UNIVERSITY OF CALIFORNIA, SAN DIEGO
SCRIPPS INSTITUTION OF OCEANOGRAPHY
VISIBILITY LABORATORY
LA JOLLA, CALIFORNIA 92093

# SOLAR EPHEMERIS ALGORITHM

WAYNE H. WILSON

SIO Ref. 80-13

July 1980

Supported by

U.S. Department of Commerce
National Oceanic and Atmospheric Administration
Grant No. GR-04-6-158-44033

Approved:

Roswell W. Austin, Director
Visibility Laboratory

Approved:

William A. Nierenberg, Director
Scripps Institution of Oceanography

# TABLE OF CONTENTS

# 1. INTRODUCTION

In the course of studies of the optical properties of the atmosphere and hydrosphere, the apparent local azimuth and zenith angle of the sun for a particular time and location are frequently required. Normally this information may be obtained from the Nautical Almanac and the Sight Reduction Tables, although this is in general a slow and time consuming procedure. Without entering a sizeable fraction of the Almanac each year into a computer, it is not feasible to use the computer for this task. However, there have been several algorithms described in the literature for computation of solar azimuth and zenith which have proven very useful (U.S. Nautical Almanac Office, 1978, Lutus, 1977). This paper describes an algorithm presently available in Fortran and HPL (Hewlett-Packard 9825 language).

This algorithm has been developed over the last several years in response to the needs of the author and his colleagues. Several extensions have been made in the algorithm, especially in the increase in accuracy of the results, which are far beyond those required by the present research. The philosophy behind the development of the algorithm, however, was to construct a basic framework to which additions or subtractions affecting the accuracy of the results could be made. As is the case in most algorithms, the more accuracy that is desired, the larger the computer program and the longer the running time. The individual researcher will have to make his own judgement of the accuracy desired and modify the algorithm appropriately. This paper provides some guidance for this decision and the modifications necessary for this task.

In describing the algorithm, it is not the desire of the author to rewrite the books on celestial mechanics and almanacs. Some concepts will be reviewed and defined and others seemingly slighted. It is hoped that the basic algorithm is described in sufficient detail that recourse to other references is not needed. If deeper understanding of a particular item is needed, however, the list of references should be consulted.

The algorithm is based upon equations derived from celestial mechanics (*e.g.* Moulton, 1914) and, most importantly, empirical equations derived for the motion of the sun by Newcomb (1898). It should be noted that the tables in the yearly edition of the Almanac for the solar position are also based strictly on Newcomb's equations and tables. Any reference below to this paper will simply refer to the "tables". A very complete reference to the Nautical Almanac is *The American Ephemeris and Nautical Almanac: Explanatory Supplement* (1961).

# 2. THEORY AND CONVENTIONS

## 2.1. CELESTIAL COORDINATE SYSTEMS

In the following, reference will be made to four types of coordinate systems. These are the *horizon, hour angle, right ascension* and *ecliptic* systems. All of these systems have their origin at the observer. Due to the small radius of the earth compared to its distance to the sun, unless otherwise stated, the origin of the observer is also assumed to be the center of the earth. Each system is based on two reference planes perpendicular to each other. Table 1 gives these planes and the primary parameters defining a point in each system. Figs. 1 to 4 also show these parameters.

**Table 1.**

Coordinate Systems

| System | Reference Planes | | Parameters | |
|---|---|---|---|---|
| horizon | horizon | meridian | zenith angle Z | azimuth A |
| hour angle | celestial equator | hour circle of observer's zenith | declination $\delta_s$ | hour angle h |
| right ascension | celestial equator | hour circle of equinox | declination $\delta_s$ | right ascension $\alpha$ |
| ecliptic | ecliptic | ecliptic meridian of the equinox | latitude $\beta$ | longitude $\lambda$ |

For reference, the *celestial equator* is the plane perpendicular to the direction of the North Celestial Pole. The *meridian* is a plane perpendicular to the equator and passing through the North Pole and the local zenith. The *hour circle* is a plane perpendicular to the celestial equator passing through the North Pole. The *ecliptic* is the plane in which the sun moves, while the *equinoxes* are the points on the celestial equator where the ecliptic plane crosses. The equinox at which the sun passes the celestial equator going from south to north is called the *vernal equinox* and occurs around March 21st. The *equinoctial colure* is the hour circle passing through the equinoxes.

Let O be the origin of the observer and S an arbitrary point on the celestial sphere.

## 2.1.1. Horizon System (Figure 1)

*Zenith Angle, Z,* is the angle between the direction OS and the observer's zenith. The *azimuth, A,* is the angle between the vertical plane of S and the meridian measured from the North Pole to the east.

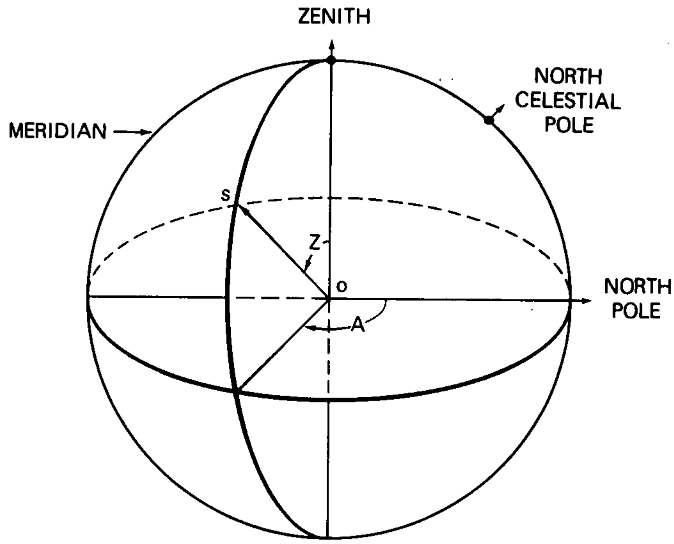



Fig. 1. Horizon System

## 2.1.2. Hour Angle System (Figure 2)

*Declination,* $\delta_s$, is the angle between the direction OS and the celestial equator. It is positive if toward the North Pole and negative if away from it. The *hour angle, h,* is the angle between the hour circle of S and the hour circle of the observer (the celestial meridian). It is measured to the west (*i.e.*, in the direction of S's apparent daily motion).

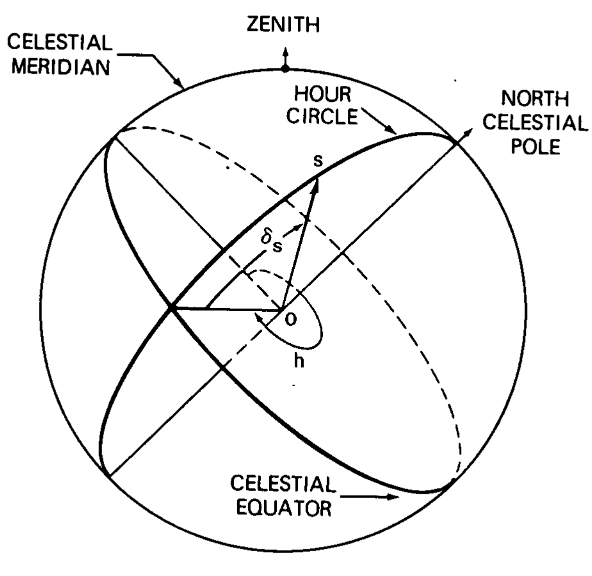



Fig. 2. Hour Angle System

### 2.1.3. Right Ascension System (Figure 3)

*Right Ascension,* $\alpha$, is the angle between the hour circle of S and the hour circle of the vernal equinox (equinoctial colure). It is measured from the vernal equinox to the east in the plane of the celestial equator. The declination, $\delta_s$, is the same as defined in the hour angle system.

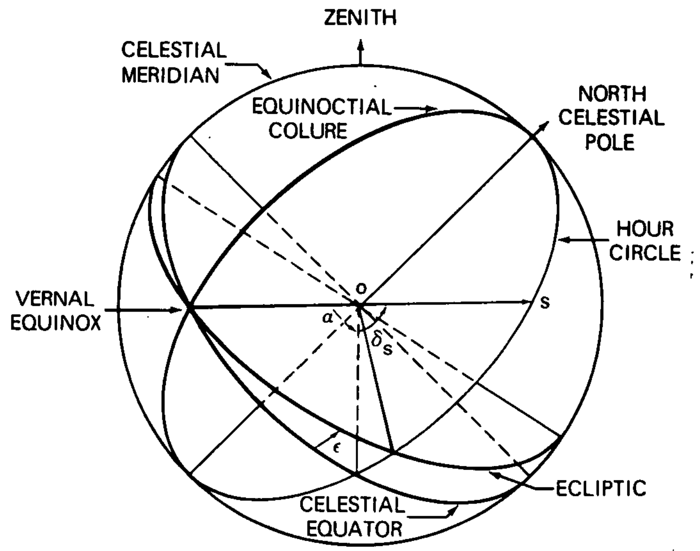



Fig. 3. Right Ascension

### 2.1.4. Ecliptic System (Figure 4)

The (ecliptic) *latitude,* $\beta$, is the angle between the direction OS and the ecliptic measured in the ecliptic meridian of S. It is positive if north of the ecliptic and negative if south. The (ecliptic) *longitude,* $\lambda$, is the angle between the ecliptic meridian of S and that of the equinox meridian (vernal equinox) measured to the east.

The angle between the celestial equator and the ecliptic is referred to as the *obliquity of the ecliptic,* $\epsilon$.

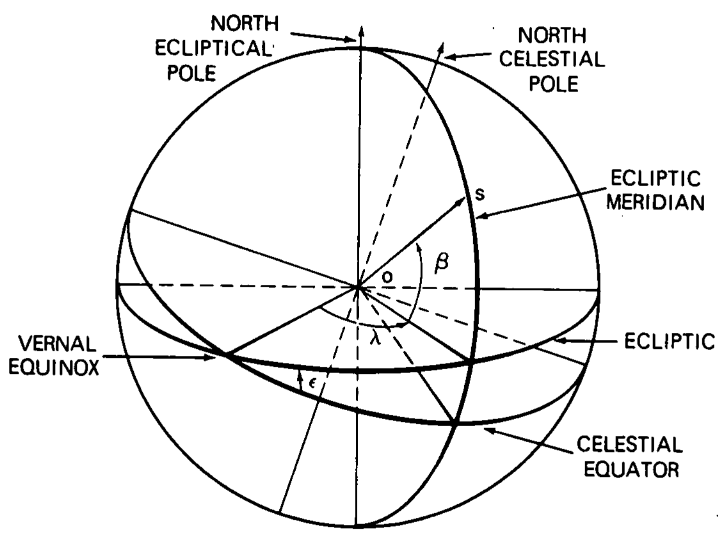



Fig. 4. Ecliptic System

## 2.2. COORDINATE SYSTEM CONVERSION

The basic equations from the "tables" are in terms of the longitude, $\lambda$, and latitude, $\beta$, (ecliptic system) of the sun.

The declination, $\delta_s$, and right ascension, $\alpha$, may be obtained from $\lambda$ and $\beta$ by

$$\sin\delta_s = \cos\beta\sin\lambda\sin\epsilon + \sin\beta\cos\epsilon \quad (1)$$

$$\cos\delta_s\cos\alpha = \cos\beta\cos\lambda \quad (2)$$

$$\cos\delta_s\sin\alpha = \cos\beta\sin\lambda\cos\epsilon - \sin\beta\sin\epsilon \ . \quad (3)$$

In general $\beta$ is on the order of 1" (0°.00028), so that from Eq. (1),

$$\sin\delta_s \approx \sin\lambda\sin\epsilon \ , \quad (4)$$

and from (2) and (3)

$$\tan\alpha \approx \tan\lambda\cos\epsilon \ . \quad (5)$$

To convert $\delta_s$ and $\alpha$ to the hour angle system it is immediately obvious that the declination is the same in both systems. Referring to Fig. 5, where *ST*, the *Local Sideral Time,* is defined as the hour angle of the vernal equinox, then,

$$h = ST - \alpha \ . \quad (6)$$

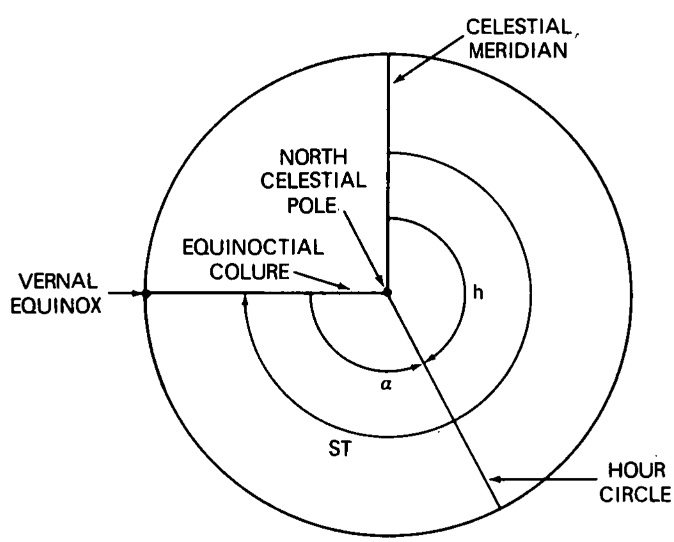



Fig. 5. View of the Plane of the Celestial Equator Viewed from the North Celestial Pole.

Finally converting to the horizon system,

$$\cos Z = \sin\delta_s\sin\phi + \cos\delta_s\cos\phi\cos h \ , \quad (7)$$

and

$$\cos A = \frac{\sin\delta_s\cos\phi - \cos\delta_s\sin\phi\cos h}{\sin Z}\,, \tag{8}$$

where $\phi$ is the astronomic latitude of the observer defined in the hour angle system (*i.e.*, using the celestial equator and the North Pole as references).

## 2.3. SOLAR MOTION IN ECLIPTIC PLANE

The motion of the earth around the sun may be described by an ellipse using Kepler's equations. In Fig. 6, the ellipse APB in which a planetary body moves is plotted with its auxilliary circle AXB. C is the center of the circle while S is one of the foci of ellipse APB.

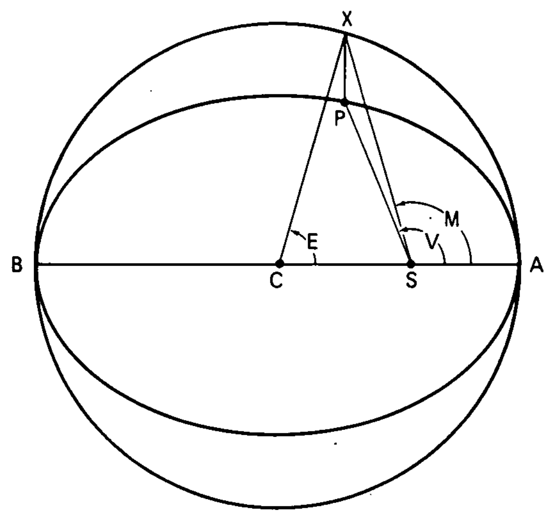



Fig. 6.

Because the body is moving in an ellipse, its motion about C is not uniform (*i.e.*, it speeds up or slows down depending on its orbital position). The angle M is defined as the angle which would have been described by the radius vector if it had moved uniformly with the average rate. It is referred to as the *mean anomaly, M,* and, in Fig. 6, is the angle ASX. The angle ACX is defined as the *eccentric anomaly, E,* while the actual angle (position of the body at P) is ASP and is called the *true anomaly, V.* Notice should be made that the anomalies *M, E,* and *V* are measured with respect to the perhelion of the orbit.

Standard orbital theory (Moulton, 1914) yields the relation

$$M = E - e \cdot \sin E\,, \tag{9}$$

where $e$ is the *eccentricity* of the elliptical orbit. This equation is referred to as Kepler's equation. The quantities $M$ (mean anomaly) and $e$ are given in the "tables" and Kepler's equation may be solved for $E$, the eccentric anomaly. It can further be shown that

$$V = 2 \cdot \tan^{-1}\left[\frac{\sqrt{1+e}}{\sqrt{1-e}} \cdot \tan\left(\frac{E}{2}\right)\right], \tag{10}$$

where again $V$ is the true anomaly.

The quantity $V - M$ is referred to as the *equation of centers* and is described as such in the "tables". Before computers allowed the quick and easy solution of Kepler's equation, much effort was devoted to developing methods of computing the quantity $V - M$.

The position of the sun in the ecliptic with respect to the vernal equinox is the sum of the *mean longitude,* $L$ (*i.e.*, the position of a uniformly moving sun), and the equation of centers, $V - M$. $L$ is again given by the "tables". Thus

$$\lambda = L + (V - M) . \tag{11}$$

## 2.4. PERTURBATIONS OF SOLAR MOTION

If the earth and the sun were strictly a two-body system, Eq. (11) would accurately describe the longitude of the sun. However, due to the gravitational effects of other celestial bodies such as the moon and planets, the celestial equator and ecliptic are continuously in motion.

The motion of the equator contains two components due to the solar and lunar gravitational effects on the earth. The first is a smooth long-period motion of the mean pole of the equator about the pole of the ecliptic with a period of about 26,000 years. This is referred to as *general precession, P.* The other component is the motion of the true pole about the mean pole. This is referred to as *nutation.* Besides these two there are other second order perturbations due to the planets and moon.

### 2.4.1. General Precession

Since the solar position is referenced to the intersection of the ecliptic with the celestial equator (the vernal equinox), the motion of the equinox due to precession will not affect the computation of solar position for a particular date. However, if the solar position needs to be referenced to an earlier or later date when the equinox was at a different location in space, then correction of the solar longitude and latitude for precession needs to be made.

This is the reason that the entries in the Nautical Almanac and the computations described below all refer to the "mean equinox of date". This means that the quantities are referred to the equator and equinox for the particular date specified.

Typically one finds that celestial positions are referred to either the beginning of a year (*i.e.*, mean equinox of 1979.0), or to the standard equinox of 1950.0.* The rate of precession, $\psi$, is approximately

* The notation 19xx.0 specifies the beginning of the year designated. It is actually defined to be the start of the Besselarian year, but for the purposes of this report can be thought of as 1 JAN 19xx 0 GMT. See *Explanatory Supplement* for further discussion.

50″/year. The algorithm described here gives solar position referred to the equinox of date and also computes the precession in longitude from the beginning of the year to date. To compute position with reference to other times, the section in *Explanatory Supplement* on general precession should be consulted.

### 2.4.2. Nutation

Nutation is mainly due to the orbital characteristics of the moon. It affects longitude and the resultant quantity is referred to as *nutation in longitude* $\Delta\psi$. Its effect is on the order of 17″. The lunar motion also affects the obliquity of the ecliptic and is referred to as *nutation in obliquity,* $\Delta\epsilon$. It is on the order of 9″.

The numerical series for $\Delta\psi$ and $\Delta\epsilon$ (Woolard, 1953) consist of 69 and 40 terms respectively. The principle term for longitude has an amplitude of 17″.2327 and obliquity 9″.21, the period being 6798 days (18.02 years). The amplitude 9″.21 is known as the *constant of nutation.* The degree of accuracy required will dictate the number of terms used in the series.

### 2.4.3. Lunar Perturbations

The moon also has an effect on the solar position due to its mass. This effect has the label $\Delta\lambda_m$ for longitude and is on the order of 6″. The similar perturbation for solar latitude is $\Delta\beta$ and is on the order of 1″.

### 2.4.4. Planetary Perturbations

All of the planets have an effect on solar position, mainly on longitude. The perturbations have the form

$$S \cos(K - jg' - ig) \,, \tag{12}$$

where $S$ and $K$ are constants and $g$ and $g'$ are the mean anomalies of the sun and particular planet. A latter section will give tables of $S$ and $K$ for the various planets. They are typically on the order of 1-10″.

Due to their planetary masses, the planets also produce what are termed *inequalities of long period in the mean longitude.* These are referred to by the symbol $\delta L$ and are on the order of 7″.

## 2.5. ABERRATION

Because the velocity of light is finite, it takes approximately 8.3 minutes for the light from the sun to reach earth. This corresponds to a change in solar longitude of approximately 20″.47.

This constant is defined as the *constant of aberration* and the aberration correction is defined as

$$\Delta\lambda_A = -\frac{20''.47}{R}, \tag{13}$$

where $R$ is the *radius vector* which has a value of 1 at the mean solar-earth distance.

## 2.6. REFRACTION

In computing the zenith angle of the sun in the horizon system, it is necessary to correct for atmospheric refraction if the highest degree of accuracy is desired. The effect is such that at the horizon ($Z$=90°) a celestial object approximately 34′ below the celestial horizon is still visible. The earth's atmosphere thus tends to refract light so that extraterrestrial objects appear higher in the sky than if there was no atmosphere.

Since the atmospheric refraction is dependent on the atmospheric thickness and composition, an accurate correction is difficult to make, especially for large zenith angles. Table 2 shows the refractive corrections which need to be added to the observed zenith angle in order to obtain the true or apparent zenith angle.

**Table 2.**

SEA-LEVEL REFRACTION CORRECTIONS FOR ZENITH ANGLES OF ASTRONOMICAL LINES OF SIGHT*

| Observed Zenith (deg) | (min) | Refraction Correction (min) | Observed Zenith (deg) | Refraction Correction (min) |
|---|---|---|---|---|
| 90 | 00 | 34.5 | 79 | 4.9 |
| 89 | 45 | 31.4 | 78 | 4.5 |
| 89 | 30 | 28.7 | 77 | 4.1 |
| 89 | 15 | 26.4 | 76 | 3.8 |
| 89 | 00 | 24.3 | 75 | 3.6 |
| 88 | 45 | 22.5 | 74 | 3.3 |
| 88 | 30 | 20.9 | 73 | 3.1 |
| 88 | 15 | 19.5 | 72 | 2.9 |
| 88 | 00 | 18.3 | 71 | 2.8 |
| 87 | 45 | 17.2 | 70 | 2.6 |
| 87 | 30 | 16.1 | 65 | 2.1 |
| 87 | 15 | 15.2 | 60 | 1.7 |
| 87 | 00 | 14.4 | 55 | 1.4 |
| 85 | 30 | 10.7 | 40 | 0.8 |
| 85 | 00 | 9.9 | 35 | 0.7 |
| 84 | 00 | 8.5 | 30 | 0.6 |
| 83 | 00 | 7.4 | 25 | 0.5 |
| 82 | 00 | 6.6 | 20 | 0.4 |
| 81 | 00 | 5.9 | 10 | 0.2 |
| 80 | 00 | 5.3 | 0 | 0.0 |

*The correction is to be added to the observed zenith angle to obtain the true zenith angle.
*Source:* Nautical Almanac

Equation (14) can be used for the correction and should be subtracted from the zenith angle, $Z$, computed from the solar ephemeris algorithm to obtain the observed zenith angle. Thus,

$$\Delta Z_R = \frac{P}{273 + T}\left[3.430289(Z - \sin^{-1}\{0.9986047 \sin(0.9967614Z)\}) - 0.01115929Z\right], \tag{14}$$

where $P$ is the atmospheric pressure in millibars and $T$ the temperature in Celsius at the observation point.

The present algorithm does not make this correction.

## 2.7. PARALLAX

Due to the fact that Earth has a finite radius, the apparent position of the sun viewed from the earth's surface will be displaced from the position computed from the "tables". These "tables" assume an observing point at the earth's center. This shift, referred to as *parallax*, $\Delta Z_P$, is, for all practical purposes, a small first order correction to the computed zenith angle. It may be computed by the equation

$$\sin\Delta Z_p = \sin\pi \cdot \sin Z \ , \tag{15}$$

where $Z$ is the apparent solar zenith angle and $\pi$ the *horizontal parallax*. For the sun,

$$\pi = \frac{8''.794}{R} \ , \tag{16}$$

where $R$ is the radius vector.

Equation (15) can thus be approximated by

$$\Delta Z_p = \pi \cdot \sin Z \ . \tag{17}$$

As is evident, parallax is a very minor correction amounting to at most about a 8″.9 (0°.0025) shift in zenith angle, and thus can almost always be ignored.

## 2.8. SYSTEMS OF TIME MEASUREMENT

There are fundamentally two different measurements of time, Ephemeris and Universal. When Newcomb first constructed his tables these two were thought to be identical. However, more precise measurements of the earth's years and daily rotation have shown that the two systems are different and that this difference is constantly changing. At the beginning of 1979 their difference was on the order of 50 sec.

### 2.8.1. Ephemeris Time

*Ephemeris Time, ET*, is theoretically uniform, depending for its determination on the laws of dynamics. It is the independent variable in the theory of motion of the sun, moon and planets and is the basis used in the construction of the tables in the Nautical Almanac.

Ephemeris time is measured from an epoch which is designated 1900 January $0^d\ 12^h$ *ET*. This instant is defined to be that time when the mean longitude of the sun, referred to the mean equinox of date, was 279° 41′ 48″.04.

The primary unit of ephemeris time is defined as the interval during which the sun's mean longitude, referred to the mean equinox of date, increases by 360°. The measure of this unit is determined by the coefficient, $T$, measured in centuries of 36525 ephemeris days.

## 2.8.2. Universal Time

*Universal Time, UT,* is the precise measure of time used as the basis for all civil time-keeping: it conforms very closely to the mean diurnal motion of the sun. It is changed at periodic intervals (1 to 2 times per year) in order to stay in synchrony with the changing diurnal rotation of the earth. Universal time may be identified with *Greenwich Mean Time, GMT.*

The difference, $\Delta T$, between Ephemeris and Universal time is constantly changing and is tabulated yearly in the Nautical Almanac. The difference is defined as

$$\Delta T = ET - UT \ . \tag{18}$$

A plot of $\Delta T$ since 1940 is given in Fig. 7.

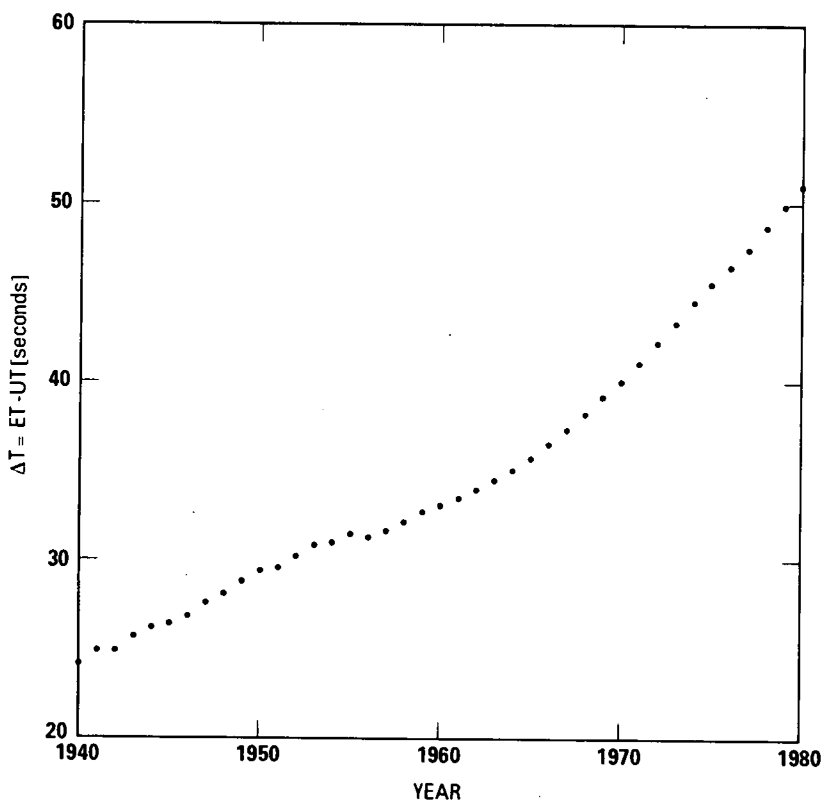



Fig. 7.

In using the algorithm, the following points should be understood.

First, the algorithm technically must use ephemeris time, $ET$, in computing the solar longitude and latitude in order to duplicate the answers in the Nautical Almanac.

Second, given the correct apparent longitude and latitude or right ascension and declination, Universal Time, *UT*, must be used in computing the correct hour angle (which is based on the earth's rotation).

However, in the algorithm it is assumed that $\Delta T$ has little effect on the computed solar longitude and latitude to the degree of accuracy required. Sample calculations have shown that a difference of $\Delta T$ of 40 secs leads to errors in the computed apparent solar longitude on the order of 2″. This error propagated in the computation of the two quantities, declination and equation of time, yields errors on the order of $0''.1$ and $0^s.005$ respectively.

If this type of accuracy is required provision has been made for inclusion of $\Delta T$ in the algorithm.

## 2.9. DATE AND TIME CONVENTIONS

As mentioned in the last section, the algorithm uses the quantity $T$ as an argument in computing the ephemeris. $T$ is the ephemeris time since 1900 January $0^d\ 12^h$ *ET*. In order to facilitate keeping track of dates, *Julian Day Numbers* are used in computing $T$. This number is defined to be 0 for the day starting at Greenwich mean noon on January 1, 4713 B.C.

For 1900 January $0^d\ 12^h$ *ET* the Julian Date is 2415020.0. There are a number of algorithms available to compute the Julian date. The algorithm used by this program has been checked for the years 1750 to 2100. Further extension should be checked by the individual user.

Universal Time, *UT* (or Greenwich Mean Time) is defined as

$$UT = h_m^G + 12^h \, , \tag{19}$$

where $h_m^G$ is the hour angle of the mean sun referred to the Greenwich meridian.

Also,

$$GTT = h_s^G + 12^h \tag{20}$$

is the Greenwich apparent (true) solar time where $h_s^G$ is the hour angle of the apparent (true) sun referred to the Greenwich meridian. Notice should be made of the earlier defined quantity *ST*, Eq. (6). *ST*, the apparent (true) sideral time is equal to $h_s$. The quantity *Eq. T*, called the *equation of time*, is the difference between the true and mean solar times. Thus,

$$Eq.T = GTT - UT = h_s^G - h_m^G \, . \tag{21}$$

For any other meridian located at longitude $\Lambda$, referenced to the Greenwich meridian, *Local Mean Time, LMT*, is defined as

$$LMT = UT - \Lambda = h_m^G + 12^h - \Lambda \, . \tag{22}$$

Note also that the local mean angle of the mean sun, $h_m$, referred to the meridian at $\Lambda$, is

$$h_m = h_m^G - \Lambda \,. \tag{23}$$

Thus,

$$LMT = h_m + 12^h \,. \tag{24}$$

To facilitate civil timekeeping, local mean time is usually referred to standard meridians 15° apart in longitude. Thus, *Local Standard Mean Time, LSMT,* is defined as

$$LSMT = UT + \Delta Z \cdot i \,, \tag{25}$$

where $\Delta Z = 15° = 1^h$, and
i = 1,2,...,12 if meridian to the east of Greenwich,
i = -1, ...,-12 if meridian to the west of Greenwich.

To convert to local mean time, *LMT* from local standard mean time, *LSMT,* the equation

$$LMT = LSMT + 15° \cdot sign(\Lambda) \cdot int\left(\frac{7.5° + |\Lambda|}{15°}\right) - \Lambda \tag{26}$$

is used.

The function *sign*(x) extracts the algebraic sign of x. Thus *sign* (1.3) = 1; *sign* (-1.3) = -1. The function *int*(x) extracts the integral part of x. Thus, *int*(1.3) = 1, while *int*(-1.3) = 1. The function $|x|$ extracts the absolute value of x. Thus |1.3| = 1.3; |-1.3| = 1.3.

In Eq. (26), care should be taken that the proper units are used. Thus *LMT* and *LSMT* are usually expressed in terms of fractions of days, or in hours, minutes and secs., while the 2nd term as written is usually in terms of degrees. All three variables must have the same units.

This equation assumes that the standard zones are 15° apart. In some parts of the world, for example in the continental United States, standard time zones do not follow these 15° intervals. The algorithm as now written *assumes* the 15° intervals. Thus care should be taken in entering the correct standard time.

In computing local zenith and azimuth angles, it is necessary to know the apparent (true) local hour angle, $h_s$. From Eqs. (21) to (26), one can compute $h_s$ given the observer's meridian $\Lambda$, the local standard time *LSMT* and the *Eq. T.* Thus,

$$h_s = Eq.T + LSMT + 15° \cdot sign(\Lambda) \cdot int\left(\frac{7.5° + |\Lambda|}{15°}\right) - \Lambda - 12^h \,. \tag{27}$$

In this equation, the units of all quantities must be similar. Often *Eq. T* and *LSMT* are given in hours, minutes and seconds. The proper conversions need to be made before computing Eq. (27) (*i.e.* $1^d = 24^h = 360°$).

# 3. ALGORITHM DESCRIPTION

The previous section described the theory and conventions behind the algorithm. This section describes the algorithm itself in terms of its logical flow and the various equations and tables used.

There are minor differences in the Fortran and HPL versions, especially in the manner of passing of arguments between the calling program and the algorithm subroutine. These differences will be described in Section 4. It is assumed for this section's discussion that preliminary setup has occurred and that the requisite parameters are in the proper form and units.

To provide some compatability between the Fortran and HPL versions most of the variables have been assigned names using the sequence PXX or pxx where XX is a 2 digit number. This variable name assignment assumes that the "p" variable option is available on the HP calculator.

In this section all equations referenced will use the pxx notation. At the same time, for connection with section 2, the variable names used here will also be noted. The appendix has a listing of all the variables used and a description of each.

All values computed are in degrees and fractions of degrees. For printing purposes, many times values are first converted to the form $xx^{\circ}\, xx'\, xx''$ or $xx^{h}\, xx^{m}\, xx^{s}$. The function routine which does this is called TOHMS.

## 3.1. STEPS

**STEP 1** Compute $p23 = T$, fraction of century from 1900 JAN $0^{d}\ 12^{h}$ *ET*

This routine first computes the number of days since 1900 JAN $0^{d}\ 12^{h}$ *ET*. The algorithm used yields a value of 694038.5 for this date. Thus, this value is subtracted to yield the actual number of days. Because the algorithm does not take proper account of the leap year century years (*i.e.* 1800,1900) additional days need to be added for these and preceeding years. To the number of whole days the standard local time (in days) and the observer's longitude is added to get the correct time; $p22 = d$ since 1900 JAN $0^{d}\ 12^{h}$ *ET*. This is then divided by 36525, the number of days in a century, to obtain the fraction of century, *p23*. Thus,

$$p23 = T = \frac{p22}{36525} \ . \qquad (28)$$

**STEP 2(a)** Compute the mean longitude of sun referenced to mean equinox of date - $p24 = L$

$$p24 = L = 279^\circ.69668 + 0^\circ.98564\,73354 \cdot p22 + 3^\circ.03 \cdot 10^{-4} \cdot p23^2 \tag{29}$$

Note that all values are converted to the range of 0-360° by using the function *mod*(x,360), which is defined as the remainder of the division $\frac{x}{360}$.

**STEP 2(b)** Mean anomaly of sun - $p25 = M$.

$$p25 = M = 358^\circ.47583 + 0^\circ.98560\,02670 \cdot p22 - 0^\circ.00015 \cdot p23^2 - 3^\circ \cdot 10^{-6} \cdot p23^3 . \tag{30}$$

**STEP 2(c)** Eccentricity - $p26 = e$

$$p26 = e = 0.01675104 - 4.18 \cdot 10^{-5} \cdot p23 - 1.26 \cdot 10^{-7} \cdot p23^2 \tag{31}$$

**STEP 3** Compute eccentric anomaly - $p13 = E$ from

$$M = E - e \cdot \sin E , \tag{32}$$

or

$$p25 = p13 - p26 \cdot \sin(p13) .$$

This transcendental equation is solved for $p13$ by successive approximation. When the change in $p13$ is smaller than $10^{-8}$, iteration is stopped.

**STEP 4** Compute true anomaly - $p27 = V$ from

$$V = 2 \cdot \tan^{-1}\left[ \frac{\sqrt{1+e}}{\sqrt{1-e}} \tan\left(\frac{E}{2}\right) \right], \tag{33}$$

if

$$sign(V) \ \# \ sign(E); \text{ Then } V = V + 180^\circ, \tag{34}$$

if

$$V < 0; \text{ Then } V = V + 360°, \tag{35}$$

or

$$p27 = 2 \cdot \tan^{-1}\left[\frac{\sqrt{1+p26}}{\sqrt{1-p26}} \tan\left(\frac{p13}{2}\right)\right],$$

or if

$$sign\,(p27) \ \# \ sign\,(p13); \text{ Then } p27 = p27 + 180°,$$

or if

$$p27 < 0; \text{ Then } p27 = p27 + 360°.$$

**STEP 5(a)** Compute radius vector - $R$

$$R = 1.0 - e \cdot \cos(E) , \tag{36}$$

or

$$R = 1.0 - p26 \cdot \cos(p13)$$

**STEP 5(b)** Compute aberration - $p29 = \Delta\lambda_A$

$$p29 = \Delta\lambda_A = \frac{-20''.47}{R} \cdot \frac{1°}{3600''} . \tag{37}$$

**STEP 5(c)** Compute mean obliquity - $p43 = \epsilon_m$

$$p43 = \epsilon_m = 23°.452294 - 0°.0130125 \cdot p23 - 1°.64 \cdot 10^{-6} \cdot p23 + 5°.03 \cdot 10^{-7} \cdot p23^3 . \tag{38}$$

**STEP 5(d)** Compute mean ascension - $p45 = \alpha_m$

$$p45 = \alpha_m = 279°.6909832 + 0°.98564734 \cdot p22 + 3°.8708 \cdot 10^{-4} \cdot p23^2 . \tag{39}$$

**STEP 6** In this step all perturbations due to the moon are computed.

The variable $p8$ controls the degree of approximation of the algorithm. If $p8 < 2$, the perturbations due to the moon are included in the algorithm.

These require the initial computation of four quantities which are:

moon's mean anomaly - $p28 = l$
moon's mean elongation - $p30 = D$
moon's longitude of ascending mode - $p31 = \Omega$
moon's mean longitude - $p32 = \gamma$.

Note that $D = \gamma - L$ where $L$ = mean longitude of sun.

$$p28 = l = 296^\circ.104608 + 1325 \cdot 360^\circ \cdot p23 + 198^\circ.8491083 \cdot p23 + 0^\circ.00919167 p23^2 + 1^\circ.4388 \cdot 10^{-5} \cdot p23^3 \tag{40}$$

$$p30 = D = 350^\circ.737486 + 1236 \cdot 360^\circ \cdot p23 + 307^\circ.1142167 \cdot p23 - 1^\circ.436 \cdot 10^{-3} \cdot p23^2 \tag{41}$$

$$p31 = \Omega = 259^\circ.183275 - 5 \cdot 360^\circ \cdot p23 - 134^\circ.14200 \cdot p23 + 2^\circ.0778 \cdot 10^{-3} \cdot p23^2 \tag{42}$$

$$p32 = \gamma = 270^\circ.434164 + 1336 \cdot 360^\circ \cdot p23 + 370^\circ.8831417 \cdot p23 - 1^\circ.1333 x 10^{-3} \cdot p23^2 . \tag{43}$$

The perturbation of the earth's orbit due to the mass of the moon is $p33 = \Delta\lambda$,

$$\begin{aligned} \text{where } p33 = & \; 6''.454 \sin D \\ & + 0''.013 \sin 3D \\ & + 0''.177 \sin (D+l) \\ & - 0''.424 \sin (D-l) \\ & + 0''.039 \sin (3D-l) \\ & - 0''.064 \sin (D+M) \\ & + 0''.172 \sin (D-M) . \end{aligned} \tag{44}$$

Note that $D = p30$, $l = p28$, $M = p25$.

The moon also causes nutation of the solar longitude, $\Delta\psi$, and obliquity of the ecliptic, $\Delta\epsilon$. As mentioned earlier, this nutation is in terms of a power series with up to 60 terms. Table 3 has a listing

## Table 3.

Series Terms for Nutation

| Period (days) | Argument Multiple of l | M | F | D | Ω | Longitude Coefficient of sine argument | | Obliquity Coefficient of cosine argument | |
|---|---|---|---|---|---|---|---|---|---|
| 6798 | | | | | +1 | −172327 | −173.7T | +92100 | +9.1T |
| 3399 | | | | | +2 | + 2088 | + 0.2T | − 904 | +0.4T |
| 1305 | −2 | | +2 | | +1 | + 45 | | − 24 | |
| 1095 | +2 | | −2 | | | + 10 | | | |
| 6786 | | −2 | +2 | −2 | +1 | − 4 | | + 2 | |
| 1616 | −2 | | +2 | | +2 | − 3 | | + 2 | |
| 3233 | +1 | −1 | | −1 | | − 2 | | | |
| 183 | | | +2 | −2 | +2 | − 12729 | − 1.3T | + 5522 | −2.9T |
| 365 | | +1 | | | | + 1261 | − 3.1T | | |
| 122 | | +1 | +2 | −2 | +2 | − 497 | + 1.2T | + 216 | −0.6T |
| 365 | | −1 | +2 | −2 | +2 | + 214 | − 0.5T | − 93 | +0.3T |
| 178 | | | +2 | −2 | +1 | + 124 | + 0.1T | − 66 | |
| 206 | +2 | | | −2 | | + 45 | | | |
| 173 | | | +2 | −2 | | − 21 | | | |
| 183 | | +2 | | | | + 16 | − 0.1T | | |
| 386 | | +1 | | | +1 | − 15 | | + 8 | |
| 91 | | +2 | +2 | −2 | +2 | − 15 | + 0.1T | + 7 | |
| 347 | | −1 | | | +1 | − 10 | | + 5 | |
| 200 | −2 | | | +2 | +1 | − 5 | | + 3 | |
| 347 | | −1 | +2 | −2 | +1 | − 5 | | + 3 | |
| 212 | +2 | | | −2 | +1 | + 4 | | − 2 | |
| 120 | | +1 | +2 | −2 | +1 | + 3 | | − 2 | |
| 412 | +1 | | | −1 | | − 3 | | | |
| 13.7 | | | +2 | | +2 | − 2037 | − 0.2T | + 884 | |
| 27.6 | +1 | | | | | + 675 | + 0.1T | | |
| 13.6 | | | +2 | | +1 | − 342 | − 0.4T | + 183 | |
| 9.1 | +1 | | +2 | | +2 | − 261 | | + 113 | −0.1T |
| 31.8 | +1 | | | −2 | | − 149 | | | |
| 27.1 | −1 | | +2 | | +2 | + 114 | | − 50 | |
| 14.8 | | | | +2 | | + 60 | | | |
| 27.7 | +1 | | | | +1 | + 58 | | − 31 | |
| 27.4 | −1 | | | | +1 | − 57 | | + 30 | |
| 9.6 | −1 | | +2 | +2 | +2 | − 52 | | + 22 | |
| 9.1 | +1 | | +2 | | +1 | − 44 | | + 23 | |
| 7.1 | | | +2 | +2 | +2 | − 32 | | + 14 | |
| 13.8 | +2 | | | | | + 28 | | | |
| 23.9 | +1 | | +2 | −2 | +2 | + 26 | | − 11 | |
| 6.9 | +2 | | +2 | | +2 | − 26 | | + 11 | |
| 13.6 | | | +2 | | | + 25 | | | |
| 27.0 | −1 | | +2 | | +1 | + 19 | | − 10 | |
| 32.0 | −1 | | | +2 | +1 | + 14 | | − 7 | |
| 31.7 | +1 | | | −2 | +1 | − 13 | | + 7 | |
| 9.5 | −1 | | +2 | +2 | +1 | − 9 | | + 5 | |
| 34.8 | +1 | +1 | | −2 | | − 7 | | | |
| 13.2 | | +1 | +2 | | +2 | + 7 | | − 3 | |
| 9.6 | +1 | | | +2 | | + 6 | | | |
| 14.8 | | | | +2 | +1 | − 6 | | + 3 | |
| 14.2 | | −1 | +2 | | +2 | − 6 | | + 3 | |
| 5.6 | +1 | | +2 | +2 | +2 | − 6 | | + 3 | |
| 12.8 | +2 | | +2 | −2 | +2 | + 6 | | − 2 | |
| 14.7 | | | | −2 | +1 | − 5 | | + 3 | |
| 7.1 | | | +2 | +2 | +1 | − 5 | | + 3 | |
| 23.9 | +1 | | +2 | −2 | +1 | + 5 | | − 3 | |
| 29.5 | | | | +1 | | − 4 | | | |
| 15.4 | | +1 | | −2 | | − 4 | | | |
| 29.8 | +1 | −1 | | | | + 4 | | | |
| 26.9 | +1 | | −2 | | | + 4 | | | |
| 6.9 | +2 | | +2 | | +1 | − 4 | | + 2 | |
| 9.1 | +1 | | +2 | | | + 3 | | | |
| 25.6 | +1 | +1 | | | | − 3 | | | |
| 9.4 | +1 | −1 | +2 | | +2 | − 3 | | | |
| 13.7 | −2 | | | | +1 | − 2 | | | |
| 32.6 | −1 | | +2 | −2 | +1 | − 2 | | | |
| 13.8 | +2 | | | | +1 | + 2 | | | |
| 9.8 | −1 | −1 | +2 | +2 | +2 | − 2 | | | |
| 7.2 | | −1 | +2 | +2 | +2 | − 2 | | | |
| 27.8 | +1 | | | | +2 | − 2 | | | |
| 8.9 | +1 | +1 | +2 | | +2 | + 2 | | | |
| 5.5 | +3 | | +2 | | +2 | − 2 | | | |

Note: *T* is the fraction of century coefficient defined in the text.

of the terms of this power series. The form of each term in the series is

$$S \cdot \sin(al + bM + cF + dD + e\Omega) \tag{45}$$

for nutation in longitude, and

$$S \cdot \cos(al + bM + cF + dD + e\Omega) \tag{46}$$

for obliquity, where $F = L - \Omega$.

The algorithm as presently implemented uses only 5 terms for longitude and 4 terms for obliquity. If a higher degree of accuracy is desired more terms may be added. The terms presently used are in bold face in Table 3.

The nutation in longitude is $p34 = \Delta\psi$.
The nutation in obliquity is $p35 = \Delta\epsilon$.

The moon also has a perturbation effect on the solar latitude. For the accuracy of the present algorithm, this effect is negligible. For illustrative purposes, however, the perturbation effect is computed in the event higher degrees of accuracy are required.

The moon's mean argument of latitude, *p63*, is first computed by

$$p63 = 11^\circ\!.250889 + 1342 \cdot 360^\circ \cdot p23 + 82^\circ\!.02515 \cdot p23 + 0^\circ\!.003211 \cdot p23^2 . \tag{47}$$

Then the perturbation of latitude due to the moon is

$$\begin{aligned} \Delta\beta = {} & 0''\!.576 \sin(p63) \\ & + 0''\!.016 \sin(p63 + l) \\ & - 0''\!.047 \sin(p63 - l) \\ & + 0''\!.021 \sin(p63 - 2(l - \Omega)) . \end{aligned} \tag{48}$$

**STEP 7** In this step perturbations due to the planets are computed. The variable *p8* again controls the degree of approximation. If $p8 < 1$, the planetary perturbations are included.

The inequalities of the long period in the mean longitude, $\delta L$, caused by the planetary masses are computed from

$$\begin{aligned} p36 = \delta L = {} & 0''\!.266 \sin(31.8^\circ + 119^\circ \cdot p23) \\ & + (1''\!.882 - 0''\!.016 \cdot p23) \sin(57^\circ\!.24 + 150^\circ\!.27 \cdot p23) \\ & + 0''\!.202 \sin(315^\circ\!.0 + 893^\circ\!.3 \cdot p23) \\ & + 1''\!.089 \cdot p23^2 \\ & + 6''\!.4 \sin(231^\circ\!.19 + 20^\circ\!.2 \cdot p23) . \end{aligned} \tag{49}$$

The other perturbations due to the planets all require the mean anomalies of each planet which are as follows:

**VENUS:**

$$p37 = 212^\circ\!.603222 + 162 \cdot 360^\circ \cdot p23 + 197^\circ\!.803875 \cdot p23 + 1^\circ\!.286 \cdot 10^{-3} \cdot p23^2 \tag{50}$$

**MARS:**

$$p38 = 319^\circ\!.529022 + 53 \cdot 360^\circ \cdot p23 + 59^\circ\!.8592194 \cdot p23 + 1^\circ\!.8083 \cdot 10^{-4} \cdot p23^2 \tag{51}$$

**JUPITER:**

$$p39 = 225^\circ\!.3225 + 8 \cdot 360^\circ \cdot p23 + 154^\circ\!.583 \cdot p23 \tag{52}$$

**SATURN:**

$$p40 = 175^\circ\!.613 + 3 \cdot 360^\circ \cdot p23 + 141^\circ\!.794 \cdot p23 \,. \tag{53}$$

The perturbations due to each planet may be computed by using the mean anomalies and the coefficients from Tables 4-7. The coefficients in Tables 4-7 are given in the form of *j, i, S,* and *K*. A single term has the form

$$S \cdot \cos(K - jg' - iM) \,, \tag{54}$$

where $g'$ is the mean anomaly of the planet and $M$ the mean anomaly of the sun.

Only the coefficients in bold face in the tables are used in the present algorithm. These coefficients account for most of the perturbations in longitude. Further discussion of this point is made later.

The perturbation of latitude by the planets is also negligible for the present purposes. However, if needed, a type of latitude correction may be made similar to the longitude correction. Table 8 gives the required coefficients.

## Table 4.

Perturbations by VENUS

| j | i | s | K |
|---|---|---|---|
| | | ″ | ° ′ |
| −1 | +0 | .075 | 296.6 |
| | 1 | **4.838** | **299 6.1** |
| | 2 | .074 | 207.9 |
| | 3 | .009 | 249 |
| −2 | +0 | .003 | 162 |
| | 1 | .116 | 148.9 |
| | 2 | **5.526** | **148 18.8** |
| | 3 | **2.497** | **315 56.6** |
| | 4 | .044 | 311.4 |
| −3 | +2 | .013 | 176 |
| | 3 | **.666** | **177.71** |
| | 4 | **1.559** | **345 15.2** |
| | 5 | **1.024** | **318.15** |
| | 6 | .017 | 315 |
| −4 | +3 | .003 | 198 |
| | 4 | .210 | 206.2 |
| | 5 | .144 | 195.4 |
| | 6 | .152 | 343.8 |
| | 7 | .006 | 322 |
| −5 | +5 | .084 | 235.6 |
| | 6 | .037 | 221.8 |
| | 7 | .123 | 195.3 |
| | 8 | .154 | 359.6 |
| −6 | +6 | .038 | 264.1 |
| | 7 | .014 | 253 |
| | 8 | .010 | 230 |
| | 9 | .014 | 12 |
| −7 | +7 | .020 | 294 |
| | 8 | .006 | 279 |
| | 9 | .003 | 288 |
| −8 | +8 | .011 | 322 |
| | 12 | .042 | 259.2 |
| | 14 | .032 | 48.8 |
| −9 | +9 | .006 | 351 |
| −10 | +10 | .003 | 18 |

## Table 6.

Perturbations by JUPITER

| j | i | s | K |
|---|---|---|---|
| | | ″ | ° ′ |
| +1 | −3 | .003 | 198 |
| | −2 | .163 | 198.6 |
| | −1 | **7.208** | **179 31.9** |
| | 0 | **2.600** | **263 13.0** |
| | +1 | .073 | 276.3 |
| +2 | −3 | .069 | 80.8 |
| | −2 | **2.731** | **87 8.7** |
| | −1 | **1.610** | **109 29.6** |
| | −0 | .073 | 252.6 |
| +3 | −4 | .005 | 158 |
| | −3 | .164 | 170.5 |
| | −2 | **.556** | **82.65** |
| | −1 | .210 | 98.5 |
| +4 | −4 | .016 | 259 |
| | −3 | .044 | 168.2 |
| | −2 | .080 | 77.7 |
| | −1 | .023 | 93 |
| +5 | −4 | .005 | 259 |
| | −3 | .007 | 164 |
| | −2 | .009 | 71 |

## Table 5.

Perturbations by MARS

| j | i | s | K |
|---|---|---|---|
| | | ″ | ° ′ |
| +1 | −2 | .006 | 218 |
| | −1 | **.273** | **217.7** |
| | 0 | .048 | 260.3 |
| +2 | −3 | .041 | 346.0 |
| | −2 | **2.043** | **343 53.3** |
| | −1 | **1.770** | **200 24.1** |
| | 0 | .028 | 148 |
| +3 | −4 | .004 | 284 |
| | −3 | .129 | 294.2 |
| | −2 | **.425** | **338.88** |
| | −1 | .008 | 7 |
| +4 | −4 | .034 | 71.0 |
| | −3 | **.500** | **105.18** |
| | −2 | **.585** | **344.06** |
| | −1 | .009 | 325 |
| +5 | −5 | .007 | 172 |
| | −4 | .085 | 54.6 |
| | −3 | .204 | 100.8 |
| | −2 | .003 | 18 |
| +6 | −5 | .020 | 186 |
| | −4 | .154 | 227.4 |
| | −3 | .101 | 96.3 |
| +7 | −6 | .006 | 301 |
| | −5 | .049 | 176.5 |
| | −4 | .106 | 222.7 |
| +8 | −7 | .003 | 72 |
| | −6 | .010 | 307 |
| | −5 | .052 | 348.9 |
| | −4 | .021 | 215.2 |
| +9 | −7 | .004 | 57 |
| | −6 | .028 | 298 |
| | −5 | .062 | 346.0 |
| +10 | −7 | .005 | 68 |
| | −6 | .019 | 111 |
| | −5 | .005 | 338 |
| +11 | −7 | .017 | 59 |
| | −6 | .044 | 105.9 |
| +12 | −7 | .006 | 232 |
| +13 | −8 | .013 | 184 |
| | −7 | .045 | 227.8 |
| +15 | −9 | .021 | 309 |
| +17 | −10 | .004 | 243 |
| | −9 | .026 | 113 |

## Table 7.

Perturbations by SATURN

| j | i | s | K |
|---|---|---|---|
| | | ″ | ° |
| +1 | −2 | .011 | 105 |
| | −1 | **.419** | **100.58** |
| | 0 | **.320** | **269.46** |
| | +1 | .008 | 270 |
| +2 | −2 | .108 | 290.6 |
| | −1 | .112 | 293.6 |
| | 0 | .017 | 277 |
| +3 | −2 | .021 | 289 |
| | −1 | .017 | 291 |
| +4 | −2 | .003 | 288 |

## Table 8.

Latitude Perturbations by VENUS

| j | i | s | K |
|---|---|---|---|
| | | ″ | ° |
| −1 | +0 | .029 | 145 |
| | 1 | .005 | 323 |
| | 2 | **.092** | **93.7** |
| | 3 | .007 | 262 |
| −2 | +1 | .023 | 173 |
| | 2 | .012 | 149 |
| | 3 | **.067** | **123.0** |
| | 4 | .014 | 111 |
| −3 | +2 | .014 | 201 |
| | 3 | .008 | 187 |
| | 4 | **.210** | **151.8** |
| | 5 | .007 | 153 |
| | 6 | .004 | 296 |
| −4 | +3 | .006 | 232 |
| | 5 | .031 | 1.8 |
| | 6 | .012 | 180 |
| −5 | +6 | .009 | 27 |
| | 7 | .019 | 18 |
| −6 | +5 | .006 | 288 |
| | 7 | .004 | 57 |
| | 8 | .004 | 57 |
| −8 | +12 | .010 | 61 |

Latitude Perturbations By MARS

| j | i | s | K |
|---|---|---|---|
| | | ″ | ° |
| +2 | −2 | .008 | 90 |
| | 0 | .008 | 346 |
| +4 | −3 | .007 | 188 |

Latitude Perturbations By JUPITER

| j | i | s | K |
|---|---|---|---|
| | | ″ | ° |
| +1 | −2 | .007 | 180 |
| | −1 | .017 | 273 |
| | 0 | .016 | 180 |
| | +1 | .023 | 268 |
| +2 | −1 | **.166** | **265.5** |
| +3 | −2 | .006 | 171 |
| | −1 | .018 | 267 |

Latitude Perturbations By SATURN

| j | i | s | K |
|---|---|---|---|
| | | ″ | ° |
| +1 | −1 | .006 | 260 |
| | +1 | .006 | 280 |

**STEP 8(a)** Computation of precession - *p42*

The precession is defined as the distance the equinox has moved from the beginning of the year. The *rate of precession, p,* is

$$p = 50''.2564 + 0''.0222 \cdot p23 \ . \tag{55}$$

Thus the precession is,

$$p42 = p \cdot (time\ since\ beginning\ of\ year) \ . \tag{56}$$

**STEP 8(b)** Computation of apparent (true) longitude - *p41* = λ

The apparent longitude of the sun is the solar longitude measured from the mean equinox of date apparent at the earth's surface, ignoring refraction. Thus

$$\lambda = (V - M) + L + \Delta\lambda_A + \delta L + \Delta\lambda + \Delta\psi \ , \tag{57}$$

or

$$p41 = (p27 - p25) + p24 + p29 + p33 + p36 + p34 \ .$$

**STEP 8(c)** Computation of obliquity *p75* = ϵ

$$\epsilon = \dot{\epsilon}_m + \Delta\epsilon \ , \tag{58}$$

or

$$p75 = p43 + p35 \ .$$

**STEP 8(d)** Computation of apparent right ascension - *p44* = α

From equation 5)

$$\alpha = \tan^{-1}(\tan\lambda \cdot \cos\epsilon) \ , \tag{59}$$

if

$$sign\ \alpha \ \# \ sign\ \lambda \ ; \text{the } \alpha = \alpha + 180° , \tag{60}$$

if

$$\alpha < 0 \text{ Then } \alpha = \alpha + 360° ,$$

or

$$p44 = \tan^{-1}(\tan(p41) \cdot \cos(p43)) \ ,$$

if

$$sign(p44) \ \# \ sign(p41); \text{ Then } p44 = p44 + 180° ,$$

or if

$$p44 < 0 \quad \text{Then } p44 = p44 + 360^\circ .$$

**STEP 8(e)** Computation of equation of time - *p46* = *Eq. T.*

$$Eq.\ T = \alpha_m - \alpha \ , \tag{61}$$

if

$$Eq.\ T > 180^\circ; \ \text{Then } Eq.\ T = Eq.\ T - 360^\circ , \tag{62}$$

or

$$p46 = p45 - p44 \ ,$$

or if

$$p46 > 180^\circ; \quad \text{Then } p46 = p46 - 360^\circ .$$

**STEP 8(f)** Computation of hour angle - *p48* = $h_m$

From Eq. (26)

$$p48 = p21 \cdot 360 + 15 \cdot int \left( \frac{7.5 + p20}{15} \right) \cdot sign\ (p2) - p20 - 180 \tag{63}$$

where $p21$ = local standard time in fractions of days

$p20$ = absolute value of longitude

$p2$ = longitude .

**STEP 8(g)** Computation of local apparent hour angle - *p49* = $h_s$

$$h_s = Eq.\ T + h_m \tag{64}$$

or

$$p49 = p46 + p48 \ .$$

**STEP 8(h)** Computation of declination - *p47* = $\delta_s$ from Eq. (1)

$$\delta_s = \sin^{-1} \left[ \cos\beta \sin\lambda \sin\epsilon + \sin\beta \cos\epsilon \right] , \tag{65}$$

or

$$p47 = \sin^{-1}\left[\cos(p60)\sin(p41)\sin(p75) + \sin(p60)\cos(p75)\right].$$

**STEP 8(i)** Computation of zenith angle - $Z$ from Eq. (7)

$$Z = \cos^{-1}\left[\sin\delta_s\sin\phi + \cos\delta_s\cos\phi\cos h_s\right], \quad (66)$$

or

$$Z = \cos^{-1}\left[\sin(p47)\sin(p19) + \cos(p47)\cos(p19)\cos(p49)\right],$$

where $p19$ = latitude of observer.

**STEP 8(j)** Computation of azimuth - $A$ from Eq. (8)

$$A = \cos^{-1}\left[\frac{\sin\delta_s\cos\phi - \cos\delta_s\sin\phi\cos h_s}{\sin Z}\right] \quad (67)$$

if

$$sign\left[\frac{-\cos\delta_s\sin h_s}{\sin Z}\right] \# \; sign(A); \text{ Then } A = 360 - A, \quad (68)$$

or

$$A = \cos^{-1}\left[\frac{\sin(p47)\cos(p19) - \cos(p47)\sin(p19)\cos(p49)}{\sin Z}\right],$$

or if

$$sign\left[\frac{-\cos(p47)\sin(p49)}{\sin Z}\right] \# \; sign(A); \text{ Then } A = 360 - A.$$

The longitude tabulated in the Nautical Almanac is the apparent longitude minus the sum of the aberration and the nutation of longitude. Therefore, the tabulated quantity is

$$\lambda - (\Delta\lambda_A + \Delta\psi), \quad (69)$$

or

$$p41 - (p29 + p34).$$

## 3.2. ALGORITHM ACCURACY

The accuracy of the solar ephemeris algorithm depends ultimately on the number of terms used for the perturbation effects. In order to give some insight into the degree of accuracy achievable, this section will explore the effect of the various component parts on the final result.

For our specific requirements, the value of major concern is the apparent zenith angle $Z$. This value, computed from Eq. (7), is a function of declination $\delta_s$, observer latitude $\phi$ and solar hour angle $h_s$. Thus

$$\cos Z = \sin\delta_s \sin\phi + \cos\delta_s \cos\phi \cos h \ . \qquad (70)$$

We also know

$$h_s = h_m + Eq.\ T = h_m + \alpha_m - \alpha \ , \qquad (71)$$

or, combining known quantities,

$$h_s = LMT - \alpha - \Lambda + C \ . \qquad (72)$$

Thus taking derivatives and assuming the maximum possible error for each component, one obtains

$$\Delta Z \approx \Delta\delta_s + \Delta\alpha + \Delta LMT + \Delta\Lambda + \Delta\phi \ . \qquad (73)$$

It should be noted that the maximum possible error will not occur simultaneously for each of the components. Thus the maximum error $\Delta\Lambda$ will occur when $\phi$=0, while the maximum error $\Delta\delta_s$ will occur when $\phi$=0, $\delta_s$=0 but $h_s$ = 0° or 90°. By doing the analysis in this manner however, the relative importance of each component is illustrated.

The present requirements for the algorithm have been to compute $\Delta Z$ to within 0°.1 or 6′. If this error is then divided proportionaly among the five components, each must have maximum errors of 0°.02 or 72″.

### 3.2.1. Latitude and Longitude

Since 0°.02 of latitude is 1.2 nautical miles, this fixes the required latitude determination. A longitude increment of 0.°02, in terms of surface distance is given by

$$\frac{0^\circ.02}{\cos\phi} \ . \qquad (74)$$

However, the maximum error $\Delta\Lambda$ is proportional to cos$\phi$, so that again a determination of $\Lambda$ to within 1.2 nautical miles will give the requisite accuracy.

### 3.2.2. Time

A change in arc degrees of 0°.02 in zenith angle is equivalent to 4.8 seconds of time. Again, this is proportional to cos$\phi$, so that at higher latitudes this requirement is relaxed.

### 3.2.3. Declination

Declination is computed using Eq. (4). Thus,

$$\sin\delta_s = \sin\lambda \cdot \sin\epsilon \ , \qquad (75)$$

and therefore

$$\Delta\delta_s = \left( \frac{\sin\lambda\cos\epsilon}{\cos\delta_s} \right) \Delta\epsilon + \left( \frac{\cos\lambda\sin\epsilon}{\cos\delta_s} \right) \Delta\lambda \ . \quad (76)$$

Since $\epsilon$, the obliquity is on the order of 23°.5, the maximum error for $\Delta\epsilon + \Delta\lambda$ is on the order of

$$\Delta\delta_s = \Delta\epsilon + 0.4x\Delta\lambda \ . \quad (77)$$

For $\Delta\delta_s = 0°.02$, and dividing maximum error proportionaly

$$\Delta\epsilon = 36'' \quad (78)$$

and

$$\Delta\lambda = 90'' = 1'\,30'' \ .$$

### 3.2.4. Right Ascension

From Eq. (5), right ascension $\alpha$ is computed as

$$\tan\alpha = \tan\lambda \cdot \cos\epsilon \ , \quad (79)$$

or

$$\Delta\alpha = \frac{\sec^2\lambda\cos\epsilon}{\sec^2\alpha} \cdot \Delta\lambda + \frac{\tan\lambda\sin\epsilon}{\sec^2\alpha} \cdot \Delta\epsilon \ . \quad (80)$$

Putting in values for maximum error, the relationship

$$\Delta\alpha = \Delta\lambda + 0.7 \cdot \Delta\epsilon \quad (81)$$

is obtained.

For $\Delta\alpha = 0°.02$ and dividing the maximum error proportionally,

$$\Delta\epsilon = 90'' = 1'30'' \quad (82)$$

and

$$\Delta\lambda = 36'' .$$

### 3.2.5. Obliquity and Solar Longitude

The analysis of errors due to declination and right ascension shows that in general for $\Delta Z = 0°.1$, the error in obliquity $\Delta\epsilon$ should be less than 36″. This is also true for the error in solar longitude $\Delta\lambda$.

The major source of error in obliquity is the inclusion of the nutation of obliquity $\Delta\epsilon$. Addition of all of the coefficients for nutation of obliquity found in Table 3 gives the sum 10″.04. If the four highest terms are summed the value 9″.94 is obtained.

Thus the maximum possible error in obliquity is not 36″ but on the order of 10″. If the four largest terms for nutation of obliquity are used this reduces the maximum error to 0″.1. Therefore the maximum error for longitude $\Delta\lambda$ may be increased to 62″ or 72″.

The errors in the computation of longitude are listed in Table 9 along with the corrections used in the present algorithm. The errors are computed by adding the coefficients given in Tables 3 and 4.

**Table 9.**

| | All Corrections | Algorithm Corrections | # of Terms |
|---|---|---|---|
| Nutation of Long $\Delta\psi$ | 19".36 | 19".03 | 5 |
| Moon perturbation of long $\Delta\lambda_m$ | 7".34 | 7".34 | 7 |
| Inequalities of Long Period $\delta L$ | 9".84 | 9".84 | 5 |
| Perturbations of Planets | | | |
| VENUS | 17".57 | 16".11 | 6 |
| MARS | 7".02 | 5".60 | 6 |
| JUPITER | 15".65 | 14".71 | 5 |
| SATURN | 1".04 | 0".74 | 2 |
| TOTAL | 77".82 | 73".37 | |

It can be seen from Table 9 that the maximum error $\Delta\lambda$ is about 78". This is just 16" higher than the maximum allowable error of 62" from $\Delta Z = 0°.1$. Since this is the maximum allowable error, the computation of $\lambda$ without any nutation and perturbation corrections should allow $Z$ to be computed to within 0°.1 under most conditions.

The addition of the nutation terms in the obliquity and longitude calculations using only those 9 terms used in the present algorithm will reduce the maximum allowable error in longitude to 49" which is well within the requirement.

The use of all the terms for nutation and perturbations in the present algorithm reduces the error in longitude computation on the order of 4".5. This degree of accuracy is far greater than is needed for most applications. Their inclusion has been merely an illustration of the technique required for a highly accurate solar ephemeris.

### 3.2.6. Sample Results

To show the accuracy of the algorithm, the ephemeris for four days have been computed and tabulated below. They are:

| | |
|---|---|
| Table 10 | 1786 MAY 3, $17^h\,30^m$ GMT |
| Table 11 | 1960 MAR 1, $0^h$ GMT |
| Table 12 | 1979 JAN 1, $0^h$ GMT |
| Table 13 | 1979 JUL 1, $0^h$ GMT |

The first date was chosen because Newcomb used this date as an example as to how to use the "tables". The second date is used in the *Supplement* as an example, and the last two are merely illustrative of the accuracy for the year 1979.

For each date the computations have been made for the approximations noted in the sections above. The first uses the full algorithm, the second excludes planetary effects, while the third excludes lunar and planetary effects.

**Table 10.** 1786 MAY 4 $5^h30^m$ GMT*

| | Nautical Almanac | Algorithm Full | Algorithm no planetary effects | Algorithm no lunar effects |
|---|---|---|---|---|
| Longitude for Mean Equinox of Date | 43° 50'49".5 | 43° 50'51".7 | 43° 50'57".3 | 43° 50'54".6 |
| Reduction to Apparent Longitude | -6".02 | -6".11 | -6".11 | -20".28 |
| Latitude for ecliptic of Date | - | -0".01 | 0".02 | - |
| Precession in Longitude from 1786.0 to Date | 17".06 | 17".22 | 17".22 | 17".22 |
| Nutation in Longitude | 14".26 | 14".17 | 14".17 | - |
| Nutation in Obliquity | 4".13 | 4".30 | 4".30 | - |
| Obliquity of Ecliptic | 25° 28'05.".8 | 23° 28'05".8 | 23° 28'05".8 | 23° 28'01".5 |
| Apparent Right Ascension | - | $2^h45^m31^s.6$ | $2^h45^m20^s.1$ | $2^h45^m31^s.0$ |
| Apparent Declination | - | 16° 00'50".0 | 16° 00'51".8 | 16° 00'43".9 |
| Radius Vector | 1.0093 | 1.0093 | 1.0093 | 1.0093 |
| ET of Ephemeris Transit | - | $11^h50^m30^s.3$ | $11^h50^m30^s.6$ | $11^h56^m29^s.6$ |

*Note that previous to 1925 GMT was measured from Greenwich noon. Thus Newcomb computed ephemeris for 1786 May 3 $17^h30^m$ GMT

**Table 11.** 1960 MARCH 7 $0^h$ GMT

| | Nautical Almanac | Algorithm Full | Algorithm no planetary effects | Algorithm no lunar effects |
|---|---|---|---|---|
| Longitude for Mean Equinox of Date | 346° 26'23".5 | 346° 26'24".3 | 346° 26'04".5 | 346° 26'01".9 |
| Reduction to Apparent Longitude | -21".37 | -21".45 | -21".45 | -20".62 |
| Latitude for ecliptic of Date | -0".65 | -0".61 | -0".57 | - |
| Precession in Longitude from 1960.0 to Date | 9".04 | 9".01 | 9".01 | 9".01 |
| Nutation in Longitude | -0".74 | -0".82 | -0".82 | - |
| Nutation in Obliquity | -8".84 | -8".89 | -8".89 | - |
| Obliquity of Ecliptic | 23° 26'31".2 | 23° 26'31".2 | 23° 26'31".2 | 23° 26'40".0 |
| Apparent Right Ascension | $23^h10^m04^s.1$ | $23^h10^m04^s.1$ | $23^h10^m02^s.9$ | $23^h10^m02^s.9$ |
| Apparent Declination | -5° 21'16".3 | -5° 21'16".0 | -5° 21'23".7 | -5° 21'25".7 |
| Radius Vector | .9925 | .9925 | .9925 | .9925 |
| ET of Ephemeris Transit | $12^h11^m05^s.8$ | $12^h11^m05^s.7$ | $12^h11^m04^s.5$ | $12^h11^m04^s.5$ |

**Table 12.** 1979 JAN 1 $0^h$ GMT

| | Nautical Almanac | Algorithm Full | Algorithm no planetary effects | Algorithm no lunar effects |
|---|---|---|---|---|
| Longitude for Mean Equinox of Date | 279° 58'14".90 | 279° 58'16".3 | 279° 58'16".7 | 279° 58'18".6 |
| Reduction to Apparent Longitude | -22".86 | -22".88 | -22".88 | -20".82 |
| Latitude for ecliptic of Date | 0".60 | 0".21 | 0".38 | - |
| Precession in Longitude from 1979.0 to Date | +0".007 | -0".03 | -0".03 | -0".03 |
| Nutation in Longitude | -2".047 | -2".059 | -2".059 | - |
| Nutation in Obliquity | -9".743 | -9"724 | -9".724 | - |
| Obliquity of Ecliptic | 23° 20'21".467 | 23° 26'21".5 | 23° 26'21".5 | 23° 26'31".3 |
| Apparent Right Ascension | $18^h43^m21^s.66$ | $18^h43^m21^s.76$ | $18^h43^m21^s.8$ | $18^h43^m22".1$ |
| Apparent Declination | -23° 03'53".8 | -23° 03'54".1 | 23° 03'53".9 | -23° 04'03".6 |
| Radius Vector | 0.9833336 | .9833 | .9833 | .9833 |
| ET of Ephemeris Transit | $12^h03^m23^s.61$ | $12^h03^m23^s.5$ | $12^h03^m23^s.5$ | $12^h03^m23^s.8$ |

**Table 13.** 1979 JUL 1 $0^h$ GMT

| | Nautical Almanac | Algorithm Full | Algorithm no planetary effects | Algorithm no lunar effects |
|---|---|---|---|---|
| Longitude for Mean Equinox of Date | 98° 35'43".40 | 98° 35'43".4 | 98° 35'43".8 | 98° 35'42".6 |
| Reduction to Apparent Longitude | -25".27 | -25".33 | -25".33 | -20".13 |
| Latitude for ecliptic of Date | +".02 | -".15 | 0".12 | - |
| Precession in Longitude from 1979.0 to Date | -25".353 | -24".88 | 24".88 | -24".88 |
| Nutation in Longitude | -5".139 | -5".19 | -5".19 | - |
| Nutation in Obliquity | -9".271 | -9".25 | -9".25 | - |
| Obliquity of Ecliptic | 23° 26'21".747 | 23° 26'21".8 | 23° 26'21".8 | 23° 26'31".0 |
| Apparent Right Ascension | $6^h37^m23^s.45$ | $6^h37^m23^s.45$ | $6^h37^m23^s.5$ | $6^h37^m23^s.8$ |
| Apparent Declination | 23° 09'40".0 | 23° 09'40".02 | 23° 09'40".1 | 23° 09'48".9 |
| Radius Vector | 1.0166819 | 1.0167 | 1.0167 | 1.0167 |
| ET of Ephemeris Transit | $12^h03^m40^s.62$ | $12^h03^m40^s.2$ | $12^h03^m40^s.3$ | $12^h03^m40^s.6$ |

# 4. EPHEMERIS SUBROUTINE LISTINGS

The previous sections have described the theory, and Section 3 in particular has outlined the steps of the algorithm. This section presents listings of the ephemeris routines and discusses their calling sequence. Section 5 will provide listings of main programs and examples of their use.

## 4.1. FORTRAN VERSION - EPHEMS

The FORTRAN version presented in Listing 1 has been written using IBM FORTRAN IV and has been used on an IBM 360/44 with FORTRAN compiler version F. The coding is such that the routine should be readily transportable to other machines. It has been used on a PRIME 500 with no changes other than reassignment of I/O FORTRAN unit numbers in the main program.

**LISTING 1.**

```
      SUBROUTINE EPHEMS (LAT,LONG,DAY,MONTH,YEAR,TIME,P5,P6,P7,P8,
     *  A,R,Z)
C
C       SUBROUTINE TO COMPUTE SOLAR POSITION
C
C      WRITTEN BY WAYNE H WILSON, JR.
C      VISIBILITY LABORATORY
C      SCRIPPS INSTITUTION OF OCEANOGRAPHY
C      UNIVERSITY OF CALIFORNIA, SAN DIEGO
C      LA JOLLA, CALIFORNIA, 92093
C      PH. (714)-294-5534
C      25 FEB 1980
C
      IMPLICIT REAL*8(P)
      IMPLICIT INTEGER*4(I-N)
      REAL*4 LAT,LONG,TIME
      REAL*8 DARSIN,DARCOS,DTAN
      REAL*4 A, R, Z
      REAL*8 DEGRD,PI,TOHMS,ONE
      INTEGER*4 P5,P6,P7,P8,P10
      INTEGER*4 DAY,MONTH,YEAR
C
C     LAT - LATITUDE (IN DEGREES AND FRACTION OF DEGREES), P19
C     LONG - LONGITUDE (IN DEGREES AND FRACTIONS OF DEGREES),P20
C     DAY - DAY OF THE MONTH,P16
C     MONTH - MONTH OF YEAR,P17
C     YEAR - YEAR,P18
C     TIME - TIME (HHMM.SS) LOCAL STANDARD LST OR GMT, P4
C     P5 - TIME TYPE (0 - LST, 1 - GMT)
```

```
C      P6 - APPARENT NOON CALCULATIONS (1-YES, 0-NO)
C      P7 - PRINT (0-NONE, 1-DATE, 2-DATA, 3 - NA COMPAR)
C      P8 - APPROXIMATION (0-FULL, 1-NO PLANETS, 2-NO LUNAR)
C
C       RETURNED VARIABLES
C
C       A - AZIMUTH ANGLE(IN DEGREES AND FRACTION OF DEGREE)
C      R - RADIUS VECTOR
C      Z - ZENITH ANGLE(IN DEGREES AND FRACTION OF DEGREE)
C      TIME - APPARENT NOON IF P6 = 1 (LST OR GMT SEE P5)
C
C      P21 - TIME (IN FRACTIONS OF DAY)
C      P51 - GMT IN DEGREES
C      P53 - ZONE LONGITUDE
C
C
         DATA ONE/1.0D0/
C
C
       DTAN(P)=DSIN(P)/DCOS(P)
       DARSIN(P)=DATAN2(P,DSQRT(1.0D0-P**2))
       DARCOS(P) = PID2 - DARSIN(P)
C
C
C--------------------------------------------------------------------
C--------------------------------------------------------------------
C
C
         PID2 = DATAN(1.0D0)*2.0D0
         PI = PID2*2.0D0
       DEGRD = PI/180D0
C
C++++++++++++++++++++++++++++++++++++++++++++
C       P57 - (ET-UT), DIFFERENCE OF EPHEMERIS & UNIVERSAL TIME
       P57 = 0
C
C
       P16 = DAY
       P17 = MONTH
       P18 = YEAR
       P19 = LAT
       P20 = LONG
       P54 = 10
       P55 = 14
       P10 = P7
     1 CONTINUE
       P53 = 15.*AINT((7.5+ABS(LONG))/15.0)*SIGN(1.0,LONG)
       P11 = P53
       IF(P5.EQ.1) P11=0D0
       IF(P6.NE.1) GO TO 10
       P56 = (P54+P55)/2D0
       P51 = P56/24D0*360D0+P53
       GO TO 20
    10 CONTINUE
       P21 = (AMOD(TIME,1.0)*100/3600+AMOD(AINT(TIME)/100.,1.0)*100./60.
      *     + AINT(AINT(TIME)/100.0))/24.0
       P51 = P21*360D0+P11
    20 CONTINUE
       I = P18
       J = P17+1
       IF(J.GT.3) GO TO 40
       J = J + 12
```

```
      I = I - 1
   40 CONTINUE
      P13 = INT(I*365.25) + INT(J*30.6) + P16
      P22 = P13 - 694038.5D0
      P11 = P18 + P17/100.D0 + P16/1.0D4
      IF(P11.LT.1900.0228D0) P22 = P22 + 1
      IF(P11.LT.1800.0228D0) P22 = P22+1
      P23 = (P51/360D0+P22+P57)/36525D0
      P22 = P23*36525D0
C
C      MEAN LONGITUDE - P24
C
      P11 = 279.69668D0
      P12 = 0.9856473354D0
      P13 = 3.03D-4
      P24 = P11 + DMOD(P12*P22,360D0) + P13*P23**2
      P24 = DMOD(P24,360D0)
C
C     MEAN ANOMALY - P25
C
      P11 = 358.47583D0
      P12 = 0.985600267D0
      P13 = -1.5D-4
      P14 = -3.D-6
      P25 = P11 + DMOD(P12*P22,360D0) + P13*P23**2 + P14*P23**3
      P25 = P11 + DMOD(P12*P22,360D0) + P13*P23**2 + P14*P23**3
      P25 = DMOD(P25,360D0)
C
C     ECCENTRICITY - P26
C
      P11 = 0.01675104D0
      P12 = -4.18D-5
      P13 = -1.26D-7
      P26 = P11 + P12*P23 + P13*P23**2
      P11 = P25*DEGRD
      P12 = P11
C
C     ECCENTRIC ANOMALY - P13 (TEMP)
C
  100 CONTINUE
      P13 = P12
      P12 = P11 + P26*DSIN(P13)
      IF(DABS((P12-P13)/P12).GT.1.0D-8) GO TO 100
      P13 = P12/DEGRD
C
C     TRUE ANOMALY - P27
C
      P27 =2.0D0*DATAN(DSQRT((1.0D0+P26)/(1.0D0-P26))*DTAN(P13/2.0D0
     *   *DEGRD))/DEGRD
      IF(DSIGN(1.0D0,P27).NE.DSIGN(1.0D0,DSIN(P13*DEGRD))) P27 = P27 + 1
      IF(DSIGN(1.0D0,P27).NE.DSIGN(1.0D0,DSIN(P13*DEGRD)))
     *  P27 = P27 + 180.0D0
      IF(P27.LT. 0.0D0) P27 = P27 + 360D0
C
C     RADIUS VECTOR - R
C
      R = 1.0D0 - P26*DCOS(P13*DEGRD)
C
C     ABERRATION - P29
C
      P29 = -20.47/R/3600.0
C
```

```
C       MEAN OBLIQUITY - P43
C
        P11 = 23.452294D0
        P12 = -0.0130125D0
        P13 = -1.64D-6
        P14 = 5.03D-7
        P43 = P11 + P12*P23 + P13*P23**2 + P14*P23**3
C
C       MEAN ASCENSION - P45
C
        P11 = 279.6909832D0
        P12 = 0.98564734D0
        P13 = 3.8708D-4
        P45 = P11 + DMOD(P12*P22,360D0) + P13*P23**2
        P45 = DMOD(P45,360D0)
C
C-----------------------------------------------------------------------
C
CAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
C
C       NUTATION AND LONGITUDE PERT
C
        IF(P8.GT.1) GO TO 300
C-----------------------------------------------------------------------
C
C       MOON'S MEAN ANOMALY - P28
        P11 = 296.104608D0
        P12 = 1325D0*360D0
        P13 = 198.8491083D0
        P14 = .00919167D0
        P15 = 1.4388D-5
        P28 = P11 + DMOD(P12*P23,360D0) + P13*P23 + P14*P23**2 +
       *  P15*P23**3
        P28 = DMOD(P28,360D0)
C
C       MEAN ELONGATION OF MOON - P30:
C
        P11 = 350.737486
        P12 = 1236D0 * 360D0
        P13 = 307.1142167D0
        P14 = -1.436D-3
        P30 = P11 + DMOD(P12*P23,360D0) + P13*P23 + P14*P23**2
        P30 = DMOD(P30,360D0)
C
C       MOON LONG OF ASCENDING NODE - P31:
C
        P11 = 259.183275D0
        P12 = -5D0 * 360D0
        P13 = -134.142008D0
        P14 = 2.0778D-3
        P31 = P11 + DMOD(P12*P23,360D0) + P13*P23 + P14*P23**2
        P31 = DMOD(P31,360D0)
C
C       MEAN LONG OF MOON - P32:
C
        P11 = 270.434164D0
        P12 = 1336D0 * 360D0
        P13 = 307.8831417D0
        P14 = -1.1333D-3
        P32 = P11 + DMOD(P12*P23,360D0) + P13*P23 + P14*P23**2
        P32 = DMOD(P32,360D0)
C
```

```
C      MOON PERTURBATION OF SUN LONG - P74:
C
       P74 = 6.454D0*DSIN(P30*DEGRD) + .013D0*DSIN(3D0*P30*DEGRD) +
      *  .177D0*DSIN((P30+P28)*DEGRD) - .424D0*DSIN((P30-P28)*DEGRD)
        P74 = P74 + 0.039D0*DSIN((3D0*P30-P28)*DEGRD)
      * - 0.064D0*DSIN((P30+P25)*DEGRD) + 0.172D0*DSIN((P30-P25)*DEGRD)
       P74 = P74/3600D0
         P33 = P74
C
C      NUTATION OF LONG - P34
C
       P34 = -(17.2327D0-.017D0*P23) * DSIN(P31*DEGRD) +
      *  .2088D0*DSIN(2D0*P31*DEGRD) - .2037D0*DSIN(2D0*P32*DEGRD)
       P34 = P34 - 1.2729D0*DSIN(2D0*P24*DEGRD) + .1261D0*DSIN(P28*DEGRD)
       P34 = P34/3600D0
C
C      NUTATION IN OBLIQUITY - P34:
C
       P35 = 9.210D0*DCOS(P31*DEGRD) + .5522D0*DCOS(2D0*P24*DEGRD) -
      *  .09D0*DCOS(2D0*P31*DEGRD) + .0889D0*DCOS(2D0*P32*DEGRD)
       P35 = P35/3600D0
C
C      INEQUALITIES OF LONG PERIOD - P36:
C
       P36 = .266D0*DSIN((31.8D0+119D0*P23)*DEGRD) +
      *((1.882D0-.016D0*P23)*DEGRD) * DSIN((57.24D0+150.27D0*P23)*DEGRD)
       P36 = P36 + .202D0*DSIN((315.6D0+893.3D0*P23)*DEGRD) +
      * 1.089D0*P23**2 + 6.4D0*DSIN((231.19D0+20.2D0*P23)*DEGRD)
       P36 = P36/3600D0
C
C      MOON MEAN ARGUMENT OF LATITUDE - P63:
C
       P11 = 11.250889D0
       P12 = 1342D0*360D0
       P13 = 82.02515D0
       P14 = .003211D0
       P63 = P11 + DMOD(P12*P23,360D0) + P13*P23 + P14*P23**2
C
C--------------------------------------------------------------------
C
CBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
C
C      PERTURBATIONS DUE TO PLANETS - P33:
C
       IF(P8.GT.0) GO TO 300
C
C      VENUS:
C
C      MEAN ANOMALY OF VENUS - P37:
C
       P11 = 212.603222D0
       P12 = 162D0*360D0
       P13 = 197.803875D0
       P14 = 1.286D-3
       P37 = P11 + DMOD(P12*P23,360D0) + P13*P23 + P14*P23**2
       P37 = DMOD(P37,360D0)
       P11 = 4.838D0*DCOS((299.171D0+P37-P25)*DEGRD) +
      *  5.526D0*DCOS((148.5259D0+2D0*P37-2D0*P25)*DEGRD)
       P11 = P11 + 2.497D0*DCOS((316.5759D0+2D0*P37-3D0*P25)*DEGRD) +
      *   .666D0*DCOS((177.71D0+3D0*P37-3D0*P25)*DEGRD)
       P11 = P11 + 1.559D0*DCOS((345.4559D0+3D0*P37-4D0*P25)*DEGRD) +
      * 1.024D0*DCOS((318.15D0+3D0*P37-5D0*P25)*DEGRD)
```

```
      P70 = P11
      P33 = P33+P11/3600D0
C
C     PERT. OF LATITUDE OF SUN BY VENUS - P61:
C
      P61 = .21D0*DCOS((151.8D0+3D0*P37-4D0*P25)*DEGRD) +
     * .092D0*DCOS((93.7D0+P37-2D0*P25)*DEGRD) +
     * .067D0*DCOS((123D0+2D0*P37-3D0*P25)*DEGRD)
C
C     MARS:
C
C     MEAN ANOMALY OF MARS - P38:
C
      P11 = 319.529022D0
      P12 = 53D0*360D0
      P13 = 59.8592194D0
      P14 = 1.8083D-4
      P38 = P11 + DMOD(P12*P23,360D0) + P13*P23 + P14*P23**2
      P38 = DMOD(P38,360D0)
      P11 = .273D0*DCOS((217.7D0-P38+P25)*DEGRD) +
     *  2.043D0*DCOS((343.888D0-2D0*P38+2D0*P25)*DEGRD)
      P11 = P11 + 1.77D0*DCOS((200.4017D0-2D0*P38+P25)*DEGRD)+
     * .425D0*DCOS((338.88D0-3D0*P38+2D0*P25)*DEGRD)
      P11 = P11 + .5D0*DCOS((105.18D0-4D0*P38+3D0*P25)*DEGRD) +
     *  .585D0*DCOS((334.06D0-4D0*P38+2D0*P25)*DEGRD)
      P71=P11
      P33 = P33+P11/3600D0
C
C     JUPITER:
C
C     MEAN ANOMALY OF JUPITER - P39:
C
      P11 = 225.3225D0
      P12 = 8D0*360D0
      P13 = 154.583D0
      P39 = P11 + DMOD(P12*P23,360D0) + P13*P23
      P39 = DMOD(P39,360D0)
      P11 = 7.208D0*DCOS((179.5317D0-P39+P25)*DEGRD) +
     *  2.6D0*DCOS((263.2167D0-P39)*DEGRD)
      P11 = P11 + 2.731*DCOS((87.1450D0-2D0*P39+2D0*P25)*DEGRD)
      P11 = P11 + 1.61D0*DCOS((109.4933D0-2D0*P39+P25)*DEGRD) +
     *  .556D0*DCOS((82.65D0-3D0*P39+2D0*P25)*DEGRD)
      P72=P11
      P33 = P33+P11/3600D0
C
C     PERT OF SOLAR LATITUDE BY JUPITER - P62:
C
      P62 = .166D0*DCOS((265.5D0-2D0*P39+P25)*DEGRD)
C
C     SATURN:
C
C     MEAN ANOMALY OF SATURN - P40:
C
      P11 = 175.613D0
      P12 = 3D0*360D0
      P13 = 141.794D0
      P40 = P11 + DMOD(P12*P23,360D0) + P13*P23
      P40 = DMOD(P40,360D0)
      P11 = .419*DCOS((100.58D0-P40+P25)*DEGRD) +
     *  .32D0*DCOS((269.46D0-P40)*DEGRD)
      P73=P11
      P33 = P33+P11/3600D0
```

```
C
C-----------------------------------------------------------------------
C
CBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
C
CAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
C
C     PARAMETERS
C
  300 CONTINUE
C
C     PRECESSION:
C
      P42 = (50.2564D0+.0222D0*(P18-1900D0)/100D0) *
     *  (P23-(P18-1900D0)/100D0) * 100D0/3600D0
C
C     APPARENT LONGITUDE - P41:
C
      P41 = P27-P25+P24+P29+P33+P36+P34
C
C     SOLAR LATITUDE - P60:
C
      P60 = (.576D0*DSIN(P63*DEGRD) + P61+P62)/3600D0
C
C      OBLIQUITY - P75
C
      P75 = P35 + P43
C
C     APPARENT RIGHT ASCENSION - P44:
C
      P44  =  DATAN(DTAN(P41*DEGRD)*DCOS(P75*DEGRD))/DEGRD
      IF(DSIGN(ONE,P44).NE.DSIGN(ONE,DSIN(P41*DEGRD))) P44 = P44+180D0
      IF(P44.LT.0.D0) P44 = 360D0+P44
C
C     EQUATION OF TIME - P46:
C
      P46 = P45-P44
      IF(P46.GT.180D0) P46 = P46-360D0
C
C     HOUR ANGLE - P48:
C
      P48 = P51+P46-180D0
C
C     LOCAL HOUR ANGLE - P49
C
      P49  =  P48-P20
      P49 = DMOD(P49,360D0)
      IF(P49.LT.0) P49 = P49 + 360D0
C
C
C
C     DECLINATION - P47:
C
      P47 = DARSIN(DSIN(P41*DEGRD)*DSIN(P75*DEGRD)*DCOS(P60*DEGRD) +
     * DSIN(P60*DEGRD) * DCOS(P75*DEGRD))/DEGRD
C
C     ZENITH ANGLE - Z:
C
      Z = DARCOS(DSIN(P19*DEGRD)*DSIN(P47*DEGRD) + DCOS(P19*DEGRD) *
     *  DCOS(P47*DEGRD) * DCOS(P49*DEGRD))/DEGRD
C
C     AZIMUTH ANGLE - A:
```

```
C
      P11 = (-DSIN(P19*DEGRD) * DCOS(P49*DEGRD) * DCOS(P47*DEGRD) +
     *  DSIN(P47*DEGRD) *DCOS(P19*DEGRD))/DSIN(Z*DEGRD)
      IF(DABS(P11).GT.1.0D0) P11 = DSIGN(1.0D0,P11)
      A= DARCOS(P11)/DEGRD
      IF(DSIGN(1.D0,-DCOS(P47*DEGRD)  * DSIN(P49*DEGRD)/DSIN(Z*DEGRD))
     * .NE.DSIGN(1D0,DSIN(A*DEGRD)))  A = 360-A
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
      IF(P10.EQ.0) GO TO 30
      CALL NUPAGE
      PRINT 2000,MONTH,DAY,YEAR
2000  FORMAT(' MONTH',I5,/,' DAY',I5,/,' YEAR',I6)
      IF(P8.EQ.0) PRINT 2001
      IF(P8.EQ.1) PRINT 2002
      IF(P8.EQ.2) PRINT 2003
2001  FORMAT(' FULL COMPUTATIONS')
2002  FORMAT(' NO PLANETS')
2003  FORMAT(' NO MOON')
      PRINT 2004,LAT,LONG
 2004  FORMAT(' LATITUDE ', F10.4,/,' LONGITUDE',F10.4)
        IF(P6.NE.1.AND.P5.EQ.0) PRINT 2050,TIME
 2050  FORMAT(' TIME (LST)',F10.2)
        IF(P6.NE.1.AND.P5.EQ.1) PRINT 2051,TIME
 2051  FORMAT(' TIME (GMT)',F10.2)
C
        IF(P10.LT.2) GO TO 30
C
CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
C
      PRINT 2005,P23,P22
2005  FORMAT(' T',F15.6,' D',F15.6)
      PO = TOHMS(P24)
      PRINT 2006,PO
2006  FORMAT(' MEAN LONG',F15.5)
      PO = TOHMS(P25)
      PRINT 2007,PO
2007  FORMAT(' MEAN ANOMALY',F15.5)
      PRINT 2008,P26
2008  FORMAT(' ECCENTRICITY',F15.5)
      PO = TOHMS(P27-P25)
      PRINT 2009,PO
2009  FORMAT(' EQUATION OF CENTER',F15.5)
      PRINT 2010,R
2010  FORMAT(' RADIUS VECTOR',F15.5)
      PO = TOHMS(P29)
      PRINT 2011,PO
 2011 FORMAT (' ABERRATION ',F15.5)
      PO = TOHMS(P43)
      PRINT 2012,PO
2012  FORMAT(' MEAN OB',F15.5)
      PO =  TOHMS(P45/360D0*24D0)
      PRINT 2013,PO
2013  FORMAT(' MEAN ASCENSION',F15.5)
        IF(P10.GT.1) GO TO 35
      PO = TOHMS(P74)
      PRINT 2014, PO
 2014 FORMAT (' P MOON',F15.5)
        IF(P10.GT.0) GO TO 35
      PO = P70/3600D0
      PO = TOHMS(PO)
```

```
      PRINT 2015, PO
2015 FORMAT (' P VENUS',F15.5)
      PO = P61/3600D0
      PO = TOHMS(PO)
      PRINT 2016, PO
2016 FORMAT (' P LAT BY VENUS ',F15.5)
      PO = P71/3600D0
      PO = TOHMS(PO)
      PRINT 2017, PO
2017 FORMAT (' P MARS ',F15.5)
      PO = P72/3600D0
      PO = TOHMS(PO)
      PRINT 2018, PO
2018 FORMAT (' P JUPITER ',F15.5)
      PO = P62/3600D0
      PO = TOHMS(PO)
      PRINT 2019, PO
2019 FORMAT (' P OF LAT BY JUP ',F15.5)
      PO = P73/3600D0
      PO = TOHMS(PO)
      PRINT 2020, PO
2020 FORMAT (' P SATURN ',F15.5)
C
   35  CONTINUE
C
      PO = TOHMS(P33)
      PRINT 2021, PO
2021 FORMAT (' PERTURBATIONS ',F15.5)
      PO = TOHMS(P36)
      PRINT 2022, PO
2022 FORMAT (' LONG PERIOD ', F15.5)
      PO = TOHMS(P34)
      PRINT 2023, PO
2023 FORMAT (' NUTATION OF LONG ', F15.5)
      PO = TOHMS(P35)
      PRINT 2024, PO
2024 FORMAT (' NUT OBLIQUITY ', F15.5)
      PO = TOHMS(P42)
      PRINT 2025, PO
2025 FORMAT (' PRECESSION ', F15.5)
      PO = TOHMS(P41)
      PRINT 2026, PO
2026 FORMAT (' APPAR. LONG. ',F15.5)
      PO = TOHMS(P60)
      PRINT 2027, PO
2027 FORMAT (' SOLAR LATITUDE ',F15.5)
      PO = TOHMS(P75)
      PRINT 2028, PO
2028 FORMAT (' OBLIQUITY ',F15.5)
      PO = P44/360D0*24D0
      PO = TOHMS(PO)
      PRINT 2029, PO
2029 FORMAT (' APPAR. ASCENSION ',F15.5)
      PO = P46/360D0*24D0
      PO = TOHMS(PO)
      PRINT 2030, PO
2030 FORMAT (' EQUATION OF TIME ', F15.5)
      PO = TOHMS(P49)
      PRINT 2032, PO
2032 FORMAT (' LHA = ',F15.5)
      PO = TOHMS(P47)
      PRINT 2033, PO
```

```
 2033 FORMAT (' DECLIN.  ',F15.5)
C
CDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
   30 CONTINUE
C
CEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
C
C
      IF(P6.EQ.0) GO TO 500
      P12 = P53/360D0*24D0
      IF(P5.EQ.0) P12=0D0
      P10 = 0
      IF(AMOD(A,180.).GT.0.01) GO TO 400
      P11 = P56+P12
      IF(P11.LT.0D0) P11 = P11+24D0
      P13 = DMOD(P11,1D0)*60.0D0
      TIME = DINT(P11)*100.0D0+DINT(P13)
     * + DMOD(P13,1.D0)*60.0D0/100.D0
      IF(P56+P12.LT.0D0) TIME = -TIME
      RETURN
C
C
  400 CONTINUE
      IF(A.LT.90.0.OR.A.GT.270.0) GO TO 410
      IF(A.LE.180.) GO TO 405
  402  CONTINUE
      P55 = P56
      GO TO 1
  405  CONTINUE
        P54 = P56
        GO TO 1
  410 CONTINUE
      IF(A.LT.90.0) GO TO 405
        GO TO 402
C
C
  500 CONTINUE
C
CEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE
C
        IF(P10.EQ.0) GO TO 550
       P0 = A
      P0 = TOHMS(P0)
      PRINT 2034, P0
 2034 FORMAT (' A       DDD.MMSSS ',F15.5)
      P0 = Z
      P0 = TOHMS(P0)
      PRINT 2035, P0
 2035 FORMAT (' Z        DD.MMSSS ',F15.5)
C
  550  CONTINUE
      IF(P7.LT.3) RETURN
C
CFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
C
      IF(P10.EQ.0) GO TO 600
      P52 = P46
      P0 = P41-(P29+P34)
      P0 = TOHMS(P0)
```

```
      PRINT 2036, PO
2036  FORMAT (' LONGITUDE ',F15.6)
      PO = (P29+P34)*10000
      PO = TOHMS(PO)
      PRINT 2037, PO
2037  FORMAT (' REDN TO LONG ',F15.6)
      PO = P60*10000
      PO = TOHMS(PO)
      PRINT 2038, PO
2038  FORMAT (' LATITUDE ',F15.6)
      PO = P42*10000
      PO = TOHMS(PO)
      PRINT 2039, PO
2039  FORMAT (' PRECESSION ',F15.6)
      PO = P34*10000
      PO = TOHMS(PO)
      PRINT 2040, PO
2040  FORMAT (' NUT IN LONG ',F15.6)
      PO = P35*10000
      PO = TOHMS(PO)
      PRINT 2041, PO
2041  FORMAT (' NUT IN OBLI ',F15.6)
      PO = TOHMS(P43)
      PRINT 2042, PO
2042  FORMAT (' OBLIQUITY ',F15.6)
      PO = P44/360D0*24D0
      PO = TOHMS(PO)
      PRINT 2043, PO
2043  FORMAT (' APP RT.  ASCEN ',F15.6)
      PO = TOHMS(P47)
      PRINT 2044, PO
2044  FORMAT (' APP DECLINTION ',F15.6)
C
 600  CONTINUE
      P16 = P16+1.D0
      IF(P10.EQ.0) GO TO 610
      P10 = 0D0
      GO TO 1
 610  CONTINUE
      P11 = 180.D0-P52-(P46-P52)*((-P52+180.D0)/360D0)
      PO = TOHMS(P11/360.D0*24D0)
      PRINT 2045,PO
2045  FORMAT (' ET ',F15.6)
C
C
      RETURN
C
CFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
C
C                 *** END SUBROUTINE EPHEMS ***
      END
C
C
      SUBROUTINE NUPAGE
C    DUMMY NEW PAGE ROUTINE
       PRINT 2000
2000   FORMAT(1H1)
      RETURN
      END
C
C
      REAL*8 FUNCTION TOHMS(P1)
```

```
C     FUNCTION RETURNS P IN HH MM SS FORMAT
      IMPLICIT REAL*8(A-H,O-Z)
      DATA ONE/1.0D0/
      P3 = DABS(P1)
      B = DMOD(P3,ONE)*60.
      C = DINT(B)
      D = C/100.
      P2 = DINT(P3) + D + DMOD(B,ONE)*60./10000.
      P2 = P2*DSIGN(ONE,P1)
      TOHMS = P2
      RETURN

      END
```

### 4.1.1. Calling Sequence

The calling sequence for EPHEMS is

CALL EPHEMS(LAT,LONG,DAY,MONTH,YEAR,TIME,P5,P6,P7,P8,A,R,Z).

### 4.1.2. Input Arguments:

| | |
|---|---|
| LAT | REAL*4 - Latitude (in degrees and fractions of degrees, + north, - south) |
| LONG | REAL*4 - Longitude (in degrees and fractions of degrees + west, - east) |
| DAY | INTEGER*4 - Day of the month |
| MONTH | INTEGER*4 - Month of the year |
| YEAR | INTEGER*4 - Year |
| TIME | REAL*4 - Time (in form HHMM.SS 24 hour clock) Local standard if P5 = 0, Greenwich mean, if P5 = 1. Note: Local standard is defined by the meridians 15° apart and not by the convention used on the North American continent. |
| P5 | INTEGER*4 - Time type (0 = Local standard, 1 = Greenwich mean) |
| P6 | INTEGER*4 - Apparent noon calculations (0 - no, 1 - yes) |
| P7 | INTEGER*4 - Print options (0 - nothing printed, 1 - date, location and sun position, 2 - full printout of intermediate results, 3 - Nautical Almanac comparison printout) |
| P8 | INTEGER*4 - Approximation used (0 - full computation, 1 - no planetary effects, 2 - no lunar effects) |

### 4.1.3. Returned Variables:

| | |
|---|---|
| A | REAL*4 - Azimuth angle of sun measured clockwise from the north (in degrees) |
| Z | REAL*4 - Zenith angle of sun (in degrees) |
| R | REAL*4 - Radius vector of sun (R = 1 at mean sun - earth distance) |

In the case P6 = 1, where apparent noon is being computed, the value for local apparent noon time is returned in TIME in units specified by P5 (either local standard or Greenwich mean). The form is HHMM.SS.

### 4.1.4. Comments:

The listing, with the help of Section 3 and the appendix, should be self-explanatory. The following points should, however, be considered:

1) The approximations involving the planetary and lunar terms and controlled by variable P8 have been coded so that their removal is easily done. The primary reasons for their permanent deletion are the reduction of program length and the increase in computational speed. However, this deletion does result in a corresponding decrease in accuracy. The sections to remove are outlined in the listing and are as follows:

a) no planetary effects - delete code between the lines with BBBBB.....'s

b) no lunar effects - delete code between AAAAA.....'s

2) If the routine is to be used in a manner where the internal print statements are not needed, then the following code may be deleted:

a) Nautical Almanac comparison - Code between FFFF.....'s. This option, which would rarely be used, presents the output in a form which allows direct comparison with the Nautical Almanac's solar ephemeris tabulations. This output option was included primarily for debugging purposes.

b) Internal intermediate results. This output is useful for debugging and for values which may be of use for certain other applications. Delete code between DDDD.....'s.

c) if no internal printing is wanted, delete code between CCCC.....'s.

3) The routine will compute the local apparent noon. This is defined as that local time when the sun is in the local meridian. (*i.e.* azimuth = 180° or 0°). It is assumed that noon will occur

between 1000 and 1400 local standard time. This option is controlled by variable P6. If it is not desired delete code between EEEE.....'s.

4) In Section 2.8.2, the difference between Universal Time, *UT*, and Ephemeris Time, *ET*, was noted. The routine ignores this difference as accuracy is not significantly affected. If it is necessary to include this difference, the variable P57 may be set to the correct value. This is located below the line with +++++...'s.

5) The programming has been done so that addition or deletion of terms of the various perturbations is easily accomplished. These changes are left to the discretion of the user.

## 4.2. HPL VERSION - EPHEMS

The HPL version of the routine is presented in Listing 2. HPL is a language Hewlett-Packard uses on some of its desk calculators. In this particular instance an HP9825 was used. The minimum requirements are 16K bytes memory and the advanced programming ROM (The "*p*" variable option is required).

**LISTING 2.**

```
0: "ephems":
1:
2: "subroutine to compute solar position":
3:
4: "written by Wayne H. Wilson,Jr.":
5: "Visibility Laboratory":
6: "Scripps Institution of Oceanography":
7: "University of California, San Diego":
8: "La Jolla, California, 92093":
9: "Ph. (714)-294-5534":
10: " 26 Feb 1980":
11:
12: "p1 - latitude (dd.mm, +north,-south)":
13: "p2 - longitude (ddd.mm, +west,-east)":
14: "p3 - date (DD.MMYYYY)":
15: "p4 - time (hhmm.ss) local standard(LST) or GMT":
16: "p5 - time type (0-LST,1-GMT)":
17: "p6 - apparent noon calculations (1-yes,0-no)":
18: "p7 - print (0-none,1-date,2-data,3-NA compar)":
19: "p8 - flag for approximations (0-all,1-no planets,2-no moon)":
20:
21: "p10 - whether to print date and loc info":
22: "p11 through p15 are scratch variables":
23:
24: "Variables passed back to calling program":
25:
26: "A - Azimuth angle":
27: "R - Radius vector":
28: "Z - Zenith angle":
29: "p4 - Apparent noon if p6=1 (LST or GMT see p5)":
30:
31: "p16 - day":
```

```
"p17 - month":
"p18 - year":
"p19 - latitude":
"p20 - longitude":
"p21 - time LST if p5=0,GMT if p5=1":
"p22 - day count from 0JAN1900":
"p23 - T - fraction of century":

"p51 - GMT in degrees":
"p53 - zone longitude":
"p54,p55,p56 - variables used in apparent noon cal":

"p57 - (ET-UT),difference of Ephemeris and Universal time":
"++++++++++++++++++++++++++++++":
0→p57

10→p54;14→p55;p7→p10;int(p3)→p16

"EPHSTRT":

int(frc(p3)*100)→p17;frc(frc(p3)*100)*10000→p18
abs(p2)→p20;abs(p1)→p19
frc(p19)*100/60+int(p19)→p19;p19*sgn(p1)→p19
frc(p20)*100/60+int(p20)→p20;sgn(p2)*p20→p20
15*int((7.5+abs(p20))/15)*sgn(p20)→p53;p53→p11;if p5=1;0→p11
if p6#1;jmp 3
(p54+p55)/2→p56;p56/24*360+p53→p51
jmp 3
(frc(p4)*100/3600+frc(int(p4)/100)*100/60+int(int(p4)/100))/24→p21
p21*360+p11→p51
fxd 6
p18→p12
p17+1→p11
if p11>3;jmp 2
p11+12→p11;p12-1→p12
int(p12*365.25)+int(p11*30.6)+p16→p13
p13-694038.5→p22
p18+p17/100+p16/10000→p11
if p11<1900.0228;p22+1→p22
if p11<1800.0228;p22+1→p22
(p51/360+p22+p57)/36525→p23
p23*36525→p22

"Mean longitude - p24":

279.69668→p11
.9856473354→p12
3.03e-4→p13
p11+p12*p22mod360+p13*p23^2→p24;p24mod360→p24

"mean anomaly - p25":

358.47583→p11
.985600267→p12
-1.5e-4→p13
-3e-6→p14
p11+p12*p22mod360+p13*p23^2+p14*p23^3→p25;p25mod360→p25

"eccentricity - p26":

.01675104→p11
-4.18e-5→p12
```

```
-1.26e-7→p13
p11+p12*p23+p13*p23^2→p26
p25*π/180→p11→p12

"eccentric anomaly - p13 (temp)":

rad
p12→p13;p11+p26*sin(p13)→p12
if abs((p12-p13)/p12)>1e-8;jmp -1
p12*180/π→p13
deg

"true anomaly - p27":

2atn(√((1+p26)/(1-p26))*tan(p13/2))→p27
if sgn(p27)#sgn(sin(p13));p27+180→p27
if p27<0;p27+360→p27

"Radius vector - R":

1-p26*cos(p13)→R

"Aberration - p29":

-20.47/R/3600→p29

"mean obliquity - p43":

23.452294→p11
-.0130125→p12
-1.64e-6→p13
5.03e-7→p14
p11+p12*p23+p13*p23^2+p14*p23^3→p43

"Mean ascension - p45":

279.6909832→p11
.98564734→p12
3.8708e-4→p13
p11+p12*p22mod360+p13*p23^2→p45;p45mod360→p45

"AAAAAAAAAAAAAAAAAA":

"*********************************************************************":
"Nutation":

if p8>1;gto "Parameters"

"Moon's mean anomaly - p28":

296.104608→p11
1325*360→p12
198.8491083→p13
.00919167→p14
1.4388e-5→p15
p11+p12*p23mod360+p13*p23+p14*p23^2+p15*p23^3→p28;p28mod360→p28

"Mean elongation of moon - p30":

350.737486→p11
1236*360→p12
307.1142167→p13
```

```
1.436e-3→p14
p11+p12*p23mod360+p13*p23+p14*p23^2→p30;p30mod360→p30
"Moon Long of ascending node - p31":
259.183275→p11
-5*360→p12
-134.142008→p13
2.0778e-3→p14
p11+p12*p23mod360+p13*p23+p14*p23^2→p31;p31mod360→p31
"Mean long of moon - p32":
270.434164→p11
1336*360→p12
307.8831417→p13
-1.1333e-3→p14
p11+p12*p23mod360+p13*p23+p14*p23^2→p32;p32mod360→p32
"Moon perturbation of sun long - p33":
6.454sin(p30)+.013sin(3p30)+.177sin(p30+p28)-.424sin(p30-p28)→p33
p33+.039sin(3p30-p28)-.064sin(p30+p25)+.172sin(p30-p25)→p33;p33/3600→p74
p74→p33
"Nutation of long - p34":
-(17.234-.017p23)sin(p31)+.209sin(2p31)-.204sin(2p32)→p34
p34-1.257sin(2p24)+.127sin(p28)→p34;p34/3600→p34
"Nutation in obliquity - p35":
9.214cos(p31)+.546cos(2p24)-.09cos(2p31)+.088cos(2p32)→p35
p35/3600→p35
"Inequalities of long period - p36":
.266sin(31.8+119p23)+(1.882-.016p23)sin(57.24+150.27p23)→p36
p36+.202sin(315.6+893.3p23)+1.089p23^2+6.4sin(231.19+20.2p23)→p36
p36/3600→p36
"Moon mean argument of latitude - p63":
11.250889→p11
1342*360→p12
82.02515→p13
.003211→p14
p11+p12*p23mod360+p13*p23+p14*p23^2→p63
"BBBBBBBBBBBBBBBBBB":
"******************************************************************":
"Perturbations due to planets - p33":
if p8>0;gto "Parameters"
"Venus":
"Mean anomaly of Venus - p37":
212.603222→p11
```

```
218: 162*360→p12
219: 197.803875→p13
220: 1.286e-3→p14
221: p11+p12*p23mod360+p13*p23+p14*p23^2→p37;p37mod360→p37
222: 4.838cos(299.171+p37-p25)+5.526cos(148.5259+2p37-2p25)→p11
223: p11+2.497cos(316.5759+2p37-3p25)+.666cos(177.71+3p37-3p25)→p11
224: p11+1.559cos(345.4559+3p37-4p25)+1.024cos(318.15+3p37-5p25)→p70
225: p33+p70/3600→p33
226:
227: "Pert. of latitude of sun by Venus - p61":
228:
229: .21cos(151.8+3p37-4p25)+.092cos(93.7+p37-2p25)+.067cos(123+2p37-3p25)→p61
230:
231: "Mars":
232:
233: "Mean anomaly of Mars - p38":
234:
235: 319.529022→p11
236: 53*360→p12
237: 59.8592194→p13
238: 1.8083e-4→p14
239: p11+p12*p23mod360+p13*p23+p14*p23^2→p38;p38mod360→p38
240: .273cos(217.7-p38+p25)+2.043cos(344.4898-2p38+2p25)→p11
241: p11+1.77cos(200.6713-2p38+p25)+.425cos(338.88-3p38+2p25)→p11
242: p11+.5cos(105.18-4p38+3p25)+.585cos(334.06-4p38+2p25)→p71
243: p33+p71/3600→p33
244:
245: "Jupiter":
246:
247: "Mean anomaly of Jupiter - p39":
248:
249: 225.3225→p11
250: 8*360→p12
251: 154.583→p13
252: p11+p12*p23mod360+p13*p23→p39;p39mod360→p39
253: 7.208cos(179.888-p39+p25)+2.6cos(263.3685-p39)→p11
254: p11+2.731cos(87.2472-2p39+2p25)→p11
255: p11+1.61cos(109.8259-2p39+p25)+.556cos(82.65-3p39+2p25)→p72
256: p33+p72/3600→p33
257:
258: "Pert of solar latitude by Jupiter - p62":
259:
260: .166cos(265.5-2p39+p25)→p62
261:
262: "Saturn":
263:
264: "Mean anomaly of Saturn - p40":
265:
266: 175.613→p11
267: 3*360→p12
268: 141.794→p13
269: p11+p12*p23mod360+p13*p23→p40;p40mod360→p40
270: .419cos(100.58-p40+p25)+.32cos(269.46-p40)→p73
271: p33+p73/3600→p33
272:
273: "BBBBBBBBBBBBBBBBBBB":
274:
275: "AAAAAAAAAAAAAAAAA":
276:
277: "*****************************************************************":
278:
279: "Parameters":
```

```
"Precession":
(50.2564+.0222(p18-1900)/100)(p23-(p18-1900)/100)*100/3600→p42
"Apparent longitude - p41":
p27-p25+p24+p29+p33+p36+p34→p41
"Solar latitude - p60":
(.576*sin(p63)+p61+p62)/3600→p60
"Obliquity - p75":
p35+p43→p75
"Apparent right ascension - p44":
atn(tan(p41)cos(p75))→p44
if sgn(p44)#sgn(sin(p41));p44+180→p44
if p44<0;360+p44→p44
"equation of time - p46":
p45-p44→p46
if p46>180;p46-360→p46
"hour angle - p48":
p51+p46-180→p48
"local hour angle - p49":
p48-p20→p49;p49mod360→p49
"declination - p47":
asn(sin(p41)sin(p75)cos(p60)+sin(p60)*cos(p75))→p47
"Zenith angle - Z":
acs(sin(p19)sin(p47)+cos(p19)cos(p47)cos(p49))→Z
"Azimuth angle - A :
(-sin(p19)cos(p49)cos(p47)+sin(p47)cos(p19))/sin(Z)→A
if abs(A)>1;sgn(A)→A
acs(A)→A
if sgn(-cos(p47)sin(p49)/sin(Z))#sgn(sin(A));360-A→A
"CCCCCCCCCCCCCCCCCC":
if p10=0;gto  noprnt"
fxd 0
spc ;prt "Month",p17,"Day",p16,"Year",p18
fxd 4
if p8=0;prt "Full Computation"
if p8=1;prt "No Planets"
if p8=2;prt "No Moon"
```

```
342: prt "Lat",p1,"Long",p2
343: fxd 2
344: if p6#1;if p5=0;prt "Time LST",p4
345: if p6#1;if p5=1;prt "Time GMT",p4
346: fxd 6
347:
348: "DDDDDDDDDDDDDDDDD":
349:
350: if p10<2;gto "noprnt"
351:
352: prt "T",p23;fxd 1;prt "d",p22
353: prt "Mean Long",'tohms'(p24)
354: prt "Mean Anomaly",'tohms'(p25)
355: prt "Eccen.",p26
356: prt "Eq. of Center",'tohms'(p27-p25)
357: prt "Radius Vector",R
358: prt "Aberration",'tohms'(p29)
359: prt "Mean Ob",'tohms'(p43)
360: prt "Mean Ascension",'tohms'(p45/360*24)
361: if p8>1;gto "noprntl"
362: prt "P Moon",'tohms'(p74)
363: if p8>0;gto "noprntl"
364: prt "P Venus",'tohms'(p70/3600)
365: prt "P Lat by Venus",'tohms'(p61/3600)
366: prt "P Mars",'tohms'(p71/3600)
367: prt "P Jupiter",'tohms'(p72/3600)
368: prt "P of Lat by Jup",'tohms'(p62/3600)
369: prt "P Saturn",'tohms'(p73/3600)
370: "noprntl":
371: prt "Perturbations",'tohms'(p33)
372: prt "Long Period",'tohms'(p36)
373: prt "Nutation of Long",'tohms'(p34)
374: prt "Nut Obliquity",'tohms'(p35)
375: prt "Precession",'tohms'(p42)
376: prt "Appar. Long",'tohms'(p41)
377: prt "Solar Latitude",'tohms'(p60)
378: prt "Obliquity",'tohms'(p75)
379: prt "Appar. ascension",'tohms'(p44/360*24)
380: prt "Equation of Time",'tohms'(p46/360*24)
381: prt "LHA =",'tohms'(p49)
382: prt "Declin.",'tohms'(p47)
383:
384: "DDDDDDDDDDDDDDDDDD":
385:
386: "CCCCCCCCCCCCCCCCCC":
387: "noprnt":
388:
389: "EEEEEEEEEEEEEEEEEEEEEEEEEEEEE":
390:
391: if p6=0;gto "EPH0"
392: fxd 2
393: p53/360*24→p12;if p5=0;0→p12
394: p56+p12→p11;if p11<0;p11+24→p11
395: int(p11)*100+int(frc(p11)*60)+frc(frc(p11)*60)*60/100→p13
396  if p56+p12<0;-p13→p13
397: dsp "relax - working on noon",p13
398: 0→p10;fxd 4;if Amod180>.01;jmp 3
399: p13→p4
400: ret
401: if A<90 or A>270;gto "EPHNOR"
402: if A>180;p56→p55;gto "EPHSTRT"
403: p56→p54;gto "EPHSTRT"
```

```
"EPHNOR":if A<90;p56→p54;gto "EPHSTRT"
p56→p55;gto "EPHSTRT"

"EEEEEEEEEEEEEEEEEEEEEEEEEEEEEE":

"EPHO":
fxd 3
if p10>0;prt "A       ddd.mmsss",'tohms'(A)
if p10>0;prt "Z        dd.mmsss",'tohms'(Z)

if p7<3;ret

"FFFFFFFFFFFFFFFFFFF":

if p10=0;gto "EPHNEX"

"p52 - Equation of time temporary save":

p46→p52
prt "Longitude ",'tohms'(p41-(p29+p34))
prt "Redn to long",'tohms'(p29+p34)*10000
prt "Latitude",'tohms'(p60)*10000
fxd 2
prt "Precession",'tohms'(p42)*10000
prt "Nut in long",'tohms'(p34)*10000
prt "Nut in Obli",'tohms'(p35)*10000
fxd 6
prt "Obliquity",'tohms'(p43)
prt "App Rt. Ascen",'tohms'(p44/360*24)
prt "App Declintion",'tohms'(p47)
"EPHNEX":p16+1→p16;if p10=0;jmp 2
0→p10;gto "EPHSTRT"
180-p52-(p46-p52)*((-p52+180)/360)→p11
prt "ET",'tohms'(p11/360*24)
ret
"FFFFFFFFFFFFFFFFFFFF":

"tohms":

abs(p1)→p3
int(p3)+int(frc(p3)*60)/100+frc(frc(p3)*60)*60/10000→p2
fxd 5
p2*sgn(p1)→p2
ret p2
end
```

### 4.2.1. Calling Sequence:

The calling sequence for EPHEMS is

cll 'EPHEMS' (X,W,D,T,r5,r6,r7,r8).

The calling arguments are those used in the main program listed in Section 5.

### 4.2.2. Input Arguments:

| | |
|---|---|
| X | Latitude (in form DD.MM where DD = degrees, MM = minutes, + north, - south) |
| W | Longitude (in form DDD.MM where DD = degrees, MM = minutes, + west, - east) |
| D | Date (DD.MMYYYY, where DD = day, MM = month (i.e. 01, 02 --, 12) YYYY = year) |
| T | Time (HHMM.SS - 24 hour clock Local standard if r5 = 0, Greenwich mean if r5 = 1) |
| r5 | Time type (0 - Local standard, 1 - Greenwich mean) |
| r6 | Apparent noon calculations (0 - no, 1 - yes) |
| r7 | Print options (0 - nothing, 1 - date, location and sun position, 2 - intermediate results, 3 - Nautical Almanac comparison) |
| r8 | Approximation used (0 - full computation, 1 - no planetary effects, 2 - no lunar effects) |

### 4.2.3. Returned Values

The returned values are returned in the global variables A, R and Z. These are:

| | |
|---|---|
| A | Azimuth angle of sun measured clockwise for the north (in degrees) |
| Z | Zenith angle of sun (in degrees) |
| R | Radius vector of sun (R = 1 for mean sun earth distance) |

If apparent noon is being computed (*i.e.* r6 = 1) then T will also be returned with the apparent noon time, either Local standard if r5 = 0, or Greenwich mean if r5 = 1. Form is HHMMSS.

### 4.2.4. Comments

The comments made for the FORTRAN version are equally applicable to the HPL version. In addition the following should be noted:

1) There are differences in the format of the passed arguments for latitude, longitude and date between the FORTRAN and HPL programs. These differences are due to the requirements for interface to other programs at the Laboratory. The elimination of these differences between the two versions is a relatively minor programming change and as such is left to the user.

2) A, Z, and R are the only global variables used in the subroutine. All other variables are "*p*" variables.

## 4.3. TOHMS and NUPAGE

The subroutine TOHMS is used in both the FORTRAN and HPL versions to convert a number which is in degrees into the form DDD.MMSS (DDD = degrees, MM = minutes, SS = seconds), the standard form used in Almanacs. To convert to another standard form HH.MMSS, a number should be multiplied by the factor 24/360 before calling TOHMS.

The routine NUPAGE is a program which ejects the line printer to the top of a page and prints the date and time in the upper right hand corner. A dummy version of NUPAGE is included in Listing 1 which simply causes a page eject.

# 5. SOLAR POSITION PROGRAMS

In this section several programs which use the ephemeris subroutines outlined in Section 4 are listed, described, and example outputs are presented. There is one program in FORTRAN and three in HPL.

## 5.1. FORTRAN PROGRAM

The FORTRAN program uses EPHEMS to compute solar position for a single location and time or for an extended time at a location. The listing of the program EPHEM is presented in Listing 3. This program has been run on an IBM 360/44 and a PRIME 500 with the only change being the reassignment of the I/O FORTRAN unit numbers (NKBIN and NTYWT). The program has been set up for an interactive user, so that NKBIN and NTYWT refer to a terminal. The program should work however if NTYWT is defined to be a line printer and NKBIN to a card reader.

The use of the program should be self-explanatory. Basically it askes the following questions:

1) What type of time will be entered, Local standard or Greenwich mean?
2) Print option
3) Approximation to use
4) Date
5) Latitude
6) Longitude
7) Is apparent noon wanted? If it is, the program computes apparent noon, prints the result and returns for more input.
8) Solar position for a given time is wanted. Does the user want a single time or an extended time?
9) Is the solar position wanted for the same location but for a different time?

In Fig. 8, an example of apparent noon calculations is presented. In Fig. 9, a typical response for a single time is listed with the extended print option specified. Figure 10 illustrates the output for an extended time response.

**LISTING 3.**

```
C      *** PROGRAM EPHEM
C
C        WRITTEN BY WAYNE H WILSON, JR.
C        VISIBILTY LABORATORY
C        SCRIPPS INSTITUTION OF OCEANOGRAPHY
C        UNIVERSITY OF CALIFORNIA, SAN DIEGO
C        LA JOLLA, CALIFORNIA, 92093
C        PH  (714)-294-5534
C        28 FEB 1980
C
C
C
C      LONG - LONGITUDE (IN DEGREES + WEST, - EAST)
C      TIME - TIME
C      LAT - LATITUDE (IN DEGREES + NORTH, - SOUTH)
C      R5 - TIME TYPE (0-LST, 1-GMT)
C      R6 - APPARENT NOON CALCULATIONS
C      R7 - PRINT (0-NONE, 1-DATE, 2-DATA, 3-NA COMPARISON)
C      R8 - APPROXIMATION (0-ALL, 1-NO PLANETS, 2-NO MOON)
C
C              *** SPECIFICATION STATEMENTS ***
       REAL*8 TOHMS, DZ, DA, DATE, DX
       INTEGER*4 NLPRT, NTYWT
       INTEGER*4 DAY, MONTH, YEAR
       REAL*4 LAT, LONG, TIME
       INTEGER*4 R5, R6, R7, R8, TIMH, TIMM, TIMHS, TIMMS, TIMINC, TIBEG, TIEND
C
C
C
C
C-----------------------------------------------------------------------------
C-----------------------------------------------------------------------------
C
C
     1 CONTINUE
         NKBIN = 1
         NTYWT = 1
C      *** BEGIN:
     5 CONTINUE
  1000  FORMAT(G14.4)
C
       WRITE (NTYWT,1001)
  1001 FORMAT (' TIME TYPE (0-LST,1-GMT)')
         R5 = 0
       READ(NKBIN,1000,END=999) R5
C
         WRITE(NTYWT,1002)
         R7 = 0
        READ(NKBIN,1000) R7
  1002  FORMAT('DATE=0,DATA=1')
         R7 = R7 + 1
C
         WRITE(NTYWT,1003)
  1003  FORMAT('APPROX. NO PLANETS=1, NO MOON=2')
        R8 = 0
         READ(NKBIN,1000) R8
C
  1004  FORMAT('DATE (DD.MMYYYY)')
  1005  FORMAT(' LATITUDE - DD.MM  + NORTH , - SOUTH')
  1006  FORMAT(' LONGITUDE - DDD.MM  + WEST , - EAST')
```

```
C
      WRITE(NTYWT,1004)
      READ(NKBIN,*) DATE
      WRITE (NTYWT,1005)
      READ (NKBIN,*) LAT
      WRITE(NTYWT,1006)
      READ(NKBIN,*) LONG
C
        DAY = DINT(DATE)
        DX = DMOD(DATE,1.D0)*100.
        MONTH = DINT(DX)
        YEAR = DMOD(DX,1.0D0)*10000.  + 0.2
      C = ABS(LAT)
        XLAT = LAT
      LAT = (AINT(C) + AMOD(C,1.0)*100.0/60.0)*SIGN(1.0,LAT)
      C = ABS(LONG)
        XLONG = LONG
      LONG = (AINT(C) + AMOD(C,1.0)*100.0/60.0)*SIGN(1.0,LONG)
C
      WRITE (NTYWT,1007)
 1007 FORMAT (' APPARENT NOON? YES(1),NO(0)')
      READ (NKBIN,1000) R6
      IF(R6.EQ.0) GO TO 30
C
      CALL EPHEMS (LAT,LONG,DAY,MONTH,YEAR,TIME,R5,R6,R7,R8,A,R,Z)
C
      IF(R5.EQ.0) WRITE (NTYWT,2000) TIME
 2000 FORMAT (' APPARENT NOON (LST) = ',F10.2)
      IF(R5.EQ.1) WRITE(NTYWT,2001)
 2001 FORMAT(' APPARENT NOON (GMT) = ',F10.2)
C
      DZ = Z
      TEMP = TOHMS(DZ)
      WRITE (NTYWT,2002) TEMP
 2002 FORMAT (' Z        DD.MM',F10.2)
      GO TO 5
C
   30 CONTINUE
C
        WRITE(NTYWT,1008)
 1008   FORMAT('SINGLE TIME(0),TABLE(1)')
        R4 = 0
        READ(NKBIN,1000) R4
        IF(R4.EQ.0) GO TO 50
C
C
        WRITE(NTYWT,1009)
 1009   FORMAT('BEGTIME(HHMM),ENDTIME(HHMM),TIME INC(HMM)')
        READ(NKBIN,*) TIBEG,TIEND,TIMINC
      TIMH=TIBEG/100
      TIMM=TIBEG-TIMH*100
      TIMHS=TIEND/100
      TIMMS=TIEND-TIMHS*100
      TIMEEN=FLOAT(TIMHS)+FLOAT(TIMMS)/60.0
        TIMHS = TIMINC/100
        TIMMS = TIMINC - TIMHS*100
        TIMINC = TIMHS*60 + TIMMS
C
       XL1 = INT((7.5+ABS(LONG))/15)*15*SIGN(1.0,LONG)
C
        R6 = 1
        R7 = 0
```

```
      CALL EPHEMS(LAT, LONG, DAY, MONTH, YEAR, TI, R5, R6, R7, R8, A, R, Z)
      R6 = 0
C
      WRITE(NTYWT, 2003) DAY, MONTH, YEAR, XLAT, XLONG, XL1
      WRITE(NTYWT, 2004) TI
C
 2003 FORMAT(1H1, 10X, 'SOLAR POSITION TABLE', //
     * , 10X, 'DATE(D/M/Y)', 2X, I2, '/', I2, '/'
     * , I4, //, 5X, 'LATITUDE', F10. 3, 5X, 'LONGITUDE', F10. 3,
     * //, 10X, 'STANDARD MERIDIAN', F10. 3 )
 2004 FORMAT(/, 7X, 'APPARENT NOON (HHMM. SS)', 2X, F7. 2, //
     * , 15X, 'ANGLES ARE DDD. MM', //, 5X, 'TIME', 5X,
     * 'ZENITH ANGLE', 5X, 'AZIMUTH ANGLE', /)
      GO TO 41
   40   CONTINUE
      TIMM=TIMM+TIMINC
   41   CONTINUE
      IF(TIMM. LT. 60) GO TO 45
      TIMM=TIMM-60
        TIMH = TIMH + 1
      IF(TIMH. LT. 25) GO TO 41
      TIMH=TIMH-24
      DAY=DAY+1
   45   CONTINUE
      TIMEH=TIMH
      TIMEM=TIMM
C
      TIME=TIMEH+TIMEM/60. 0
      IF(TIME. GT. 1. 01*TIMEEN) GO TO 1
      TI=TIMH*100+TIMM
C
        CALL EPHEMS(LAT, LONG, DAY, MONTH, YEAR, TI, R5, R6, R7, R8, A, R, Z)
C
C
        DZ = TOHMS(DBLE(Z))
        DA = TOHMS(DBLE(A))
        WRITE(NTYWT, 2005) TI, DZ, DA
 2005 FORMAT(3X, F7. 2, 6X, F5. 2, 13X, F6. 2)
        GO TO 40
C
C

   50 CONTINUE
C
      IF(R5. EQ. 0) WRITE(NTYWT, 1010)
 1010 FORMAT (' TIME - HHMM. SS LOCAL STANDARD')
      IF(R5. EQ. 1) WRITE (NTYWT, 1011)
 1011 FORMAT (' TIME - HHMM. SS GMT')
      READ (NKBIN, *) TIME
C
      CALL EPHEMS (LAT, LONG, DAY, MONTH, YEAR, TIME, R5, R6, R7, R8, A, R, Z)
C
      WRITE (NTYWT, 2006)
 2006 FORMAT (' REPEAT W/DIFFERENT TIME,  1-YES')
      READ (NKBIN, 1000) I
       IF(I. EQ. 1) GO TO 50
      GO TO 1
C
  999 CONTINUE
      CALL EXIT
C
C             *** END PROGRAM EPHEM ***
      END
```

```
 1) TIME TYPE (O-LST, 1-GMT)
 2)0
 3)DATE=0, DATA=1
 4)0
 5)APPROX.  NO PLANETS=1,  NO MOON=2
 6)0
 7)DATE (DD. MMYYYY)
 8)6. 031980
 9) LATITUDE - DD. MM   + NORTH ,  - SOUTH
10)32.
11) LONGITUDE - DDD. MM   + WEST ,  - EAST
12)120.
13) APPARENT NOON? YES(1), NO(0)
14)1
15)1
16) MONTH    3
17) DAY    6
18) YEAR  1980
19) FULL COMPUTATIONS
20) LATITUDE     32. 0000
21) LONGITUDE   120. 0000
22) APPARENT NOON (LST) =      1211. 10
23) Z          DD. MM       37. 21
```

Fig. 8

```
 1) TIME TYPE (O-LST, 1-GMT)
 2)0
 3)DATE=0, DATA=1
 4)1
 5)APPROX.  NO PLANETS=1,  NO MOON=2
 6)0
 7)DATE (DD. MMYYYY)
 8)6. 031980
 9) LATITUDE - DD. MM   + NORTH ,  - SOUTH
10)32.
11) LONGITUDE - DDD. MM   + WEST ,  - EAST
12)120.
13) APPARENT NOON? YES(1), NO(0)
14)0
15)SINGLE TIME(0), TABLE(1)
16)0
17) TIME - HHMM. SS LOCAL STANDARD
18)1200.
19)1
20) MONTH    3
21) DAY    6
22) YEAR  1980
23) FULL COMPUTATIONS
24) LATITUDE     32. 0000
25) LONGITUDE   120. 0000
26) TIME (LST)     1200. 00
27) T          0. 801789 D     27285. 333333
28) MEAN LONG          344. 42275
29) MEAN ANOMALY            62. 06291
30) ECCENTRICITY             0. 01672
31) EQUATION OF CENTER                1. 42345
32) RADIUS VECTOR               0. 99240
33) ABERRATION             -0. 00206
34) MEAN OB            23. 26307
35) MEAN ASCENSION              22. 58485
36) PERTURBATIONS               -0. 00066
37) LONG PERIOD            -0. 00051
38) NUTATION OF LONG                -0. 00089
39) NUT OBLIQUITY               -0. 00074
40) PRECESSION               0. 00090
41) APPAR. LONG.             346. 24208
42) SOLAR LATITUDE                0. 00005
43) OBLIQUITY              23. 26233
44) APPAR. ASCENSION              23. 09578
45) EQUATION OF TIME               -0. 11093
46) LHA =            357. 12409
47) DECLIN.               -5. 21529
48) A           DDD. MMSSS           175. 25549
49) Z            DD. MMSSS           37. 27324
50) REPEAT W/DIFFERENT TIME,  1-YES
51)0
```

Fig. 9

```
 1) TIME TYPE (O-LST, 1-GMT)
 2)0
 3)DATE=0, DATA=1
 4)0
 5)APPROX.  NO PLANETS=1,  NO MOON=2
 6)0
 7)DATE (DD. MMYYYY)
 8)6. 031980
 9) LATITUDE - DD. MM   + NORTH ,  - SOUTH
10)32.
11) LONGITUDE - DDD. MM   + WEST ,  - EAST
12)120.
13) APPARENT NOON? YES(1), NO(O)
14)0
15)SINGLE TIME(O), TABLE(1)
16)1.
17)BEGTIME(HHMM), ENDTIME(HHMM), TIME INC(HMM)
18)0600 1800 30
19)1             SOLAR POSITION TABLE
20)
21)            DATE(D/M/Y)     6/ 3/1980
22)
23)      LATITUDE     32. 000      LONGITUDE    120. 000
24)
25)            STANDARD MERIDIAN    120. 000
26)
27)          APPARENT NOON (HHMM. SS)   1211. 10
28)
29)                  ANGLES ARE DDD. MM
30)
31)       TIME       ZENITH ANGLE       AZIMUTH ANGLE
32)      600. 00        95. 15                 93. 09
33)      630. 00        88. 55                 97. 06
34)      700. 00        82. 38                101. 09
35)      730. 00        76. 26                105. 24
36)      800. 00        70. 22                109. 57
37)      830. 00        64. 29                114. 56
38)      900. 00        58. 51                120. 30
39)      930. 00        53. 33                126. 50
40)     1000. 00        48. 42                134. 08
41)     1030. 00        44. 28                142. 37
42)     1100. 00        41. 02                152. 24
43)     1130. 00        38. 38                163. 27
44)     1200. 00        37. 27                175. 26
45)     1230. 00        37. 37                187. 42
46)     1300. 00        39. 07                199. 30
47)     1330. 00        41. 47                210. 15
48)     1400. 00        45. 26                219. 43
49)     1430. 00        49. 50                227. 54
50)     1500. 00        54. 48                234. 57
51)     1530. 00        60. 12                241. 05
52)     1600. 00        65. 54                246. 30
53)     1630. 00        71. 50                251. 22
54)     1700. 00        77. 56                255. 50
55)     1730. 00        84. 09                260. 02
56)     1800. 00        90. 26                264. 04
```

Fig. 10

## 5.2. HPL PROGRAMS

### 5.2.1. EPHEM

This HPL program is very similar to the FORTRAN program except it will not compute extended time tables. The listing for the program is in Listing 4. The input responses are almost identical to the FORTRAN version.

**LISTING 4.**

```
"ephem":
"written by Wayne H. Wilson, Jr.":
"Visibility Laboratory":
"Scripps Institution of Oceanography":
"University of California, San Diego":
"La Jolla , California, 92093":
"Ph. (714)-294-5534":
" 06 Mar 1980":
"D - date":
"T - time":
"X - latitude":
"W - longitude":
"r5 - time type (0-LST,1-GMT)":
"r6 - flag for apparent noon calculations":
"r7 - print (0-none,1-date,2-data,3-NA compar)":
"r8 - approximation (0-all,1-no planets,2-no nut)":
"begin":
0→D→T→W→X
0→r5;ent "time type (0-LST,1-GMT)",r5
0→r7;ent "print date-0,data-1",r7;r7+1→r7
0→r8;ent "approx. no planets-1,no moon-2",r8
ent "Day,Month,Year - DD.MMYYYY",D
ent "Latitude - dd.mm",X,"Longitude - ddd.mm",W
0→r6;ent "Apparent noon? yes(1),no(0)",r6
if r6=0;gto "EPHE"
"apparent noon calculations":
cll 'ephems'(X,W,D,T,r5,r6,r7,r8)
fxd 2
if r5=0;prt "Apparent noon   ","  LST",T
if r5=1;prt "Apparent noon   "," GMT  ",T
prt "Z        dd.mmsss",'tohms'(Z);prt " "," ";gto "begin"
"EPHE":
if r5=0;0→T;ent "Time - hhmm.ss Local standard",T
if r5=1;0→T;ent "Time - hhmm.ss GMT",T
cll 'ephems'(X,W,D,T,r5,r6,r7,r8)
"repeat":prt " "," "
0→I;ent "repeat w/ different time,1-yes",I
if I=1;gto "EPHE"
gto "begin"
```

In Figs. 11, 12 and 13, typical output is presented. Figure 11 is an apparent noon calculation, Fig. 12 is for a single time and Fig. 13 is for the same time but with the extended print option specified.

```
time type (0-LST,1-GMT)
            0
print date-0,data-1
            0
approx. no planets-1,no moon-2
            0
Day,Month,Year - DD.MMYYYY
         6.031980
Latitude - dd.mm
           32.000
Longitude - ddd.mm
          120.000
Apparent Noon? yes(1),no(0)
            1
```

INPUT

```
Month              3
Day                6
Year            1980
Full Computation
Lat          32.0000
Long        120.0000
Apparent noon
  LST        1211.10
Z           dd.mmsss
            37.21421
```

OUTPUT

Fig. 11

```
time type (0-LST,1-GMT)
            0
print date-0,data-1
            0
approx. no planets-1,no moon-2
            0
Day,Month,Year - DD.MMYYYY
         6.031980
Latitude - dd.mm
           32.000
Longitude - ddd.mm
          120.000
Single time(0),Table(1)
            0
Time - hhmm.ss Local standard
          1200.00
```

INPUT

```
Month              3
Day                6
Year            1980
Full Computation
Lat          32.0000
Long        120.0000
Time LST     1200.00
A          ddd.mmsss
           175.25548
Z           dd.mmsss
            37.27324
```

OUTPUT

Fig. 12

```
time type (0-LST,1-GMT)
                    0
print date-0,data-1
                    1
approx. no planets-1,no moon-2
                    0
Day,Month,Year - DD.MMYYYY
          6.031980
Latitude - dd.mm
             32.000
Longitude - ddd.mm
            120.000
Single time(0),Table(1)
                    0
Time - hhmm.ss Local standard
            1200.00
```

INPUT

Fig. 13

```
Month              3
Day                6
Year            1980
Full Computation
Lat          32.0000
Long        120.0000
Time LST 1200.00
T           0.801789
d            29285.3
Mean Long
           344.42275
Mean Anomaly
            62.06291
Eccen.       0.01672
Eq. of Center
             1.42345
Radius Vector
             0.99240
Aberration
            -0.00206
Mean Ob 23.26307
Mean Ascension
            22.58485
P Moon     -0.00055
P Venus     0.00028
P Lat by Venus
            -0.00002
P Mars     -0.00007
P Jupiter
            -0.00029
P of Lat by Jup
             0.00001
P Saturn-0.00003
Perturbations
            -0.00066
Long Period
            -0.00051
Nutation of Long
            -0.00089
Nut Obliquity
            -0.00074
Precession
             0.00090
Appar. Long
           346.24208
Solar Latitude
             0.00005
Obliquity
            23.26233
Appar. ascension
            23.09578
Equation of Time
            -0.11093
LHA =   357.12408
Declin. -5.21529
A          ddd.mmsss
           175.25548
Z           dd.mmsss
            37.27324
```

OUTPUT

## 5.2.2. EPHTAB

This program is run the same as EPHEM but outputs the solar position for an extended time period. It is normally used to obtain a listing of the sun position for a whole day. The listing for the program is in Listing 5.

The program assumes there is some type of line printer connected to the HP9825 (*i.e.* HP9871A). Its address is specified by the use of the statement flagged with ++++.....'s.

A typical output is presented in Fig. 14. The only difference between EPHEM and EPHTAB is in the specification of beginning and ending time and in the time increment desired in the table for the latter program. Input responses are obvious.

**LISTING 5.**

```
0: "Ephtab":
1:
2: "written by Wayne H. Wilson, Jr.":
3: "Visibility Laboratory":
4: "Scripps Institution of Oceanography":
5: "University of California, San Diego":
6: "La Jolla , California, 92093":
7: "Ph. (714)-294-5534":
8: " 06 Mar 1980":
9:
10: "+++++++++++++++++++++++++++++":
11: dev "prt",701
12:
13: "D - date":
14: "T - time":
15: "X - latitude":
16: "W - longitude":
17: "r5 - time type (0-LST,1-GMT)":
18: "r6 - flag for apparent noon calculations":
19: "r7 - print (0-none,1-date,2-data,3-NA compar)":
20: "r8 - approximation (0-all,1-no planets,2-no nut)":
21:
22: "begin":
23:
24: 0→D→T→W→X
25: 0→r5;ent "time type (0-LST,1-GMT)",r5
26: 0→r7;ent "print date-0,data-1",r7;r7+1→r7
27: 0→r8;ent "approx. no planets-1,no moon-2",r8
28: ent "Day,Month,Year - DD.MMYYYY",D
29: ent "Latitude - dd.mm",X,"Longitude - ddd.mm",W
30: 0→r4;ent "Single time(0),Table(1)",r4
31: if r4=0;gto "EPHE"
32:
33: "Table option":
34:
35: ent "beginning time - HHMM.SS",B
36: ent "ending time - HHMM.SS",E
37: ent "time increment - HMM.SS",I
38: int(B/100)→H;B-H*100→M
39: int(E/100)→S;E-S*100→T;S+T/60→N
40: int(I/100)→S;I-S*100→T;S*60+T→I
```

```
41: abs(W)→C;sgn(W)*(int(C)+frc(C)*100/60)→L
42: int((abs(L)+7.5)/15)*15*sgn(L)→L
43:
44: dsp "Top of Printer - push Continue";stp
45:
46: fmt 0,15x,"Solar Position Table"
47: wrt "prt.0"
48: fmt 1,/,10x,"Date(DD/MM/YYYY)",2x,f2.0,"/",f2.0,"/",f4.0
49: wrt "prt.1",int(D),int(frc(D)*100),int(frc(D*100)*10000)
50: fmt 2,/,5x,"Latitude",f10.3,5x,"Longitude",f10.3
51: wrt "prt.2",X,W
52: fmt 3,/,10x,"Standard Meridian",f10.3
53: wrt "prt.3",L
54:
55: 1→r6;0→r7;0→r8
56: cll 'ephems'(X,W,D,T,r5,r6,r7,r8)
57: fmt 4,/,7x,"Apparent Noon (HHMM.SS)",2x,f7.2
58: wrt "prt.4",T
59: fmt 5,/,15x,"angles are DDD.MM"
60: wrt "prt.5"
61: fmt 6,/,5x,"Time",5x,"Zenith Angle",5x,"Azimuth Angle",/
62: wrt "prt.6"
63: 0→r6;gto "S101"
64:
65: "S100":
66: M+I→M
67: "S101":if M>=60;M-60→M;H+1→H;if H>24;H-24→H;D+1→D
68: H+M/60→Y;if Y>1.01*N;gto "begin"
69: H*100+M→T
70: cll 'ephems'(X,W,D,T,r5,r6,r7,r8)
71: fxd 2
72: fmt 5,3x,f7.2,6x,f5.2,13x,f6.2
73: wrt "prt.5",T,'tohms'(Z),'tohms'(A)
74: gto "S100"
75:
76: "EPHE":
77:
78: if r5=0;0→T;ent "Time - hhmm.ss Local standard",T
79: if r5=1;0→T;ent "Time - hhmm.ss GMT",T
80:
81: cll 'ephems'(X,W,D,T,r5,r6,r7,r8)
82:
83: "repeat":prt " "," "
84:
85: 0→I;ent "repeat w/ different time,1-yes",I
86: if I=1;gto "EPHE"
87: gto "begin"
88:
```

```
time type (0-LST,1-GMT)
              0
print date-0,data-1
              0
approx. no planets-1,no moon-2
              0
Day,Month,Year - DD.MMYYYY
        6.031980
Latitude - dd.mm
        32.000
Longitude - ddd.mm
        120.000
Apparent Noon? yes(1),no(0)
              0
Single time(0),Table(1)
              1
beginning time - HHMM.SS
        600.00
ending time - HHMM.SS
        1800.00
time increment - HMM.SS
        100.00
```

INPUT

Solar Position Table

Date(DD/MM/YYYY) 6/ 3/1980

Latitude 32.000 Longitude 120.000

Standard Meridian 120.000

Apparent Noon (HHMM.SS) 1211.10

angles are DDD.MM

| Time | Zenith Angle | Azimuth Angle |
|---|---|---|
| 600.00 | 95.15 | 93.09 |
| 700.00 | 82.38 | 101.09 |
| 800.00 | 70.22 | 109.57 |
| 900.00 | 58.51 | 120.30 |
| 1000.00 | 48.42 | 134.08 |
| 1100.00 | 41.02 | 152.24 |
| 1200.00 | 37.27 | 175.26 |
| 1300.00 | 39.07 | 199.30 |
| 1400.00 | 45.26 | 219.43 |
| 1500.00 | 54.48 | 234.57 |
| 1600.00 | 65.54 | 246.30 |
| 1700.00 | 77.56 | 255.50 |
| 1800.00 | 90.26 | 264.04 |

OUTPUT

Fig. 14

### 5.2.3. SUNPOS

This program computes the solar position and plots the zenith angle versus azimuth on prepared polar graph paper. The listing is given in Listing 6. The program assumes a plotter is available (*e.g.* HP9872A) and its bus address is specified in the listing at the statement flagged with ++++.....'s.

In Fig. 15, an example of the graph paper used is given. In Fig. 16, an example output is illustrated. The program as noted in Fig. 16 also computes sunrise and sunset. These times are assumed to be those times when the "true" or apparent center of the solar disk is on the horizon. No correction is made for refractive effects.

The running of the program is obvious from the displayed messages. The plotter should be set up so that the p1 and p2 points are set as marked on the graph paper.

**LISTING 6.**

```
0: "sunpos ":
1:
2: "program to plot sun position on plotter":
3:
4: "written by Wayne H. Wilson,Jr.":
5: "Visibility Laboratory":
6: "Scripps Institution of Oceanography":
7: "University of California, San Diego":
```

```
8: "La Jolla, California, 92093":
9: "Ph. (714)-294-5534":
10: " 26 Feb 1980":
11:
12: dim A[100],Z[100],X[100],Y[100]
13:
14: "++++++++++++++++++++++++++++":
15: dev "plt",705
16:
17: ent "Latitude - dd.mm",X
18: ent "Longitude - ddd.mm",W
19: ent "Date - DD.MMYYYY",D
20: ent "Begin time -  hhmm.ss",r0
21: ent "End time - hhmm.ss",r1
22: ent "Time increment - hhmm.ss",r2
23:
24: "use LST":0→r5
25: "no print":0→r7
26: "complete calculation":0→r8
27: 0→r6
28:
29: 0→r9
30: r0-r2→Y
31: "loop":
32: Y+r2→Y
33: if frc(Y/100)*100>=60;Y-60+100→Y;jmp 0
34: if Y>r1;gto "end loop"
35: cll 'ephems'(X,W,D,Y,r5,r6,r7,r8)
36: prt "time",Y
37: if Z>100;gto "loop"
38: r9+1→r9
39: Y→A[r9]
40: Z*sin(A)→X[r9]
41: Z*cos(A)→Y[r9]
42: gto "loop"
43:
44: "end loop":
45:
46: "find noon":1→r6
47: cll 'ephems'(X,W,D,Y,r5,r6,r7,r8)
48: "Found noon":Y→r10
49: 0→r6
50: dsp "now  sunrise - ho hum";4→r1;10→r2
51: "Loop1":
52: (r1+r2)/2→r3
53: int(r3)*100+int(frc(r3)*60)+frc(frc(r3)*60)*60/100→Y
54: cll 'ephems'(X,W,D,Y,r5,r6,r7,r8)
55: fxd 2;dsp "now sunrise - ho hum",Y
56: fxd 4
57: if Zmod90<.01;gto "foundsunrise"
58: if Z<90;r3→r2;gto "Loop1"
59: r3→r1;gto "Loop1"
60: "foundsunrise":Y→r11
61: 14→r1;22→r2;dsp "finally sunset"
62: "Loop2":
63: (r1+r2)/2→r3
64: int(r3)*100+int(frc(r3)*60)+frc(frc(r3)*60)*60/100→Y
65: cll 'ephems'(X,W,D,Y,r5,r6,r7,r8)
66: fxd 2;dsp "finally sunset",Y
67: fxd 4
68: if Zmod90<.01;gto "foundsunset"
69: if Z>90;r3→r2;gto "Loop2"
```

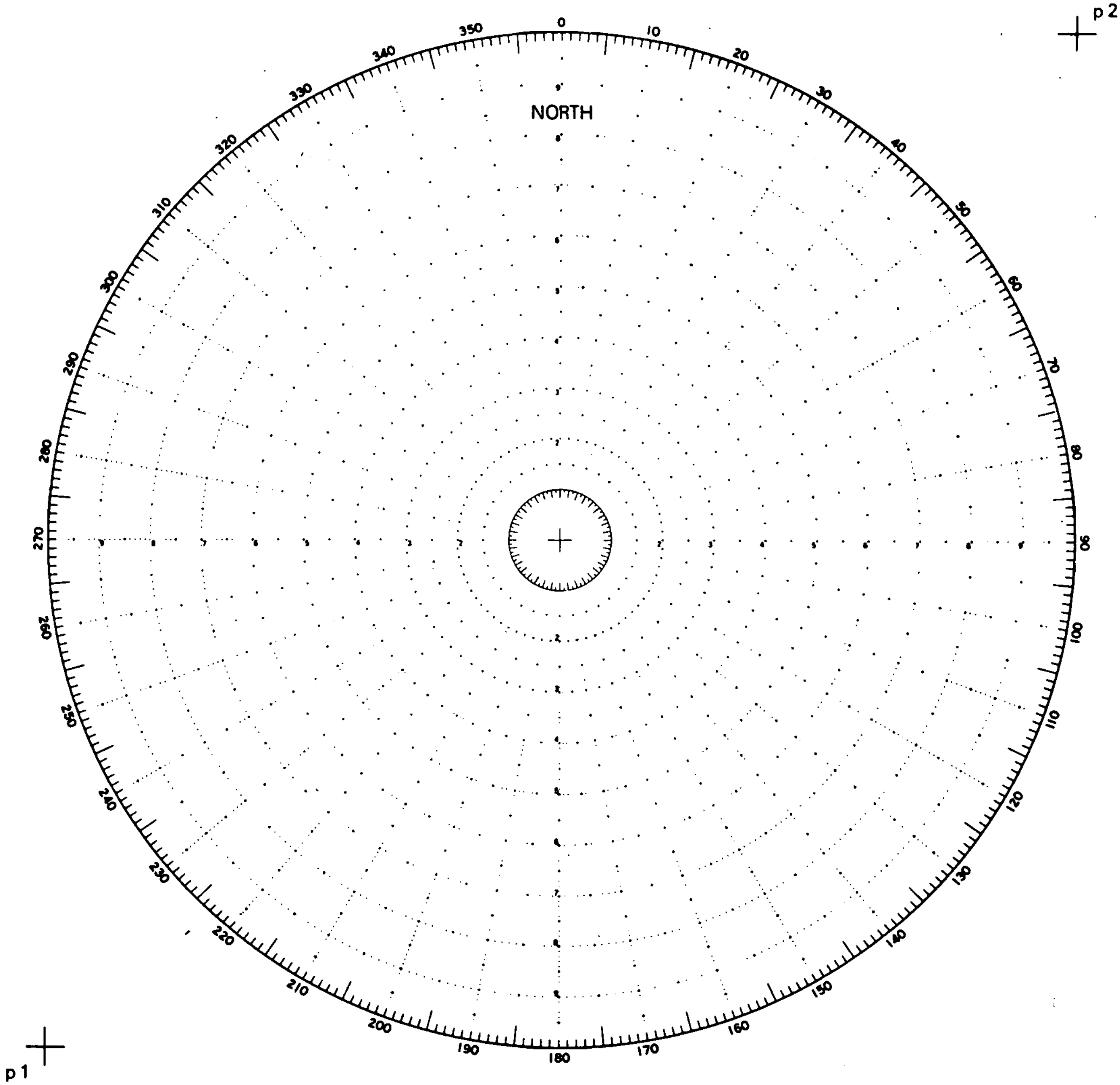
```
70: r3→r1;gto "Loop2"
71: "foundsunset":Y→r12
72: fxd 2
73: prt "# of points",r9
74:
75: "plot":
76:
77: dsp "ready plotter";stp
78: wrt "plt","DF";lbl " ";pclr
79: scl -100,100,-100,100
80: pen# 1
81: csiz 1,1,1,0
82: fxd 0
83: -2→r1
84: for I=1 to r9
85: wrt "plt","sm*"
86: plt X[I],Y[I],r1
87: if frc(A[I]/100)*100#0;gto "noplttime"
88: pen
89: wrt "plt","sm"
90: plt X[I]-8,Y[I]+2.5,1
91: lbl A[I]
92: plt X[I],Y[I],-2
93: "noplttime":
94: 0→r1
95: next I
96:
97: pen
98: wrt "plt","sm"
99: csiz 1,1,1,0
100: fxd 1
101: plt -53.71,-121.52,1;cplt -1,0
102: lbl int(X),frc(X)*100," "
103: if X>=0;lbl "N"
104: if X<0;lbl "S"
105: plt -50.35,-130.25,1;cplt -1,0
106: lbl int(W),frc(W)*100," "
107: if W>=0;lbl "W"
108: if W<0;lbl "E"
109: fxd 0
110: plt 28.21,-121.77,1
111: lbl int(frc(D)*100),"/"
112: cplt -1,0;lbl int(D),"/"
113: cplt -1,0;lbl frc(D*100)*10000
114: "Noon":plt 29.4,-130.55,1;lbl r10
115: "sunrise":plt 33.5,-139.2,1;lbl r11
116: "sunset":plt 32,-147.87,1;lbl r12
117: dsp "finished";stp
118:
```

## 5.3. PROGRAM AVAILABILITY

The programs are available from the author for a nominal charge for the tapes. If tapes are sent with the request there will be no charge.

The FORTRAN version is available on 9-track 800 or 1600 BPI tape written in EBCDIC or ASCII in 80 byte records.

The HPL versions are available on HP data cartridges (HP9162-0061 or equivalent).




LAT. ____________________

LONG. ____________________

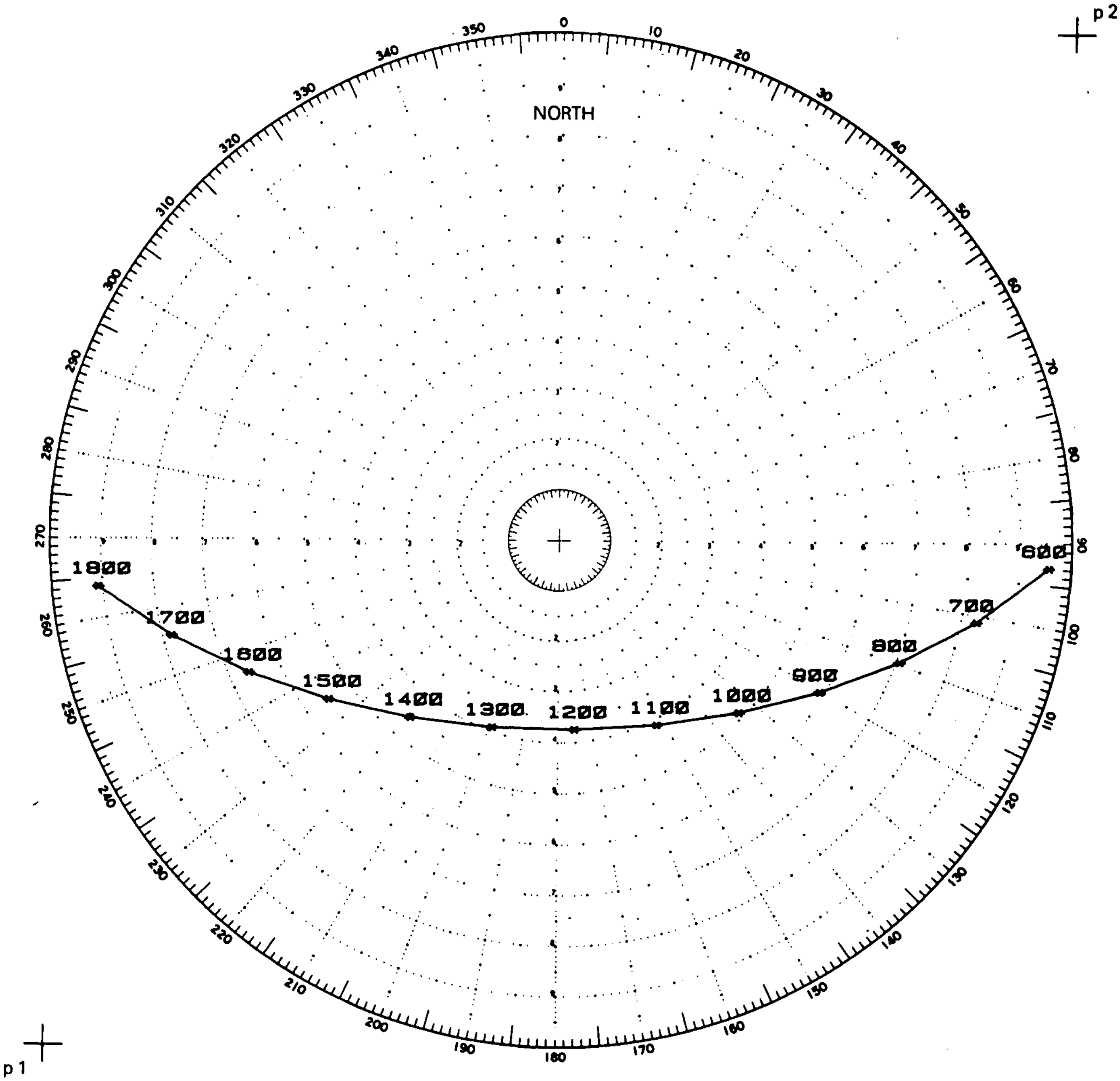
DATE ____________________

NOON ____________________

SUNRISE ____________________

SUNSET ____________________

Fig. 15




LAT. 32. Ø Ø. Ø N

LONG. 12Ø. Ø Ø. Ø W

DATE 3/8/198Ø

NOON 1211

SUNRISE 825

SUNSET 1758

Fig. 16

## ACKNOWLEDGEMENTS

The programs were written over a period of seven years (1972-1979) and at times were funded by a variety of sources. The vast majority of the funding was obtained through NOAA grant GR-04-6-158-44033.

The author greatly appreciates the assistance in preparing this report given by Charleen Epstein for programming, Alicia G. Hill for typing and Carrie Wilson for editing the manuscript.

## REFERENCES

*U. S. Nautical Almanac Office* "Explanatory Supplement to the Astronomical Ephemeris and the American Ephemeris on a Nautical Almanac", Her Majesty's Stationery Office, London (1961).

*U. S. Nautical Almanac Office* "The American Ephemeris and Nautical Almanac", U.S. Government Printing Office, Washington, D.C., (various years).

Newcomb, S., "Tables of the Sun", Astronomical Papers prepared for the use of the American Ephemeris and Nautical Almanac **6** (part 1), (1895).

Moulton, F. R., "An Introduction to Celestial Mechanics", MacMillian, New York, (1914).

Woolard, E. W., "Theory of the rotation of the Earth around its center of Mass", Astronomical Papers prepared for the use of the American Ephemeris and Nautical Almanac **15** (part 1), (1953).

*U. S. Nautical Almanac Office* "Improved Lunar Ephemeris 1957-1959", U. S. Government Printing Office, Washington, D.C. (1954).

*U. S. Nautical Almanac Office* "Almanac for Computers - 1978", U. S. Government Printing Office, Washington, D.C. (1978).

Lutus, P., "Following the Sun with a Hand Held Calculator", Optical Spectra, p. 64-66 (July 1977).

# APPENDIX

The following is a list of variables in the ephemeris programs.

## HPL INPUT ARGUMENTS

| | |
|---|---|
| *p1* | Latitude (DD.MM, DD = degrees, MM = minutes, + north, - south) |
| *p2* | Longitude (DDD.MM, DDD = degrees, MM = minutes, + west, - east) |
| *p3* | Date (DD.MMYYYY, DD = day, MM = month (*i.e.* 01, 02, --, 12), YYYY = year) |
| *p4* | Time (HHMM.SS - 24 hour clock - Local standard if *p5* = 0, Greenwich mean if *p6* = 1 |
| *p5* | Time type (0 - Local standard, 1 - Greenwich mean) |
| *p6* | Apparent noon calculations (0 - no, 1 - yes) |
| *p7* | Print options (0 - nothing printed, 1 - date, location and sun position, 2 - Nautical Almanac comparison) |
| *p8* | Approximation (0 - full computation, 1 - no planetary effects, 2 - no lunar effects) |

## VARIABLES PASSED BACK TO CALLING PROGRAM

| | |
|---|---|
| *A* | Azimuth angle (in degrees) |
| *R* | Radius vector (1=mean earth sun distance) |
| *Z* | Zenith angle of sun (in degrees) |
| *p5* | Apparent noon time if *p6* = 1, (HHMM.SS, Local standard if *p5* = 0, Greenwich mean if *p5* = 1) |

## FORTRAN INPUT ARGUMENTS

| | |
|---|---|
| *LAT* | Latitude (in degrees & fraction of degrees) |
| *LONG* | Longitude (in degrees & fractions of degrees) |
| *TIME* | Time of day (HHMM.SS - Local standard time if *p5* = 0, Greenwich mean if *p6* = 1) |
| *DAY* | Day of month (1-31) |
| *MONTH* | Month (1-12) |
| *YEAR* | Year (1700-2100) |

P5, P6, P7 AND P8 ARE SAME AS ABOVE

## FORTRAN RETURNED ARGUMENTS

| | |
|---|---|
| *A* | Azimuth (in degrees) |
| *R* | Radius vector |
| *Z* | Zenith angle (in degrees) |
| *TIME* | Apparent noon time if *p6* = 1 (HHMM.SS - 24 hour clock, Local standard if *p5* = 0, Greenwich mean if *p5* = 1) |

## GENERAL VARIABLES

All variables have units of degrees and fraction of degrees unless otherwise stated.

*p0, p11 - p15* - are used as scratch (temporary) variables
*p10* - print control - nominally *p7*
*p16* - day (1-31)
*p17* - month (1-12)
*p18* - year (1700-2100)
*p19* - latitude (in degrees)
*p20* - long (in degrees)
*p21* - time (time of day in fractions of day)
*p22* - day count from 1900 JAN 0.5 *ET*
*p23* - *T* - fraction of century from 1900 JAN 0.5 *ET*
*p24* - *L* - mean longitude of sun
*p25* - *M* - mean anomaly of sun
*p26* - *e* - eccentricity - (no units)
*p27* - $\nu$ - true anomaly
*p28* - *l* - Moon's mean anomaly
*p29* - $\Delta\lambda_A$ - aberration
*p30* - D - Moon's mean elongation
*p31* - $\Omega$ - Moon's longitude of mean ascending mode
*p32* - $\gamma$ - Moon's mean longitude
*p33* - $\Delta\lambda$ moon perturbation of sun's longitude
*p34* - $\Delta\psi$ - nutation of solar longitude
*p35* - $\Delta\epsilon$ - nutation of obliquity
*p36* - $\delta L$ - inequalities of long period in mean longitude
*p37* - Venus mean anomaly
*p38* - Mars mean anomaly
*p39* - Jupiter mean anomaly
*p40* - Saturn mean anomaly
*p41* - $\lambda$ - apparent longitude
*p42* - *P* - precession
*p43* - $\epsilon_m$ - mean obliquity
*p44* - $\alpha$ - apparent right ascension
*p45* - $\alpha_m$ - mean right ascension

*p46* - *Eq. of Time* - equation of time

*p47* - $\delta_s$ - declination

*p48* - $h_m$ - hour angle

*p49* - *LHA* - local apparent hour angle

*p51* - Greenwich mean time in degrees

*p53* - standard time zone longitude

*p54, p55* - variables used in apparent noon calculations

*p56* - time variable used in apparent noon calculation

*p57* - (*ET* - *UT*) - difference of Ephemeris and Universal Time

*p60* - $\beta$ solar latitude

*p61* - $\Delta\beta_v$ - perturbation of solar latitude by Venus

*p62* - $\Delta\beta_j$ - perturbation of solar latitude by Jupiter

*p63* - moon mean argument of latitude

*p70* - perturbation of solar longitude by Venus

*p71* - perturbation of solar longitude by Mars

*p72* - perturbation of solar longitude by Jupiter

*p73* - perturbation of solar longitude by Saturn

*p75* - $\epsilon$ - obliquity