NOAA Technical Memorandum NESS 111

EARTH LOCATING IMAGE DATA OF SPIN-STABILIZED GEOSYNCHRONOUS SATELLITES

Washington, D.C.
August 1980

U.S. DEPARTMENT OF COMMERCE / National Oceanic and Atmospheric Administration / National Environmental Satellite Service

NOAA TECHNICAL MEMORANDUMS

National Environmental Satellite Service Series

The National Environmental Satellite Service (NESS) is responsible for the establishment and operation of NOAA's environmental satellite systems.

NOAA Technical Memorandums facilitate rapid distribution of material that may be preliminary in nature and so may be published formally elsewhere at a later date. Publications 1 to 20 and 22 to 25 are in the earlier ESSA National Environmental Satellite Center Technical Memorandum (NESCTM) series. The current NOAA Technical Memorandum NESS series includes 21, 26, and subsequent issuances.

Publications listed below are available (also in microfiche form) from the National Technical Information Service, U.S. Department of Commerce, Sills Bldg., 5285 Port Royal Road, Springfield, VA 22161. Prices on request. Order by accession number (given in parentheses). Information on memorandums not listed below can be obtained from Environmental Data and Information Service (D822), 6009 Executive Boulevard, Rockville, MD 20852.

NESS 66 A Summary of the Radiometric Technology Model of the Ocean Surface in the Microwave Region. John C. Alishouse, March 1975, 24 pp. (COM-75-10849/AS)

NESS 67 Data Collection System Geostationary Operational Environmental Satellite: Preliminary Report. Merle L. Nelson, March 1975, 48 pp. (COM-75-10679/AS)

NESS 68 Atlantic Tropical Cyclone Classifications for 1974. Donald C. Gaby, Donald R. Cochran, James B. Lushine, Samuel C. Pearce, Arthur C. Pike, and Kenneth O. Poteat, April 1975, 6 pp. (COM-75-10676/AS)

NESS 69 Publications and Final Reports on Contracts and Grants, NESS-1974. April 1975, 7 pp. (COM-75-10850/AS)

NESS 70 Dependence of VTPR Transmittance Profiles and Observed Radiances on Spectral Line Shape Parameters. Charles Braun, July 1975, 17 pp. (COM-75-11234/AS)

NESS 71 Nimbus-5 Sounder Data Processing System, Part II: Results. W. L. Smith, H. M. Woolf, C. M. Hayden, and W. C. Shen, July 1975, 102 pp. (COM-75-11334/AS)

NESS 72 Radiation Budget Data From the Meteorological Satellites, ITOS 1 and NOAA 1. Donald H. Flanders and William L. Smith, August 1975, 20 pp. (PB-246-877/AS)

NESS 73 Operational Processing of Solar Proton Monitor Data (Revision of NOAA TM NESS 49). Stanley R. Brown, September 1975, 15 pp. (COM-73-11647)

NESS 74 Monthly Winter Snowline Variation in the Northern Hemisphere From Satellite Records, 1966-75. Donald R. Wiesnet and Michael Matson, November 1975, 21 pp. (PB-248-437/6ST)

NESS 75 Atlantic Tropical and Subtropical Cyclone Classifications for 1975. D. C. Gaby, J. B. Lushine, B. M. Mayfield, S. C. Pearce, and K.O. Poteat, March 1976, 14 pp. (PB-253-968/AS)

NESS 76 The Use of the Radiosonde in Deriving Temperature Soundings From the Nimbus and NOAA Satellite Data. Christopher M. Hayden, April 1976, 19 pp. (PB-256-755/AS)

NESS 77 Algorithm for Correcting the VHRR Imagery for Geometric Distortions Due to the Earth Curvature, Earth Rotation, and Spacecraft Roll Attitude Errors. Richard Legeckis and John Pritchard, April 1976, 31 pp. (PB-258-027/AS)

NESS 78 Satellite Derived Sea-Surface Temperatures From NOAA Spacecraft. Robert L. Brower, Hilda S. Gohrband, William G. Pichel, T. L. Signore, and Charles C. Walton, June 1976, 74 pp. (PB-258-026/AS)

NESS 79 Publications and Final Reports on Contracts and Grants, NESS-1975. National Environmental Satellite Service, June 1976, 10 pp. (PB-258-450/AS)

NESS 80 Satellite Images of Lake Erie Ice: January-March 1975. Michael C. McMillan and David Forsyth, June 1976, 15 pp. (PB-258-458/AS)

NESS 81 Estimation of Daily Precipitation Over China and the USSR Using Satellite Imagery. Walton A. Follansbee, September 1976, 30 pp. (PB-261-970/AS)

NESS 82 The GOES Data Collection System Platform Address Code. Wilfred E. Mazur, Jr., October 1976, 26 pp. (PB-261-968/AS)

NESS 83 River Basin Snow Mapping at the National Environmental Satellite Service. Stanley R. Schneider, Donald R. Wiesnet, and Michael C. McMillan, November, 1976, 19 pp. (PB-263-816/AS)

NESS 84 Winter Snow-Cover Maps of North America and Eurasia From Satellite Records, 1966-1976. Michael Matson, March 1977, 28 pp. (PB-267-393/AS)

NESS 85 A Relationship Between Weakening of Tropical Cyclone Cloud Patterns and Lessening of Wind Speed. James B. Lushine, March 1977, 12 pp. (PB-267-392/AS)

NESS 86 A Scheme for Estimating Convective Rainfall From Satelliue Imagery. Roderick A. Scofield and Vincent J. Oliver, April 1977, 47 pp. (PB-270-762/AS)

(Continued on inside back cover)

NOAA Technical Memorandum NESS 111

EARTH LOCATING IMAGE DATA OF SPIN-STABILIZED GEOSYNCHRONOUS SATELLITES

Larry N. Hambrick
Office of Systems Integration

Dennis R. Phillips
Scientific Programming and Applied Mathematics, Inc.
Contracts: DOC 7-35229, DOC 01-8-M01-5347,
NA79-KAC00026, and NA79-SAC00744

Washington, D.C.
August 1980

UNITED STATES
DEPARTMENT OF COMMERCE
Philip M. Klutznick, Secretary

NATIONAL OCEANIC AND
ATMOSPHERIC ADMINISTRATION
Richard A. Frank, Administrator

National Environmental
Satellite Service
David S. Johnson, Director

PREFACE

The National Environmental Satellite Service (NESS) operates a central facility for determining accurate earth-location information for the images derived from its Geostationary Operational Environmental Satellite (GOES). The facility is called the VISSR Image Registration and Gridding System (VIRGS).

The main purpose of this document is to set forth a mathematical formulation of the entire earth-location process for a spin-stabilized geosynchronous satellite. The implementation of this process on the VIRGS is described. Preceding the main topics is a discussion of the relevant aspects of the GOES system. An error analysis and some actual results are presented in the final section.

The original objective of the VIRGS was that it provide a stand-alone capability to operationally produce highly accurate earth-location parameters. The "stand-alone" feature is important because it eliminates the expense of operating ranging stations and performing separate data processing tasks. "High accuracy" is as important; without it the users of GOES imagery are faced with the undesirable alternatives of either using poorly located data or determining accurate earth-location parameters on their own.

The objectives of the VIRGS have not yet been fully realized. This is due in large to the lag-time between the development of a new technique and its full operational implementation. A technique that uses the positions of stars in the VISSR image to determine some of the earth-location parameters has not yet been implemented. The star position technique streamlines the earth-location process and provides the needed accuracy and stand-alone features. Also remaining to be fully implemented is the closed-loop capability of the VIRGS--a capability for complete quality control of the earth-location process.

The authors' intent is to show that the objectives of the VIRGS can be achieved; and it is their hope that the demonstration given herein will aid in that achievement.

Some acknowledgements are deserved. Hank Schmidt began casually reading an early version of the document and ended up making significant improvements to the presentation. Jim Ellickson and Matt Jurotich reviewed an early version. Ron Gird is thanked for his encouragement and energetic participation in the VIRGS implementation. Vince Oliver and others, inside and outside of NESS, who as users actively supported improvements to the operational earth-location of VISSR images, were instrumental in bringing about VIRGS. The McIDAS team at the University of Wisconsin (SSEC) planted the seed and then provided their know-how in delivering the basic VIRGS. Eric

Smith, J.T. Young and others were responsible for the development and demonstration of the early navigation software on McIDAS.

VIRGS is operated and maintained daily, in accordance with procedures and guidelines passed through management, by a group consisting of satellite meteorologists (oddly enough), software analysts, computer operators, and electronic technicians. Members of this group are recognized for their daily efforts.

Contents

Appendices:

A. Converting a pointing vector to line and element numbers

B. Determination of attitude and orbit plane orientation

C. Accuracy of approximate orbit equation

D. Gaussian elimination

E. Fortran listings of orbit and attitude determination routines: FINDAT and UPGORB

F. Fortran listings of standard geometry transformation routines: EARLOC and IMGLOC

EARTH LOCATING IMAGE DATA OF SPIN-STABILIZED GEOSYNCHRONOUS SATELLITES

Larry N. Hambrick
Office of Systems Integration

Dennis R. Phillips
Scientific Programming and Applied Mathematics, Inc.

ABSTRACT. Aspects of the GOES system relevant to earth-location are reviewed. The essential features of an earth-location capability that satisfies the needs of users of the data are specified. A mathematical formulation of the geometry and orbit/attitude determinations, including the actual algorithms, is presented. The implementation of these methods on VIRGS is described. Finally, the rationale for a centralized, once-for-all users earth-location service and an error budget with some actual results are presented.

I. INTRODUCTION

The Geostationary Operational Environmental Satellites (GOES) are examples of spin-stabilized geosynchronous satellites. The GOES provide continuous viewing of the weather features over the Americas and their adjacent waters. The images, or views, are obtained every half-hour, and sometimes more frequently, from each of two satellites. The imaging instrument of the GOES is called a Visible and Infra-red Spin-Scan Radiometer (VISSR). About a dozen users receive VISSR data directly from the GOES and over a hundred others are served by a Central Data Distribution Facility.

The GOES program began in 1974 and is expected to remain in operation throughout the 1980's. During the next ten years the number of uses and users of GOES data is expected to increase.

In order for the VISSR imagery to be useful, it must be earth-located. "Earth-location" means different things to different users. It can mean (1) the ability to determine precisely the earth-location (latitude and longitude) of any feature in an image; (2) the presence of fixed surface features of earth in time-lapsed (animated) sequences of images; or (3) having geographic and political boundaries accurately laid over the images.

Today many users of VISSR images use manual techniques to shift and align images. These methods are cumbersome, time consuming, and inaccurate. They provide, at best, partial solutions to the earth-location problem. A general solution --one that enables a user to map any picture element (VISSR field-of-view) into precise earth coordinates--must be based on a complete mathematical formulation of the problem. There are two reasons for this: (1) the basic geometry that describes a mapping or transformation from satellite/instrument (GOES/VISSR) coordinates to earth coordinates is complex; and (2) the dynamics of the satellite's orbit and attitude must be modelled.

With the information needed to perform accurate earth-location inserted into the VISSR data stream, all direct VISSR users would benefit by not having to determine it on their own.

II. GENERAL REVIEW

A. The Satellite

A satellite in an ideal geosynchronous orbit remains motionless with respect to an earth-based observer. For this to occur, the satellite must be in the equatorial plane and the centrifugal force of its orbital motion must, at all times, equal the gravitational force between it and the earth. However, there are perturbating forces that make it highly impractical to maintain such an ideal orbit.

The GOES orbit is not perfectly circular and it is slightly inclined to the equatorial plane. Furthermore, the perturbative forces, caused by anomolies in the earth's gravitational force field, the gravitational affect of the moon and the sun, and the pressure of the solar wind continually vary the orbit. Departure from an ideal geosynchronous orbit causes the satellite subpoint to trace out a "distorted figure eight" during its one-day orbital period.

The VISSR is rigidly mounted in an upright position to an axis about which the satellite spins at 100 rpm. Ideally the attitude of the spin axis would be normal to the orbit plane. An ideal orbit and spin axis attitude would produce the same "perspective" in all VISSR images. In practice external torques from solar radiation and impulses from VISSR scan stepping cause the satellite spin axis to wobble around a vector normal to the orbit plane in a manner governed by rigid body dynamics. The practical manifestation is a nearly linear precession of the spin axis over a period of several days accompanied by a small nutation with a cycle period of several seconds.

A set of orbit parameters determines the satellite's position and a set of attitude parameters determines the orientation of the satellite's spin axis relative to the earth, as functions of time. The satellite's position and orientation are controlled so as to stay within certain bounds and hence avoid excessive apparent earth motion in the VISSR images. The method of control is to fire the on-board rockets in a manner that appropriately alters the orbit and attitude. These are called "maneuvers" and such maneuvers are vital in maintaining the quality of the GOES service. When these maneuvers occur, significant discontinuities in the orbit and attitude result. The maneuvers are usually scheduled several days in advance and can be made as frequently as once a week. The orbit and attitude discontinuities caused by satellite manuevers interrupt the process of predicting the orbit and attitude state with a mathematical model. Thus the earth location capability is also interrupted. A navigational system, such as VIRGS, requires accurate orbit and attitude states within several hours following a maneuver.

B. The VISSR

The imaging instrument on GOES is the VISSR. The VISSR is rigidly mounted to the satellite's body. An image is formed by the VISSR Field of View (FOV) scanning the earth, west to east, as the satellite spins. After each spin, a scan mirror is moved (stepped down one notch) so that the next scan sweep is slighly south of the previous sweep.

Electronic sampling generates equal angle (or time) spacing between successive image elements on each scan line. Throughout each scan line the scan mirror is at a fixed position. The scan mirror stepping increment is 192 μradians and the sampling interval is 84μradians for the infrared channel(s) and 21μradians for the visible channel.

There is a single infra-red detector. It has a 192 by 84 μradian field of view and provides 7 by 3 km subpoint resolution. There are 8 vertically aligned visible detectors which are sampled simulataneously giving a 21μradian square field of view, for 0.8 km resolution at the subpoint.

The normal operation of the VISSR is to acquire a full earth disc image frame (1821 successive scans) once every 30 minutes. Sometimes the VISSR is commanded to acquire less than a full earth disc image by scanning over any interval of scan positions within the range of 1 to 1821. When this is done, images can be obtained for example during 3, 7, and 15 minute intervals, depending on the size of the images. The earth-location capability described herein is compatible with these various imaging modes.

The mechanical alignment of the VISSR to the satellite spin axis and the sun pulse detector (see next paragraph) is imperfect. This creates significant values for three misalignment angles that are analogous to the roll, yaw, and pitch angles of a spacecraft. The VISSR's mechanical frame could be tilted forward (pitch), tilted to the side (yaw) as well as being rotated around the vertical (roll). The determination of these misalignment angles, which are constant over periods of weeks, is a vital step in achieving accurate earth location.

A sun sensor and an earth-edge detector are used to initiate the sampling along each scan line. During each spin of the satellite the delay between sensing the sun and detecting the earth's west edge is measured. The delay that is measured on one spin n is used to initiate sampling on spin n+1. This process merely ensures that the earth view is contained in the data acquisition interval. The task of precisely referencing the west border of the image frame on successive scan lines is performed on the ground in real-time by the Synchronous Data Buffer.

C. Functions of the Synchronous Data Buffer (SDB)

The VISSR data are transmitted in real-time at 28 Mbps* to a Command and Data Acquisition (CDA) facility on the ground. At the CDA an SDB is used to "stretch" the data stream for retransmission through GOES at 1.74 Mbps. The stretching operation uses the 340° portion of each satellite revolution, when the VISSR is not pointing at the earth, to rebroadcast the VISSR data at 1/16 the raw data rate. Users of the data are

____

*Megabits per second (Mbps)

therefore interested in the form of the SDB's VISSR output rather than the satellite's raw output.

The SDB accomplishes other critical tasks in the process of stretching the data stream. From the point of view of earth-locating the data, one of the SDB's most critical functions is to center each VISSR scan line on the earth. The SDB, while maintaining synchronization with the satellite's spin rate, precisely times the start (west border) of each stretched scan line. To do this the time of the sun pulse and a "beta" angle are used. The "beta" angle is the angle subtended at the satellite by lines projected from the sun and earth to the satellite, in the satellite's spin plane. The beta angle is computed by a sun position model using the satellite's orbit and attitude and the VISSR misalignment parameters. This computation is readily available (as a by-product) from an earth-location capability such as VIRGS. The value of beta used on each scan line references each sample of that scan line with respect to the sun's position. The well-defined relationship between beta and every sample of the scan line provides an absolute reference. Hence, even if the "beta" provided to the SDB is inaccurate, the user of the data can still reference the imagery data to the sun's position. The "beta" used by the SDB is documented in the stretched VISSR data stream (SDB output).

The image frame as seen by the user consists of up to 1821 scan lines, each with 3822 infrared elements and 15288 visible elements across the line. Each infrared element is referenced to a coincident array of 8 by 4 visible picture elements (pixels). The referencing along each scan line is accomplished by the SDB. The referencing across scan lines is governed by the VISSR step scan characteristics.

For each scan line the actual VISSR scan line number and the value of "Beta" are contained in a "documentation block" that is prefixed to each line of stretched VISSR data. The documentation block also contains values of the parameters that specify the orbit, the attitude, and the misalignment angles. This block is described in the appendix of reference (6). These parameters are computed at the central facility (VIRGS) and transferred to the SDB. They are intended to furnish the users of VISSR data with the information needed to perform predictive earth-location. Clearly any significant errors in these parameters force the user to either work with mislocated image data, or alternatively, to do the navigation independently. If the orbit-attitude documentation block contains precise, or the best achievable, values on a highly reliable basis, the user needs only a standardized geometry routine to which the documented parameters are passed as inputs.

Of interest to some types of users is the NESS standard full disc grids. They are embedded by the SDB as the 9th bit of each 8 bit infrared (IR) sample. The locations of the NESS standard grid points are generated by VIRGS and transferred to the SDB. For accuracies within the resolution of an IR pixel these grids can be used. Thus a user can avoid the large computational task of applying the geometry transform to some 30,000 grid points that outline political and geographic boundaries. For users whose applications require full visible resolution grid point location, the documented orbit and attitude parameters should be used.

Figure 1 illustrates the overall data flow.

D. The Earth-Location Capability

The foundation of a VISSR earth-location capability is the geometrical transformation and its inverse. Given the earth-location parameters (the satellite's position and attitude and the VISSR's misalignment angles) the transformation maps the J^{th} sample on the I^{th} scan line to the latitude and longitude of the point on the earth to which that sample corresponds.

The most convenient and accurate method for determining the values of the earth-location parameters uses earth landmarks and stars that are discernable in the VISSR image. The image coordinates (I, J) of known stars, over an interval of time, specify the values of the satellite's attitude and the VISSR's misalignment angles. These values and the coordinates (I, J)

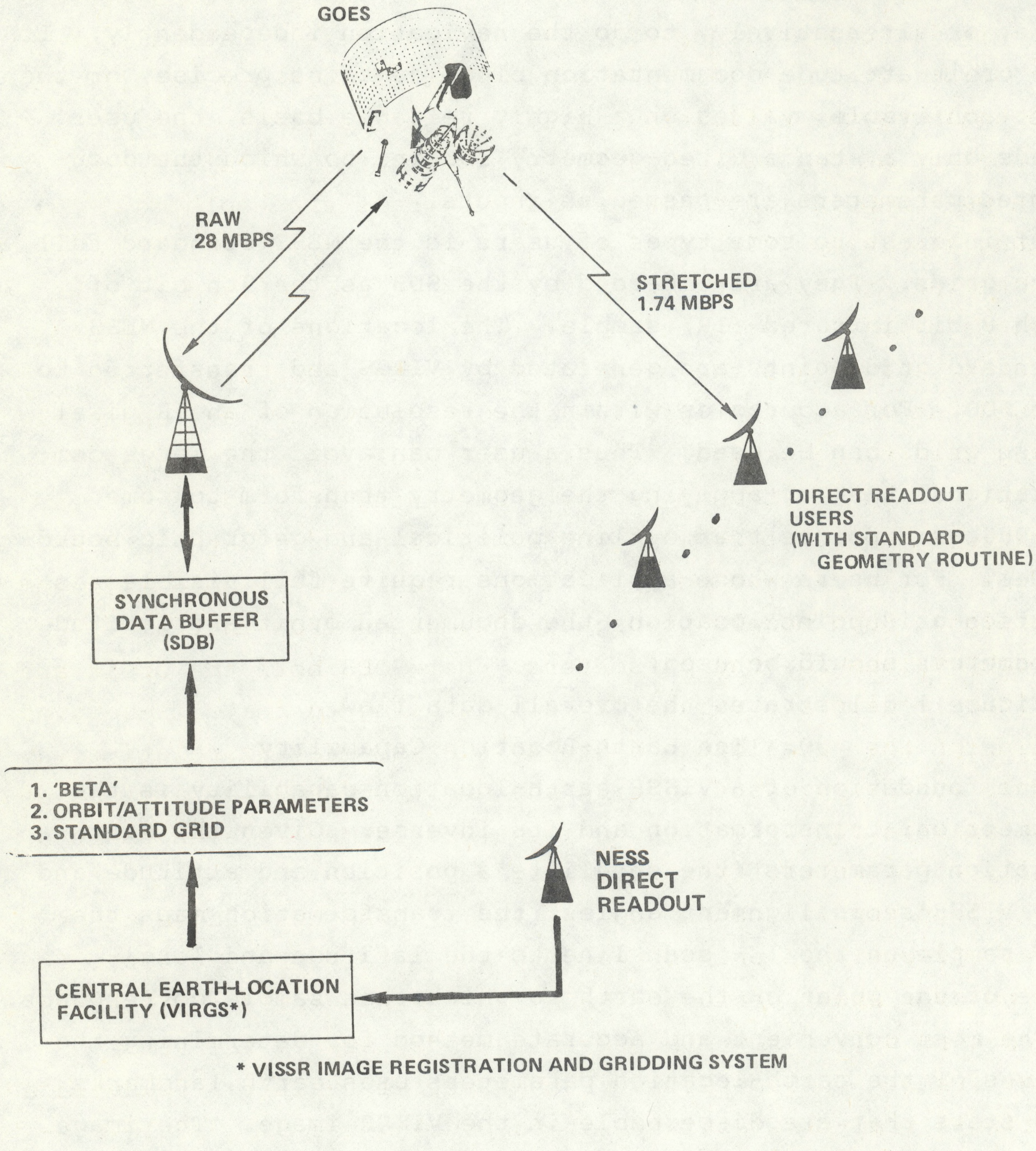

Figure 1. Overall data flow for earth locating VISSR data

of one or more recognizable landmarks (measured during a time interval close to the the time interval of the star measurements) provide a solution for the satellite's position as a function of time, i.e., for its orbit.

Using star measurements in the navigational process separates the satellite's attitude and misalignment determination from the orbit determination. This disjointness greatly simplifies, and enhances the efficiency of, the orbit/attitude determination process. These advantages are maintained even when more sophisticated mathematical models are employed. This method also simplifies the structure of the software and streamlines the process so that the cpu time required for the orbit/attitude computation is negligible. Finally, the use of stars eliminates the need for the expensive operation of ground stations that provide ranging data which are currently being used to determine the orientation of the orbital plane.

Implicit in the technique for determining attitude from star positions and orbit from landmark positions, are mathematical models of the dynamic behavior (or motion) of the spin axis and position. Such models propagate the motion from initial (epoch) values for the parameters. For GOES both the orbit and attitude models are relatively simple owing to the well-behaved motion of the satellite and its spin axis. Having relatively simple models suggests that if the orbit/attitude determination methods were cleverly designed, they could be implemented as a small, efficient software package.

The less sophisticated models, i.e., those that account for fewer effects, can only be used to accurately predict satellite motion over short periods. Real-time data users need accurate parameters only a few minutes in advance, say in the prior image frame. The operators of the central facility, to avoid a continual real-time activity of supporting the SDB, would prefer a once-per-day cycle. This requires at least a 24 hour predictive capability. Being able to predict over several days or a week has some potential advantages in procedural efficiency. It would support such things as maneuver planning and batch processing of the grid point locations during off hours. In brief, the prediction capability should be a minimum of one day, and preferably several days.

Since maneuvers of the satellite occur frequently, and inevitably cause some degree of discontinuity in the earth-location process, an extremely important aspect of the earth-location capability is quick recovery of full accuracy following a maneuver. Predictions of the post-maneuver orbit and attitude are done but cannot be expected to be highly accurate. The modeling of maneuvers is not precise and many maneuvers are not conducted as planned. Using the "stars and landmarks" navigation technique immediately following a maneuver to make successively improved estimates of the orbit and attitude allows recovery of full accuracy within a few hours. The amount of time required depends on the time-of-day that the maneuvers were performed, relative to the time when the stars and landmarks are visible. Orbit/attitude determination methods should enable the best possible estimates of parameters with a minimal set of observations.

Finally, the central facility must generate and transfer the necessary parameters to the SDB. These parameters, when their values are precise, allow the SDB to center the scan lines; and by using these parameters, users can perform accurate geometric transforms with standard geometry routines.

III. MATHEMATICAL ALGORITHMS AND MODELS FOR THE EARTH LOCATION PROCESS

A. Preview

The algorithms for navigating geosynchronous satellite images will now be described. These algorithms use the positions of recognizable stars and earth-based landmarks measured in VISSR image frames to determine the attitude, the misalignment angles and the orbit parameters. The set of algorithms is divided into three parts: (1) the transformation of image coordinates to earth coordinates, and the corresponding inverse transformation, using values for the attitude, misalignment, and orbit parameters, (2) the determination of attitude and misalignment parameters from measured correspondences for stars between inertial pointing vectors and image frame positions, and (3) the determination of orbit parameters from the measured image positions of identifiable earth-based landmarks and attitude and misalignment parameters.

B. Geometric Transformations Between Image and Earth Coordinates

The approach for transforming image coordinates to earth coordinates is now outlined. The details of the mathematical formulation of the transformation follow shortly.

The first step is to use the attitude and misalignment parameters to find the inertial pointing direction of the spin scan camera (e.g., the VISSR). The inertial pointing must be found as a function of line number, element number and image start time. Next, the position of the satellite as a function of time is found. For given line and element numbers the satellites orbital position and the spin scan camera's inertial pointing direction are known, and it is a straight forward procedure to determine the point on the earth's surface being observed by the satellite's camera. This is done by projecting a ray in the inertial pointing direction from the satellite position and determining the point where it would intersect an oblate spheriod (fig. 2).

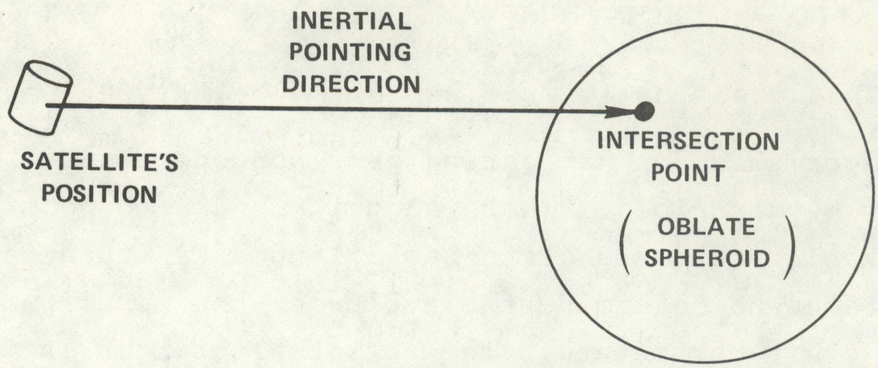

Figure 2.

The inverse problem, i.e., determining the line and element number of the sample which observes a particular point on the earth's surface in a given image, is more complicated. The complications arise from not knowing the time that the point in question is being observed. Knowing the time is essential since the inertial pointing direction of the vector from the satellite to a particular earth location varies as a function of time due to orbital motion of the satellite and the rotation of the earth with respect to inertial space. Since this time is not known, a guess must be made. Using this "first guess" one normalizes the inertial vector from the satellite to the point on the earth's surface into an inertial pointing vector and then computes the line and element number at which the pointing direction of the spin scan camera was coincident with this pointing vector. In general, the time at which this particular sample was taken is different from the first guess. The new time is taken and this procedure is repeated iteratively until two consecutive times agree to within the time between two successive scan line sweeps.

The discussion shows that three basic computations must be made in order to transform image coordinates to earth coordinates and vice versa. These are: (1) the determination of the inertial pointing direction of the spin scan camera as a function of line number, element number, picture time, beta count, and misalignment and attitude parameters, (2) the determination of the intersection between a ray and the surface of an oblate sphere, and (3) the determination of the line and element number at which the axis of view of the spin scan camera is parallel to a specific inertial pointing vector for a given set of beta counts and misalignment and attitude parameters.

The reader should be aware that "spin scan camera" is substituted for "VISSR" to connote the generality of the formulation. Also the "pointing direction" of a spin scan camera is the axis of the optical field of view of the camera or, for brevity, the "axis of view."

Figure 3 illustrates the elements of the basic geometry.

1. Determination of Pointing Direction of the Spin Scan Camera

The pointing direction of the spin scan camera is found as a function of time, beta counts, and line and element numbers. To determine this one must have an accurate attitude state, i.e., the attitude itself and the precession of the attitude. One must also know the parameters that determine the state of misalignment between the satellite's body axis and its actual spin axis. Those parameters establish the stepping path of the spin scan camera as a function of line number.

Nominally the spin axis and the body axis of the satellite are expected to coincide. However, in general, some small misalignments are observed. The misalignment causing a bias in the line positions where identifiable stars and earth-based landmarks appear in the image is designated as pitch and parameterized with the angle ζ . The misalignment that causes a bias in the element direction is designated as roll and parameterized with the angle ρ and, finally, the misalignment causing skew in the element direction as a function of line number is designated as yaw and parameterized with the angle η .

The pointing position at the start of each scan line is referenced to the sun pulse detection for that scan line by the VISSR to sun pulse detector sweep angle, γ , the current beta count, β , and whatever element bias and skew that are present. It is natural to study the position of the spin scan camera as a function of scan line number at those instances when the sun pulse is detected. Two satellite centered coordinate systems are created for this study. The first coordinate system has

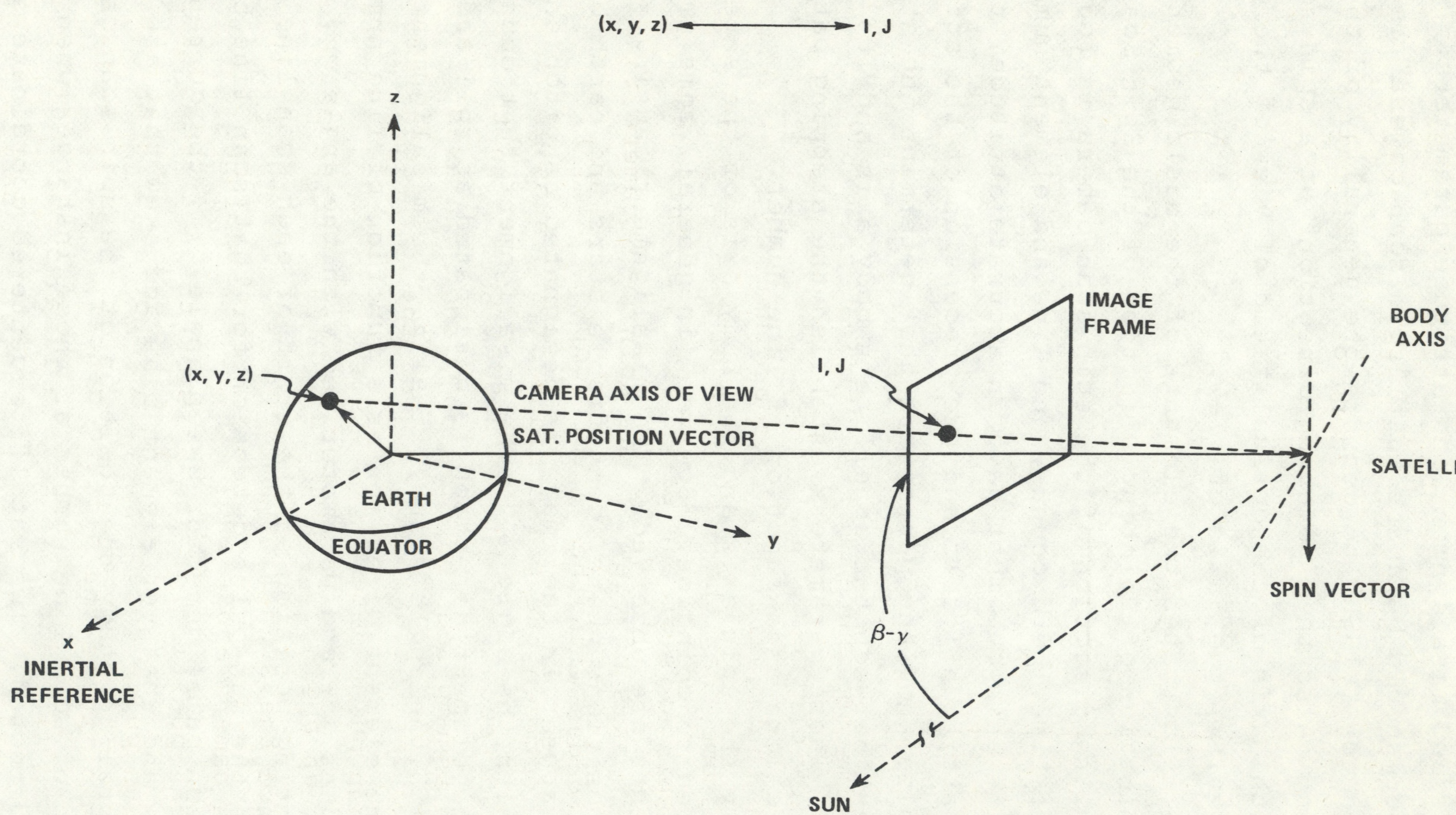

Figure 3.--Geometric transformation: (x,y,z) ⟷ I, J

its z-axis pointing opposite the spin axis and its x-axis pointing at the projection of the sun in the spin plane--the plane perpendicular to the spin axis of the satellite and which passes through the center of the satellite (fig. 4). The y-axis is added to form a right-handed, orthogonal coordinate system.

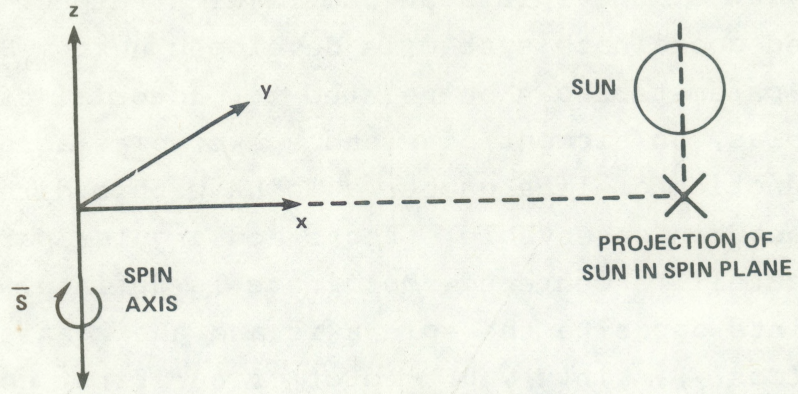

Figure 4

At the time the sun pulse is detected, the projection of the axis of view of the sun pulse detector in the spin plane coincides with the x-axis of this coordinate system. Initially, one may assume that the body axis of the satellite coincides with the spin axis of the satellite. In this case the projection of the axis of view of the spin scan camera into the spin plane coincides with the vector $(\cos(\gamma), \sin(\gamma), 0)$ where γ is the angle between the projections of the axes of view of the sun pulse detector and the spin scan camera in the spin plane. Under the assumption that the body and spin axes of the satellite coincide, the axis of view of the spin scan camera, at the time of sun pulse detection, projects onto the x-y plane (the spin plane) at an angle γ from the x-z plane of the coordinate system. Since the scan line number specifies the position of the spin scan camera above and below the x-y plane, one can now express the position of the axis of the spin scan camera at the time of sun pulse detection as a function of scan line number. First set c_ℓ to be the picture center line, r_ℓ to be the radians per line and ℓ to be the line number. The pointing direction (axis of view) of the spin scan camera is given by the vector:

$$\begin{pmatrix} \cos(\gamma).\cos((c_\ell-\ell).r_\ell) \\ \sin(\gamma).\cos((c_\ell-\ell).r_\ell) \\ \sin((c_\ell-1).r_\ell) \end{pmatrix} \qquad (2.1)$$

Since the body axis does not, in general, coincide with the spin axis of the satellite, a basis for describing the motion of the axis of view of the spin scan camera with respect to the above established coordinate system is developed next. Specific misalignment parameters can be related to pictorial effects such as a line bias, an element bias and a skew bias across the elements as a function of line number. To have this direct correspondence between observable effects and angular parameters a second satellite centered coordinate is defined. The z-axis again points opposite the spin axis and the x-axis coincides with the $(\cos(\gamma), \sin(\gamma), 0)$ vector of our first coordinate system. Hence, if the body axis does in fact coincide with the spin axis of the satellite, the x-axis of our second coordinate system coincides with the axis of view of the spin scan camera at the center line position when the sun pulse is detected. This coordinate system is arrived at from the first coordinate system by rotating the x-axis around the z-axis towards the y-axis by the angle γ.

Within this coordinate system the pointing direction of the spin scan camera as a function of scan line number at the time of sun pulse detection is as follows:

$$\begin{pmatrix} \cos((c_\ell-\ell).r_\ell) \\ 0 \\ \sin((c_\ell-\ell).r_\ell) \end{pmatrix} \qquad (2.2)$$

A perturbation matrix M enables one to describe the pointing direction (axis of view) of the spin scan camera as a function of scan line number for the case of misalignment between the spin axis and body axis of the satellite. The misalignments are so small that they can be broken down into three essential degrees of freedom: (1) rotation around the y-axis of the z-axis towards the x-axis being positive "pitch" (it makes the earth appear higher in the image), (2) rotation around the x-axis of the z-axis towards the y-axis being positive "yaw" (it skews the picture making the top of the earth appear shifted to the right with the bottom of the earth appearing shifted to the left in the image) and (3) rotation around the z-axis of the y-axis towards the x-axis being positive "roll" (it makes the earth appear to move towards the left in the image). The order in which one corrects for these misalignment angles is really arbitrary. However, one ordering must be selected to be definitive. Furthermore, because the misalignment angles are so small, the effect of applying these several misalignment effects in the wrong order would be secondary and undetectable. The misalignment matrix is defined as:

$$M = \begin{pmatrix} \cos(\rho) & \sin(\rho) & 0 \\ -\sin(\rho) & \cos(\rho) & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\eta) & \sin(\eta) \\ 0 & -\sin(\eta) & \cos(\eta) \end{pmatrix} \cdot \begin{pmatrix} \cos(\zeta) & 0 & \sin(\zeta) \\ 0 & 1 & 0 \\ -\sin(\zeta) & 0 & \cos(\zeta) \end{pmatrix} \quad (2.3)$$

where ζ is pitch, η is skew and ρ is roll misalignments respectively. Matrix M, follows the conventions used in Mottershead and Phillips (1).

The pointing direction of the spin scan camera as a function of scan line number accounting for misalignment is as follows:

$$M \cdot \begin{pmatrix} \cos((c_\ell - \ell) \cdot r_\ell) \\ 0 \\ \sin((c_\ell - \ell) \cdot r_\ell) \end{pmatrix} \quad (2.3a)$$

The pointing direction of the spin scan camera as a function of scan line number and accounting for misalignment in the first satellite centered coordinate system is as follows:

$$\begin{pmatrix} \cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{pmatrix} \cdot M \cdot \begin{pmatrix} \cos((c_\ell - \ell) \cdot r_\ell) \\ 0 \\ \sin((c_\ell - \ell) \cdot r_\ell) \end{pmatrix} \quad (2.4)$$

A way of envisioning these effects in the acquired image data is now presented. Superimpose a spherical latitude-longitude grid around the spacecraft with the equatorial plane coinciding with the spin plane of the spacecraft. Portrayed is the scanned portion of this grid, +10° to - 10° (actually a little bit more), on paper with a grid of horizontal and vertical lines; the horizontal lines represent constant latitudes corresponding to constant line numbers and the vertical lines represent constant longitude positions. A nominal rectangular shaped object within this framework is represented in figure 5.

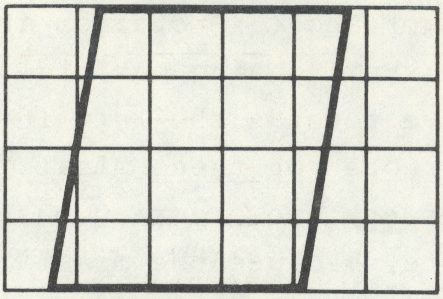

Figure 5

The skew from right to left with increasing line number is caused by decreasing beta counts which compensate for an apparent motion of the earth from right to left due to orbital motion. With similar diagrams one can portray the effects of the pitch, roll, and yaw misalignment parameters. This is done by plotting their effects on the motion of the spin scan camera as it steps. The location of the sun pulse detector is designated with an * and a trace is made of the path of the motion of the axis of view of the spin scan camera as the camera steps from the top to the bottom. The four plots in figures 6 and 7 are simplifications for a nonspinning spacecraft or, if the reader prefers, for the successive cases where the sun pulse detector points in the same inertial direction. The path of the axis of view of the spin scan camera is presented as a broken line. For non-zero roll and yaw misalignment angles the path of the nominal motion of the camera is presented with a solid line for comparison.

Note that a positive pitch has the camera pointing lower and hence makes the earth appear high in an image frame. Positive yaw has the camera pointing further to the left at the top of the frame. Positive roll moves the camera over to the right and hence makes the earth appear to be further to the left. The conventions presented here are not consistent for all

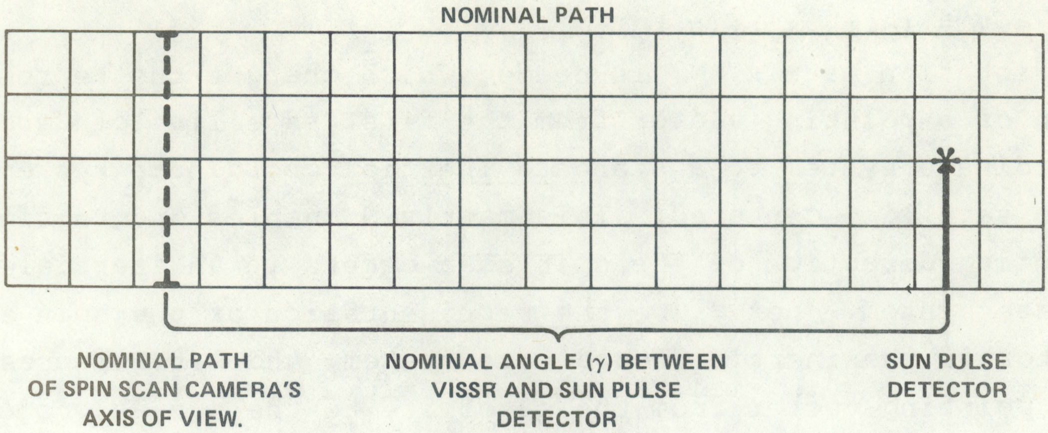

Figure 6

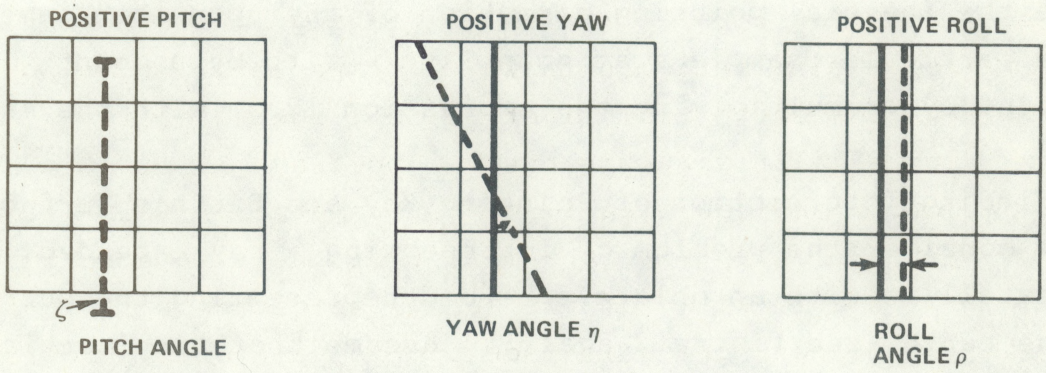

Figure 7

pieces of software used to navigate geostationary satellite data.

The model is applied to find the pointing direction of the spin scan camera in an inertial coordinate system from the inertial coordinates of the spin axis vector and the pointing vector from the satellite to the sun, the line number, ℓ , the sample number, s, and the beta count, β. The constants r_s, t_β, and s_r respectively equal the radians per sample, the time per beta count and the spin rate of the satellite in radians per time. When sample s is viewed by the spin scan camera at scan line ℓ with beta count β , the pointing direction of the spin scan camera in the second satellite centered coordinate system is given by

$$\begin{pmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{pmatrix} . M . \begin{pmatrix} \cos((c_\ell - \ell).r_\ell) \\ 0 \\ \sin((c_\ell - \ell).r_\ell) \end{pmatrix} \qquad (2.5)$$

where $\alpha = (\beta . t_\beta . s_r) + (s . r_s) - \gamma$.

A rotation matrix, N, is needed which changes the representation of a pointing vector from the first satellite centered coordinate system to a standard inertial coordinate system, also satellite centered. This matrix N enables one to find the pointing direction of the spin scan camera in an inertial coordinate system. Let S_p be the representation of the spin axis vector in the inertial coordinate system, and let P represent the pointing vector from the satellite to the sun as projected into the spin plane. Then

$$N = (P, PxS_p, -S_p) \qquad (2.6)$$

Hence the inertial pointing direction of the spin scan camera for a particular sample s at scan line ℓ with beta count β can be found by premultiplying the expression (2.5) with the matrix N.

2. Finding Intersection of Pointing Ray and Earth's Surface

Now consider the problem of intersecting a ray extended from the satellite with an oblate spheroid representing the surface of the earth (see figure 2 again). Assume that the inertial coordinates of the satellite position vector are (u, v, w) and the pointing vector coinciding with the axis of view of the spin scan camera for that particular line number and sample number is (d, e, f). The earth's surface is represented by the equation

$$(x^2+y^2)/a^2+z^2/b^2 = 1$$

where a is the equatorial radius of the earth and b is the polar radius. One intersects the ray (u+sd, v+se, w+sf) with the earth surface by substituting u+sd, v+se and w+sf respectively for x, y and z in the equation for the earth's surface and solving for s: solve the equation

$$((u+sd)^2+(v+se)^2)/a^2 + (w+sf)^2/b^2=1$$

for s. This equation is quadratic in s and, if the ray does in fact intersect this oblate surface, then the radical of the quadratic is nonnegative. In the case of the radical being positive, the ray intersects the oblate surface at two points, the one closer to the satellite is visible to the satellite and the point away from the satellite is on the opposite side of the earth.

This completes the discussion of transforming image coordinates to earth coordinates. For the inverse transform, an additional technique is needed.

3. Determining the Line and Element of an Inertial Vector

The problem of converting a satellite centered inertial coordinate vector into line and element counts for a particular satellite image is now considered. Such vectors occur naturally when one is converting earth coordinates to image coordinates. The vector in this case is the normalization of the vector extending from the satellite's position to the earth's surface (see figure 2).

The line and element numbers at which the axis of the spin scan camera is parallel to this vector is determined in a two step procedure. First the components of a vector, which is parallel to this vector, and has its base at the origin of the first satellite centered coordinate system, are determined. Next the line and element numbers at which the axis of the spin scan camera coincides with this second vector are determined using (2.5) and appropriate arcsine and arctangent formulae. The details for this transformation are given in Appendix A.

C. Finding the Misalignment Parameters and Attitude

To transform image coordinates to earth coordinates, and vice versa, the attitude state and the misalignment parameters of the spacecraft are needed. The attitude state can be computed in three ways: (1) attitude determination alone, (2) attitude determination along with determination of the pitch misalignment parameter and (3) attitude state along with a set of precession parameters. All these ways are useful in an operational environment. One may wish to compute only the attitude if the data base is not large enough to compute more parameters. For the same reason one may be interested in computing only the attitude and pitch misalignment parameter. Computing both the attitude and precession requires that star measurements be made over several days, and this cannot be done immediately following a maneuver.

The methods used to compute the attitude, the pitch misalignment parameter, and the attitude precession could also be used to compute the roll and yaw misalignments. However, fitting a line through measured element discrepancies, as a function of line number, and using the slope of the derived line and the appropriate intercept as yaw and roll misalignment angles is a simpler, equally effective approach.* This accounts for all first order effects. The second order effects are far below the resolution of the measurements. The effect of the spin axis' nutation can be removed from the star measurements by monitoring the sun pulse documentation to find the phase and the amplitude of the sine wave effects of this nutation. The main impact of this effect is to compress and stretch the images in the vertical direction. This effect should be removed.

The data base for determining the attitude state and pitch misalignment consists of inertial pointing vectors associated with image line and element pairs at which there is an identifiable star or a recognizable earth-based landmark. In the case of an earth-based landmark one assumes that a correct set of orbital parameters are available to use in computing the inertial vector from the satellite to the earth-based landmark. In the case of a star measurement, the inertial pointing vector is computed from the right ascension and declination of the star. Here for brevity only star measurements are considered for attitude determination. There is no intention, however, to obviate the use of earth-based landmarks for attitude determination.

---

*Note: Currently on the VIRGS system, roll and yaw are computed manually. First the element discrepencies of two star measurements at different scan lines are extrapolated linearly to the element discrepancy at the center scan line of the picture. This element discrepancy is then converted into an angle and entered as roll. Next the ratio of the difference of element discrepencies to the difference in scan line is computed in terms of angular measures and entered as yaw.

Consider now the case of having several star measurements with the spin axis vector essentially unchanged during the time period in which the measurements were taken. Represent the pointing direction of the spin axis with the unit vector (a, b, c) and assume k star measurements consisting of the quadruples (δ_i, α_i, ℓ_i, s_i) with i=1,...,k where δ_i, α_i, ℓ_i, and s_i are respectively the declination, right ascension, line number, and sample number of the star measurement. Then, by setting c_ℓ and r_ℓ equal to the picture center line and radians per line, the following relationship is obtained:

$$a.\cos(\zeta_i).\cos(\alpha_i)+b.\cos(\zeta_i).\sin(\alpha_i)+c.\sin(\zeta_i)$$
$$=\cos(\pi/2+r_\ell\cdot(c_\ell-\ell_i))$$

for i=1,...,k. It may appear that at least three measurements are required to solve for the three entries of the spin axis vector, but in fact one needs only two measurements along with the constraint that $a^2+b^2+c^2=1$. Let (x_i, y_i, z_i) for i=1,...,k be the unit vectors pointing at the stars in inertial coordinates. See figure 8.

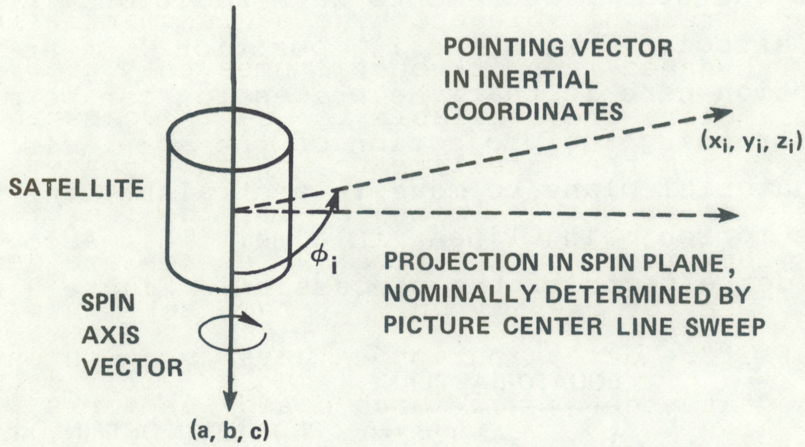

Figure 8

Mathematically the task is to minimize the following expression:

$$S(a, b, c)=\sum_{i=1}^{k}(ax_i+by_i+cz_i-\cos(\phi_i))^2 \qquad (3.1)$$

subject to the constraint $a^2+b^2+c^2=1$ where ϕ_i equals

$\pi/2+r_\ell(c_\ell-\ell_i)$ and where (a, b, c) is nearly parallel to the vector (0, 0, -1) which points opposite to the earth's spin axis to satisfy the operational constraint of centering the earth in each image. The details for minimizing the sum in (3.1) are given in Appendix B.

A nonzero line misalignment parameter ζ changes the angle between the spin axis and the axis of the spin scan camera by the amount ζ. In this case it is appropriate to find (a, b, c) and ζ to minimize:

$$S(a, b, c, \zeta) = \sum_{i=1}^{k} (ax_i+by_i+cz_i-\cos(\phi_i-\zeta))^2 \quad (3.2)$$

subject to the constraint $a^2+b^2+c^2=1$. The details of the mathematics leading to such a minimizatin are given in Appendix B.

The element and skew misalignment parameters can be ignored because they have a negligible, secondary impact on the angle between the spin axis vector and the spin scan camera.

The next problem is to find the precession of the satellite. Since this is a study of parameters varying with time, it is assumed that the star measurements have additionally a record of time, t_i associated with the i^{th} position. An underlying assumption here is that the precession can be modeled well by constraining the projection of the spin axis vector into the equatorial plane to move in a straight line with the rate of this motion being linear in time. This accounts for the first order effects of the precession. Figure 9 shows the

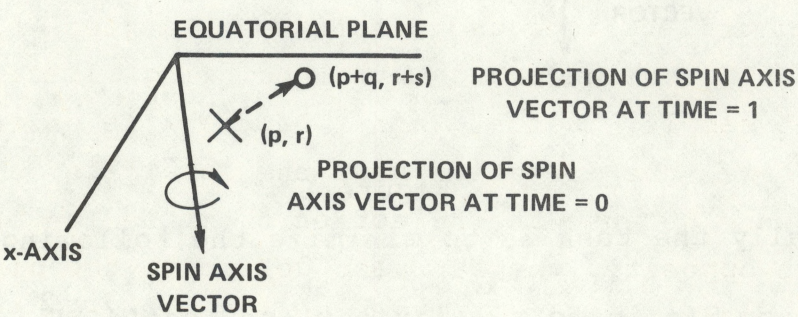

Figure 9

model: Here the (p,r) position is the projection of the spin axis vector at time equal to zero, and the (q,s) vector gives the direction of the precession movement with the length of the vector corresponding to the precession rate.

The normalization constraint is that the spin axis vector point at $(p+qt,\ r+st,\ -(1-((p+qt)^2+(r+st)^2))^{1/2})$ for all times t. Consider then the problem of minimizing the following function:

$$S(p,q,r,s) = \qquad (3.3)$$

$$\sum_{i=1}^{k} ((p+qt_i)x_i+(r+st_i)y_i-(1-((p+qt_i)^2+(r+st_i)^2))^{1/2} z_i-\cos(\phi_i))^2.$$

The mathematics of this minimization are similar to that presented in Appendix B. If one so desires, the pitch misalignment can be included in these calculations with the main effect being that a five by five matrix instead of a four by four matrix must be inverted in the minimization process.

At least two days of measurements are required to apply this scheme and three days of data would certainly provide more confidence in one's results.

D. Orbit Determination Algorithms

Next to be presented are algorithms to determine the orbit of a geosynchronous satellite by using position measurements in image condinates of earth-based landmarks and assuming one already has the satellite's attitude and misalignment parameters (i.e., from star positions). To get started, the geometry and algorithms will be illustrated briefly with diagrams and discussions that emphasize the simplicity of the approach while leaving the technical details for later formulation.

By knowing the attitude and misalignment angles of the spacecraft one can determine the inertial pointing direction for any line-element sample. Hence, when a landmark is observed on the earth's surface, the direction from the satellite to the landmark is known. Because the direction from the landmark to the satellite is opposite, one can also determine the direction from the landmark to the satellite by changing the sign of the components of the pointing vector. Hence a ray locus on which the satellite must lie extends from the landmark towards the satellite. By intersecting this ray (see fig. 10) with a sphere whose radius nominally equals the distance of a geosynchronous satellite from the earth's center and with its center coinciding with the earth's center, a vector approximating the satellite's position is found.

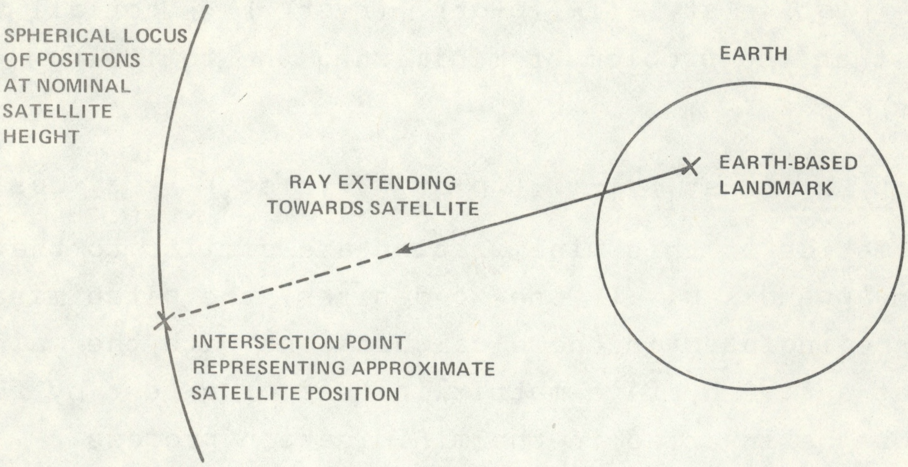

Figure 10

Thus far the height of the satellite has not been determined. Only a nominal height has been assigned so as to enable the determination the subsatellite point. If the landmark measurement being used is close to the subsatellite point, the method of determining the subpoint position by intersecting a sphere with a ray introduces almost no error. The goal is to track the motion of the subpoint in order to determine a set of Keplerian orbit parameters.

The task of finding Keplerian orbit parameters is split into two parts: (1) determining the plane in inertial space containing the orbital track and (2) approximating the angular motion of the satellite around the center of the earth as a function of time. For each landmark measurement, one is able to approximate a pointing vector in inertial coordinates which points from the center of the earth towards the satellite position. Index these pointing vectors and their associated times by i yeilding (t_i, x_i, y_i, z_i) for i equal to from one to the number of landmark measurements being considered.

To determine the orbit plane one determines a vector that is perpendicular or nearly perpendicular to all these inertial vectors. Since the satellite moves along with the earth's motion from west to east, the right-hand rule determines that the orbit plane vector points nearly parallel to the earth's spin axis. As with the attitude determination, let (a, b, c) be the inertial unit vector that is perpendicular to the orbital plane. Hence this vector must satisfy the constraints

$$a x_i + b y_i + c z_i = 0 \text{ for } i=1, \ldots, N$$

where N is the number of landmark measurements under consideration. Because of measurement errors and other imprecisions in the system it is appropriate to minimize the following sum:

$$S(a,b,c) = \sum_{i=1}^{N} (a x_i + b y_i + c z_i)^2 \qquad (4.1)$$

The mathematical form of this problem is identical to that presented with equation (3.1). The methods used to minimize this expression are similar to methods used to minimize (3.1). Referred to Appendix B for further information.

Figure 11 illustrates the orbit plane perpendicular that is determined by these calculations.

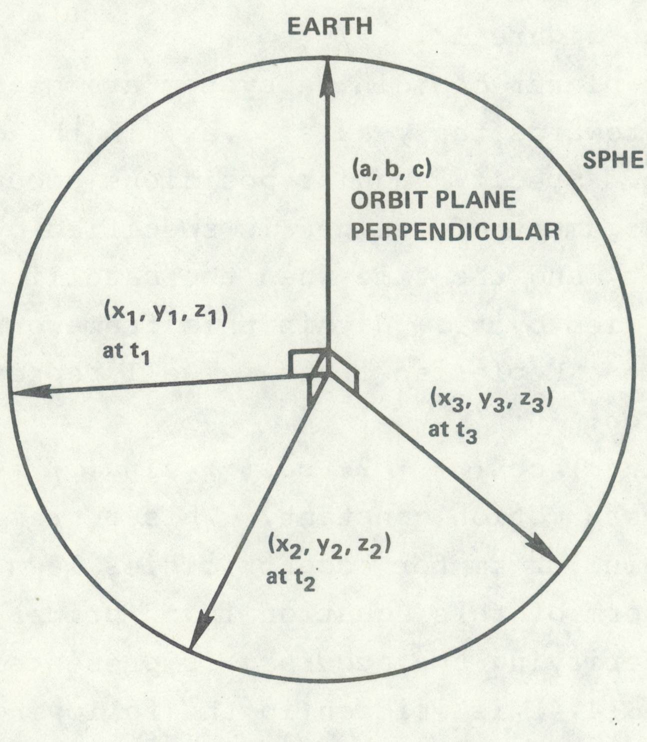

Figure 11

The orbit plane perpendicular (a,b,c) determines the plane in which the satellite moves. However, to obtain the more familiar Keplerian orbit parameters, one must manipulate the components of this unit vector to obtain the ascending node and inclination of the orbit. By definition the pointing vector (a,b,c) is parallel to the vector

$$(\sin(\Omega)\cdot\sin(i),\ -\sin(i)\cdot\cos(\Omega),\ \cos(i)) \qquad (4.2)$$

where Ω is the ascending node and i is the inclination of the orbit. Since the inclination is always taken to be positive, the ascending node and the inclination can be determined as follows (in Fortran):

$$\Omega = \mathrm{ATAN2}(a,\ -b) \text{ and} \qquad (4.3)$$
$$i = \mathrm{ASIN}(\mathrm{SQRT}(a^2+b^2)).$$

This completes the discussion of determining the orbit plane for the satellite.

In order to study the angular motion of the satellite within its orbit plane, a planar coordinate system is set up in the orbital plane. The x-axis points from the center of the earth to the ascending node and the y-axis, also lying in the orbital plane, is 90° beyond the x-axis in the direction of satellite motion. Refer to figure 12.

Angles in this planar coordinate system are measured as going from the x-axis towards the y-axis, i.e., in the direction of satellite motion. Specify angular positions occuring at time t_i as α_i. The argument of perigee is specified by ω, time is specified by t, and the time when the satellite is at perigee is specified by t_p. Within this framework the motion of a satellite in a Keplerian orbit is well represented by the following equation:

$$\alpha - \omega - 2\epsilon\cdot\sin(\alpha)\cdot\cos(\omega) + 2\epsilon\cdot\cos(\alpha)\cdot\sin(\omega) = n\cdot(t-t_p), \qquad (4.4)$$

where n is the mean motion constant. This representation is accurate to within .12 km for eccentricities less than 10^{-3}. Since only the form of this equation is of primary interest, the derivation verifying the accuracy is presented in Appendix C. Normally eq. (4.4) is written in the following equivalent form

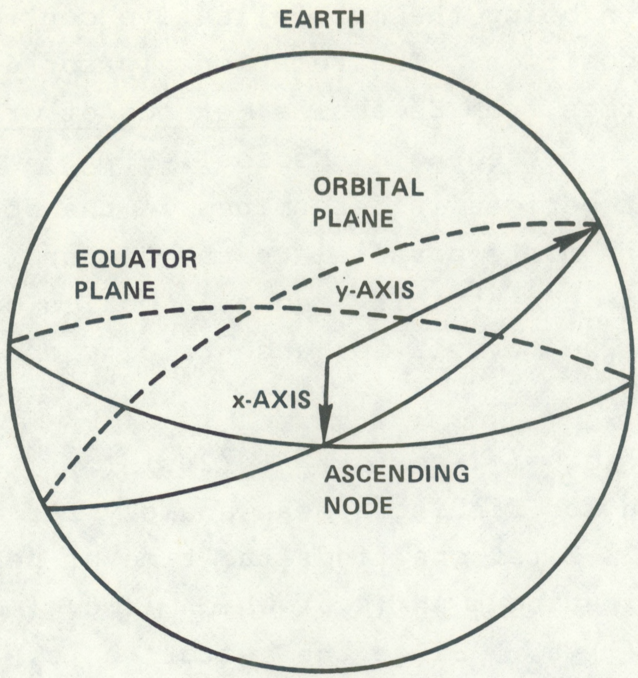

Figure 12

$$t_i = c_1 + c_2 \cdot \alpha_i + c_3 \sin(\alpha_i) + c_4 \cos(\alpha_i) \tag{4.5}$$

where the index i indicates the times and angular positions corresponding to the landmarks and $c_1 = t_p - \omega/n$, $c_2 = 1/n$, $c_3 = -2\epsilon \cdot \cos(\omega)/n$, and $c_4 = 2\epsilon \cdot \sin(\omega)/n$. It is vital that the constants $\{c\}$ be related directly to Keplerian orbit parameters so that the reason for determining these constants becomes clear.

Observe the following relations:

$$n = 1/c_2 \;, \tag{4.6}$$

where n is the mean motion constant;

$$a = r \cdot k^{2/3}/n^{2/3} \;, \tag{4.7}$$

where a is the semimajor axis, r is the radius of the earth, and k is the gravitational constant of the earth;

$$\epsilon = (c_3^{\;2} + c_4^{\;2})^{1/2}/(2c_2) \text{ and} \tag{4.8}$$
$$\omega = \text{ATAN2}(c_4, -c_3) \;,$$

where ϵ and ω are the eccentricity and argument of perigee, respectively. The inverse relationship for c_2, c_3 and c_4 are given later in (4.13).

The inverse relationship between the mean motion constant n and c_2 is due to n being the multiplicative constant relating time to angular position. The relationship for determining the semimajor axis comes from Escabol's Methods of Orbit Determination, henceforth referred to as Escabol.

The argument of perigee is determined in the above manner because $(c_4, -c_3)$ points parallel to the vector $(\sin(\omega), \cos(\omega))$ due to the eccentricity and mean motion constant both being positive. The formula for eccentricity comes from the observation that

$$(c_3^2+c_4^2)/c_2 = (-2\epsilon\cos(\omega))^2 + (2\epsilon\sin(\omega))^2 = 4\epsilon^2. \quad (4.9)$$

The computation for finding a mean anamoly for a given epoch is more involved. First one finds the time of perifocal passage by substituting ω in (4.5) which is $c_1 + c_2\omega + c_3\sin(\omega) + c_4\cos(\omega) = t_p$. Since the vector (c_3, c_4) is parallel to $(-\cos(\omega), \sin(\omega))$ and hence perpendicular to $(\sin(\omega), \cos(\omega))$, the last two terms cancel, giving $t_p = c_1 + c_2\omega$. Now, by definition, the mean anamoly at an epoch time t_e equals $n\cdot(t_e - t_p)$. Hence, adopting the programming symbol MEANOM for the mean anamoly,

$$\text{MEANOM} = n(t_e - c_1 - c_2\cdot\omega) = (t_e - c_1)/c_2 - \omega. \quad (4.10)$$

This completes the process of transforming the $\{c\}$ into orbital elements.

Now consider the problem of finding the $\{c\}$ from a set of measurements (α_i, t_i). For this purpose linear regression is applied. The following expression is minimized:

$$S = \sum_{i=1}^{N} (c_1 + c_2\alpha_i + c_3\sin(\alpha_i) + c_4\cos(\alpha_i) - t_i)^2. \quad (4.11)$$

To solve for the $\{c\}$, set the following four partial derivatives to zero:

$$\frac{\partial S}{\partial c_1}(c_1, c_2, c_3, c_4) = 0,$$

$$\frac{\partial S}{\partial c_2}(c_1, c_2, c_3, c_4) = 0,$$

$$\frac{\partial S}{\partial c_3}(c_1, c_2, c_3, c_4) = 0 \text{ and}$$

$$\frac{\partial S}{\partial c_4}(c_1, c_2, c_3, c_4) = 0,$$

This results in a matrix equation for the c_i's which is now presented with the following abuse of notation: SM stands for sum over i for i=1,.., N, A for the angles α_i, SN for $\sin(\alpha_i)$, CS for $\cos(\alpha_i)$, T for the time t_i, and 2 means that the immediately proceeding term is squared.

$$\begin{pmatrix} N & SMA & SMSN & SMCS \\ SMA & SMA2 & SMASN & SMACS \\ SMSN & SMASN & SMSN2 & SMSNCS \\ SMCS & SMACS & SMSNCS & SMCS2 \end{pmatrix} \cdot \begin{pmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{pmatrix} = \begin{pmatrix} SMT \\ SMAT \\ SMSNT \\ SMCST \end{pmatrix} \quad (4.12)$$

The details of the specific mathematical technique used to invert this matrix are given in Appendix D. Notice here that the inverse of the matrix is ill-conditioned over short time intervals. This results from the inability to resolve the difference between a straight line and part of a sine wave over a short arc. For this reason, to achieve results comparable in accuracy to one's measurements, one should make landmark measurements distributed over a time interval at least 18 hours long. A convenient way to circumvent making such a long stretch of landmark measurements is to make 8-hour stretches of landmark measurements 24 hours apart.

In daily operations it is often the case that one uses the semimajor axis computed from yesterday's and today's data but only uses today's data to compute the three along-track orbit parameters: mean anamoly, eccentricity, and argument of perigee. This is a two-step procedure. First compute the constant c_2 by inverting the method used to compute the semi-major axis from the constant c_2. Then compute the remaining c_i's by applying linear regression in the same manner used to find all four c_i's. In this case the resultant three-by-three matrix is inverted by applying Cramer's Rule. The along-track orbit parameters are then found in the same manner as before. This particular approach is potentially valuable for the time period immediately following orbit maneuvers if one can obtain an estimate of the orbit's semimajor axis from energy considerations.

The estimate for the orbit's mean anamoly can be refined when there are good estimates of the other orbit parameters by requesting that only one further parameter be generated. In this case c_2, c_3 and c_4 are generated by the following formulae:

$$c_2 = k \cdot (r/a)^{3/2}$$
$$c_3 = -2 \cdot \epsilon \cdot c_2 \cos(\omega) \text{ and} \tag{4.13}$$
$$c_4 = 2 \cdot \epsilon \cdot c_2 \sin(\omega).$$

Finally, if only two along-track orbital elements are wanted, linear regression is used to solve for the parameters c_1 and c_2, and these parameters are transformed into a semimajor axis and mean anamoly while at the same time making the eccentricity and argument of perigee equal to zero. There are also ways of computing c_1 and c_2 while retaining nonzero values for the eccentricity and argument of perigee. These are currently not implemented.

This ends the discussion of orbit determination.

IV. DESCRIPTION OF IMPLEMENTATION

A. The Orbit and Attitude Determination Software

The functions of the software implementation of the star navigation can be split into three categories: (1) the scheduled ingest of images containing either stars or earth-based landmarks and the measurements of the image locations of these features, (2) the maintenance of navigational files containing the measurements and parameters necessary for navigation and (3) the execution of software that determines the necessary orbit, attitude and misalignment parameters. Further details of the software execution for tasks (1) and (2) are omitted because they are not germane to the purpose of this paper. Instead the mathematical aspects of measurement errors in task (2) affecting the scheduling in task (1) are discussed. These aspects affect the scheduling of stars for the determination of the attitude state and misalignment parameters and the scheduling of earth based landmarks for the determination of orbital state.

Star measurements should be distributed with respect to time and the right ascension of the stars' coordinates to provide an adequate data base for generating the attitude parameters, the precession parameters and the misalignment parameters. Similarly landmark measurements should be distributed appropriately with respect to time to provide an adequate data base for generating orbit parameters. Such distributions of measurements are essential to prevent an ill-conditioned inverse of a matrix from blowing up the impact of measurement errors. It is important to know what distribution of particular measurements are necessary for the accurate determination of specific subsets of these parameters. Such knowledge enables one to establish a schedule for ingesting stars and earth based landmarks.

For attitude determination alone, assuming the precession rate is small, at least two star measurements are necessary with the right ascension separation, mod (180°), being 15° or more. Mathematically the ideal case is to have the right ascensions separated by 90°. For attitude and pitch alignment determination, three star measurements are necessary, two with right ascension separation, mod (180°), of 15° or more and a different pair (i.e., a member of the first pair matched up with the third star measurement) with right ascension measurement separation, mod (360°), of 15° or more. For determining the attitude and precession parameters, four star measurements consisting of two disjoint pairs with each pair of stars having right ascension separation, mod(180°), of 15° or more and with time separation from the first pair to the second pair of at least 18 hours. In this case the same stars can occur in each pair. For determining the roll and yaw misalignment parameters, two star measurements are needed with line separation of 5,000 to 10,000 lines.

Questions have been raised about the viability of routinely distinguishing stars from background noise. Such doubts need to be examined because of the central role star measurements play in a stand-alone navigational system. Since star measurements are not currently being used to determine the satellite's attitude, an auxillary system of ground stations and a second computer has to be maintained.

It is vitally important to consider each star measurement as a single point in a consistent data set. Regarding each star measurement as a separate entity makes the measurement task more difficult and less fruitful. Simply stated, at a single time the spin axis vector points in only one direction. Hence, if a set of star measurements yields large residuals after a spin axis determination, at least one measurement is wrong. The measurement(s) whose residuals are noticely distinct from the residuals of other measuremenets should be deleted. This idea can also be used predictively since the change of attitude over a one day period is small. Hence the measured position of a star should be close to its predicted position. Use of this idea in a procedual manner should greatly reduce problems of distinguishing stars from noise.

In addition, a star often occurs more than once on a given day. By adding the image frames containing a recurrent star, with an appropriate element lag displacement, the signal to noise ratio of the star's occurrence will be significantly enhanced. Also, the full precision (6 bit) visible image should be used in scanning for stars.

For the determination of the orbit's inclination and ascending node, at least two landmarks are necessary with time separation of 4 to 6 hours. For the determination of the orbit's mean anamoly once the other five orbit parameters are known, only one landmark measurement is needed. For the determination of the orbit's mean anamoly, eccentricity, and argument of perigee once the other three orbital parameters are known, at least three landmark measurements distributed over a six to eight hour interval are necessary. In order to determine all the orbital parameters at once, at least four landmark measurements distributed over a sixteen to eighteen hour interval are necessary.

The computer memory space necessary for both the attitude and orbit determination programs is less than 16,000 words and their execution times are below 1 second. Both programs are easily transferable to other computers with Fortran compilers. To achieve such a transfer one has to provide files for a navigational data base, routines to monitor and maintain this data base and interface programs between the navigational data base and the orbit and attitude determination software.

Fortran listings of the orbit and attitude determination routines, UPGORB and FINDAT, are provided in Appendix E.

B. Standard Geometry Routine for Users

Direct readout users have access to the orbit, attitude, and misalignment parameters. These parameters as determined by VIRGS are updated in the stretched VISSR orbit/attitude block. A description of the format and definitions of this block is given in reference (6).

A geometric transformation that uses these parameters as input is available from NESS in a standard form as two Fortran routines. "IMGLOC" transforms from earth latitude and longitude to image frame line and element. "EARLOC" is the inverse routine, transforming from VISSR line and element to earth latitude and longitude. Listing of these two routines are provided in Appendix F.

These routines do not use "Beta", the angle subtended at the satellite by the sun and earth, even though it is provided in the documentation block. The SDB uses a "Beta" function that is precisely derived from the documented orbit, attitude, and misalignment parameters. Therefore, centering of scan lines on the earth disc has to be assumed by the user. This simplification's benefit for the user is that fewer parameters have to be stored. Modification of the routines to account for earth disc offsets is a small, straightforward task.

An assembly language version of "IMGLOC" was implemented on the control computer of the Central Data Distribution Facility operated by NOAA/NESS for the GOES-TAP service. Using the earth-location parameters in the VISSR data stream, the routine locates the image sectors in near real-time. The routine has been operating since June, 1979.

C. The VISSR Image Registration and Gridding System (VIRGS)

VIRGS is the central earth location facility. VIRGS, in terms of hardware and basic software, is a copy of the McIDAS II (Man-computer Interative Data Access System). McIDAS II was designed and built by the Space Sciences and Engineering Center (SSEC) of the University of Wisconsin as an upgraded version of the original McIDAS (3,4). The VIRGS was installed by SSEC at the World Weather Building in Camp Springs, Md in July 1978. The operational use of VIRGS began in May, 1979.

The orbit/attitude determination techniques and software package described in this paper were developed by SPAAM, Incorporated under contract to NESS during the past 2 years as a follow-on to work began by Dennis Phillips while at SSEC.

The VIRGS (or McIdas II) consists of a small computer (the Harris/6 with 64,000 words of memory), an on-line 40 Mbyte disc, and an interactive display console for the analyst. The important characteristic of McIdas is the elaborate set of "key-ins" the analyst has available for processing and display. The demonstrations of accurate VISSR image location were first performed on the McIdas in 1974 using the interactive key-ins to iteratively adjust orbit-attitude parameters to achieve accurate locations of visible landmarks in the images (2,5).

The VIRGS performs the central earth location process on a daily basis for each GOES in four basic steps:

i. measurement of star and landmark positions in the visible imagery (and sometimes in the infra-red imagery).

ii. determination of the misalignment angles and attitude from the star observations and the orbit from the landmark observations

iii. production and transfer (to the SDB) of the "Beta" function, earth-location parameters for the documentation block, and the locations of points in the NESS standard grid table, all for the upcoming 24-hour period.

iv. monitoring of accuracy of the earth-location throughout the day, with updating if necessary.

The primary roles of the human operators-analysts in performing this daily operation are:

1. Scheduling the ingest of image sectors containing stars and then the measurement, using a joystick controlled cursor, of the bright spot corresponding to each star.
2. Selecting cloud-free areas on earth that contain favored landmarks, scheduling the ingest of the appropriate sectors, and then measuring with the cursor the image positions of the recognizeable features.
3. Judging a sufficient set of star and landmark observations and entering, in proper sequence, the several key-ins that determine misalignment parameters, attitude, and orbit.
4. Verifying that error residuals as output by various routines are suitably small and that the various files have been updated.
5. Entering key-ins that execute routines that produce files for the SDB, quality checking these files, and

transferring the files to the SDB.

6. Routinely checking that the parameters and embedded grids that are currently on line (i.e., in the data stream) are suitably accurate, generating updates for the SDB as required.
7. Coordinating with the satellite control center, on the scheduling of maneuvers and special VISSR imaging sequences, and appropriately changing the standard operational schedule to best deal with such events.

Clearly the human plays a critical role in the VIRGS operation. While the computational tasks are fully automated as Fortran routines executable by simple key-ins, the human is required to supervise their execution, as a safeguard against erroneous measurements under normal circumstances and as a reserve of judgement against abnormal circumstances. Given the number of contingencies that exist daily, the VIRGS implementation appears to have an effective mix of human and computer talents.

An important feature of the VIRGS implementation is the closed loop formed when the VISSR data, appended with the VIRGS produced earth-location information (parameters and embedded grids), are ingested and displayed at VIRGS. Figure 13 portrays the loop. When ingesting VISSR data, VIRGS is emulating any other direct readout user and hence has the capability to monitor the performance of the system, end to end. The VIRGS operator should be the first data user to become aware of degradations in the earth-location accuracy. In most cases, the operator will be able to correct them quickly through coordination with SDB operators or satellite control center.

The issue of optimal versus minimal sets of star and landmark position measurements is discussed in section IV-A as a part of the orbit/attitude determination software descriptions. The screening of observations, based on the confidence factor resulting from any sort of difficulty in making the measurements, is an intuitive manual task for the analyst. Although most

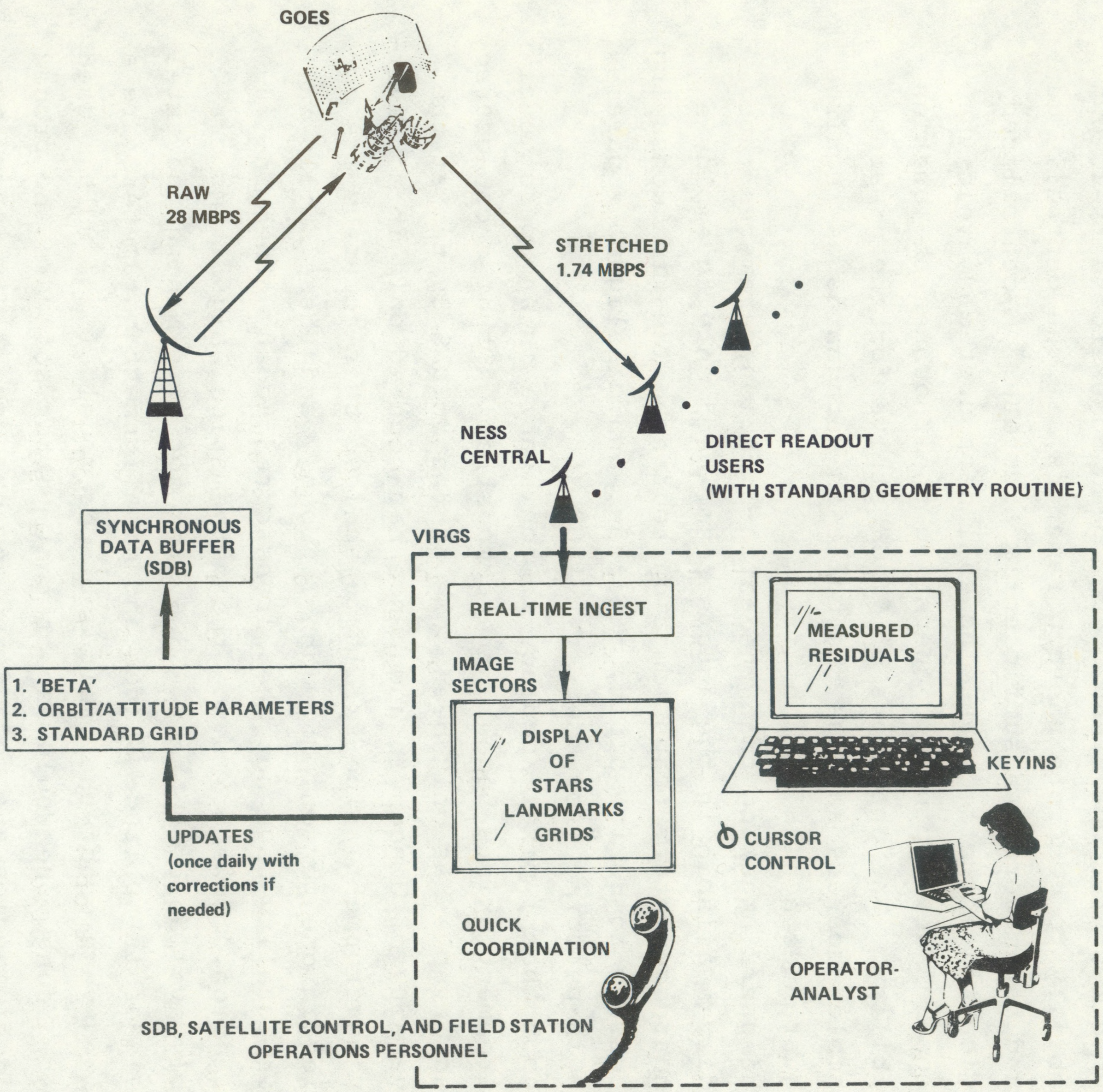

Figure 13.--Closed loop for monitoring of end to end performance

situations are clear-cut, some require the collection of additional measurements for a reliable solution.

D. Rationale for Centralized Service

Since the earth-location parameters generated by VIRGS at the central facility are appended to the stretched VISSR data, other direct readout users do not need to exercise their own capability to generate these parameters, <u>if</u> the centrally generated parameters are sufficiently accurate. A centralized service is cost effective since all data user can avoid the expense of developing and operating their own earth-location capability. The operators of VIRGS strive for the greatest possible accuracy and use the built-in safeguards as required to guarantee a high reliability.

The users, knowing that the earth-location parameters are readily available in the data stream, need only to employ a standard geometery transformation routine. The time consuming processing to derive orbit, attitude, and misalignment parameters is performed for users by VIRGS.

Doubt that 1 pixel accuracy is achievable in an operational environment exists because the demonstration that such accuracy is achievable has only been done on the McIDAS System at the University of Wisconsin and hence has occured only in a university environment. The skills to achieve this accuracy are embodied in one person, Mr. John T. Young, and these skills in their totality have not been transferred to any other installation. However, since the mathematical algorithms currently available on the VIRGS System for determining the satellite's orbit/attitude states and misalignment parameters are a refined version of the orbit/attitude model of the McIDAS used by Mr. Young, one would expect that, after software and procedual deficiencies are recognized and removed, 1 pixel accuracy will be achieved. The actual results discussed in Section IV-E, obtained on VIRGS by Larry Hambrick, lend strength to this expectation. One advantage of having these algorithms in the form of computer programs (as they are now available) is that an operator can objectively generate parameters describing satellite's orbit/attitude state without having a deep understanding of orbit/attitude relationships.

An accuracy of 1 pixel, as used herein, is defined as the root-mean-square of the earth location errors, line and element errors taken separately, distributed over the earth disc throughout a 24 hour period. One pixel is being used here as the angular distance between successive pixels in a full resolution picture. With this definition one is not guaranteed that any given point on any given frame will be located to within 1 pixel. A specific point at a given time might be in error by several pixels. However, loosening this requirement further, say to two or more pixels rms, significantly increases the possibility of larger errors. In terms of ground distance, a one pixel error spans one kilometer at the subpoint but spans two kilometers 50° from the subpoint. Finally, it is noted that, although 1 pixel accuracy rms is probably acceptable to all current users, even sharper accuracies are possible with the precision alignment capabilities inherent in the data stream.

The software currently available on the VIRGS System has been verified to approach 1 pixel accuracy. Even though routine operators are not now expected to attain this level of accuracy, it appears likely that information transfer and training will improve the routine operation. The goal is to make the routinely achieved accuracy a characteristic of the software package alone. This can be accomplished by improving the software packages to reduce the reliance on an operators's judgement and by eliminating the impact of an operator's bias through appropriate training.

E. An Error Budget and Some Actual Results

1. Error Budget

The accuracy goal, stated here for overall system performance, is one full resolution visible pixel, which is a 21 μradian angular interval at the satellite that translates into 0.8 km at the subsatellite point. The natural unit of error measure for the software, satellite, and SDB is the full resolution angular interval or pixel. All elements of the earth location process involving computations and data handling have been designed and implemented to achieve accuracies of a small fraction of a pixel. However, the end to end process has several potential error sources that must be dealt with and which limit the ultimately achieveable accuracy.

The potential error sources are exhaustively listed below.

1. Tolerances in the orbit and attitude modeling.
2. Tolerances in the geometry transformation.
3. Measurements of landmark and star positions.
4. Misassignment of the earth coordinates of the landmarks.
5. Tolerances on the satellite's sun detector.
6. Biases in the SDB scan-line interpolation and sun position referencing.
7. Timing jitter in the satellite and SDB.
8. Tolerances in the solution for the orbit, attitude and misalignment parameters.

The timing jitter of the satellite and SDB and the satellite's sun detection error might be taken as the hard limits on accuracy. There are, however, 22 time-counts per full resolution pixel. The jitter amounts to no more than a few counts (i.e., less than a quarter of a pixel). The sun detection error is the combination of random errors in the leading-edge-detection, which should be less than two or three counts, and a slowly changing offset due to changes in the sun's relative position in the sun sensor's slot-shaped field of view. The slowly changing offset can be determined frequently enough to be accounted for in the "roll" parameter to within a tolerance of a fraction of a pixel. Errors in the SDB's scan line interpolation and sun position reference combined, appear to be less than a half pixel.

The accuracy of the geometry formulation is known to be well within 1 pixel and is intended to be within 0.1 of a pixel but cannot be truly verified to that level. The orbit model and the attitude model are intended to be within 1 pixel. The operation of these models has been verified at SSEC to be within a tolerance approaching 1 pixel. An effort has been made to keep these models commensurate with 1 pixel accuracy. Nutation of the satellite's spin axis can be modelled and determined, but has not as yet been accounted for in the earth-location parameters. Nutation, if not accounted for, can give errors of two or three visible lines (peak), cycling over about 8 scan lines.

The potential accuracy of star and landmark position measurements (i.e., observed image frame coordinates) is within 1 pixel under normal conditions. Human error and variations in human judgement on features, can combine to reduce the routinely achieveable accuracy to about 1 pixel rms. The earth coordinates of landmarks can be verified to within a 1 pixel tolerance. The star coordinates are available from standard tables to an accuracy of better than 0.2 pixel (i.e., 0.5 arc seconds in earth centered celestial coordinates). The solution for orbit, attitude and misalignment parameters is susceptible only to biases in the measurements since it incorporates least-square-error regression.

When the errors discussed above are included, the resulting accuracy for a data user with a standard geometry transform routine that uses the documented parameters should be 1 pixel rms. This figure represents errors over the entire earth disc and throughout the day. The effect of spin axis nutation contributes in the worst of cases to more than a 1 visible pixel rms error in line. It's contribution to overall accuracy is however greatly reduced because of its high frequency cycle. Also the nutation dynamically deforms earth features but a human or computerized recognition process filters out much of its effect.

2. Actual Results

Figure 14 presents the results of a determination of orbit, attitude, and misalignment parameters from star and landmark measurements. The results shown are not the best obtainable and in fact are on the verge of being unsatisfactory. This set was selected for discussion here because the nature of the

SOLUTION

LANDMARK RESIDUALS

N	SSYYDDD	HHMMSS	LCODE	LINDIF	ELEDIF	SUBLAT	SUBLON
1	2479268	1500	0	-.63	.48	-200	-1352116
2	2479268	1500	301	-.07	1.90	-200	-1352116
3	2479268	11500	351	-.22	1.55	-6	-1352048
4	2479268	21500	351	-.22	.96	148	-1352023
5	2479268	21500	353	.22	-.74	148	-1352023
6	2479268	31500	353	-.77	.53	334	-1352001
7	2479268	151500	0	.13	2.39	-337	-1352232
8	2479268	161500	0	.38	.52	-509	-1352246
9	2479268	161500	301	.13	-.33	-509	-1352246
10	2479268	164500	355	-.51	3.25	-547	-1352252
11	2479268	174500	0	-.43	1.51	-645	-1352256
12	2479268	174500	301	-.65	1.80	-645	-1352256
13	2479268	201500	0	.52	1.37	-704	-1352237
14	2479268	201500	301	.55	1.46	-704	-1352237
15	2479268	221500	0	.58	.49	-510	-1352154

PREDICTION

LANDMARK RESIDUALS

N	SSYYDDD	HHMMSS	LCODE	LINDIF	ELEDIF	SUBLAT	SUBLON
1	2479269	1500	0	.27	.88	-153	-1352058
2	2479269	1500	301	1.12	2.05	-153	-1352058
3	2479269	11500	351	1.13	-.19	1	-1352031
4	2479269	21500	351	1.13	-.03	155	-1352005
5	2479269	21500	353	.49	2.32	155	-1352005
6	2479269	34500	354	1.86	2.94	429	-1351935
7	2479269	154500	0	1.89	-.16	-431	-1352225
8	2479269	171500	0	2.67	2.48	-623	-1352239
9	2479269	171500	301	3.58	1.70	-623	-1352239
10	2479269	184500	0	2.52	-.27	-716	-1352238
11	2479269	184500	301	3.78	-.60	-716	-1352238
12	2479269	201500	0	4.19	-.17	-702	-1352220
13	2479269	201500	301	2.87	-.97	-702	-1352220
14	2479269	221500	0	3.37	-1.40	-505	-1352136
15	2479269	221500	301	2.63	1.13	-505	-1352136

Figure 14. Results

various error contributions is illustrated.

The measured star and landmark positions are listed and identified by satellite number (SS, West GOES is #24), year (YY) and Julian (DDD) in the first column, SSYYDDD. The GMT of the start of the image frames in which the particular measurement where made is given as hours-minutes-seconds in the second column, HHMMSS. The LCODE given in the third column identifies the measurements as:

0 a landmark, in Baja, CA
301 a landmark, in islands of the South Pacific
351 visible star #1 (β-Orion, "Rigel")
353 visible star #3 (α-Canis Minor A, "Procyon")
354 visible star #4 (α-Aquilae, "Altair")
355 visible star #5 (γ-Orion, "Bellatrix")

The star α-Orion ("Betelgeuse") is also usually visible in the images.

The fourth and fifth columns are respectively the line and element residuals (differences) between the observed image positions and the positions specified by the solution. The units for the residuals are the full resolution visible pixel. The last two columns give the latitude and longitude, in degrees-minutes-seconds (DDDMMSS), of the satellite's subpoint.

The solution was obtained from the measurements on day 268 and predicted to day 269. The prediction is a linear extrapolation in time of the attitude and an evaluation of the orbit model with epoch parameters (for 0 GMT on day 268) at the particular times on day 269. The solution residuals are roughly 0.5 rms in line and 1.5 rms in element. The prediction residuals are roughly 2.5 rms in line and 1.5 rms in element. The measurements are spread over nearly 24 hours and geographically dispersed with landmark 0 being far above and to the right of the subpoint and landmark 301 slightly below and to the left of the subpoint. The different stars vary in right ascension by about 220° and in declination between ±8°.

The largest component of the prediction error, the "bulge" in line residuals during the period 17 to 22 hours GMT, suggests an imprecise solution for the precesion rate of the satellite's attitude. The combination of all the other error sources discussed above is seen as the 1- to 2-pixel variation that is somewhat random.

3. A Chronology of Results with VIRGS

When the VIRGS became operational in May, 1979, the orbit/attitude determination software was at an intermediate stage of its development. The level of accuracy achieved in the operation with this version of the software during the first several months of operation was sporadic, ranging from 3 pixels rms on some days to 10 pixels on other days. On an average of once per week during this "break-in" period, severely degraded performance occurred as a result of procedural and training deficiences in dealing with non-routine matters such as satellite maneuvers.

The intermediate version of the software, while still being used in the operation, was unable to adequately distinguish the orbit plane from the attitude of the spin axis using landmark and earth edge measurements. By August 1, 1979 the software had been revised by Dennis Phillips for the "star technique" described in this document. Preliminary tests of the revised software verified that the attitude and misalignment parameters could be accurately determined from the star positions alone, leaving the landmark positions for an independent orbit determination.

During the months of August, September, and October 1979, the star technique's operational viability was tested by Larry Hambrick. While having to work around the regular operation on VIRGS, image sectors containing stars were ingested from both the East and the West GOES. The star positions were measured and entered on the navigation files. The attitude and misalignment parameters were determined from the star measurements. With these parameters and the landmark measurements (i.e., from the ongoing operation), the orbit was determined. The accuracy of the predictions using these solutions were then checked. About fifty daily navigations were performed in this manner. A formal demonstration was held in September, 1979.

The following conclusions are drawn from the results of those tests:

i. A sufficient set of stars, for the star technique, are routinely available in the images.

ii. Using the star technique, VIRGS has a stand-alone orbit/attitude determination capability.

iii. Several deficiencies remain in the revised software (see next paragraph); however, accuracies better than 3 pixels rms are routinely achievable on the predictions, using the star technique.

At the time of these tests a discrepancy in "beta", apparently between VIRGS and the SDB, existed at the level of about 2 pixels. Also there was a discrepancy, internal to VIRGS, between the attitude and orbit routines as manifested in their residual outputs. The computation of the attitude's precession rates was done manually, a cumbersome task to do precisely. Work by Dennis Phillips subsequent to the tests is expected to have remedied all of these problems.

A separate type of deficiency that had to be dealt with during these tests was a discrepancy in the earth coordinates of landmarks. Landmarks not on the American Continents appeared to have inconsistent earth coordinates with respect to American landmarks. The coordinates for all land features had been obtained from official detailed maps. An extensive test was conducted by Larry Hambrick on VIRGS (using the star technique) to establish the size of the discrepancy for about a dozen non-American land features. For some islands in the South Pacific, the discrepancy was as high as five image pixels. A discrepancy of at least two pixels was found for most of the non-American features. Based on this test it was concluded that a calibration of the coordinates of non-American features with respect to American land features was needed and would further improve the accuracy of VIRGS.

V. SUMMARY

The accurate earth location of GOES/VISSR image data requires a complete mathematical formulation that is founded on the basic geometry. The use of stars together with landmarks as the observables in the images affords a small, efficient software package that employs relatively simple algorithms and models for determining the true misalignment angles, attitude, and orbit.

The capability described in this document has been implemented on VIRGS, the central VISSR earth-location facility operated by NOAA/NESS and on the McIdas at the University of Wisconsin. Direct readout users have convenient access to the VIRGS generated earth-location parameters, and by using a standard geometry transform routine, can locate the VISSR data automatically. The NESS standard grids, also produced by VIRGS, are embedded in the VISSR data at the infrared pixel resolution.

The VIRGS should be operated at the best achievable level of accuracy. A commitment to an accurate, reliable centralized service would save individual users a sizable task. The users face the choice of exercising their own VIRGS-type operation or simply using the earth-location parameters of the centralized service.

REFERENCES

1. Mottershead, T. and Phillips, D., 1976: Image Navigation for Geosynchronous Meteorological Satellites, presented at 7th Conference on Aerospace and Aeronautical Meteorology and Symposium on Remote Sensing from Satellites, Florida Institute of Technology, Melbourne, FL.
2. Phillips, D., and E. A. Smith, 1974: Geosynchronous Satellite Navigation Model, in Studies of the Atmosphere Using Aerospace Probes 1974, Space Science and Engineering Center, University of Wisconsin Press, Madison, WI.
3. Proceedings of Interactive Video Displays for Atmospheric Studies, Sponsored by NSF, held at the University of Wisconsin June, 1977.
4. Smith, E. A., 1975: The McIdas system, IEEE Trans. Geosci, Electronics, pp. 123-136.
5. Smith, E. A., and Phillips, D. R., July, 1972, Automated Cloud Tracking Using Precisely Aligned Digital ATS Pictures, IEEE Trans. on Computers, pp. 715-729.
6. Westinghouse Electric Corporation, Report Entitled "Earth Location Equations," Contract NAS 5-23582, July 20, 1977.

APPENDIX A

CONVERTING A POINTING VECTOR TO LINE AND ELEMENT NUMBERS

The pointing vector in the first satellite centered coordinate system is designated as $\vec{k}$. The projection of $\vec{k}$ onto the z-axis of the first satellite coordinate system is simply $\vec{k}\cdot(-\vec{S})$, where $\vec{S}$ is the spin axis vector. In terms of scan line number and the misalignment parameters, the projection of the pointing direction of the spin scan camera onto the z-axis of the first satellite coordinate system is simply the z component of (2.5) or

$$m_{3,1}\cdot\cos((c_\ell-\ell)\cdot r_\ell) + m_{3,3}\cdot\sin((c_\ell-\ell)\cdot r_\ell). \qquad \text{(A.O)}$$

To determine the line number at which the spin scan camera is pointing in the direction $\vec{k}$, simply equate (A.0) to $\vec{k}\cdot(-S)$ and solve for ℓ. Solving this equation requires a trigometric manipulation.

Divide (A.O) by $(m_{3,1}{}^2+m_{3,3}{}^2)^{1/2}$,
set θ=ATAN2$(m_{3,1},m_{3,3})$ and
substitute $\sin(\theta)$ for $m_{3,1}/(m_{3,1}{}^2+m_{3,3}{}^2)^{1/2}$ and
$\cos(\theta)$ for $m_{3,3}/(m_{3,1}{}^2+m_{3,3}{}^2)^{1/2}$.

The result is

$$\sin(\theta)\cdot\cos((c_\ell-\ell)\cdot r_\ell) + \cos(\theta)\cdot\sin((c_\ell-\ell)\cdot r_\ell)$$
$$=\vec{k}\cdot(-\vec{S})/(m^2_{3,1}+m^2_{3,3})^{1/2}$$

Setting $\phi = \text{ASIN}(\vec{k}\cdot(-\vec{S})/(m^2_{3,1}+m^2_{3,3})^{1/2})$, using the sum angle sine angle formula on the left-hand side and applying the arcsine function to both sides gives:

$$(c_\ell-\ell)\cdot r_\ell + \theta = \phi. \qquad \text{(A.1)}$$

This linear expression for line number is then solved. The resultant value of the line number is checked to ensure it lies within the bounds for the image, i.e., $1\le\ell\le 14568$.

Finally, the element count is to be found. This is done by equating the angular positions in the spin plane, around the z-axis of our first satellite coordinate system, as given by the camera geometry and as given by the $\vec{k}$ projection. By taking arctangents of the ratio of the x and y components of the vector formulated by (2.5) one finds that the angular component of this vector in the spin plane is given as follows:

$$-(\gamma-\beta.t_\beta.s_r - s.r_s) +$$

$$\text{ATAN2}[m_{2,1}\cos(\lambda)+m_{2,3}\sin(\lambda):m_{1,1}\cos(\lambda)+m_{1,3}\sin(\lambda)] \quad (A.2)$$

where $\lambda = ((c_\ell-\ell).r_\ell)$.

The x component and y component of the pointing vector $\vec{k}$ in the same coordinate system are found respectively by projections $\vec{k}\cdot\vec{P}$ and $\vec{k}\cdot\vec{P}\vec{X}\vec{S}_p$ where $\vec{P}$ and $\vec{S}_p$ come from the matrix formulated in (2.6). Hence, the angular position of this pointing vector in the spin plane equals ATAN2($\vec{k}\cdot\vec{P}\vec{X}\vec{S}_p$, $\vec{k}\cdot\vec{P}$). This is simply equated to (A.2) and the resultant linear equation is solved for the sample number. Some care is required so that an incorrect multiple of 2π is not inadvertently included in one's answer.

APPENDIX B

DETERMINATION OF ATTITUDE AND ORBITAL PLAN ORIENTATION

In order to determine the satellite's attitude and the orientation of the satellite's orbital plane one has the task to minimize the following expression:

$$S(a, b, c) = \sum_{i=1}^{k} (ax_i + by_i + cz_i - \cos(\phi_i))^2 \qquad \text{(B.1)}$$

subject to the constraint $a^2+b^2+c^2=1$ and where for attitude determination ϕ_i equals $\pi/2+r_1(c_1-l_i)$, see figure 8, and for the orientation of the orbital plane all the ϕ_i's equal $\pi/2$. The constraint is absorbed by setting $c = -(1-(a^2+b^2))^{1/2}$. Starting with a and b equal to zero sets the initial spin axis vector guess and the orbital plane perpendicular guess pointing opposite to the spin axis of the earth. This is a good operational approximation and, consequently, reduces the number of iterations required during computation. The change of variables modifies the problem to minimizing the following:

$$S(a,b) = \sum_{i=1}^{k} (ax_i + by_i - (1-(a^2+b^2))^{1/2} \cdot z_i - \cos(\phi_i))^2. \qquad \text{(B.2)}$$

The first partials of S with respect to a and b are set equal to zero resulting in the following two equations:

$$\sum_{i=1}^{k} (x_i - az_{i/c}) \cdot (ax_i + by_i + cz_i - \cos(\phi_i)) = 0$$
$$\sum_{i=1}^{k} (y_i - bz_{i/c}) \cdot (ax_i + by_i + cz_i - \cos(\phi_i)) = 0 \qquad \text{(B.3)}$$

Note the use of the symbol c here for $-(1-(a^2+b^2))^{1/2}$. Manipulation yields the following matrix equation for the individual terms of the summation:

$$\begin{pmatrix} x_i^2 - z_i^2 + z_i\cos(\phi_i)/c & x_i y_i \\ x_i y_i & y_i^2 - z_i^2 + z_i \dfrac{\cos(\phi_i)}{c} \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \qquad \text{(B.4)}$$
$$\begin{pmatrix} x_i\cos(\phi_i) - cx_i z_i + a^2 x_i z_i/c + aby_i z_i/c \\ y_i\cos(\phi_i) - cy_i z_i + abx_i z_i/c + b^2 y_i z_i/c \end{pmatrix}$$

The matrix on the left is inverted and the resulting system of equations is solved iteratively. Note that all the terms on the right-hand side outside of the constant terms are second order. The advantage of such a separation is that the linear equation is solved for first and then the results are updated with the second order terms. This approach is a variation of Newton's Method. The summation signs have been omitted to compact the presentation. Convergence is achieved within three or four iterations.

For attitude determination with an unknown pitch misalignment angle, ζ, the following expression of three variables has to be minimized:

$$S(a, b, c, \zeta) = \sum_{i=1}^{k} (ax_i + by_i + cz_i - \cos(\phi_i - \zeta))^2 \qquad \text{(B.5)}$$

subject to the constraint $a^2+b^2+c^2=1$.

The expression (B.5) is not easily amenable to the variation of Newton's Method used earlier to find the attitude and hence the expression is reformulated to an equivalent problem. The constraint $a^2+b^2+c^2=1$ is absorbed by setting $c= -(1-(a^2+b^2))^{1/2}$ and u is set to $\sin(\zeta)$ and $-(1-u^2)^{1/2}$ is set to $\cos(\zeta)$. Once u is solved for, ζ can be found as the arctangent of $(-u/(1-u^2)^{1/2})$, or in Fortran as ATAN2$(u,-(1-u^2)^{1/2})$. With this formulation, the following is minimized:

$$S(a,b,u) = \sum_{i=1}^{k} (ax_i + by_i - (1-(a^2+b^2))^{1/2} z_i + u\cos(\phi_i) + (1-u^2)^{1/2}\sin(\phi_i))^2. \qquad \text{(B-6)}$$

To solve this minimization problem the first partials of S with respect to a, b, and u are set equal to zero and the resulting expressions are rearranged to have a matrix times the vector (a, b, u) on the left-hand side and constant terms plus second order terms on the right-hand side, as in (B-4). The inverse of the matrix is then applied to both sides and the resulting equations are solved by iteration using Newton's Method. Convergence is again achieved within three or four iterations.

APPENDIX C

ACCURACY OF APPROXIMATE ORBIT EQUATION

(Refer to Section III-D, Equation 4.4)

In this paper the equation

$$n(t-T) = \gamma - 2\epsilon \sin\gamma \qquad (C.1)$$

has been used to approximate the motion of a satellite in a Keplerian orbit. Equation C.1 is deriveable from (4.4) with the substitution $\gamma = \alpha - \omega$. In eq. (C.1), n is the mean motion constant, t is time, T is the time of a perifocus passage, γ is the true anamoly and ϵ the eccentricity. The accuracy of this equation for eccentricities less than or equal to 0.001 will now be established.

Consider the following three equations taken from Escabol:

$$n(t-T) = E - \epsilon \sin E, \qquad (C.2)$$

$$\sin E = \frac{\sqrt{1-\epsilon^2}\sin\gamma}{1+\epsilon\cdot\cos\gamma} \quad \text{and} \qquad (C.3)$$

$$\cos E = \frac{\cos\gamma + \epsilon}{1+\epsilon\cos\gamma} \qquad (C.4)$$

where E is the eccentric anamoly.

Multiplying (C.3) by $\cos\gamma$ and (C.4) by $\sin\gamma$ and subtracting yields: $\cos\gamma \sin E - \sin\gamma \cos E =$

$$\frac{\sqrt{1-\epsilon^2}\sin\gamma\cos\gamma - \sin\gamma\cos\gamma - \epsilon\cdot\sin\gamma}{1+\epsilon\cdot\cos\gamma} \qquad (C.5)$$

Using the sine angle difference formula changes that left hand side to $\sin(E-\gamma)$.

The next two inequalities are well know for $|E-\gamma|$ within the range of consideration:

$$|\sin(E-\gamma)| \geq |E-\gamma| \left(1-\frac{(E-\gamma)^2}{3!}\right) \quad \text{and} \qquad (C.6)$$

$$|\sin(E-\gamma) - (E-\gamma)| \leq \frac{|E-\gamma)^3}{3!}) \qquad (C.7)$$

Using (C.5) and restricting ϵ to be less than or equal to 0.001 yields

$$|\sin(E-\gamma)| \leq 0.001002.$$

Combining this with (C.6) yields

$$|E-\gamma| \leq .001003$$

Finally, (C.7) yields

$$|\sin(E-\gamma) - (E-\gamma)| \le 1.682\times 10^{-10} \qquad (C.8)$$

Now (C.5) is rewritten as

$$\sin(E-\gamma) = -\epsilon\sin\gamma + \frac{\sqrt{1-\epsilon^2}\sin\gamma\cos\gamma - \sin\gamma\cos\gamma + \epsilon^2\sin\gamma\cos\gamma}{1+\epsilon\cdot\cos\gamma} \qquad (C.9)$$

Combining (C.9) with the inequality (C.8) and appropriately bounding the last term in (C.9) yields

$$|E-\gamma+\epsilon\sin\gamma| \le 1.6\times 10^{-6} \qquad (C.10)$$

Next, (C.3) is rewritten as

$$\sin E - \sin\gamma = \frac{\sqrt{1-\epsilon^2}\sin\gamma - \sin\gamma - \epsilon\cdot\cos\gamma\sin\gamma}{1+\epsilon\cdot\cos\gamma} \qquad (C.11)$$

This yields the inequality

$$|\sin E - \sin\gamma| \le .001002 \qquad (C.12)$$

Combining (I.2) with the inequalities (C.10) and (C.12) yields

$$|n(t-T) - \gamma + 2\epsilon\sin\gamma| \le 2.7\times 10^{-6} \qquad (C.13)$$

This yields a positional accuracy to within 0.12 kilometers. Thus the accuracy of the approximation of using (C.1) is established.

APPENDIX D

GAUSSIAN ELIMINATION

The process of Guassian elimination is now recalled for the reader. One begins with a matrix pair (A:I), where A is the matrix whose inverse is sought and I is the identity matrix, and applies a sequence of non-singular matrices B_i to both members of the matrix pair from the left until $B_n B_{n-1} \dots B_1 A$ is the identity matrix. At that point the product accumulated on the right, $B_n B_{n-1} \dots B_1$, is the inverse of the matrix A since its product with A yields the identify matrix. Usually the B_i's are selected so that the matrix multiplication affects only one row of the matrix pair. Because the matrix (4.12) is symmetric, positive definite, i.e., $a_{i,j}=a_{j,i}$, and all the eigenvalues are positive, the matrix operands B_i can be selected to act on more than one row in one operation. In this case the B_i's act on two rows. The matrix (4.12) is treated as if it were partitioned into four parts, each part consisting of a two by two matrix. Gaussian elimination is then applied as if each two by two submatrix was a single element.

Start with the matrix pair $\begin{pmatrix} A_{1,1} & A_{1,2} \\ A_{2,1} & A_{2,2} \end{pmatrix} : \begin{pmatrix} I & O \\ O & I \end{pmatrix}$ and apply $\begin{pmatrix} I & O \\ O & A_{2,2}^{-1} \end{pmatrix}$ to both sides with the result

$$\begin{pmatrix} A_{1,1} & A_{1,2} \\ A_{2,2}^{-1}A_{2,1} & I \end{pmatrix} : \begin{pmatrix} I & O \\ O & A_{2,2}^{-1} \end{pmatrix} .$$

Next apply $\begin{pmatrix} I & -A_{1,2} \\ O & I \end{pmatrix}$ with the result

$$\begin{pmatrix} A_{1,1}-A_{1,2}A_{2,2}^{-1}A_{2,1} & O \\ A_{2,2}^{-1}A_{2,1} & I \end{pmatrix} : \begin{pmatrix} I & -A_{1,2}A_{2,2}^{-1} \\ O & A_{2,2}^{-1} \end{pmatrix} .$$

Then set $S = (A_{1,1} - A_{1,2}A_{2,2}^{-1}A_{2,1})^{-1}$ and apply $\begin{pmatrix} S & O \\ O & I \end{pmatrix}$ with the result

$$\begin{pmatrix} I & O \\ A_{2,2}^{-1}A_{2,1} & I \end{pmatrix} : \begin{pmatrix} S & -SA_{1,2}A_{2,2}^{-1} \\ O & A_{2,2}^{-1} \end{pmatrix} .$$

Finally, apply $\begin{pmatrix} I & O \\ -A_{2,2}^{-1}A_{2,1} & I \end{pmatrix}$ to both matrices with the result

$$\begin{pmatrix} I & O \\ O & I \end{pmatrix} : \begin{pmatrix} S & -SA_{1,2}A_{2,2}^{-1} \\ -A_{2,2}^{-1}A_{2,1}S & A_{2,2}^{-1}+A_{2,2}^{-1}A_{2,1}SA_{1,2}A_{2,2}^{-1} \end{pmatrix} .$$

The inverse of (4.12) is the right-hand term. Actually, computing the lower left-hand term is unnecessary since it is the transpose of the upper right term by symmetry.

Appendix E

Fortran Listings of Orbit and Attitude Determination Routines: FINDAT and UPGORB

```
      SUBROUTINE FINDAT(NOPCLN,RADLIN,DECLIN,RASCEN,PCLN)
      DOUBLE PRECISION UX,UY,UZ,PSI
      COMMON/MINCOM/UX(80),UY(80),UZ(80),PSI(80),NP
      DATA RDPDG/1.745329252E-2/
C     WRITTEN NOVEMBER 13,1979 BY DENNIS PHILLIPS TO SIMPLIFY ATTITUDE
C     COMPUTATION.  PROGRAM RIGHTS BELONG TO SCIENTIFIC PROGRAMMING
C     AND APPLIED MATHEMATICS , INC.
C     THIS PROGRAM ELIMINATES THE USE OF STEEPEST DESCENT METHODS IN
C     THE ATTITUDE COMPUTATIONS AND INSTEAD REPLACES THESE METHODS WITH
C     LINEAR REGRESSION METHODS COMBINED WITH ITERATION STEPS.
C     TWO PROBLEMS ARE WORKED OUT IN THIS CODE.  ONE PROBLEM IS TO
C     MINIMIZE THE SUM OVER I OF (A*X(I)+B*Y(I)+C*Z(I)-COS(PSI(I)))**2.
C     WITH THE CONSTRAINT A**2+B**2+C**2=1.
C     THE OTHER PROBLEM IS TO MINIMIZE THE SUM OVER I OF
C     (A*X(I)+B*Y(I)+C*Z(I)-COS(PSI(I)-PSIREF))**2 OR EQUIVALENTLY THE
C     SUM OVER I OF (A*X(I)+B*Y(I)+C*Z(I)-V*COS(PSI(I))-U*SIN(PSI(I)))**2
C     WITH THE CONSTRAINTS A**2+B**2+C**2 AND U**2+V**2=1.
C     ***
C     FOR PICTURE CENTER LINE SOLUTION GO TO 30
C     ***
      IF(NOPCLN.EQ.1)GO TO 30
C     RETURN AND SEND ERROR MESSAGE FOR INSUFFICIENT MEASUREMENTS
      IF(NP.LT.2)GO TO 75
C     ***
C     COMPUTE ATTITUDE
C     ***
C     COLLECT REGRESSION COEFFICIENTS
      SMXSQ=0.0
      SMXY=0.0
      SMXZ=0.0
      SMXCS=0.0
      SMYSQ=0.0
      SMYZ=0.0
      SMYCS=0.0
      SMZSQ=0.0
      SMZCS=0.0
      DO 10 I=1,NP
      X=UX(I)
      Y=UY(I)
      Z=UZ(I)
      ANG=PSI(I)
      CS=COS(ANG)
      SMXSQ=SMXSQ+X*X
      SMXY=SMXY+X*Y
      SMXZ=SMXZ+X*Z
      SMXCS=SMXCS+X*CS
      SMYSQ=SMYSQ+Y*Y
      SMYZ=SMYZ+Y*Z
      SMYCS=SMYCS+Y*CS
      SMZSQ=SMZSQ+Z*Z
   10 SMZCS=SMZCS+Z*CS
C     SET INITIAL VALUES AND UPDATE VALUES
      A=0.0
```

```
      B=0.0
      C=-1.0
      ANEW=0.0
      BNEW=0.0
C     SET ITERATION COUNT
      ITER=0
C     COMPUTE MATRIX COEFFICIENTS FOR MATRIX EQUATION (A11 A12) A = F1
C                                                     (A21 A22) B = F2
   20 A11=SMXSQ+SMZCS/C-SMZSQ
      A12=SMXY
      A21=SMXY
      A22=SMYSQ+SMZCS/C-SMZSQ
C     INVERT THE MATRIX A
      DET=1.0/(A11*A22-A12*A21)
      B11=A22*DET
      B12=-A12*DET
      B21=B12
      B22=A11*DET
C     COMPUTE CONSTANT PLUS SECOND ORDER TERMS OF THE VECTOR F
      F1=SMXCS-C*SMXZ+ANEW**2*SMXZ/C+ANEW*BNEW*SMYZ/C
      F2=SMYCS-C*SMYZ+BNEW**2*SMYZ/C+ANEW*BNEW*SMXZ/C
C     COMPUTE NEW A AND B VALUES
      ANEW=B11*F1+B12*F2
      BNEW=B21*F1+B22*F2
      C=-SQRT(1.0-(ANEW**2+BNEW**2))
C     COUNT ITERATION STEPS
      ITER=ITER+1
C     RETURN AND SEND ERROR MESSAGE IF ITERATION LIMIT IS EXCEEDED
      IF(ITER.GE.10)GO TO 80
C     CHECK CONVERGENCE CRITERIA
      IF(ABS(A-ANEW)+ABS(B-BNEW).LT.1.0E-8)GO TO 25
C     UPDATE VALUES
      A=ANEW
      B=BNEW
C     ITERATE AGAIN
      GO TO 20
   25 DECLIN=-90.0+ASIN(SQRT(A**2+B**2))/RDPDG
      RASCEN=0.0
      IF(ABS(A)+ABS(B).GT.1.0E-8)RASCEN=ATAN2(B,A)/RDPDG
      RETURN
C     ***
C     COMPUTE ATTITUDE AND PICTURE CENTER LINE OFFSET
C     ***
   30 IF(NP.LT.3)GO TO 85
C     COLLECT REGRESSION COEFFICIENTS
      SMXSQ=0.0
      SMXY=0.0
      SMXZ=0.0
      SMXCS=0.0
      SMXSN=0.0
      SMYSQ=0.0
      SMYZ=0.0
      SMYCS=0.0
```

```
      SMYSN=0.0
      SMZSQ=0.0
      SMZCS=0.0
      SMZSN=0.0
      SMCSSQ=0.0
      SMCSSN=0.0
      SMSNSQ=0.0
      DO 35 I=1,NP
      X=UX(I)
      Y=UY(I)
      Z=UZ(I)
      ANG=PSI(I)
      SN=SIN(ANG)
      CS=COS(ANG)
      SMXSQ=SMXSQ+X*X
      SMXY=SMXY+X*Y
      SMXZ=SMXZ+X*Z
      SMXCS=SMXCS+X*CS
      SMXSN=SMXSN+X*SN
      SMYSQ=SMYSQ+Y*Y
      SMYZ=SMYZ+Y*Z
      SMYCS=SMYCS+Y*CS
      SMYSN=SMYSN+Y*SN
      SMZSQ=SMZSQ+Z*Z
      SMZCS=SMZCS+Z*CS
      SMZSN=SMZSN+Z*SN
      SMCSSQ=SMCSSQ+CS*CS
      SMCSSN=SMCSSN+CS*SN
   35 SMSNSQ=SMSNSQ+SN*SN
C     SET INITIAL AND UPDATE VALUES
      A=0.0
      B=0.0
      C=-1.0
      ANEW=0.0
      BNEW=0.0
      U=0.0
      V=1.0
      UNEW=0.0
C     SET ITERATION COUNT
      ITER=0
C     COMPUTE COEFFICIENTS FOR MATRIX EQUATION (A11 A12 A13) A = F1
C                                                (A21 A22 A23) B = F2
C                                                (A31 A32 A33) U = F3
   40 A11=SMXSQ-SMZSQ+V*SMZCS/C
      A12=SMXY
      A13=-SMXSN
      A21=A12
      A22=SMYSQ-SMZSQ+V*SMZCS/C
      A23=-SMYSN
      A31=A13
      A32=A23
      A33=C*SMZCS/V-SMCSSQ+SMSNSQ
C     ***
```

```
C     THE INVERSION OF THE MATRIX A IS DONE BY PARTITIONING THE MATRIX
C     INTO FOUR PARTS: (A11), (A12 A13) , (A21) AND (A22 A23)
C                                         (A31)     (A32 A33)
C     THEN EACH MEMBER OF THE PARTITION IS TREATED AS A SINGLE ELEMENT
C     AND GAUSSIAN ELIMINATION IS CARRIED OUT STARTING IN THE LOWER LEFT
C     HAND CORNER
C     ***
C     FIND THE INVERSE OF THE LOWER LEFT HAND CORNER
      DET=1.Ø/(A22*A33-A23*A32)
      C11=A33*DET
      C22=A22*DET
      C12=-A23*DET
      C21=C12
C     MULTIPLY THIS INVERSE BY THE LAST TWO ENTRIES OF THE FIRST ROW
      B1=-A12*C11-A13*C21
      B2=-A12*C12-A13*C22
C     FIND FIRST ELEMENT OF INVERSE MATRIX
      CDIVID=1.Ø/(A11+B1*A21+B2*A31)
      B11=CDIVID
C     FIND THE SECOND AND THIRD ELEMENTS OF FIRST ROW OF MATRIX
      B12=B1*CDIVID
      B13=B2*CDIVID
C     SET TWO MORE ENTRIES BY SYMMETRY
      B21=B12
      B31=B13
C     COMPUTE THE REMAINING TWO BY TWO PART
      B22=C11+CDIVID*B1*B1
      B23=C12+CDIVID*B1*B2
C     BY SYMMETRY
      B32=B23
      B33=C22+CDIVID*B2*B2
C     COMPUTE RIGHT-HAND SIDE OF LINEAR EQUATION
      F1=V*SMXCS-C*SMXZ+A**2*SMXZ/C+A*B*SMYZ/C-A*U*SMZSN/C
      F2=V*SMYCS-C*SMYZ+A*B*SMXZ/C+B**2*SMYZ/C-B*U*SMZSN/C
      F3=-V*SMCSSN+C*SMZSN-A*U*SMXCS/V-B*U*SMYCS/V+U**2*SMCSSN/V
C     COMPUTE NEW A, B AND U VALUES
      ANEW=B11*F1+B12*F2+B13*F3
      BNEW=B21*F1+B22*F2+B23*F3
      UNEW=B31*F1+B32*F2+B33*F3
      ITER=ITER+1
C     RETURN AND SEND ERROR MESSAGE IF ITERATION LIMIT IS EXCEEDED
      IF(ITER.GE.15)GO TO 9Ø
C     CHECK CONVERGENCE CRITERIA
      IF(ABS(A-ANEW)+ABS(B-BNEW)+ABS(U-UNEW).LT.1.ØE-8)GO TO 45
C     UPDATE VALUES
      A=ANEW
      B=BNEW
      C=-SQRT(1.Ø-(A*A+B*B))
      U=UNEW
      V=SQRT(1.Ø-U**2)
C     ITERATE AGAIN
      GO TO 4Ø
C     COMPUTE DECLINATION, RIGHT ASCENSION AND PICTURE CENTER LINE OFFST
```

```
   45 DECLIN=-90.0+ASIN(SQRT(A*A+B*B))/RDPDG
      RASCEN=0.0
      IF(ABS(A)+ABS(B).GT.1.0E-8)RASCEN=ATAN2(B,A)/RDPDG
      PCLN=PCLN-ATAN2(U,V)/RADLIN
      RETURN
C     SEND ERROR MESSAGES
   75 WRITE(6,95)
   95 FORMAT(5X,'INSUFFICIENT MEASUREMENTS FOR ATTITUDE DETERMINATION')
      RETURN
   80 WRITE(6,96)
   96 FORMAT(5X,'ITERATION LIMIT EXCEEDED IN ATTITUDE DETERMINATION')
      RETURN
   85 WRITE(6,97)
   97 FORMAT(5X,'INSUFFICIENT MEASUREMENTS FOR ATTITUDE AND PCLN DET.')
      RETURN
   90 WRITE(6,98)
   98 FORMAT(5X,'ITERATION LIMIT EXCEEDED IN ATTITUDE AND PCLN DET.')
      RETURN
      END
```

SIZE 945 01661

```
C UPGORB PHILLI Ø679 NAVL COMPUTE ORBIT FROM LANDMARKS,BETAS, AND ATTITUØØØ4
C
C     THIS PROGRAM HAS TWO MODES: ONE MODE TO COMPUTE ALL THE ORBIT
C     PARAMETERS AND THE OTHER MODE TO COMPUTE JUST THE ALONG-TRACK
C     ORBIT PARAMETERS ASSUMING THE ORBIT'S INCLINATION AND
C          ASCENDING NODE HAVE BEEN PROVIDED.  MIN(3) CONTROLS THE
C          NUMBER OF ALONG-TRACK ORBIT PARAMETERS TO BE COMPUTED.
C     MIN(6) IS THE FLAG CONTROLING THE ORBITAL PLANE CALCULATION.
C
C     MIN(1) = SSYYDDD
C     MIN(2) = HHHH, THE FIRST TWO DIGITS ARE THE FIRST HOUR ON THE YYDD
C                 TO START.  THE LAST TWO DIGITS ARE THE NUMBER OF
C                  HOURS FROM THE FIRST HOUR TO CONSIDER.
C     MIN(3) = NO. OF PARAMETERS TO COMPUTE.
C     MIN(4) = EPOCH OF THE CALCULATED ORBIT PARAMETERS.
C     MIN(5) = PRINT AND STORE OPTION.
C     MIN(6) = ORBITAL PLANE CALCULATION FLAG.  OP MEANS ON.
      SUBROUTINE MAIN
      REAL MEANOM,PTIMLM(1ØØ),XLINLM(1ØØ),XELELM(1ØØ)
      REAL XLATLM(1ØØ),XLONLM(1ØØ)
      DOUBLE PRECISION BETA(1ØØ),V(1ØØ),T(1ØØ)
      INTEGER HHHH
      INTEGER MIN(8),LCODE(1ØØ),ILNDBT(1ØØ)
      INTEGER LNDPNT(1ØØ)
      COMMON/NAVCOM/NAVDAY,LINTOT,DEGLIN,IELTOT,DEGELE,SPINRA,IETIMY,IET
     1IMH,SEMIMA,OECCEN,ORBINC,PERHEL,ASNODE,NOPCLN,DECLIN,RASCEN,PICLIN
     2,PRERAT,PREDIR,PITCH,YAW,ROLL,SKEW
      DATA MIN/'UPGORB',6*Ø/
      DATA MAXLND/1ØØ/,NUMDAY/1/
      CALL IQ(MIN)
      ISYD=MIN(1)
      HHHH=MIN(2)
      NUMDAY=(HHHH/1ØØ+MOD(HHHH,1ØØ))/24+1
C     THE CALL TO SETUPN INITIALIZES THE PARAMETERS IN THE NAVCOM AND
C     NAVIN COMMON BLOCKS
      CALL SETUPN(ISYD,Ø)
C     FETCH THE LANDMARKS FOR THE TIME PERIOD
      CALL GTLM(ISYD,NUMDAY,MAXLND,NUMLND,
     *     PTIMLM,LCODE,XLINLM,XELELM,XLATLM,XLONLM)
C     THIS CALL TO GTLMBT MATCHES BETA VALUES TO THE AVAILABLE LANDMARKS
C     BY INTERPOLATING OR EXTRAPOLATING THE BETA VALUES THAT ARE
C     AVAILABLE IN THE SAME IMAGE WITH THE LANDMARKS.  WHEN SUCH BETAS
C     ARE AVAILABLE FOR A SPECIFIC LANDMARK, THE CORRESPONDING INDEX
C     POSITION OF THE LANDMARK IN THE ARRAY ILNDBT IS SET TO ONE;
C     OTHERWISE THIS VALUE IS SET TO A MINUS ONE.
      CALL GTLMBT(ISYD,NUMDAY,HHHH,NUMLND,PTIMLM,XLINLM,XELELM,LCODE,
     *     ILNDBT,BETA,T)
C     COUNT LANDMARKS HAVING ASSOCIATED BETA VALUES
      IMATCH=Ø
      DO 2 I=1,NUMLND
      IF(ILNDBT(I).EQ.1)IMATCH=IMATCH+1
2     CONTINUE
C     TRANSMIT ERROR MESSAGE AND RETURN WHEN NOT ENOUGH LANDMARKS HAVE
```

```
C     ASSOCIATED BETA COUNTS
      IF(IMATCH.GE.MIN(3))GO TO 3                                       0037
      CALL TQMES('NOT ENOUGH MATCHES BETWEEN LANDMARKS AND BETAS$',IMATC0038
     *H)                                                                0039
      RETURN                                                            0040
3     CONTINUE                                                          0041
      IF(MIN(3).GE.1.AND.MIN(3).LE.4)GO TO 4                            0042
      CALL TQMES('WRONG NUMBER OF ORBIT PARAMETERS TO BE COMPUTED$',    0043
     *MIN(3))                                                           0044
      RETURN                                                            0045
4     ISEMI=MIN(3)                                                      0046
      EPTIME=FTIME(MIN(4))                                              0047
      IHMS=MIN(4)                                                       0048
      IORBPL=0                                                          0049
      IF(MIN(6),EQ.2HOP)IORBPL=1                                        0050
C
C     THE ITERATION LOOP (DO 6 I=1,3) CORRECTS THE HEIGHT OF THE
C     SATELLITE AS A FUNCTION OF TIME USING THE MOST RECENTLY
C     COMPUTED VALUES FOR THE ORBIT PARAMETERS.  WHEN NO ORBIT
C     PARAMETERS ARE AVAILABLE, THE NOMINAL GEOSTATIONARY HEIGHT OF
C     42165.0 IS USED.  THIS HEIGHT CORRECTION IS DONE IN ANGORB
C
      IETIMH=MIN(4)                                                     0054
      IETIMY=MOD(ISYD,100000)                                           0055
      ITER=5
      MEANOM=0.0
      DO 6 I=1,ITER                                                     0052
C
C     THE CALL TO ANGORB SERVES TWO FUNCTIONS: (1) IF IORBPL=1, THE
C     INCLINATION AND ASCENDING NODE OF THE ORBITAL PLANE ARE DETERMINED
C     AND (2) THE TRUE ANAMOLY POSITIONS + OFFSET OF THE SATELLITE ARE
C     COMPUTED.  THE TRUE ANAMOLIES+OFFSET ARE STORED IN THE ARRAY V(I).
C
      CALL ANGORB(ISYD,NUMLND,MEANOM,PTIMLM,XLATLM,XLONLM,
     *XLINLM,XELELM,ILNDBT,BETA,V,T,NV,LNDPNT,IORBPL)                   0059
C
C     COMPUTE THE ALONG-TRACK ORBIT PARAMETERS.  THE NUMBER OF
C     PARAMETERS COMPUTED DEPENDS ON THE INDEX ISEMI.  HENCE ISEMI
C     EQUALS ONE, TWO, THREE OR FOUR.
C
      CALL ORBPR(V,T,NV,MEANOM,ISEMI,EPTIME)                            0060
    6 CONTINUE                                                          0061
      IOUT=1                                                            0062
C
C     COMPUTE RESIDUALS FOR LANDMARK MEASUREMENTS USING THE NEWLY
C     COMPUTED ORBITAL STATE AND OUTPUT RESIDUALS TO TERMINAL.
C
      CALL ORBRES(ISYD,IOUT,NV,T,XLINLM,XELELM,BETA,MEANOM,
     *    LCODE,XLATLM,XLONLM,LNDPNT)                                   0064
      IOUT=2                                                            0065
C
C     OUTPUT RESIDUALS TO PRINTER IF PRINTER OPTION IS ON.
C
```

```
      IF(MIN(5).EQ.2HPS.OR.MIN(5).EQ.2HSP.OR.MIN(5).EQ.1HP)CALL ORBRES
     *(ISYD,IOUT,NV,T,XLINLM,XELELM,BETA,MEANOM,LCODE,XLATLM,XLONLM,LNDP
     *NT)
C
C     STORE NAVIGATIONAL PARAMETERS IN NAVFILES IF STORAGE OPTION IS ON.
C
      IF(MIN(5).EQ.2HPS.OR.MIN(5).EQ.2HSP.OR. MIN(5).EQ.1HS)CALL DQ
     *(ISYD,IHMS,SEMIMA,OECCEN,ORBINC,MEANOM,PERHEL,ASNODE)
      IOUT=1
C
C     OUTPUT RESULTANT ORBIT PARAMETERS TO THE TERMINAL CRT.
C
      CALL ORBOUT(IOUT,ISYD,IHMS,SEMIMA,OECCEN,ORBINC,MEANOM,PERHEL,
     *     ASNODE)
      IOUT=2
C
C     IF PRINTER OPTION IS ON, ALSO OUTPUT ORBIT PARAMETERS TO PRINTER.
C
      IF(MIN(5).EQ.2HPS.OR.MIN(5).EQ.2HSP.OR.MIN(5).EQ.1HP)CALL ORBOUT
     *(IOUT,ISYD,IHMS,SEMIMA,OECCEN,ORBINC,MEANOM,PERHEL,ASNODE)
      RETURN
      END
```

SIZE 2221 04255

```
      SUBROUTINE ORBOUT(IOUT,ISYD,IHMS,SEMIMA,OECCEN,ORBINC,MEANOM,
     *  PERHEL,ASNODE)
C
C     THE SUBROUTINE ORBOUT TRANSMITS THE NEWLY COMPUTED ORBIT
C     PARAMETERS TO THE TERMINAL CRT OR TO THE PRINTER DEPENDING ON THE
C     VALUE OF IOUT.  IOUT=1 MEANS TERMINAL.  IOUT =2 MEANS PRINTER.
C
C     INPUTS:  ALL PARAMETERS
C     OUTPUTS: NO NEW PARAMETERS OR VALUES ARE RETURNED TO CALLING
C              PROGRAM.  THE ORBITAL PARAMETERS ARE TRANSMITED TO THE
C              OUTPUT DEVICE SPECIFIED BY IOUT.
C
      REAL MEANOM
      INTEGER MOUT(132),IARRAY(8)
      CALL YDDMY(ISYD,ID,IM,IY)
      IARRAY(1)=(IY-1900)*10000+IM*100+ID
      IARRAY(2)=IHMS
      IARRAY(3)=IROUND(SEMIMA*100.0)
      IARRAY(4)=IROUND(OECCEN*1000000.0)
      IARRAY(5)=IROUND(ORBINC*1000.0)
      IARRAY(6)=IROUND(MEANOM*1000.0)
      IARRAY(7)=IROUND(PERHEL*1000.0)
      IARRAY(8)=IROUND(ASNODE*1000.0)
C
C     TP TRANSMITS A HOLLERETH FIELD TO THE OUTPUT DEVICE, CRT OR
C          PRINTER
C
      CALL TP(IOUT,132H  UPGORB OUTPUT RESULTS
     *
     *                                                                )
      CALL TP(IOUT,132H KEPLERIAN ORBIT PARAMETERS
     *
     *                                                                )
C
C     MVCHAR MOVEW A HOLLERETH FIELD INTO THE ARRAY MOUT
C
C     FIRST BLANK OUT FIELD
      CALL MVCHAR('BLA','BLA',MOUT,1,132)
C     WRITE FIELD LABELS
      CALL MVCHAR(
     *'   ETIMY   ETIMH  SEMIMA    ECCEN   ORBINC    MEANA PERIGEE  ASNODE'
     *,1,MOUT,1,64)
      CALL TP(IOUT,MOUT)
C     BLANK FIELD AGAIN
      CALL MVCHAR('BLA','BLA',MOUT,1,132)
      CALL TP(IOUT,MOUT)
      DO 10 I=1,8
C     ENCODE ORBIT PARAMETERS IN INTEGER HOLLERETH FIELD
      CALL MVCHAR(IARRAY(I),'INT',MOUT,I*8,1)
   10 CONTINUE
C     TRANSMIT HOLLERETH FIELD TO OUTPUT DEVICE
      CALL TP(IOUT,MOUT)
      RETURN
```

18.050575 PAGE 2

END 0109

SIZE 400 00620

```
      SUBROUTINE DQ(ISYD,IHMS,SEMIMA,OECCEN,ORBINC,MEANOM,PERHEL,ASNODE)
C
C     DQ TRANSMITS THE ORBITAL PARAMETERS TO THE NAVIGATIONAL FILES
C
C     INPUTS:  ORBITAL PARAMETERS
C     OUTPUTS:  THE ORBIT PARAMETERS ARE TRANSMITED TO THE NAVIGATIONAL
C            FILES BY NAVFIL.
C
      REAL MEANOM
      INTEGER MESONE(10),MESTWO(10)
      DATA MESONE/'NAVFIL',4,7*0/
      DATA MESTWO/'NAVFIL',5,7*0/
      CALL YDDMY(ISYD,ID,IM,IY)
      IETYMD=(IY-1900)*10000+IM*100+ID
      ISEMI=IROUND(SEMIMA*100.0)
      IOECC=IROUND(OECCEN*1000000.0)
      IASND=IROUND(ASNODE*1000.0)
      IOINC=IROUND(ORBINC*1000.0)
      IPERH=IROUND(PERHEL*1000.0)
      IMEAN=IROUND(MEANOM*1000.0)
      MESONE(4)=ISYD
      MESONE(6)=1
      MESONE(7)=IETYMD
      MESONE(8)=IHMS
      MESONE(9)=ISEMI
      MESONE(5)=MESONE(9)
      MESONE(10)=IOECC
      CALL SQ(MESONE)
      MESTWO(4)=ISYD
      MESTWO(6)=1
      MESTWO(7)=IOINC
      MESTWO(8)=IMEAN
      MESTWO(9)=IPERH
      MESTWO(5)=MESTWO(9)
      MESTWO(10)=IASND
      CALL SQ(MESTWO)
      RETURN
      END
```

SIZE 139 00213

```
      SUBROUTINE GTLMBT(ISYD,NUMDAY,HHHH,NUMLND,PTIMLN,XLINLM,XELELM,
     *     LCODE,ILNDBT,BETA,T)
C
C     GTLMBT GETS BETA COUNTS WITH CALLS TO GETBET AND CHECKS TO SEE
C     IF THE BETA COUNT PAIRS OCCUR AT A PICTURE TIME OF ANY OF THE
C     LANDMARKS.  IF SO, A BETA VALUE FOR THE LANDMARK IS FOUND BY
C     INTERPOLATION OR EXTRAPOLATION.  ALSO, THE LANDMARK CODE IS
C     EXAMINED TO DISCARD STARS.
C     INPUTS:  ISYD.  SATELLITE ID. YEAR DAY
C              NUMDAY.  THE NUMBER OF DAYS OF LANDMARK MEASUREMENTS TO
C                       CHECK.
C              HHHH.  THE BEGINNING HOUR AND ENDING HOUR TO CONSIDER
C                     LANDMARK MEASUREMENTS.
C              NUMLND:  THE NUMBER OF LANDMARKS THAT CAN BE HANDLED.
C     OUTPUTS:  PTIMLM. PICTURE START TIME OF LANDMARK
C               XLINLM.  LINE NUMBER OF LANDMARK
C               XELELM. ELEMENT NUMBER OF LANDMARK
C               LCODE.  LANDMARK CODE
C               ILNDBT.  LANDMARK-BETA ASSOCIATION FLAG.  EQUALS 1 WITH
C                        BETA.  EQUALS -1 WITHOUT BETA.
C               T.  LANDMARK TIME.
C
      IMPLICIT DOUBLE PRECISION (Q)
      DOUBLE PRECISION BETA(1),RDPBT,BETDIF,T(1),BETCMP
      REAL PTIMLN(1),XLINLM(1),XELELM(1)
      INTEGER HHHH
      INTEGER LCODE(1),ILNDBT(1)
      INTEGER NOMTIM(100)
      INTEGER ISCAN1(100),ISTIM1(100),ISMIL1(100),IBET1(100)
      INTEGER ISCAN2(100),ISTIM2(100),ISMIL2(100),IBET2(100)
      COMMON/NAVCOM/NAVDAY,LINTOT,DEGLIN,IELTOT,DEGELE,SPINRA,IETIMY,IET
     1IMH,SEMIMA,OECCEN,ORBINC,PERHEL,ASNODE,NOPCLN,DECLIN,RASCEN,PICLIN
     2,PRERAT,PREDIR,PITCH,YAW,ROLL,SKEW
      COMMON/NAVINI/
     1  EMEGA,AB,ASQ,BSQ,R,RSQ,
     2  RDPDG,
     3  NUMSEN,TOTLIN,RADLIN,
     4  TOTELE,RADELE,PICELE,
     5  CPITCH,CYAW,CROLL,
     6  PSKEW,
     7  RFACT,ROASIN,TMPSCL,
     8  B11,B12,B13,B21,B22,B23,B31,B32,B33,
     9  GAMMA,GAMDOT,
     A  ROTM11,ROTM13,ROTM21,ROTM23,ROTM31,ROTM33,
     B  PICTIM,XREF
      DATA MAXNUM/100/,NEGBET/3144960/,RDPBT/9.989292881D-7/
      DATA BETCMP/3144960.0D0/
      DATA TWPI/6.28318530/
      ISAT=ISYD/100000
      IYR=MOD(ISYD,100000)/1000
      IDAY=MOD(ISYD,1000)
C     COMPUTE BEGINNING AND END TIME FOR TIME INTERVAL CHECK
      BTIME=HHHH/100
```

```
      ETIME=BTIME+MOD(HHHH,100)                                          0170
      DO 10 I=1,NUMLND                                                   0171
   10 ILNDBT(I)=-1                                                       0172
C     THE BETAS ARE FETCHED ONE DAY AT A TIME.
      DO 50 I=1,NUMDAY                                                   0173
      JSYD=ISAT*100000+IYR*1000+IDAY                                     0174
      CALL GETBET(JSYD,MAXNUM,NUM,NOMTIM,                                0175
     *            ISCAN1,ISTIM1,ISMIL1,IBET1,                            0176
     *            ISCAN2,ISTIM2,ISMIL2,IBET2)                            0177
      IF(NUM.EQ.0) GO TO 40                                              0178
      DO 30 J=1,NUM                                                      0179
C     COMPUTE BETA TIME USING THE I INDEX TO COUNT THE HOURS OF EACH DAY
      PTIME=FTIME(NOMTIM(J))+(I-1)*24.0                                  0180
C     IF TIME LIES OUTSIDE SET INTERVAL FOR ORBIT COMPUTATION, EXIT.
      IF(PTIME.LT.BTIME.OR.PTIME.GT.ETIME)GO TO 30                       0181
C
C     THE BETA PAIRS ARE TAKEN ONE AT A TIME USING THE INDEX J AND ALL
C     THE LANDMARKS ARE LOOKED THROUGH USING THE INDEX K.
C
      DO 25 K=1,NUMLND                                                   0182
C     CHECK EACH LANDMARK OR STAR FOR MATCH.
C     IF IT IS A STAR, DISCARD.
      IF(MOD(LCODE(K),100).GE.5)GO TO 25                                 0183
C     IF THE TIMES DON'T MATCH, DISCARD.
      IF(PTIME.NE.PTIMLN(K))GO TO 25                                     0184
C     IF LANDMARK HAD PRIOR MATCH, DISCARD.
      IF(ILNDBT(K).EQ.1)GO TO 25                                         0185
C     OTHERWISE LINEARLY INTERPOLATE OR EXTRAPOLATE MATCHING BETA
C     AND TIME VALUES AND SET MATCH ARRAY ILNDBT EQUAL TO ONE AT INDEX K
      IF(IBET1(J).LT.0) IBET1(J)=(IBET1(J).AND.'37777777)+NEGBET         0186
      IF(IBET2(J).LT.0) IBET2(J)=(IBET2(J).AND.'37777777)+NEGBET         0187
         BETDIF=IBET2(J)-IBET1(J)                                        0188
      IF(BETDIF.GT.BETCMP)BETDIF=BETDIF-2.0D0*BETCMP                     0189
      IF(BETDIF.LT.-BETCMP)BETDIF=BETDIF+2.0D0*BETCMP                    0190
      ISCNLN=(IROUND(XLINLM(K))+3)/8-ISCAN1(J)+1
      QTIME1=FTIME(ISTIM1(J))+(I-1)*24.0D0
      QTIME2=FTIME(ISTIM2(J))+(I-1)*24.0D0
      QDTIME=QTIME2-QTIME1
      QTM1ML=ISMIL1(J)/360000.0D0
      QTM2ML=ISMIL2(J)/360000.0D0
      QDTIME=(QTM2ML-QTM1ML)+QDTIME
      QTIME1=QTIME1+QTM1ML
      QTMPSL=QDTIME/(ISCAN2(J)-ISCAN1(J))
      QBTPSL=BETDIF/(ISCAN2(J)-ISCAN1(J))
      QPCTLN=(XELELM(K)-1.0)/(TOTELE-1.0)*(DEGELE/360.0)
      T(K)=QTIME1+QTMPSL*(ISCNLN+QPCTLN)
      BETA(K)=(IBET1(J)+QBTPSL*(ISCNLN))*RDPBT
      T(K)=T(K)+QTMPSL*BETA(K)/TWPI
C     LANDMARK POINTER ARRAY WHICH IS USED FOR RESIDUAL COMPUTATION.
      ILNDBT(K)=1                                                        0197
   25 CONTINUE                                                           0198
   30 CONTINUE                                                           0199
C     INCREMENT DAY WHILE ACCOUNTING FOR LEAPYEAR
```

```
      IDAY=IDAY+1
      IF(IDAY.LE.365)GO TO 40
      IF(IDAY.EQ.366.AND.MOD(IYR,4).EQ.0)GO TO 40
      IDAY=1
      IYR=IYR+1
      IYR=MOD(IYR,100)
   40 CONTINUE
   50 CONTINUE
      RETURN
      END
```

SIZE 1317 02445

```
      SUBROUTINE ANGORB(ISYD,NUMLND,MEANOM,PTIME,XLAT,XLON,
     *    XLIN,XELE,ILNDBT,BETA,V,T,NV,LNDPNT,IORBPL)                   0205
C
C     WRITTEN 04/17/78 BY DENNIS PHILLIPS TO SET UP GEOMETRY TO FIND
C     ORBITAL POSITION ANGLE, I. E. TRUE ANAMOLY+OFFSET POSITON,
C     IN THE ORBITAL PLANE OF THE SATELLITE.
C     MODIFICATION BY DENNIS PHILLIPS ON MAY 2, 1979 TO ACCOUNT FOR THE 0209
C     VARIATION OF THE SATELLITE'S HEIGHT.                              0210
C     MODIFICATION 07/17/78 BY DENNIS PHILLIPS TO DETERMINE
C     ORIENTATION OF THE ORBIT PLANE FROM LANDMARK MEASUREMENTS IN      0212
C     ADDITION TO THE ALONG-PLANE ORBIT PARAMETER DETERMINATION.        0213
C
C     INPUTS:  ISYD. SATELLITE ID.  YEAR DAY
C              NUMLND.  NUMBER OF LANDMARKS
C              MEANOM  MEAN ANAMOLY OF SATELLITE POSITION AT  EPIC
C              PTIME.  PICTURE START TIME OF LANDMARK
C              XLAT.  LATITUDE OF LANDMARK
C              XLON.  LONGITUDE OF LANDMARK
C              XLIN.  LINE NUMBER OF LANDMARK
C              XELE.  ELEMENT NUMBER OF LANDMARK
C              ILNDBT.  LANDMARK-BETA CORRESPONDENCE FLAG
C              BETA.  BETA ANGULAR SWEEP TO PICTURE START IN RADIANS
C              T.  TIME OF LANDMARK MEASUREMENT
C              IORBPL.  ORBIT PLANE CALCULATION FLAG
C     OUTPUTS:  V.  THE TRUE ANAMOLIES+OFFSET
C               NV.  NUMBER OF LANDMARKS WITH CORRESPONDING BETA VALUES
C               LNDPNT.  INDEX POINTING TO LANDMARKS WITH MATCHING BETAS
C               ORBINC.  ORBIT INCLINATION IN NAVCOM COMMON
C               ASNODE.  ASCENDING NODE IN NAVCOM COMMON.
C
      INTEGER LNDPNT(1)                                                 0214
      INTEGER ILNDBT(1)                                                 0215
      REAL MEANOM
      REAL PTIME(1),XLAT(1),XLON(1),XLIN(1),XELE(1)                     0216
      DIMENSION ESTST1(100),ESTST2(100),ESTST3(100)                     0217
      DOUBLE PRECISION XCOR,YCOR                                        0218
      DOUBLE PRECISION BETA(1),V(1),T(1)                                0219
      DOUBLE PRECISION COSB,SINB,BETAR,TEMP,YCSNST,XCSNST,XNORM         0220
      DOUBLE PRECISION SAMTIM                                           0221
      DOUBLE PRECISION UX,UY,UZ,PSI
      DOUBLE PRECISION XSN,YSN,ZSN                                      0223
      COMMON/MINCOM/UX(80),UY(80),UZ(80),PSI(80),NP                     0225
      COMMON/NAVCOM/NAVDAY,LINTOT,DEGLIN,IELTOT,DEGELE,SPINRA,IETIMY,IET0226
     1IMH,SEMIMA,OECCEN,ORBINC,PERHEL,ASNODE,NOPCLN,DECLIN,RASCEN,PICLIN0227
     2,PRERAT,PREDIR,PITCH,YAW,ROLL,SKEW                                0228
      COMMON/NAVINI/                                                    0229
     1  EMEGA,AB,ASQ,BSQ,R,RSQ,                                         0230
     2  RDPDG,                                                          0231
     3  NUMSEN,TOTLIN,RADLIN,                                           0232
     4  TOTELE,RADELE,PICELE,                                           0233
     5  CPITCH,CYAW,CROLL,                                              0234
     6  PSKEW,                                                          0235
     7  RFACT,ROASIN,TMPSCL,                                            0236
```

```
     8  D11,D12,D13,D21,D22,D23,D31,D32,D33,                                    0237
     9  GAMMA,GAMDOT,                                                           0238
     A  ROTM11,ROTM13,ROTM21,ROTM23,ROTM31,ROTM33,                              0239
     B  PICTIM,XREF                                                             0240
        COMMON/BETCOM/IAJUST,IBTCON,NEGBET,ISEANG                               0241
        DATA HH/42165.0/                                                        0242
        DATA PI/3.14159265/                                                     0243
        DATA IENT/1/
C       SET UP PARAMETERS BEFORE LANDMARK LOOP
        IYD=MOD(ISYD,100000)                                                    0244
        NV=0                                                                    0245
C       INIALIZE ORBIT PARAMETERS IF THE SEMI MAJOR AXIS IS AVAILABLE.
        NAVDAY=NAVDAY-1
        IF(SEMIMA.GT.42000.0)CALL STVEC(0.0,MEANOM,XC,YC,ZC)
        NAVDAY=NAVDAY+1
        SENANG=FLALO(ISEANG)*RDPDG                                              0247
C       THIS IS THE LANDMARK LOOP.  FOR EACH LANDMARK WITH MATCHING BETA
C       VALUES AN ANGULAR POSITION OF THE SATELLITE IN ITS ORBITAL PLANE
C       IS COMPUTED.
        DO 2 I=1,NUMLND                                                         0263
        IF(ILNDBT(I).EQ.-1)GO TO 2                                              0264
C       COUNT MATCHES BETWEEN LANDMARKS AND BETAS
        NV=NV+1                                                                 0265
C       TIME VALUES CORRESPONDING TO ORBITAL ANGULAR POSITIONS ARE ONLY
C       SORTED ON THE FIRST ENTRY.  OTHERWISE THESE VALUES GET SHUFFLED
C       AND LOST.
        IF(IENT.EQ.1)T(NV)=T(I)
        TIME=T(NV)
        SAMTIM=T(NV)
C       LANDMARK POINTER ARRAY WHICH IS USED FOR RESIDULA COMPUTATION.
        LNDPNT(NV)=I                                                            0269
C       PRECESS ATTITUDE
        CALL ATPREC(TIME,DEC,RAS)
C       SET UP SATELLITE CENTERED COORDINATE SYSTEM DETERMINED BY
C       DECLINATION AND RIGHT ASCENSION
        DEC=DEC*RDPDG
        RAS=RAS*RDPDG
        SR=SIN(RAS)                                                             0250
        CR=COS(RAS)                                                             0251
        SD=SIN(DEC)                                                             0252
        CD=COS(DEC)                                                             0253
        X1=-SR                                                                  0254
        Y1=CR                                                                   0255
        Z1=0.0                                                                  0256
        X2=-SD*CR                                                               0257
        Y2=-SD*SR                                                               0258
        Z2=CD                                                                   0259
        X3=CD*CR                                                                0260
        Y3=CD*SR                                                                0261
        Z3=SD                                                                   0262
C       FIND VECTOR FROM EARTH'S CENTER TO LANDMARK IN INERTIAL COODINATES
C       TO LANDMARK.
        JTIME=ITIME(TIME)                                                       0271
```

```
      RA=RAERAC(IYD,JTIME,XLON(I))*RDPDG                                 Ø272
      YLAT=XLAT(I)*RDPDG                                                 Ø273
      YLAT=GEOLAT(YLAT,1)                                                Ø274
      TANLAT=TAN(YLAT)**2                                                Ø275
      RR=SQRT((1.Ø+TANLAT)/(BSQ+ASQ*TANLAT))*AB                          Ø276
      X=COS(RA)*COS(YLAT)*RR                                             Ø277
      Y=SIN(RA)*COS(YLAT)*RR                                             Ø278
      Z=SIN(YLAT)*RR                                                     Ø279
C     GET VECTOR FROM EARTH TO SUN
      CALL SUNVEC(IYD,SAMTIM,XSN,YSN,ZSN)                                Ø28Ø
C     SET SATELLITE VECTOR TO ZERO FOR FIRST ITERATION
      XSAT=Ø.Ø                                                           Ø281
      YSAT=Ø.Ø                                                           Ø282
      ZSAT=Ø.Ø                                                           Ø283
      H=HH                                                               Ø284
      IF(SEMIMA.LT.1.Ø)GO TO 1                                           Ø285
C     OTHERWISE COMPUTE THE SATELLITE VECTOR IN INERTIAL COORDINATES
C     AND USE HEIGHT TO IMPROVE POSITION ESTIMATE.
      CALL STVEC(TIME,MEANOM,XSAT,YSAT,ZSAT)
      H=SQRT(XSAT**2+YSAT**2+ZSAT**2)                                    Ø287
    1 CONTINUE                                                           Ø292
C     COMPUTE X, Y COORDINATES OF THE VECTOR FROM THE SATELLITE TO THE
C     SUN IN OUR SATELLITE CENTERED COORDINATE SYSTEM.
      XCSNST=X1*(-XSN-XSAT)+Y1*(-YSN-YSAT)                               Ø293
      YCSNST=X2*(-XSN-XSAT)+Y2*(-YSN-YSAT)+Z2*(-ZSN-ZSAT)                Ø294
C     THE NEXT SIX STATEMENTS YIELD THE POINTING DIRECTION OF THE
C     SPIN SCAN CAMERA IN A COORDINATE SYSTEM WITH THE Z-AXIS COINCIDING
C     WITH THE VECTOR OPPOSITE THE SPIN AXIS AND THE X-AXIS
C     PERPENDICULAR TO THE Z-AXIS AT AN ANGLE EQUAL TO SENANG FROM THE
C     SUN PULSE DETECTOR.
      YLIN=(PICLIN-XLIN(I))*RADLIN                                       Ø295
      CLIN=COS(YLIN)                                                     Ø296
      SLIN=SIN(YLIN)                                                     Ø297
      U=ROTM11*CLIN+ROTM13*SLIN                                          Ø298
      VV=ROTM21*CLIN+ROTM23*SLIN                                         Ø299
      W=ROTM31*CLIN+ROTM33*SLIN                                          Ø3ØØ
C     THE ANGULAR SWEEP DISTANCE OF THE SPIN SCAN CAMERA FROM THE SUN
C     AT THE TIME WHEN THIS PARTICULAR LANDMARK IS BEING VIEWED.  THE
C     (VV,U) TERM ACCOUNTS FOR THE ROLL AND YAW MISALIGNMENT EFFECTS.
      BETAR=BETA(I)-SENANG+XELE(I)*RADELE-ATAN2(VV,U)                    Ø3Ø1
C     THE NEXT 8 STATEMENTS COMPUTE A UNIT VECTOR POINTING AT THE
C     LANDMARK IN THE SATELLITE CENTERED COORDINATE SYSTEM.
      COSB=DCOS(BETAR)                                                   Ø3Ø2
      SINB=DSIN(BETAR)                                                   Ø3Ø3
      TEMP=COSB*XCSNST+SINB*YCSNST                                       Ø3Ø4
      YCSNST=-SINB*XCSNST+COSB*YCSNST                                    Ø3Ø5
      XCSNST=TEMP                                                        Ø3Ø6
      XNORM=DSQRT((1.ØDØ-W**2)/(XCSNST**2+YCSNST**2))                    Ø3Ø7
      XCSNST=XCSNST*XNORM                                                Ø3Ø8
      YCSNST=YCSNST*XNORM                                                Ø3Ø9
C     FINALLY, WE HAVE THE POINTING DIRECTION OF THE SPIN SCAN CAMERA
C     IN INERTIAL COORDINATES.
      XLNDST=-XCSNST*X1-YCSNST*X2-W*X3                                   Ø31Ø
```

```
      YLNDST=-XCSNST*Y1-YCSNST*Y2-W*Y3
      ZLNDST=-XCSNST*Z1-YCSNST*Z2-W*Z3
C     COMPUTE SLANT RANGE (SLTRNG) TO LANDMARK FROM SATELLITE.
      COSANG=XLNDST*X+YLNDST*Y+ZINDST*Z
      SLTRNG=-COSANG+SQRT(H**2-RR**2+COSANG**2)
C     THE INTERSECTION OF THE RAY EXTENDING FROM THE LANDMARK WITH  A
C     SPHERE CENTERED AT THE EARTH'S CENTER AND RADIUS EQUAL TO H.
      ESTST1(NV)=X+XLNDST*SLTRNG
      ESTST2(NV)=Y+YLNDST*SLTRNG
      ESTST3(NV)=Z+ZLNDST*SLTRNG
      IF(IORBPL.NE.1)GO TO 2
C     THIS VECTOR IS NORMALIZED AND THE ANGLES PSI(I) ARE SET WHEN THE
C     ORIENTATION OF THE ORBITAL PLANE IS TO BE CALCULATED.
      YNORM=1.0/SQRT(ESTST1(NV)**2+ESTST2(NV)**2+ESTST3(NV)**2)
      UX(NV)=ESTST1(NV)*YNORM
      UY(NV)=ESTST2(NV)*YNORM
      UZ(NV)=ESTST3(NV)*YNORM
      PSI(NV)=PI/2.0
    2 CONTINUE
      IENT=0
C     BRANCH IF THE INCLINATION AND ASCENDING NODE ARE NOT TO BE
C     CALCULATED.
      IF(IORBPL.NE.1)GO TO 3
      NP=NV
      NOPCLN=0
C     CHANGE ON MARCH 22, 1980 BY DENNIS PHILLIPS TO REPLACE CALL TO
C     MINMIZ, THE OLD WAY TO COMPUTE ATTITUDE.  INSTEAD FINDAT, A
C     SUBROUTINE USING A VARIATION OF NEWTON'S METHOD, IS CALLED TO
C     COMPUTE THE ORIENTATION OF THE ORBITAL PLANE.
      CALL FINDAT(NOPCLN,RADLIN,ORBNC,ASNDE,PCLN)
      ORBINC=90.0+ORBNC
      ASNODE=ASNDE+270.0
      IF(ASNODE.GE.360.0)ASNODE=ASNODE-360.0
    3 CONTINUE
      ASN=ASNODE*RDPDG
      CASN=COS(ASN)
      SASN=SIN(ASN)
      XINC=ORBINC*RDPDG
      CINC=COS(XINC)
      SINC=SIN(XINC)
      U1=CASN
      V1=SASN
      W1=0.0
      U2=-SASN*CINC
      V2=CASN*CINC
      W2=SINC
C
C     COMPUTE THE ORBITAL ANGLE POSITIONS OF THE SATELLITE AROUND
C     THE ORBIT PERPENDICULAR.  CARE IS TAKEN TO AVOID A WRONG MULTIPLE
C     OF 2*PI.
C
      DO 10 I=1,NV
      XCOR=U1*ESTST1(I)+V1*ESTST2(I)+W1*ESTST3(I)
```

```
      YCOR=U2*ESTST1(I)+V2*ESTST2(I)+W2*ESTST3(I)
      V(I)=DATAN2(YCOR,XCOR)
      IF(I.NE.1)GO TO 5
      ANG2=V(1)
      TIME=T(1)
      GO TO 7
    5 CONTINUE
      K=IROUND((T(I)-T(1))/24.0+(ANG2-V(I))/(2.0*PI))
      V(I)=V(I)+2.0*PI*K
    7 CONTINUE
   10 CONTINUE
      RETURN
      END
```

SIZE 1456 02660

```
      SUBROUTINE STVEC(SAMTIM,MEANOM,X,Y,Z)
C  COPIED 0480 BY DENNIS PHILLIPS FROM SATVEC.
C     STVEC COMPUTES THE INERTIAL POSITION (X, Y, Z) AT TIME SAMTIM.
C     INPUTS:  SAMTIM  ZULU TIME IN HOURS FOR WHICH POSITION IS REQUIRED
C              MEANOM  MEAN ANAMOLY OF SATELLITE POSITION AT  EPIC
C     OUTPUTS:  X, Y, Z   THE INERTIAL POSITION OF THE SATELLITE AT ZULU
C     TIME SAMTIM.
      DOUBLE PRECISION RDPDG,RE,GRACON,DIFTIM,EACAN1,ECANOM,XMANOM
      DOUBLE PRECISION DABS,DSQRT,DSIN,DCOS
      REAL MEANOM
      COMMON/NAVCOM/NAVDAY,LINTOT,DEGLIN,IELTOT,DEGELE,SPINRA,IETIMY,IET
     1IMH,SEMIMA,OECCEN,ORBINC,PERIGE,ASNODE,NOPCLN,DECLIN,RASCEN,PICLIN
     2,PRERAT,PREDIR,PITCH,YAW,ROLL,SKEW
      DATA NAVSAV/0/
      DATA GRACON,RE/.07436574D0,6378.388/
      DATA RDPDG/.0174532925D0/
C     STVEC WILL RE-INITIALIZE ORBIT PARAMETERS IF NAVDAY IS CHANGED.
      IF(NAVDAY.EQ.NAVSAV) GO TO 10
      O=RDPDG*ORBINC
      P=RDPDG*PERIGE
      A=RDPDG*ASNODE
      SO=SIN(O)
      CO=COS(O)
      SP=SIN(P)*SEMIMA
      CP=COS(P)*SEMIMA
      SA=SIN(A)
      CA=COS(A)
      PX=CP*CA-SP*SA*CO
      PY=CP*SA+SP*CA*CO
      PZ=SP*SO
      QX=-SP*CA-CP*SA*CO
      QY=-SP*SA+CP*CA*CO
      QZ=CP*SO
      SROME2=SQRT(1.0-OECCEN)*SQRT(1.0+OECCEN)
      XMMC=GRACON*RE*DSQRT(RE/SEMIMA)/SEMIMA
      EPSILN=1.0E-8
C   COMPUTE THE DIFFERENCE IN TIMES OF (NAVDAY,0)-(IETIMY,IETIMH).
C     THIS DIFFERENCE IS COMPUTED IN MINUTES.
      IEY=MOD(IETIMY/1000,100)
      INY=MOD(NAVDAY/1000,100)
      IYRMN=0
      IF(IEY.EQ.INY)GO TO 5
      IF(INY.LT.IEY)GO TO 3
      IE=INY-1
      DO 2 I=IEY,IE
      IYRMN=IYRMN+365*1440
      IF(MOD(I,4).EQ.0)IYRMN=IYRMN+1440
    2 CONTINUE
      GO TO 5
    3 CONTINUE
      IE=IEY-1
      DO 4 I=INY,IE
      IYRMN=IYRMN-365*1440
```

```
      IF(MOD(I,4).EQ.0)IYRMN=IYRMN-1440
    4 CONTINUE
    5 CONTINUE
      IED=MOD(IETIMY,1000)
      IND=MOD(NAVDAY,1000)
      IYRMN=IYRMN+(IND-IED)*1440
      TE=FLOAT(IYRMN)-FTIME(IETIMH)*60.0
   10 DIFTIM=SAMTIM*60.0+TE
      XMANOM=XMMC*DIFTIM+MEANOM*RDPDG
      ECANM1=XMANOM
      DO 20 I=1,20
      ECANOM=XMANOM+OECCEN*DSIN(ECANM1)
      IF(DABS(ECANOM-ECANM1).LT.EPSILN)GO TO 30
   20 ECANM1=ECANOM
   30 XOMEGA=DCOS(ECANOM)-OECCEN
      YOMEGA=SROME2*DSIN(ECANOM)
      X=XOMEGA*PX+YOMEGA*QX
      Y=XOMEGA*PY+YOMEGA*QY
      Z=XOMEGA*PZ+YOMEGA*QZ
      RETURN
      END
```

SIZE 345 00531

```
      SUBROUTINE ORBPR(V,T,N,MEANOM,ISEMI,EPTIME)
C     WRITEEN BY DENNIS PHILLIPS OF SCIENTIFIC PROGRAMMING AND APPLIED
C     MATHEMATICS, INC.
C     ORBPR CONVERTS ANGULAR MEASUREMENTS (TRUE ANAMOLY+OFFSET) IN THE O
C     ORBPR CONVERTS ANGULAR MEASUREMENTS (TRUE ANAMOLY+OFFSET) IN THE
C     ORBITAL PLANE WHICH HAVE ASSOCIATED TIME TAGS INTO KEPLERIAN
C     ORBITAL ELEMENTS.
C     INPUTS:  V. TRUE ANAMOLY+OFFSET ANGLES.
C              T.  TIME TAGS OF ANGLES
C              N.  NUMBER OF ANGLES.
C              ISEMI.  NUMBER OF KEPLERIAN ORBITAL PARAMETERS TO BE
C                      COMPUTED.
C              EPTIME.  EPIC TIME FOR KEPLERIAN ORBITAL ELEMENTS.
C     OUTPUTS:  MEANOM.  MEAN ANAMOLY OF ORBIT AT EPIC TIME.
C               SEMIMA.  ORBIT'S SEMIMAJOR AXIS IN NAVCOM COMMON BLOCK.
C               OECCEN.  ORBIT'S ECCENTRICITY IN NAVCOM COMMON BLOCK
C               PERIGEE.  ORBIT'S PERIGEE IN NAVCOM COMMON BLOCK.
      DOUBLE PRECISION T(1),V(1)
      REAL MEANOM
      COMMON/NAVCOM/NAVDAY,LINTOT,DEGLIN,IELTOT,DEGELE,SPINRA,IETIMY,IET
     1IMH,SEMIMA,OECCEN,ORBINC,PERIGE,ASNODE,NOPCLN,DECLIN,RASCEN,PICLIN
     2,PRERAT,PREDIR,PITCH,YAW,ROLL,SKEW
      DATA GRACON,RE/.07436574D0,6378.388/
      DATA PI,RDPDG/3.14159265,.01745329252/
C     COMPLETELY REWRITTEN SEPTEMBER 25, 1979 BY DENNIS PHILLIPS OF SPAA
C     M TO CLARIFY  THE CODING METHODS AND TO CONFORM TO A STRUCTURED
C     PROGRAMMING PHILOSOPHY.  ALSO, THE METHOD FOR COMPUTINGE THE
C     SEMIMAJOR AXIS IS MUCH IMPROVED.
      IF(ISEMI.NE.4)GO TO 20
C     COMPUTE ALL FOUR ALONG-TRACK PARAMETERS BY LINEAR REGRESSION.
      CALL FOURCF(C1,C2,C3,C4,V,T,N)
      SEMIMA=(GRACON*C2*60.0)**(2.0/3.0)*RE
      CALL ORBTHR(C1,C2,C3,C4,EPTIME,MEANOM)
      RETURN
   20 IF(ISEMI.NE.3)GO TO 30
C     COMPUTE ORBIT'S MEAN ANAMOLY, ECCENTRICITY, AND ARGUMENT OR PERIGE
      C2=(SEMIMA/RE)**(3.0/2.0)/(60.0*GRACON)
      CALL THRCOF(C1,C2,C3,C4,V,T,N)
      CALL ORBTHR(C1,C2,C3,C4,EPTIME,MEANOM)
      RETURN
   30 IF(ISEMI.NE.2)GO TO 40
C     COMPUTE ORBIT'S SEMIMAJOR AXIS AND MEAN ANAMOLY WHILE ADJUSTING
C     FOR THE PERTURBATIVE EFFECTS OF A NONZERO ECCENTRICITY.
      C3=0.0
      C4=0.0
      DO 36 J=1,4
      SME=0.0
      SMESQ=0.0
      SMT=0.0
      SMET=0.0
      XN=FLOAT(N)
      DO 35 I=1,N
      ANG=V(I)
```

```
      TT=T(I)-C3*SIN(ANG)-C4*COS(ANG)
      SME=SME+ANG
      SMESQ=SMESQ+ANG**2
      SMT=SMT+TT
   35 SMET=SMET+ANG*TT
      DET=1.0/(XN*SMESQ-SME*SME)
      C2=(SMET*XN-SME*SMT)*DET
      C3=-2.0*OECCEN*C2*COS(PERIGE*RDPDG)
      C4=2.0*C2*OECCEN*SIN(PERIGE*RDPDG)
   36 CONTINUE
      C1=(SMT*SMESQ-SMET*SME)*DET
      SEMIMA=(GRACON*C2*60.0)**(2.0/3.0)*RE
      MEANOM=(EPTIME-C1)/(C2*RDPDG)-PERIGE
      MEANOM=AMOD(MEANOM,360.0)
      IF(MEANOM.LT.0.0)MEANOM=MEANOM+360.0
      RETURN
   40 IF(ISEMI.NE.1)RETURN
C     COMPUTE ORBIT'S MEAN ANAMOLY.
      C2=(SEMIMA/RE)**(3.0/2.0)/(60.0*GRACON)
      C3=-2.0*OECCEN*C2*COS(PERIGE*RDPDG)
      C4=2.0*C2*OECCEN*SIN(PERIGE*RDPDG)
      CALL ONEPAR(C2,C3,C4,V,T,N,EPTIME,PERIGE,MEANOM)
      RETURN
      END
```

SIZE 321 00501

```
      SUBROUTINE FOURCF(C1,C2,C3,C4,V,T,N)
C     WRITTEN BY DENNIS PHILLIPS OF SCIENTIFIC PROGRAMMING AND APPLIED
C     MATHEMATICS, INC.
C     FOURCF USES LINEAR REGRESSION TO MINIMIZE THE SUM OVER I OF
C     (C1 + C2*V(I) + C3*COS(V(I)) + C4*SIN(V(I)) - T(I))**2.
C     INPUTS:  TRUE ANAMOLY+OFFSET ANGULAR POSITIONS.
C              T.  ANGULAR TIME TAGS.
C              N.  NUMBER OF ANGULAR POSITIONS.
C     OUTPUTS:  C1, C2, C3 AND C4  LINEAR PLUS SINE WAVE CONSTANTS OF
C               ORBITAL MOTION.
      DOUBLE PRECISION A(4,4),B(4,4),C(4),D(4),V(1),T(1)
      DO 5 I=1,4
      JS=I
      D(I)=0.0
      DO 5 J=JS,4
    5 A(I,J)=0.0
C     SUM REGRESSION TERMS INTO THE APPROPIATE ELEMENTS OF THE MATRIX A.
      XK=N
      A(1,1)=XK
      DO 10 I=1,N
      ANG=V(I)
      TT=T(I)
      CANG=COS(ANG)
      SANG=SIN(ANG)
      A(1,2)=A(1,2)+ANG
      A(2,2)=A(2,2)+ANG**2
      A(1,3)=A(1,3)+SANG
      A(1,4)=A(1,4)+CANG
      A(2,3)=A(2,3)+ANG*SANG
      A(2,4)=A(2,4)+ANG*CANG
      A(3,3)=A(3,3)+SANG**2
      A(3,4)=A(3,4)+SANG*CANG
      A(4,4)=A(4,4)+CANG**2
      D(1)=D(1)+TT
      D(2)=D(2)+TT*ANG
      D(3)=D(3)+SANG*TT
      D(4)=D(4)+TT*CANG
   10 CONTINUE
C     INVERT A
      CALL INVE(A,B)
      DO 20 I=1,4
      C(I)=0.0
      DO 20 K=1,4
   20 C(I)=C(I)+B(I,K)*D(K)
      C1=C(1)
      C2=C(2)
      C3=C(3)
      C4=C(4)
      RETURN
      END
```

```
      SUBROUTINE INVE(A,B)
C     WRITTEN BY DENNIS PHILLIPS OF SCIENTIFIC PROGRAMMING AND APPLIED
C     MATHEMATICS, INC.
C     INVERTS FOUR BY FOUR REGRESSION MATRIX.
C     INPUTS: FOUR BY FOUR MATRIX A.
C     OUTPUTS:  B, THE INVERSE OF A.
C     A IS PARTITIONED INTO FOUR TWO BY TWO SUBMATRICES AND THEN THE
C     RESULTANT TWO BY TWO MATRIX CONSISTING OF SUBMATRICES IS INVERTED
C     BY USING GAUSSIAN ELIMINATION.
      DOUBLE PRECISION A(4,4),B(4,4),C(2,2),D(2,2)
      DOUBLE PRECISION DET,TEMP
      DET=1.0D0/(A(1,1)*A(2,2)-A(1,2)*A(1,2))
      B(1,1)=A(2,2)*DET
      B(1,2)=-A(1,2)*DET
      B(2,1)=B(1,2)
      B(2,2)=A(1,1)*DET
      C(1,1)=B(1,1)*A(1,3)+B(1,2)*A(2,3)
      C(2,1)=B(2,1)*A(1,3)+B(2,2)*A(2,3)
      C(1,2)=B(1,1)*A(1,4)+B(1,2)*A(2,4)
      C(2,2)=B(2,1)*A(1,4)+B(2,2)*A(2,4)
      B(3,1)=-A(1,3)*B(1,1)-A(2,3)*B(2,1)
      B(4,1)=-A(1,4)*B(1,1)-A(2,4)*B(2,1)
      B(3,2)=-A(1,3)*B(1,2)-A(2,3)*B(2,2)
      B(4,2)=-A(1,4)*B(1,2)-A(2,4)*B(2,2)
      D(1,1)=A(3,3)-A(1,3)*C(1,1)-A(2,3)*C(2,1)
      D(1,2)=A(3,4)-A(1,3)*C(1,2)-A(2,3)*C(2,2)
      D(2,1)=A(3,4)-A(1,4)*C(1,1)-A(2,4)*C(2,1)
      D(2,2)=A(4,4)-A(1,4)*C(1,2)-A(2,4)*C(2,2)
      DET=1.0D0/(D(1,1)*D(2,2)-D(1,2)*D(2,1))
      B(3,3)=D(2,2)*DET
      B(3,4)=-D(1,2)*DET
      B(4,3)=B(3,4)
      B(4,4)=D(1,1)*DET
      TEMP=B(3,1)
      B(3,1)=B(3,3)*B(3,1)+B(3,4)*B(4,1)
      B(4,1)=B(4,3)*TEMP+B(4,4)*B(4,1)
      TEMP=B(3,2)
      B(3,2)=B(3,3)*B(3,2)+B(3,4)*B(4,2)
      B(4,2)=B(4,3)*TEMP+B(4,4)*B(4,2)
      B(1,3)=B(3,1)
      B(1,4)=B(4,1)
      B(2,3)=B(3,2)
      B(2,4)=B(4,2)
      B(1,1)=B(1,1)-C(1,1)*B(3,1)-C(1,2)*B(4,1)
      B(1,2)=B(1,2)-C(1,1)*B(3,2)-C(1,2)*B(4,2)
      B(2,1)=B(1,2)
      B(2,2)=B(2,2)-C(2,1)*B(3,2)-C(2,2)*B(4,2)
      RETURN
      END
```

SIZE 271 00417

```
      SUBROUTINE ORBTHR(C1,C2,C3,C4,EPTIME,MEANOM)
C     PULLED OUT OF ORBPR BY DENNIS PHILLIPS ON MAY 2, 1979
C     CHANGED BY DENNIS PHILLIPS ON MAY 11, 1979
C     ANAMOLY AND EQUALS ONE WHEN THE COEFFICIENTS ARE COMPUTED FROM THE
C     ECCENTRIC ANAMOLY.
C     COMPUTE ECCENTRICITY, ARGUMENT OF PERIGEE AND MEAN ANAMOLY.
C     COMPUTES ECCENTRICITY, ARGUMENT OF PERIGEE AND MEAN ANAMOLY FROM
C     C1, C2, C3 AND C4 COEFFICIENTS.
C     INPUTS:  C1, C2, C3 AND C4 COEFFICIENTS.
C              EPTIME.  THE EPIC TIME OF ORBIT COMPUTATION.
C     OUTPUTS: OECCEN.  ORBIT ECCENTRICITY.
C              PERIGE.  ORBIT'S PERIGEE
C              MEANOM.  ORBIT'S MEAN ANAMOLY.
C
      REAL MEANOM
      COMMON/NAVCOM/NAVDAY,LINTOT,DEGLIN,IELTOT,DEGELE,SPINRA,IETIMY,IET
     1IMH,SEMIMA,OECCEN,ORBINC,PERIGE,ASNODE,NOPCLN,DECLIN,RASCEN,PICLIN
     2,PRERAT,PREDIR,PITCH,YAW,RCLL,SKEW
      COMMON/NAVINI/
     1  EMEGA,AB,ASQ,BSQ,R,RSQ,
     2  RDPDG,
     3  NUMSEN,TOTLIN,RADLIN,
     4  TOTELE,RADELE,PICELE,
     5  CPITCH,CYAW,CROLL,
     6  PSKEW,
     7  RFACT,ROASIN,TMPSCL,
     8  B11,B12,B13,B21,B22,B23,B31,B32,B33,
     9  GAMMA,GAMDOT,
     A  ROTM11,ROTM13,ROTM21,ROTM23,ROTM31,ROTM33,
     B  PICTIM,XREF
      OECCEN=SQRT(C3**2+C4**2)/(2.Ø*C2)
      PERIGE=Ø.Ø
      PERIGE=ATAN2(C4,-C3)/RDPDG
      PERIGE=AMOD(PERIGE,36Ø.Ø)
      IF(PERIGE.LT.Ø.Ø)PERIGE=PERIGE+36Ø.Ø
   4Ø CONTINUE
      MEANOM=(EPTIME-C1)/(C2*RDPDG)-PERIGE
      MEANOM=AMOD(MEANOM,36Ø.Ø)
      IF(MEANOM.LT.Ø.Ø)MEANOM=MEANOM+36Ø.Ø
      RETURN
      END
```

SIZE 78 ØØ116

```
      SUBROUTINE THRCOF(C1,C2,C3,C4,E,T,N)
C
C     WRITTEN MAY 2,1980 BY DENNIS PHILLIPS FOR NESS.
C     THRCOF FITS A SINE WAVE PLUS CONSTANT TO THE CURVE T(I)-C2*E(I).
C     EFFECTIVELY A LEAST SQUARES FIT IS PERFORMED TO FIND C1, C3 AND C4
C     IN THE EXPRESSION C1+C2*E(I)+C3*SIN(E(I))+C4*COS(E(I))=T(I).
C     INPUTS:  E.  THE TRUE ANAMOLY + OFFSET SATELLITE POSITION ARRAY.
C              T.  THE TIME OF THE ESTIMATED SATELLITE POSITIONS.
C              N.  THE NUMBER OF ESTIMATED SATELLITE POSITIONS.
C              C2.  THE COEFFICIENT DETERMINING THE SATELLITE'S MEAN
C     OUTPUTS:  C1, C3 AND C4.  THE CONSTANT PLUS SINE WAVE COEFFICIENTS
C
      DOUBLE PRECISION E(1),T(1)
      SMSN=0.0
      SMCS=0.0
      SMSNSQ=0.0
      SMSNCS=0.0
      SMCSSQ=0.0
      SMCN=0.0
      SMSNCN=0.0
      SMCSCN=0.0
C     COLLECT REGRESSION COEFFICIENTS
      DO 10 I=1,N
      ANG=E(I)
      CN=T(I)-C2*ANG
      SN=SIN(ANG)
      CS=COS(ANG)
      SMSN=SMSN+SN
      SMCS=SMCS+CS
      SMSNSQ=SMSNSQ+SN*SN
      SMSNCS=SMSNCS+SN*CS
      SMCSSQ=SMCSSQ+CS*CS
      SMCN=SMCN+CN
      SMSNCN=SMSNCN+SN*CN
      SMCSCN=SMCSCN+CS*CN
   10 CONTINUE
      XN=N
C     INVERT THRREE BY THREE REGRESSION MATRIX USING CRAMER'S RULE.
      DET=XN*(SMSNSQ*SMCSSQ-SMSNCS**2)-SMSN*(SMSN*SMCSSQ-SMCS*SMSNCS)
     *+SMCS*(SMSN*SMSNCS-SMCS*SMSNSQ)
      DET=1.0/DET
      C1=(SMCN*(SMSNSQ*SMCSSQ-SMSNCS**2)-SMSNCN*(SMSN*SMCSSQ-SMCS*SMSNCS
     *)+SMCSCN*(SMSN*SMSNCS-SMCS*SMSNSQ))*DET
      C3=(XN*(SMSNCN*SMCSSQ-SMSNCS*SMCSCN)-SMSN*(SMCN*SMCSSQ-SMCS*SMCSCN
     *)+SMCS*(SMCN*SMSNCS-SMCS*SMSNCN))*DET
      C4=(XN*(SMSNSQ*SMCSCN-SMSNCN*SMSNCS)-SMSN*(SMSN*SMCSCN-SMCN*SMSNCS
     *)+SMCS*(SMSN*SMSNCN-SMCN*SMSNSQ))*DET
      RETURN
      END
```

SIZE 250 00372

```
      SUBROUTINE ONEPAR(C2,C3,C4,V,T,N,EPTIME,PERIGE,MEANOM)
C     WRITTEN BY DENNIS PHILLIPS OF SCIENTIFIC PROGRAMMING AND APPLIED
C     MATHEMATICS, INC.
C     COMPUTE THE ORBIT'S MEAN ANAMOLY BY REGRESSION.
C     INPUTS:  C2, C3 AND C4  LINEAR AND SINE WAVE CONSTANTS OF ORBITAL
C              MOTION.
C              V.  TRUE ANAMOLY + OFFSET ANGULAR POSITIONS.
C              T.  ANGULAR TIME TAGS.
C              N.  NUMBER OF ANGULAR POSITIONS.
C              EPTIME.  EPIC TIME OF DESIRED MEAN ANAMOLY POSITION.
C              PERIGE.  ARGUMENT OF PERIGEE.
C     OUTPUTS:  MEANOM: MEAN ANAMOLY.
      DOUBLE PRECISION V(1),T(1)
      REAL MEANOM
      DATA RDPDG/.01745329252/
      SUM=0.0
      XN=N
      DO 10 I=1,N
      ANG=V(I)
   10 SUM=SUM+T(I)-C2*ANG-C3*SIN(ANG)-C4*COS(ANG)
      C1=SUM/XN
      MEANOM=(EPTIME-C1)/(C2*RDPDG)-PERIGE
      MEANOM=AMOD(MEANOM,360.0)
      IF(MEANOM.LT.0.0)MEANOM=MEANOM+360.0
      RETURN
      END
```

SIZE 100 00144

```
      SUBROUTINE ORBRES(ISYD,IOUT,NUMLND,T,XLIN,XELE,BETA,MEANOM,
     *      LCODE,XLAT,XLON,LNDPNT)
C                                                                    Ø367
C     ADAPTED 1Ø/Ø5/78 BY DENNIS PHILLIPS FROM SUBROUTINE RESIDU     Ø368
C     OUTPUTS LINE AND ELEMENT RESIUDES FROM UPGORB                  Ø369
C
C     INPUTS:  ISYD.  SATELLITE ID.  YEAR  DAY
C              IOUT.  OUTPUT DEVICE.  ONE MEANS CRT.  TWO MEANS PRINTER.
C              NUMLND.  THE NUMBER OF LANDMARKS TO COMPUTE RESIDUALS FOR
C              T.  THE TIME OF EACH LANDMARK MEASUREMENT.
C              XLIN.  THE LINE NUMBER OF EACH LANDMARK MEASUREMENT.
C              XELE.  THE ELEMENT NUMBER OF EACH LANDMARK MEASUREMNT.
C              BETA.  THE SWEEP ANGLE UP TO THE BEGINNING OF THE SCAN
C                     LINE ON WHICH THE LANDMARK WAS MEASURED.
C              MEANOM  MEAN ANAMOLY OF SATELLITE POSITION AT  EPIC
C              LCODE.  THE LANDMARK CODE.
C              XLAT.  THE LATITUDE OF THE MEASURED LANDMARK.
C              XLON.  THE LONGITUDE OF THE MEASURED LANDMARK.
C              LNDPNT.  AN INDEX ARRAY POINTING AT LANDMARKS WITH
C                       ASSOCIATED BETA COUNTS.
C
      INTEGER LCODE(1),LNDPNT(1)                                     Ø37Ø
      REAL MEANOM
      REAL XLAT(1),XLON(1),XLIN(1),XELE(1)                           Ø371
      DOUBLE PRECISION T(1),BETA(1)                                  Ø372
      INTEGER LINE(44)                                               Ø373
      COMMON/NAVCOM/NAVDAY,LINTOT,DEGLIN,IELTOT,DEGELE,SPINRA,IETIMY,IET
     1IMH,SEMIMA,OECCEN,ORBINC,PERIGE,ASNODE,NOPCLN,DECLIN,RASCEN,PICLIN
     2,PRERAT,PREDIR,PITCH,YAW,ROLL,SKEW
      COMMON/NAVINI/
     1  EMEGA,AB,ASQ,BSQ,R,RSQ,
     2  RDPDG,
     3  NUMSEN,TOTLIN,RADLIN,
     4  TOTELE,RADELE,PICELE,
     5  CPITCH,CYAW,CROLL,
     6  PSKEW,
     7  RFACT,ROASIN,TMPSCL,
     8  B11,B12,B13,B21,B22,B23,B31,B32,B33,
     9  GAMMA,GAMDOT,
     A  ROTM11,ROTM13,ROTM21,ROTM23,ROTM31,ROTM33,
     B  PICTIM,XREF
      IF(NUMLND.EQ.Ø) GO TO 91                                       Ø374
C     RE-INITIALIZE STVEC WITH NEW ORBIT PARAMETERS
      NAVDAY=NAVDAY-1
      CALL STVEC(Ø.Ø,MEANOM,X,Y,Z)
      NAVDAY=NAVDAY+1
      CALL STVEC(Ø.Ø,MEANOM,X,Y,Z)
C
C     TP SENDS A HOLLERETH FIELD TO THE OUTPUT DEVICE IOUT.
C
      CALL TP(IOUT,132H LANDMARK RESIDUALS FROM UPGORB               Ø375
     *                                                               Ø376
     *                                                               Ø377
```

```
C     SEND SECOND HOLLERETH FIELD
      CALL TP(IOUT,132H  N SSYYDDD  HHMMSS   LCODE  LINDIF  ELEDIF LANDL
     *AT  LANDLON
     *                                                                  )
      INAV=1
      IYD=MOD(ISYD,100000)
      DO 10 I=1,NUMLND
C
C     MVCHAR MOVES A HOLLERETH FIELD INTO THE ARRAY LINE.
C
C     FIRST BLANK OUT THE FIELD
      CALL MVCHAR('BLA','BLA',LINE,1,132)
C     PLACE INTEGER HOLLERETH FIELD IN ARRAY
      CALL MVCHAR(I,'INT',LINE,3,1)
C     PLACE INTEGER HOLLERETH FIELD IN ARRAY
      CALL MVCHAR(ISYD,'INT',LINE,11,1)
      TIME=T(I)
      JTIME=ITIME(TIME)
C     PLACE INTEGER HOLLERETH FIELD IN ARRAY
      CALL MVCHAR(JTIME,'INT',LINE,19,1)
      K=LNDPNT(I)
C     PLACE INTEGER HOLLERETH FIELD IN ARRAY
      CALL MVCHAR(LCODE(K),'INT',LINE,27,1)
      ILINE=(XLIN(K)-1.0)/8.0 + 1.0
      PTIME=T(I)-TMPSCL*FLOAT(ILINE)
C     TRANSFORM LANDMARK'S LATITUDE AND LONGITUDE TO IMAGE COORDINATES
C     USING THE NEW ORBIT PARAMETERS.
      CALL CALRES(IYD,T(I),BETA(K),MEANOM,XLAT(K),XLON(K),YLIN,YELE)
C     CALCULATE THE DISTANCE BETWEEN THE CALCULATED AND MEASURED
C     IMAGE POSITIONS.
      DLIN=XLIN(K)-YLIN
      DELE=XELE(K)-YELE
C     TRANSFER RESIDUALS INTO A FLOATING HOLLERETH FIELD.
      CALL FF(8,2,DLIN,LINE,27)
      CALL FF(8,2,DELE,LINE,35)
C     TRANSFER THE LATITUDE AND LONGITUDE OF LANDMARK INTO AN INTEGER
      CALL MVCHAR(ILALO(XLAT(K)),'INT',LINE,51,1)
C     HOLLERETH FILE.
      CALL MVCHAR(ILALO(XLON(K)),'INT',LINE,60,1)
C     OUTPUT GENERATED HOLLERETH LINE.
      CALL TP(IOUT,LINE)
   10 CONTINUE
      RETURN
   91 CALL TQMES('NO LANDMARKS FOR SSYYDDD:$',ISYD)
      RETURN
      END
```

SIZE 419 00643

```
      SUBROUTINE CALRES(IYD,T,BETA,MEANOM,XLAT,XLON,XLIN,XELE)
C     WRITTEN BY DENNIS PHILLIPS OF SCIENTIFIC PROGRAMMING AND APPLIED
C     MATHEMATICS, INC.  CALRES COMPUTES THE LINE AND ELEMENT POSITION
C     FOR A GIVEN (XLAT, XLON) USING THE BETA SWEEP ANGLE.
C     INPUTS:  IYD.  YEAR  DAY
C              T.  TIME OF LANDMARK SCAN LINE.
C              BETA.  SWEEP ANGLE ASSOCIATED WITH LANDMARK SCAN LINE.
C              MEANOM  MEAN ANAMOLY OF SATELLITE POSITION AT  EPIC
C              XLAT.  THE LATITUDE OF THE EARTH LANDMARK.
C              XLON.  THE LONGITUDE OF THE EARTH LANDMARK.
C     OUTPUTS:  XLIN.  THE COMPUTED LINE POSITION.
C               XELE.  THE COMPUTED ELEMENT POSITION.
      REAL MEANOM
      DOUBLE PRECISION T,XSN,YSN,ZSN,XSNST,YSNST,XST,YST,ANG,TWPI,BETA  0411
      COMMON/NAVCOM/NAVDAY,LINTOT,DEGLIN,IELTOT,DEGELE,SPINRA,IETIMY,IET0412
     1IMH,SEMIMA,OECCEN,ORBINC,PERHEL,ASNODE,NOPCLN,DECLIN,RASCEN,PICLIN0413
     2,PRERAT,PREDIR,PITCH,YAW,ROLL,SKEW                                0414
      COMMON/NAVINI/                                                    0415
     1  EMEGA,AB,ASQ,BSQ,R,RSQ,                                         0416
     2  RDPDG,                                                          0417
     3  NUMSEN,TOTLIN,RADLIN,                                           0418
     4  TOTELE,RADELE,PICELE,                                           0419
     5  CPITCH,CYAW,CROLL,                                              0420
     6  PSKEW,                                                          0421
     7  RFACT,ROASIN,TMPSCL,                                            0422
     8  D11,D12,D13,D21,D22,D23,D31,D32,D33,                            0423
     9  GAMA,GAMDOT,                                                    0424
     A  ROTM11,ROTM13,ROTM21,ROTM23,ROTM31,ROTM33,                      0425
     B  PICTIM,XREF                                                     0426
      COMMON/BETCOM/IAJUST,IBTCON,NEGBET,ISEANG                         0427
      DATA TWPI/6.28318531D0/                                           0428
      TIME=T                                                            0430
      JTIME=ITIME(TIME)                                                 0431
C     GET VECTOR TO THE SUN IN INERTIAL COORDINATES
      CALL SUNVEC(IYD,T,XSN,YSN,ZSN)                                    0432
C     GET VECTOR TO THE SATELLITE IN INERTIAL COORDINATES
      CALL STVEC(TIME,MEANOM,XSAT,YSAT,ZSAT)
C     COMPUTE THE X, Y, Z VECTOR POINTING TO THE LANDMARK IN INERTIAL
C     COORDINATES FROM THE EARTH'S CENTER.
      JTIME=ITIME(TIME)
      YLON=RAERAC(IYD,JTIME,XLON)*RDPDG
      TDIF=TIME-FTIME(JTIME)
      YLON=YLON+TDIF*TWPI/24.0
      YLAT=XLAT*RDPDG
      YLAT=GEOLAT(YLAT,1)
      TANLAT=TAN(YLAT)**2
      RR=SQRT((1.0+TANLAT)/(BSQ+ASQ*TANLAT))*AB
      XE=COS(YLON)*COS(YLAT)*RR
      YE=SIN(YLON)*COS(YLAT)*RR
      ZE=SIN(YLAT)*RR
      SENANG=FLALO(ISEANG)*RDPDG                                        0439
C     FETCH CURRENT ATTITUDE STATE AND SETUP SATELLITE CENTERED
C     COORDINATE SYSTEM WITH Z-AXIS OPPOSITE THE SPIN AXIS.
```

```
      CALL ATPREC(TIME,DEC,RAS)
      DEC=DEC*RDPDG
      RAS=RAS*RDPDG
      SR=SIN(RAS)
      CR=COS(RAS)
      SD=SIN(DEC)
      CD=COS(DEC)
      X1=-SR
      Y1=CR
      Z1=0.0
      X2=-SD*CR
      Y2=-SD*SR
      Z2=CD
      X3=CD*CR
      Y3=CD*SR
      Z3=SD
C     FIND LINE NUMBER
      COSANG=X3*(XE-XSAT)+Y3*(YE-YSAT)+Z3*(ZE-ZSAT)
      COSANG=COSANG/SQRT((XE-XSAT)**2+(YE-YSAT)**2+(ZE-ZSAT)**2)
      THETA=ATAN2(ROTM31,ROTM33)
      PHI=ASIN(COSANG/SQRT(ROTM13**2+ROTM33**2))
      YLIN=PHI-THETA
      XLIN=PICLIN-YLIN/RADLIN
C     FIND ELEMENT NUMBER
C     FIND COORDINATES OF THE VECTOR FROM THE SATELLITE TO THE SUN
C     IN THE SATELLITE COORDINATE SYSTEM
      XSNST=X1*(-XSN-XSAT)+Y1*(-YSN-YSAT)
      YSNST=X2*(-XSN-XSAT)+Y2*(-YSN-YSAT)+Z2*(-ZSN-ZSAT)
C     FIND THE COORDINATES OF THE VECTOR FROM THE SATELLITE TO THE EARTH
C     LANDMARK IN THE SATELLITE COORDINATE SYSTEM.
      XST=X1*(XE-XSAT)+Y1*(YE-YSAT)
      YST=X2*(XE-XSAT)+Y2*(YE-YSAT)+Z2*(ZE-ZSAT)
C     CORRECT FOR ROLL AND SKEW
      CLIN=COS(YLIN)
      SLIN=SIN(YLIN)
      U=ROTM11*CLIN+ROTM13*SLIN
      V=ROTM21*CLIN+ROTM23*SLIN
      ANG=DATAN2(YSNST,XSNST)-DATAN2(YST,XST)-BETA+SENANG+ATAN2(V,U)
C     NORMALIZE THIS ANGLE TO LIE BETWEEN -PI/2.0 AND PI/2.0
      ANG=DMOD(ANG,TWPI)
      IF(ANG.GT.(TWPI/2.0D0)) ANG=ANG-TWPI
      IF(ANG.LT.(-TWPI/2.0D0)) ANG=ANG+TWPI
      XELE=ANG/RADELE
      RETURN
      END
```

SIZE 436 00664

Appendix F

Fortran Listings of

Standard Geometry

Transformation Routines:

EARLOC and IMGLOC

```
      SUBROUTINE EARLOC(ISC,FRTIME,DATE,SLINE,ELEM,ELAT,ELONG)
C
C
C        PURPOSE:
C          EARLOC COMPUTES THE GEODETIC LATITUDE (ELAT) AND LONGITUDE
C          (ELONG) CORRESPONDING TO THE GIVEN LINE (SLINE) AND ELEMENT
C          (ELEM) COORDINATES OF THE VISSR IMAGE FRAME STARTING AT FRTIME.
C          EARLOC PERFORMS THE INVERSE TRANSFORM OF IMGLOC.
C
C
C        PROGRAMMER: LARRY HAMBRICK, NESS/OSI
C
C        ORIGINATION DATE: FEBRUARY 18,1978
C
C
C        ARGUMENTS:
C
C          THE UNITS OF ELAT AND ELONG ARE DECIMAL DEGREES WITH ELAT
C          POSITIVE NORTH AND ELONG POSITIVE EAST.
C
C          THE UNITS OF SLINE AND ELEM ARE SCAN LINES AND IR SAMPLES
C          RESPECTIVELY. THEIR FRACTIONAL PARTS SPECIFY HIGHER
C          RESOLUTIONS.
C
C            NOTE: THE RANGE FOR ELEM IS 0.5 TO 3822.5. THE IR SAMPLE
C                  NUMBER IS OBTAINED BY ROUNDING-OFF ELEM. THE VISIBLE
C                  SAMPLE NUMBER IS OBTAINED BY ROUNDING-OFF ((ELEM-0.5)
C                  *4 + 0.5).
C
C                  THE RANGE FOR SLINE IS 0.5 TO 1821.5. THE IR (SCAN)
C                  LINE NUMBER IS OBTAINED BY ROUNDING-OFF SLINE. THE
C                  VISIBLE LINE NUMBER IS OBTAINED BY ROUNDING-OFF
C                  ((SLINE-0.5)*8 + 0.5).
C
C          ISC DESIGNATES THE SATELLITE: 1 FOR EAST GOES, 2 FOR WEST.
C
C          FRTIME IS THE GMT IN SECONDS OF THE START OF THE IMAGE FRAME.
C
C          DATE SPECIFIES THE YEAR AND JULIAN DAY AS YYDDD.
C
C
C
C
C        REFERENCES:
C
C          THE MATHEMATICAL BASIS FOR THIS ROUTINE IS THE PAPER BY
C          DENNIS PHILLIPS AND C.T.MOTTERSHEAD. THE REPORT BY WESTING-
C          HOUSE (CONTRACT NAS5-23582,DATED JULY,1977) IS MORE DIRECTLY
C          LINKED TO THE FORTRAN CODE AND GIVES DETAILED DEFINITIONS
C          AND ILLUSTRATIONS OF THE ORBIT/ATTITUDE PARAMETERS AVAILABLE
C          IN THE VISSR DOCUMENTATION BLOCK.
C
C          EARLOC IS NOT IN THE MOST EFFICIENT COMPUTATIONAL FORM;
C          EMPASIS HERE IS ON CLARITY OF THE GEOMETRY.
```

```
C
C
      DIMENSION PHI(6),S(3,3)
      COMMON / GRNICH / GRA1,GRA2
      COMMON / TIM / TIME1,DATE1,D
      COMMON / SATATT / SPRA1,SPRA2,SPDC1,SPDC2
      COMMON / SATPOS / CX(11),CY(11),CZ(11)
      COMMON / ALNANG / ZETA,RHO,ETA
      COMMON / SPNRAT / SPPER
C
C           GRA1 GREENICH ANGLE IN DEGREES AT TIME TIME1
C           GRA2 GREENWICH ANGLE IN DEGREES AT TIME TIME1 + D
C           TIME1 GMT IN SECONDS OF EPOCH FOR NAVIGATION PARAMETERS
C           D PERIOD IN SECONDS OVER WHICH TIME IS NORMALIZED
C           DATE1 DATE AS YYDDD,YEAR AND JULIAN DAY,OF EPOCH FOR NAV. PAR
C           SPRA-,SPDC- RIGHT ASCENSION AND DECLINATION OF POSITIVE SPIN
C                       AXIS IN DEGREES AT TIMES: (1)TIME1 AND (2) TIME1
C                                                                 PLUS D
C           CX(I),CY(I),CZ(I),I=1 TO 11  CHEBYCHEV COEFFICIENTS REPRESENT
C                  -ING THE SATELLITE POSITION IN KM BEGINNING AT EPOCH
C                  (TIME1) IN EARTH CENTERED INERTIAL COORDINATES
C                  NORMALIZED WRT TIME OVER D
C           RHO IS ROLL OR ELEMENT BIAS OF THE VISSR IN DEGREES
C           ZETA IS PITCH OR LINE BIAS OF THE VISSR IN DEGREES
C           ETA IS YAW OR SKEW OF THE VISSR IN DEGREES
C           SPPER SPIN PERIOD OF SATELLITE IN SECONDS
C
      DATA A,B,ACQANG,TOTSMP,SCENCA,SCENCS,TOTSCL / 6378.144,6356.759,
     1   9.1875,3822.0,45.0,4096.0,1821.0 /
C
C           A: EQATORIAL RADIUS OF OBLATE ELLIPSOID EARTH  IN KM
C           B: POLAR RADIUS OF OBLATE ELLIPSOID EARTH IN KM
C           ACQANG: DATA ACQUISTION ANGLE OF VISSR IN DEGREES
C           TOTSMP: TOTAL IR SAMPLES ACQUIRED IN SCAN LINE
C           SCENCA: VISSR SCAN ENCODER CHARACTERISTIC ANGLE IN DEGREES
C           SCENCS: NUMBER OF VISSR SCAN ENCODER POSITIONS
C           TOTSCL: TOTAL NUMBER OF SCAN LINES
C
      PI=3.1415926535
C
C     COMPUTE SOME CONSTANT PARAMETERS
C
      E=(A**2 - B**2)/B**2
      BSAS= B**2/A**2
      AMUE= (2.0*ACQANG/TOTSMP)*PI/180.0
      AMUL= (SCENCA/SCENCS)*PI/180.0
      CE= (TOTSMP + 1.0)/2.0
      CL= (TOTSCL + 1.0)/2.0
C
C      COMPUTE VISSR ALIGNMENT ANGLES
C
      PITCH = ZETA*PI/180.0
      YAW= ETA*PI/180.0
```

```
      ROLL = RHO*PI/180.0
C
C          CHECK THAT ELEM AND SLINE ARE IN LIMITS
C
      IF((ELEM.GT.3822.5).OR.(ELEM.LT.0.5)) GO TO 10
      IF((SLINE.GT.1821.5).OR.(SLINE.LT.0.5)) GO TO 10
      GO TO 15
   10 PRINT 30
   30 FORMAT(41H  LINE OR ELEMENT NUMBER IS OUT OF LIMITS )
      ELAT=999.9
      ELONG=999.9
      GO TO 20
   15 CONTINUE
C
C        COMPUTE POLAR COORDINATES OF THE UNIT VIEW VECTOR IN
C          THE VISSR COORDINATE SYSTEM
C
      AZM=AMUE*(ELEM-CE)
      ELV = AMUL*(CL-SLINE)
C
C        TRANSFORM THE UNIT VECTOR TO SATELLITE INERTIAL COORDINATES
C
      VS1=COS(AZM+ROLL)*COS(PITCH+ELV)-SIN(AZM+ROLL)*SIN(YAW)*SIN(PITCH
     1 +ELV)
      VS2=+SIN(AZM+ROLL)*COS(PITCH+ELV)+COS(AZM+ROLL)*SIN(YAW)*SIN(PITCH
     1 +ELV)
      VS3=-COS(YAW)*SIN(PITCH+ELV)
C
C          DETERMINE TIME OF THE GIVEN SAMPLE (APPROX. TO THE SCAN
C          LINE TIME)
C
      T = FRTIME +  AINT(SLINE +0.5)*SPPER
C
C        TRANSFORM THE UNIT VIEW VECTOR TO EARTH CENTERED INERTIAL
C          COORDINATES
C
      CALL SATCRD(T,S)
      VC1= S(1,1)*VS1 + S(2,1)*VS2 + S(3,1)*VS3
      VC2 = S(1,2)*VS1 + S(2,2)*VS2 + S(3,2)*VS3
      VC3 = S(1,3)*VS1 + S(2,3)*VS2 + S(3,3)*VS3
C
C        EXTEND THE UNIT VECTOR TO INTERSECTION WITH THE EARTH
C             ELLIPSOID
C
      CALL SATVEC(T,PCX,PCY,PCZ)
      F=(PCX*VC1+PCY*VC2+PCZ*VC3 + E*PCZ*VC3)/(1.0+E*VC3**2)
      G=(PCX**2+PCY**2+PCZ**2-A**2 + E*PCZ**2)/(1.0 + E*VC3**2)
      Q = -F - SQRT(F**2-G)
C
C        COMPUTE THE VECTOR FROM THE EARTH CENTER TO THE SAMPLE
C          POINT IN EARTH CENTERED INERTIAL COORDINATES
C
      RC1 = PCX + Q*VC1
```

```
      RC2 = PCY + Q*VC2
      RC3 = PCZ + Q*VC3
C
C          COMPUTE THE EARTH GEODETIC LATITUDE AND LONGITUDE OF SAMPLE
C
      ELAT = ATAN((1.Ø/BSAS)*RC3/SQRT(RC1**2+RC2**2)) *18Ø.Ø/PI
C
C            THE LONGITUDE IS COMPUTED UNDER THE CONVENTION -18Ø TO +18Ø
C                                  WITH EAST OF GREENWICH +
C
      ANG = ATAN2(RC2,RC1)
C
C            COMPUTE THE GREENWICH ANGLE
C
      CALL GRANG(T,W)
C
      ELONG = ANG - W
      IF(ELONG.LE.-PI) ELONG=2.Ø*PI+ELONG
      IF(ELONG.GT.PI) ELONG=ELONG-2.Ø*PI
      ELONG = ELONG*18Ø.Ø/PI
   2Ø CONTINUE
      RETURN
      END
```

SIZE 499 ØØ763

```
      SUBROUTINE IMGLOC(ISC,FRTIME,DATE,ELAT,ELONG,SLINE,ELEM)
C
C
C        PURPOSE:
C          IMGLOC COMPUTES THE LINE (SLINE) AND ELEMENT (ELEM) COOR-
C          DINATES,IN THE VISSR IMAGE FRAME STARTING AT TIME (FRTIME),
C          CORRESPONDING TO THE GIVEN EARTH GEODETIC LATITUDE (ELAT) AND
C          LONGITUDE (ELONG).
C          IMGLOC PERFORMS THE INVERSE TRANSFORM OF EARLOC.
C
C
C        PROGRAMMER: LARRY HAMBRICK, NESS/OSI
C
C        ORIGINATION DATE: FEBRUARY 18,1978
C
C
C        ARGUMENTS:
C
C          THE UNITS OF ELAT AND ELONG ARE DECIMAL DEGREES WITH ELAT
C          POSITIVE NORTH AND ELONG POSITIVE EAST.
C
C          THE UNITS OF SLINE AND ELEM ARE SCAN LINES AND IR SAMPLES
C          RESPECTIVELY. THEIR FRACTIONAL PARTS SPECIFY HIGHER
C          RESOLUTIONS.
C
C            NOTE: THE RANGE FOR ELEM IS Ø.5 TO 3822.5. THE IR SAMPLE
C                  NUMBER IS OBTAINED BY ROUNDING-OFF ELEM. THE VISIBLE
C                  SAMPLE NUMBER IS OBTAINED BY ROUNDING-OFF ((ELEM-Ø.5)
C                  *4 + Ø.5).
C
C                  THE RANGE FOR SLINE IS Ø.5 TO 1821.5. THE IR (SCAN)
C                  LINE NUMBER IS OBTAINED BY ROUNDING-OFF SLINE. THE
C                  VISIBLE LINE NUMBER IS OBTAINED BY ROUNDING-OFF
C                  ((SLINE-Ø.5)*8 + Ø.5).
C
C          ISC DESIGNATES THE SATELLITE: 1 FOR EAST GOES, 2 FOR WEST.
C
C          FRTIME IS THE GMT IN SECONDS OF THE START OF THE IMAGE FRAME.
C
C          DATE SPECIFIES THE YEAR AND JULIAN DAY AS YYDDD.
C
C
C        REFERENCES:
C
C          THE MATHEMATICAL BASIS FOR THIS ROUTINE IS THE PAPER BY
C          DENNIS PHILLIPS AND C.T.MOTTERSHEAD. THE REPORT BY WESTING-
C          HOUSE (CONTRACT NAS5-23582,DATED JULY,1977) IS MORE DIRECTLY
C          LINKED TO THE FORTRAN CODE AND GIVES DETAILED DEFINITIONS
C          AND ILLUSTRATIONS OF THE ORBIT/ATTITUDE PARAMETERS AVAILABLE
C          IN THE VISSR DOCUMENTATION BLOCK.
C
C          IMGLOC IS NOT IN THE MOST EFFICIENT COMPUTATIONAL FORM;
C          EMPASIS HERE IS ON CLARITY OF THE GEOMETRY.
```

```
C
C
      DIMENSION PHI(6),T(6),S(3,3)
      COMMON / GRNICH / GRA1,GRA2
      COMMON / TIM / TIME1,DATE1,D
      COMMON / SATATT / SPRA1,SPRA2,SPDC1,SPDC2
      COMMON / SATPOS / CX(11),CY(11),CZ(11)
      COMMON / ALNANG / ZETA,RHO,ETA
      COMMON / SPNRAT / SPPER
C
C         GRA1 GREENICH ANGLE IN DEGREES AT TIME TIME1
C         GRA2 GREENWICH ANGLE IN DEGREES AT TIME TIME1 + D
C         TIME1 GMT IN SECONDS OF EPOCH FOR NAVIGATION PARAMETERS
C         D PERIOD IN SECONDS OVER WHICH TIME IS NORMALIZED
C         DATE1 DATE AS YYDDD,YEAR AND JULIAN DAY,OF EPOCH FOR NAV. PAR
C         SPRA-,SPDC- RIGHT ASCENSION AND DECLINATION OF POSITIVE SPIN
C                     AXIS IN DEGREES AT TIMES: (1)TIME1 AND (2) TIME1
C                                                              PLUS D
C         CX(I),CY(I),CZ(I),I=1 TO 11  CHEBYCHEV COEFFICIENTS REPRESENT
C                 -ING THE SATELLITE POSITION IN KM BEGINNING AT EPOCH
C                 (TIME1) IN EARTH CENTERED INERTIAL COORDINATES
C                 NORMALIZED WRT TIME OVER D
C         RHO IS ROLL OR ELEMENT BIAS OF THE VISSR IN DEGREES
C         ZETA IS PITCH OR LINE BIAS OF THE VISSR IN DEGREES
C         ETA IS YAW OR SKEW OF THE VISSR IN DEGREES
C         SPPER SPIN PERIOD OF SATELLITE IN SECONDS
C
      DATA STDEC,A,B,ACQANG,TOTSMP,SCENCA,SCENCS,TOTSCL /
     1  6.611,6378.144,6356.759,9.1875,3822.0,45.0,4096.0,1821.0 /
C
C         STDEC:RATIO OF NOMINAL SATELLITE RANGE AND EARTH RADIUS
C         A: EQATORIAL RADIUS OF OBLATE ELLIPSOID EARTH  IN KM
C         B: POLAR RADIUS OF OBLATE ELLIPSOID EARTH IN KM
C         ACQANG: DATA ACQUISTION ANGLE OF VISSR IN DEGREES
C         TOTSMP: TOTAL IR SAMPLES ACQUIRED IN SCAN LINE
C         SCENCA: VISSR SCAN ENCODER CHARACTERISTIC ANGLE IN DEGREES
C         SCENCS: NUMBER OF VISSR SCAN ENCODER POSITIONS
C         TOTSCL: TOTAL NUMBER OF SCAN LINES
C
      PI= 3.1415926535
C
C     COMPUTE SOME CONSTANT PARAMETERS
C
      E=(A**2 - B**2)/B**2
      BSAS= B**2/A**2
      AMUE= (2.0*ACQANG/TOTSMP)*PI/180.0
      AMUL= (SCENCA/SCENCS)*PI/180.0
      CE= (TOTSMP + 1.0)/2.0
      CL= (TOTSCL + 1.0)/2.0
C
C      COMPUTE VISSR ALIGNMENT ANGLES
C
      PITCH = ZETA*PI/180.0
```

```
      YAW= ETA*PI/180.0
      ROLL = RHO*PI/180.0
C
C          CONVERT GEODETIC LATITUDE AND LONGITUDE TO RADIANS
C
      PLAMDA= FLAT*PI/180.0
      GLONG= ELONG*PI/180.0
C
C        COMPUTE GFOCENTRIC LATITUDE
C
      GLAMDA=ATAN(BSAS*TAN(PLAMDA))
C
C        ESTIMATE THE TIME AT WHICH THE POINT WAS SCANNED
C
      PHI(1)= ATAN( 1.0*SIN(PLAMDA)/(STDEC-COS(PLAMDA)))-PITCH
      T(1)=FRTIME-(AINT(PHI(1)/AMUL + 0.5)-CL)*SPPER
C
C        THIS LOOP ITERATES TO REFINE THE TIME ESTIMATE
C
      DO 40 I=1,5
      TI=T(I)
C
C        COMPUTE THE GREENWICH ANGLE
C
      CALL GRANG(TI,W)
C
C        DETERMINE THE EARTH POINT VECTOR IN EARTH CENTERED INERTIAL
C                COORDINATES
C
      XL1=COS(GLAMDA)*COS(GLONG+W)
      XL2=COS(GLAMDA)*SIN(GLONG+W)
      XL3=SIN(GLAMDA)
      R=A/SQRT(1.0 + E*SIN(GLAMDA)**2)
      RC1=R*XL1
      RC2=R*XL2
      RC3=R*XL3
C
C        DETERMINE THE SATELLITE VECTOR
C
      CALL SATVEC(TI,PX,PY,PZ)
C
C        CHECK WHETHER EARTH POINT IS IN VIEW
C
      XLDP=XL1*PX + XL2*PY + XL3*PZ
      IF(XLDP.LT.A) GO TO 50
C
C        COMPUTE VIEW VECTOR IN EARTH CENTERED INERTIAL COORDINATES
C
      VC1=RC1-PX
      VC2=RC2-PY
      VC3=RC3-PZ
C
C        TRANSFORM THE VIEW VECTOR TO SATELLITE INERTIAL COORDINATES
```

```
C
      CALL SATCRD(TI,S)
      VS1=S(1,1)*VC1 + S(1,2)*VC2 + S(1,3)*VC3
      VS2=S(2,1)*VC1 + S(2,2)*VC2 + S(2,3)*VC3
      VS3=S(3,1)*VC1 + S(3,2)*VC2 + S(3,3)*VC3
C
C          DETERMINE AZIMUTH AND ELEVATION OF EARTH POINT IN SAT. COORD.
C
      SIGMA=ATAN(VS2/VS1)
      TANYAW=TAN(YAW)
      COSYAW=COS(YAW)
      XI=ATAN(VS3*TANYAW/SQRT(VS1**2+VS2**2-VS3**2*TANYAW**2))
      PHI(I+1)=ATAN(-VS3/(COSYAW*COS(XI)*SQRT(VS1**2+VS2**2))) -PITCH
C
C          CHECK FOR CONVERGENCE OF THE TIME ESTIMATE
C
      DSLINE=(AINT(PHI(I+1)/AMUL + 0.5)-AINT(PHI(I)/AMUL + 0.5))
      K=I+1
      IF(DSLINE.GE.1.0) GO TO 15
      GO TO 25
C
C          CORRECT THE ESTIMATE OF TIME
C
 15   T(I+1)=T(I)-DSLINE*SPPER
  40  CONTINUE
C
C          COMPUTE THE LINE AND ELEMENT
C
  25  THETA=XI+SIGMA-ROLL
      ELEM= CE + THETA/AMUE
      SLINE=CL-(PHI(K)/AMUL)
C
C          ROUGH CHECKS ON REASONABLENESS OF RESULTS
      IF(ELEM.LT.0.5) GO TO 35
      IF(ELEM.LE.3822.5) GO TO 30
  35  PRINT 39
  39  FORMAT(17H  ELEMENT ERROR  )
  30   IF(SLINE.LT.0.5) GO TO 45
      IF(SLINE.LE.1821.5) GO TO 60
  45  PRINT 49
  49  FORMAT(14H   LINE ERROR  )
      GO TO 60
   50 PRINT 55
  55  FORMAT(18H  BEHIND THE EARTH)
      ELEM=0.0
      SLINE=0.0
  60  CONTINUE
      RETURN
      END
```

SIZE 566 01066

```
      SUBROUTINE SATCRD(T,S)
C          COMPUTES THE BASE VECTORS ( THE MATRIX S) OF THE SATELLITE
C            COORDINATE SYSTEM IN TERMS OF THE EARTH CENTERED INERTIAL
C            COORDINATE SYSTEM AT TIME (T) BASED ON THE SATELLITE
C            POSITION AND ATTITUDE
      COMMON / SATPOS / CX(11),CY(11),CZ(11)
      COMMON / SATATT / SPRA1,SPRA2,SPDC1,SPDC2
      COMMON / TIM / TIME1,DATE,D
      DIMENSION S(3,3)
      CALL SPNATT(T,RASC,DECL)
      S(3,1) = COS(DECL)*COS(RASC)
      S(3,2)=COS(DECL)*SIN(RASC)
      S(3,3)=SIN(DECL)
      CALL SATVEC(T,PX,PY,PZ)
      PDOTS=PX*S(3,1)+PY*S(3,2)+PZ*S(3,3)
      PSQ=PX**2+PY**2+PZ**2
      S1DEN=SQRT(PSQ-PDOTS**2)
      S(1,1)=(-PX+PDOTS*S(3,1))/S1DEN
      S(1,2)=(-PY+PDOTS*S(3,2))/S1DEN
      S(1,3)=(-PZ+PDOTS*S(3,3))/S1DEN
      S(2,1)=S(3,2)*S(1,3)-S(3,3)*S(1,2)
      S(2,2)=S(3,3)*S(1,1)-S(3,1)*S(1,3)
      S(2,3)=S(3,1)*S(1,2)-S(3,2)*S(1,1)
      RETURN
      END
```

SIZE 145 00221

```
      SUBROUTINE SATVEC(T,PCX,PCY,PCZ)
C     COMPUTES THE COMPONENTS (PCX,PCY,AND PCZ) OF THE VECTOR TO THE
C       SATELLITE IN EARTH CENTERED INERTIAL COORDINATES AT
C         TIME (T) BASED ON THE CHEBYCHEV COEFFICIENTS CX(I),CY(I),CZ(I)
      DIMENSION BX(13),BY(13),BZ(13)
      COMMON / SATPOS / CX(11),CY(11),CZ(11)
      COMMON / TIM / TIME1,DATE,D
      CALL NTIM(T,U)
      DO 5 I=12,13
      BX(I)=Ø.Ø
      BY(I)=Ø.Ø
      BZ(I)=Ø.Ø
    5 CONTINUE
      DO 15 J=1,11
      BX(12-J)=CX(12-J)+2.Ø*U*BX(12-J+1)-BX(12-J+2)
      BY(12-J)=CY(12-J)+2.Ø*U*BY(12-J+1)-BY(12-J+2)
      BZ(12-J)=CZ(12-J)+2.Ø*U*BZ(12-J+1)-BZ(12-J+2)
   15 CONTINUE
      PCX=(BX(1)-BX(3))/2.Ø
      PCY=(BY(1)-BY(3))/2.Ø
      PCZ=(BZ(1)-BZ(3))/2.Ø
      RETURN
      END
```

SIZE 156 ØØ234

```
      SUBROUTINF SPNATT(T,SPRASC,SPDECL)
C          COMPUTFS THE RIGHT ASCFNSION(SPRASC) AND DECLINATION(SPDFCL)
C           OF THE SPIN AXIS AT TIME (T) WITH A LINEAR
C           INTERPOLATION BASED ON VALUFS OF SPRA1,2 AND SPDC1,2
C
C          THE INTERPOLATION TECHNIQUE USED HERE IS A SIMPLE APPROXIM-
C          ATION BUT IS SUFFICIENTLY PRECISE FOR THF SMALL PRECESSION
C          RATFS ENCOUNTERED.
C
C            SPRA1,2 HAS RANGE -18Ø TO +18Ø DEGREES
C            SPDC1,2 HAS RANGE -9Ø TO +9Ø DEGREES
C            SPRASC HAS RANGE -PI TO +PI RADIANS
C            SPDFCL HAS RANGE -PI/2 TO +PI/2 RADIANS
C          NOTE  IF THE ABSOLUTE VALUE OF THE DIFFERENCE BETWEEN
C                SPRA1 AND SPRA2 IS GREATER THAN 9Ø DEGREES, THF
C                INTERPOLATION OF DECLINATION WILL BF INCORRECT
      COMMON/ SATATT / SPRA1,SPRA2,SPDC1,SPDC2
      COMMON / TIM / TIME1,DATE,D
      PI=3.1415926535
      CALL NTIM(T,U)
      RASC1=SPRA1*PI/18Ø.Ø
      RASC2=SPRA2*PI/18Ø.Ø
      DECL1=SPDC1*PI/18Ø.Ø
      DECL2=SPDC2*PI/18Ø.Ø
      IF(RASC1.LT.Ø.Ø) RASC1=2.Ø*PI+RASC1
      IF(RASC2.LT.Ø.Ø) RASC2=2.Ø*PI+RASC2
      SPRASC=Ø.5*(RASC2+RASC1+(RASC2-RASC1)*U)
      SPRASC=AMOD(SPRASC,2.Ø*PI)
      IF(SPRASC.GT.PI) SPRASC=SPRASC-2.Ø*PI
      SPDECL=Ø.5*(DECL2+DECL1+(DFCL2-DECL1)*U)
      RETURN
      END
```

SIZE 1Ø1 ØØ145

```
      SUBROUTINE NTIM(T,U)
C           COMPUTES NORMALIZED TIME (U) FROM ACTUAL TIME (T) BASED
C             ON TIME1 AND D.
C           TIME1 IS IN SECONDS
C             D IS 13 HOURS IN SECONDS
      COMMON / TIM / TIME1,DATE,D
      IF (TIME1.GT.T) GO TO 5
      U=2.0*(T-TIME1)/D-1.0
      GO TO 10
 5    U=2.0*(T+24.0*60.0*60.0-TIME1)/D-1.0
 10   RETURN
      END
```

SIZE 33 00041

```
      SUBROUTINE GRANG(T,W)
C
C           COMPUTE GREENWICH ANGLE(W) IN RADIANS FROM
C             TIME(T) BASED ON VALUES OF GRA1 AND GRA2 ( IN DEGREES)
C                                         RANGE IS -180 TO +180 DEGREES
C             W HAS RANGE -PI TO +PI IN RADIANS
C
      COMMON / GRNICH / GRA1,GRA2
      COMMON / TIM / TIME1,DATE,D
      PI=3.1415926535
      CALL NTIM(T,U)
      W1=GRA1*PI/180.0
      W2=GRA2*PI/180.0
      IF(W1.LT.0.0) W1=2.0*PI + W1
      IF(W2.LT.0.0) W2=2.0*PI + W2
      IF(W2.LT.W1) W2=W2+2.0*PI
      W=0.5*(W2+W1+(W2-W1)*U)
      W=AMOD(W,2.0*PI)
      IF(W.GT.PI) W=W-2.0*PI
      RETURN
      END
```

SIZE 84 00124

*U.S. GOVERNMENT PRINTING OFFICE : 1980 0-311-046/223

(Continued from inside front cover)

NESS 87 Atlantic Tropical and Subtropical Cyclone Classifications for 1976. D. C. Gaby, J. B. Lushine, B. M. Mayfield, S. C. Pearce, K.O. Poteat, and F. E. Torres, April 1977, 13 pp. (PB-269-674/AS)
NESS 88 National Environmental Satellite Service Catalog of Products. Dennis C. Dismachek (Editor), June 1977, 102 pp. (PB-271-315/AS)
NESS 89 A Laser Method of Observing Surface Pressure and Pressure-Altitude and Temperature Profiles of the Troposphere From Satellites. William L. Smith and C. M. R. Platt, July 1977, 38 pp. (PB-272-660/AS)
NESS 90 Lake Erie Ice: Winter 1975-76. Jenifer H. Wartha, August 1977, 68 pp. (PB-276-386/AS)
NESS 91 In-Orbit Storage of NOAA-NESS Standby Satellites. Bruce Sharts and Chris Dunker, September 1977, 3 pp. (PB-283-078/AS)
NESS 92 Publications and Final Reports on Contracts and Grants, 1976. Catherine M. Frain (Compiler), August 1977, 11 pp. (PB-273-169/AS)
NESS 93 Computations of Solar Insolation at Boulder, Colorado. Joseph H. Pope, September 1977, 13 pp. (PB-273-679/AS)
NESS 94 A Report on the Chesapeake Bay Region Nowcasting Experiment. Roderick A. Scofield and Carl E. Weiss, December 1977, 52 pp. (PB-277-102/AS)
NESS 95 The TIROS-N/NOAA A-G Satellite Series. Arthur Schwalb, March 1978, 75 pp. (PB-283-859/AS)
NESS 96 Satellite Data Set for Solar Incoming Radiation Studies. J. Dan Tarpley, Stanley R. Schneider, J. Emmett Bragg, and Marshall P. Waters, III, May 1978, 36 pp. (PB-284-740/AS)
NESS 97 Publications and Final Reports on Contracts and Grants, 1977. Catherine M. Frain (Compiler), August 1978, 13 pp. (PB-287-855/AS)
NESS 98 Quantitative Measurements of Sea Surface Temperature at Several Locations Using the NOAA-3 Very High Resolution Radiometer. Laurence Breaker, Jack Klein, and Michael Pitts, September 1978, 28 pp. (PB-288-488/AS)
NESS 99 An Empirical Model for Atmospheric Transmittance Functions and Its Application to the NIMBUS-6 HIRS Experiment. P.G. Abel and W.L. Smith, NESS, and A. Arking, NASA, September 1978, 29 pp. (PB-288-487/AS)
NESS 100 Characteristics and Environmental Properties of Satellite-Observed Cloud Rows. Samuel K. Beckman (in consultation).
NESS 101 A Comparison of Satellite Observed Middle Cloud Motion With GATE Rawinsonde Data. Leroy D. Herman, January 1979, 13 pp. (PB-292-341/AS)
NESS 102 Computer Tracking of Temperature-Selected Cloud Patterns. Lester F. Hubert, January 1979, 15 pp. (PB-292-159/AS)
NESS 103 Objective Use of Satellite Data To Forecast Changes in Intensity of Tropical Disturbances. Carl O. Erickson, April 1979, 44 pp. (PB-298-915)
NESS 104 Publications and Final Reports on Contracts and Grants. Catherine M. Frain, (Compiler), September 1979. (PB80 122385)
NESS 105 Optical Measurements of Crude Oil Samples Under Simulated Conditions. Warren A. Hovis and John S. Knoll, October 1979, 20 pp. (PB80 120603)
NESS 106 An Improved Model for the Calculation of Longwave Flux at 11 μm. P. G. Abel and A. Gruber, October 1979, 24 pp. (PB80 119431)
NESS 107 Data Extraction and Calibration of TIROS-N/NOAA Radiometers. Levin Lauritson, Gary J. Nelson, and Frank W. Porto, November 1979. (PB80 150824)
NESS 108 Publications and Final Reports on Contracts and Grants. Catherine M. Frain, (Compiler), August 1980.
NESS 109 Catalog of Products, Third Edition. Dennis C. Dismachek, Arthur L. Booth, and John A. Leese, April 1980, 130 pp.
NESS 110 GOES Data Collection Program. Merle Nelson, August 1980.

NOAA SCIENTIFIC AND TECHNICAL PUBLICATIO

The National Oceanic and Atmospheric Administration was established as part of th nt of Commerce on October 3, 1970. The mission responsibilities of NOAA are to assess the socio pact of natural and technological changes in the environment and to monitor and predict the state o arth, the oceans and their living resources, the atmosphere, and the space environment of the Eart

The major components of NOAA regularly produce various types of scientific and te rma- tion in the following kinds of publications:

PROFESSIONAL PAPERS — Important definitive research results, major techniques, and special investigations.

CONTRACT AND GRANT REPORTS — Reports prepared by contractors or grantees under NOAA sponsorship.

ATLAS — Presentation of analyzed data generally in the form of maps showing distribution of rainfall, chemical and physical conditions of oceans and atmosphere, distribution of fishes and marine mammals, ionospheric conditions, etc.

TECHNICAL SERVICE PUBLICATIONS — Reports containing data, observations, instructions, etc. A partial listing includes data serials; prediction and outlook periodicals; technical manuals, training papers, planning reports, and information serials; and miscellaneous technical publications.

TECHNICAL REPORTS — Journal quality with extensive details, mathematical developments, or data listings.

TECHNICAL MEMORANDUMS — Reports of preliminary, partial, or negative research or technology results, interim instructions, and the like.

Information on availability of NOAA publications can be obtained from:

ENVIRONMENTAL SCIENCE INFORMATION CENTER (D822)
ENVIRONMENTAL DATA AND INFORMATION SERVICE
NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
U.S. DEPARTMENT OF COMMERCE

6009 Executive Boulevard
Rockville, MD 20852

NOAA--S/T 80-183