		SAE. AS681	REV. H

SAE *The Engineering Society For Advancing Mobility Land Sea Air and Space*®
INTERNATIONAL®
400 Commonwealth Drive, Warrendale, PA 15096-0001

AEROSPACE STANDARD

Issued 1959-06
Revised 1999-03

Submitted for recognition as an American National Standard

Gas Turbine Engine Steady-State and Transient Performance Presentation for Digital Computer Programs

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

1. SCOPE:

This Aerospace Standard (AS) provides the method for presentation of gas turbine engine steady-state and transient performance calculated using digital computer programs. It also provides for the presentation of parametric gas turbine data including performance, weight and dimensions computed by digital computer programs.

This standard is intended to facilitate calculations by the program user without unduly restricting the method of calculation used by the program supplier.

2. REFERENCES:

2.1   Applicable Documents:

The following publications form a part of this document to the extent specified herein. The latest issue of SAE publications shall apply. In the event of conflict between the text of this document and references cited herein, the text of this document takes precedence. Nothing in this document, however, supersedes applicable laws and regulations unless a specific exemption has been obtained.

2.1.1   SAE Publications: Available from SAE, 400 Commonwealth Drive, Warrendale, PA 15096-0001.

AS755	Aircraft Propulsion System Performance Station Designation and Nomenclature
ARP1210	Gas Turbine Engine Interface Test Data Reduction Computer Programs
ARP1420	Gas Turbine Engine Inlet Flow Distortion Guidelines
ARP4191	Gas Turbine Engine Performance Presentation for Digital Computer Programs Using FORTRAN 77
ARP4868	Application Program Requirements for the Presentation of Gas Turbine Engine Performance on Digital Computers. (when published)

2.1.2   International Organization for Standardization (ISO): Available from National Institute of Standards and Technology, Public Inquiries, Administration Bldg. A903, NIST, Gaithersburg, MD 20899-0001.

ISO 1000	SI units and recommendations for the use of their multiples and of certain other units
ISO 2533	ISO Standard Atmosphere.
ISO 2955	Representations for SI and other units to be used in systems with limited character sets

2.1.3   Other Documents:

NIST-JANAF Thermochemical Tables, Fourth Edition, Parts I and II, Malcolm W. Chase, Jr., Editor, August 1998, ISBN: 1-56396-831-2

2.2   Definitions:

STEADY-STATE PROGRAM: Steady-state engine programs predict engine performance for steady-state conditions (time derivatives are zero). Steady-state engine performance programs in this standard are confined to four categories: Preliminary Design Programs (also called Study Programs), Specification Programs, Status Programs, and Parametric Programs.

PRELIMINARY DESIGN PROGRAM (also STUDY PROGRAM): Preliminary design programs estimate the engine performance until the engine is defined by a specification. A typical application of a preliminary design program is to provide data to support preliminary aircraft design.

SPECIFICATION PROGRAM: Specification programs establish the guaranteed performance for a specific engine model. A typical use of specification programs is to generate performance data for engine and aircraft guarantees.

STATUS PROGRAM: Status programs are created during engine design and development to reflect the current status of engine performance. They are normally based on greater test experience than a specification program. Status programs do not alter the commitments established by specification programs, but indicate the current status of meeting those commitments. Applications of status programs include:

(1) prediction for an engine test or as a baseline for comparison with test data reduction output,
(2) modeling an engine to investigate test operating procedures for ground test and/or flight test,
(3) extrapolation and/or interpolation of limited test data.
(4) generation of data for use in the construction of simulators

PARAMETRIC PROGRAM: Parametric engine programs are a special type of preliminary design program. They estimate the performance and possibly other figures of merit (e.g., weight and dimensions) of a class of engines of similar configuration over a range of design parameters (e.g., compressor pressure ratio, turbine inlet temperature, airflow). The performance, weight, and dimensions may be derived from a less detailed analysis than a conventional preliminary design, specification, or status program. A typical application of a parametric engine program is to provide data to narrow the range of system design parameters for more detailed studies.

2.2   (Continued):

TRANSIENT PROGRAM: Transient engine programs predict engine performance under time-varying conditions. Transient programs may simulate the engine without a control, or they may simulate the engine and its control system. The categories of transient engine performance programs mirror the steady-state performance categories: preliminary design, specification, status, or parametric with corresponding definitions. Applications of transient programs include:

(1) prediction of engine acceleration and deceleration performance,
(2) support of airframe/engine integration studies, including flight control development,
(3) generation of data for use in the design of flight simulators,
(4) simulation of the engine during checkout of the control in control benches.

DATA REDUCTION PROGRAM: Data reduction programs are used to interpret results from performance testing. A data reduction program may directly compute component performance characteristics (e.g., efficiency) from measured data using standard engineering equations. Alternatively, a data reduction engine program may employ a steady-state or transient engine program as its core, and extend its iteration scheme to match component parameters to engine test data. In the latter case, the predictive capability of the engine model can be used to extrapolate the test results to the desired test condition (which may be unattainable). The engine model used as the heart of the data reduction program will nearly always be a status model as defined above. Applications of data reduction programs include:

(1) interpretation of test data to facilitate the generation of status models,
(2) analysis of performance verification tests,
(3) interpretation of engine/airframe performance issues based on aircraft flight tests.

ARP1210 provides additional information on data reduction programs.

NOTE: Providers of engine performance computer programs may combine two or three of the program categories (steady-state, transient or data reduction) within a single delivered program.

3. GENERAL REQUIREMENTS:

This aerospace standard describes a class of software whose audience is one or a few specific users. It offers standards or recommended practices for the delivery of that software. However, there is every opportunity for the developer and the user to meet to establish the specific needs for that software. This standard provides a basis for identifying the issues of that discussion. Elements of this standard may only be waived by mutual consent of the supplier and the user. Failing agreement to a waiver by both parties, this standard establishes the requirements for the delivered program.

3.1   Program Operation:

3.1.1   The program will be supplied in a form adaptable to the user's computer. Provision to operate the program on more than one computer type or operating system will be coordinated between user and supplier.

3.1.2   The program will be able to function as a subprogram (herein called the engine subprogram) and as an independent program consisting of a main program which calls the engine subprogram.

3.2   User Documentation:

3.2.1   The program supplier will deliver user documentation with the program. User documentation may be in the form of paper manuals, electronic media, on-line help or remotely accessible instructions (e.g., via the Internet). User documentation will normally be completely independent of documentation for earlier releases of the program. With the explicit approval of the program user, it is permitted to provide an update to prior documentation.

The program supplier is urged to separate the documentation that describes installation and checkout of the program from the documentation that describes program operation and interpretation of program output.

3.2.2   The "Installation and Program Checkout" documentation will include the following minimum information.

Program Description: This section will describe the required computer hardware (computer model, minimum memory, disk space, etc.), software environment (operating system, compiler, etc.), programming practices, and software quality assurance standards upon which the program relies.

Program Setup: This section will include instructions and information to enable the program users or the users' computer staff to set up the program. It will include an overall program structure flowchart and sufficient information to set up and execute the program in independent mode or as a subprogram. This information should be provided in paper documentation and may also be included in an on-line help system. An example flowchart for a user calling program is included as Appendix A.

Listings: Source code listings of the calling program and of the input and output statements used directly by the calling program will be supplied with the computer program. The supplier is urged to include descriptive comments in the source code listings to facilitate understanding.

Test Cases: This section will describe the test cases that the supplier provides for program checkout by the user. All test case inputs and associated results including reasonable bounds for variance of key parameters will be supplied.

3.2.3   The "Program Operating Instructions" documentation will include the following minimum information. Additional requirements for specific model types are given elsewhere in this standard.

Introduction: This section will provide an overview of the engine model, and of the gas turbine engine being modeled. It will describe the intended uses for the model. It will document the nature of, and reason for, revisions to engine computer programs from earlier versions.

Table of Contents: This section should list all documents delivered with the engine program and list the contents of each document. The on-line equivalent should list all major topics included in the on-line help system.

Engine Description: A description of the engine will be provided, as represented by this engine program, and any other information to understand the performance and operation of the engine. This must include specification of the "performance level" of the engine model (e.g., minimum new engine, average engine).

Nomenclature: This section describes the rules of nomenclature used in the model and provides an engine cross-section to illustrate station identification.

Input/Output: This section will include information required to understand the user documentation and program input and output (see AS755). All program inputs and outputs will be described in sufficient detail to avoid ambiguity, including their units of measure.

Engine Program Description: This section will describe the interfaces, options, limitations, and other features of the program, to enable the user to fully understand the capability of the program.

Program Messages: This section will explain the messages produced during program operation and define all Numerical Status Indicator (NSI) codes.

Identification and Revision Procedure: This section will provide the unique identification of the program, an associated date, and other information needed to understand the relationship of this engine program to earlier releases.

References: Include details of any citations called out in the document.

Index: Although not mandatory, an index is highly desirable (or a search engine for on-line help systems).

3.2.4   Data Packs/Card Packs: Engine programs can include a mechanism to allow modification of the performance representation (or other program features) using so-called data packs or card packs. This often eliminates the need to deliver a completely new engine program. Whenever an engine program provides such a capability, its use is to be described in the user documentation. Further, whenever a new data pack or card pack is issued, associated documentation will be provided to assist users in installation, checkout and interpretation of program results with the data pack/card pack. All items listed in paragraphs 3.2.2 and 3.2.3 are to be addressed except for those that obviously do not apply.

3.3   Program Scope:

3.3.1   Engine Limits and Program Limitations: Engine limits are those conditions beyond which the engine is not intended to operate. Engine limits observed by the computer program will be specified in the user documentation including any functional dependence on other parameters.

When input to the engine program calls for operation beyond engine limits, the program will perform the calculation, if possible, and include an appropriate numerical status indicator in the output. The numerical status indicator will clearly define output validity, limiting parameters and/or the applicable limiting engine controlling variable.

Program limitations define the bounds within which the program will function and still adequately simulate the engine (these need not coincide with the engine limits). Program limitations will normally exceed the corresponding engine limits, however, more restrictive program limitations may be used based on agreement between the program supplier and user.

3.3.2   Consistency: Consistency is used here to mean the agreement between transient model outputs and reference data. Reference data may be a more rigorous model or test data. Consistency criteria vary significantly and depend upon the specific application. The coordination of model consistency is a key element of model delivery.

3.3.3   Validity: The boundaries within which the engine program is valid will be described in the user documentation.

3.3.3.1   When test data is limited, the valid range of an engine program may only be a part of the complete flight envelope. Any such limits of program validity will be identified in the user documentation.

3.3.3.2   When test and design data are limited, the range of a parametric engine program will generally be a subset of the full flight envelope and power level range. Also, capabilities such as anti-ice bleed, distortion effects, windmilling, etc., are often omitted in these programs. Such limitations will be described in the user documentation. Life and duty cycle assumptions will also be clearly described in the engine description section of the user documentation to aid interpretation of the predicted performance, weight, and dimensions.

4. PROGRAMMING PRACTICES:

4.1   Computer Capabilities:

As engine performance programs increase in size or complexity, they place greater demands upon their computer hosts. The program execution time will depend on the inherent capability of the host computer, the complexity and level of detail embedded in the model, and the extent of the studies being accomplished using the engine program. These issues need to be understood by both the program supplier and the user so that the program delivers acceptable computing performance on the user's computer system.

4.2   Program Language:

Any ANSI standard computer language (for instance, C, Ada, FORTRAN 77) may be used for engine computer programs, provided the language has been agreed to by the user and the supplier. Languages for which an ANSI standard is not available may also be used, but careful coordination between the supplier and the user regarding language issues (such as compiler choice and specific features to be used) is strongly recommended. If not otherwise coordinated, engine computer programs will use FORTRAN 77 as a default language.

ARP4191 describes the use of FORTRAN 77 and FORTRAN COMMON blocks to manage data communication between user and supplier. The approach of ARP4191 is, in essence, the standard practice used during the last 30 years.

ARP4868 describes the use of a "function-call application program interface" (API) as an alternative to the position-dependent program interface method of ARP4191. The ARP4868 API approach may be implemented using FORTRAN, or in newer languages, such as Ada, C or C++.

Both ARP4191 and ARP4868 describe interface protocols commonly used and well understood throughout the gas turbine industry. Engine computer program developers who elect to use a language not described in these practices are expected to comply with the requirements of AS681, and must pattern their implementation after ARP4191 or ARP4868, choosing the recommended practice that most closely matches their selected program interface strategy.

4.3   Precision:

Hardware and software differences (e.g., word length, compiler, operating system) between computers can result in differences in performance output from otherwise identical programs. Differences in output of steady-state programs from the supplied test cases greater than 0.25% should be reviewed with the program supplier. For transient programs, differences smaller than ±1% at a given output time are considered normal for engine transients of 10 seconds or less. Larger differences should be reviewed with the program supplier.

4.4   Station Identification:

The numbering system described in AS755 will be used in program input and output to identify points in the gas flow path significant to engine performance definition. Supplements to AS755, deemed necessary by the program supplier, will be detailed in the user documentation.

4.5   Nomenclature:

4.5.1   Modern programming languages and operating systems permit use of fully descriptive labels for program variables and parameters. Engine programs are permitted, but not required, to use these descriptive labels for variable identification. However, there will continue to be situations where abbreviated symbols will be desirable (either for economy or because the computer environment precludes use of long strings).   In cases where symbols are to be substituted for fully descriptive labels, the list of symbols contained in AS755 will be used to identify input and output parameters of the engine program. AS755 does not require use of specific symbols within the engine subprogram.

4.5.2   The following symbols, which refer specifically to computer programs, are added to those of AS755.

      a. CASE   Numerical case identification
      b. CCLASS   Engine program security classification
      c. CIDENT   Engine program identification
      d. NIN   Input File Number
      e. NOUT   Output File Number
      f. NSI   Numerical status indicator
      g. PC   Power code
      h. RC   Rating code
      i. SDIST   Inlet pressure and temperature distortion selection
      j. SERAM   Ram pressure recovery selection
      k. SIM   Inlet mode selection
      l. SWIND   Windmilling performance selection
      m. CTITLE   Title supplied by program user
      n. Z (leading)   Prefix for a variable name, used in input, that may also be used in output.

4.5.3   Some compilers limit the number of characters per parameter label. When such a limitation precludes compatibility with 4.5.1 and 4.5.2, each parameter label will be determined by the program supplier and defined in the appropriate section of the user documentation. Included as part of the definition, the program supplier will indicate the standard parameter name per AS755 or AS681.

4.6   Power Definition:

Power level and other performance parameters for the significant engine ratings (e.g., takeoff, intermediate) will be available from a steady-state engine program by simple input (the user will be able to request these significant engine ratings without needing to know the intricacies of power management).

The power level selection will be implemented using the following input hierarchy: Power Lever Angle (PLA), Power Code (PC), Rating Code (RC), and any additional performance selection. The user documentation will describe the selection process.

4.6.1   Power Lever Angle or Power Code: The Power Code is defined as follows:

4.6.1.1   When positive, the power code will represent the power level specification. When power lever angle is known, it should replace the power code. Positive Power Code values are assigned as shown in Table 1. Many engines will have power levels covering emergency ratings and/or water injection. The value of Power Code and Rating Code in the range 51-59 may be used to define these power levels while maintaining consistency with Tables 1 and 2.   Unassigned power codes will be coordinated between the program user and supplier and defined in the user documentation.

TABLE 1 - Power Code Assignments

	Power Code	Definition
Augmented	100.0 to 60.0	Maximum to Minimum
Nonaugmented	50.0 to 20.0	Maximum to Idle
Reverse	15.0 to 5.0	Idle to Maximum

4.6.1.2   When Power Code has an input value of zero, the input value of Rating Code will be the basis for the calculated performance. Rating Code definitions are given in Table 2.

TABLE 2 - Rating Code Assignments

	Rating Code	Definition Military	Definition Commercial
	100.0	Maximum	Emergency
Augmented	90.0	Maximum Continuous	Maximum
	60.0	Minimum	Minimum
	55.0	---	Wet Takeoff
	55.0	Maximum	---
	50.0	Intermediate	Dry Takeoff
	45.0	Maximum Continuous	Maximum Continuous
Nonaugmented	40.0	---	Maximum Climb
	35.0	---	Maximum Cruise
	21.0	Flight Idle	Flight Idle
	20.0	Ground Idle	Ground Idle
Reverse	15.0	Idle	Idle
	5.0	Maximum	Maximum

4.6.1.3   A negative input value of Power Code will be used to select alternate methods of power level specification (e.g., iteration to a specified net thrust or selection of a specified series of power settings). The options available using negative Power Codes will be fully described in the user documentation.

4.6.2   Rating Code: The Rating Code permits selection of specified ratings that may require different Power Lever Angles as flight or atmospheric conditions vary.

Rating Codes are defined in Table 2. Unassigned Rating Codes may be used for additional ratings which should be coordinated between program user and supplier. Additional Rating Code options will be defined in the user documentation.

4.6.3   Additional Performance Selection: Additional performance selection may be required for unusual engine operating conditions (e.g., SWIND).

4.7   Standard Thermodynamics:

The following references will be used for engine performance determination:

4.7.1   Standard Atmosphere: Ambient temperature and pressure in engine performance programs will be consistent with the geopotential pressure altitude of ISO 2533.

4.7.2   Standard Dry Air: The composition of standard dry air is to be based on that used for ISO 2533, but with the number of molecular species reduced by the reallocation of trace elements. Upon request by the program user, the composition will be included in the user documentation.

4.7.3   Fuel: Fuel Heating Value assumed by the computer program will be specified in the user documentation. Fuel Specific Gravity will also be specified if it is used in the performance calculation or in an associated control model. Other fuel characteristics used in the program will be provided in the user documentation, if specifically requested by the program user.

4.7.4   Thermochemical Data: Thermochemical data used in the computer program will be provided in the user documentation, if requested by the program user. Thermochemical data, for the purpose of gas turbine engine performance calculations will be consistent with data presented in the NIST-JANAF Thermochemical Tables, Fourth Edition, Parts I and II, within the limits of physical representation (e.g., reaction and heat transfer rates), engineering accuracy, and computational efficiency requirements. Upon request by the user, the supplier will provide a comparison of the thermodynamic process calculations based on his thermochemical data and that given by the NIST-JANAF Thermochemical Tables, within the range of applicable engine operating conditions.

4.8   Programming Standards:

4.8.1   The choice of compiler optimization should be coordinated between user and supplier. The highest level of compiler optimization available to the program supplier is preferred, to provide rapid computation, although the supplier should be careful to insure that errors are not introduced as a result.

4.8.2   The program supplier will provide automatic preventative action for the following illegal arithmetic operations or processes:

    a.   The square root of a negative number
    b.   Illegal arguments to exponential functions
    c.   Illegal arguments to logarithmic and inverse trigonometric routines
    d.   Division by zero

This preventative action will preclude these illegal arguments from being transferred to the user's system supplied routines. When any of these conditions renders the case invalid an appropriate Numerical Status Indicator will indicate the specific problem(s) encountered. The program will always be capable of continuing with the next case provided the user's program does not override this capability.

4.8.3   The program supplier will assume that the computer memory is NOT cleared when the program is loaded.

4.8.4   The engine program, while operating in subprogram mode, will be available as one call, rather than separate calls to a collection of routines which, when taken in combination, represent the engine.

5.   PROGRAM CAPABILITIES:

Features described in this section will be included when they apply.

5.1   Time Interval Control (Transient Programs):

It is particularly important that there be close coordination between the program supplier and the program user for transient programs. The paragraphs that follow indicate some of the issues requiring discussion, and some of the more common practices for transient models.

5.1.1   An input value of time, ZTIME, will be used to inform the engine subprogram of the value of time from the start of the transient case at which it is to return control to the calling program. Control will also be returned if certain values of Numerical Status Indicator are created. Engine subprogram output data will be available at each return.

5.1.2   A steady-state solution providing initial conditions for the transient will be generated whenever the input value of time, ZTIME, has a value of zero.

5.1.3   TIMEO will be used to define the time interval between generated outputs. ZTIMET will be used to define the termination time of a given transient case.

5.1.4   It may be necessary for the user to iterate between a user program and the engine subprogram. In this case, repeated input to the engine subprogram for the same value of time (ZTIME) may be required, until convergence of the iteration is achieved. In these situations, both the program user and the program supplier need to have a thorough understanding of the user's requirements, and agree to the methods used to achieve these requirements.

5.1.5   Both program user and supplier need to understand the user's frequency requirements. The engine subprogram must contain logic to ensure that its internally-selected time step is adequate to properly model the frequency range required by the user.   Failure of the engine subprogram to converge can also be caused by severe ramps or step changes of input. If such changes are anticipated, coordination between program user and supplier is essential.

5.1.6   The program may have the capability for the user to change the input frequency from an initial value (FYPH) to a secondary value (FYSH) and back, once during the transient. TIMEF1 is the time at which frequency is changed from FYPH to FYSH and TIMEF2 is the time at which frequency reverts to FYPH.

5.2   Variable Input (Transient Programs):

5.2.1   When the program user is required to perform studies that vary an input parameter with time, the responsibility to provide this capability belongs to the user.

5.2.2   Provision to schedule an input parameter against internal engine parameters (e.g., distortion parameters versus intake mass flow, power extraction versus shaft speed) shall be the responsibility of the program user. The program user will obtain the internal engine parameters required for these calculations from the engine subprogram output.

5.2.3   In some cases, the transient engine program operates as a subprogram and its time increment differs from the output time increment (TIMEO) required by the user's program. When this is required, it is recommended that the user's and supplier's programs be synchronized. -- meaning that the ratio of the two time increments be an exact integer value.   Asynchronous operation will require complex interface schemes that use interpolation and/or extrapolation.

5.3   Inlet Representation:

5.3.1   Inlet Mode: The engine program will calculate performance when provided with any of the
following combinations:

1.   (SIM = 1) Altitude, free stream Mach number, deviation of ambient temperature from standard,
and ram pressure recovery (one of the options of 5.3.2).

2.

(SIM = 2) Ambient pressure and temperature, total pressure and total temperature at the inlet/
engine interface, and a temperature increment for inlet heat transfer to be added to the inlet
total temperature (T1A = T0 + DT1A).

3.   (SIM > 2) Other options, as coordinated between user and supplier.

5.3.2   Ram Pressure Recovery: Several options for ram pressure recovery will be provided (identified by
the ram pressure recovery selection, SERAM). Two categories of options are provided. The first is
used for engine inlet average recovery. The second provides the ability to differentiate primary and
secondary stream recoveries. The specific options are defined in Table 3.

TABLE 3 - Ram Pressure Recovery (SERAM) Assignments

SERAM	Definition
Average Recovery Options	
1	A specified ram pressure recovery standard.
2	The input ratio of engine inlet total pressure to free stream total pressure.
3	The ratio of engine inlet total pressure to free stream total pressure (ERAM) as a function of referred engine airflow (WR*) and/or free stream Mach number (XM).
Differentiated Primary and Secondary Stream Options	
4	The input ratio of primary stream engine inlet total pressure to free stream total pressure; and the input ratio of secondary stream engine inlet total pressure to free stream total pressure.
5	The input ratio of primary stream engine inlet total pressure to free stream total pressure; and the ratio of secondary stream engine inlet total pressure to free stream total pressure as a function of referred engine airflow (WR*) and/or free stream Mach number (XM).
6	The ratio of primary stream engine inlet total pressure to free stream total pressure as a function of referred engine airflow (WR*) and/or free stream Mach number (XM); and the ratio of secondary stream engine inlet total pressure to free stream total pressure as a function of referred engine airflow (WR*) and/or free stream Mach number (XM).

5.3.2   (Continued):

SERAM options 4, 5, and 6 are intended for use with bypass engines.

SERAM options 3, 5, and 6, which provide for a functional relationship between ram recovery and engine airflow and/or free stream Mach number, require a user supplied subprogram with arguments, XM, WR*, SECALC and ERAM, where WR* denotes W1R or W1AR as appropriate.

The user supplied subroutine will interpret the value of SECALC to return the values of ERAM as presented in Table 4.

TABLE 4 - Ram Pressure Recovery Calculation Options

Option (SECALC)	Ram Recovery (ERAM)
1.0	Average
2.0	Primary Stream
3.0	Secondary Stream

5.3.3   Distortion: The effects of inlet pressure and temperature distortion (radial, circumferential, and temporal) on the engine may be included in the program when data or estimates are available. ARP1420 should be consulted for details.

5.4   Customer Services:

5.4.1   Bleed Air: The engine program will calculate performance when air is extracted for customer services. The user will be able to specify the amount of air extracted from each bleed station as a flow rate or, where applicable, as a ratio to the component inlet gas flow rate. If the user specifies non-zero values for both rate and ratio of the bleed air the resulting total bleed flow is the sum of the two specified values (subject to override by program limits). The corresponding output temperatures and pressures will include engine induced heat transfer and pressure losses up to the bleed port interface.

This requirement is waived for preliminary design programs, which often do not include modeling of customer bleeds.

5.4.2   Power Extraction: The engine program will calculate engine performance when power is extracted.

5.5   Engine Supplied Nozzle Effects:

5.5.1   Nozzle parameters used in the calculation of engine performance will be included in the program output. The nozzle parameters will include:

   a. Nozzle area(s)
   b. All gas flows
   c. All engine discharge gas temperatures and pressures
   d. Ideal gross thrust
   e. Required user-supplied nozzle cooling airflow

5.5.2   When a separate subprogram, or supplementary data are needed to define external geometry of an engine supplied nozzle for user performance calculations, the appropriate data will be included in the program output.

5.6   Parasitic Flows:

Parasitic flows include planned flows, other than the customer bleed flow, discharged from the engine at points other than the exhaust nozzle. The engine program will calculate the effect on engine performance of such planned flows, including the total pressure, total temperature, and rate of flow for each parasitic flow in the program output.

5.7   Anti-Icing:

The engine program may have the capability of calculating engine performance when operation involves the use of inlet and/or wing anti-icing.

5.8   Fluid Injection:

When engine operation allows the use of fluid (e.g., water/alcohol) injection for augmentation, the engine program will compute engine performance for such operation. The injected flow rate used in the performance calculation will be included in the program output.

5.9   Fuel Properties:

When the engine program allows the user to vary fuel properties (lower heating value, specific gravity), the following restrictions apply:

   a. The range of fuel property values is limited to that of the fuels with which the engine can be operated.

   b. The allowable variation in fuel properties for a specific fuel will be based on requirements set forth by appropriate agencies.

5.10 Windmilling:

The engine program may offer prediction of performance under windmilling conditions. If so, the user documentation will specify the conditions (e.g., altitude versus Mach number regime) for which program output is valid.

5.11 Reverse Thrust:

The engine program may include prediction of the performance of the engine with an attached thrust reverser or spoiler. If so, the user documentation will specify the flight conditions and engine limits (e.g., power setting) for which program output is valid.

5.12 Variable Control Schedules:

If the engine program includes variable control schedules that can be scheduled by the user, a physical and functional description of the control schedule options will be provided in the user documentation. The engine program output will define the position of the variable control schedule parameter(s) for which the performance is calculated. If the user has the option to reschedule any control schedules without consulting the program supplier, a description of the method of input and the permissible variation will be included in the user documentation.

5.13 Engine Stability:

The engine program will indicate component stability problems, (e.g., compressor surge, burner blow-out) by use of an appropriate Numerical Status Indicator.

5.14 User Supplied Programs:

The engine program, when used as a subprogram, may be required to interface with user supplied programs representing such features as engine inlet, ejector inlet for an engine supplied nozzle, exhaust nozzle, ports for customer services, or secondary nozzles for boundary layer control. The operation of the user supplied program and the engine subprogram must be closely coordinated to allow for successful interaction between them, and to ensure the necessary input and output is provided. The program user will provide documentation to the program supplier describing the intended use of all interface items.

6.   INPUT/OUTPUT:

6.1   Program Interface Definition:

6.1.1   For many years, engine programs were developed using the FORTRAN language. AS681 was developed, in part, to specify the standard for communicating between the calling program and the engine subprogram. Today, many languages are available to develop the engine subprogram and the calling program. Other communication methods may be preferable, even if FORTRAN is the language of choice. This standard no longer insists on the traditional program interface. ARP4191 defines the traditional program interface, for those who choose to use it. ARP4868 describes an alternate program interface that is also compliant to this Aerospace Standard.

All programs should provide the following inputs and outputs if appropriate to the engines being modeled. If the program uses 6-character labels in its input and output facilities, the labels given below are the standard and must be provided. This does not preclude use of longer labels or descriptions.

6.2   Minimum Input List:

CASE	Numerical case identification
FYPH	Primary maximum response frequency
FYSH	Secondary maximum response frequency
IND	Engine program designation
SDIST	Inlet pressure and temperature distortion selection
SERAM	Ram pressure recovery selection
SIM	Inlet Mode Selection
SWIND	Windmilling selection
TIMEO	Output time interval
TIMEF1	Time at which frequency is changed to secondary value, FYSH, (for transient case)
TIMEF2	Time at which frequency reverts to primary value, FYPH, (for transient case)
TITLE	User title
ZALT	Geopotential pressure altitude
ZDT1A	Temperature to be added to T1A
ZDTAMB	Ambient temperature minus standard atmospheric temperature
ZERAM1	Primary stream ram pressure recovery
ZERM11	Secondary stream ram pressure recovery
ZERM1A	Ram pressure recovery at station 1A
ZPAMB	Ambient pressure
ZP1A	Engine inlet total pressure at station 1A
ZPC	Power code
ZPLA	Power lever angle
ZPWSD	Delivered shaft power
ZPWXH	Customer high pressure rotor power extraction
ZRC	Rating code
ZT1A	Engine inlet total temperature at station 1A

6.2   (Continued):

ZTAMB	Ambient temperature
ZTIME	Time from start of transient case at which control is to be returned to calling program
ZTIMET	Termination time of transient case
ZTRQSD	Delivered shaft torque
ZWB3	High pressure compressor bleed flow rate
ZWB3Q	High pressure compressor discharge bleed flow ratio (discharge over component inlet)
ZXJPTL	Polar moment of inertia of power turbine load
ZXM	Free stream Mach number
ZXNSD	Delivered shaft rotational speed

6.3   Minimum Output List:

AE18	Bypass exhaust nozzle throat effective area
AE8	Primary exhaust nozzle throat effective area
ALT	Geopotential pressure altitude
ANGBT	Boat-tail angle
CLASS	Engine program security classification
DT1A	Temperature added to T1A
DTAMB	Ambient temperature minus standard atmosphere temperature
ERAM1	Primary stream ram pressure recovery
ERAM11	Secondary stream ram pressure recovery
ERAM1A	Ram pressure recovery at station 1A
FG	Gross thrust
FG19	Bypass stream gross thrust
FGI	Ideal gross thrust
FGI19	Bypass stream ideal gross thrust
FHV	Fuel lower heating value
FN	Net thrust
FRAM	Ram drag
IDENT	Engine program title(s)
NSI	Numerical Status Indicator. See 6.6
P17	Bypass exhaust flow total pressure
P1A	Engine inlet total pressure at station 1A
P7	Primary exhaust flow total pressure
PAMB	Ambient pressure
PB3	High pressure compressor discharge bleed flow total pressure
PC	Power code
PLA	Power lever angle
PWSD	Delivered shaft power
PWXH	Customer high pressure rotor power extraction
RC	Rating code
SFC	Specific fuel consumption
SMH	High Pressure Compressor Surge Margin
SMI	Intermediate Pressure Compressor Surge Margin
SML	Low Pressure Compressor Surge Margin

6.3   (Continued):

    T17   Bypass exhaust flow total temperature
    T1A   Engine inlet total temperature at station 1A
    T7   Primary exhaust flow total temperature
    TAMB   Ambient temperature
    TB3   High pressure compressor discharge bleed flow total temperature
    TC   Control temperature (cockpit display)
    TIME   Output Time, from start of transient case
    TRQSD   Delivered shaft torque
    W17   Bypass exhaust flow rate
    W1A   Engine inlet flow rate at station 1A
    W2   High pressure compressor inlet flow rate
    W7   Primary exhaust flow rate
    WB3   High pressure compressor discharge total bleed flow rate (combined result of inputs of total bleed flow, WB3, and bleed flow ratio, WB3Q)
    WB3Q   High pressure compressor total bleed flow ratio -- discharge over component inlet (combined result of inputs of total bleed flow, WB3, and bleed flow ratio, WB3Q)
    WFE   Engine fuel flow rate
    WFT   Total fuel flow rate
    XM   Free stream Mach number
    XNH   High pressure rotor rotational speed
    XNI   Intermediate pressure rotor rotational speed
    XNL   Low pressure rotor rotational speed
    XNSD   Delivered shaft rotational speed

6.4   Program Output:

Display of program output will be under the control of the calling program.

6.5   Units:

Input and output parameter values may be based on either the metric International System of Units (SI) or US customary units (pound, foot, etc.). The selection of one set or the other should be coordinated between user and supplier. Appendix B defines the units that should be used for either system.

6.6   Numerical Status Indicators:

6.6.1   Numerical Status Indicators will be used to notify the user of any limitation or qualification of the program output. The Numerical Status Indicator codes defined in 6.6.2 will be used. The program supplier may also elect to include a suitable message in the program output.

6.6.2 A four-digit number will be used as the message indicator. The left-most digit will indicate the quality of the output, using the definitions in Table 5. The second digit (from the left) will indicate the nature of the problem affecting the output, using the definitions in Table 6. The third and fourth digits will complete the code for the specific problem. All Numerical Status Indicator codes will be defined in the user documentation. A minimum of ten Numerical Status Indicators (the first nine and the last one encountered) will be saved for output during steady-state operation.

TABLE 5 - First Digit Numerical Status Indicator Assignments

NSI	First Digit, Quality Indicators
0000	All output valid, no errors, no limits exceeded
0XXX	Input reset to limit, output valid as specified
1XXX	Limit exceeded, output provided for interpolation only
2XXX to 8XXX	Open for program supplier's use
9XXX	Output not valid

During transient operation, at least one Numerical Status Indicator will be saved for output. Additional Numerical Status Indicators may be saved as agreed between supplier and user. If an invalid-output Numerical Status Indicator (9XXX) occurs, the transient should be terminated and the value of time (TIME) at which failure occurred should be available.