AUTOMATED SAFETY AND TRAINING AVIONICS FOR GENERAL AVIATION AIRCRAFT

A Thesis

by

JEFFREY ALAN TRANG

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

May 1997

Major Subject: Electrical Engineering

AUTOMATED SAFETY AND TRAINING AVIONICS FOR GENERAL AVIATION AIRCRAFT

A Thesis

by

JEFFREY ALAN TRANG

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Approved as to style and content by:

John H. Painter
(Chair of Committee)

Norman C. Griswold
(Member)

Karan L. Watson
(Member)

Donald T. Ward
(Member)

Norman C. Griswold
(Head of Department)

May 1997

Major Subject: Electrical Engineering

ABSTRACT

Automated Safety and Training Avionics for

General Aviation Aircraft. (May 1997)

Jeffrey Alan Trang, B.S., Rose-Hulman Institute of Technology

Chair of Advisory Committee: Dr. John H. Painter

The past decade has seen the U.S. general aviation community plagued by substantial cost increases while operating in an increasingly complex and crowded air traffic control structure. Unfortunately, there has been a corresponding rise in accident rates involving these aircraft. In an attempt to improve safety factors and training programs for this aviation sector, researchers at Texas A&M University are investigating "smart cockpit systems." This research program is titled *Automated Safety and Training Avionics* (ASTRA).

ASTRA research is focused on integrating low-cost, yet sophisticated, computing technology into general aviation aircraft. The system architecture includes a *Flight Mode Interpreter* (FMI), which provides real-time identification of the aircraft operational maneuvering mode, through interpretation by *fuzzy logic* of aircraft state variables. This inference controls a *Head-Up Display* (HUD) to automatically present a *unique* display format appropriate to the operational situation. The FMI also drives a rule-based *Pilot Advisor* for generation of *alarms* and piloting advice. The pilot communicates with ASTRA through the *Head-Down Display* (HDD), which is configured similarly to the Multi-Function

Displays found in many "glass cockpit" aircraft. This configuration permits the pilot to readily access, edit, and display a wide variety of information.

The research reported in this thesis was to formally define the performance and test specifications for ASTRA and its various subsystems, as well as to design the system displays. Performance of these research tasks drew heavily on the author's experience as an Army experimental test pilot. Because the FMI is a unique development in modern aeronautics, definition of its functionality and integration with other system components could not rely on existing methodology and called for a imaginative approach. Likewise, design of the HUD and HDD display formats, as integrated with the FMI, was equally challenging.

It is hoped that the research contributions of this thesis will form a firm foundation for the implementation and evaluation of the ASTRA system. It is felt that the success of the system will hinge on its functionality and perceived utility from the perspective of the general aviation pilot.

DEDICATION

"J'aime et j'espere."

—Thomas Jefferson

To Dianna, Allen, and Amy, for their love and patience.

ACKNOWLEDGMENTS

"I just want to thank everyone who made this day necessary."

—Yogi Berra

I owe special thanks to those individuals who made it possible for me to pursue this degree. In particular, LTC Gary Sharon (a fellow Aggie!) and LTC David Downey were instrumental in my receiving an Army graduate fellowship. Likewise, I would not have been accepted into the Electrical Engineering program at Texas A&M University without the kind recommendations, support, and vision of Dr. Ronald Anderson (Department Head of Mathematics at Texas Tech University), LTC Kevin Scherrer (presently serving in Korea as a battalion commander), and LTC Travis Simpson (who's never been short on "vision").

I am particularly indebted to those serving on my graduate advisory committee: Dr. John Painter, Dr. Norman Griswold, Dr. Karan Watson, and Dr. Donald Ward. They were always available to assist and answer my many questions, and allowed me to take frequent advantage of their time. Their guidance was especially helpful in tailoring my degree plan and research efforts to the benefit of both the Army and my career goals. Any errors found in this thesis are mine alone, and should not be attributed to them. Their efforts have made my brief tour in Aggieland personally and professionally rewarding.

I would also like to acknowledge the fellow students who toiled along side me during the past 21 months—their blood, sweat, and tears make up the soul of this research effort. Many warm thanks go out to B.J. Alcorn, Paul Branham, Karthik Krishnamurthy, Kris Lee, Tay Nguyen, Drew Robbins, and Dick Woo. Justin Ridge and Wallace Kelly were particularly helpful in laying out and formatting this manuscript. A special word of gratitude also goes to Wally, whose computer programming skills and expertise on artificial intelligence applications are second to none. Were this thesis written as elegantly as his computer code, it would undoubtedly be 60% shorter. I wish each of you success and happiness as you finish your studies at Texas A&M, and in all your future endeavors.

I am further grateful to my parents, who have always supported my many undertakings and follies, and taught me the importance of education. They have sacrificed much through the years for me, my sister, and my family.

Last, but certainly not least, I must thank my beautiful wife Dianna. My achievements would have been impossible without her constant support and undying belief in my abilities. She bolstered me when I wanted to quit, chided me when my head grew too large, and never let me forget the importance of maintaining my sense of humor. *"With all my heart and soul..."*

J. Trang
College Station, Texas
March 1997

TABLE OF CONTENTS

Page

LIST OF FIGURES

FIGURE Page

FIGURE Page

LIST OF TABLES

INTRODUCTION

"Research is what I'm doing when I don't know what I'm doing."

—Wernher von Braun

Background

There is presently a significant amount of technology being investigated in the use of "smart cockpit computing systems" for assisting the pilot in flying his aircraft. One such research program resulted in the development of the *General Aviation Pilot Advisor and Training System* (GAPATS) [20]. This system was developed through a collaborative effort between the Departments of Aerospace Engineering and Electrical Engineering at Texas A&M University (TAMU), and Knowledge Based Systems, Inc. (KBSI), of College Station, Texas. The primary goal of the GAPATS program was to improve flight management, control, safety, and training for the *general aviation* (GA) pilot flying in today's rapidly evolving air traffic control (ATC) and airspace structures. Other so-called "intelligent" cockpit systems have concentrated in more sophisticated aviation arenas, such as airliners, airfreight, and military aircraft. However, these other intelligent systems require the use of complex and expensive equipment, making their use impractical for GA.

This thesis follows the format of *IEEE Transactions on Fuzzy Systems*.

A secondary goal of GAPATS was to develop a commercially viable product—affordable enough for the GA pilot to buy and reliable enough for flight certification by the Federal Aviation Administration (FAA). To this end, the GAPATS system was designed to operate on a Pentium-class computer. This arrangement also facilitated integrating GAPATS into the TAMU Engineering Flight Simulator (EFS), maintained by the Department of Aerospace Engineering under the direction of Professor Donald Ward. For the development of GAPATS, the EFS was modeled as a Commander-700 (a light twin-engine GA aircraft). This particular aircraft model was chosen because TAMU also maintains a fully instrumented Commander-700 as a research/flight test platform. Consequently, this arrangement was seen to permit a smooth transition from simulation to flight demonstration as GAPATS research matured. Recent upgrades to the EFS facilities (both hardware and software) have significantly enhanced the EFS as an engineering research tool.

GAPATS System Architecture

The GAPATS system architecture consisted of several major subsystems, as depicted in Figure 1. Phase I of the research effort focused primarily on the development of the Flight Mode Interpreter (FMI), using *fuzzy logic* algorithms. The fundamental task of the FMI was to produce continuous estimates of the aircraft's operational flight mode in terms of aircraft state information. Harral [10] modeled these basic aircraft flight modes, which included *taxi*, *takeoff*, *climbout*, *cruise*, *initial approach*, *final approach*, and *landing*. Aircraft state information included such variables as *airspeed*, *altitude*, *rate of climb*, and *engine power level*. By using these state variables as input parameters, the FMI could automatically determine the current flight mode. That is, no additional informational inputs were required from the pilot. However, the implemented system was sensitive to variations in pilot technique, resulting in unreliable system output—data that was "nervous" (constantly changing) or inaccurate.

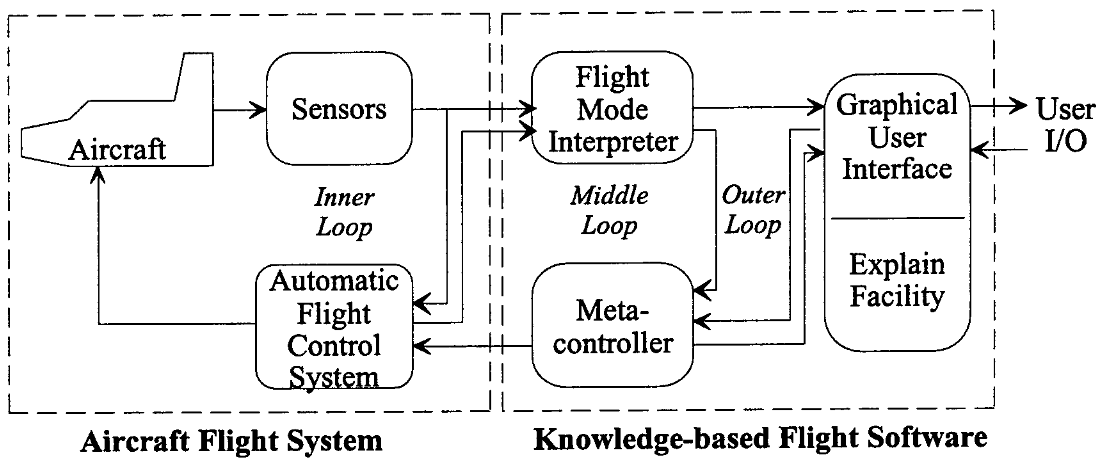


Figure 1. Original GAPATS System Architecture

Data generated by the FMI was displayed to the pilot on a graphical user interface (GUI), which was essentially an engineering flight display, rather than a useful cockpit display. Display output included: (1) the current inferred flight mode; (2) the flight mode which the aircraft appeared to be pursuing; (3) the confidence and certainty values of the flight mode inference; and (4) any of a number of *alarms*, which were displayed when GAPATS detected piloting errors were being made. For example, the GUI might display the alarm, "Airspeed is Inappropriate for Mode: Cruise." Unfortunately, alarms such as these tended to be vague (if not confusing): is the airspeed inappropriate because it is too high or too low? Further, the alarms did little to tell the pilot what corrective action might be appropriate.

GAPATS Matures

As GAPATS research continued, it was seen that the architecture for the pilot advisory system , now called *Automated Safety and Training Avionics* (ASTRA), would require significant modification. The changes, reflected in Figure 2, are summarized in the paragraphs below. As with GAPATS, the ASTRA architecture was designed such that the system could be ported from the EFS to the Commander-700 without additional modification. That is, the ASTRA software would be client-transparent.

The modified system architecture shows that ASTRA is no longer directly coupled to the aircraft through the Automatic Flight Control System. In other words, the pilot retains complete control of flying the aircraft. The figures also reveal that the developmental GUI has been replaced by two components: a *Head-Up Display* (HUD) and a *Head-Down Display* (HDD). The former is a primary flight display, used for pilotage (controlling the aircraft) and navigation; the latter is a secondary flight display, generally used for navigation, flight planning, and data entry. Despite being classified as a "secondary" display, the design of the HDD and its functionality are key to the utility of the entire ASTRA system, since the HDD serves as the data-entry unit for the pilot to communicate with ASTRA.

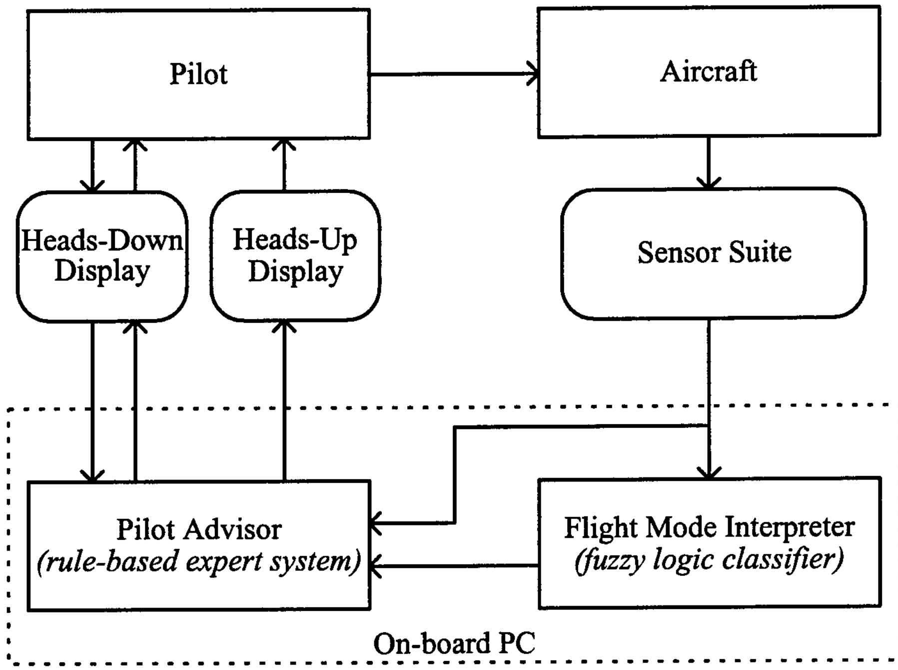


Figure 2. ASTRA System Architecture

Finally, a comparison of the figures reveals that the original GAPATS Meta-controller has been replaced by a Pilot Advisor (PA). Like the FMI, the PA is a rule-based expert system. Unlike the FMI, the PA uses a "crisp" rule-base rather than a fuzzy rule-base. Consequently, the PA will be able to automatically determine display configurations (of both the HUD and HDD) as a function of sensor data, FMI output, and pilot desires.

GAPATS Limitations

While early GAPATS research demonstrated the feasibility of a fuzzy rule-based flight mode interpreter, it demonstrated several fundamental limitations. The first, previously alluded to, was that GAPATS was sensitive to variations in pilot technique, resulting in unreliable output from the FMI. Clearly the robustness of the FMI would require improvement, so that the FMI would provide correct flight mode inference, independent of aircraft configuration or pilot technique.

The second, also noted earlier, relates to how GAPATS displayed any *alarms* to the pilot whenever the system interpreted that piloting errors were being made. For example, alarms could be generated not only for an incorrect aircraft *configuration* (such as an inappropriate flap setting for take-off), but also for an improper aircraft *state* (such as excessive airspeed while on final approach), each as a function of flight mode. However, the initial system design was limited in that it only told the pilot that a particular configuration or state was "inappropriate" for the current mode. In most circumstances, such alarms tended to be difficult to interpret, making it difficult for the pilot to make timely, appropriate corrective action. To reiterate an earlier example, if the FMI generated an alarm such as, "Airspeed is Inappropriate for Mode: Cruise," then was the pilot flying too slow or too fast? Further, how much of an airspeed correction should the pilot make? In other words, alarms generated by GAPATS indicated nothing more than that piloting errors were being made, but did little to indicate what type of corrective action might be appropriate.

Third, only seven possible flight modes (as detailed in previous sections) were identified. Clearly, the FMI would need the ability to properly interpret additional flight modes. For example, should ATC give a clearance for the pilot to "*climb* and maintain 9,000 feet" from a present altitude of 7,000 feet, it would be possible for FMI to interpret the aircraft's mode as *take-off* or *climbout*. Similar arguments could be made when the pilot executes a *missed approach* (an approach that does not terminate with a landing, because of adverse weather, for example). In short, the possibility of adding new flight modes to the FMI's repertoire (to further enhance system robustness) required investigation.

Finally, the initial GAPATS design tended to focus only on *non-nominal flight* conditions, such as those when piloting errors have been made. To make the system more powerful, ASTRA would need the ability to address *nominal* flight conditions, in which the pilot has made no piloting errors. For example, ASTRA could monitor the aircraft's fuel state, recommend an appropriate heading to intercept an airway, recommend the optimum point to begin a descent as the aircraft approaches its destination, or display appropriate checklist information to the pilot. Only by defining both types of flight condition—nominal and non-nominal—could ASTRA truly be used as both a *safety* and a *training* system, as its name implies.

The Future of ASTRA

The initial phase of the GAPATS research program demonstrated that the essential flight interpretation scheme would work. Based on that result, the basic ASTRA architecture of Figure 2 was postulated. Before ASTRA can be fully designed and implemented in software, however, a thorough top-down system design is necessary. This system design includes defining its performance specification, which details the functionality of the total system while addressing limitations of earlier designs. Likewise, the functionality requirements for each of the ASTRA subsystems must be specified.

Whereas defining a functional specification for ASTRA is the start point for the ASTRA system development, the end point is a performance evaluation of the system, as it is implemented in software. Such evaluation can be most readily accomplished using the EFS, wherein GAPATS was originally evaluated. In fact, since the EFS is part of the system software development environment, evaluation may be done incrementally as each of the subsystems are developed, augmented, or modified.

Many of the tasks described in the preceding two paragraphs require an expert knowledge of flight operations, of software development and integration, and of system performance evaluation. The author of this thesis, with his qualifications as a research test pilot, possesses the background which uniquely fits him for performing these research tasks. Taken together, they form the basis for this thesis, as detailed below.

- **Describing the need of a pilot advisory system for General Aviation aircraft.** This first task addresses how smart cockpit computing can be used to the benefit of the GA community. It will identify specific problem areas within GA, which justify the cost and complexity of integrating an intelligent system such as ASTRA into the cockpit.

- **Development of a general functional specification for ASTRA.** This second, and perhaps most important task, is to specify the total system performance of ASTRA, in terms of its modules and their interaction with each other and the pilot. The functional specifications include the FMI, the PA, and system displays (HUD and HDD). This defines the functions ASTRA should be capable of performing, independent of aircraft type or flight locale. The original GAPATS functionality will be significantly augmented as a result of this task. For example, the specification will permit integration of other state-of-the-art technology (such as a moving-map display) into the ASTRA system architecture. Furthermore, *nominal* and *non-nominal* flight conditions will be defined, so that appropriate alarms can be generated for display. Finally, the concept of *automatic mode switching* will be introduced, whereby the system displays change their symbology configuration as a function of inferred flight mode. This concept, novel to GA aircraft, has the potential to significantly reduce pilot workload and increase situational awareness.

- **Increasing the robustness of the Flight Mode Interpreter.** The FMI is the "heart" of ASTRA. Without reliable information from the FMI, the pilot advisory system would be of little use. Simply stated, a robust FMI must correctly infer the current flight mode throughout the entire operational flight envelope of the aircraft, independent of aircraft configuration or pilot technique. Further, it must infer the flight mode in a timely manner, without "nervousness" or error. Increasing FMI robustness will entail expanding the fuzzy rule-base defined by Harral, so that the state-space is more completely defined. The rule-base will be further modified to exclude aircraft configuration parameters, such as landing gear position. Finally, the rule-base will be extended to include *distance* information, which will be provided by the Navigation Module.

- **Specification for a Navigation Module.** This subsystem will make use of aircraft position data and will require the integration of the Global Position System (GPS) into ASTRA. The Navigation Module will perform several critical functions for ASTRA. As noted in the previous bullet, it will provide *distance* information to the FMI, thereby further increasing the robustness of flight mode inference. Second, it will give the pilot the additional ability of using ASTRA for real-time flight-planning and navigation. For example, the module should calculate such variables as *ground speed*, *aircraft heading/track*, *wind speed/direction*, and *arrival time*. Because GAPATS did not include any provisions for a Navigation Module, its development will play a significant role in

ASTRA. Its specification will include defining data input/output requirements needed to provide ASTRA with these enhancements and capabilities.

- **Development of pilot interface functional requirements.** This specification is critical to ensuring ASTRA meets its stated goals of improving safety and the capability for training in GA aircraft. Furthermore, the specification will address how the pilot interacts with the ASTRA flight displays. Considering the HUD to be the primary flight display for pilotage and navigation, how the pilot interprets and reacts to different HUD *symbology configurations* (during nominal flight conditions) and *alarms* (during non-nominal flight conditions) must be carefully considered. Similar arguments can be made for how the pilot interfaces with the HDD, when used for navigation and mission planning. Finally, the specification will address how the pilot communicates with ASTRA through the HDD. Specifically, the pilot must have a means to enter (and edit!) mission planning data prior to or during the conduct of a flight. For example, he might wish to change a display mode on the HUD, select the moving-map display on the HDD, or change his flight plan in response to a clearance from ATC.

- **Incremental system performance evaluation.** Evaluation of ASTRA will be necessary to: (1) verify the functionality of individual subsystems, especially as existing modules are modified or new modules are added to ASTRA; (2) validate and optimize pilot interface issues, such as HUD/HDD symbology display sets, automatic mode switching, and data entry; and (3) validate ASTRA as a candidate for commercialization. As previously noted, this performance evaluation can be readily accomplished in the EFS, since the EFS is part of the system software development environment. By conducting a thorough performance evaluation in the EFS, it is further hoped that ASTRA will be mature and robust enough to install in the Commander-700 with only minor modification. In other words, ASTRA and each of its subsystems may be thoroughly evaluated in the EFS before it is ever flown in an actual aircraft.

The set of research tasks detailed above, which require an expert knowledge of aeronautical flight operations and aircraft performance, is necessary in the creation and evaluation of a mature, integrated system design. It is hoped that by providing solutions to each of these research tasks, this thesis will provide significant contributions to the research area of "smart cockpit computing," will greatly benefit the development of the ASTRA system, and will assist in validating ASTRA as a commercially viable product.

EXPERT SYSTEMS AND THE PILOT ADVISOR

"An expert is a man who has made all the mistakes which can be made in a very narrow field."

—Niels Bohr

The Need for a General Aviation Pilot Advisor

Before defining a general system specification for the Automated Safety and Training Avionics (ASTRA) system, it might first be appropriate to address several important questions. Namely, does the general aviation (GA) community have a real need for a system such as ASTRA? Put another way, what added value does ASTRA bring to general aviation, that justifies the additional cost and complexity? Finally, how can ASTRA specifically benefit general aviation?

Data in recent years have demonstrated a need for cockpit automation in the general aviation community. The reasons are simple. First, the density and variety of all air traffic, which includes GA aircraft (generally light, fixed-wing airplanes), commercial interests (airliners, airfreight, and other large aircraft), and the military community (high performance jets, fixed-wing, rotary wing, and tilt-rotor), is increasing throughout the United States. As one might expect, with an increase in air traffic there is a corresponding increase in the number of accidents. In fact, the National Transportation Safety Board (NTSB) reported last year that the accident rate (per 100,000 flying hours) for GA aircraft rose to its highest level since 1984 [1]. An examination of the NTSB data (summarized in Appendix B) also reveals an alarming trend—the accident rate for GA aircraft has risen in each of the previous six years. Consequently, cockpit automation could be a valuable means for reducing accidents within general aviation. This reduction in accidents implies an increase in safety for not only the GA community, but for commercial and military aviation interests as well.

Cockpit automation in GA aircraft could provide additional benefits. For example, it is a costly and time-consuming process to obtain a pilot's license (issued by the Federal Aviation Administration, or FAA) which allows flight under Instrument Flight Rules (IFR). Furthermore, once a person obtains an FAA license, maintaining pilot proficiency, especially when flying IFR, may be extremely difficult. In fact, during the same six-year period in which the GA accident rate rose (1990-1995), the total number of hours flown actually decreased in each of those years. Such data suggest that pilot proficiency is decreasing, despite flying aircraft with increasingly sophisticated equipment.

This issue of maintaining pilot proficiency may be due to a number of factors, the most important being that the pilot must do much more than simply manipulate the controls to fly the aircraft well. For example, he must be intimately familiar with all current Federal Aviation Regulations, or FAR's (which are not only regulatory in nature, but procedural as well). Further, the pilot must know, and be able to instantly recall, his aircraft's normal operating procedures, limitations, and emergency procedures. For example, which immediate action should the pilot take if the plane's engine were to quit *now*? Finally, the aviator must understand the type of weather he plans to fly through, the terrain he intends to fly over, and the implications of each. For instance, does he plan to fly over mountainous terrain when icing conditions or turbulence might be present? In short, the pilot must mentally integrate, process, and manage all the information necessary to safely fly while meeting the requirements of ATC [20]. Consequently, cockpit automation could serve well as an information manager in training and assisting the pilot. Pilot proficiency could rise significantly.

It should also be noted that the pilot advisor, when used as a training device and information manager, would likely provide as much benefit for any pilot, whether he be a newly licensed student or a seasoned "combat ace." In fact, De Silva [5] notes that, "an expert system may be equally useful to both an expert and a layperson. For example, it is difficult for one expert to possess a complete knowledge in all aspects of a problem, and the solutions can be quite complex. The expert may turn to a good expert system which will provide solutions that the expert could evaluate further…" This is particularly true in the dynamic environment of aviation, where *any* pilot will routinely encounter vastly differing flight situations—such as weather, air traffic control, and other aircraft traffic—when flying between the same two airfields any given number of times.

The implications of using a cockpit manager to assist the pilot in GA aircraft appear subtle, but are important nonetheless. By assisting the pilot in all the functions (routine or otherwise) necessary to execute a flight, a number of important results can be anticipated. First is the reduction in pilot workload, because the pilot advisor will be able to continuously track and monitor all data. This will allow the pilot to concentrate on those items most critical to the phase of flight he is in, while the PA continues to monitor all items. The second benefit is an increase in pilot proficiency, because the pilot will have the ability to use the PA for training. In other words, the PA can reinforce, in real-time, those actions the pilot should be concerned with, commensurate with the flight mode he is in. For example, the PA might advise the pilot that he needs to reduce his airspeed to enter a holding pattern, to change a navigation radio, or to fly a recommended heading to enter a holding pattern. Third, the pilot will have greater situational awareness while flying, as the PA can give immediate feedback regarding any non-nominal aircraft state (for example, did the pilot forget to retract the landing gear after take off?). Likewise, situational awareness will improve because the pilot will be encumbered with fewer tasks to directly manage. Taken together—the reduced pilot workload, the increased pilot proficiency, and the increased situational awareness—the pilot advisor will bring about a significant increase in safety for the GA pilot.

In short, a pilot advisor would bring many benefits to the GA community. It would assist in the training of new pilots. It would increase the proficiency of current pilots. It would make the airways safer for all pilots. Consequently, a revival in the GA aircraft market could be expected. These benefits describe the overall objectives in designing a commercially viable expert system such as ASTRA.

A General ASTRA Functional Specification

As might be concluded from the previous paragraphs, ASTRA should have the capability to monitor and advise the pilot in all phases of flight. This is true under nominal flight conditions under visual or instrument flight rules (VFR and IFR, respectively), as well as in emergency situations. Consequently, ASTRA should be able to perform the following functions:

- Correctly identify the current flight mode. Display advice (in symbolic or text format) appropriate for this flight mode.
- Display alarms to the pilot when abnormal conditions exist.
- When operating on a cross-country flight plan, provide comprehensive navigation information, to include course, altitude, and aircraft configuration guidance.
- Provide the pilot with real-time mission planning (such as route selection, fuel requirements, or weight and balance information).
- Provide training advice for pilots wishing to increase their proficiency.
- Display check-list information, to include procedures for normal operations and emergency situations.
- Display aircraft operating limitations information.
- Provide advice on alternate course of action. Such advice might be automatically displayed under emergency conditions, or when requested by the pilot.

- Include provisions for the insertion of developing technologies, such as moving-map displays, aircraft collision avoidance systems, and digital data-link communications.

Updated ASTRA System Architecture

The general design goals described in the last section, along with the addition of several new modules to ASTRA noted in the previous chapter, will naturally bring an increase in the complexity of the system architecture. Figure 3 reflects these changes, followed by a more detailed discussion of the ASTRA subsystems.

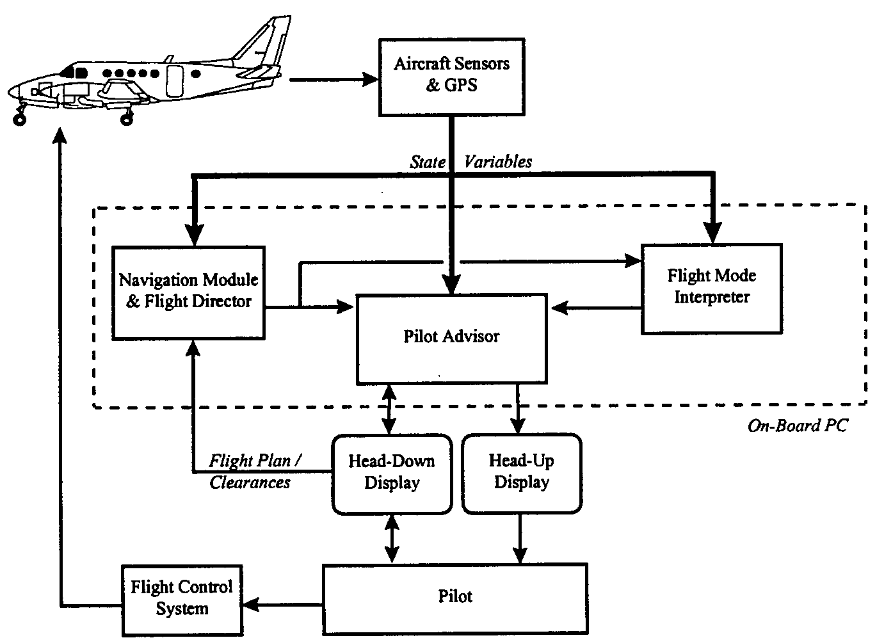


Figure 3. Augmented ASTRA System Architecture

Aircraft Sensors. As their name implies, the aircraft sensors provide ASTRA and its subsystems with state-variable data. These data can be used directly as raw data, or can be used to derive additional variables which cannot be directly measured. These state variables include altitude, heading, indicated airspeed, pitch and roll attitude, turn rate. A comprehensive list of aircraft instrumentation/sensors is included in Appendix C. One significant difference between the GAPATS and ASTRA sensor suites is the inclusion of the Global Positioning System (GPS) into the latter. By integrating GPS into ASTRA, several additional parameters are made available: aircraft position (in latitude and longitude) and time (given with respect to Greenwich Mean Time). Consequently, GPS data can be used by both the FMI and the Navigation Module, as detailed below.

The Flight Mode Interpreter (FMI). As previously detailed, the purpose of the FMI is to evaluate the current *flight mode* of the aircraft. It does this by evaluating (using *fuzzy logic classification*) aircraft state variables such as altitude, airspeed, and power setting. It is important to note that the FMI must make this calculation <u>independent</u> of aircraft configuration and of pilot input. In other words, aircraft configuration (such as landing gear position or the flap setting) should not influence how the FMI infers the aircraft flight mode. Consequently, should the pilot make a configuration error (such as forgetting to lower the gear in preparation for landing), the error will not result in an improper flight mode classification . In other words, the FMI, as the "heart" of ASTRA, must correctly infer the flight mode from what the aircraft "could be" doing, based on the fuzzy-classification of the aircraft state variables.

The Pilot Advisor (PA). If the FMI is the "heart" of ASTRA, then the PA is its "brains." The PA, another rule-based expert system, takes the flight mode inference from the FMI, along with raw sensor data, to generate a series of commands, advice, and "alarms" to the pilot. This information is displayed to the pilot in one (or both) of two places: the Head-Up Display (HUD) or the Head-Down Display (HDD). In other words, by comparing the raw sensor data with FMI output, the PA can interpret what the pilot "should be" doing (recall that the FMI made its inference on what the aircraft "could be" doing). In this manner, the PA can be designed to perform many of the functions that an "expert" (instructor pilot) would perform while performing duties as the copilot.

Unlike the FMI, the PA uses a crisp rule-base, which is being developed using CLIPS, an expert system language tool developed by NASA. CLIPS was an excellent candidate for use in ASTRA for a number of reasons. These reasons include the flexibility CLIPS provides in modeling knowledge and its ability to be fully integrated with other languages, such as C++ (which was used in the development of GAPATS). Giarratano [8] provides a more detailed description of how CLIPS can be used in the development of expert systems.

As an expert system, the PA must have access to a number of data bases. For example, the PA would require access to aircraft specific information, including operating procedures (such as check-list data), operating limitations (such as maneuvering and airframe/component limitations), and mission planning data (such as weight and balance information and fuel/cargo capacities). This information, in read-only format, would be located in the *aircraft* *data base.* A second data base, containing information relating to airfields, navigation aids (NAVAIDS), controlled and special-use airspace, airways, and instrument approaches would also be necessary. This *navigation data base* (again, in read-only format) would be essential to provide for mission planning and navigation.

A more detailed description of the FMI and the development of its fuzzy rule-base are developed in the next chapter of this thesis.

ASTRA Cockpit Displays. As noted earlier, ASTRA consists of two different displays. The first of these, the HUD, is considered the primary flight display. This is because the HUD is designed to provide the pilot, while looking outside through the display, with all the essential flight information necessary to fly the aircraft, even in Instrument Meteorological Conditions (IMC). Consequently, the pilot is able to focus on flying the aircraft while simultaneously searching for other traffic, scanning for airfields in poor weather, or transitioning from instrument to visual flight. In this respect, the HUD is an essential component for increasing the pilot's situational awareness and safety.

The second ASTRA cockpit display is the HDD, which is designed as a Multi-Function Display (MFD). Such displays are common in many modern military and commercial aircraft with "glass cockpits" designs. These MFD's allow the pilot to readily change the display configuration, permitting a wide variety of information to be presented. For example, the pilot might choose to view a moving-map display (which shows the current position of the aircraft superimposed over a digital map image); aircraft checklist information; or other flight planning and navigation information. The HDD also provides the pilot with one additional essential capability—data input. Hence, the HDD also acts as an interface module for the pilot, where he can, for example, enter flight data (such as an ATC clearance), change display settings (on either the HUD or HDD), or acknowledge alarms generated by the PA.

Navigation Module. As noted in the previous chapter, the Navigation Module will give ASTRA significant capabilities over its GAPATS predecessor. First, the Navigation Module can increase the robustness of the FMI by providing it with *distance* information. In other words, the Navigation Module will be providing additional state-variable information to the FMI, which will serve to increase the certainty of flight mode inference.

Second, the Navigation Module will allow the pilot to use ASTRA for real-time mission planning. When used in this manner, the Navigation Module can further provide the pilot with comprehensive navigation information. For example, by integrating a flight director into the HUD symbology set, the Navigation Module can generate course and altitude guidance. These concepts are presented in greater detail in subsequent chapters.

As might be anticipated, the Navigation Module will require information from a variety of sources. The first, noted above, is the *position* and *time* data made available from GPS. The second, also previously alluded to, is data from the read-only *navigation data base*. In fact, extensive navigation data bases (already certified for in-flight use by the FAA) are commercially available today. The final piece of information which the Navigation Module can use is flight plan data, as entered by the pilot. This information is used when the pilot wishes to fly from one airport to another. However, because the pilot may wish to simply fly "traffic patterns" at a single airfield, rather than flying "cross country" to another, this flight plan information should be considered optional data. In other words, the Navigation Module should function correctly independent of flight locale.

Nominal and Non-Nominal Flight Conditions

As noted in the first chapter, ASTRA can be most useful only if it considers both *nominal flight conditions* and *non-nominal flight conditions*. For the purpose of designing a pilot advisory system such as ASTRA, *nominal flight conditions* can be considered as those in which no piloting errors have been made and for which no *abnormal aircraft conditions* exist. (An example of an abnormal aircraft conditions might include excessively low engine oil pressure.) In other words, during nominal flight conditions, the flight is progressing as an "expert" pilot might expect.

Non-nominal flight conditions, on the other hand, occur whenever piloting mistake(s) have occurred or when an abnormal aircraft condition warrants alerting the pilot. In other words, the PA will generate an *alarm* in response to any defined non-nominal flight condition. Implicit with non-nominal flight condition is that the pilot must make some type of positive action—either to correct a piloting error; to react to an impending emergency situation; or to notify ASTRA that the pilot is aware of, but chooses to ignore, the alarm. These pilot actions each have varying degrees of urgency, which is a function of the non-nominal flight condition which causes the alarm. The degree of urgency of each alarm will, in turn, drive how the alarm is displayed to the pilot.

Partitioning the aircraft flight mode into these two states facilitates how (and where) information is displayed to the pilot. In other words, how ASTRA passes information on to the pilot—via the HUD and the HDD—is a function of whether the aircraft is in a nominal or non-nominal flight condition. Further, it is important to note that, even under nominal flight conditions, meaningful information must still be generated for display to the pilot. Such information might include navigation and basic pilotage data (altitude, attitude, airspeed, etc.) on the HUD, or checklist and flight planning information on the HDD. Consequently, information provided to the pilot during nominal flight conditions should enhance his *situational awareness* and reinforce positive learning habits, while reminding the pilot to perform actions he might otherwise have forgotten.

Under non-nominal flight conditions, on the other hand, information must be displayed in such a way that it catches the pilot's attention as soon as possible. Further, the information must be formatted in an unambiguous manner—so the pilot immediately understands not only what problem exists, but also what corrective action is appropriate. As noted above, such information generated during non-nominal flight conditions is categorized as an *alarm*. Describing these alarms in terms of their seriousness, as well as defining how and where to display them on the HUD/HDD, will be described in greater detail in later chapters. At this point, it is sufficient to state that alarms are displayed *by exception*. That is, if no alarms are displayed, then the pilot may assume that all systems are functioning nominally and that the flight is progressing satisfactorily.

Automatic Mode Switching

One of the important implications of displaying flight information under *nominal* conditions is that display information should change as a function of *flight mode*. In other words, some information that is appropriate for display under one flight mode, such as *take-off*, may be inappropriate for display in another, such as *landing* (and vice versa). In the latter case, such information might only serve to "clutter" the display, providing the pilot with information he does not need. This can, in turn, make interpretation of the remaining essential flight information more difficult. Consequently, as an aircraft's flight modes change, so should the format of its flight displays.

Providing a unique display format for each flight mode supports two important tenets of the ASTRA design. First, such flight displays can increase the pilot's situational awareness, because they adapt to the appropriate situation in which the pilot is flying. Consequently, these display formats will help reduce the number of piloting errors made. Further, when an error is made, the display can assist the pilot in providing an appropriate corrective action, minimizing the severity of the piloting error or abnormal condition.

As will be described later in greater detail, designing the format and layout of a given flight display is a complex task. The process is further complicated by the conclusion reached in the previous paragraphs—namely, that each flight mode should have a unique display format. Such capability is commonly found in tactical military aircraft, but has never been seen in GA aircraft. For example, an attack helicopter may have one display configuration for each of its different weapons systems, in addition to the different display configurations for the aircraft's flight modes (such as hovering flight, terrain flight, or cruise flight) [17]. GA aircraft, on the other hand, have typically seen fixed display formatting, which stays fixed regardless of the aircraft flight mode [3]. It is also important to note that even with the tactical aircraft, the pilot must *manually* select the display mode he desires, resulting in an increase in pilot workload (which could, in turn, result in *additional* pilot errors). It is entirely possible for our example attack helicopter to be flying in a *cruise* mode, when in fact the pilot has selected the *hover* mode for display.)

Consequently, the ASTRA found itself facing an interesting dilemma: that multiple display configurations could increase situational awareness and reduce piloting errors, but only at the cost of increased pilot workload, because of manual display mode switching (which might in turn reduce situational awareness and increase errors). This conclusion contradicts one of the basic tenets of the ASTRA—that the system should not increase pilot workload. However, ASTRA contains one key subsystem not found in any other pilot advisory system—the FMI. Because the FMI is designed to reliably interpret the current aircraft flight mode, then its output could also be used to *automatically* change the display format of the HUD and the HDD alike. This concept is known as *automatic mode switching* (AMS).

As might be expected, the implications of AMS are important and far-reaching. By carefully implementing its use, not only can pilot workload be kept to a minimum, but the potential for a significant reduction in workload might also be anticipated. Such potential can be realized if the expert system rule-base carefully anticipates the situations the pilot might encounter, as well as the corresponding information needed for each flight mode. Of course, the use of AMS must allow for the pilot to override a particular display mode chosen by ASTRA—the pilot must always have the final say in how the aircraft displays are configured for continued safe flight.

FLIGHT MODE INTERPRETATION USING FUZZY LOGIC

"Logic is the art of going wrong with confidence."

—Joseph Wood Krutch

Background

As briefly described in the Introduction, the original GAPATS system architecture demonstrated the power and utility of using *fuzzy logic* as an integral element of a "smart cockpit computing system." Specifically, fuzzy logic algorithms were developed by Harral [10] in the Flight Mode Interpreter (FMI) to provide continuous estimates of the aircraft flight mode, based solely on aircraft state information.

There are several reasons motivating the use of fuzzy logic in the application of "intelligent" cockpit systems. First, fuzzy logic can provide a useful means of providing approximate solutions to complex problems (such as flight mode interpretation). Second, the use of fuzzy logic provides an excellent linkage mechanism between "qualitative" descriptors and the numeric domain, because the use of such qualitative descriptors closely mirrors how humans think and reason. Consequently, humans can more easily understand and interpret these complex problems. Finally, fuzzy logic can accomplish these tasks with in a cost-effective manner, without using sophisticated computing systems or requiring extensive computer memory (a pitfall of most other artificial intelligence applications) [16].

The philosophy of fuzzy logic embodies two principles: the representation of truth in degrees (called *certainty values*) and the use of these degrees of truth in decision making. The certainty values are described in terms of a *membership function*. Given a universe of discourse, X, this membership function gives the degree of membership, μ_A, for any element x within a set A. That is, $\mu_A(x)$ measures the certainty value of $x \in A$, satisfying the relationship:

$$0 \leq \mu_A(x) \leq 1 \quad : \forall x \in X \tag{1}$$

Figure 4 depicts an example of two membership functions for the parameter *aircraft altitude*. By using this figure, we can make a *fuzzy inference* of whether the aircraft flight mode is *landing* or *final approach*:

$$\mu_L < \mu_F : \text{ infer "Mode = Final Approach"} \tag{2}$$

$$\mu_L > \mu_F : \text{ infer "Mode = Landing"} \tag{3}$$

At the point where the membership functions intersect (i.e., $\mu_L(h) = \mu_F(h)$, $h = 250$ ft), it is equally likely that the aircraft flight mode is *landing* or *final approach*; no inference can be made. This point is formally called the *Hard Decision Threshold.*

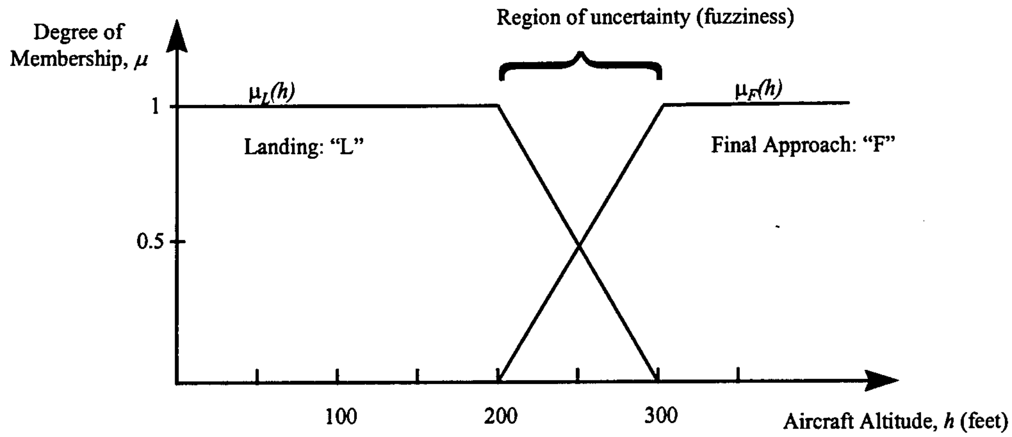


Figure 4. Overlapping Membership Functions

Several items in the figure are worth noting. First is how the membership functions are described in linguistic, qualitative terms, as defined by an expert. Collectively, these membership functions partition the universe of discourse into subspaces that define possible aircraft flight modes, such as *landing* or *final approach*. While only two flight modes are indicated in Figure 4, partitioning the universe of discourse (in this example, aircraft altitude) would naturally require defining membership functions for all possible flight modes.

Also of interest in the figure is the so called *region of uncertainty*, or fuzziness. In this region, the aircraft flight mode could be either *landing* or *final approach* (although with unequal certainty). Note that for this example, the membership functions are linear within this region of uncertainty. In fact, the membership functions could take on any form, such as a Gaussian or exponential function, as long as (1) is satisfied. However, using linear functions can greatly reduce the complexity of finding a certainty value, with little loss in the accuracy of the fuzzy inference.

Similarly, we could build additional membership functions describing the aircraft flight mode by using other aircraft state variables, such as *airspeed*, *rate of climb*, or *engine power level*. In this manner, the aircraft flight mode can be more completely defined, in terms of all these aircraft state variables, taken together. The membership functions collected for each mode form sets of rules, called the *fuzzy rule base*, from which a more complete inference of the aircraft flight mode can be made.

To summarize then, by applying fuzzy logic algorithms to aircraft state parameters (such as altitude), the FMI is able to reach a decision about the flight mode of the aircraft. This information can be in turn used to draw further conclusions about how well the pilot is flying the aircraft. Consequently, correctly inferring the aircraft flight mode is essential in providing sound "advice" to the pilot.

The GAPATS Flight Mode Interpreter

Harral's model for the original FMI included the definition of six flight modes: *taxi*, *takeoff*, *climbout*, *cruise*, *initial approach*, *final approach*, and *landing*. He further formed membership functions for eight aircraft state parameters: *thrust*, *alpha* (angle of attack), *roll*, *landing gear position* (retracted or extended), *flap position*, *airspeed*, *altitude*, and *rate of climb*. These membership functions are depicted in Figure 5. Note the use of linear functions to simplify the computational requirements of the FMI.

	Taxi	Takeoff	Climbout	Cruise	Initial Approach	Final Approach	Landing
thrust (%)	N/A	98 / 95	98 / 95	45 75 / 40 100	45 / 65	45 / 55	35 / 45
alpha (degrees)	0 / -3 +3	8 / 10	4 6 / 0 8	+1 +4 / -2 +8	2 8 / 0 10	4 6 / 1 9	4 8 / 0 10
roll (degrees)	0 / -10 +10	0 / -10 +10	N/A	-30 +30 / -40 +40	N/A	-10 +10 / -15 +15	-10 +10 / -15 +15
landing gear	1	1	0	0	N/A	1	1
flaps (degrees)	0 / -3 +3	12 / 9 15	0 / -3 +3	0 / -3 +3	12 / 15	30 / 27 33	30 / 27 33
airspeed (knots)	10 / 50	50 105 / 10 125	110 120 / 105 135	135 170 / 120 180	125 140 / 115 150	90 115 / 75 125	50 105 / 10 115
altitude (feet)	10 / 15	50 / 70	70 1500 / 50 3000	3000 / 1500	1200 1500 / 800 3000	200 800 / 100 1200	100 / 200
rate of climb (fpm)	0 / -5 +5	100 / 1000	1000 / 100	-500 +500 / -1500 +1500	-1500 -200 / -2000 0	-1000 -100 / -2000 0	0 5 / -300 +300

Figure 5. The Original GAPATS Membership Function Rule Base

The inference scheme used by the FMI entailed finding the product of the eight certainty values for each flight mode, then selecting the maximum certainty as the "correct" flight mode. For example, the certainty for *taxi* would be calculated as:

$$\mu_T = \prod_{i=1}^{8} \mu_{Ti} \tag{4}$$

where μ_{Ti} corresponds to each of the associated aircraft state parameters (thrust, alpha, roll, etc.) for *taxi*. Once the certainty values were calculated for the remaining five flight modes, that flight mode associated with the maximum certainty value was selected as the inferred flight mode:

$$Mode = \max[\mu_T, \mu_{T/O}, \ldots, \mu_L] \tag{5}$$

While this implementation of the FMI verified the utility of using fuzzy logic in flight mode inference, it possessed several weaknesses. First, the FMI was sensitive to variations in pilot technique, resulting in "nervous" (constantly changing) decisions. An analysis of the rule base indicated that state-space for each parameter was not adequately partitioned. Consequently, the membership functions for each flight parameter required more careful redefinition.

The analysis of the FMI rule base also revealed the inclusion of aircraft configuration parameters (for landing gear and flap position) within the rule base. While it might seem logical to include aircraft configuration in the rule base, it is, in fact, undesirable. The reason for this lies in the fact that aircraft configuration is a function of pilot input—the pilot must *manually* change the gear or flap setting. Consequently, any errors in aircraft configuration will propagate into errors in flight mode interpretation. In short, to provide meaningful advice and alarms for the pilot, the FMI must determine the flight mode *independent* of aircraft configuration.

Finally, an examination of FMI input data suggested that several of the state variables were not useful in flight mode inference, resulting in inaccurate system decisions and spurious alarms. For example, angle of attack data varied so greatly during every phase of flight that it was impossible to partition this state-space for each flight mode. Hence, refining the FMI rule base would require that only meaningful flight parameters be included in the rule base.

The ASTRA Flight Mode Interpreter

Based on the analysis detailed in the previous section, the FMI rule base was extensively refined. Figure 6 reflects these changes, which are further described in the following paragraphs.

Making the FMI less sensitive to pilot input and pilot technique required "broadening" the membership functions for each of the flight parameters. In fact, a close inspection of the prior figure shows that many membership functions have relatively restrictive tolerances. That is, the membership functions have boundaries which are: (1) very nearly "crisp," or (2) which are closely associated with the value where the parameter "should be," rather than "could be." The use of such restrictive membership functions greatly reduces the ability of the FMI to correctly infer the flight mode using the fuzzy logic algorithms. Consequently, by using such restrictive membership functions, one can expect to lose much of the power afforded by the use of fuzzy logic.

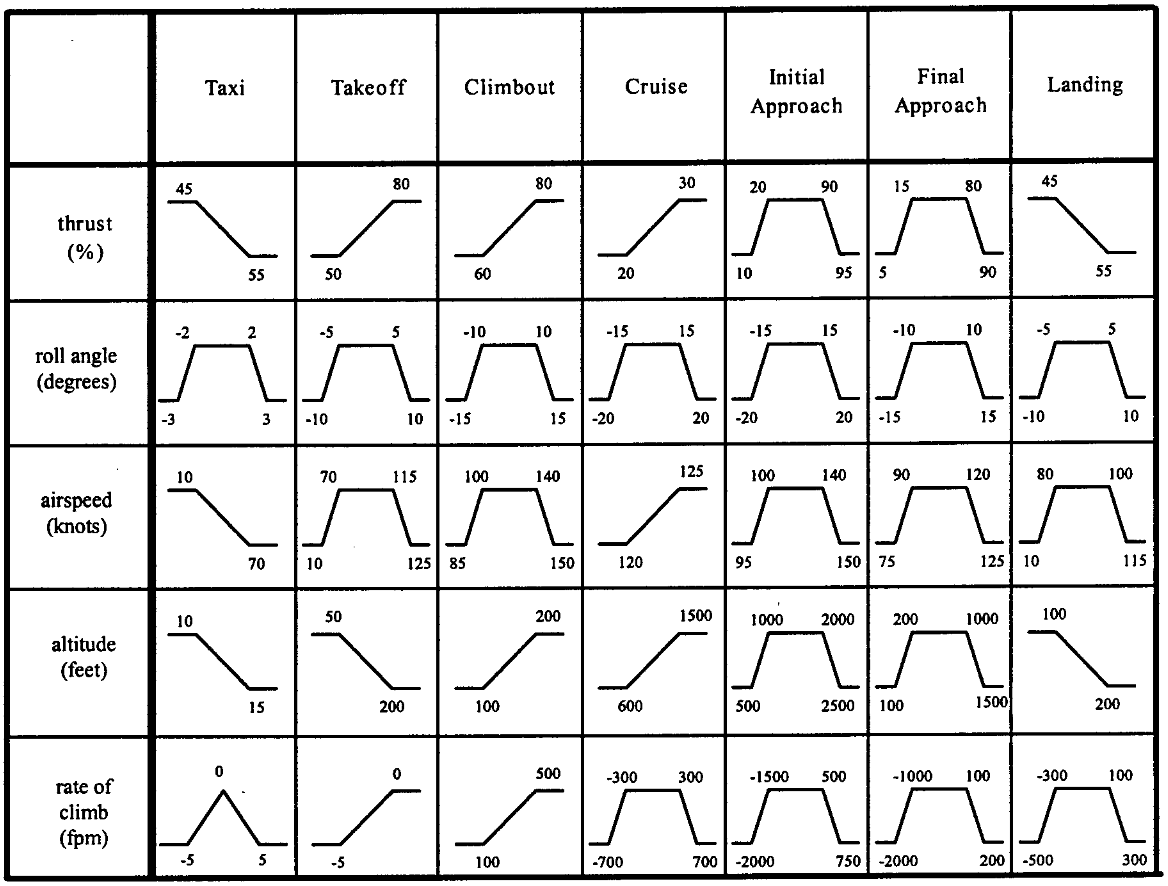


Figure 6. The Revised ASTRA Rule Base

Several examples, using the parameter *thrust*, will more clearly illustrate this point. Note in Figure 5 that the initial GAPATS model did not consider *thrust* for the *taxi* mode. This seems unusual, given that some thrust is required to taxi an aircraft, albeit less than that required for *takeoff*, *climbout*, or *cruise*. Likewise, when in the *cruise* mode, the original model gives little weight for the case in which the pilot is flying the aircraft at maximum thrust—a situation that is not at all uncommon. Finally, the GAPATS model did not consider thrust values beyond 45-55% during *final approach*. In fact, many pilots vary the thrust greatly, especially when cross-winds or gusty conditions exist, in an effort to maintain airspeed and glideslope for landing. Figure 6 shows how the membership functions in the ASTRA model have been revised to allow for the conditions just described.

Revising the rule base to exclude aircraft configuration parameters is a simple matter. An examination of Figure 6 shows that the two configuration parameters originally included—landing gear position and wing flap position—have been omitted entirely. In fact, one additional parameter has also been excluded from the rule base: *alpha* (angle of attack). This omission was done after a careful analysis of FMI data input, which revealed an extremely "noisy" signal with large, rapid fluctuations. In fact, no discernible information could be inferred about the flight mode as a function of *alpha*. While the use of this parameter has not been ruled out for use in future versions of the FMI rule base, at this point there is no justification for its inclusion, at least for this particular aircraft.

A Comparison of Flight Mode Inference Schemes

To provide a rapid means of modeling the FMI rule base and simulating its inference output, Kelly [13] developed the GAPATS/FMI Tool Box, a MATLAB® software application. This tool box permits the FMI developer to quickly build and modify fuzzy membership functions for inclusion in the FMI rule base. Once the rule base has been defined, the tool box will simulate, among other things, the inference output of the FMI, providing an excellent means of analyzing FMI performance.

After the developer defines the parameters defining the membership functions, the tool box provides plots for each, allowing the developer to graphically verify the partitioning of the state space. The developer can then simulate the FMI inference output by including a data file which includes the appropriate aircraft state variables. For example, as a pilot flies typical procedures (for *taxi*, *takeoff*, *climbout*, etc.) in the Engineering Flight Simulator (EFS), the developer can "record" each of the appropriate parameters into a data file. This data file can then be used any number of times to compare various configurations of the FMI. Likewise, various data files, containing the same procedures flown by different pilots, can be used with a single FMI configuration to evaluate its sensitivity to variations in pilot technique.

Sample plots generated by the GAPATS/FMI Tool Box are presented in Figure 7 and Figure 8. Note that the "correct" flight mode is plotted together with the FMI output in the figures. This "correct" flight mode, which must be manually entered by the pilot while flying the EFS (so that it can be stored as part of the flight data file), gives a baseline value by which the FMI output can be readily compared.

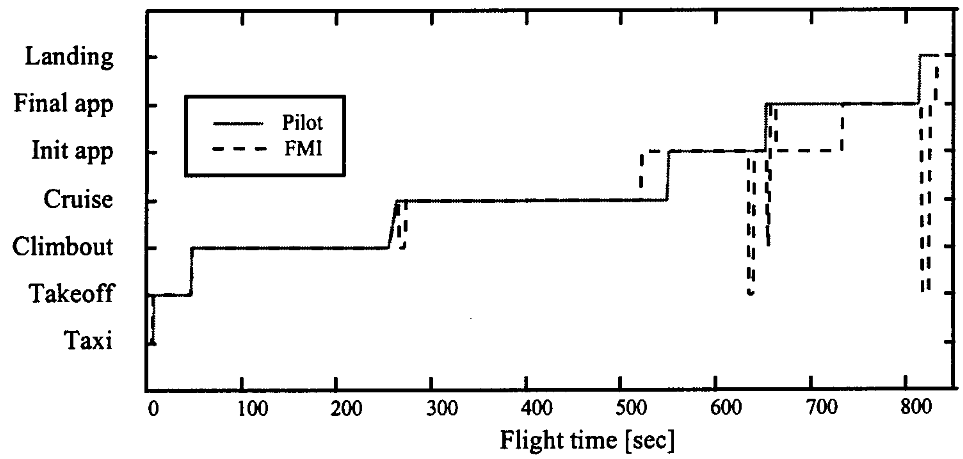


Figure 7. Original GAPATS Flight Mode Inference

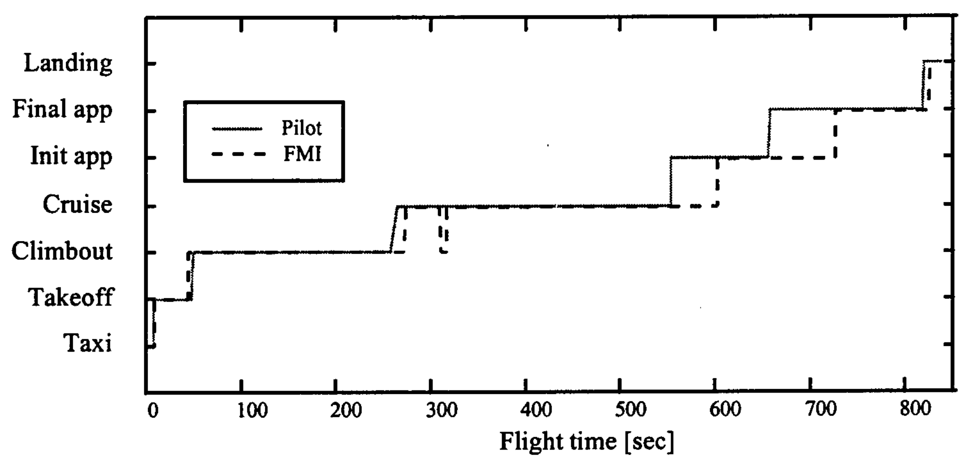


Figure 8. ASTRA Flight Mode Inference

Figure 7 illustrates the inferred FMI output, using Harral's original fuzzy rule base (that shown in Figure 5). Inspection of this figure shows that when the aircraft is in a static situation, the FMI inference generally agrees with pilot data. However, the mode inference tends to be very "nervous" in the dynamic situations—when the aircraft transitions from one mode to the next. This nervousness is especially apparent during perhaps the most critical phases of flight, as the aircraft transitions from *cruise* through the *approach* modes, and into *landing*.

Figure 8 shows the FMI output for the same flight data as before, but using the modified fuzzy rule base (corresponding to Figure 6). In this case, virtually all the nervousness has been removed, resulting in greatly improved FMI performance, even as the aircraft transitions from one flight mode to the next. The figure raises one point of concern, however. As the aircraft transitions from *cruise* to each of the *approach* modes, note that the FMI demonstrates an appreciable delay in recognizing the next mode. One possible explanation for this might be the similarity for many of membership functions in these flight modes. This analysis suggests that there might be additional aircraft parameters which could be included in the fuzzy rule base to further improve FMI performance, especially for the flight modes discussed in this paragraph.

A New Aircraft State Parameter: Distance

One important aircraft state parameter—especially when flying "cross country" from one airfield to another—is that of *distance*. As will be seen in the following chapter, the navigation process requires the pilot to, among other things, constantly calculate his distance from any number of points, such as the destination airfield. Consequently, the *distance* between these points and the aircraft (most of which the pilot must define in the flight plan) can be used to help define additional membership functions for inclusion in the fuzzy rule base. In fact, it is possible that *distance* can also be used in those situations in which the pilot is simply flying traffic patterns around a single airfield.

The use of a distance parameter naturally requires means of accurately measuring the aircraft position, then comparing it with the location of the various waypoints defined in the flight plan. The next chapter proposes the use of the Global Positioning System (GPS) for providing (among other things) aircraft location. The chapter also describes a Navigation Module, which will calculate the appropriate *distance* parameters detailed in the remainder of this chapter.

Defining which *distance* parameters to use is a function of the intended type of flight, for which there are two general cases: (1) the pilot is flying cross-country between two airfields, and (2) the pilot is flying "locally," practicing basic flight maneuvers (such a traffic patterns) at a single airfield. In the first case, an ASTRA "flight plan" must be entered and activated, whether the pilot is flying under visual flight rules (VFR) or instrument flight rules (IFR). In the second case, no ASTRA flight plan is required for flight mode inference, since the aircraft is operating about a single airfield. Specific examples of each situation are described in greater detail below.

Cross Country Flights (Flight Plan Entered and Activated). Using a flight plan to assist in flight mode interpretation is a natural consequence of how cross-country flights are planned and executed. For example, when executing an instrument approach, an aircraft is considered, by definition, to be on *initial approach* once it has passed the *initial approach fix* (IAF). Likewise, the aircraft is defined to be on *final approach* once it has passed the *final approach fix* (FAF) and is established on the inbound course. Consequently, these fixes can be used to partition the physical region surrounding the destination airfield (the *distance* state-space). By including all such waypoints (which define the route of flight and approach to be executed) in a detailed flight plan, the Navigation Module can then provide the FMI with the exact parameters needed to assist in flight mode inference, especially during the critical *approach* and *landing* phases. In other words, the Navigation Module reads the appropriate fields from the ASTRA flight plan to calculate all distance calculations.

One subtlety in partitioning the *distance* state-space is that, for a given instrument approach, the IAF and FAF may or may not be the same point in space. The next two figures illustrate this difference by partitioning the same instrument approach in two ways. (The instrument approach depicted in these examples—ILS Runway17L at the TSTC Airport near Waco, Texas—coincides with the approach which will be used during the flight test phase). Because it is possible for the IAF and FAF to be the same point in space, it is essential for the pilot to explicitly define these points in the Flight Plan.

Figure 9 depicts the situation when the IAF and FAF are not collocated. In other words, the pilot intends to fly directly over MACHO (the IAF) and proceed directly to LEROI (the FAF). Upon reaching LEROI, the pilot will execute a left turn onto the final approach course, which will take the aircraft towards the Missed Approach Point (MAP) and the destination airfield (Waco-TSTC).

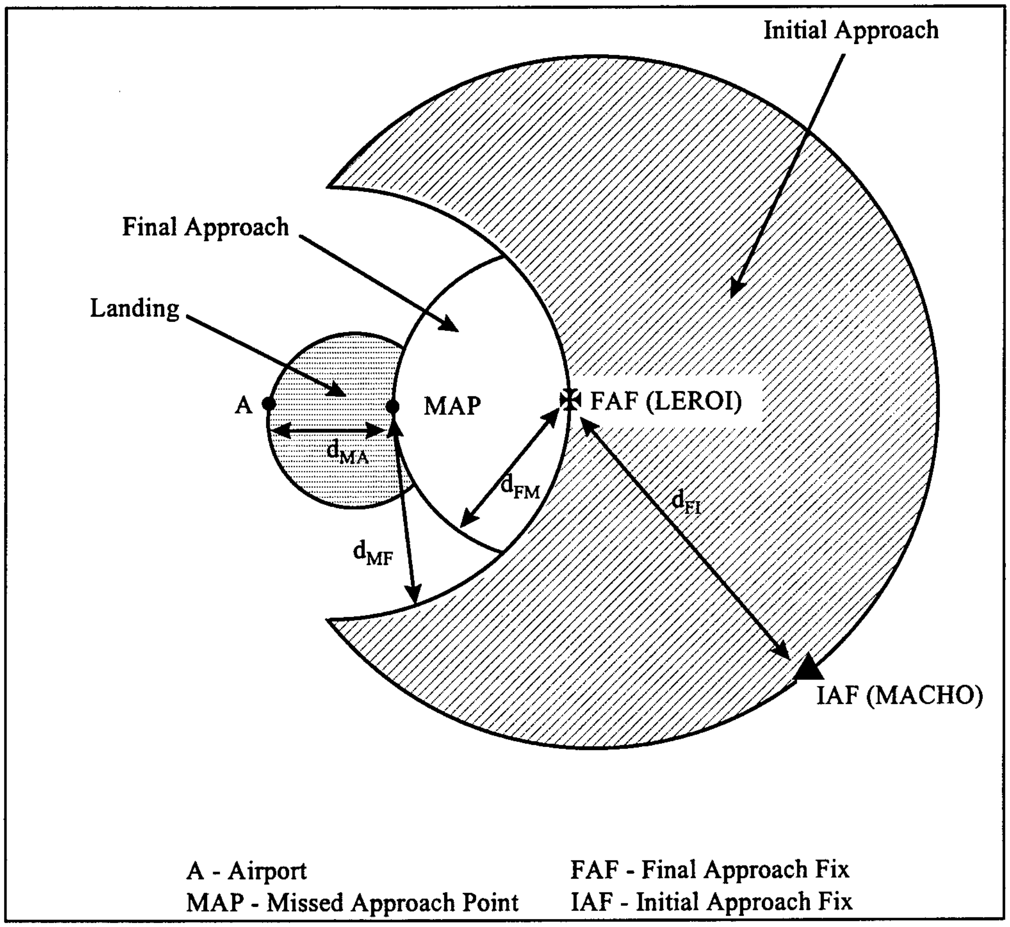


Figure 9. Space Partitioning for Differing IAF and FAF: ILS Approach to Runway 17-Left, Waco-TSTC Airport

The diagram shows that the region surrounding the FAF to be separated by a series of circular regions. For example, d_{FI} defines the circular region surrounding the FAF, with a radius equal to the distance from the FAF to the IAF. Similar regions circumscribing the FAF and MAP, defined by radii of d_{MF} , d_{FM}, and d_{MA}, are also defined. Note that each region is shown in the diagram as a *static* distance, because the distance is that between two fixed points over the ground.

By calculating the distance from the aircraft to each of these fixes (a *dynamic* distance), it is possible to infer the aircraft flight mode. In other words, the flight mode inference is made by comparing several dynamic distances with the static distances shown in the figure. The following equations demonstrate the required comparisons:

Infer *initial approach* if: $$(d_{ac-M} \geq d_{MF}) \cap (d_{ac-F} \leq d_{FI}) \qquad (6)$$

Infer *final approach* if: $$(d_{ac-M} \leq d_{MF}) \cap (d_{ac-F} \leq d_{FM}) \qquad (7)$$

Infer *landing* if: $$(d_{ac-F} \geq d_{FM}) \cap (d_{ac-M} \leq d_{MA}) \qquad (8)$$

These equations can be easily translated into membership functions for inclusion in the fuzzy rule base. For example, Figure 10 depicts the membership functions corresponding to Equation (6). The slope of the membership functions might be determined by using, for example, 10% of the distance where $\mu = 1$.

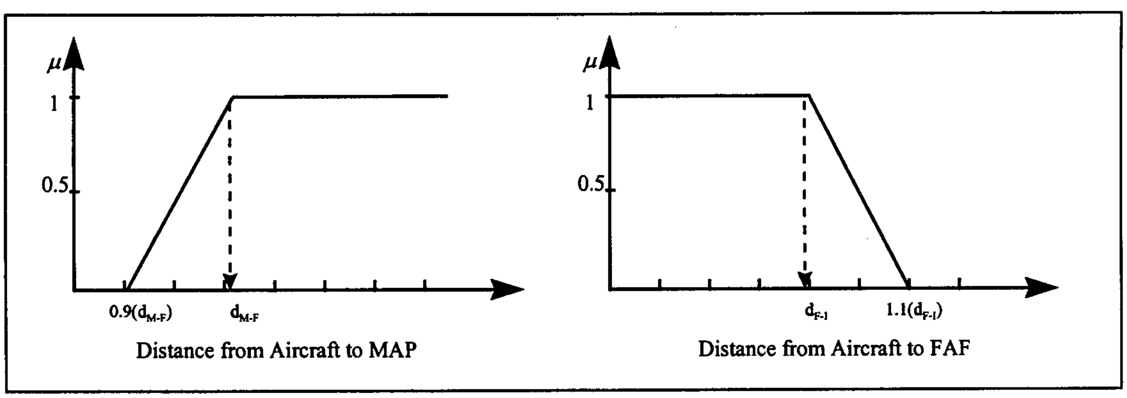


Figure 10. *Distance* Membership Functions Corresponding with *Initial Approach* (IAF and FAF not Identical)

Figure 11 depicts the space partitioning for the same instrument approach at Waco-TSTC, but with the IAF and FAF collocated at LEROI. When flying this particular approach, the pilot passes over LEROI (the IAF), proceeds on the appropriate outbound course, then makes a *procedure turn* (while remaining within 10 NM) to the inbound course and back to LEROI (which is now the FAF). The remainder of the approach (from the FAF to the airport) is identical to that previously described. Consequently, when the IAF and FAF are collocated, only one modification need be made to the fuzzy rule base. Specifically, Equation (6) becomes:

$$(d_{ac-M} \geq d_{MF}) \cap (d_{ac-F} \leq 10) \tag{9}$$

Note that only one value changes in this new equation. Further, the remaining two equations describing the distance relationships are unchanged. Figure 12 shows the result of transforming Equation (9) into its corresponding membership functions. Once again, the slope of the membership functions might be determined by using 10% of the distance value where $\mu = 1$.

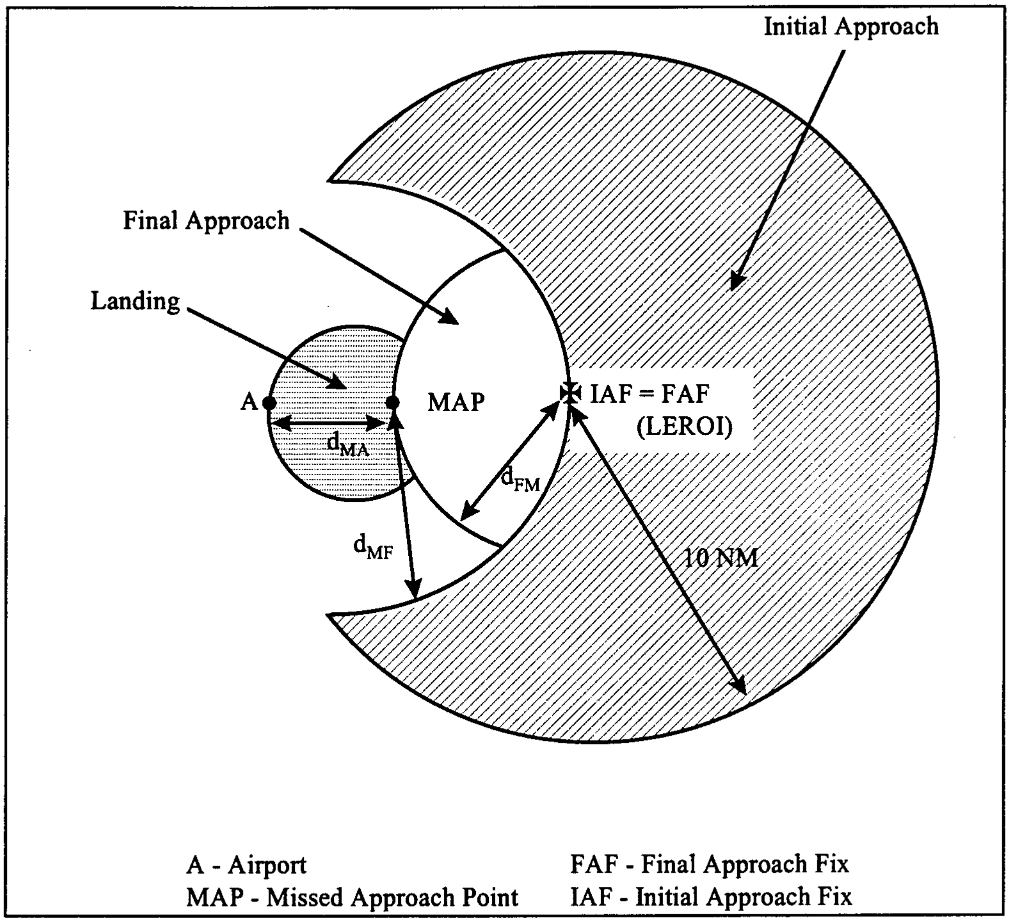


Figure 11. Space Partitioning for Identical IAF and FAF: ILS Approach to Runway 17-Left, Waco-TSTC Airport

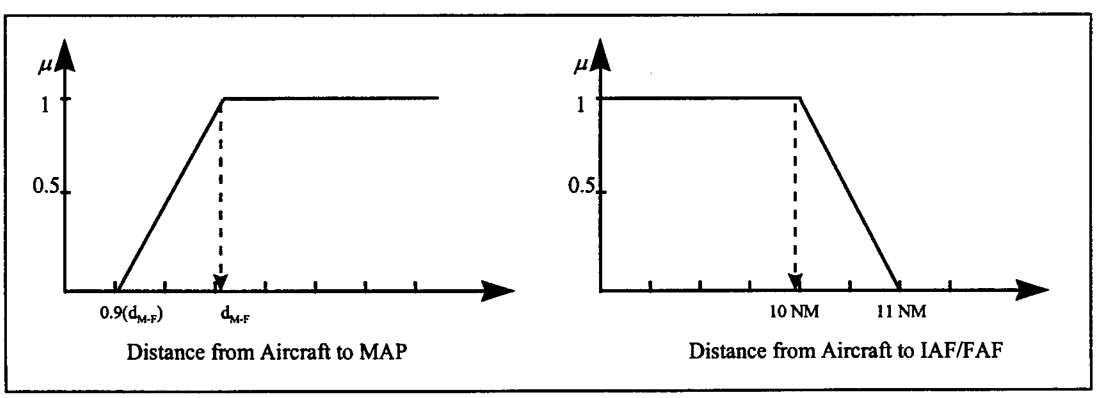


Figure 12. *Distance* Membership Functions Corresponding with *Initial Approach* (Identical IAF and FAF)

Local Flights (No Flight Plan Activated). For local flights, it is assumed that the pilot intends to practice basic VFR flight maneuvers, such as traffic patterns or touch-and-go landings, from a single airfield. Consequently, local flights do not require the pilot to enter and activate an ASTRA flight plan. (Note however, that should the pilot desire to practice any IFR flight maneuvers, to include holding or instrument approaches, the flight should be planned and executed as a "cross country" flight. Hence, it would be necessary for the pilot to complete an ASTRA flight plan, even if the flight were to take place about the departure airfield.) In short, the Navigation Module needs only two pieces of data to provide input for the FMI: (1) the location of the runway about which flight operations (departures and landings) will occur, and (2) aircraft position.

Figure 13 shows how a "typical" traffic pattern and how the airspace surrounding the airfield might be partitioned to include a *distance* parameter for the FMI. As can be clearly seen in the figure, flight mode inference can be made by calculating two parameters: (1) the distance from the aircraft to the airfield ($d_{ac\text{-}A}$) and the rate at which $d_{ac\text{-}A}$ changes ($\Delta d_{ac\text{-}A}$). To make the flight mode inference, $d_{ac\text{-}A}$ is compared with a fixed value, d_A, while $\Delta d_{ac\text{-}A}$ is checked for sign (a *positive* $\Delta d_{ac\text{-}A}$ indicates the aircraft is going away from the airfield, while a *negative* $\Delta d_{ac\text{-}A}$ indicates the aircraft is approaching the airfield).

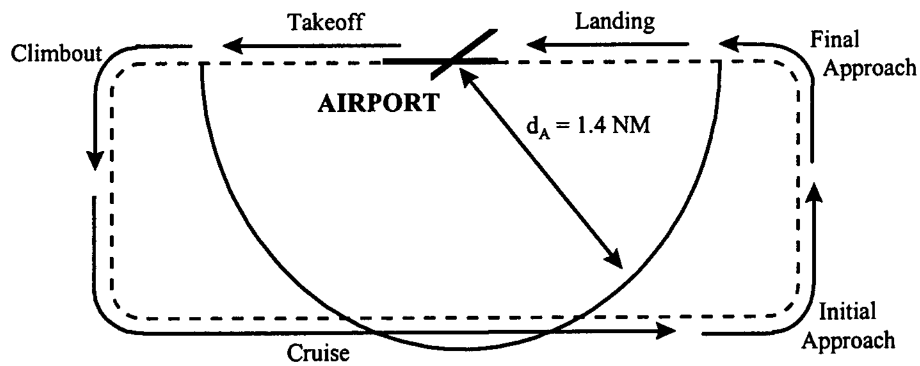


Figure 13. Space Partitioning for *Distance* Parameter: "Typical" Traffic Pattern

With these points in mind, the following five inferences are made using the *distance* parameter. Note that no inference can be made concerning the *cruise* mode.

Infer *Takeoff* if: $$(d_{ac-A} \leq d_A) \cap (positive \Delta d_{ac-A}) \tag{10}$$

Infer *Climbout* if: $$(positive \Delta d_{ac-A}) \tag{11}$$

Infer *Initial Approach* if: $$(d_{ac-A} \geq d_A) \tag{12}$$

Infer *Final Approach* if: $$(d_{ac-A} \geq d_A) \cap (negative \Delta d_{ac-A}) \tag{13}$$

Infer *Landing* if: $$(d_{ac-A} \leq d_A) \cap (negative \Delta d_{ac-A}) \tag{14}$$

It should be noted that Figure 13 depicts an arbitrary value of $d_A = 1.4\ NM$. This value was calculated by using conservative rates of climb (during takeoff) and rates of descent (during an approach), along with their associated airspeeds, to approximate the distance required to climb to an altitude of 200 feet AGL. It is anticipated that this value chosen for d_A will require some minor adjustment to optimize FMI inference. This task can be accomplished once the EFS has been modified to simulate GPS and a functional Navigation Module has been integrated into ASTRA.

Finally, these last five equations are transformed into their corresponding membership functions, in a manner analogous to that seen in the previous section, for inclusion in the FMI fuzzy rule base. For example, Figure 14 shows how the membership functions for *final approach* might appear.

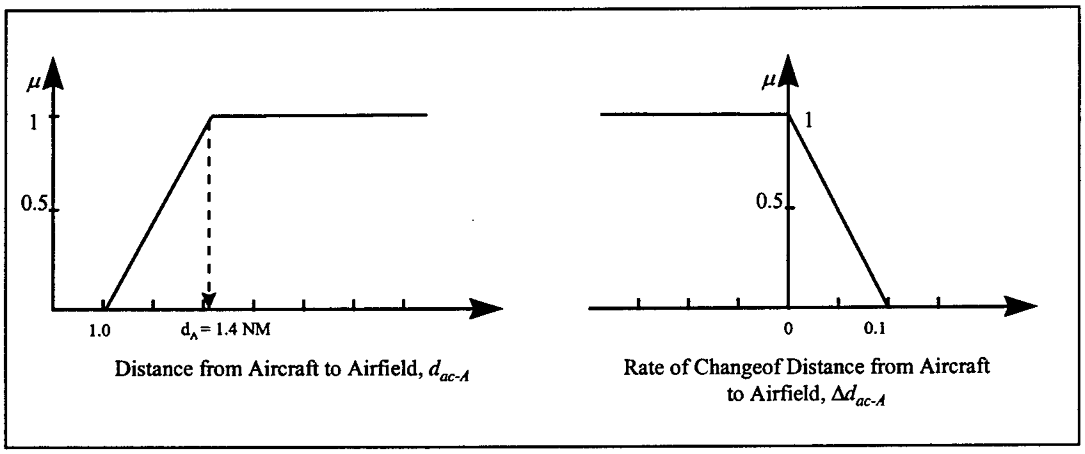


Figure 14. *Distance* Membership Functions Corresponding with *Final Approach*

To summarize then, the *distance* parameter can be used to build additional membership functions in the FMI fuzzy rule base. Including this parameter in the rule base will provide more robust FMI decisions, especially as the aircraft transitions the critical phases of *initial approach* to *final approach* and *landing*. The distance parameter can be used whether the pilot is flying "cross country" from one airfield to another, or whether he is flying "locally" about a single airport. In the former case, the pilot must enter a detailed flight plan into an

ASTRA data base, which allows the Navigation Module to make the appropriate calculations for the FMI. No flight plan is required for local flights.

ASTRA AND THE NAVIGATION MODULE

*"The great thing in this world is not so much where we are,
but in what direction we are moving."*

—Oliver Wendell Holmes

The Requirement for an ASTRA Navigation Module

To say that navigation is important to the General Aviation (GA) pilot is quite an understatement. In fact, navigating the aircraft is probably the single most time-consuming task that the pilot must perform throughout a flight. This is true whether the pilot is flying cross-country under Instrument Flight Rules (IFR), or making a short flight under Visual Flight Rules (VFR). Regardless of the type of flight he is conducting, the pilot must be continuously aware of his navigation status, (the exception to this, of course, might be during landing, when the aircraft has arrived at its destination!).

The specific navigation information the pilot must know is relatively basic. Most fundamentally, the pilot is concerned with only a few parameters. First, the pilot must know *where* he is (that is, his *present position*) and where he intends to fly (his *destination*). Knowing these two pieces of information, the pilot can calculate the *distance* and *direction* to the destination. The pilot must further know the aircraft *groundspeed* (which is influenced by the aircraft velocity and winds aloft) and present *time*, so that he can calculate *when* he will arrive at his destination and his estimated enroute time. In short, by knowing his position, groundspeed, and the location of his destination, the pilot can calculate the basic planning data needed to reach the destination. A navigation module could readily accomplish these tasks, significantly reducing the pilot's workload.

Integrating a navigation module into ASTRA further gives the pilot another important capability: *mission planning*. That is, as part of the normal pre-flight procedure, the pilot can enter his intended route of flight, permitting him to plan, before his departure, what his estimated flight parameters will be. Consequently the pilot will have the ability to answer such questions as, "Am I carrying enough fuel to reach my destination?"; "Will I arrive before sunset?"; and "How much of a delay in my arrival time can I expect, should I take an alternate route?" Answering such questions is an essential element in the mission planning process, one that requires fundamental calculations that the Navigation Module can easily provide.

While these navigation parameters are not difficult to calculate, they must be continuously updated (as the aircraft position or the winds aloft change, for example) and can be quite time consuming. Furthermore, these parameters must be immediately recalculated should the pilot need to change his original destination. For example, should the pilot encounter unexpected bad weather or an aircraft malfunction, he must quickly decide whether he should: (1) continue to the original destination, (2) return to the point of departure, or (3) find an alternate airport which is closer than the destination or the departure point. Consequently, while the basics of navigation are relatively simple, it is a dynamic and time-consuming process.

Navigation and Flight Mode Interpretation

While the most obvious use of integrating a navigation module into ASTRA is assisting the pilot in flying from one airfield to another, another use for the module is also apparent: flight mode interpretation. In other words, the calculation of the time, distance, and heading parameters require that the navigation module knows (among other things) the aircraft's *present position*. Further, since a flight is made up of a series of segments joined by waypoints, then it should be possible to define a fuzzy rule-base, based on distance and/or direction from designated waypoints, to assist the FMI in inferring the aircraft flight mode.

The Navigation Module must make one additional calculation for the FMI, namely that of the aircraft's *approximate height above ground level* (AGL). While the parameter is subtle, it is absolutely critical to ensuring the proper functionality of the FMI. The reason for this lies in how aircraft altitude was modeled in the Engineering Flight Simulator (EFS) during GAPATS development. Stated simply, the airfield model used in the EFS had a field elevation of sea level (0 feet). Consequently, as the EFS was "flown," the aircraft *indicated* altitude (the altitude, based on barometric pressure, above Mean Sea Level (MSL)) was the same as the *true* altitude (the altitude above the surface). Unfortunately, the fuzzy rule-base parameter of "Altitude," as implemented for GAPATS, equated to *true* altitude. While this parameter was adequate for use in the EFS, it would not be satisfactory for use in the aircraft. For example, an aircraft sitting on the runway of Denver International Airport would have an indicated altitude of approximately 5430' MSL. The FMI, however, would interpret this value as 5430' *above the airport surface.* Clearly the Navigation Module could (and must!) provide the correct altitude parameter to the FMI.

Used in this way, the navigation module can significantly increase the robustness and ensure the reliability of the Flight Mode Interpreter (FMI). That is, by supplementing the FMI with additional aircraft state data not previously available, it should be possible to reduce incorrect or spurious inferences, especially as the aircraft transitions from one mode to another.

To summarize, it is readily apparent that the addition of the Navigation Module to ASTRA will provide the system with tremendous capability in assisting the pilot and in increasing the robustness of flight mode inference. While GAPATS proved that it could adequately assist the pilot in basic airmanship, it had no capability to assist him in navigation. Hence, development of the Navigation Module within ASTRA was seen as essential in the development a truly valuable pilot advisory system. Such a module provides information for display to the pilot on both the Head-Up Display (HUD) and Head-Down Display (HDD), as commanded by the Pilot Advisor. HUD information may include, among other things, a *flight director*, which assists the pilot in maintaining the proper *course* and *altitude* to the destination. Likewise, HDD information may include a moving-map display, a continuously updated display which superimposes the aircraft position and route of flight (and other related data) over a digitized map. The ASTRA cockpit displays are described in greater detail in the next chapter.

The Global Positioning System

Integration of a navigation module into ASTRA requires an accurate means of sensing the aircraft's *present position*. Perhaps the most accurate means available today for measuring the position of anything—be it a person on foot, in a boat, or on an aircraft—is the Global Positioning System, or GPS. The GPS is comprised of a group of 24 satellites (collectively called a *constellation*) in six orbital plans approximately 11,000 miles above the earth. By comparing time signals from the various satellites (at least three must be in view), a civil GPS receiver can calculate it's position to an accuracy within 100 meters. In fact, when compared with conventional land-based navigation aids (NAVAIDS), GPS position accuracy is unsurpassed. Consequently, using GPS to provide aircraft position data makes it an ideal system to integrate into the ASTRA navigation module.

There are additional reasons for integrating GPS into ASTRA, as opposed to simply using the conventional NAVAIDS, which for years have been the standard in civil aviation. Most notably, the use of GPS is becoming more commonplace, even in the GA sector. In fact, its popularity has resulted in a dramatic reduction in the cost of GPS navigation systems. Consequently, the Federal Aviation Administration (FAA) is actively integrating GPS into the National Airspace System through their "GPS Approach Overlay Program" [9]. This program began in 1994 by "overlaying" GPS approaches over existing conventional NAVAID non-precision approaches. The program was then expanded to include new GPS "stand-alone" approaches, which are not overlaid over any existing approach. In the not too distant future, the use of other NAVAIDS may be phased out entirely; all non-precision instrument approaches will likely require the use of GPS.

Finally, commercial GPS systems require the use of a GPS data base for navigation. These data bases contain a wide variety of information, which are essential to the mission planning and navigation processes. Since ASTRA also requires this information (to assist the pilot with these same functions), it follows that integrating GPS and its associated data bases into ASTRA makes sense.

ASTRA Navigation Module Functionality Specification

With the background complete on what functions the Navigation Module should perform, we can now detail the input/output requirements for this subsystem. Specifically, these requirements define what calculations the module must make, along with the corresponding data the module requires to make these calculations.

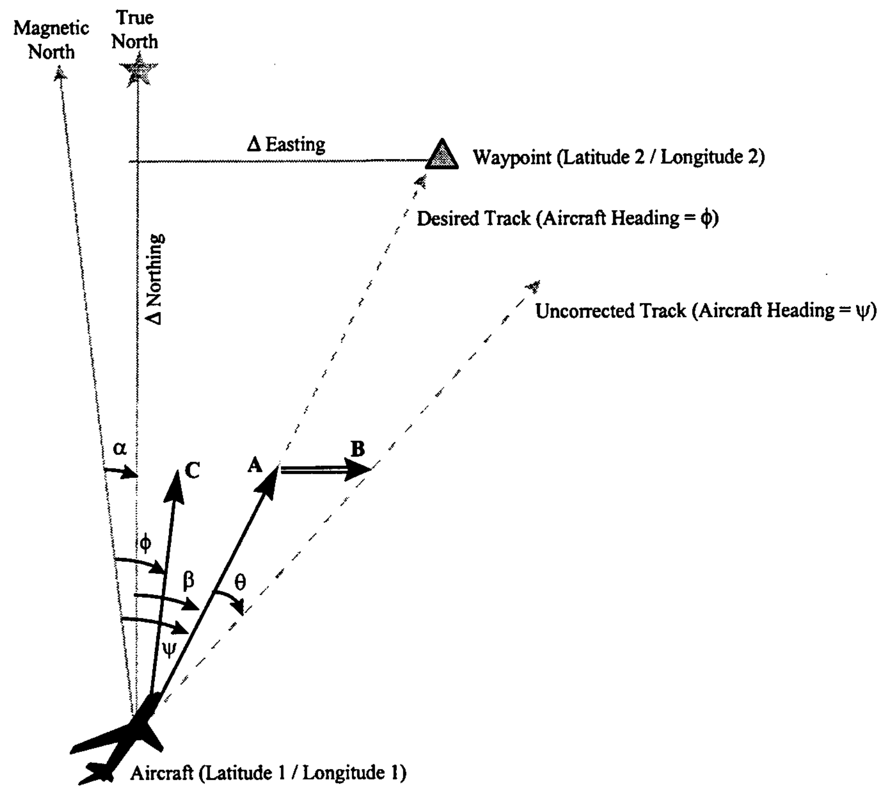
Navigation Module Input Requirements. The ASTRA Navigation Module requires the following data to make its calculations. Note that each parameter is followed by its variable name.

- Aircraft present position ($PP_{Lat/Long}$)—provided by the aircraft GPS receiver, in the form of latitude/longitude.
- Current time ($CurrentTime_{GPS}$)—also provided by the GPS receiver, with respect to "Zulu," or Greenwich Mean Time (GMT).
- Indicated Airspeed (*IAS*)—provided by aircraft pitot-static instruments and measured in "knots" (nautical miles / hour). When used together with aircraft heading (HDG), this value can be treated as a vector.
- Vertical Speed (*VS*)—provided by aircraft pitot-static instruments and measured in "feet/min."
- Indicated Altitude (h_{Ind})—provided by aircraft pitot-static instruments (barometric altitude corrected for altimeter setting) and measured in "feet MSL."
- Magnetic heading (*HDG*)—provided by aircraft heading instruments and measured in "degrees," clockwise from Magnetic North.
- Flight Plan Data Base—provided by the pilot during his pre-flight activities. This data base, used to specify the intended route of flight, is described in greater detail in the following section. As noted in the previous chapter, this data base is used for cross-country flights or flights using instrument procedures, such as holding or instrument approaches.

- Aircraft Data Base—a pre-defined data base that includes operational information pertaining to the specific aircraft being flown. Information which might be included in this data base is described in the following section.
- Navigation Data Base—a read-only data base that contains essential navigation information (such as waypoint/airfield location and elevation; location and identification of special use airspace; airway designation, location, and routing; and NAVAID position, altitude, and magnetic variation).

Navigation Module Output. As previously noted, the data from the Navigation Module will be used in other ASTRA subsystems to assist in flight mode inference, to facilitate mission planning, and to perform basic navigation. Specific parameters to be calculated are detailed in the equations below, as well as where the output will be used. (These equations will be used to generate the functional Navigation Module software, which will be developed in other graduate research efforts.)

Many of the parameters described in the equations are also depicted graphically in Figure 15. Unless otherwise noted, the following output data are *dynamic* (that is, the data must be continuously calculated and updated); *static* data need only be calculated one time. Also worth noting is that many of the calculations require the use of spherical geometry; a flat-earth model (using plane geometry) will not provide the accuracy required to certify ASTRA for flight.



α = Magnetic Variation
β = True Bearing to Waypoint ($\tan^{-1}$[Δ Easting/ Δ Northing])
ψ = Magnetic Bearing to Waypoint ($\alpha + \beta$)
θ = Wind Drift Correction
ϕ = Magnetic Course to Waypoint ($\psi - \theta$)

A: Aircraft Velocity Vector (Heading = ψ)
B: Wind Vector
C: Aircraft Course Vector (Heading = ϕ)

Figure 15. Navigation Concepts

- The following notation is used with respect to the following <u>distance</u> calculations, where $d_{AC\text{-}WP}$ denotes the distance from the aircraft present position to the waypoint position. It should be noted that this distance measure is a metric which follows the *great circle route* (the shortest arc-distance between two points on the Earth's surface, formed by a circle which passes through the two points). The distance parameters will be used by the FMI and the pilot advisor (to generate alarms). Some distance values may also be displayed on the HUD and/or HDD.

$$d_{AC-WP} = \left| PP_{Lat/Long}, WP_{Lat/Long} \right| \tag{15}$$

- Distance from aircraft to departure airfield: $d_{AC\text{-}Dep}$
- Distance from aircraft to the destination airfield: $d_{AC\text{-}Dest}$
- Distance from aircraft to the next waypoint: $d_{AC\text{-}NxtWP}$
- Distance from aircraft to the previous waypoint: $d_{AC\text{-}PrvWP}$
- Distance from aircraft to the nearest airfield: $d_{AC\text{-}Near}$
- Distance from aircraft to the Initial Approach Fix (IAF): $d_{AC\text{-}IAF}$
- Distance from aircraft to the Final Approach Fix (FAF): $d_{AC\text{-}FAF}$
- Distance from aircraft to the Missed Approach Point (MAP): $d_{AC\text{-}MAP}$
- Distance from the aircraft to the nearest special use airspace: $d_{AC\text{-}SUA}$ (In this situation, special use airspace would include Prohibited Areas, Restricted Areas, Warning Areas, Military Operations Areas, Alert Areas, and other high density traffic areas, such as Class B and Class C airspace.)
- Distance from the IAF to the FAF (static): $d_{IAF\text{-}FAF}$
- Distance from the FAF to the MAP (static): $d_{FAF\text{-}MAP}$
- Distance from the destination airfield to the MAP (static): $d_{MAP\text{-}Dest}$

- The following notation is used with respect to the following direction calculations, where $\angle \psi_{P\text{-}W}$ denotes the *true* direction from the aircraft present position to the waypoint position.. Note that any true direction must be converted to a *magnetic* value by applying the appropriate magnetic variation (α) from the navigation data base.

- Bearing from aircraft to waypoint (*BRG*). This value is the straight line magnetic direction (no wind) from the aircraft to the waypoint.

$$\angle BRG = \tan^{-1}(PP_{Lat/Long}, WP_{Lat/Long}) + \alpha \tag{16}$$

- Aircraft Track (*TRK*). This value is the straight line magnetic direction formed by the path which the aircraft traces over the ground. Note that when winds are present, the aircraft track and aircraft bearing will not be the same.

$$\angle TRK = \tan^{-1}(PP_{Lat/Long1}, PP_{Lat/Long2}) + \alpha \tag{17}$$

- Aircraft Course (ϕ). This is the *wind-corrected* magnetic direction that the aircraft must fly to maintain the proper track to the desired waypoint. The course would be used by the pilot advisor and displayed on the HUD and HDD.

$$\angle\phi = \angle BRG + \theta \tag{18}$$

- Groundspeed (*GS*). Groundspeed can be found by dividing the distance traveled by the elapsed GPS time. By associating this value with the aircraft track (*TRK*), *GS* can also be treated as a vector. Groundspeed could be displayed on the HUD and/or HDD.

$$GS = \frac{d_{1-2}}{\Delta t} \tag{19}$$

- Current winds aloft (*WS / WD*). This includes the calculation of two variables, wind speed and wind direction. These values can be found by treating Indicated Airspeed (*IAS*)and Groundspeed (*GS*) as vectors. Winds aloft will be displayed on the HUD and HDD.

$$\left|\vec{WS}\right| = \left|\vec{IAS} - \vec{GS}\right| \tag{20}$$

$$\angle WD = \tan^{-1}\left(\vec{WS}\right) + \alpha \tag{21}$$

- Aircraft altitude above ground level (h_{AGL}). This value can be estimated as the difference between the aircraft indicated altitude and the surface elevation at the aircraft's present position. This latter value can in turn be approximated by the field elevation of the nearest airfield, waypoint, or NAVAID. The altitude above ground level is required for use in FMI calculations.

$$h_{AGL} = h_{Ind} - h_{PP} \tag{22}$$

- Estimated time enroute and estimated time of arrival to a Waypoint (*ETE / ETA*). These times must be calculated for several waypoints: the destination, the next waypoint, and the previous waypoint. The calculation of these times is important to pilots when conducting mission planning, and will be displayed on the HUD and HDD. Note that the ETA should be formatted to a 24-hour display, with respect to GMT. The general formula is:

$$ETE = \frac{d_{AC-WP}}{GNDSPD} \tag{23}$$

$$ETA = CurrentTime_{GPS} + ETE \tag{24}$$

- Descent point (*DP*). The descent point is that position (along the desired track when in the *cruise* mode) where the pilot should begin his descent (into an airfield) in order to maintain a desired descent angle (γ) or rate of descent (*ROD*). (Both these values would be included as part of the Flight Plan Data Base, along with appropriate default values.) Note that when using *ROD* (typically expressed in "ft/min"), the aircraft groundspeed will require conversion from "knots" to provide consistent units. The DP can be approximated by:

$$DP = \frac{h_{AGL}}{\tan \gamma} \tag{25}$$

$$DP = \frac{h_{AGL}}{\tan\left({}^{ROD}\!/_{GNDSPD} \right)} \tag{26}$$

Data Bases

As indicated in the previous section, the Navigation Module requires access to a number of data bases in order to be fully functional. A brief description of each of these data bases follows in the paragraphs below.

The Flight Plan Data Base. This first data base stems from the requirement for the pilot to identify, as part of his mission planning, all of the essential parameters which together define the flight plan. This information, which encompasses the Flight Plan Data Base, must contain (as a minimum) the route of flight, the desired flight altitude and airspeed, and the destination airfield. In general, the more detailed the flight plan, the better—giving the Navigation Module a "better idea" of the pilot's intentions. Specifically, the flight plan data base should include:

- Departure airfield and time of departure (with respect to GMT).
- A detailed route of flight, to include the standard routing (if appropriate), airways, waypoints, and the destination airfield.
- Cruise altitude and airspeed.
- If holding is desired (or anticipated), the holding fix, the holding airspeed, and the expected number/direction of turns (if non-standard).
- Expected winds aloft (direction/velocity).
- Estimated time enroute.

- Approach identification (type and direction of landing).

Note that the route of flight is defined by a series of navigation waypoints (to include NAVAIDS and fixes), airways, and any appropriate standard instrument departure/arrival routing (known as a SID and STAR, respectively). The route of flight should also include "special" waypoints, such as the Initial and Final Approach Fixes and the Missed Approach Point, which will be determined by the type of instrument approach being flown. The pilot identifies each segment of the flight plan by using the appropriate name or identifier associated with the waypoint, as found in the Navigation Data Base. (For example, the College Station VOR is identified by "CLL.") The Navigation Module can then call the Navigation Data Base to cull the appropriate information (such as latitude, longitude, and altitude) required to make its calculations. Should the pilot enter an incorrect identifier, an error message would be generated, notifying the pilot of the error and prompting him for a correction.

The Aircraft Data Base. This data base contains operational information specific to the aircraft being flown (in the case of ASTRA, the Commander-700). Such data might include stall airspeeds (as a function of aircraft configuration); best cruise, endurance, and climb airspeeds; basic weight and balance information; operating limitations (such as wing flap and landing gear airspeed limits); and basic fuel information (capacities, consumption rates, etc.). Figure 16 depicts a data file that contains an abbreviated sample of airspeed information (taken from the aircraft operator's manual) which would be included in the Commander-700 Aircraft Data Base.

```
// RECOMMENDED AIRSPEED OPERATING PROCEDURES (all airspeeds are KIAS!)

 V_rotate=80           // V_rot = 80
 V_climb_min=120       // V_y = 120
 V_approach=90         // V_app = 87 (minimum)

// AIRCRAFT AIRSPEED LIMITATIONS (all airspeeds are KIAS!)
//  In general, a buffer of ~5 knots has been utilized, erring on the
//  side of safety...  This should allow the pilot sufficient time to
//  react before exceeding a limit.

//  Landing Gear and Flap Limits

V_GearRetract=130      // V_lor = 137
V_GearExtend=150       // V_loe = 155
V_GearDown=150         // V_le = 155
V_FlapsTakeoff=150     // V_fe = 155 (for 12 degrees)
V_FlapsLanding=120     // V_fe = 128 (for 35 degrees)

//  Aircraft Stall Speeds (as a function of configuration)

V_CleanStall=90        // V_st = 86 (gear UP and flaps UP)
V_NoflapStall=100      // Estimated (gear DOWN and flaps UP)
V_LandingStall=70      // V_st = 65 (gear DOWN and flaps LANDING)

//  Single-Engine Operations

V_SingleEngineMin=80         // V_mca = 75
V_SingleEngineClimb=105      // V_yse = 103
```

Figure 16. Abbreviated Aircraft Data Base for the Commander-700

Although the information from Figure 16 would be "pre-loaded" as an integral part of the ASTRA software module, the pilot should have the ability to view/verify this data base as necessary. Furthermore, the pilot should be able to update non-critical fields of the data base, such as weight and balance information. The remaining flight-critical data fields, however, should be in a "read-only" format.

The Navigation Data Base. This data base, which must be procured from an external commercial source, contains a myriad of navigation information concerning virtually every facet of aviation—to include comprehensive data on airports, airspace, navigation aids, and communications. For example, one such data base (issued by Jeppesen) contains more than 28,000 records for the geographic area encompassed by Dallas, Austin, and Houston. Further, each *airport* record type contains 19 data fields, including: airport identifier, speed limit altitude, longest runway, latitude, longitude, magnetic variation, field elevation, VHF frequency, and airport name. It is evident that to utilize a data base of this magnitude, ASTRA will require an extremely efficient search algorithm, which extracts only the necessary data required by the Navigation Module. (The data extracted will be that associated with a particular identifier or name, as entered by the pilot into the Flight Plan Data Base). It should also be noted that the Navigation Data Base is only available in a "read-only" software format and must be updated every 28 days, in order to comply with the FAA's safety-of-flight requirements.

In summary, the Navigation Module plays two key roles in the ASTRA system architecture. It first makes the appropriate *altitude* and *distance* calculations required by the FMI to make flight mode inferences. It also facilitates the mission planning and navigation process, thereby greatly reducing pilot workload and enhancing situational awareness. This in turn allows the pilot to focus his attention on controlling the aircraft. Consequently, the specification detailed in this chapter provides the structure required to develop a fully functional Navigation Module which meets the systems requirements of ASTRA.

ASTRA FLIGHT DISPLAYS

"Few are those who see with their own eyes..."

—Albert Einstein

Background

One of the critical issues which any smart cockpit system must address is how the pilot interacts with the system. This issue is especially pertinent when the goal of the system is to enhance safety and situational awareness, without increasing pilot workload. In fact, even a perfectly designed Flight Mode Interpreter (FMI) and Pilot Advisor (PA) would be of little use if the pilot could not effectively communicate with them.

In the current ASTRA system design, such pilot interaction is provided for by two separate flight displays, the Head-Up Display (HUD) and the Head-Down Display (HDD). The HUD, which is considered a primary flight display, is designed to provide the pilot with all the essential information necessary to fly the aircraft, in any environment. By looking through the HUD, the pilot has the ability to view this flight information while looking outside the aircraft. Consequently, the pilot is able to maintain focus on controlling the aircraft while simultaneously searching for unknown flight hazards. In this respect, the HUD is an essential element for enhancing the pilot's situation awareness and increasing flight safety.

The HDD, while not a primary flight display, is an equally critical component of the ASTRA system architecture. The importance of the HDD lies in the fact that the pilot can not only view a wide variety of information displayed on the HDD, but also because the pilot communicates with ASTRA through the HDD. That is, the pilot must use the HDD to perform any mission planning, such as editing a flight plan, entering a clearance from Air Traffic Control, or modifying the aircraft weight and balance calculations. Consequently, the HDD design must permit the display of a wide variety of information.

After providing a general philosophy for the design of the ASTRA flight displays, this chapter provides detailed examples of how the HUD and HDD display sets might be configured. Appendix D details a rule base which the PA can use to appropriately configure the HUD and HDD. As will be seen, these display sets are a function of (1) the flight mode, and (2) the specific tasks the pilot is performing in the conduct of the flight. It is important to note, however, that the proposed display sets (as well as the PA rule base) should be considered a "springboard," from which many modifications will take place. That is, once the displays are integrated into the Engineering Flight Simulator (EFS), a thorough evaluation for each display configuration will be necessary. The display configurations can only be optimized by using the EFS to fly a variety of mission tasks, such as traffic patterns and instrument approaches.

General Design Philosophy

There are a number of important issues that weigh in on the design of any aircraft display, especially when it is to be used for pilotage, navigation, or mission planning. These issue might be broadly classified into two major categories: (1) flight certification, and (2) human factors. The first category entails satisfying the minimum requirements, established by Federal Aviation Administration (FAA), which validate the safe use of the displays for flight. For example, the layout of a primary flight display must meet the stipulations detailed in [6] and [12], as summarized below:

- The display of basic flight instruments should preserve the so called "T-format" of the counterpart mechanical aircraft flight instruments.
- Many flight instruments, such as Bank Angle Indicators, have several acceptable symbolic formats. In general, if more than one display format exists, then the format which most closely mimics the arrangement and behavior of its mechanical counterpart should be used.
- The failure of any sensor, instrumentation item, or ASTRA subsystem, which provides data to be displayed on the HUD or HDD, should result in that information disappearing from the display, or in the display format of that information being modified (thereby indicating to the pilot the presentation of unreliable flight data).

The second general category, that of ergonomics, is concerned with displaying the appropriate type of flight information, in a manner which enhances the pilot's situational awareness, increases safety, and reduces the amount of pilot workload. Some of the basic human factors issues considered in the design of the HUD and HDD are detailed as follows:

- The display should not have excessive or unnecessary symbology, resulting in a "cluttered" appearance. Consider the use of "display by exception:" a parameter is displayed <u>only</u> when approaching/exceeding an acceptable value.
- All information displayed should be completely unambiguous. In other words, the pilot should not be able to misinterpret any information, such that he performs an unsafe act in reaction to a displayed piece of symbology.
- The symbology should be intuitive and simple to learn, without the requirement of formal training (a user-friendly display). Data input must be structured in a logical manner.
- Whenever possible, information should be displayed using symbols rather than words. This will, in general, tend to reduce display clutter. However, this guideline should not be used at the expense of safety—if alphanumerics are the best way to display a message, then do so.
- Use and manipulation of the display must not increase pilot workload. The use of *Automatic Mode Switching* (AMS) should be maximized (this concept was detailed in Chapter 2), as should "default" parameter values. However, the pilot should have the ability to override the display selected by AMS at any time.

- Information must be displayed in a timely manner, allowing sufficient pilot reaction time.
- The display must be readable and legible under all lighting conditions.

Display Alarm Definition

As detailed in earlier chapters, *alarms* are generated by the Pilot Advisor (PA) any time it detects a *non-nominal* flight condition. Such alarms can be the result of a piloting error (such as forgetting to retract the landing gear after takeoff) or an abnormal aircraft condition (such as excessively low engine oil pressure). In either case, the alarm must be displayed in a manner appropriate for the pilot to take corrective action. Consequently, the alarm may be displayed on the HUD and/or the HDD, as a function of how "serious" the alarm is.

Keeping this in mind, alarms may be categorized into three distinct levels which facilitates the design of alarm display (that is, how and where it should be displayed). As will be seen, each alarm category is displayed in a *unique* region of the HUD/HDD, allowing the pilot to immediately recognized the "degree" of urgency. Appendix D describes in great detail the conditions which may generate many of these alarms, as well as how the alarms might be presented on the HUD/HDD. The alarm levels are defined as follows:

- Level I (*Advisory*): This "low-level" alarm is advisory in nature and generally does not require immediate pilot action. Advisories are probably best displayed on the HDD, but may be displayed on the HUD as necessary.
- Level II (*Caution*): This "medium-level" alarm indicates that the aircraft could be damaged if appropriate pilot action is not taken. Cautions should be displayed on both the HUD and HDD.
- Level III (*Warning*): This "high-level" alarm indicates that immediate pilot action is required to prevent pilot injury or death, or to preclude serious aircraft damage. Warnings should be displayed on both the HUD and HDD.

The Head-Up Display Modes

As will be seen in subsequent paragraphs, the display modes for the HUD correspond with the flight mode inference of the FMI. Consequently, each flight mode will have a *unique* display configuration, which includes that information appropriate for that flight mode. Each display set was essentially built "from the ground up," by considering a number of factors. For example, each display set follows the minimum requirements stipulated by the FAA [12]. From this starting point, the display sets were then modified or augmented with features seen in a number of sources [3], [23] and by using the author's engineering flight test experience in various aircraft and flight simulators with "glass cockpits." The displays for two such aircraft are described in [17] and [18].

Note that AMS is used to select the appropriate display configuration as the aircraft transitions from one flight mode to the next. However, provisions are made which allow the pilot to manually select an alternate display configuration (other than that selected by AMS).

To date, a HUD has not been procured for integration into the EFS for ASTRA development. Consequently, it is assumed that the HUD will be a monochrome display. With the exception of a power switch, it is further assumed that the pilot will control the HUD (such as varying brightness/contrast or selecting alternate display modes) entirely through the HDD.

Basic (Default) Mode. In *Basic* mode, the HUD displays a default symbology set, providing the minimum required information for basic flight. Consequently, should ASTRA or any of its subsystems fail, this mode will provide enough information for continued safe flight in Instrument Meteorological Conditions (IMC). Figure 17 depicts the HUD *Basic* symbology set, which includes the following information: horizon line with pitch ladder; aircraft symbol; integrated barometric altitude and vertical velocity; indicated airspeed; and heading/turn information (heading tape, present heading, bank angle, slip/skid, and rate-of-turn).

Note that the display includes unique fields for *Warning*, *Caution*, and *Advisory* (WCA) data. These fields, indicated by dashed boxes in the figure, are displayed only when an alarm is active. That is, if there are no WCA's, their corresponding display fields remain blank.

Conversely, if there is an active WCA, the corresponding alarm field will display a message in a manner appropriate to the alarm. For example, a *Warning* may be displayed in "reverse video" or so that it flashes, subsequently catching the pilot's attention more quickly. Note that the location of the WCA fields remains fixed for all HUD display modes.

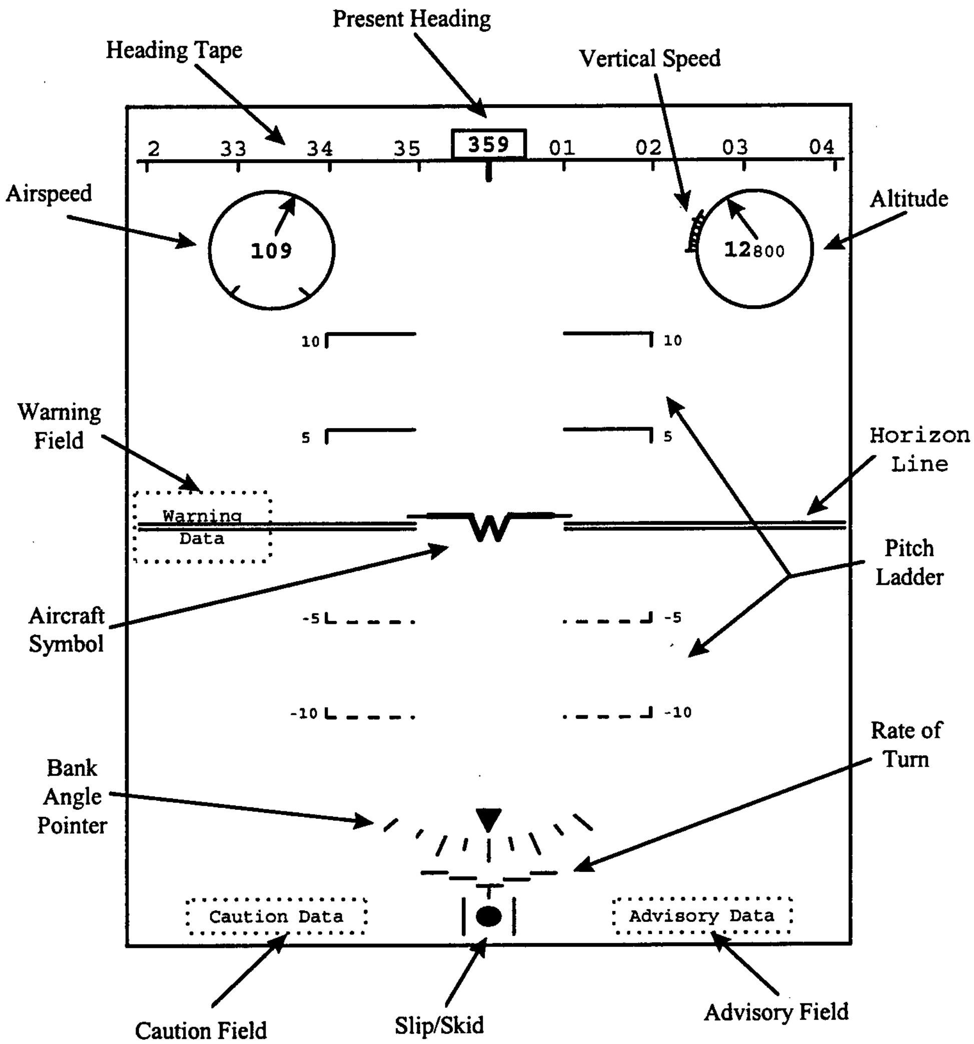


Figure 17. HUD *Basic* Symbology Display Set

The remaining HUD display formats, presented in subsequent sections, each build from the *Basic* display set. In other words, additional information, appropriate to the current FMI mode inference, is added to the new display set. Consequently, subsequent display formats will display, at minimum, that information depicted in the *Basic* display set.

Taxi Mode. The *Taxi* Mode is identical to the *Basic* mode, except that two additional pieces of information may be displayed, as shown in Figure 18. The first, is a "Mode: TAXI" advisory which indicates that ASTRA is functioning properly. The second is a *heading*

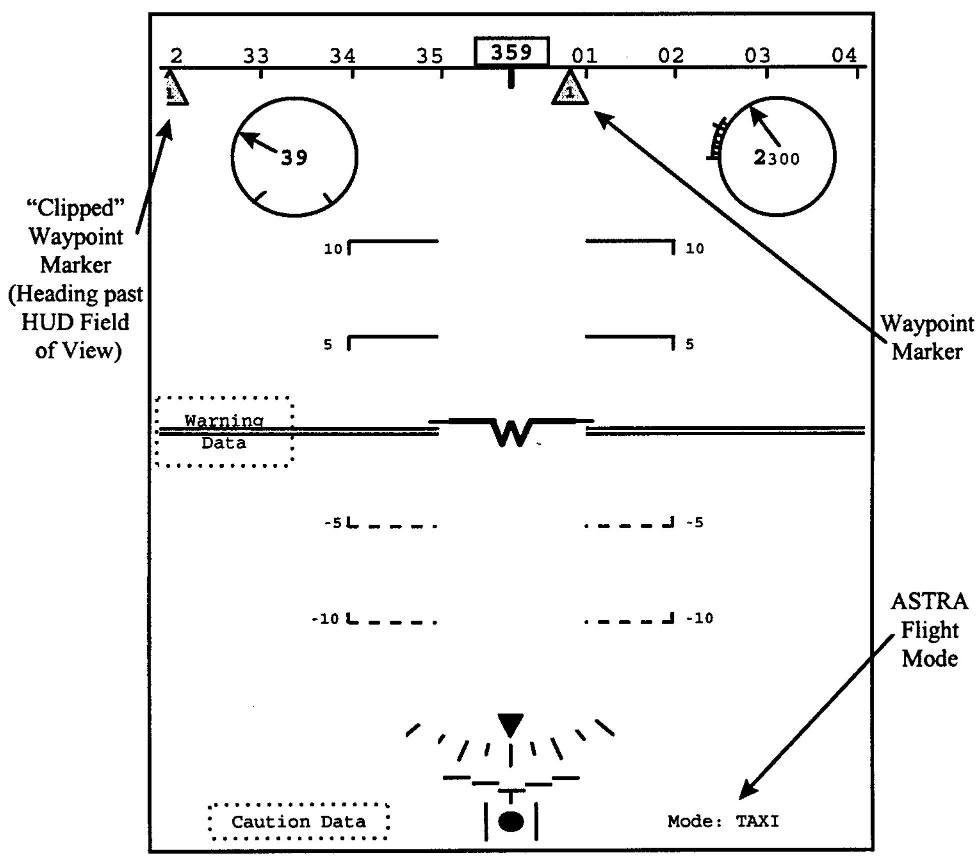


Figure 18. HUD *Taxi* Symbology Display Set

carat, which may be displayed beneath the heading tape, showing the magnetic course to the first waypoint. This heading marker allows the pilot to anticipate the initial aircraft following takeoff. Note that the heading carat would be displayed only in the case when the pilot has activated an ASTRA flight plan through the HDD. In the case that the initial heading is beyond the HUD "field of view," (for example, at 180° in Figure 18), then a "clipped" heading carat is be displayed on the side of the Heading Tape nearest the initial heading. As the initial heading enters the HUD field of view, the clipped heading carat is displayed normally.

Takeoff Mode. The *Takeoff* mode further builds from the display set seen for *Taxi* mode. First, the aircraft groundspeed is presented next to the indicated airspeed. Second, several *airspeed carats* are included within the airspeed display, indicating several critical airspeeds: V_r, the airspeed at which the aircraft is rotated to initiate flight; V_y, the airspeed yielding the best rate of climb; and V_x, the best angle of climb airspeed. The first of these carats, corresponding to V_r, could be automatically removed from the display once the PA senses that the aircraft is airborne.

Also note the addition of a *flight director* (FD), which gives the pilot *longitudinal* and *lateral* steering cues. To follow the FD commands, the pilot need only to superimpose the aircraft symbol directly over the FD. In the present case, the longitudinal steering cue indicates the proper pitch attitude needed to maintain flight at V_y, while the lateral steering cue indicates that the pilot should maintain the current runway heading until the aircraft reaches sufficient altitude to initiate a turn. As might be expected, the ASTRA advisory is also changed to indicate the new flight mode inference. Figure 19 depicts the HUD *Takeoff* display set.

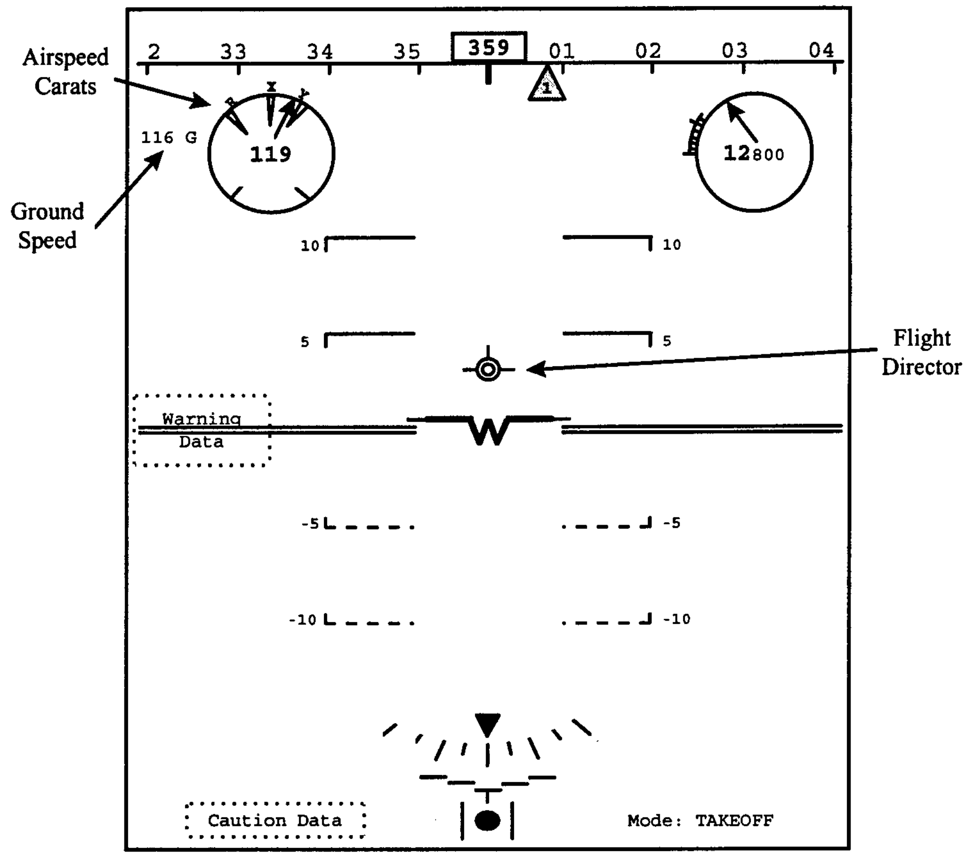


Figure 19. HUD *Takeoff* Symbology Display Set

Climbout Mode. As with the previous case, the *Climbout* display mode builds from its predecessor. Two additional airspeed carats are added within the airspeed display: V_{lor}, the maximum airspeed at which the landing gear can be retracted; and V_{fe}, the maximum airspeed at which the aircraft can be flown with the flaps extended. As was seen for the

Takeoff mode, the PA automatically removes these two carats when the pilot retracts the landing gear and wing flaps. Once again, the advisory field is updated to indicate the transition to this newly inferred flight mode, as reflected in Figure 20.

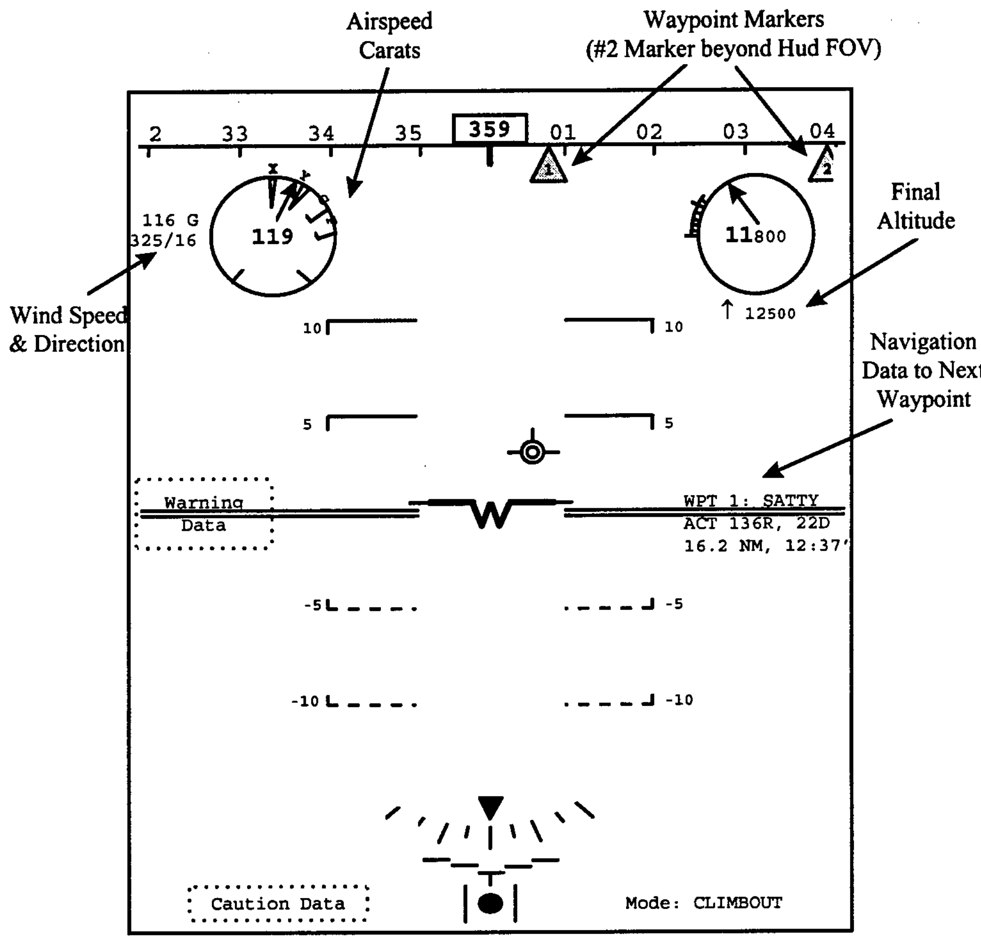


Figure 20. HUD *Climbout* Symbology Display Set

The figure also shows the addition of several items taken from the ASTRA flight plan. The *initial cruising altitude* is displayed beneath the altimeter. The FD will provide longitudinal steering guidance to maintain V_y until this altitude is reached. Note that the FD is now providing new lateral steering guidance, indicating the course the pilot should maintain to reach the first waypoint. As the aircraft approaches the first waypoint (say, within two minutes), a heading carat for the next waypoint appears, allowing the pilot to anticipate his next course. Should this subsequent heading lie beyond the HUD field of view, then the new waypoint carat would be "clipped," as was described in the *Taxi* display mode. Once the aircraft passes the initial waypoint, its heading carat disappears from the display.

Specific navigation information regarding the first waypoint is also displayed on the right of the horizon line. This information might include waypoint number, identifier, and location, as well as time and distance information to the waypoint. Finally, winds aloft data (wind speed and direction) is presented beneath the aircraft groundspeed.

Cruise Mode. The *Cruise* display mode, depicted in Figure 21, represents a decluttered version of that seen for *Climbout*, with all airspeed carats having been removed. In the *Cruise* configuration, the FD continues to give lateral guidance to the next waypoint and longitudinal guidance for the assigned altitude. Note that a minimum safe altitude is added to the display, directly beneath the assigned cruising altitude.

Initial Approach Mode. As might be expected, this HUD mode adds the information necessary to execute an instrument approach. While the FD continues to provide cues for maintaining course and altitude, raw ILS approach data is also displayed. This data enhances the pilot's situational awareness throughout execution of the approach. It further permits the pilot to continue using the HUD to fly the approach, should the FD fail. Furthermore, the waypoint carats are now labeled with "I" and "F," corresponding to the Initial Approach Fix (IAF) and Final Approach Fix (FAF), respectively. Likewise, navigation data is updated as the aircraft passes over each of these fixes, as are assigned and minimum altitudes.

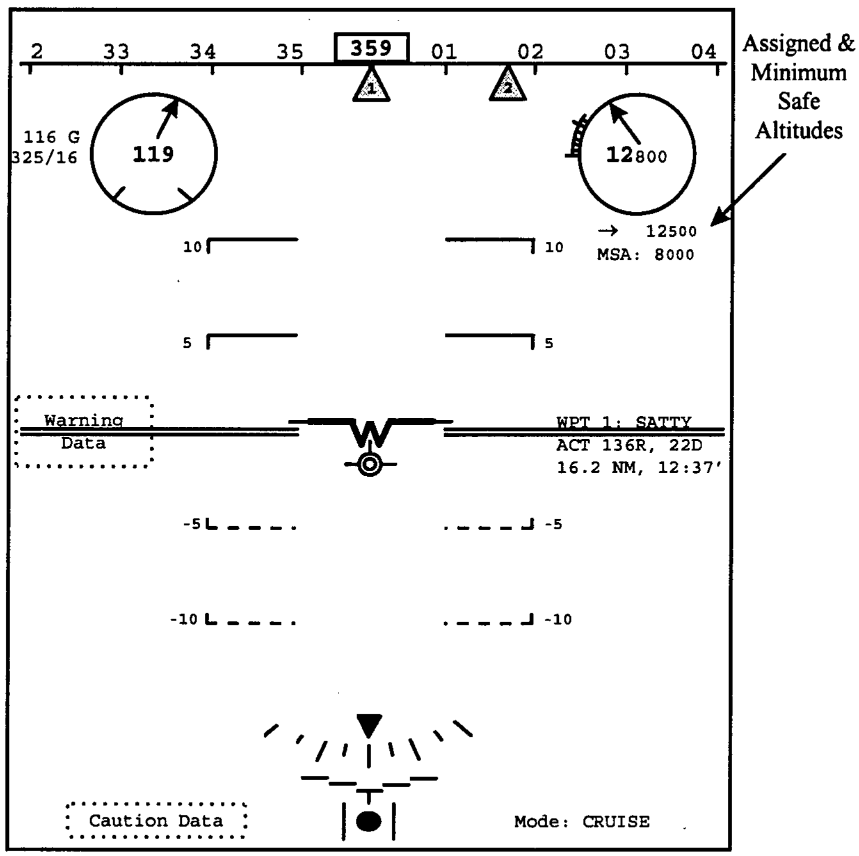


Figure 21. HUD *Cruise* Symbology Display Set

Airspeed carats are reintegrated within the airspeed display, indicating appropriate airspeed limits for flap and landing gear extension, as well as the recommended airspeed for approach execution. Furthermore, should the PA detect adverse crosswind conditions relative to the approach being flown, an appropriate caution is generated. As with the other HUD configurations, the flight mode advisory is updated accordingly. Figure 22 depicts the *Initial Approach* display set.

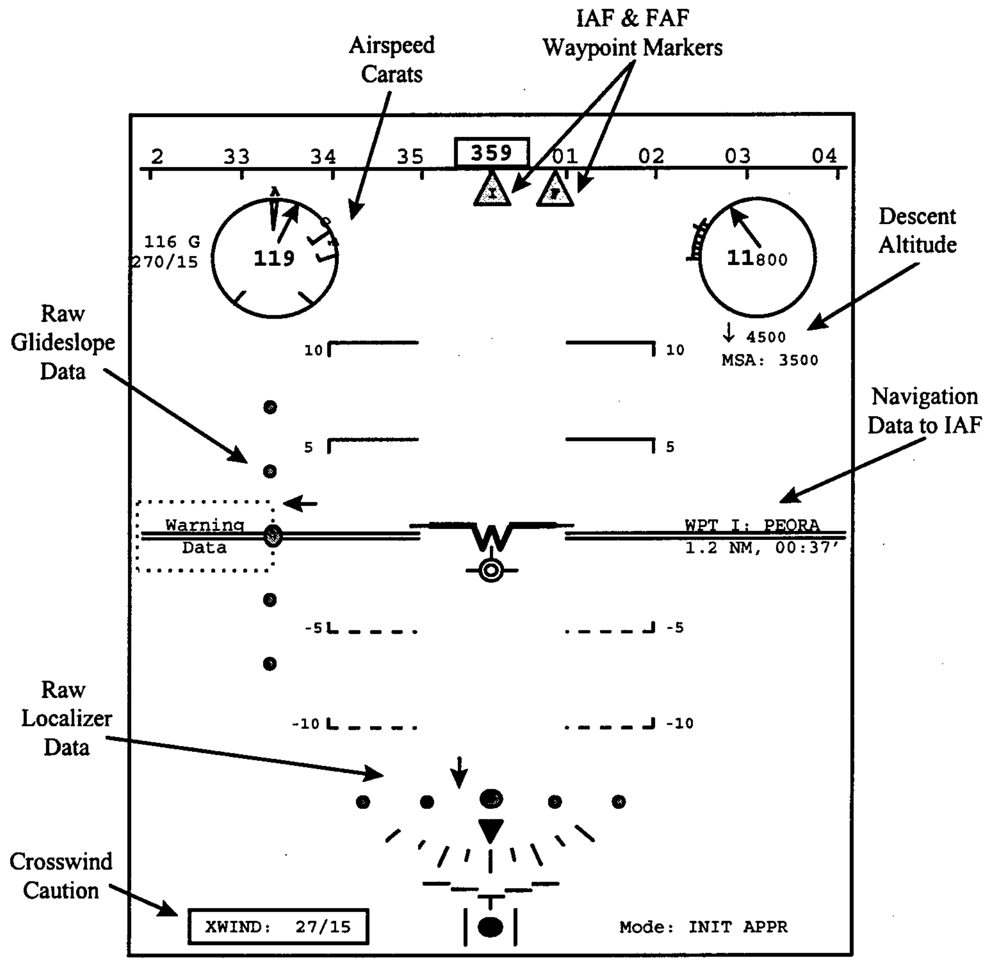


Figure 22. HUD *Initial Approach* Symbology Display Set

Final Approach Mode. The HUD configuration in this display mode is nearly identical to that corresponding with *Initial Approach*, as seen in Figure 23. To further enhance the pilot's situational awareness, a symbolic runway is added to the display, which accurately depicts the location and orientation of the intended runway for landing. Raw ILS data is displayed as before, as is the FD. Though not depicted in the figure, alternate FD display formatting might also be appropriate, such as those described by Ward and Woo [23]. In general, the goal of these alternate FD's is to further enhance safety and situational awareness while performing the critical tasks associated with an instrument approach.

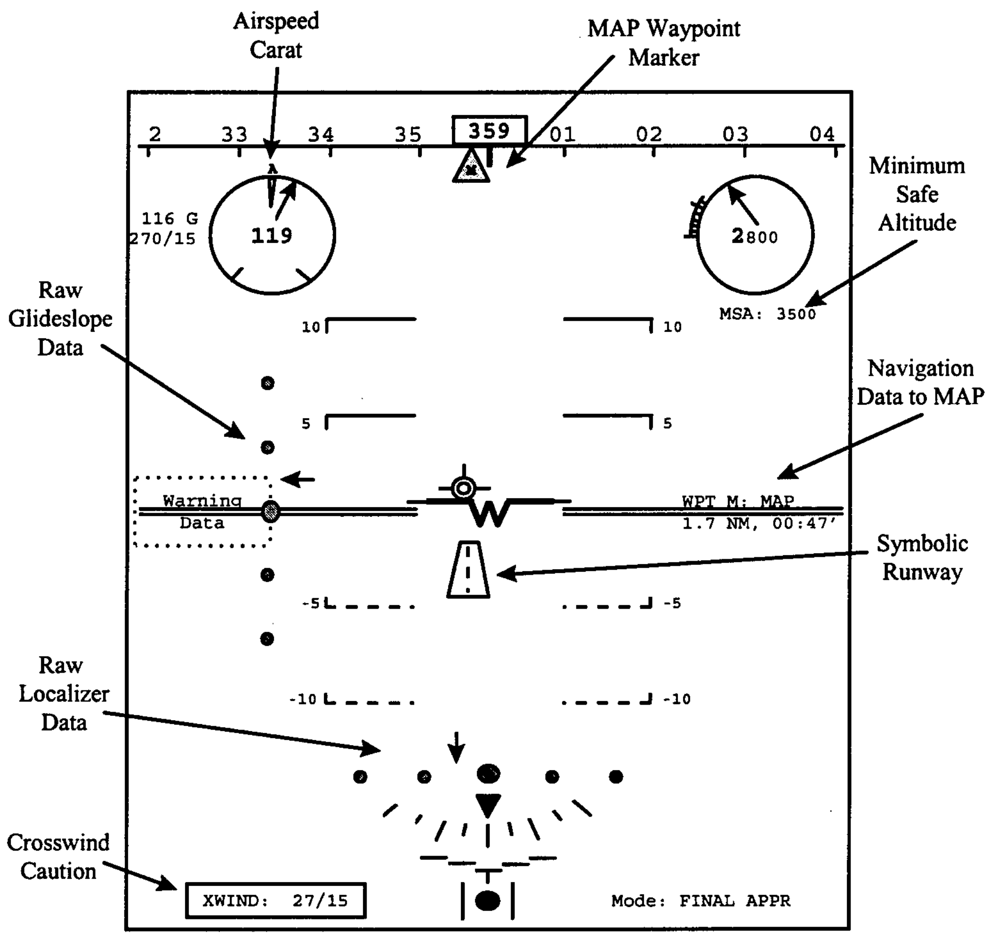


Figure 23. HUD Final Approach Symbology Display Set

Navigation and altitude information, as before, is updated and displayed for the FAF and the Missed Approach Point (MAP). Likewise, as the pilot configures the aircraft for landing (by extending the wing flaps and landing gear), the corresponding airspeed carats are removed from the display. The airspeed carat corresponding to the approach airspeed is still displayed. Crosswind cautions are also displayed, whenever appropriate. Finally, the flight mode advisory is updated to indicate the transition to the newly inferred flight mode.

Landing Mode. The symbology set displayed for the *Landing* mode attempts to declutter the display to the maximum extent possible, as shown in Figure 24. By removing all extraneous information from the HUD, the *Landing* mode therefore facilitates the transition the pilot makes from instrument flight to a visual landing.

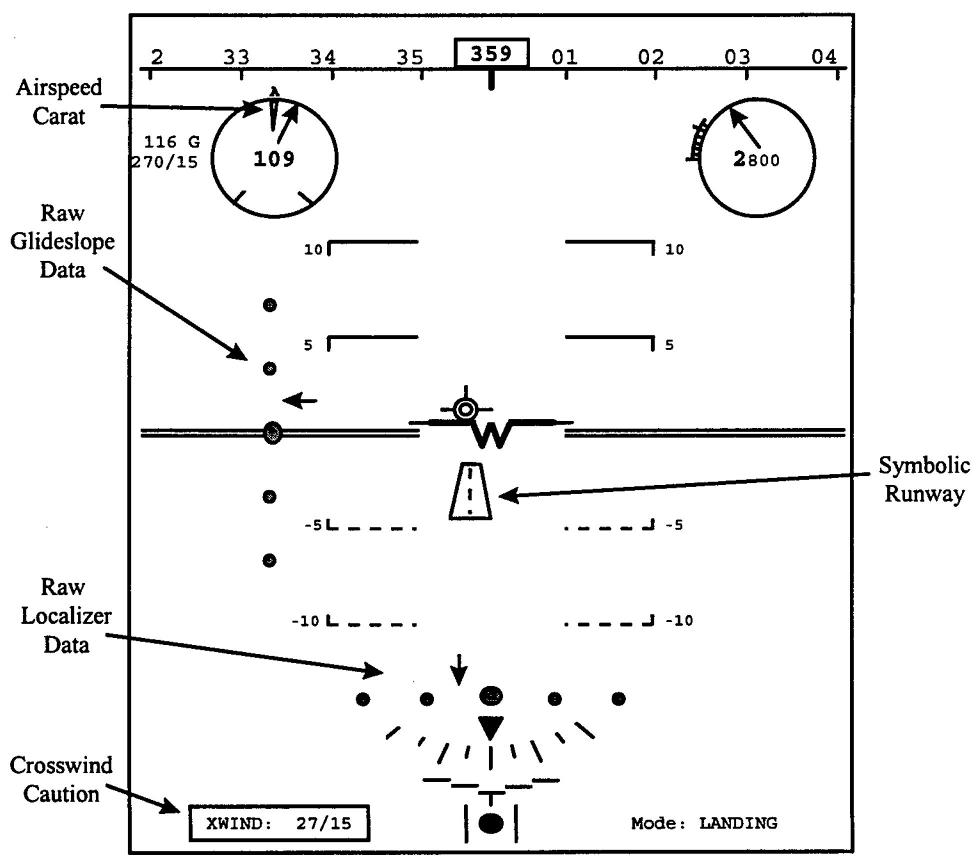


Figure 24. HUD *Landing* Symbology Display Set

The symbolic runway is still displayed so that the pilot might anticipate the runway location and orientation. Raw ILS data is also displayed, in the event that the pilot must execute a missed approach. The airspeed carat, corresponding to the appropriate instrument approach speed, is still displayed as well. Crosswind cautions are displayed and flight mode advisories are updated as appropriate.

The Head-Down Display Modes

Because the HDD must have the ability to display such a wide variety of data, it has been designed to function as a *Multi-Function Display* (MFD) in ASTRA. Such MFD's, which are commonly used in many military and commercial air carrier applications, allow the pilot to enter and manipulate data and change display configurations. However, the use of these MFD's has not been seen in general aviation. Consequently, the HDD display sets presented in subsequent paragraphs were built "from the ground up."

As was the case with the HUD, the HDD display sets adhere to the same stipulations detailed by the FAA [12]. These HDD display sets were developed by using a display architecture similar to that in one military aircraft [18] as a foundation. However, the ASTRA display architecture for the HDD focuses on the pilot interface requirements from a *general aviation* perspective. That is, the author's flight test experience was used to develop the HDD display sets by focusing those tasks the GA pilot routinely performs. Finally, it is important to note that while the use of an MFD requires the pilot to manually select the specific display set he wishes to use, it will be seen that Automatic Mode Switching can be incorporated into several of the display modes, reducing much of the pilot workload requirements.

As can be seen in Figure 25, MFD's typically have a number of push-buttons (hereafter called *function keys*) surrounding the display area. Each function key has an associated display label, which indicates the type of information that will be displayed when that key is pushed. Furthermore, once a function key is pushed, the display labels for any (or all) of the remaining keys may change, giving the pilot access to additional display options. An MFD may also have one or more *data keys*, by which the pilot can enter specific data pertaining to the function key previously selected. Data keys may take the form of a push-button (as in a keyboard) or a rotary dial (as in a volume or tuning knob).

The display area, located in the center of the HDD, is used to show the information relating to the function key that the pilot selects. Note that when the pilot selects a function key, its associated display label changes to "reverse video," allowing the pilot to immediately recognize in which mode the HDD is configured. Like the HUD, the HDD includes unique display fields for WCA data, which are displayed only when an alarm is active. These WCA field locations remain fixed for all HDD display configurations.

As was the case for the HUD, hardware for a HDD has yet to be procured. Consequently, the display architecture described in this section is intended to be as general as possible. However, it is assumed that the HDD will have a color display, to facilitate the FAA certification process discussed earlier. It is further assumed that the operational HDD will have a minimum of 12 function keys and four data keys, configured as shown in Figure 25 (four functions keys to the left, top, and right of the display area, and four data keys below the display area).

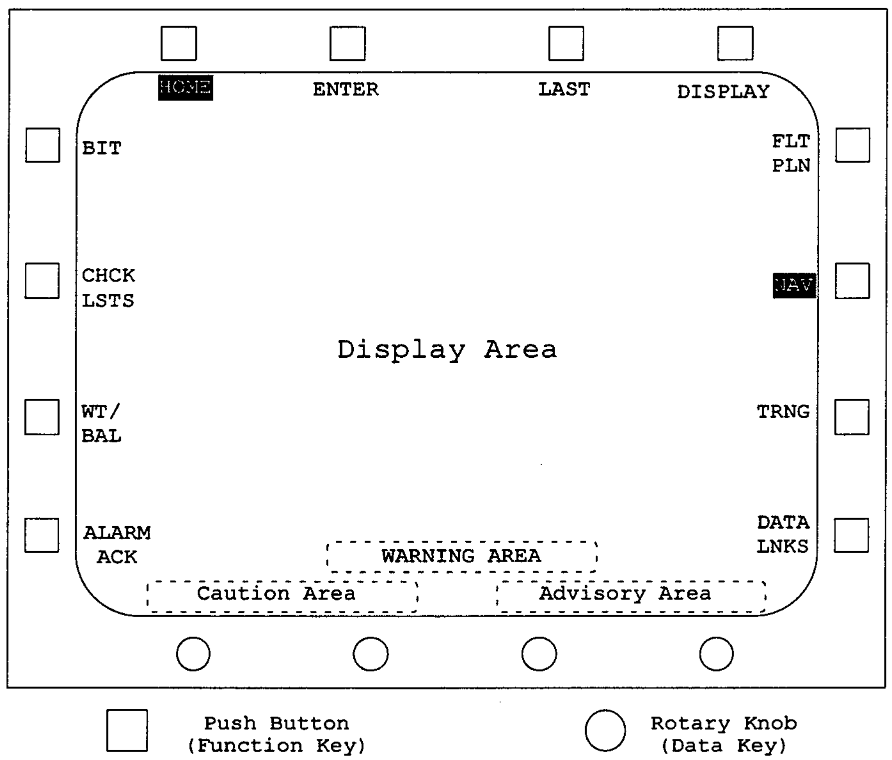


Figure 25. HDD General Layout

These assumptions were made to facilitate development of the HDD in the EFS. That is, until a prototype HDD is procured, the HDD can be simulated by using standard personal computer (PC) hardware. For example, the HDD display area can be simulated by using a standard PC monitor. The HDD functions keys can be simulated by mapping each to the function keys (labeled F1 through F12) found on most PC keyboards. Finally, the HDD data keys can be simulated by simply typing in the equivalent information using the PC keyboard.

Home Mode. By pressing the *HOME* function key, the pilot calls the "top-level" menu available, without changing the information shown in the display area. This permits the pilot to rapidly select any other HDD mode, and facilitates how the pilot "navigates" through the various HDD modes and menus. Consequently, the *HOME* function key is depicted in the same location for all HDD modes. The same is true for the *ALARM ACK* key, which the pilot uses to acknowledge any WCA, and the *ENTER* key, which the pilot uses to accept information into ASTRA once entered by the data keys.

As with all function keys, selection of the *HOME* key is indicated by the "reverse video" over its display label, as shown in Figure 25. (Note in the figure that the display label *NAV* also has reverse video. This indicates that the *Navigation* mode information is being displayed in the HDD. To see the appropriate *Navigation* display labels once again, the pilot need only to select the *NAV* or the *LAST* function key.)

Built-In-Test (*BIT*) Mode. A system BIT, which is required by the FAA [12] for flight certification, is used to verify the overall functionality of ASTRA and each of its subsystems (not just the HDD). While a system BIT should be automatically initiated whenever the

ASTRA is powered up, the pilot should be able to initiate a system or subsystem BIT at anytime.

The BIT should give a clear indication to the pilot that the system has successfully passed the test. If any part of the BIT fails, then appropriate error codes should be generated so that the pilot might take corrective action. The error codes should also give the pilot some indication as to what restrictions, if any, exist should the error go uncorrected.

The following items should be included as part of the ASTRA BIT:

- Displays: check functionality of HUD and HDD (to include display color, brightness, and contrast); check status of symbology fields and alphanumerics (to include the flight director, digital fields, and alarms).
- Check communication links between each of the ASTRA subsystems (HDD, HUD, GPS, data bases, etc.).
- Verify functionality of HDD *function* and *data* keys.
- Check status of GPS constellation.
- Verify integrity of ASTRA software and remaining hardware configuration.

Figure 26 shows how the HDD *BIT* mode might be configured and displayed. In addition to the information detailed above, the figure shows the amount of time remaining on the BIT. The pilot may also use the *SEQ UP* and *SEQ DN* function keys to select BIT for a specific subsystem, which is indicated with *reverse video* after selection. By pressing the *DISPLAY*

FAULTS function key, the pilot can view specific fault status/codes for the selected subsystem.

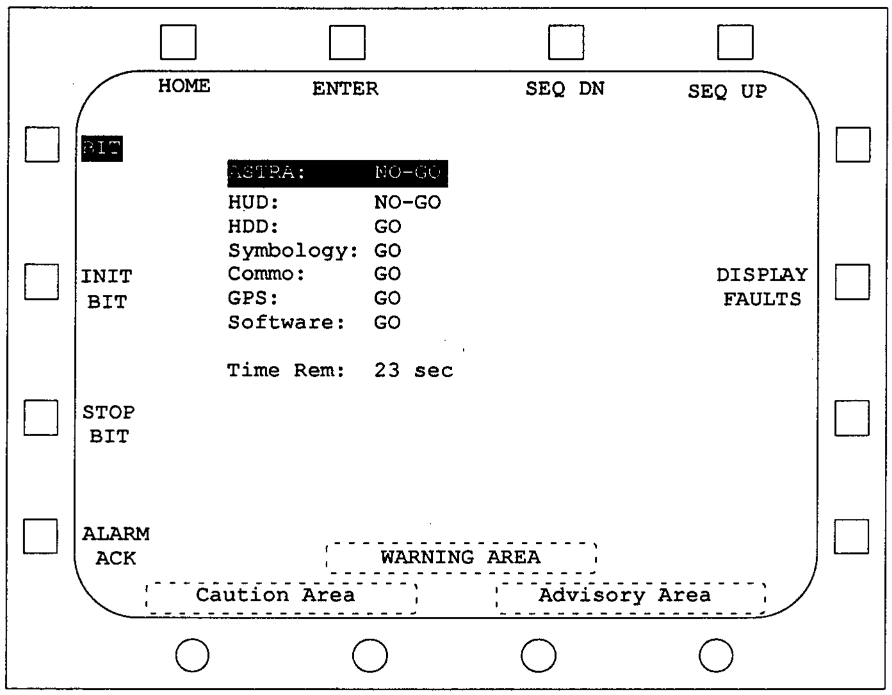


Figure 26. HDD *Built-In-Test* (*BIT*) Symbology Display Set

Check Lists (*CHCK LSTS*) Mode. This mode permits the pilot to view the aircraft Normal Operating Procedures, Emergency Operating Procedures, and aircraft operating limits. The operating procedures depicted in the HDD correspond to the "Abbreviated" Operating procedures detailed in the Commander-700 operator's manual [21]. The operating limits depicted in the HDD might include a summary of critical data (such as airspeed and power plant limitations), also detailed throughout [21].

Figure 27 shows an example of how the HDD might be configured for "Normal Operating Procedures." The left side of the display provides a sequential list of all checklist menus, while the right side details the appropriate procedures for a selected menu item, (which is indicated through the use of arrows and reverse video on the left side of the display). Emergency Operating Procedures could be configured and displayed similarly. Function keys with reverse video indicate which type of procedure (*NORMAL* or *EMERG*) is being displayed.

When the pilot selects the HDD *CHCK LSTS* mode, AMS could be applied to automatically select the appropriate checklist corresponding with the current flight mode. For example, if the FMI infers the flight mode to be *cruise*, then the *CHCK LSTS* mode should display the normal checklist procedures detailed under "Cruise," rather than those for "Before Starting Engines." By using the *SEQ DN* and *SEQ UP* function keys, however, the pilot can manually select any checklist procedure for viewing. Similarly, should the PA detect an aircraft emergency (such as an engine failure), AMS could be used to automatically select and display the appropriate emergency procedure. This would permit the pilot to maintain positive control of the aircraft throughout the emergency.

A final word on checklists: it is important to note that the FAA [12] requires electronically displayed checklists to be certified for flight, just as those published in the aircraft operator's manual.

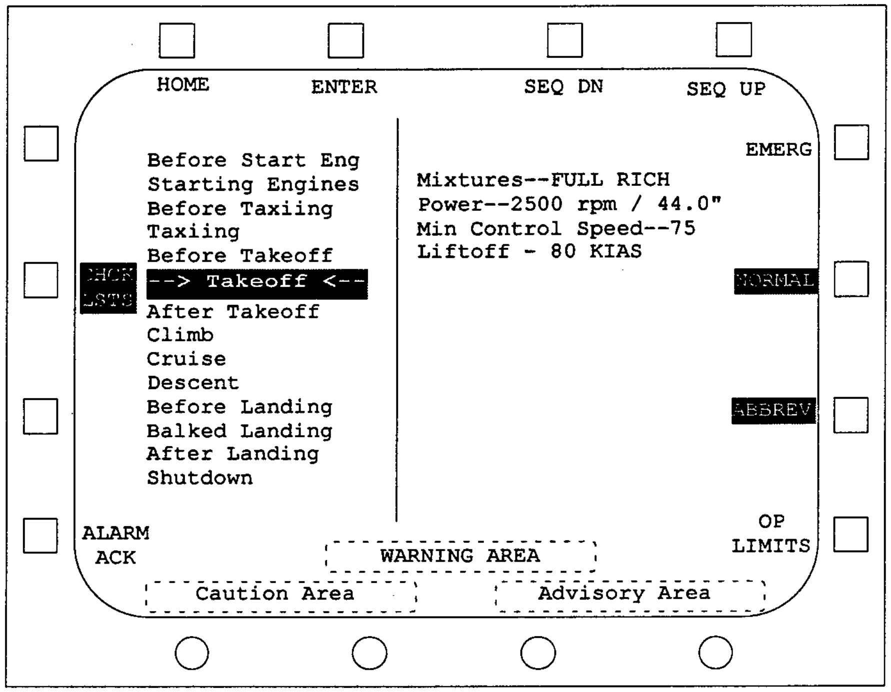


Figure 27. HDD *Check Lists* (*CHCK LSTS*) Symbology Display Set

Displays Mode. This mode allows the pilot to manually select a display mode, adjust display parameters (such as brightness or contrast), or customize display settings . Note that this mode is used to control both the HDD and the HUD. For example, the pilot may wish to override the HUD mode selected by AMS, and view another instead. Customizing a display might entail allowing the pilot to add/remove certain symbology items to the setting, or choosing between a number of symbol types. Figure 28 shows an example of how the pilot could adjust the brightness and contrast settings for the HUD.

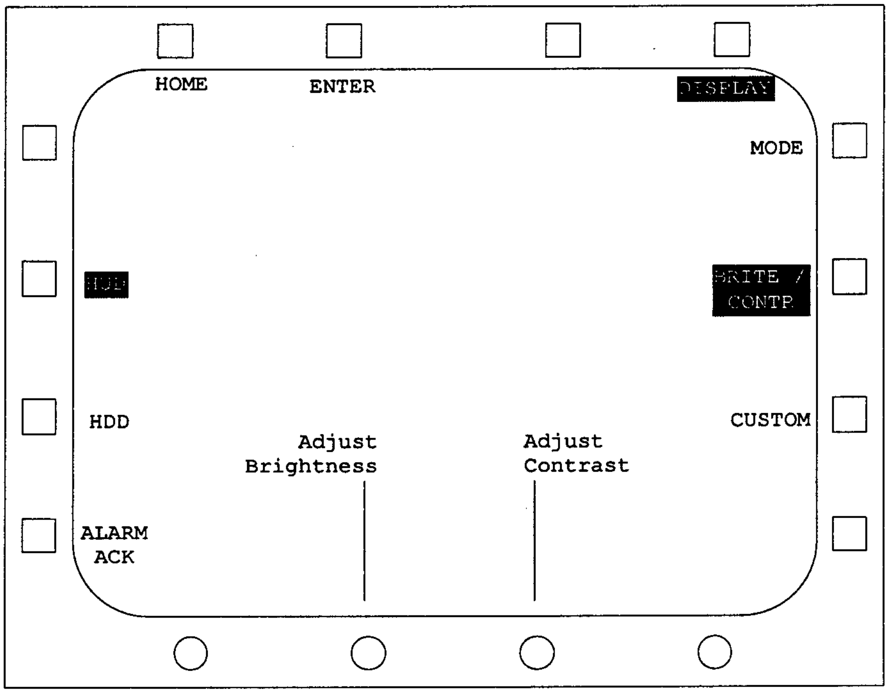


Figure 28. HDD *Displays* Symbology Display Set

Flight Planning (*FLT PLN*) Mode. This HDD mode allows the pilot to enter a detailed flight plan into the "ASTRA Flight Plan data base." Consequently, the pilot uses this mode to conduct any flight planning, as described in earlier chapters. Once the flight plan has been entered, the pilot "activates" it through the *Navigate* mode, described in the next section. Figure 29 illustrates how this mode might display a typical flight plan.

The *FLT PLN* mode makes use of a standardized display format, permitting the pilot to quickly enter data in a logical order (generally corresponding with an FAA flight plan). However, this flight plan contains much more detail than typically contained in an FAA flight plan, in order to provide adequate information to the Navigation Module and FMI for flight mode inference (as described in earlier chapters).

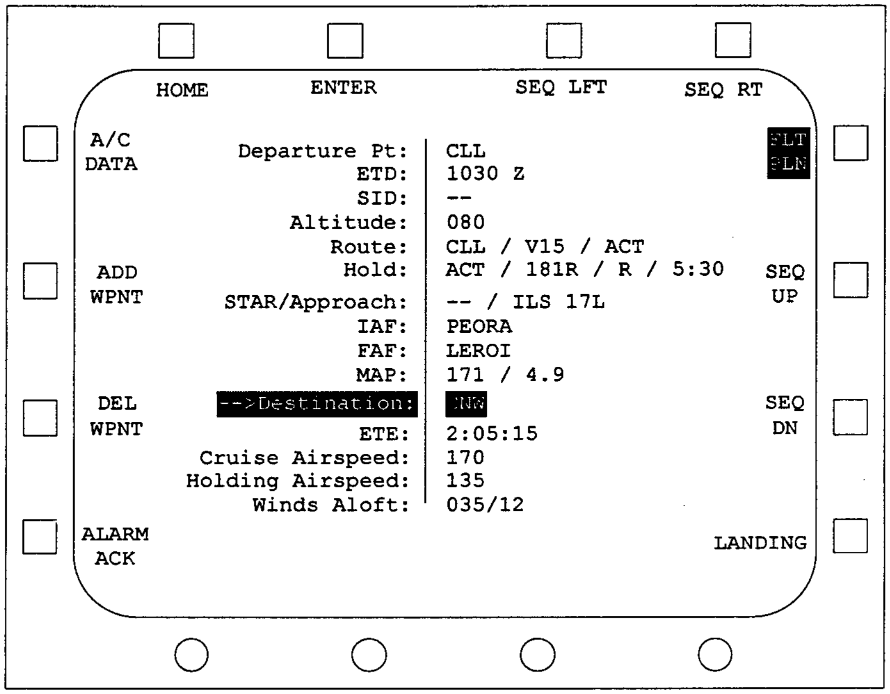


Figure 29. HDD *Flight Planning* (*FLT PLN*) Display Set

As with the *CHCK LSTS* mode, the *FLT PLN* mode divides the display is in half. The left side of the display shows the *flight plan fields*, which the pilot may edit, such as "Departure Point," "Altitude," and "Route." The pilot selects these fields by using the *SEQ DN* or *SEQ UP* function keys. To the right of each flight plan field is a corresponding *data field*, which details specific information the pilot enters via the data keys. For example, Figure 29 shows the data field "CLL" (College Station), which corresponds to the "Departure Point" flight plan field. Different items within a data field are selected by using the *SEQ RT* or *SEQ LFT* function keys.

The pilot may add or delete waypoints from the flight plan by using the *ADD WPNT* or *DEL WPNT* function keys, respectively. The pilot can further use the *LANDING* function key to inform ASTRA that he wishes to execute several approaches/landings to the destination. Once each of the data fields have been filled or edited, the pilot must select the *ENTER* function key to "accept" the information into the data base.

Note that not every item must be manually entered by the pilot into the flight plan. In other words, information from the Aircraft Data Base could be used to provide many default settings. For example, the airspeeds associated with "Cruise" or "Holding" could automatically be loaded into their data fields. Of course, the pilot can manually override these selections by entering another value. By selecting the *A/C DATA* function key, the pilot can also view the parameters listed in the Aircraft Data Base.

Finally, it is important to recall that the use of this mode makes the same assumptions about navigation stated in earlier chapters—namely, that GPS and a functioning Navigation Module have been integrated into the ASTRA system architecture.

Navigation (*NAV*) Mode. As alluded to in the previous section, the *NAV* mode assists the pilot in performing real-time navigation. Because of this, it is anticipated that the pilot will use this mode more extensively than any other (at least while airborne), meaning the pilot must have a rapid means of entering an ATC clearance (such as a vector).

Figure 30 depicts a typical display with the HDD *NAV* mode selected. The basic display is incorporated as a Horizontal Situation Indicator (HSI), but with additional information included to enhance the pilot's situational awareness. For example, specific flight plan data is depicted graphically in the form of waypoints and courselines, once the *ACTIVATE FLT PLN* function key has been selected. By using the *MAP ON/OFF* function key, this graphic representation can further be superimposed over a "moving map display," allowing to the pilot see the flight plan plotted over a digitized aeronautical chart. Navigation aids, fixes, special use airspace, and airfields are also plotted on this display, along with their identifiers and other pertinent data.

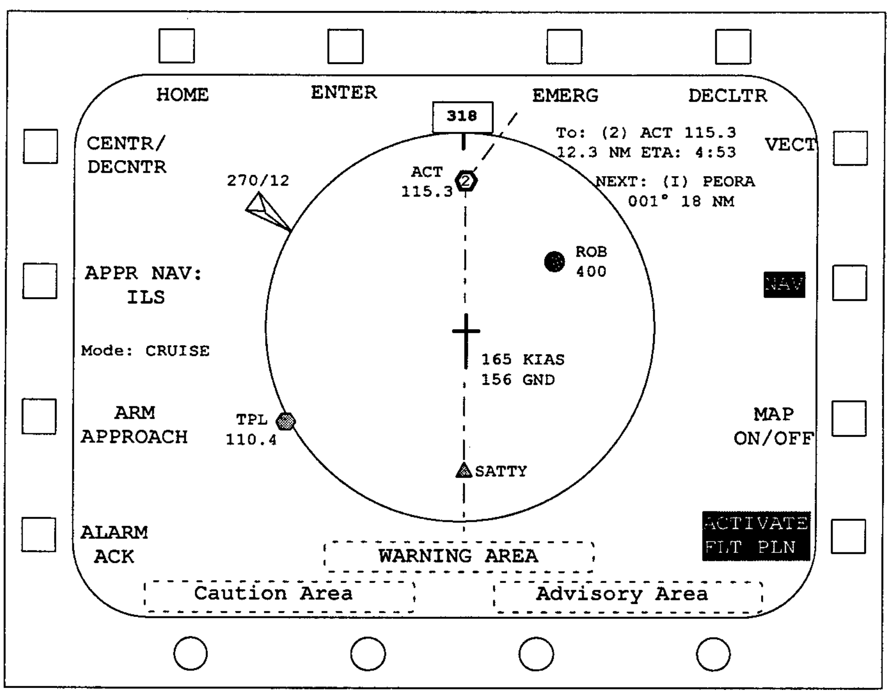


Figure 30. HDD *Navigation* (*NAV*) Display Set

Flight plan information, detailed in the margin of the display, includes waypoint name and number, course, distance, estimated time of arrival, and subsequent waypoint information.

Selection of the *APPR NAV* function key allows the pilot to change the NAVAID that will be used to execute the instrument approach. Selection of the *ARM APPROACH* function key couples the FD to this NAVAID, rather than GPS. Winds aloft data (wind speed and direction) is depicted graphically (in the form of a tetrahedron), allowing the pilot to immediately assess its effect on each flight plan leg. Miscellaneous data depicted on the display includes the inferred flight modem, aircraft ground speed, and any active WCA's.

The *CENTR/DECNTR* function key allows the pilot to change the aircraft present position from the center of the display to the bottom of the display. Selecting the *EMERG* function key display pertinent navigation information to the nearest airfield, permitting the pilot to immediately react to emergency situations. The *DECLTR* function key allows the pilot to optimize the amount of display symbology (such as NAVAIDS and airfields) to be included in the *NAV* mode.

Selecting the *VECT* function key permits the pilot to enter vectors and clearances that alter any data previously entered via the *FLT PLN* mode. For example ATC might issue *vectors* to an instrument approach, or might amend previously issued clearances (such as to *hold* at a NAVAID, to increase/reduce the indicated airspeed, or to expect an alternate instrument approach). Figure 31 proposes a HDD configuration when the pilot selects the *VECT* function key. The appearance and functionality of the *VECT* mode is very similar to that for the *FLT PLN* mode, but with modifications structured to anticipate most ATC clearances.

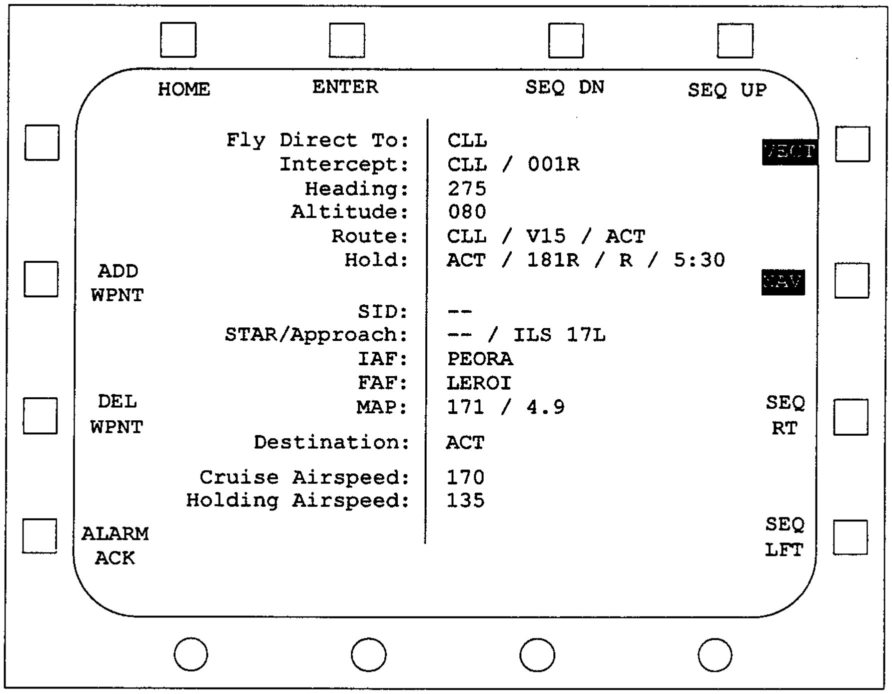


Figure 31. HDD *Vector* (*VECT*) Display Submode to *Navigation* Mode

Training (*TRNG*) Mode. This mode assists the pilot in training and evaluation by making use of the FMI flight mode inference. The pilot may take advantage of this mode in one of two ways, *active* and *passive*. When used actively, the *TRNG* mode displays real-time training information to the pilot, consummate with the inferred flight mode. For example, when the FMI infers the aircraft to be in the *cruise* mode, the HDD might display prompts reminding the pilot to check the fuel status; likewise, it might recommend a power setting and airspeed corresponding to maximum endurance or maximum range conditions. When used in this manner, the *TRNG* mode is considered *active* because the HDD is displaying training information, precluding its use for other HDD modes, such as *Navigation*.

When used passively, the *TRNG* mode allows the pilot to "record" the flight into a data file. He could then conduct a post-flight review of his flight by using a variation of the GAPATS graphical user interface or by using the GAPATS/FMI toolbox, both of which described in earlier chapters. Used in this manner, the pilot can use the HDD for any of the other modes discussed in this chapter, while retaining the ability to conduct a thorough evaluation of his flight afterwards. Of course, the pilot could use the *TRNG* mode actively and record flight data at the same time.

Figure 32 depicts an example of how the *TRNG* mode might appear. The pilot activates this mode by selecting the *TRNG* function key. To record data, the pilot need only to press the *RECORD ON/OFF* function key. The corresponding label changes to reverse video when data is being recorded. The display includes fields for FMI inference (mode, certainty, and confidence); navigation data (next waypoint, winds aloft, current track/heading, and subsequent track/heading); aircraft configuration (landing gear and flaps); and miscellaneous information (fuel status, NAVAID frequency change time, and descent point data).

Weight and Balance (*WT/BAL*) Mode. The *WT/BAL* mode is the counterpart to the *FLT PLN* mode, in that it assists the pilot in mission planning. While it might be possible to incorporate each mode into a single "Mission Planning Mode," maintaining two distinct display modes offers two advantages. First, it facilitates the design of the each display set, since the information managed by each are generally unrelated. Second, it provides the pilot a simpler display architecture to use, since integrating each into a single mode would likely require two or more submodes.

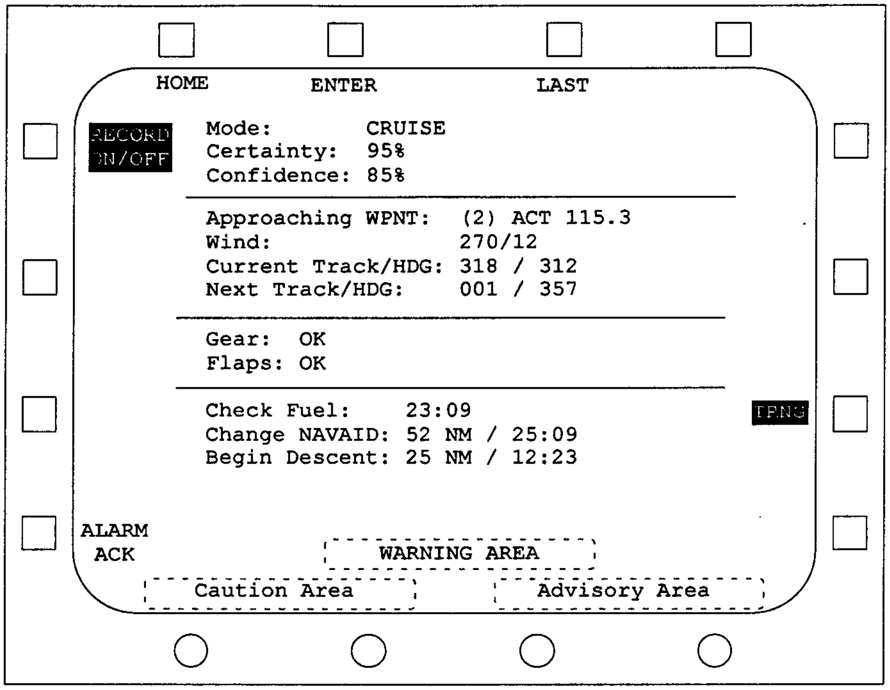


Figure 32. HDD *Training* (*TRNG*) Display Set

Selecting the *WT/BAL* function key, permits the pilot to quickly verify and update the weight and balance status for the aircraft, as seen in Figure 33. The mode retrieves basic aircraft information (such as empty aircraft weight/moment) from the Aircraft Data Base, and displays data fields which the may pilot edit prior to each flight. These data fields permit the pilot to calculate the aircraft weight for each mission, by including data for pilots, passengers, cargo, and fuel. After editing a data field, the pilot must select the *ENTER* function key, which will accept the data and update the appropriate calculations. Consequently, this mode will permit the pilot find how much fuel and cargo he can carry for a particular flight.

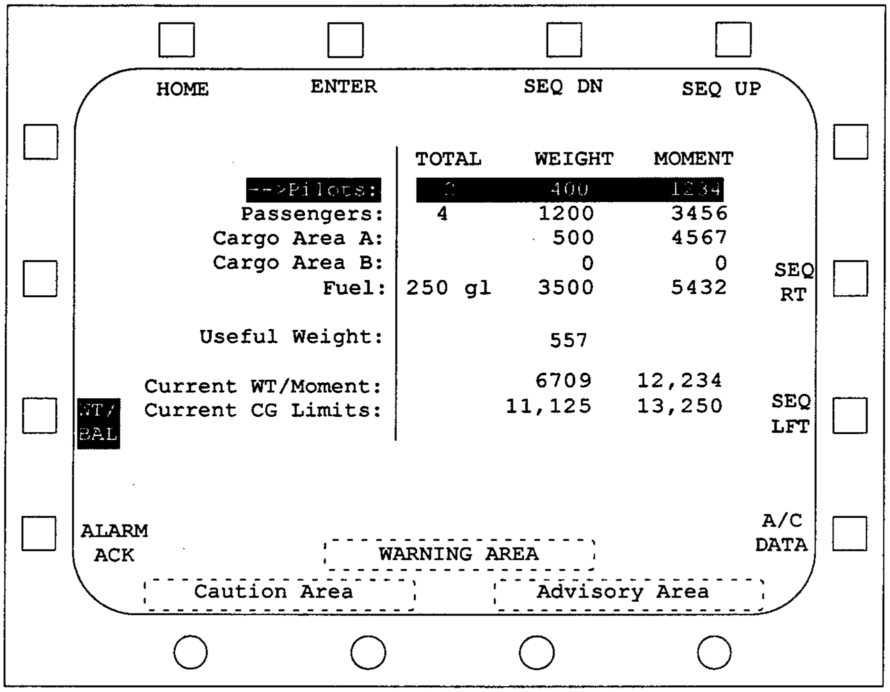


Figure 33. HDD *Weight and Balance* (*WT/BAL*) Display Set

If the system calculates that any weight or moment limitation is not within acceptable limits, an immediate indication must be made to the pilot. For example, if the aircraft gross weight exceeds that permitted for takeoff, the HDD should display an error message indicating how much weight must be removed. Furthermore, the system should not accept invalid data inputs from the pilot. For example, if the pilot attempted to enter a value of 10,000 gallons in the fuel data field, the HDD should display an error message indicating the mistake. By selecting the *A/C DATA* function key, the pilot can review the pertinent weight and balance data contained in the Aircraft Data Base.

Data Links (*DATA LNKS*) Mode. This mode has been included in the HDD display architecture to provide for future system requirements. Consequently, the ASTRA will become even more powerful—further enhancing situational awareness and reducing pilot workload—as new technologies are integrated into the system. For example, selecting the *DATA LNKS* function key might allow ASTRA to receive critical flight information via a digital data burst. For example, ATC could transmit clearance information, weather advisories, and traffic information in this manner. Likewise, aircraft collision avoidance systems could transmit information regarding a "near miss." When used in this manner, ASTRA could display a number of WCA's, as well as appropriate pilot action, on both the HUD and HDD. Because these technologies are not yet mature, no diagram is depicted for this display mode.

To summarize, the two ASTRA flight displays provide the critical linkage for the pilot to interact with this smart cockpit system. The HUD is a primary flight display which can generate several display sets, each of which assist in pilotage and navigation. By applying the concept of Automatic Mode Switching (as a function of FMI mode inference), the HUD selects that display set, without the need for pilot input, appropriate for the current flight mode. The HDD, on the other hand, is a secondary flight display which is used for navigation and mission planning. It is configured as an MFD in order to provide the pilot with the ability to view and manipulate data. Despite being called a "secondary" flight display, it is nevertheless essential in allowing the pilot to communicate his mission planning requirements to ASTRA. Consequently, the display architecture for the HDD was specifically designed from the perspective of the general aviation pilot.

EVALUATING ASTRA

"Do not put too much confidence in experimental results until they have been confirmed by theory."

—Sir Arthur Eddington

The Role of Evaluation in System Development

A thorough system evaluation of ASTRA will play a critical role in its development, especially as the system matures from prototype hardware/software to a candidate for commercialization. Such an evaluation, which may be done incrementally throughout the system development process, is important for a number of reasons. First, and perhaps most obviously, a performance evaluation will verify the functionality of individual subsystems, particularly as existing modules are modified or new modules are added to ASTRA. In short, each subsystem must operate such that it will satisfy the flight certification requirements of the Federal Aviation Administration (FAA) [6], [7], [12]. These documents describe, for example, such issues as system reliability, system failures and failure modes, software integration, and electromagnetic interference. Anderson [3] describes these airworthiness issues, in addition to several others, in an evaluation he conducted for one general aviation HUD of much less complexity than that seen in ASTRA.

An incremental evaluation is furthermore essential in validating and optimizing how the pilot interfaces with the system. As was detailed in earlier chapters, such pilot interaction with ASTRA occurs through the Head-Up Display (HUD) and Head-Down Display (HDD). Consequently, it is necessary to evaluate how the pilot interprets and reacts to individual HUD/HDD symbology display sets, and how he enters data into the system. In short then, the goal of such evaluation focuses on minimizing pilot workload requirements, while optimizing his situational awareness. The importance of this task cannot be overstated—if the pilot finds ASTRA to be a difficult system to use and operate, then he will not use it, even if the FAA has certified it for flight. Consequently, a total system evaluation will ensure that ASTRA remains "on track" for commercialization.

Such an incremental performance evaluation can be readily accomplished in the Engineering Flight Simulator (EFS). This is especially desirable since the EFS is an integral part of the system software development environment. Furthermore, by conducting a detailed performance evaluation in the EFS, it is possible that ASTRA will be mature and robust enough to install in the Commander-700 with only minor modifications. In other words, the EFS provides a tremendous opportunity to thoroughly evaluate ASTRA and each of its subsystems before they are integrated for flight in the actual aircraft.

Using the EFS in the evaluation of ASTRA provides one additional advantage to the developer. Namely, flight data taken in the EFS can be "recorded" for immediate data analysis <u>and</u> for future analysis of subsequent system modifications. These analyses can then be compared to evaluate whether or not system performance improved, *based on the* *same data set.* An example using ASTRA's Flight Mode Interpreter (FMI) illustrates this point. Suppose an EFS flight has been made to evaluate the FMI fuzzy rule base, and the immediate data analysis shows that the rule base requires "fine tuning" to improve its robustness. After modifying the rule base, the same EFS data is then used to show whether the FMI performance has indeed improved. Likewise, as new subsystems are developed for integration into ASTRA, these data can be used yet again in their evaluation (This assumes, of course, that the data sets contain the appropriate information needed for the new subsystems).

Evaluation Philosophy

The ASTRA system architecture presented in this thesis was developed from the perspective of the General Aviation (GA) pilot. Consequently, the evaluation philosophy described in subsequent sections focuses on a *pilot evaluation* of ASTRA, in terms of performing various mission maneuvers. That is, it is possible to simultaneously evaluate ASTRA system performance and the associated pilot interface issues by considering those tasks the GA pilot routinely performs. Such pilot evaluation must concentrate on ensuring that the FMI correctly infers the aircraft flight mode, since this inference determines not only which display configuration will be presented (as described in Chapter 5), but also how the Pilot Advisor (PA) classifies and formats any potential alarms for display (as described in Appendix D). Consequently, the first step in conducting a pilot evaluation of ASTRA is to identify representative *mission tasks* which correspond to each of the FMI flight modes.

Identifying and defining such mission tasks must consider a number of points. First, each task must be *representative* of those seen in the general aviation environment. Specifically, the tasks should correspond to the flight maneuvers for which ASTRA was designed to recognize, such as those associated with instrument approaches. Consequently, it would be appropriate to define tasks such as *final approach* or *landing*, whereas it would be entirely inappropriate to define tasks such as *barrel roll* or *loop*.

Additionally, the mission tasks should be *repeatable*, characterized by detailed *standards* for flying the task, but without any specific *procedures* for flying the task. That is, each task should be defined such that it is *independent* of pilot style or technique. Assumed here, of course, is that the use of any one pilot technique remains within "acceptable" limits for how the aircraft is operated (in terms of both the aircraft operating limits/procedures and standard Air Traffic Control (ATC) procedures).

Finally, the mission tasks should consider both *nominal* and *non-nominal* conditions, as defined in previous chapters. Considering both types of flight conditions is required for several reasons. Evaluation under nominal conditions is first necessary to ensure that the FMI operates properly throughout the entire operational flight envelope of the aircraft. Once the FMI is known to be correctly inferring the flight mode under nominal conditions, these same mission tasks can then be used to evaluate how well the pilot interprets and reacts to the various flight displays which ASTRA automatically presents. (Recall that ASTRA applies Automatic Mode Switching to select the appropriate display mode.)

By also considering non-nominal conditions, such as incorrect landing gear or flap settings, the FMI can be further evaluated for its robustness. That is, it is necessary to verify that the FMI continues to infer the correct flight mode in spite of the non-nominal conditions. For example, an approach may be executed by using no flaps, which is considered a non-nominal flight condition corresponding to landing with a cross wind condition. It is important to note that in flying these mission tasks under non-nominal conditions, it is implicit that the PA must generate some type of alarm. Consequently, these same non-nominal conditions can be used to evaluate how the pilot reacts to alarms.

In short then, properly identifying and defining appropriate mission tasks will ensure that recorded data is consistent between different pilots, in varying flight environments (such as the aircraft and the EFS), and ultimately, under different actual flight conditions (due to aircraft loading, traffic, or weather).

Mission Tasks for Evaluating ASTRA

The evaluation mission tasks detailed below correspond one-for-one with the flight modes which ASTRA can interpret. The numerical specifications in the task descriptions are for the Commander-700, which is the test aircraft. They are defined in terms of the guidelines described in the preceding paragraphs and in terms of the procedures detailed in the operator's manual [21], with one notable exception. Specifically, the tasks are designed for use in the EFS, which has only one set of levers which effect *power*, rather than the three sets of levers found in the aircraft (corresponding with *throttles*, *propellers*, and *mixture*). Consequently, references to these latter controls are not made in subsequent task definition, and flying the same tasks in the aircraft will require but slight modification. Consideration is also given as to how the tasks may be safely conducted in terms of *non-nominal* flight conditions, especially in terms of common piloting errors. Unless otherwise noted in the mission task definitions below, airspeed should be maintained within ±5 knots and altitude within ±100 feet.

Taxi. In general, the pilot should *taxi* the aircraft at a speed no greater than a "brisk walk." Consequently, just enough power should be applied to maintain this speed, with corresponding changes in power being smooth (that is, not too abrupt or rapid).

The aircraft flaps should nominally be retracted during taxi. However, the pilot may wish to set the flaps at the Takeoff (T/O) position, in anticipation of his departure. Consequently, the evaluation must include taxiing the aircraft with the flaps extended to all possible settings.

Takeoff. *Takeoff* may occur immediately following either *taxi* or *landing*. Consequently, the pilot may begin his takeoff roll with the aircraft at rest (as in a "full stop") or with the aircraft already moving (as in a "touch and go"). Regardless of how the task is initiated, maximum continuous power should be applied throughout the maneuver. The aircraft should be rotated at or above 80 knots. This airspeed is maintained until 50 feet above ground level (AGL), at which point, the aircraft should be smoothly accelerated to 85 knots, then to best rate-of-climb speed or best angle-of-climb speed.

Nominally, the landing gear should be retracted when no remaining runway exists for an emergency landing. In general, this will occur at or above 200' AGL. The flaps should be retracted as soon as possible after the gear are retracted and obstacles cleared. To evaluate the effect of these parameters under non-nominal conditions, they may be retracted in reverse sequence, and at varying altitudes.

Climbout. *Climbout* normally begins immediately after the pilot has retracted the gear and flaps following *takeoff*, and the aircraft is continuing to climb. For a "cruise climb," airspeed should be maintained between 120 and 140 knots, with power set at slightly less than maximum. For a "maximum continuous power climb," airspeed should be maintained at 120 knots, with power set at maximum. These airspeed ranges could be varied by as much as ±10 knots when evaluating for non-nominal airspeed conditions.

Nominally, the aircraft is flown in a "clean" configuration (landing gear and flaps retracted) during *climbout*. Consequently, the gear and/or flaps may be extended (taking care that no aircraft limitations are exceeded) to evaluate for non-nominal aircraft configuration conditions.

Cruise. *Cruise* will typically begin immediately following *climbout*, although it may follow any intermediate climb or descent in response to an ATC clearance. Nominally the airspeed for *cruise* will lie between that for maximum range and that for maximum endurance, with the power set appropriately. To evaluate for non-nominal effects, this airspeed range should be extended to include an approach speed (minimum value) and the maximum continuous airspeed (maximum value).

Altitude should be held constant during *cruise*. Typical altitudes for this task under Instrument Flight Rules (IFR) will generally be greater than 2,000' AGL. Under Visual Flight Rules (VFR), however, cruise altitudes may be as low as 600' AGL, as when flying traffic patterns about an airfield [2]. Consequently, evaluation of *cruise* must occur from 600' AGL upward. Corrections in altitude should be made using vertical speeds of no more than ±300 feet per minute (fpm), which may be increased to ±500 fpm to evaluate the effects of vertical speed under non-nominal conditions.

As with *climbout*, the aircraft is nominally flown in a "clean" configuration for *cruise*. Consequently, the gear and/or flaps may be extended (again taking care that no aircraft limitations are exceeded) to evaluate for non-nominal aircraft configuration conditions.

Initial Approach. *Initial approach* will, in general, follow the *cruise* mode. When flying on an IFR flight plan, *initial approach* begins upon passing the Initial Approach Fix (IAF). Otherwise, the mode begins as the pilot prepares the aircraft for landing. Because of these factors, *initial approach* will generally be flown at lower airspeeds than *cruise*, which allows the pilot to reconfigure the aircraft for landing (that is, extend the landing gear and/or flaps). Consequently, this mode should be evaluated at airspeeds up to those used in *cruise*, simulating that the aircraft is on *initial approach*, but that the pilot has overlooked the need to slow the aircraft for reconfiguration.

The altitude range for *initial approach* corresponds with those from which the *cruise* mode may be flown. Consequently, this mode should be evaluated from 600' AGL and upwards. It is also important to note that the aircraft may or may not be descending during *initial approach*. Subsequently, this parameter should be varied within the range -2000 to +300 fpm.

As previously noted, the pilot typically reconfigures the aircraft for landing during *initial approach*, although there is no requirement to do so. Consequently, the gear and/or flaps may be extended (taking care that no aircraft limitations are exceeded) at varying altitudes and airspeeds, and in differing order, to evaluate the effects of non-nominal aircraft configuration conditions.

Final Approach. *Final approach* is very similar to *initial approach*, except that the aircraft should be descending, aligned with the runway heading, and configured for *landing*. Under IFR, *final approach* begins upon passage of the Final Approach Fix (FAF). Consequently, the evaluation for final approach is similar to the previous mode, but with correspondingly lower airspeeds and altitudes.

Airspeeds should not exceed the maximum airspeed limits with the landing gear and/or flaps extended (even if a "no-flap" landing is intended). Minimum airspeeds should be about five knots above the aircraft stall speed. Variations in vertical speed will be similar to those in *initial approach.*

Landing. This mode logically follows that of *final approach*, and begins at that point where the pilot transitions from instrument flight (controlling the aircraft through HUD flight symbology) to visual flight (controlling the aircraft by referencing the runway environment) in order to safely land the aircraft. Consequently, this point will be at or above 200' AGL (which corresponds with "Decision Height" for most ILS approaches). Once again, the maximum airspeed should correspond with aircraft configuration limits, while the minimum airspeed should be limited to five knots above the aircraft stall speed.

The aircraft should nominally be configured to land by the time this mode is reached—the pilot should now be concentrating entirely on controlling the aircraft. As with previous flight modes, non-nominal conditions could include extending the flaps and/or gear at various times and in differing order.

Evaluating the Flight Mode Interpreter

As detailed in earlier chapters, a great deal of effort has already been conducted in improving the robustness of the FMI. In particular, Kelly's GAPATS/FMI Toolbox [14] has been exceptionally helpful to this end, because it allows for the rapid development of prototype membership functions. The mode inference of these prototype functions can be readily generated for evaluation and comparison with the output of previously defined fuzzy rules. As might be expected, the toolbox will have to incorporate additional data, such as the *distance* parameter generated by the Navigation Module.

It is important to note that evaluating the FMI with the GAPATS/FMI Toolbox requires the pilot to indicate when he is transitioning from one flight mode to the next. This "truth" data is then plotted together with the FMI decision output, yielding a rapid means of readily comparing what mode the pilot "says" the aircraft is in, with the mode the FMI "thinks" it is in. This is consistent with the way in which the mission tasks were defined—in enough detail to describe each flight mode but without mandating a particular flying technique.

Finally, it is important to note that a complete evaluation of the FMI does not end here. Because FMI decisions are used to determine which display modes are presented (through Automatic Mode Switching), a total FMI evaluation can only be completed with that of the ASTRA flight displays, as discussed in the next section. For if an inappropriate display is presented to the pilot during any phase of flight, it can only be the result of incorrect or nervous FMI output.

Evaluating the ASTRA Flight Displays

As alluded to in previous sections, subjective pilot ratings will play an important role in the evaluation of the ASTRA flight displays. Historically, many such subjective ratings have been patterned in the form of the Cooper-Harper Pilot Rating [4], which uses a "decision tree" to assist the pilot in making his rating. That is, the logic tree guides the pilot, by posing a series of questions, to help expose what problems exist with the display, especially in terms of pilot workload. The use of subjective ratings is a subtle, yet important concept—after all, a pilot may be able to perform a mission task extremely well, but only at the expense of excessive pilot workload. In fact, it may be possible to significantly reduce the pilot workload, with only a minor reduction in task performance.

Newman and Haworth [11] present two such rating scales, which were specifically designed for the evaluation of flight displays. Their first rating scale is used to evaluate the readability of display parameters, whereas the second is used to evaluate the adequacy of display parameter dynamics. Taken together, the scales provide an excellent means of evaluating ASTRA's flight displays. It is important to emphasize that both scales require the use of selected mission tasks to evaluate the display. The mission tasks presented earlier in this chapter meet this requirement.

One advantage in using rating scales such as those proposed by Newman and Haworth is that they generally produce very consistent results. This is particularly true when trained evaluators, such as engineering test pilots, conduct the evaluation. With novice evaluators, however, more time is often required in learning the use of the logic tree, and results may be less consistent. Because there will likely be an extremely limited number of trained evaluators available during ASTRA development, the following questions can be used to assist the evaluator in clearly articulating what problems exist. These questions, help the evaluation pilot to focus on potential display problems—especially in terms of situational awareness and pilot workload.

- Were any of the symbology sets ambiguous or confusing?
- Were the symbology sets simple to use and easy to learn?
- Were any of the display sets cluttered? If so, which information would you remove?
- Were any of the display sets missing information (for example, *Power*) that you would like to see?
- Did you find the information displayed on each HUD display set applicable to the flight mode?
- Did Automatic Mode Switching provide display changes at appropriate times? Was the presentation of new modes premature or excessively late? Did the HUD demonstrate any "nervousness," by switching back and forth between several display sets?

- Did you dislike any of the design formats for individual symbology pieces? How you like to see them presented? (Analog vs. Digital, Linear vs. Circular, etc.)
- Were the Flight Director commands reasonable to follow, or did it seem "nervous?" Was maintaining such parameters as *airspeed*, *heading*, and *altitude* easy or hard to do?
- Was rate or trend information acceptable, particularly with such parameters as *airspeed* and *altitude*?
- Did you notice any tendency to "fixate" on a particular piece of symbology, such as the FD or any alarms?
- Did you have any tendency to become disoriented? Could you recognize and recover from an unusual attitude with this symbology?
- Were the displays legible and readable under all lighting conditions?
- Was the timeliness in display and format of *alarms* appropriate? Did they cause you to react in an appropriate manner (e.g., to correct the condition without excessive delay)?
- Were any false alarms generated that should not have been? Were any alarms *missing* that should have been presented? Were there any "nervous" or otherwise annoying alarms?
- Was the pilot workload associated with entering/changing data excessive or confusing?

Conducting a total system evaluation of ASTRA will no doubt be a time consuming, demanding effort. This evaluation will require the concerted effort of many individuals, especially as subsystems are modified, or new subsystems are added to the ASTRA architecture. The mission tasks described in this chapter will provide an excellent means of providing a subjective pilot evaluation, especially in terms of situational awareness and pilot workload. They will also provide a good "springboard" from which detailed test plans and test procedures may be generated.

One final note on evaluating ASTRA. While a great deal of the evaluation will take place in the EFS, additional evaluation must take place in the aircraft itself. This latter evaluation must not be taken lightly, for a total system evaluation can only take place in the actual mission environment. Consequently, ASTRA must be subjected to the real world demands that the general aviation pilot routinely faces.

CONCLUSIONS AND RECOMMENDATIONS

"Men occasionally stumble over the truth, but most of them pick themselves up and hurry off as if nothing had happened."

—Winston Churchill

Summary of the ASTRA Research Project

After describing the benefits of introducing "smart cockpit" technology into the General Aviation (GA) community, this thesis proposed a functional specification for one such system, ASTRA. Its functional specification describes—from the perspective of the GA pilot—those functions that a pilot advisory system must perform to increase safety and enhance situational awareness, without undue pilot workload. To this end, the specification augmented the system architecture—in terms of hardware and software—defined previously during the GAPATS research effort.

Two new concepts were introduced in the system specification, which greatly facilitated addressing the issues associated with how the pilot interfaces with ASTRA. The specification first classified flight conditions as being either *nominal* or *non-nominal*. This definition was in turn used to determine how ASTRA could best display "advice" to the pilot. The specification also introduced the concept of *Automatic Mode Switching*, whereby the ASTRA displays were automatically reconfigured as a function of Flight Mode

Interpreter (FMI) output. Consequently, pilot workload and situational awareness could be further optimized through the automatic, timely presentation of critical flight information to the pilot.

The role of using *fuzzy logic* in the ASTRA FMI was described. In short, this reasoning mechanism applies fuzzy algorithms to several aircraft state parameters so that the FMI can correctly interpret the aircraft flight mode. The FMI rule base developed for GAPATS was significantly modified so that the state-space for each aircraft parameter was more adequately partitioned. That is, the inference scheme needed to consider all conditions which *could* define a flight mode, rather than which *should* define the flight mode. Consequently, there was sought an FMI rule base for ASTRA ensuring that its inference was *independent* of aircraft configuration and pilot technique.

A new aircraft state parameter, *distance*, was proposed for inclusion in the FMI rule base. This parameter, which is used to measure the range between several points in space (such as the aircraft position and the destination airfield), should make the FMI even more reliable and further reduce any nervousness previously demonstrated. However, using *distance* in the inference scheme requires the integration of the Global Positioning System (GPS) into the system architecture, as well as the development of an ASTRA Navigation Module and several associated data bases.

In addition to providing *distance* information to the FMI, the ASTRA Navigation Module must also provide *altitude* information, by estimating the height of the aircraft above the ground. Likewise, the Navigation Module calculates a number of additional parameters which greatly assist the pilot in basic pilotage and navigation. By continuously calculating and updating these parameters (which include *time*, *distance*, and *heading*), this subsystem will greatly reduce the amount of pilot workload required for navigation and significantly enhance his situational awareness. These parameters can further be used to generate the display of an aircraft *Flight Director* (FD), which presents navigation/steering information to the pilot on a flight display.

The two flight displays described in the system architecture provide the critical communication linkage between ASTRA and the pilot. The first of these is the *Head-Up Display* (HUD), which the pilot uses as a primary flight display. That is, the HUD provides the pilot with all essential flight information necessary for pilotage and navigation, without requiring him to look down into the cockpit. The second of these is the *Head-Down Display* (HDD), a multi-function display designed so that the pilot may view and manipulate data from a wide variety of display options. In other words, the pilot also uses the HDD to perform the critical task of communicating with ASTRA—it is there that he enters all mission planning data associated with his flight.

Evaluating ASTRA and its various subsystems is a critical task for ensuring that the system evolves into a commercially viable product. Implied in this goal is that ASTRA satisfies the flight certification requirements of the FAA. In fact, flight certification is a minimum prerequisite, because a system that is difficult to operate, upgrade, and maintain, even if certified for flight, will not be commercially viable. To this end, it is essential that a subjective pilot evaluation be used to address and optimize the pilot interface issues—such as pilot workload and situational awareness—throughout the development of ASTRA.

The Engineering Flight Simulator (EFS) provides an excellent tool for this evaluation effort, since it is currently modeled as a Commander-700. Consequently, by flying representative *mission tasks* in the EFS, it is possible to record and subjectively evaluate the performance of several subsystems in a single setting. This then, is the first step in evaluating ASTRA: the identification of representative, repeatable tasks (corresponding to each of the FMI flight modes), which any pilot can fly in nominal and non-nominal conditions. Proper task definition must ensure that data being recorded for evaluation is consistent, despite variations in pilot style or technique. In this way, the FMI, the ASTRA flight displays, and the PA alarm rule base can be thoroughly evaluated as the pilot flies each mission task.

Future Challenges for ASTRA

The ASTRA system specification and architecture design presented in this thesis was designed to include provisions for future systems, such as digital data link communications. It was further designed to consider most situations the GA pilot might expect or encounter when conducting flights, either under Visual Flight Rules (VFR) or Instrument Flight Rules (IFR). During the course of the ASTRA system design, however, other possibilities for system improvement and future research were identified, as enumerated below.

Definition of Additional FMI Flight Modes. As was detailed in earlier chapters, the FMI recognizes seven distinct flight modes—*taxi*, *takeoff*, *climbout*, *cruise*, *initial approach*, *final approach*, and *landing*. While these seven modes generally describe most situations seen during the course of any flight, it may be appropriate to further define several new flight modes, such as *descent*, *holding*, and *go-around*. The first mode, *descent*, might occur when ATC clears the aircraft to an lower (intermediate) altitude, but the aircraft is too far from the destination to be in an approach mode. The second of these, *holding*, occurs when ATC clears the aircraft to fly (generally) a "one-minute pattern" about a navigation aid or fix on a specified course. The third mode, *go-around*, might occur whenever the pilot decides to abort an attempted landing or when he executes a "missed approach procedure" during an instrument approach.

There are two issues associated with defining new flight modes for inclusion in the FMI rule base. The first issue might be addressed as, "Are the new flight modes necessary?" In other words, the inclusion of a new flight mode should be used to generate new information/alarms that other flight modes do not already generate. These data should in turn be used to generate a new, *unique* HUD display set, which is selected for presentation (as before) through Automatic Mode Switching. If a new display set is unnecessary, or if the additional flight mode does not generate new information for display, then defining a new flight mode is probably not necessary.

The second issue concerns how “similar” the newly proposed flight modes are to previously defined modes, in terms of the aircraft state variables used in the fuzzy rule base. For example, the *descent* mode would probably be modeled very much like *initial approach*. Likewise, *holding* would likely be patterned similarly to *cruise*, while *go-around* would bear a close resemblance to *takeoff*. Consequently, it is likely that defining new flight modes would require the use of additional aircraft state parameters, or an extension of the “mode memory” algorithm currently implemented in the FMI. Without these modifications, it is probable that the FMI would be unable to differentiate between the similar flight modes, resulting in nervousness or incorrect decisions.

Application of the FMI Fuzzy Rule-Base Scheme to other General Aviation Aircraft. As might be discerned from this thesis and other related research projects (past and on-going), a great deal of effort went into the development of ASTRA, *specifically for the Commander-700 aircraft*. While ASTRA’s general system architecture is applicable to any GA aircraft, integrating the system into other aircraft would require extensive modification of several ASTRA subsystems. For example, the FMI membership functions concerning the parameter *airspeed* would be different for each individual aircraft. Likewise, the PA crisp rule base for displaying alarms, though written using general parameters, would require some revision. Finally, the aircraft data base would certainly be unique for each aircraft type.

One of the long-term goals for the ASTRA project is to provide a commercially viable advisory system for *any* GA aircraft. Consequently, a means for quickly codifying the

ASTRA fuzzy and crisp rule bases, as well as the aircraft data base, for new aircraft types is essential. In short, this would entail transforming appropriate operator's handbook data into the various formats described in this thesis. It would further entail the validation of these rule/data bases, to facilitate the FAA's flight certification process.

Displaying Multiple Alarms. As detailed in earlier chapters and in Appendix D, alarms were categorized into three distinct categories: *Warnings*, *Cautions*, and *Advisories* (WCA's). Furthermore, the display areas of the HUD and HDD were partitioned so that a WCA was displayed in a distinct, consistent location (i.e., all *Warnings* displayed in one location, all *Cautions* in a second, and all *Advisories* in a third). Development of these WCA's consequently concentrated on identifying the alarms to display and describing the conditions which would trigger an alarm. However, it was generally assumed that only one alarm would be active at any time.

Consequently, it may be necessary to investigate how best to display multiple alarms, particularly when two or more alarms of the *same category* are active. For example, during execution of an approach it would be possible to have one advisory active because of an incorrect aircraft configuration and another for an inappropriate airspeed. As this example illustrates, it may be necessary to develop an "alarm hierarchy," whereby all alarms within one category are prioritized in order of "seriousness." Furthermore, it will be necessary to develop a means to indicate to the pilot that several alerts are active, and a mechanism for the pilot to view all active alarms.

Development of the ASTRA Flight Director (FD). The FD, as previously described, provides one of the most useful pieces of data displayed on the HUD. In short, the FD described in this thesis provides steering cues to assist the pilot in maintaining a desired course and altitude. Consequently, the pilot can immediately tell, with a quick glance at the FD, whether the aircraft is maintaining the desired flight profile, and what corrective action to take if it is not.

The FD described herein provides its steering commands through the integration of GPS and the Navigation Module. Conventional FD's, on the other hand, have generally provided steering commands as function of data received from navigation aids (NAVAIDS) [22]. Consequently, a fully functional ASTRA FD should provide for both capabilities, so that the FD can be coupled to either the ASTRA GPS or to any civil NAVAIDS. Furthermore, it should permit the aircraft to smoothly transition from one mode to the other. For example, the pilot should be able to navigate with the FD by first using GPS, then following an ATC vector to intercept an ILS localizer, then tracking inbound on the ILS to execute the instrument approach. A system as powerful as ASTRA should have all these capabilities.

Development of a PC-Based Flight Planning Station. The ASTRA system architecture described in this thesis gives the pilot tremendous mission planning capabilities. A logical extension of this capability would be to give the pilot the same planning tools at a personal computer (PC). This capability is a logical extension of the fact that many pilots are already using a PC to file their flight plans, to check weather conditions, and to verify notices to airmen (NOTAM).

Consequently, all these functions could be integrated into a single ASTRA PC Flight Planning Station. The pilot could accordingly then plan a flight, check enroute and destination weather conditions, verify NOTAM's, investigate alternate routes, calculate fuel requirements, verify aircraft weight and balance conditions, and file a flight plan at a single sitting. The mission planning data could be recorded on a diskette, which the pilot would download into the aircraft when ready for departure. A mission planning scheme such as this would greatly reduce the amount of workload required in the cockpit, while resulting in more detailed flight planning. Furthermore, once a pilot has completed his flight, he could use the Flight Planning to review the details of his flight, focusing on errors he made throughout in an effort to improve his piloting skills. Naturally, the PC Flight Planning Station would have to have access to the exact same information (such as the Aircraft and Navigation Data Bases) as the onboard ASTRA system.

Conclusions

The ASTRA system architecture described in this thesis should be considered a blueprint for a simple, yet powerful, system still very much in its infancy. Undoubtedly ASTRA will continue to mature and improve as other control and inference methods are investigated and developed. Some of these other inference schemes, such as Kelly's hypertrapezoidal membership functions [13] and Nguyen's neural network engine [15], will likely provide a nice complement to existing control and inference ASTRA algorithms. When the system becomes reality, ASTRA will be a truly robust pilot advisory system, one which can infer and advise the pilot on virtually all imaginable situations.

The pilot advisory system being developed in conjunction with this thesis, along with its associated systems, software, and development tools, make up but one of many steps that must be taken in addressing the complex problems associated with smart cockpit technology. It is the author's hope that this thesis will provide a significant contribution to this important research area, and that it will form a solid foundation from which a commercially viable pilot advisory system may be produced. Indeed, if ASTRA were to save one aircraft or one life in the future, that goal would have been reached.

REFERENCES

[1] *Airline Accidents, Fatalities, and Accident Rates (1995 Preliminary Data)*, National Transportation Safety Board Report, SB 96-03, Washington, DC: United States Government Printing Office, January 1996.

[2] *Airman's Information Manual (AIM) 1995*, U.S. Department of Transportation, Federal Aviation Administration, Washington, DC: United States Government Printing Office, November 1994.

[3] M. W. Anderson, "Flight Test Certification of Multipurpose Head-Up Display for General Aviation Aircraft," *Journal of Aircraft*, Vol. 33, No. 3, pp. 532—538, May — June 1996.

[4] G. E. Cooper and R. P. Harper, *The Use of Pilot Rating in the Evaluation of Aircraft Handling Qualities,* National Aeronautics and Space Administration Technical Note, NASA TN D-5153, Washington, DC: National Aeronautics and Space Administration, April 1969.

[5] C. W. de Silva, *Intelligent Control: Fuzzy Logic Applications*, New York: CRC Press, 1995.

[6] *Equipment, Systems, and Installations in Part 23 Airplanes*, U.S. Department of Transportation, Federal Aviation Administration Advisory Circular, FAA AC 23.1309-1B, Washington, DC: United States Government Printing Office, July 1995.

[7] *Flight Test Guide for Certification of Part 23 Airplanes*, U.S. Department of Transportation, Federal Aviation Administration Advisory Circular, FAA AC 23-8A, Washington, DC: United States Government Printing Office, August 1993.

[8] J. C. Giarratano, *CLIPS User's Guide, CLIPS Version 6.0*, National Aeronautics and Space Administration, Johnson Space Center Document, JSC-25013, Houston, TX: Lyndon B. Johnson Space Center, May 1993.

[9] *Guidelines for Using Global Positioning System Equipment for IFR Enroute and Terminal Operations and for Non-precision Instrument Approaches in the U.S. National Airspace System*, U.S. Department of Transportation, Federal Aviation Administration Advisory Circular, FAA AC 90-94, Washington, DC: United States Government Printing Office, December 1994.

[10] V. Q. Harral, "Genetic Algorithms for Tuning Fuzzy Membership Functions," M. S. Thesis, Department of Electrical Engineering, Texas A&M University, College Station, TX 77843.

[11] L. A. Haworth and R. L. Newman, *Flight Test Techniques for Evaluating Flight Displays,* National Aeronautics and Space Administration Technical Memorandum, NASA TM-103947, Moffett Field, CA: Ames Research Center, February 1993.

[12] *Installation of Electronic Display Instrument Systems in Part 23 Airplanes*, U.S. Department of Transportation, Federal Aviation Administration Advisory Circular, FAA AC 23.1311-1, Washington, DC: United States Government Printing Office, June 1993.

[13] W. Kelly and J. H. Painter, "Hypertrapezoidal Fuzzy Membership Functions," *Proceedings of the 5th IEEE International Conference on Fuzzy Systems*, New Orleans, LA, September 1996.

[14] W. Kelly and J. H. Painter, "Soft Computing in the General Aviation Cockpit," *1st On-Line Workshop on Soft Computing,* http://www.biole.nuee.nagoya-u.ac.jp/wsc1, Society of Fuzzy Theory and Systems, August 19-30, 1996.

[15] T. X. Nguyen, "A Neural Network Mode Inference Engine for the Advisory System for Training and Safety," M. S. Thesis, Department of Aeronautical Engineering, Texas A&M University, College Station, TX 77843.

[16] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Upper Saddle River, NJ: Prentice Hall, 1995.

[17] *Operator's Manual for Army AH-64A Helicopter*, United States Army Technical Manual, TM-1520-238-10, Washington, DC: Headquarters, Department of the Army, June 1989.

[18] *Operator's Manual for Army MH-60K Helicopter*, United States Army Technical Manual, TM-1520-250-10, Washington, DC: Headquarters, Department of the Army, June 1994.

[19] J. H. Painter, "Fuzzy Decision and Control, The Bayes Context," *Proceedings of the 32nd IEEE Conference on Decision and Control*, San Antonio, TX, December 15-17, 1993.

[20] J. H. Painter and D.T. Ward (Texas A&M University); J. W. Crump and W. W. Nash (Knowledge Based Systems, Inc.), "General Aviation Pilot Advisor and Training System (GAPATS)," Final Report, NASA Langley Research Center, Contract Number NAS1-20290, Hampton, VA, September 1995.

[21] *Rockwell Commander-700 Pilots Operating Handbook*, Rockwell International, P/N M700001-1, Bethany, OK: Rockwell International (General Aviation Division), March 1979.

[22] B. Stevens and F. Lewis, *Aircraft Control and Simulation*, New York: Wiley, 1992.

[23] D. T. Ward and D. L. Y. Woo, "A Simulator Study of Advanced Head-Up Display Symbology for Use in General Aviation Aircraft," *35th AIAA Aerospace Sciences Meeting & Exhibit*, Reno, NV, January 1997.

APPENDIX A

AERONAUTICAL / ASTRA ACRONYMS AND ABBREVIATIONS

"Never use a big word when a diminutive one will suffice."

—Anonymous

This appendix provides a consolidated listing of the abbreviations and acronyms common to both general aviation and the ASTRA program. Specific definition and explanation of these terms may be found in this thesis, as well as in the Airman's Information Manual [2].

ADF	Automatic Direction Finder (see also NDB)
ADIZ	Air Defense Identification Zone
AGL	Above Ground Level (altitude)
ASTRA	Automated Safety and TRaining Avionics
ATC	Air Traffic Control
AMS	Automatic Mode Switching
ATIS	Automatic Terminal Information Service
BIT	Built-in-Test
DH	Decision Height
DME	Distance Measuring Equipment

EFC	Expect Further Clearance time
EFS	Engineering Flight Simulator
ETA	Estimated Time of Arrival
ETE	Estimated Time Enroute
FAA	Federal Aviation Administration
FAF	Final Approach Fix (of an instrument approach)
FMI	Flight Mode Interpreter
fpm	feet per minute
GA	General Aviation
GAPATS	General Aviation Pilot Advisor and Training System
GMT	Greenwich Mean Time (also called "Zulu" Time)
GND	GrouND (traffic controller)
GPS	Global Positioning System
G/S	Glide Slope (of an Instrument Landing System)
GUI	Graphical User Interface
HDD	Head-Down Display
HSI	Horizontal Situation Indicator
HUD	Head-Up Display
IAF	Initial Approach Fix (of an instrument approach)
IAS	Indicated Airspeed
IFR	Instrument Flight Rules
ILS	Instrument Landing System (a precision instrument approach)
IM	Inner Marker

IMC	Instrument Meteorological Conditions
KIAS	Knots Indicated Airspeed
LOC	Localizer (of an Instrument Landing System)
MAP	Missed Approach Point
MDA	Minimum Descent Altitude
MEA	Minimum Enroute Altitude
MFD	Multi-Function Display
MM	Middle Marker
MOA	Military Operations Area
MOCA	Minimum Obstacle Clearance Altitude
MRA	Minimum Reception Altitude
MSA	Minimum Safe Altitude
MSL	Altitude above Mean Sea Level
NAVAIDS	Navigational aids (such as DME, GPS, ILS, NDB, TACAN, VOR, VORTAC)
NDB	Non-Directional Beacon (see also ADF)
NOTAM	Notice to Airmen
NTSB	National Transportation Safety Board
OM	Outer Marker
PA	Pilot Advisor
RA	Restricted Area
SIGMET	SIGnificant METeorological advisory
SID	Standard Instrument Departure

SR	Situation Recognizer
STAR	Standard Terminal Arrival
TACAN	TACtical Airborne Navigation (navigation aid)
TWR	ToWeR (air traffic controller)
VFR	Visual Flight Rules
VMC	Visual Meteorological Conditions
VOR	VHF Omni-Directional Ranging (navigation aid)
VORTAC	VOR-TACAN (navigation aid)
VSI	Vertical Speed Indicator
WA	Warning Area
WCA	Warnings, Cautions, and Advisories

APPENDIX B

SUMMARY OF NATIONAL TRANSPORTATION SAFETY BOARD (NTSB) AVIATION ACCIDENT DATA

"A ship in harbor is safe, but that is not what ships are built for."

—John A. Shedd

This appendix provides an excerpt of aviation accident data reported by the NTSB [1]. The tables summarize accident rate information pertaining specifically to general aviation aircraft. Table 1 details preliminary accident rate information for all aircraft in 1995.

Table 1.

Accidents, Fatalities, and Rates, 1995 Preliminary Data: Air Carriers and General Aviation

Aircraft Operator	Accidents		Fatalities		Aircraft Hours Flown	Departures	Accident Rates			
							Per 100,00 Aircraft Hours		Per 100,000 Departures	
	Total	Fatal	Total	Fatal			Total	Fatal	Total	Fatal
Air Carriers (Part 121)										
Scheduled	33	2	166	160	12,648,000	8,220,000	0.261	0.016	0.401	0.024
Nonscheduled	2	1	2	2	861,000	447,000	0.232	0.116	0.447	0.224
Air Carriers (Part 135)										
Scheduled	12	2	9	9	2,580,000	3,506,000	0.456	0.078	0.342	0.057
Nonscheduled	76	24	52	52	2,000,000	N/A	3.80	1.20	N/A	N/A
General Aviation [1]	2,066	408	732	725	20,000,000	N/A	10.33	2.04	N/A	N/A
U.S. Civil Aviation [2]	2,188	437	961	948						

Notes:

[1] Accidents involving U.S. registered civil aircraft not operated under CFR 121 or CFR 135.

[2] Accidents and fatalities in the categories do not necessarily sum to the figures in U.S. Civil Aviation. Difference are due to collisions involving aircraft in different categories.

N/A Data not available.

Table 2 summarizes accident and accident rate data for general aviation aircraft during the period 1982-1995.

Table 2.

Accidents, Fatalities, and Rates, 1982-1995: U.S. General Aviation [1]

Year	Accidents		Fatalities		Aircraft	Accident Rates per 100,000 Aircraft Hours [3]	
	Total	Fatal	Total	Aboard	Hours Flown [2]	Total	Fatal
1982	3,233	591	1,187	1,170	29,640,000	10.90	1.99
1983	3,078	556	1,069	1,062	28,673,000	10.73	1.94
1984	3,017	545	1,042	1,021	29,099,000	10.36	1.87
1985	2,739	498	955	944	28,322,000	9.66	1.75
1986	2,582	474	967	878	27,073,000	9.54	1.75
1987	2,495	447	838	823	26,972,000	9.25	1.65
1988	2,385	460	800	792	27,446,000	8.69	1.68
1989	2,232	431	768	765	27,920,000	7.98	1.53
1990	2,215	442	766	761	28,510,000	7.77	1.55
1991	2,175	432	786	772	27,226,000	7.98	1.58
1992	2,073	446	857	855	23,792,000	8.71	1.87
1993	2,039	398	736	732	22,531,000	9.05	1.77
1994	1,990	401	723	716	21,873,000	9.09	1.83
1995 [4]	2,066	408	732	725	20,000,000	10.33	2.04

Notes:

[1] U.S. Registered civil aircraft not operated under CFR 121 or CFR 135.

[2] Source of estimate: FAA.

[3] Suicide and sabotage accidents excluded from rates.

[4] Preliminary data.

APPENDIX C

ASTRA INSTRUMENTATION AND SENSORS

"Measure what is measurable, and make measurable what is not so."

—Galileo Galilei

Table 3 provides a comprehensive listing of the instrumentation and sensors planned for use in the Commander-700 aircraft during ASTRA development.

Table 3.

ASTRA Instrumentation / Sensor Suite

Data Type	Flight Parameter	Comments / Status
Aerodynamic	Static Pressure	Pressure transducers to measure altitude, airspeed,
	Total Pressure	& rate of climb; installed on aircraft [1]
	Outside Air Temperature	Used for airspeed & engine performance; installed on aircraft
	Angle of Attack (Alpha)	Optical encoders installed on aircraft [1]
	Angle of Sideslip (Beta)	
Aircraft Attitude	Roll Angle	Awaiting procurement
	Pitch Angle	
Engine Performance	Manifold Pressure	Pressure transducer installed on aircraft
	RPM	Hall Effect sensor installed on aircraft
	Fuel Flow	Flow transducer installed on aircraft
Aircraft Configuration	Gear Position	Awaiting procurement
	Flap Position	
Navigation	Magnetic Heading	Awaiting procurement
	Position (latitude/longitude)	GPS receiver (digital output) with supporting software
	Time (GMT)	procured but not installed

Note:

[1] Aerodynamic sensors are installed as part of the existing flight-test boom on the left wing of the Commander-700 research aircraft.

To ensure a smooth transition from simulation to flight test during ASTRA development, each sensor parameter was simulated, using the same data format of the actual sensor, for use in the Engineering Flight Simulator.

APPENDIX D

PILOT ADVISOR FLIGHT DISPLAY RULE BASE

"Decide promptly, but never give any reason. Your decisions may be right, but your reasons are sure to be wrong."

—Lord Mansfield

Introduction

This appendix provides a crisp rule base summary which the Pilot Advisor (PA) may use to configure the display sets for the Head-Up Display (HUD) and Head-Down Display (HDD). The rule base is written in pseudo-code to facilitate translation into CLIPS (which is used by the PA) and C++ (which is used to generate the individual display pieces). To further assist in translating the rule base into working software code, the "Aircraft Constants" depicted in Figure 34 are defined. These parameters were derived from the Commander-700 operator's manual [21].

While Chapter 5 and this appendix recommend the *location*, *format*, and *appearance* for each symbology piece, it is important to note that these can only be determined after a thorough pilot evaluation using the Engineering Flight Simulator and flight test. Consequently, it is anticipated that the final display configurations—and the corresponding rule base—will significantly change as ASTRA matures.

```
// AIRCRAFT CONFIGURATION DEFINITIONS

//Define Aircraft Configuration: Landing Gear
   GearUp=0
   GearDown=1

//Define Aircraft Configuration: Flaps
   NoFlaps = 0                  // Flap setting 0 degrees
   TakeoffFlaps=12              // Flap setting 12 degrees ("Takeoff")
   LandFlaps=35                 // Flap setting 35 degrees ("Landing")

//RECOMMENDED OPERATING PROCEDURES (all airspeeds are KIAS!)
   V_rotate=80                  // V_rot = 80
   V_climb_min=120              // V_y = 120
   V_climb_max=140
   V_approach=90                // V_app = 87 (minimum)

//AIRCRAFT AIRSPEED LIMITATIONS (all airspeeds are KIAS!).  In general,
//a buffer of ~5 knots has been utilized, erring on the side of safety.
//This should allow the pilot sufficient time to react before exceeding
//a limit.

//Landing Gear Limits
   V_GearRetract=130            // V_lor = 137
   V_GearExtend=150             // V_loe = 155
   V_GearDown=150               // V_le = 155

//Flap Limits
   V_FlapsTakeoff=150           // V_fe = 155 (for 12 degrees)
   V_FlapsLanding=120           // V_fe = 128 (for 35 degrees)

// Aircraft Stall Speeds (as a function of configuration)
   V_CleanStall=90              // V_st = 86 (gear UP and flaps UP)
   V_NoflapStall=100            // Estimated (gear DOWN and flaps UP)
   V_LandingStall=70            // V_st = 65 (gear DOWN and flaps LANDING)

// Single-Engine Operations
   V_SingleEngineMin=80         // V_mca = 75
   V_SingleEngineClimb=105      // V_yse = 103
```

Figure 34. Pilot Advisor "Aircraft Constants" for the Commander-700

Generation of Flight Displays and Alarms

The HUD symbology display sets described in Chapter 5 were developed for *nominal* conditions. Note that they correspond exactly with FMI the inference of the Flight Mode interpreter (FMI). In other words, each inferred flight mode is associated with a unique flight display.

The PA adds alarms to the display modes whenever *non-nominal* flight conditions exist, as indicated in the following sections. For example, the PA will generate alarms as a result of aircraft configuration errors (recall from earlier chapters that the PA generates these alarms from its inference of what "should be," whereas the FMI makes its inference on what the flight mode "could be.") Likewise, the PA can generate alerts pertaining to navigation, mission planning, and training.

As was detailed in Chapter 5, display alarms are categorized into three levels: *warnings*, *cautions*, and *advisories* (WCA's), listed in decreasing level of urgency. As will be seen in this appendix, the manner in which these WCA's are displayed corresponds directly with the urgency of the alarm. Recall that the HDD architecture permits the pilot to "acknowledge" an alarm by selecting the *ALARM ACK* function key. This feature permits the pilot to override the display of an alarm. For example, the pilot may wish to leave the landing gear down while flying a traffic pattern. When the FMI infers the aircraft to be in the *Cruise* mode (during the downwind leg of the pattern), an alarm will be displayed indicating the pilot's error. The pilot overrides this alarm by selecting the HDD *ALARM ACK* function key.

This acknowledgment does not remove the alarm *per se*; rather, it changes how the alarm is formatted and displayed.

Alarms Associated with Aircraft Configuration

The PA generates aircraft configuration alarms as a result of piloting errors regarding landing gear and wing flaps. In general, such aircraft configuration alarms are a function of airspeed, because several airspeed limitations exist. For example, there are airspeed limitations associated with extending and retracting the landing gear, as well as airspeed limitations for flying the aircraft with the landing gear extended. Finally, each of these aircraft configurations directly effects the airspeed at which the aircraft will stall.

Taxi Mode. The aircraft flaps should be RETRACTED anytime the aircraft is being taxied. However, the pilot may wish to set the flaps to the TAKEOFF (T/O) position any time prior to actually starting the takeoff roll (in order to expedite the departure, for example). Likewise, the flaps are likely to be in the LANDING position as the aircraft completes its landing and transitions to the *Taxi* mode. Consequently, it is possible for the flap setting to be in any one of three settings. Despite this, it is still appropriate to display an advisory (Level I), which the pilot can acknowledge should he wish to override the alarm.

```
If [! NoFlaps]
   →Display "Flaps Set at XX degrees" Advisory; //XX = current setting
```

Note that landing gear errors are not appropriate for display during Taxi. If the pilot is taxiing with the gear up, he is likely to have more serious problems to worry about!

Takeoff Mode. In the *Takeoff* mode, the flaps should nominally be set to the T/O position. However, it is possible that the pilot wishes to fly a no-flap departure, should crosswind conditions warrant. Nevertheless, the PA should still alert the pilot to the fact that the flap setting is not correct. It is appropriate to display an advisory (Level I), which the pilot can acknowledge to override the alarm.

```
If [! TakeoffFlaps]
   →Display "Flaps Set at XX Degrees" Advisory; //XX = current setting
```

In this flight mode, the landing gear should be extended throughout the maneuver. However, since the *Takeoff* mode does not terminate until the aircraft reaches approximately 200 feet AGL, it is permissible for the pilot to have retracted the gear sooner. Consequently the alarms presented are limited to those relating to landing gear limitations. It is also assumed that the pilot will always retract the landing gear in flight, even when flying traffic patterns with the intent of executing several landings.

First consider that the pilot has not retracted the landing gear and is accelerating the aircraft; the aircraft is approaching V_{lor}. An advisory is displayed (Level I).

```
If [GearDown] && [IAS < V_GearRetract] && [IAS > (V_GearRetract - 10)]
```

```
→ Display flashing "Approaching Gear Retraction Speed" Advisory
```

Now consider that the gear are still down and the aircraft is above V_{lor}. As long as the pilot remains below V_{le}, no limitations are being exceeded. However, a caution (Level II) is displayed so that the pilot does not inadvertently retract the gear above V_{lor}.

```
If [GearDown] && [IAS > V_GearRetract] AND [IAS < (V_GearDown - 5)]
   → Display "Gear Retraction Speed Exceeded" Caution
```

Next consider that the aircraft is approaching V_{le} and the gear have still not been retracted. A caution (Level II) is displayed advising the pilot of the configuration error.

```
If [GearDown] && [IAS > (V_GearDown - 5)] && [IAS ≤ V_GearDown ]
   → Display flashing "Approaching Max Gear Speed" Caution
```

Finally, consider that the pilot has exceeded V_{le} with the gear extended. The previous caution is elevated to a warning (Level III).

```
If [GearDown] && [IAS > V_GearDown]
   → Display "Gear Speed Exceeded" Warning
```

The final group of alarms for the *Takeoff* mode considers that the pilot is flying at an airspeed near stall speed. This speed (V_{st}) is primarily a function of the aircraft configuration (flap and gear setting). First consider the aircraft to be in a "clean" configuration (gear and flaps UP) and approaching V_{st}. Display a caution (Level II) indicating that an increase in airspeed is appropriate.

```
If [NoFlaps] && [GearUp] && [IAS < (V_CleanStall + 5)] && [IAS >
V_CleanStall]
   → Display "Increase Airspeed" Caution
```

Next consider that the pilot has failed to notice the previous caution, yet continues to slow the aircraft. The previous caution is upgraded to a warning (Level III).

```
If [NoFlaps] && [GearUp] && [IAS ≤ V_CleanStall]
   → Display "Approaching Stall Speed" Warning
```

Now consider the previous situations, but with the aircraft in a "no-flap" landing configuration (gear DOWN and flaps UP), which can be appropriate for cross-wind landings. Display a caution (Level II) as the aircraft approaches V_{st}.

```
If [NoFlaps] && [GearDown] && [IAS < (V_NoflapStall + 5)] && [IAS >
V_NoflapStall]
   → Display "Increase Airspeed" Caution
```

Again consider that the pilot fails to notice the previous caution, and continues to slow the aircraft. The previous caution is upgraded to a warning (Level III).

```
If [NoFlaps] && [GearDown] && [ IAS ≤ V_NoflapStall]
    → Display "Approaching Stall Speed" Warning
```

Now consider the previous situations, but with the aircraft in a "landing" configuration (gear and flaps DOWN). Display a caution (Level II) as the aircraft approaches V_{st}.

```
If [! NoFlaps] && [GearDown] && [IAS < (V_LandingStall + 5)] && [IAS >
V_LandingStall]
   → Display "Increase Airspeed" Caution
```

The pilot, still flying in the landing configuration, fails to notice the previous caution and continues to slow the aircraft. The previous caution is upgraded to a warning (Level III).

```
If [! NoFlaps] && [GearDown] && [IAS ≤ V_LandingStall]
   → Display "Approaching Stall Speed" Warning
```

Climbout Mode. Generation of flap setting alarms in the *Climbout* assumes that the pilot has failed to notice any of the alarms during *Takeoff.* First consider that the flaps are not up and that no airspeed limits are being approached. An advisory (Level I) is displayed on both the HDD and HUD.

```
If [! NoFlaps] && [IAS < ((V_FlapsLanding - 10) || (V_FlapsTakeoff   -
10))]
   → Display "Retract Flaps" Advisory
```

Next consider that the pilot fails to notice the previous advisory. The pilot has not retracted the flaps and the aircraft is approaching a airspeed limit. The previous advisory is upgraded to a caution (Level II).

```
If ([TakeoffFlaps] && [IAS > (V_FlapsTakeoff - 10)] && [IAS    <
(V_FlapsTakeoff - 5)) ||  ([LandFlaps] && [IAS > (V_FlapsLanding    -
10) && [IAS < (V_FlapsLanding - 5)])
   → Display flashing "Retract Flaps" Caution
```

Now consider that the aircraft is approaching V_{fe} and the flaps are still extended. A caution (Level II) is displayed advising the pilot of the configuration error.

```
If ([LandFlaps] && [IAS > (V_FlapsLanding - 5)] && [IAS ≤
V_FlapsLanding]) ||  ([TakeoffFlaps] && [IAS > (V_FlapsLanding - 5)]
&& [IAS ≤ V_FlapsLanding])
   → Display  flashing "Approaching Max Flap Speed" Caution
```

Finally consider that the pilot has failed to notice all previous advisories and cautions; the aircraft is above V_{fe}. The previous caution is elevated to a warning (Level III).

```
If ([LandFlaps] && [IAS > V_FlapsLanding]) || ([TakeoffFlaps] && [IAS
> V_FlapsLanding])
   → Display "Flap Speed Exceeded" Warning
```

The pilot should retract his gear in this flight mode, if he has not done so already. As with previous modes, these errors are a function of the aircraft airspeed limitations. First consider that the gear are down, but no limitations have been exceeded. An advisory (Level I) is displayed in both the HUD and HDD.

```
If [GearDown] && [IAS < (V_GearRetract - 10)]
   → Display  "Retract Gear" Advisory
```

Next consider that the pilot has not seen this advisory and is accelerating the aircraft; the gear are still down and the aircraft is approaching V_{lor}. The previous advisory is upgraded to a caution (Level II).

```
If [GearDown] && [IAS < V_GearRetract] && [IAS > (V_GearRetract - 10)]
   → Display flashing "Retract Gear" Caution
```

Now consider that the gear are still down and the aircraft is above V_{lor}. As long as the pilot remains below V_{le}, no limitations will be exceeded. However, a caution (Level II) is displayed so that the pilot does not inadvertently retract the gear above V_{lor}.

```
If [GearDown] && [IAS > V_GearRetract] AND [IAS < (V_GearDown - 5)]
   → Display "Gear Retraction Speed Exceeded" Caution
```

Next consider that the aircraft is approaching V_{le} and the gear have yet to be retracted. A caution (Level II) is displayed advising the pilot of the configuration error.

```
If [GearDown] && [IAS > (V_GearDown - 5)] && [IAS ≤ V_GearDown ]
   → Display flashing "Approaching Max Gear Speed" Caution
```

Finally consider that the pilot has exceeded V_{le} with the gear extended. The previous caution is elevated to a warning (Level III).

```
If [GearDown] && [IAS > V_GearDown]
   → Display "Gear Speed Exceeded" Warning
```

The final group of alarms for the *Climbout* mode considers that the pilot is flying at an airspeed near stall speed. This speed (V_{st}) is primarily a function of the aircraft configuration (flap and gear setting). First consider the aircraft to be in a "clean" configuration (gear and flaps UP) and approaching V_{st}. Display a caution (Level II) indicating that an increase in airspeed is appropriate.

```
If [NoFlaps] && [GearUp] && [IAS < (V_CleanStall + 5)] && [IAS >
V_CleanStall]
   → Display "Increase Airspeed" Caution
```

Next consider that the pilot has failed to notice the previous caution, yet continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [NoFlaps] && [GearUp] && [IAS ≤ V_CleanStall]
   → Display "Approaching Stall Speed" Warning
```

Now consider the previous situation, but with the aircraft in a "no-flap" landing configuration (gear DOWN and flaps UP; this configuration can be appropriate for cross-wind landings). Display a caution (Level II) as the aircraft approaches V_{st}.

```
If [NoFlaps] && [GearDown] && [IAS < (V_NoflapStall + 5)] && [IAS >
V_NoflapStall]
   → Display "Increase Airspeed" Caution
```

Next consider that the pilot has failed to notice the previous caution, and continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [NoFlaps] && [GearDown] && [ IAS ≤ V_NoflapStall]
    → Display "Approaching Stall Speed" Warning
```

Consider the situation with the aircraft in a "landing" configuration (gear and flaps DOWN) and approaching V_{st}. Display a caution (Level II) as the aircraft approaches V_{st}.

```
If [! NoFlaps] && [GearDown] && [IAS < (V_LandingStall + 5)] && [IAS >
V_LandingStall]
   → Display "Increase Airspeed" Caution
```

Finally consider the aircraft in the landing configuration, but the pilot fails to notice the previous caution, and continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [! NoFlaps] && [GearDown] && [IAS ≤ V_LandingStall]
   → Display "Approaching Stall Speed" Warning
```

Cruise Mode. Aircraft configuration alarms for *Cruise* once again assume that the pilot failed to notice alarms from previous modes, and are generated similarly. First consider that the flaps are not up and that no airspeed limits are being approached. An advisory (Level I) is displayed on both the HDD and HUD.

```
If [! NoFlaps] && [IAS < ((V_FlapsLanding - 10) || (V_FlapsTakeoff-10))]
   → Display "Retract Flaps" Advisory
```

Next consider that the pilot fails to notice the previous advisory. The flaps are still not retracted and the aircraft is approaching a airspeed limit. The previous advisory is upgraded to a caution (Level II).

```
If ([TakeoffFlaps] && [IAS > (V_FlapsTakeoff - 10)] && [IAS    <
(V_FlapsTakeoff - 5)) || ([LandFlaps] && [IAS > (V_FlapsLanding-10) &&
[IAS < (V_FlapsLanding - 5)])
   → Display flashing "Retract Flaps" Caution
```

Next consider that the aircraft is approaching V_{fe} and the flaps are still extended. A caution (Level II) is displayed advising the pilot of the configuration error.

```
If ([LandFlaps] && [IAS > (V_FlapsLanding - 5)] && [IAS ≤
V_FlapsLanding]) || ([TakeoffFlaps] && [IAS > (V_FlapsLanding - 5)]
&& [IAS ≤ V_FlapsLanding])
   → Display  flashing "Approaching Max Flap Speed" Caution
```

Now consider that the pilot has failed to notice all previous advisories and cautions, and the aircraft is above V_{fe}. The previous caution is elevated to a warning (Level III).

```
If ([LandFlaps] && [IAS > V_FlapsLanding]) || ([TakeoffFlaps] && [IAS
> V_FlapsLanding])
   → Display "Flap Speed Exceeded" Warning
```

Consider that the pilot may have forgotten to retract the landing gear, but that no limitations have been exceeded. An advisory (Level I) is displayed in both the HUD and HDD.

```
If [GearDown] && [IAS < (V_GearRetract - 10)]
   → Display  "Retract Gear" Advisory
```

Next consider that the pilot has not seen this advisory and is accelerating the aircraft; the gear are still down and the aircraft is approaching V_{lor}. The previous advisory is upgraded to a caution (Level II).

```
If [GearDown] && [IAS < V_GearRetract] && [IAS > (V_GearRetract - 10)]
   → Display flashing "Retract Gear" Caution
```

Now consider that the gear are still down and the aircraft is above V_{lor}. As long as the pilot remains below V_{le}, no limitations will be exceeded. However, a caution (Level II) is displayed so that the pilot does not inadvertently retract the gear above V_{lor}.

```
If [GearDown] && [IAS > V_GearRetract] AND [IAS < (V_GearDown - 5)]
   → Display "Gear Retraction Speed Exceeded" Caution
```

Next consider that the aircraft is approaching V_{le} and the gear have yet to be retracted. A caution (Level II) is displayed advising the pilot of the configuration error.

```
If [GearDown] && [IAS > (V_GearDown - 5)] && [IAS ≤ V_GearDown ]
   → Display flashing "Approaching Max Gear Speed" Caution
```

Finally consider that the pilot has exceeded V_{le} with the landing gear extended. The previous caution (Level II) is elevated to a warning (Level III).

```
If [GearDown] && [IAS > V_GearDown]
   → Display "Gear Speed Exceeded" Warning
```

The next group of alarms considers that the pilot is flying at an airspeed near the stall margin. First consider the aircraft is in a "clean" configuration (gear and flaps UP) and is approaching V_{st}. Display a caution (Level II) indicating that an increase in airspeed is appropriate.

```
If [NoFlaps] && [GearUp] && [IAS < (V_CleanStall + 5)] && [IAS >
V_CleanStall]
   → Display "Increase Airspeed" Caution
```

Next consider that the pilot has failed to notice the previous caution, yet continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [NoFlaps] && [GearUp] && [IAS ≤ V_CleanStall]
   → Display "Approaching Stall Speed" Warning
```

Now consider the previous situation, but with the aircraft in a "no-flap" landing configuration (gear DOWN and flaps UP; this configuration can be appropriate for cross-wind landings). Display a caution (Level II) as the aircraft approaches V_{st}.

```
If [NoFlaps] && [GearDown] && [IAS < (V_NoflapStall + 5)] && [IAS >
V_NoflapStall]
   → Display "Increase Airspeed" Caution
```

Next consider that the pilot has failed to notice the previous caution, and continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [NoFlaps] && [GearDown] && [ IAS ≤ V_NoflapStall]
    → Display "Approaching Stall Speed" Warning
```

Consider the situation with the aircraft in a "landing" configuration (gear and flaps DOWN) and approaching V_{st}. Display a caution (Level II) as the aircraft approaches V_{st}.

```
If [! NoFlaps] && [GearDown] && [IAS < (V_LandingStall + 5)] && [IAS >
V_LandingStall]
   → Display "Increase Airspeed" Caution
```

Finally consider the aircraft in the landing configuration, but that the pilot fails to notice the previous caution, and continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [! NoFlaps] && [GearDown] && [IAS ≤ V_LandingStall]
   → Display "Approaching Stall Speed" Warning
```

Initial Approach Mode and Final Approach Mode. The aircraft configuration alarms for these two modes are identical, because the rule base takes into account differing pilot techniques as to when the flaps and landing are extended. In other words, these modes take into account that the flaps and gear can be retracted (at the beginning of the instrument approach) but should be extended at some point during these flight modes.

First consider that the aircraft is approaching V_{fe} and the flaps are extended. An advisory (Level I) is displayed.

```
If ([LandFlaps] && [IAS > (V_FlapsLanding - 5)] && [IAS ≤
V_FlapsLanding]) ||  ([TakeoffFlaps] && [IAS > (V_FlapsLanding - 5)]
&& [IAS ≤ V_FlapsLanding])
   → Display "Approaching Max Flap Speed" Advisory
```

Finally consider that the pilot has exceeded V_{fe} with the flaps extended. The previous caution is elevated to a warning (Level II).

```
If ([LandFlaps] && [IAS > V_FlapsLanding]) || ([TakeoffFlaps] && [IAS
> V_FlapsLanding])
   → Display "Flap Speed Exceeded" Caution
```

Next consider that the landing gear are down and the aircraft is above V_{lor}. As long as the pilot remains below V_{le}, no limitations will be exceeded. However, a caution (Level II) is displayed so that the pilot does not inadvertently retract the gear above V_{lor}.

```
If [GearDown] && [IAS > V_GearRetract] AND [IAS < (V_GearDown - 5)]
   → Display "Gear Retraction Speed Exceeded" Caution
```

Now consider that the pilot has exceeded V_{le} with the gear extended. The previous caution (Level II) is elevated to a warning (Level III).

```
If [GearDown] && [IAS > V_GearDown]
```

```
→ Display "Gear Speed Exceeded" Warning
```

The next group of alarms consider that the pilot is flying at an airspeed near the stall margin. First consider the aircraft is in a "clean" configuration (gear and flaps UP) and is approaching V_{st}. Display a caution (Level II) indicating that an increase in airspeed is appropriate.

```
If [NoFlaps] && [GearUp] && [IAS < (V_CleanStall + 5)] && [IAS >
V_CleanStall]
   → Display "Increase Airspeed" Caution
```

Next consider that the pilot has failed to notice the previous caution, yet continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [NoFlaps] && [GearUp] && [IAS ≤ V_CleanStall]
   → Display "Approaching Stall Speed" Warning
```

We now consider the previous situations, but with the aircraft in a "no-flap" landing configuration (gear DOWN and flaps UP; this configuration can be appropriate for cross-wind landings). Display a caution (Level II) as the aircraft approaches V_{st}.

```
If [NoFlaps] && [GearDown] && [IAS < (V_NoflapStall + 5)] && [IAS >
V_NoflapStall]
   → Display "Increase Airspeed" Caution
```

Now consider that the pilot has failed to notice the previous caution, and continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [NoFlaps] && [GearDown] && [ IAS ≤ V_NoflapStall]
   → Display "Approaching Stall Speed" Warning
```

Consider the situation with the aircraft in a "landing" configuration (gear and flaps DOWN) and approaching V_{st}. Display a caution (Level II) as the aircraft approaches V_{st}.

```
If [! NoFlaps] && [GearDown] && [IAS < (V_LandingStall + 5)] && [IAS >
V_LandingStall]
   → Display "Increase Airspeed" Caution
```

Finally consider the aircraft in the landing configuration, but the pilot fails to notice the previous caution, and continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [! NoFlaps] && [GearDown] && [IAS ≤ V_LandingStall]
   → Display "Approaching Stall Speed" Warning
```

Landing Mode. As will be seen with the landing gear, flap alarms in this flight mode must consider the fact that it is permissible for the flap setting to be either full up or full down. While many pilots will have extended the flaps during *Initial Approach* or *Final Approach*, no assumption has been made as to this. In fact, it may be appropriate for the pilot to have the flaps set at an intermediate setting (such as T/O) during *Initial Approach* or *Final Approach*. Alarms generated for the *Landing* mode assumes the aircraft is at or below Decision Height (nominally 200 feet AGL); the flaps should be fully extended by this point, if they are to be used. If the pilot elects to leave his flaps retracted (due to crosswind considerations, for example), the system should still alert him with an advisory. He can override such an alarm by acknowledging it.

First consider that the flaps are retracted or in the T/O position (12°) and no limits are being exceeded. An advisory (Level I) is displayed.

```
If ([NoFlaps] && [IAS ≤ (V_FlapsTakeoff)]) || ([TakeoffFlaps] && [IAS ≤
(V_FlapsLanding)])
   → Display "Check Flaps" Advisory
```

Next consider that the flaps are retracted or in the takeoff position, but that the aircraft is being flown too fast to fully extend the flaps.

```
If ([NoFlaps] && [IAS > (V_FlapsTakeoff)]) || ([TakeoffFlaps] && [IAS >
V_FlapsLanding)])
   → Display "Reduce Airspeed to Extend Flaps" Caution
```

Now consider that the flaps are extended, but that the aircraft is approaching the maximum flap speed. A caution is displayed (Level II).

```
If ([LandFlaps] && [IAS ≥ (V_FlapsLanding - 5)] && [IAS <
V_FlapsLanding) || ([TakeoffFlaps] && ([IAS ≥  (V_FlapsTakeoff - 5)] &&
[IAS < V_FlapsTakeoff)])
   → Display "Approaching MAX Flap Speed" Caution
```

Finally consider that the pilot has exceeded V_{fe} with the flaps extended. The previous caution is elevated to a warning (Level III).

```
If ([LandFlaps] && [IAS > (V_FlapsLanding)]) || ([TakeoffFlaps] && [IAS
>  (V_FlapsTakeoff)])
   → Display "Flap Speed Exceeded" Warning
```

While many pilots will have extended the landing gear during *Initial Approach* or *Final Approach*, no assumption has been made as to this. However, the gear must be down any time the aircraft is at or below Decision Height (nominally 200 feet AGL), . First consider that the gear are up and that no limitations are being exceeded. Display an advisory (Level I).

```
If [GearUp] && [IAS ≤ (V_GearExtend)]
   → Display "Check Gear" Advisory
```

Next consider that the gear are up but that aircraft is being flown too fast to extend to the landing gear. Display a caution (Level II).

```
If  [GearUp] && [IAS > (V_GearExtend)]
   → Display "Reduce Airspeed to Extend Gear" Caution
```

Now consider that the gear are down and the aircraft is above V_{lor}. As long as the pilot remains below V_{le}, no limitations will be exceeded. However, a caution (Level II) is displayed so that the pilot does not inadvertently retract the gear above V_{lor}.

```
If [GearDown] && [IAS > V_GearRetract] && [IAS < (V_GearDown - 5)]
   → Display "Approaching MAX Gear Speed" Caution
```

Finally consider that the pilot has exceeded V_{le} with the gear extended. The previous caution (Level II) is elevated to a warning (Level III).

```
If [GearDown] && [IAS > V_GearDown]
   → Display "Gear Speed Exceeded" Warning
```

The last group of alarms in this section consider that the pilot is flying at an airspeed near the stall margin, a very real consideration during the *Landing* phase of an approach. First consider the aircraft is in a "clean" configuration (gear and flaps UP) and is approaching V_{st}. Display a caution (Level II) indicating that an increase in airspeed is appropriate.

```
If [NoFlaps] && [GearUp] && [IAS < (V_CleanStall + 5)] && [IAS >
V_CleanStall]
   → Display "Increase Airspeed" Caution
```

Next consider that the pilot has failed to notice the previous caution, yet continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [NoFlaps] && [GearUp] && [IAS ≤ V_CleanStall]
   → Display "Approaching Stall Speed" Warning
```

Now consider the previous situations, but with the aircraft in a "no-flap" landing configuration (gear DOWN and flaps UP; this configuration can be appropriate for cross-wind landings). Display a caution (Level II) as the aircraft approaches V_{st}.

```
If [NoFlaps] && [GearDown] && [IAS < (V_NoflapStall + 5)] && [IAS >
V_NoflapStall]
   → Display "Increase Airspeed" Caution
```

Next consider that the pilot has failed to notice the previous caution, and continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [NoFlaps] && [GearDown] && [ IAS ≤ V_NoflapStall]
    → Display "Approaching Stall Speed" Warning
```

Now consider the situation with the aircraft in a "landing" configuration (gear and flaps DOWN) and approaching V_{st}. Display a caution (Level II) as the aircraft approaches V_{st}.

```
If [! NoFlaps] && [GearDown] && [IAS < (V_LandingStall + 5)] && [IAS >
V_LandingStall]
   → Display "Increase Airspeed" Caution
```

Finally consider the aircraft in the landing configuration, but the pilot fails to notice the previous caution, and continues to slow the aircraft. The caution is upgraded to a warning (Level III).

```
If [! NoFlaps] && [GearDown] && [IAS ≤ V_LandingStall]
   → Display "Approaching Stall Speed" Warning
```

Decluttering the HUD

As detailed in the Chapter 5, the HUD presents several items, as a function of aircraft flight mode, that assist the pilot in properly configuring the aircraft. For example, during *takeoff*, the HUD displays an airspeed carat associated with the maximum airspeed for landing gear retraction (denoted V_{lor}). The goal of displaying these items is to *preclude* the pilot from ever seeing an alarm (by exceeding an airspeed limit, for example). That is, the airspeed carat just described reminds the pilot to retract the landing gear prior to reaching V_{lor}. Once the pilot has retracted the landing gear, however, there is no need to display this item. Consequently, the items detailed in this section serve to "declutter" the HUD by "removing" those items from the display which are no longer of concern to the pilot.

```
If [FMI Mode == Takeoff] && [Rate of Climb > Zero] || [Airspeed > Vr]
   → Remove "Vr Carat" from HUD Airspeed Indicator

If [FMI Mode == Climbout] && [GearUp]
   → Remove "Vg Carat" from HUD Airspeed Indicator

If [FMI Mode == Climbout] && [NoFlaps]
   → Remove "Vf Carat" from HUD Airspeed Indicator

If [FMI Mode == Final Approach] && [GearDown]
   → Remove "Vg Carat" from HUD Airspeed Indicator

If [FMI Mode == Final Approach] && [NoFlaps]
   → Remove "Vf Carat" from HUD Airspeed Indicator
```

Alarms Associated with HDD Modules

This class of alarms and alerts are closely associated with the various display modes available in the HDD, as described in Chapter 5. Unless noted otherwise, they will be displayed only on the HDD.

Built-In-Test Mode. Should ASTRA or any of its subsystems fail a BIT, the pilot should receive an immediate indication in both the HUD and HDD. The pilot can then acknowledge the alarm and check for specific error status through the HDD *BIT* menu.

```
If [BIT Failure] || [Subsystem Failure]
   → Display "BIT FAILURE" Caution
```

Check Lists Mode. The alerts associated with this mode indicate which specific check list menu to display, when the pilot selects the *CHCK LSTS* function key. Consequently, this mode demonstrates features similar to that of Automatic Mode Switching used in the HUD. The checks indicated in the code below correspond with the Commander-700 checklist.

```
If [FMI Mode == Taxi] && [No Previous FMI Mode]
   → Display "Before Start Engine" Checks

If [FMI Mode == Takeoff]
   → Display "Takeoff" Checks

If [FMI Mode == Climbout]
   → Display "After Takeoff" Checks

If [FMI Mode == Cruise]
   → Display "Cruise" Checks

If [FMI Mode == Initial Approach]
   → Display "Descent" Checks

If [FMI Mode == Final Approach]
   → Display "Before Landing" Checks

If [FMI Mode == Landing] && [No Previous FMI Mode]
   → Display "Before Landing" Checks

If [FMI Mode == Taxi] && [Previous FMI Mode == Landing]
   → Display "After Landing" Checks
```

Flight Planning Mode. The alarms associated with this flight mode are intended to preclude the pilot from entering incorrect or inaccurate information. The first alarm ensures that the pilot has entered accurate identifiers for the departure point and destination.

```
If [(Departure Pt. || Destination) != Navigation Data Base]
   → Display "Erroneous Identifier" Advisory
   → Display the faulty field in reverse video
```

The next alarm ensures that the pilot has entered accurate routing information (to include SID, STAR, airways, waypoints, NAVAIDS, and fixes).

```
If [(SID || STAR || Airway || Waypoint || NAVAID || Fixes) != Navigation
Data Base]
   → Display "Erroneous Routing" Advisory
   → Display the faulty field in reverse video
```

The next alarm ensures that the pilot has entered correct approach information for the designated destination.

```
If [(IAF || FAF || MAP || Approach) != Destination (Navigation Data
Base)]
   → Display "Erroneous Routing" Advisory
   → Display the faulty field in reverse video
```

The final flight planning alarm verifies that waypoints entered on a flight plan (though identified correctly) are logically ordered. Specifically, each waypoint must be within 80 NM of the previous waypoint, with a change in heading of 90 degrees or less. (Note however, that the pilot can override this alarm, were he to use GPS to fly "directly" to the destination.

```
If [dist(waypoint(x), waypoint(x+1)) ≥ 90] || [angle(waypoint(x),
waypoint(x+1)) > 90]
   → Display "Verify Routing" Advisory
   → Display the faulty field in reverse video
```

Navigation Mode. The Navigation Module generates virtually all information displayed on the HDD in this mode. However, several alerts and alarms are required for display on the HUD as well. The first alert cues the pilot that the aircraft is within two minutes of the next waypoint. The permits the pilot to anticipate a turn.

```
If [ETE(Next Waypoint) ≤ 2 min]
   → Display Second Waypoint Carat on HUD Heading Tape
```

The next alert likewise permits the pilot to anticipate a descent.

```
If [dist(Descent Point) < 2 min]
   → Display "Descend in X:xx" Advisory
```

The next alarms alert the pilot, should he ignore the FD cues presented in the HUD, resulting in excessive deviation from the assigned altitude or course

```
If [|Barometric alt - Assigned alt| > 300 feet]
   → Display "Check Altitude" Advisory
   → Display Flashing FD

If [|Vertical speed| > 300 ft/min] && [FMI Mode == Cruise]
```

```
  → Display "Rate of Climb" Advisory
  → Display Flashing FD

If [Vertical Speed > 300 ft/min] && [ FMI Mode == Approach || Landing]
  → Display "Rate of Climb" Advisory
  → Display Flashing FD

If [|Track - Course| > 3 NM]
  → Display "Check Course" Advisory
  → Display Flashing FD
```

The following alarm indicates to the pilot that he is approaching special use airspace (such as a Restricted Area) or other high density airspace (such as Class A or Class B Airspace). This alarm could also be repeated at various intervals, such as 20 NM, 10 NM, and airspace penetration.

```
If [Distance to Special Use Airspace < 20 NM]
  → Display "Within 20 NM of Special Use Airspace" Advisory

If [Distance to Special Use Airspace < 1 NM]
  → Display Flashing "Penetrating Special Use Airspace" Advisory
```

The following alarm, displayed on the HUD and HDD, notifies the pilot of excessive crosswind conditions (defined here to be 10 knots). It should be displayed whenever the inferred flight mode is *taxi*, *takeoff*, *final approach*, or *landing*.

```
If [(Wind speed > 10 knots) && (|Wind Direction - Final Course| > 30
degrees)]
  → Display "XWIND: Dir/Speed" Advisory
  → Display the faulty field in reverse video
```

Should the pilot select the *VECT* function key within the HDD *NAV* module, alarms are generated exactly as described under the *Flight Planning* Mode.

Weight and Balance Mode. Alarms in this mode, like that for Flight Planning, are designed to prevent the pilot from entering incorrect or inaccurate information. In this manner erroneous weight and balance calculations will be precluded. Checks are made for the number of pilots, number of passengers, amount of weight, and amount of fuel entered by the pilot, as well as the total weight and balance (calculated by ASTRA).

```
If [# Pilots > 2]
   → Display "Check # of Pilots" Advisory
   → Display the faulty field in reverse video


If [# Pax > 4]
   → Display "Check # of Passengers" Caution
   → Display the faulty field in reverse video


If [Weight(Cargo Area A) > Max Weight(Cargo Area A)]
   → Display "Check Weight: Area A" Caution
   → Display the faulty field in reverse video


If [Weight(Cargo Area B) > Max Weight(Cargo Area B)]
   → Display "Check Weight: Area B" Caution
   → Display the faulty field in reverse video


If [Weight(Fuel) > MaxWeight(Fuel) || Gallons(Fuel) > MaxGallons(Fuel)]
   → Display "Check Fuel Onboard" Caution
   → Display the faulty field in reverse video


If [Useful Weight(Aircraft) < 0]
   → Display "Check Aircraft Weight" Caution
   → Display the faulty field in reverse video


If [Moment(Aircraft) > Max Moment(Aircraft)]
   → Display "Check Aft CG!" Caution
   → Display the faulty field in reverse video
```

```
If [Moment(Aircraft) < Min Weight(Aircraft)]
   → Display "Check Forward CG!" Caution
   → Display the faulty field in reverse video
```

VITA

"When I was a boy of fourteen, my father was so ignorant I could hardly stand to have the old man around. But when I got to be twenty-one, I was astonished at how much the old man had learned in seven years."

—Mark Twain

Jeffrey Alan Trang received the B.S. Degree in Electrical Engineering from Rose-Hulman Institute of Technology, Terre Haute, Indiana in May 1983. Upon graduation he was commissioned a second lieutenant and entered active duty in the U.S. Army. After completing tours as an air cavalry and attack helicopter pilot in Germany, Texas, and *Operation Desert Shield*, Jeff was selected to attend the U.S. Naval Test Pilot School. He then served a tour as an experimental test pilot at Edwards Air Force Base, before receiving an Army fellowship to pursue his master's degree. Jeff presently holds the rank of major and has been awarded the Army's Senior Aviator Wings, having logged more than 2,000 hours in 43 different models of rotary-wing and fixed-wing aircraft.

Jeff's graduate work has concentrated on the interface and application of controls and artificial intelligence in aircraft and aircraft displays. He will receive his M.S. Degree in Electrical Engineering from Texas A&M University in May 1997. Upon graduation, Jeff will attend the Army's Command and General Staff College in Fort Leavenworth, Kansas.

Jeff is married to the former Dianna Lydia Svatek. They have two children, Allen and Amy, and an English Springer Spaniel, Cody. Their permanent mailing address is 5724 73rd Street, Lubbock, Texas 79424.