The KRC Planetary ORBit (PORB) System

Hugh H. Kieffer   File= /krc/Doc/hporb.tex 2014jan24

April 28, 2014

Contents

This file largely documents the design of PORB. It should not be needed by the general KRC user. A separate document *PUG.tex* and its derivatives are a Users Guide to the PORB system.

several revisions, primarily is how L_S is calculated. To avoid confusion in Version 2 with version 1 routines that did similar things, some routine names have been changed:

PORB1 ⇒ PORBIG: create the geometry arrays for a KRC input file

PORB ⇒ PORBIT: compute values at a specific date

The main program **porbmn** is basically a pre-processor to generate the geometry matrix used by KRC. The default output of **porbmn** is a file *PORBCM.mat* which concatonates new 7-lines matricies on the bottom; any 7-line set can be pasted into a KRC version 2.2.1 or later KRC input file.

There are several coordinate orientation systems involved, listed in §3. To minimize repetative calculations, the version 2 PORB system works largely in the F (orbital plane) coordinate system.

2 Source files

Files that can be read by the PORB system cover four catagories, each file has its own format:

Planets: One file for orbits and another for spin-axes

Minor (or small) bodies: Orbits and spin-axis direction for each object. Two versions.

Comets: Orbit size specified by perihelion distance.

ExoPlanets: Orbit specified by semi-major axis, period and eccentricity. Relative orientation of the body spin axis specified by obliquity and Ls at periapsis.

All these files are read by **porbel.f**

2.1 Planets

standish.tab [5] Keplerian elements and rates of change for 9 planets in mean ecliptic and equinox of J2000 valid for 1800 to 2050.

spinaxis.tab [2] Planets: Direction of the pole and rates of change in ICRF, which differs from J2000 equatorial system by less than 0.1 arcsecond.

2.2 Minor planets / asteroids

2.2.1 Web sources

There are several sources on the web. Some I found did not have unambiguous definition of the coordinate systems used; these were established through email exchanges

PDS Small Bodies Node http://pdssbn.astro.umd.edu/

Primary source for spin axis.

ecliptic coordinates of equinox 1950

```
 PDS Small Bodies Node http://pdssbn.astro.umd.edu/
 DATA ARCHIVES> Archived at SBN> by target > Asteroids
   > orbital date   or > Physical properties > Spin Vectors
Downloaded data files stored in : /work2/ephem/SmallBodies/
   AstSpinVector/  Version 12: obsolete
     EAR-A-5-DDR-PROPER-ELEMENTS-V1.0 ->     see orbit/num70pro.tab
      EAR-A-5-DDR-ASTEROID-SPIN-VECTORS-V5.0  see document/spin.pdf
   LightcurveV13/ version 13, current.  See  Guides  for aligned column labels
     data/lc_spinaxis.tab
In: catalog/dataset.cat
    Kryszczynska, A., A. La Spina, P. Paolicchi, A.W. Harris, S.
    Breiter, and P. Pravec, New findings on asteroid spin-vector
    distributions, Icarus 192, 223-237, 2007."
  I have as /work2/Reprints/Ephem/Krys07AsteroidSpin.pdf

Found web note by Bill Owen (JPL): the obliquity in 1950 was 23 deg 26' 44.836".
```

Minor Planet Center http://www.minorplanetcenter.net/iau/info/OrbElsExplanation.html

ArgPeri, Node and Inclin are in J2000.0, must be ecliptic

JPL Small-Body Database Browser http://ssd.jpl.nasa.gov/sbdb.cgi

heliocentric ecliptic J2000. No spin-axis data.

2.2.2 PORB input files

There are two PORB-system ASCII files that differ slightly in format. Both contain one element per line and include spin-axis.

minor.tab One value per line. Items are:
(this is the order for *minor.tab*, the order in *small.tab* is slightly different)
Object name. May be within single quotes
Epoch in full Julian date 2000.0= 2451545.D0
Semi-Major axis in A.U.: a
Eccentricity: e or ϵ
Inclination of mean orbit to ecliptic : i [deg]
Longitude of the ascending node: Ω [deg]
Argument of perihelion: ω [deg]
Mean anomaly at epoch: M [deg]
Right Ascension of Pole, J2000 [deg]
Declination of Pole, J2000 [deg]
Prime meridian at epoch [deg]
Sideral rotation period, [days]

small.tab Newer format designed to allow cut-and-paste from the JPL Small-Body Database Browser. Rotation period in hours. Last two items are spin-axis direction; must get these data from other sources.

2.3 Comets

comet.tab Some comets: orbit and spin. Originally developed for the comet version of KRC, which included sublimation with a moving depth coordinate system, development of a coma, and non-gravitational accelleration terms. None of these are in the current version of KRC.
WARNING: Some entries are old. Consistency of source orientation systems uncertain.

Format nearly the same as *minor.tab* except for use of perihelion distance q rather than semi-major axis a. They are related by: $q = a(1 - \epsilon)$.

2.4 ExoPlanets

Obliquity and season at periastron were chosen as alternatives to the conventional three items that specify orbit orientation (node, inclination, argument of periapsis) but which are relatively meaningless for planetary system without a well-defined reference plane. The two chosen are easy to understand and have direct effect on temperatures. **probel.f** forces planet obliquity to be at least 0.1°because a value of zero causes NAN's.

To implement this, must set J2000 obliquity `OBL` to zero in **probig.f** to nullify the conversion from ecliptic to equatorial coordinates.

The input items include the host star distance in LightYears and Visual Magnitude; these are used to compute the radiated power from the star in comparison to the Sun and this factor is printed as a convenience for the user.
factor = 10ˆ(- 0.4 (VisMag star - AbsMag Sun)) * (Distance/StandardDistance)^2

Absolute magnitude of the Sun is 4.83 . The standard distance is 10 parsecs, or 32.616 LightYears
This calculation ignores the spectral type of the star so is only an approximation.

The synodic day is computed from the siderial day and the length of the year; this value is printed as a convenience for the user.
SydDay = SidDay(1.+ SidDay/year)

Last part of file has unformatted info on some objects and host stars

exoplan.tab Contains a few sample concensus exoplanets

2.5 Reprints

Reprints related to PORB are stored at: /work2/Reprints/Ephem/

2.6 Obsolete

seidel.tab [4] 9 planets with Pole
sturms.tab Mercury, Mars, Halley; each with Pole. Must be from [6], which is not available. Similar data is in [3]

The notation described below promotes straight-forward code generation. Matrix and vector names indicate precise meaning. First letter is “to” and second letter is “from”

Rotation matricies have ’RM’ as 3rd and 4th letter

Matrix products drop the common middle letter;

Vectors should 5 or more letters

Vector 3rd letter is the orientation system

Letters 4 and 5:

xx (or xxx) indicates elements are X,Y,Z

xu means it is a unit-length vector

Scalar values usually have 4 letters; 4th is the element

X,Y,Z are those elements

A or P is latitude, B or Q is right-hand longitude, C is co-latitude

Vectors should be added or subtracted only if all letters after the first are the same, and only the first two letters change for the result.

Orientation systems used are:

A = Astronomic: master reference inertial system (J2000, virtually identical to the ICRS or FK5)

Z toward Earth mean N. pole at J2000.0; X toward Vernal equnox at J2000.0

Older implementations were EME50 and B1950

B = Body: Target body spin axis inertial;

Z toward bodies right-hand spin axis, X toward its falling node.

i.e., When direction from Planet to Sun is along X, it is the spring equinox

E = Ecliptic: Z toward pole of Earth’s orbit; X toward Vernal Equinox

F = ”focal”: Z toward right-hand pole of body’s orbit; X from focus (central body) to periapsis

Vector ends are:

H=Sun (Helios)

P=Planet or orbiting body

X,Y,Z unit vectors along those axes

E.g., ZPAXU is unit vector toward body positive spin axis (Z) from the body(P) in the J2000 system (A), all 3 components and of unit length (XU)

PHEXX is vector from Sun to body in Ecliptic coordinates.

4 What KRC needs

TCARD calls PORBo to initiate all time-independent quantities

TSEAS increments the date and needs to be able to compute L_S.

Calls PORBIT to get the current SDEC, Ls and DAU

TLATS needs the current solar declination and the heliocentric distance in AU.

Use SDEC and DAU computed in TSEAS

As of 2013jun17 begin revision of PORB and KRC to use only the following.

All orbits specified in J2000 ecliptic

All spin axes specified in J2000 equatorial

?? allow flag for others. e.g, B1950 ecliptic

Primary system for geometry calculation is J2000 equitorial.

Get planet postion at any time in orbital system.

Convert to ecliptic

rotate to J2000

porbcm.inc will contain

PORB version number, object name, epoch. source table[s]

Orbit constants (6+) spin axis direction (2), rate and base (2)

rotation matricies (9 words each) and values needed to convert between systems

Values for the current MJD: SDEC, DAU, Lsubs

For eccentric objects, want time (or fraction of period) from perihelion.

But this is simply (MJD-TJP)/OrbitPeriod

5 PORBEL: get elements

Will read elements and spin-vector data for one object from a (or two) data file, move several values into common and return others.

2 e: eccentricity [radians, radians/century]
3 I : inclination [degrees, degrees/century]
4 L : mean longitude [degrees, degrees/century]
$L = \Omega + \omega + M$ is a compound angle measured in two planes:
From VE to node in plane of reference
then to periapsis and on to mean anomaly in orbital plane
5 ϖ : longitude of perihelion [degrees, degrees/century] ($\varpi = \Omega + \omega$)
ϖ is a compound angle measured in two planes:
ω, the argument of periapsis, measured in the plane of orbit
6 Ω: longitude of the ascending node [degrees, degrees/century]
measured in the reference plane.

Standish gives his algorithm for getting position vector: which included solveing for eccentric anomaly in degrees. I do things a little differently,

1. Compute the value of each element at the requested epoch: =base+T*rate
   E.g. $a_t = a_0 + T\dot{a}$ T is time from J2000.0 in Julian centuries;
   $T = (T_{\rm eph} - 2451545.0)/36525$. I ignore the difference between ephemeris time and UTC.
2. Compute the requested epoch in days: $t = T * 36525$.
   Below omit the $_t$ subscript.
3. Compute the orbit period in days: $P_o = Y_s a^{3/2}$ where Y_s is the siderial year for Earth; 365.256363004
4. NOPE Compute argument of perihelion: $\omega = \varpi - \Omega$
5. Compute the mean anomaly at epoch: $M = L - \varpi$ in degrees
6. Compute a time at periapsis: $t_p = t - P_o * M/360$.
7. Compute Ls at periapsis L_{Sp}. This is the angle from the planets spring equinox to periapsis in the orbital plane.
8. Compute rotation matrix from ecliptic to seasonal system **BF**
   THEN, FOR EACH TIME:
9. Mean anomaly: $M = (t - t_p) * 360/P_o$
10. Obtain the eccentric anomaly E by iterating Kepler's equation for M:
    $M = E - e\sin E$ where e is the eccentricity
    Standish does this in degrees, PORB does this in radians
11. Compute the heliocentric coordinates in the orbital plane (F)
    $x = a(cosE - e)$ and $y = a\sqrt{1-e^2}sinE$ and PHFxx=[x,y,0]
    True anomaly: $\nu = \arctan(y, x)$ and $L_S = L_{Sp} + \nu$
    Heliocentric distance is $a(1. - e\cos E)$ or $\sqrt{x^2 + y^2}$.
12. Rotate HPF into the seasonal coordinate system to get the sub-solar latitude
    HPBxx= **BF**$\cdot$ (-PHFxx)

5.0.1 Solution working in the orbit-plane (F) system

t_p =TJP is independent of coordinate system
Will need **FA** to move spin axis into F.
Create **AF** from Kepler elements and transpose it.
Will need **BF** to convert PHF into Sub-solar latitude
Create from: Z= SpinPole expressed in F system
X is toward spring equinox
spring equinox vector XB is spinAxis-cross-OrbitPole (in any coord sustem)
in F system, XB = in direction of ZBF cross ZFF; and ZFF$\equiv$[0,0,1]
normalize to get XBFu
(ZB cross XB) will generate vector in direction of YB
normalize to get YBFu
XBFu, YBFu, ZBFu make up the three rows of **BF**
True anomaly at spring equinox is $\nu_V = \arctan(y/x)$
where x,y are the components of XBFu.

Convert to Eccentric anomaly by Keplers equation

Take negative of PHFxx to get vector from planet to SUn: HPFxx

Convert X and Y components of PHFxx to get true anomaly of Sun ν_S

Subtract true anomaly of Sun at spring equinox to get L_S

L_S in radians is simply $\nu_S - \nu_V$

5.0.2 Inverse: Ls to date

Need mean anomaly

Convert Ls to True anomaly of planet from sun: $\nu = L_S - 180. + \nu_V$

Convert True anomaly ν to eccentric anomaly E:

$\cos E = (e + \cos\nu)/(1 + e\cos\nu)$ and $\sin E = \sqrt{1-e^2}\sin\nu/(1 + e\cos\nu)$

Because denominators are the same, $\tan E = (\sqrt{1-e^2}\sin\nu)/(e + \cos\nu)$

Convert eccentric anomaly to mean anomaly: Keplers equation: $M = E - e\sin E$

Convert to days from periapsis: tfp= M[deg] * P/360.

Add the date at periapsis: t=tfp+t_p

Construct HPFxx, then similar steps as above to get DAU and sub-solar declination

$r = a(1-e^2)/(1 + e\cos\nu)$

5.0.3 Orbital element manipulation

Standish: $M = L - \varpi \Leftrightarrow M_T = L_0 + T\dot{L} - (\varpi_0 + T\dot{\varpi})$

KRC: $M_t = (t - t_p)M'$ where t is Julian days from J2000.0 (thus, $t = T * 36525$), t_p is a time of perihelion (M mod 360 =0) and M' is the effective mean anomaly rate.

$$M_T = (L_0 - \varpi_0) + T\left(\dot{L} - \dot{\varpi}\right) \quad \Leftrightarrow \quad M_T = (T - T_p)\left(\dot{L} - \dot{\varpi}\right) \equiv M_T = (t - t_p)\left(\dot{L} - \dot{\varpi}\right)/36525 \tag{1}$$

Yields $T_p = -(L_0 - \varpi_0)/\left(\dot{L} - \dot{\varpi}\right)$

Convert from centuries to days: $M_t = (t - t_p)\,M'$ where $M' = \left(\dot{L} - \dot{\varpi}\right)/36525$; $\dot{L}$ is dominant.

User selects a T near the year of interest. Use Eq. 1:middle to get M_1; modulo 360 to get the equivalent M_2. Then find $t_p = -M_2/M'$??

Compute orbital period P from Keplers third law: $P = Y_s\sqrt{a^3}$

where Y_s is the siderial year for Earth

Compute time of periapsis near the desired date as $t_p = T * 36525 - P \cdot M_2/360.$ in Julian Days from J2000

$t_p \equiv$ `TJO` is effectively the reference time for later orbital calculations.

IDL tests (qtev.pro @ 28) of the orbital period P_a derived from semi-major axis a and P_M derived from the effective anomaly rate M' show fractional differences $(P_a - P_M)/P_M$ of up to 5.E-6 for terrestrial planets. The fractional rate of change of semi-major axis $\dot{a}/a$ is <1.2E-7 /year for the terrestrial planets and up to 1.3E-6 for the outer planets.

Then reads the spin-vector data and adjust the pole direction to the requested epoch. Get the siderial days in hours.

6 Coordinate conversions

6.1 Basic principals

Coordinate systems are defined by a center and an orientation. Common centers are: barycenter of the solar system, gravitational center of Earth. Orientations are defined relative to another coordinate system or on some physical basis.

Within a coordinate system, a vector $\overline{PO}$ from the origin O to a point P can be rotated around an axis (e.g., the Z axis) by an angle θ (positive anti-clockwise) to a new position by $\overline{P_2O} = \mathcal{R}_Z(\theta)\overline{PO}$ where

$$\mathcal{R}_Z(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

The vector $v \equiv \overline{PO}$ can be expressed in a second coordinate system whose axes are rotated around the original Z axis by θ as $v' = \mathcal{R}_Z(-\theta)v$

If the axes of coordinate system **B** are attained by three successive rotations beginning with the axes of coordinate system **A**, typicially in the sequence: 1:$Z(\alpha)$, 2:$X(\beta)$, 3:$Z(\gamma)$, then $v_B = \mathcal{M}v_A$ where the rotation matrix to **B** from **A** is

$$\mathbf{BA} \equiv \mathcal{M} = \mathcal{R}_Z(-\gamma) * \mathcal{R}_X(-\beta) * \mathcal{R}_Z(-\alpha) * \mathbf{I} \tag{3}$$

Also, **AB** can be generated by reversing the sequence of axial rotations and not negating the angles.

6.2 From orbit plane F to J2000=A

Defining angles:
Ω: argument of the node: `ODE` radians
i: inclination of the orbit: `CLIN` radians
ω: argument of perihelion: `ARGP` radians

$$\mathbf{AF} = \mathcal{R}_Z(-\Omega) * \mathcal{R}_X(-i) * \mathcal{R}_Z(-\omega) * \mathbf{I} \tag{4}$$

This is done in the routine **ROTORB**

6.3 From J2000=A to seasonal=B (Body)

Z axis of the B systen is the right-hand spin axis of the body e.g., planet).
X axis is the body's Vernal equinox: where the Sun (in the planet's orbit) rises above the planets equator. This occurs when the direction from the planet to the Sun is along the direction of: Spin-axis cross orbit-normal
Defining angles: those for the orbit, plus:
Right Ascension of spin axis in J2000 equitorial, α, `ZBAB` radians
Declination of spin axis in J2000 equitorial, δ. `ZBAA` radians
COCOSC: Convert α and δ to Cartesian vector: Z_{bA},`ZBAXU`
pole of orbit plane in J2000 is the Z axis of the B system expressed in the A system, or, the last row of the **AF** matrix
ROTCOL: extract Z_{FA}, `ZFAXU`
Get vector to Body vernal equinox: γ, `BV`
$\gamma \equiv X_{bA} = Z_{bA} \times Z_{fA}$, normalised to unity, `XBAXU`
$Y_{bA}Y = Z_{bA} \times X_{bA}$, Y axis of B system in J2000

$$\mathbf{BA} = \begin{bmatrix} X_{bA} \\ Y_{bA} \\ Z_{bA} \end{bmatrix}$$

Finally: $\mathbf{BF} = \mathbf{BA} * \mathbf{AF}$ will convert the Sun to Planet vector in the orbital system to the season system.

7 L_S, the planetocentric longitude of the Sun

L_S ('"L sub S") is the planet's "season" measured as the angle from the planets Vernal Equinox to the planet-to-Sun vector in the orbital plane. As such, it is linear with the True Anomaly of the Planet around the Sun. The term is commonly used for Mars but rarely for other bodies.

7.1 Generic (any body) Ls algorithm

Vernal Equinox is toward the intersection of body equator and body orbit at which the Sun rises into the "North' hemisphere. Thus VE is along spinAxis-cross-OrbitPole .

Do once: Using body orbital elements and spin axis in the same coordinate system; compute the time of periapsis, the angle from the VE to periapsis, and the rotation matrix from the F to the B system

Each time: Calculate the true anomaly, get the HP vector in the F system, rotate it into the B system to get the sub-solar declination; the magnitude of the vector is the heliocentric distance. Offset the true anomaly to get L_S.

My IDL routine LSUBSGEN does this in two steps:
Kode=0: Given the Keplerian elements (and the central body gravitational constant), compute a set of intermediate constants
Kode=1: for a single or vector of request dates (MJD, relative to J2000.0):
Compute the mean anomaly (radians)
Subtract the true anomaly at the Vernal Equinox and convert to degrees.
The SPICE Fortran version lspcn.f accommodates aberrations; its implementation of coordinate systems seems peculiar.

7.2 Approximation to Ls for Mars

7.2.1 Summary

Major innovations of Allison and McEwen are the revised pole position and inclusion of the major planetary perturbations (PBS). However, once the PBS are averaged out so that there is a single relation between days from $L_S = 0$ and L_S, the difference between my older routines and new versions are quite small.

Mars [page 221.8 left]). They evaluated these data for a period of ±67 Mars Years (MY) from 2000.0; 1874:2127. They provide as Equation 14 the generation of MJD4 ≡ JD-2400000.5 as a function of L_S and MY.

Equations 16 to 20 give relations for the mean anomaly, the Fictitious Mean Sun, pertubations by the planets, L_S and the equation of time (EOT)

The IDL routine **lsam.pro** impliments all these relations, along with the obliquity of Mars, Heliocentric range and sub-solar latitude. LSAM operates in MJD ≡ JD-2451545.0 corresponding to the IAU J2000 system.

LSAM also impliments the reverse relation, from MJD to MY and L_S, including an optional empirical correction I developed for 1985 to 2026. It reduces the mean absolute closure from 0.0187 to .0023°.

Δt_{J2000} is in TT; 222.4b and Eq. 15

7.2.3 Prior Ls routine

The IDL routine **l_s.pro** in use from 1997 (or earlier) to 2011 was the basis for L_S calculations in KRC. It was derived from tables of L_S based on USNO MICA runs of every 5 days that had 0.01° resolution.

l_s.pro has forward and reverse in form of:

$$L_S = a_0 + a_1 x + a_2 \cos(x - a_3) + a_4 \cos(2x - a_5) + a_6 \cos(3x - a_7) \tag{5}$$

where x is the fractional Mars year.

7.2.4 Conversion from non-linear to linear fit

Convert Eq. 5 to a linear set of basis functions by the relation

$$\cos(A - B) = \cos A \cos B + \sin A \sin B \qquad \text{OR} \qquad c_1 \cos A + c_2 \sin A \tag{6}$$

where the right side holds when B is constant. E.g.,

$$a_6 \cos(A - B) = a_6 [\cos A \cos B + \sin A \sin B] \qquad \text{Fit} \qquad c_1 \cos 3x + c_2 \sin 3x \tag{7}$$

Then have:

$$l_s = a_0 + a_1 x + c_2 \cos x + c_3 \sin x + c_4 \cos 2x + c_5 \sin 2x + c_6 \cos 3x + c_7 \sin 3x \tag{8}$$

E.g.,

$$+a_6 \cos(3x - a_7) = c_6 \cos 3x + c_7 \sin 3x \tag{9}$$

$$a_6 \cos B = c_6 \qquad \text{AND} \qquad a_6 \sin B = c_7 \tag{10}$$

$$\tan B \equiv \frac{\sin B}{\cos B} \qquad \text{Thus} \qquad B = \arctan \frac{c_7}{c_6} \tag{11}$$

$$a_6 = c_6 / \cos B \qquad \text{AND} \qquad a_7 = B \tag{12}$$

All implimented in **qlsam.pro** @27.

qlsam can use four different time periods, set by parameters

0:1 Check on TT-UTC

2:4 Evaluation of the closure function

5:7 Find mean Mars tropical year and mean-anomaly rate at $L_S = 0$

Using 40 MY to each side of J2000 yields 686.97053 and 0.49925897

the year is .00203 smaller than AM list in Table 3

8:10 Fit annual cosine function

is must fall within the 5:7 range

FTO make the annual approximation: for a set of MY, convert JD uniformly spaced and offset so as to never cross a MY boundary; compute the L_S using LSAM. Fit L_S as function of days from start of year

7.2.5 Find mean martian year length and MJD of Ls=0

AM give mean motion as meanmo=0.52402075D0, with tropical orbit period 686.97256, or m=0.52403840

Pick a time range, and compute Ls for at least 6 times each Mars' year

print,25+[-67,67]

SVD fit minimizes RMS, whereas the AMOEBA minimization can used either RMS or MAR.

to get vector from the planet to the Sun in the season system, $HPB = \mathbf{BF} * {}^{T}HH$. The latitude is the sub solar latitude.

7.4 Form for KRC system

Running PORBMN produces the geometry block as text:
calls PORBIG which
calls PORBEL to read orbital elements file and convert `DSJA` to period.
Fundamental constant used is: SIDYR=365.256363004 ! siderial year in days
Calculations done in REAL*8 but returned in REAL*4
PORBEL has no access to common, returns all values as arguments
PORBIG fills common
PORBIO writes common as a block of text into a file
Default file name is PORBCM.mat

Within KRC, Ls is computed in TSEAS based on true anomaly derived from PORB planet position and the Ls at periapsis; TSEAS uses the day of each season and does not consider convergence days. Because KRC version 2 uses modern pole direction and mean orbit, these should be within the magnitude of planetary perturbations of the full Allison and McEwen version in LSAM.
KRC program: normal mode:
KRC calls PORB0 to read the geometry matrix into common
TSEAS call PORBIT with DJU5 to produce $L_S \equiv$ SUBS, SDEC, DAU
PORBIT accesses porbcm and converts DJU5 to Time from periapsis
then calls ORBIT with TPER and PERIOD to get heliocentric radius and Cartesian coord. in the orbital plane.
Calls ECCANOM to solve Kepler's equation for eccentric anomaly
ORBIT uses SMA to compute the position in the orbital plane
PORBIT uses the Cartesian coordinates to get the true anomaly in radians, TAR
L_S is angle from True anomaly at spring equinox,TAV, converted to degrees
The onePoint mode calls PORBIT to convert L_S to a modified Julian date, then computes the starting date.

7.4.1 Preparation

There are several coordinate orientation systems involved, listed in §3.
Derive Year $\equiv$ Mars mean tropical year from separation of the extreme $L_S = 0$ times from Allison for whatever period is chosen.

8 Common porbcm.inc

porbcm.inc will contain
PORB version number, object name, epoch. source table[s]
Orbit constants (6+) spin axis direction (2), rate and base (2)
rotation matricies (9 words each) and values needed to convert between systems
Values for the current MJD: SDEC, DAU, Lsubs
For eccentric objects, want time (or fraction of period) from perihelion.
But this is simply (MJD-TJP)/OrbitPeriod

9 FORTRAN routine list

```
CALDATE convert julian date (base 2440000) to year,month,day,day-of-week
ECCANOM iterative solution of  Kepler's equations for eccentric orbit.
EPHEMR  prints orbital position and date table.  PORB system
J2000   Rotation matrix from J2000 to other epochs
JDATE   convert julian date (base 2440000) to year,month,day,day-of-week
KEPLER  compute orbiting body position from classic elements and time.
MPRINT  Print a 3x3 matrix with ID;  E format
MPROD3  Matrix product (hard-coded for size=3).
OBLIP   Obliquity of a planet.  Default precision
ORBIT   Compute radius and coordinates for elliptical orbit. DefPrec.
PORB    computes planetary angles and location for specific time.
PORB0   Planetary orbit. Read pre-computed matrices and do rotation; minimal for KRC
PORBEL  read planetary orbital element file, compute basic constants
PROBIG  Read orbital elements from disk files. Initiate porbcm
```

```
RECONC  C-kernel reconcilliation for tri-axial ellipsoids (worst case)
ROTAX   Change rotation matrix to include additional rotation 'R'
ROTDIA  Form diagonal matrix of magnitude  R.
ROTORB  construct rotation matrix from classic orbital elements
SPCREV  returns spacecraft revolution number for Viking
TRUEANOM iterative solution of keplers equations for eccentric orbit
YMD2J2  convert year, month, day to Julian date offset from J2000
 cocosp.f
  COCOSP  General coordinate conversion package, many routines.
  COCOCM  Coordinate conversion: cartesian to mapping
  COCOMC  Coordinate conversion: mapping to cartesian
  COCOSC  Coordinate conversion: spherical to cartesian
  COCOCS  Coordinate conversion: cartesian to spherical.
  COCOMS  Coordinate conversion: mapping to spherical angles
  COCOSM  Coordinate conversion: spherical to mapping angles
  COCEMC  Coordinate conversion: ellip. mapping to cartesian
  COCECM  Coordinate conversion: cartesian to ellip. mapping
 rotmsp.f
  ROTMSP  General 3-dimension rotation matrix geometry package.
  ROTORB  Construct rotation matrix from classic orbital elements
  ROTMAT  Derive rotation matrix from pointing triple.
  ROTATE  Rotate a vector  !  U =  B *  V
  ROTAX   Change rotation matrix to include additional rotation 'R'
  ROTDIA  Form diagonal matrix of magnitude  R.
  ROTRIP  Converts rotation matrix to pointing triple.
  ROTCOL  Extract  N'th column from a 3x3 matrix. Consecutive in 9-vector
  ROTROW  Extract  N'th row    from a 3x3 matrix. Spaced by 3 in 9-vector
  ROTSHO  Print a 3x3 rotation matrix with ID.
  ROTV    Rotate a vector about a Cartesian axis
  VROTV   Vector rotation about another vector
  ROTZXM  Make rotation matrix from vectors along Z-axis, and in X-Z plane
  TRANS3  Transpose a 3x3 matrix,  A and  B may be same array.
  ROTEXM  Modify rotation matrix to new system with axes interchanged
  ROTEST  Tests deviation of matrix from a rotation matrix
  ROTEXV  Rotate a vector to system with axes interchanged
  MEQUAL  Equate one 3x3 matrix to another.
  MPROD3  Matrix product (hard-coded for size=3).
 vaddsp.f
  VADD    Add two vectors of dimension 3.   single precision
  VCROSS  Cross product of two vectors of dimension 3.
  VDOT    Calculates the dot product of two vectors of dimension 3.
  VEQUAL  Equate one vector of dimension 3 to another.
  VMAG    Get magnitude (length) of a vector of dimension 3.
  VNEG    Negate each element of a vector of dimension 3.
  VNORM   Normalize a vector of dimension 3.
  VPRF    Print a vector of dimension 3 in user format.
  VSCALE  Multiply a vector of dimension 3 by a constant.
  VSHOW   Print a vector of dimension 3 as cartesion and spherical angles
  VSUB    Find difference of two vectors of dimension 3
  VSUBR   Find reduced-precision difference of two vectors of dimension 3
  VUNIT   Construct unit vector of dimension 3 along one axis

Call sequence: extract from ~/krc/flow.txt

PORBMN  Main program. Includes 'porbcm.inc'
  DATIME   Returns current date and time
 may call the following in any order. All INCLUDE 'porbcm.inc'
  PORBIG    Read orbital elements, compute matrices
    PORBEL    Read any of 4 orbital element files, compute basic constants
```

```
    Vector, cordinateConversion, RotationMatix routines
  PORBIO   Read/write Common to file as text or binary
  EPHEMR   Print ephemeris == geometry versus time
    PORB     Computes planetary angles and location for specific time
      ORBIT    Compute location of body in its orbital plane
      ROTVEC    Apply rotation matrix to rotate a vector
      VNEGS
      COCOCS
    CALDATE
    SPCREV
    ANG360
  PORBQQ   Test computation of matrices
    ROTDIA 2x
    ROTAX  4x
    MPRINT 3x
    MPROD3 2x
Vpkg consists of:
  VADDSP  Vector operations
  ROTMSP  Rotation matrix generation and use
  COCOSP  Cordinate conversions between Cartesian, various forms of spherical/ellip

PORBIT
  ORBIT
    ECCANOM
  Vpkg

PORBIG Read orbital elements from disk file. Initiate porbcm
  User interaction
  PORBEL   read planetary orbital element file, compute basic constants
      Firm-coded four file names
      Reads elements. Converts to R*4 radians
    UPCASE
  Vpkg  with options for display
```

10 Test program qlsam.pro: notes

```
Float values
       0      10000.0  @21 Days before/after j2000
       1      200.000   " delta day
       2      1500.00  @22 Num days INT
       3     -5500.00   " first day
       4      10.0000   " delta day
       5     0.100000  @26 ls of each year
       6     -20.0000   " Start Delta MY from j2000
       7      30.0000   " End "
       8      12.0000  @27 times per year
       9     -20.0000   " Start Delta MY from j2000
      10      30.0000   " End "

Generate LS for set of days after Date of Ls=0, based on
mean Mars yr=  686.97122   jpm0 151.28721
MAR of SVD fit  0.0213669
cca=[-10.329222,57.293064,10.691792,-0.15049282,-0.62572111,1.2711097,-0.05001682,-0.41212631]
 AMOEBA yields  686.9728932  151.2994586  MAR= 0.021764502 so actually worse

Fit reverse
Mean,SD for Abs SVD residual    0.0388522    0.0318471
cca=[19.726299,109.33483,-20.425055,-0.33039734,-0.71641655,0.9093619,0.029254431,-0.9966046]
 AMOEBA yields         1.000079834  0.01697570801  MAR= 0.039837902 worse
```

```
cca=[-10.327384,57.292118,10.690188,-0.15174407,-0.62669644,1.2677819
,-0.050847963,-0.40253532]

from Ls/360. to days from start of year  MAR= 0.0329554
cca=[19.727741,109.3347036,-20.423396,-0.33147164,-0.71622357,0.90714352
,0.029221122,-1.002307]

qlsam Enter selection: 99=help 0=stop 123=auto> 27
 0   const -10.3338651   0.0000196
 1  linear  57.2930678   0.0000059
 2    cos1  10.5684773   0.0000043
 3    sin1  -1.6134259   0.0000049
 4    cos2  -0.1859014   0.0000051
 5    sin2  -0.5974615   0.0000051
 6    con3  -0.0457721   0.0000050
 7    sin3   0.0201688   0.0000048
dell=[-10.333865,57.293068,10.568477,-1.6134259,-0.18590144,-0.5974615,-0.045772095,0.020168757]
Mean,SD for SVD residual    0.0185023    0.0170166

mean Mars yr=       686.97053
Mean Mdot at ls=0:      0.49917967
    FORWARD
Mean,SD for SVD residual  4.60042e-13    0.0251578
Mean,SD for Abs SVD residual    0.0183557    0.0171841
cca=[-10.279889,57.292076,10.690189,-0.15257164,-0.62670167,1.2661439,-0.050841373,-0.40489779]
Mean,SD for SVD residual  4.63891e-13    0.0251578
Mean,SD for Abs SVD residual    0.0183557    0.0171841
    REVERSE
Mean,SD for SVD residual -6.40863e-13    0.0433527
Mean,SD for Abs SVD residual    0.0329746    0.0281055
cca=[19.637114,109.3347,-20.423396,-0.33147169,-0.71622305,0.907144,0.029221461,-1.0022922]
Mean,SD for SVD residual -6.42476e-13    0.0433527
Mean,SD for Abs SVD residual    0.0329746    0.0281055

qlsam Enter selection: 99=help 0=stop 123=auto> 28
L_s  -LSAM mean.Std:    0.0188621    0.0488719
Lsubs-LSAM mean.Std:    0.0203785    0.0242276

Above with 12/year. WIth 19/yr, but using LSUBS based on 12.
cca=[-10.282388,57.293034,10.690925,-0.15239109,-0.62571515,1.2673568,-0.050011825,-0.41772512]
Any key to go
Mean,SD for SVD residual  9.65577e-13    0.0252099
Mean,SD for Abs SVD residual    0.0185286    0.0170818
cca=[19.636891,109.33478,-20.423395,-0.33147983,-0.71612938,0.90714254,0.029221834,-0.99989311]
Any key to go
Mean,SD for SVD residual -9.91491e-13    0.0434017
Mean,SD for Abs SVD residual    0.0332932    0.0278180

qlsam Enter selection: 99=help 0=stop 123=auto> 28
L_s  -LSAM mean.Std:    0.0188071    0.0488767
Lsubs-LSAM mean.Std:    0.0673968    0.0257574
```

References

[1] M. Allison and M. McEwen. A post-Pathfinder evaluation of areocentric solar coordinates with improved timing recipes for Mars seasonal/diurnal climate studies. *Plan. Space Sci.*, 48:215–235, February 2000.

[2] B. A. Archinal, M. F. A'Hearn, E. Bowell, A. Conrad, G. J. Consolmagno, R. Courtin, T. Fukushima, D. Hestroffer, J. L. Hilton, G. A. Krasinsky, G. Neumann, J. Oberst, P. K. Seidelmann, P. Stooke, D. J. Tholen, P. C.

[3] W.G. Melbourne, J.D. Mulholland, W.J. Sjogren, and Jr. F.M. Sturms. Constants and related information for astrodynamic calculations, 1968. *JPL Tech. Rep. 32-106*, http://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19690001472_1969001472.pdf:1–68, 1968.

[4] P. K. Seidelmann, L. E. Doggett, and M. R. Deluccia. Mean elements of the principal planets. *Astron. Jour.*, 79:57–+, January 1974.

[5] E.M. Standish and J.G. Williams. Keplerian elements of the approximate positions of the major planets. In *Explanatory supplement to the Astronomical Almanac, Section 8.10*, pages 1–4. online at http://iau-comm4.jpl.nasa.gov/keplerformulae/kepform.pdf, 2006.

[6] F. M. Sturms Jr. Polynomial expressions for planetary equators and orbit elements with respect to the mean 1950.0 coordinate system. Jet Propulsion Laboratory Technical Report, 32-1508, 1971.