SPACETRACK REPORT NO. 3

Models for Propagation of NORAD Element Sets

FELIX R. HOOTS
RONALD L. ROEHRICH

DECEMBER 1980

Package Compiled by
TS Kelso

31 December 1988

General perturbations element sets generated by NORAD can be used to predict position and velocity of Earth-orbiting objects. To do this one must be careful to use a prediction method which is compatible with the way in which the elements were generated. Equations for five compatible models are given here along with corresponding FORTRAN IV computer code. With this information a user will be able to make satellite predictions which are completely compatible with NORAD predictions.

Contents

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

Requests for additional copies by agencies of the Department of Defense, their contractors, and other government agencies should be directed to the:[1]

Defense Documentation Center
Cameron Station
Alexandria VA 22314

All other persons and organizations should apply to the:

Department of Commerce
National Technical Information Service
5285 Port Royal Road
Springfield VA 22161

Address special inquiries to:

Project Spacetrack Reports
Office of Astrodynamics
Aerospace Defense Center
ADC/DO6
Peterson AFB CO 80914

[1] Editor's note: These are the addresses from the original document and may no longer be valid.

1 INTRODUCTION

NORAD maintains general perturbation element sets on all resident space objects. These element sets are periodically refined so as to maintain a reasonable prediction capability on all space objects. In turn, these element sets are provided to users. The purpose of this report is to provide the user with a means of propagating these element sets in time to obtain a position and velocity of the space object.

The most important point to be noted is that not just any prediction model will suffice. The NORAD element sets are "mean" values obtained by removing periodic variations in a particular way. In order to obtain good predictions, these periodic variations must be reconstructed (by the prediction model) in exactly the same way they were removed by NORAD. Hence, inputting NORAD element sets into a different model (even though the model may be more accurate or even a numerical integrator) will result in degraded predictions. The NORAD element sets must be used with one of the models described in this report in order to retain maximum prediction accuracy.

All space objects are classified by NORAD as near-Earth (period less than 225 minutes) or deep-space (period greater than or equal 225 minutes). Depending on the period, the NORAD element sets are automatically generated with the near-Earth or deep-space model. The user can then calculate the satellite period and know which prediction model to use.

2 THE PROPAGATION MODELS

Five mathematical models for prediction of satellite position and velocity are available. The first of these, SGP, was developed by Hilton & Kuhlman (1966) and is used for near-Earth satellites. This model uses a simplification of the work of Kozai (1959) for its gravitational model and it takes the drag effect on mean motion as linear in time. This assumption dictates a quadratic variation of mean anomaly with time. The drag effect on eccentricity is modeled in such a way that perigee height remains constant.

The second model, SGP4, was developed by Ken Cranford in 1970 (see Lane and Hoots 1979) and is used for near-Earth satellites. This model was obtained by simplification of the more extensive analytical theory of Lane and Cranford (1969) which uses the solution of Brouwer (1959) for its gravitational model and a power density function for its atmospheric model (see Lane, et al. 1962).

The next model, SDP4, is an extension of SGP4 to be used for deep-space satellites. The deep-space equations were developed by Hujsak (1979) and model the gravitational effects of the moon and sun as well as certain sectoral and tesseral Earth harmonics which are of particular importance for half-day and one-day period orbits.

The SGP8 model (see Hoots 1980) is used for near-Earth satellites and is obtained by simplification of an extensive analytical theory of Hoots (to appear) which uses the same gravitational and atmospheric models as Lane and Cranford did but integrates the differential equations in a much different manner.

Finally, the SDP8 model is an extension of SGP8 to be used for deep-space satellites. The deep-space effects are modeled in SDP8 with the same equations used in SDP4.

3 COMPATIBILITY WITH NORAD ELEMENT SETS

The NORAD element sets are currently generated with either SGP4 or SDP4 depending on whether the satellite is near-Earth or deep-space. For element sets sent to external users, the value of mean motion is altered slightly and a pseudo-drag term ($\dot{n}/2$) is generated. These changes allow an SGP user to make compatible predictions in the following manner. If the satellite is near-Earth, then the pseudo-drag term used in SGP simulates the drag effect of the SGP4 model. If the satellite is deep-space, then the pseudo-drag term used in SGP simulates the deep-space secular effects of SDP4.

For SGP4 and SDP4 users, the mean motion is first recovered from its altered form and the drag effect is obtained from the SGP4 drag term (B^*) with the pseudo-drag term being ignored. The value of the mean motion can be used to determine whether the satellite is near-Earth or deep-space (and hence whether SGP4 or SDP4 was used to generate the element set). From this information the user can decide whether to use SGP4 or SDP4 for propagation and hence be assured of agreement with NORAD predictions.

The SGP8 and SDP8 models have the same gravitational and atmospheric models as SGP4 and SDP4, although the form of the solution equations is quite different. Additionally, SGP8 and SDP8 use a ballistic coefficient (B term) in the drag equations rather than the B^* drag term. However, compatible predictions can be made with NORAD element sets by first calculating a B term from the SGP4 B^* drag term.

At the present time consideration is being given to replacing SGP4 and SDP4 by SGP8 and SDP8 as the NORAD satellite models. In such a case the new NORAD element sets would still give compatible predictions for SGP, SGP4, and SDP4 users and, for SGP8 and SDP8 users, would give agreement with NORAD predictions.

4 GENERAL PROGRAM DESCRIPTION

The five ephemeris packages cited in Section Two have each been programmed in FORTRAN IV as stand-alone subroutines. They each access the two function subroutines ACTAN and FMOD2P and the deep-space equations access the function subroutine THETAG. The function subroutine ACTAN is a two argument (quadrant preserving) arctangent subroutine which has been specifically designed to return the angle within the range of 0 to 2π. The function subroutine FMOD2P takes an angle and returns the modulo by 2π of that angle. The function subroutine THETAG calculates the epoch time in days since 1950 Jan 0.0 UTC, stores this in COMMON, and returns the right ascension of Greenwich at epoch.

One additional subroutine DEEP is accessed by SDP4 and SDP8 to obtain the deep-space perturbations to be added to the main equations of motion.

The main program DRIVER reads the input NORAD 2-line element set in either G-card internal format or T-card transmission format and calls the appropriate ephemeris package as specified by the user. The DRIVER converts the elements to the units of radians and minutes before calling the appropriate subroutine. The ephemeris package returns position and velocity in units of Earth radii and minutes. These are converted by the DRIVER to kilometers and seconds for printout.

All physical constants are contained in the constants COMMON C1 and can be changed through the data statements in the DRIVER. The one exception is the physical constants used only in DEEP which are set in the data statements in DEEP.

In the following sections the equations and program listing are given for each ephemeris model. Every effort has been made to maintain a strict parallel structure between the equations and the computer code.

5 THE SGP MODEL

The NORAD mean element sets can be used for prediction with SGP. All symbols not defined below are defined in the list of symbols in Section Twelve. Predictions are made by first calculating the constants

$$a_1 = \left(\frac{k_e}{n_o}\right)^{\frac{2}{3}}$$

$$\delta_1 = \frac{3}{4} J_2 \frac{{a_E}^2}{{a_1}^2} \frac{(3\cos^2 i_o - 1)}{(1 - {e_o}^2)^{\frac{3}{2}}}$$

$$a_o = a_1 \left[1 - \frac{1}{3}\delta_1 - {\delta_1}^2 - \frac{134}{81}{\delta_1}^3\right]$$

$$p_o = a_o(1 - {e_o}^2)$$

$$q_o = a_o(1 - e_o)$$

$$L_o = M_o + \omega_o + \Omega_o$$

$$\frac{d\Omega}{dt} = -\frac{3}{2} J_2 \frac{{a_E}^2}{{p_o}^2} n_o \cos i_o$$

$$\frac{d\omega}{dt} = \frac{3}{4} J_2 \frac{{a_E}^2}{{p_o}^2} n_o (5\cos^2 i_o - 1).$$

The secular effects of atmospheric drag and gravitation are included through the equations

$$a = a_o \left\{ \frac{n_o}{n_o + 2\left(\frac{\dot{n}_o}{2}\right)(t - t_o) + 3\left(\frac{\ddot{n}_o}{6}\right)(t - t_o)^2} \right\}^{\frac{2}{3}}$$

$$e = \left\{ \begin{array}{ll} 1 - \frac{q_o}{a}, & \text{for } a > q_o \\ 10^{-6}, & \text{for } a \leq q_o \end{array} \right\}$$

$$p = a(1 - e^2)$$

$$\Omega_{s_o} = \Omega_o + \frac{d\Omega}{dt}(t - t_o)$$

$$\omega_{s_o} = \omega_o + \frac{d\omega}{dt}(t - t_o)$$

$$L_s = L_o + \left(n_o + \frac{d\omega}{dt} + \frac{d\Omega}{dt}\right)(t - t_o) + \frac{\dot{n}_o}{2}(t - t_o)^2 + \frac{\ddot{n}_o}{6}(t - t_o)^3$$

where $(t - t_o)$ is time since epoch.

Long-period periodics are included through the equations

$$a_{yNSL} = e \sin \omega_{s_o} - \frac{1}{2} \frac{J_3}{J_2} \frac{a_E}{p} \sin i_o$$

$$L = L_s - \frac{1}{4} \frac{J_3}{J_2} \frac{a_E}{p} a_{xNSL} \sin i_o \left[\frac{3 + 5 \cos i_o}{1 + \cos i_o} \right]$$

where

$$a_{xNSL} = e \cos \omega_{s_o}.$$

Solve Kepler's equation for $E + \omega$ (by iteration to the desired accuracy), where

$$(E + \omega)_{i+1} = (E + \omega)_i + \Delta(E + \omega)_i$$

with

$$\Delta(E+\omega)_i = \frac{U - a_{yNSL}\cos(E+\omega)_i + a_{xNSL}\sin(E+\omega)_i - (E+\omega)_i}{-a_{yNSL}\sin(E+\omega)_i - a_{xNSL}\cos(E+\omega)_i + 1}$$

$$U = L - \Omega_{s_o}$$

and

$$(E+\omega)_1 = U.$$

Then calculate the intermediate (partially osculating) quantities

$$e\cos E = a_{xNSL}\cos(E+\omega) + a_{yNSL}\sin(E+\omega)$$

$$e\sin E = a_{xNSL}\sin(E+\omega) - a_{yNSL}\cos(E+\omega)$$

$${e_L}^2 = (a_{xNSL})^2 + (a_{yNSL})^2$$

$$p_L = a(1 - {e_L}^2)$$

$$r = a(1 - e\cos E)$$

$$\dot{r} = k_e \frac{\sqrt{a}}{r} e\sin E$$

$$r\dot{v} = k_e \frac{\sqrt{p_L}}{r}$$

$$\sin u = \frac{a}{r}\left[\sin(E+\omega) - a_{yNSL} - a_{xNSL}\frac{e\sin E}{1+\sqrt{1-{e_L}^2}}\right]$$

$$\cos u = \frac{a}{r}\left[\cos(E+\omega) - a_{xNSL} + a_{yNSL}\frac{e\sin E}{1+\sqrt{1-{e_L}^2}}\right]$$

$$u = \tan^{-1}\left(\frac{\sin u}{\cos u}\right).$$

Short-period perturbations are now included by

$$r_k = r + \frac{1}{4} J_2 \frac{{a_E}^2}{p_L} \sin^2 i_o \cos 2u$$

$$u_k = u - \frac{1}{8} J_2 \frac{{a_E}^2}{{p_L}^2} (7\cos^2 i_o - 1) \sin 2u$$

$$\Omega_k = \Omega_{s_o} + \frac{3}{4} J_2 \frac{{a_E}^2}{{p_L}^2} \cos i_o \sin 2u$$

$$i_k = i_o + \frac{3}{4} J_2 \frac{{a_E}^2}{{p_L}^2} \sin i_o \cos i_o \cos 2u.$$

Then unit orientation vectors are calculated by

$$\mathbf{U} = \mathbf{M} \sin u_k + \mathbf{N} \cos u_k$$

$$\mathbf{V} = \mathbf{M} \cos u_k - \mathbf{N} \sin u_k$$

where

$$\mathbf{M} = \left\{ \begin{array}{l} M_x = -\sin \Omega_k \cos i_k \\ M_y = \cos \Omega_k \cos i_k \\ M_z = \sin i_k \end{array} \right\}$$

$$\mathbf{N} = \left\{ \begin{array}{l} N_x = \cos \Omega_k \\ N_y = \sin \Omega_k \\ N_z = 0 \end{array} \right\}.$$

Then position and velocity are given by

$$\mathbf{r} = r_k \mathbf{U}$$

and

$$\dot{\mathbf{r}} = \dot{r} \mathbf{U} + (r\dot{v}) \mathbf{V}.$$

A FORTRAN IV computer code listing of the subroutine SGP is given below.

```
*      SGP                                                  31 OCT 80
      SUBROUTINE SGP(IFLAG,TSINCE)
      COMMON/E1/XMO,XNODEO,OMEGAO,EO,XINCL,XNO,XNDT2O,XNDD6O,BSTAR,
     1         X,Y,Z,XDOT,YDOT,ZDOT,EPOCH,DS50
      COMMON/C1/CK2,CK4,E6A,QOMS2T,S,TOTHRD,
     1        XJ3,XKE,XKMPER,XMNPDA,AE
      DOUBLE PRECISION EPOCH, DS50

      IF(IFLAG.EQ.0) GO TO 19

*      INITIALIZATION

      C1= CK2*1.5
      C2= CK2/4.0
      C3= CK2/2.0
      C4= XJ3*AE**3/(4.0*CK2)
      COSIO=COS(XINCL)
      SINIO=SIN(XINCL)
      A1=(XKE/XNO)**TOTHRD
      D1=     C1/A1/A1*(3.*COSIO*COSIO-1.)/(1.-EO*EO)**1.5
      AO=A1*(1.-1./3.*D1-D1*D1-134./81.*D1*D1*D1)
      PO=AO*(1.-EO*EO)
      QO=AO*(1.-EO)
      XLO=XMO+OMEGAO+XNODEO
      D1O= C3 *SINIO*SINIO
      D2O= C2 *(7.*COSIO*COSIO-1.)
      D3O=C1*COSIO
      D4O=D3O*SINIO
      PO2NO=XNO/(PO*PO)
      OMGDT=C1*PO2NO*(5.*COSIO*COSIO-1.)
      XNODOT=-2.*D3O*PO2NO
      C5=.5*C4*SINIO*(3.+5.*COSIO)/(1.+COSIO)
      C6=C4*SINIO
      IFLAG=0

*      UPDATE FOR SECULAR GRAVITY AND ATMOSPHERIC DRAG

   19 A=XNO+(2.*XNDT2O+3.*XNDD6O*TSINCE)*TSINCE
      A=AO*(XNO/A)**TOTHRD
      E=E6A
      IF(A.GT.QO) E=1.-QO/A
      P=A*(1.-E*E)
      XNODES= XNODEO+XNODOT*TSINCE
      OMGAS= OMEGAO+OMGDT*TSINCE
      XLS=FMOD2P(XLO+(XNO+OMGDT+XNODOT+(XNDT2O+XNDD6O*TSINCE)*
     1 TSINCE)*TSINCE)

*      LONG PERIOD PERIODICS
```

```
      AXNSL=E*COS(OMGAS)
      AYNSL=E*SIN(OMGAS)-C6/P
      XL=FMOD2P(XLS-C5/P*AXNSL)

*      SOLVE KEPLERS EQUATION

      U=FMOD2P(XL-XNODES)
      ITEM3=0
      EO1=U
      TEM5=1.
   20 SINEO1=SIN(EO1)
      COSEO1=COS(EO1)
      IF(ABS(TEM5).LT.E6A) GO TO 30
      IF(ITEM3.GE.10) GO TO 30
      ITEM3=ITEM3+1
      TEM5=1.-COSEO1*AXNSL-SINEO1*AYNSL
      TEM5=(U-AYNSL*COSEO1+AXNSL*SINEO1-EO1)/TEM5
      TEM2=ABS(TEM5)
      IF(TEM2.GT.1.) TEM5=TEM2/TEM5
      EO1=EO1+TEM5
      GO TO 20

*      SHORT PERIOD PRELIMINARY QUANTITIES

   30 ECOSE=AXNSL*COSEO1+AYNSL*SINEO1
      ESINE=AXNSL*SINEO1-AYNSL*COSEO1
      EL2=AXNSL*AXNSL+AYNSL*AYNSL
      PL=A*(1.-EL2)
      PL2=PL*PL
      R=A*(1.-ECOSE)
      RDOT=XKE*SQRT(A)/R*ESINE
      RVDOT=XKE*SQRT(PL)/R
      TEMP=ESINE/(1.+SQRT(1.-EL2))
      SINU=A/R*(SINEO1-AYNSL-AXNSL*TEMP)
      COSU=A/R*(COSEO1-AXNSL+AYNSL*TEMP)
      SU=ACTAN(SINU,COSU)

*      UPDATE FOR SHORT PERIODICS

      SIN2U=(COSU+COSU)*SINU
      COS2U=1.-2.*SINU*SINU
      RK=R+D10/PL*COS2U
      UK=SU-D20/PL2*SIN2U
      XNODEK=XNODES+D30*SIN2U/PL2
      XINCK =XINCL+D40/PL2*COS2U

*      ORIENTATION VECTORS
```

```
      SINUK=SIN(UK)
      COSUK=COS(UK)
      SINNOK=SIN(XNODEK)
      COSNOK=COS(XNODEK)
      SINIK=SIN(XINCK)
      COSIK=COS(XINCK)
      XMX=-SINNOK*COSIK
      XMY=COSNOK*COSIK
      UX=XMX*SINUK+COSNOK*COSUK
      UY=XMY*SINUK+SINNOK*COSUK
      UZ=SINIK*SINUK
      VX=XMX*COSUK-COSNOK*SINUK
      VY=XMY*COSUK-SINNOK*SINUK
      VZ=SINIK*COSUK

*       POSITION AND VELOCITY

      X=RK*UX
      Y=RK*UY
      Z=RK*UZ
      XDOT=RDOT*UX
      YDOT=RDOT*UY
      ZDOT=RDOT*UZ
      XDOT=RVDOT*VX+XDOT
      YDOT=RVDOT*VY+YDOT
      ZDOT=RVDOT*VZ+ZDOT

      RETURN
      END
```

6 THE SGP4 MODEL

The NORAD mean element sets can be used for prediction with SGP4. All symbols not defined below are defined in the list of symbols in Section Twelve. The original mean motion (n_o'') and semimajor axis (a_o'') are first recovered from the input elements by the equations

$$a_1 = \left(\frac{k_e}{n_o}\right)^{\frac{2}{3}}$$

$$\delta_1 = \frac{3}{2}\frac{k_2}{{a_1}^2}\frac{(3\cos^2 i_o - 1)}{(1-{e_o}^2)^{\frac{3}{2}}}$$

$$a_o = a_1\left(1 - \frac{1}{3}\delta_1 - {\delta_1}^2 - \frac{134}{81}{\delta_1}^3\right)$$

$$\delta_o = \frac{3}{2}\frac{k_2}{{a_o}^2}\frac{(3\cos^2 i_o - 1)}{(1-{e_o}^2)^{\frac{3}{2}}}$$

$$n_o'' = \frac{n_o}{1+\delta_o}$$

$$a_o'' = \frac{a_o}{1-\delta_o}.$$

For perigee between 98 kilometers and 156 kilometers, the value of the constant s used in SGP4 is changed to

$$s^* = a_o''(1-e_o) - s + a_E$$

For perigee below 98 kilometers, the value of s is changed to

$$s^* = 20/\text{XKMPER} + a_E.$$

If the value of s is changed, then the value of $(q_o - s)^4$ must be replaced by

$$(q_o - s^*)^4 = \left[[(q_o - s)^4]^{\frac{1}{4}} + s - s^*\right]^4.$$

Then calculate the constants (using the appropriate values of s and $(q_o - s)^4$)

$$\theta = \cos i_o$$

$$\xi = \frac{1}{a_o'' - s}$$

$$\beta_o = (1 - {e_o}^2)^{\frac{1}{2}}$$

$$\eta = a_o'' e_o \xi$$

$$\begin{aligned} C_2 = {} & (q_o - s)^4 \xi^4 n_o'' (1-\eta^2)^{-\frac{7}{2}} \left[a_o'' \left(1 + \frac{3}{2}\eta^2 + 4e_o\eta + e_o\eta^3 \right) \right. \\ & \left. + \frac{3}{2} \frac{k_2\xi}{(1-\eta^2)} \left(-\frac{1}{2} + \frac{3}{2}\theta^2 \right) (8 + 24\eta^2 + 3\eta^4) \right] \end{aligned}$$

$$C_1 = B^* C_2$$

$$C_3 = \frac{(q_o - s)^4 \xi^5 A_{3,0} n_o'' a_E \sin i_o}{k_2 e_o}$$

$$\begin{aligned} C_4 = {} & 2n_o''(q_o - s)^4 \xi^4 a_o'' {\beta_o}^2 (1-\eta^2)^{-\frac{7}{2}} \left(\left[2\eta(1 + e_o\eta) + \frac{1}{2}e_o + \frac{1}{2}\eta^3 \right] - \frac{2k_2\xi}{a_o''(1-\eta^2)} \times \right. \\ & \left. \left[3(1-3\theta^2) \left(1 + \frac{3}{2}\eta^2 - 2e_o\eta - \frac{1}{2}e_o\eta^3 \right) + \frac{3}{4}(1-\theta^2)(2\eta^2 - e_o\eta - e_o\eta^3)\cos 2\omega_o \right] \right) \end{aligned}$$

$$C_5 = 2(q_o - s)^4 \xi^4 a_o'' {\beta_o}^2 (1-\eta^2)^{-\frac{7}{2}} \left[1 + \frac{11}{4}\eta(\eta + e_o) + e_o\eta^3 \right]$$

$$D_2 = 4a_o''\xi {C_1}^2$$

$$D_3 = \frac{4}{3} a_o'' \xi^2 (17a_o'' + s) {C_1}^3$$

$$D_4 = \frac{2}{3} a_o'' \xi^3 (221a_o'' + 31s) {C_1}^4.$$

The secular effects of atmospheric drag and gravitation are included through the equations

$$M_{DF} = M_o + \left[1 + \frac{3k_2(-1+3\theta^2)}{2a_o''^2 {\beta_o}^3} + \frac{3{k_2}^2(13 - 78\theta^2 + 137\theta^4)}{16a_o''^4 {\beta_o}^7} \right] n_o''(t - t_o)$$

$$\begin{aligned} \omega_{DF} = {} & \omega_o + \left[-\frac{3k_2(1-5\theta^2)}{2a_o''^2 {\beta_o}^4} + \frac{3{k_2}^2(7 - 114\theta^2 + 395\theta^4)}{16a_o''^4 {\beta_o}^8} \right. \\ & \left. + \frac{5k_4(3 - 36\theta^2 + 49\theta^4)}{4a_o''^4 {\beta_o}^8} \right] n_o''(t - t_o) \end{aligned}$$

$$\Omega_{DF} = \Omega_o + \left[-\frac{3k_2\theta}{a_o''^2\beta_o{}^4} + \frac{3k_2{}^2(4\theta - 19\theta^3)}{2a_o''^4\beta_o{}^8} + \frac{5k_4\theta(3 - 7\theta^2)}{2a_o''^4\beta_o{}^8} \right] n_o''(t - t_o)$$

$$\delta\omega = B^* C_3 (\cos\omega_o)(t - t_o)$$

$$\delta M = -\frac{2}{3}(q_o - s)^4 B^* \xi^4 \frac{a_E}{e_o\eta}[(1 + \eta\cos M_{DF})^3 - (1 + \eta\cos M_o)^3]$$

$$M_p = M_{DF} + \delta\omega + \delta M$$

$$\omega = \omega_{DF} - \delta\omega - \delta M$$

$$\Omega = \Omega_{DF} - \frac{21}{2}\frac{n_o'' k_2\theta}{a_o''^2\beta_o{}^2} C_1 (t - t_o)^2$$

$$e = e_o - B^* C_4 (t - t_o) - B^* C_5 (\sin M_p - \sin M_o)$$

$$a = a_o''[1 - C_1(t - t_o) - D_2(t - t_o)^2 - D_3(t - t_o)^3 - D_4(t - t_o)^4]^2$$

$$\begin{aligned} I\!L = \ & M_p + \omega + \Omega + n_o'' \left[\frac{3}{2} C_1 (t - t_o)^2 + (D_2 + 2C_1{}^2)(t - t_o)^3 \right. \\ & + \frac{1}{4}(3D_3 + 12C_1D_2 + 10C_1{}^3)(t - t_o)^4 \\ & \left. + \frac{1}{5}(3D_4 + 12C_1D_3 + 6D_2{}^2 + 30C_1{}^2D_2 + 15C_1{}^4)(t - t_o)^5 \right] \end{aligned}$$

$$\beta = \sqrt{(1 - e^2)}$$

$$n = k_e \Big/ a^{\frac{3}{2}}$$

where $(t - t_o)$ is time since epoch. It should be noted that when epoch perigee height is less than 220 kilometers, the equations for a and $I\!L$ are truncated after the C_1 term, and the terms involving C_5, $\delta\omega$, and δM are dropped.

Add the long-period periodic terms

$$a_{xN} = e\cos\omega$$

$$\mathit{I\!L}_L = \frac{A_{3,0}\sin i_o}{8k_2 a\beta^2}(e\cos\omega)\left(\frac{3+5\theta}{1+\theta}\right)$$

$$a_{yNL} = \frac{A_{3,0}\sin i_o}{4k_2 a\beta^2}$$

$$\mathit{I\!L}_T = \mathit{I\!L} + \mathit{I\!L}_L$$

$$a_{yN} = e\sin\omega + a_{yNL}.$$

Solve Kepler's equation for $(E+\omega)$ by defining

$$U = \mathit{I\!L}_T - \Omega$$

and using the iteration equation

$$(E+\omega)_{i+1} = (E+\omega)_i + \Delta(E+\omega)_i$$

with

$$\Delta(E+\omega)_i = \frac{U - a_{yN}\cos(E+\omega)_i + a_{xN}\sin(E+\omega)_i - (E+\omega)_i}{-a_{yN}\sin(E+\omega)_i - a_{xN}\cos(E+\omega)_i + 1}$$

and

$$(E+\omega)_1 = U.$$

The following equations are used to calculate preliminary quantities needed for short-period periodics.

$$e\cos E = a_{xN}\cos(E+\omega) + a_{yN}\sin(E+\omega)$$

$$e\sin E = a_{xN}\sin(E+\omega) - a_{yN}\cos(E+\omega)$$

$$e_L = ({a_{xN}}^2 + {a_{yN}}^2)^{\frac{1}{2}}$$

$$p_L = a(1 - {e_L}^2)$$

$$r = a(1 - e\cos E)$$

$$\dot{r} = k_e \frac{\sqrt{a}}{r} e \sin E$$

$$r\dot{f} = k_e \frac{\sqrt{p_L}}{r}$$

$$\cos u = \frac{a}{r}\left[\cos(E+\omega) - a_{xN} + \frac{a_{yN}(e \sin E)}{1+\sqrt{1-{e_L}^2}}\right]$$

$$\sin u = \frac{a}{r}\left[\sin(E+\omega) - a_{yN} - \frac{a_{xN}(e \sin E)}{1+\sqrt{1-{e_L}^2}}\right]$$

$$u = \tan^{-1}\left(\frac{\sin u}{\cos u}\right)$$

$$\Delta r = \frac{k_2}{2p_L}(1-\theta^2)\cos 2u$$

$$\Delta u = -\frac{k_2}{4{p_L}^2}(7\theta^2 - 1)\sin 2u$$

$$\Delta \Omega = \frac{3k_2\theta}{2{p_L}^2}\sin 2u$$

$$\Delta i = \frac{3k_2\theta}{2{p_L}^2}\sin i_o \cos 2u$$

$$\Delta \dot{r} = -\frac{k_2 n}{p_L}(1-\theta^2)\sin 2u$$

$$\Delta r\dot{f} = \frac{k_2 n}{p_L}\left[(1-\theta^2)\cos 2u - \frac{3}{2}(1-3\theta^2)\right]$$

The short-period periodics are added to give the osculating quantities

$$r_k = r\left[1 - \frac{3}{2}k_2\frac{\sqrt{1-{e_L}^2}}{{p_L}^2}(3\theta^2 - 1)\right] + \Delta r$$

$$u_k = u + \Delta u$$

$$\Omega_k = \Omega + \Delta\Omega$$

$$i_k = i_o + \Delta i$$

$$\dot{r}_k = \dot{r} + \Delta\dot{r}$$

$$r\dot{f}_k = r\dot{f} + \Delta r\dot{f}.$$

Then unit orientation vectors are calculated by

$$\mathbf{U} = \mathbf{M}\sin u_k + \mathbf{N}\cos u_k$$

$$\mathbf{V} = \mathbf{M}\cos u_k - \mathbf{N}\sin u_k$$

where

$$\mathbf{M} = \left\{ \begin{array}{l} M_x = -\sin\Omega_k \cos i_k \\ M_y = \cos\Omega_k \cos i_k \\ M_z = \sin i_k \end{array} \right\}$$

$$\mathbf{N} = \left\{ \begin{array}{l} N_x = \cos\Omega_k \\ N_y = \sin\Omega_k \\ N_z = 0 \end{array} \right\}.$$

Then position and velocity are given by

$$\mathbf{r} = r_k \mathbf{U}$$

and

$$\dot{\mathbf{r}} = \dot{r}_k \mathbf{U} + (r\dot{f})_k \mathbf{V}.$$

A FORTRAN IV computer code listing of the subroutine SGP4 is given below. These equations contain all currently anticipated changes to the SCC operational program. These changes are scheduled for implementation in March, 1981.

```
*      SGP4                                                 3 NOV 80
      SUBROUTINE SGP4(IFLAG,TSINCE)
      COMMON/E1/XMO,XNODEO,OMEGAO,EO,XINCL,XNO,XNDT2O,
     1           XNDD6O,BSTAR,X,Y,Z,XDOT,YDOT,ZDOT,EPOCH,DS50
      COMMON/C1/CK2,CK4,E6A,QOMS2T,S,TOTHRD,
     1           XJ3,XKE,XKMPER,XMNPDA,AE
      DOUBLE PRECISION EPOCH, DS50

      IF (IFLAG .EQ. 0) GO TO 100

*      RECOVER ORIGINAL MEAN MOTION (XNODP) AND SEMIMAJOR AXIS (AODP)
*      FROM INPUT ELEMENTS

      A1=(XKE/XNO)**TOTHRD
      COSIO=COS(XINCL)
      THETA2=COSIO*COSIO
      X3THM1=3.*THETA2-1.
      EOSQ=EO*EO
      BETAO2=1.-EOSQ
      BETAO=SQRT(BETAO2)
      DEL1=1.5*CK2*X3THM1/(A1*A1*BETAO*BETAO2)
      AO=A1*(1.-DEL1*(.5*TOTHRD+DEL1*(1.+134./81.*DEL1)))
      DELO=1.5*CK2*X3THM1/(AO*AO*BETAO*BETAO2)
      XNODP=XNO/(1.+DELO)
      AODP=AO/(1.-DELO)

*      INITIALIZATION

*      FOR PERIGEE LESS THAN 220 KILOMETERS, THE ISIMP FLAG IS SET AND
*      THE EQUATIONS ARE TRUNCATED TO LINEAR VARIATION IN SQRT A AND
*      QUADRATIC VARIATION IN MEAN ANOMALY.  ALSO, THE C3 TERM, THE
*      DELTA OMEGA TERM, AND THE DELTA M TERM ARE DROPPED.

      ISIMP=0
      IF((AODP*(1.-EO)/AE) .LT. (220./XKMPER+AE)) ISIMP=1

*      FOR PERIGEE BELOW 156 KM, THE VALUES OF
*      S AND QOMS2T ARE ALTERED

      S4=S
      QOMS24=QOMS2T
      PERIGE=(AODP*(1.-EO)-AE)*XKMPER
      IF(PERIGE .GE. 156.) GO TO 10
      S4=PERIGE-78.
      IF(PERIGE .GT. 98.) GO TO 9
      S4=20.
    9 QOMS24=((120.-S4)*AE/XKMPER)**4
      S4=S4/XKMPER+AE
```

```
10 PINVSQ=1./(AODP*AODP*BETAO2*BETAO2)
   TSI=1./(AODP-S4)
   ETA=AODP*EO*TSI
   ETASQ=ETA*ETA
   EETA=EO*ETA
   PSISQ=ABS(1.-ETASQ)
   COEF=QOMS24*TSI**4
   COEF1=COEF/PSISQ**3.5
   C2=COEF1*XNODP*(AODP*(1.+1.5*ETASQ+EETA*(4.+ETASQ))+.75*
  1         CK2*TSI/PSISQ*X3THM1*(8.+3.*ETASQ*(8.+ETASQ)))
   C1=BSTAR*C2
   SINIO=SIN(XINCL)
   A3OVK2=-XJ3/CK2*AE**3
   C3=COEF*TSI*A3OVK2*XNODP*AE*SINIO/EO
   X1MTH2=1.-THETA2
   C4=2.*XNODP*COEF1*AODP*BETAO2*(ETA*
  1         (2.+.5*ETASQ)+EO*(.5+2.*ETASQ)-2.*CK2*TSI/
  2         (AODP*PSISQ)*(-3.*X3THM1*(1.-2.*EETA+ETASQ*
  3         (1.5-.5*EETA))+.75*X1MTH2*(2.*ETASQ-EETA*
  4         (1.+ETASQ))*COS(2.*OMEGAO)))
   C5=2.*COEF1*AODP*BETAO2*(1.+2.75*(ETASQ+EETA)+EETA*ETASQ)
   THETA4=THETA2*THETA2
   TEMP1=3.*CK2*PINVSQ*XNODP
   TEMP2=TEMP1*CK2*PINVSQ
   TEMP3=1.25*CK4*PINVSQ*PINVSQ*XNODP
   XMDOT=XNODP+.5*TEMP1*BETAO*X3THM1+.0625*TEMP2*BETAO*
  1         (13.-78.*THETA2+137.*THETA4)
   X1M5TH=1.-5.*THETA2
   OMGDOT=-.5*TEMP1*X1M5TH+.0625*TEMP2*(7.-114.*THETA2+
  1         395.*THETA4)+TEMP3*(3.-36.*THETA2+49.*THETA4)
   XHDOT1=-TEMP1*COSIO
   XNODOT=XHDOT1+(.5*TEMP2*(4.-19.*THETA2)+2.*TEMP3*(3.-
  1         7.*THETA2))*COSIO
   OMGCOF=BSTAR*C3*COS(OMEGAO)
   XMCOF=-TOTHRD*COEF*BSTAR*AE/EETA
   XNODCF=3.5*BETAO2*XHDOT1*C1
   T2COF=1.5*C1
   XLCOF=.125*A3OVK2*SINIO*(3.+5.*COSIO)/(1.+COSIO)
   AYCOF=.25*A3OVK2*SINIO
   DELMO=(1.+ETA*COS(XMO))**3
   SINMO=SIN(XMO)
   X7THM1=7.*THETA2-1.
   IF(ISIMP .EQ. 1) GO TO 90
   C1SQ=C1*C1
   D2=4.*AODP*TSI*C1SQ
   TEMP=D2*TSI*C1/3.
   D3=(17.*AODP+S4)*TEMP
   D4=.5*TEMP*AODP*TSI*(221.*AODP+31.*S4)*C1
```

```
      T3COF=D2+2.*C1SQ
      T4COF=.25*(3.*D3+C1*(12.*D2+10.*C1SQ))
      T5COF=.2*(3.*D4+12.*C1*D3+6.*D2*D2+15.*C1SQ*(
     1         2.*D2+C1SQ))
   90 IFLAG=0

*      UPDATE FOR SECULAR GRAVITY AND ATMOSPHERIC DRAG

  100 XMDF=XMO+XMDOT*TSINCE
      OMGADF=OMEGAO+OMGDOT*TSINCE
      XNODDF=XNODEO+XNODOT*TSINCE
      OMEGA=OMGADF
      XMP=XMDF
      TSQ=TSINCE*TSINCE
      XNODE=XNODDF+XNODCF*TSQ
      TEMPA=1.-C1*TSINCE
      TEMPE=BSTAR*C4*TSINCE
      TEMPL=T2COF*TSQ
      IF(ISIMP .EQ. 1) GO TO 110
      DELOMG=OMGCOF*TSINCE
      DELM=XMCOF*((1.+ETA*COS(XMDF))**3-DELMO)
      TEMP=DELOMG+DELM
      XMP=XMDF+TEMP
      OMEGA=OMGADF-TEMP
      TCUBE=TSQ*TSINCE
      TFOUR=TSINCE*TCUBE
      TEMPA=TEMPA-D2*TSQ-D3*TCUBE-D4*TFOUR
      TEMPE=TEMPE+BSTAR*C5*(SIN(XMP)-SINMO)
      TEMPL=TEMPL+T3COF*TCUBE+
     1         TFOUR*(T4COF+TSINCE*T5COF)
  110 A=AODP*TEMPA**2
      E=EO-TEMPE
      XL=XMP+OMEGA+XNODE+XNODP*TEMPL
      BETA=SQRT(1.-E*E)
      XN=XKE/A**1.5

*      LONG PERIOD PERIODICS

      AXN=E*COS(OMEGA)
      TEMP=1./(A*BETA*BETA)
      XLL=TEMP*XLCOF*AXN
      AYNL=TEMP*AYCOF
      XLT=XL+XLL
      AYN=E*SIN(OMEGA)+AYNL

*      SOLVE KEPLERS EQUATION

      CAPU=FMOD2P(XLT-XNODE)
```

```
      TEMP2=CAPU
      DO 130 I=1,10
      SINEPW=SIN(TEMP2)
      COSEPW=COS(TEMP2)
      TEMP3=AXN*SINEPW
      TEMP4=AYN*COSEPW
      TEMP5=AXN*COSEPW
      TEMP6=AYN*SINEPW
      EPW=(CAPU-TEMP4+TEMP3-TEMP2)/(1.-TEMP5-TEMP6)+TEMP2
      IF(ABS(EPW-TEMP2) .LE. E6A) GO TO 140
  130 TEMP2=EPW

*      SHORT PERIOD PRELIMINARY QUANTITIES

  140 ECOSE=TEMP5+TEMP6
      ESINE=TEMP3-TEMP4
      ELSQ=AXN*AXN+AYN*AYN
      TEMP=1.-ELSQ
      PL=A*TEMP
      R=A*(1.-ECOSE)
      TEMP1=1./R
      RDOT=XKE*SQRT(A)*ESINE*TEMP1
      RFDOT=XKE*SQRT(PL)*TEMP1
      TEMP2=A*TEMP1
      BETAL=SQRT(TEMP)
      TEMP3=1./(1.+BETAL)
      COSU=TEMP2*(COSEPW-AXN+AYN*ESINE*TEMP3)
      SINU=TEMP2*(SINEPW-AYN-AXN*ESINE*TEMP3)
      U=ACTAN(SINU,COSU)
      SIN2U=2.*SINU*COSU
      COS2U=2.*COSU*COSU-1.
      TEMP=1./PL
      TEMP1=CK2*TEMP
      TEMP2=TEMP1*TEMP

*      UPDATE FOR SHORT PERIODICS

      RK=R*(1.-1.5*TEMP2*BETAL*X3THM1)+.5*TEMP1*X1MTH2*COS2U
      UK=U-.25*TEMP2*X7THM1*SIN2U
      XNODEK=XNODE+1.5*TEMP2*COSIO*SIN2U
      XINCK=XINCL+1.5*TEMP2*COSIO*SINIO*COS2U
      RDOTK=RDOT-XN*TEMP1*X1MTH2*SIN2U
      RFDOTK=RFDOT+XN*TEMP1*(X1MTH2*COS2U+1.5*X3THM1)

*      ORIENTATION VECTORS

      SINUK=SIN(UK)
      COSUK=COS(UK)
```

```
      SINIK=SIN(XINCK)
      COSIK=COS(XINCK)
      SINNOK=SIN(XNODEK)
      COSNOK=COS(XNODEK)
      XMX=-SINNOK*COSIK
      XMY=COSNOK*COSIK
      UX=XMX*SINUK+COSNOK*COSUK
      UY=XMY*SINUK+SINNOK*COSUK
      UZ=SINIK*SINUK
      VX=XMX*COSUK-COSNOK*SINUK
      VY=XMY*COSUK-SINNOK*SINUK
      VZ=SINIK*COSUK

*       POSITION AND VELOCITY

      X=RK*UX
      Y=RK*UY
      Z=RK*UZ
      XDOT=RDOTK*UX+RFDOTK*VX
      YDOT=RDOTK*UY+RFDOTK*VY
      ZDOT=RDOTK*UZ+RFDOTK*VZ

      RETURN
      END
```

7 THE SDP4 MODEL

The NORAD mean element sets can be used for prediction with SDP4. All symbols not defined below are defined in the list of symbols in Section Twelve. The original mean motion (n''_o) and semimajor axis (a''_o) are first recovered from the input elements by the equations

$$a_1 = \left(\frac{k_e}{n_o}\right)^{\frac{2}{3}}$$

$$\delta_1 = \frac{3}{2}\frac{k_2}{a_1{}^2}\frac{(3\cos^2 i_o - 1)}{(1 - e_o{}^2)^{\frac{3}{2}}}$$

$$a_o = a_1\left(1 - \frac{1}{3}\delta_1 - \delta_1{}^2 - \frac{134}{81}\delta_1{}^3\right)$$

$$\delta_o = \frac{3}{2}\frac{k_2}{a_o{}^2}\frac{(3\cos^2 i_o - 1)}{(1 - e_o{}^2)^{\frac{3}{2}}}$$

$$n''_o = \frac{n_o}{1 + \delta_o}$$

$$a''_o = \frac{a_o}{1 - \delta_o}.$$

For perigee between 98 kilometers and 156 kilometers, the value of the constant s used in SDP4 is changed to

$$s^* = a''_o(1 - e_o) - s + a_E.$$

For perigee below 98 kilometers, the value of s is changed to

$$s^* = 20/\mathrm{XKMPER} + a_E.$$

If the value of s is changed, then the value of $(q_o - s)^4$ must be replaced by

$$(q_o - s^*)^4 = \left[[(q_o - s)^4]^{\frac{1}{4}} + s - s^*\right]^4.$$

Then calculate the constants (using the appropriate values of s and $(q_o - s)^4$)

$$\theta = \cos i_o$$

$$\xi = \frac{1}{a_o'' - s}$$

$$\beta_o = (1 - {e_o}^2)^{\frac{1}{2}}$$

$$\eta = a_o'' e_o \xi$$

$$\begin{aligned} C_2 = \; & (q_o - s)^4 \xi^4 n_o'' (1 - \eta^2)^{-\frac{7}{2}} \left[a_o'' (1 + \frac{3}{2}\eta^2 + 4 e_o \eta + e_o \eta^3) \right. \\ & \left. + \frac{3}{2} \frac{k_2 \xi}{(1 - \eta^2)} \left(-\frac{1}{2} + \frac{3}{2}\theta^2 \right) (8 + 24\eta^2 + 3\eta^4) \right] \end{aligned}$$

$$C_1 = B^* C_2$$

$$\begin{aligned} C_4 = \; & 2 n_o'' (q_o - s)^4 \xi^4 a_o'' {\beta_o}^2 (1 - \eta^2)^{-\frac{7}{2}} \left(\left[2\eta (1 + e_o \eta) + \frac{1}{2} e_o + \frac{1}{2}\eta^3 \right] - \frac{2 k_2 \xi}{a_o'' (1 - \eta^2)} \times \right. \\ & \left. \left[3(1 - 3\theta^2) \left(1 + \frac{3}{2}\eta^2 - 2 e_o \eta - \frac{1}{2} e_o \eta^3 \right) + \frac{3}{4}(1 - \theta^2)(2\eta^2 - e_o \eta - e_o \eta^3) \cos 2\omega_o \right] \right) \end{aligned}$$

$$\dot{M} = \left[1 + \frac{3 k_2 (-1 + 3\theta^2)}{2 a_o''^2 {\beta_o}^3} + \frac{3 {k_2}^2 (13 - 78\theta^2 + 137\theta^4)}{16 a_o''^4 {\beta_o}^7} \right] n_o''$$

$$\dot{\omega} = \left[-\frac{3 k_2 (1 - 5\theta^2)}{2 a_o''^2 {\beta_o}^4} + \frac{3 {k_2}^2 (7 - 114\theta^2 + 395\theta^4)}{16 a_o''^4 {\beta_o}^8} + \frac{5 k_4 (3 - 36\theta^2 + 49\theta^4)}{4 a_o''^4 {\beta_o}^8} \right] n_o''$$

$$\dot{\Omega}_1 = -\frac{3 k_2 \theta}{a_o''^2 {\beta_o}^4} n_o''$$

$$\dot{\Omega} = \dot{\Omega}_1 + \left[\frac{3 {k_2}^2 (4\theta - 19\theta^3)}{2 a_o''^4 {\beta_o}^8} + \frac{5 k_4 \theta (3 - 7\theta^2)}{2 a_o''^4 {\beta_o}^8} \right] n_o''.$$

At this point SDP4 calls the initialization section of DEEP which calculates all initialized quantities needed for the deep-space perturbations (see Section Ten).

The secular effects of gravity are included by

$$M_{DF} = M_o + \dot{M}(t - t_o)$$

$$\omega_{DF} = \omega_o + \dot{\omega}(t - t_o)$$

$$\Omega_{DF} = \Omega_o + \dot{\Omega}(t - t_o)$$

where $(t - t_o)$ is time since epoch. The secular effect of drag on longitude of ascending node is included by

$$\Omega = \Omega_{DF} - \frac{21}{2}\frac{n_o'' k_2 \theta}{a_o''^2 \beta_o{}^2} C_1 (t - t_o)^2.$$

Next, SDP4 calls the secular section of DEEP which adds the deep-space secular effects and long-period resonance effects to the six classical orbital elements (see Section Ten).

The secular effects of drag are included in the remaining elements by

$$a = a_{DS}[1 - C_1(t - t_o)]^2$$

$$e = e_{DS} - B^* C_4 (t - t_o)$$

$$I\!\!L = M_{DS} + \omega_{DS} + \Omega_{DS} + n_o'' \left[\frac{3}{2} C_1 (t - t_o)^2\right]$$

where a_{DS}, e_{DS}, M_{DS}, ω_{DS}, and Ω_{DS}, are the values of n_o, e_o, M_{DF}, ω_{DF}, and Ω after deep-space secular and resonance perturbations have been applied.

Here SDP4 calls the periodics section of DEEP which adds the deep-space lunar and solar periodics to the orbital elements (see Section Ten). From this point on, it will be assumed that n, e, I, ω, Ω, and M are the mean motion, eccentricity, inclination, argument of perigee, longitude of ascending node, and mean anomaly after lunar-solar periodics have been added.

Add the long-period periodic terms

$$a_{xN} = e \cos \omega$$

$$\beta = \sqrt{(1 - e^2)}$$

$$I\!\!L_L = \frac{A_{3,0} \sin i_o}{8 k_2 a \beta^2} (e \cos \omega) \left(\frac{3 + 5\theta}{1 + \theta}\right)$$

$$a_{yNL} = \frac{A_{3,0} \sin i_o}{4 k_2 a \beta^2}$$

$$I\!\!L_T = I\!\!L + I\!\!L_L$$

$$a_{yN} = e \sin \omega + a_{yNL}.$$

Solve Kepler's equation for $(E + \omega)$ by defining

$$U = \mathbb{L}_T - \Omega$$

and using the iteration equation

$$(E + \omega)_{i+1} = (E + \omega)_i + \Delta(E + \omega)_i$$

with

$$\Delta(E + \omega)_i = \frac{U - a_{yN} \cos(E + \omega)_i + a_{xN} \sin(E + \omega)_i - (E + \omega)_i}{-a_{yN} \sin(E + \omega)_i - a_{xN} \cos(E + \omega)_i + 1}$$

and

$$(E + \omega)_1 = U.$$

The following equations are used to calculate preliminary quantities needed for short-period periodics.

$$e \cos E = a_{xN} \cos(E + \omega) + a_{yN} \sin(E + \omega)$$

$$e \sin E = a_{xN} \sin(E + \omega) - a_{yN} \cos(E + \omega)$$

$$e_L = ({a_{xN}}^2 + {a_{yN}}^2)^{\frac{1}{2}}$$

$$p_L = a(1 - {e_L}^2)$$

$$r = a(1 - e \cos E)$$

$$\dot{r} = k_e \frac{\sqrt{a}}{r} e \sin E$$

$$r\dot{f} = k_e \frac{\sqrt{p_L}}{r}$$

$$\cos u = \frac{a}{r} \left[\cos(E + \omega) - a_{xN} + \frac{a_{yN}(e \sin E)}{1 + \sqrt{1 - {e_L}^2}} \right]$$

$$\sin u = \frac{a}{r}\left[\sin(E+\omega) - a_{yN} - \frac{a_{xN}(e\sin E)}{1+\sqrt{1-{e_L}^2}}\right]$$

$$u = \tan^{-1}\left(\frac{\sin u}{\cos u}\right)$$

$$\Delta r = \frac{k_2}{2p_L}(1-\theta^2)\cos 2u$$

$$\Delta u = -\frac{k_2}{4{p_L}^2}(7\theta^2-1)\sin 2u$$

$$\Delta\Omega = \frac{3k_2\theta}{2{p_L}^2}\sin 2u$$

$$\Delta i = \frac{3k_2\theta}{2{p_L}^2}\sin i_o \cos 2u$$

$$\Delta\dot{r} = -\frac{k_2 n}{p_L}(1-\theta^2)\sin 2u$$

$$\Delta r\dot{f} = \frac{k_2 n}{p_L}\left[(1-\theta^2)\cos 2u - \frac{3}{2}(1-3\theta^2)\right]$$

The short-period periodics are added to give the osculating quantities

$$r_k = r\left[1 - \frac{3}{2}k_2\frac{\sqrt{1-{e_L}^2}}{{p_L}^2}(3\theta^2-1)\right] + \Delta r$$

$$u_k = u + \Delta u$$

$$\Omega_k = \Omega + \Delta\Omega$$

$$i_k = I + \Delta i$$

$$\dot{r}_k = \dot{r} + \Delta\dot{r}$$

$$r\dot{f}_k = r\dot{f} + \Delta r\dot{f}.$$

Then unit orientation vectors are calculated by

$$\mathbf{U} = \mathbf{M}\sin u_k + \mathbf{N}\cos u_k$$

$$\mathbf{V} = \mathbf{M}\cos u_k - \mathbf{N}\sin u_k$$

where

$$\mathbf{M} = \left\{ \begin{array}{l} M_x = -\sin\Omega_k \cos i_k \\ M_y = \cos\Omega_k \cos i_k \\ M_z = \sin i_k \end{array} \right\}$$

$$\mathbf{N} = \left\{ \begin{array}{l} N_x = \cos\Omega_k \\ N_y = \sin\Omega_k \\ N_z = 0 \end{array} \right\}.$$

Then position and velocity are given by

$$\mathbf{r} = r_k \mathbf{U}$$

and

$$\dot{\mathbf{r}} = \dot{r}_k \mathbf{U} + (r\dot{f})_k \mathbf{V}.$$

A FORTRAN IV computer code listing of the subroutine SDP4 is given below. These equations contain all currently anticipated changes to the SCC operational program. These changes are scheduled for implementation in March, 1981.

```
*      SDP4                                                 3 NOV 80
      SUBROUTINE SDP4(IFLAG,TSINCE)
      COMMON/E1/XMO,XNODEO,OMEGAO,EO,XINCL,XNO,XNDT2O,
     1           XNDD6O,BSTAR,X,Y,Z,XDOT,YDOT,ZDOT,EPOCH,DS50
      COMMON/C1/CK2,CK4,E6A,QOMS2T,S,TOTHRD,
     1           XJ3,XKE,XKMPER,XMNPDA,AE
      DOUBLE PRECISION EPOCH, DS50

      IF  (IFLAG .EQ. 0) GO TO 100

*      RECOVER ORIGINAL MEAN MOTION (XNODP) AND SEMIMAJOR AXIS (AODP)
*      FROM INPUT ELEMENTS

      A1=(XKE/XNO)**TOTHRD
      COSIO=COS(XINCL)
      THETA2=COSIO*COSIO
      X3THM1=3.*THETA2-1.
      EOSQ=EO*EO
      BETAO2=1.-EOSQ
      BETAO=SQRT(BETAO2)
      DEL1=1.5*CK2*X3THM1/(A1*A1*BETAO*BETAO2)
      AO=A1*(1.-DEL1*(.5*TOTHRD+DEL1*(1.+134./81.*DEL1)))
      DELO=1.5*CK2*X3THM1/(AO*AO*BETAO*BETAO2)
      XNODP=XNO/(1.+DELO)
      AODP=AO/(1.-DELO)

*      INITIALIZATION

*      FOR PERIGEE BELOW 156 KM, THE VALUES OF
*      S AND QOMS2T ARE ALTERED

      S4=S
      QOMS24=QOMS2T
      PERIGE=(AODP*(1.-EO)-AE)*XKMPER
      IF(PERIGE .GE. 156.) GO TO 10
      S4=PERIGE-78.
      IF(PERIGE .GT. 98.) GO TO 9
      S4=20.
    9 QOMS24=((120.-S4)*AE/XKMPER)**4
      S4=S4/XKMPER+AE
   10 PINVSQ=1./(AODP*AODP*BETAO2*BETAO2)
      SING=SIN(OMEGAO)
      COSG=COS(OMEGAO)
      TSI=1./(AODP-S4)
      ETA=AODP*EO*TSI
      ETASQ=ETA*ETA
      EETA=EO*ETA
      PSISQ=ABS(1.-ETASQ)
```

```
      COEF=QOMS24*TSI**4
      COEF1=COEF/PSISQ**3.5
      C2=COEF1*XNODP*(AODP*(1.+1.5*ETASQ+EETA*(4.+ETASQ))+.75*
     1         CK2*TSI/PSISQ*X3THM1*(8.+3.*ETASQ*(8.+ETASQ)))
      C1=BSTAR*C2
      SINIO=SIN(XINCL)
      A3OVK2=-XJ3/CK2*AE**3
      X1MTH2=1.-THETA2
      C4=2.*XNODP*COEF1*AODP*BETAO2*(ETA*
     1         (2.+.5*ETASQ)+EO*(.5+2.*ETASQ)-2.*CK2*TSI/
     2         (AODP*PSISQ)*(-3.*X3THM1*(1.-2.*EETA+ETASQ*
     3         (1.5-.5*EETA))+.75*X1MTH2*(2.*ETASQ-EETA*
     4         (1.+ETASQ))*COS(2.*OMEGAO)))
      THETA4=THETA2*THETA2
      TEMP1=3.*CK2*PINVSQ*XNODP
      TEMP2=TEMP1*CK2*PINVSQ
      TEMP3=1.25*CK4*PINVSQ*PINVSQ*XNODP
      XMDOT=XNODP+.5*TEMP1*BETAO*X3THM1+.0625*TEMP2*BETAO*
     1         (13.-78.*THETA2+137.*THETA4)
      X1M5TH=1.-5.*THETA2
      OMGDOT=-.5*TEMP1*X1M5TH+.0625*TEMP2*(7.-114.*THETA2+
     1         395.*THETA4)+TEMP3*(3.-36.*THETA2+49.*THETA4)
      XHDOT1=-TEMP1*COSIO
      XNODOT=XHDOT1+(.5*TEMP2*(4.-19.*THETA2)+2.*TEMP3*(3.-
     1         7.*THETA2))*COSIO
      XNODCF=3.5*BETAO2*XHDOT1*C1
      T2COF=1.5*C1
      XLCOF=.125*A3OVK2*SINIO*(3.+5.*COSIO)/(1.+COSIO)
      AYCOF=.25*A3OVK2*SINIO
      X7THM1=7.*THETA2-1.
   90 IFLAG=0
      CALL DPINIT(EOSQ,SINIO,COSIO,BETAO,AODP,THETA2,
     1         SING,COSG,BETAO2,XMDOT,OMGDOT,XNODOT,XNODP)

*      UPDATE FOR SECULAR GRAVITY AND ATMOSPHERIC DRAG

  100 XMDF=XMO+XMDOT*TSINCE
      OMGADF=OMEGAO+OMGDOT*TSINCE
      XNODDF=XNODEO+XNODOT*TSINCE
      TSQ=TSINCE*TSINCE
      XNODE=XNODDF+XNODCF*TSQ
      TEMPA=1.-C1*TSINCE
      TEMPE=BSTAR*C4*TSINCE
      TEMPL=T2COF*TSQ
      XN=XNODP
      CALL DPSEC(XMDF,OMGADF,XNODE,EM,XINC,XN,TSINCE)
      A=(XKE/XN)**TOTHRD*TEMPA**2
      E=EM-TEMPE
```

```
      XMAM=XMDF+XNODP*TEMPL
      CALL DPPER(E,XINC,OMGADF,XNODE,XMAM)
      XL=XMAM+OMGADF+XNODE
      BETA=SQRT(1.-E*E)
      XN=XKE/A**1.5

*      LONG PERIOD PERIODICS

      AXN=E*COS(OMGADF)
      TEMP=1./(A*BETA*BETA)
      XLL=TEMP*XLCOF*AXN
      AYNL=TEMP*AYCOF
      XLT=XL+XLL
      AYN=E*SIN(OMGADF)+AYNL

*      SOLVE KEPLERS EQUATION

      CAPU=FMOD2P(XLT-XNODE)
      TEMP2=CAPU
      DO 130 I=1,10
      SINEPW=SIN(TEMP2)
      COSEPW=COS(TEMP2)
      TEMP3=AXN*SINEPW
      TEMP4=AYN*COSEPW
      TEMP5=AXN*COSEPW
      TEMP6=AYN*SINEPW
      EPW=(CAPU-TEMP4+TEMP3-TEMP2)/(1.-TEMP5-TEMP6)+TEMP2
      IF(ABS(EPW-TEMP2) .LE. E6A) GO TO 140
  130 TEMP2=EPW

*      SHORT PERIOD PRELIMINARY QUANTITIES

  140 ECOSE=TEMP5+TEMP6
      ESINE=TEMP3-TEMP4
      ELSQ=AXN*AXN+AYN*AYN
      TEMP=1.-ELSQ
      PL=A*TEMP
      R=A*(1.-ECOSE)
      TEMP1=1./R
      RDOT=XKE*SQRT(A)*ESINE*TEMP1
      RFDOT=XKE*SQRT(PL)*TEMP1
      TEMP2=A*TEMP1
      BETAL=SQRT(TEMP)
      TEMP3=1./(1.+BETAL)
      COSU=TEMP2*(COSEPW-AXN+AYN*ESINE*TEMP3)
      SINU=TEMP2*(SINEPW-AYN-AXN*ESINE*TEMP3)
      U=ACTAN(SINU,COSU)
      SIN2U=2.*SINU*COSU
```

```
      COS2U=2.*COSU*COSU-1.
      TEMP=1./PL
      TEMP1=CK2*TEMP
      TEMP2=TEMP1*TEMP

*       UPDATE FOR SHORT PERIODICS

      RK=R*(1.-1.5*TEMP2*BETAL*X3THM1)+.5*TEMP1*X1MTH2*COS2U
      UK=U-.25*TEMP2*X7THM1*SIN2U
      XNODEK=XNODE+1.5*TEMP2*COSIO*SIN2U
      XINCK=XINC+1.5*TEMP2*COSIO*SINIO*COS2U
      RDOTK=RDOT-XN*TEMP1*X1MTH2*SIN2U
      RFDOTK=RFDOT+XN*TEMP1*(X1MTH2*COS2U+1.5*X3THM1)

*       ORIENTATION VECTORS

      SINUK=SIN(UK)
      COSUK=COS(UK)
      SINIK=SIN(XINCK)
      COSIK=COS(XINCK)
      SINNOK=SIN(XNODEK)
      COSNOK=COS(XNODEK)
      XMX=-SINNOK*COSIK
      XMY=COSNOK*COSIK
      UX=XMX*SINUK+COSNOK*COSUK
      UY=XMY*SINUK+SINNOK*COSUK
      UZ=SINIK*SINUK
      VX=XMX*COSUK-COSNOK*SINUK
      VY=XMY*COSUK-SINNOK*SINUK
      VZ=SINIK*COSUK

*       POSITION AND VELOCITY

      X=RK*UX
      Y=RK*UY
      Z=RK*UZ
      XDOT=RDOTK*UX+RFDOTK*VX
      YDOT=RDOTK*UY+RFDOTK*VY
      ZDOT=RDOTK*UZ+RFDOTK*VZ

      RETURN
      END
```

8 THE SGP8 MODEL

The NORAD mean element sets can be used for prediction with SGP8. All symbols not defined below are defined in the list of symbols in Section Twelve. The original mean motion (n_o'') and semimajor axis (a_o'') are first recovered from the input elements by the equations

$$a_1 = \left(\frac{k_e}{n_o}\right)^{\frac{2}{3}}$$

$$\delta_1 = \frac{3}{2}\frac{k_2}{a_1{}^2}\frac{(3\cos^2 i_o - 1)}{(1 - e_o{}^2)^{\frac{3}{2}}}$$

$$a_o = a_1\left(1 - \frac{1}{3}\delta_1 - \delta_1{}^2 - \frac{134}{81}\delta_1{}^3\right)$$

$$\delta_o = \frac{3}{2}\frac{k_2}{a_o{}^2}\frac{(3\cos^2 i_o - 1)}{(1 - e_o{}^2)^{\frac{3}{2}}}$$

$$n_o'' = \frac{n_o}{1 + \delta_o}$$

$$a_o'' = \frac{a_o}{1 - \delta_o}.$$

The ballistic coefficient (B term) is then calculated from the B^* drag term by

$$B = 2B^*/\rho_o$$

where

$$\rho_o = (2.461 \times 10^{-5})\ \text{XKMPER kg/m}^2/\text{Earth radii}$$

is a reference value of atmospheric density.

Then calculate the constants

$$\beta^2 = 1 - e^2$$

$$\theta = \cos i$$

$$\dot{M}_1 = -\frac{3}{2}\frac{n'' k_2}{a''^2 \beta^3}(1 - 3\theta^2)$$

$$\dot{\omega}_1 = -\frac{3}{2}\frac{n'' k_2}{a''^2 \beta^4}(1 - 5\theta^2)$$

$$\dot{\Omega}_1 = -3\frac{n'' k_2}{a''^2 \beta^4}\theta$$

$$\dot{M}_2 = \frac{3}{16}\frac{n'' {k_2}^2}{a''^4 \beta^7}(13 - 78\theta^2 + 137\theta^4)$$

$$\dot{\omega}_2 = \frac{3}{16}\frac{n'' k_2 2}{a''^4 \beta^8}(7 - 114\theta^2 + 395\theta^4) + \frac{5}{4}\frac{n'' k_4}{a''^4 \beta^8}(3 - 36\theta^2 + 49\theta^4)$$

$$\dot{\Omega}_2 = \frac{3}{2}\frac{n'' {k_2}^2}{a''^4 \beta^8}\theta(4 - 19\theta^2) + \frac{5}{2}\frac{n'' k_4}{a''^4 \beta^8}\theta(3 - 7\theta^2)$$

$$\dot{\ell} = n'' + \dot{M}_1 + \dot{M}_2$$

$$\dot{\omega} = \dot{\omega}_1 + \dot{\omega}_2$$

$$\dot{\Omega} = \dot{\Omega}_1 + \dot{\Omega}_2$$

$$\xi = \frac{1}{a''\beta^2 - s}$$

$$\eta = es\xi$$

$$\psi = \sqrt{1 - \eta^2}$$

$$\alpha^2 = 1 + e^2$$

$$C_o = \frac{1}{2}B\rho_o(q_o - s)^4 n'' a'' \xi^4 \alpha^{-1} \psi^{-7}$$

$$C_1 = \frac{3}{2}n'' \alpha^4 C_o$$

$$D_1 = \xi\psi^{-2}/a''\beta^2$$

$$D_2 = 12 + 36\eta^2 + \frac{9}{2}\eta^4$$

$$D_3 = 15\eta^2 + \frac{5}{2}\eta^4$$

$$D_4 = 5\eta + \frac{15}{4}\eta^3$$

$$D_5 = \xi\psi^{-2}$$

$$B_1 = -k_2(1 - 3\theta^2)$$

$$B_2 = -k_2(1 - \theta^2)$$

$$B_3 = \frac{A_{3,0}}{k_2}\sin i$$

$$C_2 = D_1 D_3 B_2$$

$$C_3 = D_4 D_5 B_3$$

$$\dot{n}_o = C_1\left(2 + 3\eta^2 + 20e\eta + 5e\eta^3 + \frac{17}{2}e^2 + 34e^2\eta^2 + D_1 D_2 B_1 + C_2\cos 2\omega + C_3\sin\omega\right)$$

$$C_4 = D_1 D_7 B_2$$

$$C_5 = D_5 D_8 B_3$$

$$D_6 = 30\eta + \frac{45}{2}\eta^3$$

$$D_7 = 5\eta + \frac{25}{2}\eta^3$$

$$D_8 = 1 + \frac{27}{4}\eta^2 + \eta^4$$

$$\dot{e}_o = -C_o\left(4\eta + \eta^3 + 5e + 15e\eta^2 + \frac{31}{2}e^2\eta + 7e^2\eta^3 + D_1D_6B_1 + C_4\cos 2\omega + C_5\sin\omega\right)$$

$$\dot{\alpha}/\alpha = e\dot{e}\alpha^{-2}$$

$$C_6 = \frac{1}{3}\frac{\dot{n}}{n''}$$

$$\dot{\xi}/\xi = 2a''\xi(C_6\beta^2 + e\dot{e})$$

$$\dot{\eta} = (\dot{e} + e\dot{\xi}/\xi)s\xi$$

$$\dot{\psi}/\psi = -\eta\dot{\eta}\psi^{-2}$$

$$\dot{C}_o/C_o = C_6 + 4\dot{\xi}/\xi - \dot{\alpha}/\alpha - 7\dot{\psi}/\psi$$

$$\dot{C}_1/C_1 = \dot{n}/n'' + 4\dot{\alpha}/\alpha + \dot{C}_o/C_o$$

$$D_9 = 6\eta + 20e + 15e\eta^2 + 68e^2\eta$$

$$D_{10} = 20\eta + 5\eta^3 + 17e + 68e\eta^2$$

$$D_{11} = 72\eta + 18\eta^3$$

$$D_{12} = 30\eta + 10\eta^3$$

$$D_{13} = 5 + \frac{45}{4}\eta^2$$

$$D_{14} = \dot{\xi}/\xi - 2\dot{\psi}/\psi$$

$$D_{15} = 2(C_6 + e\dot{e}\beta^{-2})$$

$$\dot{D}_1 = D_1(D_{14} + D_{15})$$

$$\dot{D}_2 = \dot{\eta}D_{11}$$

$$\dot{D}_3 = \dot{\eta} D_{12}$$

$$\dot{D}_4 = \dot{\eta} D_{13}$$

$$\dot{D}_5 = D_5 D_{14}$$

$$\dot{C}_2 = B_2(\dot{D}_1 D_3 + D_1 \dot{D}_3)$$

$$\dot{C}_3 = B_3(\dot{D}_5 D_4 + D_5 \dot{D}_4)$$

$$\dot{\omega} = -\frac{3}{2}\frac{n'' k_2}{a''^2 \beta^4}(1 - 5\theta^2)$$

$$D_{16} = D_9 \dot{\eta} + D_{10} \dot{e} + B_1(\dot{D}_1 D_2 + D_1 \dot{D}_2) + \dot{C}_2 \cos 2\omega + \dot{C}_3 \sin \omega + \dot{\omega}(C_3 \cos \omega - 2C_2 \sin 2\omega)$$

$$\ddot{n}_o = \dot{n}\dot{C}_1 / C_1 + C_1 D_{16}$$

$$\begin{aligned} \ddot{e}_o = \; & \dot{e}\dot{C}_o / C_o - C_o \left\{ \left(4 + 3\eta^2 + 30 e\eta + \frac{31}{2} e^2 + 21 e^2 \eta^2\right) \dot{\eta} + (5 + 15\eta^2 + 31 e\eta + 14 e\eta^3)\dot{e} \right. \\ & + B_1 \left[\dot{D}_1 D_6 + D_1 \dot{\eta} \left(30 + \frac{135}{2}\eta^2\right)\right] + B_2 \left[\dot{D}_1 D_7 + D_1 \dot{\eta} \left(5 + \frac{75}{2}\eta^2\right)\right] \cos \omega \\ & \left. + B_3 \left[\dot{D}_5 D_8 + D_5 \eta \dot{\eta} \left(\frac{27}{2} + 4\eta^2\right)\right] \sin \omega + \dot{\omega}(C_5 \cos \omega - 2C_4 \sin 2\omega) \right\} \end{aligned}$$

$$D_{17} = \ddot{n}/n'' - (\dot{n}/n'')^2$$

$$\ddot{\xi}/\xi = 2(\dot{\xi}/\xi - C_6)\dot{\xi}/\xi + 2a''\xi \left(\frac{1}{3} D_{17} \beta^2 - 2C_6 e\dot{e} + \dot{e}^2 + e\ddot{e}\right)$$

$$\ddot{\eta} = (\ddot{e} + 2\dot{e}\dot{\xi}/\xi) s\xi + \eta \ddot{\xi}/\xi$$

$$D_{18} = \ddot{\xi}/\xi - (\dot{\xi}/\xi)^2$$

$$D_{19} = -(\dot{\psi}/\psi)^2 (1 + \eta^{-2}) - \eta \ddot{\eta} \psi^{-2}$$

$$\ddot{D}_1 = \dot{D}_1 (D_{14} + D_{15}) + D_1 \left(D_{18} - 2D_{19} + \frac{2}{3} D_{17} + 2\alpha^2 \dot{e}^2 \beta^{-4} + 2e\ddot{e}\beta^{-2}\right)$$

$$
\begin{aligned}
\dddot{n}_o = \;& \dot{n}\left[\frac{4}{3}D_{17} + 3\dot{e}^2\alpha^{-2} + 3e\ddot{e}\alpha^{-2} - 6(\dot{\alpha}/\alpha)^2 + 4D_{18} - 7D_{19}\right] \\
& +\ddot{n}\dot{C}_1/C_1 + C_1\Big\{D_{16}\dot{C}_1/C_1 + D_9\ddot{\eta} + D_{10}\ddot{e} + \dot{\eta}^2(6 + 30e\eta + 68e^2) \\
& +\dot{\eta}\dot{e}(40 + 30\eta^2 + 272e\eta) + \dot{e}^2(17 + 68\eta^2) \\
& +B_1[\ddot{D}_1D_2 + 2\dot{D}_1\dot{D}_2 + D_1(\ddot{\eta}D_{11} + \dot{\eta}^2(72 + 54\eta^2))] \\
& +B_2[\ddot{D}_1D_3 + 2\dot{D}_1\dot{D}_3 + D_1(\ddot{\eta}D_{12} + \dot{\eta}^2(30 + 30\eta^2))]\cos 2\omega \\
& +B_3\left[(\dot{D}_5D_{14} + D_5(D_{18} - 2D_{19}))D_4 + 2\dot{D}_4\dot{D}_5 + D_5\left(\ddot{\eta}D_{13} + \frac{45}{2}\eta\dot{\eta}^2\right)\right]\sin\omega \\
& +\dot{\omega}[(7C_6 + 4e\dot{e}\beta^{-2})(C_3\cos\omega - 2C_2\sin 2\omega) + 2C_3\cos\omega \\
& -4C_2\sin 2\omega - \dot{\omega}(C_3\sin\omega + 4C_2\cos 2\omega)]\Big\}
\end{aligned}
$$

$$p = \frac{2\ddot{n}_o{}^2 - \dot{n}_o\,\dddot{n}_o}{\ddot{n}_o{}^2 - \dot{n}_o\,\dddot{n}_o}$$

$$\gamma = -\frac{\dddot{n}_o}{\ddot{n}_o}\frac{1}{(p-2)}$$

$$n_D = \frac{\dot{n}_o}{p\gamma}$$

$$q = 1 - \frac{\ddot{e}_o}{\dot{e}_o\gamma}$$

$$e_D = \frac{\dot{e}_o}{q\gamma}$$

where all quantities are epoch values.

The secular effects of atmospheric drag and gravitation are included by

$$n = n_o'' + n_D[1 - (1 - \gamma(t - t_o))^p]$$

$$e = e_o + e_D[1 - (1 - \gamma(t - t_o))^q]$$

$$\omega = \omega_o + \dot{\omega}_1\left[(t - t_o) + \frac{7}{3}\frac{1}{n_o''}Z_1\right] + \dot{\omega}_2(t - t_o)$$

$$\Omega = \Omega_o'' + \dot{\Omega}_1 \left[(t - t_o) + \frac{7}{3} \frac{1}{n_o''} Z_1 \right] + \dot{\Omega}_2 (t - t_o)$$

$$M = M_o + n_o''(t - t_o) + Z_1 + \dot{M}_1 \left[(t - t_o) + \frac{7}{3} \frac{1}{n_o''} Z_1 \right] + \dot{M}_2 (t - t_o)$$

where

$$Z_1 = \frac{\dot{n}_o}{p\gamma} \left\{ (t - t_o) + \frac{1}{\gamma(p+1)} [(1 - \gamma(t - t_o))^{p+1} - 1] \right\}.$$

If drag is very small ($\frac{\dot{n}}{n_o''}$ less than 1.5×10^{-6}/min) then the secular equations for n, e, and Z_1 should be replaced by

$$n = n_o'' + \dot{n}(t - t_o)$$

$$e = e_o'' + \dot{e}(t - t_o)$$

$$Z_1 = \frac{1}{2} \dot{n}_o (t - t_o)^2$$

where $(t - t_o)$ is time since epoch and where

$$\dot{e} = -\frac{2}{3} \frac{\dot{n}_o}{n_o''} (1 - e_o).$$

Solve Kepler's equation for E by using the iteration equation

$$E_{i+1} = E_i + \Delta E_i$$

with

$$\Delta E_i = \frac{M + e \sin E_i - E_i}{1 - e \cos E_i}$$

and

$$E_1 = M + e \sin M + \frac{1}{2} e^2 \sin 2M.$$

The following equations are used to calculate preliminary quantities needed for the short-period periodics.

$$a = \left(\frac{k_e}{n}\right)^{\frac{2}{3}}$$

$$\beta = (1 - e^2)^{\frac{1}{2}}$$

$$\sin f = \frac{\beta \sin E}{1 - e \cos E}$$

$$\cos f = \frac{\cos E - e}{1 - e \cos E}$$

$$u = f + \omega$$

$$r'' = \frac{a\beta^2}{1 + e \cos f}$$

$$\dot{r}'' = \frac{nae}{\beta} \sin f$$

$$(r\dot{f})'' = \frac{na^2\beta}{r}$$

$$\delta r = \frac{1}{2}\frac{k_2}{a\beta^2}[(1 - \theta^2)\cos 2u + 3(1 - 3\theta^2)] - \frac{1}{4}\frac{A_{3,0}}{k_2}\sin i_o \sin u$$

$$\delta \dot{r} = -n\left(\frac{a}{r}\right)^2 \left[\frac{k_2}{a\beta^2}(1 - \theta^2)\sin 2u + \frac{1}{4}\frac{A_{3,0}}{k_2}\sin i_o \cos u\right]$$

$$\delta I = \theta \left[\frac{3}{2}\frac{k_2}{a^2\beta^4}\sin i_o \cos 2u - \frac{1}{4}\frac{A_{3,0}}{k_2 a\beta^2} e \sin \omega\right]$$

$$\delta(r\dot{f}) = -n\left(\frac{a}{r}\right)^2 \delta r + na\left(\frac{a}{r}\right)\frac{\sin i_o}{\theta}\delta I$$

$$\begin{aligned}\delta u = \; & \frac{1}{2}\frac{k_2}{a^2\beta^4}\left[\frac{1}{2}(1 - 7\theta^2)\sin 2u - 3(1 - 5\theta^2)(f - M + e \sin f)\right] \\ & -\frac{1}{4}\frac{A_{3,0}}{k_2 a\beta^2}\left[\sin i_o \cos u(2 + e \cos f) + \frac{1}{2}\frac{\theta^2}{\sin i_o/2 \;\cos i_o/2} e \cos \omega\right]\end{aligned}$$

$$
\begin{aligned}
\delta\lambda = \; & \frac{1}{2}\frac{k_2}{a^2\beta^4}\left[\frac{1}{2}(1+6\theta-7\theta^2)\sin 2u - 3(1+2\theta-5\theta^2)(f-M+e\sin f)\right] \\
& +\frac{1}{4}\frac{A_{3,0}}{k_2 a\beta^2}\sin i_o\left[\frac{e\theta}{1+\theta}\cos\omega-(2+e\cos f)\cos u\right]
\end{aligned}
$$

The short-period periodics are added to give the osculating quantities

$$r = r'' + \delta r$$

$$\dot{r} = \dot{r}'' + \delta\dot{r}$$

$$r\dot{f} = (r\dot{f})'' + \delta(r\dot{f})$$

$$y_4 = \sin\frac{i_o}{2}\sin u + \cos u \sin\frac{i_o}{2}\delta u + \frac{1}{2}\sin u\cos\frac{i_o}{2}\delta I$$

$$y_5 = \sin\frac{i_o}{2}\cos u - \sin u \sin\frac{i_o}{2}\delta u + \frac{1}{2}\cos u\cos\frac{i_o}{2}\delta I$$

$$\lambda = u + \Omega + \delta\lambda.$$

Unit orientation vectors are calculated by

$$U_x = 2y_4(y_5\sin\lambda - y_4\cos\lambda) + cos\lambda$$

$$U_y = -2y_4(y_5\cos\lambda + y_4\sin\lambda) + sin\lambda$$

$$U_z = 2y_4\cos\frac{I}{2}$$

$$V_x = 2y_5(y_5\sin\lambda - y_4\cos\lambda) - sin\lambda$$

$$V_y = -2y_5(y_5\cos\lambda + y_4\sin\lambda) + cos\lambda$$

$$V_z = 2y_5\cos\frac{I}{2}$$

where

$$\cos\frac{I}{2} = \sqrt{1 - {y_4}^2 - {y_5}^2}.$$

Position and velocity are given by

$$\mathbf{r} = r\mathbf{U}$$

$$\dot{\mathbf{r}} = \dot{r}\mathbf{U} + r\dot{f}\mathbf{V}.$$

A FORTRAN IV computer code listing of the subroutine SGP8 is given below.

```
*      SGP8                                             14 NOV 80
      SUBROUTINE SGP8(IFLAG,TSINCE)
      COMMON/E1/XMO,XNODEO,OMEGAO,EO,XINCL,XNO,XNDT2O,
     1           XNDD6O,BSTAR,X,Y,Z,XDOT,YDOT,ZDOT,EPOCH,DS50
      COMMON/C1/CK2,CK4,E6A,QOMS2T,S,TOTHRD,
     1           XJ3,XKE,XKMPER,XMNPDA,AE
      DOUBLE PRECISION EPOCH, DS50
      DATA RHO/.15696615/

      IF (IFLAG .EQ. 0) GO TO 100

*      RECOVER ORIGINAL MEAN MOTION (XNODP) AND SEMIMAJOR AXIS (AODP)
*      FROM INPUT ELEMENTS --------- CALCULATE BALLISTIC COEFFICIENT
*      (B TERM) FROM INPUT B* DRAG TERM

      A1=(XKE/XNO)**TOTHRD
      COSI=COS(XINCL)
      THETA2=COSI*COSI
      TTHMUN=3.*THETA2-1.
      EOSQ=EO*EO
      BETAO2=1.-EOSQ
      BETAO=SQRT(BETAO2)
      DEL1=1.5*CK2*TTHMUN/(A1*A1*BETAO*BETAO2)
      AO=A1*(1.-DEL1*(.5*TOTHRD+DEL1*(1.+134./81.*DEL1)))
      DELO=1.5*CK2*TTHMUN/(AO*AO*BETAO*BETAO2)
      AODP=AO/(1.-DELO)
      XNODP=XNO/(1.+DELO)
      B=2.*BSTAR/RHO

*      INITIALIZATION

      ISIMP=0
      PO=AODP*BETAO2
      POM2=1./(PO*PO)
      SINI=SIN(XINCL)
      SING=SIN(OMEGAO)
      COSG=COS(OMEGAO)
      TEMP=.5*XINCL
      SINIO2=SIN(TEMP)
      COSIO2=COS(TEMP)
      THETA4=THETA2**2
      UNM5TH=1.-5.*THETA2
      UNMTH2=1.-THETA2
      A3COF=-XJ3/CK2*AE**3
      PARDT1=3.*CK2*POM2*XNODP
      PARDT2=PARDT1*CK2*POM2
      PARDT4=1.25*CK4*POM2*POM2*XNODP
      XMDT1=.5*PARDT1*BETAO*TTHMUN
```

```
      XGDT1=-.5*PARDT1*UNM5TH
      XHDT1=-PARDT1*COSI
      XLLDOT=XNODP+XMDT1+
     2           .0625*PARDT2*BETAO*(13.-78.*THETA2+137.*THETA4)
      OMGDT=XGDT1+
     1      .0625*PARDT2*(7.-114.*THETA2+395.*THETA4)+PARDT4*(3.-36.*
     2         THETA2+49.*THETA4)
      XNODOT=XHDT1+
     1       (.5*PARDT2*(4.-19.*THETA2)+2.*PARDT4*(3.-7.*THETA2))*COSI
      TSI=1./(PO-S)
      ETA=EO*S*TSI
      ETA2=ETA**2
      PSIM2=ABS(1./(1.-ETA2))
      ALPHA2=1.+EOSQ
      EETA=EO*ETA
      COS2G=2.*COSG**2-1.
      D5=TSI*PSIM2
      D1=D5/PO
      D2=12.+ETA2*(36.+4.5*ETA2)
      D3=ETA2*(15.+2.5*ETA2)
      D4=ETA*(5.+3.75*ETA2)
      B1=CK2*TTHMUN
      B2=-CK2*UNMTH2
      B3=A3COF*SINI
      C0=.5*B*RHO*QOMS2T*XNODP*AODP*TSI**4*PSIM2**3.5/SQRT(ALPHA2)
      C1=1.5*XNODP*ALPHA2**2*C0
      C4=D1*D3*B2
      C5=D5*D4*B3
      XNDT=C1*(
     1  (2.+ETA2*(3.+34.*EOSQ)+5.*EETA*(4.+ETA2)+8.5*EOSQ)+
     1   D1*D2*B1+   C4*COS2G+C5*SING)
      XNDTN=XNDT/XNODP

*      IF DRAG IS VERY SMALL, THE ISIMP FLAG IS SET AND THE
*      EQUATIONS ARE TRUNCATED TO LINEAR VARIATION IN MEAN
*      MOTION AND QUADRATIC VARIATION IN MEAN ANOMALY

      IF(ABS(XNDTN*XMNPDA) .LT. 2.16E-3) GO TO 50
      D6=ETA*(30.+22.5*ETA2)
      D7=ETA*(5.+12.5*ETA2)
      D8=1.+ETA2*(6.75+ETA2)
      C8=D1*D7*B2
      C9=D5*D8*B3
      EDOT=-C0*(
     1   ETA*(4.+ETA2+EOSQ*(15.5+7.*ETA2))+EO*(5.+15.*ETA2)+
     1   D1*D6*B1 +
     1   C8*COS2G+C9*SING)
      D20=.5*TOTHRD*XNDTN
```

```
      ALDTAL=EO*EDOT/ALPHA2
      TSDTTS=2.*AODP*TSI*(D20*BETAO2+EO*EDOT)
      ETDT=(EDOT+EO*TSDTTS)*TSI*S
      PSDTPS=-ETA*ETDT*PSIM2
      SIN2G=2.*SING*COSG
      C0DTC0=D20+4.*TSDTTS-ALDTAL-7.*PSDTPS
      C1DTC1=XNDTN+4.*ALDTAL+C0DTC0
      D9=ETA*(6.+68.*EOSQ)+EO*(20.+15.*ETA2)
      D10=5.*ETA*(4.+ETA2)+EO*(17.+68.*ETA2)
      D11=ETA*(72.+18.*ETA2)
      D12=ETA*(30.+10.*ETA2)
      D13=5.+11.25*ETA2
      D14=TSDTTS-2.*PSDTPS
      D15=2.*(D20+EO*EDOT/BETAO2)
      D1DT=D1*(D14+D15)
      D2DT=ETDT*D11
      D3DT=ETDT*D12
      D4DT=ETDT*D13
      D5DT=D5*D14
      C4DT=B2*(D1DT*D3+D1*D3DT)
      C5DT=B3*(D5DT*D4+D5*D4DT)
      D16=
     1     D9*ETDT+D10*EDOT +
     1     B1*(D1DT*D2+D1*D2DT) +
     1     C4DT*COS2G+C5DT*SING+XGDT1*(C5*COSG-2.*C4*SIN2G)
      XNDDT=C1DTC1*XNDT+C1*D16
      EDDOT=C0DTC0*EDOT-C0*(
     1     (4.+3.*ETA2+30.*EETA+EOSQ*(15.5+21.*ETA2))*ETDT+(5.+15.*ETA2
     '         +EETA*(31.+14.*ETA2))*EDOT +
     1     B1*(D1DT*D6+D1*ETDT*(30.+67.5*ETA2))  +
     1     B2*(D1DT*D7+D1*ETDT*(5.+37.5*ETA2))*COS2G+
     1     B3*(D5DT*D8+D5*ETDT*ETA*(13.5+4.*ETA2))*SING+XGDT1*(C9*
     '         COSG-2.*C8*SIN2G))
      D25=EDOT**2
      D17=XNDDT/XNODP-XNDTN**2
      TSDDTS=2.*TSDTTS*(TSDTTS-D20)+AODP*TSI*(TOTHRD*BETAO2*D17-4.*D20*
     '         EO*EDOT+2.*(D25+EO*EDDOT))
      ETDDT =(EDDOT+2.*EDOT*TSDTTS)*TSI*S+TSDDTS*ETA
      D18=TSDDTS-TSDTTS**2
      D19=-PSDTPS**2/ETA2-ETA*ETDDT*PSIM2-PSDTPS**2
      D23=ETDT*ETDT
      D1DDT=D1DT*(D14+D15)+D1*(D18-2.*D19+TOTHRD*D17+2.*(ALPHA2*D25
     '         /BETAO2+EO*EDDOT)/BETAO2)
      XNTRDT=XNDT*(2.*TOTHRD*D17+3.*
     1  (D25+EO*EDDOT)/ALPHA2-6.*ALDTAL**2 +
     1  4.*D18-7.*D19 )   +
     1  C1DTC1*XNDDT+C1*(C1DTC1*D16+
     1  D9*ETDDT+D10*EDDOT+D23*(6.+30.*EETA+68.*EOSQ)+
```

```
     1  ETDT*EDOT*(40.+30.*
     '  ETA2+272.*EETA)+D25*(17.+68.*ETA2) +
     1    B1*(D1DDT*D2+2.*D1DT*D2DT+D1*(ETDDT*D11+D23*(72.+54.*ETA2))) +
     1    B2*(D1DDT*D3+2.*D1DT*D3DT+D1*(ETDDT*D12+D23*(30.+30.*ETA2))) *
     1    COS2G+
     1      B3*((D5DT*D14+D5*(D18-2.*D19)) *
     1 D4+2.*D4DT*D5DT+D5*(ETDDT*D13+22.5*ETA*D23)) *SING+XGDT1*
     1         ((7.*D20+4.*EO*EDOT/BETAO2)*
     '         (C5*COSG-2.*C4*SIN2G)
     '         +((2.*C5DT*COSG-4.*C4DT*SIN2G)-XGDT1*(C5*SING+4.*
     '         C4*COS2G))))
      TMNDDT=XNDDT*1.E9
      TEMP=TMNDDT**2-XNDT*1.E18*XNTRDT
      PP=(TEMP+TMNDDT**2)/TEMP
      GAMMA=-XNTRDT/(XNDDT*(PP-2.))
      XND=XNDT/(PP*GAMMA)
      QQ=1.-EDDOT/(EDOT*GAMMA)
      ED=EDOT/(QQ*GAMMA)
      OVGPP=1./(GAMMA*(PP+1.))
      GO TO 70
   50 ISIMP=1
      EDOT=-TOTHRD*XNDTN*(1.-EO)
   70 IFLAG=0

*      UPDATE FOR SECULAR GRAVITY AND ATMOSPHERIC DRAG

  100 XMAM=FMOD2P(XMO+XLLDOT*TSINCE)
      OMGASM=OMEGAO+OMGDT*TSINCE
      XNODES=XNODEO+XNODOT*TSINCE
      IF(ISIMP .EQ. 1) GO TO 105
      TEMP=1.-GAMMA*TSINCE
      TEMP1=TEMP**PP
      XN=XNODP+XND*(1.-TEMP1)
      EM=EO+ED*(1.-TEMP**QQ)
      Z1=XND*(TSINCE+OVGPP*(TEMP*TEMP1-1.))
      GO TO 108
  105 XN=XNODP+XNDT*TSINCE
      EM=EO+EDOT*TSINCE
      Z1=.5*XNDT*TSINCE*TSINCE
  108 Z7=3.5*TOTHRD*Z1/XNODP
      XMAM=FMOD2P(XMAM+Z1+Z7*XMDT1)
      OMGASM=OMGASM+Z7*XGDT1
      XNODES=XNODES+Z7*XHDT1

*      SOLVE KEPLERS EQUATION

      ZC2=XMAM+EM*SIN(XMAM)*(1.+EM*COS(XMAM))
      DO 130 I=1,10
```

```
      SINE=SIN(ZC2)
      COSE=COS(ZC2)
      ZC5=1./(1.-EM*COSE)
      CAPE=(XMAM+EM*SINE-ZC2)*
     1   ZC5+ZC2
      IF(ABS(CAPE-ZC2) .LE. E6A) GO TO 140
  130 ZC2=CAPE

*       SHORT PERIOD PRELIMINARY QUANTITIES

  140 AM=(XKE/XN)**TOTHRD
      BETA2M=1.-EM*EM
      SINOS=SIN(OMGASM)
      COSOS=COS(OMGASM)
      AXNM=EM*COSOS
      AYNM=EM*SINOS
      PM=AM*BETA2M
      G1=1./PM
      G2=.5*CK2*G1
      G3=G2*G1
      BETA=SQRT(BETA2M)
      G4=.25*A3COF*SINI
      G5=.25*A3COF*G1
      SNF=BETA*SINE*ZC5
      CSF=(COSE-EM)*ZC5
      FM=ACTAN(SNF,CSF)
      SNFG=SNF*COSOS+CSF*SINOS
      CSFG=CSF*COSOS-SNF*SINOS
      SN2F2G=2.*SNFG*CSFG
      CS2F2G=2.*CSFG**2-1.
      ECOSF=EM*CSF
      G10=FM-XMAM+EM*SNF
      RM=PM/(1.+ECOSF)
      AOVR=AM/RM
      G13=XN*AOVR
      G14=-G13*AOVR
      DR=G2*(UNMTH2*CS2F2G-3.*TTHMUN)-G4*SNFG
      DIWC=3.*G3*SINI*CS2F2G-G5*AYNM
      DI=DIWC*COSI

*       UPDATE FOR SHORT PERIOD PERIODICS

      SNI2DU=SINIO2*(
     1   G3*(.5*(1.-7.*THETA2)*SN2F2G-3.*UNM5TH*G10)-G5*SINI*CSFG*(2.+
     2         ECOSF))-.5*G5*THETA2*AXNM/COSIO2
      XLAMB=FM+OMGASM+XNODES+G3*(.5*(1.+6.*COSI-7.*THETA2)*SN2F2G-3.*
     1      (UNM5TH+2.*COSI)*G10)+G5*SINI*(COSI*AXNM/(1.+COSI)-(2.
     2      +ECOSF)*CSFG)
```

```
      Y4=SINIO2*SNFG+CSFG*SNI2DU+.5*SNFG*COSIO2*DI
      Y5=SINIO2*CSFG-SNFG*SNI2DU+.5*CSFG*COSIO2*DI
      R=RM+DR
      RDOT=XN*AM*EM*SNF/BETA+G14*(2.*G2*UNMTH2*SN2F2G+G4*CSFG)
      RVDOT=XN*AM**2*BETA/RM+
     1      G14*DR+AM*G13*SINI*DIWC

*       ORIENTATION VECTORS

      SNLAMB=SIN(XLAMB)
      CSLAMB=COS(XLAMB)
      TEMP=2.*(Y5*SNLAMB-Y4*CSLAMB)
      UX=Y4*TEMP+CSLAMB
      VX=Y5*TEMP-SNLAMB
      TEMP=2.*(Y5*CSLAMB+Y4*SNLAMB)
      UY=-Y4*TEMP+SNLAMB
      VY=-Y5*TEMP+CSLAMB
      TEMP=2.*SQRT(1.-Y4*Y4-Y5*Y5)
      UZ=Y4*TEMP
      VZ=Y5*TEMP

*       POSITION AND VELOCITY

      X=R*UX
      Y=R*UY
      Z=R*UZ
      XDOT=RDOT*UX+RVDOT*VX
      YDOT=RDOT*UY+RVDOT*VY
      ZDOT=RDOT*UZ+RVDOT*VZ

      RETURN
      END
```

9 THE SDP8 MODEL

The NORAD mean element sets can be used for prediction with SDP8. All symbols not defined below are defined in the list of symbols in Section Twelve. The original mean motion (n''_o) and semimajor axis (a''_o) are first recovered from the input elements by the equations

$$a_1 = \left(\frac{k_e}{n_o}\right)^{\frac{2}{3}}$$

$$\delta_1 = \frac{3}{2}\frac{k_2}{{a_1}^2}\frac{(3\cos^2 i_o - 1)}{(1 - {e_o}^2)^{\frac{3}{2}}}$$

$$a_o = a_1\left(1 - \frac{1}{3}\delta_1 - {\delta_1}^2 - \frac{134}{81}{\delta_1}^3\right)$$

$$\delta_o = \frac{3}{2}\frac{k_2}{{a_o}^2}\frac{(3\cos^2 i_o - 1)}{(1 - {e_o}^2)^{\frac{3}{2}}}$$

$$n''_o = \frac{n_o}{1 + \delta_o}$$

$$a''_o = \frac{a_o}{1 - \delta_o}.$$

The ballistic coefficient (B term) is then calculated from the B^* drag term by

$$B = 2B^*/\rho_o$$

where

$$\rho_o = (2.461 \times 10^{-5})\ \text{XKMPER kg/m}^2\text{/Earth radii}$$

is a reference value of atmospheric density.

Then calculate the constants

$$\beta^2 = 1 - e^2$$

$$\theta = \cos i$$

$$\dot{M}_1 = -\frac{3}{2}\frac{n''k_2}{a''^2\beta^3}(1 - 3\theta^2)$$

$$\dot{\omega}_1 = -\frac{3}{2}\frac{n''k_2}{a''^2\beta^4}(1 - 5\theta^2)$$

$$\dot{\Omega}_1 = -3\frac{n''k_2}{a''^2\beta^4}\theta$$

$$\dot{M}_2 = \frac{3}{16}\frac{n''{k_2}^2}{a''^4\beta^7}(13 - 78\theta^2 + 137\theta^4)$$

$$\dot{\omega}_2 = \frac{3}{16}\frac{n''k_2 2}{a''^4\beta^8}(7 - 114\theta^2 + 395\theta^4) + \frac{5}{4}\frac{n''k_4}{a''^4\beta^8}(3 - 36\theta^2 + 49\theta^4)$$

$$\dot{\Omega}_2 = \frac{3}{2}\frac{n''{k_2}^2}{a''^4\beta^8}\theta(4 - 19\theta^2) + \frac{5}{2}\frac{n''k_4}{a''^4\beta^8}\theta(3 - 7\theta^2)$$

$$\dot{\ell} = n''_o + \dot{M}_1 + \dot{M}_2$$

$$\dot{\omega} = \dot{\omega}_1 + \dot{\omega}_2$$

$$\dot{\Omega} = \dot{\Omega}_1 + \dot{\Omega}_2$$

$$\xi = \frac{1}{a''\beta^2 - s}$$

$$\eta = es\xi$$

$$\psi = \sqrt{1 - \eta^2}$$

$$\alpha^2 = 1 + e^2$$

$$C_o = \frac{1}{2}B\rho_o(q_o - s)^4 n''a''\xi^4\alpha^{-1}\psi^{-7}$$

$$C_1 = \frac{3}{2}n''\alpha^4 C_o$$

$$D_1 = \xi\psi^{-2}/a''\beta^2$$

$$D_2 = 12 + 36\eta^2 + \frac{9}{2}\eta^4$$

$$D_3 = 15\eta^2 + \frac{5}{2}\eta^4$$

$$D_4 = 5\eta + \frac{15}{4}\eta^3$$

$$D_5 = \xi\psi^{-2}$$

$$B_1 = -k_2(1 - 3\theta^2)$$

$$B_2 = -k_2(1 - \theta^2)$$

$$B_3 = \frac{A_{3,0}}{k_2}\sin i$$

$$C_2 = D_1 D_3 B_2$$

$$C_3 = D_4 D_5 B_3$$

$$\dot{n}_o = C_1\left(2 + 3\eta^2 + 20e\eta + 5e\eta^3 + \frac{17}{2}e^2 + 34e^2\eta^2 + D_1 D_2 B_1 + C_2\cos 2\omega + C_3\sin\omega\right)$$

$$\dot{e}_o = -\frac{2}{3}\frac{\dot{n}}{n''}(1 - e)$$

where all quantities are epoch values.

At this point SDP8 calls the initialization section of DEEP which calculates all initialized quantities needed for the deep-space perturbations (see Section Ten).

The secular effect of gravity is included in mean anomaly by

$$M_{DF} = M_o + \dot{\ell}(t - t_o)$$

and the secular effects of gravity and atmospheric drag are included in argument of perigee and longitude of ascending node by

$$\omega = \omega_o + \dot{\omega}(t - t_o) + \dot{\omega}_1 Z_7$$

$$\Omega = \Omega_o + \dot{\Omega}(t - t_o) + \dot{\Omega}_1 Z_7$$

where

$$Z_7 = \frac{7}{3} Z_1 / n''_o$$

with

$$Z_1 = \frac{1}{2} \dot{n}_o (t - t_o)^2.$$

Next, SDP8 calls the secular section of DEEP which adds the deep-space secular effects and long-period resonance effects to the six classical orbital elements (see Section Ten).

The secular effects of drag are included in the remaining elements by

$$n = n_{DS} + \dot{n}_o (t - t_o)$$

$$e = e_{DS} + \dot{e}_o (t - t_o)$$

$$M = M_{DS} + Z_1 + \dot{M}_1 Z_7$$

where n_{DS}, e_{DS}, M_{DS} are the values of n_o, e_o, M_{DF} after deep-space secular and resonance perturbations have been applied.

Here, SDP8 calls the periodics section of DEEP which adds the deep-space lunar and solar periodics to the orbital elements (see Section Ten). From this point on, it will be assumed that n, e, I, ω, Ω, and M are the mean motion, eccentricity, inclination, argument of perigee, longitude of ascending node, and mean anomaly after lunar-solar periodics have been added.

Solve Kepler's equation for E by using the iteration equation

$$E_{i+1} = E_i + \Delta E_i$$

with

$$\Delta E_i = \frac{M + e \sin E_i - E_i}{1 - e \cos E_i}$$

and

$$E_1 = M + e \sin M + \frac{1}{2} e^2 \sin 2M.$$

The following equations are used to calculate preliminary quantities needed for the short-period periodics.

$$a = \left(\frac{k_e}{n}\right)^{\frac{2}{3}}$$

$$\beta = (1 - e^2)^{\frac{1}{2}}$$

$$\sin f = \frac{\beta \sin E}{1 - e\cos E}$$

$$\cos f = \frac{\cos E - e}{1 - e\cos E}$$

$$u = f + \omega$$

$$r'' = \frac{a\beta^2}{1 + e\cos f}$$

$$\dot{r}'' = \frac{nae}{\beta}\sin f$$

$$(r\dot{f})'' = \frac{na^2\beta}{r}$$

$$\delta r = \frac{1}{2}\frac{k_2}{a\beta^2}[(1-\theta^2)\cos 2u + 3(1-3\theta^2)] - \frac{1}{4}\frac{A_{3,0}}{k_2}\sin i_o \sin u$$

$$\delta\dot{r} = -n\left(\frac{a}{r}\right)^2\left[\frac{k_2}{a\beta^2}(1-\theta^2)\sin 2u + \frac{1}{4}\frac{A_{3,0}}{k_2}\sin i_o\cos u\right]$$

$$\delta I = \theta\left[\frac{3}{2}\frac{k_2}{a^2\beta^4}\sin i_o\cos 2u - \frac{1}{4}\frac{A_{3,0}}{k_2 a\beta^2}e\sin\omega\right]$$

$$\delta(r\dot{f}) = -n\left(\frac{a}{r}\right)^2\delta r + na\left(\frac{a}{r}\right)\frac{\sin i_o}{\theta}\delta I$$

$$\begin{aligned}\delta u = \; & \frac{1}{2}\frac{k_2}{a^2\beta^4}\left[\frac{1}{2}(1-7\theta^2)\sin 2u - 3(1-5\theta^2)(f - M + e\sin f)\right] \\ & -\frac{1}{4}\frac{A_{3,0}}{k_2 a\beta^2}\left[\sin i_o\cos u(2 + e\cos f) + \frac{1}{2}\frac{\theta^2}{sini_o/2\ \cos i_o/2}e\cos\omega\right]\end{aligned}$$

$$\delta\lambda = \frac{1}{2}\frac{k_2}{a^2\beta^4}\left[\frac{1}{2}(1+6\theta-7\theta^2)\sin 2u - 3(1+2\theta-5\theta^2)(f-M+e\sin f)\right]$$
$$+\frac{1}{4}\frac{A_{3,0}}{k_2 a\beta^2}\sin i_o\left[\frac{e\theta}{1+\theta}\cos\omega - (2+e\cos f)\cos u\right]$$

The short-period periodics are added to give the osculating quantities

$$r = r'' + \delta r$$

$$\dot{r} = \dot{r}'' + \delta\dot{r}$$

$$r\dot{f} = (r\dot{f})'' + \delta(r\dot{f})$$

$$y_4 = \sin\frac{I}{2}\sin u + \cos u\sin\frac{i_o}{2}\delta u + \frac{1}{2}\sin u\cos\frac{i_o}{2}\delta I$$

$$y_5 = \sin\frac{I}{2}\cos u - \sin u\sin\frac{i_o}{2}\delta u + \frac{1}{2}\cos u\cos\frac{i_o}{2}\delta I$$

$$\lambda = u + \Omega + \delta\lambda.$$

Unit orientation vectors are calculated by

$$U_x = 2y_4(y_5\sin\lambda - y_4\cos\lambda) + cos\lambda$$

$$U_y = -2y_4(y_5\cos\lambda + y_4\sin\lambda) + sin\lambda$$

$$U_z = 2y_4\cos\frac{I}{2}$$

$$V_x = 2y_5(y_5\sin\lambda - y_4\cos\lambda) - sin\lambda$$

$$V_y = -2y_5(y_5\cos\lambda + y_4\sin\lambda) + cos\lambda$$

$$V_z = 2y_5\cos\frac{I}{2}$$

where

$$\cos\frac{I}{2} = \sqrt{1 - {y_4}^2 - {y_5}^2}.$$

Position and velocity are given by

$$\mathbf{r} = r\mathbf{U}$$

$$\dot{\mathbf{r}} = \dot{r}\mathbf{U} + r\dot{f}\mathbf{V}.$$

A FORTRAN IV computer code listing of the subroutine SDP8 is given below.

```
*      SDP8                                              14 NOV 80
      SUBROUTINE SDP8(IFLAG,TSINCE)
      COMMON/E1/XMO,XNODEO,OMEGAO,EO,XINCL,XNO,XNDT2O,
     1           XNDD6O,BSTAR,X,Y,Z,XDOT,YDOT,ZDOT,EPOCH,DS50
      COMMON/C1/CK2,CK4,E6A,QOMS2T,S,TOTHRD,
     1           XJ3,XKE,XKMPER,XMNPDA,AE
      DOUBLE PRECISION EPOCH, DS50
      DATA RHO/.15696615/

      IF (IFLAG .EQ. 0) GO TO 100

*      RECOVER ORIGINAL MEAN MOTION (XNODP) AND SEMIMAJOR AXIS (AODP)
*      FROM INPUT ELEMENTS --------- CALCULATE BALLISTIC COEFFICIENT
*      (B TERM) FROM INPUT B* DRAG TERM

      A1=(XKE/XNO)**TOTHRD
      COSI=COS(XINCL)
      THETA2=COSI*COSI
      TTHMUN=3.*THETA2-1.
      EOSQ=EO*EO
      BETAO2=1.-EOSQ
      BETAO=SQRT(BETAO2)
      DEL1=1.5*CK2*TTHMUN/(A1*A1*BETAO*BETAO2)
      AO=A1*(1.-DEL1*(.5*TOTHRD+DEL1*(1.+134./81.*DEL1)))
      DELO=1.5*CK2*TTHMUN/(AO*AO*BETAO*BETAO2)
      AODP=AO/(1.-DELO)
      XNODP=XNO/(1.+DELO)
      B=2.*BSTAR/RHO

*      INITIALIZATION

      PO=AODP*BETAO2
      POM2=1./(PO*PO)
      SINI=SIN(XINCL)
      SING=SIN(OMEGAO)
      COSG=COS(OMEGAO)
      TEMP=.5*XINCL
      SINIO2=SIN(TEMP)
      COSIO2=COS(TEMP)
      THETA4=THETA2**2
      UNM5TH=1.-5.*THETA2
      UNMTH2=1.-THETA2
      A3COF=-XJ3/CK2*AE**3
      PARDT1=3.*CK2*POM2*XNODP
      PARDT2=PARDT1*CK2*POM2
      PARDT4=1.25*CK4*POM2*POM2*XNODP
      XMDT1=.5*PARDT1*BETAO*TTHMUN
      XGDT1=-.5*PARDT1*UNM5TH
```

```
      XHDT1=-PARDT1*COSI
      XLLDOT=XNODP+XMDT1+
     2           .0625*PARDT2*BETAO*(13.-78.*THETA2+137.*THETA4)
      OMGDT=XGDT1+
     1      .0625*PARDT2*(7.-114.*THETA2+395.*THETA4)+PARDT4*(3.-36.*
     2         THETA2+49.*THETA4)
      XNODOT=XHDT1+
     1       (.5*PARDT2*(4.-19.*THETA2)+2.*PARDT4*(3.-7.*THETA2))*COSI
      TSI=1./(PO-S)
      ETA=EO*S*TSI
      ETA2=ETA**2
      PSIM2=ABS(1./(1.-ETA2))
      ALPHA2=1.+EOSQ
      EETA=EO*ETA
      COS2G=2.*COSG**2-1.
      D5=TSI*PSIM2
      D1=D5/PO
      D2=12.+ETA2*(36.+4.5*ETA2)
      D3=ETA2*(15.+2.5*ETA2)
      D4=ETA*(5.+3.75*ETA2)
      B1=CK2*TTHMUN
      B2=-CK2*UNMTH2
      B3=A3COF*SINI
      C0=.5*B*RHO*QOMS2T*XNODP*AODP*TSI**4*PSIM2**3.5/SQRT(ALPHA2)
      C1=1.5*XNODP*ALPHA2**2*C0
      C4=D1*D3*B2
      C5=D5*D4*B3
      XNDT=C1*(
     1  (2.+ETA2*(3.+34.*EOSQ)+5.*EETA*(4.+ETA2)+8.5*EOSQ)+
     1   D1*D2*B1+   C4*COS2G+C5*SING)
      XNDTN=XNDT/XNODP
      EDOT=-TOTHRD*XNDTN*(1.-EO)
      IFLAG=0
      CALL DPINIT(EOSQ,SINI,COSI,BETAO,AODP,THETA2,SING,COSG,
     1          BETAO2,XLLDOT,OMGDT,XNODOT,XNODP)

*      UPDATE FOR SECULAR GRAVITY AND ATMOSPHERIC DRAG

  100 Z1=.5*XNDT*TSINCE*TSINCE
      Z7=3.5*TOTHRD*Z1/XNODP
      XMAMDF=XMO+XLLDOT*TSINCE
      OMGASM=OMEGAO+OMGDT*TSINCE+Z7*XGDT1
      XNODES=XNODEO+XNODOT*TSINCE+Z7*XHDT1
      XN=XNODP
      CALL DPSEC(XMAMDF,OMGASM,XNODES,EM,XINC,XN,TSINCE)
      XN=XN+XNDT*TSINCE
      EM=EM+EDOT*TSINCE
      XMAM=XMAMDF+Z1+Z7*XMDT1
```

```
      CALL DPPER(EM,XINC,OMGASM,XNODES,XMAM)
      XMAM=FMOD2P(XMAM)

*      SOLVE KEPLERS EQUATION

      ZC2=XMAM+EM*SIN(XMAM)*(1.+EM*COS(XMAM))
      DO 130 I=1,10
      SINE=SIN(ZC2)
      COSE=COS(ZC2)
      ZC5=1./(1.-EM*COSE)
      CAPE=(XMAM+EM*SINE-ZC2)*
     1   ZC5+ZC2
      IF(ABS(CAPE-ZC2) .LE. E6A) GO TO 140
  130 ZC2=CAPE

*      SHORT PERIOD PRELIMINARY QUANTITIES

  140 AM=(XKE/XN)**TOTHRD
      BETA2M=1.-EM*EM
      SINOS=SIN(OMGASM)
      COSOS=COS(OMGASM)
      AXNM=EM*COSOS
      AYNM=EM*SINOS
      PM=AM*BETA2M
      G1=1./PM
      G2=.5*CK2*G1
      G3=G2*G1
      BETA=SQRT(BETA2M)
      G4=.25*A3COF*SINI
      G5=.25*A3COF*G1
      SNF=BETA*SINE*ZC5
      CSF=(COSE-EM)*ZC5
      FM=ACTAN(SNF,CSF)
      SNFG=SNF*COSOS+CSF*SINOS
      CSFG=CSF*COSOS-SNF*SINOS
      SN2F2G=2.*SNFG*CSFG
      CS2F2G=2.*CSFG**2-1.
      ECOSF=EM*CSF
      G10=FM-XMAM+EM*SNF
      RM=PM/(1.+ECOSF)
      AOVR=AM/RM
      G13=XN*AOVR
      G14=-G13*AOVR
      DR=G2*(UNMTH2*CS2F2G-3.*TTHMUN)-G4*SNFG
      DIWC=3.*G3*SINI*CS2F2G-G5*AYNM
      DI=DIWC*COSI
      SINI2=SIN(.5*XINC)
```

```
*      UPDATE FOR SHORT PERIOD PERIODICS

      SNI2DU=SINIO2*(
     1   G3*(.5*(1.-7.*THETA2)*SN2F2G-3.*UNM5TH*G10)-G5*SINI*CSFG*(2.+
     2         ECOSF))-.5*G5*THETA2*AXNM/COSIO2
      XLAMB=FM+OMGASM+XNODES+G3*(.5*(1.+6.*COSI-7.*THETA2)*SN2F2G-3.*
     1      (UNM5TH+2.*COSI)*G10)+G5*SINI*(COSI*AXNM/(1.+COSI)-(2.
     2      +ECOSF)*CSFG)
      Y4=SINI2*SNFG+CSFG*SNI2DU+.5*SNFG*COSIO2*DI
      Y5=SINI2*CSFG-SNFG*SNI2DU+.5*CSFG*COSIO2*DI
      R=RM+DR
      RDOT=XN*AM*EM*SNF/BETA+G14*(2.*G2*UNMTH2*SN2F2G+G4*CSFG)
      RVDOT=XN*AM**2*BETA/RM+
     1      G14*DR+AM*G13*SINI*DIWC

*      ORIENTATION VECTORS

      SNLAMB=SIN(XLAMB)
      CSLAMB=COS(XLAMB)
      TEMP=2.*(Y5*SNLAMB-Y4*CSLAMB)
      UX=Y4*TEMP+CSLAMB
      VX=Y5*TEMP-SNLAMB
      TEMP=2.*(Y5*CSLAMB+Y4*SNLAMB)
      UY=-Y4*TEMP+SNLAMB
      VY=-Y5*TEMP+CSLAMB
      TEMP=2.*SQRT(1.-Y4*Y4-Y5*Y5)
      UZ=Y4*TEMP
      VZ=Y5*TEMP

*      POSITION AND VELOCITY

      X=R*UX
      Y=R*UY
      Z=R*UZ
      XDOT=RDOT*UX+RVDOT*VX
      YDOT=RDOT*UY+RVDOT*VY
      ZDOT=RDOT*UZ+RVDOT*VZ

      RETURN
      END
```

10 THE DEEP-SPACE SUBROUTINE

The two deep-space models, SDP4 and SDP8, both access the subroutine DEEP to obtain the deep-space perturbations of the six classical orbital elements. The perturbation equations are quite extensive and will not be repeated here. Rather, this section will concentrate on a general description of the flow between the main program and the deep-space subroutines. A specific listing of the equations is available in Hujsak (1979) or Hujsak and Hoots (1977).

The first time the deep-space subroutine is accessed is during the initialization portion of SDP4/ SDP8 and is via the entry DPINIT. Through this entry, certain constants already calculated in SDP4/ SDP8 are passed to the deep-space subroutine which in turn calculates all initialized (time independent) quantities needed for prediction in deep space. Additionally, a determination is made and flags are set concerning whether the orbit is synchronous and whether the orbit experiences resonance effects.

The next access to the deep-space subroutine occurs during the secular update portion of SDP4/ SDP8 and is via the entry DPSEC. Through this entry, the current secular values of the "mean" orbital elements are passed to the deep-space subroutine which in turn adds the appropriate deep-space secular and long-period resonance effects to these mean elements.

The last access to the deep-space subroutine occurs at the beginning of the osculation portion (periodics application) of SDP4/SDP8 and is via the entry DPPER. Through this entry, the current values of the orbital elements are passed to the deep-space subroutine which in turn adds the appropriate deep- space lunar and solar periodics to the orbital elements.

During initialization the deep-space subroutine calls the function subroutine THETAG to obtain the location of Greenwich at epoch and to convert epoch to minutes since 1950. All physical constants which are unique to the deep-space subroutine are set via data statements in DEEP rather than being passed through a COMMON.

A FORTRAN IV computer code listing of the subroutine DEEP is given below. These equations contain all currently anticipated changes to the SCC operational program. These changes are scheduled for implementation in March, 1981.

```
*      DEEP SPACE                                            31 OCT 80
      SUBROUTINE DEEP
      COMMON/E1/XMO,XNODEO,OMEGAO,EO,XINCL,XNO,XNDT2O,
     1           XNDD6O,BSTAR,X,Y,Z,XDOT,YDOT,ZDOT,EPOCH,DS50
      COMMON/C1/CK2,CK4,E6A,QOMS2T,S,TOTHRD,
     1           XJ3,XKE,XKMPER,XMNPDA,AE
      COMMON/C2/DE2RA,PI,PIO2,TWOPI,X3PIO2
      DOUBLE PRECISION EPOCH, DS50
      DOUBLE PRECISION
     *     DAY,PREEP,XNODCE,ATIME,DELT,SAVTSN,STEP2,STEPN,STEPP
      DATA              ZNS,           C1SS,          ZES/
     A                  1.19459E-5,    2.9864797E-6, .01675/
      DATA              ZNL,           C1L,           ZEL/
     A                  1.5835218E-4,  4.7968065E-7,  .05490/
      DATA              ZCOSIS,        ZSINIS,        ZSINGS/
     A                  .91744867,     .39785416,     -.98088458/
      DATA              ZCOSGS,        ZCOSHS,        ZSINHS/
     A                  .1945905,      1.0,           0.0/
      DATA Q22,Q31,Q33/1.7891679E-6,2.1460748E-6,2.2123015E-7/
      DATA G22,G32/5.7686396,0.95240898/
      DATA G44,G52/1.8014998,1.0508330/
      DATA G54/4.4108898/
      DATA ROOT22,ROOT32/1.7891679E-6,3.7393792E-7/
      DATA ROOT44,ROOT52/7.3636953E-9,1.1428639E-7/
      DATA ROOT54/2.1765803E-9/
      DATA THDT/4.3752691E-3/

*     ENTRANCE FOR DEEP SPACE INITIALIZATION

      ENTRY DPINIT(EQSQ,SINIQ,COSIQ,RTEQSQ,AO,COSQ2,SINOMO,COSOMO,
     1         BSQ,XLLDOT,OMGDT,XNODOT,XNODP)
      THGR=THETAG(EPOCH)
      EQ = EO
      XNQ = XNODP
      AQNV = 1./AO
      XQNCL = XINCL
      XMAO=XMO
      XPIDOT=OMGDT+XNODOT
      SINQ = SIN(XNODEO)
      COSQ = COS(XNODEO)
      OMEGAQ = OMEGAO

*     INITIALIZE LUNAR SOLAR TERMS

    5 DAY=DS50+18261.5D0
      IF (DAY.EQ.PREEP)    GO TO 10
      PREEP = DAY
      XNODCE=4.5236020-9.2422029E-4*DAY
```

```
      STEM=DSIN (XNODCE)
      CTEM=DCOS (XNODCE)
      ZCOSIL=.91375164-.03568096*CTEM
      ZSINIL=SQRT (1.-ZCOSIL*ZCOSIL)
      ZSINHL= .089683511*STEM/ZSINIL
      ZCOSHL=SQRT (1.-ZSINHL*ZSINHL)
      C=4.7199672+.22997150*DAY
      GAM=5.8351514+.0019443680*DAY
      ZMOL = FMOD2P(C-GAM)
      ZX= .39785416*STEM/ZSINIL
      ZY= ZCOSHL*CTEM+0.91744867*ZSINHL*STEM
      ZX=ACTAN(ZX,ZY)
      ZX=GAM+ZX-XNODCE
      ZCOSGL=COS (ZX)
      ZSINGL=SIN (ZX)
      ZMOS=6.2565837D0+.017201977D0*DAY
      ZMOS=FMOD2P(ZMOS)

*     DO SOLAR TERMS

   10 LS = 0
      SAVTSN=1.D20
      ZCOSG=ZCOSGS
      ZSING=ZSINGS
      ZCOSI=ZCOSIS
      ZSINI=ZSINIS
      ZCOSH=COSQ
      ZSINH=SINQ
      CC=C1SS
      ZN=ZNS
      ZE=ZES
      ZMO=ZMOS
      XNOI=1./XNQ
      ASSIGN 30 TO LS
   20 A1=ZCOSG*ZCOSH+ZSING*ZCOSI*ZSINH
      A3=-ZSING*ZCOSH+ZCOSG*ZCOSI*ZSINH
      A7=-ZCOSG*ZSINH+ZSING*ZCOSI*ZCOSH
      A8=ZSING*ZSINI
      A9=ZSING*ZSINH+ZCOSG*ZCOSI*ZCOSH
      A10=ZCOSG*ZSINI
      A2= COSIQ*A7+ SINIQ*A8
      A4= COSIQ*A9+ SINIQ*A10
      A5=- SINIQ*A7+ COSIQ*A8
      A6=- SINIQ*A9+ COSIQ*A10
C
      X1=A1*COSOMO+A2*SINOMO
      X2=A3*COSOMO+A4*SINOMO
      X3=-A1*SINOMO+A2*COSOMO
```

```
      X4=-A3*SINOMO+A4*COSOMO
      X5=A5*SINOMO
      X6=A6*SINOMO
      X7=A5*COSOMO
      X8=A6*COSOMO
C
      Z31=12.*X1*X1-3.*X3*X3
      Z32=24.*X1*X2-6.*X3*X4
      Z33=12.*X2*X2-3.*X4*X4
      Z1=3.*(A1*A1+A2*A2)+Z31*EQSQ
      Z2=6.*(A1*A3+A2*A4)+Z32*EQSQ
      Z3=3.*(A3*A3+A4*A4)+Z33*EQSQ
      Z11=-6.*A1*A5+EQSQ *(-24.*X1*X7-6.*X3*X5)
      Z12=-6.*(A1*A6+A3*A5)+EQSQ *(-24.*(X2*X7+X1*X8)-6.*(X3*X6+X4*X5))
      Z13=-6.*A3*A6+EQSQ *(-24.*X2*X8-6.*X4*X6)
      Z21=6.*A2*A5+EQSQ *(24.*X1*X5-6.*X3*X7)
      Z22=6.*(A4*A5+A2*A6)+EQSQ *(24.*(X2*X5+X1*X6)-6.*(X4*X7+X3*X8))
      Z23=6.*A4*A6+EQSQ *(24.*X2*X6-6.*X4*X8)
      Z1=Z1+Z1+BSQ*Z31
      Z2=Z2+Z2+BSQ*Z32
      Z3=Z3+Z3+BSQ*Z33
      S3=CC*XNOI
      S2=-.5*S3/RTEQSQ
      S4=S3*RTEQSQ
      S1=-15.*EQ*S4
      S5=X1*X3+X2*X4
      S6=X2*X3+X1*X4
      S7=X2*X4-X1*X3
      SE=S1*ZN*S5
      SI=S2*ZN*(Z11+Z13)
      SL=-ZN*S3*(Z1+Z3-14.-6.*EQSQ)
      SGH=S4*ZN*(Z31+Z33-6.)
      SH=-ZN*S2*(Z21+Z23)
      IF(XQNCL.LT.5.2359877E-2) SH=0.0
      EE2=2.*S1*S6
      E3=2.*S1*S7
      XI2=2.*S2*Z12
      XI3=2.*S2*(Z13-Z11)
      XL2=-2.*S3*Z2
      XL3=-2.*S3*(Z3-Z1)
      XL4=-2.*S3*(-21.-9.*EQSQ)*ZE
      XGH2=2.*S4*Z32
      XGH3=2.*S4*(Z33-Z31)
      XGH4=-18.*S4*ZE
      XH2=-2.*S2*Z22
      XH3=-2.*S2*(Z23-Z21)
      GO TO LS
```

```
*     DO LUNAR TERMS

   30 SSE = SE
      SSI=SI
      SSL=SL
      SSH=SH/SINIQ
      SSG=SGH-COSIQ*SSH
      SE2=EE2
      SI2=XI2
      SL2=XL2
      SGH2=XGH2
      SH2=XH2
      SE3=E3
      SI3=XI3
      SL3=XL3
      SGH3=XGH3
      SH3=XH3
      SL4=XL4
      SGH4=XGH4
      LS=1
      ZCOSG=ZCOSGL
      ZSING=ZSINGL
      ZCOSI=ZCOSIL
      ZSINI=ZSINIL
      ZCOSH=ZCOSHL*COSQ+ZSINHL*SINQ
      ZSINH=SINQ*ZCOSHL-COSQ*ZSINHL
      ZN=ZNL
      CC=C1L
      ZE=ZEL
      ZMO=ZMOL
      ASSIGN 40 TO LS
      GO TO 20
   40 SSE = SSE+SE
      SSI=SSI+SI
      SSL=SSL+SL
      SSG=SSG+SGH-COSIQ/SINIQ*SH
      SSH=SSH+SH/SINIQ

*     GEOPOTENTIAL RESONANCE INITIALIZATION FOR 12 HOUR ORBITS

      IRESFL=0
      ISYNFL=0
      IF(XNQ.LT.(.0052359877).AND.XNQ.GT.(.0034906585)) GO TO 70
      IF (XNQ.LT.(8.26E-3) .OR. XNQ.GT.(9.24E-3))    RETURN
      IF (EQ.LT.0.5)    RETURN
      IRESFL =1
      EOC=EQ*EQSQ
      G201=-.306-(EQ-.64)*.440
```

```
      IF(EQ.GT.(.65)) GO TO 45
      G211=3.616-13.247*EQ+16.290*EQSQ
      G310=-19.302+117.390*EQ-228.419*EQSQ+156.591*EOC
      G322=-18.9068+109.7927*EQ-214.6334*EQSQ+146.5816*EOC
      G410=-41.122+242.694*EQ-471.094*EQSQ+313.953*EOC
      G422=-146.407+841.880*EQ-1629.014*EQSQ+1083.435*EOC
      G520=-532.114+3017.977*EQ-5740*EQSQ+3708.276*EOC
      GO TO 55
   45 G211=-72.099+331.819*EQ-508.738*EQSQ+266.724*EOC
      G310=-346.844+1582.851*EQ-2415.925*EQSQ+1246.113*EOC
      G322=-342.585+1554.908*EQ-2366.899*EQSQ+1215.972*EOC
      G410=-1052.797+4758.686*EQ-7193.992*EQSQ+3651.957*EOC
      G422=-3581.69+16178.11*EQ-24462.77*EQSQ+12422.52*EOC
      IF(EQ.GT.(.715)) GO TO 50
      G520=1464.74-4664.75*EQ+3763.64*EQSQ
      GO TO 55
   50 G520=-5149.66+29936.92*EQ-54087.36*EQSQ+31324.56*EOC
   55 IF(EQ.GE.(.7)) GO TO 60
      G533=-919.2277+4988.61*EQ-9064.77*EQSQ+5542.21*EOC
      G521 = -822.71072+4568.6173*EQ-8491.4146*EQSQ+5337.524*EOC
      G532 = -853.666+4690.25*EQ-8624.77*EQSQ+5341.4*EOC
      GO TO 65
   60 G533=-37995.78+161616.52*EQ-229838.2*EQSQ+109377.94*EOC
      G521 = -51752.104+218913.95*EQ-309468.16*EQSQ+146349.42*EOC
      G532 = -40023.88+170470.89*EQ-242699.48*EQSQ+115605.82*EOC
   65 SINI2=SINIQ*SINIQ
      F220=.75*(1.+2.*COSIQ+COSQ2)
      F221=1.5*SINI2
      F321=1.875*SINIQ*(1.-2.*COSIQ-3.*COSQ2)
      F322=-1.875*SINIQ*(1.+2.*COSIQ-3.*COSQ2)
      F441=35.*SINI2*F220
      F442=39.3750*SINI2*SINI2
      F522=9.84375*SINIQ*(SINI2*(1.-2.*COSIQ-5.*COSQ2)
     1     +.33333333*(-2.+4.*COSIQ+6.*COSQ2))
      F523 = SINIQ*(4.92187512*SINI2*(-2.-4.*COSIQ+10.*COSQ2)
     *       +6.56250012*(1.+2.*COSIQ-3.*COSQ2))
      F542 = 29.53125*SINIQ*(2.-8.*COSIQ+COSQ2*(-12.+8.*COSIQ
     *       +10.*COSQ2))
      F543=29.53125*SINIQ*(-2.-8.*COSIQ+COSQ2*(12.+8.*COSIQ-10.*COSQ2))
      XNO2=XNQ*XNQ
      AINV2=AQNV*AQNV
      TEMP1 = 3.*XNO2*AINV2
      TEMP = TEMP1*ROOT22
      D2201 = TEMP*F220*G201
      D2211 = TEMP*F221*G211
      TEMP1 = TEMP1*AQNV
      TEMP = TEMP1*ROOT32
      D3210 = TEMP*F321*G310
```

```
      D3222 = TEMP*F322*G322
      TEMP1 = TEMP1*AQNV
      TEMP = 2.*TEMP1*ROOT44
      D4410 = TEMP*F441*G410
      D4422 = TEMP*F442*G422
      TEMP1 = TEMP1*AQNV
      TEMP = TEMP1*ROOT52
      D5220 = TEMP*F522*G520
      D5232 = TEMP*F523*G532
      TEMP = 2.*TEMP1*ROOT54
      D5421 = TEMP*F542*G521
      D5433 = TEMP*F543*G533
      XLAMO = XMAO+XNODEO+XNODEO-THGR-THGR
      BFACT = XLLDOT+XNODOT+XNODOT-THDT-THDT
      BFACT=BFACT+SSL+SSH+SSH
      GO TO 80

*       SYNCHRONOUS RESONANCE TERMS INITIALIZATION

   70 IRESFL=1
      ISYNFL=1
      G200=1.0+EQSQ*(-2.5+.8125*EQSQ)
      G310=1.0+2.0*EQSQ
      G300=1.0+EQSQ*(-6.0+6.60937*EQSQ)
      F220=.75*(1.+COSIQ)*(1.+COSIQ)
      F311=.9375*SINIQ*SINIQ*(1.+3.*COSIQ)-.75*(1.+COSIQ)
      F330=1.+COSIQ
      F330=1.875*F330*F330*F330
      DEL1=3.*XNQ*XNQ*AQNV*AQNV
      DEL2=2.*DEL1*F220*G200*Q22
      DEL3=3.*DEL1*F330*G300*Q33*AQNV
      DEL1=DEL1*F311*G310*Q31*AQNV
      FASX2=.13130908
      FASX4=2.8843198
      FASX6=.37448087
      XLAMO=XMAO+XNODEO+OMEGAO-THGR
      BFACT = XLLDOT+XPIDOT-THDT
      BFACT=BFACT+SSL+SSG+SSH
   80 XFACT=BFACT-XNQ
C
C     INITIALIZE INTEGRATOR
C
      XLI=XLAMO
      XNI=XNQ
      ATIME=0.D0
      STEPP=720.D0
      STEPN=-720.D0
      STEP2 = 259200.D0
```

```
      RETURN

*     ENTRANCE FOR DEEP SPACE SECULAR EFFECTS

      ENTRY DPSEC(XLL,OMGASM,XNODES,EM,XINC,XN,T)
      XLL=XLL+SSL*T
      OMGASM=OMGASM+SSG*T
      XNODES=XNODES+SSH*T
      EM=EO+SSE*T
      XINC=XINCL+SSI*T
      IF(XINC .GE. 0.) GO TO 90
      XINC = -XINC
      XNODES = XNODES + PI
      OMGASM = OMGASM - PI
   90 IF(IRESFL .EQ. 0) RETURN
  100 IF (ATIME.EQ.0.D0)    GO TO 170
      IF(T.GE.(0.D0).AND.ATIME.LT.(0.D0)) GO TO 170
      IF(T.LT.(0.D0).AND.ATIME.GE.(0.D0)) GO TO 170
  105 IF(DABS(T).GE.DABS(ATIME)) GO TO 120
      DELT=STEPP
      IF (T.GE.0.D0)    DELT = STEPN
  110 ASSIGN 100 TO IRET
      GO TO 160
  120 DELT=STEPN
      IF (T.GT.0.D0)    DELT = STEPP
  125 IF (DABS(T-ATIME).LT.STEPP)    GO TO 130
      ASSIGN 125 TO IRET
      GO TO 160
  130 FT = T-ATIME
      ASSIGN 140 TO IRETN
      GO TO 150
  140 XN = XNI+XNDOT*FT+XNDDT*FT*FT*0.5
      XL = XLI+XLDOT*FT+XNDOT*FT*FT*0.5
      TEMP = -XNODES+THGR+T*THDT
      XLL = XL-OMGASM+TEMP
      IF (ISYNFL.EQ.0)    XLL = XL+TEMP+TEMP
      RETURN
C
C     DOT TERMS CALCULATED
C
  150 IF (ISYNFL.EQ.0)    GO TO 152
      XNDOT=DEL1*SIN (XLI-FASX2)+DEL2*SIN (2.*(XLI-FASX4))
     1     +DEL3*SIN (3.*(XLI-FASX6))
      XNDDT = DEL1*COS(XLI-FASX2)
     *       +2.*DEL2*COS(2.*(XLI-FASX4))
     *       +3.*DEL3*COS(3.*(XLI-FASX6))
      GO TO 154
  152 XOMI = OMEGAQ+OMGDT*ATIME
```

```
      X2OMI = XOMI+XOMI
      X2LI = XLI+XLI
      XNDOT = D2201*SIN(X2OMI+XLI-G22)
     *       +D2211*SIN(XLI-G22)
     *       +D3210*SIN(XOMI+XLI-G32)
     *       +D3222*SIN(-XOMI+XLI-G32)
     *       +D4410*SIN(X2OMI+X2LI-G44)
     *       +D4422*SIN(X2LI-G44)
     *       +D5220*SIN(XOMI+XLI-G52)
     *       +D5232*SIN(-XOMI+XLI-G52)
     *       +D5421*SIN(XOMI+X2LI-G54)
     *       +D5433*SIN(-XOMI+X2LI-G54)
      XNDDT = D2201*COS(X2OMI+XLI-G22)
     *       +D2211*COS(XLI-G22)
     *       +D3210*COS(XOMI+XLI-G32)
     *       +D3222*COS(-XOMI+XLI-G32)
     *       +D5220*COS(XOMI+XLI-G52)
     *       +D5232*COS(-XOMI+XLI-G52)
     *       +2.*(D4410*COS(X2OMI+X2LI-G44)
     *       +D4422*COS(X2LI-G44)
     *       +D5421*COS(XOMI+X2LI-G54)
     *       +D5433*COS(-XOMI+X2LI-G54))
  154 XLDOT=XNI+XFACT
      XNDDT = XNDDT*XLDOT
      GO TO IRETN
C
C     INTEGRATOR
C
  160 ASSIGN 165 TO IRETN
      GO TO 150
  165 XLI = XLI+XLDOT*DELT+XNDOT*STEP2
      XNI = XNI+XNDOT*DELT+XNDDT*STEP2
      ATIME=ATIME+DELT
      GO TO IRET
C
C     EPOCH RESTART
C
  170 IF (T.GE.0.D0)    GO TO 175
      DELT=STEPN
      GO TO 180
  175 DELT = STEPP
  180 ATIME = 0.D0
      XNI=XNQ
      XLI=XLAMO
      GO TO 125
C
C     ENTRANCES FOR LUNAR-SOLAR PERIODICS
C
```

```
C
      ENTRY DPPER(EM,XINC,OMGASM,XNODES,XLL)
      SINIS = SIN(XINC)
      COSIS = COS(XINC)
      IF (DABS(SAVTSN-T).LT.(30.D0))    GO TO 210
      SAVTSN=T
      ZM=ZMOS+ZNS*T
  205 ZF=ZM+2.*ZES*SIN (ZM)
      SINZF=SIN (ZF)
      F2=.5*SINZF*SINZF-.25
      F3=-.5*SINZF*COS (ZF)
      SES=SE2*F2+SE3*F3
      SIS=SI2*F2+SI3*F3
      SLS=SL2*F2+SL3*F3+SL4*SINZF
      SGHS=SGH2*F2+SGH3*F3+SGH4*SINZF
      SHS=SH2*F2+SH3*F3
      ZM=ZMOL+ZNL*T
      ZF=ZM+2.*ZEL*SIN (ZM)
      SINZF=SIN (ZF)
      F2=.5*SINZF*SINZF-.25
      F3=-.5*SINZF*COS (ZF)
      SEL=EE2*F2+E3*F3
      SIL=XI2*F2+XI3*F3
      SLL=XL2*F2+XL3*F3+XL4*SINZF
      SGHL=XGH2*F2+XGH3*F3+XGH4*SINZF
      SHL=XH2*F2+XH3*F3
      PE=SES+SEL
      PINC=SIS+SIL
      PL=SLS+SLL
  210 PGH=SGHS+SGHL
      PH=SHS+SHL
      XINC = XINC+PINC
      EM = EM+PE
      IF(XQNCL.LT.(.2)) GO TO 220
      GO TO 218
C
C     APPLY PERIODICS DIRECTLY
C
  218 PH=PH/SINIQ
      PGH=PGH-COSIQ*PH
      OMGASM=OMGASM+PGH
      XNODES=XNODES+PH
      XLL = XLL+PL
      GO TO 230
C
C     APPLY PERIODICS WITH LYDDANE MODIFICATION
C
  220 SINOK=SIN(XNODES)
```

```
      COSOK=COS(XNODES)
      ALFDP=SINIS*SINOK
      BETDP=SINIS*COSOK
      DALF=PH*COSOK+PINC*COSIS*SINOK
      DBET=-PH*SINOK+PINC*COSIS*COSOK
      ALFDP=ALFDP+DALF
      BETDP=BETDP+DBET
      XLS = XLL+OMGASM+COSIS*XNODES
      DLS=PL+PGH-PINC*XNODES*SINIS
      XLS=XLS+DLS
      XNODES=ACTAN(ALFDP,BETDP)
      XLL = XLL+PL
      OMGASM = XLS-XLL-COS(XINC)*XNODES
  230 CONTINUE
      RETURN
      END
```

11 DRIVER AND FUNCTION SUBROUTINES

The DRIVER controls the input and output function and the selection of the model. The input consists of a program card which specifies the model to be used and the output times and either a G-card or T-card element set.

The DRIVER reads and converts the input elements to units of radians and minutes. These are communicated to the prediction model through the COMMON E1. Values of the physical and mathematical constants are set and communicated through the COMMONs C1 and C2, respectively.

The program card indicates the mathematical model to be used and the start and stop time of prediction as well as the increment of time for output. These times are in minutes since epoch.

In the interest of efficiency the DRIVER sets a flag (IFLAG) the first time the model is called. This flag tells the model to calculate all initialized (time independent) quantities. After initialization, the model subroutine turns off the flag so that all subsequent calls only access the time dependent part of the model. This mode continues until another input case is encountered.

The DRIVER takes the output from the mathematical model (communicated through the COMMON E1) and converts it to units of kilometers and seconds for printout.

The function subroutine ACTAN is passed the values of sine and cosine in that order and it returns the angle in radians within the range of 0 to 2π. The function subroutine FMOD2P is passed an angle in radians and returns the angle in radians within the range of 0 to 2π. The function subroutine THETAG is passed the epoch time exactly as it appears on the input element cards.[1] The routine converts this time to days since 1950 Jan 0.0 UTC, stores this in the COMMON E1, and returns the right ascension of Greenwich at epoch (in radians).

FORTRAN IV computer code listings of the routines DRIVER, ACTAN, FMOD2P, and THETAG are given below.

[1] If only one year digit is given (as on standard G-cards) the program assumes the 80 decade. This may be overridden by putting a 2 digit year in columns 30–31 of the first G-card.

```
*      DRIVER                                                     3 NOV 80

*   WGS-72 PHYSICAL AND GEOPOTENTIAL CONSTANTS
*          CK2= .5*J2*AE**2     CK4=-.375*J4*AE**4

      DOUBLE PRECISION EPOCH,DS50
      COMMON/E1/XMO,XNODEO,OMEGAO,EO,XINCL,XNO,XNDT2O,XNDD6O,BSTAR,
     1            X,Y,Z,XDOT,YDOT,ZDOT,EPOCH,DS50
      COMMON/C1/CK2,CK4,E6A,QOMS2T,S,TOTHRD,
     1           XJ3,XKE,XKMPER,XMNPDA,AE
      COMMON/C2/DE2RA,PI,PIO2,TWOPI,X3PIO2
      DATA IHG/1HG/
      DATA DE2RA,E6A,PI,PIO2,QO,SO,TOTHRD,TWOPI,X3PIO2,XJ2,XJ3,
     1            XJ4,XKE,XKMPER,XMNPDA,AE/.174532925E-1,1.E-6,
     2            3.14159265,1.57079633,120.0,78.0,.66666667,
     4            6.2831853,4.71238898,1.082616E-3,-.253881E-5,
     5            -1.65597E-6,.743669161E-1,6378.135,1440.,1./
      DIMENSION ISET(5)
      CHARACTER ABUF*80(2)
      DATA (ISET(I),I=1,5)/3HSGP,4HSGP4,4HSDP4,4HSGP8,4HSDP8/

*     SELECT EPHEMERIS TYPE AND OUTPUT TIMES

      CK2=.5*XJ2*AE**2
      CK4=-.375*XJ4*AE**4
      QOMS2T=((QO-SO)*AE/XKMPER)**4
      S=AE*(1.+SO/XKMPER)
    2 READ (5,700) IEPT, TS,TF,DELT
      IF(IEPT.LE.0) STOP
      IDEEP=0

*     READ IN MEAN ELEMENTS FROM 2 CARD T(TRANS) OR G(INTERN) FORMAT

      READ (5,706) ABUF
      DECODE(ABUF(1),707)  ITYPE
      IF(ITYPE.EQ.IHG) GO TO 5
      DECODE (ABUF,702) EPOCH,XNDT2O,XNDD6O,IEXP,BSTAR,IBEXP,XINCL,
     1        XNODEO,EO,OMEGAO,XMO,XNO
      GO TO 7
    5 DECODE(ABUF,701) EPOCH,XMO,XNODEO,OMEGAO,EO,XINCL,XNO,XNDT2O,
     1        XNDD6O,IEXP,BSTAR,IBEXP
   7  IF(XNO.LE.0.) STOP
      WRITE(6,704) ABUF,ISET(IEPT)
      IF(IEPT.GT.5) GO TO 900
      XNDD6O=XNDD6O*(10.**IEXP)
      XNODEO=XNODEO*DE2RA
      OMEGAO=OMEGAO*DE2RA
      XMO=XMO*DE2RA
```

```
      XINCL=XINCL*DE2RA
      TEMP=TWOPI/XMNPDA/XMNPDA
      XNO=XNO*TEMP*XMNPDA
      XNDT2O=XNDT2O*TEMP
      XNDD6O=XNDD6O*TEMP/XMNPDA

*     INPUT CHECK FOR PERIOD VS EPHEMERIS SELECTED
*     PERIOD GE 225 MINUTES  IS DEEP SPACE

      A1=(XKE/XNO)**TOTHRD
      TEMP=1.5*CK2*(3.*COS(XINCL)**2-1.)/(1.-EO*EO)**1.5
      DEL1=TEMP/(A1*A1)
      AO=A1*(1.-DEL1*(.5*TOTHRD+DEL1*(1.+134./81.*DEL1)))
      DELO=TEMP/(AO*AO)
      XNODP=XNO/(1.+DELO)
      IF((TWOPI/XNODP/XMNPDA) .GE. .15625) IDEEP=1

      BSTAR=BSTAR*(10.**IBEXP)/AE
      TSINCE=TS
      IFLAG=1
      IF(IDEEP .EQ. 1 .AND. (IEPT .EQ. 1 .OR. IEPT .EQ. 2
     1           .OR. IEPT .EQ. 4)) GO TO 800
    9 IF(IDEEP .EQ. 0 .AND. (IEPT .EQ. 3 .OR. IEPT .EQ. 5))
     1           GO TO 850
   10 GO TO (21,22,23,24,25), IEPT
   21 CALL SGP(IFLAG,TSINCE)
      GO TO 60
   22 CALL SGP4(IFLAG,TSINCE)
      GO TO 60
   23 CALL SDP4(IFLAG,TSINCE)
      GO TO 60
   24 CALL SGP8(IFLAG,TSINCE)
      GO TO 60
   25 CALL SDP8(IFLAG,TSINCE)
   60 X=X*XKMPER/AE
      Y=Y*XKMPER/AE
      Z=Z*XKMPER/AE
      XDOT=XDOT*XKMPER/AE*XMNPDA/86400.
      YDOT=YDOT*XKMPER/AE*XMNPDA/86400.
      ZDOT=ZDOT*XKMPER/AE*XMNPDA/86400.
      WRITE(6,705) TSINCE,X,Y,Z,XDOT,YDOT,ZDOT
      TSINCE=TSINCE+DELT
      IF(ABS(TSINCE) .GT. ABS(TF)) GO TO 2
      GO TO 10
  700 FORMAT(I1,3F10.0)
  701 FORMAT(29X,D14.8,1X,3F8.4,/,6X,F8.7,F8.4,1X,2F11.9,1X,F6.5,I2,
     1       4X,F8.7,I2)
  702 FORMAT(18X,D14.8,1X,F10.8,2(1X,F6.5,I2),/,7X,2(1X,F8.4),1X,
```

```
   1       F7.7,2(1X,F8.4),1X,F11.8)
703 FORMAT(79X,A1)
704 FORMAT(1H1,A80,/,1X,A80,//,1X,A4,7H TSINCE,
   1 14X,1HX,16X,1HY,16X,1HZ,14X,
   1 4HXDOT,13X,4HYDOT,13X,4HZDOT,//)
705 FORMAT(7F17.8)
706 FORMAT(A80)
707 FORMAT(79X,A1)
930 FORMAT("SHOULD USE DEEP SPACE EPHEMERIS")
940 FORMAT("SHOULD USE NEAR EARTH EPHEMERIS")
950 FORMAT("EPHEMERIS NUMBER",I2," NOT LEGAL, WILL SKIP THIS CASE")
800 WRITE(6,930)
    GO TO 9
850 WRITE(6,940)
    GO TO 10
900 WRITE(6,950) IEPT
    GO TO 2
    END
```

```
      FUNCTION  ACTAN(SINX,COSX)
      COMMON/C2/DE2RA,PI,PIO2,TWOPI,X3PIO2
      ACTAN=0.
      IF (COSX.EQ.0.  ) GO TO 5
      IF (COSX.GT.0.  ) GO TO 1
      ACTAN=PI
      GO TO 7
    1 IF (SINX.EQ.0.  ) GO TO 8
      IF (SINX.GT.0.  ) GO TO 7
      ACTAN=TWOPI
      GO TO 7
    5 IF (SINX.EQ.0.  ) GO TO 8
       IF (SINX.GT.0.  ) GO TO 6
      ACTAN=X3PIO2
      GO TO 8
    6 ACTAN=PIO2
      GO TO 8
    7 TEMP=SINX/COSX
      ACTAN=ACTAN+ATAN(TEMP)
    8 RETURN
      END
```

```
FUNCTION FMOD2P(X)
COMMON/C2/DE2RA,PI,PIO2,TWOPI,X3PIO2
FMOD2P=X
I=FMOD2P/TWOPI
 FMOD2P=FMOD2P-I*TWOPI
IF(FMOD2P.LT.0) FMOD2P=FMOD2P+TWOPI
RETURN
END
```

```
      FUNCTION THETAG(EP)
      COMMON /E1/XMO,XNODEO,OMEGAO,EO,XINCL,XNO,XNDT2O,XNDD6O,BSTAR,
     1 X,Y,Z,XDOT,YDOT,ZDOT,EPOCH,DS50
      DOUBLE PRECISION EPOCH,D,THETA,TWOPI,YR,TEMP,EP,DS50
      TWOPI=6.28318530717959D0
      YR=(EP+2.D-7)*1.D-3
      JY=YR
      YR=JY
      D=EP-YR*1.D3
      IF(JY.LT.10) JY=JY+80
      N=(JY-69)/4
      IF(JY.LT.70) N=(JY-72)/4
      DS50=7305.D0 + 365.D0*(JY-70) +N + D
      THETA=1.72944494D0 + 6.3003880987D0*DS50
      TEMP=THETA/TWOPI
      I=TEMP
      TEMP=I
      THETAG=THETA-TEMP*TWOPI
      IF(THETAG.LT.0.D0) THETAG=THETAG+TWOPI
      RETURN
      END
```

12 USERS GUIDE, CONSTANTS, AND SYMBOLS

The first input card is the program card. The format is as follows:

Column	Format	Description
1	I1	Ephemeris program desired 1 = SGP 2 = SGP4 3 = SDP4 4 = SGP8 5 = SDP8
2-11	F10.0	Prediction start time
12-21	F10.0	Prediction stop time
22-31	F10.0	Time increment

All times are in minutes since epoch and can be positive or negative. The second and third input cards consist of either a 2-card transmission or 2-card G type element set. Either type can be used with the only condition being that the two cards must be in the correct order. For reference a format sheet for the T-card and G-card element sets follows this section.

The values of the physical and mathematical constants used in the program are given below.

Variable name	Definition	Value
CK2	$\frac{1}{2}J_2{a_E}^2$	5.413080E-4
CK4	$-\frac{3}{8}J_4{a_E}^4$	.62098875E-6
E6A	10^{-6}	1.0 E-6
QOMS2T	$(q_o - s)^4$ (er)^4	1.88027916E-9
S	s (er)	1.01222928
TOTHRD	2/3	.66666667
XJ3	J_3	-.253881E-5
XKE	$k_e \left(\frac{\text{er}}{\text{min}}\right)^{\frac{3}{2}}$	.743669161E-1
XKMPER	kilometers/Earth radii	6378.135
XMNPDA	time units/day	1440.0

AE	distance units/Earth radii	1.0
DE2RA	radians/degree	.174532925E-1
PI	π	3.14159265
PIO2	$\pi/2$	1.57079633
TWOPI	2π	6.2831853
X3PIO2	$3\pi/2$	4.71238898

where er = Earth radii. Except for the deep-space models, all ephemeris models are independent of units. Thus, units input or output as well as physical constants can be changed by making the appropriate changes in only the DRIVER program.

Following is a list of symbols commonly used in this report.

n_o = the SGP type "mean" mean motion at epoch

e_o = the "mean" eccentricity at epoch

i_o = the "mean" inclination at epoch

M_o = the "mean" mean anomaly at epoch

ω_o = the "mean" argument of perigee at epoch

Ω_o = the "mean" longitude of ascending node at epoch

$\dot{n}_o$ = the time rate of change of "mean" mean motion at epoch

$\ddot{n}_o$ = the second time rate of change of "mean" mean motion at epoch

B^* = the SGP4 type drag coefficient

$k_e = \sqrt{GM}$ where G is Newton's universal gravitational constant and M is the mass of the Earth

a_E = the equatorial radius of the Earth

J_2 = the second gravitational zonal harmonic of the Earth

J_3 = the third gravitational zonal harmonic of the Earth

J_4 = the fourth gravitational zonal harmonic of the Earth

$(t - t_o)$ = time since epoch

$k_2 = \frac{1}{2} J_2 {a_E}^2$

$k_4 = -\frac{3}{8} J_4 {a_E}^4$

$A_{3,0} = -J_3 {a_E}^3$

q_o = parameter for the SGP4/SGP8 density function

s = parameter for the SGP4/SGP8 density function

$B = \frac{1}{2} C_D \frac{A}{m}$, the ballistic coefficient for SGP8 where C_D is a dimensionless drag coefficient and A is the average cross-sectional area of the satellite of mass m

13 SAMPLE TEST CASES

For reference a sample test case is given for each of the five models.[1] The input used was standard T-cards and the output is given at 360 minute intervals in units of kilometers and seconds.

When implemented on a given computer, the accuracies with which the test cases are duplicated will be dominated by the accuracy of the epoch mean motion. If, after reading and converting, the epoch mean motion has an error $\Delta n = j \times 10^{-k}$ radians/time, then the predicted positions at time t may differ from the test cases by numbers on the order of

$$\Delta r = \Delta n(t - t_o)(6,378.135) \text{ kilometers}$$

[1]The test cases were generated on a machine with 8 digits of accuracy. After a one day prediction, the test cases have only 5 to 6 digits of accuracy.

```
1 88888U          80275.98708465  .00073094  13844-3  66816-4 0    8
2 88888  72.8435 115.9689 0086731  52.6988 110.5714 16.05824518  105

SGP  TSINCE              X                Y                Z

      0.             2328.96594238   -5995.21600342    1719.97894287
    360.00000000     2456.00610352   -6071.94232177    1222.95977784
    720.00000000     2567.39477539   -6112.49725342     713.97710419
   1080.00000000     2663.03179932   -6115.37414551     195.73919105
   1440.00000000     2742.85470581   -6079.13580322    -328.86091614


                      XDOT              YDOT              ZDOT

                       2.91110113       -0.98164053       -7.09049922
                       2.67852119       -0.44705850       -7.22800565
                       2.43952477        0.09884824       -7.31889641
                       2.19531813        0.65333930       -7.36169147
                       1.94707947        1.21346101       -7.35499924
```

```
1 88888U          80275.98708465  .00073094  13844-3  66816-4 0    8
2 88888  72.8435 115.9689 0086731  52.6988 110.5714 16.05824518  105

SGP4 TSINCE              X                Y                Z

      0.            2328.97048951   -5995.22076416    1719.97067261
    360.00000000    2456.10705566   -6071.93853760    1222.89727783
    720.00000000    2567.56195068   -6112.50384522     713.96397400
   1080.00000000    2663.09078980   -6115.48229980     196.39640427
   1440.00000000    2742.55133057   -6079.67144775    -326.38095856

                        XDOT             YDOT             ZDOT

                      2.91207230      -0.98341546      -7.09081703
                      2.67938992      -0.44829041      -7.22879231
                      2.44024599       0.09810869      -7.31995916
                      2.19611958       0.65241995      -7.36282432
                      1.94850229       1.21106251      -7.35619372
```

```
1 11801U          80230.29629788  .01431103  00000-0  14311-1
2 11801  46.7916 230.4354 7318036  47.4722  10.4117  2.28537848

SDP4 TSINCE              X                Y                Z

      0.            7473.37066650     428.95261765    5828.74786377
    360.00000000   -3305.22537232   32410.86328125  -24697.17675781
    720.00000000   14271.28759766   24110.46411133   -4725.76837158
   1080.00000000   -9990.05883789   22717.35522461  -23616.89062501
   1440.00000000    9787.86975097   33753.34667969  -15030.81176758


                      XDOT             YDOT             ZDOT

                       5.10715413       6.44468284      -0.18613096
                      -1.30113538      -1.15131518      -0.28333528
                      -0.32050445       2.67984074      -2.08405289
                      -1.01667246      -2.29026759       0.72892364
                      -1.09425066       0.92358845      -1.52230928
```

```
1 88888U          80275.98708465  .00073094  13844-3  66816-4 0    8
2 88888  72.8435 115.9689 0086731  52.6988 110.5714 16.05824518  105

SGP8 TSINCE              X                Y                Z

      0.            2328.87265015   -5995.21289063    1720.04884338
    360.00000000    2456.04577637   -6071.90490722    1222.84086609
    720.00000000    2567.68383789   -6112.40881348     713.29282379
   1080.00000000    2663.49508667   -6115.18182373     194.62816810
   1440.00000000    2743.29238892   -6078.90783691    -329.73434067


                      XDOT             YDOT             ZDOT

                       2.91210661      -0.98353850      -7.09081554
                       2.67936245      -0.44820847      -7.22888553
                       2.43992555       0.09893919      -7.32018769
                       2.19525236       0.65453661      -7.36308974
                       1.94680957       1.21500109      -7.35625595
```

```
1 11801U          80230.29629788  .01431103  00000-0  14311-1
2 11801  46.7916 230.4354 7318036  47.4722  10.4117  2.28537848

SDP8 TSINCE              X                Y                Z

      0.            7469.47631836     415.99390792    5829.64318848
    360.00000000   -3337.38992310   32351.39086914  -24658.63037109
    720.00000000   14226.54333496   24236.08740234   -4856.19744873
   1080.00000000  -10151.59838867   22223.69848633  -23392.39770508
   1440.00000000    9420.08203125   33847.21875000  -15391.06469727


                     XDOT             YDOT             ZDOT

                      5.11402285       6.44403201      -0.18296110
                     -1.30200730      -1.15603013      -0.28164955
                     -0.33951668       2.65315416      -2.08114153
                     -1.00112480      -2.33532837       0.76987664
                     -1.11986055       0.85410149      -1.49506933
```

14 SAMPLE IMPLEMENTATION

These FORTRAN IV routines have been implemented on a Honeywell-6000 series computer. This machine has a processing speed in the 1MIPS range and a 36 bit floating point word providing 8 significant figures of accuracy in single precision. The information in the following table is provided to allow a comparison of the relative size and speed of the different models[1].

Model	core used (words)	CPU time per call (milliseconds) Initialize	Continue
SGP	541	.8	2.7
SGP4	1,041	1.9	2.5
SDP4	3,095	5.1	3.6
SGP8	1,601	1.8	2.2
SDP8	3,149	5.4	3.2

[1] The timing results are for the test cases in Section Thirteen with a one day prediction. Times may vary slightly with orbital characteristics and, for deep-space satellites, with prediction interval.

ACKNOWLEDGEMENTS

The authors would like to express our appreciation to Bob Morris for his helpful comments, to Linda Crawford for supplying one of the function subroutines, and to Don Larson for his help in validating the computer programs. Finally, we wish to acknowledge the skill and patience of Maria Gallegos who accomplished the unenviable task of typing this report.

References

[1] Brouwer, D., "Solution of the Problem of Artificial Satellite Theory without Drag", *Astronomical Journal* **64**, 378–397, November 1959.

[2] Hilton, C.G. and Kuhlman, J.R., "Mathematical Models for the Space Defense Center", Philco-Ford Publication No. U-3871, 17–28, November 1966.

[3] Hoots, F.R., "A Short, Efficient Analytical Satellite Theory". *AIAA* Paper No. 80-1659, August 1980.

[4] Hoots, F.R., "Theory of the Motion of an Artificial Earth Satellite", accepted for publication in *Celestial Mechanics*.

[5] Hujsak, R.S., "A Restricted Four Body Solution for Resonating Satellites with an Oblate Earth", *AIAA* Paper No. 79-136, June 1979.

[6] Hujsak, R.S. and Hoots, F.R., "Deep Space Perturbations Ephemeris Generation", Aerospace Defense Command Space Computational Center Program Documentation, DCD 8, Section 3, 82–104, September 1977.

[7] Kozai, Y., "The Motion of a Close Earth Satellite", *Astronomical Journal* **64**, 367–377, November 1959.

[8] Lane, M.H. and Cranford, K.H., "An Improved Analytical Drag Theory for the Artificial Satellite Problem", *AIAA* Paper No. 69-925, August 1969.

[9] Lane, M.H., Fitzpatrick, P.M., and Murphy, J.J., "On the Representation of Air Density in Satellite Deceleration Equations by Power Functions with Integral Exponents", Project Space Track Technical Report No. APGC-TDR-62-15, March 1962, Air Force Systems Command, Eglin AFB, FL.

[10] Lane, M.H. and Hoots, F.R., "General Perturbations Theories Derived from the 1965 Lane Drag Theory", Project Space Track Report No. 2, December 1979, Aerospace Defense Command, Peterson AFB, CO.