

NASA-TM-81957

NASA-TM-81957 19810011639

N8120167

FOR REFERENCE

NOT TO BE TAKEN FROM THIS ROOM

LIBRARY COPY

AUG 27 1993

LANGLEY RESEARCH CENTER
LIBRARY NASA
HAMPTON, VIRGINIA

A Service of:



National Aeronautics and
Space Administration

**Scientific and Technical
Information Program Office**

Center for AeroSpace Information

NOTICE

**THIS REPRODUCTION WAS MADE FROM THE BEST AVAILABLE
COPY OF WHICH A NUMBER OF PAGES WERE OF POOR
REPRODUCTION QUALITY.**



NASA Technical Memorandum 81957

The Aerospace Vehicle Interactive Design System

Alan W. Wilhite

February 1981

(NASA-TM-81957) THE AEROSPACE VEHICLE
INTERACTIVE DESIGN SYSTEM (NASA) 25 p
HC A02/MF A01 CSCL 22B

N81-20167

Unclas
G3/1d 41799



National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

THE AEROSPACE VEHICLE INTERACTIVE DESIGN SYSTEM

Alan W. Wilhite
NASA Langley Research Center

SUMMARY

The Aerospace Vehicle Interactive Design (AVID) is a computer-aided design system that was developed for the conceptual and preliminary design of aerospace vehicles. The AVID system evolved from the application of several design approaches in an advanced concepts environment in which both mission requirements and vehicle configurations are continually changing. The basic AVID software facilitates the integration of independent analysis programs into a design system where the programs can be executed individually for analysis or executed in groups for design iterations and parametric studies. Programs that have been integrated into an AVID system for launch vehicle design include geometry, aerodynamics, propulsion, flight performance, mass properties, and economics. This paper describes the evolution of the AVID based on the experience from past design approaches. The benefits of using AVID are illustrated by the conceptual design of a launch vehicle system.

INTRODUCTION

At the Langley Research Center, advanced launch vehicles are being studied to determine the high-yield technology areas that need to be developed for the evolution of an efficient transportation system that would enhance or eventually replace the space shuttle. Technology studies have been conducted in the areas of advanced propulsion, aerothermodynamics, materials, flight control, and structures. A quantitative assessment of the effect of these technologies has been defined based on their effects on vehicle mass and life-cycle cost.

A number of mission and vehicle concepts have been studied. Mission requirements have ranged from small payload capability (~10 metric tons), for personnel and priority payloads to large payload capability (~500 metric tons), for bulk cargoes such as propellants. Concepts have ranged from air, sled, or vertically launched vehicles to ballistic single- or two-staged vehicles.

It is an immense task to design and analyze the matrix of missions and concepts that are envisioned for the future. In order to complement our studies, a general computer-aided design system was developed called Aerospace Vehicle Interactive Design (AVID). The purpose of this paper is to describe the evolution of the AVID system based on the experience from past design approaches. The computer software and hardware that makes the system run efficiently are described. Finally, an advanced launch vehicle design system is presented to illustrate the benefits of using the AVID system.

DESIGN APPROACHES

The design of an aerospace vehicle is a challenging task because many engineering disciplines must be integrated together in order to define a concept which satisfies a given mission and set of requirements. Since a person is unlikely to be an expert in all the engineering disciplines, the classical approach to designing a vehicle is to identify a design leader who is in charge of a group of specialists in each of the engineering disciplines, as illustrated in figure 1. Usually the design leader has a rough idea of a concept that will perform the mission requirements. This concept is then analyzed by each of the specialists. The management of the total design process is difficult, since the specialists depend on each other for information or analyses. In addition, each discipline in the design process generally is constrained by the requirements of the other disciplines. Because of these design conflicts, many iterations are required before a suitable vehicle design emerges. Because the transfer of information is slow or incomplete, the design leader has a difficult time controlling the interface between disciplines. Design iterations are expensive and time consuming, and optimal design requiring many iterations is nearly impossible. Recent experience has shown that for companies that have continued to use this approach for advanced vehicle studies, a single concept iteration may take days or weeks. The major advantages of this approach are the flexibility of utilizing specialists and sophisticated single discipline computer programs for specific tasks and no additional investment to develop a design system for a one-time study. These advantages make this classical approach the most widely accepted design method today for advanced concept studies.

With the development of large computer systems, attempts have been made to integrate the various engineering disciplines into a single large computer program for preliminary design and mission analysis. Several of these synthesis programs have been successful (refs. 1-4). These programs closely couple the technologies so the design process can be very fast to allow parametric and sensitivity studies and optimization. Although the design process has been automated with these single synthesis programs, there are disadvantages that have limited the success of this approach. The systems are usually hard-wired for the synthesis of one particular vehicle class; thus, there is little flexibility in modifying the program or changing the analysis iteration cycle for a different type of vehicle or mission. The lack of flexibility is due to insufficient or restrictive data intercommunication between the technical disciplines. Another disadvantage is that the analysis techniques are often tailored toward the single vehicle class for simplification and increased computational speed. Finally, the designer cannot interact with the design process until a final point design is established by the computer, which nearly eliminates an important engineering design contribution--creativity.

In the late 1960's, systems were developed to combine the advantages of the previous design approaches by integrating independent analysis programs into a single design system. One system was developed for the Langley Research Center called ODIN (Optimal Design Integration, refs. 5 and 6). The ODIN system consisted of a data manager to allow the various technical

programs to communicate with each other and an executive to define the sequence of program executions needed in the design process (ref. 6). The program sequencing and looping is shown schematically in figure 1. This multiprogram approach has many advantages. The analysis programs can be used with little or no modifications, encouraging the integration of existing analysis tools. The confidence that exists with using familiar analysis programs leads to a procedure that can be used with confidence. Also, the same analysis programs do not have to be reprogrammed for each design synthesis simulation, resulting in reduced computer software overhead. The design process can be altered with changing requirements, and the execution of separate analysis programs can be restructured to design an entirely different class of vehicle. This flexibility decreases the development time of a design procedure and allows the analysis of a wider range of potential concepts.

The main problem of the ODIN approach is similar to the single-program synthesis approach in that the designer is not an integral part of the design process because of the batch (noninteractive) environment of the system. To use the system successfully, the complete design cycle must be known beforehand. For innovative concepts, design is an iterative, man-in-the-loop process where the results from each analysis step dictate the next step in the design process. The design cycle is often poorly defined ahead of time for this phase. Once a concept is defined after many design iterations, the pattern of analysis might be established. If so, the pattern can be easily programmed into the ODIN system. At this point, the concept can be rapidly modified, enhanced, or optimized. These final design changes and optimization are the problems best handled by the ODIN approach.

Finally, developing a geometry model of a vehicle concept is a major task using the batch mode. Programs were developed to automatically generate the geometry in ODIN, but innovative concepts with completely different shapes were rarely analyzed, since the geometry generation programs could not handle these shapes. Putting new configurations into the simulation by hand took from 1-6 days.

The AVID system was developed as a result of the experience gained from these previous design approaches. The objective was to incorporate the creative and flexible features of the classical design approach with the analysis integration and speed of the ODIN approach into an interactive computer-aided preliminary design system.

HISTORY OF AVID DEVELOPMENT

The AVID system evolved over the past 5 years. The development began with using interactive graphics on a minicomputer for geometric modeling of configurations and for interpreting a large volume of data generated on a mainframe computer. The geometric modeling of a vehicle was reduced from a 1-6 day task to a 1-2 hour task. Using the minicomputer for these tasks has been shown to greatly increase productivity at a substantial savings, since the interactive tasks have been off-loaded from an expensive batch-oriented mainframe and transferred to a low-cost interactive oriented minicomputer.

The bottleneck in the design cycle became the process of constructing the order of execution of analysis programs within the ODIN system for the design process. Since many different design concepts were under development and each had its own design cycle and mission constraints, the programmed batch design synthesis procedure had to be replaced. Experience had shown that a designer should be actively involved in the design cycle to observe intermediate results of analysis programs and to select the next design step or steps in real time based on those results. Instead of a complete automatic design sequence, an automated process was developed that is similar to the approach where a design team leader and his discipline specialists conduct preliminary design studies. At each step in the computerized design process, the team leader has the choice of requesting a specialist to perform the analysis or a computer program developed and interfaced to the design system by that specialist to perform the same analysis. These requirements led to the implementation of a computer-aided design system based on designer participation instead of a computer design system where there is no designer interaction once the design process has been initiated. Thus, the key to AVID is interactive design.

THE AVID SYSTEM

To provide an interactive design system, the software and hardware must allow rapid design interaction. The AVID software consists of an executive to control the design process, application programs for engineering analysis, and a data management system for data communications (fig. 2).

The main purpose of the executive is to allow the user to interactively execute technology programs without having to know computer languages or computer commands. The executive displays a menu of analysis or utility programs. After making the appropriate selection, the user may be asked to respond to questions about data input to insure that the correct information has been retrieved. The executive then steps through a series of computer commands that can execute one program or a group of programs that comprise one step in the design process. After finishing this design step, the menu is again displayed to the user for selection of the next step. If a definite pattern of design iterations emerges from those single design steps, the executive can be easily programmed to perform a sequence of design steps automatically. This function is not only useful for design iterations and also for parametric studies.

The AVID data management system provides the generality and flexibility of integrating and executing independent analysis programs for a given design project. This data management system provides two functions - to communicate design data from one program to another and to archive the design data for each individual configuration.

The communication database system is used to store design data consisting of single elements and arrays. Bulk data such as geometric coordinates are stored in separate files. Data can be stored, retrieved, or changed by users interactively and by programs through subroutines.

A typical example of interfacing an independent program with AVID is shown in figure 3. A pre-processor program is written to retrieve design data from the database. Using this design data, preset constant data in the pre-processor program, and data input by the user, the pre-processor program constructs an input file in the exact format that the analysis program expected before it was integrated into the AVID system. If the correct data is not in the database, the user is notified. After the input file is created, the analysis program is executed, and the desired design data is stored in the database to be retrieved by other analysis programs. With this system, the analysis program does not have to be changed, but only a small subroutine has to be added to communicate data from the program to the database. With this process, a specialist can recognize and correct mistakes easily, since he can analyze a familiar input file that has not changed from when the program was used in an independent mode. Since most analysis programs are interfaced in this manner, a generalized pre-processor was developed (similar to ODIN) to eliminate the development of many pre-processor programs.

The archive system stores all the design data and bulk data files that pertain to one configuration. When this data is saved, the user is requested to type in his name and a description of the current design. This descriptive data, plus the date and time of day, is saved with the design data. This capability is used to freeze the design so the engineer can break for an interruption and return at a later time. This design data can also be retrieved by other engineers for more detailed analysis or further configuration refinement.

AVID HARDWARE

The AVID hardware consists of graphical equipment, a minicomputer, and a mainframe computer as shown in figure 4. The graphical equipment consists of a graphics terminal, a graphics tablet for configuration geometric input, and a hard-copy unit. The minicomputer is a multilanguage, multiuser, time-share system with virtual memory. Because of the large capacity of the minicomputer, it is not dedicated to the AVID system. The computer is located remotely from the users and is accessed by several other groups to support several research efforts.

Because response on the minicomputer degrades with increasing size and execution time of computer programs, large programs (taking more than 1 minute of execution time) are executed on the mainframe computer. The minicomputer operating system was developed to support primarily interactive users and the mainframe system to support large batch tasks. Although the mainframe can support interactive users, it is not very efficient. Presently, the mainframe is supporting approximately 10 times more users than the minicomputer system, but only at low communication rates (combination of 300 and 1200 baud) as compared to the high communication rate (4800 and 9600 baud) on the minicomputer. The ratio of computer capital costs per user for the minicomputer is approximately \$25K/user, whereas for the mainframe it is \$70K/user. If the number of users on the mainframe computer was restricted to provide adequate

response like the minicomputer, the difference in costs would be more dramatic.

AVID APPLICATION TO LAUNCH VEHICLE ANALYSIS

Over the past several years, the AVID system has been used by the Langley Research Center for several space system preliminary design projects that include orbital-transfer vehicles, large advanced space structures, and launch vehicles. The following is a description of the independent analysis programs and design process that were developed for advanced launch vehicle analysis and synthesis.

As shown in figure 2, the application programs integrated into AVID for analyzing launch vehicles include geometry, mass properties, aerodynamics, propulsion, flight-performance, and economics. A typical design process may begin by creating a numerical model of the configuration that is used by the other analysis programs. The aerodynamic characteristics can then be determined. A propulsion system is selected and the ascent flight performance is computed based on flight constraints and mission requirements. The geometric, propulsion, and flight performance data are then used by the mass properties program to compute vehicle mass and center-of-gravity location.

Aerodynamic mission constraints such as minimum landing speed, glide-path angle, and aerodynamic trim can be checked based on the current center-of-gravity location. If these constraints are violated, the wing and/or tail size and location can be changed. Also, the internal components of the configuration can be rearranged to change center-of-gravity location.

After the geometric changes have been made, the analysis programs can be executed again to complete another iteration. These iterations are repeated until a configuration meets all requirements and constraints. Then the economic analyses can be performed.

Since the design process is interactive, this procedure can be changed, and several analysis steps can be eliminated based on their results. Thus, engineering judgment can be used extensively throughout the design process. Also, the system can be used for concepts unrelated to launch vehicles. For example, a configuration can be defined and aerodynamics can be computed independent of all other analysis programs. Finally, other analysis programs can easily be added to this system for design of another vehicle class. Thus, most vehicles can be designed if the analysis computer programs exist.

The following is a description of the analysis programs that have been used for launch vehicle analysis:

Geometry.- The design process begins by making a sketch of the configuration. This sketch consists of a top view, a side view, and several body cross sections as shown in figure 5. The coordinates of the sketch are put into the computer by using a digitizing tablet (fig. 4). Because the geometry program

uses several curve-fitting techniques, only a minimum amount of vehicle geometry needs to be digitized. In this mode, preliminary vehicle coordinates can be defined in minutes. If exacting detail is required, all body coordinates can be digitized, and no curvefit techniques are used. Using this geometry system, the numerical model in figure 6 was defined in less than an hour from the drawing in figure 5. After the outer mold lines are described, the numerical model can be processed by the geometric analysis program. This program analyzes each geometric component - body, wing, tail, etc. - for volume, areas, inertias, and centroid location. After the geometric modeling and analysis are completed, the configuration numerical model and geometric data can be archived.

At any time, the design information can be retrieved, changes can be made to any geometric component, or new components can be constructed for geometric analysis. The geometric data can also be processed through other application programs for analysis in other technical areas.

For the mass properties program, numerical models need to be generated for the major internal components such as propellant tanks, structural interfaces, and subsystems. The same type of algorithms used to generate the outer mold lines can also be used for the internal components. Also, special geometric generation routines have been developed for axisymmetric components, multibubble tanks, and rocket engines. Once the individual components are generated, they can be moved, scaled, or rotated. An example of internal packaging for a configuration is illustrated in figure 7. These internal components can be analyzed by the previously described geometric analysis program. When propellant requirements are known, tanks can be scaled and moved to make them compatible with the outer mold lines.

Aerodynamics. - There are several aerodynamic analysis programs that can be used, depending on the execution time and depth of analysis desired. For an interactive analysis (~30 seconds), a digital version of DATCOM (ref. 8) can be executed for subsonic and supersonic force and moment data. Also, a program based on modified Newtonian theory and comparable in execution speed to DATCOM, has been developed for the hypersonic regime.

There are also more sophisticated programs that have been interfaced with the system such as a vortex lattice program (ref. 9) for analysis in the subsonic regime and the Hypersonic Arbitrary Body Program (ref. 10). Both of these programs are executed on the mainframe computer, and turnaround time is from 10 minutes to an hour. Generalized aerodynamic data plotting programs can be used to display the output data or print the data in tabular form.

An aerodynamic analysis program called the Aerodynamic Preliminary Analysis System (ref. 11) has been developed for AVID by Rockwell International. This program analyzes arbitrary configurations at hypersonic to subsonic speeds. In addition to the basic static aerodynamic characteristics, control increments, stability derivatives, and rotary derivatives are also computed. This program is a complete system--interactive geometry input and manipulation, interactive analysis selection input, aerodynamic analysis, and interactive output displays. This system will soon become an integral part of the AVID system.

Propulsion.- A propulsion program was developed to reflect advanced rocket engine concepts that include the space shuttle main engine, modified space shuttle main engine (two-position nozzle), hydrocarbon engines with staged-combustion or gas-generator cycles, and a dual-mode engine that can burn a hydrocarbon and a hydrogen propellant in series. This program reflects data from a study conducted by the Aerojet Corporation (ref. 12).

For a propulsion analysis, the user is required to input the engine type, nozzle area ratios, propellant types, and the ratio of liftoff thrust to vehicle gross weight. The program computes thrust, specific impulse, exit areas, and engine masses that are used by the trajectory and mass properties programs. If a rocket engine is desired that is not in this program, the engine characteristics can be interactively input directly into the database without executing the propulsion program.

Trajectory.- After the aerodynamic and propulsion data have been established and a liftoff thrust-to-weight ratio selected, an ascent trajectory can be integrated to yield the required vehicle performance.

For new concepts, a generalized computer program called POST (Program to Optimize Simulated Trajectories (ref. 13), is used to maximize the burnout mass while satisfying a number of in-flight constraints that include limits on maximum q 's, maximum dynamic pressure, and maximum normal force. This program has not yet been integrated into the AVID system directly, since it resides on the mainframe computer and may have to be iterated several times to establish a nominal trajectory. Performance results, that usually require a one day turnaround from POST, can be interactively input into the database for communication with the vehicle sizing program.

A trajectory program which takes approximately 1 minute to execute has been developed for the AVID system. Using a POST nominal trajectory for guidance, this trajectory program can be used to compute performance changes due to small perturbations in the propulsion system and the liftoff thrust-to-weight ratio. The results from this trajectory program are shown in figure 8 where altitude, relative velocity, flight-path angle, angle of attack, dynamic pressure times angle of attack (=normal load factor), and g 's are displayed. The pertinent design parameters--mass ratio, optimum thrust, and propellant splits--obtained from this program are automatically put into the database for vehicle sizing.

Mass Properties.- The launch vehicle mass properties program has three main functions-- component mass estimation, vehicle sizing, and component center-of-gravity and inertias summation.

The mass estimation is based on semi-analytical and empirical techniques (refs. 1-3). For many of the major structural components, empirical equations are used. As an example, exposed wing mass is a function of the following parameters: sweep, aspect ratio, area, thickness ratio, and vehicle mass. The form of this function is determined from historical data and detailed analytical studies. For many studies, vehicle component mass is first

estimated with empirical equations. When a design of interest emerges, a structural analysis may be performed. The component empirical mass estimation techniques are then updated to reflect the detailed analytical results. These updated estimation techniques are used for further concept optimization and future studies of similar concepts.

To aid the vehicle synthesis process of iterating a design to meet certain mission requirements, a sizing algorithm has been incorporated into the mass estimation program. The first option determines the mass of the vehicle as given. Although there are no external sizing iterations, the vehicle mass estimation must still be iterated, since many of the components are a function of vehicle mass, and vehicle mass is the summation of all the individual component masses. The second option scales the thrust (and the associated mass) of the propulsion system to satisfy a liftoff thrust-to-gross weight ratio and determines the resulting payload mass based on trajectory mass ratio requirements (the ratio of gross mass to burnout mass, ref. 14). The last option scales the propulsion system and all the geometric entities (wing, tail, body, propellant tanks, etc.) as a function of a change in propellant tankage needed to meet the propellant requirements (trajectory performance) to deliver a specified payload to orbit. The masses of the vehicle components are also scaled automatically, since they are functions of the entities being scaled by the sizing algorithm. This last sizing option is used to determine the synergistic effects on a total vehicle system by changing the technology of individual components. It is also used to compare various vehicle concepts using a common mission requirement of delivering a specified payload to orbit. Approximately 30 mass and sizing iterations are performed to converge to a solution which takes only several seconds on a minicomputer. A component mass report for a typical single-stage vehicle is shown in Table I.

Economics. - After estimating component masses of a vehicle that meet the mission requirements, an economic analysis which consists of a vehicle cost analysis and a life-cycle analysis of the complete system can be made.

The cost analysis consists of a first unit, production, program initiation, recurring, and research, development, testing, and evaluation costs. The cost analysis is similar to the weights analysis, since most cost functions are formulated from historical data (ref. 15). With this cost analysis and an assumed mission model (number of flights each year) over the lifetime of the launch system, a total life-cycle analysis can be performed.

An example of economic results for a single-stage launch vehicle shown in Table II and figure 9. This system cost analysis has been used to compare systems in terms of development and recurring costs (ref. 15). To compare the merit of an advanced system over the continued use of a current system, a life-cycle analysis is performed as shown in figure 9. This figure illustrates accumulated costs over a period of 40 years for the development and utilization of an advanced single-stage-to-orbit vehicle, the space shuttle, and the space shuttle modified for heavy-lift payload capability as compared to just using the shuttle and its derivatives. The change in slope of the costs for the new system is the transition from development to operation. Since the new system has lower recurring costs, a break-even point occurs after about 10 years where the cost curves intersect.

AVTD SYSTEM EVALUATION

Over the past several years, the AVID system has not only been used for the design of launch vehicles but also for orbital-transfer vehicles (ref. 16) and large space structures (ref. 17). Based on the experiences with the AVID system, there are a number of capabilities that have been defined for preliminary computer-aided design.

As stated previously as an advantage to the AVID system, a design system must be highly interactive. Response times must be on the order of seconds for effective involvement with the design process. Specific interactive tasks include geometry generation and manipulation, analysis program selection, design data manipulation, and results interpretation by graphical displays and tables. Analysis programs that execute in seconds are very effective in an interactive design environment. A complete cycle through the launch vehicle analysis programs, as described previously, can take as little as 20 minutes. Thus, many iterations or vehicle enhancements can be completed in a single day. Analysis programs that take minutes or hours to execute can still be used effectively. Before executing these large programs, the input can be thoroughly checked by graphical displays or printouts to reduce the chances of errors. While waiting for the results of these programs, other tasks can be performed. After the large programs have finished, the results can be checked before starting the next design step.

Another advantage of the AVID system is the flexibility in executing independent computer programs in any desired sequence. With the flexibility of analysis selection, the computer can be molded to the design task, not vice versa. Also, engineers want to incorporate their computer programs into the system, since they can easily correct problems and add additional capability.

Geometry is a major component of a computer design system. A single three-dimensional numerical representation--such as coordinates, curve fitting, or surface fitting--is difficult to establish, since each analysis program uses a different type of geometrical numerical model. Mappings from the single numerical representation are needed for each analysis program that uses geometry. The present AVID system uses coordinates that are curve fitted to meet program requirements. A better approach for geometry would be a mathematical representation which is now being studied.

Once the design progresses beyond the conceptual level, large data management requirements emerge. The AVID data management was developed for quick transfer of small amounts of data ($\approx 10^3$ elements). Bulk data such as geometry are handled by external files. Although this system handles the communication of design data between analysis programs, there are several other requirements that would demand the development of a comprehensive data management system.

The first requirement is the efficient handling of large amounts of data. Usually, there are many alternate designs of a baseline concept. Instead of completely storing all the data relating to each design, only the changes from the baseline need to be stored. With this capability, the best features from all versions could be combined to develop a new baseline vehicle.

A second requirement is to describe the vehicle data in a data dictionary. This description would not only include the definitions of the data, but the origin (program or user), the programs that use this data, and the range of values of this data. With this information, the entire data flow can be traced throughout the design system. Data integrity can be checked, and errors in retrieving missing or obsolete data can be resolved.

A third requirement of a data management system is to provide a library of analysis programs and procedures to execute these programs. This library would describe the programs, the input requirements, and the results. By combining the program library with the data dictionary, the input and output data requirements can be determined, and a data integrity system could be developed. Another library feature would be to document new versions of the same program. When data for a concept is being stored, an identifier of the version of each of the programs used during the design cycle should also be stored in order to get data repeatability, if the same design is processed through the system again. If the results do change, at least the reasons for these changes can be traced. The data dictionary and program library can also provide all the necessary information that is needed to interface a new analysis program into the system.

In summary, the AVID system has provided a framework for preliminary design, but major changes in the geometry and data management areas have been identified to provide an improved integrated preliminary design system.

CONCLUDING REMARKS

The AVID system provides a foundation for the analysis and design of aerospace vehicles at the conceptual and preliminary levels. Its main purpose is to provide a system that can gracefully react to design requirements. The AVID system can adapt to changes in the generic class of vehicles by the easy integration of independent programs, changes in mission requirements for a single vehicle by the interactive selection of analysis procedures, and changes in vehicle configuration by interactive manipulation of geometry and analysis input. Innovative concepts, that in the past would be quickly forgotten because of a lack of analysis depth, can now be analyzed with the AVID system by its flexibility in analysis procedure selection and design strategy.

The AVID system was first used for the analysis and design of future launch vehicles to determine the high yield technology areas that are needed for efficient systems to exist in the future. The AVID system has aided in quantifying technology and configuration trades based on studies that included both mass and economic comparisons. Today the AVID concept is also being used for orbital-transfer vehicles, large space structures, and shuttle-derivative studies.

REFERENCES

1. Gregory, T. J.; Petersen, R. J.; and Wyss, J. A.: Performance Trade-Offs and Research Problems for Hypersonic Transports. AIAA Journal of Aircraft, July-August 1965.
2. Space Shuttle Synthesis Program (SSSP), Final Report. General Dynamics Convair Division Report No. GDC-DBB70-002, December 1970.
3. Oman, B. H.: Vehicle Design Evaluation Program (VDEP). NASA CR-145070, January 1977.
4. Garrison, J. M.: Development of a Weight/Sizing Synthesis Computer Program. McDonnell-Douglas Astronautics Company, MDC-E0746, February 1973.
5. Glatt, C. R.; Hague, D. S.: ODIN: Optimal Design Integration System. NASA CR-2492, February 1975.
6. Glatt, C. R.; Hague, D. S.; and Watson, D. A.: DIALOG - An Executive Computer Program for Linking Independent Programs. NASA CR-2296, September 1973.
7. Development of Integrated Programs for Aerospace-Vehicle Design (IPAD), IPAD Executive Summary, D6-IPAD-70020-4, April 1978.
8. Valrand, C. B.: Configuration Design Analysis Program (DATCOM). NASP-12140, September 1972.
9. Mangason, R. J.; and Lamar, J. E.: Vortex-Lattice Fortran Program for Estimating Subsonic Aerodynamic Characteristics of Complex Planforms. NASA CR-26102, February 1971.
10. Centry, A. E.: Hypersonic Arbitrary-Body Aerodynamic Computer Program (Mark III Version). Report DAC 61552, Vol. I, April 1968.
11. Dixon, P. E.: Aerodynamic Analysis System for Conceptual and Preliminary Analysis from Subsonic to Hypersonic Speeds. Presented at the AIAA Aircraft Systems and Technology Meeting, AIAA 80-1897, August 1980.
12. Luscher, W. P.; and Mellish, J. A.: Advanced High Pressure Engine Study for Mixed-Mode Vehicle Applications. NASA CR-135141, January 1977.
13. Brauer, G. L.; Cornick, D. E.; and Stevenson, R.: Capabilities and Applications of the Program to Optimize Simulated Trajectories (POST). NASA CR-2770, 1977.
14. Rehder, J. J.: Parametric Study of Ascent Performance of A Vertically Launched Hydrogen-Fueled Single-Stage Reusable Transport. NASA TP-1045, October 1977.
15. Martin, J. A.: Economic and Programmatic Considerations for Advanced Transportation Propulsion Technology. Journal of Spacecraft and Rockets, AIAA 79-1222R, Vol. 17, No. 5, September-October 1980.

16. Rehder, J. J.: Propulsion Technology Challenges for Future Orbital Transfer Vehicles. Presented at the AIAA/SAE/ASME 15th Joint Propulsion Conference, AIAA 79-1777, June 1979.

17. Leonidis, A.: Large Advanced Space Systems Computer-Aided Design and Analysis Program. NASA CR 152191-1, July 1980.

TABLE I

MASS REPORT

MARTIN TASK II SINGLE-STAGE-TO-ORBIT CONCEPT -- 30K METRIC TON PAYLOAD

WING GROUP			19589.	kg
TAIL GROUP			3162.	kg
BODY GROUP			34216.	kg
BASIC STRUCTURE	10443.	kg		
THRUST STRUCTURE	4889.	kg		
RP-1 TANK	1570.	kg		
LOX TANK	10221.	kg		
LH2 TANK	7092.	kg		
BODY FLAP	1.	kg		
INDUCED ENVIRONMENT			16341.	kg
LANDING GEAR			5294.	kg
PROPULSION			30231.	kg
PROPULSION, RCS			1561.	kg
PROPULSION, OMS			1732.	kg
PRIME POWER			1260.	kg
ELEC CONV AND DISTR			1728.	kg
HYDRAULICS AND SURFACE CONTROLS			4986.	kg
AUTONICS			1984.	kg
ENVIRONMENTAL CONTROL			1755.	kg
PERSONNEL PROVISIONS			763.	kg
MARGIN			9437.	kg
DRY WEIGHT			134038.	kg
PERSONNEL			1220.	kg
RESIDUAL FLUIDS			10092.	kg
LANDED WEIGHT W/O CARGO			145420.	kg
CARGO (RETURNED)			29500.	kg
LANDED WEIGHT			174920.	kg
ENTRY WEIGHT			174920.	kg
ACPS PROPELLANT			15288.	kg
RCS	3621.	kg		
OMS	11667.	kg		
CARGO DELIVERED			0.	kg
ASCENT RESERVES			4658.	kg
INFLIGHT LOSSES			1506.	kg
ASCENT PROPELLANT			1552568.	kg
RP-1	251971.	kg		
LH2	81411.	kg		
LOX	730717.	kg		
LOX	488468.	kg		
GROSS LIFT OFF WEIGHT			1748938.	kg

TABLE II

COSTS FOR A SINGLE-STAGE-TO-ORBIT VEHICLE

MILLIONS OF DOLLARS, ADJUSTED FOR INFLATION FROM 1970 BY 1.50

FIRST UNIT COST COMPLEXITY 1.50			RATE	
	MASS (KG)	COST		
WTNG	19589.8	9.977	FORMULATION AND DEFINITION	15.000
TAIL	3162.0	1.465	AIRFRAME DESIGN, DEVEL	1001.020
BODY BASIC	10443.0	3.530	SUBSYSTEMS DESIGN, DEVEL	282.555
THRUST STRUCTURE	4889.4	4.147	AVIONICS DESIGN DEVEL	150.000
HYDROGEN TANK	7092.0	3.900	ENGINE DESIGN, DEVEL, TYPE 1	500.000
OXIDIZER TANK	10221.0	4.795	TYPE 2	0.000
HYDROCARBON TANK	1569.6	1.497	TOOLING FOR DEVEL	186.306
TPS, 2694.4 SQ M METAL	16340.7	46.284	TEST HARDWARE	412.929
2694.4 SQ M INSUL			NO. GROUND TEST VEH 1.500	
0.500 SQ M/PANEL			NO. FLIGHT TEST VEH 1.000	
100.0 S/SOM INSUL			FLIGHT TEST	17.631
LANDING GEAR	5294.3	2.481	GROUND SERVICE EQUIP	108.249
PROPELLANT SYSTEM	3354.4	6.119	DOCUMENTATION	3.422
CMS	1731.7	7.264	FEE (10%)	264.471
PCS	1561.2	6.876	MANAGEMENT	132.236
ELECT POWER & CONV DIST	2989.3	0.329	FACILITIES	300.000
HVD CONV DIST	2692.9	2.165	TOTAL RATE	3341.418
AERO SIMS CONT	2492.9	3.297		
ENVIRON CONT	1755.0	1.500	PROGRAM INITIATION	
PERSONNEL PROV	763.0	0.605	GROUND SERVICE EQUIP	119.962
FINAL ASSEMBLY		21.246	SPARES	103.967
TOTAL AIRFRAME		127.479	TRAINING	1.500
AVIONICS		15.000	FACILITIES	150.000
ENGINE, TYPE 1, T=2020202.0 N		7.180	MISC EQUIP	0.030
TYPE 2, T=2020202.0 N		5.533	TOTAL	375.460
PRODUCTION			RECURRING COST PER FLIGHT 200 FLIGHTS PER YEAR	
AIRFRAME		438.348	PROPELLANTS	0.565
NO. OF OPER VEH 5.00			VEHICLE MAINT, FACTOR= 0.048	0.177
NO. TRANSFERRED 1.00			TPS MAINT, FACTOR= 0.0048	0.151
LEARNING 0.95			ENGINE MAINT, FACTOR= 0.0110	0.257
AVIONICS		51.579	FACILITIES MAINT	0.030
ENGINE S, TYPE 1, 0.00/VEH		0.000	CREWS, CREW OF 2	0.000
TYPE 2, 5.90/VEH		98.477	TRAINING REPLACEMENT	0.005
LEARNING 0.95			MISC EQUIP REPLACEMENT	0.000
TOOLING		211.346	FEE	0.075
EE		79.975	MANAGEMENT	0.086
TOTAL		879.724	COST PER FLIGHT CONSTANT	0.100
			TOTAL	1.763

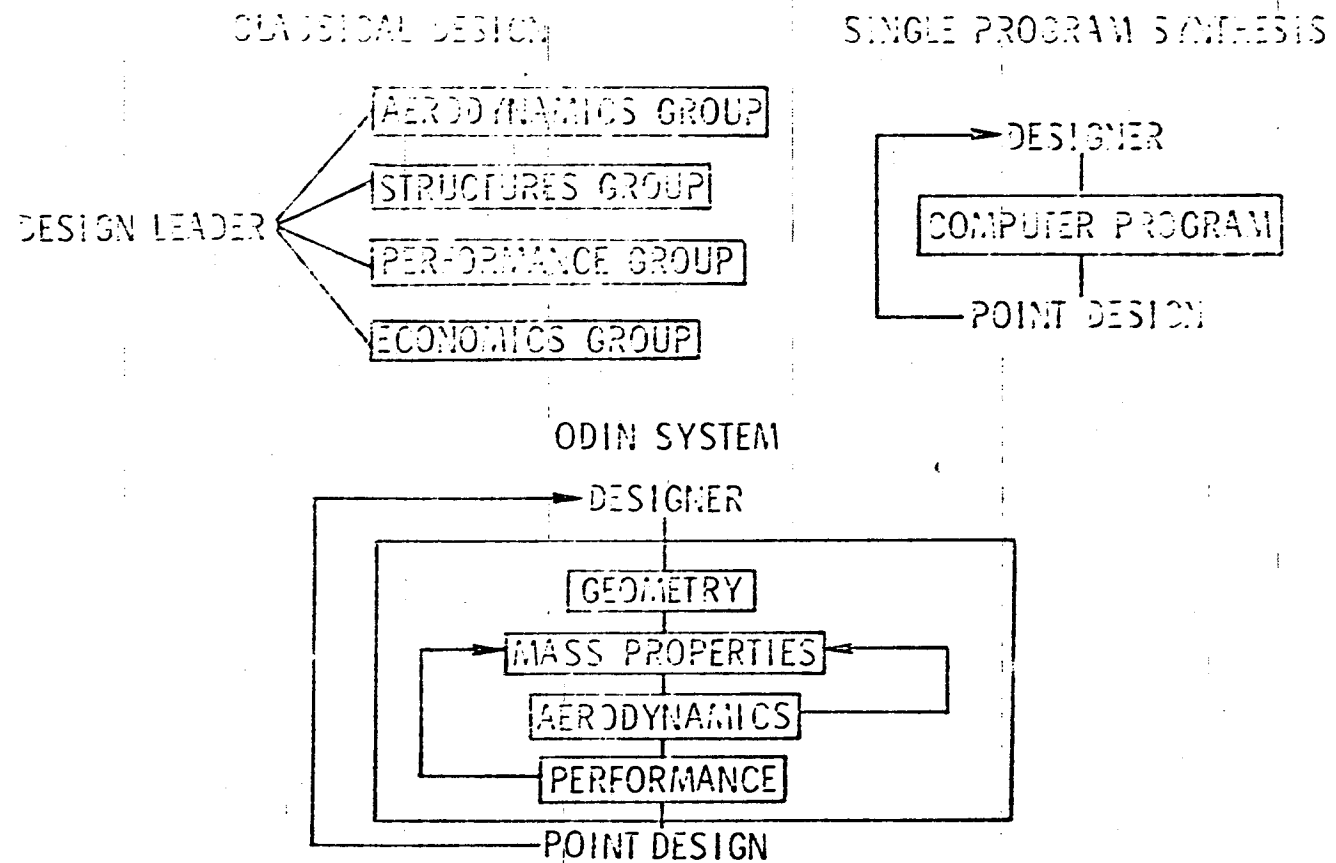


Figure 1.- Approaches to aerospace vehicle design.

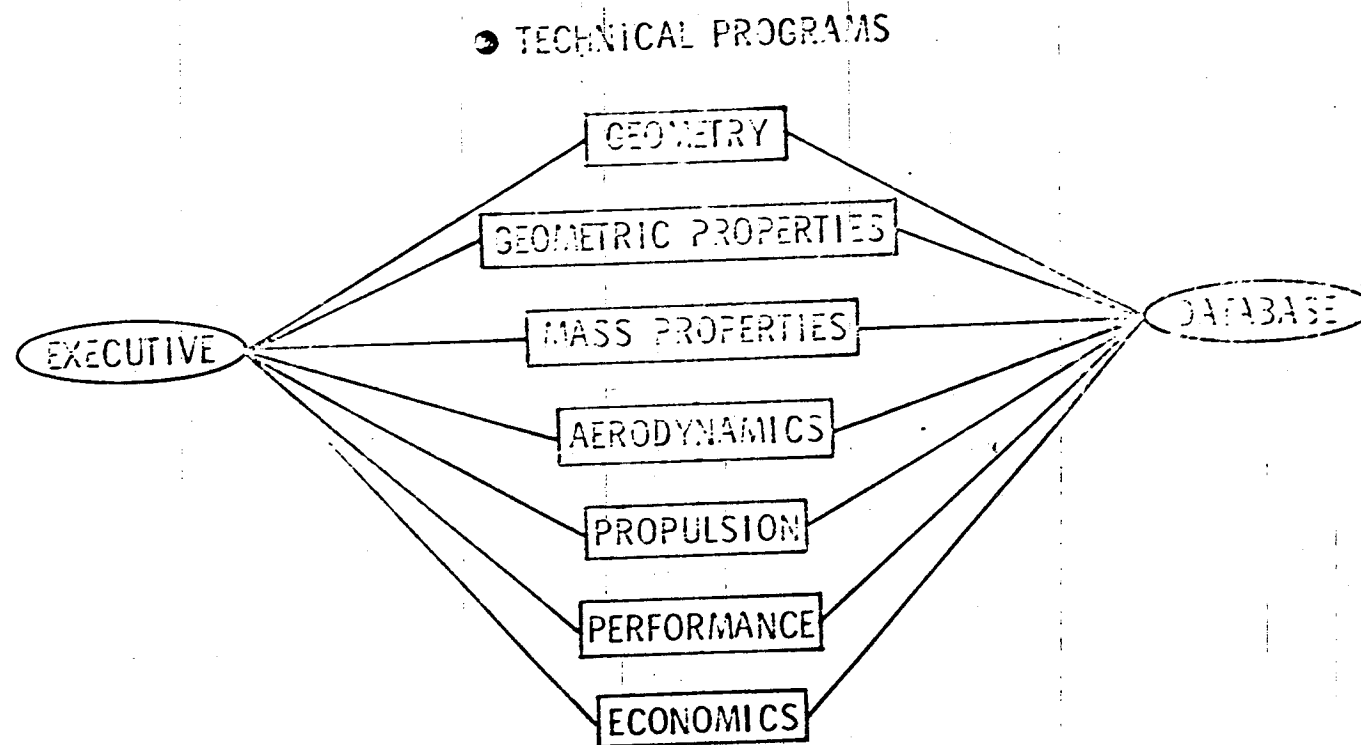


Figure 2.- AVID software system.

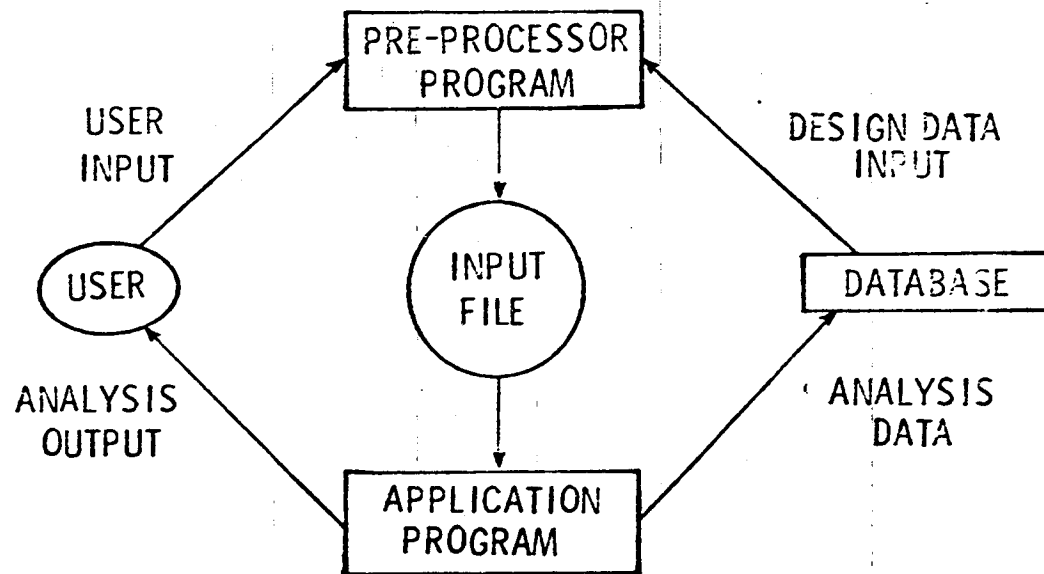


Figure 3.- Interfacing computer programs to the AVID system.

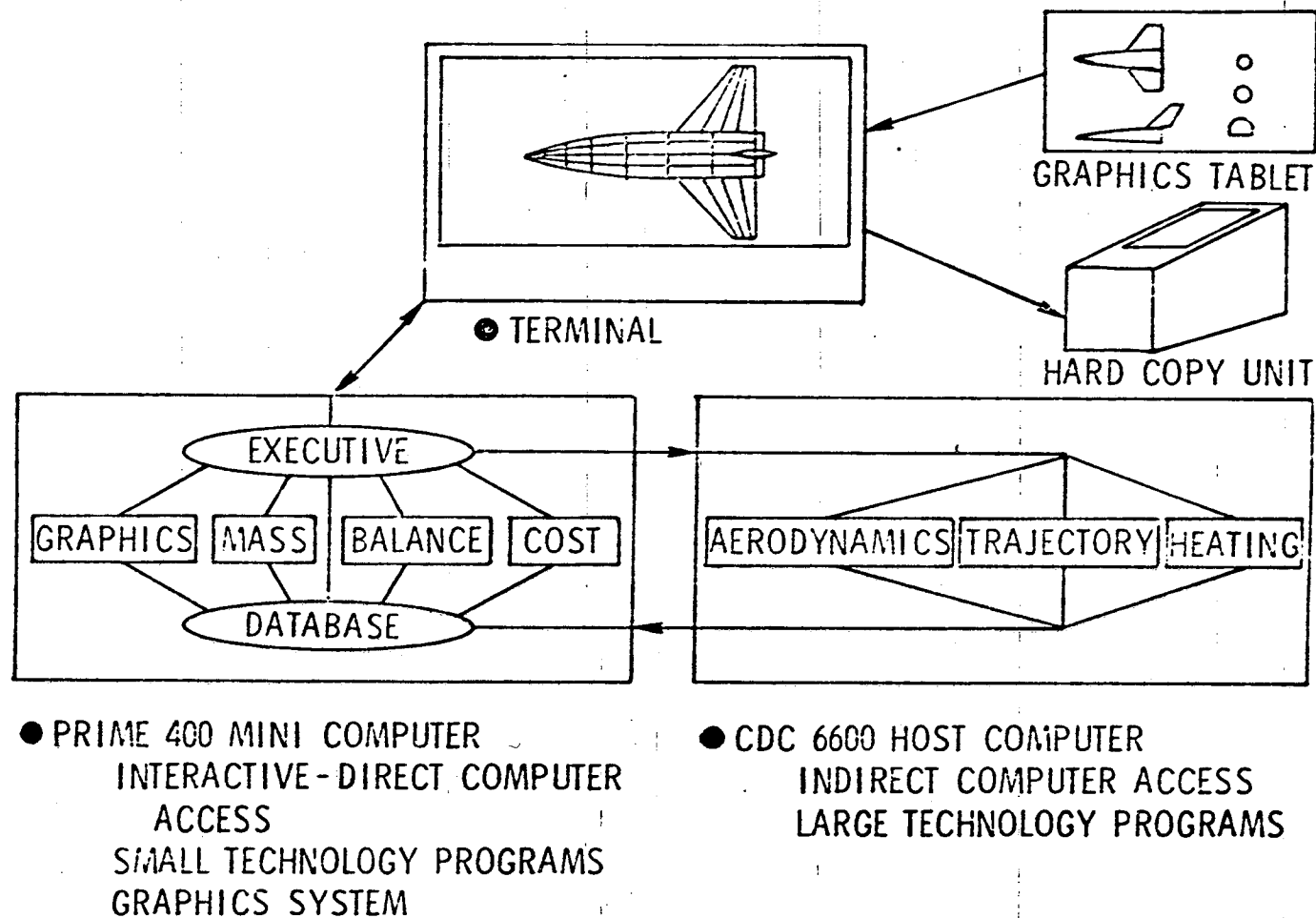


Figure 4.- AVID system hardware.

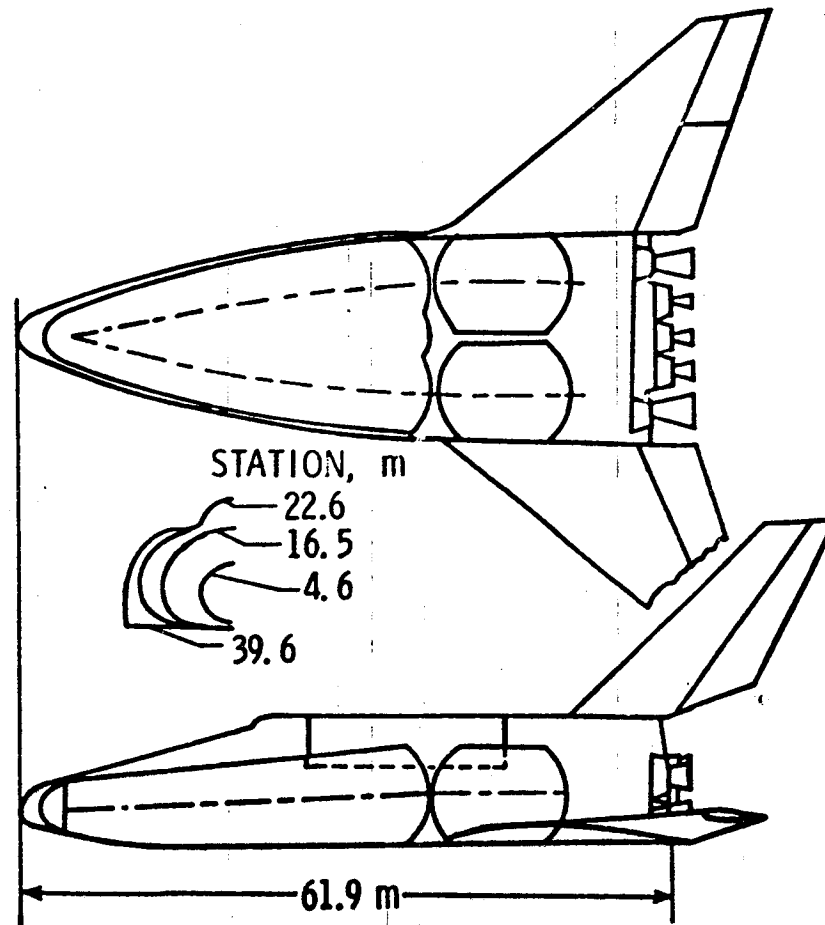


Figure 5:- Sketch of a single-stage launch vehicle.

INPUT VEHICLE VIEW

- 1 - ORTHOGRAPHIC VIEW
- 2 - TOP VIEW
- 3 - SIDE VIEW
- 4 - FRONT VIEW
- 5 - ZOOM
- 6 - FINISHED

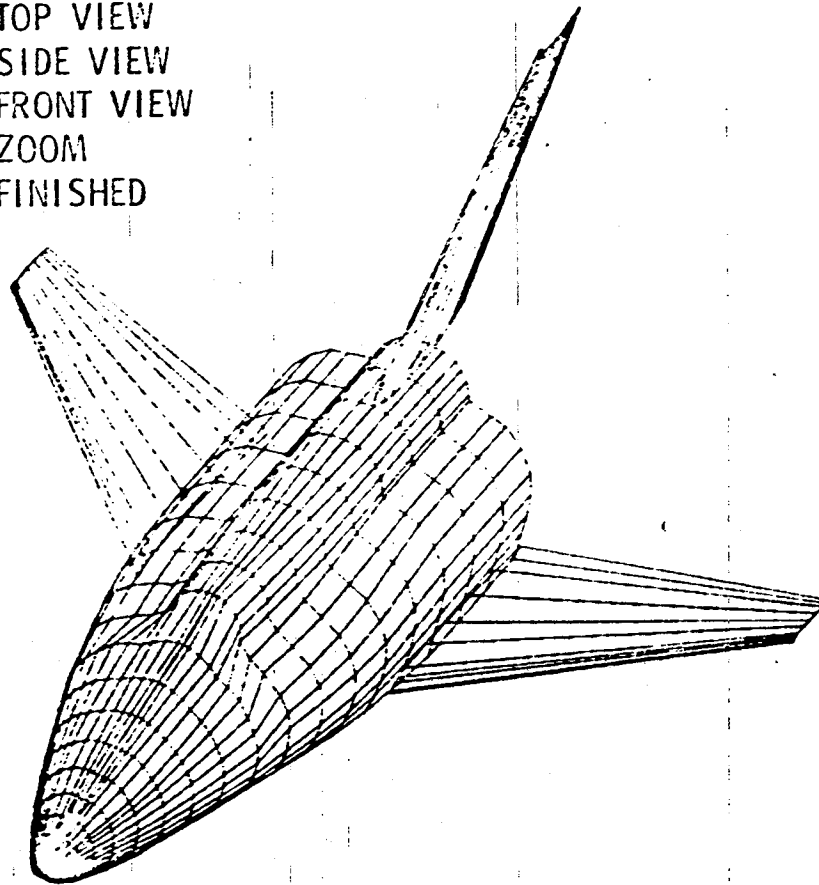


Figure 6.- Numerical model of a single-stage launch vehicle.

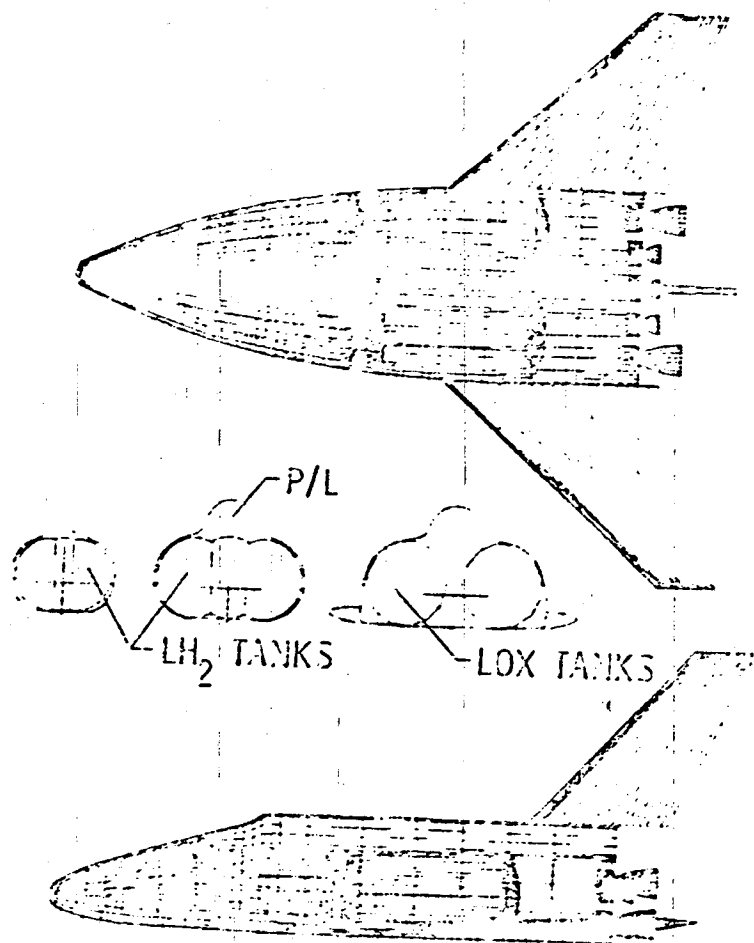


Figure 7.- Numerical model of internal components.

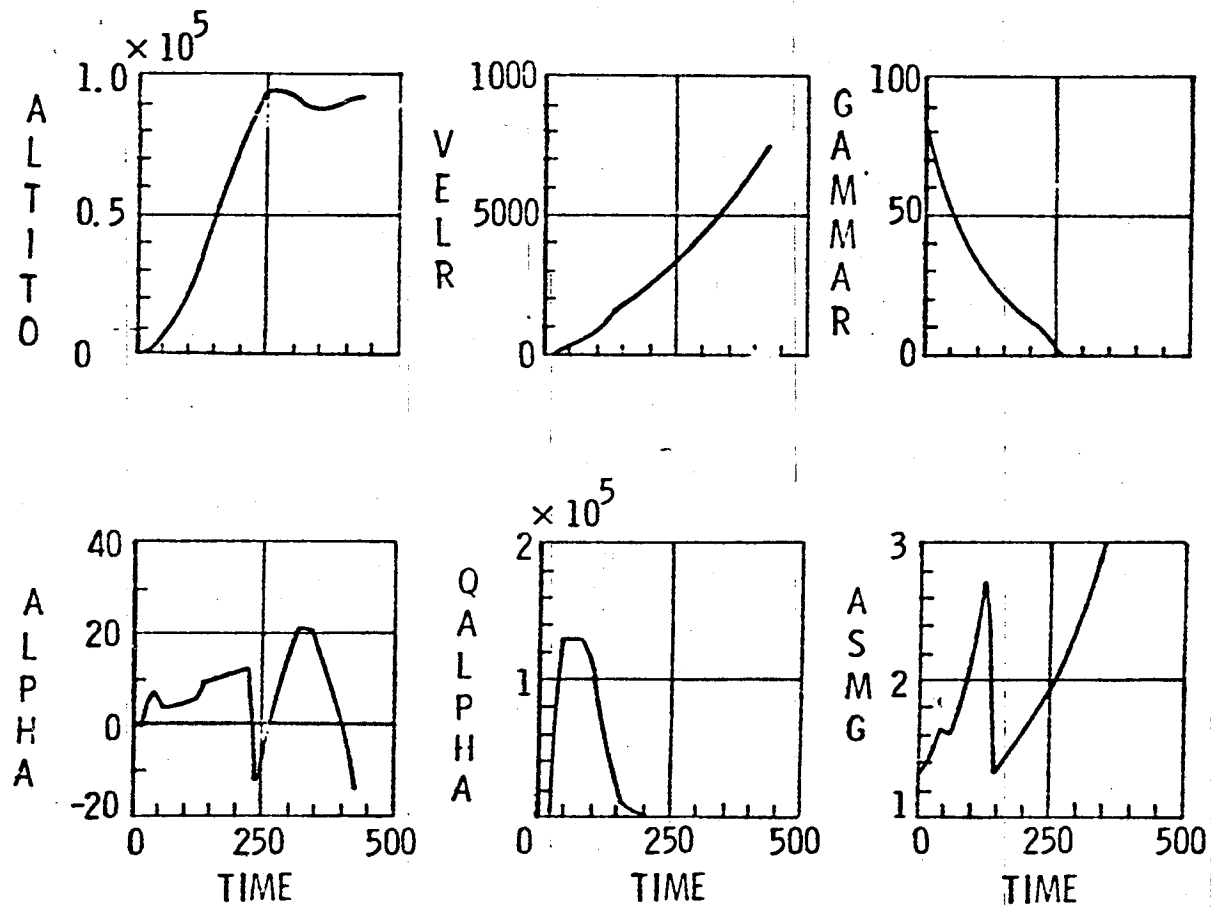


Figure 8.- Trajectory results for a single-stage launch vehicle.

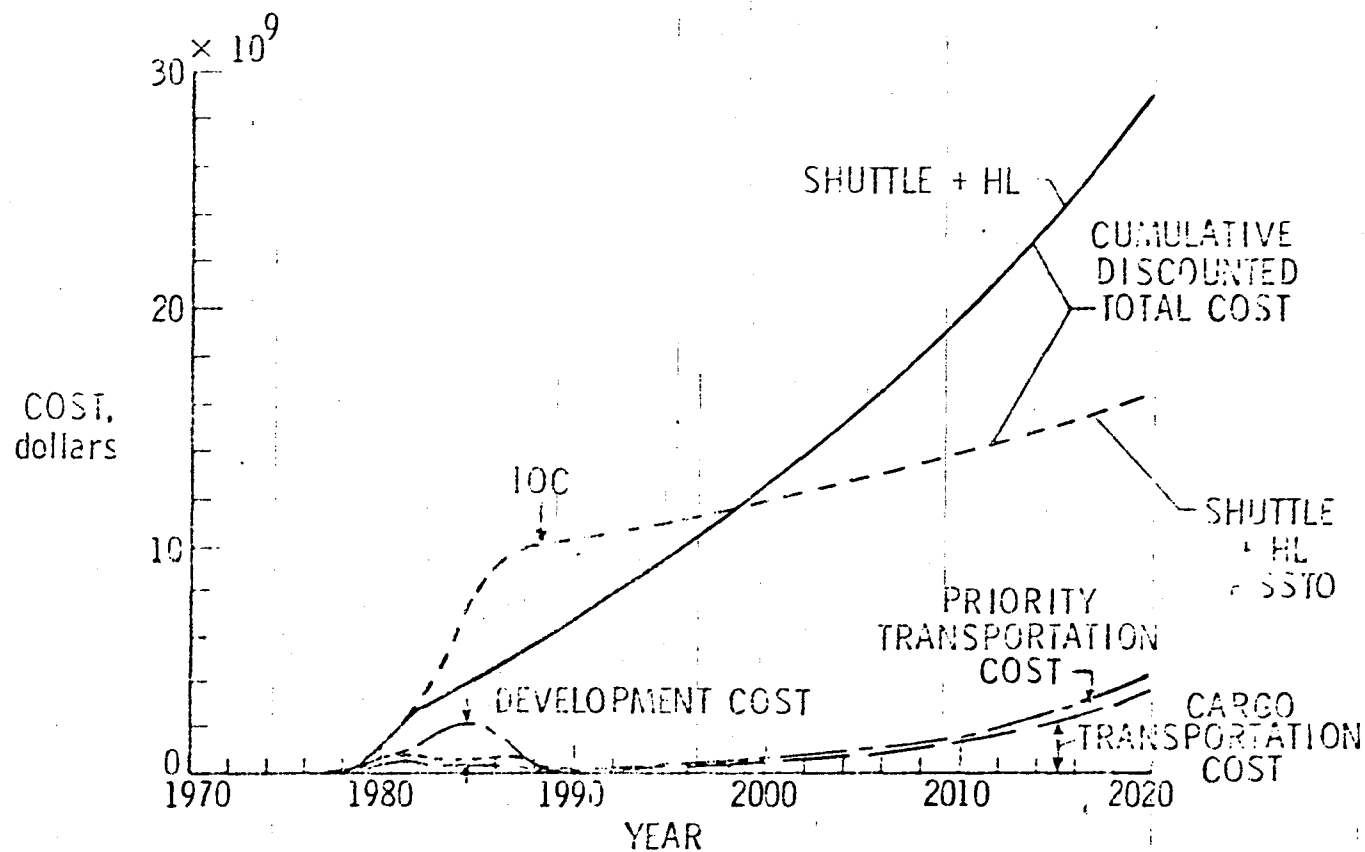


Figure 9.- Shuttle, heavy-lift derivative, and single-stage-to-orbit life cycle costs for 10 percent annual traffic growth.

NASA Technical Library



3 1176 01440 9487