



ESDU 00034

Issued June 2000
With Amendments A to E
October 2013

Aircraft Performance Program Part 4: Brief form of input and output Programs APP02, APP03



**ROYAL
AERONAUTICAL
SOCIETY**

Endorsed by
The Royal Aeronautical Society



Documents
Document Sharing Hub

ESDU DATA ITEMS

Data Items provide validated information in engineering design and analysis for use by, or under the supervision of, professionally qualified engineers. The data are founded on an evaluation of all the relevant information, both published and unpublished, and are invariably supported by original work of ESDU staff engineers or consultants. The whole process is subject to independent review for which crucial support is provided by industrial companies, government research laboratories, universities and others from around the world through the participation of some of their leading experts on ESDU Technical Committees. This process ensures that the results of much valuable work (theoretical, experimental and operational), which may not be widely available or in a readily usable form, can be communicated concisely and accurately to the engineering community.

We are constantly striving to develop new work and review data already issued. Any comments arising out of your use of our data, or any suggestions for new topics or information that might lead to improvements, will help us to provide a better service.

THE PREPARATION OF THIS DATA ITEM

The work on this particular Data Item was monitored and guided by the Performance Committee. This Committee first met in 1946 and at the time of first issue of this Item had the following membership:

Chairman	
Mr K.J. Balkwill	— Independent
Members	
Mr P.V. Aidala*	— Northrop Grumman ESID, Bethpage, New York, USA
Mr T. Bartup	— BAE SYSTEMS, Regional Aircraft, Woodford
Mr M. Broad	— Independent
Mr G.M.J. Davis	— Defence Evaluation and Research Agency, Boscombe Down
Mr N.J. Herniman	— Airbus Industrie, Toulouse, France
Mr R.G. Humpston	— Rolls-Royce plc, Aero Division, Derby
Mr T.S.R. Jordan	— Independent
Mr R.J. Orlowski*	— Boeing Commercial Airplane Company, Seattle, Washington, USA
Dr P. Render	— University of Loughborough
Mr P. Renze*	— Raytheon Aircraft Company, Wichita, Kansas, USA
Mr D.N. Sinton	— Independent
Mr G.J.R. Skillen	— Civil Aviation Authority, Safety Regulation Group, Gatwick
Mr G.E. Smith	— Independent
Mr A. Stanbrook	— Independent
Mr R. Storey	— BAE SYSTEMS, Military Aircraft and Aerostructures, Brough
Prof. E. Torenbeek*	— Delft University of Technology, The Netherlands
Mr S. Tsach*	— Israel Aircraft Industries, Jerusalem, Israel
Mr C.J. Turner*	— Boeing, Douglas Products Group, Long Beach, Calif., USA
Mr M. Wilson	— Britten-Norman Ltd, Isle of Wight.

* Corresponding Member.

The construction and subsequent development of the Data Item was undertaken by

Mr A.J. Grundy — under contract to ESDU.

The person with overall responsibility for the work in this subject area was Mr D.J. Mitchell.

AIRCRAFT PERFORMANCE PROGRAM PART 4: BRIEF FORM OF INPUT AND OUTPUT PROGRAMS APP02 AND APP03

CONTENTS

	Page
1. INTRODUCTION	1
2. FLIGHT PERFORMANCE PROGRAM, APP02	1
2.1 Data Input	1
2.1.1 Input formats	1
2.1.2 Restricted use of tables	5
2.2 Program Output	5
2.3 Table Format	6
2.3.1 Facilities	6
3. INTERPOLATION PROGRAM, APP03	7
4. REFERENCES	8
5. TABLES	9
APPENDIX A – FLIGHT PERFORMANCE PROGRAM, APP02	13
A.1 Introduction	13
A.2 Keywords	13
A.2.1 Text Only	13
A.2.1.1 Calculation types	14
A.2.1.2 Calculation options	15
A.2.1.3 Atmospheric definitions	17
A.2.1.4 Integration step length options	17
A.2.1.5 Factor and increment variables	18
A.2.1.6 Miscellaneous items	18
A.2.2 Engine Data	19
A.2.3 Aerodynamic (Wing-Body or Trimmed Aircraft)	20
A.2.4 Trimming Surface Aerodynamic Data	22
A.2.5 Common Case Data	23
A.2.6 Aircraft and Engine Limitations	23
A.2.7 Special Case Data	24
A.2.8 Atmosphere Data	25
A.2.9 Initial or Constant and Final Speeds	26
A.2.10 Special Speeds	27
A.2.11 Pressure error corrections	27

APPENDIX B – INTERPOLATION PROGRAM, APP03	29
B.1 Data Input	29
B.1.1 Basic online input	29
B.1.2 Special Keyword Input	30
B.2 Output	31

AIRCRAFT PERFORMANCE PROGRAM PART 4: BRIEF FORM OF INPUT AND OUTPUT PROGRAMS APP02 and APP03

1. INTRODUCTION

(Note: In this Data Item, data options and facilities which are not available in the current release of the program are presented in *italic script*.)

A ready-reference summary of the input and output options and the keywords used by the programs that make up the ESDU Aircraft Performance Program are given in this Data Item. These programs are as follows.

- APP02, the Flight Performance Program. (Input and output data are described in detail in ESDU 00035 (Reference 10).)
- APP03, the Interpolation Program.

2. FLIGHT PERFORMANCE PROGRAM, APP02

Program APP02 contains the '**point performance**' routines, the '**path performance**' routines and time histories of the aircraft performance parameters. Some of the major lower-level routines below these performance calculations are available for use in isolation.

2.1 Data Input

Input data are described by the data Keywords and Symbolic Names summarised in Appendix A (and listed in full in Reference 10). The units used for input items are listed here in Table 5.1.

The user can specify input data either as a constant value, a range or list of values (to form a data permutation or combination), or as tabular data varying with a range of possible variables. For example, engine power setting could be either constant or vary with time (say) to represent a slam acceleration. Engine thrust and fuel flow can be input for up to five sets of jet engine or propeller-engine combinations having different performance per engine in each set (see examples in Section 2.3.2 of ESDU 00035 (Reference 10)). It is assumed that the engine data are fully installed (*i.e.* the data have been corrected, for example, for intake spillage, base drag and aerodynamic interferences) but provision is made for additional thrust factors, and incremental aerodynamic effects which can be functions of engine-dependent variables.

2.1.1 Input formats

All numeric input uses a keyword or symbolic name followed by the data value and units (if the input quantity has units which are known to the program), *e.g.* MASS 23000 kg. These data are supplemented by keywords defining calculation and other options. Keyword input must follow the rules listed here:

- (a) Keywords may be input in any mixture of upper case and lower case (**letters** 'O' or 'o' are always used in keywords not the number zero) and followed by at least one space.

Issued June 2000 – 23 pages
With Amendments A to E, October 2013 – 31 pages
This page Amendment E

- (b) The value and its units must be separated by at least one space. The values may consist of up to 20 characters; including the sign, leading zero, decimal point and exponent (when required using the FORTRAN format, *e.g.* -0.1234E-04)
- (c) All input associated with a particular keyword must be on the same line (maximum line length 100 characters).
No continuation character is provided, but additional cases could be input as a follow-on case after the END_CASE keyword.
- (d) Blank lines are permitted (but will be removed during data input).
- (e) Comment lines (starting with '!') or following data input and separated by at least one space or tab from the input, may appear anywhere in the data file.
- (f) The units **must** be input in the correct upper and lower case letters.
- (g) The input units declared with a table input do not have to be the same as those used in the table file.
- (h) Most keywords may be input in any order and may be repeated to change the value of an input.
- (i) For keywords that have units that are not known to the program (for example RATING12, ENGVAR3, FLAP and so on), no units name is input. These variables may be given a synonym (introduced by '=') in the data input. This synonym will be used on the output. There should be at least one space before the '=' sign and no spaces after the '=' sign or in the rest of the synonym.

The synonyms given to the miscellaneous variables, VARIABLE and ENGVAR families, may also be used as variable names in tables in place of the keyword (*e.g.* input, VARIABLE3 1.6 =SHP; then use SHP as a table variable which will be given the value 1.6). To be used as a table variable, a synonym must not be the name of any of the standard variable names in Table 5.2. A synonym can be allocated to a different miscellaneous variable when it is used in a new set of input case data; the synonym is not tied to any particular miscellaneous variable; this provides flexibility in the definition and use of the tables.

Input values may be specified in any of the following six formats.

- (1) Single constant value, *e.g.*

SWING	13.45 sqm
CLOWING	-0.78

- (2) A list of values separated by spaces or commas; the calculation is done at each of these values in turn, *e.g.*

SPLAY1	1.56 5.67 -0.89, 5.89 deg
VARIABLE1	13476, 35890, 45, -16.76
HP	14000 16000 12000 18000 19000 19500 ft

The three variables will produce a permutation of $4 \times 4 \times 6 = 96$ cases.

- (3) A family of lists for several variables with the same number of values for each variable, and with each list starting with @. This family then forms a combination of cases, *e.g.*

DELTAT	@ 1,	2,	3,	4	C
HP	@ 10,	20,	30,	40	m

This page Amendments D and E

PRIMARY1 @ 100 200 300 400

This is equivalent to the first case input of,

DELTAT	1 C
HP	10 m
PRIMARY1	100

the second case using the values,

DELTAT	2 C
HP	20 m
PRIMARY1	200

and so on until the last (fourth) case uses,

DELTAT	4 C
HP	40 m
PRIMARY1	400

- (4) A range of values starting from an initial value (X_1), with a constant, signed, increment (DeltaX) and finishing at a final value (X_f). The format is X_1 (DeltaX) X_f . The calculation is then done at each of these values in turn, *e.g.*

DELTAT	-15 (5) +25 C
HG	40000 (-5000) 0 ft

The first input is equivalent to the input, DELTAT -15, -10, -5, 0, 5, 10, 15, 20, 25 C

The two variables will produce a permutation of $9 \times 9 = 81$ cases.

- (5) A family of ranges for several variables with the same number of values for each variable, and with each range starting with @. This family then forms a combination of cases, *e.g.*

DELTAT	@ 1 (1) 4 C
HP	@ 10 (10) 40 m
PRIMARY1	@ 100 (100) 400

This is equivalent to the first case input of,

DELTAT	1 C
HP	10 m
PRIMARY1	100

the second case using the values,

DELTAT	2 C
HP	20 m
PRIMARY1	200

and so on until the last (fourth) case uses,

DELTAT	4 C
HP	40 m
PRIMARY1	400

- (6) As a table, with the additional option to specify a positive factor (Fact) and a signed increment (Delta) to the data on the table. The format is: TAB(table_name)<op1> Fact <op2> Delta (the operators can be: <op1> * or /, <op2> + or -). These factors and increments can be,

- Constants,
- Special factor and increment variables: MODTAB1 to MODTABO, see ESDU 00035 (Reference 10),
- A mixture of a constant and a MODTAB variable.

For example,

```
XGMAIN1    TAB(XENGINE) lb
XDMAIN1    TAB(XDENGINE)+100 N
CLWING     TAB(CL)*0.99 -0.02
```

! MODTAB1 is a variable factor on fuel flow

```
MODTAB1    TAB(Factor1) #
FEMAIN1    TAB(FuelFlow)*MODTAB1 lb/hr
FEMAIN2    TAB(FuelFlow)*MODTAB1 + 100 lb/hr
```

! MODTAB5 is a variable gross thrust increment, with synonym DelXG

```
MODTAB5    TAB(DeltaXG) N =DelXG
XGMAIN1    TAB(Thrust)+DelXG N
XGMAIN2    TAB(Thrust)*0.95 +MODTAB5 N
```

Notes

- (1) All cases

The input units declared with the table input do not have to be the same as those used in the table file. The case data file units are used to define the units to be used for output. The units on the table file(s) define the units of the dependent variable(s) (Section 3 and ESDU 00035 (Reference 10)). So if the output of (say) thrust is required in SI units but the engine data are in Imperial Units, the use of Newtons on the case data input will produce the output thrust in Newtons.

- (2) Constant increment

The input units declared with the table input are used to define the units of any increment applied.

- (3) Using MODTAB variables

- Factors. The dependent data in the MODTAB tables must be non-dimensional.
- Increments. The units of the dependent data in the MODTAB tables must imply the same SI units as those of the table to which the increment is applied.

The MODTAB tables can also be modified by further constant factors and increments, *e.g.*

! MODTAB5 is a variable gross thrust increment
 MODTAB5 TAB(DeltaXG)*1.05 N
 XGMAIN1 TAB(Thrust) +MODTAB5 N

Note: MODTAB tables can also be used to modify Special Speeds (Section 2.10).

Options (2), (3), (4) and (5) above can be used to define permutations and combinations on two or more variables. The order of input of the variables determines the order in which the permutation is expanded. If a permutation and a combination are used together, the combination is processed first (for the first set of values of the permutation) then repeated for each expansion of the permutation.

2.1.2 Restricted use of tables

Whilst any numeric item may be defined by a table, the initial values of a limited set of items must not change during calculation for a given set of data input (*i.e.* the table variables do not change). These inputs are shown in Table 5.2.

2.2 Program Output

The calculation output summarises the main variables likely to be used (*e.g.* height, speed), the main elements in the performance calculation (*e.g.* total resolved net thrust, total drag), and the predicted parameter(s). The user has options to add more detail for specific groups of data. Input data are output in the same units as specified in the input. New data created by the performance calculations are output in units determined by the output keyword. A detailed description of the output and of the extra, debugging, output that is produced when the debug keyword is used is given in ESDU Data Item 00035 (Reference 10).

The program output file (OUTFILE) contains the following.

- (a) ESDU Header Block
- (b) Copy of the input data file (INFILE) with any messages produced during data input
- (c) Data input summary
- (d) Name of table file directory (TABLEDIR) and number of different tables used
- (e) Table name, table variable names and up to five comment lines taken from the start of each table
- (f) Synonyms used for implicit and independent variables
- (g) Summary of program storage space used by tables
- (h) Calculation type, options used and title lines
- (i) Output variable headers with their output units. The output units are determined as follows.
 - For items that are input, the output units are determined by the input.
 - For items that are not input, the output units are determined on a case by case basis, for example
 - one option used from a set of alternatives,

When
tables
are
used

This page Amendments D and E

- values which have taken a default,
- values generated by the program (Table 5.3). The output keyword specifies which unit set will be used.

- (j) Blocks of results for each permutation and combination produced
- (k) Calculation messages, limitations *etc.*

If follow-on cases are used (END_CASE) the output repeats from item (d) to (k).

- (l) Summary of number of cases.

Descriptive error messages are output by the Flight Performance Program, see ESDU 00037 (Reference 12).

2.3 Table Format

Input data which are required to be tables of values (*e.g.* engine thrust, drag coefficient, design diving speed) are presented in one of the two formats described in ESDU 00035 (Reference 10). The names of the variables available for tables are listed in Table 5.4.

2.3.1 Facilities

- (a) **Table Filenames:** Tables are stored in files having the same name (1 to 10 characters) as the table, with the file extension **.TAB**. The table names must conform to the lowest common standard of computer operating system conventions for file names – the name is limited to 10 characters which may not include spaces or the symbols ‘(’, ‘)’ and ‘!’. These table files are kept in the sub-directory or folder defined by the **TABLEDIR** keyword, or the local directory or folder if this keyword is omitted.

- (b) **Table Formats:**

Type 1: The table contains keywords to identify the data items,

Type 2: Fixed format table.

- (c) **‘Rectangular’ Tables:** Only tables of data with the array completed to cover the full range of all the independent variables are allowed. However, missing values at the ‘corners’ of a table (which may have up to 5 variables) may be indicated by the character ‘M’ or ‘m’ (an ‘M-point’).

When M-points are encountered during an interpolation for the first variable, linear interpolation or extrapolation is used irrespective of the order defined on the table. The other variables are treated as normal but their results may be obtained using a value resulting from this M-point interpolation. If linear interpolation is successful, that is, it can be achieved using the pair of non-M-point values that bracket the required variable value, the result is considered correct. If the result can only be obtained by extrapolating from the nearest pair of non-M-point values, this will be accepted during an iteration (because extrapolation rules are ignored) but will result in a fatal error when the iteration convergence is checked.

- (d) **Comment Lines:** Comment lines (first character !) and blank lines may be used to improve readability. They may be placed anywhere in the table that uses keywords, and used in many places in the fixed format file that contains the table. In tables, only complete comment lines are permitted; comments may not be added to the line after the data input values.

- (e) **Table Units:** The units (Table 5.1) of the dependent variables are specified in the table definition. The values are converted to SI for calculation. These units do not have to be the same as those specified in the case data file. The case data file units are used to define the units of any increment applied and the units to be used for output.
- (f) **Five Variables:** From one to five variables are allowed on a table. The units (Table 5.1) of the variables are defined (if appropriate) and the values are converted to SI for calculation.
- (g) **Implicit Variables:** Up to nine implicit variables may be specified. These specify the constant value of a likely variable for which the data in the table are valid. The implicit variable values must match the values input in the case data. For example, engine data may be presented for ISA+10 C; the use of the implicit variable **DELISA** with the value and units **+10 C** would ensure that the table was only used at ISA+10 C conditions.
- (h) **Monotonic Variable Values:** Monotonically increasing or decreasing values of the independent variables are used.
- (i) **Independent Variable Values:** The values of the independent variables may be input as a range or a list of values, similar to the options used in the case data file. To ensure that the correct number of values is input, the number of values required (NVal) is also input. The values may consist of up to 20 characters; including the sign, leading zero, decimal point and exponent (when required using the FORTRAN format, *e.g.* -0.1234E-04).

Note: the final value in a range may be added as a reminder to the user, but the value is not used by the table handling system. *e.g.*,

Input of a Range	NVal X_1 (ΔX)
or a Range including the final value	NVal X_1 (ΔX) X_{NVal}
or a List	NVal $X_1, X_2, \dots, X_{NVal}$

- (j) **Interpolation:** The program provides linear interpolation (default on Type 1 tables) and second to fourth order polynomial interpolation if there are enough values of the independent variable, as required.
- (k) **Extrapolation:** Extrapolation at each end (lower and upper) of the range of each independent variable may be specified as,
 - no extrapolation (default on Type 1 tables),
 - constant value equal to the end value,
 - linear extrapolation.
- (l) **Table Storage:** All the tables are stored in a 1-dimensional array (to optimise on the use of space). If the storage array is not large enough to contain all the tables, the excess tables are written to a direct access file; then the storage array contains only a 'stub' header for the tables on the direct access file.

The interpolation routine may ignore the **no extrapolation** marker during an iteration so that a solution can be obtained. If the final converged result requires the use of extrapolation and **no extrapolation** has been violated, the result will be discarded.

3. INTERPOLATION PROGRAM, APP03

APP03 is a free-standing program which allows the user to interpolate (including inverse interpolation) on

This page Amendments C and E

tables used by the performance program. The program uses the same table input and interpolation routines as the performance program but does not apply factors to convert the data to SI units: the table, the input variables used (see ESDU 00035 (Reference 10), Section 3.3 for definitions) in the interpolations and the interpolated results remain in the units defined on the table.

Data are input to the Interpolation Program in response to prompts issued by the program.

For a guide to using the Interpolation Program see Appendix B. Examples of the input prompts, the data requirements and the output produced are shown in ESDU 00036 (Reference 11).

4. REFERENCES

For the list of References see ESDU Data Item 00031.