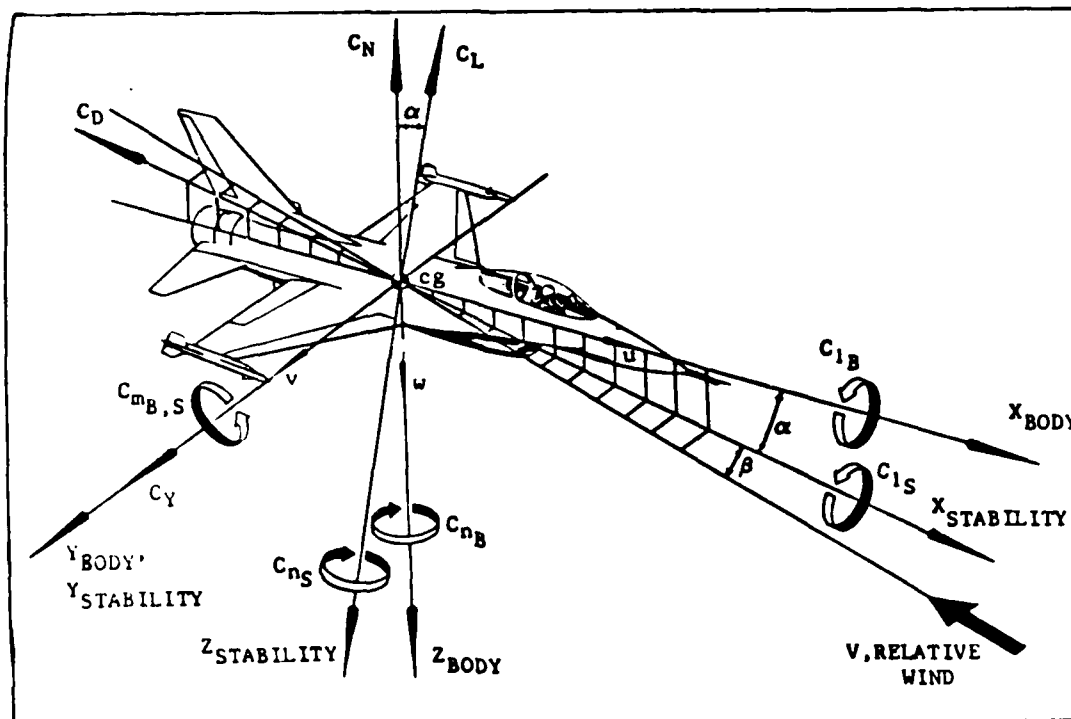


Figure 4.2. F-16 Axes System

Equations of Motion

The equations of motion for the aircraft, which are derived from Newton's Second Law of motion, form the basis for determining the motion of the aircraft through its environment when acted upon by external forces and moments.

The equations of motion describe six degrees of freedom and they are manipulated through the body, stability, wind and earth axes in order to completely determine the various positions, velocities and accelerations of the aircraft.

Assumptions made in this derivation are that the mass of the aircraft is constant, the earth is an inertial reference frame and the atmosphere is at rest with respect to the earth.

The equations in this section were taken from Reference (14) unless otherwise noted.

Stability Axis Translational Forces. The inputs to these equations include the thrust (TH) and the lift, drag and side force coefficients. The lift, drag and side force coefficients are dimensionalized by multiplying them by the dynamic pressure ($\bar{q}$) and the wing area (S). The dimensionalizing is done for all of the aerodynamic coefficients used in the equations of motion. The force due to gravity and thrust are both forces normally given in the body axis and are appropriately converted to the stability axis to conform with the aerodynamic coefficients.

$$X_s = (TH - AW\sin\theta)\cos\alpha + (AW\cos\theta\cos\Phi)\sin\alpha - \bar{q}SCD_{TOTAL} \quad (IV-1)$$

$$Y_s = AW\cos\theta\sin\Phi + \bar{q}SCY_{TOTAL} \quad (IV-2)$$

$$Z_s = (AW\cos\theta\cos\Phi)\cos\alpha - (TH - AW\sin\theta)\sin\alpha - \bar{q}SCL_{TOTAL} \quad (IV-3)$$

Wind Axis Translational Forces. The results from Eqs (IV-1), (IV-2) and (IV-3) are converted to the wind axis.

$$X_w = X_s\cos\beta + Y_s\sin\beta \quad (IV-4)$$

$$Y_w = Y_s\cos\beta + X_s\sin\beta \quad (IV-5)$$

$$Z_w = Z_s \quad (IV-6)$$

The translational forces in the wind axis (6:141) are also known to be equal to

$$X_w = m\dot{V}_T \quad (IV-7)$$

$$Y_w = mV_T r_w \quad (IV-8)$$

$$Z_w = -mV_T q_w \quad (IV-9)$$

Body Axis Moments. The moments are due totally to the aerodynamic coefficients because all external forces act upon the aircraft through the center of gravity.

$$L = \bar{q}Sb(Cl_{TOTAL} \cos \alpha - Cn_{TOTAL} \sin \alpha) \quad (IV-10)$$

$$M = \bar{q}S\bar{c}Cm_{TOTAL} \quad (IV-11)$$

$$N = \bar{q}Sb(Cl_{TOTAL} \sin \alpha + Cn_{TOTAL} \cos \alpha) \quad (IV-12)$$

Body Axis Angular Accelerations. The full equations appear here, but the thesis assumes the aircraft is symmetrical and that the longitudinal axis is in a plane of symmetry, thus the products of inertia, I_{yz} and I_{xy} are set to zero. Also, the thesis does not take into account any engine gyroscopic terms so the engine moment of inertia (I_e) is set to zero.

$$\begin{aligned} \dot{p} = [L + (I_y - I_z)qr + I_{xz}(\dot{r} + pq) \\ + I_{yz}(q^2 - r^2) + I_{xy}(\dot{q} - rp)]/I_x \end{aligned} \quad (IV-13)$$

$$\begin{aligned} \dot{q} = [M + (I_z - I_x)rp + I_{xz}(r^2 - p^2) \\ + I_{xy}(\dot{p} + qr) + I_{yz}(\dot{r} - pq) - I_e \omega_e r]/I_y \end{aligned} \quad (IV-14)$$

$$\begin{aligned} \dot{r} = [N + (I_x - I_y)pq + I_{xz}(\dot{p} - qr) \\ + I_{xy}(p^2 - q^2) + I_{yz}(\dot{q} + rp) + I_e \omega_e q]/I_z \end{aligned} \quad (IV-15)$$

Simply integrating Equations (IV-13), (IV-14) and (IV-15) yields the body axis angular velocities.

Stability Axis Angular Accelerations. The angular accelerations in the stability axes are necessary for finding the aerodynamic angles.

$$\dot{p}_s = p \cos \alpha + r \sin \alpha \quad (\text{IV-16})$$

$$\dot{q}_s = q \quad (\text{IV-17})$$

$$\dot{r}_s = r \cos \alpha - p \sin \alpha \quad (\text{IV-18})$$

True Velocity and Aerodynamic Angles. Using Eqs (IV-7), (IV-8), (IV-9) and the relationships between the axes, the following becomes true.

$$\dot{V}_T = X_w/m \quad (\text{IV-19})$$

$$\dot{\alpha} = q_s - p_s \tan \beta + Z_w/(mV_T \cos \beta) \quad (\text{IV-20})$$

$$\dot{\beta} = Y_w/(mV_T) - r_s \quad (\text{IV-21})$$

Integrating Equations (IV-19), (IV-20) and (IV-21) gives the true velocity and the aerodynamic angles.

Euler Angle Rates. The Euler angle rates are used to determine the Euler angles. The Euler angles describe the rotation of the aircraft in the body axis with respect to the vehicle-carried axis. The vehicle-carried axis, called the navigation frame, has origin at the aircraft center of gravity, with X-axis pointing north, Y-axis pointing east and Z-axis pointing downward towards the origin of the earth axes.

$$\dot{\psi} = (r \cos \theta + q \sin \theta)/\cos \theta \quad (\text{IV-22})$$

$$\dot{\theta} = q \cos \theta - r \sin \theta \quad (\text{IV-23})$$

$$\dot{\phi} = p + \dot{\psi} \sin \theta \quad (\text{IV-24})$$

Integrating Eqs (IV-22), (IV-23) and (IV-24) give the Euler angles.

Body Axis Translational Velocities. Finding the translational velocities in the body axes are needed for the calculation of the position and velocity of the aircraft with respect to the earth.

$$u = V_T \cos \alpha \cos \beta \quad (\text{IV-25})$$

$$v = V_T \sin \beta \quad (\text{IV-26})$$

$$w = V_T \sin \alpha \cos \beta \quad (\text{IV-27})$$

Direction Cosines. The direction cosines help define the position of the body axes with respect to the earth axes based on the Euler angles found in Eqs (IV-22), (IV-23) and (IV-24).

$$a_{11} = \cos \theta \cos \psi \quad (\text{IV-28a})$$

$$a_{21} = \sin \theta \cos \psi \sin \phi - \sin \psi \cos \phi \quad (\text{IV-28b})$$

$$a_{31} = \sin \theta \cos \psi \cos \phi + \sin \psi \sin \phi \quad (\text{IV-28c})$$

$$a_{12} = \cos \theta \sin \psi \quad (\text{IV-28d})$$

$$a_{22} = \sin \theta \sin \psi \sin \phi + \cos \psi \cos \phi \quad (\text{IV-28e})$$

$$a_{32} = \sin \theta \sin \psi \cos \phi - \cos \psi \sin \phi \quad (\text{IV-28f})$$

$$a_{13} = -\sin \theta \quad (\text{IV-28g})$$

$$a_{23} = \cos \theta \sin \phi \quad (\text{IV-28h})$$

$$a_{33} = \cos \theta \cos \phi \quad (\text{IV-28i})$$

Earth Axis Velocities. The direction cosines from Eqs (IV-28a) through (IV-28i) are used to convert the body axis translational velocities (u, v and w) from Equations (IV-25), (IV-26) and (IV-27) into velocities in the earth axes.

$$\begin{bmatrix} \dot{X}_e \\ \dot{Y}_e \\ \dot{Z}_e \end{bmatrix} = \begin{bmatrix} a_{11}u + a_{21}v + a_{31}w \\ a_{12}u + a_{22}v + a_{32}w \\ a_{13}u + a_{23}v + a_{33}w \end{bmatrix} = A \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (\text{IV-29})$$

$$(\text{IV-30})$$

$$(\text{IV-31})$$

Integrating Equations (IV-29), (IV-30) and (IV-31) give the position of the aircraft in the earth axes.

Miscellaneous Equations. The following important equations give further information on the state of the aircraft. Equation (IV-32) defines the altitude rate of change as the negative of the earth Z-axis velocity. Integrating Equation (IV-32) gives the altitude. Equation (IV-33) defines the flight path angle. Equations (IV-34) and (IV-35) respectively, define the normal load factors at the center of gravity and at the accelerometer. Equations (IV-36) and (IV-37) respectively, define the side load factors at the center of gravity and at the accelerometer. The distance from the accelerometers to the leading edge of the wing reference chord measured along the body X-axis is 9.96 feet and along the body Z-axis is 0.289 feet. Equation (IV-38) calculates the thrust required for no acceleration along the body X-axis.

$$\dot{h} = -\dot{Z}_e \quad (\text{IV-32})$$

$$\gamma = \arcsin(\dot{h}/V_T) \quad (\text{IV-33})$$

$$N_{zcg} = (\bar{q}S/mg)(CD_{TOTAL}\sin\alpha + CL_{TOTAL}\cos\alpha) \quad (\text{IV-34})$$

$$N_z = N_{zcg} + [(9.96 + \bar{c}CG)(\dot{q} - pr) + 0.289(p^2 + q^2)]/g \quad (\text{IV-35})$$

$$A_{ycg} = (Y_s/mg) \quad (\text{IV-36})$$

$$A_y = A_{ycg} + [(9.96 + \bar{c}CG)(\dot{r} + pq) + 0.289(qr - \dot{p})]/g \quad (\text{IV-37})$$

$$TH = AW\sin\theta + (CD_{TOTAL} - AW\cos\theta\sin\alpha)/\cos\alpha \quad (\text{IV-38})$$

An overall scheme is summarized in the simulation flow diagram of Figure 4.3.

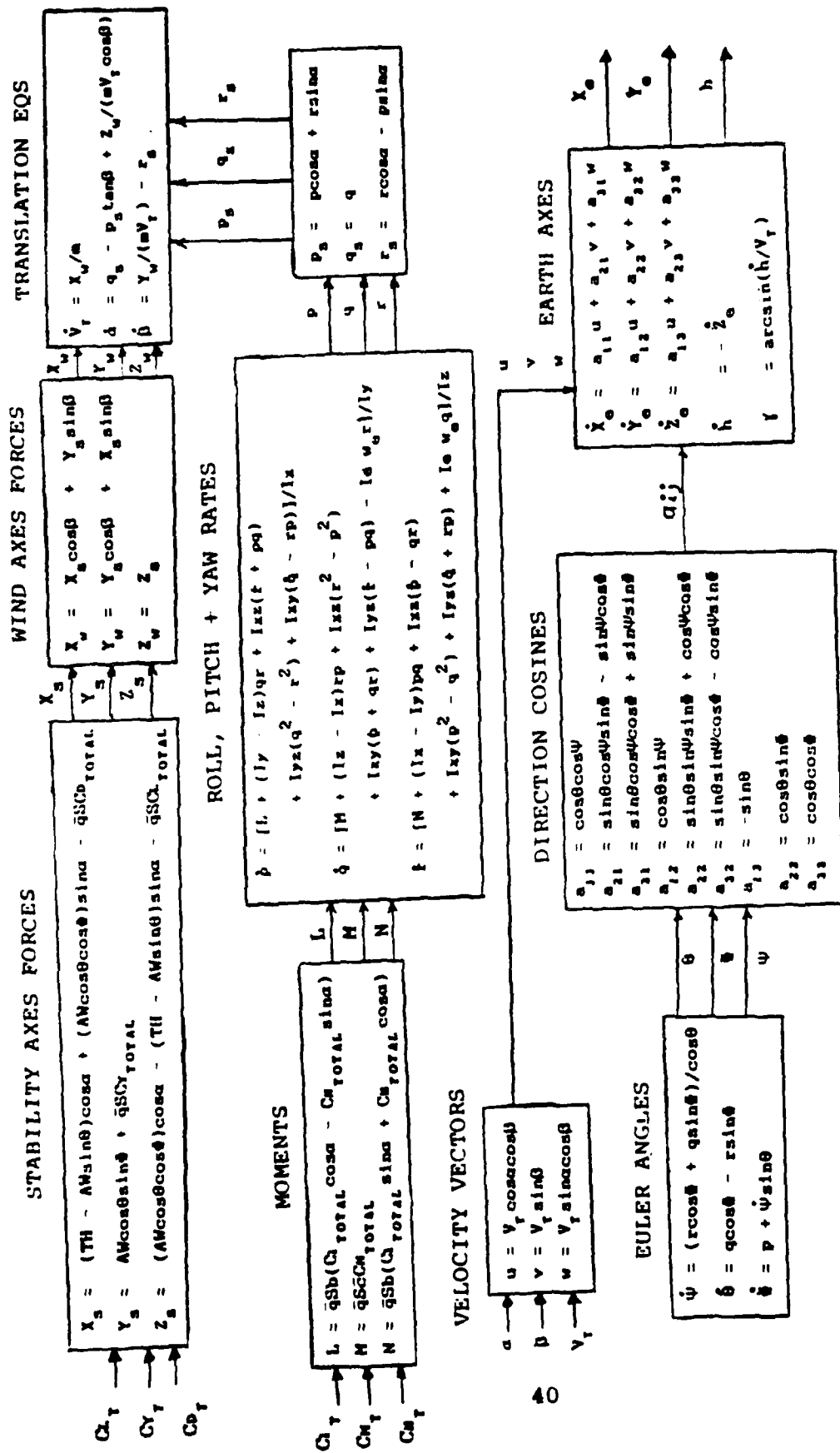


Figure 4.3. Equations of Motion Simulation Flow Diagram

Atmospheric Model

The atmospheric model defines the density, sonic velocity, Mach number, dynamic pressure, static pressure and total pressure. The model considers two regions of the atmosphere; the troposphere and the stratosphere. The troposphere is defined as the portion of the atmosphere starting at the earth's surface and extending to an altitude of 36,089.24 feet. The stratosphere is the portion of the atmosphere extending above the troposphere. The equations in this section are taken from Reference (14).

Density of Air. The density of air is determined by using the combined gas laws of Boyle and Charles which is a relationship between the temperature, pressure and density of a gas. Using the assumption that temperature is a linear function of altitude in the troposphere and a constant in the stratosphere a ratio of the density of air at a given altitude and the density at sea level can be determined as shown in Equation (IV-39). With the air density at sea level given as 0.002378 slug/cubic ft, the equation results in the values listed in Table II for altitude up to 50,000 feet.

The density below is used in calculating the dynamic pressure.

$$\rho = \rho_0 (1.0 - 0.00000688 * h)^{4.256} \quad (\text{IV-39})$$

Altitude (ft)	Temp (°F)	(slug/cu ft)	
0	59.0	0.002378	↑ Troposphere ↓
1000	55.4	0.002309	
2000	51.8	0.002242	
3000	48.4	0.002176	
4000	44.8	0.002112	
5000	41.2	0.002049	
6000	37.6	0.001987	
7000	34.0	0.001928	
8000	30.6	0.001869	
9000	27.0	0.001812	
10000	23.4	0.001756	
11000	19.8	0.001701	
12000	16.2	0.001648	
13000	12.6	0.001596	
14000	9.2	0.001545	
15000	5.5	0.001496	
20000	-12.3	0.001267	↑ Stratosphere ↓
25000	-30.1	0.001065	
30000	-48.1	0.000889	
35000	-65.8	0.000736	
40000	-67.0	0.000582	
45000	-67.0	0.000459	
50000	-67.0	0.000361	

Table II. Altitude-Density Relation based on NACA No. 218
(18:7)

Sonic Velocity. The sonic velocity is necessary for calculating the Mach number used in the aerodynamic data base. Its determination is based upon the altitude but is constant in the stratosphere region due to the assumed constant temperature.

For $h < 36,089.24$ ft:

$$a = 1116.45(1.0 - 0.00000688 * h)^{1/2} \text{ ft/sec (IV-40)}$$

For $h \geq 36,089.024$ ft:

$$a = 968.088 \text{ ft/sec}$$

Mach Number. The Mach number is the ratio between the true speed of the aircraft and the sonic speed and is used as an independent variable in the aerodynamic data base.

$$\text{MACH} = V_T/a \quad (\text{IV-41})$$

Dynamic Pressure. The dynamic pressure is used in the equations of motion to dimensionalize the aerodynamic coefficients.

$$\bar{q} = 0.5 \rho V_T^2 \quad \text{lb/ft}^2 \quad (\text{IV-42})$$

Static Pressure.

In the troposphere:

$$P_S = P_0 (1.0 - 0.00000688 * h)^{5.2561} \quad \text{lb/ft}^2 \quad (\text{IV-43})$$

where $P_0 = 2116.216 \text{ lb/ft}^2 = \text{sea level static pressure}$

In the stratosphere:

$$P_S = 472.68 * \exp[-4.80634E-5 * (h - 36089.24)] \quad \text{lb/ft}^2 \quad (\text{IV-44})$$

Total Pressure. Total pressure is separated into two equations based on whether the aircraft is in subsonic or supersonic flight. The total pressure of the aircraft is used primarily in flight control system gain scheduling.

For $\text{MACH} \leq 1.0$

$$q_c = P_S * [(1.0 + 0.2 * \text{MACH}^2)^{3.5} - 1.0] \quad \text{lb/ft}^2 \quad (\text{IV-45})$$

For MACH > 1.0

$$q_c = p_s \{ 166.921 [MACH^7 / (7 * MACH^2 - 1.0)^{2.5}] - 1.0 \}$$

lb/ft² (IV-46)

Aerodynamic Data Format

The aerodynamic data base, consisting of 18,816 values, is organized into eight groups of coefficients, which contains all non-linear stability derivatives necessary for changing the aircraft plant characteristics for a changing flight condition. Each group consists of data for several aerodynamic coefficients related by their independent variables. These independent variables, numbering from two to four per coefficient, are specified over a certain range and at certain intervals. The technique for looking up any particular coefficient in the data base is to vary the first independent variable through its breakpoint values until the maximum breakpoint value is reached, then increase the second independent variable to its next breakpoint value and again step through the first variable. Continue this until all variables and their breakpoint values have been exhausted. This process holds for the entire data base. The aerodynamic data is given completely in the stability axes. The data format was taken from Reference (14).

Group I: Longitudinal Data. The lift, drag and pitching moment coefficients are in this group of 6300 entries and they have the following format:

CL = CL (AOA, MACH, DH, DLEF)
CD = CD (AOA, MACH, DH, DLEF)
CM = CM (AOA, MACH, DH, DLEF)

The independent variable ranges and breakpoint values are

AOA = -4 to +34 degrees in 2 degree increments
MACH = 0.2, 0.6, 0.8, 0.9, 1.2, 1.6, 2.0
DH = -25, -10, 0, +10, +25 degrees
DLEF = 0, 15, 25 degrees

where

AOA is angle of attack,
MACH is Mach number,
DH is symmetrical horizontal tail deflection and
DLEF is leading-edge flap deflection.

Group II: Lateral-Directional Data. The side force, rolling moment and yawing moment coefficients are in this group of 3276 entries and they have the following format:

CY = CY ($|\beta|$, MACH, AOA, DLEF)
Cl = Cl ($|\beta|$, MACH, AOA, DLEF)
CN = CN ($|\beta|$, MACH, AOA, DLEF)

The independent variable ranges and breakpoint values are

$|\beta|$ = 0, 1, 2, 3, 4, 6, 8, 10, 12, 14, 16, 18, 20 deg
MACH = 0.2, 0.6, 0.8, 0.9, 1.2, 1.6, 2.0
AOA = 0, 10, 20, 30 degrees
DLEF = 0, 15, 25 degrees

where

β is sideslip angle,
MACH is Mach number,
AOA is angle of attack and
DLEF is leading-edge flap deflection.

All lateral-directional data are assumed to vary symmetrically with sideslip angle, therefore the absolute value of sideslip angle is used for the independent variable. The simulation must later take into account the sign with the following

$$CY(\beta, MACH, AOA, DLEF) = CY(|\beta|, MACH, AOA, DLEF) \times \text{sign}(\beta)$$

$$Cl(\beta, MACH, AOA, DLEF) = Cl(|\beta|, MACH, AOA, DLEF) \times \text{sign}(\beta)$$

$$CN(\beta, MACH, AOA, DLEF) = CN(|\beta|, MACH, AOA, DLEF) \times \text{sign}(\beta)$$

Group III: Roll and Yaw Effects. Linearized control derivatives due to flaperon and differential horizontal tail deflections are in this group of 1680 entries and they are in the following format:

$$Cl_{\delta FA} = Cl_{\delta FA} (AOA, MACH, DLEF)$$

$$CN_{\delta FA} = CN_{\delta FA} (AOA, MACH, DLEF)$$

$$Cl_{\delta HA} = Cl_{\delta HA} (AOA, MACH, DLEF)$$

$$CN_{\delta HA} = CN_{\delta HA} (AOA, MACH, DLEF)$$

The independent variable ranges and breakpoint values are

AOA = -4 to +34 degrees in 2 degree increments
MACH = 0.2, 0.6, 0.8, 0.9, 1.2, 1.6, 2.0
DLEF = 0, 15, 25 degrees

where

AOA is angle of attack,
MACH is Mach number and
DLEF is leading-edge flap deflection.

Group IV: Side Force Effects and Flight Test Corrections.

This group of 560 entries contains the side force effects of the flaperons and the differential horizontal tail, along with flight test corrections to the pitching moment and lift coefficients. They have the following format:

$$CY_{\delta FA} = CY_{\delta FA} (AOA, MACH)$$

$$CY_{\delta HA} = CY_{\delta HA} (AOA, MACH)$$

$$\Delta CL_{FT} = \Delta CL_{FT} (AOA, MACH)$$

$$\Delta CM_{FT} = \Delta CM_{FT} (AOA, MACH)$$

The independent variable ranges and breakpoint values are

AOA = -4 to +34 degrees in 2 degree increments
MACH = 0.2, 0.6, 0.8, 0.9, 1.2, 1.6, 2.0

where

AOA is angle of attack and
MACH is Mach number.

Group V: Rudder Effects. This group of 1260 entries, contains the rudder effects and they have the following format:

$CY_{\delta r} = CY_{\delta r} (AOA, MACH, |\delta r|)$
 $CN_{\delta r} = CN_{\delta r} (AOA, MACH, |\delta r|)$
 $Cl_{\delta r} = Cl_{\delta r} (AOA, MACH, |\delta r|)$

The independent variable ranges and breakpoint values are

AOA = -4 to +34 degrees in 2 degree increments
MACH = 0.2, 0.6, 0.8, 0.9, 1.2, 1.6, 2.0
 $|\delta r| = 10, 20, 30$ degrees

where

AOA is angle of attack,
MACH is Mach number and
 δr is rudder deflection.

The coefficients are not tabulated for negative rudder deflection because they are assumed the same as the positive values. Also, any absolute value less than 10 degrees deflection should be treated as a deflection of 10 degrees only for the purpose of this look-up.

Group VI: Dynamic Derivatives. Lift and pitching moment derivatives with respect to pitch rate and AOA rate, and side force, rolling and yawing moment derivatives with respect to roll and yaw rate are in this group of 1400 entries. They have the following format:

$$\begin{aligned}
CL_q &= CL_q (AOA, MACH) \\
CM_q &= CM_q (AOA, MACH) \\
CL_\alpha &= CL_\alpha (AOA, MACH) \\
CM_\alpha &= CM_\alpha (AOA, MACH) \\
CY_p &= CY_p (AOA, MACH) \\
Cl_p &= Cl_p (AOA, MACH) \\
CN_p &= CN_p (AOA, MACH) \\
CY_r &= CY_r (AOA, MACH) \\
Cl_r &= Cl_r (AOA, MACH) \\
CN_r &= CN_r (AOA, MACH)
\end{aligned}$$

The independent variable ranges and breakpoint values are

AOA = -4 to +34 degrees in 2 degree increments
MACH = 0.2, 0.6, 0.8, 0.9, 1.2, 1.6, 2.0

where

AOA is angle of attack and
MACH is Mach number.

Group VI: Flex/Rigid Ratios. The flexibility corrections, in terms of flex/rigid ratios, due to rotation rates, surface positions and sideslip angle are in this group of 1140 entries.

$$\begin{aligned}
Rl_p &= Rl_p (MACH, H1) \\
RL_q &= RL_q (MACH, H1) \\
RM_q &= RM_q (MACH, H1) \\
RY_{rVT} &= RY_{rVT} (MACH, H1) \\
RN_{rVT} &= RN_{rVT} (MACH, H1) \\
RL_{\delta H} &= RL_{\delta H} (MACH, H1) \\
RM_{\delta H} &= RM_{\delta H} (MACH, H1) \\
RY_{\delta r} &= RY_{\delta r} (MACH, H1) \\
Rl_{\delta r} &= Rl_{\delta r} (MACH, H1) \\
RN_{\delta r} &= RN_{\delta r} (MACH, H1) \\
Rl_{\delta a} &= Rl_{\delta a} (MACH, H1) \\
Rl_{\delta FA} &= Rl_{\delta FA} (MACH, H1) \\
RY_\beta &= RY_\beta (MACH, H2) \\
Rl_\beta &= Rl_\beta (MACH, H2)
\end{aligned}$$

$$RN_{\beta} = RN_{\beta} (MACH, H2)$$

The independent variable ranges and breakpoint values are

MACH = 0.2 to 2.0 in 0.1 increments
H1 = 0, 10,000, 25,000, 50,000 feet
H2 = 0, 20,000, 40,000, 50,000 feet

where

MACH is Mach number and
H1 and H2 are altitude.

Group VIII: Flexibility Increments. The flexibility increments of lift and pitching moment are in this group of 3200 entries. They have the following format:

$$\Delta CL_{FLEX} = \Delta CL_{FLEX} (AOA, H1, DLEF, MACH)$$

$$\Delta CM_{FLEX} = \Delta CM_{FLEX} (AOA, H1, DLEF, MACH)$$

The independent variable ranges and breakpoint values are

AOA = -4 to +34 degrees in 2 degree increments
H1 = 0, 10,000, 25,000, 50,000 feet
DLEF = 0, 5, 10, 25 degrees
MACH = 0.2, 0.6, 0.9, 1.2, 1.6

where

AOA is angle of attack,
H1 is altitude,
DLEF is leading-edge flap deflection and
MACH is Mach number.

Aerodynamic Interpolation Equations

In order to determine coefficient values at any flight condition, this thesis interpolates between breakpoints with a straight line approximation. This technique is demonstrated by the following two-variable problem.

Consider X and Y as the independent variables for the discrete function $f(X,Y)$ for which data exists only at certain

breakpoint values. Let X_1 and X_2 be consecutive breakpoint values for the independent variable X , and Y_1 and Y_2 be consecutive breakpoint values for the independent variable Y . The data for the discrete function, $f(X_1, Y_1)$, $f(X_1, Y_2)$, $f(X_2, Y_1)$ and $f(X_2, Y_2)$, are given at the breakpoints. Also X and Y for a given flight condition must satisfy the following condition.

$$\begin{aligned} X_1 &\leq X \leq X_2 \\ Y_1 &\leq Y \leq Y_2 \end{aligned}$$

To find $f(X, Y)$ the following equations are computed in order.

$$f(X, Y_1) = f(X_1, Y_1) + \frac{(X - X_1)}{X_2 - X_1} (f(X_2, Y_1) - f(X_1, Y_1)) \quad (\text{IV-47})$$

$$f(X, Y_2) = f(X_1, Y_2) + \frac{(X - X_1)}{X_2 - X_1} (f(X_2, Y_2) - f(X_1, Y_2)) \quad (\text{IV-48})$$

$$f(X, Y) = f(X, Y_1) + \frac{(Y - Y_1)}{Y_2 - Y_1} (f(X, Y_2) - f(X, Y_1)) \quad (\text{IV-49})$$

The technique is to interpolate on the first variable first, followed by the second. Eq (IV-47) takes the value at the first breakpoint, $f(X_1, Y_1)$ and holding the second independent variable constant as Y_1 , adds a correction term based upon the straight line distance of X between the two breakpoints, X_1 and X_2 . Eq (IV-48) does the same as Eq (IV-47) except holds the second independent variable constant as Y_2 . Eq (IV-49) then takes the results of Eqs (IV-47) and (IV-48) and adds a correction term to the first breakpoint result found in Eq (IV-1), $f(X, Y_1)$ based upon the straight line distance of the second variable, Y , between the two breakpoints, Y_1 and Y_2 .

This procedure is extended to three and four variable functions.

Aerodynamic Coefficient Equations

After all coefficient values are determined, they are combined, with aircraft motion variables and control surface deflections, in the aerodynamic coefficient equations to yield total force and moment coefficients. These equations are given entirely in terms of the stability axes. The stability and control derivatives, except the control surface deflection derivatives, are defined as per radian therefore all angles and their derivatives are expressed in radian measure. The control surface deflection derivatives which include all derivatives based on δr , δHA , δH , δFA and δa , are defined as per degree therefore all deflection angles are expressed in degrees. The totals given in the following equations are incorporated into the equations of motion discussed earlier. The equations in this section are taken from Reference (14).

Lift.

$$\begin{aligned} CL_{TOTAL} = & (CL + \Delta CL_{FLEX})|_{\delta H=0 \text{ deg}} + RL_{\delta H}(CL|_{\delta H=x \text{ deg}} \\ & - CL|_{\delta H=0 \text{ deg}}) + (RL_q CL_q q_s + RL_{\dot{q}} CL_{\dot{q}} \dot{\alpha}) (\bar{c}/2V_T) \\ & + \Delta CL_{FT} \end{aligned} \quad (IV-50)$$

The coefficient of lift (CL) is evaluated at the current flight condition ($\delta H = x \text{ deg}$) and at the condition with zero symmetrical horizontal tail deflection ($\delta H = 0 \text{ deg}$). This is shown by the subscript notation.

Side Force.

$$\begin{aligned} CY_{TOTAL} = & RY_{\beta} CY + Rl_{\delta FA} CY_{\delta FA} \delta FA + CY_{\delta HA_{FLEX}} \delta HA \\ & + RY_{\delta r} CY_{\delta r} \delta r + (CY_{p_s} + RY_{rVT} CY_{r_s})(b/2V_T) \quad (IV-51) \end{aligned}$$

where

$$\begin{aligned} CY_{\delta HA} &= 4(Rl_{\delta a} CY_{\delta a} - Rl_{\delta FA} CY_{\delta FA}) \\ CY_{\delta a} &= CY_{\delta FA} + 0.25(CY_{\delta HA}) \end{aligned}$$

Drag.

$$CD_{TOTAL} = CD + \Delta CD \quad (IV-52)$$

In this thesis, the fixed drag increment, ΔCD , is considered equal to zero. ΔCD could be useful in simulating an aircraft with a different store configuration by giving ΔCD a constant value.

Rolling Moment.

$$\begin{aligned} Cl_{TOTAL} = & Rl_{\beta} Cl + Rl_{\delta FA} Cl_{\delta FA} \delta FA + Cl_{\delta HA_{FLEX}} \delta HA \\ & + Rl_{\delta r} Cl_{\delta r} \delta r + (Rl_{p_s} Cl_{p_s} + Cl_{r_s})(b/2V_T) \\ & + (CG - 0.35)CY_{TOTAL} \bar{c}(\sin \alpha)/b \quad (IV-53) \end{aligned}$$

where

$$\begin{aligned} Cl_{\delta HA} &= 4(Rl_{\delta a} Cl_{\delta a} - Rl_{\delta FA} Cl_{\delta FA}) \\ Cl_{\delta a} &= Cl_{\delta FA} + 0.25(Cl_{\delta HA}) \\ CG &= (X_{ac} - X_{cg})/\bar{c} \end{aligned}$$

If the center of gravity (CG) is other than the nominal of 0.35 then a rolling moment not accounted for in the data, is produced in the aircraft. Therefore, for other than a nominal CG, a component of side force is added to Eq (IV-53). For this

thesis, the nominal $CG=0.35$ is used therefore eliminating the side force component.

Pitching Moment.

$$CM_{TOTAL} = CM_{0.35\bar{c}} + (CG - 0.35)(CL_{TOTAL} \cos \alpha + CD_{TOTAL} \sin \alpha) \quad (IV-54)$$

where

$$CM_{0.35\bar{c}} = (CM - \Delta CM_{FLEX})|_{\delta H=0 \text{ deg}} + RM_{\delta H}(CM|_{\delta H=x \text{ deg}} - CM|_{\delta H=0 \text{ deg}}) + (RM_q CM_q q_s + RM_{\dot{\alpha}} CM_{\dot{\alpha}} \dot{\alpha})(\bar{c}/2V_T) + \Delta CM_{FT} + DELCM$$

A component of lift and drag are added to the pitching moment coefficient to account for other than a nominal CG. Again this thesis used $CG=0.35$ eliminating the addition. Also the term, $DELCM$, is part of the pitching moment equation as a way of altering the characteristics of the pitching moment for special simulations. The term is not included in the thesis simulations.

Yawing Moment.

$$CN_{TOTAL} = CN_{0.35\bar{c}} + (CG - 0.35)CY_{TOTAL} \bar{c}(\cos \alpha)/b \quad (IV-55)$$

where

$$CN_{0.35\bar{c}} = RN_{\beta} CN + Rl_{\delta FA} CN_{\delta FA} \delta FA + CN_{\delta HA_{FLEX}} \delta HA + RN_{\delta r} CN_{\delta r} \delta r + (CN_{p_s} + RN_{rVT} CN_{r_s} r_s)(b/2V_T)$$

$$CN_{\delta HA_{FLEX}} = 4(Rl_{\delta a} CN_{\delta a} - Rl_{\delta FA} CN_{\delta FA})$$

$$CN_{\delta a} = CN_{\delta FA} + 0.25(CN_{\delta HA})$$

This thesis, again, ignores the side force component caused by a CG other than 0.35.

Backward Difference Equation

The backward difference equation is used for transforming Laplace domain transfer functions of the flight control system to a time domain approximation for use in a digital FORTRAN program. The first-backward difference equation (9:92) is

$$Dc(kT) = \left. \frac{dc(t)}{dt} \right|_{t=kT} \approx \frac{c(kT) - c[(k-1)T]}{T} \quad (\text{IV-56})$$

where

$k = 0, 1, 2, \dots$
 T is the sampling interval.
 D is the derivative operator.

As an example of its use consider the transfer function

$$\frac{C(s)}{U(s)} = \frac{10}{s + 10} \quad (\text{IV-57})$$

The first step is to rearrange Eq (IV-57) and then take the inverse Laplace transform.

$$(s + 10)C(s) = 10U(s)$$

$$(D + 10)c(t) = 10u(t)$$

$$Dc(t) + 10c(t) = 10u(t) \quad (\text{IV-58})$$

Substituting $Dc(t)$ in Eq (IV-58) with Eq (IV-56) when $t=kT$ to derive the difference equation at the discrete points.

$$\frac{c(kT) - c[(k-1)T]}{T} + 10c(kT) = 10u(kT)$$

Solving for $c(kT)$ gives

$$c(kT) = \frac{10Tu(kT) + c[(k-1)T]}{1 + 10T} \quad (IV-59)$$

Eq (IV-59) shows that the current output, $c(kT)$, is computed from the previous output, the current input and the sampling interval. All terms on the right hand side of Eq (IV-59) are available to the flight control system with the assumption that for $t \leq 0$ all past values are equal to zero. The only restriction in using the backward difference equation is that the order of the transfer function denominator be greater than or equal to the order of the numerator. If the transfer function is second-order then the second-backward difference equation must be applied. This and greater order transfer functions are explained in Reference (9), pages 89-96.

The modelling of the aircraft and flight control system described here are used in the design of the simulator discussed in the next chapter.

V. Simulation Design

Overview

The programmed tasks of the simulator are divided among the two SIMSTARs. The tasks include: a routine that takes current aircraft output variables and determines from them the various aerodynamic coefficients of the aircraft (changing plant characteristics), a routine that computes the dynamics of the aircraft through the earth's atmosphere, a routine which is the flight control system for the aircraft (compensator) and a routine for generating the cockpit display. Figure 5.1 shows the relationship of the routines in a system block diagram.

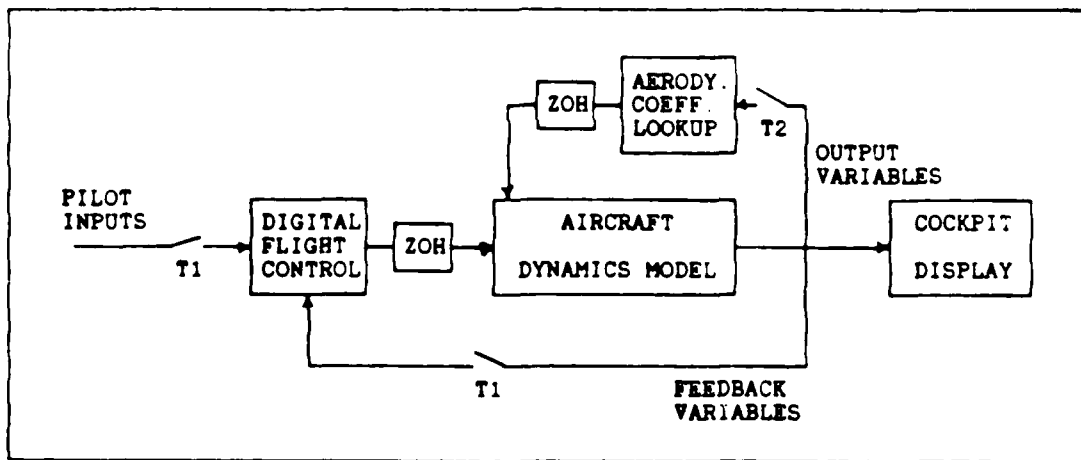


Figure 5.1. Simulator System Block Diagram

Interpolation and Look-up Routine

The aerodynamic data base as described in Chapter IV, is accessed by the Interpolation and Look-up Routine (ILUR). The routine is written in the digital region of SIMSTAR #114. Because of the anticipated long computation time for this routine, SIMSTAR #114 is dedicated to this task.

SIMSTAR #115 computes the dynamics model and outputs on the analog ports the variables α , β , $\dot{\alpha}$, p_s , q_s , r_s , V_T , h , δH , δHA , δFA , δr and δLEF (see Figure 5.2). The routine runs independently of the other routines accepting the SIMSTAR #115 inputs to the SIMSTAR #114's analog ports, sampling them and computing the six total aerodynamic coefficients, CL_{TOTAL} , CD_{TOTAL} , CM_{TOTAL} , Cl_{TOTAL} , CY_{TOTAL} and CN_{TOTAL} . The routine then sends these six dimensionalized coefficient values to SIMSTAR #114's analog ports where they are returned to SIMSTAR #115.

The interpolation and look-up routine is described next. The INITIAL region, which is executed only once at the beginning of the run, stores the aerodynamic data base into the memory of the SIMSTAR. The storage of the data base in the memory decreases the access time when compared with accessing data directly from a disk file. The INITIAL region calls a FORTRAN subroutine called SETUP which reads in values from two disk files called FILELIST and AC.DAT. The file FILELIST contains the array NOV which defines a format for reading AC.DAT such that SETUP knows what data is associated with which

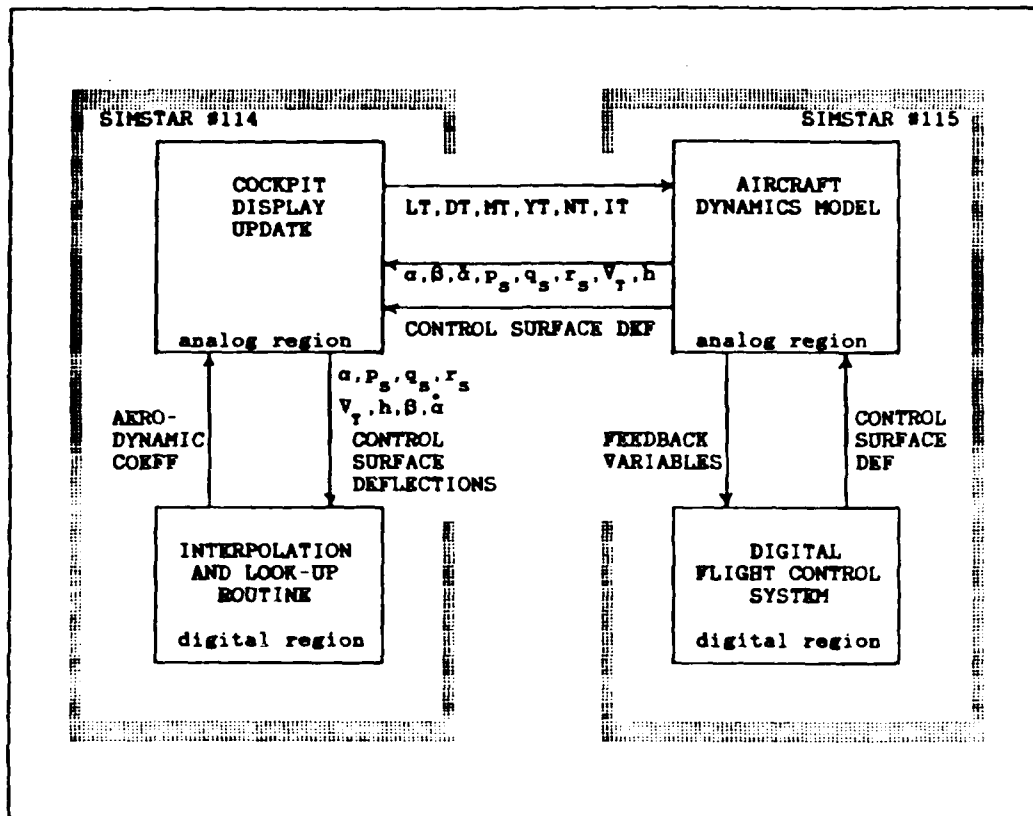


Figure 5.2. Simulator Flow Diagram displaying routine residence

two-, three- or four-variable function. The file AC.DAT contains the 18,816 table values described in Chapter IV. Once NOV is setup, AC.DAT is read into the five arrays, AA2, AA3, AA41, AA42, and AA43. The data is divided into five arrays of different dimensions to reduce the amount of memory used. Array AA2 contains two-variable functions, AA3 contains three-variable functions and AA41, AA42 and AA43 contain four-variable functions. Also defined in SETUP is an array which defines the breakpoint values for the independent variables

used for table look-up. The SETUP subroutine takes 100 seconds to execute and displays the message 'END OF INITIAL' when done.

After the initialization is completed, the FORTRAN subroutine, LOOKUP is executed. It resides in the digital DERIVATIVE region and as such is executed once every cycle as dictated by the user-defined variable, PERIOD. The aircraft output variables are the inputs to LOOKUP which determines the values of 46 two-, three- and four-variable functions. It accomplishes this by first looking-up the values of each function at the breakpoint values near the current independent variable values for each function. Then the routine uses the linear interpolation technique described in Chapter IV, to compute the actual function value at the current independent variable values. These 46 functions are used by Equations (IV-50) through (IV-55) to derive the total coefficient values. The total coefficient values are dimensionalized in the derivative region then sent to the analog region and placed on SIMSTAR #114s analog ports leading to SIMSTAR #115.

Dynamics Model Routine

The aircraft dynamics model routine acts as the plant of the overall system and resides in the analog region of SIMSTAR #115. The routine takes as its inputs the total coefficient values determined by the interpolation and lookup routine and uses them in the equations of motion to determine the aircraft output variables. The aircraft output variables include those variables necessary to be fed-back through the flight control

system, the variables for the table lookup routine on SIMSTAR #114, and the variables for the cockpit display. The equations of motion described in Chapter IV are computed in their complete nonlinear form. The analog region of the SIMSTAR is an analog computer. This region performs all mathematical equations simultaneously. The SIMSTAR's uniqueness lies in the fact that all mathematical equations and interconnections are specified in software. For instance, an integration is specified by the command $Y = \text{INTEG}(X, Y_0)$ which performs an integration of the variable X with initial value Y_0 and the result becomes the variable Y .

Two different dynamic models are written for this thesis. The first is a completely uncoupled dynamics model separating longitudinal and lateral modes. The second is a fully coupled dynamics model including all nonlinearities of the full equations of motion. It should be noted that though two different dynamic models are used, only one interpolation and look-up routine is used and as such, any coupling incorporated in the aerodynamic data base can affect the system if both inertially decoupled longitudinal and lateral-directional modes are simulated simultaneously.

Also included, as part of the dynamics model, are the first order lag filters representing the servoactuators of the control surfaces. These filters accept values from the aircraft control system representing the control surface deflections.

Though the atmospheric model is part of the dynamics model, it is computed in SIMSTAR #114s analog region due to a lack of analog component space in SIMSTAR #115. The atmospheric model is used only by the interpolation and look-up routine therefore this task is best performed in SIMSTAR #114.

Control System Design Routine

The Digital Flight Control System (DFCS) routine resides in the digital derivative region of SIMSTAR #115 and is written in FORTRAN. This routine is provided by the user as a compensator for the aircraft. The routine takes as its inputs those feedback aircraft output variables and pilot input signals required for flight control law computation and outputs the control surface deflection commands to be sent to the interpolation and look-up routine.

If the user has a design in transfer function form defined in either the s - or z -domain the approach to coding is simple. The user converts his s or z -domain functions into discrete time difference equations keeping track of past values. Then the user codes these equations into the FORTRAN routine, DFCS, in an algorithm as dictated by the use of variables in the user's block diagram. At the end of the coded algorithm the user must assign all new calculated output values of the difference equations to variable names which represent the values at the previous computation time. This is so that at the next time interval the new values can be used as the previous old values in the difference equations. This is shown

in Chapter IV where the backwards difference equation is discussed. In the INITIAL region of SIMSTAR #115, the user sets the output values of the difference equations initially to zero. Also, the user specifies the coefficients of the difference equations, which are based on the sampling time of the control system in a FORTRAN routine called INIT. The aircraft control system code accesses the data in INIT by a COMMON block. The INIT routine which resides in the INITIAL region reduces some computation time and allows easy changing of the sampling time. The sampling time is specified by a combination of variables PERIOD, CINT and BET which are accessible to the user. The steps to coding an aircraft control system design are completely explained in Appendix A, the SIMTACS Users' Manual.

Cockpit Display Routine

This routine resides in the analog region of SIMSTAR #114. The oscilloscope, used as the first version of the display, is placed in an X-Y mode such that two in-phase sinusoids can be placed on both channels to produce a straight line. The slope of this straight line can be varied by changing the relative magnitudes of these sinusoids. This line becomes a roll indicator for the pilot (see Figure 5.3). A third channel is used for a moving horizon indicating change in pitch (see Figure 5.3). This is done simply by sending to the third channel the negative of the pitch signal, θ , generated by the dynamics model.

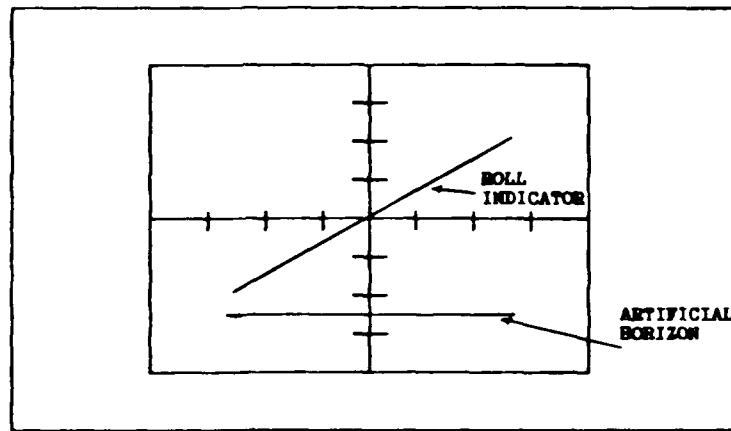


Figure 5.3. Oscilloscope Cockpit Display

The magnitude of the sinusoids for the roll indicator is based on the sine and cosine functions of the roll angle, Φ , generated by the dynamics model. A simple undamped oscillator using two integrators and a gain generates the sinusoids for the display. Because the same oscillator generates both sinusoids, the signals to the oscilloscope are always in phase. The cosine of roll is generated and multiplied by the oscillator output to derive the sinusoid for the X channel and the sine of roll is multiplied by the oscillator output to produce the Y channel sinusoid. This provides ± 360 degrees of freedom for the roll indicator. The frequency of the oscillator is arbitrarily set to 100 hertz which is sufficiently high to provide a steady line on the oscilloscope screen.

Implementation of Routines

The overall implementation is shown in Figure 5.2. This figure shows where the tasks previously discussed in this chapter, reside and what variables are being passed between them. Of course, the user has access to change the variables between the dynamics model and the control system so that feedback variables specific to his design can be used. The routines are set up so the user can easily find the various parts of the code. At run-time, the user is also able to change any initial flight conditions or trim the aircraft to a desired flight condition. The user must be aware of the limitations of the data base, so that a condition is not entered for which the data base cannot generate data.

This chapter described the software design of the simulator. Once the simulator is designed, validating the aircraft model and evaluating the simulator's performance are accomplished. The next chapter explains this procedure.

VI. Simulator Testing and Verification

Overview

Once the simulator is designed, the responses of the aircraft are compared to verified aircraft responses from other sources. The performance of the simulator as a research tool for man-in-the-loop testing is also examined.

Validating Aircraft Model

The F-16A aircraft and the simplified digital flight control system described in Chapter III are validated by comparing closed-loop time response plots to those of the aircraft true responses (19). Also, common sense is used in determining the reasonableness of the responses based on the dynamics of the true aircraft. Plots are obtained for both the inertially cross-coupled (Figures 6.1-6.4) and the uncoupled aircraft (Figures 6.5-6.9). The inputs are the roll stick, pitch stick and rudder pedal inputs which result in commanded inputs of roll rate, normal acceleration and sideslip respectively. The responses are for a trim flight condition of Mach number of 0.9 and altitude of 20,000 feet for one-g, wings level, non-accelerating flight.

The mass properties and the initial conditions of the F-16A aircraft for the following strip chart plots are shown in Table III.

Inertially cross-coupled aircraft time responses. These responses are shown in Figures 6.1 through 6.4. Figure 6.1 is

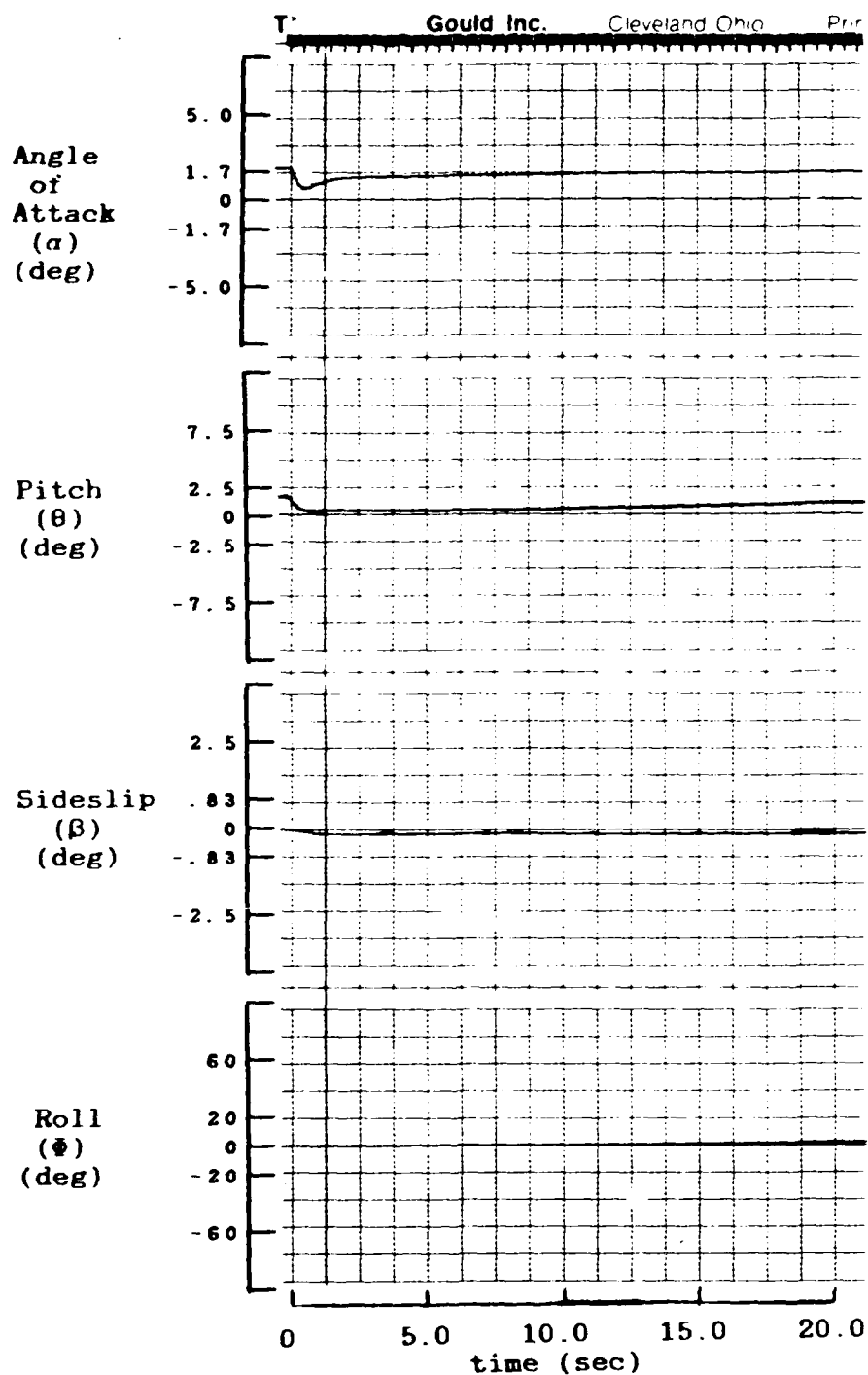


Figure 6.1a. No Input - Trimmed Condition, Coupled Responses

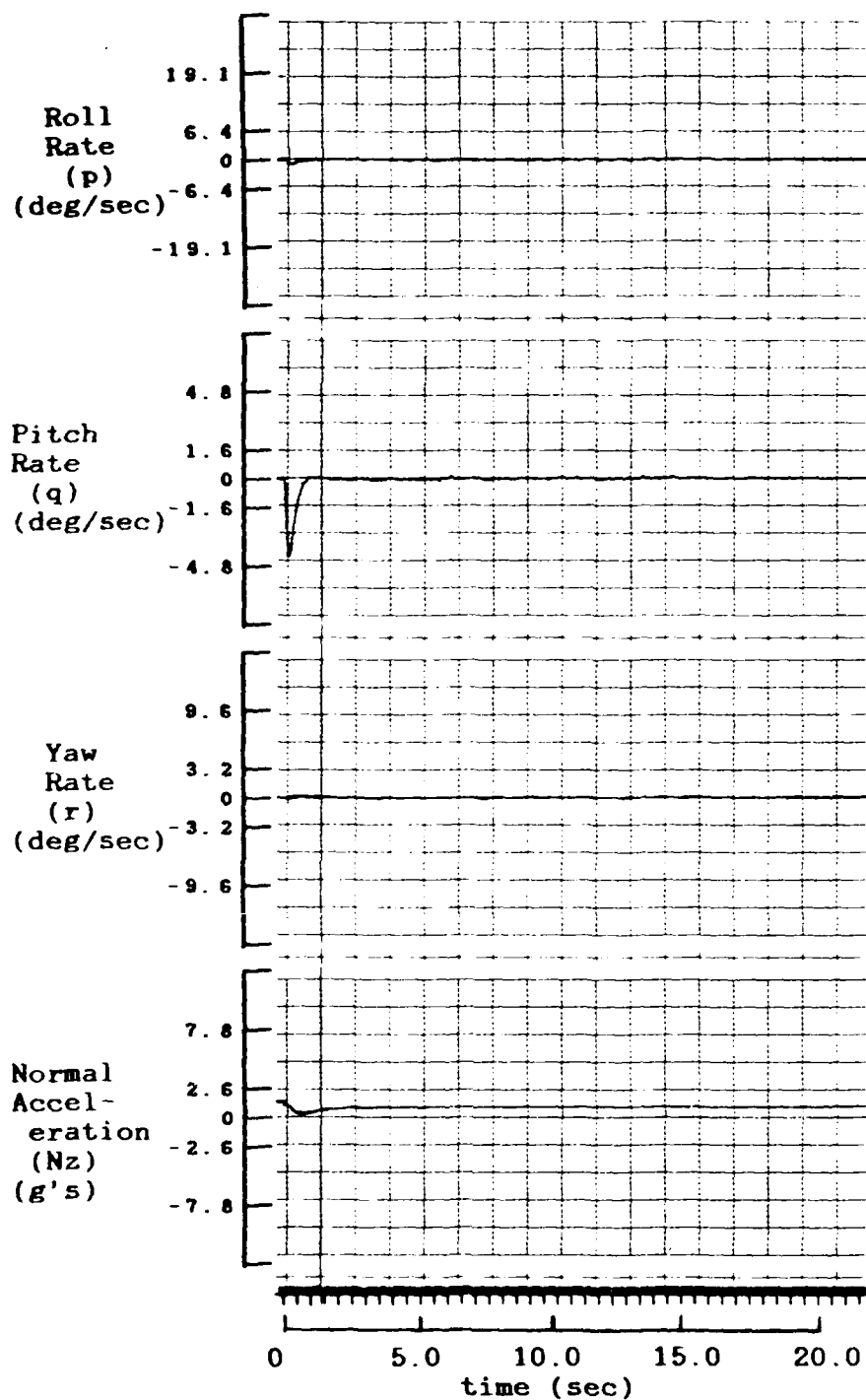


Figure 6.1b. No Input - Trimmed Condition, Coupled Responses

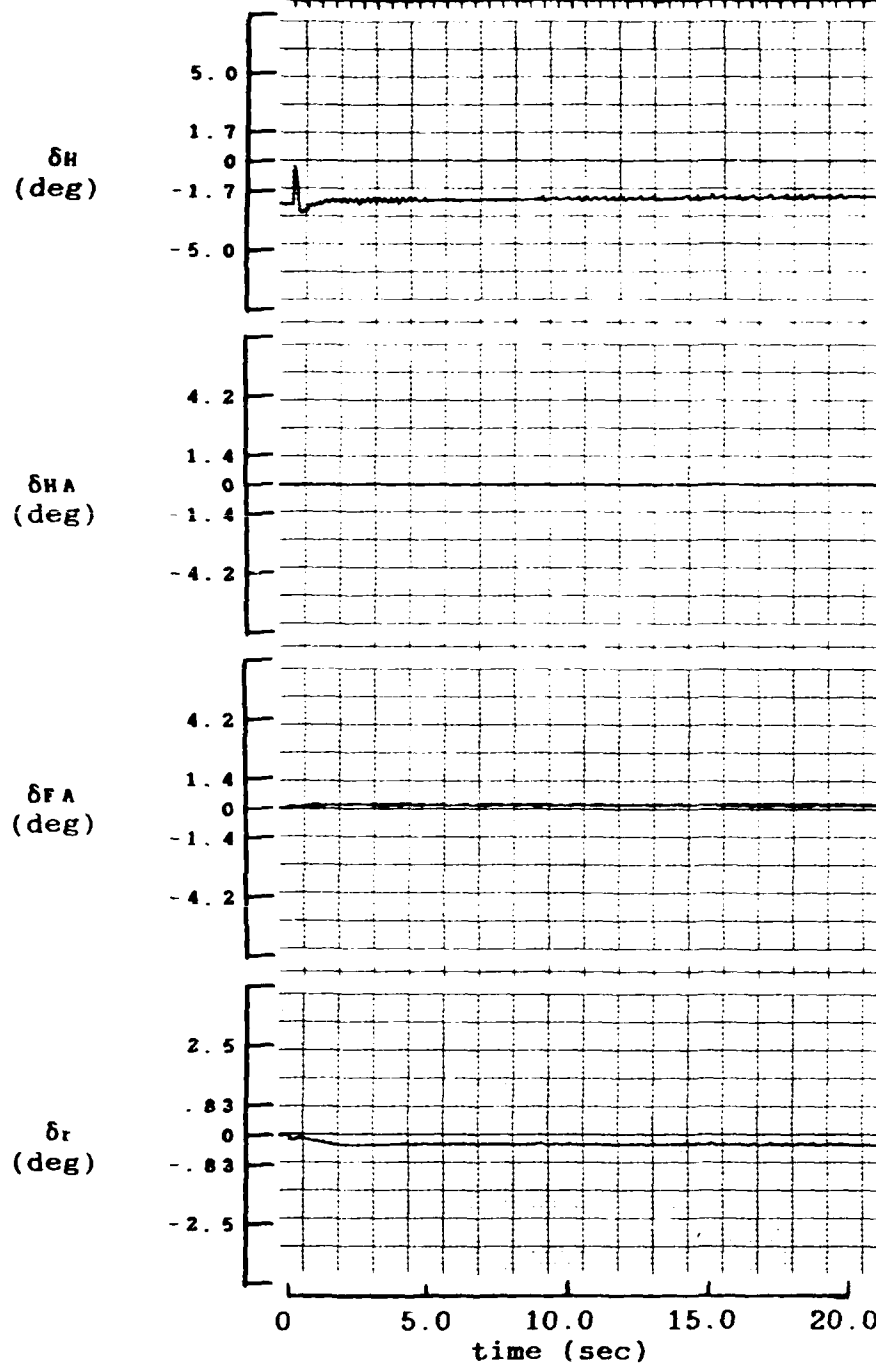


Figure 6.1c. No Input - Trimmed Condition, Coupled Responses

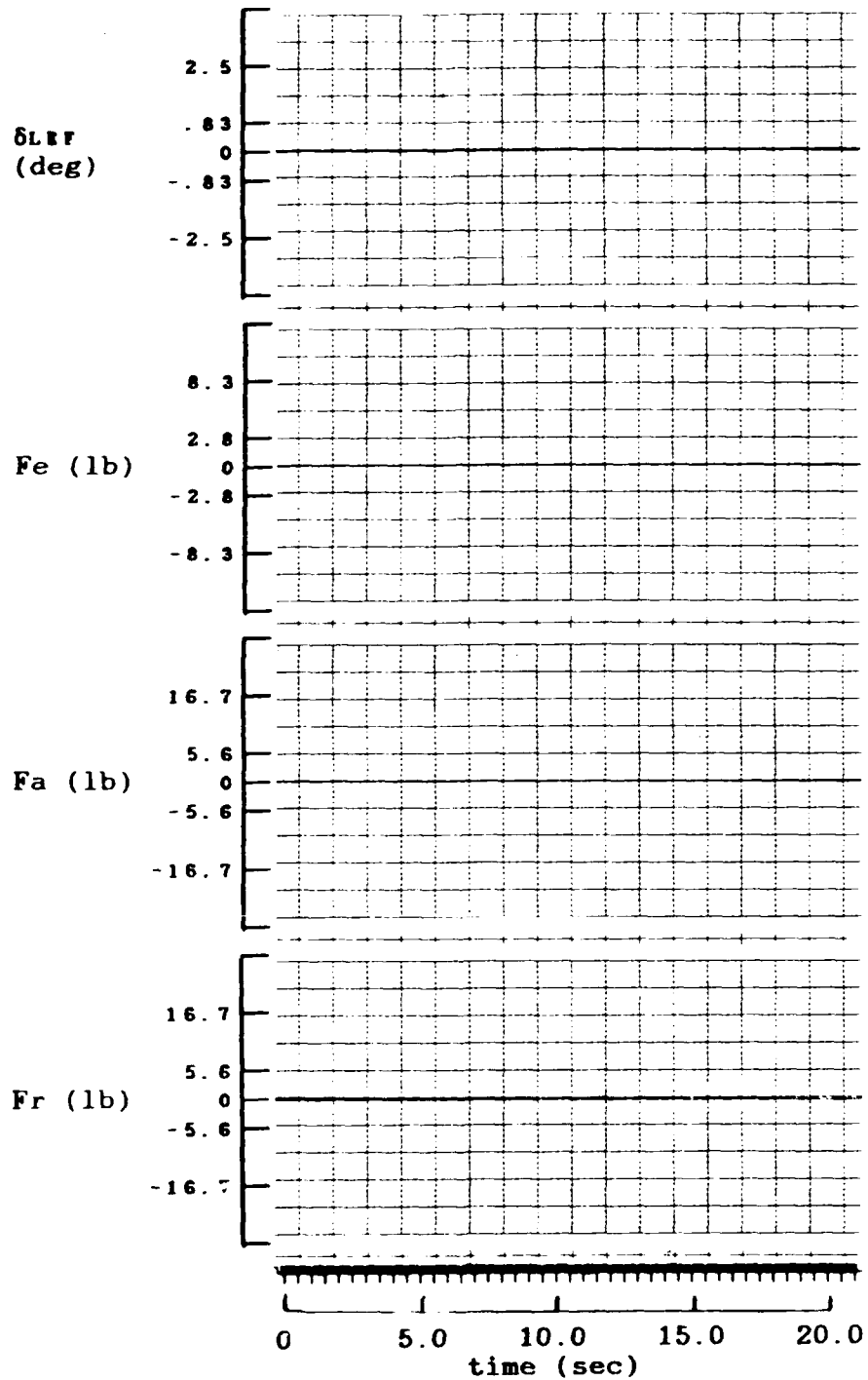


Figure 6.1d. No Input - Trimmed Condition, Coupled Responses

	F-16A	units
weight	23555	lb
I _x	10691	slug-sq ft
I _y	62749	slug-sq ft
I _z	69803	slug-sq ft
I _{xz}	-14	slug-sq ft
CG	0.35	
Mach number	0.9	
Altitude (h)	20000	feet
δH	-2.8	degrees
δLEF	0	degrees
AOA (α)	1.9	degrees
pitch (θ)	1.9	degrees
sideslip (β)	0	degrees

Table III. Mass properties and initial conditions used in time response plots.

a time response of the aircraft settling to the trimmed condition of one-g, level flight when no input is applied. If no input is applied to the aircraft then one should expect the flight control system to bring the aircraft to an equilibrium flight condition. In this case, the equilibrium condition is the chosen flight condition. As shown by the figure, the flight condition is reached in approximately two seconds after starting the run. It is important to have the aircraft at the flight condition at the time the inputs are applied.

Figure 6.2 is a time response of the aircraft executing, from trimmed level flight, a one-g pull-up maneuver followed ten seconds later by a return of the stick to the trim position. The F-16A starts at the trimmed one-g condition and

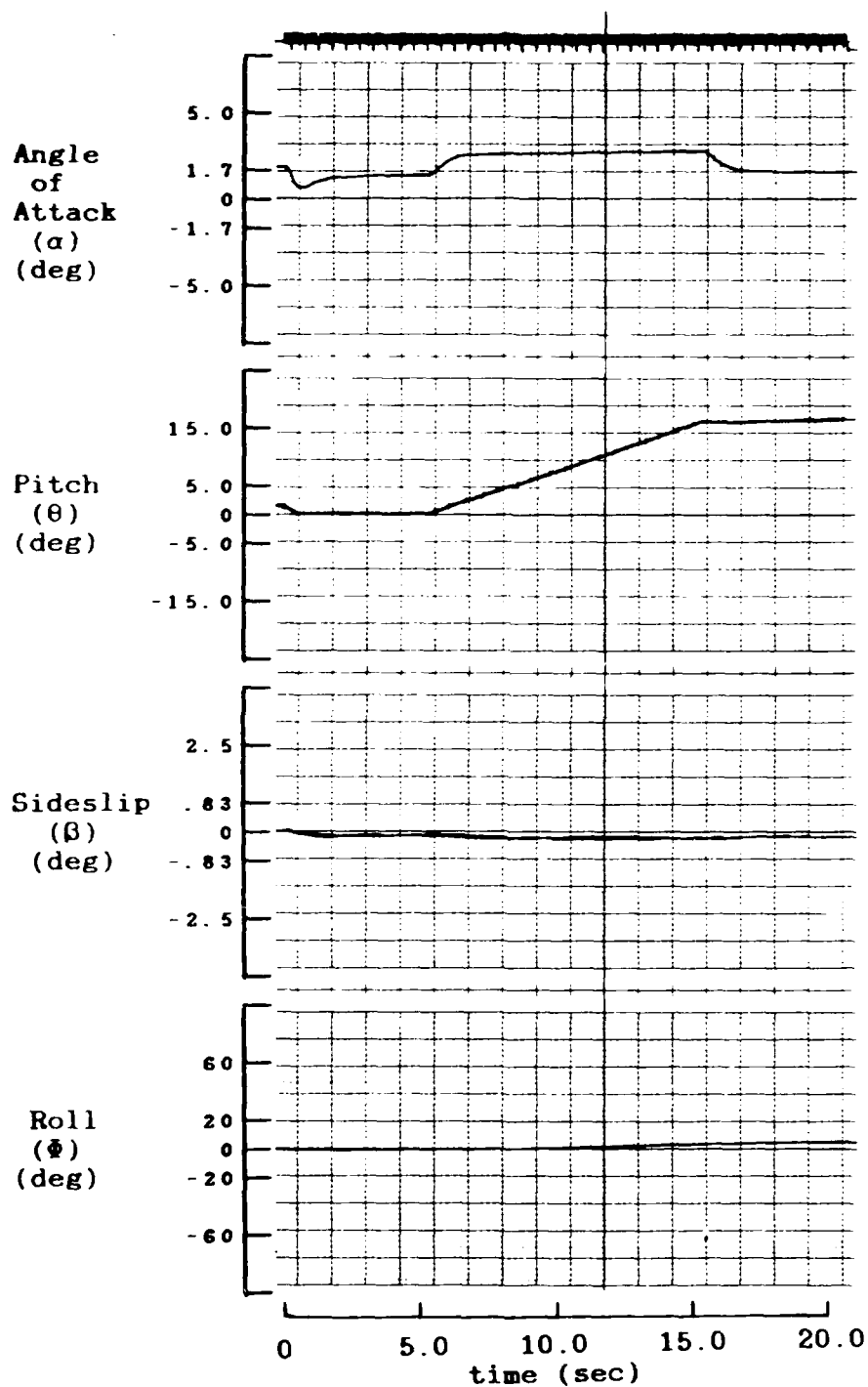


Figure 6.2a. One-g Pitch Stick Input, Coupled Responses

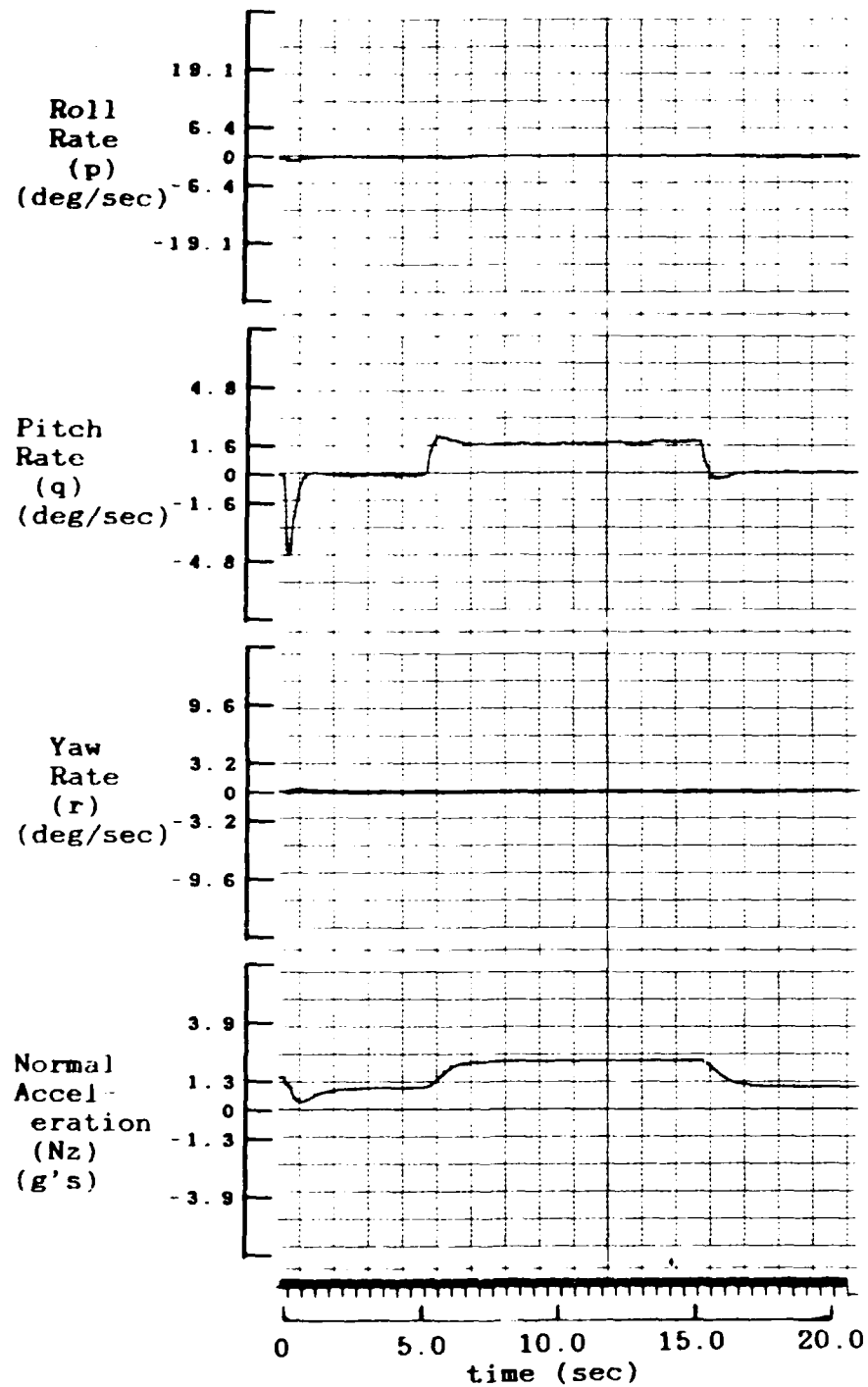


Figure 6.2b. One-g Pitch Stick Input, Coupled Responses

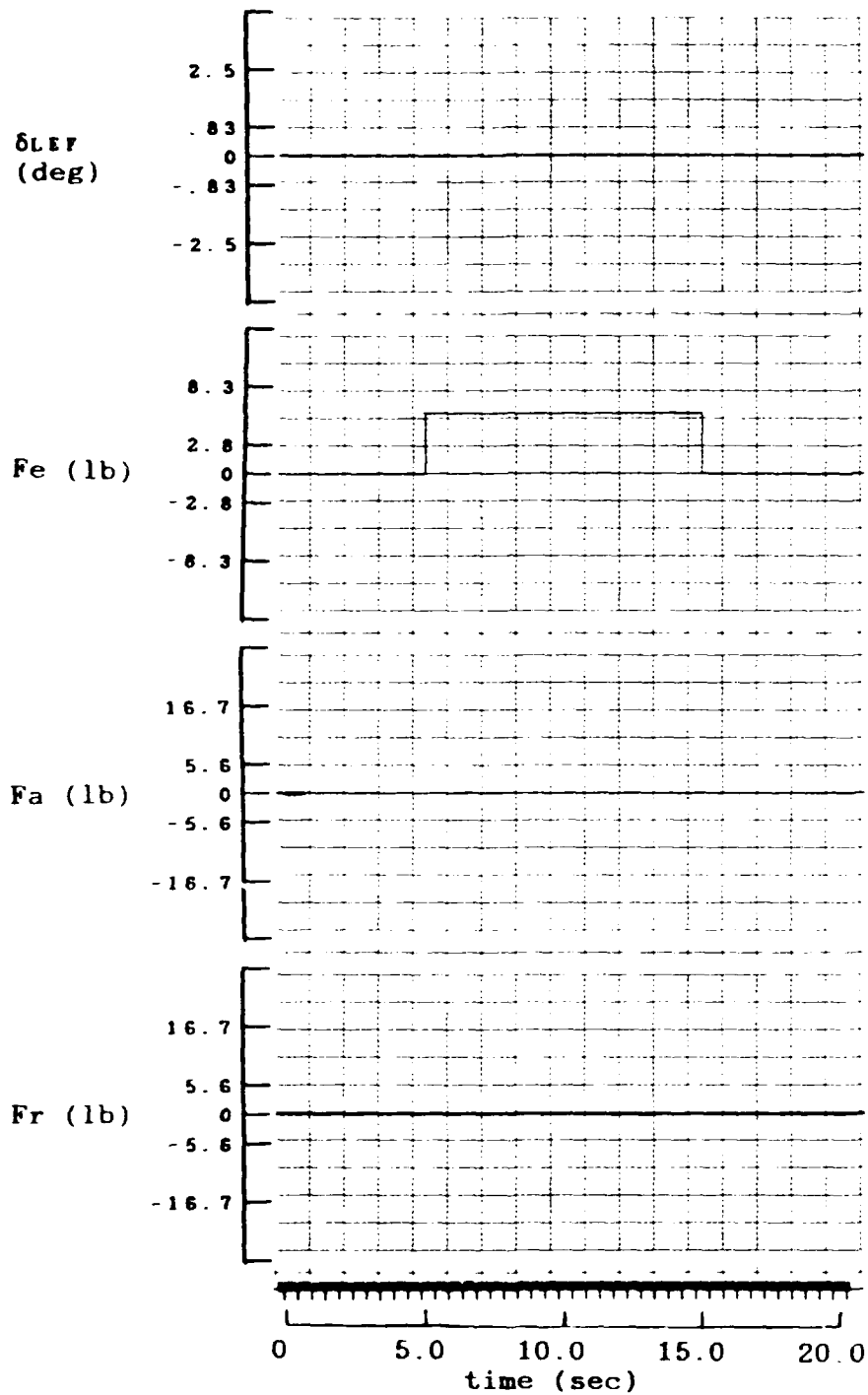


Figure 6.2c. One-g Pitch Stick Input, Coupled Responses

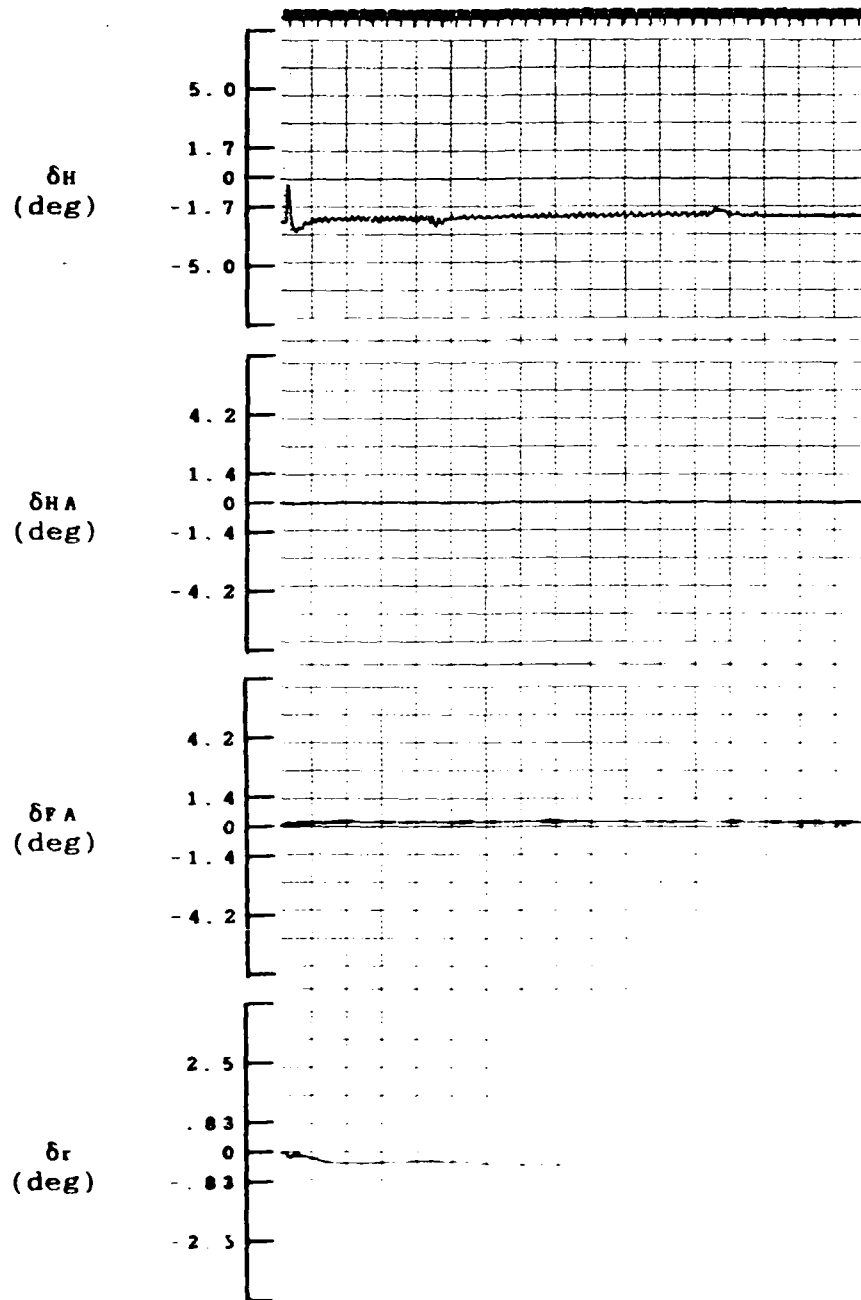


Figure 6.12

AD-A189 675

F-16 SIMULATOR FOR MAN-IN-THE-LOOP TESTING OF AIRCRAFT
CONTROL SYSTEMS (SIMTACS) (U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI... M W KASSAM
DEC 87 AFIT/GE/ENG/87D-30 F/G 1/4

2/2

UNCLASSIFIED

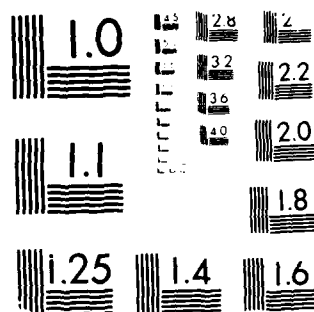
NL

END

DATE

4 8 8

FILE



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

one second after the input is applied, steady state is reached at two g's. Likewise when the pitch stick is released, the F-16A behaves as one would expect with a quick response down one g to the original one-g condition. The angle of attack quickly rises to a steady state value after the input is applied, as the pitch ramps up, which is expected in a pull-up maneuver. The slight overshoot in the pitch rate response is typical for pitch rate responses for this aircraft. One might note the distinct biases in the rudder and flaperon deflections. This is due to yaw and roll trim set by the author to achieve the level flight condition. These biases appear throughout the coupled aircraft responses and do not affect the performance of the aircraft.

The time response plots of the F-16A executing a commanded 45 deg/sec roll rate is shown in Figure 6.3. The roll rate appears to respond very quickly to the input and reaches a steady state value of about 32 deg/sec. The reason the roll rate does not reach the commanded 45 deg/sec is because the flight control system in the roll mode is a type zero system, and thus cannot accurately track the input. One of the things a pilot would like to accomplish while rolling is keeping himself in a one-g environment. The flight control system assists him in this task. The flight control system increases the angle of attack because the normal acceleration of the pilot is not located at the center of gravity.

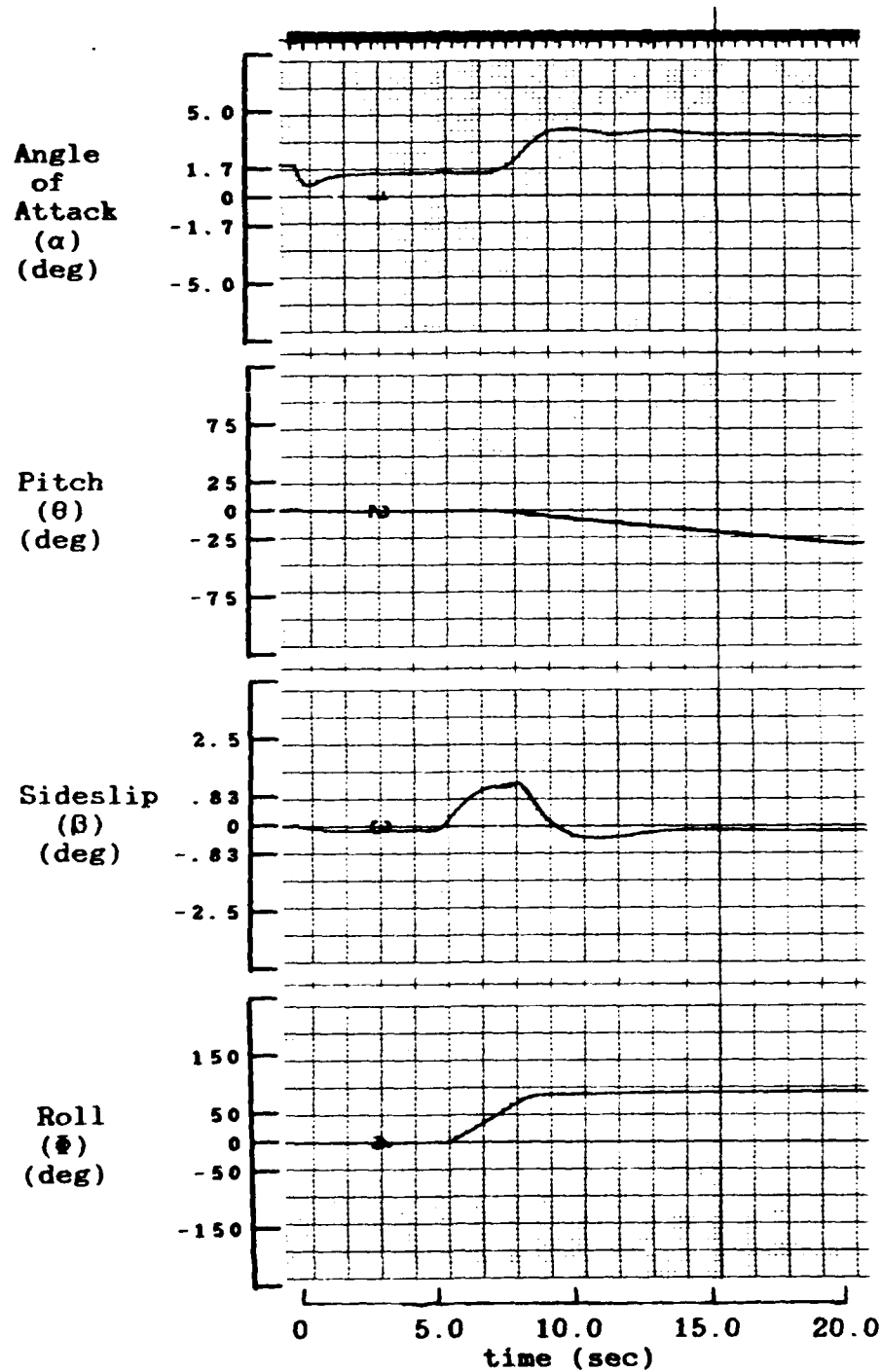


Figure 6.3a. 45 deg/sec Roll Stick Input, Coupled Responses

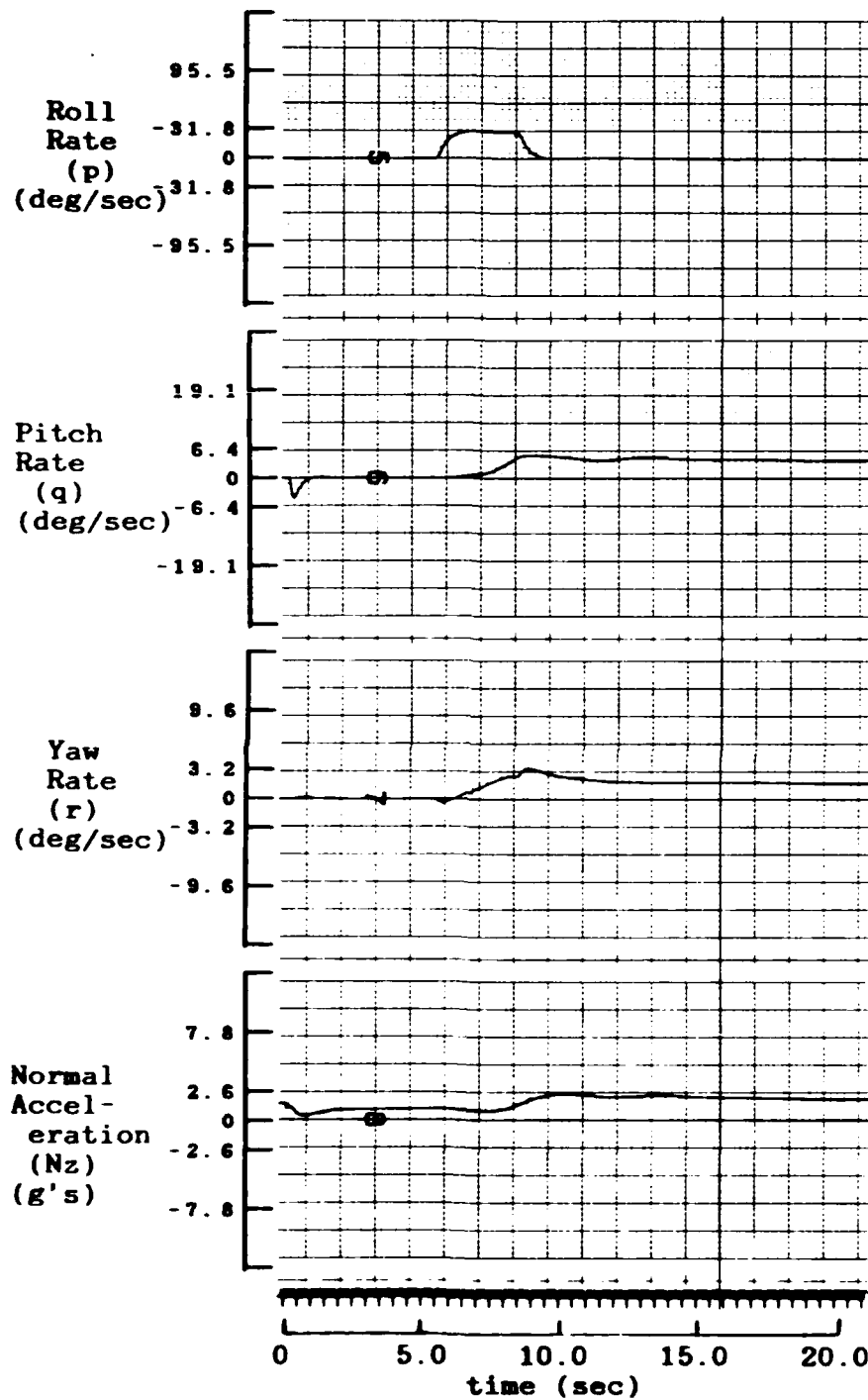


Figure 6.3b. 45 deg/sec Roll Stick Input, Coupled Responses

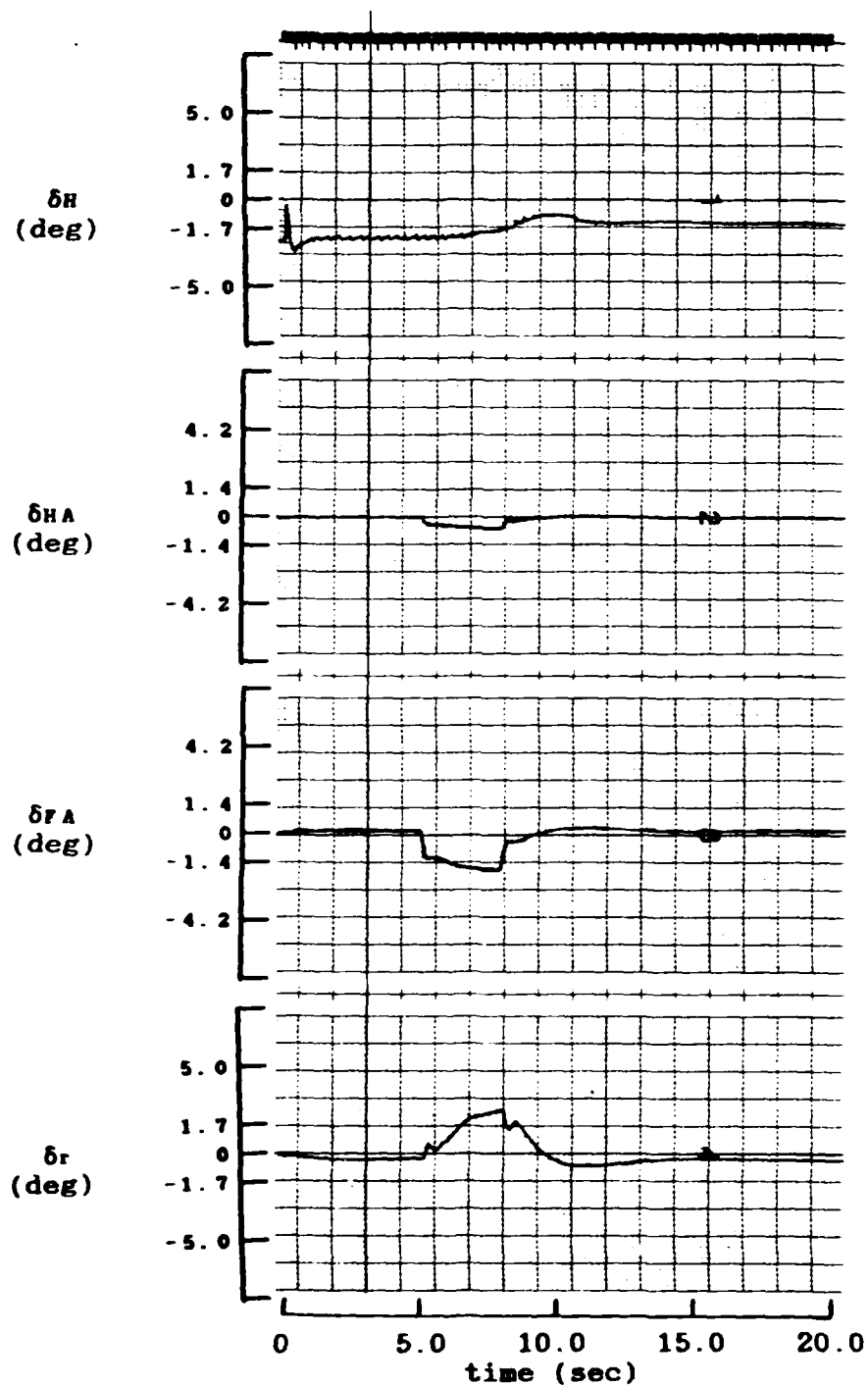


Figure 6.3c. 45 deg/sec Roll Stick Input, Coupled Responses

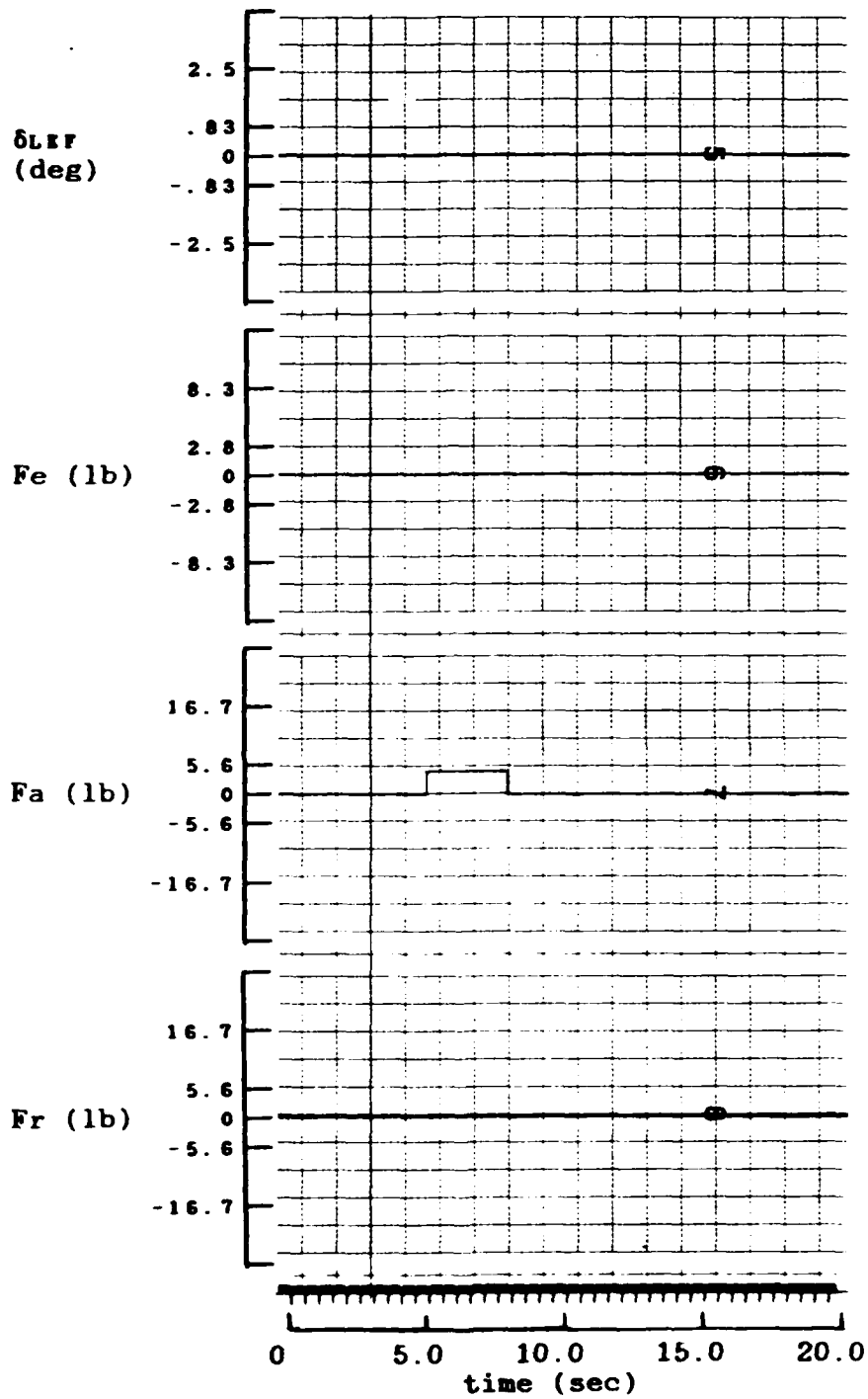


Figure 6.3d. 45 deg/sec Roll Stick Input, Coupled Responses

The control system is also designed to roll the aircraft about the stability axis thereby introducing a yaw rate and sideslip. Because the input is a positive roll rate, a positive roll angle is expected. Also, because the aircraft is rolling about the stability axes, the control system commands a positive yaw rate as the aircraft rolls. The sideslip is introduced mainly due to the contribution of side force during the maneuver. The maneuver ends with the aircraft in a constant bank angle of about 90 degrees, therefore the yaw rate remains positive and the pitch continues to decrease due to the lack of lift to hold the aircraft nose up.

The time response plots of the F-16A executing a commanded negative one deg of sideslip is shown in Figure 6.4. The convention for inputting a positive rudder pedal force is right pedal forward. This results in the nose of the aircraft pointing to the right. The rudder must deflect negatively to point the nose of the aircraft right. This produces positive yaw rate which gives negative sideslip. This is shown in Figure 6.4. Note that the pedal force is shown negative. This error is due to a reversed polarity setting on the strip chart for this variable.

As the nose of the plane points right, the left wing moves into the direction of the forward velocity vector. This creates extra lift under this wing causing the aircraft to roll toward the right wing down direction. This is a positive roll angle as shown in Figure 6.4. Due to a washout filter in the

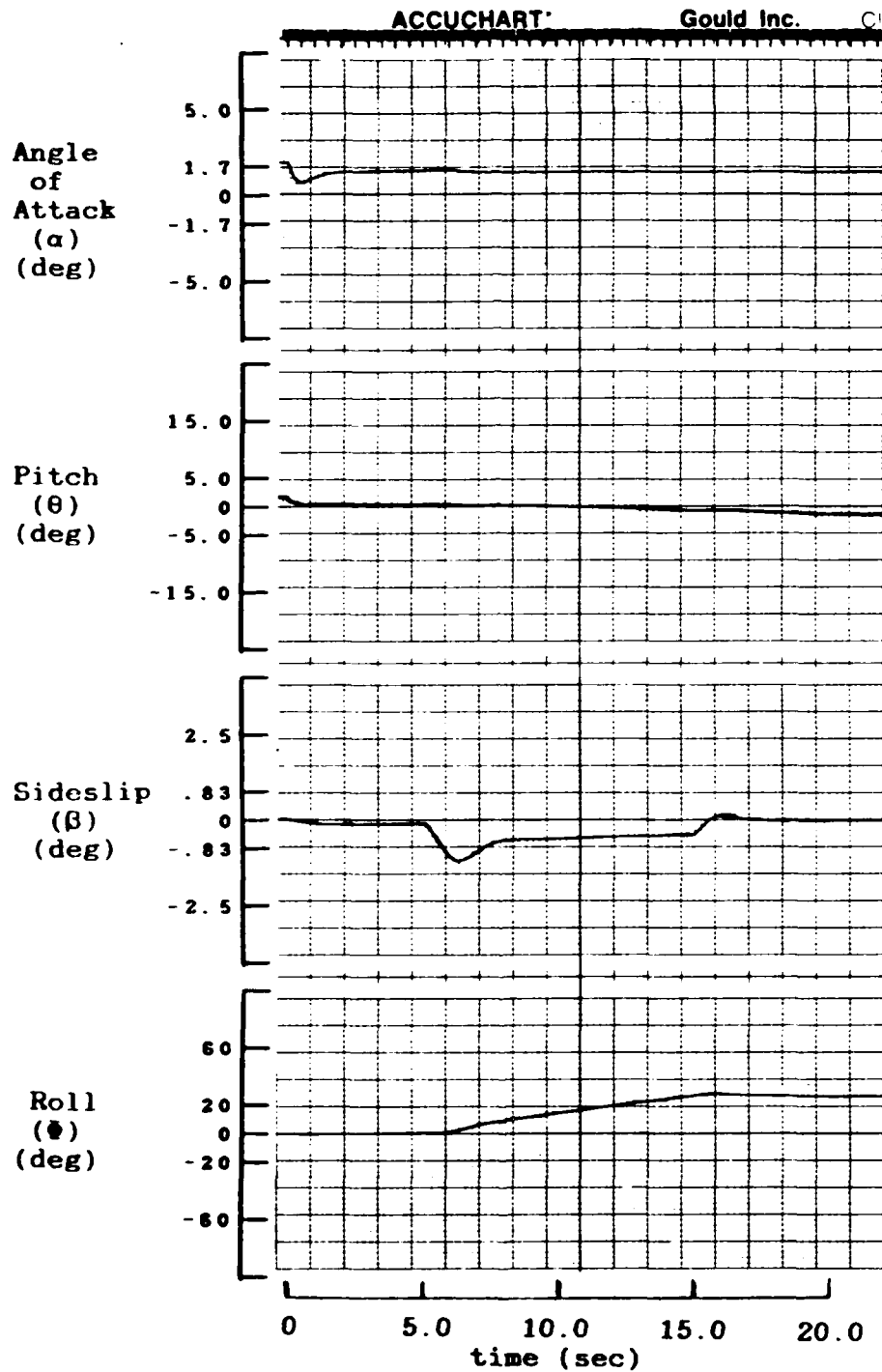


Figure 6.4a. One deg Rudder Pedal Input, Coupled Responses

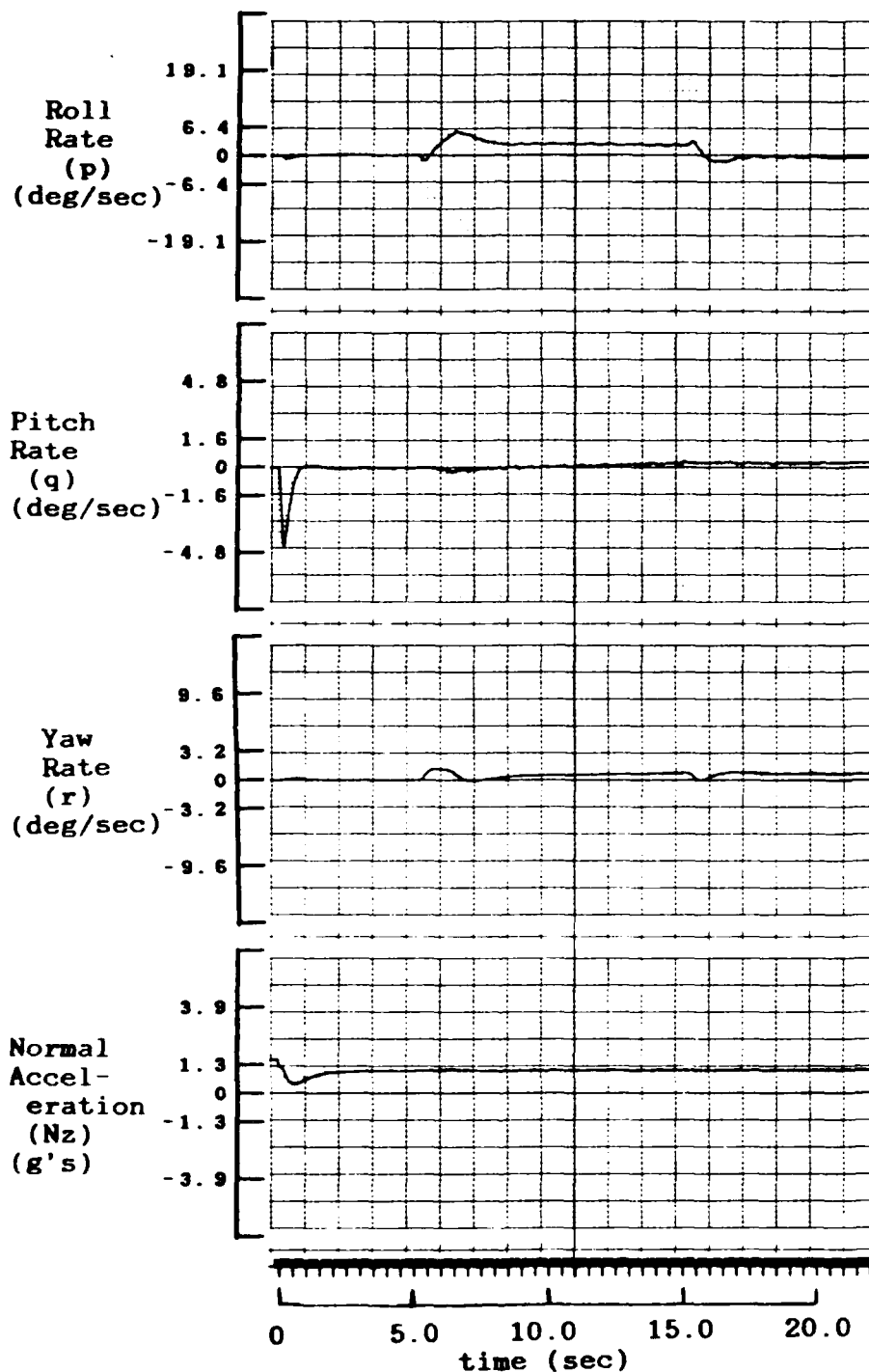


Figure 6.4b. One deg Rudder Pedal Input, Coupled Responses

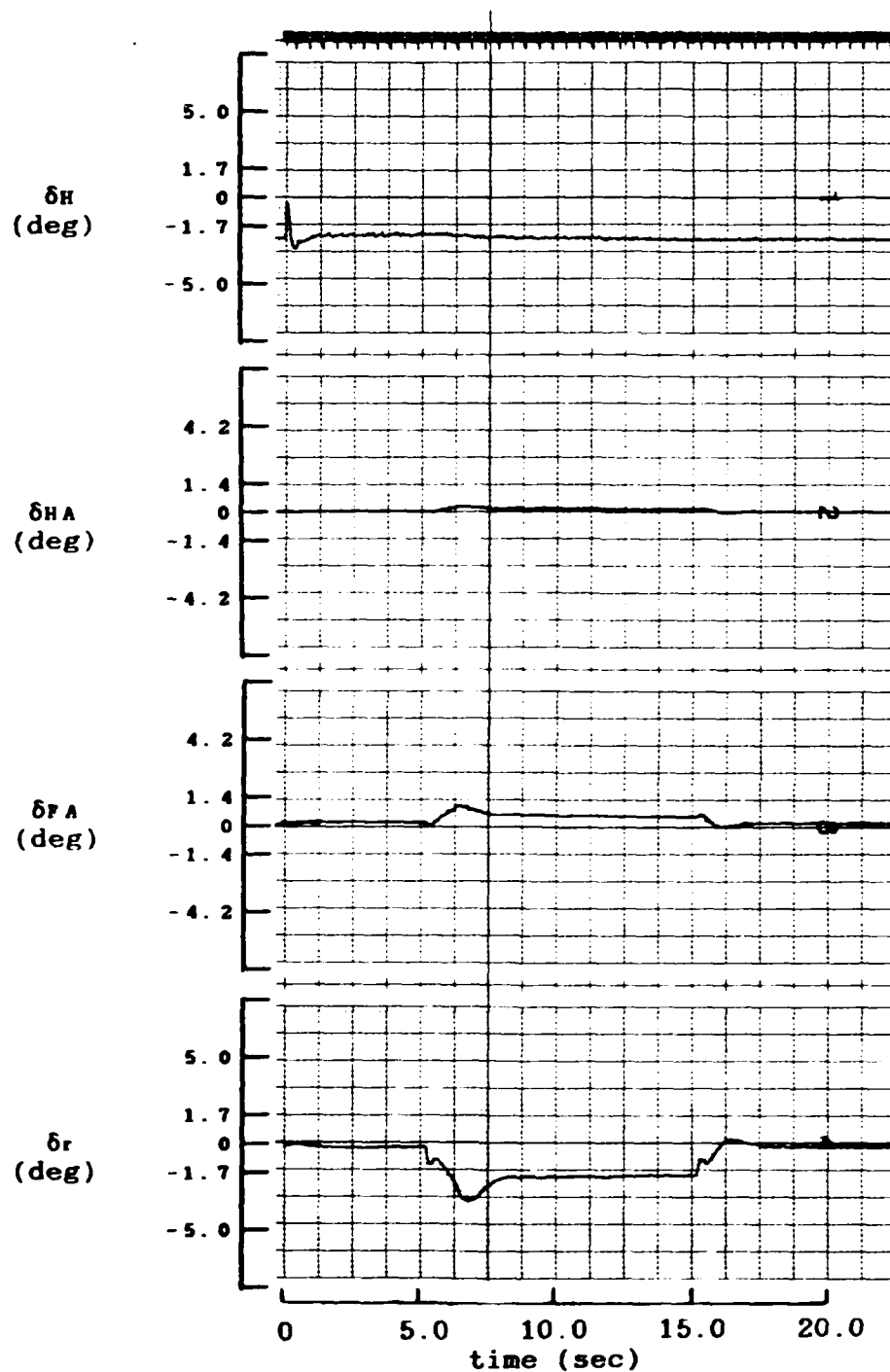


Figure 6.4c. One deg Rudder Pedal Input, Coupled Responses

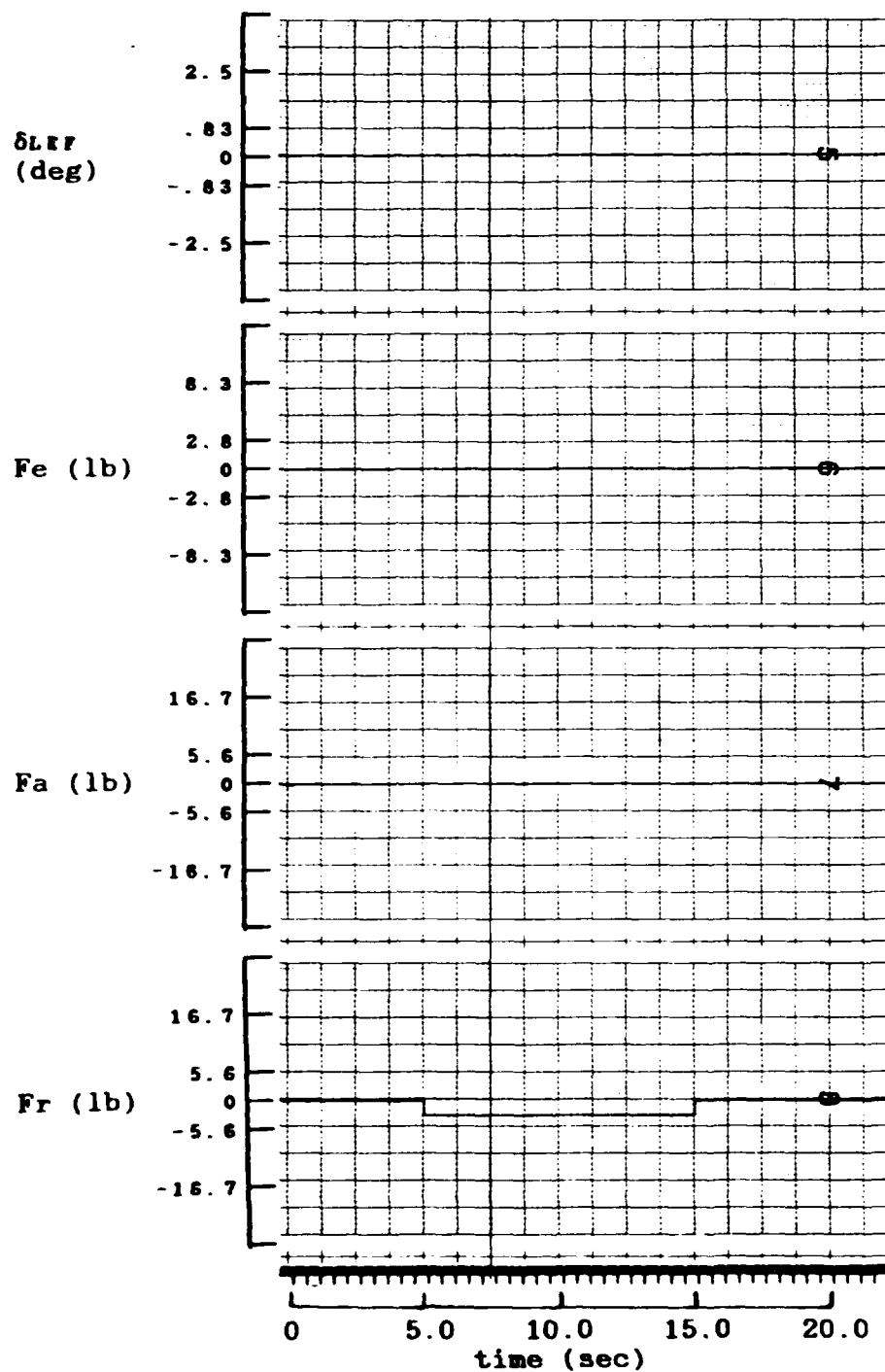


Figure 6.4d. One deg Rudder Pedal Input, Coupled Responses

control system, yaw rate is effectively reduced after about two seconds. Apparently because yaw acceleration is incorporated into the moment equations, it creates the nonlinear hump displayed in the roll rate response. This roll rate affects the deflection of the rudder through the flight control system coupling and produces the large overshoot shown in the sideslip response. The gradual decrease of the sideslip towards zero is due to the gradual increase in the yaw rate as the aircraft slowly rolls about the stability axes.

Uncoupled aircraft responses. The major reason for including these plots as part of this thesis is to provide an additional comparison for the inertially cross-coupled plots described earlier. These plots of Figures 6.5 through 6.8 are uncoupled aircraft responses with inputs applied which are similar to those applied to the inertially cross-coupled aircraft. Table III provides information for these plots.

Figure 6.5 shows that the aircraft is indeed at the trimmed flight condition of one-g in level flight at Mach number of 0.9 and altitude of 20,000 feet.

The responses of Figure 6.6 due to a commanded one-g input are very similar to the inertially cross-coupled plots of Figure 6.2. It is clear that the inputs affecting the longitudinal mode of the aircraft do not have any affect on the lateral-directional mode. This is only true if the trim condition of the aircraft is that of level flight. If

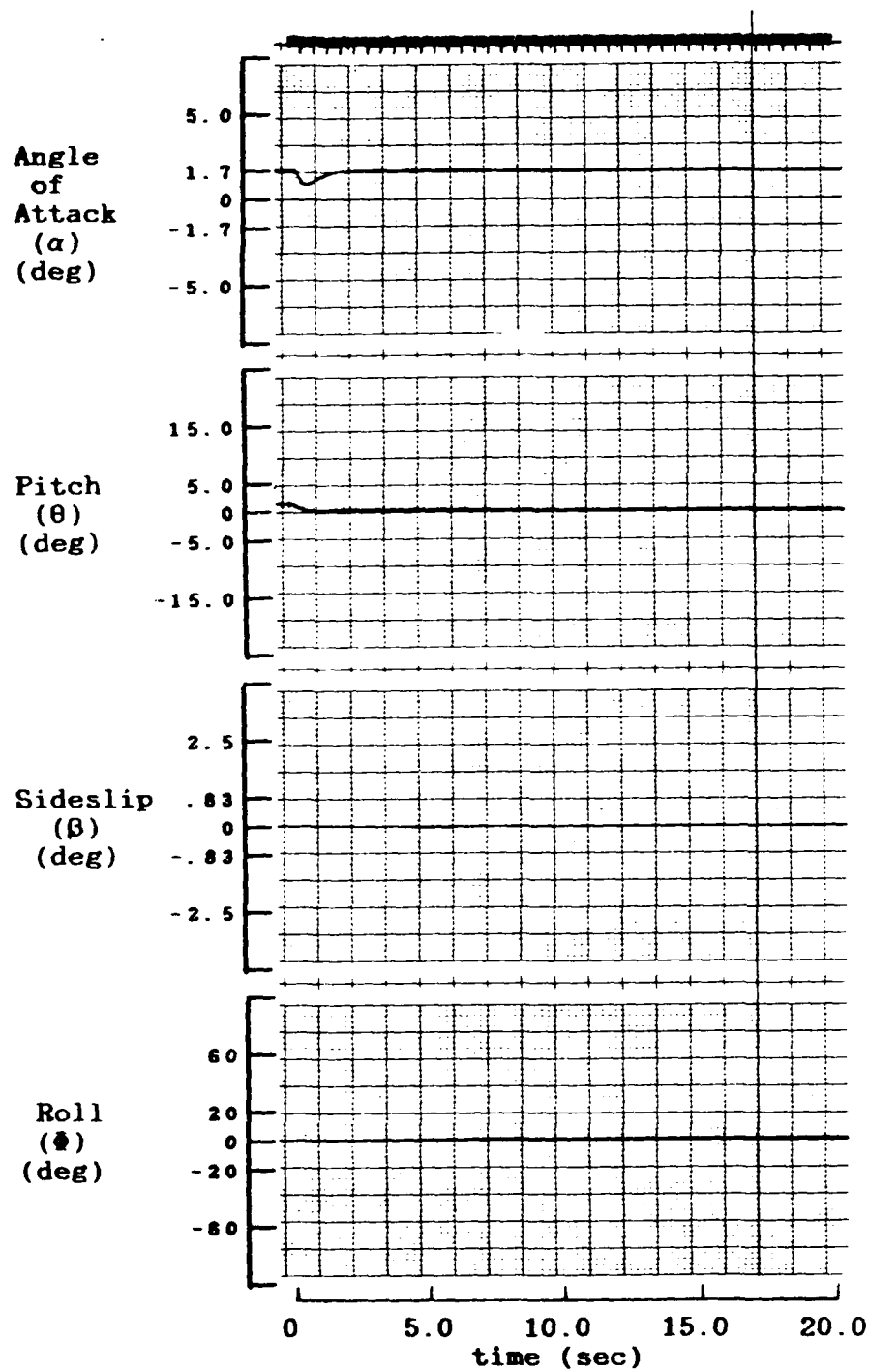


Figure 6.5a. No Input - Trimmed Condition, Uncoupled Responses

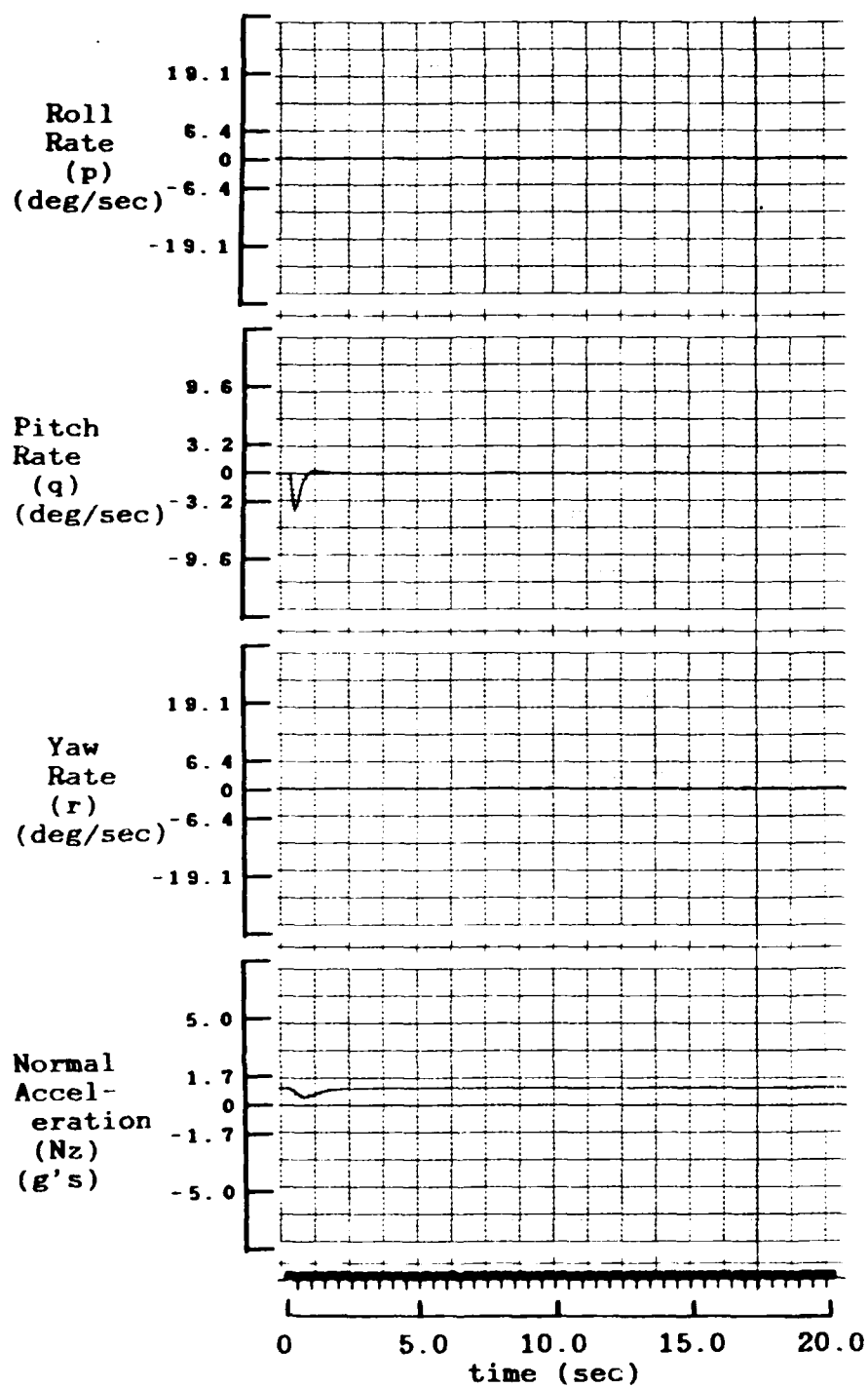


Figure 6.5b. No Input - Trimmed Condition, Uncoupled Responses

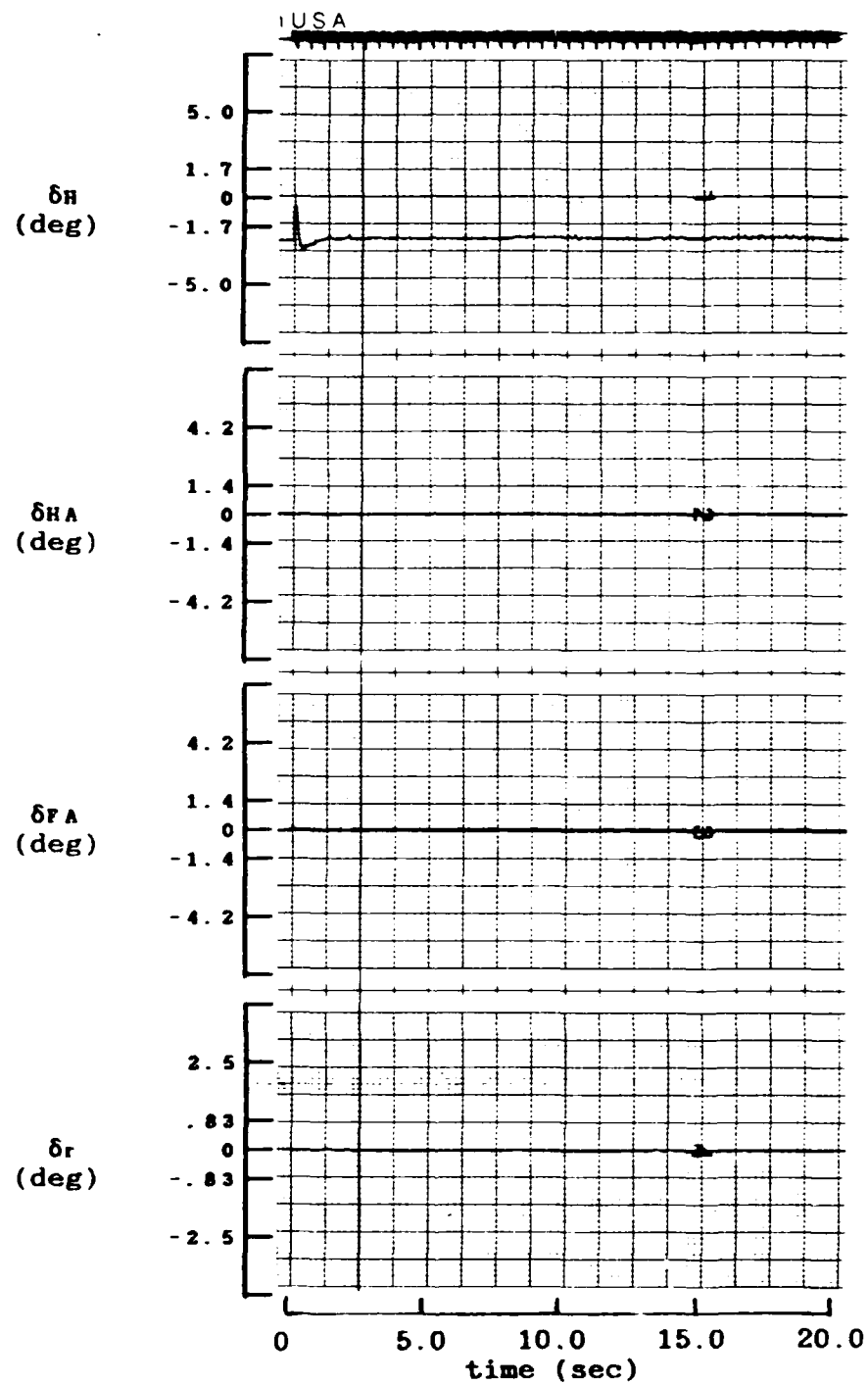


Figure 6.5c. No Input - Trimmed Condition, Uncoupled Responses

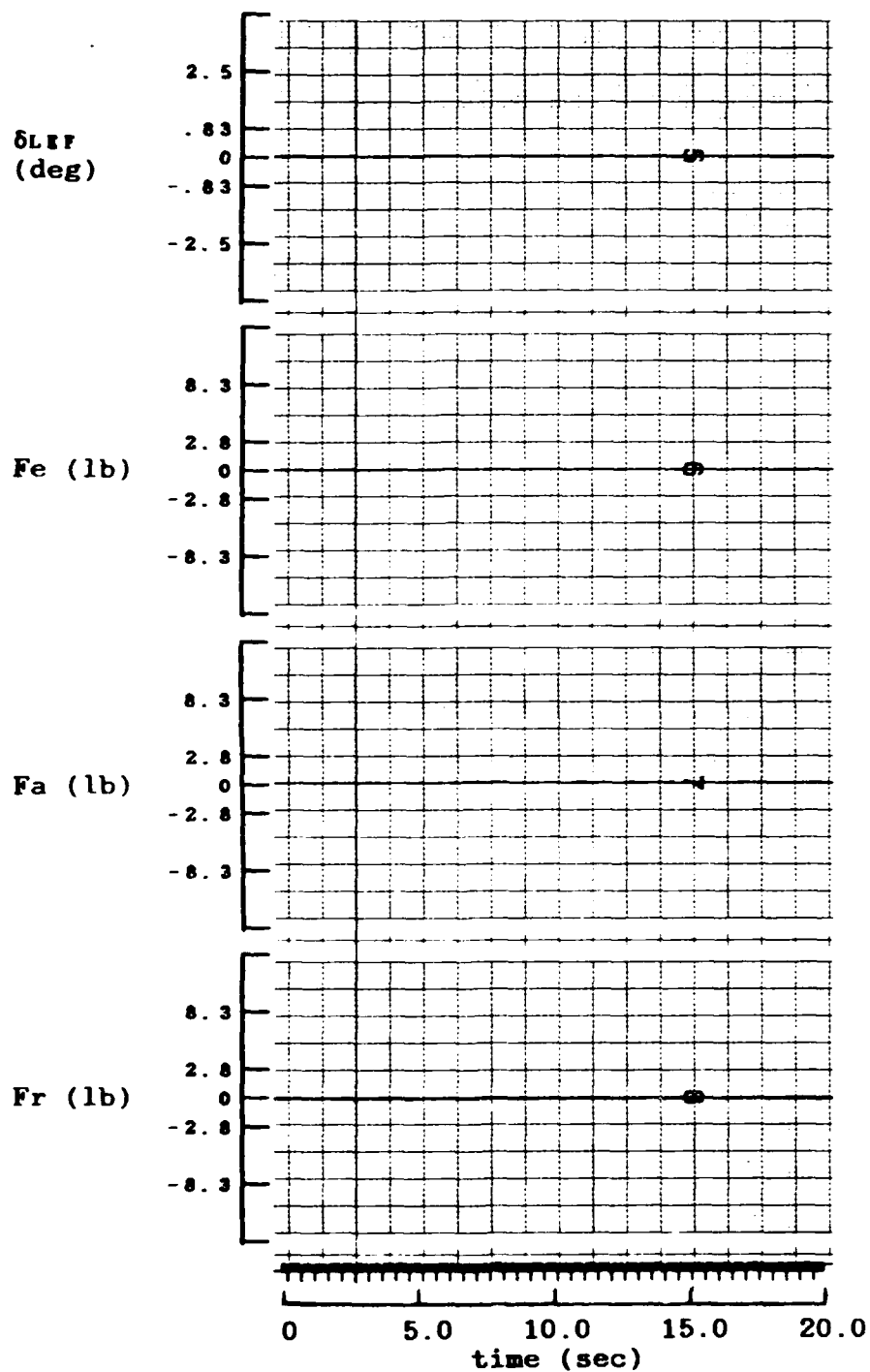


Figure 6.5d. No Input - Trimmed Condition, Uncoupled Responses

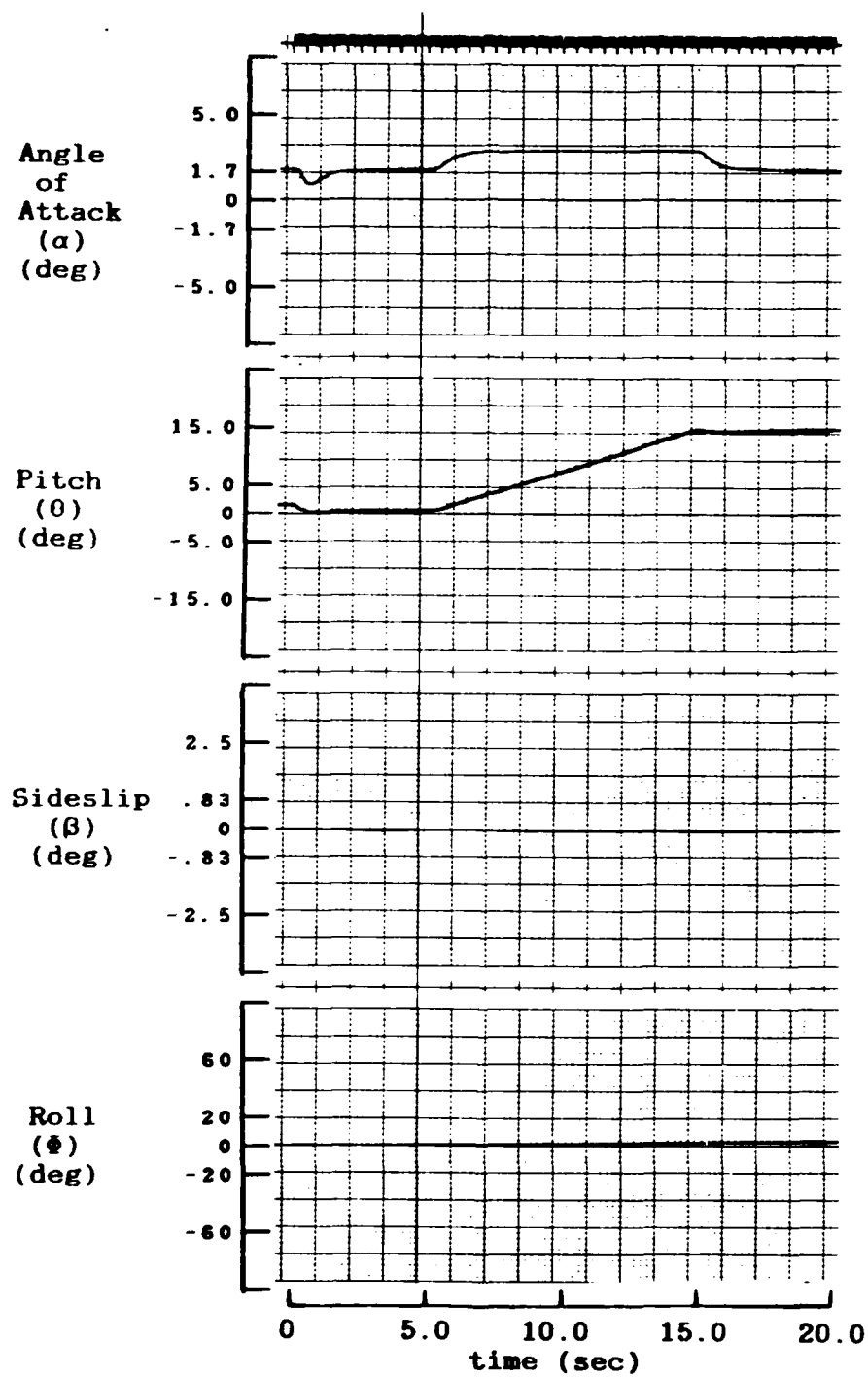


Figure 6.6a. One-g Pitch Stick Input, Uncoupled Responses

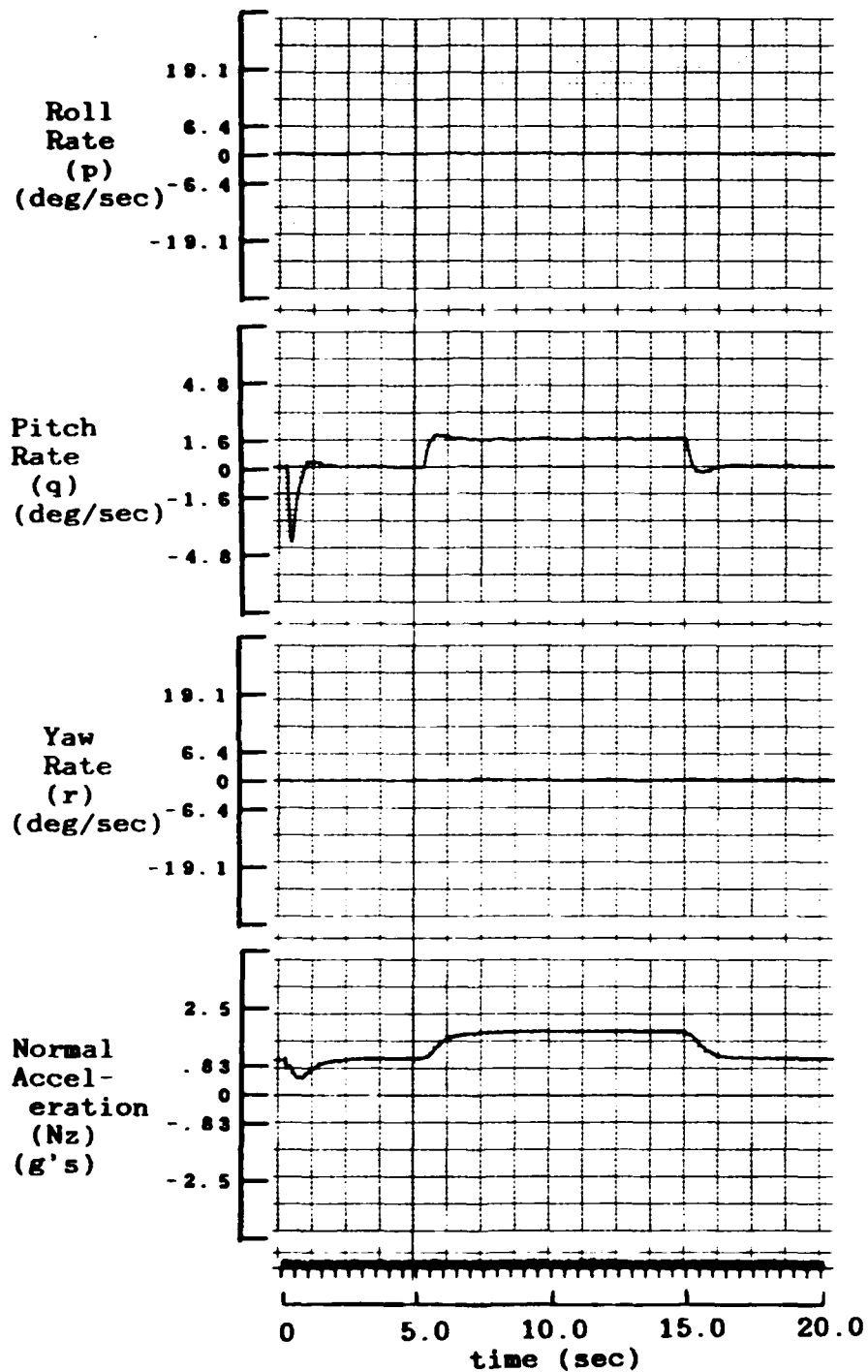


Figure 6.6b. One-g Pitch Stick Input, Uncoupled Responses

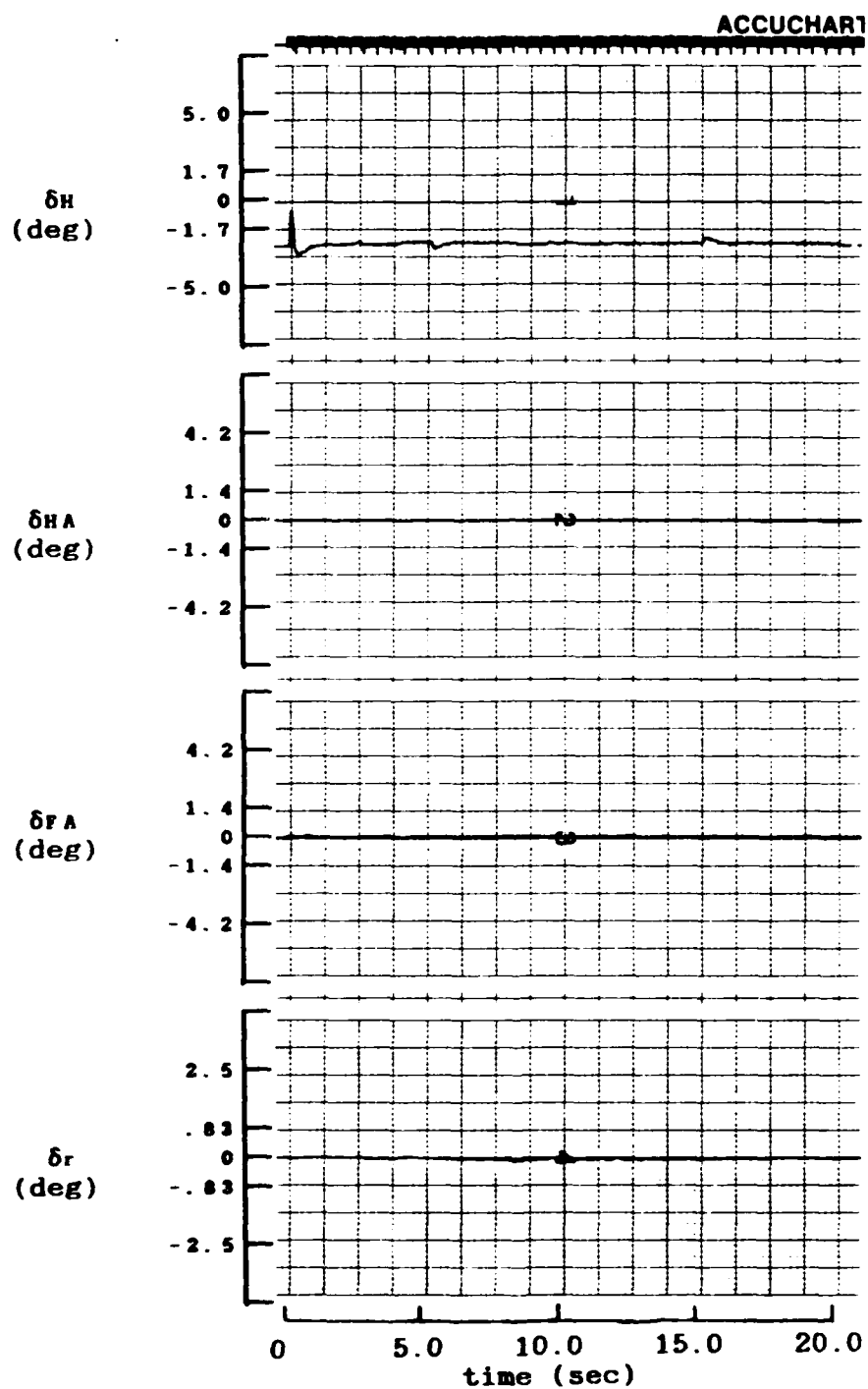


Figure 6.6c. One-g Pitch Stick Input, Uncoupled Responses

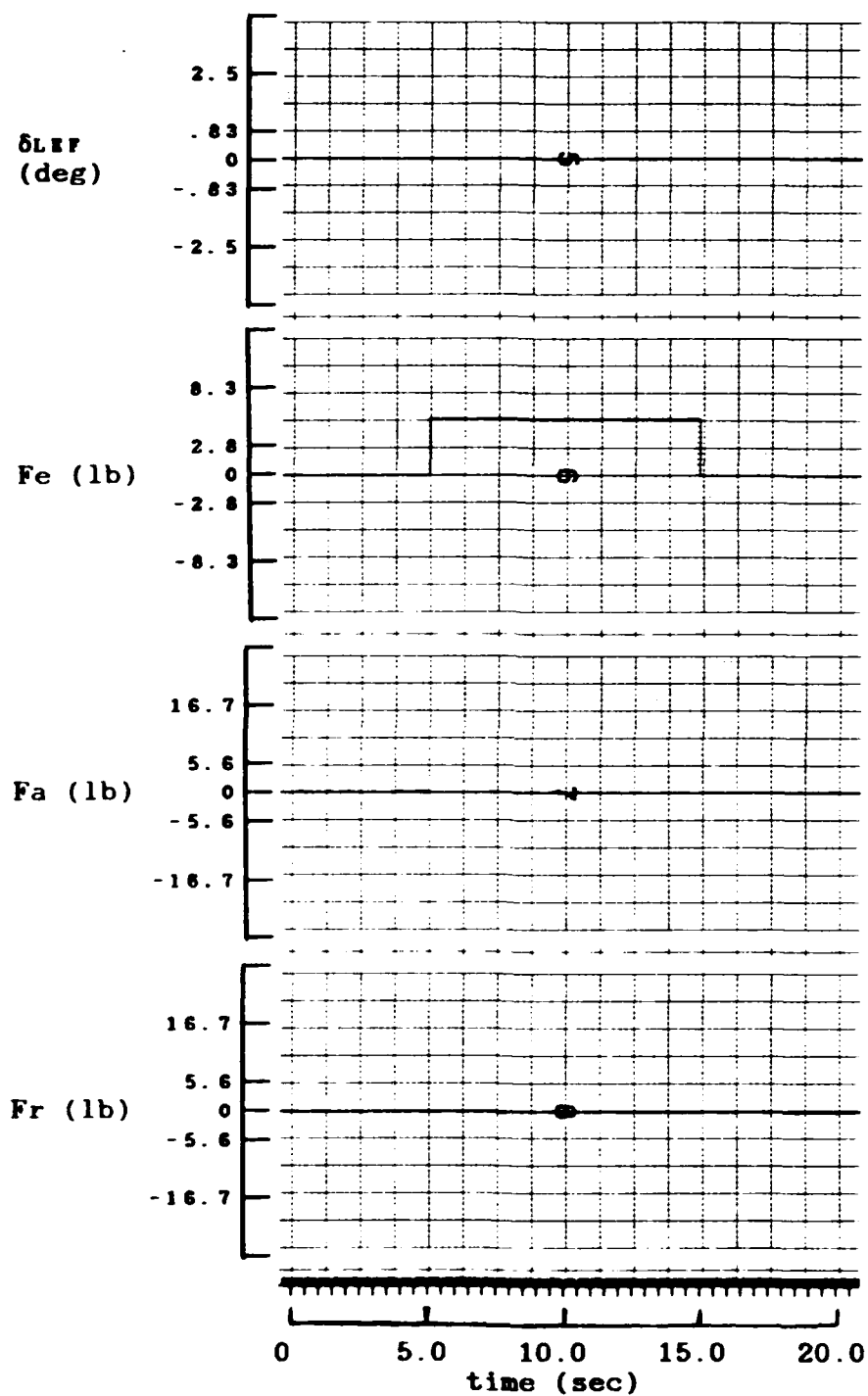


Figure 6.6d. One-g Pitch Stick Input, Uncoupled Responses

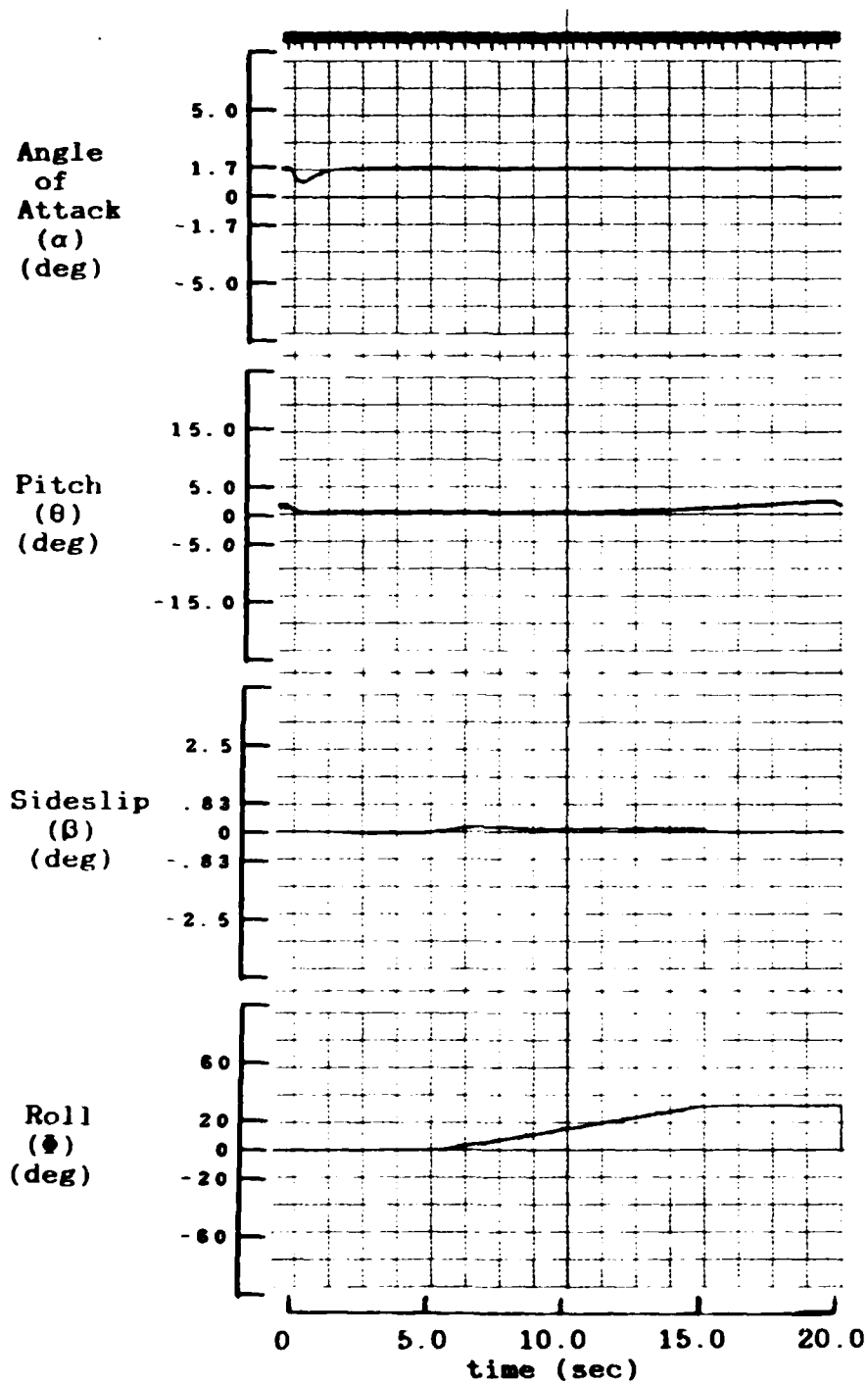


Figure 6.7a. 5 deg/sec Roll Stick Input, Uncoupled Responses

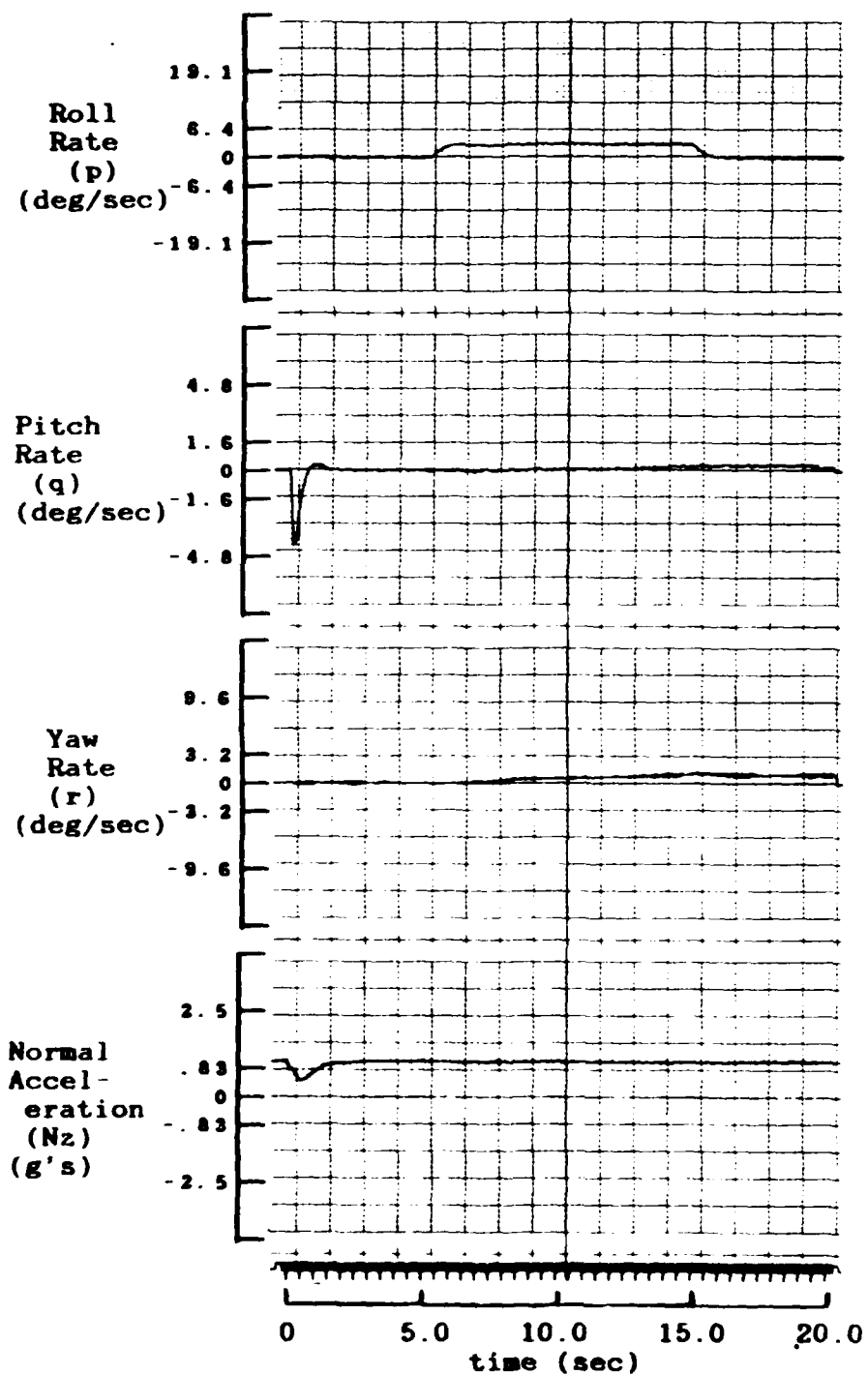


Figure 6.7b. 5 deg/sec Roll Stick Input, Uncoupled Responses

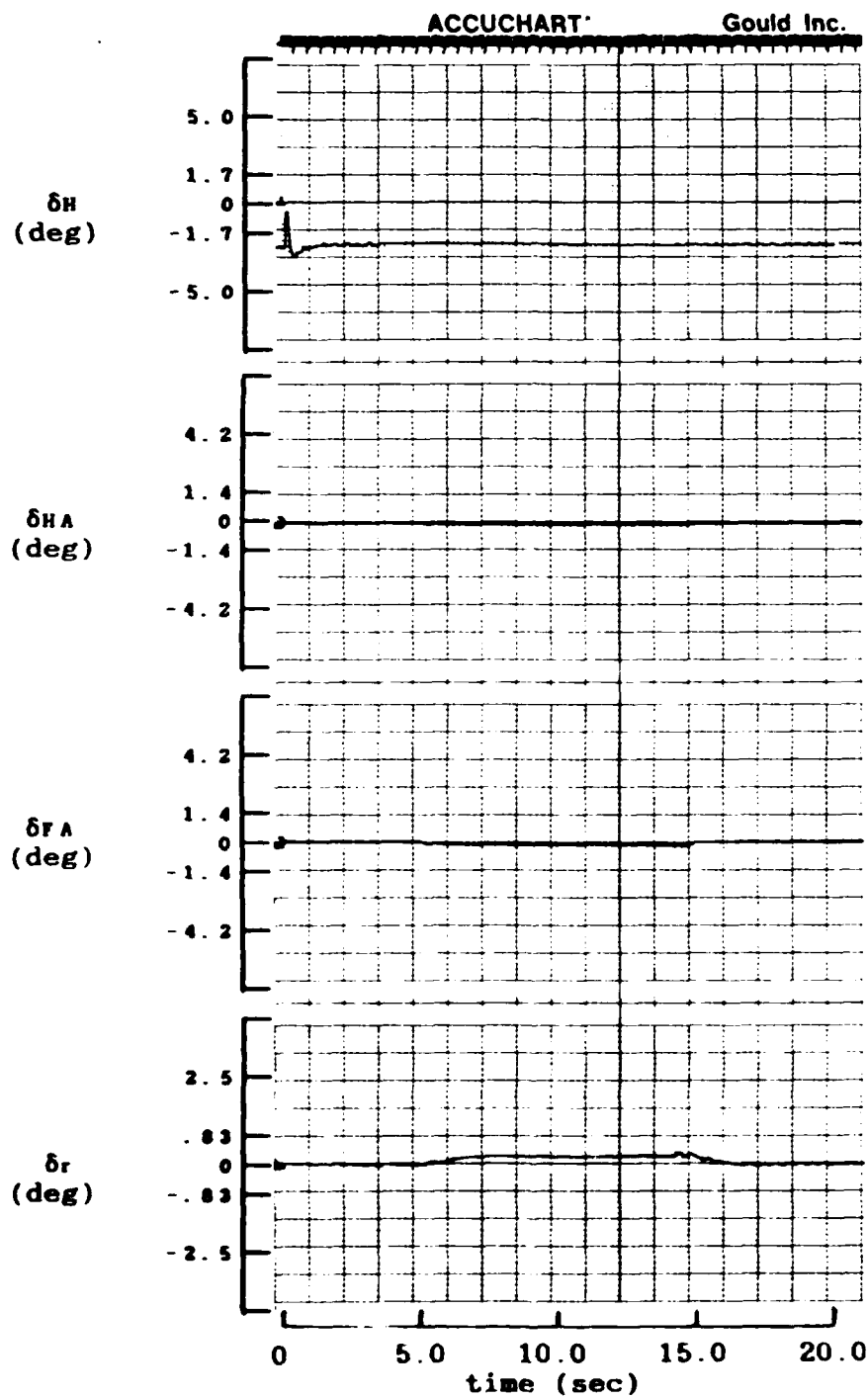


Figure 6.7c. 5 deg/sec Roll Stick Input, Uncoupled Responses

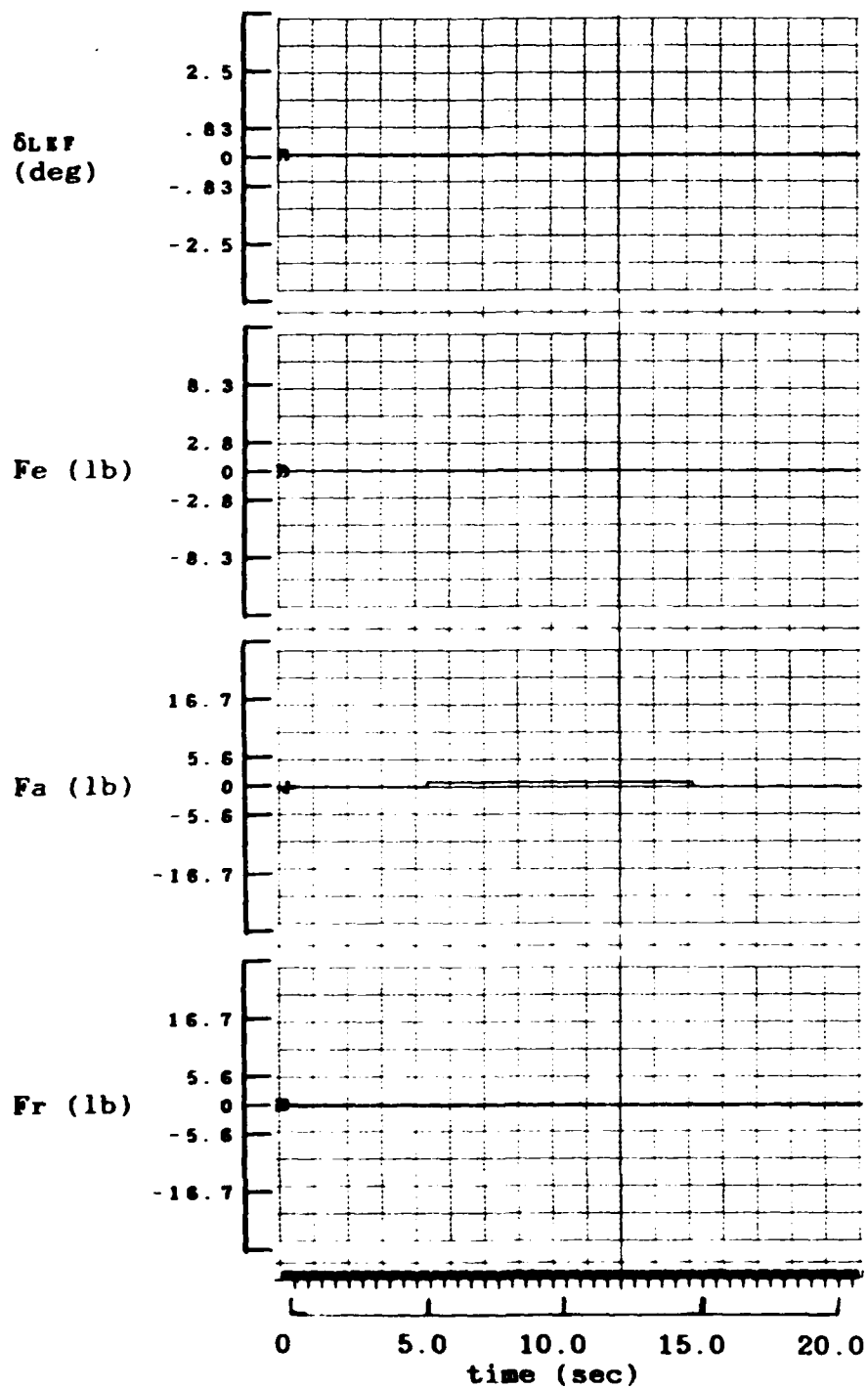


Figure 6.7d. 5 deg/sec Roll Stick Input, Uncoupled Responses

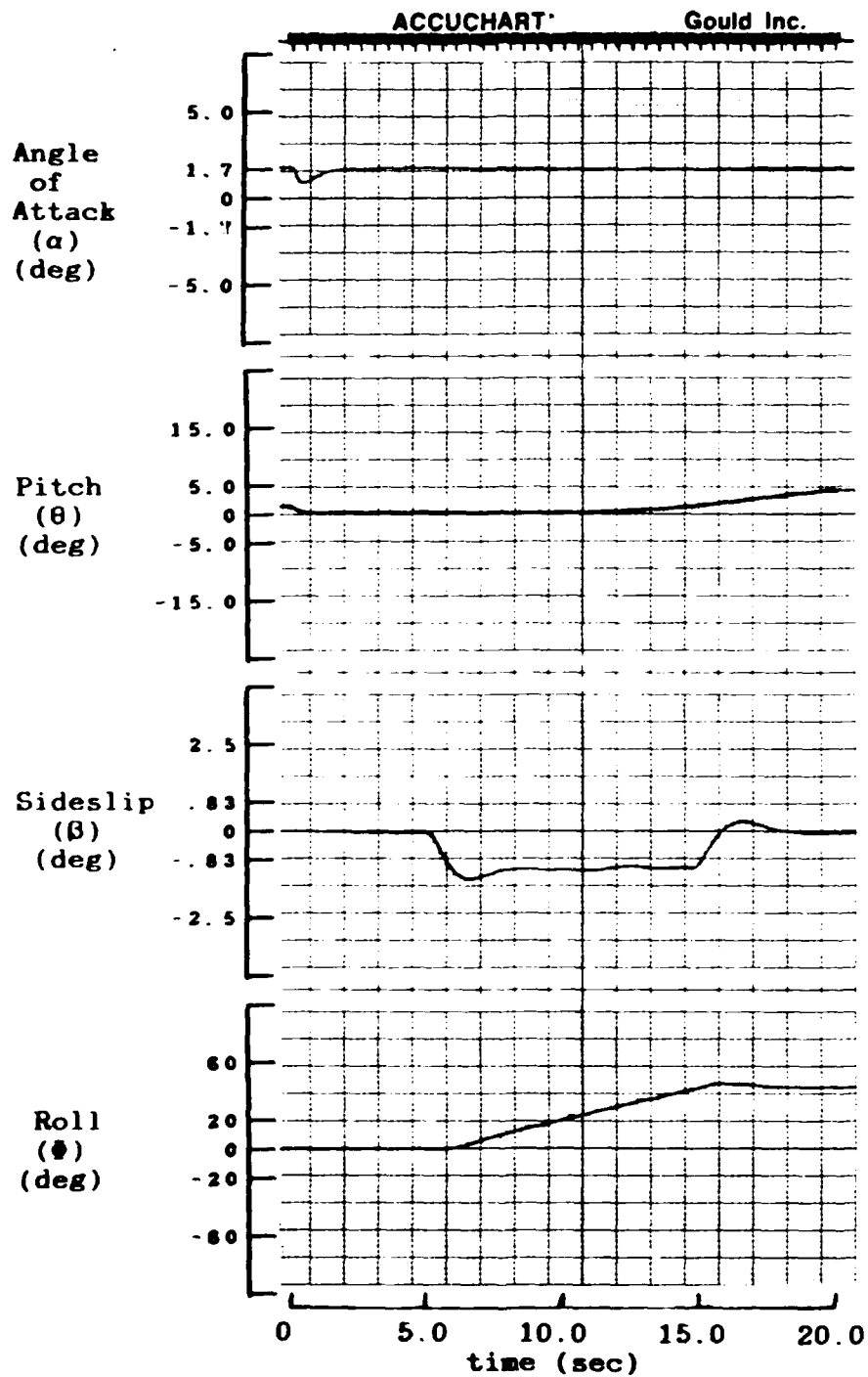


Figure 6.8a. One deg Rudder Pedal Input, Uncoupled Responses

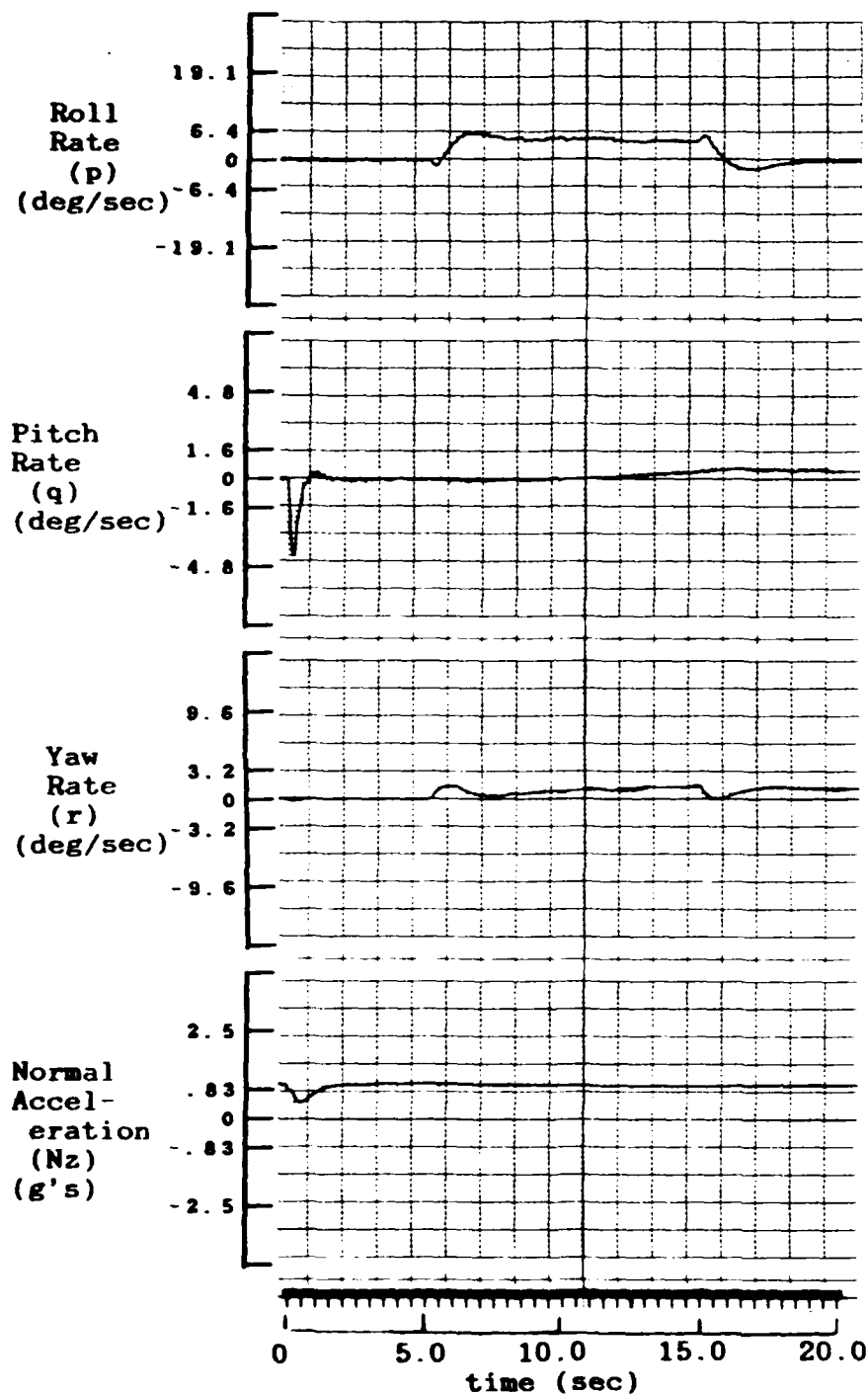


Figure 6.8b. One deg Rudder Pedal Input, Uncoupled Responses

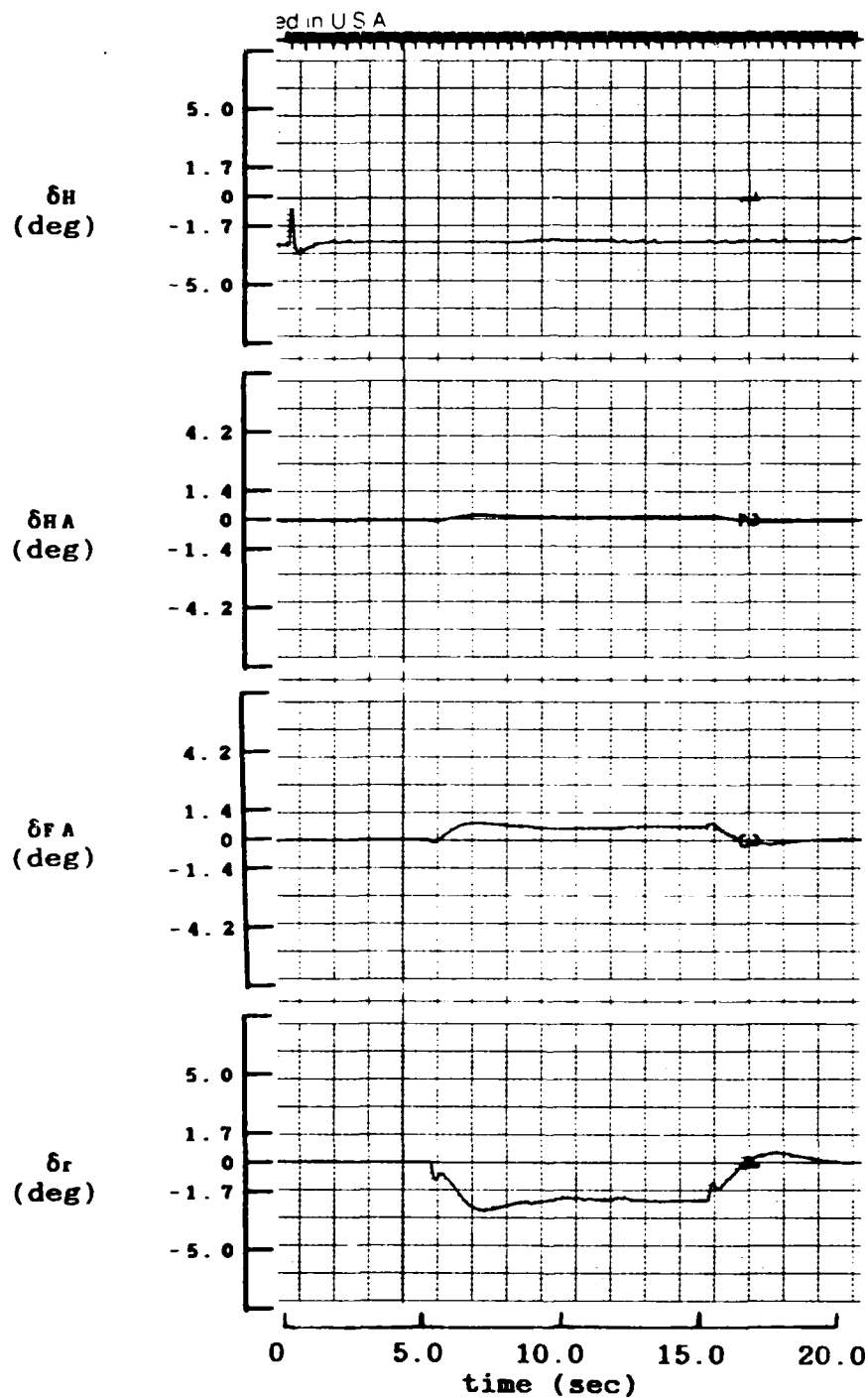


Figure 6.8c. One deg Rudder Pedal Input, Uncoupled Responses

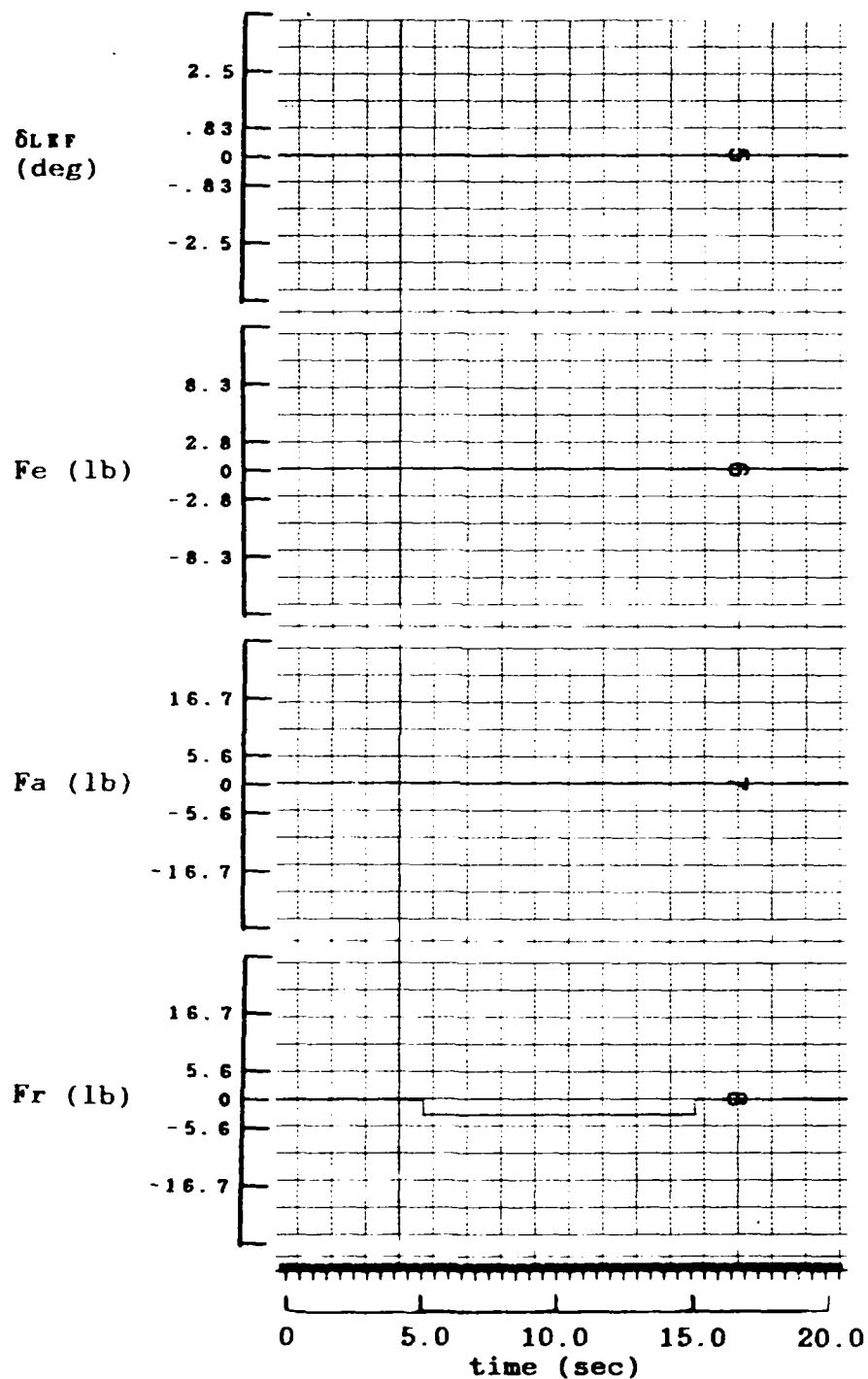


Figure 6.8d. One deg Rudder Pedal Input, Uncoupled Responses

roll rate or yaw rate become non-zero, then a longitudinal input would affect an inertially cross-coupled aircraft.

Figure 6.7 shows the responses of the aircraft to a commanded five deg/sec roll rate input. Comparison with the inertial cross-coupled responses of Figure 6.3 showing a commanded 45 deg/sec roll rate input, demonstrates the nonlinearities that occur due to the cross-coupling terms. Figure 6.7 shows the roll rate response reaching a steady state value lower than the 5 deg/sec input. This is due again to the type zero nature of the control system. Because of decoupling of the equations, longitudinal mode is unaffected by the roll input as is evident in the lack of response by the normal acceleration, pitch and angle of attack. Though in the coupled aircraft responses, normal acceleration, pitch and angle of attack are significantly affected.

A commanded sideslip input of one degree is commanded for the response plots of Figure 6.8. Again the longitudinal mode is unaffected by the input. Also, the nonlinearities apparent in the inertially cross-coupled aircraft in sideslip and rudder deflection are absent. Instead, sideslip overshoots slightly before it reaches a steady state value. Due to the aircraft's left wing moving in the direction of the velocity, a positive roll angle occurs. This is consistent with the coupled aircraft responses of Figure 6.4. And again, the rudder pedal force is shown as negative when it is actually positive of the

same magnitude. This is the result of a plotting error and does not affect the outputs.

Simulator Performance

In addition to strip chart plots, the simulator provides output variables to drive a crude cockpit. From this cockpit, a human pilot can input aileron and elevator commands through a force stick and can observe the displayed roll and pitch states of the aircraft on the oscilloscope display described in Chapter V.

Due to the limitation of the computational speed of the interpolation and look-up routine on SIMSTAR #114, the simulator cannot presently run in real-time. This prevents the evaluation of the performance of the simulator with the man-in-the-loop. Currently, a pilot can control the aircraft from the crude cockpit but must realize that the time-lag of the aircraft is large due to the current large time scaling of the simulation. The purpose for setting up a cockpit is to demonstrate the fully closed loop operation and show that it is possible to construct a cockpit for the simulator. Of course, future research can be devoted to improving the cockpit design. The current crude cockpit is sufficient for this thesis in proving that it can be done.

This chapter describes the validation of the aircraft model and briefly explains the performance of the simulator. The next chapter develops conclusions and recommendations based on this chapter's results.

VII. Conclusions and Recommendations

The conclusion based on the validated aircraft model is that it is possible to design a simulator for testing aircraft control system designs for the F-16A aircraft using the SIMSTAR hybrid computers. The current time scaling of the simulation solution is a temporary limitation to real-time operation, but it can be easily removed with commercially available and inexpensive computation hardware.

Recommendations toward achieving the goals of real-time capability and true man-in-the-loop testing include upgrading the SIMSTARS and improving the cockpit environment. Before the SIMSTARS can be upgraded they should be made more reliable and more user-friendly. Improvements in the SIMSTAR operating system can enhance the user-friendliness. Additionally, the computer downtime can be decreased by stocking spare parts and by more diligent documenting of problems and solutions so that re-occurrences can be easily spotted and quickly fixed.

Upgrading the SIMSTARS to achieve the speed necessary for real-time computation for large tasks such as those associated with this thesis can be achieved several different ways. The slowest component of the entire simulator is the interpolation and look-up routine on SIMSTAR #114. One way to speed this up is to convert the function generation task to EAI's FGS300 system whose sole purpose is to perform multivariable function generation in real-time. Another way and less specific in task is to replace the DAP with a faster processor and use a special

software package called FGSYS to increase the A/D and D/A conversions. FGSYS is not currently available on the AFIT SIMSTARs. This package carries out function generation using software much quicker than FORTRAN written code. The DAP in SIMSTAR #115 which executes the flight control system code can also use a boost in speed. A faster DAP in both SIMSTARs would help things considerably in achieving real-time.

The cockpit environment can be improved by providing a higher quality graphics display. It would be best to have another computer sample analog values from the SIMSTARs and update the display. The difficulty arises from the fact that the SIMSTAR's capability to send data from its derivative region in real-time is currently non-existent. It is highly desirable for a FORTRAN "write" statement to be executed from inside the derivative region. The digital values can then be sent from the SIMSTAR directly to a graphics terminal pilot display.

Other improvements to the cockpit include giving the pilot trim capability for adjusting, in real-time, the aircraft's pitch, roll and yaw trim setting. Also a throttle control is needed to replace the current configuration of automatic throttle control zeroing out forward acceleration. Ideally, using a real F-16 control stick and throttle would make the cockpit most realistic.

There are many areas of research that this simulator can be used for. It provides a full inertially cross-coupled

fighter aircraft simulator to students, faculty and engineers for substantially more capable research and laboratory use. Flight control system design can be evaluated with higher fidelity and pilot feedback. Additionally, engineers can use the simulator as a rapidly programmable, high fidelity evaluation of any design work as an alternative to using the large complex simulators. The benefits of the simulator as a research tool are limited only by the current hardware configuration. Subsequent upgrades can eventually provide a research tool with nearly unlimited capability.

This thesis evaluates the simulator at a single flight condition. Additional evaluation of the simulator and aircraft model at various flight conditions in the flight envelope is necessary. Scaling, which is always a challenge with analog computers, should be investigated and improved upon. Many of the scaling values used were "recommended" by the SIMSTAR translator. However, these scaling values are not necessarily the most appropriate and can be improved upon.

One last recommendation includes improving the current aerodynamic data search algorithm. The current algorithm is not the most efficient. Using a more efficient algorithm for table look-up can save a considerable amount of time.

Future work beyond this thesis should concentrate on improving the simulator until true man-in-the-loop high fidelity simulation and testing capability is achieved.

Appendix A: Simulator for Interactive Man-in-the-Loop
Testing of Aircraft Control Systems
(SIMTACS) User's Guide

This appendix is a user's guide for modifying the various programs that make up SIMTACS on the SIMSTAR computer systems.

SIMULATOR FOR INTERACTIVE MAN-IN-THE-LOOP
TESTING OF AIRCRAFT CONTROL SYSTEMS (SIMTACS)

SIMTACS USER'S GUIDE

Mark W. Kassan, B.S.E.E.
Second Lieutenant, USAF

December 1987

Table of Contents

	Page
I. Introduction	110
II. SIMTACS Routines	111
III. Coding SIMTACS	113
IV. Running SIMTACS	123
V. Debugging SIMTACS	128
VI. Analog Controllers	133
VII. Advanced Work	135

I. Introduction

This purpose of this manual is to provide the background and instructions necessary for a reader to use SIMTACS on the hybrid SIMSTAR computer system. The author assumes the reader is familiar with the SIMSTAR and its manuals and can write code in the FORTRAN computer language. He also assumes the reader has in hand a flight control system block diagram he wishes to use with SIMTACS and he fully understands his design of that flight control system. With these assumptions, the manual explains how to code, run and debug SIMTACS for a digital flight control system (DFCS) design. A DFCS is used as an example.

II. SIMTACS Routines

These routines reside on SIMSTAR #115 as part of SIMTACS and should be used as skeleton programs for the implementation of the designer's own control law. Picking one of the skeleton programs that pertains to the type of control system design is the first step to using SIMTACS. Once chosen, the program should be copied into the designer's SIMSTAR directory and manipulated as explained in this guide. Also, the designer must also make sure that the appropriate ILUR program on SIMSTAR #114 is used.

S1.DLONG	Digital longitudinal aircraft (uses S1.ILUR)
S1.DLAT	Digital lateral-directional aircraft (uses S1.ILUR) NOTE: There are problems with this program dealing with transfer of a PADC variable. Sometimes it works. The problem has been brought up to the attention of the EAI representative, but has not been solved as of this printing.
S1.DUAC	Digital inertially uncoupled aircraft with ideal inputs (uses S1.ILUR1) S1.DUAC1 is identical except for the '@RECORD' statement.
S1.DCAC	Digital inertially cross-coupled aircraft with ideal inputs (uses S1.ILUR1) S1.DCAC1 is identical except for the '@RECORD' statement.
S1.IDCAC	Digital inertially cross-coupled aircraft with pilot inputs and oscilloscope display. (uses S1.ILUR2)

For those designers interested in implementing an analog controller the following programs are available. Due to the number of MCBs that are used for an analog controller, the programs cannot offer the coupling feature available with a digital controller.

S1.ALONG	Analog longitudinal aircraft (uses S1.ILUR) with ideal inputs
S1.ALAT	Analog lateral-directional aircraft (uses S1.ILUR) with ideal inputs

III. Coding SIMTACS

This chapter tells the reader how to convert his control law block diagram to a useable form for implementing on the SIMSTAR as part of SIMTACS.

Step one. Take each transfer function and convert to a time discrete difference equation using the backward difference equation (for specifics, see backwards difference equation, Chapter IV of original document). For example, the transfer function

$$\frac{OUT}{IN} = \frac{20}{S + 20}$$

is converted to the FORTRAN code

$$OUT = (B20 * IN + OUTL)/A20 \quad (A-1)$$

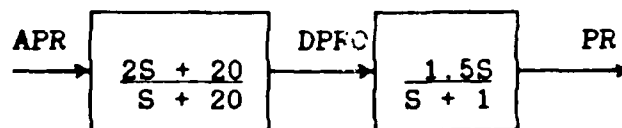
where A20 and B20 are coefficients equal to

$$\begin{aligned} A20 &= 1 + 20*T \\ B20 &= 20*T \end{aligned}$$

and OUTL is the value of OUT from the last sample interval.

All variable names are arbitrary. One, though, must keep track of, and be able to relate to them for easy debugging.

Step two. Take each difference equation from step one and sort them according to their use in the design. One must realize that two transfer functions in a row in a block diagram share a variable name in that one has it as an output while the other has it as an input. For example, the transfer functions



share DPRO between them. The actual FORTRAN code is

$$DPRO = (2*(A10*APR - APRL) + DPROL)/A20 \quad (A-2)$$

$$PR = (1.5*(DPRO - DPROL) + PRL)/A1 \quad (A-3)$$

where

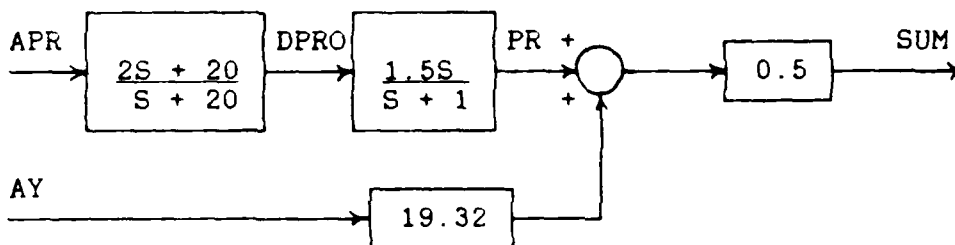
$$A10 = 1 + 10*T$$

$$A20 = 1 + 20*T$$

$$A1 = 1 + T$$

and APRL, DPROL and PRL are the values computed during the last sample interval.

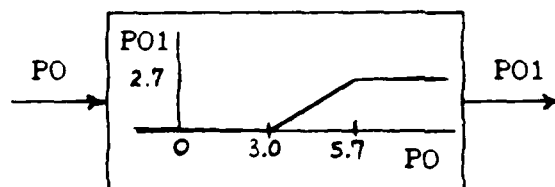
Gains are just multiplied by the variable name and summers add together two quantities. For example, this block diagram



becomes, with PR defined by Equations (A-2) and (A-3),

$$SUM = 0.5*(PR + 19.32*AY) \quad (A-4)$$

For a limiter, the following is a good example. Suppose we have



This is coded using the FORTRAN IF-statement. First, calculate

the equation for a straight line for the line between 3.0 and 5.7. This is

$$PO1 = 0.67*PO - 2 \quad (A-5)$$

Then use the IF-statement to cut off the straight line equation at 0.0 and 2.7 with

```
IF (PO1.GT.2.7) PO1 = 2.7
IF (PO1.LT.0.0) PO1 = 0.0
```

NOTE: The variable names DH1, DHA1, DFA1, DR1 and DLEF1 are reserved as the names of the inputs to the analog servoactuators and should be used for this purpose.

Place the resulting FORTRAN code in the routine DFCS between the COMMON statement and the RETURN statement (see below).

SAMPLE DFCS CODE

```
COMMON //.....
*
* SIMPLIFIED CONTROL LAW FOR LATERAL CASE
*
  FRO=(B15*FRT*.316+FROL)/A15
*
  PO=(B67*.15*ABS(PT/.01745)+POL)/A67
  PO1=.67*PO-2
  IF (PO1.GT.2.7) PO1=2.7
  IF (PO1.LT.0) PO1=0
*
  AOA=(B10*AOAT/.01745+AOAOL)/A10
*
  DRPA=FRO*(4*PO1+AOA)+YTR1
  AOAM=(.67*(ABS(AOA)/10-1)+.06)*AOA
*
  IF (FAT.LE.4) THEN
    FAT=5*FAT
  ELSE
    FAT=24.2*FAT-76.7
  END IF
  FAO=(B25*FAT+FAOL)/A25
```

```

*
DFA1=.12*(PT/.01745+RTR1-FAO)
FRM=DRPA+AOAM*DFA1
APR=RT/.01745-AOAO*PT
DPRO=(2*(A10*APR-APRL)+DPROL)/A20
*
PR =(1.5*(DPRO-DPROL)+PRL)/A1
*
DR1=FRM+.5*(19.32*AY+PR)
*
SIMPLIFIED CONTROL LAW FOR LONGITUDINAL CASE
*
QSO=(QST/.01745-QSL+QSOL)/A1
*
PRO=((3*(QSO-QSOL)+B14*QSO)*CON2+PROL)/A14
*
AZO=(3*(DAT-DATL)+B14*DAT+AZOL)/A14
*
IF (FET.LE.4.25) THEN
  FE1=.1035*FET
ELSE
  FE1=.4*FET-1.26
END IF
*
FE1=FE1+ET1
FEO=(B7*FE1+FEOL)/A7
*
INP=(PRO+AZO-FEO)*CON3
INPO=A5*INP-INPL+INPOL
*
DH1=INPO+CON1*AOAO
*
PAST VALUES
*
DLEF1=0
DHA1=DFA1
*
FROL=FRO
POL=PO
AOAOL=AOAO
FAOL=FAO
APRL=APR
PRL=PR
DPROL=DPRO
*
QSL=QST/.01745
QSOL=QSO
PROL=PRO
AZOL=AZO
DATL=DAT

```

```

FEOL=FEO
INPL=INP
INPOL=INPL
*
RETURN

```

Step three. Group all past values immediately after the code of difference equations from step two and set them equal to their current values as shown above in the sample DFCS. The order of these equations does not matter as long as they appear after the rest of the code in this routine.

Step four. The feedback variables necessary for the control system design are placed in the FEEDBACK BOX in the PADC statement. For example, if the control system needs the variables AOA, P, R, FR, FA, AY, QS, NZ and FE fed back, then the following code is used.

```

'!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!'
'|
'|                                FEEDBACK BOX                                '|
'|
'|
DCA(1)=DFA,DHA,DH,DR,DLEF,TH
PADC(1)=AOA,P,R,FR,FA,AY,QS,NZ,FE,DTD,PITCH
'|
'|!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!'

```

The DCA statement should not have to be changed. The variables DTD and PITCH are used for other purposes and should be left in the PADC statement. The PADC statement declares the analog variables that the SIMSTAR will sample for use in the digital region. The DCA statement declares the digital variables that will be passed to the analog region.

Next the PREP1 subroutine just below the FEEDBACK BOX needs to reflect the variables defined with the PADC statement. For example with the above PADC variables, PREP1 is

```

SUBROUTINE PREP1
*
  INCLUDE E1.IDCAC 'This name must be of the form E1.NAME'
                    'where NAME is the name of the S1.NAME'
*
  COMMON /QQADC/NADC, LVNS(11), ADC(0:10)
                    '11 reflects number of PADC variables'
                    '0:10 reflects 11 PADC variables'
                    'starting at 0'

  CALL QRBPADR
  AOA =ADC(0)*S:AOA
  P   =ADC(1)*S:P
  R   =ADC(2)*S:R
  FR  =ADC(3)*S:FR
  FA  =ADC(4)*S:FA
  AY  =ADC(5)*S:AY
  QS  =ADC(6)*S:QS
  NZ  =ADC(7)*S:NZ
  FE  =ADC(8)*S:FE
  DTD =ADC(9)*S:DTD
  PITCH=ADC(10)*S:PITCH
*
  RETURN
  ENTRY PREPIN
  CALL PREPBD
  RETURN
  END

  BLOCKDATA PREPBD
  COMMON /QQADC/NADC, LVNS(11), ADC(0:10) 'Same as above'
  DATA NADC/11/, LVNS/0,1,2,3,4,5,6,7,8,9,10/
                    'The DATA statement defines the '
                    'COMMON variables and must be'
                    'changed to reflect number of PADC'
                    'variables used'

  END

```

The feedback variables must also appear on the right-hand side of an assignment statement in the DISCRETE BOX. For instance:

```

FRI=FR
AYI=AY
AOAI=AOA

```

PI=P
RI=R
FAI=FA
FEI=FE
NZI=NZ*0.03106
QSI=QS

Note the 'I' extension on the left-hand side of the equations. This is not necessary to be an 'I' but a consistent format should be used as described in step six.

Step five. Now take all of the past values from the DFCS routine, add the suffix 'I' on them and place them in the INITIAL BOX and set equal to zero. This is so that the value at the beginning of each run is initially zero for all variables when time is less than or equal to zero. For example

[illegible]

```
'* SET PAST VALUES INITIALLY TO ZERO *'
```

FROLI=0
POLI=0
AOALI=0
FAOLI=0
APRLI=0
DPROLI=0
PROLI=0
PRLI=0
QSLI=0
AZOLI=0
NZLI=0
FEOLI=0
INPOLI=0
INPLI=0
QSOLI=0

1
 2
 3
 4
 5
 6
 7
 8
 9
 10
 11
 12
 13
 14
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
 27
 28
 29
 30
 31
 32
 33
 34
 35
 36
 37
 38
 39
 40
 41
 42
 43
 44
 45
 46
 47
 48
 49
 50
 51
 52
 53
 54
 55
 56
 57
 58
 59
 60
 61
 62
 63
 64
 65
 66
 67
 68
 69
 70
 71
 72
 73
 74
 75
 76
 77
 78
 79
 80
 81
 82
 83
 84
 85
 86
 87
 88
 89
 90
 91
 92
 93
 94
 95
 96
 97
 98
 99
 100
 101
 102
 103
 104
 105
 106
 107
 108
 109
 110
 111
 112
 113
 114
 115
 116
 117
 118
 119
 120
 121
 122
 123
 124
 125
 126
 127
 128
 129
 130
 131
 132
 133
 134
 135
 136
 137
 138
 139
 140
 141
 142
 143
 144
 145
 146
 147
 148
 149
 150
 151
 152
 153
 154
 155
 156
 157
 158
 159
 160
 161
 162
 163
 164
 165
 166
 167
 168
 169
 170
 171
 172
 173
 174
 175
 176
 177
 178
 179
 180
 181
 182
 183
 184
 185
 186
 187
 188
 189
 190
 191
 192
 193
 194
 195
 196
 197
 198
 199
 200
 201
 202
 203
 204
 205
 206
 207
 208
 209
 210
 211
 212
 213
 214
 215
 216
 217
 218
 219
 220
 221
 222
 223
 224
 225
 226
 227
 228
 229
 230
 231
 232
 233
 234
 235
 236
 237
 238
 239
 240
 241
 242
 243
 244
 245
 246
 247
 248
 249
 250
 251
 252
 253
 254
 255
 256
 257
 258
 259
 260
 261
 262
 263
 264
 265
 266
 267
 268
 269
 270
 271
 272
 273
 274
 275
 276
 277
 278
 279
 280
 281
 282
 283
 284
 285
 286
 287
 288
 289
 290
 291
 292
 293
 294
 295
 296
 297
 298
 299
 300
 301
 302
 303
 304
 305
 306
 307
 308
 309
 310
 311
 312
 313
 314
 315
 316
 317
 318
 319
 320
 321
 322
 323
 324
 325
 326
 327
 328
 329
 330
 331
 332
 333
 334
 335
 336
 337
 338
 339
 340
 341
 342
 343
 344
 345
 346
 347
 348
 349
 350
 351
 352
 353
 354
 355
 356
 357
 358
 359
 360
 361
 362
 363
 364
 365
 366
 367
 368
 369
 370
 371
 372
 373
 374
 375
 376
 377
 378
 379
 380
 381
 382
 383
 384
 385
 386
 387
 388
 389
 390
 391
 392
 393
 394
 395
 396
 397
 398
 399
 400
 401
 402
 403
 404
 405
 406
 407
 408
 409
 410
 411
 412
 413
 414
 415
 416
 417
 418
 419
 420
 421
 422
 423
 424
 425
 426
 427
 428
 429
 430
 431
 432
 433
 434
 435
 436
 437
 438
 439
 440
 441
 442
 443
 444
 445
 446
 447
 448
 449
 450
 451
 452
 453
 454
 455
 456
 457
 458
 459
 460
 461
 462
 463
 464
 465
 466
 467
 468
 469
 470
 471
 472
 473
 474
 475
 476
 477
 478
 479
 480
 481
 482
 483
 484
 485
 486
 487
 488
 489
 490
 491
 492
 493
 494
 495
 496
 497
 498
 499
 500
 501
 502
 503
 504
 505
 506
 507
 508
 509
 510
 511
 512
 513
 514
 515
 516
 517
 518
 519
 520
 521
 522
 523
 524
 525

Step six. The call to the DFCS routine in the DISCRETE BOX must contain all past values, PADC variables and DCA variables, all with the 'I' extension. Order in the CALL statement does not matter. The order, though, must be duplicated when the DFCS subroutine heading is typed in.

In the DISCRETE BOX:

```
CALL DFCS(DHAI,DH1,DFAI,DRI,DLEF1,AOAOLI,PROLI,FROLI,POLI, ...
          FAOLI,APRLI,DPROLI,PRLI,AYI,AOAI,PI,RI,FRI,FAI, ...
          RTR,YTR,QLI,AZOLI,NZLI,INPOLI,FEOLI,QSOLI,INPLI, ...
          NZI,QSI,FEI,CUT1,CUT2,CUT3,CUT4,CUT5,ET)
```

In the DFCS heading:

```
SUBROUTINE DFCS(DHA1,DH1,DFA1,DR1,DLEF1,AOAOL,PROL,FROL,
+               POL,FAOL,APRL,DPROL,PRL,AYT,AOAT,PT,RT,
+               FRT,FAT,RTR1,YTR1,QL,AZOL,DATL,INPOL,
+               FEOL,QSOL,INPL,DAT,QST,FET,CON1,CON2,
+               CON3,CON4,CON5,ET1)
```

Additionally, other parameters can be passed to the DFCS routine if one wishes, by including the variables in the DFCS CALL and the DFCS heading such as CUT1, CUT2 and CUT3. It is important to use dummy names in the DFCS heading for the variables in the CALL statement such as AOAT for AOAI, PT for PI etc.. Always use names that can be easily recognized. For instance, here the 'T' extension means the variable is inside the DFCS routine. The 'I' extension means the variable appears in the DISCRETE BOX as PADC variables or in the INITIAL BOX as past values. If the reader uses another extension other than 'I', then wherever this guide uses an extension 'I', the reader should replace with his extension. The only significance in the 'I' extension is that when the PADC variables come from the

If one is using the strip chart recorders or the XY plotter then the appropriate analog variables must be listed in the appropriate command located in the OUTPUT BOX. For example

If one is using a dynamic model program that uses ideal pilot inputs, the user must be aware of the analog code that creates these ideal inputs. This code is meant to be manipulated by the user for his specific code and is located in the analog region near the top of the code. The code will look similar to the following.

```
FE=FEGA*(FASTP1-FASTP2)
FA=FAGA*(FASTP1-FASTP2)
FR=FRGA*(FASTP1-FASTP2)
```

```
FASTP1=STEP(TDLY1)
FASTP2=STEP(TDLY2)
```

The code above creates pulse inputs for FE, FA and FR based on the time delays TDLY1 and TDLY2.

The last step in the coding process is compilation. This is the point when the SIMSTAR will tell you what errors your program has. Some of these are discussed in the debugging chapter. Next, the running of error-free code is discussed.

IV. Running SIMTACS

After successful compilation, the simulator is ready to run. The following is a step by step procedure for loading and running SIMTACS.

Step one. The first step is to make sure that all hardware connections are made between the two SIMSTARs, the oscilloscope cockpit display, the pilot inputs and the strip chart recorder. All connections are explained in Appendix B. If only strip chart recorder plots are needed with ideal pilot inputs then the connections for the pilot inputs and the cockpit display are not important. Just remember to use the correct aircraft model program.

Step two. Next, the interpolation and look-up routine (ILUR) on SIMSTAR #114 must be started. Again, the appropriate ILUR program must be used. Loading of the program and starting the run are done as follows.

```
TSM> ILUR (or appropriate name)
```

```
SIMRUN - SIMSTAR RUN-TIME EXEC REV. D02
```

```
** LOADING P1.ILUR **
```

```
>S CINT=.05,PERIOD=.0501  
S CINT=.05,PERIOD=.0501
```

```
>START  
START
```

Usually after the loading message above, an 'OVERRANGE ERROR OCCURRED IN SETTING RUNTIME DCA' error will occur. This error

is not important and will not influence the results. Also, if a 'MACRO IN OVERLOAD' error occurs, ignore it. This message at this point just means that one of the analog lines from SIMSTAR #115 has a large voltage on it. This large voltage will "go away" when the dynamics model is loaded on SIMSTAR #115.

After starting the ILUR program, the INITIAL region will run for about 100 seconds. It will signal its completion with an 'END OF INITIAL' message. Once this message appears, the ILUR program is running. Again, ignore any 'MACRO IN OVERLOAD' errors that occur at this time.

The ILUR program is meant to run for a full two hours, so that testing on SIMSTAR #115 can be done continuously for the period without interruption from SIMSTAR #114. Because of the setting of CINT and PERIOD on SIMSTAR #114, SIMSTAR #114 will be locked up for the two hour duration. Currently, the only way to unlock it without waiting two hours is to reboot the system. After the routine is finished an error message called an 'INTERRUPT RATE ERROR' will appear. This is not a serious error on SIMSTAR #114 when running the ILUR program because no calculations are based on a frame time. The error means that execution of the digital code could not be done in the 50 millisecond time frame that is set using CINT and PERIOD. If this error appears while running on SIMSTAR #115 then this means your last run which incurred the errors, has bad data and the run should be reaccomplished.

Step three. The loading of the dynamics model on SIMSTAR #115 is the next step. Let's say that the name of the dynamics model that you compiled with your flight control system design code is called S1.ACMOD. The following demonstrates loading ACMOD on SIMSTAR #115.

```
TSM>SIMTACS ACMOD
```

```
SIMRUN - SIMSTAR RUN-TIME EXEC REV. D02
```

```
** LOADING P1.ACMOD **
```

```
>
```

If the 'OVERRANGE ERROR OCCURRED IN SETTING RUNTIME DCA' error message appears, ignore it again. It will go away and it means that at least one of the DCA variables is overloaded. This is temporary because the DCA's will reset themselves within range once the routine is run. Also, ignore any 'MACRO IN OVERLOAD' errors at this stage. SIMTACS is a routine used on the dual disk system of SIMSTAR #115 so that all of the necessary executable programs are loaded into the RUNS directory located on the @SIMSYS drive. From the RUNS directory, SIMTACS loads ACMOD.

Step four. Setting initial conditions and any parameters important to your simulation run is covered next. Some of the variables that one might want to set, appear in the following example.

```

>S CINT=.06,PERIOD=.0601
  S CINT=.06,PERIOD=.0601

>S PITCH0=1.9,AOAO=1.9,DHTR=-2.8
  S PITCH0=1.9,AOAO=1.9,DHTR=-2.8

>S BET=20,RUNTIM=20,ET=6
  S BET=20,RUNTIM=20,ET=6

>START
  START

```

CINT, which is set in seconds, is the communications interval variable. This variable controls the rate of the periodic sending of data to the analog region. While PERIOD, which is set in seconds, controls the rate at which the digital region is executed. These two variables are set to the sampling period that the control system is designed at, minus the time it takes for the ILUR program on SIMSTAR #114 to execute. The ILUR program sampling time is 125 ms but must be reduced by the time scaling factor, BET, which is set on SIMSTAR #115. If we assume BET is set to 20, then the ILUR program appears to SIMSTAR #115 as running at 6.25 ms. If the user's control system sampling period is meant to be 15 ms then this gives 8.75 ms for running the control system code. But again, the BET factor on SIMSTAR #115 must be taken into account. Multiplying 8.75 by 20 gives 175 ms. This is the setting on SIMSTAR #115 of CINT. Because CINT and PERIOD cannot be a multiple of each other (unless an ITC card is being used). PERIOD must be offset by 0.0001 seconds from CINT to a value in

this case of 0.1751 seconds. If they are set equal to each other, the program will run the first time but will lock up when it tries to return control to the terminal.

The variables PITCH0, AOA0 and DHTR are initial conditions in this example that set the aircraft to level flight. These and other initial conditions can be set anytime before a run.

The variable BET is important because it determines how much slower than real-time the simulation will run. Typical simulations using SIMTACS run at about BET=20 for good stable aircraft responses. If this variable is not set, the aircraft will immediately go unstable displaying the 'MACRO IN OVERLOAD' error. If you get this error immediately, stop the simulation, do a SIM/IC and a SIM/OVD to clear the values and check the BET value. Try running again and if the same thing happens then something else is wrong with the program. RUNTIM is the terminal time for the program. In this example it is set to 20 seconds. One must be careful in setting too large a RUNTIM, because with a BET equal to 20, the simulation will run for 400 seconds unless aborted early. The ET variable is the pitch trim set by the designer of the flight control system. The user has the capability to define many parameters to do anything he wants. For instance, gains can be parameters for adjusting designs and also cutoffs can be placed in the control system to actually open feedback loops of the control system. The flexibility is left to the ingenuity of the control system designer.

V. Debugging SIMTACS

There are many errors that can occur using the SIMSTARs. Only some of the major ones associated with SIMTACS are discussed here.

RUNTIME ERRORS

Address Specification error - If this error occurs on SIMSTAR #114, it means that there was an attempt to run an ILUR program more than once without exiting RUNSIM first. The ILUR programs access data using FORTRAN read statements. Because of this, the addresses that pick the data out of the data files need to be reset. The solution is to exit RUNSIM and reload the ILUR program.

SIMLOAD error - There are two ways that this error can occur on either machine. The first is due to a macro board "going bad" or the macro board is not in the slot where the compilation thought it should be. The other is due to a problem with the connection matrix. A solution is to recompile the program. If this does not work then the EAI representative should be informed.

MACRO IN OVERLOAD error - This error occurs when an analog variable goes beyond the scaling set by the user. Typing SIM/OVD will list the overloaded variables. First thing to check, if the error is occurring on SIMSTAR #115, is to make sure the ILUR program on SIMSTAR #114 is running and all hardware connections are made. The following is a test case to check the total coefficients coming from the ILUR program.

While in SIMRUN do the following.

```
>S H0=20000, AOA0=1.9, PITCH0=1.9, BETA0=0
>S H0=20000, AOA0=1.9, PITCH0=1.9, BETA0=0

>S ADOT0=0, PS0=0, QS0=0, RS0=0, VT0=933.2
>S ADOT0=0, PS0=0, QS0=0, RS0=0, VT0=933.2

>S DHTR=-2.8, DHATR=0, DFATR=0, DLEFTR=0, DRTR=0
>S DHTR=-2.8, DHATR=0, DFATR=0, DLEFTR=0, DRTR=0
```

>D LT,DT,MT
>D LT,DT,MT

LT/P1 26000 ? DT/P1 4600 ? MT/P1 0 ?

>D IT,NT,YT
>D IT,NT,YT

IT/P1 0 ? NT/P1 0 ? YT/P1 0 ?

If the values of LT,DT,MT,IT,NT and YT are in the general area to these values then the ILUR program and SIMSTAR #114 are working correctly. The values coming from SIMSTAR #114 will not be identically equal to these numbers because these are approximations. If they are not close then there is something wrong with SIMSTAR #114 or the connections between the machines.

If there seems to be no problem with the data coming from SIMSTAR #114, and rescaling using the rescaling command S/SF (see SIMRUN manual for details) doesn't work, the problem is most likely due to the controller or the transfer of the PADC variables. Next check the digital and the analog values of the PADC variables while running. This can be done using the OUTPUT command. If the digital values are not tracking the analog values, then seek help. This problem has not been solved yet and occasionally occurs for no currently known reason. If all digital values are tracking, the next thing to try is to actually track down the variables that are causing the overload and figure out how the digital controller is influencing these values. The only way to actually see variables inside the digital controller is to include the questionable variable or variables in the DFCS heading and CALL statements. Then they can be observed like analog variables. This is all the help that can be offered based on the author's experiences. One last bit of advise. Use your developed problem solving techniques to the utmost when tracking down errors.

OVERRRANGE ERROR OCCURRED DURING SETTING RUNTIME DCA error -
This is due to an overload of a DCA variable. It can be solved using the same technique laid out for the previous error. If it occurs during the loading process as discussed earlier in this guide, ignore the error. It will go away during the run.

INTERRUPT RATE ERRORS - This error can occur at the end of any single run. If rate errors occur at the end of a run on SIMSTAR #115 then that run should be reaccomplished. On

this next run increase the PERIOD and CINT variables. This error is due to the DFCS code not being able to execute in the time the designer set using CINT and PERIOD. One will notice that this error occurs when the ILUR program is finished on SIMSTAR #114. This is normal and should be ignored.

If an ILUR program on SIMSTAR #114 is aborted by halting the machine then the LCU should also be reset. This is done by pressing the LCU reset button once and waiting for three messages on the user terminal signifying the restoration of the firmware. If SIMSTAR #114 locks up while loading an ILUR program then resetting the DAP and the LCU should fix the problem. If it doesn't then see the EAI representative.

Another important point to remember is not to remove the connections between the two machines. Doing this can cause strange voltages on the ports or short out one of the ports which may create problems to the connection matrix. If connections are removed while at least one machine is in RUNSIM mode and strange things are happening, clearing the matrix may solve the problem. See the EAI representative for clearing the matrix.

If any error not listed here, occurs that either "boots" the user out of RUNSIM or locks up the machine, try loading the program once more. If that doesn't work then seek the aid of the EAI representative.

A word about scaling is necessary. If the user manipulates the scaling of any variables on SIMSTAR #115 that are sent to SIMSTAR #114 then the scaling of the variables in the ILUR program being used must also be changed. If this is

done, the program must be recompiled using the SIMILUR macro and not the SIMSTAR macro. For compiling an ILUR program enter SIMILUR ILUR,1,F instead of SIMSTAR ILUR,1,F.

COMPILATION ERRORS

During compilation, errors from SYNTAX to NO COMPONENTS AVAILABLE can pop up. If there are SYNTAX errors and you have a question as to why it is a SYNTAX error, look up the usage of the particular code used in the manuals. This should dawn some light on the problem, otherwise the EAI representative should be able to help.

Whenever level 4 errors occur, the compilation is halted. These problems are numerous and are not explained here. But a little overview of the situation is in order. Most level 4 errors occur due to scaling problems. If one is merely implementing his own DFCS code then all errors should be FORTRAN related and as such should be standard FORTRAN errors.

An error that can occur when manipulating the analog region especially when changing ideal inputs, is the NO MORE COMPONENTS AVAILABLE error. If this occurs then the component list on the compilation print out should describe what is missing and the solution is inserting into the machine (let EAI representative do this) the number of boards needed or doing some clever manipulation of variables that uses fewer boards freeing up the needed components. If the error occurs and there are not any components listed as being needed, then the

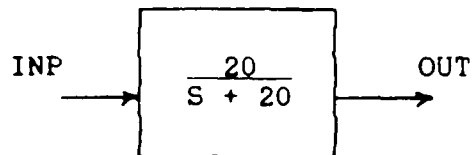
problem lies in using a DCA variable name in a CONNECT statement. Assigning the DCA variable to a dummy name in the analog region and using the dummy name in the CONNECT statement will solve the problem.

Another common mistake is using a variable on the right-hand side of an equation in the analog region without defining it in the analog region. Many errors will occur for this one mistake including the SIMSTAR defining the variable as EXTERNAL and expecting scaling on it. The solution is defining or removing the variable in question.

VI. Analog Controllers

Sometimes it is desired to design an analog controller for an aircraft such as for a classroom project. Programs S1.ALAT and S1.ALONG provide very good examples of working analog controllers. Each program defines a region for entering the controller. The nice thing about an analog controller when working with the SIMSTAR is that CINT, PERIOD (assuming they are not equal) and the whole digital region can be ignored.

The code for an analog controller, unlike that for a digital controller, must be implemented in P-TRAN. But the conversion to the language from a block diagram is straightforward. Use a direct algebraic relationship between the output derivative and the input for each transfer function and then include an integration to determine the output. As an example consider the transfer function



or

$$\begin{aligned} \text{OUTD} &= 20 * \text{INP} - 20 * \text{OUT} \\ \text{OUT} &= \text{INTEG}(\text{OUTD}, \text{OUT0}) \end{aligned}$$

where

OUTD is the derivative of OUT
OUT0 is the initial condition of OUT, most likely zero.

Using this procedure for each transfer function provides a P-TRAN equivalent to the block diagram. DO NOT use transfer function commands of the P-TRAN language for this. They do not always work correctly and are more trouble than they are worth.

VII. Advanced Work

This section provides some information for someone who wants to manipulate the ILUR programs, or manipulate the dynamic models for cockpit usage.

ILUR PROGRAM MANIPULATION

If there is some question as to whether or not the data being generated by the ILUR programs is correct, the routine LOOKUP can be called from the INITIAL region just after the call to SETUP with values of interest in the CALL heading. A routine called VALUE, which is sitting unused immediately after the PREPDCA routine, can be called in the INITIAL region after the call to the LOOKUP routine. Routine VALUE lists to the screen all 46 functions that were interpolated from the data for the conditions set in the LOOKUP heading. It then lists the total non-dimensionalized coefficients. This little setup can be used for advanced debugging purposes especially if one plans to manually look-up and interpolate the data to see if the compared values are correct. Included here are three conditions and all of the functions with correct values using this method.

Data for the following flight condition

$h = 20000$ $MACH = 0.9$ $\alpha = 0$ $\beta = 0$ $\dot{\alpha} = 0$ $p_s = 0$
 $q_s = 0$ $r_s = 0$ $\delta H = 0$ $\delta HA = 0$ $\delta FA = 0$ $\delta LEF = 0$
 $\delta r = 0$ $V_T = 933.2$

1.	CL	0.0218	24.	CN _P	0.0013
2.	CD	0.0216	25.	CY _r	0.6180
3.	CM	-0.0322	26.	Cl _r	-0.0579
4.	CY	0	27.	CN _r	-0.1060
5.	Cl	0	28.	Rl _P	0.9380
6.	CN	0	29.	RL _q	0.9160
7.	Cl _{δFA}	-0.0021	30.	RM _q	0.9790
8.	CN _{δFA}	-0.0003	31.	RY _{rVT}	0.9390
9.	Cl _{δHA}	-0.0013	32.	RN _{rVT}	0.9520
10.	CN _{δHA}	-0.0010	33.	RL _{δH}	0.9510
11.	CY _{δFA}	0.0006	34.	RM _{δH}	0.9490
12.	CY _{δHA}	0.0024	35.	RY _{δr}	0.7740
13.	ΔCL _{FT}	-0.0082	36.	Rl _{δr}	0.7630
14.	ΔCM _{FT}	0	37.	RN _{δr}	0.7960
15.	CY _{δr}	0.0035	38.	Rl _{δa}	0.7280
16.	CN _{δr}	-0.0019	39.	Rl _{δFA}	0.6560
17.	Cl _{δr}	0.0006	40.	RY _β	0.9630
18.	CL _q	3.3480	41.	Rl _β	0.9580
19.	CM _q	-2.6000	42.	RN _β	0.9430
20.	CL _α	-1.4095	43.	ΔCL _{FLEX}	0.0015
21.	CM _α	-1.4820	44.	ΔCM _{FLEX}	0.0013
22.	CY _P	-0.0012	45.	CL _{δH=0}	0.0218
23.	Cl _P	-0.2110	46.	CM _{δH=0}	-0.0322

Total Coefficient Values

CL _{TOTAL}	0.0150	Cl _{TOTAL}	0
CD _{TOTAL}	0.0216	CY _{TOTAL}	0
CM _{TOTAL}	-0.3347	CN _{TOTAL}	0

Data for the following flight condition

h = 20000 MACH = 0.8 $\alpha = 0$ $\beta = 0$ $\dot{\alpha} = 0$ $P_s = 0$
 $q_s = 0$ $r_s = 0$ $\delta H = 0$ $\delta HA = 0$ $\delta FA = 0$ $\delta LEF = 0$
 $\delta r = 0$ $V_T = 829.6$

1.	CL	0.0131	24.	CN _P	0.0019
2.	CD	0.0200	25.	CY _r	0.5910
3.	CM	-0.0255	26.	Cl _r	-0.0590
4.	CY	0	27.	CN _r	-0.1195
5.	Cl	0	28.	Rl _P	0.9553
6.	CN	0	29.	RL _q	0.9413
7.	Cl _{δFA}	-0.0020	30.	RM _q	0.9843
8.	CN _{δFA}	-0.0003	31.	RY _{rVT}	0.9577
9.	Cl _{δHA}	-0.0013	32.	RN _{rVT}	0.9667
10.	CN _{δHA}	-0.0010	33.	RL _{δH}	0.9643
11.	CY _{δFA}	0.0004	34.	RM _{δH}	0.9650
12.	CY _{δHA}	0.0022	35.	RY _{δr}	0.8380
13.	ΔCL_{FT}	-0.0255	36.	Rl _{δr}	0.8363
14.	ΔCM_{FT}	0	37.	RN _{δr}	0.8560
15.	CY _{δr}	0.0038	38.	Rl _{δa}	0.8033
16.	CN _{δr}	-0.0019	39.	Rl _{δFA}	0.7360
17.	Cl _{δr}	0.0007	40.	RY _{β}	0.9630
18.	CL _q	2.7880	41.	Rl _{β}	0.9580
19.	CM _q	-2.7500	42.	RN _{β}	0.9430
20.	CL _{$\dot{\alpha}$}	-0.7281	43.	ΔCL_{FLEX}	0.0013
21.	CM _{$\dot{\alpha}$}	-0.9280	44.	ΔCM_{FLEX}	0.0010
22.	CY _P	-0.0080	45.	CL $\delta H=0$	0.0131
23.	Cl _P	-0.2230	46.	CM $\delta H=0$	-0.0255

Total Coefficient Values

CL _{TOTAL}	-0.0143	Cl _{TOTAL}	0
CD _{TOTAL}	0.0200	CY _{TOTAL}	0
CM _{TOTAL}	-0.0265	CN _{TOTAL}	0

Data for the following flight condition

h = 20000 MACH = 0.9 $\alpha = 0$ $\beta = 2$ $\alpha = 0$ $p_s = 0.0349$
 $q_s = 0$ $r_s = 0.0349$ $\delta H = 0$ $\delta HA = 1$ $\delta FA = 1$ $\delta LEF = 0$
 $\delta r = -2$ $V_T = 933.2$

1.	CL	0.0218	24.	CN _p	0.0013
2.	CD	0.0216	25.	CY _r	0.6180
3.	CM	-0.0322	26.	Cl _r	-0.0579
4.	CY	-0.0442	27.	CN _r	-0.1060
5.	Cl	-0.0038	28.	Rl _p	0.9450
6.	CN	0.0053	29.	RL _q	0.9220
7.	Cl _{δFA}	-0.0021	30.	RM _q	0.9790
8.	CN _{δFA}	-0.0003	31.	RY _{rVT}	0.9430
9.	Cl _{δHA}	-0.0013	32.	RN _{rVT}	0.9510
10.	CN _{δHA}	-0.001	33.	RL _{δH}	0.9560
11.	CY _{δFA}	0.0006	34.	RM _{δH}	0.9570
12.	CY _{δHA}	0.0024	35.	RY _{δr}	0.7690
13.	ΔCL_{FT}	-0.0082	36.	Rl _{δr}	0.7670
14.	ΔCM_{FT}	0	37.	RN _{δr}	0.8010
15.	CY _{δr}	0.0035	38.	Rl _{δa}	0.7210
16.	CN _{δr}	-0.0019	39.	Rl _{δFA}	0.6470
17.	Cl _{δr}	0.0006	40.	RY _{β}	0.9510
18.	CL _q	3.3480	41.	Rl _{β}	0.9350
19.	CM _q	-2.6000	42.	RN _{β}	0.9240
20.	CL _{α}	-1.4095	43.	ΔCL_{FLEX}	0.0015
21.	CM _{α}	-1.4820	44.	ΔCM_{FLEX}	0.0013
22.	CY _p	-0.0012	45.	CL $\delta H=0$	0.0218
23.	Cl _p	-0.2110	46.	CM $\delta H=0$	-0.0322

Total Coefficient Values

CL _{TOTAL}	0.0150	Cl _{TOTAL}	-0.0075
CD _{TOTAL}	0.0216	CY _{TOTAL}	-0.0448
CM _{TOTAL}	-0.3347	CN _{TOTAL}	0.0069

DYNAMIC MODEL AND COCKPIT

It might be desirable to use the dynamic models with the cockpit, that don't currently interact with the cockpit. The process of changing the dynamics program to use the cockpit is rather simple. All that one has to do is assign PITCH and ROLL to a dummy name such as PITOUT and ROLLOUT and CONNECT these dummy names to CHVAR27 and to AOT31 respectively. Change the scaling of ROLL to 6.28, HEAD to 3.14, PS MAX to 2 and RS MAX to 2. Use S1.ILUR2 now instead of S1.ILUR or S1.ILUR1 and the oscilloscope display will be used. The use of S1.ILUR2 in place of the others is because S1.ILUR2 has the code that accepts PITCH and ROLL from SIMSTAR #115 and sends the appropriate values to the oscilloscope display.

To get the pilot inputs make the assignments FE=FEIN and FA=FAIN and make sure that the corresponding CONNECT statements are uncommented. Also, delete any commands that compute FE and FA that conflict with the above assignments and provide any conversion from pound force of the pilot force stick inputs to g's, degs/sec and degrees of the commanded inputs. Making the above simple changes should allow any of the current dynamic model programs access to the pilot inputs and oscilloscope display.

Appendix B: Intermachine Cable Hookups

Figure B.1 is a diagram depicting the back of the two SIMSTARs and the analog connections between them.

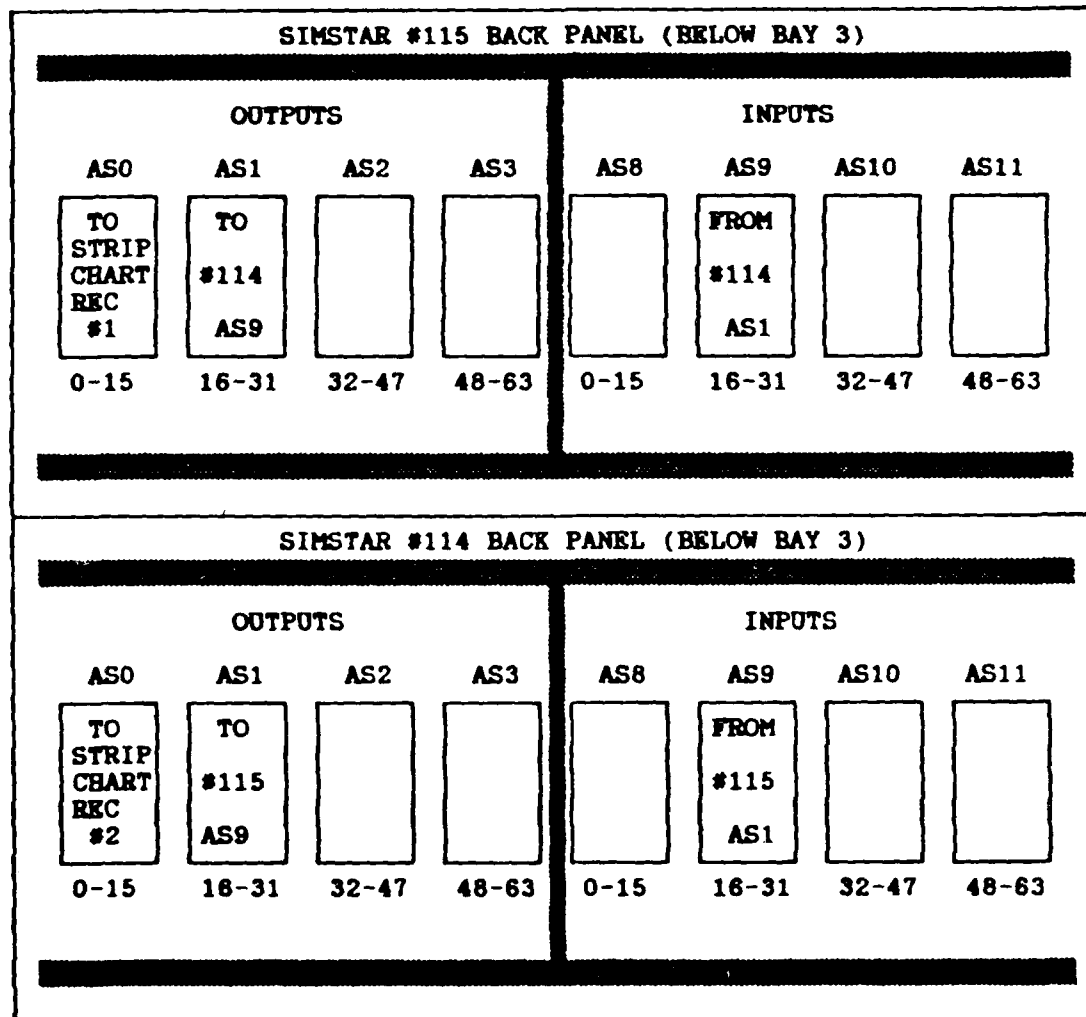
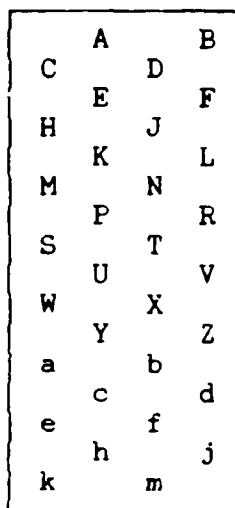


Figure B.1. SIMSTAR Back Panels

The diagram below shows one of the analog port blocks on the SIMSTAR backs and assigns a letter code to each port as assigned on the actual physical port component. Following the diagram is a list of the CONNECT components and each variable assigned to each particular port.



SIMSTAR #115 AS1 OUTPUT PORT

C	CHVAR21 = PSOUT	A	GND
B	CHVAR22 = QSOUT	D	GND
H	CHVAR23 = RSOUT	E	GND
F	CHVAR24 = HOUT	J	GND
M	CHVAR25 = BETOUT/BETA	K	GND
L	CHVAR26 = VTOUT/VT	N	GND
S	CHVAR27 = PITOUT	P	GND
R	CHVAR28 = DHAOUT	T	GND
W	XYPLH2 = DFAOUT	U	GND
V	XYPLV2 = DROUT	X	GND
a	AOT26 = DLEOUT	Y	GND
Z	AOT27 = ADOOUT	b	GND
e	AOT28	c	GND
d	AOT29 = AOAOUT	f	GND
k	AOT30 = DHOUT	h	GND
j	AOT31 = ROLLOUT/ROLL	m	GND

SIMSTAR #115 AS9 INPUT PORT

C	AIN16		A	GND
B	AIN17		D	GND
H	AIN18		E	GND
F	AIN19		J	GND
M	AIN20	= ITIN	K	GND
L	AIN21	= YTIN	N	GND
S	AIN22	= NTIN	P	GND
R	AIN23	= LTIN	T	GND
W	AIN24	= DTIN	U	GND
V	AIN25	= MTIN	X	GND
a	AIN26	= FEIN	Y	GND
Z	AIN27	= FAIN	b	GND
e	AIN28	= FRIN	c	GND
d	AIN29	= THIN	f	GND
k	AIN30	= ETIN	h	GND
j	AIN31		m	GND

SIMSTAR #114 AS9 INPUT PORT

C	AIN16	= PSIN	A	GND
B	AIN17	= QSIN	D	GND
H	AIN18	= RSIN	E	GND
F	AIN19	= HIN	J	GND
M	AIN20	= BETAIN	K	GND
L	AIN21	= VTIN	N	GND
S	AIN22	= PITCH	P	GND
R	AIN23	= DHAIN	T	GND
W	AIN24	= DFAIN	U	GND
V	AIN25	= DRIN	X	GND
a	AIN26	= DLEFIN	Y	GND
Z	AIN27	= ADOTIN	b	GND
e	AIN28		c	GND
d	AIN29	= AOAIN	f	GND
k	AIN30	= DHIN	h	GND
j	AIN31	= ROLL	m	GND

SIMSTAR #114 AS1 OUTPUT PORT

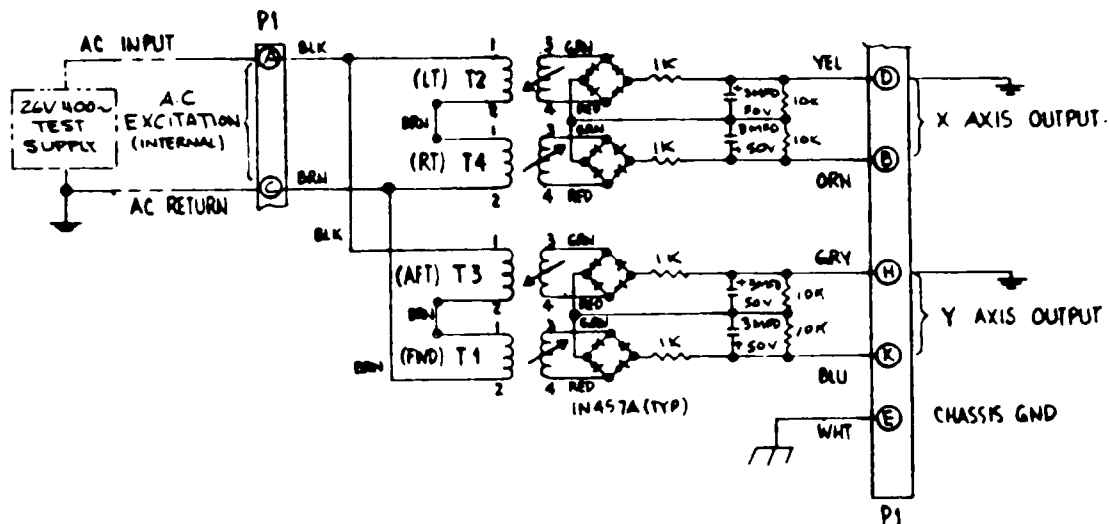
C	CHVAR21 = ITOUT	A	GND
B	CHVAR22 = YTOUT	D	GND
H	CHVAR23 = NTOUT	E	GND
F	CHVAR24 = LTOUT	J	GND
M	CHVAR25 = DTOUT	K	GND
L	CHVAR26 = MTOUT	N	GND
S	CHVAR27	P	GND
R	CHVAR28	T	GND
W	XYPLH2 = PITOUT	U	GND
V	XYPLV2 = X	X	GND
a	AOT26	Y	GND
Z	AOT27	b	GND
e	AOT28 = Y	c	GND
d	AOT29	f	GND
k	AOT30	h	GND
j	AOT31	m	GND

As shown in the last list above, the variables PITOUT, X and Y are output from SIMSTAR #114. These variables drive the oscilloscope display. The variable Y is the input for the Y channel on the scope. The variable X is the input for the primary X input on the scope and PITOUT is the input to the second X input on the scope.

Also shown, in the input list from SIMSTAR #115 are the pilot stick inputs. These variables, FEIN and FAIN, are explained in Appendix C.

Appendix C: Pilot Control Schematics

This appendix contains the schematics (Figures C.1 and C.2) for the Two Axis Hand Control, Model 435 DC, made by Measurement Systems, Inc., which is used in this thesis for the pilot force stick inputs of elevator (FEIN) and aileron (FAIN). A signal generator outputting a 1KHZ sine wave of amplitude about ± 15 volts peak to peak (maximum of the signal generator) is used as the input power supply to the hand control. The X-axis of the hand control is used for aileron control and outputs the variable FAIN to the analog port block of SIMSTAR #115 (as specified in Appendix B). The Y-axis of the hand control is used for elevator control and outputs the variable FEIN to the same analog port block of SIMSTAR #115 (as specified by Appendix B).



7. BALANCE CONTROL: INTERNAL MECHANICAL.
 6. OUTPUT (AT 26V, 400 CPS TEST SUPPLY): 0.43 VDC / LB PHASE REVERSING UP TO 10 POUNDS FORCE MAX.
 5. POWER SUPPLY FREQUENCY: 50 TO 2000 CPS.
 4. POWER SUPPLY VOLTAGE: 12 TO 28 VDC.
 3. MATING CONNECTOR: MRA-MSGHVL SUPPLIED.
 2. MOUNTING RING MAY BE ROTATED TO ANY POSITION BY LOOSENING SERVO CLIPS.
- DIMENSIONS IN PARENTHESES DENOTE MILLIMETERS.

Figure C.1. Hand Control Electrical Schematic

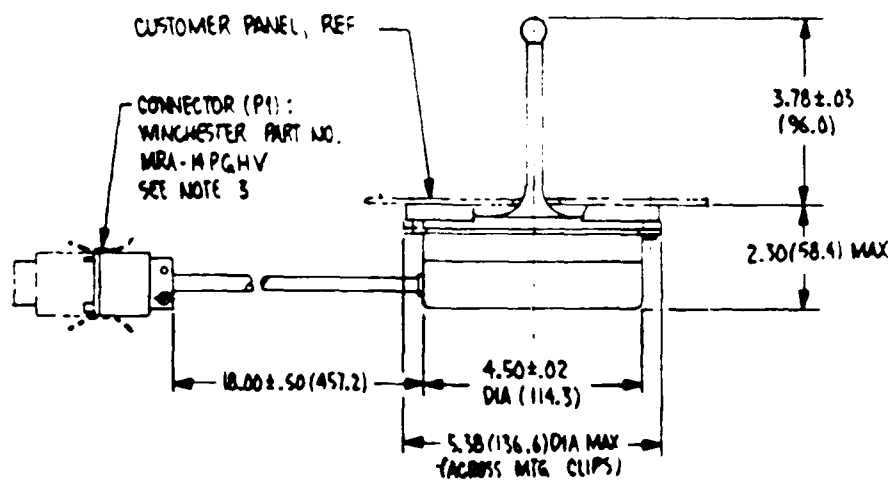
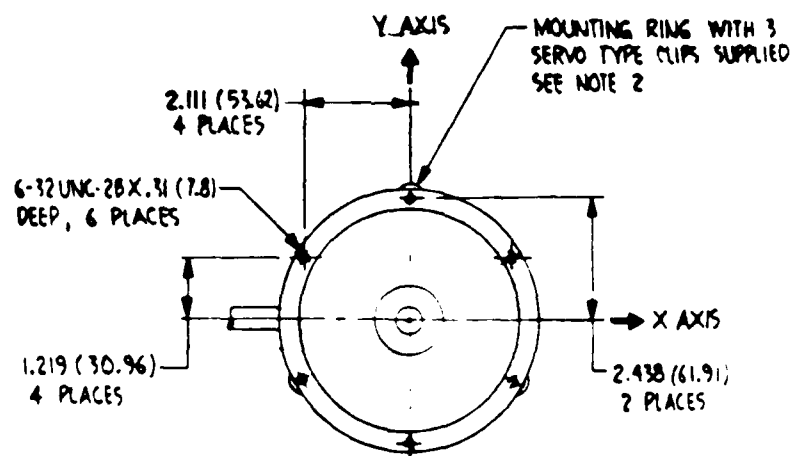


Figure C.2. Hand Control Mechanical Schematic

Appendix D: Cockpit Graphics Display

Overview

This appendix explains some work that was done with the Evans and Sutherland PS340 located in RM 245. This work is important to this thesis because it demonstrates a potential for a real-time graphics display as a component of a man-in-the-loop simulator and provides information that may help with future work in this area.

The PS340 is connected to SIMSTAR #115 through a direct RS-232 port. Graphics programs that currently reside on AFIT's ISL VAX 780 and on a 5.25" floppy disc are downloaded into the PS340 after the PS340 is reconfigured to a PS330 terminal. This is explained in the following.

Setting up the PS340

The Evans and Sutherland PS340 graphics terminal is reconfigured to a PS330 terminal in order to utilize the special user-update function, F:USERUPD. The PS340 in its normal form does not have this function.

The PS330 firmware package has this function because it does not have the special rendering functions of the PS340. The PS330 firmware consists of two diskettes labelled 'PS330 ASYNC DISKS.' These disks are located in the software data-case next to the PS340. One is labelled 'disk A' and the other is labelled 'disk B.' With the system powered down, place disk A in drive 1 and disk B in drive 2. Turn on the power and wait

for the monochrome screen to display the message 'PS330 A2.V01.' At this time, some error messages might appear. Ignore these messages and enter the Command Interpreter mode by pressing the 'SHFT' and 'LINE/LOCAL' keys simultaneously. Then press the 'RETURN' key. The '@@' prompt should be displayed on the screen signifying the PS330 is in the Command Interpreter (CI) mode.

Accessing the VAX 780 for Graphics

To access the VAX 780, first make sure the switch on the port box on top of the disk drive cabinet is set to 'A-VAX 780'. Then type the following at the CI prompt.

```
@@ SETUP INTERFACE PORT10/SPEED=19200;
```

Press 'RETURN', followed by 'LINE/LOCAL' and 'RETURN' again and the VAX 780 should give the login header and ask for the username for login.

Once logged into a directory that contains the graphics programs (the user directory, MKASSAN under DUA1 has these programs), display the graphics by typing the following with each line followed by a 'RETURN' key.

```
ty panel.dat  
ty ball.dat  
ty inst.dat  
ty amark.dat  
ty frame.dat  
ty track.dat
```

It will take time between lines especially after 'ty ball.dat.' The typing of these files to the PS330 screen is in effect executing them and no text will appear on the screen.

The graphics will form piece by piece as the files are typed to the screen. Next, logout out of the VAX 780. Then again enter the Command Interpreter mode by pressing simultaneously the 'SHFT' and 'LINE/LOCAL' keys followed 'RETURN'. The CI prompt, '@@' will appear.

Accessing the SIMSTAR for updates

Accessing the SIMSTAR is done by typing the following series of lines in the CI mode.

```
@@ setup interface port10/speed=9600;  
@@ setup interface port10/bits per character=7;  
@@ setup interface port10/stop bits per character=2;  
@@ setup interface port10/even parity;
```

After typing the following, switch the port box on top of the disk drive cabinet to 'C-SIMSTAR' and make sure that the raster screen is OFF. Next press the 'LINE/LOCAL' key and 'RETURN' to enter the terminal emulator mode. Make sure that the connection to the SIMSTAR is made by pressing simultaneously the 'SHFT' and '?' keys. The SIMSTAR header will be displayed and it will ask for an ownername. Press the 'RETURN' key. If the the SIMSTAR header doesn't come up, check to make sure that the PS340 is connected to SIMSTAR #115 at the SIMSTAR end. The EAI representative will be able to help with this.

Sending data to a terminal from a SIMSTAR program must be done from a FORTRAN "write" statement in the DYNAMIC region. The FORTRAN "write" statement must use the UNIT=1 in it. This is only because when loading using the SIMTACS macro, SIMTACS makes the assignment of the PS340 terminal port number to the

Logical File Code (LFC) of 1. But this can be set to almost any number. SIMTACS does it with the single command as follows.

ASSIGN 1 TO DEV=TY7EC1

This command can also be completed at the TSM prompt. The PS340 is connected as terminal number TY7EC1. The author has been able to send data to the PS340 from another SIMSTAR terminal, but has not been able to get the data to update the graphics. A file is written on the VAX 780 that executes while the PS340 and the VAX are connected and its data is able to update the graphics. So the author knows that the problem is not on the PS340 end, but probably with the format the SIMSTAR is sending its data in. The format, though, is decided by the programmer so the author obviously made some mistakes somewhere in how he was sending the data. The speed the data is getting to the PS340 from the SIMSTAR in this setup is a larger problem though. As stated earlier in the report, and as understood so far, the speed is too slow for updating in real-time.

Manipulating Graphics Programs

This appendix cannot go into the semantics of programming the PS340. Refer to the Evans and Sutherland PS300 manuals for help in programming.

The graphics programs, though, can be changed using the VAX 780 editor while they reside on the VAX. This makes editing the programs very simple.

An important point about these programs deal with routing bytes. Analyzing one of the graphics programs, one will notice the characters '^0' at the beginning and the characters '^>' at the end. When the program is 'typed' to the screen of the PS340, as described earlier in this appendix about loading the graphics programs into the PS340, the PS340 sees the beginning characters and recognizes them as routing bytes. These routing bytes send the code to the designated location. In this case, the code is sent to the graphics part of the PS340. This makes the code read to the screen as the graphics, making the cockpit display. The characters at the end of the code tell the PS340 to stop sending the code to graphics and return to the terminal emulator mode returning the prompt after the graphics are done.

Appendix E: Simulator Source Code

This appendix lists the various programs that reside on the SIMSTARs and the ISL VAX 780. These programs are not included as part of this thesis because of their bulk. Instead, they are include in a separate document listed as Reference (20) in the Bibliography.

The following routines reside on SIMSTAR #115 as part of SIMTACS.

S1.DLONG	Digital longitudinal aircraft (uses S1.ILUR)
S1.DLAT	Digital lateral-directional aircraft (uses S1.ILUR) NOTE: There are problems with this program dealing with transfer of a PADC variable. Sometimes it works. The problem has been brought up to the attention of the EAI representative but has not been solved as of this printing.
S1.DUAC	Digital inertially uncoupled aircraft with ideal (ideal inputs are described in Appendix A under Coding SIMTACS) inputs (uses S1.ILUR1). S1.DUAC1 is identical except for the '@RECORD' statement.
S1.DCAC	Digital inertially cross-coupled aircraft with ideal (ideal inputs are described in Appendix A under Coding SIMTACS) inputs (uses S1.ILUR1) S1.DCAC1 is identical except for the '@RECORD' statement.
S1.IDCAC	Digital inertially cross-coupled aircraft with human pilot inputs and oscilloscope display. (uses S1.ILUR2)

For those designers interested in the design of an analog controller the following programs are available. Due to the number of analog components that are used for an analog

controller, the programs cannot offer the coupling feature available with a digital controller.

- S1.ALONG Analog longitudinal aircraft (uses S1.ILUR) with ideal inputs
- S1.ALAT Analog lateral-directional aircraft (uses S1.ILUR) with ideal inputs. There may be problems with this program. The program has a statement whose value is supposed to be zero but the SIMSTAR assigns the value of 1.026 to it. It might be solved with the new EAI system software. This problem has been brought up to the attention of the EAI representative but has not been solved as of this printing.

The following programs do various other functions as described.

- S1.TEST This is a test program for checking values coming from SIMSTAR #114 for various aircraft state values sent to SIMSTAR #114.
- SIMTACS This is a macro for loading and running of any dynamic model programs. Its original intent was to provide the extra assignment necessary for the graphics terminal as explained in Appendix D.

The following programs reside on SIMSTAR #114.

- S1.ILUR Interpolation and Look-up program used for separate lateral and longitudinal dynamic model programs with ideal inputs.
- S1.ILUR1 Interpolation and Look-up program used for combined lateral/longitudinal dynamic model programs with ideal inputs.
- S1.ILUR2 Interpolation and Look-up program used for running the cockpit display and the pilot inputs.
- FILELIST A file that is read by the ILUR programs. It specifies data values and breakpoint variables so that the ILUR programs can use the file AC.DAT

AC.DAT This file contains the F-16A aerodynamic data such
 that each value in it is one record to be read by
 the ILUR programs.

SIMILUR This is the macro that compiles the ILUR programs.
 It contains extra assignments so that FILELIST and
 AC.DAT can be read during the run. Its format is:
 SIMILUR ILUR,1,F

The following graphics programs reside on the ISL VAX 780.

PANEL.DAT
BALL.DAT
FRAME.DAT
INST.DAT
AMARK.DAT
TRACK.DAT

These programs make up various parts of the graphics cockpit display on the Evans and Sutherland PS340 as described in Appendix D. Program PANEL.DAT contains the major code for making necessary connections for updates. Program BALL.DAT provides the detail for the attitude ball. Program FRAME.DAT contains code for the various borders. Program INST.DAT completes many of the instrument displays. Program AMARK.DAT places scales on the altitude instrument and program TRACK.DAT prints to the screen a track for an airplane to follow located below the instruments.

Bibliography

1. Arpasi, Dale J. and Richard A. Blech. "Hardware for a Real-time Multiprocessor Simulator," Distributed Simulation 1985. 43-49. La Jolla, CA: Simulation Councils, Inc., 1985.
2. Blakelock, John H. Automatic Control of Aircraft and Missiles. New York: John Wiley and Sons Inc., April 1965.
3. Dikilitas, Zeki Evaluation of the Angle of Attack Limiter of the F-16 C/D Aircraft. MS thesis, AFIT/GE/ENG/86D-9. School of Engineering, Air Force Institute of Technology (AU), Wright Patterson AFB OH, December 1986.
4. D'Azzo, John J. and Constantine H. Houpis. Linear Control Systems, Analysis and Design. New York: McGraw-Hill Book Company, 1981.
5. D'Azzo, John J. and Constantine H. Houpis. Class handout distributed in EENG 708, Multivariable Control Systems. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, March 1987.
6. Etkin, Bernard. Dynamics of Atmospheric Flight. New York: John Wiley & Sons, Inc., 1972.
7. Grodsky, Milton A. "Man-Machine Simulation," Prospects for Simulation and Simulators of Dynamic Systems. 83-103. New York: Spartan Books, 1967.
8. Houpis, Constantine, H. Quantitative Feedback Theory (QFT) Technique for Designing Multivariable Control Systems. AFWAL-TR-86-3107. Flight Dynamics Laboratory, Wright -Patterson AFB, OH. January 1987.
9. Houpis, C. H. and Gary B. Lamont. Digital Control Systems Theory, Hardware, Software. New York: McGraw-Hill Book Company, 1985.
10. Maybeck, Peter S. Stochastic Models, Estimation, and Control, Volume 3. New York: Academic Press, 1982.
11. Naville, G. L. and others AFTI-16 Simulation Data Package, Volume 1. Contract F33615-78-c-3022 Project 2061, General Dynamics, Fort Worth Division. 23 September 1983.
12. Rodger, M. and G. Shapiro. Prospects for Simulation and Simulators of Dynamic Systems. Preface. New York: Spartan Books, 1967.

13. Roskam, Jan. Flight Dynamics of Rigid and Elastic Airplanes Part One. Lawrence, KA: Roskam Aviation and Engineering Corporation, 1972.
14. Ruley, J. M. F-16 Simulation User's Guide. (Version 1), Unpublished report used by ASD/ENFTC. Wright-Patterson AFB OH, 12 May 1987.
15. Shinn, Capt Thomas B. and Frederick E. Unfried. "Evaluation of the AFTI/F-16 on the LAMARS (Large Amplitude Multi-Mode Aerospace Research Simulator)," Aerospace Simulation, 13: 60-66 (February 1984).
16. Simstar Digital Translator Reference Manual. EAI, 1985.
17. Simstar Parallel Translator Reference Manual. EAI, 1985.
18. Truitt, Robert Wesley. Intermediate Aerodynamics. New York: Pitman Publishing Corp., 1944.
19. Barfield, Finley. Expert engineer on the F-16 and AFTI/F-16. Personal interview. AFTI/F-16 Office, Wright-Patterson AFB OH, 25 November 1987.
20. Kassan, Mark W., SIMTACS Program Source Code. Unpublished listings of the program code that make up SIMTACS on the SIMSTAR computer systems., December 1987.
21. F-16A OHT 'CLEAN' Aerodynamic Data. Obtained from Jim Ruley on magnetic tape, ASD/ENFTC, Wright-Patterson AFB OH.*
22. Block 25 F-16 Digital Flight Control System Functional Block Diagrams. Data obtained from Lt. Bruce Peet, F-16 System Program Office, Wright-Patterson OH.

* This data package is only representative of the F-16A OHT and ASD/ENFTC does not guarantee its validity.

VITA

Second Lieutenant Mark W. Kassan was born on 3 November 1964 in Merced, California. He graduated from high school in Medford, New Jersey, in 1982 and attended Syracuse University from which he graduated Cum Laude with a Bachelor of Science with honors in Electrical Engineering in May 1986. Upon graduation he received a commission in the United States Air Force through the ROTC program. He entered the School of Engineering, Air Force Institute of Technology in June 1986.

Permanent Home Address: 105 Hickory Lane
Medford, New Jersey 08055

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution unlimited.	
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE			5. MONITORING ORGANIZATION REPORT NUMBER(S)	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GE/ENG/87D-30			7a. NAME OF MONITORING ORGANIZATION	
6a. NAME OF PERFORMING ORGANIZATION School of Engineering	6b. OFFICE SYMBOL (If applicable) AFIT/ENG	7b. ADDRESS (City, State, and ZIP Code)		
6c. ADDRESS (City, State, and ZIP Code) Air Force Institute of Technology Wright-Patterson AFB, Ohio 45433		9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER		
8a. NAME OF FUNDING / SPONSORING ORGANIZATION	8b. OFFICE SYMBOL (If applicable)	10. SOURCE OF FUNDING NUMBERS		
8c. ADDRESS (City, State, and ZIP Code)		PROGRAM ELEMENT NO	PROJECT NO.	TASK NO
		WORK UNIT ACCESSION NO		
11. TITLE (Include Security Classification) See Box 19				
12. PERSONAL AUTHOR(S) Mark W. Kassan, B.S.E.E., 2d Lt, USAF				
13a. TYPE OF REPORT MS Thesis	13b. TIME COVERED FROM _____ TO _____	14. DATE OF REPORT (Year, Month, Day) 1987 December	15. PAGE COUNT 175	
16. SUPPLEMENTARY NOTATION				
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB-GROUP	Flight Control Systems, Flight Simulation, Fighter Aircraft, Hybrid Computers, Man-in-the-Loop, Human Factors Engineering	
01	04			
19. ABSTRACT (Continue on reverse if necessary and identify by block number)				
Title: F-16 SIMULATOR FOR MAN-IN-THE-LOOP TESTING OF AIRCRAFT CONTROL SYSTEMS (SIMTACS)				
Thesis Chairman: Zdzislaw H. Lewantowicz, Lt Colonel, USAF				
Approved for public release: 10W 37A 4004 Lynn Williams 24 Feb 88 Air Force Institute of Technology Development Wright-Patterson AFB, Ohio 45433-6100				
20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL Zdzislaw H. Lewantowicz, Lt Colonel, USAF			22b. TELEPHONE (Include Area Code) 513-255-6913	22c. OFFICE SYMBOL AFIT/ENG

The focus of this project is the development of a real-time flight simulator for man-in-the-loop evaluation of flight control system designs. The F-16A is the simulated aircraft and a simplified F-16 digital flight control system is used. The simulation models nonlinear aerodynamic data over most of the flight envelope and nonlinear inertially cross-coupled aircraft dynamics.

The simulator is implemented on two Electronic Associates, Inc. SIMSTAR hybrid computer systems and an initial cockpit is designed for human pilot interaction. The simulated aircraft dynamic behavior is validated using AFTI/F-16 closed-loop time responses provided by the Air Force Flight Dynamics Laboratory.

The real-time, man-in-the-loop simulator is designed using the SIMSTAR computer systems. Currently, the simulator performance is limited in the speed of response. This temporary lack of speed is the result of the computational limitations of the current digital processors of the SIMSTAR computer systems. These limitations can be removed with hardware upgrades.

FILM
4