T H E

```
        SSSSS           OOOOOO       FFFFFFFFFFFFF      AAAAAAA
     SSSSSSSSSS     OOOOOOOOOOOO     FFFFFFFFFFFF      AAAAAAAA
    SSSSSSSSSSS   OOOOOOOOOOOOOO    FFFFFFFFFFFF      AAAA AAAA
   SSSS      S   OOOOOO    OOOOO    FFFF             AAAA  AAAA
  SSSSS         OOOOO       OOOO   FFFFF            AAAA   AAAA
  SSSSSSSSSS    OOOO       OOOOO   FFFFFFFFFFFF    AAAA    AAAA
    SSSSSSSSS  OOOOO       OOOO   FFFFFFFFFFFF    AAAAAAAAAAAAA
        SSSSS  OOOO       OOOO    FFFF           AAAAAAAAAAAAAA
 S      SSSS   OOOOO    OOOOO    FFFF           AAAAAAAAAAAAAAA
SSSSSSSSSSS    OOOOOOOOOOOOO     FFFF          AAAA       AAAAA
SSSSSSSSS       OOOOOOOOOO      FFFF          AAAA        AAAAA
  SSSS            OOOOO         FFFF         AAAA         AAAAA
```

S O F T W A R E

L I B R A R I E S

International Astronomical Union

Division 1: Fundamental Astronomy

Commission 19: Rotation of the Earth

Standards Of Fundamental Astronomy Board

Release 9

2012 March 1

CONTENTS

THE IAU-SOFA SOFTWARE LIBRARIES

SOFA stands for "Standards Of Fundamental Astronomy". The SOFA software libraries are a collection of subprograms, in source-code form, which implement official IAU algorithms for fundamental-astronomy computations. The subprograms at present comprise 131 "astronomy" routines supported by 55 "vector/matrix" routines, available in both Fortran77 and C implementations.

THE SOFA INITIATIVE

SOFA is an IAU Service which operates under Division 1 (Fundamental Astronomy) and reports through Commission 19 (Rotation of the Earth).

The IAU set up the SOFA initiative at the 1994 General Assembly, to promulgate an authoritative set of fundamental-astronomy constants and algorithms. At the subsequent General Assembly, in 1997, the appointment of a SOFA Review Board and the selection of a site for the SOFA Center (the outlet for SOFA products) were announced.

The SOFA initiative was originally proposed by the IAU Working Group on Astronomical Standards (WGAS), under the chairmanship of Toshio Fukushima. The proposal was for "...new arrangements to establish and maintain an accessible and authoritative set of constants, algorithms and procedures that implement standard models used in fundamental astronomy". The SOFA Software Libraries implement the "algorithms" part of the SOFA initiative. They were developed under the supervision of an international panel called the SOFA Review Board. The current membership of this panel is listed in an appendix.

A feature of the original SOFA software proposals was that the products would be self-contained and not depend on other software. This includes basic documentation, which, like the present file, will mostly be plain ASCII text. It should also be noted that there is no assumption that the software will be used on a particular computer and Operating System. Although OS-related facilities may be present (Unix make files for instance, use by the SOFA Center of automatic code management systems, HTML versions of some documentation), the routines themselves will be visible as individual text files and will run on a variety of platforms.

ALGORITHMS

The SOFA Review Board's initial goal has been to create a set of callable subprograms. Whether "subroutines" or "functions", they are all referred to simply as "routines". They are designed for use by software developers wishing to write complete applications; no runnable, free-standing applications are included in SOFA's present plans.

The algorithms are drawn from a variety of sources. Because most of the routines so far developed have either been standard "text-book" operations or implement well-documented standard algorithms, it has not been necessary to invite the whole community to submit algorithms, though consultation with authorities has occurred where necessary. It should also be noted that consistency with the conventions published by the International Earth Rotation Service was a stipulation in the original SOFA proposals, further constraining the software designs. This state of affairs will continue to exist for some time, as there is a large backlog of agreed extensions to work on. However, in the future the Board may decide to call for proposals, and is in the meantime willing to look into any suggestions that are received by the SOFA Center.

SCOPE

The routines currently available are listed in the next two chapters of this document.

The "astronomy" library comprises 131 routines (plus one obsolete Fortran routine that now appears under a revised name). The areas addressed include calendars, time scales, ephemerides, precession-nutation, star space-motion, star catalog transformations and geodetic/geocentric transformations.

The "vector-matrix" library, comprising 55 routines, contains a collection of simple tools for manipulating the vectors, matrices and angles used by the astronomy routines.

There is no explicit commitment by SOFA to support historical models, though as time goes on a legacy of superseded models will naturally accumulate. There is, for example, no support of B1950/FK4 star coordinates, or pre-1976 precession models, though these capabilities could be added were there significant demand.

Though the SOFA software libraries are rather limited in scope, and are likely to remain so for a considerable time, they do offer distinct advantages to prospective users. In particular, the routines are:

* authoritative: they are IAU-backed and have been constructed with great care;

* practical: they are straightforward to use in spite of being precise and rigorous (to some stated degree);

* accessible and supported: they are downloadable from an easy-to-find place, they are in an integrated and consistent form, they come with adequate internal documentation, and help for users is available.

VERSIONS

Once it has been published, an issue is never revised or updated, and remains accessible indefinitely. Subsequent issues may, however, include corrected versions under the original routine name and filenames. However, where a different model is introduced, it will have a different name.

The issues will be referred to by the date when they were announced. The frequency of re-issue will be decided by the Board, taking into account the importance of the changes and the impact on the user community.

DOCUMENTATION

At present there is little free-standing documentation about individual routines. However, each routine has preamble comments which specify in detail what the routine does and how it is used.

The file sofa_pn.pdf describes the SOFA tools for precession-nutation and other aspects of Earth attitude and includes example code and (see the appendix) diagrams showing the interrelationships between the routines supporting the latest (IAU 2006/2000A) models.

PROGRAMMING LANGUAGES AND STANDARDS

The SOFA routines are available in two programming languages at present: Fortran77 and ANSI C. Related software in other languages is under consideration.

The Fortran code conforms to ANSI X3.9-1978 in all but two minor respects: each has an IMPLICIT NONE declaration, and its name has a prefix of "iau_" and may be longer than 6 characters. A global edit to erase both of these will produce ANSI-compliant code with no change in its function.

Coding style, and restrictions on the range of language features, have been much debated by the Board, and the results comply with the majority view. There is (at present) no document that defines the standards, but the code itself offers a wide range of examples of what is acceptable.

The Fortran routines contain explicit numerical constants (the INCLUDE statement is not part of ANSI Fortran77). These are drawn from the file consts.lis, which is listed in an appendix. Constants for the SOFA/C functions are defined in a header file sofam.h.

The naming convention is such that a SOFA routine referred to generically as "EXAMPL" exists as a Fortran subprogram iau_EXAMPL and a C function iauExampl. The calls for the two versions are very similar, with the same arguments in the same order. In a few cases, the C equivalent of a Fortran SUBROUTINE subprogram uses a return value rather than an argument.

Each language version includes a "testbed" main-program that can be used to verify that the SOFA routines have been correctly compiled on the end user's system. The Fortran and C versions are called t_sofa_f.for and t_sofa_c.c respectively. The testbeds execute every SOFA routine and check that the results are within expected accuracy margins. It is not possible to guarantee that all platforms will meet the rather stringent criteria that have been used, and an occasional warning message may be encountered on some systems.

COPYRIGHT ISSUES

Copyright for all of the SOFA software and documentation is owned by the IAU SOFA Review Board. The Software is made available free of charge for all classes of user, including commercial. However, there are strict rules designed to avoid unauthorized variants coming into circulation. It is permissible to distribute derived works and other modifications, but they must be clearly marked to avoid confusion with the SOFA originals.

Further details are included in the block of comments which concludes every routine. The text is also set out in an appendix to the present document.

ACCURACY

The SOFA policy is to organize the calculations so that the machine accuracy is fully exploited. The gap between the precision of the underlying model or theory and the computational resolution has to be kept as large as possible, hopefully leaving several orders of magnitude of headroom.

The SOFA routines in some cases involve design compromises between rigor and ease of use (and also speed, though nowadays this is seldom a major concern).

ACKNOWLEDGEMENTS

The Board is indebted to a number of contributors, who are acknowledged in the preamble comments of the routines concerned.

The Board's effort is provided by the members' individual institutes.

Resources for operating the SOFA Center are provided by Her Majesty's Nautical Almanac Office, operated by the United Kingdom Hydrographic Office.

SOFA Astronomy Library

PREFACE

The routines described here comprise the SOFA astronomy library. Their general appearance and coding style conforms to conventions agreed by the SOFA Review Board, and their functions, names and algorithms have been ratified by the Board. Procedures for soliciting and agreeing additions to the library are still evolving.

PROGRAMMING LANGUAGES

The SOFA routines are available in two programming languages at present: Fortran 77 and ANSI C.

Except for a single obsolete Fortran routine, which has no C equivalent, there is a one-to-one relationship between the two language versions. The naming convention is such that a SOFA routine referred to generically as "EXAMPL" exists as a Fortran subprogram iau_EXAMPL and a C function iauExampl. The calls for the two versions are very similar, with the same arguments in the same order. In a few cases, the C equivalent of a Fortran SUBROUTINE subprogram uses a return value rather than an argument.

GENERAL PRINCIPLES

The principal function of the SOFA Astronomy Library is to provide definitive algorithms. A secondary function is to provide software suitable for convenient direct use by writers of astronomical applications.

The astronomy routines call on the SOFA vector/matrix library routines, which are separately listed.

The routines are designed to exploit the full floating-point accuracy of the machines on which they run, and not to rely on compiler optimizations. Within these constraints, the intention is that the code corresponds to the published formulation (if any).

Dates are always Julian Dates (except in calendar conversion routines) and are expressed as two double precision numbers which sum to the required value.

A distinction is made between routines that implement IAU-approved models and those that use those models to create other results. The former are referred to as "canonical models" in the preamble comments; the latter are described as "support routines".

Using the library requires knowledge of positional astronomy and time-scales. These topics are covered in "Explanatory Supplement to the Astronomical Almanac", P. Kenneth Seidelmann (ed.), University Science Books, 1992. Recent developments are documented in the journals, and references to the relevant papers are given in the SOFA code as required. The IERS Conventions are also an essential reference. The routines concerned with Earth attitude (precession-nutation etc.) are described in the SOFA document sofa_pn.pdf.

ROUTINES

Calendars

CAL2JD	Gregorian calendar to Julian Day number
EPB	Julian Date to Besselian Epoch
EPB2JD	Besselian Epoch to Julian Date
EPJ	Julian Date to Julian Epoch

```
   EPJ2JD    Julian Epoch to Julian Date
   JD2CAL    Julian Date to Gregorian year, month, day, fraction
   JDCALF    Julian Date to Gregorian date for formatted output

Time scales

   D2DTF     format 2-part JD for output
   DAT       Delta(AT) (=TAI-UTC) for a given UTC date
   DTDB      TDB-TT
   DTF2D     encode time and date fields into 2-part JD
   TAITT     TAI to TT
   TAIUT1    TAI to UT1
   TAIUTC    TAI to UTC
   TCBTDB    TCB to TDB
   TCGTT     TCG to TT
   TDBTCB    TDB to TCB
   TDBTT     TDB to TT
   TTTAI     TT to TAI
   TTTCG     TT to TCG
   TTTDB     TT to TDB
   TTUT1     TT to UT1
   UT1TAI    UT1 to TAI
   UT1TT     UT1 to TT
   UT1UTC    UT1 to UTC
   UTCTAI    UTC to TAI
   UTCUT1    UTC to UT1

Earth rotation angle and sidereal time

   EE00      equation of the equinoxes, IAU 2000
   EE00A     equation of the equinoxes, IAU 2000A
   EE00B     equation of the equinoxes, IAU 2000B
   EE06A     equation of the equinoxes, IAU 2006/2000A
   EECT00    equation of the equinoxes complementary terms, IAU 2000
   EQEQ94    equation of the equinoxes, IAU 1994
   ERA00     Earth rotation angle, IAU 2000
   GMST00    Greenwich mean sidereal time, IAU 2000
   GMST06    Greenwich mean sidereal time, IAU 2006
   GMST82    Greenwich mean sidereal time, IAU 1982
   GST00A    Greenwich apparent sidereal time, IAU 2000A
   GST00B    Greenwich apparent sidereal time, IAU 2000B
   GST06     Greenwich apparent ST, IAU 2006, given NPB matrix
   GST06A    Greenwich apparent sidereal time, IAU 2006/2000A
   GST94     Greenwich apparent sidereal time, IAU 1994

Ephemerides (limited precision)

   EPV00     Earth position and velocity
   PLAN94    major-planet position and velocity

Precession, nutation, polar motion

   BI00      frame bias components, IAU 2000
   BP00      frame bias and precession matrices, IAU 2000
   BP06      frame bias and precession matrices, IAU 2006
   BPN2XY    extract CIP X,Y coordinates from NPB matrix
   C2I00A    celestial-to-intermediate matrix, IAU 2000A
   C2I00B    celestial-to-intermediate matrix, IAU 2000B
   C2I06A    celestial-to-intermediate matrix, IAU 2006/2000A
   C2IBPN    celestial-to-intermediate matrix, given NPB matrix, IAU 2000
   C2IXY     celestial-to-intermediate matrix, given X,Y, IAU 2000
   C2IXYS    celestial-to-intermediate matrix, given X,Y and s
   C2T00A    celestial-to-terrestrial matrix, IAU 2000A
   C2T00B    celestial-to-terrestrial matrix, IAU 2000B
   C2T06A    celestial-to-terrestrial matrix, IAU 2006/2000A
   C2TCIO    form CIO-based celestial-to-terrestrial matrix
   C2TEQX    form equinox-based celestial-to-terrestrial matrix
   C2TPE     celestial-to-terrestrial matrix given nutation, IAU 2000
   C2TXY     celestial-to-terrestrial matrix given CIP, IAU 2000
   EO06A     equation of the origins, IAU 2006/2000A
   EORS      equation of the origins, given NPB matrix and s
   FW2M      Fukushima-Williams angles to r-matrix
   FW2XY     Fukushima-Williams angles to X,Y
```

```
   NUM00A    nutation matrix, IAU 2000A
   NUM00B    nutation matrix, IAU 2000B
   NUM06A    nutation matrix, IAU 2006/2000A
   NUMAT     form nutation matrix
   NUT00A    nutation, IAU 2000A
   NUT00B    nutation, IAU 2000B
   NUT06A    nutation, IAU 2006/2000A
   NUT80     nutation, IAU 1980
   NUTM80    nutation matrix, IAU 1980
   OBL06     mean obliquity, IAU 2006
   OBL80     mean obliquity, IAU 1980
   PB06      zeta,z,theta precession angles, IAU 2006, including bias
   PFW06     bias-precession Fukushima-Williams angles, IAU 2006
   PMAT00    precession matrix (including frame bias), IAU 2000
   PMAT06    PB matrix, IAU 2006
   PMAT76    precession matrix, IAU 1976
   PN00      bias/precession/nutation results, IAU 2000
   PN00A     bias/precession/nutation, IAU 2000A
   PN00B     bias/precession/nutation, IAU 2000B
   PN06      bias/precession/nutation results, IAU 2006
   PN06A     bias/precession/nutation results, IAU 2006/2000A
   PNM00A    classical NPB matrix, IAU 2000A
   PNM00B    classical NPB matrix, IAU 2000B
   PNM06A    classical NPB matrix, IAU 2006/2000A
   PNM80     precession/nutation matrix, IAU 1976/1980
   P06E      precession angles, IAU 2006, equinox based
   POM00     polar motion matrix
   PR00      IAU 2000 precession adjustments
   PREC76    accumulated precession angles, IAU 1976
   S00       the CIO locator s, given X,Y, IAU 2000A
   S00A      the CIO locator s, IAU 2000A
   S00B      the CIO locator s, IAU 2000B
   S06       the CIO locator s, given X,Y, IAU 2006
   S06A      the CIO locator s, IAU 2006/2000A
   SP00      the TIO locator s', IERS 2003
   XY06      CIP, IAU 2006/2000A, from series
   XYS00A    CIP and s, IAU 2000A
   XYS00B    CIP and s, IAU 2000B
   XYS06A    CIP and s, IAU 2006/2000A

Fundamental arguments for nutation etc.

   FAD03     mean elongation of the Moon from the Sun
   FAE03     mean longitude of Earth
   FAF03     mean argument of the latitude of the Moon
   FAJU03    mean longitude of Jupiter
   FAL03     mean anomaly of the Moon
   FALP03    mean anomaly of the Sun
   FAMA03    mean longitude of Mars
   FAME03    mean longitude of Mercury
   FANE03    mean longitude of Neptune
   FAOM03    mean longitude of the Moon's ascending node
   FAPA03    general accumulated precession in longitude
   FASA03    mean longitude of Saturn
   FAUR03    mean longitude of Uranus
   FAVE03    mean longitude of Venus

Star space motion

   PVSTAR    space motion pv-vector to star catalog data
   STARPV    star catalog data to space motion pv-vector

Star catalog conversions

   FK52H     transform FK5 star data into the Hipparcos system
   FK5HIP    FK5 to Hipparcos rotation and spin
   FK5HZ     FK5 to Hipparcos assuming zero Hipparcos proper motion
   H2FK5     transform Hipparcos star data into the FK5 system
   HFK5Z     Hipparcos to FK5 assuming zero Hipparcos proper motion
   STARPM    proper motion between two epochs

Geodetic/geocentric
```

```
     EFORM     a,f for a nominated Earth reference ellipsoid
     GC2GD     geocentric to geodetic for a nominated ellipsoid
     GC2GDE    geocentric to geodetic given ellipsoid a,f
     GD2GC     geodetic to geocentric for a nominated ellipsoid
     GD2GCE    geodetic to geocentric given ellipsoid a,f


  Obsolete

     C2TCEO    former name of C2TCIO


CALLS: FORTRAN VERSION

   CALL iau_BI00   ( DPSIBI, DEPSBI, DRA )
   CALL iau_BP00   ( DATE1, DATE2, RB, RP, RBP )
   CALL iau_BP06   ( DATE1, DATE2, RB, RP, RBP )
   CALL iau_BPN2XY ( RBPN, X, Y )
   CALL iau_C2I00A ( DATE1, DATE2, RC2I )
   CALL iau_C2I00B ( DATE1, DATE2, RC2I )
   CALL iau_C2I06A ( DATE1, DATE2, RC2I )
   CALL iau_C2IBPN ( DATE1, DATE2, RBPN, RC2I )
   CALL iau_C2IXY  ( DATE1, DATE2, X, Y, RC2I )
   CALL iau_C2IXYS ( X, Y, S, RC2I )
   CALL iau_C2T00A ( TTA, TTB, UTA, UTB, XP, YP, RC2T )
   CALL iau_C2T00B ( TTA, TTB, UTA, UTB, XP, YP, RC2T )
   CALL iau_C2T06A ( TTA, TTB, UTA, UTB, XP, YP, RC2T )
   CALL iau_C2TCEO ( RC2I, ERA, RPOM, RC2T )
   CALL iau_C2TCIO ( RC2I, ERA, RPOM, RC2T )
   CALL iau_C2TEQX ( RBPN, GST, RPOM, RC2T )
   CALL iau_C2TPE  ( TTA, TTB, UTA, UTB, DPSI, DEPS, XP, YP, RC2T )
   CALL iau_C2TXY  ( TTA, TTB, UTA, UTB, X, Y, XP, YP, RC2T )
   CALL iau_CAL2JD ( IY, IM, ID, DJM0, DJM, J )
   CALL iau_D2DTF  ( SCALE, NDP, D1, D2, IY, IM, ID, IHMSF, J )
   CALL iau_DAT    ( IY, IM, ID, FD, DELTAT, J )
   D =  iau_DTDB   ( DATE1, DATE2, UT, ELONG, U, V )
   CALL iau_DTF2D  ( SCALE, IY, IM, ID, IHR, IMN, SEC, D1, D2, J )
   D =  iau_EE00   ( DATE1, DATE2, EPSA, DPSI )
   D =  iau_EE00A  ( DATE1, DATE2 )
   D =  iau_EE00B  ( DATE1, DATE2 )
   D =  iau_EE06A  ( DATE1, DATE2 )
   D =  iau_EECT00 ( DATE1, DATE2 )
   CALL iau_EFORM  ( N, A, F, J )
   D =  iau_EO06A  ( DATE1, DATE2 )
   D =  iau_EORS   ( RNPB, S )
   D =  iau_EPB    ( DJ1, DJ2 )
   CALL iau_EPB2JD ( EPB, DJM0, DJM )
   D =  iau_EPJ    ( DJ1, DJ2 )
   CALL iau_EPJ2JD ( EPJ, DJM0, DJM )
   CALL iau_EPV00  ( DJ1, DJ2, PVH, PVB, J )
   D =  iau_EQEQ94 ( DATE1, DATE2 )
   D =  iau_ERA00  ( DJ1, DJ2 )
   D =  iau_FAD03  ( T )
   D =  iau_FAE03  ( T )
   D =  iau_FAF03  ( T )
   D =  iau_FAJU03 ( T )
   D =  iau_FAL03  ( T )
   D =  iau_FALP03 ( T )
   D =  iau_FAMA03 ( T )
   D =  iau_FAME03 ( T )
   D =  iau_FANE03 ( T )
   D =  iau_FAOM03 ( T )
   D =  iau_FAPA03 ( T )
   D =  iau_FASA03 ( T )
   D =  iau_FAUR03 ( T )
   D =  iau_FAVE03 ( T )
   CALL iau_FK52H  ( R5, D5, DR5, DD5, PX5, RV5,
  :                  RH, DH, DRH, DDH, PXH, RVH )
   CALL iau_FK5HIP ( R5H, S5H )
   CALL iau_FK5HZ  ( R5, D5, DATE1, DATE2, RH, DH )
   CALL iau_FW2M   ( GAMB, PHIB, PSI, EPS, R )
   CALL iau_FW2XY  ( GAMB, PHIB, PSI, EPS, X, Y )
   CALL iau_GC2GD  ( N, XYZ, ELONG, PHI, HEIGHT, J )
```

```
 CALL iau_GC2GDE ( A, F, XYZ, ELONG, PHI, HEIGHT, J )
 CALL iau_GD2GC  ( N, ELONG, PHI, HEIGHT, XYZ, J )
 CALL iau_GD2GCE ( A, F, ELONG, PHI, HEIGHT, XYZ, J )
 D =  iau_GMST00 ( UTA, UTB, TTA, TTB )
 D =  iau_GMST06 ( UTA, UTB, TTA, TTB )
 D =  iau_GMST82 ( UTA, UTB )
 D =  iau_GST00A ( UTA, UTB, TTA, TTB )
 D =  iau_GST00B ( UTA, UTB )
 D =  iau_GST06  ( UTA, UTB, TTA, TTB, RNPB )
 D =  iau_GST06A ( UTA, UTB, TTA, TTB )
 D =  iau_GST94  ( UTA, UTB )
 CALL iau_H2FK5  ( RH, DH, DRH, DDH, PXH, RVH,
:                  R5, D5, DR5, DD5, PX5, RV5 )
 CALL iau_HFK5Z  ( RH, DH, DATE1, DATE2, R5, D5, DR5, DD5 )
 CALL iau_JD2CAL ( DJ1, DJ2, IY, IM, ID, FD, J )
 CALL iau_JDCALF ( NDP, DJ1, DJ2, IYMDF, J )
 CALL iau_NUM00A ( DATE1, DATE2, RMATN )
 CALL iau_NUM00B ( DATE1, DATE2, RMATN )
 CALL iau_NUM06A ( DATE1, DATE2, RMATN )
 CALL iau_NUMAT  ( EPSA, DPSI, DEPS, RMATN )
 CALL iau_NUT00A ( DATE1, DATE2, DPSI, DEPS )
 CALL iau_NUT00B ( DATE1, DATE2, DPSI, DEPS )
 CALL iau_NUT06A ( DATE1, DATE2, DPSI, DEPS )
 CALL iau_NUT80  ( DATE1, DATE2, DPSI, DEPS )
 CALL iau_NUTM80 ( DATE1, DATE2, RMATN )
 D =  iau_OBL06  ( DATE1, DATE2 )
 D =  iau_OBL80  ( DATE1, DATE2 )
 CALL iau_PB06   ( DATE1, DATE2, BZETA, BZ, BTHETA )
 CALL iau_PFW06  ( DATE1, DATE2, GAMB, PHIB, PSIB, EPSA )
 CALL iau_PLAN94 ( DATE1, DATE2, NP, PV, J )
 CALL iau_PMAT00 ( DATE1, DATE2, RBP )
 CALL iau_PMAT06 ( DATE1, DATE2, RBP )
 CALL iau_PMAT76 ( DATE1, DATE2, RMATP )
 CALL iau_PN00   ( DATE1, DATE2, DPSI, DEPS,
:                  EPSA, RB, RP, RBP, RN, RBPN )
 CALL iau_PN00A  ( DATE1, DATE2,
:                  DPSI, DEPS, EPSA, RB, RP, RBP, RN, RBPN )
 CALL iau_PN00B  ( DATE1, DATE2,
:                  DPSI, DEPS, EPSA, RB, RP, RBP, RN, RBPN )
 CALL iau_PN06   ( DATE1, DATE2, DPSI, DEPS,
:                  EPSA, RB, RP, RBP, RN, RBPN )
 CALL iau_PN06A  ( DATE1, DATE2,
                   DPSI, DEPS, RB, RP, RBP, RN, RBPN )
 CALL iau_PNM00A ( DATE1, DATE2, RBPN )
 CALL iau_PNM00B ( DATE1, DATE2, RBPN )
 CALL iau_PNM06A ( DATE1, DATE2, RNPB )
 CALL iau_PNM80  ( DATE1, DATE2, RMATPN )
 CALL iau_P06E   ( DATE1, DATE2,
:                  EPS0, PSIA, OMA, BPA, BQA, PIA, BPIA,
:                  EPSA, CHIA, ZA, ZETAA, THETAA, PA, GAM, PHI, PSI )
 CALL iau_POM00  ( XP, YP, SP, RPOM )
 CALL iau_PR00   ( DATE1, DATE2, DPSIPR, DEPSPR )
 CALL iau_PREC76 ( EP01, EP02, EP11, EP12, ZETA, Z, THETA )
 CALL iau_PVSTAR ( PV, RA, DEC, PMR, PMD, PX, RV, J )
 D =  iau_S00    ( DATE1, DATE2, X, Y )
 D =  iau_S00A   ( DATE1, DATE2 )
 D =  iau_S00B   ( DATE1, DATE2 )
 D =  iau_S06    ( DATE1, DATE2, X, Y )
 D =  iau_S06A   ( DATE1, DATE2 )
 D =  iau_SP00   ( DATE1, DATE2 )
 CALL iau_STARPM ( RA1, DEC1, PMR1, PMD1, PX1, RV1,
:                  EP1A, EP1B, EP2A, EP2B,
:                  RA2, DEC2, PMR2, PMD2, PX2, RV2, J )
 CALL iau_STARPV ( RA, DEC, PMR, PMD, PX, RV, PV, J )
 CALL iau_TAITT  ( TAI1, TAI2, TT1, TT2, J )
 CALL iau_TAIUT1 ( TAI1, TAI2, DTA, UT11, UT12, J )
 CALL iau_TAIUTC ( TAI1, TAI2, UTC1, UTC2, J )
 CALL iau_TCBTDB ( TCB1, TCB2, TDB1, TDB2, J )
 CALL iau_TCGTT  ( TCG1, TCG2, TT1, TT2, J )
 CALL iau_TDBTCB ( TDB1, TDB2, TCB1, TCB2, J )
 CALL iau_TDBTT  ( TDB1, TDB2, DTR, TT1, TT2, J )
 CALL iau_TTTAI  ( TT1, TT2, TAI1, TAI2, J )
 CALL iau_TTTCG  ( TT1, TT2, TCG1, TCG2, J )
```

```
   CALL iau_TTTDB  ( TT1, TT2, DTR, TDB1, TDB2, J )
   CALL iau_TTUT1  ( TT1, TT2, DT, UT11, UT12, J )
   CALL iau_UT1TAI ( UT11, UT12, TAI1, TAI2, J )
   CALL iau_UT1TT  ( UT11, UT12, DT, TT1, TT2, J )
   CALL iau_UT1UTC ( UT11, UT12, DUT, UTC1, UTC2, J )
   CALL iau_UTCTAI ( UTC1, UTC2, DTA, TAI1, TAI2, J )
   CALL iau_UTCUT1 ( UTC1, UTC2, DUT, UT11, UT12, J )
   CALL iau_XY06   ( DATE1, DATE2, X, Y )
   CALL iau_XYS00A ( DATE1, DATE2, X, Y, S )
   CALL iau_XYS00B ( DATE1, DATE2, X, Y, S )
   CALL iau_XYS06A ( DATE1, DATE2, X, Y, S )
```

CALLS: C VERSION

```
       iauBi00   ( &dpsibi, &depsbi, &dra );
       iauBp00   ( date1, date2, rb, rp, rbp );
       iauBp06   ( date1, date2, rb, rp, rbp );
       iauBpn2xy ( rbpn, &x, &y );
       iauC2i00a ( date1, date2, rc2i );
       iauC2i00b ( date1, date2, rc2i );
       iauC2i06a ( date1, date2, rc2i );
       iauC2ibpn ( date1, date2, rbpn, rc2i );
       iauC2ixy  ( date1, date2, x, y, rc2i );
       iauC2ixys ( x, y, s, rc2i );
       iauC2t00a ( tta, ttb, uta, utb, xp, yp, rc2t );
       iauC2t00b ( tta, ttb, uta, utb, xp, yp, rc2t );
       iauC2t06a ( tta, ttb, uta, utb, xp, yp, rc2t );
       iauC2tcio ( rc2i, era, rpom, rc2t );
       iauC2teqx ( rbpn, gst, rpom, rc2t );
       iauC2tpe  ( tta, ttb, uta, utb, dpsi, deps, xp, yp, rc2t );
       iauC2txy  ( tta, ttb, uta, utb, x, y, xp, yp, rc2t );
   i = iauCal2jd ( iy, im, id, &djm0, &djm );
   i = iauD2dtf  ( scale, ndp, d1, d2, &iy, &im, &id, ihmsf );

   i = iauDat    ( iy, im, id, fd, &deltat );
   d = iauDtdb   ( date1, date2, ut, elong, u, v );
   i = iauDtf2d  ( scale, iy, im, id, ihr, imn, sec, &d1, &d2 );
   d = iauEe00   ( date1, date2, epsa, dpsi );
   d = iauEe00a  ( date1, date2 );
   d = iauEe00b  ( date1, date2 );
   d = iauEe06   ( date1, date2 );
   d = iauEect00 ( date1, date2 );
   i = iauEform  ( n, &a, &f );
   d = iauEo06   ( date1, date2 );
   d = iauEors   ( rnpb, s );
   d = iauEpb    ( dj1, dj2 );
       iauEpb2jd ( epb, &djm0, &djm );
   d = iauEpj    ( dj1, dj2 );
       iauEpj2jd ( epj, &djm0, &djm );
   i = iauEpv00  ( dj1, dj2, pvh, pvb );
   d = iauEqeq94 ( date1, date2 );
   d = iauEra00  ( dj1, dj2 );
   d = iauFad03  ( t );
   d = iauFae03  ( t );
   d = iauFaf03  ( t );
   d = iauFaju03 ( t );
   d = iauFal03  ( t );
   d = iauFalp03 ( t );
   d = iauFama03 ( t );
   d = iauFame03 ( t );
   d = iauFane03 ( t );
   d = iauFaom03 ( t );
   d = iauFapa03 ( t );
   d = iauFasa03 ( t );
   d = iauFaur03 ( t );
   d = iauFave03 ( t );
       iauFk52h  ( r5, d5, dr5, dd5, px5, rv5,
                   &rh, &dh, &drh, &ddh, &pxh, &rvh );
       iauFk5hip ( r5h, s5h );
       iauFk5hz  ( r5, d5, date1, date2, &rh, &dh );
       iauFw2m   ( gamb, phib, psi, eps, r );
       iauFw2xy  ( gamb, phib, psi, eps, &x, &y );
```

```
i = iauGc2gd  ( n, xyz, &elong, &phi, &height );
i = iauGc2gde ( a, f, xyz, &elong, &phi, &height );
i = iauGd2gc  ( n, elong, phi, height, xyz );
i = iauGd2gce ( a, f, elong, phi, height, xyz );
d = iauGmst00 ( uta, utb, tta, ttb );
d = iauGmst06 ( uta, utb, tta, ttb );
d = iauGmst82 ( uta, utb );
d = iauGst00a ( uta, utb, tta, ttb );
d = iauGst00b ( uta, utb );
d = iauGst06  ( uta, utb, tta, ttb, rnpb );
d = iauGst06a ( uta, utb, tta, ttb );
d = iauGst94  ( uta, utb );
    iauH2fk5  ( rh, dh, drh, ddh, pxh, rvh,
                &r5, &d5, &dr5, &dd5, &px5, &rv5 );
    iauHfk5z  ( rh, dh, date1, date2,
                &r5, &d5, &dr5, &dd5 );
i = iauJd2cal ( dj1, dj2, &iy, &im, &id, &fd );
i = iauJdcalf ( ndp, dj1, dj2, iymdf );
    iauNum00a ( date1, date2, rmatn );
    iauNum00b ( date1, date2, rmatn );
    iauNum06a ( date1, date2, rmatn );
    iauNumat  ( epsa, dpsi, deps, rmatn );
    iauNut00a ( date1, date2, &dpsi, &deps );
    iauNut00b ( date1, date2, &dpsi, &deps );
    iauNut06a ( date1, date2, &dpsi, &deps );
    iauNut80  ( date1, date2, &dpsi, &deps );
    iauNutm80 ( date1, date2, rmatn );
d = iauObl06  ( date1, date2 );
d = iauObl80  ( date1, date2 );
    iauPb06   ( date1, date2, &bzeta, &bz, &btheta );
    iauPfw06  ( date1, date2, &gamb, &phib, &psib, &epsa );
i = iauPlan94 ( date1, date2, np, pv );
    iauPmat00 ( date1, date2, rbp );
    iauPmat06 ( date1, date2, rbp );
    iauPmat76 ( date1, date2, rmatp );
    iauPn00   ( date1, date2, dpsi, deps,
                &epsa, rb, rp, rbp, rn, rbpn );
    iauPn00a  ( date1, date2,
                &dpsi, &deps, &epsa, rb, rp, rbp, rn, rbpn );
    iauPn00b  ( date1, date2,
                &dpsi, &deps, &epsa, rb, rp, rbp, rn, rbpn );
    iauPn06   ( date1, date2, dpsi, deps,
                &epsa, rb, rp, rbp, rn, rbpn );
    iauPn06a  ( date1, date2,
                &dpsi, &deps, &epsa, rb, rp, rbp, rn, rbpn );
    iauPnm00a ( date1, date2, rbpn );
    iauPnm00b ( date1, date2, rbpn );
    iauPnm06a ( date1, date2, rnpb );
    iauPnm80  ( date1, date2, rmatpn );
    iauP06e   ( date1, date2,
                &eps0, &psia, &oma, &bpa, &bqa, &pia, &bpia,
                &epsa, &chia, &za, &zetaa, &thetaa, &pa,
                &gam, &phi, &psi );
    iauPom00  ( xp, yp, sp, rpom );
    iauPr00   ( date1, date2, &dpsipr, &depspr );
    iauPrec76 ( ep01, ep02, ep11, ep12, &zeta, &z, &theta );
i = iauPvstar ( pv, &ra, &dec, &pmr, &pmd, &px, &rv );
d = iauS00    ( date1, date2, x, y );
d = iauS00a   ( date1, date2 );
d = iauS00b   ( date1, date2 );
d = iauS06    ( date1, date2, x, y );
d = iauS06a   ( date1, date2 );
d = iauSp00   ( date1, date2 );
i = iauStarpm ( ra1, dec1, pmr1, pmd1, px1, rv1,
                ep1a, ep1b, ep2a, ep2b,
                &ra2, &dec2, &pmr2, &pmd2, &px2, &rv2 );
i = iauStarpv ( ra, dec, pmr, pmd, px, rv, pv );
i = iauTaitt  ( tai1, tai2, &tt1, &tt2 );
i = iauTaiut1 ( tai1, tai2, dta, &ut11, &ut12 );
i = iauTaiutc ( tai1, tai2, &utc1, &utc2 );
i = iauTcbtdb ( tcb1, tcb2, &tdb1, &tdb2 );
i = iauTcgtt  ( tcg1, tcg2, &tt1, &tt2 );
i = iauTdbtcb ( tdb1, tdb2, &tcb1, &tcb2 );
```

```
i = iauTdbtt  ( tdb1, tdb2, dtr, &tt1, &tt2 );
i = iauTttai  ( tt1, tt2, &tai1, &tai2 );
i = iauTttcg  ( tt1, tt2, &tcg1, &tcg2 );
i = iauTttdb  ( tt1, tt2, dtr, &tdb1, &tdb2 );
i = iauTtut1  ( tt1, tt2, dt, &ut11, &ut12 );
i = iauUt1tai ( ut11, ut12, &tai1, &tai2 );
i = iauUt1tt  ( ut11, ut12, dt, &tt1, &tt2 );
i = iauUt1utc ( ut11, ut12, dut, &utc1, &utc2 );
i = iauUtctai ( utc1, utc2, dta, &tai1, &tai2 );
i = iauUtcut1 ( utc1, utc2, dut, &ut11, &ut12 );
    iauXy06   ( date1, date2, &x, &y );
    iauXys00a ( date1, date2, &x, &y, &s );
    iauXys00b ( date1, date2, &x, &y, &s );
    iauXys06a ( date1, date2, &x, &y, &s );
```

SOFA Vector/Matrix Library

PREFACE

The routines described here comprise the SOFA vector/matrix library. Their general appearance and coding style conforms to conventions agreed by the SOFA Review Board, and their functions, names and algorithms have been ratified by the Board. Procedures for soliciting and agreeing additions to the library are still evolving.

PROGRAMMING LANGUAGES

The SOFA routines are available in two programming languages at present: Fortran 77 and ANSI C.

There is a one-to-one relationship between the two language versions. The naming convention is such that a SOFA routine referred to generically as "EXAMPL" exists as a Fortran subprogram iau_EXAMPL and a C function iauExampl. The calls for the two versions are very similar, with the same arguments in the same order. In a few cases, the C equivalent of a Fortran SUBROUTINE subprogram uses a return value rather than an argument.

GENERAL PRINCIPLES

The library consists mostly of routines which operate on ordinary Cartesian vectors (x,y,z) and 3x3 rotation matrices. However, there is also support for vectors which represent velocity as well as position and vectors which represent rotation instead of position. The vectors which represent both position and velocity may be considered still to have dimensions (3), but to comprise elements each of which is two numbers, representing the value itself and the time derivative. Thus:

* "Position" or "p" vectors (or just plain 3-vectors) have dimension (3) in Fortran and [3] in C.

* "Position/velocity" or "pv" vectors have dimensions (3,2) in Fortran and [2][3] in C.

* "Rotation" or "r" matrices have dimensions (3,3) in Fortran and [3][3] in C. When used for rotation, they are "orthogonal"; the inverse of such a matrix is equal to the transpose. Most of the routines in this library do not assume that r-matrices are necessarily orthogonal and in fact work on any 3x3 matrix.

* "Rotation" or "r" vectors have dimensions (3) in Fortran and [3] in C. Such vectors are a combination of the Euler axis and angle and are convertible to and from r-matrices. The direction is the axis of rotation and the magnitude is the angle of rotation, in radians. Because the amount of rotation can be scaled up and down simply by multiplying the vector by a scalar, r-vectors are useful for representing spins about an axis which is fixed.

* The above rules mean that in terms of memory address, the three velocity components of a pv-vector follow the three position components. Application code is permitted to exploit this and all other knowledge of the internal layouts: that x, y and z appear in that order and are in a right-handed Cartesian coordinate system etc. For example, the cp function (copy a p-vector) can be used to copy the velocity component of a pv-vector (indeed, this is how the CPV routine is coded).

* The routines provided do not completely fill the range of operations that link all the various vector and matrix options, but are confined to functions that are required by other parts of the SOFA software or which are likely to prove useful.

In addition to the vector/matrix routines, the library contains some routines related to spherical angles, including conversions to and from sexagesimal format.

Using the library requires knowledge of vector/matrix methods, spherical trigonometry, and methods of attitude representation. These topics are covered in many textbooks, including "Spacecraft Attitude Determination and Control", James R. Wertz (ed.), Astrophysics and Space Science Library, Vol. 73, D. Reidel Publishing Company, 1986.

OPERATIONS INVOLVING P-VECTORS AND R-MATRICES

Initialize

```
ZP        zero p-vector
ZR        initialize r-matrix to null
IR        initialize r-matrix to identity
```

Copy/extend/extract

```
CP        copy p-vector
CR        copy r-matrix
```

Build rotations

```
RX        rotate r-matrix about x
RY        rotate r-matrix about y
RZ        rotate r-matrix about z
```

Spherical/Cartesian conversions

```
S2C       spherical to unit vector
C2S       unit vector to spherical
S2P       spherical to p-vector
P2S       p-vector to spherical
```

Operations on vectors

```
PPP       p-vector plus p-vector
PMP       p-vector minus p-vector
PPSP      p-vector plus scaled p-vector
PDP       inner (=scalar=dot) product of two p-vectors
PXP       outer (=vector=cross) product of two p-vectors
PM        modulus of p-vector
PN        normalize p-vector returning modulus
SXP       multiply p-vector by scalar
```

Operations on matrices

```
RXR       r-matrix multiply
TR        transpose r-matrix
```

Matrix-vector products

```
RXP       product of r-matrix and p-vector
TRXP      product of transpose of r-matrix and p-vector
```

Separation and position-angle

```
SEPP      angular separation from p-vectors
SEPS      angular separation from spherical coordinates
PAP       position-angle from p-vectors
PAS       position-angle from spherical coordinates
```

Rotation vectors

```
RV2M      r-vector to r-matrix
RM2V      r-matrix to r-vector
```

OPERATIONS INVOLVING PV-VECTORS

```
  Initialize

     ZPV       zero pv-vector

  Copy/extend/extract

     CPV       copy pv-vector
     P2PV      append zero velocity to p-vector
     PV2P      discard velocity component of pv-vector

  Spherical/Cartesian conversions

     S2PV      spherical to pv-vector
     PV2S      pv-vector to spherical

  Operations on vectors

     PVPPV     pv-vector plus pv-vector
     PVMPV     pv-vector minus pv-vector
     PVDPV     inner (=scalar=dot) product of two pv-vectors
     PVXPV     outer (=vector=cross) product of two pv-vectors
     PVM       modulus of pv-vector
     SXPV      multiply pv-vector by scalar
     S2XPV     multiply pv-vector by two scalars
     PVU       update pv-vector
     PVUP      update pv-vector discarding velocity

  Matrix-vector products

     RXPV      product of r-matrix and pv-vector
     TRXPV     product of transpose of r-matrix and pv-vector


OPERATIONS ON ANGLES

     ANP       normalize radians to range 0 to 2pi
     ANPM      normalize radians to range -pi to +pi
     A2TF      decompose radians into hours, minutes, seconds
     A2AF      decompose radians into degrees, arcminutes, arcseconds
     AF2A      degrees, arcminutes, arcseconds to radians
     D2TF      decompose days into hours, minutes, seconds
     TF2A      hours, minutes, seconds to radians
     TF2D      hours, minutes, seconds to days


CALLS: FORTRAN VERSION

   CALL iau_A2AF  ( NDP, ANGLE, SIGN, IDMSF )
   CALL iau_A2TF  ( NDP, ANGLE, SIGN, IHMSF )
   CALL iau_AF2A  ( S, IDEG, IAMIN, ASEC, RAD, J )
   D =  iau_ANP   ( A )
   D =  iau_ANPM  ( A )
   CALL iau_C2S   ( P, THETA, PHI )
   CALL iau_CP    ( P, C )
   CALL iau_CPV   ( PV, C )
   CALL iau_CR    ( R, C )
   CALL iau_D2TF  ( NDP, DAYS, SIGN, IHMSF )
   CALL iau_IR    ( R )
   CALL iau_P2PV  ( P, PV )
   CALL iau_P2S   ( P, THETA, PHI, R )
   CALL iau_PAP   ( A, B, THETA )
   CALL iau_PAS   ( AL, AP, BL, BP, THETA )
   CALL iau_PDP   ( A, B, ADB )
   CALL iau_PM    ( P, R )
   CALL iau_PMP   ( A, B, AMB )
   CALL iau_PN    ( P, R, U )
   CALL iau_PPP   ( A, B, APB )
   CALL iau_PPSP  ( A, S, B, APSB )
   CALL iau_PV2P  ( PV, P )
   CALL iau_PV2S  ( PV, THETA, PHI, R, TD, PD, RD )
   CALL iau_PVDPV ( A, B, ADB )
   CALL iau_PVM   ( PV, R, S )
   CALL iau_PVMPV ( A, B, AMB )
```

```
   CALL iau_PVPPV ( A, B, APB )
   CALL iau_PVU   ( DT, PV, UPV )
   CALL iau_PVUP  ( DT, PV, P )
   CALL iau_PVXPV ( A, B, AXB )
   CALL iau_PXP   ( A, B, AXB )
   CALL iau_RM2V  ( R, P )
   CALL iau_RV2M  ( P, R )
   CALL iau_RX    ( PHI, R )
   CALL iau_RXP   ( R, P, RP )
   CALL iau_RXPV  ( R, PV, RPV )
   CALL iau_RXR   ( A, B, ATB )
   CALL iau_RY    ( THETA, R )
   CALL iau_RZ    ( PSI, R )
   CALL iau_S2C   ( THETA, PHI, C )
   CALL iau_S2P   ( THETA, PHI, R, P )
   CALL iau_S2PV  ( THETA, PHI, R, TD, PD, RD, PV )
   CALL iau_S2XPV ( S1, S2, PV )
   CALL iau_SEPP  ( A, B, S )
   CALL iau_SEPS  ( AL, AP, BL, BP, S )
   CALL iau_SXP   ( S, P, SP )
   CALL iau_SXPV  ( S, PV, SPV )
   CALL iau_TF2A  ( S, IHOUR, IMIN, SEC, RAD, J )
   CALL iau_TF2D  ( S, IHOUR, IMIN, SEC, DAYS, J )
   CALL iau_TR    ( R, RT )
   CALL iau_TRXP  ( R, P, TRP )
   CALL iau_TRXPV ( R, PV, TRPV )
   CALL iau_ZP    ( P )
   CALL iau_ZPV   ( PV )
   CALL iau_ZR    ( R )
```

CALLS: C VERSION

```
       iauA2af  ( ndp, angle, &sign, idmsf );
       iauA2tf  ( ndp, angle, &sign, ihmsf );
   i = iauAf2a  ( s, ideg, iamin, asec, &rad );
   d = iauAnp   ( a );
   d = iauAnpm  ( a );
       iauC2s   ( p, &theta, &phi );
       iauCp    ( p, c );
       iauCpv   ( pv, c );
       iauCr    ( r, c );
       iauD2tf  ( ndp, days, &sign, ihmsf );
       iauIr    ( r );
       iauP2pv  ( p, pv );
       iauP2s   ( p, &theta, &phi, &r );
   d = iauPap   ( a, b );
   d = iauPas   ( al, ap, bl, bp );
   d = iauPdp   ( a, b );
   d = iauPm    ( p );
       iauPmp   ( a, b, amb );
       iauPn    ( p, &r, u );
       iauPpp   ( a, b, apb );
       iauPpsp  ( a, s, b, apsb );
       iauPv2p  ( pv, p );
       iauPv2s  ( pv, &theta, &phi, &r, &td, &pd, &rd );
       iauPvdpv ( a, b, adb );
       iauPvm   ( pv, &r, &s );
       iauPvmpv ( a, b, amb );
       iauPvppv ( a, b, apb );
       iauPvu   ( dt, pv, upv );
       iauPvup  ( dt, pv, p );
       iauPvxpv ( a, b, axb );
       iauPxp   ( a, b, axb );
       iauRm2v  ( r, p );
       iauRv2m  ( p, r );
       iauRx    ( phi, r );
       iauRxp   ( r, p, rp );
       iauRxpv  ( r, pv, rpv );
       iauRxr   ( a, b, atb );
       iauRy    ( theta, r );
       iauRz    ( psi, r );
       iauS2c   ( theta, phi, c );
```

```
    iauS2p   ( theta, phi, r, p );
    iauS2pv  ( theta, phi, r, td, pd, rd, pV );
    iauS2xpv ( s1, s2, pv );
d = iauSepp  ( a, b );
d = iauSeps  ( al, ap, bl, bp );
    iauSxp   ( s, p, sp );
    iauSxpv  ( s, pv, spv );
i = iauTf2a  ( s, ihour, imin, sec, &rad );
i = iauTf2d  ( s, ihour, imin, sec, &days );
    iauTr    ( r, rt );
    iauTrxp  ( r, p, trp );
    iauTrxpv ( r, pv, trpv );
    iauZp    ( p );
    iauZpv   ( pv );
    iauZr    ( r );
```

```
      SUBROUTINE iau_A2AF ( NDP, ANGLE, SIGN, IDMSF )
*+
*  - - - - - - - - -
*   i a u _ A 2 A F
*  - - - - - - - - -
*
*  Decompose radians into degrees, arcminutes, arcseconds, fraction.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     NDP       i         resolution (Note 1)
*     ANGLE     d         angle in radians
*
*  Returned:
*     SIGN      c         '+' or '-'
*     IDMSF     i(4)      degrees, arcminutes, arcseconds, fraction
*
*  Called:
*     iau_D2TF     decompose days to hms
*
*  Notes:
*
*  1) NDP is interpreted as follows:
*
*     NDP         resolution
*      :      ...0000 00 00
*     -7         1000 00 00
*     -6          100 00 00
*     -5           10 00 00
*     -4            1 00 00
*     -3            0 10 00
*     -2            0 01 00
*     -1            0 00 10
*      0            0 00 01
*      1            0 00 00.1
*      2            0 00 00.01
*      3            0 00 00.001
*      :            0 00 00.000...
*
*  2) The largest positive useful value for NDP is determined by the
*     size of ANGLE, the format of DOUBLE PRECISION floating-point
*     numbers on the target platform, and the risk of overflowing
*     IDMSF(4).  On a typical platform, for ANGLE up to 2pi, the
*     available floating-point precision might correspond to NDP=12.
*     However, the practical limit is typically NDP=9, set by the
*     capacity of a 32-bit IDMSF(4).
*
*  3) The absolute value of ANGLE may exceed 2pi.  In cases where it
*     does not, it is up to the caller to test for and handle the
*     case where ANGLE is very nearly 2pi and rounds up to 360 degrees,
*     by testing for IDMSF(1)=360 and setting IDMSF(1-4) to zero.
*
*-
```

```
      SUBROUTINE iau_A2TF ( NDP, ANGLE, SIGN, IHMSF )
*+
*  - - - - - - - - - -
*   i a u _ A 2 T F
*  - - - - - - - - - -
*
*  Decompose radians into hours, minutes, seconds, fraction.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     NDP       i        resolution (Note 1)
*     ANGLE     d        angle in radians
*
*  Returned:
*     SIGN      c        '+' or '-'
*     IHMSF     i(4)     hours, minutes, seconds, fraction
*
*  Called:
*     iau_D2TF     decompose days to hms
*
*  Notes:
*
*  1) NDP is interpreted as follows:
*
*     NDP         resolution
*      :      ...0000 00 00
*     -7         1000 00 00
*     -6          100 00 00
*     -5           10 00 00
*     -4            1 00 00
*     -3            0 10 00
*     -2            0 01 00
*     -1            0 00 10
*      0            0 00 01
*      1            0 00 00.1
*      2            0 00 00.01
*      3            0 00 00.001
*      :            0 00 00.000...
*
*  2) The largest useful value for NDP is determined by the size
*     of ANGLE, the format of DOUBLE PRECISION floating-point numbers
*     on the target platform, and the risk of overflowing IHMSF(4).
*     On a typical platform, for ANGLE up to 2pi, the available
*     floating-point precision might correspond to NDP=12.  However,
*     the practical limit is typically NDP=9, set by the capacity of
*     a 32-bit IHMSF(4).
*
*  3) The absolute value of ANGLE may exceed 2pi.  In cases where it
*     does not, it is up to the caller to test for and handle the
*     case where ANGLE is very nearly 2pi and rounds up to 24 hours,
*     by testing for IHMSF(1)=24 and setting IHMSF(1-4) to zero.
*
*-
```

```
      SUBROUTINE iau_AF2A ( S, IDEG, IAMIN, ASEC, RAD, J )
*+
*  - - - - - - - - - -
*   i a u _ A F 2 A
*  - - - - - - - - - -
*
*  Convert degrees, arcminutes, arcseconds to radians.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     S          c      sign:  '-' = negative, otherwise positive
*     IDEG       i      degrees
*     IAMIN      i      arcminutes
*     ASEC       d      arcseconds
*
*  Returned:
*     RAD        d      angle in radians
*     J          i      status:  0 = OK
*                                1 = IDEG outside range 0-359
*                                2 = IAMIN outside range 0-59
*                                3 = ASEC outside range 0-59.999...
*
*  Notes:
*
*  1)  If the s argument is a string, only the leftmost character is
*      used and no warning status is provided.
*
*  2)  The result is computed even if any of the range checks fail.
*
*  3)  Negative IDEG, IAMIN and/or ASEC produce a warning status, but
*      the absolute value is used in the conversion.
*
*  4)  If there are multiple errors, the status value reflects only the
*      first, the smallest taking precedence.
*
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_ANP ( A )
*+
*  - - - - - - - -
*   i a u _ A N P
*  - - - - - - - -
*
*  Normalize angle into the range 0 <= A < 2pi.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A          d       angle (radians)
*
*  Returned:
*     iau_ANP    d       angle in range 0-2pi
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_ANPM ( A )
*+
*  - - - - - - - - -
*   i a u _ A N P M
*  - - - - - - - - -
*
*  Normalize angle into the range -pi <= A < +pi.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A          d       angle (radians)
*
*  Returned:
*     iau_ANPM   d       angle in range +/-pi
*
*-
```

```
      SUBROUTINE iau_BI00 ( DPSIBI, DEPSBI, DRA )
*+
*  - - - - - - - - - -
*   i a u _ B I 0 0
*  - - - - - - - - - -
*
*  Frame bias components of IAU 2000 precession-nutation models (part of
*  MHB2000 with additions).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Returned:
*     DPSIBI,DEPSBI  d   longitude and obliquity corrections
*     DRA            d   the ICRS RA of the J2000.0 mean equinox
*
*  Notes:
*
*  1) The frame bias corrections in longitude and obliquity (radians)
*     are required in order to correct for the offset between the GCRS
*     pole and the J2000.0 mean pole.  They define, with respect to the
*     GCRS frame, a J2000.0 mean pole that is consistent with the rest
*     of the IAU 2000A precession-nutation model.
*
*  2) In addition to the displacement of the pole, the complete
*     description of the frame bias requires also an offset in right
*     ascension.  This is not part of the IAU 2000A model, and is from
*     Chapront et al. (2002).  It is returned in radians.
*
*  3) This is a supplemented implementation of one aspect of the IAU
*     2000A nutation model, formally adopted by the IAU General Assembly
*     in 2000, namely MHB2000 (Mathews et al. 2002).
*
*  References:
*
*     Chapront, J., Chapront-Touze, M. & Francou, G., Astron.Astrophys.,
*     387, 700, 2002.
*
*     Mathews, P.M., Herring, T.A., Buffet, B.A., "Modeling of nutation
*     and precession   New nutation series for nonrigid Earth and
*     insights into the Earth's interior", J.Geophys.Res., 107, B4,
*     2002.  The MHB2000 code itself was obtained on 9th September 2002
*     from ftp://maia.usno.navy.mil/conv2000/chapter5/IAU2000A.
*
*-
```

```
      SUBROUTINE iau_BP00 ( DATE1, DATE2, RB, RP, RBP )
*+
*  - - - - - - - - - -
*   i a u _ B P 0 0
*  - - - - - - - - - -
*
*  Frame bias and precession, IAU 2000.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RB             d(3,3)    frame bias matrix (Note 2)
*     RP             d(3,3)    precession matrix (Note 3)
*     RBP            d(3,3)    bias-precession matrix (Note 4)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix RB transforms vectors from GCRS to mean J2000.0 by
*     applying frame bias.
*
*  3) The matrix RP transforms vectors from J2000.0 mean equator and
*     equinox to mean equator and equinox of date by applying
*     precession.
*
*  4) The matrix RBP transforms vectors from GCRS to mean equator and
*     equinox of date by applying frame bias then precession.  It is the
*     product RP x RB.
*
*  Called:
*     iau_BI00     frame bias components, IAU 2000
*     iau_PR00     IAU 2000 precession adjustments
*     iau_IR       initialize r-matrix to identity
*     iau_RX       rotate around X-axis
*     iau_RY       rotate around Y-axis
*     iau_RZ       rotate around Z-axis
*     iau_RXR      product of two r-matrices
*
*  Reference:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003)
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
```

*–

```
      SUBROUTINE iau_BP06 ( DATE1, DATE2, RB, RP, RBP )
*+
*  - - - - - - - - -
*   i a u _ B P 0 6
*  - - - - - - - - -
*
*  Frame bias and precession, IAU 2006.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RB             d(3,3)    frame bias matrix (Note 2)
*     RP             d(3,3)    precession matrix (Note 3)
*     RBP            d(3,3)    bias-precession matrix (Note 4)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix RB transforms vectors from GCRS to mean J2000.0 by
*     applying frame bias.
*
*  3) The matrix RP transforms vectors from mean J2000.0 to mean of date
*     by applying precession.
*
*  4) The matrix RBP transforms vectors from GCRS to mean of date by
*     applying frame bias then precession.  It is the product RP x RB.
*
*  Called:
*     iau_PFW06    bias-precession F-W angles, IAU 2006
*     iau_FW2M     F-W angles to r-matrix
*     iau_PMAT06   PB matrix, IAU 2006
*     iau_TR       transpose r-matrix
*     iau_RXR      product of two r-matrices
*
*  References:
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      SUBROUTINE iau_BPN2XY ( RBPN, X, Y )
*+
*  - - - - - - - - - - -
*   i a u _ B P N 2 X Y
*  - - - - - - - - - - -
*
*  Extract from the bias-precession-nutation matrix the X,Y coordinates
*  of the Celestial Intermediate Pole.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     RBPN      d(3,3)    celestial-to-true matrix (Note 1)
*
*  Returned:
*     X,Y         d       Celestial Intermediate Pole (Note 2)
*
*  Notes:
*
*  1) The matrix RBPN transforms vectors from GCRS to true equator (and
*     CIO or equinox) of date, and therefore the Celestial Intermediate
*     Pole unit vector is the bottom row of the matrix.
*
*  2) X,Y are components of the Celestial Intermediate Pole unit vector
*     in the Geocentric Celestial Reference System.
*
*  Reference:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003)
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*-
```

```
      SUBROUTINE iau_C2I00A ( DATE1, DATE2, RC2I )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 I 0 0 A
*  - - - - - - - - - - -
*
*  Form the celestial-to-intermediate matrix for a given date using the
*  IAU 2000A precession-nutation model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RC2I         d(3,3)    celestial-to-intermediate matrix (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix RC2I is the first stage in the transformation from
*     celestial to terrestrial coordinates:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), ERA is the Earth
*     Rotation Angle and RPOM is the polar motion matrix.
*
*  3) A faster, but slightly less accurate result (about 1 mas), can be
*     obtained by using instead the iau_C2I00B routine.
*
*  Called:
*     iau_PNM00A   classical NPB matrix, IAU 2000A
*     iau_C2IBPN   celestial-to-intermediate matrix, given NPB matrix
*
*  References:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003)
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
```

*
*_

```
      SUBROUTINE iau_C2I00B ( DATE1, DATE2, RC2I )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 I 0 0 B
*  - - - - - - - - - - -
*
*  Form the celestial-to-intermediate matrix for a given date using the
*  IAU 2000B precession-nutation model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RC2I         d(3,3)    celestial-to-intermediate matrix (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix RC2I is the first stage in the transformation from
*     celestial to terrestrial coordinates:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), ERA is the Earth
*     Rotation Angle and RPOM is the polar motion matrix.
*
*  3) The present routine is faster, but slightly less accurate (about
*     1 mas), than the iau_C2I00A routine.
*
*  Called:
*     iau_PNM00B   classical NPB matrix, IAU 2000B
*     iau_C2IBPN   celestial-to-intermediate matrix, given NPB matrix
*
*  References:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003)
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
```

*
*_

```
      SUBROUTINE iau_C2I06A ( DATE1, DATE2, RC2I )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 I 0 6 A
*  - - - - - - - - - - -
*
*  Form the celestial-to-intermediate matrix for a given date using the
*  IAU 2006 precession and IAU 2000A nutation models.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RC2I         d(3,3)    celestial-to-intermediate matrix (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix RC2I is the first stage in the transformation from
*     celestial to terrestrial coordinates:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), ERA is the Earth
*     Rotation Angle and RPOM is the polar motion matrix.
*
*  Called:
*     iau_PNM06A   classical NPB matrix, IAU 2006/2000A
*     iau_BPN2XY   extract CIP X,Y coordinates from NPB matrix
*     iau_S06      the CIO locator s, given X,Y, IAU 2006
*     iau_C2IXYS   celestial-to-intermediate matrix, given X,Y and s
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), 2004, IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      SUBROUTINE iau_C2IBPN ( DATE1, DATE2, RBPN, RC2I )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 I B P N
*  - - - - - - - - - - -
*
*  Form the celestial-to-intermediate matrix for a given date given
*  the bias-precession-nutation matrix.  IAU 2000.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*     RBPN         d(3,3)    celestial-to-true matrix (Note 2)
*
*  Returned:
*     RC2I         d(3,3)    celestial-to-intermediate matrix (Note 3)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix RBPN transforms vectors from GCRS to true equator (and
*     CIO or equinox) of date.  Only the CIP (bottom row) is used.
*
*  3) The matrix RC2I is the first stage in the transformation from
*     celestial to terrestrial coordinates:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), ERA is the Earth
*     Rotation Angle and RPOM is the polar motion matrix.
*
*  4) Although its name does not include "00", this routine is in fact
*     specific to the IAU 2000 models.
*
*  Called:
*     iau_BPN2XY   extract CIP X,Y coordinates from NPB matrix
*     iau_C2IXY    celestial-to-intermediate matrix, given X,Y
*
*  References:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003)
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
```

```
*           intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_C2IXY ( DATE1, DATE2, X, Y, RC2I )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 I X Y
*  - - - - - - - - - - -
*
*  Form the celestial to intermediate-frame-of-date matrix for a given
*  date when the CIP X,Y coordinates are known.  IAU 2000.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*     X,Y            d      Celestial Intermediate Pole (Note 2)
*
*  Returned:
*     RC2I         d(3,3)   celestial-to-intermediate matrix (Note 3)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The Celestial Intermediate Pole coordinates are the x,y components
*     of the unit vector in the Geocentric Celestial Reference System.
*
*  3) The matrix RC2I is the first stage in the transformation from
*     celestial to terrestrial coordinates:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), ERA is the Earth
*     Rotation Angle and RPOM is the polar motion matrix.
*
*  4) Although its name does not include "00", this routine is in fact
*     specific to the IAU 2000 models.
*
*  Called:
*     iau_C2IXYS   celestial-to-intermediate matrix, given X,Y and s
*     iau_S00      the CIO locator s, given X,Y, IAU 2000A
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_C2IXYS ( X, Y, S, RC2I )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 I X Y S
*  - - - - - - - - - - -
*
*  Form the celestial to intermediate-frame-of-date matrix given the CIP
*  X,Y and the CIO locator s.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     X,Y        d       Celestial Intermediate Pole (Note 1)
*     S          d       the CIO locator s (Note 2)
*
*  Returned:
*     RC2I     d(3,3)    celestial-to-intermediate matrix (Note 3)
*
*  Notes:
*
*  1) The Celestial Intermediate Pole coordinates are the x,y components
*     of the unit vector in the Geocentric Celestial Reference System.
*
*  2) The CIO locator s (in radians) positions the Celestial
*     Intermediate Origin on the equator of the CIP.
*
*  3) The matrix RC2I is the first stage in the transformation from
*     celestial to terrestrial coordinates:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), ERA is the Earth
*     Rotation Angle and RPOM is the polar motion matrix.
*
*  Called:
*     iau_IR       initialize r-matrix to identity
*     iau_RZ       rotate around Z-axis
*     iau_RY       rotate around Y-axis
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_C2S ( P, THETA, PHI )
*+
*  - - - - - - - -
*   i a u _ C 2 S
*  - - - - - - - -
*
*  P-vector to spherical coordinates.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     P        d(3)      p-vector
*
*  Returned:
*     THETA    d         longitude angle (radians)
*     PHI      d         latitude angle (radians)
*
*  Notes:
*
*  1) P can have any magnitude; only its direction is used.
*
*  2) If P is null, zero THETA and PHI are returned.
*
*  3) At either pole, zero THETA is returned.
*
*-
```

```
      SUBROUTINE iau_C2T00A ( TTA, TTB, UTA, UTB, XP, YP, RC2T )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 T 0 0 A
*  - - - - - - - - - - -
*
*  Form the celestial to terrestrial matrix given the date, the UT1 and
*  the polar motion, using the IAU 2000A nutation model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     TTA,TTB    d       TT as a 2-part Julian Date (Note 1)
*     UTA,UTB    d       UT1 as a 2-part Julian Date (Note 1)
*     XP,YP      d       coordinates of the pole (radians, Note 2)
*
*  Returned:
*     RC2T     d(3,3)    celestial-to-terrestrial matrix (Note 3)
*
*  Notes:
*
*  1) The TT and UT1 dates TTA+TTB and UTA+UTB are Julian Dates,
*     apportioned in any convenient way between the arguments UTA and
*     UTB.  For example, JD(UT1)=2450123.7 could be expressed in any of
*     these ways, among others:
*
*             UTA            UTB
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution is
*     acceptable.  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  In the case of UTA,UTB, the
*     date & time method is best matched to the Earth rotation angle
*     algorithm used:  maximum accuracy (or, at least, minimum noise) is
*     delivered when the UTA argument is for 0hrs UT1 on the day in
*     question and the UTB argument lies in the range 0 to 1, or vice
*     versa.
*
*  2) XP and YP are the coordinates (in radians) of the Celestial
*     Intermediate Pole with respect to the International Terrestrial
*     Reference System (see IERS Conventions 2003), measured along the
*     meridians to 0 and 90 deg west respectively.
*
*  3) The matrix RC2T transforms from celestial to terrestrial
*     coordinates:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), RC2I is the
*     celestial-to-intermediate matrix, ERA is the Earth rotation angle
*     and RPOM is the polar motion matrix.
*
*  4) A faster, but slightly less accurate result (about 1 mas), can be
*     obtained by using instead the iau_C2T00B routine.
*
*  Called:
*     iau_C2I00A   celestial-to-intermediate matrix, IAU 2000A
*     iau_ERA00    Earth rotation angle, IAU 2000
*     iau_SP00     the TIO locator s', IERS 2000
*     iau_POM00    polar motion matrix
```

```
*     iau_C2TCIO   form CIO-based celestial-to-terrestrial matrix
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_C2T00B ( TTA, TTB, UTA, UTB, XP, YP, RC2T )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 T 0 0 B
*  - - - - - - - - - - -
*
*  Form the celestial to terrestrial matrix given the date, the UT1 and
*  the polar motion, using the IAU 2000B nutation model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     TTA,TTB    d       TT as a 2-part Julian Date (Note 1)
*     UTA,UTB    d       UT1 as a 2-part Julian Date (Note 1)
*     XP,YP      d       coordinates of the pole (radians, Note 2)
*
*  Returned:
*     RC2T     d(3,3)    celestial-to-terrestrial matrix (Note 3)
*
*  Notes:
*
*  1) The TT and UT1 dates TTA+TTB and UTA+UTB are Julian Dates,
*     apportioned in any convenient way between the arguments UTA and
*     UTB.  For example, JD(UT1)=2450123.7 could be expressed in any of
*     these ways, among others:
*
*             UTA            UTB
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution is
*     acceptable.  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  In the case of UTA,UTB, the
*     date & time method is best matched to the Earth rotation angle
*     algorithm used:  maximum accuracy (or, at least, minimum noise) is
*     delivered when the UTA argument is for 0hrs UT1 on the day in
*     question and the UTB argument lies in the range 0 to 1, or vice
*     versa.
*
*  2) XP and YP are the coordinates (in radians) of the Celestial
*     Intermediate Pole with respect to the International Terrestrial
*     Reference System (see IERS Conventions 2003), measured along the
*     meridians to 0 and 90 deg west respectively.
*
*  3) The matrix RC2T transforms from celestial to terrestrial
*     coordinates:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), RC2I is the
*     celestial-to-intermediate matrix, ERA is the Earth rotation angle
*     and RPOM is the polar motion matrix.
*
*  4) The present routine is faster, but slightly less accurate (about
*     1 mas), than the iau_C2T00A routine.
*
*  Called:
*     iau_C2I00B   celestial-to-intermediate matrix, IAU 2000B
*     iau_ERA00    Earth rotation angle, IAU 2000
*     iau_POM00    polar motion matrix
*     iau_C2TCIO   form CIO-based celestial-to-terrestrial matrix
```

```
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_C2T06A ( TTA, TTB, UTA, UTB, XP, YP, RC2T )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 T 0 6 A
*  - - - - - - - - - - -
*
*  Form the celestial to terrestrial matrix given the date, the UT1 and
*  the polar motion, using the IAU 2006 precession and IAU 2000A
*  nutation models.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     TTA,TTB    d       TT as a 2-part Julian Date (Note 1)
*     UTA,UTB    d       UT1 as a 2-part Julian Date (Note 1)
*     XP,YP      d       coordinates of the pole (radians, Note 2)
*
*  Returned:
*     RC2T     d(3,3)    celestial-to-terrestrial matrix (Note 3)
*
*  Notes:
*
*  1) The TT and UT1 dates TTA+TTB and UTA+UTB are Julian Dates,
*     apportioned in any convenient way between the arguments UTA and
*     UTB.  For example, JD(UT1)=2450123.7 could be expressed in any of
*     these ways, among others:
*
*             UTA            UTB
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution is
*     acceptable.  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  In the case of UTA,UTB, the
*     date & time method is best matched to the Earth rotation angle
*     algorithm used:  maximum accuracy (or, at least, minimum noise) is
*     delivered when the UTA argument is for 0hrs UT1 on the day in
*     question and the UTB argument lies in the range 0 to 1, or vice
*     versa.
*
*  2) XP and YP are the coordinates (in radians) of the Celestial
*     Intermediate Pole with respect to the International Terrestrial
*     Reference System (see IERS Conventions 2003), measured along the
*     meridians to 0 and 90 deg west respectively.
*
*  3) The matrix RC2T transforms from celestial to terrestrial
*     coordinates:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), RC2I is the
*     celestial-to-intermediate matrix, ERA is the Earth rotation angle
*     and RPOM is the polar motion matrix.
*
*  Called:
*     iau_C2I06A   celestial-to-intermediate matrix, IAU 2006/2000A
*     iau_ERA00    Earth rotation angle, IAU 2000
*     iau_SP00     the TIO locator s', IERS 2000
*     iau_POM00    polar motion matrix
*     iau_C2TCIO   form CIO-based celestial-to-terrestrial matrix
*
```

```
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), 2004, IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG
*
*-
```

```
      SUBROUTINE iau_C2TCEO ( RC2I, ERA, RPOM, RC2T )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 T C E O
*  - - - - - - - - - - -
*
*  Assemble the celestial to terrestrial matrix from CIO-based
*  components (the celestial-to-intermediate matrix, the Earth Rotation
*  Angle and the polar motion matrix).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  obsolete routine.
*
*  Given:
*     RC2I     d(3,3)    celestial-to-intermediate matrix
*     ERA        d       Earth rotation angle
*     RPOM     d(3,3)    polar-motion matrix
*
*  Returned:
*     RC2T     d(3,3)    celestial-to-terrestrial matrix
*
*  Notes:
*
*  1) The name of the present routine, iau_C2TCEO, reflects the original
*     name of the celestial intermediate origin (CIO), which before the
*     adoption of IAU 2006 Resolution 2 was called the "celestial
*     ephemeris origin" (CEO).
*
*  2) When the name change from CEO to CIO occurred, a new SOFA routine
*     called iau_C2TCIO was introduced as the successor to the existing
*     iau_C2TCEO.  The present routine is merely a front end to the new
*     one.
*
*  3) The present routine is included in the SOFA collection only to
*     support existing applications.  It should not be used in new
*     applications.
*
*  Called:
*     iau_C2TCIO   form CIO-based celestial-to-terrestrial matrix
*
*-
```

```
      SUBROUTINE iau_C2TCIO ( RC2I, ERA, RPOM, RC2T )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 T C I O
*  - - - - - - - - - - -
*
*  Assemble the celestial to terrestrial matrix from CIO-based
*  components (the celestial-to-intermediate matrix, the Earth Rotation
*  Angle and the polar motion matrix).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     RC2I     d(3,3)    celestial-to-intermediate matrix
*     ERA        d       Earth rotation angle
*     RPOM     d(3,3)    polar-motion matrix
*
*  Returned:
*     RC2T     d(3,3)    celestial-to-terrestrial matrix
*
*  Notes:
*
*  1) This routine constructs the rotation matrix that transforms
*     vectors in the celestial system into vectors in the terrestrial
*     system.  It does so starting from precomputed components, namely
*     the matrix which rotates from celestial coordinates to the
*     intermediate frame, the Earth rotation angle and the polar motion
*     matrix.  One use of the present routine is when generating a
*     series of celestial-to-terrestrial matrices where only the Earth
*     Rotation Angle changes, avoiding the considerable overhead of
*     recomputing the precession-nutation more often than necessary to
*     achieve given accuracy objectives.
*
*  2) The relationship between the arguments is as follows:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003).
*
*  Called:
*     iau_CR       copy r-matrix
*     iau_RZ       rotate around Z-axis
*     iau_RXR      product of two r-matrices
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), 2004, IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG
*
*-
```

```
      SUBROUTINE iau_C2TEQX ( RBPN, GST, RPOM, RC2T )
*+
*  - - - - - - - - - - -
*   i a u _ C 2 T E Q X
*  - - - - - - - - - - -
*
*  Assemble the celestial to terrestrial matrix from equinox-based
*  components (the celestial-to-true matrix, the Greenwich Apparent
*  Sidereal Time and the polar motion matrix).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     RBPN     d(3,3)    celestial-to-true matrix
*     GST        d       Greenwich (apparent) Sidereal Time
*     RPOM     d(3,3)    polar-motion matrix
*
*  Returned:
*     RC2T     d(3,3)    celestial-to-terrestrial matrix (Note 2)
*
*  Notes:
*
*  1) This routine constructs the rotation matrix that transforms
*     vectors in the celestial system into vectors in the terrestrial
*     system.  It does so starting from precomputed components, namely
*     the matrix which rotates from celestial coordinates to the
*     true equator and equinox of date, the Greenwich Apparent Sidereal
*     Time and the polar motion matrix.  One use of the present routine
*     is when generating a series of celestial-to-terrestrial matrices
*     where only the Sidereal Time changes, avoiding the considerable
*     overhead of recomputing the precession-nutation more often than
*     necessary to achieve given accuracy objectives.
*
*  2) The relationship between the arguments is as follows:
*
*        [TRS]  =  RPOM * R_3(GST) * RBPN * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003).
*
*  Called:
*     iau_CR       copy r-matrix
*     iau_RZ       rotate around Z-axis
*     iau_RXR      product of two r-matrices
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_C2TPE ( TTA, TTB, UTA, UTB, DPSI, DEPS, XP, YP,
     :                       RC2T )
*+
*  - - - - - - - - - -
*   i a u _ C 2 T P E
*  - - - - - - - - - -
*
*  Form the celestial to terrestrial matrix given the date, the UT1, the
*  nutation and the polar motion.  IAU 2000.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     TTA,TTB     d       TT as a 2-part Julian Date (Note 1)
*     UTA,UTB     d       UT1 as a 2-part Julian Date (Note 1)
*     DPSI,DEPS   d       nutation (Note 2)
*     XP,YP       d       coordinates of the pole (radians, Note 3)
*
*  Returned:
*     RC2T      d(3,3)    celestial-to-terrestrial matrix (Note 4)
*
*  Notes:
*
*  1) The TT and UT1 dates TTA+TTB and UTA+UTB are Julian Dates,
*     apportioned in any convenient way between the arguments UTA and
*     UTB.  For example, JD(UT1)=2450123.7 could be expressed in any of
*     these ways, among others:
*
*             UTA            UTB
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution is
*     acceptable.  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  In the case of UTA,UTB, the
*     date & time method is best matched to the Earth rotation angle
*     algorithm used:  maximum accuracy (or, at least, minimum noise) is
*     delivered when the UTA argument is for 0hrs UT1 on the day in
*     question and the UTB argument lies in the range 0 to 1, or vice
*     versa.
*
*  2) The caller is responsible for providing the nutation components;
*     they are in longitude and obliquity, in radians and are with
*     respect to the equinox and ecliptic of date.  For high-accuracy
*     applications, free core nutation should be included as well as
*     any other relevant corrections to the position of the CIP.
*
*  3) XP and YP are the coordinates (in radians) of the Celestial
*     Intermediate Pole with respect to the International Terrestrial
*     Reference System (see IERS Conventions 2003), measured along the
*     meridians to 0 and 90 deg west respectively.
*
*  4) The matrix RC2T transforms from celestial to terrestrial
*     coordinates:
*
*        [TRS]  =  RPOM * R_3(GST) * RBPN * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), RBPN is the
*     bias-precession-nutation matrix, GST is the Greenwich (apparent)
*     Sidereal Time and RPOM is the polar motion matrix.
*
```

```
*  5) Although its name does not include "00", this routine is in fact
*     specific to the IAU 2000 models.
*
*  Called:
*     iau_PN00     bias/precession/nutation results, IAU 2000
*     iau_GMST00   Greenwich mean sidereal time, IAU 2000
*     iau_SP00     the TIO locator s', IERS 2000
*     iau_EE00     equation of the equinoxes, IAU 2000
*     iau_POM00    polar motion matrix
*     iau_C2TEQX   form equinox-based celestial-to-terrestrial matrix
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_C2TXY ( TTA, TTB, UTA, UTB, X, Y, XP, YP, RC2T )
*+
*  - - - - - - - - - -
*   i a u _ C 2 T X Y
*  - - - - - - - - - -
*
*  Form the celestial to terrestrial matrix given the date, the UT1, the
*  CIP coordinates and the polar motion.  IAU 2000.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     TTA,TTB    d       TT as a 2-part Julian Date (Note 1)
*     UTA,UTB    d       UT1 as a 2-part Julian Date (Note 1)
*     X,Y        d       Celestial Intermediate Pole (Note 2)
*     XP,YP      d       coordinates of the pole (radians, Note 3)
*
*  Returned:
*     RC2T     d(3,3)    celestial-to-terrestrial matrix (Note 4)
*
*  Notes:
*
*  1) The TT and UT1 dates TTA+TTB and UTA+UTB are Julian Dates,
*     apportioned in any convenient way between the arguments UTA and
*     UTB.  For example, JD(UT1)=2450123.7 could be expressed in any of
*     these ways, among others:
*
*             UTA            UTB
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution is
*     acceptable.  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  In the case of UTA,UTB, the
*     date & time method is best matched to the Earth rotation angle
*     algorithm used:  maximum accuracy (or, at least, minimum noise) is
*     delivered when the UTA argument is for 0hrs UT1 on the day in
*     question and the UTB argument lies in the range 0 to 1, or vice
*     versa.
*
*  2) The Celestial Intermediate Pole coordinates are the x,y components
*     of the unit vector in the Geocentric Celestial Reference System.
*
*  3) XP and YP are the coordinates (in radians) of the Celestial
*     Intermediate Pole with respect to the International Terrestrial
*     Reference System (see IERS Conventions 2003), measured along the
*     meridians to 0 and 90 deg west respectively.
*
*  4) The matrix RC2T transforms from celestial to terrestrial
*     coordinates:
*
*        [TRS]  =  RPOM * R_3(ERA) * RC2I * [CRS]
*
*               =  RC2T * [CRS]
*
*     where [CRS] is a vector in the Geocentric Celestial Reference
*     System and [TRS] is a vector in the International Terrestrial
*     Reference System (see IERS Conventions 2003), ERA is the Earth
*     Rotation Angle and RPOM is the polar motion matrix.
*
*  5) Although its name does not include "00", this routine is in fact
*     specific to the IAU 2000 models.
*
*  Called:
*     iau_C2IXY    celestial-to-intermediate matrix, given X,Y
```

```
*     iau_ERA00    Earth rotation angle, IAU 2000
*     iau_SP00     the TIO locator s', IERS 2000
*     iau_POM00    polar motion matrix
*     iau_C2TCIO   form CIO-based celestial-to-terrestrial matrix
*
* Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_CAL2JD ( IY, IM, ID, DJM0, DJM, J )
*+
*  - - - - - - - - - - -
*   i a u _ C A L 2 J D
*  - - - - - - - - - - -
*
*  Gregorian Calendar to Julian Date.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     IY,IM,ID    i     year, month, day in Gregorian calendar (Note 1)
*
*  Returned:
*     DJM0        d     MJD zero-point: always 2400000.5
*     DJM         d     Modified Julian Date for 0 hrs
*     J           i     status:
*                           0 = OK
*                          -1 = bad year   (Note 3: JD not computed)
*                          -2 = bad month  (JD not computed)
*                          -3 = bad day    (JD computed)
*
*  Notes:
*
*  1) The algorithm used is valid from -4800 March 1, but this
*     implementation rejects dates before -4799 January 1.
*
*  2) The Julian Date is returned in two pieces, in the usual SOFA
*     manner, which is designed to preserve time resolution.  The
*     Julian Date is available as a single number by adding DJM0 and
*     DJM.
*
*  3) In early eras the conversion is from the "Proleptic Gregorian
*     Calendar";  no account is taken of the date(s) of adoption of
*     the Gregorian Calendar, nor is the AD/BC numbering convention
*     observed.
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     Section 12.92 (p604).
*
*-
```

```
      SUBROUTINE iau_CP ( P, C )
*+
*  - - - - - - - -
*   i a u _ C P
*  - - - - - - - -
*
*  Copy a p-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     P        d(3)     p-vector to be copied
*
*  Returned:
*     C        d(3)     copy
*
*-
```

```
      SUBROUTINE iau_CPV ( PV, C )
*+
*  - - - - - - - -
*   i a u _ C P V
*  - - - - - - - -
*
*  Copy a position/velocity vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     PV       d(3,2)    position/velocity vector to be copied
*
*  Returned:
*     C        d(3,2)    copy
*
*  Called:
*     iau_CP       copy p-vector
*
*-
```

```
      SUBROUTINE iau_CR ( R, C )
*+
*  - - - - - - - -
*   i a u _ C R
*  - - - - - - - -
*
*  Copy an r-matrix.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     R        d(3,3)    r-matrix to be copied
*
*  Returned:
*     C        d(3,3)    copy
*
*  Called:
*     iau_CP       copy p-vector
*
*-
```

```
      SUBROUTINE iau_D2DTF ( SCALE, NDP, D1, D2, IY, IM, ID, IHMSF, J )
*+
*  - - - - - - - - - -
*   i a u _ D 2 D T F
*  - - - - - - - - - -
*
*  Format for output a 2-part Julian Date (or in the case of UTC a
*  quasi-JD form that includes special provision for leap seconds).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     SCALE      c*(*)  time scale ID (Note 1)
*     NDP        i      resolution (Note 2)
*     D1,D2      d      time as a 2-part Julian Date (Notes 3,4)
*
*  Returned:
*     IY,IM,ID   i      year, month, day in Gregorian calendar (Note 5)
*     IHMSF      i(4)   hours, minutes, seconds, fraction (Note 1)
*     J          i      status: +1 = dubious year (Note 5)
*                                0 = OK
*                               -1 = unacceptable date (Note 6)
*
*  Notes:
*
*  1) SCALE identifies the time scale.  Only the value 'UTC' (in upper
*     case) is significant, and enables handling of leap seconds (see
*     Note 4).
*
*  2) NDP is the number of decimal places in the seconds field, and can
*     have negative as well as positive values, such as:
*
*     NDP         resolution
*     -4            1 00 00
*     -3            0 10 00
*     -2            0 01 00
*     -1            0 00 10
*      0            0 00 01
*      1            0 00 00.1
*      2            0 00 00.01
*      3            0 00 00.001
*
*     The limits are platform dependent, but a safe range is -5 to +9.
*
*  3) D1+D2 is Julian Date, apportioned in any convenient way between
*     the two arguments, for example where D1 is the Julian Day Number
*     and D2 is the fraction of a day.  In the case of UTC, where the
*     use of JD is problematical, special conventions apply:  see the
*     next note.
*
*  4) JD cannot unambiguously represent UTC during a leap second unless
*     special measures are taken.  The SOFA internal convention is that
*     the quasi-JD day represents UTC days whether the length is 86399,
*     86400 or 86401 SI seconds.
*
*  5) The warning status "dubious year" flags UTCs that predate the
*     introduction of the time scale and that are too far in the future
*     to be trusted.  See iau_DAT for further details.
*
*  6) For calendar conventions and limitations, see iau_CAL2JD.
*
*  Called:
*     iau_JD2CAL   JD to Gregorian calendar
*     iau_D2TF     decompose days to hms
*     iau_DAT      delta(AT) = TAI-UTC
*
*-
```

```
      SUBROUTINE iau_D2TF ( NDP, DAYS, SIGN, IHMSF )
*+
*  - - - - - - - - -
*   i a u _ D 2 T F
*  - - - - - - - - -
*
*  Decompose days to hours, minutes, seconds, fraction.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     NDP       i        resolution (Note 1)
*     DAYS      d        interval in days
*
*  Returned:
*     SIGN      c        '+' or '-'
*     IHMSF     i(4)     hours, minutes, seconds, fraction
*
*  Notes:
*
*  1) NDP is interpreted as follows:
*
*     NDP         resolution
*      :      ...0000 00 00
*     -7         1000 00 00
*     -6          100 00 00
*     -5           10 00 00
*     -4            1 00 00
*     -3            0 10 00
*     -2            0 01 00
*     -1            0 00 10
*      0            0 00 01
*      1            0 00 00.1
*      2            0 00 00.01
*      3            0 00 00.001
*      :            0 00 00.000...
*
*  2) The largest positive useful value for NDP is determined by the
*     size of DAYS, the format of DOUBLE PRECISION floating-point
*     numbers on the target platform, and the risk of overflowing
*     IHMSF(4).  On a typical platform, for DAYS up to 1D0, the
*     available floating-point precision might correspond to NDP=12.
*     However, the practical limit is typically NDP=9, set by the
*     capacity of a 32-bit IHMSF(4).
*
*  3) The absolute value of DAYS may exceed 1D0.  In cases where it
*     does not, it is up to the caller to test for and handle the
*     case where DAYS is very nearly 1D0 and rounds up to 24 hours,
*     by testing for IHMSF(1)=24 and setting IHMSF(1-4) to zero.
*
*-
```

```
      SUBROUTINE iau_DAT ( IY, IM, ID, FD, DELTAT, J )
*+
*  - - - - - - - -
*   i a u _ D A T
*  - - - - - - - -
*
*  For a given UTC date, calculate delta(AT) = TAI-UTC.
*
*     :------------------------------------------:
*     :                                          :
*     :                 IMPORTANT                :
*     :                                          :
*     :  A new version of this routine must be   :
*     :  produced whenever a new leap second is  :
*     :  announced.  There are five items to     :
*     :  change on each such occasion:           :
*     :                                          :
*     :  1) The parameter NDAT must be           :
*     :     increased by 1.                      :
*     :                                          :
*     :  2) The set of DATA statements that      :
*     :     initialize the arrays IDAT and       :
*     :     DATS must be extended by one line.   :
*     :                                          :
*     :  3) The parameter IYV must be set to     :
*     :     the current year.                    :
*     :                                          :
*     :  4) The "Latest leap second" comment     :
*     :     below must be set to the new leap    :
*     :     second date.                         :
*     :                                          :
*     :  5) The "This revision" comment, later,  :
*     :     must be set to the current date.     :
*     :                                          :
*     :  Change (3) must also be carried out     :
*     :  whenever the routine is re-issued,      :
*     :  even if no leap seconds have been       :
*     :  added.                                  :
*     :                                          :
*     :  Latest leap second:  2012 June 30       :
*     :                                          :
*     :__________________________________________:
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     IY       i     UTC:  year (Notes 1 and 2)
*     IM       i           month (Note 2)
*     ID       i           day (Notes 2 and 3)
*     FD       d           fraction of day (Note 4)
*
*  Returned:
*     DELTAT   d     TAI minus UTC, seconds
*     J        i     status (Note 5):
*                       1 = dubious year (Note 1)
*                       0 = OK
*                      -1 = bad year
*                      -2 = bad month
*                      -3 = bad day (Note 3)
*                      -4 = bad fraction (Note 4)
*
*  Notes:
*
*  1) UTC began at 1960 January 1.0 (JD 2436934.5) and it is improper
*     to call the routine with an earlier date.  If this is attempted,
*     zero is returned together with a warning status.
*
*     Because leap seconds cannot, in principle, be predicted in
*     advance, a reliable check for dates beyond the valid range is
```

```
*     impossible.  To guard against gross errors, a year five or more
*     after the release year of the present routine (see parameter IYV)
*     is considered dubious.  In this case a warning status is returned
*     but the result is computed in the normal way.
*
*     For both too-early and too-late years, the warning status is J=+1.
*     This is distinct from the error status J=-1, which signifies a
*     year so early that JD could not be computed.
*
*  2) If the specified date is for a day which ends with a leap second,
*     the UTC-TAI value returned is for the period leading up to the
*     leap second.  If the date is for a day which begins as a leap
*     second ends, the UTC-TAI returned is for the period following the
*     leap second.
*
*  3) The day number must be in the normal calendar range, for example
*     1 through 30 for April.  The "almanac" convention of allowing
*     such dates as January 0 and December 32 is not supported in this
*     routine, in order to avoid confusion near leap seconds.
*
*  4) The fraction of day is used only for dates before the introduction
*     of leap seconds, the first of which occurred at the end of 1971.
*     It is tested for validity (0 to 1 is the valid range) even if not
*     used;  if invalid, zero is used and status J=-4 is returned.  For
*     many applications, setting FD to zero is acceptable;  the
*     resulting error is always less than 3 ms (and occurs only
*     pre-1972).
*
*  5) The status value returned in the case where there are multiple
*     errors refers to the first error detected.  For example, if the
*     month and day are 13 and 32 respectively, J=-2 (bad month) will be
*     returned.
*
*  6) In cases where a valid result is not available, zero is returned.
*
*  References:
*
*  1) For dates from 1961 January 1 onwards, the expressions from the
*     file ftp://maia.usno.navy.mil/ser7/tai-utc.dat are used.
*
*  2) The 5ms timestep at 1961 January 1 is taken from 2.58.1 (p87) of
*     the 1992 Explanatory Supplement.
*
*  Called:
*     iau_CAL2JD   Gregorian calendar to Julian Day number
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_DTDB ( DATE1, DATE2,
     :                                     UT, ELONG, U, V )
*+
*  - - - - - - - - -
*   i a u _ D T D B
*  - - - - - - - - -
*
*  An approximation to TDB-TT, the difference between barycentric
*  dynamical time and terrestrial time, for an observer on the Earth.
*
*  The different time scales - proper, coordinate and realized - are
*  related to each other:
*
*            TAI             <-  physically realized
*             :
*          offset            <-  observed (nominally +32.184s)
*             :
*            TT              <-  terrestrial time
*             :
*    rate adjustment (L_G)   <-  definition of TT
*             :
*            TCG             <-  time scale for GCRS
*             :
*      "periodic" terms      <-  iau_DTDB is an implementation
*             :
*    rate adjustment (L_C)   <-  function of solar-system ephemeris
*             :
*            TCB             <-  time scale for BCRS
*             :
*    rate adjustment (-L_B)  <-  definition of TDB
*             :
*            TDB             <-  TCB scaled to track TT
*             :
*      "periodic" terms      <-  -iau_DTDB is an approximation
*             :
*            TT              <-  terrestrial time
*
*  Adopted values for the various constants can be found in the IERS
*  Conventions (McCarthy & Petit 2003).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2     d    date, TDB (Notes 1-3)
*     UT              d    universal time (UT1, fraction of one day)
*     ELONG           d    longitude (east positive, radians)
*     U               d    distance from Earth spin axis (km)
*     V               d    distance north of equatorial plane (km)
*
*  Returned:
*    iau_DTDB         d    TDB-TT (seconds)
*
*  Notes:
*
*  1) The date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the arguments DATE1 and DATE2.  For
*     example, JD(TDB)=2450123.7 could be expressed in any of these
*     ways, among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in cases
*     where the loss of several decimal digits of resolution is
*     acceptable.  The J2000 method is best matched to the way the
```

```
*     argument is handled internally and will deliver the optimum
*     resolution.  The MJD method and the date & time methods are both
*     good compromises between resolution and convenience.
*
*     Although the date is, formally, barycentric dynamical time (TDB),
*     the terrestrial dynamical time (TT) can be used with no practical
*     effect on the accuracy of the prediction.
*
*  2) TT can be regarded as a coordinate time that is realized as an
*     offset of 32.184s from International Atomic Time, TAI.  TT is a
*     specific linear transformation of geocentric coordinate time TCG,
*     which is the time scale for the Geocentric Celestial Reference
*     System, GCRS.
*
*  3) TDB is a coordinate time, and is a specific linear transformation
*     of barycentric coordinate time TCB, which is the time scale for
*     the Barycentric Celestial Reference System, BCRS.
*
*  4) The difference TCG-TCB depends on the masses and positions of the
*     bodies of the solar system and the velocity of the Earth.  It is
*     dominated by a rate difference, the residual being of a periodic
*     character.  The latter, which is modeled by the present routine,
*     comprises a main (annual) sinusoidal term of amplitude
*     approximately 0.00166 seconds, plus planetary terms up to about
*     20 microseconds, and lunar and diurnal terms up to 2 microseconds.
*     These effects come from the changing transverse Doppler effect
*     and gravitational red-shift as the observer (on the Earth's
*     surface) experiences variations in speed (with respect to the
*     BCRS) and gravitational potential.
*
*  5) TDB can be regarded as the same as TCB but with a rate adjustment
*     to keep it close to TT, which is convenient for many applications.
*     The history of successive attempts to define TDB is set out in
*     Resolution 3 adopted by the IAU General Assembly in 2006, which
*     defines a fixed TDB(TCB) transformation that is consistent with
*     contemporary solar-system ephemerides.  Future ephemerides will
*     imply slightly changed transformations between TCG and TCB, which
*     could introduce a linear drift between TDB and TT;  however, any
*     such drift is unlikely to exceed 1 nanosecond per century.
*
*  6) The geocentric TDB-TT model used in the present routine is that of
*     Fairhead & Bretagnon (1990), in its full form.  It was originally
*     supplied by Fairhead (private communications with P.T.Wallace,
*     1990) as a Fortran subroutine.  The present routine contains an
*     adaptation of the Fairhead code.  The numerical results are
*     essentially unaffected by the changes, the differences with
*     respect to the Fairhead & Bretagnon original being at the 1D-20 s
*     level.
*
*     The topocentric part of the model is from Moyer (1981) and
*     Murray (1983), with fundamental arguments adapted from
*     Simon et al. 1994.  It is an approximation to the expression
*     ( v / c ) . ( r / c ), where v is the barycentric velocity of
*     the Earth, r is the geocentric position of the observer and
*     c is the speed of light.
*
*     By supplying zeroes for U and V, the topocentric part of the
*     model can be nullified, and the routine will return the Fairhead
*     & Bretagnon result alone.
*
*  7) During the interval 1950-2050, the absolute accuracy is better
*     than +/- 3 nanoseconds relative to time ephemerides obtained by
*     direct numerical integrations based on the JPL DE405 solar system
*     ephemeris.
*
*  8) It must be stressed that the present routine is merely a model,
*     and that numerical integration of solar-system ephemerides is the
*     definitive method for predicting the relationship between TCG and
*     TCB and hence between TT and TDB.
*
*  References:
*
*     Fairhead, L., & Bretagnon, P., Astron.Astrophys., 229, 240-247
```

```
*     (1990).
*
*     IAU 2006 Resolution 3.
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Moyer, T.D., Cel.Mech., 23, 33 (1981).
*
*     Murray, C.A., Vectorial Astrometry, Adam Hilger (1983).
*
*     Seidelmann, P.K. et al., Explanatory Supplement to the
*     Astronomical Almanac, Chapter 2, University Science Books (1992).
*
*     Simon, J.L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G. & Laskar, J., Astron.Astrophys., 282, 663-683 (1994).
*
*-
```

```
      SUBROUTINE iau_DTF2D ( SCALE, IY, IM, ID, IHR, IMN, SEC,
     :                       D1, D2, J )
*+
*  - - - - - - - - - - -
*   i a u _ D T F 2 D
*  - - - - - - - - - - -
*
*  Encode date and time fields into 2-part Julian Date (or in the case
*  of UTC a quasi-JD form that includes special provision for leap
*  seconds).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     SCALE      c*(*)  time scale ID (Note 1)
*     IY,IM,ID   i      year, month, day in Gregorian calendar (Note 2)
*     IHR,IMN    i      hour, minute
*     SEC        d      seconds
*
*  Returned:
*     D1,D2      d      2-part Julian Date (Notes 3,4)
*     J          i      status: +3 = both of next two
*                               +2 = time is after end of day (Note 5)
*                               +1 = dubious year (Note 6)
*                                0 = OK
*                               -1 = bad year
*                               -2 = bad month
*                               -3 = bad day
*                               -4 = bad hour
*                               -5 = bad minute
*                               -6 = bad second (<0)
*
*  Notes:
*
*  1) SCALE identifies the time scale.  Only the value 'UTC' (in upper
*     case) is significant, and enables handling of leap seconds (see
*     Note 4).
*
*  2) For calendar conventions and limitations, see iau_CAL2JD.
*
*  3) The sum of the results, D1+D2, is Julian Date, where normally D1
*     is the Julian Day Number and D2 is the fraction of a day.  In the
*     case of UTC, where the use of JD is problematical, special
*     conventions apply:  see the next note.
*
*  4) JD cannot unambiguously represent UTC during a leap second unless
*     special measures are taken.  The SOFA internal convention is that
*     the quasi-JD day represents UTC days whether the length is 86399,
*     86400 or 86401 SI seconds.
*
*  5) The warning status "time is after end of day" usually means that
*     the SEC argument is greater than 60D0.  However, in a day ending
*     in a leap second the limit changes to 61D0 (or 59D0 in the case of
*     a negative leap second).
*
*  6) The warning status "dubious year" flags UTCs that predate the
*     introduction of the time scale and that are too far in the future
*     to be trusted.  See iau_DAT for further details.
*
*  7) Only in the case of continuous and regular time scales (TAI, TT,
*     TCG, TCB and TDB) is the result D1+D2 a Julian Date, strictly
*     speaking.  In the other cases (UT1 and UTC) the result must be
*     used with circumspection;  in particular the difference between
*     two such results cannot be interpreted as a precise time
*     interval.
*
*  Called:
*     iau_CAL2JD   Gregorian calendar to JD
*     iau_DAT      delta(AT) = TAI-UTC
```

```
*      iau_JD2CAL   JD to Gregorian calendar
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_EE00 ( DATE1, DATE2, EPSA, DPSI )
*+
*  - - - - - - - - -
*   i a u _ E E 0 0
*  - - - - - - - - -
*
*  The equation of the equinoxes, compatible with IAU 2000 resolutions,
*  given the nutation in longitude and the mean obliquity.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*     EPSA           d      mean obliquity (Note 2)
*     DPSI           d      nutation in longitude (Note 3)
*
*  Returned:
*     iau_EE00       d      equation of the equinoxes (Note 4)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The obliquity, in radians, is mean of date.
*
*  3) The result, which is in radians, operates in the following sense:
*
*        Greenwich apparent ST = GMST + equation of the equinoxes
*
*  4) The result is compatible with the IAU 2000 resolutions.  For
*     further details, see IERS Conventions 2003 and Capitaine et al.
*     (2002).
*
*  Called:
*     iau_EECT00   equation of the equinoxes complementary terms
*
*  References:
*
*     Capitaine, N., Wallace, P.T. and McCarthy, D.D., "Expressions to
*     implement the IAU 2000 definition of UT1", Astronomy &
*     Astrophysics, 406, 1135-1149 (2003)
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_EE00A ( DATE1, DATE2 )
*+
*  - - - - - - - - - -
*   i a u _ E E 0 0 A
*  - - - - - - - - - -
*
*  Equation of the equinoxes, compatible with IAU 2000 resolutions.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_EE00A      d      equation of the equinoxes (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The result, which is in radians, operates in the following sense:
*
*        Greenwich apparent ST = GMST + equation of the equinoxes
*
*  3) The result is compatible with the IAU 2000 resolutions.  For
*     further details, see IERS Conventions 2003 and Capitaine et al.
*     (2002).
*
*  Called:
*     iau_PR00     IAU 2000 precession adjustments
*     iau_OBL80    mean obliquity, IAU 1980
*     iau_NUT00A   nutation, IAU 2000A
*     iau_EE00     equation of the equinoxes, IAU 2000
*
*  References:
*
*     Capitaine, N., Wallace, P.T. and McCarthy, D.D., "Expressions to
*     implement the IAU 2000 definition of UT1", Astronomy &
*     Astrophysics, 406, 1135-1149 (2003)
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_EE00B ( DATE1, DATE2 )
*+
*  - - - - - - - - - -
*   i a u _ E E 0 0 B
*  - - - - - - - - - -
*
*  Equation of the equinoxes, compatible with IAU 2000 resolutions but
*  using the truncated nutation model IAU 2000B.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d     TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_EE00B     d     equation of the equinoxes (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The result, which is in radians, operates in the following sense:
*
*        Greenwich apparent ST = GMST + equation of the equinoxes
*
*  3) The result is compatible with the IAU 2000 resolutions except that
*     accuracy has been compromised for the sake of speed.  For further
*     details, see McCarthy & Luzum (2001), IERS Conventions 2003 and
*     Capitaine et al. (2003).
*
*  Called:
*     iau_PR00     IAU 2000 precession adjustments
*     iau_OBL80    mean obliquity, IAU 1980
*     iau_NUT00B   nutation, IAU 2000B
*     iau_EE00     equation of the equinoxes, IAU 2000
*
*  References:
*
*     Capitaine, N., Wallace, P.T. and McCarthy, D.D., "Expressions to
*     implement the IAU 2000 definition of UT1", Astronomy &
*     Astrophysics, 406, 1135-1149 (2003)
*
*     McCarthy, D.D. & Luzum, B.J., "An abridged model of the
*     precession-nutation of the celestial pole", Celestial Mechanics &
*     Dynamical Astronomy, 85, 37-49 (2003)
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_EE06A ( DATE1, DATE2 )
*+
*  - - - - - - - - - -
*   i a u _ E E 0 6 A
*  - - - - - - - - - -
*
*  Equation of the equinoxes, compatible with IAU 2000 resolutions and
*  IAU 2006/2000A precession-nutation.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_EE06A      d      equation of the equinoxes (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The result, which is in radians, operates in the following sense:
*
*        Greenwich apparent ST = GMST + equation of the equinoxes
*
*  Called:
*     iau_ANPM     normalize angle into range +/- pi
*     iau_GST06A   Greenwich apparent sidereal time, IAU 2006/2000A
*     iau_GMST06   Greenwich mean sidereal time, IAU 2006
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), 2004, IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_EECT00 ( DATE1, DATE2 )
*+
*  - - - - - - - - - - -
*   i a u _ E E C T 0 0
*  - - - - - - - - - - -
*
*  Equation of the equinoxes complementary terms, consistent with
*  IAU 2000 resolutions.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2   d    TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_EECT00    d    complementary terms (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The "complementary terms" are part of the equation of the
*     equinoxes (EE), classically the difference between apparent and
*     mean Sidereal Time:
*
*        GAST = GMST + EE
*
*     with:
*
*        EE = dpsi * cos(eps)
*
*     where dpsi is the nutation in longitude and eps is the obliquity
*     of date.  However, if the rotation of the Earth were constant in
*     an inertial frame the classical formulation would lead to apparent
*     irregularities in the UT1 timescale traceable to side-effects of
*     precession-nutation.  In order to eliminate these effects from
*     UT1, "complementary terms" were introduced in 1994 (IAU, 1994) and
*     took effect from 1997 (Capitaine and Gontier, 1993):
*
*        GAST = GMST + CT + EE
*
*     By convention, the complementary terms are included as part of the
*     equation of the equinoxes rather than as part of the mean Sidereal
*     Time.  This slightly compromises the "geometrical" interpretation
*     of mean sidereal time but is otherwise inconsequential.
*
*     The present routine computes CT in the above expression,
*     compatible with IAU 2000 resolutions (Capitaine et al., 2002, and
*     IERS Conventions 2003).
*
*  Called:
*     iau_FAL03    mean anomaly of the Moon
```

```
*     iau_FALP03   mean anomaly of the Sun
*     iau_FAF03    mean argument of the latitude of the Moon
*     iau_FAD03    mean elongation of the Moon from the Sun
*     iau_FAOM03   mean longitude of the Moon's ascending node
*     iau_FAVE03   mean longitude of Venus
*     iau_FAE03    mean longitude of Earth
*     iau_FAPA03   general accumulated precession in longitude
*
*  References:
*
*     Capitaine, N. & Gontier, A.-M., Astron. Astrophys., 275,
*     645-650 (1993)
*
*     Capitaine, N., Wallace, P.T. and McCarthy, D.D., "Expressions to
*     implement the IAU 2000 definition of UT1", Astronomy &
*     Astrophysics, 406, 1135-1149 (2003)
*
*     IAU Resolution C7, Recommendation 3 (1994)
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_EFORM ( N, A, F, J )
*+
*  - - - - - - - - - -
*   i a u _ E F O R M
*  - - - - - - - - - -
*
*  Earth reference ellipsoids.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     N         i     ellipsoid identifier (Note 1)
*
*  Returned:
*     A         d     equatorial radius (meters, Note 2)
*     F         d     flattening (Note 2)
*     J         i     status:  0 = OK
*                             -1 = illegal identifier (Note 3)
*
*  Notes:
*
*  1) The identifier N is a number that specifies the choice of
*     reference ellipsoid.  The following are supported:
*
*        N   ellipsoid
*
*        1    WGS84
*        2    GRS80
*        3    WGS72
*
*     The number N has no significance outside the SOFA software.
*
*  2) The ellipsoid parameters are returned in the form of equatorial
*     radius in meters (A) and flattening (F).  The latter is a number
*     around 0.00335, i.e. around 1/298.
*
*  3) For the case where an unsupported N value is supplied, zero A and
*     F are returned, as well as error status.
*
*  References:
*
*     Department of Defense World Geodetic System 1984, National Imagery
*     and Mapping Agency Technical Report 8350.2, Third Edition, p3-2.
*
*     Moritz, H., Bull. Geodesique 66-2, 187 (1992).
*
*     The Department of Defense World Geodetic System 1972, World
*     Geodetic System Committee, May 1974.
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     p220.
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_EO06A ( DATE1, DATE2 )
*+
*  - - - - - - - - - -
*   i a u _ E O 0 6 A
*  - - - - - - - - - -
*
*  Equation of the origins, IAU 2006 precession and IAU 2000A nutation.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_EO06A      d      equation of the origins in radians
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The equation of the origins is the distance between the true
*     equinox and the celestial intermediate origin and, equivalently,
*     the difference between Earth rotation angle and Greenwich
*     apparent sidereal time (ERA-GST).  It comprises the precession
*     (since J2000.0) in right ascension plus the equation of the
*     equinoxes (including the small correction terms).
*
*  Called:
*     iau_PNM06A   classical NPB matrix, IAU 2006/2000A
*     iau_BPN2XY   extract CIP X,Y coordinates from NPB matrix
*     iau_S06      the CIO locator s, given X,Y, IAU 2006
*     iau_EORS     equation of the origins, given NPB matrix and s
*
*  References:
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_EORS ( RNPB, S )
*+
*  - - - - - - - - -
*   i a u _ E O R S
*  - - - - - - - - -
*
*  Equation of the origins, given the classical NPB matrix and the
*  quantity s.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     RNPB    d(3,3)    classical nutation x precession x bias matrix
*     S         d       the quantity s (the CIO locator)
*
*  Returned:
*     iau_EORS  d       the equation of the origins in radians.
*
*  Notes:
*
*  1)  The equation of the origins is the distance between the true
*      equinox and the celestial intermediate origin and, equivalently,
*      the difference between Earth rotation angle and Greenwich
*      apparent sidereal time (ERA-GST).  It comprises the precession
*      (since J2000.0) in right ascension plus the equation of the
*      equinoxes (including the small correction terms).
*
*  2)  The algorithm is from Wallace & Capitaine (2006).
*
*  References:
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855
*
*     Wallace, P. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_EPB ( DJ1, DJ2 )
*+
*  - - - - - - - -
*   i a u _ E P B
*  - - - - - - - -
*
*  Julian Date to Besselian Epoch.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DJ1,DJ2   d         Julian Date (see note)
*
*  The result is the Besselian Epoch.
*
*  Note:
*
*     The Julian Date is supplied in two pieces, in the usual SOFA
*     manner, which is designed to preserve time resolution.  The
*     Julian Date is available as a single number by adding DJ1 and
*     DJ2.  The maximum resolution is achieved if DJ1 is 2451545D0
*     (J2000.0).
*
*  Reference:
*
*     Lieske, J.H., 1979, Astron.Astrophys. 73, 282.
*
*-
```

```
      SUBROUTINE iau_EPB2JD ( EPB, DJM0, DJM )
*+
*  - - - - - - - - - - -
*   i a u _ E P B 2 J D
*  - - - - - - - - - - -
*
*  Besselian Epoch to Julian Date.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     EPB         d     Besselian Epoch (e.g. 1957.3D0)
*
*  Returned:
*     DJM0        d     MJD zero-point: always 2400000.5
*     DJM         d     Modified Julian Date
*
*  Note:
*
*     The Julian Date is returned in two pieces, in the usual SOFA
*     manner, which is designed to preserve time resolution.  The
*     Julian Date is available as a single number by adding DJM0 and
*     DJM.
*
*  Reference:
*
*     Lieske, J.H., 1979, Astron.Astrophys. 73, 282.
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_EPJ ( DJ1, DJ2 )
*+
*  - - - - - - - - 
*   i a u _ E P J
*  - - - - - - - -
*
*  Julian Date to Julian Epoch.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DJ1,DJ2   d         Julian Date (see note)
*
*  The result is the Julian Epoch.
*
*  Note:
*
*     The Julian Date is supplied in two pieces, in the usual SOFA
*     manner, which is designed to preserve time resolution.  The
*     Julian Date is available as a single number by adding DJ1 and
*     DJ2.  The maximum resolution is achieved if DJ1 is 2451545D0
*     (J2000.0).
*
*  Reference:
*
*     Lieske, J.H., 1979, Astron.Astrophys. 73, 282.
*
*-
```

```
      SUBROUTINE iau_EPJ2JD ( EPJ, DJM0, DJM )
*+
*  - - - - - - - - - - - -
*   i a u _ E P J 2 J D
*  - - - - - - - - - - - -
*
*  Julian Epoch to Julian Date.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     EPJ         d     Julian Epoch (e.g. 1996.8D0)
*
*  Returned:
*     DJM0        d     MJD zero-point: always 2400000.5
*     DJM         d     Modified Julian Date
*
*  Note:
*
*     The Julian Date is returned in two pieces, in the usual SOFA
*     manner, which is designed to preserve time resolution.  The
*     Julian Date is available as a single number by adding DJM0 and
*     DJM.
*
*  Reference:
*
*     Lieske, J.H., 1979, Astron.Astrophys. 73, 282.
*
*-
```

```
      SUBROUTINE iau_EPV00 ( DATE1, DATE2, PVH, PVB, JSTAT )
*+
*  - - - - - - - - - - -
*   i a u _ E P V 0 0
*  - - - - - - - - - - -
*
*  Earth position and velocity, heliocentric and barycentric, with
*  respect to the Barycentric Celestial Reference System.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1    d         TDB date part A (Note 1)
*     DATE2    d         TDB date part B (Note 1)
*
*  Returned:
*     PVH      d(3,2)    heliocentric Earth position/velocity (AU,AU/day)
*     PVB      d(3,2)    barycentric Earth position/velocity (AU,AU/day)
*     JSTAT    i         status: 0 = OK
*                               +1 = warning: date outside 1900-2100 AD
*
*  Notes:
*
*  1) The epoch EPOCH1+EPOCH2 is a Julian Date, apportioned in
*     any convenient way between the two arguments.  For example,
*     JD(TDB)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*             EPOCH1        EPOCH2
*
*          2450123.7D0        0D0        (JD method)
*           2451545D0      -1421.3D0     (J2000 method)
*          2400000.5D0     50123.2D0     (MJD method)
*          2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*     However, the accuracy of the result is more likely to be
*     limited by the algorithm itself than the way the epoch has been
*     expressed.
*
*  2) On return, the arrays PVH and PVB contain the following:
*
*        PVH(1,1)  x       }
*        PVH(2,1)  y       } heliocentric position, AU
*        PVH(3,1)  z       }
*
*        PVH(1,2)  xdot    }
*        PVH(2,2)  ydot    } heliocentric velocity, AU/d
*        PVH(3,2)  zdot    }
*
*        PVB(1,1)  x       }
*        PVB(2,1)  y       } barycentric position, AU
*        PVB(3,1)  z       }
*
*        PVB(1,2)  xdot    }
*        PVB(2,2)  ydot    } barycentric velocity, AU/d
*        PVB(3,2)  zdot    }
*
*     The vectors are with respect to the Barycentric Celestial
*     Reference System.  The time unit is one day in TDB.
*
*  3) The routine is a SIMPLIFIED SOLUTION from the planetary theory
*     VSOP2000 (X. Moisson, P. Bretagnon, 2001, Celes. Mechanics &
*     Dyn. Astron., 80, 3/4, 205-213) and is an adaptation of original
```

```
*     Fortran code supplied by P. Bretagnon (private comm., 2000).
*
*  4) Comparisons over the time span 1900-2100 with this simplified
*     solution and the JPL DE405 ephemeris give the following results:
*
*                                RMS    max
*           Heliocentric:
*              position error    3.7   11.2   km
*              velocity error    1.4    5.0   mm/s
*
*           Barycentric:
*              position error    4.6   13.4   km
*              velocity error    1.4    4.9   mm/s
*
*     Comparisons with the JPL DE406 ephemeris show that by 1800 and
*     2200 the position errors are approximately double their 1900-2100
*     size.  By 1500 and 2500 the deterioration is a factor of 10 and by
*     1000 and 3000 a factor of 60.  The velocity accuracy falls off at
*     about half that rate.
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_EQEQ94 ( DATE1, DATE2 )
*+
*  - - - - - - - - - - -
*   i a u _ E Q E Q 9 4
*  - - - - - - - - - - -
*
*  Equation of the equinoxes, IAU 1994 model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2     d         TDB date (Note 1)
*
*  Returned:
*     iau_EQEQ94      d         equation of the equinoxes (Note 2)
*
*  Notes:
*
*  1) The date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TDB)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The result, which is in radians, operates in the following sense:
*
*        Greenwich apparent ST = GMST + equation of the equinoxes
*
*  Called:
*     iau_NUT80    nutation, IAU 1980
*     iau_OBL80    mean obliquity, IAU 1980
*     iau_ANPM     normalize angle into range +/- pi
*
*  References:
*
*     IAU Resolution C7, Recommendation 3 (1994)
*
*     Capitaine, N. & Gontier, A.-M., Astron. Astrophys., 275,
*     645-650 (1993)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_ERA00 ( DJ1, DJ2 )
*+
*  - - - - - - - - - -
*   i a u _ E R A 0 0
*  - - - - - - - - - -
*
*  Earth rotation angle (IAU 2000 model).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DJ1,DJ2     d      UT1 as a 2-part Julian Date (see note)
*
*  The result is the Earth rotation angle (radians), in the range 0 to
*  2pi.
*
*  Notes:
*
*  1) The UT1 date DJ1+DJ2 is a Julian Date, apportioned in any
*     convenient way between the arguments DJ1 and DJ2.  For example,
*     JD(UT1)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*             DJ1            DJ2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  The date & time method is
*     best matched to the algorithm used:  maximum accuracy (or, at
*     least, minimum noise) is delivered when the DJ1 argument is for
*     0hrs UT1 on the day in question and the DJ2 argument lies in the
*     range 0 to 1, or vice versa.
*
*  2) The algorithm is adapted from Expression 22 of Capitaine et al.
*     2000.  The time argument has been expressed in days directly,
*     and, to retain precision, integer contributions have been
*     eliminated.  The same formulation is given in IERS Conventions
*     (2003), Chap. 5, Eq. 14.
*
*  Called:
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  References:
*
*     Capitaine N., Guinot B. and McCarthy D.D, 2000, Astron.
*     Astrophys., 355, 398-405.
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAD03 ( T )
*+
*  - - - - - - - - - -
*   i a u _ F A D 0 3
*  - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean elongation of the Moon from the Sun.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAD03   d    D, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     is from Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAE03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A E 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean longitude of Earth.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAE03   d    mean longitude of Earth, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     comes from Souchay et al. (1999) after Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*     Souchay, J., Loysel, B., Kinoshita, H., Folgueira, M. 1999,
*     Astron.Astrophys.Supp.Ser. 135, 111
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAF03 ( T )
*+
*  - - - - - - - - - -
*   i a u _ F A F 0 3
*  - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean longitude of the Moon minus mean longitude of the ascending
*  node.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAF03   d    F, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     is from Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAJU03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A J U 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean longitude of Jupiter.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAJU03  d    mean longitude of Jupiter, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     comes from Souchay et al. (1999) after Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*     Souchay, J., Loysel, B., Kinoshita, H., Folgueira, M. 1999,
*     Astron.Astrophys.Supp.Ser. 135, 111
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAL03 ( T )
*+
*  - - - - - - - - - - 
*   i a u _ F A L 0 3
*  - - - - - - - - - - 
*
*  Fundamental argument, IERS Conventions (2003):
*  mean anomaly of the Moon.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAL03   d    l, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     is from Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FALP03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A L P 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean anomaly of the Sun.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FALP03  d    l', radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     is from Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAMA03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A M A 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean longitude of Mars.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAMA03  d    mean longitude of Mars, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     comes from Souchay et al. (1999) after Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*     Souchay, J., Loysel, B., Kinoshita, H., Folgueira, M. 1999,
*     Astron.Astrophys.Supp.Ser. 135, 111
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAME03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A M E 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean longitude of Mercury.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAME03  d    mean longitude of Mercury, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     comes from Souchay et al. (1999) after Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*     Souchay, J., Loysel, B., Kinoshita, H., Folgueira, M. 1999,
*     Astron.Astrophys.Supp.Ser. 135, 111
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FANE03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A N E 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean longitude of Neptune.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FANE03  d    mean longitude of Neptune, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     is adapted from Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAOM03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A O M 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean longitude of the Moon's ascending node.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAOM03  d    Omega, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     is from Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAPA03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A P A 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  general accumulated precession in longitude.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAPA03  d    general precession in longitude, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003).  It
*     is taken from Kinoshita & Souchay (1990) and comes originally from
*     Lieske et al. (1977).
*
*  References:
*
*     Kinoshita, H. and Souchay J. 1990, Celest.Mech. and Dyn.Astron.
*     48, 187
*
*     Lieske, J.H., Lederle, T., Fricke, W. & Morando, B. 1977,
*     Astron.Astrophys. 58, 1-16
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FASA03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A S A 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean longitude of Saturn.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FASA03  d    mean longitude of Saturn, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     comes from Souchay et al. (1999) after Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*     Souchay, J., Loysel, B., Kinoshita, H., Folgueira, M. 1999,
*     Astron.Astrophys.Supp.Ser. 135, 111
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAUR03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A U R 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean longitude of Uranus.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAUR03  d    mean longitude of Uranus, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     is adapted from Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_FAVE03 ( T )
*+
*  - - - - - - - - - - -
*   i a u _ F A V E 0 3
*  - - - - - - - - - - -
*
*  Fundamental argument, IERS Conventions (2003):
*  mean longitude of Venus.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     T           d    TDB, Julian centuries since J2000.0 (Note 1)
*
*  Returned:
*     iau_FAVE03  d    mean longitude of Venus, radians (Note 2)
*
*  Notes:
*
*  1) Though T is strictly TDB, it is usually more convenient to use TT,
*     which makes no significant difference.
*
*  2) The expression used is as adopted in IERS Conventions (2003) and
*     comes from Souchay et al. (1999) after Simon et al. (1994).
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
*
*     Souchay, J., Loysel, B., Kinoshita, H., Folgueira, M. 1999,
*     Astron.Astrophys.Supp.Ser. 135, 111
*
*-
```

```
      SUBROUTINE iau_FK52H ( R5, D5, DR5, DD5, PX5, RV5,
     :                       RH, DH, DRH, DDH, PXH, RVH )
*+
*  - - - - - - - - - -
*   i a u _ F K 5 2 H
*  - - - - - - - - - -
*
*  Transform FK5 (J2000.0) star data into the Hipparcos system.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given (all FK5, equinox J2000.0, epoch J2000.0):
*     R5        d      RA (radians)
*     D5        d      Dec (radians)
*     DR5       d      proper motion in RA (dRA/dt, rad/Jyear)
*     DD5       d      proper motion in Dec (dDec/dt, rad/Jyear)
*     PX5       d      parallax (arcsec)
*     RV5       d      radial velocity (km/s, positive = receding)
*
*  Returned (all Hipparcos, epoch J2000.0):
*     RH        d      RA (radians)
*     DH        d      Dec (radians)
*     DRH       d      proper motion in RA (dRA/dt, rad/Jyear)
*     DDH       d      proper motion in Dec (dDec/dt, rad/Jyear)
*     PXH       d      parallax (arcsec)
*     RVH       d      radial velocity (km/s, positive = receding)
*
*  Notes:
*
*  1) This routine transforms FK5 star positions and proper motions into
*     the system of the Hipparcos catalog.
*
*  2) The proper motions in RA are dRA/dt rather than cos(Dec)*dRA/dt,
*     and are per year rather than per century.
*
*  3) The FK5 to Hipparcos transformation is modeled as a pure rotation
*     and spin;  zonal errors in the FK5 catalog are not taken into
*     account.
*
*  4) See also iau_H2FK5, iau_FK5HZ, iau_HFK5Z.
*
*  Called:
*     iau_STARPV   star catalog data to space motion pv-vector
*     iau_FK5HIP   FK5 to Hipparcos rotation and spin
*     iau_RXP      product of r-matrix and p-vector
*     iau_PXP      vector product of two p-vectors
*     iau_PPP      p-vector plus p-vector
*     iau_PVSTAR   space motion pv-vector to star catalog data
*
*  Reference:
*
*     F.Mignard & M.Froeschle, Astron. Astrophys. 354, 732-739 (2000).
*
*-
```

```
      SUBROUTINE iau_FK5HIP ( R5H, S5H )
*+
*  - - - - - - - - - - - -
*   i a u _ F K 5 H I P
*  - - - - - - - - - - - -
*
*  FK5 to Hipparcos rotation and spin.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Returned:
*     R5H     d(3,3)   r-matrix: FK5 rotation wrt Hipparcos (Note 2)
*     S5H     d(3)     r-vector: FK5 spin wrt Hipparcos (Note 3)
*
*  Notes:
*
*  1) This routine models the FK5 to Hipparcos transformation as a
*     pure rotation and spin;  zonal errors in the FK5 catalogue are
*     not taken into account.
*
*  2) The r-matrix R5H operates in the sense:
*
*           P_Hipparcos = R5H x P_FK5
*
*     where P_FK5 is a p-vector in the FK5 frame, and P_Hipparcos is
*     the equivalent Hipparcos p-vector.
*
*  3) The r-vector S5H represents the time derivative of the FK5 to
*     Hipparcos rotation.  The units are radians per year (Julian,
*     TDB).
*
*  Called:
*     iau_RV2M     r-vector to r-matrix
*
*  Reference:
*
*     F.Mignard & M.Froeschle, Astron. Astrophys. 354, 732-739 (2000).
*
*-
```

```
      SUBROUTINE iau_FK5HZ ( R5, D5, DATE1, DATE2, RH, DH )
*+
*  - - - - - - - - - -
*   i a u _ F K 5 H Z
*  - - - - - - - - - -
*
*  Transform an FK5 (J2000.0) star position into the system of the
*  Hipparcos catalogue, assuming zero Hipparcos proper motion.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     R5              d      FK5 RA (radians), equinox J2000.0, at date
*     D5              d      FK5 Dec (radians), equinox J2000.0, at date
*     DATE1,DATE2     d      TDB date (Notes 1,2)
*
*  Returned:
*     RH              d      Hipparcos RA (radians)
*     DH              d      Hipparcos Dec (radians)
*
*  Notes:
*
*  1) This routine converts a star position from the FK5 system to
*     the Hipparcos system, in such a way that the Hipparcos proper
*     motion is zero.  Because such a star has, in general, a non-zero
*     proper motion in the FK5 system, the routine requires the date
*     at which the position in the FK5 system was determined.
*
*  2) The date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TDB)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  3) The FK5 to Hipparcos transformation is modeled as a pure
*     rotation and spin;  zonal errors in the FK5 catalogue are
*     not taken into account.
*
*  4) The position returned by this routine is in the Hipparcos
*     reference system but at date DATE1+DATE2.
*
*  5) See also iau_FK52H, iau_H2FK5, iau_HFK5Z.
*
*  Called:
*     iau_S2C      spherical coordinates to unit vector
*     iau_FK5HIP   FK5 to Hipparcos rotation and spin
*     iau_SXP      multiply p-vector by scalar
*     iau_RV2M     r-vector to r-matrix
*     iau_TRXP     product of transpose of r-matrix and p-vector
*     iau_PXP      vector product of two p-vectors
*     iau_C2S      p-vector to spherical
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  Reference:
*
*     F.Mignard & M.Froeschle, Astron. Astrophys. 354, 732-739 (2000).
```

*
*_

```
      SUBROUTINE iau_FW2M ( GAMB, PHIB, PSI, EPS, R )
*+
*  - - - - - - - - -
*   i a u _ F W 2 M
*  - - - - - - - - -
*
*  Form rotation matrix given the Fukushima-Williams angles.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     GAMB       d      F-W angle gamma_bar (radians)
*     PHIB       d      F-W angle phi_bar (radians)
*     PSI        d      F-W angle psi (radians)
*     EPS        d      F-W angle epsilon (radians)
*
*  Returned:
*     R        d(3,3)   rotation matrix
*
*  Notes:
*
*  1) Naming the following points:
*
*           e = J2000.0 ecliptic pole,
*           p = GCRS pole,
*           E = ecliptic pole of date,
*     and   P = CIP,
*
*     the four Fukushima-Williams angles are as follows:
*
*        GAMB = gamma = epE
*        PHIB = phi = pE
*        PSI = psi = pEP
*        EPS = epsilon = EP
*
*  2) The matrix representing the combined effects of frame bias,
*     precession and nutation is:
*
*        NxPxB = R_1(-EPS).R_3(-PSI).R_1(PHIB).R_3(GAMB)
*
*  3) Three different matrices can be constructed, depending on the
*     supplied angles:
*
*     o  To obtain the nutation x precession x frame bias matrix,
*        generate the four precession angles, generate the nutation
*        components and add them to the psi_bar and epsilon_A angles,
*        and call the present routine.
*
*     o  To obtain the precession x frame bias matrix, generate the
*        four precession angles and call the present routine.
*
*     o  To obtain the frame bias matrix, generate the four precession
*        angles for date J2000.0 and call the present routine.
*
*     The nutation-only and precession-only matrices can if necessary
*     be obtained by combining these three appropriately.
*
*  Called:
*     iau_IR       initialize r-matrix to identity
*     iau_RZ       rotate around Z-axis
*     iau_RX       rotate around X-axis
*
*  Reference:
*
*     Hilton, J. et al., 2006, Celest.Mech.Dyn.Astron. 94, 351
*
*-
```

```
      SUBROUTINE iau_FW2XY ( GAMB, PHIB, PSI, EPS, X, Y )
*+
*  - - - - - - - - - -
*   i a u _ F W 2 X Y
*  - - - - - - - - - -
*
*  CIP X,Y given Fukushima-Williams bias-precession-nutation angles.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     GAMB       d      F-W angle gamma_bar (radians)
*     PHIB       d      F-W angle phi_bar (radians)
*     PSI        d      F-W angle psi (radians)
*     EPS        d      F-W angle epsilon (radians)
*
*  Returned:
*     X,Y        d      CIP X,Y ("radians")
*
*  Notes:
*
*  1) Naming the following points:
*
*           e = J2000.0 ecliptic pole,
*           p = GCRS pole
*           E = ecliptic pole of date,
*     and   P = CIP,
*
*     the four Fukushima-Williams angles are as follows:
*
*        GAMB = gamma = epE
*        PHIB = phi = pE
*        PSI = psi = pEP
*        EPS = epsilon = EP
*
*  2) The matrix representing the combined effects of frame bias,
*     precession and nutation is:
*
*        NxPxB = R_1(-EPSA).R_3(-PSI).R_1(PHIB).R_3(GAMB)
*
*     X,Y are elements (3,1) and (3,2) of the matrix.
*
*  Called:
*     iau_FW2M     F-W angles to r-matrix
*     iau_BPN2XY   extract CIP X,Y coordinates from NPB matrix
*
*  Reference:
*
*     Hilton, J. et al., 2006, Celest.Mech.Dyn.Astron. 94, 351
*
*-
```

```
      SUBROUTINE iau_GC2GD ( N, XYZ, ELONG, PHI, HEIGHT, J )
*+
*  - - - - - - - - - -
*   i a u _ G C 2 G D
*  - - - - - - - - - -
*
*  Transform geocentric coordinates to geodetic using the specified
*  reference ellipsoid.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical transformation.
*
*  Given:
*     N          i     ellipsoid identifier (Note 1)
*     XYZ        d(3)  geocentric vector (Note 2)
*
*  Returned:
*     ELONG      d     longitude (radians, east +ve)
*     PHI        d     latitude (geodetic, radians, Note 3)
*     HEIGHT     d     height above ellipsoid (geodetic, Notes 2,3)
*     J          i     status:  0 = OK
*                              -1 = illegal identifier (Note 3)
*                              -2 = internal error (Note 3)
*
*  Notes:
*
*  1) The identifier N is a number that specifies the choice of
*     reference ellipsoid.  The following are supported:
*
*        N   ellipsoid
*
*        1    WGS84
*        2    GRS80
*        3    WGS72
*
*     The number N has no significance outside the SOFA software.
*
*  2) The geocentric vector (XYZ, given) and height (HEIGHT, returned)
*     are in meters.
*
*  3) An error status J=-1 means that the identifier N is illegal.  An
*     error status J=-2 is theoretically impossible.  In all error
*     cases, PHI and HEIGHT are both set to -1D9.
*
*  4) The inverse transformation is performed in the routine iau_GD2GC.
*
*  Called:
*     iau_EFORM    Earth reference ellipsoids
*     iau_GC2GDE   geocentric to geodetic transformation, general
*
*-
```

```
      SUBROUTINE iau_GC2GDE ( A, F, XYZ, ELONG, PHI, HEIGHT, J )
*+
*  - - - - - - - - - - -
*   i a u _ G C 2 G D E
*  - - - - - - - - - - -
*
*  Transform geocentric coordinates to geodetic for a reference
*  ellipsoid of specified form.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     A         d      equatorial radius (Notes 2,4)
*     F         d      flattening (Note 3)
*     XYZ       d(3)   geocentric vector (Note 4)
*
*  Returned:
*     ELONG     d      longitude (radians, east +ve)
*     PHI       d      latitude (geodetic, radians)
*     HEIGHT    d      height above ellipsoid (geodetic, Note 4)
*     J         i      status:  0 = OK
*                              -1 = illegal F
*                              -2 = illegal A
*
*  Notes:
*
*  1) This routine is closely based on the GCONV2H subroutine by
*     Toshio Fukushima (see reference).
*
*  2) The equatorial radius, A, can be in any units, but meters is
*     the conventional choice.
*
*  3) The flattening, F, is (for the Earth) a value around 0.00335,
*     i.e. around 1/298.
*
*  4) The equatorial radius, A, and the geocentric vector, XYZ,
*     must be given in the same units, and determine the units of
*     the returned height, HEIGHT.
*
*  5) If an error occurs (J<0), ELONG, PHI and HEIGHT are unchanged.
*
*  6) The inverse transformation is performed in the routine iau_GD2GCE.
*
*  7) The transformation for a standard ellipsoid (such as WGS84) can
*     more conveniently be performed by calling iau_GC2GD, which uses a
*     numerical code (1 for WGS84) to identify the required A and F
*     values.
*
*  Reference:
*
*     Fukushima, T., "Transformation from Cartesian to geodetic
*     coordinates accelerated by Halley's method", J.Geodesy (2006)
*     79: 689-693
*
*-
```

```
      SUBROUTINE iau_GD2GC ( N, ELONG, PHI, HEIGHT, XYZ, J )
*+
*  - - - - - - - - - -
*   i a u _ G D 2 G C
*  - - - - - - - - - -
*
*  Transform geodetic coordinates to geocentric using the specified
*  reference ellipsoid.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical transformation.
*
*  Given:
*     N          i     ellipsoid identifier (Note 1)
*     ELONG      d     longitude (radians, east +ve)
*     PHI        d     latitude (geodetic, radians, Note 3)
*     HEIGHT     d     height above ellipsoid (geodetic, Notes 2,3)
*
*  Returned:
*     XYZ        d(3)  geocentric vector (Note 2)
*     J          i     status:  0 = OK
*                              -1 = illegal identifier (Note 3)
*                              -2 = illegal case (Note 3)
*
*  Notes:
*
*  1) The identifier N is a number that specifies the choice of
*     reference ellipsoid.  The following are supported:
*
*        N   ellipsoid
*
*        1    WGS84
*        2    GRS80
*        3    WGS72
*
*     The number N has no significance outside the SOFA software.
*
*  2) The height (HEIGHT, given) and the geocentric vector (XYZ,
*     returned) are in meters.
*
*  3) No validation is performed on the arguments ELONG, PHI and HEIGHT.
*     An error status J=-1 means that the identifier N is illegal.  An
*     error status J=-2 protects against cases that would lead to
*     arithmetic exceptions.  In all error cases, XYZ is set to zeros.
*
*  4) The inverse transformation is performed in the routine iau_GC2GD.
*
*  Called:
*     iau_EFORM    Earth reference ellipsoids
*     iau_GD2GCE   geodetic to geocentric transformation, general
*     iau_ZP       zero p-vector
*
*-
```

```
      SUBROUTINE iau_GD2GCE ( A, F, ELONG, PHI, HEIGHT, XYZ, J )
*+
*  - - - - - - - - - - -
*   i a u _ G D 2 G C E
*  - - - - - - - - - - -
*
*  Transform geodetic coordinates to geocentric for a reference
*  ellipsoid of specified form.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     A         d     equatorial radius (Notes 1,4)
*     F         d     flattening (Notes 2,4)
*     ELONG     d     longitude (radians, east +ve)
*     PHI       d     latitude (geodetic, radians, Note 4)
*     HEIGHT    d     height above ellipsoid (geodetic, Notes 3,4)
*
*  Returned:
*     XYZ       d(3)  geocentric vector (Note 3)
*     J         i     status:  0 = OK
*                             -1 = illegal case (Note 4)
*
*  Notes:
*
*  1) The equatorial radius, A, can be in any units, but meters is
*     the conventional choice.
*
*  2) The flattening, F, is (for the Earth) a value around 0.00335,
*     i.e. around 1/298.
*
*  3) The equatorial radius, A, and the height, HEIGHT, must be
*     given in the same units, and determine the units of the
*     returned geocentric vector, XYZ.
*
*  4) No validation is performed on individual arguments.  The error
*     status J=-1 protects against (unrealistic) cases that would lead
*     to arithmetic exceptions.  If an error occurs, XYZ is unchanged.
*
*  5) The inverse transformation is performed in the routine iau_GC2GDE.
*
*  6) The transformation for a standard ellipsoid (such as WGS84) can
*     more conveniently be performed by calling iau_GD2GC, which uses a
*     numerical code (1 for WGS84) to identify the required A and F
*     values.
*
*  References:
*
*     Green, R.M., Spherical Astronomy, Cambridge University Press,
*     (1985) Section 4.5, p96.
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     Section 4.22, p202.
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_GMST00 ( UTA, UTB, TTA, TTB )
*+
*  - - - - - - - - - - -
*   i a u _ G M S T 0 0
*  - - - - - - - - - - -
*
*  Greenwich Mean Sidereal Time (model consistent with IAU 2000
*  resolutions).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     UTA, UTB     d      UT1 as a 2-part Julian Date (Notes 1,2)
*     TTA, TTB     d      TT as a 2-part Julian Date (Notes 1,2)
*
*  Returned:
*     iau_GMST00   d      Greenwich mean sidereal time (radians)
*
*  Notes:
*
*  1) The UT1 and TT dates UTA+UTB and TTA+TTB respectively, are both
*     Julian Dates, apportioned in any convenient way between the
*     argument pairs.  For example, JD=2450123.7 could be expressed in
*     any of these ways, among others:
*
*            Part A         Part B
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable (in the case of UT;  the TT is not at all critical
*     in this respect).  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  For UT, the date & time
*     method is best matched to the algorithm that is used by the Earth
*     Rotation Angle routine, called internally:  maximum accuracy (or,
*     at least, minimum noise) is delivered when the UTA argument is for
*     0hrs UT1 on the day in question and the UTB argument lies in the
*     range 0 to 1, or vice versa.
*
*  2) Both UT1 and TT are required, UT1 to predict the Earth rotation
*     and TT to predict the effects of precession.  If UT1 is used for
*     both purposes, errors of order 100 microarcseconds result.
*
*  3) This GMST is compatible with the IAU 2000 resolutions and must be
*     used only in conjunction with other IAU 2000 compatible components
*     such as precession-nutation and equation of the equinoxes.
*
*  4) The result is returned in the range 0 to 2pi.
*
*  5) The algorithm is from Capitaine et al. (2003) and IERS Conventions
*     2003.
*
*  Called:
*     iau_ERA00    Earth rotation angle, IAU 2000
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  References:
*
*     Capitaine, N., Wallace, P.T. and McCarthy, D.D., "Expressions to
*     implement the IAU 2000 definition of UT1", Astronomy &
*     Astrophysics, 406, 1135-1149 (2003)
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
```

*_

```
      DOUBLE PRECISION FUNCTION iau_GMST06 ( UTA, UTB, TTA, TTB )
*+
*  - - - - - - - - - - - -
*   i a u _ G M S T 0 6
*  - - - - - - - - - - - -
*
*  Greenwich mean sidereal time (consistent with IAU 2006 precession).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     UTA, UTB     d      UT1 as a 2-part Julian Date (Notes 1,2)
*     TTA, TTB     d      TT as a 2-part Julian Date (Notes 1,2)
*
*  Returned:
*     iau_GMST06   d      Greenwich mean sidereal time (radians)
*
*  Notes:
*
*  1) The UT1 and TT dates UTA+UTB and TTA+TTB respectively, are both
*     Julian Dates, apportioned in any convenient way between the
*     argument pairs.  For example, JD=2450123.7 could be expressed in
*     any of these ways, among others:
*
*            Part A         Part B
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable (in the case of UT;  the TT is not at all critical
*     in this respect).  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  For UT, the date & time
*     method is best matched to the algorithm that is used by the Earth
*     rotation angle routine, called internally:  maximum accuracy (or,
*     at least, minimum noise) is delivered when the UTA argument is for
*     0hrs UT1 on the day in question and the UTB argument lies in the
*     range 0 to 1, or vice versa.
*
*  2) Both UT1 and TT are required, UT1 to predict the Earth rotation
*     and TT to predict the effects of precession.  If UT1 is used for
*     both purposes, errors of order 100 microarcseconds result.
*
*  3) This GMST is compatible with the IAU 2006 precession and must not
*     be used with other precession models.
*
*  4) The result is returned in the range 0 to 2pi.
*
*  Called:
*     iau_ERA00    Earth rotation angle, IAU 2000
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  Reference:
*
*     Capitaine, N., Wallace, P.T. & Chapront, J., 2005,
*     Astron.Astrophys. 432, 355
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_GMST82 ( DJ1, DJ2 )
*+
*  - - - - - - - - - - -
*   i a u _ G M S T 8 2
*  - - - - - - - - - - -
*
*  Universal Time to Greenwich Mean Sidereal Time (IAU 1982 model).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DJ1, DJ2     d      UT1 Julian Date (see note)
*
*  Returned:
*     iau_GMST82   d      Greenwich mean sidereal time (radians)
*
*  Notes:
*
*  1) The UT1 epoch DJ1+DJ2 is a Julian Date, apportioned in any
*     convenient way between the arguments DJ1 and DJ2.  For example,
*     JD(UT1)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*             DJ1            DJ2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  The date & time method is
*     best matched to the algorithm used:  maximum accuracy (or, at
*     least, minimum noise) is delivered when the DJ1 argument is for
*     0hrs UT1 on the day in question and the DJ2 argument lies in the
*     range 0 to 1, or vice versa.
*
*  2) The algorithm is based on the IAU 1982 expression.  This is always
*     described as giving the GMST at 0 hours UT1.  In fact, it gives the
*     difference between the GMST and the UT, the steady 4-minutes-per-day
*     drawing-ahead of ST with respect to UT.  When whole days are ignored,
*     the expression happens to equal the GMST at 0 hours UT1 each day.
*
*  3) In this routine, the entire UT1 (the sum of the two arguments DJ1
*     and DJ2) is used directly as the argument for the standard formula,
*     the constant term of which is adjusted by 12 hours to take account
*     of the noon phasing of Julian Date.  The UT1 is then added, but
*     omitting whole days to conserve accuracy.
*
*  4) The result is returned in the range 0 to 2pi.
*
*  Called:
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  References:
*
*     Transactions of the International Astronomical Union,
*     XVIII B, 67 (1983).
*
*     Aoki et al., Astron. Astrophys. 105, 359-361 (1982).
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_GST00A ( UTA, UTB, TTA, TTB )
*+
*  - - - - - - - - - - -
*   i a u _ G S T 0 0 A
*  - - - - - - - - - - -
*
*  Greenwich Apparent Sidereal Time (consistent with IAU 2000
*  resolutions).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     UTA, UTB     d      UT1 as a 2-part Julian Date (Notes 1,2)
*     TTA, TTB     d      TT as a 2-part Julian Date (Notes 1,2)
*
*  Returned:
*     iau_GST00A   d      Greenwich apparent sidereal time (radians)
*
*  Notes:
*
*  1) The UT1 and TT dates UTA+UTB and TTA+TTB respectively, are both
*     Julian Dates, apportioned in any convenient way between the
*     argument pairs.  For example, JD=2450123.7 could be expressed in
*     any of these ways, among others:
*
*            Part A         Part B
*
*        2450123.7D0        0D0        (JD method)
*         2451545D0      -1421.3D0     (J2000 method)
*        2400000.5D0     50123.2D0     (MJD method)
*        2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable (in the case of UT;  the TT is not at all critical
*     in this respect).  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  For UT, the date & time
*     method is best matched to the algorithm that is used by the Earth
*     Rotation Angle routine, called internally:  maximum accuracy (or,
*     at least, minimum noise) is delivered when the UTA argument is for
*     0hrs UT1 on the day in question and the UTB argument lies in the
*     range 0 to 1, or vice versa.
*
*  2) Both UT1 and TT are required, UT1 to predict the Earth rotation
*     and TT to predict the effects of precession-nutation.  If UT1 is
*     used for both purposes, errors of order 100 microarcseconds
*     result.
*
*  3) This GAST is compatible with the IAU 2000 resolutions and must be
*     used only in conjunction with other IAU 2000 compatible components
*     such as precession-nutation.
*
*  4) The result is returned in the range 0 to 2pi.
*
*  5) The algorithm is from Capitaine et al. (2003) and IERS Conventions
*     2003.
*
*  Called:
*     iau_GMST00   Greenwich mean sidereal time, IAU 2000
*     iau_EE00A    equation of the equinoxes, IAU 2000A
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  References:
*
*     Capitaine, N., Wallace, P.T. and McCarthy, D.D., "Expressions to
*     implement the IAU 2000 definition of UT1", Astronomy &
*     Astrophysics, 406, 1135-1149 (2003)
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
```

```
*      IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_GST00B ( UTA, UTB )
*+
*  - - - - - - - - - - -
*   i a u _ G S T 0 0 B
*  - - - - - - - - - - -
*
*  Greenwich Apparent Sidereal Time (consistent with IAU 2000
*  resolutions but using the truncated nutation model IAU 2000B).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     UTA, UTB     d      UT1 as a 2-part Julian Date (Notes 1,2)
*
*  Returned:
*     iau_GST00B   d      Greenwich apparent sidereal time (radians)
*
*  Notes:
*
*  1) The UT1 date UTA+UTB is a Julian Date, apportioned in any
*     convenient way between the argument pair.  For example,
*     JD=2450123.7 could be expressed in any of these ways, among
*     others:
*
*             UTA            UTB
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in cases
*     where the loss of several decimal digits of resolution is
*     acceptable.  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  For UT, the date & time
*     method is best matched to the algorithm that is used by the Earth
*     Rotation Angle routine, called internally:  maximum accuracy (or,
*     at least, minimum noise) is delivered when the UTA argument is for
*     0hrs UT1 on the day in question and the UTB argument lies in the
*     range 0 to 1, or vice versa.
*
*  2) The result is compatible with the IAU 2000 resolutions, except
*     that accuracy has been compromised for the sake of speed and
*     convenience in two respects:
*
*     . UT is used instead of TDB (or TT) to compute the precession
*       component of GMST and the equation of the equinoxes.  This
*       results in errors of order 0.1 mas at present.
*
*     . The IAU 2000B abridged nutation model (McCarthy & Luzum, 2001)
*       is used, introducing errors of up to 1 mas.
*
*  3) This GAST is compatible with the IAU 2000 resolutions and must be
*     used only in conjunction with other IAU 2000 compatible components
*     such as precession-nutation.
*
*  4) The result is returned in the range 0 to 2pi.
*
*  5) The algorithm is from Capitaine et al. (2003) and IERS Conventions
*     2003.
*
*  Called:
*     iau_GMST00   Greenwich mean sidereal time, IAU 2000
*     iau_EE00B    equation of the equinoxes, IAU 2000B
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  References:
*
*     Capitaine, N., Wallace, P.T. and McCarthy, D.D., "Expressions to
```

```
*     implement the IAU 2000 definition of UT1", Astronomy &
*     Astrophysics, 406, 1135-1149 (2003)
*
*     McCarthy, D.D. & Luzum, B.J., "An abridged model of the
*     precession-nutation of the celestial pole", Celestial Mechanics &
*     Dynamical Astronomy, 85, 37-49 (2003)
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_GST06 ( UTA, UTB, TTA, TTB, RNPB )
*+
*  - - - - - - - - - -
*   i a u _ G S T 0 6
*  - - - - - - - - - -
*
*  Greenwich apparent sidereal time, IAU 2006, given the NPB matrix.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     UTA, UTB     d      UT1 as a 2-part Julian Date (Notes 1,2)
*     TTA, TTB     d      TT as a 2-part Julian Date (Notes 1,2)
*     RNPB       d(3,3)   nutation x precession x bias matrix
*
*  Returned:
*     iau_GST06    d      Greenwich apparent sidereal time (radians)
*
*  Notes:
*
*  1) The UT1 and TT dates UTA+UTB and TTA+TTB respectively, are both
*     Julian Dates, apportioned in any convenient way between the
*     argument pairs.  For example, JD=2450123.7 could be expressed in
*     any of these ways, among others:
*
*            Part A         Part B
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable (in the case of UT;  the TT is not at all critical
*     in this respect).  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  For UT, the date & time
*     method is best matched to the algorithm that is used by the Earth
*     rotation angle routine, called internally:  maximum accuracy (or,
*     at least, minimum noise) is delivered when the UTA argument is for
*     0hrs UT1 on the day in question and the UTB argument lies in the
*     range 0 to 1, or vice versa.
*
*  2) Both UT1 and TT are required, UT1 to predict the Earth rotation
*     and TT to predict the effects of precession-nutation.  If UT1 is
*     used for both purposes, errors of order 100 microarcseconds
*     result.
*
*  3) Although the routine uses the IAU 2006 series for s+XY/2, it is
*     otherwise independent of the precession-nutation model and can in
*     practice be used with any equinox-based NPB matrix.
*
*  4) The result is returned in the range 0 to 2pi.
*
*  Called:
*     iau_BPN2XY   extract CIP X,Y coordinates from NPB matrix
*     iau_S06      the CIO locator s, given X,Y, IAU 2006
*     iau_ANP      normalize angle into range 0 to 2pi
*     iau_ERA00    Earth rotation angle, IAU 2000
*     iau_EORS     equation of the origins, given NPB matrix and s
*
*  Reference:
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_GST06A ( UTA, UTB, TTA, TTB )
*+
*  - - - - - - - - - - -
*   i a u _ G S T 0 6 A
*  - - - - - - - - - - -
*
*  Greenwich apparent sidereal time (consistent with IAU 2000 and 2006
*  resolutions).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     UTA, UTB     d      UT1 as a 2-part Julian Date (Notes 1,2)
*     TTA, TTB     d      TT as a 2-part Julian Date (Notes 1,2)
*
*  Returned:
*     iau_GST06A   d      Greenwich apparent sidereal time (radians)
*
*  Notes:
*
*  1) The UT1 and TT dates UTA+UTB and TTA+TTB respectively, are both
*     Julian Dates, apportioned in any convenient way between the
*     argument pairs.  For example, JD=2450123.7 could be expressed in
*     any of these ways, among others:
*
*            Part A         Part B
*
*        2450123.7D0        0D0        (JD method)
*         2451545D0      -1421.3D0     (J2000 method)
*        2400000.5D0     50123.2D0     (MJD method)
*        2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable (in the case of UT;  the TT is not at all critical
*     in this respect).  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  For UT, the date & time
*     method is best matched to the algorithm that is used by the Earth
*     rotation angle routine, called internally:  maximum accuracy (or,
*     at least, minimum noise) is delivered when the UTA argument is for
*     0hrs UT1 on the day in question and the UTB argument lies in the
*     range 0 to 1, or vice versa.
*
*  2) Both UT1 and TT are required, UT1 to predict the Earth rotation
*     and TT to predict the effects of precession-nutation.  If UT1 is
*     used for both purposes, errors of order 100 microarcseconds
*     result.
*
*  3) This GAST is compatible with the IAU 2000/2006 resolutions and
*     must be used only in conjunction with IAU 2006 precession and
*     IAU 2000A nutation.
*
*  4) The result is returned in the range 0 to 2pi.
*
*  Called:
*     iau_PNM06A   classical NPB matrix, IAU 2006/2000A
*     iau_GST06    Greenwich apparent ST, IAU 2006, given NPB matrix
*
*  Reference:
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_GST94 ( UTA, UTB )
*+
*  - - - - - - - - - -
*   i a u _ G S T 9 4
*  - - - - - - - - - -
*
*  Greenwich Apparent Sidereal Time (consistent with IAU 1982/94
*  resolutions).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     UTA, UTB     d      UT1 as a 2-part Julian Date (Notes 1,2)
*
*  Returned:
*     iau_GST94    d      Greenwich apparent sidereal time (radians)
*
*  Notes:
*
*  1) The UT1 date UTA+UTB is a Julian Date, apportioned in any
*     convenient way between the argument pair.  For example,
*     JD=2450123.7 could be expressed in any of these ways, among
*     others:
*
*             UTA            UTB
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in cases
*     where the loss of several decimal digits of resolution is
*     acceptable.  The J2000 and MJD methods are good compromises
*     between resolution and convenience.  For UT, the date & time
*     method is best matched to the algorithm that is used by the Earth
*     Rotation Angle routine, called internally:  maximum accuracy (or,
*     at least, minimum noise) is delivered when the UTA argument is for
*     0hrs UT1 on the day in question and the UTB argument lies in the
*     range 0 to 1, or vice versa.
*
*  2) The result is compatible with the IAU 1982 and 1994 resolutions,
*     except that accuracy has been compromised for the sake of
*     convenience in that UT is used instead of TDB (or TT) to compute
*     the equation of the equinoxes.
*
*  3) This GAST must be used only in conjunction with contemporaneous
*     IAU standards such as 1976 precession, 1980 obliquity and 1982
*     nutation.  It is not compatible with the IAU 2000 resolutions.
*
*  4) The result is returned in the range 0 to 2pi.
*
*  Called:
*     iau_GMST82   Greenwich mean sidereal time, IAU 1982
*     iau_EQEQ94   equation of the equinoxes, IAU 1994
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  References:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*     IAU Resolution C7, Recommendation 3 (1994)
*
*-
```

```
      SUBROUTINE iau_H2FK5 ( RH, DH, DRH, DDH, PXH, RVH,
     :                       R5, D5, DR5, DD5, PX5, RV5 )
*+
*  - - - - - - - - - -
*   i a u _ H 2 F K 5
*  - - - - - - - - - -
*
*  Transform Hipparcos star data into the FK5 (J2000.0) system.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given (all Hipparcos, epoch J2000.0):
*     RH        d      RA (radians)
*     DH        d      Dec (radians)
*     DRH       d      proper motion in RA (dRA/dt, rad/Jyear)
*     DDH       d      proper motion in Dec (dDec/dt, rad/Jyear)
*     PXH       d      parallax (arcsec)
*     RVH       d      radial velocity (km/s, positive = receding)
*
*  Returned (all FK5, equinox J2000.0, epoch J2000.0):
*     R5        d      RA (radians)
*     D5        d      Dec (radians)
*     DR5       d      proper motion in RA (dRA/dt, rad/Jyear)
*     DD5       d      proper motion in Dec (dDec/dt, rad/Jyear)
*     PX5       d      parallax (arcsec)
*     RV5       d      radial velocity (km/s, positive = receding)
*
*  Notes:
*
*  1) This routine transforms Hipparcos star positions and proper
*     motions into FK5 J2000.0.
*
*  2) The proper motions in RA are dRA/dt rather than cos(Dec)*dRA/dt,
*     and are per year rather than per century.
*
*  3) The FK5 to Hipparcos transformation is modeled as a pure rotation
*     and spin;  zonal errors in the FK5 catalog are not taken into
*     account.
*
*  4) See also iau_FK52H, iau_FK5HZ, iau_HFK5Z.
*
*  Called:
*     iau_STARPV   star catalog data to space motion pv-vector
*     iau_FK5HIP   FK5 to Hipparcos rotation and spin
*     iau_RV2M     r-vector to r-matrix
*     iau_RXP      product of r-matrix and p-vector
*     iau_TRXP     product of transpose of r-matrix and p-vector
*     iau_PXP      vector product of two p-vectors
*     iau_PMP      p-vector minus p-vector
*     iau_PVSTAR   space motion pv-vector to star catalog data
*
*  Reference:
*
*     F.Mignard & M.Froeschle, Astron. Astrophys. 354, 732-739 (2000).
*
*-
```

```
      SUBROUTINE iau_HFK5Z ( RH, DH, DATE1, DATE2, R5, D5, DR5, DD5 )
*+
*  - - - - - - - - - - -
*   i a u _ H F K 5 Z
*  - - - - - - - - - - -
*
*  Transform a Hipparcos star position into FK5 J2000.0, assuming
*  zero Hipparcos proper motion.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     RH              d      Hipparcos RA (radians)
*     DH              d      Hipparcos Dec (radians)
*     DATE1,DATE2     d      TDB date (Note 1)
*
*  Returned (all FK5, equinox J2000.0, date DATE1+DATE2):
*     R5              d      RA (radians)
*     D5              d      Dec (radians)
*     DR5             d      FK5 RA proper motion (rad/year, Note 4)
*     DD5             d      Dec proper motion (rad/year, Note 4)
*
*  Notes:
*
*  1) The date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TDB)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The proper motion in RA is dRA/dt rather than cos(Dec)*dRA/dt.
*
*  3) The FK5 to Hipparcos transformation is modeled as a pure
*     rotation and spin;  zonal errors in the FK5 catalogue are
*     not taken into account.
*
*  4) It was the intention that Hipparcos should be a close
*     approximation to an inertial frame, so that distant objects
*     have zero proper motion;  such objects have (in general)
*     non-zero proper motion in FK5, and this routine returns those
*     fictitious proper motions.
*
*  5) The position returned by this routine is in the FK5 J2000.0
*     reference system but at date DATE1+DATE2.
*
*  6) See also iau_FK52H, iau_H2FK5, iau_FK5ZHZ.
*
*  Called:
*     iau_S2C      spherical coordinates to unit vector
*     iau_FK5HIP   FK5 to Hipparcos rotation and spin
*     iau_RXP      product of r-matrix and p-vector
*     iau_SXP      multiply p-vector by scalar
*     iau_RXR      product of two r-matrices
*     iau_TRXP     product of transpose of r-matrix and p-vector
*     iau_PXP      vector product of two p-vectors
*     iau_PV2S     pv-vector to spherical
```

```
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  Reference:
*
*     F.Mignard & M.Froeschle, Astron. Astrophys. 354, 732-739 (2000).
*
*-
```

```
      SUBROUTINE iau_IR ( R )
*+
*  - - - - - - - -
*   i a u _ I R
*  - - - - - - - -
*
*  Initialize an r-matrix to the identity matrix.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Returned:
*     R        d(3,3)    r-matrix
*
*-
```

```
      SUBROUTINE iau_JD2CAL ( DJ1, DJ2, IY, IM, ID, FD, J )
*+
*  - - - - - - - - - - -
*   i a u _ J D 2 C A L
*  - - - - - - - - - - -
*
*  Julian Date to Gregorian year, month, day, and fraction of a day.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DJ1,DJ2     d     Julian Date (Notes 1, 2)
*
*  Returned:
*     IY          i     year
*     IM          i     month
*     ID          i     day
*     FD          d     fraction of day
*     J           i     status:
*                           0 = OK
*                          -1 = unacceptable date (Note 3)
*
*  Notes:
*
*  1) The earliest valid date is -68569.5 (-4900 March 1).  The
*     largest value accepted is 10^9.
*
*  2) The Julian Date is apportioned in any convenient way between
*     the arguments DJ1 and DJ2.  For example, JD=2450123.7 could
*     be expressed in any of these ways, among others:
*
*             DJ1            DJ2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*  3) In early eras the conversion is from the "Proleptic Gregorian
*     Calendar";  no account is taken of the date(s) of adoption of
*     the Gregorian Calendar, nor is the AD/BC numbering convention
*     observed.
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     Section 12.92 (p604).
*
*-
```

```
      SUBROUTINE iau_JDCALF ( NDP, DJ1, DJ2, IYMDF, J )
*+
*  - - - - - - - - - - -
*   i a u _ J D C A L F
*  - - - - - - - - - - -
*
*  Julian Date to Gregorian Calendar, expressed in a form convenient
*  for formatting messages:  rounded to a specified precision, and with
*  the fields stored in a single array.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     NDP         i     number of decimal places of days in fraction
*     DJ1,DJ2     d     DJ1+DJ2 = Julian Date (Note 1)
*
*  Returned:
*     IYMDF       i(4)  year, month, day, fraction in Gregorian
*                       calendar
*     J           i     status:
*                          -1 = date out of range
*                           0 = OK
*                          +1 = NDP not 0-9 (interpreted as 0)
*
*  Notes:
*
*  1) The Julian Date is apportioned in any convenient way between
*     the arguments DJ1 and DJ2.  For example, JD=2450123.7 could
*     be expressed in any of these ways, among others:
*
*             DJ1            DJ2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*  2) In early eras the conversion is from the "Proleptic Gregorian
*     Calendar";  no account is taken of the date(s) of adoption of
*     the Gregorian Calendar, nor is the AD/BC numbering convention
*     observed.
*
*  3) Refer to the routine iau_JD2CAL.
*
*  4) NDP should be 4 or less if internal overflows are to be
*     avoided on machines which use 16-bit integers.
*
*  Called:
*     iau_JD2CAL   JD to Gregorian calendar
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     Section 12.92 (p604).
*
*-
```

```
      SUBROUTINE iau_NUM00A ( DATE1, DATE2, RMATN )
*+
*  - - - - - - - - - - -
*   i a u _ N U M 0 0 A
*  - - - - - - - - - - -
*
*  Form the matrix of nutation for a given date, IAU 2000A model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RMATN        d(3,3)    nutation matrix
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(true) = RMATN * V(mean),
*     where the p-vector V(true) is with respect to the true
*     equatorial triad of date and the p-vector V(mean) is with
*     respect to the mean equatorial triad of date.
*
*  3) A faster, but slightly less accurate result (about 1 mas), can be
*     obtained by using instead the iau_NUM00B routine.
*
*  Called:
*     iau_PN00A    bias/precession/nutation, IAU 2000A
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     Section 3.222-3 (p114).
*
*-
```

```
      SUBROUTINE iau_NUM00B ( DATE1, DATE2, RMATN )
*+
*  - - - - - - - - - - -
*   i a u _ N U M 0 0 B
*  - - - - - - - - - - -
*
*  Form the matrix of nutation for a given date, IAU 2000B model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RMATN       d(3,3)   nutation matrix
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(true) = RMATN * V(mean),
*     where the p-vector V(true) is with respect to the true
*     equatorial triad of date and the p-vector V(mean) is with
*     respect to the mean equatorial triad of date.
*
*  3) The present routine is faster, but slightly less accurate (about
*     1 mas), than the iau_NUM00A routine.
*
*  Called:
*     iau_PN00B    bias/precession/nutation, IAU 2000B
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     Section 3.222-3 (p114).
*
*-
```

```
      SUBROUTINE iau_NUM06A ( DATE1, DATE2, RMATN )
*+
*  - - - - - - - - - - -
*   i a u _ N U M 0 6 A
*  - - - - - - - - - - -
*
*  Form the matrix of nutation for a given date, IAU 2006/2000A model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RMATN        d(3,3)    nutation matrix
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(true) = RMATN * V(mean),
*     where the p-vector V(true) is with respect to the true
*     equatorial triad of date and the p-vector V(mean) is with
*     respect to the mean equatorial triad of date.
*
*  Called:
*     iau_OBL06    mean obliquity, IAU 2006
*     iau_NUT06A   nutation, IAU 2006/2000A
*     iau_NUMAT    form nutation matrix
*
*  References:
*
*     Capitaine, N., Wallace, P.T. & Chapront, J., 2005, Astron.
*     Astrophys. 432, 355
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      SUBROUTINE iau_NUMAT ( EPSA, DPSI, DEPS, RMATN )
*+
*  - - - - - - - - - -
*   i a u _ N U M A T
*  - - - - - - - - - -
*
*  Form the matrix of nutation.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     EPSA          d       mean obliquity of date (Note 1)
*     DPSI,DEPS     d       nutation (Note 2)
*
*  Returned:
*     RMATN       d(3,3)    nutation matrix (Note 3)
*
*  Notes:
*
*
*  1) The supplied mean obliquity EPSA, must be consistent with the
*     precession-nutation models from which DPSI and DEPS were obtained.
*
*  2) The caller is responsible for providing the nutation components;
*     they are in longitude and obliquity, in radians and are with
*     respect to the equinox and ecliptic of date.
*
*  3) The matrix operates in the sense V(true) = RMATN * V(mean),
*     where the p-vector V(true) is with respect to the true
*     equatorial triad of date and the p-vector V(mean) is with
*     respect to the mean equatorial triad of date.
*
*  Called:
*     iau_IR       initialize r-matrix to identity
*     iau_RX       rotate around X-axis
*     iau_RZ       rotate around Z-axis
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     Section 3.222-3 (p114).
*
*-
```

```
      SUBROUTINE iau_NUT00A ( DATE1, DATE2, DPSI, DEPS )
*+
*  - - - - - - - - - - -
*   i a u _ N U T 0 0 A
*  - - - - - - - - - - -
*
*  Nutation, IAU 2000A model (MHB2000 luni-solar and planetary nutation
*  with free core nutation omitted).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2   d    TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     DPSI,DEPS     d    nutation, luni-solar + planetary (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The nutation components in longitude and obliquity are in radians
*     and with respect to the equinox and ecliptic of date.  The
*     obliquity at J2000.0 is assumed to be the Lieske et al. (1977)
*     value of 84381.448 arcsec.
*
*     Both the luni-solar and planetary nutations are included.  The
*     latter are due to direct planetary nutations and the perturbations
*     of the lunar and terrestrial orbits.
*
*  3) The routine computes the MHB2000 nutation series with the
*     associated corrections for planetary nutations.  It is an
*     implementation of the nutation part of the IAU 2000A precession-
*     nutation model, formally adopted by the IAU General Assembly in
*     2000, namely MHB2000 (Mathews et al. 2002), but with the free core
*     nutation (FCN - see Note 4) omitted.
*
*  4) The full MHB2000 model also contains contributions to the
*     nutations in longitude and obliquity due to the free-excitation of
*     the free-core-nutation during the period 1979-2000.  These FCN
*     terms, which are time-dependent and unpredictable, are NOT
*     included in the present routine and, if required, must be
*     independently computed.  With the FCN corrections included, the
*     present routine delivers a pole which is at current epochs
*     accurate to a few hundred microarcseconds.  The omission of FCN
*     introduces further errors of about that size.
*
*  5) The present routine provides classical nutation.  The MHB2000
*     algorithm, from which it is adapted, deals also with (i) the
*     offsets between the GCRS and mean poles and (ii) the adjustments
*     in longitude and obliquity due to the changed precession rates.
*     These additional functions, namely frame bias and precession
```

```
*     adjustments, are supported by the SOFA routines iau_BI00 and
*     iau_PR00.
*
*  6) The MHB2000 algorithm also provides "total" nutations, comprising
*     the arithmetic sum of the frame bias, precession adjustments,
*     luni-solar nutation and planetary nutation.  These total nutations
*     can be used in combination with an existing IAU 1976 precession
*     implementation, such as iau_PMAT76, to deliver GCRS-to-true
*     predictions of sub-mas accuracy at current epochs.  However, there
*     are three shortcomings in the MHB2000 model that must be taken
*     into account if more accurate or definitive results are required
*     (see Wallace 2002):
*
*       (i) The MHB2000 total nutations are simply arithmetic sums,
*           yet in reality the various components are successive Euler
*           rotations.  This slight lack of rigor leads to cross terms
*           that exceed 1 mas after a century.  The rigorous procedure
*           is to form the GCRS-to-true rotation matrix by applying the
*           bias, precession and nutation in that order.
*
*      (ii) Although the precession adjustments are stated to be with
*           respect to Lieske et al. (1977), the MHB2000 model does
*           not specify which set of Euler angles are to be used and
*           how the adjustments are to be applied.  The most literal and
*           straightforward procedure is to adopt the 4-rotation
*           epsilon_0, psi_A, omega_A, xi_A option, and to add DPSIPR to
*           psi_A and DEPSPR to both omega_A and eps_A.
*
*     (iii) The MHB2000 model predates the determination by Chapront
*           et al. (2002) of a 14.6 mas displacement between the J2000.0
*           mean equinox and the origin of the ICRS frame.  It should,
*           however, be noted that neglecting this displacement when
*           calculating star coordinates does not lead to a 14.6 mas
*           change in right ascension, only a small second-order
*           distortion in the pattern of the precession-nutation effect.
*
*     For these reasons, the SOFA routines do not generate the "total
*     nutations" directly, though they can of course easily be generated
*     by calling iau_BI00, iau_PR00 and the present routine and adding
*     the results.
*
*  7) The MHB2000 model contains 41 instances where the same frequency
*     appears multiple times, of which 38 are duplicates and three are
*     triplicates.  To keep the present code close to the original MHB
*     algorithm, this small inefficiency has not been corrected.
*
*  Called:
*     iau_FAL03    mean anomaly of the Moon
*     iau_FAF03    mean argument of the latitude of the Moon
*     iau_FAOM03   mean longitude of the Moon's ascending node
*     iau_FAME03   mean longitude of Mercury
*     iau_FAVE03   mean longitude of Venus
*     iau_FAE03    mean longitude of Earth
*     iau_FAMA03   mean longitude of Mars
*     iau_FAJU03   mean longitude of Jupiter
*     iau_FASA03   mean longitude of Saturn
*     iau_FAUR03   mean longitude of Uranus
*     iau_FAPA03   general accumulated precession in longitude
*
*  References:
*
*     Chapront, J., Chapront-Touze, M. & Francou, G. 2002,
*     Astron.Astrophys. 387, 700
*
*     Lieske, J.H., Lederle, T., Fricke, W. & Morando, B. 1977,
*     Astron.Astrophys. 58, 1-16
*
*     Mathews, P.M., Herring, T.A., Buffet, B.A. 2002, J.Geophys.Res.
*     107, B4.  The MHB_2000 code itself was obtained on 9th September
*     2002 from ftp//maia.usno.navy.mil/conv2000/chapter5/IAU2000A.
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J. 1994, Astron.Astrophys. 282, 663-683
```

```
*
*     Souchay, J., Loysel, B., Kinoshita, H., Folgueira, M. 1999,
*     Astron.Astrophys.Supp.Ser. 135, 111
*
*     Wallace, P.T., "Software for Implementing the IAU 2000
*     Resolutions", in IERS Workshop 5.1 (2002)
*
*-
```

```
      SUBROUTINE iau_NUT00B ( DATE1, DATE2, DPSI, DEPS )
*+
*  - - - - - - - - - - -
*   i a u _ N U T 0 0 B
*  - - - - - - - - - - -
*
*  Nutation, IAU 2000B model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2   d   TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     DPSI,DEPS     d   nutation, luni-solar + planetary (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in cases
*     where the loss of several decimal digits of resolution is
*     acceptable.  The J2000 method is best matched to the way the
*     argument is handled internally and will deliver the optimum
*     resolution.  The MJD method and the date & time methods are both
*     good compromises between resolution and convenience.
*
*  2) The nutation components in longitude and obliquity are in radians
*     and with respect to the equinox and ecliptic of date.  The
*     obliquity at J2000.0 is assumed to be the Lieske et al. (1977)
*     value of 84381.448 arcsec.  (The errors that result from using
*     this routine with the IAU 2006 value of 84381.406 arcsec can be
*     neglected.)
*
*     The nutation model consists only of luni-solar terms, but includes
*     also a fixed offset which compensates for certain long-period
*     planetary terms (Note 7).
*
*  3) This routine is an implementation of the IAU 2000B abridged
*     nutation model formally adopted by the IAU General Assembly in
*     2000.  The routine computes the MHB_2000_SHORT luni-solar nutation
*     series (Luzum 2001), but without the associated corrections for
*     the precession rate adjustments and the offset between the GCRS
*     and J2000.0 mean poles.
*
*  4) The full IAU 2000A (MHB2000) nutation model contains nearly 1400
*     terms.  The IAU 2000B model (McCarthy & Luzum 2003) contains only
*     77 terms, plus additional simplifications, yet still delivers
*     results of 1 mas accuracy at present epochs.  This combination of
*     accuracy and size makes the IAU 2000B abridged nutation model
*     suitable for most practical applications.
*
*     The routine delivers a pole accurate to 1 mas from 1900 to 2100
*     (usually better than 1 mas, very occasionally just outside 1 mas).
*     The full IAU 2000A model, which is implemented in the routine
*     iau_NUT00A (q.v.), delivers considerably greater accuracy at
*     current epochs;  however, to realize this improved accuracy,
*     corrections for the essentially unpredictable free-core-nutation
*     (FCN) must also be included.
```

```
*
*  5) The present routine provides classical nutation.  The
*     MHB_2000_SHORT algorithm, from which it is adapted, deals also
*     with (i) the offsets between the GCRS and mean poles and (ii) the
*     adjustments in longitude and obliquity due to the changed
*     precession rates.  These additional functions, namely frame bias
*     and precession adjustments, are supported by the SOFA routines
*     iau_BI00 and iau_PR00.
*
*  6) The MHB_2000_SHORT algorithm also provides "total" nutations,
*     comprising the arithmetic sum of the frame bias, precession
*     adjustments, and nutation (luni-solar + planetary).  These total
*     nutations can be used in combination with an existing IAU 1976
*     precession implementation, such as iau_PMAT76, to deliver GCRS-to-
*     true predictions of mas accuracy at current epochs.  However, for
*     symmetry with the iau_NUT00A routine (q.v. for the reasons), the
*     SOFA routines do not generate the "total nutations" directly.
*     Should they be required, they could of course easily be generated
*     by calling iau_BI00, iau_PR00 and the present routine and adding
*     the results.
*
*  7) The IAU 2000B model includes "planetary bias" terms that are fixed
*     in size but compensate for long-period nutations.  The amplitudes
*     quoted in McCarthy & Luzum (2003), namely Dpsi = -1.5835 mas and
*     Depsilon = +1.6339 mas, are optimized for the "total nutations"
*     method described in Note 6.  The Luzum (2001) values used in this
*     SOFA implementation, namely -0.135 mas and +0.388 mas, are
*     optimized for the "rigorous" method, where frame bias, precession
*     and nutation are applied separately and in that order.  During the
*     interval 1995-2050, the SOFA implementation delivers a maximum
*     error of 1.001 mas (not including FCN).
*
*  References:
*
*     Lieske, J.H., Lederle, T., Fricke, W., Morando, B., "Expressions
*     for the precession quantities based upon the IAU /1976/ system of
*     astronomical constants", Astron.Astrophys. 58, 1-2, 1-16. (1977)
*
*     Luzum, B., private communication, 2001 (Fortran code
*     MHB_2000_SHORT)
*
*     McCarthy, D.D. & Luzum, B.J., "An abridged model of the
*     precession-nutation of the celestial pole", Cel.Mech.Dyn.Astron.
*     85, 37-49 (2003)
*
*     Simon, J.-L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G., Laskar, J., Astron.Astrophys. 282, 663-683 (1994)
*
*-
```

```
      SUBROUTINE iau_NUT06A ( DATE1, DATE2, DPSI, DEPS )
*+
*  - - - - - - - - - - -
*   i a u _ N U T 0 6 A
*  - - - - - - - - - - -
*
*  IAU 2000A nutation with adjustments to match the IAU 2006 precession.
*
*  Given:
*     DATE1,DATE2   d    TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     DPSI,DEPS     d    nutation, luni-solar + planetary (Note 2)
*
*  Status:  canonical model.
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The nutation components in longitude and obliquity are in radians
*     and with respect to the mean equinox and ecliptic of date,
*     IAU 2006 precession model (Hilton et al. 2006, Capitaine et al.
*     2005).
*
*  3) The routine first computes the IAU 2000A nutation, then applies
*     adjustments for (i) the consequences of the change in obliquity
*     from the IAU 1980 ecliptic to the IAU 2006 ecliptic and (ii) the
*     secular variation in the Earth's dynamical form factor J2.
*
*  4) The present routine provides classical nutation, complementing
*     the IAU 2000 frame bias and IAU 2006 precession.  It delivers a
*     pole which is at current epochs accurate to a few tens of
*     microarcseconds, apart from the free core nutation.
*
*  Called:
*     iau_NUT00A   nutation, IAU 2000A
*
*  Reference:
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      SUBROUTINE iau_NUT80 ( DATE1, DATE2, DPSI, DEPS )
*+
*  - - - - - - - - - - 
*   i a u _ N U T 8 0
*  - - - - - - - - - - 
*
*  Nutation, IAU 1980 model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2     d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     DPSI            d      nutation in longitude (radians)
*     DEPS            d      nutation in obliquity (radians)
*
*  Notes:
*
*  1) The DATE DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TDB)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The nutation components are with respect to the ecliptic of
*     date.
*
*  Called:
*     iau_ANPM     normalize angle into range +/- pi
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     Section 3.222 (p111).
*
*-
```

```
      SUBROUTINE iau_NUTM80 ( DATE1, DATE2, RMATN )
*+
*  - - - - - - - - - - -
*   i a u _ N U T M 8 0
*  - - - - - - - - - - -
*
*  Form the matrix of nutation for a given date, IAU 1980 model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2      d       TDB date (Note 1)
*
*  Returned:
*     RMATN          d(3,3)    nutation matrix
*
*  Notes:
*
*  1) The date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TDB)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(true) = RMATN * V(mean),
*     where the p-vector V(true) is with respect to the true
*     equatorial triad of date and the p-vector V(mean) is with
*     respect to the mean equatorial triad of date.
*
*  Called:
*     iau_NUT80    nutation, IAU 1980
*     iau_OBL80    mean obliquity, IAU 1980
*     iau_NUMAT    form nutation matrix
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_OBL06 ( DATE1, DATE2 )
*+
*  - - - - - - - - - -
*   i a u _ O B L 0 6
*  - - - - - - - - - -
*
*  Mean obliquity of the ecliptic, IAU 2006 precession model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_OBL06      d      obliquity of the ecliptic (radians, Note 2)
*
*  Notes:
*
*  1) The date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*             DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The result is the angle between the ecliptic and mean equator of
*     date DATE1+DATE2.
*
*  Reference:
*
*     Hilton, J. et al., 2006, Celest.Mech.Dyn.Astron. 94, 351
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_OBL80 ( DATE1, DATE2 )
*+
*  - - - - - - - - - -
*   i a u _ O B L 8 0
*  - - - - - - - - - -
*
*  Mean obliquity of the ecliptic, IAU 1980 model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2     d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_OBL80       d      obliquity of the ecliptic (radians, Note 2)
*
*  Notes:
*
*  1) The date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TDB)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*             DATE1         DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The result is the angle between the ecliptic and mean equator of
*     date DATE1+DATE2.
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     Expression 3.222-1 (p114).
*
*-
```

```
      SUBROUTINE iau_P06E ( DATE1, DATE2,
     :                      EPS0, PSIA, OMA, BPA, BQA, PIA, BPIA,
     :                      EPSA, CHIA, ZA, ZETAA, THETAA, PA,
     :                      GAM, PHI, PSI )
*+
*  - - - - - - - - -
*   i a u _ P 0 6 E
*  - - - - - - - - -
*
*  Precession angles, IAU 2006, equinox based.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical models.
*
*  Given:
*     DATE1,DATE2   d    TT as a 2-part Julian Date (Note 1)
*
*  Returned (see Note 2):
*     EPS0          d    epsilon_0
*     PSIA          d    psi_A
*     OMA           d    omega_A
*     BPA           d    P_A
*     BQA           d    Q_A
*     PIA           d    pi_A
*     BPIA          d    Pi_A
*     EPSA          d    obliquity epsilon_A
*     CHIA          d    chi_A
*     ZA            d    z_A
*     ZETAA         d    zeta_A
*     THETAA        d    theta_A
*     PA            d    p_A
*     GAM           d    F-W angle gamma_J2000
*     PHI           d    F-W angle phi_J2000
*     PSI           d    F-W angle psi_J2000
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) This routine returns the set of equinox based angles for the
*     Capitaine et al. "P03" precession theory, adopted by the IAU in
*     2006.  The angles are set out in Table 1 of Hilton et al. (2006):
*
*      EPS0   epsilon_0   obliquity at J2000.0
*      PSIA   psi_A       luni-solar precession
*      OMA    omega_A     inclination of equator wrt J2000.0 ecliptic
*      BPA    P_A         ecliptic pole x, J2000.0 ecliptic triad
*      BQA    Q_A         ecliptic pole -y, J2000.0 ecliptic triad
*      PIA    pi_A        angle between moving and J2000.0 ecliptics
*      BPIA   Pi_A        longitude of ascending node of the ecliptic
*      EPSA   epsilon_A   obliquity of the ecliptic
*      CHIA   chi_A       planetary precession
*      ZA     z_A         equatorial precession: -3rd 323 Euler angle
```

```
*      ZETAA  zeta_A      equatorial precession: -1st 323 Euler angle
*      THETAA theta_A     equatorial precession: 2nd 323 Euler angle
*      PA     p_A         general precession
*      GAM    gamma_J2000 J2000.0 RA difference of ecliptic poles
*      PHI    phi_J2000   J2000.0 codeclination of ecliptic pole
*      PSI    psi_J2000   longitude difference of equator poles, J2000.0
*
*     The returned values are all radians.
*
*  3) Hilton et al. (2006) Table 1 also contains angles that depend on
*     models distinct from the P03 precession theory itself, namely the
*     IAU 2000A frame bias and nutation.  The quoted polynomials are
*     used in other SOFA routines:
*
*     . iau_XY06 contains the polynomial parts of the X and Y series.
*
*     . iau_S06 contains the polynomial part of the s+XY/2 series.
*
*     . iau_PFW06 implements the series for the Fukushima-Williams
*       angles that are with respect to the GCRS pole (i.e. the variants
*       that include frame bias).
*
*  4) The IAU resolution stipulated that the choice of parameterization
*     was left to the user, and so an IAU compliant precession
*     implementation can be constructed using various combinations of
*     the angles returned by the present routine.
*
*  5) The parameterization used by SOFA is the version of the Fukushima-
*     Williams angles that refers directly to the GCRS pole.  These
*     angles may be calculated by calling the routine iau_PFW06.  SOFA
*     also supports the direct computation of the CIP GCRS X,Y by
*     series, available by calling iau_XY06.
*
*  6) The agreement between the different parameterizations is at the
*     1 microarcsecond level in the present era.
*
*  7) When constructing a precession formulation that refers to the GCRS
*     pole rather than the dynamical pole, it may (depending on the
*     choice of angles) be necessary to introduce the frame bias
*     explicitly.
*
*  Reference:
*
*     Hilton, J. et al., 2006, Celest.Mech.Dyn.Astron. 94, 351
*
*  Called:
*     iau_OBL06    mean obliquity, IAU 2006
*
*-
```

```
      SUBROUTINE iau_P2PV ( P, PV )
*+
*  - - - - - - - - -
*   i a u _ P 2 P V
*  - - - - - - - - -
*
*  Extend a p-vector to a pv-vector by appending a zero velocity.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     P        d(3)      p-vector
*
*  Returned:
*     PV       d(3,2)    pv-vector
*
*  Called:
*     iau_CP       copy p-vector
*     iau_ZP       zero p-vector
*
*-
```

```
      SUBROUTINE iau_P2S ( P, THETA, PHI, R )
*+
*  - - - - - - - -
*   i a u _ P 2 S
*  - - - - - - - -
*
*  P-vector to spherical polar coordinates.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     P        d(3)      p-vector
*
*  Returned:
*     THETA    d         longitude angle (radians)
*     PHI      d         latitude angle (radians)
*     R        d         radial distance
*
*  Notes:
*
*  1) If P is null, zero THETA, PHI and R are returned.
*
*  2) At either pole, zero THETA is returned.
*
*  Called:
*     iau_C2S      p-vector to spherical
*     iau_PM       modulus of p-vector
*
*-
```

```
      SUBROUTINE iau_PAP ( A, B, THETA )
*+
*  - - - - - - - -
*   i a u _ P A P
*  - - - - - - - -
*
*  Position-angle from two p-vectors.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3)      direction of reference point
*     B        d(3)      direction of point whose PA is required
*
*  Returned:
*     THETA    d         position angle of B with respect to A (radians)
*
*  Notes:
*
*  1) The result is the position angle, in radians, of direction B with
*     respect to direction A.  It is in the range -pi to +pi.  The sense
*     is such that if B is a small distance "north" of A the position
*     angle is approximately zero, and if B is a small distance "east" of
*     A the position angle is approximately +pi/2.
*
*  2) A and B need not be unit vectors.
*
*  3) Zero is returned if the two directions are the same or if either
*     vector is null.
*
*  4) If A is at a pole, the result is ill-defined.
*
*  Called:
*     iau_PN       decompose p-vector into modulus and direction
*     iau_PM       modulus of p-vector
*     iau_PXP      vector product of two p-vectors
*     iau_PMP      p-vector minus p-vector
*     iau_PDP      scalar product of two p-vectors
*
*-
```

```
      SUBROUTINE iau_PAS ( AL, AP, BL, BP, THETA )
*+
*  - - - - - - - -
*   i a u _ P A S
*  - - - - - - - -
*
*  Position-angle from spherical coordinates.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     AL       d       longitude of point A (e.g. RA) in radians
*     AP       d       latitude of point A (e.g. Dec) in radians
*     BL       d       longitude of point B
*     BP       d       latitude of point B
*
*  Returned:
*     THETA    d       position angle of B with respect to A
*
*  Notes:
*
*  1) The result is the bearing (position angle), in radians, of point
*     B with respect to point A.  It is in the range -pi to +pi.  The
*     sense is such that if B is a small distance "east" of point A,
*     the bearing is approximately +pi/2.
*
*  2) Zero is returned if the two points are coincident.
*
*-
```

```
      SUBROUTINE iau_PB06 ( DATE1, DATE2, BZETA, BZ, BTHETA )
*+
*  - - - - - - - - -
*   i a u _ P B 0 6
*  - - - - - - - - -
*
*  This routine forms three Euler angles which implement general
*  precession from epoch J2000.0, using the IAU 2006 model.  Frame
*  bias (the offset between ICRS and mean J2000.0) is included.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d    TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     BZETA         d    1st rotation: radians clockwise around z
*     BZ            d    3rd rotation: radians clockwise around z
*     BTHETA        d    2nd rotation: radians counterclockwise around y
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the arguments DATE1 and DATE2.  For
*     example, JD(TT)=2450123.7 could be expressed in any of these
*     ways, among others:
*
*             DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The traditional accumulated precession angles zeta_A, z_A, theta_A
*     cannot be obtained in the usual way, namely through polynomial
*     expressions, because of the frame bias.  The latter means that two
*     of the angles undergo rapid changes near this date.  They are
*     instead the results of decomposing the precession-bias matrix
*     obtained by using the Fukushima-Williams method, which does not
*     suffer from the problem.  The decomposition returns values which
*     can be used in the conventional formulation and which include
*     frame bias.
*
*  3) The three angles are returned in the conventional order, which
*     is not the same as the order of the corresponding Euler rotations.
*     The precession-bias matrix is R_3(-z) x R_2(+theta) x R_3(-zeta).
*
*  4) Should zeta_A, z_A, theta_A angles be required that do not contain
*     frame bias, they are available by calling the SOFA routine
*     iau_P06E.
*
*  Called:
*     iau_PMAT06   PB matrix, IAU 2006
*     iau_RZ       rotate around Z-axis
*
*-
```

```
      SUBROUTINE iau_PDP ( A, B, ADB )
*+
*  - - - - - - - -
*   i a u _ P D P
*  - - - - - - - -
*
*  p-vector inner (=scalar=dot) product.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3)      first p-vector
*     B        d(3)      second p-vector
*
*  Returned:
*     ADB      d         A . B
*
*-
```

```
      SUBROUTINE iau_PFW06 ( DATE1, DATE2, GAMB, PHIB, PSIB, EPSA )
*+
*  - - - - - - - - - -
*   i a u _ P F W 0 6
*  - - - - - - - - - -
*
*  Precession angles, IAU 2006 (Fukushima-Williams 4-angle formulation).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2   d    TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     GAMB          d    F-W angle gamma_bar (radians)
*     PHIB          d    F-W angle phi_bar (radians)
*     PSIB          d    F-W angle psi_bar (radians)
*     EPSA          d    F-W angle epsilon_A (radians)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) Naming the following points:
*
*           e = J2000.0 ecliptic pole,
*           p = GCRS pole,
*           E = mean ecliptic pole of date,
*     and   P = mean pole of date,
*
*     the four Fukushima-Williams angles are as follows:
*
*        GAMB = gamma_bar = epE
*        PHIB = phi_bar = pE
*        PSIB = psi_bar = pEP
*        EPSA = epsilon_A = EP
*
*  3) The matrix representing the combined effects of frame bias and
*     precession is:
*
*        PxB = R_1(-EPSA).R_3(-PSIB).R_1(PHIB).R_3(GAMB)
*
*  4) The matrix representing the combined effects of frame bias,
*     precession and nutation is simply:
*
*        NxPxB = R_1(-EPSA-dE).R_3(-PSIB-dP).R_1(PHIB).R_3(GAMB)
*
*     where dP and dE are the nutation components with respect to the
*     ecliptic of date.
*
*  Reference:
*
```

```
*     Hilton, J. et al., 2006, Celest.Mech.Dyn.Astron. 94, 351
*
*  Called:
*     iau_OBL06    mean obliquity, IAU 2006
*
*-
```

```
      SUBROUTINE iau_PLAN94 ( DATE1, DATE2, NP, PV, J )
*+
*  - - - - - - - - - - -
*   i a u _ P L A N 9 4
*  - - - - - - - - - - -
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Approximate heliocentric position and velocity of a nominated major
*  planet:  Mercury, Venus, EMB, Mars, Jupiter, Saturn, Uranus or
*  Neptune (but not the Earth itself).
*
*  Given:
*     DATE1    d       TDB date part A (Note 1)
*     DATE2    d       TDB date part B (Note 1)
*     NP       i       planet (1=Mercury, 2=Venus, 3=EMB ... 8=Neptune)
*
*  Returned:
*     PV       d(3,2)  planet pos,vel (heliocentric, J2000.0, AU, AU/d)
*     J        i       status: -1 = illegal NP (outside 1-8)
*                               0 = OK
*                              +1 = warning: date outside 1000-3000 AD
*                              +2 = warning: solution failed to converge
*
*  Notes:
*
*  1) The date DATE1+DATE2 is in the TDB timescale and is a Julian Date,
*     apportioned in any convenient way between the two arguments.  For
*     example, JD(TDB)=2450123.7 could be expressed in any of these
*     ways, among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*     The limited accuracy of the present algorithm is such that any
*     of the methods is satisfactory.
*
*  2) If an NP value outside the range 1-8 is supplied, an error
*     status (J = -1) is returned and the PV vector set to zeroes.
*
*  3) For NP=3 the result is for the Earth-Moon Barycenter.  To
*     obtain the heliocentric position and velocity of the Earth,
*     use instead the SOFA routine iau_EPV00.
*
*  4) On successful return, the array PV contains the following:
*
*        PV(1,1)  x       }
*        PV(2,1)  y       } heliocentric position, AU
*        PV(3,1)  z       }
*
*        PV(1,2)  xdot    }
*        PV(2,2)  ydot    } heliocentric velocity, AU/d
*        PV(3,2)  zdot    }
*
*     The reference frame is equatorial and is with respect to the
*     mean equator and equinox of epoch J2000.0.
*
*  5) The algorithm is due to J.L. Simon, P. Bretagnon, J. Chapront,
*     M. Chapront-Touze, G. Francou and J. Laskar (Bureau des
```

```
*     Longitudes, Paris, France).  From comparisons with JPL
*     ephemeris DE102, they quote the following maximum errors
*     over the interval 1800-2050:
*
*                     L (arcsec)    B (arcsec)      R (km)
*
*        Mercury          4             1             300
*        Venus            5             1             800
*        EMB              6             1            1000
*        Mars            17             1            7700
*        Jupiter         71             5           76000
*        Saturn          81            13          267000
*        Uranus          86             7          712000
*        Neptune         11             1          253000
*
*     Over the interval 1000-3000, they report that the accuracy is no
*     worse than 1.5 times that over 1800-2050.  Outside 1000-3000 the
*     accuracy declines.
*
*     Comparisons of the present routine with the JPL DE200 ephemeris
*     give the following RMS errors over the interval 1960-2025:
*
*                      position (km)     velocity (m/s)
*
*        Mercury            334               0.437
*        Venus             1060               0.855
*        EMB               2010               0.815
*        Mars              7690               1.98
*        Jupiter          71700               7.70
*        Saturn          199000              19.4
*        Uranus          564000              16.4
*        Neptune         158000              14.4
*
*     Comparisons against DE200 over the interval 1800-2100 gave the
*     following maximum absolute differences.  (The results using
*     DE406 were essentially the same.)
*
*                   L (arcsec)   B (arcsec)     R (km)   Rdot (m/s)
*
*        Mercury        7            1            500       0.7
*        Venus          7            1           1100       0.9
*        EMB            9            1           1300       1.0
*        Mars          26            1           9000       2.5
*        Jupiter       78            6          82000       8.2
*        Saturn        87           14         263000      24.6
*        Uranus        86            7         661000      27.4
*        Neptune       11            2         248000      21.4
*
*  6) The present SOFA re-implementation of the original Simon et al.
*     Fortran code differs from the original in the following respects:
*
*       *  The date is supplied in two parts.
*
*       *  The result is returned only in equatorial Cartesian form;
*          the ecliptic longitude, latitude and radius vector are not
*          returned.
*
*       *  The result is in the J2000.0 equatorial frame, not ecliptic.
*
*       *  More is done in-line:  there are fewer calls to other
*          routines.
*
*       *  Different error/warning status values are used.
*
*       *  A different Kepler's-equation-solver is used (avoiding
*          use of COMPLEX*16).
*
*       *  Polynomials in T are nested to minimize rounding errors.
*
*       *  Explicit double-precision constants are used to avoid
*          mixed-mode expressions.
*
*       *  There are other, cosmetic, changes to comply with SOFA
```

```
*          style conventions.
*
*     None of the above changes affects the result significantly.
*
*  7) The returned status, J, indicates the most serious condition
*     encountered during execution of the routine.  Illegal NP is
*     considered the most serious, overriding failure to converge,
*     which in turn takes precedence over the remote epoch warning.
*
*  Called:
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  Reference:  Simon, J.L, Bretagnon, P., Chapront, J.,
*              Chapront-Touze, M., Francou, G., and Laskar, J.,
*              Astron. Astrophys. 282, 663 (1994).
*
*-
```

```
      SUBROUTINE iau_PM ( P, R )
*+
*  - - - - - - - -
*   i a u _ P M
*  - - - - - - - -
*
*  Modulus of p-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     P        d(3)      p-vector
*
*  Returned:
*     R        d         modulus
*
*-
```

```
      SUBROUTINE iau_PMAT00 ( DATE1, DATE2, RBP )
*+
*  - - - - - - - - - - -
*   i a u _ P M A T 0 0
*  - - - - - - - - - - -
*
*  Precession matrix (including frame bias) from GCRS to a specified
*  date, IAU 2000 model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RBP         d(3,3)    bias-precession matrix (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the arguments DATE1 and DATE2.  For
*     example, JD(TT)=2450123.7 could be expressed in any of these
*     ways, among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(date) = RBP * V(GCRS), where
*     the p-vector V(GCRS) is with respect to the Geocentric Celestial
*     Reference System (IAU, 2000) and the p-vector V(date) is with
*     respect to the mean equatorial triad of the given date.
*
*  Called:
*     iau_BP00     frame bias and precession matrices, IAU 2000
*
*  Reference:
*
*     IAU: Trans. International Astronomical Union, Vol. XXIVB;  Proc.
*     24th General Assembly, Manchester, UK.  Resolutions B1.3, B1.6.
*     (2000)
*
*-
```

```
      SUBROUTINE iau_PMAT06 ( DATE1, DATE2, RBP )
*+
*  - - - - - - - - - - -
*   i a u _ P M A T 0 6
*  - - - - - - - - - - -
*
*  Precession matrix (including frame bias) from GCRS to a specified
*  date, IAU 2006 model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RBP          d(3,3)    bias-precession matrix (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the arguments DATE1 and DATE2.  For
*     example, JD(TT)=2450123.7 could be expressed in any of these
*     ways, among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(date) = RBP * V(GCRS), where
*     the p-vector V(GCRS) is with respect to the Geocentric Celestial
*     Reference System (IAU, 2000) and the p-vector V(date) is with
*     respect to the mean equatorial triad of the given date.
*
*  Called:
*     iau_PFW06    bias-precession F-W angles, IAU 2006
*     iau_FW2M     F-W angles to r-matrix
*
*  References:
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      SUBROUTINE iau_PMAT76 ( DATE1, DATE2, RMATP )
*+
*  - - - - - - - - - - -
*   i a u _ P M A T 7 6
*  - - - - - - - - - - -
*
*  Precession matrix from J2000.0 to a specified date, IAU 1976 model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       ending date, TT (Note 1)
*
*  Returned:
*     RMATP          d(3,3)  precession matrix, J2000.0 -> DATE1+DATE2
*
*  Notes:
*
*  1) The ending date DATE1+DATE2 is a Julian Date, apportioned
*     in any convenient way between the arguments DATE1 and DATE2.
*     For example, JD(TT)=2450123.7 could be expressed in any of
*     these ways, among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(date) = RMATP * V(J2000),
*     where the p-vector V(J2000) is with respect to the mean
*     equatorial triad of epoch J2000.0 and the p-vector V(date)
*     is with respect to the mean equatorial triad of the given
*     date.
*
*  3) Though the matrix method itself is rigorous, the precession
*     angles are expressed through canonical polynomials which are
*     valid only for a limited time span.  In addition, the IAU 1976
*     precession rate is known to be imperfect.  The absolute accuracy
*     of the present formulation is better than 0.1 arcsec from
*     1960AD to 2040AD, better than 1 arcsec from 1640AD to 2360AD,
*     and remains below 3 arcsec for the whole of the period
*     500BC to 3000AD.  The errors exceed 10 arcsec outside the
*     range 1200BC to 3900AD, exceed 100 arcsec outside 4200BC to
*     5600AD and exceed 1000 arcsec outside 6800BC to 8200AD.
*
*  Called:
*     iau_PREC76   accumulated precession angles, IAU 1976
*     iau_IR       initialize r-matrix to identity
*     iau_RZ       rotate around Z-axis
*     iau_RY       rotate around Y-axis
*     iau_CR       copy r-matrix
*
*  References:
*
*     Lieske, J.H., 1979, Astron.Astrophys. 73, 282.
*      equations (6) & (7), p283.
*
*     Kaplan, G.H., 1981, USNO circular no. 163, pA2.
*
*-
```

```
      SUBROUTINE iau_PMP ( A, B, AMB )
*+
*  - - - - - - - -
*   i a u _ P M P
*  - - - - - - - -
*
*  P-vector subtraction.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3)      first p-vector
*     B        d(3)      second p-vector
*
*  Returned:
*     AMB      d(3)      A - B
*
*-
```

```
      SUBROUTINE iau_PN ( P, R, U )
*+
*  - - - - - - -
*   i a u _ P N
*  - - - - - - -
*
*  Convert a p-vector into modulus and unit vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     P        d(3)      p-vector
*
*  Returned:
*     R        d         modulus
*     U        d(3)      unit vector
*
*  Note:
*     If P is null, the result is null.  Otherwise the result is
*     a unit vector.
*
*  Called:
*     iau_PM       modulus of p-vector
*     iau_ZP       zero p-vector
*     iau_SXP      multiply p-vector by scalar
*
*-
```

```
      SUBROUTINE iau_PN00 ( DATE1, DATE2, DPSI, DEPS,
     :                      EPSA, RB, RP, RBP, RN, RBPN )
*+
*  - - - - - - - - - -
*   i a u _ P N 0 0
*  - - - - - - - - - -
*
*  Precession-nutation, IAU 2000 model:  a multi-purpose routine,
*  supporting classical (equinox-based) use directly and CIO-based
*  use indirectly.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d       TT as a 2-part Julian Date (Note 1)
*     DPSI,DEPS     d       nutation (Note 2)
*
*  Returned:
*     EPSA          d       mean obliquity (Note 3)
*     RB          d(3,3)    frame bias matrix (Note 4)
*     RP          d(3,3)    precession matrix (Note 5)
*     RBP         d(3,3)    bias-precession matrix (Note 6)
*     RN          d(3,3)    nutation matrix (Note 7)
*     RBPN        d(3,3)    GCRS-to-true matrix (Note 8)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The caller is responsible for providing the nutation components;
*     they are in longitude and obliquity, in radians and are with
*     respect to the equinox and ecliptic of date.  For high-accuracy
*     applications, free core nutation should be included as well as
*     any other relevant corrections to the position of the CIP.
*
*  3) The returned mean obliquity is consistent with the IAU 2000
*     precession-nutation models.
*
*  4) The matrix RB transforms vectors from GCRS to J2000.0 mean equator
*     and equinox by applying frame bias.
*
*  5) The matrix RP transforms vectors from J2000.0 mean equator and
*     equinox to mean equator and equinox of date by applying
*     precession.
*
*  6) The matrix RBP transforms vectors from GCRS to mean equator and
*     equinox of date by applying frame bias then precession.  It is the
*     product RP x RB.
*
*  7) The matrix RN transforms vectors from mean equator and equinox of
*     date to true equator and equinox of date by applying the nutation
*     (luni-solar + planetary).
```

```
*
*  8) The matrix RBPN transforms vectors from GCRS to true equator and
*     equinox of date.  It is the product RN x RBP, applying frame bias,
*     precession and nutation in that order.
*
*  Called:
*     iau_PR00     IAU 2000 precession adjustments
*     iau_OBL80    mean obliquity, IAU 1980
*     iau_BP00     frame bias and precession matrices, IAU 2000
*     iau_NUMAT    form nutation matrix
*     iau_RXR      product of two r-matrices
*
*  Reference:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003)
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*-
```

```
      SUBROUTINE iau_PN00A ( DATE1, DATE2,
     :                       DPSI, DEPS, EPSA, RB, RP, RBP, RN, RBPN )
*+
*  - - - - - - - - - -
*   i a u _ P N 0 0 A
*  - - - - - - - - - -
*
*  Precession-nutation, IAU 2000A model:  a multi-purpose routine,
*  supporting classical (equinox-based) use directly and CIO-based
*  use indirectly.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     DPSI,DEPS     d       nutation (Note 2)
*     EPSA          d       mean obliquity (Note 3)
*     RB          d(3,3)    frame bias matrix (Note 4)
*     RP          d(3,3)    precession matrix (Note 5)
*     RBP         d(3,3)    bias-precession matrix (Note 6)
*     RN          d(3,3)    nutation matrix (Note 7)
*     RBPN        d(3,3)    GCRS-to-true matrix (Notes 8,9)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The nutation components (luni-solar + planetary, IAU 2000A) in
*     longitude and obliquity are in radians and with respect to the
*     equinox and ecliptic of date.  Free core nutation is omitted;  for
*     the utmost accuracy, use the iau_PN00 routine, where the nutation
*     components are caller-specified.  For faster but slightly less
*     accurate results, use the iau_PN00B routine.
*
*  3) The mean obliquity is consistent with the IAU 2000 precession.
*
*  4) The matrix RB transforms vectors from GCRS to J2000.0 mean equator
*     and equinox by applying frame bias.
*
*  5) The matrix RP transforms vectors from J2000.0 mean equator and
*     equinox to mean equator and equinox of date by applying
*     precession.
*
*  6) The matrix RBP transforms vectors from GCRS to mean equator and
*     equinox of date by applying frame bias then precession.  It is the
*     product RP x RB.
*
*  7) The matrix RN transforms vectors from mean equator and equinox of
*     date to true equator and equinox of date by applying the nutation
*     (luni-solar + planetary).
```

```
*
*  8) The matrix RBPN transforms vectors from GCRS to true equator and
*     equinox of date.  It is the product RN x RBP, applying frame bias,
*     precession and nutation in that order.
*
*  9) The X,Y,Z coordinates of the IAU 2000A Celestial Intermediate Pole
*     are elements (3,1-3) of the matrix RBPN.
*
*  Called:
*     iau_NUT00A   nutation, IAU 2000A
*     iau_PN00     bias/precession/nutation results, IAU 2000
*
*  Reference:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003).
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*-
```

```
      SUBROUTINE iau_PN00B ( DATE1, DATE2,
     :                       DPSI, DEPS, EPSA, RB, RP, RBP, RN, RBPN )
*+
*  - - - - - - - - - -
*   i a u _ P N 0 0 B
*  - - - - - - - - - -
*
*  Precession-nutation, IAU 2000B model:  a multi-purpose routine,
*  supporting classical (equinox-based) use directly and CIO-based
*  use indirectly.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     DPSI,DEPS     d       nutation (Note 2)
*     EPSA          d       mean obliquity (Note 3)
*     RB          d(3,3)    frame bias matrix (Note 4)
*     RP          d(3,3)    precession matrix (Note 5)
*     RBP         d(3,3)    bias-precession matrix (Note 6)
*     RN          d(3,3)    nutation matrix (Note 7)
*     RBPN        d(3,3)    GCRS-to-true matrix (Notes 8,9)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The nutation components (luni-solar + planetary, IAU 2000B) in
*     longitude and obliquity are in radians and with respect to the
*     equinox and ecliptic of date.  For more accurate results, but
*     at the cost of increased computation, use the iau_PN00A routine.
*     For the utmost accuracy, use the iau_PN00 routine, where the
*     nutation components are caller-specified.
*
*  3) The mean obliquity is consistent with the IAU 2000 precession.
*
*  4) The matrix RB transforms vectors from GCRS to J2000.0 mean equator
*     and equinox by applying frame bias.
*
*  5) The matrix RP transforms vectors from J2000.0 mean equator and
*     equinox to mean equator and equinox of date by applying
*     precession.
*
*  6) The matrix RBP transforms vectors from GCRS to mean equator and
*     equinox of date by applying frame bias then precession.  It is the
*     product RP x RB.
*
*  7) The matrix RN transforms vectors from mean equator and equinox of
*     date to true equator and equinox of date by applying the nutation
*     (luni-solar + planetary).
```

```
*
*  8) The matrix RBPN transforms vectors from GCRS to true equator and
*     equinox of date.  It is the product RN x RBP, applying frame bias,
*     precession and nutation in that order.
*
*  9) The X,Y,Z coordinates of the IAU 2000B Celestial Intermediate Pole
*     are elements (3,1-3) of the matrix RBPN.
*
*  Called:
*     iau_NUT00B   nutation, IAU 2000B
*     iau_PN00     bias/precession/nutation results, IAU 2000
*
*  Reference:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003).
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*-
```

```
      SUBROUTINE iau_PN06 ( DATE1, DATE2, DPSI, DEPS,
     :                      EPSA, RB, RP, RBP, RN, RBPN )
*+
*  - - - - - - - - -
*   i a u _ P N 0 6
*  - - - - - - - - -
*
*  Precession-nutation, IAU 2006 model:  a multi-purpose routine,
*  supporting classical (equinox-based) use directly and CIO-based use
*  indirectly.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d       TT as a 2-part Julian Date (Note 1)
*     DPSI,DEPS     d       nutation (Note 2)
*
*  Returned:
*     EPSA          d       mean obliquity (Note 3)
*     RB          d(3,3)    frame bias matrix (Note 4)
*     RP          d(3,3)    precession matrix (Note 5)
*     RBP         d(3,3)    bias-precession matrix (Note 6)
*     RN          d(3,3)    nutation matrix (Note 7)
*     RBPN        d(3,3)    GCRS-to-true matrix (Note 8)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The caller is responsible for providing the nutation components;
*     they are in longitude and obliquity, in radians and are with
*     respect to the equinox and ecliptic of date.  For high-accuracy
*     applications, free core nutation should be included as well as
*     any other relevant corrections to the position of the CIP.
*
*  3) The returned mean obliquity is consistent with the IAU 2006
*     precession.
*
*  4) The matrix RB transforms vectors from GCRS to mean J2000.0 by
*     applying frame bias.
*
*  5) The matrix RP transforms vectors from mean J2000.0 to mean of date
*     by applying precession.
*
*  6) The matrix RBP transforms vectors from GCRS to mean of date by
*     applying frame bias then precession.  It is the product RP x RB.
*
*  7) The matrix RN transforms vectors from mean of date to true of date
*     by applying the nutation (luni-solar + planetary).
*
*  8) The matrix RBPN transforms vectors from GCRS to true of date
*     (CIP/equinox).  It is the product RN x RBP, applying frame bias,
```

```
*     precession and nutation in that order.
*
*  9) The X,Y,Z coordinates of the IAU 2006/2000A Celestial Intermediate
*     Pole are elements (3,1-3) of the matrix RBPN.
*
*  Called:
*     iau_PFW06    bias-precession F-W angles, IAU 2006
*     iau_FW2M     F-W angles to r-matrix
*     iau_TR       transpose r-matrix
*     iau_RXR      product of two r-matrices
*
*  References:
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      SUBROUTINE iau_PN06A ( DATE1, DATE2,
     :                       DPSI, DEPS, EPSA, RB, RP, RBP, RN, RBPN )
*+
*  - - - - - - - - - - -
*   i a u _ P N 0 6 A
*  - - - - - - - - - - -
*
*  Precession-nutation, IAU 2006/2000A models:  a multi-purpose routine,
*  supporting classical (equinox-based) use directly and CIO-based use
*  indirectly.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     DPSI,DEPS     d       nutation (Note 2)
*     EPSA          d       mean obliquity (Note 3)
*     RB          d(3,3)    frame bias matrix (Note 4)
*     RP          d(3,3)    precession matrix (Note 5)
*     RBP         d(3,3)    bias-precession matrix (Note 6)
*     RN          d(3,3)    nutation matrix (Note 7)
*     RBPN        d(3,3)    GCRS-to-true matrix (Notes 8,9)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The nutation components (luni-solar + planetary, IAU 2000A) in
*     longitude and obliquity are in radians and with respect to the
*     equinox and ecliptic of date.  Free core nutation is omitted;  for
*     the utmost accuracy, use the iau_PN06 routine, where the nutation
*     components are caller-specified.
*
*  3) The mean obliquity is consistent with the IAU 2006 precession.
*
*  4) The matrix RB transforms vectors from GCRS to mean J2000.0 by
*     applying frame bias.
*
*  5) The matrix RP transforms vectors from mean J2000.0 to mean of date
*     by applying precession.
*
*  6) The matrix RBP transforms vectors from GCRS to mean of date by
*     applying frame bias then precession.  It is the product RP x RB.
*
*  7) The matrix RN transforms vectors from mean of date to true of date
*     by applying the nutation (luni-solar + planetary).
*
*  8) The matrix RBPN transforms vectors from GCRS to true of date
*     (CIP/equinox).  It is the product RN x RBP, applying frame bias,
*     precession and nutation in that order.
```

```
*
*  9) The X,Y,Z coordinates of the IAU 2006/2000A Celestial Intermediate
*     Pole are elements (3,1-3) of the matrix RBPN.
*
*  Called:
*     iau_NUT06A   nutation, IAU 2006/2000A
*     iau_PN06     bias/precession/nutation results, IAU 2006
*
*  Reference:
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855
*
*-
```

```
      SUBROUTINE iau_PNM00A ( DATE1, DATE2, RBPN )
*+
*  - - - - - - - - - - -
*   i a u _ P N M 0 0 A
*  - - - - - - - - - - -
*
*  Form the matrix of precession-nutation for a given date (including
*  frame bias), equinox-based, IAU 2000A model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RBPN          d(3,3)    classical NPB matrix (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(date) = RBPN * V(GCRS), where
*     the p-vector V(date) is with respect to the true equatorial triad
*     of date DATE1+DATE2 and the p-vector V(GCRS) is with respect to
*     the Geocentric Celestial Reference System (IAU, 2000).
*
*  3) A faster, but slightly less accurate result (about 1 mas), can be
*     obtained by using instead the iau_PNM00B routine.
*
*  Called:
*     iau_PN00A    bias/precession/nutation, IAU 2000A
*
*  Reference:
*
*     IAU: Trans. International Astronomical Union, Vol. XXIVB;  Proc.
*     24th General Assembly, Manchester, UK.  Resolutions B1.3, B1.6.
*     (2000)
*
*-
```

```
      SUBROUTINE iau_PNM00B ( DATE1, DATE2, RBPN )
*+
*  - - - - - - - - - - -
*   i a u _ P N M 0 0 B
*  - - - - - - - - - - -
*
*  Form the matrix of precession-nutation for a given date (including
*  frame bias), equinox-based, IAU 2000B model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RBPN         d(3,3)    bias-precession-nutation matrix (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(date) = RBPN * V(GCRS), where
*     the p-vector V(date) is with respect to the true equatorial triad
*     of date DATE1+DATE2 and the p-vector V(GCRS) is with respect to
*     the Geocentric Celestial Reference System (IAU, 2000).
*
*  3) The present routine is faster, but slightly less accurate (about
*     1 mas), than the iau_PNM00A routine.
*
*  Called:
*     iau_PN00B    bias/precession/nutation, IAU 2000B
*
*  Reference:
*
*     IAU: Trans. International Astronomical Union, Vol. XXIVB;  Proc.
*     24th General Assembly, Manchester, UK.  Resolutions B1.3, B1.6.
*     (2000)
*
*-
```

```
      SUBROUTINE iau_PNM06A ( DATE1, DATE2, RNPB )
*+
*  - - - - - - - - - - -
*   i a u _ P N M 0 6 A
*  - - - - - - - - - - -
*
*  Form the matrix of precession-nutation for a given date (including
*  frame bias), IAU 2006 precession and IAU 2000A nutation models.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     RNPB         d(3,3)    bias-precession-nutation matrix (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(date) = RNPB * V(GCRS), where
*     the p-vector V(date) is with respect to the true equatorial triad
*     of date DATE1+DATE2 and the p-vector V(GCRS) is with respect to
*     the Geocentric Celestial Reference System (IAU, 2000).
*
*  Called:
*     iau_PFW06    bias-precession F-W angles, IAU 2006
*     iau_NUT06A   nutation, IAU 2006/2000A
*     iau_FW2M     F-W angles to r-matrix
*
*  Reference:
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855.
*
*-
```

```
      SUBROUTINE iau_PNM80 ( DATE1, DATE2, RMATPN )
*+
*  - - - - - - - - - -
*   i a u _ P N M 8 0
*  - - - - - - - - - -
*
*  Form the matrix of precession/nutation for a given date, IAU 1976
*  precession model, IAU 1980 nutation model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2     d         TDB date (Note 1)
*
*  Returned:
*     RMATPN          d(3,3)    combined precession/nutation matrix
*
*  Notes:
*
*  1) The date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TDB)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The matrix operates in the sense V(date) = RMATPN * V(J2000),
*     where the p-vector V(date) is with respect to the true
*     equatorial triad of date DATE1+DATE2 and the p-vector
*     V(J2000) is with respect to the mean equatorial triad of
*     epoch J2000.0.
*
*  Called:
*     iau_PMAT76   precession matrix, IAU 1976
*     iau_NUTM80   nutation matrix, IAU 1980
*     iau_RXR      product of two r-matrices
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992),
*     Section 3.3 (p145).
*
*-
```

```
      SUBROUTINE iau_POM00 ( XP, YP, SP, RPOM )
*+
*  - - - - - - - - - - -
*   i a u _ P O M 0 0
*  - - - - - - - - - - -
*
*  Form the matrix of polar motion for a given date, IAU 2000.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     XP,YP      d      coordinates of the pole (radians, Note 1)
*     SP         d      the TIO locator s' (radians, Note 2)
*
*  Returned:
*     RPOM     d(3,3)   polar-motion matrix (Note 3)
*
*  Notes:
*
*  1) XP and YP are the coordinates (in radians) of the Celestial
*     Intermediate Pole with respect to the International Terrestrial
*     Reference System (see IERS Conventions 2003), measured along the
*     meridians to 0 and 90 deg west respectively.
*
*  2) SP is the TIO locator s', in radians, which positions the
*     Terrestrial Intermediate Origin on the equator.  It is obtained
*     from polar motion observations by numerical integration, and so is
*     in essence unpredictable.  However, it is dominated by a secular
*     drift of about 47 microarcseconds per century, and so can be taken
*     into account by using s' = -47*t, where t is centuries since
*     J2000.0.  The routine iau_SP00 implements this approximation.
*
*  3) The matrix operates in the sense V(TRS) = RPOM * V(CIP), meaning
*     that it is the final rotation when computing the pointing
*     direction to a celestial source.
*
*  Called:
*     iau_IR       initialize r-matrix to identity
*     iau_RZ       rotate around Z-axis
*     iau_RY       rotate around Y-axis
*     iau_RX       rotate around X-axis
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_PPP ( A, B, APB )
*+
*  - - - - - - - -
*   i a u _ P P P
*  - - - - - - - -
*
*  P-vector addition.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3)      first p-vector
*     B        d(3)      second p-vector
*
*  Returned:
*     APB      d(3)      A + B
*
*-
```

```
      SUBROUTINE iau_PPSP ( A, S, B, APSB )
*+
*  - - - - - - - - -
*   i a u _ P P S P
*  - - - - - - - - -
*
*  P-vector plus scaled p-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3)      first p-vector
*     S        d         scalar (multiplier for B)
*     B        d(3)      second p-vector
*
*  Returned:
*     APSB     d(3)      A + S*B
*
*-
```

```
      SUBROUTINE iau_PR00 ( DATE1, DATE2, DPSIPR, DEPSPR )
*+
*  - - - - - - - - -
*   i a u _ P R 0 0
*  - - - - - - - - -
*
*  Precession-rate part of the IAU 2000 precession-nutation models
*  (part of MHB2000).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2    d   TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     DPSIPR,DEPSPR  d   precession corrections (Notes 2,3)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The precession adjustments are expressed as "nutation components",
*     corrections in longitude and obliquity with respect to the J2000.0
*     equinox and ecliptic.
*
*  3) Although the precession adjustments are stated to be with respect
*     to Lieske et al. (1977), the MHB2000 model does not specify which
*     set of Euler angles are to be used and how the adjustments are to
*     be applied.  The most literal and straightforward procedure is to
*     adopt the 4-rotation epsilon_0, psi_A, omega_A, xi_A option, and
*     to add DPSIPR to psi_A and DEPSPR to both omega_A and eps_A.
*
*  4) This is an implementation of one aspect of the IAU 2000A nutation
*     model, formally adopted by the IAU General Assembly in 2000,
*     namely MHB2000 (Mathews et al. 2002).
*
*  References:
*
*     Lieske, J.H., Lederle, T., Fricke, W. & Morando, B., "Expressions
*     for the precession quantities based upon the IAU (1976) System of
*     Astronomical Constants", Astron.Astrophys., 58, 1-16 (1977)
*
*     Mathews, P.M., Herring, T.A., Buffet, B.A., "Modeling of nutation
*     and precession   New nutation series for nonrigid Earth and
*     insights into the Earth's interior", J.Geophys.Res., 107, B4,
*     2002.  The MHB2000 code itself was obtained on 9th September 2002
*     from ftp://maia.usno.navy.mil/conv2000/chapter5/IAU2000A.
*
*     Wallace, P.T., "Software for Implementing the IAU 2000
*     Resolutions", in IERS Workshop 5.1 (2002).
*
*-
```

```
      SUBROUTINE iau_PREC76 ( EP01, EP02, EP11, EP12, ZETA, Z, THETA )
*+
*  - - - - - - - - - - -
*   i a u _ P R E C 7 6
*  - - - - - - - - - - -
*
*  IAU 1976 precession model.
*
*  This routine forms the three Euler angles which implement general
*  precession between two epochs, using the IAU 1976 model (as for
*  the FK5 catalog).
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     EP01,EP02   d      TDB starting epoch (Note 1)
*     EP11,EP12   d      TDB ending epoch (Note 1)
*
*  Returned:
*     ZETA        d      1st rotation: radians clockwise around z
*     Z           d      3rd rotation: radians clockwise around z
*     THETA       d      2nd rotation: radians counterclockwise around y
*
*  Notes:
*
*  1) The epochs EP01+EP02 and EP11+EP12 are Julian Dates, apportioned
*     in any convenient way between the arguments EPn1 and EPn2.  For
*     example, JD(TDB)=2450123.7 could be expressed in any of these
*     ways, among others:
*
*             EPn1          EPn2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in cases
*     where the loss of several decimal digits of resolution is
*     acceptable.  The J2000 method is best matched to the way the
*     argument is handled internally and will deliver the optimum
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*     The two epochs may be expressed using different methods, but at
*     the risk of losing some resolution.
*
*  2) The accumulated precession angles zeta, z, theta are expressed
*     through canonical polynomials which are valid only for a limited
*     time span.  In addition, the IAU 1976 precession rate is known to
*     be imperfect.  The absolute accuracy of the present formulation is
*     better than 0.1 arcsec from 1960AD to 2040AD, better than 1 arcsec
*     from 1640AD to 2360AD, and remains below 3 arcsec for the whole of
*     the period 500BC to 3000AD.  The errors exceed 10 arcsec outside
*     the range 1200BC to 3900AD, exceed 100 arcsec outside 4200BC to
*     5600AD and exceed 1000 arcsec outside 6800BC to 8200AD.
*
*  3) The three angles are returned in the conventional order, which
*     is not the same as the order of the corresponding Euler rotations.
*     The precession matrix is R_3(-z) x R_2(+theta) x R_3(-zeta).
*
*  Reference:
*
*     Lieske, J.H., 1979, Astron.Astrophys. 73, 282.
*      equations (6) & (7), p283.
*
*-
```

```
      SUBROUTINE iau_PV2P ( PV, P )
*+
*  - - - - - - - - - -
*   i a u _ P V 2 P
*  - - - - - - - - - -
*
*  Discard velocity component of a pv-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     PV       d(3,2)      pv-vector
*
*  Returned:
*     P        d(3)        p-vector
*
*  Called:
*     iau_CP       copy p-vector
*
*-
```

```
      SUBROUTINE iau_PV2S ( PV, THETA, PHI, R, TD, PD, RD )
*+
*  - - - - - - - - -
*   i a u _ P V 2 S
*  - - - - - - - - -
*
*  Convert position/velocity from Cartesian to spherical coordinates.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     PV       d(3,2)    pv-vector
*
*  Returned:
*     THETA    d         longitude angle (radians)
*     PHI      d         latitude angle (radians)
*     R        d         radial distance
*     TD       d         rate of change of THETA
*     PD       d         rate of change of PHI
*     RD       d         rate of change of R
*
*  Notes:
*
*  1) If the position part of PV is null, THETA, PHI, TD and PD
*     are indeterminate.  This is handled by extrapolating the
*     position through unit time by using the velocity part of
*     PV.  This moves the origin without changing the direction
*     of the velocity component.  If the position and velocity
*     components of PV are both null, zeroes are returned for all
*     six results.
*
*  2) If the position is a pole, THETA, TD and PD are indeterminate.
*     In such cases zeroes are returned for all three.
*
*-
```

```
      SUBROUTINE iau_PVDPV ( A, B, ADB )
*+
*  - - - - - - - - - -
*   i a u _ P V D P V
*  - - - - - - - - - -
*
*  Inner (=scalar=dot) product of two pv-vectors.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3,2)      first pv-vector
*     B        d(3,2)      second pv-vector
*
*  Returned:
*     ADB      d(2)        A . B (see note)
*
*  Note:
*
*     If the position and velocity components of the two pv-vectors are
*     ( Ap, Av ) and ( Bp, Bv ), the result, A . B, is the pair of
*     numbers ( Ap . Bp , Ap . Bv + Av . Bp ).  The two numbers are the
*     dot-product of the two p-vectors and its derivative.
*
*  Called:
*     iau_PDP      scalar product of two p-vectors
*
*-
```

```
      SUBROUTINE iau_PVM ( PV, R, S )
*+
*  - - - - - - - -
*   i a u _ P V M
*  - - - - - - - -
*
*  Modulus of pv-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     PV       d(3,2)    pv-vector
*
*  Returned:
*     R        d         modulus of position component
*     S        d         modulus of velocity component
*
*  Called:
*     iau_PM       modulus of p-vector
*
*-
```

```
      SUBROUTINE iau_PVMPV ( A, B, AMB )
*+
*  - - - - - - - - - -
*   i a u _ P V M P V
*  - - - - - - - - - -
*
*  Subtract one pv-vector from another.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3,2)      first pv-vector
*     B        d(3,2)      second pv-vector
*
*  Returned:
*     AMB      d(3,2)      A - B
*
*  Called:
*     iau_PMP      p-vector minus p-vector
*
*-
```

```
      SUBROUTINE iau_PVPPV ( A, B, APB )
*+
*  - - - - - - - - - -
*   i a u _ P V P P V
*  - - - - - - - - - -
*
*  Add one pv-vector to another.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3,2)      first pv-vector
*     B        d(3,2)      second pv-vector
*
*  Returned:
*     APB      d(3,2)      A + B
*
*  Called:
*     iau_PPP      p-vector plus p-vector
*
*-
```

```
      SUBROUTINE iau_PVSTAR ( PV, RA, DEC, PMR, PMD, PX, RV, J )
*+
*  - - - - - - - - - - -
*   i a u _ P V S T A R
*  - - - - - - - - - - -
*
*  Convert star position+velocity vector to catalog coordinates.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given (Note 1):
*     PV       d(3,2)    pv-vector (AU, AU/day)
*
*  Returned (Note 2):
*     RA       d         right ascension (radians)
*     DEC      d         declination (radians)
*     PMR      d         RA proper motion (radians/year)
*     PMD      d         Dec proper motion (radians/year)
*     PX       d         parallax (arcsec)
*     RV       d         radial velocity (km/s, positive = receding)
*     J        i         status:
*                           0 = OK
*                          -1 = superluminal speed (Note 5)
*                          -2 = null position vector
*
*  Notes:
*
*  1) The specified pv-vector is the coordinate direction (and its rate
*     of change) for the epoch at which the light leaving the star
*     reached the solar-system barycenter.
*
*  2) The star data returned by this routine are "observables" for an
*     imaginary observer at the solar-system barycenter.  Proper motion
*     and radial velocity are, strictly, in terms of barycentric
*     coordinate time, TCB.  For most practical applications, it is
*     permissible to neglect the distinction between TCB and ordinary
*     "proper" time on Earth (TT/TAI).  The result will, as a rule, be
*     limited by the intrinsic accuracy of the proper-motion and radial-
*     velocity data;  moreover, the supplied pv-vector is likely to be
*     merely an intermediate result (for example generated by the
*     routine iau_STARPV), so that a change of time unit will cancel
*     out overall.
*
*     In accordance with normal star-catalog conventions, the object's
*     right ascension and declination are freed from the effects of
*     secular aberration.  The frame, which is aligned to the catalog
*     equator and equinox, is Lorentzian and centered on the SSB.
*
*     Summarizing, the specified pv-vector is for most stars almost
*     identical to the result of applying the standard geometrical
*     "space motion" transformation to the catalog data.  The
*     differences, which are the subject of the Stumpff paper cited
*     below, are:
*
*     (i) In stars with significant radial velocity and proper motion,
*     the constantly changing light-time distorts the apparent proper
*     motion.  Note that this is a classical, not a relativistic,
*     effect.
*
*     (ii) The transformation complies with special relativity.
*
*  3) Care is needed with units.  The star coordinates are in radians
*     and the proper motions in radians per Julian year, but the
*     parallax is in arcseconds; the radial velocity is in km/s, but
*     the pv-vector result is in AU and AU/day.
*
*  4) The proper motions are the rate of change of the right ascension
*     and declination at the catalog epoch and are in radians per Julian
*     year.  The RA proper motion is in terms of coordinate angle, not
```

```
*     true angle, and will thus be numerically larger at high
*     declinations.
*
*  5) Straight-line motion at constant speed in the inertial frame is
*     assumed.  If the speed is greater than or equal to the speed of
*     light, the routine aborts with an error status.
*
*  6) The inverse transformation is performed by the routine iau_STARPV.
*
*  Called:
*     iau_PN       decompose p-vector into modulus and direction
*     iau_PDP      scalar product of two p-vectors
*     iau_SXP      multiply p-vector by scalar
*     iau_PMP      p-vector minus p-vector
*     iau_PM       modulus of p-vector
*     iau_PPP      p-vector plus p-vector
*     iau_PV2S     pv-vector to spherical
*     iau_ANP      normalize angle into range 0 to 2pi
*
*  Reference:
*
*     Stumpff, P., Astron.Astrophys. 144, 232-240 (1985).
*
*-
```

```
      SUBROUTINE iau_PVU ( DT, PV, UPV )
*+
*  - - - - - - - -
*   i a u _ P V U
*  - - - - - - - -
*
*  Update a pv-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     DT       d            time interval
*     PV       d(3,2)       pv-vector
*
*  Returned:
*     UPV      d(3,2)       p updated, v unchanged
*
*  Notes:
*
*  1) "Update" means "refer the position component of the vector
*     to a new epoch DT time units from the existing epoch".
*
*  2) The time units of DT must match those of the velocity.
*
*  Called:
*     iau_PPSP     p-vector plus scaled p-vector
*     iau_CP       copy p-vector
*
*-
```

```
      SUBROUTINE iau_PVUP ( DT, PV, P )
*+
*  - - - - - - - - - -
*   i a u _ P V U P
*  - - - - - - - - - -
*
*  Update a pv-vector, discarding the velocity component.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     DT       d           time interval
*     PV       d(3,2)      pv-vector
*
*  Returned:
*     P        d(3)        p-vector
*
*  Notes:
*
*  1) "Update" means "refer the position component of the vector to a
*     new date DT time units from the existing date".
*
*  2) The time units of DT must match those of the velocity.
*
*-
```

```
      SUBROUTINE iau_PVXPV ( A, B, AXB )
*+
*  - - - - - - - - - -
*   i a u _ P V X P V
*  - - - - - - - - - -
*
*  Outer (=vector=cross) product of two pv-vectors.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3,2)      first pv-vector
*     B        d(3,2)      second pv-vector
*
*  Returned:
*     AXB      d(3,2)      A x B
*
*  Note:
*
*     If the position and velocity components of the two pv-vectors are
*     ( Ap, Av ) and ( Bp, Bv ), the result, A x B, is the pair of
*     vectors ( Ap x Bp, Ap x Bv + Av x Bp ).  The two vectors are the
*     cross-product of the two p-vectors and its derivative.
*
*  Called:
*     iau_CPV      copy pv-vector
*     iau_PXP      vector product of two p-vectors
*     iau_PPP      p-vector plus p-vector
*
*-
```

```
      SUBROUTINE iau_PXP ( A, B, AXB )
*+
*  - - - - - - - -
*   i a u _ P X P
*  - - - - - - - -
*
*  p-vector outer (=vector=cross) product.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3)      first p-vector
*     B        d(3)      second p-vector
*
*  Returned:
*     AXB      d(3)      A x B
*
*-
```

```
      SUBROUTINE iau_RM2V ( R, W )
*+
*  - - - - - - - - -
*   i a u _ R M 2 V
*  - - - - - - - - -
*
*  Express an r-matrix as an r-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     R        d(3,3)    rotation matrix
*
*  Returned:
*     W        d(3)      rotation vector (Note 1)
*
*  Notes:
*
*  1) A rotation matrix describes a rotation through some angle about
*     some arbitrary axis called the Euler axis.  The "rotation vector"
*     returned by this routine has the same direction as the Euler axis,
*     and its magnitude is the angle in radians.  (The magnitude and
*     direction can be separated by means of the routine iau_PN.)
*
*  2) If R is null, so is the result.  If R is not a rotation matrix
*     the result is undefined.  R must be proper (i.e. have a positive
*     determinant) and real orthogonal (inverse = transpose).
*
*  3) The reference frame rotates clockwise as seen looking along
*     the rotation vector from the origin.
*
*-
```

```
      SUBROUTINE iau_RV2M ( W, R )
*+
*  - - - - - - - - -
*   i a u _ R V 2 M
*  - - - - - - - - -
*
*  Form the r-matrix corresponding to a given r-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     W        d(3)      rotation vector (Note 1)
*
*  Returned:
*     R        d(3,3)    rotation matrix
*
*  Notes:
*
*  1) A rotation matrix describes a rotation through some angle about
*     some arbitrary axis called the Euler axis.  The "rotation vector"
*     supplied to this routine has the same direction as the Euler axis,
*     and its magnitude is the angle in radians.
*
*  2) If W is null, the unit matrix is returned.
*
*  3) The reference frame rotates clockwise as seen looking along the
*     rotation vector from the origin.
*
*-
```

```
      SUBROUTINE iau_RX ( PHI, R )
*+
*  - - - - - - -
*   i a u _ R X
*  - - - - - - -
*
*  Rotate an r-matrix about the x-axis.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     PHI      d         angle (radians)
*
*  Given and returned:
*     R        d(3,3)    r-matrix, rotated
*
*  Notes:
*
*  1) Calling this routine with positive PHI incorporates in the
*     supplied r-matrix R an additional rotation, about the x-axis,
*     anticlockwise as seen looking towards the origin from positive x.
*
*  2) The additional rotation can be represented by this matrix:
*
*         (  1        0            0      )
*         (                               )
*         (  0   + cos(PHI)   + sin(PHI)  )
*         (                               )
*         (  0   - sin(PHI)   + cos(PHI)  )
*
*-
```

```
      SUBROUTINE iau_RXP ( R, P, RP )
*+
*  - - - - - - - -
*   i a u _ R X P
*  - - - - - - - -
*
*  Multiply a p-vector by an r-matrix.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     R        d(3,3)    r-matrix
*     P        d(3)      p-vector
*
*  Returned:
*     RP       d(3)      R * P
*
*  Called:
*     iau_CP       copy p-vector
*
*-
```

```
      SUBROUTINE iau_RXPV ( R, PV, RPV )
*+
*  - - - - - - - - -
*   i a u _ R X P V
*  - - - - - - - - -
*
*  Multiply a pv-vector by an r-matrix.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     R        d(3,3)    r-matrix
*     PV       d(3,2)    pv-vector
*
*  Returned:
*     RPV      d(3,2)    R * PV
*
*  Called:
*     iau_RXP      product of r-matrix and p-vector
*
*-
```

```
      SUBROUTINE iau_RXR ( A, B, ATB )
*+
*  - - - - - - - - 
*   i a u _ R X R
*  - - - - - - - -
*
*  Multiply two r-matrices.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3,3)    first r-matrix
*     B        d(3,3)    second r-matrix
*
*  Returned:
*     ATB      d(3,3)    A * B
*
*  Called:
*     iau_CR       copy r-matrix
*
*-
```

```
      SUBROUTINE iau_RY ( THETA, R )
*+
*  - - - - - - - -
*   i a u _ R Y
*  - - - - - - - -
*
*  Rotate an r-matrix about the y-axis.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     THETA    d         angle (radians)
*
*  Given and returned:
*     R        d(3,3)    r-matrix, rotated
*
*  Notes:
*
*  1) Calling this routine with positive THETA incorporates in the
*     supplied r-matrix R an additional rotation, about the y-axis,
*     anticlockwise as seen looking towards the origin from positive y.
*
*  2) The additional rotation can be represented by this matrix:
*
*         (  + cos(THETA)     0      - sin(THETA)  )
*         (                                        )
*         (       0           1           0        )
*         (                                        )
*         (  + sin(THETA)     0      + cos(THETA)  )
*
*-
```

```
      SUBROUTINE iau_RZ ( PSI, R )
*+
*  - - - - - - -
*   i a u _ R Z
*  - - - - - - -
*
*  Rotate an r-matrix about the z-axis.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     PSI      d         angle (radians)
*
*  Given and returned:
*     R        d(3,3)    r-matrix, rotated
*
*  Notes:
*
*  1) Calling this routine with positive PSI incorporates in the
*     supplied r-matrix R an additional rotation, about the z-axis,
*     anticlockwise as seen looking towards the origin from positive z.
*
*  2) The additional rotation can be represented by this matrix:
*
*         (  + cos(PSI)   + sin(PSI)     0  )
*         (                                 )
*         (  - sin(PSI)   + cos(PSI)     0  )
*         (                                 )
*         (       0            0         1  )
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_S00 ( DATE1, DATE2, X, Y )
*+
*  - - - - - - - -
*   i a u _ S 0 0
*  - - - - - - - -
*
*  The CIO locator s, positioning the Celestial Intermediate Origin on
*  the equator of the Celestial Intermediate Pole, given the CIP's X,Y
*  coordinates.  Compatible with IAU 2000A precession-nutation.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*     X,Y            d      CIP coordinates (Note 3)
*
*  Returned:
*     iau_S00        d      the CIO locator s in radians (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The CIO locator s is the difference between the right ascensions
*     of the same point in two systems:  the two systems are the GCRS
*     and the CIP,CIO, and the point is the ascending node of the
*     CIP equator.  The quantity s remains below 0.1 arcsecond
*     throughout 1900-2100.
*
*  3) The series used to compute s is in fact for s+XY/2, where X and Y
*     are the x and y components of the CIP unit vector;  this series is
*     more compact than a direct series for s would be.  This routine
*     requires X,Y to be supplied by the caller, who is responsible for
*     providing values that are consistent with the supplied date.
*
*  4) The model is consistent with the IAU 2000A precession-nutation.
*
*  Called:
*     iau_FAL03    mean anomaly of the Moon
*     iau_FALP03   mean anomaly of the Sun
*     iau_FAF03    mean argument of the latitude of the Moon
*     iau_FAD03    mean elongation of the Moon from the Sun
*     iau_FAOM03   mean longitude of the Moon's ascending node
*     iau_FAVE03   mean longitude of Venus
*     iau_FAE03    mean longitude of Earth
*     iau_FAPA03   general accumulated precession in longitude
*
*  References:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
```

```
*     model", Astron.Astrophys. 400, 1145-1154 (2003)
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_S00A ( DATE1, DATE2 )
*+
*  - - - - - - - - -
*   i a u _ S 0 0 A
*  - - - - - - - - -
*
*  The CIO locator s, positioning the Celestial Intermediate Origin on
*  the equator of the Celestial Intermediate Pole, using the IAU 2000A
*  precession-nutation model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_S00A       d      the CIO locator s in radians (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The CIO locator s is the difference between the right ascensions
*     of the same point in two systems.  The two systems are the GCRS
*     and the CIP,CIO, and the point is the ascending node of the
*     CIP equator.  The CIO locator s remains a small fraction of
*     1 arcsecond throughout 1900-2100.
*
*  3) The series used to compute s is in fact for s+XY/2, where X and Y
*     are the x and y components of the CIP unit vector;  this series is
*     more compact than a direct series for s would be.  The present
*     routine uses the full IAU 2000A nutation model when predicting the
*     CIP position.  Faster results, with no significant loss of
*     accuracy, can be obtained via the routine iau_S00B, which uses
*     instead the IAU 2000B truncated model.
*
*  Called:
*     iau_PNM00A   classical NPB matrix, IAU 2000A
*     iau_BNP2XY    extract CIP X,Y from the BPN matrix
*     iau_S00      the CIO locator s, given X,Y, IAU 2000A
*
*  References:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003)
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
```

```
*      IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_S00B ( DATE1, DATE2 )
*+
*  - - - - - - - - -
*   i a u _ S 0 0 B
*  - - - - - - - - -
*
*  The CIO locator s, positioning the Celestial Intermediate Origin on
*  the equator of the Celestial Intermediate Pole, using the IAU 2000B
*  precession-nutation model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_S00B       d      the CIO locator s in radians (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The CIO locator s is the difference between the right ascensions
*     of the same point in two systems.  The two systems are the GCRS
*     and the CIP,CIO, and the point is the ascending node of the
*     CIP equator.  The CIO locator s remains a small fraction of
*     1 arcsecond throughout 1900-2100.
*
*  3) The series used to compute s is in fact for s+XY/2, where X and Y
*     are the x and y components of the CIP unit vector;  this series is
*     more compact than a direct series for s would be.  The present
*     routine uses the IAU 2000B truncated nutation model when
*     predicting the CIP position.  The routine iau_S00A uses instead
*     the full IAU 2000A model, but with no significant increase in
*     accuracy and at some cost in speed.
*
*  Called:
*     iau_PNM00B   classical NPB matrix, IAU 2000B
*     iau_BNP2XY    extract CIP X,Y from the BPN matrix
*     iau_S00      the CIO locator s, given X,Y, IAU 2000A
*
*  References:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003)
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
```

```
*      IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_S06 ( DATE1, DATE2, X, Y )
*+
*  - - - - - - - -
*   i a u _ S 0 6
*  - - - - - - - -
*
*  The CIO locator s, positioning the Celestial Intermediate Origin on
*  the equator of the Celestial Intermediate Pole, given the CIP's X,Y
*  coordinates.  Compatible with IAU 2006/2000A precession-nutation.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*     X,Y            d      CIP coordinates (Note 3)
*
*  Returned:
*     iau_S06        d      the CIO locator s in radians (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The CIO locator s is the difference between the right ascensions
*     of the same point in two systems:  the two systems are the GCRS
*     and the CIP,CIO, and the point is the ascending node of the
*     CIP equator.  The quantity s remains below 0.1 arcsecond
*     throughout 1900-2100.
*
*  3) The series used to compute s is in fact for s+XY/2, where X and Y
*     are the x and y components of the CIP unit vector;  this series is
*     more compact than a direct series for s would be.  This routine
*     requires X,Y to be supplied by the caller, who is responsible for
*     providing values that are consistent with the supplied date.
*
*  4) The model is consistent with the "P03" precession (Capitaine et
*     al. 2003), adopted by IAU 2006 Resolution 1, 2006, and the
*     IAU 2000A nutation (with P03 adjustments).
*
*  Called:
*     iau_FAL03    mean anomaly of the Moon
*     iau_FALP03   mean anomaly of the Sun
*     iau_FAF03    mean argument of the latitude of the Moon
*     iau_FAD03    mean elongation of the Moon from the Sun
*     iau_FAOM03   mean longitude of the Moon's ascending node
*     iau_FAVE03   mean longitude of Venus
*     iau_FAE03    mean longitude of Earth
*     iau_FAPA03   general accumulated precession in longitude
*
*  References:
*
*     Capitaine, N., Wallace, P.T. & Chapront, J., 2003, Astron.
```

```
*     Astrophys. 432, 355
*
*     McCarthy, D.D., Petit, G. (eds.) 2004, IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_S06A ( DATE1, DATE2 )
*+
*  - - - - - - - - -
*   i a u _ S 0 6 A
*  - - - - - - - - -
*
*  The CIO locator s, positioning the Celestial Intermediate Origin on
*  the equator of the Celestial Intermediate Pole, using the IAU 2006
*  precession and IAU 2000A nutation models.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_S06A       d      the CIO locator s in radians (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The CIO locator s is the difference between the right ascensions
*     of the same point in two systems.  The two systems are the GCRS
*     and the CIP,CIO, and the point is the ascending node of the
*     CIP equator.  The CIO locator s remains a small fraction of
*     1 arcsecond throughout 1900-2100.
*
*  3) The series used to compute s is in fact for s+XY/2, where X and Y
*     are the x and y components of the CIP unit vector;  this series is
*     more compact than a direct series for s would be.  The present
*     routine uses the full IAU 2000A nutation model when predicting the
*     CIP position.
*
*  Called:
*     iau_PNM06A   classical NPB matrix, IAU 2006/2000A
*     iau_BPN2XY   extract CIP X,Y coordinates from NPB matrix
*     iau_S06      the CIO locator s, given X,Y, IAU 2006
*
*  References:
*
*     Capitaine, N., Chapront, J., Lambert, S. and Wallace, P.,
*     "Expressions for the Celestial Intermediate Pole and Celestial
*     Ephemeris Origin consistent with the IAU 2000A precession-nutation
*     model", Astron.Astrophys. 400, 1145-1154 (2003)
*
*     n.b. The celestial ephemeris origin (CEO) was renamed "celestial
*          intermediate origin" (CIO) by IAU 2006 Resolution 2.
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855
*
*     McCarthy, D. D., Petit, G. (eds.), 2004, IERS Conventions (2003),
```

```
*      IERS Technical Note No. 32, BKG
*
*      Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      SUBROUTINE iau_S2C ( THETA, PHI, C )
*+
*  - - - - - - - -
*   i a u _ S 2 C
*  - - - - - - - -
*
*  Convert spherical coordinates to Cartesian.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     THETA    d         longitude angle (radians)
*     PHI      d         latitude angle (radians)
*
*  Returned:
*     C        d(3)      direction cosines
*
*-
```

```
      SUBROUTINE iau_S2P ( THETA, PHI, R, P )
*+
*  - - - - - - - - 
*   i a u _ S 2 P
*  - - - - - - - - 
*
*  Convert spherical polar coordinates to p-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     THETA    d         longitude angle (radians)
*     PHI      d         latitude angle (radians)
*     R        d         radial distance
*
*  Returned:
*     P        d(3)      Cartesian coordinates
*
*  Called:
*     iau_S2C      spherical coordinates to unit vector
*     iau_SXP      multiply p-vector by scalar
*
*-
```

```
      SUBROUTINE iau_S2PV ( THETA, PHI, R, TD, PD, RD, PV )
*+
*  - - - - - - - - - 
*   i a u _ S 2 P V
*  - - - - - - - - - 
*
*  Convert position/velocity from spherical to Cartesian coordinates.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     THETA    d         longitude angle (radians)
*     PHI      d         latitude angle (radians)
*     R        d         radial distance
*     TD       d         rate of change of THETA
*     PD       d         rate of change of PHI
*     RD       d         rate of change of R
*
*  Returned:
*     PV       d(3,2)    pv-vector
*
*-
```

```
      SUBROUTINE iau_S2XPV ( S1, S2, PV, SPV )
*+
*  - - - - - - - - - -
*   i a u _ S 2 X P V
*  - - - - - - - - - -
*
*  Multiply a pv-vector by two scalars.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     S1       d         scalar to multiply position component by
*     S2       d         scalar to multiply velocity component by
*     PV       d(3,2)    pv-vector
*
*  Returned:
*     SPV      d(3,2)    pv-vector: p scaled by S1, v scaled by S2
*
*  Called:
*     iau_SXP      multiply p-vector by scalar
*
*-
```

```
      SUBROUTINE iau_SEPP ( A, B, S )
*+
*  - - - - - - - - - -
*   i a u _ S E P P
*  - - - - - - - - - -
*
*  Angular separation between two p-vectors.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     A        d(3)      first p-vector (not necessarily unit length)
*     B        d(3)      second p-vector (not necessarily unit length)
*
*  Returned:
*     S        d         angular separation (radians, always positive)
*
*  Notes:
*
*  1) If either vector is null, a zero result is returned.
*
*  2) The angular separation is most simply formulated in terms of
*     scalar product.  However, this gives poor accuracy for angles
*     near zero and pi.  The present algorithm uses both cross product
*     and dot product, to deliver full accuracy whatever the size of
*     the angle.
*
*  Called:
*     iau_PXP      vector product of two p-vectors
*     iau_PM       modulus of p-vector
*     iau_PDP      scalar product of two p-vectors
*
*-
```

```
      SUBROUTINE iau_SEPS ( AL, AP, BL, BP, S )
*+
*  - - - - - - - - - -
*   i a u _ S E P S
*  - - - - - - - - - -
*
*  Angular separation between two sets of spherical coordinates.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     AL       d         first longitude (radians)
*     AP       d         first latitude (radians)
*     BL       d         second longitude (radians)
*     BP       d         second latitude (radians)
*
*  Returned:
*     S        d         angular separation (radians)
*
*  Called:
*     iau_S2C      spherical coordinates to unit vector
*     iau_SEPP     angular separation between two p-vectors
*
*-
```

```
      DOUBLE PRECISION FUNCTION iau_SP00 ( DATE1, DATE2 )
*+
*  - - - - - - - - -
*   i a u _ S P 0 0
*  - - - - - - - - -
*
*  The TIO locator s', positioning the Terrestrial Intermediate Origin
*  on the equator of the Celestial Intermediate Pole.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2    d      TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     iau_SP00       d      the TIO locator s' in radians (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The TIO locator s' is obtained from polar motion observations by
*     numerical integration, and so is in essence unpredictable.
*     However, it is dominated by a secular drift of about
*     47 microarcseconds per century, which is the approximation
*     evaluated by the present routine.
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_STARPM ( RA1, DEC1, PMR1, PMD1, PX1, RV1,
     :                        EP1A, EP1B, EP2A, EP2B,
     :                        RA2, DEC2, PMR2, PMD2, PX2, RV2, J )
*+
*  - - - - - - - - - - -
*   i a u _ S T A R P M
*  - - - - - - - - - - -
*
*  Star proper motion:  update star catalog data for space motion.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     RA1      d         right ascension (radians), before
*     DEC1     d         declination (radians), before
*     PMR1     d         RA proper motion (radians/year), before
*     PMD1     d         Dec proper motion (radians/year), before
*     PX1      d         parallax (arcseconds), before
*     RV1      d         radial velocity (km/s, +ve = receding), before
*     EP1A     d         "before" epoch, part A (Note 1)
*     EP1B     d         "before" epoch, part B (Note 1)
*     EP2A     d         "after" epoch, part A (Note 1)
*     EP2B     d         "after" epoch, part B (Note 1)
*
*  Returned:
*     RA2      d         right ascension (radians), after
*     DEC2     d         declination (radians), after
*     PMR2     d         RA proper motion (radians/year), after
*     PMD2     d         Dec proper motion (radians/year), after
*     PX2      d         parallax (arcseconds), after
*     RV2      d         radial velocity (km/s, +ve = receding), after
*     J        i         status:
*                           -1 = system error (should not occur)
*                            0 = no warnings or errors
*                            1 = distance overridden (Note 6)
*                            2 = excessive velocity (Note 7)
*                            4 = solution didn't converge (Note 8)
*                        else = binary logical OR of the above warnings
*
*  Notes:
*
*  1) The starting and ending TDB epochs EP1A+EP1B and EP2A+EP2B are
*     Julian Dates, apportioned in any convenient way between the two
*     parts (A and B).  For example, JD(TDB)=2450123.7 could be
*     expressed in any of these ways, among others:
*
*             EPnA          EPnB
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) In accordance with normal star-catalog conventions, the object's
*     right ascension and declination are freed from the effects of
*     secular aberration.  The frame, which is aligned to the catalog
*     equator and equinox, is Lorentzian and centered on the SSB.
*
*     The proper motions are the rate of change of the right ascension
*     and declination at the catalog epoch and are in radians per TDB
*     Julian year.
*
```

```
*     The parallax and radial velocity are in the same frame.
*
*  3) Care is needed with units.  The star coordinates are in radians
*     and the proper motions in radians per Julian year, but the
*     parallax is in arcseconds.
*
*  4) The RA proper motion is in terms of coordinate angle, not true
*     angle.  If the catalog uses arcseconds for both RA and Dec proper
*     motions, the RA proper motion will need to be divided by cos(Dec)
*     before use.
*
*  5) Straight-line motion at constant speed, in the inertial frame,
*     is assumed.
*
*  6) An extremely small (or zero or negative) parallax is interpreted
*     to mean that the object is on the "celestial sphere", the radius
*     of which is an arbitrary (large) value (see the iau_STARPV routine
*     for the value used).  When the distance is overridden in this way,
*     the status, initially zero, has 1 added to it.
*
*  7) If the space velocity is a significant fraction of c (see the
*     constant VMAX in the routine iau_STARPV), it is arbitrarily set to
*     zero.  When this action occurs, 2 is added to the status.
*
*  8) The relativistic adjustment carried out in the iau_STARPV routine
*     involves an iterative calculation.  If the process fails to
*     converge within a set number of iterations, 4 is added to the
*     status.
*
*  Called:
*     iau_STARPV   star catalog data to space motion pv-vector
*     iau_PVU      update a pv-vector
*     iau_PDP      scalar product of two p-vectors
*     iau_PVSTAR   space motion pv-vector to star catalog data
*
*-
```

```
      SUBROUTINE iau_STARPV ( RA, DEC, PMR, PMD, PX, RV, PV, J )
*+
*  - - - - - - - - - - -
*   i a u _ S T A R P V
*  - - - - - - - - - - -
*
*  Convert star catalog coordinates to position+velocity vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given (Note 1):
*     RA       d         right ascension (radians)
*     DEC      d         declination (radians)
*     PMR      d         RA proper motion (radians/year)
*     PMD      d         Dec proper motion (radians/year)
*     PX       d         parallax (arcseconds)
*     RV       d         radial velocity (km/s, positive = receding)
*
*  Returned (Note 2):
*     PV       d(3,2)    pv-vector (AU, AU/day)
*     J        i         status:
*                           0 = no warnings
*                           1 = distance overridden (Note 6)
*                           2 = excessive velocity (Note 7)
*                           4 = solution didn't converge (Note 8)
*                        else = binary logical OR of the above
*
*  Notes:
*
*  1) The star data accepted by this routine are "observables" for an
*     imaginary observer at the solar-system barycenter.  Proper motion
*     and radial velocity are, strictly, in terms of barycentric
*     coordinate time, TCB.  For most practical applications, it is
*     permissible to neglect the distinction between TCB and ordinary
*     "proper" time on Earth (TT/TAI).  The result will, as a rule, be
*     limited by the intrinsic accuracy of the proper-motion and radial-
*     velocity data;  moreover, the pv-vector is likely to be merely an
*     intermediate result, so that a change of time unit would cancel
*     out overall.
*
*     In accordance with normal star-catalog conventions, the object's
*     right ascension and declination are freed from the effects of
*     secular aberration.  The frame, which is aligned to the catalog
*     equator and equinox, is Lorentzian and centered on the SSB.
*
*  2) The resulting position and velocity pv-vector is with respect to
*     the same frame and, like the catalog coordinates, is freed from
*     the effects of secular aberration.  Should the "coordinate
*     direction", where the object was located at the catalog epoch, be
*     required, it may be obtained by calculating the magnitude of the
*     position vector PV(1-3,1) dividing by the speed of light in AU/day
*     to give the light-time, and then multiplying the space velocity
*     PV(1-3,2) by this light-time and adding the result to PV(1-3,1).
*
*     Summarizing, the pv-vector returned is for most stars almost
*     identical to the result of applying the standard geometrical
*     "space motion" transformation.  The differences, which are the
*     subject of the Stumpff paper referenced below, are:
*
*     (i) In stars with significant radial velocity and proper motion,
*     the constantly changing light-time distorts the apparent proper
*     motion.  Note that this is a classical, not a relativistic,
*     effect.
*
*     (ii) The transformation complies with special relativity.
*
*  3) Care is needed with units.  The star coordinates are in radians
*     and the proper motions in radians per Julian year, but the
*     parallax is in arcseconds; the radial velocity is in km/s, but
```

```
*     the pv-vector result is in AU and AU/day.
*
*  4) The RA proper motion is in terms of coordinate angle, not true
*     angle.  If the catalog uses arcseconds for both RA and Dec proper
*     motions, the RA proper motion will need to be divided by cos(Dec)
*     before use.
*
*  5) Straight-line motion at constant speed, in the inertial frame,
*     is assumed.
*
*  6) An extremely small (or zero or negative) parallax is interpreted
*     to mean that the object is on the "celestial sphere", the radius
*     of which is an arbitrary (large) value (see the constant PXMIN).
*     When the distance is overridden in this way, the status, initially
*     zero, has 1 added to it.
*
*  7) If the space velocity is a significant fraction of c (see the
*     constant VMAX), it is arbitrarily set to zero.  When this action
*     occurs, 2 is added to the status.
*
*  8) The relativistic adjustment involves an iterative calculation.
*     If the process fails to converge within a set number (IMAX) of
*     iterations, 4 is added to the status.
*
*  9) The inverse transformation is performed by the routine iau_PVSTAR.
*
*  Called:
*     iau_S2PV     spherical coordinates to pv-vector
*     iau_PM       modulus of p-vector
*     iau_ZP       zero p-vector
*     iau_PN       decompose p-vector into modulus and direction
*     iau_PDP      scalar product of two p-vectors
*     iau_SXP      multiply p-vector by scalar
*     iau_PMP      p-vector minus p-vector
*     iau_PPP      p-vector plus p-vector
*
*  Reference:
*
*     Stumpff, P., Astron.Astrophys. 144, 232-240 (1985).
*
*-
```

```
      SUBROUTINE iau_SXP ( S, P, SP )
*+
*  - - - - - - - -
*   i a u _ S X P
*  - - - - - - - -
*
*  Multiply a p-vector by a scalar.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     S        d          scalar
*     P        d(3)       p-vector
*
*  Returned:
*     SP       d(3)       S * P
*
*-
```

```
      SUBROUTINE iau_SXPV ( S, PV, SPV )
*+
*  - - - - - - - - -
*   i a u _ S X P V
*  - - - - - - - - -
*
*  Multiply a pv-vector by a scalar.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     S        d         scalar
*     PV       d(3,2)    pv-vector
*
*  Returned:
*     SPV      d(3,2)    S * PV
*
*  Called:
*     iau_S2XPV    multiply pv-vector by two scalars
*
*-
```

```
      SUBROUTINE iau_TAITT ( TAI1, TAI2, TT1, TT2, J )
*+
*  - - - - - - - - - - -
*   i a u _ T A I T T
*  - - - - - - - - - - -
*
*  Time scale transformation:  International Atomic Time, TAI, to
*  Terrestrial Time, TT.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TAI1,TAI2    d      TAI as a 2-part Julian Date
*
*  Returned:
*     TT1,TT2      d      TT as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Note:
*
*     TAI1+TAI2 is Julian Date, apportioned in any convenient way
*     between the two arguments, for example where TAI1 is the Julian
*     Day Number and TAI2 is the fraction of a day.  The returned
*     TT1,TT2 follow suit.
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*-
```

```
      SUBROUTINE iau_TAIUT1 ( TAI1, TAI2, DTA, UT11, UT12, J )
*+
*  - - - - - - - - - - - 
*   i a u _ T A I U T 1
*  - - - - - - - - - - - 
*
*  Time scale transformation:  International Atomic Time, TAI, to
*  Universal Time, UT1.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TAI1,TAI2    d      TAI as a 2-part Julian Date
*     DTA          d      UT1-TAI in seconds
*
*  Returned:
*     UT11,UT12    d      UT1 as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Notes:
*
*  1) TAI1+TAI2 is Julian Date, apportioned in any convenient way
*     between the two arguments, for example where TAI1 is the Julian
*     Day Number and TAI2 is the fraction of a day.  The returned
*     UT11,UT12 follow suit.
*
*  2) The argument DTA, i.e. UT1-TAI, is an observed quantity, and is
*     available from IERS tabulations.
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*-
```

```
      SUBROUTINE iau_TAIUTC ( TAI1, TAI2, UTC1, UTC2, J )
*+
*  - - - - - - - - - - -
*   i a u _ T A I U T C
*  - - - - - - - - - - -
*
*  Time scale transformation:  International Atomic Time, TAI, to
*  Coordinated Universal Time, UTC.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TAI1,TAI2    d      TAI as a 2-part Julian Date (Note 1)
*
*  Returned:
*     UTC1,UTC2    d      UTC as a 2-part quasi Julian Date (Notes 1-3)
*     J            i      status: +1 = dubious year (Note 4)
*                                  0 = OK
*                                 -1 = unacceptable date
*
*  Notes:
*
*  1) TAI1+TAI2 is Julian Date, apportioned in any convenient way
*     between the two arguments, for example where TAI1 is the Julian
*     Day Number and TAI2 is the fraction of a day.  The returned UTC1
*     and UTC2 form an analogous pair, except that a special convention
*     is used, to deal with the problem of leap seconds - see the next
*     note.
*
*  2) JD cannot unambiguously represent UTC during a leap second unless
*     special measures are taken.  The convention in the present routine
*     is that the JD day represents UTC days whether the length is
*     86399, 86400 or 86401 SI seconds.
*
*  3) The routine iau_D2DTF can be used to transform the UTC quasi-JD
*     into calendar date and clock time, including UTC leap second
*     handling.
*
*  4) The warning status "dubious year" flags UTCs that predate the
*     introduction of the time scale and that are too far in the future
*     to be trusted.  See iau_DAT for further details.
*
*  Called:
*     iau_JD2CAL   JD to Gregorian calendar
*     iau_DAT      delta(AT) = TAI-UTC
*     iau_CAL2JD   Gregorian calendar to JD
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*-
```

```
      SUBROUTINE iau_TCBTDB ( TCB1, TCB2, TDB1, TDB2, J )
*+
*  - - - - - - - - - - -
*   i a u _ T C B T D B
*  - - - - - - - - - - -
*
*  Time scale transformation:  Barycentric Coordinate Time, TCB, to
*  Barycentric Dynamical Time, TDB.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TCB1,TCB2    d      TCB as a 2-part Julian Date
*
*  Returned:
*     TDB1,TDB2    d      TDB as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Notes:
*
*  1) TCB1+TCB2 is Julian Date, apportioned in any convenient way
*     between the two arguments, for example where TCB1 is the Julian
*     Day Number and TCB2 is the fraction of a day.  The returned
*     TDB1,TDB2 follow suit.
*
*  2) The 2006 IAU General Assembly introduced a conventional linear
*     transformation between TDB and TCB.  This transformation
*     compensates for the drift between TCB and terrestrial time TT,
*     and keeps TDB approximately centered on TT.  Because the
*     relationship between TT and TCB depends on the adopted solar
*     system ephemeris, the degree of alignment between TDB and TT over
*     long intervals will vary according to which ephemeris is used.
*     Former definitions of TDB attempted to avoid this problem by
*     stipulating that TDB and TT should differ only by periodic
*     effects.  This is a good description of the nature of the
*     relationship but eluded precise mathematical formulation.  The
*     conventional linear relationship adopted in 2006 sidestepped
*     these difficulties whilst delivering a TDB that in practice was
*     consistent with values before that date.
*
*  3) TDB is essentially the same as Teph, the time argument for the
*     JPL solar system ephemerides.
*
*  Reference:
*
*     IAU 2006 Resolution B3
*
*-
```

```
      SUBROUTINE iau_TCGTT ( TCG1, TCG2, TT1, TT2, J )
*+
*  - - - - - - - - - -
*   i a u _ T C G T T
*  - - - - - - - - - -
*
*  Time scale transformation:  Geocentric Coordinate Time, TCG, to
*  Terrestrial Time, TT.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TCG1,TCG2    d      TCG as a 2-part Julian Date
*
*  Returned:
*     TT1,TT2      d      TT as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Note:
*
*     TCG1+TCG2 is Julian Date, apportioned in any convenient way
*     between the two arguments, for example where TCG1 is the Julian
*     Day Number and TCG2 is the fraction of a day.  The returned
*     TT1,TT2 follow suit.
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),.
*     IERS Technical Note No. 32, BKG (2004)
*
*     IAU 2000 Resolution B1.9
*
*-
```

```
      SUBROUTINE iau_TDBTCB ( TDB1, TDB2, TCB1, TCB2, J )
*+
*  - - - - - - - - - - -
*   i a u _ T D B T C B
*  - - - - - - - - - - -
*
*  Time scale transformation:  Barycentric Dynamical Time, TDB, to
*  Barycentric Coordinate Time, TCB.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TDB1,TDB2    d      TDB as a 2-part Julian Date
*
*  Returned:
*     TCB1,TCB2    d      TCB as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Notes:
*
*  1) TDB1+TDB2 is Julian Date, apportioned in any convenient way
*     between the two arguments, for example where TDB1 is the Julian
*     Day Number and TDB2 is the fraction of a day.  The returned
*     TCB1,TCB2 follow suit.
*
*  2) The 2006 IAU General Assembly introduced a conventional linear
*     transformation between TDB and TCB.  This transformation
*     compensates for the drift between TCB and terrestrial time TT,
*     and keeps TDB approximately centered on TT.  Because the
*     relationship between TT and TCB depends on the adopted solar
*     system ephemeris, the degree of alignment between TDB and TT over
*     long intervals will vary according to which ephemeris is used.
*     Former definitions of TDB attempted to avoid this problem by
*     stipulating that TDB and TT should differ only by periodic
*     effects.  This is a good description of the nature of the
*     relationship but eluded precise mathematical formulation.  The
*     conventional linear relationship adopted in 2006 sidestepped
*     these difficulties whilst delivering a TDB that in practice was
*     consistent with values before that date.
*
*  3) TDB is essentially the same as Teph, the time argument for the
*     JPL solar system ephemerides.
*
*  Reference:
*
*     IAU 2006 Resolution B3
*
*-
```

```
      SUBROUTINE iau_TDBTT ( TDB1, TDB2, DTR, TT1, TT2, J )
*+
*  - - - - - - - - - - 
*   i a u _ T D B T T
*  - - - - - - - - - - 
*
*  Time scale transformation:  Barycentric Dynamical Time, TDB, to
*  Terrestrial Time, TT.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TDB1,TDB2    d      TDB as a 2-part Julian Date
*     DTR          d      TDB-TT in seconds
*
*  Returned:
*     TT1,TT2      d      TT as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Notes:
*
*  1) TDB1+TDB2 is Julian Date, apportioned in any convenient way
*     between the two arguments, for example where TDB1 is the Julian
*     Day Number and TDB2 is the fraction of a day.  The returned
*     TT1,TT2 follow suit.
*
*  2) The argument DTR represents the quasi-periodic component of the
*     GR transformation between TT and TCB.  It is dependent upon the
*     adopted solar-system ephemeris, and can be obtained by numerical
*     integration, by interrogating a precomputed time ephemeris or by
*     evaluating a model such as that implemented in the SOFA routine
*     iau_DTDB.  The quantity is dominated by an annual term of 1.7 ms
*     amplitude.
*
*  3) TDB is essentially the same as Teph, the time argument for the
*     JPL solar system ephemerides.
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     IAU 2006 Resolution 3
*
*-
```

```
      SUBROUTINE iau_TF2A ( S, IHOUR, IMIN, SEC, RAD, J )
*+
*  - - - - - - - - -
*   i a u _ T F 2 A
*  - - - - - - - - -
*
*  Convert hours, minutes, seconds to radians.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     S          c      sign:  '-' = negative, otherwise positive
*     IHOUR      i      hours
*     IMIN       i      minutes
*     SEC        d      seconds
*
*  Returned:
*     RAD        d      angle in radians
*     J          i      status:  0 = OK
*                                1 = IHOUR outside range 0-23
*                                2 = IMIN outside range 0-59
*                                3 = SEC outside range 0-59.999...
*
*  Notes:
*
*  1)  If the s argument is a string, only the leftmost character is
*      used and no warning status is provided.
*
*  2)  The result is computed even if any of the range checks fail.
*
*  3)  Negative IHOUR, IMIN and/or SEC produce a warning status, but the
*      absolute value is used in the conversion.
*
*  4)  If there are multiple errors, the status value reflects only the
*      first, the smallest taking precedence.
*
*-
```

```
      SUBROUTINE iau_TF2D ( S, IHOUR, IMIN, SEC, DAYS, J )
*+
*  - - - - - - - - - 
*   i a u _ T F 2 D
*  - - - - - - - - - 
*
*  Convert hours, minutes, seconds to days.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     S         c     sign:  '-' = negative, otherwise positive
*     IHOUR     i     hours
*     IMIN      i     minutes
*     SEC       d     seconds
*
*  Returned:
*     DAYS      d     interval in days
*     J         i     status:  0 = OK
*                              1 = IHOUR outside range 0-23
*                              2 = IMIN outside range 0-59
*                              3 = SEC outside range 0-59.999...
*
*  Notes:
*
*  1)  If the s argument is a string, only the leftmost character is
*      used and no warning status is provided.
*
*  2)  The result is computed even if any of the range checks fail.
*
*  3)  Negative IHOUR, IMIN and/or SEC produce a warning status, but the
*      absolute value is used in the conversion.
*
*  4)  If there are multiple errors, the status value reflects only the
*      first, the smallest taking precedence.
*
*-
```

```
      SUBROUTINE iau_TR ( R, RT )
*+
*  - - - - - - - -
*   i a u _ T R
*  - - - - - - - -
*
*  Transpose an r-matrix.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     R        d(3,3)    r-matrix
*
*  Returned:
*     RT       d(3,3)    transpose
*
*  Called:
*     iau_CR       copy r-matrix
*
*-
```

```
      SUBROUTINE iau_TRXP ( R, P, TRP )
*+
*  - - - - - - - - -
*   i a u _ T R X P
*  - - - - - - - - -
*
*  Multiply a p-vector by the transpose of an r-matrix.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     R        d(3,3)    r-matrix
*     P        d(3)      p-vector
*
*  Returned:
*     TRP      d(3)      R * P
*
*  Called:
*     iau_TR       transpose r-matrix
*     iau_RXP      product of r-matrix and p-vector
*
*-
```

```
      SUBROUTINE iau_TRXPV ( R, PV, TRPV )
*+
*  - - - - - - - - - -
*   i a u _ T R X P V
*  - - - - - - - - - -
*
*  Multiply a pv-vector by the transpose of an r-matrix.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Given:
*     R        d(3,3)    r-matrix
*     PV       d(3,2)    pv-vector
*
*  Returned:
*     TRPV     d(3,2)    R * PV
*
*  Called:
*     iau_TR       transpose r-matrix
*     iau_RXPV     product of r-matrix and pv-vector
*
*-
```

```
      SUBROUTINE iau_TTTAI ( TT1, TT2, TAI1, TAI2, J )
*+
*  - - - - - - - - - -
*   i a u _ T T T A I
*  - - - - - - - - - -
*
*  Time scale transformation:  Terrestrial Time, TT, to International
*  Atomic Time, TAI.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TT1,TT2      d      TT as a 2-part Julian Date
*
*  Returned:
*     TAI1,TAI2    d      TAI as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Note:
*
*     TT1+TT2 is Julian Date, apportioned in any convenient way between
*     the two arguments, for example where TT1 is the Julian Day Number
*     and TT2 is the fraction of a day.  The returned TAI1,TAI2 follow
*     suit.
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*-
```

```
      SUBROUTINE iau_TTTCG ( TT1, TT2, TCG1, TCG2, J )
*+
*  - - - - - - - - - -
*   i a u _ T T T C G
*  - - - - - - - - - -
*
*  Time scale transformation:  Terrestrial Time, TT, to Geocentric
*  Coordinate Time, TCG.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TT1,TT2      d      TT as a 2-part Julian Date
*
*  Returned:
*     TCG1,TCG2    d      TCG as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Note:
*
*     TT1+TT2 is Julian Date, apportioned in any convenient way between
*     the two arguments, for example where TT1 is the Julian Day Number
*     and TT2 is the fraction of a day.  The returned TCG1,TCG2 follow
*     suit.
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     IAU 2000 Resolution B1.9
*
*-
```

```
      SUBROUTINE iau_TTTDB ( TT1, TT2, DTR, TDB1, TDB2, J )
*+
*  - - - - - - - - - -
*   i a u _ T T T D B
*  - - - - - - - - - -
*
*  Time scale transformation:  Terrestrial Time, TT, to Barycentric
*  Dynamical Time, TDB.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TT1,TT2      d      TT as a 2-part Julian Date
*     DTR          d      TDB-TT in seconds
*
*  Returned:
*     TDB1,TDB2    d      TDB as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Notes:
*
*  1) TT1+TT2 is Julian Date, apportioned in any convenient way between
*     the two arguments, for example where TT1 is the Julian Day Number
*     and TT2 is the fraction of a day.  The returned TDB1,TDB2 follow
*     suit.
*
*  2) The argument DTR represents the quasi-periodic component of the
*     GR transformation between TT and TCB.  It is dependent upon the
*     adopted solar-system ephemeris, and can be obtained by numerical
*     integration, by interrogating a precomputed time ephemeris or by
*     evaluating a model such as that implemented in the SOFA routine
*     iau_DTDB.  The quantity is dominated by an annual term of 1.7 ms
*     amplitude.
*
*  3) TDB is essentially the same as Teph, the time argument for the JPL
*     solar system ephemerides.
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     IAU 2006 Resolution 3
*
*-
```

```
      SUBROUTINE iau_TTUT1 ( TT1, TT2, DT, UT11, UT12, J )
*+
*  - - - - - - - - - - 
*   i a u _ T T U T 1
*  - - - - - - - - - - 
*
*  Time scale transformation:  Terrestrial Time, TT, to Universal Time,
*  UT1.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     TT1,TT2      d      TT as a 2-part Julian Date
*     DT           d      TT-UT1 in seconds
*
*  Returned:
*     UT11,UT12    d      UT1 as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Notes:
*
*  1) TT1+TT2 is Julian Date, apportioned in any convenient way between
*     the two arguments, for example where TT1 is the Julian Day Number
*     and TT2 is the fraction of a day.  The returned UT11,UT12 follow
*     suit.
*
*  2) The argument DT is classical Delta T.
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*-
```

```
      SUBROUTINE iau_UT1TAI ( UT11, UT12, DTA, TAI1, TAI2, J )
*+
*  - - - - - - - - - - -
*   i a u _ U T 1 T A I
*  - - - - - - - - - - -
*
*  Time scale transformation:  Universal Time, UT1, to International
*  Atomic Time, TAI.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     UT11,UT12    d      UT1 as a 2-part Julian Date
*     DTA          d      UT1-TAI in seconds
*
*  Returned:
*     TAI1,TAI2    d      TAI as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Notes:
*
*  1) UT11+UT12 is Julian Date, apportioned in any convenient way
*     between the two arguments, for example where UT11 is the Julian
*     Day Number and UT12 is the fraction of a day.  The returned
*     TAI1,TAI2 follow suit.
*
*  2) The argument DTA, i.e. UT1-TAI, is an observed quantity, and is
*     available from IERS tabulations.
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*-
```

```
      SUBROUTINE iau_UT1TT ( UT11, UT12, DT, TT1, TT2, J )
*+
*  - - - - - - - - - -
*   i a u _ U T 1 T T
*  - - - - - - - - - -
*
*  Time scale transformation:  Universal Time, UT1, to Terrestrial Time,
*  TT.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     UT11,UT12    d      UT1 as a 2-part Julian Date
*     DT           d      TT-UT1 in seconds
*
*  Returned:
*     TT1,TT2      d      TT as a 2-part Julian Date
*     J            i      status:  0 = OK
*
*  Notes:
*
*  1) UT11+UT12 is Julian Date, apportioned in any convenient way
*     between the two arguments, for example where UT11 is the Julian
*     Day Number and UT12 is the fraction of a day.  The returned
*     TT1,TT2 follow suit.
*
*  2) The argument DT is classical Delta T.
*
*  Reference:
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*-
```

```
      SUBROUTINE iau_UT1UTC ( UT11, UT12, DUT1, UTC1, UTC2, J )
*+
*  - - - - - - - - - - -
*   i a u _ U T 1 U T C
*  - - - - - - - - - - -
*
*  Time scale transformation:  Universal Time, UT1, to Coordinated
*  Universal Time, UTC.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     UT11,UT12    d      UT1 as a 2-part Julian Date (Note 1)
*     DUT1         d      Delta UT1: UT1-UTC in seconds (Note 2)
*
*  Returned:
*     UTC1,UTC2    d      UTC as a 2-part quasi Julian Date (Notes 3,4)
*     J            i      status: +1 = dubious year (Note 5)
*                                  0 = OK
*                                 -1 = unacceptable date
*
*  Notes:
*
*  1) UT11+UT12 is Julian Date, apportioned in any convenient way
*     between the two arguments, for example where UT11 is the Julian
*     Day Number and UT12 is the fraction of a day.  The returned UTC1
*     and UTC2 form an analogous pair, except that a special convention
*     is used, to deal with the problem of leap seconds - see Note 3.
*
*  2) Delta UT1 can be obtained from tabulations provided by the
*     International Earth Rotation and Reference Systems Service.  The
*     value changes abruptly by 1s at a leap second;  however, close to
*     a leap second the algorithm used here is tolerant of the "wrong"
*     choice of value being made.
*
*  3) JD cannot unambiguously represent UTC during a leap second unless
*     special measures are taken.  The convention in the present routine
*     is that the returned quasi JD day UTC1+UTC2 represents UTC days
*     whether the length is 86399, 86400 or 86401 SI seconds.
*
*  4) The routine iau_D2DTF can be used to transform the UTC quasi-JD
*     into calendar date and clock time, including UTC leap second
*     handling.
*
*  5) The warning status "dubious year" flags UTCs that predate the
*     introduction of the time scale and that are too far in the future
*     to be trusted.  See iau_DAT for further details.
*
*  Called:
*     iau_JD2CAL   JD to Gregorian calendar
*     iau_DAT      delta(AT) = TAI-UTC
*     iau_CAL2JD   Gregorian calendar to JD
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*-
```

```
      SUBROUTINE iau_UTCTAI ( UTC1, UTC2, TAI1, TAI2, J )
*+
*  - - - - - - - - - - -
*   i a u _ U T C T A I
*  - - - - - - - - - - -
*
*  Time scale transformation:  Coordinated Universal Time, UTC, to
*  International Atomic Time, TAI.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     UTC1,UTC2    d      UTC as a 2-part quasi Julian Date (Notes 1-4)
*
*  Returned:
*     TAI1,TAI2    d      TAI as a 2-part Julian Date (Note 5)
*     J            i      status: +1 = dubious year (Note 3)
*                                  0 = OK
*                                 -1 = unacceptable date
*
*  Notes:
*
*  1) UTC1+UTC2 is quasi Julian Date (see Note 2), apportioned in any
*     convenient way between the two arguments, for example where UTC1
*     is the Julian Day Number and UTC2 is the fraction of a day.
*
*  2) JD cannot unambiguously represent UTC during a leap second unless
*     special measures are taken.  The convention in the present routine
*     is that the JD day represents UTC days whether the length is
*     86399, 86400 or 86401 SI seconds.
*
*  3) The warning status "dubious year" flags UTCs that predate the
*     introduction of the time scale and that are too far in the future
*     to be trusted.  See iau_DAT for further details.
*
*  4) The routine iau_DTF2D converts from calendar date and time of day
*     into 2-part Julian Date, and in the case of UTC implements the
*     leap-second-ambiguity convention described above.
*
*  5) The returned TAI1,TAI2 are such that their sum is the TAI Julian
*     Date.
*
*  Called:
*     iau_JD2CAL   JD to Gregorian calendar
*     iau_DAT      delta(AT) = TAI-UTC
*     iau_CAL2JD   Gregorian calendar to JD
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*-
```

```
      SUBROUTINE iau_UTCUT1 ( UTC1, UTC2, DUT1, UT11, UT12, J )
*+
*  - - - - - - - - - - -
*   i a u _ U T C U T 1
*  - - - - - - - - - - -
*
*  Time scale transformation:  Coordinated Universal Time, UTC, to
*  Universal Time, UT1.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical.
*
*  Given:
*     UTC1,UTC2    d      UTC as a 2-part quasi Julian Date (Notes 1-4)
*     DUT1         d      Delta UT1 = UT1-UTC in seconds (Note 5)
*
*  Returned:
*     UT11,UT12    d      UT1 as a 2-part Julian Date (Note 6)
*     J            i      status: +1 = dubious year (Note 7)
*                                  0 = OK
*                                 -1 = unacceptable date
*
*  Notes:
*
*  1) UTC1+UTC2 is quasi Julian Date (see Note 2), apportioned in any
*     convenient way between the two arguments, for example where UTC1
*     is the Julian Day Number and UTC2 is the fraction of a day.
*
*  2) JD cannot unambiguously represent UTC during a leap second unless
*     special measures are taken.  The convention in the present routine
*     is that the JD day represents UTC days whether the length is
*     86399, 86400 or 86401 SI seconds.
*
*  3) The warning status "dubious year" flags UTCs that predate the
*     introduction of the time scale and that are too far in the future
*     to be trusted.  See iau_DAT for further details.
*
*  4) The routine iau_DTF2D converts from calendar date and time of day
*     into 2-part Julian Date, and in the case of UTC implements the
*     leap-second-ambiguity convention described above.
*
*  5) Delta UT1 can be obtained from tabulations provided by the
*     International Earth Rotation and Reference Systems Service.  It
*     It is the caller's responsibility to supply a DUT argument
*     containing the UT1-UTC value that matches the given UTC.
*
*  6) The returned UT11,UT12 are such that their sum is the UT1 Julian
*     Date.
*
*  7) The warning status "dubious year" flags UTCs that predate the
*     introduction of the time scale and that are too far in the future
*     to be trusted.  See iau_DAT for further details.
*
*  References:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*     Explanatory Supplement to the Astronomical Almanac,
*     P. Kenneth Seidelmann (ed), University Science Books (1992)
*
*  Called:
*     iau_JD2CAL   JD to Gregorian calendar
*     iau_DAT      delta(AT) = TAI-UTC
*     iau_UTCTAI   UTC to TAI
*     iau_TAIUT1   TAI to UT1
*
*-
```

```
      SUBROUTINE iau_XY06 ( DATE1, DATE2, X, Y )
*+
*  - - - - - - - - -
*   i a u _ X Y 0 6
*  - - - - - - - - -
*
*  X,Y coordinates of celestial intermediate pole from series based
*  on IAU 2006 precession and IAU 2000A nutation.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  canonical model.
*
*  Given:
*     DATE1,DATE2    d       TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     X,Y            d       CIP X,Y coordinates (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The X,Y coordinates are those of the unit vector towards the
*     celestial intermediate pole.  They represent the combined effects
*     of frame bias, precession and nutation.
*
*  3) The fundamental arguments used are as adopted in IERS Conventions
*     (2003) and are from Simon et al. (1994) and Souchay et al. (1999).
*
*  4) This is an alternative to the angles-based method, via the SOFA
*     routine iau_FW2XY and as used in iau_XYS06A for example.  The
*     two methods agree at the 1 microarcsecond level (at present),
*     a negligible amount compared with the intrinsic accuracy of the
*     models.  However, it would be unwise to mix the two methods
*     (angles-based and series-based) in a single application.
*
*  Called:
*     iau_FAL03    mean anomaly of the Moon
*     iau_FALP03   mean anomaly of the Sun
*     iau_FAF03    mean argument of the latitude of the Moon
*     iau_FAD03    mean elongation of the Moon from the Sun
*     iau_FAOM03   mean longitude of the Moon's ascending node
*     iau_FAME03   mean longitude of Mercury
*     iau_FAVE03   mean longitude of Venus
*     iau_FAE03    mean longitude of Earth
*     iau_FAMA03   mean longitude of Mars
*     iau_FAJU03   mean longitude of Jupiter
*     iau_FASA03   mean longitude of Saturn
*     iau_FAUR03   mean longitude of Uranus
*     iau_FANE03   mean longitude of Neptune
*     iau_FAPA03   general accumulated precession in longitude
*
*  References:
```

```
*
*     Capitaine, N., Wallace, P.T. & Chapront, J., 2003,
*     Astron.Astrophys., 412, 567
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855
*
*     McCarthy, D. D., Petit, G. (eds.), 2004, IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG
*
*     Simon, J.L., Bretagnon, P., Chapront, J., Chapront-Touze, M.,
*     Francou, G. & Laskar, J., Astron.Astrophys., 1994, 282, 663
*
*     Souchay, J., Loysel, B., Kinoshita, H., Folgueira, M., 1999,
*     Astron.Astrophys.Supp.Ser. 135, 111
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      SUBROUTINE iau_XYS00A ( DATE1, DATE2, X, Y, S )
*+
*  - - - - - - - - - - -
*   i a u _ X Y S 0 0 A
*  - - - - - - - - - - -
*
*  For a given TT date, compute the X,Y coordinates of the Celestial
*  Intermediate Pole and the CIO locator s, using the IAU 2000A
*  precession-nutation model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d    TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     X,Y           d    Celestial Intermediate Pole (Note 2)
*     S             d    the CIO locator s (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The Celestial Intermediate Pole coordinates are the x,y components
*     of the unit vector in the Geocentric Celestial Reference System.
*
*  3) The CIO locator s (in radians) positions the Celestial
*     Intermediate Origin on the equator of the CIP.
*
*  4) A faster, but slightly less accurate result (about 1 mas for X,Y),
*     can be obtained by using instead the iau_XYS00B routine.
*
*  Called:
*     iau_PNM00A   classical NPB matrix, IAU 2000A
*     iau_BPN2XY   extract CIP X,Y coordinates from NPB matrix
*     iau_S00      the CIO locator s, given X,Y, IAU 2000A
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_XYS00B ( DATE1, DATE2, X, Y, S )
*+
*  - - - - - - - - - - -
*   i a u _ X Y S 0 0 B
*  - - - - - - - - - - -
*
*  For a given TT date, compute the X,Y coordinates of the Celestial
*  Intermediate Pole and the CIO locator s, using the IAU 2000B
*  precession-nutation model.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d    TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     X,Y           d    Celestial Intermediate Pole (Note 2)
*     S             d    the CIO locator s (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The Celestial Intermediate Pole coordinates are the x,y components
*     of the unit vector in the Geocentric Celestial Reference System.
*
*  3) The CIO locator s (in radians) positions the Celestial
*     Intermediate Origin on the equator of the CIP.
*
*  4) The present routine is faster, but slightly less accurate (about
*     1 mas in X,Y), than the iau_XYS00A routine.
*
*  Called:
*     iau_PNM00B   classical NPB matrix, IAU 2000B
*     iau_BPN2XY   extract CIP X,Y coordinates from NPB matrix
*     iau_S00      the CIO locator s, given X,Y, IAU 2000A
*
*  Reference:
*
*     McCarthy, D. D., Petit, G. (eds.), IERS Conventions (2003),
*     IERS Technical Note No. 32, BKG (2004)
*
*-
```

```
      SUBROUTINE iau_XYS06A ( DATE1, DATE2, X, Y, S )
*+
*  - - - - - - - - - - -
*   i a u _ X Y S 0 6 A
*  - - - - - - - - - - -
*
*  For a given TT date, compute the X,Y coordinates of the Celestial
*  Intermediate Pole and the CIO locator s, using the IAU 2006
*  precession and IAU 2000A nutation models.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  support routine.
*
*  Given:
*     DATE1,DATE2   d    TT as a 2-part Julian Date (Note 1)
*
*  Returned:
*     X,Y           d    Celestial Intermediate Pole (Note 2)
*     S             d    the CIO locator s (Note 2)
*
*  Notes:
*
*  1) The TT date DATE1+DATE2 is a Julian Date, apportioned in any
*     convenient way between the two arguments.  For example,
*     JD(TT)=2450123.7 could be expressed in any of these ways,
*     among others:
*
*            DATE1          DATE2
*
*         2450123.7D0        0D0        (JD method)
*          2451545D0      -1421.3D0     (J2000 method)
*         2400000.5D0     50123.2D0     (MJD method)
*         2450123.5D0       0.2D0       (date & time method)
*
*     The JD method is the most natural and convenient to use in
*     cases where the loss of several decimal digits of resolution
*     is acceptable.  The J2000 method is best matched to the way
*     the argument is handled internally and will deliver the
*     optimum resolution.  The MJD method and the date & time methods
*     are both good compromises between resolution and convenience.
*
*  2) The Celestial Intermediate Pole coordinates are the x,y components
*     of the unit vector in the Geocentric Celestial Reference System.
*
*  3) The CIO locator s (in radians) positions the Celestial
*     Intermediate Origin on the equator of the CIP.
*
*  4) Series-based solutions for generating X and Y are also available:
*     see Capitaine & Wallace (2006) and iau_XY06.
*
*  Called:
*     iau_PNM06A   classical NPB matrix, IAU 2006/2000A
*     iau_BPN2XY   extract CIP X,Y coordinates from NPB matrix
*     iau_S06      the CIO locator s, given X,Y, IAU 2006
*
*  References:
*
*     Capitaine, N. & Wallace, P.T., 2006, Astron.Astrophys. 450, 855
*
*     Wallace, P.T. & Capitaine, N., 2006, Astron.Astrophys. 459, 981
*
*-
```

```
      SUBROUTINE iau_ZP ( P )
*+
*  - - - - - - - -
*   i a u _ Z P
*  - - - - - - - -
*
*  Zero a p-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Returned:
*     P        d(3)      p-vector
*
*-
```

```
      SUBROUTINE iau_ZPV ( PV )
*+
*  - - - - - - - -
*   i a u _ Z P V
*  - - - - - - - -
*
*  Zero a pv-vector.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Returned:
*     PV       d(3,2)      pv-vector
*
*  Called:
*     iau_ZP       zero p-vector
*
*-
```

```
      SUBROUTINE iau_ZR ( R )
*+
*  - - - - - - -
*   i a u _ Z R
*  - - - - - - -
*
*  Initialize an r-matrix to the null matrix.
*
*  This routine is part of the International Astronomical Union's
*  SOFA (Standards of Fundamental Astronomy) software collection.
*
*  Status:  vector/matrix support routine.
*
*  Returned:
*     R        d(3,3)    r-matrix
*
*-
```

COPYRIGHT NOTICE

Text equivalent to the following appears at the end of every SOFA routine. (There are small formatting differences between the Fortran and C versions.)

```
*  5. The SOFA software is provided "as is" and SOFA makes no warranty
*     as to its use or performance.   SOFA does not and cannot warrant
*     the performance or results which the user may obtain by using the
*     SOFA software.  SOFA makes no warranties, express or implied, as
*     to non-infringement of third party rights, merchantability, or
*     fitness for any particular purpose.  In no event will SOFA be
*     liable to the user for any consequential, incidental, or special
*     damages, including any lost profits or lost savings, even if a
*     SOFA representative has been advised of such damages, or for any
*     claim by any third party.
*
*  6. The provision of any version of the SOFA software under the terms
*     and conditions specified herein does not imply that future
*     versions will also be made available under the same terms and
*     conditions.
*
*  In any published work or commercial product which uses the SOFA
*  software directly, acknowledgement (see www.iausofa.org) is
*  appreciated.
*
*  Correspondence concerning SOFA software should be addressed as
*  follows:
*
*      By email:  sofa@ukho.gov.uk
*      By post:   IAU SOFA Center
*                 HM Nautical Almanac Office
*                 UK Hydrographic Office
*                 Admiralty Way, Taunton
*                 Somerset, TA1 2DN
*                 United Kingdom
*
*-----------------------------------------------------------------------
```

SOFA Fortran constants

These must be used exactly as presented below.

```
*  Pi
      DOUBLE PRECISION DPI
      PARAMETER ( DPI = 3.141592653589793238462643D0 )

*  2Pi
      DOUBLE PRECISION D2PI
      PARAMETER ( D2PI = 6.283185307179586476925287D0 )

*  Radians to hours
      DOUBLE PRECISION DR2H
      PARAMETER ( DR2H = 3.819718634205488058453210D0 )

*  Radians to seconds
      DOUBLE PRECISION DR2S
      PARAMETER ( DR2S = 13750.98708313975701043156D0 )

*  Radians to degrees
      DOUBLE PRECISION DR2D
      PARAMETER ( DR2D = 57.29577951308232087679815D0 )

*  Radians to arc seconds
      DOUBLE PRECISION DR2AS
      PARAMETER ( DR2AS = 206264.8062470963551564734D0 )

*  Hours to radians
      DOUBLE PRECISION DH2R
      PARAMETER ( DH2R = 0.2617993877991494365385536D0 )

*  Seconds to radians
      DOUBLE PRECISION DS2R
      PARAMETER ( DS2R = 7.272205216643039903848712D-5 )

*  Degrees to radians
      DOUBLE PRECISION DD2R
      PARAMETER ( DD2R = 1.745329251994329576923691D-2 )

*  Arc seconds to radians
      DOUBLE PRECISION DAS2R
      PARAMETER ( DAS2R = 4.848136811095359935899141D-6 )
```

SOFA C constants

The constants used by the C version of SOFA are defined in the header file sofam.h.

IAU STANDARDS OF FUNDAMENTAL ASTRONOMY BOARD

Current Membership

John Bangert	United States Naval Observatory
Steven Bell	Her Majesty’s Nautical Almanac Office
Mark Calabretta	Australia Telescope National Facility
Nicole Capitaine	Paris Observatory
William Folkner	Jet Propulsion Laboratory
George Hobbs	Australia Telescope National Facility
Catherine Hohenkerk	Her Majesty’s Nautical Almanac Office (Chair)
Wen−Jing Jin	Shanghai Observatory
Brian Luzum	United States Naval Observatory (IERS)
Zinovy Malkin	Pulkovo Observatory, St Petersburg
Jeffrey Percival	University of Wisconsin
Patrick Wallace	Rutherford Appleton Laboratory

Past Members

Wim Brouw	University of Groningen
Anne−Marie Gontier	Paris Observatory
George Kaplan	United States Naval Observatory
Dennis McCarthy	United States Naval Observatory
Skip Newhall	Jet Propulsion Laboratory

The e−mail for the Board chair is Catherine.Hohenkerk@ukho.gov.uk