PROGRAM MANUAL FOR HILTOP

A Heliocentric Interplanetary Low Thrust Trajectory Optimization Program

Part II - Subroutine Descriptions

(NASA-CR-143895) PROGRAM MANUAL FOR HILTOP, A HELIOCENTRIC INTERPLANETARY LOW THRUST TRAJECTORY OPTIMIZATION PROGRAM. PART 2: SUBROUTINE DESCRIPTIONS (Analytical Mechanics Associates, Inc.) 860 p HC $19.25 G3/13

N75-28085

Unclas
29199

F. I. Mann
J. L. Horsewood

Report No. 74-34
Contract NAS8-29945
December 1974

ANALYTICAL MECHANICS ASSOCIATES, INC.

SUMMARY

This report describes a Phase A performance-analysis computer program, HILTOP, that has been developed explicitly to generate optimum electric propulsion trajectory data for missions of interest in the exploration of the solar system. HILTOP is a double-precision, FORTRAN IV, IBM 360 production program which is primarily designed to evaluate the performance capabilities of electric propulsion systems and which may, in the hands of a skilled analyst, perform efficiently in the simulation of a wide variety of interplanetary missions. HILTOP uses numerical integration of the two-body, three-dimensional equations of motion and the Euler-Lagrange equations. It contains transversality conditions which permit the rapid generation of converged maximum-payload trajectory data, and allows the optimization of numerous other performance indices for which no transversality conditions are included. In addition to optimizing the thrust direction and on-off switch times, other significant performance parameters that can be optimized are jet exhaust speed, power level, hyperbolic excess speeds, launch asymptote geocentric declination, flight time and launch date. The ability to simulate constrained optimum solutions, including trajectories having specified propulsion time and constant thrust cone angle, are also optionally available. The program is designed to handle multiple-target missions with various types of encounters, such as rendezvous, stopover, orbital capture, and flyby. Performance requirements for a variety of launch vehicles may be determined. The documentation includes problem formulation, program usage specifications, sample problems, and detailed subroutine descriptions.

Preceding Page Blank

TABLE OF CONTENTS

VI. Subroutine Descriptions (cont)

VI. Subroutine Descriptions (cont)

NOMENCLATURE

Generally, upper-case symbols denote vectors and lower-case symbols denote scalars. Lower-case symbols with bars denote unit vectors. The abbreviations EPS for electric propulsion system and BVP for boundary value problem are used.

a	EPS instantaneous thrust acceleration; semi-major axis
a_c	Semi-major axis of primary-target capture orbit
a_i	Solar power law coefficients
$\bar{a}_1$, $\bar{a}_2$	Arbitrary unit vectors used in (132) and (139)
b	A coefficient in the efficiency law
b_1, b_2, b_3	Launch vehicle coefficients
C	Vector constant of optimal rocket problem, expression (63)
C^o	Radians-to-degrees conversion factor
c	EPS jet exhaust speed (constant); abbreviation for cosine function
c_r	Retro stage jet exhaust speed
c_1	Auxiliary quantity given by expression (74)
c_1, c_2, c_3	Coefficients in quadratic expression for Δv_i, expression (78).
d	A coefficient in the efficiency law; an auziliary quantity in the coast-phase solution; solar flux density
E	Eccentric anomaly (a scalar)

e — A coefficient in the efficiency law; the base of the natural logarithms; eccentricity; subscript denoting Earth

$\bar{e}_h$ — Spacecraft unit angular momentum vector

$\bar{e}_r$ — Spacecraft unit radius vector

$\bar{e}_t$ — EPS unit thrust vector

$\bar{e}_v$ — Spacecraft unit velocity vector

e_x — Retro stage characteristic speed exponential factor given by expression (76)

$\bar{e}_\lambda$ — Unit primer vector

F — Auxiliary scalar function defined by (215)

f — EPS instantaneous thrust magnitude; f-function of the f and g series; subscript denoting a desired value; true anomaly; auxiliary variable defined by equation (147)

f_r — Retro stage thrust magnitude

f_x — Auxiliary quantity given by expression (77)

G_i — Auxiliary scalar functions in the coast-phase solution, equation (45)

g — EPS reference thrust acceleration; g-function of the f and g series; BVP point-constraint geometric mean of the weighting factors

g_x — Auxiliary quantity given by expression (97)

H — Spacecraft angular momentum vector

h — Magnitude of spacecraft angular momentum vector

$\bar{h}$ — Spacecraft unit angular momentum vector

h_v — Variational Hamiltonian

$\left.\begin{matrix} h_x \\ h_y \\ h_z \end{matrix}\right\}$ — Cartesian components of spacecraft angular momentum vector

h_σ	Thrust-switching step-function
i	Subscript pertaining to an intermediate target; inclination to ecliptic; general subscript or running index; inclination of parking orbit about Earth
$\bar{i}$	Unit vector along x-axis
i_{max}	Parking orbit inclination associated with range safety limit
J	Index-set of the BVP dependent variables
$\bar{j}$	Unit vector along y-axis
j_p	Unspecified-reference-power indicator
j_{ps}	EPS propulsion system jettison indicator (retro maneuver)
j_r	Retro stage existence indicator
j_t	EPS tankage jettison indicator (retro maneuver)
k	Arbitrary positive constant associated with performance index; temporary variable ultimately equated to inverse of the characteristic degradation time
$\bar{k}$	Unit vector along z-axis
k_c	Auxiliary quantity given by expression (75)
k_{drop}	Intermediate-target drop-mass factor defined by expression (6)
k_{rt}	Retro stage tankage mass factor defined by expression (11)
k_s	EPS structure mass factor defined by expression (8)
k_{samp}	Intermediate-target sample-mass factor defined by expression (6)
k_t	EPS tankage mass factor defined by expression (7)
L	Launch site latitude (scalar)
M	Mean anomaly (scalar)

M_0, M_1, M_2, M_3, M_4, M_5	Coefficients used in computing nuclear and total magnitudes of a celestial body (scalars)
M_N	Nuclear magnitude (scalar)
M_T	Total magnitude (scalar)
m	Spacecraft total mass variable
$\bar{m}$	Auxiliary unit vector given by expression (32)
m_{drop}	Intermediate-target drop-mass given by expression (6)
m_{net}	Net spacecraft mass
m_o	Initial spacecraft mass (payload of launch vehicle) given by expression (2)
m_p	EPS propellant mass
m_{ps}	Electric propulsion system mass given by expression (4)
m_r	Retro stage mass
m_{rp}	Retro stage propellant mass given by expression (9)
m_{rs}	Retro stage structure mass defined by expression (11)
m_{rst}	Retro stage structure and tankage mass given by expression (11)
m_s	EPS structure mass
m_{samp}	Intermediate-target sample-mass given by expression (6)
m_t	EPS tankage mass
Δm_p	Propellant mass increment due to primary-target spiral maneuver
n	Exponent in step-size law, expression (39); subscript denoting time at the primary target; number of BVP dependent variables

$\bar{n}$	Unit vector normal to the solar arrays
$\bar{n}_p$	Unit vector directed along a planet's north pole
o	Subscript denoting launch time; subscript denoting the beginning of a computation step
P	A celestial body's position vector; BVP partial derivative matrix
p	EPS instantaneous power; subscript denoting a perturbed, or neighboring, parameter; auxiliary variable in equations (79)
Δp	Ratio of housekeeping to reference power, p_h/p_{ref}
p_a	Total instantaneous power developed by arrays
p_h	Housekeeping power
p_{ref}	EPS reference power
p_1, p_2	Auxiliary quantities in coast-phase solution, expressions (54) and (55)
q	Auxiliary variable in equations (79); solar array radiation damage factor
R	Spacecraft position vector
r	Magnitude of R
r_a	Primary-target capture-orbit apocenter distance
r_c	Earth-to-spacecraft communication distance
$\bar{r}_n$	Unit vector along line of ascending node
r_p	Primary-target capture-orbit pericenter distance; primary-target swingby passage-distance
$\bar{r}_p$	Swingby passage-distance unit vector
r_{peak}	Value of r for which γ-curve is at a maximum
s	Abbreviation for sine function; auxiliary variable used in equations (79); degradation time

$\bar{s}$	Unit vector directed toward Canopus
t	Time
t_b	Retro maneuver burn time given by expression (12)
Δt	Time-increment due to primary-target spiral maneuver
u	Generalized universal anomaly during thrust phases
Δu	Generalized universal anomaly increment, equivalent to the computation step-size during numerical integration
v	Magnitude of spacecraft velocity
v_c	Characteristic speed of a rocket maneuver
v_e	Escape speed from launch parking orbit
v_g	Minimum velocity impulse required for non-coplanar injection from a circular orbit to a given excess velocity
v_o	Speed of a spacecraft in a circular orbit
v_p	Planetocentric speed at primary-target swingby closest-approach point; auxiliary speed given by equation (72)
V_∞	Hyperbolic excess velocity (or encounter velocity)
$V_{\infty A}$	Swingby planet arrival hyperbolic excess velocity
$V_{\infty D}$	Swingby planet departure hyperbolic excess velocity
v_∞	Hyperbolic excess speed (or encounter speed)
Δv	Retro stage impulsive velocity increment magnitude; characteristic velocity associated with primary-target spiral maneuver; incremental speed required at powered swingby
$\Delta v'$	Retro stage total velocity increment magnitude
Δv_o	Minimum incremental velocity (magnitude) for coplanar boost out of circular orbit
Δv_g	Velocity penalty due to noncoplanar boost out of circular orbit

Δv_ℓ	Velocity penalty due to launch azimuth
w	Auxiliary variable in equations (79)
x	First Cartesian component of position; a general variable; a general state variable; auxiliary variable in equations (79)
y	Second Cartesian component of position; auxiliary variable in equations (79)
z	Third Cartesian component of position
α	EPS specific mass; geocentric right ascension of launch excess velocity
α_A, α_D	Auxiliary parameters defined by equations (211) and (212)
α_a	Specific mass of the solar arrays
α_c	Communication angle (Sun-Earth-spacecraft)
α_t	Specific mass of the power conditioning and thruster subsystem
α_1, α_2	Arbitrary, independent angles defining orientation of excess velocity in (132) and (139)
β	Independent variable of coast-phase solution, also generalized to be the independent variable on the entire trajectory
β_o	Value of β at the beginning of a computation step
$\Delta\beta$	Computation step size (increment of trajectory independent variable)
γ	Normalized power function
γ'	$\partial\gamma/\partial r$
γ^*	$\partial\gamma/\partial d$, where d is the solar flux density
δ	Launch hyperbolic-excess-velocity asymptote declination; BVP dependent-variable tolerance
δ_A, δ_D	Bend angles of hyperbolic arrival and departure trajectories, expression (213)

δ_T Total bend angle given by expression (214)

δ_{ij} Kronecker delta function

ϵ Auxiliary quantity in the coast-phase solution; obliquity of the Earth's equator to the ecliptic

η EPS efficiency

η' $d\eta/dc$

θ In-plane thrust angle

θ_i Travel angle increment

θ_t Travel angle

Λ Primer vector (adjoint to spacecraft velocity)

λ Magnitude of Λ; a general adjoint variable; the iterator inhibitor

λ_c Adjoint variable associated with jet exhaust speed

λ_g Adjoint variable associated with reference thrust acceleration

λ_s Adjoint variable associated with degradation time

λ_x Thrust cone angle Lagrange multiplier

λ_ν Adjoint variable associated with mass ratio

λ_τ Adjoint variable associated with propulsion time

λ_ϕ Adjoint variable associated with thrust cone angle

μ Gravitational constant of the sun; a general gravitational constant

μ_t Gravitational constant of the primary target

ν Mass ratio

$\Delta\nu$ Mass ratio increment at an intermediate target

π Performance index; ratio of circle circumference to diameter

π_x Partial derivative of π with respect to arbitrary variable x.

ρ Auxiliary variable used in equations (79)

σ Thrust switch function

σ^* Special form of thrust switch function, given by equation (186)

σ_r Portion of total thrust switch function, given by (193)

$\Delta\sigma$ Propulsion-corner-proximity tolerance-interval

τ EPS propulsion time

τ_d Characteristic degradation time

Φ Transformation matrix for rotating from ecliptic to equatorial coordinate system

ϕ Thrust cone angle (between thrust and radius)

χ Angle between normal to solar arrays and the spacecraft-sunline

ψ Out-of-plane thrust angle

Ω Longitude of ascending node of an orbit

ω Angular position from the ascending node of an orbit to the spacecraft; argument of perifocus of an orbit

VI. SUBROUTINE DESCRIPTIONS

In this section are presented detailed descriptions of every subroutine of the HILTOP program. The descriptions are given in alphabetic order of the subroutine name. Entry points are not described separately, but are included in the description of the primary subroutine. Each description is comprised of the following sections:

1) Name of subroutine.
2) List of calling arguments.
3) List of sub-programs referenced by the subroutine being described.
4) List of commons referenced by the subroutine.
5) Entry points in the subroutine.
6) List of sub-programs referencing the subroutine being described.
7) Detailed discussion of pertinent equations and logic.
8) Description of printout and messages generated by the subroutine.
9) List of any documents referenced in the Discussion.
10) Table of external variables used by the subroutine.
11) Detailed flow chart and list of non-executable statements.

Items 8 and 9 are included only if applicable.

The table of external variables include for each variable listed the Fortran name, the use of the variable, the name of the common array, if any, in which the variable appears, and a definition of the variable. The variables included in this table include only those referenced within the subroutine which are available for use in other routines; e.g., common variables or variables contained in argument lists. Temporary variables which are evaluated in, and not transmitted out of, the subroutine are not listed. For any array included in the table, the dimension of the array is enclosed in parentheses beside the Fortran name. The use of the variable is indicated by one or more of five alphabetic codes, which are defined as follows:

A - the variable appears in the argument list of a sub-program called by the subroutine.

E - the variable is equivalenced to a common variable.

S - the value of the variable is changed and stored within the subroutine.

U - the value of the variable is used within the subroutine; i.e., the variable name appears on the right hand side of an equation, in an IF statement, or in the list of a WRITE statement.

X - the variable name appears in the argument list of the subroutine or one of its entry points.

If the variable represents a variable used in the Discussion section of the subroutine description, the mathematical symbol of the variable is included in the definition of the variable.

Name: AEINWT

Calling Argument: MPLAN, TDATEX, SAI, ECI, CNI, OMI, SOI, TPI, EMUODD, RADODD

Referenced Sub-programs: DATE1

Referenced Commons: None

Entry Points: None

Referencing Sub-programs: QSTART, SWING

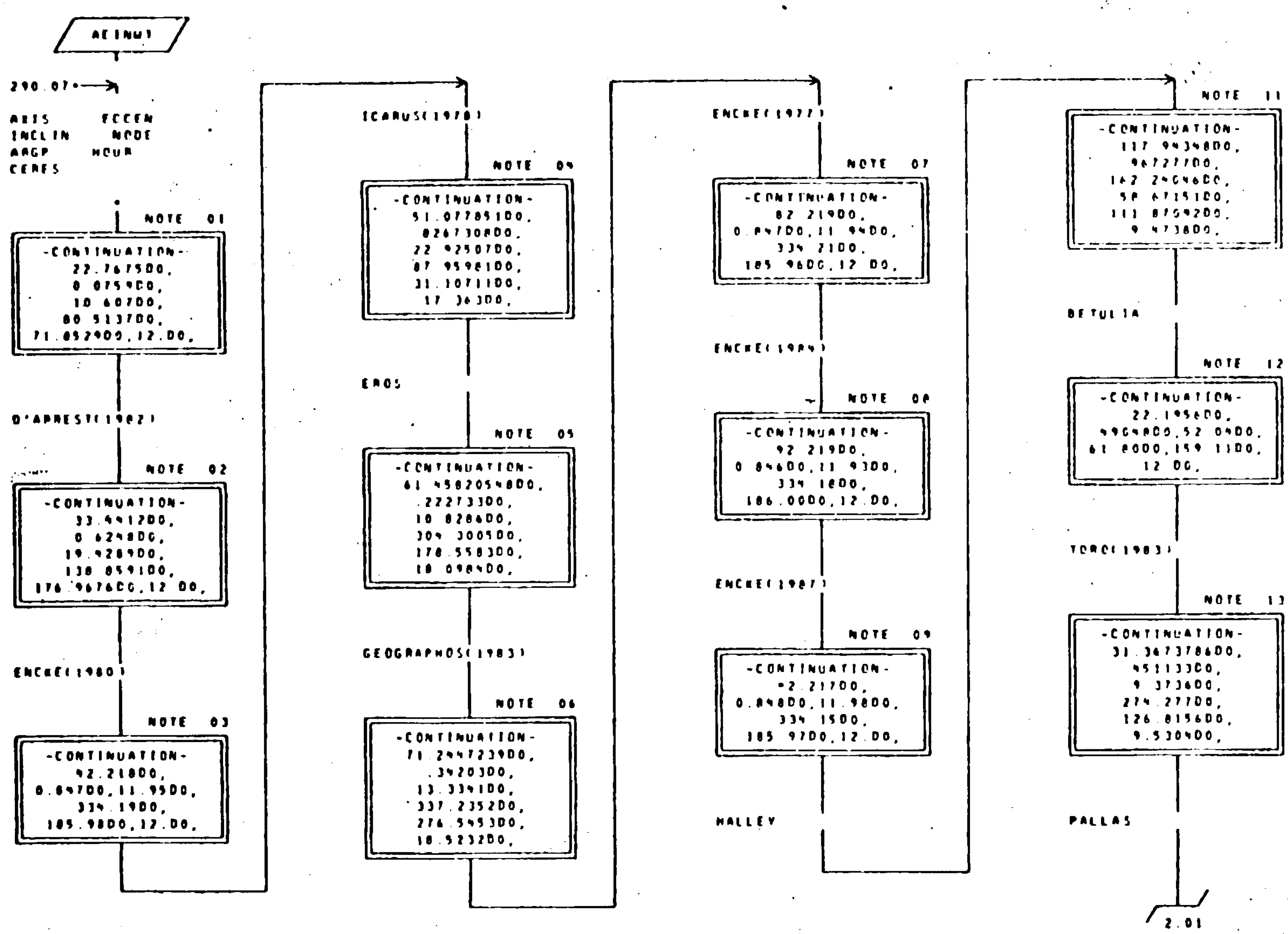
Discussion: The name AEINWT is pronounced "a, e, i, node, omega, t" and stands for the six standard elements which define the motion of a body in its orbit: a, e, i, Ω, ω, t_p, which are semi-major axis, eccentricity, inclination, node angle, argument of perifocus, and time of perifocus passage, where perifocus is perihelion in this program. These elements are defined with respect to the standard ecliptic coordinate system referenced at various times in the twentieth century, mostly in the second half of the century, depending on the celestial object under consideration. The reference dates are given by the MDY array ("Month, Day, Year"), which happen to be the perihelion passage times t_p. The elements and reference dates are considered to be sufficiently accurate for the Phase A type of analyses for which this program is intended. The analyst should improve the a, e, i, Ω, ω, t_p values for his particular celestial target, or add new target element values, as he requires. The element values are fed into subroutine EFM.

The purpose of AEINWT is to procure the desired orbital elements from the DATA arrays and load them in locations used by the program. The COMMENTS interspersed within the DATA arrays in the source listing are sufficient to define the contents of the arrays, i.e., to define the various celestial targets available together with their six standard orbital elements. This routine contains the orbital elements of the essentially massless (except for Ceres) celestial objects such as asteroids and comets, the elements of the planets being found in subroutine EFM.

AEINWT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
CNI	SX		Inclination of orbit to ecliptic, i, in degrees.
ECI	SX		Eccentricity of orbit, e.
OMI	SX		Ascending node angle of orbit, Ω, in degrees.
SAI	SX		Semi-major axis of orbit, a, in AU.
SOI	SX		Argument of perihelion of orbit, ω, in degrees.
TPI	SX		Time of perihelion passage of object in orbit, t_p, in days elapsed since the reference date TDATEX.
MPLAN	UX		Integer variable which selects the celestial object desired.
EMUODD	SX		Gravitational constant μ of the celestial object, in m^3/sec^2.
RADODD	SX		Radius of the celestial object, in meters.
TDATEX	UX		Reference date, in Julian days less 2400000, defined by the program inputs MYEAR, MONTH, MDAY, and HOUR.

CHART TITLE - SUBROUTINE AEINWT(MPLAN,IDATEX,SAI,ECI,CNI,OMI,SOI,TPI,EMUODD,RAD



AEINWT-3

CHART TITLE - SUBROUTINE AEINHTIMPLAN,TDATEX,SAI,ECI,CNI,DMI,SOI,TPI,EMUODD,RAD

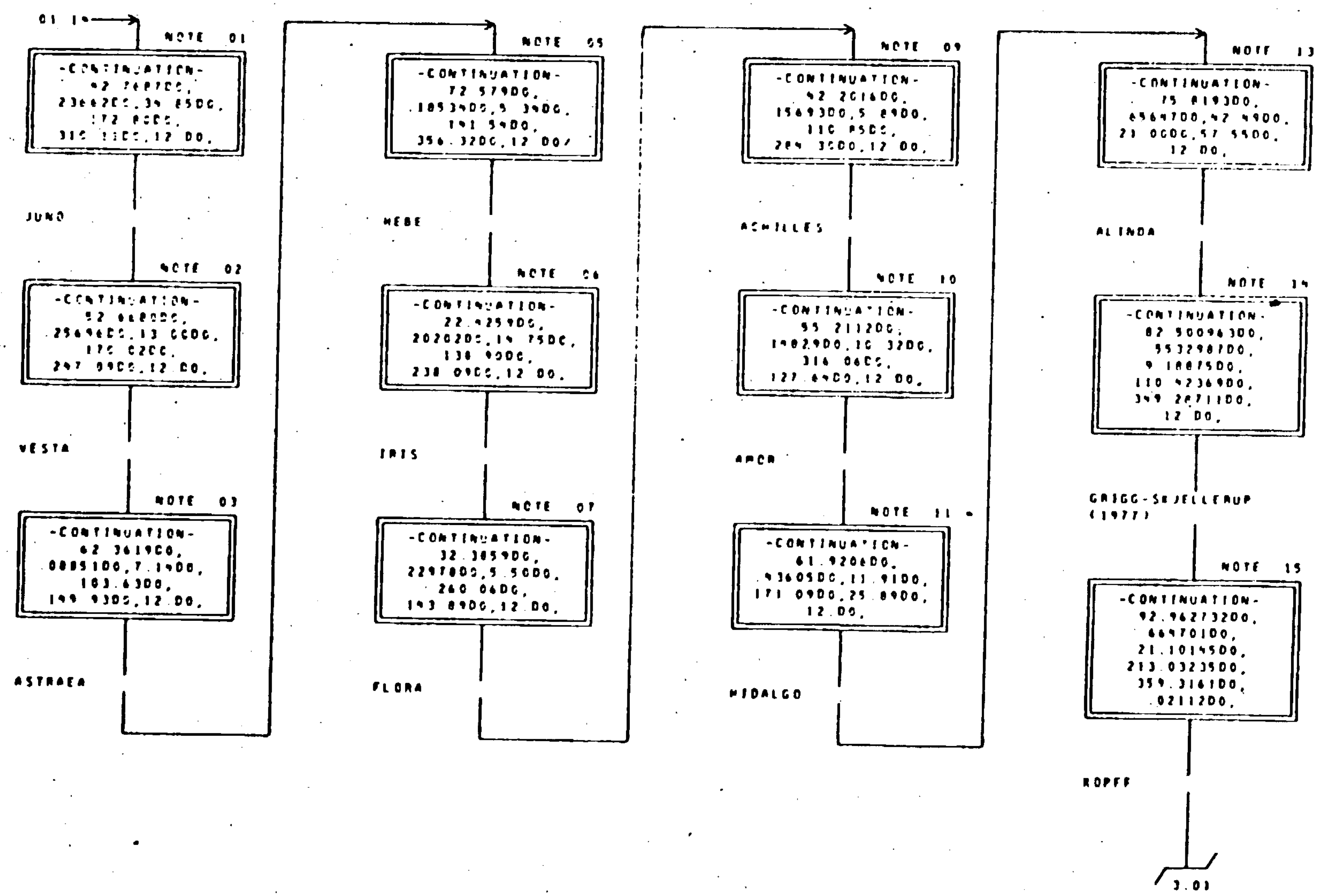
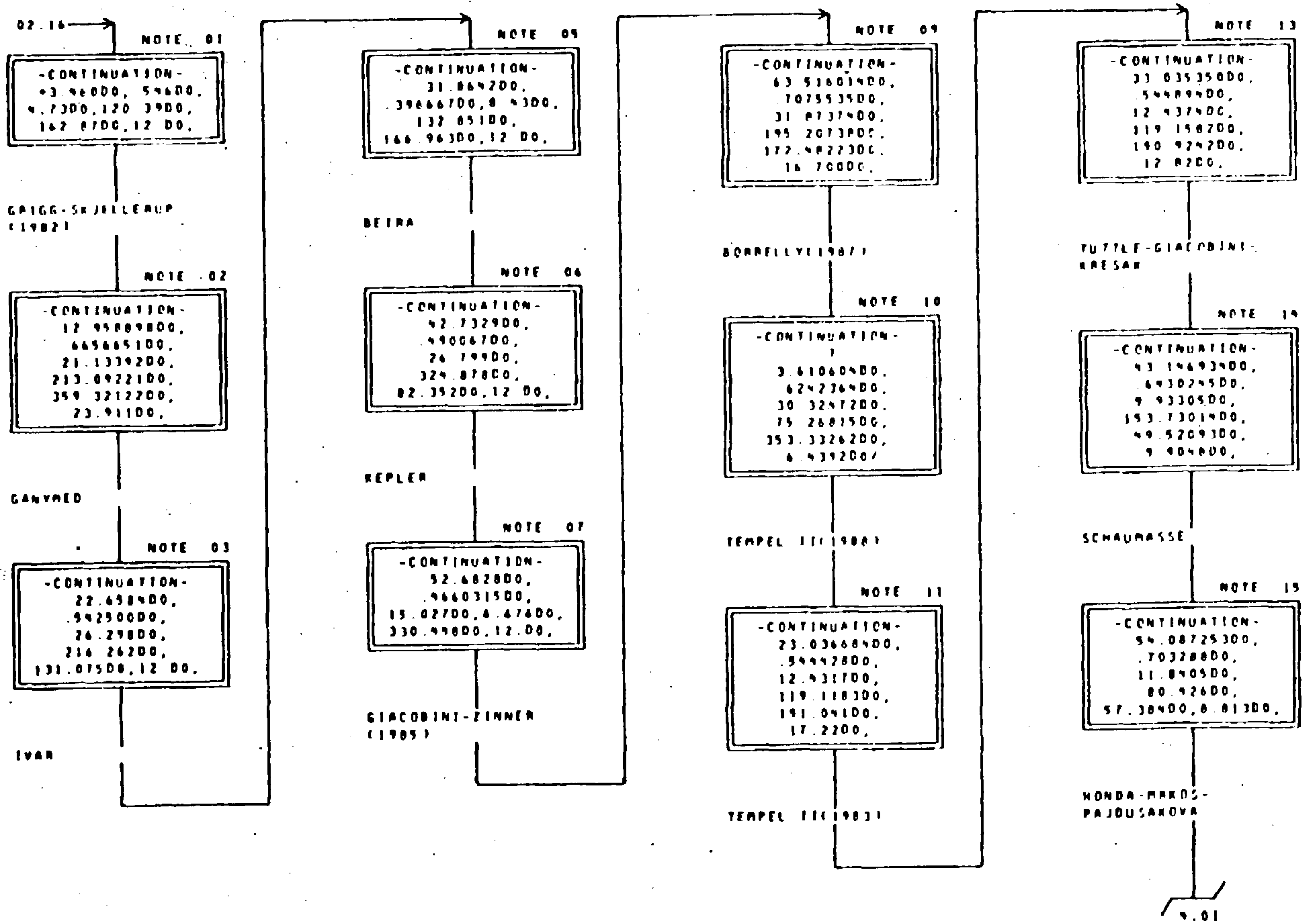


CHART TITLE - SUBROUTINE AEINWT(MPLAN,TDATEX,SAI,ECI,CNI,OMI,SOI,TPI,EMUODO,RAD



AEINWT-5

CHART TITLE - SUBROUTINE AEINWT(MPLAN,TDATE,SAI,ECI,CNI,OMI,SOI,TPI,EMUODD,RAD

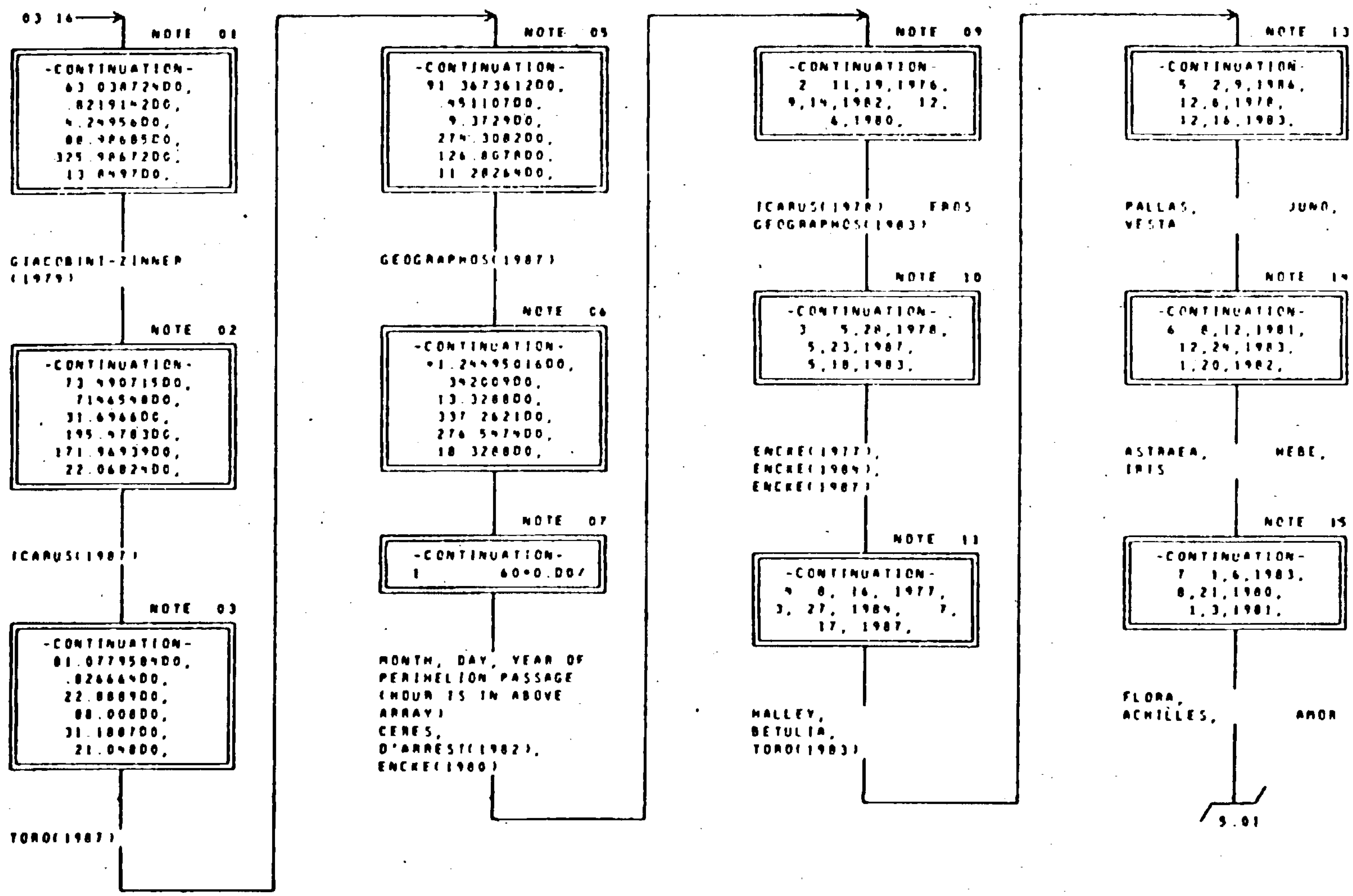
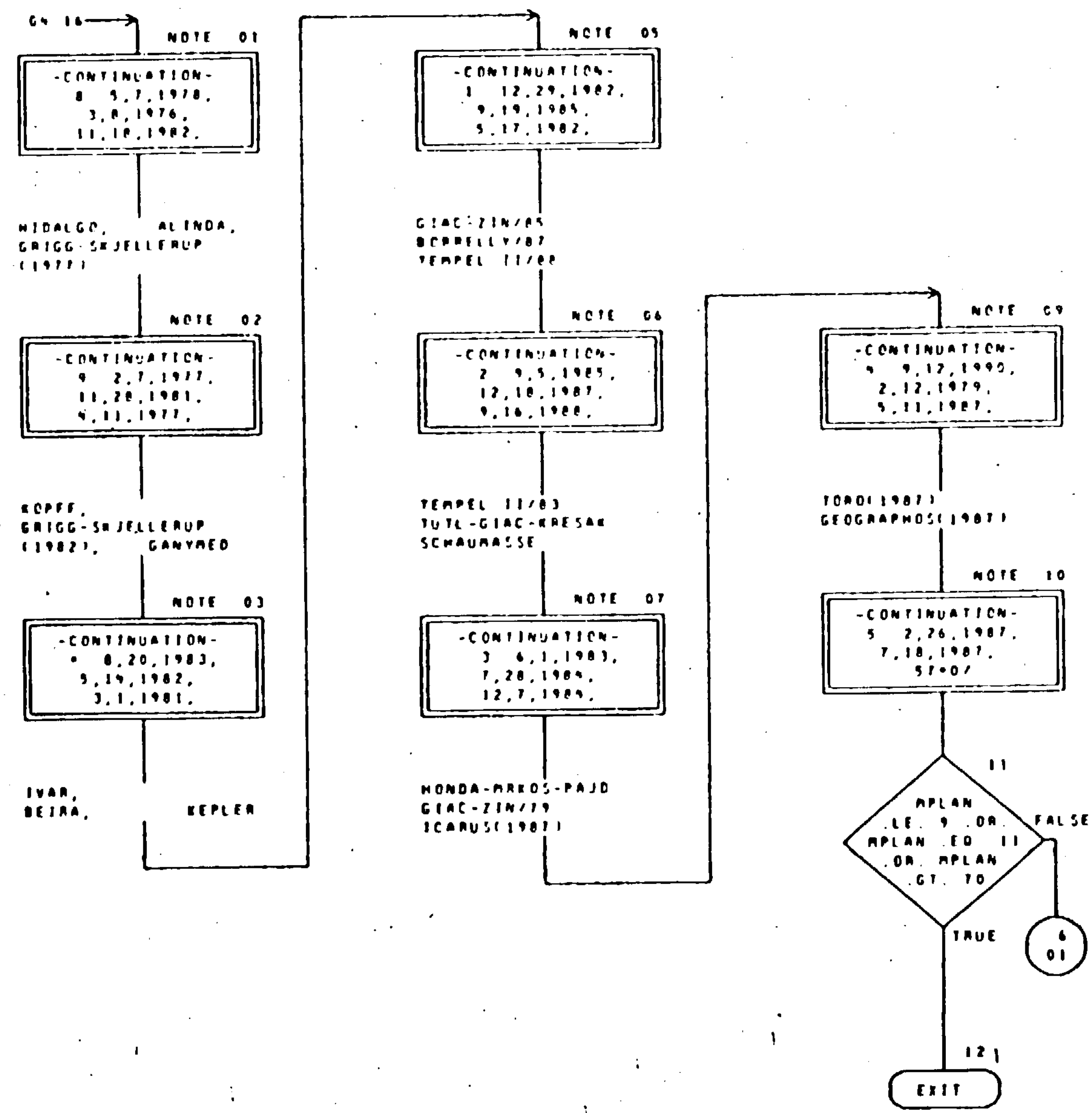


CHART TITLE - SUBROUTINE AEINWT(MPLAN,TDATEX,SAI,ECI,CNI,OMI,SOI,TPI,EMUODD,RAD)



AEINWT-7

CHART TITLE - SUBROUTINE AEINWT(MPLAN,TDATER,SAI,ECI,CNI,OMI,SOI,TPI,EMUODD,RAD

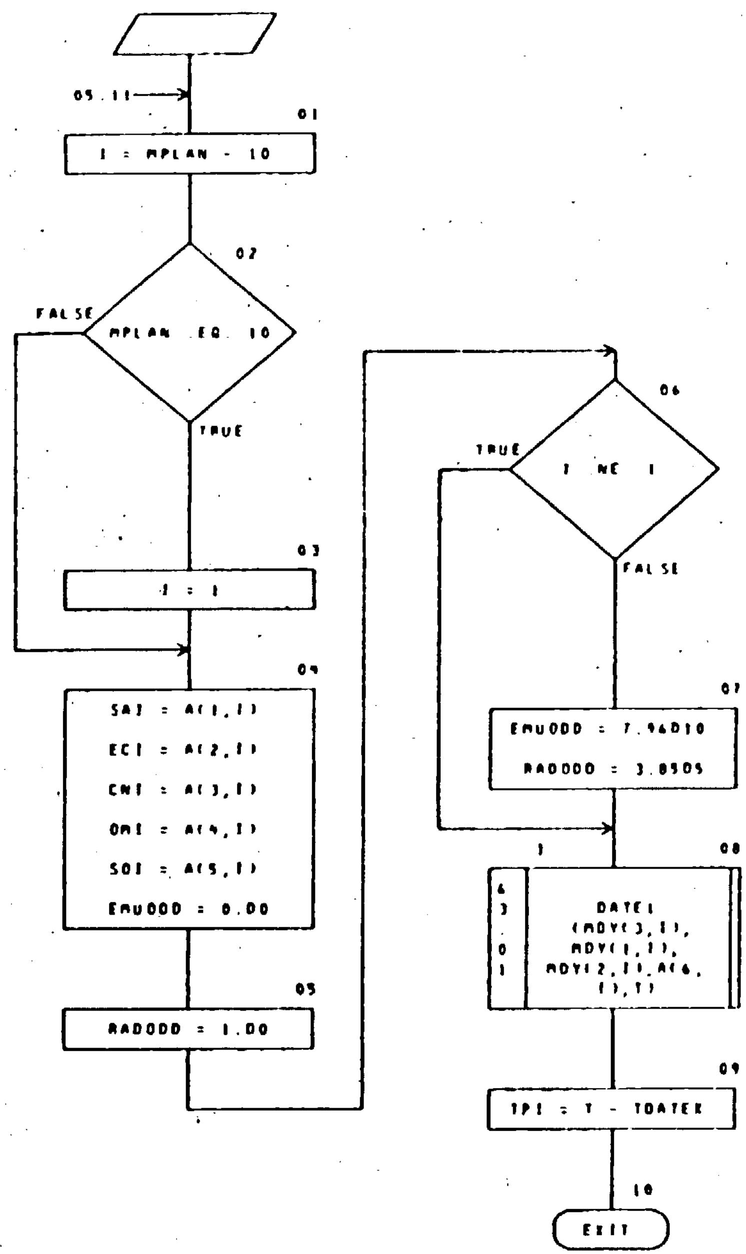


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION A(6,60),MDY(3,60),B(6,16),C(6,16),D(6,19),E(6,9)
DATA B /
DATA C /
DATA D /
DATA E /        54*0 DC/
DATA MDY /
EQUIVALENCE (A(1,1),B(1,1)),(A(1,17),C(1,1)),(A(1,33),D(1,1))
,(A(1,52),E(1,1))
```

Name: ALBEDO

Calling Argument: MTARG, TIME, TARGET, XREL, SMAGN, SMAGT, EMAGN, EMAGT

Referenced Sub-programs: EFM, VDOT, VMAG, VSUB

Referenced Commons: REAL8

Entry Points: None

Referencing Sub-programs: SPRINT

Discussion: This routine computes the photometric magnitude of a given astronomical body as seen by both the spacecraft and an observer on Earth; for comets, both the nuclear and total magnitudes are calculated. These magnitudes may be computed at each computation step, as the spacecraft moves along the trajectory.

The nuclear magnitude (of comet) of the next astronomical body to be encountered along the trajectory, as seen by the spacecraft, is computed as

$$M_N = M_0 + M_1 \log_{10} |R - R_{targ}| + M_2 \log_{10} |R_{targ}|$$

$$+ .03 \cos^{-1}\left[\frac{R_{targ} \cdot (R_{targ} - R)}{|R_{targ}|\ |R_{targ} - R|} \right] C^o$$

where M_0, M_1, and M_2 are magnitude constants associated with the target, and C^o is the radians-to-degrees conversion factor. The arc-cosine term is the phase angle. The total magnitude of the next astronomical body to be encountered along the trajectory, as seen by the spacecraft, is computed as

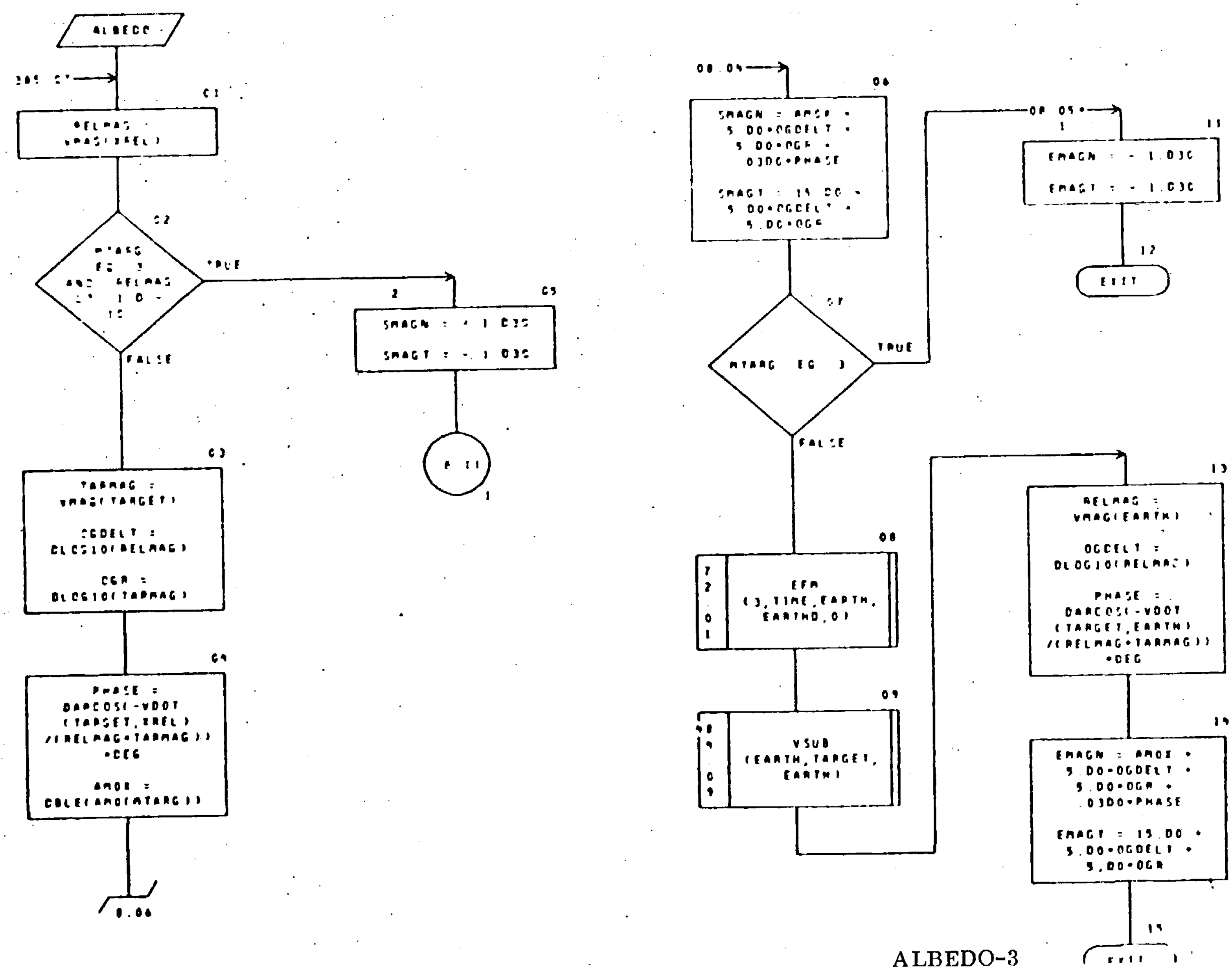
$$M_T = M_3 + M_4 \log_{10} |R - R_{targ}| + M_5 \log_{10} |R_{targ}|$$

where M_3, M_4, and M_5 are magnitude constants associated with the target. R is the spacecraft's instantaneous position vector and R_{targ} is the target's instantaneous position vector, in AU, in ecliptic coordinates. The nuclear and total magnitudes of the target, as seen by Earth, are obtained by substituting R_{earth} for R in these two formulae. Currently, there is a very limited set of values for the magnitude constants coded into the program.

ALBEDO EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
DEG	U	REAL8	Radians to degrees conversion factor.
TIME	AX		Current time, in Julian date less 2400000.
XREL(3)	UX		Target-relative position of spacecraft, $R - R_{targ}$, in AU (ecliptic reference).
EMAGN	SX		Nuclear magnitude of target as seen by Earth.
EMAGT	SX		Total magnitude of target as seen by Earth.
MTARG	UX		Target selector.
SMAGN	SX		Nuclear magnitude of target as seen by the spacecraft.
SMAGT	SX		Total magnitude of target as seen by the spacecraft.
TARGET(3)	UX		Target position, R_{targ}, in AU (ecliptic reference).

CHART TITLE - SUBROUTINE ALBEDO(MTARG,TIME,TARGET,XREL,SMAGN,SMAGT,EMAGN,EMAGT)



```
IMPLICIT REAL*8 (A-H,O-Z)
REAL*4 AMO
DIMENSION AMO(50),TARGET(3),RREL(3),EARTH(6),EARTHO(6)
COMMON /REAL8/ RO1(312),DEG,RO2(1687)
DATA AMO /19*15.,9.6,13*15.,15.5,15.,15.5,4*15.,14.5,
14.5,2*15.,18.,5*15./
```

ORIGINAL PAGE IS
OF POOR QUALITY

Name: ANSTEP

Calling Argument: DBETA, GO

Referenced Sub-programs: PRIOR, SCOMP

Referenced Commons: REAL8

Entry Points: None

Referencing Sub-programs: IMPULS, STEP

Discussion: ANSTEP is a contraction of "analytic step". During any coast phase, the two-body equations of motion and the associated adjoint equations are known to possess analytic solutions obtainable in closed form with respect to the universal trajectory independent-variable β defined below. The particular form of the solution used in the program is derived in Reference 1 and is simply repeated here. This solution employs a universal variable, β, defined implicitly through the differential equation

$$\dot{\beta} = \frac{\sqrt{\mu}}{r} ,$$

where μ is the gravitational constant of the attracting body.

For elliptic, two-body trajectories, this equation has the solution

$$\beta = \sqrt{a} \ (E - E_o) ,$$

where E is the eccentric anomaly and a is the semi-major axis.

The problem to be solved is that of evaluating the state and adjoint variables at $\beta = \beta_o + \Delta\beta$ given the values of these variables at $\beta = \beta_o$. This is accomplished as follows. Given R_o and $\dot{R}_o$ (the state at $\beta = \beta_o$) compute

$$r_o = (R_o \cdot R_o)^{\frac{1}{2}},$$

$$v_o^2 = \dot{R}_o \cdot \dot{R}_o,$$

$$d_o = R_o \cdot \dot{R}_o ,$$

$$\frac{1}{a} = \frac{2}{r_o} - \frac{v_o^2}{\mu} ,$$

$$\epsilon = (\Delta\beta)^2/a = (E - E_o)^2 .$$

Then, using the truncated infinite series expression

$$G_i = (\Delta\beta)^i \sum_{k=0}^{16} \frac{(-\epsilon)^k}{(2k + i)!} ,$$

compute the functions G_i (in subroutine SCOMP) for $i = 5$ and 4. Thereafter, the functions G_i, $i = 0, 1, 2, 3$ are computed from the recursion formula,

$$G_i = \frac{(\Delta\beta)^i}{i!} - \frac{1}{a} G_{i+2} ,$$

and are employed to evaluate the familiar f and g functions, i.e.,

$$f = 1 - \frac{G_2}{r_o} ,$$

$$g = \frac{1}{\sqrt{\mu}} \left(r_o G_1 + \frac{d_o}{\sqrt{\mu}} G_2\right),$$

$$r = r_o G_o + \frac{d_o}{\sqrt{\mu}} G_1 + G_2 ,$$

$$t - t_o = g + \frac{G_3}{\sqrt{\mu}} ,$$

$$\dot{f} = - \frac{\sqrt{\mu}\, G_1}{r\, r_o} ,$$

$$\dot{g} = 1 - \frac{G_2}{r} ,$$

$$R = f R_o + g \dot{R}_o ,$$

$$\dot{R} = \dot{f} R_o + \dot{g} \dot{R}_o .$$

which provide the state and time at the given value of $\beta = \beta_o + \Delta\beta$. g in this subroutine description is not to be confused with the reference thrust acceleration used widely throughout this document. The corresponding equations for the adjoint variables are:

$$\lambda_i(t) = \frac{\partial x_i(t)}{\partial x_j(t_o)} \lambda_j(t_o) + \frac{\partial x_i(t)}{\partial \dot{x}_j(t_o)} \dot{\lambda}_j(t_o),$$

$$\dot{\lambda}_i(t) = \frac{\partial \dot{x}_i(t)}{\partial x_j(t_o)} \lambda_j(t_o) + \frac{\partial \dot{x}_i(t)}{\partial \dot{x}_j(t_o)} \dot{\lambda}_j(t_o),$$

where $i, j = 1, 2, 3$, and repeated subscripts in the same term imply summation over the range of the subscripts. The variables $x_i(t)$ represent the three Cartesian components of the spacecraft position $R(t)$ while the $\lambda_i(t)$ represent the components of the primer vector $\Lambda(t)$. The partial derivatives indicated are given as follows, with δ_{ij} denoting the Kronecker delta function:

$$\frac{\partial x_i}{\partial x_{o_j}} = (\dot{x}_i - \dot{x}_{o_i})\left[\frac{p_1 x_{o_j}}{r_o^3} + \frac{r}{\mu}(\dot{x}_j - \dot{x}_{o_j})\right] + f\delta_{ij} + \frac{x_{o_j}}{r_o^3}\left[\left(G_2 + \frac{2G_4 - \Delta\beta G_3}{r_o}\right)x_{o_i} + (3G_5 - \Delta\beta G_4)\frac{\dot{x}_{o_i}}{\sqrt{\mu}}\right],$$

$$\frac{\partial x_i}{\partial \dot{x}_{o_j}} = \frac{\dot{x}_i - \dot{x}_{o_i}}{\mu}\left[p_1 \dot{x}_{o_j} - G_2 x_{o_j}\right] + \frac{\dot{x}_{o_j}}{\mu}\left[(2G_4 - \Delta\beta G_3)\frac{x_{o_i}}{r_o} + (3G_5 - \Delta\beta G_4)\frac{\dot{x}_{o_i}}{\sqrt{\mu}}\right] + g\delta_{ij},$$

$$\frac{\partial \dot{x}_i}{\partial x_{o_j}} = -\frac{\mu x_i}{r^3}\left[\frac{p_1 x_{o_j}}{r_o^3} + \frac{r}{\mu}(\dot{x}_j - \dot{x}_{o_j})\right] + \frac{\dot{x}_i - \dot{x}_{o_i}}{r}\left[\frac{p_2 x_{o_j}}{r_o^3} - \left(\frac{x_{o_j}}{r_o}G_o + \frac{\dot{x}_{o_j}}{\sqrt{\mu}}G_1\right)\right] + \frac{x_{o_j}}{r r_o^3}\left[\left(G_1 + \frac{G_3 - \Delta\beta G_2}{r_o}\right)\sqrt{\mu}\, x_{o_i} + \left(2G_4 - \Delta\beta G_3\right)\dot{x}_{o_i}\right] + \dot{f}\delta_{ij},$$

$$\frac{\partial \dot{x}_i}{\partial \dot{x}_{o_j}} = -\frac{x_i}{r^3}\left[p_1 \dot{x}_{o_j} - G_2 x_{o_j}\right] + \frac{\dot{x}_i - \dot{x}_{o_i}}{r\sqrt{\mu}}\left[\frac{p_2 \dot{x}_{o_j}}{\sqrt{\mu}} - G_1 x_{o_j}\right]$$

$$+ \frac{\dot{x}_{o_j}}{\mu r}\left[(G_3 - \Delta\beta G_2)\frac{\sqrt{\mu}}{r_o} x_{o_i} + (2G_4 - \Delta\beta G_3)\dot{x}_{o_i}\right] + \dot{g}\,\delta_{ij} ,$$

where

$$p_1 = \frac{1}{\sqrt{\mu}}\left[3G_5 - \Delta\beta G_4 + \frac{d_o}{\sqrt{\mu}}(2G_4 - \Delta\beta G_3) + r_o(G_3 - \Delta\beta G_2)\right] ,$$

$$p_2 = 2G_4 - \Delta\beta G_3 + \frac{d_o}{\sqrt{\mu}}(G_3 - \Delta\beta G_2) - r_o \Delta\beta G_1 .$$

The values of the derivatives of all other state and adjoint variables vanish during coast phases; therefore, their solutions are constants equal to the values they possessed at the start of the coast phases.

Reference:

1. Pines, S. and Fang, T.C., "A Uniform Closed Solution of the Variational Equations for Optimal Trajectories During Coast," presented at the Colloquium on Advanced Problems and Methods for Space Flight Optimization, Liege, Belgium, June 1967.

ANSTEP EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
F	SU	REAL8	The f function.
G	SU	REAL8	The g function.
X(50)	SU	REAL8	Array of trajectory dependent-variables (see RKSTEP for a description of contents).
FX(6)	SU	REAL8	$\partial x_i(t)/\partial x_j(t_o)$ and $\partial x_i(t)/\partial \dot{x}_j(t_o)$.

ANSTEP EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
GO	AX		Indicator for stepping forward or not; in the latter case, the program is iterating to find the root of some function, using β_o as the anchor-point.
GX(6)	SU	REAL8	$\partial \dot{x}_i(t)/\partial x_j(t_o)$ and $\partial \dot{x}_i(t)/\partial \dot{x}_j(t_o)$.
RC	SU	REAL8	Cube of spacecraft solar distance, r^3, in AU^3.
RT	SU	REAL8	Spacecraft solar distance, r, in AU.
SX(50)	U	REAL8	Array of trajectory dependent-variables x_{saved} corresponding to the beginning of the current computation step. Allocated the same as X(50).
XD(50)	S	REAL8	Array of trajectory dependent-variable derivatives; time-derivatives in this routine (during coast phases).
BETA	S	REAL8	Trajectory independent-variable, β, in radians $\cdot$ $AU^{1/2}$.
DBETA	UAX		Computation step-size, i.e., increment of trajectory independent-variable, $\Delta\beta$, in radians $\cdot$ $AU^{1/2}$.
SBETA	U	REAL8	Trajectory independent-variable at beginning of current-computation step, β_o, in radians $\cdot$ $AU^{1/2}$.

CHART TITLE - SUBROUTINE ANSTEP(DBETA,GO)

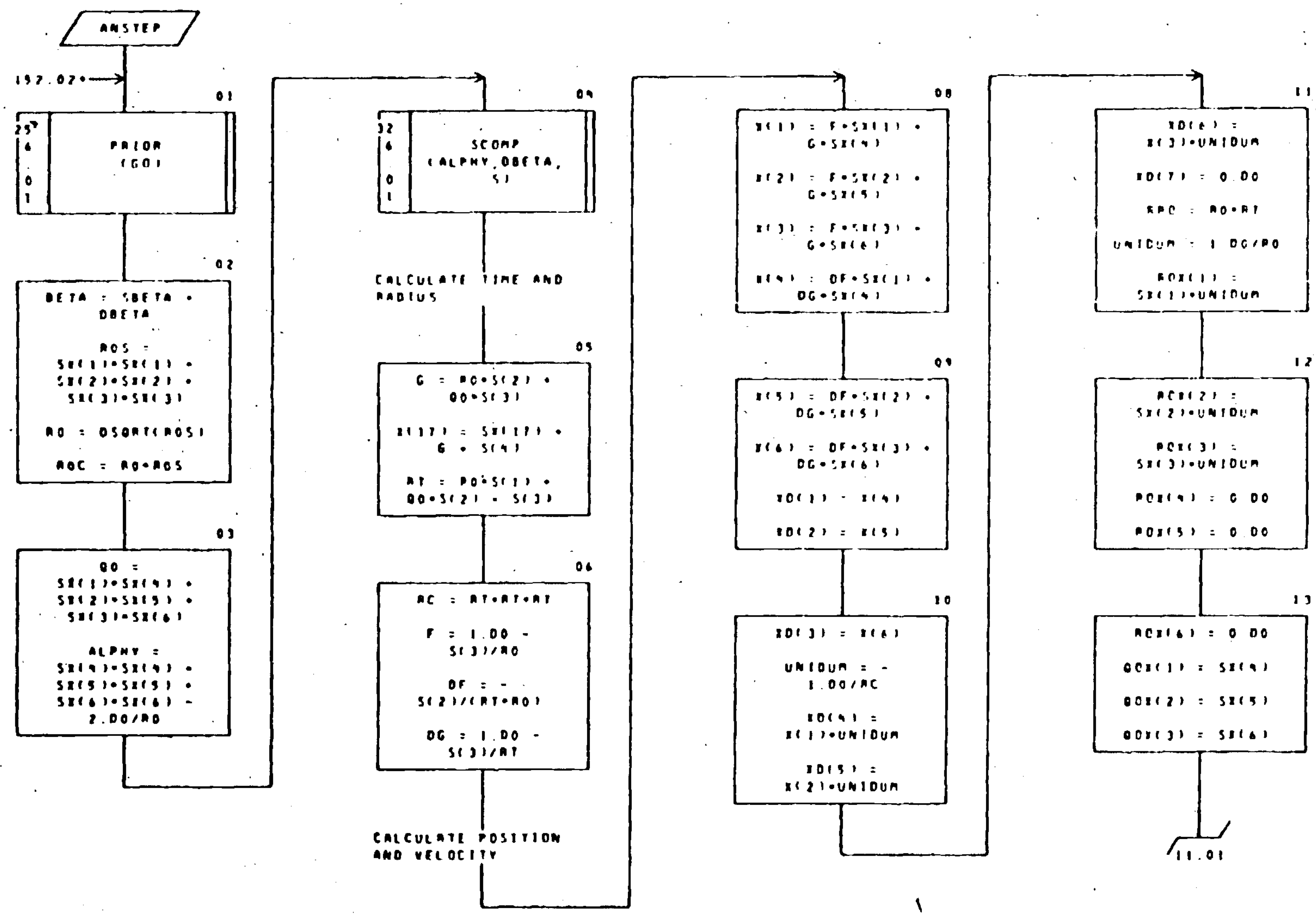
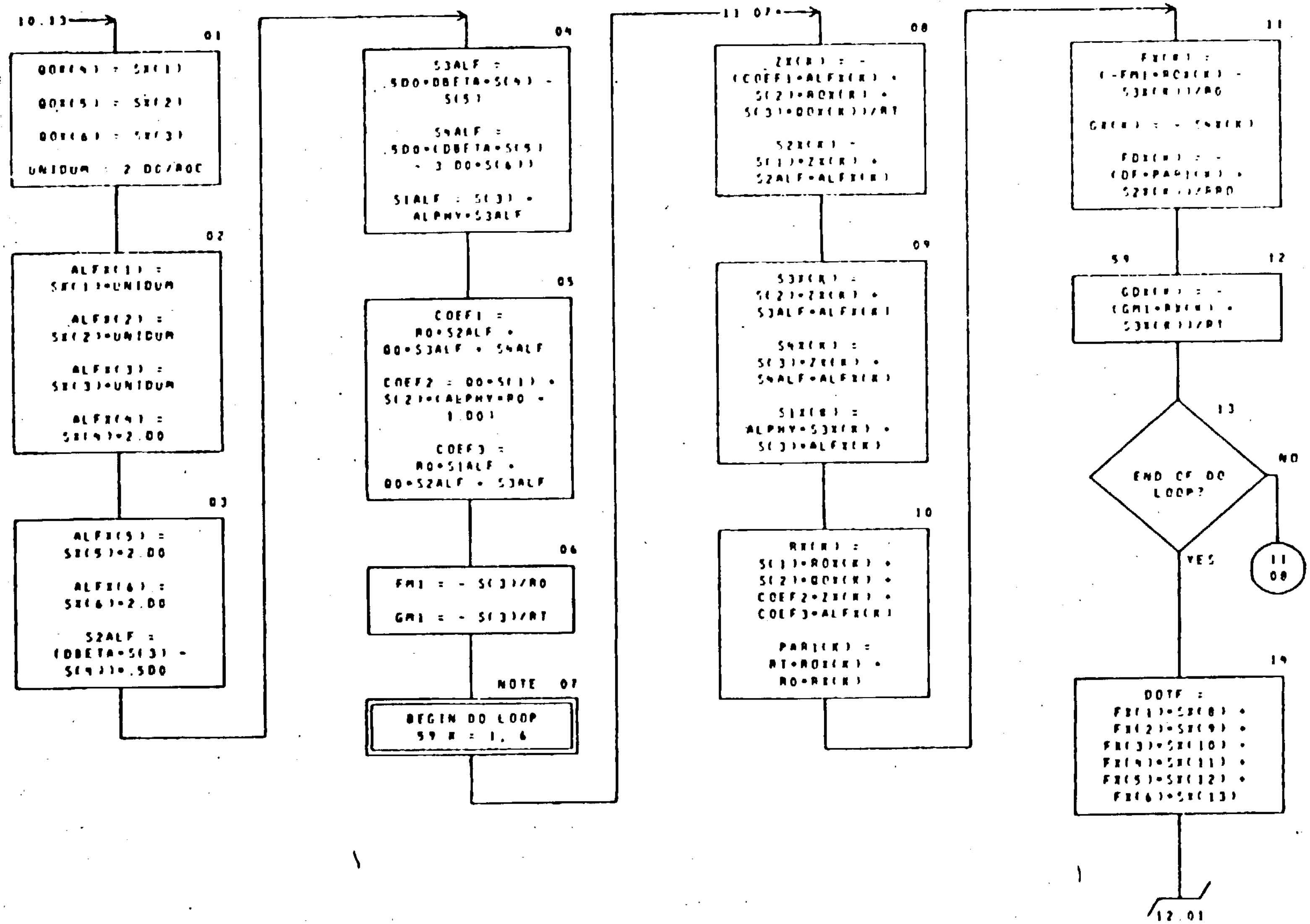


CHART TITLE - SUBROUTINE ANSTEP(DBETA,GO)



ANSTEP-7

CHART TITLE - SUBROUTINE ANSTEP(DBETA,GO)

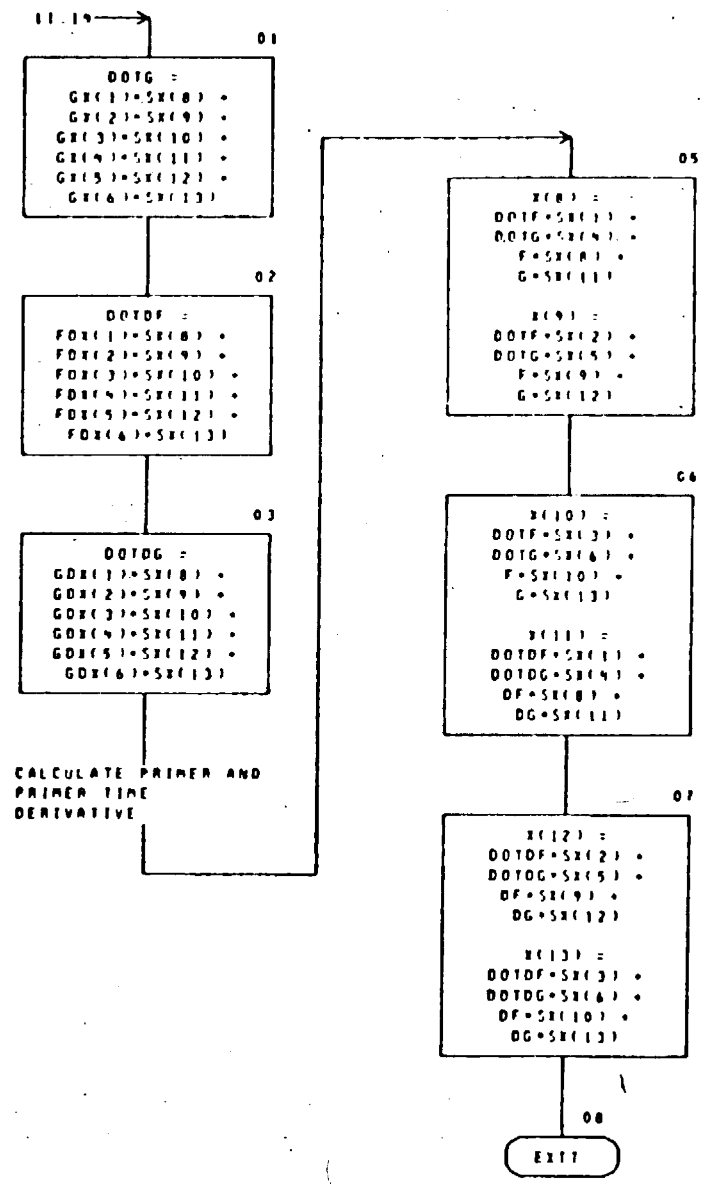


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL GO
DIMENSION      ROX(6),DOX(6),ALFX(6),ZX(6),RX(6),PARX(6),
S1X(6),S2X(6),S3X(6),S4X(6),FDX(6),GDX(6),S(6)
COMMON /REAL8/ RO1(645),F,G,FX(6),GX(6),RO2(488),RT,RO3,
RC,RO4(50),SX(50),X(50),XD(50),SBETA,BETA,RG5(648)
```

Name: BEGIN

Calling Argument: None

Referenced Sub-programs: None

Referenced Commons: EXTREM, INTGR4, ITERAT, ITER2, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: MAIN

Discussion: Subroutine BEGIN is called once per run, from the top of the MAIN program at the beginning of execution, and performs the function of initializing, or setting the default values of, the program input quantities. In addition, the subroutine zeroes out the locations of all major program COMMON statements and computes the basic unit conversion factors used by the program, which are essentially based on two numbers. These two numbers are (1) the gravitational constant of the sun, $1.32715445 \times 10^{20}$ m^3/sec^2 = SUNMU, and (2) the number of meters in one astronomical unit, 1.49599×10^{11} meters = CONDS. The routine also sets the obliquity of the ecliptic $\epsilon = 23^{\circ}.45$.

BEGIN EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
E	SU	REAL8	Obliquity of the ecliptic, ϵ, in radians.
AN	S	REAL8	Trajectory-integration exponent in regularization formula.
AR	S	REAL8	Desired final extra-ecliptic perihelion distance, in AU.
BI	S	REAL8	Efficiency coefficient b in equation for efficiency.

BEGIN EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
BX(5, 70)	SE	ITERAT	Iterator independent variable array.
BY(3, 70)	S	ITERAT	Iterator dependent variable array.
CE	S	REAL8	Cosine of obliquity of the ecliptic, $\cos \epsilon$.
DI	S	REAL8	Efficiency coefficient d in equation for efficiency, in km/sec.
GM(70)	S	SOLSYS	Array of planetary gravitational constants, in m^3/sec^2.
PI	SU	REAL8	Ratio of circumference of circle to diameter, π.
SE	S	REAL8	Sine of obliquity of the ecliptic, $\sin \epsilon$.
T2(10)	SU	REAL8	Initial estimates of swingby-continuation-trajectory-segment flight times, in days.
X0(7)	S	REAL8	Spacecraft initial state vector, x_o, $\dot{x}_o$, M_o, where x_o is in AU, $\dot{x}_o$ is in EMOS, and $M_o = 1$ is set elsewhere.
APL(2, 70)	S	SOLSYS	Array of planet names.
DEG	SU	REAL8	Conversion factor between radians and degrees; number of degrees in one radian.
GAP	S	REAL8	Propulsion-corner proximity tolerance-interval.
IRK	S	INTGR4	Numerical integration option indicator (currently not used).
IRL	SE	INTGR4	Primer-origin-proximity step-size-control indicator.

BEGIN EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ITF	S	INTGR4	Estimated time remaining to halt computer run with full printout, in case of proximity to maximum machine time, in seconds.
RAP	S	REAL8	Apoapse distance of capture orbit about primary target, in planet radii.
SAI	S	REAL8	Semi-major axis of primary-target orbit, in AU.
TDV	S	REAL8	Time of deep space burn, in days.
TGO	S	REAL8	Ballistic trajectory-extension print option indicator.
CONG	S	REAL8	Conversion factor between specific impulse and jet exhaust speed, in meters/sec^2.
CONX(70)	S	ITERAT	Array of print conversion factors for iterator independent-variables.
CONY(70)	S	ITERAT	Array of print conversion factors for iterator dependent-variables.
HOUR	S	REAL8	Hour-of-day of reference date.
MDAY	S	INTGR4	Day-of-month of reference date.
MODE	S	INTGR4	Power variation option selector.
NSET(5)	SU	INTGR4	Iteration-sequence control array.
RPER	S	REAL8	Periapse distance of capture orbit about primary target, in planet radii.
TOFF(20)	S	REAL8	Array of times, from the start of the trajectory, at which imposed coast phases are to begin, in days.
CONA0	SU	REAL8	Acceleration conversion factor, from AU/tau^2 to meters/sec^2.

BEGIN EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
CONDS	SU	REAL8	Distance conversion factor, AU to meters.
CONPW	S	REAL8	Power conversion factor, in m^2/sec^3.
CONSP	SU	REAL8	Speed conversion factor, from AU/tau to meters/second.
CONTM	SU	REAL8	Time conversion factor, tau to days.
CTANK	S	REAL8	Electric propulsion system propellant tankage factor, k_t.
CTRET	S	REAL8	Retro tankage factor, k_{rt}, for retro maneuver at the primary target.
ERROR	S	LOGIC4	Program master error indicator.
MONTH	S	INTGR4	Month-of-year of reference date.
MOPT3	S	INTGR4	Planet-number of primary target.
MYEAR	S	INTGR4	Year of reference date.
NDIST	S	INTGR4	Identification number of celestial body to be used as the reference for the communication distance and angle measurement printed in the Extremum Point Summary Table.
NHUNG	S	INTGR4	Maximum number of propulsion-corner-proximity occurrences allowed in a given iteration-sequence.
NTAPE	S	INTGR4	Specifies the unit-number for the trajectory-tape.
PSIGN	S	REAL8	Coefficient defining the sense of the launch hyperbolic excess velocity relative to the initial primer vector.

BEGIN EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
STATE(6)	S	REAL8	Array containing the Cartesian position and velocity components of the primary target, in AU and EMOS.
STEP1	S	REAL8	Thrust-phase computation step-size, Δu.
STEP2	S	REAL8	Coast-phase computation step size, $\Delta\beta$.
SUNMU	SU	REAL8	Gravitational constant of the sun, in m^3/sec^2.
THRET	S	REAL8	Retro-stage thrust in retro maneuver at the primary target, in pounds.
TPMAX	S	REAL8	Obsolete variable.
TWOPI	S	REAL8	Twice pi, 2π.
ALPHAA	S	REAL8	Specific mass of solar arrays, α_a, in kg/kw.
ALPHAT	S	REAL8	Specific mass of thruster subsystem, α_t, in kg/kw.
CONLBS	S	REAL8	Conversion factor, newtons to pounds.
CONVRG	S	LOGIC4	Iteration-sequence convergence indicator.
FPSNMH	S	REAL8	Conversion factor, knots to fps.
GAMMAX	S	REAL8	Maximum permissible value of the power function, γ.
MAXHAM	S	INTGR4	Maximum number of times program will check Hamiltonian constancy.
MUPDAT	S	INTGR4	Indicator for whether iterator independent variables at the finish of the iteration sequence of a given case are to be used as initial guesses for the next case.

BEGIN EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
NPRINT	S	INTGR4	Printout amount selection indicator.
NSWPAR	S	INTGR4	Iterator independent-variable perturbation-increment control indicator.
POWFIX	S	REAL8	Launch-vehicle-independent trajectory option indicator, in which the value of POWFIX is the spacecraft's reference power in kilowatts.
RADIUS(70)	S	SOLSYS	Array of planetary-body radii, in meters.
RADODD	S	REAL8	Radius of primary target; in meters.
SPIRET	S	REAL8	Retro-stage specific impulse pertaining to the retro-maneuver at the primary target, in seconds.
TPOWER	S	REAL8	Solar-cell degradation characteristic-time, in days.
TSCALE	S	REAL8	Iterator dependent-variable tolerance-interval scaling factor.

CHART TITLE - SUBROUTINE BEGIN

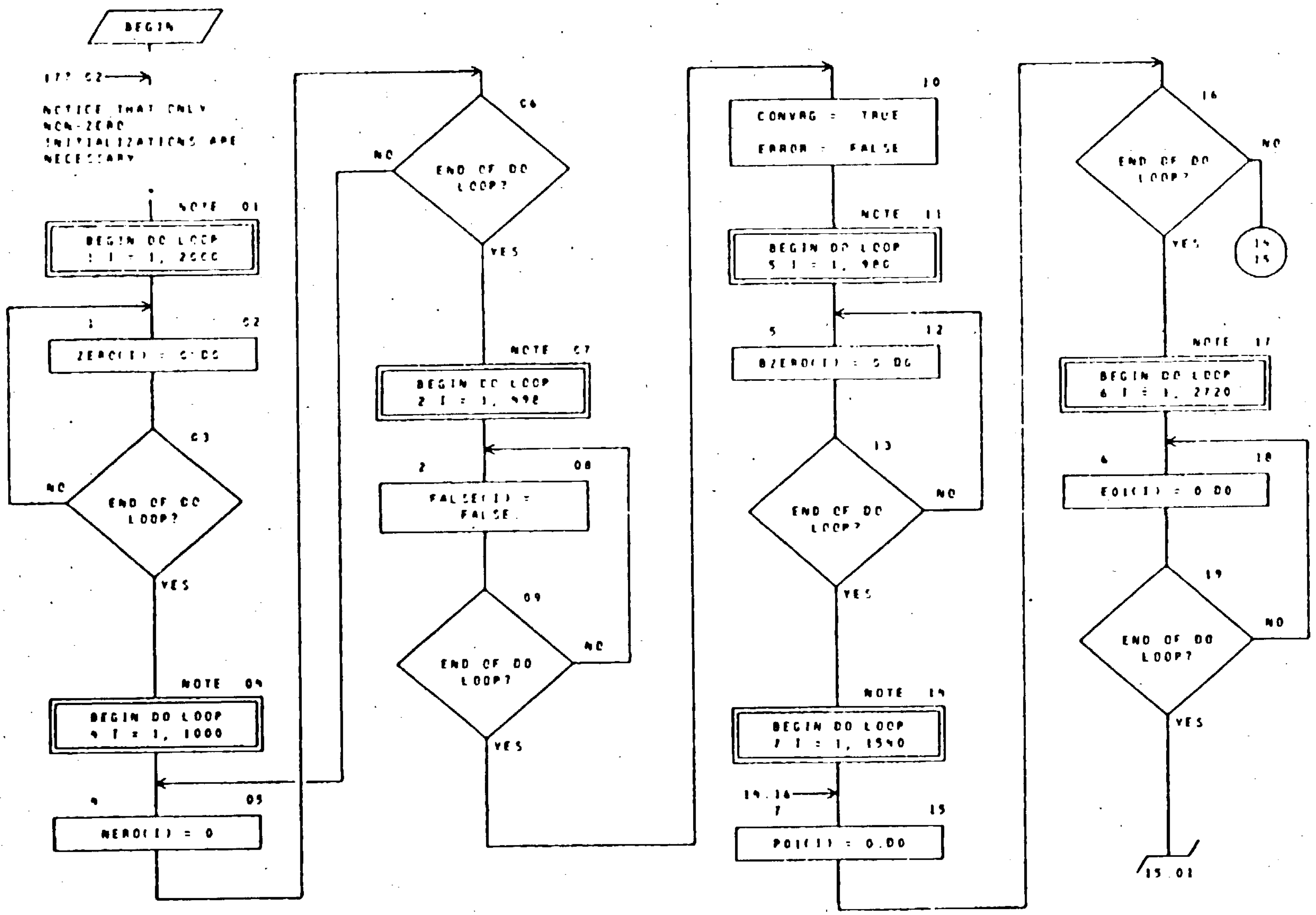


CHART TITLE - SUBROUTINE BEGIN

ORIGINAL PAGE IS OF POOR QUALITY

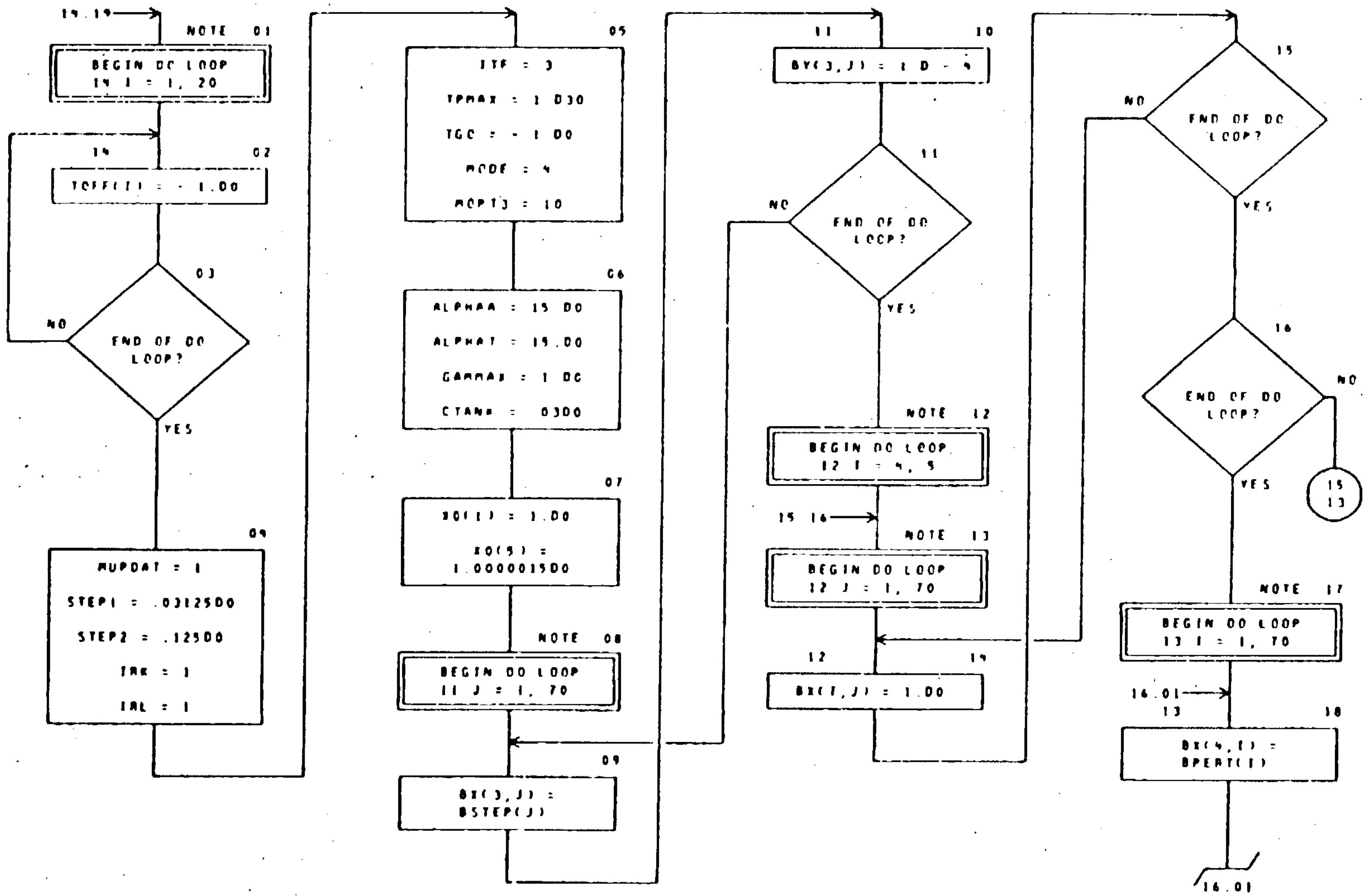


CHART TITLE - SUBROUTINE BEGIN

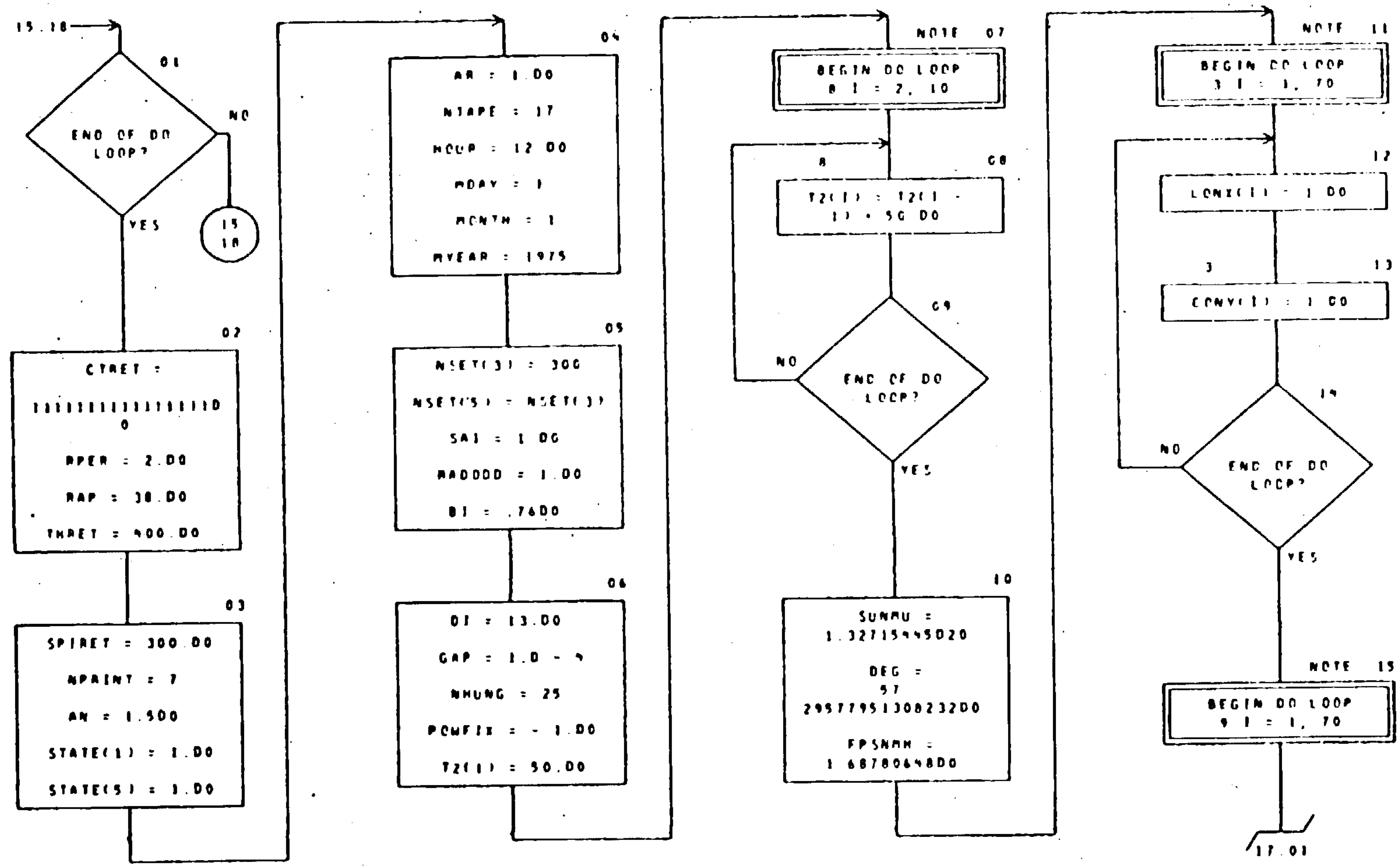


CHART TITLE - SUBROUTINE BEGIN

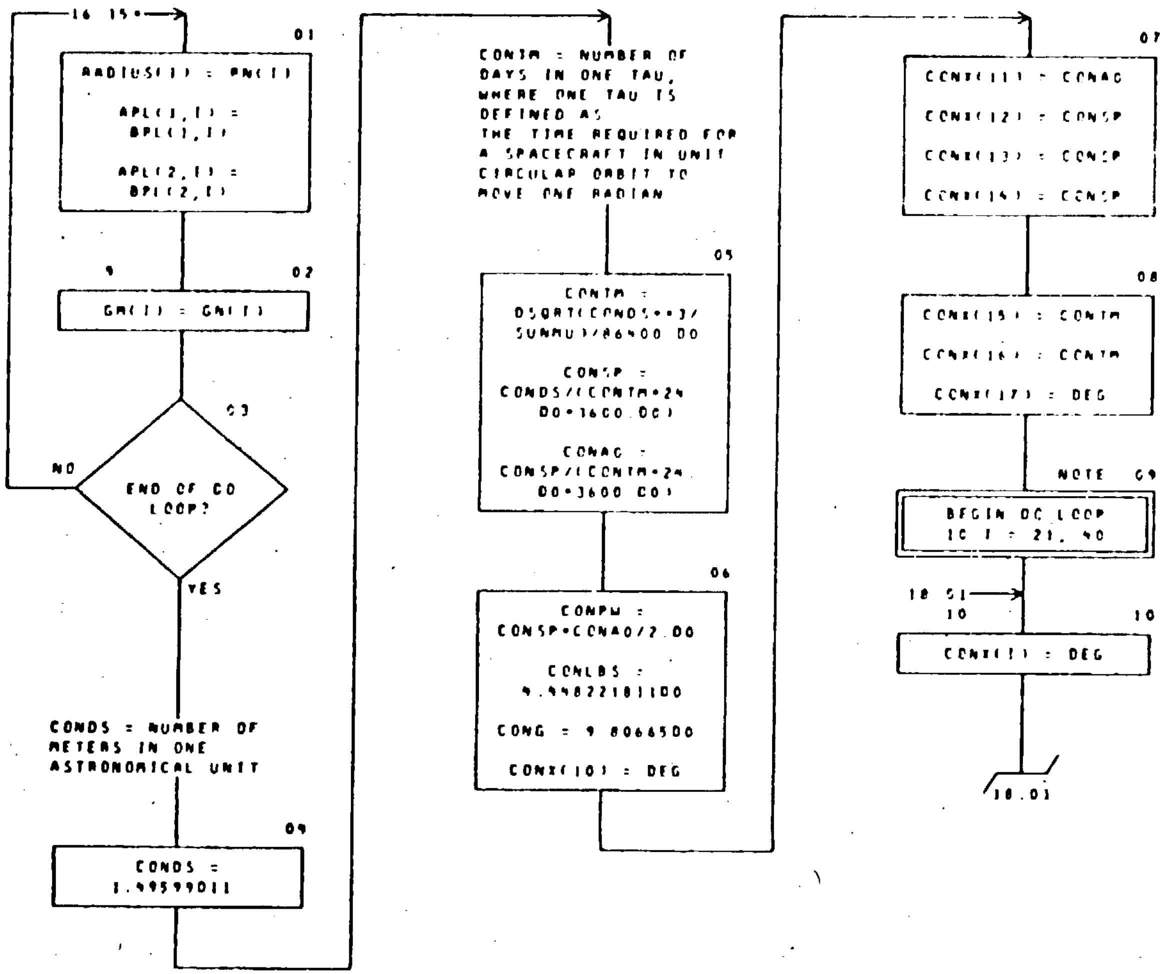
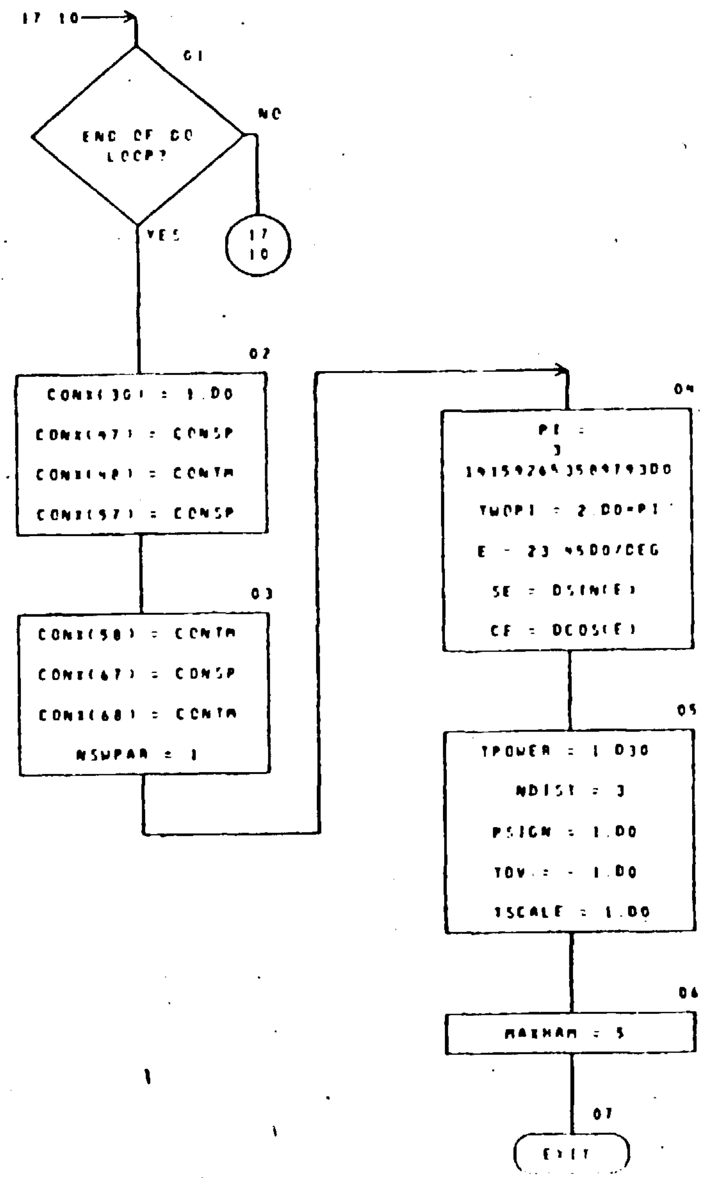


CHART TITLE - SUBROUTINE BEGIN



BEGIN-11

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL ERROR,CONVRG,FALSE
DIMENSION GN(70),RN(70),BPL(2,70),ZERO(2000),NERO(1000),BSTEP(70)
,BZERO( 980),BPERT(70)
COMMON /REAL8/ R01(9),CTANK,R02(10),CTRET,RPER,RAP,THRET,
SPIRET,R03(16),HOUR,TPMAX,R04,STATE(6),X0(7),R05(7),BI,DI,
R06(4),POWFIX,AR,R07(5),SAI,R08(6),RADODD,R13,STEP1,STEP2,R09,
T2(10),R10,TDV,R20,
            GAP,R11(66),E,SE,CE,R15(4),TGO,R17(88),
ALPHAA,ALPHAT,R21(33),
AN,R12(6),PSIGN,TSCALE,R18(3),DEG,EPSNMH,SUNMU,CONTM,CONSP,
CONAO,CONPW,PI,TWOPI,CONDS,CONLBS,CONG, R19(201),
 TOFF(20),R14(588),GAMMAX,R22(54),TPOWER,R16(811)
COMMON /INTGR4/ IRL,MODF,IRK,ITF,I02(4),NTAPE,MDAY,MONTH,MYEAR,
I03,MUPDAT,I04,NSET(5),I05(4),MOPT3,I06(16),NPRINT,I07(4),
NSWPAR,I08(4),NHUNG,I09(89),NDIST,I10(4),MAXHAM,I01(853)
COMMON /LOGIC4/ ERROR,CONVRG,FALSE(498)
COMMON /ITERAT/ BX(5,70),BY(3,70),CONX(70),CONY(70),B01(280)
COMMON /ITER2/ P01(1540)
COMMON /EXTREM/ E01(2720)
COMMON /SOLSYS/ GM(70),RADIUS(70),APL(2,70)
EQUIVALENCE (R01(1),ZERO(1)),(IRL,NERO(1)),(BZERO(1),BX(1,1))
DATAGN/2.17562D13,3.248534D14,3.986032D14,.429778D14,
           1.267069D17,3.791794D16,5.786726D15,6.876309D15,
3.317819D14,7.96D10,60*0.D0/
```

ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
DATA AN /2.5D6,6.1D6,6.3781650A,3.415D6          ,6.988D7,
5.754D7,2.55D7,2.5D7,6.349D6, 385D6,60*1.D0/
DATA BPL /
8H      MER,5HCURY,8H      VEN,2HUS,8H      EAR,2HTH,
8H      MAR,1HS                  ,8H      JUP,5HITER,
8H      SAT,3HURN,8H      URA,3HNUS,8H      NEP,5HTUNE,
8H      PLU,2HTO,8H      CER,2HES,8H  INPUT ,6HTARGET,
8H D'ARRES,7HT(1982),8H  ENCKE(,5H1980),8H  ICARUS,6H(1978),
8H      ERO,1HS,8HGEOGRAPH,8HOS(1983),8H  ENCKE(,5H1977),
8H  ENCKE(,5H1984),8H  ENCKE(,5H1987),8H      HAL,3HLEY,
8H     BETU,3HLIA,8H   TORO(,5H1983),8H      PAL,3HLAS,
8H       JU,2HNO,8H      VES,2HTA,8H     ASTR,3HAEA,
8H       HE,2HBE,8H       IR,2HIS,8H      FLO,2HRA,
8H     ACHI,5HLLES,8H       AM,2HOR,8H     HIDA,3HLGO,
8H      ALI,3HNDA,8HGRIGG-SK,8HJEL-1977,8H      KOP,2HFF,
8HGRIGG-SK,8HJEL-1982,8H     GANY,3HMED,8H       IV,2HAR,
8H      BET,2HRA,8H      KEP,3HLER,8HGIACOB-Z,8HIN(1985),
8H BORRELL,7HY(1987),8HTEMPEL I,7HI(1988),8HTEMPEL I,7HI(1983),
8HTUTL-GIA,8HC-KRESAK,8H    SCHAU,5HMASSE,8HHONDA-MP,8HKOS-PAJD,
8HGIACOB-Z,8HIN(1979),8H  ICAPUS,6H(1987),8H   TORO(,5H1987),
8HGEOGRAPH,8HOS(1987),      38*0.D0/
DATA BSTEP /9*3.D0,9.D1,1.D-3,2.D3,2*5.D2,8.D0,1.D2,1.D1,3*1.D0,
20*10.D0,6*3.D0,5.D2,5.D1,8*3.D0,5.D2,5.D1,8*3.D0,5.D2,5.D1,2*3.D0
/
DATA BPERT /6*1.D-8,5*1.D-6,1.D-11,1.D-3,1.D-4,1.D-3,2*1.D-6,
```

ORIGINAL PAGE IS
OF POOR QUALITY

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
24*1.D-4,6*1.D-8,1.D-4,1.D-6,8*1.D-8,1.D-4,1.D-6,8*1.D-8,1.D-4,
1.D-6,2*1.D-8/
```

Name: CARKEP

Calling Argument: ENP, A, E, EYEP, SOMP, OMP, G, H, VP, SRP, R, RDH, EMU

Referenced Sub-programs: VCROSS, VDOT, VMAG

Referenced Commons: None

Entry Points: None

Referencing Sub-programs: SPRINT

Discussion: CARKEP calculates classical orbital elements using the Cartesian input position and velocity. The computational algorithm is specified by the following:

The angular momentum vector, H, is given by

$$H = R \times \dot{R}$$

The inclination, i, is given by

$$i = \cos^{-1} (H \cdot \hat{e}_n / h)$$

where $\hat{e}_n$ is the unit pole vector of the basic reference system and

$$h = |H|$$

The longitude of the ascending node, Ω, is given by

$$\Omega = \tan^{-1} \left[\frac{\hat{y}_p \cdot (\hat{e}_n \times H)}{\hat{x}_p \cdot (\hat{e}_n \times H)} \right] + \pi$$

where $\hat{y}_p$ and $\hat{x}_p$ are axes defining the reference system given by
$\hat{y}_p = [\hat{i} \times \hat{e}_n]$, $\hat{x}_p = [\hat{y}_p \times \hat{e}_n]$.

The argument of position, ω, measured in the orbit plane from the ascending node is given by

$$\omega = \tan^{-1}\left[\frac{R \cdot [H \times (\hat{e}_n \times H)]}{R \cdot [\hat{e}_n \times H]\, h}\right]$$

The semi-major axis is given by

$$a = \frac{1}{\frac{2}{r} - \frac{v^2}{\mu}}$$

The flight path angle is given by

$$\gamma = \sin^{-1}\left(\frac{R \cdot V}{r v}\right)$$

The eccentricity is given by

$$e = \left[1 - \frac{h^2}{\mu a}\right]^{\frac{1}{2}}$$

CARKEP EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
ENP(3)	UX		Unit pole vector $\hat{e}_n$.
A	SUX		Semi-major axis, a .
E	SX		Eccentricity, e.
EYEP	SX		Inclination, i, radians.
SOMP	SX		Argument of position, ω, radians.
OMP	SX		Longitude of ascending node, Ω, radians.
G	SX		Flight path angle, γ, radians.

CARKEP EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
H(3)	SUX		Angular momentum vector, H.
VP	SUX		Magnitude of the velocity, v.
SRP	SUX		Magnitude of the position, r.
R(3)	UX		Position vector, R.
RDH(3)	UX		Velocity vector, V.
EMU	UX		Gravitational parameter, μ.

Output units are consistent with input units.

CHART TITLE - SUBROUTINE CARKEP(ENP,A,E,EVEP,SOMP,OMP,G,H,VP,SRP,R,RDH,EMU)

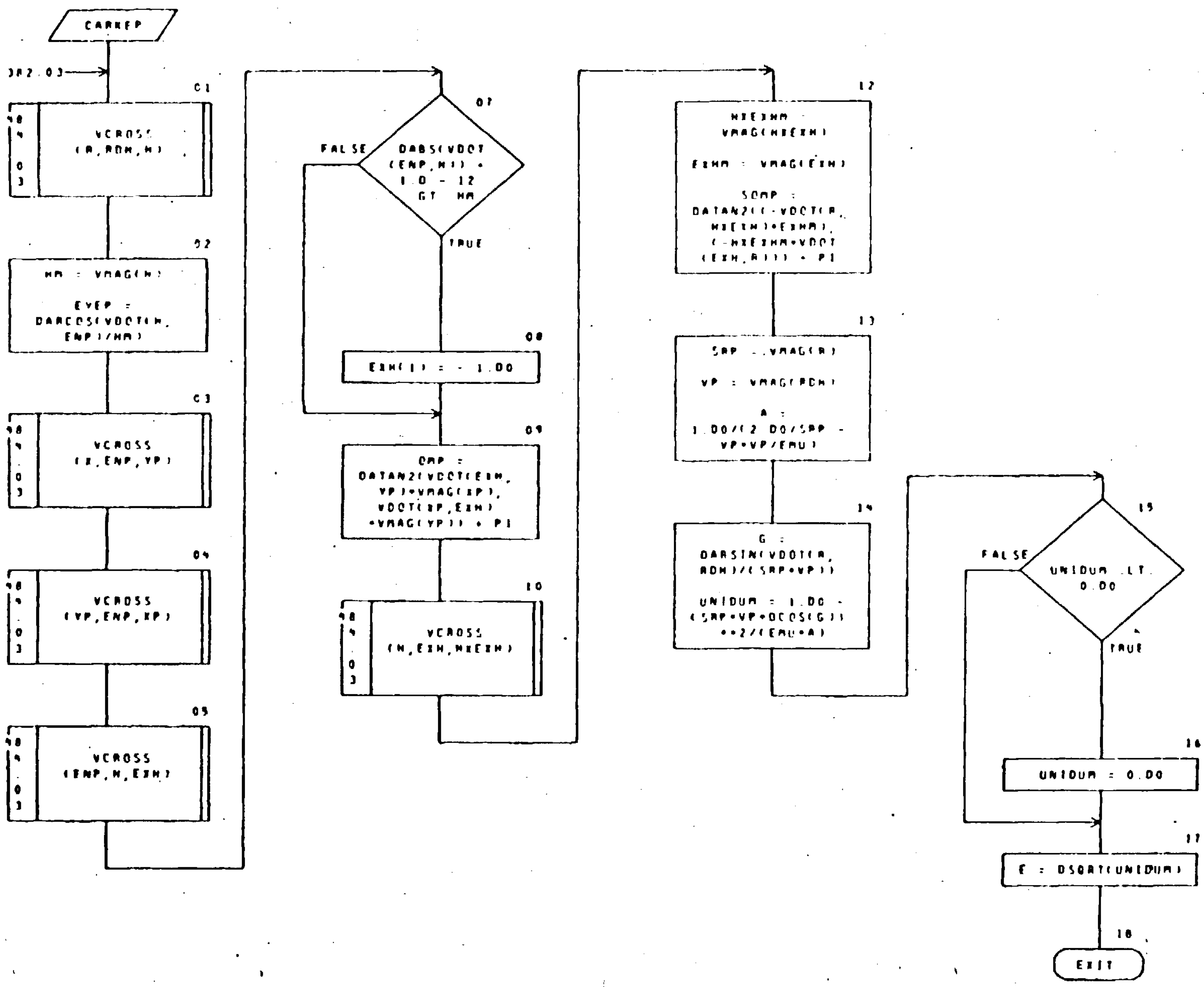


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION R(3),RDH(3),H(3),ENP(3),EXH(3),HXEXH(3),X(3),YP(3),XP(3)
DATA PI /3.14159265358979D0/
DATA X /1.D0,0.D0,0.D0/
```

<u>Name:</u> CDERIV

<u>Calling Argument:</u> None

<u>Referenced Sub-programs:</u> RADAR, THANGD, UNITD, VADD, VCROSS, VDOT VMAG, VSCAL, VSUB

<u>Referenced Commons:</u> EXTREM, INTGR4, ITERAT, LOGIC4, REAL8

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> CHECK, INTERP, TAP

<u>Discussion:</u> CDERIV computes all the functions monitored by subroutine CHECK, in which CHECK controls the iterations to isolate the function roots; many of these functions are the time-derivatives of functions for which the isolation of extrema is desired, and therefore the name CDERIV may be thought of as the derivative routine associated with CHECK.

Extrema points are evaluated by locating points at which the derivatives of the functions go to zero. The other special points (e.g., thrust switch points) are obtained by defining a function which goes to zero at the special point (e.g., the thrust switch function σ) and isolating these roots. The B array contains the monitored functions. B(1, i) contains the value of the i^{th} function at the beginning of the current computation step (stored value), and B(2, i) contains the current value. $i = 30$ denotes storage for the trajectory independent variable β, which of course is not a monitored function. The contents of B are as follows:

<u>i</u>	<u>function</u>
1	End of trajectory segment, $t - t_i$.
2	Thrust switch function, σ.
3	Thrust switch function time derivative, $\dot{\sigma}$.
4	Time derivative of communications distance, $\dot{r}_{com}$.
5	Solar distance time derivative, $\dot{r}$.
6	Thrust angle ψ time derivative, $\dot{\psi}$.

(continued)

PRECEDING PAGE BLANK NOT FILMED

$\underline{i}$	$\underline{\text{function}}$
7	Thrust angle θ time derivative, $\dot{\theta}$.
8	Thrust cone-angle ϕ time derivative, $\dot{\phi}$.
9	Critical solar distance, $r - r_c$.
10	Two-dimensional fixed thrust-angle flip-flop condition, $R(1)\ \Lambda(2) - R(2)\ \Lambda(1)$.
11	Time derivative of communications angle, $\dot{\theta}_{com}$.
12	Time derivative of power function, $d(q\gamma)/dt$.
13	Solar array orientation power-curve boundary function, f_{ch1}.
14	Solar array orientation edgewise function, f_{ch2}.

These functions are either computed in CDERIV or obtained directly from other subroutines, as follows:

(1) End of trajectory segment, $t - t_i$, is computed directly in CDERIV.

(2) Thrust switch function, σ, is obtained from subroutine FUNCT.

(3) Thrust switch function time derivative, $\dot{\sigma}$, is obtained from subroutine FUNCT.

(4) The communication distance is defined to be the distance from the Earth to the spacecraft and is given by

$$r_{com} = |R - P_e| ,$$

where P_e is the heliocentric position of the Earth, or other reference body. The extrema in r_{com} are evaluated by locating the points when

$$(R - P_e) \cdot (\dot{R} - \dot{P}_e) = 0 .$$

(5) The solar distance is simply the magnitude of the spacecraft's heliocentric position vector. Extrema of the function are obtained by isolating those points where $\dot{r} = R \cdot \dot{R}/r$ is zero.

(6), (7), and (8) The extrema of the thrust angles ψ, θ and ϕ are defined by isolating the points at which their time derivatives vanish. For the case of unconstrained thrust angles, these derivatives are defined:

$$\dot{\psi} = \frac{1}{\cos\psi}\left(\dot{\bar{e}}_t \cdot \bar{e}_h + \bar{e}_t \cdot \dot{\bar{e}}_h\right),$$

$$\dot{\phi} = -\frac{1}{\sin\phi}\left(\dot{\bar{e}}_t \cdot \bar{e}_r + \bar{e}_t \cdot \dot{\bar{e}}_r\right),$$

$$\dot{\theta} = \frac{1}{\sin\theta\cos\psi}\left(\sin\phi\,\dot{\phi} - \cos\theta\sin\psi\,\dot{\psi}\right)$$

$$= \frac{1}{\cos\theta\cos\psi}\left(\dot{\bar{e}}_t \cdot \bar{e}_v + \bar{e}_t \cdot \dot{\bar{e}}_v + \sin\theta\sin\psi\,\dot{\psi}\right),$$

where $\dot{\bar{e}}_t$, the time derivative of the unit thrust vector, is defined for the unconstrained-thrust-angle case as

$$\dot{\bar{e}}_t = \frac{1}{|\Lambda|^3}\left[(\Lambda \times \dot{\Lambda}) \times \Lambda\right],$$

and also

$$H = R \times \dot{R},$$

$$h = |H|,$$

$$\bar{e}_h = H/h,$$

$$\dot{\bar{e}}_h = \frac{1}{h^3}\left[(H \times \dot{H}) \times H\right],$$

$$\bar{e}_r = R/r,$$

$$\dot{\bar{e}}_r = \frac{1}{r^3}\left[(R \times \dot{R}) \times R\right],$$

$$\dot{H} = h_\sigma \frac{g\gamma q}{\nu}\left(R \times \bar{e}_t\right),$$

$$\bar{e}_v = \bar{e}_h \times \bar{e}_r ,$$

$$\dot{\bar{e}}_v = -\frac{h}{r^2}\, \bar{e}_r .$$

The two expressions for $\dot{\theta}$ are employed to avoid singularities when either $\sin\theta$ or $\cos\theta$ vanish.

For the case of constrained thrust cone angle, no extrema of the cone angle ϕ is sought since the angle is a constant. The derivatives of the other two angles are as defined above using the value for $\dot{\bar{e}}_t$ generated by subroutine THANGD and setting $\dot{\phi} = 0$.

(9) Critical solar distance, $r - r_c$, which demarks a corner in the power-function (or its derivative), is computed directly in CDERIV, where $r = |R|$ is obtained from subroutine FUNCT.

(10) The thrust-direction (prograde or retrograde) switching condition when fixed-thrust-angle simulations are constrained to be two dimensional is obtained by isolating the roots of $R \times \Lambda$, which demark the times at which the primer vector is instantaneously aligned with the spacecraft position vector.

(11) Extrema of the communications angle correspond to the roots of the equation,

$$\frac{\left[(R - P_e) \times (\dot{R} - \dot{P}_e)\right] \times (R - P_e)}{|P_e|\; |R - P_e|^3} \cdot P_e + \frac{(P_e \times \dot{P}_e) \times P_e}{|R - P_e|\,|P_e|^3} \cdot (R - P_e) = 0 ,$$

where P_e is defined under (4) above.

(12) Extrema of the power function $q\gamma$ are obtained by isolating the roots of the function,

$$\frac{d(q\gamma)}{dt} = q\dot{\gamma} + \dot{q}\gamma = 0 .$$

When the solar arrays are operating at an orientation corresponding to being on the power-curve (as opposed to being below the curve), the derivatives in the above expression are given by,

$$\dot{\gamma} = \frac{R \cdot \dot{R}}{r} \left.\frac{\partial \gamma}{\partial r}\right|_{\text{curve}} ,$$

$$\dot{q} = - h_{\sigma} q d/\tau_d \, .$$

Otherwise, when power degradation is being simulated and the arrays are tilted away from the sun, corresponding to operating beneath the power curve, the expression for $\dot{\gamma}$ becomes

$$\dot{\gamma} = - \frac{\partial \gamma}{\partial d} \left[\frac{1}{g} (\dot{\nu}\lambda_s + \nu\dot{\lambda}_s) + q \frac{\partial \gamma}{\partial d} \left\{ \dot{\lambda} - \frac{1}{c} (\dot{\nu}\lambda_{\nu} + \nu\dot{\lambda}_{\nu}) - \frac{d}{\tau_d} \left(\Lambda \cdot \bar{e}_t - \frac{\nu\lambda_{\nu}}{c} \right) \right\} \right] / q \frac{\partial^2 \gamma}{\partial d^2} \left(\Lambda \cdot \bar{e}_t - \frac{\nu\lambda_{\nu}}{c} \right) .$$

(13) and (14) The two functions $f_{ch\,1}$ and $f_{ch\,2}$ which demark the switching of the solar array tilt angle χ from its limiting boundaries are obtained from subroutine SOLAR.

Functions (1), (2), (3), (5), (9), (10), (13), and (14) are monitored on all trajectories, as they are potentially important, depending on the options being simulated, in determining events which fundamentally affect the trajectory itself. The other functions, (4), (6), (7), (8), (11), and (12), are monitored only on the final, summary trajectory of each case, for the purpose of obtaining informative printout of the associated extrema.

CDERIV EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(2, 30)	SU	EXTREM	Array of monitored functions.
D	U	REAL8	Power density, d, in AU^{-2}.
O(70)	U	ITERAT	Array of iterator independent-variables; O(21) is the thrust angle, when it is held constant, in radians.
R(2)	UA	REAL8	Spacecraft solar distance, r, at start of computation step (R(1)) and instantaneously (R(2)), in AU.
X(50)	UA	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
FT	UA	REAL8	Reference thrust acceleration, g, in AU/tau^2.
JC	U	INTGR4	Counter corresponding to the JC^{th} specified time-function value (i.e., the time) isolated, or to be isolated, thus far on the current trajectory segment. Attains values greater than one when imposed coast phases are invoked.
PP(2)	UA	REAL8	Primer magnitude, λ, defined similarly to R(2).
VJ	U	REAL8	Jet exhaust speed, c, in EMOS.
XD(50)	U	REAL8	Array of trajectory dependent-variable derivatives, corresponding to X(i).
ETH(3)	A	REAL8	Thrust unit vector, $\bar{e}_t$.
EVC(6)	A	REAL8	Communications reference body position and velocity, P_e and $\dot{P}_e$, in AU and EMOS, respectively.
PHI	SU	REAL8	Thrust angle ϕ, in radians.

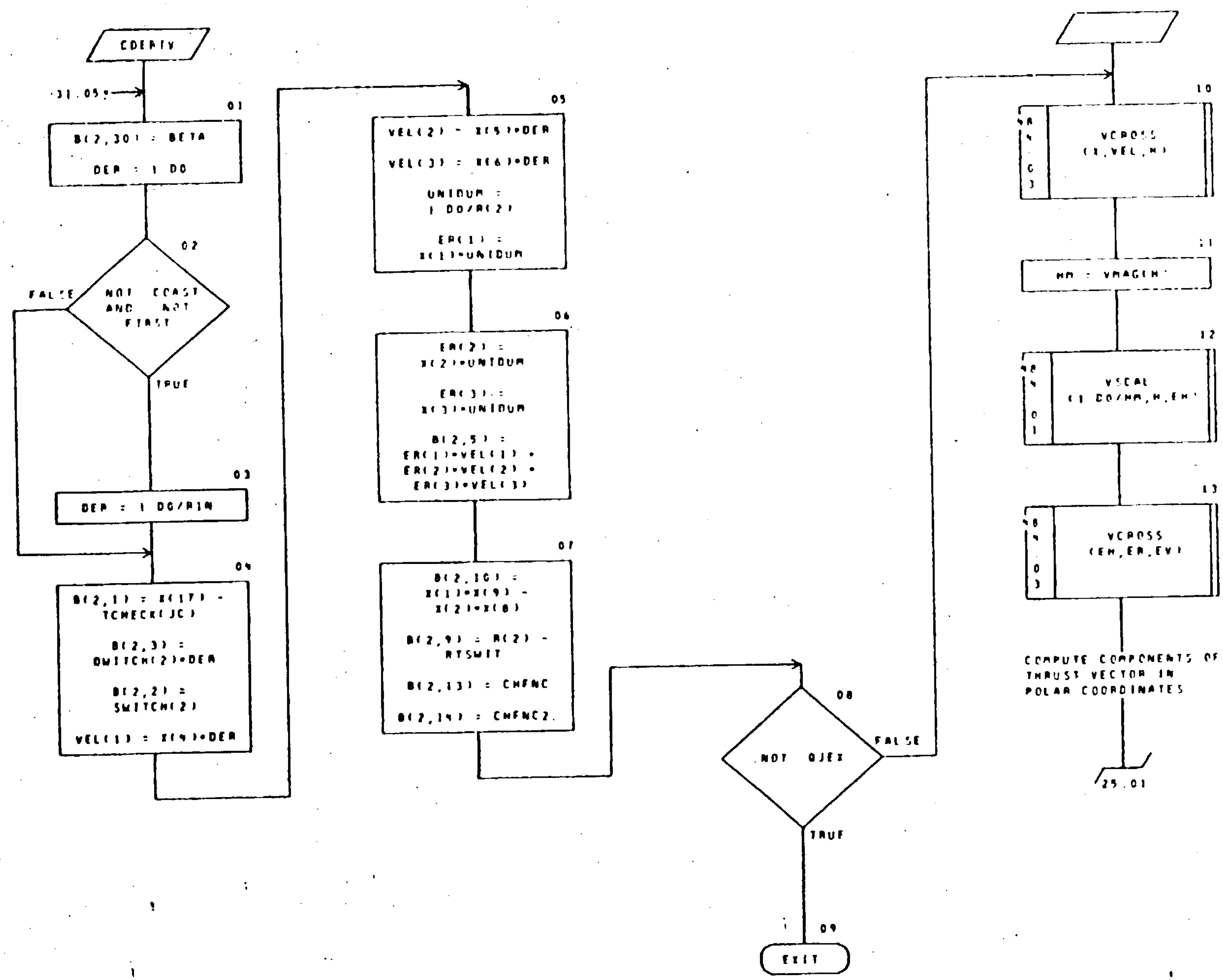
CDERIV EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
PSI	SU	REAL8	Thrust angle ψ, in radians.
R1N	U	REAL8	Conversion factor from generalized derivatives to time derivatives, r^n, in AU^n.
BETA	U	REAL8	Trajectory independent-variable, β, in $AU^{1/2}$ radians.
ETHD(3)	SA	REAL8	Thrust unit-vector time-derivative, $\dot{\bar{e}}_t$, in tau^{-1}.
POWR	SA	REAL8	Power function, $q\gamma$.
QJEX	U	LOGIC4	Detailed printout indicator.
SWIT	U	REAL8	$\sigma_1 + \sigma_2$, as defined in subroutine FUNCT.
XCOM(6)	UA	REAL8	Communication-distance vector and its time derivative, R_{com} and $\dot{R}_{com}$, in AU and EMOS, respectively.
CHFNC	U	REAL8	A function whose roots determine the switch points of the array tilt angle to and from the power-curve boundary, f_{ch1}.
COAST	U	LOGIC4	Indicator for coasting flight or thrusting flight; used in place of h_σ.
COPHI	U	REAL8	Cosine of fixed-thrust-angle, $\cos\phi_{fixed}$.
DPOWD	U	REAL8	$q\ \partial\gamma/\partial d$.
DPOWR	U	REAL8	$q\ \partial\gamma/\partial r$.
ERODE	U	LOGIC4	Power degradation option indicator.
FIRST	U	LOGIC4	Indicator for beginning of trajectory segment.

CDERIV EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
PCURV	U	LOGIC4	Indicator for condition in which solar arrays are oriented to receive the maximum power permissible under the current power-curve assumption, or to be tilted away from the maximum permissible due to degradation considerations.
PMDOT	UA	REAL8	Primer-magnitude time-derivative, $\dot{\lambda}$.
SIPHI	U	REAL8	Sine of fixed-thrust-angle, $\sin\phi_{fixed}$.
THETA	S	REAL8	Thrust angle θ, in radians.
CHFNC2	U	REAL8	A function whose roots determine the switch points of the array tilt angle to and from the stowed-edgewise boundary ($\chi = \pi/2$), $f_{ch\,2}$.
DPOWDD	U	REAL8	$q\, \partial^2\gamma/\partial d^2$.
DWITCH(2)	U	REAL8	Time derivative of thrust switch function, $\dot{\sigma}$, defined similarly to R(2).
FIXTHR	U	LOGIC4	Indicator for fixed thrust-angle.
RTSWIT	U	REAL8	Critical solar distance corresponding to a special point in the solar power curve, in AU.
SWITCH(2)	U	REAL8	Thrust switch function, σ, defined similarly to R(2).
TAUPOW	U	REAL8	Negative inverse of characteristic degradation time, $-1/\tau_d$, in tau^{-1}.
TCHECK(41)	U	REAL8	Array of time values, isolated by subroutine CHECK, in tau.

CHART TITLE - SUBROUTINE CDERIV



ORIGINAL PAGE IS OF POOR QUALITY

CDERIV-9

CHART TITLE - SUBROUTINE CDERIV

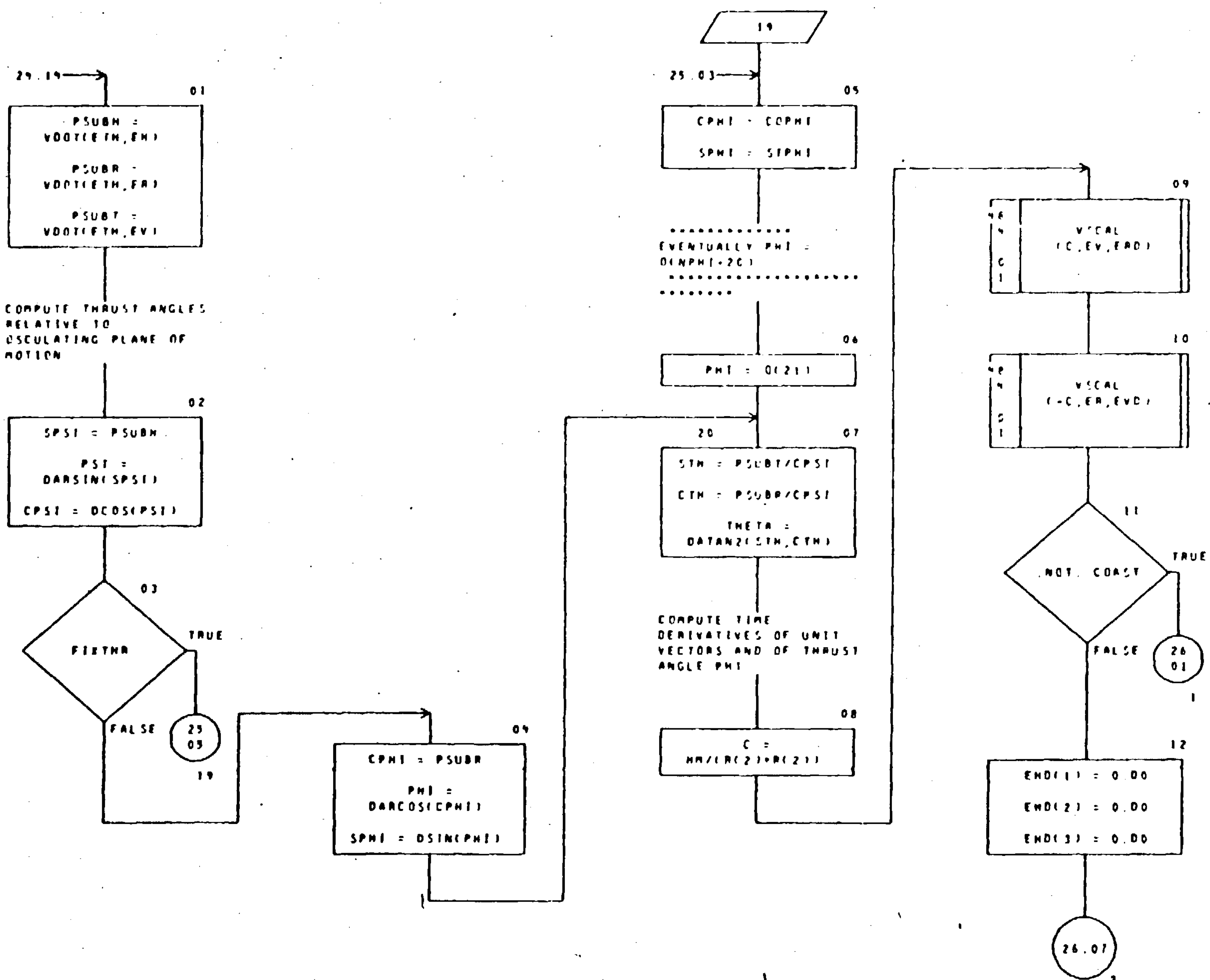


CHART TITLE - SUBROUTINE CDERIV

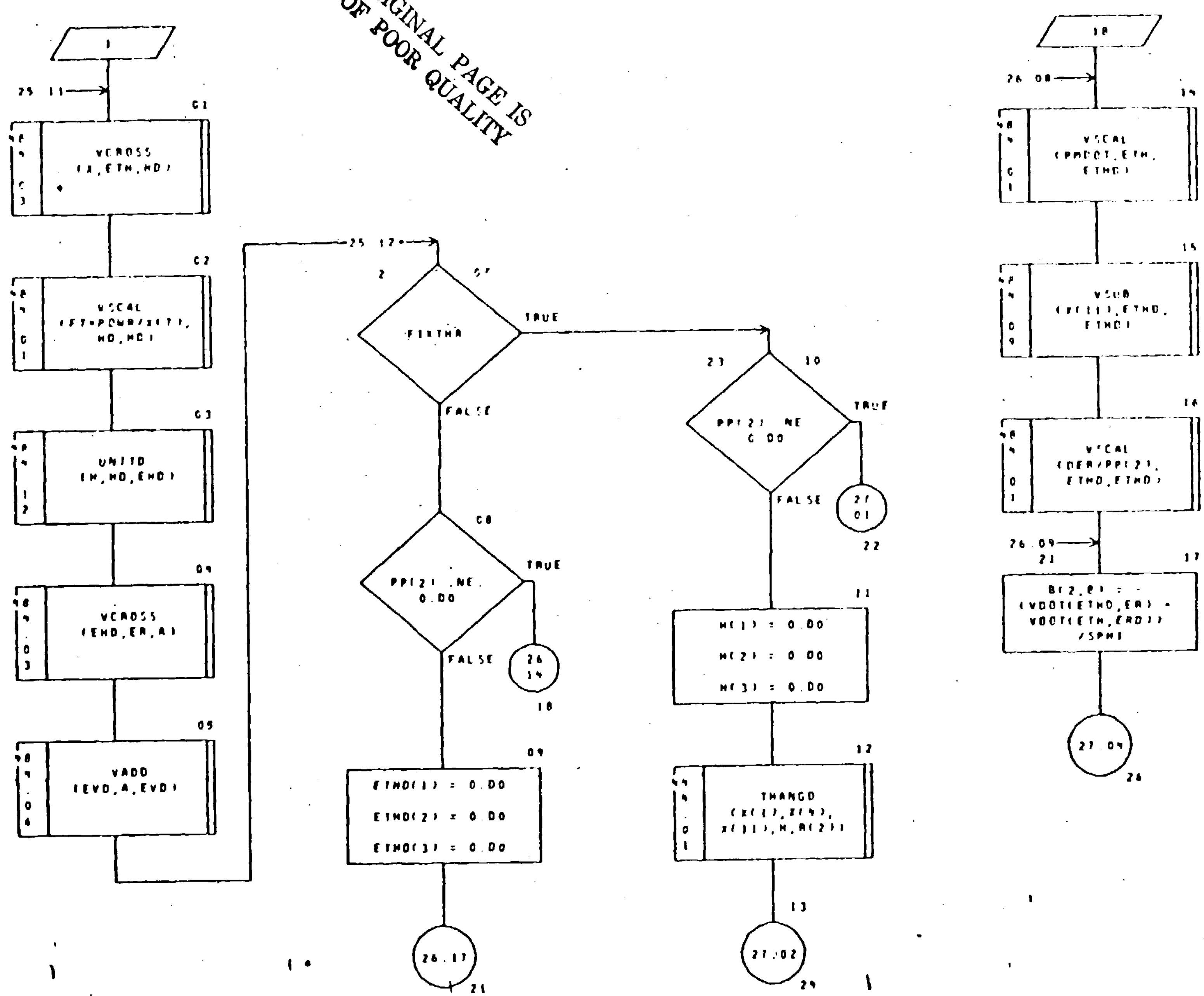


CHART TITLE - SUBROUTINE COFRIV

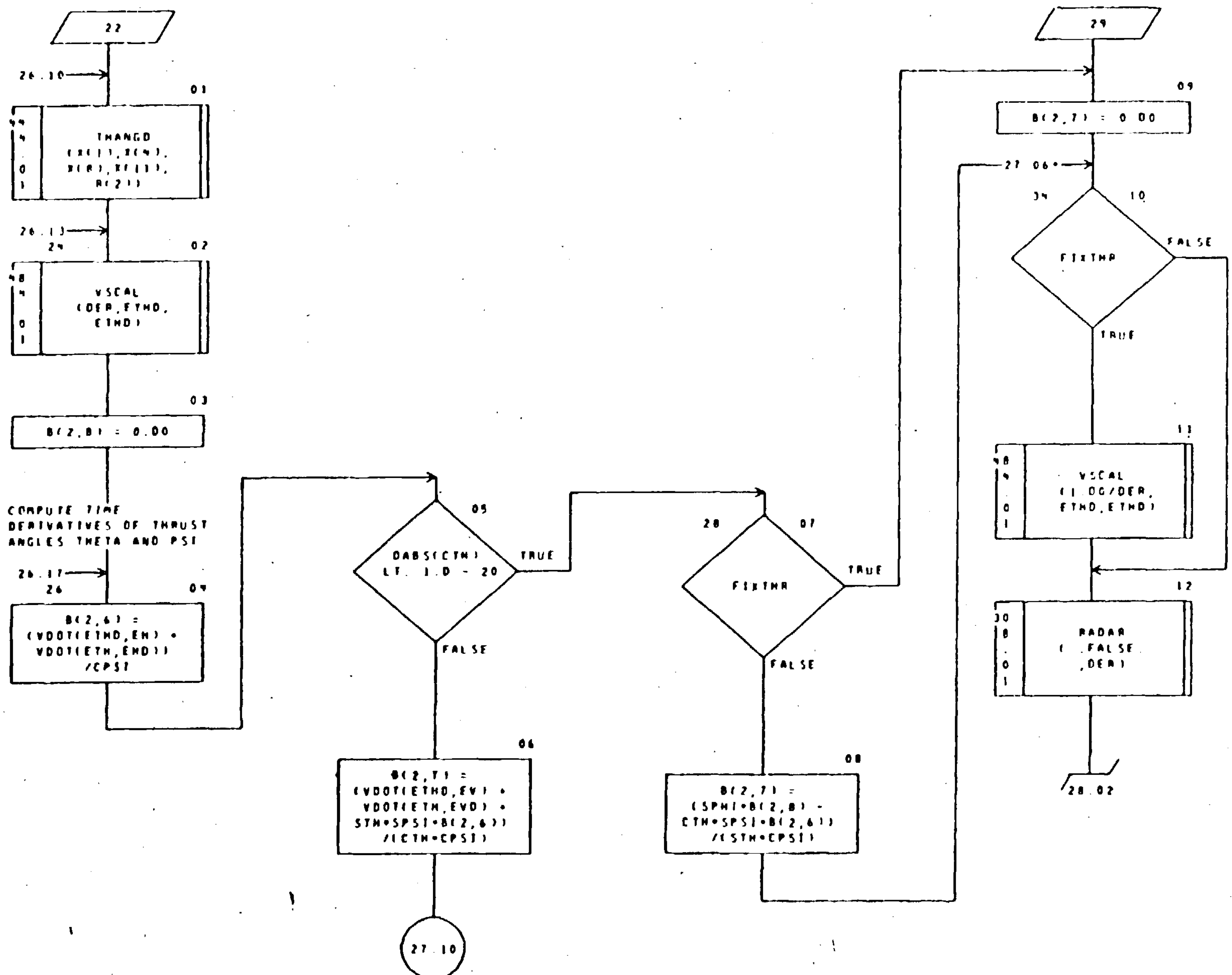
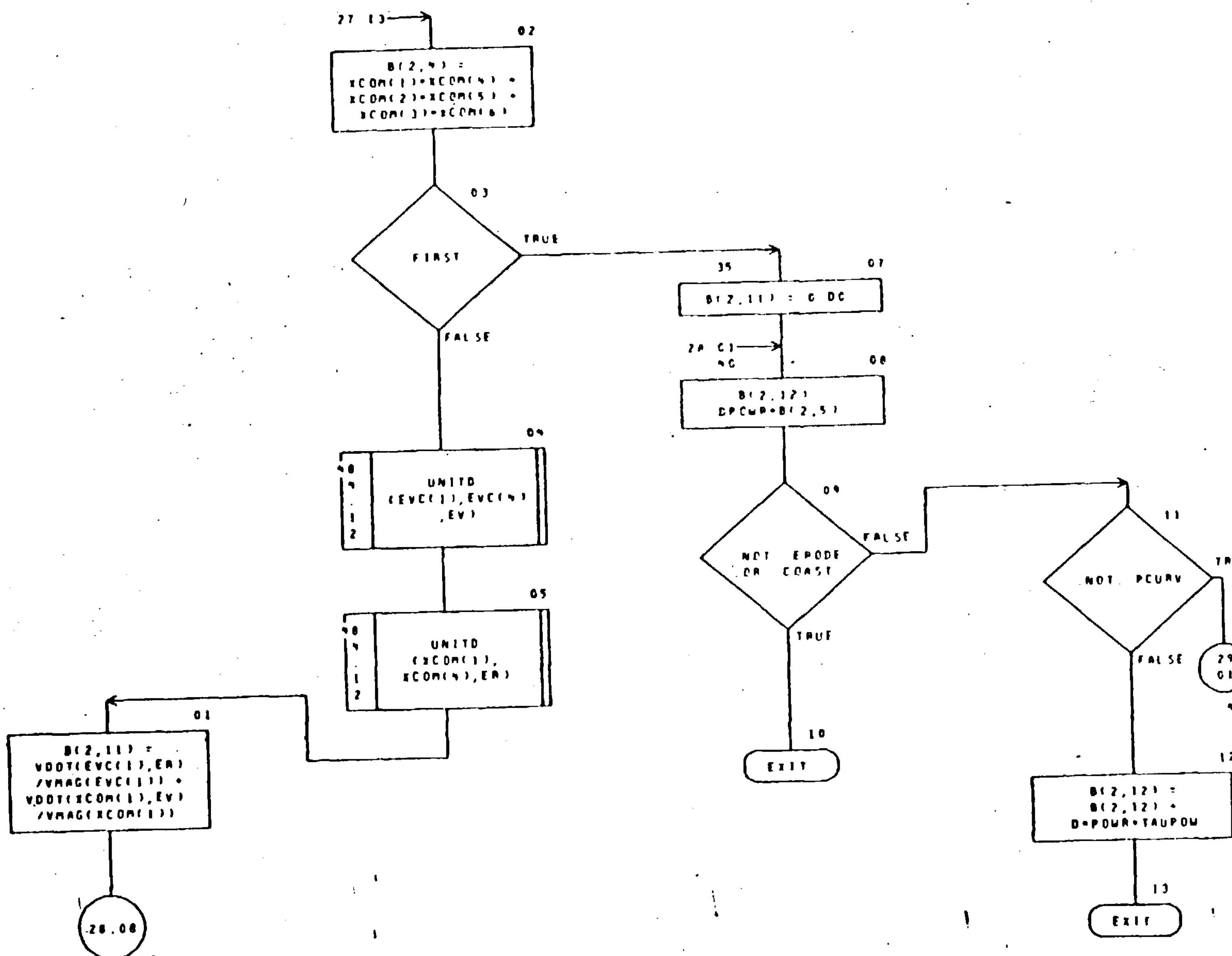
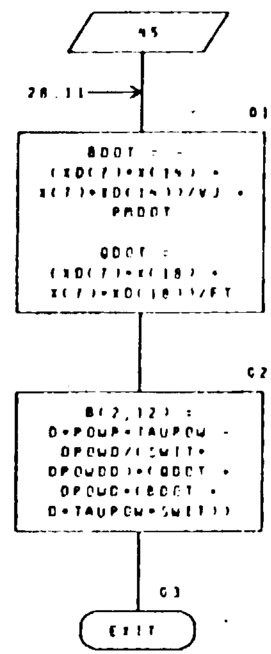


CHART TITLE - SUBROUTINE CDERIV



CDERIV-13

CHART TITLE - SUBROUTINE COFRIV



ORIGINAL PAGE IS
OF POOR QUALITY

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL COAST,FIXTHR,OJFX,FIRST,FRODE,PCURV
DIMENSION H(3),VEL(3),FR(3),FH(3),EV(3),FRD(3),EHD(3),EVD(3),A(3)
,HD(3)
COMMON /REAL8/ RO1(11),FT,VJ,RO2(35),TMAX,RO3(29),XCOM(6),RO4(58),
RTSWIT,PSI,THETA,PHI,RO5(6),EVC(6),RO6(2),R(2),PP(2),SWITCH(2),
DWITCH(2),R15(93),RIM,RO7(33),CONTM,R22(249),TCHECK(4),R16(219),
ETH(3),ETHD(3),RO8(9),SWIT,SIPHI,COPHI,
RO9(293),CHFNC2,
          R17(2),CHFMC,R18(2),PROCT,R1C(2),PRN,R19(8),POWR,
DPOWR,TAUPOW,R11(19),D,R20(5),DPOWDD,R21(5),TPOWER,R23(4),
DPOWD,R12(56),X(50),XD(50),R13,BETA,R14(498)
COMMON /INTGR4/ IO1(311),JC,JCMAX,IO2(467)
COMMON /LOGIC4/ LO1(20),FIXTHR,LO2(8),FRODE,LO5(6),OJFX,LO3(3),
FIRST,COAST,LO4(4),PCURV,LO6(96))
COMMON /ITERAT/ BG1(700),O(70),BO2(210)
COMMON /EXTREM/ EO1(2469),B(2,30)
```

<u>Name:</u> CHECK

<u>Calling Argument:</u> None

<u>Referenced Sub-programs:</u> CDERIV, FUNCT, INTERP, LOAD, SOLAR, STEP, STORE for CHECK;
STOREI for CHECKI;
CDERIV, LOAD, STORE for CHKINT

<u>Referenced Commons:</u> EXTREM, INTGR4, LOGIC4, REAL8

<u>Entry Points:</u> CHECKI, CHKINT

<u>Referencing Sub-programs:</u> TAP for CHECK;
TRAJI for CHECKI;
TAP for CHKINT

<u>Discussion:</u> This subroutine monitors the trajectory across each computation step to determine if any remarkable points have been passed. If so, CHECK executes the iteration to isolate each remarkable point by calling subroutine INTERP. The remarkable points on each trajectory are discussed in subroutine CDERIV. The absolute maximum power and maximum and minimum solar distances over the entire trajectory are also determined (in subroutine STORE).

Entry point CHECKI performs initialization associated with an entire trajectory, and entry point CHKINT performs the required initialization associated with each trajectory segment.

The following are definitions of several key variables which are not contained in the External Variables Table:

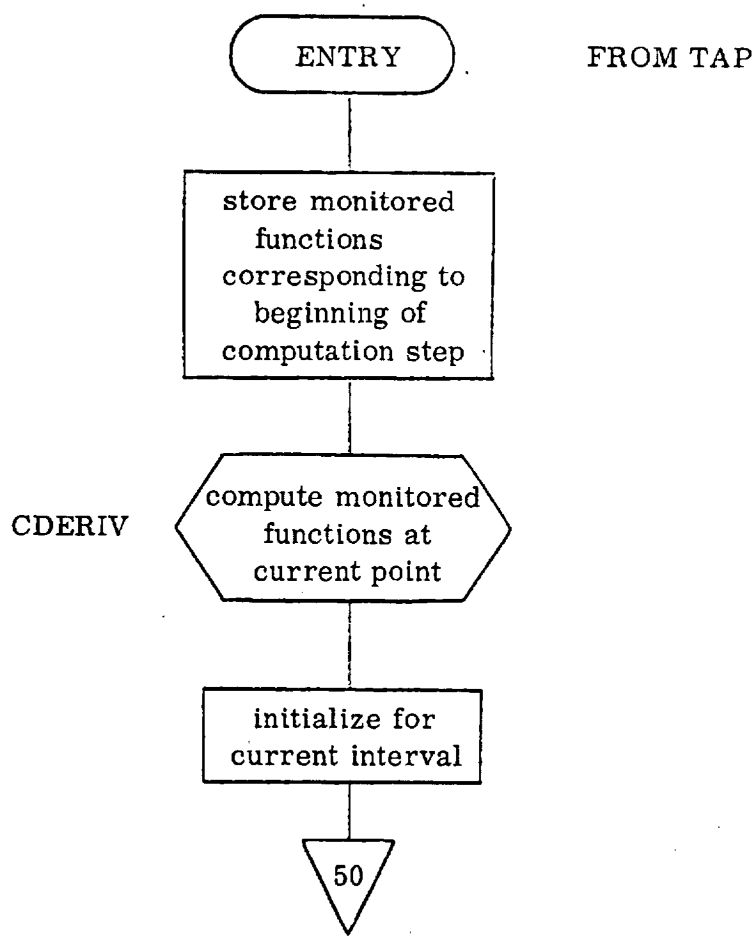
Parameter	Definition
I	Index of priority array ICHK.
J	Current number of remarkable points found in the current computing interval.
IC	Index which specifies the particular function currently being monitored.

(continued)

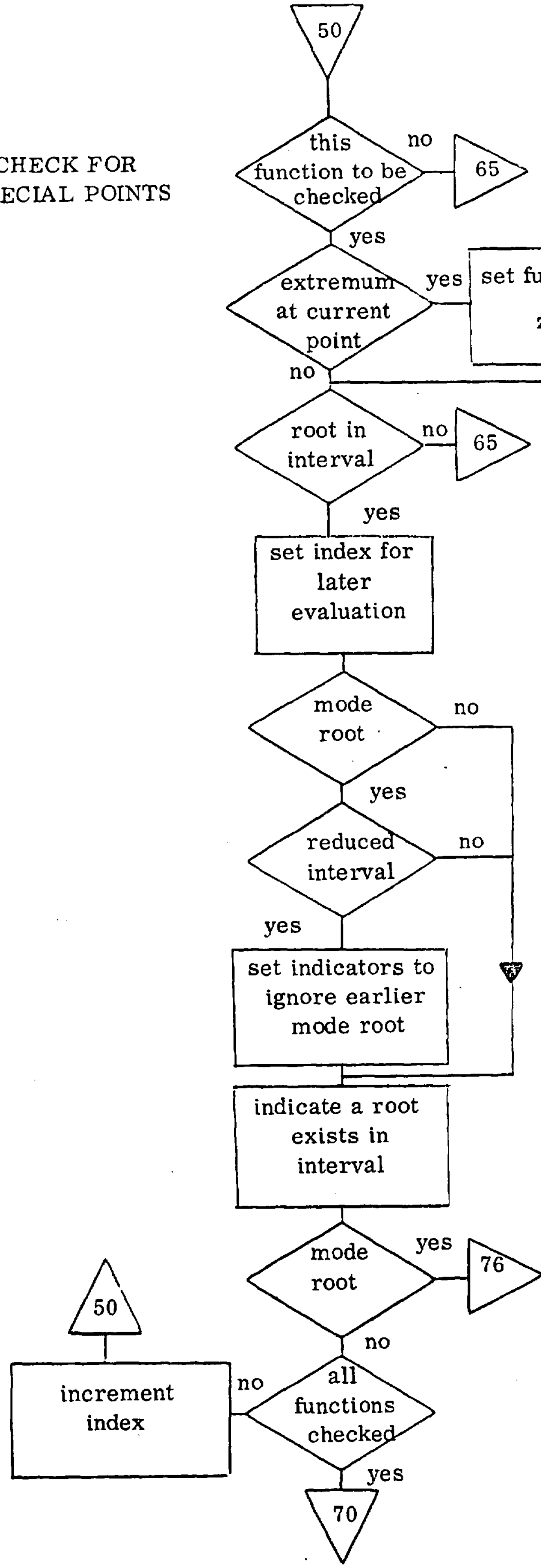
PRECEDING PAGE BLANK NOT FILMED

Parameter	Definition
ICHK	Priority array for the monitored functions, B.
JCHK	Function monitor bypass indicator array.
JMAX	Maximum permissible value for J.
NCEPS	Permanent value of NCEP.
MAXCHK	Permanent value of NCHK.

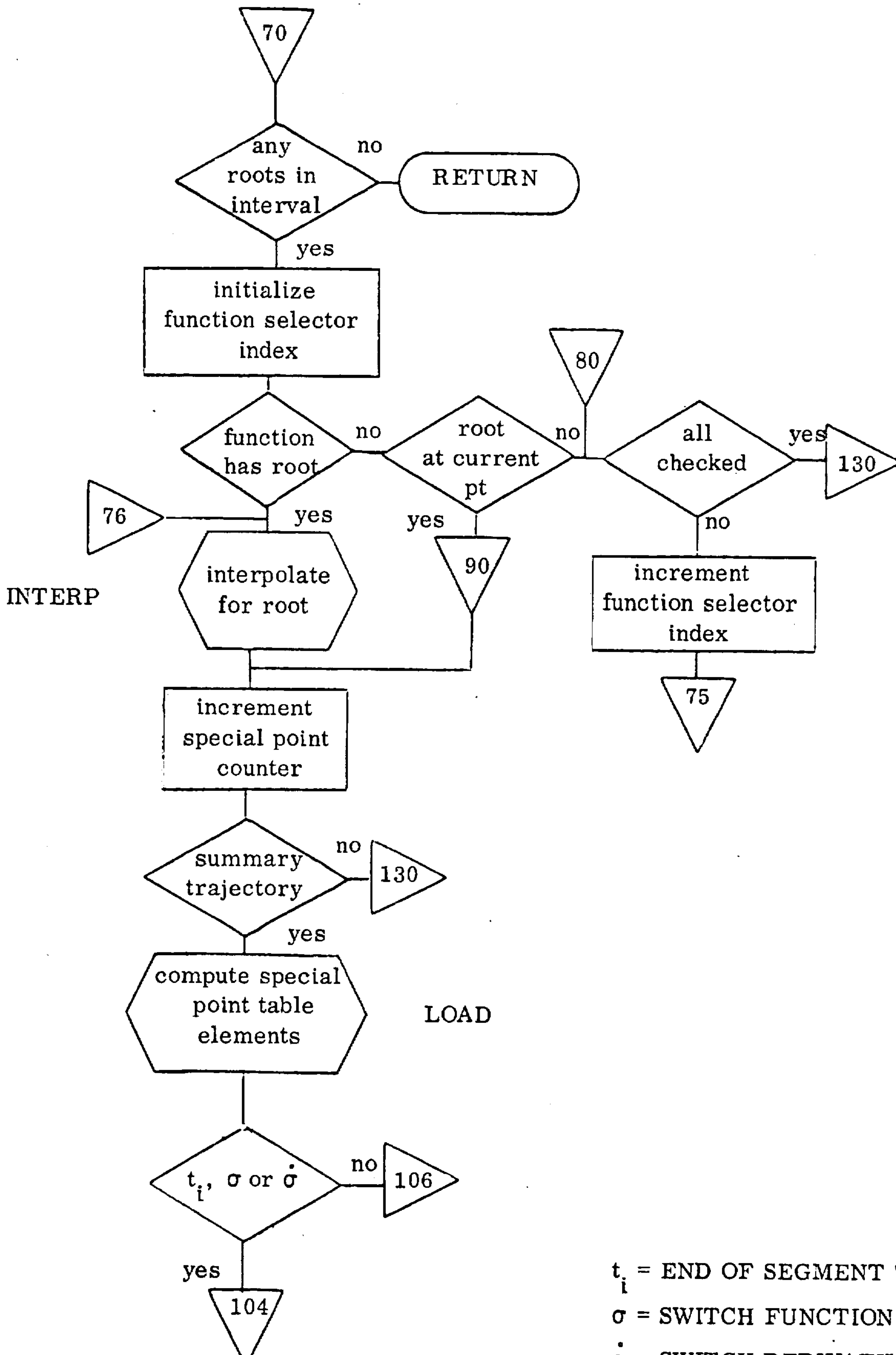
The following special flow chart is included to elucidate the basic aspects of CHECK's operation:



CHECK FOR
SPECIAL POINTS



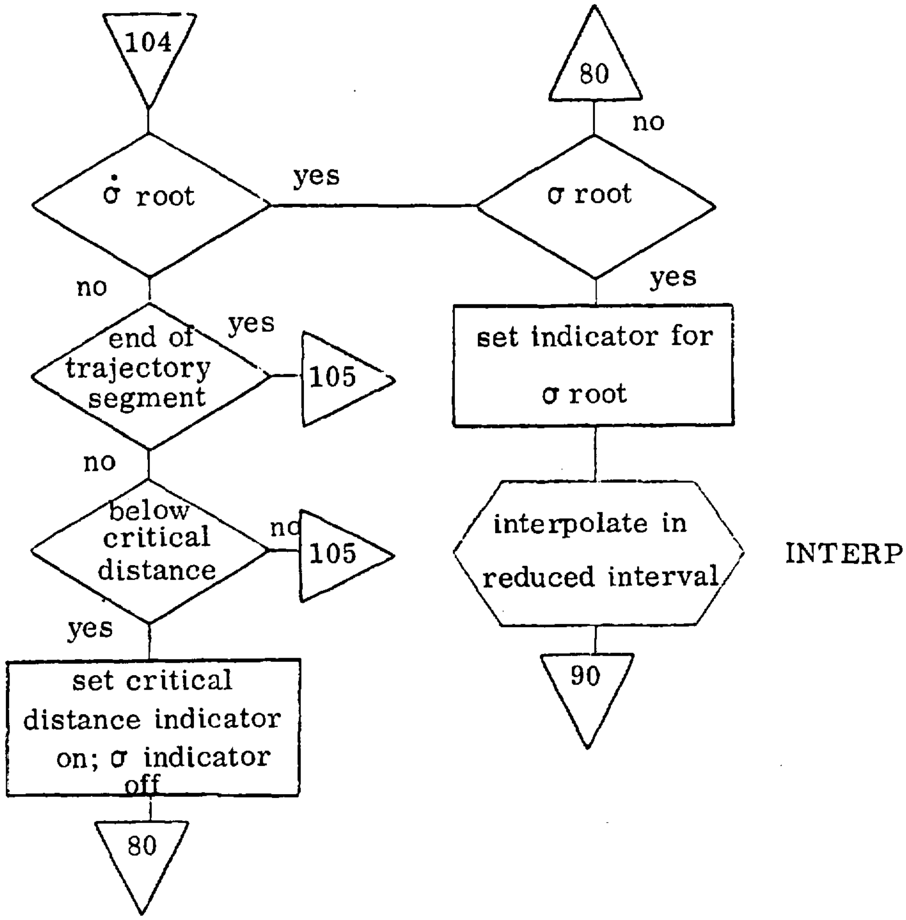
CHECK-3

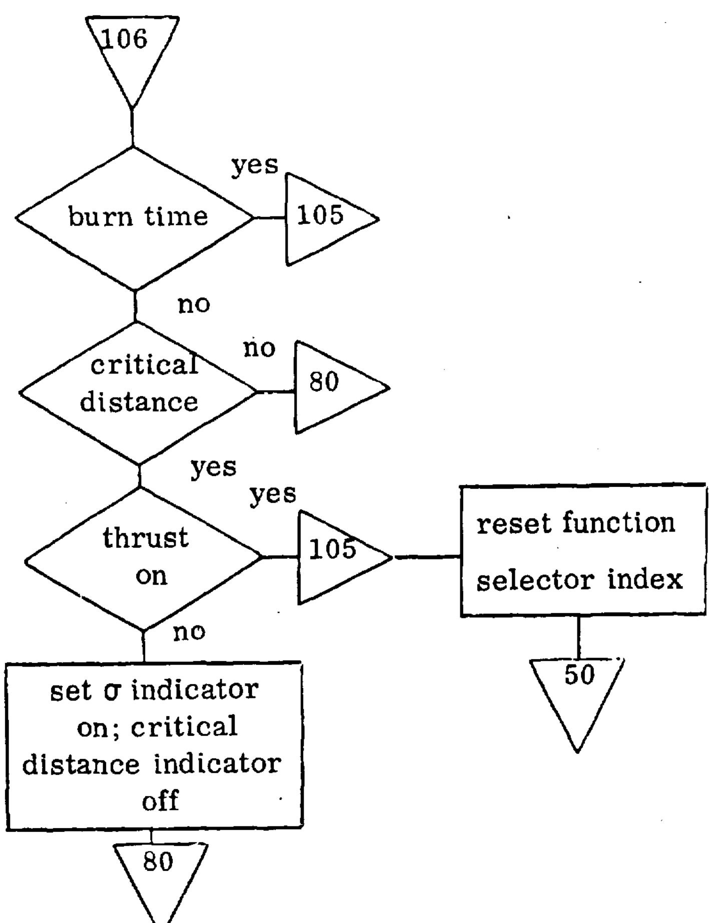


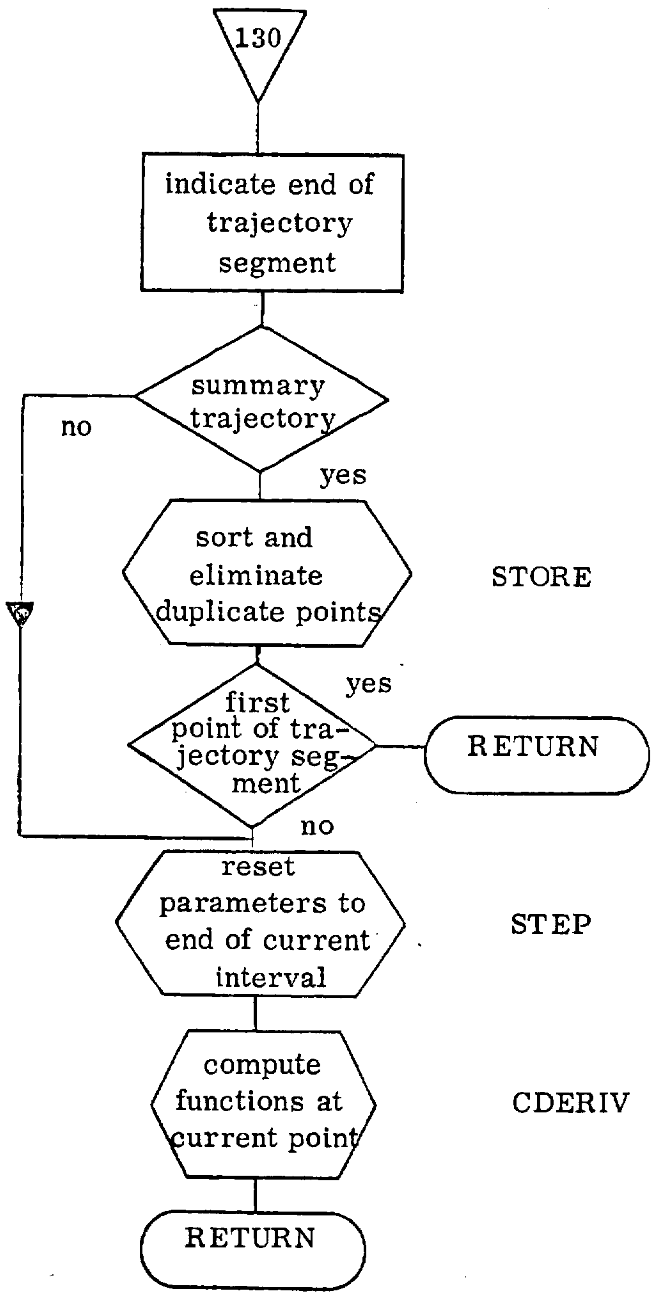
t_i = END OF SEGMENT TIME

σ = SWITCH FUNCTION

$\dot{\sigma}$ = SWITCH DERIVATIVE


104
$\dot{\sigma}$ root
yes
no
80
no
σ root
yes
set indicator for σ root
end of trajectory segment
yes
105
no
below critical distance
no
105
interpolate in reduced interval
INTERP
yes
90
set critical distance indicator on; σ indicator off
80


106
burn time
yes
105
no
critical distance
no
80
yes
thrust on
yes
105
reset function selector index
50
no
set σ indicator on; critical distance indicator off
80



130
indicate end of trajectory segment
summary trajectory
no
yes
sort and eliminate duplicate points
STORE
first point of trajectory segment
yes
RETURN
no
reset parameters to end of current interval
STEP
compute functions at current point
CDERIV
RETURN

<u>Messages and printouts:</u> An error condition, consisting of an array overflow, would exist if CHECK attempted to isolate more than JMAX = 20 points within any computation step. When the limit JMAX is exceeded, the following diagnostic message and array-dump is output on unit 6:

STORAGE TABLE EXCEEDED IN SUBROUTINE CHECK AT TIME = <u>(t)</u> DAYS.
MAXIMUM OF <u>(JMAX)</u> SPECIAL POINTS ARE ALLOWED WITHIN A COMPUTING STEP.
DUMP OF STORAGE ARRAY CEPS (*,*,2) FOLLOWS.

_____ _____ _____ _____ _____ _____ _____ _____ . . . .

_____ _____ _____ _____ _____ _____ _____ _____

_____ _____ _____ _____ _____ _____ _____ _____

_____ _____ _____ _____ _____ _____ _____ _____

.
.
.

(line 20) _____ _____ _____ _____ _____ _____ _____

in which t is the time elapsed since the beginning of the trajectory, in days, and CEPS is the storage array of remarkable points and related trajectory-function values. Concurrently, the message is output on unit 12:

FATAL ERROR. STORAGE TABLE *CEPS* EXCEEDED.

Subsequently, the program master error indicator is set, and the routine is exited forthwith.

<u>CHECK EXTERNAL VARIABLES TABLE</u>

Variable	Use	Common	Description
B(2,30)	SU	EXTREM	Array of monitored functions, described in subroutine CDERIV.
X(50)	U	REAL8	Array of trajectory dependent-variables described in subroutine RKSTEP.

CHECK EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
JC	SU	INTGR4	Counter corresponding to the JC^{th} specified time-function value (i.e., the time) isolated, or to be isolated, thus far on the current trajectory segment. Attains values greater than one when imposed coast phases are invoked.
JJ	S	INTGR4	Thrust switch point or critical solar distance indicator.
KF	S	INTGR4	End of trajectory-segment indicator.
PP(2)	U	REAL8	Primer vector magnitude, λ, at start of computation step (PP(1)) and instantaneously (PP(2)).
GAP	U	REAL8	Propulsion-corner proximity tolerance-interval, $\Delta\sigma$.
BETA	U	REAL8	Trajectory independent-variable, β, in $AU^{1/2}$ radians.
CEPS (14, 20, 2)	U	EXTREM	Storage array of remarkable points and related trajectory-function values; temporary values corresponding to the current computation step.
CHIX (2, 100)	SU	EXTREM	Storage array for solar-panel array angle, χ, in degrees.
DMIN	U	REAL8	Minimum value of density (in the expression for the power function γ), d_{min}, in AU^{-2}.
EDGE	SU	LOGIC4	Indicator for solar arrays being oriented edgewise to the sun; used only if power degradation is simulated.
FLAP	U	LOGIC4	Indicator for power-curve options corresponding to nuclear-electric propulsion or solar-electric propulsion using reflecting flaps to maintain maximum power.

CHECK EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
HEAT	U	LOGIC4	Indicator that the solar panels are maintained normal to the sun line at all times.
MODE	U	INTGR4	Power variation option selector.
NCEP	SU	INTGR4	Maximum number of monitored functions which are stored for printing.
NCHK	SU	INTGR4	Maximum value of I (see discussion).
PLUS	SU	LOGIC4	Indicator for determining appropriate region in two-dimensional simulations, as described in subroutine THANGD.
PMAX	S	REAL8	Maximum value of power ratio, $(q\gamma)_{max}$, encountered along the trajectory.
QJEX	U	LOGIC4	Detailed printout indicator.
RMAX	S	REAL8	Maximum solar distance, r_{max}, encountered by the spacecraft along the trajectory, in AU.
RMIN	S	REAL8	Minimum solar distance, r_{min}, encountered by the spacecraft along the trajectory, in AU.
SWIT	U	REAL8	$\sigma_1 + \sigma_2$ (See subroutine FUNCT discussion).
TILT	U	LOGIC4	Indicator that solar arrays are to be tilted during solar proximity (to maintain constant power when there is no power-degradation).
XCOM(6)	U	REAL8	Communication-distance vector and its time derivative, R_{com} and $\dot{R}_{com}$, in AU and EMOS, respectively.

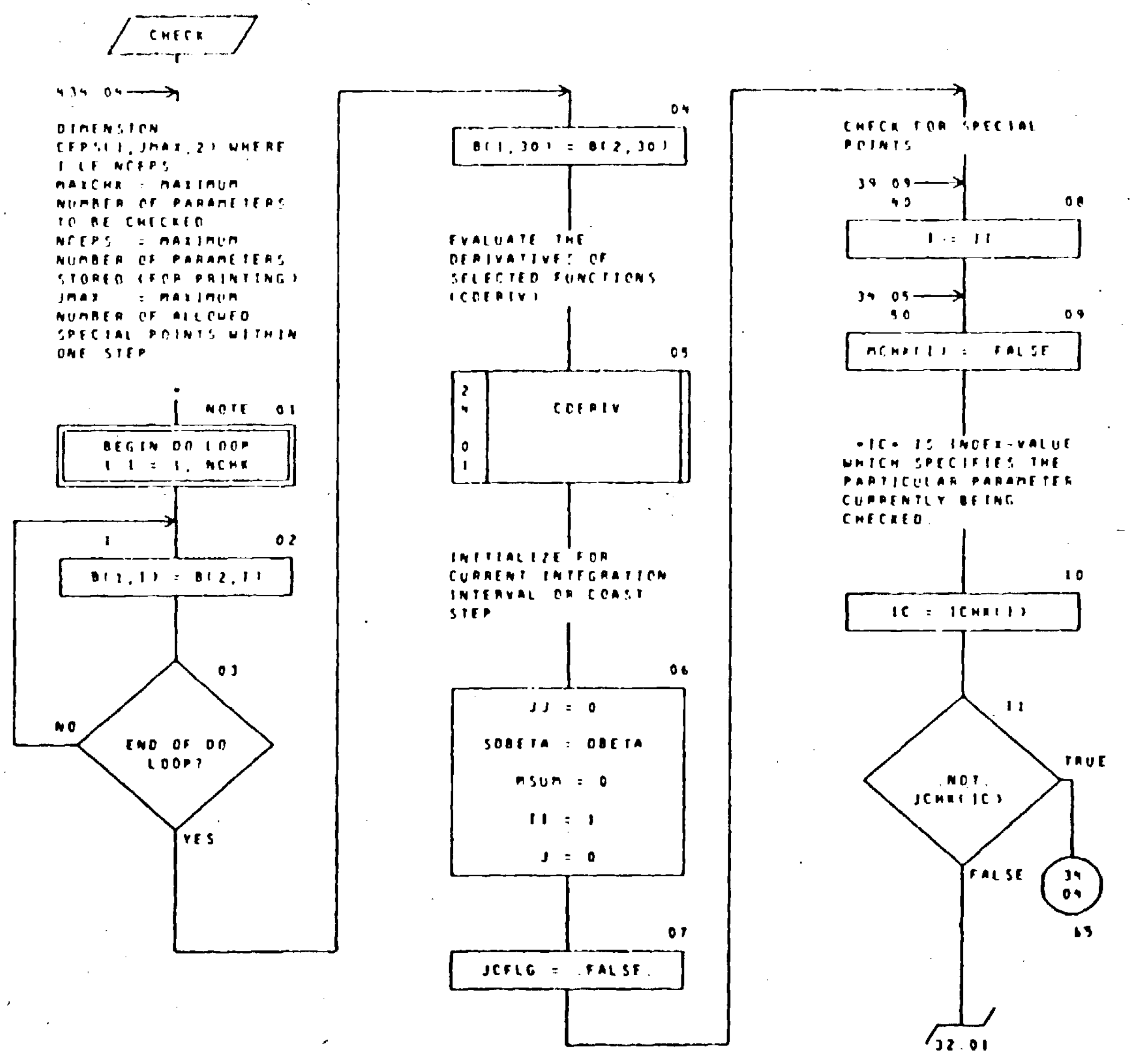
CHECK EXTERNAL VARIABLES TABLE (cont)

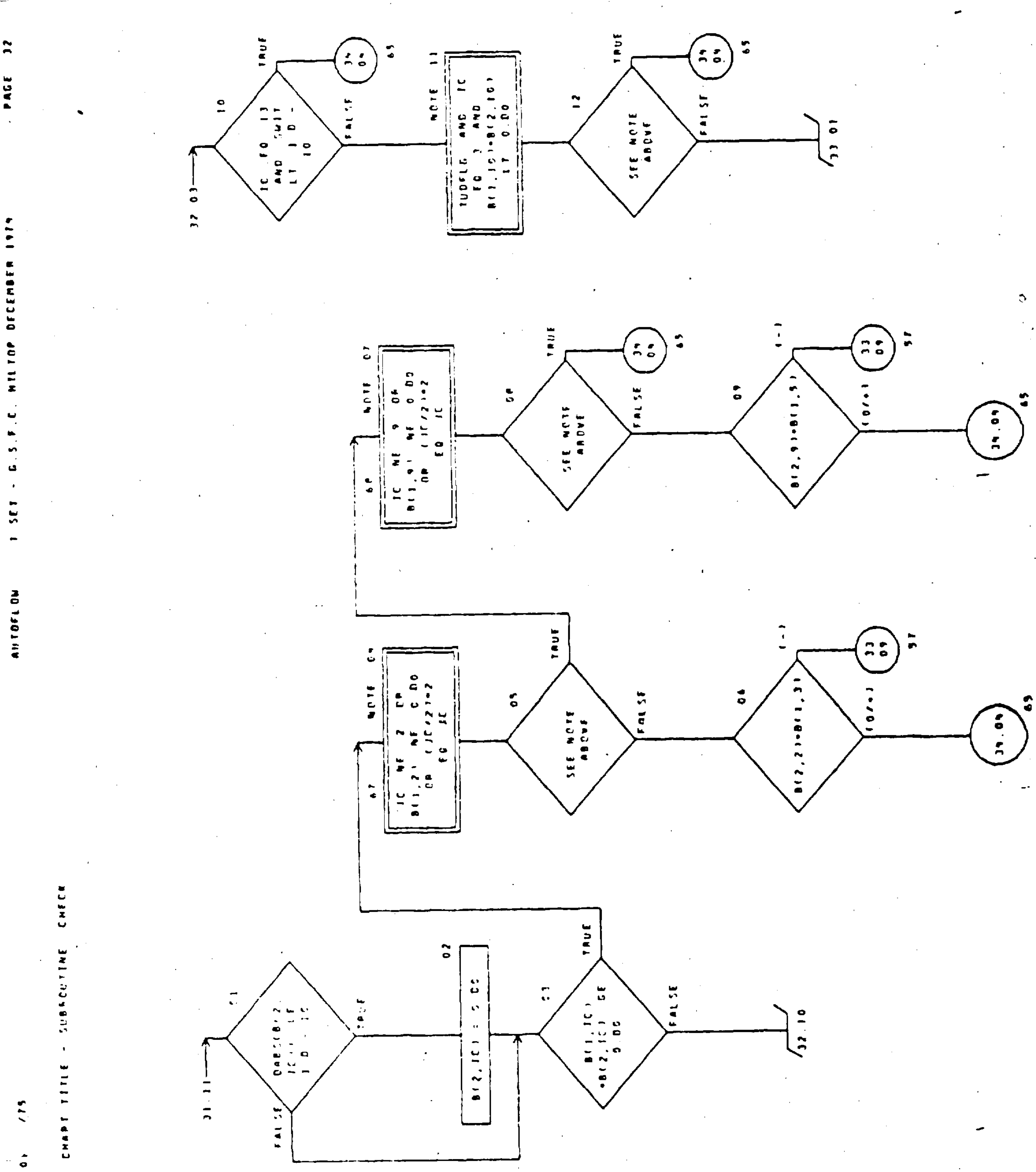
Variable	Use	Common	Description
CHFNC	U	REAL8	A function whose roots determine the switch points of the solar array tilt angle to and from the power-curve boundary, f_{ch1}.
COAST	U	LOGIC4	Indicator for coasting flight or thrusting flight.
CONTM	U	REAL8	Time conversion factor, tau to days.
DBETA	SUA	REAL8	Computation step size, $\Delta\beta$ (increment of the trajectory independent variable).
ERODE	U	LOGIC4	Power degradation option indicator.
ERROR	S	LOGIC4	Program master error indicator.
IHUNG	SU	INTGR4	Counter of the number of propulsion-corner-proximity occurrences along the current trajectory.
JCMAX	U	INTGR4	Maximum value which JC may attain, corresponding to the end of the current trajectory segment.
NSPEC	U	INTGR4	Master array index (and counter) for extremum-table storage arrays.
PCURV	SU	LOGIC4	Indicator for condition in which solar arrays are oriented to receive the maximum power permissible under the current power-curve assumption, or to be tilted away from the maximum permissible due to degradation considerations.
THUNG(5)	SU	REAL8	Times-of-occurrence of consecutive propulsion-corner-proximities, in tau.
BALLIS	U	LOGIC4	Indicator that trajectory segment is all-ballistic (pure coast).

CHECK EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
FIXTAU	U	LOGIC4	Indicator for non-zero λ_τ.
FIXTHR	U	LOGIC4	Indicator for fixed thrust-angle.
HOHUNG (5)	S	REAL8	Thrust switch function time-derivative values, corresponding to THUNG(i).
MAXPOW	U	LOGIC4	Indicator for mode of operation in which solar panels are maintained in orientation to receive maximum permissible power, when power degradation option in invoked.
REGION	SU	LOGIC4	Indicator for spacecraft solar proximity; demarks two possible regions in space, separated by sphere about sun of specified radius, at which power function (or its derivative) has a corner.
SKOUNT	S	REAL8	Number of iterations in subroutine INTERP required to isolate the (current) remarkable point.
SWHUNG (5)	S	REAL8	Thrust switch-function values, corresponding to THUNG(i).
SWITCH (2)	U	REAL8	Thrust switch function, σ, at current time (SWITCH(2)) and at start of current computation step (SWITCH(1)).
TCHECK (41)	U	REAL8	Array of time values, each of which is isolated by subroutine CHECK, in tau; intermediate values correspond to engine switch times associated with imposed coast phases.
TUDFLG	U	LOGIC4	Indicator for two-dimensional trajectory simulation (motion in the xy plane).
WONDER	U	LOGIC4	Indicator which allows or disallows the program to consider if the current trajectory is in the proximity of a propulsion-time corner.

CHART TITLE - SUBROUTINE CHECK





AUTOFLOW
CHART TITLE - SUBROUTINE CHECK
PAGE 32
SEE NOTE ABOVE
TRUE
FALSE

CHECK-13

CHART TITLE - SUBROUTINE CHECK

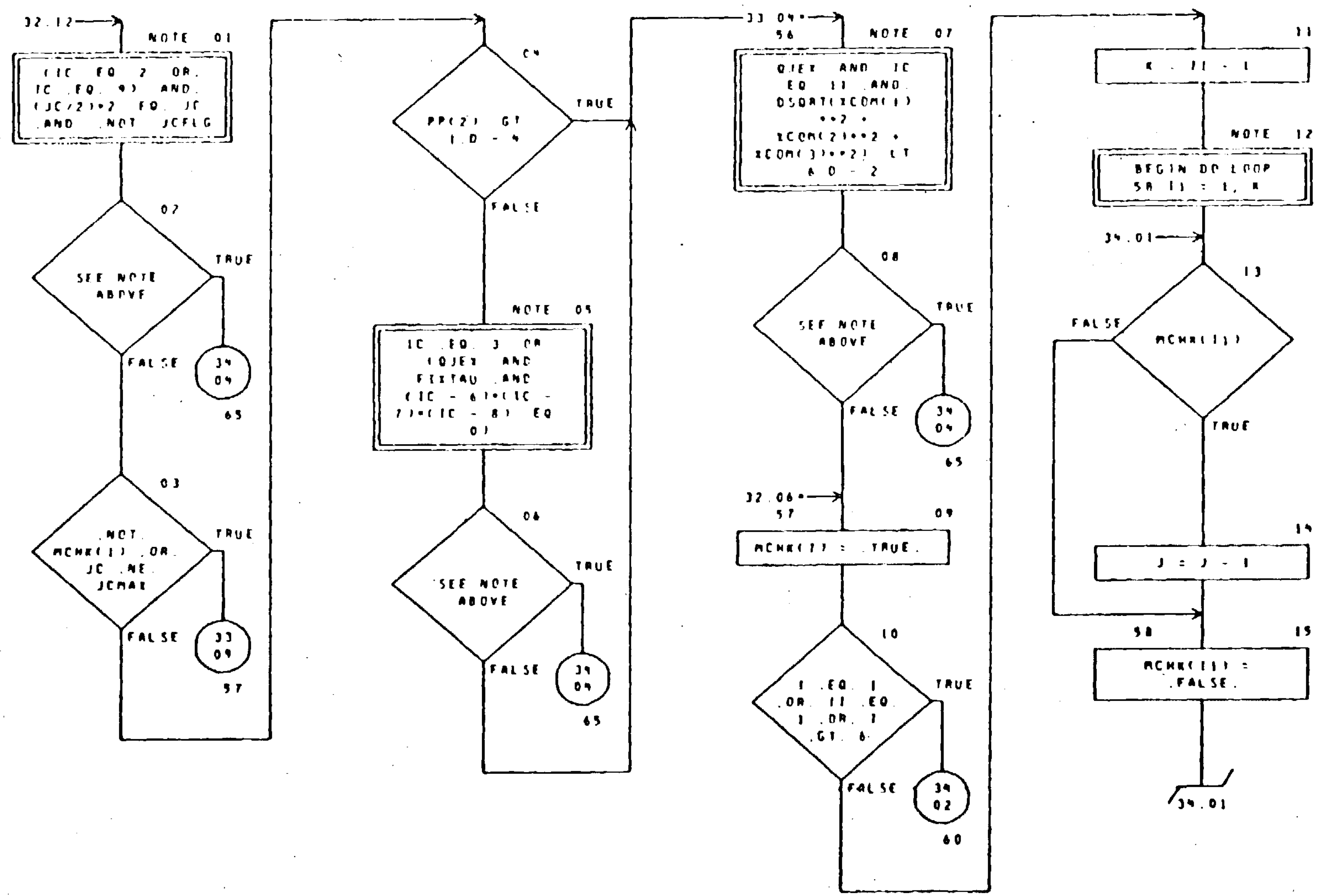
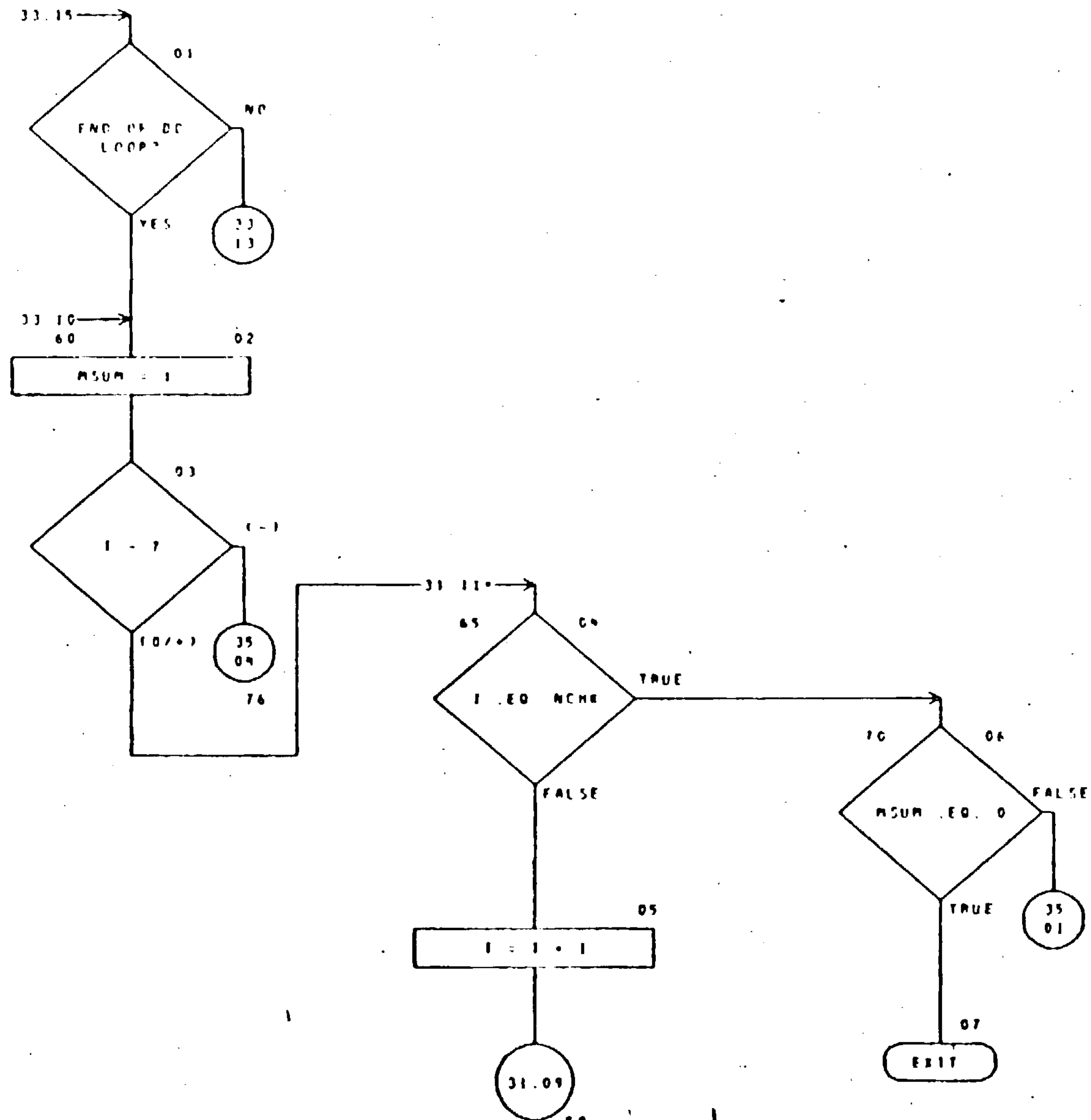


CHART TITLE - SUBROUTINE CHECK



CHECK-15

CHART TITLE - SUBROUTINE CHECK

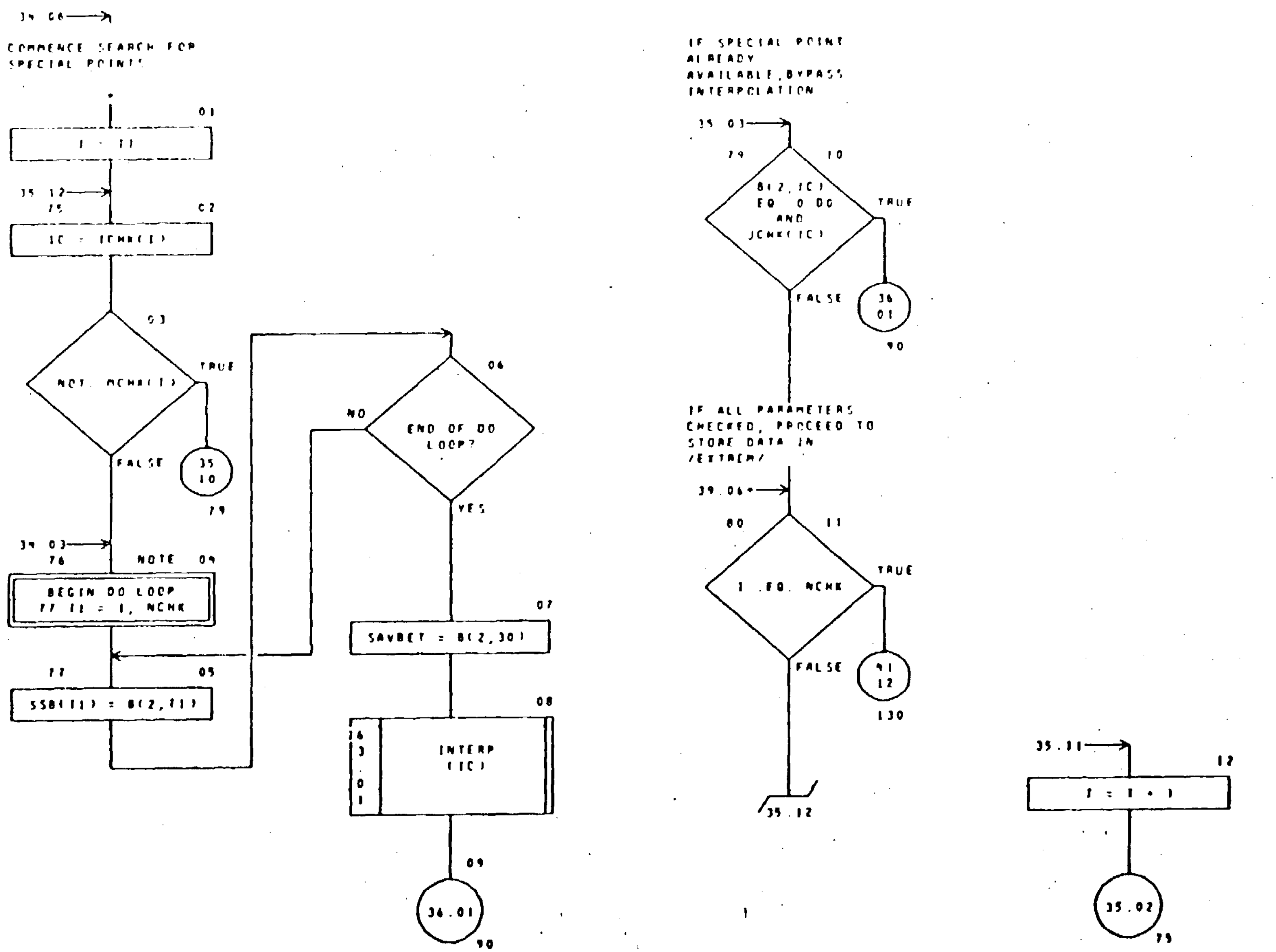
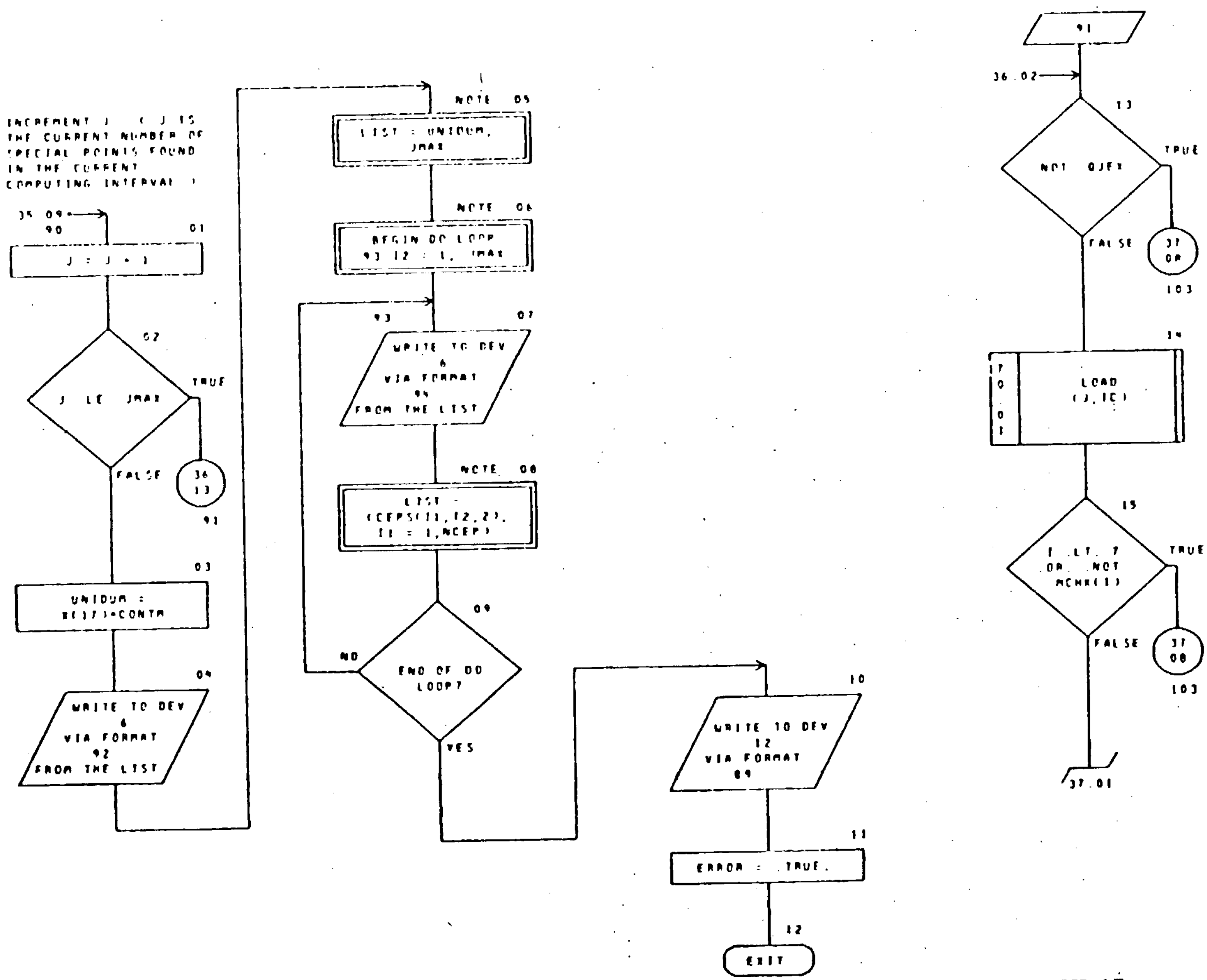


CHART TITLE - SUBROUTINE CHECK



CHECK-17

CHART TITLE - SUBROUTINE CHECK

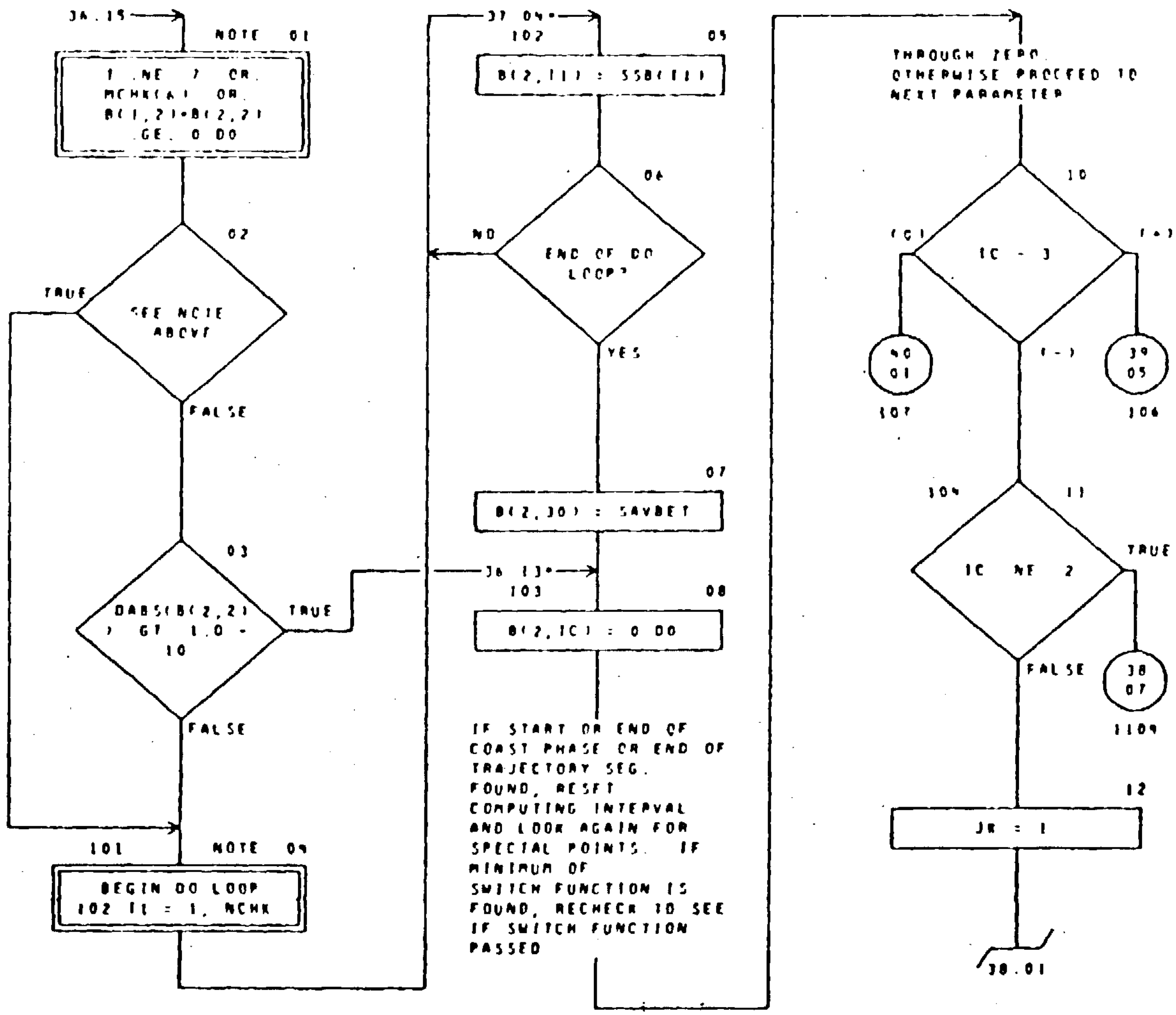
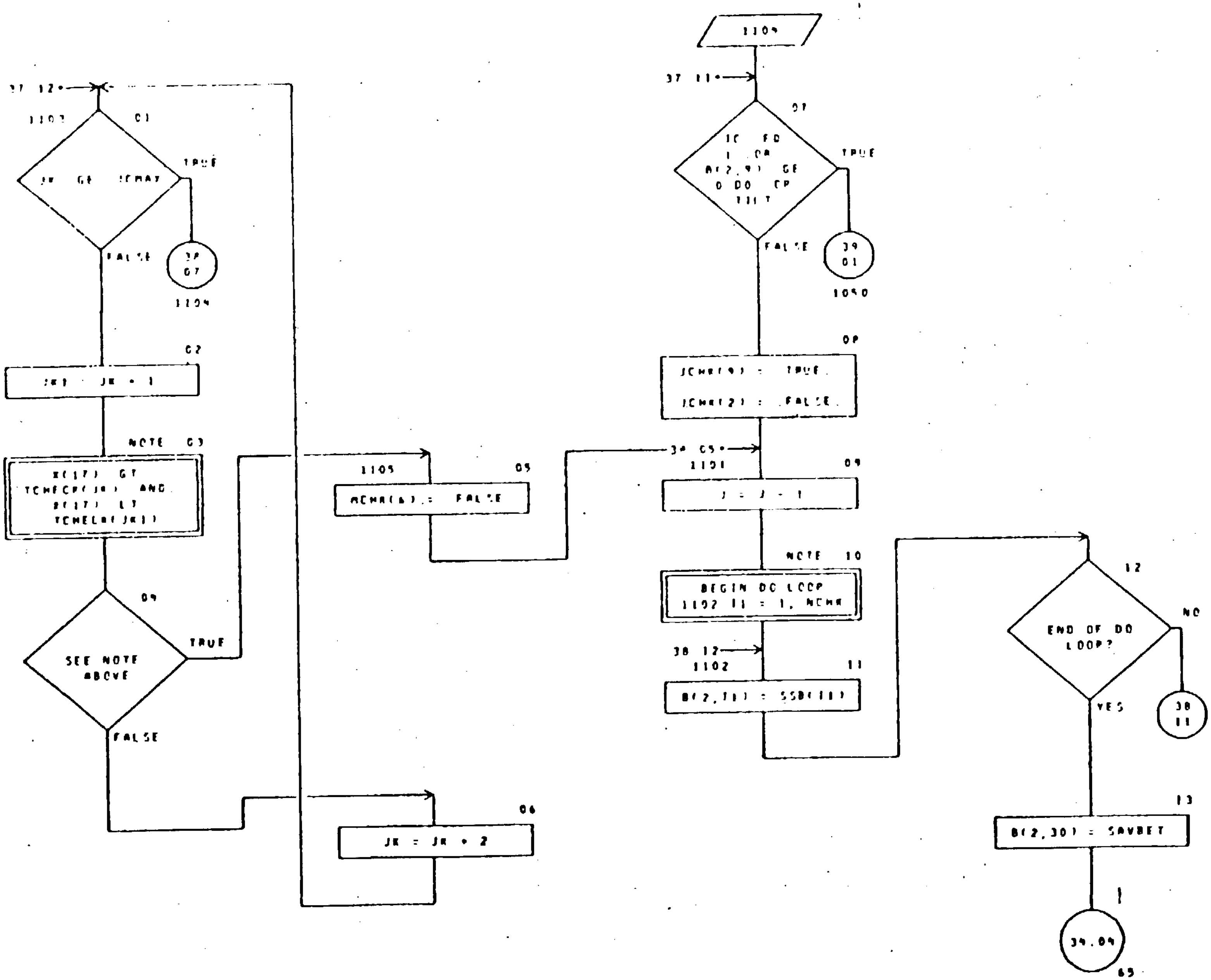


CHART TITLE - SUBROUTINE CHECK



CHECK-19

CHART TITLE - SUBROUTINE CHECK

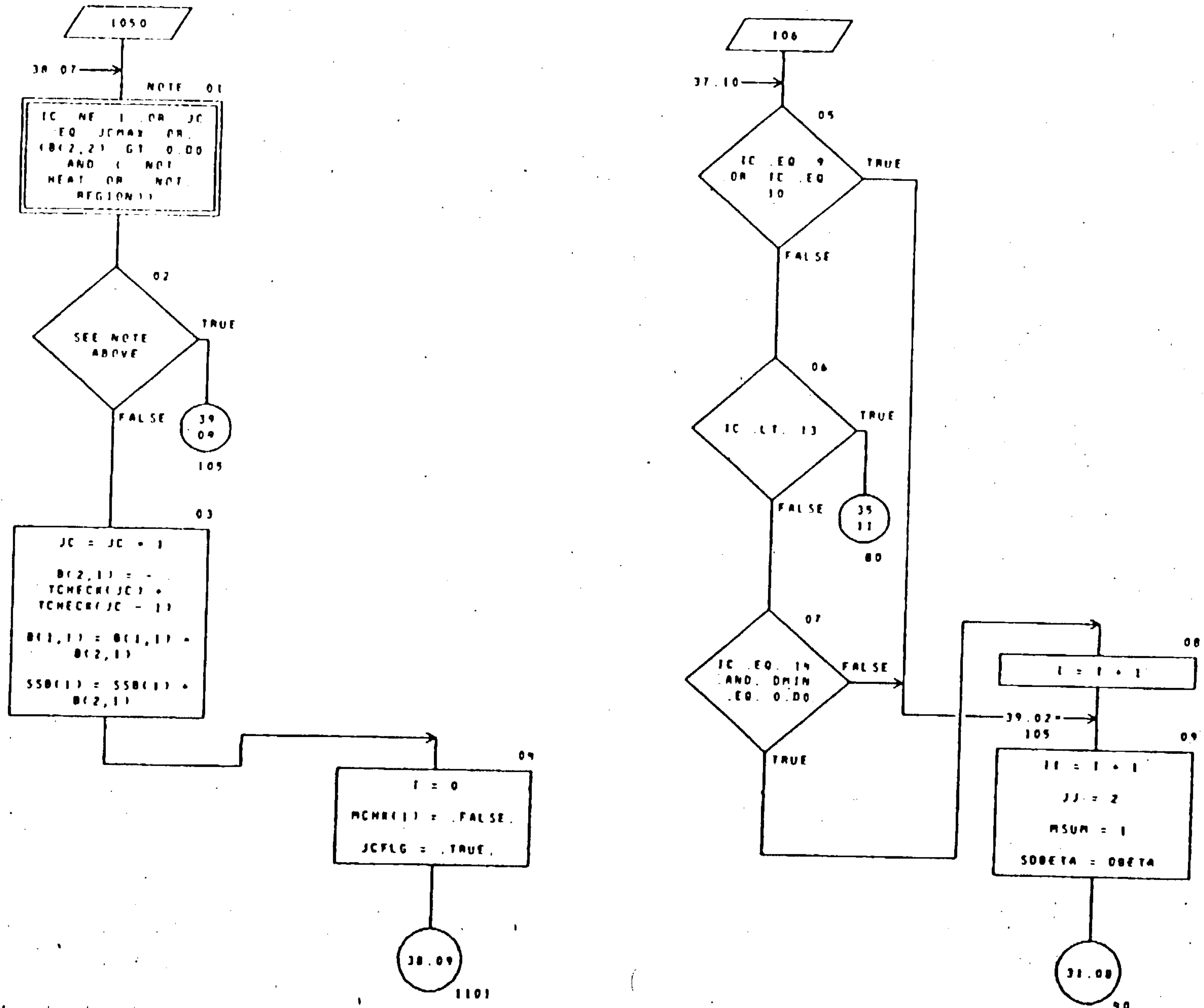


CHART TITLE - SUBROUTINE CHECK

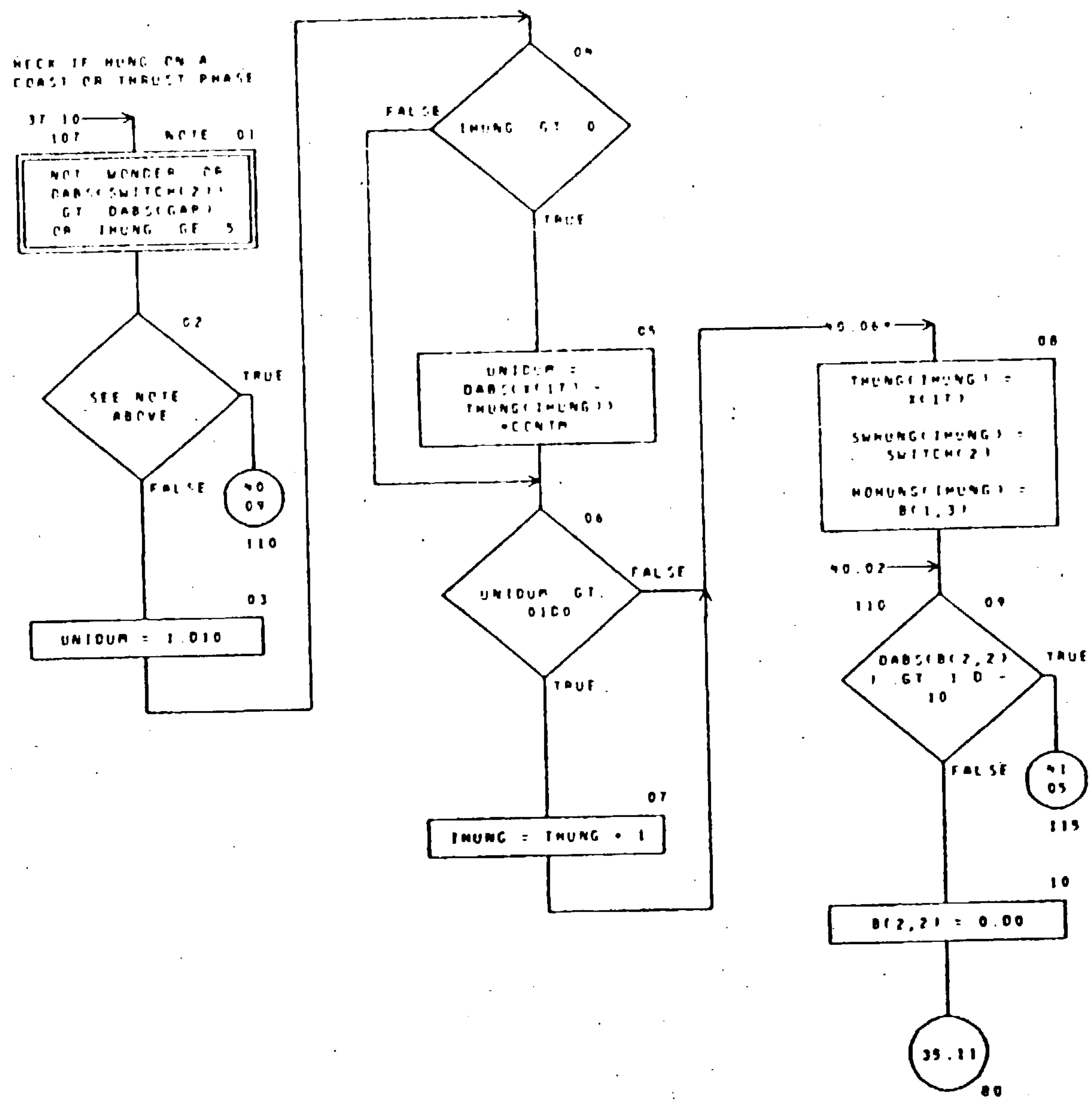


CHART TITLE - SUBROUTINE CHECK

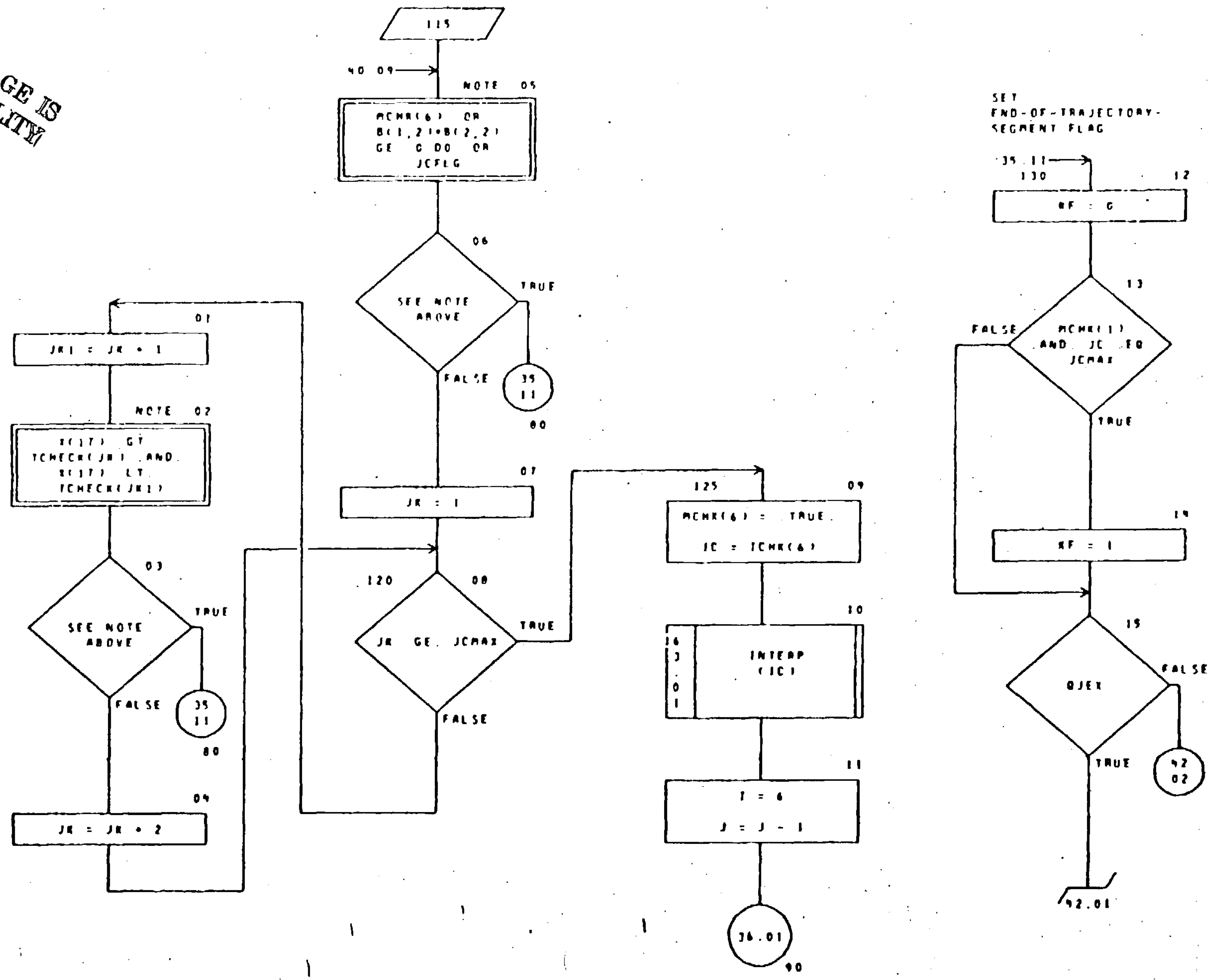


CHART TITLE - SUBROUTINE CHECK

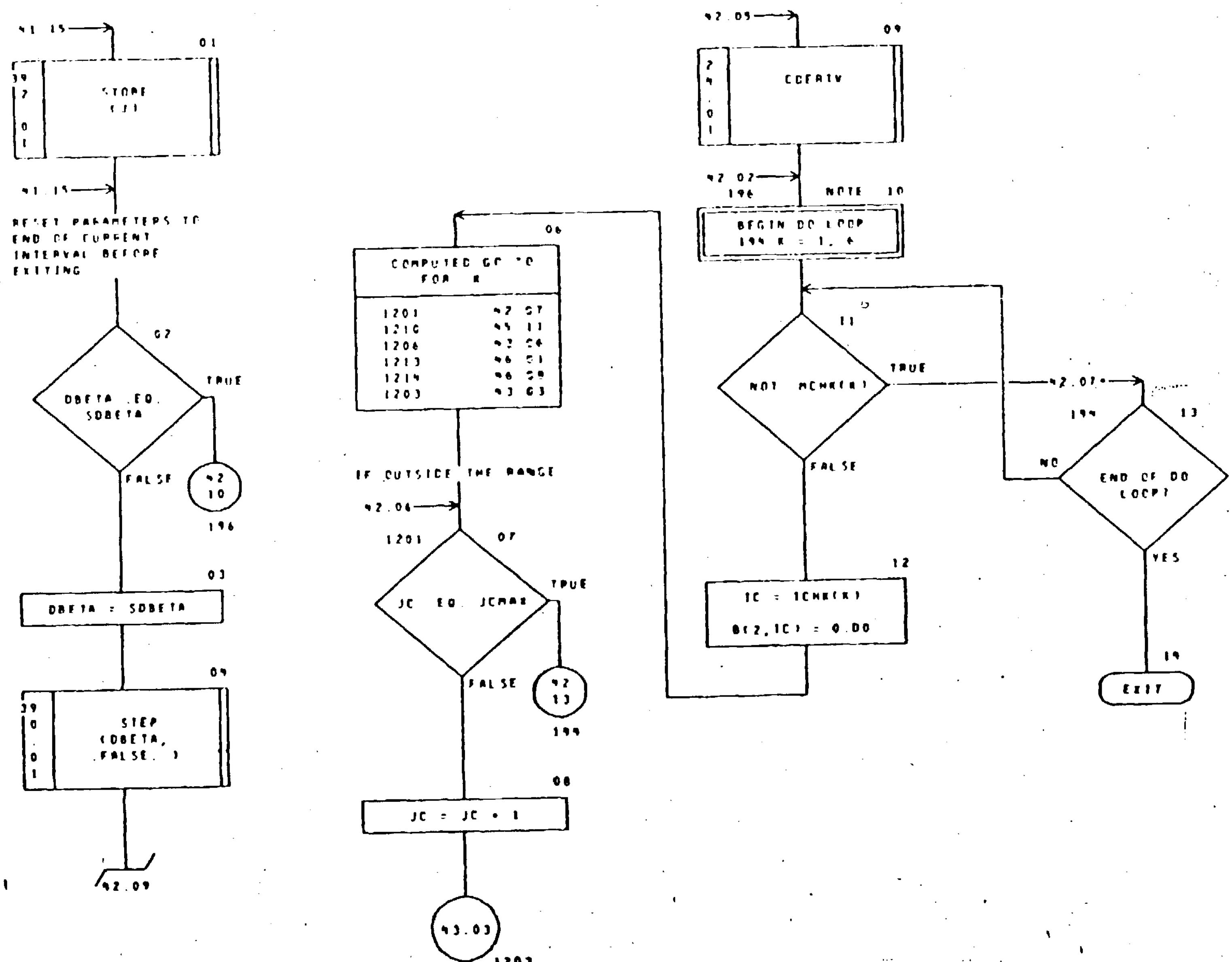


CHART TITLE - SUBROUTINE CHECK

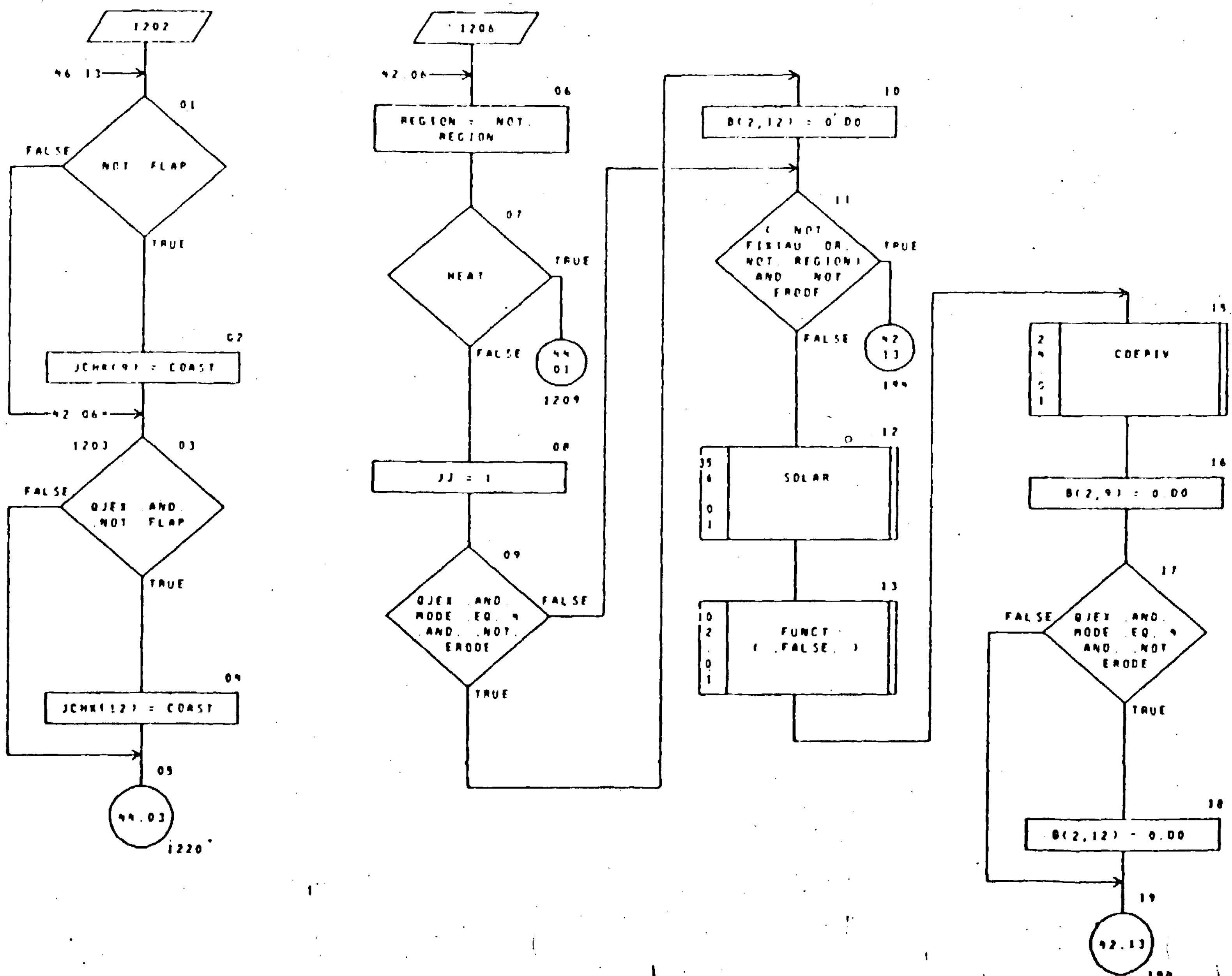


CHART TITLE - SUBROUTINE CHECK

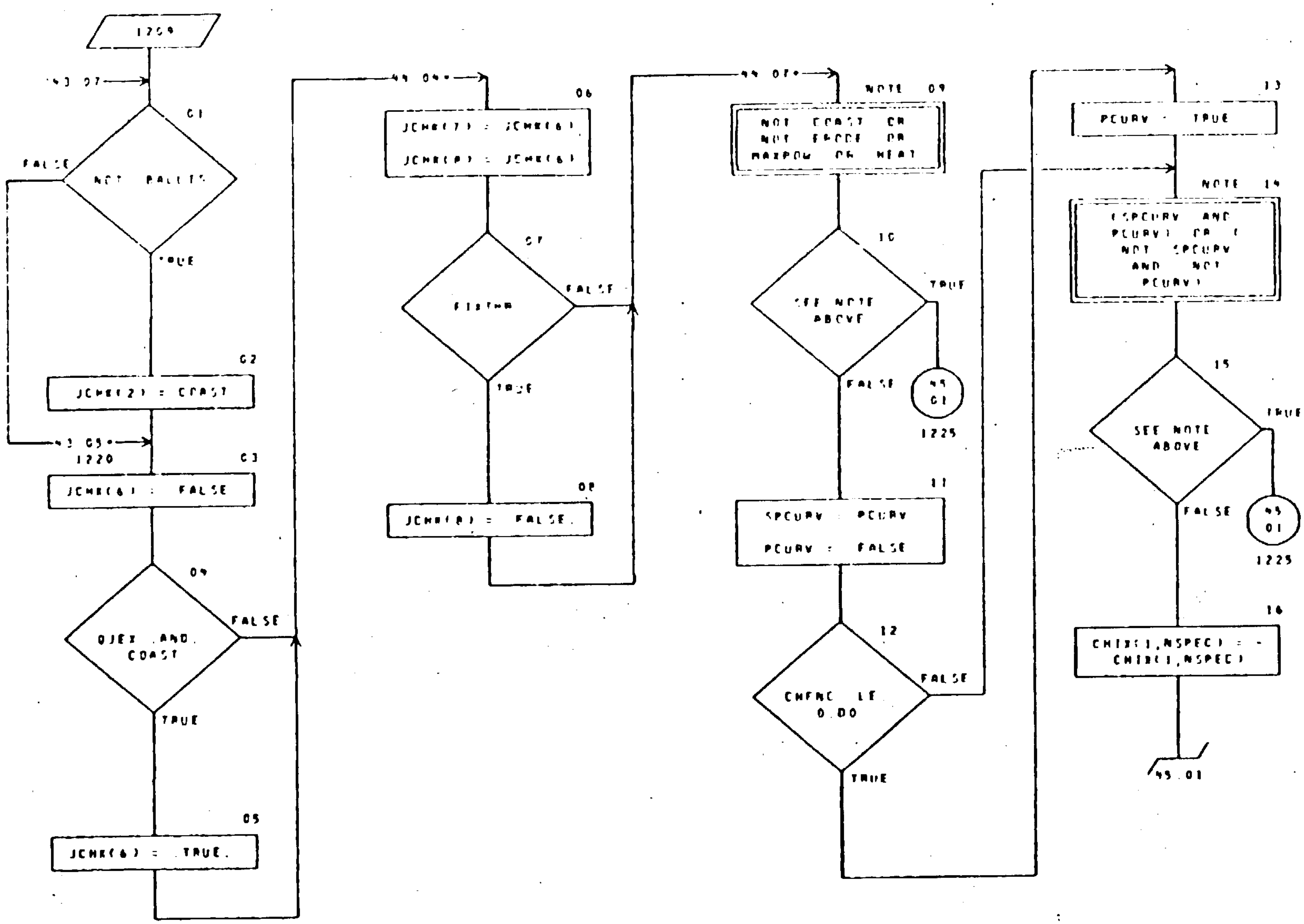


CHART TITLE - SUBROUTINE CHECK

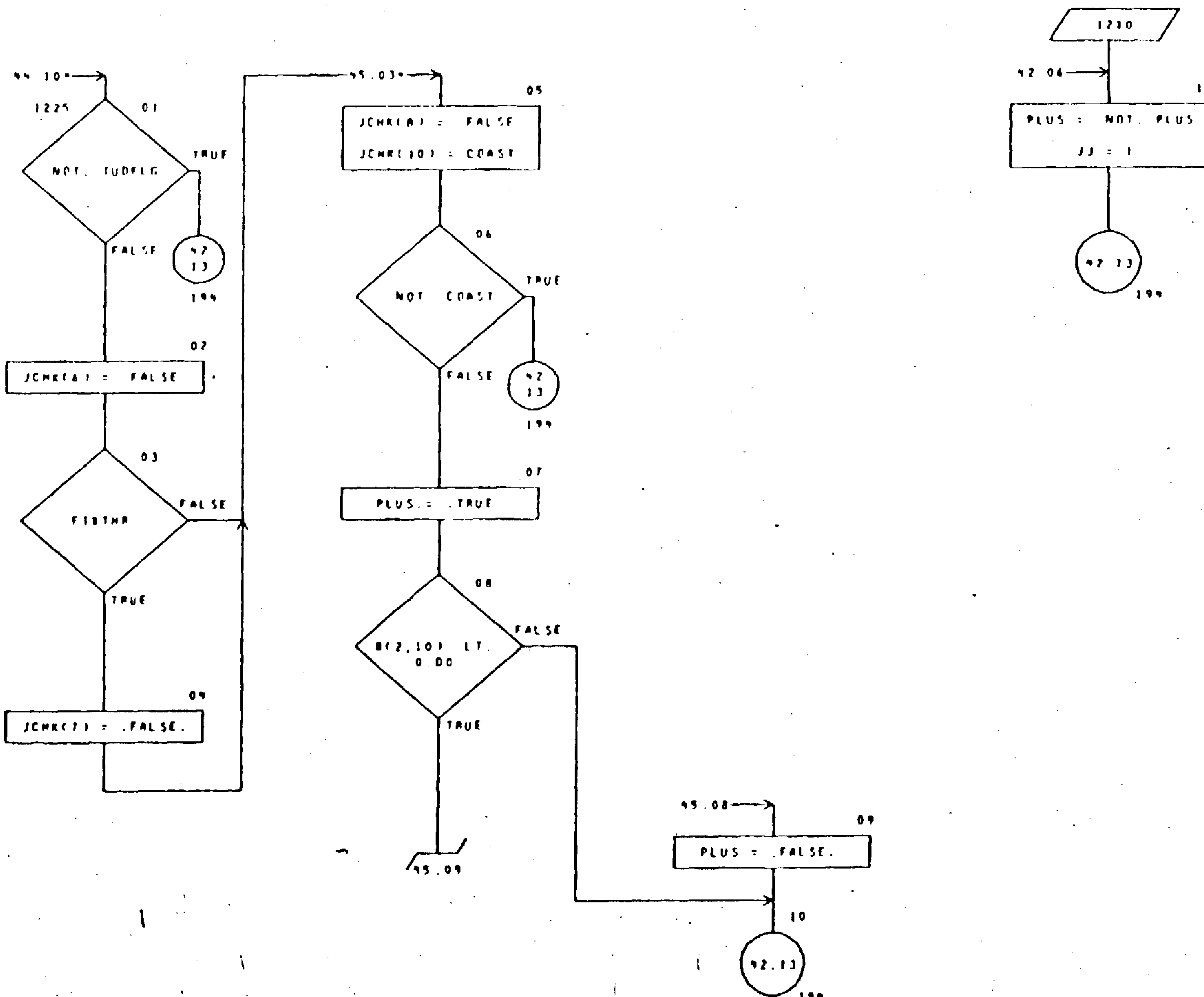


CHART TITLE - SUBROUTINE CHECK

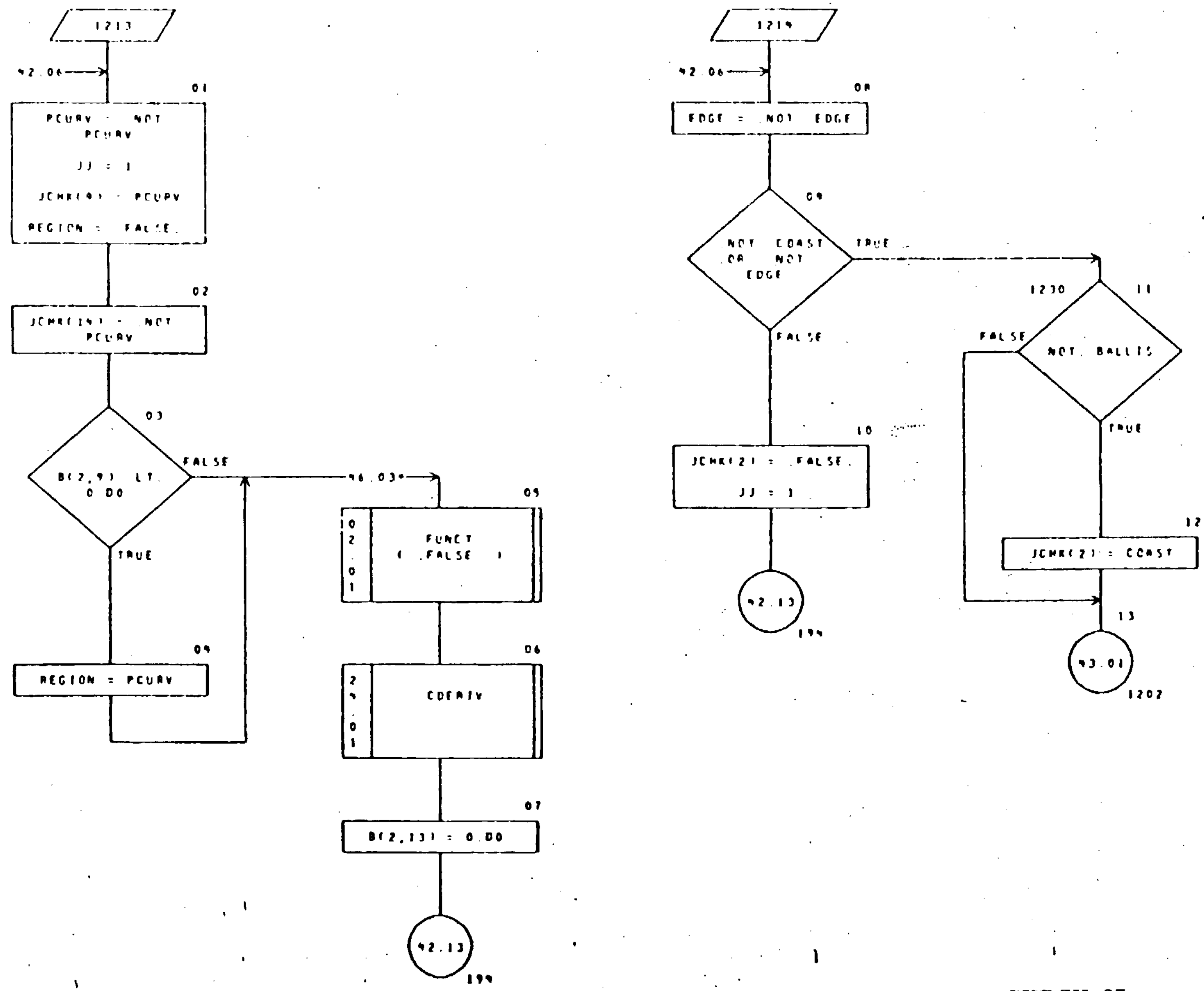


CHART TITLE - SUBROUTINE CHECK

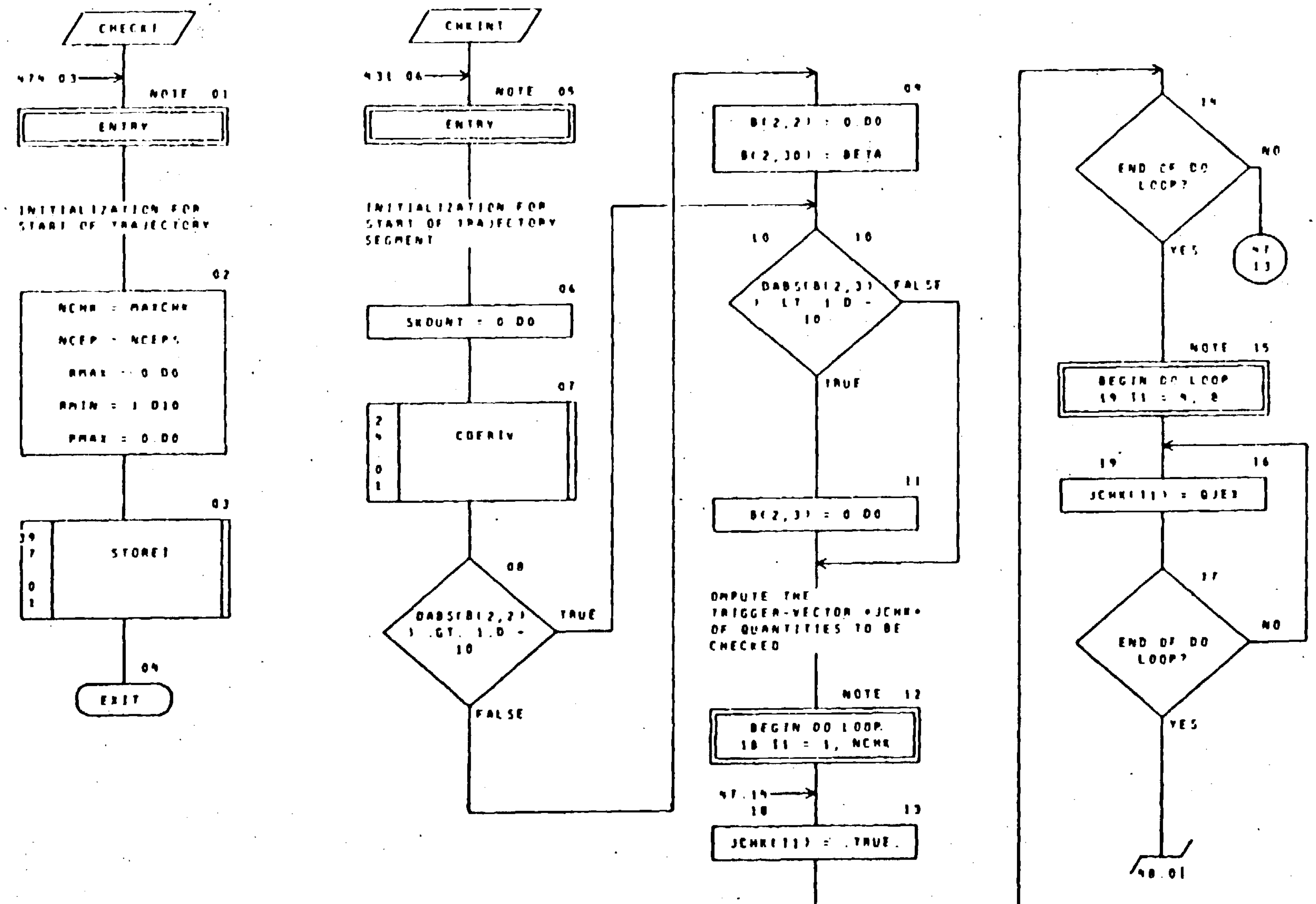
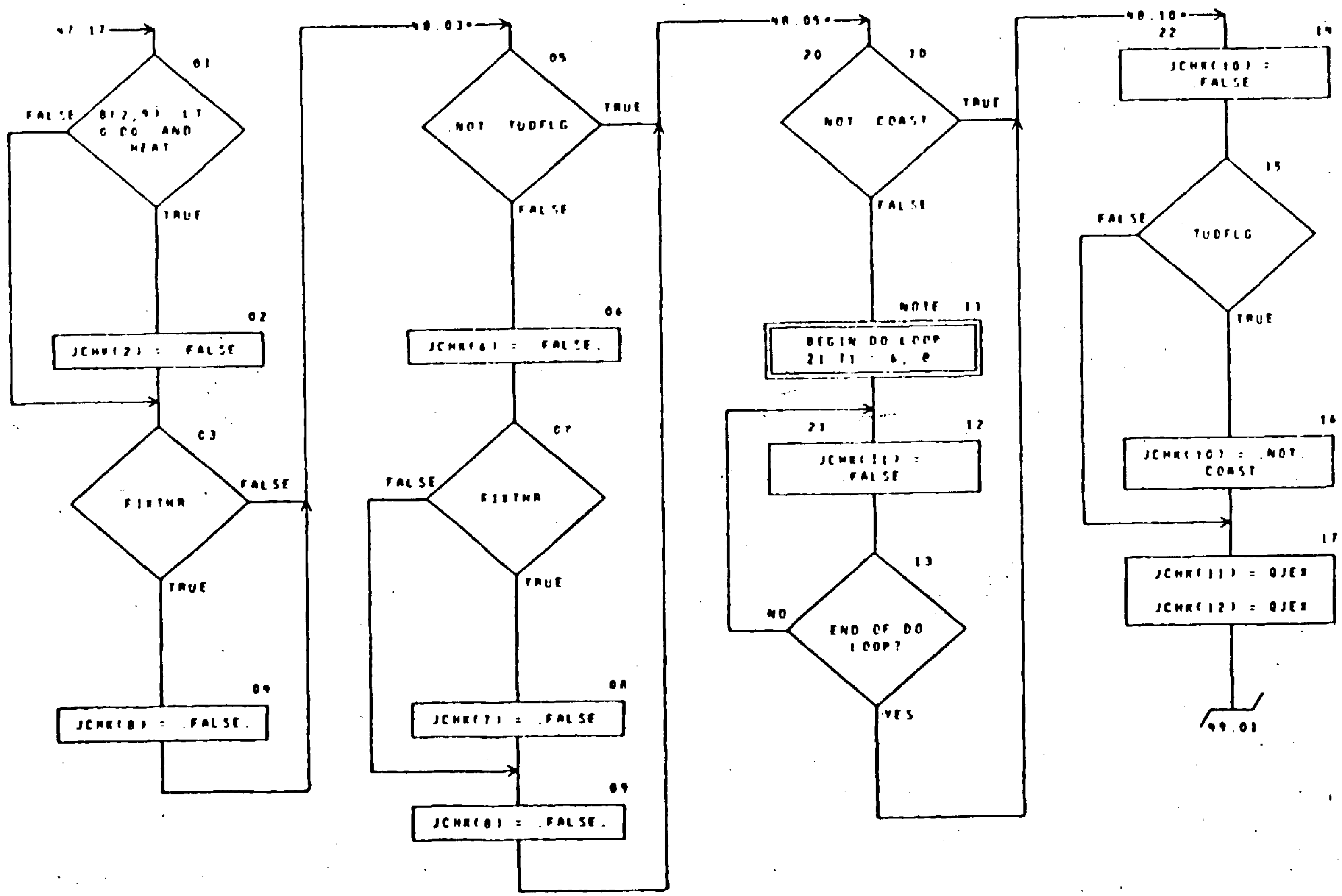


CHART TITLE - SUBROUTINE CHECK



CHECK-29

CHART TITLE - SUBROUTINE CHECK

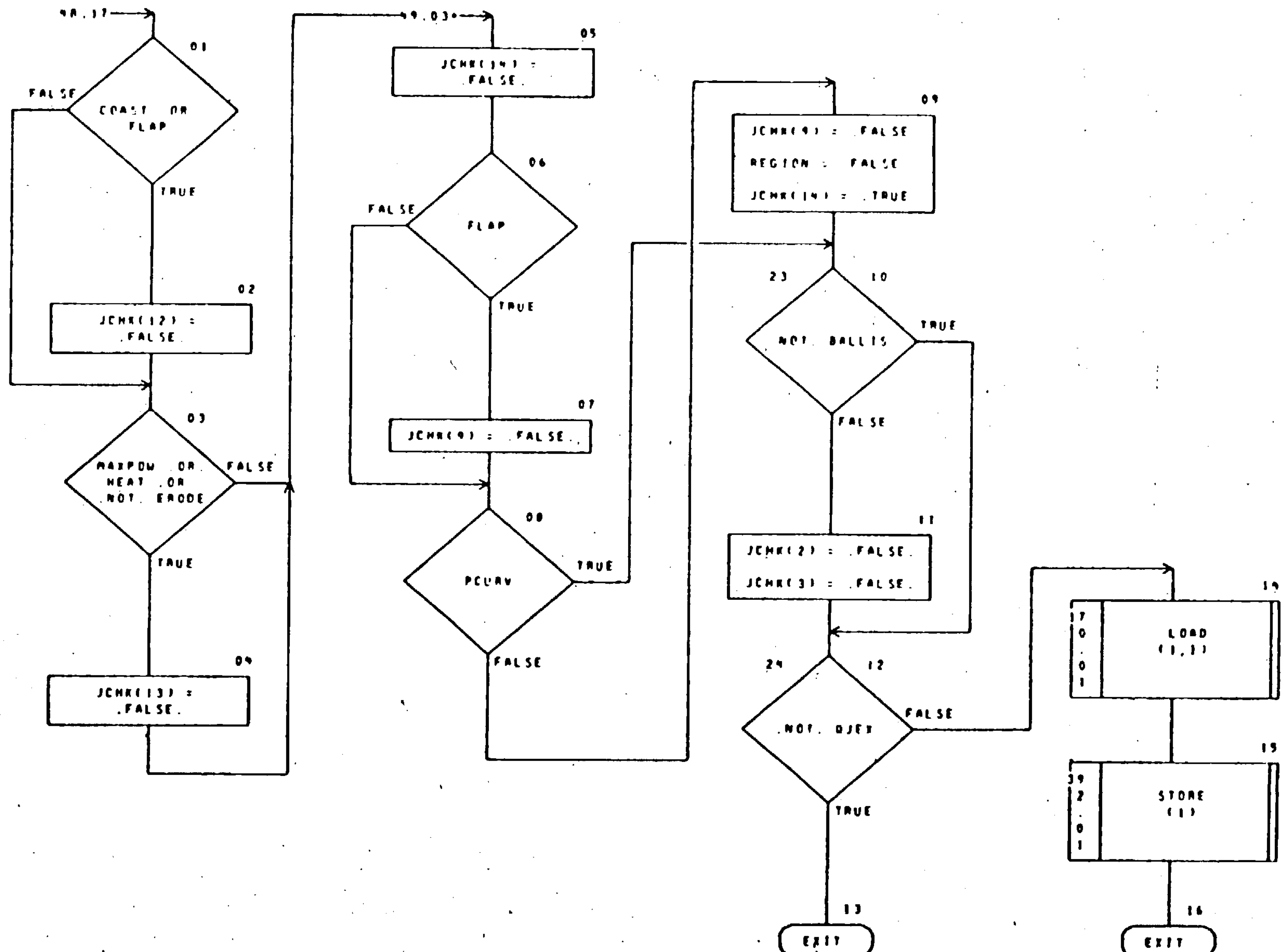


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL  TUDFLG,WONDER,MCHK,FIXTHR,SPCURV,REGION,COAST,OJEX,JCHK
     .,ERROR,ERODE,FIXTAU,TILT,HEAT,FLAP,MAXPOW,PCURV,JCFLG,EDGE,PLUS
     .,BALLIS
      DIMENSION          MCHK(30),ICHK(30),JCHK(30),SSB(30)
      COMMON /REAL8/ RO1(45),DBETA,RO2(16),GAP,THUNG(5),SWHUNG(5),
     WOHUNG(5),XCOM(6),RO3(53)          ,RMAX,RMIN,PMAX,RO4(22),
                                        PP(2),SWITCH(2),RO5(109),
     CONTM,RO7(299),TCHECK(41),R10(234),SWIT,R11(293),DMIN,R13(4)
     .,CHENC,R12(33),SRCUNT,RO6(76),X(50),RO8(51),    BETA,RO9(448)
      COMMON /INTGR4/ IO4,MODE,IO1(35),NSPEC,IO7(12),IHUNG,IO2(12),
     JJ,KF,IO5(262),
     NCHK,NCEP,IO3(2),JC,JCMAX,IO6(667)
      COMMON /LOGIC4/ ERROR,LO1(3),WONDER,LO2(3),TUDFLG,LO3,FIXTHR,
     LO4(7),REGION,ERODE,FLAP,LO5,TILT,LO8(3),OJEX,FIXTAU,HEAT,LO6(2),
     COAST     ,LO7(4),PCURV,MAXPOW,EDGE,PLUS,LO9(3),BALLIS,L10(456)
      COMMON /EXTREM/ EO1(1800),CHIX(2,100),EO2( 100),CEPS(14,20,2),
     B(2,30)
      DATA ICHK /1,10,9,13,14,2,3,5,4,6,7,8,11,12,16*15/
      DATA MAXCHK,NCEPS,JMAX /14,12,20/
   92 FORMAT(1H1,52HSTORAGE TABLE EXCEEDED IN SUBROUTINE CHECK AT TIME =
     F14.8,6H DAYS.//1H ,10HMAXIMUM OFI3,15H SPECIAL POINTS.
     31H ARE ALLOWED WITHIN A COMPUTING STEP.//1H ,
     42HDUMP OF STORAGE ARRAY CEPS(*,*,2) FOLLOWS./1H )
   94 FORMAT(1H ,(10D13.5))
```


CHECK-31

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
89      FORMAT(1H ,43HFATAL ERROR. STORAGE TABLE *CEPS* EXCEEDED )
```

ORIGINAL PAGE IS
OF POOR QUALITY

CONVER-1

Name: CONVER

Calling Argument: T, ION, TO EQUA, TORIAL, SYSTEM, VECL, Z, RA, VPLAN

Referenced Sub-programs: None

Referenced Commons: None

Entry Points: None

Referencing Sub-programs: SWING

Discussion: This routine performs the conversion to the equatorial system (of the planet) of any vector expressed in the ecliptic system, i.e., the routine expresses the same vector in terms of the planet's standard coordinate system (z toward planet's north pole, defined by counterclockwise rotation looking down, x along ascending node of equatorial plane on the ecliptic, y completes right-handed system).

The planet's obliquity ϵ and equator ascending node ϕ are expressed as quadratic functions of the time elapsed since 1900, in Julian centuries:

$$\epsilon = \epsilon_0 + \epsilon_1 t + \epsilon_2 t^2 ,$$

$$\phi = \phi_0 + \phi_1 t + \phi_2 t^2 ,$$

where ϵ_0, ϵ_1, ϵ_2, ϕ_0, ϕ_1, and ϕ_2 are constants associated with the given planet, and are displayed in the DATA statement of the program source listing. Then the desired output vector V_p, which is the representation of the same vector (which may be position, speed, or any vector) V_e in planetary coordinates, where V_e is an input vector expressed in terms of ecliptic coordinates, is given by

$$V_p = A V_e$$

where

$$A = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi\cos\epsilon & \cos\phi\cos\epsilon & -\sin\epsilon \\ -\sin\phi\sin\epsilon & \cos\phi\sin\epsilon & \cos\epsilon \end{bmatrix} .$$

In the planetary reference, the vector's declination is computed as

$$\delta = \sin^{-1} (V_{pz}/|V_p|) ,$$

and its right ascension is

$$\alpha = \tan^{-1} (V_{py}/V_{px}) .$$

CONVER EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
T	UX		Time elapsed since 1900, t, in Julian centuries.
Z	SX		Declination of V_p, δ, in degrees.
RA	SUX		Right ascension of V_p, α, in degrees.
ION	UX		Planet selector.
VECL(3)	UX		Input vector, V_e, expressed in ecliptic coordinates, in arbitrary units.
VPLAN(3)	SUX		Output vector, V_p, expressed in planetary reference system and having same units as V_e.
SYSTEM	SX		Dummy descriptive argument.
TOEQUA	SX		Dummy descriptive argument.
TORIAL	SX		Dummy descriptive argument.

CHART TITLE - SUBROUTINE CONVER(T,ION,TOEQUA,TORIAL,SYSTEM,VECL,Z,RA,VPLAN)

C-2

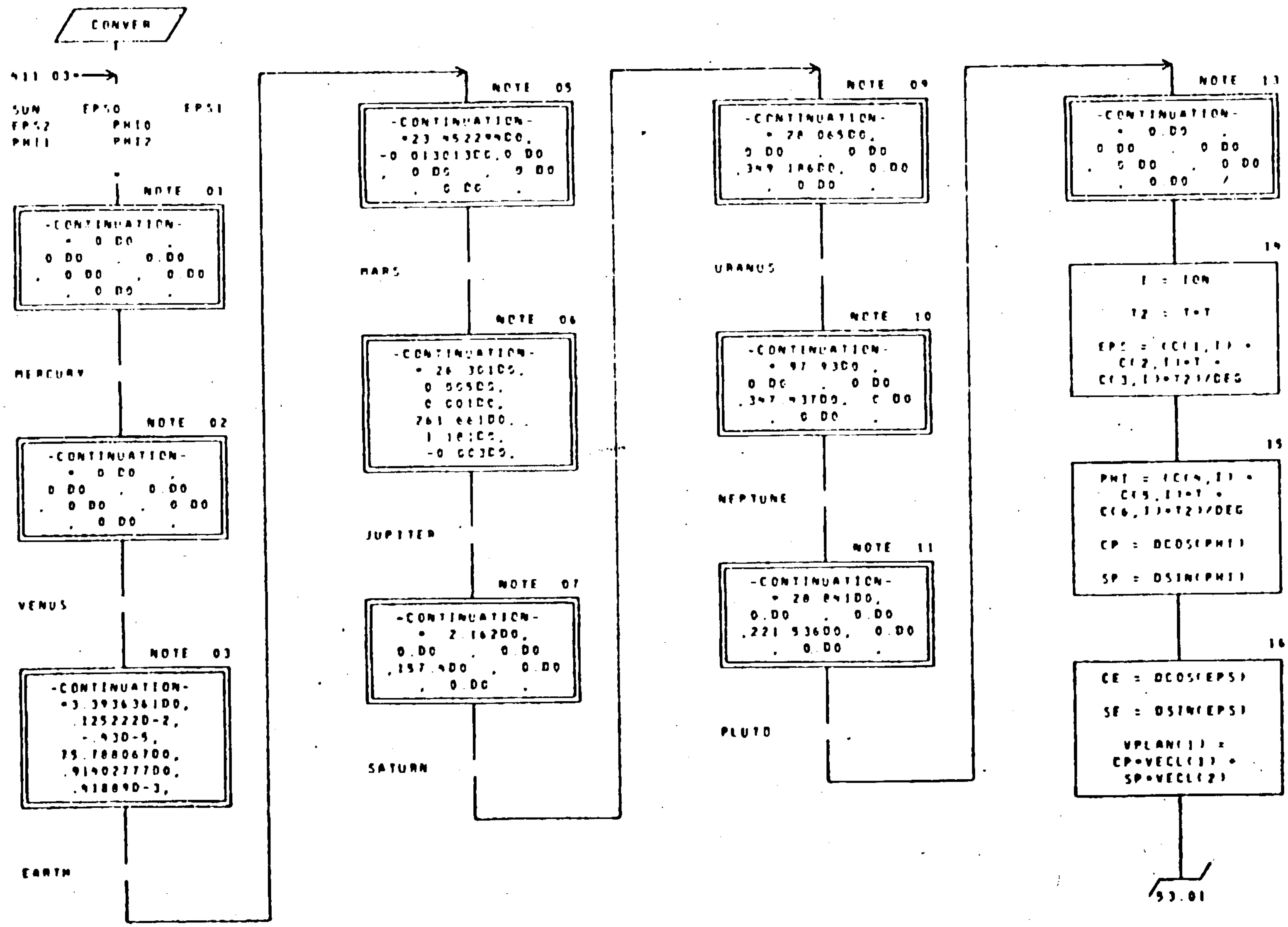


CHART TITLE - SUBROUTINE CONVERT(T,ION,TOEQUA,TORIAL,SYSTEM,VECL,Z,RA,VPLAN)

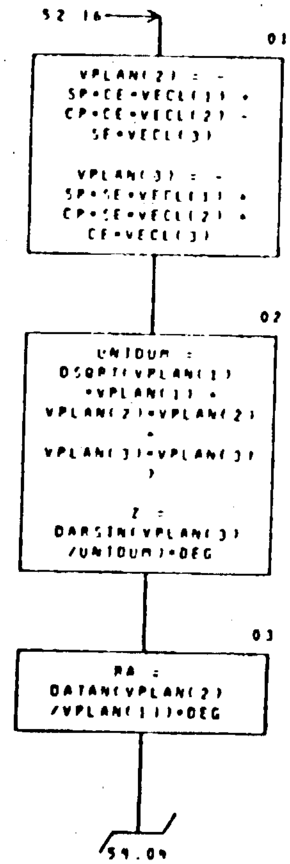
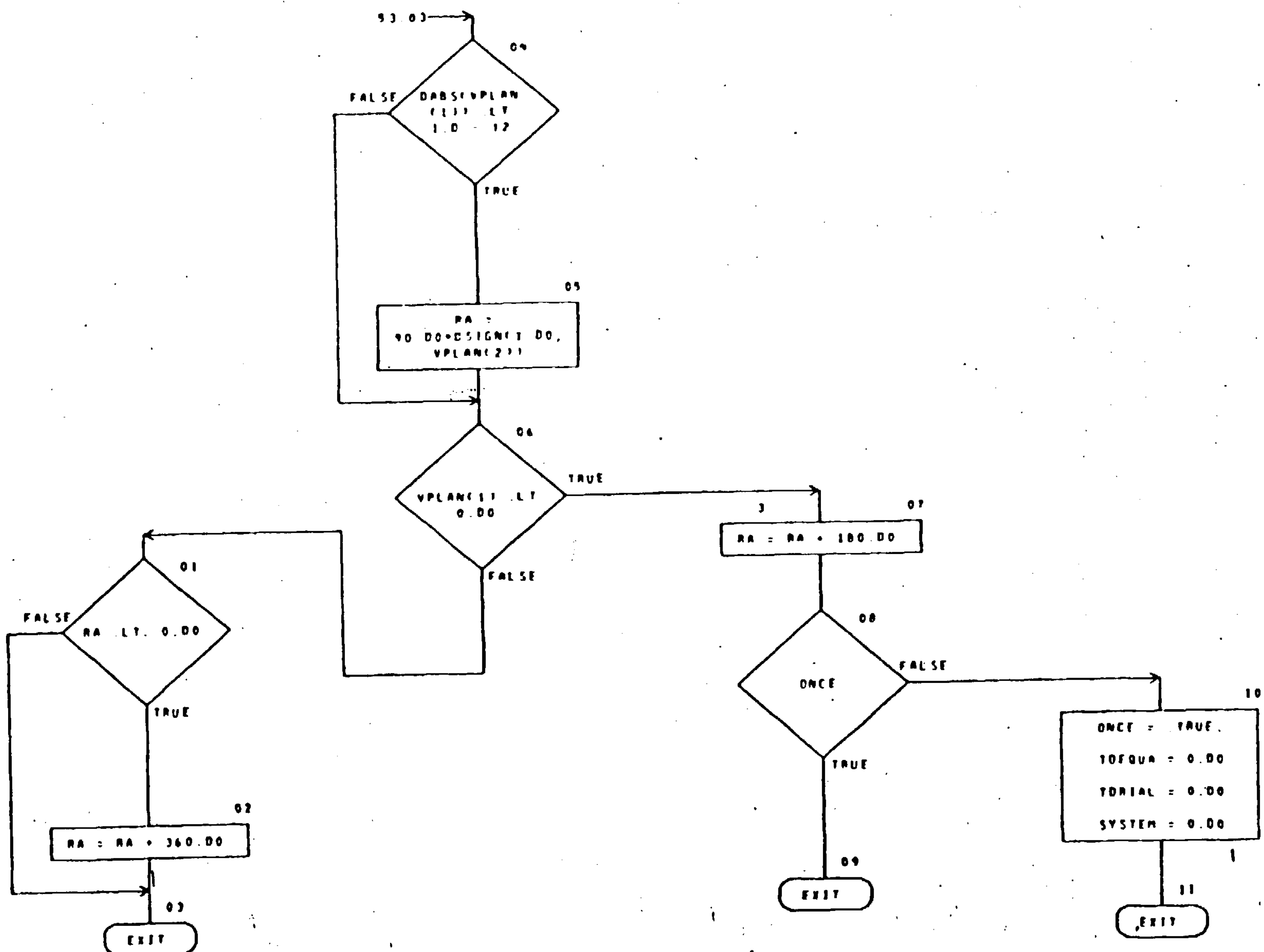


CHART TITLE - SUBROUTINE CONVER(T,ION,TOFQUA,TORIAL,SYSTEM,VECL,Z,RA,VPLAN)



CONVER-5

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION VECL(3),VPLAN(3),C(6,10)
LOGICAL ONCE
DATA C /
DATA DEG /57.295779513082320D0/
DATA ONCE / FALSE./
```

Name: CONVRT

Calling Argument: TRUE

Referenced Sub-programs: None

Referenced Commons: REAL8

Entry Points: None

Referencing Sub-programs: TAP

Discussion: Integration of the differential equations associated with the state and adjoint variables is done in two distinct modes which depend upon the value of the thrust switch function, σ. Numerical integration is required on thrust intervals, which occur whenever $\sigma > 0$. Otherwise, during any coast phase, the two-body equations of motion and the associated adjoint equations are known to possess analytic solutions obtainable in closed form. Time derivatives are used during coast phases, whereas generalized derivatives are used during thrust phases. The particular generalized derivatives used in the program are related to time derivatives by the formula

$$x' = r^n \dot{x}$$

where ' denotes generalized derivatives, $\cdot$ denotes time derivatives, r is the spacecraft's solar distance and n is an input constant. The problem formulation in the program is set up solely in terms of first-order derivatives, so that the above formula suffices to describe the conversion of derivatives between thrust and coast phases. In particular, the above formula applies explicitly at any switch point from coast to thrust, whereas

$$\dot{x} = x'/r^n$$

is employed at any switch point from thrust to coast.

CONVRT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
R(2)	U	REAL8	The spacecraft's distance from the center of the sun at the current time is stored in R(2), in AU.
X(50)	SU	REAL8	Array of trajectory dependent variables (including the derivatives which need to be converted.)
AN	U	REAL8	Input constant n used in the formula which converts the derivatives.
R1N	U	REAL8	The solar distance of the spacecraft raised to the n^{th} power, r^n, where n is the input constant AN. This is equivalent to R(2)**AN, but unnecessary exponentiation is avoided since r^n is always available during thrust phases.
TRUE	UX		Logical indicator: when true, spacecraft is beginning a coast phase; when false, spacecraft is beginning a thrust phase.

CHART TITLE - SUBROUTINE CONVRT(TRUE)

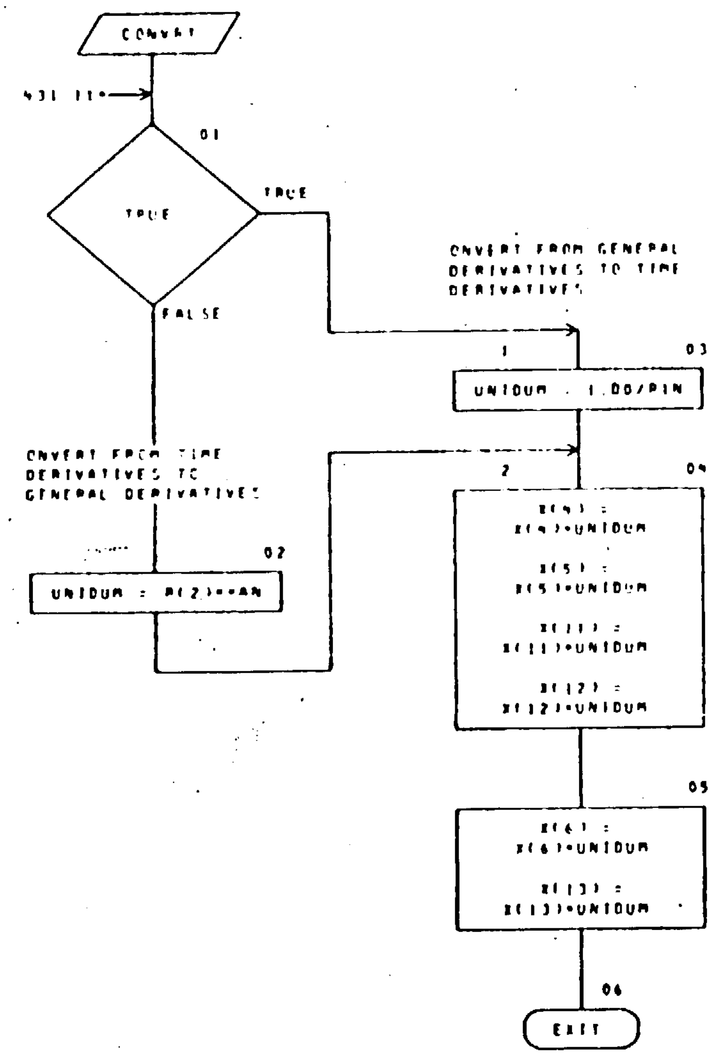


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL TRUE
COMMON /REALR/ R01(200),R(2),R02(98),AN,RIN,R03(948),
X(50),R04(700)
```

ORIGINAL PAGE IS
OF POOR QUALITY

Name:	CORNER
Calling Argument:	LOGIC for CORNER: TEST, ERROR for RIDGE
Referenced Sub-programs:	None
Referenced Commons:	INTGR4, ITERAT, LOGIC4, REAL8
Entry Points:	RIDGE
Referencing Sub-programs:	FINISH, MAIN for CORNER; TRAJ, TRAJI for RIDGE

Discussion: Subroutine CORNER controls the logic which automatically, but optionally, generates an intermediate forced-thrusting case in an attempt to avoid a so-called propulsion-time corner, which corresponds to a trajectory having at least one region with a very small thrust phase or a very small coast phase. When a trajectory is very near to a propulsion-time corner, the case is said to "be hung" on a propulsion-time corner if the MINMX3 iterator is not able to make any significant progress. If an automatic attempt to avoid the propulsion-time corner is optionally invoked via the NAMELIST input variable GAP, then subroutine CORNER makes preparations for the inclusion of two additional cases, in addition to any user-input cases, which immediately follow the hung case. The first additional case attempts to converge using forced-thrusting (no coast phases) on all trajectories, and, if convergence is achieved with forced-thrusting, then the second additional case attempts to obtain the user's desired solution (possibly with coast phases) using the trajectory from the first additional case as an initial guess.

Entry point RIDGE performs initialization at the beginning of each trajectory, and makes a key test at the end of each trajectory which is in a small neighborhood of a propulsion-time corner; if a sufficient number of such trajectories have accrued, then entry point RIDGE determines that the current iteration sequence is in fact hung on a propulsion-time corner, and halts the iteration sequence by declaring the existence of an error condition.

Messages and printouts: When entry point RIDGE determines that an iteration sequence is hung on a propulsion-time corner, the following is printed on unit 6:

CASE n ABORTED AFTER m TRAJECTORIES ENCOUNTERED
NUMERICAL DIFFICULTY WITH ENGINE SWITCHING

SUMMARY OF LAST TRAJECTORY

type (1) type (2) PHASE AT t-hung (1) DAYS
type (1) type (2) PHASE AT t-hung (2) DAYS
.
.
.

where n is the case number, m is the number of accrued trajectories (including perturbation trajectories) in a small neighborhood of a propulsion-time corner, type (1) = SMALL or NEAR-, and type (2) = THRUST or COAST, and the program will detect the first five propulsion-time corners encountered along the trajectory at times t-hung (1), t-hung (2), etc., and ignore the remainder. The following is also printed on unit 12:

CASE n HUNG ON SMALL THRUST/COAST PHAST AT T = t-hung (1) DAYS

Then, on unit 6, the following heading appears for the first additional case:

(TEMPORARY FORCED-THRUSTING CASE)

and, if that case (= case n+1) converges, then the following heading appears for the second additional case:

(TO GET CASE n FROM CASE n+1)

However, the above two additional cases are bypassed if any iterator triggers associated with propulsion time (inputs X8 or Y8) are on, in which case the following is printed on unit 6:

AUTOMATIC FORCED-THRUSTING CASE IS BYPASSED BECAUSE
X8 AND/OR Y8 TRIGGERS ARE ON.

and the following is printed on unit 12:

FORCED THRUSTING BYPASSED BECAUSE X8 AND/OR Y8 ARE ON

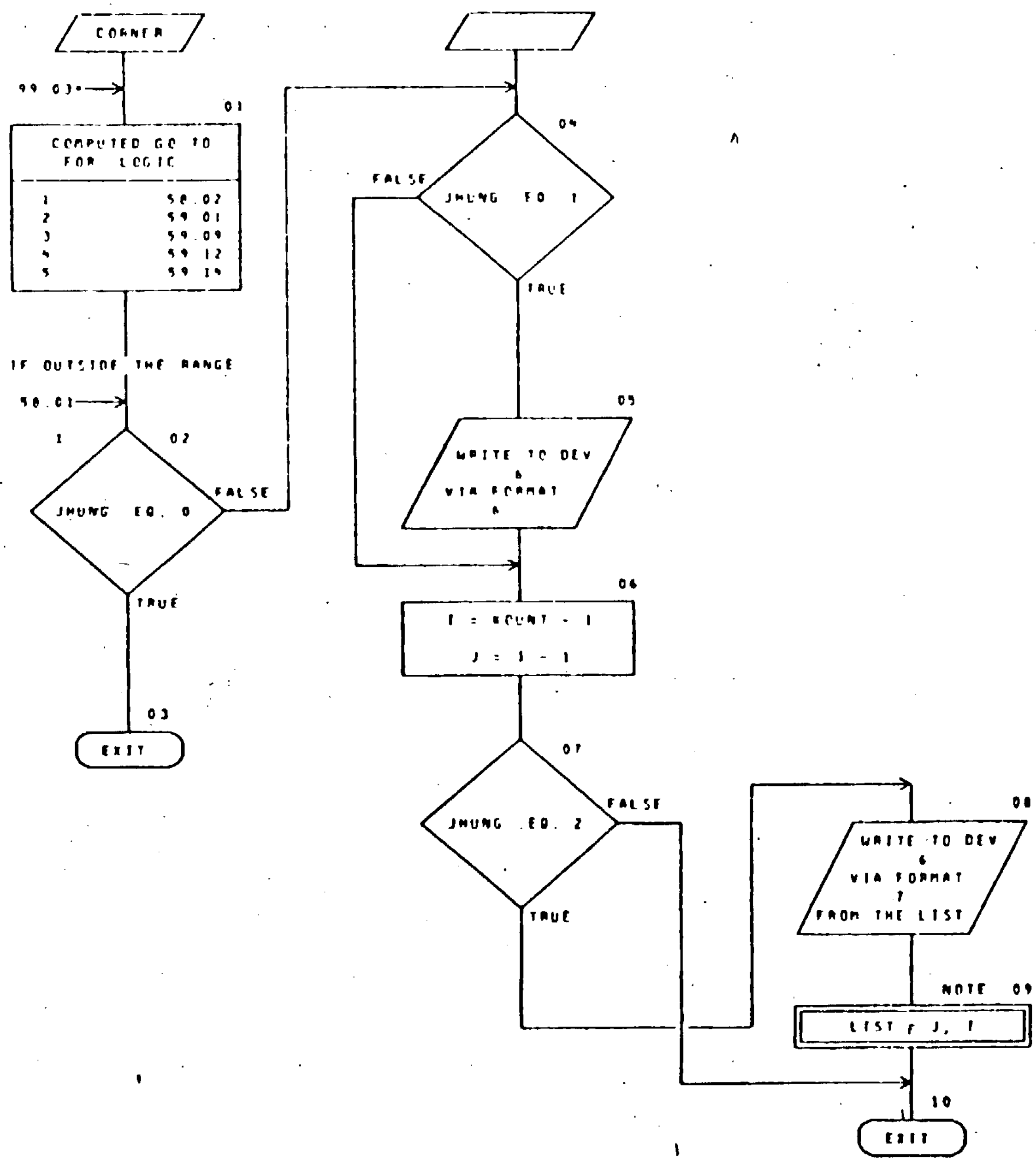
CORNER EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
BX(5,70)	SU	ITERAT	Iterator independent variable array.
BY(3, 70)	U	ITERAT	Iterator dependent variable array.
HUNG	S	LOCIC4	Status indicator as to whether current iteration sequence is hung on a propulsion-time corner or not.
IROT	SU	INTGR4	A program input parameter which must be temporarily nullified by subroutine CORNER; initial primer vector rotation indicator.
MOPT	SU	INTGR4	A program input parameter which must be temporarily nullified by subroutine CORNER; ballistic option indicator.
QJEX	U	LOGIC4	Final (summary) trajectory indicator for a given case.
TEST	UX		Indicator for distinguishing between initialization and testing.
CONTM	U	REAL8	Time conversion factor, tau to days.
ERROR	UX		Error indicator for the current case.
IHUNG	SU	INTGR4	Current number of propulsion-time corner occurrences along the current trajectory.
JHUNG	SU	INTGR4	Indicator for controlling the two additional cases which attempt to avoid the propulsion-time corner(s).
KOUNT	U	INTGR4	Case counter.

CORNER EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
KPART	SU	INTGR4	A program input parameter which must be temporarily nullified by subroutine CORNER; option indicator for automatically selecting improved independent parameter perturbations for generating the iterator's partial derivative matrix.
LOGIC	UX		Indicator for controlling the saving and restoring operations performed by subroutine CORNER.
MREAD	SU	INTGR4	A program input parameter which must be temporarily nullified by subroutine CORNER; card input option indicator for iterator independent variables.
NHUNG	U	INTGR4	Iteration sequence cutoff condition, input to the program, equal to the maximum permissible number of trajectories which may encounter a propulsion-time corner within a given iteration sequence.
THUNG(5)	SU	REAL8	Array containing up to five consecutive times at which propulsion-time corners occurred along the current trajectory.
HOHUNG(5)	U	REAL8	Array containing up to five consecutive thrust switch function time derivatives, corresponding to THUNG(i).
MUPDAT	SU	INTGR4	A program input parameter which must be temporarily nullified by subroutine CORNER; indicator for subsequent-case initial-guess update.
SWHUNG(5)	U	REAL8	Array containing up to five consecutive thrust switch function values, corresponding to THUNG(i).

CHART TITLE - SUBROUTINE CORNER(LOGIC)



CORNER-5

CHART TITLE - SUBROUTINE CORNER(LOGIC)

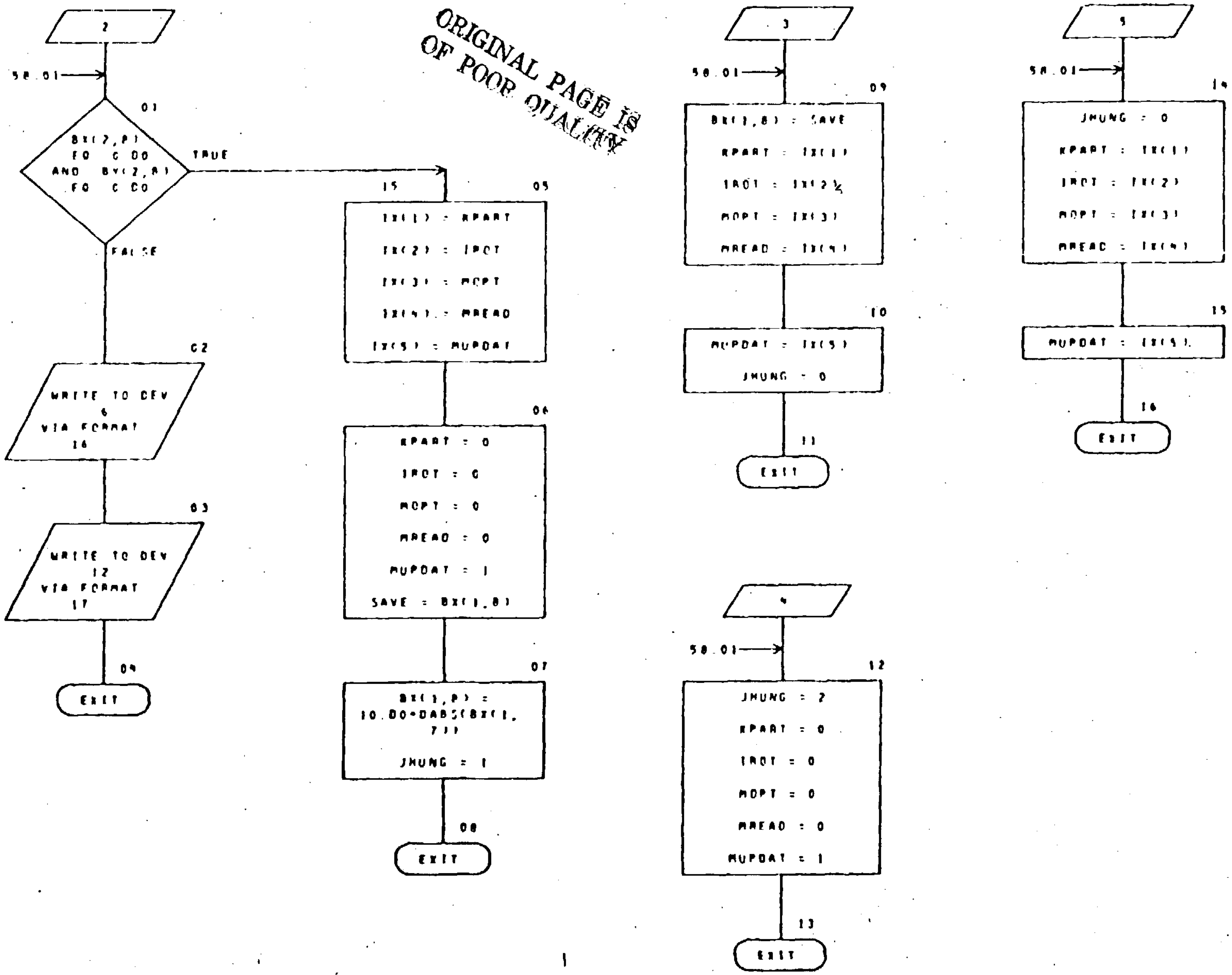
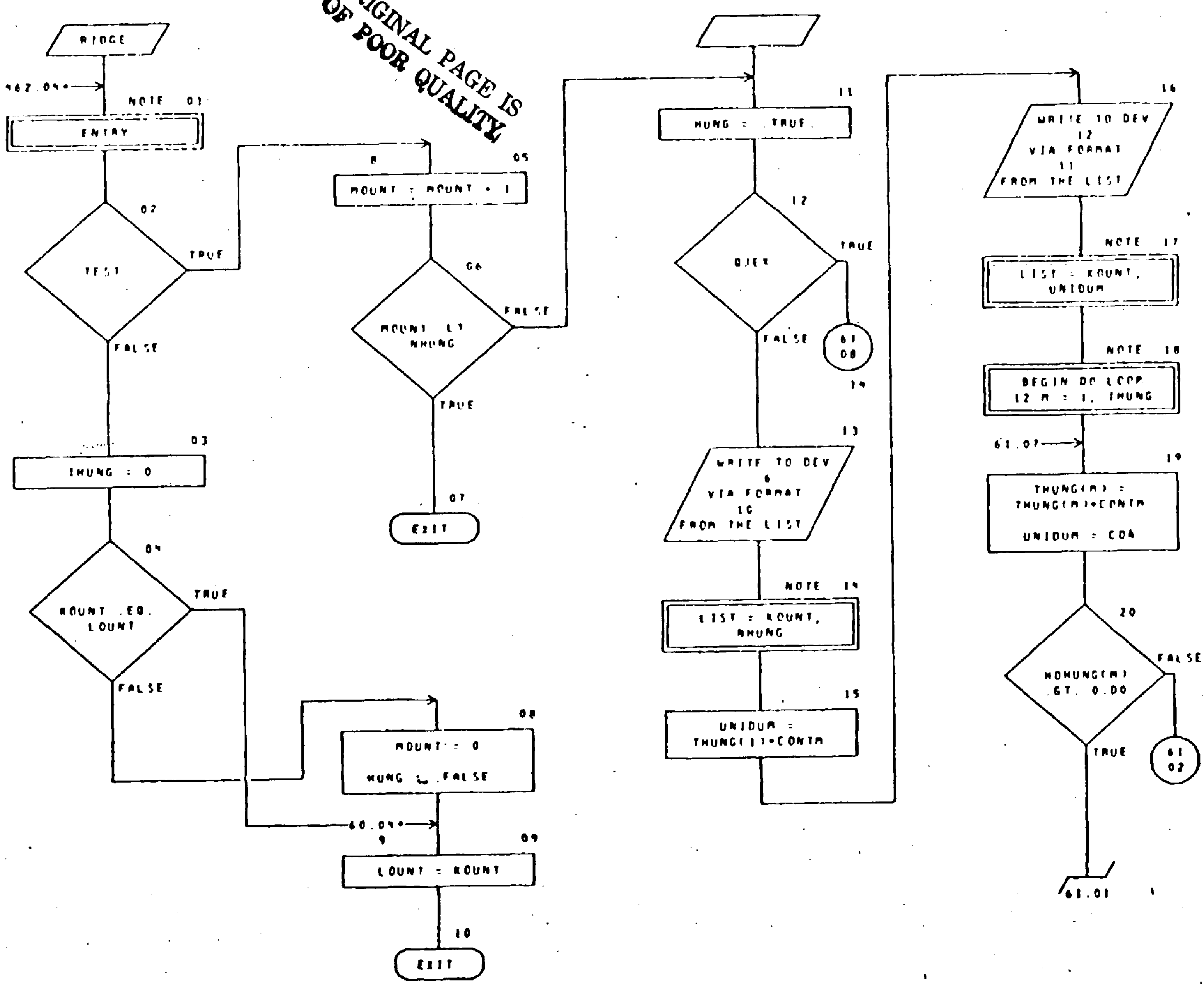


CHART TITLE - SUBROUTINE CORNER(LOGIC)



CORNER-7

CHART TITLE - SUBROUTINE CORNER(LOGIC)

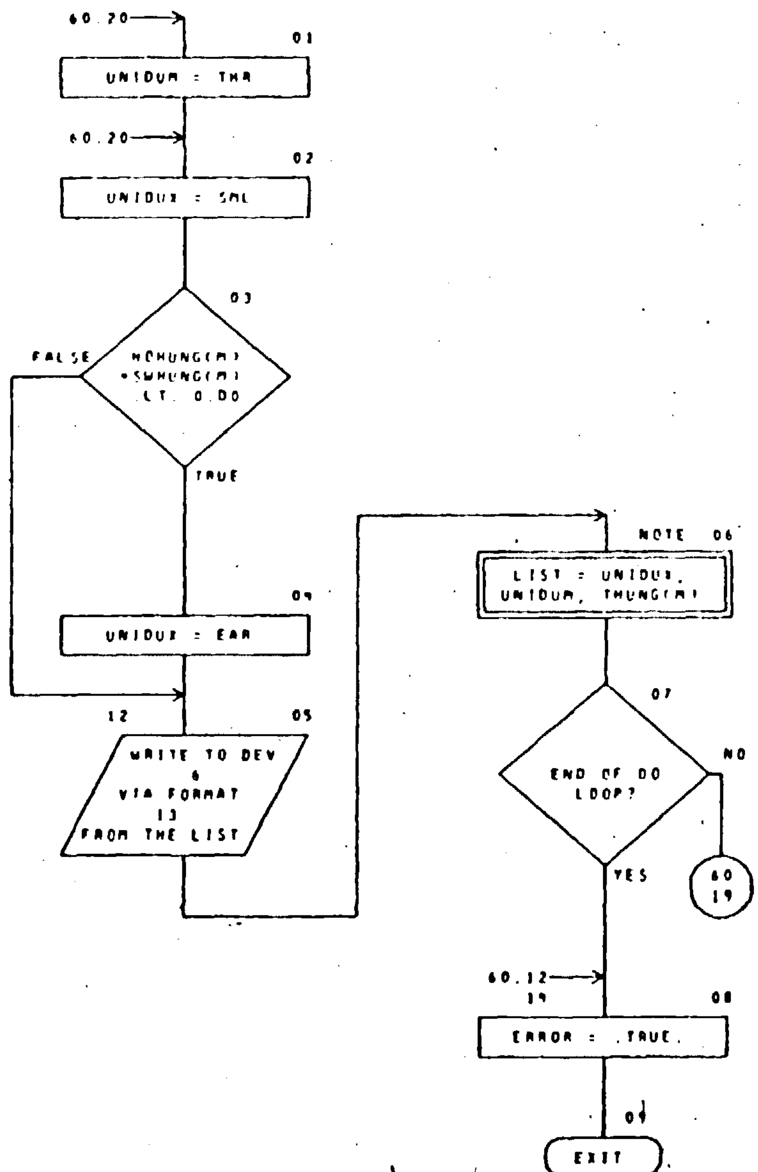


CHART TITLE NON-PROCEDURAL STATEMENTS


```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL TEST,HUNG,ERROR,OJFX
      DIMENSION IX(5)
      COMMON /REAL8/ RO1(103),THUNG(5),SWHUNG(5),HOHUNG(5),RO2(197),
     CONTN,RO3(1A84)
      COMMON /INTGR4/ IO1(7),MCPT,IO2(4),MREAD,MUPDAT,IO3(6),IROT,
     IO4(28),KPART,IHUNG,NHUNG,IO6,KOUNT,IO7(11),JHUNG,IO8(934)
      COMMON /LOGIC4/ LO1(5),HUNG,LO2(20),OJFX,LO3(473)
      COMMON /ITERAT/ BX(5,70),BY(3,70),BO1(420)
      DATA THP,COA,SML,EAR /6HTHRUST,5HCOAST,5HSMALL,5HNEAR-/
      DATA KOUNT /0/
6     FORMAT(1H+,59X33H(TEMPORARY FORCED-THRUSTING CASE))
7     FORMAT(1H+,59X12H(TO GET CASE13,10H FROM CASE13,1H))
16    FORMAT(1H0,55HAUTOMATIC FORCED-THRUSTING CASE IS BYPASSED ,
     .37HBECAUSE XB AND/OR YB TRIGGERS ARE ON )
17    FORMAT(1H ,53HFORCED-THRUSTING BYPASSED BECAUSE XB AND/OR YB ARE O
     N)
10    FORMAT(1H1,4HCASE13,14H ABORTED AFTER13,60H TRAJECTORIES ENCOUNTER
     ED NUMERICAL DIFFICULTY WITH ENGINE SWITCHING//1H ,
     26HSUMMARY OF LAST TRAJECTORY/1H )
11    FORMAT(1H ,4HCASE13,33H HUNG ON SMALL THRUST/COAST PHASE,
     7H AT T =F7.1,5H DAYS)
13    FORMAT(1H ,A5,I1,A6,9H PHASE ATF12.6,5H DAYS)
```


CORNER-9

<u>Name:</u> DATE1

<u>Calling Argument:</u> IY, IM, ID, HOUR, DJ

<u>Referenced Sub-programs:</u> None

<u>Referenced Commons:</u> None

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> AEINWT, QSTART

<u>Discussion:</u> This routine computes the Julian date DJ given the year IY, month IM, day ID, and the hour of the day. The leap year adjustment formula is valid up to the year 2300. The routine was originally written in single precision and was later converted to double precision by adjusting the single precision logic appropriately.

The number of full days elapsed in the current year is computed:

day = (accumulated days to first of month) plus (days elapsed this month)

and if

(1) month $\geq$ March

and

(2) $4.01\,[.251[\text{year} - 1900]] > [\text{year} - 1900]$

then

day $\rightarrow$ day + 1 (leap year adjustment),

where [ ] is "the integer part of" the quantity in brackets.

The Julian date is then given by:

$$\text{Julian date} = [365.249[\text{year} - 1900]] + \text{day} + \text{hour}/24 + 15019.5$$

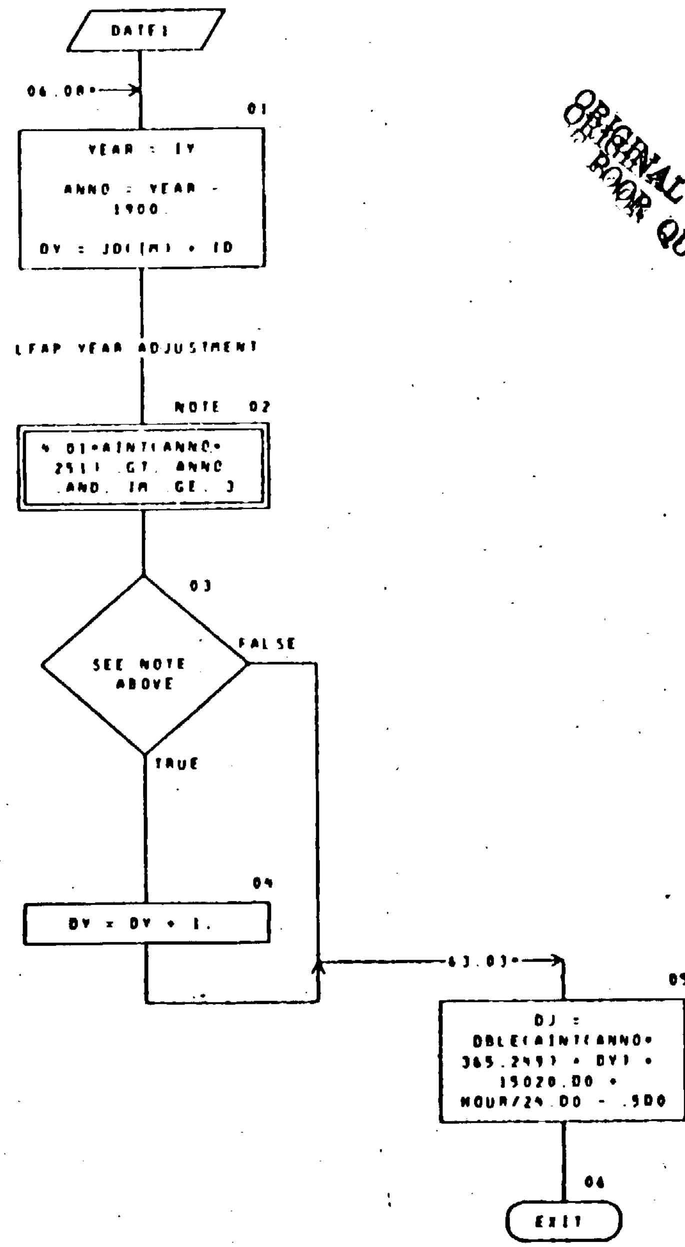
where this is the Julian date less 2400000, and 15019.5 is the Julian date (less 2400000) of 1900.

PRECEDING PAGE BLANK NOT FILMED

DATE1 EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
DJ	SX		Julian date, less 2400000, output from the routine, in days.
ID	UX		Input whole days elapsed in the month IM, plus one; that is, ID is the day indicator.
IM	UX		Input whole months elapsed in the year IY, plus one; that is, IM is the month indicator.
IY	UX		Input year (e.g., 1988).
HOUR	UX		Hours elapsed (including fraction of hour) in the current day (e.g., 17.80439).

CHART TITLE - SUBROUTINE DATE1(IY,IM,ID,HOUR,DJ)



01/08/75

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
DIMENSION JD(12)
DOUBLE PRECISION HOUR,DJ
DATA JD /0,31,59,90,120,151,181,212,243,273,304,334/
```

ORIGINAL PAGE IS
OF POOR QUALITY

<u>Name</u>: DECLIN

<u>Calling Argument</u>: None

<u>Referenced Sub-programs</u>: None

<u>Referenced Commons</u>: LOGIC4, REAL8

<u>Entry Points</u>: None

<u>Referencing Sub-programs</u>: OMASS, PRINT

<u>Discussion</u>: This subroutine computes the launch hyperbolic excess velocity asymptote declination δ given the launch hyperbolic excess velocity. When running two-dimensional trajectories in the xy plane,

$$\delta = 0.$$

Otherwise, in general,

$$\delta = \sin^{-1}\left[\frac{v_{\infty y}\sin\epsilon + v_{\infty z}\cos\epsilon}{\sqrt{v_{\infty x}^{2} + v_{\infty y}^{2} + v_{\infty z}^{2}}}\right]$$

where $V_{\infty} = (v_{\infty x}, v_{\infty y}, v_{\infty z})$ is the launch hyperbolic excess velocity in the ecliptic coordinate system and $\epsilon = 23^{\circ}.45$ is the obliquity of the Earth's equatorial plane to the ecliptic.

DECLIN EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
CE	U	REAL8	Cosine of obliquity, cos ϵ.
SE	U	REAL8	Sine of obliquity, sin ϵ.
DECL	S	REAL8	Launch hyperbolic excess velocity asymptote declination, δ, in radians.
VIMP	U	REAL8	Magnitude of launch hyperbolic excess velocity, in AU/tau.
V00D(3)	U	REAL8	Launch hyperbolic excess velocity in ecliptic coordinates, in AU/tau.
TUDFLG	U	LOGIC4	Indicator for two-dimensional trajectories in the xy plane.

CHART TITLE - SUBROUTINE DECLIN

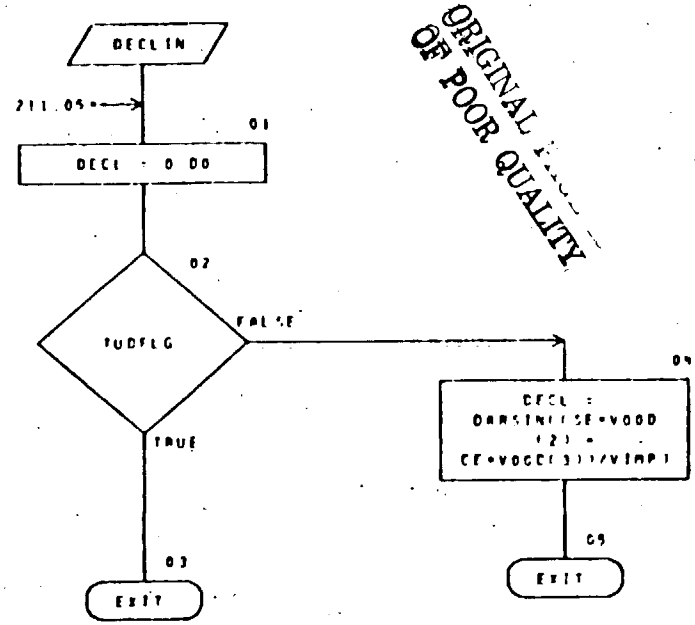


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL TUDFLG
COMMON /REAL8/ R01(13),VIMP,R05(155),E,SE,CF,DECL,
 R02(130),VOOD(3),R03(1694)
COMMON /LOGIC4/ L01(8),TUDFLG,L02(491)
```

Name: DERIV

Calling Argument: None

Referenced Sub-programs: SOLAR

Referenced Commons: INTGR4, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: RKSTEP, TAP

Discussion: This routine contains the differential equations governing the motion of the spacecraft in heliocentric space during thrust periods; the primary purpose is therefore to compute the derivatives XD(i) of the trajectory dependent variables, which are itemized in the discussion of subroutine RKSTEP, with respect to the trajectory independent variable, which is the generalized universal anomaly, u (and which may be represented elsewhere in this document by the more general symbol β).

The relationship between derivatives with respect to time and u is given by

$$\frac{du}{dt} = r^{-n} ,$$

where r is the spacecraft's solar distance and n is an input constant. Denoting derivatives with respect to u with the prime, the conversions from time to u derivatives are

$$x' = r^n \dot{x} ,$$

$$x'' = r^{2n} \left(\frac{R \cdot \dot{R}}{r^2} n\dot{x} + \ddot{x} \right) .$$

The time-derivatives of the trajectory dependent-variables, which are transformed to generalized derivatives via the above relations, are expressed as follows, where h_σ has the value unity:

The spacecraft position satisfies the differential equation

$$\ddot{R} = h_\sigma \frac{g\gamma q}{\nu} \bar{e}_t - \frac{\mu}{r^3} R .$$

The mass ratio satisfies the differential equation

$$\dot{\nu} = - h_\sigma \frac{g\gamma q}{c} ,$$

The adjoint equations are

$$\ddot{\Lambda} = - \frac{\mu}{r^3} \Lambda + \frac{3\mu}{r^5} (R \cdot \Lambda) R + h_\sigma \left[\frac{g\gamma^* q}{\nu} (\Lambda \cdot \bar{e}_t - \frac{\nu}{c} \lambda_\nu) + \lambda_s \right] \frac{\partial d}{\partial R} ,$$

$$\dot{\lambda}_\nu = h_\sigma \frac{g\gamma q}{\nu^2} (\Lambda \cdot \bar{e}_t),$$

$$\dot{\lambda}_g = - h_\sigma \frac{\gamma q}{\nu} (\Lambda \cdot \bar{e}_t - \frac{\nu}{c} \lambda_\nu) ,$$

$$\dot{\lambda}_c = - h_\sigma \frac{g\gamma q}{c^2} \lambda_\nu ,$$

$$\dot{\lambda}_s = h_\sigma \frac{g\gamma q}{\nu \tau_d} (\Lambda \cdot \bar{e}_t - \frac{\nu}{c} \lambda_\nu)$$

where

$$\gamma^* = \partial\gamma/\partial d,$$

and

$$\frac{\partial d}{\partial R} = \frac{1}{r^3} \left[\bar{n} - 3(\bar{e}_r \cdot \bar{n}) \bar{e}_r \right] = - \frac{2}{r^4} R, \text{ when it is not zero.}$$

The time time-derivative is, of course, unity ($\dot{t} = 1$), and the time generalized derivative is simply $t' = r^n$. The degradation-time derivative is

$$\dot{s} = h_\sigma d .$$

The adjoint variable derivative associated with fixed thrust cone-angle is computed as follows: let λ_x be a Lagrange multiplier that is identically zero if thrust cone angle is unconstrained and defined

$$\lambda_x = -h_\sigma \frac{g\gamma q}{\nu} \frac{\Lambda \cdot (\bar{m} \times \bar{e}_t)}{R \cdot (\bar{m} \times \bar{e}_t)} ,$$

if the cone angle is fixed. The unit vector $\bar{m}$ is defined

$$\bar{m} = \frac{R \times \Lambda}{|R \times \Lambda|} .$$

Then the differential equation for the variable adjoint to the thrust cone angle ϕ is

$$\dot{\lambda}_\phi = \lambda_x R \cdot (\bar{m} \times \bar{e}_t).$$

The thrust unit vector $\bar{e}_t$ is required in subroutine DERIV when the thrust cone-angle is held constant. (Otherwise, $\bar{e}_t$ is aligned with Λ and therefore need not be computed). Rather than obtaining $\bar{e}_t$ from subroutine THANG, $\bar{e}_t$ is computed directly within DERIV, in order to minimize computation time. The equations for the computation of $\bar{e}_t$ are given in the discussion of subroutine THANGD.

DERIV EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
X(50)	U	REAL8	Array of trajectory dependent-variables (see subroutine RKSTEP).
AN	U	REAL8	Exponent n in the conversion formula for generalized derivatives.
FT	U	REAL8	Reference thrust acceleration, g, in AU/tau^2.
RC	SU	REAL8	Cube of solar distance, r^3, in AU^3.
RS	SU	REAL8	Square of solar distance, r^2, in AU^2.
RT	SU	REAL8	Spacecraft solar distance, r, in AU.

DERIV EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
XD(50)	SU	REAL8	Array of trajectory dependent-variable derivatives (with respect to generalized universal anomaly, u), in same order as indicated by the discussion of subroutine RKSTEP.
ACC	SU	REAL8	Instantaneous thrust acceleration, $a = g\gamma q/\nu$, in AU/tau^2.
AVJ	U	REAL8	Inverse of jet exhaust speed, 1/c, in EMOS^{-1}.
ETH(3)	SU	REAL8	Thrust unit vector, $\bar{e}_t$.
PLC	SU	REAL8	First component of thrust switch functions, σ_1, (see FUNCT).
PMN	SU	REAL8	Primer magnitude, λ.
PMS	SU	REAL8	Square of primer magnitude, λ^2.
R1N	SU	REAL8	Spacecraft solar distance raised to the n^{th} power, r^n, used in converting between generalized and time derivatives.
R2N	SU	REAL8	r^{2n} (see R1N).
AXIS(3)	S	REAL8	Spacecraft spin-axis unit vector (not used at present).
FLAP	U	LOGIC4	Indicator for a power function curve which is a constant.
PLUS	U	LOGIC4	Indicator for determining appropriate region in two-dimensional simulations, as described in THANGD.
POWR	U(S)	REAL8	Power ratio, γq.
SWIT	SU	REAL8	$\sigma_1 + \sigma_2$; first two terms of the thrust switch function, as described in subroutine FUNCT.

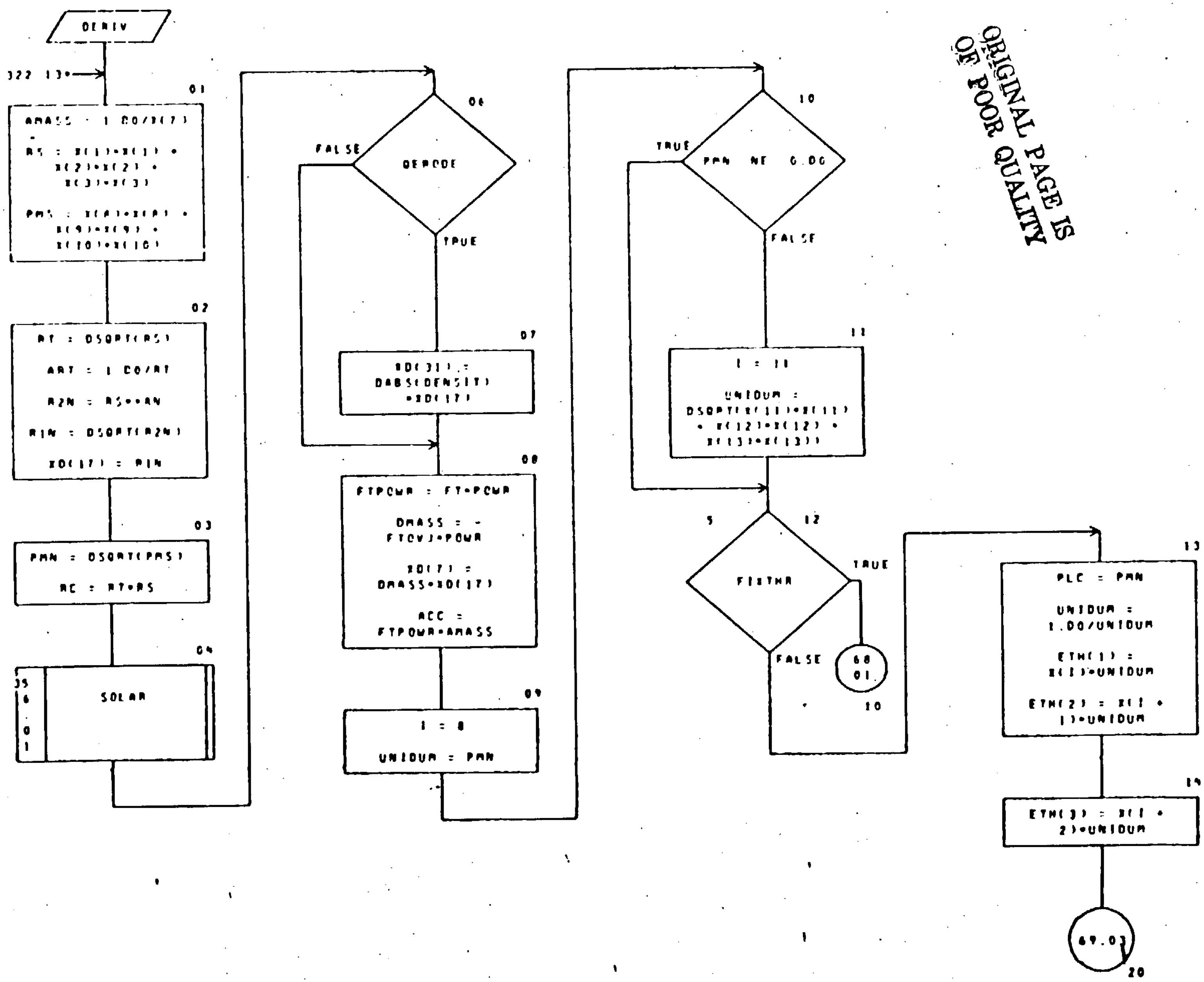
DERIV EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
COPHI	U	REAL8	Cosine of fixed thrust angle, cos ϕ.
DMASS	SU	REAL8	Mass-ratio time-derivative, $\dot{\nu}$, in tau^{-1}.
DPOWR	U (S)	REAL8	$q\,\partial\gamma/\partial r$.
ERODE	U	LOGIC4	Power degradation option indicator.
FTOVJ	U	REAL8	Ratio of reference thrust acceleration to jet exhaust speed, g/c, in tau^{-1}.
PCURV	U	LOGIC4	Indicator for condition in which solar arrays are oriented to receive the maximum power permissible under the current power-curve assumption, or to be tilted away from the maximum permissible due to degradation considerations.
SIPHI	U	REAL8	Sine of fixed thrust angle, sin ϕ.
DENSIT	U (S)	REAL8	Power density-function, d, in AU^{-2}.
FIXTHR	U	LOGIC4	Indicator for fixed thrust-angle.
NPHI20	U	INTGR4	Index for currently-active fixed thrust angle; only one value is presently allowed, 21. Index selects adjoint variable derivative in DERIV.
QERODE	U	LOGIC4	Indicator which is true when either ERODE or QJEX (final case-summary trajectory) is true.
REGION	U	LOGIC4	Indicator for spacecraft solar proximity; demarks two possible regions in space, separated by sphere about sun of specified radius, at which power function (or its derivative) has a corner.

DERIV EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
TAUPOW	U	REAL8	Negative inverse of characteristic degradation time, $-1/\tau_d$, in tau^{-1}.
TUDFLG	U	LOGIC4	Indicator for two dimensional trajectory simulation ("2D flag").

CHART TITLE - SUBROUTINE DERIV



DERIV-7

CHART TITLE - SUBROUTINE DERIV

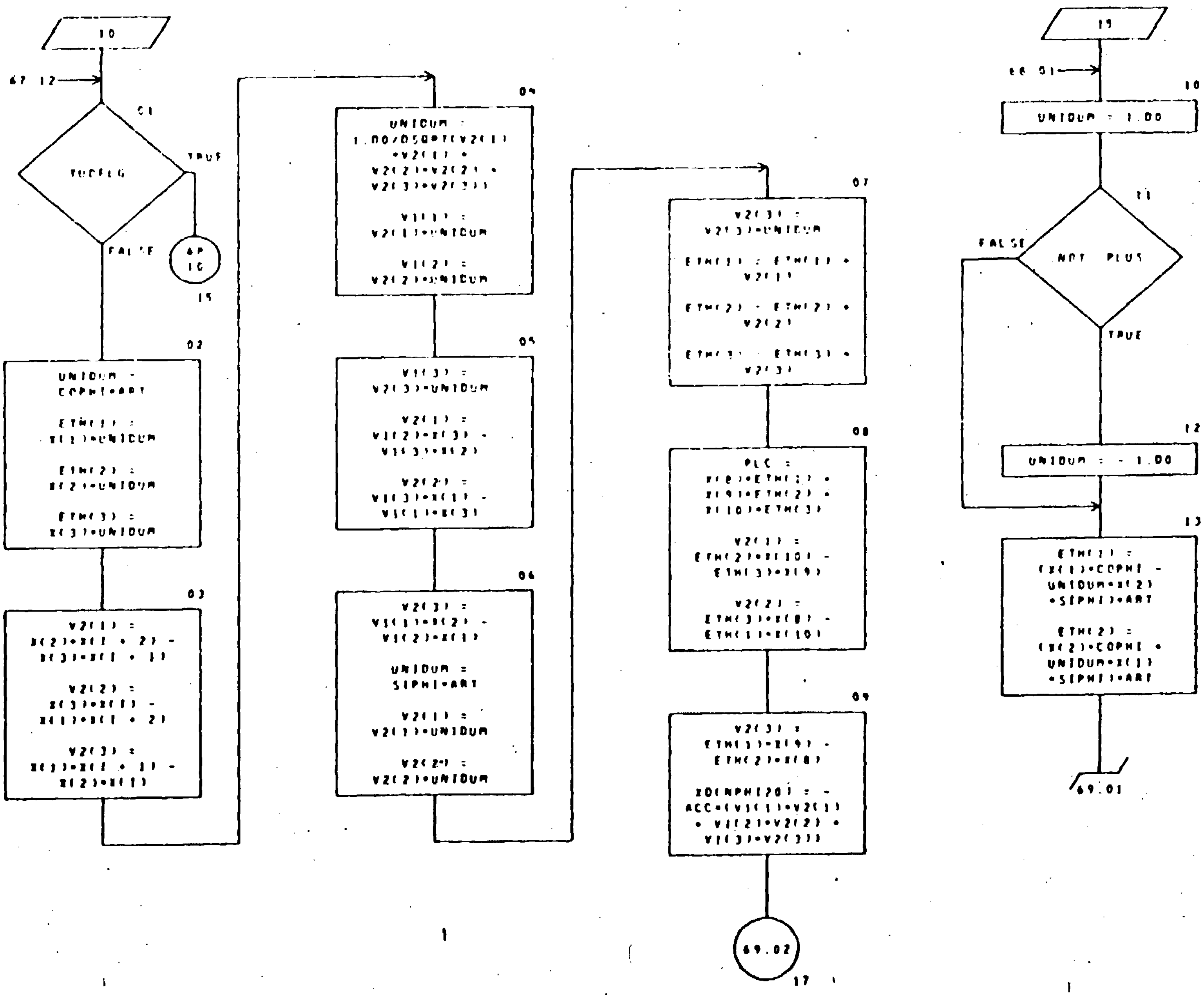
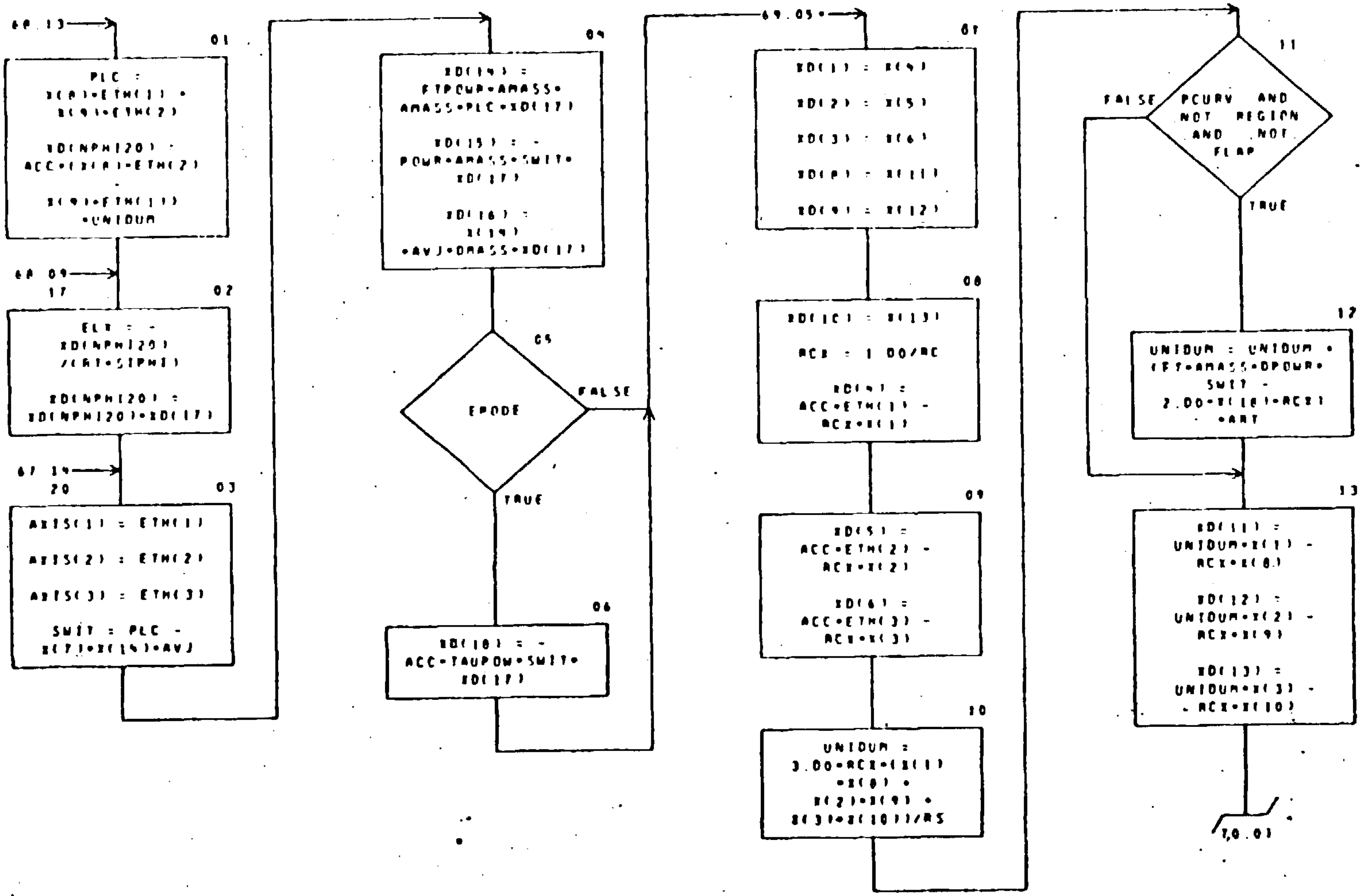


CHART TITLE - SUBROUTINE DERIV



DERIV-9

CHART TITLE - SUBROUTINE DERIV

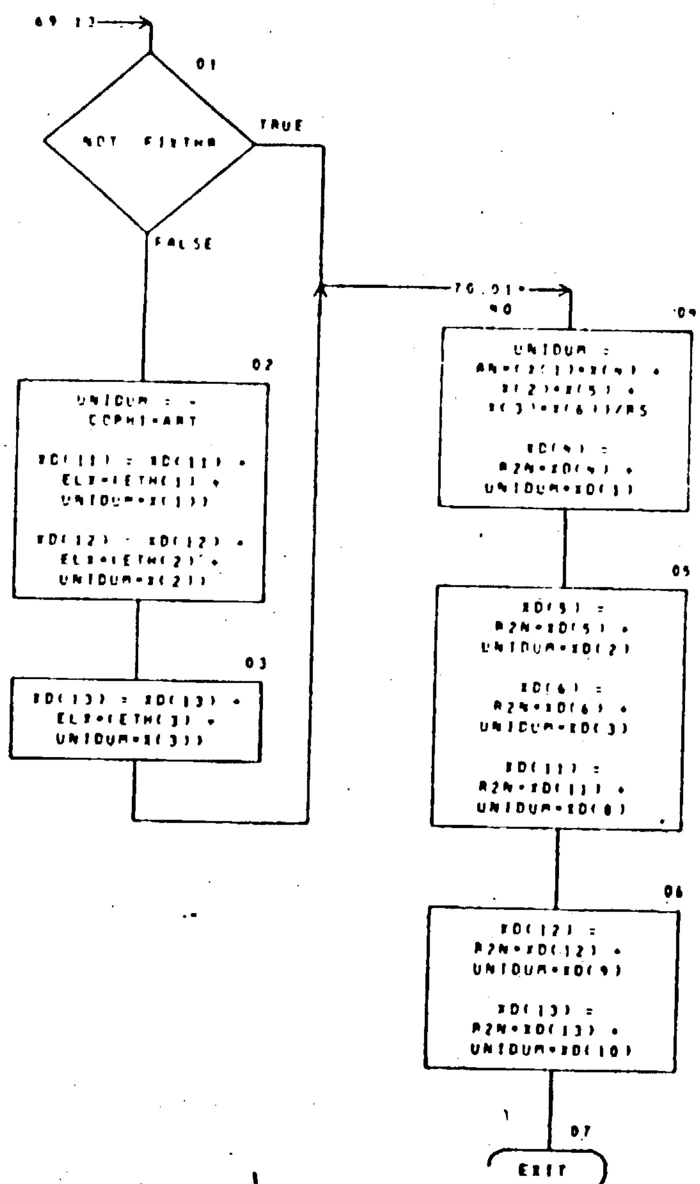


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL FIXTHR,DERODE,ERODE,REGION,PCURV,FLAP,TUDFLG,PLUS
DIMENSION V1(3),V2(3)
COMMON /REAL8/ RO1(11),FT,RO2(288)
,AN,RIN,R2N,RO5(390),AXIS(3),RO6(129),ETH(3),RO7(12),
SWIT,SIPHI,COPHI,PLC,RO8(301),FMN,
PMS,RT,RS,RC,ACC,R10(3),     POWR,DPOWR,TAUPOW,DMASS,R12(11),AVI,
R11
,FTOVJ,R13(4),DENSIT,R14(73),             X(50),XD(50),R16(650)
COMMON /INTGR4/ IO1(326),NPH120,IO2(673)
COMMON /LOGIC4/ LO1(8),TUDFLG,LO6,FIXTHR,LO3(7),REGION,ERODE,FLAP,
LO5(11),DERODE,LO4(3),PCURV,LO2(2),PLUS,LO7(460)
```

Name: EFM

Calling Arguments: N, T, X, XD, INTER

Referenced Sub-programs: None

Referenced Commons: REAL8

Entry Points: None

Referencing Sub-programs: ALBEDO, PUNCH, QSTART, RADAR, SPRINT, SWING, SWTRAJ, TRAJ, TRAJI

Discussion: Subroutine EFM is an analytical ephemeris routine which computes the heliocentric position and velocity of a planet or other solar system object in the mean equinox and ecliptic frame of date. Mean elements are evaluated as quadratic functions of time for eccentricity e, inclination i, longitude of node Ω, longitude of periapse Π, and mean longitude θ. The sixth element, the semi-major axis a, is assumed constant in time. Coefficients and constants are stored in data arrays for each of the nine planets of the solar system. The values currently in use are listed in the table. The gravitational constant μ for each planet is obtained from the GM array. Provisions are available for inputting constant elements for an arbitrary body, referred to as Oddball in this subroutine description.

The mean elements are evaluated with the formulas,

$$e = e_1 + e_2 \tau + e_3 \tau^2,$$

$$i = i_1 + i_2 \tau + i_3 \tau^2,$$

PRECEDING PAGE BLANK NOT FILMED

$$\Omega = \Omega_1 + \Omega_2 \tau + \Omega_3 \tau^2,$$

$$\Pi = \Pi_1 + \Pi_2 \tau + \Pi_3 \tau^2,$$

$$\theta = \mathrm{mod}\left[\theta_1 + \mathrm{sign}(\tau')(\theta_2 |\tau'| + \theta_3 \tau'^2)\right],$$

where the subscripted characters denote stored coefficients for a given celestial body, τ is the date on which the position and velocity are to be evaluated measured in Julian centuries from January 0.5, 1900, τ' is the same date but measured from January 0.5, 1965, and mod signifies that θ is evaluated modulo 2π (i.e., $0 \le \theta \le 2\pi$ radians).

The constant elements for Oddball are input in terms of the semi-major axis, the eccentricity, the inclination, the longitude of ascending node, the argument of perihelion, ω, and the Julian date of perihelion passage, t_p. The eccentricity, inclination, and longitude of ascending node are immediately stored in the locations for e_1, i_1, and Ω_1, respectively. Π_1 is equated to $\Omega_1 + \omega$; τ' is set to the current date minus the date of perihelion passage; and

$$\theta_1 = \Pi_1 ,$$

$$\theta_2 = 200\pi\sqrt{1/a^3} .$$

The coefficients e_2, e_3, i_2, i_3, Ω_2, Ω_3, Π_2, Π_3 and θ_3 are set to zero, and the quadratic equations defined above are then used to evaluate the current osculating elements for the arbitrary body.

The mean anomaly M is evaluated,

$$M = \theta - \Pi ,$$

from which the eccentric anomaly E is calculated by solving the equation,

$$M = E - e \sin E ,$$

iteratively using Newton's method. A total of 100 iterations are permitted to converge on E to a tolerance of 10^{-15} radians. If this number of iterations is reached, a message is printed and the current value of E is used for all subsequent calculations. The argument of perigee ω is evaluated

$$\omega = \Pi - \Omega$$

The radial distance is then evaluated,

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

and the true anomaly is given by

$$f = 2\tan^{-1}\left[\sqrt{\frac{1+e}{1-e}}\ \frac{\sin(E/2)}{\cos(E/2)}\right]$$

and the position vector is obtained in the orbit plane coordinates

$$x_\omega = r\cos f,$$

$$y_\omega = r\sin f,$$

where x_ω is the component along the perihelion vector and y_ω is the component along the semi-latus rectum, positive in the sense of motion at perihelion.

The transformation matrix A that expresses the position vector in the ecliptic system used by the program is then evaluated. This matrix is

$$A = \begin{bmatrix} (\cos\Omega\cos\omega - \sin\Omega\sin\omega\cos i) & (-\cos\Omega\sin\omega - \sin\Omega\cos\omega\cos i) \\ (\sin\Omega\cos\omega + \cos\Omega\sin\omega\cos i) & (-\sin\Omega\sin\omega + \cos\Omega\cos\omega\cos i) \\ \sin\omega\sin i & \cos\omega\sin i \end{bmatrix}$$

so that the planetary position vector is defined

$$R = A\begin{bmatrix} x_\omega \\ y_\omega \end{bmatrix}.$$

The planetary velocity is first evaluated in orbit plane coordinates as follows:

$$\dot{x}_\omega = \frac{-\sin f}{\sqrt{a(1-e^2)}}$$

$$\dot{y}_\omega = \frac{e+\cos f}{\sqrt{a(1-e^2)}}$$

and then in ecliptic coordinates:

$$\dot{R} = A \begin{bmatrix} \dot{x}_\omega \\ \dot{y}_\omega \end{bmatrix} .$$

The planetary acceleration is computed approximately as:

$$\ddot{R} = - \left[(1 + \mu/\mu_s)/(x_\omega^2 + y_\omega^2)^{3/2} \right] R,$$

where μ_s is the sun's gravitational constant ($=1.32715445 \times 10^{20}$ m^3/sec^2).

PLANETARY GRAVITATIONAL AND ORBITAL CONSTANTS

	Mercury	Venus	Earth	Mars	Jupiter
μ (m^3/sec^2)	2.17562D13	3.248534D14	3.986032D14	4.29778D13	1.267069D17
a (AU)	.3870986	.7233316	1.00000023	1.5236915	5.202561
e_1	.20561421	6.82069D-3	1.675104D-2	9.33129D-2	4.833475D-2
e_2 (1/cent.)	2.046D-5	-4.774D-5	-4.18D-5	9.2064D-5	1.6418D-4
e_3 (1/cent.^{2})	-3.D-8	9.1D-8	-1.26D-7	-7.7D-8	-4.676D-7
i_1 (rad)	.12222565	5.9230124D-2	0	3.2293876D-2	2.2841754D-2
i_2 (rad/cent.)	3.033964D-5	2.1855401D-5	0	-1.1327672D-5	-9.9415893D-5
i_3 (rad/cent.^{2})	-2.7149566D-7	-7.5630934D-8	0	4.5814893D-7	6.7873915D-8
Ω_1 (rad)	.82283029	1.3227513	0	.85148815	1.7356145
Ω_2 (rad/cent.)	2.0677958D-2	1.5952794D-2	0	1.3560239D-2	1.7637076D-2
Ω_3 (rad/cent.^{2})	4.0481942D-6	7.3109903D-6	0	-1.0520457D-5	6.1474375D-6
Π_1 (rad)	1.3246548	2.2713807	1.7666368	5.8332094	.22202207
Π_2 (rad/cent.)	2.7113157D-2	2.3951105D-2	3.0005264D-2	3.2120943D-2	2.8099132D-2
Π_3 (rad/cent.^{2})	5.38628D-6	2.8749451D-5	7.902463D-6	5.8628518D-6	1.8435428D-5
θ_1 (rad)	2.36146727	3.83718291	1.74446001	2.36878283	.90102455
θ_2 (rad/cent.)	2608.8147	1021.35293	628.331958	334.085683	52.9934743
θ_3 (rad/cent.^{2})	5.2552D-6	5.4048D-6	5.279621D-6	5.4222D-6	5.8413262D-6

PLANETARY GRAVITATIONAL AND ORBITAL CONSTANTS (continued)

	Saturn	Uranus	Neptune	Pluto
μ (m^3/sec^2)	3.791794D16	5.786726D15	6.876309D15	3.317819D14
a (AU)	9.554747	19.21814	30.10957	39.537355
e_1	5.589232D-2	4.6344D-2	8.99704D-3	.2515024
e_2 (1/cent.)	-3.455D-4	-2.658D-5	6.33D-6	0
e_3 (1/cent.^{2})	-7.28D-7	7.7D-8	-2.D-9	0
i_1 (rad)	4.3502671D-2	1.3482038D-2	3.1053625D-2	.29959349
i_2 (rad/cent.)	-6.8397514D-5	1.0913156D-5	-1.6656744D-4	0
i_3 (rad/cent.^{2})	-2.7033211D-7	6.8940505D-7	-1.5901889D-7	0
Ω_1 (rad)	1.9685636	1.282417	2.28082	1.9174073
Ω_2 (rad/cent.)	1.524013D-2	8.7033946D-3	1.9180034D-2	0
Ω_3 (rad/cent.^{2})	-2.6560518D-6	2.2892902D-5	4.360996D-6	0
Π_1 (rad)	1.5899628	2.9940888	.81554567	3.9056323
Π_2 (rad/cent.)	3.4180804D-2	2.590824D-2	2.4863514D-2	0
Π_3 (rad/cent.^{2})	1.4422722D-5	4.139824D-6	6.8210376D-6	0
θ_1 (rad)	5.96619116	2.85546276	3.96857273	3.2780198
θ_2 (rad/cent.)	21.3542815	7.50254138	3.837734	2.5273709
θ_3 (rad/cent.^{2})	5.6643206D-6	5.5159192D-6	5.5934894D-6	0

Messages and Printouts: If the Newton's iteration for eccentric anomaly fails, the following message is printed:

KEPLERS EQUATION ITERATION FAILS IN SUBROUTINE EFM.
EI = __________, E1 = __________, DELTA = __________

where E1 and EI are the values of eccentric anomaly, in radians, on the 99th and 100th iterations, and DELTA is the error, i.e., the absolute value of their difference. A looser tolerance is used when N = 20, corresponding to Comet Halley, the orbit of which is highly eccentric and energetic, often causing the tight iteration to fail.

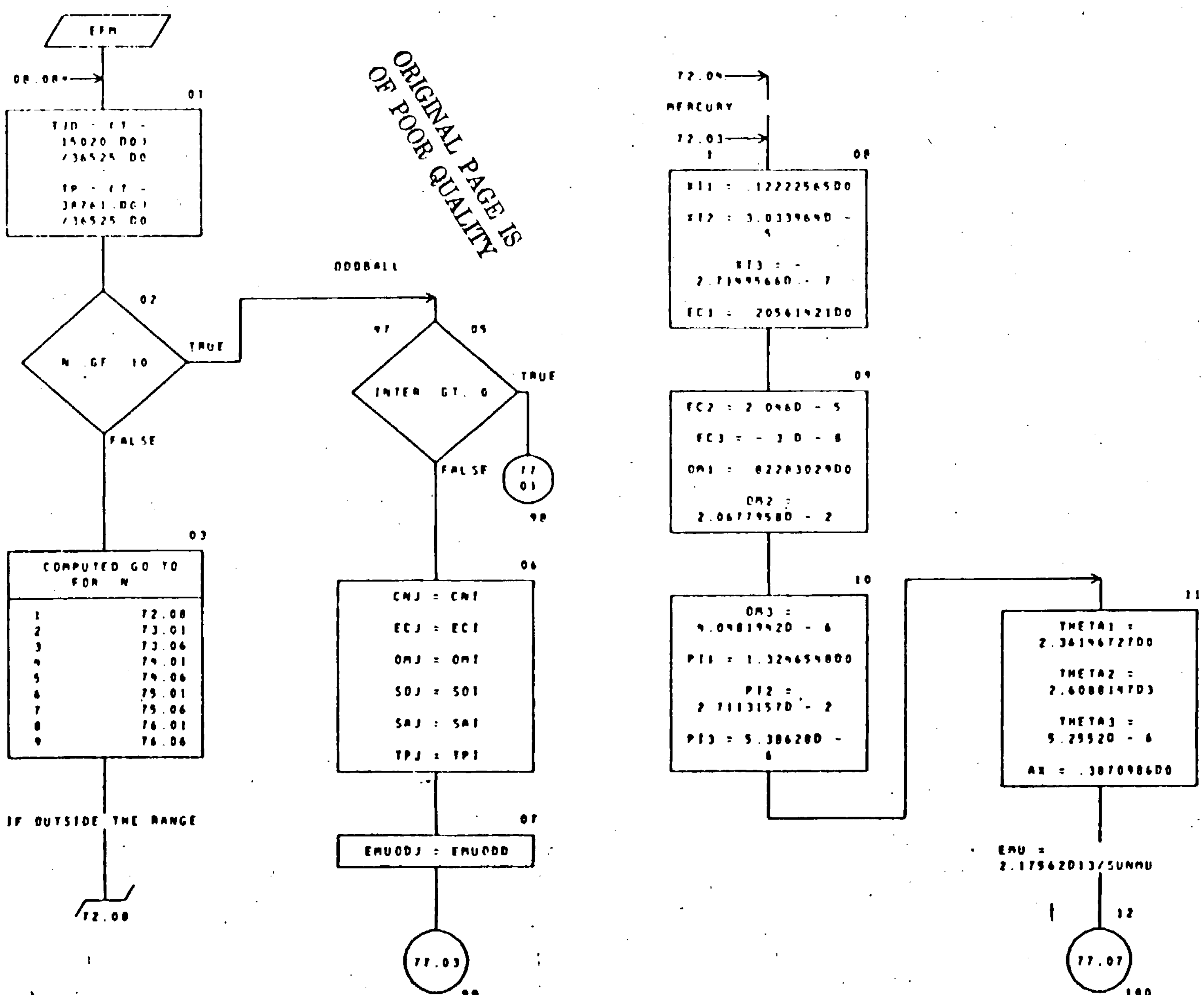
EFM EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
N	UX		Planet identification number.
T	UX		Time, in days, measured from the input reference date.
X(3)	SX		Ecliptic position vector, R, of planet, in AU.
GM(50)	U	REAL8	Array of planetary gravitational constants, μ, in m^3/sec^2.
XD(3)	SUX		Ecliptic velocity vector, $\dot{R}$, of planet, in units of Earth mean orbital speed.
CNI	U	REAL8	Ecliptic inclination i of Oddball's orbit, in radians.
DEG	U	REAL8	Radians to degrees conversion factor.
ECI	U	REAL8	Eccentricity e of Oddball's orbit.
OMI	U	REAL8	Longitude of ascending node Ω of Oddball's orbit on ecliptic, in radians.

EFM EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
SAI	U	REAL8	Semi-major axis a of Oddball's orbit, in AU.
SOI	U	REAL8	Argument of perihelion ω of Oddball's orbit, in radians.
TPI	U	REAL8	Time from reference date to perihelion passage of Oddball, in days.
CNIX(5)	U	REAL8	Same definition as CNI, except pertains to an intermediate target.
ECIX(5)	U	REAL8	Same definition as ECI, except pertains to an intermediate target.
OMIX(5)	U	REAL8	Same definition as OMI, except pertains to an intermediate target.
SAIX(5)	U	REAL8	Same definition as SAI, except pertains to an intermediate target.
SOIX(5)	U	REAL8	Same definition as SOI, except pertains to an intermediate target.
TPIX(5)	U	REAL8	Same definition as TPI, except pertains to an intermediate target.
INTER	UX		Index which selects an intermediate target.
SUNMU	U	REAL8	Gravitational constant of the sun, μ_s, in meters^3/sec^2.
TWOPI	U	REAL8	2π.
EMUODD	U	REAL8	Gravitational constant of Oddball, in meters^3/sec^2.
EMUODX(5)	U	REAL8	Same definition as EMUODD, except pertains to an intermediate target.
TDATEX	U	REAL8	Reference Julian date (minus 2400000).

CHART TITLE - SUBROUTINE EFM(N,T,X,XD,INTER)



ORIGINAL PAGE IS OF POOR QUALITY

EFM-9

CHART TITLE - SUBROUTINE EFMIN,T,X,XD,INTER)

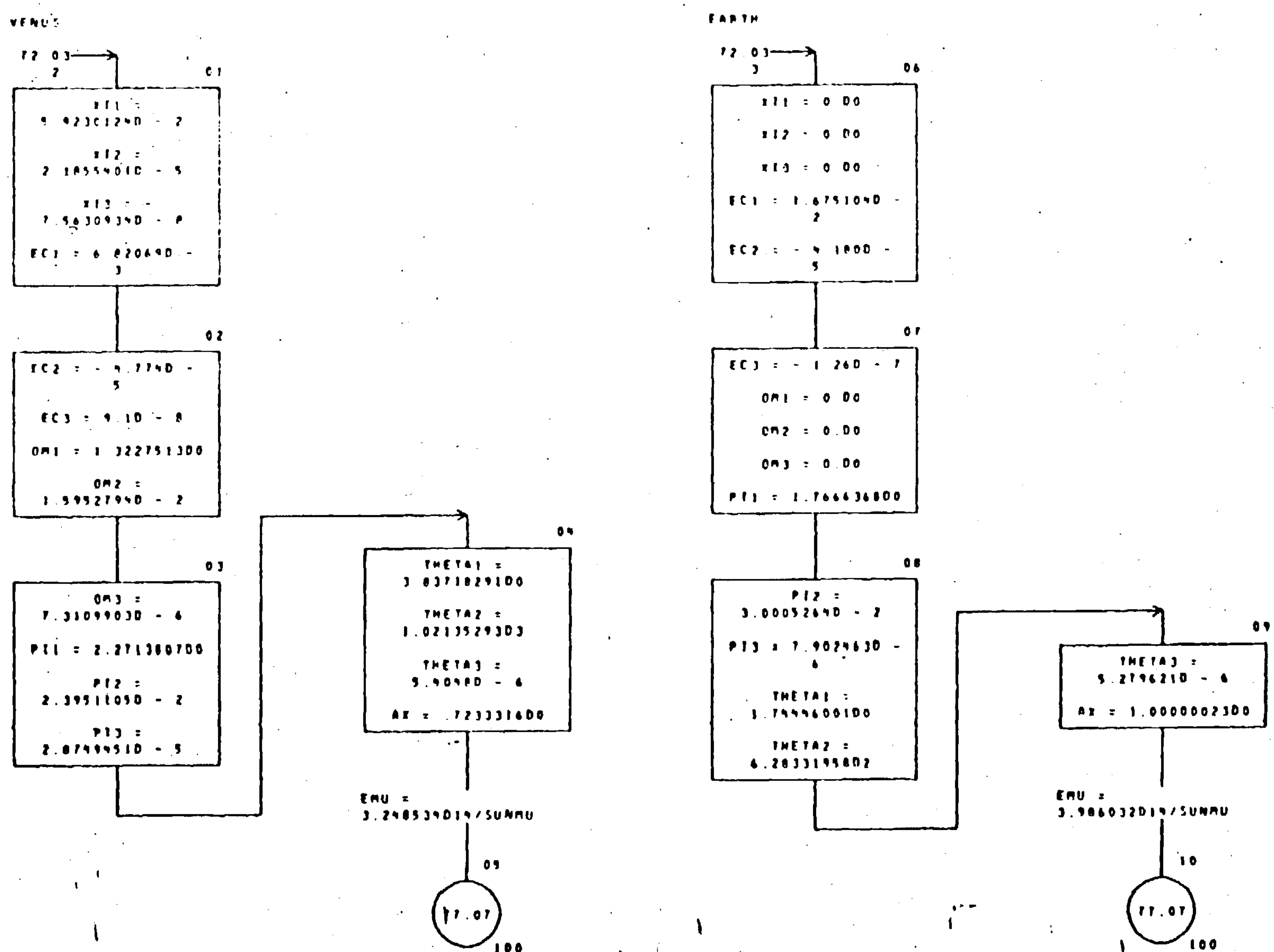


CHART TITLE - SUBROUTINE EFM(N,T,X,XD,INTER)

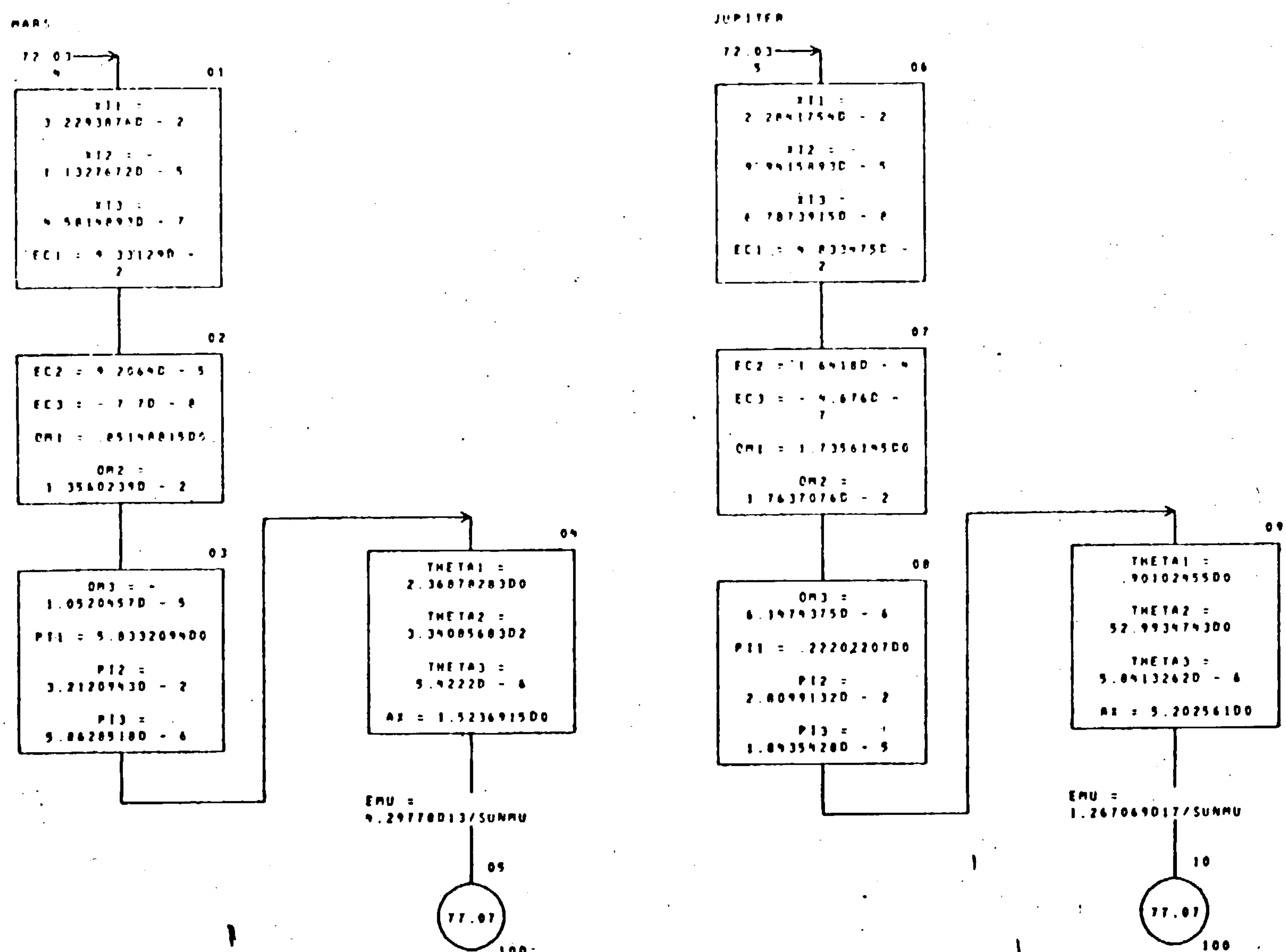


CHART TITLE - SUBROUTINE FFM(N,T,X,XD,INTFR)

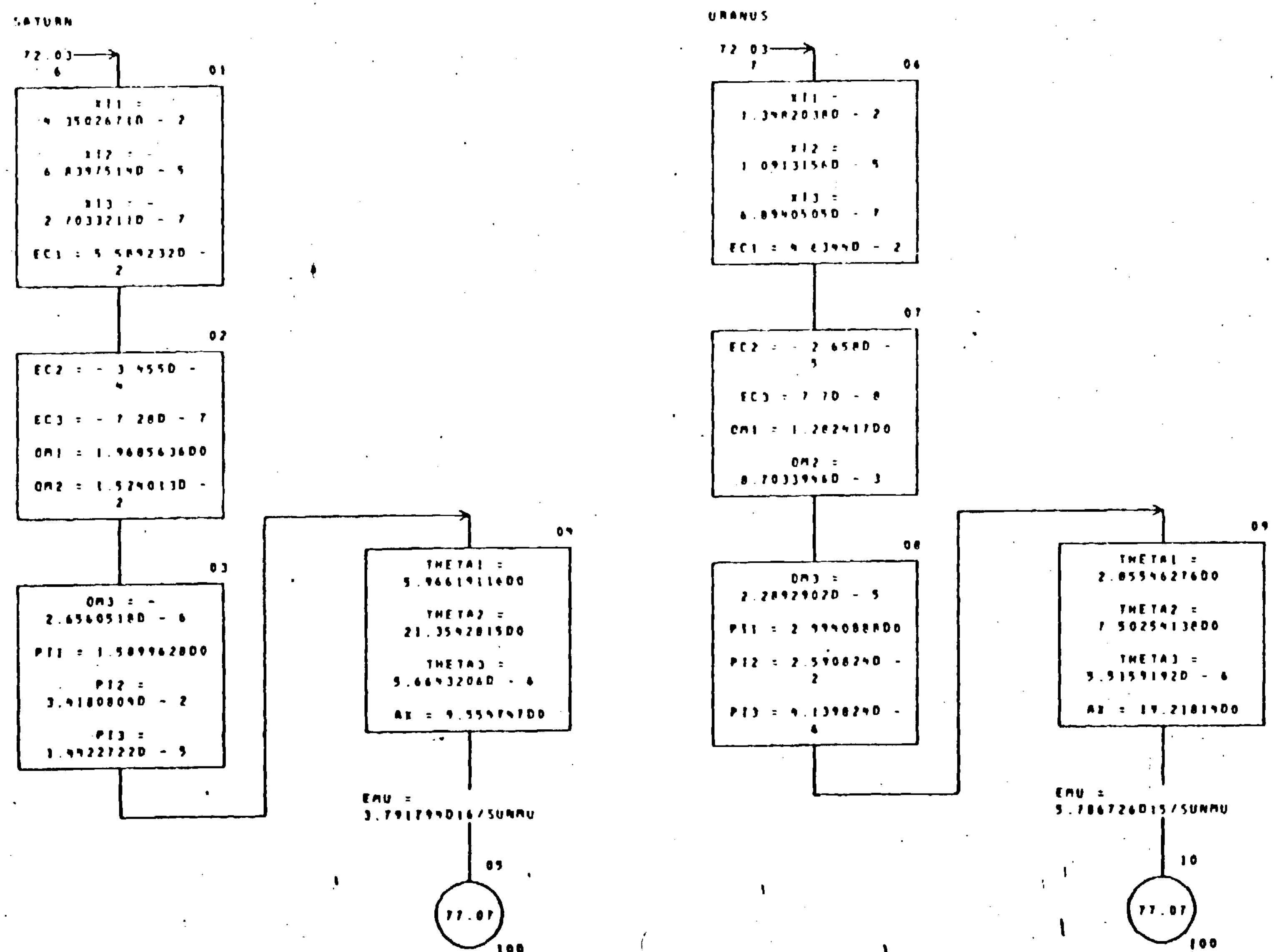
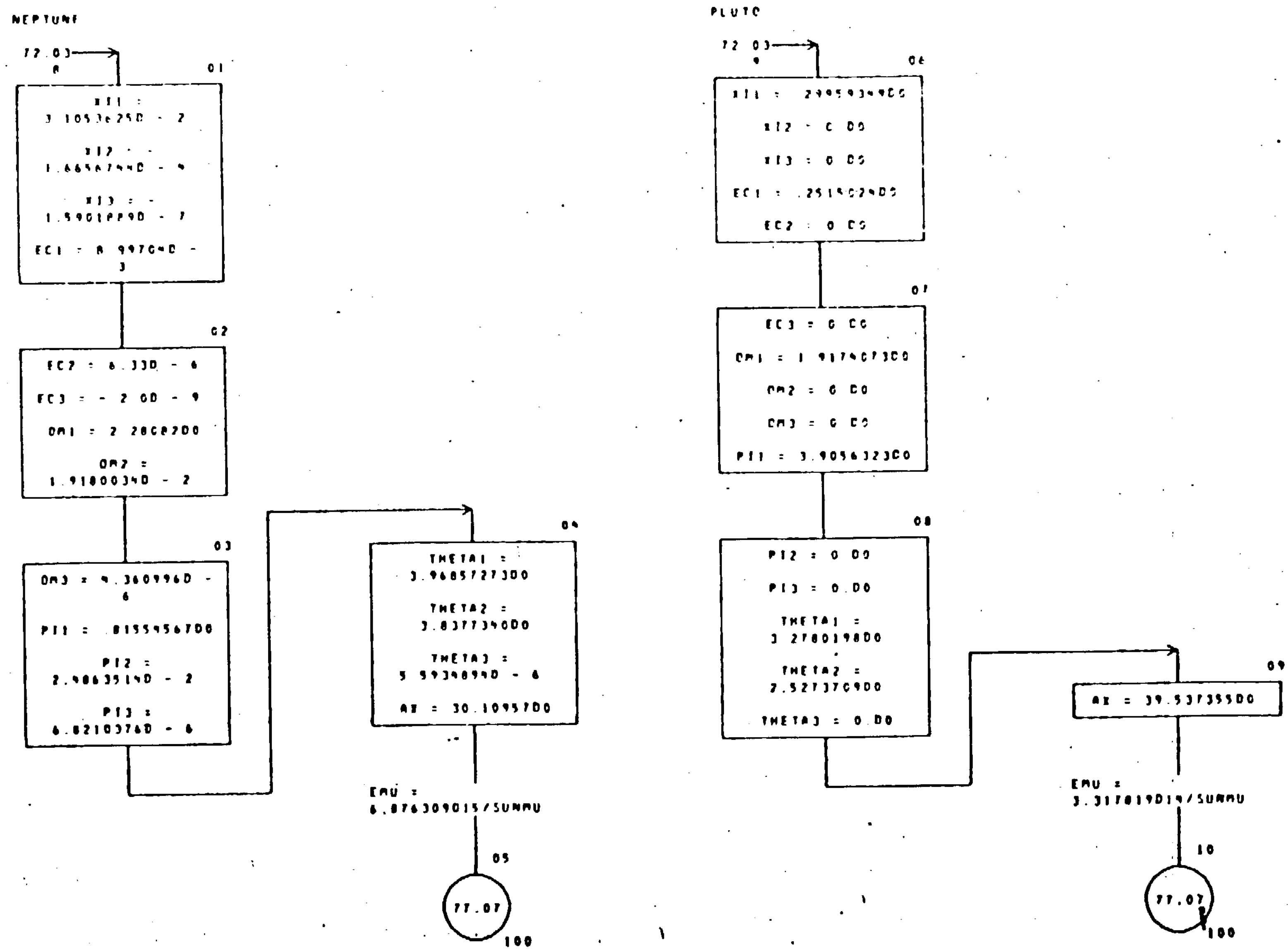


CHART TITLE - SUBROUTINE EFMIN,T,X,XD,INTER)



EFM-13

CHART TITLE - SUBROUTINE EFM(N,T,X,XD,INTER)

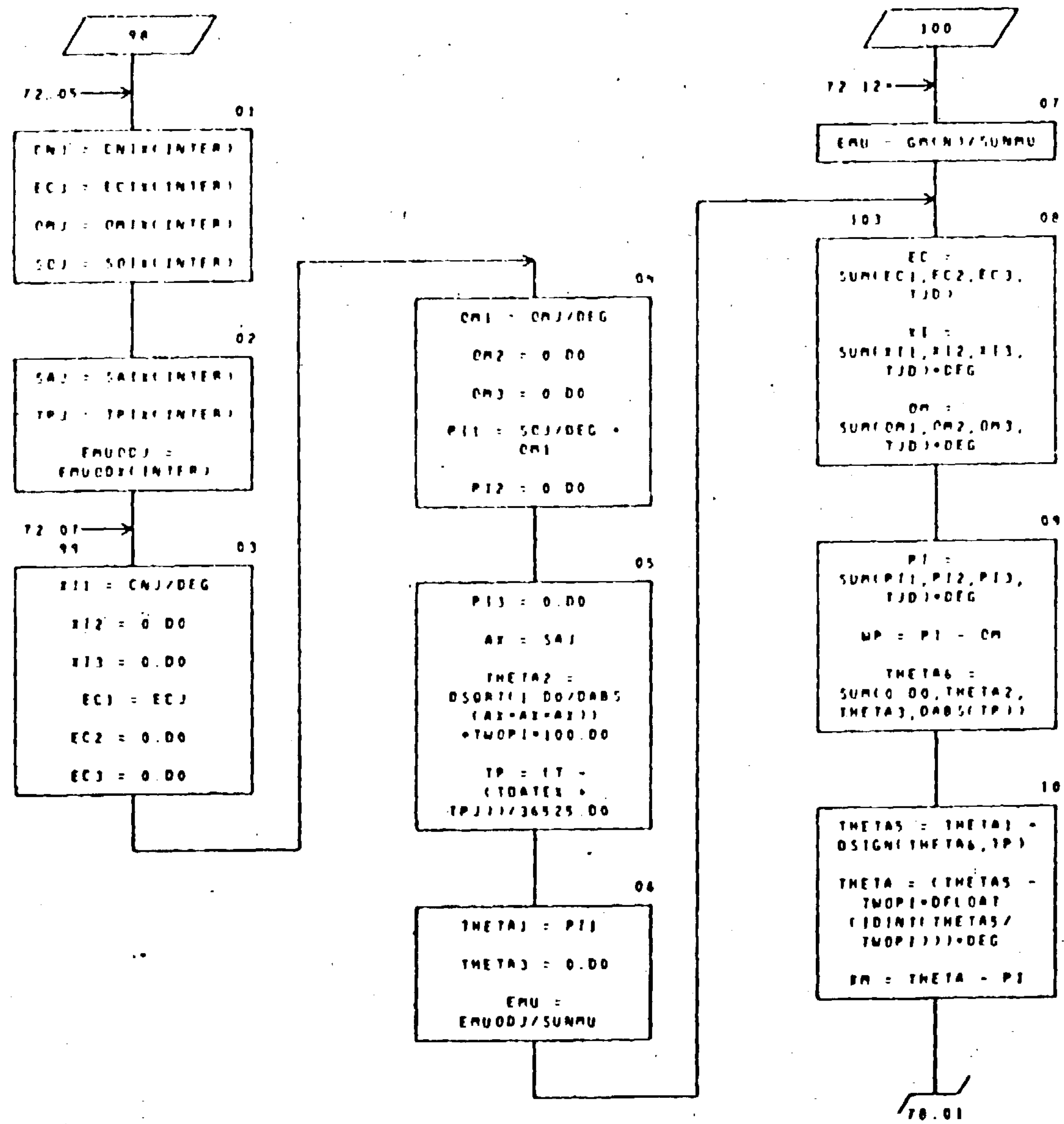


CHART TITLE - SUBROUTINE EFM(N,T,X,XD,INTER)

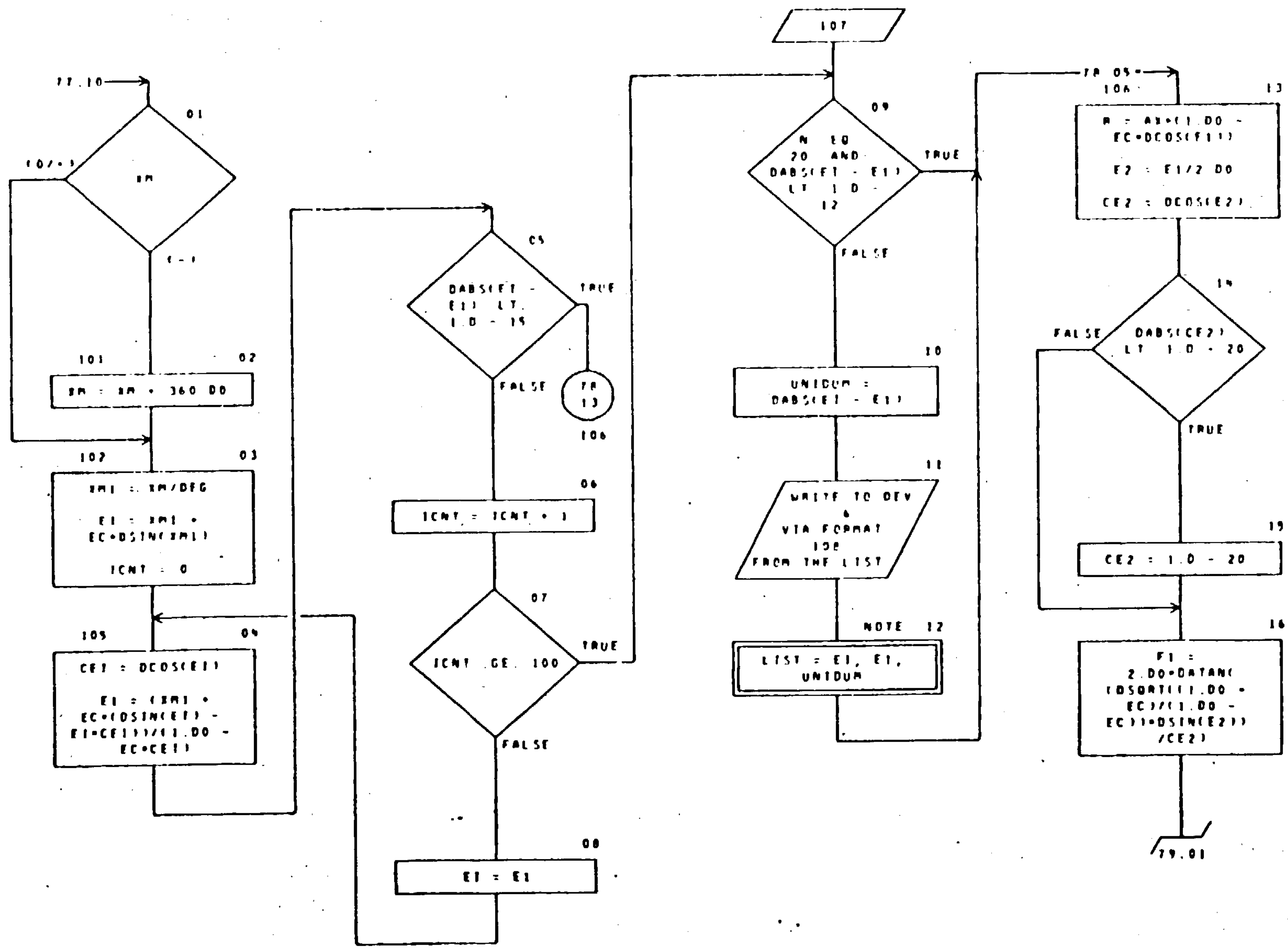


CHART TITLE - SUBROUTINE EFM(N,T,X,XD,INTER)

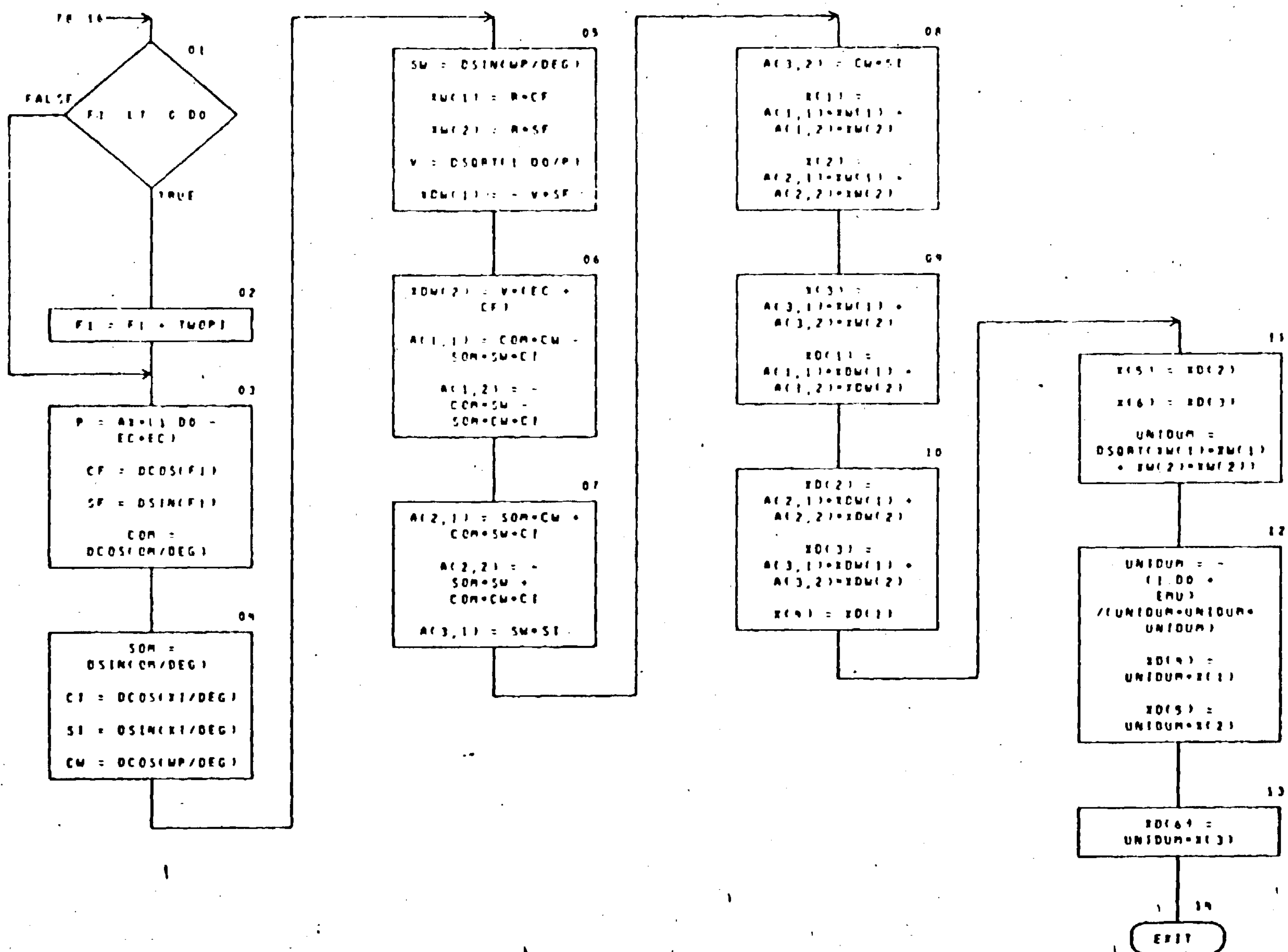


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION X(6),XD(6)          ,XM(2),XDM(2),A(3,2)
      COMMON /REAL8/ RO1(77),SAI,ECI,CNI,OMI,SCI,TPI,EMUCDD,RO2(73),
     XDATEX,RO3(67),SATX(5),ECIX(5),CNIX(5),OMIX(5),SCIX(5),TPIX(5),
     EMUCDX(5),RO4(52),
                DEG,RON,SUNMU,RO5(5),TWOPI,RO6(1679)
      COMMON /SOLSYS/ GM(70),SOLC1(210)
      STATEMENT FUNCTION DEFINITION,  SUM(X9,Y9,Z9,W9) = X9*W9*(Y9+Z9*W9)
108   FORMAT(1H ,51HKEPLERS EQUATION ITERATION FAILS IN SUBROUTINE EFM ,
     3HHE1 =1PD16.8,3HHE1 =D16.8,3HTHDELTA =D12.4)
```

ORIGINAL PAGE IS OF POOR QUALITY

EFM-17

<u>Name:</u> EFMPRT

<u>Calling Argument:</u> N, TIME, X, XSC, ANGLE, STAY

<u>Referenced Sub-programs:</u> PDATE

<u>Referenced Commons:</u> INTGR4, REAL8, SOLSYS

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> MORE, QPRINT

<u>Discussion:</u> EFMPRT is a contraction of "ephemeris print", and consists of the portion of printout on unit 6 beginning with the words MISSION SCHEDULE and followed by target and spacecraft ephemeris data enclosed within dashed lines. The contents of a single set of ephemeris data (enclosed within dashed lines), which is generated by a single call to EFMPRT, is described in the next section.

The computations performed in this routine consist of the solar distances of spacecraft and target,

$$r = \sqrt{x^2 + y^2 + z^2},$$

$$r_p = \sqrt{x_p^2 + y_p^2 + z_p^2},$$

where subscript p refers to the target ("planet"); the ecliptic latitudes and longitudes of the same,

$$LAT_{s/c} = \tan^{-1}\left(z/\sqrt{x^2 + y^2}\right)$$

$$LAT_{targ} = \tan^{-1}\left(z_p/\sqrt{x_p^2 + y_p^2}\right)$$

$$LONG_{s/c} = \tan^{-1}(y/x)$$

$$LONG_{targ} = \tan^{-1}(y_p/x_p)$$

The two- body transfer angle between the launch planet, having position vector P_L, and the target, having position vector P, is computed from

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

where

$$\cos\theta = \frac{P_L \cdot P}{\sqrt{|P_L||P|}}$$

and

$$\sin\theta = \pm\frac{\sqrt{(P_L \times P)\cdot(P_L \times P)}}{\sqrt{|P_L||P|}}$$

in which $(\pm) = \text{sign}\left[(P_L \times P)\cdot(P_L \times \dot{P}_L)\right]$.

<u>Messages and printouts</u>: When there are two, or less, ephemeris data sets (defined by the dashed lines) to be printed, such as when simulating missions involving Earth launch and a single (primary) target, the output from EFMPRT will appear on the same page as, and beneath, the Extremum Points of Selected Functions table. Otherwise, when there are three or more ephemeris data sets to be printed, such as when there are intermediate targets present, the output will begin at the top of the next page, with the case identification:

CASE__________

The general heading is printed:

MISSION SCHEDULE

followed by sequential groups of information separated by dashed lines. Each group of information consists of the following:

<u>(month)</u> <u>(day)</u> <u>(year)</u> <u>(hour)</u> G. M. T.

____________ JULIAN DATE

in which the month, day, year, and hour (Greenwich mean time) and the corresponding Julian date of the trajectory-segment endpoint associated with the named target is displayed, followed by

(maneuver) (target name) (AT (speed) KM/SEC)

where "maneuver" is DEPART, ARRIVE, or PASS depending on whether the spacecraft-target relationship is departure (launch or departure-after-rendezvous), arrival (rendezvous or orbiter), or flyby; "target name" is the name of the target; and the remainder of the line appears only if "maneuver" is PASS, in which the flyby speed is printed.

This is followed by two lines of target (PLANET) and spacecraft (S/C) ephemeris data:

	X	Y	Z	XDOT	YDOT	ZDOT	RADIUS	LAT.	LONG.
PLANET	___	___	___	___	___	___	___	___	___
S/C	___	___	___	___	___	___	___	___	___

in which X, Y, Z is the position in AU; XDOT, YDOT, ZDOT is the velocity in EMOS, these two vectors being expressed in ecliptic coordinates of date; RADIUS is the magnitude of the position vector in AU; LAT. and LONG. are the ecliptic latitude and longitude of the position vector, in degrees.

Finally, the two-body transfer angle θ between the original departure planet (which is usually the launch planet, Earth) and the target in question is printed in degrees:

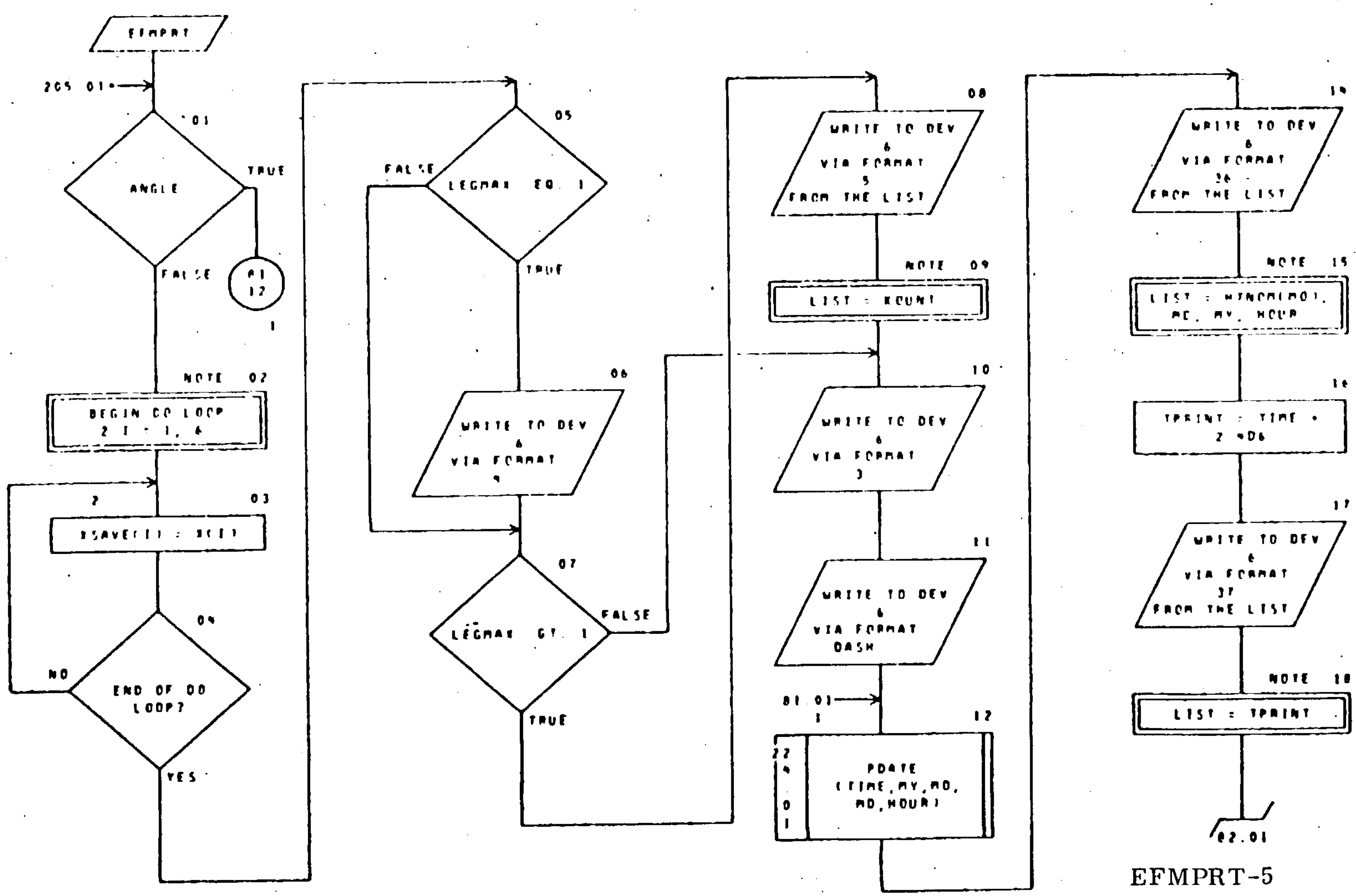
TWO-BODY TRANSFER ANGLE BETWEEN t_1 AND t_2 IS θ DEGREES.

where t_1 is the launch planet name and t_2 is the name of the target.

EFMPRT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
N	UX		Planet selector.
X(6)	UX		Planet position P and velocity $\dot{P}$ at time of encounter, in AU and EMOS, respectively.
APL (2,70)	U	SOLSYS	Array of planet names.
DEG	U	REAL8	Radians to degrees conversion factor.
XSC(6)	UX		Spacecraft position R and velocity $\dot{R}$ at time of encounter, in AU and EMOS, respectively.
STAY	UX		Indicates rendezvous and stopover at an intermediate target.
TIME	UAX		Julian date (less 2400000) at time of target intercept.
ANGLE	UX		Indicator for computation of two-body transfer angle.
CONSP	U	REAL8	Speed conversion factor, EMOS to meters/second.
KOUNT	U	INTGR4	Case counter.
MOPT2	U	INTGR4	Launch planet selector.
LEGMAX	U	INTGR4	Total number of legs (trajectory segments) between start of trajectory and primary target; equals the number of targets up to and including the primary target.

CHART TITLE - SUBROUTINE EFMPRT(IN,TIME,B,BSC,ANGLE,STAY)



EFMPRT-5

CHART TITLE - SUBROUTINE EFMPRTCN,TIME,X,XSC,ANGLE,STAY)

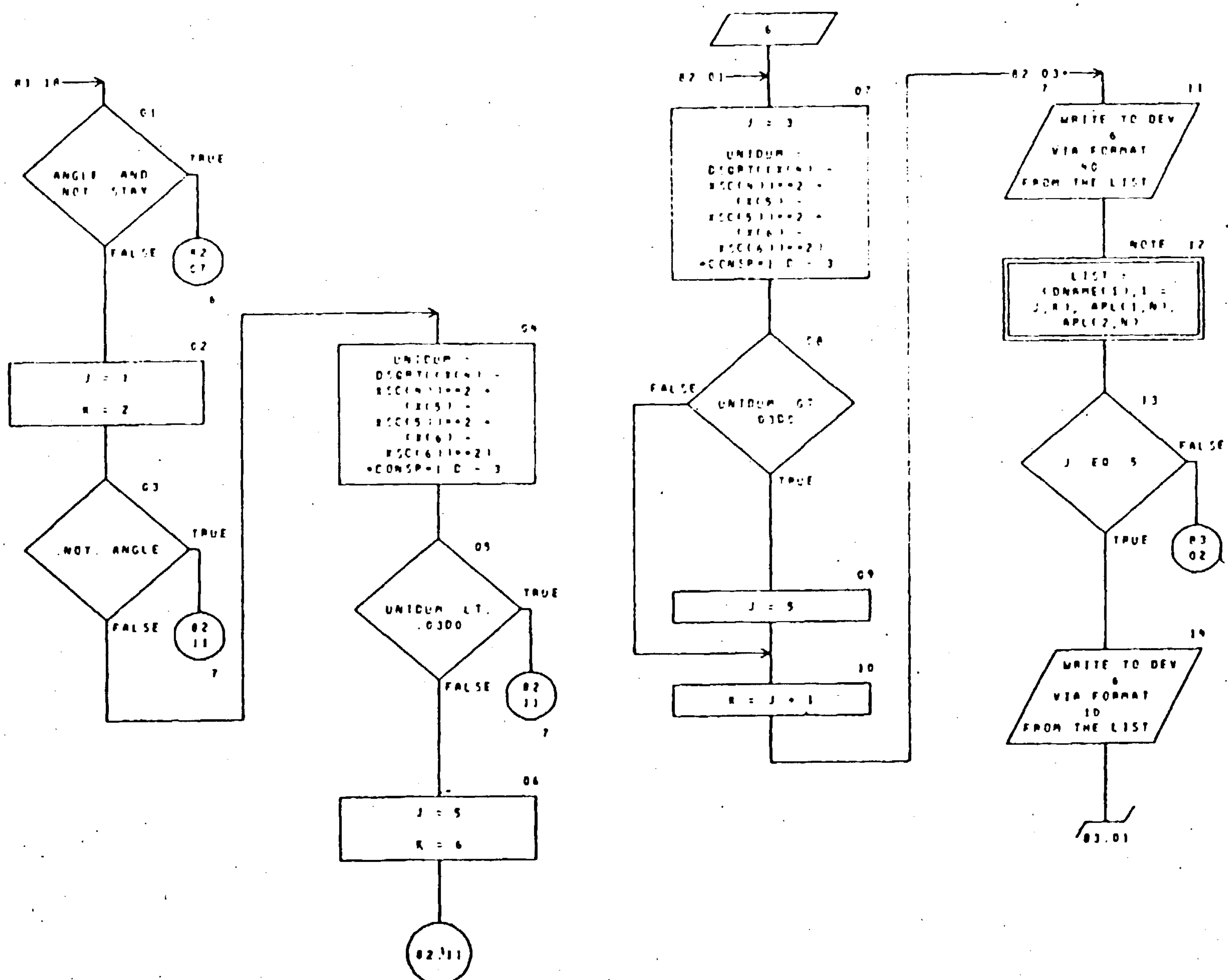
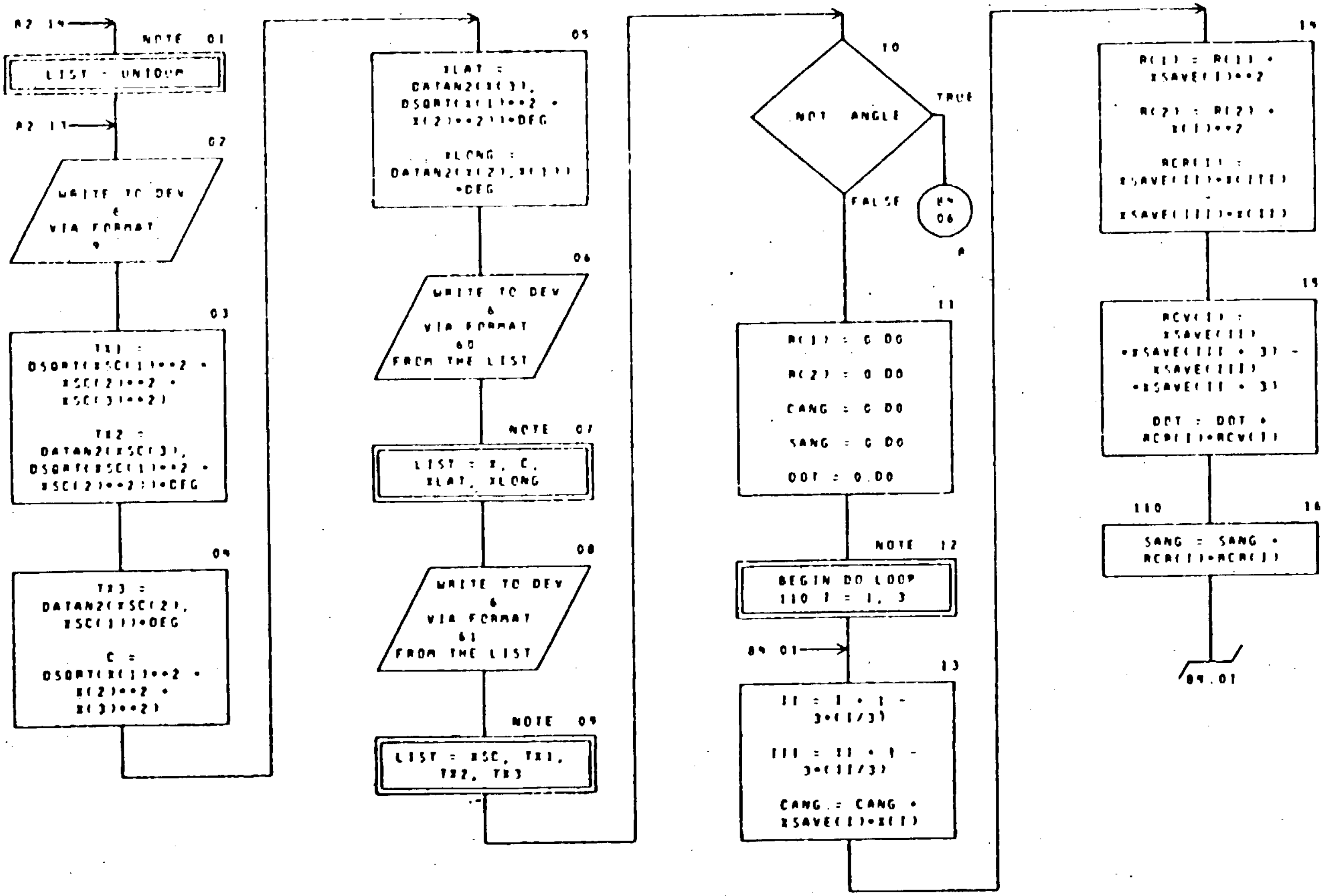


CHART TITLE - SUBROUTINE EFMPRT(N,TIME,X,XSC,ANGLE,STAV)



EFMPRT-7

CHART TITLE - SUBROUTINE EFMPRT(N,TIME,X,XSC,ANGLE,STAY)

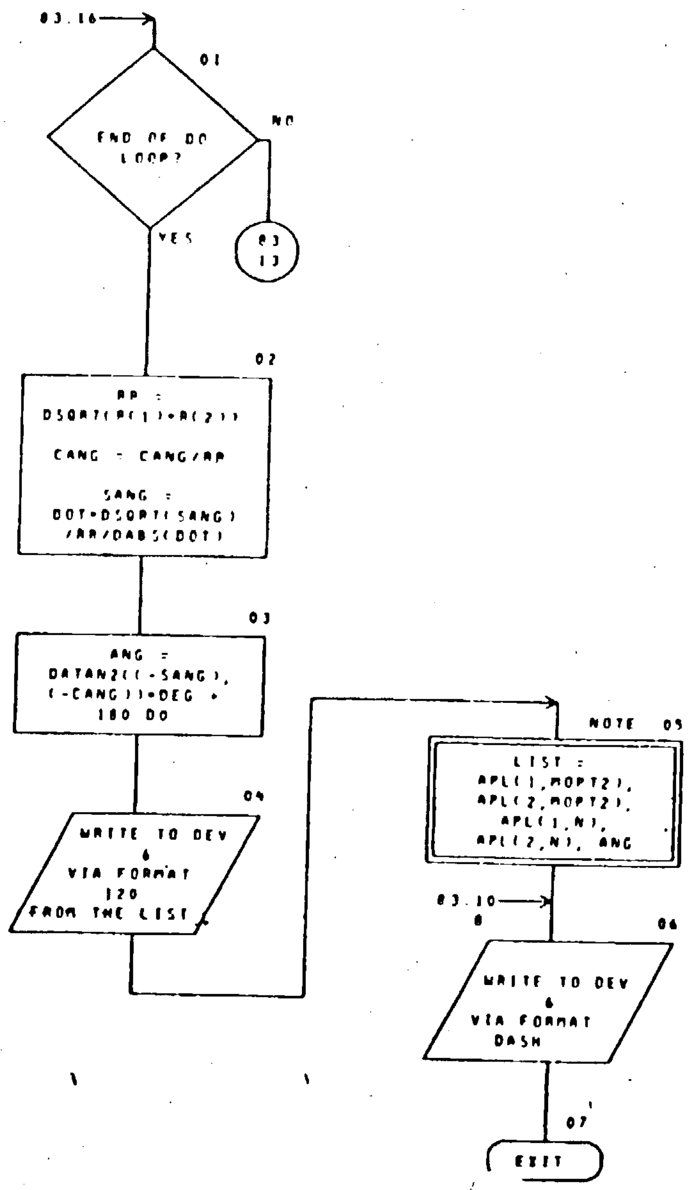


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
          IMPLICIT REAL*8 ( A-H, O-Z)
          LOGICAL ANGLE,STAY
          DIMENSION X(6),DNAME(6),R(2),RCR(3),RCV(3),MTNCM(12),XSC(6)
          ,XSAVE(6),DASH(10)
          COMMON /REAL8/ RO1(312),DEG,RO2(3),CONSP,RON(1683)
          COMMON /INTGR4/IG1(23),MOPT2,IC2(29),KOUNT,IO3(73),LEGMAX,ICN(872)
          COMMON /SOLSYS/ SOLO1(140),APL(2,70)
          DATA DNAME /8H   DEPAR,1HT,8HARRIVE A,1HT,8H      PAS,1HS/
          DATA DASH /8H(1HO130H,16*8H--------,3H--)/
          DATA MTNCM /8H JANUARY,8HFEBRUARY,8H   MARCH,8H   APRIL,8H      MAY
          ,8H    JUNE,8H    JULY,8H  AUGUST,8HSEPTEMBR,8H OCTOBER,8HNOVEMBER
          ,8HDECEMBER/
4         FORMAT(1HO/1H )
5         FORMAT(1H1,4HCASEI3)
3         FORMAT(1H ,53X16HMISSION SCHEDULE)
36        FORMAT(1HO,A8,I3,1H,,I5,1PD16.8,7H G M T.)
37        FORMAT(1H+,40H----------------------------------------
          /1H ,7PD20.3,12H JULIAN DATE)
40        FORMAT(41H+---------------------------------------- 10XA8,A2,2A8)
9         FORMAT(1HO,15X1HX,14X
           1HY, 14X, 1HZ, 13X, 4HXDOT, 11X, 4HYDOT, 11X, 4HZDOT, 10X,
           6HRADIUS, 7X, 4HLAT., 4X, 5HLONG. )
10        FORMAT(1H+,77X2HATF7.3,7H KM/SEC)
60        FORMAT(1HO,'PLANET ',1P7D15.7,0P2F9.3)
61        FORMAT(1H ,'   S/C ',1P7D15.7,0P2F9.3)
```


EFMPRT-9

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
120       FORMAT(33H0TWO-BODY TRANSFER ANGLE BETWEEN ,2A8,5HAND 2A8,3H IS,
           F10.4,9H DEGREES.)
```

Name: ETA

Calling Argument: VJ, ETAOP, DETAOP for ETA; BI, DI, EI for ETAINT

Referenced Sub-programs: None

Referenced Commons: REAL8

Entry Points: ETAINT

Referencing Sub-programs: TRAJ for ETA; QSTART for ETAINT

Discussion: This routine computes the total thruster subsystem efficiency,

$$\eta = \frac{b c^2}{c^2 + d^2} + e$$

and its derivative,

$$\frac{d\eta}{dc} = \frac{2\eta}{c}\left(1 - \frac{\eta}{b}\right)$$

where c is the jet exhaust speed and b, d, and e are coefficients which characterize the thruster hardware being simulated.

Entry point ETAINT performs initialization for subroutine ETA at the beginning of each case of the computer run, avoiding unnecessary repeated computations during an iteration sequence.

ETA EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
BI	UX		Efficiency coefficient, b.
DI	UX		Efficiency coefficient, d, in km/sec.
EI	UX		Efficiency coefficient, e.
VJ	UX		Electric propulsion jet exhaust speed, c, in EMOS.
CONSP	U	REAL8	Speed conversion factor, EMOS to meters/second.
ETAOP	SUX		Total thruster subsystem efficiency, η.
DETAOP	SX		Efficiency derivative, $d\eta/dc$, in $EMOS^{-1}$.

CHART TITLE - SUBROUTINE ETA(VJ,ETADP,DETADP)

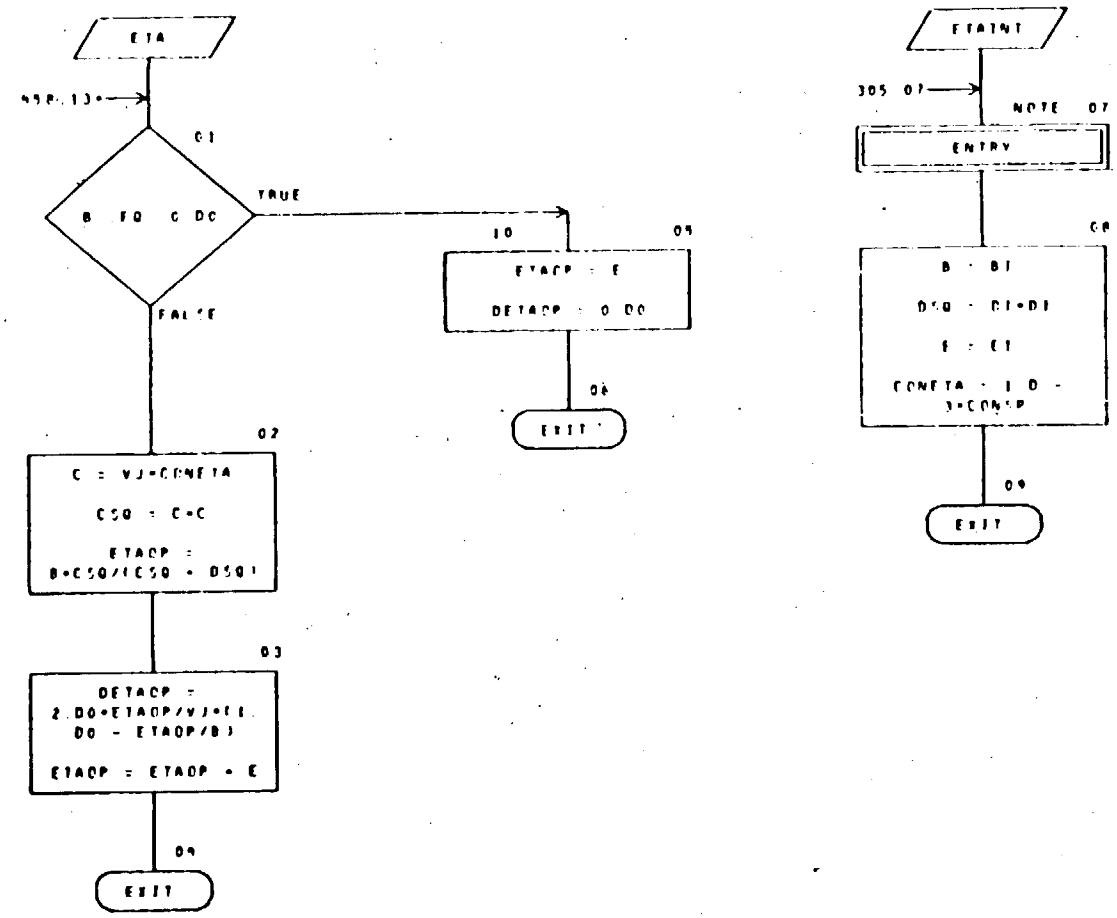


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
COMMON /REAL8/ R01(316),CONSP,R02(1683)
```

Name: EXTAB

Calling Argument: None

Referenced Sub-programs: None

Referenced Commons: EXTREM, INTGR4, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: FINISH, MORE

Discussion: This routine prints the "Extremum Points of Selected Functions" (on unit 6), and is the final subroutine in the sequence: CHECK → LOAD → STORE → EXTAB. The Extremum Table is limited to 100 entries (lines) per trajectory, to conserve storage space. This limit has never yet been exceeded, except in instances where erroneous trajectory starting conditions resulted in an extraordinary (and useless) trajectory.

Remarkable parameters on each line are dubbed with MAX or MIN to denote an extremum value, * to denote a special point (i.e., special solar distance), or ON or OFF to denote thrust switching events and also solar array orientation switching to and from the maximum power boundary.

Messages and printouts: The standard informative output of this subroutine is as follows:

CASE (n)

EXTREMUM POINTS OF SELECTED FUNCTIONS

I	TIME	ECLIPTIC LONGITUDE	SOLAR DISTANCE	COMMUNICATION ANGLE	DISTANCE	SWITCH FUNCTION ...
(i)	(t)	(θ_ϵ)	(r)	(θ_{com})	(r_{com})	(σ)
___	___	___	___	___	___	___
___	___	___	___	___	___	___
___	___	___	___	___	___	___

⋮

. . .	THRUST ANGLES			INPUT	ARRAY
	PSI	THETA	PHI	POWER	ANGLE
	(ψ)	(θ)	(ϕ)	(p)	(χ)
	___	___	___	___	___
	___	___	___	___	___
	___	___	___	___	___
	⋮				

where n is the case number, i is the number of iterations required in subroutine INTERP to isolate the remarkable point, t is the time elapsed since the beginning of the trajectory (in days), θ_ϵ is the elapsed ecliptic longitude (in degrees), r is the spacecraft solar distance (in AU), θ_{com} and r_{com} are the "communication" angle (in degrees) and distance (in AU) with respect to a reference body specified by program input quantity NDIST, σ is the thrust switching function, ψ, θ, and ϕ are the thrust angles (in degrees) identical to those output in the trajectory block print, p is the power input to the power conditioners (in kw), and χ is the solar array tilt angle (in degrees).

EXTAB EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
CHIX(2,100)	SU	EXTREM	Storage array for solar-panel array angle, χ, in degrees.
DIST(2,100)	U	EXTREM	Storage array for spacecraft solar distance, r, in AU.
MODE	U	INTGR4	Power variation option selector.
MPOW	U	INTGR4	Maximum or optimum power indicator during solar panel degradation option.
TIME (100)	U	EXTREM	Storage array for time elapsed since start of trajectory, t, in days.
TRAV (100)	U	EXTREM	Storage array for ecliptic longitude elapsed since start of trajectory, in degrees.

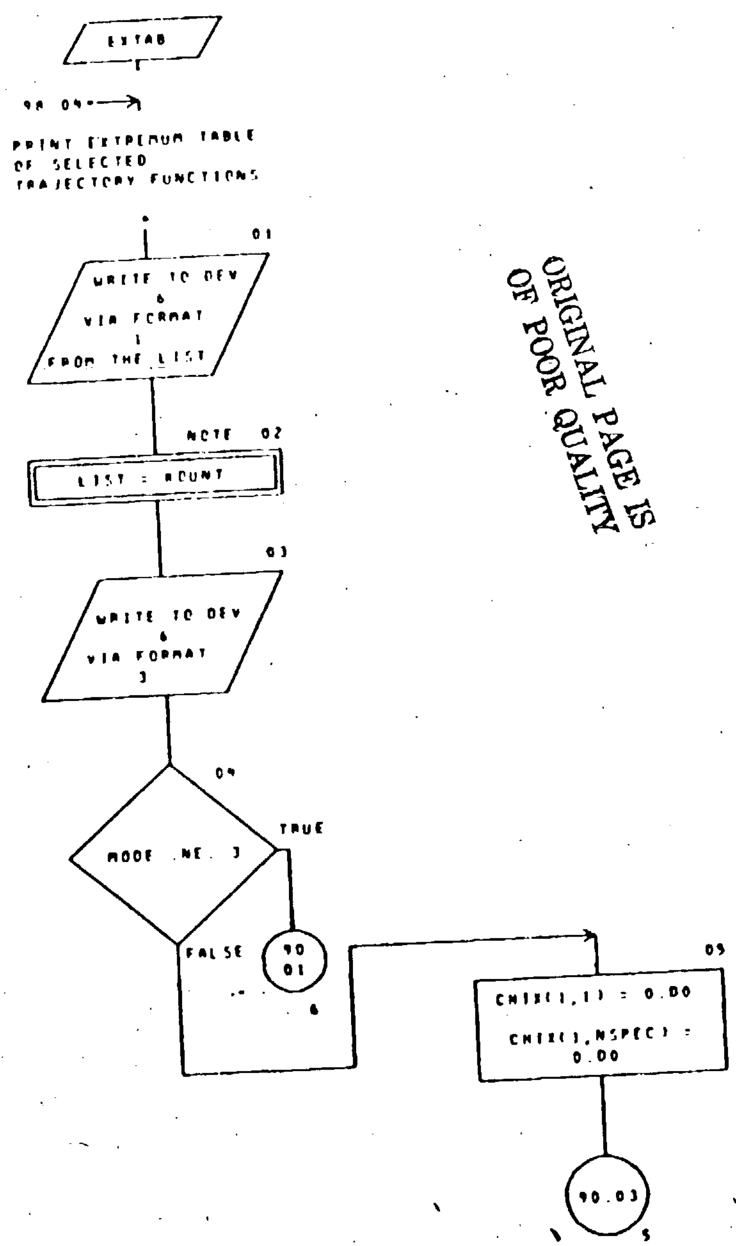
EXTAB EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ERODE	U	LOGIC4	Power degradation option indicator.
KOUNT	U	INTGR4	Case counter.
NSPEC	U	INTGR4	Master array index (and counter) for storage arrays; has maximum value in subroutine EXTAB, corresponding to trajectory endpoint.
ONOFF (2,100)	U	EXTREM	Storage array for thrust switching function, σ.
POWEX (2,100)	U	EXTREM	Storage array for power ratio, γ.
XMASS(7)	U	REAL8	Mass and related-parameter array; XMASS(6) is reference power, p_{ref}, in watts.
AKOUNT (100)	U	EXTREM	Storage array for the number of iterations required to isolate the associated point.
ANGCOM (2,100)	U	EXTREM	Storage array for communication angle, θ_{com}, in degrees.
ANGPHI (2,100)	U	EXTREM	Storage array for thrust angle ϕ, in degrees.
ANGPSI (2,100)	U	EXTREM	Storage array for thrust angle ψ, in degrees.
ANGTHE (2,100)	U	EXTREM	Storage array for thrust angle θ, in degrees.
COMANG	S	REAL8	Communication angle at time of primary-target intercept, in degrees.
COMDIS	S	REAL8	Communication distance at time of primary-target intercept, in AU.

EXTAB EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
DISCOM (2,100)	U	EXTREM	Storage array for communication distance, r_{com}, in AU.
RTSWIT	U	REAL8	Critical solar distance corresponding to a special point in the solar power curve, in AU.

CHART TITLE - SUBROUTINE EXTAB



ORIGINAL PAGE IS OF POOR QUALITY

EXTAB-5

CHART TITLE - SUBROUTINE EXTAB

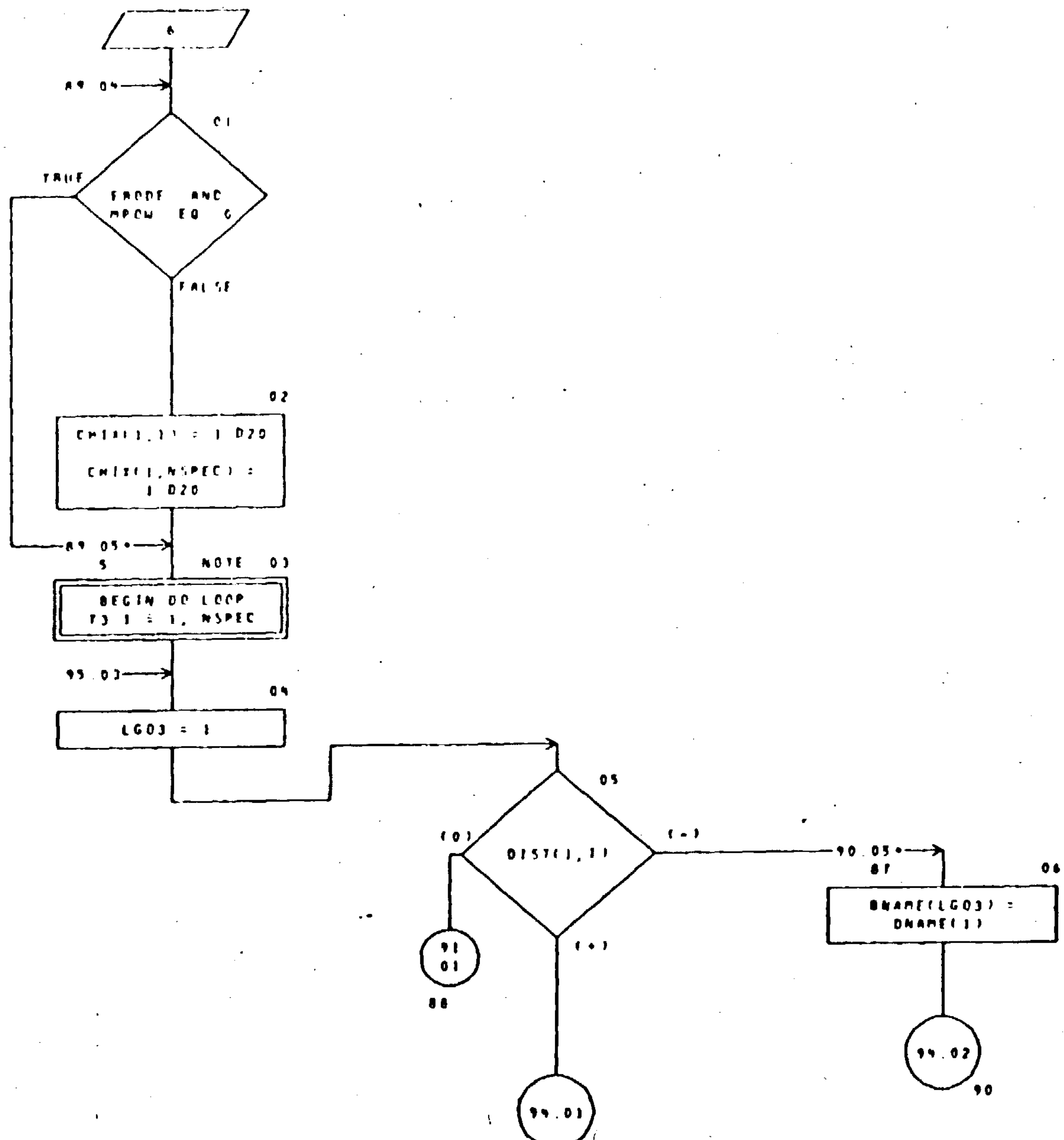
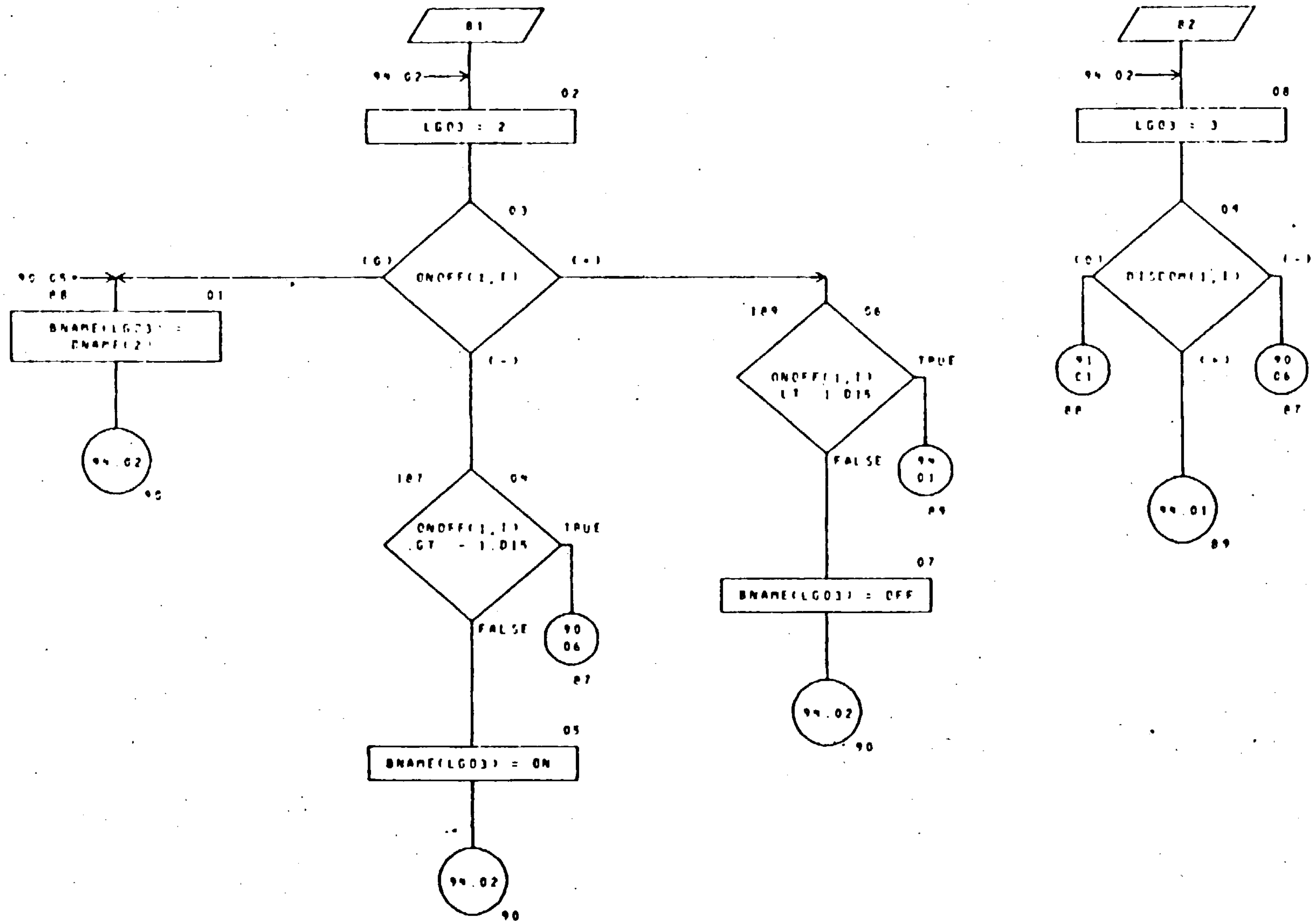


CHART TITLE - SUBROUTINE EXTAB



EXTAB-7

CHART TITLE - SUBROUTINE EXTAB

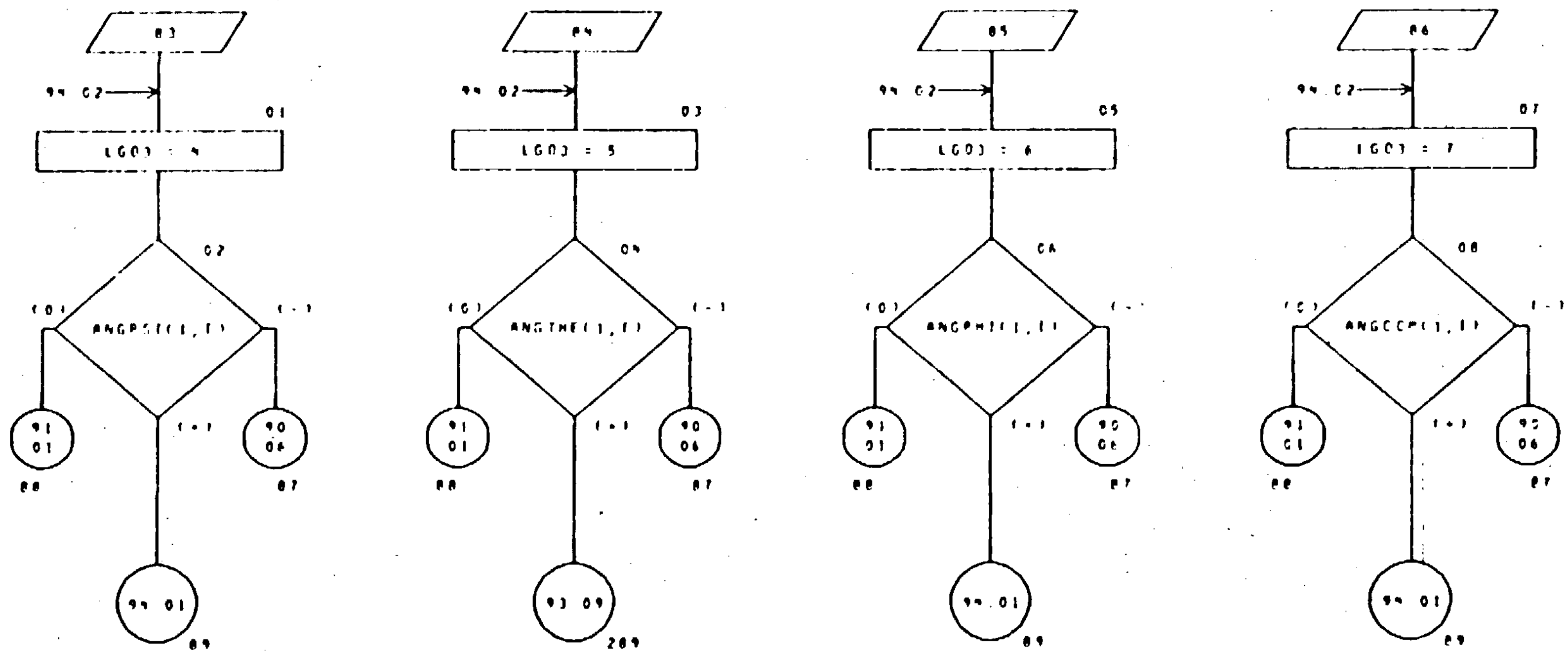


CHART TITLE - SUBROUTINE EXTAB

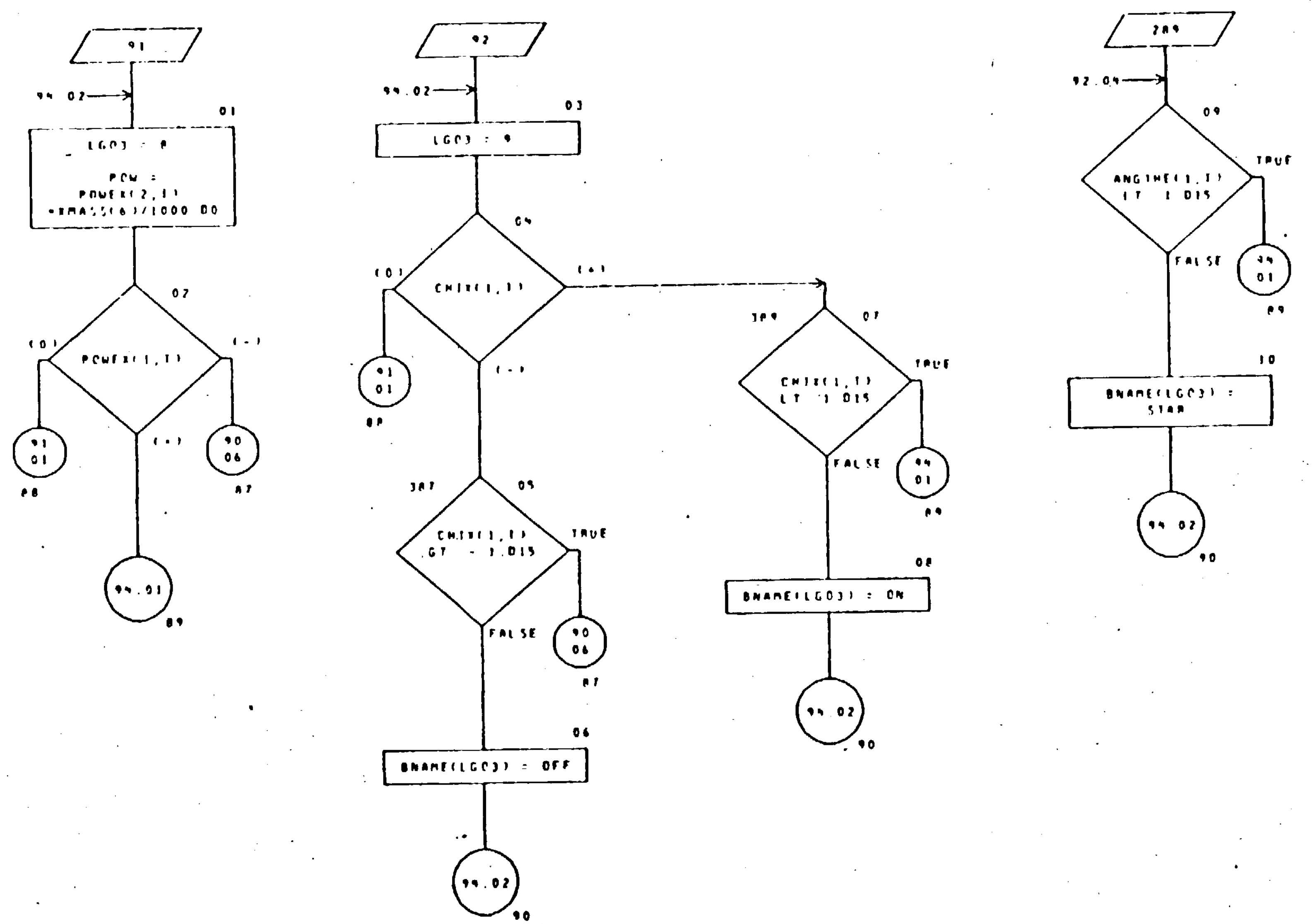


CHART TITLE - SUBROUTINE EXTAB

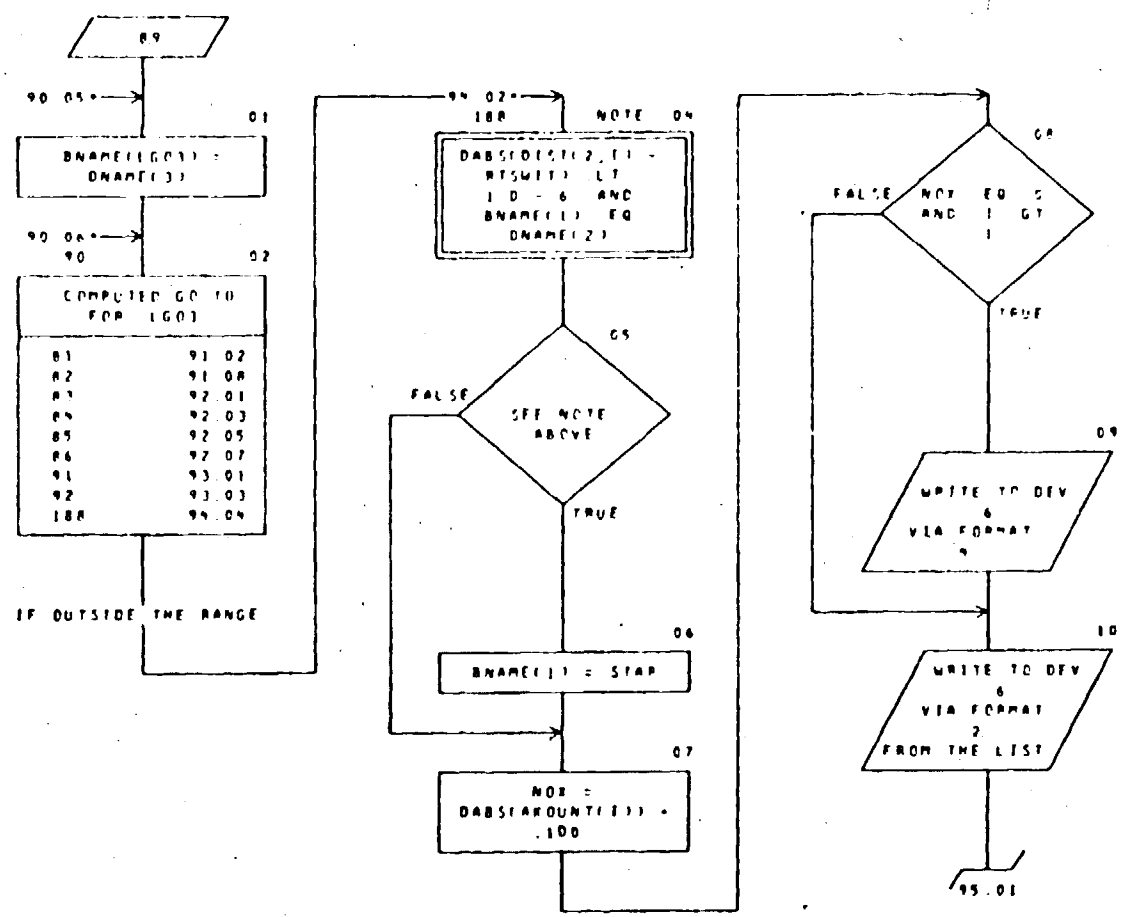


CHART TITLE - SUBROUTINE EXTAB

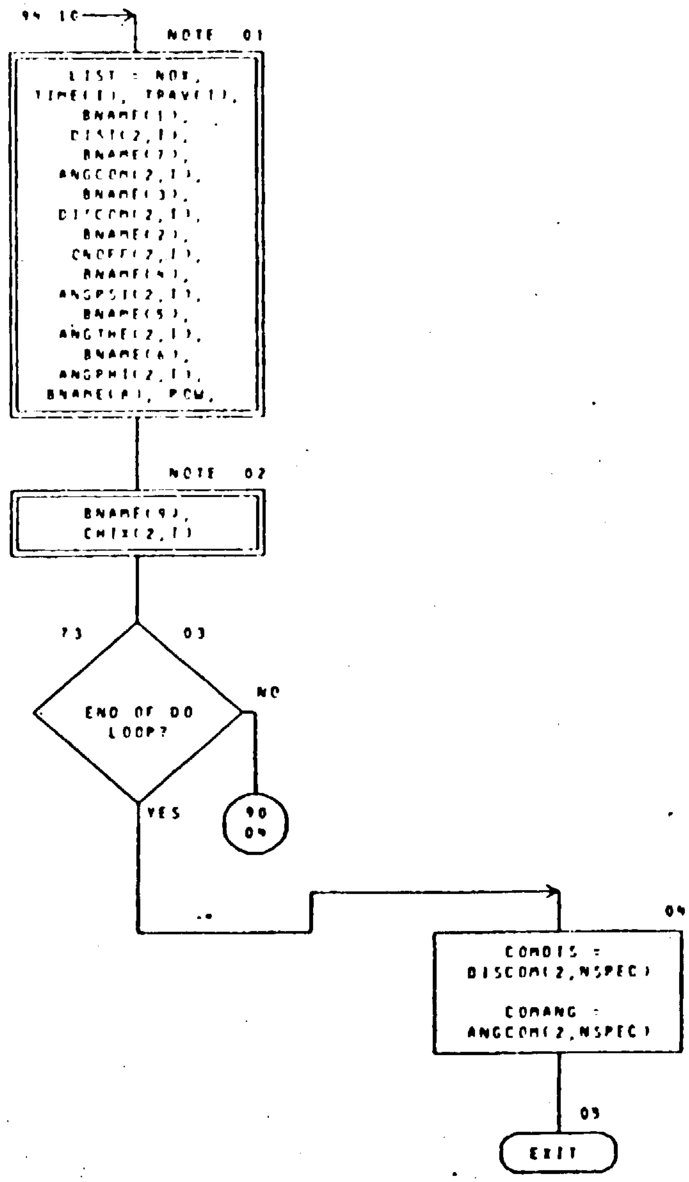


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL ERODE
      DIMENSION DNAME(3),BNAME(9)
      COMMON /REALA/R01,XMASS(7),R02(172),COMDIS,COMANG,RISWIT,R03(181)
      COMMON /INTGRS/ I01,MODE,I02(35),NSPEC,I03(15),KOUNT,I04(85),
      MPOW,I05(860)
      COMMON /LOGICS/ L01(19),ERODE,L02(480)
      COMMON /EXTREM/ TIME(100),TRAV(100),DIST(2,100),ONOFF(2,100),
      DISCOM(2,100),ANGCOM(2,100),ANGPSI(2,100),ANGTHE(2,100),
      ANGPHI(2,100),POWER(2,100),CHIX(2,100),AKOUNT(100),E01(620)
      DATA ON,OFF,DNAME /5H   ON,5H  OFF,5H  MIN,5H     ,5H  MAX/
      DATA STAR /5H    */
1     FORMAT(1H1,9HCASE13,
                 30X37HEXTREMUM POINTS OF SELECTED FUNCTIONS/1H )
3     FORMAT(2H  20H1    TIME    ECLIPTIC,5X5HSOLAR,6X13HCOMMUNICATION,
      10X6HSWITCH,14X13HTHRUST ANGLES,14X5HINPUT,6X5HARRAY/1H ,
      12X9HLONGITUDE,2X8HDISTANCE,5X16HANGLE    DISTANCE,
       7X8HFUNCTION,8X3HPSI,7X5HTHETA,8X3HPHI,6X5HPOWER,6X5HANGLE)
4     FORMAT(1H )
2     FORMAT(1H ,I2,F9.3,F7.1,A6,F6.3,A6,F5.1,A6,F5.2,A6,1PD9.2,
      A6,0PF5.1,A6,F6.1,A6,F5.1,A6,F5.1,A6,F5.1)
```

Name: FINISH

Calling Argument: None

Referenced Sub-programs: CORNER, EXTAB, PRINT, PRINTR, PUNCH, QPRINT, SUMMRY, TRAJ

Referenced Commons: INTGR4, ITERAT, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: MAIN, TIKTOK

Discussion: This routine "finishes" each case of the computer run. It executes the so-called Summary Trajectory (QJEX = .TRUE. : CALL TRAJ), in which detailed trajectory information is output. It updates the iterator independent-variable values, if desired. It outputs these values, and also the dependent-variable values, on units 6 (PRINTR) and 11 (PRINT); it prints the Extremum Table (EXTAB) and Summary Page (QPRINT(5)) on unit 6, stores selected case information (SUMMRY), provides control when the trajectory is in high proximity to a propulsion-time corner (CORNER), and optionally punches onto cards or writes onto magnetic tape selected trajectory and performance information (PUNCH).

FINISH EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
O(70)	U	ITERAT	Array of iterator independent variables, in internal units.
BX(5,70)	S	ITERAT	Iterator independent-variable value and control array.
OO(70)	SU	ITERAT	Array of iterator independent variables, in external units.
GAP	U	REAL8	Propulsion-corner proximity tolerance-interval, $\Delta\sigma$.

FINISH EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
NPR(4)	U	INTGR4	Printout amount control array.
CONX(70)	U	ITERAT	Array of iterator independent-variable conversion factors, initialized in subroutine BEGIN.
HUNG	U	LOGIC4	Indicator for the condition in which the trajectory is in high proximity to a propulsion-time corner; "hung on a thrust phase or a coast phase".
QJEX	S	LOGIC4	Detailed printout indicator.
ERROR	UA	LOGIC4	Program master error indicator.
EXTRA	S	LOGIC4	Indicator for extra printout at each computation step (trajectory block print).
JHUNG	U	INTGR4	Controls generation of extra cases when condition indicated by HUNG is applicable.
KOUNT	A	INTGR4	Case counter.
TRACK	S	LOGIC4	Indicator for trajectory long block printout (at each computation step).
CONVRG	U	LOGIC4	Iteration sequence convergence indicator.
MPRINT	U	INTGR4	Indicator for printout amount at each computation step.
MPUNCH	U	INTGR4	Indicator controlling card punching and magnetic tape generation.
MUPDAT	U	INTGR4	Indicator for updating iterator independent-variable values.
PLANET	U	LOGIC4	Ephemeris-option indicator.

CHART TITLE - SUBROUTINE FINISH

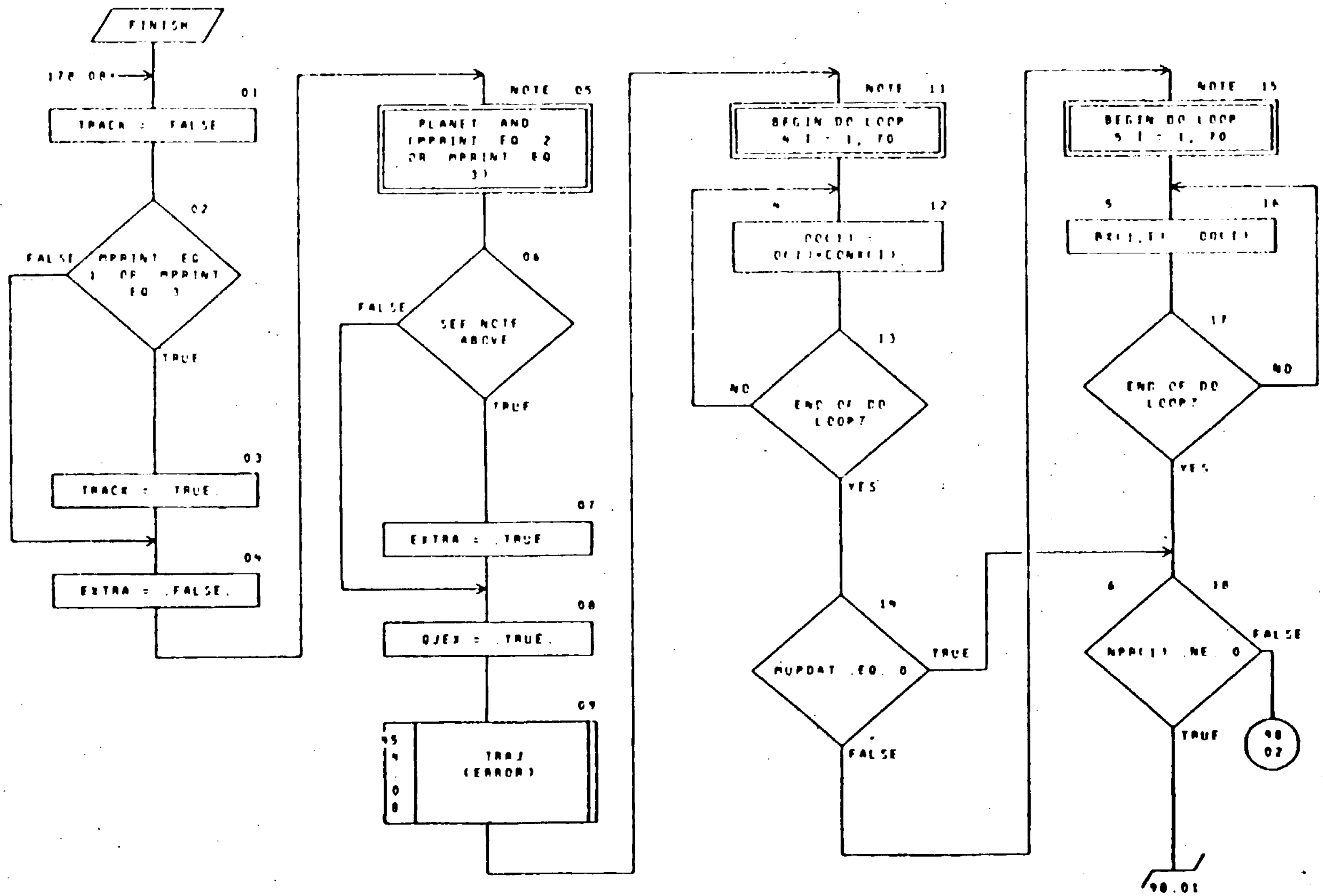


CHART TITLE - SUBROUTINE FINISH

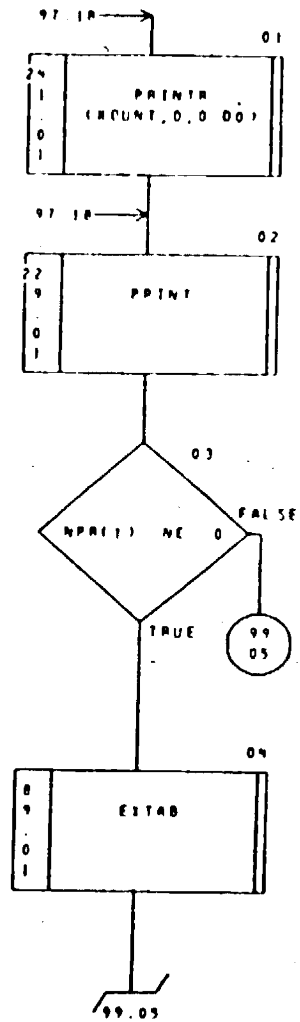
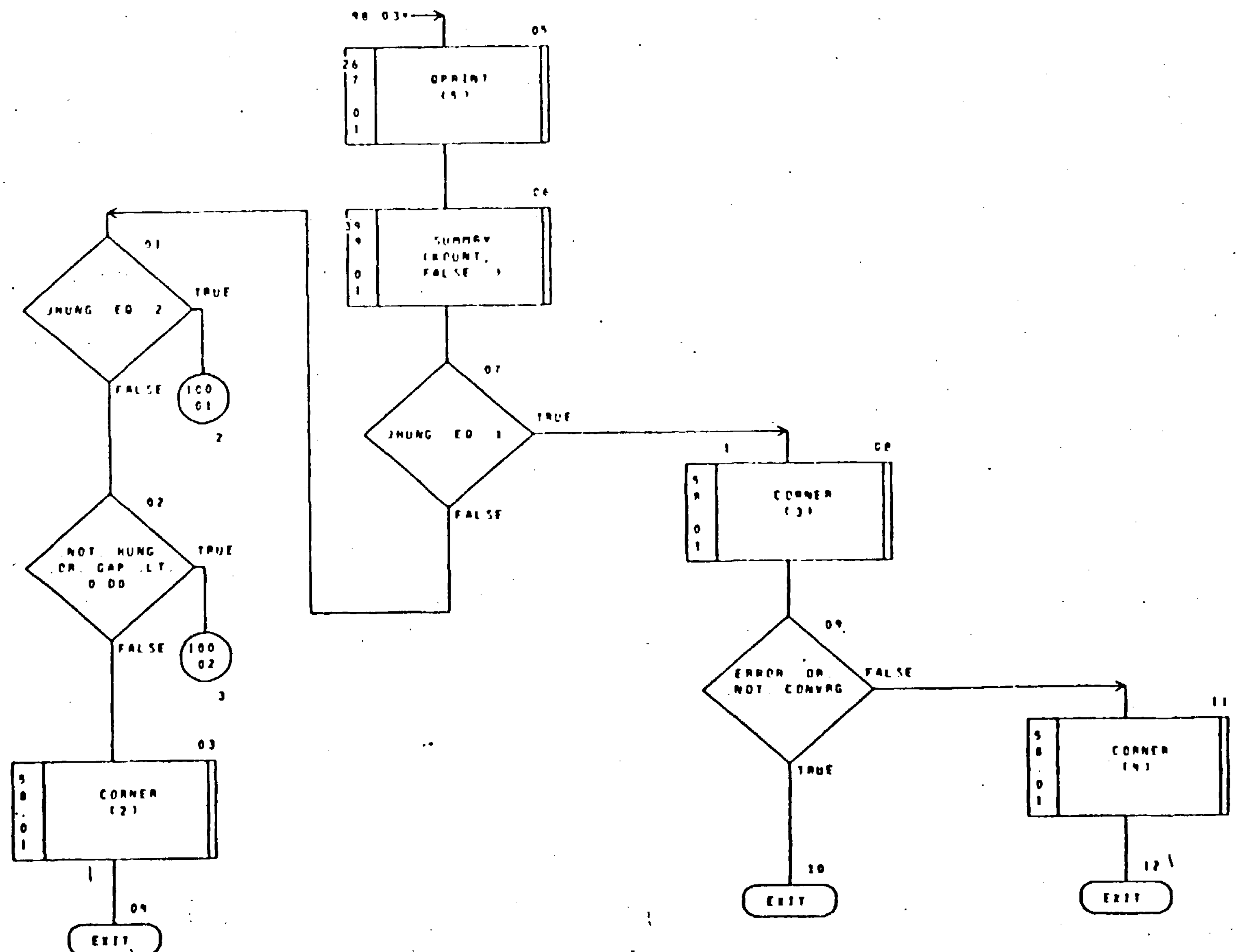


CHART TITLE - SUBROUTINE FINISH



FINISH-5

CHART TITLE - SUBROUTINE FINISH

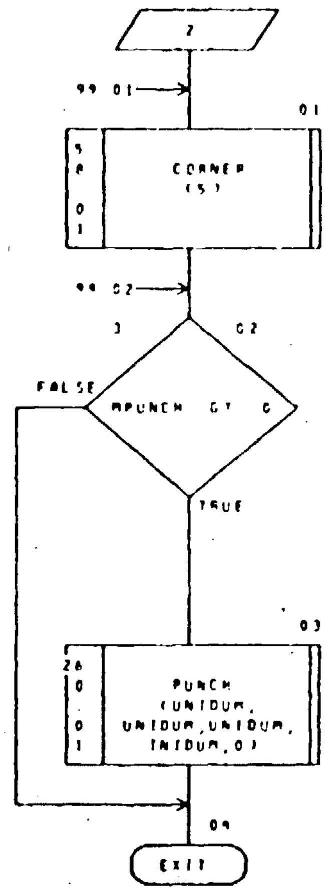


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL ERROR,CONVRG,HUNG,TRACK,QIEX,EXTRA,PLANET
COMMON /REALS/ RO1(1021),GAP,RO2(1897)
COMMON /INTGRS/ IO1(13),MUPDAT,MPRINT,IO2(24)      ,MPUNCH,IO3(2)
,NPR(4),IO4(7),NCOUNT,IO5(11),IHUNG,IO6(934)
COMMON /LOGICS/ ERROR,CONVRG,LO1(3),HUNG,LO2(18),TRACK,PLANET,
QIEX,LO4(18),EXTRA,LO3(454)
COMMON /ITERAT/ BX(5,70),BO1(215),CONX(70),BO2(70),Q(70),QQ(70),
BO3(142)
```

Name:	FUNCT
Calling Argument:	GO
Referenced Sub-programs:	SOLAR, THANG, THANGD
Referenced Commons:	LOGIC4, REAL8
Entry Points:	None
Referencing Sub-programs:	CHECK, STEP, TAP

PRECEDING PAGE BLANK NOT FILMED

Discussion: The name FUNCT is short for "function", and the purpose of the routine is to compute and store certain function values which are auxiliary with respect to the local integration of the equations of motion (subroutine DERIV or ANSTEP), but which are necessary for the isolation of remarkable points (subroutine CDERIV), which fundamentally affect the whole trajectory solution. Examples of essential remarkable points are thrust switch function roots and critical solar distances representing corners in the power curve.

Four functions are computed and their values are maintained at the start of the current computation step and at the end of the step (instantaneous values): solar distance, r; primer vector magnitude, λ; thrust switch function, σ; and thrust switch function time-derivative, $\dot{\sigma}$. Certain computations are bypassed if the value in question has already been computed in subroutine DERIV or ANSTEP, in which case the value is immediately available in COMMON.

Solar distance r (of the spacecraft) is always available from either DERIV or ANSTEP, and therefore is merely stored in FUNCT. The primer magnitude λ is available from DERIV, during thrusting flight, but not from ANSTEP, during coasting flight, and so must be directly computed,

$$\lambda = \sqrt{\Lambda \cdot \Lambda},$$

before it is stored.

The bulk of the subroutine is involved with the computation of the thrust switch function σ and its time derivative. In the usual exposition of the subject of optimal rocket flight, the thrust switch function is displayed as

$$\sigma = \frac{g\gamma q}{\nu}\left(\Lambda \cdot \bar{e}_t - \frac{\nu\lambda_\nu}{c}\right) + \lambda_s d + \lambda_\tau ,$$

where the symbols used in this discussion are defined in the Nomenclature section. However, the actual form of the switch function used internally in the program consists of the above expression multiplied by the positive factor $\nu / g\gamma q$,

$$\sigma = \Lambda \cdot \bar{e}_t - \frac{\nu\lambda_\nu}{c} + \frac{\nu}{g\gamma q}(\lambda_s d + \lambda_\tau) = \sigma_1 + \sigma_2 + \sigma_3 \text{ (respectively).}$$

The time derivative is a somewhat cumbersome expression obtained from the straightforward differentiation of the above expression. Thus

$$\dot{\sigma} = \dot{\sigma}_1 + \dot{\sigma}_2 + \dot{\sigma}_3 ,$$

where

$$\dot{\sigma}_1 = \frac{d}{dt}[\Lambda \cdot \bar{e}_t] = \dot{\Lambda} \cdot \bar{e}_t + \Lambda \cdot \dot{\bar{e}}_t ,$$

$$\dot{\sigma}_2 = -\frac{1}{c}\frac{d}{dt}[\nu\lambda_\nu] = -\frac{1}{c}(\dot{\nu}\lambda_\nu + \nu\dot{\lambda}_\nu),$$

$$\dot{\sigma}_3 = \frac{1}{g}\frac{d}{dt}\left[\frac{\nu}{\gamma q}(\lambda_s d + \lambda_\tau)\right].$$

Upon substituting the expressions for the differential equations

$$\dot{\nu} = -h_\sigma \frac{g\gamma q}{c} ,$$

and

$$\dot{\lambda}_\nu = h_\sigma \frac{g\gamma q}{\nu^2}(\Lambda \cdot \bar{e}_t),$$

into the above expression for $\dot{\sigma}_2$, the result is obtained,

$$\dot{\sigma}_2 = h_\sigma \frac{g\gamma q}{c}\left(\frac{\lambda_\nu}{c} - \frac{\Lambda \cdot \bar{e}_t}{\nu}\right),$$

which has value zero during coasting flight, due to the presence of the factor h_σ. Upon substitution of the differential equations

$$\dot{q} = - qd/\tau_d \ ,$$

$$\dot{\gamma} = \frac{\partial \gamma}{\partial d} \dot{d} ,$$

$$\dot{\lambda}_s = h_\sigma \frac{g\gamma q}{\nu \tau_d} \left(\Lambda \cdot \bar{e}_t - \frac{\nu \lambda_\nu}{c} \right) ,$$

into the expression for $\dot{\sigma}_3$ and rearranging terms, one obtains

$$\dot{\sigma}_3 = h_\sigma \left[\frac{d}{\tau_d} (\sigma_1 + \sigma_2 - \frac{\tau_d}{c} (\lambda_s + \frac{\lambda_\tau}{d})) \right]$$

$$+ \frac{\nu}{g\gamma q} \left[\dot{d} (\lambda_s - \frac{(\lambda_s d + \lambda_\tau)}{\gamma} \frac{\partial \gamma}{\partial d}) + \frac{d}{\tau_d} (\lambda_s d + \lambda_\tau) \right]$$

in which the factor $\nu/g\gamma q$ is the inverse of the instantaneous acceleration, $1/a$. In the program, the acceleration, a, is used in place of $g\gamma q/\nu$, and special precautions are taken for situations where the acceleration tends toward zero (vanishes) along the trajectory, such as when the solar array tilt angle tends toward 90^o.

The time derivative of the density is zero $(\dot{d} = 0)$ when the solar panels are tilted so as to maintain constant power; when the spacecraft is at a sufficient distance from the sun such that the solar arrays are maintained normal to the sun line, the density derivative is given by

$$\dot{d} = - 2 \frac{R \cdot \dot{R}}{r^4} .$$

When the solar arrays are tilted to such a degree that less power than the maximum instantaneous power is being accepted, which may occur when solar panel degradation is being simulated, then

$$\dot{d} = \left[\frac{\dot{\sigma}_1}{(\sigma_1 + \sigma_2)} + \frac{\gamma q}{\tau_d}\right]\left(\frac{\partial^2 \gamma}{\partial d^2}\right)^{-1} .$$

Actually, $\partial \gamma/\partial d$ and $\partial^2 \gamma/\partial d^2$ are total derivatives, but the partials-symbol, ∂, is used to avoid confusion with the symbol for density, d.

The term $\lambda_s d\nu/g\gamma q$, which is one component of σ_3, gives numerical difficulty when d tends toward zero, in which case γ also vanishes. However,

$$\lim_{d \to 0} \left(\frac{\gamma}{d}\right) = a_o ,$$

where a_o is the leading coefficient in the solar power law. Therefore, the troublesome term σ_3 is computed as $\lambda_s \nu/g a_o q$ whenever the instantaneous acceleration, $a = g\gamma q/\nu$, is less than a very small number (10^{-14}), unless the propulsion-time adjoint variable λ_τ is nonzero, in which case σ_3 is set to 10^{30} sign (λ_τ) since the other component of σ_3, $\nu\lambda_\tau/g\gamma q$, then dominates.

FUNCT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
R(2)	SU	REAL8	Spacecraft solar distance, r, at start of computation step (R(1)) and instantaneously (R(2)), in AU.
X(50)	UA	REAL8	Array of trajectory dependent-variables. (See subroutine RKSTEP).
FT	U	REAL8	Reference thrust acceleration, g, in AU/tau^2.
GO	UX		Logical indicator for stepping forward; when true, perform saving operation; when false, bypass saving operation.
PP(2)	SU	REAL8	Primer magnitude, λ, defined similar to R(2).

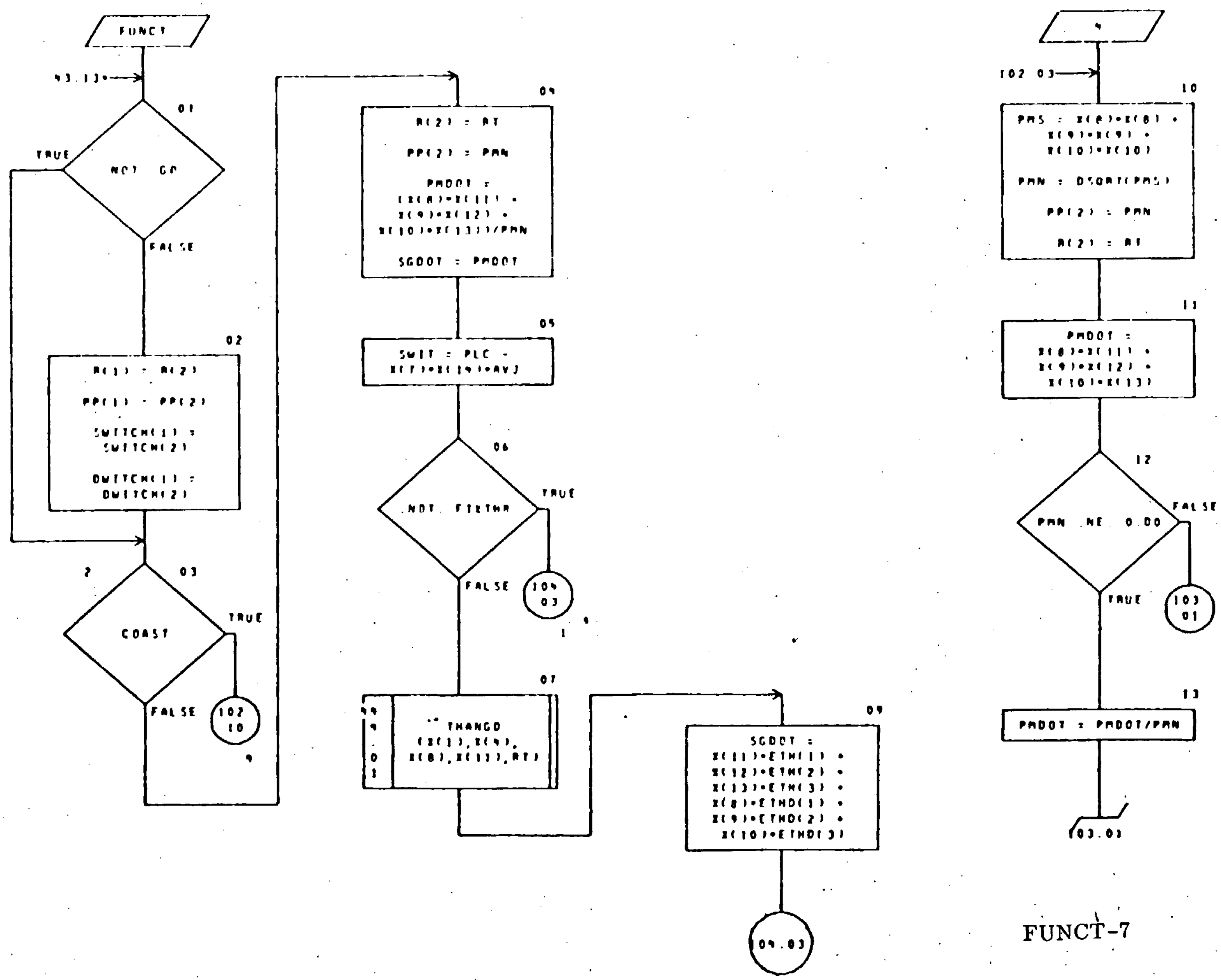
FUNCT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
RC	U	REAL8	Cube of spacecraft solar distance, r^3, in AU^3.
RT	UA	REAL8	Spacecraft solar distance, r, in AU.
XD(50)	U	REAL8	Array of trajectory dependent-variable derivatives; XD(17) is $dt/d\beta = r^n$, used for conversion from generalized derivatives to time derivatives.
ACC	SU	REAL8	Thrust acceleration, a, in AU/tau^2.
AVJ	U	REAL8	Inverse of jet exhaust speed, 1/c, in $EMOS^{-1}$.
A1S	U	REAL8	Leading power-law coefficient, a_o.
ETH(3)	SU	REAL8	Thrust unit vector, $\bar{e}_t$.
PLC	SU	REAL8	First component of thrust switch function, σ_1.
PMN	SU	REAL8	Primer magnitude, λ.
PMS	SU	REAL8	Square of primer magnitude, λ^2.
ETHD(3)	U(S)	REAL8	Thrust unit vector time-derivative, $\dot{\bar{e}}_t$, in tau^{-1}.
HEAT	U	LOGIC4	Indicator for maintaining solar panels normal to sun at all times, including during high solar proximity.
POWR	U(S)	REAL8	Power ratio, γq.
SWIT	SU	REAL8	$\sigma_1 + \sigma_2$.
ALTAU	U	REAL8	Propulsion-time adjoint variable, λ_τ.
COAST	U	LOGIC4	Indicator for coasting flight or thrusting flight; used in place of h_σ.

FUNCT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
DPOWR	U(S)	REAL8	$q\,\partial\gamma/\partial r$.
ERODE	U	LOGIC4	Power degradation option indicator.
PCURV	U	LOGIC4	Indicator for condition in which solar arrays are oriented to receive the maximum power permissible under the current power-curve assumption, or to be tilted away from the maximum permissible due to degradation considerations.
PMDOT	SU	REAL8	Primer-magnitude time derivative, $\dot{\lambda}$.
DEGRAD	U(S)	REAL8	Degradation damage factor, q.
DENSIT	U(S)	REAL8	Power particle-density function, d, in AU^{-2}.
DPOWDD	U(S)	REAL8	$q\,\partial^2\gamma/\partial d^2$.
DWITCH(2)	SU	REAL8	$\dot{\sigma}$, defined similarly to R(2).
FIXTAU	U	LOGIC4	Indicator for non-zero λ_τ.
FIXTHR	U	LOGIC4	Indicator for fixed thrust-angle.
REGION	U	LOGIC4	Indicator for spacecraft solar proximity; demarks two possible regions in space, separated by sphere about sun of specified radius, at which power function (or its derivative) has a corner.
SWITCH(2)	SU	REAL8	Thrust switch function, σ, defined similarly to R(2).
TAUPOW	U	REAL8	Negative inverse of characteristic degradation time, $-1/\tau_d$, in tau^{-1}.

CHART TITLE - SUBROUTINE FUNCT(GO)



FUNCT-7

CHART TITLE - SUBROUTINE FUNCT(GO)

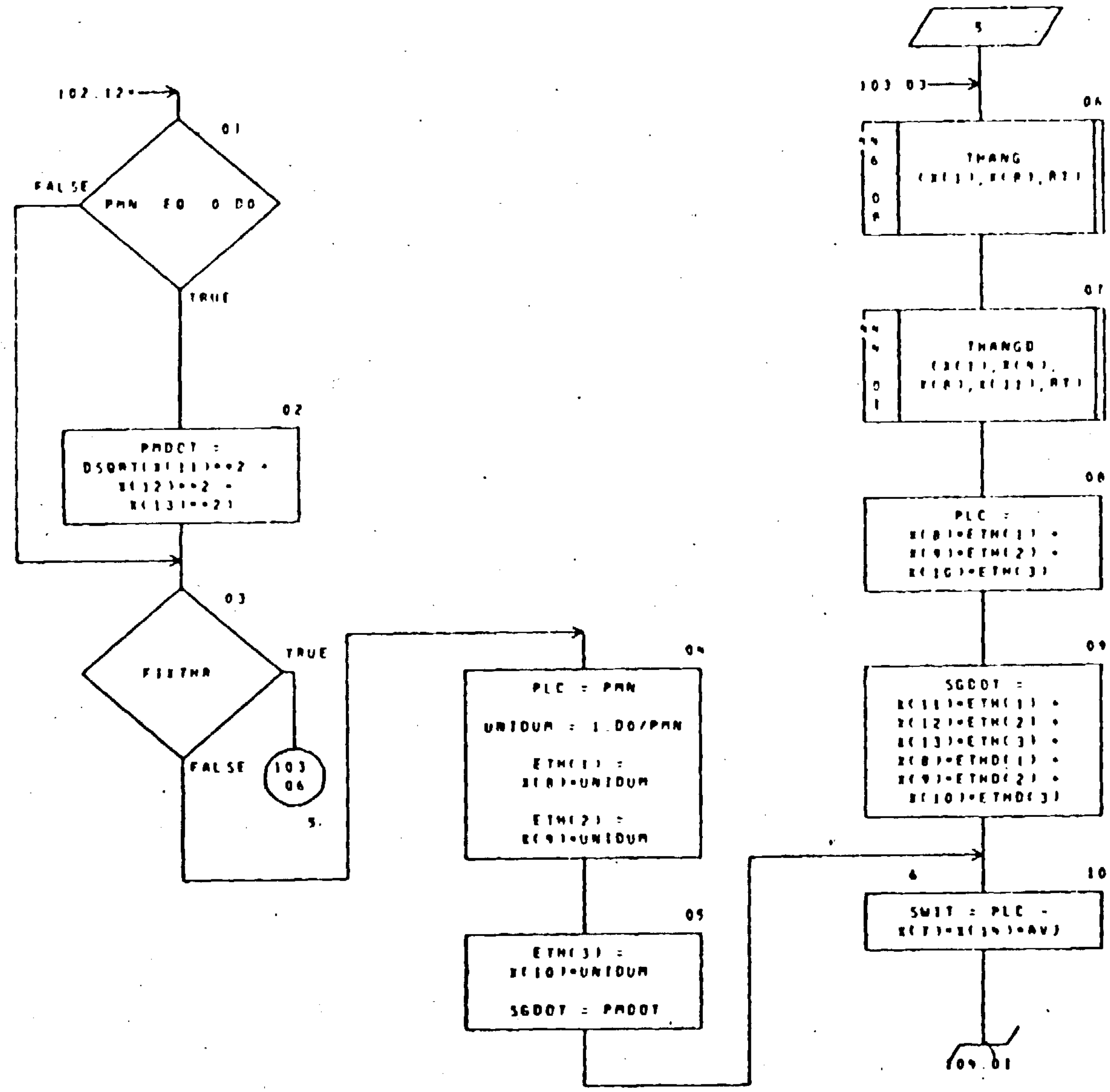
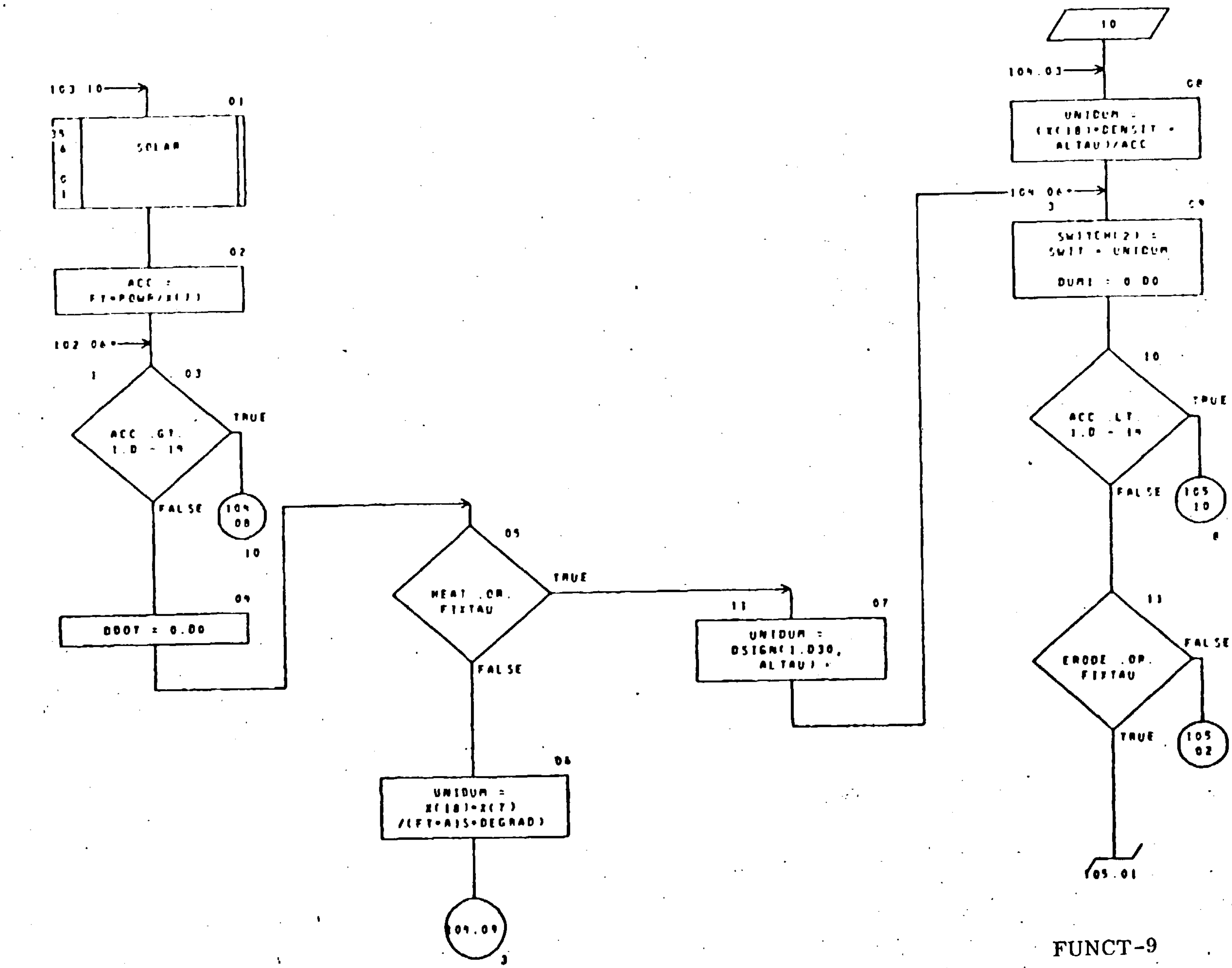


CHART TITLE - SUBROUTINE FUNCT(GO)



FUNCT-9

CHART TITLE - SUBROUTINE FUNCT(GO)

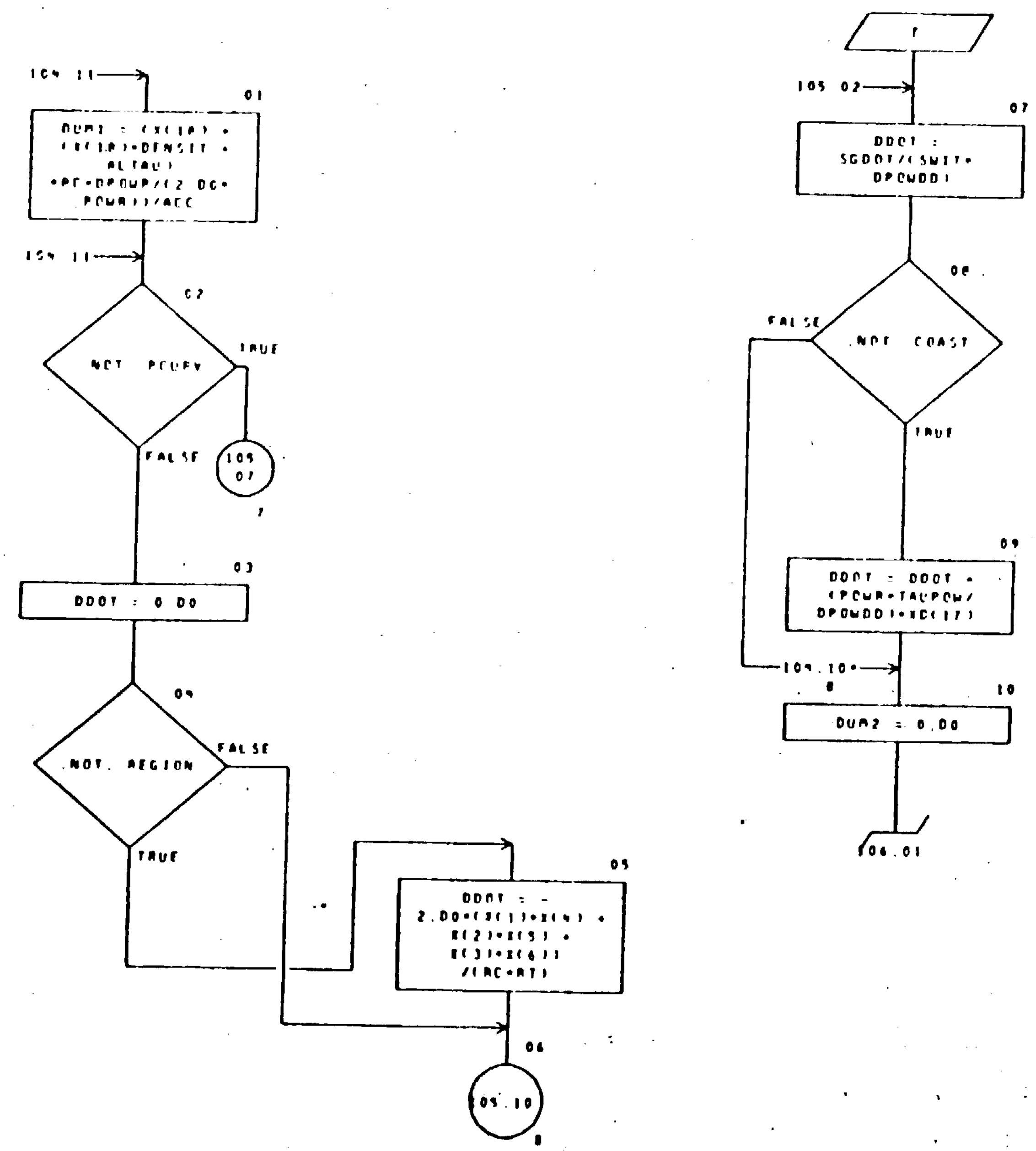
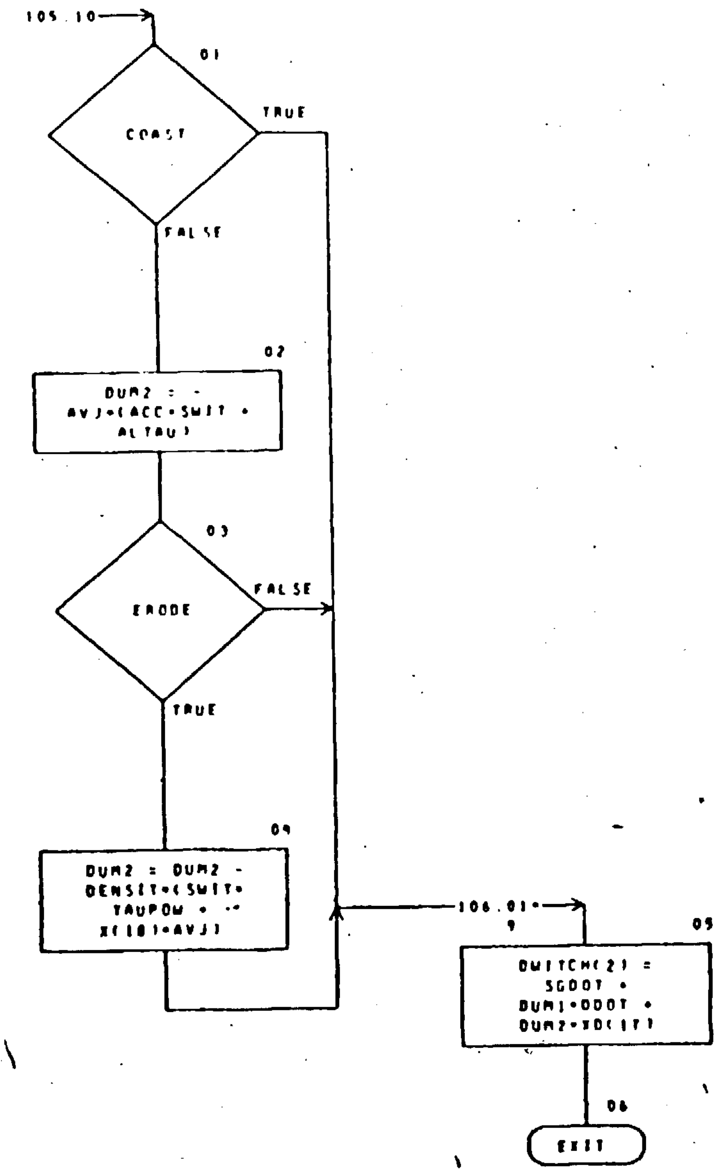


CHART TITLE - SUBROUTINE FUNCT(GO)



FUNCT-11

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL FIXTHR,FIXTAU,COAST,GO,ERODE,PCURV,REGION,HEAT
COMMON /REAL8/ R01(11),FT,R02(161),A15,P15,ALTAU,R03(24),
P(2),PP(2),
SWITCH(2),DWITCH(2),R04(617),ETH(3),ETHD(3),R05(9),SWIT,P16(2),
PLC,R06(298),
PMDOT,R07(2),PMN,PMS,RT,RG9,PC,ACC,RCP(3),POWR,DPOWR,TAUPOW,
R10(12),AVJ,R11(6),DENSIT,R13(5),DPOWDD,R14(6),DEGRAD,R17(60),
X(50),XD(50),R12(650)
COMMON /LOGIC4/ L01(10),FIXTHR,L02(7),REGION,ERODE,L05(7),
FIXTAU,HEAT,L03(2),COAST,L04(4),PCURV,L06(46))
```

C-3

...AL PAGE IS
POOR QUALITY

Name: GET I

Calling Argument: V, DECL

Referenced Sub-programs: GUNTHR

Referenced Commons: GUNCOM, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: OMASS

Discussion: FUNCTION subroutine GET I obtains, by iteration, the optimum parking orbit inclination i whenever the departure asymptote declination δ is sufficiently large in magnitude to warrant a non-due-east launch, and provided the optimization of i has not been assigned to the MINMX3 iterator via subroutine GET Q.

The transversality function whose root is obtained by Newton's iteration is

$$T(i) = 2c_1 i + c_2 - \frac{\partial v_g}{\partial i_\infty},$$

where c_1 and c_2 are coefficients in the expression for the characteristic-speed penalty associated with a non-due-east launch, v_g is the minimum incremental speed in departing the parking orbit (the symbol Δv is used in the description of subroutine GUNTHR), and i_∞ is the departure asymptote out-of-plane angle.

The initial guess for the optimum i is obtained by generating a grid of ten values of i between the values of the launch site latitude and the declination δ, and selecting the value of i from that grid for which $|T(i)|$ is a minimum. The derivative $\partial T(i)/\partial i$ used in Newton's iteration is constructed by finite differences, since, although $\partial v_g/\partial i_\infty$ is available from subroutine GUNTHR, $\partial^2 v_g/\partial i_\infty^2$ is not. Finally, if, during the iteration, the value of i converges asymptotically to the value of the launch site latitude, the iteration is forced to quit, and i is set equal to the launch site latitude in value.

Messages and printouts: Should Newton's iteration to obtain the optimum parking orbit inclination fail (exceed 100 steps), the iteration is halted, the error indicator is set, and the following is printed on unit 6:

MAX ITERATIONS TO GET PARKING ORBIT INCLINATION

INCLINATION = (i) DEGREES F = (T(i)) DELXIN = (Δi)

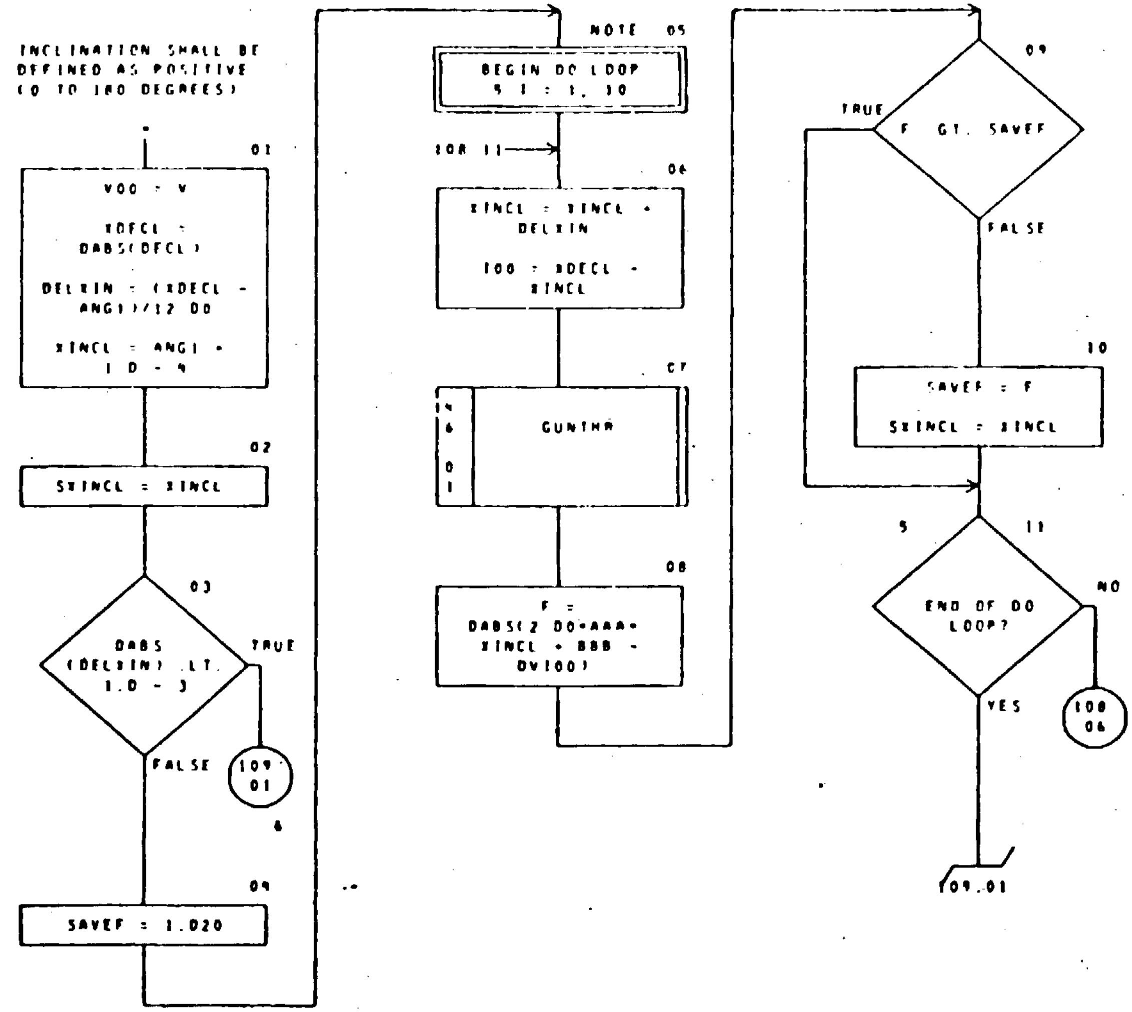
where i is the inclination, in degrees, T(i) is the function whose root is sought, in EMOS/radian, and Δi is the inclination iteration-step-size, in radians, all at the last step of the iteration. Concurrently, the message is printed on unit 12:

MAX ITERS TO GET I (i) (T(i)) (Δi)

GET I EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
V	UX		Departure hyperbolic excess speed, $v_{\infty o}$, in EMOS.
AAA	U	REAL8	Non-due-east velocity penalty coefficient, c_1, in EMOS/radian^2.
BBB	U	REAL8	Non-due-east velocity penalty coefficient, c_2, in EMOS/radian.
DEG	U	REAL8	Radians to degrees conversion factor.
I00	SU	GUNCOM	Out-of-plane angle, i_∞, in radians.
V00	S	GUNCOM	Departure hyperbolic excess speed, $v_{\infty o}$, in EMOS.
ANG1	U	REAL8	Launch site latitude, in radians.
DECL	UX		Departure asymptote declination, δ, in radians.
GET I	SX		Parking orbit inclination, i, in radians.
DVI00	U	GUNCOM	$\partial v_g / \partial i_\infty$, in EMOS/radian.
ERROR	S	LOGIC4	Program master error indicator.

CHART TITLE - FUNCTION GETI(V,DECL)



GET I-3

CHART TITLE - FUNCTION GETI(V,DECL)

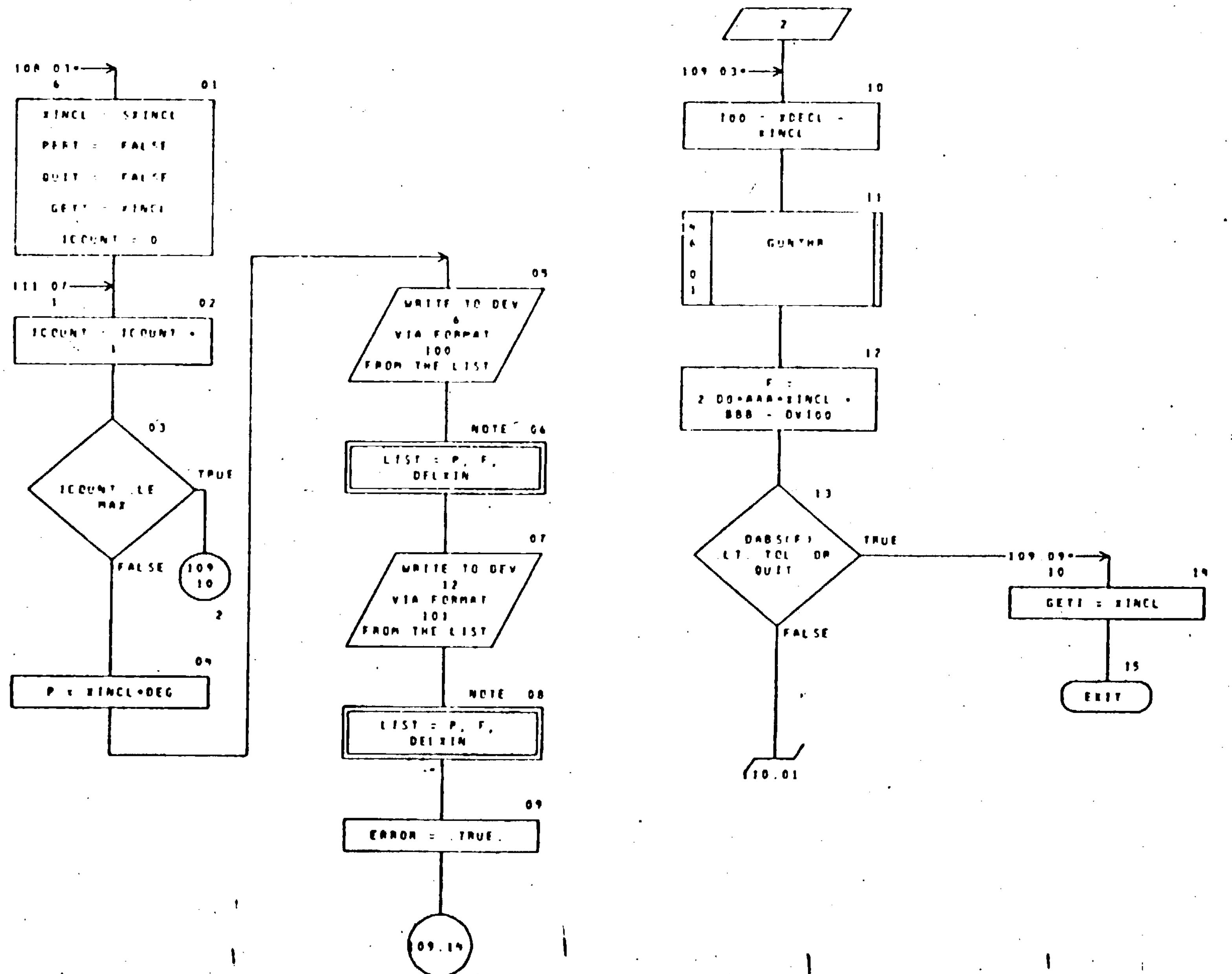
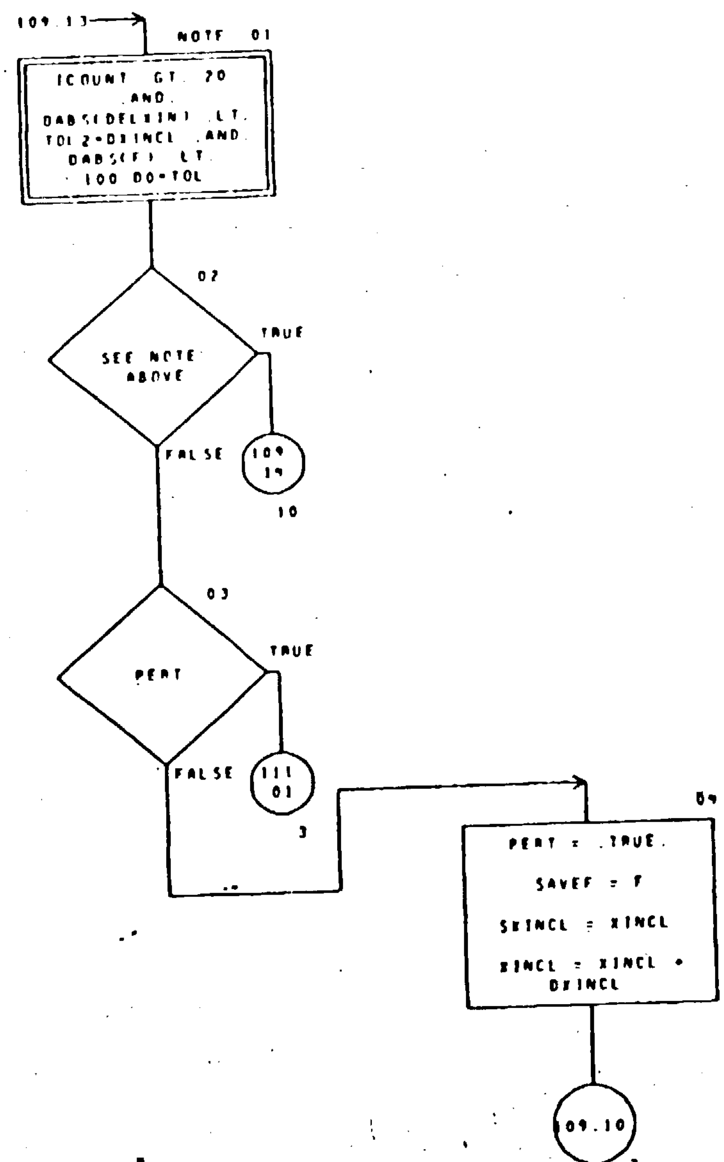


CHART TITLE - FUNCTION GETIIV,DECL)



GET I-5

CHART TITLE - FUNCTION GETI(V,DECL)

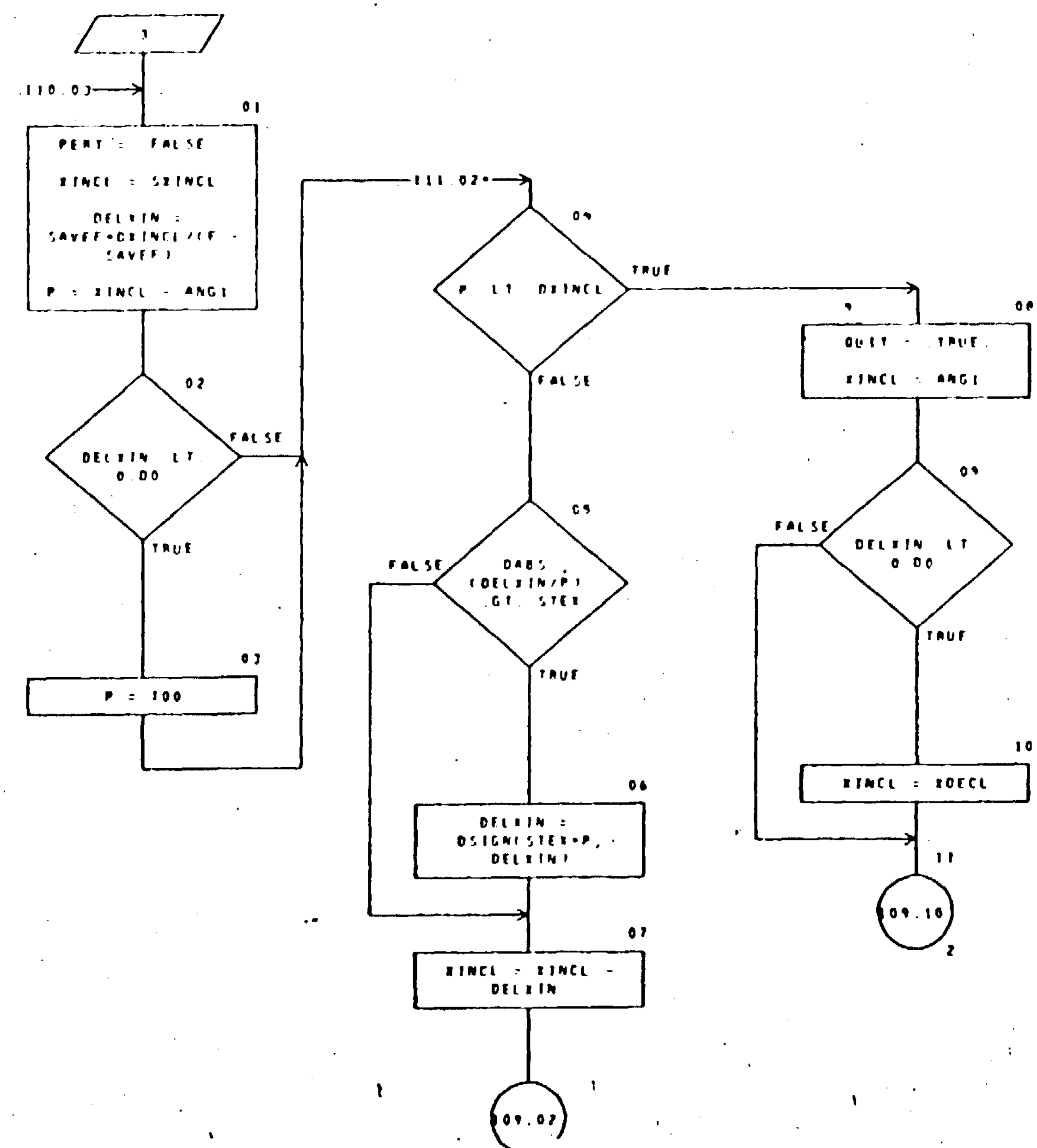


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
          IMPLICIT REAL*8 (A-H,O-Z)
          REAL*8 IOO
          LOGICAL PERT,ERROR,LDVVOO,LDVIOO,QUIT
          COMMON /REAL8/ RO1(164),AAA,BBB,RO2,ANG1,RO3(194),DEG,RO4(1687)
          COMMON /LOGIC4/ ERROR,LO1(499)
          COMMON /GUNCOM/ VC,VODIV,VOG,IGO,DV,DVVOC,DVIOC,LDVVOS,LDVIOO
          DATA MAX,TOL,DXINCL,STEP,TOL2 /100,1.D-14,1.D-9,.99D0,1.D-4/
100       FORMAT(1HO,47HMAX ITERATIONS TO GET PARKING ORBIT INCLINATION/1H ,
         113HINCLINATION =D25.16,8H DEGREES,8X3HF =D16.8,
         5X6HDELXIN =D16.8)
101       FORMAT(1H ,18HMAX ITERS TO GET I3D16.8)
```

ORIGINAL PAGE IS
OF POOR QUALITY

GET I-7

PRECEDING PAGE BLANK NOT FILMED

Name: GETQ

Calling Argument: None

Referenced Sub-programs: None

Referenced Commons: GUNCOM, INTGR4, ITERAT, ITER2, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: TRAJ

Discussion: Subroutine GETQ generates the values of the iterator (active) dependent-variables, which are contained in the Q array. All trajectories are integrated forward from the launch date t_o and are terminated at the time $(t_n - t_o)$ later. Primary target conditions are computed using the variables evaluated at this trajectory termination time. For multiple-target missions in which intermediate targets are present, the integration is interrupted at intermediate times t_i and appropriate quantities are stored for the computation of dependent conditions. The total (active and inactive) iterator dependent-variable array is the FXL array; this array is allocated the same as the iterator dependent-variable array Yi which is input to the program. However, only the required subset of the FXL array is computed, since the logical selection-array ABY causes the bypass of FXL computations which are not required. The condensed array Q is then obtained from the expanded array FXL by employing the index-set array MM at the end of the subroutine.

The Fortran statement numbers 1, 2, 3, ..., 70 correspond to the index of the FXL array, similar to the Yi array input to the program, and the documentation on the latter should be referenced to determine which end conditions or transversality conditions are being computed in conjunction with a given Fortran statement number. Within each set of computations associated with a given Fortran statement number, only one of the IF tests will hold true at any given time (i.e., during any given case of the computer run). These IF tests contain the logical variables A1A, A1B, A1C, A2A,...A69A, A70A which are defined in subroutine SETUP and which correspond to the trigger settings of the individual iterator dependent-variables input to the program.

At the beginning of the subroutine, a function FOIL (A1, A2) is defined, which simply computes the ratio A1/A2 and foils the computer from printing overflow messages when A2 is extremely small.

The only FXL computations which are not grouped under their respective Fortran statement numbers are those corresponding to the targeting and transversality conditions for extra-ecliptic missions when program input quantity IOUT = 2, and these consist of six conditions computed at the top of the subroutine, the first three of which correspond to specified final semi-major axis a, eccentricity e, and inclination i (to the ecliptic), and the latter three of which are transversality conditions which yield the optimization of the unspecified final orbital parameters ω (argument of perihelion), Ω (node), and f (true anomaly):

$$\hat{h} \cdot C = 0,$$

$$\hat{k} \cdot C = 0,$$

$$\frac{\mu}{r} (\Lambda \cdot R) + r^2 (\dot{\Lambda} \cdot \dot{R}) = 0,$$

where $\hat{h}$ is the unit angular momentum vector of the final orbit, $\hat{k}$ is the unit vector normal to the ecliptic (northward), and C is the vector constant of the motion defined by

$$C = (R \times \dot{\Lambda}) - (\dot{R} \times \Lambda).$$

The final orbital elements a, e, and i are computed as follows:

$$r^2 = x^2 + y^2 + z^2,$$

$$v^2 = \dot{x}^2 + \dot{y}^2 + \dot{z}^2,$$

$$d = x\dot{x} + y\dot{y} + z\dot{z},$$

$$a = r/(2 - r v^2)$$

$$e = \sqrt{\frac{d^2}{a} + (r v^2 - 1)^2}$$

$$\hat{h} = R \times \dot{R} / |R \times \dot{R}| ,$$

$$i = \cos^{-1} (\hat{h}_z) .$$

The above computations appear above Fortran statement number 1 in the source listing. The remainder of the computations are catalogued under Fortran statement numbers as described earlier.

Denoting a specific target's position and velocity as P_i and $\dot{P}_i$, respectively, a constraint on the spacecraft position R_i is imposed by nulling the position error; i.e., by forcing the satisfaction of the equation

$$\Delta R_i = R_i - P_i = 0.$$

Similarly, a constraint on the spacecraft velocity $\dot{R}_i$ is imposed by nulling the velocity error,

$$\Delta \dot{R}_i = \dot{R}_i - \dot{P}_i - V_{\infty i} = 0,$$

where $V_{\infty i}$ is the arrival excess velocity at the i^{th} target. These equations apply both to the primary target, FXL(1) through FXL(6) with trigger settings of 1 (see program inputs Yi), and to intermediate targets, FXL(41) to FXL(46), FXL(51) to FXL(56), and FXL(61) to FXL(66) with trigger settings of 1. (Trigger settings of 1 correspond to logical indicators A1A, A2A, A3A, ..., A41A, ..., A44A, etc. being .TRUE., where the final "A" in these Fortran variable names refers to the trigger setting; a final "B" corresponds to a setting of 2, etc.)

The open angle option is restricted to problems of two-dimensional motion in the x-y plane. The option is designed specifically for the problem of open angle transfer from a given point to a specified solar distance r_f. This target condition is written, simply,

$$|R_n| = r_f ,$$

where the subscript n denotes the primary, or final, target. This corresponds to

FXL(1) under A1B. The capability of imposing circular orbit conditions at this solar distance is also available. This vector target condition is written in the form of a velocity error as follows:

$$\Delta\dot{R} = \dot{R}_n - \sqrt{\frac{\mu}{r_f}}\left(\bar{k} \times \frac{R_n}{|R_n|}\right) - V_{\infty n} = 0,$$

where μ is the gravitational constant of the sun and $\bar{k}$ is a unit vector along the z-axis. This applies to the x and y components of velocity and corresponds to FXL(4) and FXL(5) under A4A and A5A (similar to the general relation for $\Delta\dot{R}_i$ above).

Relations governed by A1C, A2C, A3C, A4C, and A5C apply to optimum departure maneuvers from a circular parking orbit about Earth (using electric propulsion or high thrust nuclear or chemical propulsion) and are given in the description of subroutine VPRINT.

Extra-ecliptic targeting conditions associated with program input quantity IOUT = 1 are generated by subroutine GET RV and are computed via FXL(1) through FXL(6) under a trigger setting of 1.

Position and velocity at intermediate targets are continuous. The intercept of an intermediate target is achieved by imposing a constraint on the position error. At the i^{th} target, this constraint may be written

$$\Delta R_i = R_i - P_i = 0.$$

This corresponds to FXL(41), FXL(42), and FXL(43) for the first intermediate target, FXL(51) through FXL(53) for the second intermediate target, and FXL(61) through FXL(63) for the third intermediate target. One may optionally constrain the passage speed $v_{\infty i}$ at an intermediate target. The constraint equation for this is

$$\Delta\dot{R}_i = \dot{R}_i - \dot{P}_i - V_{\infty i} = 0.$$

This corresponds to FXL(44) through FXL(46) for the first, FXL(54) through FXL(56) for the second, and FXL(64) through FXL(66) for the third intermediate target, respectively, with trigger settings of 1.

When the flight time from launch to time of primary-target intercept is constrained, the end condition is computed via FXL(16) under A16C:

$$t_{max} = t_n - t_o ,$$

where t_{max} is the specified total flight time and t_o and t_n are the launch and primary-target intercept times, respectively.

The capability for constraining propulsion time is implemented through satisfaction of the optional end condition,

$$\tau_f = \int_{t_o}^{t_n} h_\sigma \, dt ,$$

where h_σ is the thrust step function and τ_f is the desired propulsion time. This is computed using FXL(8) under the logical trigger A8B.

Several spacecraft parameters are available as constraints. These are the net spacecraft mass, m_{net}, using FXL(7) and A7C, the intermediate-target drop masses $m_{drop\,i}$ under FXL(50), FXL(60), and FXL(70), the sample masses $m_{samp\,i}$ under FXL(49), FXL(59), and FXL(69), and the reference power, p_{ref}, under FXL(11) governed by A11C. Also available as constraints are virtually all of the iterator independent-parameters, under a trigger setting of 2 for each FXL(i).

The above constitutes the discussion for the two-point boundary value problem end-conditions except for the transversality conditions. The application of the indirect method of optimization leads to a set of necessary conditions known as transversality conditions, that must be satisfied by the solution. For a given performance index π which is to be minimized, the general equation for the transversality conditions is written

$$kd\pi + \sum_{i=1}^{n} \left[\Lambda \cdot d\dot{R} - \dot{\Lambda} \cdot dR + \lambda_\nu d\nu + \lambda_g dg + \lambda_c dc + \lambda_\phi d\phi + \lambda_\tau d\tau - h_v dt \right]_{t_{i-1}}^{t_i} = 0.$$

The convenient choice is made whereby λ_g, λ_c, and λ_ϕ are forced to be continuous at each intermediate target, which means that, for example, only $\lambda_g(t_n)$ need appear in the derived transversality expressions rather than the cumbersome expression

$$\lambda_g(t_n) - \sum_{i=1}^{n-1} (\lambda_g^+(t_i) - \lambda_g^-(t_i)) - \lambda_g(t_o).$$

This is because $\lambda_g(t_n)$ alone, with $\lambda_g(t_o) = 0$ and $\lambda_g^+(t_i) = \lambda_g^-(t_i)$ for each i, has the same value as the cumbersome expression cited above if $\lambda_g(t_o)$ were not zero and $\lambda_g(t_i)$ were not continuous, and this is due to the absence of λ_g in the differential equations, the same being true for λ_c and λ_ϕ. The scalar k is an arbitrary positive constant which expresses the arbitrariness of the performance index; in other words, the minimization of π is equivalent to the minimization of 2π, 3π, ..., etc. k effectively renders the general transversality condition linear and homogeneous in the adjoint variables, thus allowing the elimination of one terminal condition from the problem by appropriate choice of a value for k. This particular aspect of the problem is discussed more thoroughly in the description of subroutine TRAJ.

All available transversality conditions are derived for the problem of maximizing net spacecraft mass, i.e., $\pi = - m_{net}$. From the earlier definition of m_{net}, one may write

$$\pi = j_r m_{rs} + m_o \Big\{ k_s + k_t - (1 + k_t) \nu_n + j_r (1 + k_{rt}) e_x \Big[(1 + j_t k_t) \nu_n - j_t k_t \Big(1 + \sum_{i=1}^{n-1} (k_{samp\,i} - k_{drop\,i})\Big) \Big] + (1 + k_t) \sum_{i=1}^{n-1} k_{samp\,i} - k_t \sum_{i=1}^{n-1} k_{drop\,i} \Big\} + m_{ps} \Big[1 - j_r j_{ps} (1 + k_{rt}) e_x \Big],$$

GETQ-7

where j_r is a constant equal to one if a retro stage is employed and equal to zero otherwise. For the launch vehicle dependent formulation, m_o is a function only of the launch excess speed $v_{\infty o}$ and possibly the geocentric declination δ and launch parking orbit inclination i. π may be written functionally in its most general form,

$$\pi = \pi(v_{\infty o}, v_{\infty n}, \nu_n, g, c, \delta, i).$$

Consequently, using the notation $\pi_x = \partial\pi/\partial x$, one obtains

$$d\pi = \pi_{v_{\infty o}} dv_{\infty o} + \pi_{v_{\infty n}} dv_{\infty n} + \pi_{\nu_n} d\nu_n + \pi_g dg + \pi_c dc + \pi_\delta d\delta + \pi_i di .$$

To write the partial derivatives indicated, it is convenient to first define a factor j_p, which is equal to zero if reference power is fixed and equal to one otherwise. Then one may write the indicated partial derivatives of π as follows for the launch vehicle dependent formulation:

$$\pi_{v_{\infty o}} = \pi_{m_o} \frac{\partial m_o}{\partial v_{\infty o}} ,$$

$$\pi_{v_{\infty n}} = j_r \frac{2(m_o \nu_n - j_t m_t - j_{ps} m_{ps})(1 + k_{rt})(1 - e_x) v_{\infty n}}{(v_{\infty n}^2 + 2v_c^2)^{\frac{1}{2}} [2c_r - c_1(f_x - e_x)^2]} \left[1 + \frac{2c_1 e_x f_x v_c^2}{(v_{\infty n}^2 + v_c^2)(v_{\infty n}^2 + 2v_c^2)^{\frac{1}{2}}}\right],$$

$$\pi_g = j_p \frac{m_{ps}}{g} (1 - j_{ps} g_x),$$

$$\pi_c = j_p \frac{m_{ps}}{c} \left(1 - \frac{c\eta'}{\eta}\right)\left(1 - j_{ps} g_x\right) .$$

The definition of symbols appearing in this discussion may be found in the general Nomenclature section of this document. Partials $\partial\pi/\partial x$ not appearing in this discussion are computed in other subroutines, and, indeed, the relations for π and π_x given above and below are generally not explicitly computed but are incorporated within the various transversality conditions.

For the launch vehicle independent formulation, the partial $\pi_{v_{\infty n}}$ remains unchanged from the expression given above. The remaining partials become

$$\pi_{v_{\infty o}} = 0,$$

$$\pi_g = -\frac{m_o}{g}\pi_{m_o},$$

$$\pi_c = -\frac{m_o}{c}\left(1 - \frac{c\eta'}{\eta}\right)\pi_{m_o}.$$

For the launch vehicle independent formulation, the factor j_p is always taken to be zero, since a condition of the formulation is that reference power is specified.

The transversality conditions are as follows. When m_o is independent of δ and i, the launch excess speed is optimized via

$$-\frac{k\pi_{v_{\infty o}}}{\lambda_o} - (1 - j_p)\frac{g\lambda_g}{\lambda_o m_o}\frac{\partial m_o}{\partial v_{\infty o}} - 1 = 0.$$

This corresponds to FXL(13) governed by A13A. When m_o is a function of launch asymptote declination δ, δ is optimized via

$$f(\partial v_g/\partial\delta) - \Lambda_o \cdot [(V_{\infty o} x \bar{n}_p) x V_{\infty o}/v_{\infty o}\cos\delta] = 0,$$

where

$$f = [k_s + k_t - (1 + k_t)\nu_n - g\lambda_g/km_o]dm_o/dv_c \quad .$$

This corresponds to FXL(10) governed by A10A.

For optimized excess speed at an intermediate target, the primer is continuous, i.e.,

$$\Lambda_i^+ - \Lambda_i^- = 0.$$

This corresponds to FXL(44), FXL(45), and FXL(46) governed by A44B, A45B, and A46B (trigger settings of 2) for the first intermediate target, and to FXL(54) etc., and FXL(64) etc., for the second and third intermediate targets, respectively.

For optimized arrival excess speed (at the primary target) in problems where a retro stage is employed, the transversality condition is,

$$- \frac{k\pi_{v_{\infty n}}}{\lambda_n} - 1 = 0.$$

This corresponds to FXL(14) under A14A.

For optimized thrust acceleration with unspecified reference power in the launch vehicle dependent formulation, the transversality condition is,

$$- \frac{k\pi_g}{\lambda_g} + 1 = 0.$$

This corresponds to FXL(11) under A11A with FIXPOW set off.

For optimized jet exhaust speed with unspecified reference power in the launch vehicle dependent formulation, the transversality condition is,

$$- \frac{k\pi_c}{\lambda_c} + 1 = 0.$$

This corresponds to FXL(12) governed by A12A with A11C set off.

These last two equations also apply for optimized thrust acceleration and optimized jet exhaust speed, respectively, in the launch vehicle independent formulation with fixed reference power. The appropriate expressions for the partials of π are used.

If the reference power is specified using the launch vehicle dependent formulation, but both reference thrust acceleration and jet exhaust speed are optimized, the last two transversality conditions are replaced in favor of the one condition,

$$1 - \frac{\lambda_g}{\lambda_c} \frac{g}{c} \left(1 - \frac{c\,\eta'}{\eta}\right) = 0.$$

This corresponds to FXL(12) using A12A with A11C set on. In the preceding equations λ_g and λ_c are evaluated at time t_n, the time at which the spacecraft is to be at the primary target.

When using the open angle transfer option, the transversality condition associated with the open angle is

$$\left[\dot{\Lambda}_n \times R_n - \Lambda_n \times \dot{R}_n \right] \cdot \bar{k} = 0 \,,$$

corresponding to FXL(2) governed by A2B.

For either the ephemeris or open angle options, if the final velocity is completely unspecified, as in the case of flyby missions, the appropriate vector transversality condition is,

$$\Lambda_n = 0,$$

corresponding to FXL(4), FXL(5), and FXL(6) governed by A4B, A5B, and A6B.

The transversality condition associated with optimized launch date for the ephemeris option is,

$$-\Lambda_o \cdot \ddot{P}_o + \dot{\Lambda}_o \cdot \dot{P}_o + h_{vo} = 0,$$

corresponding to FXL(15) under A15A with A16C set off.

The condition for optimized encounter date at an intermediate target is,

$$-(\Lambda_i^+ - \Lambda_i^-) \cdot \ddot{P}_i + (\dot{\Lambda}_i^+ - \dot{\Lambda}_i^-) \cdot \dot{P}_i + h_{vi}^+ - h_{vi}^- = 0,$$

corresponding to FXL(48), FXL(5 8), and FXL(68) using A48A, A58A, and A68A. The appropriate condition for optimized arrival date (at the primary target) with the ephemeris option is,

$$\Lambda_n \cdot \ddot{P}_n - \dot{\Lambda}_n \cdot \dot{P}_n - h_{vn} = 0,$$

corresponding to FXL(16) under A16A.

Since the variational Hamiltonian h_v is a constant of the motion on a given trajectory segment, the time at which it is evaluated on that segment is arbitrary. The preceding conditions pertaining to initial and final time are applicable if the total flight time is unconstrained. In the event that the total flight time is specified while both t_o and t_n are optimized, the two preceding conditions are replaced with the single condition represented by the sum of the two, i.e.,

$$\Lambda_n \cdot \ddot{P}_n - \dot{\Lambda}_n \cdot \dot{P}_n - h_{vn} - \Lambda_o \cdot \ddot{P}_o + \dot{\Lambda}_o \cdot \dot{P}_o + h_{vo} = 0.$$

This corresponds to FXL(15) governed by A15A with A16C set on.

For either the open angle or extra ecliptic options, the appropriate transversality condition associated with optimized arrival date (i.e., optimized flight time) is

$$- h_v = 0,$$

corresponding to FXL(16) under A16A.

Finally, the transversality condition associated with unspecified, but constant, thrust cone angle is,

$$\lambda_\phi (t_n) = 0,$$

corresponding to FXL(21) under APHI(1,1).

GETQ EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
O(70)	U	ITERAT	Array of iterator independent-variables, in program internal units, allocated the same as the program input iterator independent-variables Xi.
Q(35)	S	ITER2	The (active) iterator dependent-variables generated by subroutine GET Q.
R(2)	U	REAL8	Spacecraft solar distance at time of primary-target intercept (R(2)), in AU.
X(50)	U	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
AK	U	REAL8	Primer vector normalizing factor a_k, computed in subroutine TRAJ.
AM	U	REAL8	Multiplicative factor, zero or one, for the term $\Lambda_n \cdot \ddot{P}_n$ in the transversality conditions associated with flight time; computed in subroutine RETINJ.
CE	U	REAL8	Cosine of obliquity of ecliptic, cos ϵ.
FT	U	REAL8	Reference thrust acceleration, g, in AU/tau^2.
MM(70)	U	INTGR4	Index set of the active iterator dependent-variables.
P0(7)	U	REAL8	Array of initial adjoint variables Λ_o, $\dot{\Lambda}_o$, $\lambda_{\nu o}$.
SE	U	REAL8	Sine of obliquity of ecliptic, sin ϵ.
VH	U	REAL8	Speed at periapse of approach hyperbolic trajectory, pertaining to the retro-maneuver at the primary target, in meters/second.
VJ	U	REAL8	Jet exhaust speed, c, in EMOS.

GETQ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
X0(7)	U	REAL8	Initial launch planet state P_o, $\dot{P}_o$; and initial mass ratio ν_o.
ABY(70)	U	LOGIC4	Master array of iterator dependent-variable indicators.
A1A	U	LOGIC4	
A1B	U	LOGIC4	
A1C	U	LOGIC4	
A2A	U	LOGIC4	Logical indicators which are defined and initialized in subroutine SETUP and used in this subroutine for the selection-for-computation of (active) iterator dependent-variable values.
.	.	.	
.	.	.	
.	.	.	
.	.	.	
A68A	U	LOGIC4	
A68B	U	LOGIC4	
A69A	U	LOGIC4	
A70B	U	LOGIC4	
DEG	U	REAL8	Radians to degrees conversion factor.
FMS	U	REAL8	$\partial m_o / \partial v_{\infty o}$, in kg/EMOS.
FXL(70)	SU	ITERAT	Array of iterator dependent variables, allocated the same as program input quantities Yi.
MXX	U	INTGR4	The number of active iterator dependent-variables.
PMN	U	REAL8	Magnitude of primer vector, λ (corresponding to primary-target intercept time in this subroutine).
PM0	U	REAL8	Magnitude of initial primer vector, λ_o.
TAU	U	REAL8	Propulsion time, τ, in tau.
TDV	U	REAL8	Time of deep space burn, in days.
AJPP	U	REAL8	Jettison indicator j_{ps} for electric propulsion system prior to primary-target retro-maneuver.

GETQ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
APHI(2,10)	U	LOGIC4	Logical indicators associated with multiple fixed thrust-cone-angles; only APHI(i,1) is applicable at present.
CONX(70)	U	ITERAT	Array of conversion factors (between program external and internal units) for the iterator independent-variables, computed in subroutine BEGIN.
DECL	U	REAL8	Departure asymptote declination, δ, in radians.
FETA	U	REAL8	$1/c - \eta'/\eta$, in $EMOS^{-1}$.
FMSI	U	REAL8	$\partial v_c/\partial i$, in EMOS/radian (see subroutine OMASS).
HAMX(4)	U	REAL8	Variational hamiltonian values at the beginning of trajectory-segments departing intermediate targets.
IOUT	U	INTGR4	Out-of-ecliptic mission indicator.
PIMO	U	REAL8	π_{m_o}, computed in subroutine TRAJ.
PM0D	U	REAL8	Magnitude of initial primer derivative, $\lvert\dot{\Lambda}_o\rvert$.
TMAX	U	REAL8	Time elapsed from launch to intercept of primary target, in tau.
VINF	U	REAL8	Hyperbolic excess speed at the primary target, $v_{\infty n}$, in meters/second.
V00D(3)	U	REAL8	Launch hyperbolic excess velocity, $V_{\infty o}$, in AU/tau (=EMOS).
XINT (50,5)	U	REAL8	Array of trajectory dependent-variable values corresponding to arrival at the intermediate targets, computed in subroutine TRAJ, and allocated the same as the X array.

GETQ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ALTAU	U	REAL8	Propulsion-time adjoint variable, λ_τ.
CONA0	U	REAL8	Acceleration conversion factor, from AU/tau^2 to meters/sec^2.
CONSP	U	REAL8	Speed conversion factor, from AU/tau to meters/second.
CONTM	U	REAL8	Time conversion factor, tau to days.
DMDVC	U	REAL8	dm_o/dv_c, in kg/EMOS (from subroutine OMASS).
DVI00	U	GUNCOM	$\partial v_g/\partial i_\infty$, in EMOS/radian (from subroutine OMASS).
FLYBY	U	LOGIC4	Indicator for flyby of primary target (as opposed to orbiter, rendezvous, etc.).
GPLAN	U	REAL8	Gravitational constant of primary target, μ_t, in m^3/sec^2.
GSUBX	U	REAL8	Auxiliary parameter g_x, computed in subroutine TRAJ.
PSIGN	U	REAL8	Coefficient (± 1) defining the sense of the launch hyperbolic excess velocity relative to the initial primer vector.
SCALE	U	REAL8	Arbitrary positive performance index constant, k.
SEFMA(7)	U	REAL8	Array containing position and velocity of launch planet, at launch time, P_o and $\dot{P}_o$, in AU and EMOS, respectively.
SEFMB(7)	U	REAL8	Array containing position and velocity of primary target at time of target intercept, P_n and $\dot{P}_n$, in AU and EMOS, respectively.

GETQ-15

GETQ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
SEFMC(7)	U	REAL8	Time derivative of SEFMA array (w.r.t. tau^{-1}).
SEFMD(7)	U	REAL8	Time derivative of SEFMB array (w.r.t. tau^{-1}).
TEMP2	U	REAL8	Auxiliary quantity, computed in subroutine TRAJ.
TEMP4	U	REAL8	Auxiliary quantity, computed in subroutine TRAJ.
WPRIM(3)	U	REAL8	Initial primer vector, Λ_{otr}, as computed in subroutine TRAJ; "weighted primer".
XMASS(7)	U	REAL8	Array of masses and related parameters, as defined in subroutine TRAJ.
XTINT (6, 5)	U	REAL8	Positions and velocities of intermediate targets at times of intercept, P_i and $\dot{P}_i$, in AU and EMOS, respectively.
ALTITU	U	REAL8	Outdated variable (see subroutine VPRINT).
ALWAYS	U	LOGIC4	Indicator for fixed (specified) non-zero propulsion-time adjoint-variable.
DELTAV	U	REAL8	Deep space burn Δv, in meters/second.
FIXPOW	U	LOGIC4	Launch-vehicle-independent trajectory option indicator.
FIXTHR	U	LOGIC4	Indicator for fixed (i.e., constant) thrust cone angle.
FPSNMH	U	REAL8	Conversion factor, knots to fps.
LEGMAX	U	INTGR4	Total (maximum) number of trajectory-segments comprising the trajectory.
LIMPHI	U	INTGR4	Highest index-value of all non-zero (multiple) fixed thrust-cone-angles; highest permissible value is currently one.

GETQ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
OUTECL	U	LOGIC4	Indicator for out-of-ecliptic mission.
PAYLOD	U	REAL8	Net spacecraft mass, m_{net}, in kg.
PLANET	U	LOGIC4	Ephemeris-option indicator.
PRZERO	U	LOGIC4	Indicator that zero initial primer vector is the desired condition.
SWITCH(2)	U	REAL8	Thrust switch function, σ_n (evaluated at endpoint of electric propulsion trajectory, in this subroutine).
XTDINT (6,5)	U	REAL8	Time derivative of XTINT array (w.r.t. tau^{-1}).

CHART TITLE - SUBROUTINE GETQ

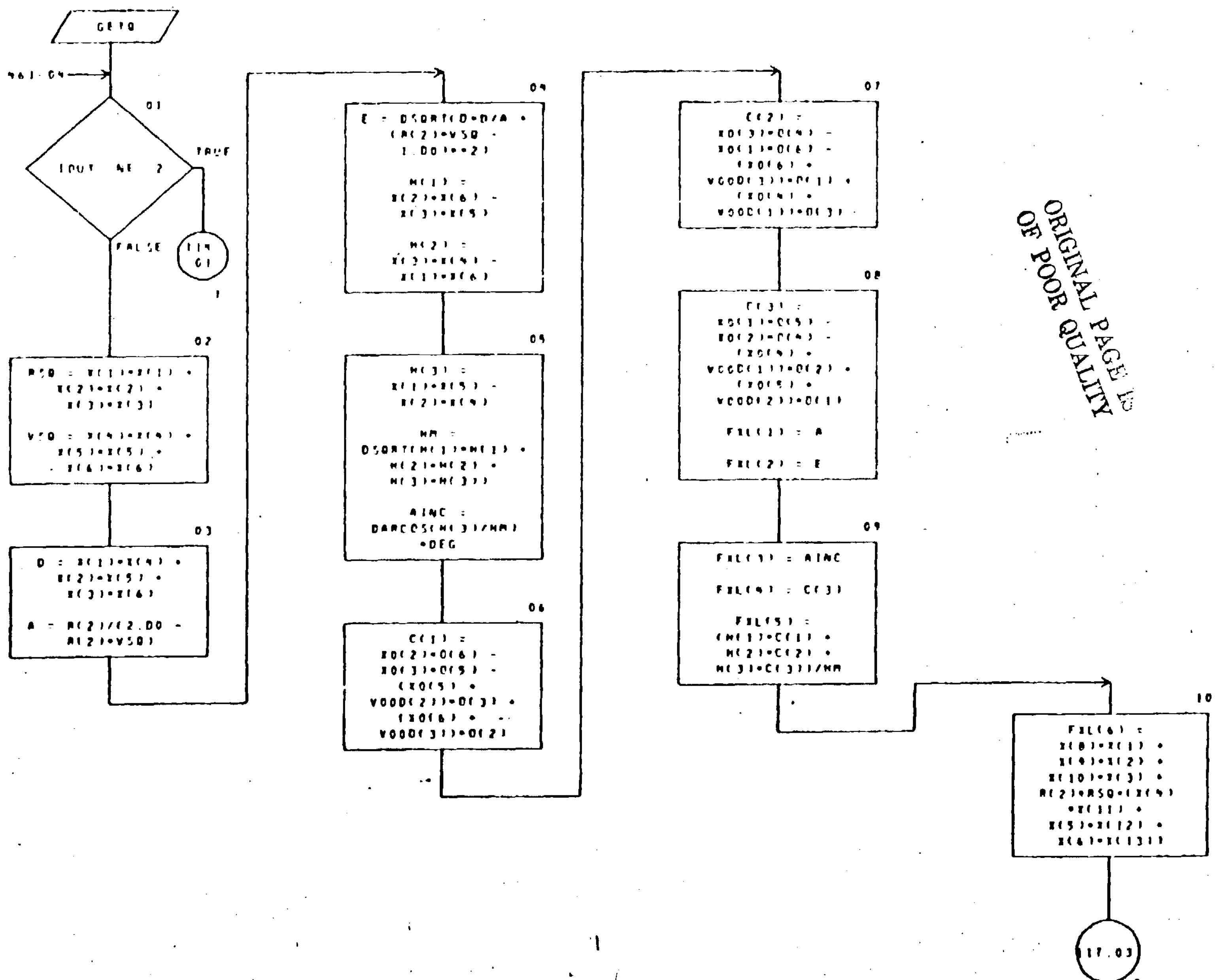


CHART TITLE - SUBROUTINE GETQ

ORIGINAL PAGE IS
OF POOR QUALITY

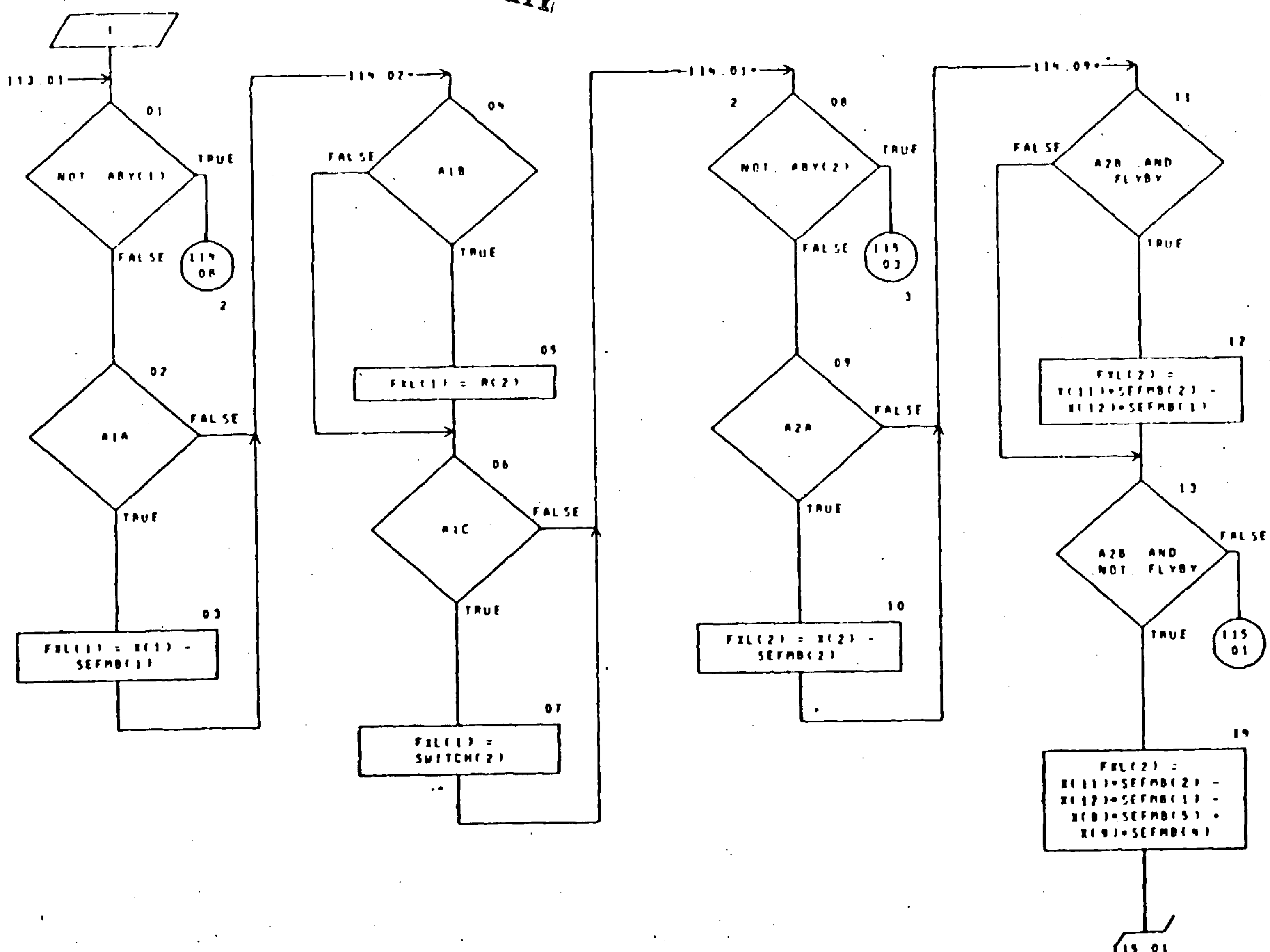


CHART TITLE - SUBROUTINE GETO

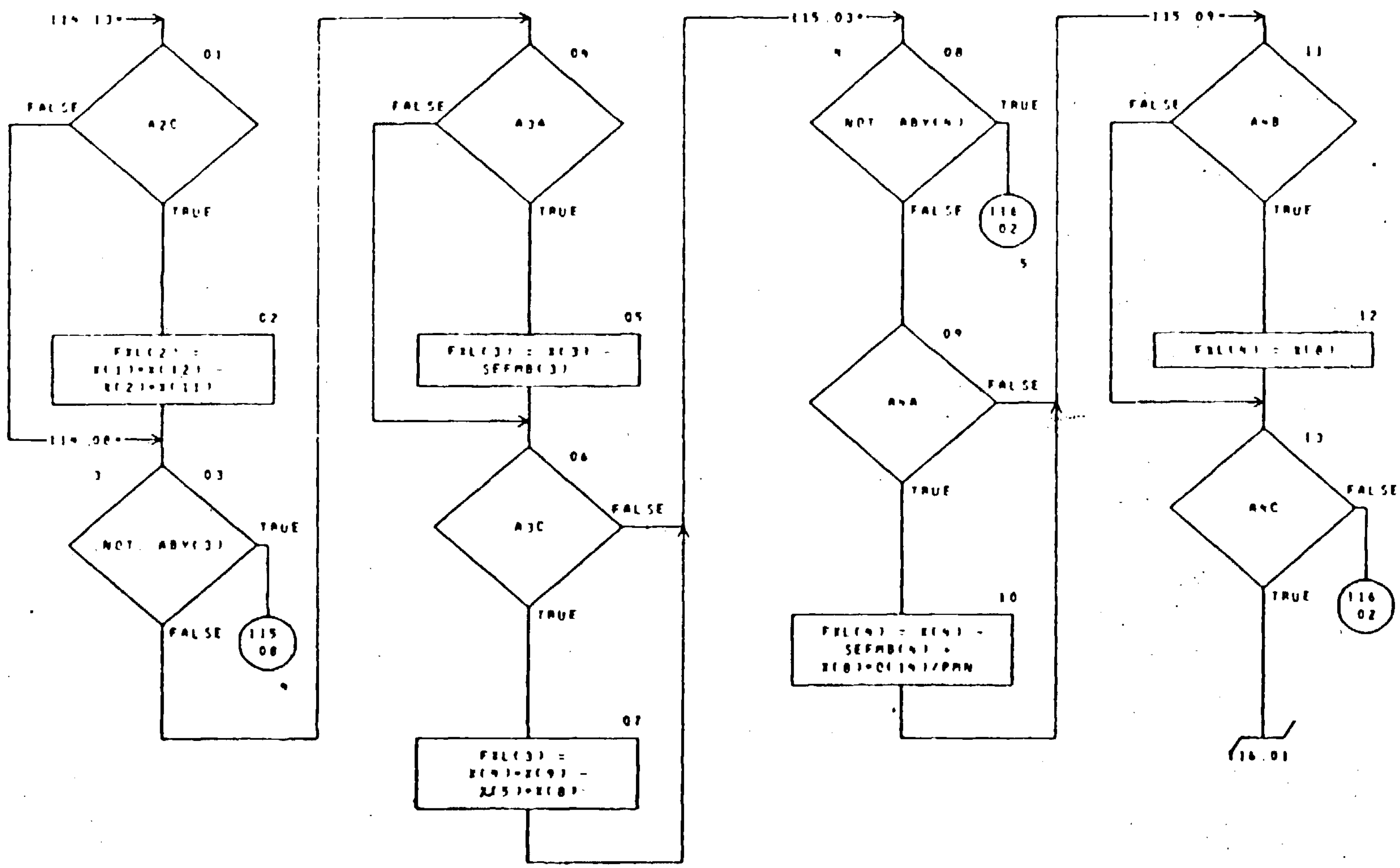


CHART TITLE - SUBROUTINE GETO

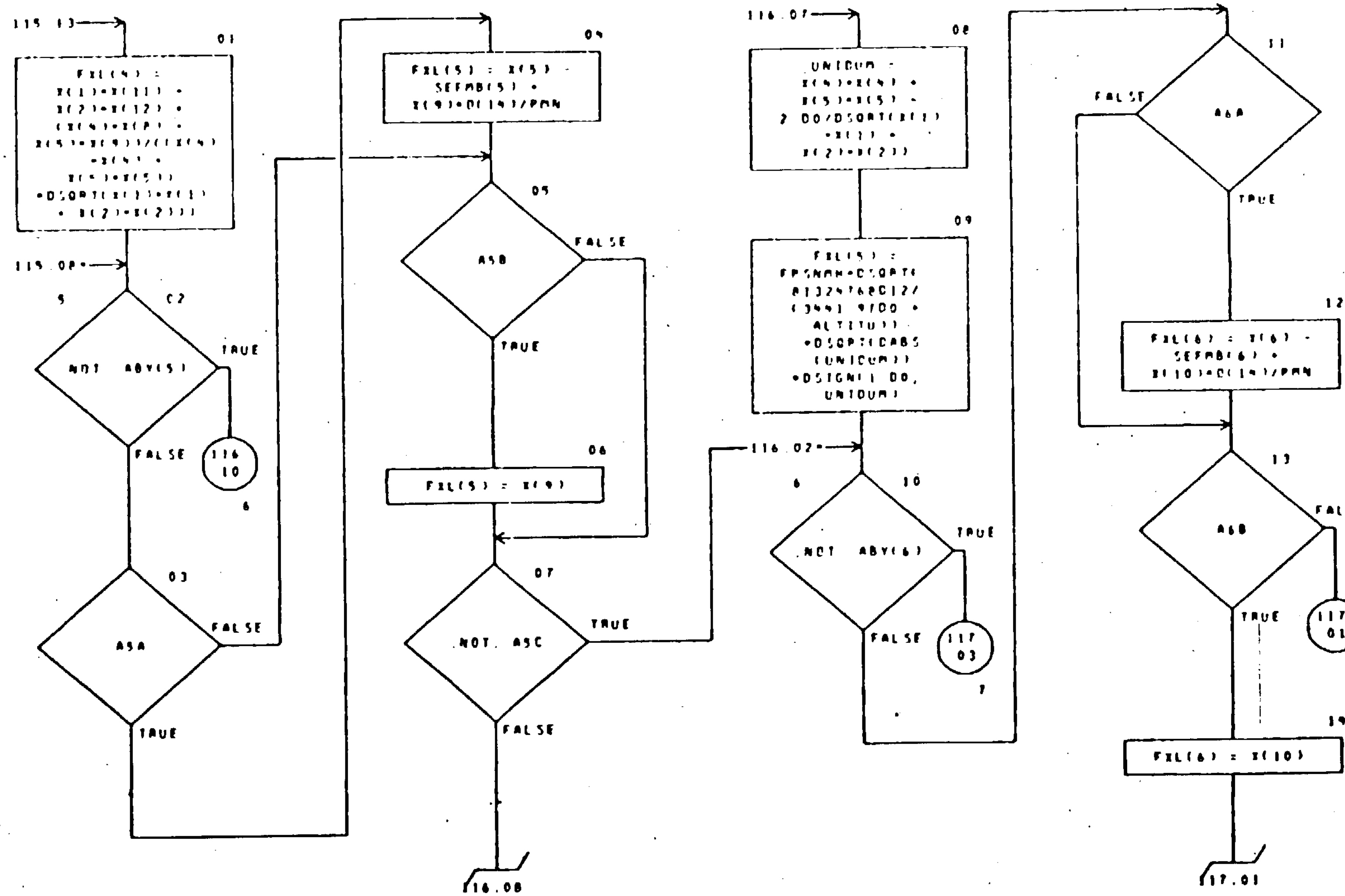


CHART TITLE - SUBROUTINE GETO

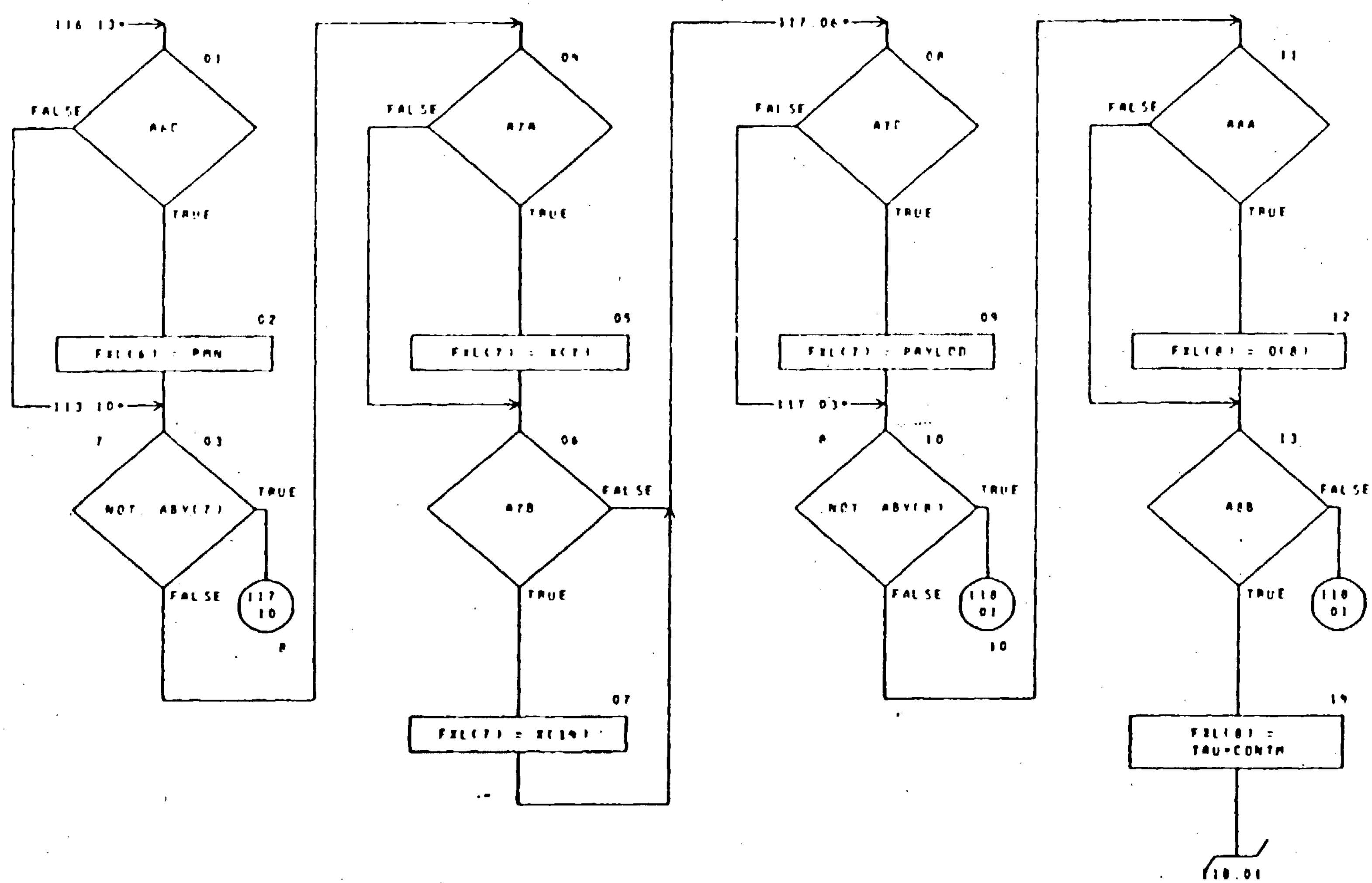
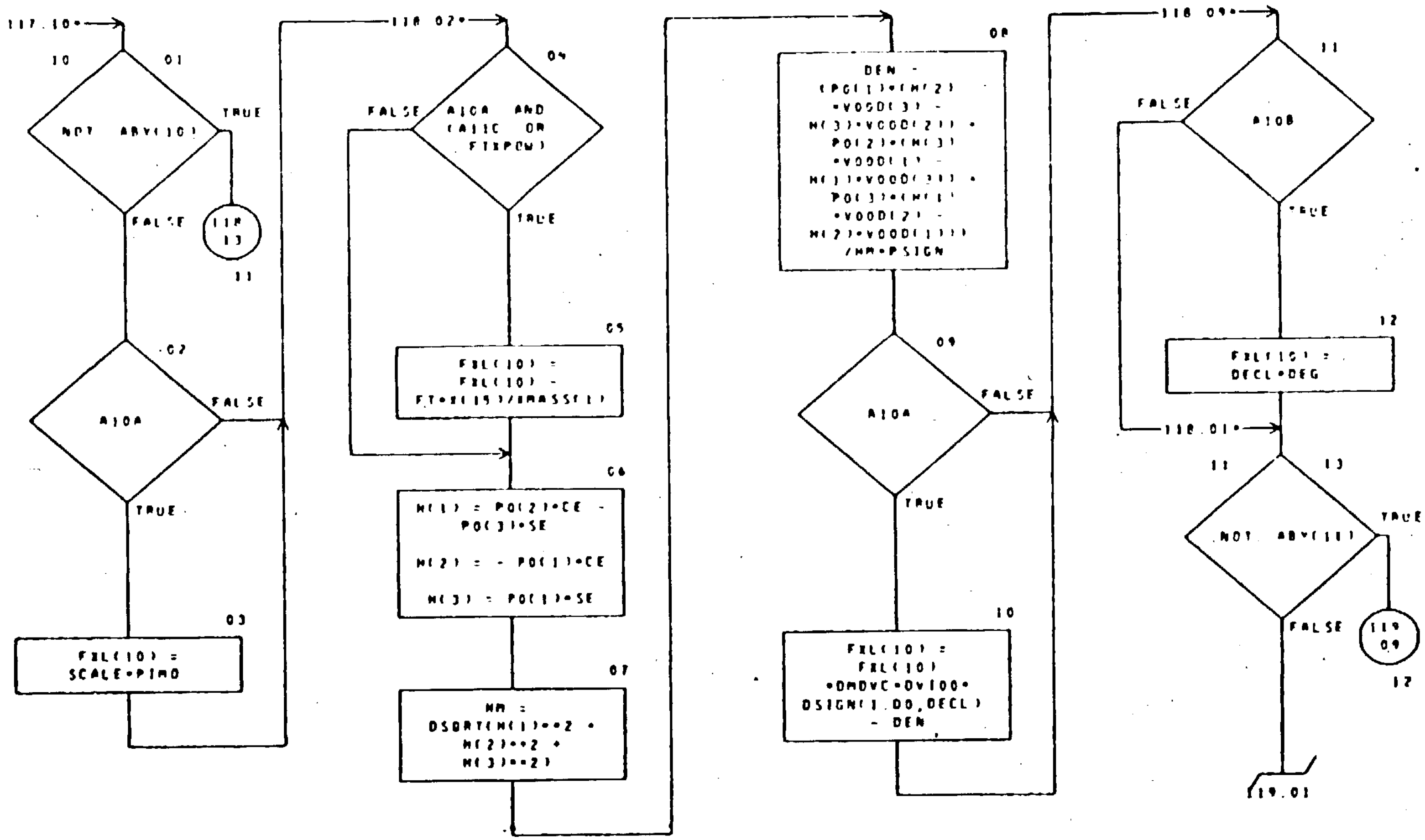


CHART TITLE - SUBROUTINE GETQ



GETQ-23

CHART TITLE - SUBROUTINE GFTO

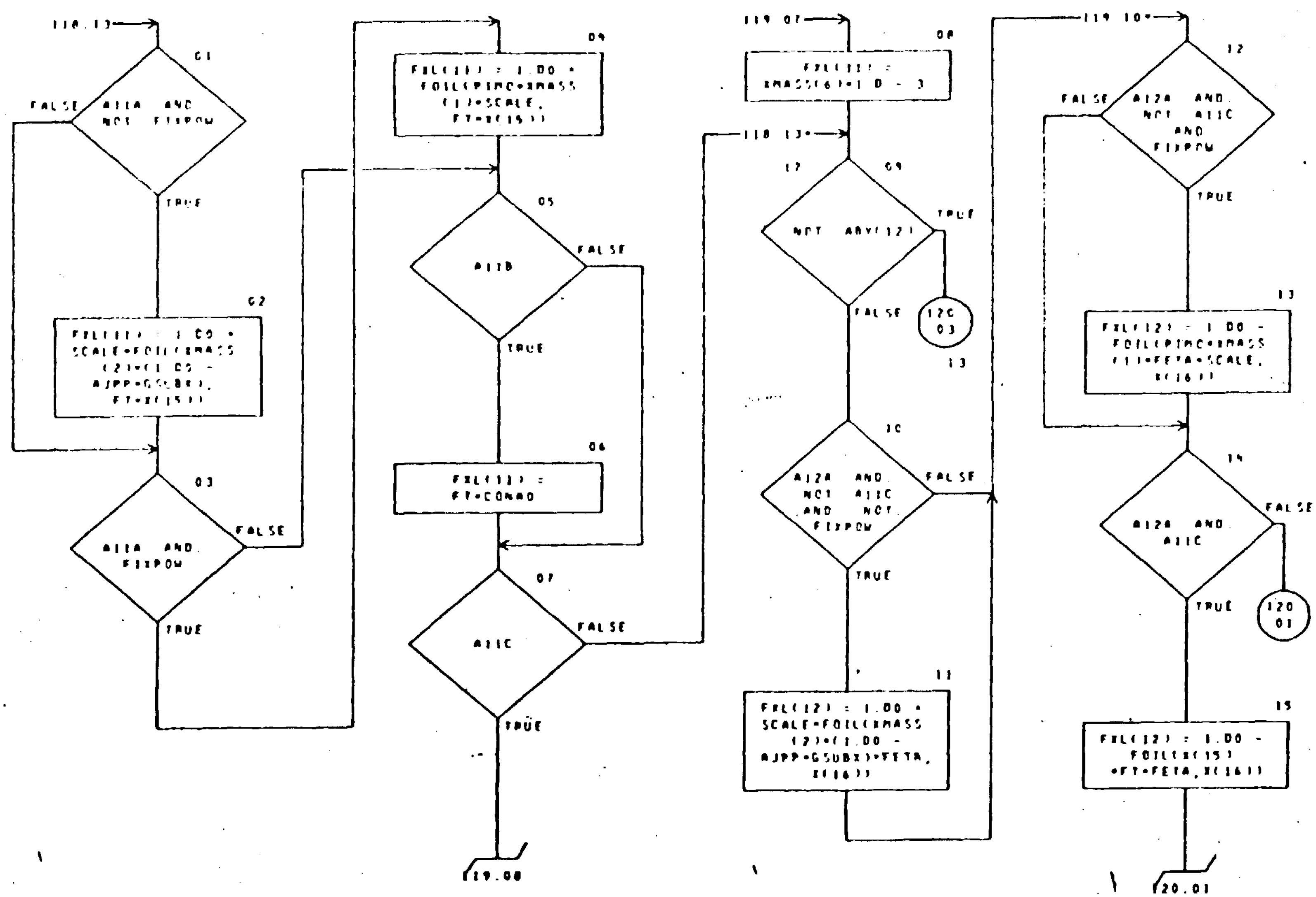
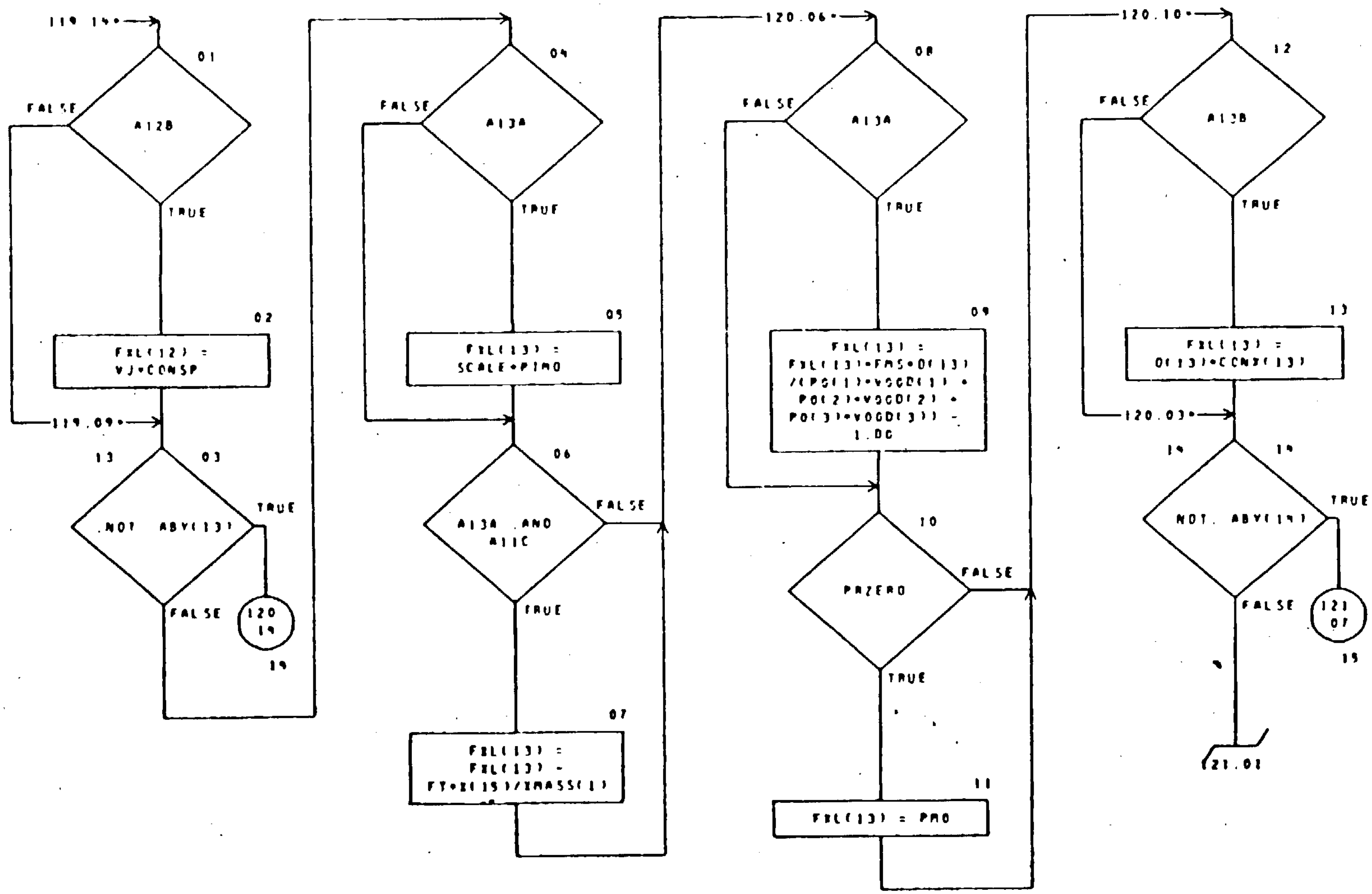


CHART TITLE - SUBROUTINE GETQ



GETQ-25

CHART TITLE - SUBROUTINE GFTO

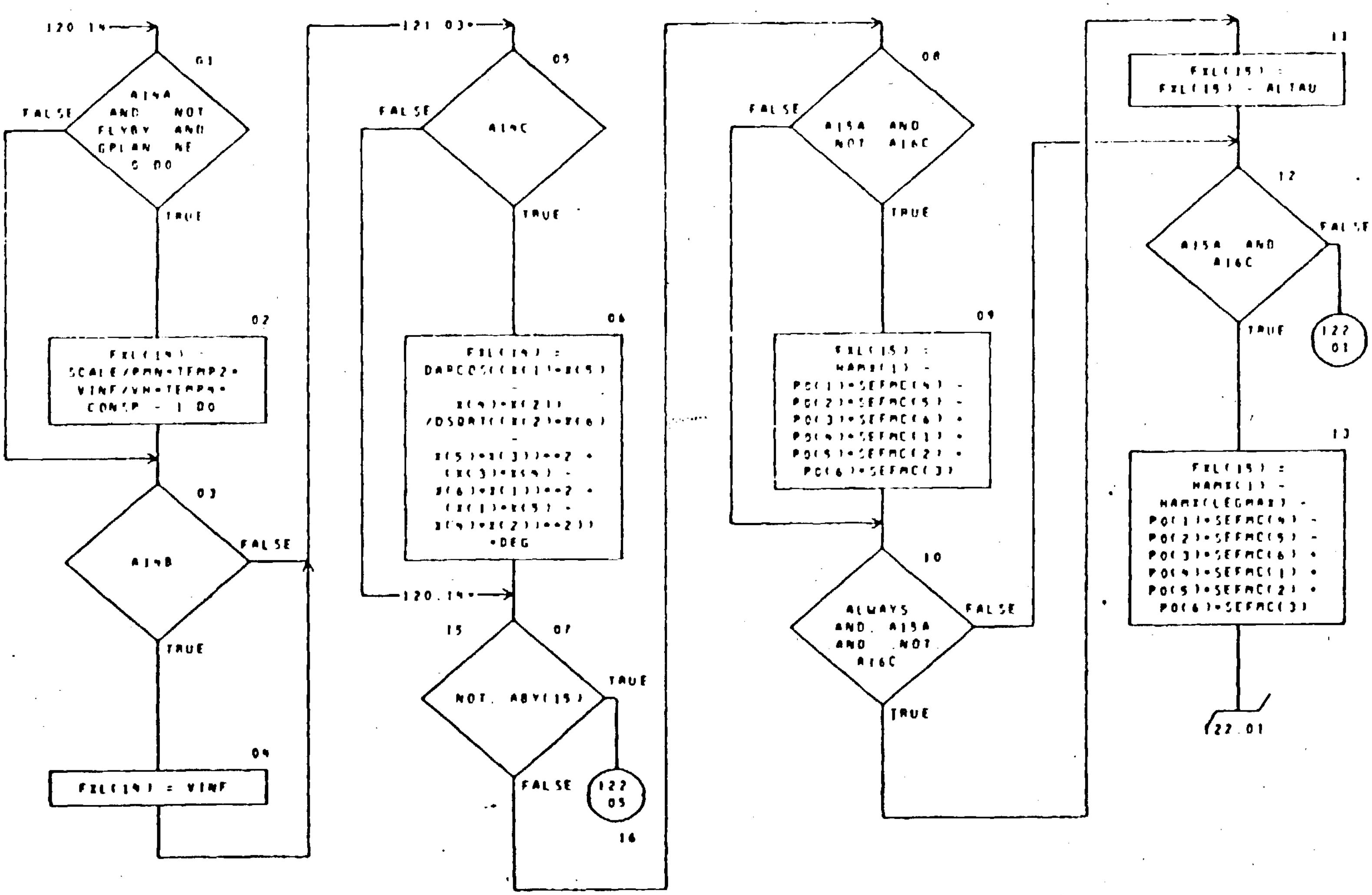


CHART TITLE - SUBROUTINE GETQ

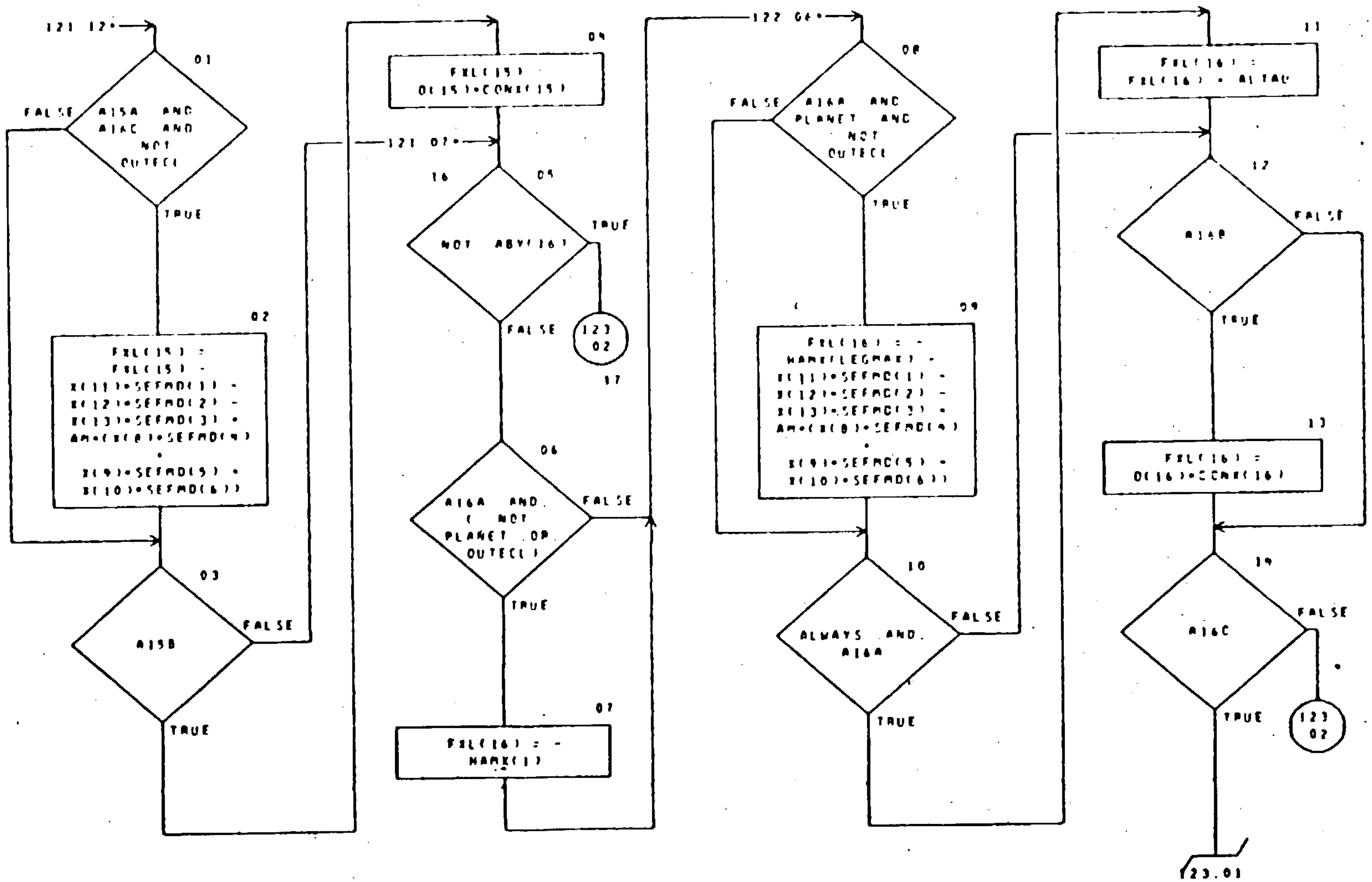


CHART TITLE - SUBROUTINE GETO

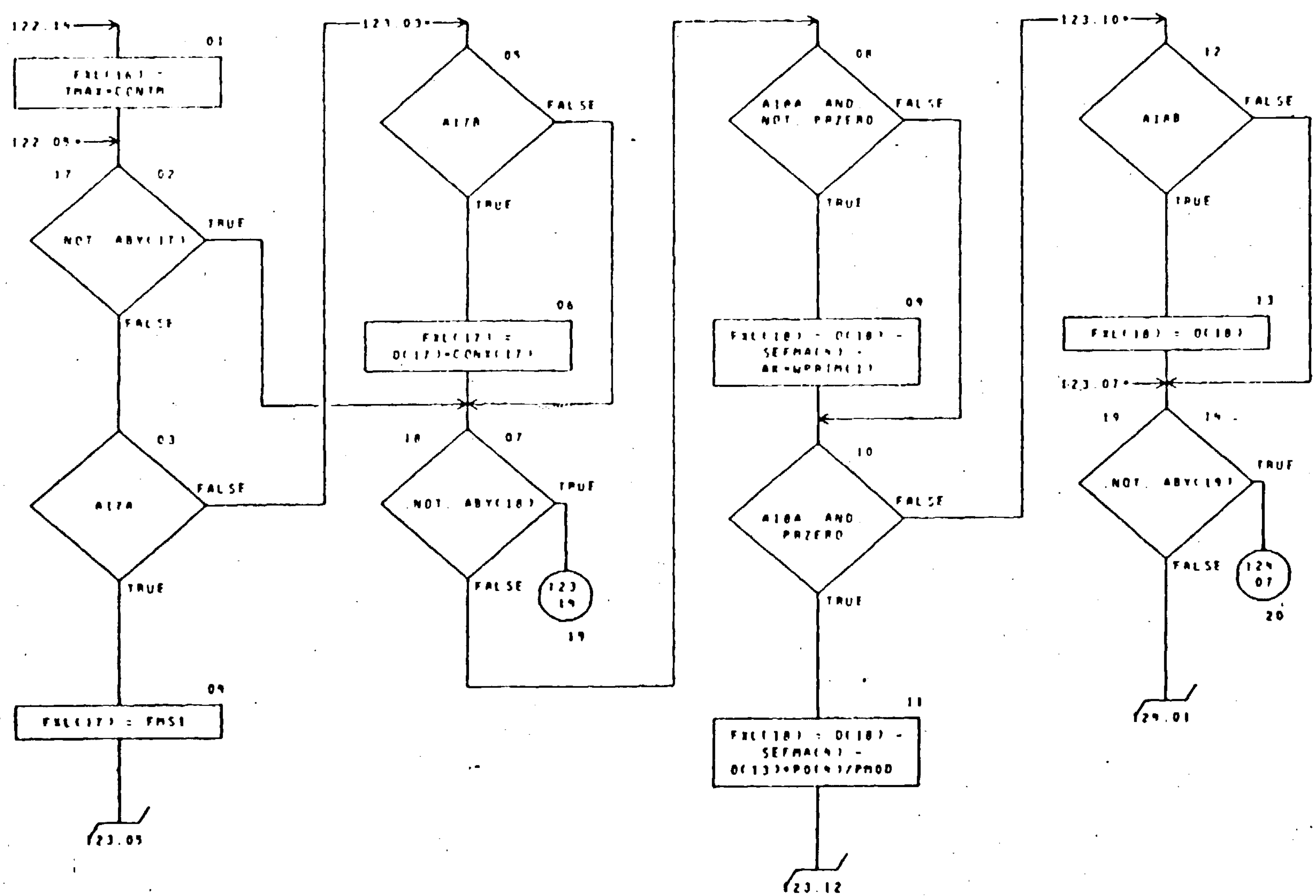


CHART TITLE - SUBROUTINE GETQ

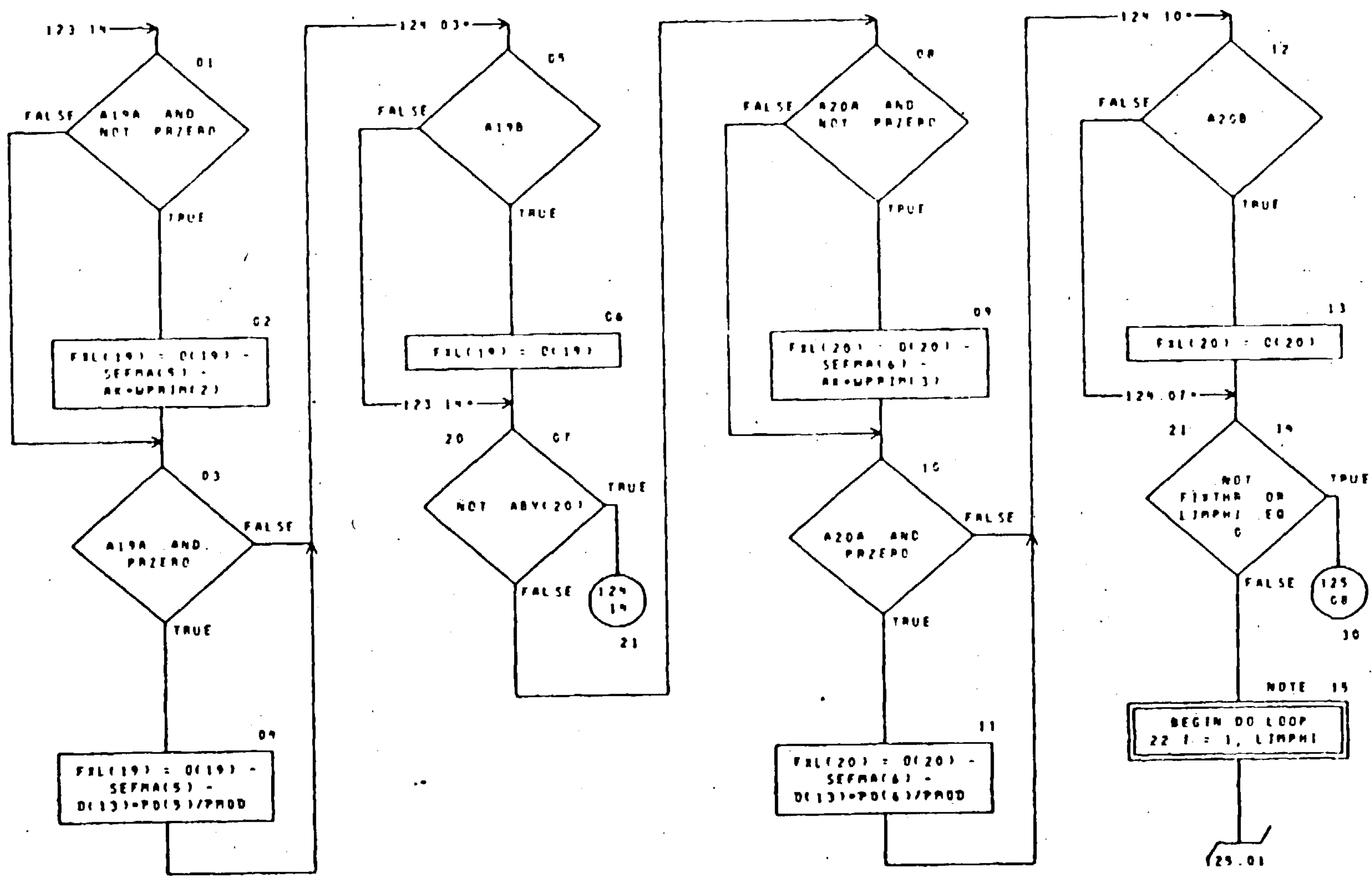


CHART TITLE - SUBROUTINE GETO

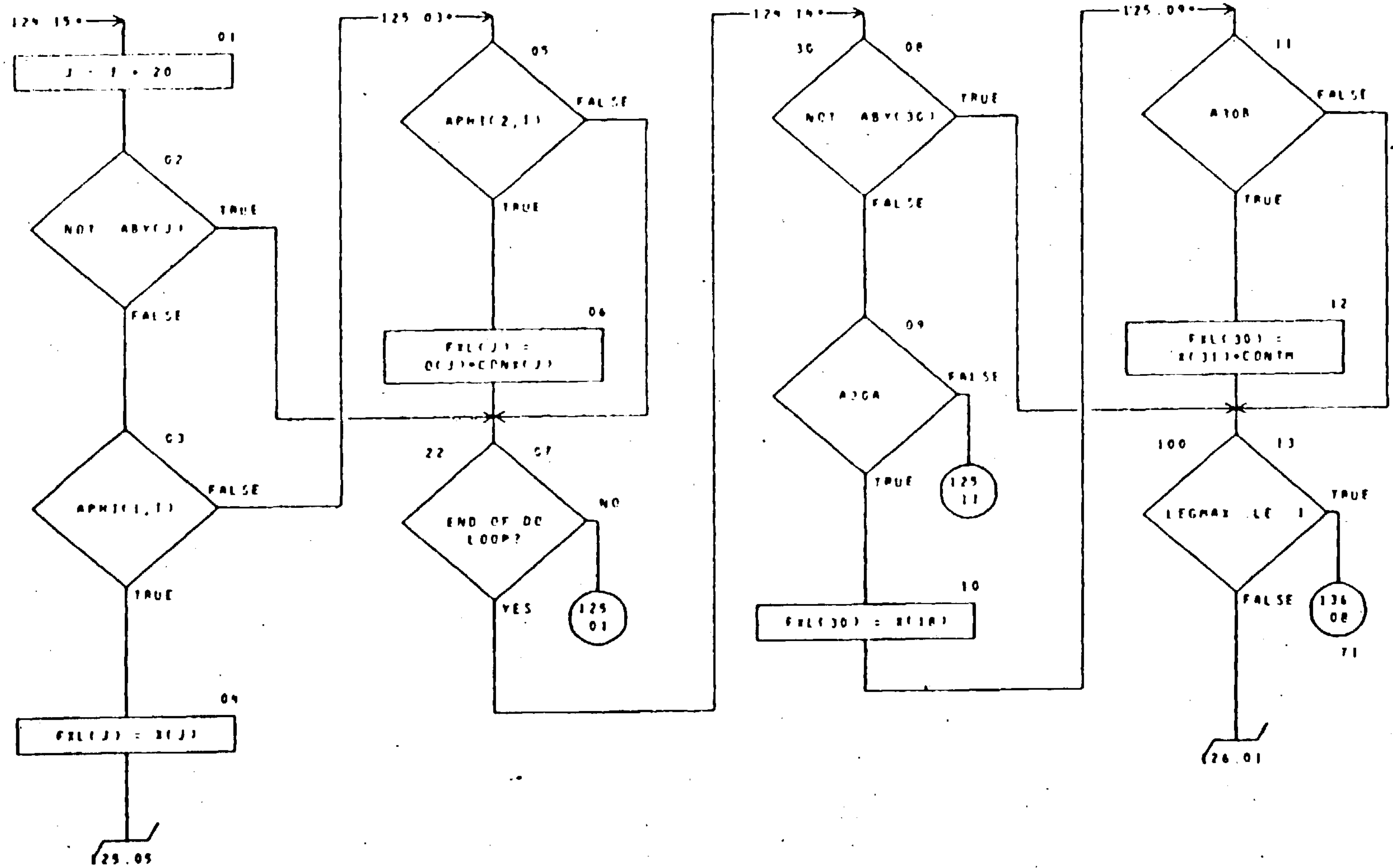
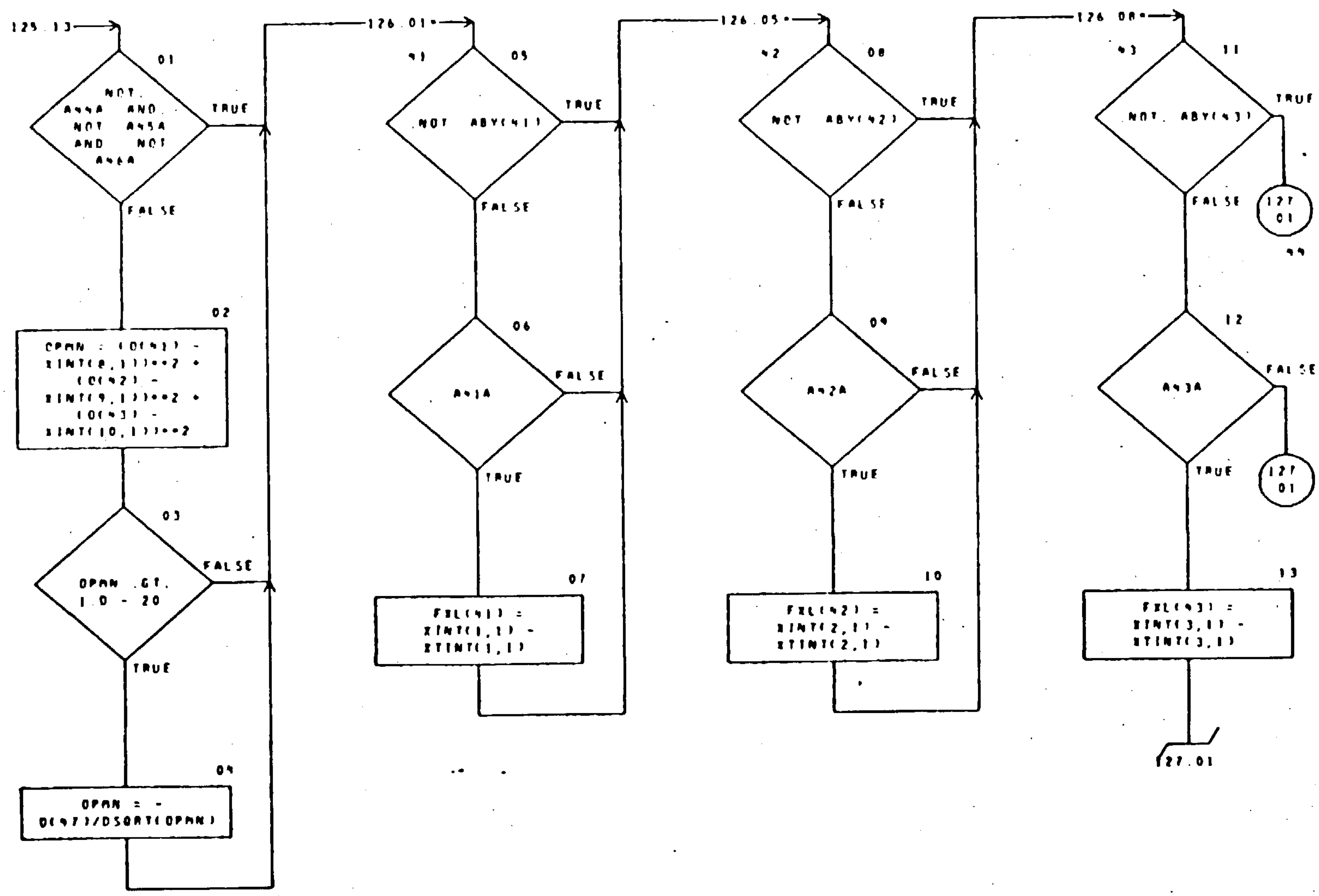


CHART TITLE - SUBROUTINE GETQ



GETQ-31

CHART TITLE - SUBROUTINE GETQ

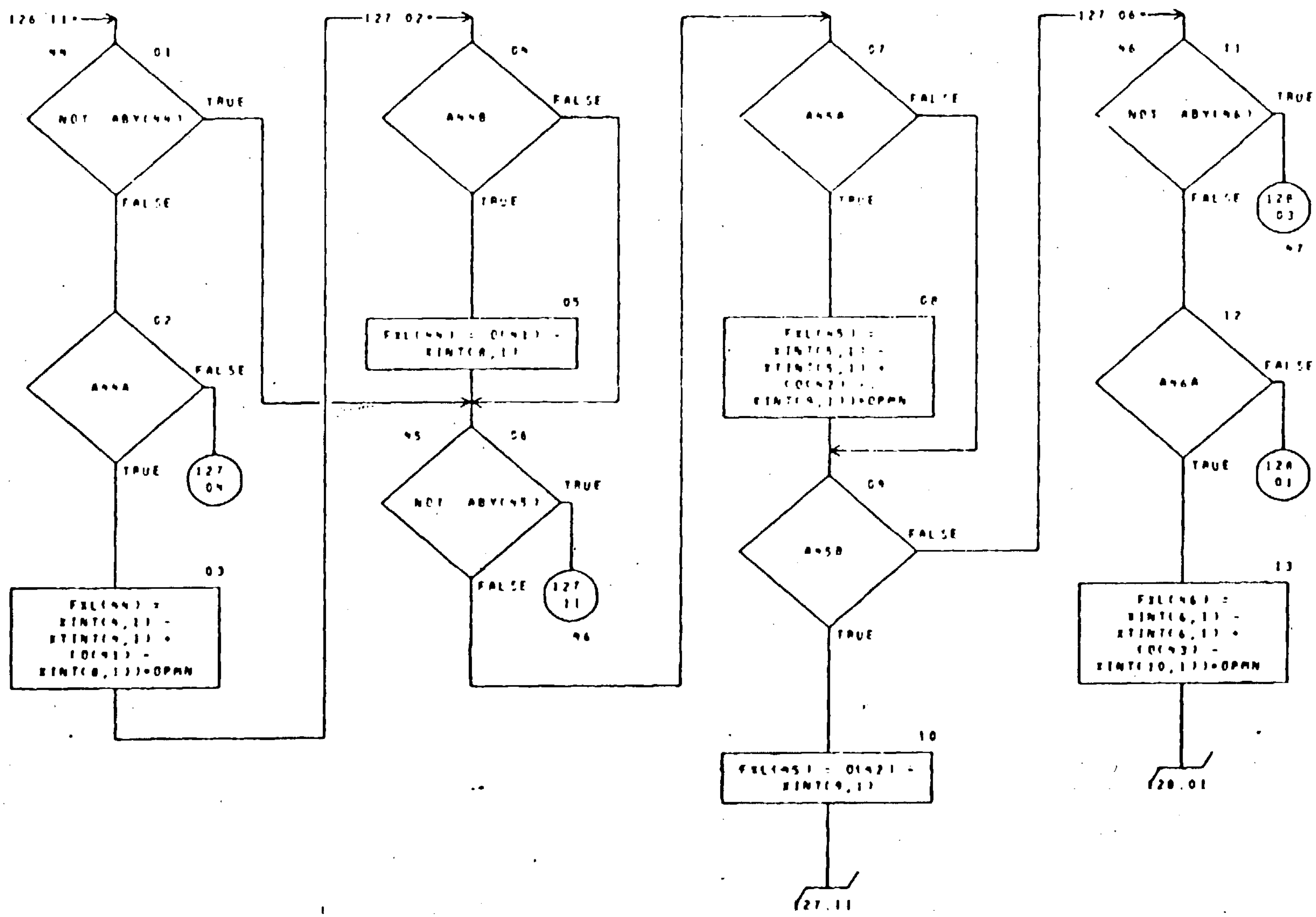


CHART TITLE - SUBROUTINE GETQ

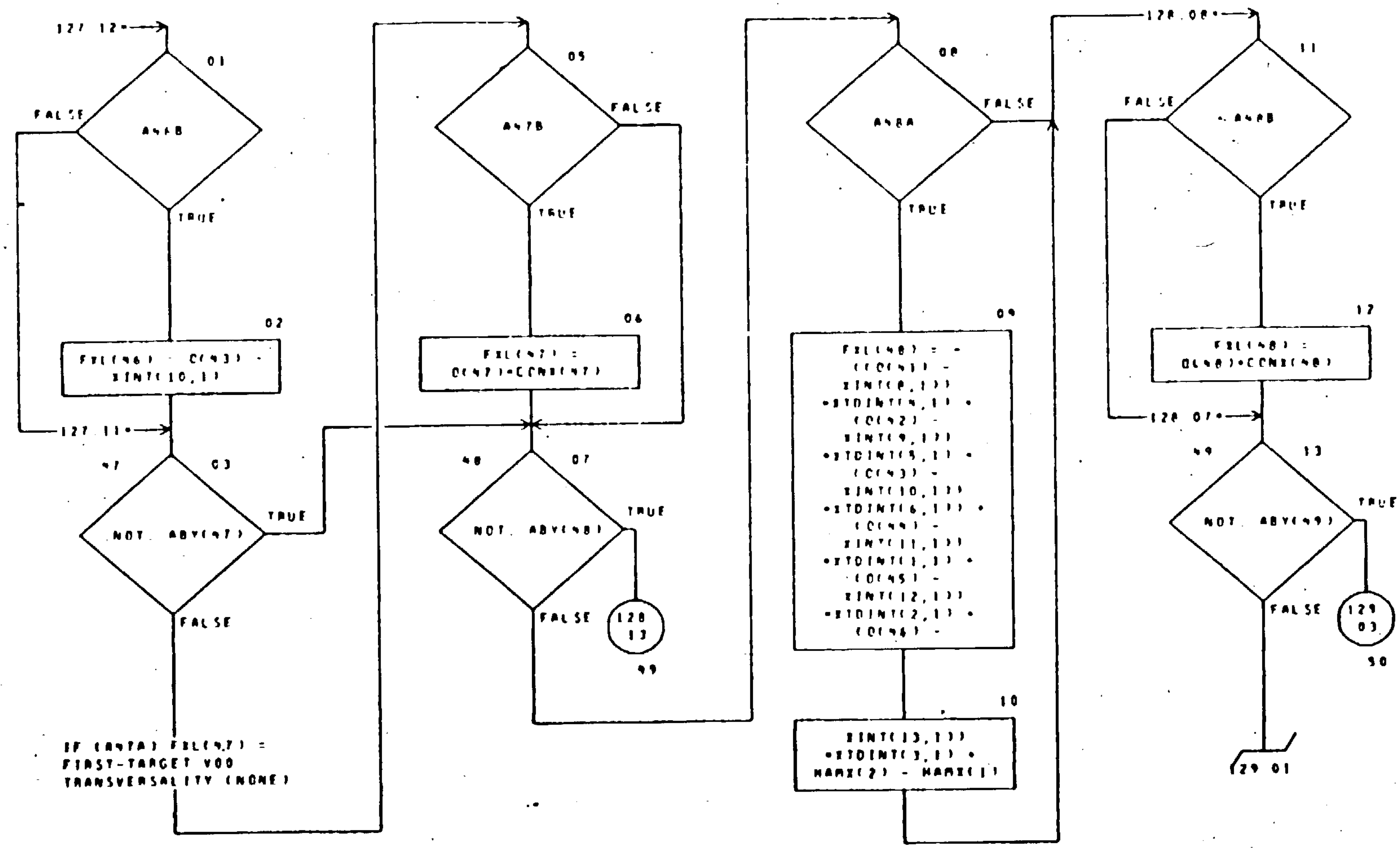


CHART TITLE - SUBROUTINE GETQ

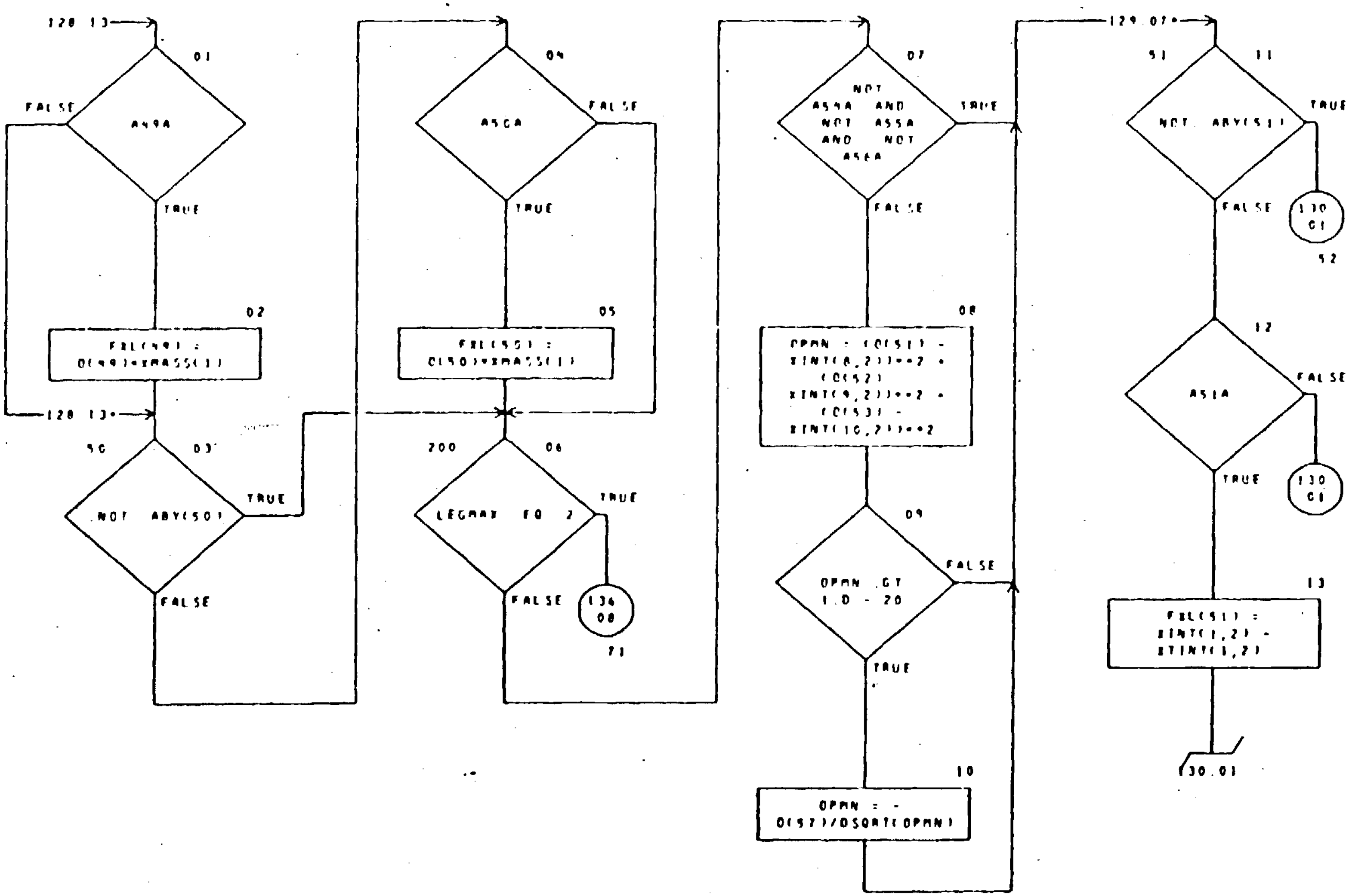


CHART TITLE - SUBROUTINE GETQ

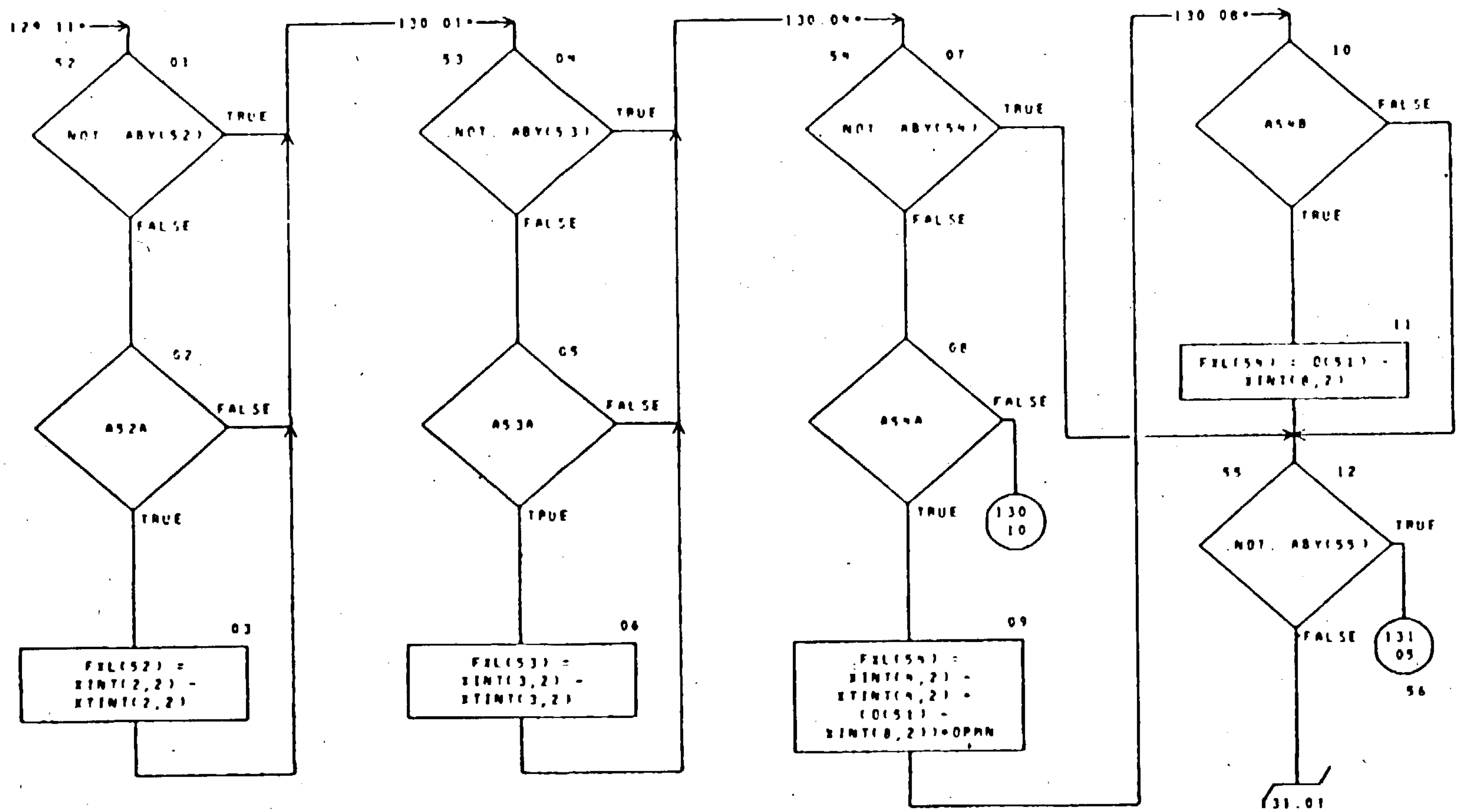


CHART TITLE - SUBROUTINE GETO

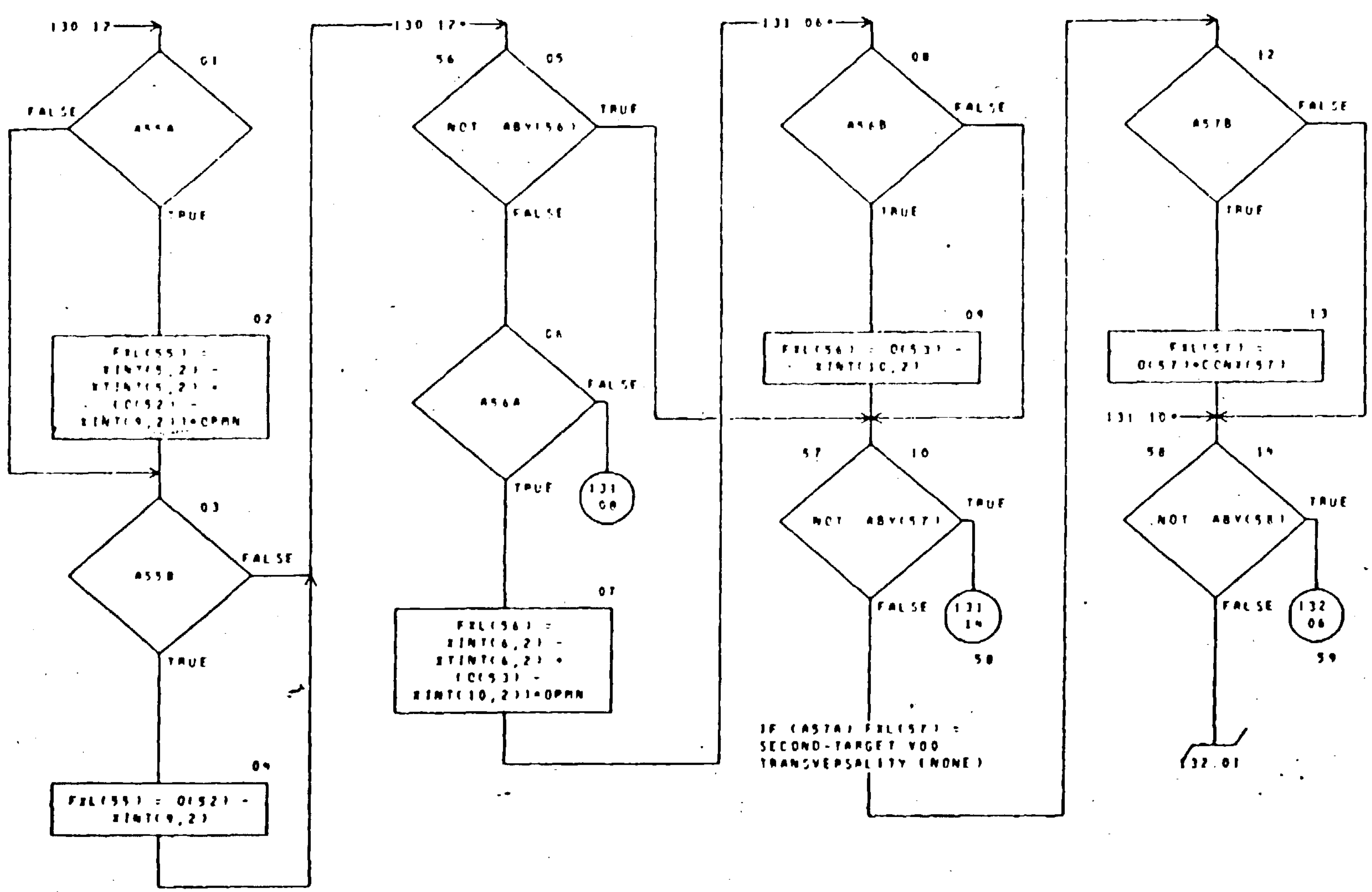


CHART TITLE - SUBROUTINE GETQ

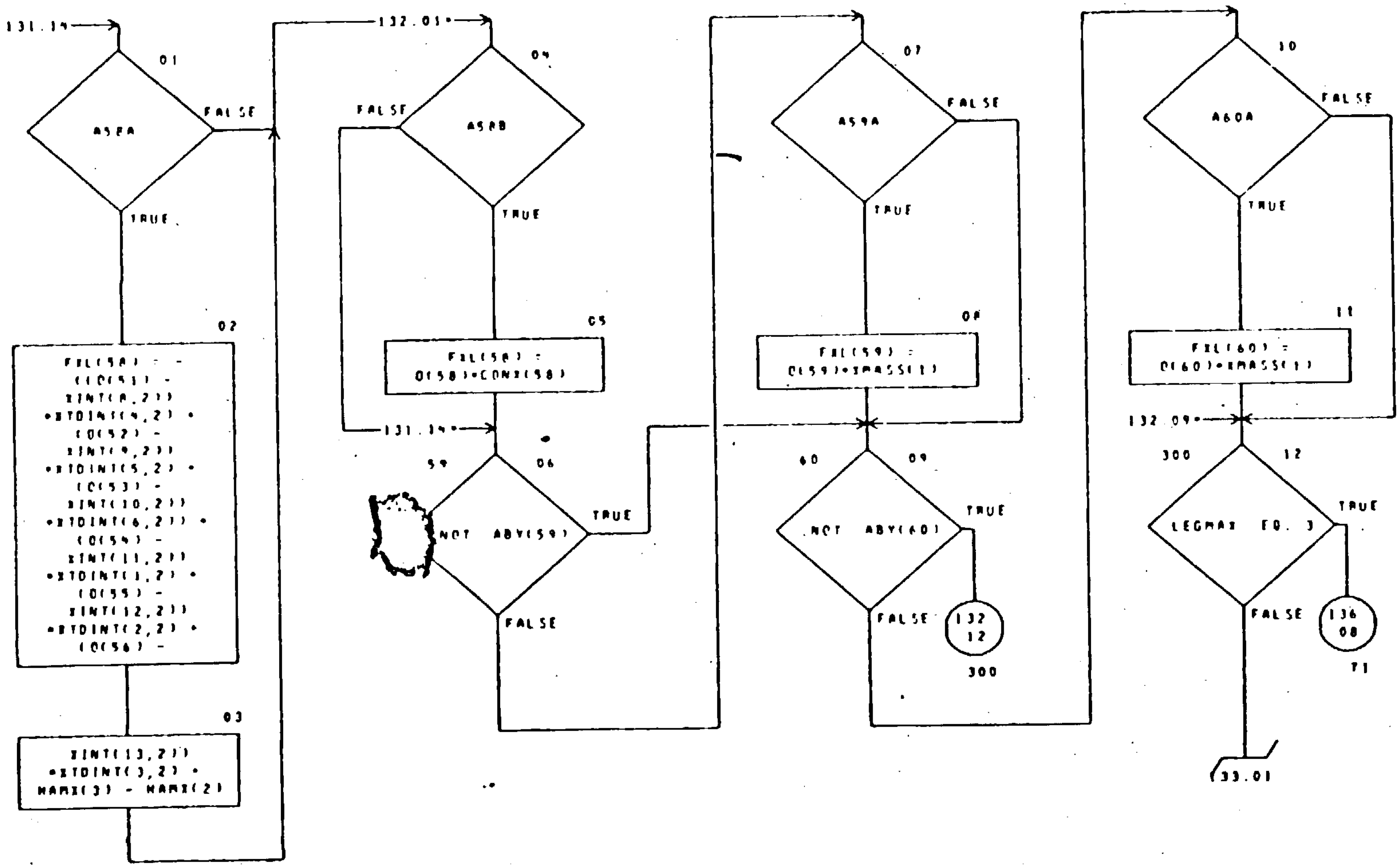


CHART TITLE - SUBROUTINE GETQ

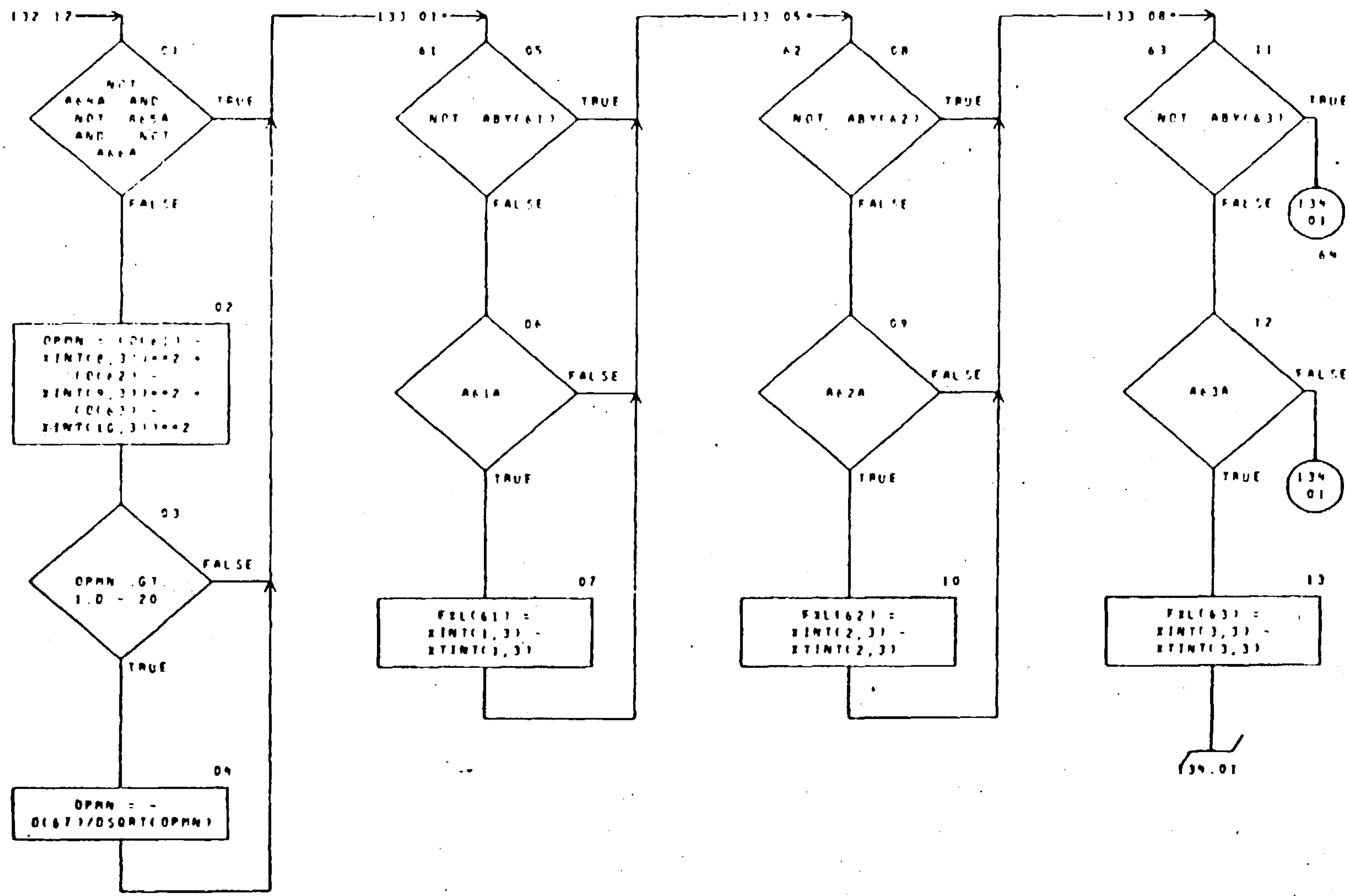
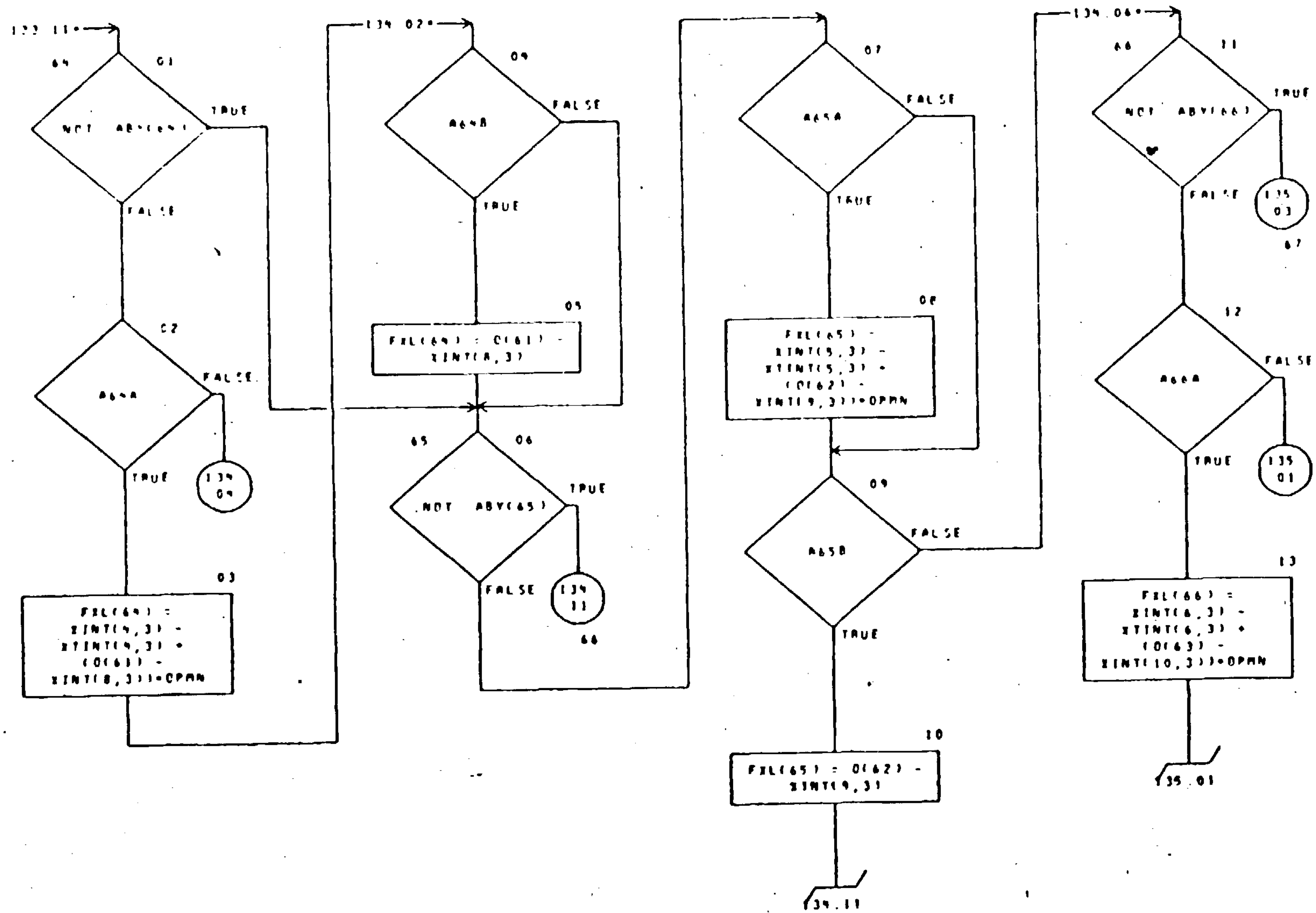


CHART TITLE - SUBROUTINE GETQ



GETQ-39

CHART TITLE - SUBROUTINE GETQ

ORIGINAL PAGE IS
OF POOR QUALITY

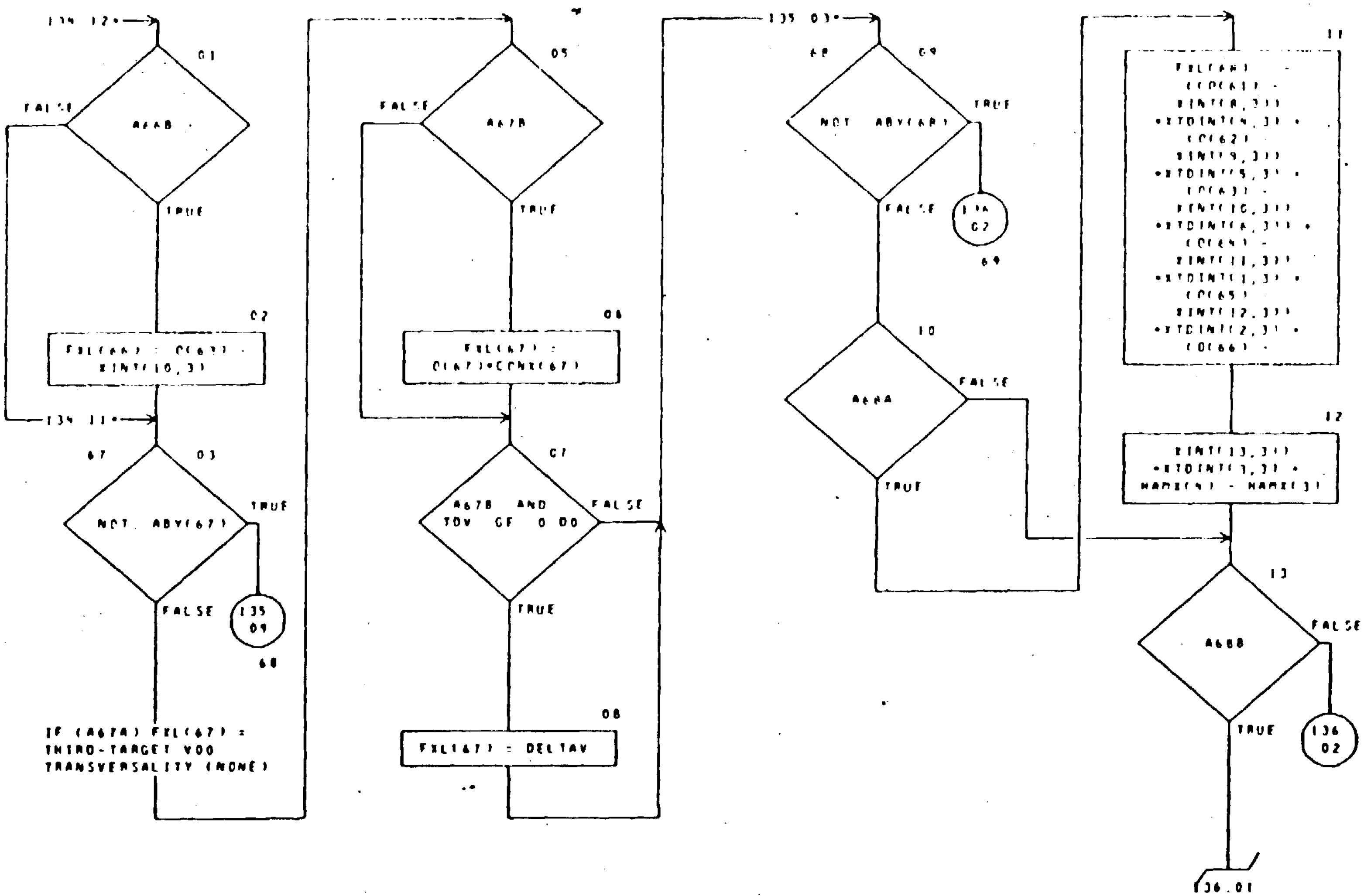
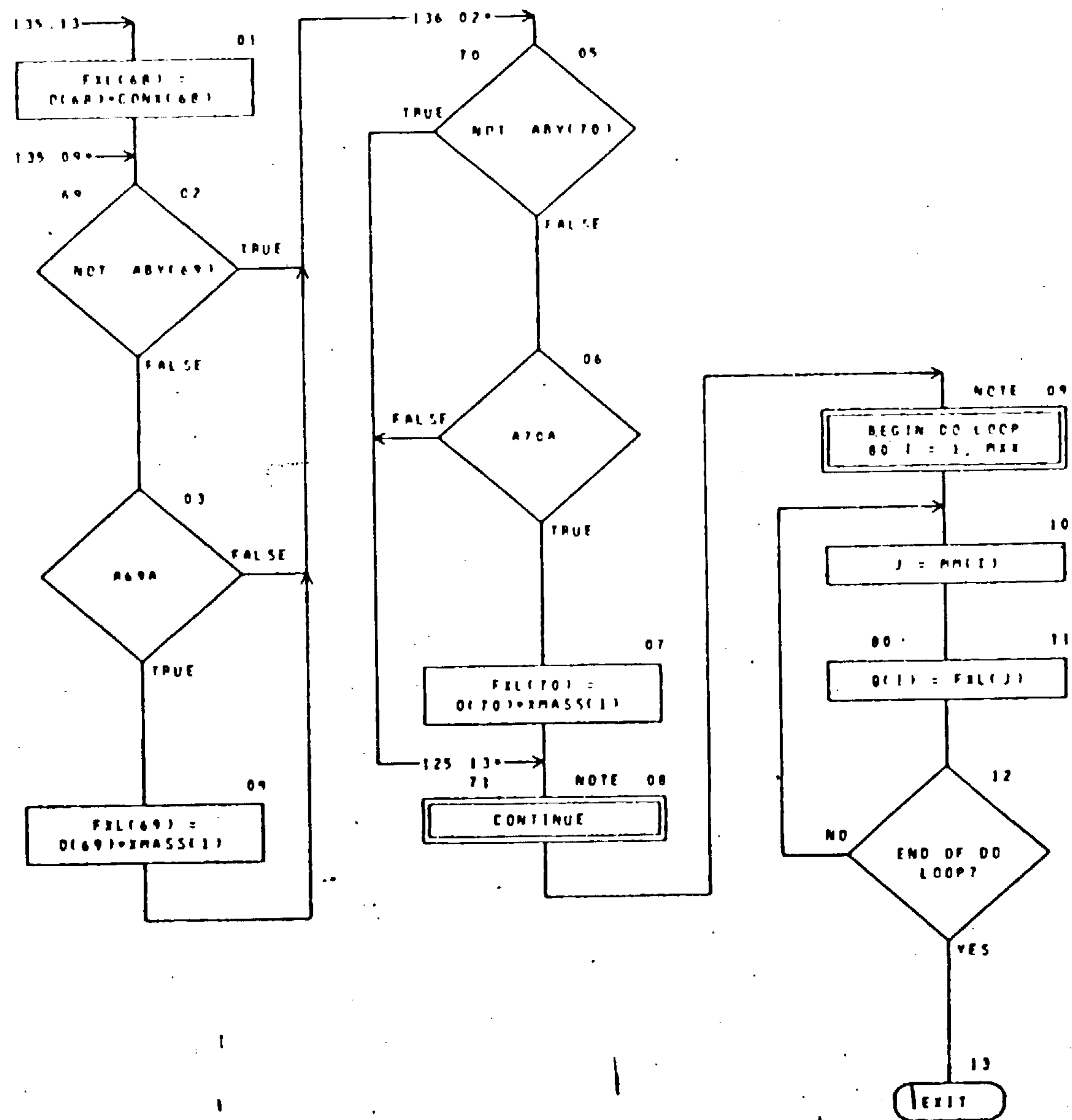


CHART TITLE - SUBROUTINE GETQ



GETQ-#1

CHART TITLE - NON-PROCEDURAL STATEMENTS


```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL FLYBY,FIXPOW,PRZERO,ABY,APHI,PLANET,FIXTHR,ALWAYS,OUTFCL
,A1A,A1B,A1C     ,A2A,A2B,A2C     ,A3A,A3C     ,A4A,A4B,A4C,A5A
,A5B,A5C     ,A6A,A6B,A6C,A7A,A7B,A7C,A8A,A8B,A11A,A11B,A11C,A12A
,A12B,A13A,A13B,A14A,A14B,A15A,A15B,A16A,A16B,A16C,A14C,A17A,A17B
,A18A,A18B,A19A,A19B,A20A,A20B,A41A,A42A,A43A,A44A,A44B,A45A,A45B
,A46A,A46B,A47A,A47B,A48A,A48B,A51A,A52A,A53A,A54A,A54B,A55A,A55B
,A56A,A56B,A57A,A57B,A58A,A58B,A61A,A62A,A63A,A64A,A64B,A65A,A65B
,A66A,A66B,A67A,A67B,A68A,A68B,A49A,A50A,A59A,A60A,A69A,A70A,A30A
,A30B,A10A,A10B,GLI
DIMENSION H(3),C(3)
COMMON /REAL8/ PAYLOD,XMASS(7),PGI,CTANK CSTR,FT,VJ,R02(13),
VLOSS,R03(23),XG(7),PO(7),R04(24),TMAX,R05(11),TDV,R33(24),
SEFMA(7),SEFMB(7),
SEFMC(7),SEFMD(7),R06(17),SE,CE,DECL,R32(2),ALTAU,R30(10),
GPLAN,R07(13),R(2),R08(2),SWITCH(2),R09(97),VOOD(3),PIMO,PSIGN,
R31(4),  DEG,FPSNMH,R10,CONTM,CONSP,CONAO,R11(6),DELTAV,
R34(374),ALTITU,
R12(196),HAMX(4),R13(290),SAMPS,DROPS,R29(3),
PMN,R14(5),FMS,R15(6),DMOVC,AM,R16(2),QCST,VH,
R17,VINF,R18(6),FMSI,R19(2),AK,R20,PMO,PMOO,R21,GSUBX,TEMP2,
R28  ,TEMP4,AJT,AJPP,FETA,SCALE,R22(6),WPRIM(3),R23(53),X(50),
R24(53),TAU,R25(21),XINT(50,5),R26(250),XTINT(6,5),XTDINT(6,5),
R27(65)
COMMON /INTGR4/I01(48),LIMPHI,I02(13),MXX,I03(10),IOUT,I06(53),
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

ORIGINAL PAGE IS
OF POOR QUALITY

```
LEGMAX,IO5(127),MM(70),IO4(675)
COMMON /LOGIC4/ L01(2),FIXPOW,L02(6),OUTECL,FIXTHR,L03,FLYBY,
L04(12),PLANET,L05(7),PRZERO,L06(7),ALWAYS,L10(108),
A1A,A1B,A1C,A2A,A2B,A2C,A3A,A1C,A4A,A4B,A4C,A5A,A5B,A5C,A6A,A6B,
A6C,A7A,A7B,A7C,A8A,A8B,A11A,A11B,A11C,A12A,A12B,A13A,A13B,A14A,
A14B,A14C,A15A,A15B,A16A,A16B,A16C,AFN1(2,15),A17A,A17B,A18A,A18B,
A19A,A19B,A20A,A20B,A41A,A42A,A43A,A44A,A44B,A45A,A45B,A46A,A46B,
A47A,L07(70),ABY(70),L08(35),A47B,A48A,A48B,A51A,A52A,A53A,A54A,
A54B,A55A,A55B,A56A,A56B,A57A,A57B,A58A,A58B,A61A,A62A,A63A,A64A
,A64B,A65A,A65B,A66A,A66B,A67A,A67B,A68A,A68B,A49A,A50A,A59A,A60A
,A69A,A70A,A30A,A30B,A10A,A10B,L09(6))
COMMON /ITERAT/ B01(560),CONT(70),B02(70),D(70),B03(70),FXL(70),
B04(70)
COMMON /ITER2/ P01(35),Q(35),P02(1970)
COMMON /GUNCOM/ G01(6),DV100,GL1(2)
STATEMENT FUNCTION DEFINITION,  FOIL(A1,A2) = DSIGN(A1,A2)/DMAX1(1 D-20,DABS(A2))
```

GETQ-43

Name: GET RV

Calling Argument: V1, V2

Referenced Sub-programs: INCOND, VCROSS, VSUB

Referenced Commons: ITERAT, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: TRAJ

Discussion: This routine computes the desired final position and velocity R_e and $\dot{V}_e$ ("get RV") pertaining to extra-ecliptic missions. Four of the final six desired orbital elements are essentially specified as follows: arrival occurs at perihelion (flight path angle $\gamma = 0$), and inclination i_f, speed v_f and radial distance r_f are specified. The ascending node angle Ω and argument of perihelion ω are optimized via the transversality conditions as follows:

$$\Omega = \tan^{-1}\left[(C \cdot \bar{j})/(C \cdot \bar{i})\right] ,$$

$$\omega = \tan^{-1}\left[(-\bar{h} \cdot \dot{\Lambda})\, r_f/(\bar{h} \cdot \Lambda)\, v_f\right] ,$$

where $$\bar{h} = \cos i_f \bar{k} + \sin i_f (\bar{k} \times C)/|\bar{k} \times C| ,$$

and where the primer Λ and its derivative $\dot{\Lambda}$ are evaluated at the final time and $\bar{i}$, $\bar{j}$, and $\bar{k}$ are the coordinate unit vectors. C is the vector constant of the motion (which, without loss of generality, may be evaluated at the initial time):

$$C = R_o \times \dot{\Lambda}_o - \dot{R}_o \times \Lambda_o .$$

Since

$$R_o = P_o ,$$

$$\dot{R}_o = \dot{P}_o + V_\infty ,$$

PRECEDING PAGE BLANK NOT FILMED

where P_o is the launch planet's position and V_∞ is the departure excess velocity, this vector constant of the motion is actually evaluated:

$$C = P_o \times \dot{\Lambda}_o - \dot{P}_o \times \Lambda_o - V_\infty \times \Lambda_o .$$

The last term is not computed when the magnitude of the departure asymptote declination is less than the parking orbit inclination, since then V_∞ is aligned with Λ_o and the term has value zero.

Finally, subroutine INCOND generates the desired final position and velocity:

$$R_e = r_f \begin{bmatrix} c\,\omega\, c\,\Omega - s\,\omega\, s\,\Omega\, c\,i_f \\ c\,\omega s\,\Omega + s\,\omega\, c\,\Omega\, c\,i_f \\ s\,\omega\, s\,i_f \end{bmatrix}$$

$$V_e = v_f \begin{bmatrix} -s\,\omega\, c\,\Omega - c\,\omega\, s\,\Omega\, c\,i_f \\ -s\,\omega\, s\,\Omega + c\,\omega\, c\,\Omega\, c\,i_f \\ c\,\omega\, s\,i_f \end{bmatrix}$$

where s and c denote sine and cosine, respectively.

GET RV EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
O(70)	U	ITERAT	Array of iterator independent variables; here, Λ_o and $\dot{\Lambda}_o$ are used.
X(50)	U	REAL8	Array of trajectory integrated variables; here, R and $\dot{R}$ of the spacecraft, at the final time, are used.
AI	A	REAL8	Desired final extra-ecliptic inclination, i_f, in radians.

GET RV EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
AR	UA	REAL8	Desired final spacecraft solar distance, r_f, in AU.
AV	UA	REAL8	Desired final spacecraft speed, v_f, in EMOS.
CI	U	REAL8	Cos i_f.
SI	U	REAL8	Sin i_f.
V1(3)	SUAX		Desired final spacecraft position, R_e, in AU.
V2(3)	SUAX		Desired final spacecraft velocity, V_e, in EMOS.
X0(7)	U	REAL8	Launch planet position and velocity at initial time, P_o and $\dot{P}_o$ in AU and EMOS, respectively.
V0OD(3)	U	REAL8	Launch hyperbolic excess velocity, V_∞, in EMOS.
LDECL	U	LOGIC4	Indicator for the condition in which the magnitude of the departure asymptote declination exceeds the parking orbit inclination.

CHART TITLE - SUBROUTINE GETRV(V1,V2)

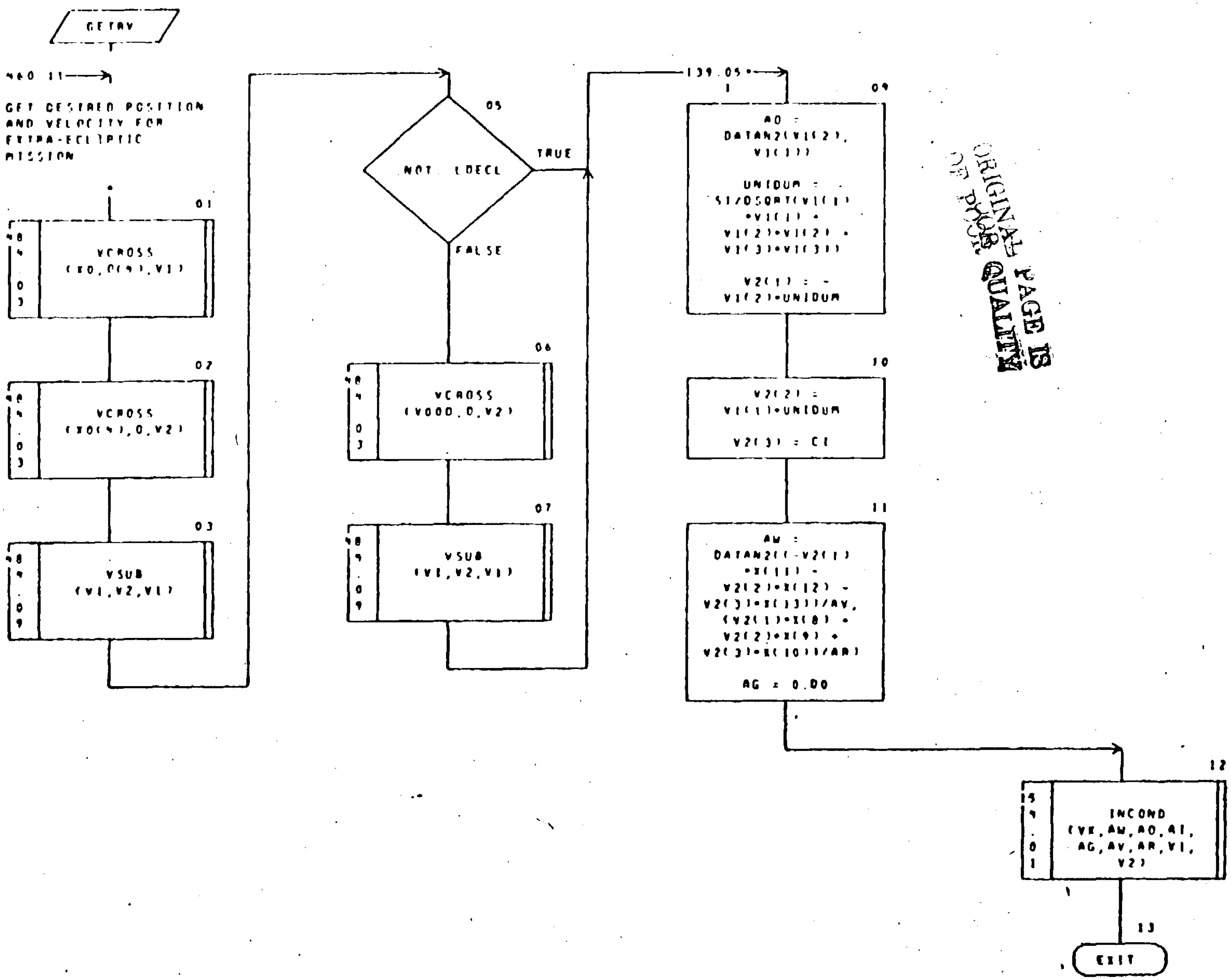


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL LDECL
DIMENSION V1(3),V2(3),VK(3)
COMMON /REAL8/ RO1(50),XO(7),RC2(14),AR,AV,AT,SI,CI,RO3(227),
VOOD(3),RO4(944),X(50),RO5(705)
COMMON /LOGIC4/ LO1(42),LDECL,LO2(457)
COMMON /ITERAT/ BC1(700),C(70),BC2(215)
DATA VK /0 D0,0 D0,1 D0/
```

Name: GUESS

Calling Argument: STATE, MOPT

Referenced Sub-programs: IMPULS, MINMX3, SIMEQ, SMQINT

Referenced Commons: ITERAT, ITER2, REAL8

Entry Points: None

Referencing Sub-programs: MAIN

Discussion: Subroutine GUESS performs the ballistic trajectory option controlled by program input quantity MOPT, providing an initial guess for the trajectory starting parameters Λ_o, $\dot{\Lambda}_o$, and $v_{\infty o}$ (the initial primer, its derivative, and the launch excess speed).

The MINMX3 iterator drives the 3x3 two-point boundary value problem to a solution using IMPULS as the subroutine which maps the iterator independent variables $\dot{R}_o$ onto the dependent variables R_n by generating a ballistic trajectory segment of specified duration Δt and requiring $R_n - P_n = 0$; the spacecraft initial position and velocity are R_o and $\dot{R}_o$, the launch planet's position and velocity at time of launch t_o, which is held constant, are P_o and $\dot{P}_o$, and the target's position and velocity at time $t_o + \Delta t$ are P_n and $\dot{P}_n$.

As is the case for all HILTOP trajectories, $R_o = P_o$. The required starting conditions Λ_o, $\dot{\Lambda}_o$, and $v_{\infty o}$ (and $v_{\infty n}$ when MOPT = 2) are then obtained by the following computations: the solar distances of the launch and target bodies at the appropriate times (launch and intercept),

$$p_o = \sqrt{P_o \cdot P_o}$$

PRECEDING PAGE BLANK NOT FILMED

$$p_n = \sqrt{P_n \cdot P_n} \quad .$$

The transfer angle θ, assuming prograde motion, is given by:

If $(P_o \times P_n)_z \geq 0$,

$$\theta = \cos^{-1}\left(\frac{P_o \cdot P_n}{p_o p_n} \right).$$

If $(P_o \times P_n)_z < 0$,

$$\theta = 2\pi - \cos^{-1}\left(\frac{P_o \cdot P_n}{p_o p_n} \right).$$

Then define

$$a_h = \left[\frac{\Delta t}{\theta + 2\pi m} \right]^{2/3},$$

where Δt is the transfer time, in tau, and m is the (integer) number of whole revolutions required by the transfer. Then compute

$$p_h = \frac{1}{2} (p_o + p_n),$$

and if $a_h < p_h$, set $a_h = p_h$.

Then the initial guess, to be fed to the iterator, for the departure heliocentric velocity of the spacecraft is

$$\dot{R}_o = \sqrt{\frac{2}{p_o} - \frac{1}{a_h}} \; \frac{(P_o \times P_n) \times P_o}{|(P_o \times P_n) \times P_o|} \; \text{sign}\left[(P_o \times P_n)_z\right].$$

Control is then passed to the MINMX3 iterator, which accomplishes the targeting of the ballistic trajectory segment.

The initial primer vector is then given by (a unit vector in the direction of $V_{\infty o}$):

$$\Lambda_o = \frac{\dot{R}_o - \dot{P}_o}{|\dot{R}_o - \dot{P}_o|}.$$

Let Q be an auxiliary vector associated with the arrival excess velocity as follows:

For flyby maneuvers: $Q = 0$.

For orbiter maneuvers: $Q = \dfrac{\dot{P}_n - \dot{R}_n}{|\dot{P}_n - \dot{R}_n|}$.

Then the initial primer derivative is given by

$$\dot{\Lambda}_o = \left(\frac{\partial R_n}{\partial \dot{R}_o} \right)^{-1} (f \Lambda_o + (F \cdot \Lambda_o) R_o + (G \cdot \Lambda_o) \dot{R}_o - Q),$$

where f is from the well-known f-and-g series, and F and G are transition vectors,all output by subroutine ANSTEP; $\partial R_n / \partial \dot{R}_o$ is the 3x3 matrix consisting of the partial derivatives of the iterator dependent-variables with respect to the independent variables.

Finally, the launch excess speed is computed as

$$v_{\infty o} = |\dot{R}_o - \dot{P}_o|,$$

and, for primary-target orbiter maneuvers, the arrival excess speed is given by

$$v_{\infty n} = |\dot{P}_n - \dot{R}_n|.$$

GUESS EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(35)	SU	ITER2	Array of (active) iterator independent-variables.
F	U	REAL8	f (from f-and-g series).
X(50)	U	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.

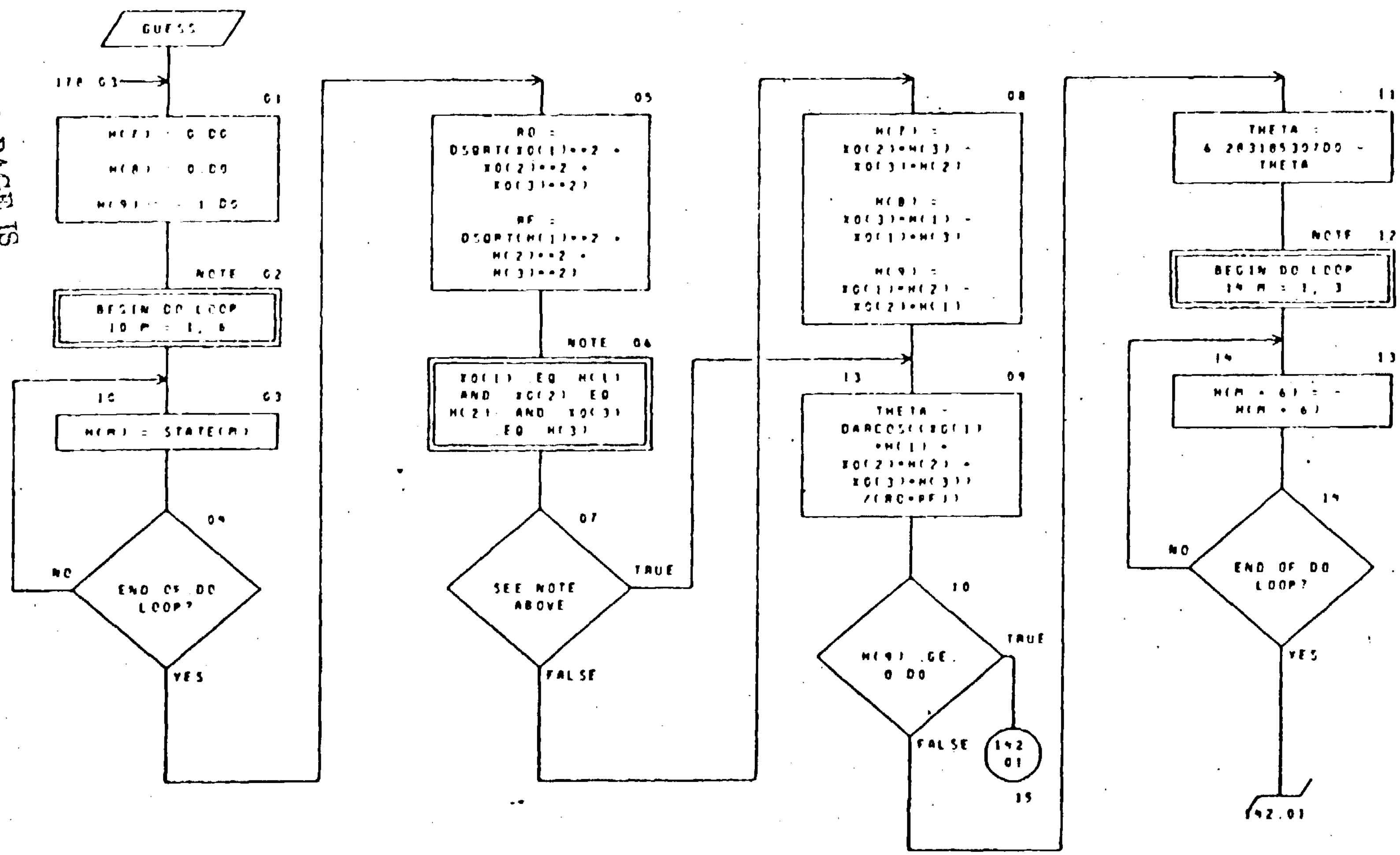
GUESS EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
BS(35)	S	ITER2	Array of maximum step-sizes for the iterator independent-variables.
BW(35)	S	ITER2	Array of iterator independent-variable weighting factors.
BX(5,70)	S	ITERAT	Array of iterator independent-variable values and related parameters, input to the program as Xi.
FX(6)	U	REAL8	Transition vector, computed in subroutine ANSTEP.
GX(6)	U	REAL8	Transition vector, computed in subroutine ANSTEP.
PM(1225)	SA	ITER2	3x3 matrix consisting of the negative of the partial derivatives of the iterator dependent-variables with respect to the independent variables, $-\partial R_n/\partial \dot{R}_o$, in tau.
P0(7)	SUA	REAL8	Array containing initial primer vector and its time derivative, Λ_o and $\dot{\Lambda}_o$.
SX(50)	SU	REAL8	Array of trajectory dependent-variables at the start of the computation step, corresponding to the launch time in this subroutine, and allocated the same as X(i).
X0(7)	U	REAL8	Array containing the launch planet's position and velocity at launch time, P_o and $\dot{P}_o$, in AU and EMOS, respectively.
BBB(35)	S	ITER2	Array of iterator independent-variable perturbation step-sizes.
CONX(70)	U	ITERAT	Array of conversion factors (between program internal and external units) for the iterator independent-variables.

GUESS EXTERNAL VARIABLES TABLE (cont)

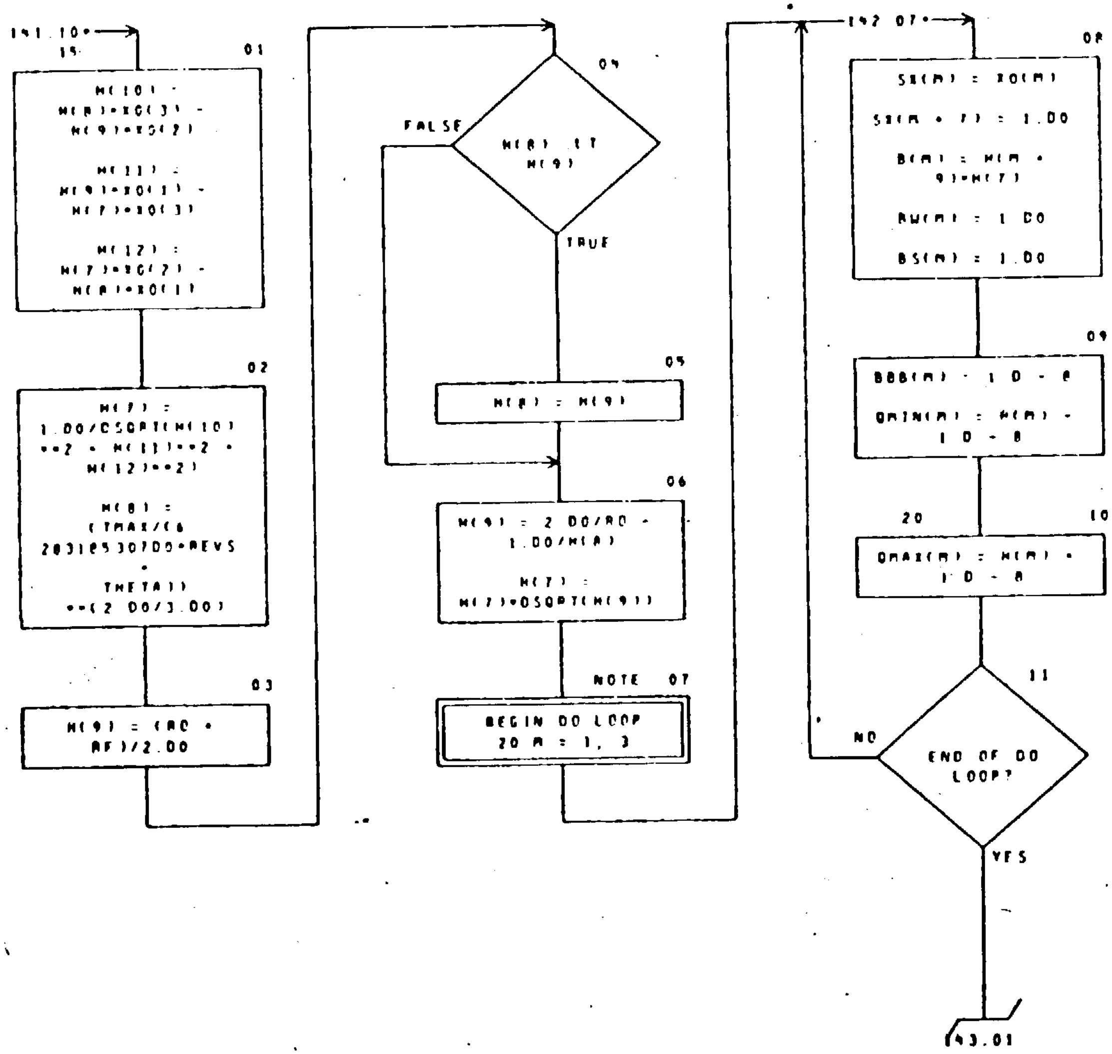
Variable	Use	Common	Description
MOPT	UX		Ballistic trajectory option indicator (input to the program).
QMAX(35)	S	ITER2	Array of upper allowable values for the iterator dependent-variables.
QMIN(35)	S	ITER2	Array of lower allowable values for the iterator dependent-variables.
REVS	U	REAL8	Number of whole revolutions about the sun, m.
TMAX	U	REAL8	Transfer time, Δt, in tau.
STATE(6)	UX		Array containing the position and velocity of the primary target at time of intercept, P_n and $\dot{P}_n$, in AU and EMOS, respectively.

CHART TITLE - SUBROUTINE GUESS(STATE,MOPT)



ORIGINAL PAGE IS
OF POOR QUALITY

CHART TITLE - SUBROUTINE GUESS(STATE,MOPT)



GUESS-7

CHART TITLE - SUBROUTINE GUESS(STATE,MOPT)

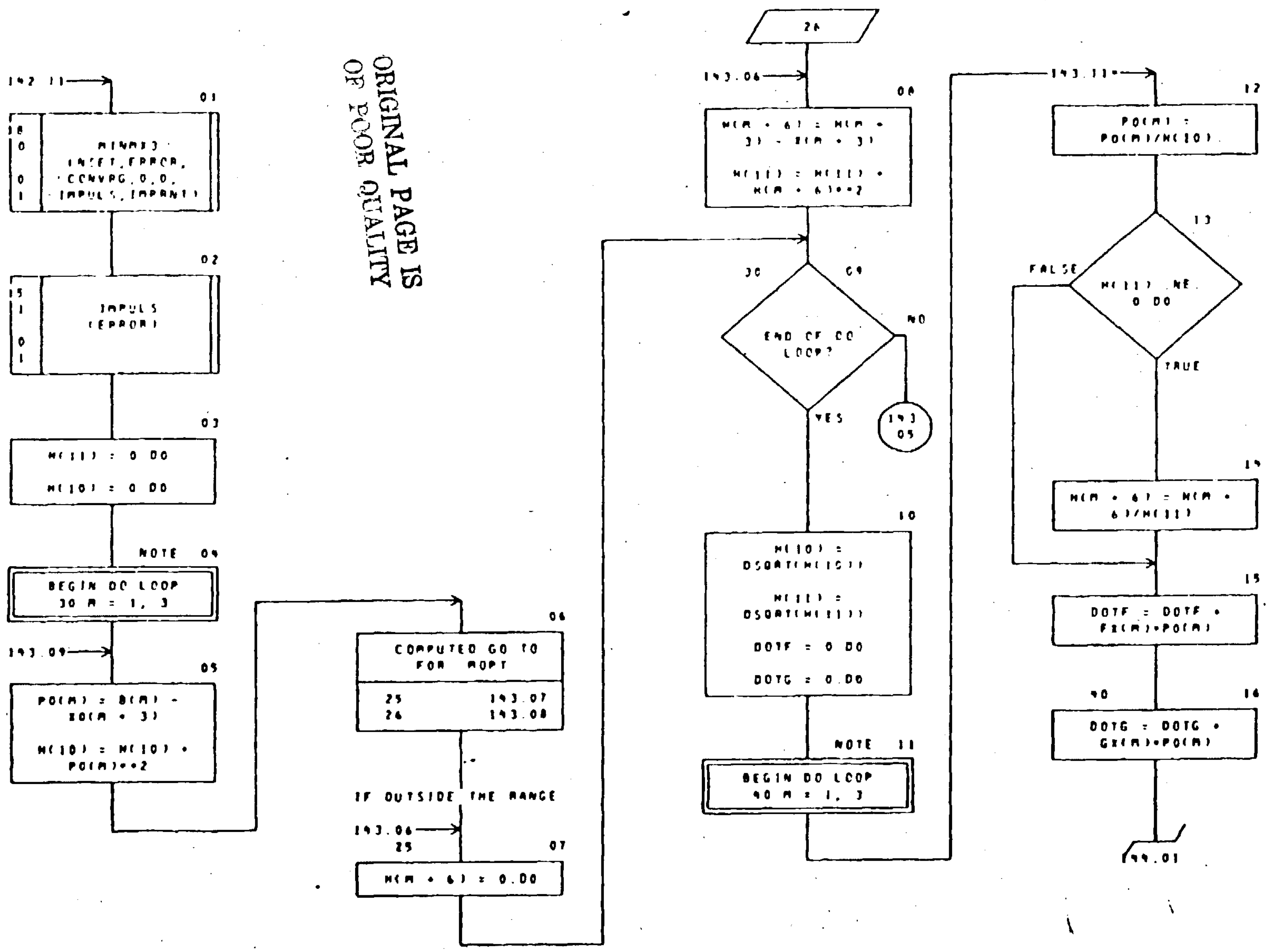
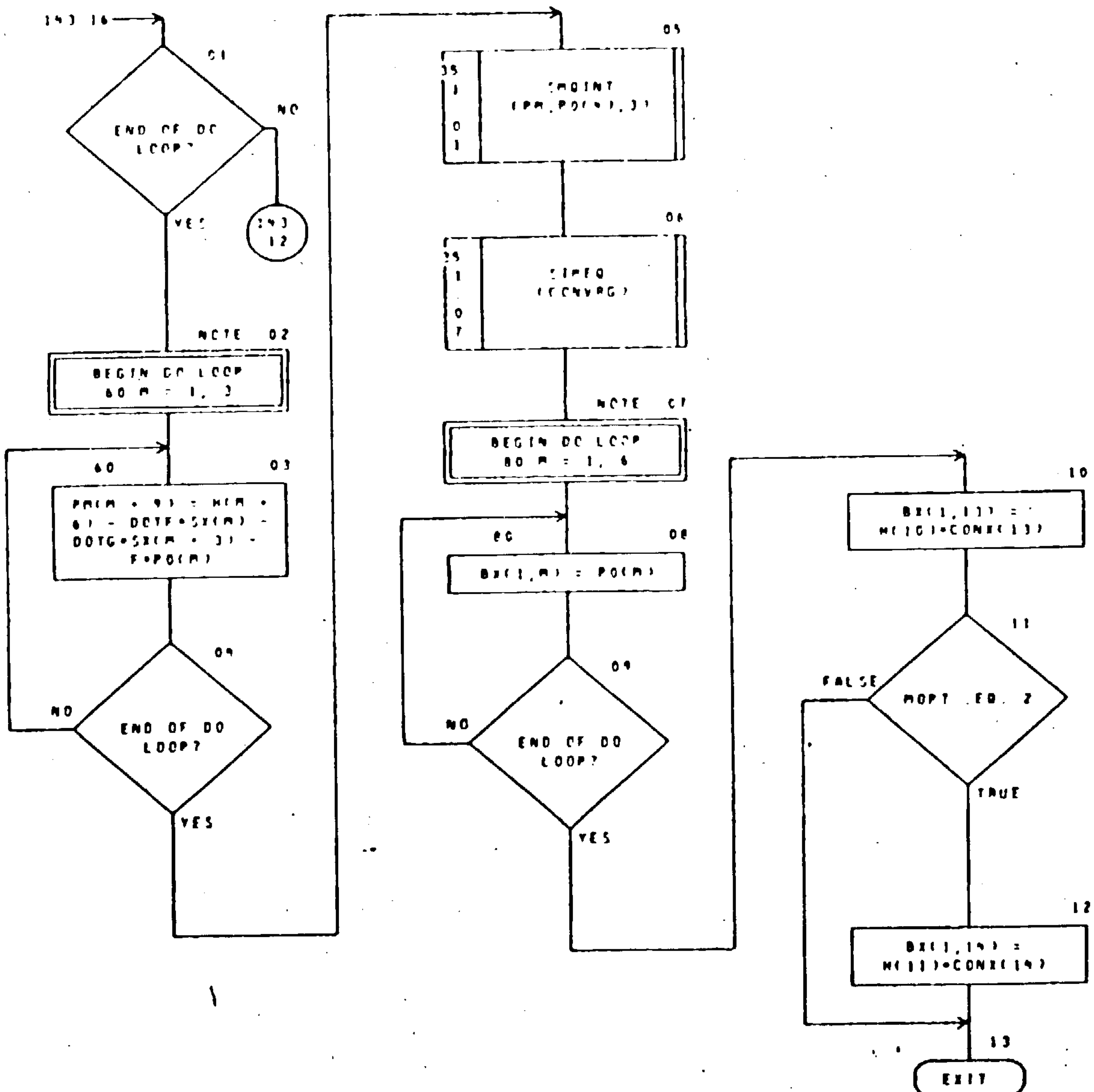


CHART TITLE - SUBROUTINE GUESS(STATE,MOPT) ,



GUESS-9

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL ERROR,CONVRG
DIMENSION NSET(5), STATE(6), H(12)
COMMON /REAL8/ RO1(50),X0(7),P0(7),RO2(24),TMAX,RO6(221),REVS,
 RO7(334),F,RO3,FX(6),GX(6),RO4(541),SX(50),X(50),RO5(700)
COMMON /ITERAT/ BX(5,70),BO1(210),CONX(70),BO2(350)
COMMON /ITER2/ B(35),PO1(35),BS(35),BW(35),QMIN(35),QMAX(35),
BBB(35),PO2(70),PM(1225)
EXTERNAL IMPULS,IMPRNT
DATA NSET / 3,3,200,0,0 /
```

<u>Name:</u> GUNTHR

<u>Calling Argument:</u> None

<u>Referenced Sub-programs:</u> None

<u>Referenced Commons:</u> GUNCOM

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> GET I, OMASS

<u>Discussion:</u> This routine computes the minimum incremental velocity required to achieve a given hyperbolic excess speed v_∞ along an asymptote not lying in the plane of a specified, initial circular orbit. The minimum Δv solution was identified by Gunther (see the Reference) as a particular root of a quartic equation in the sine of the asymptote out-of-plane angle i_∞. Also computed are $\partial \Delta v/\partial v_\infty$ and $\partial \Delta v/\partial i_\infty$. The computations, in order of solution, are as follows, where v_o is the speed in the circular orbit:

$$s = \sin(i_\infty); \quad \rho = v_\infty/v_o;$$

$$p = s^2(\rho^2 + 4);$$

$$q = s^2(1 - s^2)\rho^2;$$

$$x = \left[\sqrt{(q/2)^2 + (p/3)^3} + q/2\right]^{\frac{1}{3}} - \left[\sqrt{(q/2)^2 + (p/3)^3} - q/2\right]^{\frac{1}{3}};$$

$$y = \sqrt{\rho^2/4 - x};$$

$$w = \frac{1}{2}\left[\rho/2 + y + \sqrt{(\rho/2 + y)^2 + 4\left(x/2 + \sqrt{x^2/4 + s^2}\right)}\right],$$

$$\Delta v = v_o\sqrt{\rho^2 + 3 - 2\sqrt{(1 + \rho w - w^2)(2 + \rho w)}}.$$

The partial derivatives are given by the following expressions:

$$\frac{\partial \Delta v}{\partial v_\infty} = \frac{v_o^2}{\Delta v}\left\{\rho \frac{\partial \rho}{\partial v_\infty} - \frac{w(3+2\rho w - w^2)(\partial\rho/\partial v_\infty) + (3\rho + 2\rho^2 w - 3\rho w^2 - 4w)(\partial w/\partial v_\infty)}{2\sqrt{(1+\rho w - w^2)(2+\rho w)}}\right\},$$

$$\frac{\partial \Delta v}{\partial i_\infty} = -\frac{v_o^2(3\rho + 2\rho^2 w - 3\rho w^2 - 4w)}{2\Delta v\sqrt{(1+\rho w - w^2)(2+\rho w)}}\ \frac{\partial w}{\partial i_\infty},$$

where

$$\partial\rho/\partial v_\infty = 1/v_o,$$

$$\frac{\partial w}{\partial v_\infty} = \frac{1}{2}\left\{\frac{1}{2}\frac{\partial\rho}{\partial v_\infty} + \frac{\partial y}{\partial v_\infty} + \left[\left(1 + \frac{x}{2\sqrt{x^2/4+s^2}}\right)\frac{\partial x}{\partial v_\infty} + \left(\frac{\rho}{2}+y\right)\left(\frac{1}{2}\frac{\partial\rho}{\partial v_\infty} + \frac{\partial y}{\partial v_\infty}\right)\right]/(2w - \rho/2 - y)\right\},$$

$$\frac{\partial w}{\partial i_\infty} = \frac{1}{2}\left\{\frac{\partial y}{\partial i_\infty} + \left[\frac{\partial x}{\partial i_\infty} + \frac{x(\partial x/\partial i_\infty) + 4s(\partial s/\partial i_\infty)}{2\sqrt{x^2/4+s^2}} + \left(\frac{\rho}{2}+y\right)\frac{\partial y}{\partial i_\infty}\right]/(2w-\rho/2-y)\right\},$$

$$\partial s/\partial i_\infty = \cos(i_\infty),$$

$$\partial y/\partial v_\infty = \left[(\rho/2)\,\partial\rho/\partial v_\infty - \partial x/\partial v_\infty\right]/2y,$$

$$\partial y/\partial i_\infty = -(\partial x/\partial i_\infty)/2y,$$

$$\frac{\partial x}{\partial u} = \frac{1}{6}\left[\frac{(q/2)(\partial q/\partial u) + (p/3)^2(\partial p/\partial u)}{\sqrt{(q/2)^2+(p/3)^3}} + \frac{\partial q}{\partial u}\right]\left[\sqrt{(q/2)^2+(p/3)^3} + q/2\right]^{-2/3}$$

$$-\frac{1}{6}\left[\frac{(q/2)(\partial q/\partial u) + (p/3)^2(\partial p/\partial u)}{\sqrt{(q/2)^2+(p/3)^3}} - \frac{\partial q}{\partial u}\right]\left[\sqrt{(q/2)^2+(p/3)^3} - q/2\right]^{-2/3},$$

with $u = v_\infty$ or i_∞,

$$\partial q/\partial v_\infty = 2\rho s^2(1 - s^2)(\partial \rho/\partial v_\infty),$$

$$\partial q/\partial i_\infty = 2\rho^2 s(1 - 2s^2)(\partial s/\partial i_\infty),$$

$$\partial p/\partial v_\infty = 2\rho s^2(\partial \rho/\partial v_\infty),$$

$$\partial p/\partial i_\infty = 2s(\rho^2 + 4)(\partial s/\partial i_\infty).$$

Reference:

P. Gunther, "Asymptotically Optimum Two-Impulse Transfer from Lunar Orbit," AIAA Journal, Vol. 4, No. 2, pp. 346-349, February 1966.

GUNTHR EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
DV	SU	GUNCOM	Minimum incremental speed, Δv.
V0	U	GUNCOM	Speed in circular orbit, v_o.
I00	SU	GUNCOM	V_∞ asymptote out-of-plane angle, i_∞, in radians.
V00	U	GUNCOM	Hyperbolic excess speed, v_∞.
DVI00	S	GUNCOM	$\partial \Delta v/\partial i_\infty$.
DVV00	S	GUNCOM	$\partial \Delta v/\partial v_\infty$.
V0DIV	U	GUNCOM	Inverse of v_o, $1/v_o$.
LDVI00	U	GUNCOM	Indicator to bypass computations of $\partial \Delta v/\partial i_\infty$.
LDVV00	U	GUNCOM	Indicator to bypass computations of $\partial \Delta v/\partial v_\infty$.

Units of speed and speed-related parameters: any units are admissible, and output units are consistent with input units. Derivatives with respect to i_∞ are in speed-units/radian.

CHART TITLE - SUBROUTINE GUNTHR

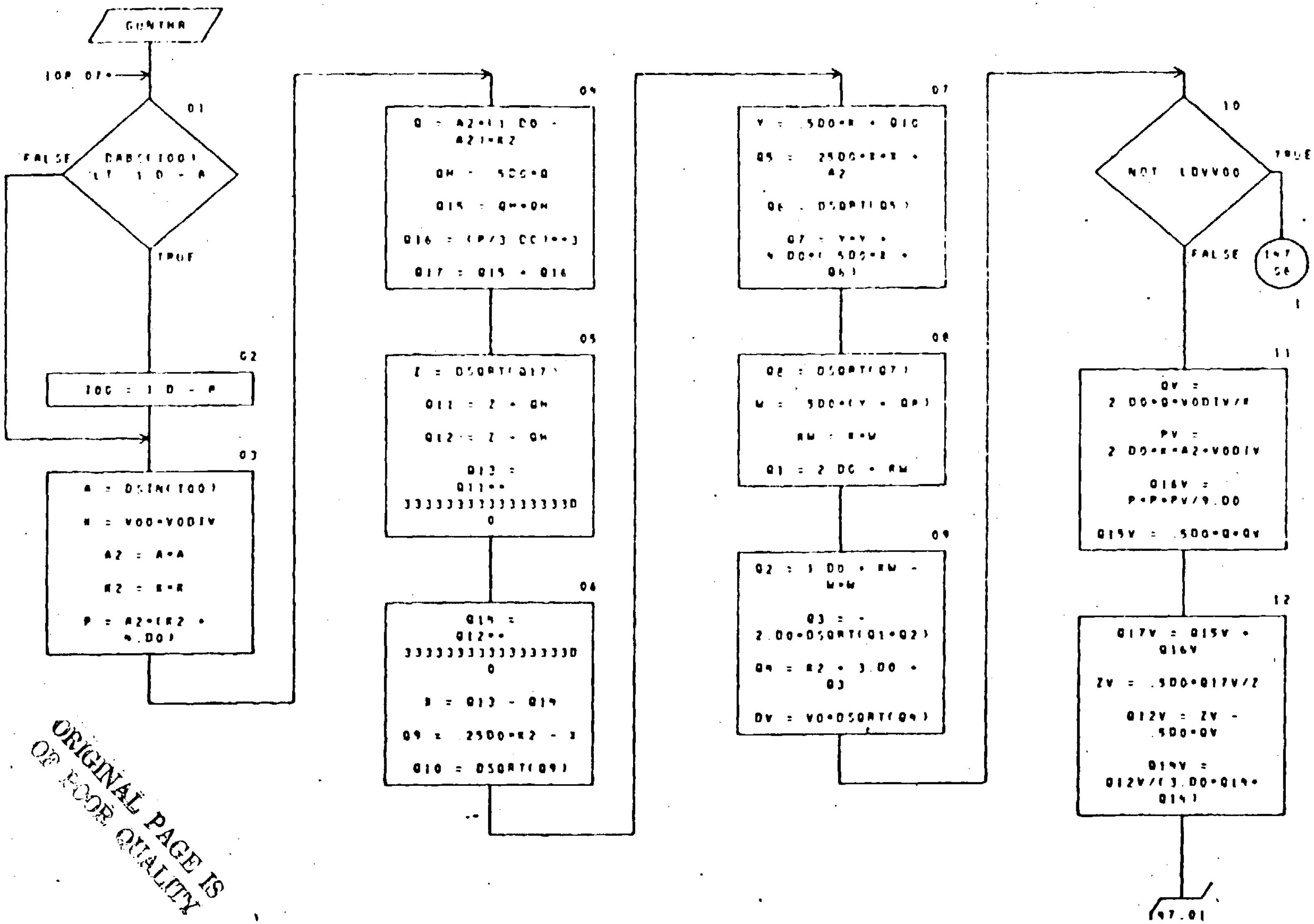
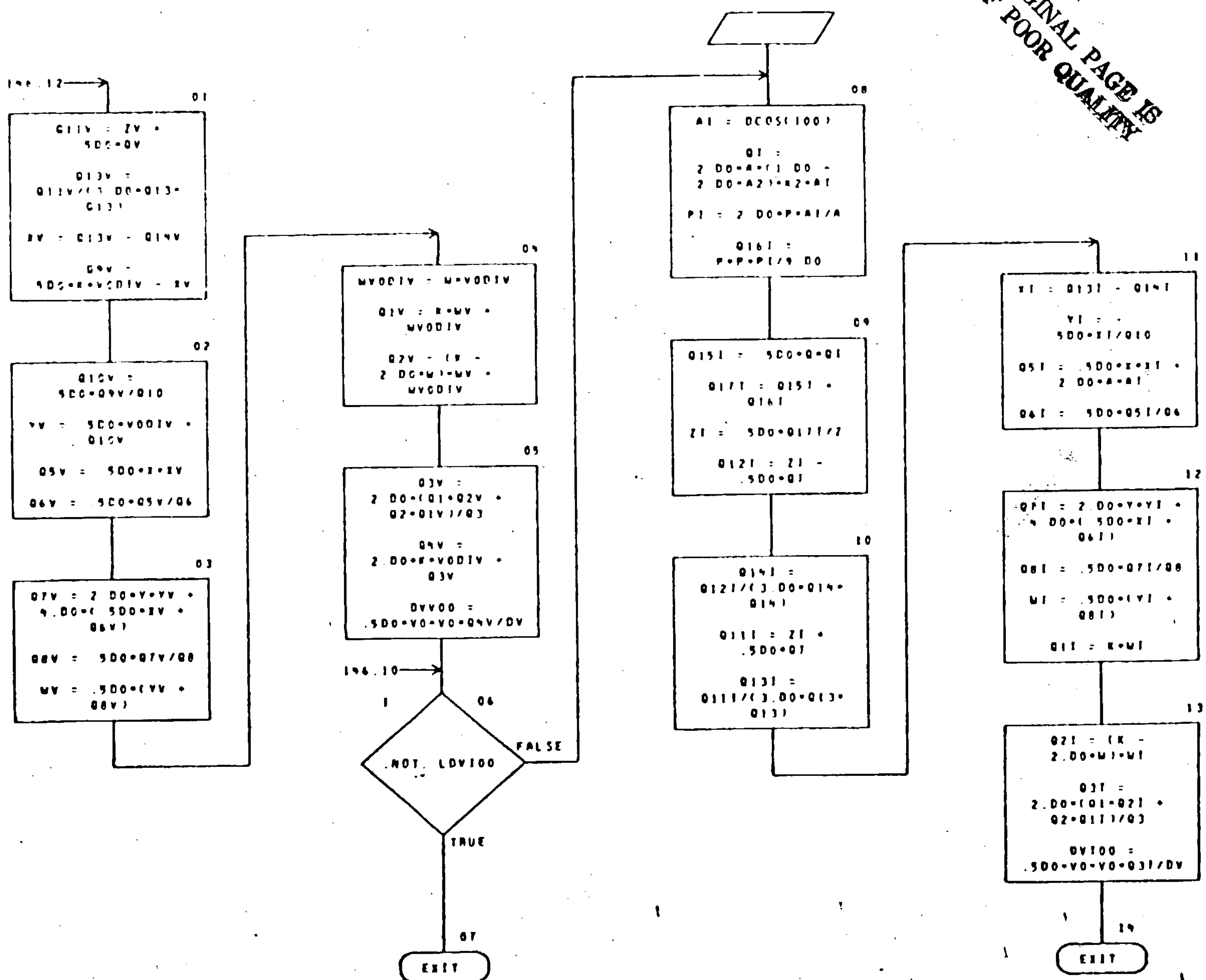


CHART TITLE - SUBROUTINE GUNTHR



GUNTHR-5

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
REAL*8 ICG,R,R2,RM
LOGICAL LDVV00,LDVI00
COMMON /GUNCOM/ V0,V0DIV,V00,I00,DV,DVV00,DVI00,LDVV00,LDVI00
```

Name: IMPRNT

Calling Argument: M1, M2, AAA

Referenced Sub-programs: None

Referenced Commons: REAL8

Entry Points: None

Referencing Sub-programs: GUESS, MINMX3

Discussion: Subroutine name IMPRNT is passed to the MINMX3 iterator in subroutine GUESS, and in subroutine MINMX3 the subroutine name becomes CCHECK instead of IMPRNT. Subroutine IMPRNT outputs the MINMX3 iteration history to the high speed printer.

Messages and printouts: The first printed line consists of three subroutine MINMX3 parameters; the number of trajectories in the current iteration sequence computed without also computing a partial derivative matrix, the number of trajectories computed having an associated partial derivative matrix, which was computed by generating a set of neighboring trajectories, and the iterator's inhibitor, as follows:

________ ________ ______________

This is followed by two lines consisting of seven quantities per line; the spacecraft position R in AU, velocity $\dot{R}$ in AU/tau, and mass ratio. The first line corresponds to the initial time and the second line to the final time.

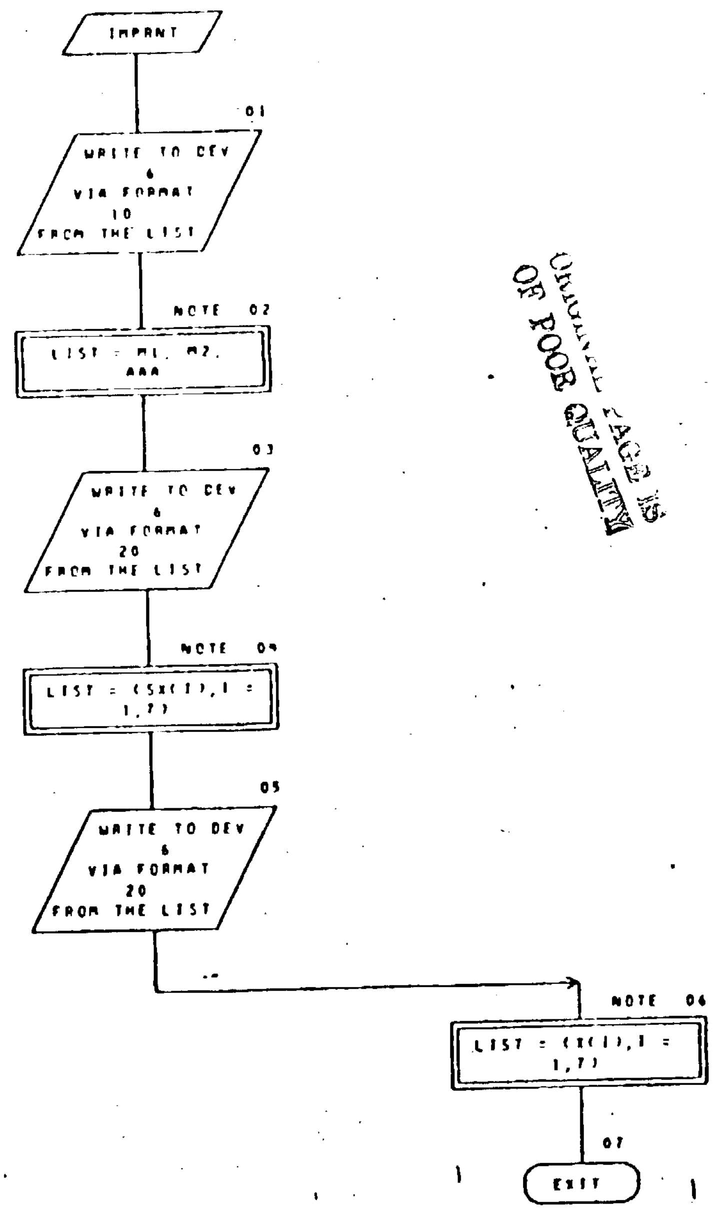
________ ________ ________ ________ ________ ________ ________

________ ________ ________ ________ ________ ________ ________

IMPRNT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
X(50)	U	REAL8	Array of trajectory dependent variables corresponding to the current time along the trajectory.
M1	UX		The number of trajectories in the current iteration sequence computed without also computing a partial derivative matrix.
M2	UX		The number of trajectories in the current iteration sequence computed having an associated partial derivative matrix.
SX(50)	U	REAL8	Array of trajectory dependent variables corresponding to the beginning of the current computation step.
AAA	UX		The iterator's inhibitor, λ.

CHART TITLE - SUBROUTINE IMPRNT(M1,M2,AAA) ,



IMPRNT-3

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H, O-Z)
      COMMON /REALP/ R01(1200),SX(50),X(50),R02(700)
10    FORMAT (1H0, 2I5, 1PD16.8)
20    FORMAT (1H , 1P7D18.10)
```

Name: IMPULS

Calling Argument: ERROR

Referenced Sub-programs: ANSTEP, SCOMP

Referenced Commons: ITER2, REAL8

Entry Points: None

Referencing Sub-programs: GUESS, MINMX3

Discussion: Subroutine name IMPULS is passed to the MINMX3 iterator in subroutine GUESS, and in subroutine MINMX3 the subroutine name becomes PD5 instead of IMPULS. The name IMPULS is short for "impulse" and refers to the type of trajectory which the subroutine generates: impulsive, or all-ballistic. The ballistic trajectory runs from the launch planet to the primary target (once convergence is achieved).

The two-point boundary value problem solved by the iterator is 3x3, where the independent variables consist of the spacecraft's heliocentric velocity at the launch planet (the launch time is specified), and the dependent variables consist of the spacecraft's position vector after a specified time interval has elapsed, this position being required to eventually match the target's position.

At entry into the routine, the error indicator is set "off", the spacecraft heliocentric velocity at the launch planet is set equal to the iterator independent variables, and the following initializations take place in preparation for generating the ballistic trajectory via the f and g series (g is not the symbol for reference thrust acceleration in this discussion):

$$d_o = R_o \cdot \dot{R}_o ,$$

$$r_o = \sqrt{x_o^2 + y_o^2 + z_o^2} ,$$

$$\alpha = v_o^2 - 2\mu/r_o ,$$

where

$$v_o^2 = \dot{x}_o^2 + \dot{y}_o^2 + \dot{z}_o^2 ,$$

and α is input to subroutine SCOMP. The time elapsed along the trajectory is then

$$\Delta t = g + \frac{G_3}{\sqrt{\mu}} = \frac{1}{\sqrt{\mu}} (r_o G_1 + \frac{d_o}{\sqrt{\mu}} G_2) + \frac{G_3}{\sqrt{\mu}} ,$$

but since the gravitational parameter of the sun, μ, is unity in the program's internal system of units, Δt is computed

$$\Delta t = r_o G_1 + d_o G_2 + G_3,$$

and α is computed

$$\alpha = v_o^2 - 2/r_o .$$

A more thorough exposition of the f and g series is given in the discussion of subroutine ANSTEP.

The trajectory independent-variable β is stepped in increments $\Delta\beta$ of one-half until $\Delta t > t_{max}$, where t_{max} is the trajectory stopping-condition time. Once this occurs, an iteration having β as the independent variable is executed until $\Delta t = t_{max}$ to within a specified tolerance. The iteration consists of Newton's method, having the dependent-variable

$$F = \Delta t - t_{max} \rightarrow 0,$$

and derivative

$$\frac{\partial F}{\partial \beta} = \frac{\partial F}{\partial \Delta t} \frac{\partial \Delta t}{\partial \beta} = (1) \left(\frac{1}{\dot{\beta}} \right) = r,$$

where the solar distance r is given by

$$r = r_o G_o + d_o G_1 + G_2.$$

Once the iteration for the final time converges, the iterator dependent-variables are set equal to the resulting spacecraft position.

Messages and printouts: When the iteration for the trajectory-endpoint time fails to converge within one hundred steps, the error indicator is set "on", the message is printed on units 6 and 12,

MAX ITERATIONS IN *IMPULS*

and the subroutine is exited.

IMPULS EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(35)	U	ITER2	Array of iterator independent-variables.
Q(35)	S	ITER2	Array of iterator dependent-variables.
X(50)	SU	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
SX(50)	SU	REAL8	Array of trajectory dependent-variables at the start of the computation step, corresponding to the launch time in this subroutine.
TMAX	U	REAL8	The trajectory-endpoint time, t_{max}, in tau.
ERROR	SX		Error indicator.

CHART TITLE - SUBROUTINE IMPULS(ERROR)

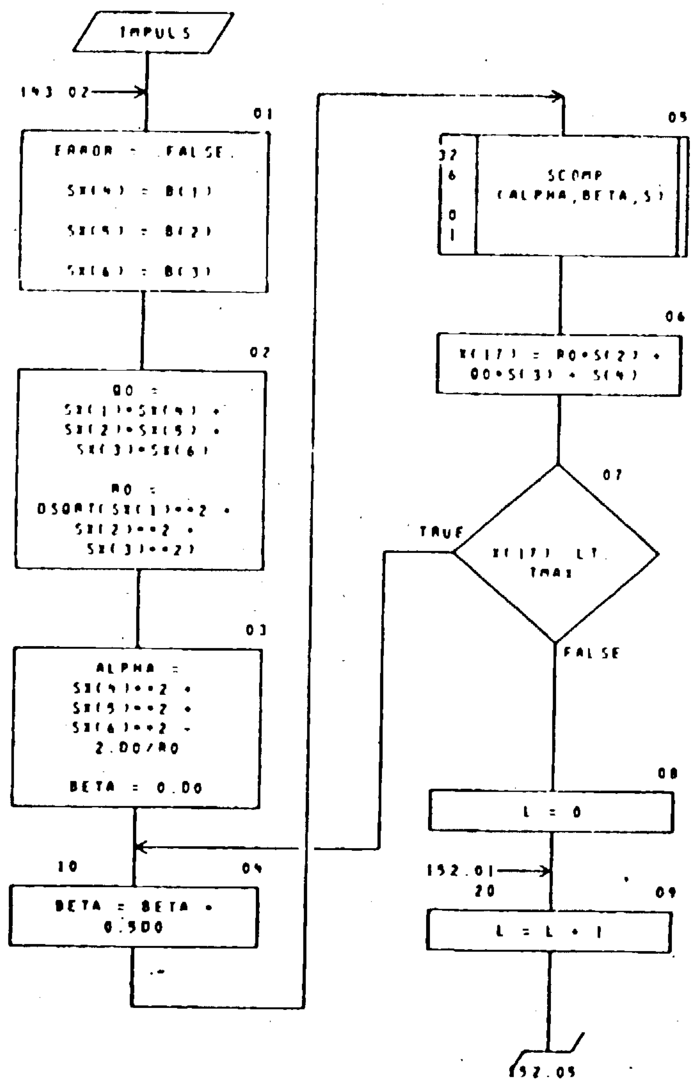


CHART TITLE - SUBROUTINE IMPULS(ERROR)

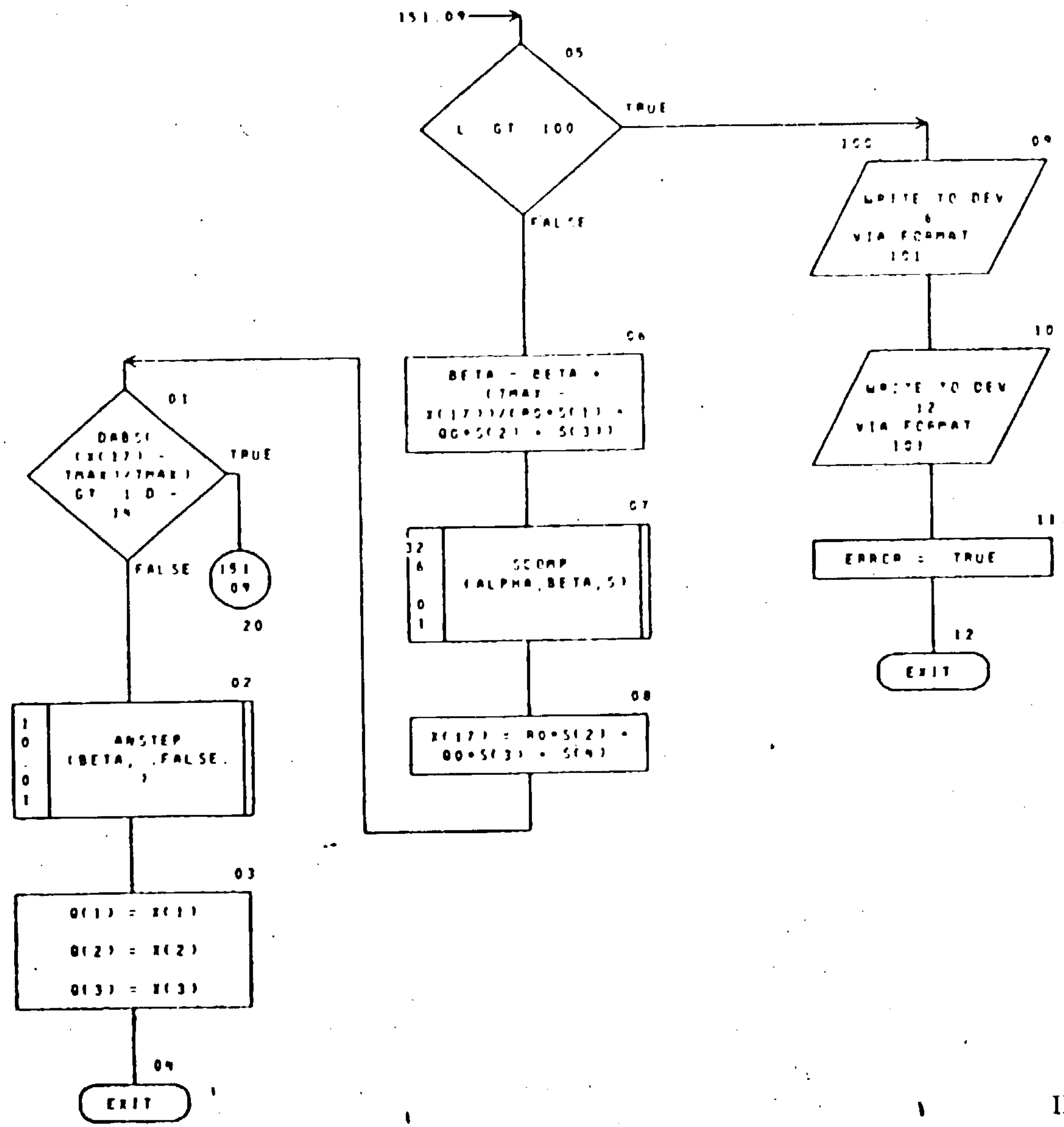


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL ERROR
      DIMENSION S(6)
      COMMON /REAL8/ PO1(88),TMAX,PO2(1111),SX(50),X(50),PO3(700)
      COMMON /ITER2/ B(35),Q(35),PO1(1470)
101   FORMAT(1H0,26HMAX ITERATIONS IN *IMPULS*)
```

ORIGINAL PAGE IS
OF POOR QUALITY

Name: INCOND

Calling Argument: ENP, SOMP, OMP, EYEP, GAMP, VP, SRP, R, RDH

Referenced Sub-programs: VCROSS, VSCAL

Referenced Commons: None

Entry Points: None

Referencing Sub-programs: GETRV

Discussion: INCOND calculates Cartesian position and velocity vectors given selected classical elements, a reference pole vector and the instantaneous radius and speed.

The reference coordinate system is defined as follows:

$$\hat{e}_\ell = \text{unit} \quad (\hat{k} \times \hat{e}_n)$$

where $\hat{e}_n$ is the input reference pole direction

$$\hat{e}_m = \hat{e}_n \times \hat{e}_\ell$$

The Cartesian vectors are given by

$$\begin{aligned} R = r\,[&(\cos\omega\cos\Omega - \sin\omega\sin\Omega\cos i) \quad \hat{e}_\ell \\ &+ (\cos\omega\sin\Omega + \sin\omega\cos\Omega\cos i) \quad \hat{e}_m \\ &+ (\sin\omega\sin i) \quad \hat{e}_n\,] \end{aligned}$$

$$\begin{aligned} V = v\,[&(-\sin(\omega-\gamma)\cos\Omega - \cos(\omega-\gamma)\sin\Omega\cos i)\,\hat{e}_\ell \\ &+ (-\sin(\omega-\gamma)\sin\Omega + \cos(\omega-\gamma)\cos\Omega\cos i)\,\hat{e}_m \\ &+ (\cos(\omega-\gamma)\sin i) \quad \hat{e}_n\,] \end{aligned}$$

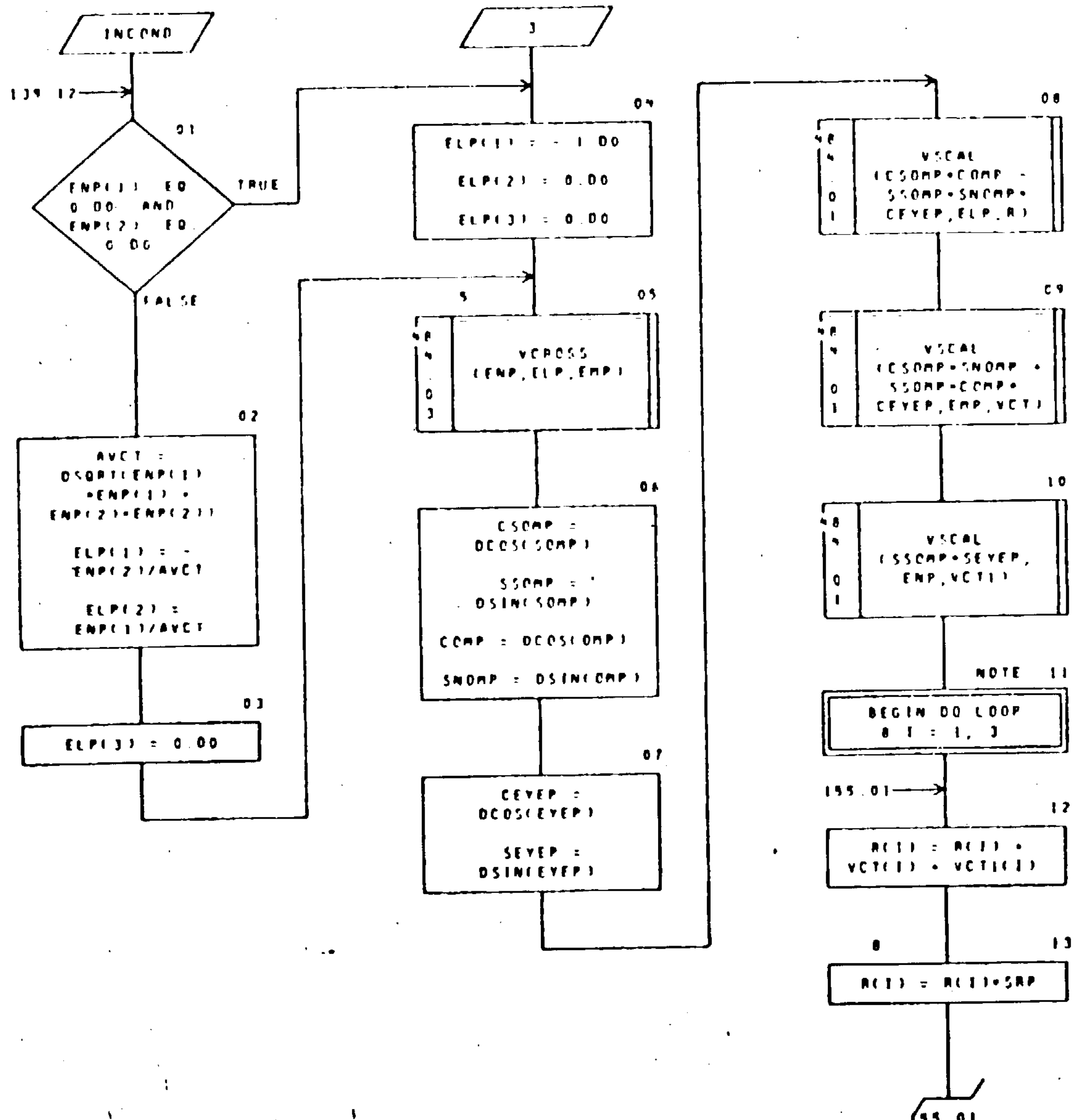
where ω is the argument of position, Ω is the longitude of the ascending node, i is the inclination, γ is the flight path angle, r is the magnitude of position and v is the speed.

INCOND EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
ENP(3)	UX		Unit pole vector, $\hat{e}_n$.
SOMP	UX		Argument of position, ω.
OMP	UX		Longitude of ascending node, Ω.
EYEP	UX		Inclination, i.
GAMP	UX		Flight path angle, γ .
VP	UX		Speed, v.
SRP	UX		Radius, r.
R(3)	SUX		Position vector, $\dot{R}$.
RDH(3)	SUX		Velocity vector, V.

Angles are input in radians; output R and V will be same units as input r and v.

CHART TITLE - SUBROUTINE INCOND(ENP,SOMP,OMP,EYEP,GAMP,VP,SRP,R,RDH)



ORIGINAL PAGE IS OF POOR QUALITY

INCOND-3

CHART TITLE - SUBROUTINE INCOND(ENP,SOMP,OMP,EVEP,GAMP,VP,SRP,R,ROH)

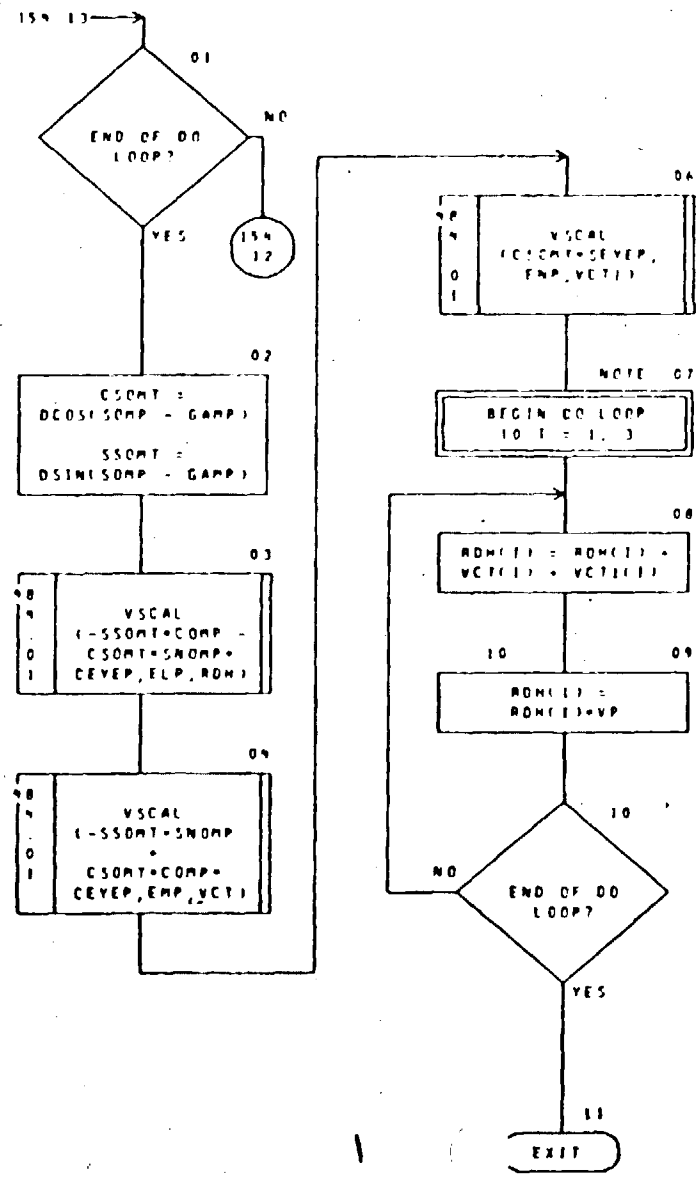


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ENP(3),ELP(3),EMP(3),        R(3),RDH(3),VCT(3),VCT1(3)
```

<u>Name:</u> INPUT

<u>Calling Argument:</u> GULP

<u>Referenced Sub-programs:</u> TIKTOK

<u>Referenced Commons:</u> INTGR4, ITERAT, REAL8

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> MAIN

<u>Discussion:</u> The read-in of program inputs via NAMELIST MINPUT is performed by this routine, and their values are printed at the beginning of each case. When all cases have been executed, the routine terminates the run by calling TIKTOK(3).

<u>Messages and printouts:</u> At the beginning of each case, all program inputs are printed following the heading

PROGRAM INPUTS

The iterator independent and dependent variable arrays are printed first, in the format:

```
X 1 = __________, __________, __________, __________, __________,
X 2 = __________, __________, __________, __________, __________,
X 3 = __________, __________, __________, __________, __________,
        .
        .
        .
X70 = __________, __________, __________, __________, __________,
Y 1 = __________, __________, __________,
Y 2 = __________, __________, __________,
Y 3 = __________, __________, __________,
        .
        .
        .
Y70 = __________, __________, __________,
```

PRECEDING PAGE BLANK NOT FILMED

In the X-arrays above, which constitute the independent variable arrays, the first quantity is the value, the second quantity is the trigger, the third quantity is the maximum allowed step size per iteration, the fourth quantity is the perturbation increment used for computing neighboring trajectories, and the fifth quantity is the weight. In the Y-arrays, which constitute the dependent variable arrays, the first quantity is the desired value, the second quantity is the trigger, and the third quantity is the convergence tolerance (full neighborhood width centered about the desired value). The remainder of the case inputs are then printed in NAMELIST format following

&OUTPUT

These parameters are printed in alphanumeric order, except that floating point quantities precede integer quantities.

INPUT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
AE	SU	REAL8	Desired final extra-ecliptic eccentricity.
AN	SU	REAL8	Trajectory-integration exponent in regularization formula.
AR	SU	REAL8	Desired final extra-ecliptic perihelion distance, in AU.
BI	SU	REAL8	Efficiency coefficient b in equation for efficiency.
BX(5, 70)	SUE	ITERAT	Iterator independent variable array.
BY(3, 70)	SUE	ITERAT	Iterator dependent variable array.
B1	SU	REAL8	Launch vehicle coefficient b_1, in kg.

INPUT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
B2	SU	REAL8	Launch vehicle coefficient b_2, in meters/second.
B3	SU	REAL8	Launch vehicle coefficient b_3, in kg.
DI	SU	REAL8	Efficiency coefficient d in equation for efficiency, in km/sec.
EI	SU	REAL8	Efficiency coefficient e in equation for efficiency.
JT	SU	INTGR4	Jettison indicator j_t for electric propulsion tankage prior to primary-target retro-maneuver.
T2(10)	SU	REAL8	Initial estimates of swingby-continuation trajectory-segment flight times, in days.
X0(7)	SU	REAL8	Spacecraft initial state vector, x_o, $\dot{x}_o$, M_o, where x_o is in AU, $\dot{x}_o$ is in EMOS, and $M_o = 1$.
AAI	SU	REAL8	Desired final extra-ecliptic inclination, in degrees.
CNI	SU	REAL8	Inclination to ecliptic of primary-target orbit, in degrees.
ECI	SU	REAL8	Eccentricity of primary-target orbit.
GAP	SU	REAL8	Propulsion-corner proximity tolerance-interval.
IRK	SU	INTGR4	Numerical integration option indicator (currently not used).
IRL	SU	INTGR4	Primer-origin-proximity step-size-control indicator.

INPUT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ITF	SU	INTGR4	Estimated time remaining to halt computer run with full printout, in case of proximity to maximum machine time, in seconds.
JPP	SU	INTGR4	Jettison indicator j_{ps} for electric propulsion system prior to primary-target retro-maneuver.
OMI	SU	REAL8	Ascending node angle (with respect to vernal equinox) of primary-target orbit, in degrees.
RAP	SU	REAL8	Apoapse distance of capture orbit about primary target, in planet radii.
SAI	SU	REAL8	Semi-major axis of primary-target orbit, in AU.
SOI	SU	REAL8	Argument of perihelion of primary-target orbit, in degrees.
TDV	SU	REAL8	Time of deep space burn, in days.
TGO	SU	REAL8	Ballistic trajectory-extension print option indicator.
TPI	SU	REAL8	Time from reference date (which is specified by MYEAR, etc.) to perihelion passage, for the primary target, in days.
ASOL(5)	SU	REAL8	Solar power law coefficients a_i in the expression for the solar power law.
CNIX(5)	SU	REAL8	Inclinations to ecliptic of intermediate-target orbits, in degrees.
CSTR	SU	REAL8	Electric propulsion system structural factor, k_s.
DPOW	SU	REAL8	Ratio of housekeeping power to reference power, p_h/p_{ref}.
ECIX(5)	SU	REAL8	Eccentricities of intermediate target orbits.

INPUT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
GULP	UX		Indicator for reading inputs or writing inputs.
HOUR	SU	REAL8	Hour-of-day of reference date.
IBAL	SU	INTGR4	Ballistic trajectory option indicator.
IOUT	SU	INTGR4	Extra-ecliptic mission indicator.
IROT	SU	INTGR4	Initial primer vector rotation indicator.
MDAY	SU	INTGR4	Day-of-month of reference date.
MODE	SU	INTGR4	Power variation option selector.
MOPT	SU	INTGR4	Ballistic trajectory option indicator.
MPOW	SU	INTGR4	Maximum or optimum power indicator during solar panel degradation option.
NSET(5)	SU	INTGR4	Iteration-sequence control array.
OMIX(5)	SU	REAL8	Ascending node angles of intermediate-target orbits, in degrees.
REVS	SU	REAL8	Indicator for number of revolutions of trajectory in ballistic trajectory option.
RPER	SU	REAL8	Periapse distance of capture orbit about primary target, in planet radii.
SAIX(5)	SU	REAL8	Semi-major axes of intermediate-target orbits, in AU.
SOIX(5)	SU	REAL8	Arguments of perihelion of inter-mediate-target orbits, in degrees.
TOFF(20)	SU	REAL8	Array of times, from the start of the trajectory, at which imposed coast phases are to begin, in days.

INPUT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
TPIX(5)	SU	REAL8	Times from reference date to perihelion passages, for the intermediate targets, in days.
CTANK	SU	REAL8	Electric propulsion system propellant tankage factor, k_t.
CTRET	SU	REAL8	Retro tankage factor, k_{rt}, for retro maneuver at the primary target.
INTPR	SU	INTGR4	Indicator which specifies print-length when the iteration in subroutine INTERP fails.
ISPIN	SU	INTGR4	Spinner indicator. Not used at present.
KPART	SU	INTGR4	Option indicator for automatically selecting improved independent parameter perturbations for generating the iterator's partial derivative matrix.
LOADX	SU	INTGR4	Indicator for invoking the intermediate-target initial-guess feature.
MONTH	SU	INTGR4	Month-of-year of reference date.
MOPTX(5)	SU	INTGR4	The target-numbers of the successive intermediate targets.
MOPT2	SU	INTGR4	Launch planet number and ephemeris-option indicator.
MOPT3	SU	INTGR4	Planet-number of primary target.
MOPT4(10)	SU	INTGR4	Planet-numbers of post-swingby targets.
MREAD	SU	INTGR4	Option indicator for card input of iterator independent variables.
MYEAR	SU	INTGR4	Year of reference date.

INPUT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
NDIST	SU	INTGR4	Identification number of celestial body to be used as the reference for the communication distance and angle measurement printed in the Extremum Point Summary Table.
NHUNG	SU	INTGR4	Maximum number of propulsion-corner-proximity occurrences allowed in a given iteration-sequence.
NPERF	SU	INTGR4	Identification number of the end condition that is to be used as the performance index when using the direct parameter optimization feature.
NTAPE	SU	INTGR4	Specifies the unit-number for the trajectory tape.
PSIGN	SU	REAL8	Coefficient defining the sense of the launch hyperbolic excess velocity relative to the initial primer vector.
STATE(6)	SU	REAL8	Array containing the Cartesian position and velocity components of the primary target, in AU and EMOS.
STEP1	SU	REAL8	Thrust-phase computation step-size, Δu.
STEP2	SU	REAL8	Coast-phase computation step-size, $\Delta \beta$.
THRET	SU	REAL8	Retro-stage thrust in retro maneuver at the primary target, in pounds.
XANG1	SU	REAL8	Latitude of the launch site, in degrees.
XANG2	SU	REAL8	Maximum parking orbit inclination permitted by range safety considerations, in degrees.

INPUT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ALPHAA	SU	REAL8	Specific mass of solar arrays, α_a, in kg/kw.
ALPHAT	SU	REAL8	Specific mass of thruster subsystem, α_t, in kg/kw.
ALTITU	SU	REAL8	Not used at present.
DMRETR	SU	REAL8	Retro engine mass, in kilograms.
EMUODD	SU	REAL8	Gravitational constant of the primary target, in m^3/sec^2.
EMUODX(5)	SU	REAL8	Gravitational constants of intermediate targets, in m^3/sec^2 (not used at present).
GAMMAX	SU	REAL8	Maximum permissible value of the power function γ.
ITPRNT	SU	INTGR4	Indicator for special printout from the iterator.
JPRINT	SU	INTGR4	Unit 11 printout-length indicator.
LAUNCH	SU	INTGR4	Launch mode selector.
MAXHAM	SU	INTGR4	Maximum number of times program will check Hamiltonian constancy.
MBOOST	SU	INTGR4	Launch vehicle selector.
MPRINT	SU	INTGR4	Indicator for printing the summary-trajectory (final case trajectory) as a function of time.
MPUNCH	SU	INTGR4	Punched-card and trajectory-tape generation control indicator.
MSWING(10)	SU	INTGR4	Swingby maneuver type selectors.
MTMASS	SU	INTGR4	Mission-type selector pertaining to the primary target.

INPUT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
MUPDAT	SU	INTGR4	Indicator for whether iterator independent variables at the finish of the iteration sequence of a given case are to be used as initial guesses for the next case.
NORMAL	SU	INTGR4	Automatic adjoint-variable normalization indicator.
NPRINT	SU	INTGR4	Printout amount selection indicator.
NSWING	SU	INTGR4	Ballistic swingby continuation analysis option flag.
NSWPAR	SU	INTGR4	Iterator independent-variable perturbation increment control indicator.
POWFIX	SU	REAL8	Launch-vehicle-independent (i.e., no launch vehicle) trajectory option indicator, in which the value of POWFIX is the spacecraft's reference power in kilowatts.
RADODD	SU	REAL8	Radius of primary target, in meters.
RADODX(5)	SU	REAL8	Radii of intermediate targets, in meters (not used at present).
SPIRET	SU	REAL8	Retro-stage specific impulse pertaining to the retro-maneuver at the primary target, in seconds.
TCOAST(20)	SU	REAL8	Array of times representing the duration of the coast-phase start times in TOFF, in days.
TPOWER	SU	REAL8	Solar-cell degradation characteristic-time, τ_d, in days.
TSCALE	SU	REAL8	Iterator dependent-variable tolerance-interval scaling factor.
XSWING(3,10)	SU	REAL8	Array of velocity vectors consisting of initial velocity guesses of a given post-swingby trajectory segment, in EMOS.

CHART TITLE - SUBROUTINE INPUT(GULP)

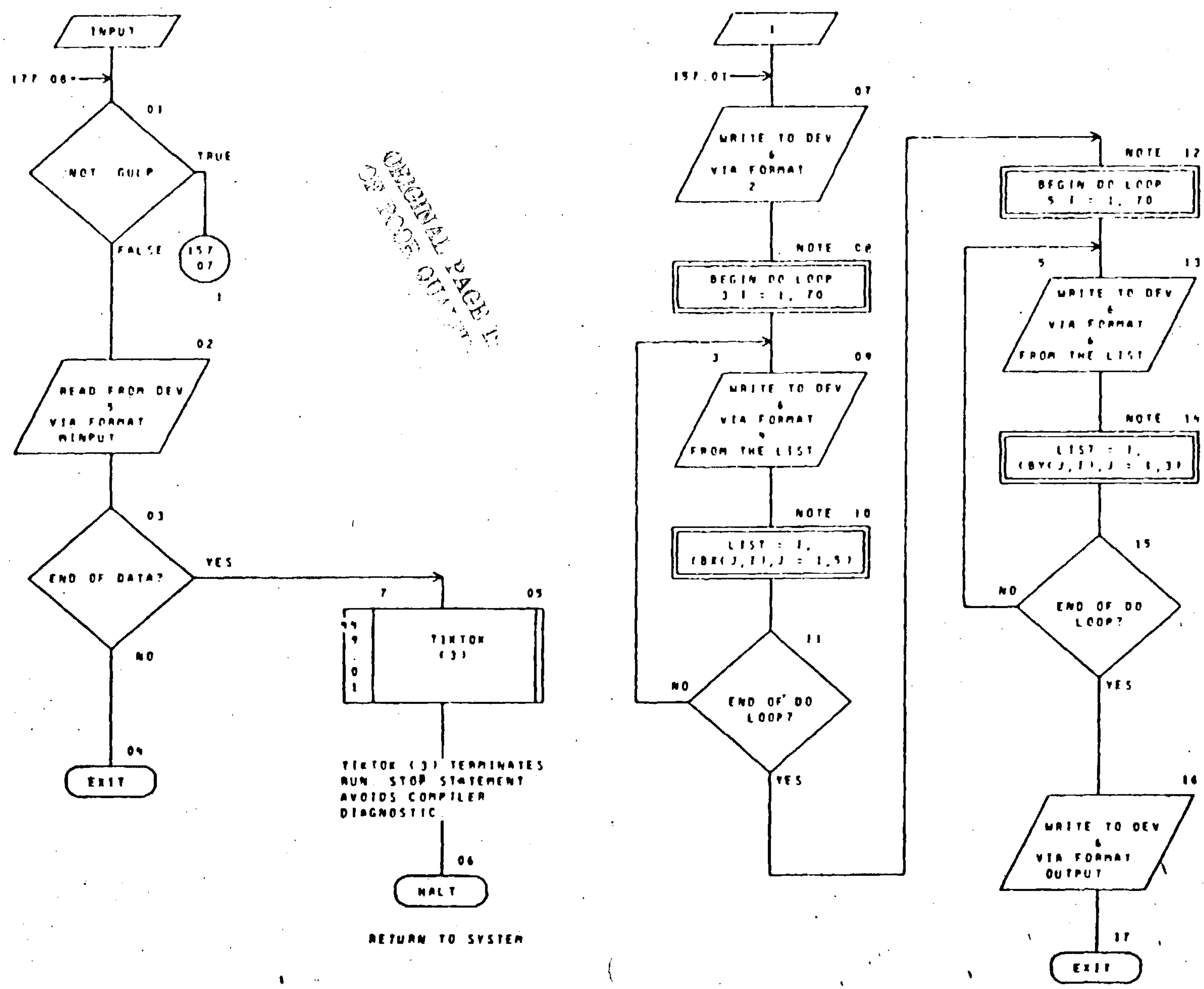


CHART TITLE - NON PROCEDURAL STATEMENTS

C-4

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL GULP
DIMENSION X1(5),X2(5),X3(5),X4(5),X5(5),X6(5),X7(5),X8(5),X9(5),
 X10(5),X11(5),X12(5),X13(5),X14(5),X15(5),X16(5),X17(5),X18(5),
 X19(5),X20(5),X21(5),X22(5),X23(5),X24(5),X25(5),X26(5),X27(5),
 X28(5),X29(5),X30(5),X31(5),X32(5),X33(5),X34(5),X35(5),X36(5),
 X37(5),X38(5),X39(5),X40(5),X41(5),X42(5),X43(5),X44(5),X45(5),
 X46(5),X47(5),X48(5),X49(5),X50(5),Y1(3),Y2(3),Y3(3),Y4(3),
 Y5(3),Y6(3),Y7(3),Y8(3),Y9(3),Y10(3),Y11(3),Y12(3),Y13(3),
 Y14(3),Y15(3),Y16(3),Y17(3),Y18(3),Y19(3),Y20(3),Y21(3),Y22(3),
 Y23(3),Y24(3),Y25(3),Y26(3),Y27(3),Y28(3),Y29(3),Y30(3),Y31(3),
 Y32(3),Y33(3),Y34(3),Y35(3),Y36(3),Y37(3),Y38(3),Y39(3),Y40(3),
 Y41(3),Y42(3),Y43(3),Y44(3),Y45(3),Y46(3),Y47(3),Y48(3),Y49(3),
 Y50(3)
DIMENSION X51(5),X52(5),X53(5),X54(5),X55(5),X56(5),X57(5),X58(5),
 X59(5),X60(5),X61(5),X62(5),X63(5),X64(5),X65(5),X66(5),X67(5),
 X68(5),X69(5),X70(5),Y51(3),Y52(3),Y53(3),Y54(3),Y55(3),Y56(3),
 Y57(3),Y58(3),Y59(3),Y60(3),Y61(3),Y62(3),Y63(3),Y64(3),Y65(3),
 Y66(3),Y67(3),Y68(3),Y69(3),Y70(3)
COMMON /REAL8/ R01(9)       ,CTANR,CSTR,RO2(8),DMRETR,
CTRET,RPER,RAP,THRET,SPIRET,RO3(3),ASCL(5),R11(8),HOUR,RO4,AAI,
STATE(6),X0(7),
RO5(7),BI,DI,EI,B1,B2,B3,POWFIX,AR,RO6(4),AE,SAI,ECI,CNI,OMI,
SOI,TPI,EMUODD,RADODD,R15,STEP1,STEP2,RO7,T2(10),RO8,TDV,R21,
GAP,RO9(73),TGD,R17(21),XANG1,XANG2,R18(25),
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
                 SATX(5),ECIX(5),CNIX(5),OMIX(5),SOIX(5),TPIX(5),
     FMUDDX(5),RADODX(5),ALPHAA,ALPHAT,
                       R1A(33),AN,R10(6),PSIGN,TSCALE,R14,REVS,
     R20(39),XSWING(3,10),R22(145),TOFF(20),TCONST(20),R19(134),
     ALTITU,R12(432),DPCW,GAMMAX,R23(54),TPOWER,R13(811)
     COMMON /INTGRS/ IPL,MODE,IRK,IIF,I08 ,JPRINT,MBODST,MCPT,NTAPE,
     MDAY,MONTH,MYEAR,MREAD,MUPDAT,MPRINT,NSET(5),IP01,JPP,JT,MCPT2,
     MCPT3,MCPT4(10),IC2,MIMASS,IC3(2) ,MPUNCH,IC4,NPRINT,IC9(4),
     NSMPAR,NORMAL,I10 ,KPART,I05,NHUNG,IC6(2),IBAL,IO7(14),LAUNCH,
     I11(2),ISPIN,IOUT,I12(51),INTPR,I13(2),MCPTX(5),I14(5),LOADX,
     MPCW,NSWING,NDIST,NPERF,I15,ITPRNT,I16,MAXHAM,I01,MSWING(10),
     I17(842)
     COMMON /ITERAT/ BX(5,70),BY(3,70),BC1(420)
     EQUIVALENCE (BX(1,1),X1(1)),(BX(1,2),X2(1)),(BX(1,3),X3(1))
     ,(BX(1,4),X4(1)),(BX(1,5),X5(1)),(BX(1,6),X6(1))
     ,(BX(1,7),X7(1)),(BX(1,8),X8(1)),(BX(1,9),X9(1))
     ,(BX(1,10),X10(1)),(BX(1,11),X11(1)),(BX(1,12),X12(1))
     ,(BX(1,13),X13(1)),(BX(1,14),X14(1)),(BX(1,15),X15(1))
     ,(BX(1,16),X16(1)),(BX(1,17),X17(1)),(BX(1,18),X18(1))
     ,(BX(1,19),X19(1)),(BX(1,20),X20(1)),(BX(1,21),X21(1))
     ,(BX(1,22),X22(1)),(BX(1,23),X23(1)),(BX(1,24),X24(1))
     ,(BX(1,25),X25(1)),(BX(1,26),X26(1)),(BX(1,27),X27(1))
     ,(BX(1,28),X28(1)),(BX(1,29),X29(1)),(BX(1,30),X30(1))
     ,(BX(1,31),X31(1)),(BX(1,32),X32(1)),(BX(1,33),X33(1))
     ,(BX(1,34),X34(1)),(BX(1,35),X35(1)),(BX(1,36),X36(1))
```

ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
,(BX(I,37),X37(I)),(BX(I,38),X38(I)),(BX(I,39),X39(I))
,(BX(I,40),X40(I)),(BX(I,41),X41(I)),(BX(I,42),X42(I))
,(BX(I,43),X43(I)),(BX(I,44),X44(I)),(BX(I,45),X45(I))
,(BX(I,46),X46(I)),(BX(I,47),X47(I)),(BX(I,48),X48(I))
,(BX(I,49),X49(I)),(BX(I,50),X50(I))
EQUIVALENCE (BY(I,1),Y1(I)),(BY(I,2),Y2(I)),(BY(I,3),Y3(I))
,(BY(I,4),Y4(I)),(BY(I,5),Y5(I)),(BY(I,6),Y6(I))
,(BY(I,7),Y7(I)),(BY(I,8),Y8(I)),(BY(I,9),Y9(I))
,(BY(I,10),Y10(I)),(BY(I,11),Y11(I)),(BY(I,12),Y12(I))
,(BY(I,13),Y13(I)),(BY(I,14),Y14(I)),(BY(I,15),Y15(I))
,(BY(I,16),Y16(I)),(BY(I,17),Y17(I)),(BY(I,18),Y18(I))
,(BY(I,19),Y19(I)),(BY(I,20),Y20(I)),(BY(I,21),Y21(I))
,(BY(I,22),Y22(I)),(BY(I,23),Y23(I)),(BY(I,24),Y24(I))
,(BY(I,25),Y25(I)),(BY(I,26),Y26(I)),(BY(I,27),Y27(I))
,(BY(I,28),Y28(I)),(BY(I,29),Y29(I)),(BY(I,30),Y30(I))
,(BY(I,31),Y31(I)),(BY(I,32),Y32(I)),(BY(I,33),Y33(I))
,(BY(I,34),Y34(I)),(BY(I,35),Y35(I)),(BY(I,36),Y36(I))
,(BY(I,37),Y37(I)),(BY(I,38),Y38(I)),(BY(I,39),Y39(I))
,(BY(I,40),Y40(I)),(BY(I,41),Y41(I)),(BY(I,42),Y42(I))
,(BY(I,43),Y43(I)),(BY(I,44),Y44(I)),(BY(I,45),Y45(I))
,(BY(I,46),Y46(I)),(BY(I,47),Y47(I)),(BY(I,48),Y48(I))
,(BY(I,49),Y49(I)),(BY(I,50),Y50(I))
EQUIVALENCE (BX(I,51),X51(I)),(BX(I,52),X52(I)),(BX(I,53),X53(I))
,(BX(I,54),X54(I)),(BX(I,55),X55(I)),(BX(I,56),X56(I))
,(BX(I,57),X57(I)),(BX(I,58),X58(I)),(BX(I,59),X59(I))
```

ORIGINAL PAGE IS
OF POOR QUALITY

INPUT-13

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
,(BX(I,60),X60(I)),(BX(I,61),X61(I)),(BX(I,62),X62(I))
,(BX(I,63),X63(I)),(BX(I,64),X64(I)),(BX(I,65),X65(I))
,(BX(I,66),X66(I)),(BX(I,67),X67(I)),(BX(I,68),X68(I))
,(BX(I,69),X69(I)),(BX(I,70),X70(I)),(BY(I,51),Y51(I))
,(BY(I,52),Y52(I)),(BY(I,53),Y53(I)),(BY(I,54),Y54(I))
,(BY(I,55),Y55(I)),(BY(I,56),Y56(I)),(BY(I,57),Y57(I))
,(BY(I,58),Y58(I)),(BY(I,59),Y59(I)),(BY(I,60),Y60(I))
,(BY(I,61),Y61(I)),(BY(I,62),Y62(I)),(BY(I,63),Y63(I))
,(BY(I,64),Y64(I)),(BY(I,65),Y65(I)),(BY(I,66),Y66(I))
,(BY(I,67),Y67(I)),(BY(I,68),Y68(I)),(BY(I,69),Y69(I))
,(BY(I,70),Y70(I))
NAMELIST/MINPUT/X1,X2,X3,X4,X5,X6,X7,X8,X9,X10,X11,X12,X13,X14,X15
,X16,X17,X18,X19,X20,X21,X22,X23,X24,X25,X26,X27,X28,X29,X30,X31,
X32,X33,X34,X35,X36,X37,X38,X39,X40,X41,X42,X43,X44,X45,X46,X47,
X48,X49,X50,X51,X52,X53,X54,X55,X56,X57,X58,X59,X60,X61,X62,X63,
X64,X65,X66,X67,X68,X69,X70,
Y1,Y2,Y3,Y4,Y5,Y6,Y7,Y8,Y9,Y10,Y11,Y12,Y13,Y14,Y15,Y16,Y17,Y18,Y19
,Y20,Y21,Y22,Y23,Y24,Y25,Y26,Y27,Y28,Y29,Y30,Y31,Y32,Y33,Y34,Y35,
Y36,Y37,Y38,Y39,Y40,Y41,Y42,Y43,Y44,Y45,Y46,Y47,Y48,Y49,Y50,Y51,
Y52,Y53,Y54,Y55,Y56,Y57,Y58,Y59,Y60,Y61,Y62,Y63,Y64,Y65,Y66,Y67,
Y68,Y69,Y70,
AAI,AE,ALTITU,AN,AR,ASOL,BI,B1,B2,B3,TSCALE,XSWING,ALPHAR
,ALPHAT,CNI,CNIX      ,CSTR,CTANK,CTRET,DI,DPCW, DMRETR,ECI,ECIX
,EI,EMUODD,EMUODX,GAP,GAMMAX,HOUR,OMI,OMIX,POWFIX,RADODD,RADODX,
RAP,RPER,SAI,SAIX,SOI,SOIX,TGO,XANG1,XANG2,TOFF,TCOAST,REVS,
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
           SPIRET,STATE,STEP1,STEP2,THRET,TPI,TPIX ,TPOWER,T2,PSIGN,TOV,
       X0,IBAL,INTPR,IOUT,LOADX,IRL,
       IRX,IROT,ISPIN,ITF,JPP,JPRINT,JT,KPART,LAUNCH,MBOOST,MDAY,MODE,
       MONTH,MOPT,MOPTX,MOPT2,MOPT3,MOPT4,MPRINT,MPUNCH,MREAD,MTMASS,
       MUPDAT,MPOW,NSWING,NDIST,NPERF,ITPRNT,MAXHAM,MSWING,
       MYEAR,NHUNG,NORMAL,NPRINT,NSET,NSWPAR,NTAPE
       NAMELIST/OUTPUT/AAI,AE,ALPHAA,ALPHAT,ALTITU,AN,AR,ASOL,BI,B1,B2,B3
              ,CNI,CNIX      ,CSTR,CTANX,CTRET,DI,DMPETP,DPOW ,ECI,ECIX
       ,FI,EMUODD,EMUODI,GAMMAX,
                           GAP,HOUR,OMI,OMIX,POWFIX,PSIGN, RADODD,RADODI,
       RAP,REVS,RPER        ,SAI,SAIX,SOI,SCIX,
       SPIRET,STATE,STEP1,STEP2,TCOAST,TOV,TGO,THRET,TOFF,TPI,TPIX,
       TPOWER,TSCALE,T2,XANG1,XANG2,ESWING,
       X0,IBAL,INTPR,IOUT,IRX,IRL,IROT,ISPIN,
       ITF,ITPRNT,         JPP,JPRINT,JT,KPART,LAUNCH,LOADX,MAXHAM,MBOOST,
       MDAY,MODE,
       MONTH,MOPT,MOPTX,MOPT2,MOPT3,MOPT4,MPOW,MPRINT,MPUNCH,MREAD,
       MSWING,MTMASS,MUPDAT,MYEAR,NDIST,NHUNG,NORMAL,
       NPERF,              NPRINT,NSET,NSWING,NSWPAR,NTAPE
2      FORMAT(1H ,50X19HPROGRAM INPUTS/1H )
4      FORMAT(1H ,1HX12,2H =5(1PD23.15,2H, ))
6      FORMAT(1H ,1HY12,2H =3(1PD23.15,2H, ))
```

<u>Name:</u> INTERP

<u>Calling Argument:</u> I

<u>Referenced Sub-programs:</u> CDERIV, SPRINT, STEP

<u>Referenced Commons:</u> EXTREM, INTGR4, LOGIC4, REAL8

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> CHECK

<u>Discussion:</u> Once subroutine CHECK determines that a root of a monitored function $f(\beta)$ lies within the current computation step $\Delta\beta$, subroutine INTERP performs the task of isolating the root of that function (i.e., of determining the value of the trajectory independent-variable $\beta = \beta_x$ such that $f(\beta_x) = 0$).

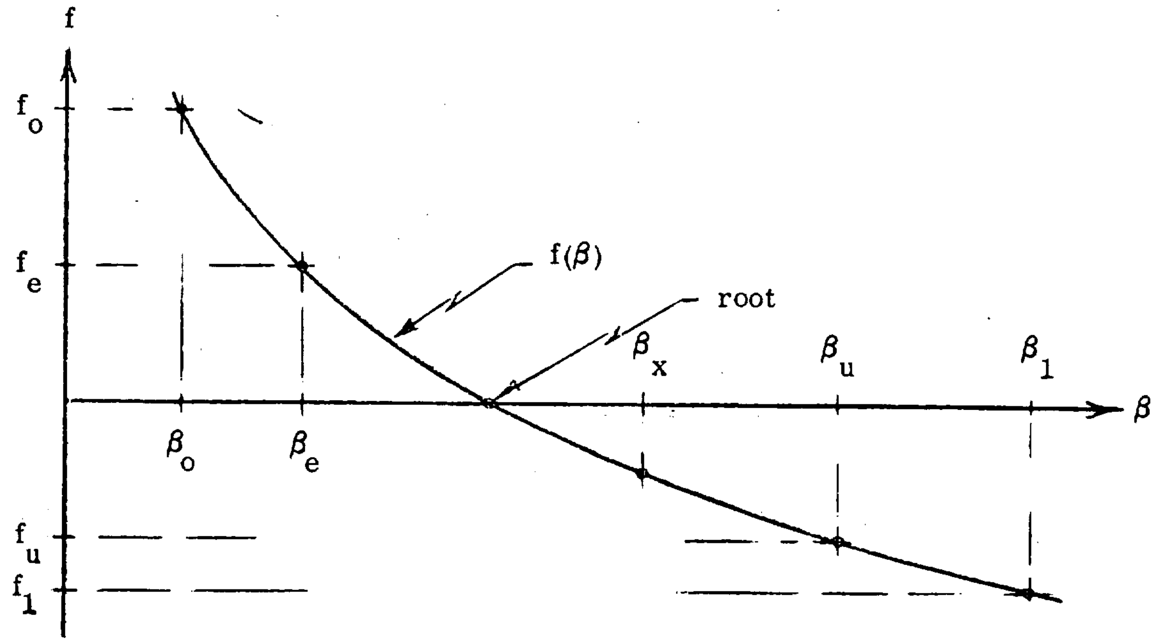
The root isolation is performed by approximating the given function $f(\beta)$ by a quadratic polynomial:

$$f(\beta) \cong c\,\beta^2 + b\,\beta + a\,,$$

and by solving analytically for a good estimate of the root, $f(\beta_x) = 0$, using the solution to the quadratic,

$$\beta_x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2c}\,,$$

in which the sign ambiguity is resolved by rejecting the root lying outside of the computation step $\Delta\beta$, i.e., outside of $\beta_0 \le \beta \le \beta_1$. The following sketch may be helpful in visualizing the several points along the function, which will be employed in the discussion below:



Three points (ordered pairs $(\beta, f(\beta))$) are chosen from the actual function $f(\beta)$ in order to generate the quadratic approximation of $f(\beta)$:

$$(\beta_e, f_e), \quad (\beta_j, f_j), \quad (\beta_u, f_u).$$

These points are each required to satisfy the quadratic formula, with the object of determining the unknown values of the coefficients a, b, and c:

$$f_e = c\beta_e^2 + b\beta_e + a\,, \tag{1}$$

$$f_j = c\beta_j^2 + b\beta_j + a\,, \tag{2}$$

$$f_u = c\beta_u^2 + b\beta_u + a\,. \tag{3}$$

The point β_j is not depicted in the sketch because it may vary with each iterative cycle toward determining the root of the actual function $f(\beta)$. The point β_j is chosen in such a manner as to narrow down the root as quickly as possible; β_j

becomes β_u and β_u becomes β_1, or β_j becomes β_e and β_e becomes β_o, with the f's being substituted in like manner.

Equations (1), (2), and (3) consist of three equations in three unknowns a, b, and c, which are solved as follows:

From (1),

$$a = f_e - \beta_e (b + c\beta_e) \quad . \tag{4}$$

Substituting (4) into (3) leads to

$$b = \frac{f_u - f_e}{\beta_u - \beta_e} - c(\beta_u + \beta_e) . \tag{5}$$

Substituting (4) and (5) into (2) leads to

$$c = -\frac{f_e(\beta_j - \beta_u) + f_u(\beta_e - \beta_j) + f_j(\beta_u - \beta_e)}{(\beta_e - \beta_j)(\beta_u - \beta_e)(\beta_j - \beta_u)} . \tag{6}$$

Therefore c is first computed from (6), then b is computed from (5), and finally a is computed using (4). The root β_x of the quadratic approximating-function is near to, but not exactly equal to, that of the actual function. Successive cycles, or iterations, through the quadratic-approximation formulas above, each time using an improved value of β_j, rapidly leads to the actual root to within a specified tolerance.

Messages and printouts: If the function root is not isolated to the specified tolerance within fifty iterations, the error message is written on unit 6:

ITERATION FOR FUNCTION ROOT DID NOT CONVERGE.
FUNCTION((i)) = (name) TIME = (t) COAST = (e)

where i is the index of the monitored function, B(2, i), "name" is the name of the function, t is the time elapsed since the beginning of the trajectory, in days, and e is a logical indicator T, for true, or F, for false, for coasting flight (the converse being thrusting flight). This is followed by a heading:

SPECIAL BLOCK PRINT AT FAIL POINT

followed by a block of trajectory information output via subroutine SPRINT.

If long-print is invoked via the program input indicator INTPR, the entire failed iteration is repeated and special printout is executed at each iteration following the block-heading,

KOUNT	TS	CSS	Z	AYE	BEE	CEE	DET
YL	YU	YJ	Y(1)	Y(2)	YJU	YUL	YLJ
FL	FU	FJ	B(1, I)	B(2, I)	TIME	YSEQ0	DELTA
(1) ______	______	______	______	______	______	______	______
______	______	______	______	______	______	______	______
______	______	______	______	______	______	______	______
(2) ______	______	______	______	______	______	______	______
______	______	______	______	______	______	______	______
______	______	______	______	______	______	______	______
⋮							
(50) ______	______	______	______	______	______	______	______
______	______	______	______	______	______	______	______
______	______	______	______	______	______	______	______

The names in the block-heading are defined as follows:

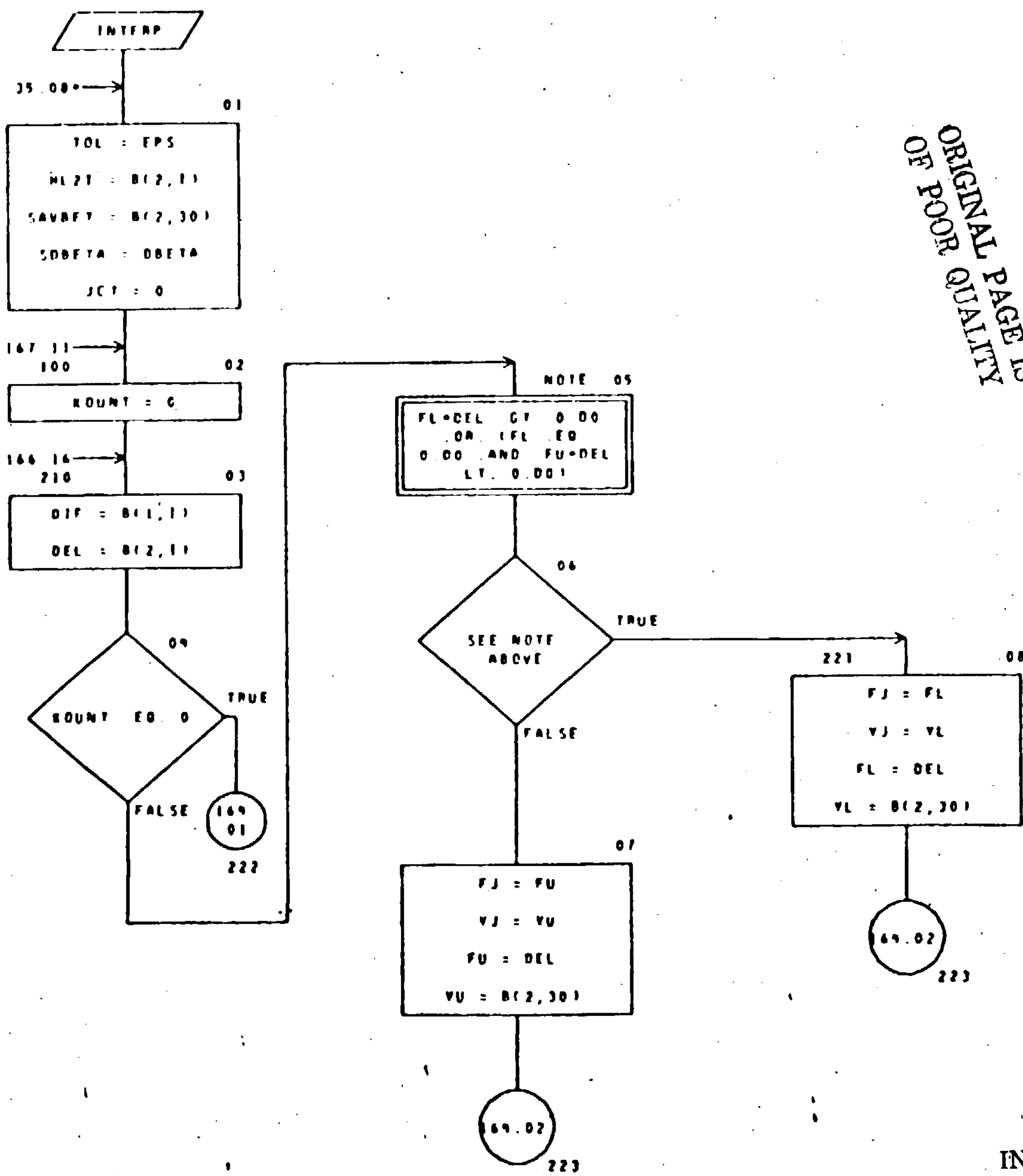
KOUNT	Iteration counter.
TS	Time at β_o, $t(\beta_o)$, in tau.
CSS	Current value of computation step, $\Delta\beta$.
Z	Identical to CSS.
AYE	Quadratic coefficient a.
BEE	Quadratic coefficient b.
CEE	Quadratic coefficient c.
DET	$-(\beta_j - \beta_u)(\beta_u - \beta_e)(\beta_e - \beta_j)$.

YL	β_e.
YU	β_u.
YJ	β_j.
Y(1)	β_o.
Y(2)	β_1.
YJU	$\beta_j - \beta_u$.
YUL	$\beta_u - \beta_e$.
YLJ	$\beta_e - \beta_j$.
FL	f_e.
FU	f_u.
FJ	f_j.
B(1, I)	Value of monitored function f at β_o.
B(2, I)	Value of monitored function f at β_1.
TIME	Time elapsed since start of trajectory, in days.
YSEQ0	β_x.
DELTA	$\beta_1 - \beta_x$.

INTERP EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(2, 30)	SU	EXTREM	Array of monitored functions.
I	UX		Function selector.
X(50)	U	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
SX(50)	U	REAL8	Array of trajectory dependent-variables corresponding to the beginning of the current computation step, allocated the same as X(i).
QJEX	U	LOGIC4	Detailed printout indicator, associated with the final summary-trajectory of a given case.
COAST	U	LOGIC4	Indicator for coasting flight or thrusting flight.
CONTM	U	REAL8	Time conversion factor, tau to days.
DBETA	SU	REAL8	Computation step size, $\Delta\beta$ (increment of trajectory independent-variable).
INTPR	U	INTGR4	Long-print indicator input to the program ("Interp print").
SBETA	U	REAL8	Value of trajectory independent-variable corresponding to the beginning of the current computation step, β_o.
MPUNCH	SU	INTGR4	Indicator for punched cards and trajectory tape generation, input to the program.
SKOUNT	SU	REAL8	The number of steps, or iterations, required to isolate the remarkable point to the specified tolerance.

CHART TITLE - SUBROUTINE INTERP(I)



INTERP-7.

CHART TITLE - SUBROUTINE INTERP(I)

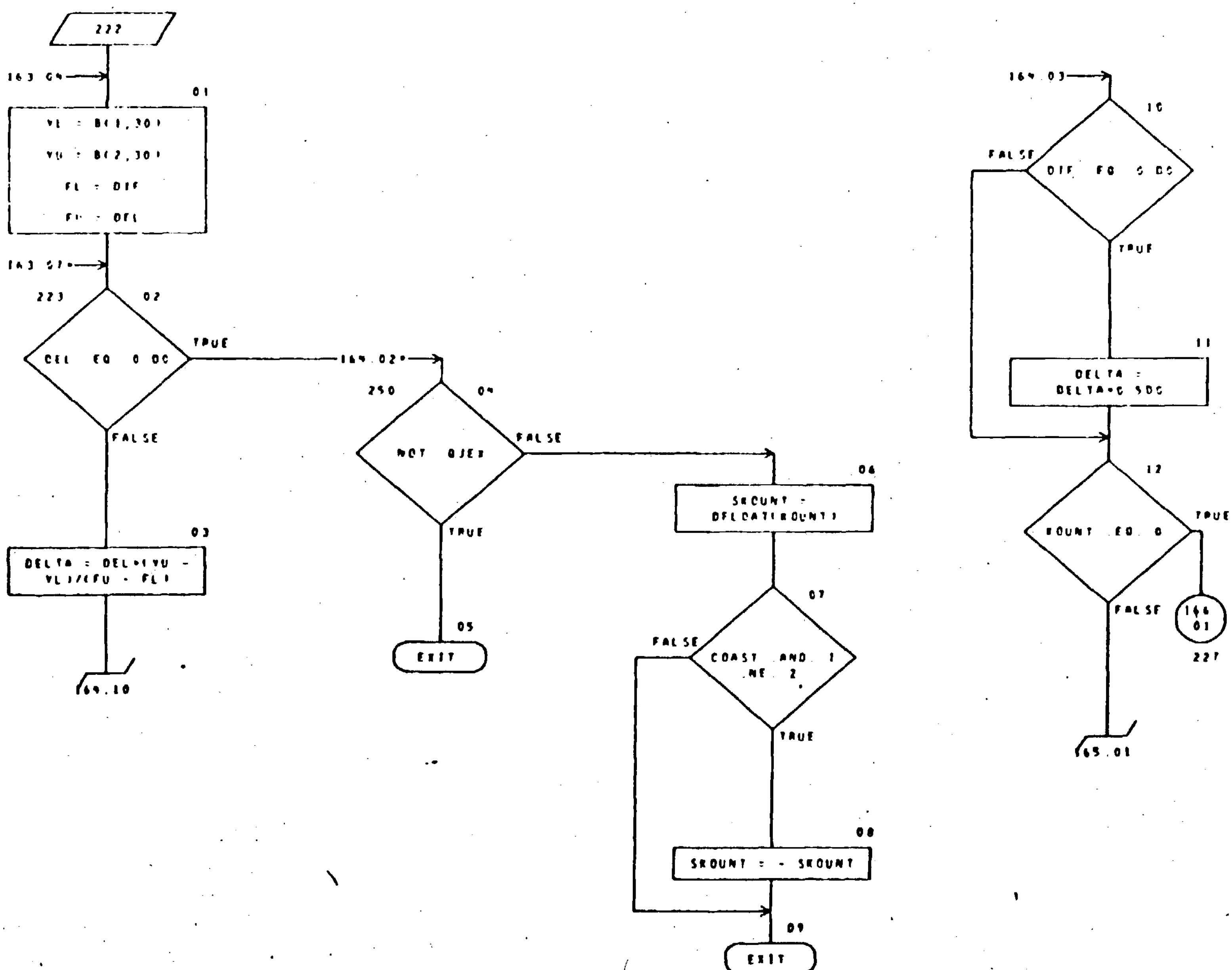
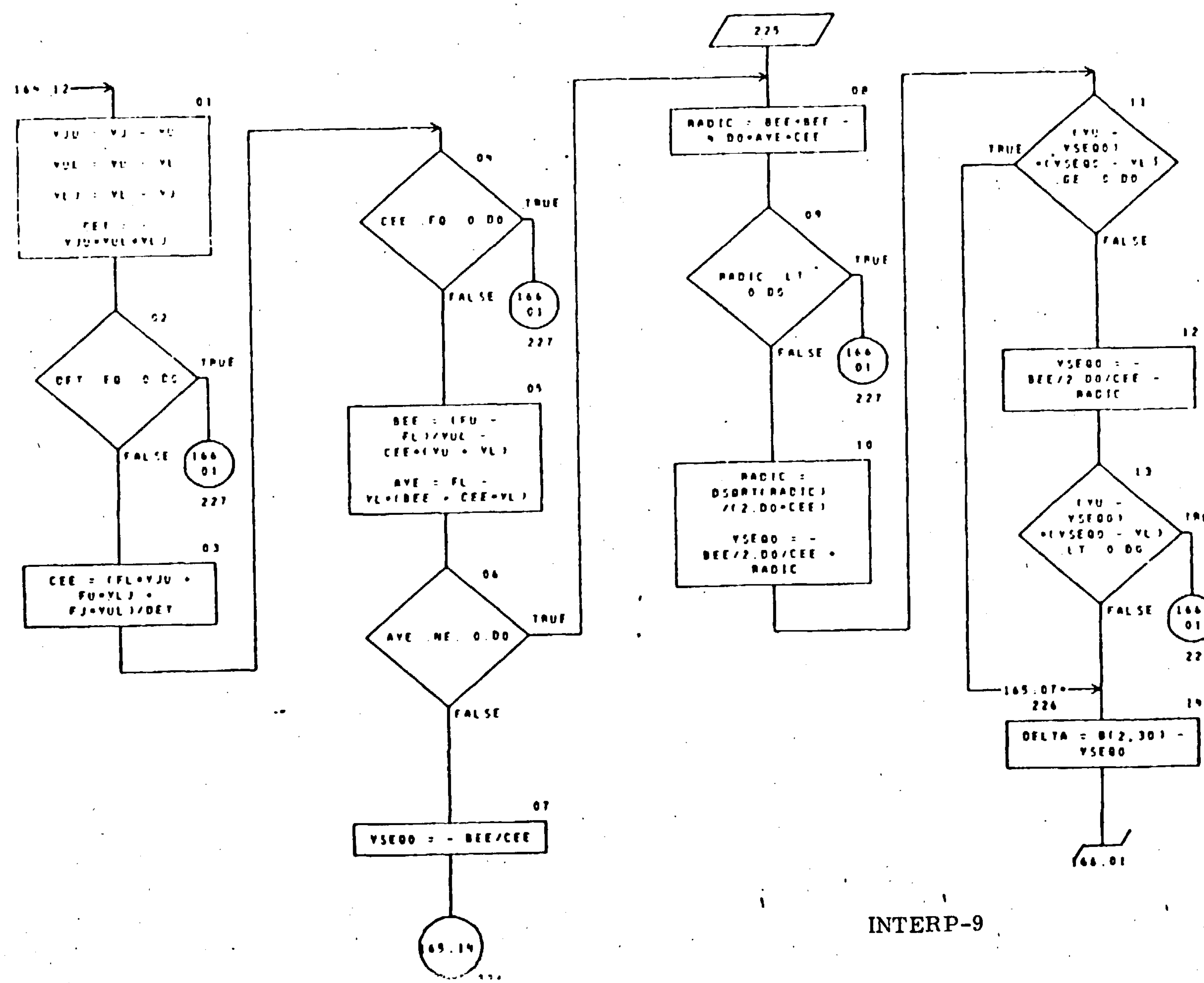


CHART TITLE - SUBROUTINE INTERP(I)



INTERP-9

CHART TITLE - SUBROUTINE INTERP(I)

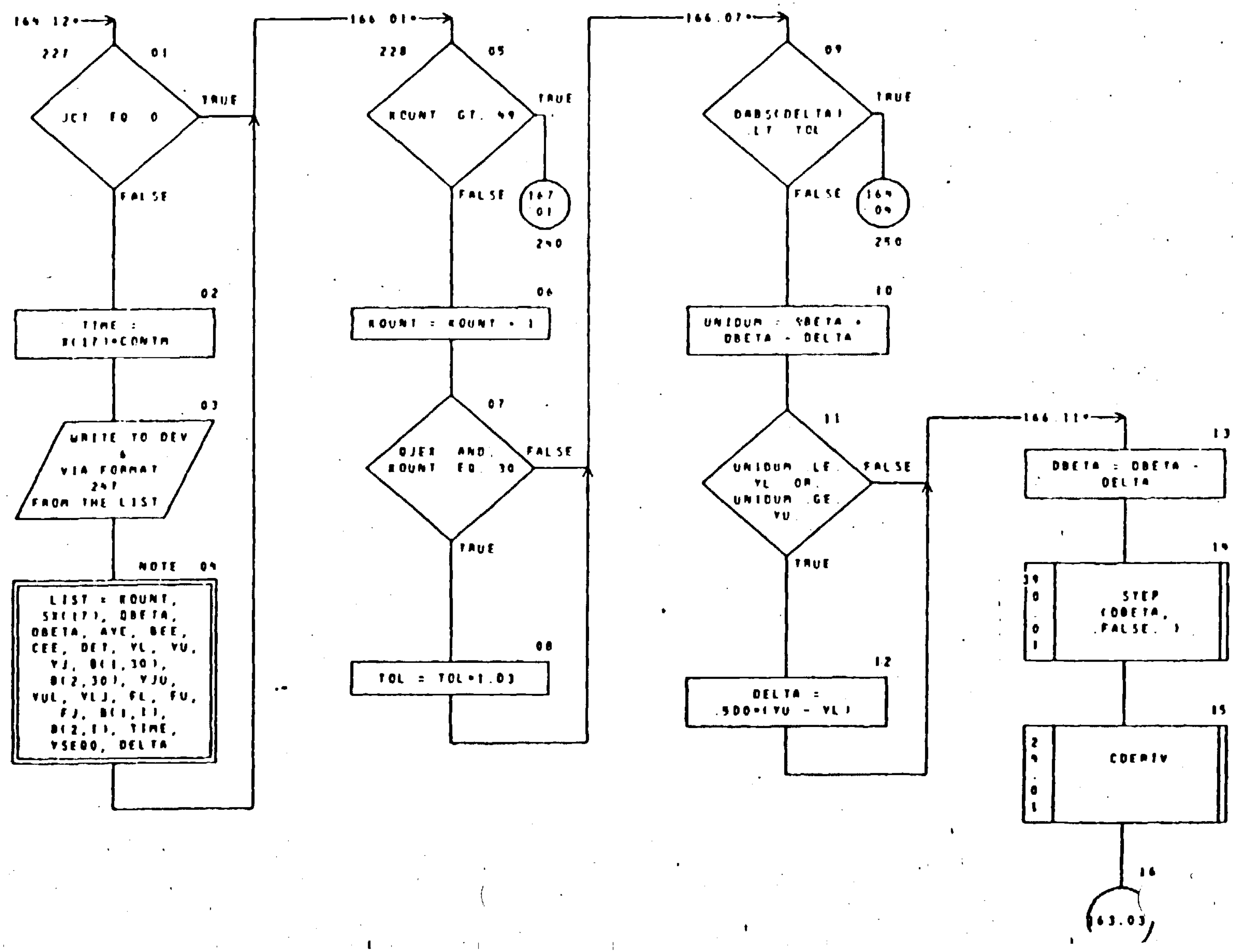
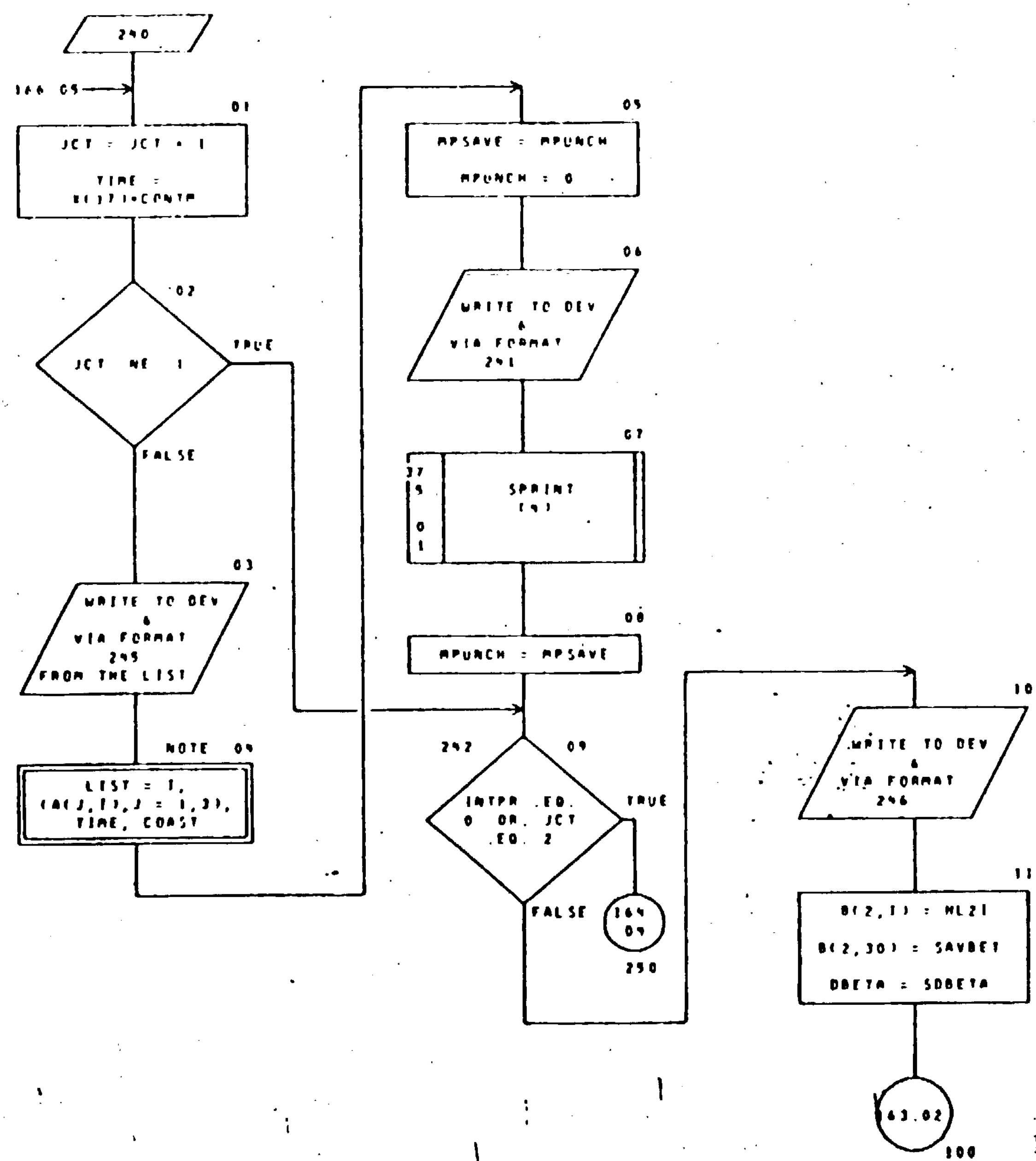


CHART TITLE - SUBROUTINE INTERP(I)



INTERP-11

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
         IMPLICIT REAL*8 (A-H,O-Z)
         LOGICAL COAST,OJEX
         DIMENSION A(3,30)
         COMMON /REALA/ RO1(85),DBETA,RO2(229),CONTM,RO3(857),SKOUNT,
         RO4(26),SX(50),X(50),RO5(50),SBETA,RO6(649)
         COMMON /INTGRS/ IO1(39),MPUNCH,IO2(85),INTPR,IO3(874)
         COMMON /LOGICS/ LO1(26),               OJEX,LO3(4),COAST,LO4(468)
         COMMON /EXTREM/ EO1(2660),B(2,30)
         DATA EPS /1.D-14/
         DATA A /
          8HDELTA TI,8HME (T-TI,8H)          ,8HTHRUST S,8HWITCH FU,8HNCTION
         ,8HTHR. SW ,8H FUNCT  ,8HDERIV.   ,8HCOMM  DI,8HSTANCE D,8HERIV
         ,8HSOLAR DI,8HSTANCE D,8HERIV.     ,8HTHRUST A,8HNGLE PSI,8H DERIV.
         ,8HTHRUST A,8HNGLE THE,8HTA DERIV,8HTHRUST A,8HNGLE PHI,8H DERIV.
         ,8HCRITICAL,8H SOLAR D,8HISTANCE ,8HPRIMER G,8HCING PAS,8HT RADIUS
         ,8HCOMM. AN,8HGLE DERI,8HVATIVE   ,8HINPUT PO,8HWER DERI,8HVATIVE
         ,8HPOWER FA,8HCTOR CUP,8HVE       ,8HARRAY ED,8HGEWISE F,8HUNCTION
         ,90*1H /
294      FORMAT(1H0,33HSPECIAL BLOCK PRINT AT FAIL POINT/1H )
295      FORMAT(1H0,45HITERATION FOR FUNCTION ROOT DID NOT CONVERGE.,4X
         9HFUNCTION(I2,4H) = 3A8,4X6HTIME =F19.12,3X7HCOAST =L2//1H )
296      FORMAT(1X,' KOUNT',11X,'TS',14X,'CSS',13X,'Z',15X,'AVE',
         13X,'BEE',13X,'CEE',13X,'DET'/1H ' YL',14X,'YU',14X,
         'YJ',14X,'Y(1)',12X,'Y(2)',12X,'YJU',13X,'YUL',13X,'YLJ'/
         1H ,' FL',14X,'FU',14X,'FJ',14X,'B(1,I)',10X,'B(2,I)',10X,
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
                'TIME',12X,'YSEQ0',11X,'DELTA'//)
 247            FORMAT(1H ,I2,14X,1P7D16.8/1X,1P8D16.8/1X,1P8D16.8//)
```

Name: LOAD PRECEDING PAGE BLANK NOT FILMED

Calling Argument: J, IC

Referenced Sub-programs: TRAVEL

Referenced Commons: EXTREM, INTGR4, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: CHECK

Discussion: This routine is called each time a special point is isolated (when comprehensive trajectory function monitoring under the QJEX indicator is invoked); all function-values to be printed are loaded into the general storage array after being converted to output dimensions.

Subroutine TRAVEL is called merely to compute and load the ecliptic longitude into the storage array. The indicator-locations of the storage array are set to code-values indicating maximum, minimum, or other remarkable point, corresponding to the isolated value in question. This information is transferred to auxiliary storage arrays in subroutine STORE and, finally, used in subroutine EXTAB.

LOAD EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(2, 30)	U	EXTREM	Array of values of functions monitored along the trajectory.
J	UX		Counter and index, corresponding to the j^{th} point isolated within the current computation step.
R(2)	UA	REAL8	Spacecraft solar distance, r, in AU, at current time (R(2)) and at start of current computation step (R(1)).
X(50)	UA	REAL8	Array of trajectory integrated variables.

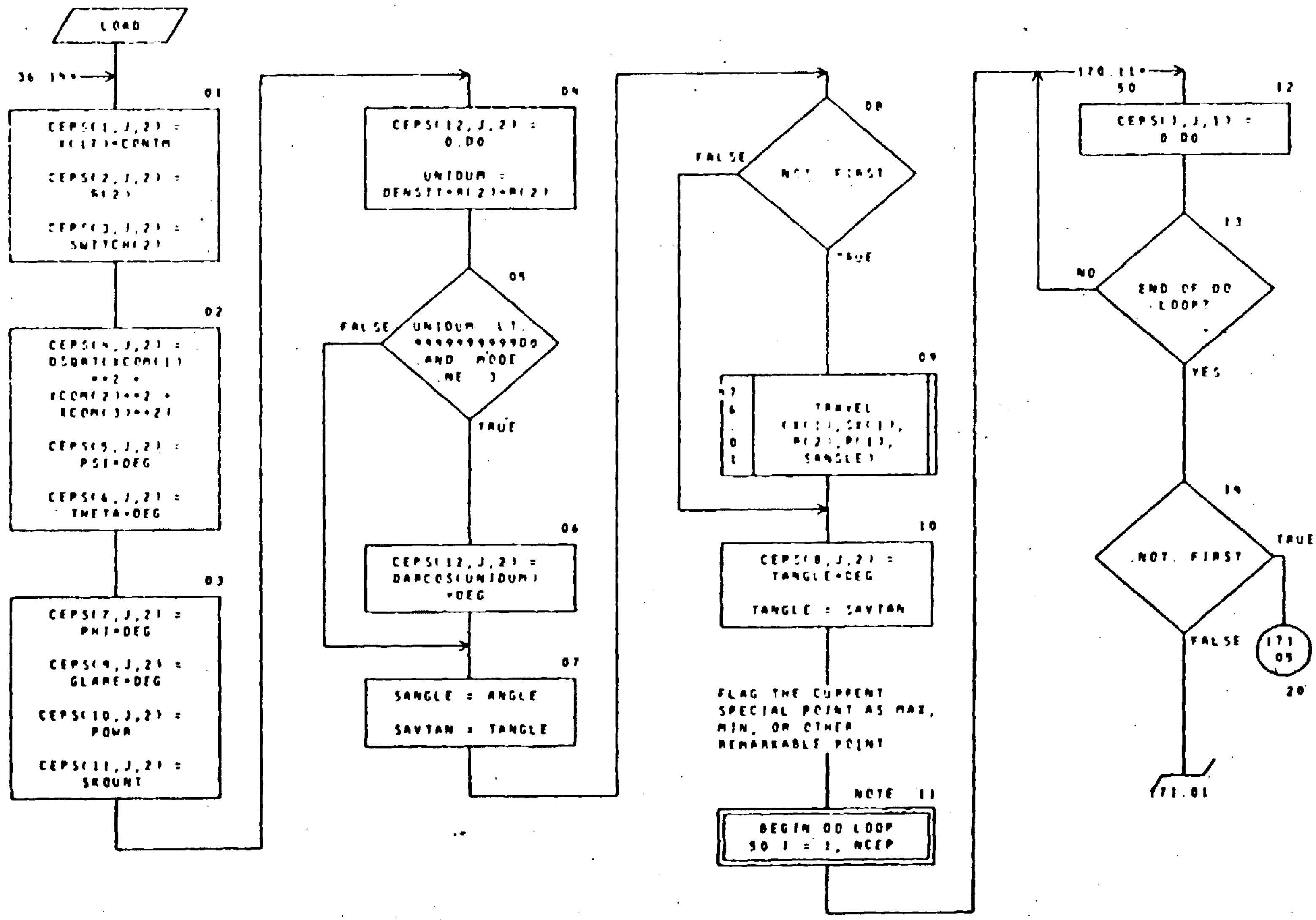
LOAD EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
IC	UX		Indicator which selects the particular trajectory function, the special value of which has been isolated.
JC	U	INTGR4	Counter corresponding to the JC th specified time-function value (i.e., the time) isolated thus far on the current trajectory segment. Attains values greater than one when imposed coast phases are invoked.
SX(50)	UA	REAL8	Array of trajectory integrated variables corresponding to the start of the current computation step.
DEG	U	REAL8	Radians to degrees conversion factor.
PHI	U	REAL8	Thrust angle ϕ, in radians.
PSI	U	REAL8	Thrust angle ψ, in radians.
CEPS (14,20,2)	S	EXTREM	General array of isolated points, containing function values and descriptive information associated with the current computation step.
HEAT	U	LOGIC4	Indicator for maintaining solar panels normal to sun during high solar proximity.
MODE	U	INTGR4	Power variation option selector.
NCEP	U	INTGR4	Maximum number of different parameters which are stored, to be printed.
POWR	U	REAL8	Power ratio, γq.
XCOM(6)	U	REAL8	Position and velocity vectors of spacecraft with respect to reference body, as specified by the program input quantity NDIST; i.e., communication distance vector.

LOAD EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ANGLE	U	REAL8	Travel angle, θ_t, in radians.
COAST	U	LOGIC4	Indicator for coast or thrust phase.
CONTM	U	REAL8	Tau to days conversion factor.
ERODE	U	LOGIC4	Indicator for power degradation.
FIRST	U	LOGIC4	Indicator for being at initial time of current trajectory segment.
GLARE	U	REAL8	Communication angle, in radians.
JCMAX	U	INTGR4	Maximum value of JC, corresponding to the end of the current trajectory segment.
PCURV	U	LOGIC4	Indicator for condition in which solar panels are in position to receive maximum power, when degradation option in invoked.
THETA	U	REAL8	Thrust angle θ, in radians.
DENSIT	U	REAL8	Power density, d, in AU^{-2}.
MAXPOW	U	LOGIC4	Indicator for mode of operation in which solar panels are maintained in position to receive maximum power, when degradation option is invoked.
SKOUNT	U	REAL8	Number of iterations in subroutine INTERP required to isolate the remarkable point.
SWITCH(2)	U	REAL8	Thrust switch function, σ (defined in similar manner to R(2)).
TANGLE	SU	REAL8	Ecliptic longitude, in radians.

CHART TITLE - SUBROUTINE LOAD(J,IC)



ORIGINAL PAGE IS
OF POOR QUALITY

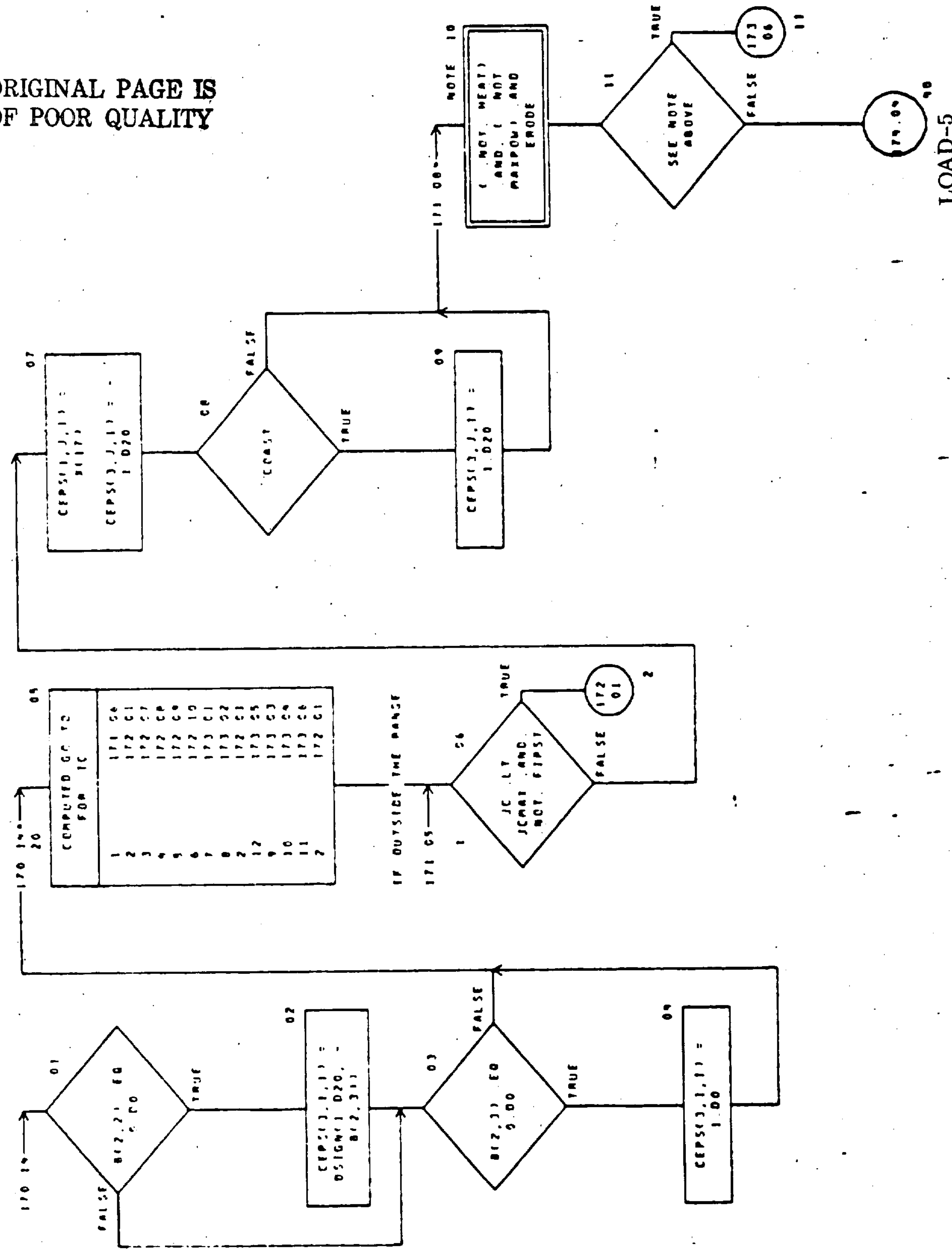


CHART TITLE - SUBROUTINE LOAD(J,IC)

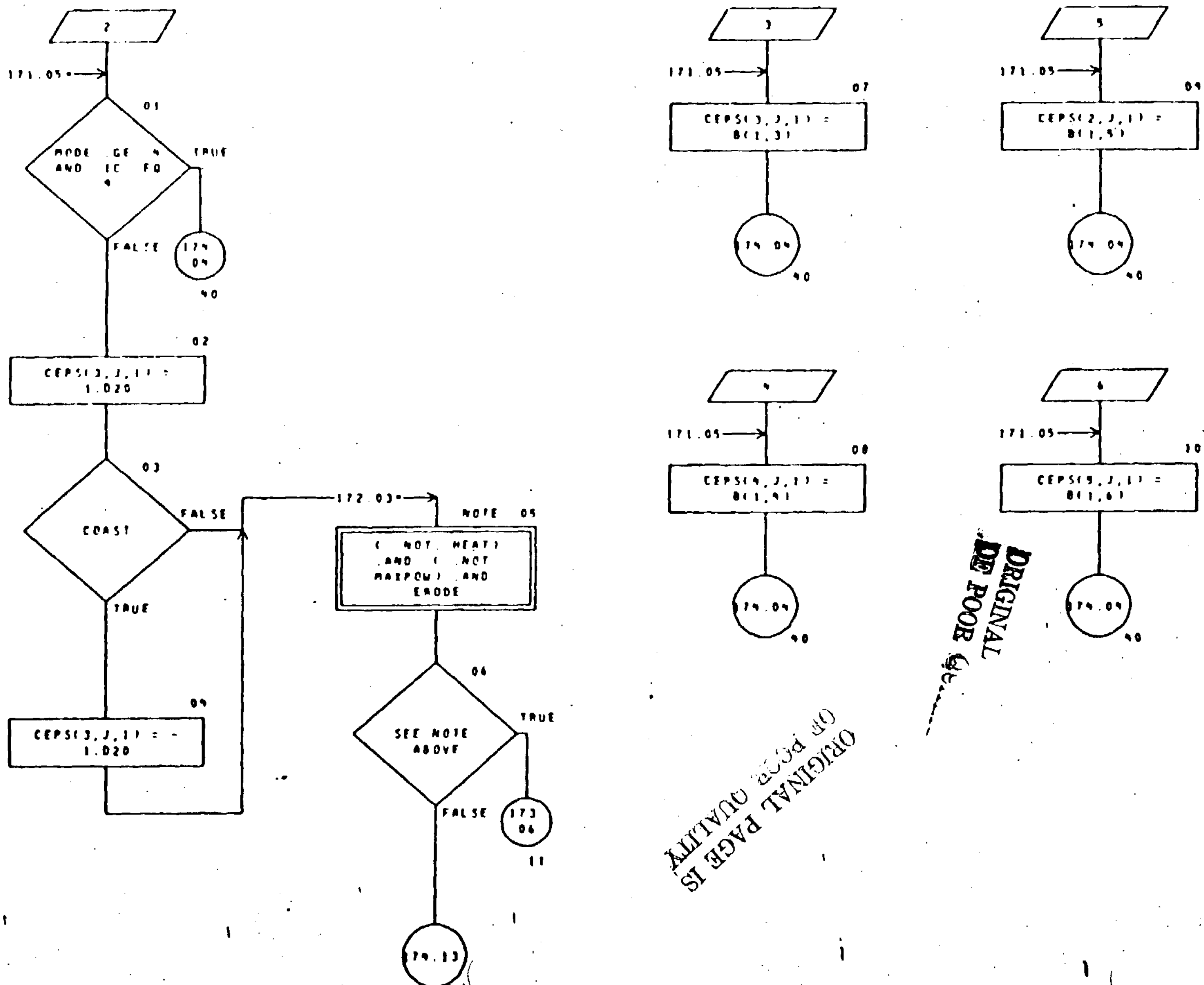


CHART TITLE - SUBROUTINE LOAD(J,IC)

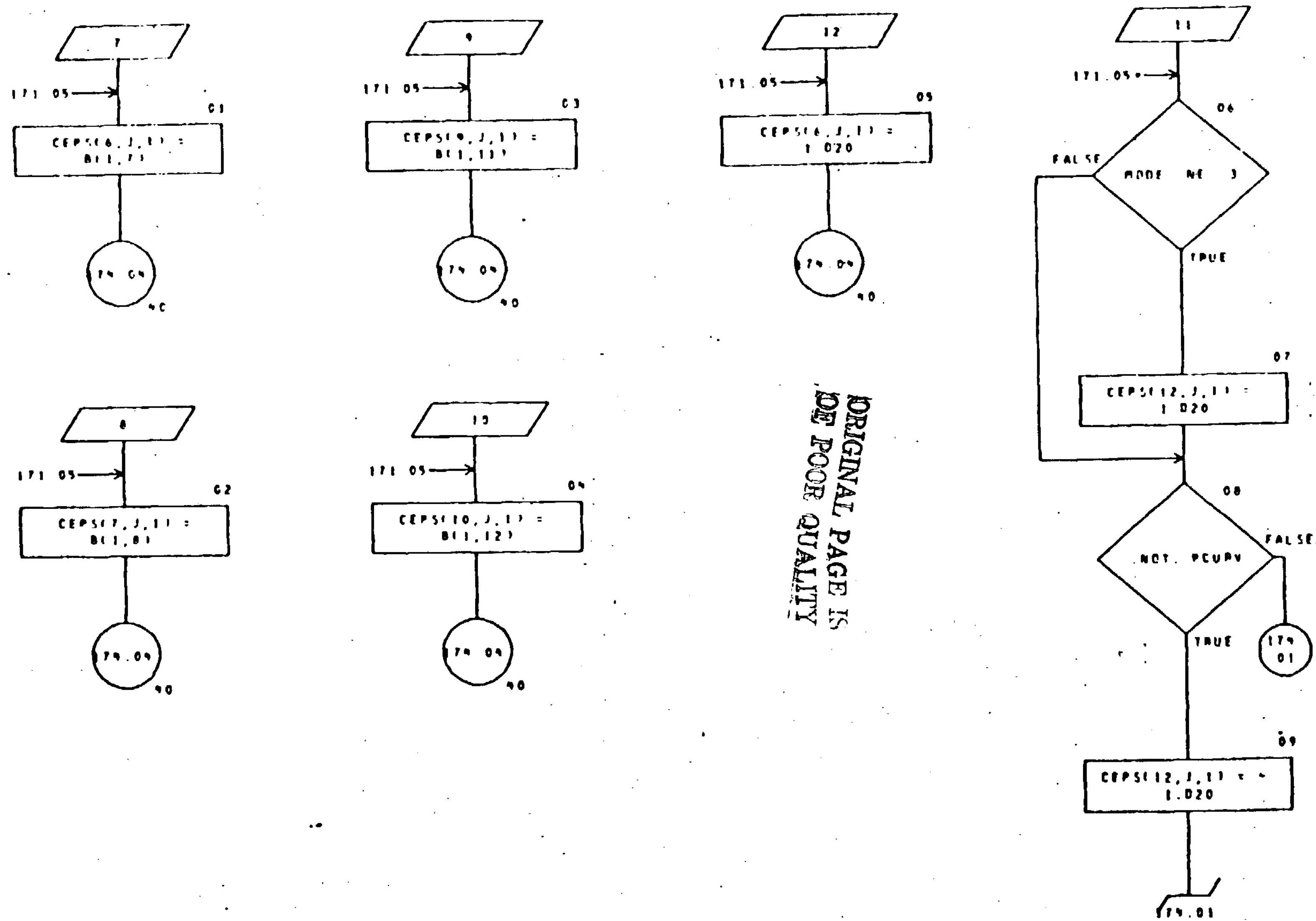


CHART TITLE - SUBROUTINE LOAD(J,IC)

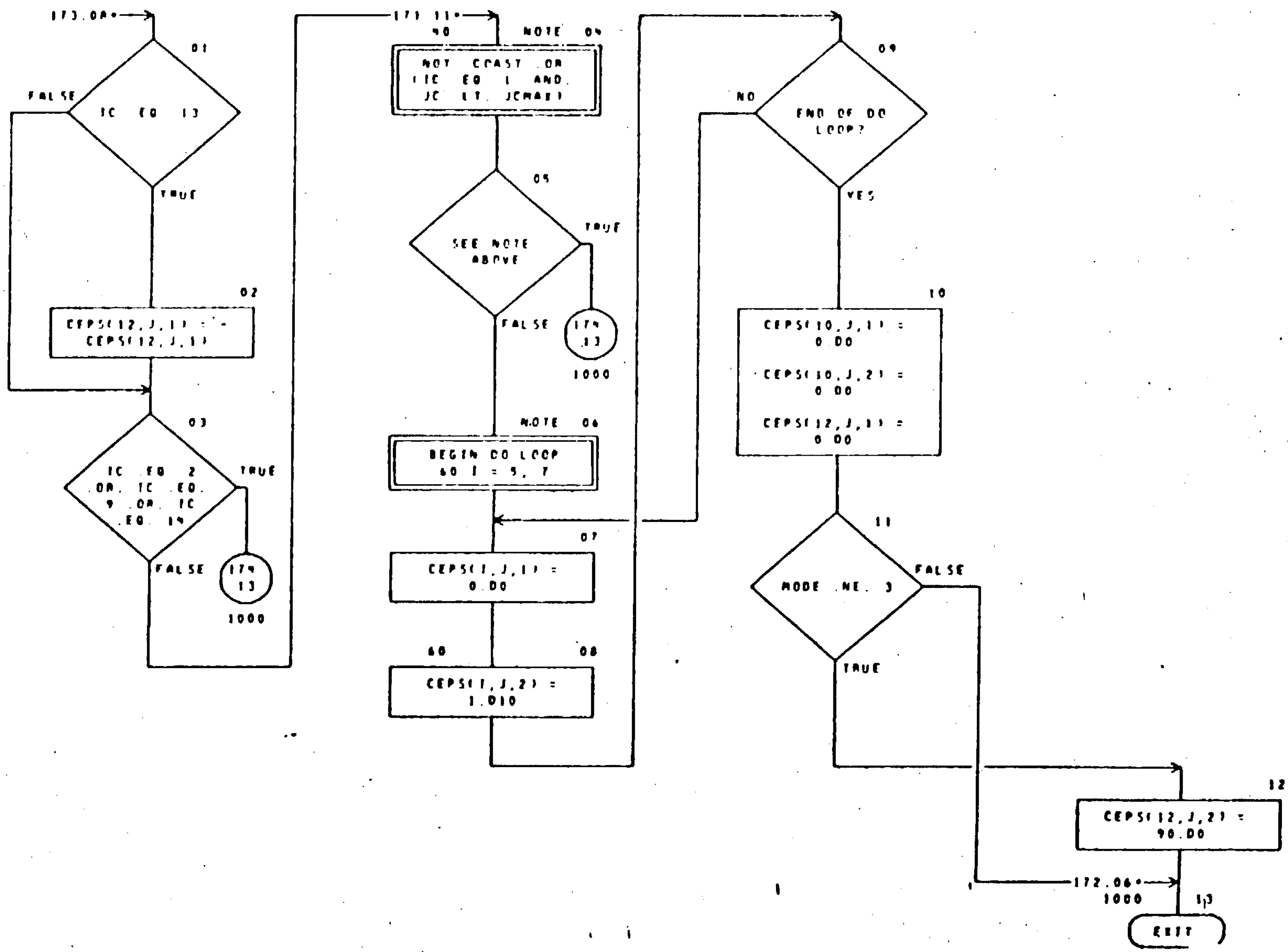


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL COAST,FIRST,ERODE,HEAT,PCURV,MAXPOW
      COMMON /REALA/ RO1(118),XCOM(6),GLARE,RO2(28),ANGLE,RO3(9),TANGLE,
      RO4(19),PSI,THETA,PHI,RO5(14),R(2),RO6(2),SWITCH(2),RO7(106)
                  ,DEG,RO9(2),CONTM,R10(838),POWR,R11(18),SKOUNT,
      R12(2),DENSIT,R13(23),SX(50),X(50),ROP(700)
      COMMON/INTGR4/IO3,MODE,IO1(326),NCEP,IO2(2),JC,ICMAX,IO4(667)
      COMMON /LOGIC4/ LO1(19),ERODE,LO3(8),HEAT,LON,FIRST,COAST,
      LO2(4),PCURV,MAXPOW,LO5(462)
      COMMON /EXTREM/ EO1(2100),CEPS(14,20,2),B(2,30)
```

LOAD-9

Name: MAIN

Calling Argument: Not applicable

Referenced Sub-programs: BEGIN, CORNER, FINISH, GUESS, INPUT, MINMX3, QSTART, READER, TIKTOK

Referenced Commons: INTGR4, LOGIC4, REAL8

Entry Points: Not applicable

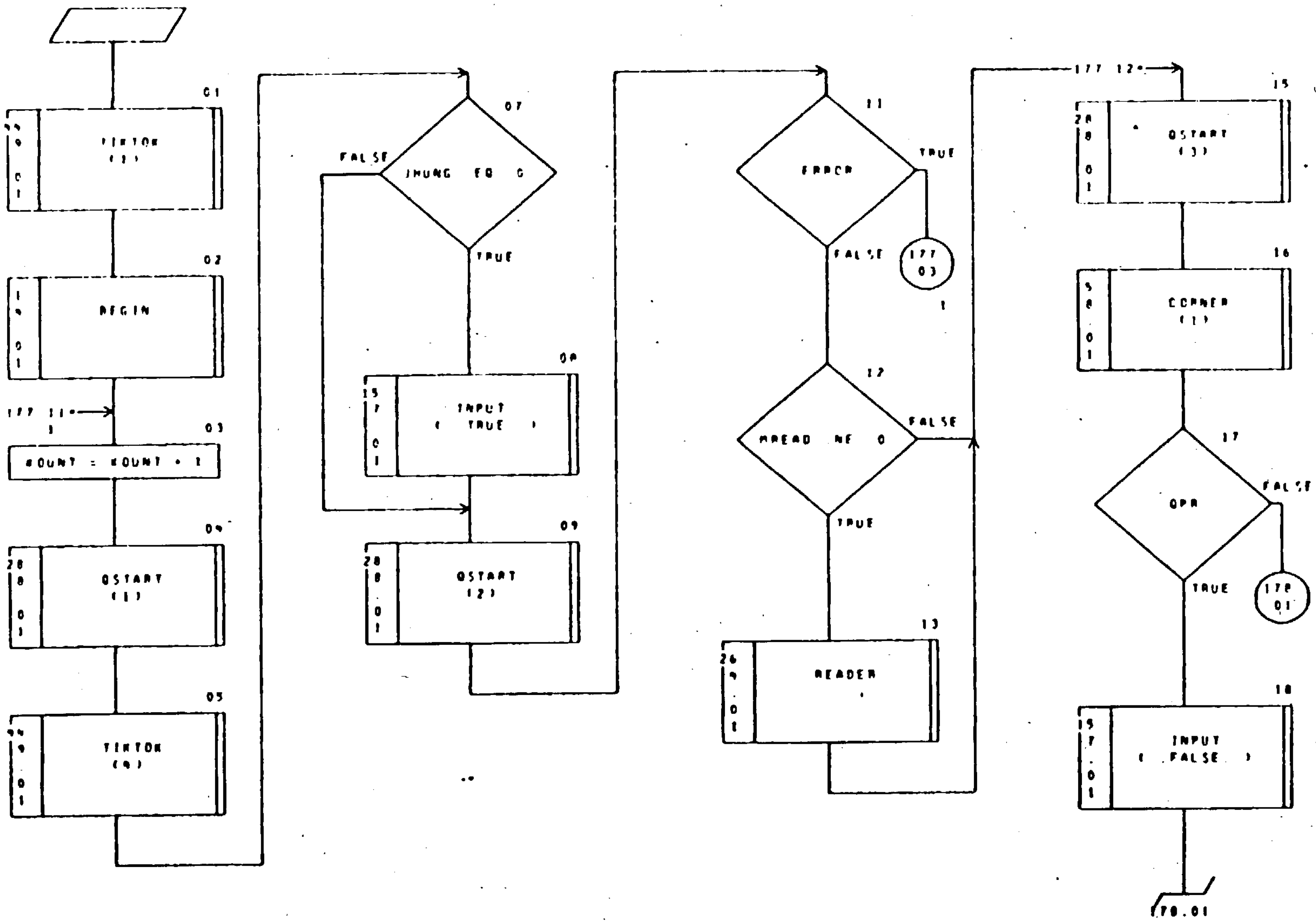
Referencing Sub-programs: Not applicable

Discussion: This main program provides the overall control for the program. Fundamental program constants are first generated by subroutine BEGIN. After this, the point of execution cycles through the MAIN program once per case, and a single computer run generally may consist of several cases. For each case, the program inputs are read in, required initializations and printouts are performed, the program's iteration logic optionally works toward obtaining a converged optimum rocket trajectory, and a final, summary trajectory is generated and printed.

MAIN EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
QPR	U	LOGIC4	Printout indicator, initialized in subroutine QSTART.
MOPT	UA	INTGR4	Ballistic option indicator, input to the program.
NSET(5)	UA	INTGR4	Iteration-sequence control array.
ERROR	UA	LOGIC4	Program master error indicator.
JHUNG	U	INTGR4	Propulsion corner control indicator, computed in subroutine CORNER.
KOUNT	SU	INTGR4	Case counter.
KPART	A	INTGR4	Indicator for option for automatically selecting improved independent parameter perturbations for generating the iterator's partial derivative matrix; an input to the program.
MPERF	A	INTGR4	Indicator which selects the quantity to be optimized when the iterator is operating in the improve mode.
MREAD	U	INTGR4	Indicator for reading the trajectory starting parameter values into the program from cards (unit 5).
STATE(6)	A	REAL8	Array containing the Cartesian position and velocity components of the primary target, in the ecliptic system, in AU and AU/tau, input to the program.
CONVRG	A	LOGIC4	Iterator convergence indicator.

CHART TITLE - PROCEDURES



MAIN-3

CHART TITLE - PROCEDURES

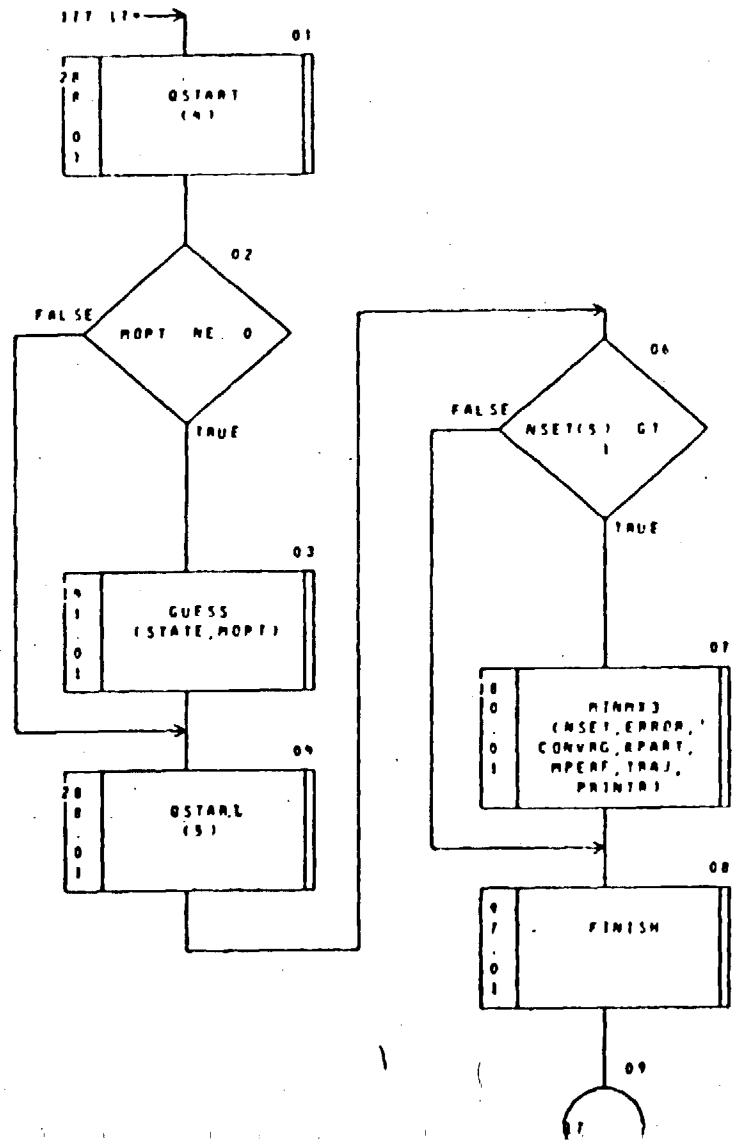


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL ERROR,CONVRG,OPR
COMMON /REAL8/ RO1(44),STATE(6),RO2(1950)
COMMON /INTGR4/ IO1(7),MOPT,IO2(4),MREAD,IO3(2),NSETS(5)
,IO4(29),KPART,IO5(3),KOUNT,IO6(11),JHUNG,IO7(77),MPERF,IO8(856)
COMMON /LOGIC4/ ERROR,CONVRG,LC1,OPP,LC2(496)
EXTERNAL TRAJ, PRINTR
```


MAIN-5

Name: MINMX3

Calling Arguments: MSET, XERROR, XCONVG, KPART, MPERF, PD5, CCHECK

Referenced Sub-programs: CCHECK*, PARINC, PD5**, PMPINT, PMPRNT, SIMEQ, SMQINT, TIKTOK

Referenced Commons: INTGR4, ITERAT, ITER2, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: GUESS, MAIN, SWING

Discussion: MINMX3 is the subroutine which drives the two-point boundary value problem to a solution. The iterator's underlying mathematical analysis is formulated as follows. Let X denote the vector of independent variables and let Y denote the vector of dependent variables. The relationship between these two vectors is given by

$$Y = F(X).$$

The vector function, F, is evaluated by integrating the trajectory; that is, given a complete set of control parameters and initial conditions, the corresponding values of the end conditions Y can be determined. Subroutine PD5 maps X onto Y and is therefore a software package which generates the function F. The problem is to find the vector X* which will result in specified values of the dependent variables Y*, that is, to solve

$$Y^* = F(X^*),$$

where Y* is known. This is formulated as a minimization problem. The weighted sum of the residuals q_i is given by

$$q_i = \left[Y^* - F(X_i)\right]^T W_y \left[Y^* - F(X_i)\right],$$

where X_i is the current estimate of the independent variables and W_y is a diagonal, positive definite weighting matrix.

*CCHECK is PRINTR, IMPRNT, or SWPRNT;
**PD5 is TRAJ, IMPULS, or SWTRAJ.

The problem is to choose a new value X_{i+1} to minimize q_{i+1}. If X_{i+1} is close to X_i, then

$$F(X_{i+1}) = F(X_i) + P\Delta X,$$

where $\Delta X = X_{i+1} - X_i$ and the partial derivative matrix, P, is given by

$$P = \frac{\partial Y}{\partial X} .$$

Evaluating q_{i+1} with this approximation leads to the expression,

$$q_{i+1} = (\Delta Y - P\Delta X)^T W_y (\Delta Y - P\Delta X),$$

where ΔY, the residual vector, is given by

$$\Delta Y = Y^* - F(X_i) .$$

The problem is then to choose ΔX to minimize q_{i+1}.

For nonlinear functions F, linear approximations work best if ΔX is small. Therefore, the following constraint is imposed,

$$\Delta X^T W_x \Delta X \le \ell ,$$

where W_x is the input diagonal, positive definite weighting matrix associated with the independent parameters.

Attaching the constraint with a positive scalar inhibitor, λ, the quantity to be minimized is given by

$$q = (\Delta Y - P\Delta X)^T W_y (\Delta Y - P\Delta X) + \lambda (\Delta X^T W_x \Delta X).$$

Finding the minimum of the function yields the solution,

$$\Delta X = (P^T W_y P + \lambda W_x)^{-1} P^T W_y \Delta Y.$$

This equation is solved by subroutine SIMEQ. It can be shown that as λ increases, ℓ decreases monotonically. Therefore, λ can always be chosen large enough to satisfy the above inequality. Moreover, if λ is sufficiently large, the correction is approximately

$$\Delta X = \frac{1}{\lambda} W_x^{-1} (P^T W_y) \Delta Y.$$

For ΔX small enough, or λ large enough, we are guaranteed that

$$q_{i+1} \leq q_i .$$

It is advantageous to take as large a step toward satisfying $Y^* - F(X^*)$ as possible. The procedure is initiated with a relatively small value of λ. The idea is to make a correction, determine if any improvement is made, and, if not, cut back on the correction. The following iteration scheme is utilized. Given X_i, the mapping F is executed again to produce Y_{i+1} starting with the values $X_{i+1} = X_i + \Delta X$, and q_{i+1} is calculated. q_{i+1} is then compared with q_i. If there is no improvement, λ is increased, ΔX is recalculated and a new mapping is executed. This is repeated until an improvement results. When this happens, the mapping is executed again and the partial derivative matrix is computed. λ is reduced by a factor of 64. The iteration continues until either the end conditions are satisfied within the prescribed tolerance or no significant improvement can be made or the maximum number of iterations is exceeded.

The constraints, Y, are divided into two types, parameters that are driven to a given value (point constraints) and parameters to be maximized or minimized (performance indices).

For a well-posed problem, there is only one performance index. For each dependent variable, y_i, two values must be specified, y_{min} and y_{max}, which define the acceptable range. If a dependent variable is a point constraint, y_{min} and y_{max} are chosen close together

$$y_{min} = y^* - \frac{\delta}{2} ; \quad y_{max} = y^* + \frac{\delta}{2} ,$$

where y^* is the desired value and δ is the neighborhood-width, or tolerance, utilized for weighting purposes. For the performance index, the interval is chosen so that it cannot possibly be attained if the other constraints are satisfied. For instance, if y is to be minimized, y_{min} and y_{max} are taken smaller than attainable, conversely if y is to be maximized, y_{min} and y_{max} are taken larger than attainable. In this way the iteration procedure drives the variable to be optimized in the correct direction until no significant improvement is possible or the input maximum number of iterations is exceeded.

Two modes of solution are available, the satisfy mode and the improve mode. In the satisfy mode, the iterator attempts only to satisfy the point constraints. The improve mode adds the performance index to the end conditions, and, through proper weighting, generates trajectories with an improved performance index while maintaining satisfaction, or near satisfaction, of the point constraints.

The scale matrices W_x and W_y are used to make elements of the vectors X and Y compatible for the iteration procedure. The relative importance of the variables is represented in this way. Differing magnitudes are compensated for through the weighting matrices. W_x is input to the program, W_y is computed internally using the input tolerances and importance factors. For point constraint variables, the elements of W_y are given by the following relation:

$$W_y = \frac{2^{-40}}{(\delta_y/2)^2} ,$$

where δ_y is the corresponding tolerance (neighborhood full-width). The weighting factor for the performance index is computed from

$$W_y = \frac{nm}{r^2} 2^{-40} ,$$

where r is the performance index residual, n is the number of variables to be satisfied, and m is 10^{-4} when the iterator is operating in the satisfy mode and 256 when in the improve mode. This balances the residual in the parameter being optimized against the weighted residuals in the other variables, to satisfy the constraints as the optimization proceeds.

Messages and printouts: Several diagnostic and informative messages are provided to describe the progress made by the iterator or any problems encountered. In all cases, upon exiting the routine, the trajectory counters, KOUNT and L, are printed on unit 6 in the following message:

____TRAJECTORIES WITHOUT PARTIAL DERIVATIVES AND____TRAJECTORIES WITH PARTIAL DERIVATIVES REQUIRED FOR THIS CASE.

and on unit 12 in the form:

____TRAJECTORIES WITHOUT AND____WITH PARTIALS.

If the problem does not involve a performance index and all point constraints are satisfied, or if a performance index is being optimized and no further improvement is possible, the following message is printed on units 6, 11, and 12:

THIS CASE IS CONVERGED.

Upon attainment of convergence in the satisfy mode and just prior to starting the improve mode, the message

ITERATOR IS NOW IN IMPROVE MODE.

is printed, on units 6, 11, and 12.

If the point constraints cannot be satisfied in the satisfy mode, the message

THIS CASE WILL NOT CONVERGE.

is printed on the same three units.

If the number of iterations in either mode exceeds the input limit, the following message is printed on the three units:

MAXIMUM NUMBER OF ITERATIONS EXCEEDED.

If an error condition is detected in PD5 on the first trajectory (usually resulting from input errors), the following message is printed on units 6, 11, and 12:

FIRST GUESSES WILL NOT RUN TRAJECTORY.

If an error condition is detected in PD5 while generating the nominal or perturbation trajectories, the following message is printed on unit 6:

ERROR IN PARTIAL DERIVATIVE CALCULATION.

If an error condition occurs when attempting to optimize the independent-variable perturbation step sizes via the option invoked by program-input KPART, the message is printed on unit 6:

TRAJ ERROR AFTER PARINC.

Any one of the last five messages above will also generate the message:

ITERATOR IS GIVING ERROR RETURN.

and indicates an error condition to be returned from MINMX3 through the calling argument XERROR.

If either the number of independent variables or the number of dependent variables exceeds 35, the message

n VARIABLES INVALID IN MINMX3. RUN TERMINATED.

is printed on units 6 and 12, and execution is immediately terminated. n is the number, greater than 35, of either independent or dependent variables which were attempted.

Whenever any alteration of any independent-variable perturbation step size occurs as a consequence of proximity of a trajectory to a propulsion-time corner, assuming the program input NSWPAR is activated, the message

MINOR __m__ OLD DX = __a__ NEW DX = __b__

is printed on unit 6, in which m is the number identifying the m^{th} neighboring (minor) trajectory being generated (in the same order as printed on the "ITERATOR PARAMETERS" page), "a" is the prior step size value, and "b" is the altered value.

When special print is invoked via the program input ITPRNT, the following is output on unit 6 for each nominal trajectory:

MINMX3 SPECIAL PRINT L, KOUNT, LAMBDA, RLOLD, RLNEW, RLMAX, DK, SCALE

____ ____ ____ ____ ____ ____ ____ ____

in which L is the number of trajectories which have been generated in the current iteration-sequence having partial derivatives, KOUNT is the number of trajectories without partial derivatives, LAMBDA is the value of the inhibitor, λ, RLOLD is the prior weighted sum of the residuals, q_i(old), RLNEW is the current weighted sum of the residuals, q_i, RLMAX is $n\,2^{-40}$ (see text above for nomenclature), DK is n, and SCALE is 2^{-40}.

MINMX3 EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(35)	SU	ITER2	Array of independent variable values, X, corresponding to program input quantities, Xi(1).
Q(35)	SU	ITER2	Array of dependent variable values, Y, corresponding to program input quantities Yi(1), which are the desired values.
BS(35)	U	ITER2	Array of maximum step-sizes for the independent variables, corresponding to program input quantities Xi(3).

MINMX3 EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
BW(35)	U	ITER2	Array of independent-variable weighting factors, corresponding to program input quantities Xi(5).
MM(70)	U	INTGR4	Index set of active dependent variables.
PM(1225)	SUA	ITER2	Array containing the negative of the partial derivative matrix, -P.
BBB(35)	SU	ITER2	Array of independent-variable perturbation step sizes, corresponding to program input quantities Xi(4).
NPR(4)	U	INTGR4	Printout control vector.
NSW	U	INTGR4	Total number (in MINMX3) of thrust-switching points along the trajectory recently generated.
PD5	UX		Name of mapping subroutine, F.
YYY(35)	SU	ITER2	Array of dependent variable values corresponding to the nominal trajectory; the nominal trajectory is a trajectory which the neighboring trajectories are generated with respect to.
FXL1(70)	S	ITERAT	Array of dependent variable values corresponding to the initial nominal trajectory, eventually used in measuring the progress of the iteration sequence.
HUNG	U	LOGIC4	Indicator for the condition in which the iteration sequence is making no significant progress because the trajectory is in high proximity to a propulsion corner; "hung on a thrust phase or a coast phase".
LINE	SU	INTGR4	Counts the number of lines which have been written on unit 11.

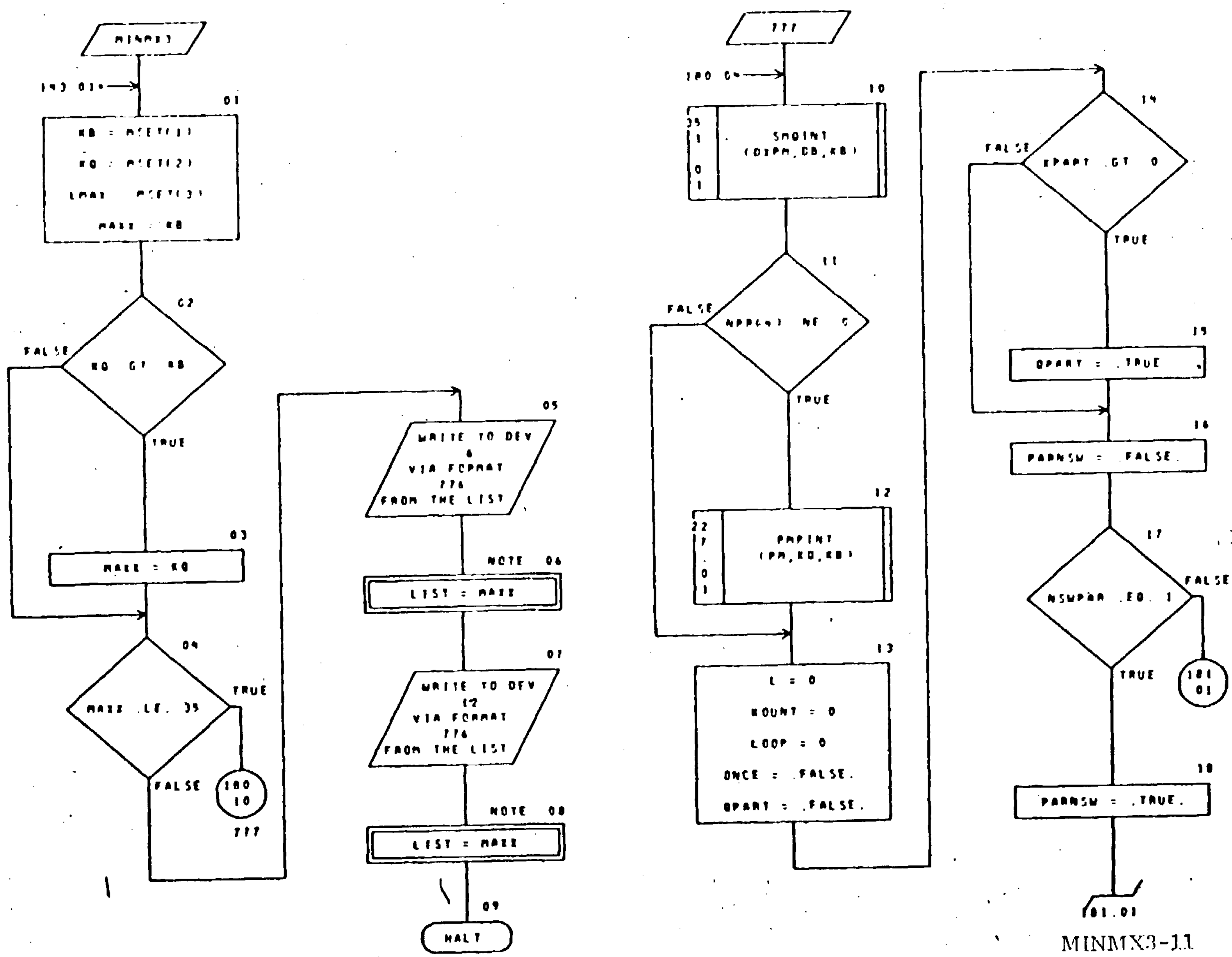
MINMX3 EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
LOOP	SUE		Counts the number of neighboring trajectories which have been generated, associated with the current nominal trajectory (or mapping).
MSET(5)	UX		Control array: (1) Number of independent variables. (2) Number of dependent variables. (3) Maximum number of iterations permitted in the satisfy mode. (4) Indicator for starting in the satisfy (=0) or improve (≠0) mode. (5) Maximum number of iterations permitted in the improve mode.
NSWX(50)	S	INTGR4	Array of stored values of NSW, used at the end of the iteration sequence to print the "switch count history".
QMAX(35)	U	ITER2	Array of upper allowable values for the dependent variables.
QMIN(35)	U	ITER2	Array of lower allowable values for the dependent variables.
BNOMX (35)	S	ITER2	Array of independent-variable stored values corresponding to the current nominal trajectory.
KOUNT	SUE		Counter of the number of trajectories generated without partial derivatives in the current iteration sequence.
KPART	UAX		Program input quantity indicating the desire to optimize the independent-variable perturbation step sizes.
MPERF	UX		Array index indicating the performance index among the dependent variables, used only in the improve mode.

MINMX3 EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
CCHECK	UX		Name of print subroutine.
ITPRNT	U	INTGR4	Program input quantity which invokes special print from MINMX3.
LAMBDA	SUE		The inhibitor, λ.
NSWPAR	U	INTGR4	Program input quantity indicating desire to control independent variable perturbation increments.
WONDER	U	LOGIC4	Indicator for bypassing all tests associated with the condition related to the indicator HUNG.
XCONVG	SX		Convergence indicator output from MINMX3.
XERROR	SX		Error indicator output from MINMX3.

CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,KPART,MPERF,PDS,CCHECK)



MINMX3-11

CHART TITLE - SUBROUTINE MINME3(MSET,IERROR,ICONVG,KPART,MPERF,PDS,CCHECK)

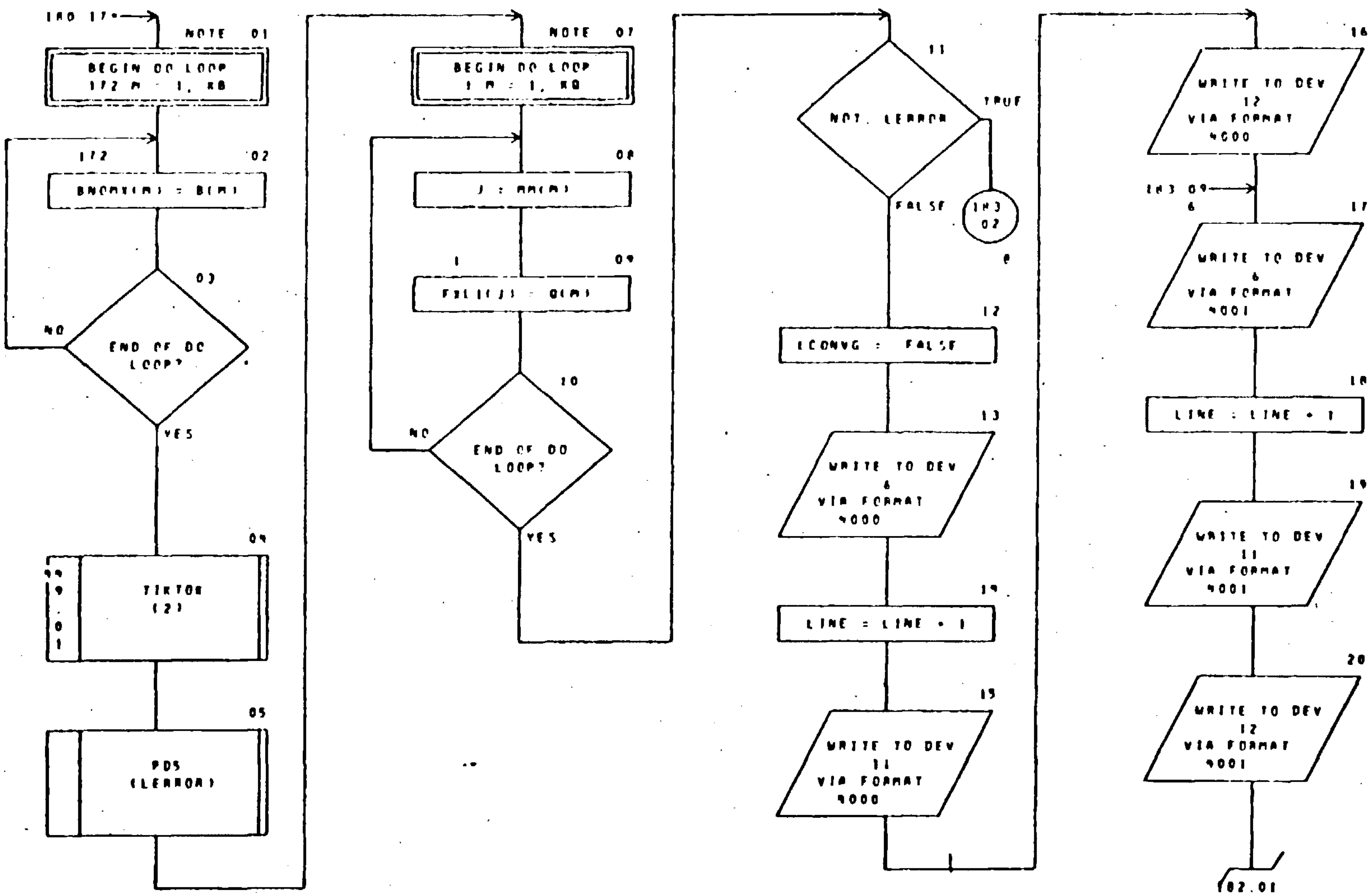
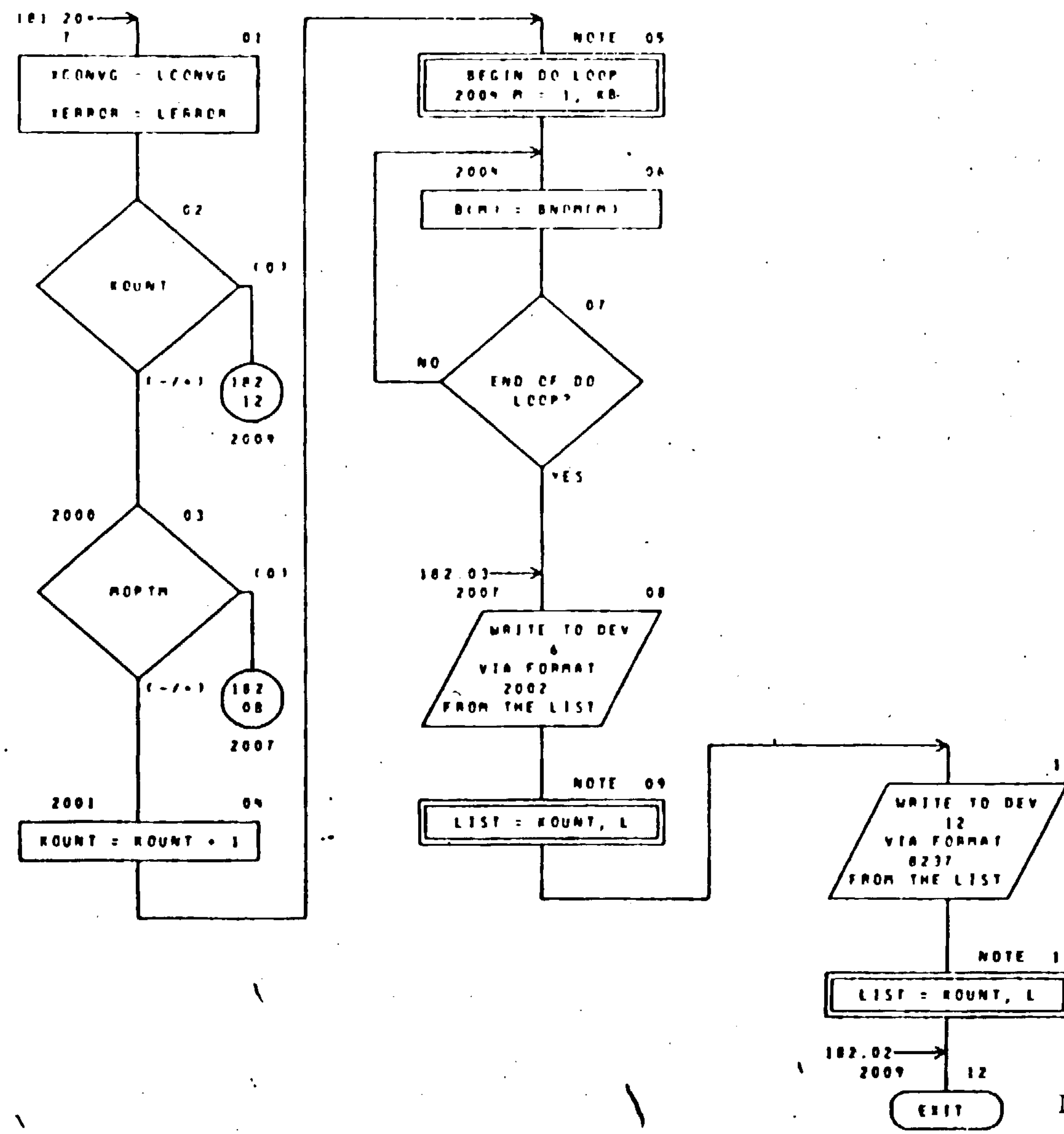


CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,KPART,MPERF,PDS,CCHECK)



MINMX3-13

CHART TITLE - SUBROUTINE MINMX3(MSET,KERROR,KCONVG,KPART,MPERF,PDS,CCHECK)

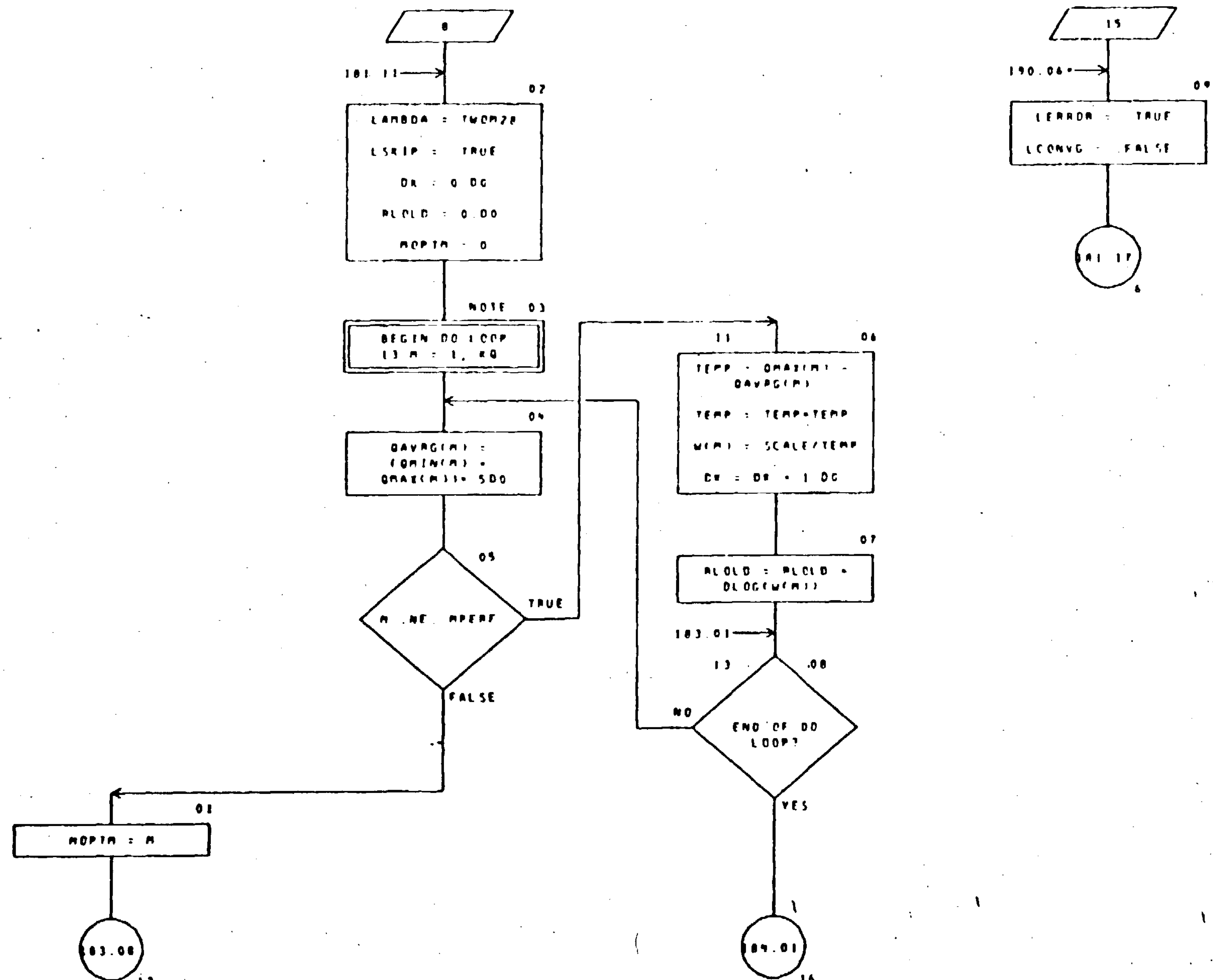
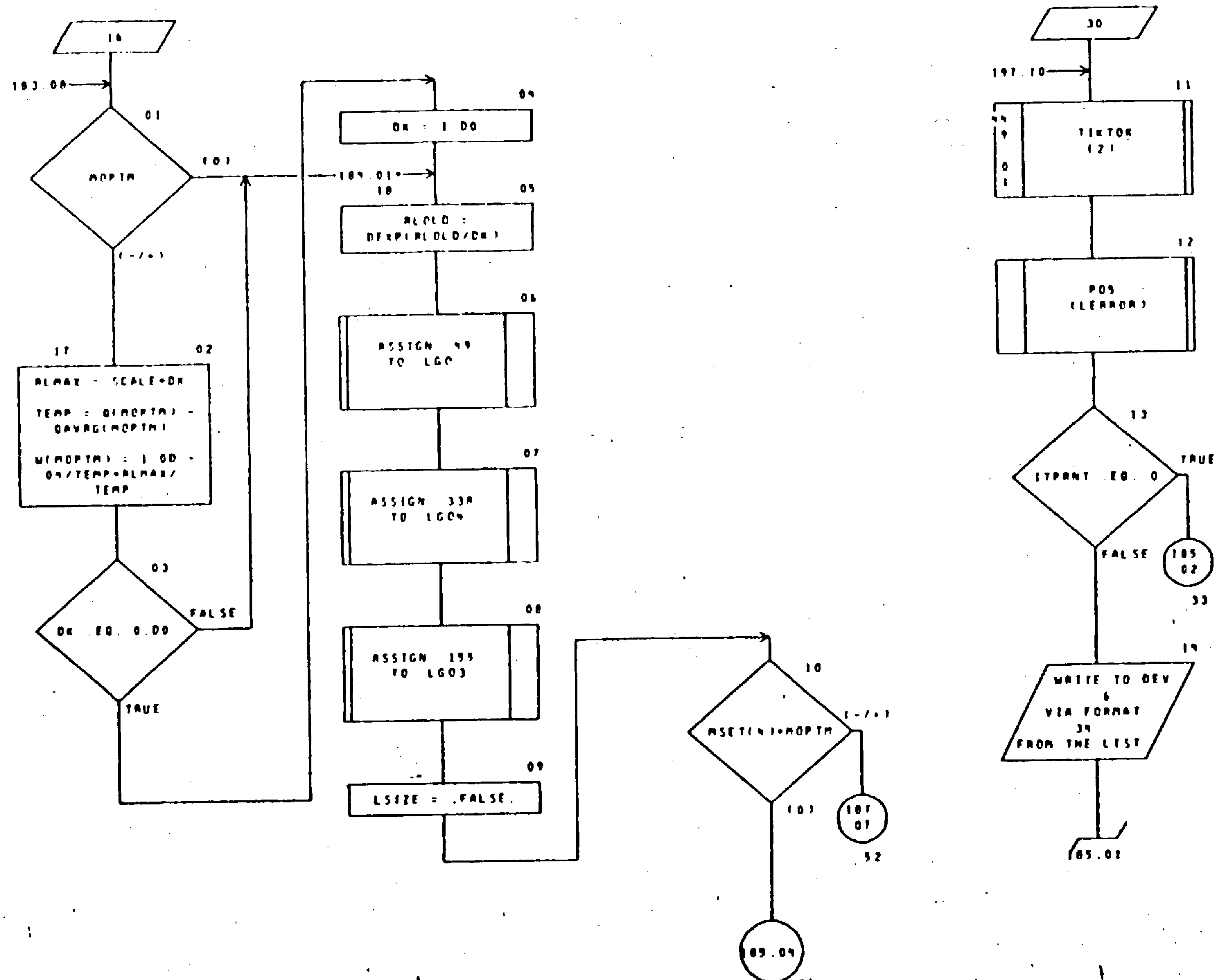


CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,KPART,MPERF,PDS,CCHECK)



MINMX3-15

CHART TITLE - SUBROUTINE MINMX(IMSET,KERROR,KCONVG,KPART,MPERF,PDS,CCHECK)

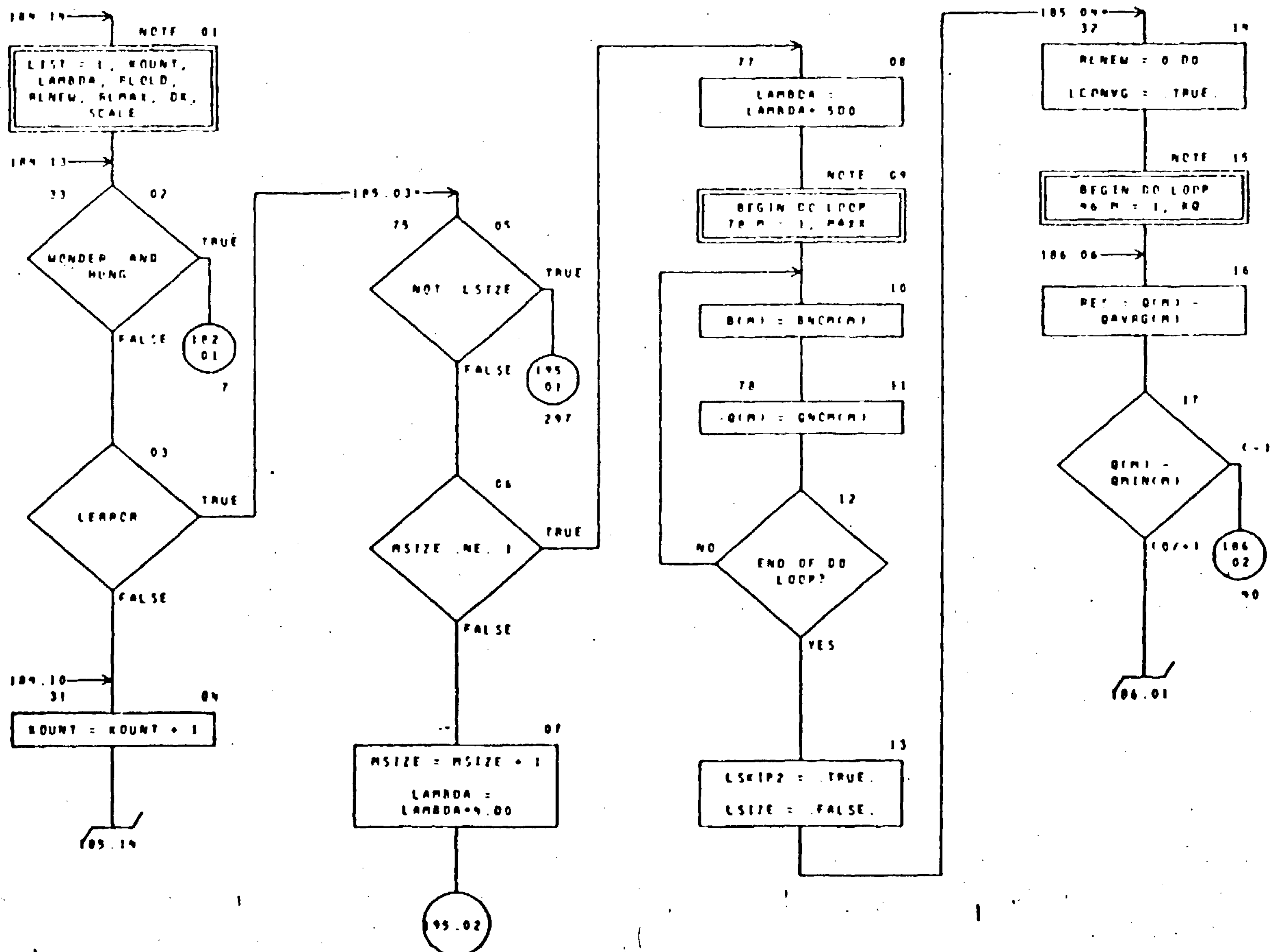
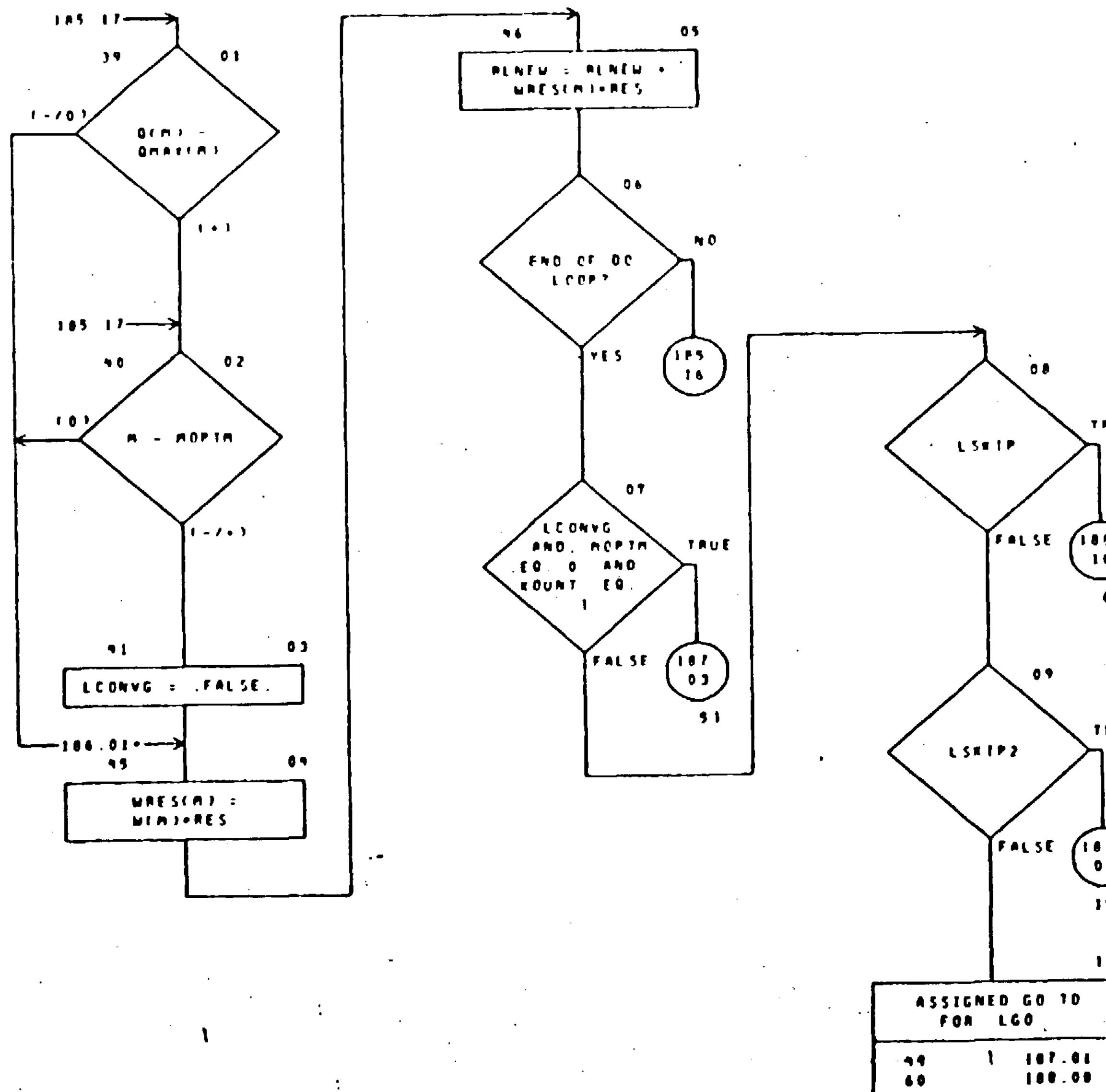


CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,KPART,MPERF,POS,CCHECK)



MINMX3-17

CHART TITLE - SUBROUTINE MINMX3(MSET,KERROR,KCONVG,KPART,MPERF,POS,CCHECK)

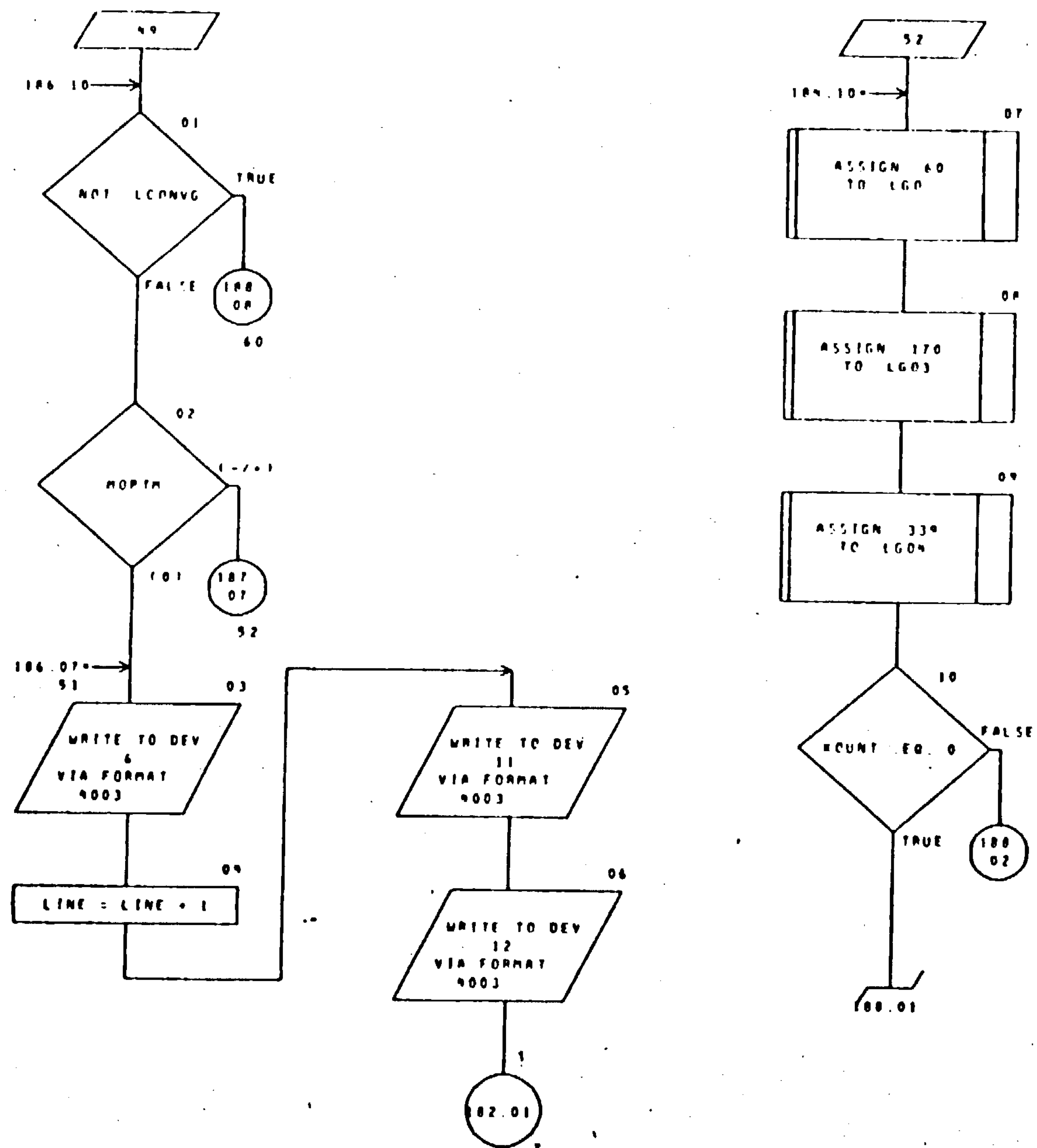
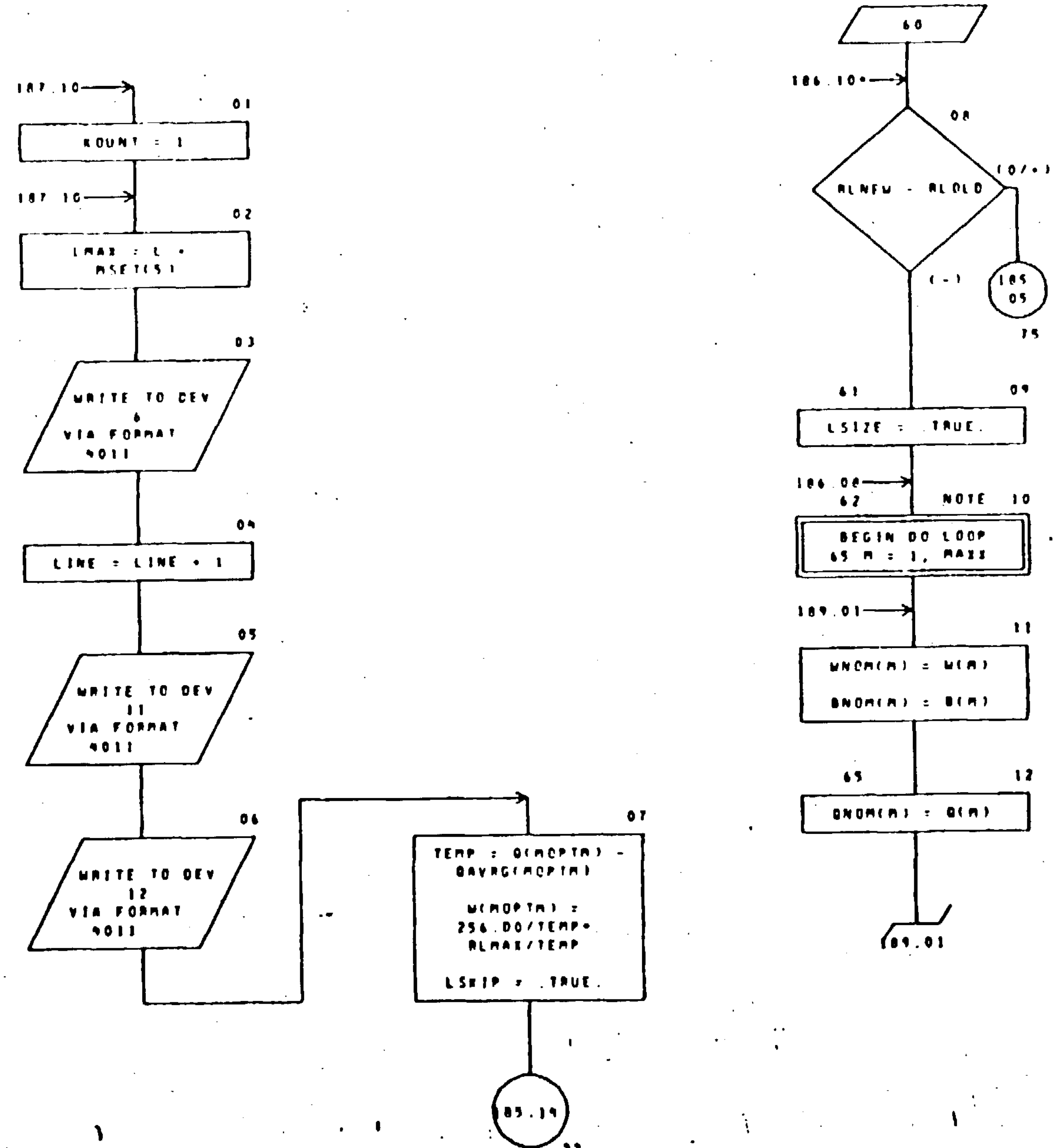


CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,XPART,MPERF,POS,CCHECK)



MINMX3-19

CHART TITLE - SUBROUTINE MINMX3(MSET,KERROR,KCONVG,KPART,MPERF,PDS,CCHECK)

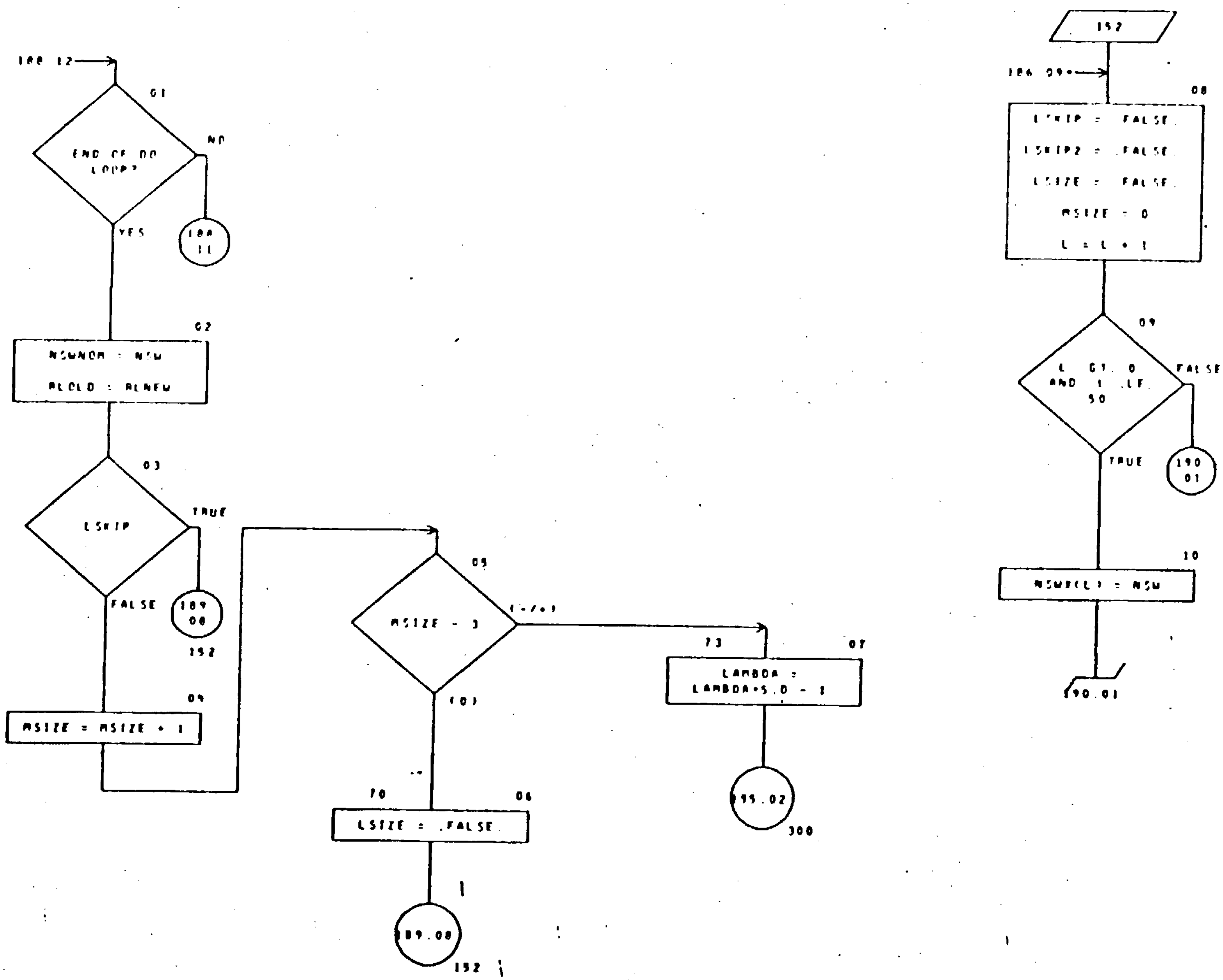
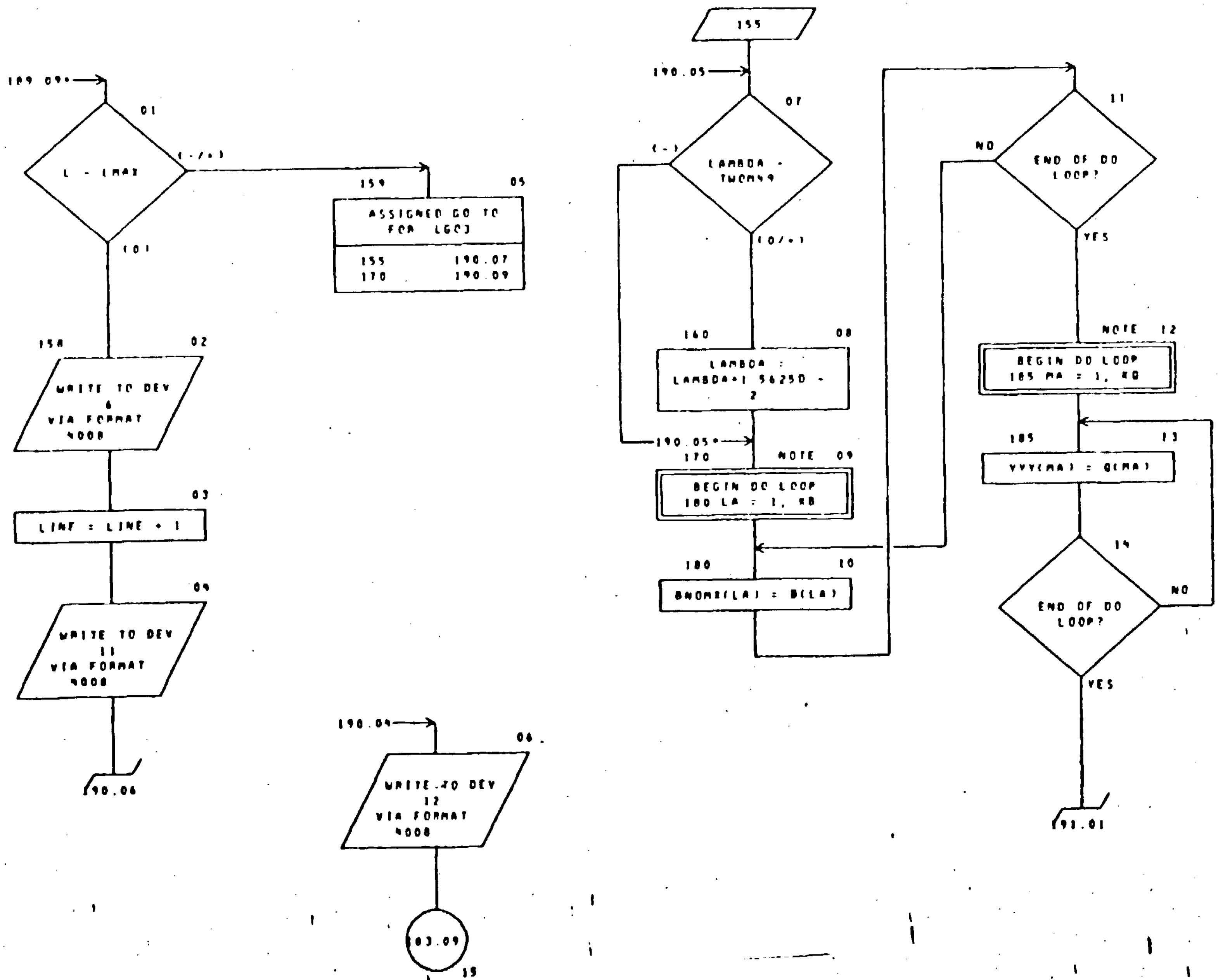


CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,NPART,MPERF,PDS,CCHECK)



MINMX3-21

CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,KPART,MPERT,PDS,CCHECK)

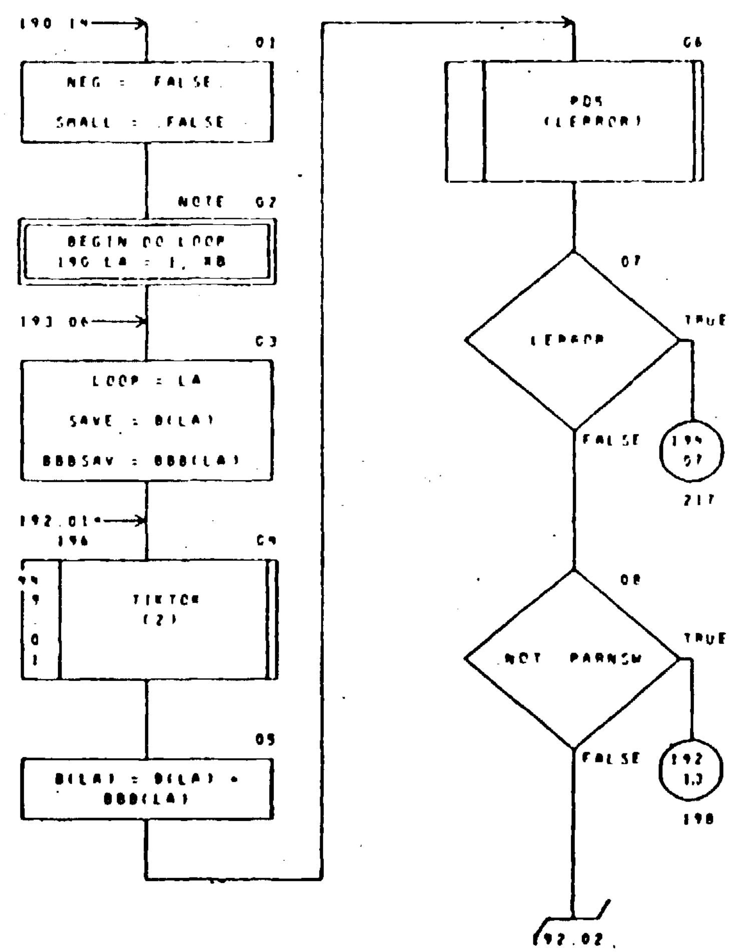
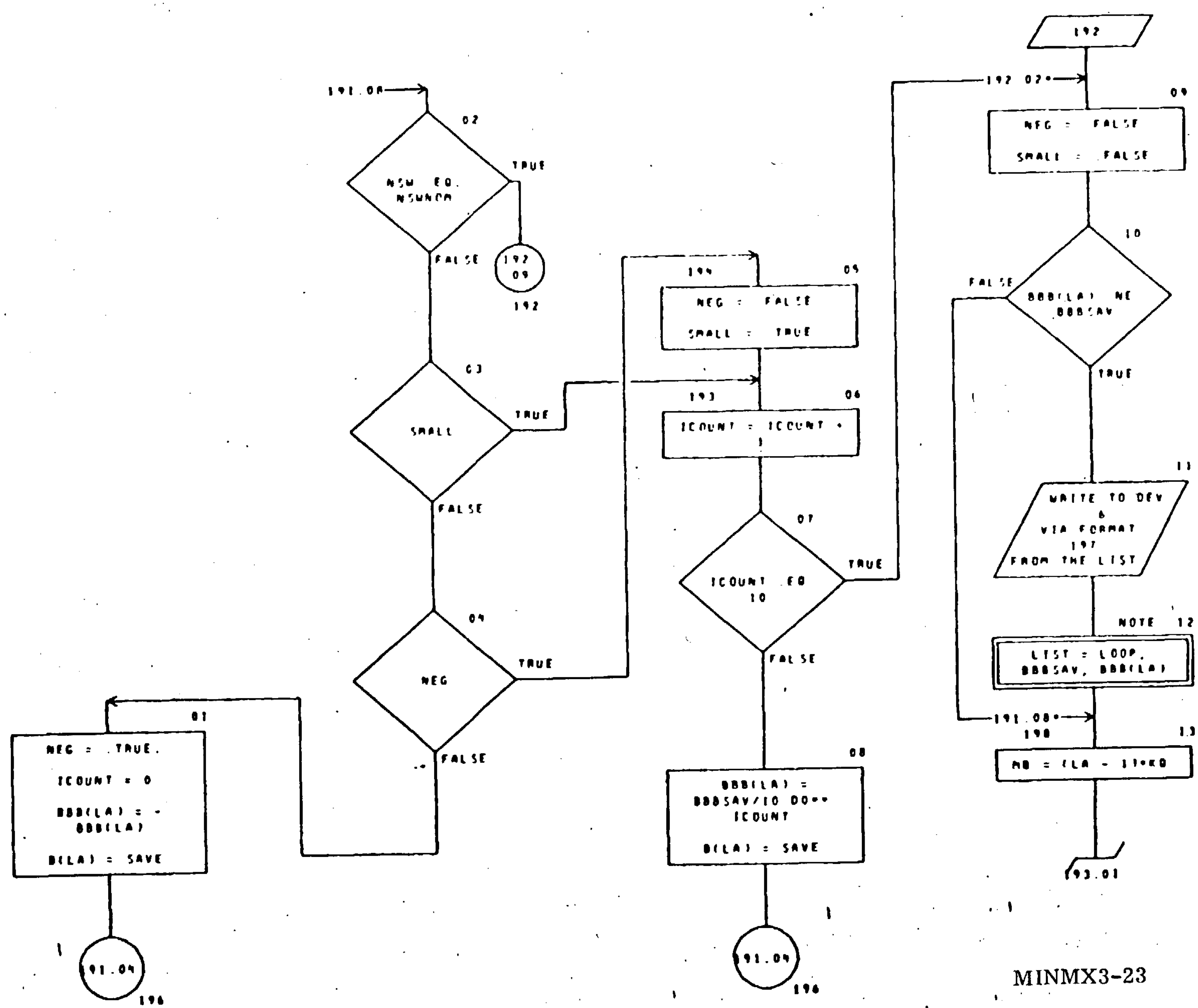


CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,ICONVG,KPART,MPERF,PDS,CCHECK)



MINMX3-23

CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,KPART,MPERF,POS,CCHECK)

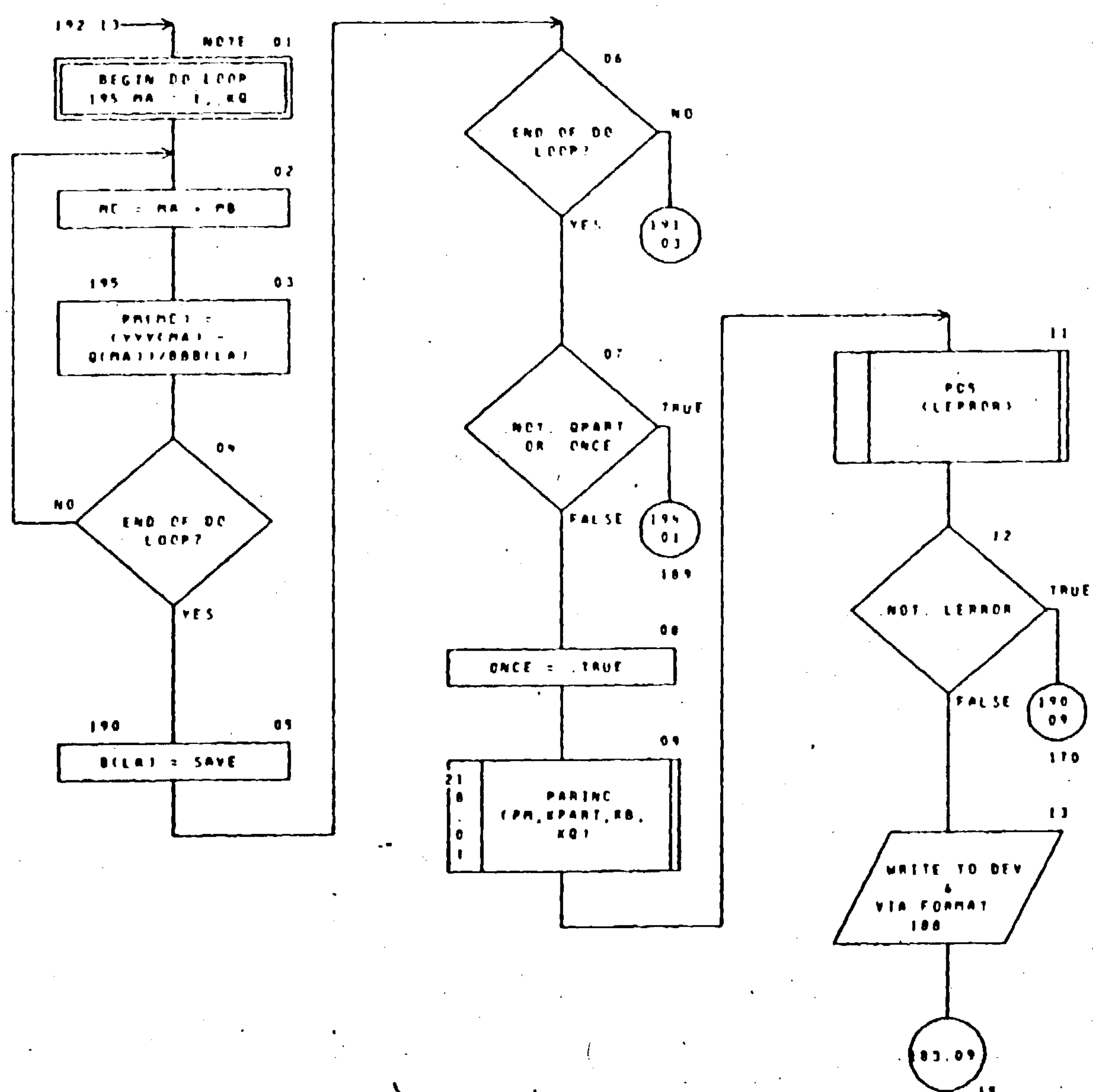
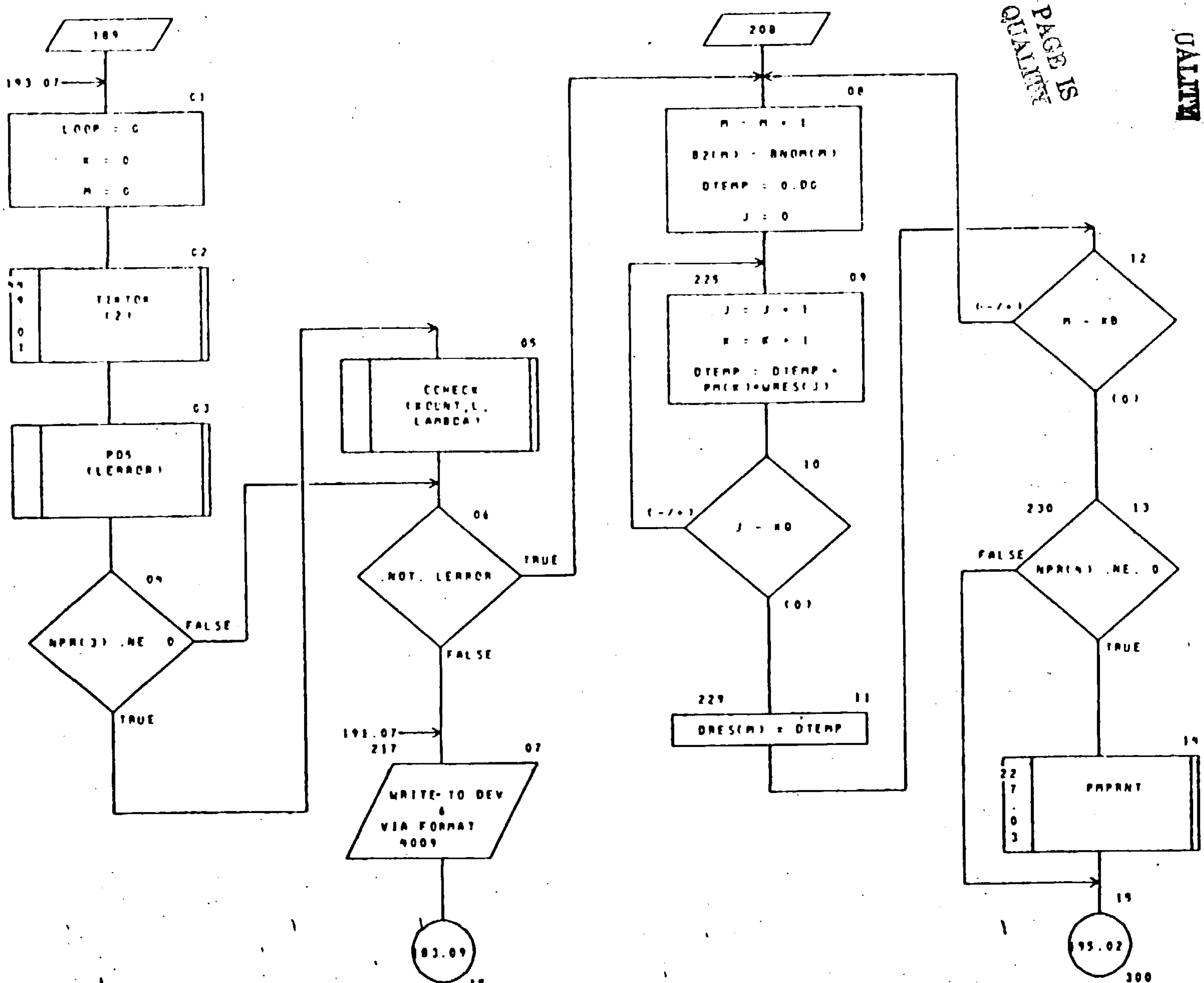


CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,NPART,MPERF,PDS,CCHECK)

PAGE IS
QUALITY



MINMX3-25

CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,KPART,MPERF,PDS,CCHECK)

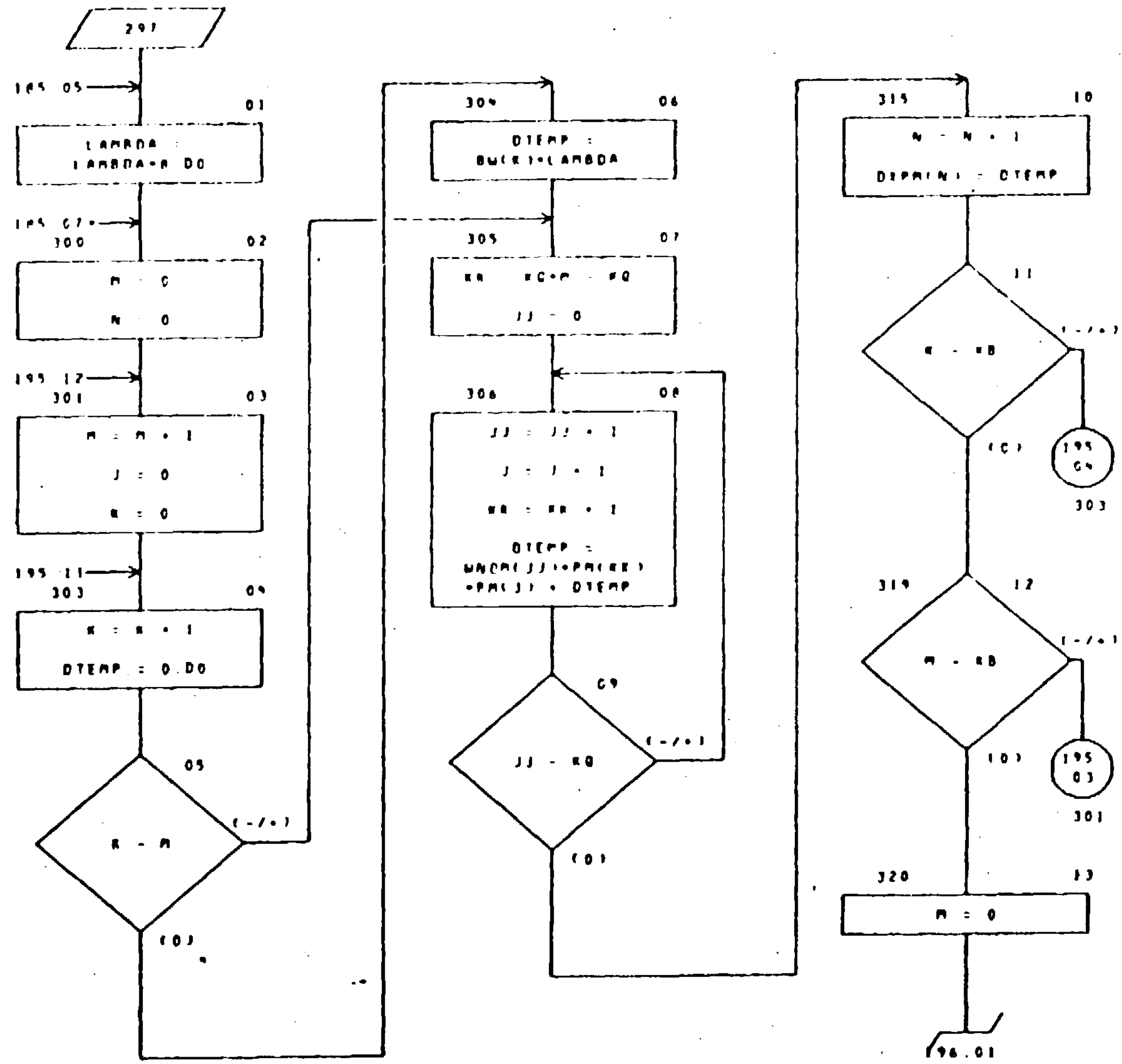


CHART TITLE - SUBROUTINE MINMX3(MSET,XERROR,XCONVG,KPART,MPEAT,POS,CCHECK)

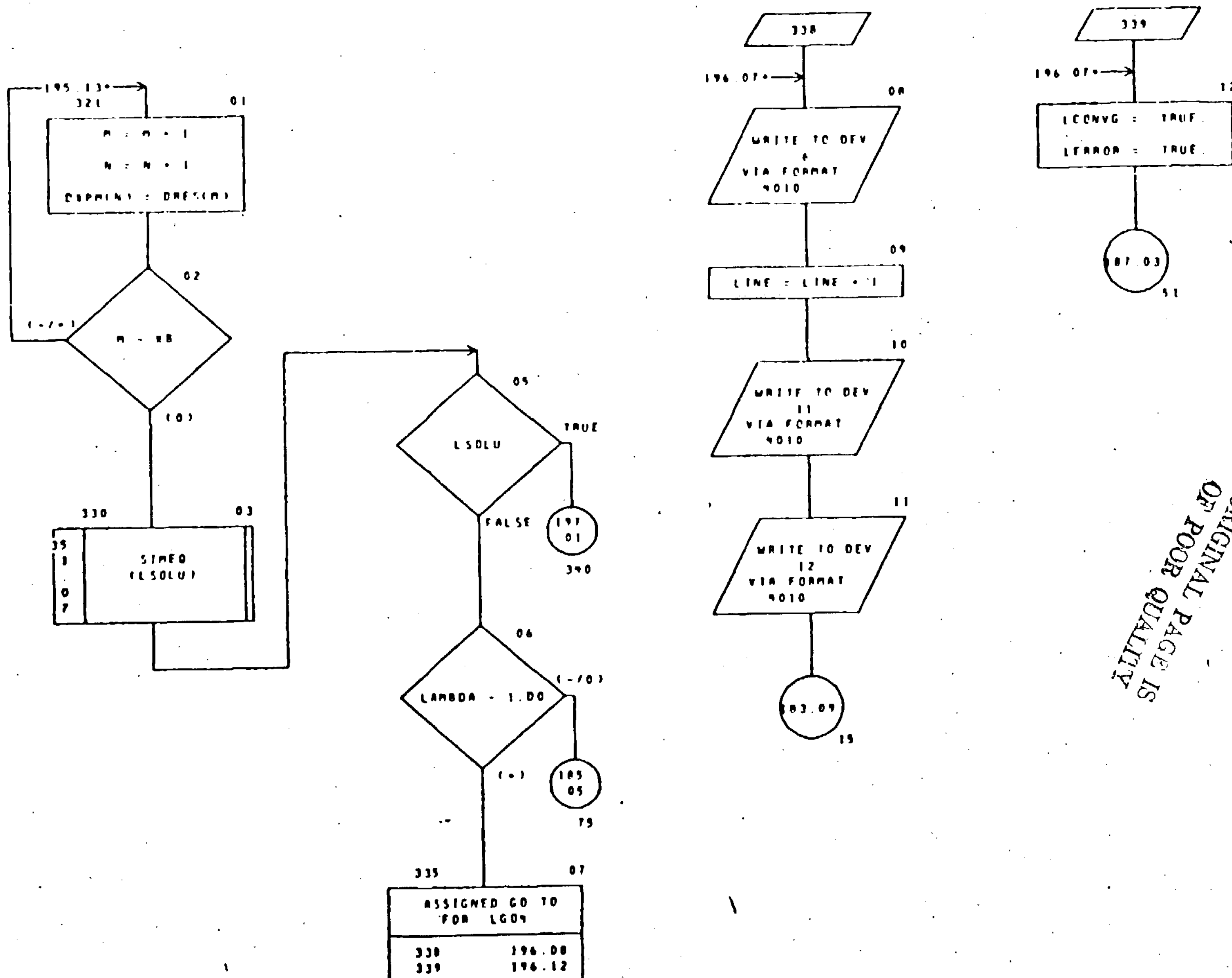


CHART TITLE - SUBROUTINE MINMX3(MSET,KERROR,KCONVG,KPART,MPERF,PDS,CCHECK)

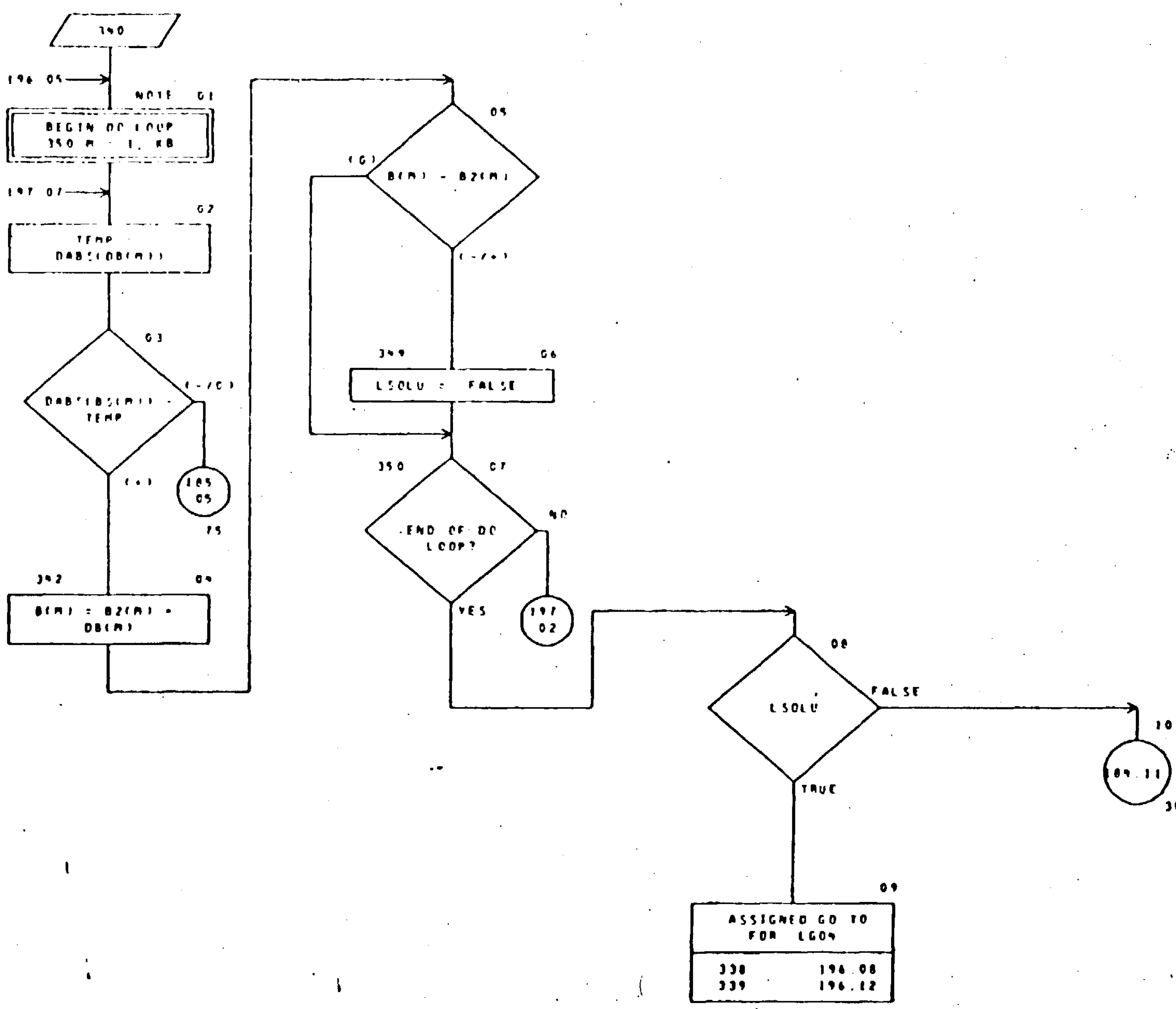


CHART TITLE - NON-PROCEDURAL STATEMENTS

ORIGINAL PAGE IS
OF POOR QUALITY

```
         IMPLICIT REAL*8 ( A-H, O-Z)
         REAL*8 LAMBDA
         LOGICAL LERROR, LCONVG, LSIZE,          LSOLU, LSKIP,NEG,SMALL
         LOGICAL LSKIP2,XCONVG,XERROR          ,WONDER,HUNG,OPART,ONCE,PARNSW
         DIMENSION MSET(5),BNOM(35),ONOM(35),QAVRG(35),WRES(35),DRES(35)
         ,W(35),B2(35),UNOM(35),DXPM(1260),DB(35)
         COMMON /REAL8/ PO1(311),XAMBDA,PO2(1688)
         COMMON /INTGR4/ IO1(4),LINE,IO2(27),NPR(4),NSUPAR,IO3(19),
         MAJOR,MAJORS,MINOR,IO4(5),NEW,IO5(69),ITPRNT,IOP(110),MM(70),
         IO7(25),NSWX(50),IO6(600)
         COMMON /LOGIC4/ LO1(4),WONDER,HUNG,PARNSW,LO2(493)
         COMMON /ITER2/  B(35),Q(35),BS(35),BW(35),QMIN(35),QMAX(35),
         BBB(35),YYY(35),BNOMX(35),PM(1225)
         COMMON /ITERAT/PO1(910),FXL1(70)
         EQUIVALENCE (MINOR,LOOP),(XAMBDA,LAMBDA),(MAJOR,L),(MAJORS,KOUNT)
         DATA TWOM28, TWOM49 / Z3A10000000000000, Z3480000000000000 /
         DATA SCALE / Z371000000000000 /
776      FORMAT(1H ,I7,45H VARIABLES INVALID IN MINMX3. RUN TERMINATED.)
4000     FORMAT (39H1FIRST GUESSES WILL NOT RUN TRAJECTORY. )
4001     FORMAT (33H0ITERATOR IS GIVING ERROR RETURN.  )
2002     FORMAT ( 1H0, I6, 45H TRAJECTORIES WITHOUT PARTIAL DERIVATIVES AND
          , I4, 62H TRAJECTORIES WITH PARTIAL DERIVATIVES REQUIRED FOR THIS
          CASE. )
8237     FORMAT(1H ,I4,25H TRAJECTORIES WITHOUT ANDI3,14H WITH PARTIALS)
34       FORMAT(1H ,20HMINMX3 SPECIAL PRINT,
```

```
         3X4IHL,KOUNT,LAMBDA,RLOLD,RLNEW,RLMAX,OK,SCALE/1H ,
         2I7,4X1P6D18.10)
 9003    FORMAT ( 24HOTHIS CASE IS CONVERGED   )
 9011    FORMAT (39HOITERATOR IS NOW IN IMPROVE MODE     )
 9008    FORMAT(39HOMAXIMUM NUMBER OF ITERATIONS EXCEEDED   )
 187     FORMAT(1H ,5HMINOR13,6XRHOLD DX =D16.8,6XRHNEW DX =D16.8)
 188     FORMAT(1HC,23HTRAJ ERROR AFTER PARINC)
 9009    FORMAT ( 41HOERROR IN PARTIAL DERIVATIVE CALCULATION     )
 9010    FORMAT( 29HOTHIS CASE WILL NOT CONVERGE    )
```

Name: MORE

Calling Argument: None

Referenced Sub-programs: EFMPRT, EXTAB, SWING, TAP, TAPSET

Referenced Commons: INTGR4, LOGIC4, REAL8, SOLSYS

Entry Points: None

Referencing Sub-programs: QPRINT

Discussion: This routine is utilized when more analysis and printout is desired in addition to that associated with subroutine QPRINT.

Subroutine MORE controls the ballistic swingby-continuation-analysis, the optional detailed-printout associated with the same, and also the optional ballistic trajectory-extension printout invoked via program input TGO.

Regarding the swingby-continuation, MORE generates the condensed planet-selection vector NOPT4(i) and associated trajectory-segment flight times TT2(i), and then controls the swingby-continuation-analysis wherein each cycle through a DO loop generates a single swingby and post-swingby trajectory segment via a call to subroutine SWING. This may encompass either multiple ballistic swingbys along one trajectory, or multiple swingbys past the primary-target (at the same time) in order to attempt to obtain different swingby solutions; the mode of swingby analysis is determined by program input.

Each ballistic trajectory-segment printout is controlled by calling a sequence of five subroutines, as follows:

TAPSET	(Initialize for ballistic trajectory segment)
TAP	(Generate segment)
EXTAB	(Print entire extremum table to end of segment)
EFMPRT	(Print post-swingby target ephemerides)
TAPSET	(Restore parameter values)

When generating multiple swingbys past the primary target and printing the resulting successfully-converged trajectory segments, multiple identical solutions are sorted-out and eliminated.

Messages and printouts: When ballistic swingby-continuation analysis is invoked, the heading is printed on unit 6, at the top of a new page following the Performance Summary page of each case:

SWINGBY CONTINUATION ANALYSIS

After the swingby-continuation analysis has been completed, for each case, the following summary is printed on unit 12:

SWINGBY SUMMARY   PLANET/SUCCESS, BURN/PASS RADIUS

n/x, y/r   n/x, y/r   n/x, y/r   ...

where n is the post-swingby-target number, x is a logical indicator (T or F) for whether or not the attempted solution was successfully obtained, y is a logical indicator (T or F) for whether or not the swingby maneuver was powered ($\Delta v > 0$), and r is the swingby passage radius, in planet radii. The set of parameters is printed up to (a maximum of) ten times, depending upon how many solutions were attempted.

Then, if printout of the post-swingby trajectory segments is invoked, each is preceded by the heading,

DETAILED PRINT OF POST-SWINGBY TRAJECTORY SEGMENT TO (name)
FOR SOLUTION HAVING (r) PASSAGE DISTANCE

in which "name" is the name of the post-swingby target, and "r" is the swingby passage distance, in planet radii.

When printout of the ballistic trajectory-extension is invoked, it is preceded by the heading,

DETAILED PRINT OF POST-TARGET TRAJECTORY FOLLOWS.

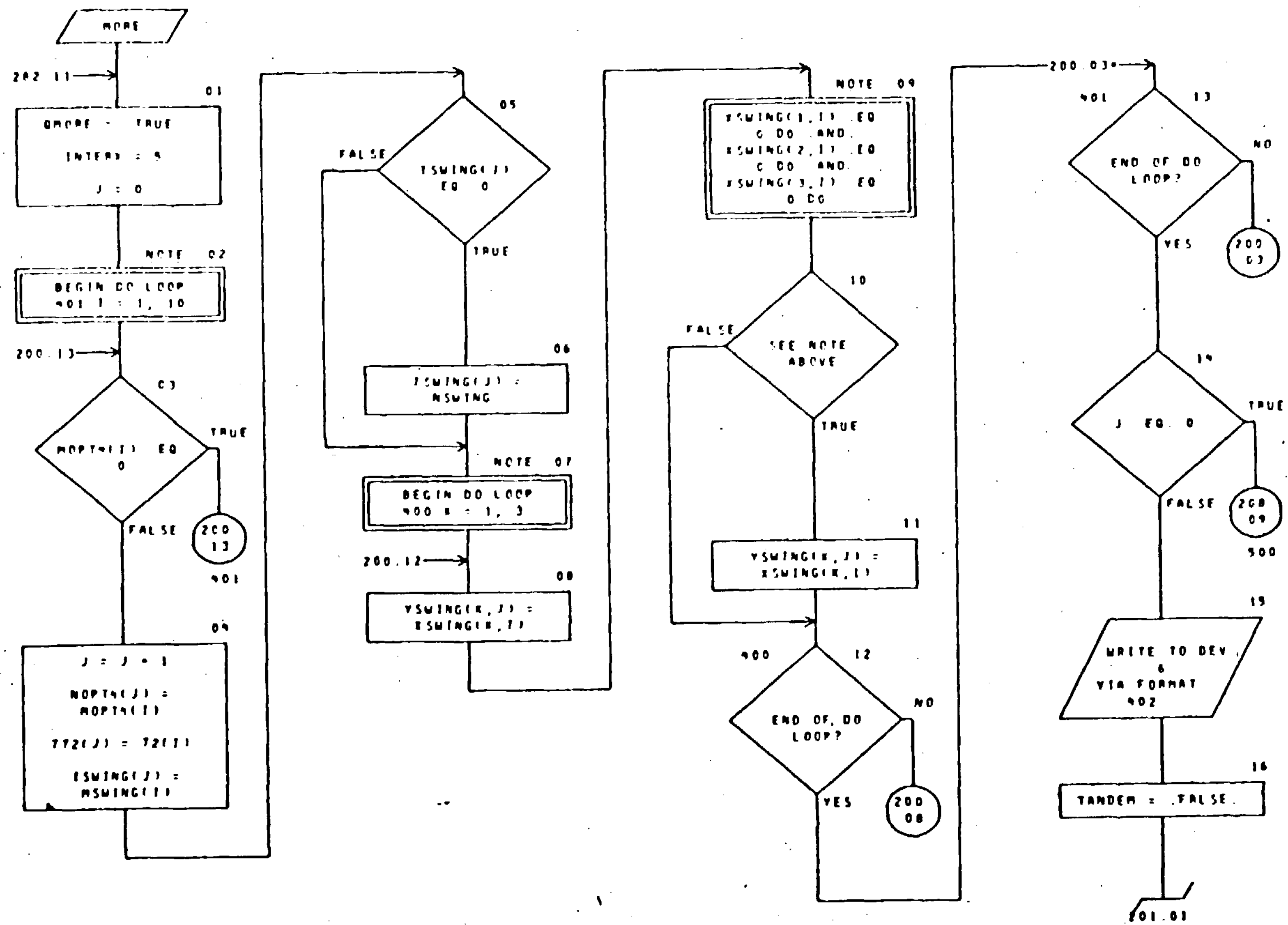
MORE EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
X(50)	SUA	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
T2(10)	U	REAL8	Initial estimates of swingby-continuation trajectory-segment flight times, in days.
APL (2,70)	U	SOLSYS	Array of target names.
TGO	UA	REAL8	Ballistic trajectory-extension print option indicator.
TSUM	SU	REAL8	Time elapsed since primary-target swingby and current swingby, in days.
EXTRA	SU	LOGIC4	Indicator for extra printout for each trajectory block print (computation step).
MOPT3	UA	INTGR4	Primary-target selector (planet number).
MOPT4(10)	U	INTGR4	Planet-numbers of post-swingby targets.
QMORE	S	LOGIC4	Indicator which tells remainder of program that subroutine MORE is in control.
INTERX	S	INTGR4	Index used in subroutine EFM which indicates which array locations are applicable in selecting orbital elements.
MSWING(10)	U	INTGR4	Swingby maneuver type selectors.
NSWING	U	INTGR4	Uniform swingby-maneuver type selector.
TANDEM	SU	LOGIC4	Indicator for multiple swingbys along a single trajectory.
TDATE2	U	REAL8	Julian date at the primary target, in days (less 2400000).

MORE EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
XSWING (3,10)	U	REAL8	Array of velocity vectors consisting of initial velocity guesses of a given post-swingby trajectory segment, input to the program, in EMOS.
YSWING (3,10)	S	REAL8	Same as XSWING, with blank entries deleted and zero entries set to first velocity vector, in EMOS (used internally).

CHART TITLE - SUBROUTINE MORE



MORE-5

CHART TITLE - SUBROUTINE MORE

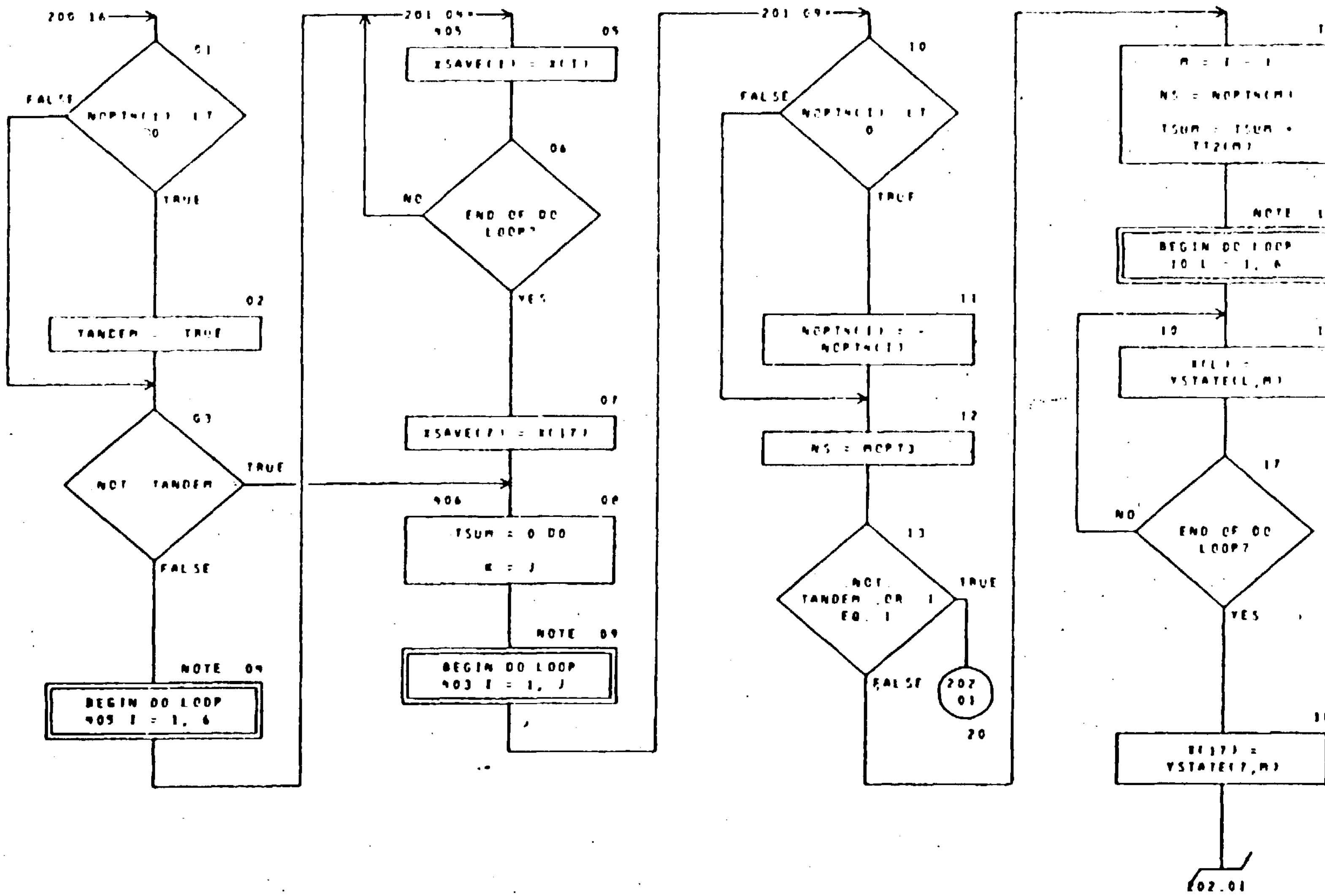


CHART TITLE - SUBROUTINE MORE

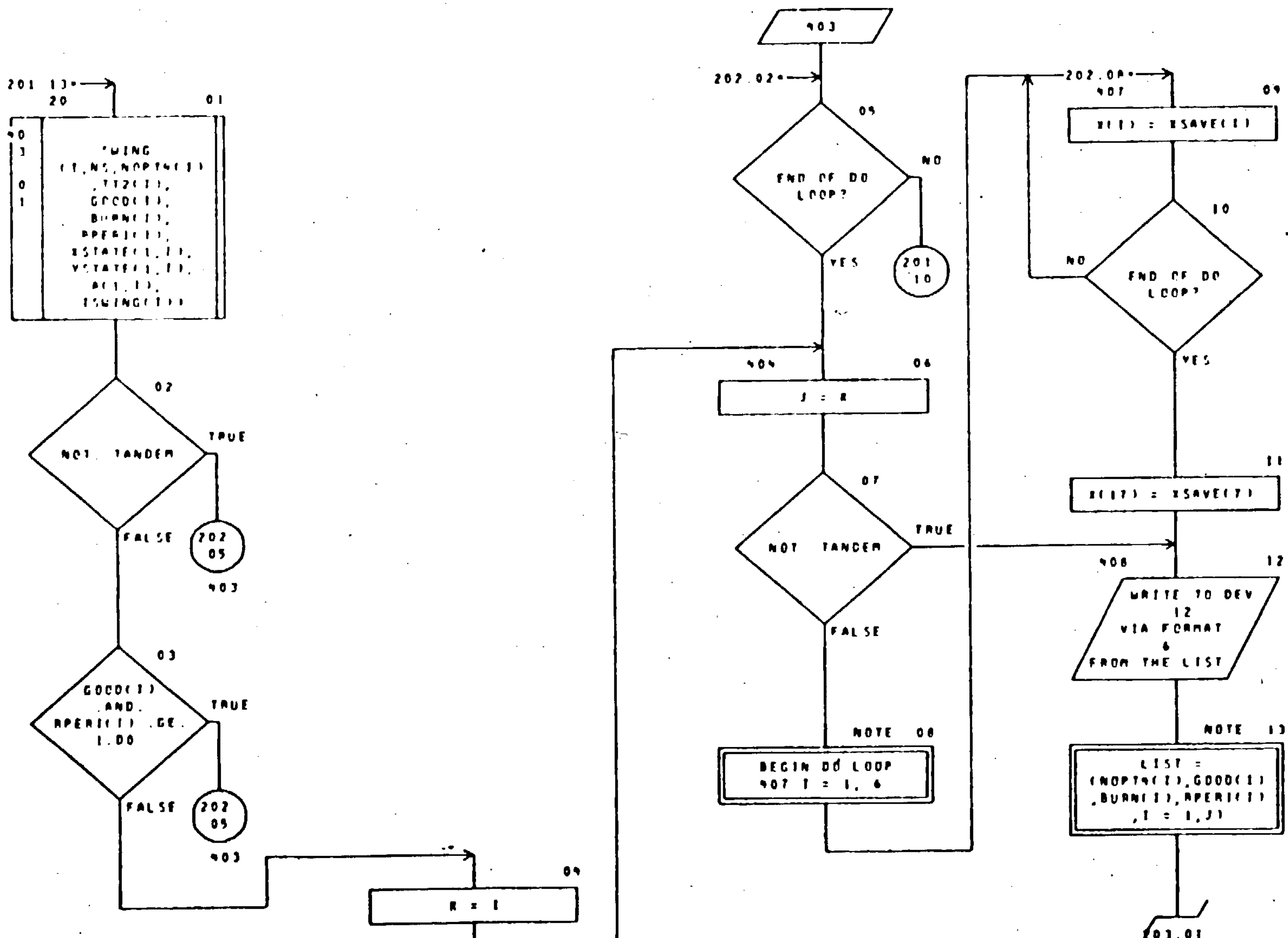


CHART TITLE - SUBROUTINE MORE

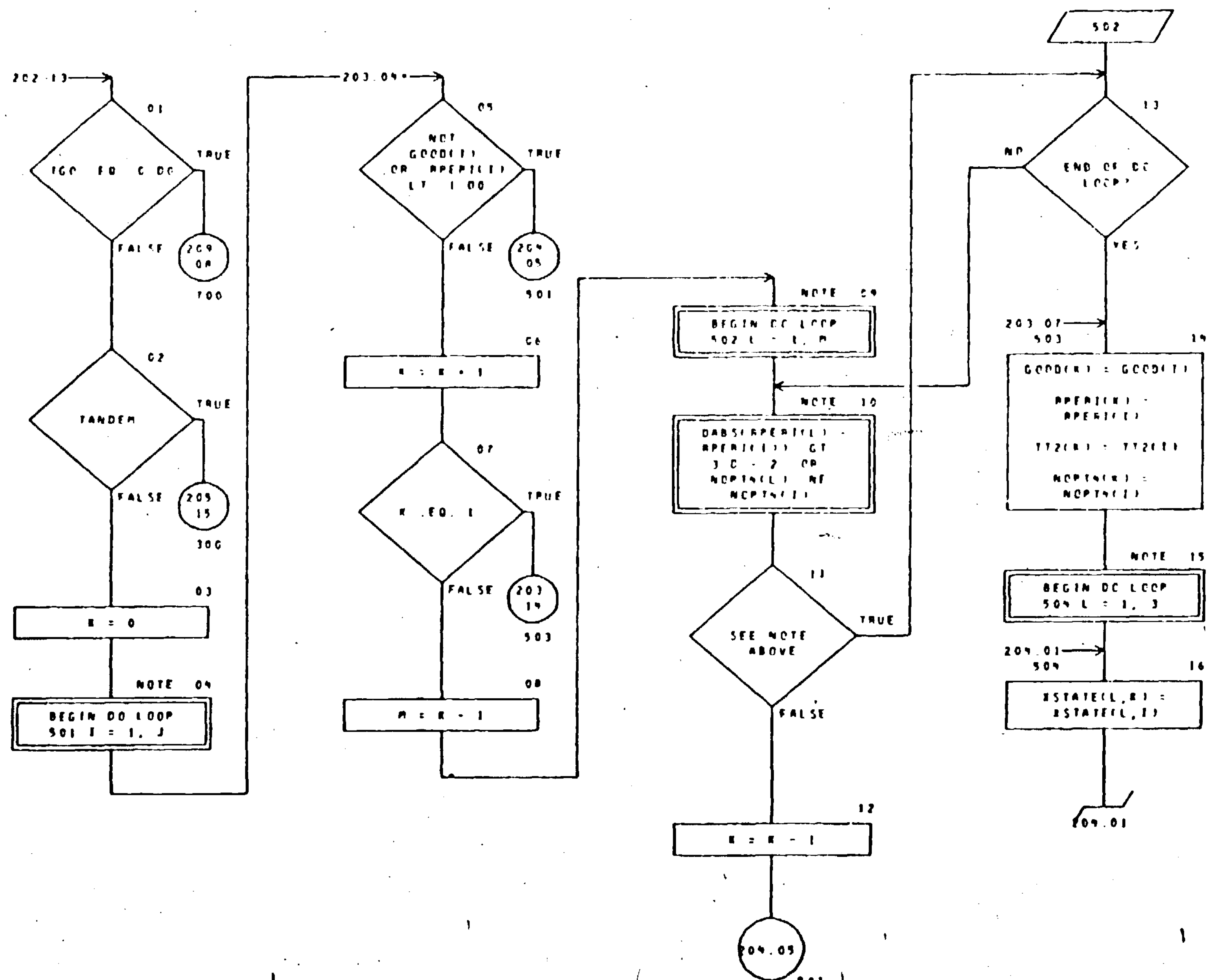


CHART TITLE - SUBROUTINE MORE

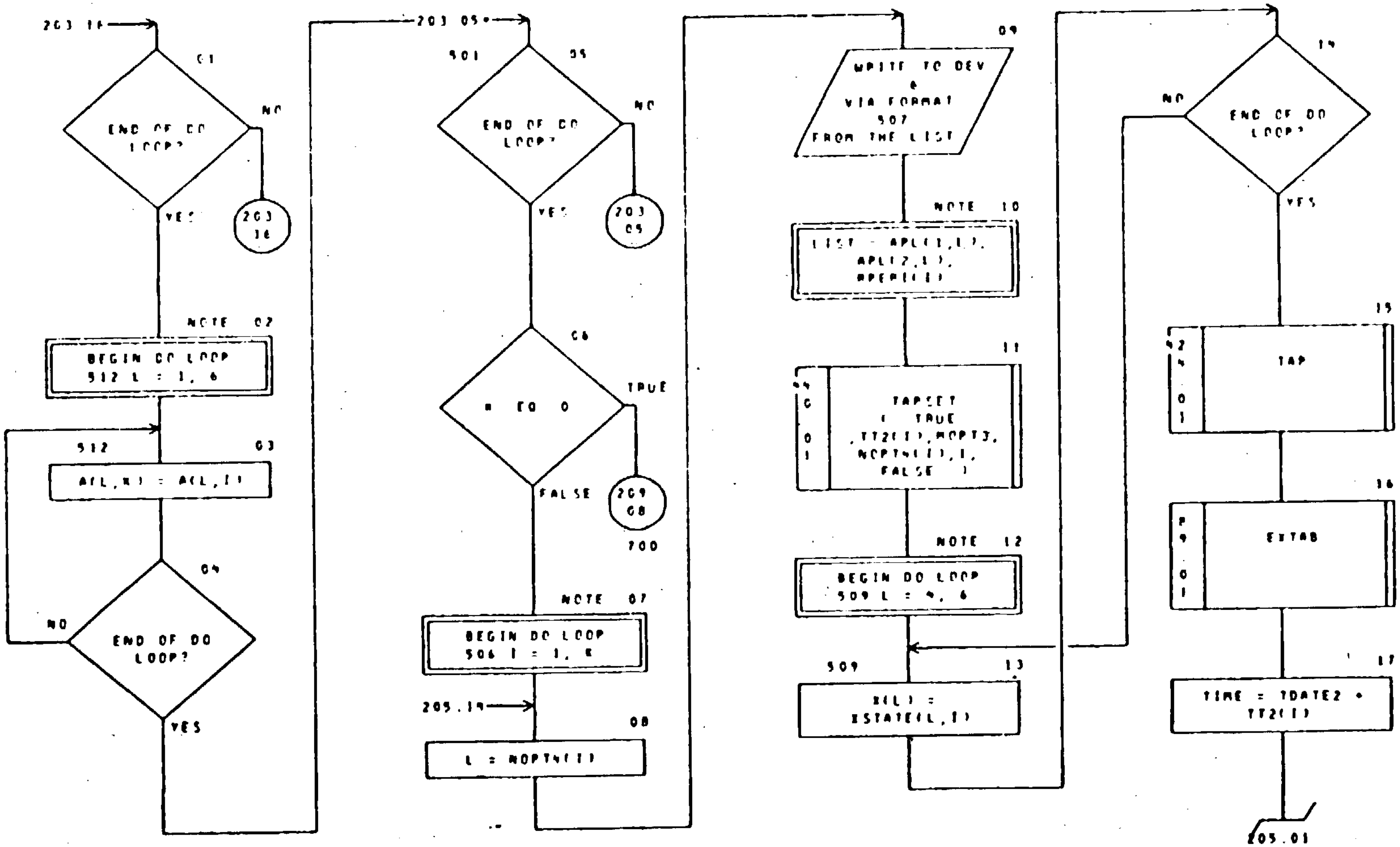
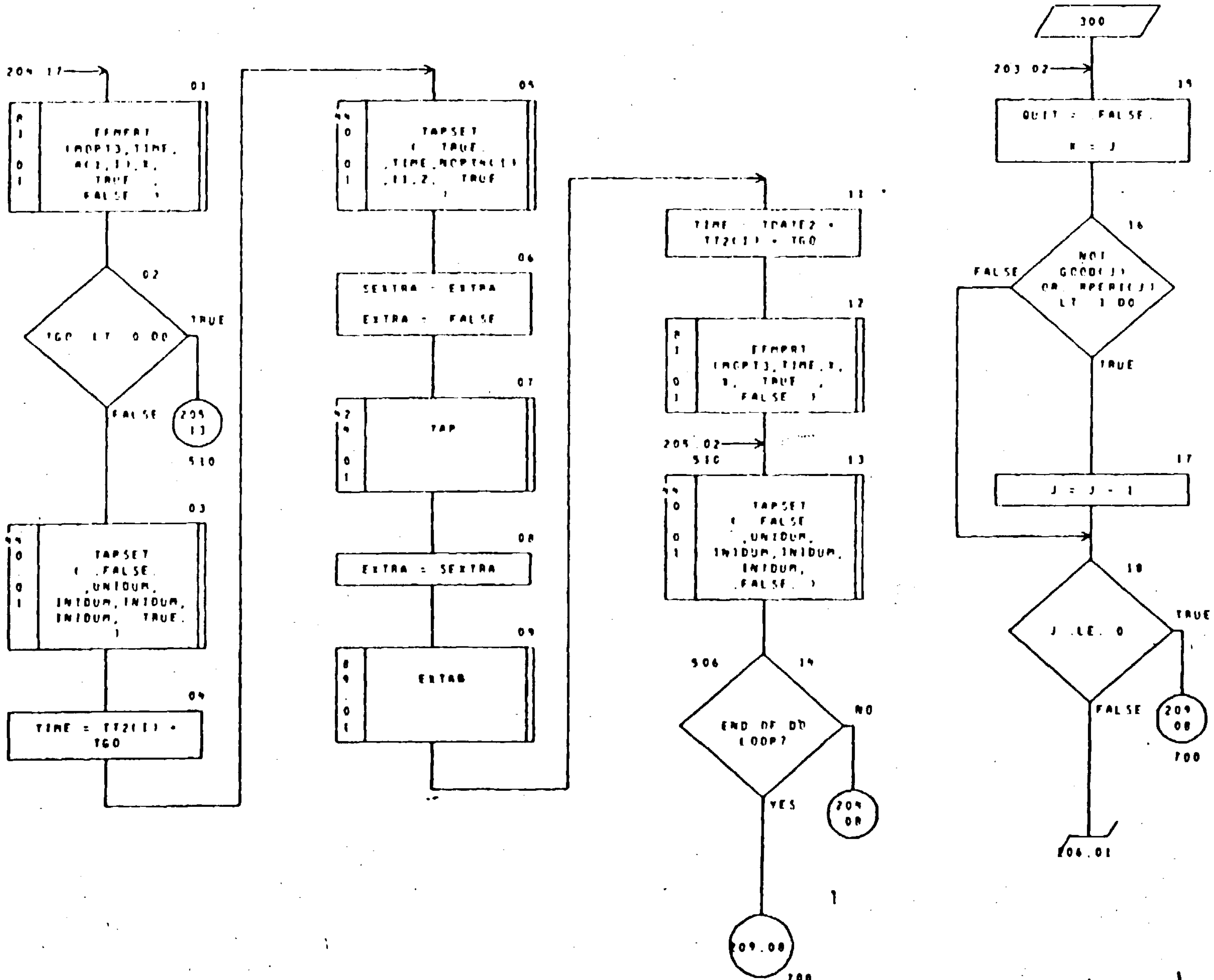
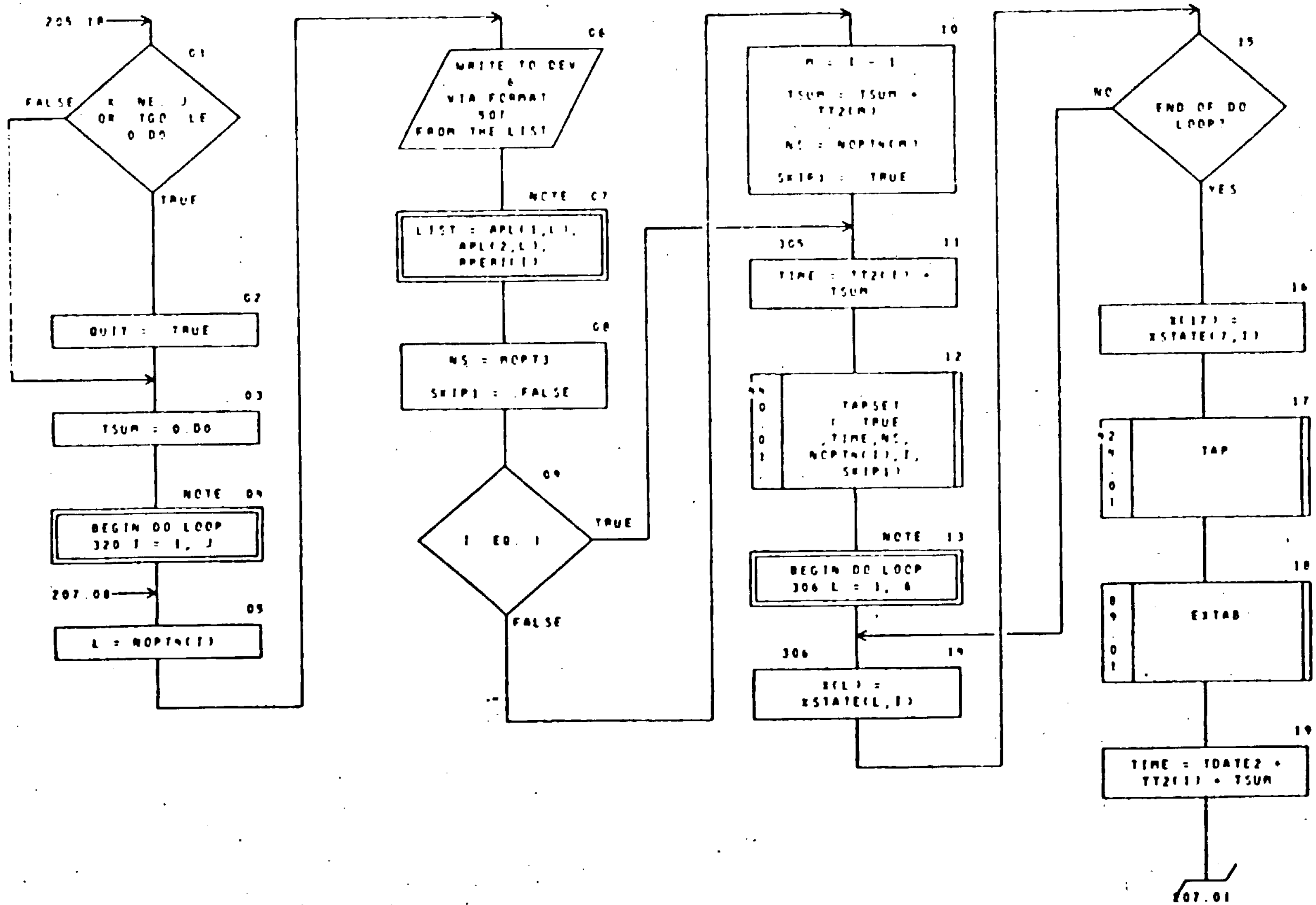


CHART TITLE - SUBROUTINE MORE



ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - SUBROUTINE MORE



MORE-11

CHART TITLE - SUBROUTINE MORE

ORIGINAL PAGE IS
OF POOR QUALITY

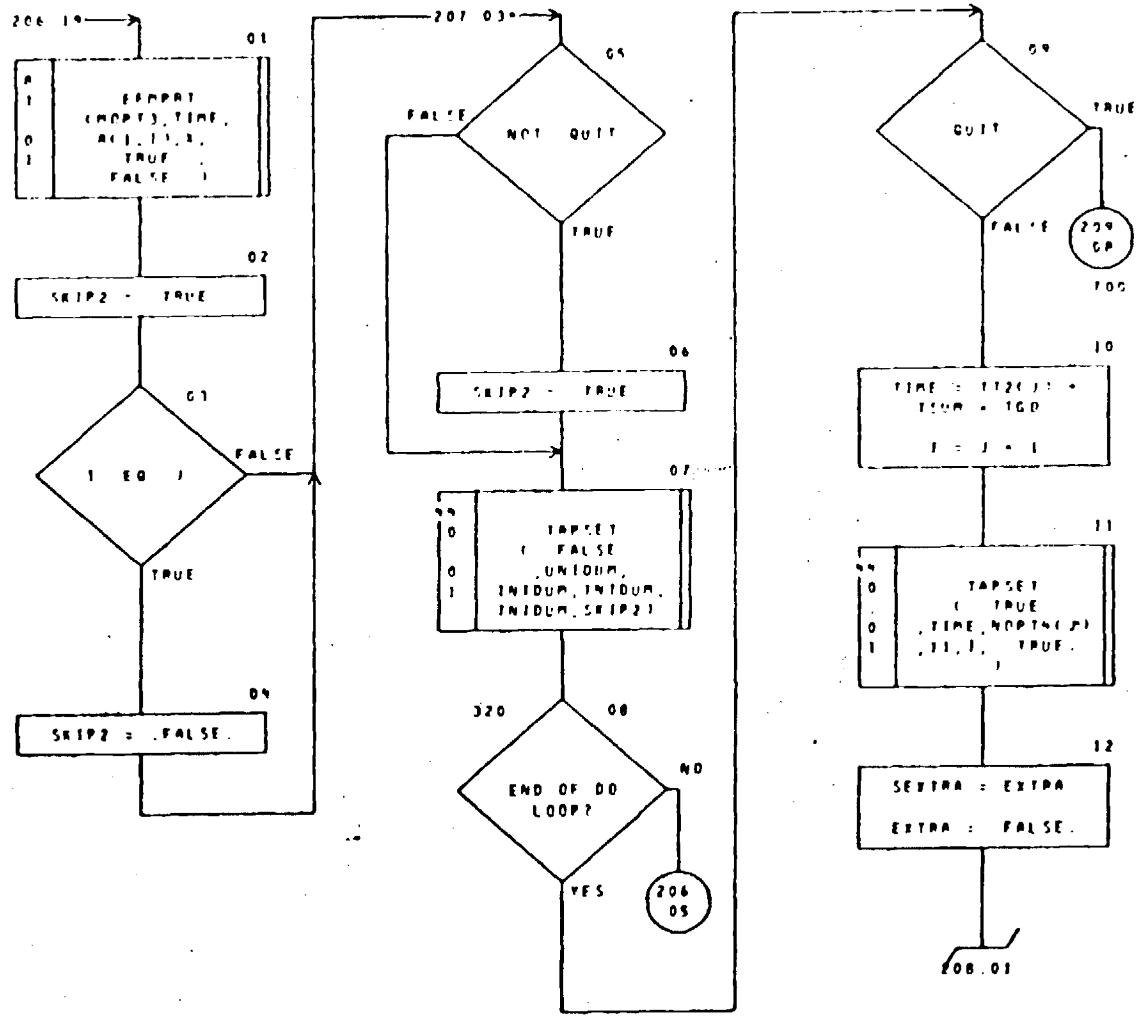


CHART TITLE - SUBROUTINE MORE

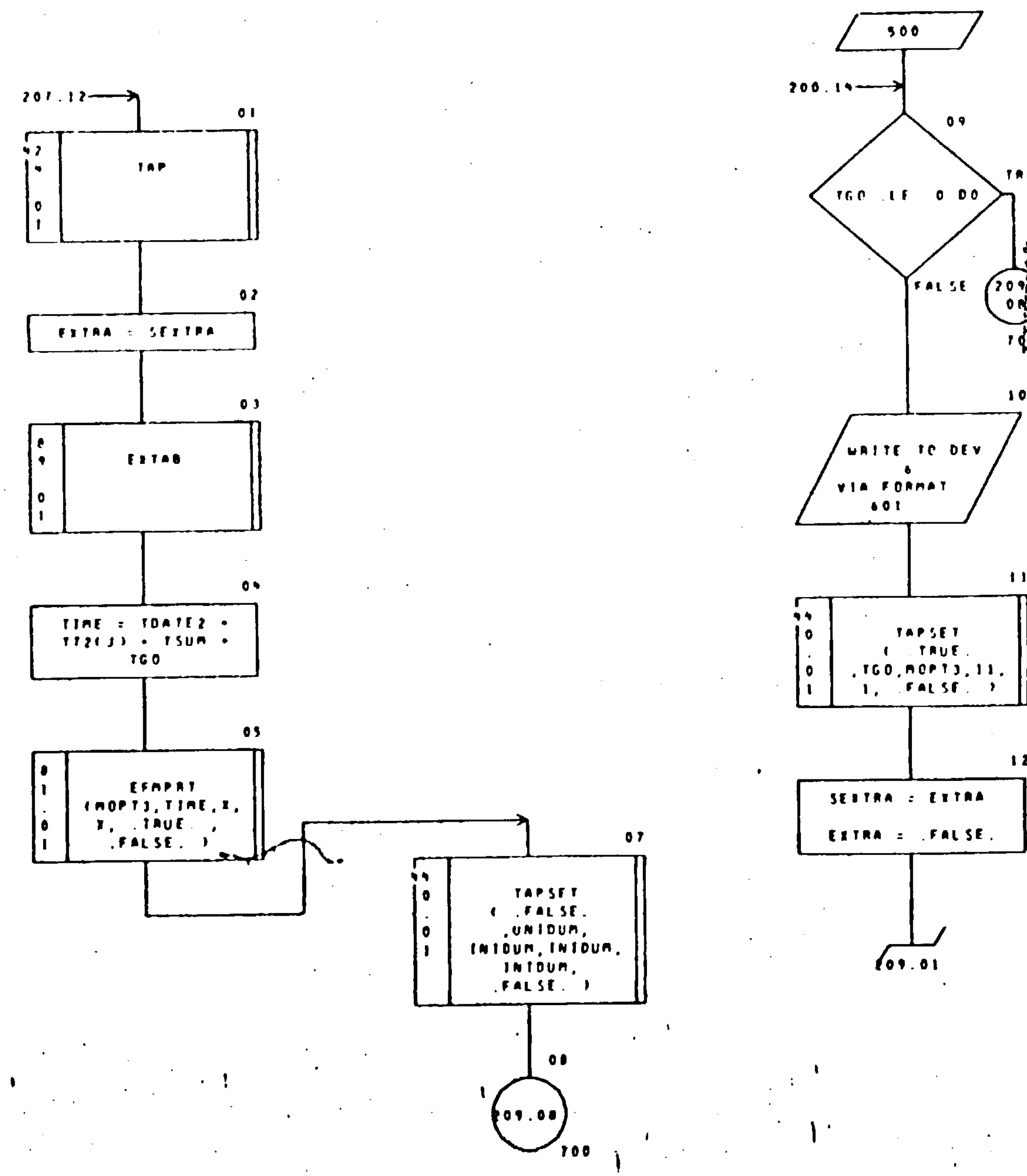


CHART TITLE - SUBROUTINE MORE

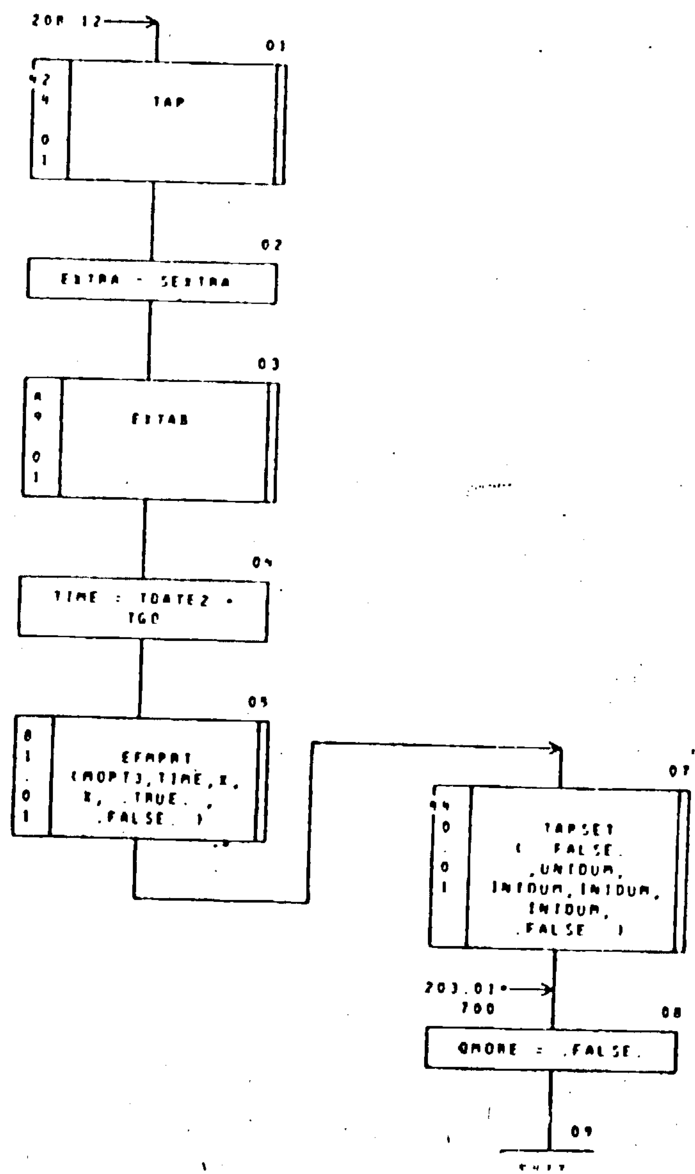


CHART TITLE - NON-PROCEDURAL STATEMENTS


```
          IMPLICIT REAL*8 (A-H,O-Z)
          LOGICAL GOOD,BURN,OMORE,EXTRA,SEXTRA,TANDEM,SKIP1,SKIP2,QUIT
          DIMENSION TT2(10),NOPTN(10),RPERI(10),GOOD(10),BURN(10),A(6,10),
          XSTATE(7,10),YSTATE(7,10),XSAVE(7),ISWING(10)
          COMMON /REAL8/ RC1(89),T2(10),RC2(57),TDATE2,RC3(19),TGC,
          RC4(171),XSWING(3,10),YSWING(3,10),RC5(394),TSUM,RC7(445),X(55),
          RC6(700)
          COMMON /INTGR4/ IO1(24),MOPT3,MOPTN(10),IO2(93),MOPTX(5),IO3(7),
          NSWING,IO4(4),INTERX,IO5(2),MSWING(10),IO6(492)
          COMMON /LOGIC4/ LO1(45),EXTRA,OMORE,TANDEM,LO2(452)
          COMMON /SOLSYS/ SOLO1(140),APL(2,70)
402       FORMAT(1H1,49X29HSWINGBY CONTINUATION ANALYSIS//1H )
4         FORMAT(1H ,50HSWINGBY SUMMARY    PLANET/SUCCESS,BURN/PASS RADIUS
          /1H ,5(I2,1H/L1,1H,L1,1H/F6.2,1H/)/1H ,5(I2,1H/L1,1H,L1,1H/F6.2,1H
          /))
507       FORMAT(1H1,49HDETAILED PRINT OF POST-SWINGBY TRAJECTORY SEGMENT,
          4H TO 2A8///1H ,10X19HFOR SOLUTION HAVINGF7.2,
          17H PASSAGE DISTANCE/1H )
601       FORMAT(1H1,49HDETAILED PRINT OF POST-TARGET TRAJECTORY FOLLOWS.)
```

Name: OMASS

Calling Argument: V for OMASS;
MBOOST, B1, B2, B3, K for BOOSTR

Referenced Sub-programs: DECLIN, GET I, GUNTHR for OMASS;
None for BOOSTR

Referenced Commons: GUNCOM, ITERAT, LOGIC4, REAL8

Entry Points: BOOSTR

Referencing Sub-programs: TRAJ for OMASS;
QPRINT, QSTART for BOOSTR

Discussion: OMASS computes the launch vehicle mass (payload) injected into heliocentric space as a function of the launch characteristic speed v_c according to the approximate formula,

$$m_o = b_1 e^{-v_c/b_2} - b_3 ,$$

where b_1, b_2, and b_3 are coefficients which characterize the launch vehicle's performance capability.

Two cases exist in the computation of v_c, depending on the relation of the departure asymptote declination δ (obtained from subroutine DECLIN) to the parking orbit inclination i:

(1) $-i \le \delta \le i$

In this case, the launch maneuver is entirely coplanar, and the characteristic speed is given by,

$$v_c = \sqrt{v_{\infty o}^2 + v_e^2} ,$$

where v_e^2 is a constant; $v_e = 2v_o$, where v_o is the circular orbit speed at an altitude of 185 km above the Earth. $v_{\infty o}$ is the launch hyperbolic excess speed.

PRECEDING PAGE BLANK NOT FILMED

(2) $|\delta| > i$

In this case,

$$v_c = \sqrt{v_{\infty o}^2 + v_e^2} + \Delta v_i + \Delta v_g ,$$

where

$$\Delta v_i = c_1 i^2 + c_2 i + c_3 ,$$

and

$$\Delta v_g = v_g - \left(\sqrt{v_{\infty o}^2 + v_e^2} - v_o\right) .$$

In the above, Δv_i is the speed penalty corresponding to a non-due-east launch, and the values of the coefficients c_1, c_2, and c_3 are chosen to fit the penalty curve given in Reference 1:

$$c_1 = .05662285136 \text{ m/sec/deg}^2$$

$$c_2 = .9039433771 \text{ m/sec/deg}$$

$$c_3 = -71.75429726451 \text{ m/sec}$$

Δv_g is the speed penalty associated with a non-coplanar burn out of the circular parking orbit, and the value of v_g is produced by subroutine GUNTHR; in the discussion of subroutine GUNTHR, v_g is denoted Δv.

If the MINMX3 iterator has been assigned the task of optimizing the parking orbit inclination i, then the transversality condition associated with i is computed,

$$T(i) = 2c_1 i + c_2 - \partial v_g / \partial i_\infty ,$$

where i_∞ is the out-of-plane angle which the departure asymptote makes with the parking orbit plane. The value of $T(i)$ is used in subroutine GETQ. When MINMX3 does not optimize i, subroutine GET I performs the task.

In either case above, the derivative of initial mass with respect to characteristic speed is computed,

$$\frac{dm_o}{dv_c} = -\frac{b_1}{b_2} e^{-v_c/b_2} .$$

Then the partial derivative of initial mass with respect to launch hyperbolic excess speed is given by

$$\frac{\partial m_o}{\partial v_{\infty o}} = \frac{dm_o}{dv_c} \frac{\partial v_c}{\partial v_{\infty o}} ,$$

where, in case (1) above,

$$\frac{\partial v_c}{\partial v_{\infty o}} = \frac{dv_c}{dv_{\infty o}} = \frac{v_{\infty o}}{v_c} ,$$

and in case (2),

$$\frac{\partial v_c}{\partial v_{\infty o}} = \frac{\partial v_g}{\partial v_{\infty o}} ,$$

which is computed in subroutine GUNTHR.

The launch vehicle coefficients b_1, b_2, and b_3 are generated by a small, auxiliary computer program which is designed to produce curve-fit coefficients yielding a least-squares-error fit; the curve-data input to the auxiliary program are obtained from Reference 1.

Entry point BOOSTR performs the initialization of the required coefficients at the beginning of each case, and the printout of a single line on the Performance Summary page giving the launch vehicle name and coefficients. The circular orbit speed at an altitude of 185 km is computed,

$$v_o = \sqrt{v_e^2/2} .$$

Messages and printouts: A single line of information is printed on the Performance Summary page of each case, as follows:

LAUNCH VEHICLE IS (name) (COEFFICIENTS = (b_1) (b_2) (b_3))

where "name" is the launch vehicle name if the vehicle is chosen from the list of those available internally in the program, and is "INPUT" if the launch vehicle coefficients are input to the program. b_1, b_2, and b_3 are the coefficients which characterize the performance capability of the launch vehicle, where b_1 and b_3 are in kilograms and b_2 is in meters/second.

Reference:

1. "Launch Vehicles Estimating Factors," NASA Office of Space Science, 1973 Edition.

OMASS EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
K	UX		Indicator for initialization or print.
O(70)	U	ITERAT	Array of iterator independent-variables; O(17) is parking orbit inclination, i, in radians.
V	UAX		Launch hyperbolic excess speed, $v_{\infty o}$, in EMOS.
AM	SU	REAL8	Initial spacecraft mass, m_o, in kilograms.
B1	UX		Launch vehicle coefficient (input to program), b_1, in kilograms.
B2	UX		Launch vehicle coefficient (input to program), b_2, in meters/second.
B3	UX		Launch vehicle coefficient (input to program), b_3, in kilograms.
DV	U	GUNCOM	Minimum incremental speed to depart the parking orbit, v_g, in EMOS.

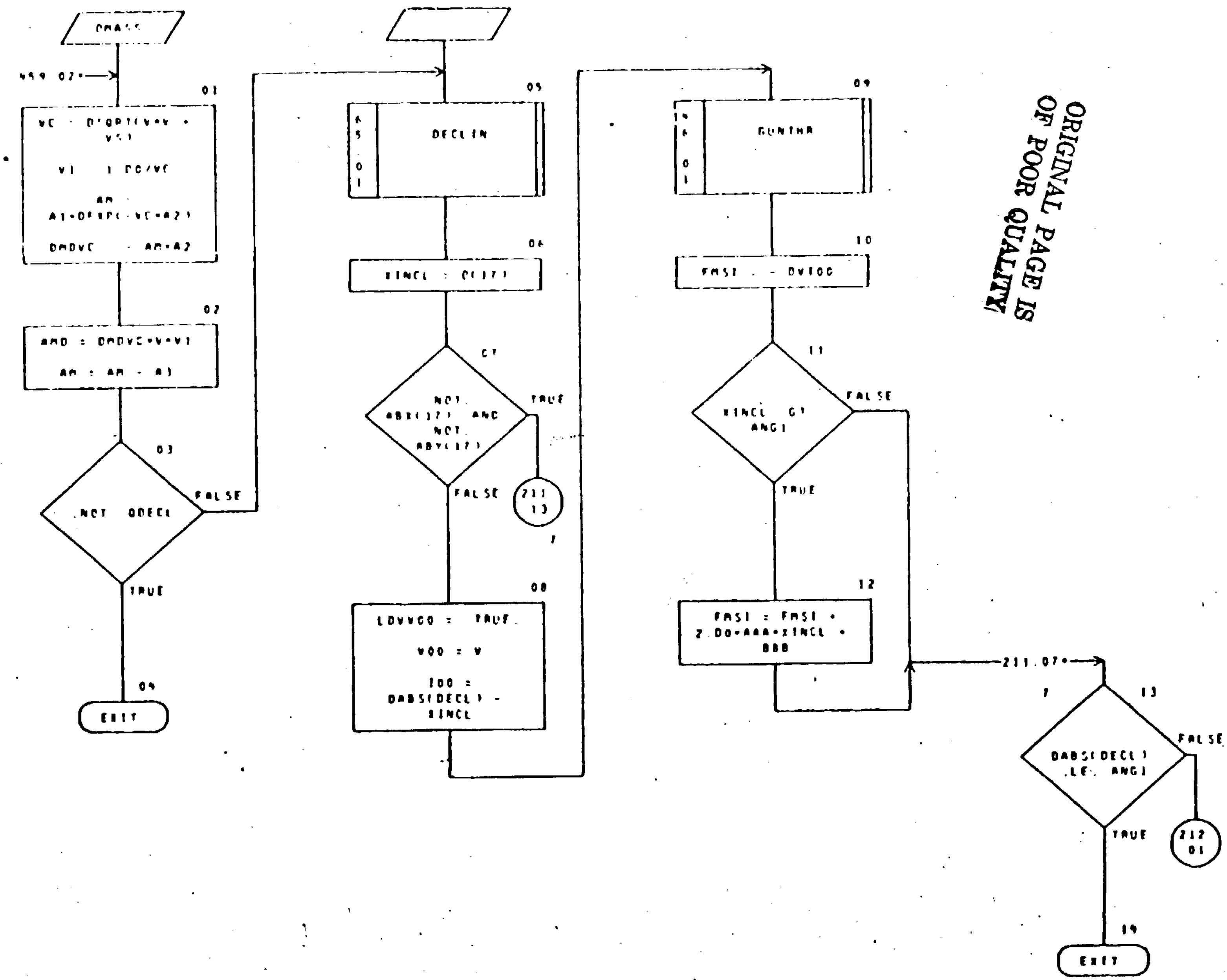
OMASS EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
V0	SU	GUNCOM	Circular orbit speed, v_o, in EMOS.
AAA	SU	REAL8	Non-due-east speed penalty coefficient, c_1, in EMOS/radian^2.
ABX(70)	U	LOGIC4	Array of iterator independent-variable indicators.
ABY(70)	U	LOGIC4	Array of iterator dependent-variable indicators.
AMD	S	REAL8	$\partial m_o/\partial v_{\infty o}$, in kg/EMOS.
BBB	SU	REAL8	Non-due-east speed penalty coefficient, c_2, in EMOS/radian.
CCC	SU	REAL8	Non-due-east speed penalty coefficient, c_3, in EMOS.
DEG	U	REAL8	Radians-to-degrees conversion factor.
I00	S	GUNCOM	Out-of-plane angle, $i_\infty = \|\delta\| - i$, in radians.
V00	S	GUNCOM	Departure hyperbolic excess speed, $v_{\infty o}$, in EMOS.
ANG1	SU	REAL8	Launch site latitude, in radians.
ANG2	SU	REAL8	Maximum permissible parking-orbit inclination, due to range safety considerations, in radians.
DECL	UA	REAL8	Departure asymptote declination, δ, in radians.
FMSI	SU	REAL8	$T(i) = \partial v_c/\partial i$, in EMOS/radian.
CONSP	U	REAL8	Speed conversion factor, EMOS to km/sec.

OMASS EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
DMDVC	SU	REAL8	dm_o/dv_c, in kg/EMOS.
DVI00	U	GUNCOM	$\partial v_g/\partial i_\infty$, in EMOS/radian.
DVV00	U	GUNCOM	$\partial v_g/\partial v_{\infty o}$.
QDECL	U	LOGIC4	Non-coplanar launch maneuver indicator.
V0DIV	S	GUNCOM	Inverse of circular orbit speed, $1/v_o$, in $EMOS^{-1}$.
XANG1	U	REAL8	Launch site latitude input to the program, in degrees.
XANG2	U	REAL8	Maximum permissible parking-orbit inclination, due to range safety considerations, input to the program, in degrees.
XINCL	SU	REAL8	Parking orbit inclination, i, in radians.
LDVI00	S	GUNCOM	Indicator to bypass computations of $\partial v_g/\partial i_\infty$ in subroutine GUNTHR.
LDVV00	S	GUNCOM	Indicator to bypass computations of $\partial v_g/\partial v_{\infty o}$ in subroutine GUNTHR.
MBOOST	UX		Launch vehicle selector.

CHART TITLE - SUBROUTINE OMASS(V)



ORIGINAL PAGE IS OF POOR QUALITY

OMASS-7

CHART TITLE - SUBROUTINE DMASS(V)

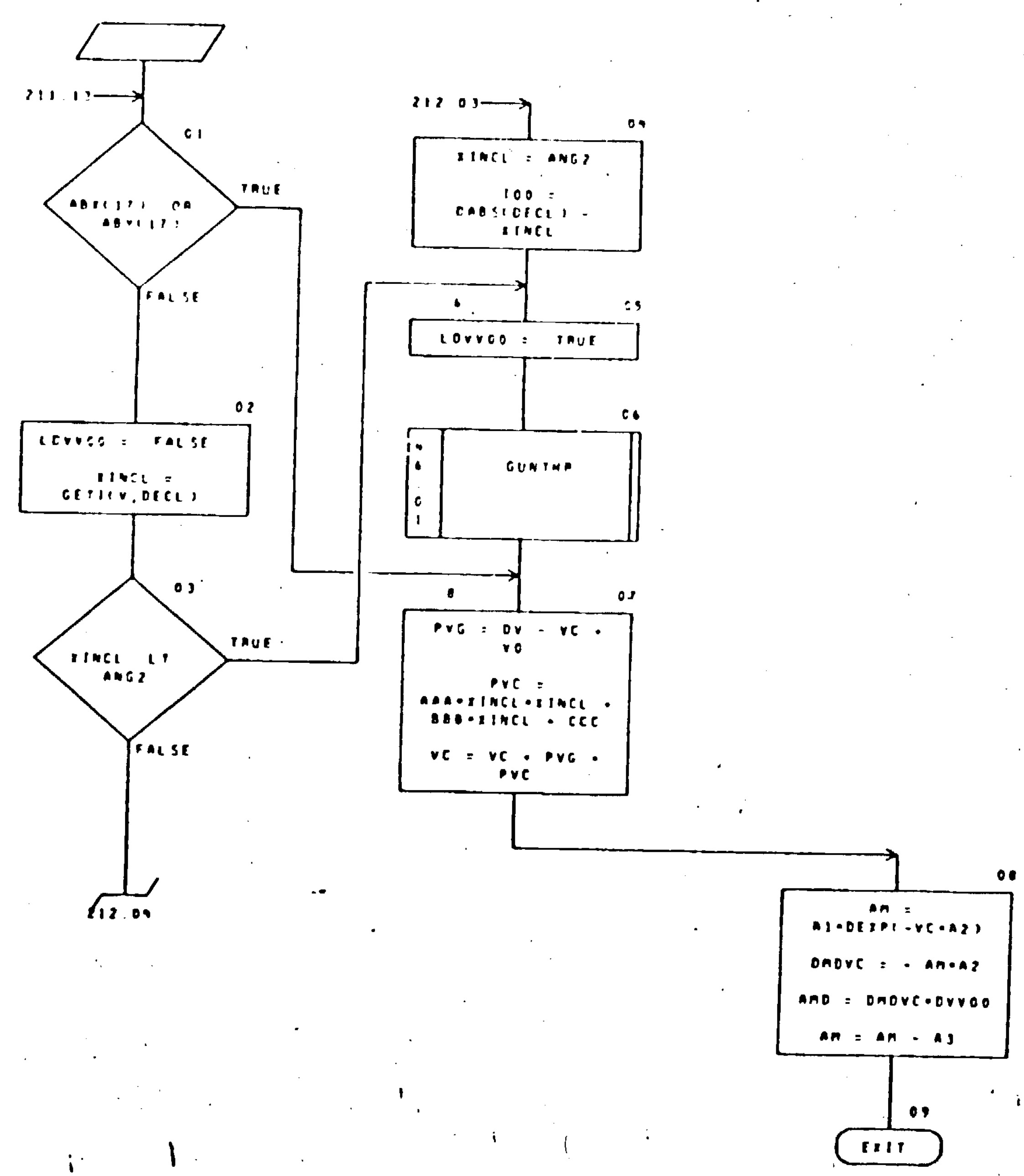


CHART TITLE - SUBROUTINE OMASS(V)

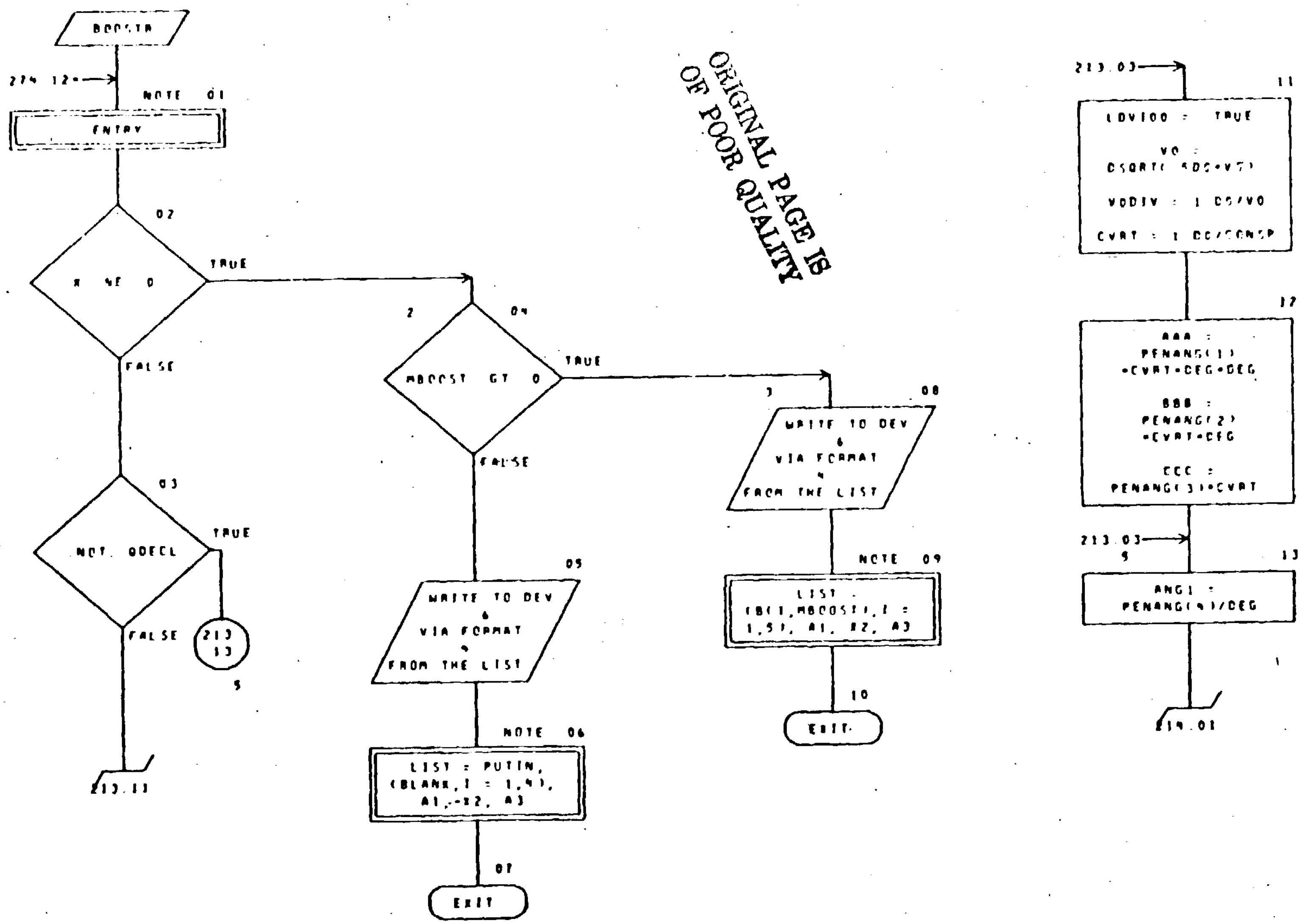


CHART TITLE - SUBROUTINE DMASSIV)

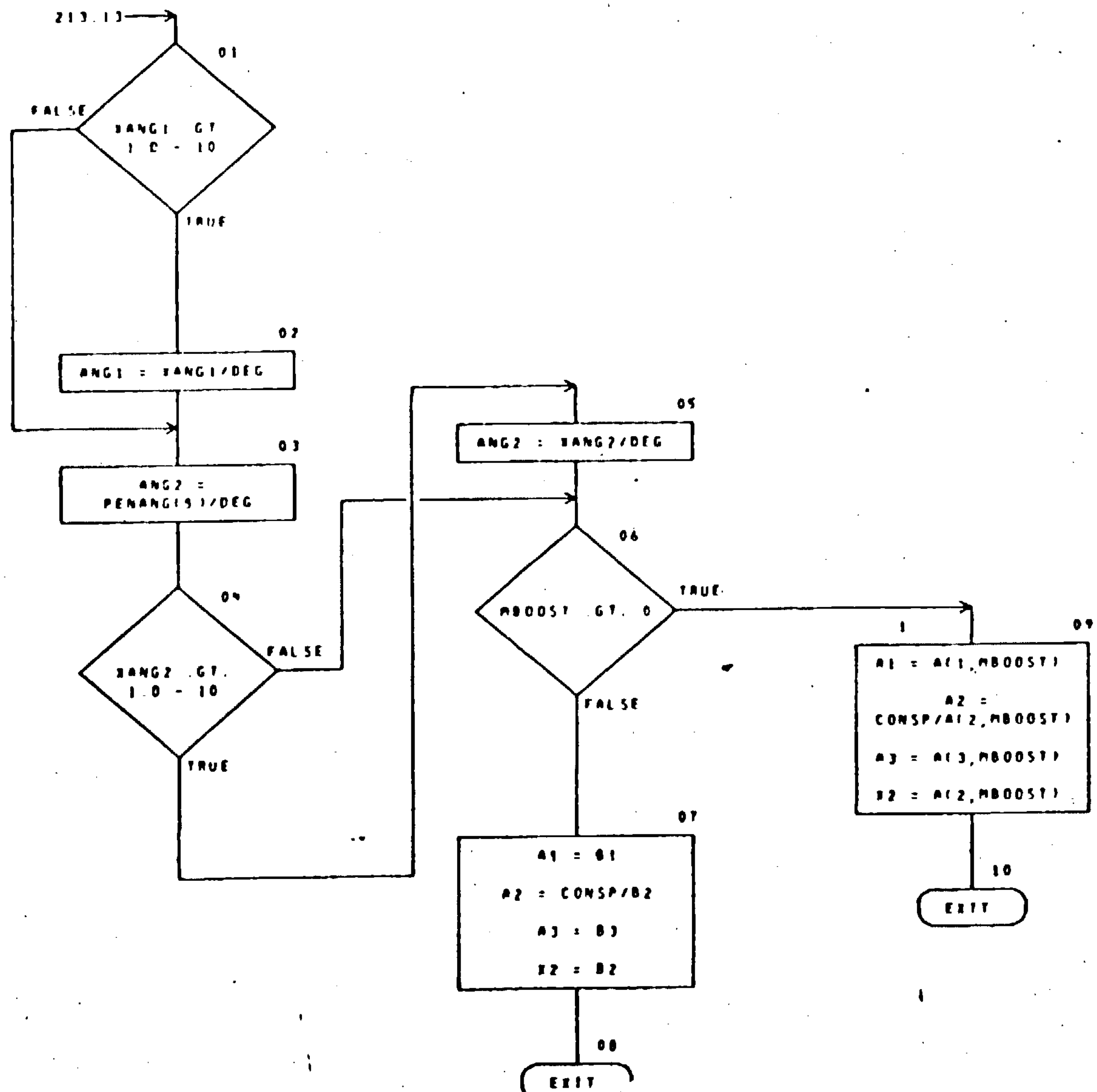


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
REAL*8 I00
LOGICAL LDVV00,LDV100,QDFCL,ABX,ABY
DIMENSION A(3,21),B(5,21),C(3,19),D(3, 2),E(5,19),F(5, 2),
PENANG(5)
COMMON /REALA/ RO1,AM,RO9(162),
                    AAA,BBB,CCC,ANG1,ANG2,RO2(3),DFCL,POA,
 XINCL,RO3(23),XANG1,XANG2,RO6(112),CFG,RO4(3),CONSP,RO5(R34),
AMO,R10(6),DMDVC,R11(13),FMS1,RO11827)
COMMON /LOGICN/LC1(7),QDFCL,LC2(217),ABX(70),ABY(70),LC3(135)
COMMON /ITERAT/ BO1(70C),D(70),BC2(210)
COMMON /GUNCOM/ V0,V0DIV,V00,I00,DV,DVV00,DV100,LDVV00,LDV100
EQUIVALENCE (C(1,1),A(1,1)),(D(1,1),A(1,20)),(E(1,1),B(1,1)),
(F(1,1),B(1,20))
DATA PENANG / .0566228513600, .9039433771D0,-71.1542972649ID0,20.5D0
,50.6D0/                          .1369151300/
DATA    VS /
DATA PUTIN,BLANK /5HINPUT,1H /
DATA C / 77360.13000,3652.191R00,1653.718000,
              385417.3600,2436.679600,1891 376900,
              399906.1900,2696.828700,1987 463R00,
              53329.55600,3950.673400,1759 268200,
              550692.8800,2428 476500,2445.796400,
              204190.9800,3556.909300,1610.652500,
              88811.88400,4079.921200,6315.053300,
```

ORIGINAL PAGE IS
OF POOR QUALITY

OMASS-11

CHART TITLE - NON-PROCEDURAL STATEMENTS

ORIGINAL PAGE IS
OF POOR QUALITY

```
         380020 70D0,2970 2699DC,1581 981200,
         721340 64D0,3221 5096D0,1317 190500,
         144286 44D0,3668 7527D0,1690 211500,
         41836 975D0,4499 6729D0,2293 2194C0,
         138726 52D0,3776 8656C0,1959 2524C5,
         50315 832D0,2199 9802C9,5544 4505C5,
         293355 54D0,2712 7944DC,3171 4496C5,
         637899 63D0,2391 6691DC,5975 5794D5,
         167238 95D0,3485 2038D0,1753 6969C5,
        2859382 94D0,1715 7632D0,1199 9231C0,
         963070 71D0,1853 9977C0,C636 764300,
         492596 49D0,2127 5511D0,C656 4230D0/
DATA D /  78549 05D0,5024 8482DC,2904 2553C0,
         178068 90D0,3359 7959D0,5786 9396C0/
DATA E /
 8HATLAS(SL,8HV3X)/CEN,8HTAUR    ,8H        ,1H ,
 8HTITAN II,8HI C     ,8H        ,8H        ,1H ,
 8HTITAN II,8HI C(1207,8H)       ,8H        ,1H ,
 8HTITAN II,8HI X/CENT,8HAUR     ,8H        ,1H ,
 8HTITAN II,8HI X(1207,8H)       ,8H        ,1H ,
 8HTITAN II,8HI X(1207,8H)/CENTAU,8HR       ,1H ,
 8HSATURN I,8HB/LM    ,8H        ,8H        ,1H ,
 8HSATURN I,8HB/CENTAU,8HR       ,8H        ,1H ,
 8HSATURN I,8HC/SIVB/C,8HENTAUR  ,8H        ,1H ,
 8HTITAN II,8HI X(1205,8H)/CENTAU,8HR       ,1H ,
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
     8HTITAN II,8HI DECORE,8H1/CENTAU,8HR          ,1H ,
     8HTITAN II,8HI DE1205,8H1/CENTAU,8HR          ,1H ,
     8HTAT(3C)/,8HDELTA/TE,8H364(1990,8H)          ,1H ,
     8HTITAN II,8HI D     ,8H         ,8H          ,1H ,
     8HTITAN II,8HI DE1205,8H1/CENTAU,8HR/TE364(,5H2250),
     8HTITAN II,8HI E/CENT,8HAUR      ,8H          ,1H ,
     8HSHUTTLE/,8HTRANSTAG,8HE        ,8H          ,1H ,
     8HSHUTTLE/,8HDELTA   ,8H         ,8H          ,1H ,
     8HSHUTTLE/,8HAGENA   ,8H         ,8H          ,1H /
    DATA F /
     8HSHUTTLE/,8HCENTAUR ,8H         ,8H          ,1H ,
     8HSHUTTLE/,8HCENTAUR/,8HBURNER I,8HI (2300),1H /
    FORMAT(1H0, 6X,18HLAUNCH VEHICLE IS ,5A8,
    *X15H(COEFFICIENTS =2E3F13.4,1H))
```

ORIGINAL PAGE IS
OF POOR QUALITY

OMASS-13

Name: PARINC

Calling Argument: P, KPART, LLL, MMM

Referenced Sub-programs: TRAJ

Referenced Commons: INTGR4, ITERAT, ITER2, LOGIC4

Entry Points: None

Referencing Sub-programs: MINMX3

Discussion: The function of this routine is to approximately determine the optimum values for the iterator's independent-variable perturbation-step-sizes. The program generates a partial derivative matrix of the dependent variables with respect to the independent variables by integrating trajectories neighboring the current nominal trajectory. The perturbation step size of each independent variable, which is used to vary that lone variable in order to generate its associated neighboring trajectory, may be input to the program. Each neighboring trajectory is used to generate one column of the partial derivative matrix, each element of which is constructed by forming the simple ratio of the difference between the neighboring and nominal dependent variable values to the perturbation step size. Each element of the matrix thus constructed represents the secant-slope-approximation to the actual dependent variable slope, and this approximation comes closest to the true value for some unique perturbation step size value for each independent variable. The approximation becomes poor for large step sizes because the secant-slope deviates farther from the true slope, and becomes poor for very small step sizes because the numerical accuracy of the computer and also of the trajectory generation algorithm with its numerous iterations introduces computational noise.

A program option controlled by the input variable KPART is available which attempts to determine the optimum perturbation step size for each independent variable. The program accomplishes this by taking a linear walk in the base-10 logarithm of each independent variable step size starting from the input or default value and not exceeding KPART steps. The program first steps in each direction (smaller and larger step size) to determine the proper direction of the walk, and each step consists of varying the step size one-half order-of-magnitude. For each linear walk (for each independent variable), that one column of the partial derivative matrix associated with the independent variable is computed for each step of the walk, and each element of that column is compared between the n^{th} step and the $(n+1)^{th}$ step of the walk. The element which has the largest normalized error in comparing the n^{th} and $(n+1)^{th}$ steps is selected as the criterion function, and this maximum-error-element is allowed to vary as the walk progresses. The walk continues until the criterion function is minimized, at which point the optimum perturbation step size is considered to be determined to within one-half order-of-magnitude. The process is repeated for each independent variable perturbation step size, to arrive at an optimum set of step sizes, which are then input to the program's iterator in place of the original values. A summary of the step size optimization is printed.

Messages and Printouts: A summary of the results of the perturbation step size optimization is printed as follows:

AUTOMATED SELECTION OF OPTIMUM PARTIALS-INCREMENTS

I	ITERATIONS	INITIAL	OPTIMUM
j	m	step (0)	step (opt)
⋮	⋮	⋮	⋮

where j is the index representing the independent variable, m is the total number of steps executed in the linear walk discussed in the preceding section, step (0) is the initial value of the independent variable perturbation step size, and step (opt) is the new value of the step size, which will differ from step (0) by a factor of either $10^{m/2}$ or $m^{-m/2}$; step (opt) is the optimum step size unless m equals KPART, in which case the end of the linear walk was probably reached before attaining the optimum step size.

If an error condition occurs while executing the trajectory subroutine TRAJ, the following message is printed

TRAJ ERROR I = __n__ LA = __i__

after which execution continues; n is the current number of steps in the linear walk, and i is the loop index of the independent variable.

If the total number of either the independent or the dependent variables exceeds 25, the following message is printed at entrance to the subroutine:

ABOVE OPTION SKIPPED.
PARTIALS-MATRICES EXCEEDING 625 LOCATIONS NOT ALLOWED FOR THIS OPTION (TO CONSERVE STORAGE SPACE).

and the subroutine is exited.

PARINC EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(35)	SU	ITER2	Iterator independent variable array.
P(625)	SUX		Iterator partial derivative matrix (of dependent variables with respect to independent variables).
Q(35)	U	ITER2	Iterator dependent variable array.
BB(35)	SU	ITER2	Iterator independent variable perturbation step size array.

PARINC EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
BX(5, 70)	S	ITERAT	Program input independent variable array.
LL(70)	U	INTGR4	Array containing the index set of the independent variables.
LLL	UX		Total number of independent variables for the current problem.
MMM	UX		Total number of dependent variables for the current problem.
CONX(70)	U	ITERAT	Array of conversion factors for the independent variables.
KPART	UX		Program input which triggers the use of this subroutine; maximum number of steps allowed in any given linear walk.
WONDER	SU	LOGIC4	Logical indicator which allows or disallows the program to consider if the current trajectory is in the proximity of a propulsion-time corner.

CHART TITLE - SUBROUTINE PARINC(P,KPART,LLL,MMM)

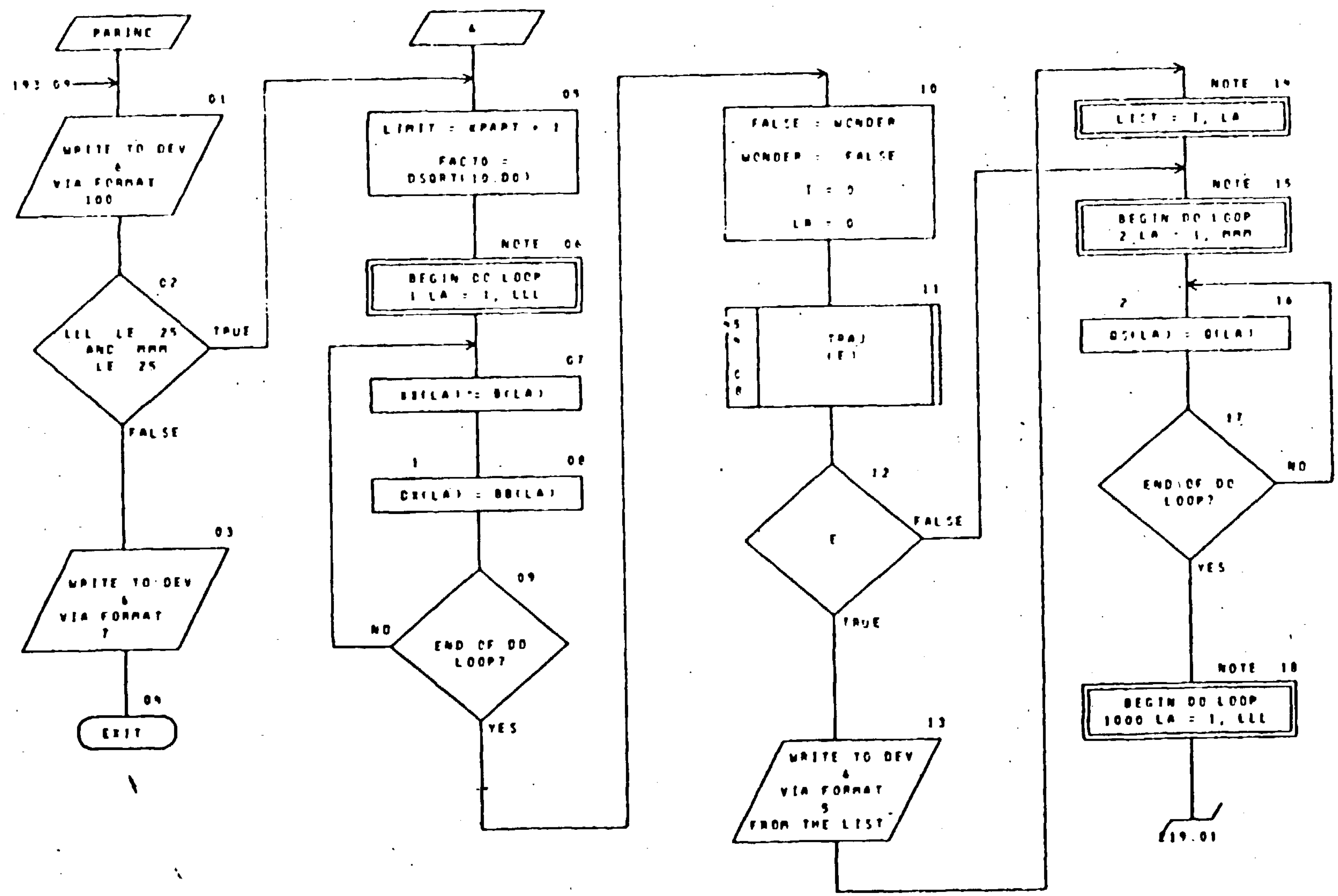


CHART TITLE - SUBROUTINE PARINC(P,XPART,LLL,MMM)

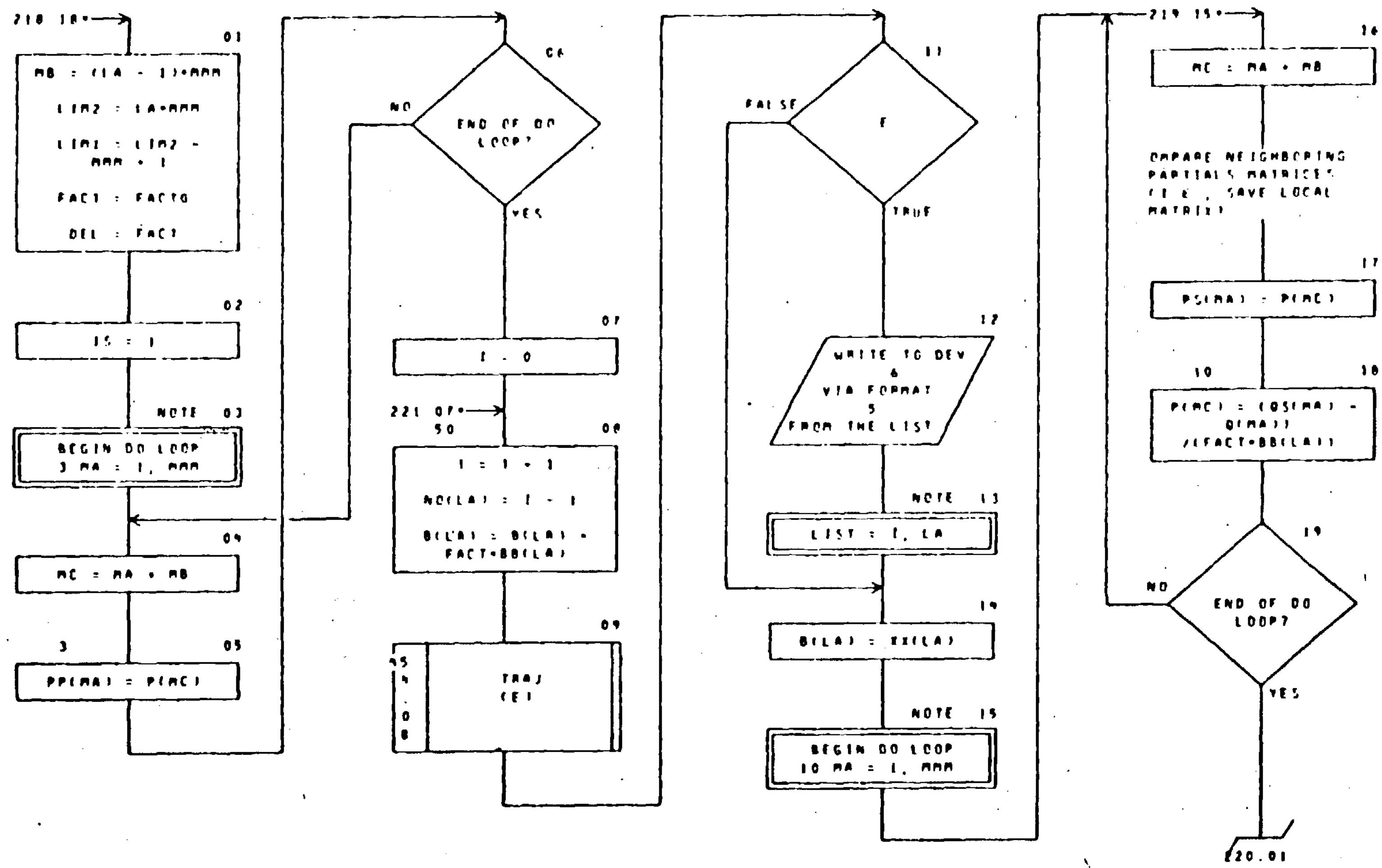


CHART TITLE - SUBROUTINE PARINC(P,NPART,LLL,MMM)

C-5

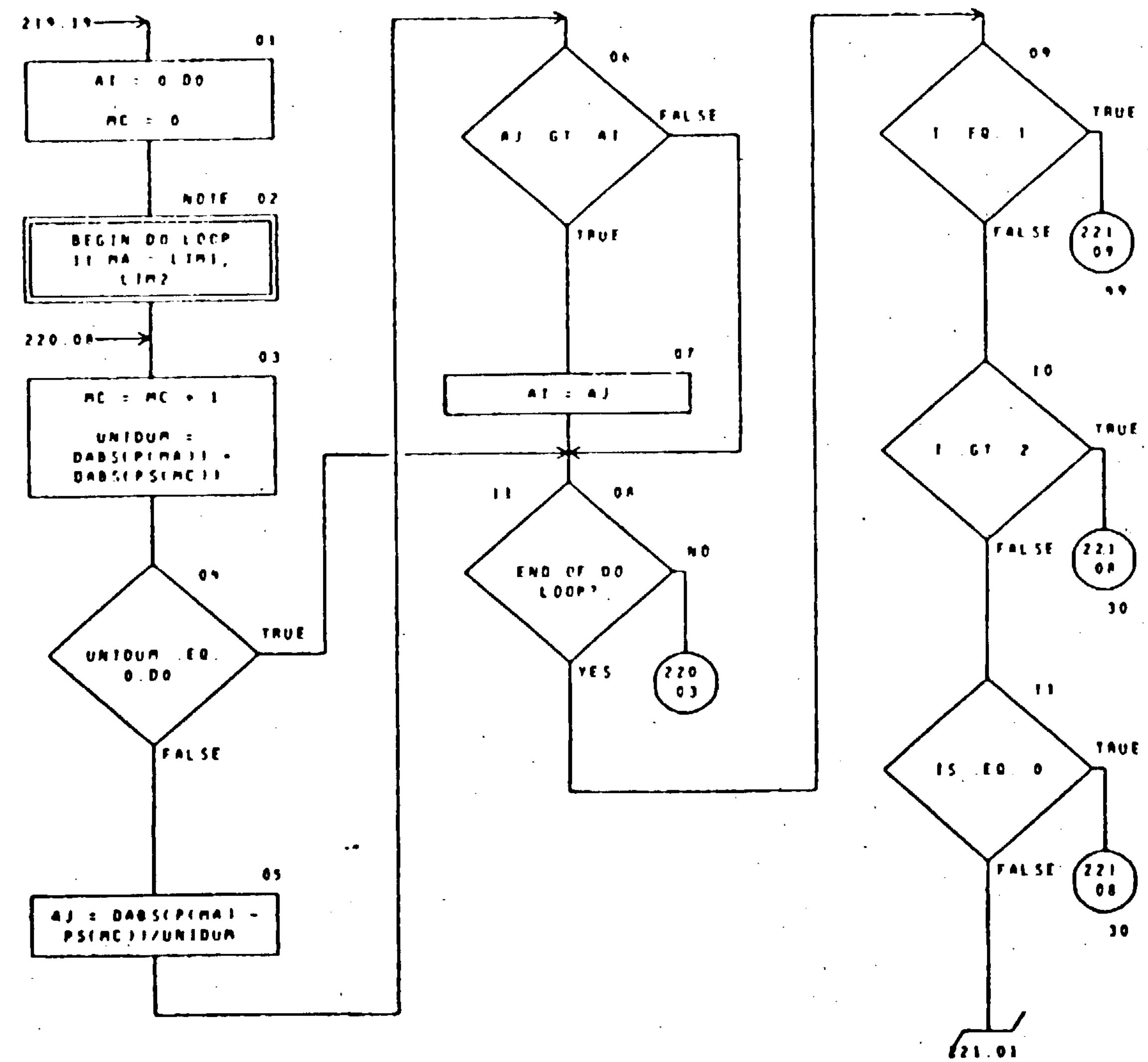


CHART TITLE - SUBROUTINE PARINC(P,KPART,LLL,MMM)

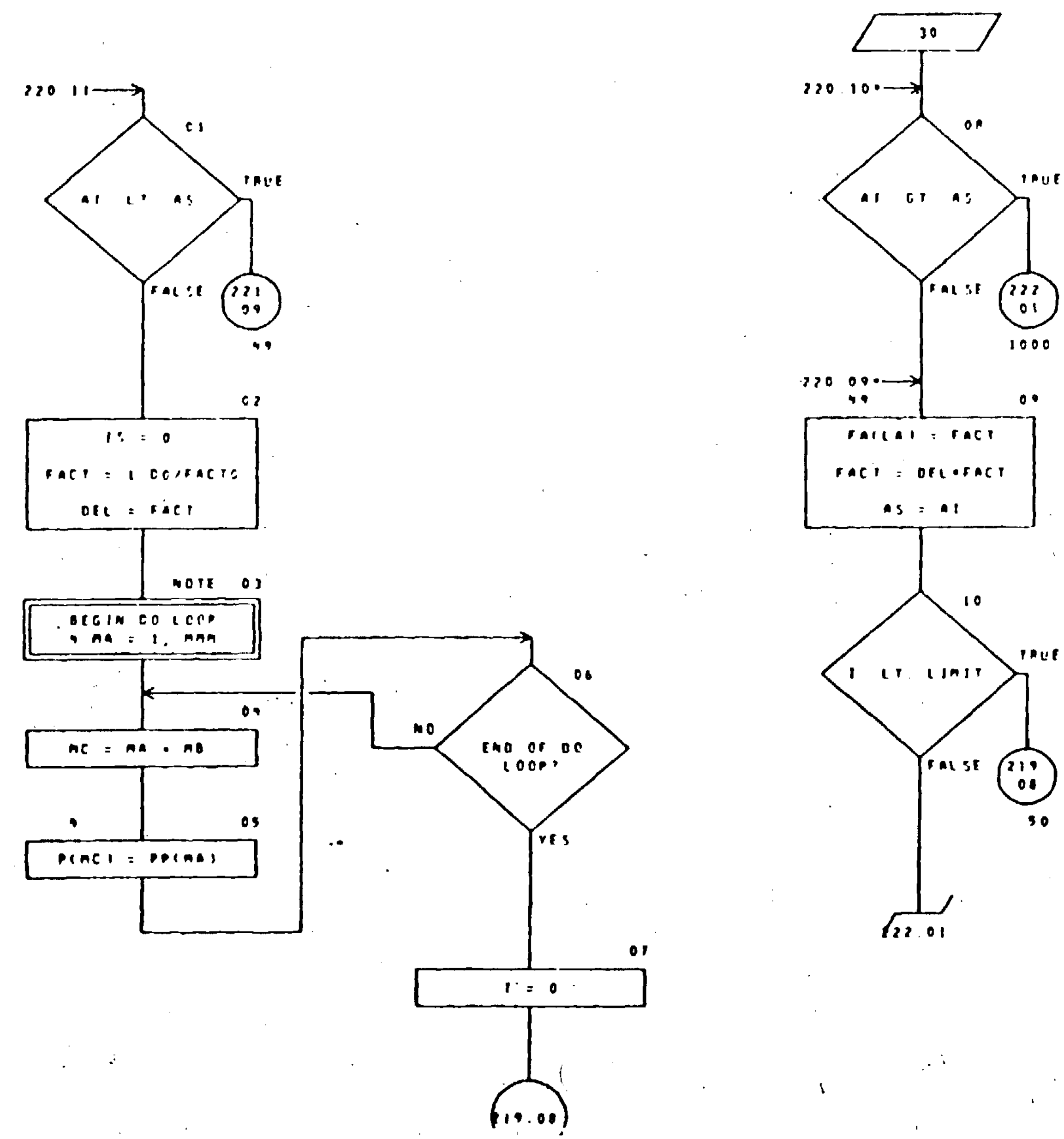
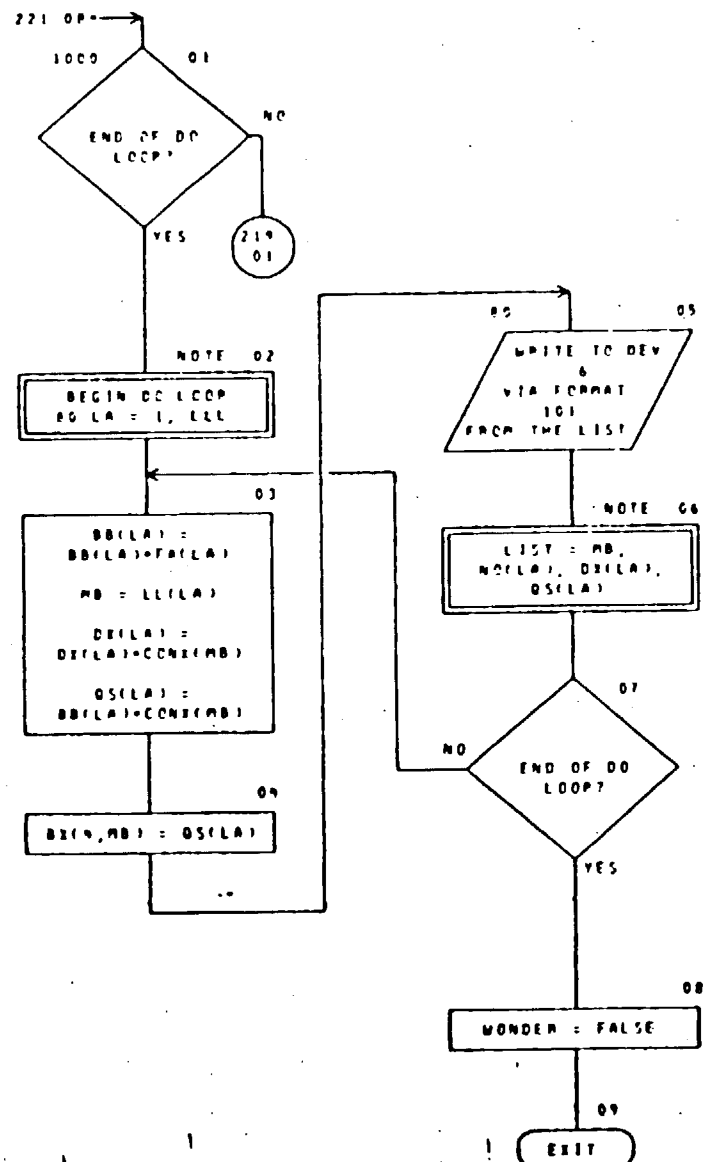


CHART TITLE - SUBROUTINE PARINC(P,KPART,LLL,MMM)



PARI NC-9

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL F,WONDER,FALSE
      DIMENSION P(625),PS(25),PP(25),XX(25),DX(25),FA(25),OS(25),NO(25)
      COMMON /INTGRS/ IO1(185),LL(70),IO2(795)
      COMMON /LOGICS/ LO1(4),WONDER,LO2(495)
      COMMON /ITERAT/ BX(5,70),BO1(210),CONX(7C),BO2(350)
      COMMON /ITER2/ B(35),Q(35),PO1(140),BB(35),PO2(1295)
7     FORMAT(1H ,21HABOVE OPTION SKIPPED /1H ,
     559HPARTIALS-MATRICES EXCEEDING 625 LOCATIONS NOT ALLOWED ,
     444HFOR THIS OPTION (TO CONSERVE STORAGE SPACE)./1H )
100   FORMAT(1H1,50HAUTOMATED SELECTION OF OPTIMUM PARTIALS-INCREMENTS
     /1HO,4X1HI,5X10HITERATIONS,5X7HINITIAL,11X7HOPTIMUM/1H )
5     FORMAT(1H ,10HTRAJ ERROR,5X3HI =I3,5X4HLA =I3)
101   FORMAT(1H ,I5,I11,2D20.8)
```

ORIGINAL PAGE IS
OF POOR QUALITY

<u>Name:</u> PDATE

<u>Calling Argument:</u> TDATE, IY, IM, ID, HOUR

<u>Referenced Sub-programs:</u> None

<u>Referenced Commons:</u> None

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> EFMPRT, QPRINT

<u>Discussion:</u> Subroutine PDATE evaluates the calendar date, given the Julian date. The procedure is to first subtract the Julian date (with leading 24 omitted) of 15020.5 from the input date to obtain the number of days from January 1.0, 1900. The year of the input date is then determined by entering a loop and accumulating the sum of days in all years from 1900 until the sum exceeds the number of days first determined on entry to PDATE. The number of days from the start of that year is then determined and used, with the aid of a data table, to define the month and day. Any fraction of a day remaining is then converted to hours and a return to the calling program is executed. The subroutine will not work correctly for input dates earlier than 1900 or later than 2100.

<u>Messages and Printouts:</u> If a date later than the year 2100 is input, the following message is printed:

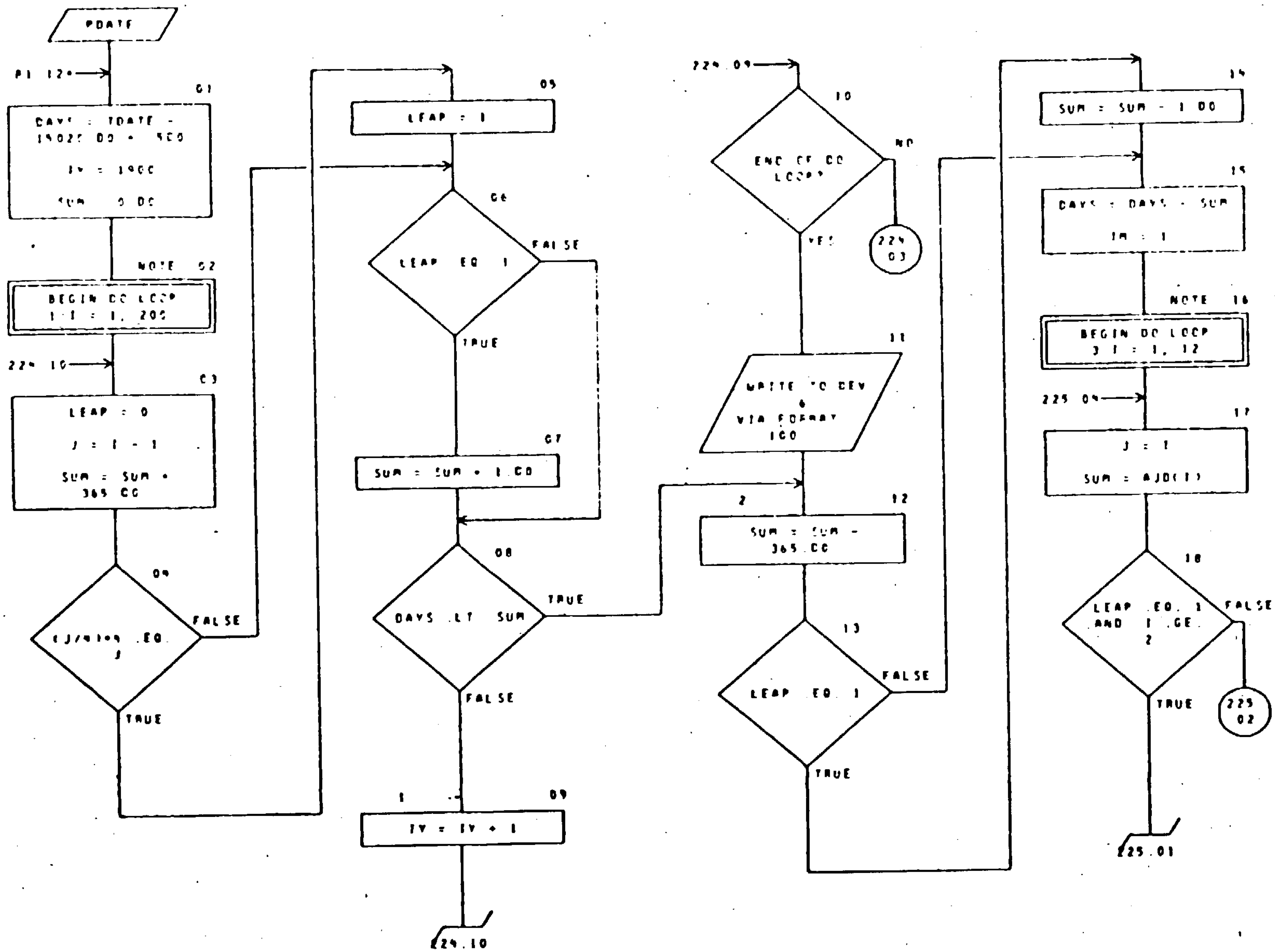
PDATE ERROR

and execution continues.

PDATE EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
ID	SX		Calendar day of the month.
IM	SUX		Calendar month index.
IY	SUX		Calendar year.
HOUR	SX		Hour of the day.
TDATE	UX		Input Julian date, with leading 24 omitted.

CHART TITLE - SUBROUTINE PDATE(TDATE,IY,IM,ID,HOUR)



PDATE-3

CHART TITLE - SUBROUTINE PDATE(TDATE,IY,IM,ID,HOUR)

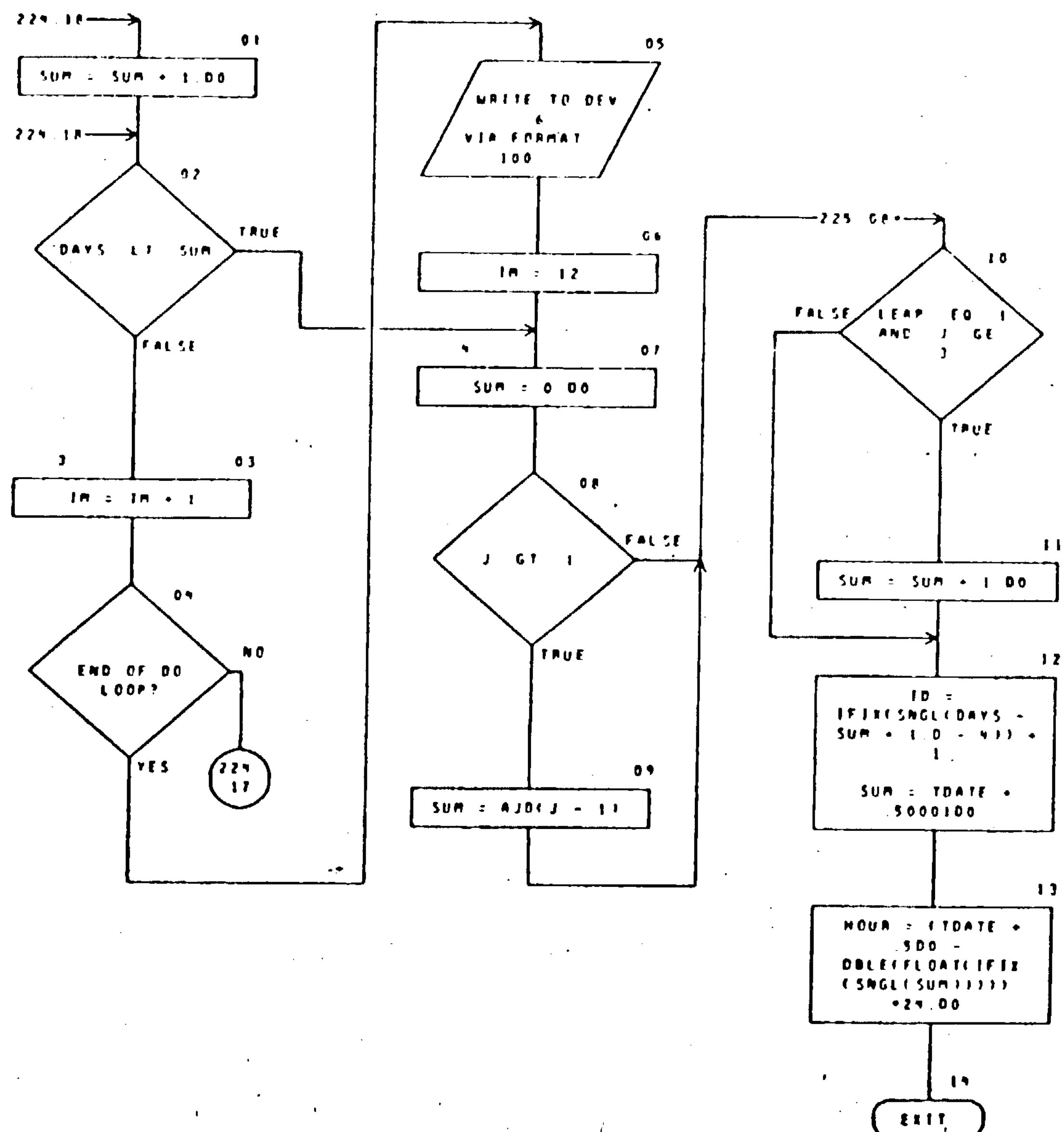


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION AJD(12)
      DATA AJD /31 D0,59 D0,90 D0,120 D0,151 D0,181 D0,212 D0,243 D0,
      273 D0,304 D0,334 D0,365 D0/
100   FORMAT(1H0,11HPDATE ERROR)
```

ORIGINAL PAGE IS
OF POOR QUALITY

PDATE-5

<u>Name:</u> PMPINT

<u>Calling Argument:</u> /PM/, KQ, KB

<u>Referenced Sub-programs:</u> None

<u>Referenced Commons:</u> None

<u>Entry Points:</u> PMPRNT

<u>Referencing Sub-programs:</u> MINMX3

<u>Discussion:</u> This subroutine stores the integer values representing the dimensions of the partial derivative matrix of the dependent variables with respect to the independent variables of the two-point boundary value problem solved by the MINMX3 iterator, and also determines the location and two-dimensional array specification of the partial derivative matrix, used in subsequent calls to entry point PMPRNT.

<u>Messages and printouts:</u> The entry point PMPRNT performs the output of the partial derivative matrix to the high speed printer in the following format:

```
(D13.5) ____  ____  ____  ____  ____  ____  ____  ____  ____
                          *
____    ____  ____  ____
____    ____  ____  ____  ____  ____  ____  ____  ____  ____
                          *
____    ____  ____  ____
                .
                .
                .
                .
                .
 **
____    ____  ____  ____  ____  ____  ____  ____  ____  ____
                          *
____    ____  ____  ____
```


which corresponds to

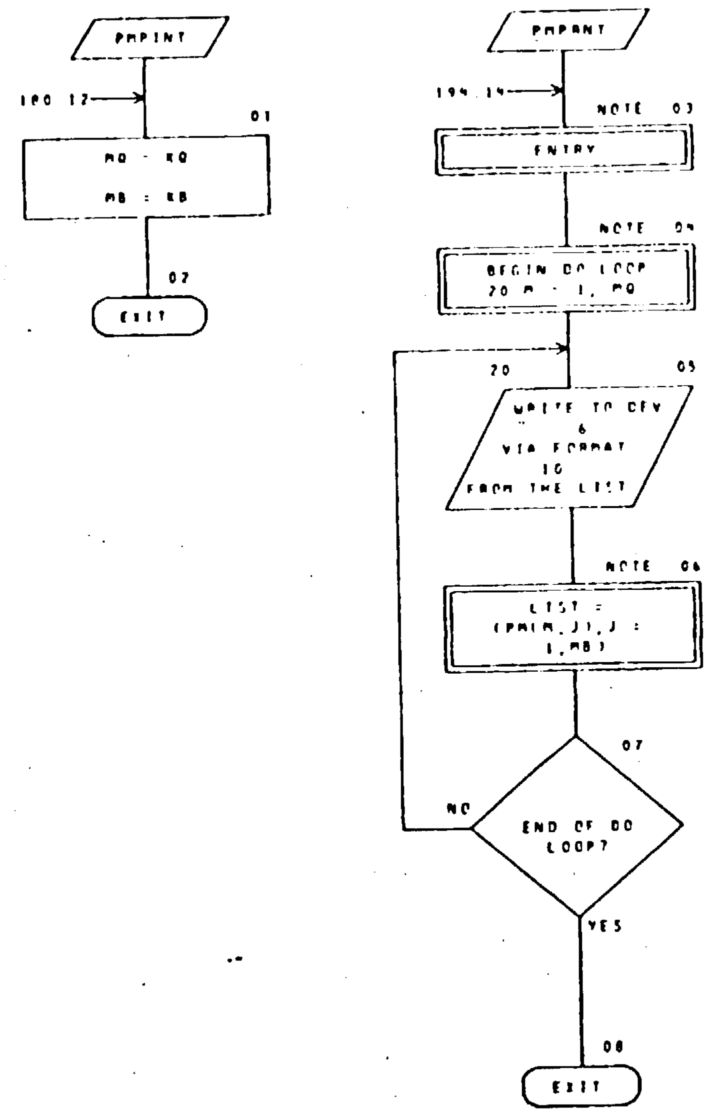
$$
\begin{bmatrix}
\frac{\partial y_1}{\partial x_1} & \frac{\partial y_1}{\partial x_2} & \cdots & \frac{\partial y_1}{\partial x_n} \\
\frac{\partial y_2}{\partial x_1} & \frac{\partial y_2}{\partial x_2} & & \\
\vdots & & \ddots & \\
\frac{\partial y_m}{\partial x_1} & \cdots & \cdots & \frac{\partial y_m}{\partial x_n}
\end{bmatrix}
$$

where n is the number of independent variables and m is the number of dependent variables, and * represents $\frac{\partial y_i}{\partial x_n}$, $i = 1, 2, \ldots, m$ and ** represents $\frac{\partial y_m}{\partial x_1}$.

PMPINT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
KB	SUX		The number of independent variables in the two-point boundary value problem solved by the MINMX3 iterator.
KQ	SUX		The number of dependent variables, similar to KB.
PM (KQ, KB)	UX		The two-point boundary value problem partial derivative matrix used by the MINMX3 iterator.

CHART TITLE - SUBROUTINE PMPINT(/PM/,KQ,KB)



```
      IMPLICIT REAL*8 ( A-H, O-Z)
      DIMENSION PM(80,80)
 10   FORMAT(1H /1H 1P10D13.5,9(/2H  1P10D13.5))
```

ORIGINAL PAGE IS
OF POOR QUALITY

Name: PRINT

Calling Argument: None

Referenced Sub-programs: DECLIN

Referenced Commons: EXTREM, INTGR4, ITERAT, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: FINISH, SOLAR, TRAJ

Discussion: This routine prints, on unit 11, those iterator independent-variable and dependent-variable values which are "turned on" for the current case, i.e., which apply to the two-point boundary value problem currently being solved. According to the user's option, these values will be printed at each iteration (from subroutine TRAJ) and/or after the final, case-summary trajectory is generated (from subroutine FINISH). The output independent variables are particularly formatted such that they may be directly input back to the program, via NAMELIST, to start the trajectory on a continuation computer run.

Only on the final, case-summary trajectory, selected information is output on unit 12 which allows the user to obtain an at-a-glance capsule summary of poignant data regarding the iteration sequence and trajectory. Much of this is described in the Messages and Printouts Section below. One specific parameter which is output (except on all-ballistic or forced-thrusting trajectories) is the propulsion-corner proximity $\Delta\xi$, defined by the arbitrary, yet simple, relation:

$$\Delta\xi = 1 - \frac{\sigma_{min}}{\sigma_{max}} ,$$

where σ_{min} is the minimum value of the thrust switch-function magnitude $|\sigma|$ encountered along the trajectory, considering only extrema of σ and ignoring roots, and σ_{max} is the maximum value of $|\sigma|$ encountered. $\Delta\xi$ is designed to lie in the range $0 \le \Delta\xi \le 1$, and, in fact, the trajectory is generally considered to be not in very great proximity to a propulsion-time corner (See the discussion of subroutine CORNER) when $\Delta\xi < .99$, and only when $\Delta\xi$ is of the order of, e.g., .999 or .9999 should the analyst be concerned.

A few sporadic pieces of information are output on unit 6. Specifically, the numbers of computation steps during coasting flight and during thrusting flight, pertaining to the summary trajectory, are printed; also output (on units 6 and 12) is the "switch count history", which displays the number of thrust switch-points corresponding to each nominal trajectory of the iteration sequence; finally, when the non-due-east launch phase is invoked (involving the optimization of the geocentric departure asymptote declination δ), the angle $\Delta\delta$ by which the launch hyperbolic excess velocity is offset from the initial primer vector, which has geocentric declination δ_λ, is computed (and printed):

$$\Delta\delta = \delta_\lambda - \delta ,$$

when the geocentric right ascension of the excess velocity is equal to that of the initial primer vector, and

$$\Delta\delta = \pi - (\delta_\lambda + \delta) \text{ if } \delta_\lambda \geq 0;$$

$$\Delta\delta = -\pi - (\delta_\lambda + \delta) \text{ if } \delta_\lambda < 0 ,$$

when the geocentric right ascension of the excess velocity is 180 degrees from that of the initial primer.

Messages and printouts: The iterator independent-variables which are currently turned-on (active) are printed on unit 11, three per line maximum until exhausted, in a format and units which are entirely consistent with NAMELIST data-input, as follows (for example):

PRINT-3

X1 = ______,	X2 = ______,	X3 = ______,
X4 = ______,	X5 = ______,	X6 = ______,
X10 = ______,	X13 = ______,	X15 = ______,
X21 = ______,	X41 = ______,	X42 = ______,
X43 = ______		

This is followed by the active iterator dependent-variable values, each preceded by its identifying index, as follows (for example):

DEPENDENT VARIABLES

1 ______	2 ______	3 ______	4 ______	5 ______
6 ______	10 ______	13 ______	15 ______	21 ______
44 ______	45 ______	46 ______		

The units of these various dependent-variable values are as given in the description of program inputs.

The above information is printed whenever subroutine PRINT is called; the following information is printed only in conjunction with the case-summary trajectory.

The sequence of values of the number of thrust-switches (off-to-on or on-to-off) along each nominal trajectory of the iteration sequence is printed on units 6 and 12 (for example):

SWITCH-COUNT HISTORY 7.7.7.8.8.9.9.9.9.10.9.9.9/

in which (for this example) the first three trajectories of the iteration sequence had 7 thrust switch points, the fourth and fifth trajectories had 8 switch points, and so on; the last value is followed by a slash. When all trajectories of an iteration sequence have the same number of thrust switch points, say 9, the printed message reduces to

SWITCH-COUNT HISTORY ALL 9

The propulsion-corner proximity $\Delta\xi$, is written on unit 12:

PROPULSION CORNER PROXIMITY = $\underline{(\Delta\xi)}$, MIN SIGMA = $\underline{(\sigma_{min})}$

where $\Delta\xi$ and σ_{min} are defined in the discussion. This allows the analyst to determine how nearly the iteration sequence may be "hung" on a propulsion-time corner. It is not printed on unit 6 because the analyst can peruse the switch-function column of the Extremum Table of Selected Functions to make a similar determination.

The total number of computation steps during thrust, and the total number during coast, pertaining to the summary trajectory, are printed on unit 6:

______THRUST COMPUTE STEPS, ______ COAST COMPUTE STEPS

Selected summary information is then printed on unit 12, as follows:

RMIN = $\underline{(r_{min})}$, RMAX = $\underline{(r_{max})}$, DECL = $\underline{(\delta)}$, TAU = $\underline{(\tau)}$

where r_{min} and r_{max} are the minimum and maximum solar distances encountered by the spacecraft along the (case summary) trajectory, in AU, δ is the geocentric launch excess-velocity asymptote declination, in degrees, and τ is the total propulsion system on-time (i.e., the propulsion time), in days;

NET MASS = $\underline{(m_{net})}$, INHIBITOR = $\underline{(\lambda)}$, ANGLE = $\underline{(\theta_t)}$, POWER= $\underline{(p_{ref})}$

where m_{net} is the net spacecraft mass, in kilograms, λ is the iterator's inhibitor (values greater than about 10^{-4} imply there is convergence difficulty), θ_t is the travel angle, in degrees, and p_{ref} is the reference power, in kilowatts;

CASE $\underline{(n)}$ COMPLETED AT LINE NO. $\underline{(m)}$, (BRING X, L = $\underline{(i_1)}$: $\underline{(i_2)}$)

where n is the case number, m refers to the m^{th} printed line on unit 11, and i_1 and i_2 are the first and last line-numbers (corresponding to printed lines on unit 11) where the iterator independent-variable values, corresponding to the summary trajectory, may be found. This is very convenient if the computer system used has the capability of "bringing" or fetching information from a specified range of lines on a specified unit.

If nonzero, the angle $\Delta\delta$ by which the launch hyperbolic excess velocity is offset from the initial primer vector is printed in degrees on unit 6:

LAUNCH ASYMPTOTE OFFSET FROM PRIMER = $\underline{(\Delta\delta)}$ DEGREES.

PRINT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
BX(5, 70)	U	ITERAT	Array of iterator independent-variables and related parameters, input to the program.
OO(70)	U	ITERAT	Array of iterator independent-variables in units consistent with program input units.
ABX(70)	U	LOGIC4	Array of iterator independent-variable indicators, which selects the active variables.
ABY(70)	U	LOGIC4	Array of iterator dependent-variable indicators,which selects the active variables.
DEG	U	REAL8	Radians to degrees conversion factor.
FXL(70)	U	ITERAT	Array of iterator dependent-variable values, in program internal units.
NSW	U	INTGR4	The total number of thrust switch-points along the current trajectory.
TAU	U	REAL8	Propulsion time, τ, in tau = AU/EMOS.
CONY(70)	U	ITERAT	Array of iterator dependent-variable conversion factors (between program internal and external units).

PRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
DECL	U	REAL8	Departure asymptote declination, δ, in radians.
LINE	SU	INTGR4	Current number of lines which have been printed on unit 11 during the current computer run.
NSET(5)	U	INTGR4	Iteration-sequence control array.
NSWX(50)	SU	INTGR4	Contains the thrust-switch-point history (values of NSW) of the iteration sequence.
QJEX	U	LOGIC4	Detailed printout (case summary) indicator.
RMAX	U	REAL8	Maximum solar distance encountered by the spacecraft along the trajectory, r_{max}, in AU.
RMIN	U	REAL8	Minimum solar distance encountered by the spacecraft along the trajectory, r_{min}, in AU.
ANGLE	U	REAL8	Travel angle, θ_t, in radians.
CONTM	U	REAL8	Time conversion factor, tau to days.
KOUNT	U	INTGR4	Case number.
LDECL	U	LOGIC4	Indicator for the condition in which the magnitude of the departure asymptote declination exceeds the parking orbit inclination.
MAJOR	U	INTGR4	Number of nominal (major) trajectories comprising the iteration sequence.
NSPEC		INTGR4	Number of entries (lines) in the Extremum Table of Selected Functions.
ONOFF (2,100)	U	EXTREM	Storage array for thrust switching function, used in conjunction with the Extremum Table of Selected Functions.

PRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
PSIGN	U	REAL8	Coefficient defining the sense of the launch hyperbolic excess velocity relative to the initial primer vector.
QDECL	U	LOGIC4	Indicator that the computation of the departure asymptote declination is required as part of the optimization problem.
XMASS(7)	U	REAL8	General mass array and related parameters. XMASS(6) is reference power, p_{ref}, in watts.
DECLAM	U	REAL8	Geocentric declination of the initial primer vector, δ_{λ}, in radians.
FIXPOW	U	LOGIC4	Indicator for the launch-vehicle-independent mode of operation.
NSTEP1	U	INTGR4	Total number of computation steps associated with thrusting flight, for the current trajectory.
NSTEP2	U	INTGR4	Total number of computation steps associated with coasting flight, for the current trajectory.
PAYLOD	U	REAL8	Net spacecraft mass, m_{net}, in kilograms.
XAMBDA	U	REAL8	The iterator's inhibitor, λ.

CHART TITLE - SUBROUTINE PRINT

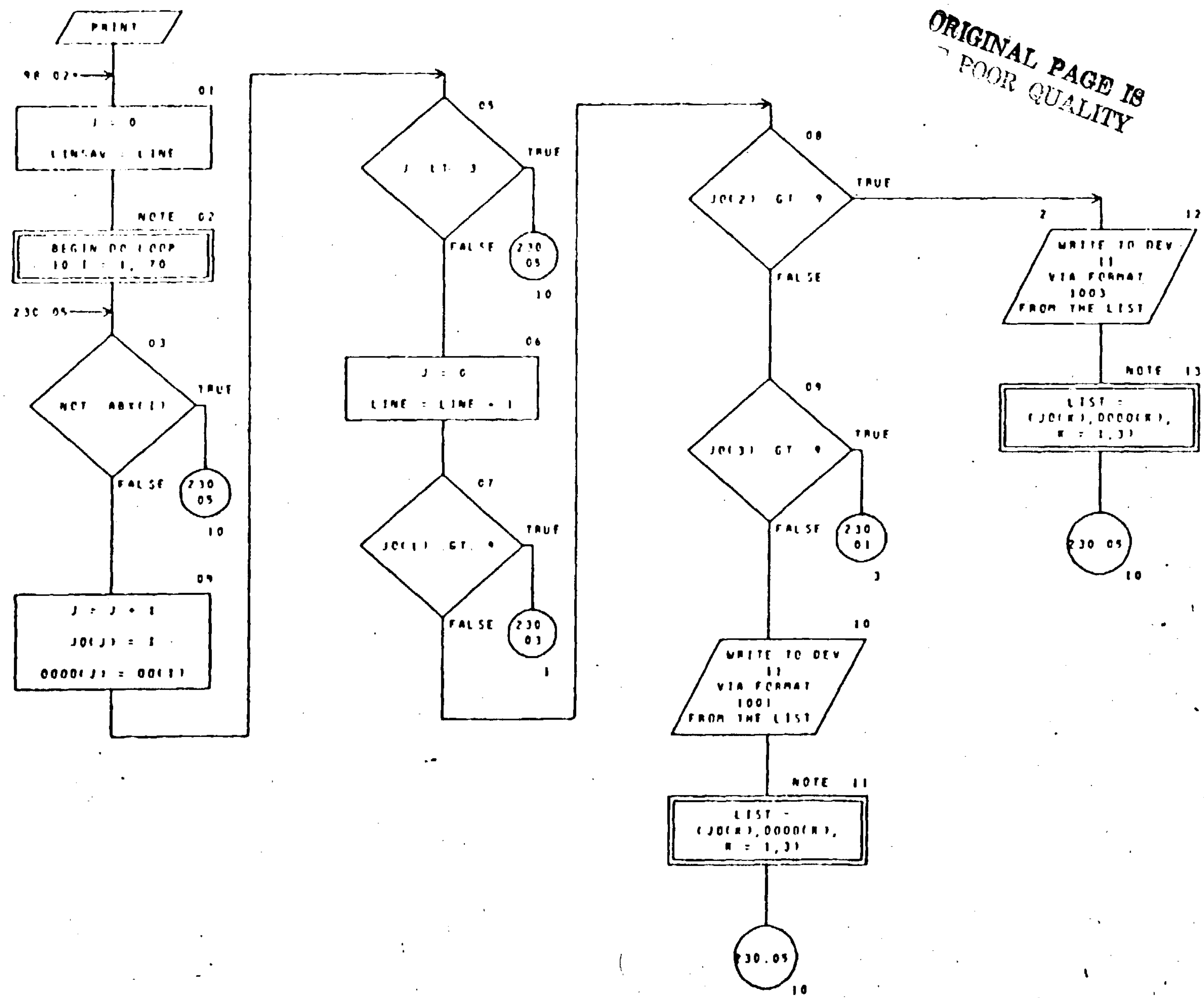


CHART TITLE - SUBROUTINE PRINT

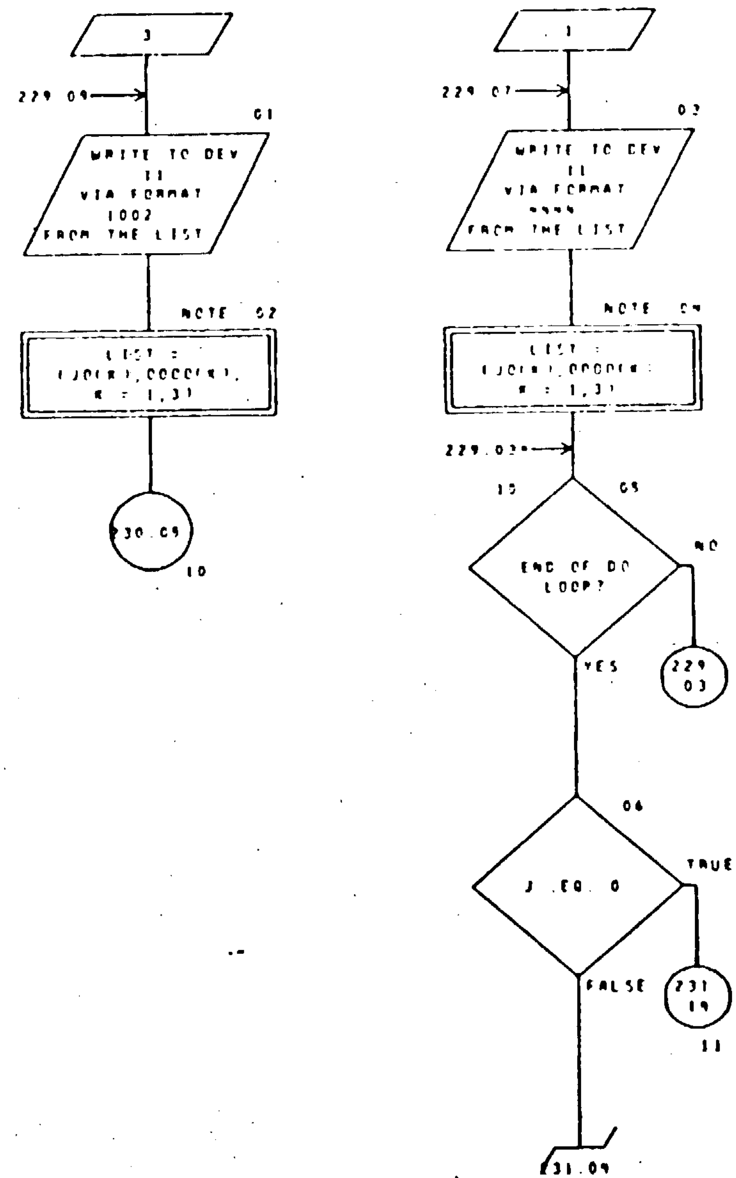


CHART TITLE - SUBROUTINE PRINT

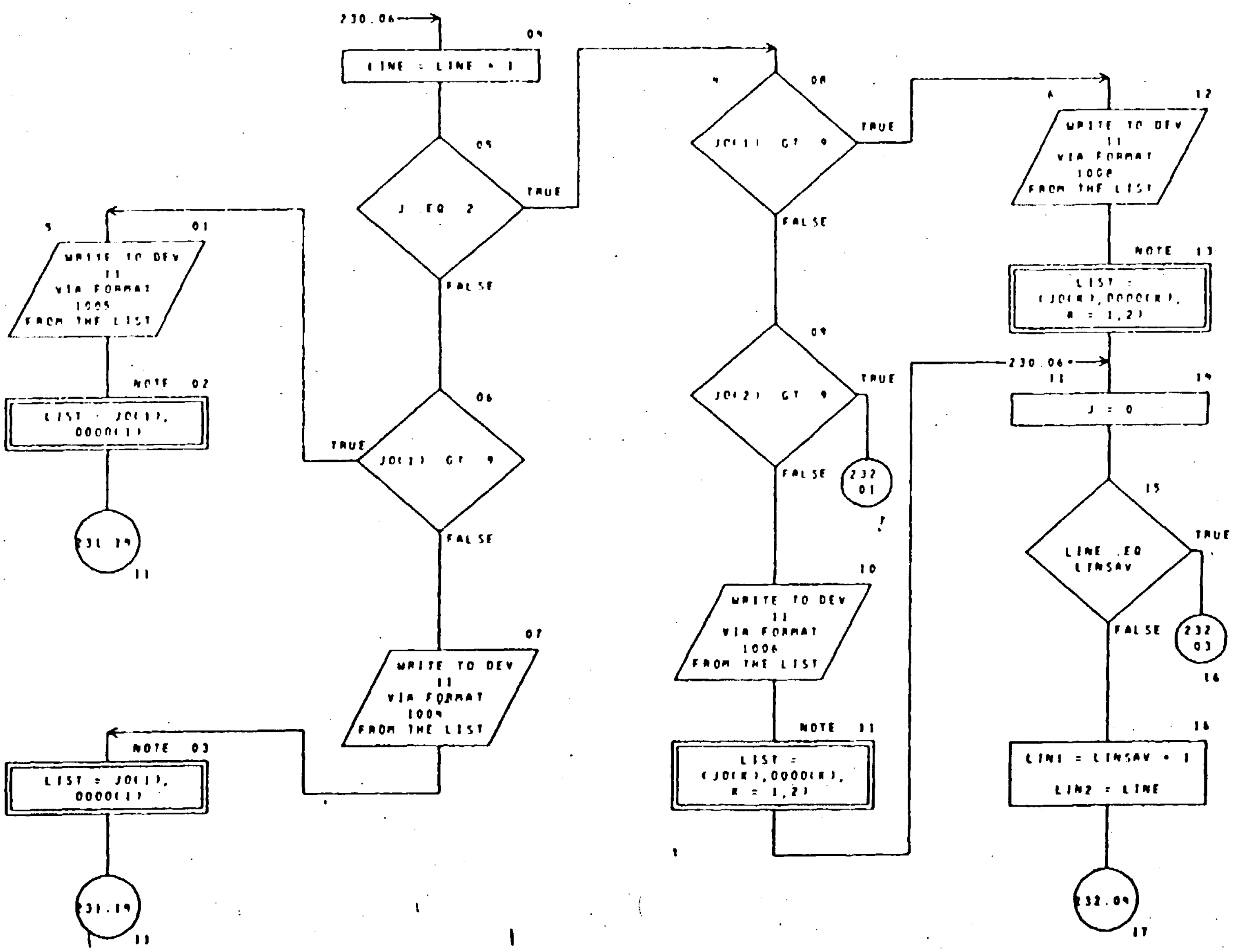
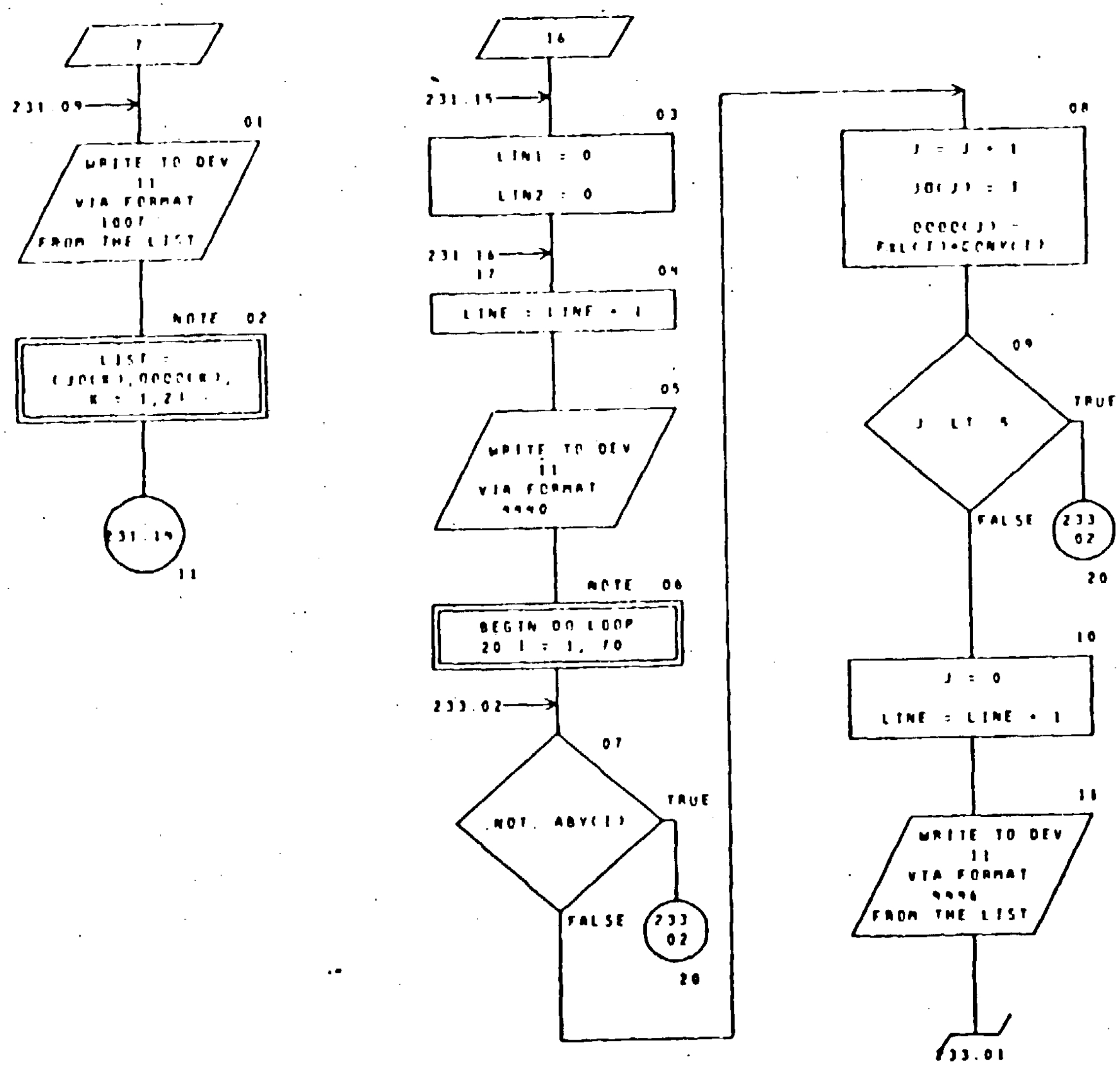


CHART TITLE - SUBROUTINE PRINT



PRINT-11

CHART TITLE - SUBROUTINE PRINT

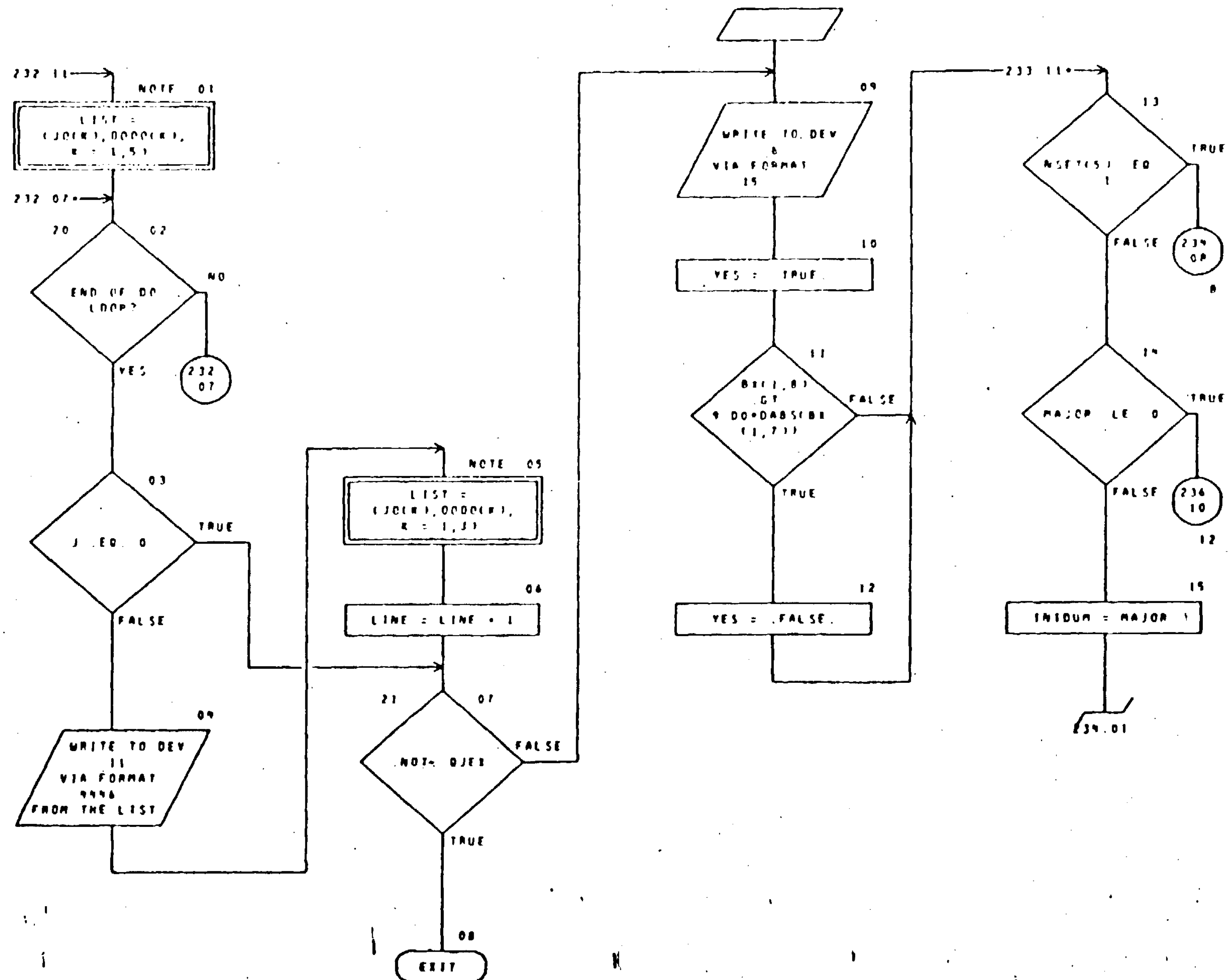
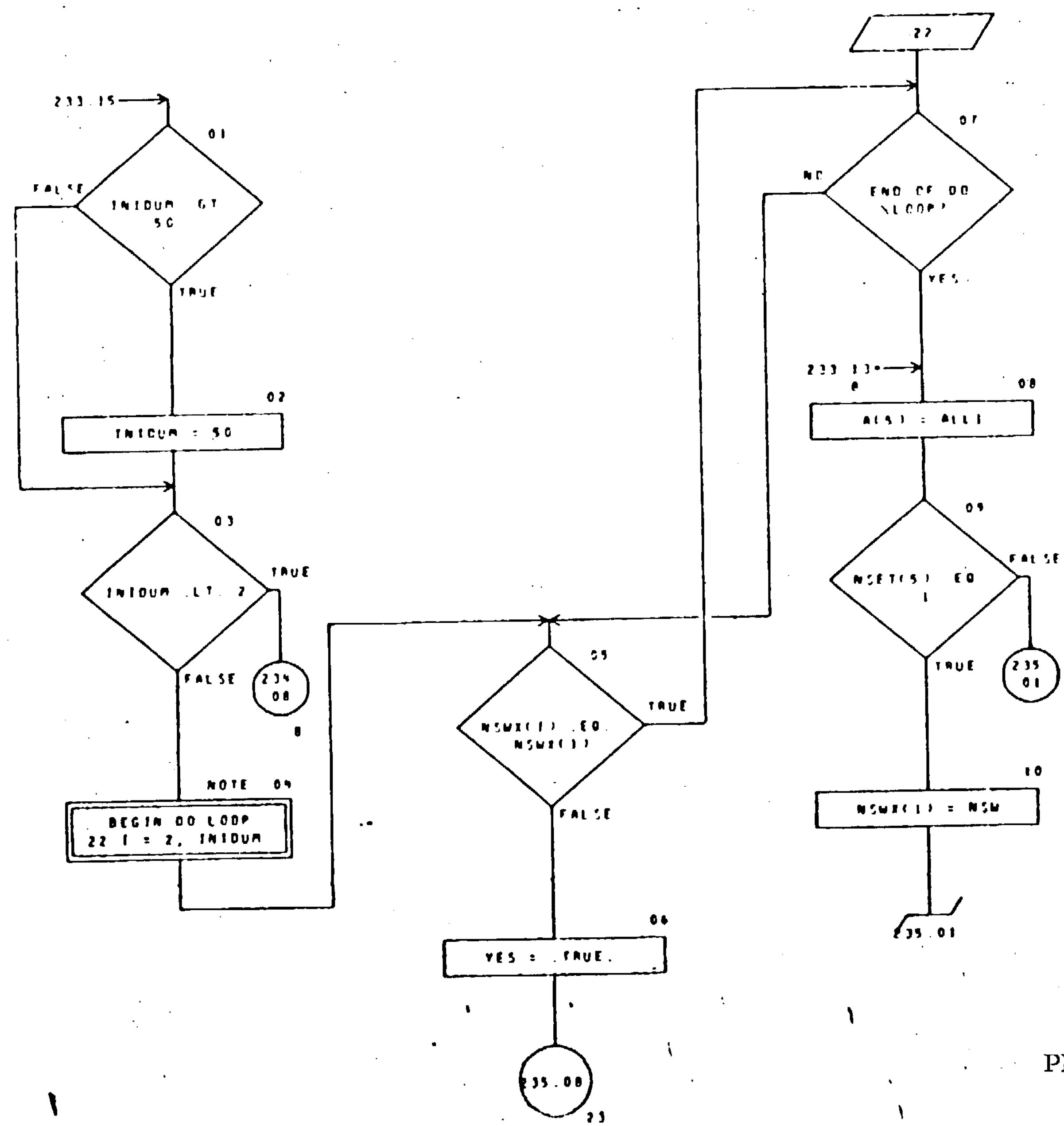


CHART TITLE - SUBROUTINE PRINT



PRINT-13

CHART TITLE - SUBROUTINE PRINT

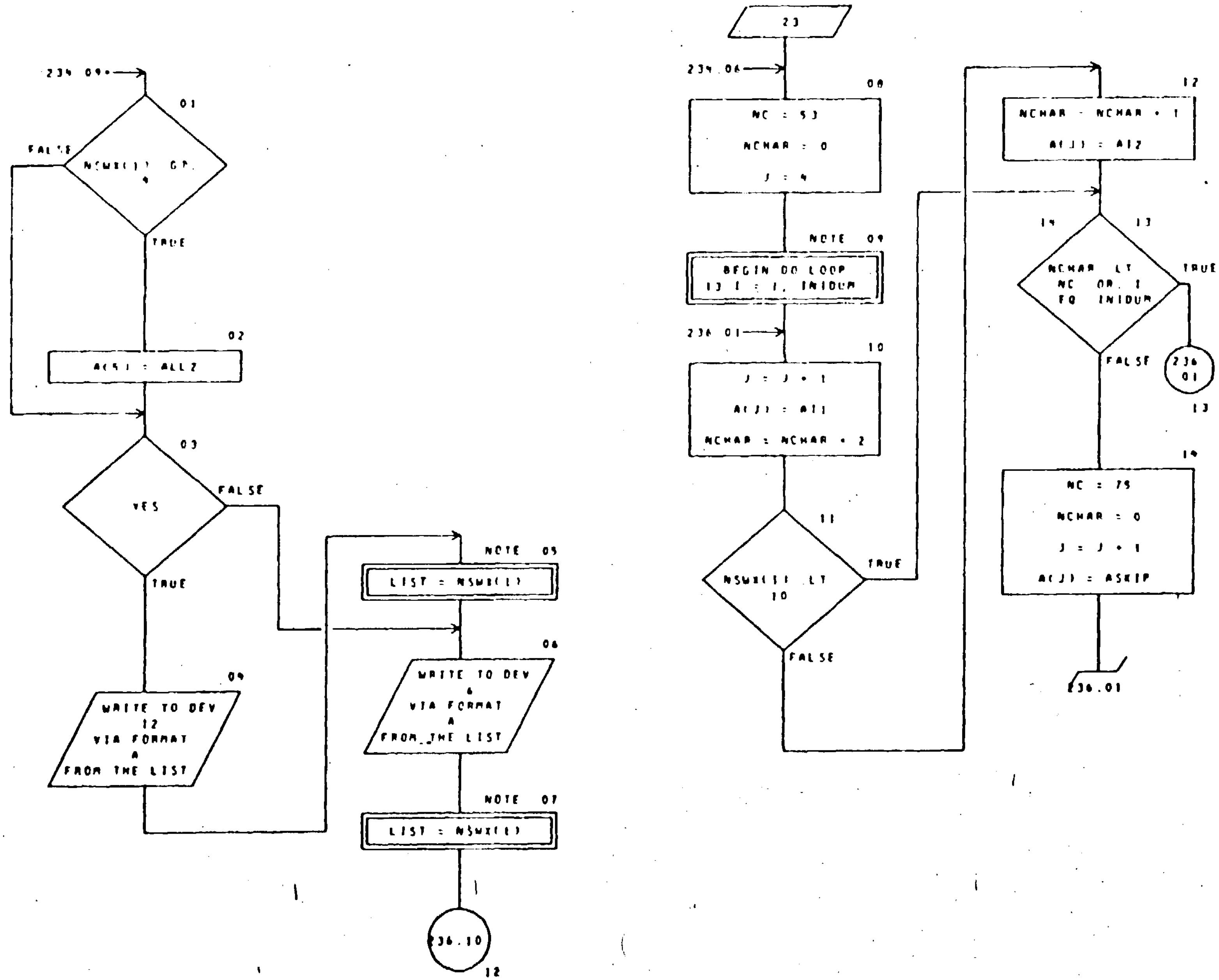
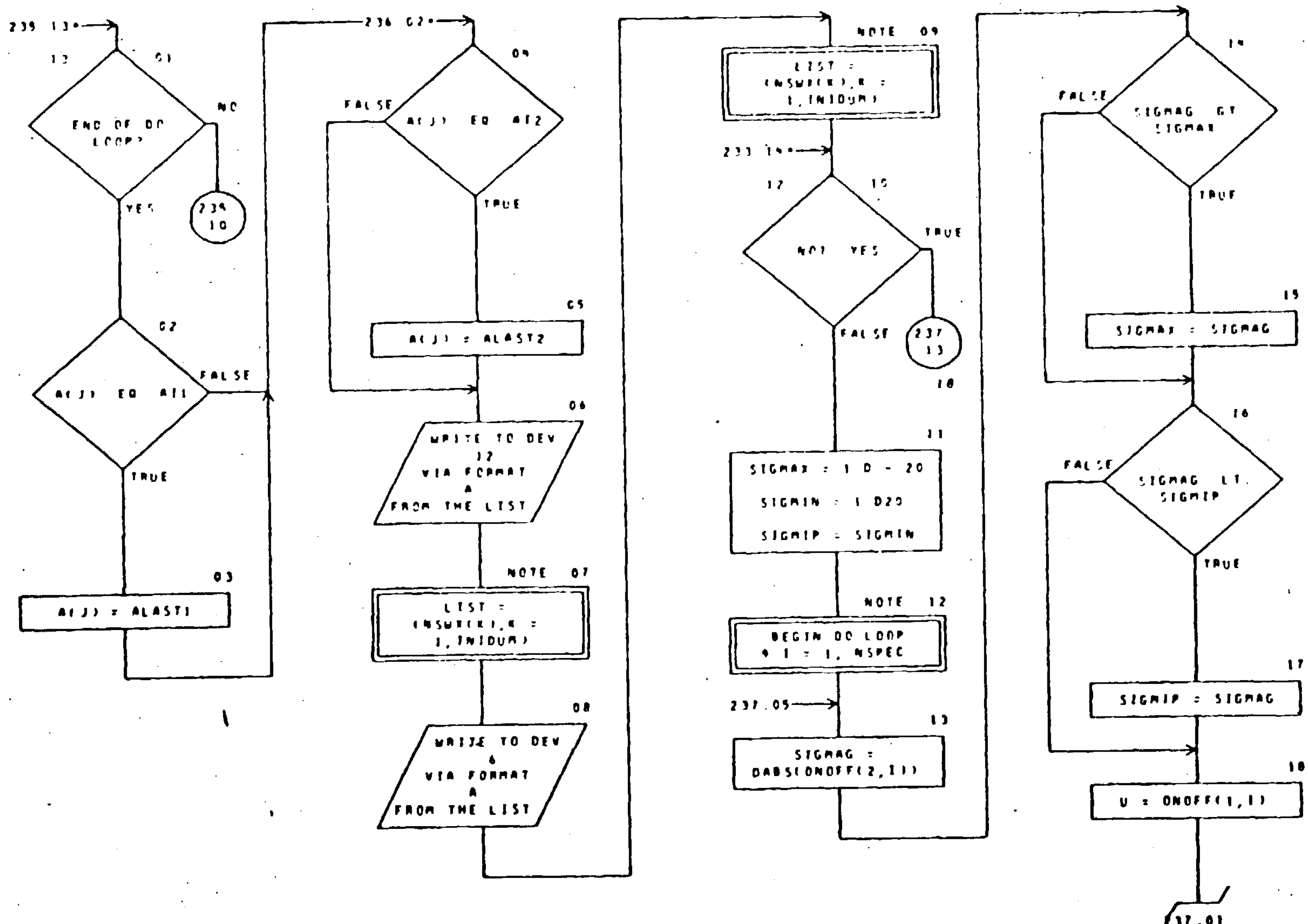


CHART TITLE - SUBROUTINE PRINT

ORIGINAL PAGE IS
OF POOR QUALITY



PRINT-15

CHART TITLE - SUBROUTINE PRINT

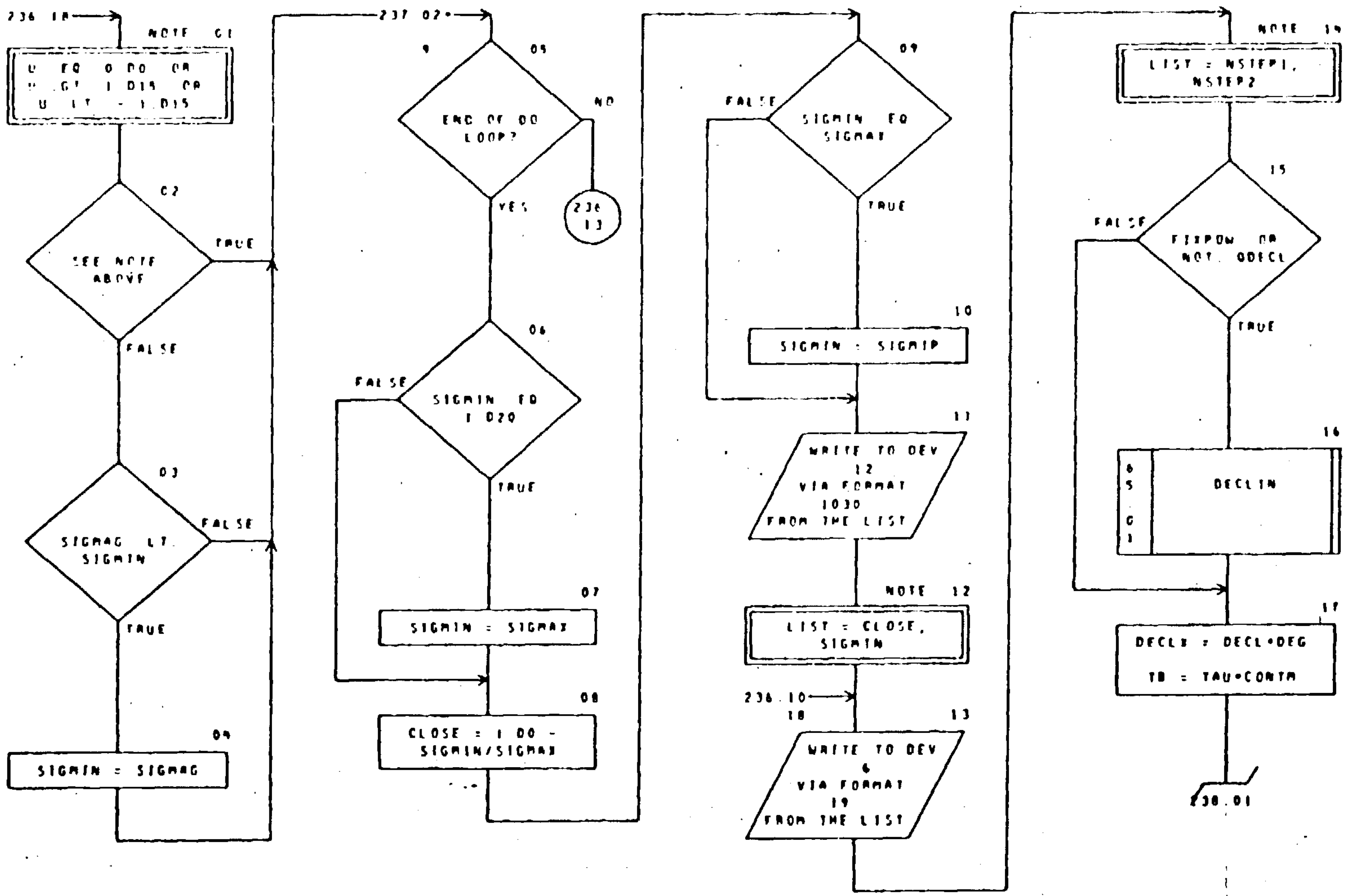
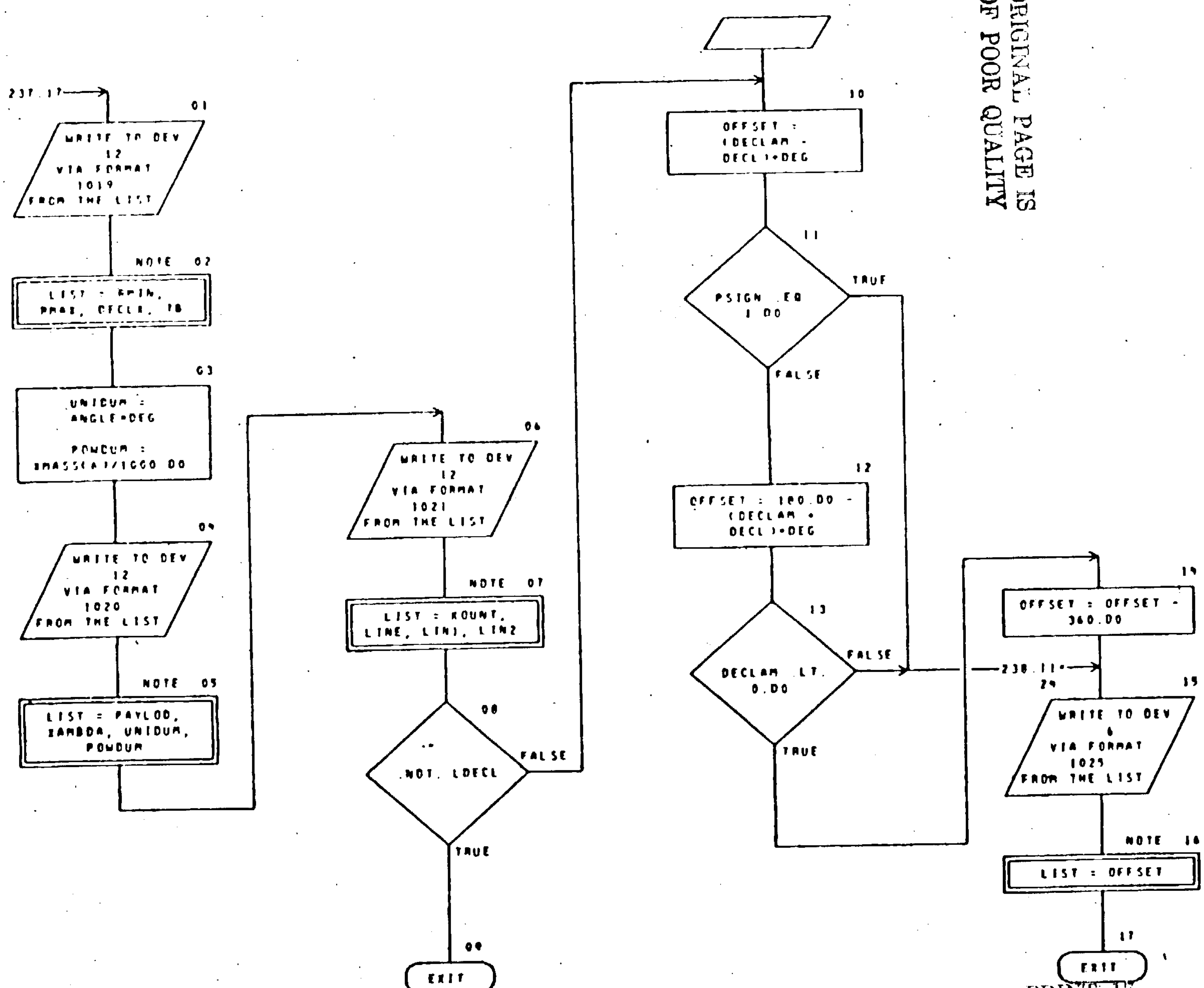


CHART TITLE - SUBROUTINE PRINT

ORIGINAL PAGE IS
OF POOR QUALITY



PRINT-17

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL ABX,ABY,OJEX,YES,ODECL,FIXPOW,LDECL
      DIMENSION CCODE(5),JC(5),A(58)
      COMMON /REAL8/ PAYLOD,XMASS(7),RO1(155),ANGLE,RO2(18),DECL,RO4(4),
     RMAX,RMIN,RO5(128),PSIGN,RO9,DECLAM,ROP,
                   XAMBDA,DEG,RO3(2),CONTM,RO6(1037),TAU,RO7(446)
      COMMON /INTGR4/ IO1(4),LINE,IO2(10),
     NSET(5),IO8(17),NSPEC,IO7(15),
     KOUNT,IO4(12),MAJOR,IO5(7),NUM,IO9(254),NSTEP1,NSTEP2,I10(19),
     NEXT(50),IO6(600)
      COMMON /LOGIC4/ LO1(2),FIXPOW,LO4(4),ODECL,LO5(18),OJEX,LO2(15),
     LDECL,LO6(182),                        ABX(70),ABY(70),LO3(135)
      COMMON /ITERAT/ BX(5,70),BO1(280),CONV(70),BO2(70),DD(70),
     FXL(70),BO3(70)
      COMMON /EXTREM/ EO1(400),ONOFF(2,100),EO2(2120)
      DATA A(1),A(2),A(3),A(4),A11,A12,ALAST1,ALAST2 /8H(1H ,20H,
     8HSWITCH-C,8HOUNT HIS,6HTORY3X,6HI1,1H ,6HI2,1H ,7HI1,1H/),
     7HI2,1H/)/
      DATA ASKIP,ALL1,ALL2 /8H2H--/1H ,8H3HALL I2),8H3HALL I3)/
 1001 FORMAT(1H ,2H X1,1H=1PD19.12,2(3H, X1,1H=1PD19.12))
 1002 FORMAT(1H ,2H X1,1H=1PD19.12,3H, X1,1H=1PD19.12,
     2H,X12,1H=1PD19.12)
 1003 FORMAT(1H ,2H X1, 1H=1PD19.12,2(2H,X12,1H=1PD19.12))
 1004 FORMAT(1H ,2H X1,1H=1PD19.12)
 1005 FORMAT(1H ,1HX12,1H=1PD19.12)
```

CHART TITLE - NON-PROCEDURAL STATEMENTS


```
1006      FORMAT(1H ,2H X11,1H=1P1D19.12,3H, X11,1H=1P1D19.12)
1007      FORMAT(1H ,2H X11,1H=1P1D19.12,2H,X12,1H=1P1D19.12)
1008      FORMAT(1H ,1HX12,1H=1P1D19.12,2H,X12,1H=1P1D19.12)
15        FORMAT(1H )
1030      FORMAT(1H ,29HPROPULSION CORNER PROXIMITY =F9.6,
          19H,  MIN SIGMA =1PD9.2)
19        FORMAT(1H0,16,22H THRUST COMPUTE STEPS,16,20H COAST COMPUTE STEPS)
1019      FORMAT(1H ,6HRMIN =F6.3,8H, RMAX =F6.2,8H, DECL =F7.2,7H, TAU =
          F7.1)
1020      FORMAT(1H ,10HNET MASS =1PD11.4,13H, INHIBITOR =1PD9.3,
          9H, ANGLE =0PF7.1,9H, POWER =0PF5.1)
1021      FORMAT(1H ,4HCASEI3,22H COMPLETED AT LINE NO I5,
          1H,4X13HIBRING X, L=I4,1H,I4,1H))
1025      FORMAT(38H0LAUNCH ASYMPTOTE OFFSET FROM PRIMER =F9.3,9H DEGREES )
9990      FORMAT(20H DEPENDENT VARIABLES)
9999      FORMAT(1H ,1HX12,1H=1P1D19.12,2(2H,X12,1H=1P1D19.12))
9996      FORMAT(1H ,5(2X,I2,1X,1P1D10.3))
```

PRECEDING PAGE BLANK NOT FILMED

Name: PRINTR

Calling Argument: M1, M2, XLAMDA

Referenced Sub-programs: None

Referenced Commons: INTGR4, ITERAT, ITER2, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: FINISH, SOLAR, TRAJ

Discussion: Subroutine PRINTR produces a printout termed the Trajectory Summary. The printout is comprised of the trajectory counters, the current value of the iterator inhibitor, the values of the independent and resulting dependent variables of the boundary value problem, and selected mass and performance parameter values for the associated trajectory. The frequency at which this printout is produced is controlled by the input integer NPRINT. If NPRINT is odd, the summary will be printed for the final trajectory of the case; if NPRINT is within the ranges of 4-7 or 12-15, then the summary is printed on every nominal trajectory (i.e., trajectories for which a partial derivative matrix is evaluated); if NPRINT is input negative, then the summary is printed for every trajectory that is computed, including perturbation trajectories.

For the printout of the final trajectory only, the printout of independent variables includes all 70 parameters, regardless of the number actually flagged as independent variables. A complete set of 70 locations is also printed for the dependent variables although those not flagged will be printed zero. Beside each value printed will appear a title to assist the analyst in identifying each parameter. For all other occurrences of the printout, only those parameters actually flagged as independent and dependent variables are printed.

The mass and performance parameters printed in the trajectory summary includes a history of the thrust switch points, the reference power p_{ref}, the total propulsion system efficiency η, the electric propulsion system burn time τ, the energy integral J, the ratio of propulsion time to mission duration, the average thrust acceleration a_{ave}, and the mass components m_o, m_{ps}, m_p, m_t, m_s and m_{net}. The energy integral and average acceleration are evaluated

$$J = \frac{g c \nu_p}{1 - \nu_p} ,$$

$$a_{ave} = \frac{g}{\sqrt{1 - \nu_p}} ,$$

where g is the reference thrust acceleration, c is the jet exhaust speed and ν_p is the ratio of electric propulsion propellant mass to initial mass.

For the expanded printout of the final trajectory only, a measure of the progress made by the iterator in achieving convergence is printed. The measure is in the form of a number for each dependent variable; a positive number denotes improvement in the end condition relative to the first nominal trajectory of the case and a negative number indicates that the discrepancy between actual and desired values has gotten worse. The magnitude of the number represents the order of magnitude of change; i.e., a value of 2 would denote an improvement of two orders of magnitude. A value is printed only for those end conditions that were not satisfied to the specified tolerance on the first nominal trajectory or on the final trajectory. Thus, this information will not be printed when the case converges.

Messages and printouts: Examples of the three types of trajectory summaries are shown on the following pages. Additional examples are shown in the Sample Problems and Results section.

Typical Intermediate Trajectory Summary with Negative NPRINT

```
------------------------------------------------------------------------------------------------------------------------------------
PECIAL TRAJECTORY PRINTOUT -------- NOMINAL =  1 ------ SUBNOMINAL =  1 ------ PERTURBATION =  4 ------ INHIBITOR =  0.58208D-10
------------------------------------------------------------------------------------------------------------------------------------
                                                    INDEPENDENT PARAMETERS
 1,PRIM1(-8.3659292D-02)   2,PRIM2(-3.0356676D-01)   3,PRIM3(-5.9620656D 00)   4,PDOT1( 8.1817375D-01)   5,PDOT2( 7.3550534D-01)
 6,PDOT3(-3.5058457D-01)  10,DECLN(-2.3735621D 01)  11,ACCEL( 3.1613231D-04)  13,VINF1( 5.8993340D 03)
                                                     DEPENDENT PARAMETERS
 1,DELTA X(-2.58966D-08)   2,DELTA Y( 2.65432D-08)   3,DELTA Z( 2.66946D-08)   4,DELT XD(-5.99531D-08)   5,DELT YD( 8.72376D-08)
 6,DELT ZD( 1.21781D-07)  10,T.DECLN( 6.46632D-08)  11, POWER ( 2.03500D 01)  13,T,VINF1(-1.20727D-08)
THRUST SWITCHING TIMES (DAYS)        0.0     ON      81.078   OFF     101.873   ON     217.160   OFF     243.832   ON     395.905   OFF
  436.041   ON     569.127   OFF     622.357   ON     730.000   ON
                                                 ELECTRIC PROPULSION PARAMETERS
      POWER                EFFICIENCY            PROP TIME                 J              PROP TIME RATIO          AVE ACCEL
   20.3500000559          0.6300000000        585.1270828542        11.1757197214         0.8075234012          0.0004690719
                                                  MASS COMPONENT BREAKDOWN
      INITIAL              PROPULSION            PROPELLANT              TANKAGE             STRUCTURE              PAYLOAD
   2756.9198783488       620.2500017045       1504.6926966798         52.6642443838           0.0              579.3129355807
```

Typical Standard Intermediate Trajectory Summary

```
------------------------------------------------------------------------------------------------------------------------------------
---------- NOMINAL TRAJECTORY  1 (TOTAL   1) ---------- INHIBITOR IS   5.82080-11 ---------------------------------------------------
------------------------------------------------------------------------------------------------------------------------------------
                                                    INDEPENDENT PARAMETERS
 1,PRIM1(-1.8492690D-02)   2,PRIM2(-3.4445459D-01)   3,PRIM3(-5.2928947D 00)   4,PDOT1( 7.2838601D-01)   5,PDOT2( 6.4895583D-01)
 6,PDOT3(-3.2776418D-01)  10,DECLN(-3.8091507D 01)  11,ACCEL( 2.7905741D-04)  13,VINF1( 5.3286065D 03)
                                                     DEPENDENT PARAMETERS
 1,DELTA X( 1.36734D-02)   2,DELTA Y(-2.04257D-02)   3,DELTA Z( 2.52203D-02)   4,DELT XD( 1.62397D-02)   5,DELT YD(-2.40807D-02)
 6,DELT ZD( 2.57904D-02)  10,T.DECLN( 1.75024D-09)  11, POWER ( 2.03500D 01)  13,T,VINF1(-3.71296D-09)
THRUST SWITCHING TIMES (DAYS)        0.0     ON      80.374   OFF     103.276   ON     218.909   OFF     247.259   ON     399.777   OFF
  439.175   ON     567.999   OFF     618.895   ON     749.493   OFF     808.322   ON     912.500   ON
                                                 ELECTRIC PROPULSION PARAMETERS
      POWER                EFFICIENCY            PROP TIME                 J              PROP TIME RATIO          AVE ACCEL
   20.3499999990          0.6300000000        712.1252770929        11.3661826370         0.7804112626          0.0004309116
                                                  MASS COMPONENT BREAKDOWN
      INITIAL              PROPULSION            PROPELLANT              TANKAGE             STRUCTURE              PAYLOAD
   3123.1976935923       620.2499999709       1813.3820629241         63.4683722023           0.0              626.0972584950
```

CASE 1

ORIGINAL PAGE IS OF POOR QUALITY

ITERATOR SUMMARY

INDEPENDENT PARAMETERS

1,PRIM1(-3.2052556D-01)	2,PRIM2(-2.9421245D 00)	3,PRIM3(-8.6413568D 00)	4,PDOT1(-8.6893763D-01)	5,PDOT2(5.1797829D-01)
6,PDOT3(6.3346989D-01)	7,LMASS(1.0000000D 00)	8, LTAU(0.0)	9, (0.0)	10,DECLN(-4.6190001D 01)
11,ACCEL(4.1765371D-04)	12,V JET(2.8439285D 04)	13,VINF1(8.0023533D 03)	14,VINF2(0.0)	15,TIME1(-8.6000000D 02)
16,TIME2(-3.0000000D 01)	17,IPARK(0.0)	18,VELO1(0.0)	19,VELO2(0.0)	20,VELO3(0.0)
21,THET1(0.0)	22,THET2(0.0)	23,THET3(0.0)	24,THET4(0.0)	25,THET5(0.0)
26,THET6(0.0)	27,THET7(0.0)	28,THET8(0.0)	29,THET9(0.0)	30,LDEGR(0.0)
31, PHI1(0.0)	32, PHI2(0.0)	33, PHI3(0.0)	34, PHI4(0.0)	35, PHI5(0.0)
36, PHI6(0.0)	37, PHI7(0.0)	38, PHI8(0.0)	39, PHI9(0.0)	40,PHI10(0.0)
41,PR1-A(-1.1138379D 01)	42,PR2-A(-7.3151963D 00)	43,PR3-A(1.6553901D 01)	44,PD1-A(1.6431712D 00)	45,PD2-A(-1.1365816D 00)
46,PD3-A(-3.7547681D 00)	47,VINFA(0.0)	48,TIMEA(-6.0509628D 02)	49,KSAMP(0.0)	50,KDROP(0.0)
51,PR1-B(-3.3126336D 00)	52,PR2-B(-4.3738311D 00)	53,PR3-B(-8.0189925D 00)	54,PD1-B(9.6128850D-01)	55,PD2-B(-1.9623345D-02)
56,PD3-B(1.7155384D 00)	57,VINFB(0.0)	58,TIMEB(-3.0930274D 02)	59,KSAMP(0.0)	60,KDROP(0.0)
61,PR1-C(0.0)	62,PR2-C(0.0)	63,PR3-C(0.0)	64,PD1-C(0.0)	65,PD2-C(0.0)
66,PD3-C(0.0)	67,VINFC(0.0)	68,TIMEC(0.0)	69,KSAMP(0.0)	70,KDROP(0.0)

DEPENDENT PARAMETERS

1,DELTA X(-1.22842D-06)	2,DELTA Y(-1.28412D-06)	3,DELTA Z(-8.03121D-08)	4,DELT XD(-2.02452D-06)	5,DELT YD(-2.09683D-06)
6,DELT ZD(-4.18634D-07)	7, (0.0)	8, (0.0)	9, (0.0)	10,T,DECLN(3.02500D-06)
11, POWER (1.00000D 01)	12, (0.0)	13,T,VINF1(-1.63911D-06)	14, (0.0)	15, (0.0)
16, (0.0)	17, (0.0)	18, (0.0)	19, (0.0)	20, (0.0)
21, (0.0)	22, (0.0)	23, (0.0)	24, (0.0)	25, (0.0)
26, (0.0)	27, (0.0)	28, (0.0)	29, (0.0)	30, (0.0)
31, (0.0)	32, (0.0)	33, (0.0)	34, (0.0)	35, (0.0)
36, (0.0)	37, (0.0)	38, (0.0)	39, (0.0)	40, (0.0)
41,DEL X A(4.46709D-12)	42,DEL Y A(4.02174D-11)	43,DEL Z A(4.50025D-12)	44,T,PR1-A(-4.44857D-11)	45,T,PR2-A(-4.00673D-11)
46,T,PR3-A(-1.62785D-11)	47, (0.0)	48,T,TIMEA(5.50540D-11)	49, (0.0)	50, (0.0)
51,DEL X B(-6.40765D-11)	52,DEL Y B(1.16818D-10)	53,DEL Z B(1.77209D-11)	54,T,PR1-B(1.07260D-09)	55,T,PR2-B(-3.03183D-10)
56,T,PR3-B(-5.32259D-10)	57, (0.0)	58,T,TIMEB(1.32105D-10)	59, (0.0)	60, (0.0)
61, (0.0)	62, (0.0)	63, (0.0)	64, (0.0)	65, (0.0)
66, (0.0)	67, (0.0)	68, (0.0)	69, (0.0)	70, (0.0)

THRUST SWITCHING TIMES (DAYS) 0.0 ON 254.904 VISIT 550.697 VISIT 760.665 OFF 774.771 ON 830.000 ON

ELECTRIC PROPULSION PARAMETERS

POWER	EFFICIENCY	PROP TIME	J	PROP TIME RATIO	AVE ACCEL
10.0000000001	0.6100000000	815.8938406641	6.8170484362	0.9830046275	0.0005239737

MASS COMPONENT BREAKDOWN

INITIAL	PROPULSION	PROPELLANT	TANKAGE	STRUCTURE	PAYLOAD
1027.1285815052	302.8500000043	374.5414398642	37.4541439864	0.0	312.2829976503

SWITCH-COUNT HISTORY ALL 8

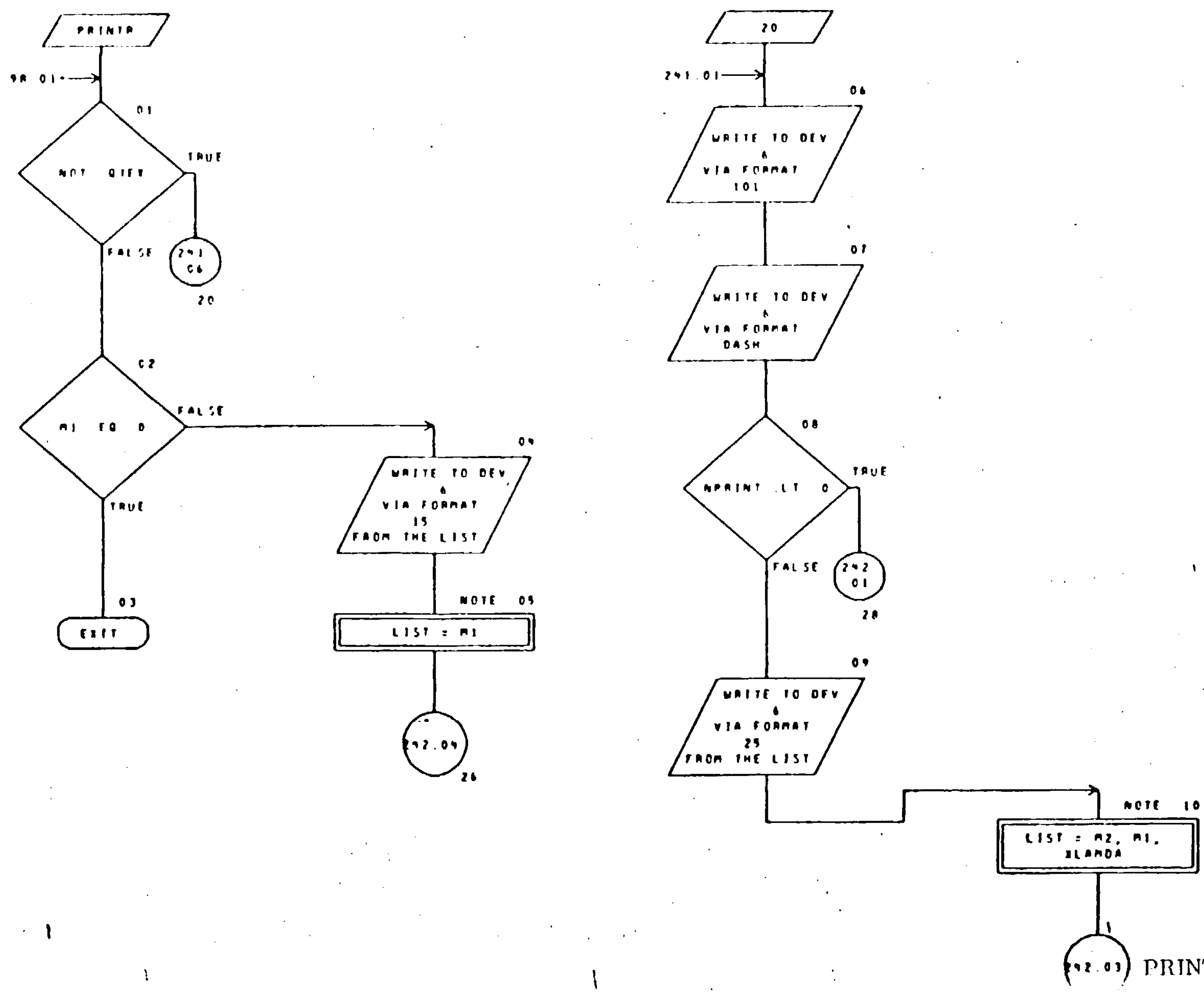
PRINTR EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
O(70)	U	ITERAT	Array of iterator independent-variables, in program internal units.
BY(3,70)	U	ITERAT	Iterator dependent variables array.
M1	UX		Counter of all nominal and trial trajectories. Does not include perturbation trajectories.
M2	UX		Counter of nominal trajectories.
ABX(70)	U	LOGIC4	Master array of iterator independent-variable indicators.
ABY(70)	U	LOGIC4	Master array of iterator dependent-variable indicators.
FXL(70)	U	ITERAT	Array of iterator dependent-variable values, in program internal units.
ISW(50)	SU	INTGR4	Stored values of indicator for thrust phase or coast phase.
NSW	SU	INTGR4	The total number of thrust switch-points along the current trajectory.
TAU	U	REAL8	Propulsion time, τ, in tau = AU/EMOS.
TSW(50)	SU	REAL8	Stored values of switch times, t_i, in tau.
CONX(70)	U	ITERAT	Array of print conversion factors for iterator independent-variables.
CONY(70)	U	ITERAT	Array of print conversion factors for iterator dependent-variables.
FXL1(70)	U	ITERAT	Array of dependent variable values corresponding to the initial nominal trajectory, eventually used in measuring the progress of the iteration sequence.
IOUT	U	INTGR4	Extra-ecliptic mission indicator.

PRINTR EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
NSET(5)	U	INTGR4	Iteration-sequence control array.
QJEX	U	LOGIC4	Detailed printout indicator.
QMAX(35)	U	ITER2	Array of upper allowable values for the iterator dependent-variables.
QMIN(35)	U	ITER2	Array of lower allowable values for the iterator dependent-variables.
MAJOR	U	INTGR4	Counter of the number of nominal ("major") trajectories, which have an associated set of neighboring trajectories, generated in the current iteration sequence.
MINOR	U	INTGR4	Perturbation trajectory counter for current iteration.
XMASS(7)	U	REAL8	General mass array and related parameters. XMASS(7) is total propulsion system efficiency, η.
LEGMAX	U	INTGR4	Total (maximum) number of trajectory-segments comprising the trajectory.
MAJORS	U	INTGR4	Total number of nominal plus search trajectories generated during the current case.
NPRINT	U	INTGR4	Printout amount selection indicator.
PAYLOD	U	REAL8	Net spacecraft mass, m_{net}, in kg.
XAMBDA	U	REAL8	The iterator's inhibitor, λ.
XLAMDA	UX		Iterator inhibitor, λ.

CHART TITLE - SUBROUTINE PRINTR(M1,M2,XLAMDA)



PRINTR-7

CHART TITLE - SUBROUTINE PRINTR(M1,M2,XLAMDA)

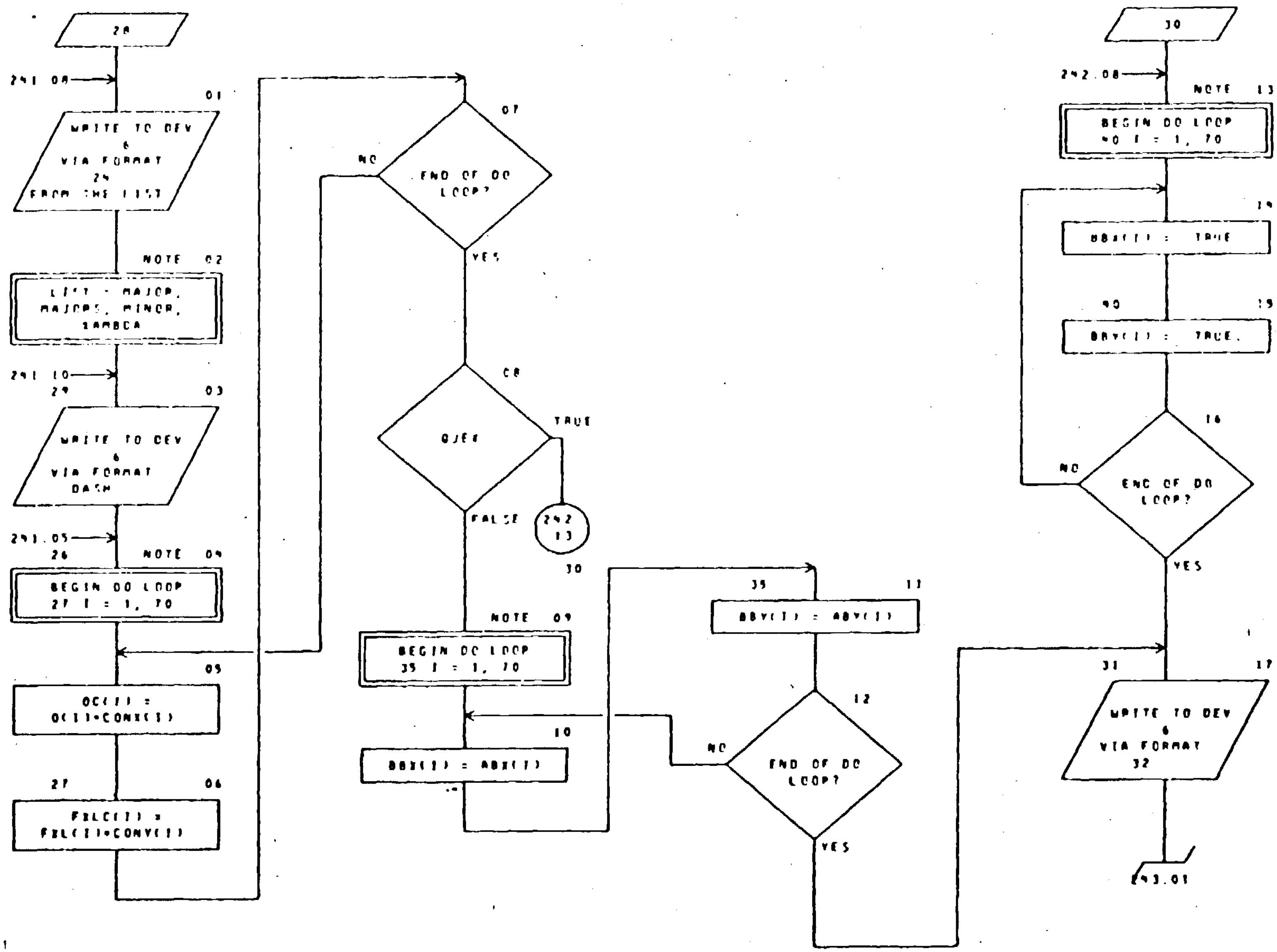
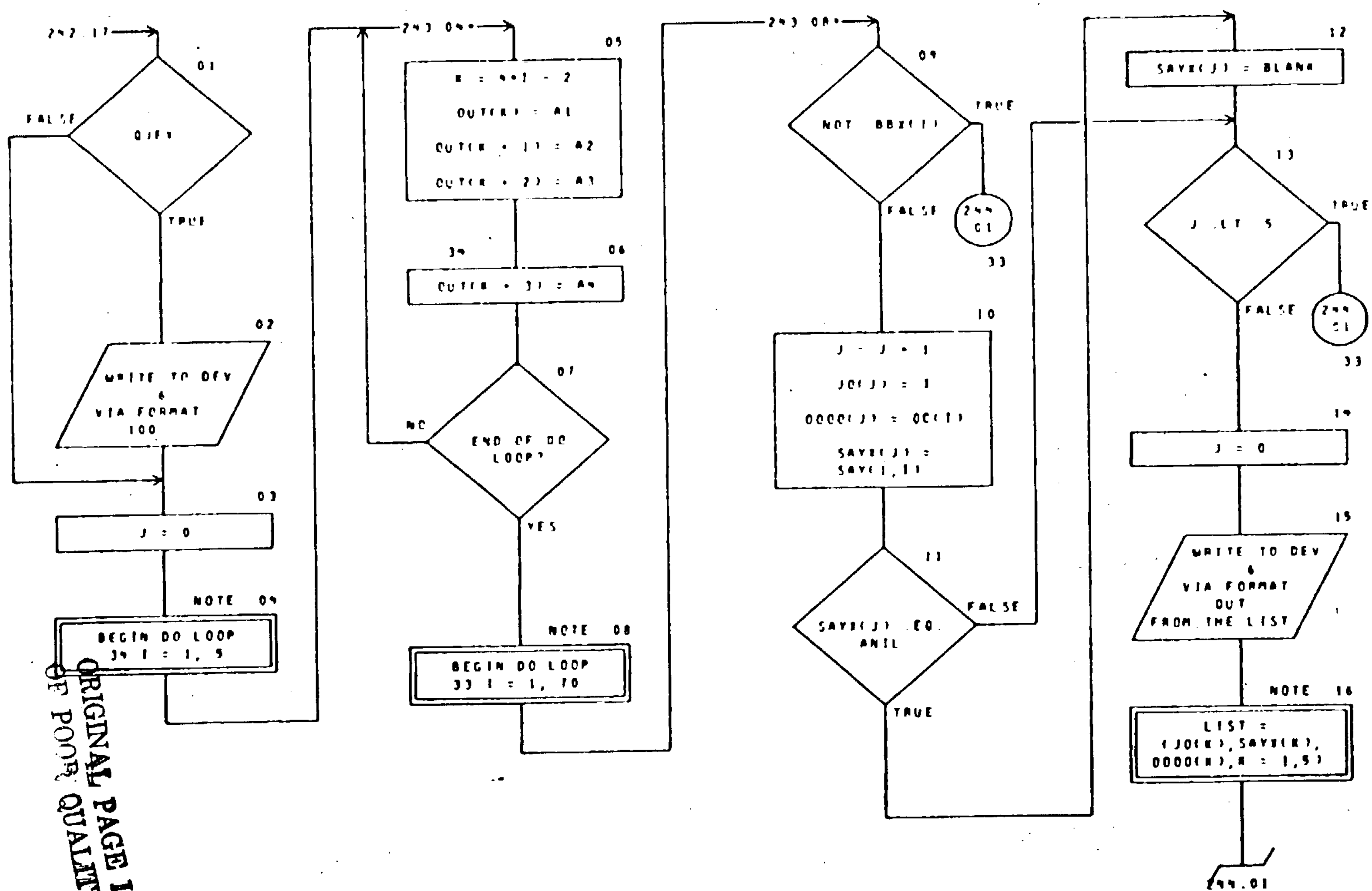


CHART TITLE - SUBROUTINE PRINTR(M1,M2,XLAMDA)



ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - SUBROUTINE PRINTR(M1,M2,XLAMDA)

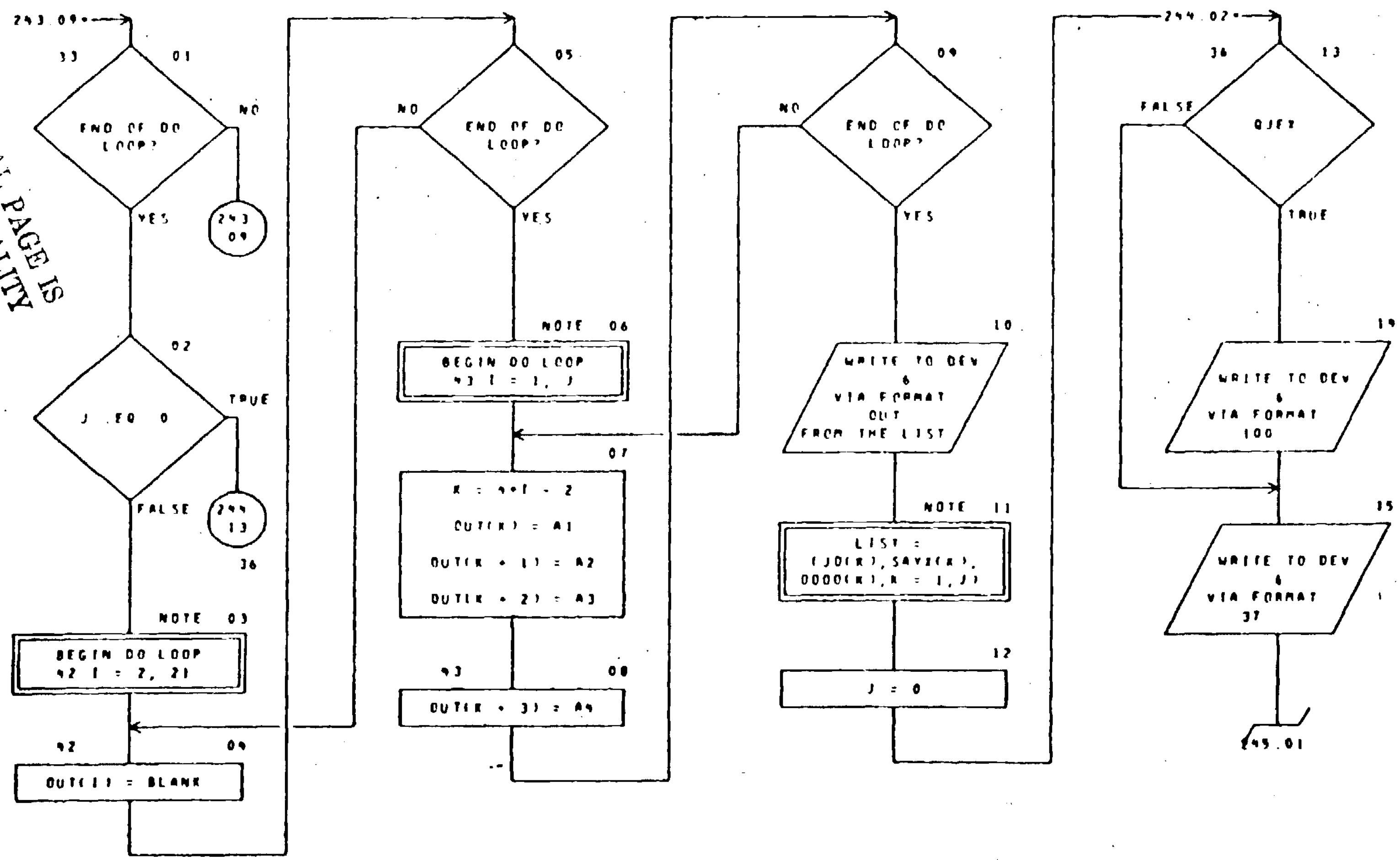


CHART TITLE - SUBROUTINE PRINTR(M1,M2,XLAND)

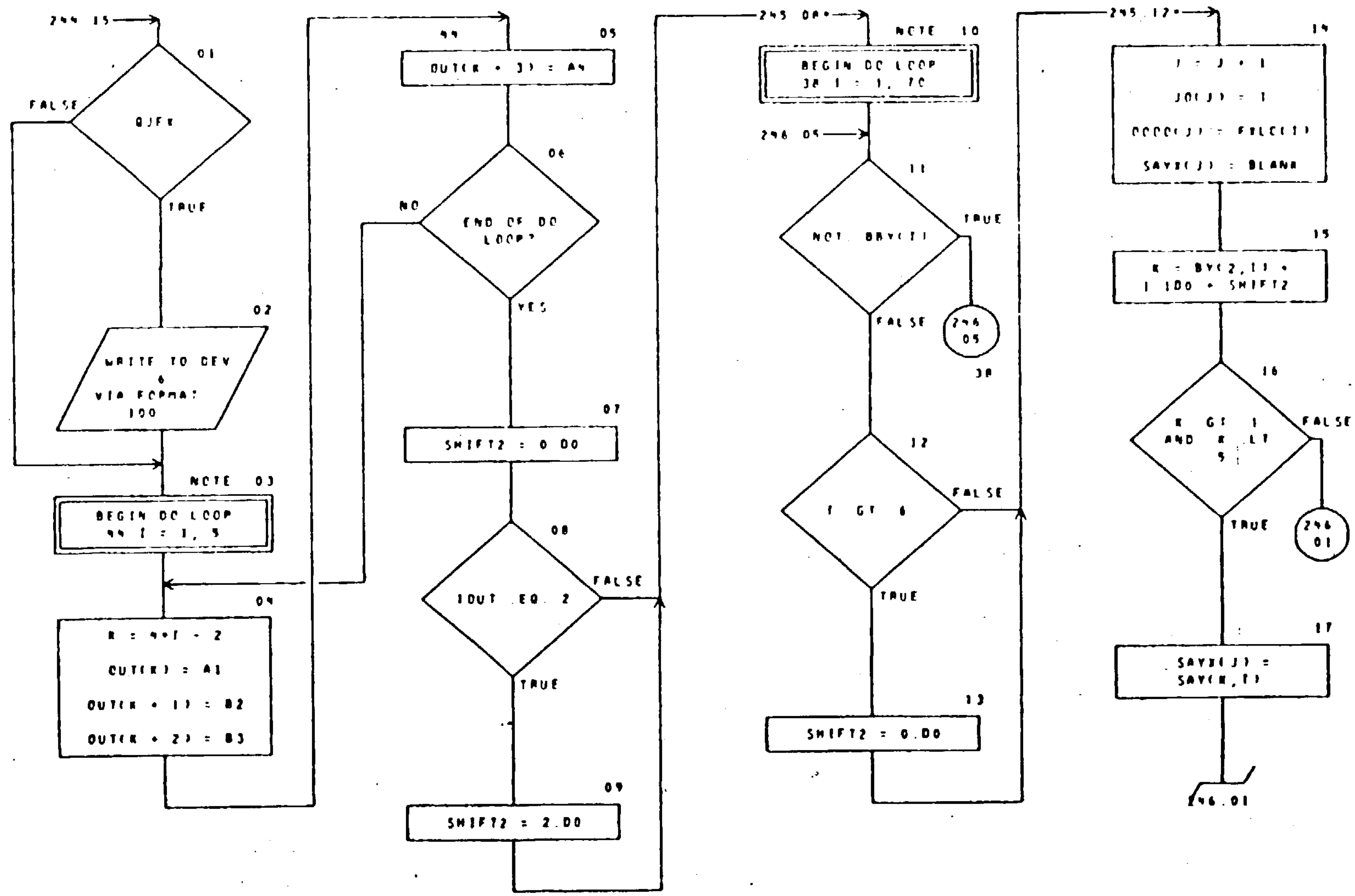


CHART TITLE - SUBROUTINE PRINTR(M1,M2,XLAMDA)

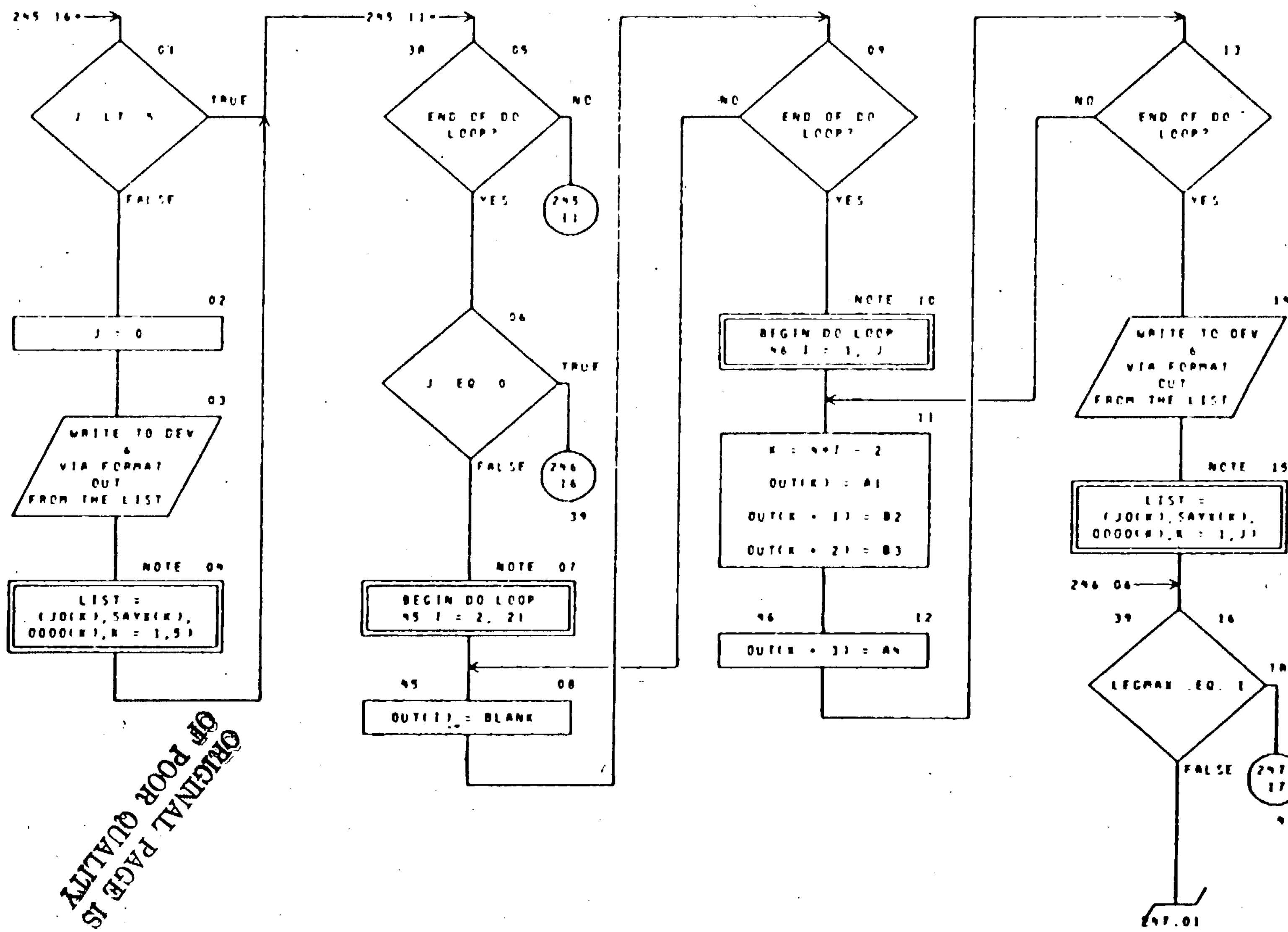


CHART TITLE - SUBROUTINE PRINTR(M1,M2,XLAMDA)

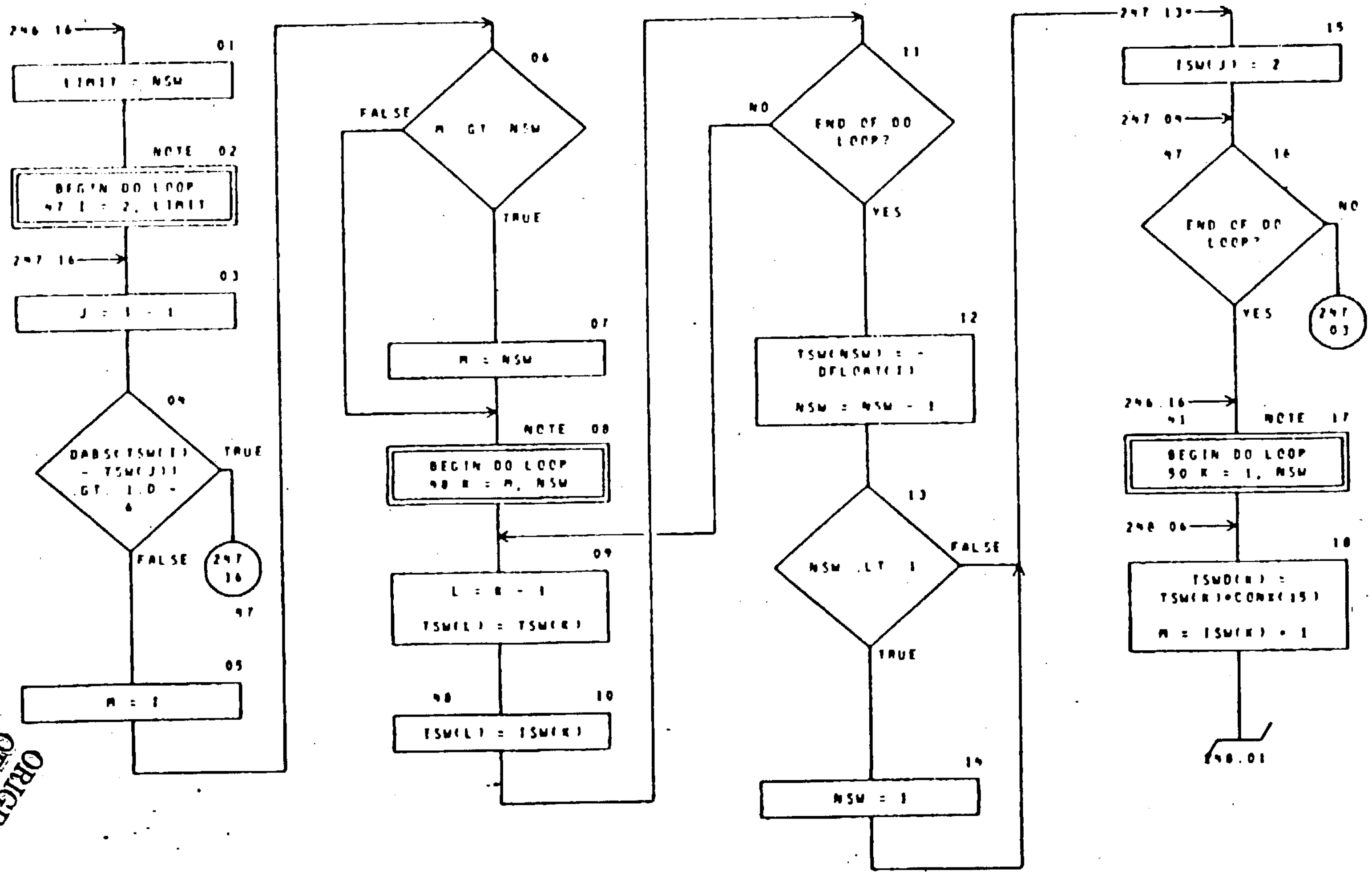


CHART TITLE - SUBROUTINE PRINTR(M1,M2,XLAMDA)

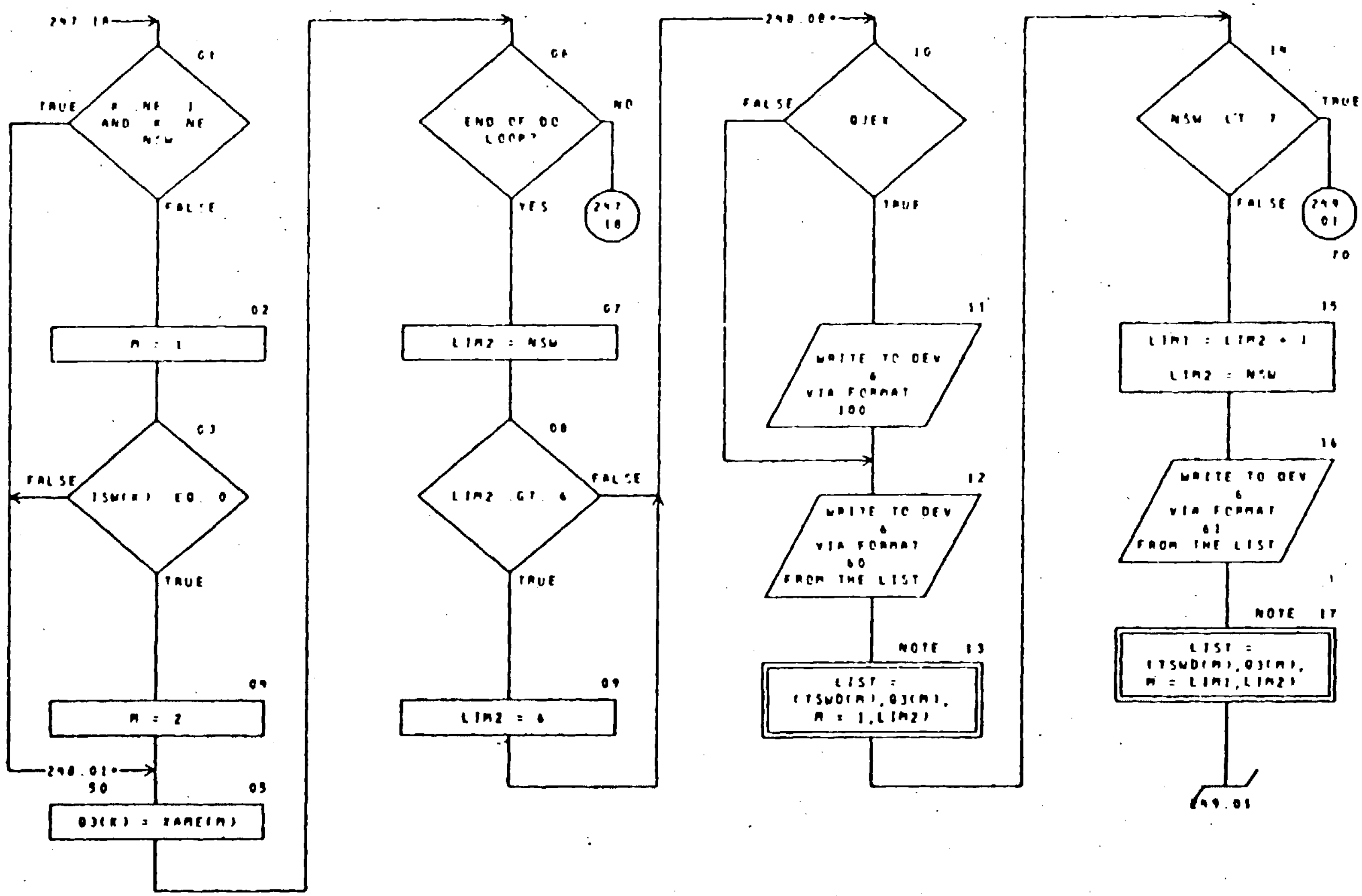
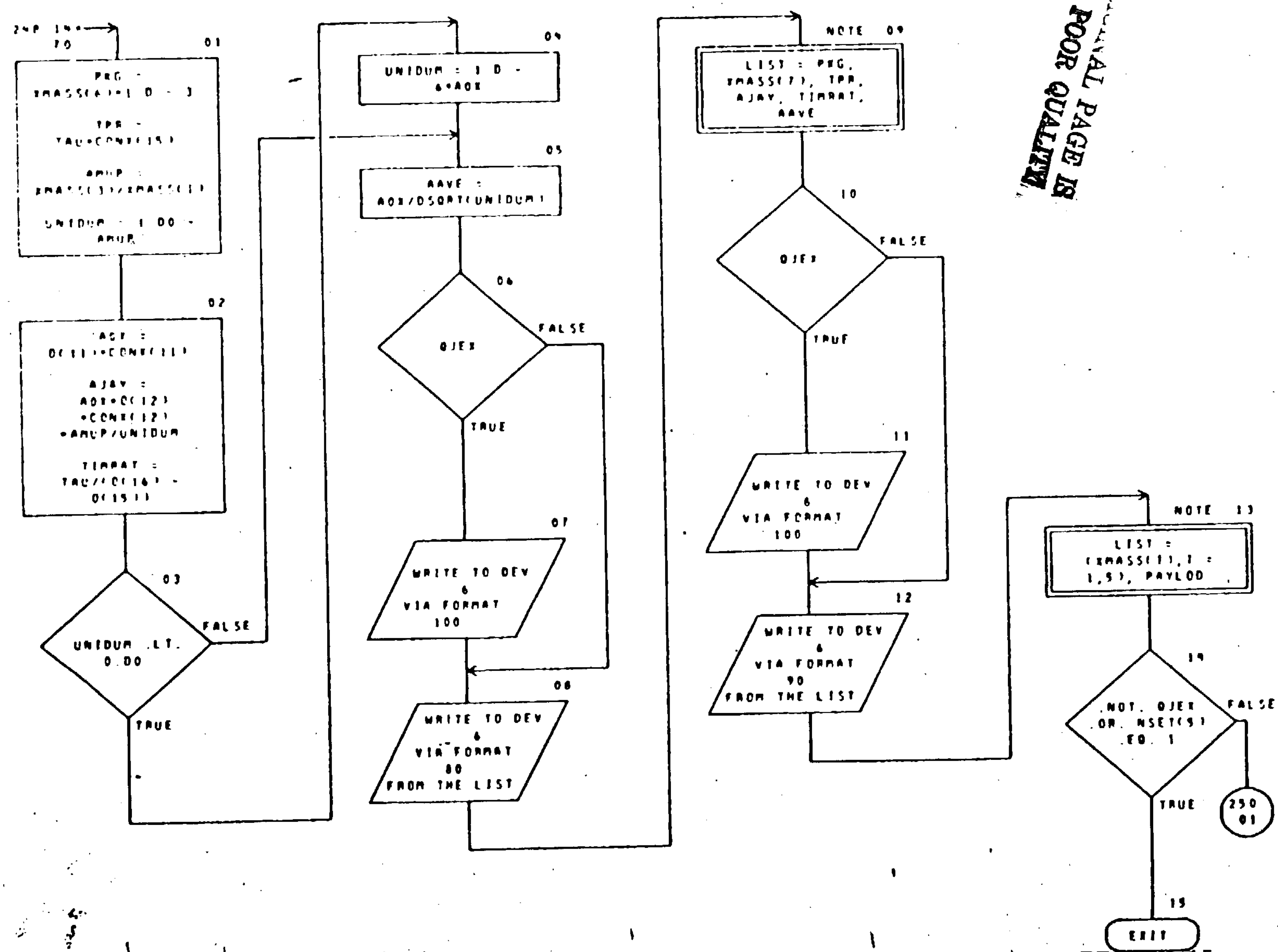


CHART TITLE - SUBROUTINE PRINTR(M1,M2,XLAMDA)



PRINTR-15

CHART TITLE - SUBROUTINE PRINT(M1,M2,XLAMDA)

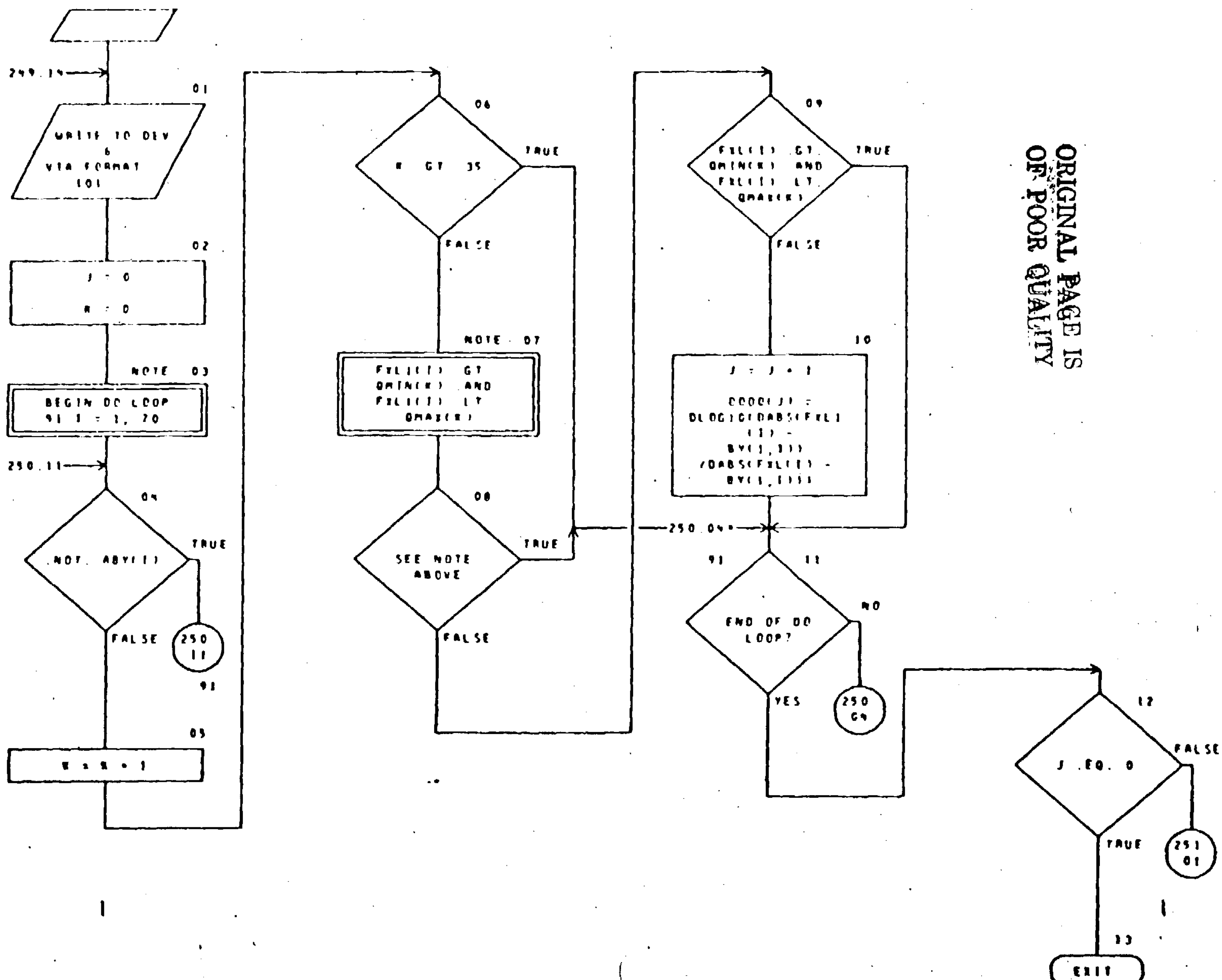


CHART TITLE - SUBROUTINE PRINTR(M1,M2,XLAMDA)

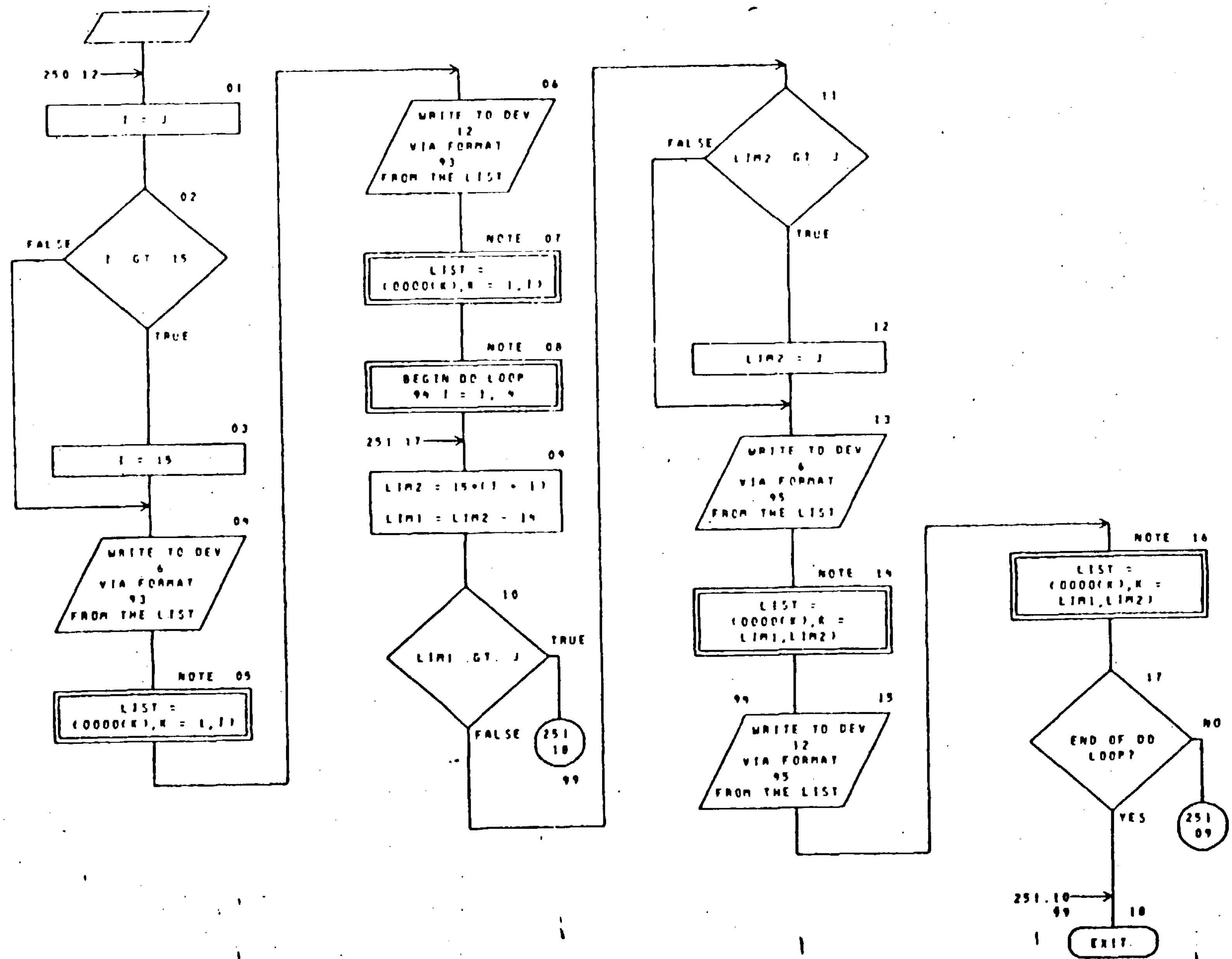


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL ABX,ABY,BBX,BBY,QIEX
DIMENSION TSWD(50),Q3(50),XAME(3),OC(70),FXLS(70),DDDD(70)
,JO(5),DASH(18),SAYEN,70),SAYX(5),OUT(22),BBY(70),BBY(70)
,ASAY(4,30),BSAY(4,40)
COMMON /REAL8/ PAYLOD,XMASS(7),PS1(303),XAMPDA,
PS2(388),TSW(50),PO3(603),TAU,PCN(646)
COMMON /INTGR4/ IO1(15),NSET(9),IO3(21),NPRIN,ICN(24),MAJOR,
MAJORS,MINOR,IOS(4),IOUT,NSW,ITW(50),IO2(2),LEGMAX,ICA(872)
COMMON /LOGIC4/ LO1(26),QIEX,LO2(198),ABX(70),ABY(70),LO3(135)
COMMON /ITERAT/ BO1(350),BY(3,75),CONX(70),CONY(70),D(70),
BO2(70),FXL(70),FXLI(70)
COMMON /ITEP2/ PO1(140),QMIN(35),QMAX(35),PS2(1320)
DATA XAME /4H  ON,5H  OFF,6H VISIT/
DATA DASH /8H(1H 130H,16*8H--------,3H--)/
DATA OUT(1),OUT(22),BLANK,A1,A2,A3,A4,B2,B3,ANIL /
4H(1H ,1H),1H ,8H,1H 12,1,8HH,A5,1H(,8H1PD14 7,,4H2H) ,
8HH,A7,1H(,8H1PD12 5,,3HNIL/
DATA ASAY/5HPRIM1,7HDELTA X,7H RADIUS,7HSEMI AX,5HPRIM2,7HDELTA Y,
7HT,ANGLE,
7H ECCENT,5HPRIM3,7HDELTA Z,3HNIL,7H INCLIN,5HPDOT1,7HDELT XD,
7HT,PRIM1,6HT,NODE,5HPDOT2,7HDELT YD,7HT,PRIM2,7HT,ARG P,5HPDOT3,
7HDELT ZD,7HT,PRIM3,7HT,TAU A,5HLMASS,6HMRATIO,6H LMASS,7HNETMASS,
5H LTAU,6H  LTAU,6H   TAU,5*3HNIL,5HDECLN,7HT,DECLN,6H DECLN,
3HNIL,5HACCEL,7HT,ACCEL,6H ACCEL,
```

ORIGINAL PAGE IS
OF POOR QUALITY

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
6H POWER,5HV JET,7HT,V JET,6H V JET,3HNIL,5HVINF1,7HT,VINF1,
6H VINF1,3HNIL,5HVINF2,7HT,VINF2,6H VINF2,7HINCLINA,5HTIME1,
7HT,TIME1,6H TIME1,3HNIL,5HTIME2,7HT,TIME2,6H TIME2,6H  TIME,
5HIPARK,7HT,IPARK,6H IPARK,3HNIL,5HVELO1,7HT,VELO1,6H VELO1,
3HNIL,5HVELO2,7HT,VELO2,6H VELO2,3HNIL,5HVELO3,7HT,VELO3,6H VELO3,
3HNIL,5HTHET1,7HT,THET1,6HTHETA1,
3HNIL,5HTHET2,7HT,THET2,6HTHETA2,3HNIL,5HTHET3,7HT,THET3,6HTHETA3,
3HNIL,5HTHET4,7HT,THET4,6HTHETA4,3HNIL,5HTHET5,7HT,THET5,6HTHETA5,
3HNIL,5HTHET6,7HT,THET6,6HTHETA6,3HNIL,5HTHET7,7HT,THET7,6HTHETA7,
3HNIL,5HTHET8,7HT,THET8,6HTHETA8,3HNIL,5HTHET9,7HT,THET9,6HTHETA9,
3HNIL,5HLDEGR,7HT,DEGRD,6HDEG TR,3HNIL/
DATA BSAY /5H PHI1,6HT,PHI1,6H  PHI1,
3HNIL,5H PHI2,6HT,PHI2,6H  PHI2,3HNIL,5H PHI3,6HT,PHI3,6H  PHI3,
3HNIL,5H PHI4,6HT,PHI4,6H  PHI4,3HNIL,5H PHI5,6HT,PHI5,6H  PHI5,
3HNIL,5H PHI6,6HT,PHI6,6H  PHI6,3HNIL,5H PHI7,6HT,PHI7,6H  PHI7,
3HNIL,5H PHI8,6HT,PHI8,6H  PHI8,3HNIL,5H PHI9,6HT,PHI9,6H  PHI9,
3HNIL,5HPHI10,7HT,PHI10,6H PHI10,3HNIL,5HPR1-A,7HDEL X A,2*1H ,
5HPR2-A,7HDEL Y A,2*1H ,5HPR3-A,7HDEL Z A,2*1H ,
5HPD1-A,7HDELXD A,7HT,PR1-A,1H ,5HPD2-A,7HDELYD A,7HT,PR2-A,1H ,
5HPD3-A,7HDELZD A,7HT,PR3-A,1H ,5HVINFA,7HT,VINFA,6H VINFA,1H ,
5HTIMEA,7HT,TIMEA,6H TIMEA,1H ,5HRSAMP,7HMSAMP-A,2*1H ,5HKDROP,
7HMDROP-A,2*1H ,5HPR1-B,7HDEL X B,2*1H ,5HPR2-B,7HDEL Y B,2*1H ,
5HPR3-B,7HDEL Z B,2*1H ,5HPD1-B,7HDELXD B,7HT,PR1-B,1H ,
5HPD2-B,7HDELYD B,7HT,PR2-B,1H ,5HPD3-B,7HDELZD B,7HT,PR3-B,1H ,
5HVINFB,7HT,VINFB,6H VINFB,1H ,5HTIMEB,7HT,TIMEB,6H TIMEB,1H ,
```

```
         5HRSAMP,7HRSAMP-B,2*1H ,5HRDROP,7HRDROP-B,2*1H ,
         5HPR1-C,7HDEL X C,2*1H ,5HPR2-C,7HDEL Y C,2*1H ,
         5HPR3-C,7HDEL Z C,2*1H ,5HPD1-C,7HDELXD C,7HT,PR1-C,1H ,
         5HPD2-C,7HDELYD C,7HT,PR2-C,1H ,5HPD3-C,7HDELZD C,7HT,PR3-C,1H ,
         5HVINFC,7HT,VINFC,6H VINFC,1H ,5HTIMEC,7HT,TIMEC,6H TIMEC,1H ,
         5HRSAMP,7HRSAMP-C,2*1H ,5HRDROP,7HRDROP-C,2*1H /
         EQUIVALENCE (ASAVE(1,1),SAVE(1,1)),(BSAVE(1,1),SAVE(1,31))
15       FORMAT(1H1,4HCASEI3,45X16HITERATOR SUMMARY//1H )
25       FORMAT(1H ,29H---------- NOMINAL TRAJECTORYI3,7H (TOTALI4,
         26H) ---------- INHIBITOR IS 1PD12 4,
         49H ------------------------------------------------)
24       FORMAT(1H ,46HSPECIAL TRAJECTORY PRINTOUT -------- NOMINAL =
         I3,20H ------ SUBNOMINAL =I3,22H ------ PERTURBATION =I3,
         19H ------ INHIBITOR =D13 5)
32       FORMAT(1H ,49X22HINDEPENDENT PARAMETERS)
37       FORMAT(1H ,50X20HDEPENDENT PARAMETERS)
60       FORMAT(1H ,29HTHRUST SWITCHING TIMES (DAYS))X6(F10 3,A6))
61       FORMAT(1H ,8(F10.3,A6) /1H ,8(F10 3,A6) /1H ,8(F10 3,A6)/
         1H ,8(F10.3,  A6)/1H ,8(F10.3,  A6)/1H ,4(F10 3,  A6))
80       FORMAT(45X,'ELECTRIC PROPULSION PARAMETERS'/7X,'POWER',15X,
         'EFFICIENCY',10X,'PROP TIME',16X,'J',12X,'PROP TIME RATIO',8X,
         'AVE ACCEL'/0P6F20.10)
90       FORMAT(50X,'MASS COMPONENT BREAKDOWN'/7X,'INITIAL',13X,'PROPULSION
         '10X,'PROPELLANT',10X,'TANKAGE',13X,'STRUCTURE',11X,'PAYLOAD'/
         0P6F20.10)
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
105      FORMAT (1H )
101      FORMAT(1H0)
93       FORMAT(1H ,9HHUNTIS(F4.1,1H/))
95       FORMAT(1H ,4X15(F4.1,1H/))
```

Name: PRIOR

Calling Argument: GO

Referenced Sub-programs: None

Referenced Commons: INTGR4, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: ANSTEP, RKSTEP

Discussion: This subroutine is always called just prior to taking a computation step. When stepping forward along the trajectory, a saving operation is performed:

$$\beta_{saved} = \beta$$

$$x_{saved} = x$$

$$\dot{x}_{saved} = \dot{x}$$

$$\tau_{saved} = \tau$$

This results in the saving of the trajectory dependent variables x and their derivatives $\dot{x}$, the trajectory independent variable β, and the accumulated propulsion time τ, at the beginning of the current computation step. The derivatives $\dot{x}$ are time derivatives during coast phases and generalized derivatives during thrust phases.

When iterating to isolate a remarkable point along a trajectory, using the beginning of the current computation step as the saved referenced point, a restoration operation is performed:

$$\dot{x} = \dot{x}_{saved}$$

PRECEDING PAGE BLANK NOT FILMED

PRIOR EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
X(50)	U	REAL8	Array of trajectory dependent variables x corresponding to the current time along the trajectory.
GO	UX		Logical indicator for stepping forward; when true, perform saving operation; when false, perform restoration operation.
SX(50)	S	REAL8	Array of trajectory dependent variables x_{saved} corresponding to the beginning of the current computation step.
XD(50)	SU	REAL8	Array of the derivatives $\dot{x}$ of the trajectory dependent variables.
TAU	U	REAL8	Accumulated propulsion time, τ, in 1 AU normalized units (tau).
BETA	U	REAL8	Trajectory independent variable, β.
STAU	S	REAL8	Saved value of accumulated propulsion time, τ_{saved}, in tau.
ERODE	U	LOGIC4	Solar array radiation damage degradation indicator.
SBETA	S	REAL8	Saved value of trajectory independent variable, β_{saved}.
FIXTHR	U	LOGIC4	Indicator for fixed thrust cone angle.
NPHI20	U	INTGR4	Array location value corresponding to the current value of the fixed thrust cone angle. Currently, only one value for fixed thrust angle is allowed along a given trajectory, and NPHI20 is set equal to 21 in subroutine TAP.

PRIOR EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
QERODE	U	LOGIC4	Indicator for either the final (summary) trajectory of a given case or solar array radiation damage degradation.
TUDFLG	U	LOGIC4	Indicator for two-dimensional trajectory in xy plane only.

CHART TITLE - SUBROUTINE PRIOR(GO)

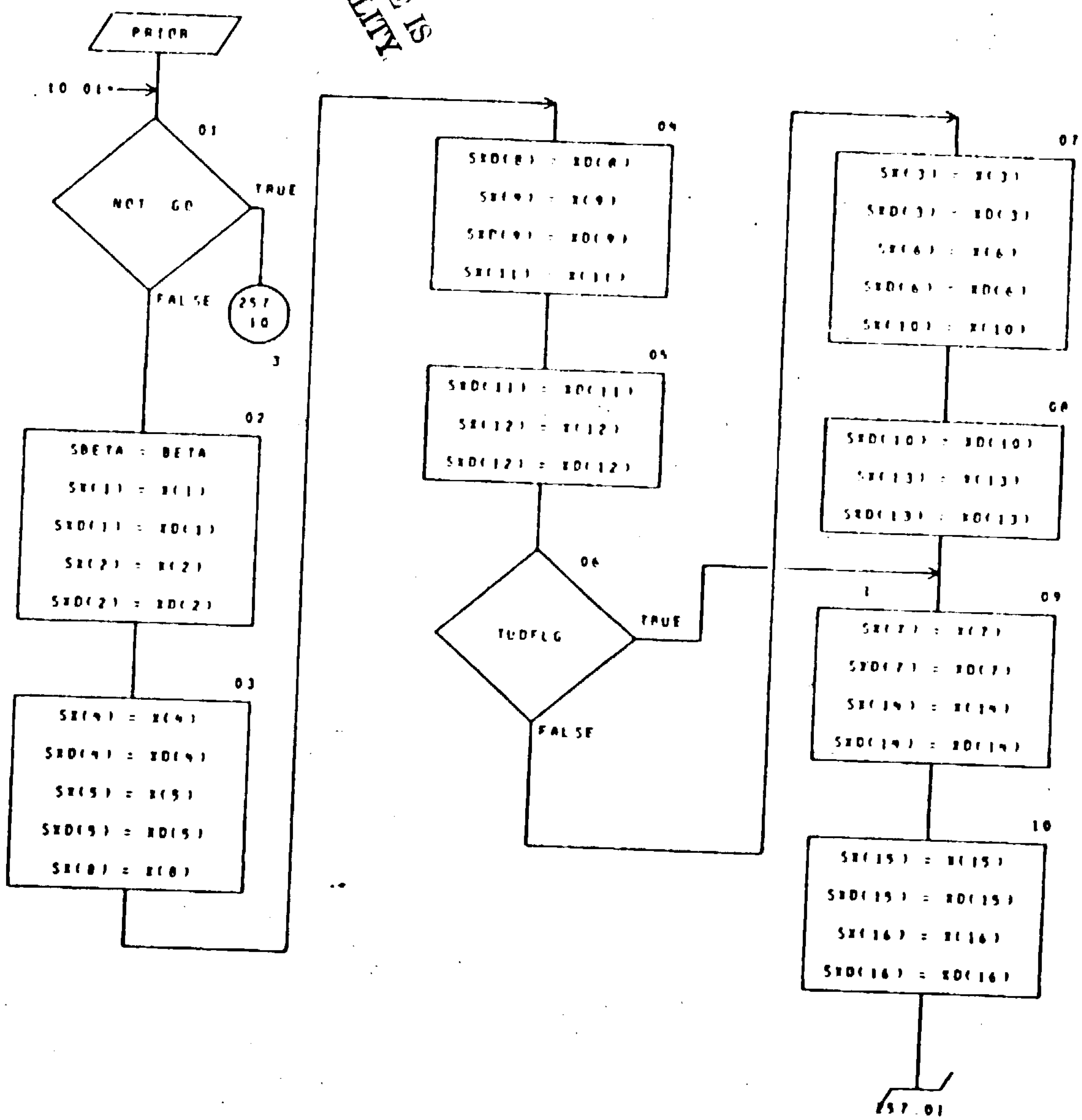
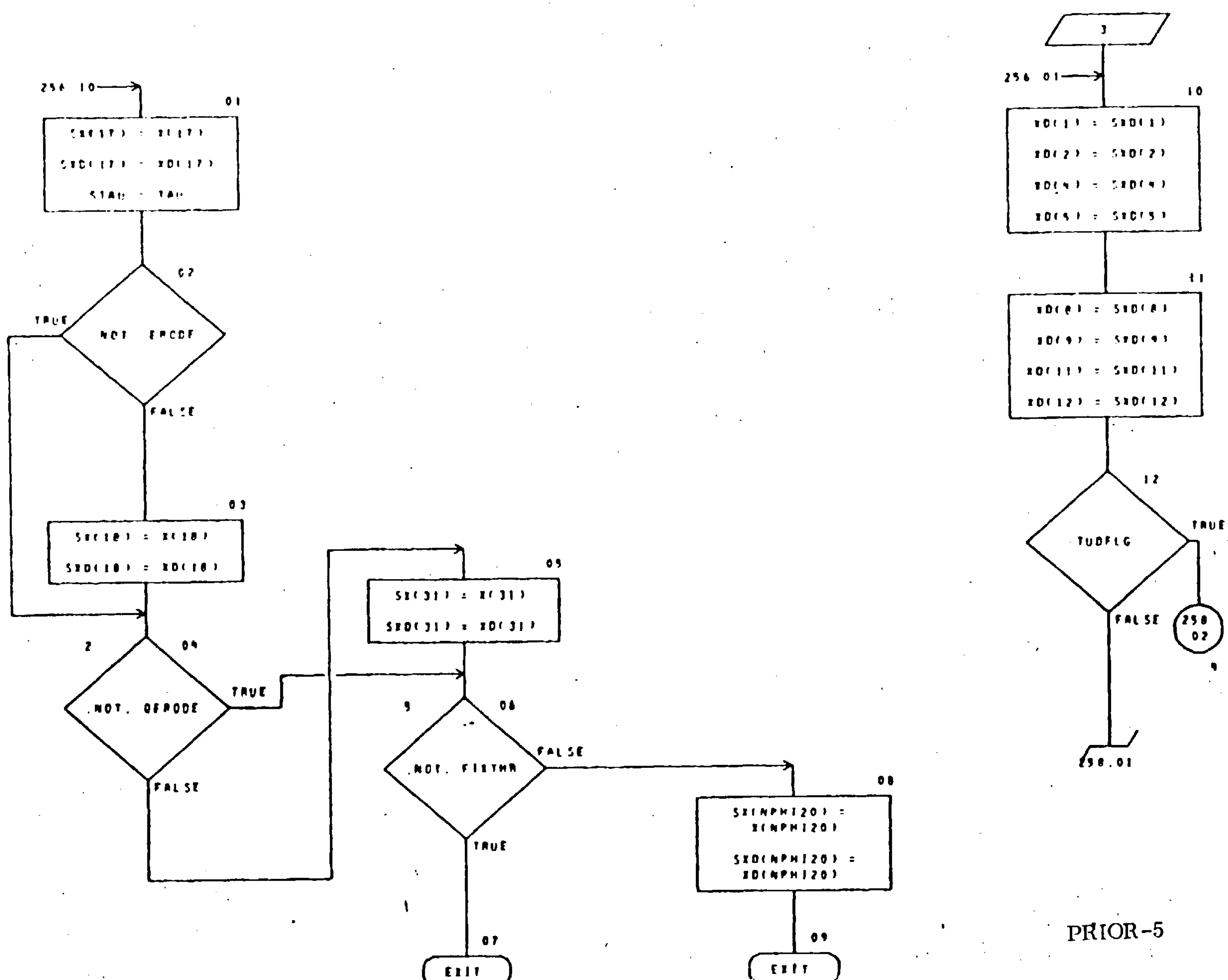


CHART TITLE - SUBROUTINE PRIOR(GO)



PRIOR-5

CHART TITLE - SUBROUTINE PRIOR(GO)

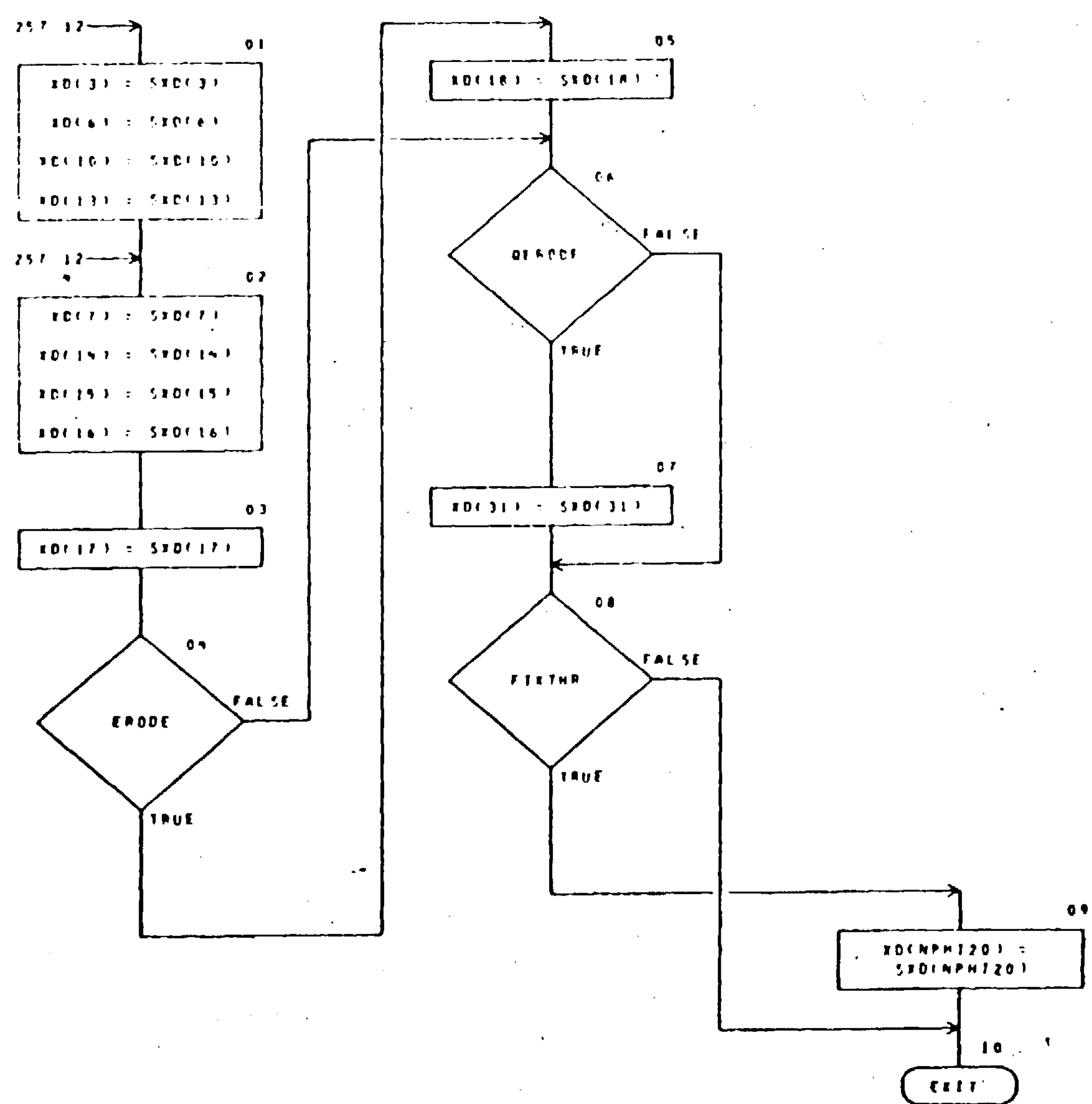


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL TUDFLG,FRODE,FIXTHR,GO,QFRODE
DIMENSION SXD(50)
COMMON /REAL8/ RO1(1200),SX(50),X(50), XD(50),SBETA,BETA,STAU,TAU,
RO3(696)
COMMON /INTGR4/ IO1(326),NPH12C,IO2(673)
COMMON /LOGIC4/ LO1(8),TUDFLG,LO2,FIXTHR,LO3(8),FRODE,LO4(12),
QFRODE,LO5(467)
```

Name:	PUNCH
Calling Argument:	PINT, TIMX, PRATIO, IFLAG, IPUNCH for PUNCH; None for READER
Referenced Sub-programs:	EFM
Referenced Commons:	INTGR4, ITERAT, LOGIC4, REAL8
Entry Points:	READER
Referencing Sub-programs:	FINISH, SPRINT for PUNCH; MAIN for READER

Discussion: Historically, this subroutine experienced heavy usage in the period 1969-1971 for the purpose of aiding in the publication of several reports by Horsewood and Mann involving electric propulsion performance analysis; the subroutine output, which originally consisted solely of punched cards containing information summarizing optimum electric propulsion trajectories, was employed, after editing, as input to the SC-4020 electronic plotter, the output of which was published. Since that time period, the subroutine has rarely been used.

The subroutine also has the capability of outputting step-by-step information at each computation point along a trajectory, in the form of either punched cards or magnetic tape (or any other device-specification in place of, but similar to, magnetic tape), to be used as input to the ASTEA Error Analysis Computer Program or for any other purpose. The description of the various quantities output is found in the External Variables Table below. The user should feel free to modify this basic output routine and tailor it to his particular requirements, if applicable.

Entry point READER performs the function of reading-in cards (from the card reader, unit 5) containing parameter values for starting a trajectory, i.e., containing values of the iterator independent variables; such cards are originally generated during a prior program run, by subroutine PUNCH, using the program input option MPUNCH = 1.

Messages and printouts: In entry point READER, if less than seven cards are input, for the purpose of starting the trajectory when such option is invoked, the error message is printed:

INPUT CARD # ___m___ IS ABSENT

where m indicates the first of possibly several absent cards needed to comprise the required seven cards.

PUNCH EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
F	U	REAL8	Reference thrust, in newtons.
R(2)	U	REAL8	Spacecraft solar distance (at prior and current computation point), in AU.
X(50)	U	REAL8	Array of trajectory integrated variables.
AE	U	REAL8	Desired final extra-ecliptic eccentricity.
AN	U	REAL8	Trajectory-integration exponent in regularization formula.
AR	U	REAL8	Desired final extra-ecliptic perihelion distance, in AU.
BI	U	REAL8	Efficiency coefficient b in equation for efficiency.
BX(5,70)	SU	ITERAT	Iterator independent variable array.
B1	U	REAL8	Launch vehicle coefficient b_1, in kg.
B2	U	REAL8	Launch vehicle coefficient b_2, in meters/second.
B3	U	REAL8	Launch vehicle coefficient b_3, in kg.
DI	U	REAL8	Efficiency coefficient d in equation for efficiency, in km/sec.

PUNCH EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
EI	U	REAL8	Efficiency coefficient e in equation for efficiency.
JT	U	INTGR4	Jettison indicator j_t for electric propulsion tankage prior to primary-target retro-maneuver.
OO(70)	U	ITERAT	Iterator independent variable array.
PT	U	REAL8	Power-function value at primary target intercept time, $(\gamma q)_t$.
X0(7)	U	REAL8	Spacecraft initial state vector, X_o, $\dot{X}_o$, M_o, where X_o is in AU, $\dot{X}_o$ is in EMOS, and $M_o = 1$.
AAI	U	REAL8	Desired final extra-ecliptic inclination, in degrees.
CNI	U	REAL8	Inclination to ecliptic of primary-target orbit, in degrees.
DEG	U	REAL8	Radians to degrees conversion factor.
ECI	U	REAL8	Eccentricity of primary-target orbit.
IRK	U	INTGR4	Numerical integration option indicator (currently not used).
JPP	U	INTGR4	Jettison indicator j_{ps} for electric propulsion system prior to primary-target retro-maneuver.
OMI	U	REAL8	Ascending node angle (with respect to vernal equinox) of primary-target orbit, in degrees.
RAP	U	REAL8	Apoapse distance of capture orbit about primary target, in planet radii.
SAI	U	REAL8	Semi-major axis of primary-target orbit, in AU.

PUNCH EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
SOI	U	REAL8	Argument of perihelion of primary-target orbit, in degrees.
TAU	U	REAL8	Electric propulsion system on-time, τ, in tau (= AU/EMOS).
TPI	U	REAL8	Time from reference date (which is specified by MYEAR, etc.) to perihelion passage, for the primary target, in days.
ANGD	U	REAL8	Travel angle, θ_t, in degrees.
ANG1	U	REAL8	Launch site latitude, in radians.
ANG2	U	REAL8	Maximum launch parking orbit inclination, in radians.
CSTR	U	REAL8	Electric propulsion system structural factor, k_s.
DECL	U	REAL8	Departure asymptote declination, δ, in radians.
DVEL	U	REAL8	Retro maneuver (at primary target) impulsive velocity increment, in meters/second.
FMAX	U	REAL8	Maximum thrust along trajectory, in newtons.
HOUR	U	REAL8	Hour-of-day of reference date.
IBAL	U	INTGR4	Ballistic trajectory option indicator.
IEND	UE	INTGR4	Indicator for endpoint of trajectory.
IROT	U	INTGR4	Initial primer vector rotation indicator.
MDAY	U	INTGR4	Day-of-month of reference date.
MODE	U	INTGR4	Power variation option selector.

PUNCH EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
MOPT	U	INTGR4	Ballistic trajectory option indicator.
PINT(17)	UX		Array of print-quantities computed in subroutine SPRINT.
PWRM	U	REAL8	Maximum power along trajectory, in kw.
RMAX	U	REAL8	Maximum solar distance along trajectory, in AU.
RMIN	U	REAL8	Minimum solar distance along trajectory, in AU.
RPER	U	REAL8	Periapse distance of capture orbit about primary target, in planet radii.
TIMX	UX		Time elapsed since the beginning of the trajectory, in days.
TRIP	U	REAL8	Flight time, from launch to the primary target, in days.
VORB	U	REAL8	Speed at periapse of capture orbit about primary target, in meters/second.
CONTM	U	REAL8	Tau to days conversion factor.
CTANK	U	REAL8	Electric propulsion system propellant tankage factor, k_t.
CTRET	U	REAL8	Retro tankage factor, k_{rt}, for retro maneuver at the primary target.
ERROR	U	LOGIC4	Program master error indicator.
IFLAG	UX		Indicator for thrusting or coasting.
ISTAR	U	INTGR4	Currently not used. (Indicator for star selection in cone and clock angle print-out. Canopus is always assumed.)

PUNCH EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
KOUNT	U	INTGR4	Case number of computer run.
MONTH	U	INTGR4	Month-of-year of reference date.
MOPT2	U	INTGR4	Launch planet number and ephemeris-option indicator.
MOPT3	UA	INTGR4	Planet-number of primary target.
MYEAR	U	INTGR4	Year of reference date.
NTAPE	U	INTGR4	Specifies the unit-number for the trajectory tape.
STATE(6)	U	REAL8	Array containing the Cartesian position and velocity components of the primary target, in AU and EMOS.
STEP1	U	REAL8	Thrust-phase computation step-size, Δu.
STEP2	U	REAL8	Coast-phase computation step-size, $\Delta\beta$.
TBURN	U	REAL8	Retro-maneuver burn time at primary target, in seconds.
THRET	U	REAL8	Retro-stage thrust in retro-maneuver at the primary target, in pounds.
VLOSS	U	REAL8	Velocity-loss, due to gravity, associated with the retro-maneuver at the primary target, in meters/second.
XINCL	U	REAL8	Parking orbit inclination, i, in degrees.
XMASS(7)	U	REAL8	Array of spacecraft weights, in kilograms, and parameters.
ALPHAA	U	REAL8	Specific mass of solar arrays, α_a, in kg/kw.
ALPHAT	U	REAL8	Specific mass of thruster subsystem, α_t, in kg/kw.

PUNCH EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
COMANG	U	REAL8	Communication angle to spacecraft at time of primary-target encounter, in degrees.
COMDIS	U	REAL8	Communication distance to spacecraft at time of primary-target encounter, in AU.
CONVRG	U	LOGIC4	Iteration-sequence convergence indicator.
DMRETR	U	REAL8	Retro engine mass, in kilograms.
EMUODD	U	REAL8	Gravitational constant of the primary target, in m^3/sec^2.
IPUNCH	UX		Indicator for punching trajectory-summary cards or outputting step-by-step trajectory information.
LAUNCH	U	INTGR4	Launch mode indicator.
MBOOST	U	INTGR4	Launch vehicle selector.
MPUNCH	U	INTGR4	Punched-card and trajectory-tape generation control indicator.
MTMASS	U	INTGR4	Mission-type selector pertaining to the primary target.
PAYLOD	U	REAL8	Net spacecraft mass, m_{net}, in kilograms.
POWFIX	U	REAL8	Launch-vehicle-independent trajectory option indicator, in which the value of POWFIX is the spacecraft's reference power in kilowatts.
PRATIO	UX		Power ratio, $q\gamma$.
QTMASS(5)	U	REAL8	Array of propulsion parameters, as described in RETINJ.
RADODD	U	REAL8	Radius of primary target, in meters.

PUNCH EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
SPIRET	U	REAL8	Retro-stage specific impulse pertaining to the retro-maneuver at the primary target, in seconds.
TDATEX	U	REAL8	Reference date expressed as Julian date less 2400000.
TDATE1	SU	REAL8	Launch date expressed as Julian date less 2400000.

CHART TITLE - SUBROUTINE PUNCH(PINT,TIMX,PRATIO,IFLAG,IPUNCH)

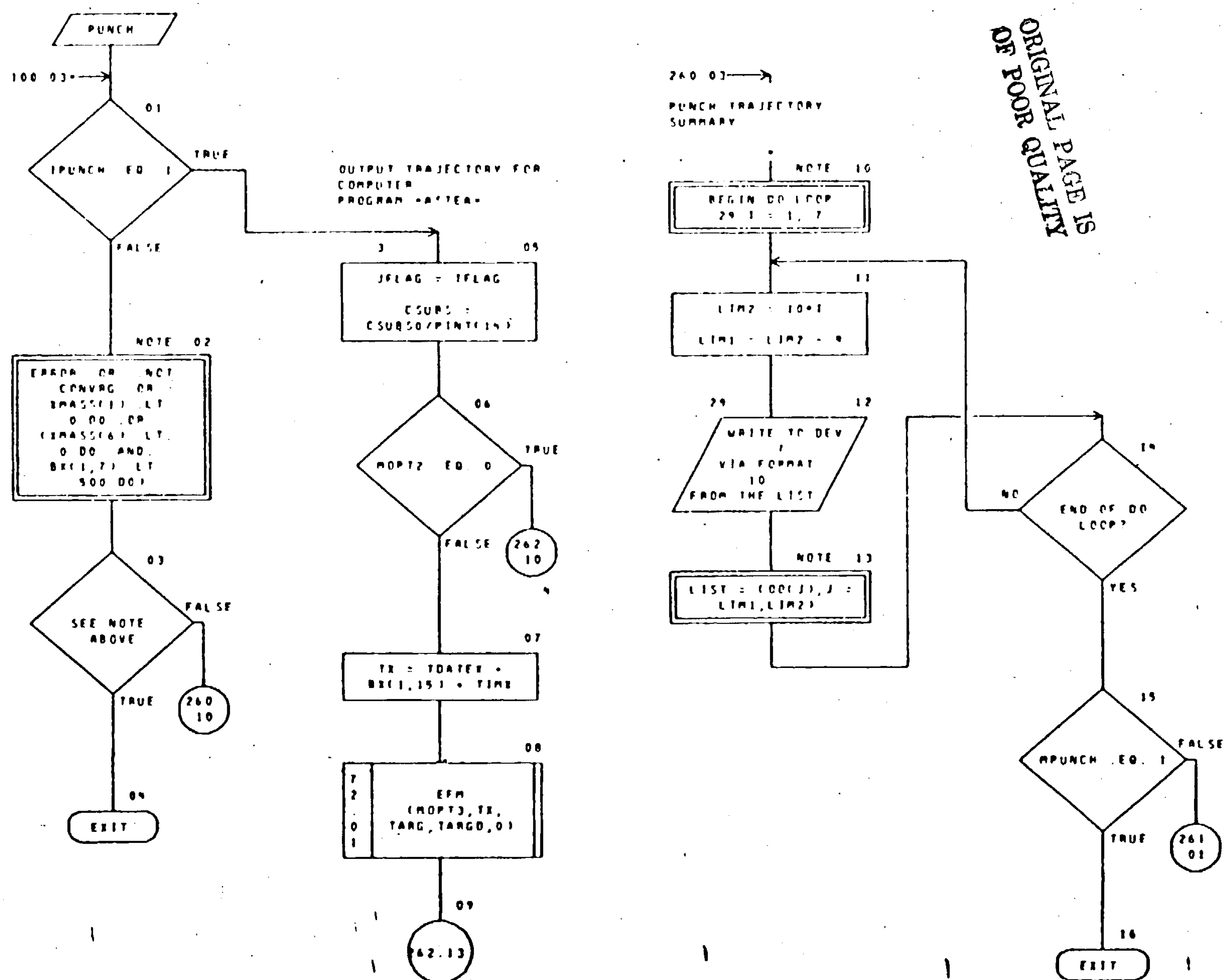


CHART TITLE - SUBROUTINE PUNCH(PINT,TIMX,PRATIO,IFLAG,IPUNCH)

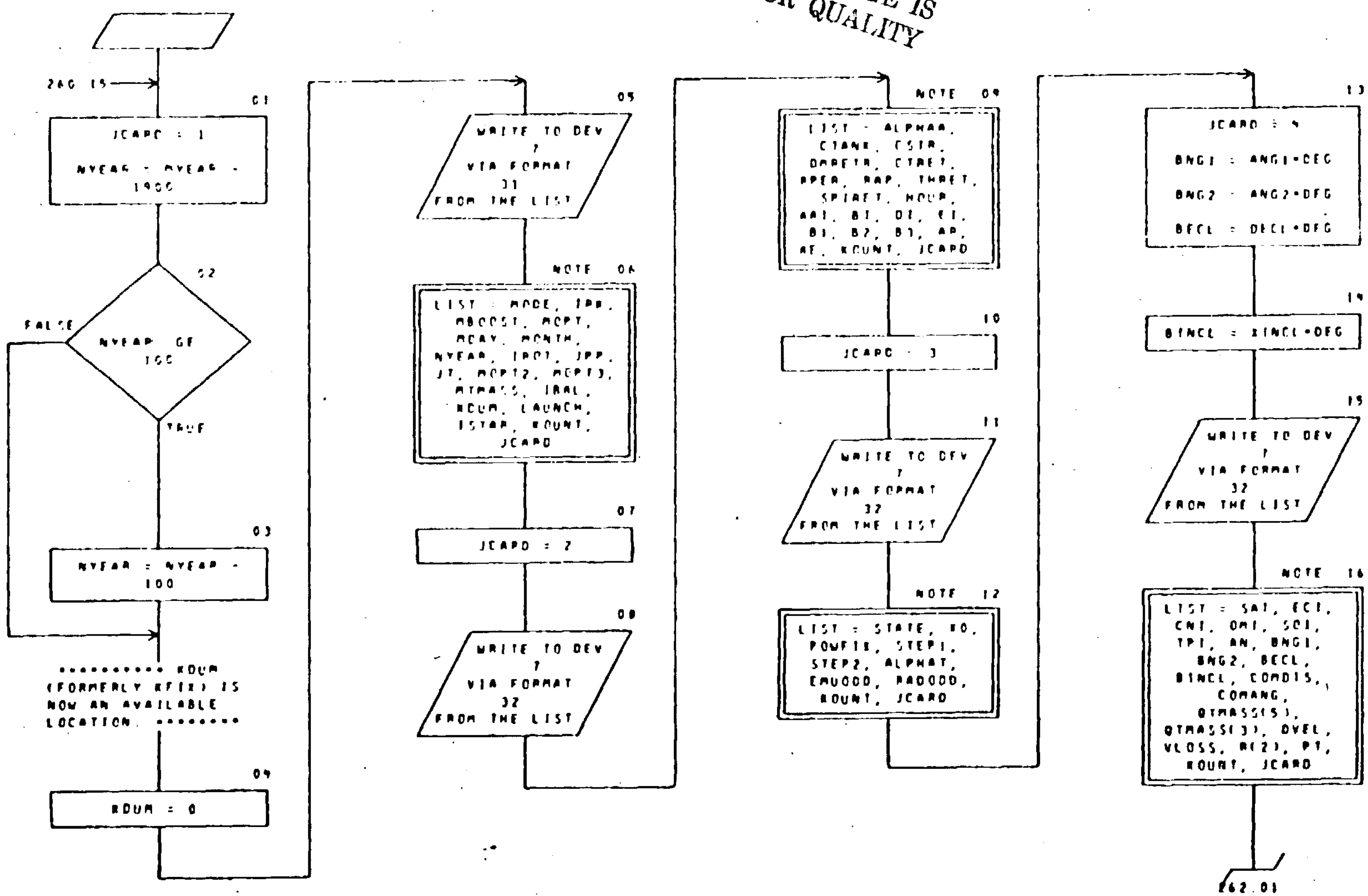
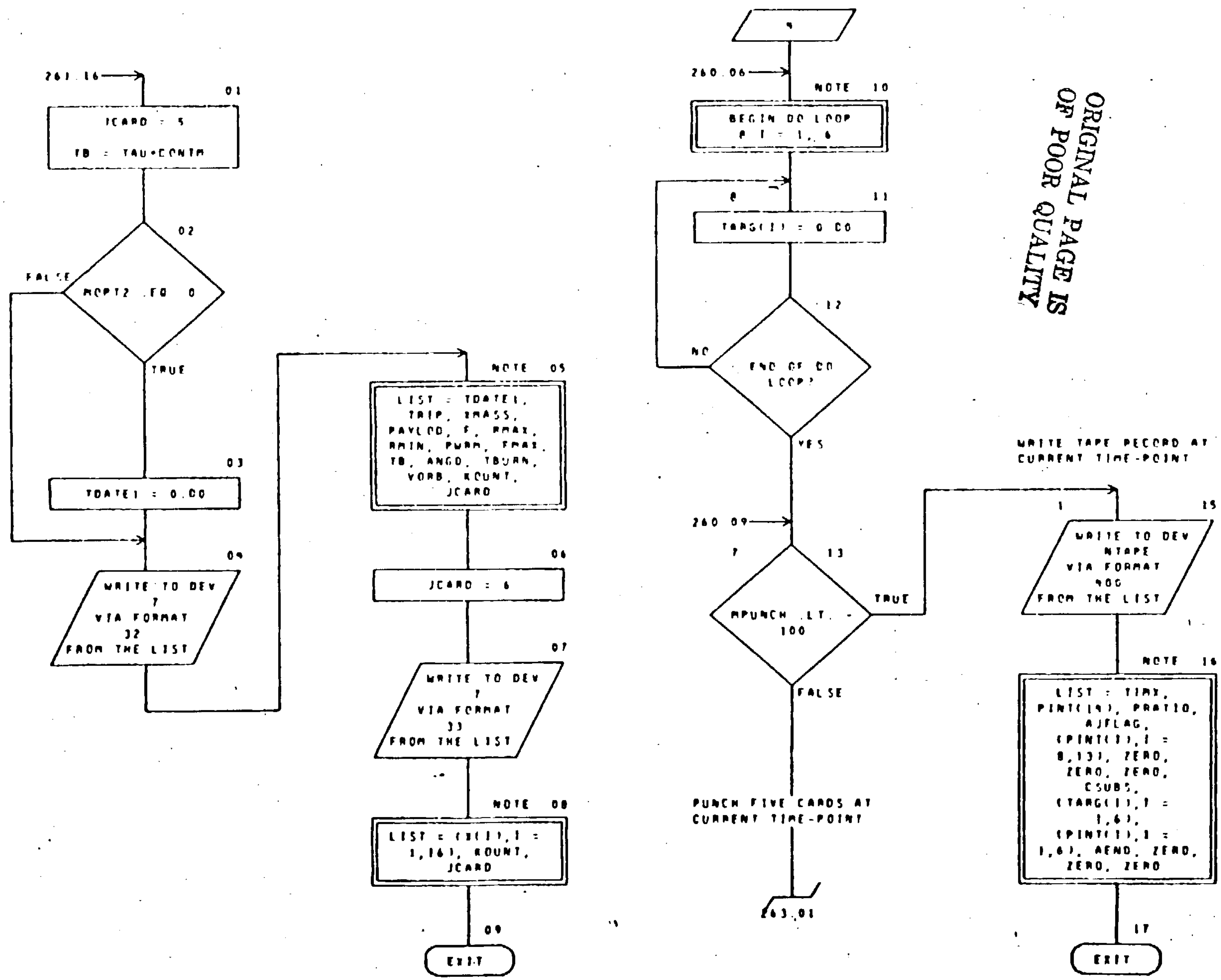
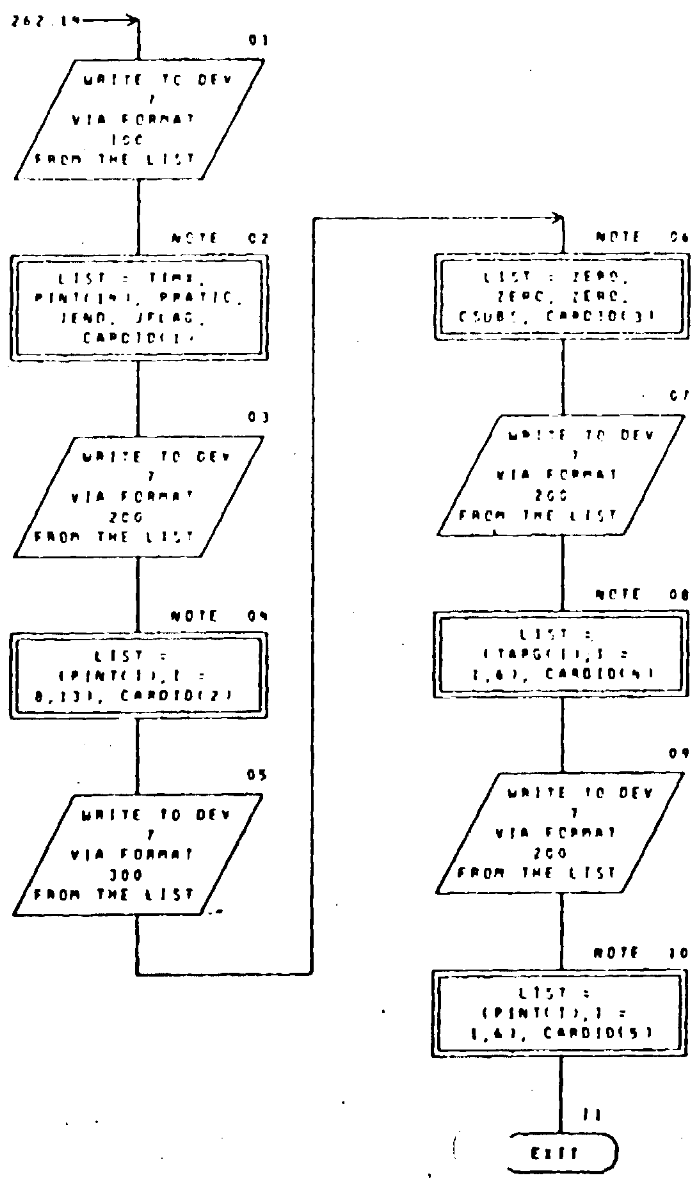


CHART TITLE - SUBROUTINE PUNCH(PINT,TIMX,PRATIO,IFLAG,IPUNCH)



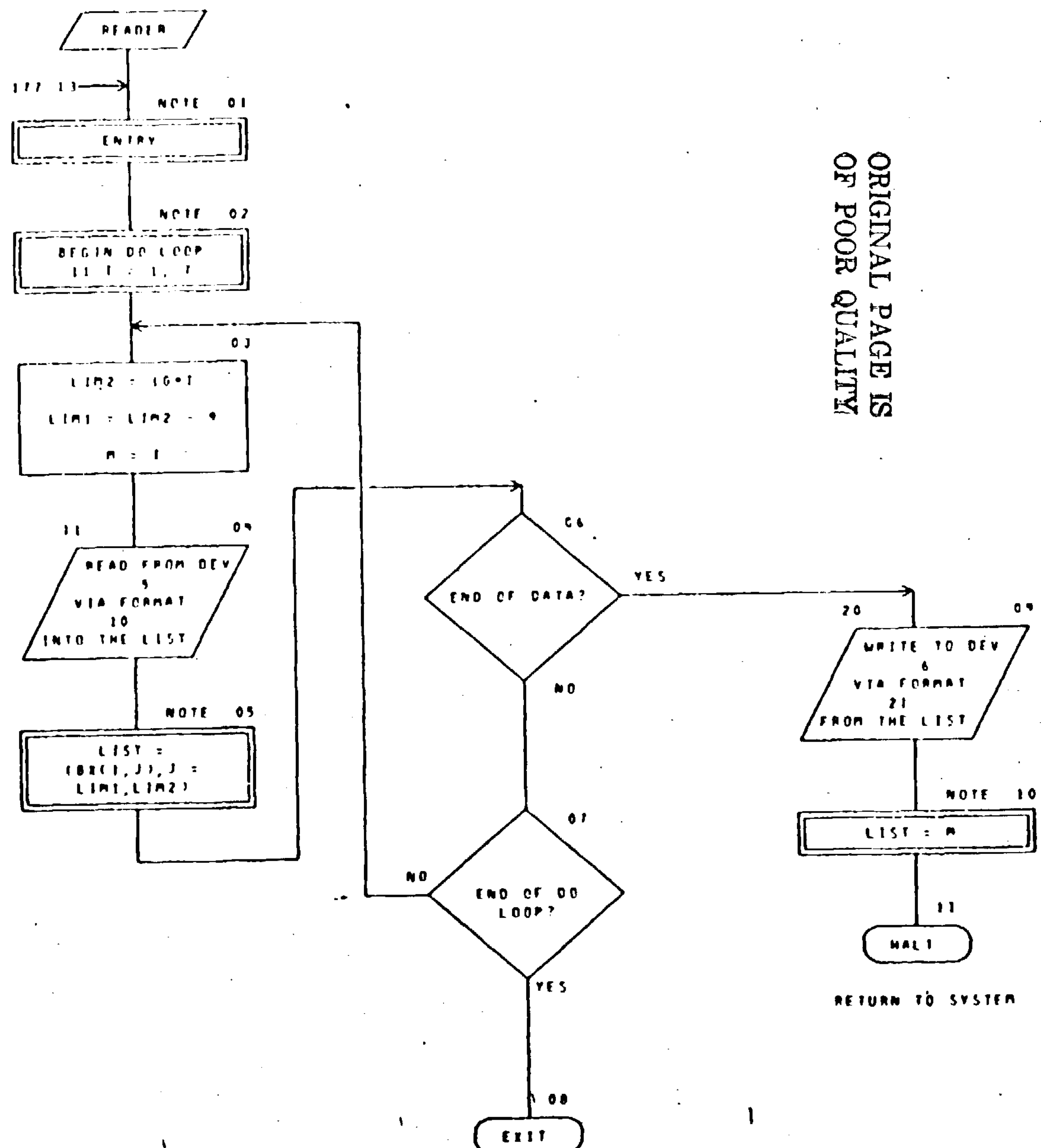
PUNCH-11

CHART TITLE - SUBROUTINE PUNCH(PINT,TIMX,PRATIO,IFLAG,IPUNCH)



ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - SUBROUTINE PUNCH(PINT,TIMX,PRATIO,IFLAG,IPUNCH)



ORIGINAL PAGE IS
OF POOR QUALITY

PUNCH-13

CHART TITLE - NON-PROCEDURAL STATEMENTS


```
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*4 AJFLAG,AEND
      LOGICAL ERROR,CONVRG
      DIMENSION PINT(17),TARG(6),TARGD(6),CARDID(5)
     .COMMON /REAL8/ PAYLOD,XMASS(7),R21,CTANK,CSTR,R01(3),DTMASS(5),
      OMPFTR,CIRET,PPER,RAP,THRET,SPIRET,OVEL,VLOSS,VORB,R02(6),TRIP,R03
     .,PT,F,PWRM,FMAX,ANGD,HOUR,R04,AAI,STATE(6),X0(7),R05(7),BI,CI,EI,
      B1,B2,B3,POWFIX,AP,R06(4),AE,SAI,ECI,CNI,OMI,SOI,TPI,EMUODD,RACODD
     .,R20,STEP1,STEP2,
                R07(66),TBURN,TDATE1,R08,TDATEX,R09(9),ANG1,ANG2,R10(3)
     .,DECL,R11,XINCL,R12(2)    ,RMAX,RMIN,R13,COMDIS,COMANG,R14(18),R(2)
     .,R15(63),ALPHAA,ALPHAT,R22(33)),
              AN,R16(11),DEG,R17(2),CONTM,R18(434),X(50),R21(93),
       TAU,R19(646)
      COMMON/INTGR4/I06,MODE,IRK,I01(3),MBOOST,MOPT,NTAPE,MDAY,MONTH,
      MYEAR,I02(8),IROT,JPP,JT,MOPT2,MOPT3,I03(11),MTMASS,I04(2),MPUNCH,
      I05(13)               ,KOUNT,IBAL,I07(14),LAUNCH,ISTAR,IEND,I08(928)
      COMMON /LOGIC4/ ERROR,CONVRG,L01(498)
      COMMON /ITERAT/ BX(5,70),B01(420),DD(70),B02(190)
      EQUIVALENCE (JFLAG,AJFLAG),(IEND,AEND)
      DATA CARDID /8HT,M,PR  ,8HR,RDOT  ,8HPHOTON  ,8HTR,TRDOT,4HP,PDOT/
      DATA ZERO,CSUBSO /0.D0,4.6D-6/
 31   FORMAT(I2,I1,I2,I1,3I2,3I1,3I2,3I1,I2,50X2I2)
 32   FORMAT(19A4,2I2)
 33   FORMAT(16A4,12X,2I2)
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
100      FORMAT(1P3D15.8,25X2I1,A8)
200      FORMAT(1P6D12.5,A8)
300      FORMAT(1P4D15.8,12XA8)
400      FORMAT(30A8)
10       FORMAT(10A8)
21       FORMAT(1H0,12HINPUT CARD *I1,1CH IS ABSENT)
```

Name: QPRINT

Calling Argument: IPRINT

Referenced Sub-programs: BOOSTR, EFMPRT, MORE, PDATE, SOLAR, VCROSS, VMAG

Referenced Commons: INTGR4, ITERAT, LOGIC4, REAL8, SOLSYS

Entry Points: None

Referencing Sub-programs: FINISH, QSTART

Discussion: Subroutine QPRINT performs the printing of the iterator parameters and the performance summary page. The calling argument IPRINT assumes a value in the range 1-5, and the particular value on entry determines what information is to be printed. Calls with IPRINT equal to 1, 2, 3 or 4 are made from QSTART and result in the printing of the iterator parameters. When IPRINT = 1, the page heading and independent variables table heading and column titles are printed. Calls with IPRINT = 2 are made in a DO loop in QSTART, with one call being made for each parameter identified as an independent variable of the boundary value problem. On each call there is printed the current value of a counter of independent variables, the index number which identifies the independent variable (equal to the i of the input independent variable arrays Xi), the initial value of the variable, the maximum step size permitted for the associated variable on one iteration, the perturbation step size used for generating partial derivatives and a weighting parameter. When IPRINT = 3, the dependent variables table heading and column titles are printed. Calls with IPRINT = 4 are made iteratively in QSTART, with one call being made for each parameter identified as a dependent variable of the boundary value problem. Each of these calls results in the printing of the current value of a counter of dependent variables, the index number which identifies the dependent variable (equal to the i of the input dependent variable arrays Yi), the desired value of the dependent variable, and the tolerance to which the dependent variable must be satisfied for convergence.

A call with IPRINT = 5 is made from FINISH to achieve the printing of the performance summary. This summary attempts to capsulize on a single page most of the important performance, spacecraft and trajectory parameters that are needed to analyze a given optimum solution. The performance summary is printed for the final trajectory of a case whether or not the case converged. The first line of the summary contains the case number, a message as to whether the case converged, and the page title. This is followed with a one-line description of the mission giving such information as mission type (e.g., flyby, rendezvous, orbiter, out of the ecliptic, etc.), the names of the primary celestial bodies involved, and other useful information that is dependent on the specific mission. For example, for cometary or asteroid missions, the time of arrival relative to perihelion passage is given. For extra ecliptic missions, the final ecliptic inclination is printed as computed from the formula

$$i = \cos^{-1}\{[(R \times \dot{R})/|R \times \dot{R}|] \cdot \bar{k}\},$$

where R and $\dot{R}$ are the final spacecraft position and velocity vectors and $\bar{k}$ is a unit vector along the ecliptic North Pole. The next line contains the name of the launch vehicle specified and lists the three coefficients b_1, b_2 and b_3 for that vehicle which are used in the computation of initial spacecraft mass

$$m_o = b_1 e^{-(v_c/b_2)} - b_3.$$

If the launch vehicle independent mode is specified, then this line is replaced with a simple message to that effect. The next two lines contain the launch date and the primary target arrival date, expressed both in the form of calendar dates and Julian dates (with leading 24 omitted). Also printed here is the overall flight time computed as the difference of these two dates. For missions not involving the use of the analytic ephemeris, only the flight time is printed.

The performance summary page continues with several sections of print, each headed with a descriptive title and containing several closely related variables. The first of these contains input electric propulsion system parameters and includes the specific masses of the solar array and thruster subsystem, α_a and α_t, respectively, the propellant tankage factor k_t and structure factor k_s, and the efficiency law coefficients b, d and e used in the formula for the efficiency, η

$$\eta = b c^2/(c^2 + d^2) + e\ ,$$

with c being the jet exhaust speed. The second section contains the spacecraft mass breakdown, including the initial mass m_o, the propulsion system mass m_{ps}, the propellant mass m_p, the tankage mass m_t, the structure mass m_s, and the net mass m_{net}. It does not include the retro stage mass, if any, because that is contained in a subsequent section. All of these mass components are evaluated in other subroutines and accessed in QPRINT through the appropriate common. The next section contains electric propulsion system performance parameters, including the reference power p_{ref}, housekeeping power p_h, power deliverable to the thruster subsystem at target arrival p_t, reference thrust level f_{ref}, reference thrust acceleration g, specific impulse I_{sp}, total propulsion system efficiency η, and the input characteristic degradation time τ_d. The equations for those variables unavailable in any common are:

$$p_h = p_{ref} \Delta p$$

$$p_t = q \gamma(t_f)\, p_{ref}$$

$$f_{ref} = g\, m_o$$

$$I_{sp} = c/9.80665$$

where Δp is the input ratio of housekeeping to reference power, $\gamma(t_f)$ is the power ratio factor evaluated at the final time, and c is the jet exhaust speed expressed in units of m/sec. The next section contains extreme trajectory and performance conditions such as the maximum and minimum solar distances encountered during the flight, the maximum power and thrust developed over the trajectory, the total propulsion system operating time, the degradation time s, and the accumulated travel angle, θ_t. Prior to printing the next section, a message is printed, for those missions utilizing a retro stage, whether the propellant tankage and/or electric propulsion system are jettisoned prior to the retro maneuver. The next section contains the departure and arrival conditions including the geocentric declination of the launch asymptote, the launch parking orbit inclination relative to the equator, the launch excess speed, the launch energy c_3, (equal to the square of the launch excess speed), the arrival excess speed at the primary target, and the arrival energy c_4. For out-of-the ecliptic missions the arrival excess speed printed is zero. For all missions employing a retro maneuver the arrival excess speed printed is the value of the independent parameter X14 (with a conversion of units). For all other missions the value printed is the magnitude of the vector difference between the spacecraft velocity and the velocity of the primary target.

For missions that require a retro maneuver, a section is printed which summarizes the important performance and spacecraft parameters involved in the maneuver. The parameters printed include the retro stage structure and propellant masses, the input thrust level f_r and specific impulse I_{sp_r} of the retro stage, the burn time t_b of the maneuver, the periapse and apoapse distances of the capture orbit, the orbital speed at the point of insertion, the incremental (impulsive) speed imparted by the maneuver, and the additional incremental velocity required due to finite thrust effects. The burn time is evaluated using the formula

$$t_b = m_{rp} I_{sp_r} / f_r ,$$

where m_{rp} is the retro stage propellant.

If the spiral capture maneuver is requested, another section of performance data is printed, including the periapse and apoapse distances of the capture orbit, the thrust level and specific impulse of the spiral maneuver, the total mission duration including the time of the spiral maneuver, and the additional propellant required during the spiral maneuver.

Finally, certain options invoked result in the printing of additional information on the performance summary page. If the fixed thrust angle option is flagged, the value of the angle used for the trajectory being summarized is printed. For multi-target missions, the inclusion of sample pickup or mass drops at intermediate targets results in printing of the amount of mass retrieved and/or deposited at each intermediate target.

Prior to exiting QPRINT, a call is made to subroutine MORE which will perform the ballistic swingby continuation analysis, if flagged.

Messages and printouts: Typical examples of the iterator parameters and the performance summary pages are shown on the following pages. Additional examples are shown in the section of Sample Problems and Results.

CASE 1

ITERATOR PARAMETERS

INDEPENDENT VARIABLES

NO.	INDEX	VALUE	STEP LIMIT	PERTURBATION	WEIGHT
1	1	-3.7284687000000000D-01	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
2	2	-2.9597932000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
3	3	-8.9227992000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
4	4	-1.0251788000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
5	5	5.9174474000000000D-01	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
6	6	8.7004218000000000D-01	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
7	10	-4.6795107000000000D 01	9.0000000000000000D 01	9.9999999999999990D-07	1.0000000000000000D 00
8	11	4.1251643000000000D-04	9.9999999999999990D-04	1.0000000000000000D-11	1.0000000000000000D 00
9	13	8.1252153000000000D 03	5.0000000000000000D 02	9.9999999999999990D-05	1.0000000000000000D 00
10	41	-1.1957498000000000D 01	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
11	42	-7.8462757000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
12	43	1.7259608000000000D 01	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
13	44	1.7539259000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
14	45	-1.2359365000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
15	46	-3.9607527000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
16	48	-6.0520825000000000D 02	5.0000000000000000D 01	9.9999999999999990D-07	1.0000000000000000D 00
17	51	-3.5162113000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
18	52	-4.8196508000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
19	53	-8.5469620000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
20	54	1.0282080000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
21	55	3.1871944000000000D-02	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
22	56	1.8149722000000000D 00	3.0000000000000000D 00	1.0000000000000000D-08	1.0000000000000000D 00
23	58	-3.0548648999999990D 02	5.0000000000000000D 01	9.9999999999999990D-07	1.0000000000000000D 00

DEPENDENT VARIABLES

NO.	INDEX	VALUE	TOLERANCE
1	1	0.0	9.9999999999999980D-06
2	2	0.0	9.9999999999999980D-06
3	3	0.0	9.9999999999999980D-06
4	4	0.0	9.9999999999999980D-06
5	5	0.0	9.9999999999999980D-06
6	6	0.0	9.9999999999999980D-06
7	10	0.0	9.9999999999999980D-06
8	11	1.0000000000000000D 01	9.9999999999999990D-05
9	13	0.0	9.9999999999999980D-06
10	41	0.0	9.9999999999999980D-06
11	42	0.0	9.9999999999999980D-06
12	43	0.0	9.9999999999999980D-06
13	44	0.0	9.9999999999999980D-06
14	45	0.0	9.9999999999999980D-06
15	46	0.0	9.9999999999999990D-06
16	48	0.0	9.9999999999999980D-06
17	51	0.0	9.9999999999999980D-06
18	52	0.0	9.9999999999999980D-06
19	53	0.0	9.9999999999999980D-06
20	54	0.0	9.9999999999999980D-06
21	55	0.0	9.9999999999999980D-06
22	56	0.0	9.9999999999999980D-06
23	8	0.0	9.999999999999998[illegible]

PERFORMANCE SUMMARY

CASE 1 (CONVERGED)

EARTH TO MERCURY RENDEZ WITH HIGH THRUST CAPTURE MANEUVER.

(COEFFICIENTS = 50315.8320 2199.9809 44.4500)

FLIGHT TIME = 510.0000 DAYS.

LAUNCH VEHICLE IS INPUT

LD = FEB 24, 1980, 12.0000 HOURS GMT
JULIAN DATE 44294.0000

AD = JUL 19, 1981, 12.0000 HOURS GMT
JULIAN DATE 44804.0000

ELECTRIC PROPULSION SYSTEM PARAMETERS

ALPHA A (KG/KW)	ALPHA T (KG/KW)	TANKAGE FACTOR	STRUCTURE FACTOR
15.0000	15.0000	0.0300	0.0

EFFICIENCY COEFFICIENTS

B	D (KM/SEC)	E
0.76000	13.00000	0.0

ELECTRIC PROPULSION SYSTEM MASS SUMMARY (KG)

INITIAL	POWER PLANT	PROPELLANT	TANKAGE	STRUCTURE	NET MASS
289.9529	74.9079	101.4991	3.0450	0.0	97.1040

ELECTRIC PROPULSION SYSTEM PERFORMANCE SUMMARY

P(1 AU) (KW)	P(HSKP) (KW)	P(TARG) (KW)	THR(1 AU) (N)	ACC(1 AU) (M/SEC**2)	ISP (SEC)	EFFIC	CHAR DEG (DAYS
2.4969	0.0	3.4865	0.082321	2.839112D-04	4231.864	0.69381	1.00000000 3(

EXTREME TRAJECTORY AND PERFORMANCE CONDITIONS

MAX DIST (AU)	MIN DIST (AU)	MAX POWER (KW)	MAX THRUST (N)	BURN TIME (DAYS)	DEGRD TIME (DAYS)	TRAV ANG (DE
1.0157910	0.3497653	3.486533	0.11494698	472.95738	853.83930	929.0116!

POWERPLANT JETTISONED PRIOR TO RETRO MANEUVER
TANKAGE MASS JETTISONED PRIOR TO RETRO MANEUVER

DEPARTURE AND ARRIVAL CONDITIONS

DEP DECL (DEG)	PARK INC (DEG)	DEP VINF (M/SEC)	C3 (KM**2/SEC**2)	ARR VINF (M/SEC)	C4 (KM**2/SEC**2)
-0.4403	28.5000	447.43351	0.200197	1380.46478	1.905683

HIGH THRUST CAPTURE MANEUVER STAGE AND ORBIT SUMMARY

STRUCTURE (KG)	PROPELLANT (KG)	THRUST (LBS)	ISP (SEC)	BURNING TIME (SEC)
1.3397	12.0573	20.0000	300.0000	40.6589

PERIAPSE (RADII)	APOAPSE (RADII)	ORBIT VEL (M/SEC)	DEL VEL (M/SEC)	VEL LOSS (M/SEC)
1.2000	22.8000	3711.9977	339.9173	1.0123

QPRINT-7

ORIGINAL PAGE IS
OF POOR QUALITY

QPRINT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
O(70)	U	ITERAT	Array of iterator independent-variables, in program internal units.
R(2)	U	REAL8	Spacecraft solar distances (at prior and current computation point), in AU.
X(50)	SUA	REAL8	Array of trajectory integrated variables.
BI	U	REAL8	Efficiency coefficient b in equation for efficiency.
BX(5,70)	U	ITERAT	Iterator independent variable array.
BY(3,70)	U	ITERAT	Iterator dependent variable array.
B1	SUA	REAL8	Launch vehicle coefficient b_1, in kg.
B2	SUA	REAL8	Launch vehicle coefficient b_2, in meters/second.
B3	SUA	REAL8	Launch vehicle coefficient b_3, in kg.
C3	SU	REAL8	Launch energy, equal to the square of the launch excess speed.
C4	SU	REAL8	Energy at arrival of primary target, equal to the square of the arrival excess speed.
DI	U	REAL8	Efficiency coefficient d in equation for efficiency, in km/sec.
EI	U	REAL8	Efficiency coefficient e in equation for efficiency.
JT	U	INTGR4	Jettison indicator j_t, for electric propulsion tankage prior to primary-target retro-maneuver.
OO(70)	SU	ITERAT	Array of iterator independent variables in program external units.

QPRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
PT	SU	REAL8	Power function value at primary target intercept time, $(\gamma q)_f$.
RT	S	REAL8	Spacecraft solar distance, r, in AU.
X0(7)	SU	REAL8	Spacecraft initial state vector, x_o, $\dot{x}_o$, ν_o, where x_o is in AU, $\dot{x}_o$ is in EMOS, and $\nu_o = 1$ is set elsewhere.
APL(2, 70)	U	SOLSYS	Array of planet names.
ARR	S	REAL8	Arrival date in hours from input reference date.
DEG	U	REAL8	Conversion factor between radians and degrees; number of degrees in one radian.
DEP	S	REAL8	Launch date in hours from input reference date.
FFF	SU	REAL8	Reference thrust level, f_{ref}, in newtons.
JPP	U	INTGR4	Jettison indicator j_{ps} for electric propulsion system prior to primary-target retro-maneuver.
LXX	U	INTGR4	The number of (active) iterator independent variables.
MXX	U	INTGR4	Value of second subscript of BX array associated with current independent variable being analyzed.
NPR(4)	U	INTGR4	Printout control vector.
RAP	U	REAL8	Apoapse distance of capture orbit about primary target, in planet radii.
SAI	U	REAL8	Semi-major axis of primary-target orbit, in AU.

QPRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
TAU	U	REAL8	Accumulated propulsion time, τ, in 1 AU normalized units (tau).
TPI	U	REAL8	Time from reference date (which is specified by MYEAR, etc.) to perihelion passage, for the primary target, in days.
XSP	SU	REAL8	Specific impulse of electric propulsion system, in seconds.
ANGD	SU	REAL8	Travel angle, θ_t, in degrees.
ANG1	U	REAL8	Launch site latitude, in radians.
ANG2	U	REAL8	Maximum launch parking orbit inclination, in radians.
CSTR	U	REAL8	Electric propulsion system structural factor, k_s.
DECL	U	REAL8	Departure asymptote declination δ, in radians.
DPOW	U	REAL8	Ratio of housekeeping power to reference power, $p_h/p_{ref} = \Delta p$.
DVEL	U	REAL8	Retro maneuver (at primary target) impulsive velocity increment, in meters/ second.
FMAX	SU	REAL8	Maximum thrust along trajectory, in newtons.
MONA	SUA	INTGR4	Calendar month of the date of arrival at primary target.
MOND	SUA	INTGR4	Calendar month of the launch date.
MPOW	U	INTGR4	Maximum or optimum power indicator during solar panel degradation option.

QPRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
NSET(5)	U	INTGR4	Iteration-sequence control array.
PMAX	U	REAL8	Maximum value of power ratio, $(q\gamma)_{max}$, encountered along the trajectory.
POWR	U	REAL8	Power ratio, γq.
PWRM	SU	REAL8	Maximum power along trajectory, in kw.
RMAX	U	REAL8	Maximum solar distance along trajectory, in AU.
RMIN	U	REAL8	Minimum solar distance along trajectory, in AU.
RPER	U	REAL8	Periapse distance of capture orbit about primary target, in planet radii.
TOFF(20)	U	REAL8	Array of times, from the start of the trajectory, at which imposed coast phases are to begin, in days.
TRIP	SU	REAL8	Flight time, from launch to the primary target, in days.
VIMP	U	REAL8	Launch hyperbolic excess speed, $v_{\infty o}$, in EMOS.
VORB	U	REAL8	Speed at periapse of capture orbit about primary target, in meters/second.
V00D(3)	U	REAL8	Launch hyperbolic excess velocity, $V_{\infty o}$, in AU/tau.
XINT(50, 5)	SUA	REAL8	Saved trajectory dependent-variable values at arrival at the intermediate targets.
XJLA	SU	REAL8	Julian date of arrival at primary target, less 2400000.
XJLD	SU	REAL8	Julian date of launch, less 2400000.

QPRINT-11

QPRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ANGLE	U	REAL8	Travel angle, θ_t, in radians.
BRAKE	U	LOGIC4	Logical flag indicating if there is a high-thrust retro maneuver at the primary target.
CONSP	U	REAL8	Speed conversion factor, from AU/tau to meters/second.
CONTM	U	REAL8	Time conversion factor, tau to days.
CTANK	U	REAL8	Electric propulsion system propellant tankage factor, k_t.
DROPS	U	REAL8	Sum of drop-mass factors $\sum_{i=1}^{n-1} k_{drop\,i}$.
ERODE	U	LOGIC4	Power degradation option indicator.
ERROR	U	LOGIC4	Program master error indicator.
FLYBY	U	LOGIC4	Indicator that maneuver at primary target is flyby.
HOURA	SUA	REAL8	Hour of the day of arrival at the primary target.
HOURD	SUA	REAL8	Hour of the day of launch.
KOUNT	U	INTGR4	Case number of computer run.
LOOSE	U	LOGIC4	Indicates that the initial heliocentric spacecraft velocity is included in the (active) iterator independent variables.
MDAYA	SUA	INTGR4	Day of the month of arrival at the primary target.
MDAYD	SUA	INTGR4	Day of the month of launch.
MOPTX(5)	SUA	INTGR4	The target-numbers of the successive intermediate targets.

QPRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
MOPT2	SUA	INTGR4	Launch planet number.
MOPT3	SUA	INTGR4	Planet-number of primary target.
QDECL	U	LOGIC4	Non-coplanar launch maneuver indicator.
SAMPS	U	REAL8	Sum of sample-mass factors $\sum_{i=1}^{n-1} k_{samp\,i}$.
SEFMA(7)	SUA	REAL8	Array containing position and velocity of launch planet at launch time, P_o and $\dot{P}_o$, in AU and EMOS, respectively.
SEFMB(7)	SUA	REAL8	Array containing position and velocity of primary target at time of target intercept, P_n, $\dot{P}_n$, in AU and EMOS, respectively.
TBURN	SU	REAL8	Retro-maneuver burn time at primary target, in seconds.
THRET	U	REAL8	Retro-stage thrust in retro-maneuver at the primary target, in pounds.
THSPY	U	REAL8	Spiral-stage thrust, in pounds.
VLOSS	U	REAL8	Velocity-loss, due to gravity, associated with the retro-maneuver at the primary target, in meters/second.
XINCL	U	REAL8	Parking orbit inclination, i, in degrees.
XMASS(7)	SU	REAL8	Array of spacecraft weights, in kilograms, and parameters.
XMSPY	U	REAL8	Spiral additional propellant, Δm_p, in kilograms.
XTINT(6, 5)	SUA	REAL8	Positions and velocities of intermediate targets at times of intercept, P_i and $\dot{P}_i$, in AU and EMOS.
ALPHAA	U	REAL8	Specific mass of solar arrays, α_a, in kg/kw.

QPRINT-13

QPRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ALPHAT	U	REAL8	Specific mass of thruster subsystem, α_t, in kg/kw.
CONVRG	U	LOGIC4	Iteration-sequence convergence indicator.
FIXPOW	U	LOGIC4	Indicator for the launch-vehicle-independent mode of operation.
FIXTHR	U	LOGIC4	Indicator for fixed thrust-angle.
IPRINT	UX		Calling argument defining the specific print desired.
LEGMAX	U	INTGR4	Total (maximum) number of trajectory-segments comprising the trajectory.
MBOOST	SUA	INTGR4	Launch vehicle selector.
MTMASS	U	INTGR4	Mission-type selector pertaining to the primary target.
MYEARA	SUA	INTGR4	Year of arrival at the primary target.
MYEARD	SUA	INTGR4	Year of launch.
OUTECL	U	LOGIC4	Extra-ecliptic mission indicator.
PAYLOD	U	REAL8	Net spacecraft mass, m_{net}, in kg.
PLANET	U	LOGIC4	Ephemeris-option indicator.
QTMASS(5)	U	REAL8	Array of mass-related parameters, as described in subroutine RETINJ.
SPIRAL	U	LOGIC4	Indicator for electric propulsion spiral capture maneuver at the primary target.
SPIRET	U	REAL8	Retro-stage specific impulse pertaining to the retro-maneuver at the primary target, in seconds.
SPISPY	U	REAL8	Spiral-stage specific impulse, in seconds.

QPRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
TCOAST (20)	U	REAL8	Array of times representing the durations of the coast-phases associated with the start times in TOFF, in days.
TDATEX	U	REAL8	Reference Julian date, less 2400000, defined by program input quantity MYEAR, etc.
TDATE1	SUA	REAL8	Launch Julian date, less 2400000.
TDATE2	SUA	REAL8	Julian date at time of primary-target intercept, less 2400000.
TIMSPY	U	REAL8	Spiral additional time, Δt, in days.
TPOWER	U	REAL8	Solar-cell degradation characteristic-time, in days.

CHART TITLE - SUBROUTINE OPRINT(IPRINT)

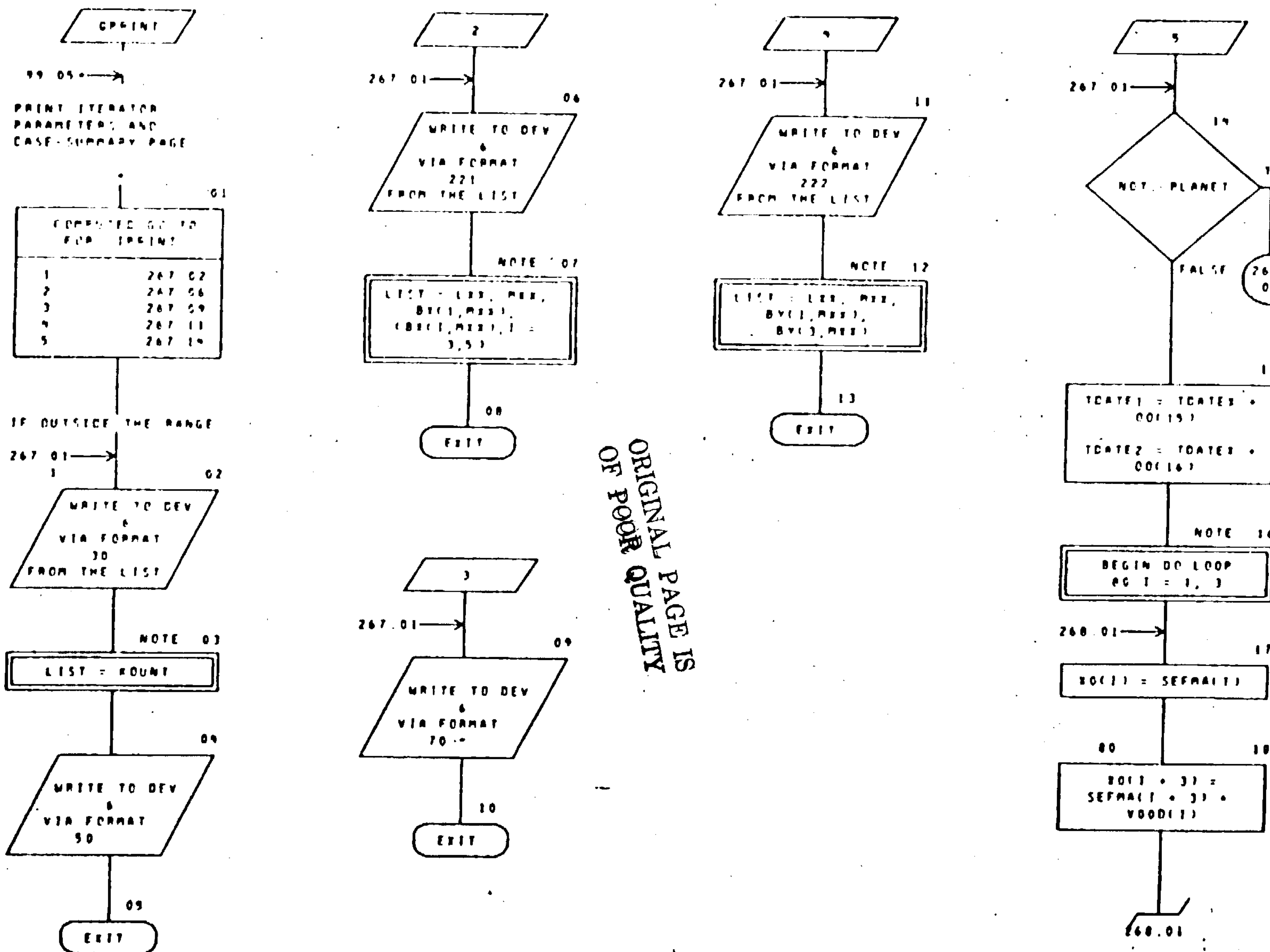


CHART TITLE - SUBROUTINE QPRINT(IPRINT)

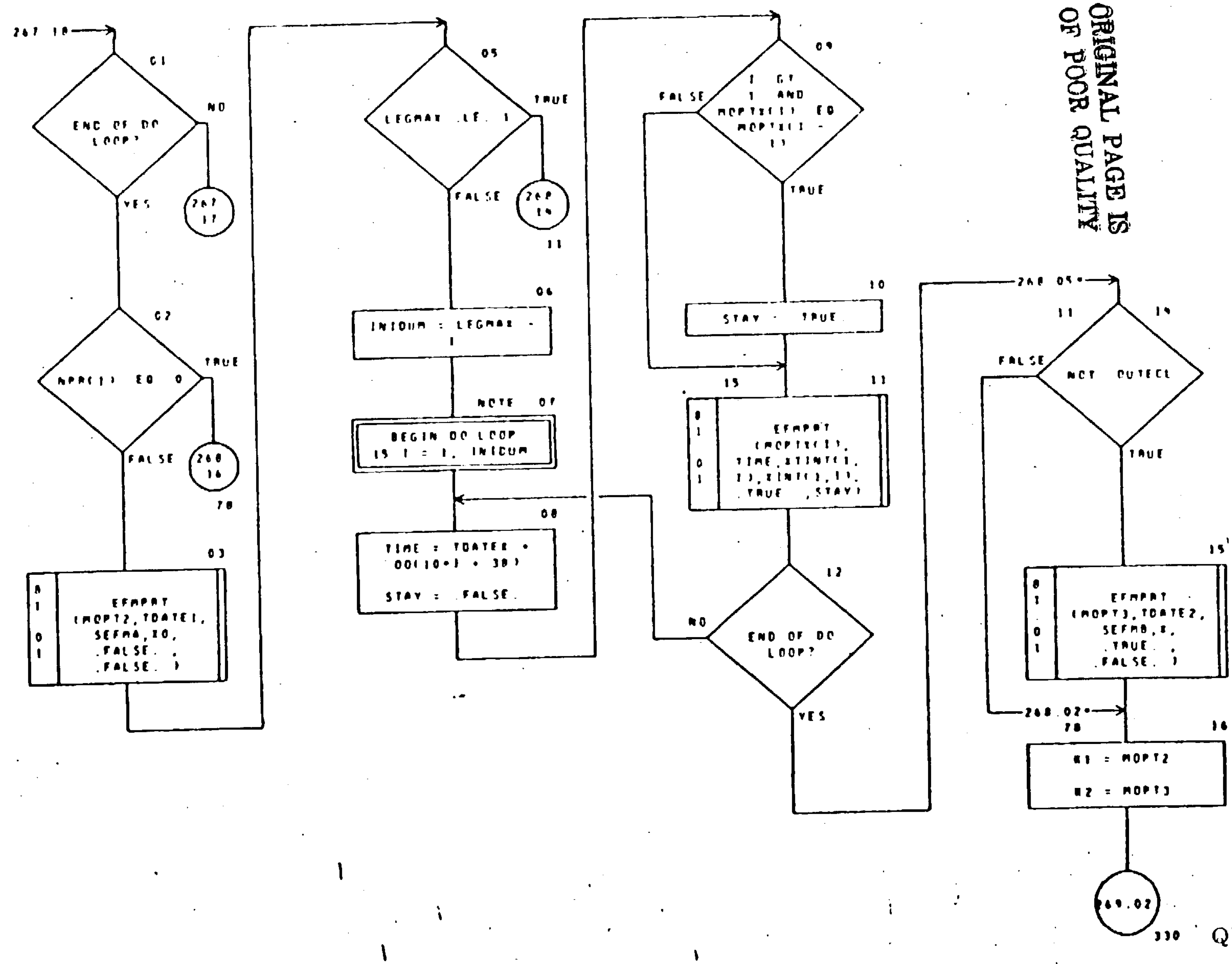


CHART TITLE - SUBROUTINE OPRINT(IPRINT)

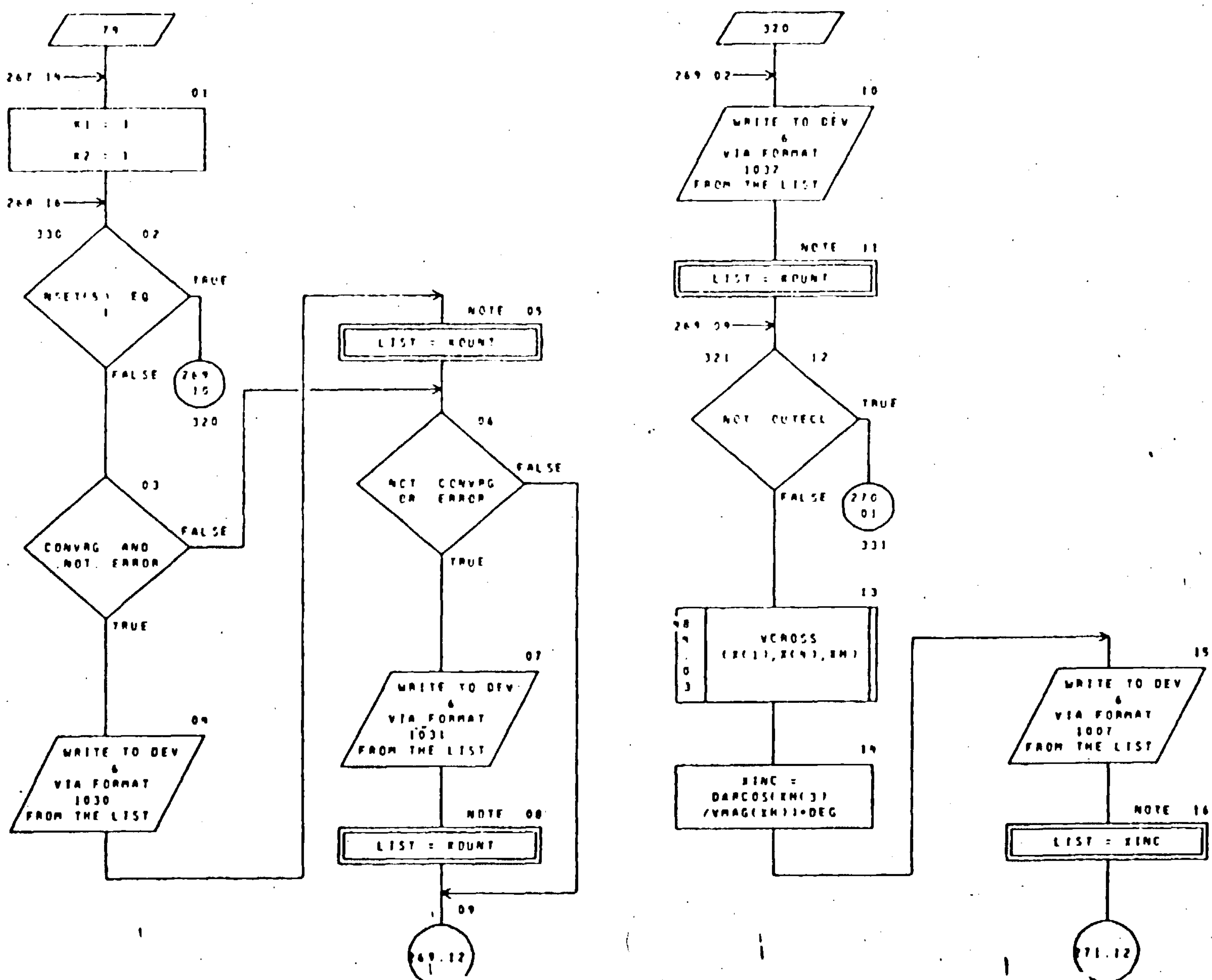
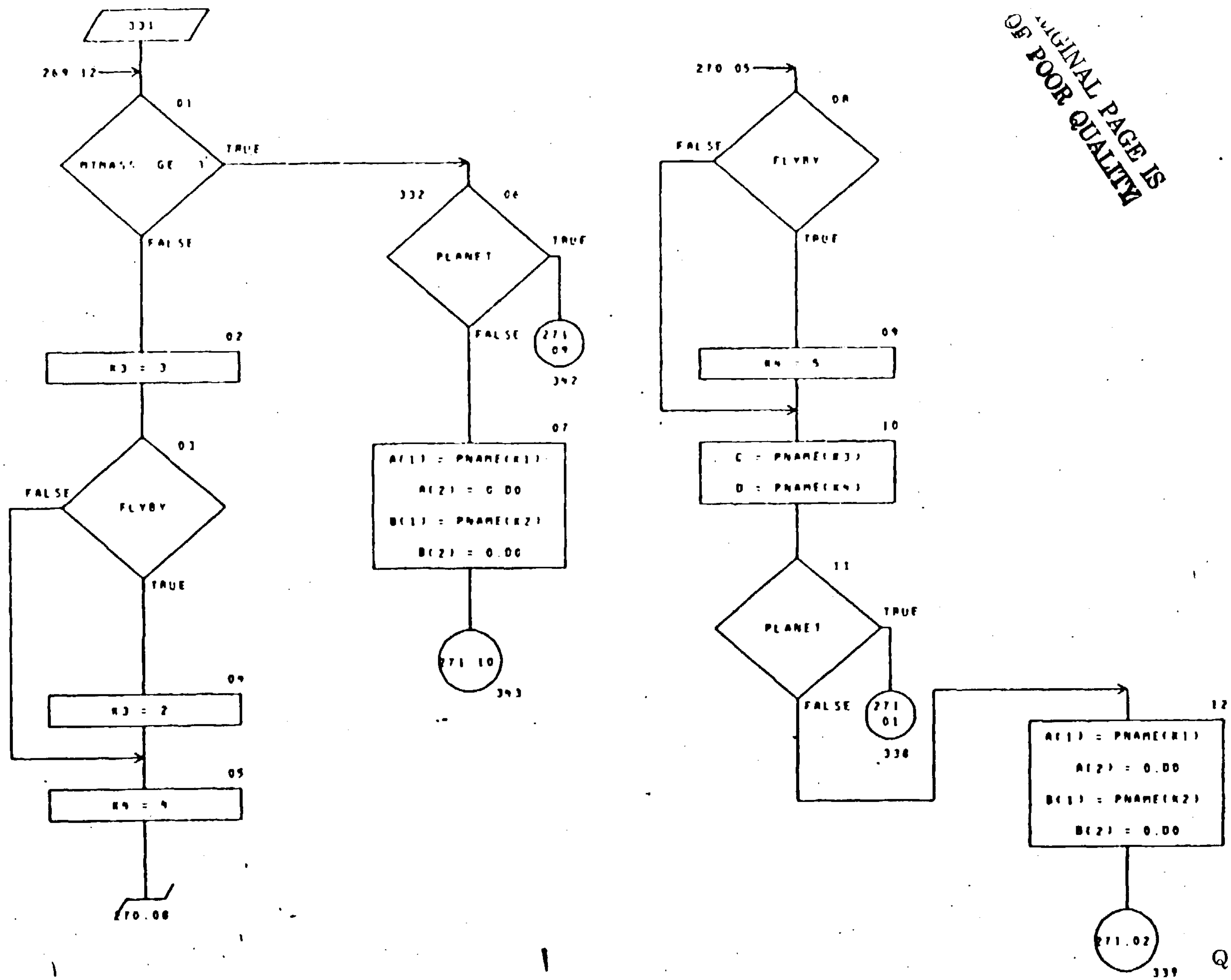


CHART TITLE - SUBROUTINE QPRINT(IPRINT)



QPRINT-19

CHART TITLE - SUBROUTINE QPRINT(IPRINT)

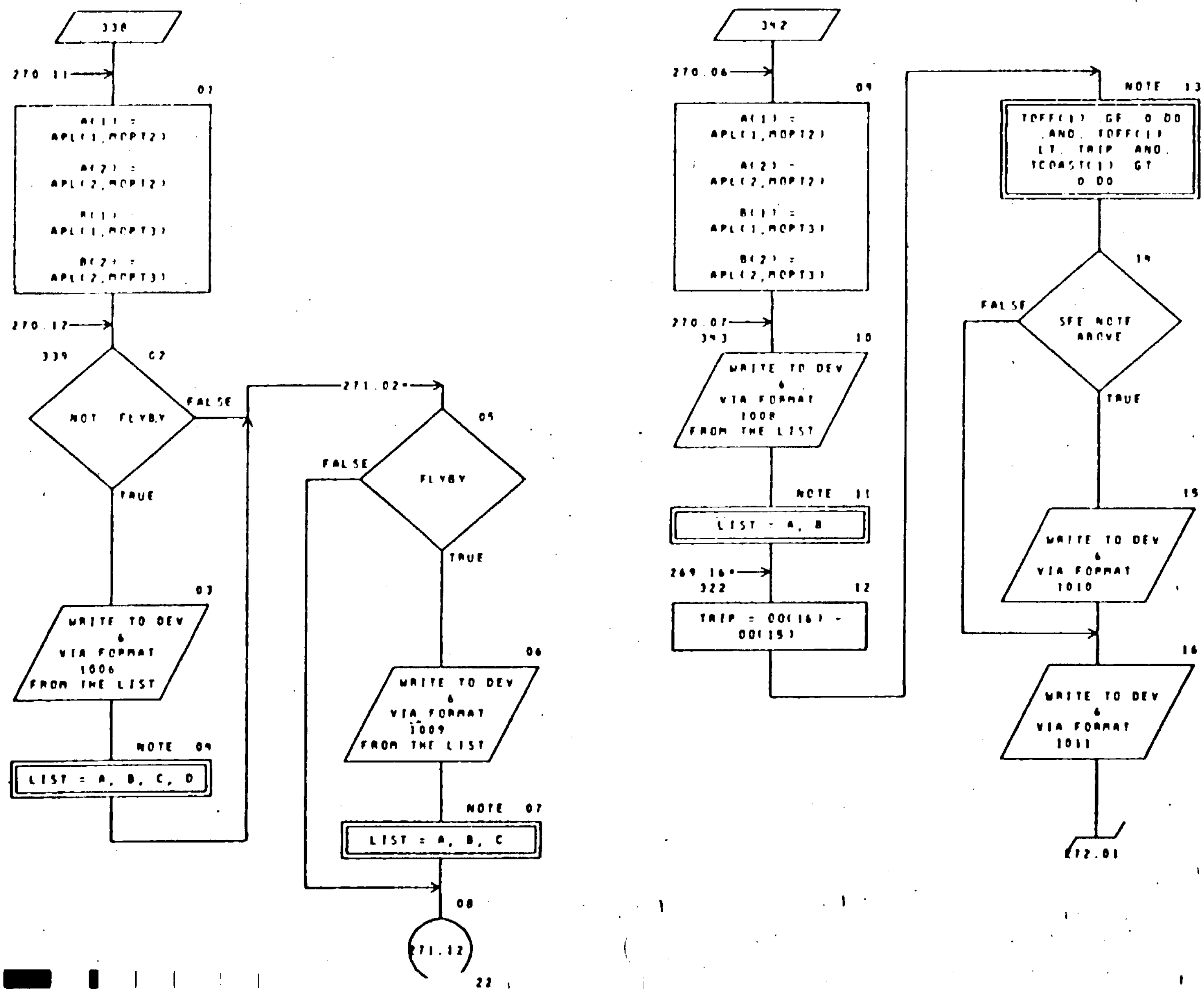
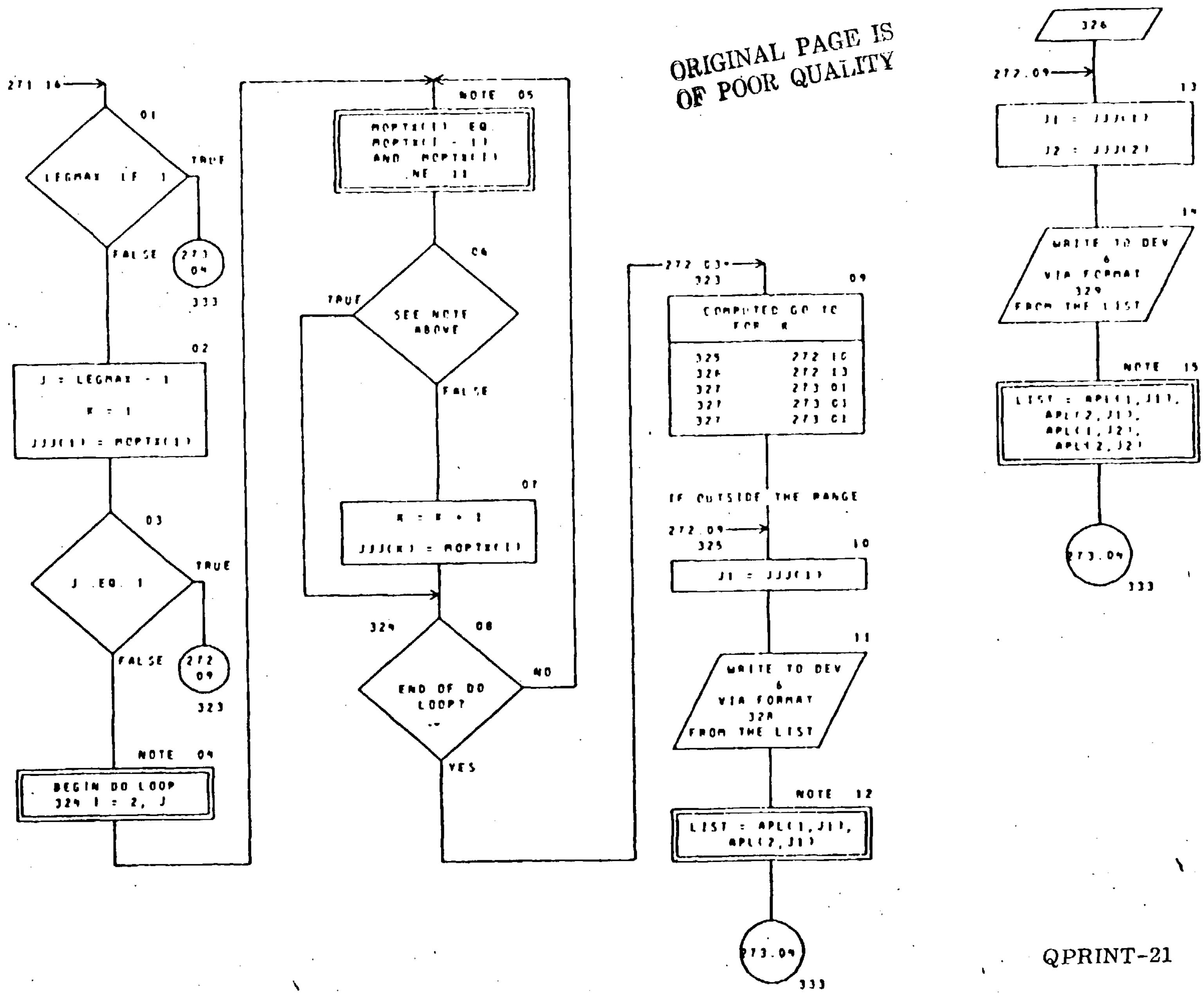


CHART TITLE - SUBROUTINE QPRINT(IPRINT)



QPRINT-21

CHART TITLE - SUBROUTINE OPRINT(PRINT)

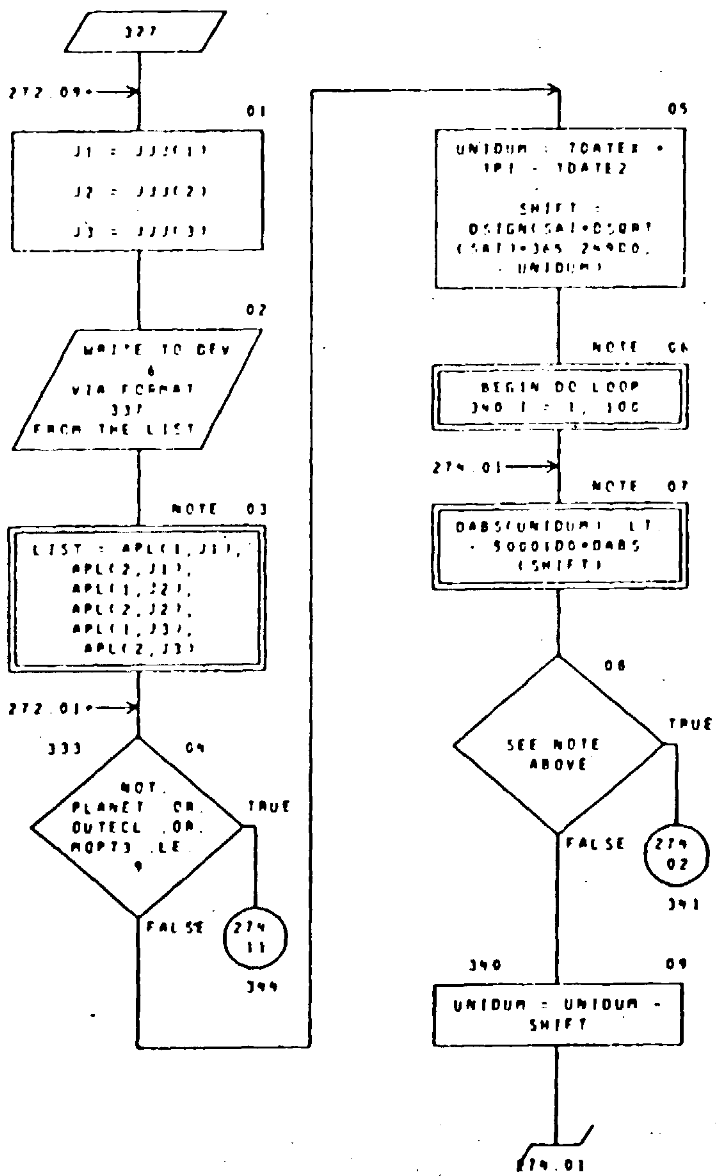


CHART TITLE - SUBROUTINE QPRINT(IPRINT)

C-6

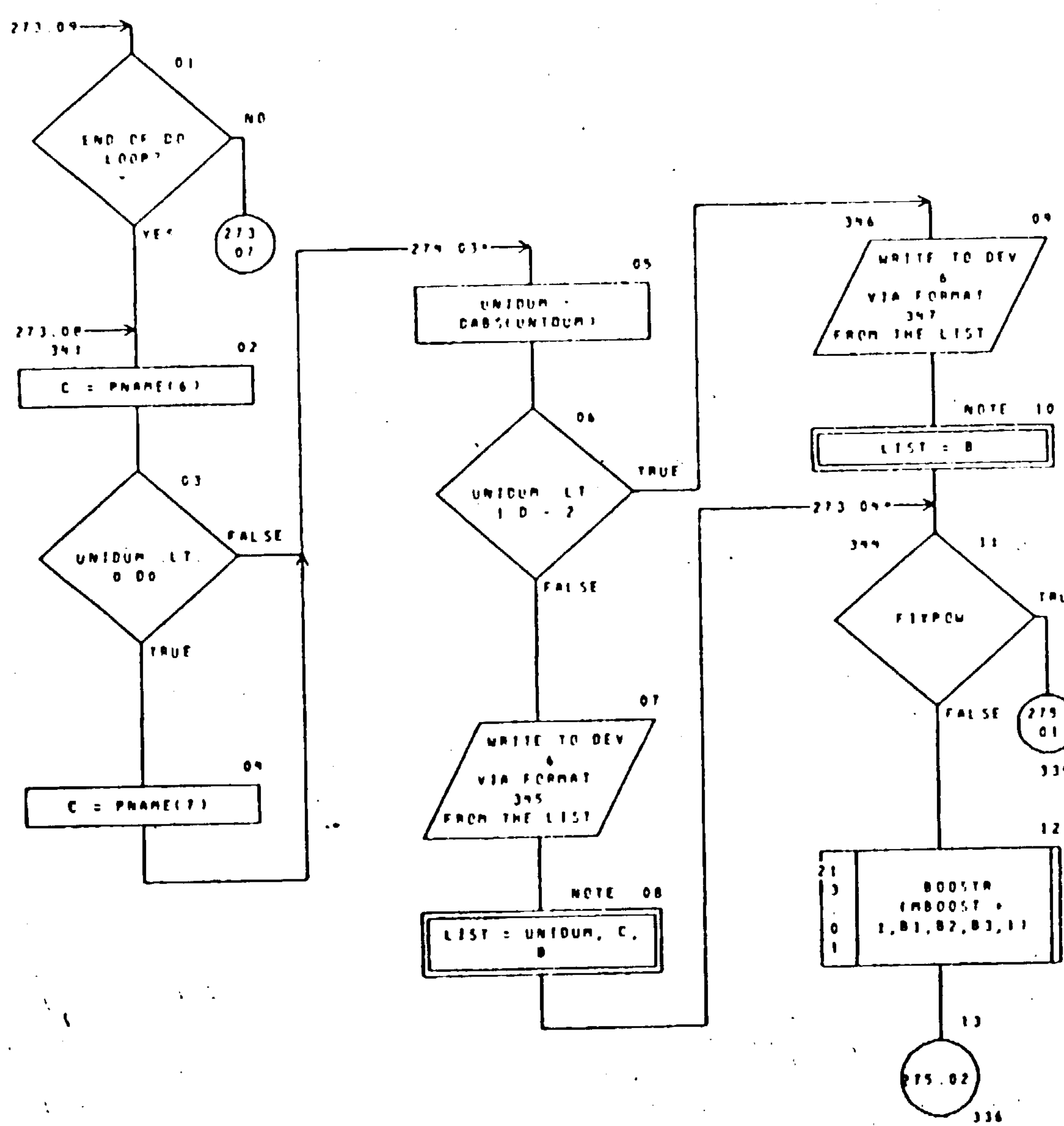


CHART TITLE - SUBROUTINE OPRINT(IPRINT)

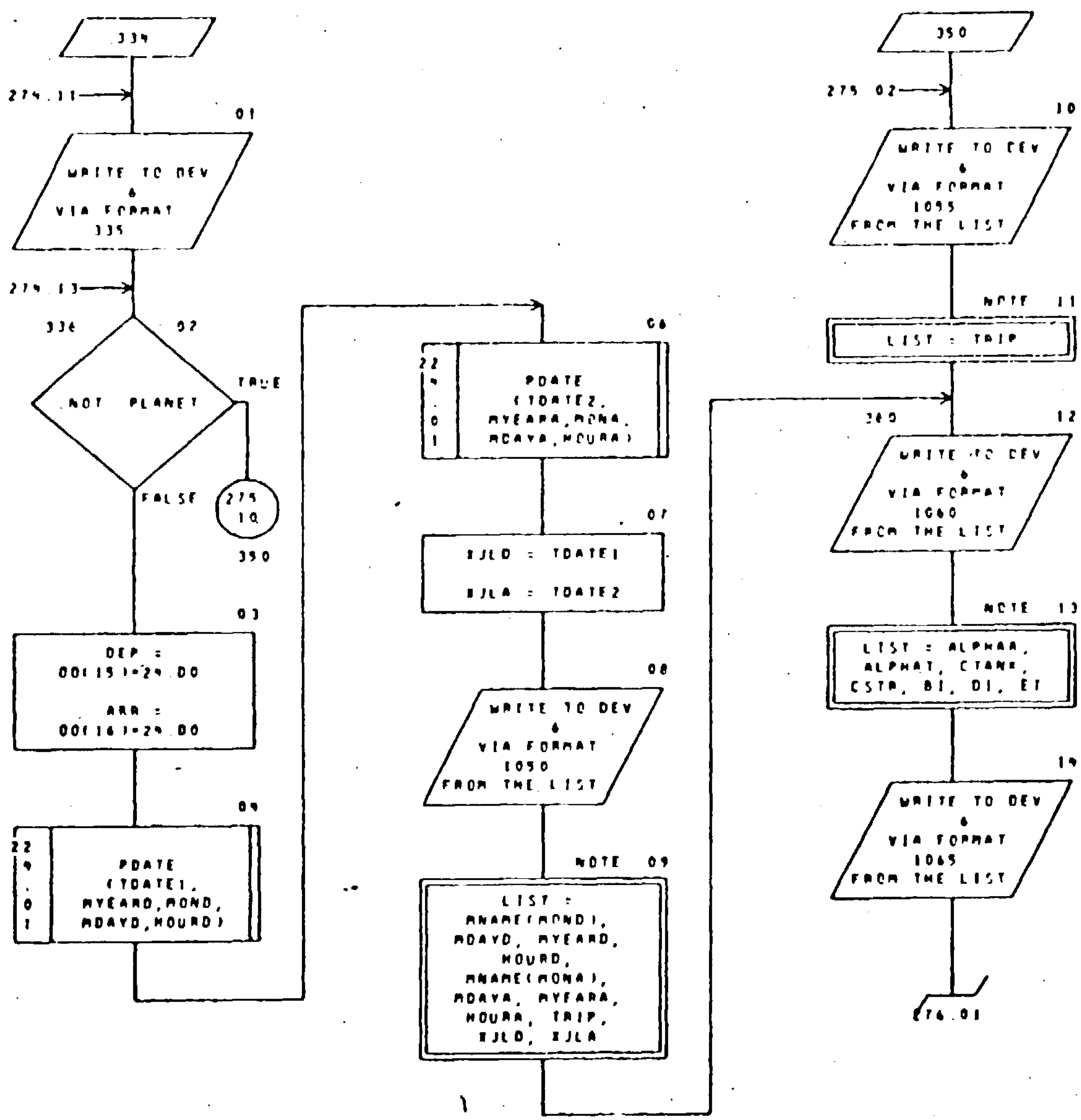
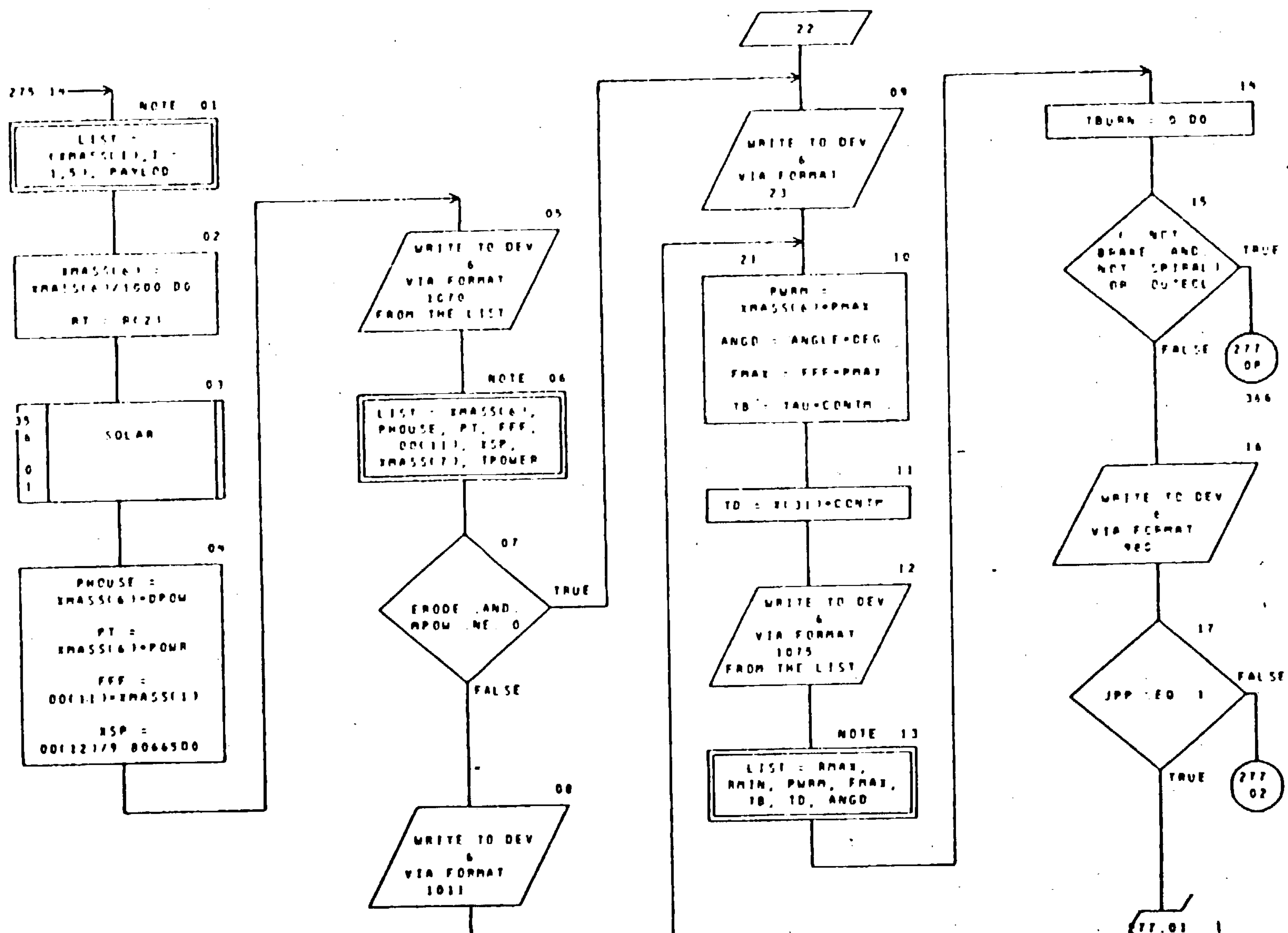


CHART TITLE - SUBROUTINE QPRINT(PRINT)



QPRINT-25

CHART TITLE - SUBROUTINE OPRINT(IPRINT)

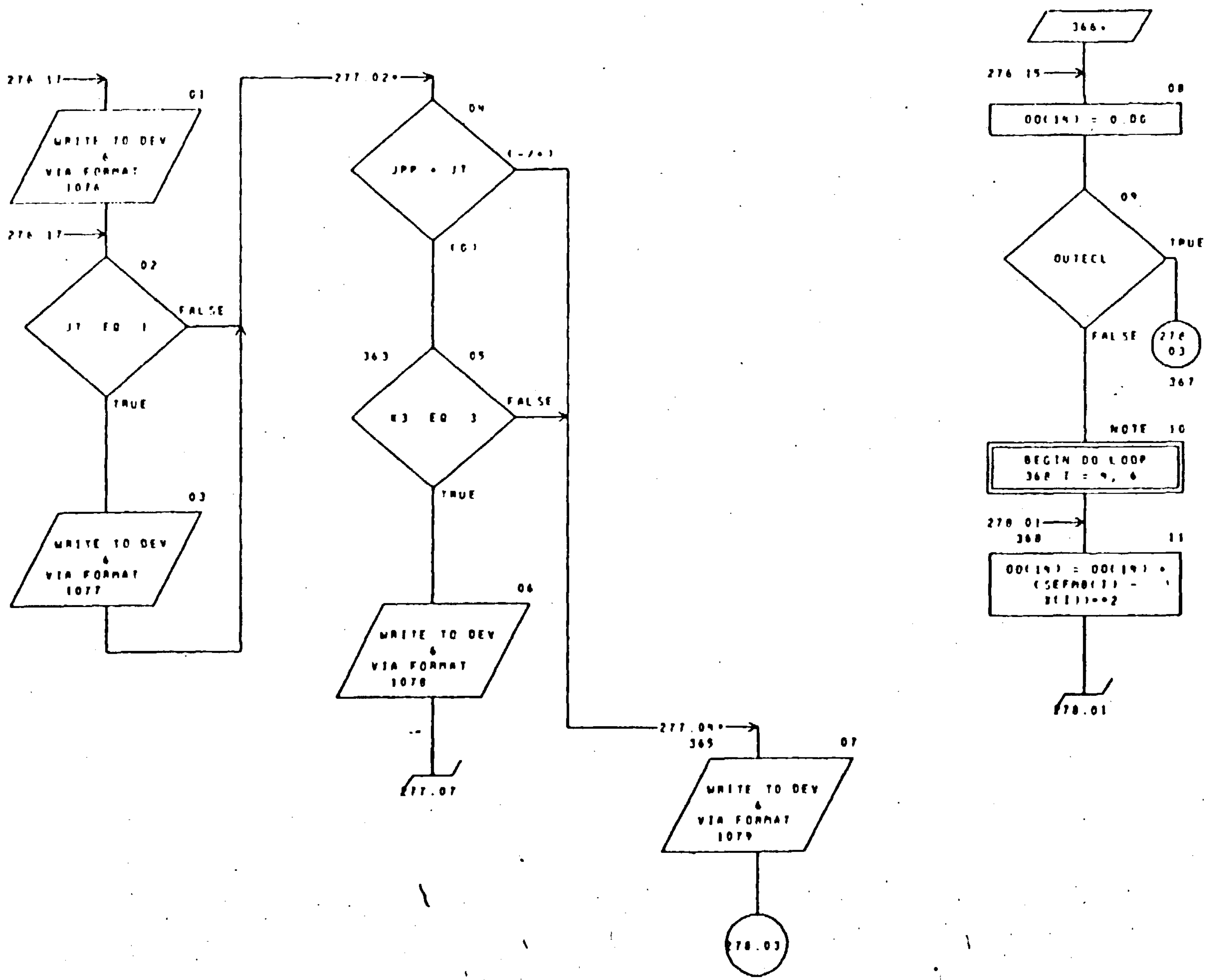
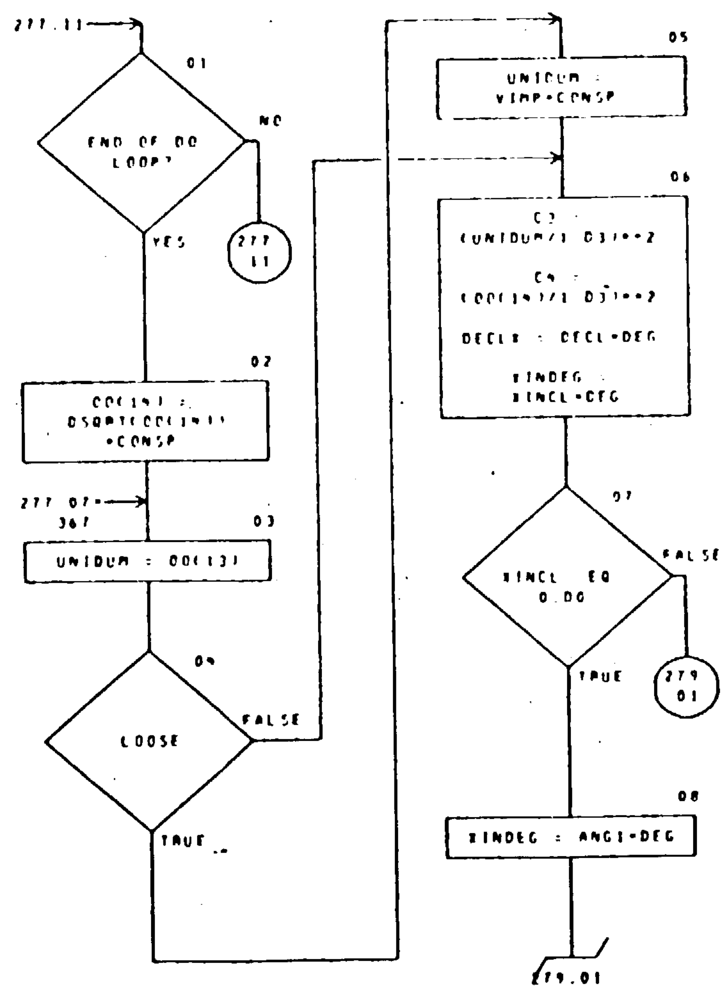


CHART TITLE - SUBROUTINE QPRINT(IPRINT)



QPRINT-27

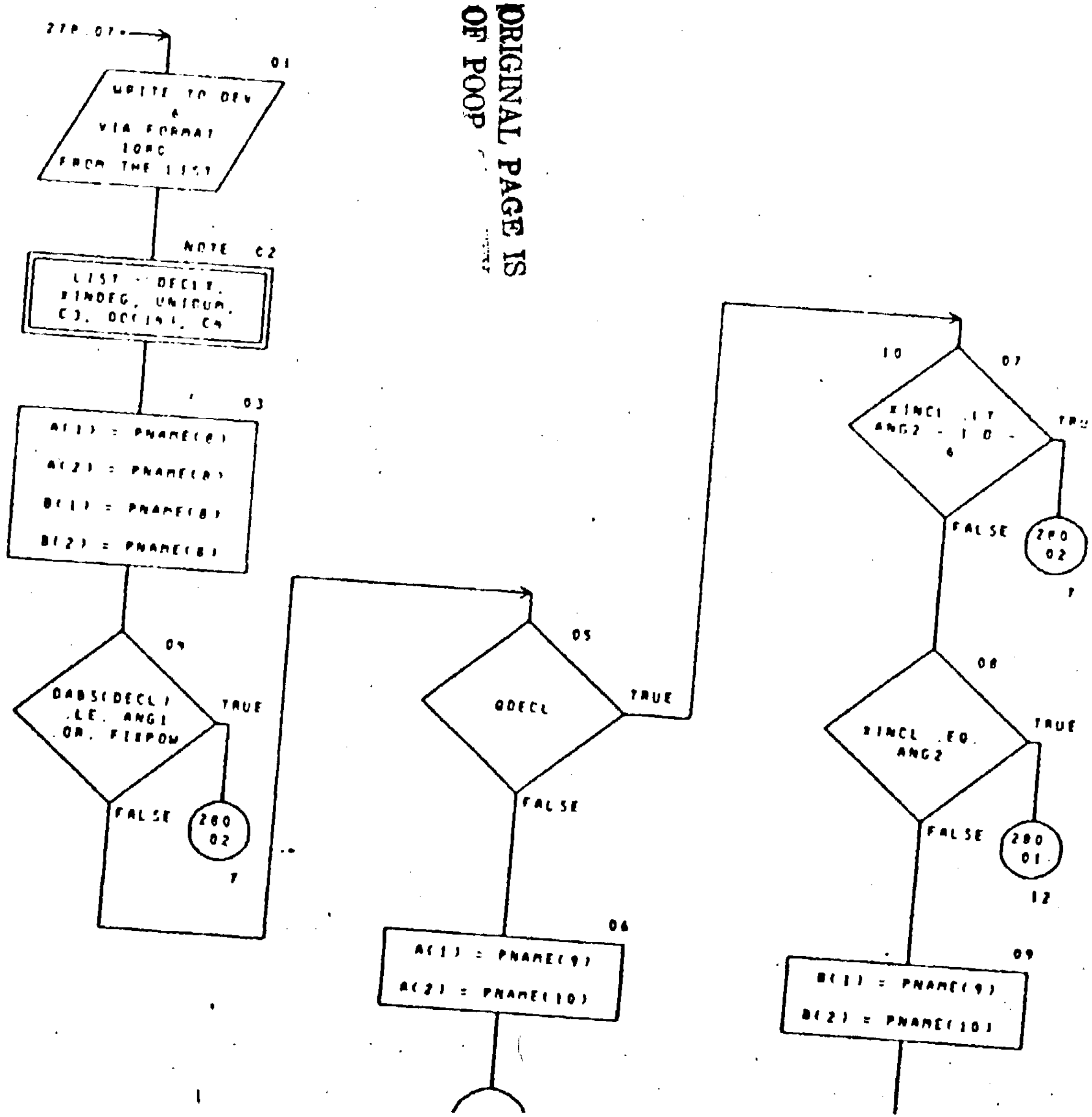

279 07
01
WRITE TO DEV 6 VIA FORMAT 10RC FROM THE LIST
NOTE C2
LIST - DECLT, XINDEG, UNIDUM, C3, DECLN1, C4
03
A(1) = PNAME(2)
A(2) = PNAME(2)
B(1) = PNAME(8)
B(2) = PNAME(8)
04
DABS(DECL) .LE. ANG1 .OR. FIXPOW
TRUE
FALSE
280 02
05
QDECL
TRUE
FALSE
06
A(1) = PNAME(9)
A(2) = PNAME(10)
10
07
XINCL .LT. ANG2 - 1.0-6
TRUE
FALSE
280 02
08
XINCL .EQ. ANG2
TRUE
FALSE
280 01
12
09
B(1) = PNAME(9)
B(2) = PNAME(10)

CHART TITLE - SUBROUTINE QPRINT(IPRINT)

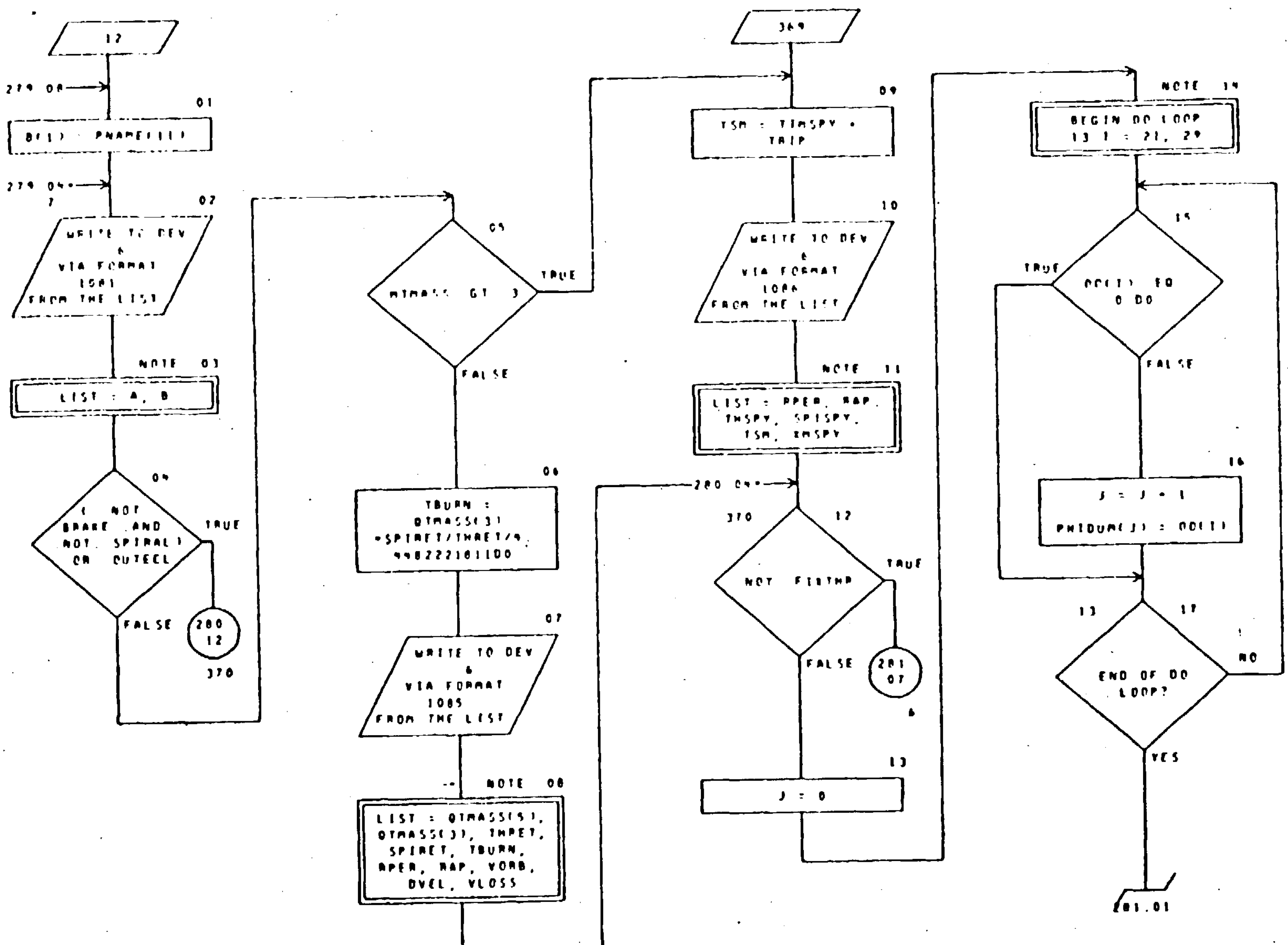


CHART TITLE - SUBROUTINE QPRINT(IPRINT)

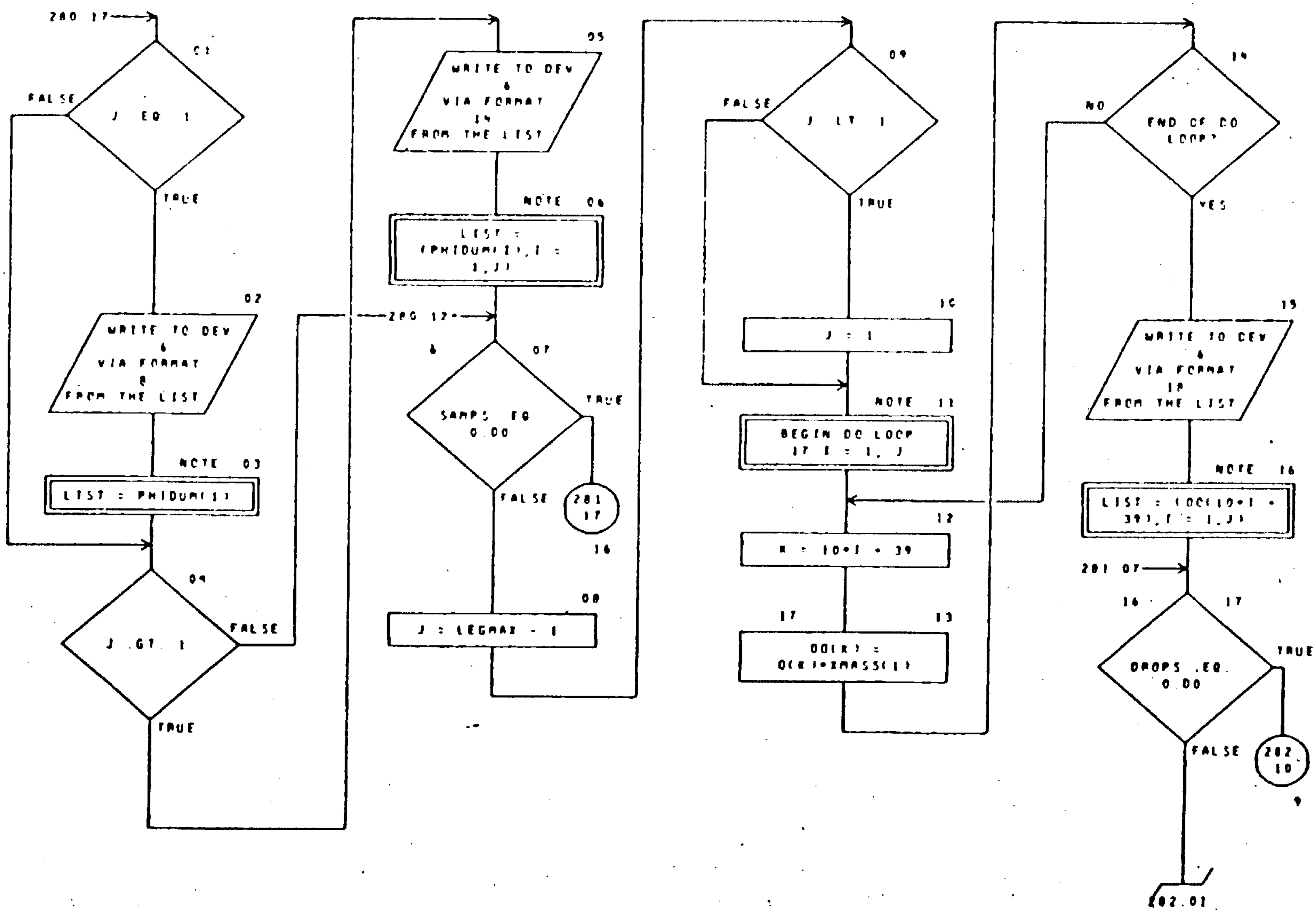
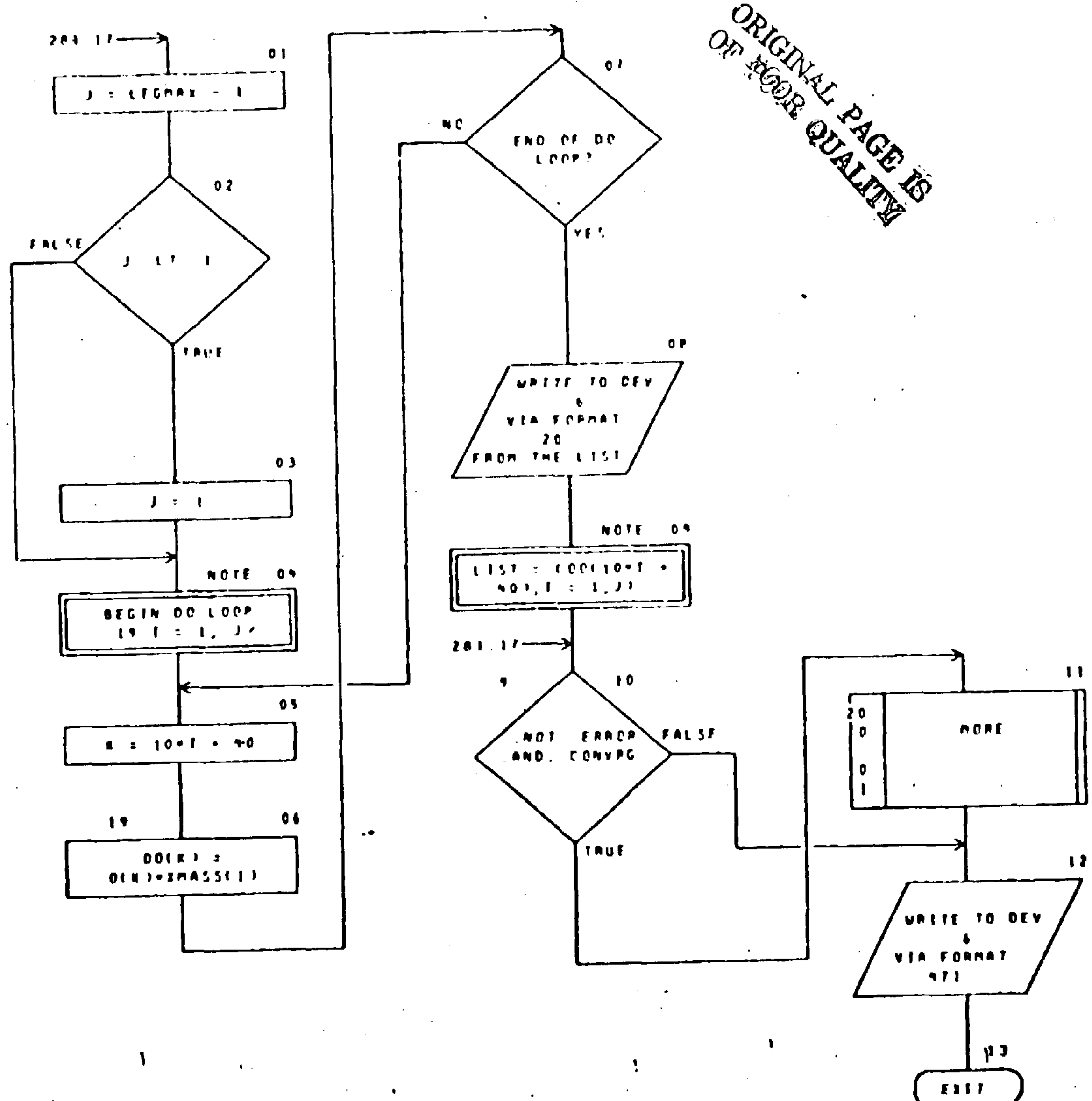


CHART TITLE - SUBROUTINE QPRINT(PRINT)



QPRINT-31

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL ERROR,CONVRG,ODECL,OUTECL        ,FIXTHR,FIXPOW,BRAKE,
SPIRAL,FLYBY,PLANET,ERODE,LOOSE,STAY
DIMENSION        A(2),B(2),PNAME(11),MNAME(12),XH(3)
                 ,PHIDUM(10),JJJ(5)
COMMON /REAL8/ PAYLOD,EMASS(7),R31,CTANK,CSTR,RO1(2),VIMP,
OTMASS(5),RO2(2),PPER,RAP,THRET,SPIRET,DVEL,VLOSS,VORB,RO3(5),
XJLD,TRIP,R32,PT,FFF,PWRM,TMAX,ANGD,RO4(9),XC(7),PO(7),BT,DT,ET,
B1,B2,B3,RO5(7),SAI,R14(4),
                 TPI,PO6(42),                    SEFMA(7),SEFMB(7)
,RO8(19),ANGLE,TBURN,TDATE1,TDATE2,TDATEX,RG9(9),ANG1,ANG2,R27(3),
DECL,R2P,XINCL,ALTAU,P10,RMAX,
RMIN,PMAX,COMDIS,COMANG,R11(6),SPISPY,THSPY,TIMSPY,XMSPY,R12(8),
R(2),R13(63),ALPHAA,ALPHAT,R33(36),
              VOOD(3),R25(6),DEG,R15(2),CONTM,CONSP,R26(208),
TOFF(20),TCOAST(20),R16(435),XJLA,HOURD,HOURA,
C3,C4,DEP,ARR,XSP,R17(124),DPOW,R34(7),SAMPS,DROPS,ROT(5),
RT,R18(6),POWR,R19(33),TPOWER,      R20(61),X(50),R21(53),TAU,
R22(21),XINT(50,5),R23(250),XTINT(6,5),R24(95)
COMMON /INTGR4/ IO7,MODE,IO1(4),MBOOST,IO2(8),NSET(5),IO6,JPP,JT,
MOPT2,MOPT3,IO9(11),MTMASS,NSPEC,IO3(4),NPR(4),
III(7),                            KOUNT,IO5 ,MYEARD,MYEARA,MOND,
MONA,MDAYD,MDAYA,LXX,MXX,IO4(64),LEGMAX,MOPTI(5),IO8(6),MPOW,
I10(860)
COMMON /LOGIC4/ ERROR,CONVRG,FIXPOW,LO1(4),ODECL,LO2,OUTECL,
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      FIXTHR,LO3,FLYBY,BRAKE,LO4(2),SPIRAL,LO5(2),ERODE,LO7(5),PLANET,
      LO6(3),LOOSE,LO8(470)
      COMMON /ITERAT/ BX(5,70),BY(3,70),BC(140),C(70),DD(70),BD2(140)
      COMMON /SOLSYS/ SOLO(140),APL(2,70)
      DATA PNAME /5HORBIT,5HFLYBY,6HRENDEZ,5HWITH ,7HWITHOUT,6HBEFORE,
      5HAFTER,1H ,9HVIOLATIO,1HN,8HAT LIMIT/
      DATA MNAME /3HJAN,3HFEB,3HMAR,3HAPR,3HMAY,3HJUN,3HJUL,3HAUG,3HSEP,
      3HOCT,3HNOV,3HDEC/
30    FORMAT(1H1,4HCASEI3,47X19HITERATOR PARAMETERS)
50    FORMAT(1H0//1H ,53X21HINDEPENDENT VARIABLES//1H ,
      11H NO   INDEX,12X5HVALUE,18X10HSTEP LIMIT,
      14X12HPERTURBATION,16X6HWEIGHT/1H )
221   FORMAT(1H ,I3,I7,3X1P5D25.16)
70    FORMAT(1H0//1H ,23X19HDEPENDENT VARIABLES//1H ,
      11H NO   INDEX,12X5HVALUE,19X9HTOLERANCE/1H )
222   FORMAT(1H ,I3,I7,3X1P2D25.16)
1030  FORMAT(1H1,4HCASEI3,5X11H(CONVERGED),32X19HPERFORMANCE SUMMARY/1H
      )
1031  FORMAT(1H1,4HCASEI3,5X15H(NOT CONVERGED),28X19HPERFORMANCE SUMMARY
      /1H )
1032  FORMAT(1H1,4HCASEI3,5X19H(SINGLE TRAJECTORY),24X19HPERFORMANCE SUM
      MARY/1H )
1007  FORMAT(1H0,21X,24H OUT OF ECLIPTIC MISSION,10X,20H FINAL INCLINATI
      ON =,F8.4,5X,4H DEG)
1006  FORMAT(1H0,20X2A8,4H TO 2A8,1X,A8,1X,A8,20HHIGH THRUST CAPTURE ,
```

ORIGINAL PAGE IS OF POOR QUALITY

QPRINT-33

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
               9HMANEUVER )
1009      FORMAT(1H0,42X2A8,4H TO 2A8,1XA8)
1008      FORMAT(1H0,28X2A8,4H TO 2A8,
              32H WITH FIXED ARRIVAL EXCESS SPEED)
1010      FORMAT(1H+,103X24HAND WITH FORCED COASTING)
1011      FORMAT(1H )
328       FORMAT(1H0,44X19HWITH VISITATION OF 2A8)
329       FORMAT(1H0,36X20HWITH VISITATIONS OF 2A8,5H AND 2A8)
337       FORMAT(1H0,28X20HWITH VISITATIONS OF 2A8,2H, 2A8,5H AND 2A8)
345       FORMAT(1H0,42X10HARRIVAL ATF9 3,6H DAYS A6,1X2A8,11H PERIHELION)
347       FORMAT(1H0,50X11HARRIVAL AT 2A8,11H PERIHELION)
335       FORMAT (1H0,45X31HLAUNCH VEHICLE INDEPENDENT MODE)
1050      FORMAT(1H0,9X5HLD = A4,I2,1H,,I5,1H,,F8 4,10H HOURS GMT,5X,
          5HAD = A4,I2,1H,,I5,1H,,F8.4,10H HOURS GMT,5X14HFLIGHT TIME =
           F9.4,6H DAYS /5X,2(17X12HJULIAN DATE F12.4)/)
1055      FORMAT(1H0,45X,20H        FLIGHT TIME = F9 4 /)
1060      FORMAT(1H0,46X,37HELECTRIC PROPULSION SYSTEM PARAMETERS
          /47X,23HEFFICIENCY COEFFICIENTS/2X,15HALPHA A (KG/KW),5X,
          15HALPHA T (KG/KW),5X,14HTANKAGE FACTOR,5X,16HSTRUCTURE FACTOR,
          18X,1HB,9X,10HD (KM/SEC),7X,1HE/F13.4,F20.4,F19 4,F20.4,F27 5,
          F15.5,F14.5/1H )
1065      FORMAT(1H0,43X,44HELECTRIC PROPULSION SYSTEM MASS SUMMARY (KG) //
           22X,7HINITIAL,6X,11HPOWER PLANT,7X,10HPROPELLANT,10X,7HTANKAGE,
           8X,9HSTRUCTURE,9X8HNET MASS  / 12X,6F17.4 /)
1070      FORMAT(1H0,42X,46HELECTRIC PROPULSION SYSTEM PERFORMANCE SUMMARY
```

ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
              //2X,12HP(1 AU) (KW),4X,12HP(HSRP) (KW),4X,12HP(TARG) (KW),4X,
              13HTHR(1 AU) (N),4X,20HACC(1 AU) (M/SEC**2),4X,9HISP (SEC),4X,
              5HEFFIC,5X,15HCHAP DEG (DAYS)/F13.4,F16.4,F16.4,F16.6,1PD22.6,
              0PF17.3,F11.5,1PD18.7)
 23           FORMAT(1H ,11X13HCONSTRAINED MAX)
 1075         FORMAT(1H0,42X,45HEXTREME TRAJECTORY AND PERFORMANCE CONDITIONS
              //2X13HMAX DIST (AU),4X13HMIN DIST (AU),6X14HMAX POWER (KW),
              4X14HMAX THRUST (N),4X16HBURN TIME (DAYS),4X17HDEGRD TIME (DAYS),
              4X14HTRAV ANG (DEG)/1H ,F12.7,F17.7,F19.6,F21.8,F18.5,F20.5,F20.5/
              1H )
 980          FORMAT (1H0)
 1076         FORMAT(1H ,41X,47HPOWERPLANT    JETTISONED PRIOR TO RETRO MANEUVER)
 1077         FORMAT(1H ,41X,47HTANKAGE MASS JETTISONED PRIOR TO RETRO MANEUVER)
 1078         FORMAT(1H ,42X,44HALL SYSTEMS CARRIED INTO TARGET PLANET ORBIT)
 1079         FORMAT (1H )
 1080         FORMAT(1H0,49X,32HDEPARTURE AND ARRIVAL CONDITIONS //
              2X14HDEP DECL (DEG),6X14HPARK INC (DEG),6X16HDEP VINF (M/SEC),
              6X17HC3 (KM**2/SEC**2),6X16HARR VINF (M/SEC),
              6X17HC4 (KM**2/SEC**2)/1H ,F12.4,F20.4,F22.5,F21.6,F24.5,F21.6)
 1081         FORMAT(1H ,4X2A8,4X2A8)
 1085         FORMAT(1H0,39X,47HHIGH THRUST CAPTURE MANEUVER STAGE AND ORBIT SU,
              5HMMARY // 16X,14HSTRUCTURE (KG),7X,15HPROPELLANT (KG),7X,9HTHRU,
              8HST (LBS),11X,9HISP (SEC),7X,18HBURNING TIME (SEC) / 7X,5F21.4
              // 15X,16HPERIAPSE (RADII),6X,15HAPOAPSE (RADII),5X,9HORBIT VEL,
              8H (M/SEC),9X,15HDEL VEL (M/SEC),5X,16HVEL LOSS (M/SEC) / 7X,
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
        5F21.4 /)
1086    FORMAT(50X,'SPIRAL CAPTURE MANEUVER SUMMARY'///7X,'PERIAPSE(RADII)'
        ,4X,'APOAPSE (RADII)',5X,'THRUST (LBS)',8X,'ISP (SEC)',11X,
        'FLIGHT TIME (DAYS)',2X,'PROPELLANT (KG)'/0P6F20.10//)
8       FORMAT(1H ,20HFIXED THRUST ANGLE =F10.4)
14      FORMAT(1H ,21HFIXED THRUST ANGLES =10F10.3)
18      FORMAT(1H ,20HSAMPLE MASSES (KG) =5F10.2)
20      FORMAT(1H ,20HDROP   MASSES (KG) =5F10.2)
471     FORMAT (1H1)
```

ORIGINAL PAGE IS
OF POOR QUALITY

Name:	QSTART
Calling Arguments:	MSTART
Referenced Sub-programs:	AEINWT, BOOSTR, DATE1, EFM, ETAINT, QPRINT, SETUP, SOLINT, TRJINT, TWINKL
Referenced Commons:	INTGR4, ITERAT, ITER2, LOGIC4, REAL8, SOLSYS
Entry Points:	None
Referencing Sub-programs:	MAIN

Discussion: Subroutine QSTART performs most of the case initialization required to execute the program. It is called a total of five times from the MAIN program, each time with a different value of the calling argument MSTART. On each call, a different portion of the case initialization is accomplished.

The first call to QSTART is performed prior to the reading of any case inputs. The purpose is to position the output tape on which the trajectory is written for the ASTEA program, if requested. Additionally, the variables LCOUNT and QJEX are initialized. On the first case only, a page is ejected on the printer file (unit 6) and a heading is written for one of the remote terminal output files (unit 12).

The second call to QSTART is performed after case inputs have been read and serves many functions, as follows:

(1) scales, on option, the initial values of all adjoint variables such that the initial value of the mass ratio adjoint variable is unity;

(2) defines the number of trajectory legs and, for multiple target missions, stores the orbital elements of the intermediate targets in selected arrays;

(3) on the basis of the numerical values of various input option indicators, defines logical flags which are used for testing throughout the program for the following options: launch vehicle independent mode, fixed thrust angle, launch asymptote declination effects, Earth orbit departure mission with finite high thrust, type of end conditions (extra ecliptic, flyby, capture, rendezvous, etc.), and solar array degradation;

(4) initializes for the high thrust Earth departure mission, if that mission is selected;

(5) initializes for the all ballistic mission, if selected by input option; and

(6) initializes for the extra ecliptic mission, if selected.

On the third call to QSTART, an array of print flags is defined. On the fourth call, the positions and velocities of the launch body at launch and the target body at arrival are evaluated and stored in arrays. Additionally, the initial primer vector and its derivative are rotated about the z-axis through an angle equal to the difference in initial longitude on the current and previous cases. Finally, the reference thrust acceleration, jet exhaust speed and mission duration are converted to internal units and stored.

The primary purpose of the final call is to initialize arrays used by the iterator and to define counters. Before exiting, however, calls are made to subroutines SETUP, BOOSTR, ETAINT, TWINKL and TRJINT to complete the initialization required for the case.

Messages and printout: On the first call, the following heading is written on unit 12:

RUN SUMMARY.

On the second call, if the number of trajectory legs specified through the MOPTX array exceeds the limit of four, the following message is written:

INPUT ERROR. HILTOP CURRENTLY LIMITED TO FOUR TRAJECTORY SEGMENTS.

On the third call, the following heading is written on unit 11:

CASE (I4)

where the parenthesized term defines the format of the case number.

On the fourth call, a check is made to see if both imposed coast phases and a high thrust deep space burn are invoked by input option. Since this combination is not permitted, the following message is written:

FORCED COASTING BYPASSED DURING *TDV* OPTION.

Also, a maximum of three legs is permitted with the deep space burn option. If this limit is exceeded, there results the message:

THIRD INTERMEDIATE TARGET NOT ALLOWED DURING *TDV* OPTION.

On the final call a check is made for a zero fixed cone angle that is flagged as an independent variable. If one is found, the following message is printed:

THRUST ANGLE (I3) IS ZERO. BX AND BY TRIGGERS ((I2)) SET TO ZERO.

Also, if a perturbation step size for an independent variable is inadvertently input zero, the following message appears:

PERTURBATION STEP SIZE IS INPUT ZERO. BY (I2) CORRECTED INTERNALLY.

QSTART EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(35)	S	ITER2	Array of (active) iterator independent-variables.
O(70)	S	ITERAT	Array of iterator independent variables, in program internal units.
X(50)	S	REAL8	Array of trajectory integrated variables.
AE	U	REAL8	Desired final extra-ecliptic eccentricity.
AI	SU	REAL8	Desired final extra-ecliptic inclination, i_f, in radians.
AR	U	REAL8	Desired final extra-ecliptic perihelion distance in AU.
AV	S	REAL8	Desired final spacecraft speed, v_f, in EMOS.
BI	SUA	REAL8	Efficiency coefficient b in equation for efficiency.
BS(35)	S	ITER2	Array of maximum step-sizes for the iterator independent-variables.

QSTART EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
BW(35)	S	ITER2	Array of independent variable weighting factors, corresponding to program input quantities Xi(5).
BX(5, 70)	SU	ITERAT	Iterator independent variable array.
BY(3, 70)	SU	ITERAT	Iterator dependent variable array.
B1	SUA	REAL8	Launch vehicle coefficient, b_1, in kg.
B2	SUA	REAL8	Launch vehicle coefficient b_2, in meters/second.
B3	SUA	REAL8	Launch vehicle coefficient b_3, in kg.
CI	S	REAL8	Cos i_f.
DI	SUA	REAL8	Efficiency coefficient d in equation for efficiency, in km/sec.
EI	SUA	REAL8	Efficiency coefficient e in equation for efficiency.
FT	S	REAL8	Reference thrust acceleration, g, in AU/tau^2.
GM(70)	U	SOLSYS	Array of planetary gravitational constants, in m^3/sec^2.
LL(70)	S	INTGR4	Index set of active iterator independent variables.
MM(70)	S	INTGR4	Index set of active dependent variables.
SI	S	REAL8	Sin i_f.
SX(50)	S	REAL8	Array of trajectory integrated variables corresponding to the start of the current computation step.
VJ	S	REAL8	Jet exhaust speed, c, in EMOS.

QSTART EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
XD(50)	S	REAL8	Array of trajectory dependent variable derivatives, corresponding to X(i).
X0(7)	SU	REAL8	Spacecraft initial state vector, x_0, $\dot{x}_0$, ν_0, where x_0 is in AU, $\dot{x}_0$ is in EMOS, and $\nu_0 = 1$.
AAI	U	REAL8	Desired final extra-ecliptic inclination, in degrees.
ABX(70)	S	LOGIC4	Master array of iterator independent variable logical indicators.
ABY(70)	S	LOGIC4	Master array of iterator dependent variable logical indicators.
BBB(35)	S	ITER2	Array of iterator independent-variable perturbation step-sizes.
CNI	SUA	REAL8	Inclination to ecliptic of primary-target orbit, in degrees.
DEG	S	REAL8	Radians to degrees conversion factor.
ECI	SUA	REAL8	Eccentricity of primary-target orbit.
ETH(3)	S	REAL8	Thrust unit vector.
FXL(70)	S	ITERAT	Array of iterator dependent-variable values, in program internal units.
GAP	U	REAL8	Propulsion-corner proximity tolerance-interval.
IRL	U	INTGR4	Primer-origin-proximity step-size-control indicator.
LXX	SU	INTGR4	The number of (active) iterator independent-variables.

QSTART EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
MXX	SU	INTGR4	Value of second subscript of BX array associated with current independent variable being analyzed.
NPR(4)	SU	INTGR4	Printout control vector.
OMI	SUA	REAL8	Ascending node angle (with respect to vernal equinox) of primary-target orbit, in degrees.
QPR	S	LOGIC4	Logical flag indicating whether formatted table of active iterator parameters is to be printed for the current case.
SAI	SUA	REAL8	Semi-major axis of primary-target orbit, in AU.
SOI	SUA	REAL8	Argument of perihelion of primary-target orbit, in degrees.
TDV	U	REAL8	Time of deep space burn, in days.
TPI	SUA	REAL8	Time from reference date to perihelion passage of primary target, in days.
AXIS(3)	S	REAL8	Spacecraft spin-axis unit vector (not used at present).
CNIX(5)	SUA	REAL8	Inclinations to ecliptic of intermediate-target orbits, in degrees.
CONX(70)	U	ITERAT	Array of print conversion factors for iterator independent-variables.
CONY(70)	U	ITERAT	Array of print conversion factors for iterator dependent variables.
ECIX(5)	SUA	REAL8	Eccentricities of intermediate target orbits.
ETHD(3)	S	REAL8	Thrust unit-vector time-derivative, $\dot{\bar{e}}_t$, in tau^{-1}.

QSTART-7

QSTART EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
HOUR	SUA	REAL8	Hour-of-day of reference date.
IBAL	U	INTGR4	Ballistic trajectory option indicator.
IOUT	U	INTGR4	Extra-ecliptic mission indicator.
IROT	U	INTGR4	Initial primer vector rotation indicator.
LINE	SU	INTGR4	Current number of lines which have been printed on unit 11 during the current computer run.
MDAY	SUA	INTGR4	Day-of-month of reference date.
MODE	S	INTGR4	Power variation option selector.
MPOW	U	INTGR4	Maximum or optimum power indicator during solar panel degradation option.
NSET(5)	SU	INTGR4	Iteration-sequence control array.
OMIX(5)	SUA	REAL8	Ascending node angles of intermediate-target orbits, in degrees.
QJEX	S	LOGIC4	Detailed printout indicator.
QMAX(35)	S	ITER2	Array of upper allowable values for the iterator dependent-variables.
QMIN(35)	S	ITER2	Array of lower allowable values for the iterator dependent-variables.
SAIX(5)	SUA	REAL8	Semi-major axes of intermediate-target orbits, in AU.
SOIX(5)	SUA	REAL8	Arguments of perihelion of intermediate-target orbits, in degrees.
TMAX	S	REAL8	Integration stopping condition; time of flight from launch to primary target, in tau.

QSTART EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
TOFF(20)	S	REAL8	Array of times, from the start of the trajectory, at which imposed coast phases are to begin, in days.
TPIX(5)	SUA	REAL8	Times from reference date to perihelion passage, for the intermediate targets, in days.
VIMP	S	REAL8	Launch hyperbolic excess speed, $v_{\infty o}$, in AU/tau (=EMOS).
WIRL	S	LOGIC4	Primer-origin proximity step-size control logical indicator.
BRAKE	S	LOGIC4	Retro stage logical indicator.
CONTM	U	REAL8	Tau to days conversion factor.
FLYBY	S	LOGIC4	Flyby mission logical indicator.
INTER(5)	S	INTGR4	Array of indices which select the correct orbital elements for the intermediate targets.
KOUNT	U	INTGR4	Case counter.
LOADX	U	INTGR4	Indicator for invoking the intermediate-target initial-guess feature.
MONTH	SUA	INTGR4	Month-of-year of reference date.
MOPTX(5)	SUA	INTGR4	The target-numbers of the successive intermediate targets.
MOPT2	SUA	INTGR4	Launch planet-number and ephemeris-option indicator.
MOPT3	SUA	INTGR4	Planet-number of primary target.
MPERF	S	INTGR4	Indicator which selects the quantities to be optimized when the iterator is operating in the improve mode.

QSTART EXTERNAL VARIAPLES TABLE (cont)

Variable	Use	Common	Description
MYEAR	SUA	INTGR4	Year of reference date.
NPERF	U	INTGR4	Identification number of the end condition that is to be used as the performance index when using the direct parameter optimization feature.
NTAPE	U	INTGR4	Specifies the unit-number for the ASTEA trajectory tape.
PCURV	S	LOGIC4	Logical flag indicating whether arrays are currently oriented to produce maximum power.
QDECL	S	LOGIC4	Logical flag indicating whether launch asymptote declination effects are to be included in simulation.
RPOW0	U	REAL8	Solar distance at which the power function γ passes through zero, $r_{\gamma=0}$, in AU.
SEFMA(7)	SUA	REAL8	Array containing position and velocity of launch planet at launch time, P_o and $\dot{P}_o$, in AU and EMOS, respectively.
SEFMB(7)	SUA	REAL8	Array containing position and velocity of primary target at time of target intercept, P_n and $\dot{P}_n$, in AU and EMOS, respectively.
SEFMC(7)	SUA	REAL8	Time derivative of SEFMA array (w.r.t. tau).
SEFMD(7)	SUA	REAL8	Time derivative of SEFMB array (w.r.t. tau).
STATE(6)	SU	REAL8	Array containing the Cartesian position and velocity components of the primary target, in AU and EMOS.
SUNMU	U	REAL8	Gravitational constant of the sun, in m^3/sec^2.

QSTART EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
TDELV	S	REAL8	Time of deep space burn, in tau.
TRACK	S	LOGIC4	Indicator for trajectory long block printout (at each computation step).
XLOAD	S	LOGIC4	Logical flag indicating if continuity at intermediate targets in the adjoint variables is to be assumed as a first guess.
XLONG	U	REAL8	Initial heliocentric longitude of final trajectory of preceding case.
ALTITU	U	REAL8	Circular orbit altitude, h, in nautical miles.
CONVRG	S	LOGIC4	Logical flag indicating whether iterator convergence was achieved.
EMUODD	SUA	REAL8	Gravitational constant of the primary target, in m^3/sec^2.
EMUODX(5)	SUA	REAL8	Gravitational constants of intermediate targets, in m^3/sec^2 (not used at present).
FIXPOW	S	LOGIC4	Logical flag indicating whether launch vehicle independent mode is invoked.
FIXTHR	S	LOGIC4	Logical flag indicating whether fixed thrust cone angle is invoked.
FPSNMH	U	REAL8	Conversion factor, knots to fps.
LAUNCH	U	INTGR4	Launch mode selector.
LCOUNT	S	INTGR4	Counter of the number of integrated points along the trajectory.
LEGMAX	SU	INTGR4	Total number of legs (trajectory segments) between start of trajectory and primary target; equals the number of targets up to and including the primary target.

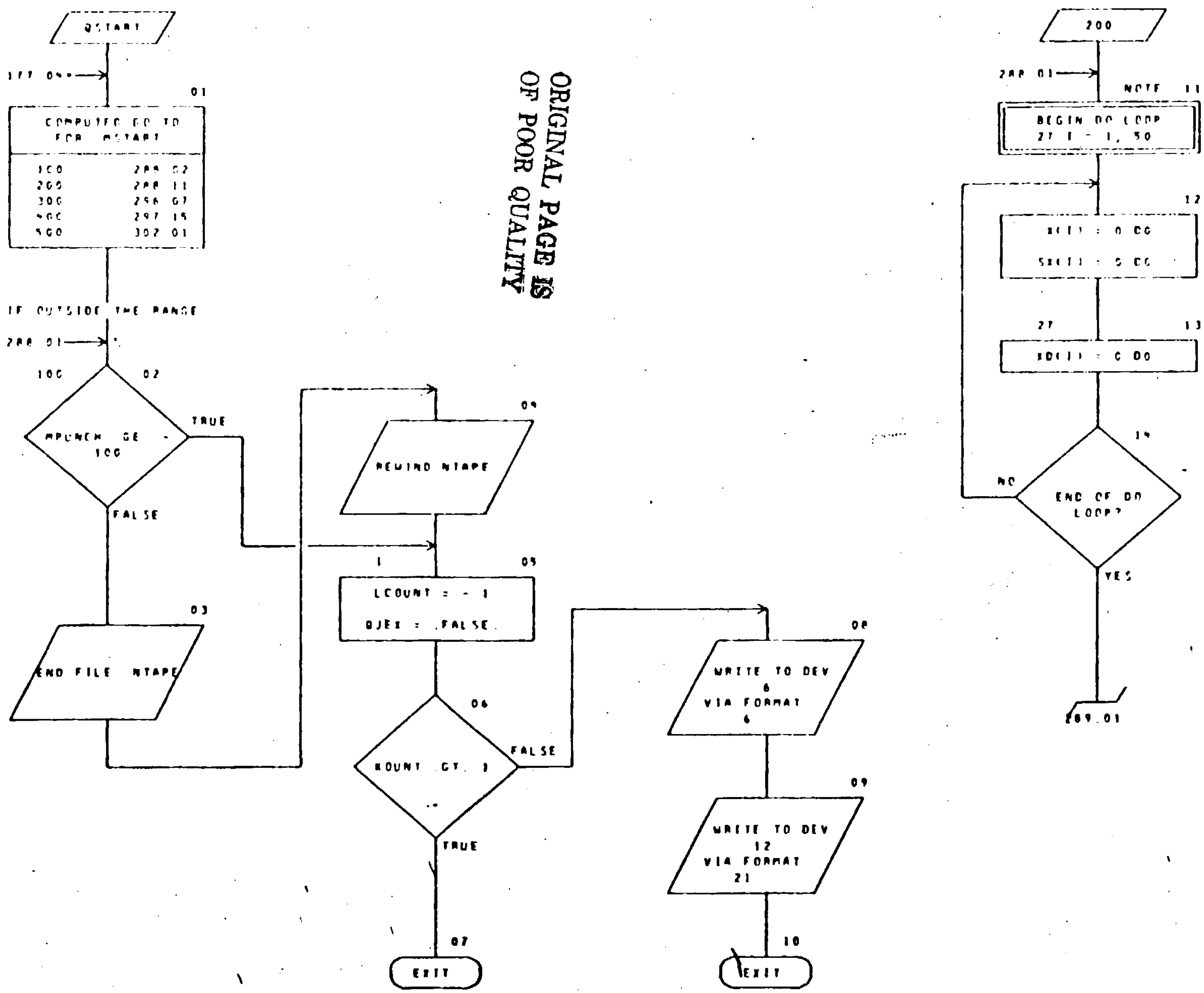
QSTART EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
MAXPOW	S	LOGIC4	Logical indicator of maximum or optimum power during solar panel degradation option.
MBOOST	SUA	INTGR4	Launch vehicle selector.
MPRINT	S	INTGR4	Indicator for printing the summary-trajectory (final case trajectory) as a function of time.
MPUNCH	U	INTGR4	Punched-card and trajectory-tape generation control indicator.
MSTART	XU		Calling argument identifying which section of code to execute.
MTMASS	S	INTGR4	Mission-type selector pertaining to the primary target.
NORMAL	U	INTGR4	Automatic adjoint-variable normalization indicator.
NPRINT	U	INTGR4	Printout amount selection indicator.
OUTECL	S	LOGIC4	Logical flag indicator for out-of-ecliptic mission.
PLANET	S	LOGIC4	Logical flag indicator for ephemeris option.
POSVEL	S	LOGIC4	Logical flag indicator for flyby mission with constrained intercept speed.
POWFIX	U	REAL8	Launch-vehicle-independent (i.e., no launch vehicle) trajectory option indicator, in which the value of POWFIX is the spacecraft's reference power in kilowatts.
QVLOSS	S	LOGIC4	Logical flag indicator for finite thrust, Earth orbit escape mission.

QSTART EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
RADODD	SUA	REAL8	Radius of primary target, in meters.
RADODX (5)	SUA	REAL8	Radii of intermediate targets, in meters (not used at present).
REGION	S	LOGIC4	Logical flag indicating whether spacecraft is currently below the critical solar distance.
RENDEZ	S	LOGIC4	Logical flag indicator for comet or asteroid rendezvous mission.
RPMAX0	U	REAL8	Solar distance at which power function γ peaks, r_{peak}, in AU.
RTSWIT	S	REAL8	Critical solar distance corresponding to a special point in the solar power curve, in AU.
SPIRAL	S	LOGIC4	Logical flag indicator for spiral capture option.
TDATEX	SUA	REAL8	Reference Julian date, less 2400000, defined by program input quantity MYEAR, etc.
TDATE1	SUA	REAL8	Launch Julian date, less 2400000.
TDATE2	SUA	REAL8	Julian date at time of primary-target intercept, less 2400000.
TSCALE	U	REAL8	Iterator dependent-variable tolerance-interval scaling factor.
TUDFLG	S	LOGIC4	Logical flag indicator for two dimensional simulation of missions.
VELOSS	S	LOGIC4	Logical flag indicator for inclusion of finite burn effects in analysis of retro stage maneuver.
WONDER	S	LOGIC4	Logical flag indicator for switch corner monitoring option.

CHART TITLE - SUBROUTINE QSTART(MSTART)



QSTART-13

CHART TITLE - SUBROUTINE QSTART(MSTART)

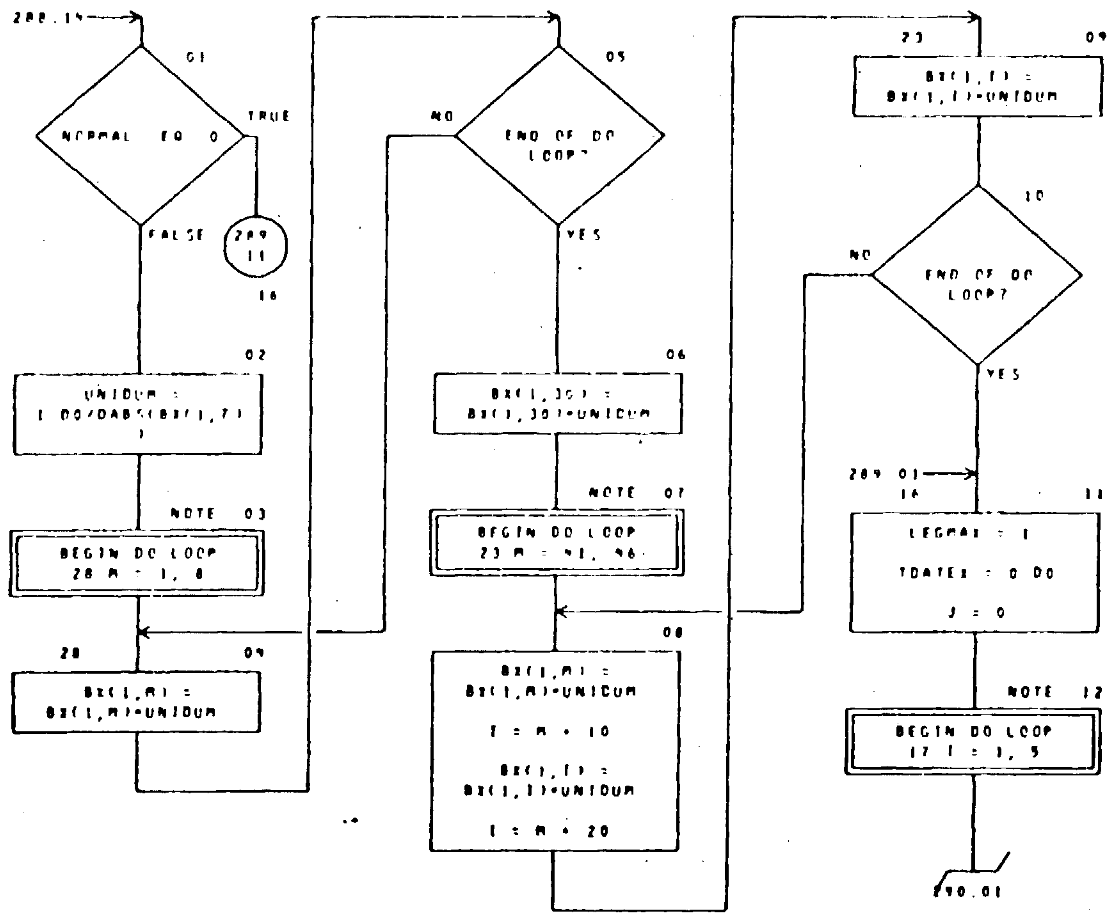
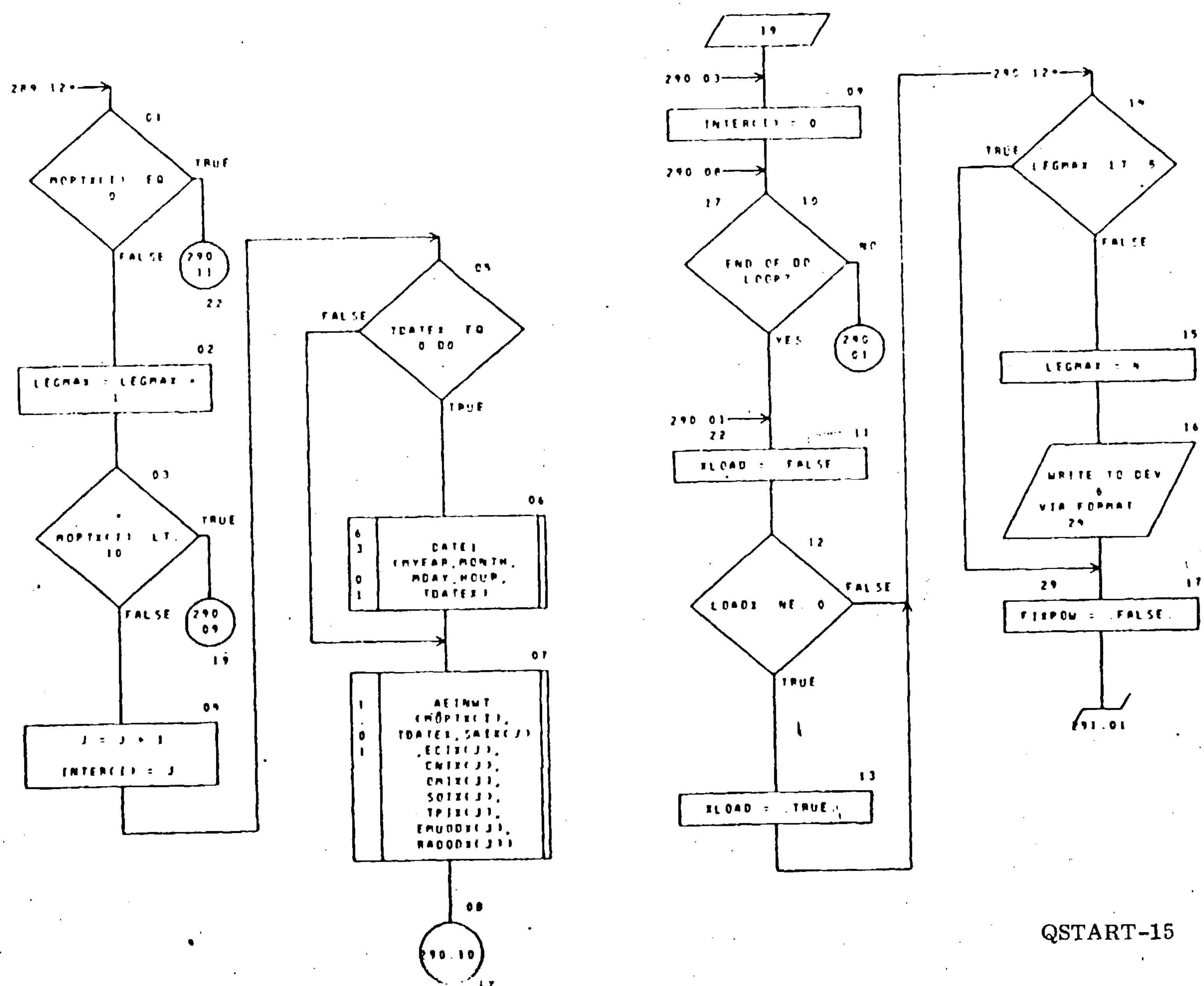


CHART TITLE - SUBROUTINE QSTART(MSTART)



QSTART-15

CHART TITLE - SUBROUTINE QSTART(MSTART)

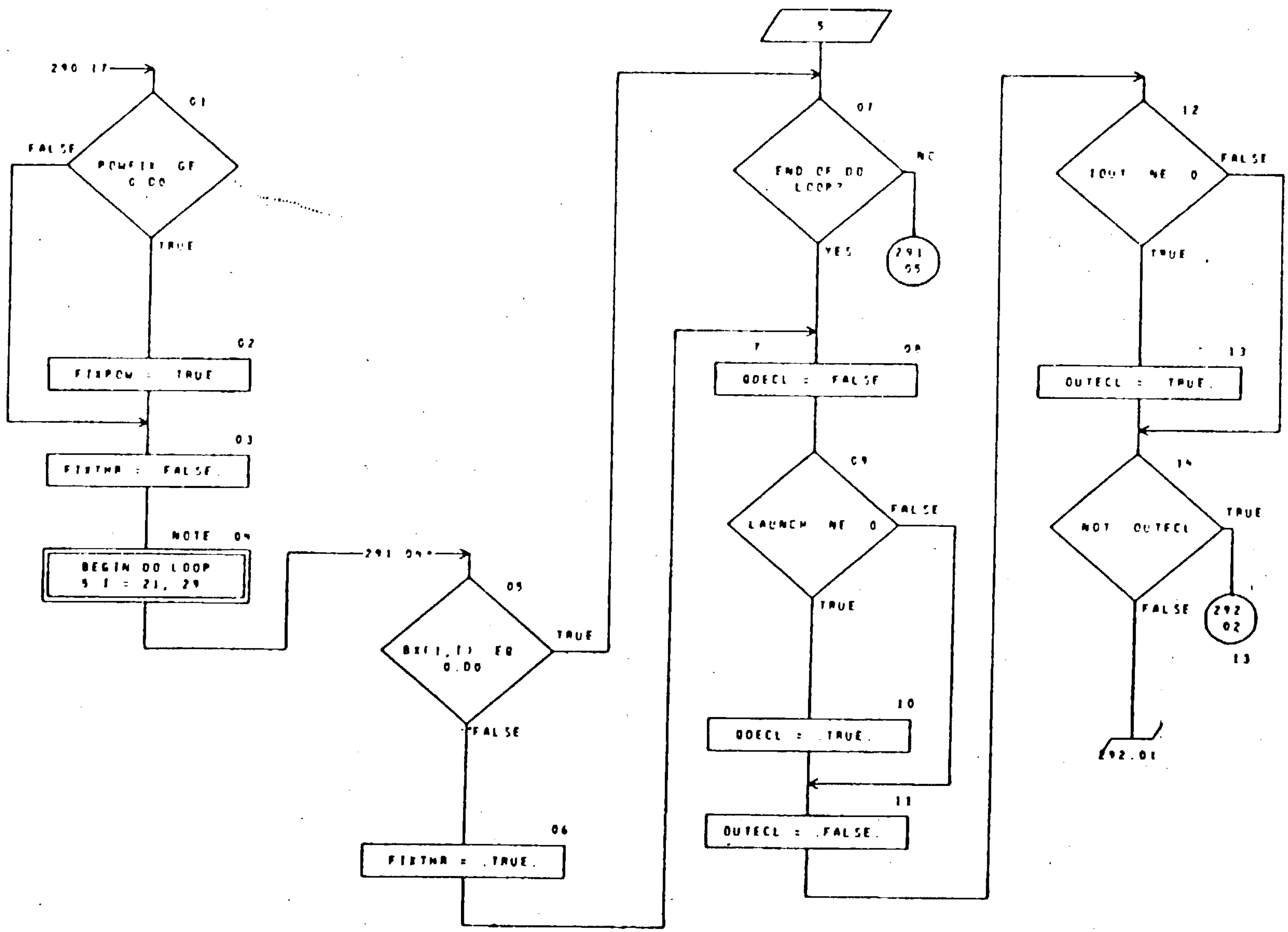
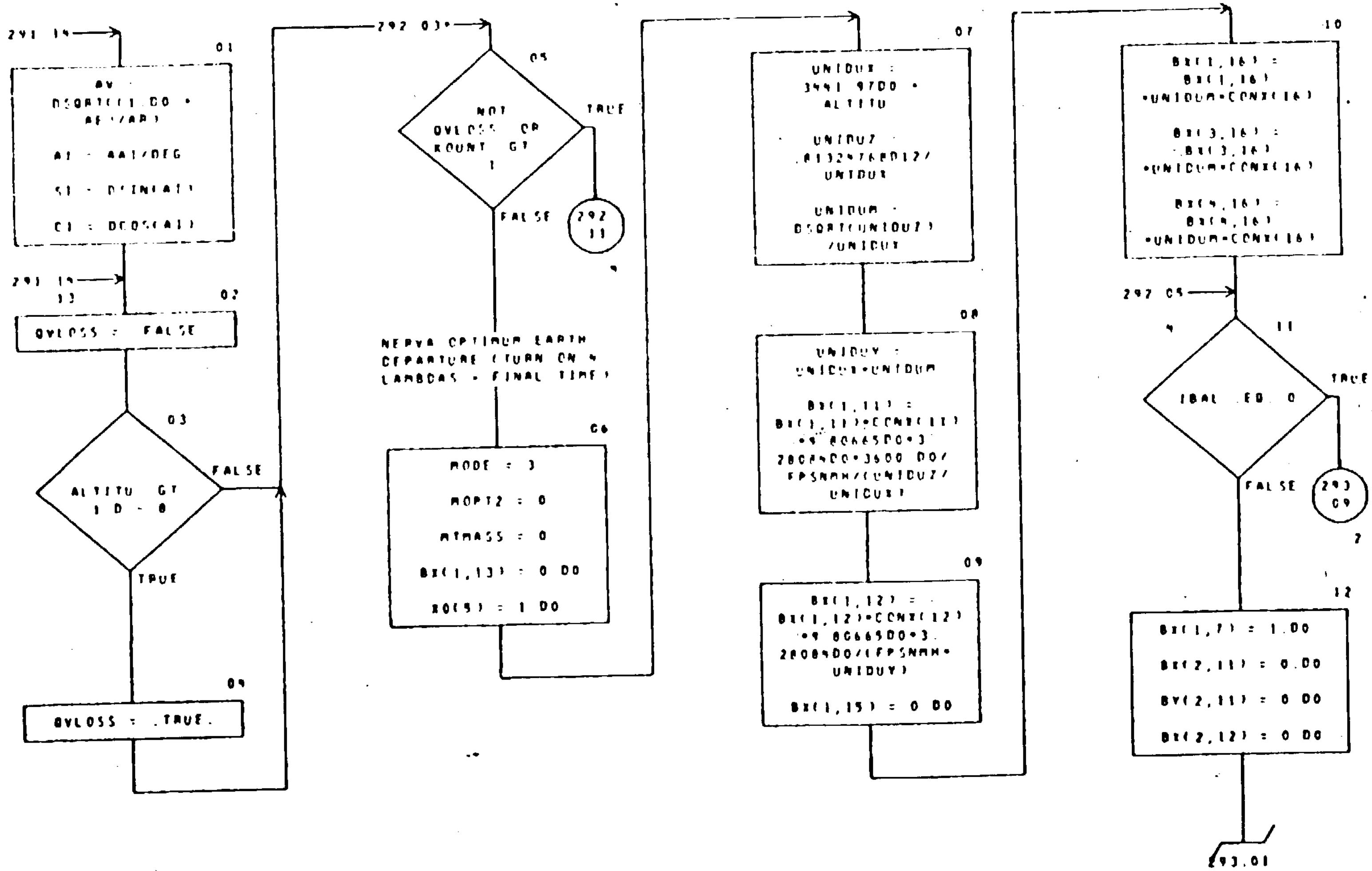


CHART TITLE - SUBROUTINE QSTART(MSTART)



QSTART-17

CHART TITLE - SUBROUTINE QSTART(MSTART)

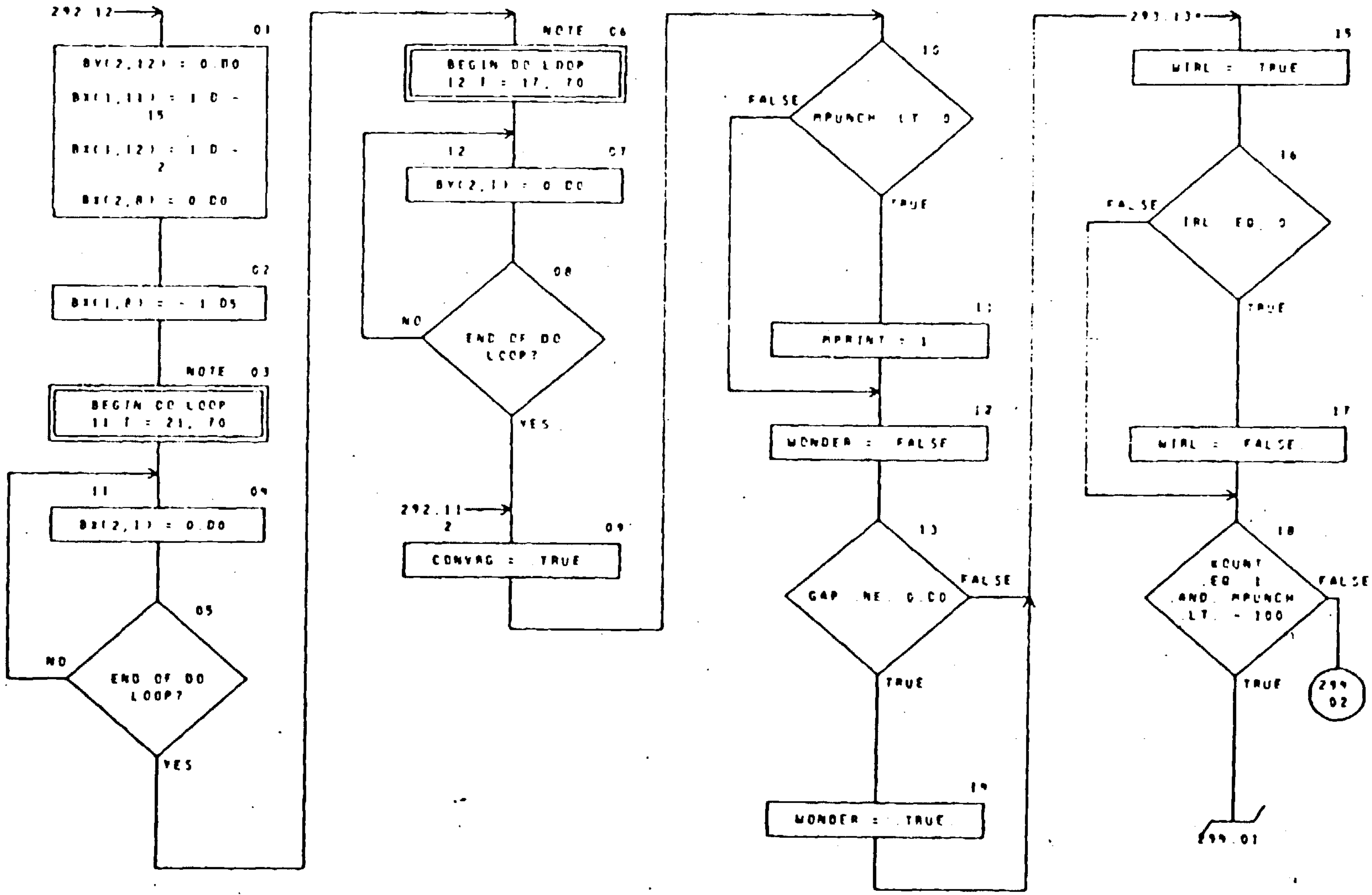
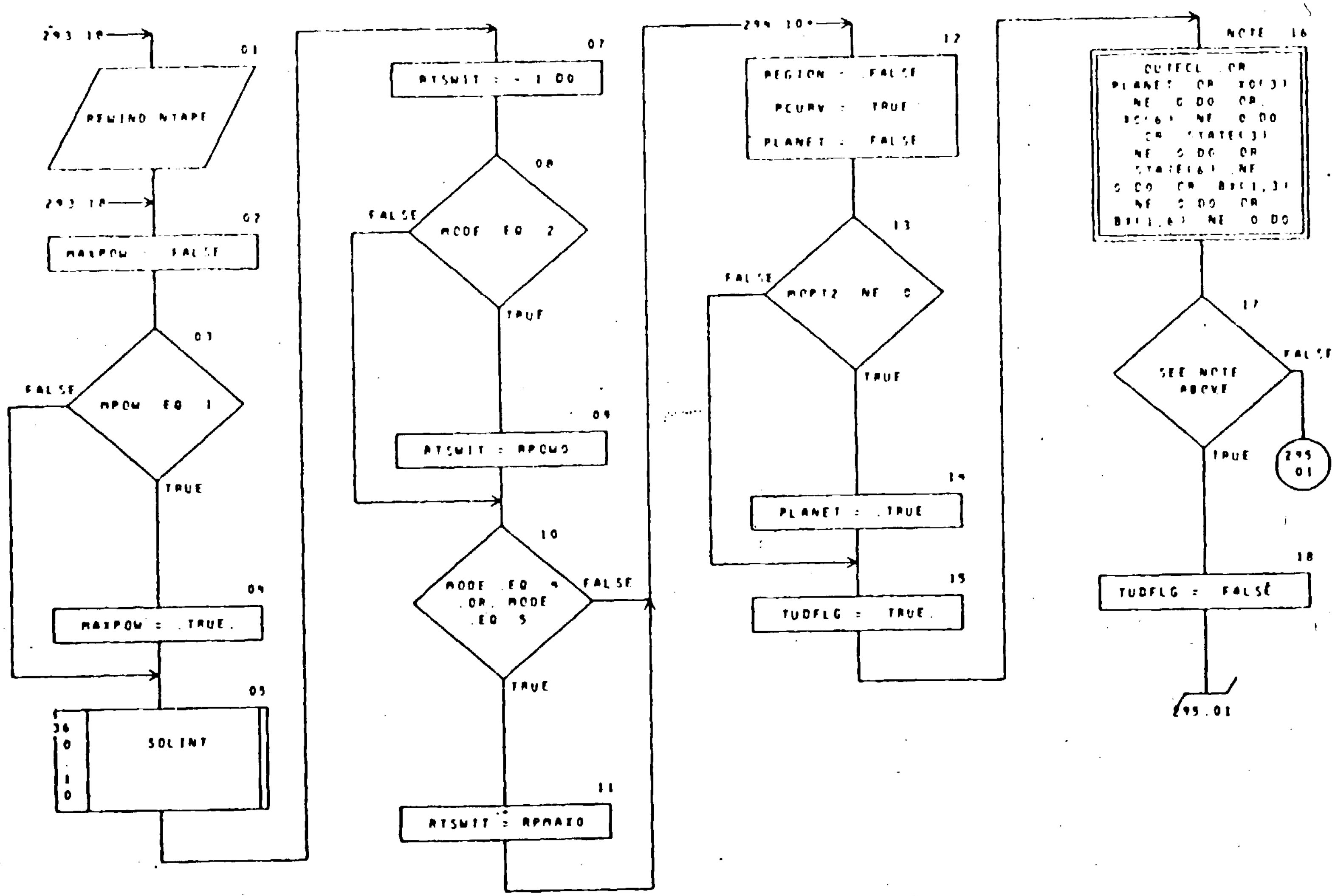


CHART TITLE - SUBROUTINE QSTART(MSTART)



ORIGINAL PAGE IS OF POOR QUALITY

QSTART-19

CHART TITLE - SUBROUTINE OSTART(MSTART)

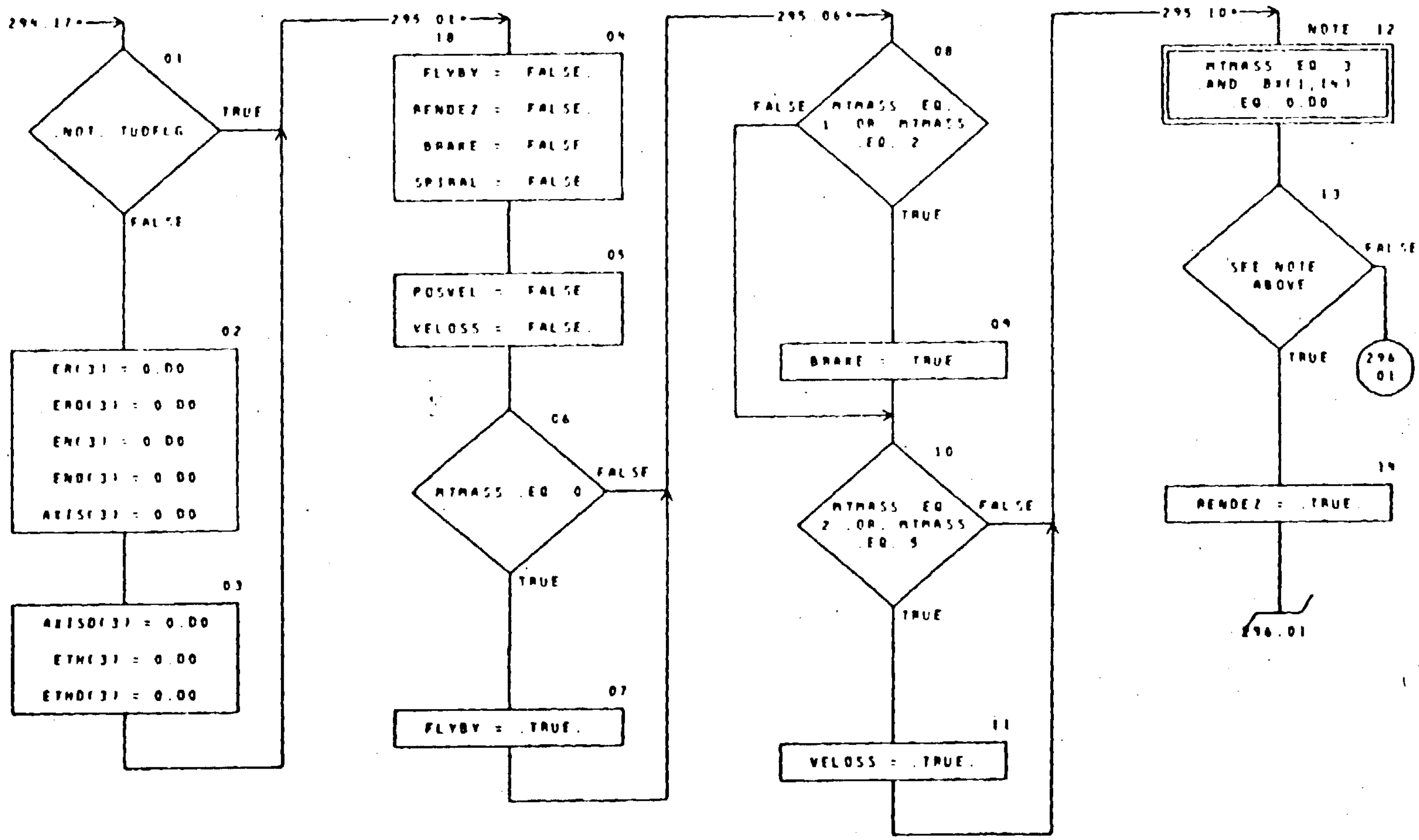
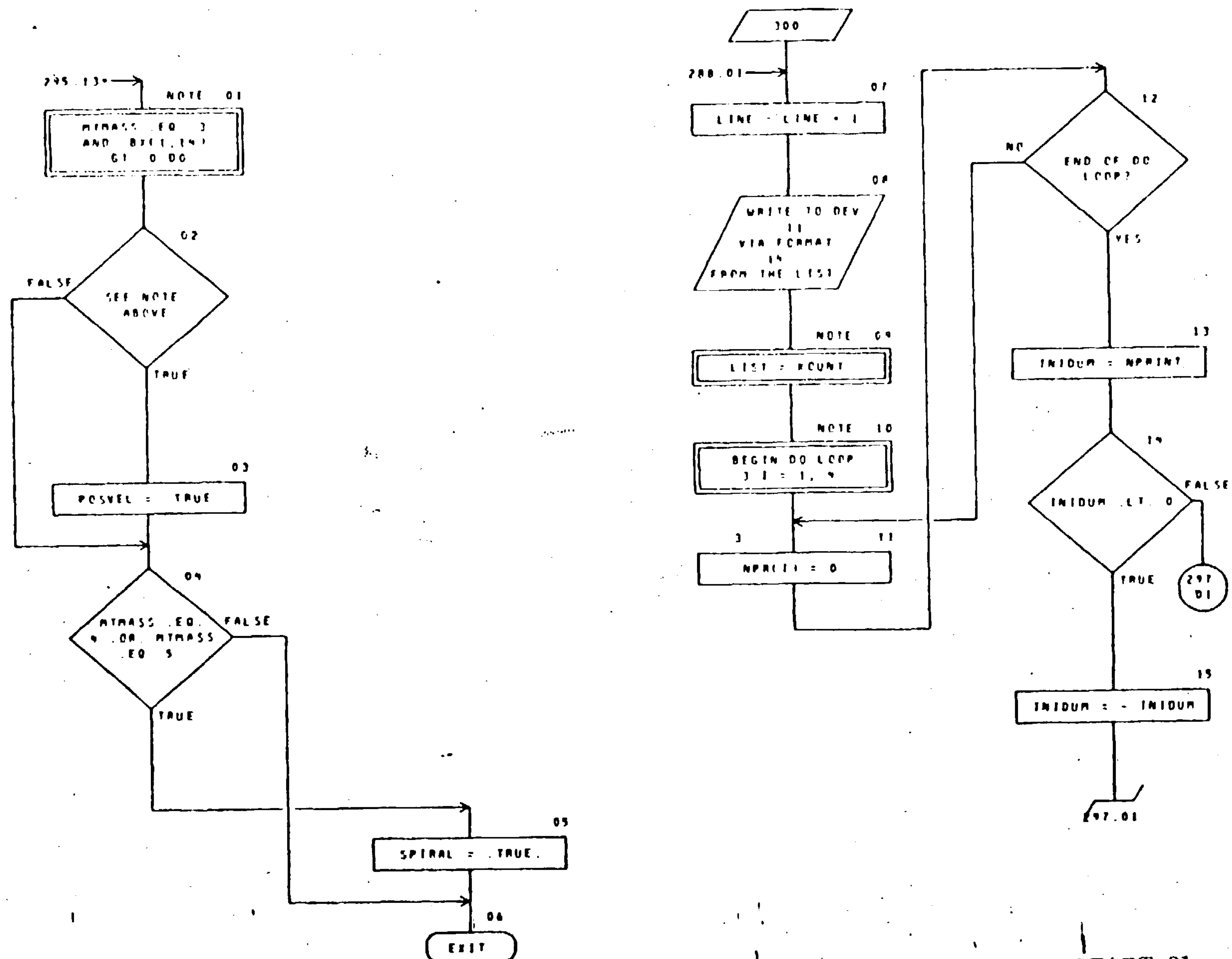


CHART TITLE - SUBROUTINE QSTART(MSTART)



QSTART-21

CHART TITLE - SUBROUTINE QSTART(MSTART)

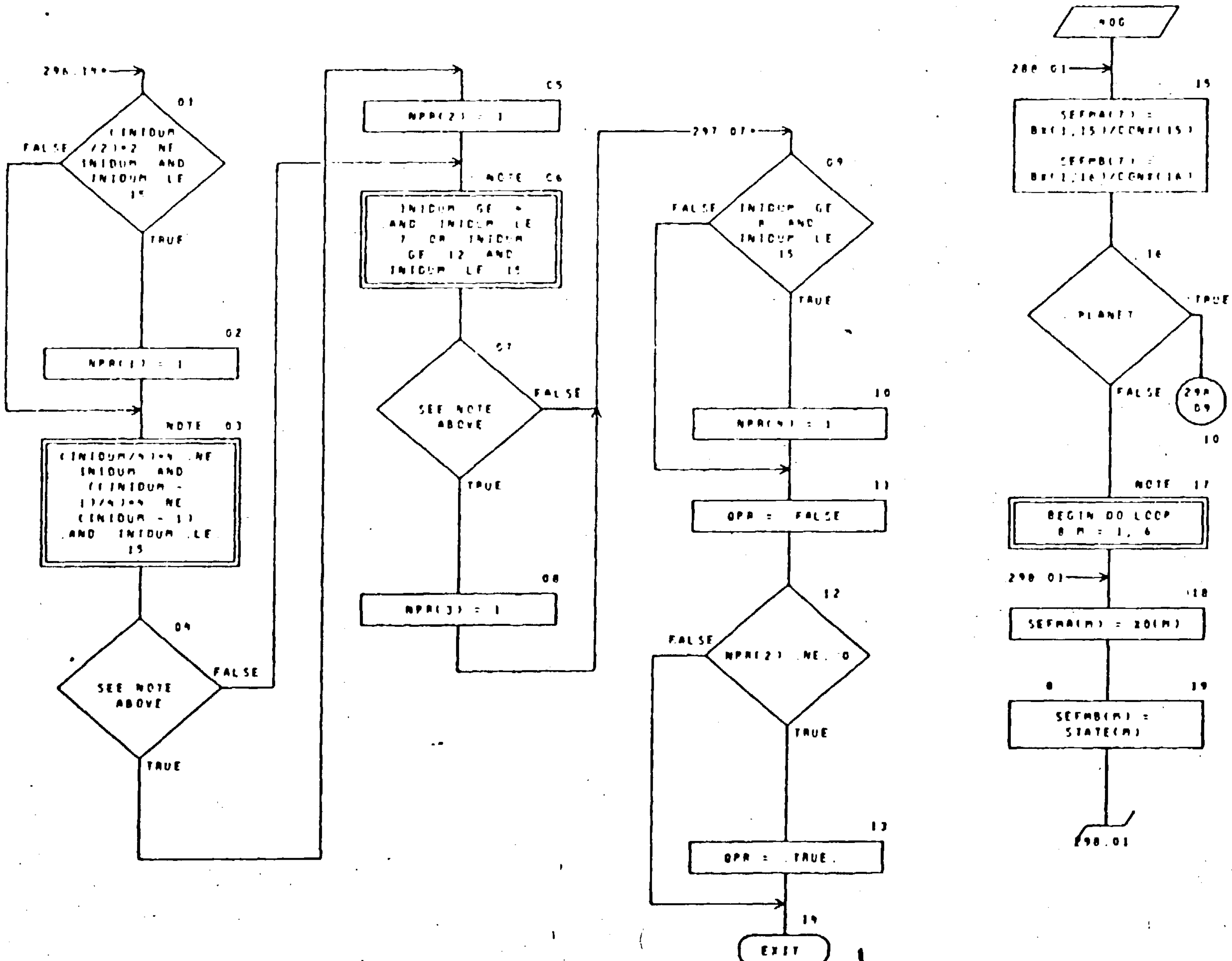
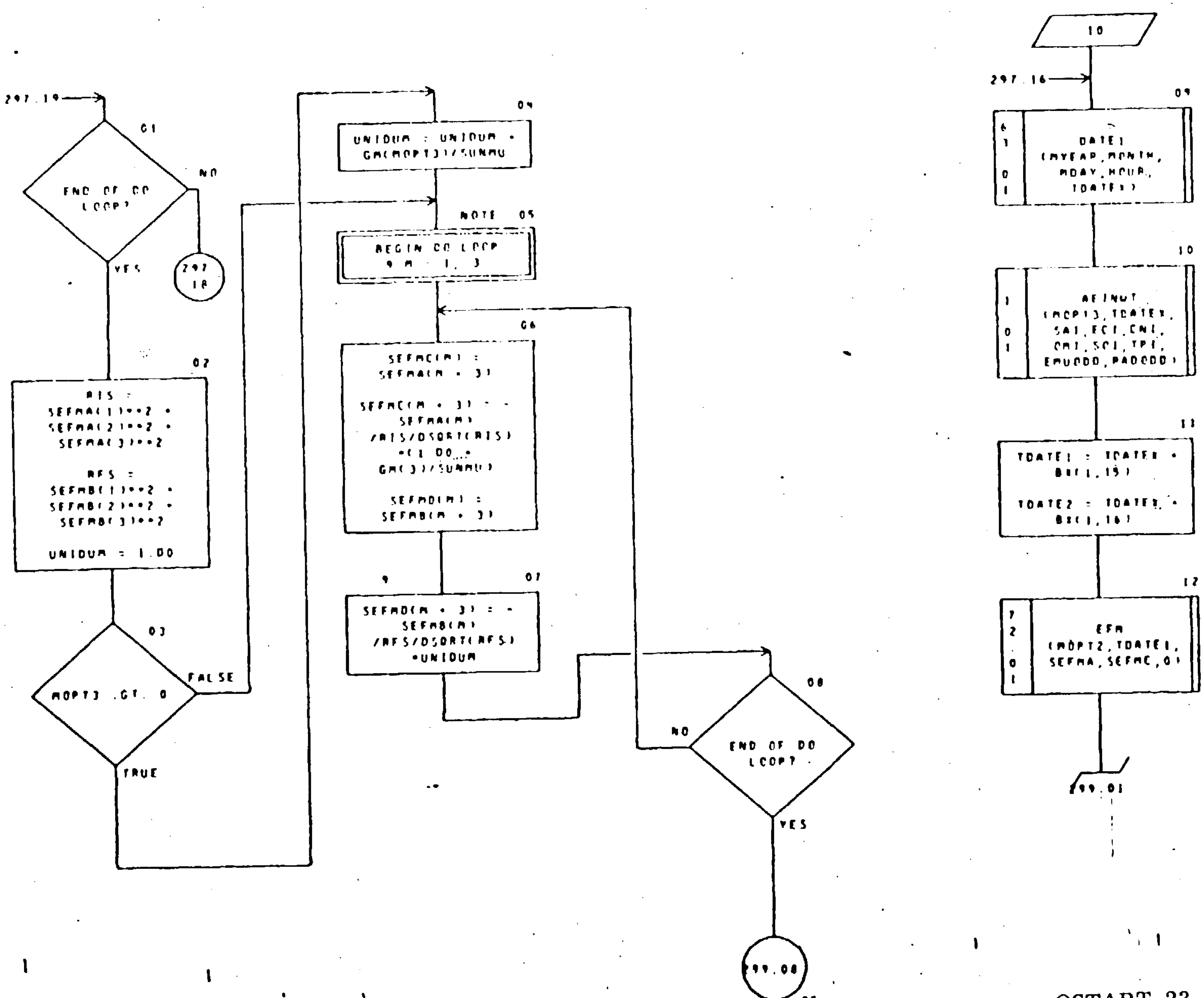


CHART TITLE - SUBROUTINE QSTART(MSTART)



QSTART-23

CHART TITLE - SUBROUTINE OSTART(MSTART)

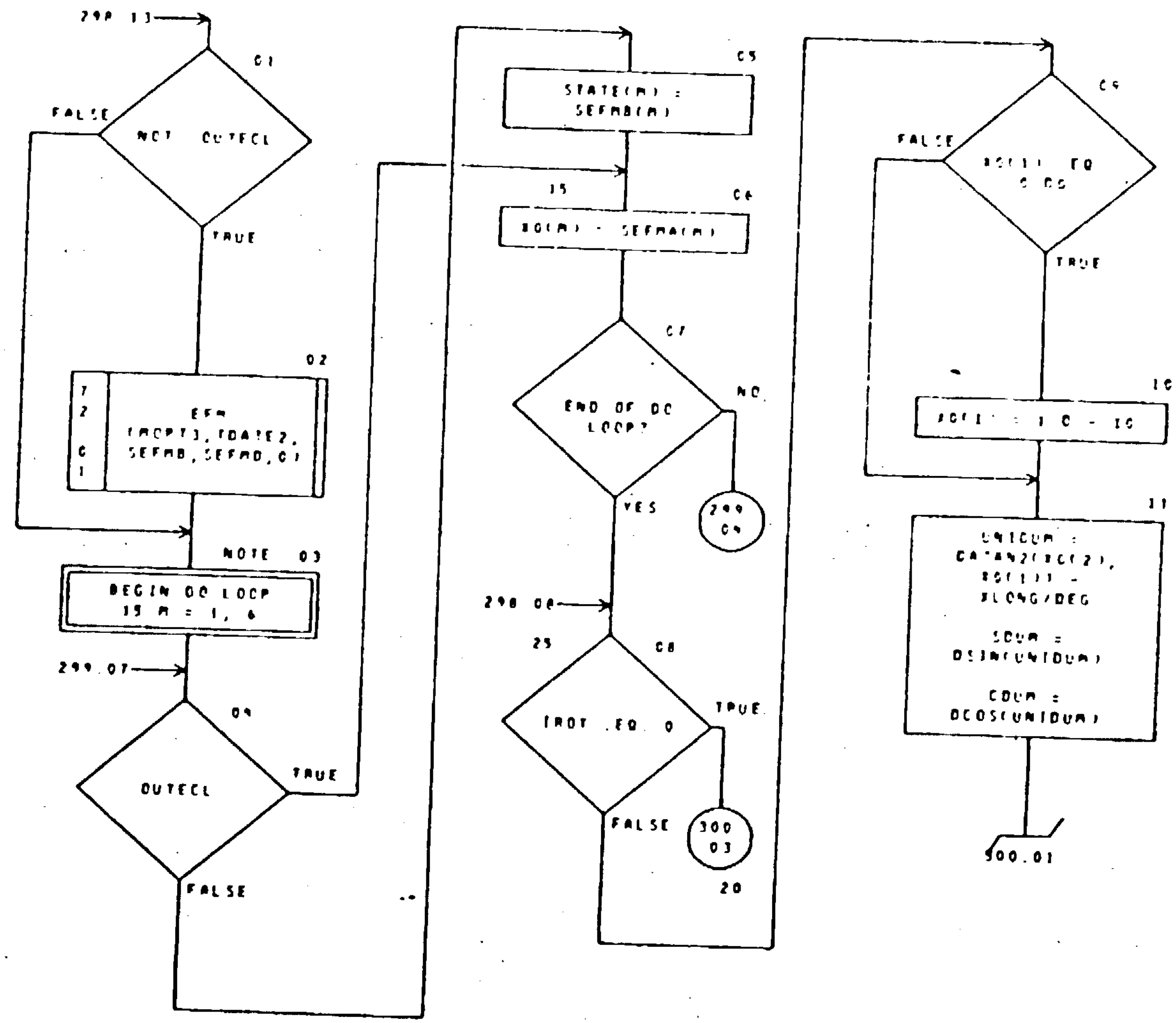


CHART TITLE - SUBROUTINE QSTART(MSTART)

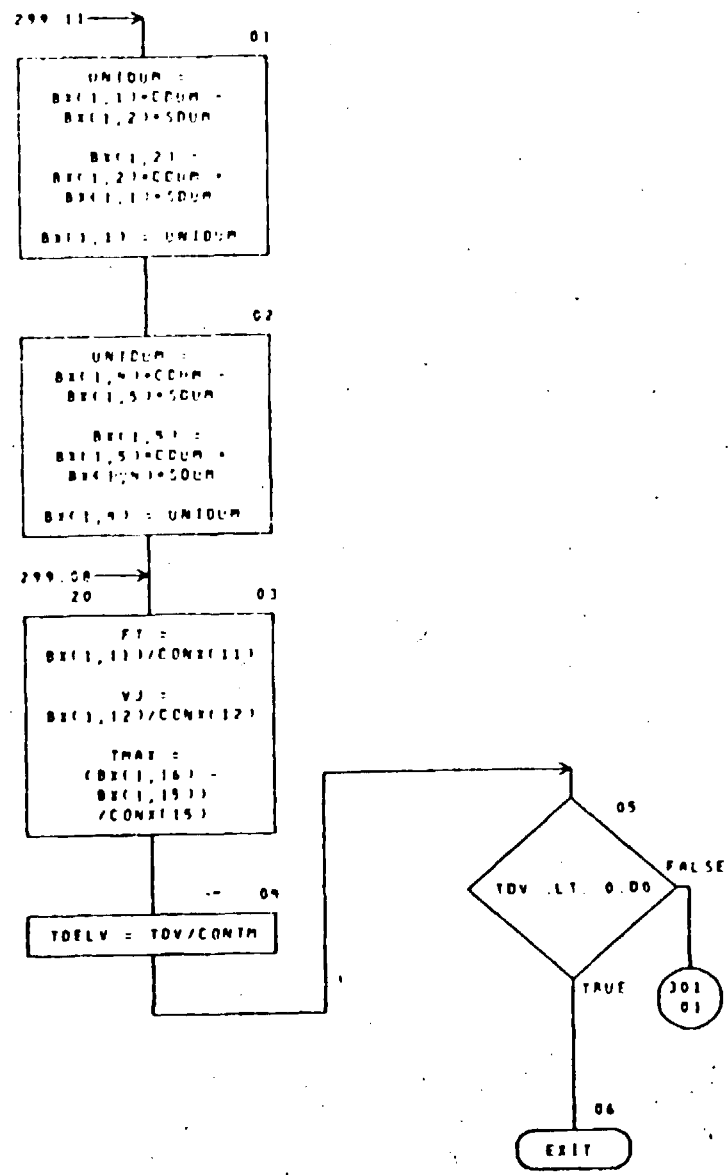


CHART TITLE - SUBROUTINE QSTART(MSTART)

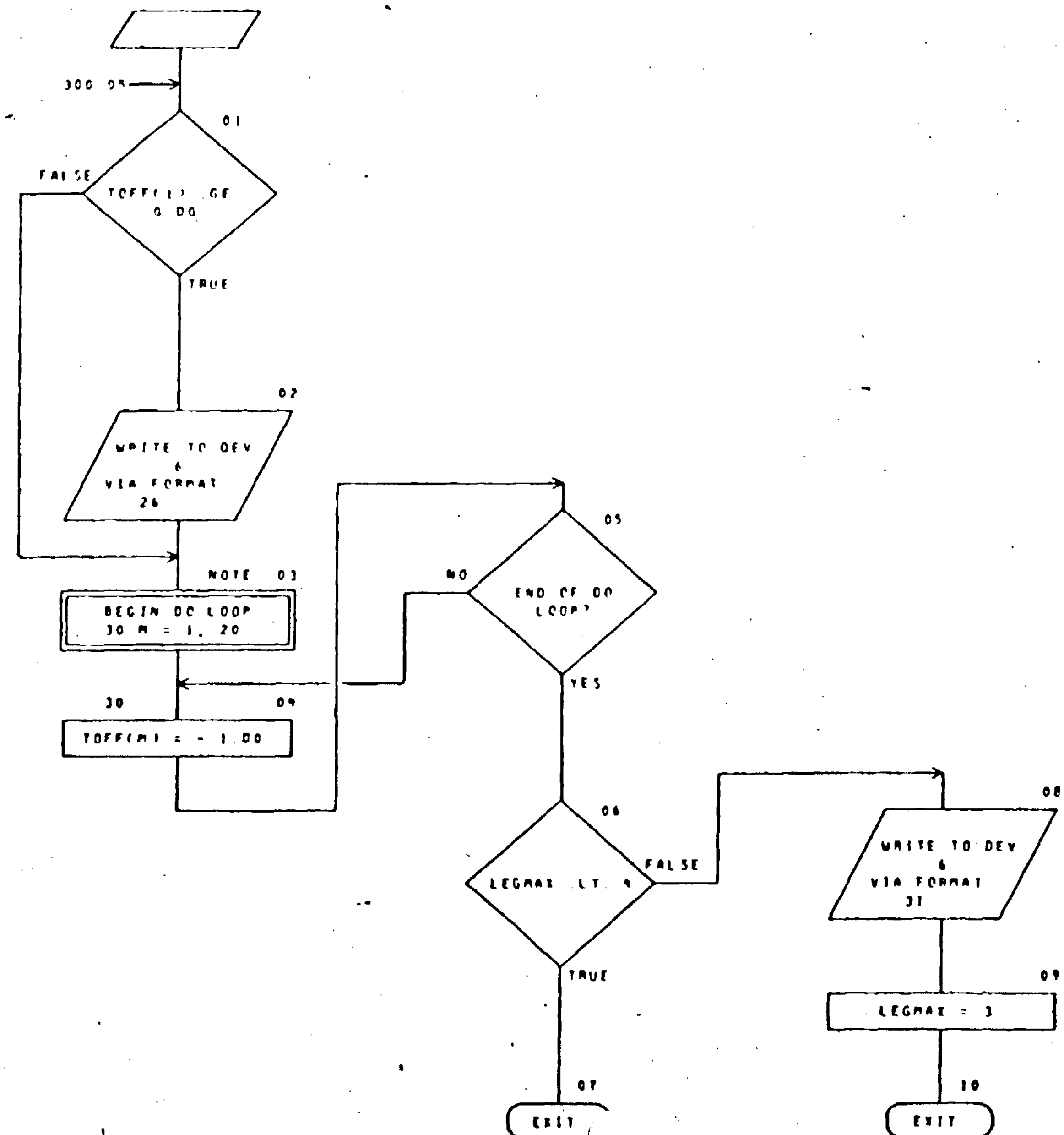


CHART TITLE SUBROUTINE QSTART(MSTART)

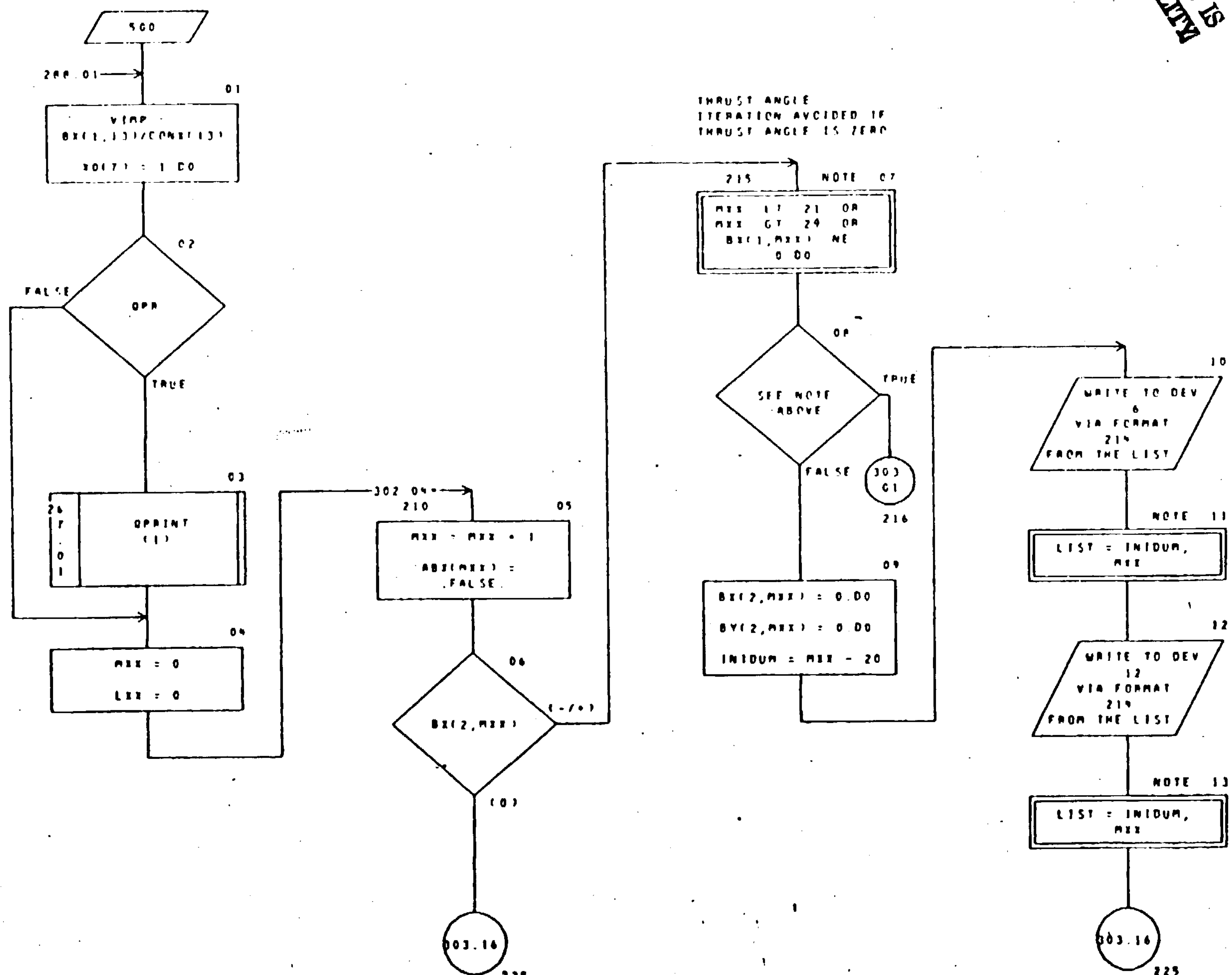


CHART TITLE - SUBROUTINE QSTART(MSTART)

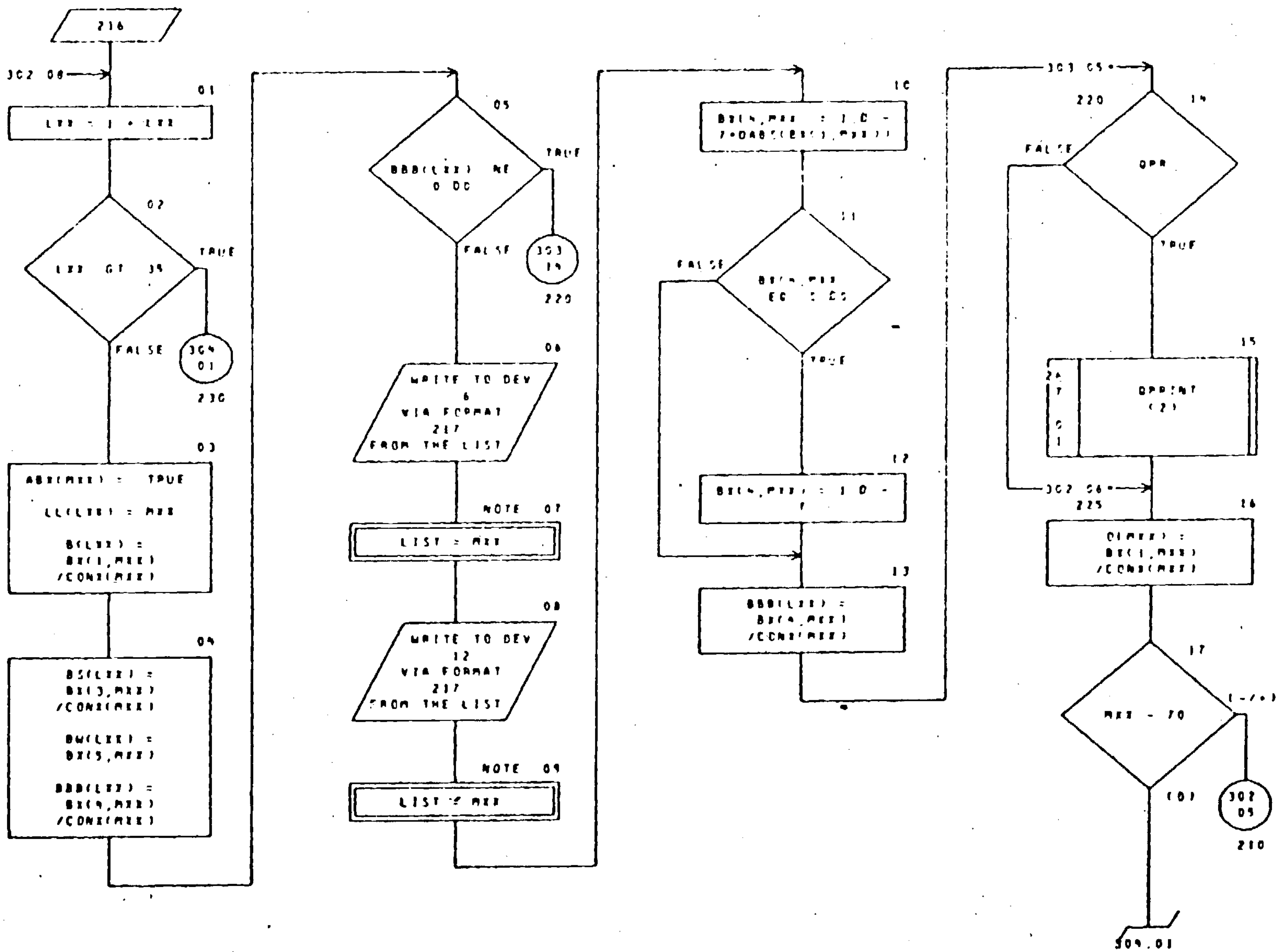
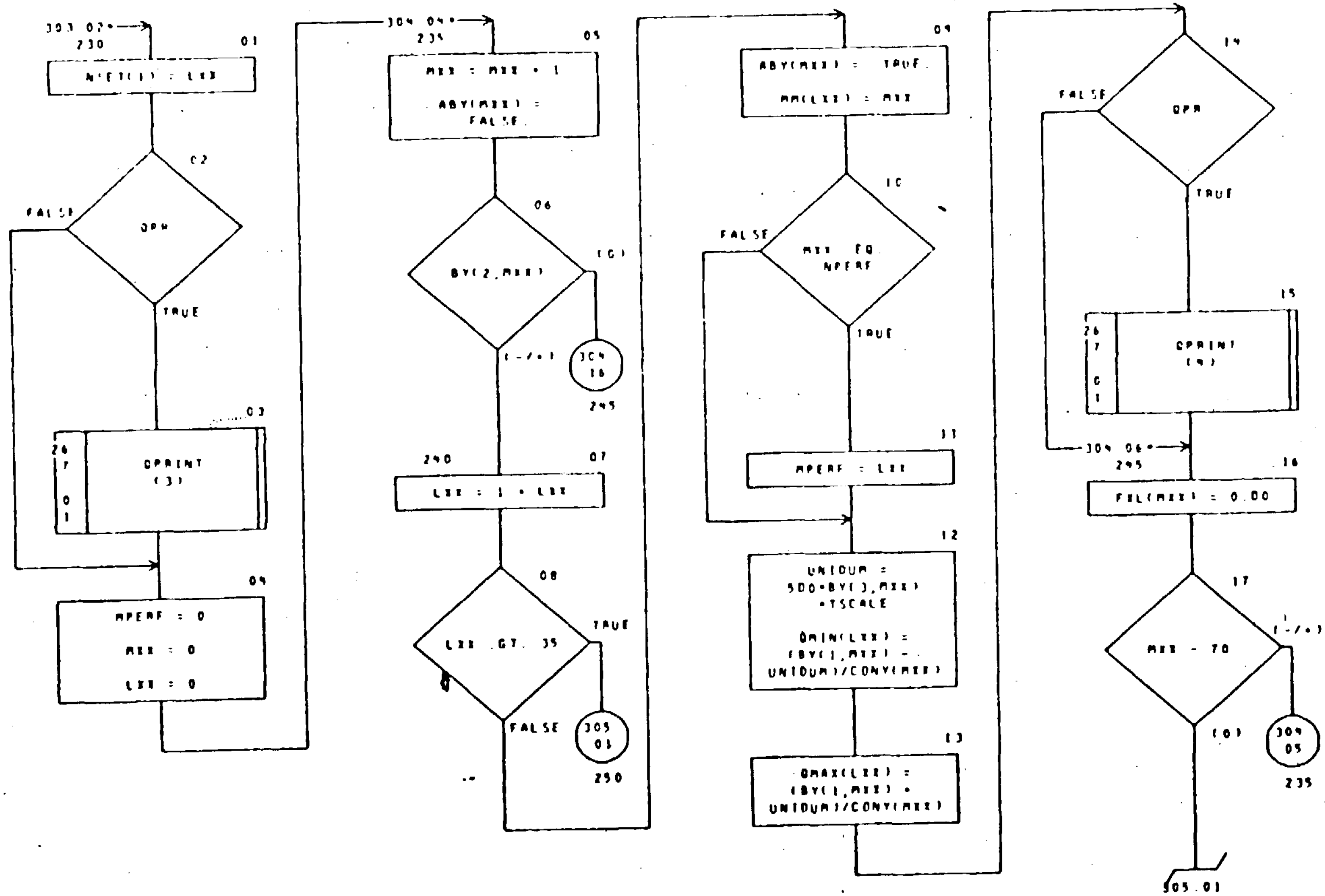
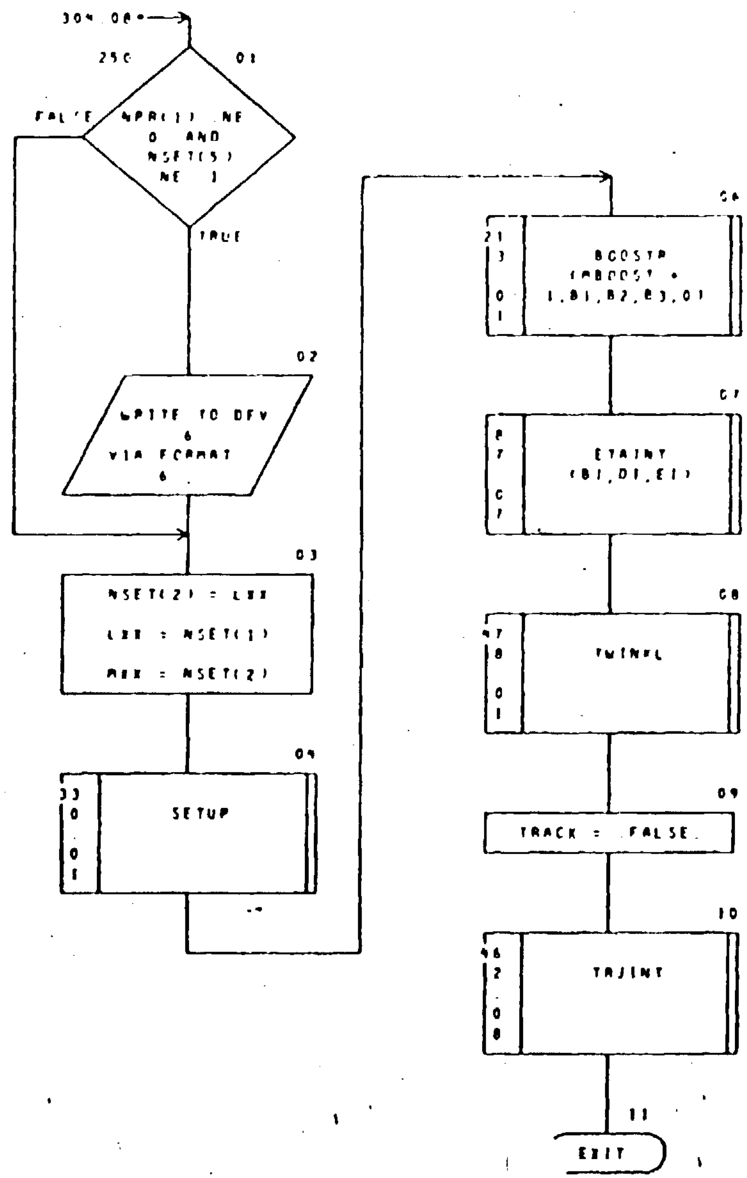


CHART TITLE - SUBROUTINE QSTART(MSTART)



QSTART-29

CHART TITLE - SUBROUTINE OSTART(MSTART)



ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL TUDFLG,FIXPOW,CONVPG,OPR,WONDER,OUTECL,OVLOSS,FIXTHR,TRACK
     ,FLYBY,BRAKE,RENDEZ,POSVEL,SPIRAL,VELOSS,REGION,ABX,ABY,PLANET
     ,ODECL,DIFX,XLOAD,WIPL,MAXPOW,PCURV
      COMMON /REAL8/ RG1(11),FT,VJ,VIMP,R03(27),
             HOUR,ROS,AAI,STATE(6),R5(7),R06(7),BT,DI,ET,B1,B2,B3,POWFIX
     ,AR,AV,AT,SI,CI,AE,SAI,ECI,CNI,OMI,SOI,TPI,EMUDOD,RADCOD,RG7(3),
      TMAX,ROP(11),TOV,TDELV,GAP,RG9(22),SEFMA(7),
      SEFMB(7),SEFMC(7),SEFMD(7),R10(2),TDATE1,TDATE2,TDATEX,R11(29),
      RTSWIT,R12(42),SAIX(5),FCIX(5),CNIX(5),OMIX(5),SOIX(5),TPIX(5),
      EMUODX(5),RADODX(5),R20(43),TSCALE,R14(3),DEG,FPSNAM,SUNMU,
      CONTM,R13(209),TOFF(20),R22(136),
      EP(3),EPD(3),EN(3),END(3),AXIS(3),AXISD(3),ALTITU,R16(125),ETM(3),
      ETMD(3),R19(337),XLONG,R18(22),PPMAXO,PPOWO,R17(7),SX(50),X(50),
      XD(50),R21(650)
      COMMON /INTGP4/ IRL,MODE,I02(2),LINE,I03,MBOOST,I04,NTAPE,MDAY,
      MONTH,MYEAR,I05(2)     ,MPRINT,NSFT(5),IROT,I06(2),MOPT2,MOPT3,
      I07(11),MTMASS,I08(2) ,MPUNCH,LCOUNT,NPRINT,NPR(4),I09,NORMAL
     ,I10(5),KOUNT,IBAL,I11(6),LXX,MXX,I12(6),LAUNCH,I15(3),IOUT,
      I13(53),LEGMAX,MOPTX(5),INTER(5),LOADX,MPOW,I16(2),NPERF,MPERF,
      I01(41),LL(70),MM(70),I14(675)
      COMMON /LOGIC4/ L01,CONVRG,FIXPOW,OPR,WONDER,L02(2),ODECL,TUDFLG,
      OUTECL,
      FIXTHR,OVLOSS,FLYBY,BRAKE,RENDEZ,POSVEL,SPIRAL,VELOSS,REGION,
      L03(5),TRACK,PLANET,OJEX,L04(7),XLOAD,WIRL,PCURV,MAXPOW,L06(187),
```

ORIGINAL PAGE IS
OF POOR QUALITY

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
             ABX(70),ABY(70),LOS(135)
             COMMON /ITERA1/ BX(5,70),BY(3,70),CONX(70),CONY(70),O(70),BO1(70),
             FXL(70),BO2(70)
             COMMON /ITER2/ B(35),PO1(35),BS(35),BW(35),QMIN(35),QMAX(35),
             BBB(35),PO2(1295)
             COMMON /SOLSYS/ GM(70),SOLOI(210)
      6      FORMAT(1H1)
      21     FORMAT(1H ,11HRUN SUMMARY)
      24     FORMAT(1H1,46HINPUT ERROR  HILTOP CURRENTLY LIMITED TO FOUR
             ,20HTRAJECTORY SEGMENTS /1H0)
      19     FORMAT(5H CASEI4)
      26     FORMAT(1H1,44HFORCED COASTING BYPASSED DURING *TOV* OPTION/1H0)
      31     FORMAT(1H1,57HTHIRD INTERMEDIATE TARGET NOT ALLOWED DURING *TOV* O
             PTION/1H0)
      214    FORMAT(1H ,12HTHRUST ANGLEI3,30H IS ZERO. BX AND BY TRIGGERS (
             I2,19H) SET TO ZERO.)
      217    FORMAT(1H0,40HPERTURBATION STEP SIZE IS INPUT ZERO  BYI2
             ,22H CORRECTED INTERNALLY.)
```

POOR QUALITY PAGE IS

<u>Name</u>: RADAR

<u>Calling Argument:</u> QSETUP, DER

<u>Referenced Sub-programs</u>: AEINWT, EFM, VDOT, VMAG

<u>Referenced Commons:</u> INTGR4, ITERAT, LOGIC4, REAL8

<u>Entry Points:</u> None

<u>Referencing Sub-programs</u>: CDERIV, TRAJI

<u>Discussion:</u> The purpose of RADAR is to compute the communication-distance vector R_{com}, its derivative $\dot{R}_{com}$, and the communication angle θ_{com}, of the spacecraft relative to Earth. However, these same vectors and angle may be computed with respect to any reference body in place of Earth, via employing the program input quantity NDIST; this allows the spacecraft proximity and closest approach distance to any object in the solar system to be determined. The communication angle is the angle subtended by the sun-spacecraft line as seen by an observer on the reference planet (usually Earth).

In normal program operation (assuming three dimensional motion), the communication-distance vector and its derivative are obtained simply from

$$R_{com} = R - R_{ref},$$

$$\dot{R}_{com} = \dot{R} - \dot{R}_{ref},$$

where R is the spacecraft position and R_{ref} is the reference body position; positions are in AU, velocities in EMOS, expressed in ecliptic coordinates of date. During thrusting flight, the spacecraft velocity $\dot{R}$ is computed as

$$\dot{R} = (1/r^n)R' = (1/r^n)\, dR/d\beta,$$

where n is the numerical integration exponent (program input quantity AN), $r = |R|$, and β is the trajectory independent-variable.

When simulating two dimensional motion (in the xy plane), it is not currently possible to employ the program's analytic ephemeris computation capability, and therefore the communications reference body must be invented. This is accomplished by assuming that the reference body commences motion at the spacecraft's initial position X_o and proceeds in a circular prograde orbit in the xy plane having radius $r_o = |X_o|$, speed $v_o = \sqrt{1/r_o}$, and angular rate $\dot{\theta} = v_o/r_o$. The initial position angle is $\theta_o = \tan^{-1}(y_o/x_o)$. Then,

$$\theta = \theta_o + \dot{\theta} t,$$

$$R_{com} = R - r_o \begin{bmatrix} \cos\theta \\ \sin\theta \\ 0 \end{bmatrix}$$

$$\dot{R}_{com} = \dot{R} - v_o \begin{bmatrix} -\sin\theta \\ \cos\theta \\ 0 \end{bmatrix}$$

such that

$$R_{ref} = r_o \begin{bmatrix} \cos\theta \\ \sin\theta \\ 0 \end{bmatrix}$$

$$\dot{R}_{ref} = v_o \begin{bmatrix} -\sin\theta \\ \cos\theta \\ 0 \end{bmatrix}$$

The communication angle is then computed (for both two and three dimensional motion) as follows, assuming t is the time elapsed since the start of the trajectory: When $t > 0$;

$$\theta_{com} = \cos^{-1}\left[\frac{-R_{ref} \cdot R_{com}}{|R_{ref}|\,|R_{com}|}\right].$$

When $t = 0$ (special consideration must be given to the start of the trajectory, when, normally, spacecraft and reference body positions coincide, and sometimes their velocities coincide also);

(1) If $|\dot{R}_{com}| > 0$, then

$$\theta_{com} = \cos^{-1}\left[\frac{-R_{ref} \cdot \dot{R}_{com}}{|R_{ref}|\ |\dot{R}_{com}|}\right],$$

since $\dot{R}_{com}$ defines the initial motion of R_{com}.

(2) If $|\dot{R}_{com}| = 0$ (Zero departure excess speed), then

$$\theta_{com} = \cos^{-1}\left[\frac{-R_{ref} \cdot \Lambda_o}{|R_{ref}|\ |\Lambda_o|}\right],$$

since Λ_o, the initial primer vector, determines the direction of thrust and therefore also determines the initial motion of R_{com}.

RADAR EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
A(6)	SUAE		Reference body position and velocity, R_{ref} and $\dot{R}_{ref}$, in AU and EMOS, respectively.
O(70)	U	ITERAT	Array of iterator independent-variables; here, O(15) is the launch date, relative to the reference date TDATEX.
X(50)	UA	REAL8	Array of trajectory dependent (integrated) variables, containing spacecraft position R(X(i), i = 1,2,3), velocity $\dot{R}$ (X(i), i = 4,5,6), and time t (X(17)).
PI	U	REAL8	π.
X0(7)	U	REAL8	Spacecraft initial position and velocity X_o, in AU and EMOS, when simulation is two dimensional.

RADAR EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
DER	UX		Conversion factor from generalized derivatives to time derivatives, $1/r^n$.
CNIX(5)	A	REAL8	Inclination to ecliptic of reference body orbit (location 4), in degrees.
ECIX(5)	A	REAL8	Eccentricity of reference body orbit (location 4).
OMIX(5)	A	REAL8	Ascending node angle of reference body orbit (location 4), in degrees.
SAIX(5)	A	REAL8	Semi-major axis of reference body orbit (location 4), in AU.
SOIX(5)	A	REAL8	Argument of perihelion of reference body orbit (location 4), in degrees.
TPIX(5)	A	REAL8	Time from reference date to perihelion passage, for the reference body (location 4), in days.
XCOM(6)	SA	REAL8	Communications-distance vector and its derivative, R_{com} and $\dot{R}_{com}$, in AU and EMOS, respectively.
CONTM	U	REAL8	Time conversion factor, tau (AU/EMOS) to days.
GLARE	SU	REAL8	Communication angle, θ_{com}, in radians.
NDIST	A	INTGR4	Reference body selector.
EMUODX (5)	A	REAL8	Gravitational constant of reference body (location 4), in m^3/sec^2 (not used at present).
OUTECL	U	LOGIC4	Indicator for out-of-ecliptic mission.
PLANET	U	LOGIC4	Ephemeris option indicator (usually equivalent to three dimensional motion simulation).

RADAR EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
QSETUP	UX		Initialization indicator.
RADODX(5)	A	REAL8	Radius of reference body (location 4), in meters (not used at present).
TDATEX	UA	REAL8	Reference date (defined by program input quantity MYEAR, etc.).
TUDFLG	U	LOGIC4	Indicator for two dimensional motion in the xy plane; "2D flag".

CHART TITLE - SUBROUTINE RADAR(QSETUP,DER)

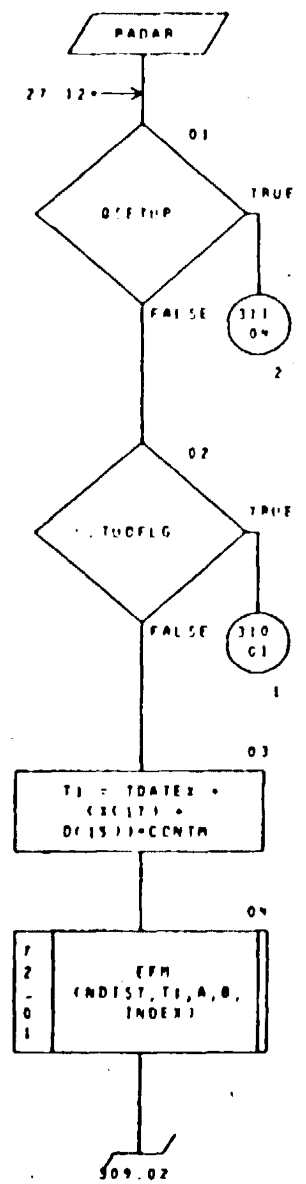
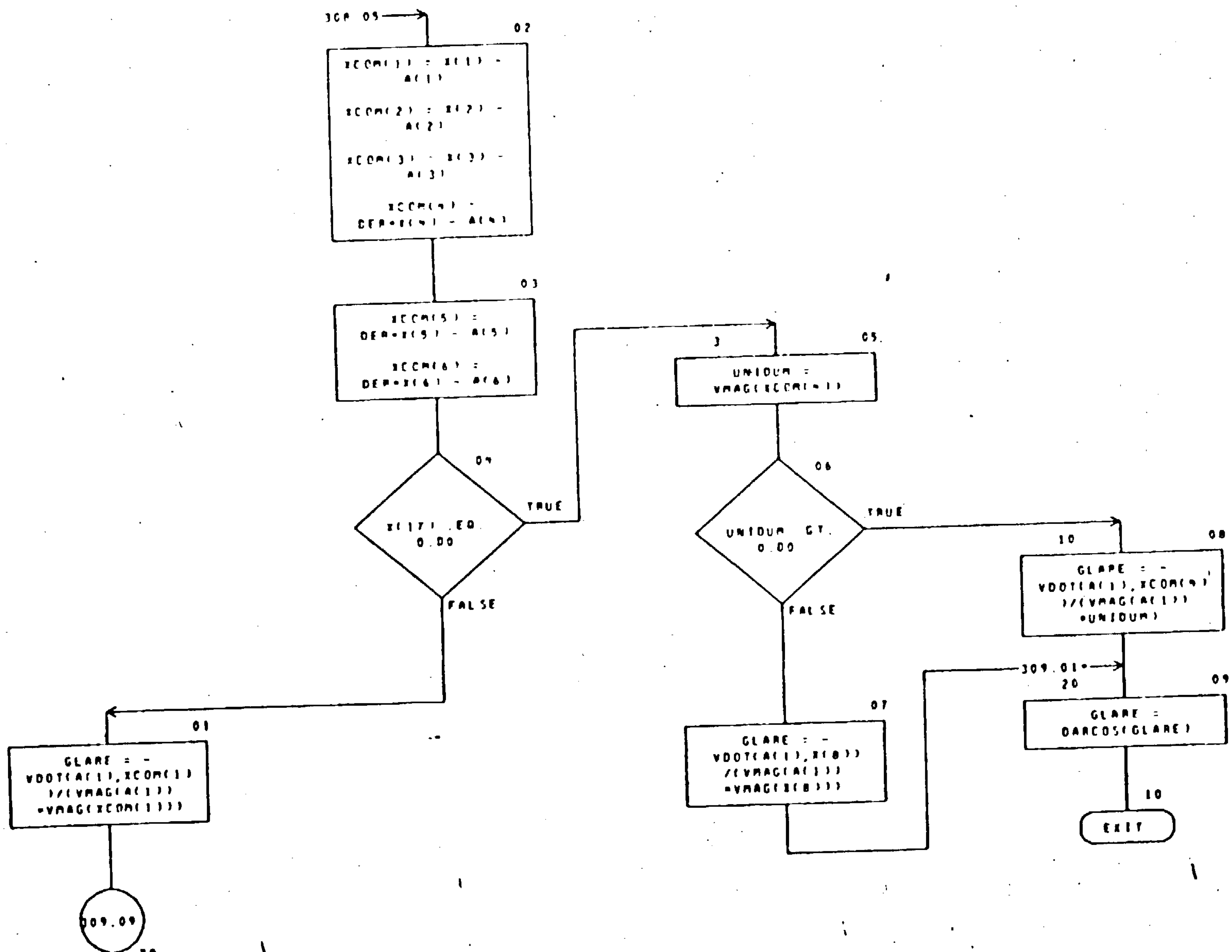


CHART TITLE - SUBROUTINE RADAR(OSETUP,DER)



RADAR-7

CHART TITLE - SUBROUTINE RADAR(QSETUP,DER)

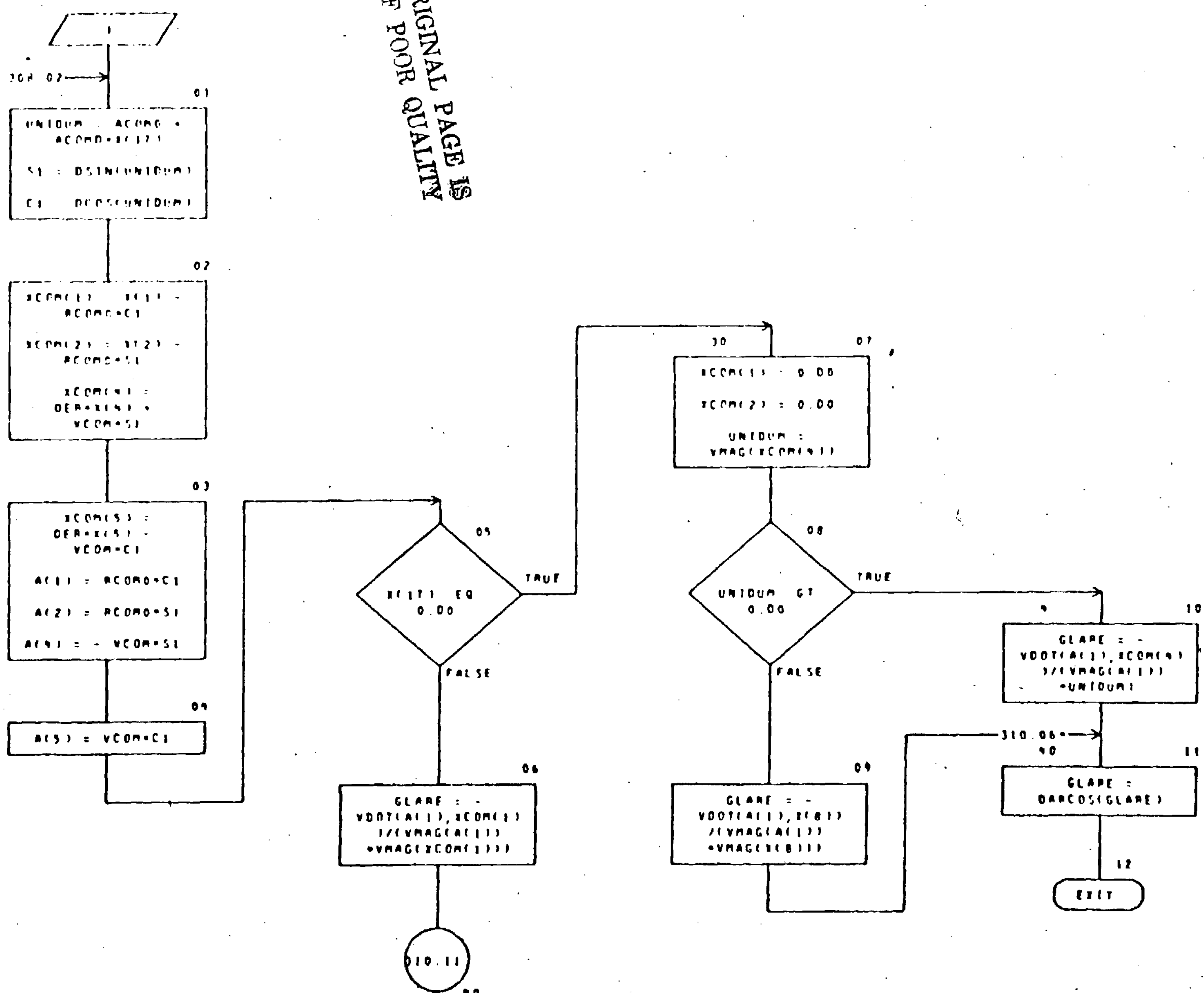


CHART TITLE - SUBROUTINE RADAR(QSETUP,DER)

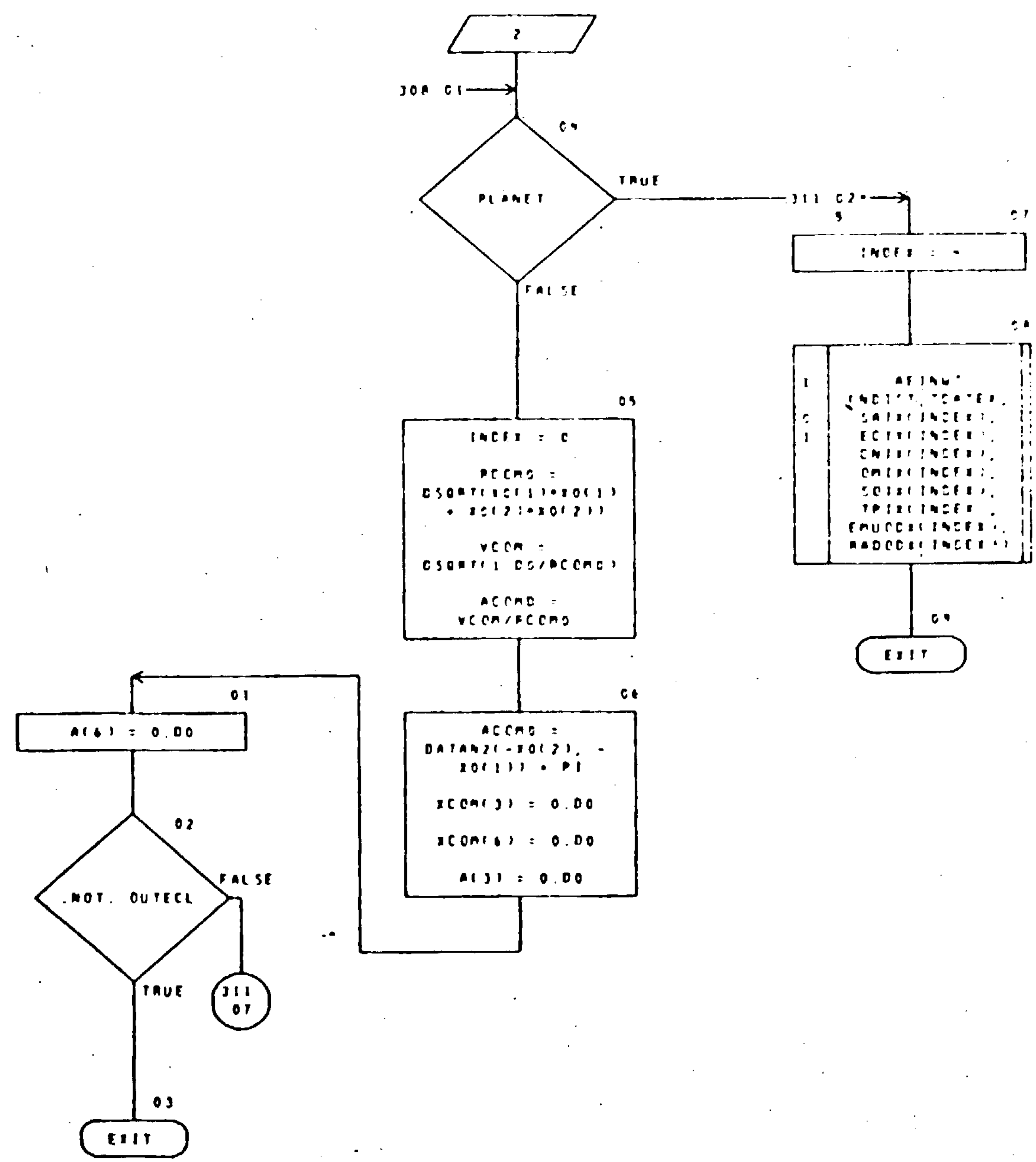


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL OSETUP,TUDFLG,PLANET,OUTECL
DIMENSION A(6),B(6)
COMMON /REAL8/ RO1(50),XO(7),RO2(61),XCOM(6),GLARE,RO3(32),
TDATEX,RO4(34),EVC(6),RO5(27),SAIX(5),ECIX(5),CNIX(5),OMIX(5),
SOIX(5),TPIX(5),EMUPDX(5),RADCCX(5),RO9(50),
                              CCNTM,RO8(3),PI,RO6(930),X(50),
RO7(700)
COMMON /INTGAN/ IC1(191),NDIST,IC2(454)
COMMON /LOGICN/ LO1(8),TUDFLG,OUTECL,LO2(15),PLANET,LO3(474)
COMMON /ITERAT/ BO1(700),D(70),BO2(210)
EQUIVALENCE (A(1),EVC(1))
```

Name: REMTIM

Calling Argument: I, J

Referenced Sub-programs: None

Referenced Commons: None

Entry Points: None

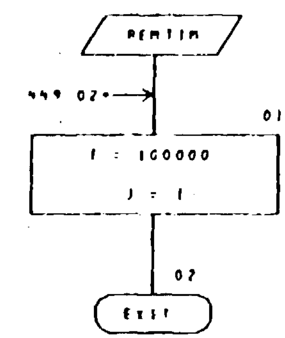
Referencing Sub-programs: TIKTOK

Discussion: This is a dummy subroutine, to replace the IBM systems subroutine REMTIM at the Goddard Space Flight Center. The machine time-out feature and printout discussed under subroutine TIKTOK are therefore not available when this dummy routine is part of the program. The arguments of this dummy routine are set to large values so that subroutine TIKTOK thinks there is a large amount of machine time remaining for the current computer run.

REMTIM EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
I	SX		Remaining CPU time for run, in seconds.
J	SX		Remaining I/O time for run, in seconds.

CHART TITLE - SUBROUTINE REMTIM(I,J)



Name: RETINJ

Calling Argument: None

Referenced Sub-programs: None

Referenced Commons: LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: TRAJ

Discussion: RETINJ is a contraction of "retro injection", and refers to the retro maneuver at the primary target, in which the spacecraft is injected into a capture orbit about that target. This may be accomplished with either a high-thrust retro stage, or a low-thrust spiral using the electric propulsion engine which performed the heliocentric transfer.

In the high-thrust maneuver, a retro stage brakes the spacecraft, approaching the primary target along a hyperbolic orbit of excess speed $v_{\infty n}$, into an elliptical capture orbit of pericenter distance r_p and apocenter distance r_a. The retro stage thrust f_r and jet exhaust speed c_r are specified by input. The retro maneuver is assumed to take place at the periapsis of the approach hyperbola; therefore, the impulsive change in velocity is given by

$$\Delta v = (v_{\infty n}^2 + 2v_c^2)^{\frac{1}{2}} - \left(\frac{2r_a v_c^2}{(r_a + r_p)}\right)^{\frac{1}{2}},$$

where

$$v_c^2 = \mu_t / r_p ,$$

and μ_t is the gravitational constant of the primary target. Provision for including the finite thrust velocity penalty is optionally available. This feature employs the theory developed by Robbins (Reference 1). The total velocity required, including the velocity penalty, is given by

$$\Delta v' = \Delta v + c_1 e_x f_x ,$$

where $\Delta v'$ is solved iteratively with the following equations,

$$c_1 = k_c \, (v_{\infty n}^2 + v_c^2)/(v_{\infty n}^2 + 2v_c^2) \, ,$$

$$k_c = c_r \left(\frac{v_c c_r (m_o \nu_n - j_{ps} m_{ps} - j_t m_t)}{2 r_p f_r} \right)^2 ,$$

$$e_x = 1 - e^{(-\Delta v'/c_r)} ,$$

$$f_x = 2 - (1 + \frac{2c_r}{\Delta v'}) \, e_x ,$$

where j_{ps} and j_t are retro jettison indicators, ν_n is the final mass ratio (at the primary target), m_o is the initial spacecraft mass, m_{ps} is the electric propulsion system mass, and m_t is the electric-propulsion tankage mass.

The approximate performance requirements of the electric propulsion spiral capture maneuver at the primary target are computed as follows. The approximation is based on asymptotic matching techniques developed by Fimple and Edelbaum (Reference 2) and by Breakwell and Rauch (Reference 3). The technique assumes that a heliocentric trajectory to a conceptually massless point, having the position and velocity of the primary target, is available. The approximation then yields the additional propellant and propulsion time that would be required above that of the heliocentric trajectory to insert the spacecraft into an orbit of periapse r_p and apoapse r_a using the electric propulsion spiral maneuver. It should be noted that the additional propellant and time computed in this approximation does not represent the propellant and time spent performing the spiral with very high accuracy because the heliocentric trajectory included a trajectory segment which was within the geometric boundaries of the sphere of influence of the planet. The additional propellant and time computed is more appropriately considered a correction to the heliocentric trajectory which, when added to the requirements of the heliocentric trajectory, yields a good estimate of the total performance requirement, including those of the spiral.

Defining the semi-major axis of the capture ellipse

$$a_c = (r_a + r_p)/2,$$

and the thrust acceleration at the end of the heliocentric trajectory (at the primary target)

$$a_n = g\gamma/\nu_n ,$$

then the incremental speed associated with the spiral maneuver is calculated

$$\Delta v = \sqrt{\frac{\mu_t}{a_c}} \left[1 - 1.84\left(\frac{a_n a_c^2}{\mu_t}\right)^{\frac{1}{4}}\right],$$

which leads to the additional propellant

$$\Delta m_p = m_o \nu_n (1 - e^{-\Delta v'/c}),$$

and the additional time

$$\Delta t = c\,(1 - e^{-\Delta v'/c})/a_n ,$$

where g is the electric propulsion reference thrust acceleration, c is the jet exhaust speed, and γ is the power ratio at the primary target (at commencement of the spiral maneuver.) A velocity penalty Δv_x is adjoined by iteratively solving $\Delta v_x = \left(\sqrt{\frac{\mu_t}{a_c}} - \Delta v\right)\left(1 - e^{\Delta v'/4c}\right)$. The total speed is then $\Delta v' = \Delta v + \Delta v_x$.

Messages and printouts: If either iteration (high thrust velocity-loss or low thrust velocity penalty) fails, the following diagnostic message is printed on unit 6:

**** RETRO MASS RATIO NOT FOUND TO TOLERANCE (f) (f') (Δv)

in which f is the function (dependent variable) for which the root is sought by iteration, f' is the function derivative with respect to the independent variable of iteration, and Δv is velocity loss in the high thrust case and total speed in the low thrust case, in meters/second.

Should the thrust acceleration at the end of the heliocentric trajectory (at the primary target), a_n, be zero or negative, as a result of the main heliocentric iteration sequence, then the message is printed on unit 6:

****SPIRAL COMPUTATIONS DELETED

In either case above, the routine is simply exited after the message is printed.

References:

1. Robbins, H.M., "An Analytical Study of the Impulsive Approximation," AIAA Journal, Vol. 4, No. 8, pp. 1417-1423, 1966.

2. Fimple, W.R. and Edelbaum, T.N., "Applications of SNAP-50 Class Powerplants to Selected Unmanned Electric Propulsion Missions," AIAA Paper 64-494, 1964.

3. Breakwell, J.V. and Rauch, H.E., "Asymptotic Matching in Power Limited Interplanetary Transfers," AAS Paper 66-114, 1966.

RETINJ EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
AM	S	REAL8	Multiplicative factor, zero or one, for the term $\Lambda_n \cdot \ddot{P}_n$ in the transversality conditions associated with flight time; used in subroutine GETQ.
EX	SU	REAL8	The quantity e_x in the high-thrust retro computations.
FP	SU	REAL8	f', the derivative of $f = \Delta v' - \Delta v - c_1 e_x f_x$ with respect to the iteration independent variable, $\Delta v'$.
FT	U	REAL8	Reference thrust acceleration of electric propulsion system, g, in AU/tau^2.

RETINJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
VH	SU	REAL8	Speed at periapse of approach hyperbolic trajectory, in meters/second.
VJ	U	REAL8	Jet exhaust speed of electric propulsion system, c, in EMOS.
VS	SU	REAL8	v_c^2, in meters^2/second^2.
RAP	U	REAL8	Capture-orbit apoapse distance, r_a, in planet radii.
VHS	SU	REAL8	VH squared.
CONG	U	REAL8	Reference acceleration of gravity at Earth's surface, in meters/second^2.
DVEL	SU	REAL8	High thrust $\Delta v'$, in meters/second.
POWR	U	REAL8	Power function $q\gamma$ evaluated at the primary target.
QCST	S	REAL8	Retro stage mass used in general formula, subroutine TRAJ.
RPER	U	REAL8	Capture-orbit periapse distance, r_p, in planet radii.
VINF	U	REAL8	Hyperbolic excess speed at the primary target, $v_{\infty n}$, in meters/second.
VORB	SU	REAL8	Spacecraft speed at periapse of capture orbit, in meters/second.
CONA0	U	REAL8	Acceleration conversion factor, AU/tau^2 to meters/second^2.
CONSP	U	REAL8	Speed conversion factor, EMOS to meters/second.
FLYBY	U	LOGIC4	Indicator for flyby of primary target.

RETINJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
GPLAN	U	REAL8	Gravitational constant of primary target, μ_t, in m^3/sec^2.
RPLAN	U	REAL8	Radius of primary target, in meters.
THRET	U	REAL8	Retro-stage thrust, in pounds.
THSPY	S	REAL8	Spiral-stage thrust, in pounds.
VJRET	SU	REAL8	Retro-stage jet exhaust speed, c_r, in meters/second.
VLOSS	SU	REAL8	Velocity loss, in meters/second.
XMASS(7)	U	REAL8	General mass array; XMASS(1) is initial spacecraft mass, m_o, in kilograms.
XMSPY	S	REAL8	Spiral additional propellant, Δm_p, in kilograms.
CONLBS	U	REAL8	Conversion factor, pounds to newtons.
DMRETR	U	REAL8	Retro engine mass, m_{rs}, in kilograms.
POSVEL	U	LOGIC4	Indicator for specified flyby speed (greater than zero).
QTMASS(5)	SU	REAL8	Array of mass-related parameters (1) $e^{-\Delta v'/c_r}$ (or unity) (2) $m_o \nu_n - j_{ps} m_{ps} - j_t m_t$ (3) QTMASS(2) x (1-QTMASS(1)) (4) Not used. (5) $m_{rs} + c_r$ x QTMASS(3)
RENDEZ	U	LOGIC4	Indicator for rendezvous (specified arrival excess speed having value zero).
SPIRAL	U	LOGIC4	Indicator for spiral maneuver.

RETINJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
SPIRET	U	REAL8	Retro-stage specific impulse, in seconds.
SPISPY	S	REAL8	Spiral-stage specific impulse, in seconds.
TIMSPY	S	REAL8	Spiral additional time, Δt, in days.
VELOSS	U	LOGIC4	Indicator for performing velocity loss or velocity penalty computations.

CHART TITLE - SUBROUTINE RETINJ

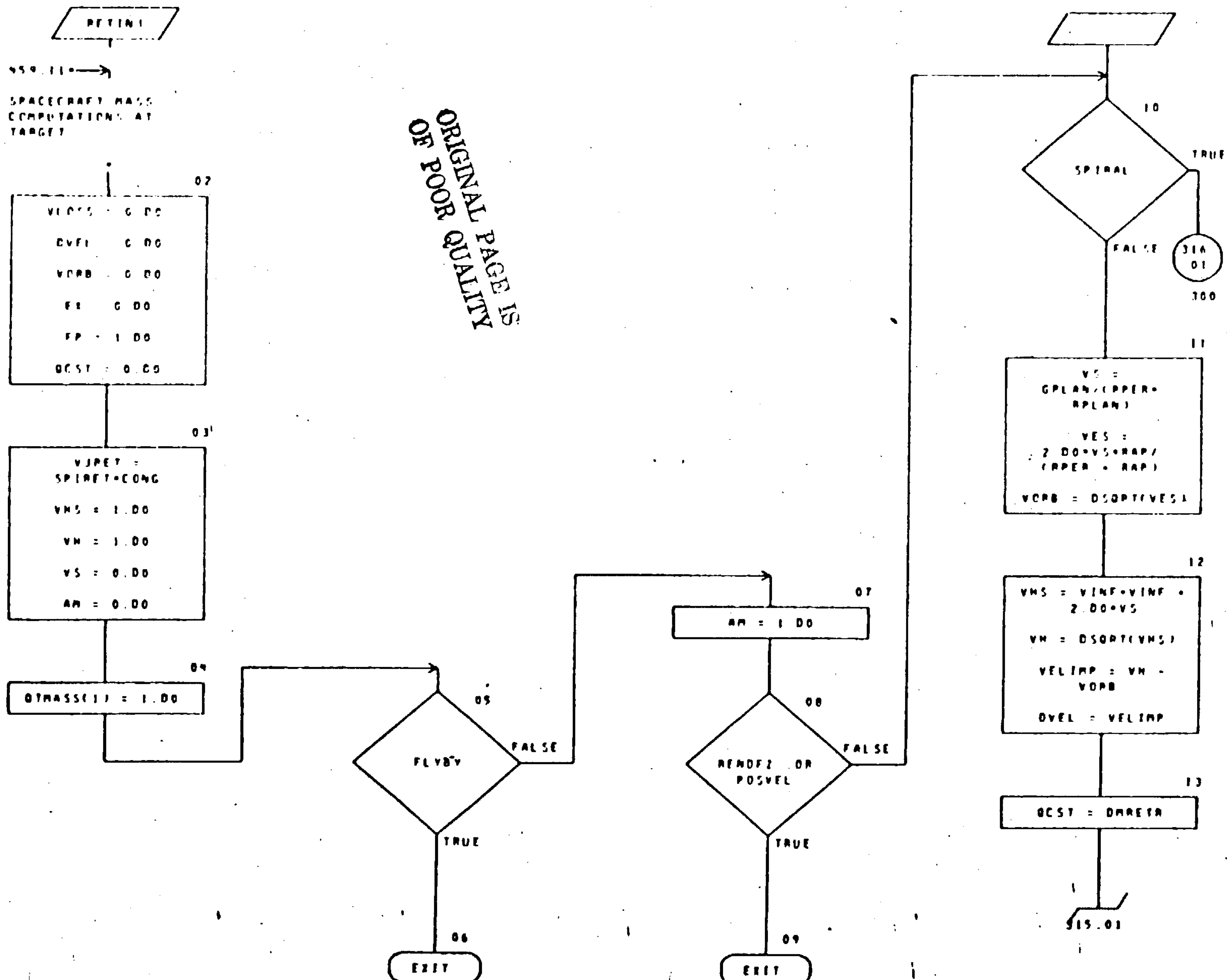
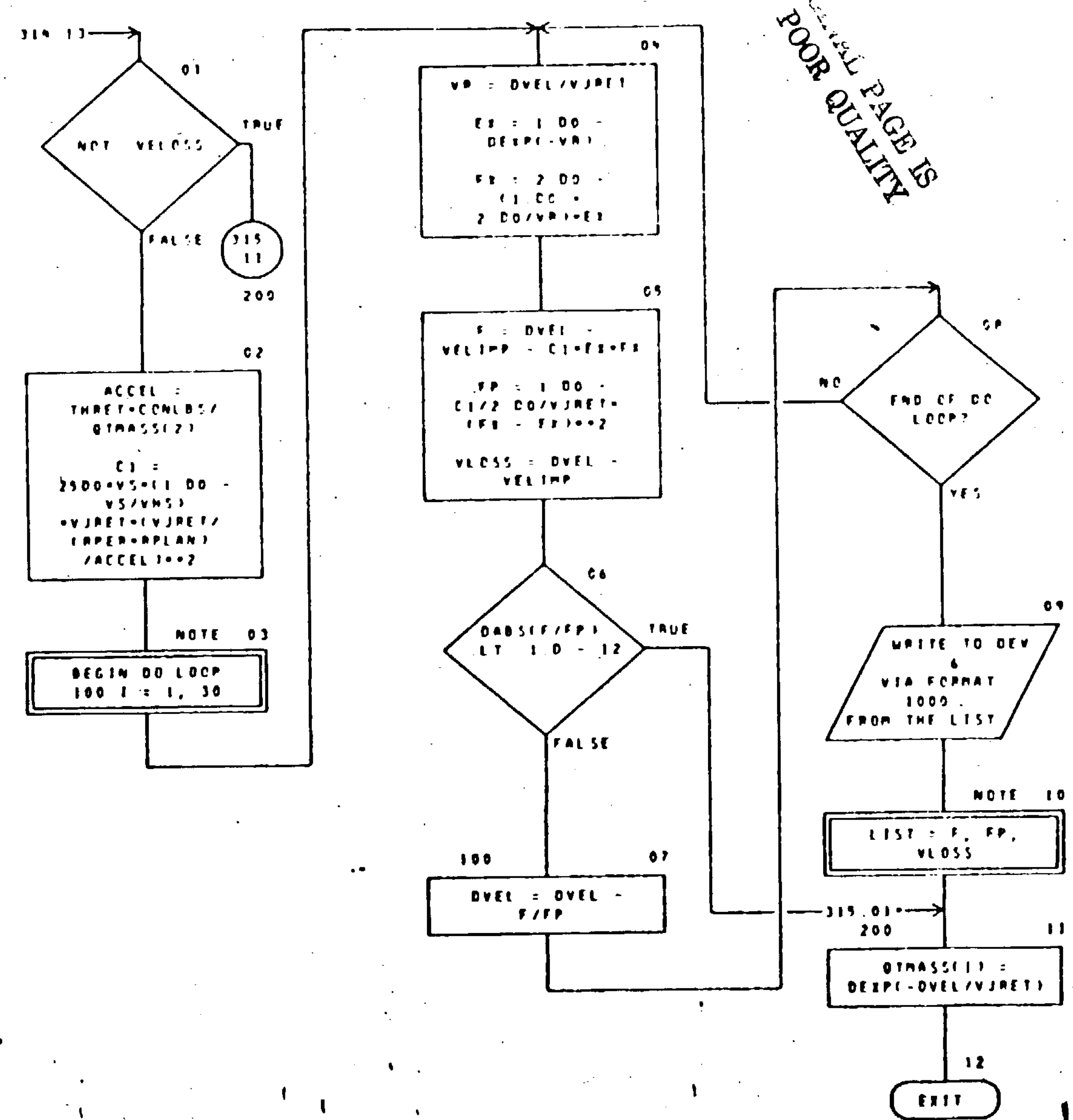
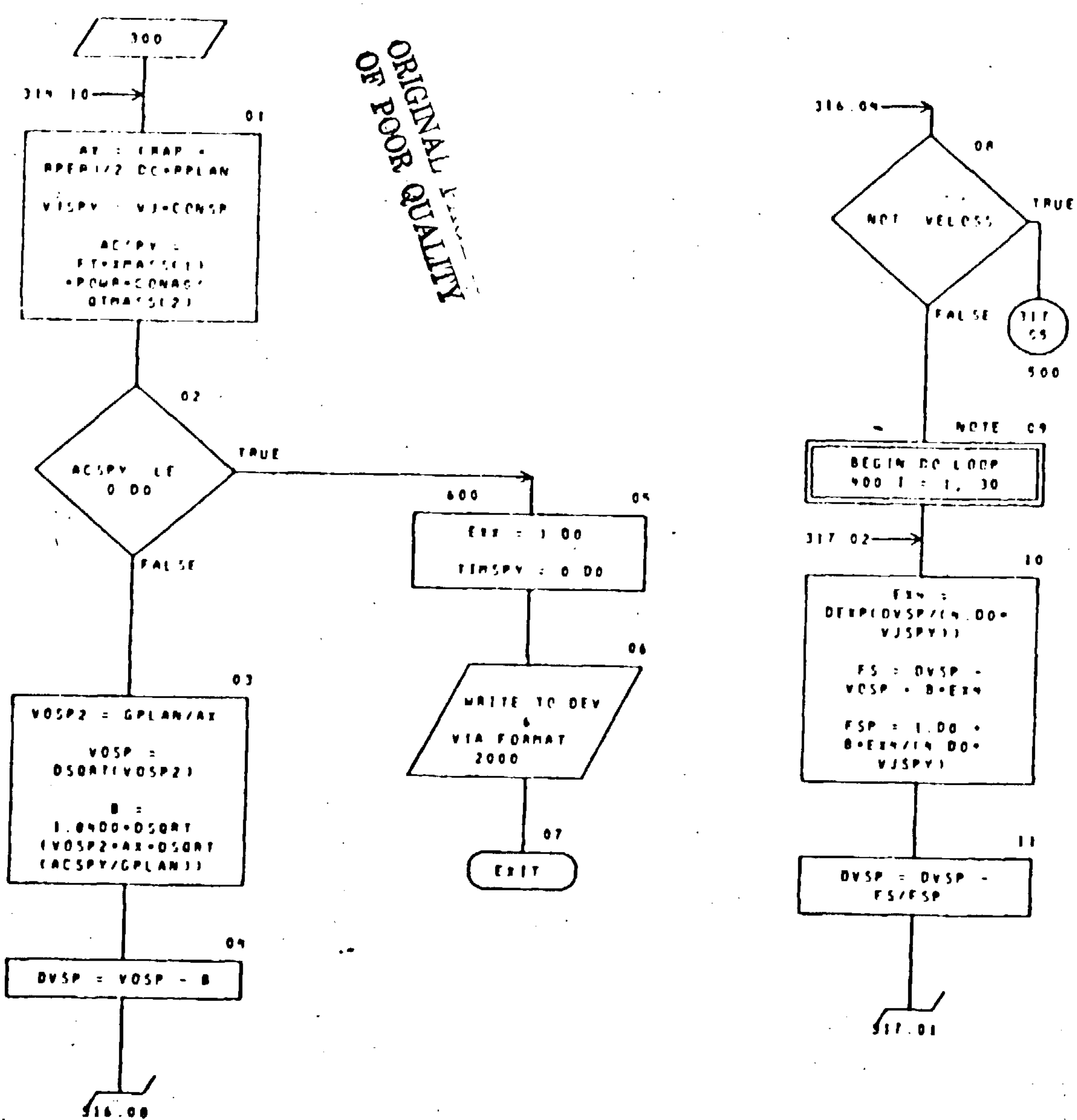


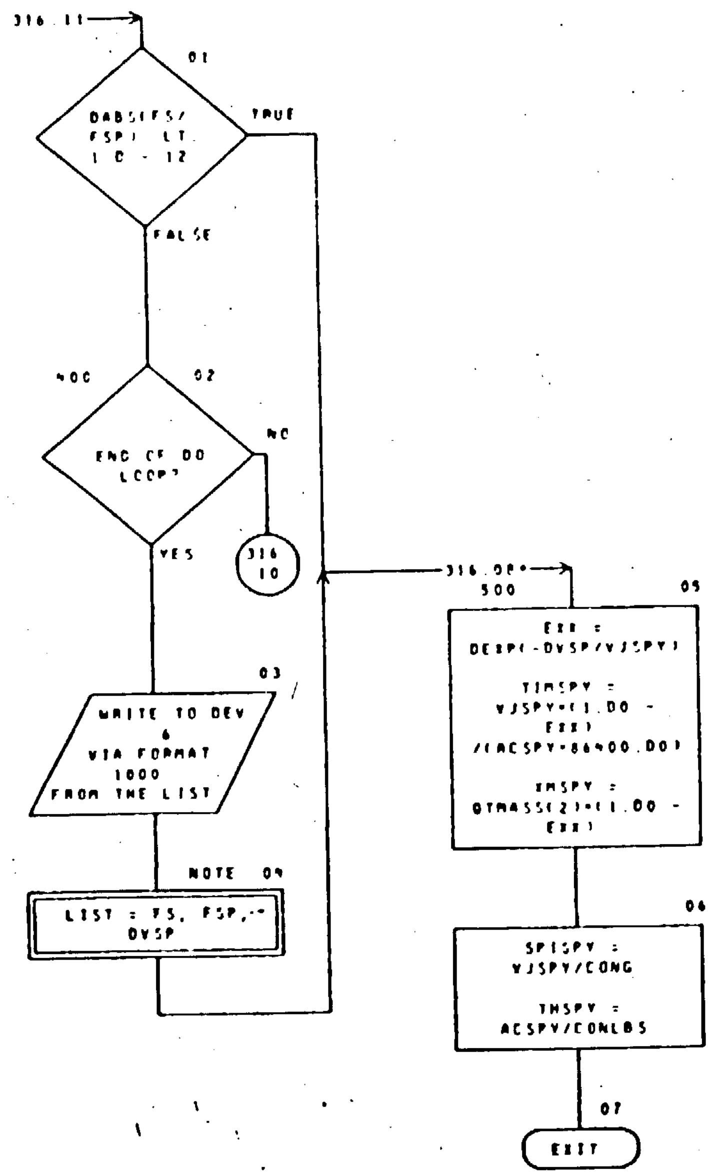
CHART TITLE - SUBROUTINE, RETINJ





ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - SUBROUTINE RETINJ



RETINJ-11

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL FLYBY,RENDEZ,POSVEL,SPIRAL,VELOSS
      COMMON /REAL8/ ROI,XMASS(7),RC2(3),FT,VJ,RC3,QTMASS(5),OMRETR,RON,
     RREA,RAP,THRET,SPIRET,DVEL,VLOSS,VORB,POS(15P),GPLAN,RPLAN,SPISPY,
     THSPY,TIMSPY,XMSPY,RG(124),CONSP,CONAO,ROT(4),CONLBS,CONG,
     RSA(30),POWR,RO9(4),AM,FX,FP,OCST,VH,VHS,VINF,VJRET,VS,RIG(8)2)
      COMMON /LOGIC4/ LC1(12),FLYBY,LC2,RENDEZ,POSVEL,SPIRAL,VELOSS,
     LC3(482)
 1000 FORMAT (1H0,'**** RETRO MASS RATIO NOT FOUND TO TOLERANCE'3D12.4)
 2000 FORMAT(1H0'**** SPIRAL COMPUTATIONS DELETED')
```

ORIGINAL PAGE IS OF POOR QUALITY

Name: RKSTEP

Calling Argument: STEP1, GO

Referenced Sub-programs: DERIV, PRIOR

Referenced Commons: INTGR4, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: STEP

Discussion: This routine performs the standard fourth-order Runge Kutta integration of the equations of motion during thrust periods. Briefly, the vector equations of motion are expressed in first-order form as

$$\dot{X} = F(X),$$

where the dot ($\dot{}$) represents differentiation with respect to the trajectory independent-variable β. The vector of trajectory dependent-variable values X at the end of a computation step $(\beta + \Delta\beta)$ are then obtained from the known values at β from

$$X_{\beta+\Delta\beta} = X_\beta + (\Delta\beta)\, S \;,$$

where

$$S = (K_1 + 2K_2 + 2K_3 + K_4)/6 \;,$$

and

$$K_1 = F(X_\beta) \;,$$

$$K_2 = F(X_\beta + (\Delta\beta)\, K_1/2) \;,$$

$$K_3 = F(X_\beta + (\Delta\beta)\, K_2/2) \;,$$

$$K_4 = F(X_\beta + (\Delta\beta)\, K_3).$$

The vector function F is represented by the derivative subroutine, DERIV.

The vector of trajectory dependent-variables, or integrated functions, contains 50 locations, to allow for future program expansion; only 20 locations are used at present, and these are allocated as follows:

X(i)	Symbol	Description
X(1)	x	
X(2)	y	R, spacecraft position, in AU.
X(3)	z	
X(4)	$\dot{x}$	
X(5)	$\dot{y}$	$\dot{R}$, spacecraft velocity, in EMOS.
X(6)	$\dot{z}$	
X(7)	ν	Mass ratio.
X(8)	λ_x	
X(9)	λ_y	Λ, the primer vector.
X(10)	λ_z	
X(11)	$\dot{\lambda}_x$	
X(12)	$\dot{\lambda}_y$	$\dot{\Lambda}$, the primer derivative.
X(13)	$\dot{\lambda}_z$	
X(14)	λ_ν	Mass ratio adjoint variable.
X(15)	λ_g	Reference thrust acceleration adjoint variable.
X(16)	λ_c	Jet exhaust speed adjoint variable.
X(17)	t	Time, in tau (AU/EMOS)
X(18)	λ_s	Degradation time adjoint variable.
X(21)	λ_ϕ	Fixed thrust angle adjoint variable.
X(31)	s	Degradation time, in tau (AU/EMOS).

Locations X(i) not specified above are available for use. Actually, the derivatives listed above are time derivatives only during coasting flight, and are generalized derivatives during thrusting flight, as described in the discussion of subroutine DERIV.

RKSTEP EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
X(50)	S	REAL8	Array of trajectory dependent-variables, X.
GO	AX		Logical indicator for stepping forward; when true, perform saving operation; when false, perform restoration operation.
SX(50)	U	REAL8	Array of trajectory dependent-variables corresponding to start of current computation step, X_β ("saved X").
XD(50)	U	REAL8	Array of trajectory dependent-variable derivatives, $\dot{X}$.
TAU	S	REAL8	Propulsion time, τ, in tau.
BETA	S	REAL8	Trajectory independent-variable, β.
STAU	U	REAL8	Propulsion time at start of current computation step ("saved τ").
ERODE	U	LOGIC4	Power degradation option indicator.
SBETA	U	REAL8	Trajectory independent-variable β at start of current computation step ("saved β").
STEP1	UX		Computation step size, $\Delta\beta$.
FIXTHR	U	LOGIC4	Indicator for fixed thrust-cone-angle.
NPHI20	U	INTGR4	Index which selects the currently applicable fixed thrust-cone-angle adjoint variable; set equal to 21 for the present.
QERODE	U	LOGIC4	Indicator which is true when either ERODE or QJEX (final case-summary trajectory) is true.

CHART TITLE - SUBROUTINE RKSTEP(STEP1,GO)

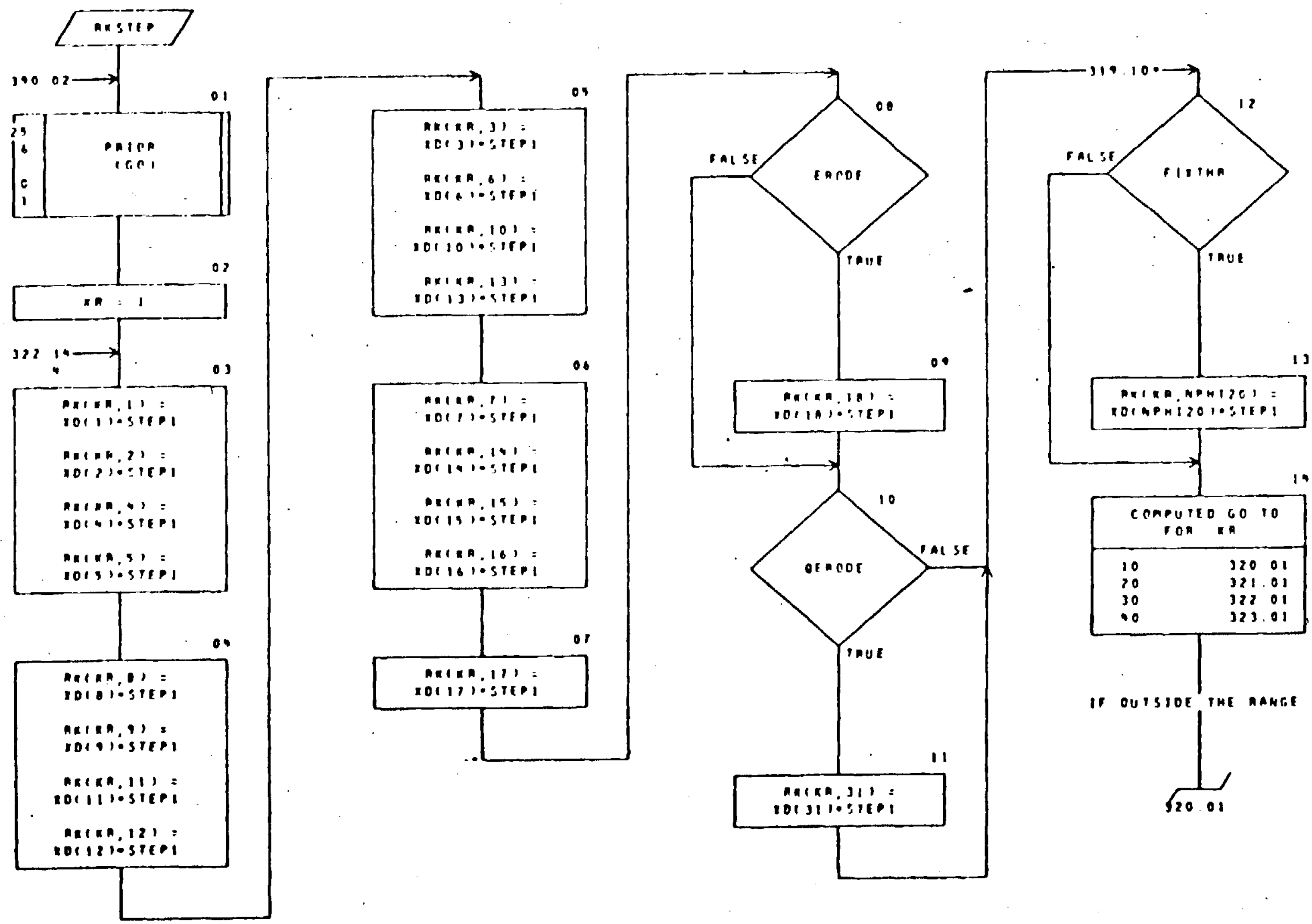


CHART TITLE - SUBROUTINE RKSTEP(STEP1,GO)

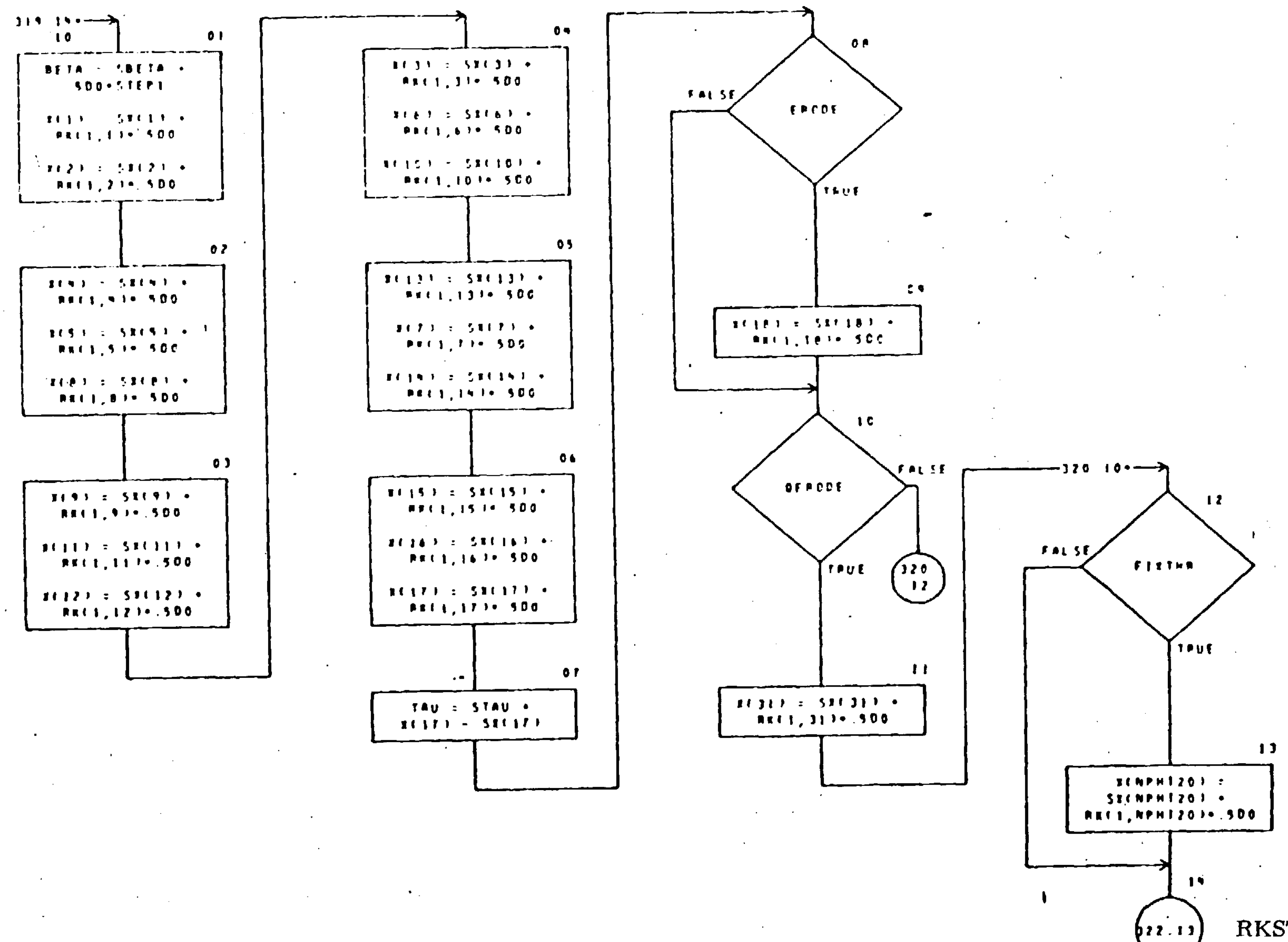


CHART TITLE - SUBROUTINE RKSTEP(STEP1,GO)

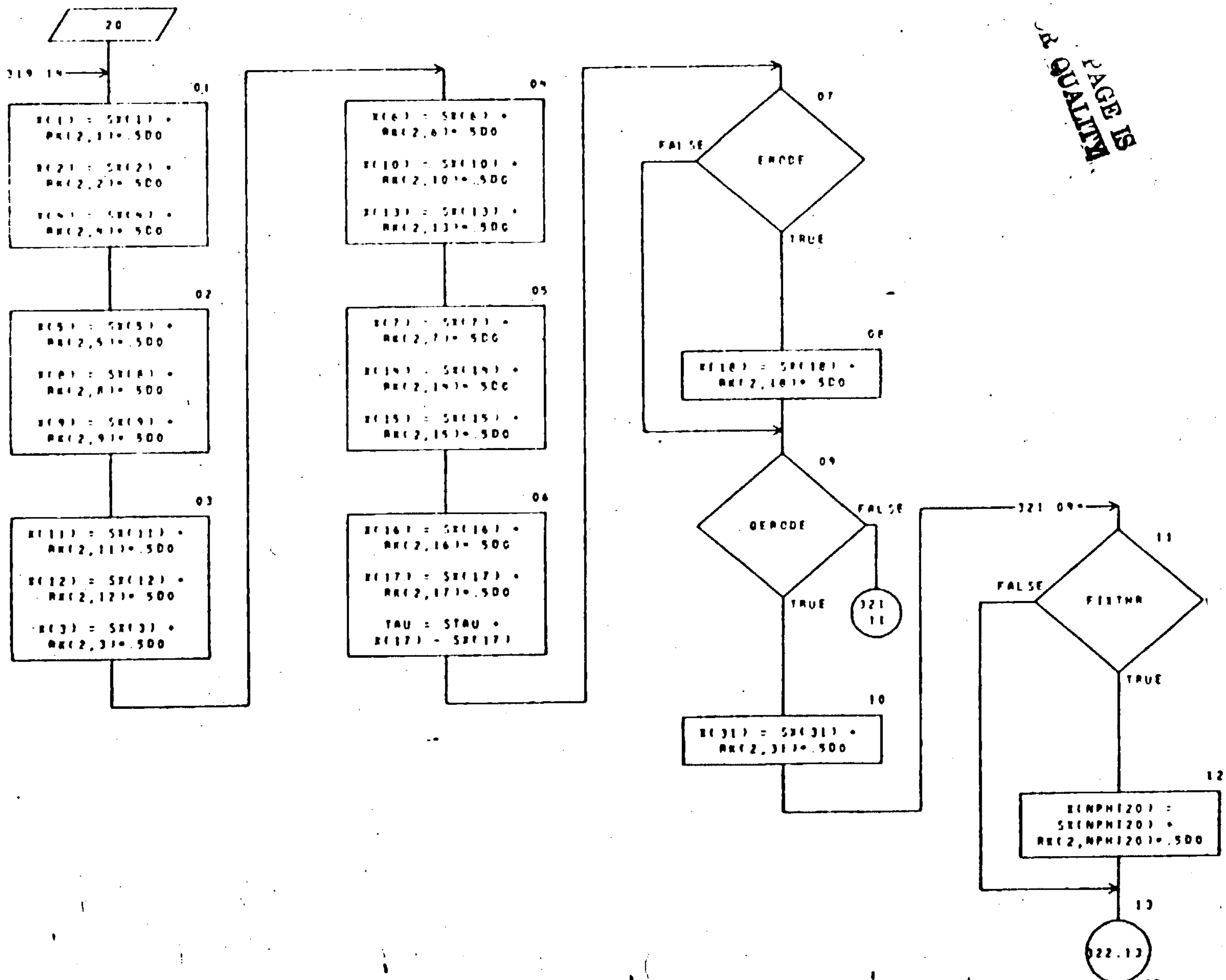


CHART TITLE - SUBROUTINE RKSTEP(STEP1,GO)

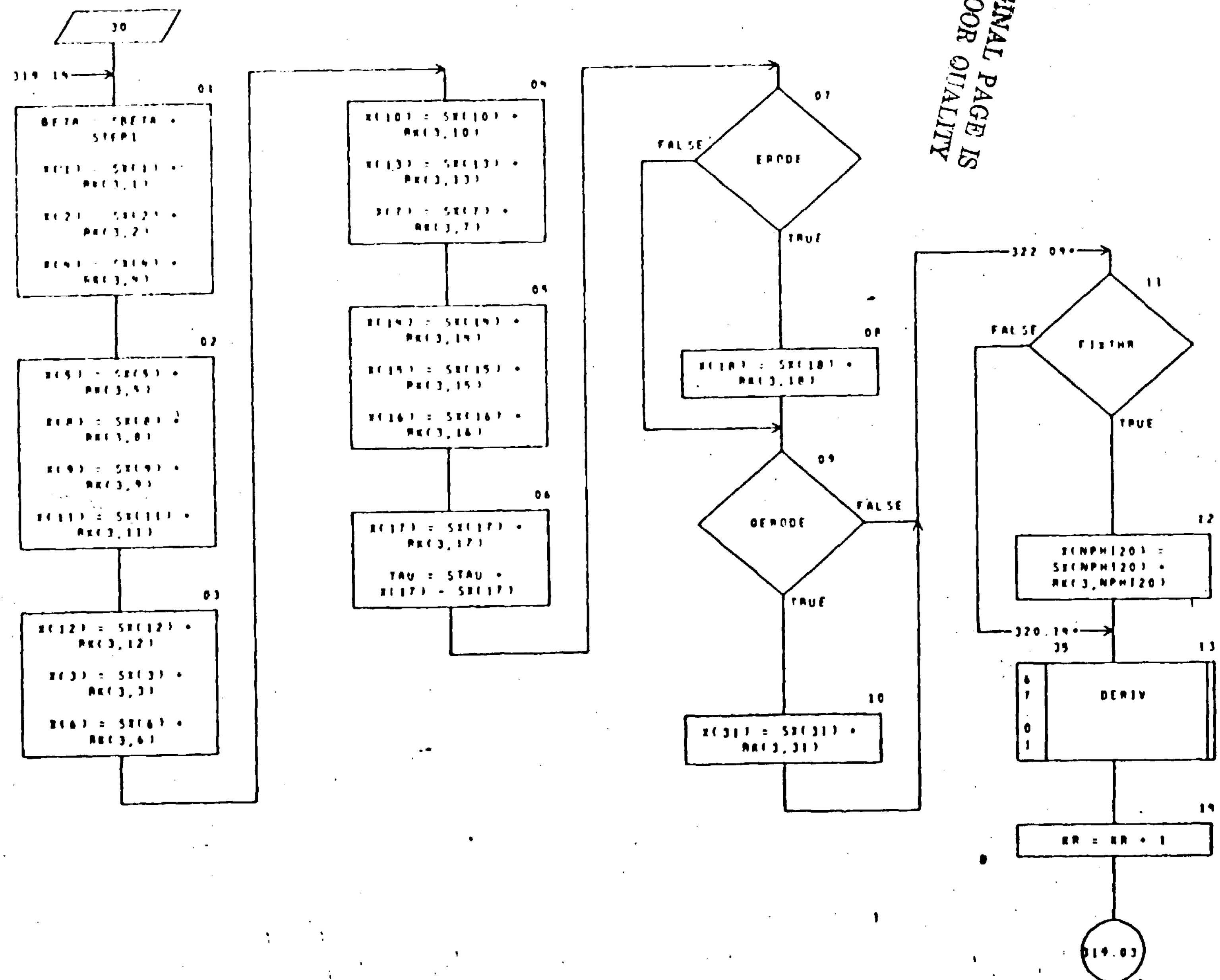


CHART TITLE - SUBROUTINE RKSTEP(STEP1,GO)

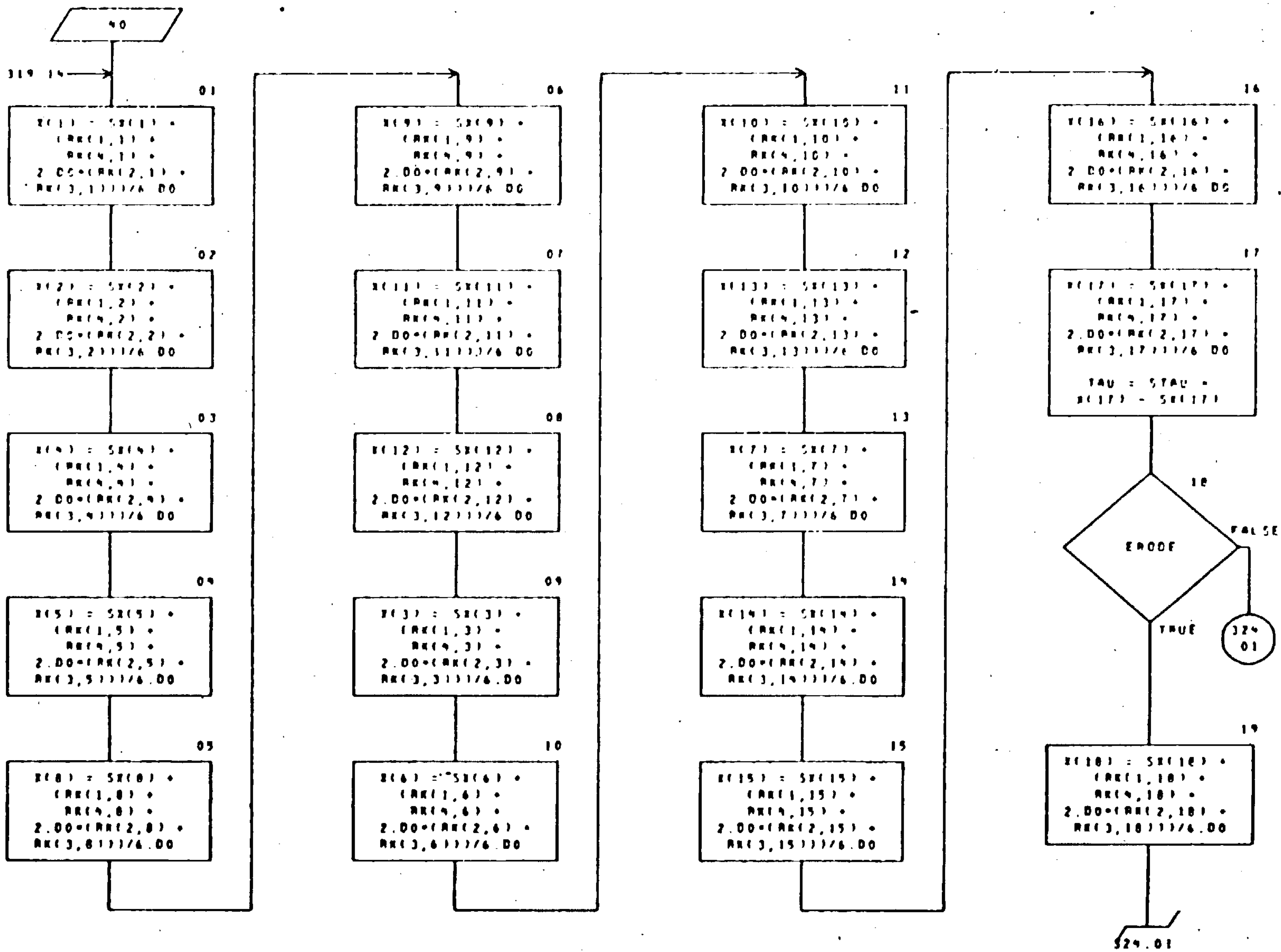
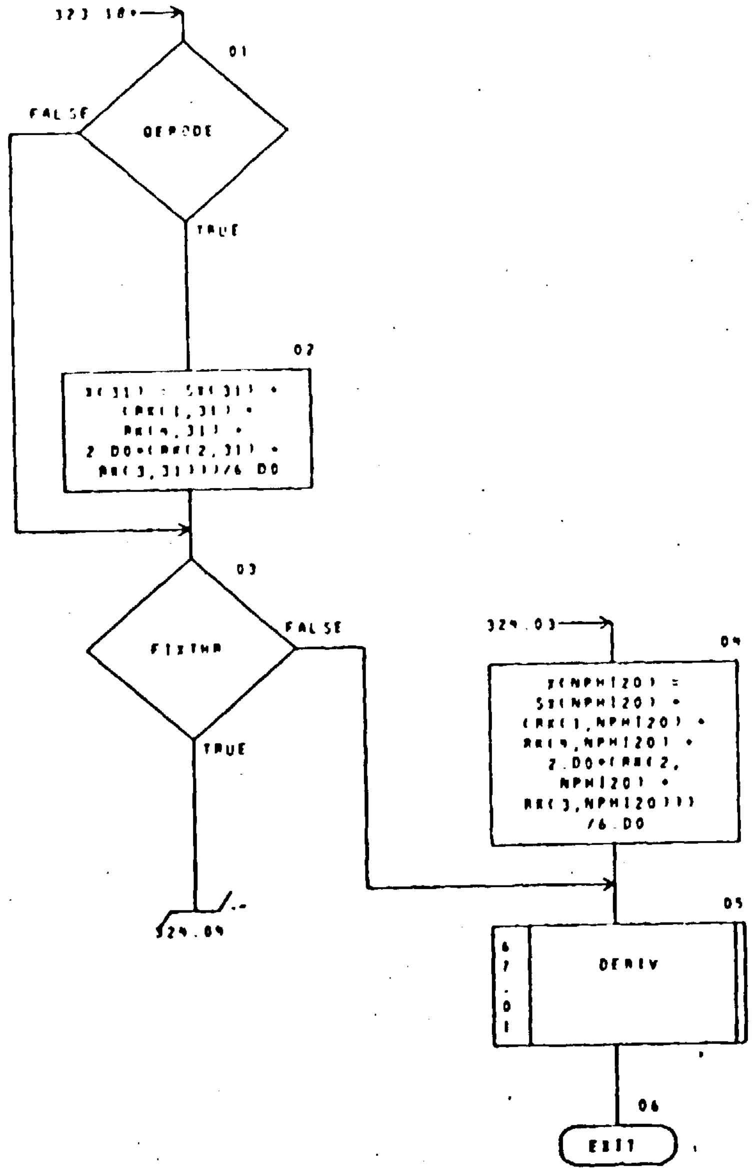


CHART TITLE - SUBROUTINE RKSTEP(STEP1,GO)



ORIGINAL PAGE IS
OF POOR QUALITY

RKSTEP-9

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL ERCODE,FIXTHR,GO,OFRODE
DIMENSION RR(4,50)
COMMON /REAL8/ R01(1200),SX(50),X(50),XD(50),SBETA,BETA,STAU,TAU,
R02(646)
COMMON /INTGR4/ I01(326),NPHI2C,I02(673)
COMMON /LOGIC4/L01(10),FIXTHR,L02(8),ERCODE,L03(12),OFRODE,L04(467)
```

Name: SCOMP

Calling Argument: ALPHA, Z, S

Referenced Sub-programs: None

Referenced Commons: None

Entry Points: None

Referencing Sub-programs: ANSTEP, IMPULS

Discussion: This subroutine evaluates the functions $G_i(\theta^2)$ required for the solution of the two-body Kepler problem. The parameter θ is related to the increment in universal anomaly $\Delta\beta$ through the formula

$$\theta^2 = (\Delta\beta)^2/a \tag{1}$$

where a is the semi-major axis. Thus, for elliptic orbits, θ represents the change in eccentric anomaly from the reference position. Denoting

$$\alpha = -\theta^2 \tag{2}$$

then the functions G_i are defined

$$F_i = \sum_{j=0}^{\infty} \frac{\alpha^j}{(2j+i)!} \tag{3}$$

$$G_i = (\Delta\beta)^i F_i . \tag{4}$$

Inspection of equation (3) will verify that the F_i satisfy the following recursion formula

$$F_i = \frac{1}{i!} + \alpha F_{i+2} . \tag{5}$$

Thus the series expression (3) need be solved for only the two highest order terms F_i after which the lower order terms are obtained with (5). The number of terms required in (3) to maintain the desired accuracy is dependent upon the magnitude of α. It has been determined that ten terms yield 16 digits of accuracy for $|\alpha| \leq 1$. For larger values of $|\alpha|$, reduction formulae permit the accurate computation of the F_i with 10 terms of the series. The procedure is to divide the input α by 4 a number of times n until the result is less than 1. Denote this result α', then the equations for $F_i(4\alpha')$ in terms of $F_i(\alpha')$ are,

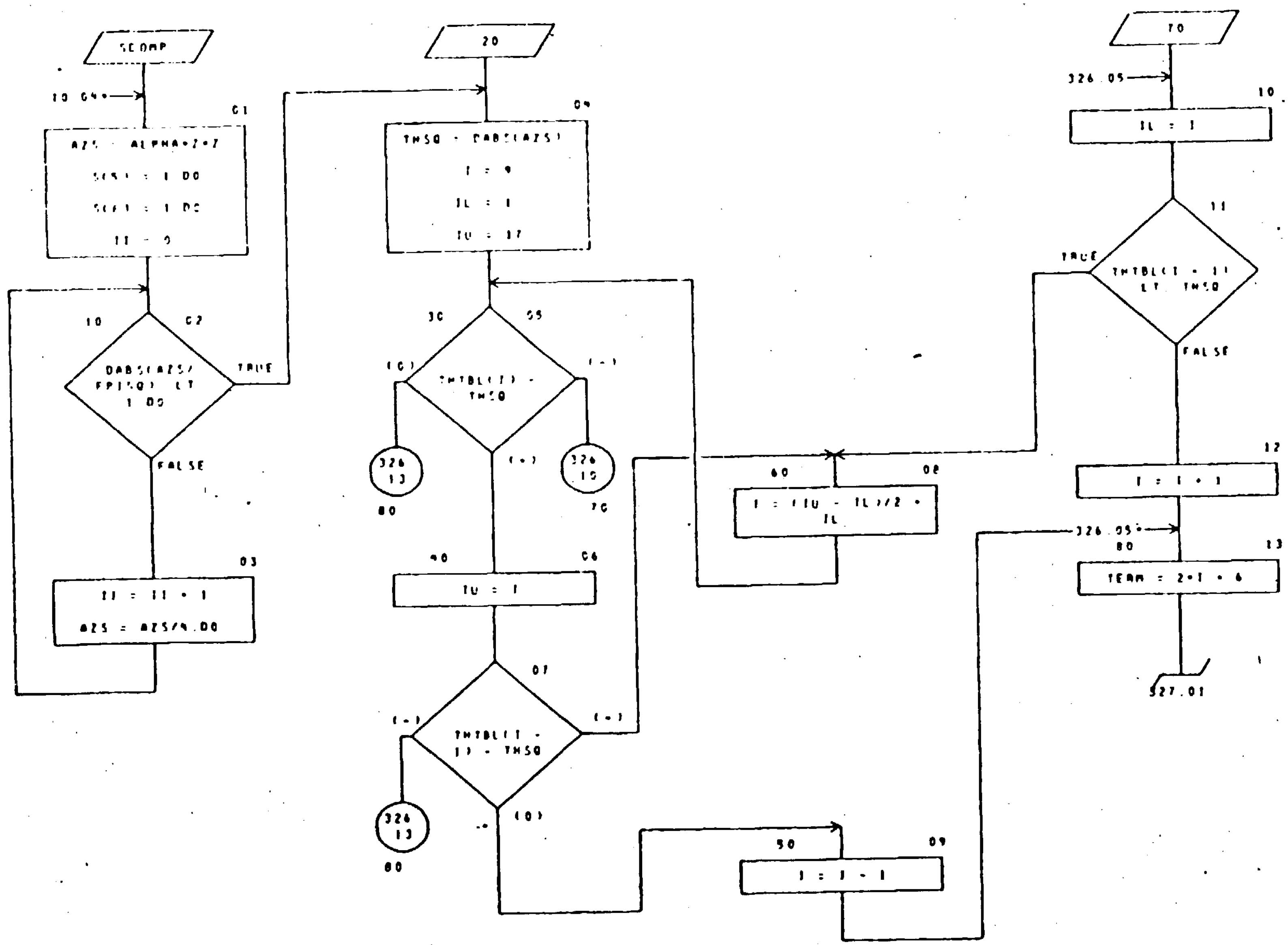
$$
\begin{aligned}
F_3(4\alpha') &= \left[F_2(\alpha') + F_0(\alpha')\,F_3(\alpha')\right]/4 \\
F_2(4\alpha') &= F_1(\alpha')^2/2 \\
F_5(4\alpha') &= \left[F_4(\alpha') + F_2(\alpha')/6 + F_0(\alpha')\,F_5(\alpha')\right]/16 \\
F_4(4\alpha') &= \left[F_3(\alpha') + F_1(\alpha')\,F_3(\alpha')\right]/8.
\end{aligned}
\qquad (6)
$$

The lower order terms in either case are obtained from the recursion formula (5). Equations (6) are cycled n times where the α' of each cycle is the $4\alpha'$ of the preceding cycle. The desired functions G_i are then obtained from (4).

SCOMP EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
S(6)	SUX		G_i, i = 0 to 5.
Z	UX		Universal anomaly increment, $\Delta\beta$.
ALPHA	UX		Semi-major axis, a, in AU.

CHART TITLE - SUBROUTINE SCOMP(ALPHA,Z,S)



SCOMP-3

CHART TITLE - SUBROUTINE SCOMP(ALPHA,Z,S)

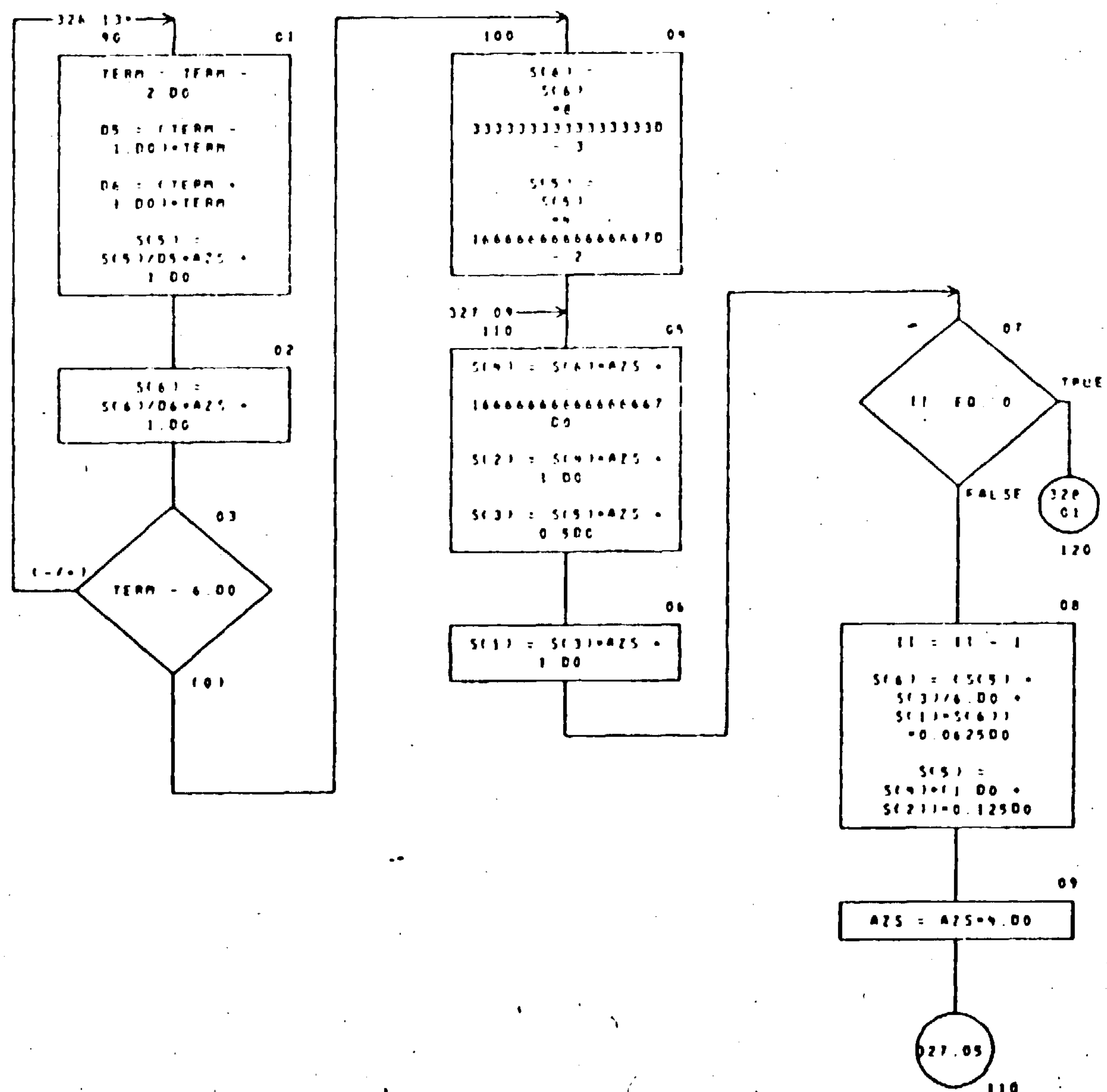
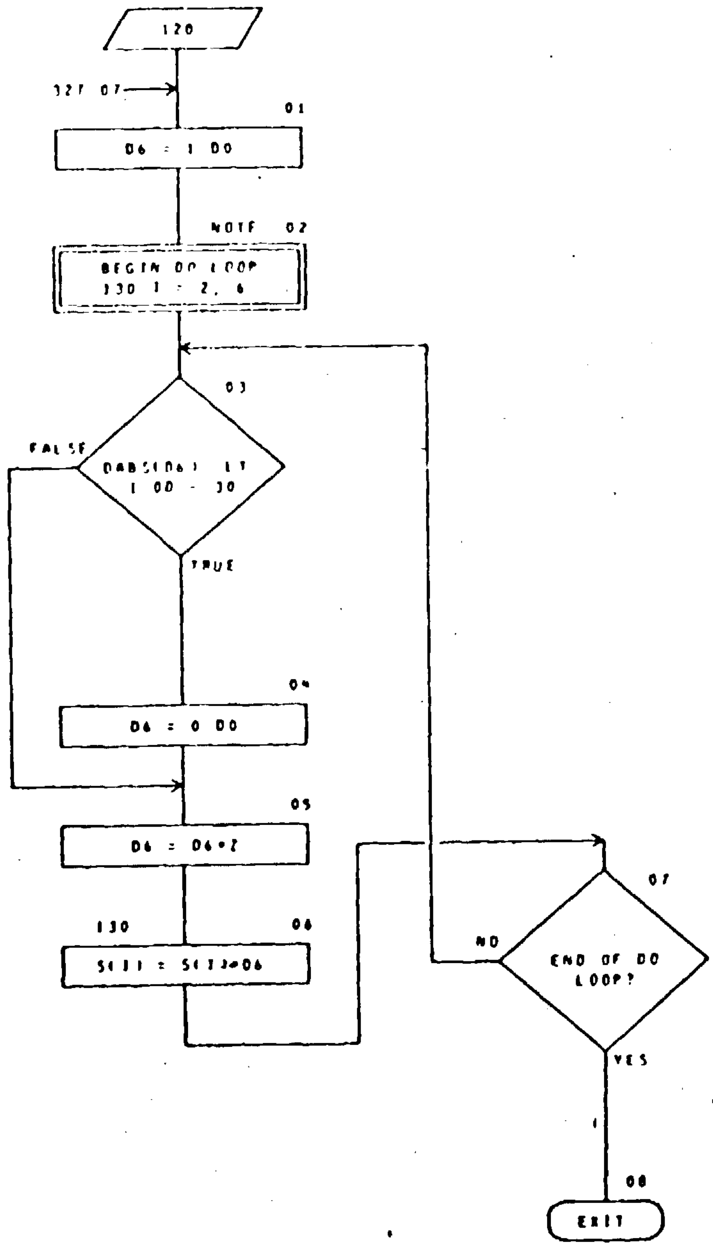


CHART TITLE - SUBROUTINE SCOMP(ALPHA,Z,S)



SCOMP-5

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION S(6),THTBL(17)
      DATA THTBL /0.D0,0.0002D0,0.0050D0,0.0406D0,0.1727D0,0.5027D0,
     1 1.5100D0,2.2407D0,3.8863D0,6.1895D0,9.2377D0,13.1040D0,17.8496D0,
     2 3.5244D0,30.1697D0,37.8193D0,46.5006D0/
      DATA FPISQ /39.478417604357D0/
```

<u>Name:</u> SETUP

<u>Calling Argument:</u> None

<u>Referenced Sub-programs:</u> None

<u>Referenced Commons:</u> INTGR4, ITERAT, LOGIC4

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> QSTART

<u>Discussion:</u> Subroutine SETUP "sets up" (initializes) the logical variables which are used (many times during an iteration sequence) in subroutine GETQ. Generally, the name of a logical variable in this subroutine begins with "A", which is followed by a number, and terminated by an alphabetic character, A, B, C, etc. The number corresponds to the index of the iterator dependent variable, and the alphabetic character corresponds to the trigger-setting of that variable, in accordance with the description of program input quantities Yi. Thus, for example, A11C corresponds to the eleventh (11) input iterator dependent variable (Y11), and in particular to the third (C) trigger setting of Y11, which is p_{ref} (specified reference power).

<u>Messages and printouts:</u> Should any Y17 triggers (indicators) be turned-on (non-zero) but the related program input quantity LAUNCH be zero, the diagnostic message is printed on units 6 and 12:

WARNING. TRIGGERS NO. 17 SHOULD BE ZERO WHEN LAUNCH = 0

<u>SETUP EXTERNAL VARIABLES TABLE</u>

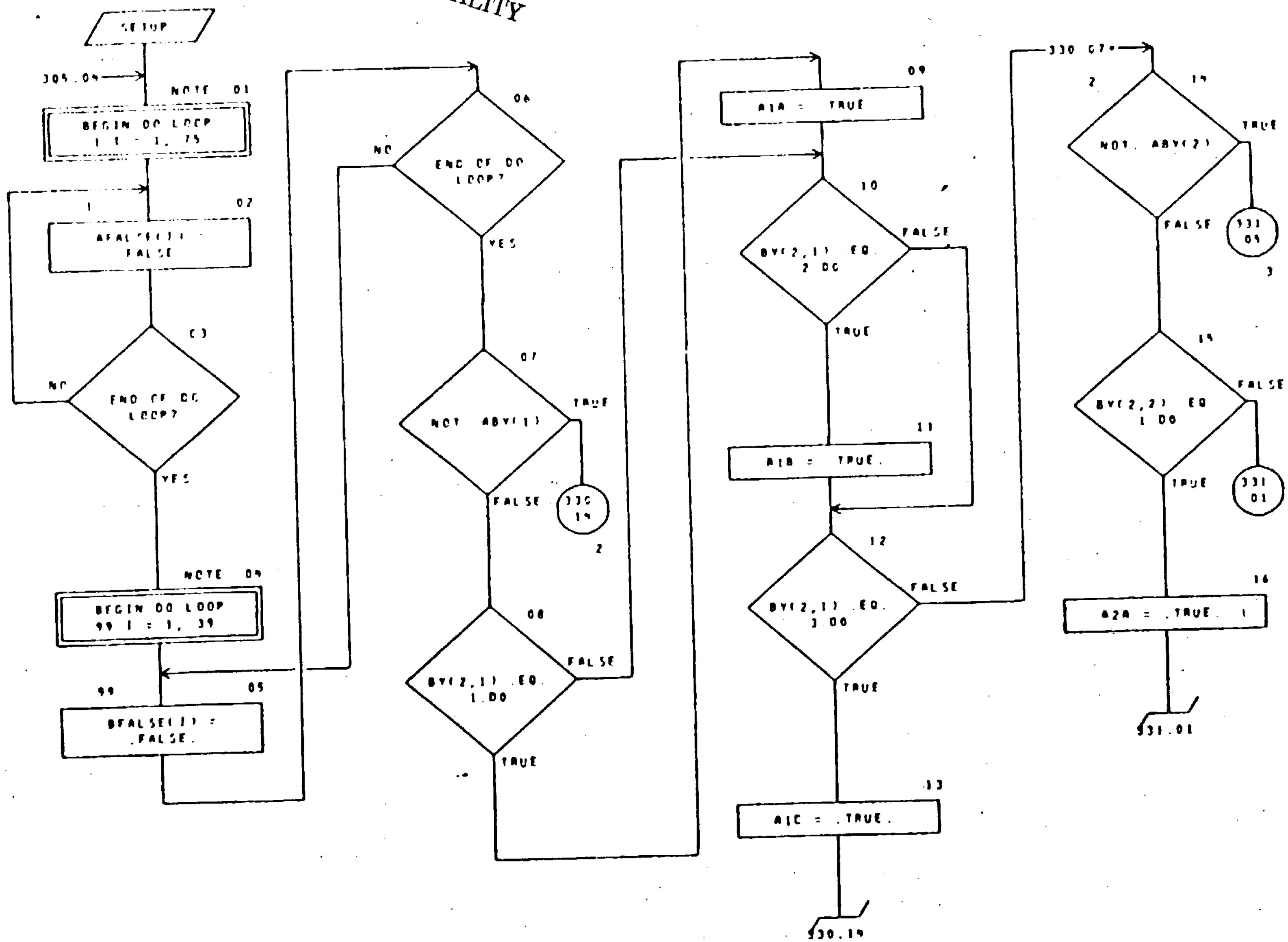
Variable	Use	Common	Description
BX(5,70)	U	ITERAT	Array of iterator independent-variable values and controls.
BY(5,70)	U	ITERAT	Array of iterator dependent-variable values and controls.

SETUP EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ABY(70)	U	LOGIC4	Master array of iterator dependent-variable indicators.
A1A	SE	LOGIC4	Logical indicators which are initialized in subroutine SETUP and used in subroutine GETQ for computation of iterator dependent-variable values; see Discussion for association-code.
A1B	S	LOGIC4	
A1C	S	LOGIC4	
A2A	S	LOGIC4	
.	.	.	
.	.	.	
.	.	.	
.	.	.	
A68A	S	LOGIC4	
A68B	S	LOGIC4	
A69A	S	LOGIC4	
A70A	S	LOGIC4	
APHI(2,10)	S	LOGIC4	Logical indicators associated with multiple fixed thrust-cone-angles; only the first one (21) is applicable at present.
LAUNCH	U	INTGR4	Launch mode selector.
LEGMAX	U	INTGR4	Total number of legs (trajectory segments) between start of trajectory and primary target; equals the number of targets up to and including the primary target.
LIMPHI	S	INTGR4	Highest index-value of all non-zero (multiple) fixed thrust-cone-angles; highest permissible value is currently one.

CHART TITLE - SUBROUTINE SETUP

ORIGINAL PAGE IS
OF POOR QUALITY



SETUP-3

CHART TITLE - SUBROUTINE SETUP

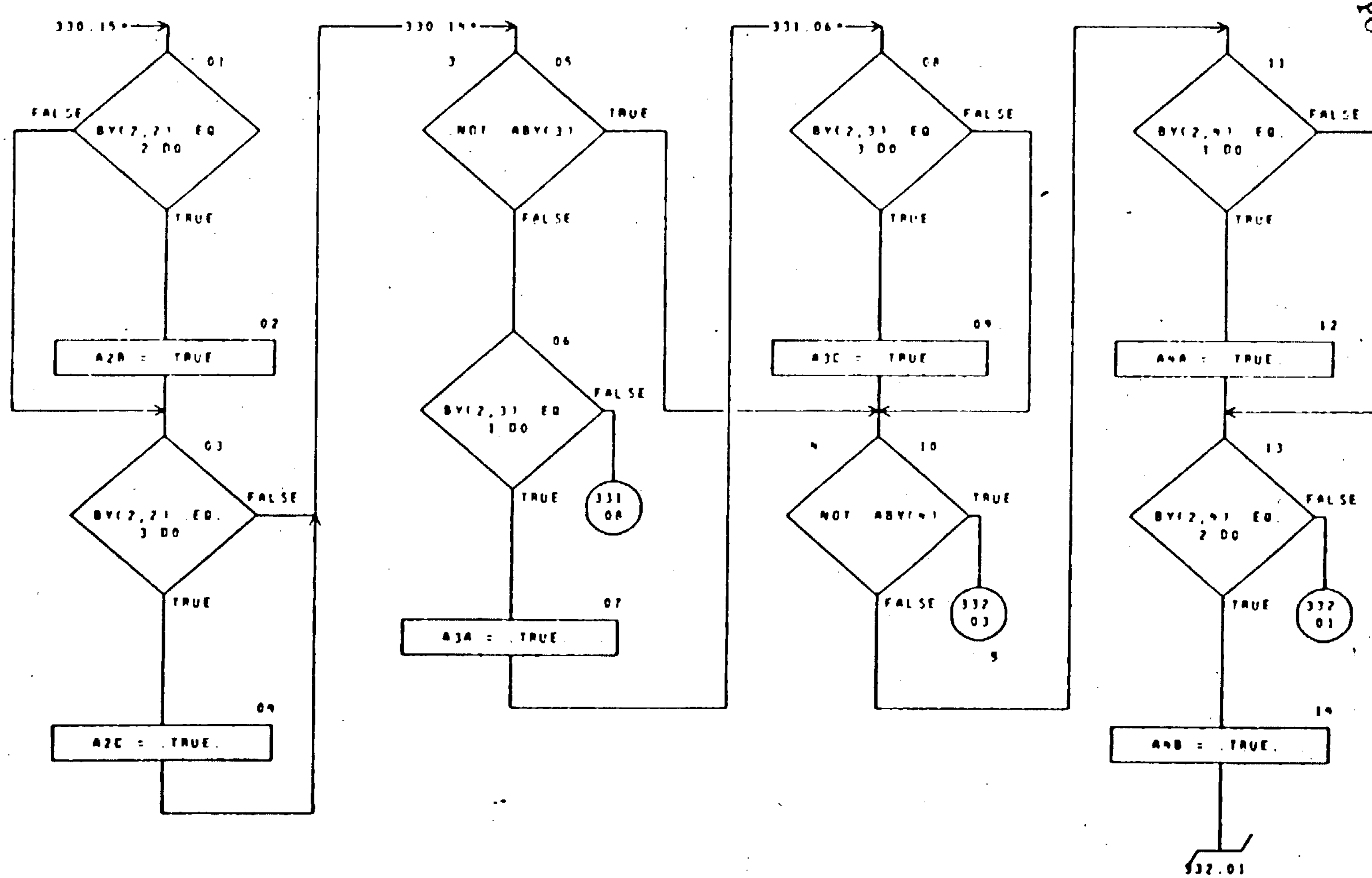


CHART TITLE - SUBROUTINE SETUP

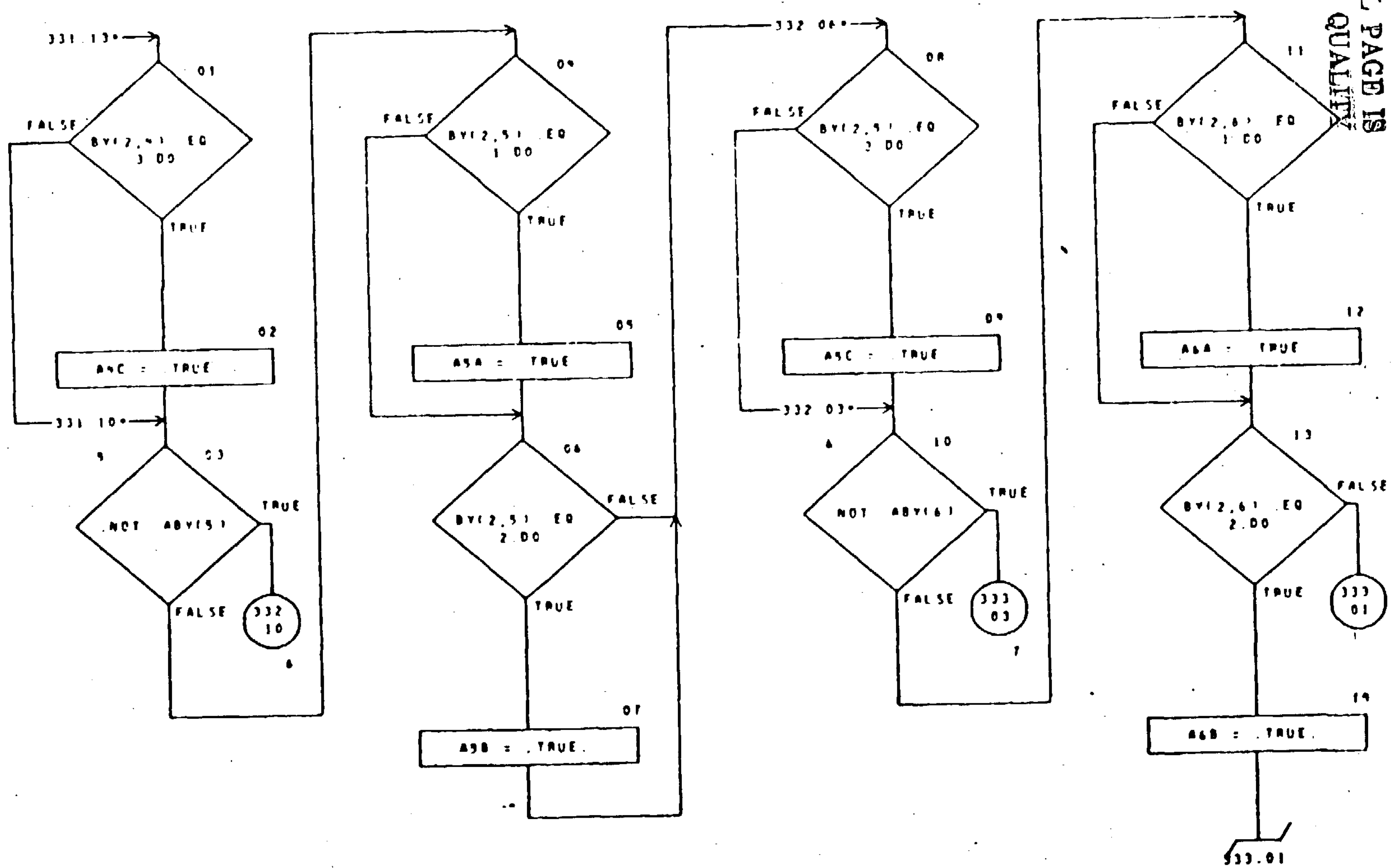


CHART TITLE - SUBROUTINE SETUP

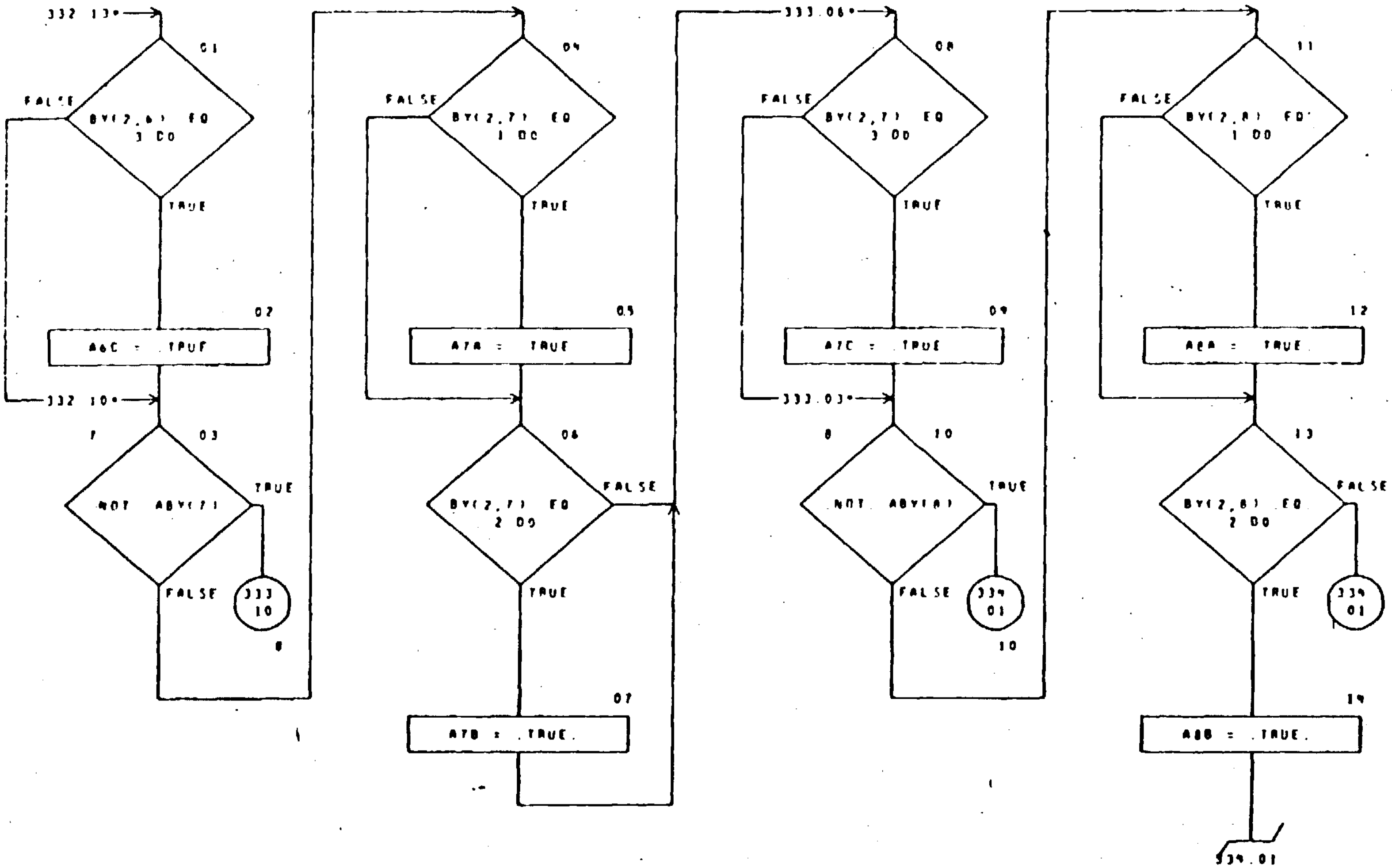


CHART TITLE - SUBROUTINE SETUP

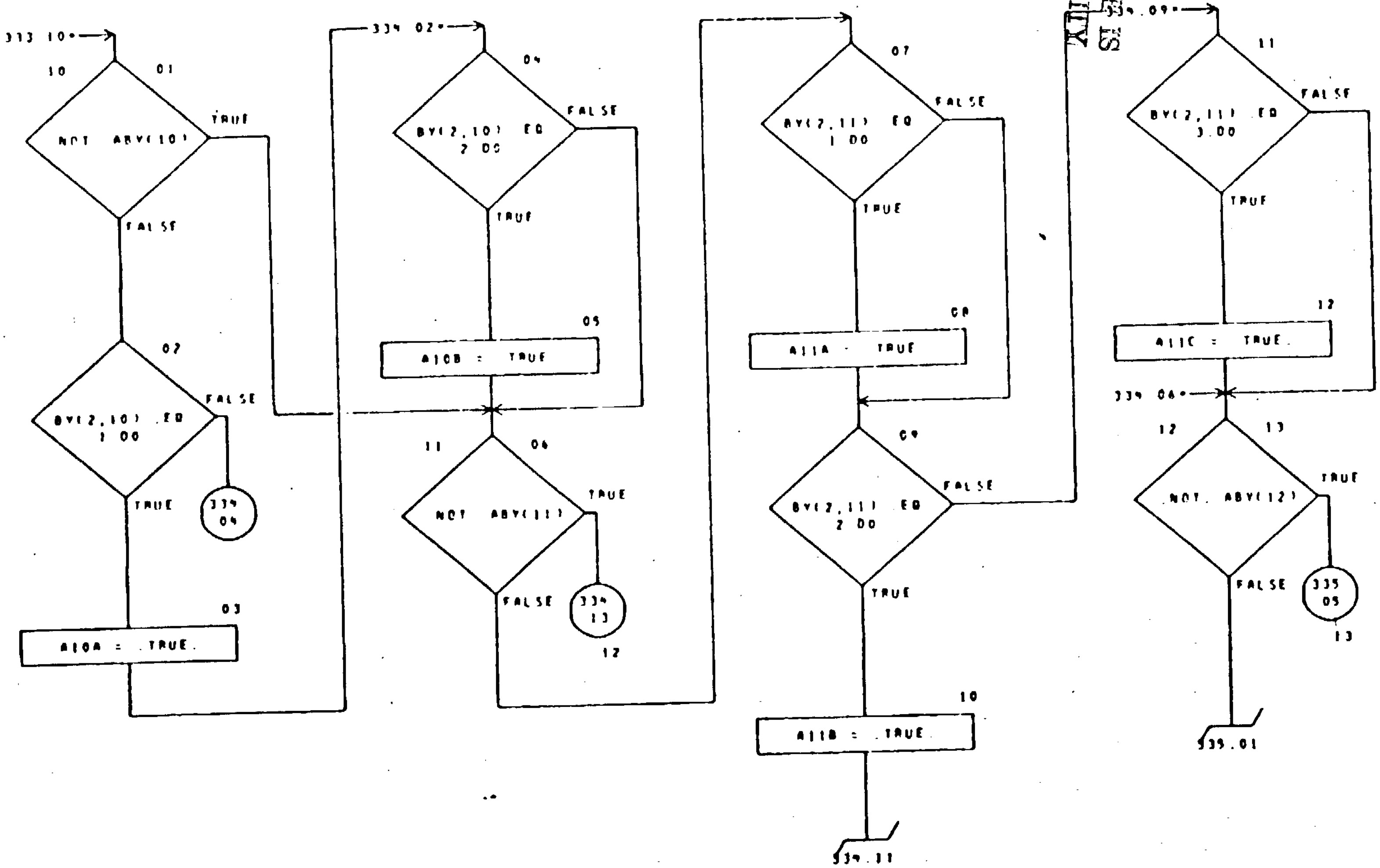


CHART TITLE - SUBROUTINE SETUP

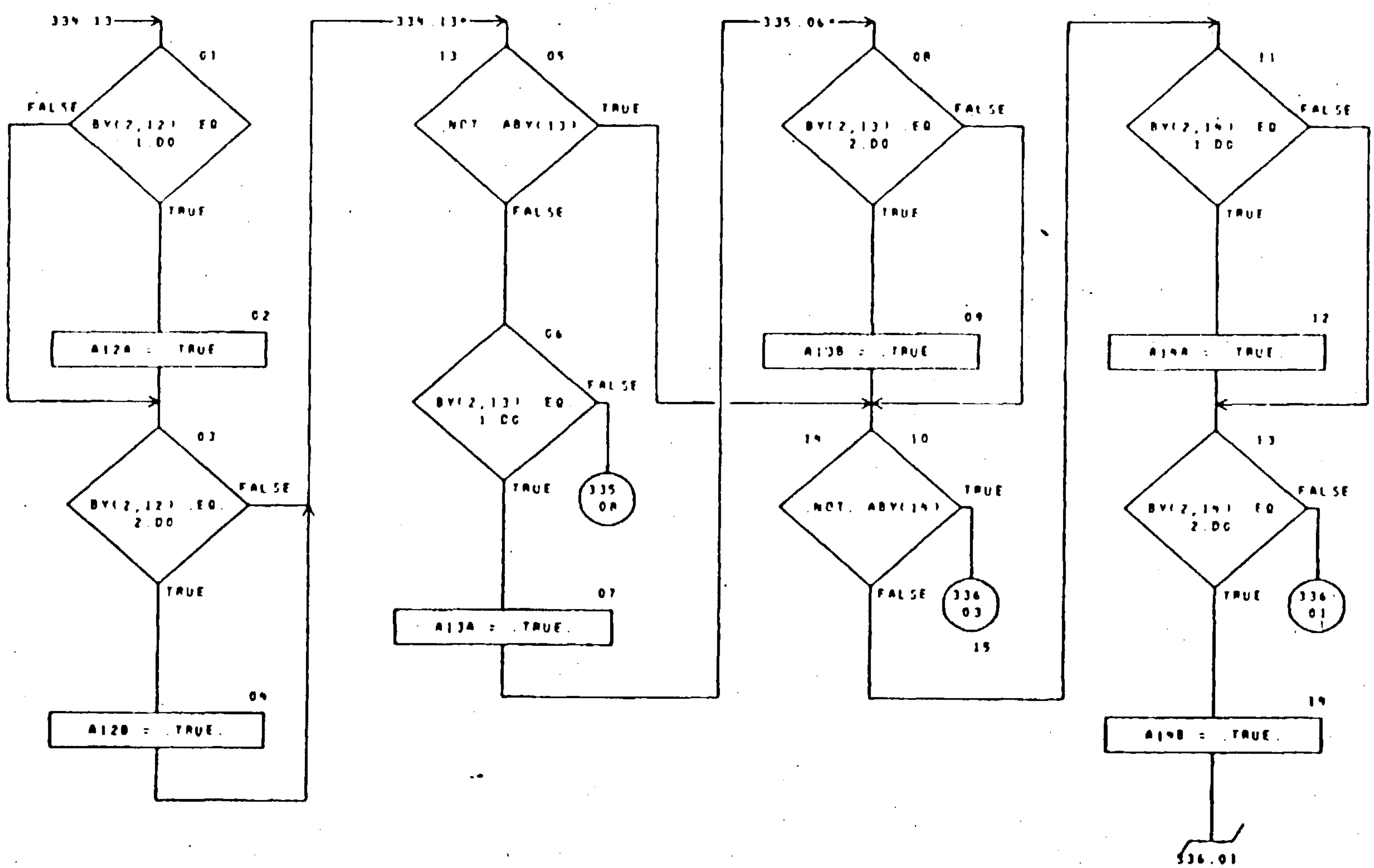


CHART TITLE - SUBROUTINE SETUP

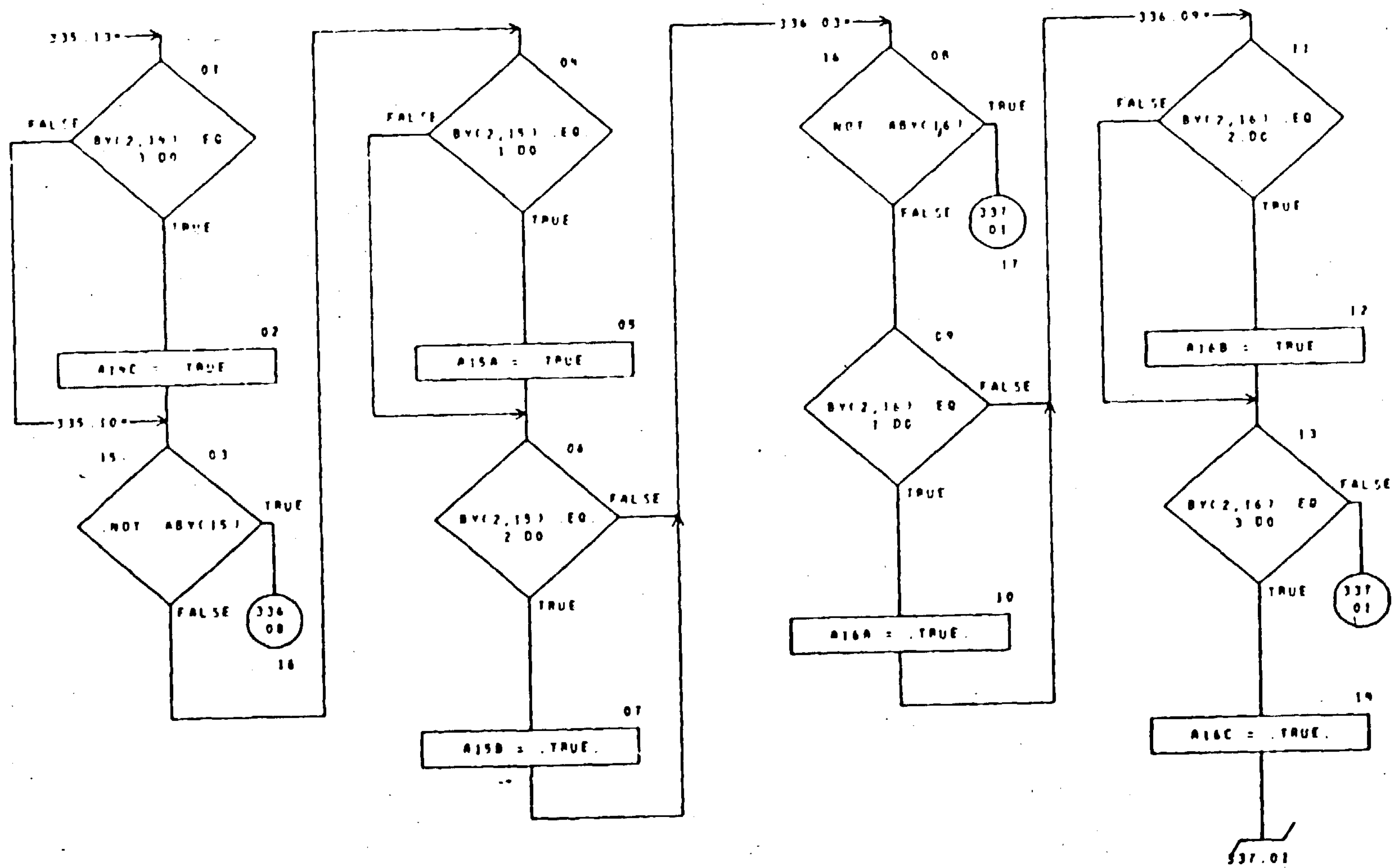


CHART TITLE - SUBROUTINE SETUP

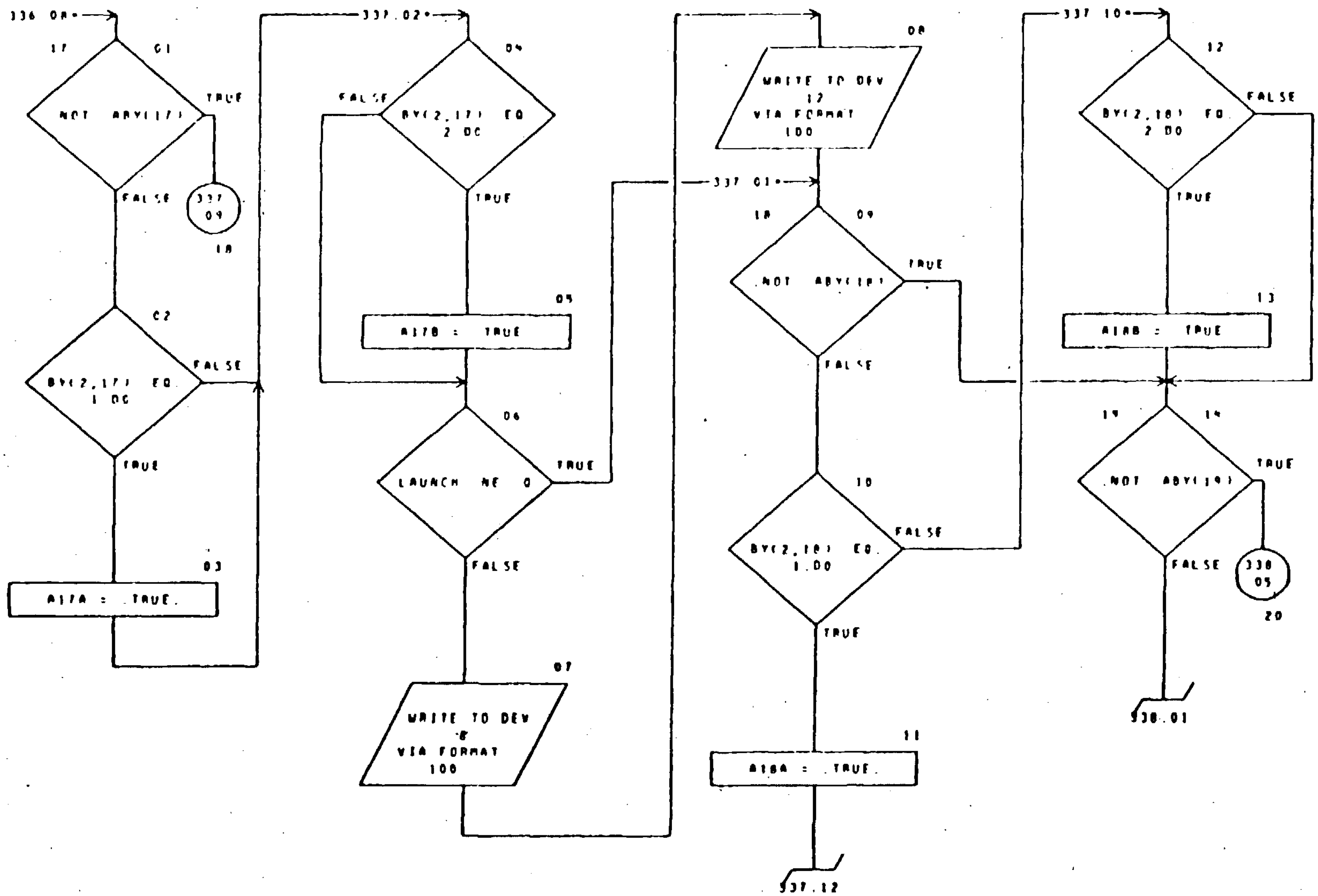
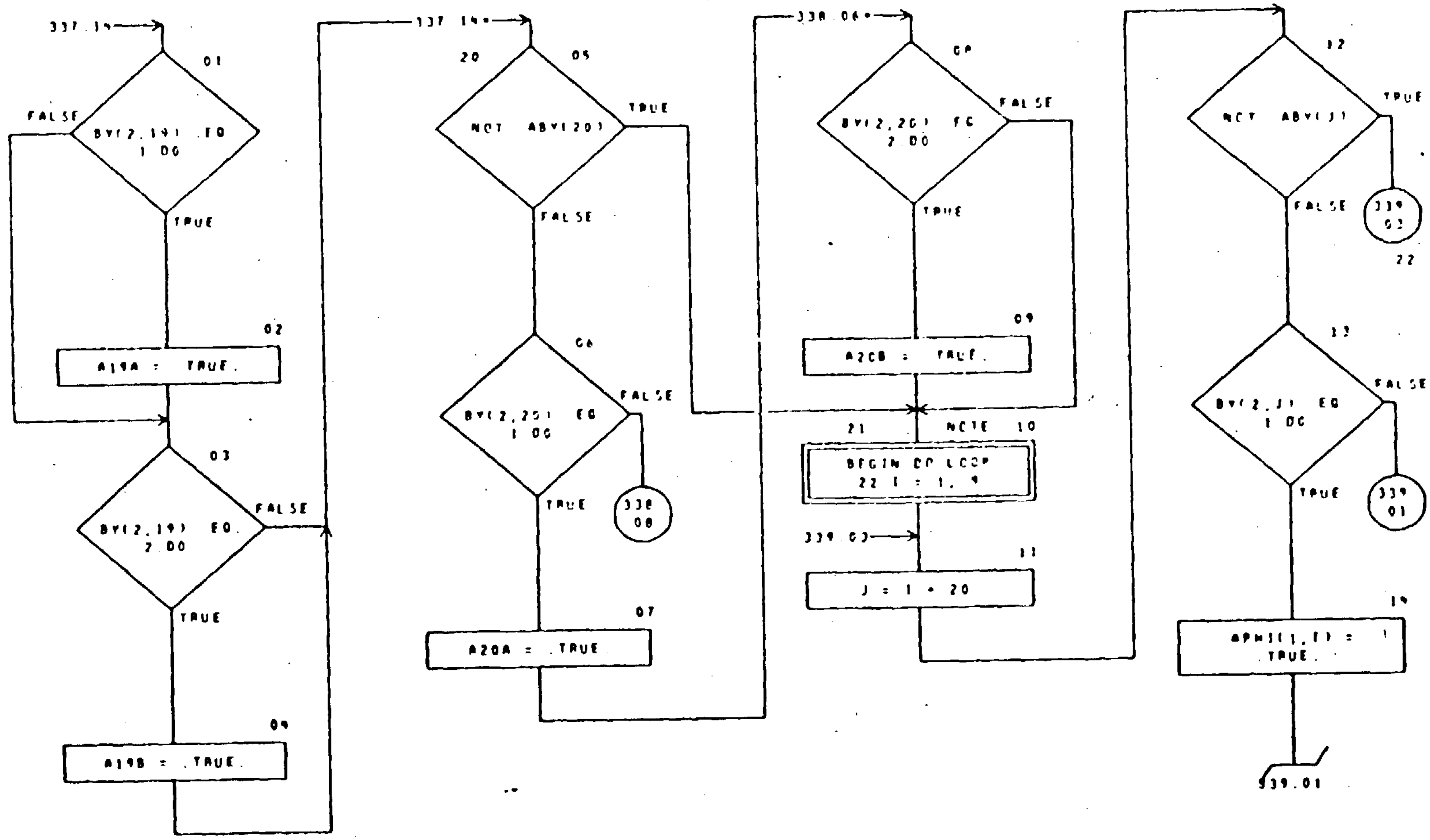


CHART TITLE - SUBROUTINE SETUP



C-7

CHART TITLE - SUBROUTINE SETUP

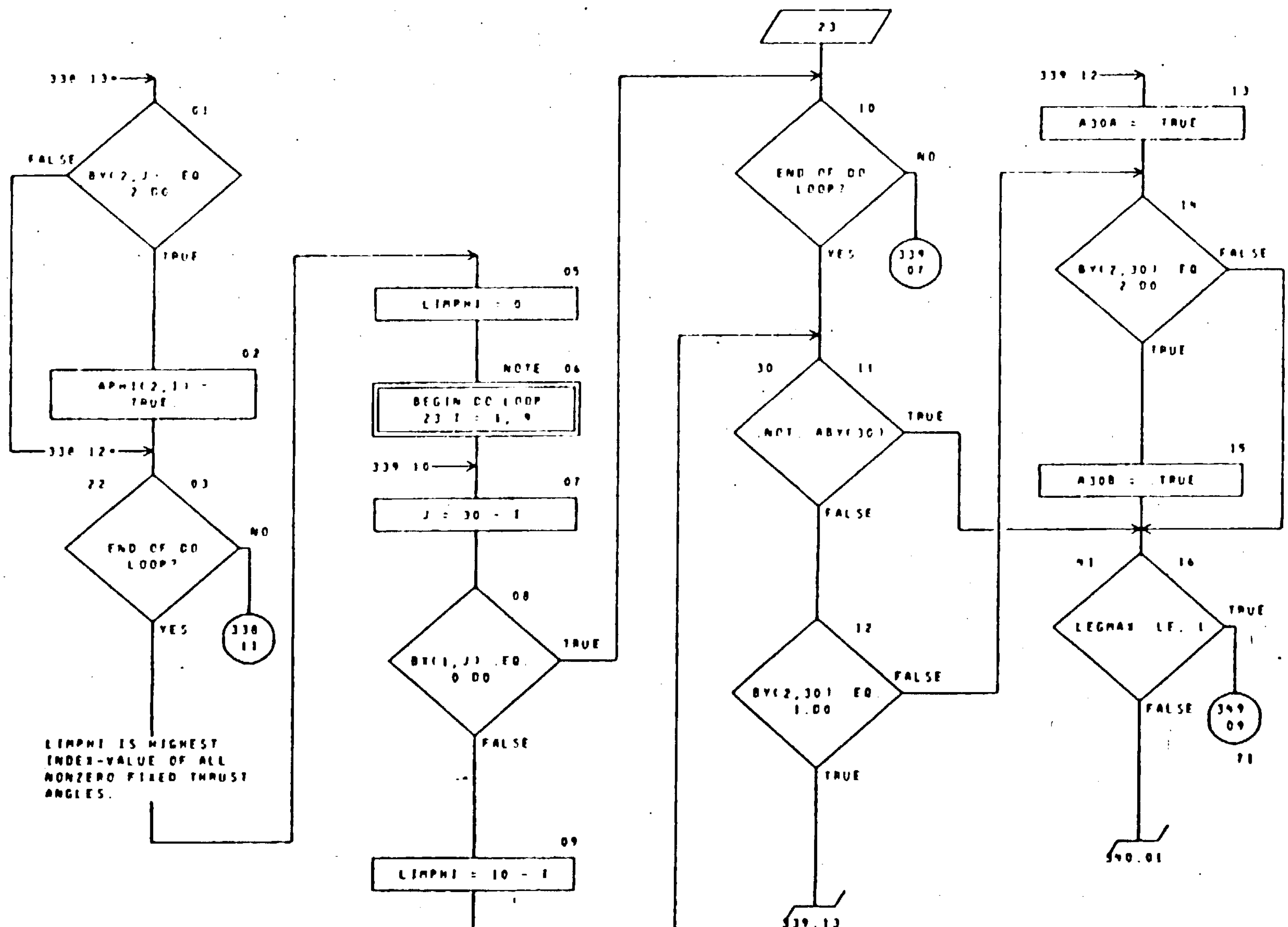


CHART TITLE - SUBROUTINE SETUP

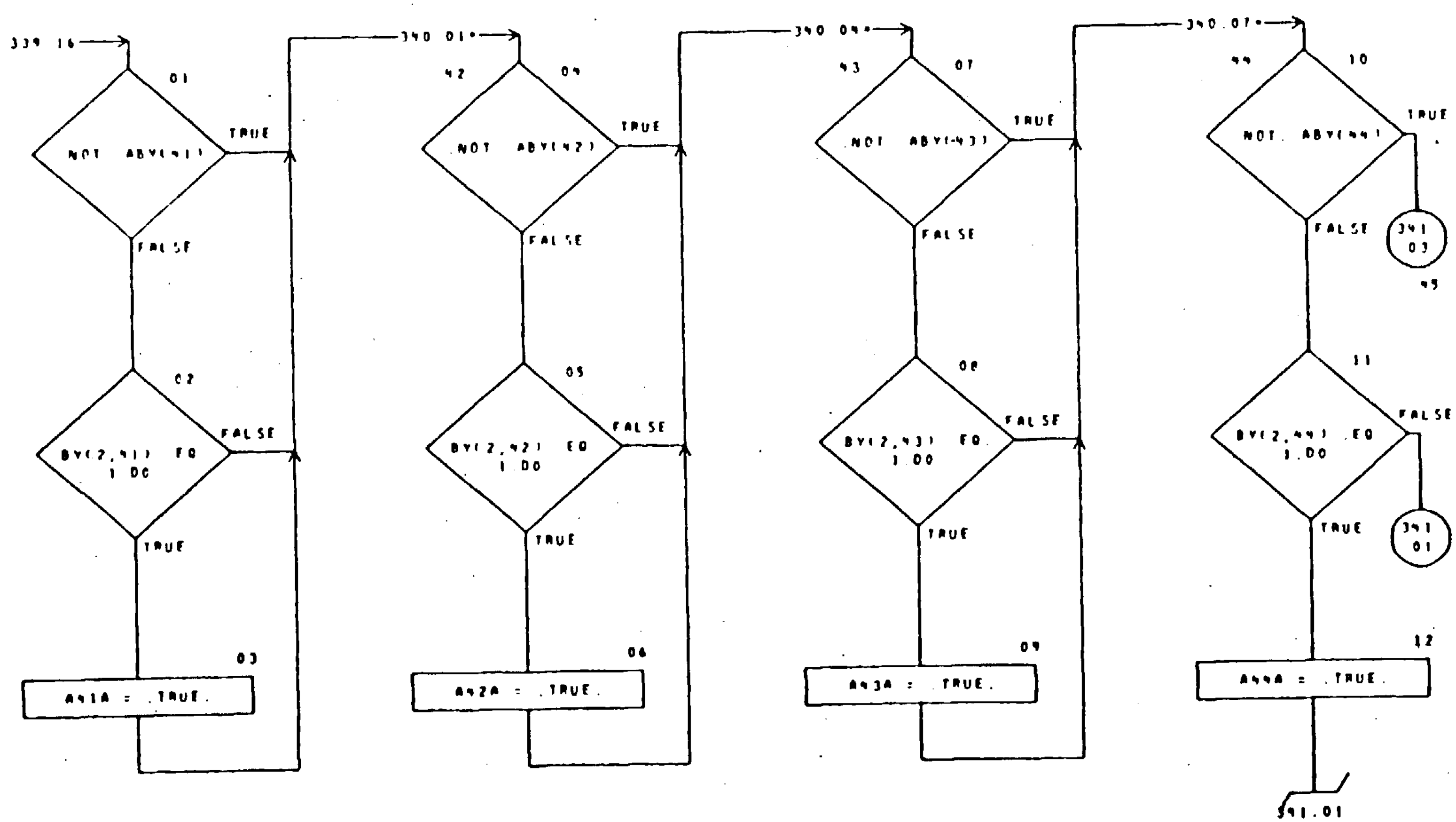


CHART TITLE - SUBROUTINE SETUP

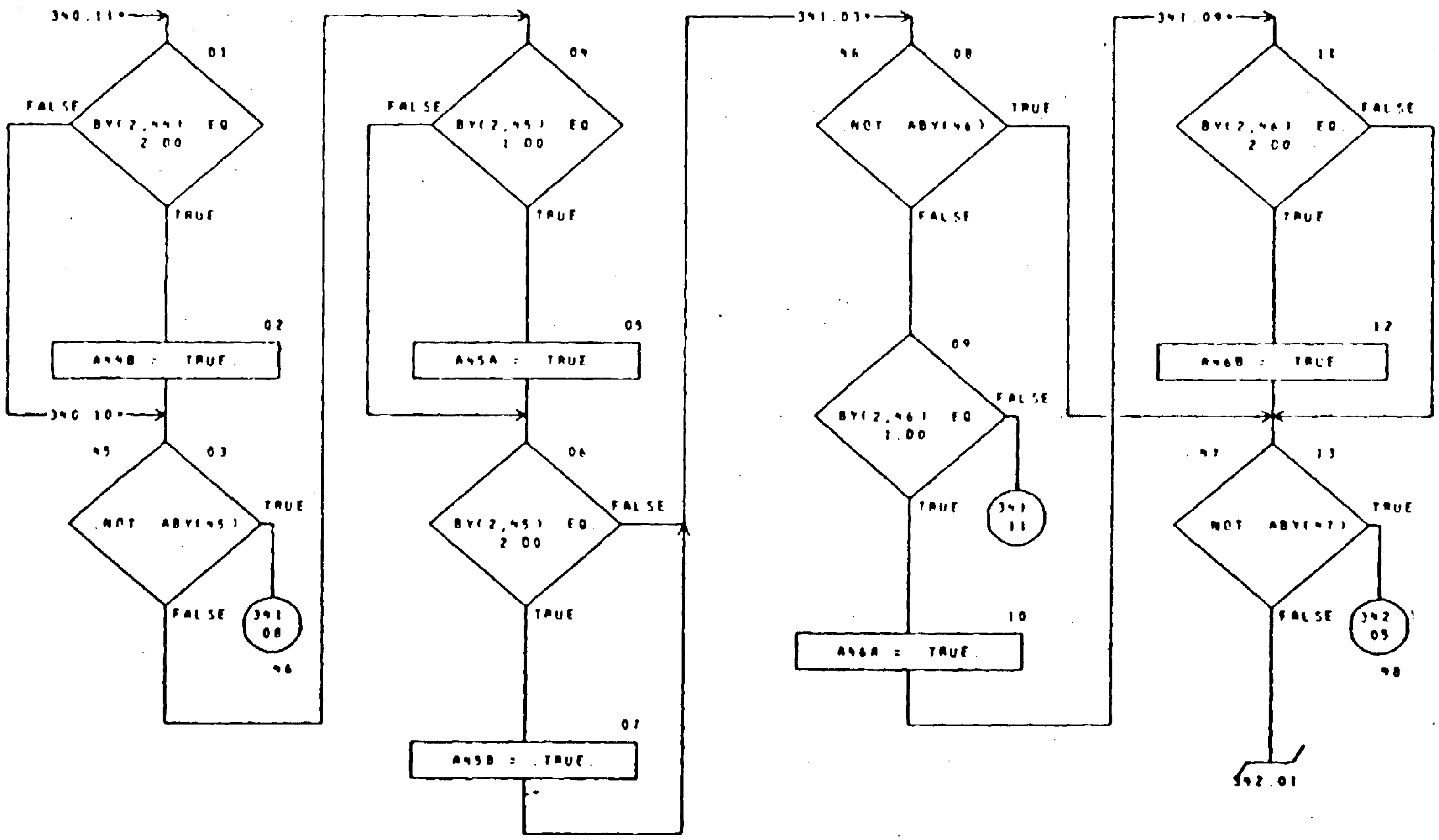


CHART TITLE - SUBROUTINE SETUP

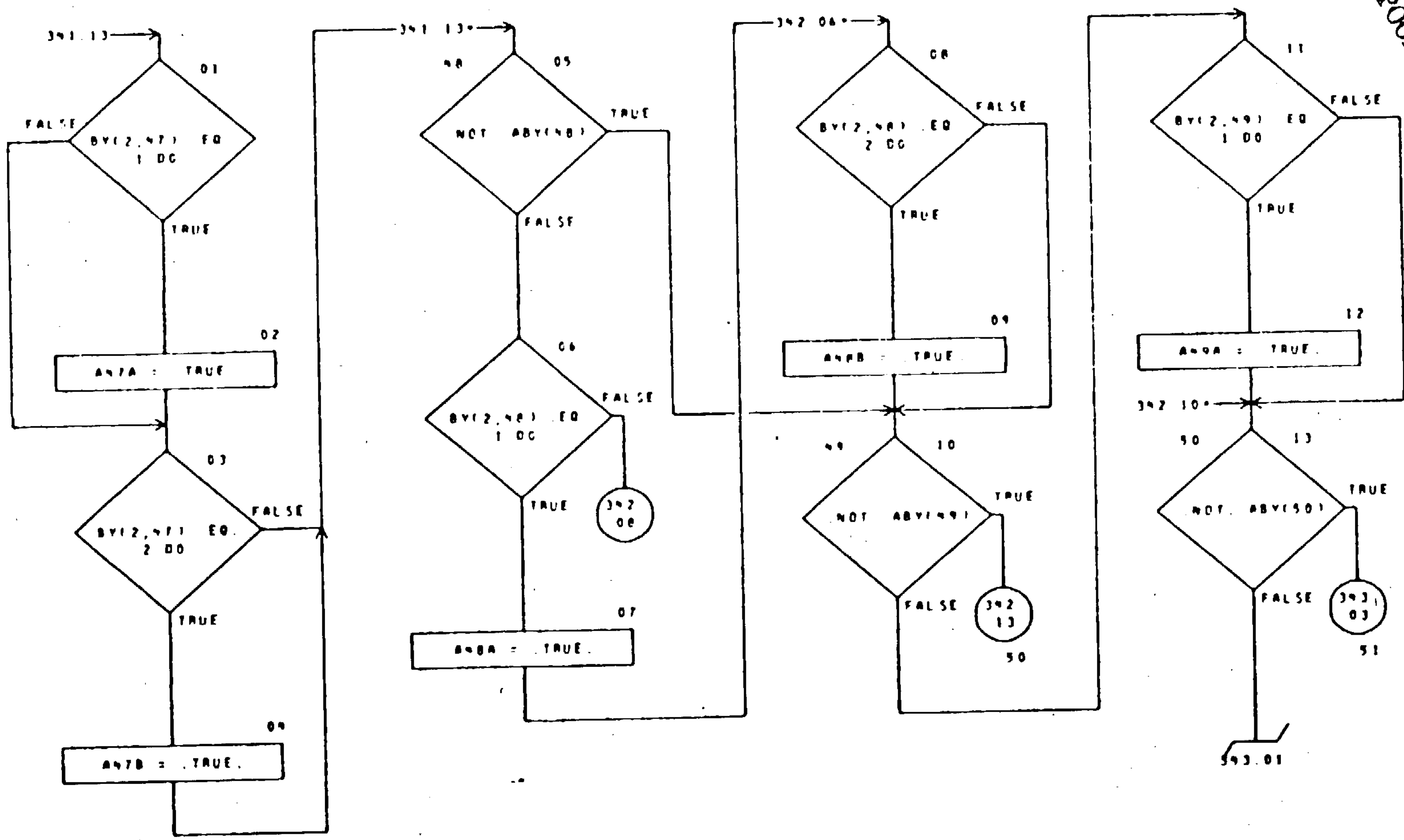


CHART TITLE - SUBROUTINE SETUP

ORIGINAL PAGE IS
OF POOR QUALITY

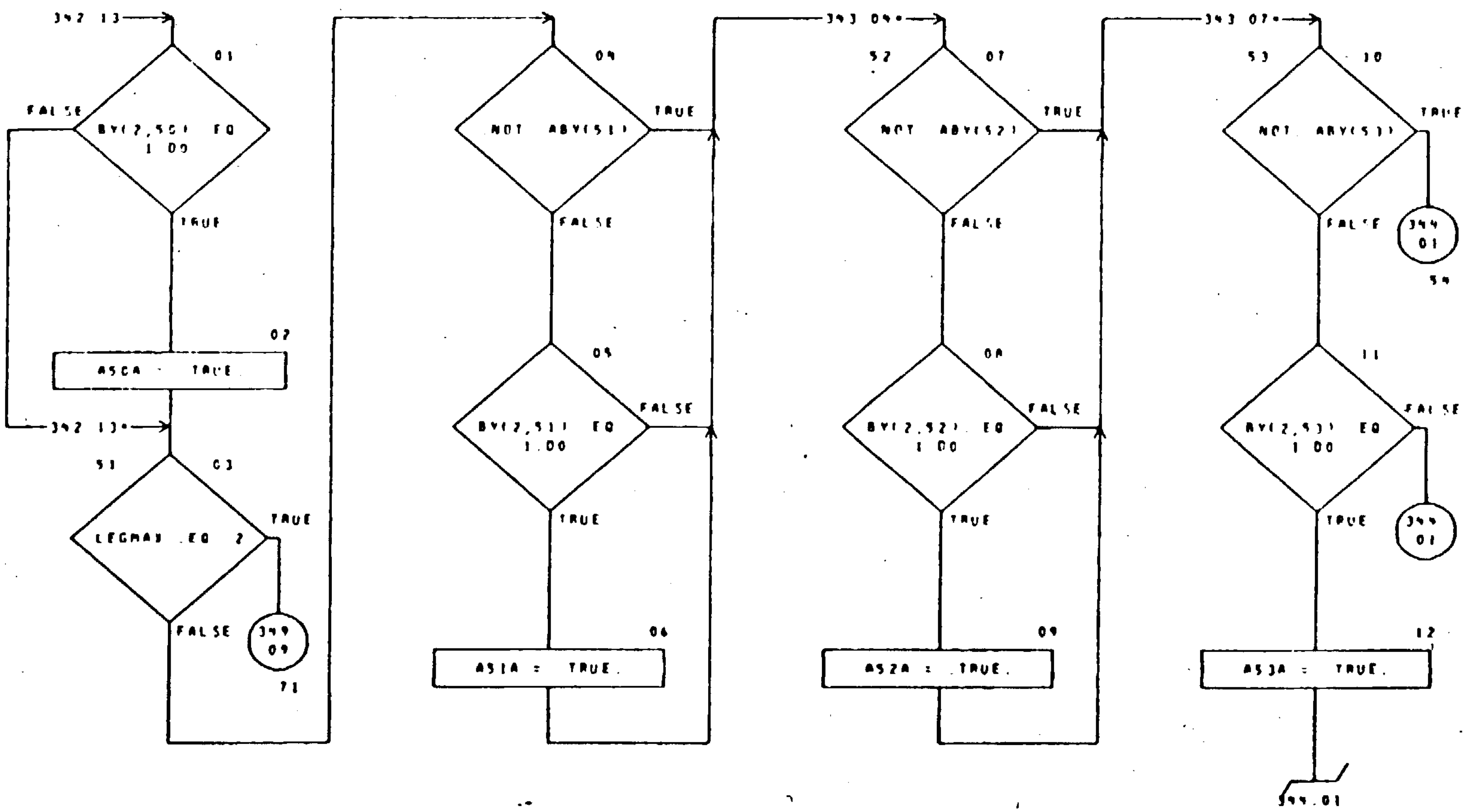


CHART TITLE - SUBROUTINE SETUP

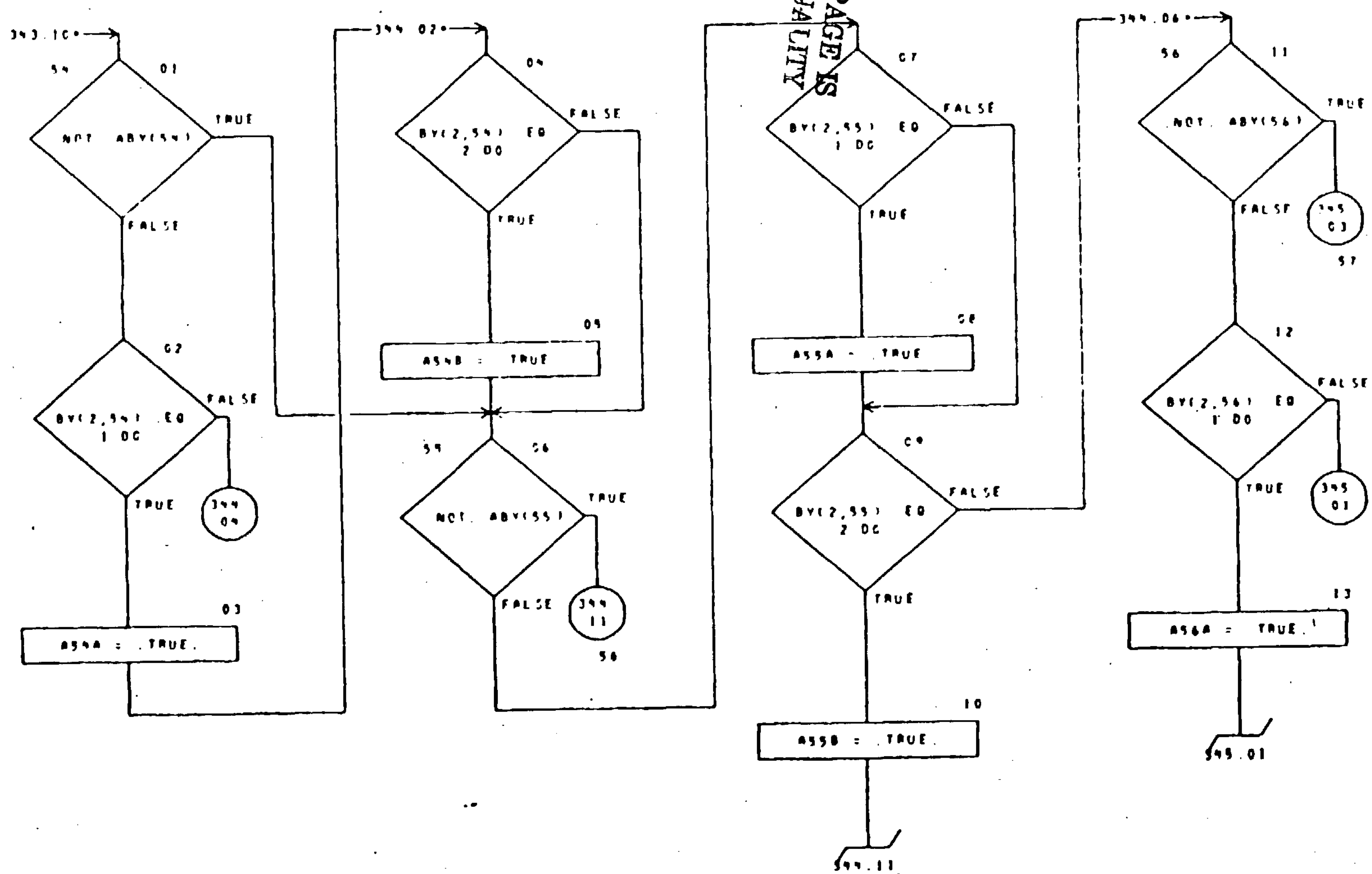
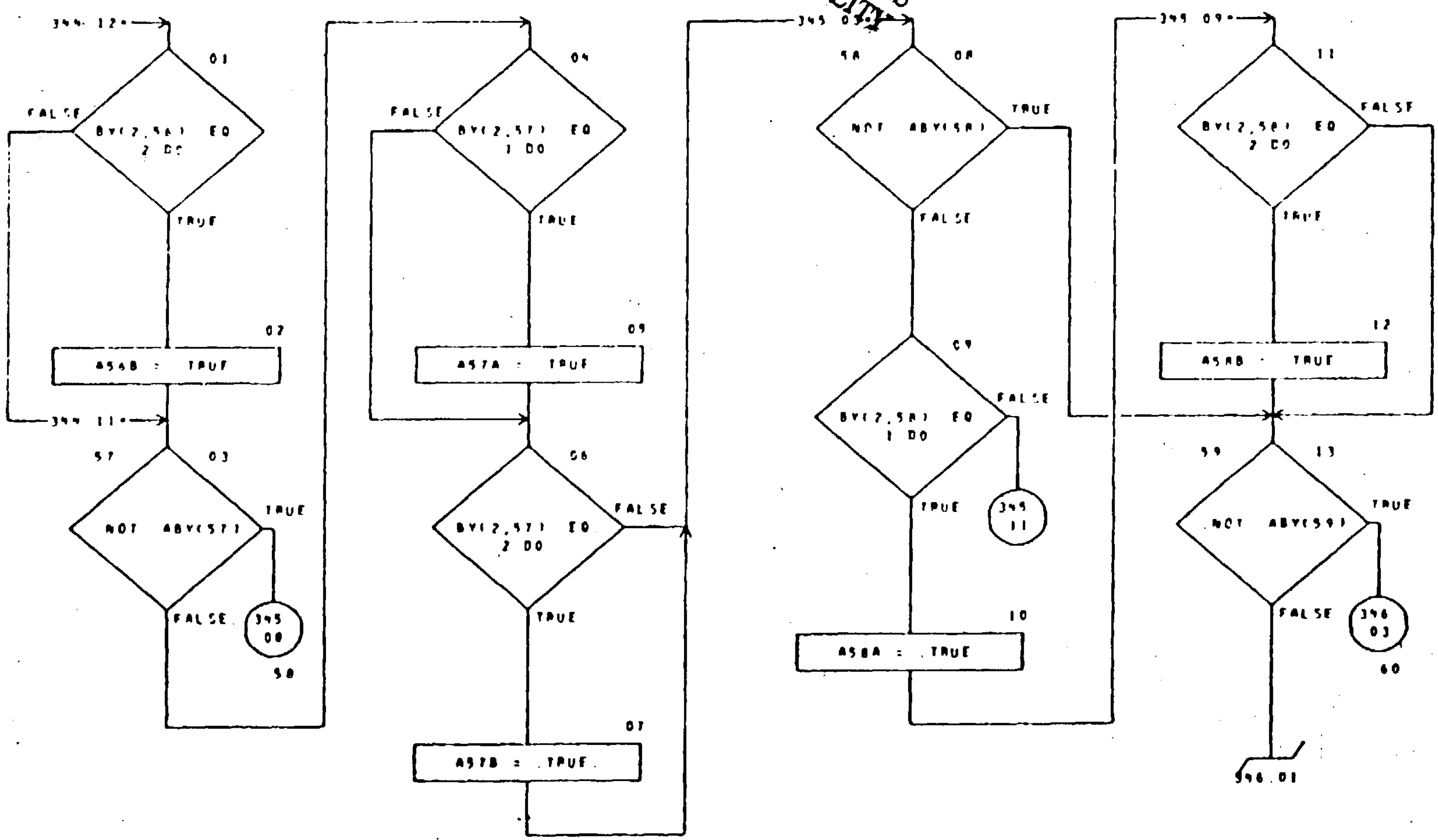
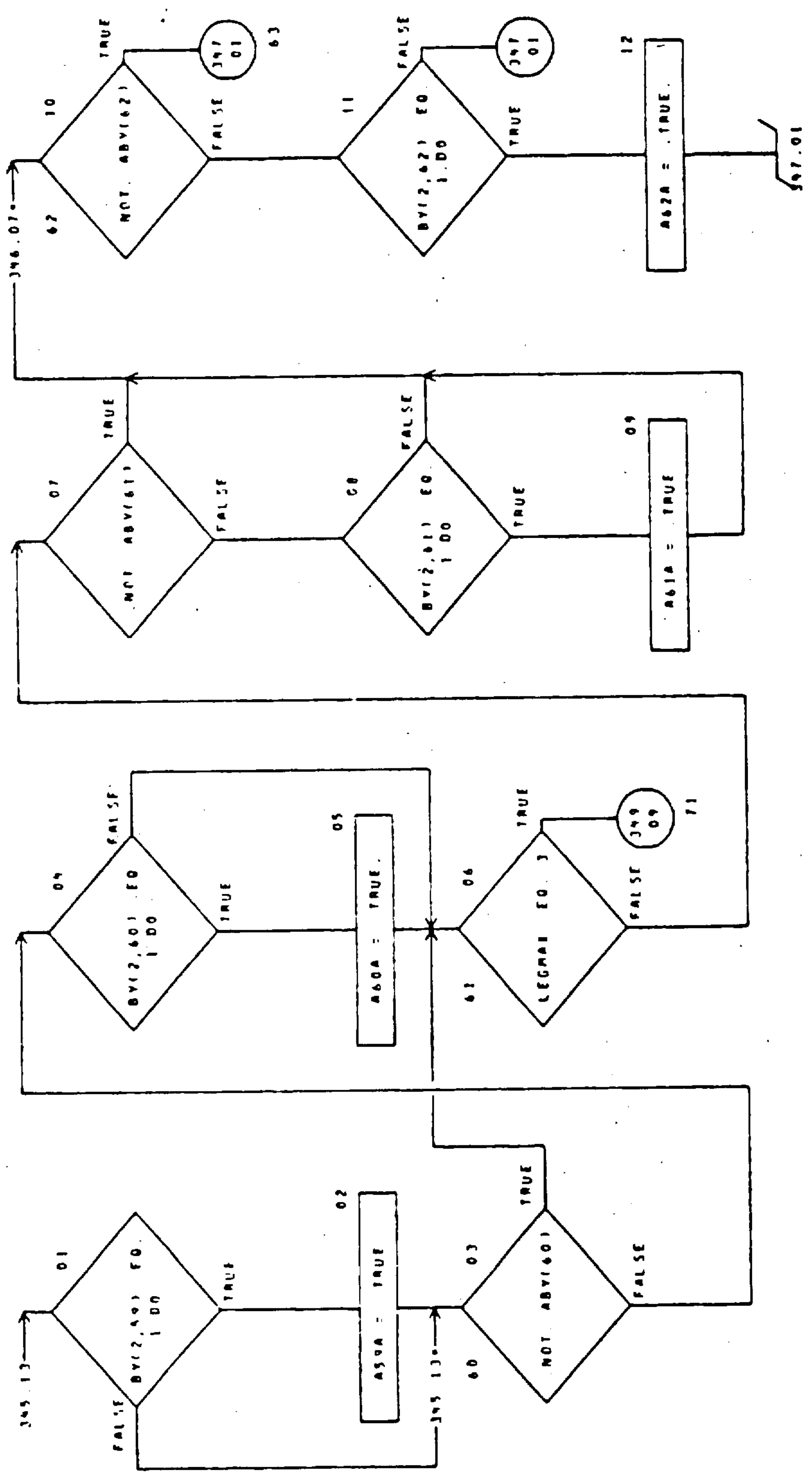


CHART TITLE - SUBROUTINE SETUP




01
BY(2,59) EQ 1.00
TRUE
FALSE
02
A59A = .TRUE.
03
NOT. ABV(60)
04
BY(2,60) EQ 1.00
05
A60A = .TRUE.
06
LEGMAX EQ. 3
07
NOT. ABV(61)
08
BY(2,61) EQ 1.00
09
A61A = .TRUE.
10
NOT. ABV(62)
11
BY(2,62) EQ 1.00
12
A62A = .TRUE.
396.07
397.01

SETUP-19

CHART TITLE - SUBROUTINE SETUP

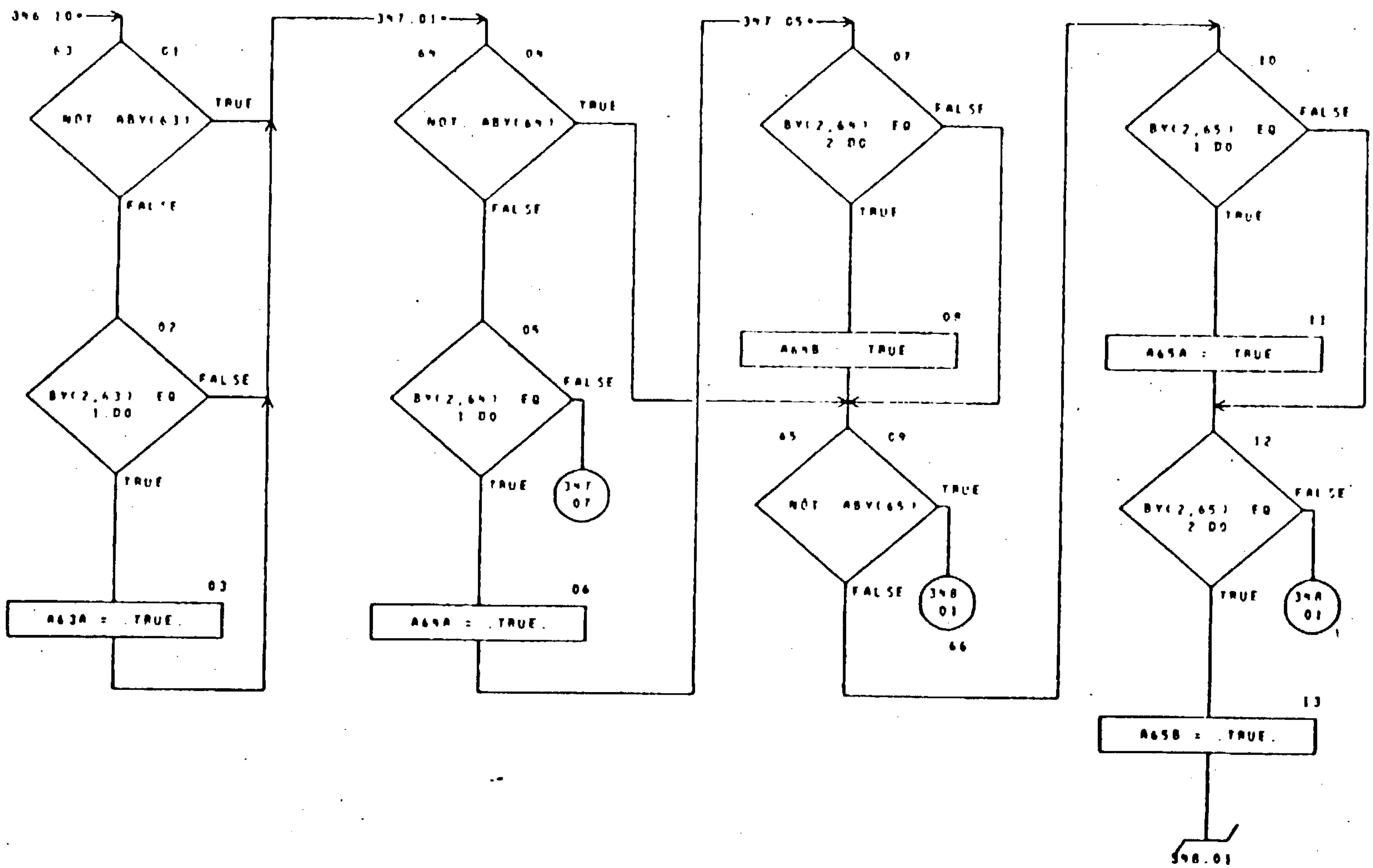


CHART TITLE - SUBROUTINE SETUP

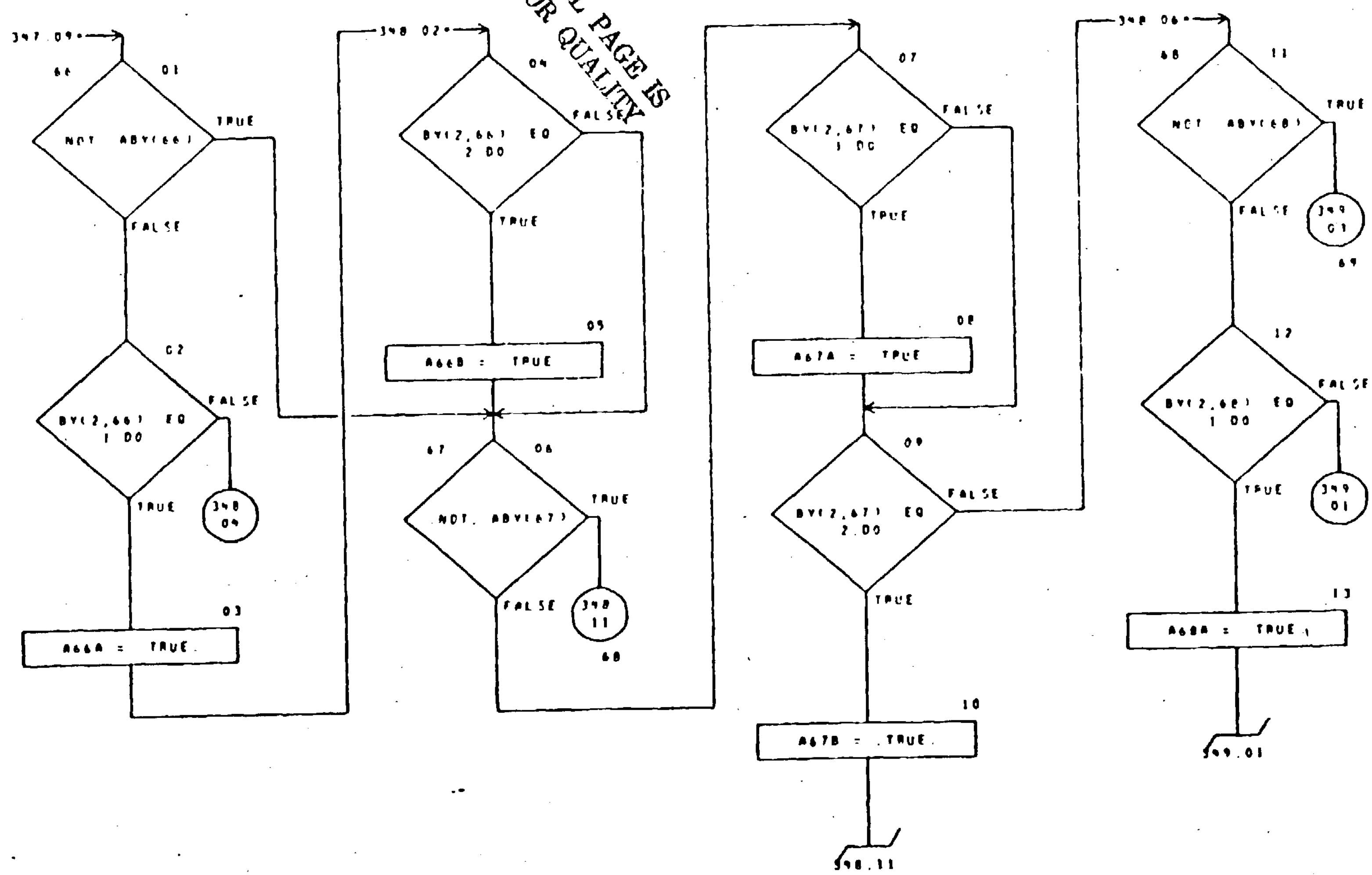


CHART TITLE - SUBROUTINE SETUP

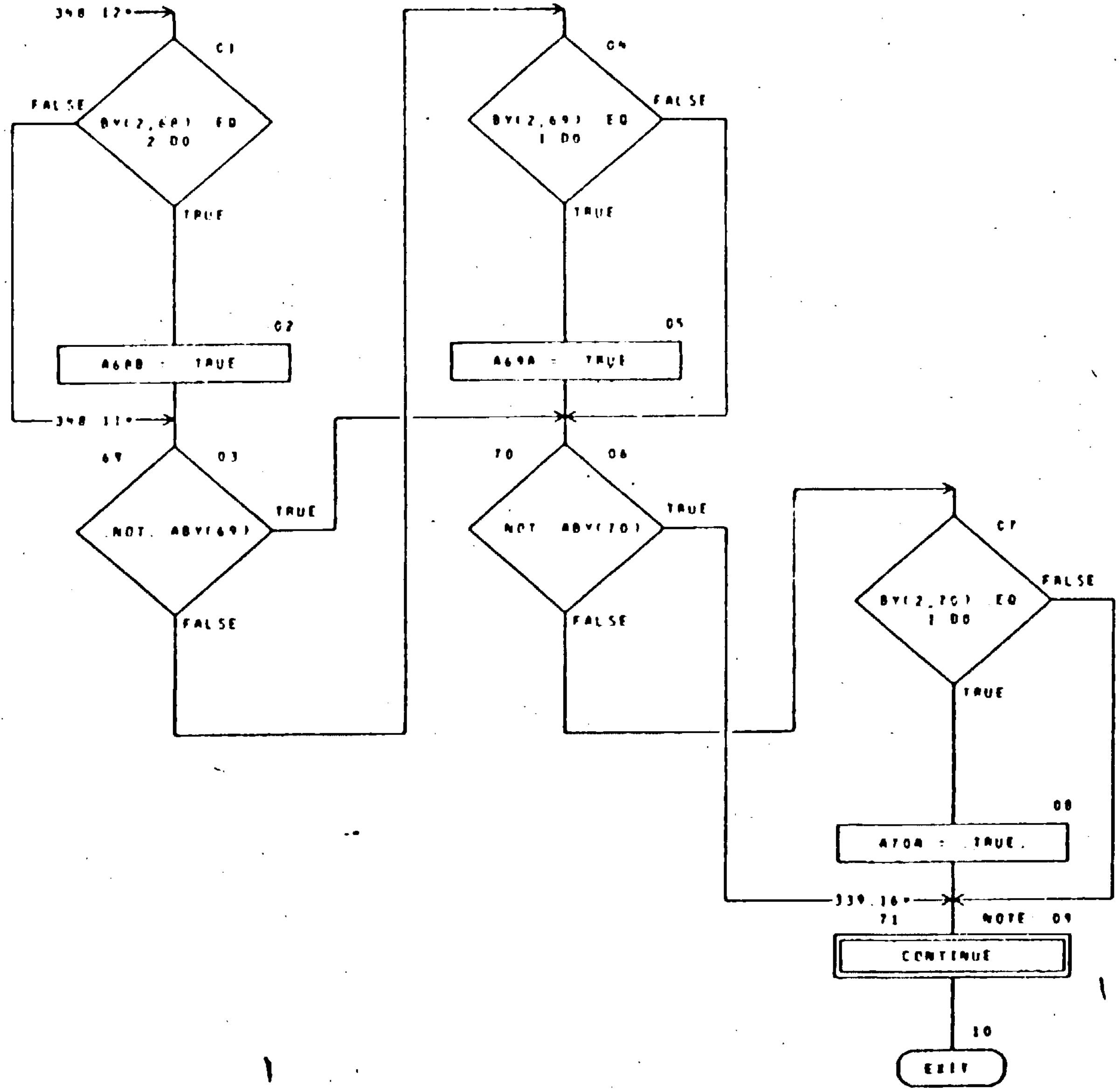


CHART TITLE - NON-PROCEDURAL STATEMENTS


```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL AFALSE,ABY,APHI,A1A,A1B,A1C,A2A,A2B,A2C,A3A,A3C,A4A,A4B,
     A4C,A5A,A5B,A5C,A6A,A6B,A6C,A7A,A7B,A7C,A8A,A8B,A11A,A11B,A11C,
     A12A,A12B,A13A,A13B,A14A,A14B,A14C,A15A,A15B,A16A,A16B,A16C,A17A,
     A17B,A18A,A18B,A19A,A19B,A20A,A20B,A41A,A42A,A43A,A44A,A44B,A45A,
     A45B,A46A,A46B,A47A,A47B,A48A,A48B,BFALSE,A51A,A52A,A53A,A54A,
     A54B,A55A,A55B,A56A,A56B,A57A,A57B,A58A,A58B,A61A,A62A,A63A,A64A,
     A64B,A65A,A65B,A66A,A66B,A67A,A67B,A68A,A68B,A49A,A50A,A59A,A60A,
     A69A,A70A,A30A,A30B,A10A,A10B
      DIMENSION AFALSE(75),BFALSE(39)
      COMMON /INTGR4/ I01(48),LIMPHI,I02(20),LAUNCH,I03(57),LEGMAX,
     I04(872)
      COMMON /LOGIC4/ L01(150)          ,A1A,A1B,A1C,A2A,A2B,A2C,A3A,A3C,
     A4A,A4B,A4C,A5A,A5B,A5C,A6A,A6B,A6C,A7A,A7B,A7C,A8A,A8B,A11A,A11B,
     A11C,A12A,A12B,A13A,A13B,A14A,A14B,A14C,A15A,A15B,A16A,A16B,A16C,
     APHI(2,10),A17A,A17B,A18A,A18B,A19A,A19B,A20A,A20B,A41A,A42A,A43A,
     A44A,A44B,A45A,A45B,A46A,A46B,A47A,L02(70),ABY(70),L03(35),A47B,
     A48A,A48B,A51A,A52A,A53A,A54A,A54B,A55A,A55B,A56A,A56B,A57A,A57B,
     A58A,A58B,A61A,A62A,A63A,A64A,A64B,A65A,A65B,A66A,A66B,A67A,A67B,
     A68A,A68B,A49A,A50A,A59A,A60A,A69A,A70A,A30A,A30B,A10A,A10B,
     L04(61)
      COMMON /ITERAT/ BX(5,70),BY(3,70),BO2(420)
      EQUIVALENCE (AFALSE(1),A1A),(BFALSE(1),A47B)
  100 FORMAT(1H0,53HWARNING. TRIGGERS NO. 17 SHOULD BE ZERO WHEN LAUNCH=
     0)
```

SETUP-23

<u>Name:</u> SMQINT

<u>Calling Argument:</u> /C/, /A/, IN for SMQINT;
SOLUTN for SIMEQ

<u>Referenced Sub-programs:</u> None

<u>Referenced Commons:</u> None

<u>Entry Points:</u> SIMEQ

<u>Referencing Sub-programs:</u> GUESS, MINMX3 for SMQINT and SIMEQ

<u>Discussion:</u> Entry point SIMEQ solves a set of simultaneous linear equations of the form

$$CA = B$$

for the n-dimensional column matrix A where C is a known square n x n matrix and B is a known n-dimensional column matrix. SMQINT performs initialization for SIMEQ.

SIMEQ employs Gauss' method of elimination to solve a set of simultaneous linear equations. Basically, this method involves the solution of one of the n equations for one of the unknown elements of A in terms of the other elements of A, substituting this result into the remaining equations and deriving the new coefficients for the reduced set of equations of order (n - 1). By successively repeating this procedure, a single equation in one unknown is generated. The solution of this equation yields one element of A in terms of the known coefficients of B and C. Using this solution and proceeding backwards successively through the equations of two unknowns, three unknowns, etc., the unique solution vector A is obtained. A detailed description of this method is given in the reference.

Upon successfully solving for the A matrix, the argument SOLUTN is set to .TRUE., and a return to the calling routine is executed. This logical indicator is set to .FALSE. if it is found that all of the equations are not linearly independent.

The coding of SIMEQ assumes that the B array is stored immediately behind the C matrix such that, in effect, C is dimensioned n x (n+1), and B is addressed as the (n+1)th column of C. The values contained in both B and C are destroyed in the computations. Note that the inverse of C is not explicitly formed.

Reference:

Pipes, Louis A. and Shahen A. Hovanessian, Matrix Computer Methods in Engineering, John Wiley & Sons, Inc., New York, 1969, pp. 15, 16.

SMQINT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
A	SUX		Array containing the solution vector A.
C(IN, IN+1)	UX		Array containing the matrix C in the first IN columns and the matrix B in the (IN+1)th column.
IN	UX		Order n of the problem, equal to the number of simultaneous equations.
SOLUTN	SX		Logical flag indicating whether the solution matrix A was successfully obtained. .TRUE. - solution successfully obtained. .FALSE. - solution not obtained because equations are not linearly independent.

CHART TITLE - SUBROUTINE SMQINT(/C/,/A/,IN)

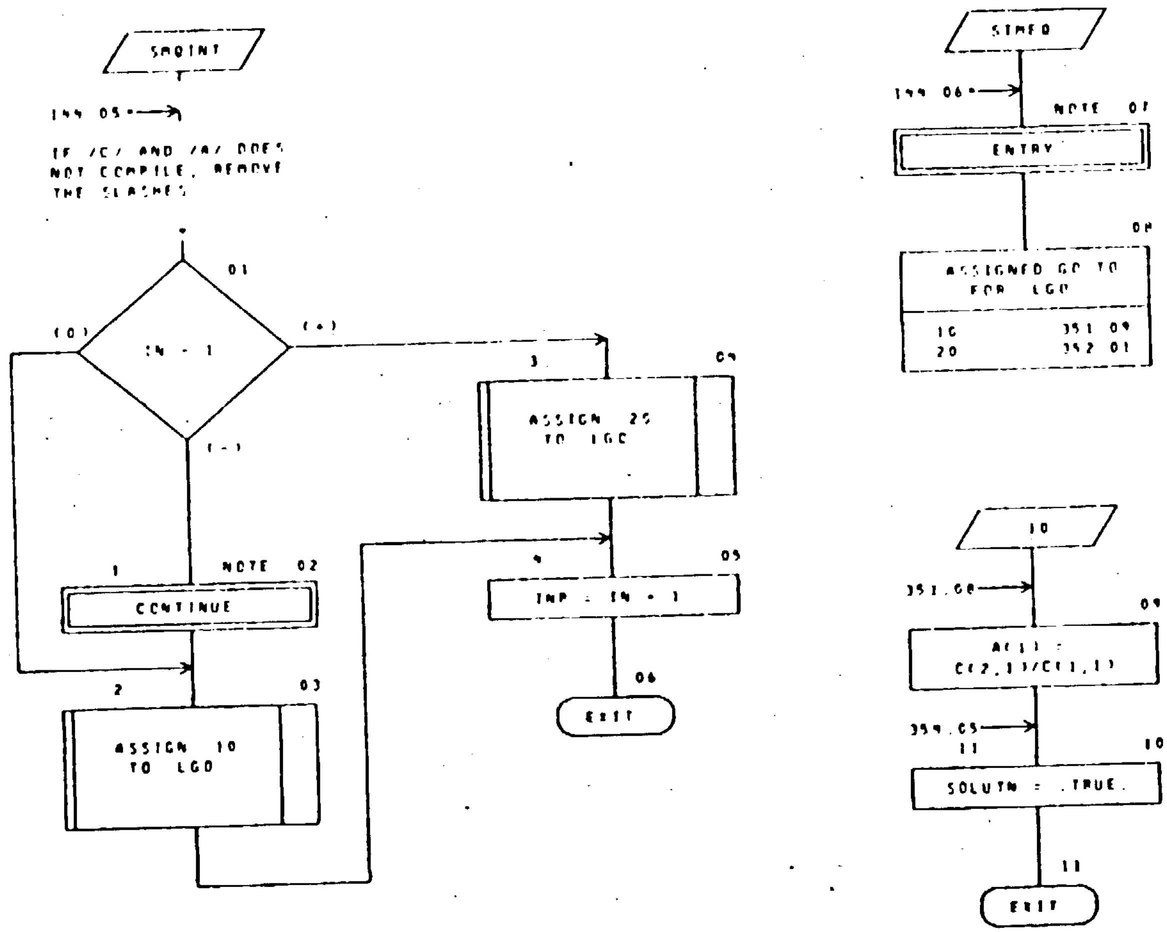


CHART TITLE - SUBROUTINE SMQINT(/C/,/A/,IN)

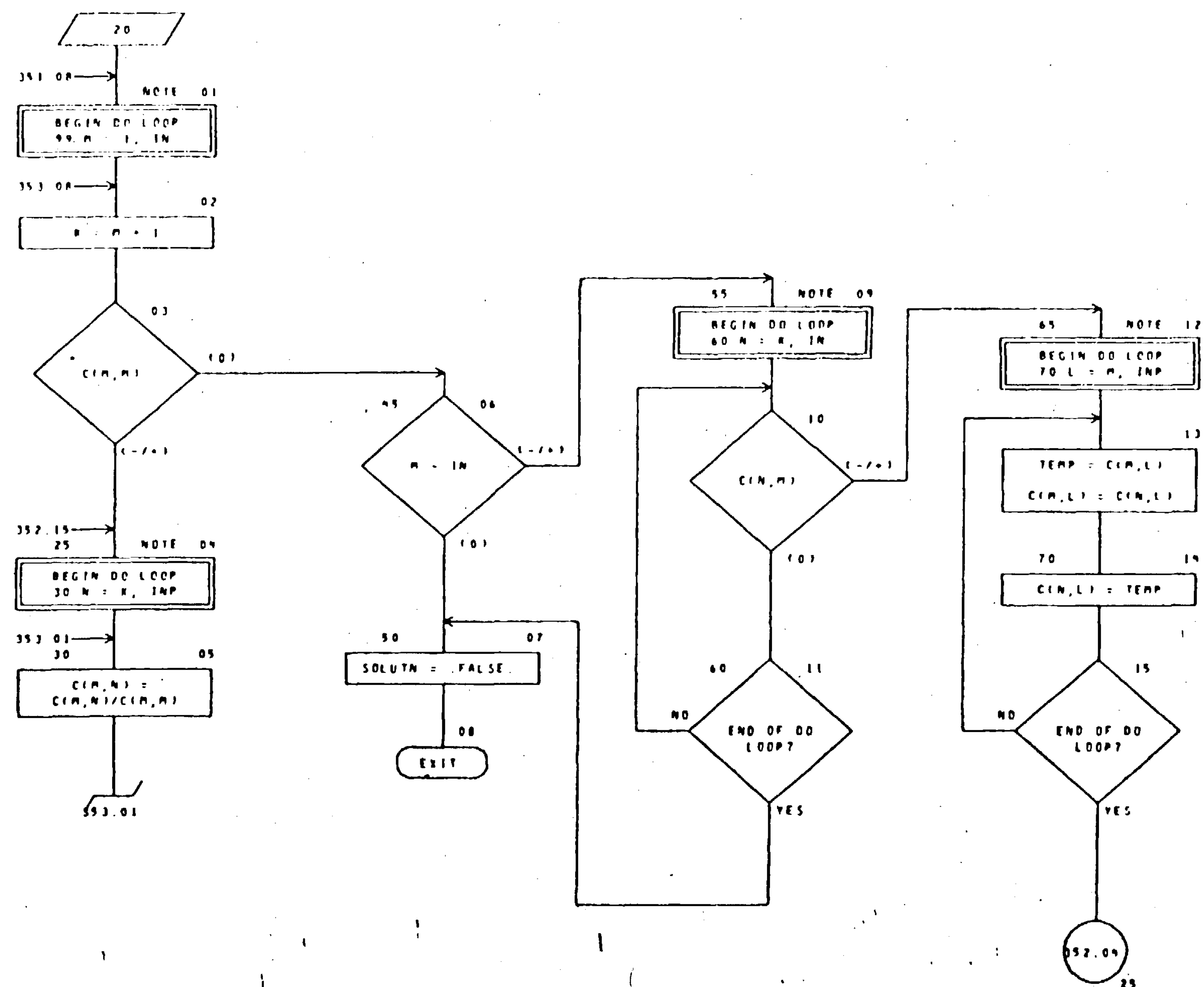
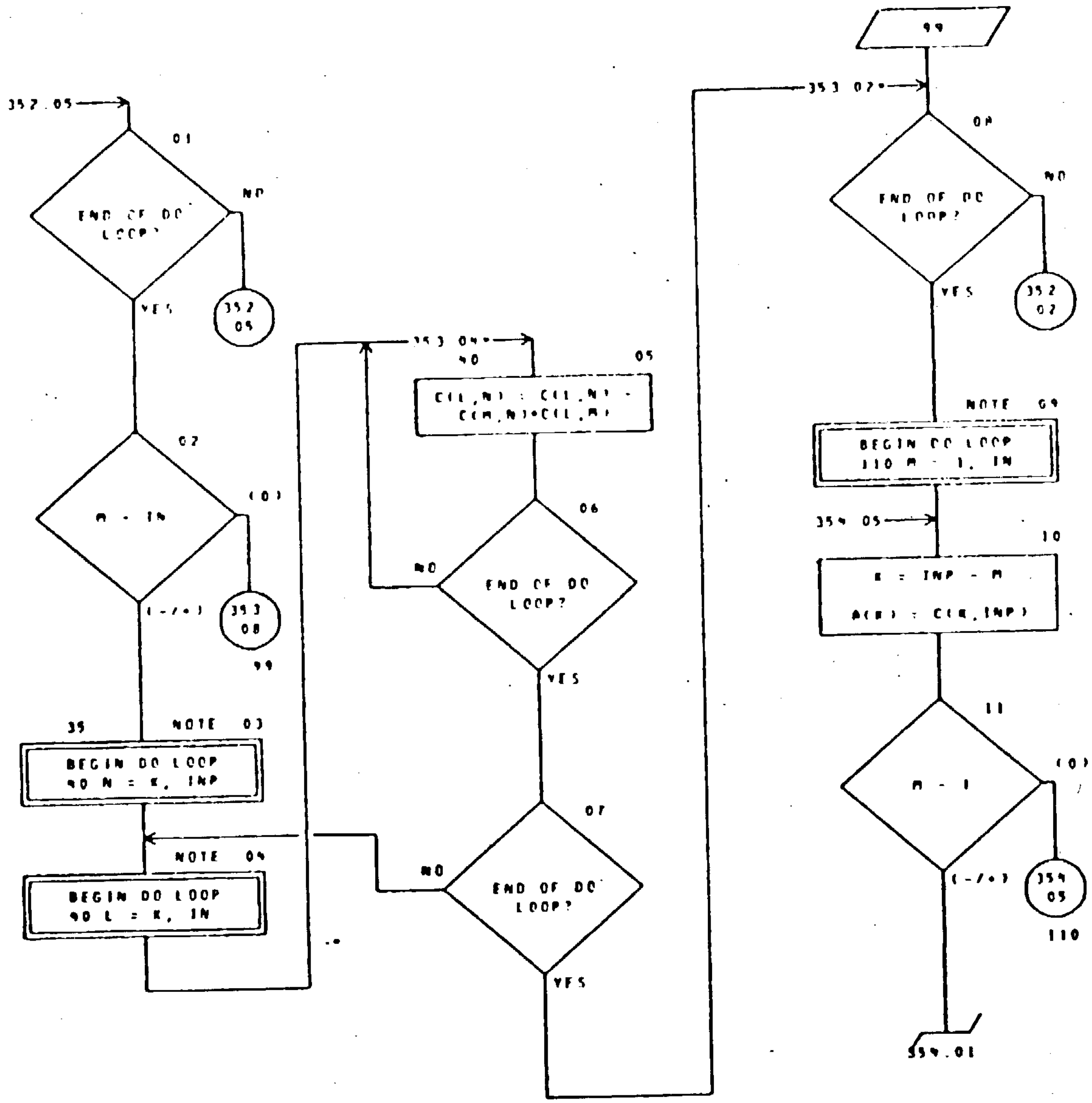


CHART TITLE - SUBROUTINE SMQINT(C/,/A/,IN)



ORIGINAL PAGE IS OF POOR QUALITY

SMQINT-5

CHART TITLE - SUBROUTINE SMQINT(C7,A7,IN)

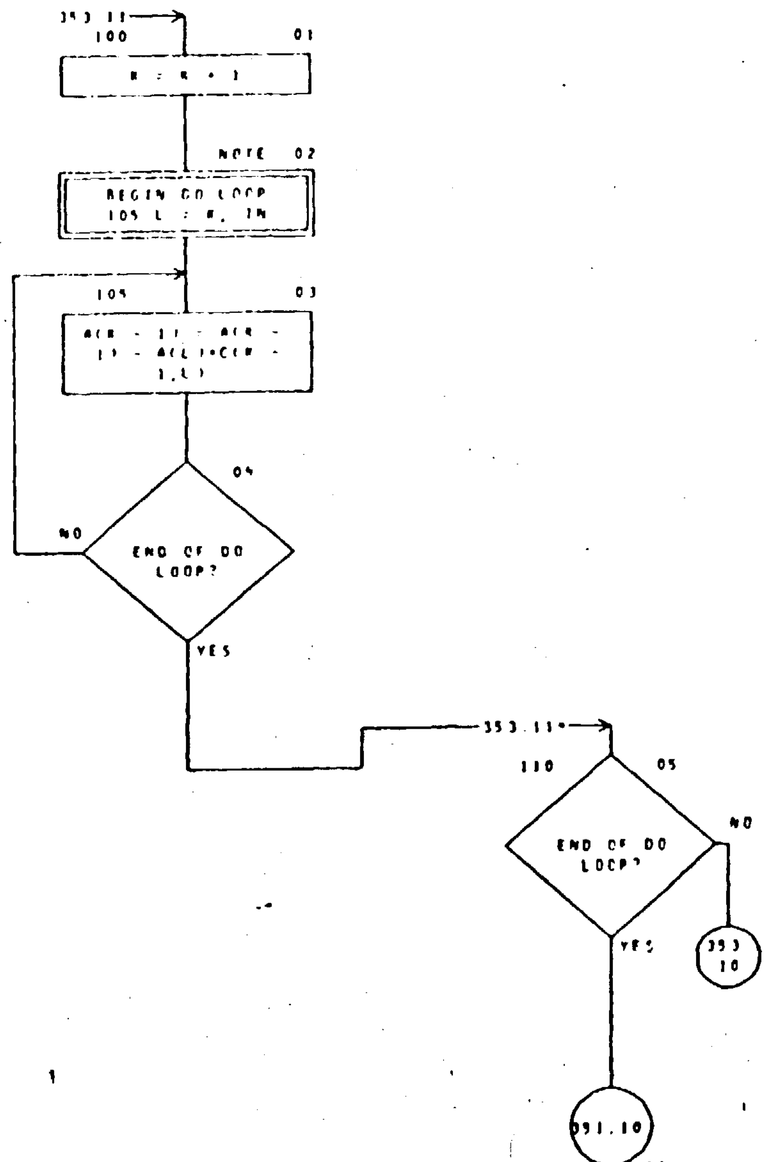


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 ( A-H, O-Z)
LOGICAL SOLUTN
DIMENSION C(IN,IN), A(IN)
```


SMQINT-7

PRECEDING PAGE BLANK NOT FILMED

Name: SOLAR

Calling Argument: None

Referenced Sub-programs: PRINT, PRINTR for SOLAR; None for SOLINT

Referenced Commons: INTGR4, ITERAT, LOGIC4, REAL8

Entry Points: SOLINT

Referencing Sub-programs: CHECK, DERIV, FUNCT, QPRINT, SPRINT, SWSTO, TAP, TRAJI for SOLAR; QSTART for SOLINT

Discussion: For all its formidable source-length, SOLAR is essentially a very simple routine, except that it becomes somewhat involved when power degradation is considered.

SOLAR's basic job is to produce the power function γ and related parameters, such as the density function, d, the damage factor, q, and the partial derivatives $\partial\gamma/\partial r$, $\partial\gamma/\partial d$, and $\partial^2\gamma/\partial d^2$.

SOLINT's basic job is to perform the initialization of certain parameters and logical indicators which are used throughout each trajectory computation; this often involves an iteration to isolate a particular solar distance at which the power function γ has a given property, such as a maximum, zero, or a specified value.

A note to the programmers: a great percentage of execution time is spent in SOLAR, since the power function value γ must be computed at each computation step and at each integration sub-cycle during thrust phases. Therefore, all efforts should be made to minimize the number of operations in SOLAR should modifications be required.

Consider first the case in which power degradation is not considered, i.e., the damage factor q is assumed to be unity at all times and is therefore absent from the formulation. In this case, it is allowable to simulate housekeeping power onboard the spacecraft, and the power factor γ is given by

$$\gamma = p/p_{ref} = (1 + \Delta p) \sum_{i=0}^{4} a_i d^{(i+4)/4} - \Delta p \ ,$$

which is the ratio of instantaneous power to reference power. Δp is a specified value, input to the program, and is defined as the ratio of housekeeping power to reference power,

$$\Delta p = p_h/p_{ref} \ .$$

This expression for γ holds also when power degradation is computed, except that Δp must be zero for the computations to be meaningful; the program automatically checks this and disallows the simultaneous simulation of housekeeping power and power degradation.

In the above expression for γ, the a_i are normalized coefficients which characterize the power-source, and d is the density function, which may simply be a constant in some cases (as with nuclear electric propulsion), or, in the case of solar electric propulsion, is assumed to be of the same form as the density of photons striking the surface of the array,

$$d = \frac{\cos \chi}{r^2} \ ,$$

where r is the spacecraft's solar distance and χ is the angle by which the arrays are tilted away from normal incidence of solar flux.

The following partial derivatives are also computed, when they are required, for utilization either within SOLAR or in other parts of the program:

$$\frac{\partial \gamma}{\partial d} = (1 + \Delta p) \sum_{i=0}^{4} a_i \left(1 + \frac{i}{4}\right) d^{i/4} \ ,$$

$$\frac{\partial^2 \gamma}{\partial d^2} = (1 + \Delta p) \sum_{i=0}^{4} a_i \left(1 + \frac{i}{4}\right)\left(\frac{i}{4}\right) d^{(i-4)/4} \ ,$$

$$\left.\frac{\partial \gamma}{\partial r}\right|_{\chi=0} = \left.\frac{\partial \gamma}{\partial d}\frac{\partial d}{\partial r}\right|_{\chi=0} = -\frac{2}{r^3}\frac{\partial \gamma}{\partial d} .$$

When power degradation is simulated, the damage factor q is computed,

$$q = e^{-s/\tau_d} ,$$

where s is the degradation time and τ_d is the characteristic degradation time. Δp must be zero during degradation simulations, so that the power factor γ may be expressed,

$$\gamma = d \sum_{i=0}^{4} a_i d^{i/4} .$$

When the propulsion mode is solar electric, and power degradation due to solar-flux damage to the solar arrays is simulated, then the possibility may exist of tilting the solar arrays away from the sun to achieve the optimum trade-off between power received and solar-cell damage. Thus the tilt angle χ will be bounded by χ_{curve} on the maximum-power side and $\pi/2$ on the minimum-power side, where the latter bound corresponds to zero power (coasting flight) with the arrays oriented edgewise to the sun. χ_{curve} is the value of χ corresponding to the power curve γ at the current solar distance, and the particular form of the power curve is decided by the program input quantity MODE.

The essential quantity computed is not the tilt angle χ but rather the density, d. The value of this quantity is chosen to satisfy the Maximum Principle of optimal control theory. When the optimum tilt angle lies between the two bounds, $\chi_{curve} < \chi < \pi/2$, the expression for the optimum density is a quartic in the variable $d^{1/4}$, and this expression is solved by Newton's iteration rather than the quartic-formula, since it is felt that Newton's iteration results in fewer computations, on the average, than the rather-involved closed form quartic solution. Nevertheless, the iteration is executed to an extremely tight tolerance, since this iteration occurs at each computation step (and at each integration sub-cycle during thrust phases), and accumulated computational noise must be minimized along the trajectory if the two-point boundary value problem is to converge.

The iteration consists of varying the independent variable d until the root of the function

$$f_{opt} = q(\lambda - \nu\lambda_{\nu}/c) \frac{\partial \gamma}{\partial d} + \lambda_s \nu/g ,$$

is obtained, making use of Newton's method employing the slope,

$$\frac{\partial f_{opt}}{\partial d} = q(\lambda - \nu\lambda_{\nu}/c) \frac{\partial^2 \gamma}{\partial d^2} .$$

A more detailed discussion of the solution for the optimum density d is given in Reference 1 for a specific set of coefficients a_i.

Once the optimum density is computed, two functions are computed for monitoring by subroutine CHECK. These are

$$f_{ch1} = \left.\frac{\partial \gamma}{\partial d}\right|_{\text{actual}} - \left.\frac{\partial \gamma}{\partial d}\right|_{\text{power curve}} ,$$

which is used in isolating the points at which the solar array tilt angle switches to or from the power-curve boundary $\chi \rightleftarrows \chi_{curve}$, and

$$f_{ch2} = \left.\frac{\partial \gamma}{\partial d}\right|_{\text{actual}} - \left.\frac{\partial \gamma}{\partial d}\right|_{\text{absolute max}} ,$$

which is used in isolating points at which the arrays are tilted edgewise to the sun $\chi \rightleftarrows \pi/2$. The roots of these two functions therefore determine when to switch to and from the boundaries corresponding to minimum and maximum χ.

Entry point SOLINT performs the initialization of the logical indicators which are used over and over in the step-by-step computation of each trajectory, and also of the critical values (of solar distance and associated parameters such as d and γ) which are required in the function monitoring by subroutine CHECK. The power law coefficients a_i are initialized; that is, their values are set to either the program's internal default values or to the user's input values. Then the coefficients b_i used in the computation of $(\partial \gamma / \partial r)_{X=0}$ are initialized,

$$b_i = -2\left(1 + \frac{i}{4}\right) a_i \ .$$

There are then four possible iterations, each involving a single independent and dependent parameter, and using Newton's method for finding a function root, which may be executed in order to find the critical point(s) of the power curve.

(1) Iteration to find the solar distance r at which

$$f = \gamma_{X=0} - \gamma_{max} = 0,$$

using the derivative

$$f' = \frac{\partial f}{\partial r} = \frac{\partial \gamma_{X=0}}{\partial r} ,$$

where γ_{max} is input to the program (GAMMAX). There may be more than one solution, but for reasonable coefficients a_i, the iteration will yield the greater solar distance.

(2) Iteration to find the solar distance r at which

$$f = \frac{\partial \gamma_{X=0}}{\partial r} = 0 ,$$

using the derivative

$$f' = \frac{\partial f}{\partial r} = \frac{\partial^2 \gamma_{X=0}}{\partial r^2} .$$

This is the iteration to find $\gamma_{X=0}$ = maximum, which is the peak of the power curve.

(3) Iteration to find the solar distance r at which

$$f = \gamma_{\chi=0} = 0 ,$$

using the derivative

$$f' = \frac{\partial f}{\partial r} = \frac{\partial \gamma_{\chi=0}}{\partial r} .$$

The solution to this iteration corresponds to that point on the power curve at which the solar arrays are heated due to solar proximity such that their power output vanishes. Due to the steep slope of the power curve in this neighborhood, an accurate initial guess of r for the iteration is obtained by sweeping out a grid of points in the neighborhood of $\gamma = 0$ until a value of r is found which is within approximately .02 AU of the solution.

(4) Iteration to find the density d for which

$$f = \frac{\partial^2 \gamma}{\partial d^2} = 0 ,$$

using the derivative

$$f' = \frac{\partial f}{\partial d} = \frac{\partial^3 \gamma}{\partial d^3} .$$

This applies to trajectories involving power degradation, and therefore $\Delta p = 0$ (no housekeeping power). The root of the function f is designed to yield the absolute maximum value of $\partial \gamma / \partial d$, which is used to compute $f_{ch\,2}$ (see above). The solution for d is then the minimum value. The third-partial in f' is given by

$$\left. \frac{\partial^3 \gamma}{\partial d^3} \right|_{\Delta p = 0} = \sum_{i=0}^{4} a_i \left(1 + \frac{i}{4}\right)\left(\frac{i}{4}\right)\left(\frac{i}{4} - 1\right) d^{(i-8)/4}$$

See Reference 1 for more insight into power degradation considerations.

Messages and printouts: When the iteration for the optimum density, in subroutine SOLAR, fails to converge, the following diagnostic message is output on unit 6:

ITERATION FOR POWER DENSITY FAILED. DIT, F, FP, BSOL, COEF, Q =

________ ________ ________ ________ ________ ________

in which DIT is the value of density, d (all values corresponding to the failure point), F is the value of the function f_{opt} (see Discussion), FP is $\partial f_{opt}/\partial d$, BSOL is $\nu\lambda_{\nu}/c$, COEF is $q(\lambda - \nu\lambda_{\nu}/c)$, and Q is $\lambda_s \nu/g$. For the first, and only the first, time this iteration fails during a given computer run, the following occurs:

(1) The message is printed on unit 12:

POWER DENSITY ITERATION FAILED

(2) The heading is printed on unit 11:

FAIL X(I)

followed immediately by all of the iterator independent-variable and dependent-variable values (via a call to subroutine PRINT). The independent-variable values may be used to regenerate the fail-trajectory; the dependent-variable values are meaningless since they correspond to the priorly-computed trajectory (if there was one).

(3) The same as in (2) is printed on unit 6, using a call to subroutine PRINTR.

All of the remaining output comes from entry point SOLINT and consists of either informative messages after successful iterations or diagnostic messages after iteration failures, all output on unit 6, except for the following termination message which is output on units 6 and 12 when the user attempts to simulate both housekeeping power and power degradation:

COMBINATION OF DEGRADATION AND HOUSEKEEPING POWER OPTIONS NOT ALLOWED

The run is then terminated.

If either Iteration (1) or Iteration (3) (see Discussion) fails, the message is printed:

SOLAR POWER LAW ITERATION FAILURE

AT RADIUS = $\underline{\quad (r) \quad}$ AU, POWER = $\underline{\quad (\gamma) \quad}$

where r is the solar distance in AU at the failure point and γ is the corresponding power function value.

When either Iteration (1) or Iteration (2) converges, the informative message is printed:

MAXIMUM POWER = $\underline{\quad (\gamma) \quad}$ AT RADIUS = $\underline{\quad (r) \quad}$ AU.

When Iteration (3) converges, the informative message is printed:

ZERO POWER AT RADIUS = $\underline{\quad (r) \quad}$ AU.

When Iteration (4) converges, the informative message is printed:

MINIMUM DENSITY = $\underline{\quad (d_{min}) \quad}$ MAXIMUM DPOWD = $\underline{\quad ((\partial\gamma/\partial d)_{max}) \quad}$

where the printed quantities are described in the Discussion.

When Iteration (2) fails, the message is printed:

SOLAR POWER LAW ITERATION FAILURE

AT RADIUS = $\underline{\quad (r) \quad}$ AU, POWER DERIVATIVE W.R.T.RADIUS = $\underline{(\partial\gamma/\partial r)}$

When Iteration (4) fails, the message is printed:

SOLAR POWER LAW ITERATION FAILURE, D = $\underline{(d)}$ DPOWDD = $\underline{(\partial^2\gamma/\partial d^2)}$

If the error in any of the failed iterations is still relatively small, the program will accept the results of the iteration (after printing the corresponding diagnostic message) and continue execution. Otherwise, before setting the program master error-indicator and exiting, an additional line is printed:

UNACCEPTABLE. CASE SKIPPED.

and also the message is printed on unit 12:

SOLAR POWER LAW ITERATION FAILURE. CASE SKIPPED.

Reference:

1. J.L. Horsewood, F.I. Mann and K.B. Brice, "The Generation and Interpretation of Electric Propulsion Mission Analysis Data," AMA, Inc. Report No. 73-38, August 1973.

SOLAR EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
D	SUE		The density, d, in AU^{-2}.
O(70)	U	ITERAT	Array of iterator independent-variables, in internal units.
X(50)	U	REAL8	Array of trajectory dependent-variables (see subroutine RKSTEP).
FT	U	REAL8	Reference thrust acceleration, g, in AU/tau^2.
OO(70)	S	ITERAT	Array of iterator independent-variables, in external units.
RT	U(S)	REAL8	Spacecraft solar distance, r, in AU; the most fundamental input to SOLAR; used as iteration variable in SOLINT.
VJ	U	REAL8	Jet exhaust speed, c, in EMOS.
A1S	S	REAL8	Leading power law coefficient, a_o.
PMN	U	REAL8	Primer magnitude, λ.

SOLAR EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ASOL(5)	U	REAL8	Program-input power law coefficients, a_i - input (normalized).
CONX(70)	U	ITERAT	Array of iterator independent-variable conversion factors.
DMAX	SU	REAL8	Maximum value of density, d_{max}, in AU^{-2}.
DMIN	SU	REAL8	Minimum value of density, d_{min}, in AU^{-2}.
DPOW	U	REAL8	Housekeeping-power ratio, Δp, input to the program.
EDGE	U	LOGIC4	Indicator for solar arrays being oriented edgewise to the sun; used only if power degradation is simulated.
FLAP	SU	LOGIC4	Indicator for power-curve options corresponding to nuclear-electric propulsion or solar-electric propulsion using reflecting flaps to maintain maximum power.
HEAT	SU	LOGIC4	Indicator that the solar panels are maintained normal to the sun line at all times.
LINE	SU	INTGR4	Counter; equals the number of lines which have been output on unit 11 during the current computer run.
MODE	U	INTGR4	Power variation option selector.
POWR	SU	REAL8	Power ratio, $q\gamma$.
QJEX	U	LOGIC4	Detailed printout indicator.
SPIN	S	LOGIC4	Spinner indicator (not used at present).

SOLAR EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
TILT	SU	LOGIC4	Indicator that arrays are to be tilted during solar proximity (to maintain constant power when there is no degradation).
CHFNC	S	REAL8	A function whose roots determine the switch points of the array tilt angle to and from the power-curve boundary, f_{ch1}.
CONTM	U	REAL8	Time conversion factor, tau to days.
DPOWD	SU	REAL8	$q\,\partial\gamma/\partial d$.
DPOWR	SU	REAL8	$q\,\partial\gamma/\partial r$.
ERODE	SU	LOGIC4	Power degradation option indicator.
ERROR	S	LOGIC4	Program master error indicator.
HOUSE	SU	LOGIC4	Housekeeping power option indicator.
ISPIN	U	INTGR4	Spinner indicator (not used at present).
PCURV	U	LOGIC4	Indicator for condition in which solar arrays are oriented to receive the maximum power permissible under the current power-curve assumption, or to be tilted away from the maximum permissible due to degradation considerations.
PMAX0	SU	REAL8	Maximum value of power function, γ_{max}.
RPOW0	SU	REAL8	Solar distance at which the power function γ passes through zero, $r_{\gamma=0}$, in AU.
CHFNC2	S	REAL8	A function whose roots determine the switch points of the array tilt angle to and from the stowed-edgewise boundary ($\chi = \pi/2$), $f_{ch\,2}$.

SOLAR-11

SOLAR EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
DEGRAD	SU	REAL8	Damage factor, q.
DPDMAX	SU	REAL8	Absolute maximum of $\partial \gamma / \partial d$, used in computation of f_{ch2}.
DPOWDD	SU	REAL8	$q \partial^2 \gamma / \partial d^2$.
GAMMAX	U	REAL8	Maximum value of power function, γ_{max}, input to the program.
MAXPOW	SU	LOGIC4	Indicator for mode of operation in which solar panels are maintained in orientation to receive maximum permissible power, when degradation option is invoked.
REGION	U	LOGIC4	Indicator for spacecraft solar proximity; demarks two possible regions in space, separated by sphere about sun of specified radius, at which power function (or its derivative) has a corner.
RPMAX0	SU	REAL8	Solar distance at which power function γ peaks, r_{peak}, in AU.
TAUPOW	SU	REAL8	Negative inverse of characteristic degradation time, $-1/\tau_d$, in tau^{-1}.
TPOWER	U	REAL8	Solar-cell degradation characteristic-time, τ_d, in days.

CHART TITLE - SUBROUTINE SOLAR

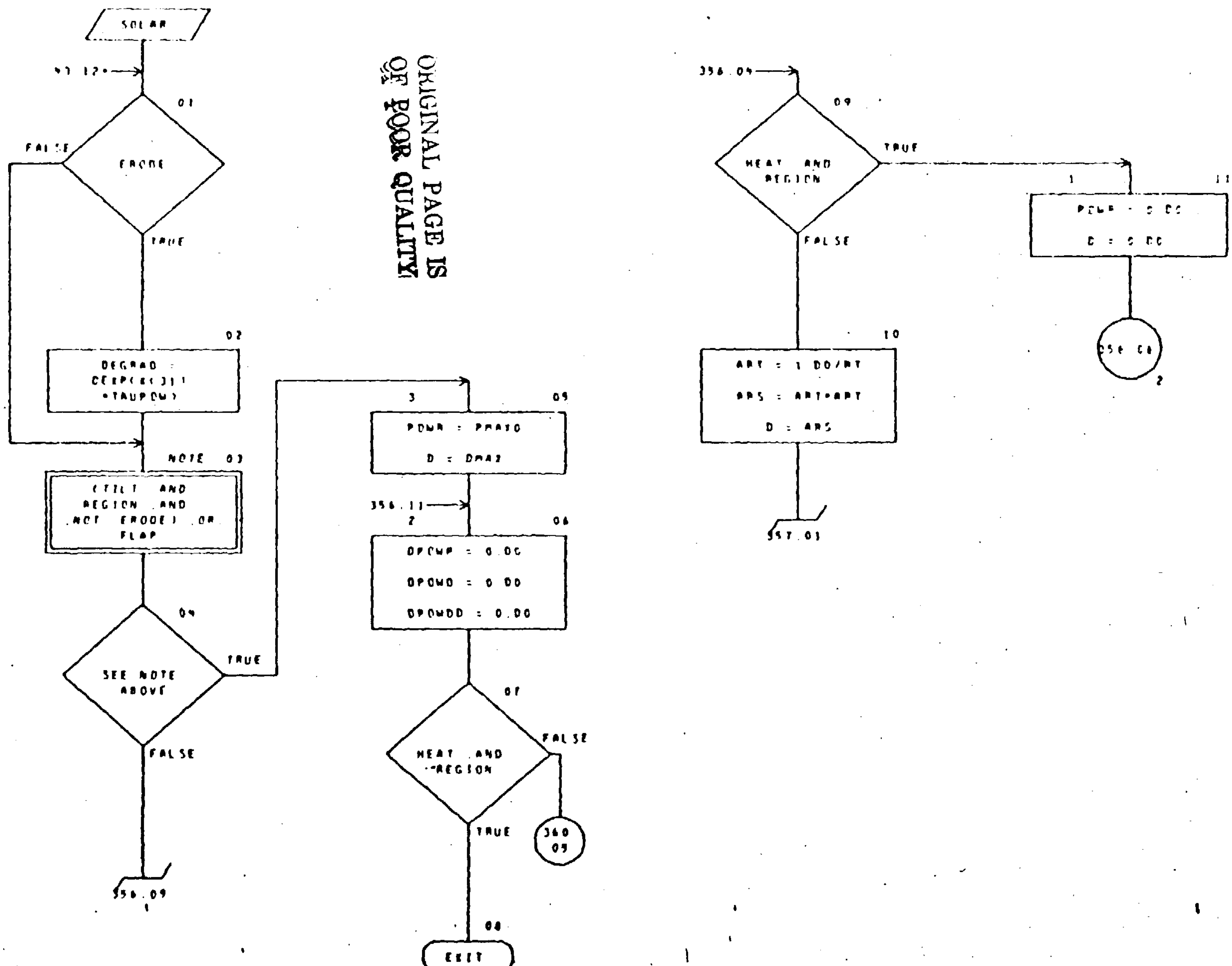


CHART TITLE - SUBROUTINE SOLAR

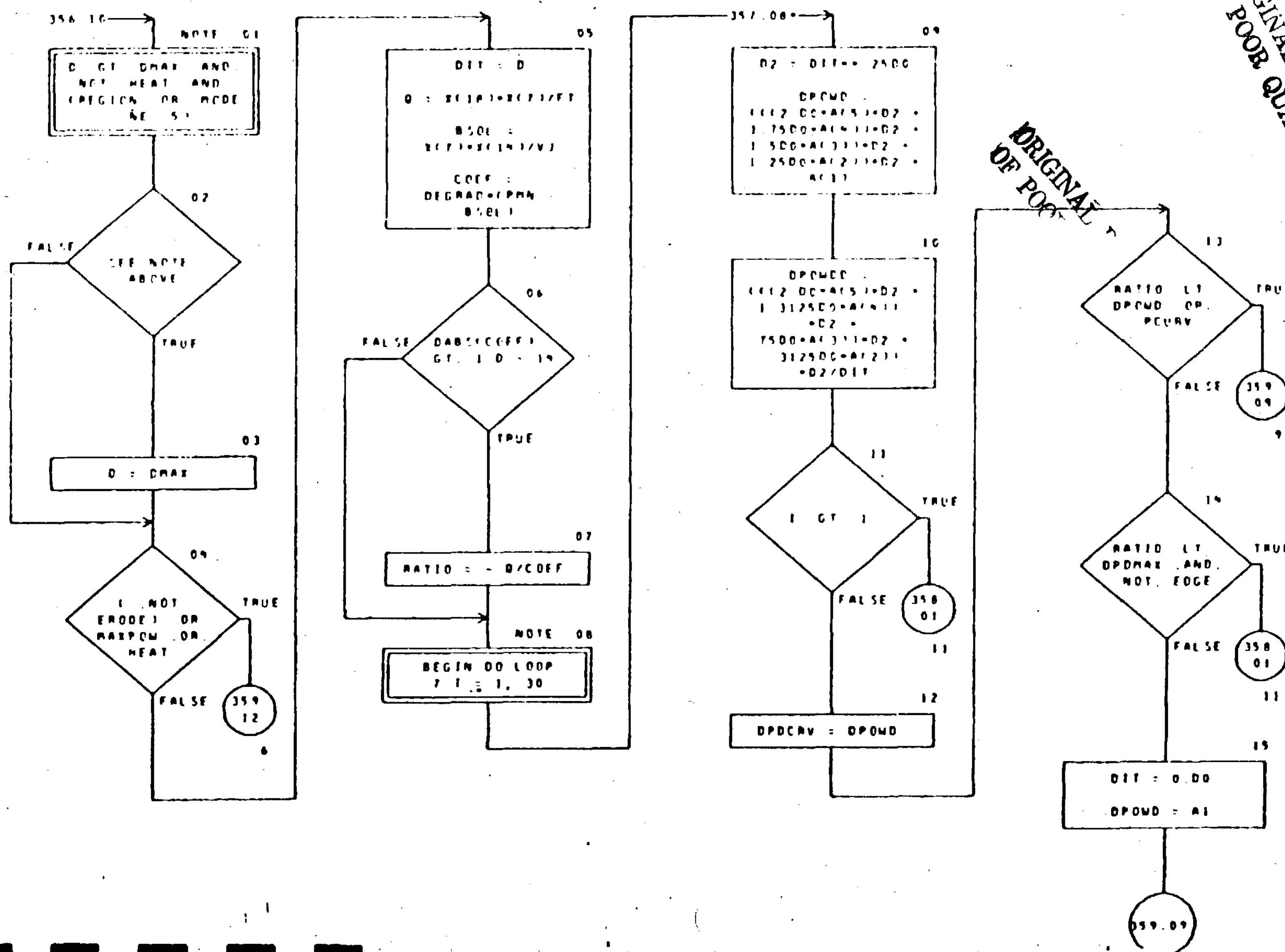


CHART TITLE - SUBROUTINE SOLAR

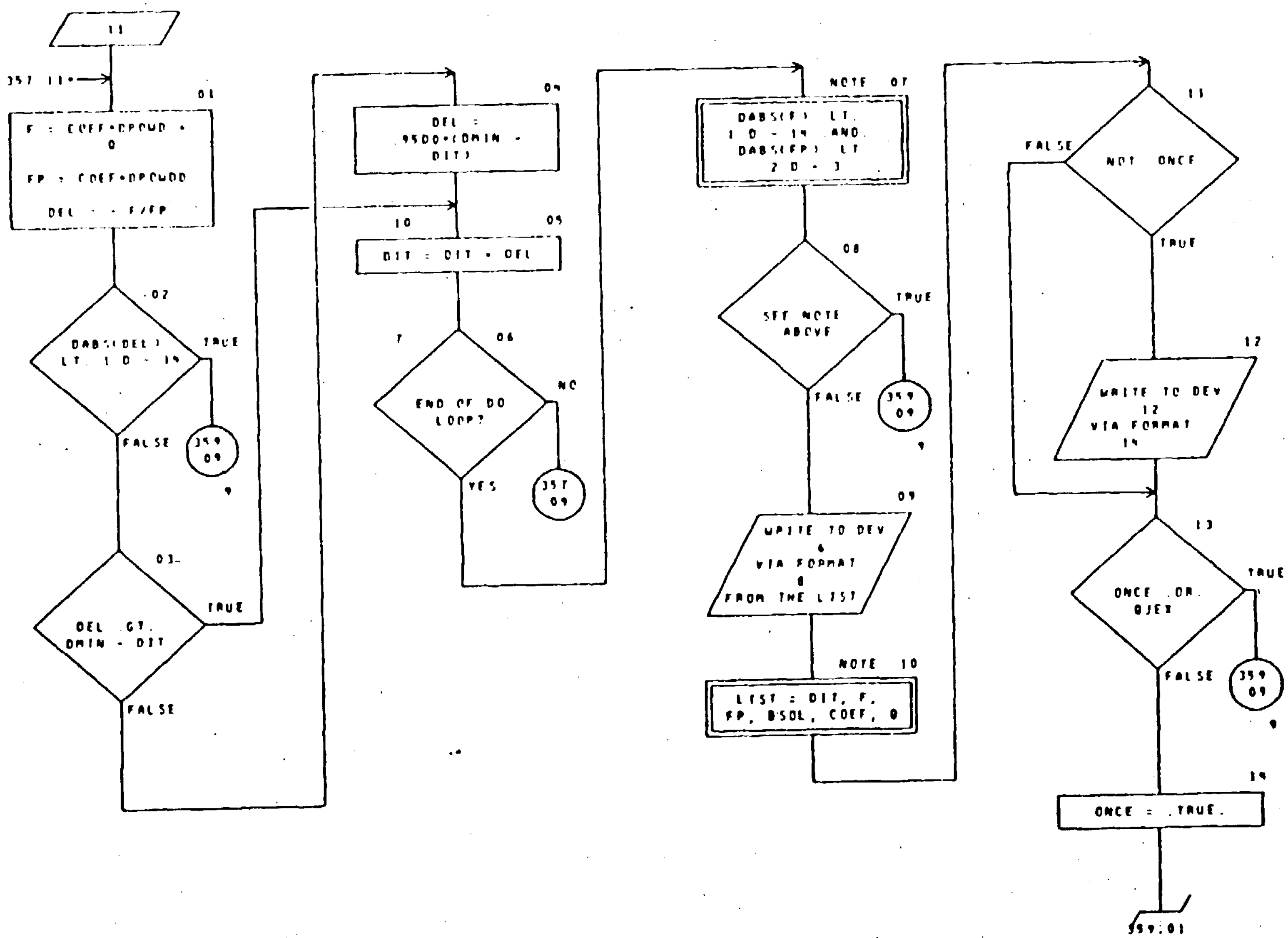


CHART TITLE - SUBROUTINE SOLAR

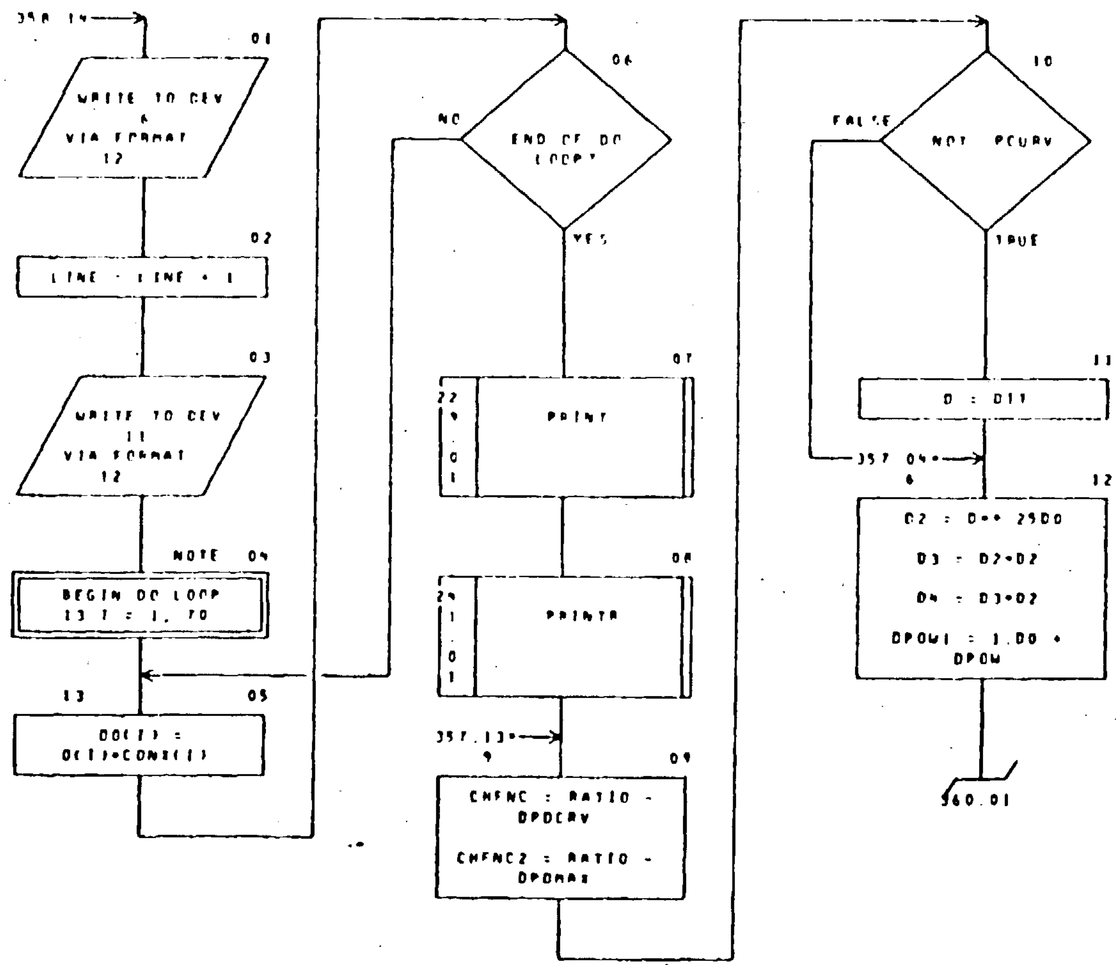
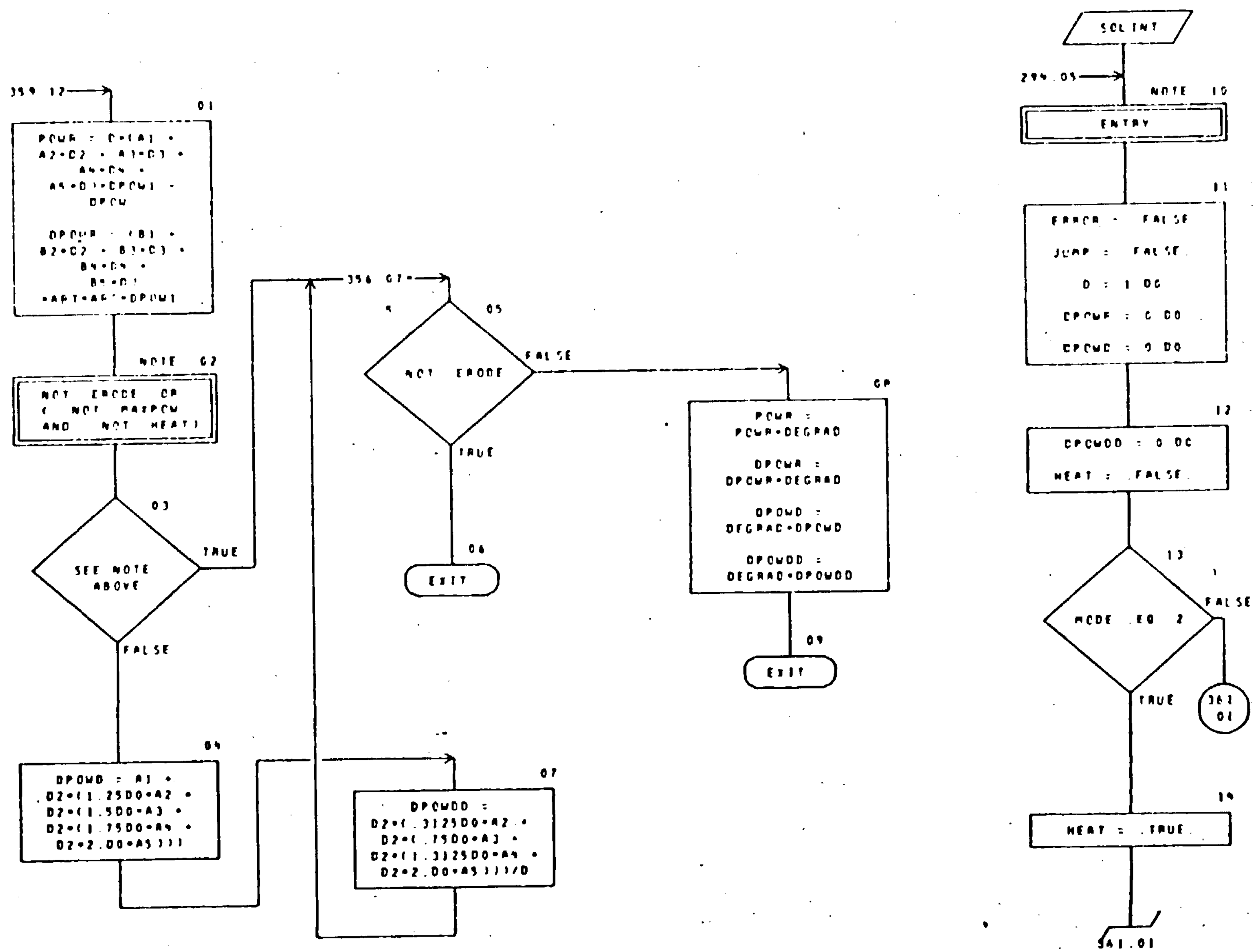


CHART TITLE - SUBROUTINE SOLAR



SOLAR-17

CHART TITLE - SUBROUTINE SOLAR

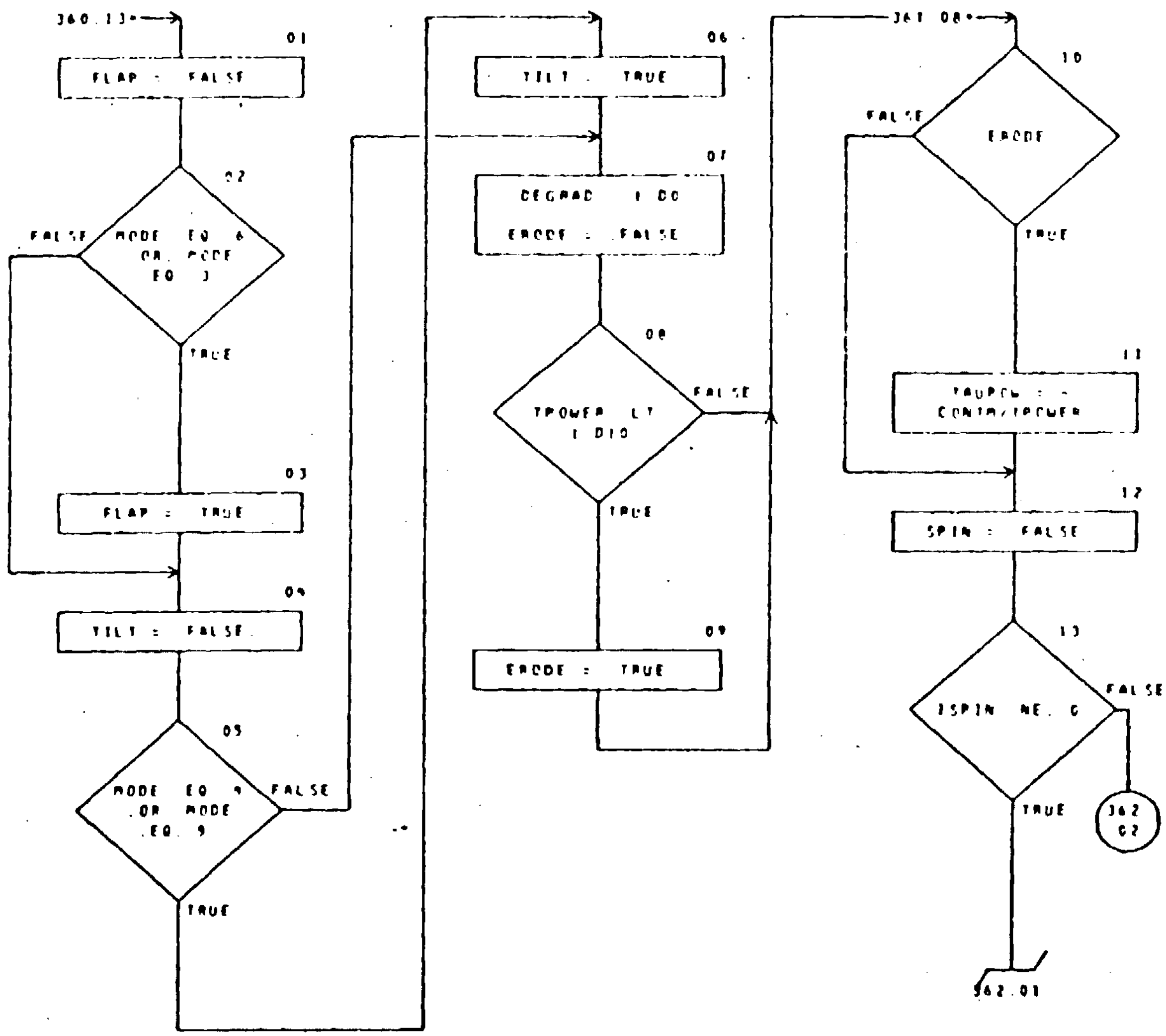
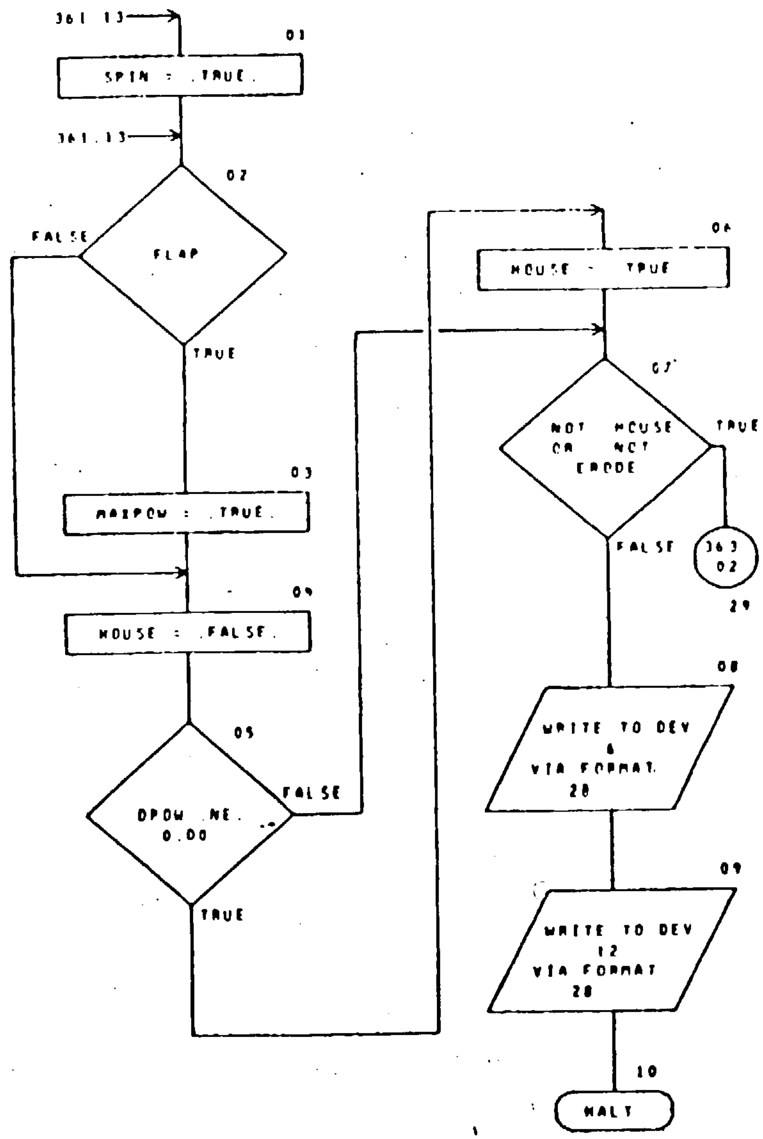


CHART TITLE - SUBROUTINE SOLAR



SOLAR-19

CHART TITLE - SUBROUTINE SOLAR

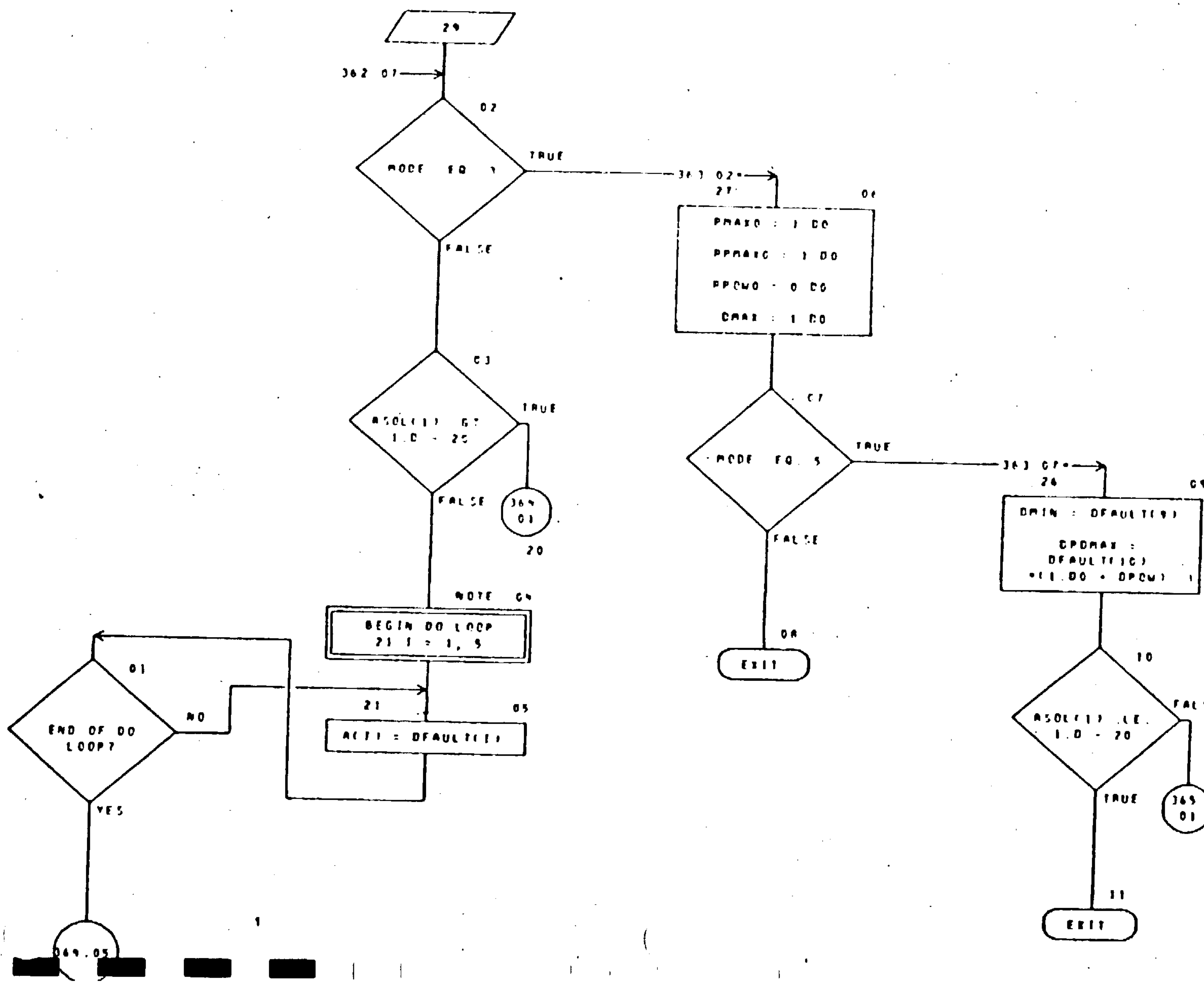


CHART TITLE - SUBROUTINE SOLAR

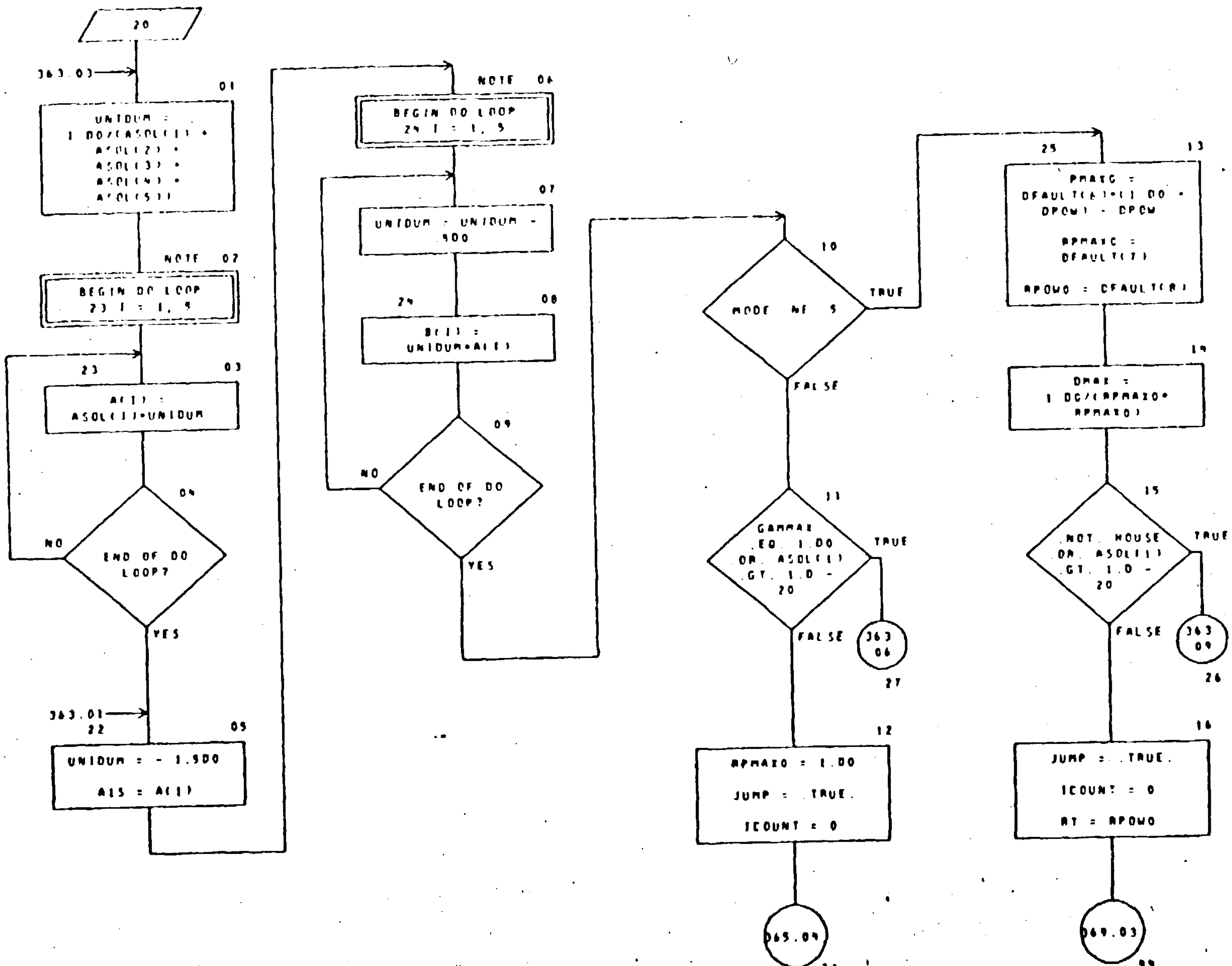


CHART TITLE - SUBROUTINE SOLAR

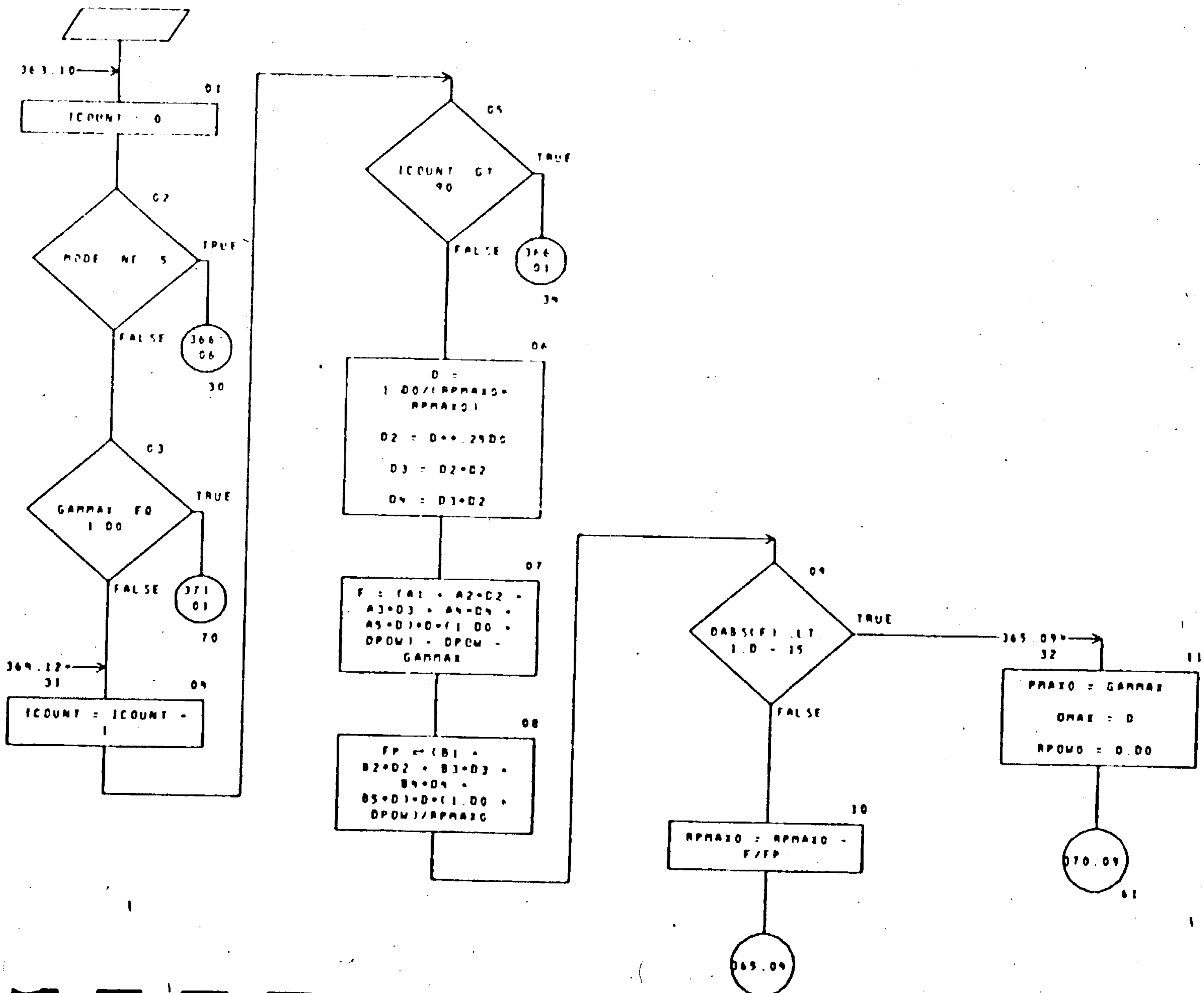
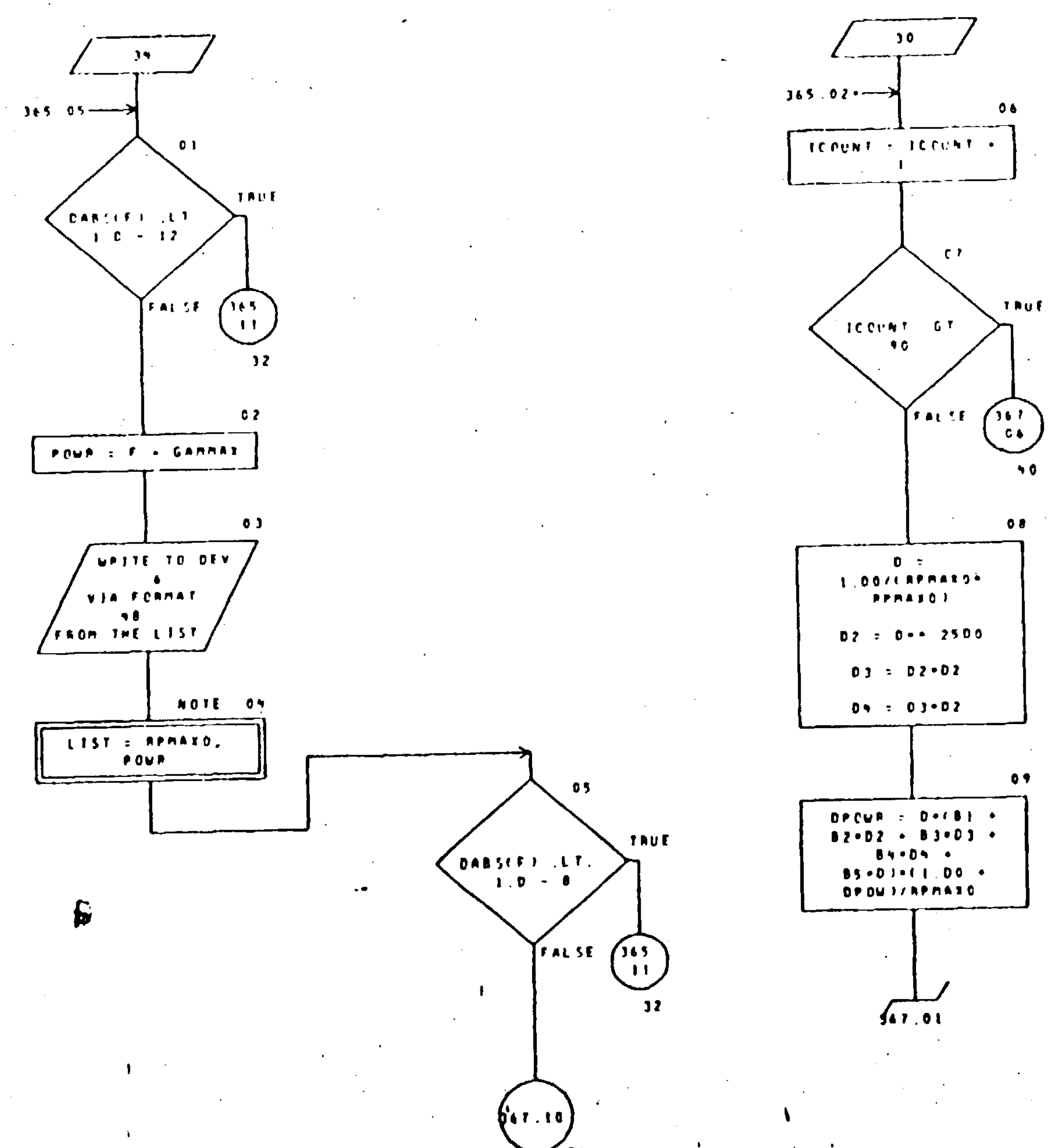


CHART TITLE - SUBROUTINE SOLAR



SOLAR-23

CHART TITLE - SUBROUTINE SOLAR

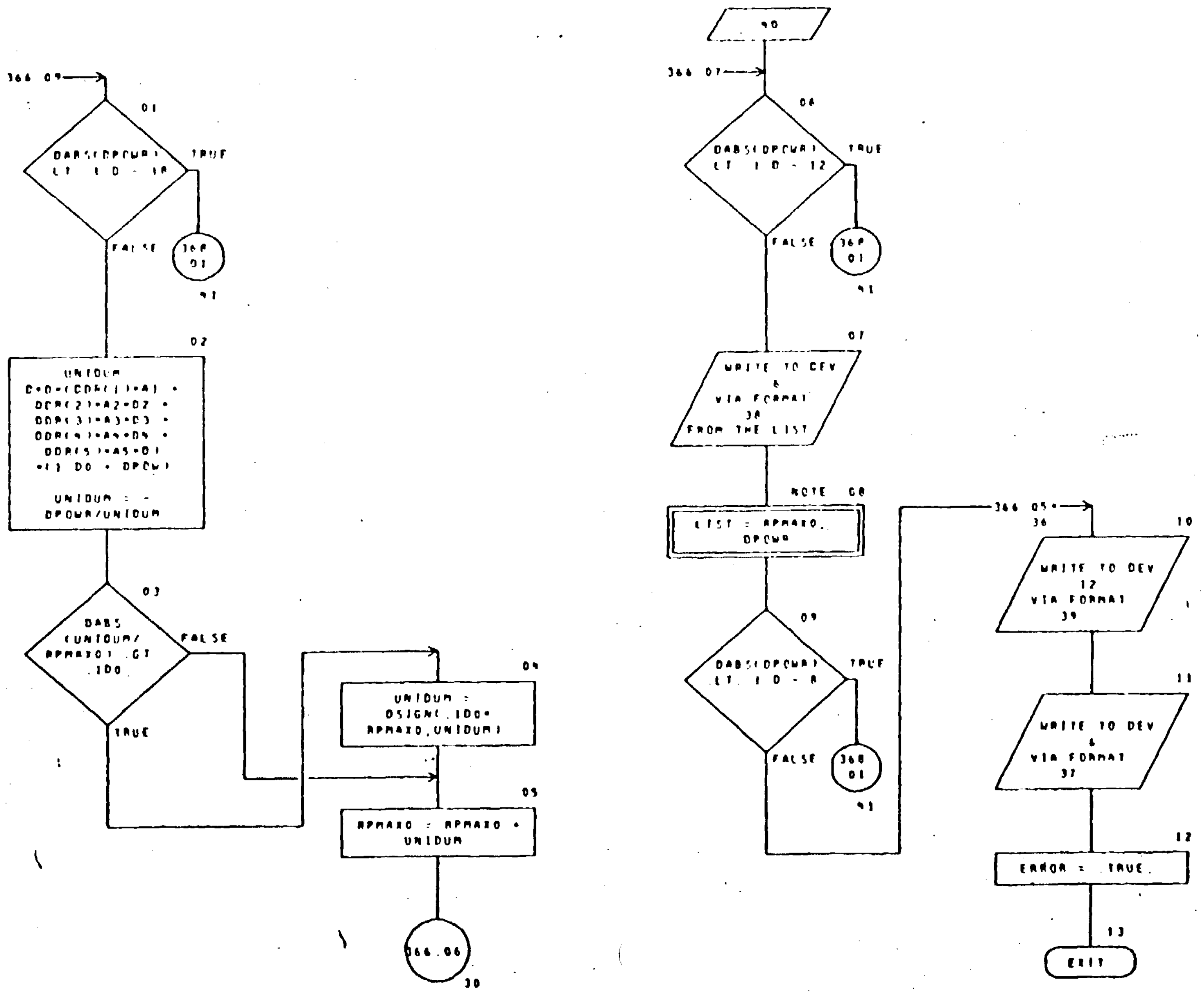


CHART TITLE - SUBROUTINE SOLAR

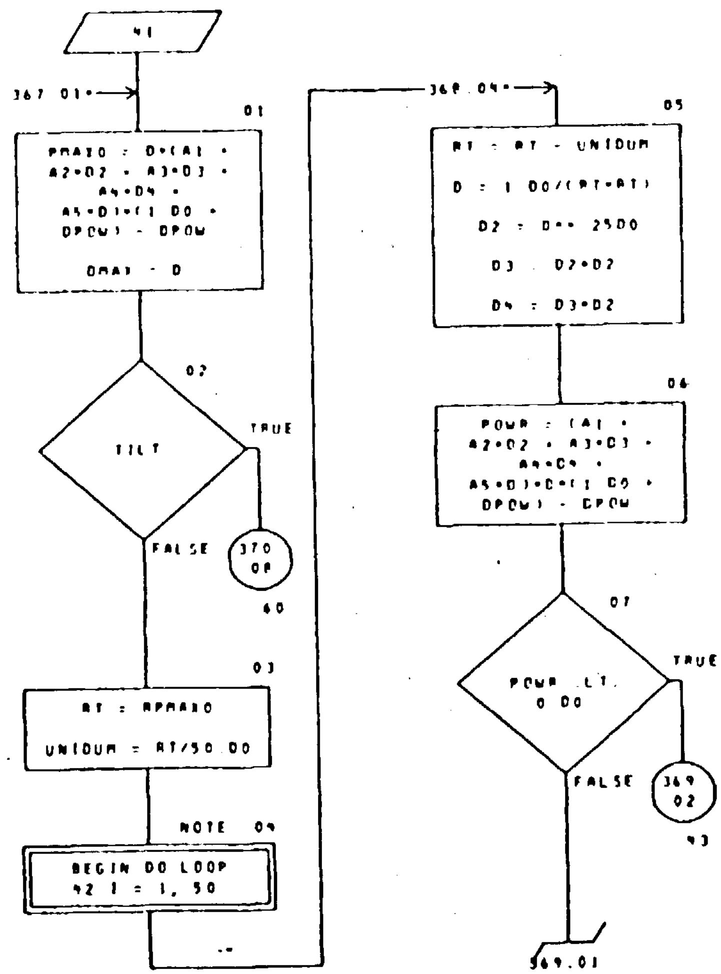


CHART TITLE - SUBROUTINE SOLAR

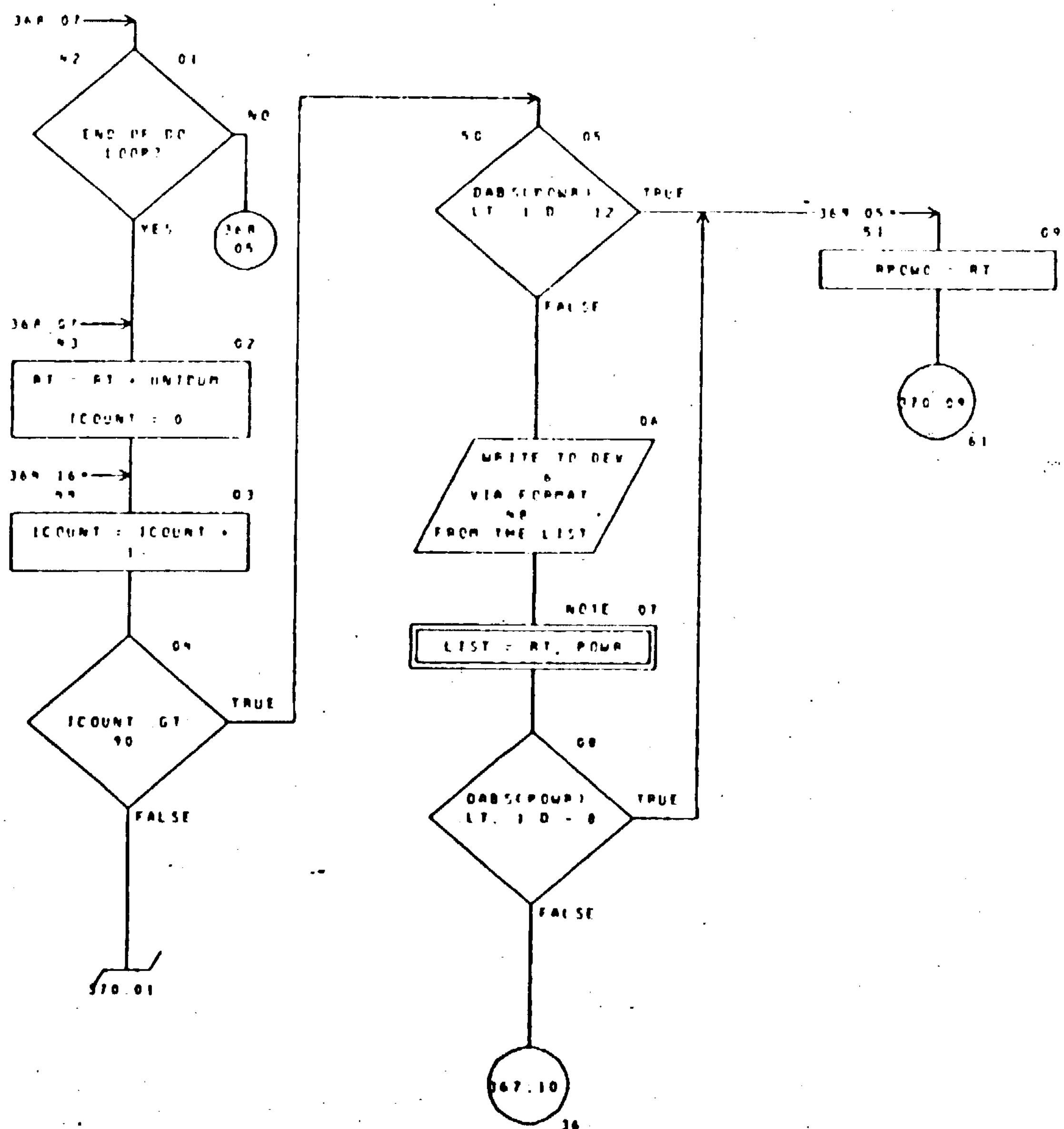
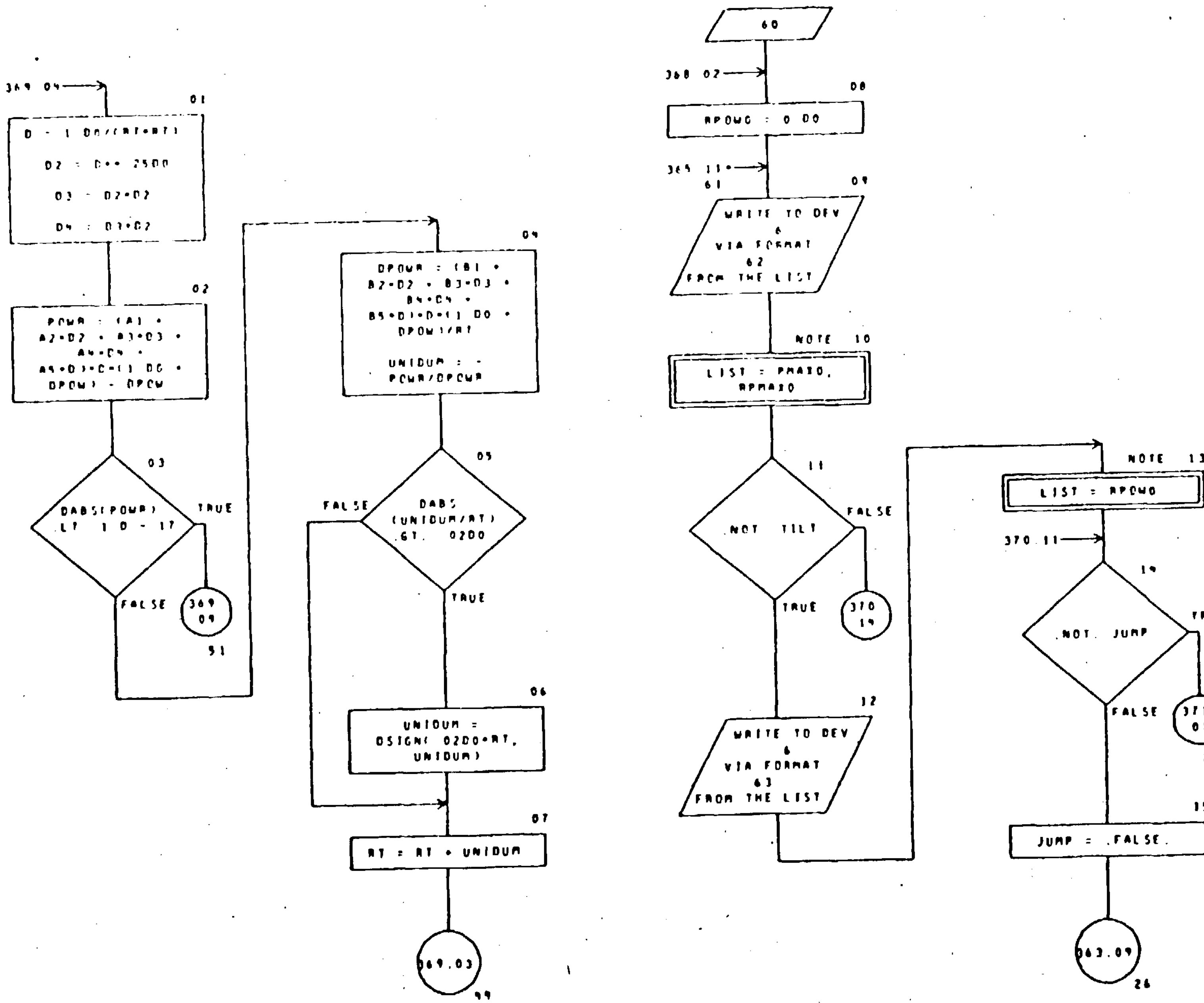
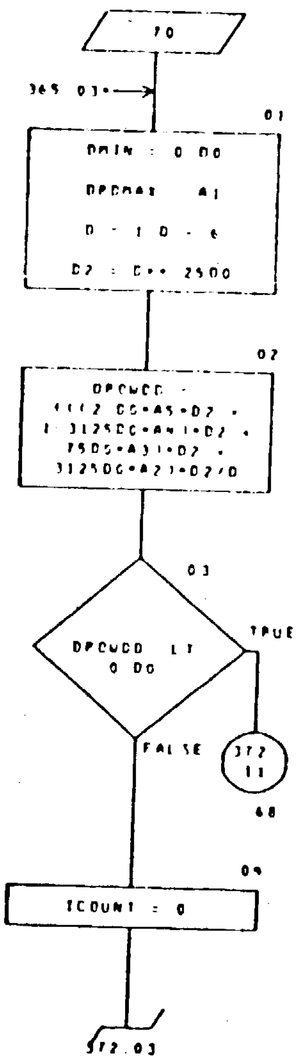
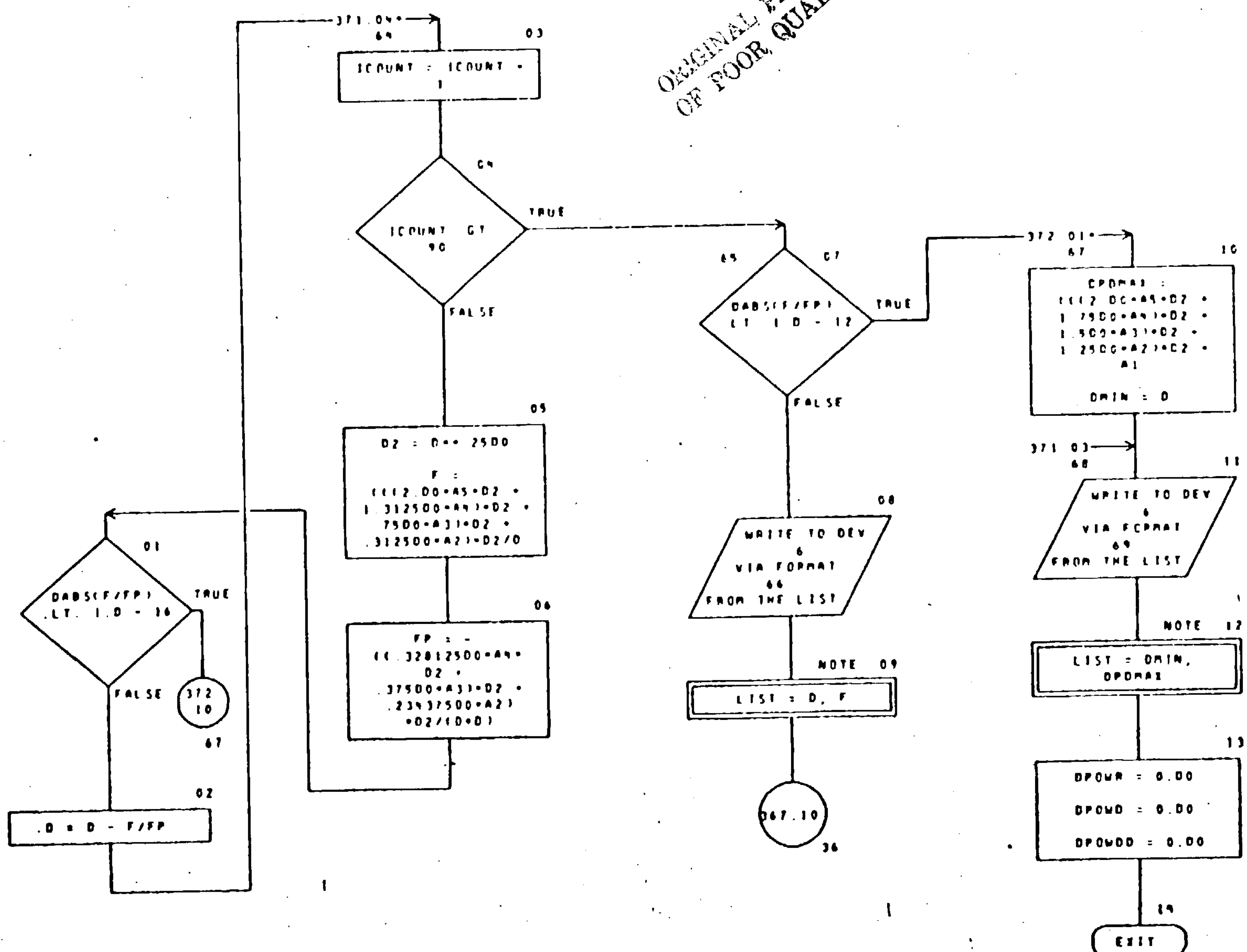


CHART TITLE - SUBROUTINE SOLAR



SOLAR-27


70
365 03*
01
DMIN = 0.D0
DPCMAX = A1
D = 1.D-6
D2 = D**.2500
02
DPCWDD =
(((2.DO*A5*D2 +
1.3125DO*A4)*D2 +
.75DO*A3)*D2 +
.3125DO*A2)*D2/D
03
DPCWDD LT
0.D0
TRUE
372
11
68
FALSE
04
ICOUNT = 0
372.03



371.04* 64
03
ICOUNT = ICOUNT + 1
04
ICOUNT .GT. 90
TRUE
FALSE
05
D2 = D** 2500
F = (((2.D0*A5*D2 + 1.312500*A4)*D2 + .7500*A3)*D2 + .312500*A2)*D2/D
06
FP = - ((.32812500*A4* D2 + .37500*A3)*D2 + .23437500*A2) *D2/(D*D)
01
DABS(F/FP) .LT. 1.D - 16
TRUE
FALSE
372 10
67
02
.D = D - F/FP
65
07
DABS(F/FP) .LT. 1.D - 12
TRUE
FALSE
08
WRITE TO DEV 6 VIA FORMAT 66 FROM THE LIST
NOTE 09
LIST = D, F
067.10
36
372 01* 67
10
DPDMAX = (((2.D0*A5*D2 + 1.7500*A4)*D2 + 1.500*A3)*D2 + 1.2500*A2)*D2 + A1
DMIN = D
371 03 68
11
WRITE TO DEV 6 VIA FORMAT 69 FROM THE LIST
NOTE 12
LIST = DMIN, DPDMAX
13
DPOWR = 0.00
DPOWD = 0.00
DPOWDD = 0.00
14
EXIT

SOLAR-29

CHART TITLE - NON-PROCEDURAL STATEMENTS


```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL FLAP,SPIN,TILT,ERODE,HOUSE,ERROR,REGION,HEAT,PCURV,MAXPOW
.EDGE,ONCE,QJEX,JUMP
DIMENSION A(5),B(5),DFAULT(10),DDR(5)
COMMON /REAL8/ RO1(11),FT,VJ,RI2(15),ASOL(5),RO2(190),AIS,
RI6(191),CONTM,
RO9(365),FR(3),PC3(3),EN(3),RO4(442),DPOW,GAMMAX,DMIN,DPOMAX,
CHENC2,RI7(2),CHENC,RI3(5),PAN,PI1,
RT,RO5(6),POWR,DPOWR,TAUPOW,RO6(17),DMAX,RIS,DENSIT,RIO(5),DPOWDD,
RI4(5),TPOWER,
DEGRAD,PMAXO,PPMAXC,RPOWO,DPOWD,RO7(56),X(50),RGR(700)
COMMON /INTGAN/ IC1,MODE,IO2(2),LINE,IO4(67),ISPIN,IO3(927)
COMMON /LOGIC4/ ERROR,LO1(17),REGION,ERODE,FLAP,SPIN,TILT,
HOUSE,LO2(2),QJEX,LO5,
            HEAT,LO3(7),PCURV,MAXPOW,EDGE,LO4(46))
COMMON /ITERAT/ BC1(560),CONX(70),BO2(70),C(70),CC(70),BO3(190)
EQUIVALENCE (A1,A(1)),(A2,A(2)),(A3,A(3)),(A4,A(4)),(A5,A(5))
,(B1,B(1)),(B2,B(2)),(B3,B(3)),(B4,B(4)),(B5,B(5)),(D,DENSIT)
DATA DFAULT /.627D0,5.3059D0,-10.03760D0,7.10730D0,-2.00210D0,
1.39632851084268I00, .66525959362495190D0, .9693829967927357D0,
1.94032666681499D0-2,1.57691309551123D0D0/
DATA DDR /6.D0,8.75D0,12.D0,15.75D0,20.D0/
DATA ONCE /.FALSE./
FORMAT(' ITERATION FOR POWER DENSITY FAILED. DIT,F,FP,BSOL,COEF,D
E'/1H ,1P6D20.12)
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
19      FORMAT(1H ,30HPOWER DENSITY ITERATION FAILED)
12      FORMAT(1H ,9HFAIL X(I))
28      FORMAT(1H ,69HCOMBINATION OF DEGRADATION AND HOUSEKEEPING POWER OP
        TIONS NOT ALLOWED)
38      FORMAT(1HO,33HSOLAR POWER LAW ITERATION FAILURE/1HO,
        11HAT RADIUS =D25.16,37H AU, POWER DERIVATIVE W R T  RADIUS =
        D12.4/1H )
39      FORMAT(1H ,48HSOLAR POWER LAW ITERATION FAILURE  CASE SKIPPED /1H
        )
37      FORMAT(1H ,27HUNACCEPTABLE  CASE SKIPPED /1H )
40      FORMAT(1HO,33HSOLAR POWER LAW ITERATION FAILURE/1HO,
        11HAT RADIUS =D25.16,12H AU, POWER =D12.4/1H )
42      FORMAT(1HO,15HMAXIMUM POWER =D25.16,12H AT RADIUS =D25.16,4H AU )
43      FORMAT(1H ,22HZERO POWER AT RADIUS =D25.16,4H AU )
44      FORMAT(1HO,38HSOLAR POWER LAW ITERATION FAILURE. D =1PD15.6,
        10H  DPCWDD =D15.6)
49      FORMAT(1HO,17HMINIMUM DENSITY =1PD25.16,5X15HMAXIMUM DPCWD =D25.16
        )
```

Name: SPRINT

Calling Argument: IND

Referenced Sub-programs: ALBEDO, CARKEP, EFM, PUNCH, SOLAR, TFORM, VCROSS, VDOT, VMAG, VSCAL

Referenced Commons: INTGR4, ITERAT, LOGIC4, REAL8, SOLSYS

Entry Points: None

Referencing Sub-programs: INTERP, TAP

Discussion: Subroutine SPRINT produces the standard block print of trajectory and spacecraft variables at the current time point of the final trajectory for the case. The program input variable MPRINT controls how frequently SPRINT is called. If MPRINT = 1, SPRINT is called at the end of each compute interval and thereby produces a complete time history of the trajectory. If MPRINT = 0, then SPRINT is called only at thrust switch points and target encounters, thereby producing what is termed the switch point summary. In either case, all switch points and target encounters are identified with a message preceding the block print. To facilitate reading the printout, a block of titles is printed at the top of each page, with the title in each print position corresponding to the numeric value printed in the same relative position of all other blocks on the page.

The block print may contain from five to eight lines of eight variables each, depending on the options invoked. The arrangement and explicit definitions of all variables included in the print are described fully in the section Auxiliary Computations.

Messages and printouts: Examples of the switch point summary and of the detailed trajectory print are illustrated on the following pages. Additional examples may be seen in the Sample Problems and Results section.

CASE 1

DETAILED TRAJECTORY PRINT


```
TIME              SEMI-MAJOR AXIS ECCENTRICITY   INCLINATION      NODE             ARG POS          RMAG             TRAVEL
R1                R2               R3               V1               V2               V3               MASS RATIO       THRUST ACC
L1                L2               L3               L4               L5               L6               L7               HAM
LG                LC               LPHI             CONE             CLOCK            HMAG             POWER FNCT       SWITCH FNCT
PSI               THETA            PHI              LATITUDE         LONGITUDE        FLT PTH ANGLE    VMAG             PROP TIME
S                 LS               DENSITY          DPOWR            DPOWD            DEGRAD           CHI              CHI REF

   EARTH                                          START OF TRAJECTORY, THRUST ON

 0.0              8.89018323D-01   1.34630548D-01   1.08645835D 01   3.08348545D 01   1.80000000D 02   1.00498235D 00   0.0
-8.62926339D-01  -5.15119079D-01   0.0              4.90151142D-01  -7.70105504D-01  -1.75131878D-01   1.00000000D 00   5.29528230D-02
-8.36582920D-02  -3.03588760D-01  -5.96206560D 00   8.18173740D-01   7.35505340D-01  -3.50584570D-01   1.00000000D 00   1.41335388D-01
 0.0              0.0              0.0              8.60527166D 01   1.57072331D 01   9.33767934D-01   9.93308384D-01   4.95797071D 00
-7.68637719D 01   8.03503788D 01   8.78167830D 01   0.0             -1.49165146D 02  -1.61139123D 00   9.29506215D-01   0.0
 0.0              0.0              9.90109272D-01  -1.33880716D 00   6.79459132D-01   1.00000000D 00   0.0              0.0

 1.82905094D 00   8.88563127D-01   1.33937587D-01   1.09646590D 01   3.08425140D 01   1.81660929D 02   1.00409628D 00   1.65844944D 00
-8.47086170D-01  -5.39095269D-01  -5.53555861D-03   5.16671053D-01  -7.53825206D-01  -1.76711655D-01   9.98313031D-01   5.30575874D-02
-5.80156962D-02  -2.80386155D-01  -5.97718508D 00   8.11640585D-01   7.39184119D-01  -1.65421669D-01   1.00997159D 00   1.41335388D-01
-1.55090461D-01  -1.71640583D-03   0.0              8.59757912D 01   1.54652900D 01   9.34142894D-01   9.93594966D-01   4.95076375D 00
-7.68816641D 01   8.01359310D 01   8.77717026D 01  -3.15871854D-01  -1.47526861D 02  -1.85799335D 00   9.30821373D-01   1.82905094D 00
 1.81252021D 00   2.40313686D-04   9.91657514D-01  -1.33907586D 00   6.77799517D-01   9.99094150D-01   0.0              0.0

 3.65551134D 00   8.89108294D-01   1.33247635D-01   1.10645863D 01   3.08652077D 01   1.83308561D 02   1.00308520D 00   3.33835680D 00
-8.30442391D-01  -5.62513932D-01  -1.11102248D-02   5.42730837D-01  -7.36778572D-01  -1.76120076D-01   9.96527323D-01   5.31715298D-02
-3.26309641D-02  -2.57119928D-01  -5.97246542D 00   8.04072694D-01   7.41641749D-01   2.01047433D-02   1.01997292D 00   1.41335388D-01
-3.10139555D-01  -3.44805796D-03   0.0              8.59051274D 01   1.52291767D 01   9.34517114D-01   9.94348020D-01   4.95792919D 00
-7.68910972D 01   7.99178358D 01   8.77245230D 01  -6.34624061D-01  -1.45887599D 02  -2.10075564D 00   9.32269375D-01   3.65551134D 00
 3.62589695D 00   4.80563212D-04   9.93958031D-01  -1.33954707D 00   6.75991845D-01   9.98188694D-01   0.0              0.0

 5.47904420D 00   8.89654193D-01   1.32561327D-01   1.11642662D 01   3.09025070D 01   1.84943432D 02   1.00195123D 00   5.00982600D 00
-8.13015338D-01  -5.85348483D-01  -1.67174249D-02   5.68309111D-01  -7.18977962D-01  -1.79355365D-01   9.94944411D-01   5.32945717D-02
-7.54035813D-03  -2.33633454D-01  -5.96892677D 00   7.95483963D-01   7.42847294D-01   2.05842560D-01   1.02999446D 00   1.41335388D-01
-4.64968102D-01  -5.19492968D-03   0.0              8.58407951D 01   1.49988344D 01   9.34890763D-01   9.94565363D-01   4.94947592D 00
-7.68922925D 01   7.96558814D 01   8.76751461D 01  -9.56016934D-01  -1.44247160D 02  -2.33939824D 00   9.33848437D-01   5.47904420D 00
 5.44024361D 00   7.20471029D-04   9.96108541D-01  -1.34021497D 00   6.74037735D-01   9.97283574D-01   0.0              0.0

 7.29931898D 00   8.90201186D-01   1.31878517D-01   1.12636023D 01   3.09539817D 01   1.86566074D 02   1.00069659D 00   6.68295006D 00
-7.94826326D-01  -6.07573227D-01  -2.23505545D-02   5.93384689D-01  -7.00436259D-01  -1.80415849D-01   9.93262326D-01   5.34265978D-02
 1.72207169D-02  -2.10570831D-01  -5.95557231D 00   7.85890577D-01   7.42770446D-01   3.91638510D-01   1.04002661D 00   1.41335388D-01
-6.19397159D-01  -6.95697632D-03   0.0              8.57828598D 01   1.47742398D 01   9.35264008D-01   9.95444160D-01   4.93542136D 00
-7.68854362D 01   7.94698595D 01   8.76234694D 01  -1.27980744D 00  -1.42605341D 02  -2.57364815D 00   9.35556537D-01   7.29931898D 00
 7.25567213D 00   9.59759835D-04   9.98608277D-01  -1.34107325D 00   6.71938863D-01   9.96378737D-01   0.0              0.0

 8.24087971D 00   8.90485092D-01   1.31527034D-01   1.13148946D 01   3.09860784D 01   1.87402013D 02   1.00000000D 00   7.55037467D 00
-7.85111667D-01  -6.18832234D-01  -2.52766691D-02   6.06169892D-01  -6.90544197D-01  -1.80895955D-01   9.92392452D-01   5.34984349D-02
 2.99066149D-02  -1.98544317D-01  -5.95245087D 00   7.80532309D-01   7.42216613D-01   4.87857508D-01   1.04522451D 00   1.41335388D-01
-6.69179338D-01  -7.87570812D-03   0.0              8.57553834D 01   1.46601563D 01   9.35457251D-01   9.95910100D-01   4.92594627D 00
-7.68787704D 01   7.93511072D 01   8.75557598D 01  -1.44840072D 00  -1.41754226D 02  -2.69318046D 00   9.36491530D-01   8.24087971D 00
 8.19657239D 00   1.08338283D-03   1.00000000D 00  -1.34159050D 00   6.70795248D-01   9.95910100D-01   0.0              0.0
```

SWITCH POINT SUMMARY

CASE 1

TIME	SEMI-MAJOR AXIS	ECCENTRICITY	INCLINATION	E	ARG POS	RMAG	TRAVEL
R1	R2	R3	V1	--	V3	MASS RATIO	THRUST C
L1	L2	L3	L4	L5	L6	L7	HAM
LG	LC	LPHI	CONE	CLOCK	HMAG	POWER FNCT	SWITCH FNCT
PSI	THETA	PHI	LATITUDE	LONGITUDE	FLT PTH ANG_E	VMAG	PROP TIME

START OF TRAJECTORY. THRUST ON

EARTH

0.0	8.98977775D-01	1.21180874D-01	9.22842759D 00	3.08348545D 01	1.80000000D 02	1.00498235D 00	0.0
-8.62926339D-01	-5.15119079D-01	0.0	4.96430685D-01	-7.80199267D-01	-1.50185951D-01	1.00000000D 00	4.46544954D-02
1.43302209D-03	-3.71273470D-01	-5.19222332D 00	7.08559116D-01	6.35286348D-01	-3.26439161D-01	1.00000000D 00	9.49111444D-02
0.0	0.0	0.0	8.60527156D 01	1.71366559D 01	9.41156981D-01	9.93095163D-01	4.19307526D 00
-7.70840302D 01	8.06481424D 01	8.79184672D 01	0.0	-1.49165146D 02	-1.61208637D 00	9.36861858D-01	0.0

SWITCH THRUST OFF

8.03383162D 01	9.24015325D-01	1.03022217D-01	1.19678031D 01	4.06593345D 01	2.52732848D 02	8.89443736D-01	8.23696730D 01
3.41100615D-01	-8.02334532D-01	-1.76124469D-01	9.42332071D-01	5.25634816D-01	-4.56244954D-02	9.37112533D-01	4.79824670D-02
3.97728374D-01	-4.69486344D-02	-1.11930619D 00	-4.70177337D-02	-6.32061682D-01	5.36711351D 00	1.25303011D 00	9.49111521D-02
-3.87479887D 00	-7.30163733D-02	0.0	3.74079507D 01	1.51105162D 01	9.56142362D-01	1.00000000D 00	-3.99680289D-15
-6.04197599D 01	4.47840023D 01	6.94899910D 01	-1.14209339D 01	-6.69679493D 01	-5.51206966D 00	1.07998300D 00	8.03383162D 01

SWITCH THRUST ON

1.03587476D 02	9.24015325D-01	1.03022217D-01	1.19676031D 01	4.05593345D 01	2.51687763D 02	8.52502666D-01	1.11324607D 02
6.63125730D-01	-5.07008397D-01	-1.73111372D-01	6.28717705D-01	9.29830855D-01	6.25634476D-02	9.37112533D-01	4.79824670D-02
4.07327513D-01	-3.99412178D-01	1.04297296D 00	1.31127036D-01	-1.05012549D 00	5.22760591D 00	1.25303011D 00	9.49111521D-02
-3.87479887D 00	-7.30163733D-02	0.0	9.04770078D 01	1.74834324D 02	9.56142362D-01	1.00000000D 00	-1.33226763D-15
7.32179111D 01	-3.52154795D 00	7.32505352D 01	-1.17161055D 01	-3.74005722D 01	-3.91080390D 00	1.12418886D 00	8.03383162D 01

SWITCH THRUST OFF

2.20047131D 02	9.26812233D-01	9.30891521D-02	1.57548141D 01	4.10752060D 01	7.03114536D 01	9.29508201D-01	2.60280593D 02
-3.17352054D-01	8.40718021D-01	2.37626552D-01	-9.84539957D-01	-2.93723195D-01	1.18969743D-01	8.45914448D-01	5.31554595D-02
-2.43332572D-01	-6.42014437D-01	1.19757838D 00	-9.53083278D-01	4.97309569D-01	-4.74623336D 00	1.61188302D 00	9.49111675D-02
-8.08440714D 00	-2.04809354D-01	0.0	8.92745773D 01	1.62097586D 02	9.58530572D-01	1.00000000D 00	-1.75415238D-14
6.39873057D 01	1.08435053D 02	9.79719829D 01	1.48119626D 01	1.10680388D 02	5.33875555D 00	1.03571631D 00	1.96797972D 02

SWITCH THRUST ON

2.48414939D 02	9.26812233D-01	9.30891521D-02	1.57548141D 01	4.10752060D 01	9.98993508D 01	9.72918798D-01	2.39868490D 02
-7.32170229D-01	5.85470265D-01	2.60235064D-01	-6.83124837D-01	-7.13729664D-01	-2.51642405D-02	8.45914448D-01	5.31554595D-02
-5.74162417D-01	-3.78433842D-01	-1.19695974D 00	-4.74087039D-01	5.95741621D-01	-4.83344364D 00	1.61188302D 00	9.49111675D-02
-8.08440714D 00	-2.04809354D-01	0.0	8.59477730D 01	3.68564479D 01	9.58530572D-01	1.00000000D 00	2.88657986D-15
-5.84964179D 01	9.92381776D 01	9.48121931D 01	1.55142903D 01	1.41352882D 02	4.51838832D 00	9.88282761D-01	1.96797972D 02

SPRINT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
O(70)	U	ITERAT	Array of iterator independent-variables, in program internal units.
R(2)	U	REAL8	Spacecraft solar distance, r, at start of computation step (R(1)) and instantaneously (R(2)), in AU.
X(50)	U	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
FT	U	REAL8	Reference thrust acceleration, g, in AU/tau^2.
RC	SU	REAL8	Cube of spacecraft solar distance, r^3, in AU^3.
RT	SU	REAL8	Spacecraft solar distance, r, in AU.
XD(50)	U	REAL8	Array of trajectory dependent-variable derivatives; XD(17) is $dt/d\beta = r^n$, used for conversion from generalized derivatives to time derivatives.
APL(2,70)	U	SOLSYS	Array of planet names.
AVJ	U	REAL8	Inverse of jet exhaust speed, 1/c, in $EMOS^{-1}$.
DEG	U	REAL8	Conversion factor between radians and degrees; number of degrees in one radian.
ETH(3)	SU	REAL8	Thrust unit vector, $\bar{e}_t$.
IND	UX		Calling argument employed to select specific printout desired on each call.
LEG	U	INTGR4	Counter indicating the current trajectory-segment.
PHI	U	REAL8	Thrust angle, ϕ, in radians.

SPRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
PLC	U	REAL8	First component of thrust switch function, σ_1.
PSI	U	REAL8	Thrust angle ψ, in radians.
SPV(3)	SU	REAL8	Star unit vector, $\bar{s}$.
TAU	U	REAL8	Propulsion time, τ, in tau.
DMAX	U	REAL8	Maximum value of density, d_{max}, in AU^{-2}.
EDGE	U	LOGIC4	Indicator for solar arrays being oriented edgewise to the sun; used only if power degradation is simulated.
ETHD(3)	U	REAL8	Thrust unit vector time-derivative, $\dot{\bar{e}}_t$, in tau^{-1}.
IEND	S	INTGR4	Indicator for endpoint of trajectory.
POWR	SUA	REAL8	Power function, $q\gamma$.
ALTAU	U	REAL8	Propulsion-time adjoint variable, λ_τ.
ANGLE	U	REAL8	Travel angle, θ_t, in radians.
COAST	U	LOGIC4	Indicator for coast or thrust phase.
CONDS	SU	REAL8	Distance conversion factor, AU to meters.
CONSP	SU	REAL8	Speed conversion factor, from AU/tau to meters/second.
CONTM	U	REAL8	Time conversion factor, tau to days.
DPOWD	U	REAL8	Ratio of housekeeping power to reference power, p_h/p_{ref}.
DPOWR	U	REAL8	$q\,\partial\gamma/\partial r$.
ERODE	U	LOGIC4	Power degradation option indicator.

SPRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
EXTRA	U	LOGIC4	Indicator for extra printout for each trajectory block print (computation step).
INTER(5)	U	INTGR4	Array of indices which select the correct orbital elements for the intermediate targets.
KOUNT	U	INTGR4	Case number of computer run.
MOPTX(5)	U	INTGR4	The target-numbers of the successive intermediate targets.
MOPT2	U	INTGR4	Launch planet number.
MOPT3	U	INTGR4	Planet-number of primary target.
PCURV	U	LOGIC4	Indicator for condition in which solar arrays are oriented to receive the maximum power permissible under the current power-curve assumption, or to be tilted away from the maximum permissible due to degradation considerations.
QMORE	U	LOGIC4	Logical flag indicating whether program is currently operating under the control of subroutine MORE which performs the ballistic swingby continuation analysis.
THETA	U	REAL8	Thrust angle θ, in radians.
TRACK	U	LOGIC4	Indicator for trajectory long block printout (at each computation step).
YLONG	S	REAL8	Initial heliocentric longitude which is saved for use in the next case.
DEGRAD	U	REAL8	Damage factor, q.
DENSIT	U	REAL8	Power density, d, in AU^{-2}.
FIXTHR	U	LOGIC4	Indicator for fixed thrust-angle.

SPRINT EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
INTERX	U	INTGR4	Index used in subroutine EFM which indicates which array locations are applicable in selecting orbital elements.
LCOUNT	SU	INTGR4	Counter of the number of integrated points along the trajectory.
LEGMAX	U	INTGR4	Total (maximum) number of trajectory-segments comprising the trajectory.
MPUNCH	U	INTGR4	Punched-card and trajectory-tape generation control indicator.
OUTECL	U	LOGIC4	Extra-ecliptic mission indicator.
PLANET	U	LOGIC4	Ephemeris-option indicator.
RTSWIT	U	REAL8	Critical solar distance corresponding to a special point in the solar power curve, in AU.
SWITCH(2)	U	REAL8	Thrust switch function, σ, defined in similar manner to R(2).
TDATEX	U	REAL8	Reference Julian date, less 2400000, defined by program input quantity MYEAR, etc.

CHART TITLE SUBROUTINE SPRINT(IND)

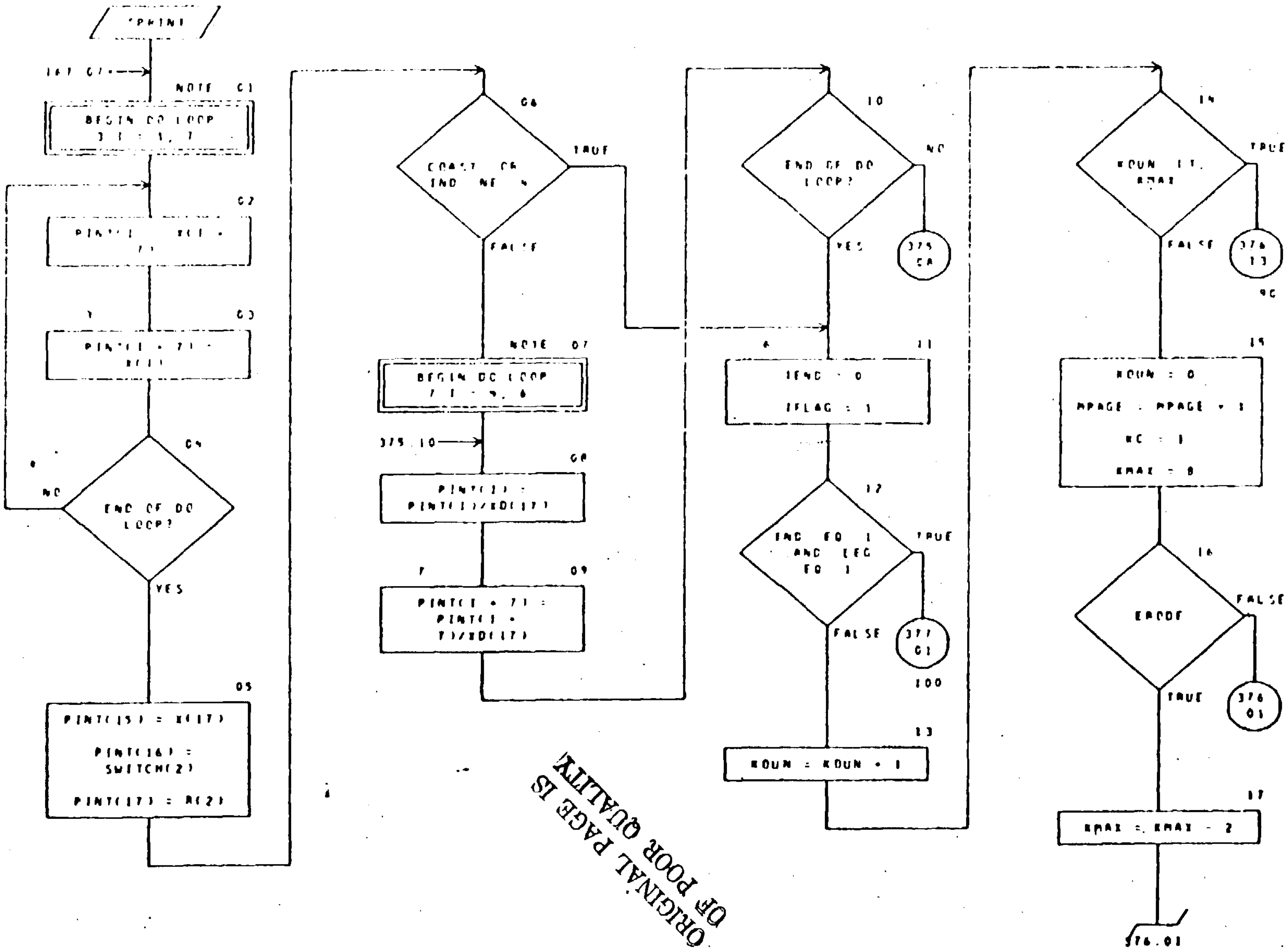
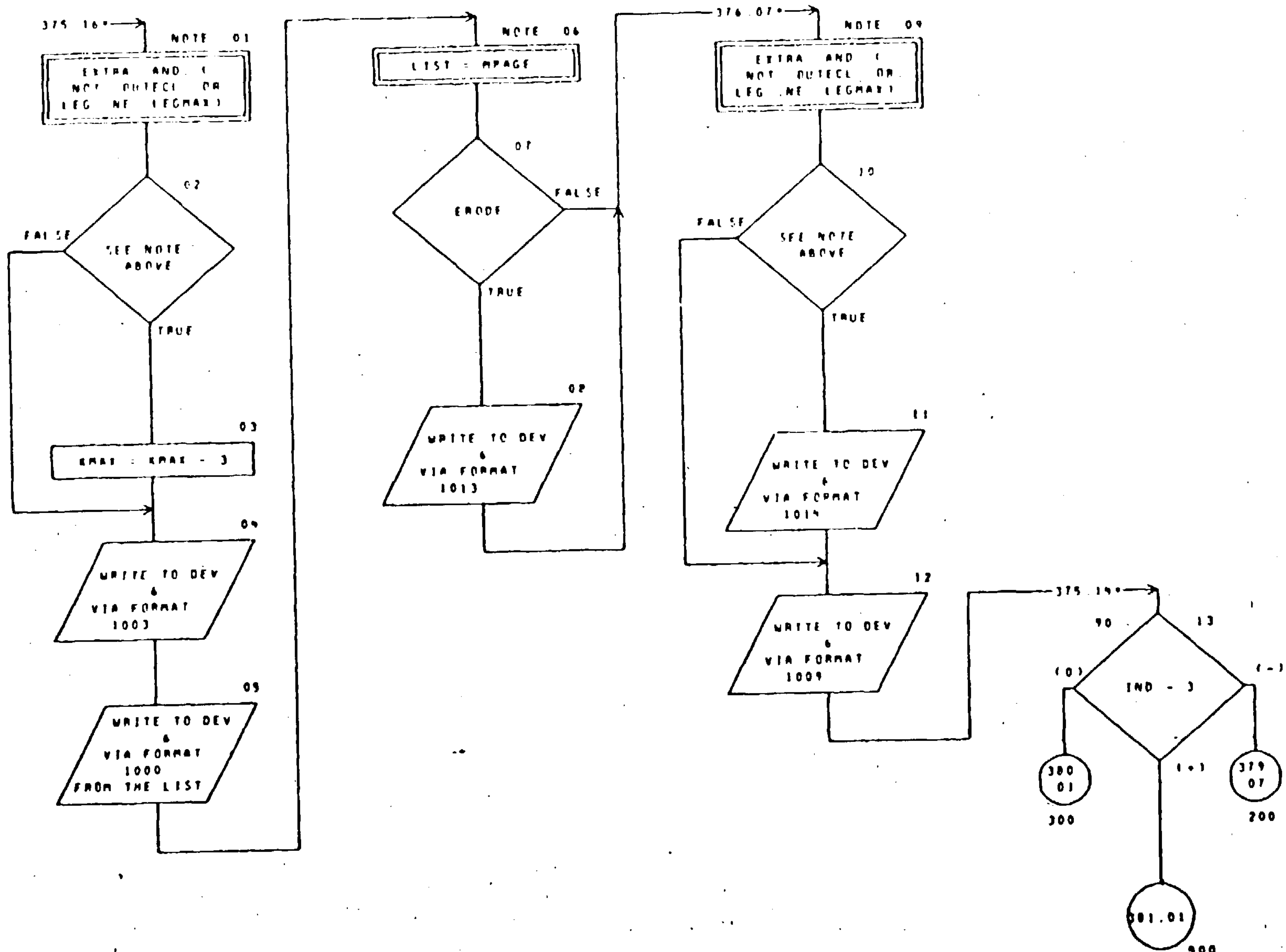


CHART TITLE - SUBROUTINE SPRINT(IND)



SPRINT-9

CHART TITLE - SUBROUTINE SPRINT(IND)

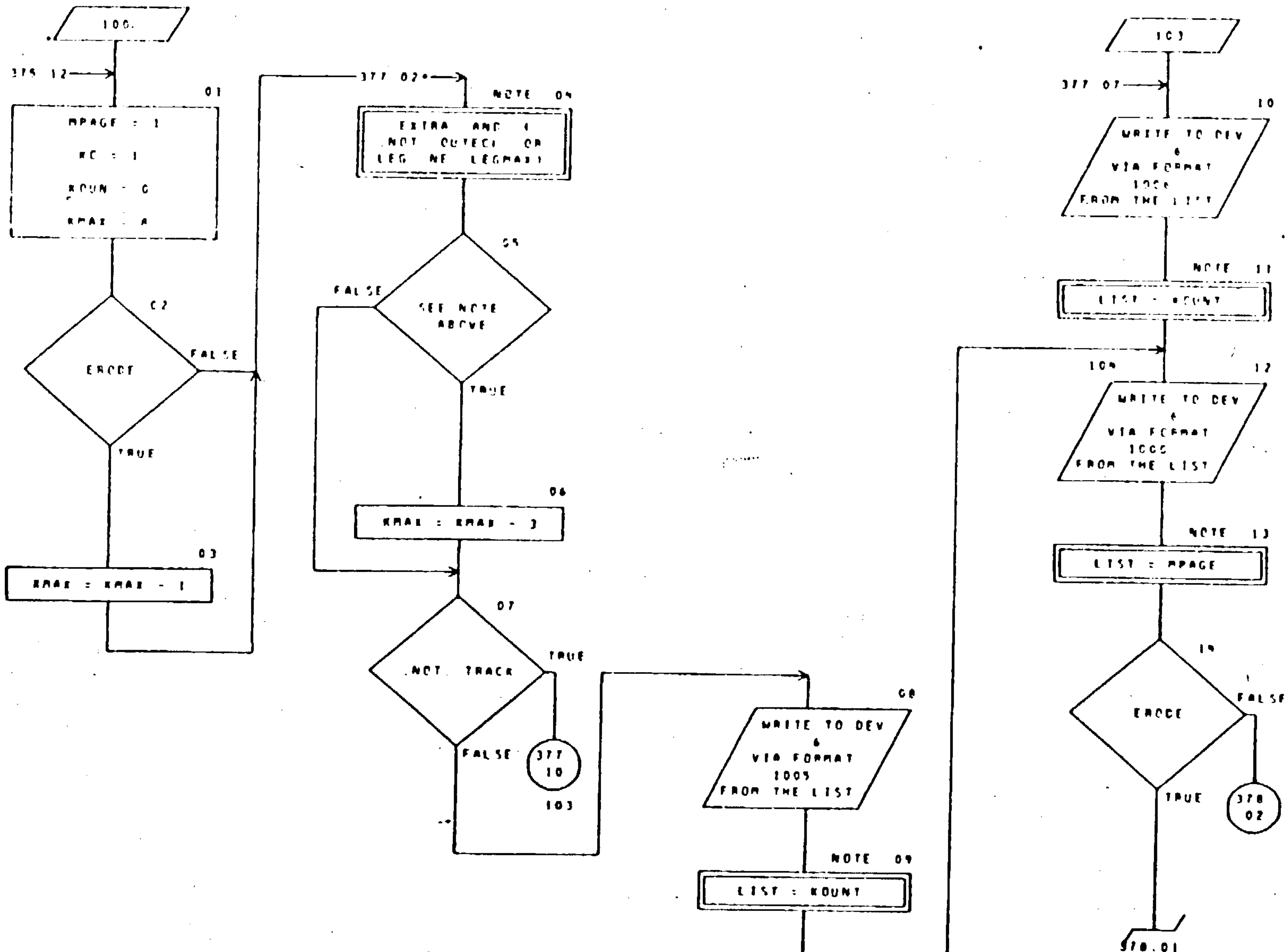
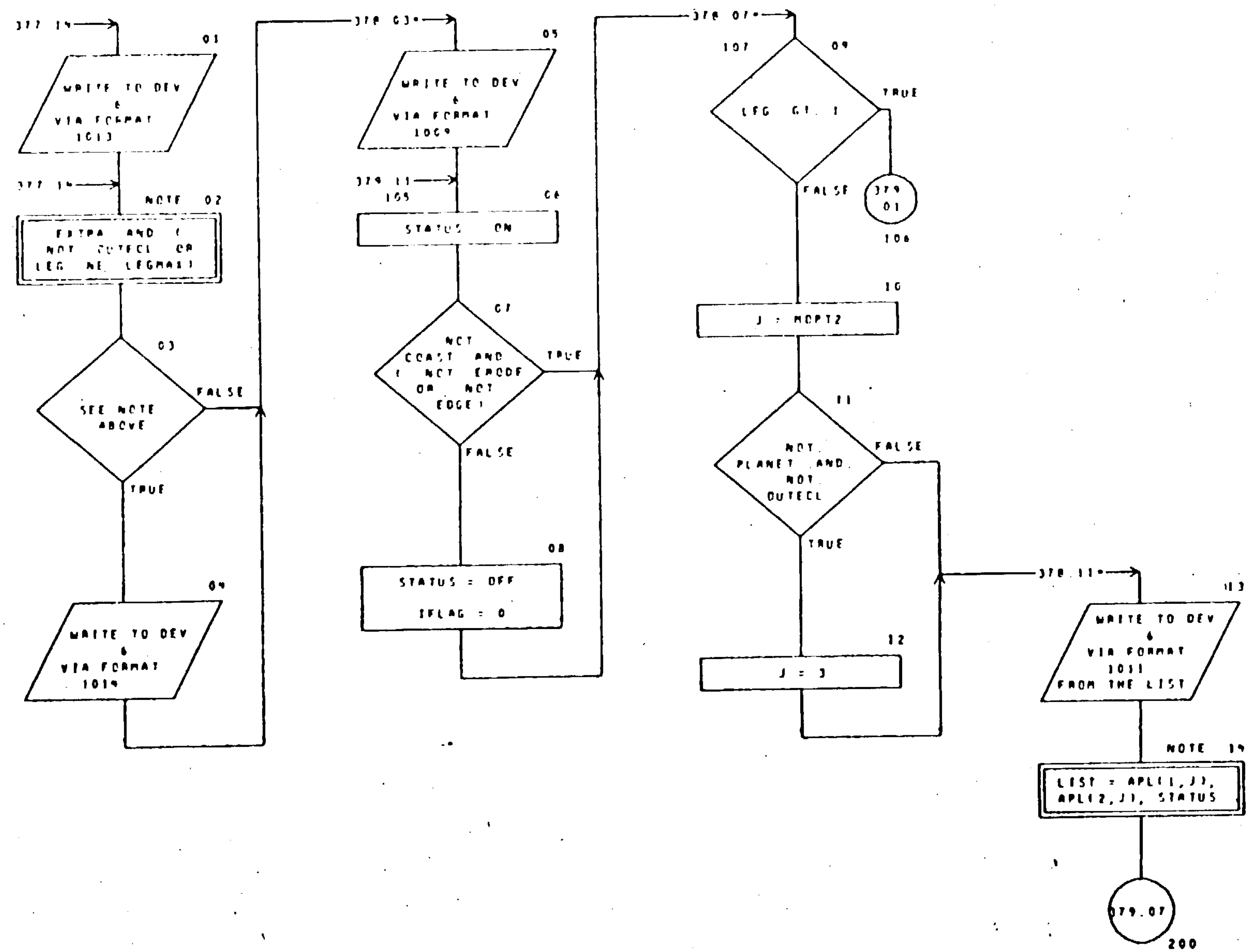


CHART TITLE - SUBROUTINE SPRINT(IND)



SPRINT-11

CHART TITLE - SUBROUTINE SPRINT(IND)

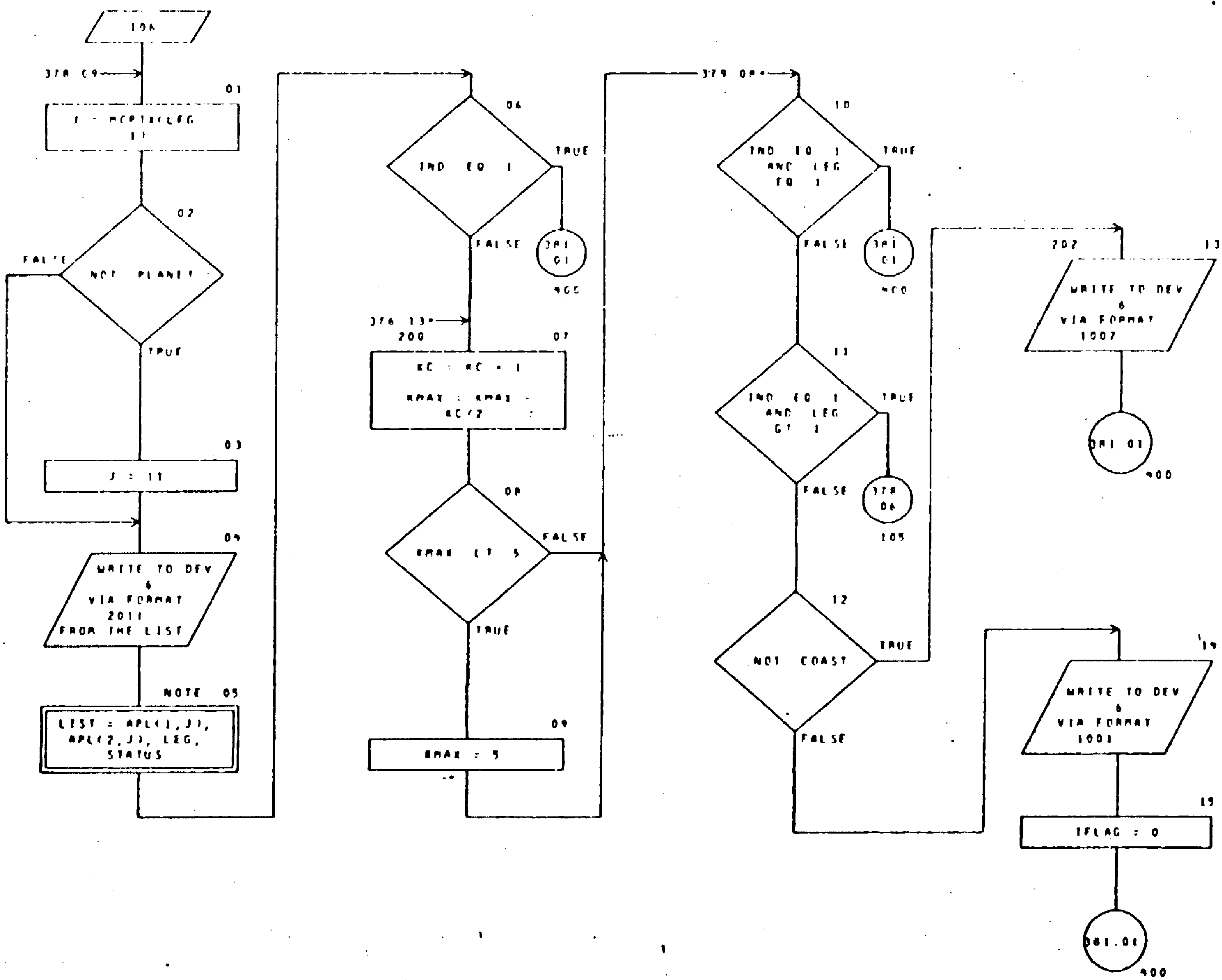
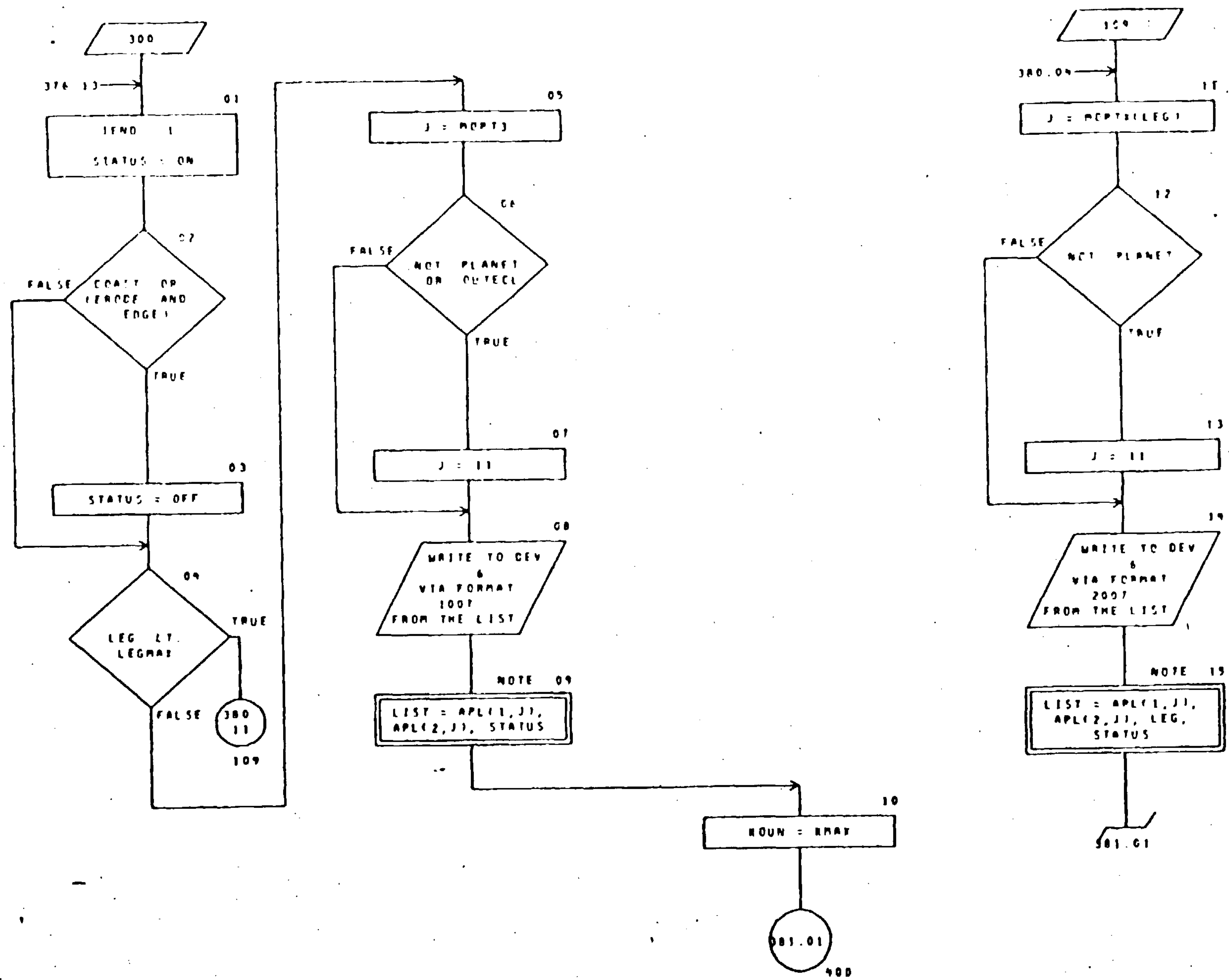


CHART TITLE - SUBROUTINE SPRINT(IND)



SPRINT-13

CHART TITLE - SUBROUTINE SPRINT(IND)

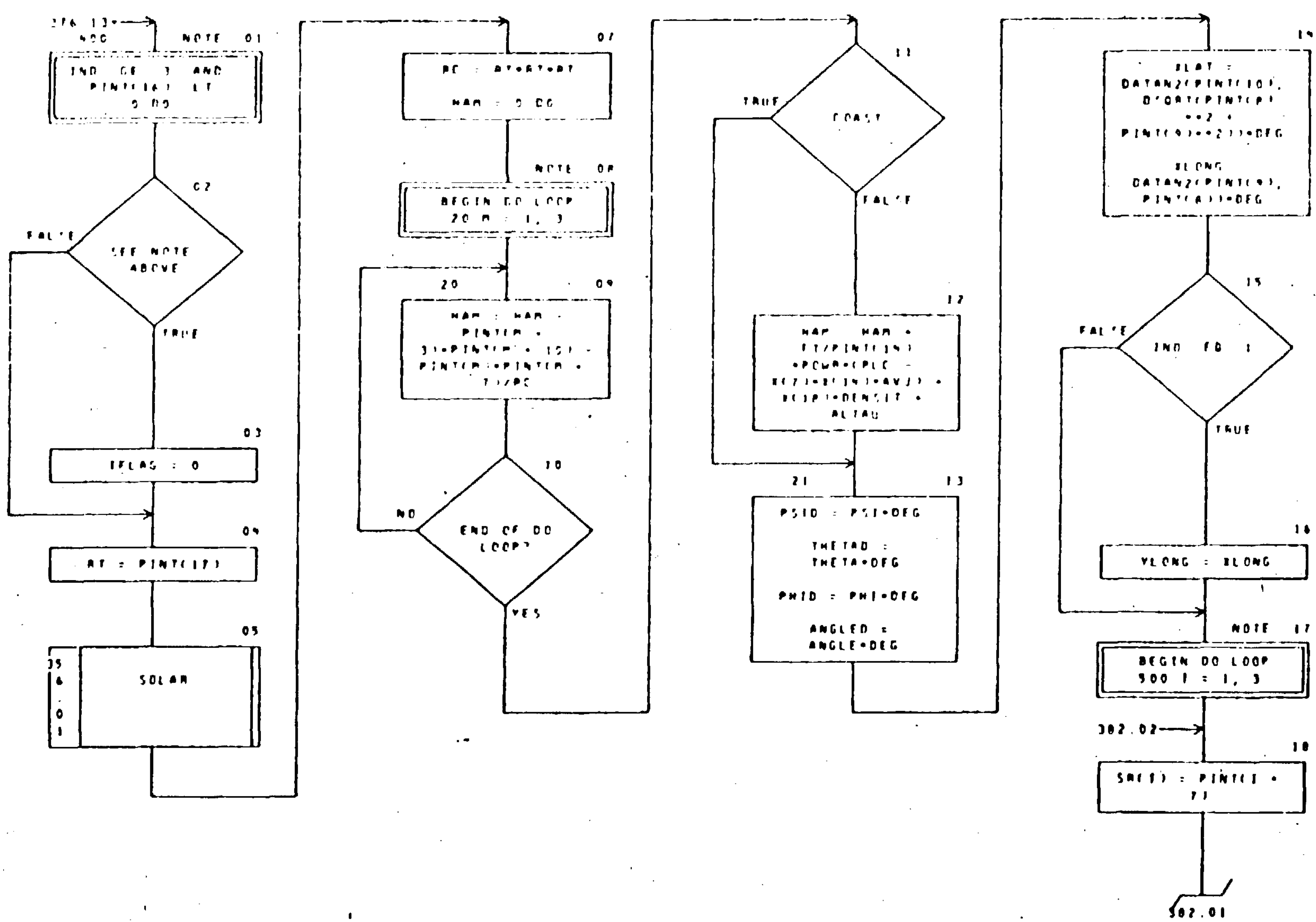


CHART TITLE - SUBROUTINE SPRINT(IND)

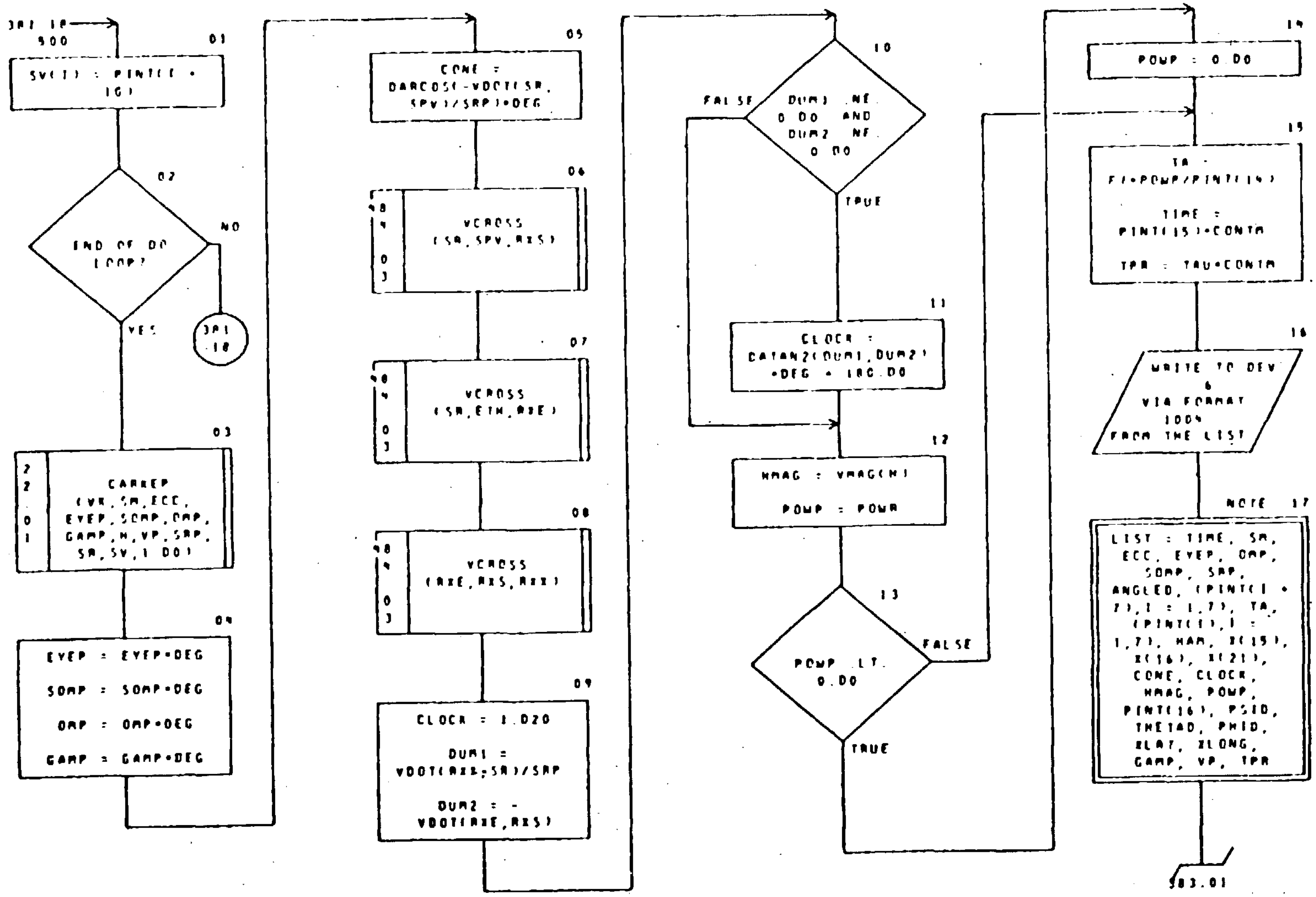


CHART TITLE - SUBROUTINE SPRINT(IND)

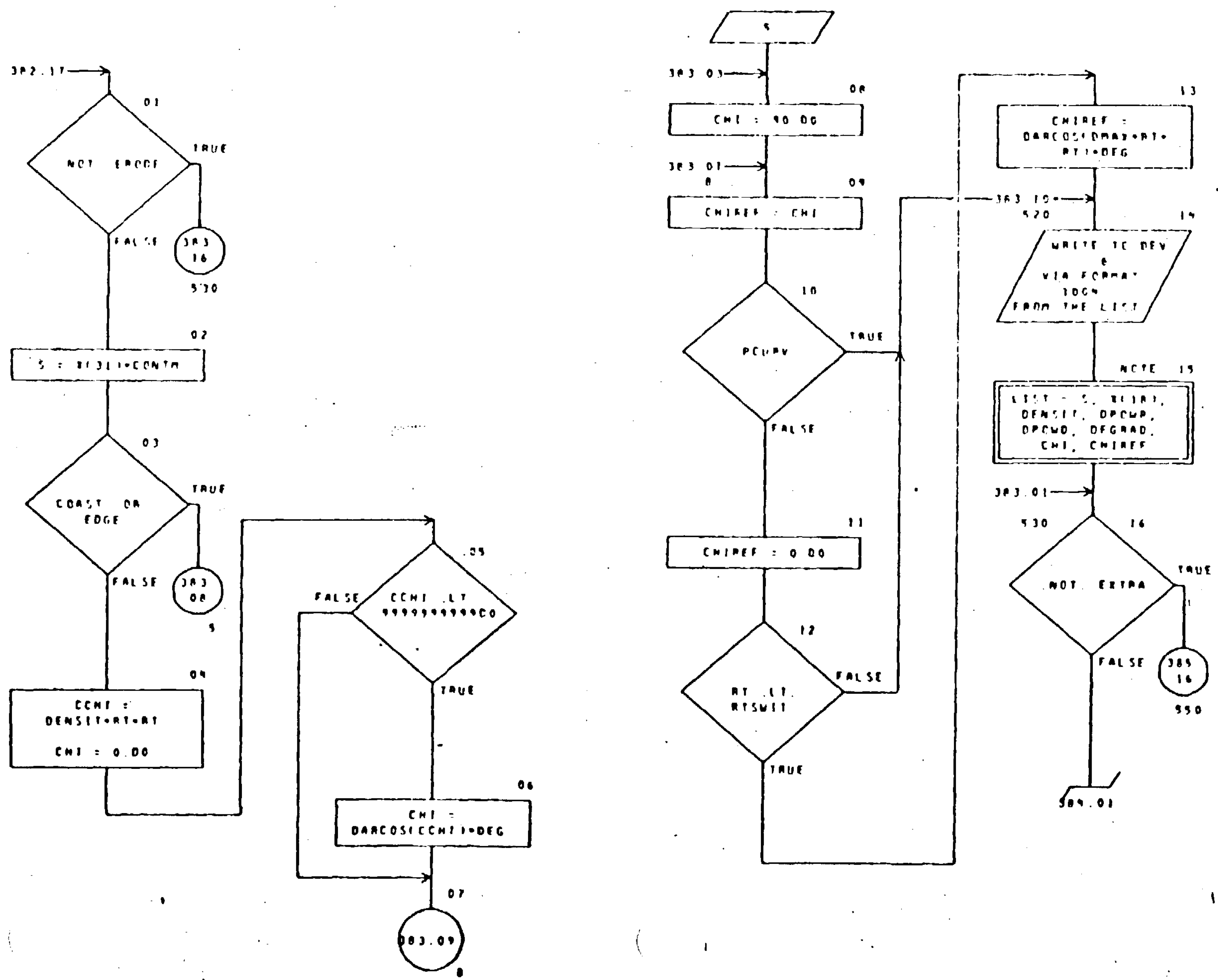
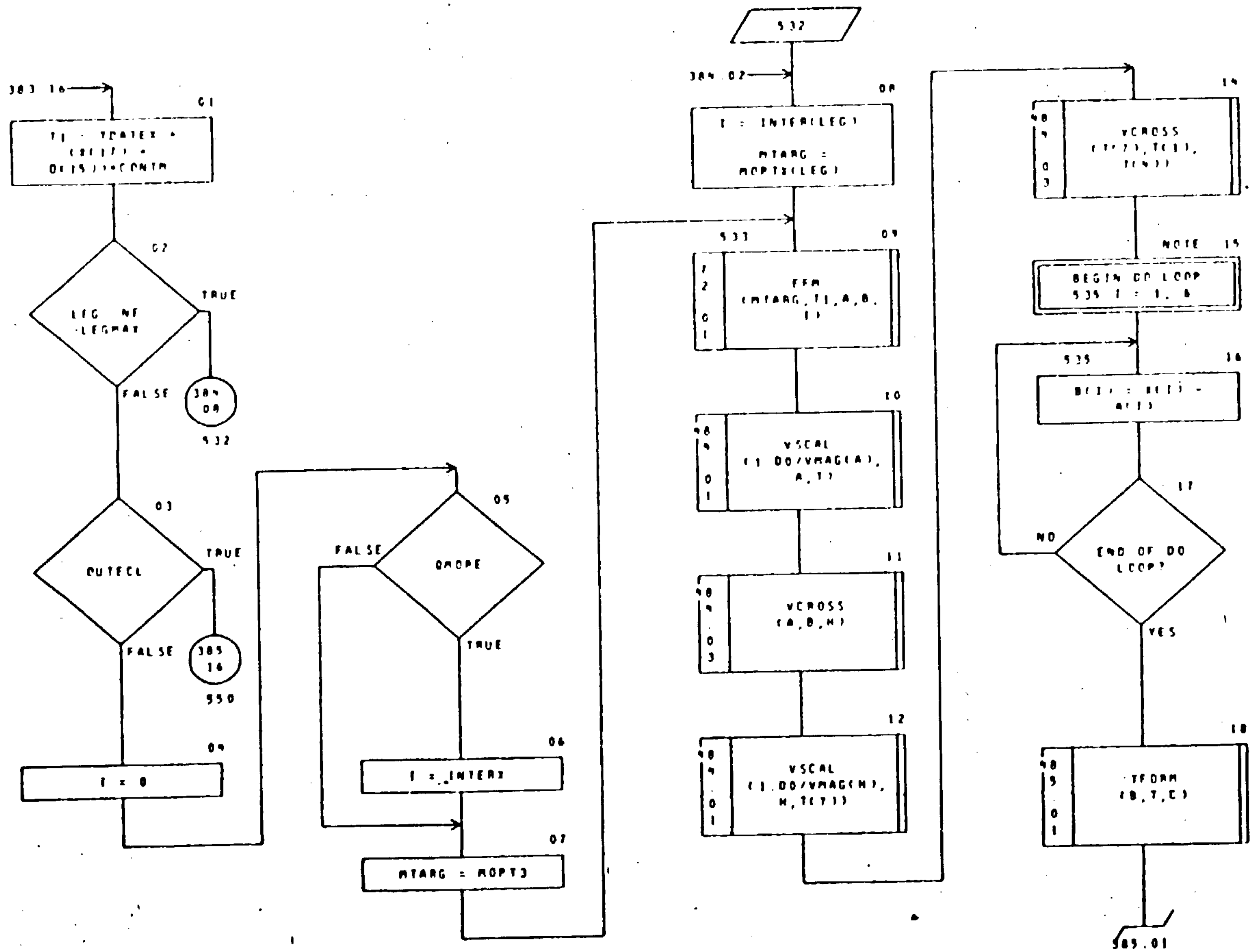


CHART TITLE - SUBROUTINE SPRINT(IND)



SPRINT-17

CHART TITLE - SUBROUTINE SPRINT(IND)

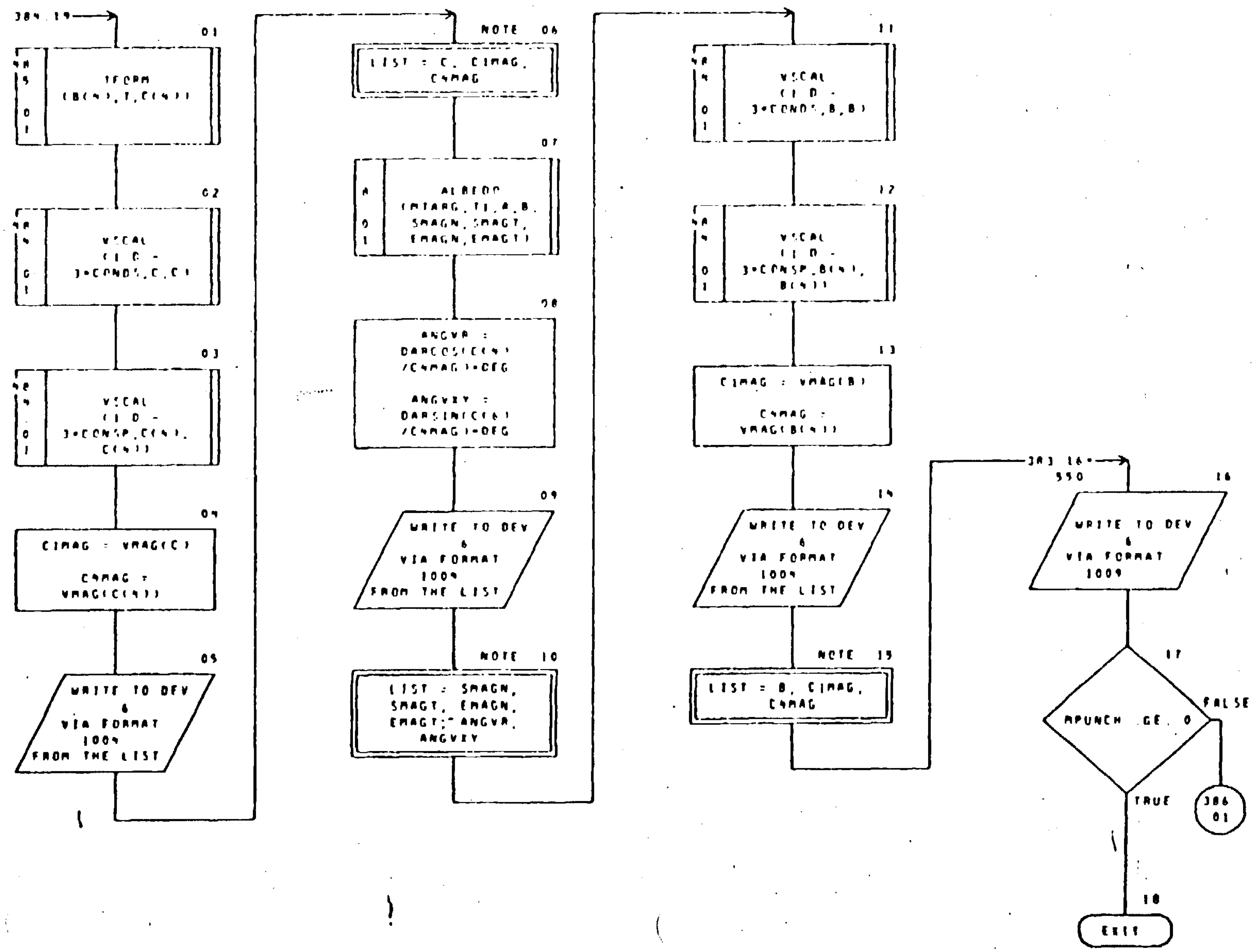
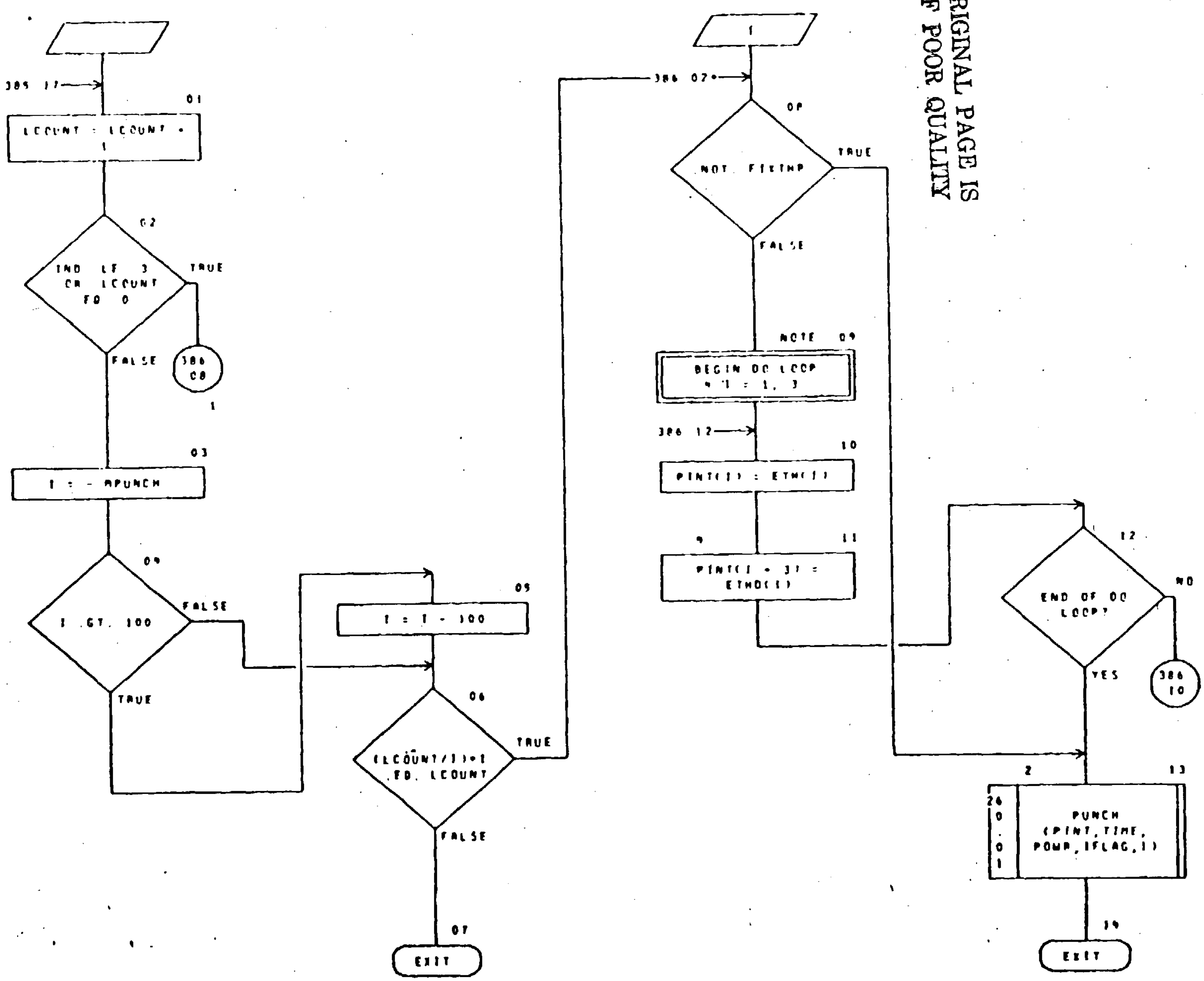


CHART TITLE - SUBROUTINE SPRINT(IND)



ORIGINAL PAGE IS OF POOR QUALITY

SPRINT-19

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL COAST,TRACK,PLANET,FIXTHR,ERODE,PCURV,EDGE,EXTRA,OMORE
     ,OUTECL
      DIMENSION PINT(12),SR(3),SV(3),VR(3),H(3),RX5(3),RXE(3),RXB(3)
     ,A(6),B(6),T(9),C(6)
      COMMON /REAL8/ RO1(11),F?,RO2(14),ANGLE,RO3(3),TDATES,RO4(2),
     SPV(3),RO5(12),ALTAU,RO6(6),RTIWT,
     PSI,THETA,PHI,RO9(14),R(2),R22(2),SWITCH(2),RO7(106),DEG,RQR(2),
     CONTM,CONSP,RO9(4),CONDS,R25(53),
                                       FTH(3),FTHD(3),R23(12),PIC,
           R12(303),RT,R13,RC,R14(4),POWR,DPOWR,R15(12),YLONG,AV1,
     R16(4),DMAX,R11,DENSIT,R19(12),DEGRAD,R20(3),CPOWR,R21(56),
           X(50),XD(50),R17(3),TAU,R18(696)
      COMMON /INTGR4/ IO1(23),MOPT2,MOPT3,IO6(14),MPUNCH,LCOUNT,IO3(12),
     KOUNT,IO4(17),IEND,IO5(54),LEG,LEGMAX,MOPTX(5),INTER(5),IO2(7),
     INTERX,IO7(854)
      COMMON /LOGIC4/ LC1(9),OUTECL,FIXTHR,LO5(8),ERODE,LO2(4),TRACK,
     PLANET,
     LO3(5),COAST,LO4(4),PCURV,LO6,EDGE,LO7(6),EXTRA,OMORE,LO8(453)
      COMMON /ITERAT/ BG1(700),C(70),BO2(210)
      COMMON /SOLSYS/ SOLO1(140),APL(2,70)
      DATA VX /0.D0,0.D0,1.D0/
      DATA ON,OFF /4H ON ,4H OFF/
 1000 FORMAT( 1H , 123X, 4HPAGE, I3, / 2X,5H TIME,11X,14H SEMI-MAJOR AXI
     S,13H ECCENTRICITY,3X,12H INCLINATION,4X,5H NODE,11X,
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      8H ARG POS,8X,5H RMAG,11X,7H TRAVEL/2X,3H R1,13X,3H R2,13X,3H R3,
      13X,3H V1,13X,3H V2,13X,3H V3,13X,11H MASS RATIO,5X,11H THRUST ACC
      /2X,3H L1,13X,3H L2,13X,3H L3,13X,3H L4,13X,3H L5,13X,3H L6,13X,
      3H L7,13X,4H HAM/2X,4H LG ,12X,3H LC,13X,5H LPHI,11X,
      5H CONE,11X,6H CLOCK,10X,5H HMAG,11X,
      11H POWER FNCT,5X,12H SWITCH FNCT/2X,4H PSI,12X,
      6H THETA,10X,4H PHI,12X,9H LATITUDE,7X,10H LONGITUDE,6X,
      14H FLT PTH ANGLE,2X,5H VMAG,11X,10H PROP TIME)
1001  FORMAT (56X,'SWITCH THRUST OFF'/1H )
1002  FORMAT (56X,'SWITCH THRUST ON'/1H )
1003  FORMAT (1H1)
1004  FORMAT   (1X,1P8D16.8 )
1005  FORMAT (5H1CASE,I3,45X,25HDETAILED TRAJECTORY PRINT)
1006  FORMAT (5H1CASE,I3,47X,20HSWITCH POINT SUMMARY)
1007  FORMAT(1H ,2A8,33X25HEND OF TRAJECTORY, THRUSTAN/1H )
1009  FORMAT(1H0)
1011  FORMAT(1H ,2A8,32X27HSTART OF TRAJECTORY, THRUSTAN/1H )
1013  FORMAT(1H ,2X1HS,15X2HLS,14X7HDENSITY,9X 5HOPOWR,11X5HOPOWD,
      11X6HDEGRAD,10X3HCHI,13X7HCHI REF)
1014  FORMAT(1H ,8H  R1 REL,10X6HR2 REL,10X6HR3 REL,10X6HV1 REL,
      10X6HV2 REL,10X6HV3 REL,10X8HRMAG REL,8X8HVMAG REL/2X,
      12H S/C NUC MAG,5X11HS/C TOT MAG,5X11HGEO NUC MAG,5X11HGEO TOT MAG
      ,5X8HANG(V,R),8X9HANG(V,XY)/2X,
      11H R1 REL ECL,6X10HR2 REL ECL,6X10HR3 REL ECL,
      6X10HV1 REL ECL,6X10HV2 REL ECL,6X10HV3 REL ECL,
```

ORIGINAL PAGE IS OF POOR QUALITY

```
         6X8HHMAG FCL,8X8HVMAG FCL)
2007     FORMAT(1H ,2A8,20X25HEND OF TRAJECTORY SEGMENT12,
         8H, THRUST=A4/1H )
2011     FORMAT(1H ,2A8,27X27HSTART OF TRAJECTORY SEGMENT12,
         8H, THRUST=A4/1H )
```

Name: STEP

Calling Argument: DBETA, GO

Referenced Sub-programs: ANSTEP, FUNCT, RKSTEP

Referenced Commons: LOGIC4

Entry Points: None

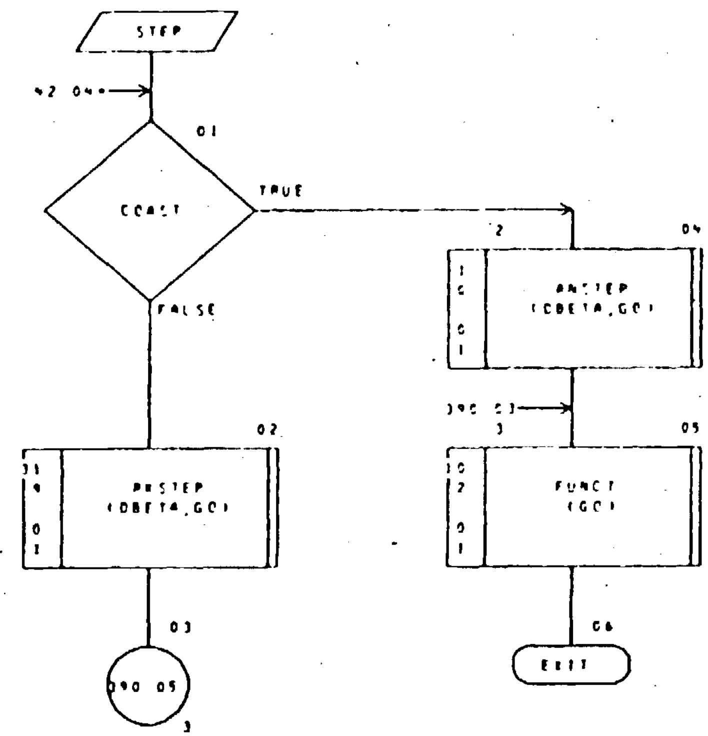
Referencing Sub-programs: CHECK, INTERP, TAP

Discussion: This subroutine performs the computation step, having size DBETA, and is the basic switchyard between the Runge Kutta integrator and the analytic coast phase solution. It also computes certain functions, in subroutine FUNCT, required by the program at each computation step.

STEP EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
GO	AX		Logical indicator for stepping forward: when true, program is stepping forward; when false, program is iterating for a remarkable point.
COAST	U	LOGIC4	Coast phase indicator.
DBETA	AX		Computation step size, $\Delta\beta$.

CHART TITLE - SUBROUTINE STEP(DBETA,GO)



ORIGINAL PAGE IS
OF POOR QUALITY

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL GO,COAST
COMMON /LOGIC4/ LO1(31),COAST,LO2(468)
```

ORIGINAL PAGE IS
OF POOR QUALITY

STEP-3

Name: STORE

PRECEDING PAGE BLANK NOT FILMED

Calling Argument: J for STORE;
None for STOREI

Referenced Sub-programs: None

Referenced Commons: EXTREM, INTGR4, LOGIC4, REAL8

Entry Points: STOREI

Referencing Sub-programs: CHECK for STORE and STOREI

Discussion: Subroutine STORE stores, in the arrays of COMMON EXTREM, the extremum and associated values of selected functions which are checked (monitored) by subroutine CHECK along the trajectory. These values are ultimately printed in the "Extremum Points of Selected Functions" page. The routine is called once per computation step, during which several extrema and/or special points along the trajectory may be isolated. Multiple entries (points at essentially the same time along the trajectory) are eliminated by consolidation after the array of extremum or special points is sorted in terms of ascending time.

Entry point STOREI zeroes-out the indicator-locations (the first locations) of the storage arrays; the second locations are the value-locations.

Messages and printouts: When the storage arrays are filled, the message is printed:

ARRAYS IN LABELLED COMMON BLOCK EXTREM FILLED.
STORAGE OF DATA IN EXTREM TERMINATED.

The comprehensive-monitoring indicator is turned off, and the routine is exited.

STORE EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
J	SUX		The number of isolated points associated with the current computation step.
CEPS (14, 20, 2)	SU	EXTREM	General array of isolated points containing values associated with the current computation step.

STORE EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
CHIX (2,100)	S	EXTREM	Storage array for solar-panel array angle, χ, in degrees.
DIST (2,100)	SU	EXTREM	Storage array for spacecraft solar distance, r, in AU.
NCEP	U	INTGR4	Maximum number of different parameters which are stored, to be printed.
PMAX	SU	REAL8	Maximum value of power ratio, $(\gamma q)_{max}$, encountered along the trajectory.
QJEX	S	LOGIC4	Indicator for comprehensive monitoring of trajectory functions.
RMAX	SU	REAL8	Maximum solar distance, r_{max}, encountered by the spacecraft along the trajectory, in AU.
RMIN	SU	REAL8	Minimum solar distance, r_{min}, encountered by the spacecraft along the trajectory, in AU.
TIME(100)	S	EXTREM	Storage array for time elapsed since start of trajectory, t, in days.
TRAV(100)	S	EXTREM	Storage array for ecliptic longitude, in degrees.
NSPEC	SU	INTGR4	Master array index (and counter) for storage arrays.
ONOFF (2,100)	S	EXTREM	Storage array for thrust switching function, σ.
POWEX (2,100)	SU	EXTREM	Storage array for power ratio, γq .
AKOUNT (100)	SU	EXTREM	Storage array for the number of iterations required to isolate the associated point.

STORE EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ANGCOM (2,100)	S	EXTREM	Storage array for communication angle, in degrees.
ANGPHI (2,100)	S	EXTREM	Storage array for thrust angle ϕ, in degrees.
ANGPSI (2,100)	S	EXTREM	Storage array for thrust angle ψ, in degrees.
ANGTHE (2,100)	S	EXTREM	Storage array for thrust angle θ, in degrees.
DISCOM (2,100)	S	EXTREM	Storage array for communication distance, in AU.

CHART TITLE - SUBROUTINE STORE(J)

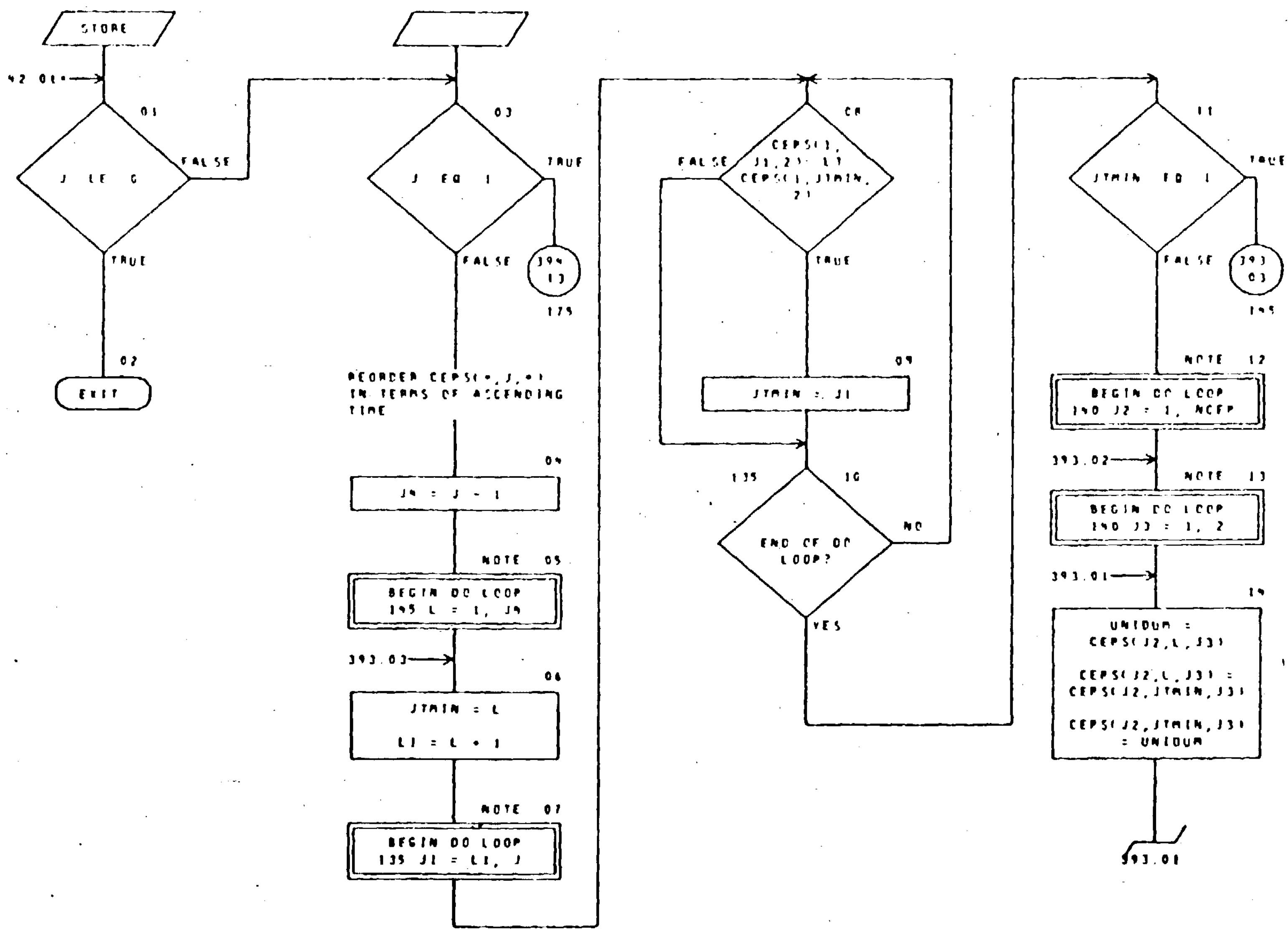


CHART TITLE - SUBROUTINE STORE(J)

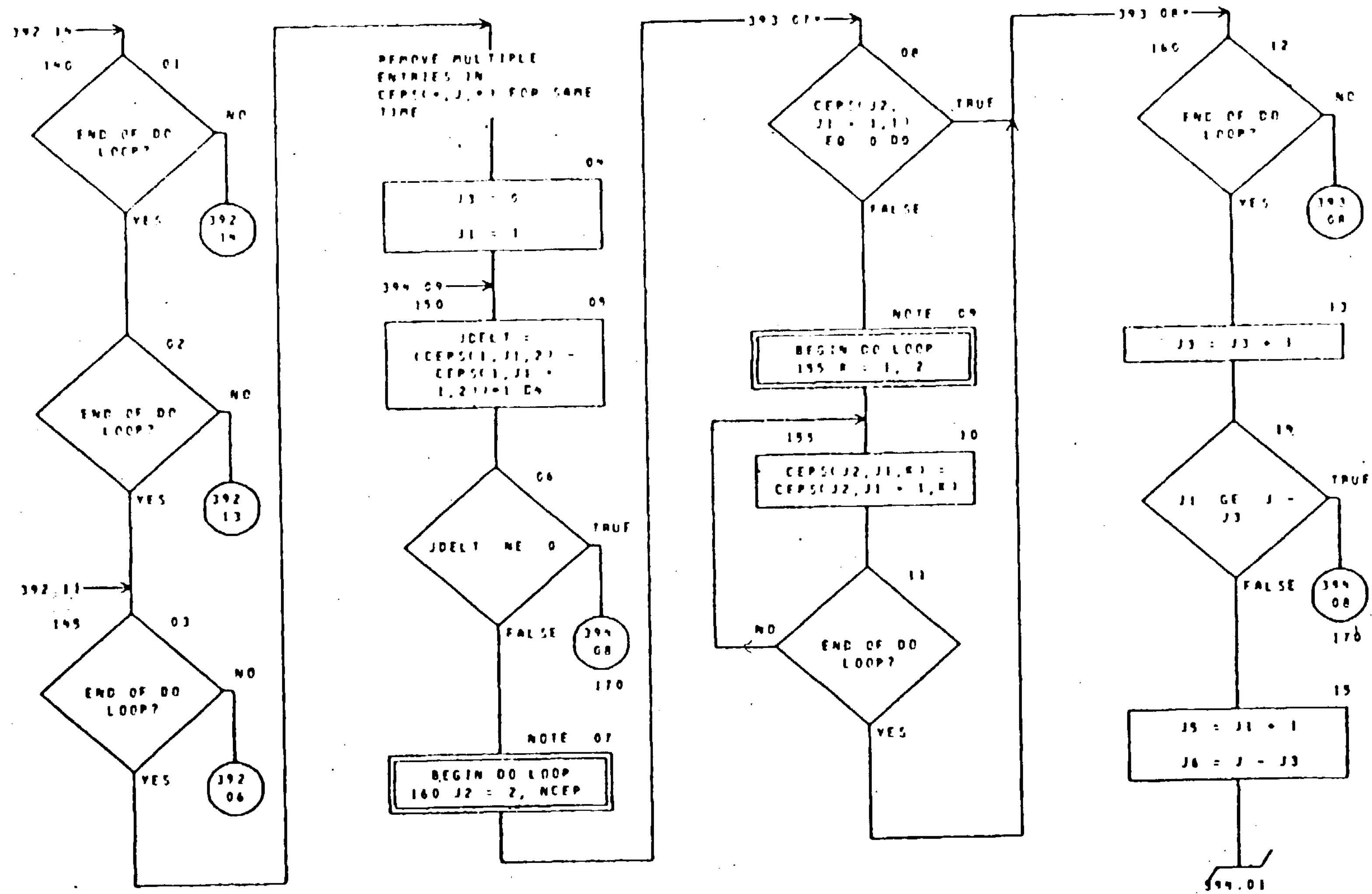


CHART TITLE - SUBROUTINE STORE(J)

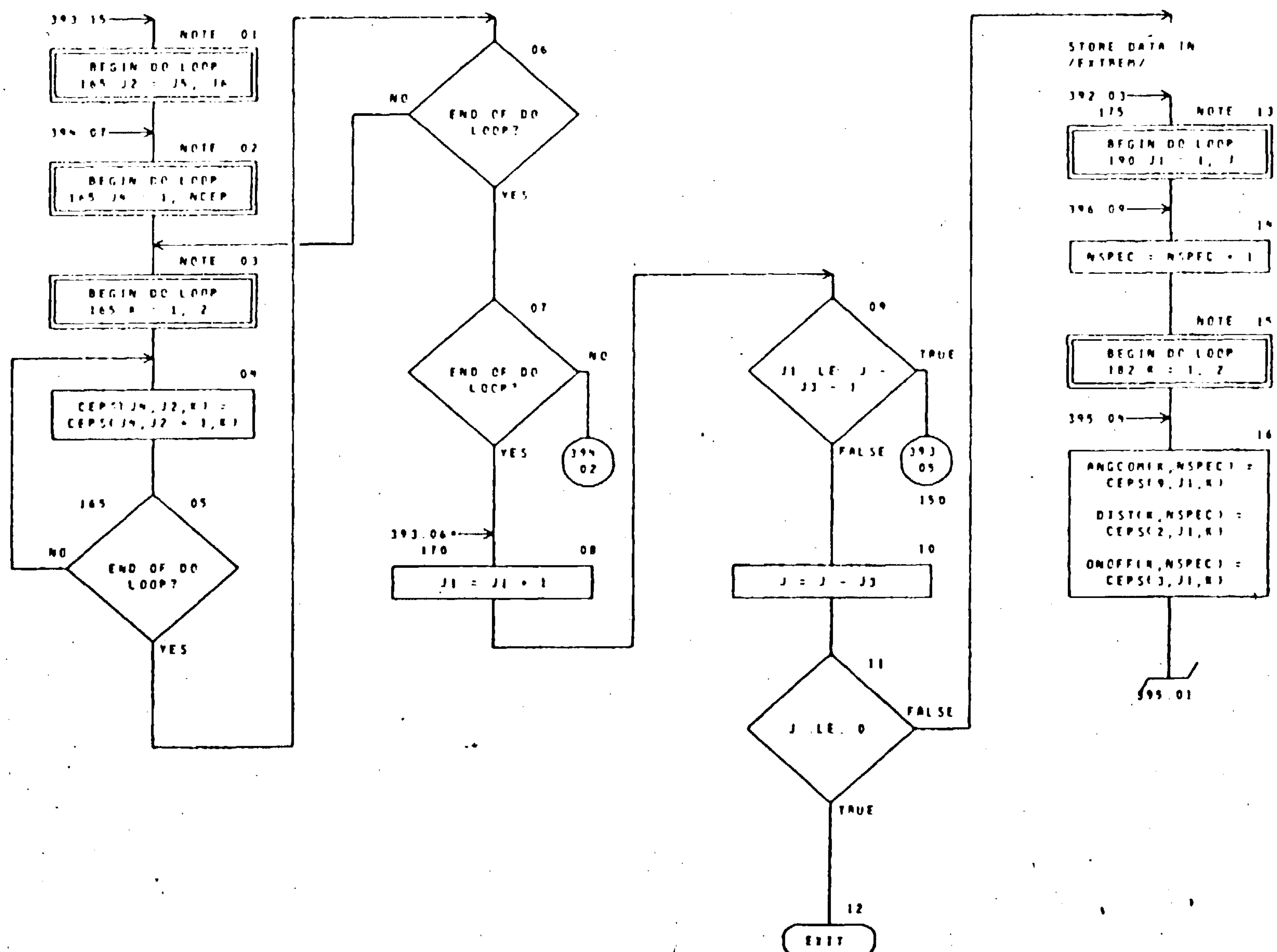


CHART TITLE - SUBROUTINE STORE(J)

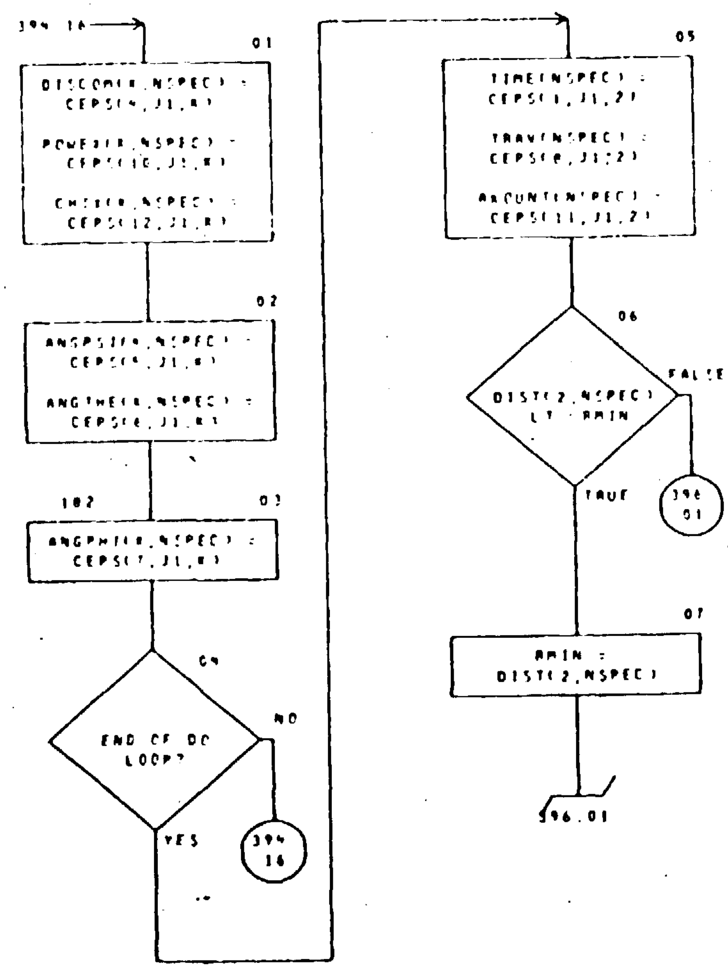


CHART TITLE - SUBROUTINE STORE(J)

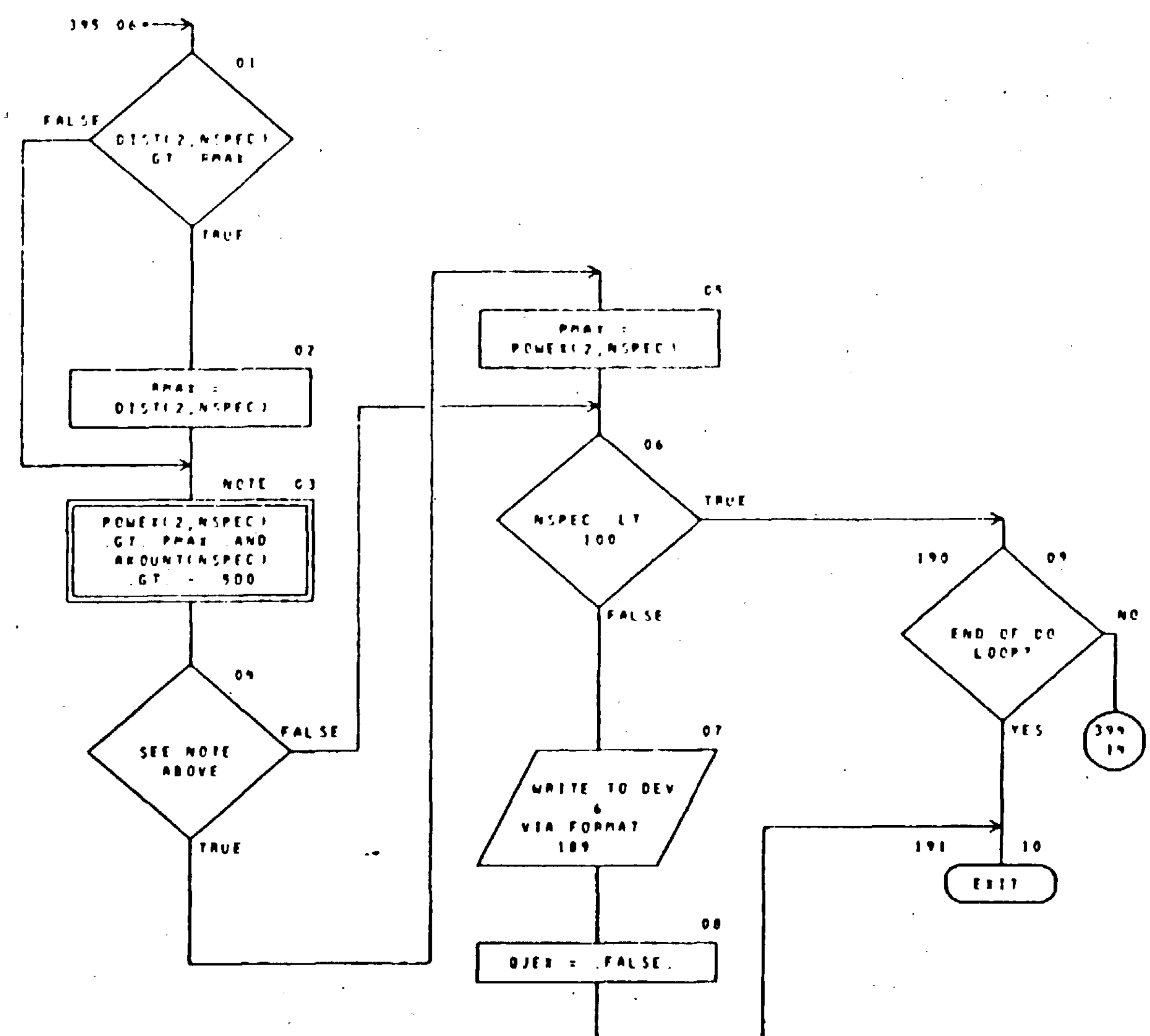
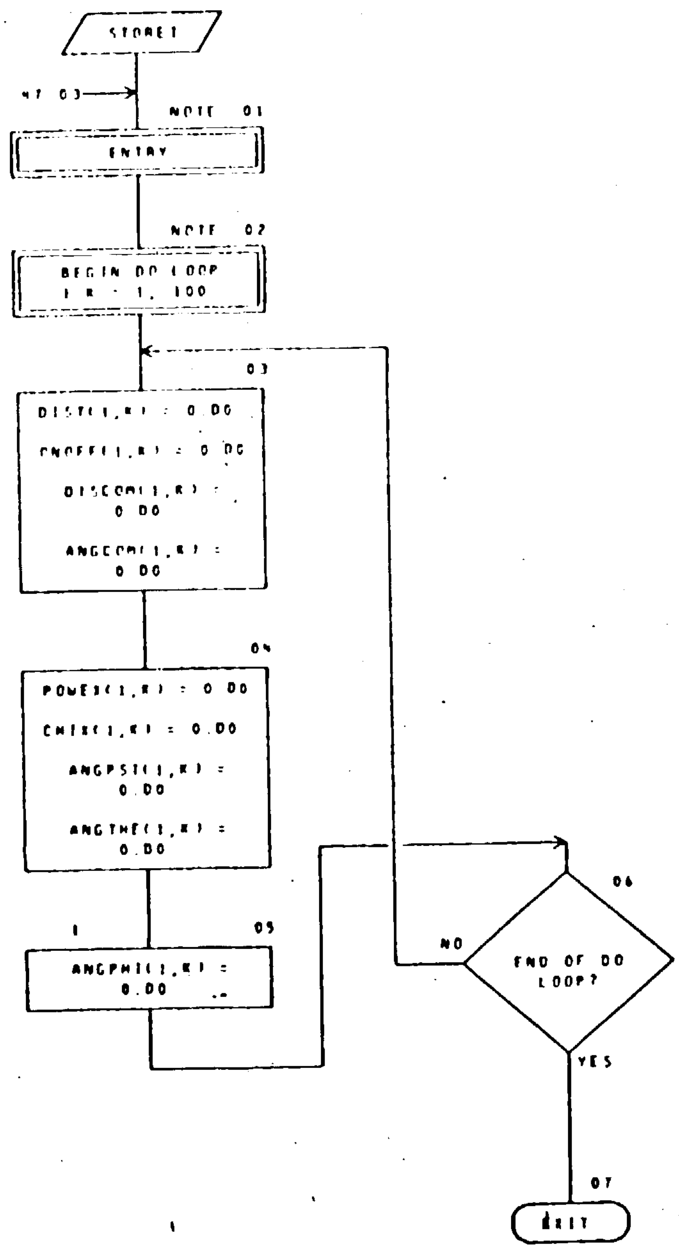


CHART TITLE - SUBROUTINE STORE(J)



STORE-9

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL OJEX
      COMMON /REAL8/ RO1(177),RMAX,RMIN,PMAX,RO2(1820)
      COMMON /INTGR4/ IO1(37),NSPEC,      IO2(290),NCEP,IO3(671)
      COMMON /LOGIC4/ LO1(26),OJEX,LO2(473)
      COMMON /EXTREM/ TIME(100),TRAV(100),DIST(2,100),ONOFF(2,100),
      DISCOM(2,100),ANGCOM(2,100),ANGPSI(2,100),ANGTHE(2,100),
      ANGPHI(2,100),POWEX(2,100),CHIX(2,100)             ,ACOUNT(100),
      CEPS(14,20,2),EOJ(60)
189   FORMAT (1H0,'ARRAYS IN LABELLED COMMON BLOCK EXTREM FILLED  STORAG
      E OF DATA IN EXTREM TERMINATED.')
```

<u>Name:</u> SUMMRY

<u>Calling Argument:</u> K, SPIT

<u>Referenced Sub-programs:</u> None

<u>Referenced Commons:</u> INTGR4, LOGIC4, REAL8

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> FINISH, TIKTOK

<u>Discussion:</u> This routine prints a single page of selected parameters at the termination of each computer run. Each case of the run corresponds to one line of output, the information for which is stored by a call to SUMMRY at the end of each case.

<u>Messages and printouts:</u> The single page of selected parameters, summarizing the computer run, are printed as follows:

RUN SUMMARY

CASE	NET MASS (KG)	REFERENCE POWER (KW)	TRAVEL ANGLE (DEG)	COMMUNICATION DISTANCE (AU)	COMMUNICATION ANGLE (DEG)	INITIAL MASS (KG)	...
____	____	____	____	____	____	____	...
____	____	____	____	____	____	____	...
____	____	____	____	____	____	____	...
.							
.							
.							

PROPEL MASS (KG)	DEPART EXCESS (M/S)	MAX POWER (KW)	BURN TIME (DAYS)	FLIGHT TIME (DAYS)	CALENDAR LAUNCH DATE	STATUS
____	____	____	____	____	____	____
____	____	____	____	____	____	____
____	____	____	____	____	____	____
.						
.						
.						

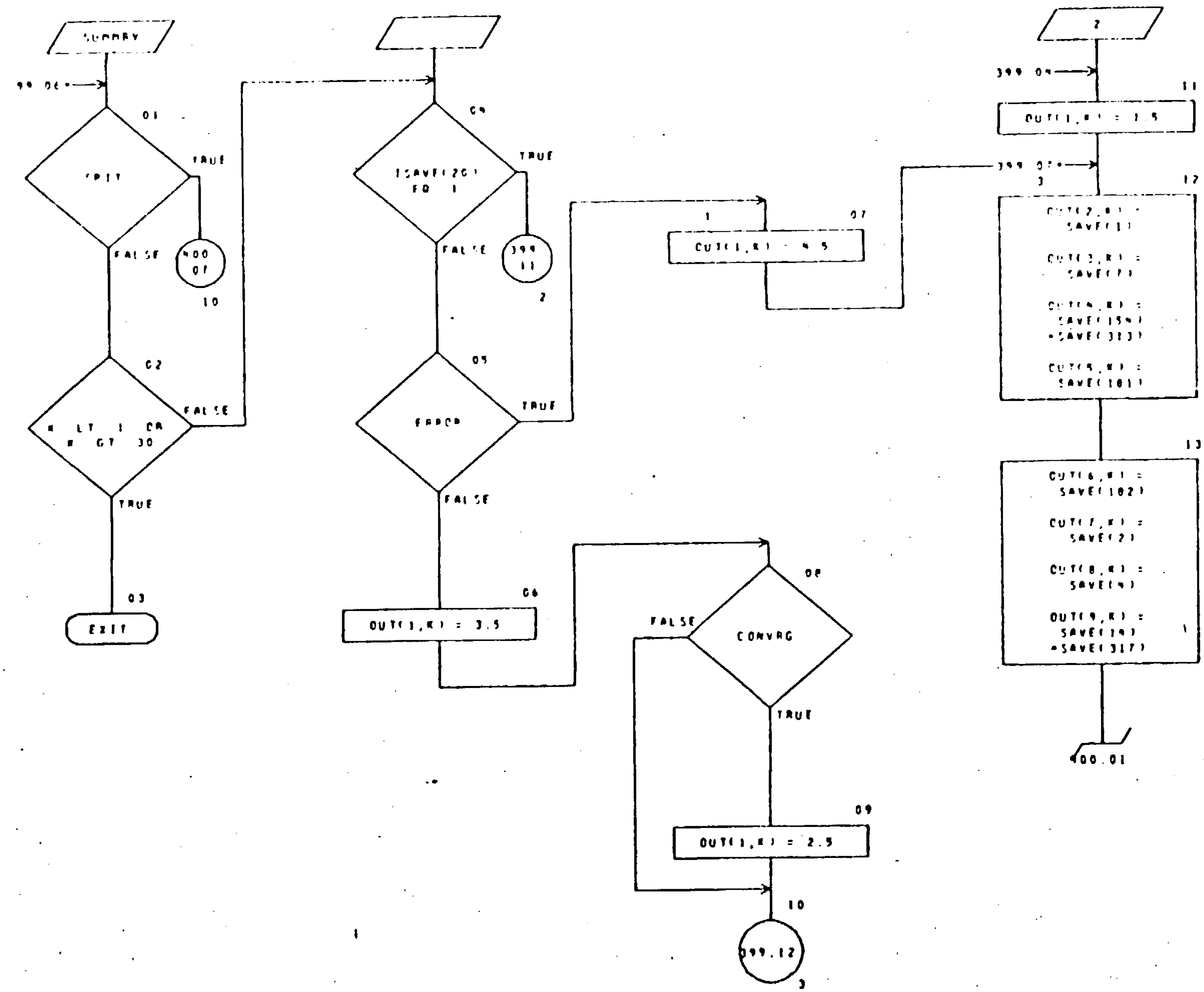
When more than 30 cases exist on a given computer run, the storage and printout of summary information is halted at the 30^{th} case, and the following line is printed:

SUMMARY GRID FILLED. SOME CASES ARE DELETED.

SUMMRY EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
K	SUX		Case-number of computer run.
SAVE (2000)	U	REAL8	Array of all COMMON/REAL8/variables.
SPIT	UX		Indicator for storage operation or summary printout operation.
ERROR	U	LOGIC4	Program master error indicator.
ISAVE (1000)	U	INTGR4	Array of all COMMON/INTGR4/variables.
CONVRG	U	LOGIC4	Iteration sequence convergence indicator.

CHART TITLE - SUBROUTINE SUMMRYER,SPIT



SUMMRY-3

CHART TITLE - SUBROUTINE SUMMRY(K,SPIT)

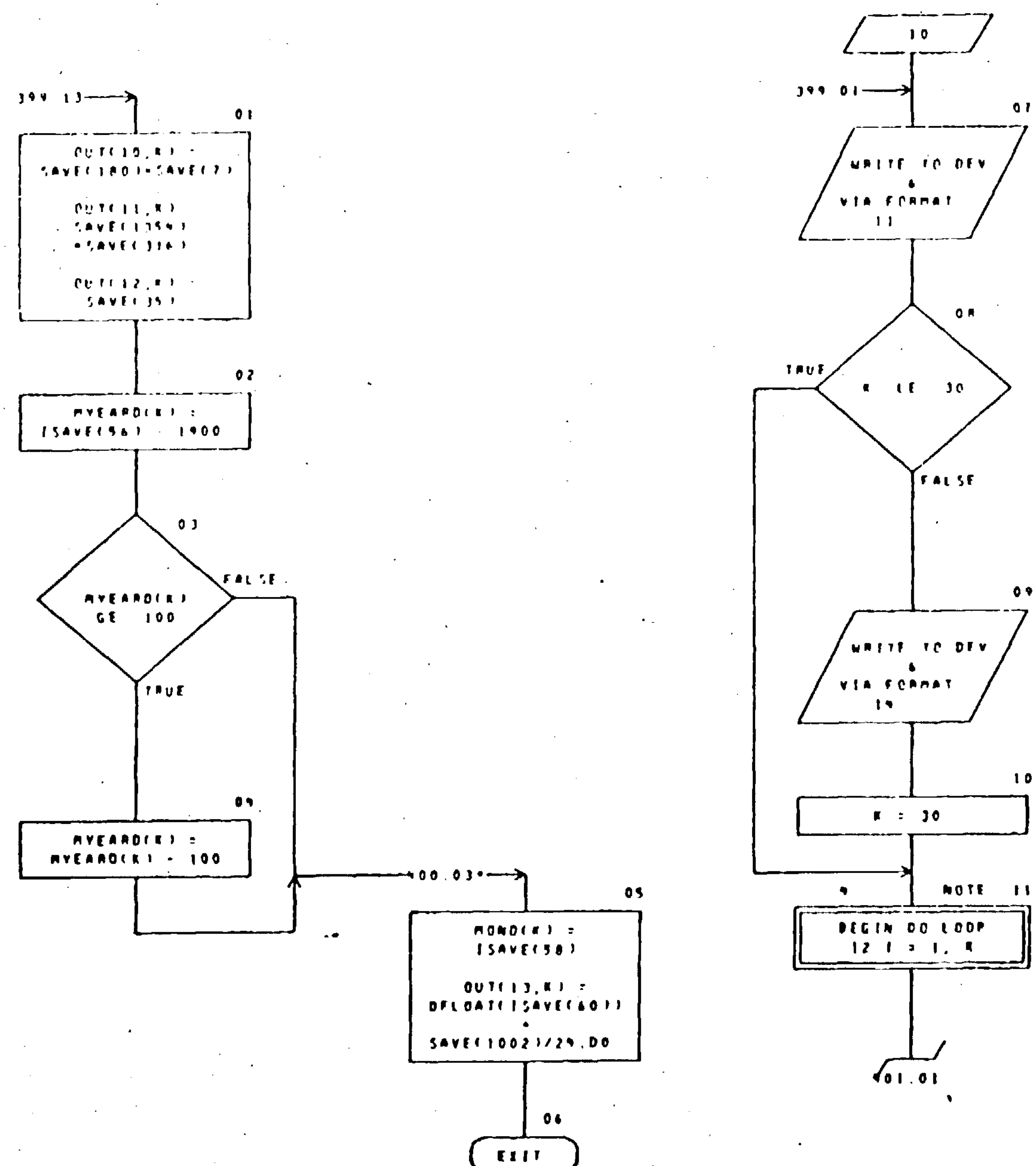
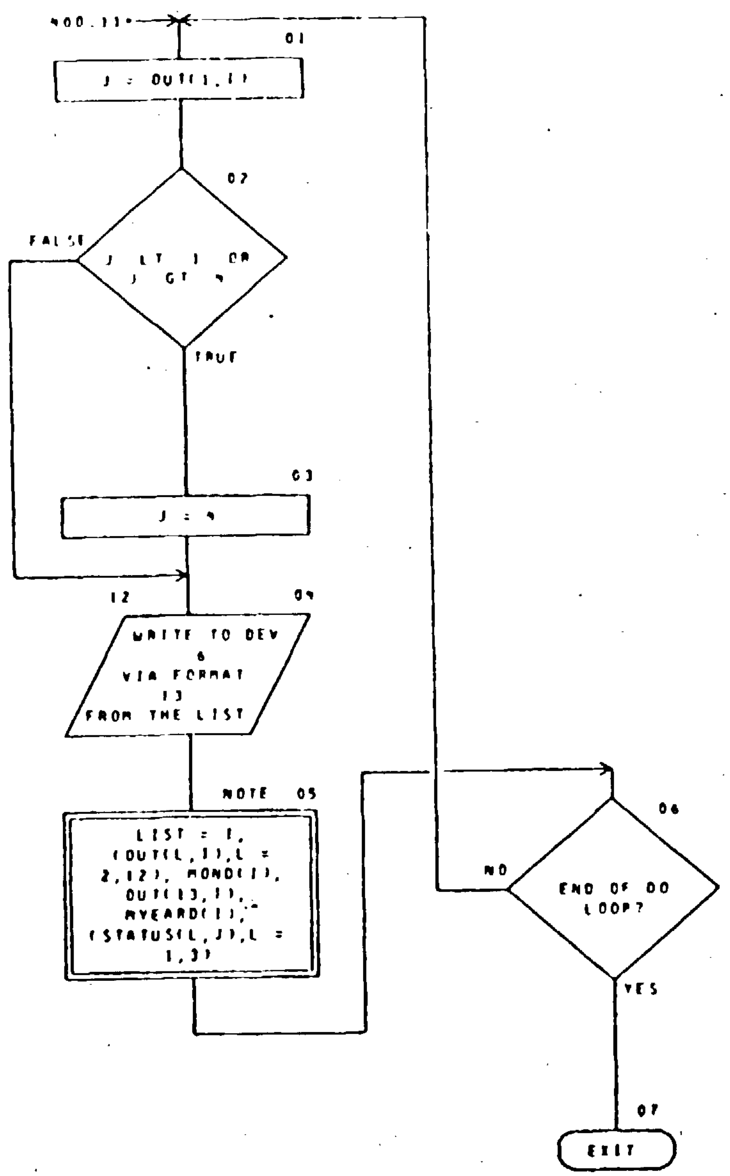


CHART TITLE - SUBROUTINE SUMMRY(K,SPIT)



SUMMRY-5

```
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*4 OUT
      LOGICAL SPIT,ERROR,CONVRG
      DIMENSION OUT(13,30),STATUS(3,4),MYEAR(30),MON(30)
      COMMON /REALR/ SAVE(2050)
      COMMON /INTGR/ ISAVE(1000)
      COMMON /LOGICL/ ERROR,CONVRG,LOGIC(98)
      DATA STATUS /8HSINGLE T,8HRAJECTOR,1HY,8HCONVERGE,1HD,1H ,
     8HNOT CONV,8HERGED,1H ,8HERROR,2*1H /
   11 FORMAT(1H ,57X,11HRUN SUMMARY//1H ,
     63X,1HNET  REFERENCE TRAVEL COMMUNICATION  INITIAL PROPEL,
     42H  DEPART    MAX    BURN   FLIGHT  CALENDAR/1H ,
     63X,0HMASS   POWER   ANGLE  DISTANCE ANGLE   MASS   MASS,
     42H   EXCESS   POWER   TIME    TIME    LAUNCH/1H ,
     54HCASE  (KG)   (KW)   (DEG)   (AU)   (DEG)  (KG)  (KG),
     59H    (M/S)    (KW)  (DAYS)  (DAYS)    DATE        STATUS/1H )
   13 FORMAT(1H ,I3,F7.1,F8.2,F8.1,F9.2,F7.1,F8.1,F7.1,F8.1,F8.2,2F8.1,
     I3,1H/F9.1,1H/I2,3X3A8)
   14 FORMAT(1H0,49HSUMMARY GRID FILLED  SOME CASES ARE DELETED )
```

ORIGINAL PAGE IS
OF POOR QUALITY

Name: SWING

Calling Argument: INDEX, NS, NT, TLEG2, GOOD, BURN, RPERI, XSTATE, YSTATE, A, NSWING

Referenced Sub-programs: AEINWT, CONVER, EFM, MINMX3, SWTRAJ, VCROSS, VDOT, VMAG, VSCAL, VSUB

Referenced Commons: INTGR4, ITERAT, ITER2, LOGIC4, REAL8, SOLSYS

Entry Points: None

Referencing Sub-programs: MORE

Discussion: Subroutine SWING performs the swingby-continuation analysis, invoked by the NAMELIST input vector MOPT4, whereby ballistic swingbys past the primary target may be simulated.

In one mode of program operation, invoked by MOPT4(1) > 0, single swingbys past the primary target may be simulated to up to ten post-swingby targets per case.

In another mode of program operation, invoked by MOPT4(1) < 0, multiple swingbys along a single trajectory may be simulated, first swinging past the primary target and then subsequently swinging past more targets downstream along the trajectory. One multiple swingby trajectory may be simulated per case.

In either mode of operation, the following basic assumptions are made. The swingby continuation computations are independent of the trajectory leg leading up to the swingby target, which may consist of an optimized electric propulsion trajectory segment (if the swingby planet is the primary target), except that the arrival V_∞ and arrival time at the swingby planet are used in the determination of the swingby passage conditions. Each swingby maneuver is calculated under the assumption of the patched-conic approximation, and the swingby planet's sphere-of-influence is assumed to have zero radius as seen from interplanetary space and infinite radius as seen from the planetary vantage point. The passage time in the swingby planet's sphere-of-influence is neglected (taken to be zero in the heliocentric frame).

Each swingby maneuver may be either unpowered or powered. Since the unpowered swingby solutions are embedded in the wider class of powered-swingby solutions, tending to appear in pairs which are separated by a region of braking powered swingbys, the more general case of powered swingbys is discussed first.

A powered swingby maneuver is restricted to occur at the mutual perifoci of the approach and departure hyperbolic arcs; the powered phase is impulsive and the thrust is colinear (pro or con) to the velocity at closest approach. Whether the swingby is powered or unpowered, the trajectory segment leading up to the swingby planet has been pre-determined, this being the method by which the subroutine has been designed to obtain swingby solutions. Therefore the swingby time and the arrival hyperbolic excess velocity $V_{\infty A}$ are known. Let subscript A pertain to arrival at the swingby planet and subscript D pertain to departure.

A basic assumption of the powered swingby problem posed here is that the flight time from the swingby planet to the next target is specified. This being so, the subroutine is able to converge, by iteration, on some ballistic trajectory from the swingby planet to the next target having the specified transfer time, implying that the departure hyperbolic excess velocity $V_{\infty D}$ at the swingby planet is thereby determined. Therefore, the heliocentric trajectory before and after the swingby planet is determined, and it then remains to perform the required computations pertaining to the hyperbolic arcs within the swingby planet's sphere of influence.

The closest approach distance is found by iteration as follows. Let

$$\alpha_A = 1 + \frac{r_p v_{\infty A}^2}{\mu},$$

and

$$\alpha_D = 1 + \frac{r_p v_{\infty D}^2}{\mu},$$

where $v_{\infty A} = |V_{\infty A}|$, $v_{\infty D} = |V_{\infty D}|$, r_p is the (unknown) passage distance, and μ is the swingby planet's gravitational parameter. Then the approach and departure hyperbolic bend angles are given by

$$\frac{\delta_A}{2} = \text{cosec}^{-1}\,\alpha_A = \sin^{-1}(1/\alpha_A),$$

$$\frac{\delta_D}{2} = \text{cosec}^{-1}\,\alpha_D = \sin^{-1}(1/\alpha_D),$$

and these must sum up to the total bend angle, which is specified in terms of $V_{\infty A}$ and $V_{\infty D}$:

$$\delta_T = \frac{\delta_A}{2} + \frac{\delta_D}{2} = \cos^{-1}\left[\frac{V_{\infty A} \cdot V_{\infty D}}{v_{\infty A}\ v_{\infty D}}\right].$$

Therefore, using r_p as the independent variable, the zero of the quantity

$$F = \sin^{-1}(1/\alpha_A) + \sin^{-1}(1/\alpha_D) - \cos^{-1}\left[\frac{V_{\infty A} \cdot V_{\infty D}}{v_{\infty A}\ v_{\infty D}}\right]$$

is obtained by Newton's iteration, using the derivative,

$$\frac{\partial F}{\partial r_p} = \left(\frac{-1}{\mu}\right)\left[\frac{v_{\infty A}^2/\alpha_A}{\sqrt{\alpha_A^2 - 1}} + \frac{v_{\infty D}^2/\alpha_D}{\sqrt{\alpha_D^2 - 1}}\right].$$

When the iteration is converged, the passage distance r_p is in hand, and the impulsive velocity increment is computed,

$$\Delta v = \sqrt{\frac{2\mu}{r_p} + v_{\infty D}^2} - \sqrt{\frac{2\mu}{r_p} + v_{\infty A}^2},$$

where the square-root-quantities are the hyperbolic speeds at closest approach. The remaining parameters defining the planetocentric transfer are computed as follows. The inclination of the swingby orbit plane to the planet's equator is given by

$$i = \cos^{-1}(\bar{h} \cdot \bar{n}_p),$$

where $\bar{h}$ is the unit vector along the angular momentum of the hyperbolic passage trajectory and $\bar{n}_p$ is a unit vector pointing toward the swingby planet's north pole. The ascending node angle of the swingby orbit plane is computed as,

$$\Omega = \tan^{-1} (-h_x/h_y),$$

and is placed in the proper quadrant by using the system library routine DATAN2. The argument of perifocus is given by,

$$\omega = \cos^{-1} (\bar{r}_p \cdot \bar{r}_n),$$

where $\bar{r}_p$ is the unit vector pointing toward the closest approach point and $\bar{r}_n$ is the unit vector lying along the line of nodes and pointing toward the ascending node. This is adjusted for the proper quadrant by the test,

$$\text{If } h_z (\bar{r}_n \times \bar{r}_p)_z < 0, \quad \omega \rightarrow 2\pi - \omega.$$

In the right-handed planetary reference frame, the z-axis is toward the planet's north and the x-axis points toward the ascending node of the planet's equator on the ecliptic.

An unpowered swingby maneuver is considered to be a powered swingby having $\Delta v = 0$. The subroutine adjusts the post-swingby heliocentric trajectory segment, by iteration, until the swingby departure V_∞ magnitude equals the given arrival V_∞ magnitude. The primary independent variable in this iteration is the post-swingby transfer time to the specified target, which was held constant in the powered swingby case. Thus $v_{\infty D} = v_{\infty A} = v_\infty$, and the swingby passage distance is obtained from the formula,

$$r_p = \frac{\mu}{v_\infty^2} \left(\frac{2v_\infty}{|V_{\infty A} - V_{\infty D}|} - 1 \right).$$

The other orbital parameters are obtained from the same relations given above.

The subroutine allows the generation of multiple-revolution ballistic arcs, and a particular solution obtained by the subroutine may not be unique, even for the same transfer time. All solutions are reachable, however, by means of inputting an appropriate initial velocity guess for the trajectory segment in question.

Messages and printouts: The normal, informative printout from this subroutine is output on unit 6 as follows:

(name 1) SWINGBY CONTINUATION TO (name 2)

where name 1 is the name of the swingby planet and name 2 is the name of the post-swingby target;

PASS DIST (RADII)	SPEED (M/SEC)	INCLIN (DEG)	NODE(DEG)	ARGPER (DEG) .....
(r_p)	(v_p)	(i)	(Ω)	(ω)

.....LEG TIME (DAYS)	MISSION TIME (DAYS)	ARR VINF (M/SEC)
(Δt)	(T)	(v_∞)

where r_p is the swingby passage distance in units of planet-radii (of the swingby planet), v_p is the planetocentric speed of the spacecraft at closest approach, in meters/second, i is the inclination of the swingby passage hyperbolic orbit to the planet's equatorial plane, in degrees, Ω and ω are the ascending node and argument of perifocus, in degrees, as described in the discussion, Δt is the time of flight from swingby planet to post-swingby target, in days, T is the total time elapsed since the start of the trajectory (i.e., launch) up to intercept of the post-swingby target, in days, and v_∞ is the hyperbolic excess speed at arrival of the post-swingby target, in meters/second.

The arrival and departure hyperbolic excess velocity vectors at the swingby planet, in ecliptic coordinates, (v_x, v_y, v_z), are then printed twice (including their magnitudes), the first time in EMOS and the second time normalized to unity:

ARRIVAL   V00 ____ ____ ____ MAG= ____ (ECLIPTIC REFERENCE SYSTEM)
DEPARTURE V00 ____ ____ ____ MAG= ____ (ECLIPTIC REFERENCE SYSTEM)

ARRIVAL   V00 ____ ____ ____ MAG= 1.0 (ECLIPTIC REFERENCE SYSTEM)
DEPARTURE V00 ____ ____ ____ MAG= 1.0 (ECLIPTIC REFERENCE SYSTEM)

Other parameters associated with the swingby maneuver are printed as follows:

HELIOCENTRIC APPROACH ANGLE=___, DEPART ANGLE=___, BEND ANGLE=____DEGREES
SWINGBY INCLINATION W.R.T. ECLIPTIC=___DEGREES

POWERED SWINGBY INCREMENTAL SPEED=___METERS/SECOND. .....
.....BEND ANGLE=___DEGREES (PLANETOCENTRIC)

where the heliocentric approach and departure angles are the angles between the swingby planet's heliocentric radius vector and the (planetocentric) approach and departure hyperbolic excess velocity vectors, respectively, in degrees, the bend angle is δ_T as described in the discussion, in degrees, the swingby inclination w.r.t. the ecliptic is the angle between the swingby orbit plane and the ecliptic, in degrees, and the powered-swingby incremental speed is given by Δv in the discussion, in meters/second.

When powered swingbys are being simulated, the above output will be preceded either by

POWERED SWINGBY ANALYSIS ONLY, FOR FIXED SWINGBY LEG FLIGHT TIME....
.... (Δt) DAYS

or by

POWERED SWINGBY ANALYSIS, FOR FIXED SWINGBY LEG FLIGHT TIME (Δt) DAYS
AND USING LAST VELOCITY GUESS FROM ABOVE ITERATION (v_x) (v_y) (v_z)

depending on the chosen method of analysis; Δt is the post-swingby transfer time, in days, and (v_x, v_y, v_z) consist of the heliocentric departure (from the swingby planet) velocity, in EMOS and in ecliptic coordinates. The latter heading will appear if the analyst first directs the program to seek the unpowered solution, as a prelude to finding the desired powered solution.

C-8

Except for summary output on unit 12, the remainder of the output consists of diagnostic and error messages pertaining to various failure modes of the subroutine printed on unit 6, the failures for the most part consisting of iterations which fail to converge. The unit 12 output consists of

INDEX = (i)
DELTA V = (Δv) M/S

where i is the so-called "index", running from 1 to a maximum of 10, which is essentially the i corresponding to program input vector MOPT4(i), and Δv is the (powered) swingby incremental speed, in meters/second.

Depending upon the mode of analysis chosen by the analyst via program input, the following may be printed when an iteration for an unpowered swingby trajectory fails:

ERROR VECTOR (q_1) (q_2) (q_3) (q_4) TIME = (Δt) DAYS.

where (q_1, q_2, q_3, q_4) are the iterator dependent-variable values (at the end of the iteration sequence), as described in the discussion of subroutine SWTRAJ, and Δt is iterator independent-variable value b_4 (see SWTRAJ), which is the transfer time to the post-swingby target, in days. This message is followed by

UNPOWERED SWINGBY FAILS. POWERED SWINGBY ANALYSIS FOLLOWS, FOR FIXED SWINGBY LEG FLIGHT TIME (Δt) DAYS.

where Δt is the same as above. When the powered swingby iteration fails, the following is printed:

ERROR VECTOR (q_1) (q_2) (q_3)
SWINGBY ITERATION FAILS. ERRORX = (e), CONVGE = (c), INDEX = (i)

where (q_1, q_2, q_3) are the same as above, e and c are condition indicators output by the iterator, and i is the index described previously.

Whenever the iteration for the powered-swingby passage-distance fails, the following is printed:

ITERATION FOR POWERED SWINGBY PASSAGE RADIUS DID NOT CONVERGE.
RP = $\underline{(r_p)}$ F = $\underline{f}$ FP = $\underline{(f')}$ INDEX = $\underline{(i)}$
POST-SWINGBY TARGET = <u>(name)</u>

where r_p is the final value of passage distance attained, in planet radii, f and f' are F and $\partial F/\partial r_p$ as given in the discussion, i is the index defined above, and "name" is self-explanatory. If the above iteration is "almost" converged, the message is printed:

NEARLY CONVERGED

and execution continues, otherwise the routine is exited after normal printout.

It is possible that all iterations might converge, yet the passage distance may be less than one, corresponding to (in reality) an impact of the surface of the swingby planet. The value of the passage distance is compared to unity, and, if less, the routine is exited after printing

SWINGBY TO <u>(name)</u> PASSAGE DISTANCE NEGATIVE
RP = $\underline{(r_p)}$ INDEX= $\underline{(i)}$

where "name" is the name of the post-swingby target, r_p is the passage distance, in planet radii, and i is the index defined previously.

When requesting that the analyst's initial guess for heliocentric velocity be employed to start the trajectory segment from swingby planet to post-swingby target, it is possible that the analyst will either then forget to input the velocity or input the velocity in the wrong locations, such that the default value of zero velocity would be attempted, corresponding to a trajectory which falls into the solar singularity (except for computational noise) at r = 0. The program checks for zero initial velocity, and, if detected, stops execution after printing on units 6 and 12:

FATAL ERROR. SWINGBY INITIAL VELOCITY GUESS = 0

SWING EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
A(6)	SUAX		Array containing position and velocity of swingby leg target body at time of intercept, in AU, AU/tau.
B(35)	SUA	ITER2	Array of (active) iterator independent variables.
Q(35)	U	ITER2	Array of (active) iterator dependent-variables.
X(50)	SUA	REAL8	Array of trajectory integrated variables.
BS(35)	S	ITER2	Array of maximum step-sizes for the independent variables.
BW(35)	S	ITER2	Array of independent-variable weighting factors.
GM(70)	U	SOLSYS	Array of planetary gravitational constants, in m^3/sec^2.
NS	SUX		Identification number of swingby planet.
NT	SUX		Identification number of target planet.
OO(70)	U	ITERAT	Array of iterator independent-variables, in external units.
APL(2, 70)	U	SOLSYS	Array of planet names.
BBB(35)	S	ITER2	Array of independent-variable perturbation step sizes.
DEG	U	REAL8	Conversion factor between radians and degrees; number of degrees in one radian.
OOO	SU	REAL8	Hyperbolic excess speed at arrival of swingby planet, in AU/tau.

SWING EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
BURN	SX		Logical flag indicating whether swingby is powered or unpowered.
CNIX(5)	SUA	REAL8	Inclinations to ecliptic of intermediate-target orbits, in degrees.
ECIX(5)	SUA	REAL8	Eccentricities of intermediate target orbits.
GOOD	SX		Swingby continuation trajectory iteration convergence indicator.
OMIX(5)	SUA	REAL8	Ascending node angles of intermediate-target orbits, in degrees.
QJEX	SU	LOGIC4	Detailed printout indicator.
QMAX(35)	SU	ITER2	Array of upper allowable values for the iterator dependent-variables.
QMIN(35)	S	ITER2	Array of lower allowable values for the iterator dependent-variables.
QUIT	S	LOGIC4	Logical variable, set in subroutine SWING, which causes bypass of last trajectory printout if time-out occurs during swingby continuation analysis.
SAIX(5)	SUA	REAL8	Semi-major axes of intermediate-target orbits, in AU.
SOIX(5)	SUA	REAL8	Arguments of perihelion of intermediate-target orbits, in degrees.
TPIX(5)	SUA	REAL8	Times from reference date to perihelion passages, for the intermediate targets, in days.
TSUM	U	REAL8	Time elapsed since primary-target swingby and current swingby, in days.
CONSP	U	REAL8	Speed conversion factor, from AU/tau to meters/second.

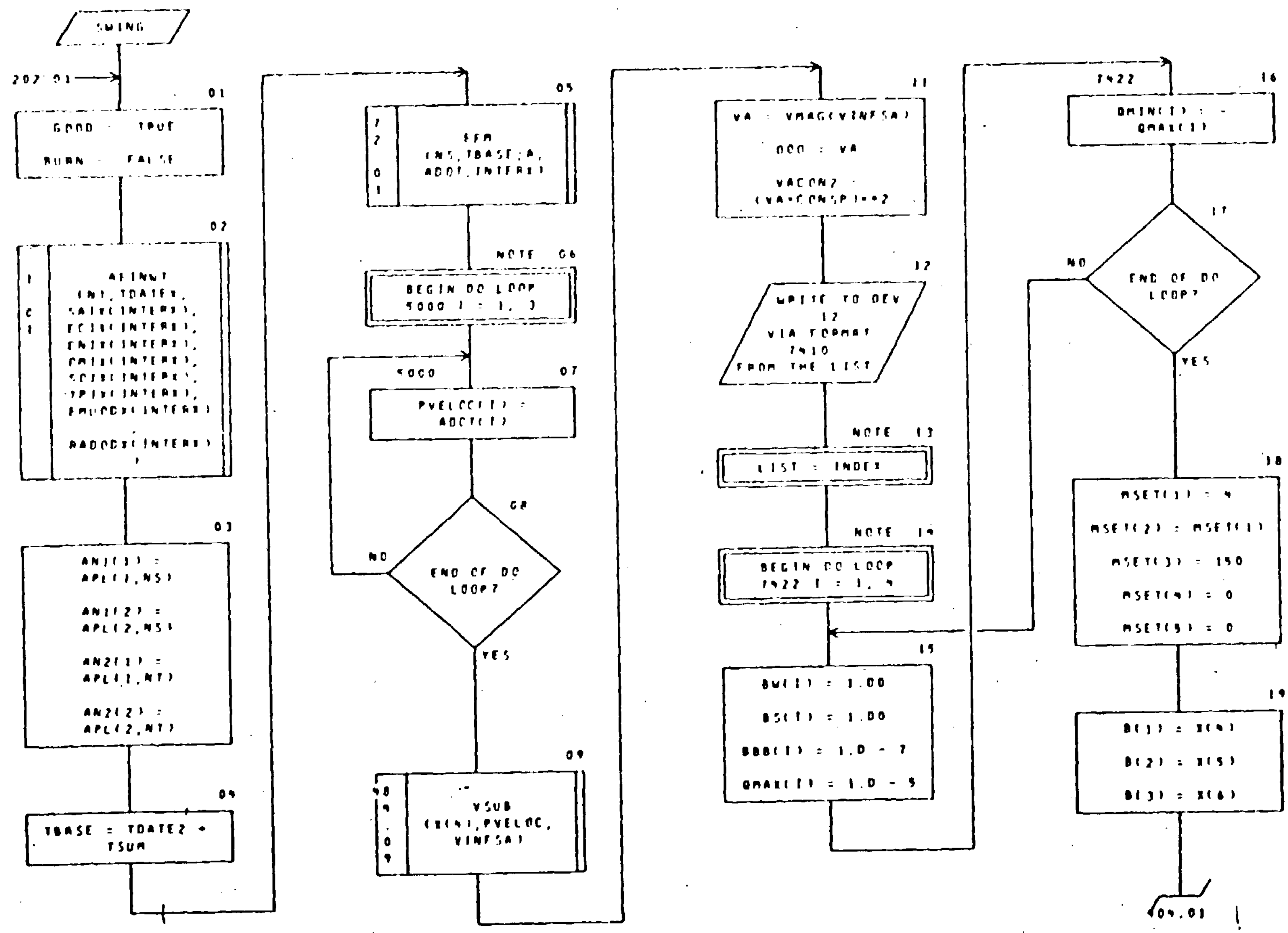
SWING EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
CONTM	U	REAL8	Time conversion factor, tau to days.
INDEX	UX		Index of swingby continuation trajectories requested for the current case.
NTARG	S	INTGR4	Identification number of the swingby leg target body.
RPERI	SUX		Periapse distance of planetocentric swing-by trajectory, in meters.
TBASE	SUA	REAL8	Julian date at the current swingby planet (less 2400000).
TLEG2	SUX		Flight time of the current swingby leg, in days.
TRACK	SU	LOGIC4	Indicator for trajectory long block print-out (at each computation step).
EMUODD	U	REAL8	Gravitational constant of the primary target, in m^3/sec^2.
EMUODX(5)	SUA	REAL8	Gravitational constants of intermediate targets, in m^3/sec^2 (not used at present).
INTERX	SUA	INTGR4	Index used in subroutine EFM which indicates which array locations are applicable in selecting orbital elements.
NLEAVE	S	INTGR4	Identification number of the current swingby planet.
NSWING	UX		Integer flag defining the type of swingby mission. See Inputs for the various options.
PVELOC(3)	SUA	REAL8	Velocity of the swingby planet at the time of encounter, in AU/tau.
RADIUS(70)	U	SOLSYS	Array of planetary-body radii, in meters.

SWING EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
RADODD	U	REAL8	Radius of primary target, in meters.
RADODX (5)	SUA	REAL8	Radii of intermediate targets, in meters (not used at present).
TANDEM	U	LOGIC4	Indicator for multiple swingbys along a single trajectory.
TDATEX	SUA	REAL8	Reference Julian date (minus 2400000).
TDATE2	U	REAL8	Julian date at the primary target, in days (less 2400000).
XSTATE (7)	SUXA		Spacecraft position and velocity vectors and time at departure of the swingby planet, in AU, AU/tau and tau, respectively.
YSTATE (7)	SUX		Spacecraft position, velocity and time at the end of the trajectory segment, in AU, AU/tau and tau, respectively.
YSWING (3,10)	U	REAL8	Array of velocity vectors consisting of initial velocity guesses of a given post-swingby trajectory segment, in AU/tau.
ZSTATE (7)	SUA	REAL8	Same as YSTATE.

CHART TITLE - SUBROUTINE SWING(INDEX,NS,NT,TIEG2,GOOD,BURN,RPERI,XSTATE,YSTATE,



ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - SUBROUTINE SWING(INDEX,NS,NT,TLEG2,GOOD,BURN,RPERI,XSTATE,VSTATE,

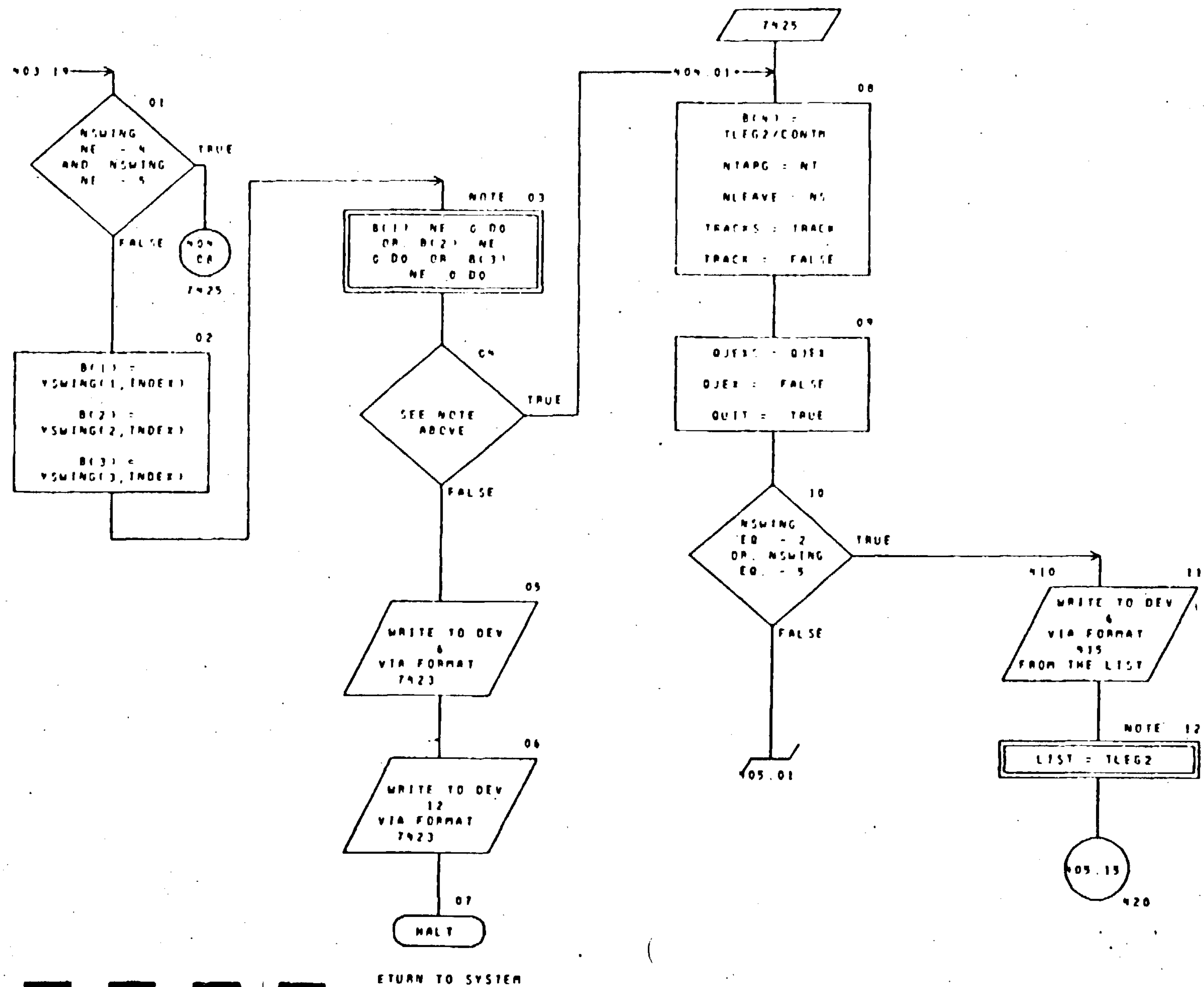
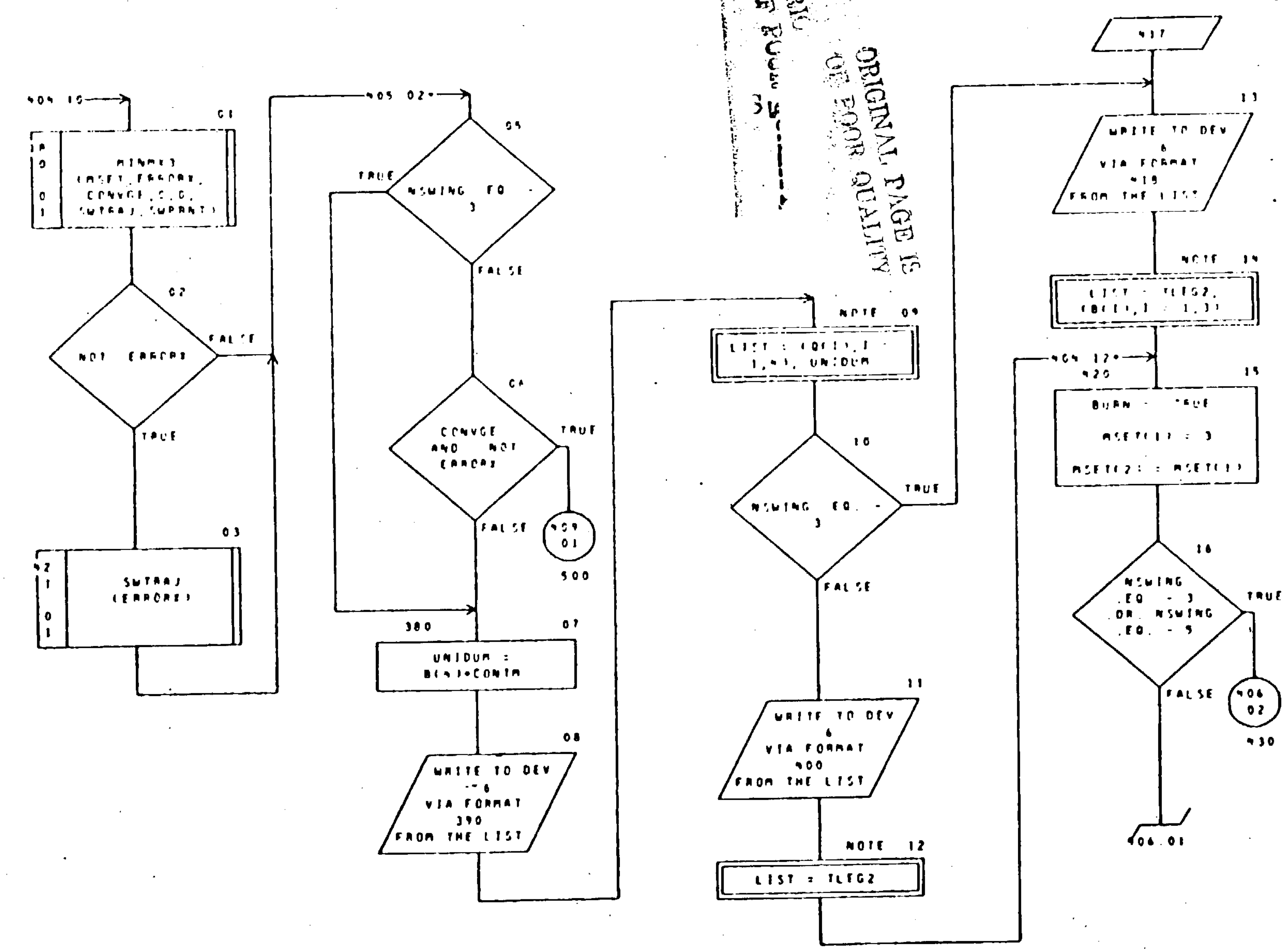


CHART TITLE - SUBROUTINE SWING(INDEX,NS,NT,TLEG2,GOOD,BURN,RPERI,XSTATE,YSTATE,



SWING-15

CHART TITLE - SUBROUTINE SWING(INDEX,NS,NT,TLEG2,GOOD,BURN,RPERI,XSTATE,YSTATE,

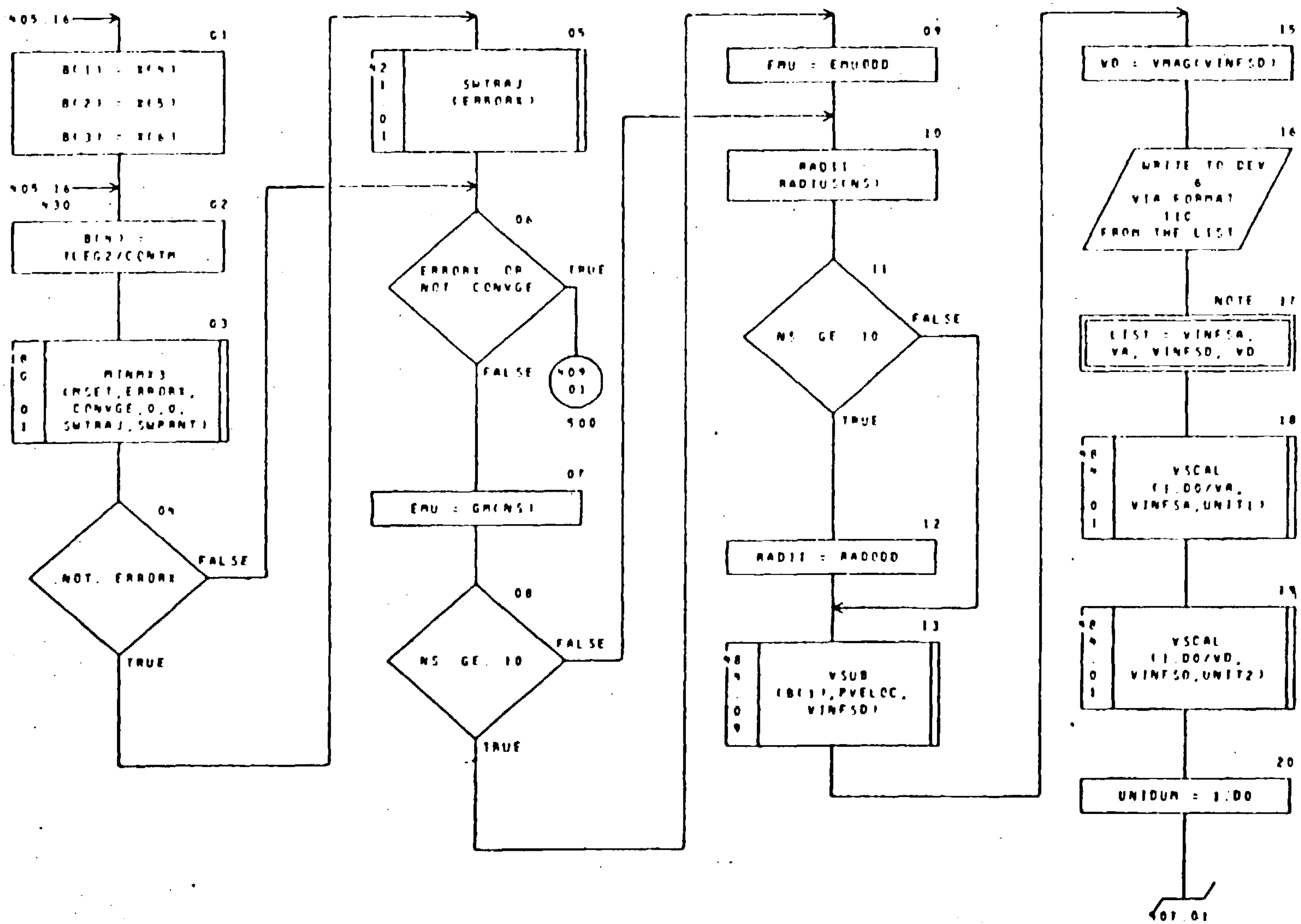
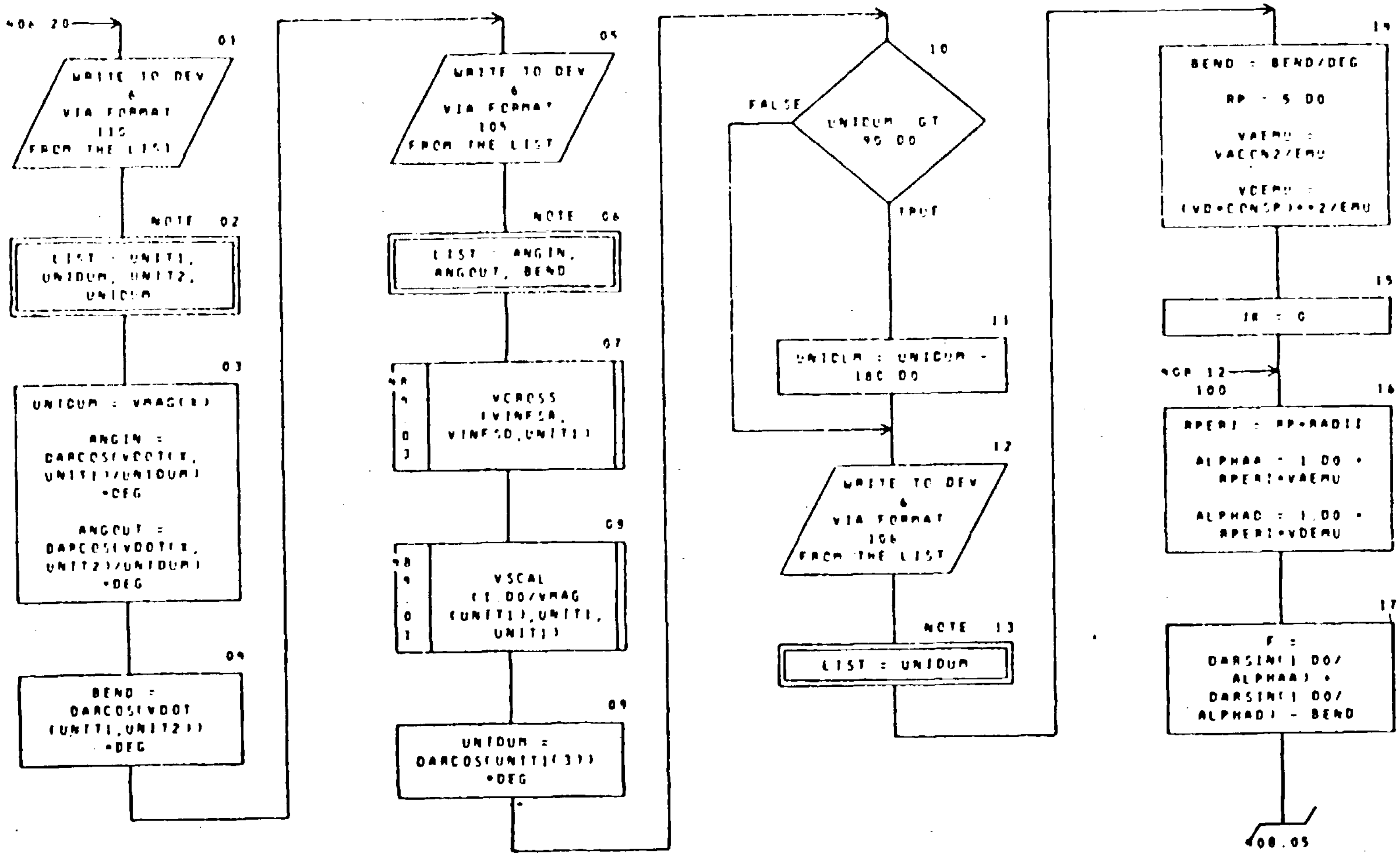


CHART TITLE - SUBROUTINE SWING(INDEX,NS,NT,TLEG2,GOOD,BURN,RPERI,XSTATE,YSTATE,



SWING-17

CHART TITLE - SUBROUTINE SWING(INDEX,NS,NT,TLEG2,GOOD,BURN,RPERI,XSTATE,YSTATE,

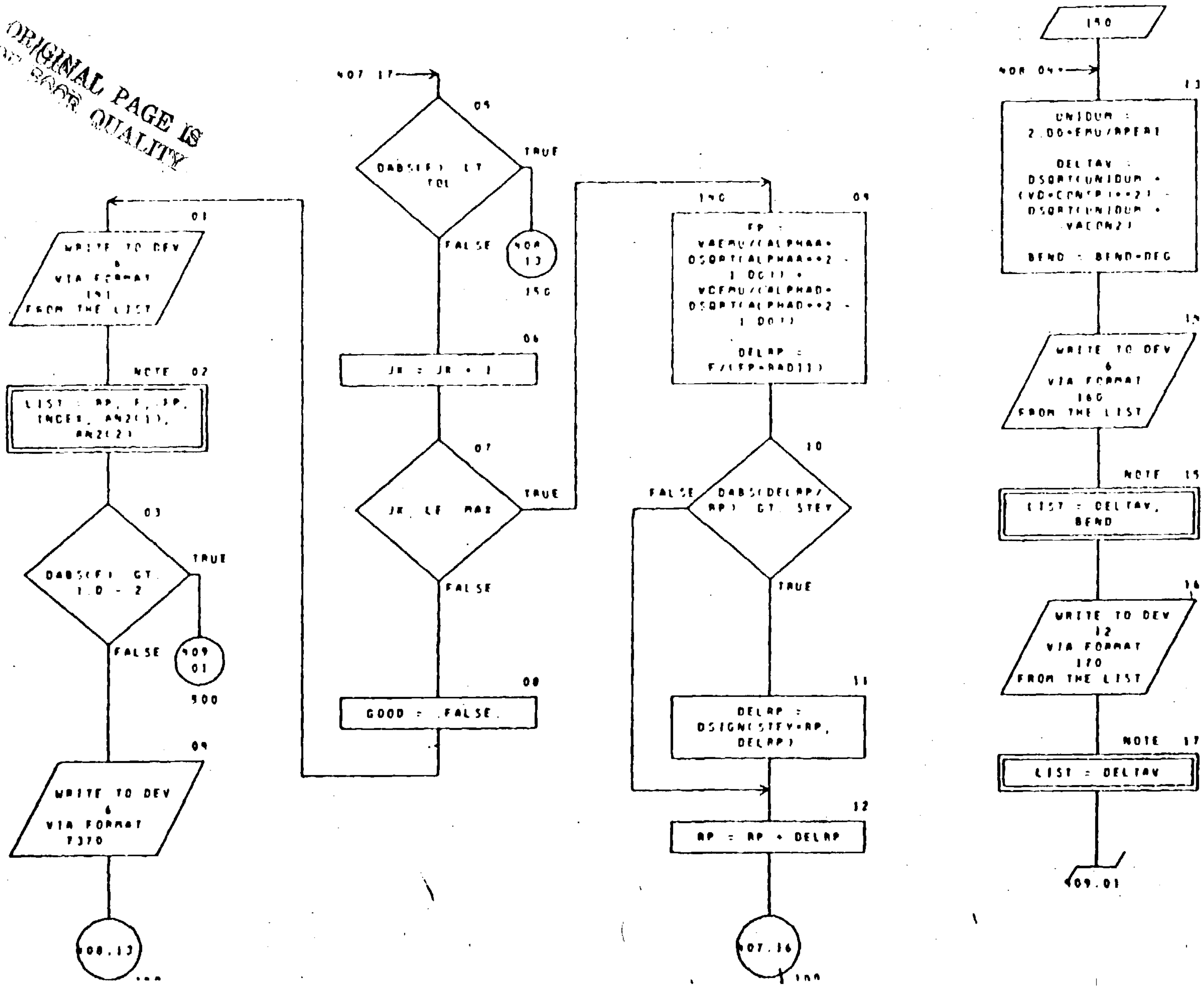


CHART TITLE - SUBROUTINE SWING(INDEX,NS,NT,TLEG2,GOOD,BURN,RPERI,XSTATE,YSTATE,

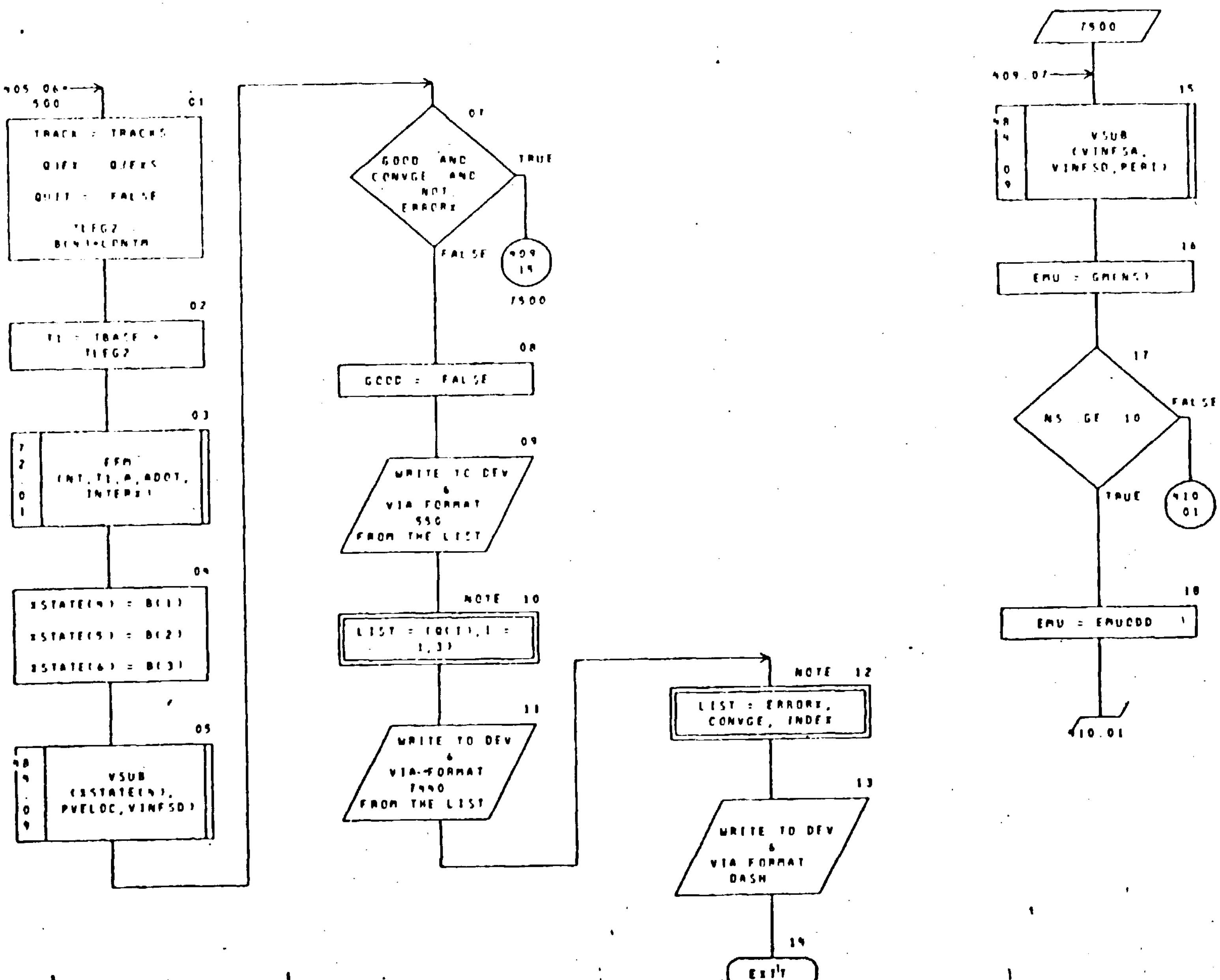


CHART TITLE - SUBROUTINE SWING(INDEX,NS,NT,TLEG2,GOOD,BURN,RPERI,XSTATE,YSTATE,

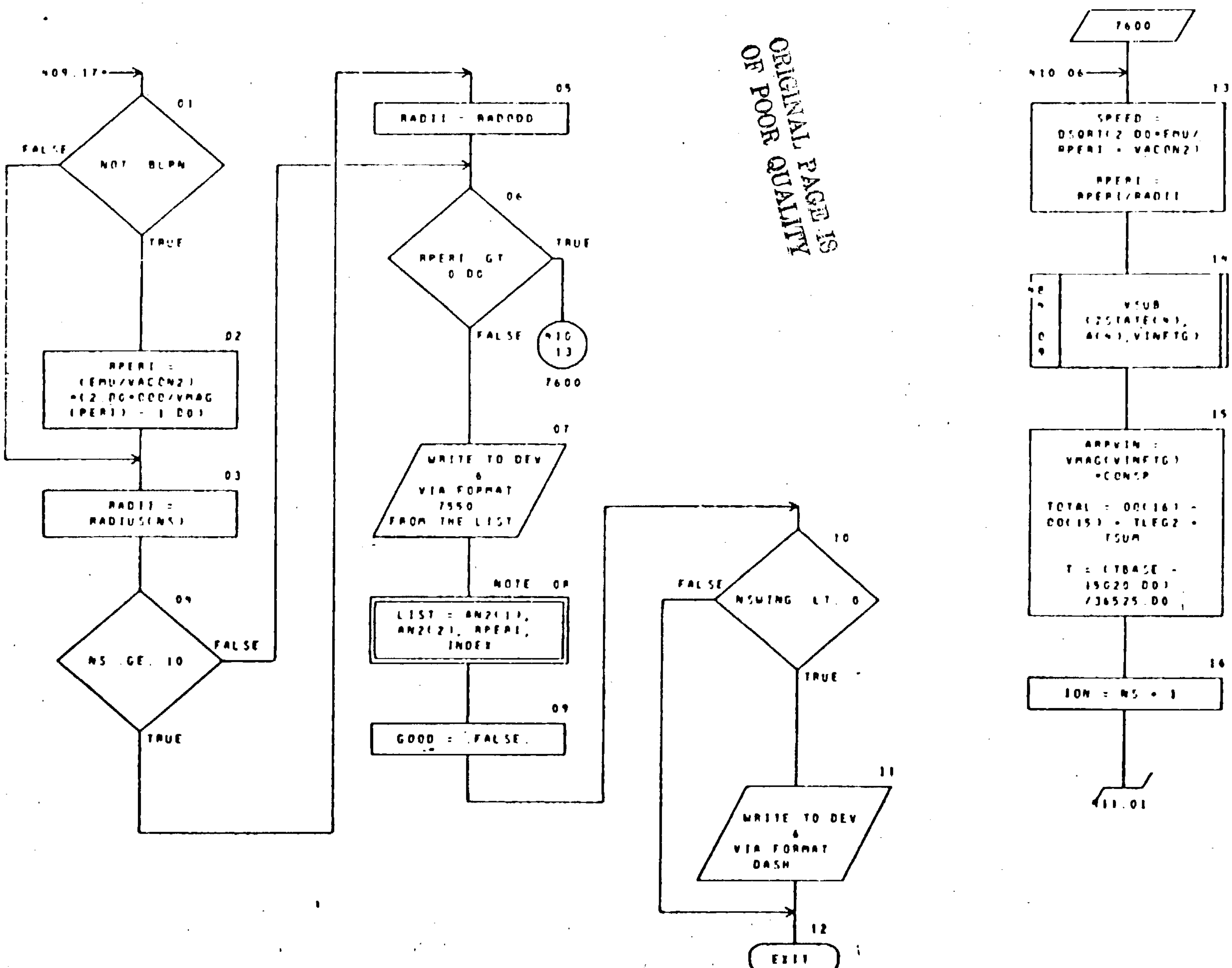
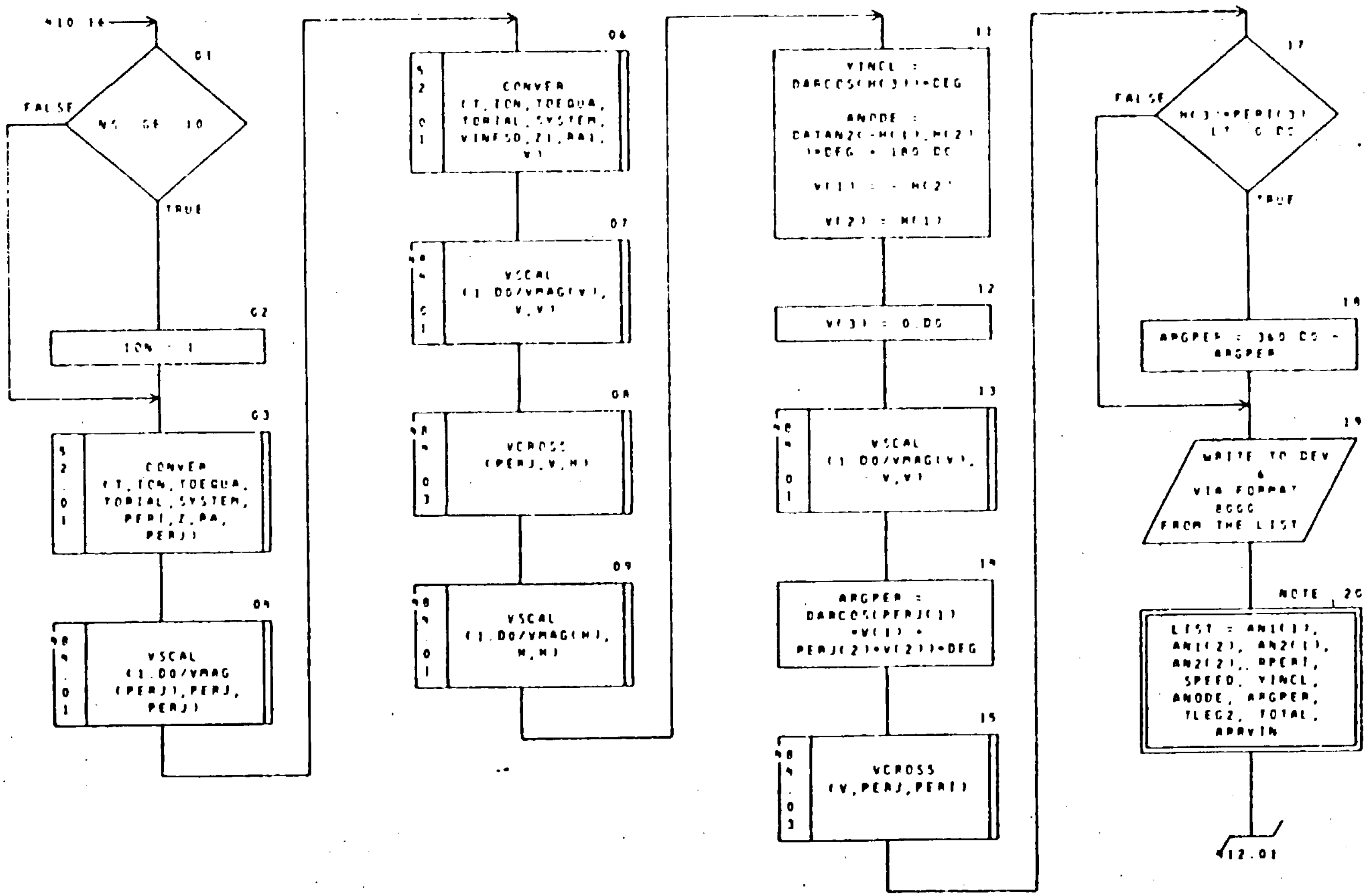


CHART TITLE - SUBROUTINE SWING(INDEX,NS,NT,TLEG2,GOOD,BURN,RPERI,FSTATE,YSTATE,



ORIGINAL PAGE IS OF POOR QUALITY

1003

SWING-21

CHART TITLE - SUBROUTINE SWING(INDEX,NS,NT,TLFG2,GOOD,BURN,RPERI,XSTATE,YSTATE,

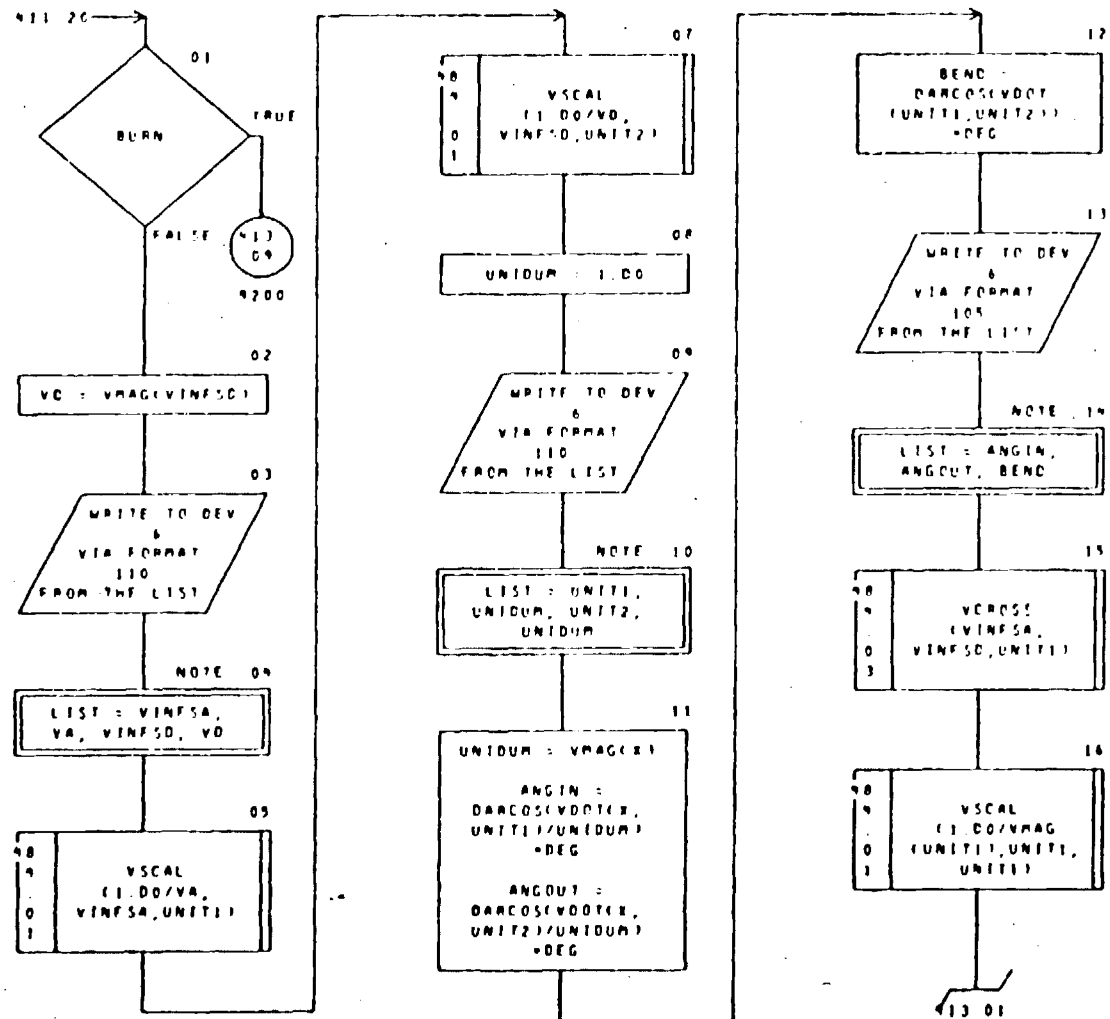
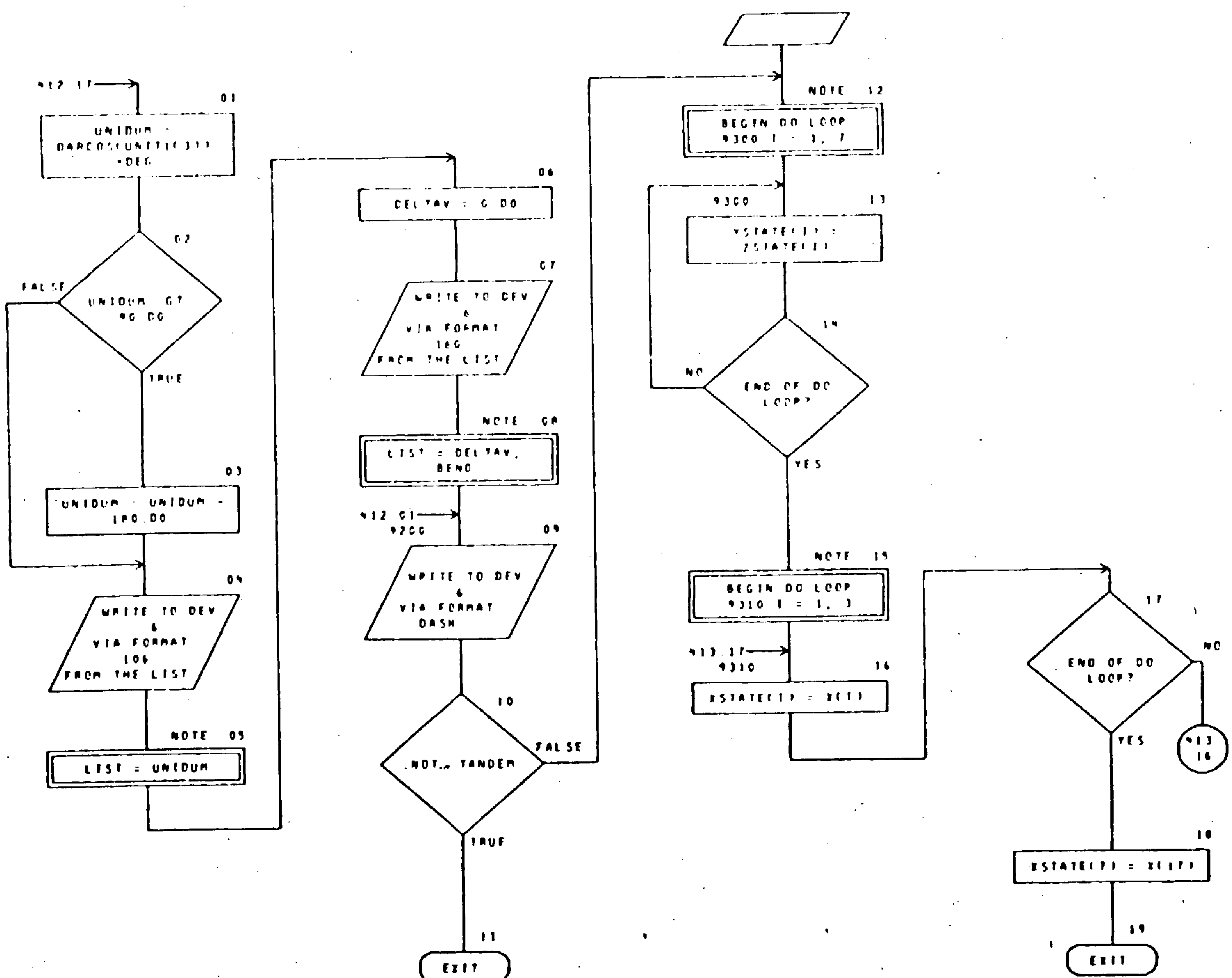


CHART TITLE - SUBROUTINE SWING(INDEX,NC,NT,TLEG2,GOOD,BURN,RPERT,XSTATE,YSTATE,



SWING-23

CHART TITLE - NON-PROCEDURAL STATEMENTS


```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL GOOD,ERRORX,CONVGE,TRACK,TRACKS,OJEX,OJEXS,BURN,QUIT
     ,TANDEM
      DIMENSION XSTATE(7),MSET(5),VINFSD(3),A(6),VINFSA(3),ADOT(6),
     PERI(3),AN1(2),AN2(2),     VINFTG(3),PERJ(3),V(3),H(3),DASH(18)
     ,UNIT1(3),UNIT2(3),YSTATE(7)
      COMMON /REAL8/ RO1(83),EMUCDD,RADCDD,RO2(14),CCC,RO3(32),
     SEPAR(7),RON(17),TDATE2,TDATE,ROS(A7),SATX(5),ECIX(5),ENIX(5),
     OMIX(5),SOIX(5),TPIX(5),EMUODX(5),RADODX(5),RO9(47),DEG,
     RO7(27),CONTM,CONSP,RO8(63),YSWING(3,10),R10(390),
     PVELOC(3),TBASE,TSUM,ZSTATE(7),R12(438),X(50),R9A(700)
      COMMON /INIGRN/ IO1(35),NTARG,IO4(109),INTERX,
     IOS,NLEAVE,IO2(852)
      COMMON /LOGICN/ LO1(24),TRACK,LO2,OJEX,LO3(17),QUIT,LO4(2),
     TANDEM,LO5(452)
      COMMON /ITERAT/ BO1(770),DD(70),BO2(140)
      COMMON /ITER2/ B(35),Q(35),BS(35),BW(35),QMIN(35),QMAX(35),BBB(35)
     ,PO1(1295)
      COMMON /SOLSYS/ GM(70),RADIUS(70),APL(2,70)
      EXTERNAL SWTRAJ,SWPRNT
      DATA TOL,MAX,STEP /1 D-12,26,.8D0/
      DATA DASH /8H(1HO130H,16*8H--------,3H--)/
 7910 FORMAT(1H ,7HINDEX =I3)
 7923 FORMAT(1H ,47HFATAL ERROR. SWINGBY INITIAL VELOCITY GUESS = 0)
  390 FORMAT(1H ,12HERROR VECTOR3X1P4D16.8,5X6HTIME =0PF10.3,6H DAYS.)
```

CHART TITLE - NON-PROCEDURAL STATEMENTS


```
400    FORMAT(1H0,33HUNPOWERED SWINGBY FAILS. POWERED ,
       48H SWINGBY ANALYSIS FOLLOWS, FOR FIXED SWINGBY LEG ,
       11HFLIGHT TIME F8.1,4H DAYS /1H )
415    FORMAT(1H0,48HPOWERED SWINGBY ANALYSIS ONLY, FOR FIXED SWINGBY,
       16H LEG FLIGHT TIME F8.1,6H DAYS /1H )
418    FORMAT(1H0,43HPOWERED SWINGBY ANALYSIS, FOR FIXED SWINGBY,
       16H LEG FLIGHT TIME F8.1,4H DAYS,/1H ,
       50HAND USING LAST VELOCITY GUESS FROM ABOVE ITERATION3X1P3D16.8/1H
        )
110    FORMAT(1H0,15HARRIVAL    VOO =1P3D16.8,5X5HMAG =D16.8,
       4X27H(ECLIPTIC REFERENCE SYSTEM)/1H ,
       15HDEPARTURE VOO =3D16.8,5X5HMAG =D16.8,
       4X27H(ECLIPTIC REFERENCE SYSTEM)/1H )
105    FORMAT(1H ,29HHELIOCENTRIC APPROACH ANGLE =F6.1,
       17H,  DEPART ANGLE =F6.1,15H,  BEND ANGLE =F6.1,9H DEGREES )
106    FORMAT(1H ,37HSWINGBY INCLINATION W R T  ECLIPTIC =F6.1,9H DEGREES
        /1H )
141    FORMAT(1H0,44HITERATION FOR POWERED SWINGBY PASSAGE RADIUS,
       25H DID NOT CONVERGE    RP =1PD12.4,3X3HF =D12.4,3X4HFP =D12.4,
       3X7HINDEX =I3/1H ,22HPOST-SWINGBY TARGET = 2A8/1H )
7370   FORMAT(1H0,16HNEARLY CONVERGED)
160    FORMAT(1H0,35HPOWERED SWINGBY INCREMENTAL SPEED =F9.1,
       30H METERS/SECOND    BEND ANGLE =F6.1,9H DEGREES.,
       2X16H(PLANETOCENTRIC)/1H )
170    FORMAT(1H ,9HDELTA V =F9.1,4H M/S)
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
550     FORMAT(1H ,12HERROR VECTOR3X1P3D16.8)
7440    FORMAT(1H0,24HSWINGBY ITERATION FAILS..,
        6X8HERRORX =L2,4X8HCONVGE =L2,4X7HINDEX =I3)
7550    FORMAT(1H ,10HSWINGBY TO1X2A8,
        33H PASSAGE DISTANCE NEGATIVE   RP =D12.4,4X7HINDEX =I3)
8000    FORMAT(1H0,40X2A8,24H SWINGBY CONTINUATION TO1X2A8//1H ,
        52HPASS DIST (RADII)  SPEED (M/SEC)  INCLIN (DEG)  NODE,
        61H (DEG)  ARG PER (DEG)    LEG TIME (DAYS)  MISSION TIME (DAYS),
        18H  ARR VINF (M/SEC)//1H ,F13.4,F17.2,F14.4,F13.4,F14.3,
        F16.2,F21.2,F18.2)
```

Name: SWSTO

Calling Argument: None

Referenced Sub-programs: SOLAR

Referenced Commons: INTGR4, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: TAP

Discussion: SWSTO stands for "switch store". The basic function of the subroutine is to store, for later printout at the finish of the final trajectory of each case, specific quantities at each thrust switching point, including the beginning and end points of each trajectory segment. The subroutine also provides the vital function of monitoring the constancy of the variational hamiltonian along the trajectory. If the variational hamiltonian is not sufficiently constant, as determined by certain criteria within the routine, then the routine prints conspicuous messages to alert the program user that either (1) the computation step size is too large for the trajectory at hand, or, (2) the changes made to the program recently are erroneous, since they did not yield the necessary condition of a constant variational hamiltonian. The latter possibility occurs frequently when complicated changes are made to the program, and the programmer's job remains unfinished so long as the program spits the BAD HAMILTONIAN message back at him.

The variational hamiltonian h_v is computed as follows. At the end of (or during) a coast phase, the total variational hamiltonian is simply computed as

$$h_{v\,coast} = -\dot{\Lambda} \cdot \dot{R} + \Lambda \cdot \ddot{R}$$

The above quantity is computed in any case, since it forms a component of the total variational hamiltonian at the end of (or during) a thrust phase, in which case

$$h_v = h_{v\ coast} + \frac{g\gamma q}{\nu}\left(\Lambda \cdot \bar{e}_t - \frac{\nu \lambda_\nu}{c}\right) + \lambda_s d + \lambda_\tau$$

is the total variational hamiltonian. Once computed, the value of the variational hamiltonian is stored in the next available location of a storage array, which contains the values of the variational hamiltonian at thrust switching points and at the beginning and end points of trajectory segments. The time is also stored.

The constancy of the variational hamiltonian is then tested as follows. If the absolute value of (the current time minus the previous time stored) is less than 10^{-8} tau, then skip the test. The test is thus avoided at junctions of trajectory segments, since the variational hamiltonian may validly be discontinuous at such points. Otherwise, let the test quantity be computed as the absolute value of (the current h_v minus the previously-stored h_v) divided by a normalizing factor. Then, if the test quantity is greater than 2×10^{-4} (an arbitrary value picked from experience, which avoids excessive message printouts), print the conspicuous warning message BAD HAMILTONIAN.

Messages and printouts: The storage arrays contain fifty locations. Should the storage array index exceed fifty, the message

50 THRUST SWITCHING POINTS EXCEEDED

is printed on units 6 and 12, the master error indicator is set, and the routine is exited.

Should the variational hamiltonian be deemed sufficiently non-constant, as determined by the test discussed in the preceding section, then the following message is printed on unit 6:

```
- - - - - - - - - - - - - - - - BAD HAMILTONIAN - - - - - - - - - - - - - - - -
                              WARNING NO. __n__
        AT TIME __t_i-1__, THE HAMILTONIAN IS __h_i-1__
        AT TIME __t_i__, THE HAMILTONIAN IS __h_i__
```

where n indicates the number of bad hamiltonian occurrences during the current computer run, t_{i-1} and t_i are the previously-stored time and the current time along the trajectory, in days, and h_{i-1} and h_i are the corresponding hamiltonian values, which will not be equal. Concurrently, the message

BAD HAMILTONIAN. NORMALIZED ERROR = <u>test</u> IN <u>Δt</u> DAYS.

is written on unit 12, where "test" is the test quantity discussed in the preceding section and Δt is $t_i - t_{i-1}$ in days. This unit 12 message will be printed out a maximum of three times, and the unit 6 message will be printed a maximum of MAXHAM times. When the maximum number of messages on unit 6 is reached, the following is also printed:

ABOVE WARNING IS LAST WARNING FOR THIS RUN

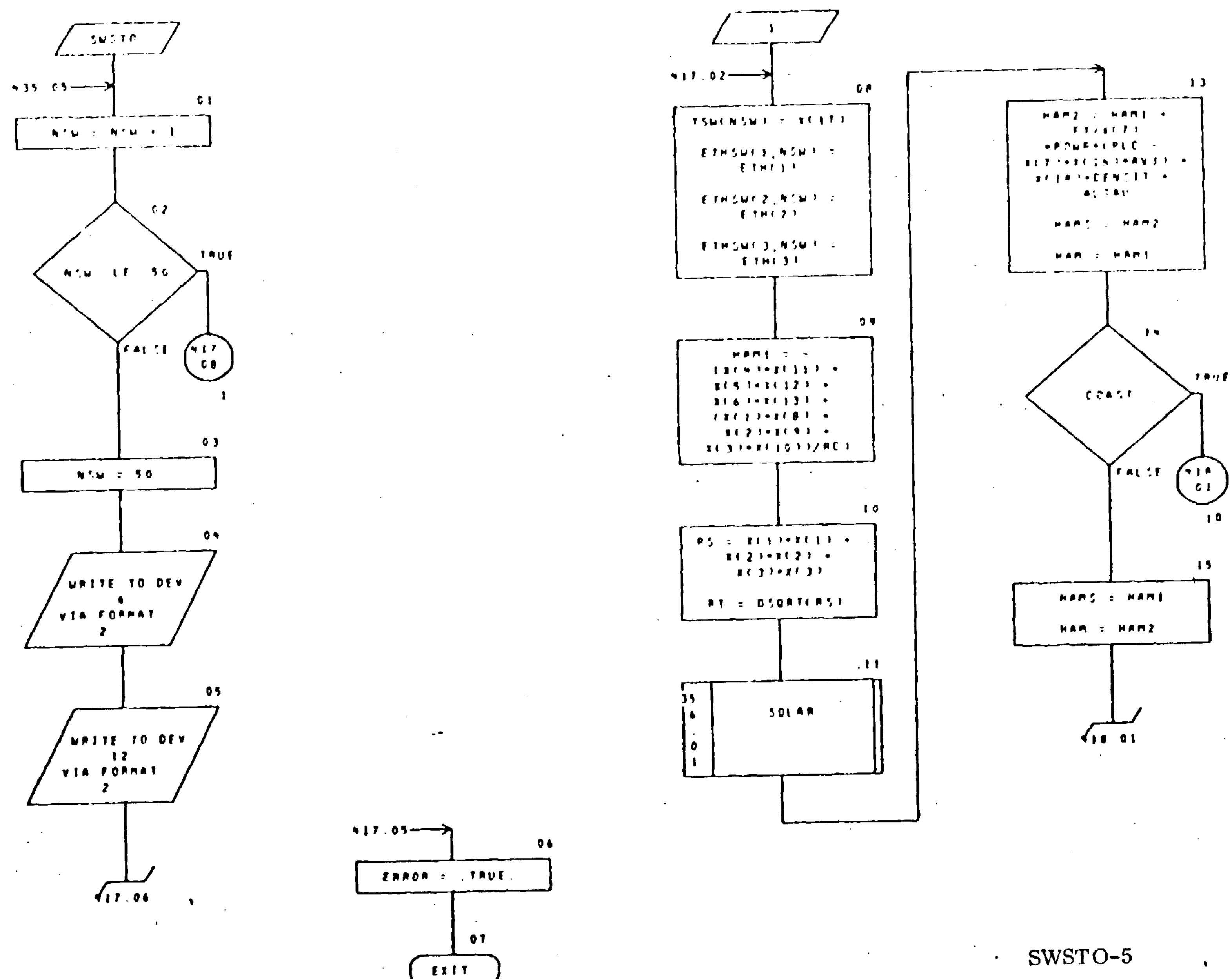
SWSTO EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
X(50)	U	REAL8	Array of trajectory integrated variables.
FT	U	REAL8	Thrust acceleration g, in AU/tau^2.
RC	U	REAL8	Cube of spacecraft solar distance, r^3, in AU^3.
RS	SU	REAL8	Square of spacecraft solar distance, r^2, in AU^2.
RT	SU	REAL8	Spacecraft solar distance, r in AU.
AVJ	U	REAL8	Inverse of spacecraft jet exhaust speed, in EMOS^{-1}.
ETH(3)	U	REAL8	Thrust unit vector.
HSW(50)	SU	REAL8	Stored values of the variational hamiltonian, h_i.

SWSTO EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ISW(50)	S	INTGR4	Stored values of indicator for thrust phase or coast phase.
NSW	SU	INTGR4	Counter or index for storage arrays, incremented by one each time SWSTO is called.
PLC	U	REAL8	Dot product of primer vector and unit thrust vector, $\Lambda \cdot \bar{e}_t$.
TSW(50)	SU	REAL8	Stored values of time, t_i, in tau.
POWR	U	REAL8	Spacecraft power function, γq.
ALTAU	U	REAL8	Propulsion-time adjoint variable, λ_τ.
COAST	U	LOGIC4	Indicator for thrust phase or coast phase.
CONTM	U	REAL8	Time conversion factor, tau to days.
ERROR	S	LOGIC4	Program master error indicator.
ETHSW (3,50)	S	REAL8	Stored values of thrust unit vector.
DENSIT	U	REAL8	Density function associated with the power law.
JUMPED	U	LOGIC4	Indicator for discontinuous primer derivative, set in subroutine TAP.
MAXHAM	U	INTGR4	Maximum number of times program will check hamiltonian constancy.

CHART TITLE - SUBROUTINE SWSTO



SWSTO-5

CHART TITLE - SUBROUTINE SMSTO

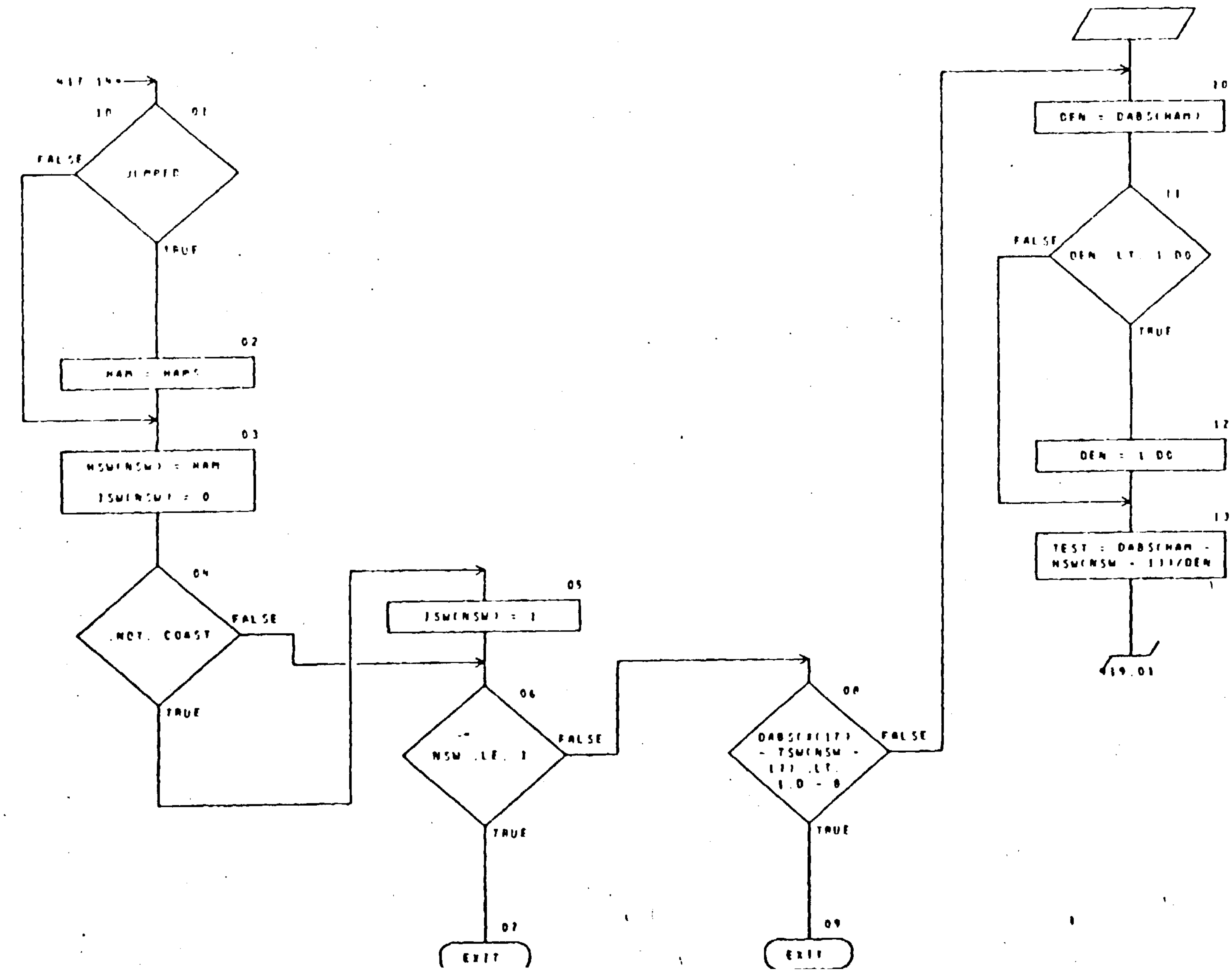
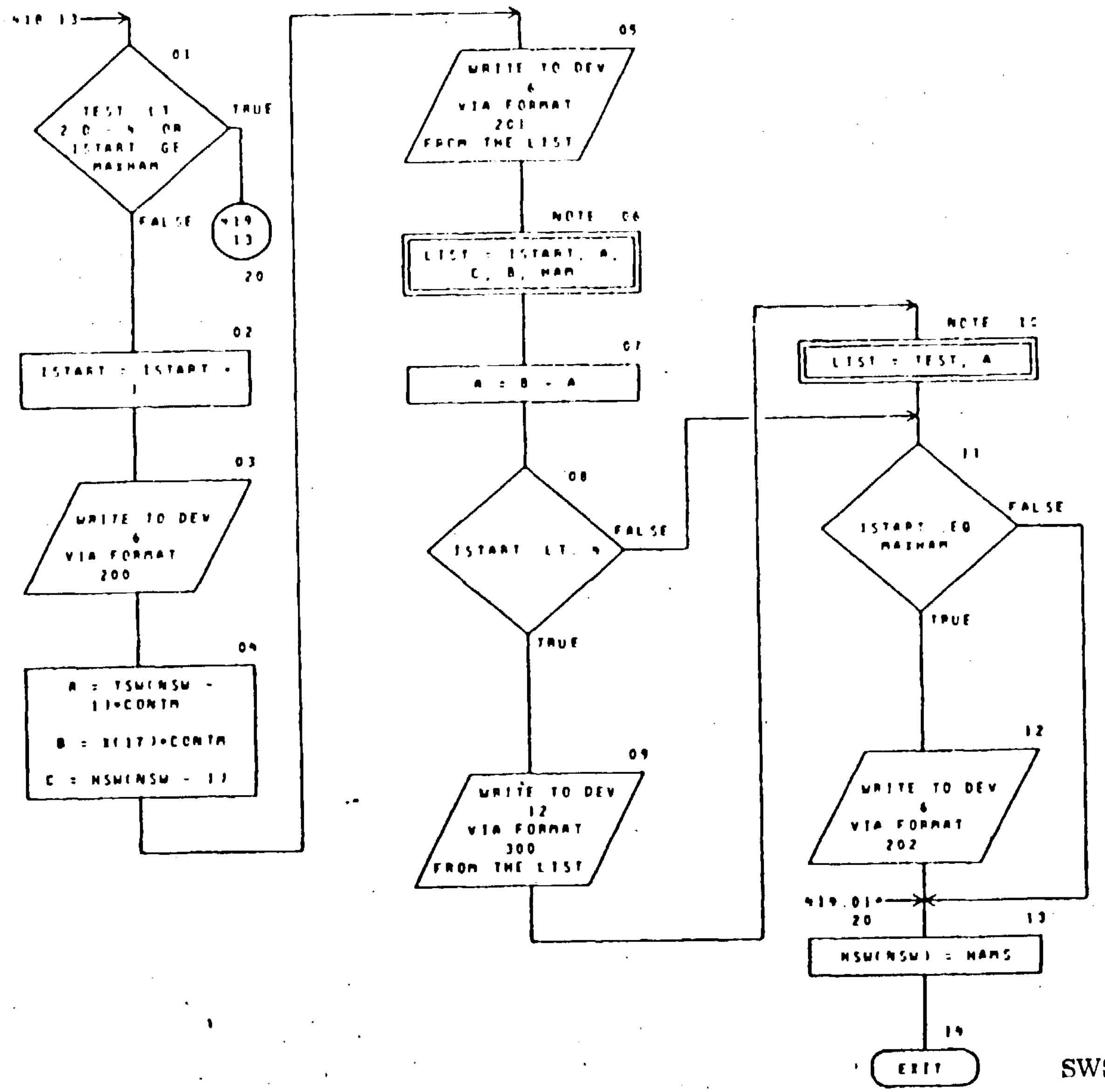


CHART TITLE - SUBROUTINE SWSTO



SWSTO-7

ORIGINAL PAGE IS
OF POOR QUALITY

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL COAST,ERROR,JUMPED
      COMMON /REAL8/ R01(11),FT,R02(163),ALTAU,R04(139),
     1                 CONTM,R05(389),TSW(50),HSW(50),R06(25),FTH(3),
     2R07(15),PLC,RC3(6),      ETHSW(3,50),R08(197),
     3RT,RS,RC,R09(4),POWE,R10(14),AVI,R12(6),DENSIT,R13(73),X(50),
     4R11(700)
      COMMON /INTGRS/ I01(74),NSW,ISW(50),I02(21),MAXHAM,I03(85)
      COMMON /LOGICS/ ERROR,LC1(30),COAST,LC2(8),JUMPED,LC3(459)
      DATA ISTART /0/
    2 FORMAT(1H ,35HSO THRUST SWITCHING POINTS EXCEEDED/1H )
  200 FORMAT(1H ,50H--------------------------------------------------
     1,15HBAD HAMILTONIAN,
     260H------------------------------------------------------------)
  201 FORMAT(1H ,30X11HWARNING NO I2/1H ,
     120X7HAT TIMEF13.6,20H, THE HAMILTONIAN IS025.15/1H ,
     220X7HAT TIMEF13.6,20H, THE HAMILTONIAN IS025.15)
  300 FORMAT(1H ,35HBAD HAMILTONIAN. NORMALIZED ERROR =F10.6,
     13H INF7.1,4H DAYS )
  202 FORMAT(1H ,42HABOVE WARNING IS LAST WARNING FOR THIS RUN)
```

Name: SWTRAJ

Calling Argument: ERRORX for SWTRAJ; INIDUM, JNIDUM, UNIDUM for SWPRNT

Referenced Sub-programs: EFM, TAP, TAPSET for SWTRAJ; None for SWPRNT

Referenced Commons: INTGR4, ITER2, REAL8

Entry Points: SWPRNT

Referencing Sub-programs: MINMX3, SWING for SWTRAJ; MINMX3 for SWPRNT

Discussion: SWTRAJ is a contraction of "swingby trajectory" and has the purpose of generating a post-swingby trajectory segment (via a call to TAP) and producing two-point boundary-value problem end conditions q_1, q_2, q_3, and q_4 for the MINMX3 iterator; SWTRAJ thus produces the mapping $X \rightarrow Y$ for the iterator. Entry point SWPRNT is a dummy print subroutine whose presence is required by MINMX3.

The iterator independent-variables for the post-swingby targeting problem, which allow the iterator to control the ballistic trajectory segment from the swing-by planet to the post-swingby target for a given phase of a total mission, are denoted b_1, b_2, b_3, and b_4, and consist of the following:

$$\begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix} = \begin{pmatrix} \dot{x}_o \\ \dot{y}_o \\ \dot{z}_o \\ \Delta t \end{pmatrix}$$

where $\dot{R}_o = (\dot{x}_o, \dot{y}_o, \dot{z}_o)$ is the spacecraft's (total) heliocentric departure velocity at the swingby planet and Δt is the flight time of the post-swingby trajectory segment.

The iterator dependent-variables, which are driven to zero, are

$$\begin{pmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{pmatrix} = \begin{pmatrix} x - x_t \\ y - y_t \\ z - z_t \\ v_{\infty D} - v_{\infty A} \end{pmatrix}$$

where $\Delta R = R - P_t = (q_1, q_2, q_3)$ is the position targeting error between spacecraft R and target P_t at target intercept time determined by b_4, and $v_{\infty D} - v_{\infty A}$ is the difference between departure and arrival hyperbolic excess speeds at the swingby planet and is used only when unpowered swingbys are required. $v_{\infty A}$ is pre-computed and $v_{\infty D}$ is computed directly as

$$v_{\infty D} = |\dot{R}_o - \dot{P}_s| = \sqrt{(\dot{x}_o - \dot{x}_s)^2 + (\dot{y}_o - \dot{y}_s)^2 + (\dot{z}_o - \dot{z}_s)^2} \quad ,$$

where $\dot{P}_s = (\dot{x}_s, \dot{y}_s, \dot{z}_s)$ is the swingby planet's velocity at swingby time.

Messages and printouts: Should subroutine CHECK fail to isolate the final time of the trajectory segment to a specified tolerance, the message is printed on unit 6:

TRANSFER TIME FAILS IN SUBROUTINE SWTRAJ X(17), TMAX = (t) (t_{max})

where t is the actual time of the trajectory endpoint, in tau = AU/EMOS, measured from the start of the whole trajectory, and t_{max} is the required value which t is to attain, which is computed in TAPSET as a function of the iterator independent-variable b_4. The error indicator is set and the subroutine is exited.

SWTRAJ EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(35)	U	ITER2	Iterator independent-variables b_1, b_2, b_3, and b_4, in EMOS and tau.
Q(35)	S	ITER2	Iterator dependent-variables q_1, q_2, q_3, and q_4, in AU and EMOS.

SWTRAJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
X(50)	SU	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
OOO	U	REAL8	Swingby arrival hyperbolic excess speed, $v_{\infty A}$, in EMOS.
TMAX	U	REAL8	Desired time at end of trajectory-segment, t_{max}, in tau.
TSUM	U	REAL8	Time elapsed since primary-target swing-by and current swingby, in days.
CONTM	U	REAL8	Time conversion factor, tau to days.
NTARG	A	INTGR4	Post-swingby target planet selector.
TBASE	U	REAL8	Julian date at time of swingby (less 2400000).
ERRORX	SX		Error indicator.
INIDUM	AX		Dummy argument.
INTERX	A	INTGR4	Index which selects the orbital elements of the post-swingby target when it is not a major planet.
JNIDUM	X		Dummy argument.
NLEAVE	A	INTGR4	Swingby planet selector.
PVELOC(3)	U	REAL8	Swingby planet's velocity at swingby time, $\dot{P}_s$, in EMOS.
UNIDUM	AX		Dummy argument; Universal Dummy Variable.
ZSTATE(7)	S	REAL8	Spacecraft position R, velocity $\dot{R}$, and the time at the end of the trajectory-segment, in AU, EMOS, and tau, respectively.

SWTRAJ-3

CHART TITLE - SUBROUTINE SWTRA(ERRORX)

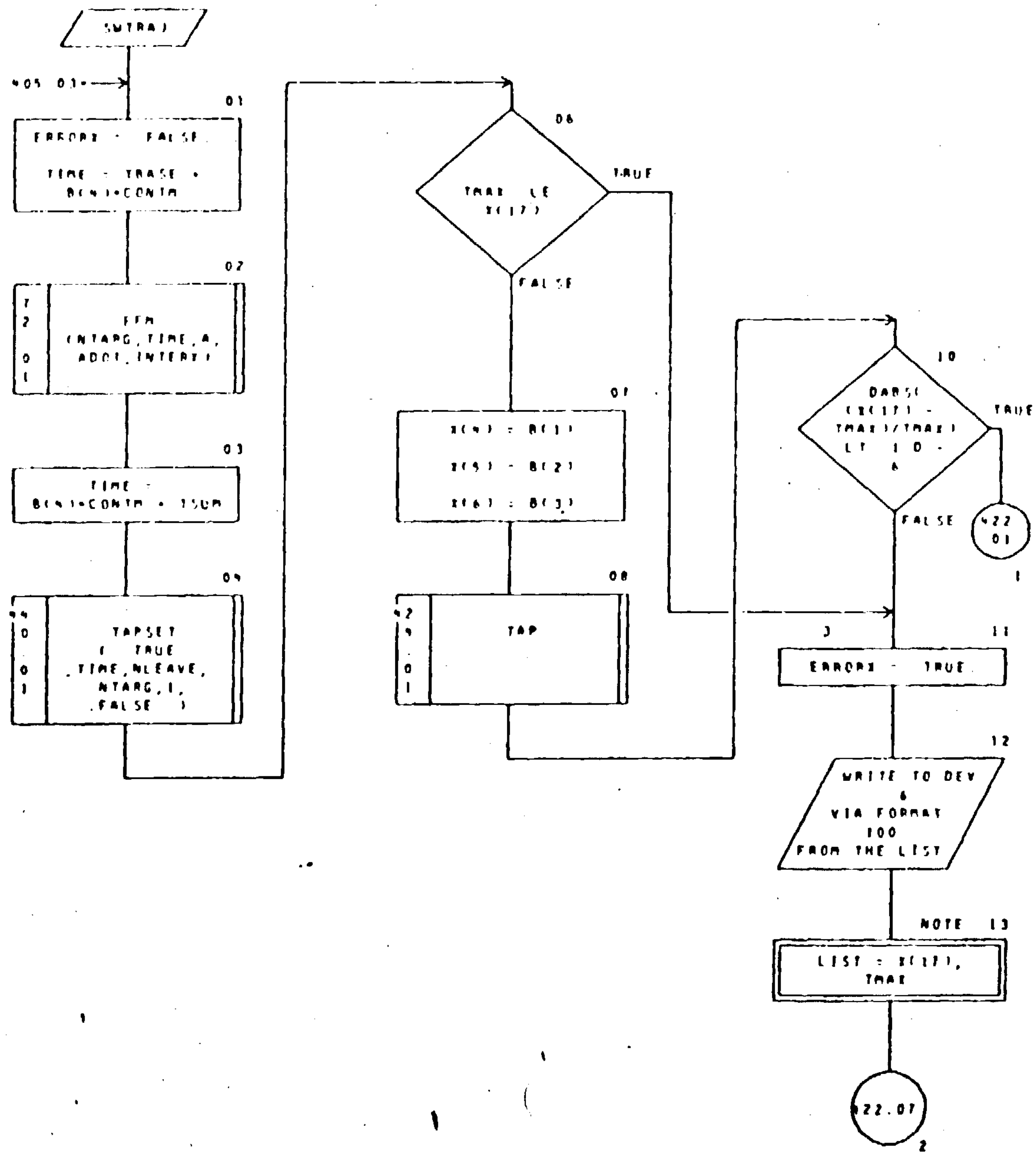


CHART TITLE - SUBROUTINE SWTRAJ(ERRORX)

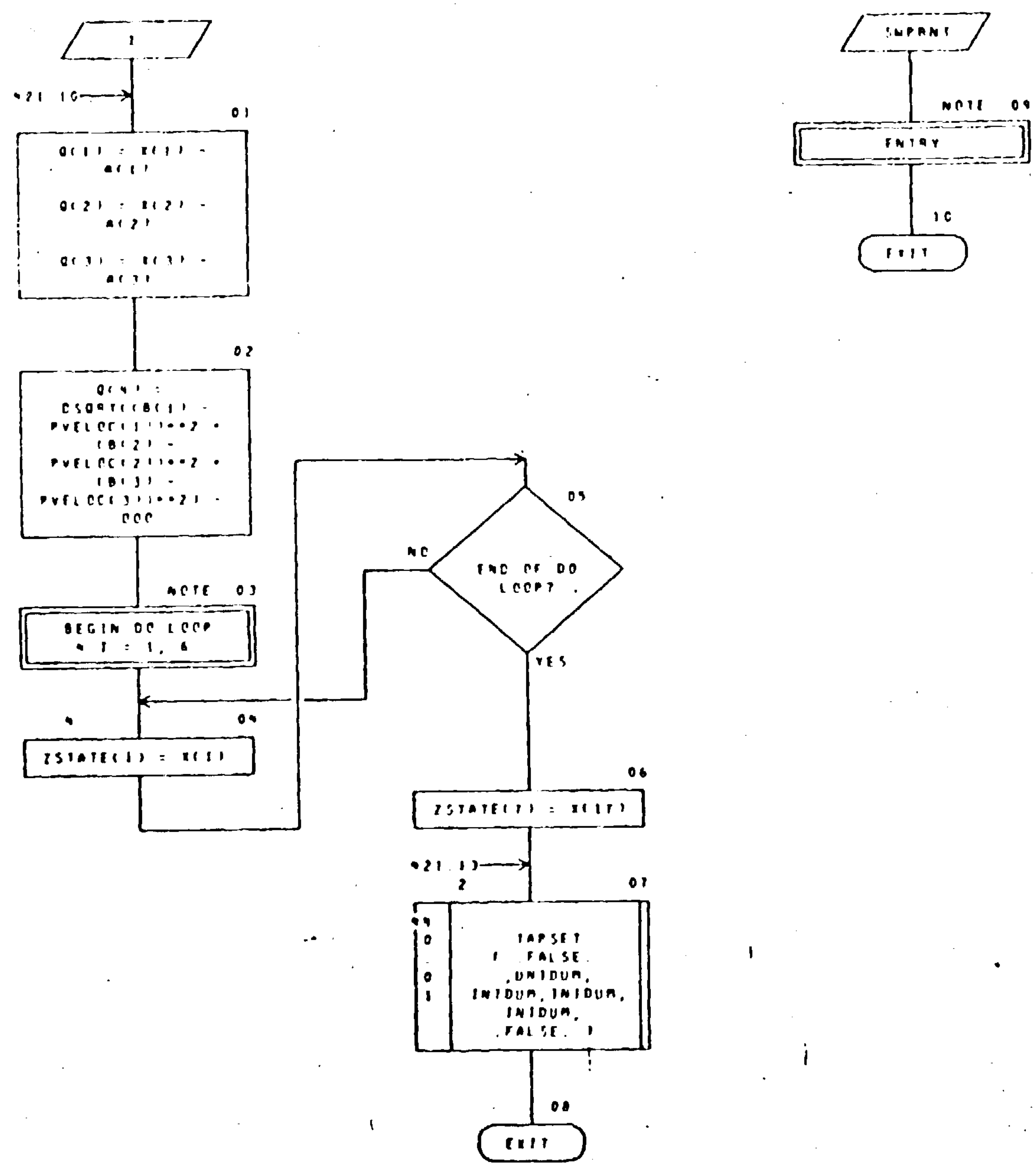


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL ERRORX
      DIMENSION A(6),ADOT(6)
      COMMON /REAL8/ RO1(84),TMAX,RO2(15),DOO,RO3(215),CONTM,
     RO4(485),PVELOC(3),TBASE,TSUM,ZSTATE(7),RO5(43P),X(50),RO7(700)
      COMMON /INTGRS/ IC1(35),NTARG,IO3(109),INTERX,IO4,NLEAVE,IO2(852)
      COMMON /ITER2/ B(35),D(35),PC1(1575)
  100 FORMAT(1H ,NONTRANSFER TIME FAILS IN SUBROUTINE SWTRAJ,
     6X12HX(17),TMAX =2D20.12)
```

Name: TAP

Calling Argument: None

Referenced Sub-programs: CDERIV, CHECK, CHKINT, CONVRT, DERIV, FUNCT, SOLAR, SPRINT, STEP, SWSTO, THANG, THANGD, TRAVEL, VADD, VMAG, VSCAL

Referenced Commons: EXTREM, INTGR4, ITERAT, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: MORE, SWTRAJ, TRAJ

Discussion: Subroutine TAP generates a single trajectory-segment, corresponding to the flight of the spacecraft from a time associated with one celestial body to a time associated with the next celestial body in the mission sequence. After targeting-convergence is achieved, the endpoints of the trajectory segment match the positions of the two celestial bodies associated with the segment; these may be the launch planet, the primary target, intermediate targets, or post-swingby targets.

The heart of the subroutine consists of the two adjacent statements where subroutines STEP and CHECK are referenced. STEP performs a computation step, and CHECK checks for remarkable points within that computation step and supervises iterations to isolate the remarkable points (such as thrust switch points and critical solar distance).

Approximately the first half of the subroutine (up to where CHKINT is referenced) consists of initialization prior to the first computation step of the trajectory segment. Many logical-indicators which do not vary over the segment are initialized here. The quantities which are computed correspond to the starting point of the trajectory segment. These consist of the primer magnitude,

$$\lambda = \sqrt{\Lambda \cdot \Lambda} \quad ,$$

the spacecraft solar distance,

$$r = \sqrt{R \cdot R} \quad ,$$

the time-derivative of the primer magnitude,

$$\dot{\lambda} = \sqrt{\Lambda \cdot \dot{\Lambda}}\,/\lambda \qquad \text{when } \lambda \neq 0,$$

$$\dot{\lambda} = \sqrt{\dot{\Lambda} \cdot \dot{\Lambda}} \qquad \text{when } \lambda = 0.$$

The bulk of the initialization computations are concerned with the computation of the variational Hamiltonian h_v, which is given by

$$h_v = h_\sigma\left[\frac{g\gamma q}{\nu}(\Lambda \cdot \bar{e}_t - \frac{\nu}{c}\lambda_\nu) + \lambda_s d + \lambda_\tau\right]$$

$$- \frac{\mu}{r^3}(\Lambda \cdot R) - \dot{\Lambda} \cdot \dot{R},$$

and also of the thrust switch function σ (corresponding to the start of the trajectory segment), which is given by

$$\sigma = \frac{\nu}{g\gamma q}\,\sigma^*$$

where

$$\sigma^* = \frac{g\gamma q}{\nu}(\Lambda \cdot \bar{e}_t - \frac{\nu}{c}\lambda_\nu) + \lambda_s d + \lambda_\tau.$$

The computation of σ follows the more detailed discussion found in the documentation of subroutine FUNCT. The degradation factor is also initialized,

$$q = e^{-s/\tau_d}.$$

An inherent singularity of the optimal rocket flight problem is characterized by an extremely high thrust rotation rate and occurs whenever the primer vector passes relatively close to the origin of primer-space ($\Lambda = 0$) during a thrust phase. The difficulty associated with the primer-origin singularity is lessened by continously cutting down the computation step size $\Delta\beta$ as the primer origin is approached. This is accomplished by the formula,

$$\Delta\beta^{+} = \Delta\beta^{-}/t_{w}$$

where the quantity t_w is greater thanunity and is given by

$$t_w = \left[\log_{10}\left(\sqrt{\dot{\Lambda}\cdot\dot{\Lambda}}\,/\lambda\right)^2\right]^{\frac{3}{2}}.$$

TAP-3

Whenever the spacecraft passes the critical solar distance r_c when the solar power law option being simulated corresponds to letting the power factor $\gamma \to 0$ due to solar cell overheating, the primer derivative $\dot{\Lambda}$ is discontinous and is jumped according to the formula,

$$\dot{\Lambda}^{+} = \dot{\Lambda}^{-} + \left[\frac{\lambda_{\tau} + \lambda_{s}/r_c^{2}}{|R_c \cdot \dot{R}_c|}\right] R\,,$$

in which subscript c corresponds to the critical solar distance $(\gamma \to 0)$.

In simulations of trajectories which are all-ballistic, the program is capable of simulating a single deep-space burn, or impulsive velocity-change, at any point prior to arrival at the primary target. The three components of the incremental velocity ΔV are independent variables of the boundary value problem, such that, at a specified time, the spacecraft velocity is incremented:

$$\dot{R}^{+} = \dot{R}^{-} + \Delta V,$$

where ΔV is produced by the MINMX3 iterator and

$$\Delta v = |\Delta V| = \sqrt{\Delta V \cdot \Delta V}$$

is computed using subroutine VMAG.

Messages and printouts: Whenever the mass ratio ν approaches zero as the spacecraft thrusts along the trajectory (a condition which is prone to occur especially when simulating nuclear electric propulsion), the program declares an error condition and exits the subroutine after printing the message on units 6 and 12:

MASS RATIO = $\underline{(\nu)\quad}$, VANISHING NEAR TIME = $\underline{(t)\quad}$ DAYS

where t is the elapsed time since the start of the trajectory, in days. The message may occur several times before the iterator finally terminates the iteration sequence. If the analyst believes the solution he seeks actually exists, he should try different trajectory starting values; otherwise, it is possible that the solution required does not exist as a physical possibility.

The remainder of messages from subroutine TAP are concerned with the deep space burn option. When this option is invoked, the following is printed during the final, case-summary trajectory at the actual moment the deep space burn velocity increment is added to the spacecraft velocity; on unit 6,

DEEP SPACE BURN $\underline{(\Delta v)\quad}$ METERS/SECOND AT $\underline{(t)\quad}$ DAYS

where Δv is the magnitude of the deep space burn, in meters per second, and t is the elapsed time since the start of the trajectory, in days. On unit 12 is printed,

DEEP SPACE BURN $\underline{(\Delta v)\quad}$ M/S

Should the integration-stopping index JCMAX be greater than one (which should not occur on an all-ballistic trajectory segment), an error condition is declared and the subroutine is exited after the message is printed on unit 6,

ERROR. JCMAX.NE.1 IN *TDV* OPTION. $\underline{(JCMAX)}$ $\underline{\quad(t')\quad}$ $\underline{\quad(t_d)\quad}$ $\underline{\quad(t_{max})\quad}$

where t' is the elapsed time since the start of the trajectory, t_d is the time of the deep space burn, and t_{max} is the trajectory-segment endpoint time, all in units of tau.

TAP EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(2,30)	S	EXTREM	Array of monitored functions.
O(70)	SUA	ITERAT	Array of iterator independent-variables; O(21) is the thrust angle, when it is held constant, in radians.
R(2)	SUA	REAL8	Spacecraft solar distance, r, at start of computation step (R(1)) and instantaneous (R(2)), in AU.
X(50)	SUA	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
AN	U	REAL8	Trajectory-integration exponent in regularization formula.
FT	U	REAL8	Reference thrust acceleration, g, in AU/tau^2.
JC	S	INTGR4	Counter corresponding to the JCth specified time function value (i.e., the time) isolated, or to be isolated, thus far on the current trajectory segment. Attains values greater than one when imposed coast phases are invoked.
JJ	S	INTGR4	Thrust switch point or critical solar distance indicator.
KF	SUA	INTGR4	End of trajectory-segment indicator.
PP(2)	SU	REAL8	Primer vector magnitude, λ, at start of computation step (PP(1)) and instantaneous (PP(2)).
RC	SU	REAL8	Cube of spacecraft solar distance, r^3, in AU^3.
RT	SUA	REAL8	Spacecraft solar distance, r, in AU.
SX(50)	SU	REAL8	Array of trajectory integrated variables, corresponding to the start of the current computation step.

TAP EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
AVJ	U	REAL8	Inverse of jet exhaust speed, 1/c, in $EMOS^{-1}$.
A1S	U	REAL8	Leading power-law coefficient, a_o.
ETH(3)	SUA	REAL8	Thrust unit vector, $\bar{e}_t$.
HAM	SU	REAL8	Variational Hamiltonian, h_v.
PLC	SU	REAL8	First component of thrust switch function, σ_1.
PMN	SUA	REAL8	Primer magnitude, λ.
R1N	SU	REAL8	Conversion factor from generalized derivatives to time derivatives, r^n, in AU^n.
R2N	S	REAL8	r^{2n}.
TDV	U	REAL8	Time of deep space burn, in days.
EDGE	S	LOGIC4	Indicator for solar arrays being oriented edgewise to the sun; used only if power degradation is simulated.
ETHD(3)	U	REAL8	Thrust unit vector time-derivative, $\dot{\bar{e}}_t$, in tau^{-1}.
HEAT	U	LOGIC4	Indicator for maintaining solar panels normal to sun at all times, including during high solar proximity.
MODE	U	INTGR4	Power variation option selector.
NPHI	SU	INTGR4	Number of fixed thrust cone angles permitted. (Currently limited to one).
PLUS	S	LOGIC4	Indicator for determining appropriate region in two-dimensional simulations, as described in subroutine THANGD.

TAP EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
POWR	U	REAL8	Power ratio, γq.
QJEX	U	LOGIC4	Detailed printout indicator.
SWIT	SU	REAL8	$\sigma_1 + \sigma_2$ (See subroutine FUNCT discussion).
TMAX	U	REAL8	Transfer time, Δt, in tau.
WIRL	U	LOGIC4	Primer-origin-proximity step size control logical indicator.
ALTAU	U	REAL8	Propulsion-time adjoint variable, λ_T.
ANGLE	SUA	REAL8	Travel angle, θ_t, in radians.
COAST	SU	LOGIC4	Indicator for coast or thrust phase.
CONSP	U	REAL8	Speed conversion factor, from AU/tau to meters/second.
CONTM	U	REAL8	Time conversion factor, tau to days.
COPHI	S	REAL8	Cosine of fixed thrust angle, $\cos \phi_{fixed}$.
DBETA	SUA	REAL8	Computation step size, $\Delta\beta$ (increment of the trajectory independent variable).
DPOWR	U	REAL8	$q \partial\gamma/\partial r$.
ERODE	U	LOGIC4	Power degradation option indicator.
ERROR	S	LOGIC4	Program master error indicator.
FIRST	S	LOGIC4	Indicator for being at initial time of current trajectory segment.
JCMAX	U	INTGR4	Maximum value of JC, corresponding to the end of the current trajectory segment.
PCURV	S	LOGIC4	Indicator for condition in which solar panels are in position to receive maximum power, when degradation option is invoked.

TAP-7

TAP EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
PMDOT	SUA	REAL8	Primer-magnitude time derivative, $\dot{\lambda}$.
SIPHI	S	REAL8	Sine of fixed-thrust-angle, $\sin \phi_{fixed}$.
STEP1	U	REAL8	Thrust-phase computation step-size, Δu.
STEP2	U	REAL8	Coast-phase computation step-size, $\Delta \beta$.
TDELV	SU	REAL8	Time of deep space velocity impulse, in tau.
TRACK	U	LOGIC4	Indicator for trajectory long block print-out (at each computation step).
BALLIS	S	LOGIC4	Indicator that trajectory segment is all-ballistic (pure coast).
DELTAV	SU	REAL8	Magnitude of deep space velocity impulse, in m/sec.
DENSIT	U	REAL8	Power density, d, in AU^{-2}.
DPDMAX	U	REAL8	Absolute maximum of $\partial\gamma/\partial d$, used in computation of $f_{ch\,2}$.
DPOWDD	U	REAL8	$q\,\partial^2\gamma/\partial d^2$.
DWITCH (2)	S	REAL8	$\dot{\sigma}$, defined similarly to R(2).
FIXTAU	U	LOGIC4	Indicator for non-zero $\dot{\lambda}_{\tau}$.
FIXTHR	U	LOGIC4	Indicator for fixed thrust-angle.
JUMPED	S	LOGIC4	Indicator for discontinuous primer derivative, set in subroutine TAP.
NPHI20	SU	INTGR4	Array location value corresponding to the current value of the fixed thrust cone angle. Currently, only one value for fixed thrust angle is allowed along a given trajectory, and NPHI20 is set equal to 21 in subroutine TAP.

TAP EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
NPRINT	UA	INTGR4	Printout amount selection indicator.
NSTEP1	SU	INTGR4	Total number of computation steps associated with thrusting flight, for the current trajectory.
NSTEP2	SU	INTGR4	Total number of computation steps associated with coasting flight, for the current trajectory.
QERODE	S	LOGIC4	Indicator for either the final (summary) trajectory of a given case or solar array radiation damage degradation.
REGION	SU	LOGIC4	Indicator for spacecraft solar proximity; demarks two possible regions in space, separated by sphere about sun of specified radius, at which power function (or its derivative) has a corner.
RTSWIT	U	REAL8	Critical solar distance corresponding to a special point in the solar power curve, in AU.
SWITCH (2)	S	REAL8	Thrust switch function, σ, defined similarly to R(2).
TAUPOW	U	REAL8	Negative inverse of characteristic degradation time, $-1/\tau_d$, in tau^{-1}.
TCHECK (41)	S	REAL8	Array of time values, isolated by subroutine CHECK, in tau.
TUDFLG	U	LOGIC4	Indicator for two-dimensional trajectory simulation (motion in the xy plane).

CHART TITLE - SUBROUTINE TAP

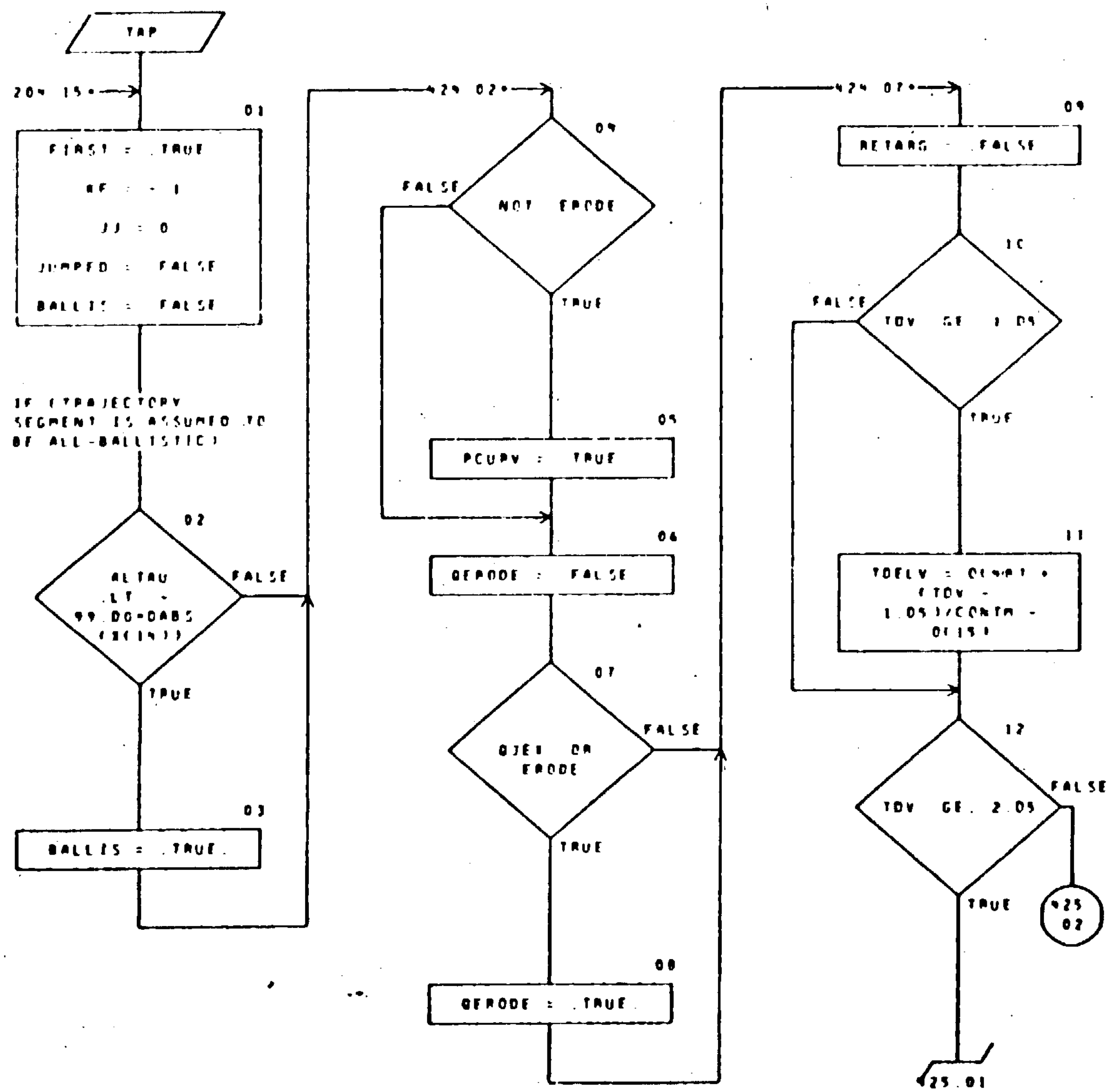
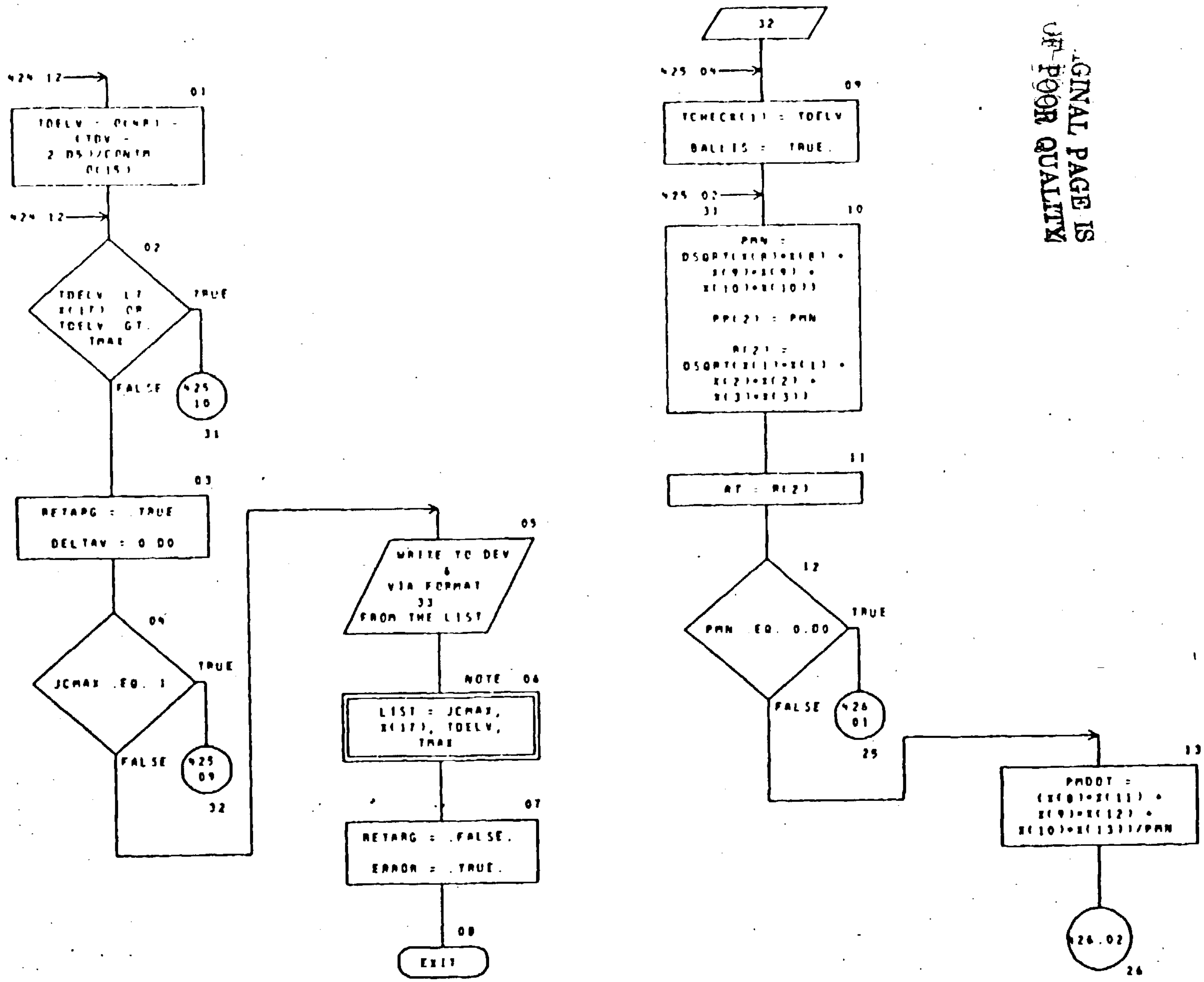


CHART TITLE - SUBROUTINE TAP



TAP-11

CHART TITLE - SUBROUTINE TAP

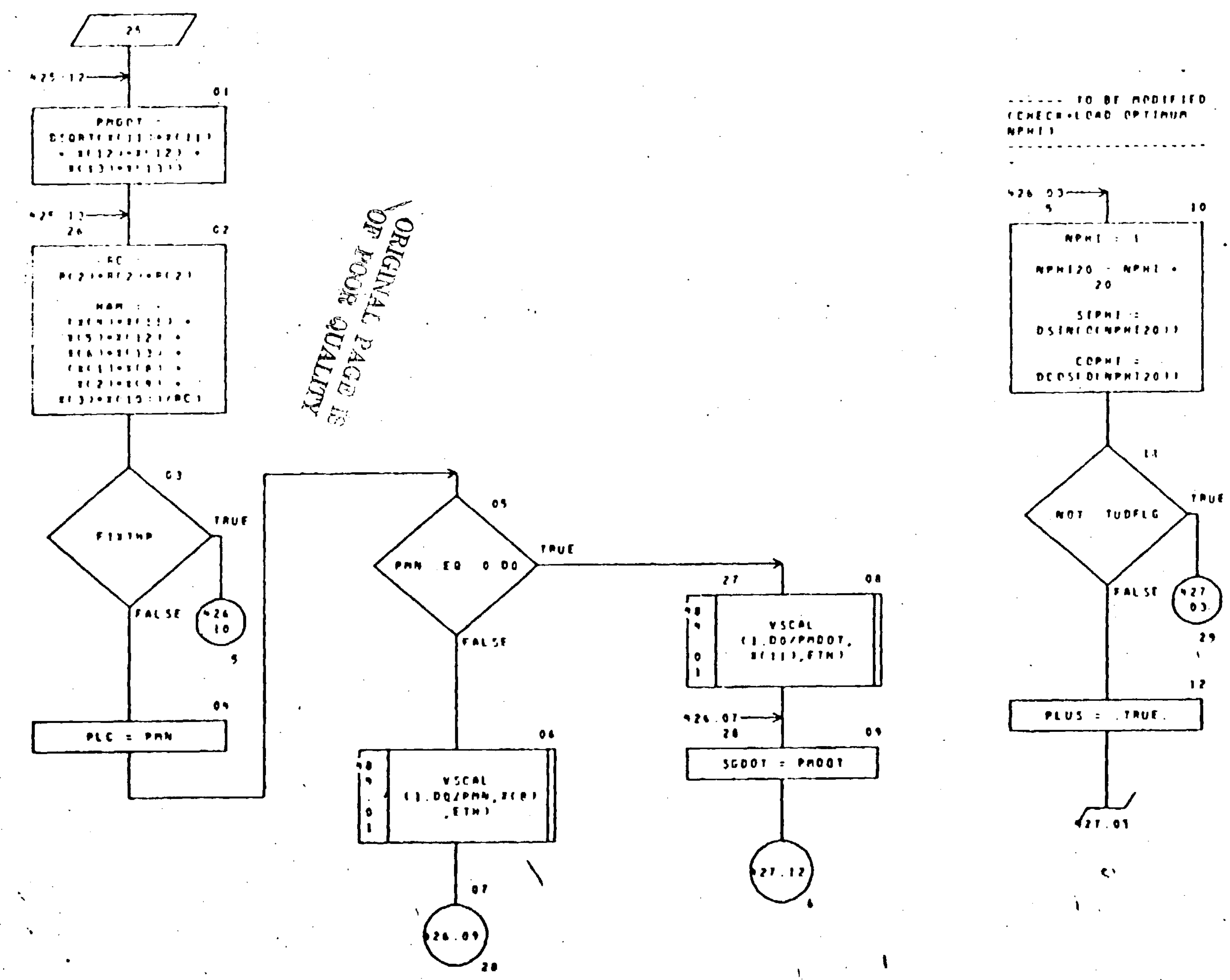


CHART TITLE - SUBROUTINE TAP

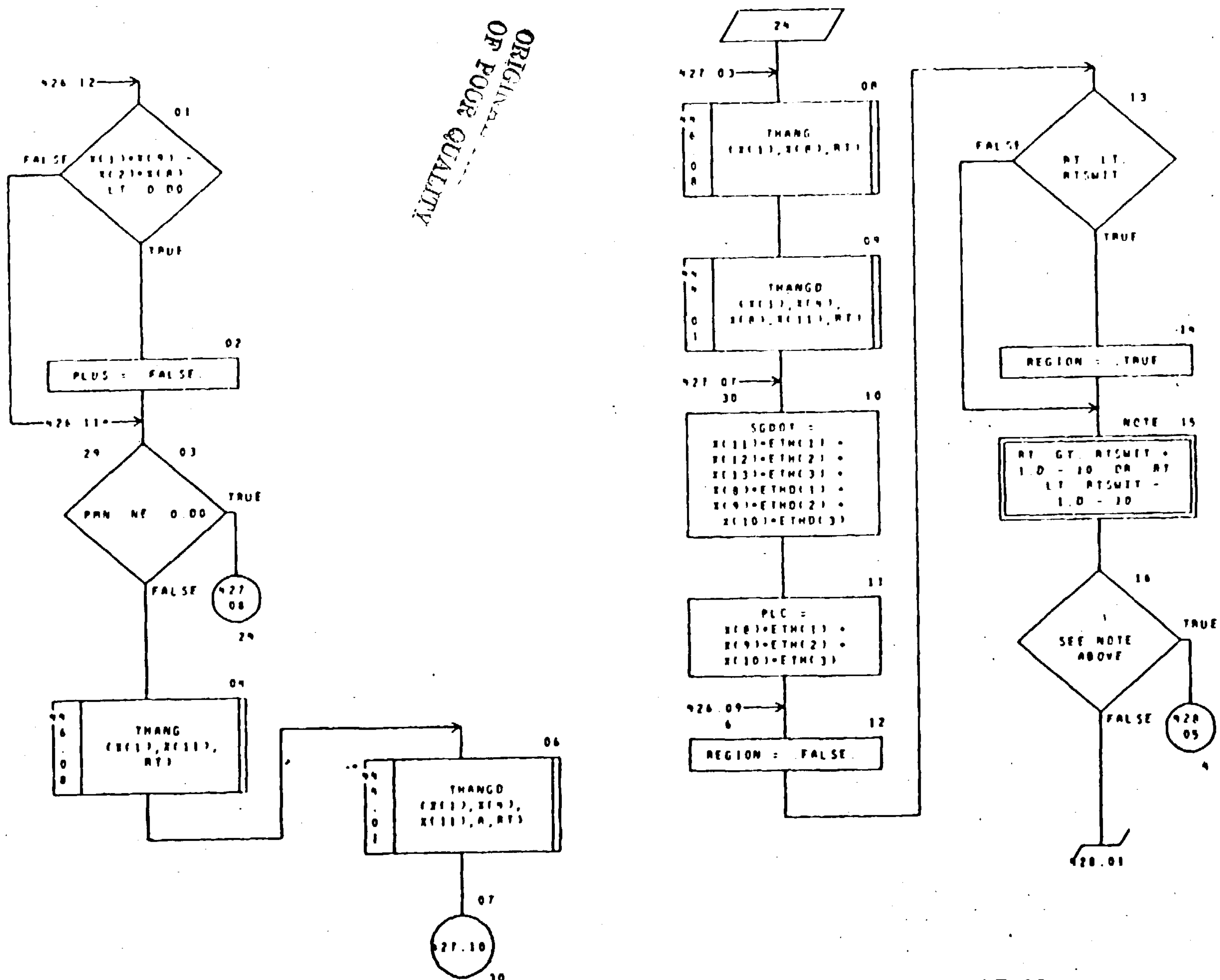


CHART TITLE - SUBROUTINE TAP

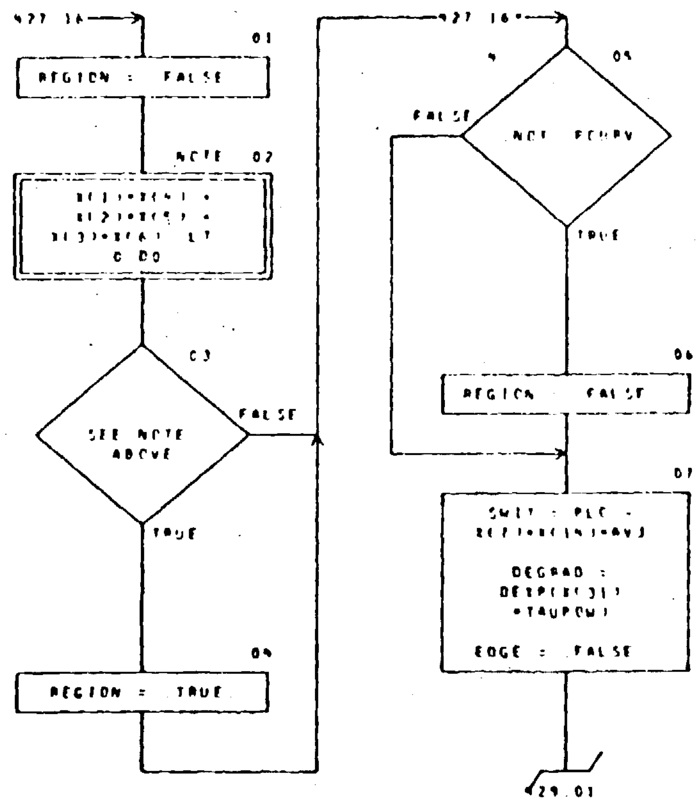
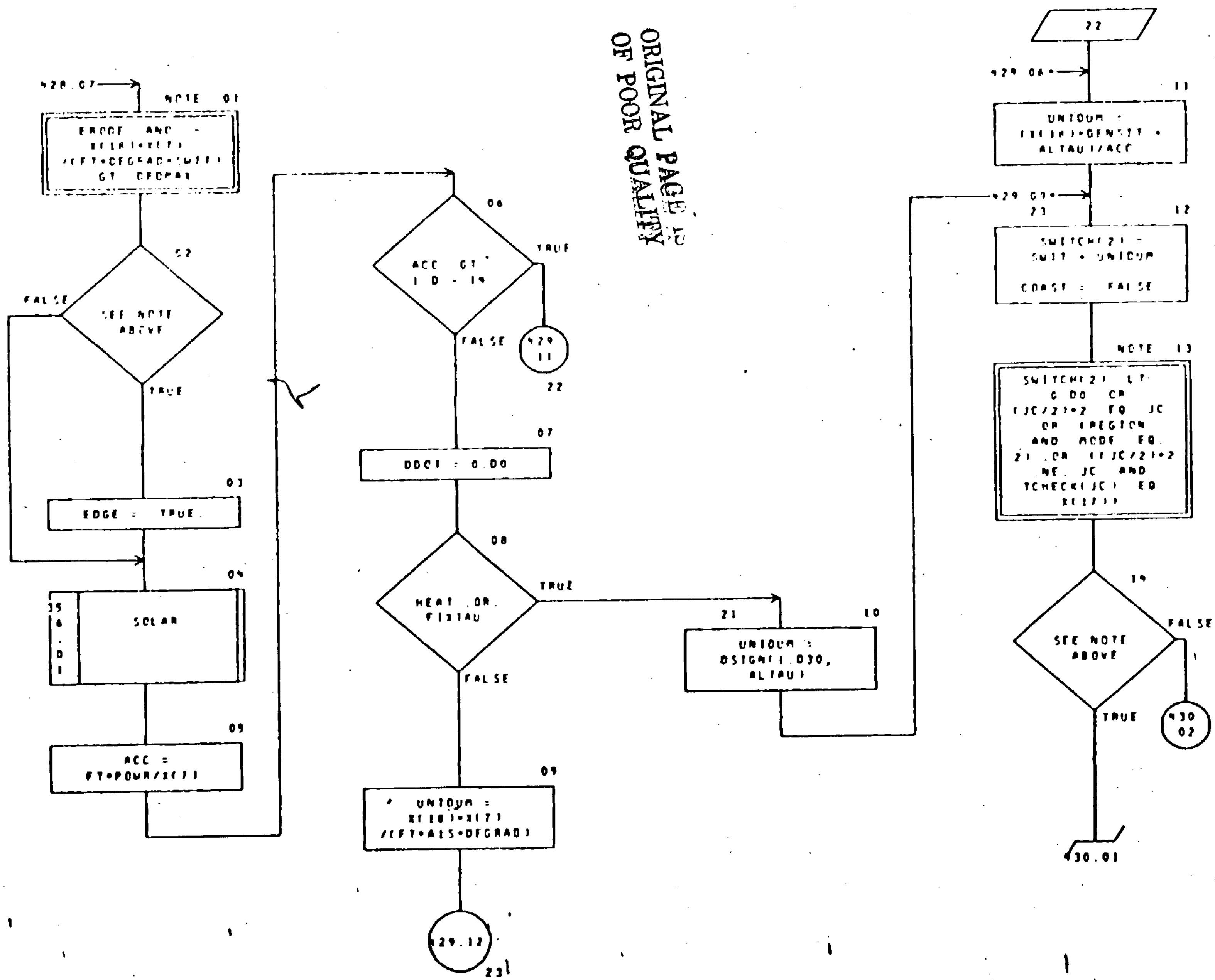


CHART TITLE - SUBROUTINE TAP



ORIGINAL PAGE IS
OF POOR QUALITY

TAP-15

CHART TITLE - SUBROUTINE TAP

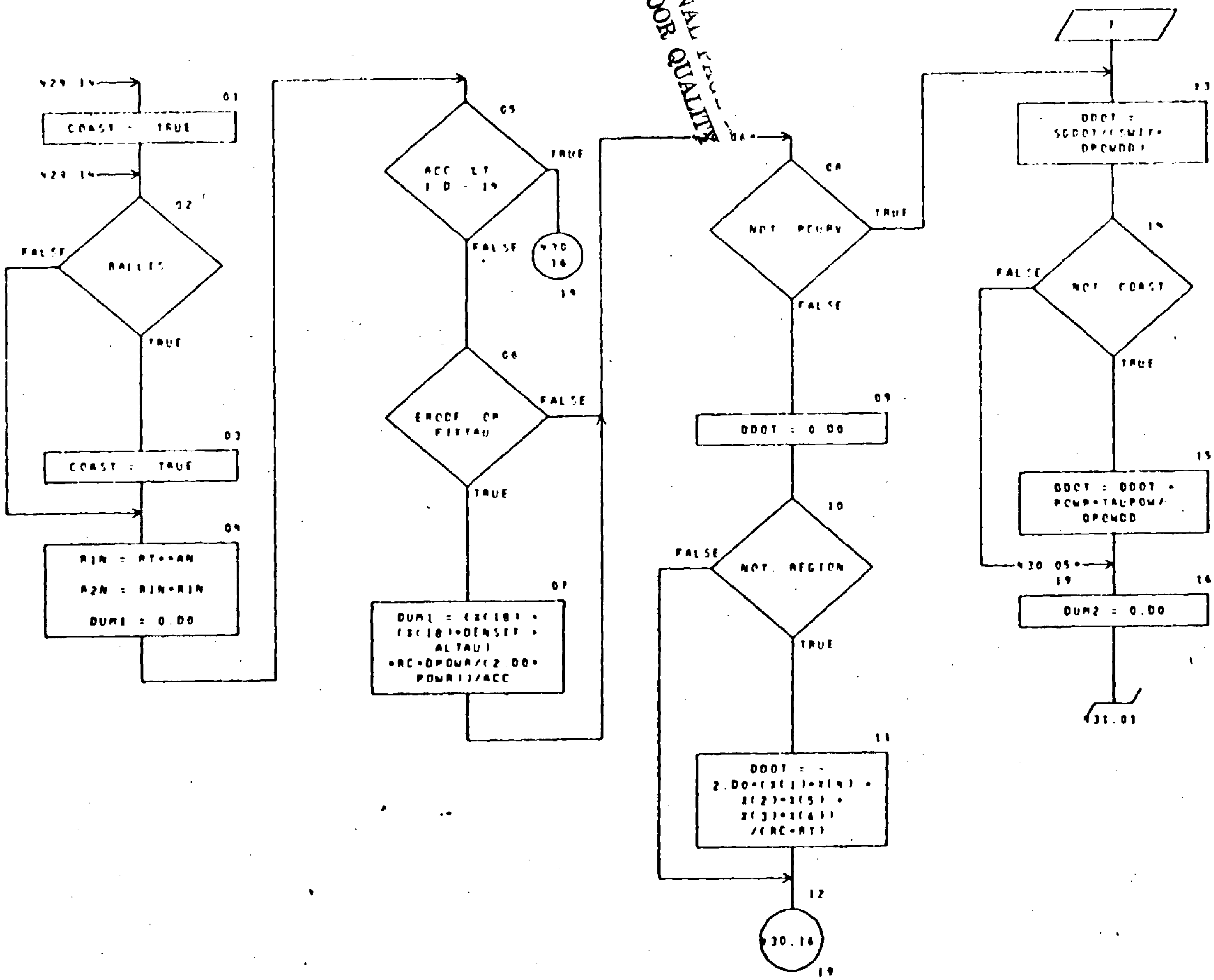
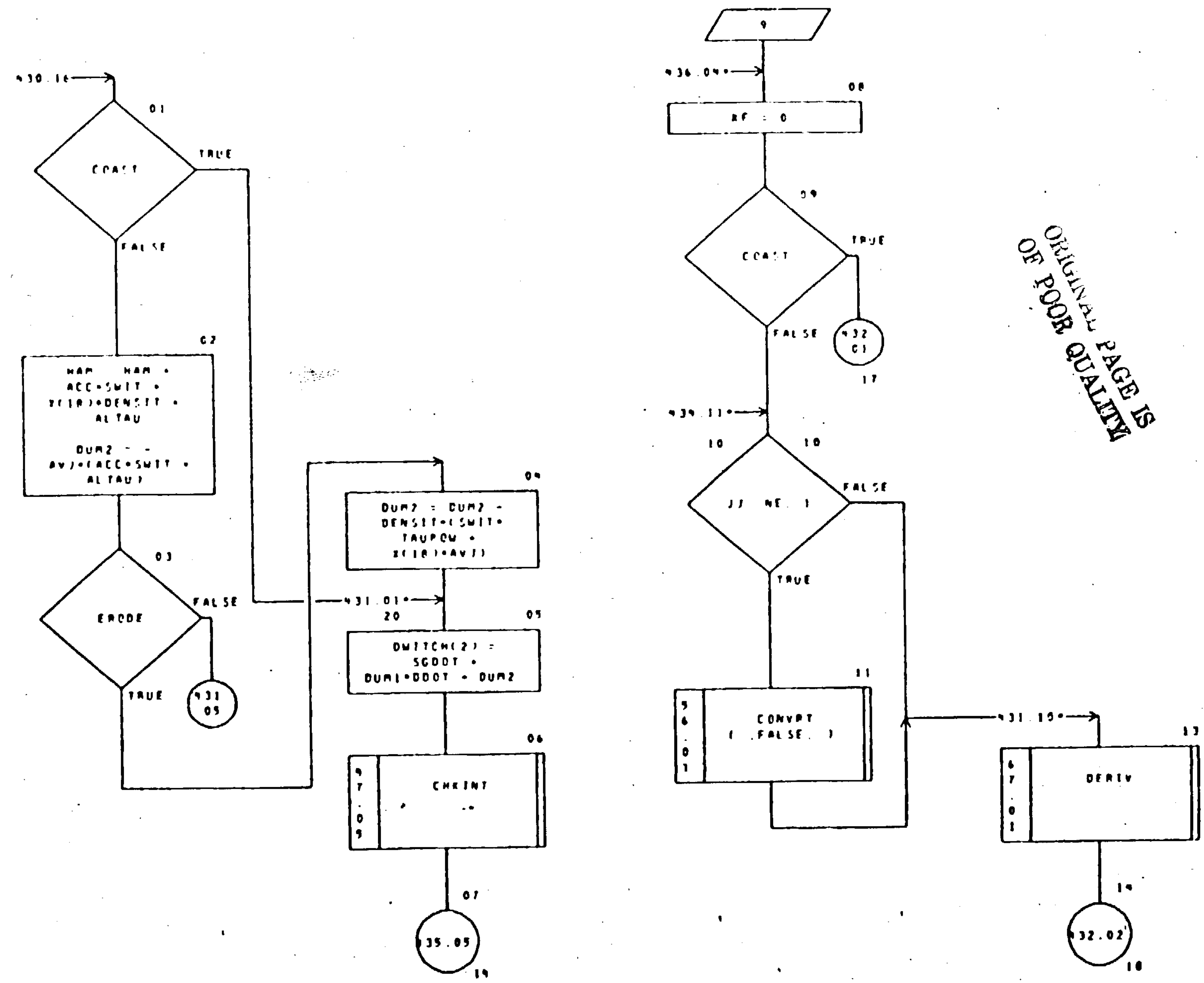


CHART TITLE - SUBROUTINE TAP



ORIGINAL PAGE IS OF POOR QUALITY

TAP-17

CHART TITLE - SUBROUTINE TAP

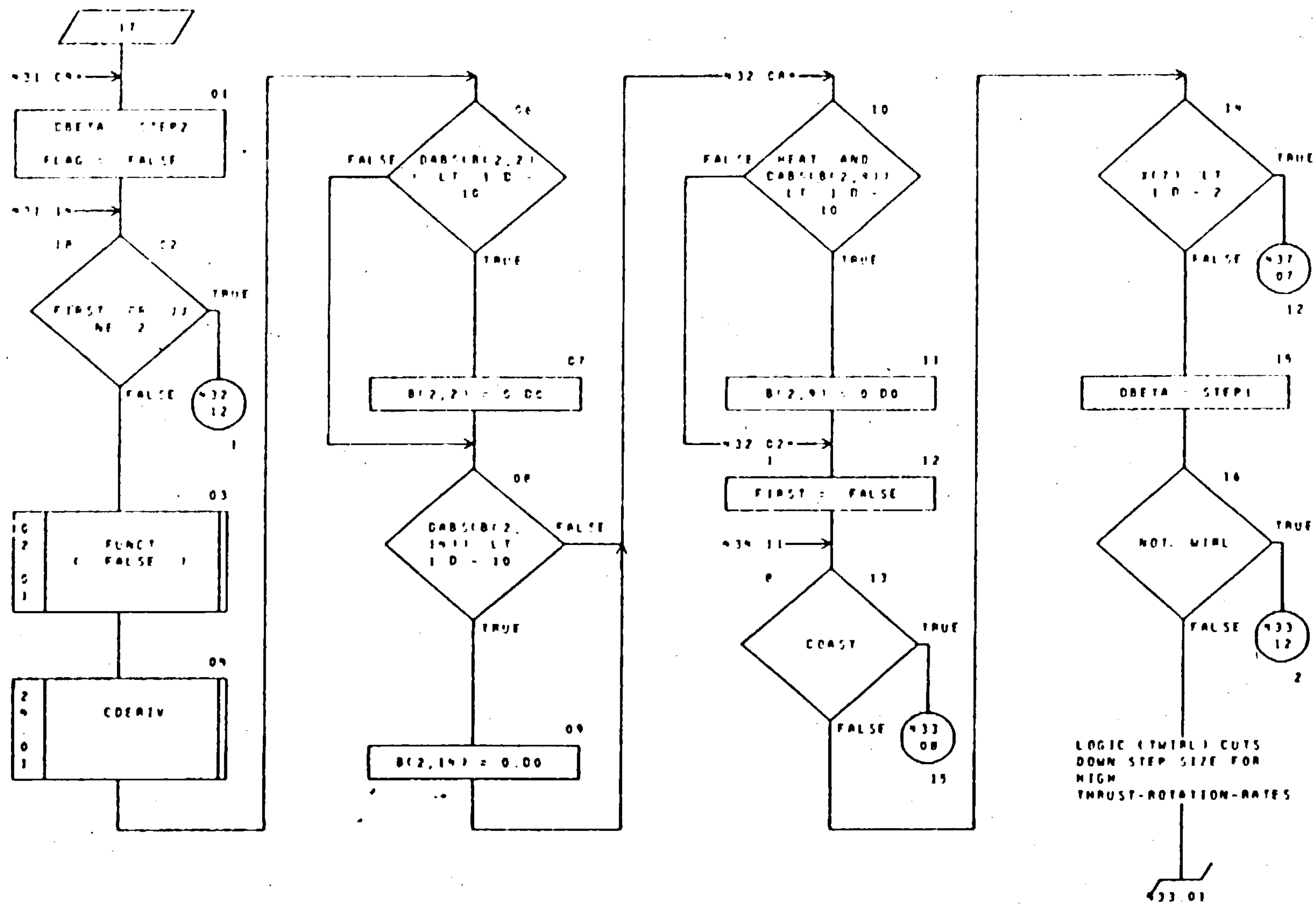
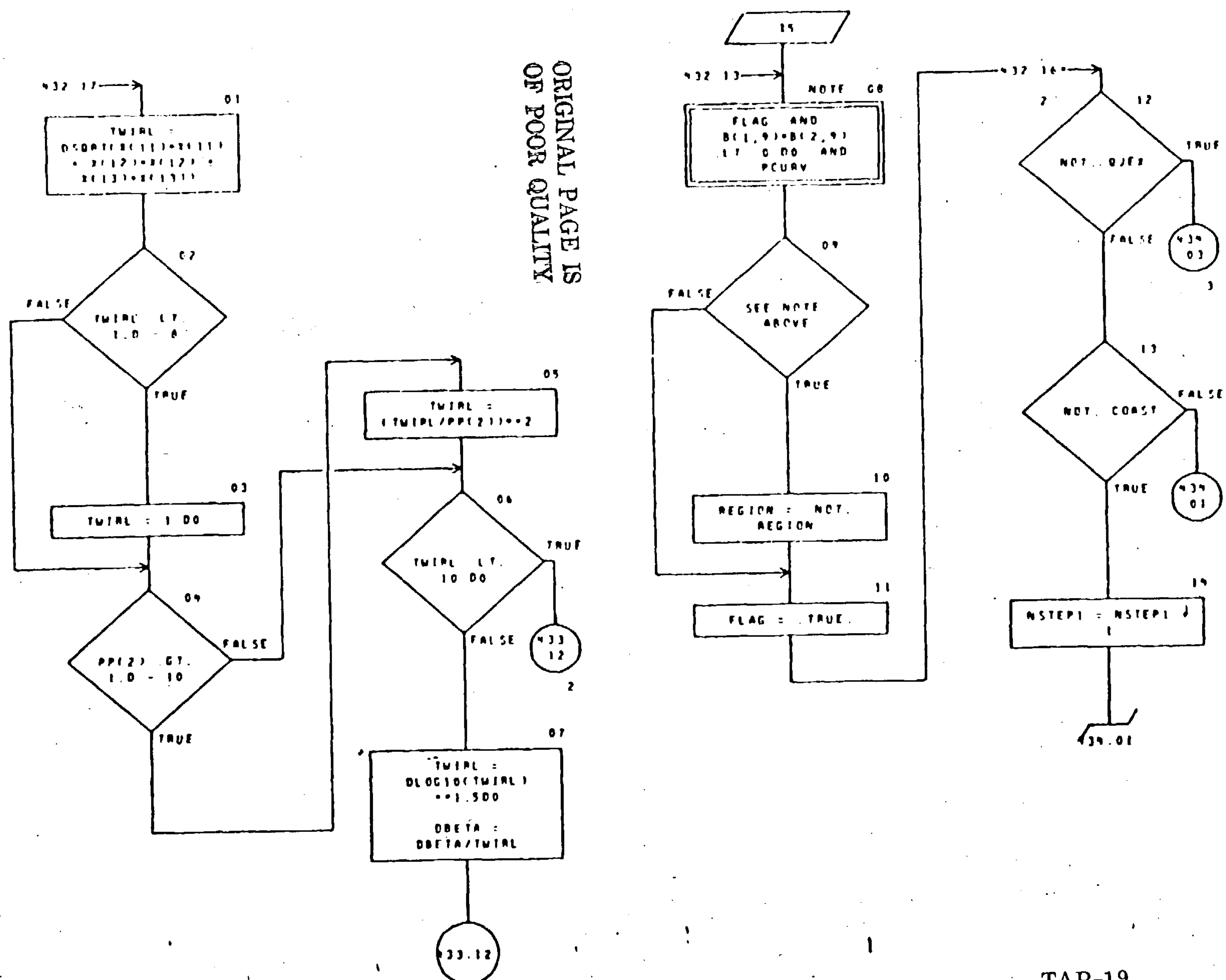


CHART TITLE - SUBROUTINE TAP

ORIGINAL PAGE IS
OF POOR QUALITY



TAP-19

CHART TITLE - SUBROUTINE TAP

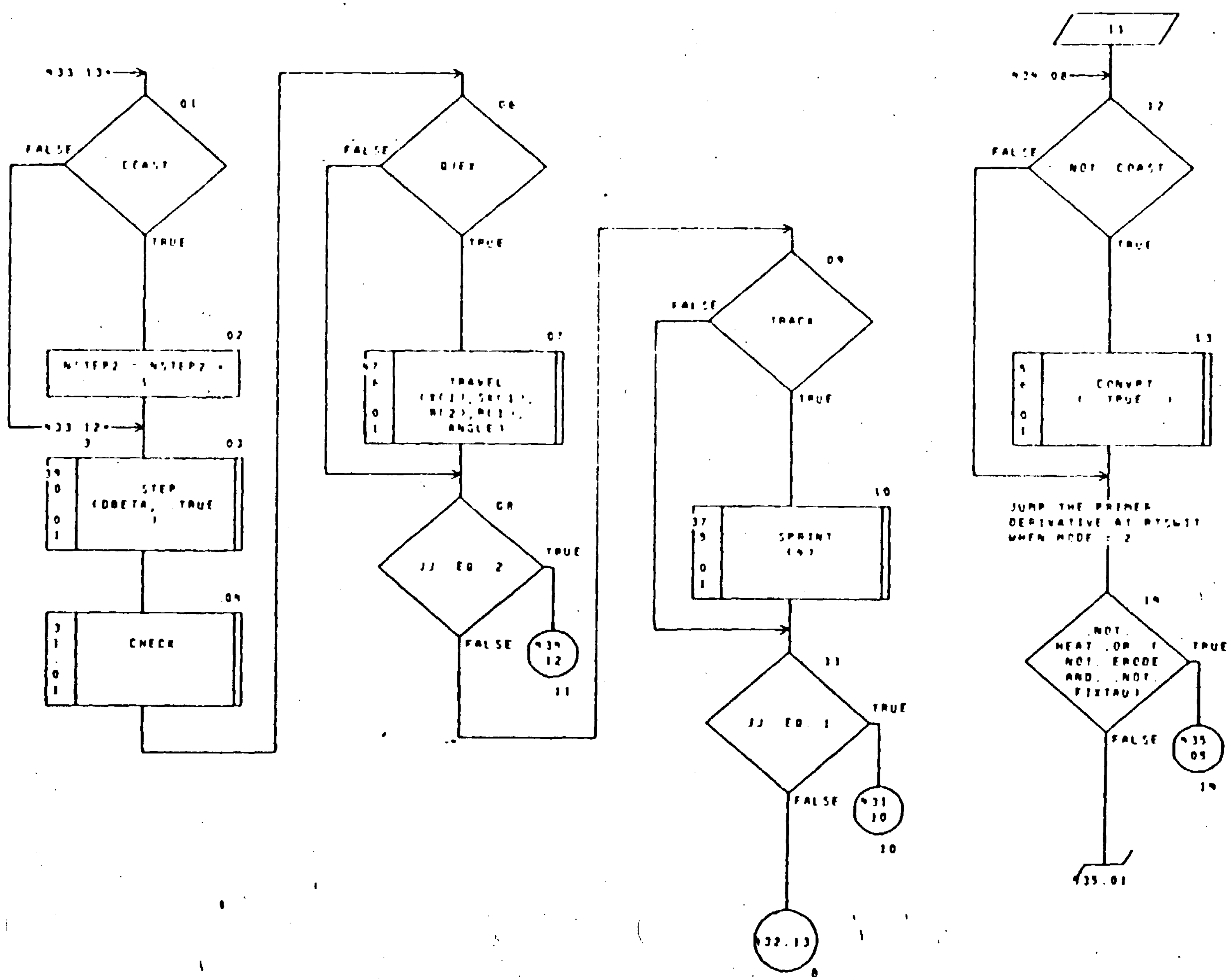
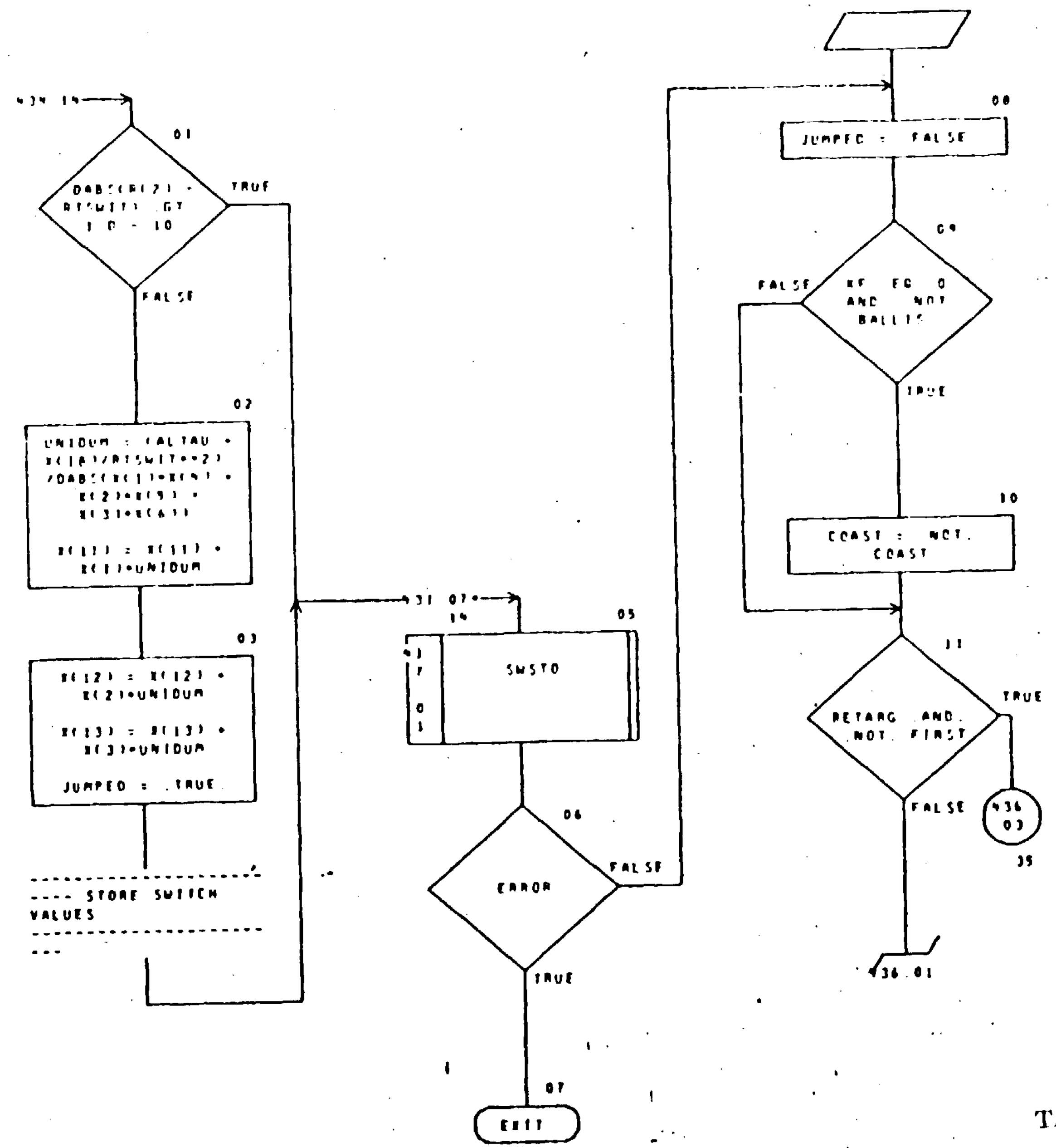


CHART TITLE - SUBROUTINE TAP



TAP-21

CHART TITLE - SUBROUTINE TAP

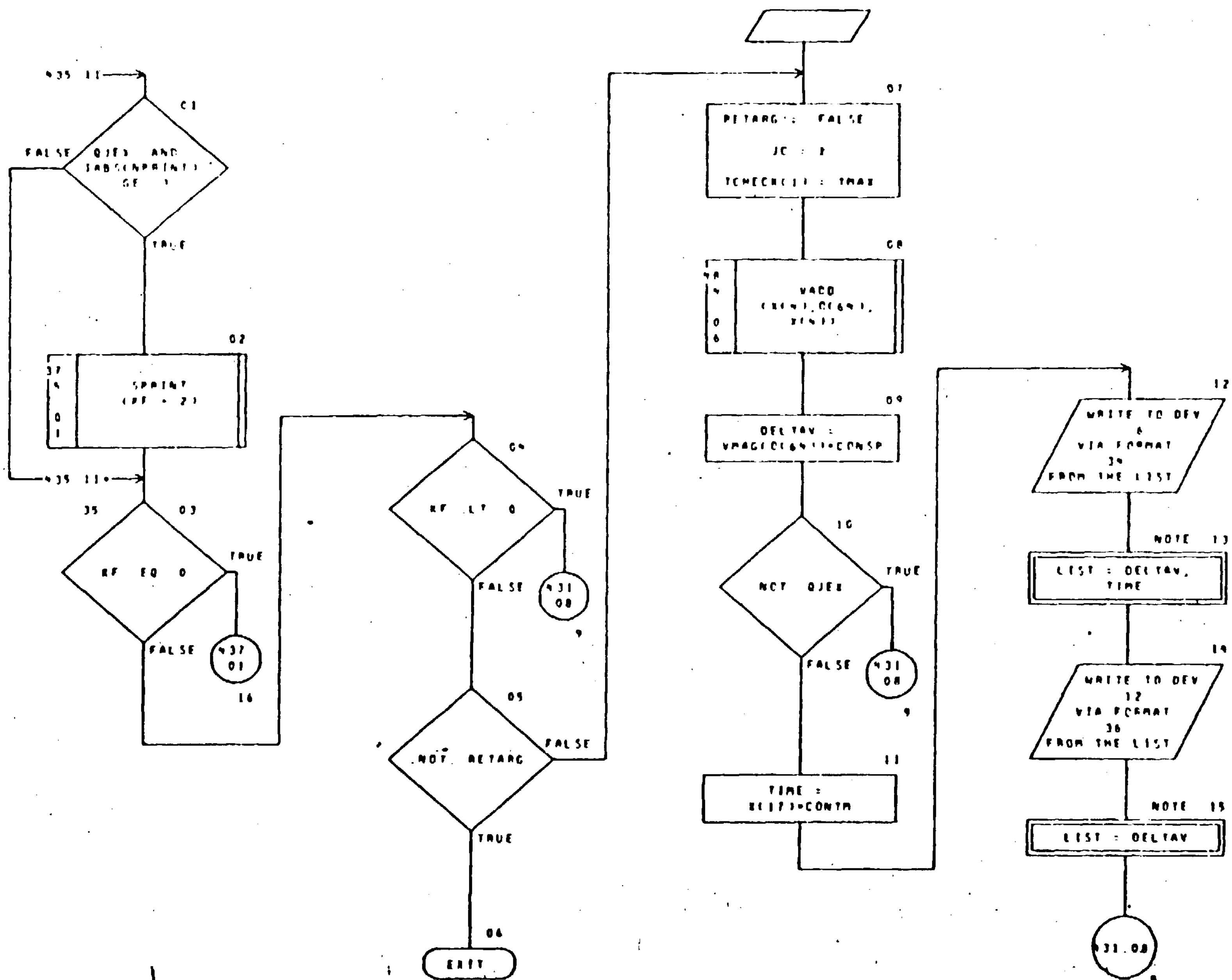
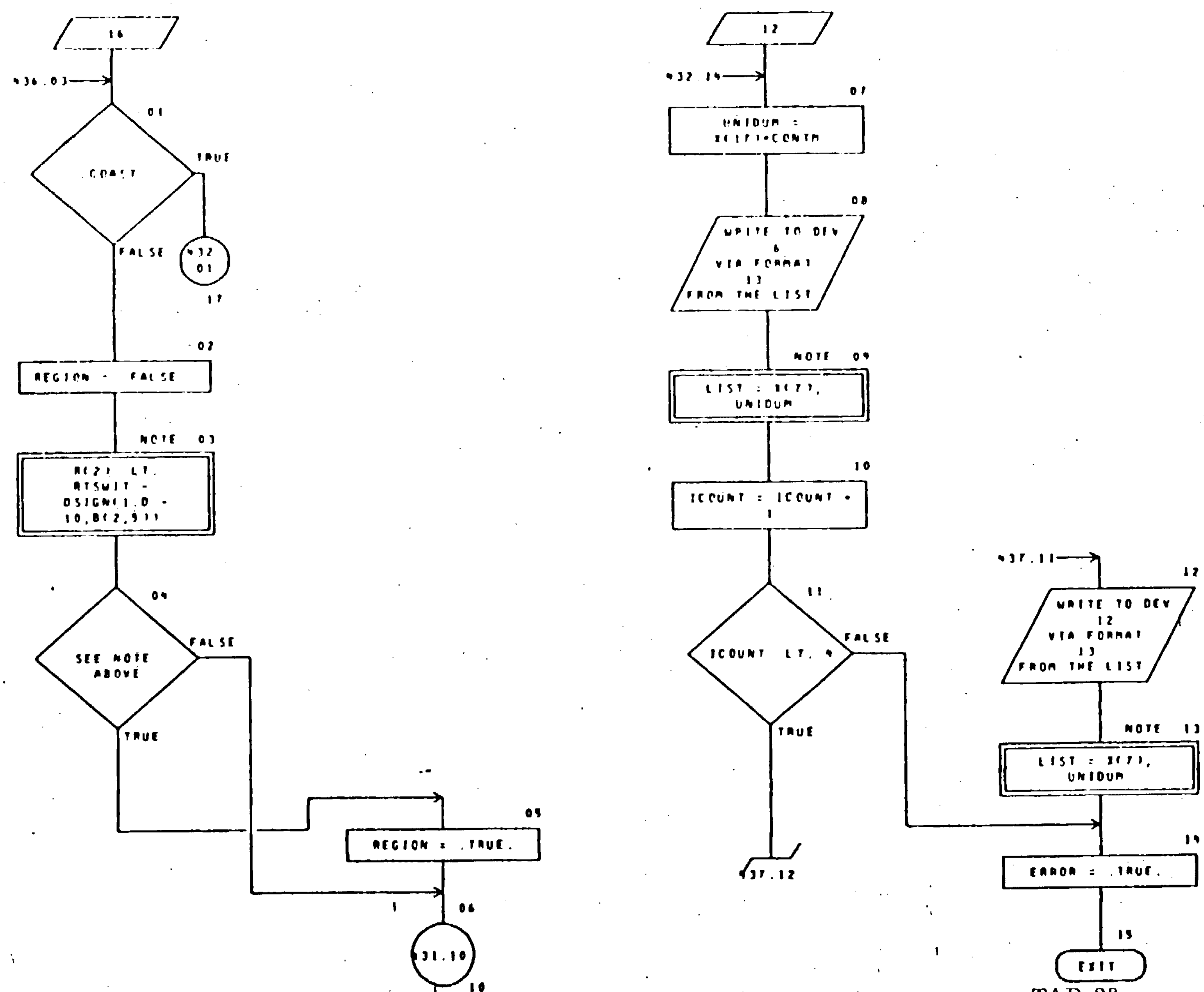


CHART TITLE - SUBROUTINE TAP



TAP-23

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL FIXTHR,FLAG,COAST,REGION,OJEX,TRACK,FIXTAU,ERROR,FIRST
,WIRL,ERODE,PCURV,EDGE,HEAT,DEPODE,JUMPED,PLUS,TUDFLG,BALLIS
,RETARG
DIMENSION A(3)
COMMON /REAL8/ R01(11),FT,R02(73),DBETA,STEP1,STEP2,TMAX,R03(11),
TDV,
TDELV,R18(51),ANGLE,R04(19),A15,R25,
         ALTAU,R05(6),RTSWIT,R06(17),R(2),RR(2),SWITCH(2),DWITCH(2)
,R07(92),AN,RIN,R24,R08(12),CONTP,CONSP,R09(7),DELTAV,R26(290),
TCHECK(41),R21(219),
ETH(3),ETHD(3),R10(9),SWIT,SIPHI,COPHI,PLC,R11,HAM,R12(289),
DPDMAX,R29(6),                   PMDOT,R13(2),PMN,RIN,RT,RIS,RC,
R20(4),POWR,DPOWR,TAUPOW,R16(12),AVJ,R17(6),DENSIT,
R22(5),DPOWDD,R23(17),SX(55),X(50),R19(750)
COMMON /INTGR4/ I01,MODE,I02(39),NPRINT,I03(21),JJ,KF,I04(260),
NPHI,NPHI20,I05(2),NSTEP1,NSTEP2,JC,JCMAX,I06(467)
COMMON /LOGIC4/ ERROR,L01(7),TUDFLG,L09,FIXTHR,L02(7),REGION,
ERODE,L03(4),
TRACK,L04,OJEX,FIXTAU,HEAT,L05,FIRST,COAST,DEPODE,L06(2),WIRL,
PCURV,L07,EDGE,PLUS,JUMPED,L08(2),BALLIS,L10(956)
COMMON /ITERAT/ B01(700),D(70),B02(210)
COMMON /EXTREM/ E01(2640),B(2,30)
DATA ICOUNT /0/
DATA A /3*0.D0/
```

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
33      FORMAT(1H0,39HERROR  JCMAX.NE 1 IN *IDV* OPTION.,
        I10,4X1P3D16.8/1H0)
34      FORMAT(1H0,15HDEEP SPACE BURNF9.1,17H METERS/SECOND AT
        F9.2,5H DAYS/1H0)
36      FORMAT(1H ,15HDEEP SPACE BURNF9.1,4H M/S)
13      FORMAT(1H ,12HMASS RATIO =D10.3,23H, VANISHING NEAR TIME =
        F7.1,6H DAYS )
```

ORIGINAL PAGE IS OF POOR QUALITY

PRECEDING PAGE BLANK NOT FILMED

Name: TAPSET

Calling Argument: SET, TIME, NPLAN1, NPLAN2, LEGADD, SKIP

Referenced Sub-programs: None

Referenced Commons: INTGR4, LOGIC4, REAL8

Entry Points: None

Referencing Sub-programs: MORE, SWTRAJ

Discussion: TAPSET is used to initialize subroutine TAP in preparation for generating a ballistic trajectory-segment. The trajectory segment is maintained ballistic (all coast) by ensuring that the thrust switching function is always negative, and this is accomplished by setting the propulsion-time adjoint variable to a relatively large negative number,

$$\lambda_\tau = -100\ |\lambda_\nu|\ ,$$

where λ_ν is the current value of the mass-ratio adjoint variable. After the trajectory segment has been generated, TAPSET is again called and performs a restoring operation, in which altered parameters are set to their original values.

Messages and printouts: If a multiple-target swingby continuation analysis is attempted which involves more than five swingbys, the MOPTX array will be exceeded, and therefore the computer run is terminated after the following message is printed on units 6 and 12:

INVALID TRAJECTORY EXTENSION
MOPTX(n) INVALID IN SUBROUTINE TAPSET
RUN TERMINATED

where n is the number of attempted swingbys.

TAPSET EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
X(50)	SU	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
JC	S	INTGR4	Counter corresponding to the JC^{th} specified time-function value (i.e., the time) isolated thus far on the current trajectory segment.
LEG	SU	INTGR4	Counter indicating the current trajectory-segment.
NSW	SU	INTGR4	Counter indicating the current number of thrust switch points along the whole trajectory; includes trajectory-segment endpoints.
SET	UX		Indicator for initialization or restoration.
BETA	SU	REAL8	Trajectory independent-variable, β.
SKIP	UX		Indicator for bypassing the restoration and re-initialization of the trajectory dependent-variables, which is desired for continuing the trajectory during multiple swingby simulations.
TIME	UX		Time elapsed between primary-target passage and current post-swingby target intercept, in days.
TMAX	SU	REAL8	Time at the end of the current trajectory-segment, elapsed since the start of the trajectory, in tau (integration stopping condition).
ALTAU	SU	REAL8	Propulsion-time adjoint variable, λ_τ.
ANGLE	SU	REAL8	Travel angle, θ_t, in radians.
CONTM	U	REAL8	Time conversion factor, tau to days.

TAPSET EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ERODE	SU	LOGIC4	Power degradation option indicator.
JCMAX	S	INTGR4	Maximum value which JC may attain, corresponding to the end of the current trajectory segment.
MOPTX(5)	S	INTGR4	The target-numbers of the successive intermediate targets, and, in this routine, also of the successive swingby planets and final target.
MOPT3	SU	INTGR4	Planet-number of primary target.
NSPEC	SU	INTGR4	Master array index (and counter) for storage arrays associated with the Extremum Table of Selected Functions.
XMASS(7)	SU	REAL8	General mass array; XMASS(6) is the reference power, p_{ref}, in either watts or kilowatts.
COMANG	SU	REAL8	Communication angle at time of primary-target intercept, in degrees.
COMDIS	SU	REAL8	Communication distance at time of primary-target intercept, in AU.
FIXTAU	SU	LOGIC4	Indicator for non-zero propulsion-time adjoint variable, λ_{τ}.
LEGADD	UX		Number of whole trajectory-segments (legs) which have been generated beyond the primary target.
LEGMAX	SU	INTGR4	Current maximum number of trajectory segments (legs).
NPLAN1	UX		Current swingby-planet selector.
NPLAN2	UX		Current post-swingby target selector.

TAPSET EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
TANGLE	SU	REAL8	Elapsed ecliptic longitude, in radians.
TCHECK (41)	S	REAL8	Array of time values, each of which is isolated by subroutine CHECK, in tau.

CHART TITLE - SUBROUTINE TAPSET(ISET,TIME,NPLAN1,NPLAN2,LEGADD,SKIP)

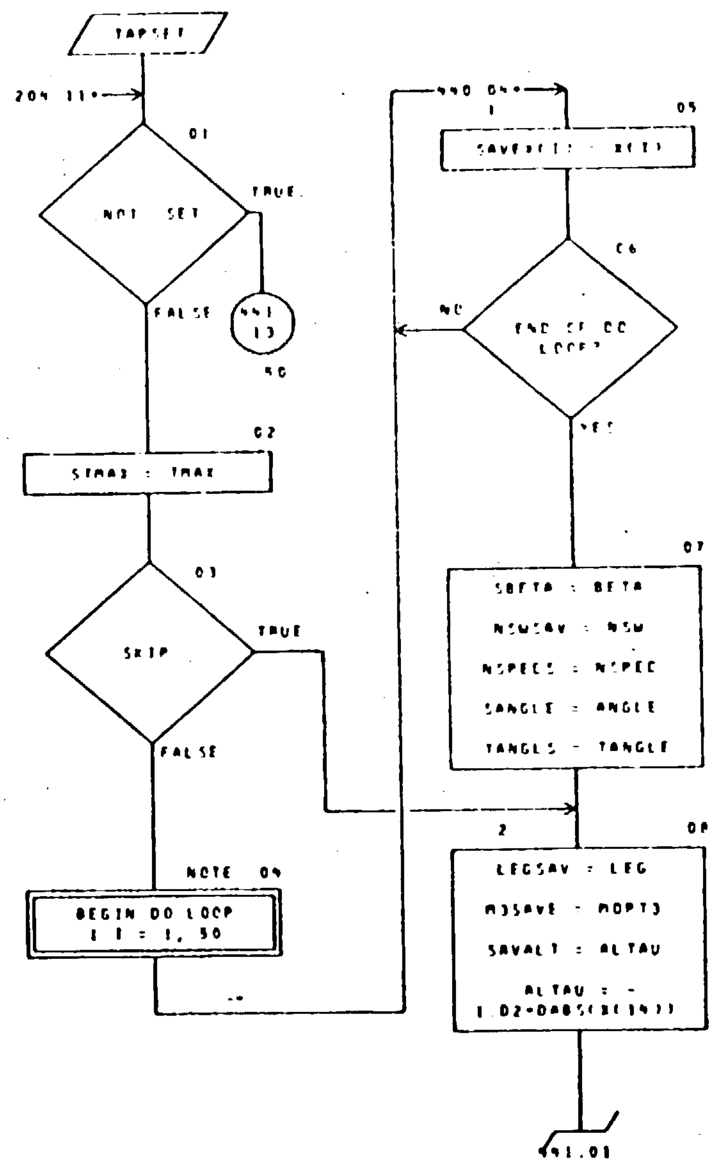


CHART TITLE - SUBROUTINE TAPSET(ISET,TIME,NPLAN1,NPLAN2,LEGADD,SKIP)

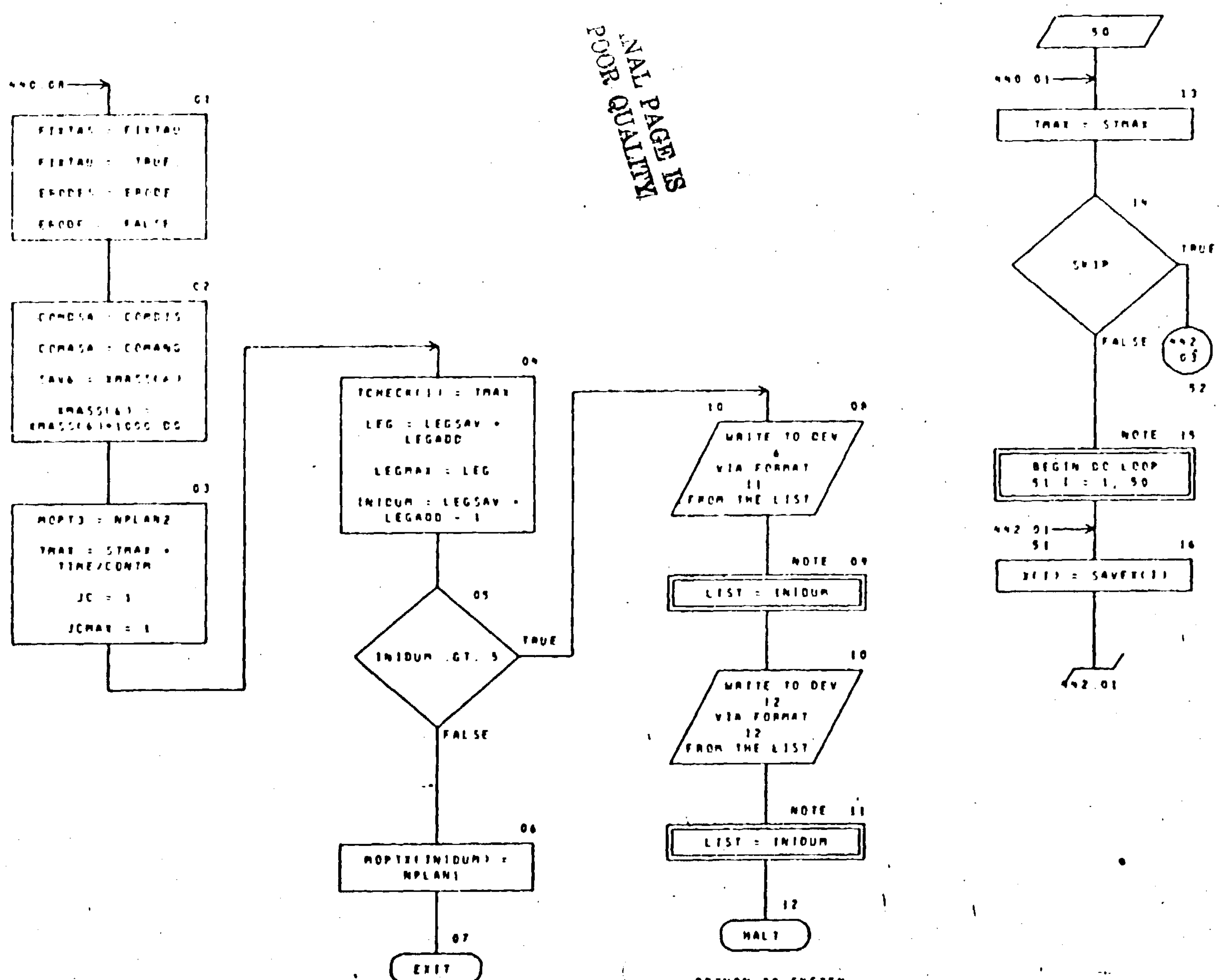
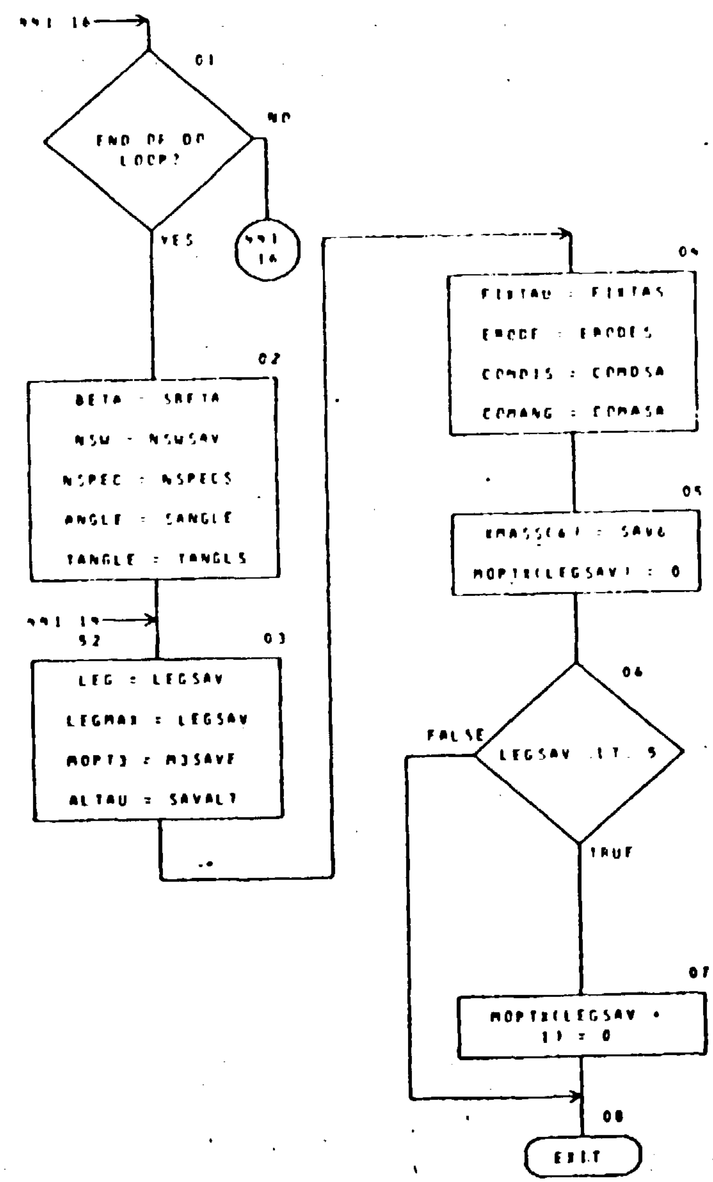


CHART TITLE - SUBROUTINE TAPSET(ISET,TIME,NPLAN1,NPLAN2,LEGADD,SKIP)



TAPSET-7

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL SET,SKIP,FIXTAU,FIXTAS,ERODE,ERODES
      DIMENSION SAVEX(50)
      COMMON /REAL8/ R01,XMASS(7),R02(80)                ,TMAX,R04(64),
      ANGLE,R05(9),TANGLE,R06(11),ALTAU,R07(4),CONDIS,COMANG,
      R08(133),CONTM,R09(249),TCHECK(41),R10(644),X(50),R11(51),
      BETA,R12(648)
      COMMON /INTGR4/ I01(29),MOPT3,I02(12),NSPEC,I03(36),NSM,I04(51),
      LEG,LEGMAX,MOPTX(5),I05(198),JC,JCMAX,I06(667)
      COMMON /LOGIC4/ L01(19),ERODE,L02(7),FIXTAU,L03(472)
11    FORMAT(1H1,36HINVALID TRAJECTORY EXTENSION. MOPTX(I3,
      47H) INVALID IN SUBROUTINE TAPSET. RUN TERMINATED )
12    FORMAT(1H ,28HINVALID TRAJECTORY EXTENSION/1H ,
      6HMOPTX(I3,30H) INVALID IN SUBROUTINE TAPSET/1H ,
      14HRUN TERMINATED)
```

<u>Name</u>: THANGD

<u>Calling Argument</u>: R, RD, P, PD, RM for THANGD
R, P, RM for THANG

<u>Referenced Sub-programs</u>: UNITD for THANGD
None for THANG

<u>Referenced Commons</u>: LOGIC4, REAL8

<u>Entry Points</u>: THANG

<u>Referencing Sub-programs</u>: CDERIV, FUNCT, TAP for THANGD
FUNCT, TAP for THANG

<u>Discussion</u>: Entry point "THANG" is a contraction of "thrust angle", and "THANGD" stands for "thrust angle derivative". However, it is actually the thrust unit vector* e_t and its time-derivative $\dot{e}_t$ (and not the so-called thrust angle, or thrust cone angle, ϕ) which are computed by this routine, and only for the case in which ϕ is constrained to a "fixed" value along the trajectory, where ϕ is the angle between the spacecraft's radius vector R and e_t. When ϕ is <u>not</u> fixed, e_t lies along the primer vector Λ, and when ϕ <u>is</u> fixed, e_t is computed to be as close to Λ as the fixed-ϕ constraint permits, the former case being a special case of the latter, with the (Λ: e_t) proximity condition being dictated by the Maximum Principle of optimal control theory. The computations of this routine are concerned with the fixed-ϕ case only. The thrust unit vector e_t is computed by entry point THANG and $\dot{e}_t$ is computed by THANGD.

Since e_t moves in a cone (of half-angle ϕ) about R, the closest e_t can get to Λ is when e_t lies in the instantaneous plane defined by R and Λ. The unit thrust vector may therefore be expressed as

$$e_t = \frac{R}{|R|} \cos \phi + \frac{R \times (\Lambda \times R)}{|R \times (\Lambda \times R)|} \sin \phi$$

*The bar($\bar{\ }$) is dropped from $\bar{e}_t$ in this subroutine description.

Letting (defining)

$$\hat{R} = \frac{R}{|R|}$$

and

$$\hat{T} = \frac{R \times (\Lambda \times R)}{|R \times (\Lambda \times R)|}$$

where $\hat{T}$ is simply a unit vector orthogonal to R in the (R, Λ) plane, then

$$e_t = \hat{R} \cos \phi + \hat{T} \sin \phi$$

which expresses e_t in terms of orthonormal vectors in the (R, Λ) plane. This may be rewritten

$$e_t = \frac{\hat{R} + \hat{T} \tan \phi}{|\hat{R} + \hat{T} \tan \phi|} \quad \text{if } \phi \neq \pi/2 \text{ ;}$$

$$e_t = \frac{\hat{T} + \hat{R}/\tan \phi}{|\hat{T} + \hat{R}/\tan \phi|} \quad \text{if } \phi \neq 0 \text{ or } \pi \text{ .}$$

The derivative $\dot{e}_t$ is then computed as follows. Defining

$$Q_1 = \frac{\dot{\hat{R}} + \dot{\hat{T}} \tan \phi}{|\hat{R} + \hat{T} \tan \phi|}$$

then

$$\dot{e}_t = Q_1 - (e_t \cdot Q_1)\, e_t$$

and if

$$Q_2 = \frac{\dot{\hat{T}} + \dot{\hat{R}} /\tan \phi}{|\hat{T} + \hat{R}/\tan \phi|}$$

then

$$\dot{e}_t = Q_2 - (e_t \cdot Q_2)\, e_t$$

in which

$$\dot{\hat{T}} = Q_3 - (\hat{T} \cdot Q_3)\, \hat{T}$$

and

$$Q_3 = \frac{R \times (\Lambda \times \dot{R}) + R \times (\dot{\Lambda} \times R) + \dot{R} \times (\Lambda \times R)}{|R \times (\Lambda \times R)|}$$

The straightforward differentiations involved above make use of the fact that

$$\cos \phi = \frac{1}{|\hat{R} + \hat{T} \tan \phi|}$$

and

$$\sin \phi = \frac{1}{|\hat{T} + \hat{R}/\tan \phi|}$$

are constants. When the simulation is three dimensional, the foregoing expressions for e_t and $\dot{e}_t$ are used to generate these two vectors, which are output from the subroutine.

When the simulation is two dimensional, motion is assumed to be in the xy plane, and the analysis is simplified as follows:

$$e_t = \hat{R} \cos \phi + \hat{T} \sin \phi$$

where $\hat{T}$ is a vector orthonormal to R in the xy plane and is given by

$$\hat{T} = \frac{1}{r} \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}.$$

when the logical indicator PLUS is true and

$$\hat{T} = \frac{1}{r} \begin{pmatrix} y \\ -x \\ 0 \end{pmatrix}$$

when PLUS is false, where $r = |R|$. The logical indicator PLUS is required in two dimensional simulations because the thrust vector e_t lies at one of two discrete positions having an angle ϕ with respect to R, and e_t "flips" discontinuously from one position to the other as Λ sweeps past $\pm R$; however, this e_t switch point must be found by iteration, and PLUS must retain its sense until the iteration is converged, at which time the sense of PLUS is reversed. For computational purposes, define $q = 1$ if PLUS is true and $q = -1$ if PLUS is false, where the sense of PLUS (which determines the position of e_t) is determined (by subroutines CHECK and CDERIV) by examining the position of

THANGD-3

Λ with respect to R, which is accomplished by considering the sign of

$$R \times \Lambda = x\lambda_y - y\lambda_x .$$

Thus $q = 1$ when in the region where $x\lambda_y - y\lambda_x > 0$ and $q = -1$ when in the region $x\lambda_y - y\lambda_x < 0$, and the iteration mentioned above is to isolate the boundary between these two regions. Therefore, the above two expressions for $\hat{T}$ may be combined as

$$\hat{T} = \frac{q}{r} \begin{pmatrix} -y \\ x \\ 0 \end{pmatrix}$$

Then letting Y be defined as

$$Y = \dot{\left(\frac{1}{r}\right)} = -(x\dot{x} + y\dot{y})/r^4$$

and

$$C_\phi = \cos \phi$$

$$S_\phi = \sin \phi$$

the computation of $\dot{e}_t$ in two dimensions is obtained by straightforward differentiation:

$$\dot{e}_{tx} = (\dot{x}\, C_\phi - q\, \dot{y}\, S_\phi)/r + (x\, C_\phi - q\, y\, S_\phi)\, Y$$

$$\dot{e}_{ty} = (\dot{y}\, C_\phi + q\, \dot{x}\, S_\phi)/r + (y\, C_\phi + q\, x\, S_\phi)\, Y$$

$$\dot{e}_{tz} = 0 \text{ (not computed).}$$

THANGD EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
P(3)	UX		Primer vector, Λ.
R(3)	UX		Spacecraft position vector, R, in AU.
PD(3)	UX		Primer derivative, $\dot{\Lambda}$.
RD(3)	UX		Position derivative, $\dot{R}$, in AU/tau.
RM	UX		Position magnitude, $r = \lvert R \rvert$, in AU.
ETH(3)	SU	REAL8	Thrust unit vector, e_t.
ETHD(3)	SA	REAL8	Thrust unit vector derivative, $\dot{e}_t$, in tau^{-1}.
PLUS	U	LOGIC4	Logical indicator for determining appropriate region (of two possible solutions) in two dimensional simulations.
COPHI	U	REAL8	Cosine of fixed thrust angle, $C_\phi = \cos \phi$.
SIPHI	U	REAL8	Sine of fixed thrust angle, $S_\phi = \sin \phi$.
TUDFLG	U	LOGIC4	Logical indicator for two dimensional trajectory simulation.

CHART TITLE - SUBROUTINE THANGD(R,RD,P,PD,RM)

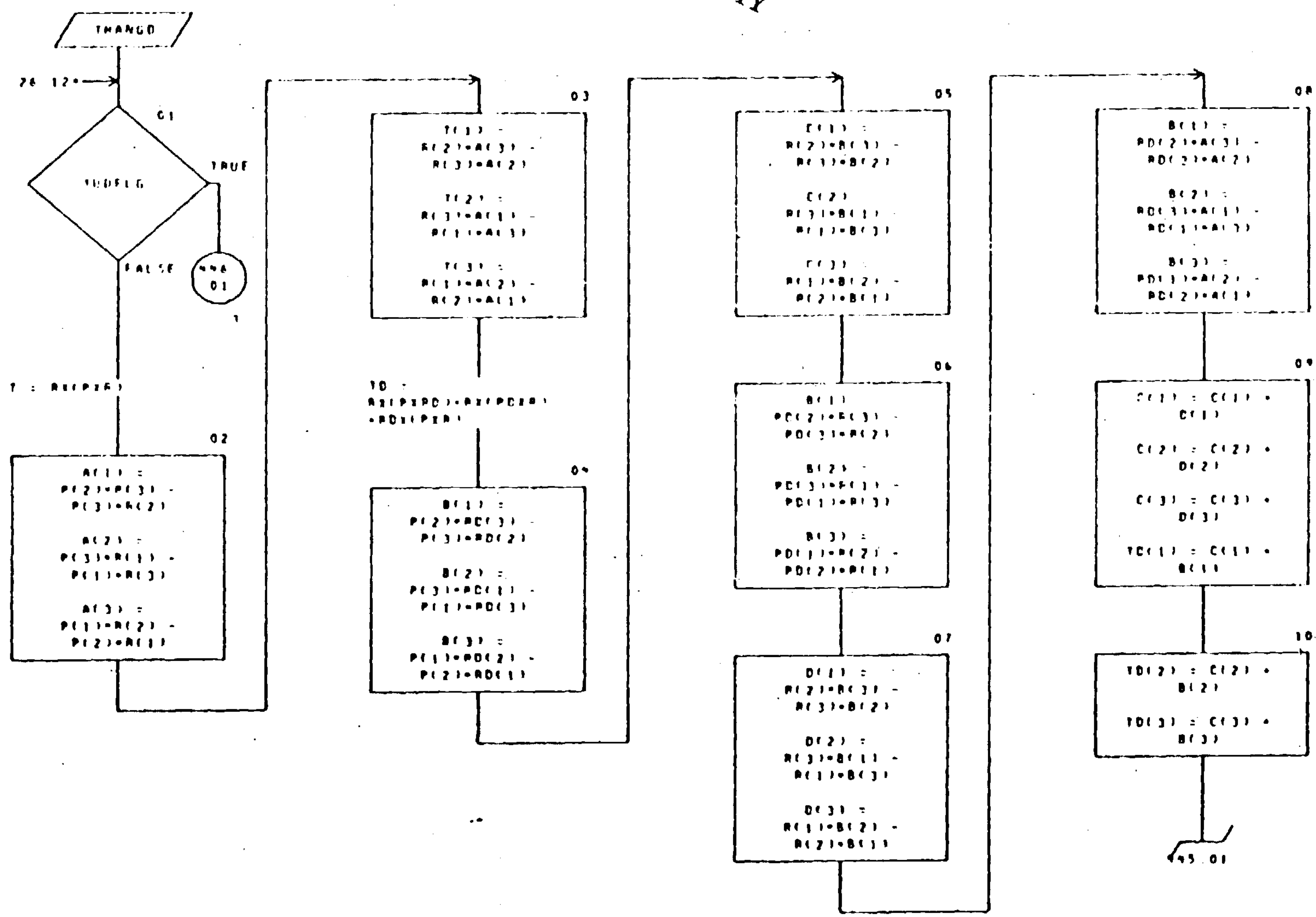
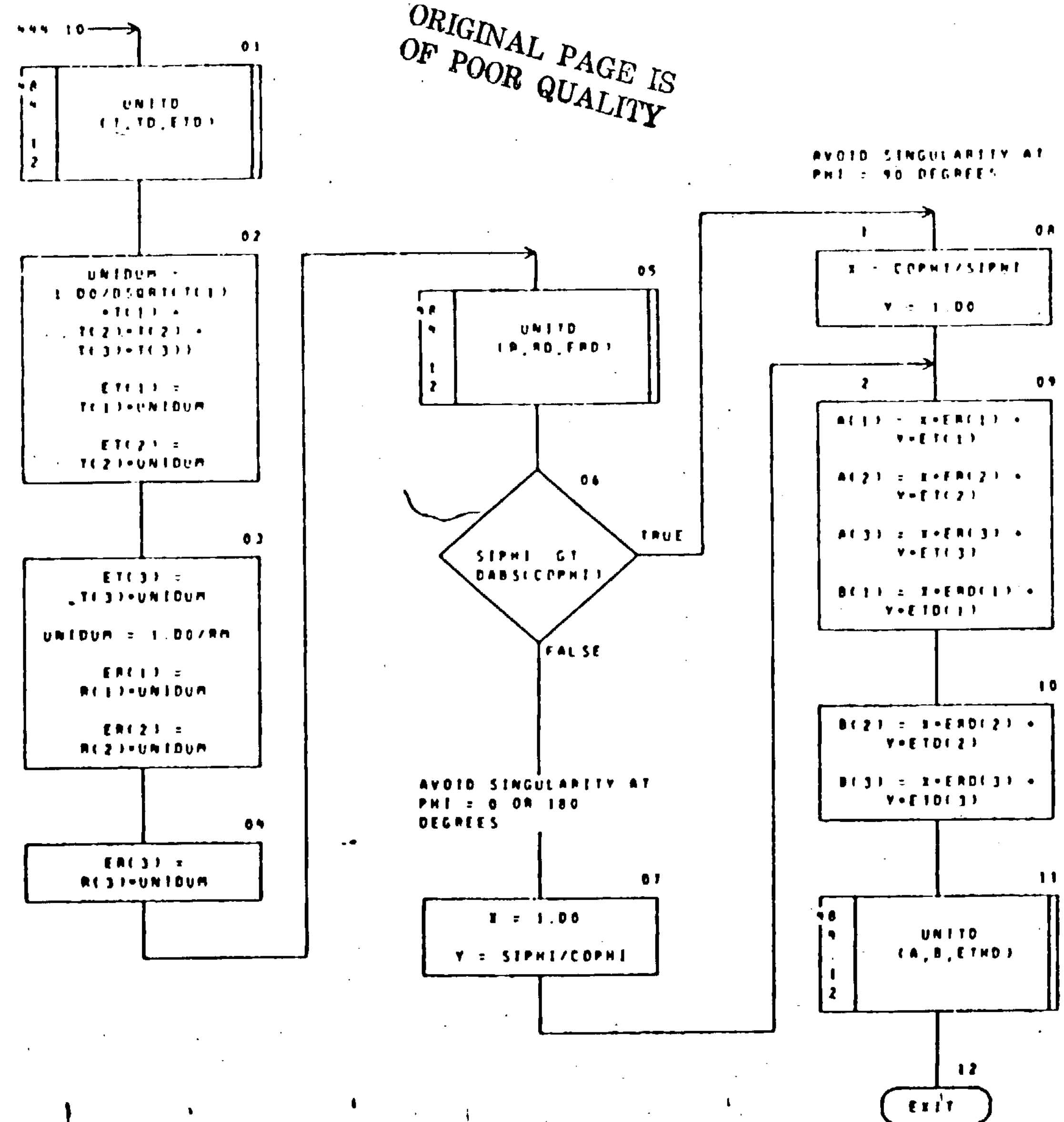


CHART TITLE - SUBROUTINE THANGD(R,RD,P,PD,RM)

ORIGINAL PAGE IS OF POOR QUALITY



THANGD-7

CHART TITLE - SUBROUTINE THANGD(R,RD,P,PD,RM)

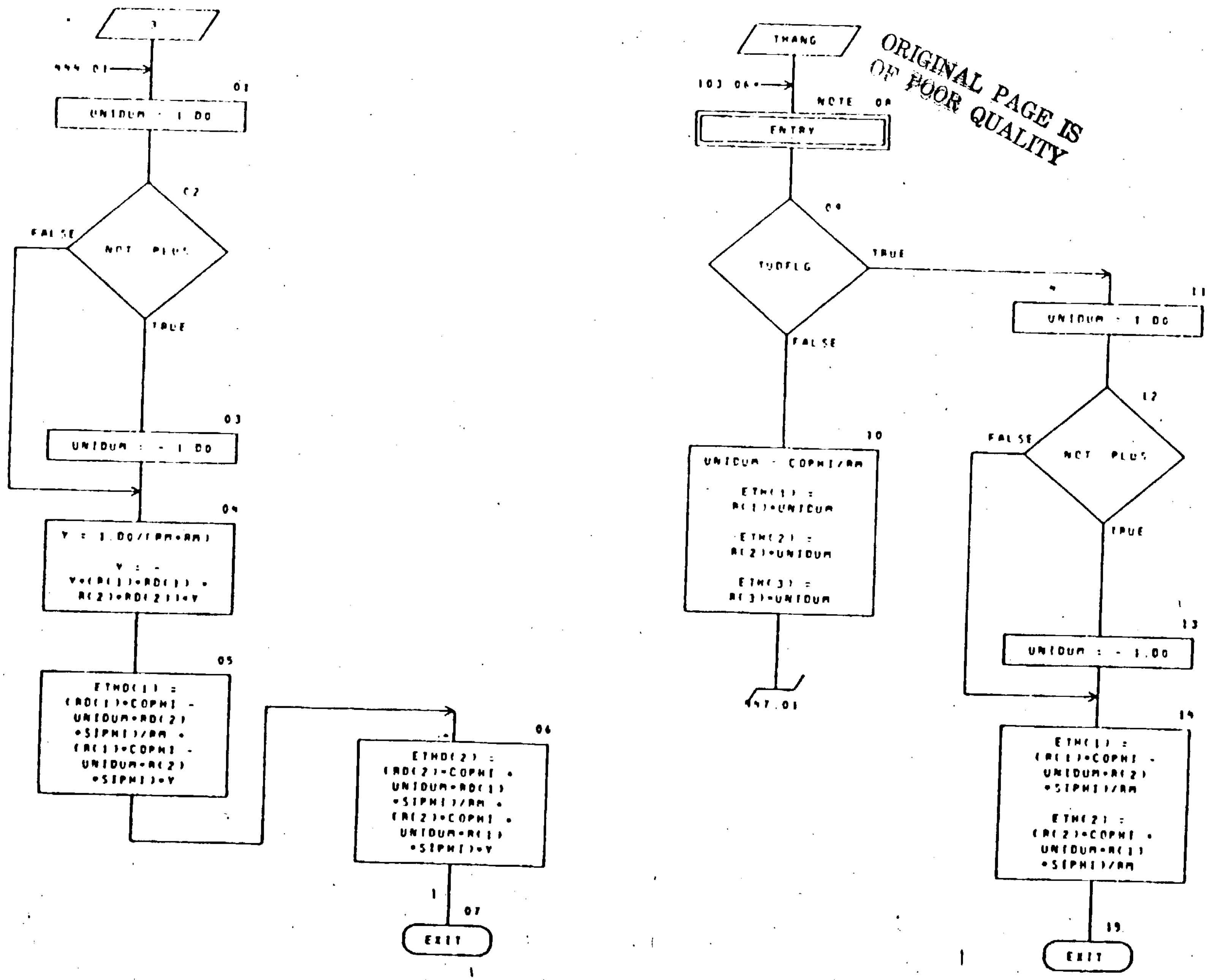
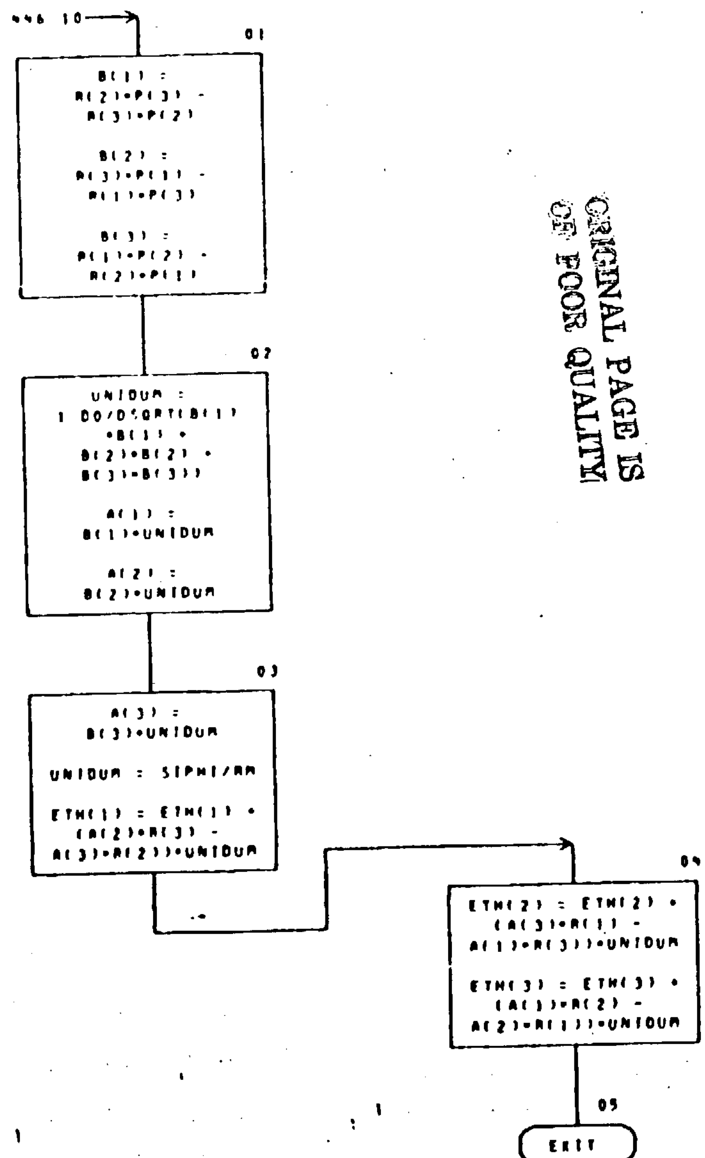


CHART TITLE - SUBROUTINE THANGD(R,RD,P,PD,RM)



THANGD-9

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL * 8 (A-H,O-Z)
LOGICAL TUDFLG,PLUS
DIMENSION R(3),RD(3),P(3),PD(3),A(3),B(3),C(3),D(3),T(3),TD(3)
.,E(3),ETD(3),EP(3),EPD(3)
COMMON /REALA/ RO1(825),ETH(3),ETHD(3),RO2(10),SIPHI,COPHI,
RO3(1157)
COMMON /LOGICA/ LO1(8),TUDFLG,LO2(30),PLUS,LO3(460)
```

<u>Name:</u> TIKTOK

<u>Calling Argument:</u> LOGIC

<u>Referenced Sub-programs:</u> FINISH, REMTIM

<u>Referenced Commons:</u> INTGR4, ITER2, LOGIC4, REAL8

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> INPUT, MAIN, MINMX3

<u>Discussion:</u> TIKTOK monitors the CPU and I/O machine time remaining regarding the current computer run, by making use of the GSFC/IBM subroutine REMTIM, and terminates the run after calling subroutine FINISH when a machine time-out is impending.

<u>Messages and Printouts:</u> When a machine time-out is impending for a given computer run, the following is output on unit 6:

MAX __type__ TIME

__n_1__ TRAJECTORIES WITHOUT PARTIAL DERIVATIVES AND __n_2__ TRAJECTORIES WITH PARTIAL DERIVATIVES REQUIRED FOR THIS CASE.

followed by a detailed printout of the "last" (most recent) trajectory which the iterator was using when the time-out condition occurred. In the above, "type" is either CPU or I/O, depending on which type of IBM machine time is about to expire, and n_1 and n_2 are trajectory-counters output from the MINMX3 iterator. The detailed printout of the last trajectory is followed by

END OF RUN

ELAPSED CPU TIME = __m_1__ MINUTES
ELAPSED I/O TIME = __m_2__ MINUTES

where m_1 and m_2 are the elapsed CPU and I/O times, respectively, computed internally by the program.

Concurrently, the following is output on unit 12:

$\underline{\quad n_1 \quad}$ TRAJECTORIES WITHOUT AND $\underline{\quad n_2 \quad}$ WITH PARTIALS

where n_1 and n_2 are defined above, and

MAX $\underline{\quad \text{type} \quad}$ TIME $\underline{\ m_1 \text{ or } m_2\ }$ MINUTES

where "type", m_1, and m_2 are defined above.

The subroutine also prints out on unit 6, at the beginning of each case, the CPU and I/O times remaining for the given computer run:

CASE $\underline{\quad n \quad}$, TIME TO GO CPU $\underline{\quad j_1 \quad}$, I/O $\underline{\quad j_2 \quad}$ SEC

where n is the case number, and j_1 and j_2 are the remaining CPU and I/O times in seconds.

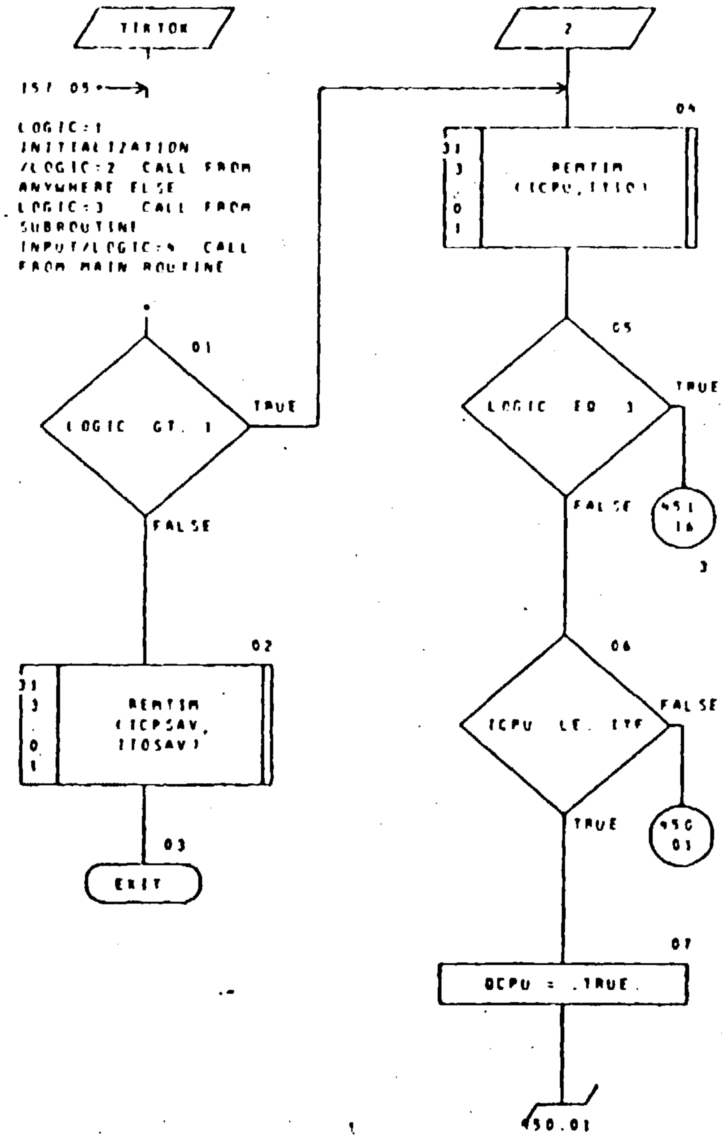
TIKTOK EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(35)	S	ITER2	Iterator independent variable array.
GAP	S	REAL8	Propulsion-corner proximity tolerance-interval, $\Delta\sigma$.
ITF	U	INTGR4	Estimated time remaining to halt computer run with full printout, in case of proximity to maximum machine time, in seconds.
LXX	U	INTGR4	Number of iterator independent variables.

TIKTOK EXTERNAL VARIABLES TABLE (cont)

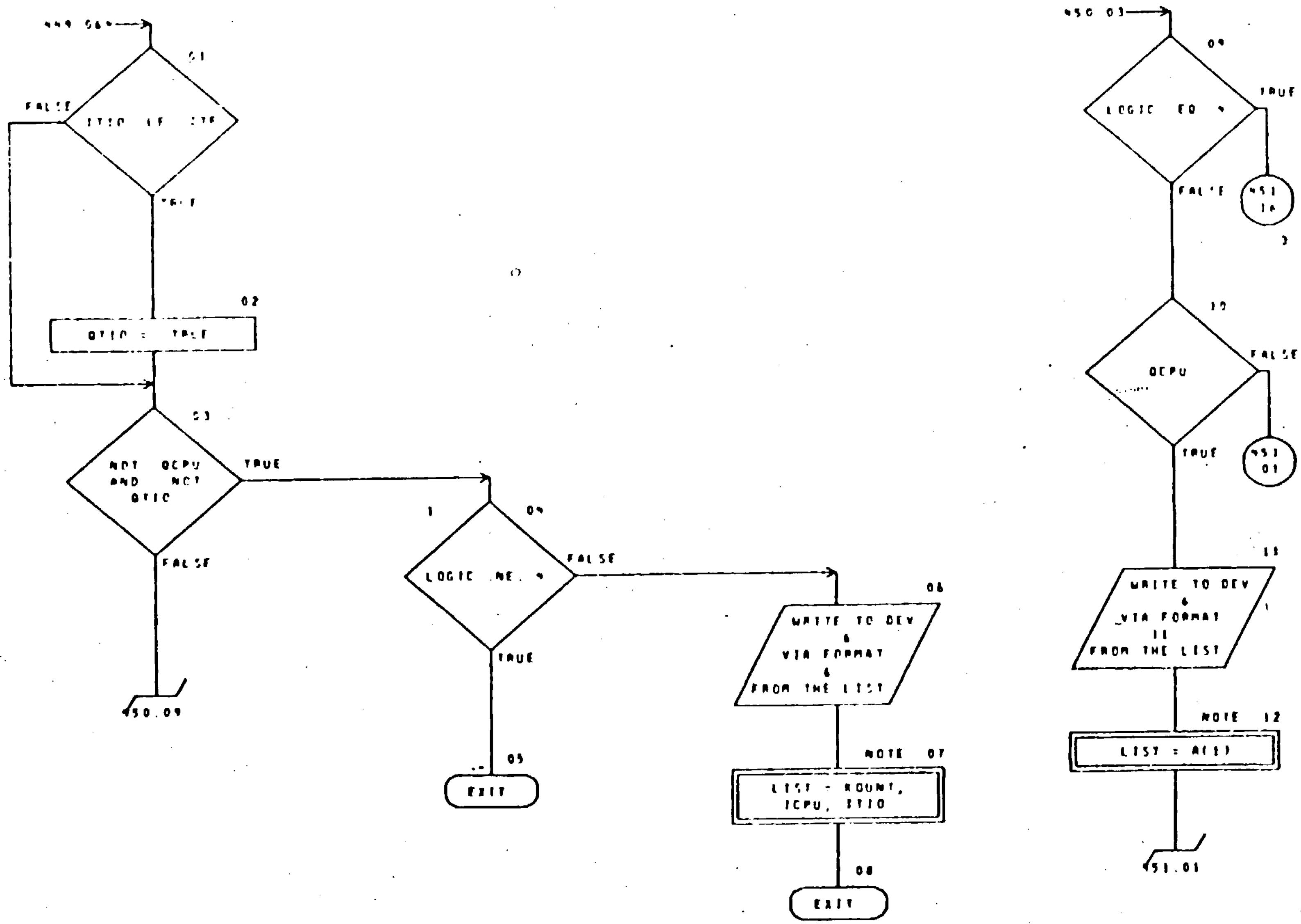
Variable	Use	Common	Description
QUIT	U	LOGIC4	Logical variable, set in subroutine SWING, which causes bypass of last trajectory printout if time-out occurs during swingby continuation analysis.
BNOMX (35)	U	ITER2	Iterator independent variable array containing saved values corresponding to the most recently executed nominal trajectory.
ERROR	S	LOGIC4	Program master error indicator.
JHUNG	S	INTGR4	Indicator for controlling the two additional cases which attempt to avoid the propulsion-time corner(s).
KOUNT	SU	INTGR4	Case counter.
LOGIC	UX		Master control indicator for this subroutine. See comments displayed at top of source-listing.
MAJOR	U	INTGR4	Total number of nominal trajectories generated during the current case.
CONVRG	S	LOGIC4	Indicator for trajectory convergence (all two-point boundary value problem conditions satisfied).
MAJORS	U	INTGR4	Total number of nominal plus search trajectories generated during the current case.

CHART TITLE - SUBROUTINE TIKTOK(LOGIC)



ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - SUBROUTINE TIKTOK(LOGIC)



TIKTOK-5

CHART TITLE - SUBROUTINE TIKTOK(LOGIC)

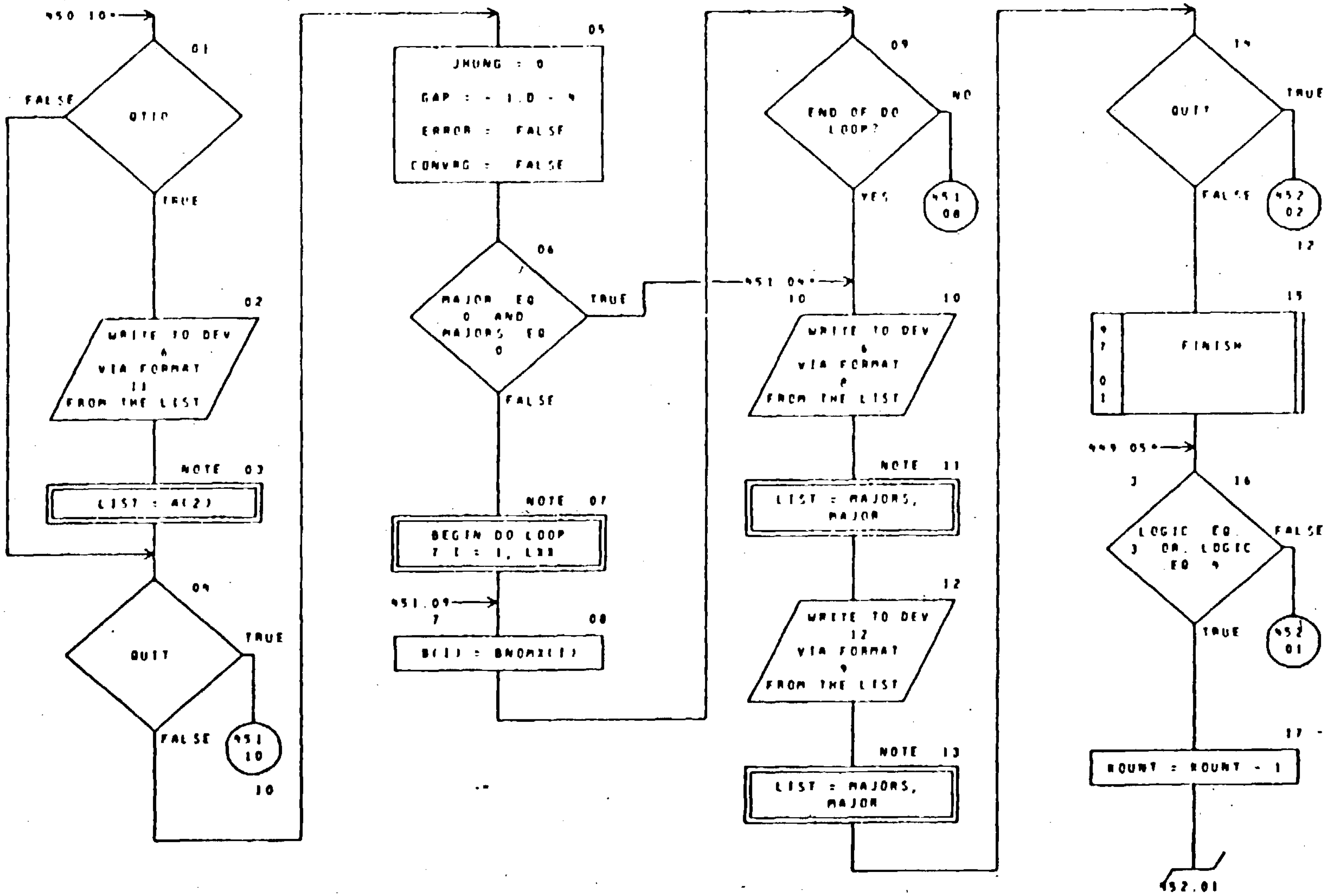
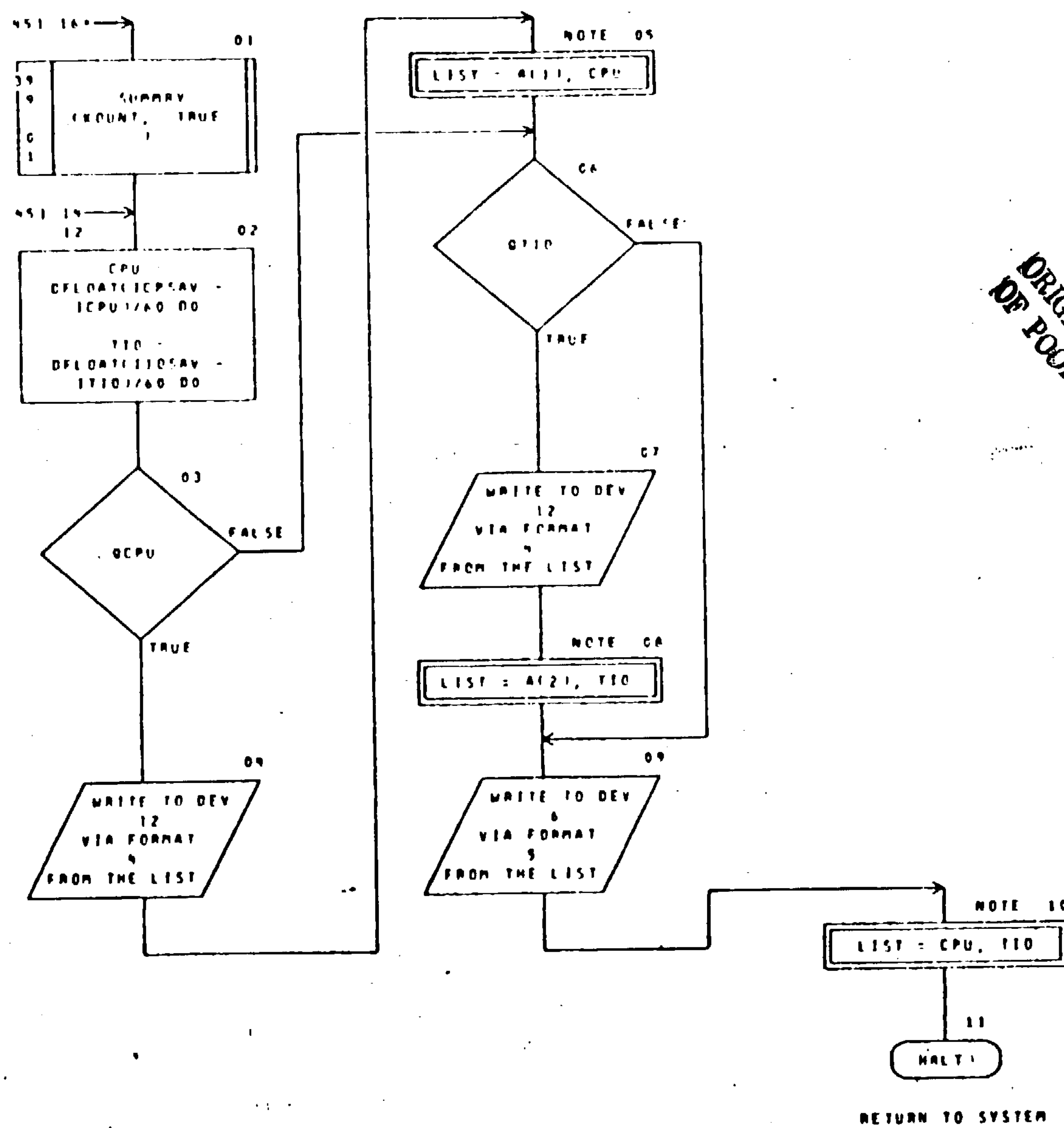


CHART TITLE - SUBROUTINE TIKTOK(LOGIC)



TIKTOK-7

CHART TITLE - NON-PROCEDURAL STATEMENTS


```
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL*4 A
      LOGICAL QCPU,QTIO,ERROR,CONVRG,QUIT
      DIMENSION A(2)
      COMMON /REAL8/ RG1(102),GAP,RO2(1897)
      COMMON /INTGR4/ IO1(3),ITF,IO2(49),ICOUNT,IO3(7),LB3,IO4(3)
     ,JHUNG,MAJOR,MAJORS,IO5(932)
      COMMON /LOGIC4/ ERROR,CONVRG,LO1(42),QUIT,LO2(455)
      COMMON /ITER2/ B(35),PO1(245),BNORM(35),PO2(1225)
      DATA QCPU,QTIO,A /2*.FALSE.,3HCPU,3HI/O/
11    FORMAT(1H0,4HMAX A4,4HTIME)
2     FORMAT(1H0,I6,45H TRAJECTORIES WITHOUT PARTIAL DERIVATIVES ANDI4,
     62H TRAJECTORIES WITH PARTIAL DERIVATIVES REQUIRED FOR THIS CASE.)
3     FORMAT(1H ,I4,25H TRAJECTORIES WITHOUT ANDI3,14H WITH PARTIALS)
4     FORMAT(1H ,4HMAX A4,4HTIMEF6.2,8H MINUTES)
5     FORMAT(1H0///1H ,10HEND OF RUN//1H ,
     18HELAPSED CPU TIME =F6 2,8H MINUTES/1H ,
     18HELAPSED I/O TIME =F6 2,8H MINUTES////1H )
6     FORMAT(1H*,4HCASEI3,8X16HTIME TO GO   CPUI4,5H, I/OI4,4H SEC/1H )
```

Name: TRAJ

Calling Argument: ERROR for TRAJ;
None for TRJINT

Referenced Sub-programs: EFM, ETA, GETQ, GETRV, OMASS, PRINT, PRINTR, RETINJ, RIDGE, TAP, TRAJI, VPRINT for TRAJ;
EFM, ETA, OMASS, RETINJ for TRJINT

Referenced Commons: INTGR4, ITERAT, LOGIC4, REAL8, SOLSYS

Entry Points: TRJINT

Referencing Sub-programs: FINISH, MINMX3 for TRAJ;
QSTART for TRJINT

Discussion: TRAJ is the software package called primarily by the MINMX3 iterator (in which it is named PD5) which produces the dependent-variable values of the electric propulsion two-point boundary value problem given the independent-variable values.

Subroutine TRAJ employs subroutine TAP to generate the required trajectory segments between celestial bodies; on each trajectory segment, the time X(17), which is expressed in tau, advances from its current value (at the start of the segment) by an amount TMAX = TCHECK (JCMAX). When imposed coast phases are invoked, appropriate values for the coast-phase start and end times are loaded into the lower elements of the TCHECK array, TCHECK(JC) for JC less than JCMAX.

When intermediate targets are present, their positions and velocities, P_i and $\dot{P}_i$, at the respective times of intercept t_i are computed by subroutine EFM. When program input quantity LOADX is invoked, the values of the primer vector and its time derivative at the endpoint of the current trajectory segment are loaded into the iterator independent-variable array for starting the next trajectory segment.

After all trajectory segments have been computed, the time X(17), the mass ratio X(7), and all other trajectory dependent-variable values X(i) correspond to the trajectory endpoint at primary-target arrival time t_n. The spacecraft mass computations and computations of trajectory-related parameters are then executed, as discussed below.

The primer magnitude is computed,

$$\lambda_n = \sqrt{\Lambda_n \cdot \Lambda_n} .$$

When the simulation is two-dimensional, and the final solar distance is specified and/or the travel angle is optimized, the position and velocity of an imaginary body moving in a circular orbit at the spacecraft's final position are computed,

$$P_n = R_n ,$$

$$\dot{P}_n = \sqrt{(1 + \mu_t/\mu_{sun})/r_n} \; \frac{\hat{k} \times R_n}{r_n} ,$$

where μ_t is the imaginary body's (primary target's) gravitational constant, μ_{sun} is the sun's gravitational constant, $r_n = |R_n|$, and $\hat{k}$ is a unit vector directed toward the North Ecliptic Pole (z axis). The square-root term in the above expression is the circular speed at solar distance r_n, and the cross-product term is the unit tangential vector.

The efficiency $\eta(c)$ and $d\eta/dc$ are computed via subroutine ETA.

The initial spacecraft mass m_o is computed via subroutine OMASS when operating in the launch-vehicle-dependent mode, and by the formula

$$m_o = \frac{2\eta p_{ref}}{g c} ,$$

when operating in the launch-vehicle-independent mode, in which case the reference power is input to the program. When OMASS is used, the reference power is computed by inverting the above formula,

$$p_{ref} = \frac{g c m_o}{2\eta} .$$

The spacecraft mass components are computed as follows:

The electric propulsion engine mass is given by,

$$m_{ps} = p_{ref} [\alpha_t + (1 + \Delta p) \alpha_a],$$

where α_t is the specific mass of the thruster and power conditioning subsystem, α_a is the specific mass of the arrays, and Δp is the ratio of housekeeping to reference power, a program input constant. The electric propulsion propellant mass m_p is obtained by integrating the derivative of mass ratio over all thrusting arcs and employing the equation,

$$m_{pn} = m_o (1 - \nu_n) + \sum_{i=1}^{n-1} (m_{samp\,i} - m_{drop\,i}),$$

where ν_n is the mass ratio at the primary target (i.e., prior to the optional retro maneuver) and $m_{samp\,i}$ is the sample mass picked up at the i^{th} target.

The sample masses and drop masses are specified as linear functions of the initial mass, as follows:

$$m_{samp\,i} = m_o k_{samp\,i}\,; \quad m_{drop\,i} = m_o k_{drop\,i}\,,$$

where $k_{samp\,i}$ and $k_{drop\,i}$ are program inputs and are available as independent parameters of the boundary value problem. Both $m_{samp\,i}$ and $m_{drop\,i}$ are available as dependent parameters.

The electric propulsion propellant tankage mass and structure mass are computed, respectively,

$$m_t = k_t m_p,$$

$$m_s = k_s m_o,$$

where k_t and k_s are program input constants.

TRAJ-3

The net spacecraft mass, which is the main performance index of the program's electric propulsion optimization problem, is given by

$$m_{net} = m_o - m_{ps} - m_p - m_t - m_s - m_r - \sum_{i=1}^{n-1} m_{drop\,i} .$$

Auxiliary quantities associated with the optional retro maneuver at the primary target are computed (the QTMASS array):

$$q_1 = e^{-\Delta v'/c_r} ,$$

$$e_x = 1 - q_1 ,$$

$$q_2 = m_o \nu_n - j_{ps} m_{ps} - j_t m_t ,$$

$$q_3 = q_2 e_x ,$$

q_4 not used,

$$q_5 = m_{rs} + q_3 c_r ,$$

and these quantities go into the computation of the retro-mass (in the expression for m_{net} above):

$$m_r = q_3 + q_5 + m_o \sum_{i=1}^{n-1} k_{drop\,i} .$$

The initial primer magnitude is computed,

$$\lambda_o = \sqrt{\Lambda_o \cdot \Lambda_o} .$$

The quantity g_x, which is used in conjunction with the transversality computations, is computed,

$$g_x = j_r e_x (1+k_{rt})\left[1 + \frac{4c_1 f_x (1-e_x)}{2c_r - c_1 (f_x - e_x)^2}\right],$$

where f_x is given by

$$f_x = 2-(1 + \frac{2c_r}{\Delta v'}) e_x .$$

The partial derivative of the performance index with respect to the initial spacecraft mass is computed,

$$\pi_{m_o} = \frac{\partial \pi}{\partial m_o} = k_s + k_t + j_p \frac{\alpha g c}{2\eta} - (1+k_t)\nu_n + (1+k_t)\sum_{i=1}^{n-1} k_{samp\,i}$$

$$-k_t \sum_{i=1}^{n-1} k_{drop\,i} + g_x\left[(1+j_t k_t)\nu_n - j_t k_t (1 + \sum_{i=1}^{n-1} (k_{samp\,i} - k_{drop\,i}))\right.$$

$$\left. - j_p j_{ps} \frac{\alpha g c}{2\eta}\right].$$

The arbitrary positive performance index constant k is assigned a value which causes the transversality condition associated with the final mass ratio to be satisfied;

$$k = \lambda_{\nu_n} / \pi_{\nu_n} ,$$

where

$$\pi_{\nu_n} = m_o [g_x (1+j_t k_t) - (1+k_t)] .$$

When the primary target corresponds to an out-of-the-ecliptic mission, the position and velocity of the imaginary target are obtained from subroutine GET RV.

The initial primer vector used in the transversality computations in subroutine GETQ is given by

$$\Lambda_{otr} = a_k \Lambda_o + (1 - a_k) \dot{\Lambda}_o ,$$

where the weighting function a_k is given by

$$a_k = \left[1 - e^{-10\,|\Lambda_o| / |\dot{\Lambda}_o|}\right]^{\frac{1}{1000}} .$$

The parameter $a_k \cong 1$ unless $|\Lambda_o|$ becomes very small with respect to $|\dot{\Lambda}_o|$. After being used in the above formula for Λ_{otr}, the parameter a_k is recomputed as a normalizing factor to be used in subroutine GETQ:

$$a_k = \pm \frac{v_{\infty o}}{|\Lambda_{otr}|} .$$

Then subroutine GETQ is invoked, which computes the transversality conditions and targeting conditions (iterator dependent-variables). The iterator independent and dependent-variable values are optionally printed, and the propulsion-time corner proximity counter is incremented via subroutine RIDGE if the trajectory is in the neighborhood of a propulsion-time corner.

Entry point TRJINT initializes logical indicators and computes parameters whose values remain constant on subsequent passes through TRAJ (during an iteration sequence).

Messages and printouts: Should the time of intercept of any intermediate target become greater than the time of intercept of the primary target, which may occur either through incorrect program input or by action of the iterator, the message is printed on units 6 and 12,

INTERMEDIATE TIME GREATER THAN FINAL TIME.
CASE TERMINATED.

and the program master error indicator is set, and the routine is exited. In like manner, should the intercept times of two intermediate targets become anti-chronological, the same procedure occurs after the message is printed on units 6 and 12,

TARGET PASSAGE TIMES NOT ASCENDING.
CASE TERMINATED.

If the iteration to obtain the desired final time value of a given trajectory segment is not satisfied to within approximately .0001%, the master error indicator is set and the routine is exited after the message is printed on units 6 and 12,

FINAL TIME VIOLATED $\underline{\quad t_i \quad}$ $\underline{\quad t_{max} \quad}$

where t_i is the actual time obtained and t_{max} is the desired time, both in units of tau.

TRAJ EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
O(70)	SUA	ITERAT	Array of iterator independent-variables, in program internal units.
R(2)	U	REAL8	Spacecraft solar distance (in this subroutine, applies to time at primary target intercept), in AU.
X(50)	SU	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
AK	SU	REAL8	Weighting factor between initial primer vector and its derivative; primer vector normalizing factor output to subroutine GETQ, a_k.
BX(5, 70)	SU	ITERAT	Iterator independent variable array input to the program in the form Xi.
EX	SU	REAL8	$e_x = 1 - e^{-\Delta v'/c_r}$.
FP	U	REAL8	f', the derivative of $f = \Delta v' - \Delta v - c_1 e_x f_x$ with respect to $\Delta v'$, as described in subroutine RETINJ.
GM(70)	U	SOLSYS	Array of planetary gravitational constants, in m^3/sec^2.
JC	S	INTGR4	Index for time-function array (see subroutine CHECK).

TRAJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
JT	U	INTGR4	Jettison indicator j_t for electric propulsion tankage prior to primary-target retro-maneuver.
OO(70)	S	ITERAT	Array of iterator independent-variables, in program external units.
P0(7)	U	REAL8	Array of initial adjoint variables Λ_o, $\dot{\Lambda}_o$, $\lambda_{\nu o}$.
VH	U	REAL8	Speed at periapse of approach hyperbolic trajectory at the primary target, in meters/second.
VJ	A	REAL8	Jet exhaust speed of electric propulsion system, c, in EMOS.
VS	U	REAL8	v_c^2, as described in subroutine RETINJ, in meters^2/second^2.
XD(50)	U	REAL8	Array of trajectory dependent-variable derivatives, allocated the same as X(i), as described in subroutine RKSTEP.
X0(7)	S	REAL8	Initial launch planet state P_o, $\dot{P}_o$, and initial mass ratio ν_o.
ABX(70)	U	LOGIC4	Master array of iterator independent-variable indicators.
AJT	SU	REAL8	Floating-point equivalent of JT.
AVJ	U	REAL8	Inverse of electric propulsion system jet exhaust speed, 1/c, in EMOS^{-1}.
A1B	U	LOGIC4	Logical indicator defined in subroutine SETUP.
A2B	U	LOGIC4	Logical indicator defined in subroutine SETUP.

TRAJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
FMS	S	REAL8	$\partial m_o / \partial v_{\infty o}$, in kg/EMOS.
HAM	U	REAL8	Variational hamiltonian h_v at the beginning of the current trajectory-segment.
JPP	U	INTGR4	Jettison indicator j_{ps} for electric propulsion system prior to primary-target retro-maneuver.
LEG	SUA	INTGR4	Counter indicating the current trajectory-segment.
PMN	S	REAL8	Magnitude of primer vector, λ.
PMS	SU	REAL8	Square of magnitude of primer vector, λ^2.
PM0	SU	REAL8	Magnitude of initial primer vector, λ_o.
VHS	U	REAL8	Square of speed at periapse of approach hyperbolic trajectory, pertaining to retro-maneuver at the primary target, in meters/second.
AJPP	SU	REAL8	Floating-point equivalent of JPP.
A11C	U	LOGIC4	Logical indicator defined in subroutine SETUP.
A13A	U	LOGIC4	Logical indicator defined in subroutine SETUP.
A18A	U	LOGIC4	Logical indicator defined in subroutine SETUP.
A19A	U	LOGIC4	Logical indicator defined in subroutine SETUP.
A20A	U	LOGIC4	Logical indicator defined in subroutine SETUP.

TRAJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
CONX(70)	U	ITERAT	Array of print conversion factors for iterator independent variables.
CSTR	U	REAL8	Electric propulsion system structural factor, k_s.
DPOW	U	REAL8	Ratio of housekeeping power to reference power, p_h/p_{ref}, a constant, input to the program.
FETA	S	REAL8	$1/c - \eta'/\eta$, in $EMOS^{-1}$.
FMSI	S	REAL8	$\partial v_c/\partial i$, in EMOS/radian (see subroutine OMASS).
FTVJ	U	REAL8	Auxiliary parameter gc, in AU^2/tau^3.
HAMX(4)	S	REAL8	Variational hamiltonian values at the beginning of trajectory-segments departing intermediate targets.
PIMO	SU	REAL8	π_{m_o}.
PM0D	SU	REAL8	Magnitude of initial primer derivative, $\lvert\dot{\Lambda}_o\rvert$.
QCST	U	REAL8	Retro stage mass, m_{rs}, in kilograms.
QJEX	U	LOGIC4	Detailed printout indicator.
TEST	S	REAL8	Tolerance value in test for zero initial primer vector, used in subroutine TRAJI.
TMAX	SU	REAL8	Elapsed time at end of current trajectory-segment (integration stopping condition), in tau.
TOFF(20)	U	REAL8	Array of times, from the start of the trajectory, at which imposed coast phases are to begin, in days.

TRAJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
VIMP	A	REAL8	Launch hyperbolic excess speed, v_{∞_0}, in EMOS.
VINF	S	REAL8	Hyperbolic excess speed at the primary target, v_{∞_n}, in meters/second.
XINT(50, 5)	S	REAL8	Saved trajectory dependent-variable values at arrival at the intermediate targets.
CONPW	U	REAL8	Power conversion factor, in m^2/sec^3.
CONTM	U	REAL8	Time conversion factor, tau to days.
CTANK	U	REAL8	Electric propulsion system propellant tankage factor, k_t.
CTRET	U	REAL8	Retro tankage factor, k_{rt}, for retro maneuver at the primary target.
DROPS	SU	REAL8	Sum of drop-mass factors, $\sum_{i=1}^{n-1} k_{drop\ i}$.
ERROR	SAX		Error-condition indicator.
FLYBY	U	LOGIC4	Indicator that maneuver at primary target is flyby.
GPLAN	SU	REAL8	Gravitational constant of primary target, μ_t, in m^3/sec^2.
GSUBX	SU	REAL8	Auxiliary parameter g_x.
IHUNG	U	INTGR4	Counter of the number of propulsion-corner-proximity occurrences along the current trajectory.
INTER(5)	A	INTGR4	Array of indices which select the correct orbital elements for the intermediate targets.
JCMAX	S	INTGR4	Maximum value which JC may attain, corresponding to the end of the current trajectory segment.

TRAJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
LOOSE	SU	LOGIC4	Indicator that the initial heliocentric spacecraft velocity is included in the (active) iterator independent variables.
MAJOR	U	INTGR4	Counter of the number of nominal ("major") trajectories, which have an associated set of neighboring trajectories, generated in the current iteration sequence.
MOPTX(5)	A	INTGR4	The target-numbers of the successive intermediate targets.
MOPT2	A	INTGR4	Launch planet number.
MOPT3	UA	INTGR4	Planet-number of primary target.
PSIGN	U	REAL8	Coefficient defining the sense of the launch hyperbolic excess velocity relative to the initial primer vector.
RPLAN	S	REAL8	Radius of primary target, in meters.
SAMPS	SU	REAL8	Sum of sample-mass factors, $\sum_{i=1}^{n-1} k_{samp\,i}$.
SCALE	S	REAL8	Arbitrary positive performance index constant, k.
SEFMA(7)	UA	REAL8	Array containing position and velocity of launch planet at launch time, P_o and $\dot{P}_o$, in AU and EMOS, respectively.
SEFMB(7)	SA	REAL8	Array containing position and velocity of primary target at time of target intercept, P_n and $\dot{P}_n$, in AU and EMOS, respectively.
SEFMC(7)	A	REAL8	Time derivative of SEFMA array (w.r.t. tau^{-1}).
SEFMD(7)	A	REAL8	Time derivative of SEFMB array (w.r.t. tau^{-1}).

C-9

TRAJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
SUNMU	U	REAL8	Gravitational constant of the sun, in m^3/sec^2.
TEMP2	S	REAL8	Auxiliary quantity, used in subroutine GETQ.
TEMP4	SU	REAL8	Auxiliary quantity, used in subroutine GETQ.
VJRET	U	REAL 8	Retro-stage jet exhaust speed, c_r, pertaining to retro maneuver at the primary target, in meters/second.
VLOSS	U	REAL 8	Velocity loss associated with retro maneuver at the primary target, in meters/second.
WPRIM(3)	SU	REAL8	Initial primer vector, designed to reflect the correct direction of $v_{\infty o}$ when $\Lambda_o \rightarrow 0$.
XDINT (50, 5)	S	REAL8	Saved trajectory dependent-variable derivative values at arrival at the intermediate targets.
XLOAD	U	LOGIC4	Indicator for invoking the intermediate-target initial-guess feature.
XMASS(7)	SUA	REAL8	Array of masses (in kilograms) and mass-related parameters: (1) Initial mass, m_o. (2) Propulsion system mass, m_{ps}. (3) Propellant mass, m_p. (4) Tankage mass, m_t. (5) Structure mass, m_s. (6) Reference power, p_{ref}, in watts. (7) Efficiency, η.
XTINT (6, 5)	A	REAL8	Positions and velocities of intermediate targets at times of intercept, P_i and $\dot{P}_i$, in AU and EMOS, respectively.

TRAJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
ALPHAA	U	REAL8	Specific mass of solar arrays, α_a, in kg/kw.
ALPHAT	U	REAL8	Specific mass of thruster subsystem, α_t, in kg/kw.
ALWAYS	S	LOGIC4	Indicator for fixed (specified) non-zero propulsion-time adjoint variable.
EMUODD	U	REAL8	Gravitational constant of the primary target, input to the program, in m^3/sec^2.
ERRORX	U	LOGIC4	Program master error indicator.
FIXPOW	U	LOGIC4	Launch-vehicle-independent trajectory option indicator.
JPRINT	U	INTGR4	Unit 11 printout-length indicator.
LEGMAX	U	INTGR4	Total (maximum) number of trajectory-segments comprising the trajectory.
NPRINT	U	INTGR4	Printout amount selection indicator.
OUTECL	U	LOGIC4	Extra-ecliptic mission indicator.
PAYLOD	SU	REAL8	Net spacecraft mass, m_{net}, in kg.
PLANET	U	LOGIC4	Ephemeris-option indicator.
POWFIX	U	REAL8	Spacecraft reference power, p_{ref}, in kw, applicable only in the launch-vehicle-independent mode of operation.
PRZERO	SU	LOGIC4	Indicator that zero initial primer vector is the desired condition.
QTMASS (5)	SU	REAL8	Array of mass-related parameters, as described in subroutine RETINJ.
QVLOSS	U	LOGIC4	Indicator for program option invoked via program input quantity ALTITU.

TRAJ EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
RADIUS(70)	U	SOLSYS	Array of planetary-body radii, in meters.
RADODD	U	REAL8	Radius of primary target, input to the program, in meters.
SPIRAL	U	LOGIC4	Indicator for electric propulsion spiral capture maneuver at the primary target.
TCHECK (41)	SU	REAL8	Array of time values, each of which is isolated by subroutine CHECK, in tau; intermediate values correspond to engine switch times associated with imposed coast phases.
TCOAST (20)	U	REAL8	Array of times representing the duration of the coast-phase start times in TOFF, in days.
TDATEX	U	REAL8	Reference Julian date, less 2400000, defined by program input quantity MYEAR, etc.
TDATE1	SA	REAL8	Launch Julian date, less 2400000.
TDATE2	SA	REAL8	Julian date at time of primary-target intercept, less 2400000.
VELOSS	U	LOGIC4	Indicator for performing velocity loss or velocity penalty computations in subroutine RETINJ.
WONDER	U	LOGIC4	Indicator for bypassing all tests associated with the condition in which the iteration sequence is stymied because the trajectory is in high proximity to a corner in the propulsion-time function.
XTDINT (6,5)	A	REAL8	Time derivative of XTINT array (w.r.t. tau^{-1}).

CHART TITLE - SUBROUTINE TRAJ(ERROR)

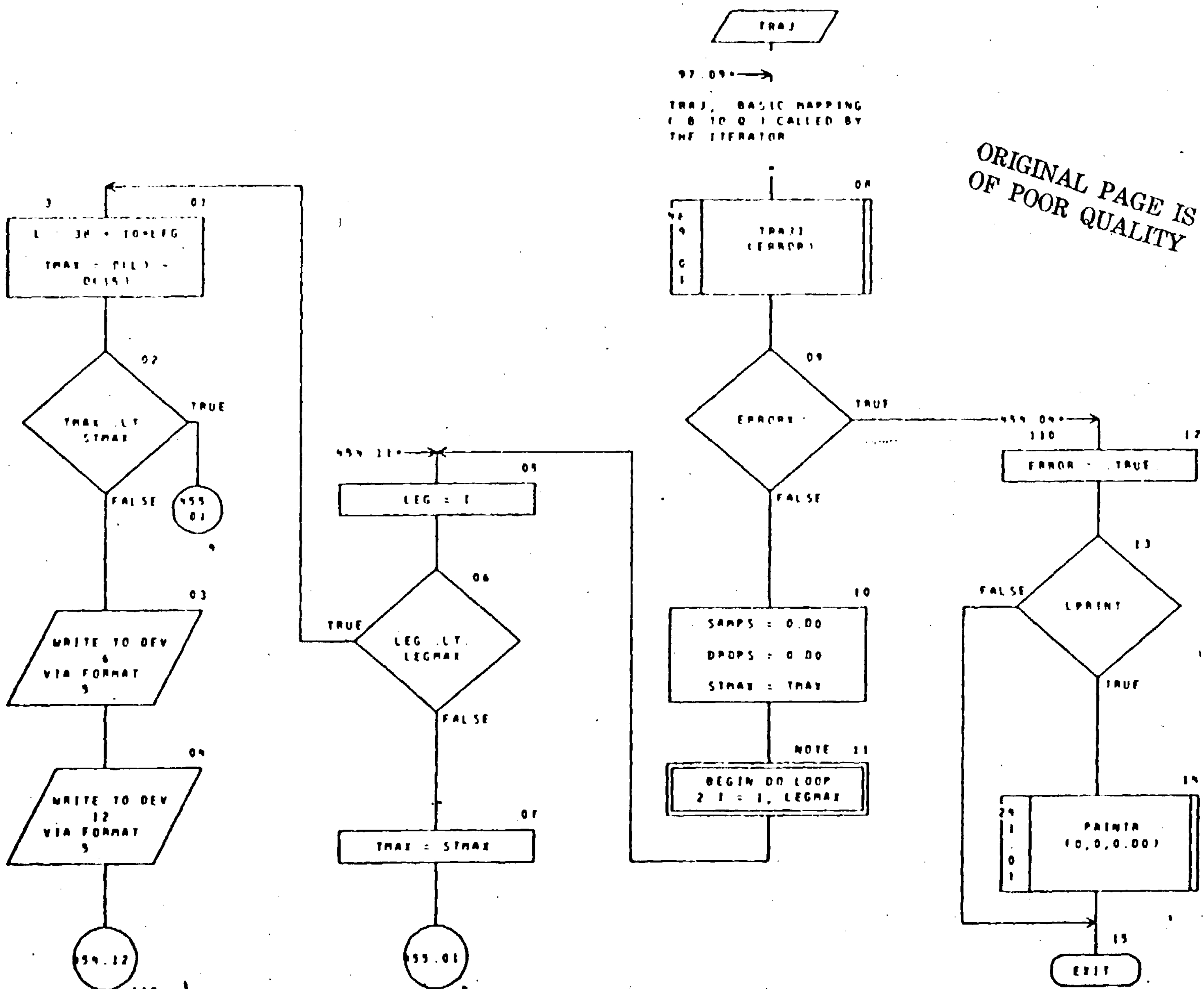
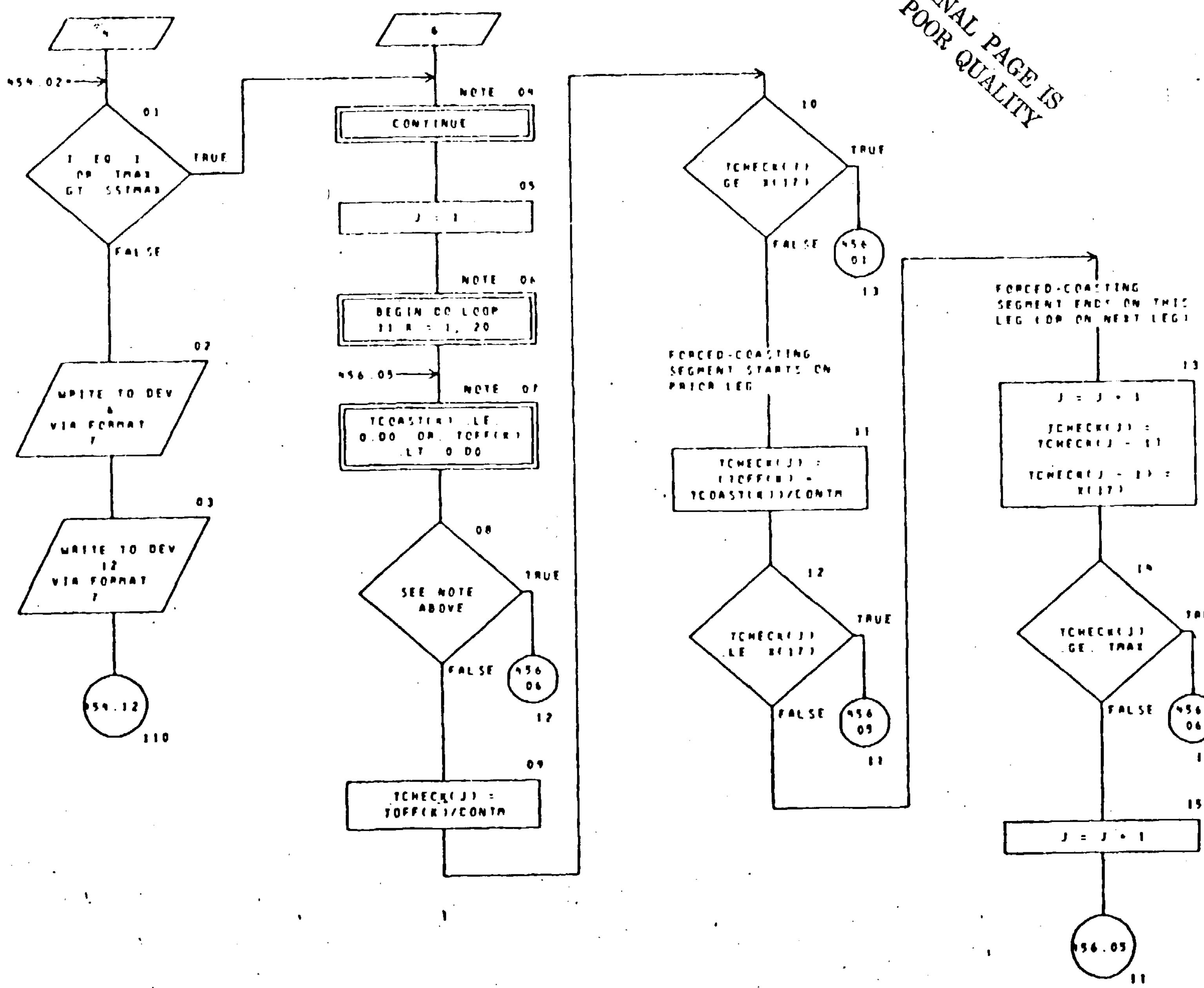


CHART TITLE - SUBROUTINE TRAJ(ERROR)

ORIGINAL PAGE IS OF POOR QUALITY



TRAJ-17

CHART TITLE - SUBROUTINE TRAJ(ERROR)

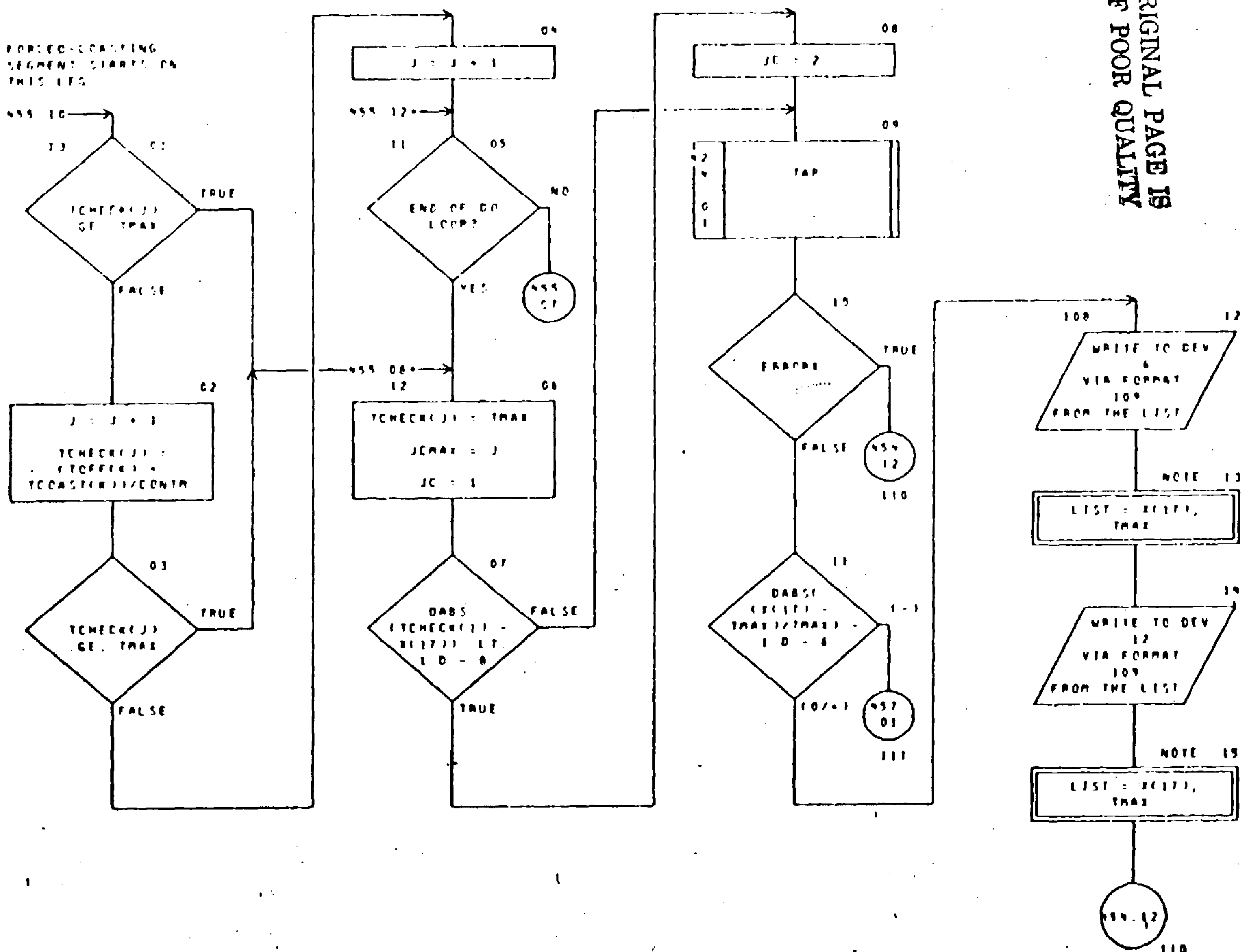
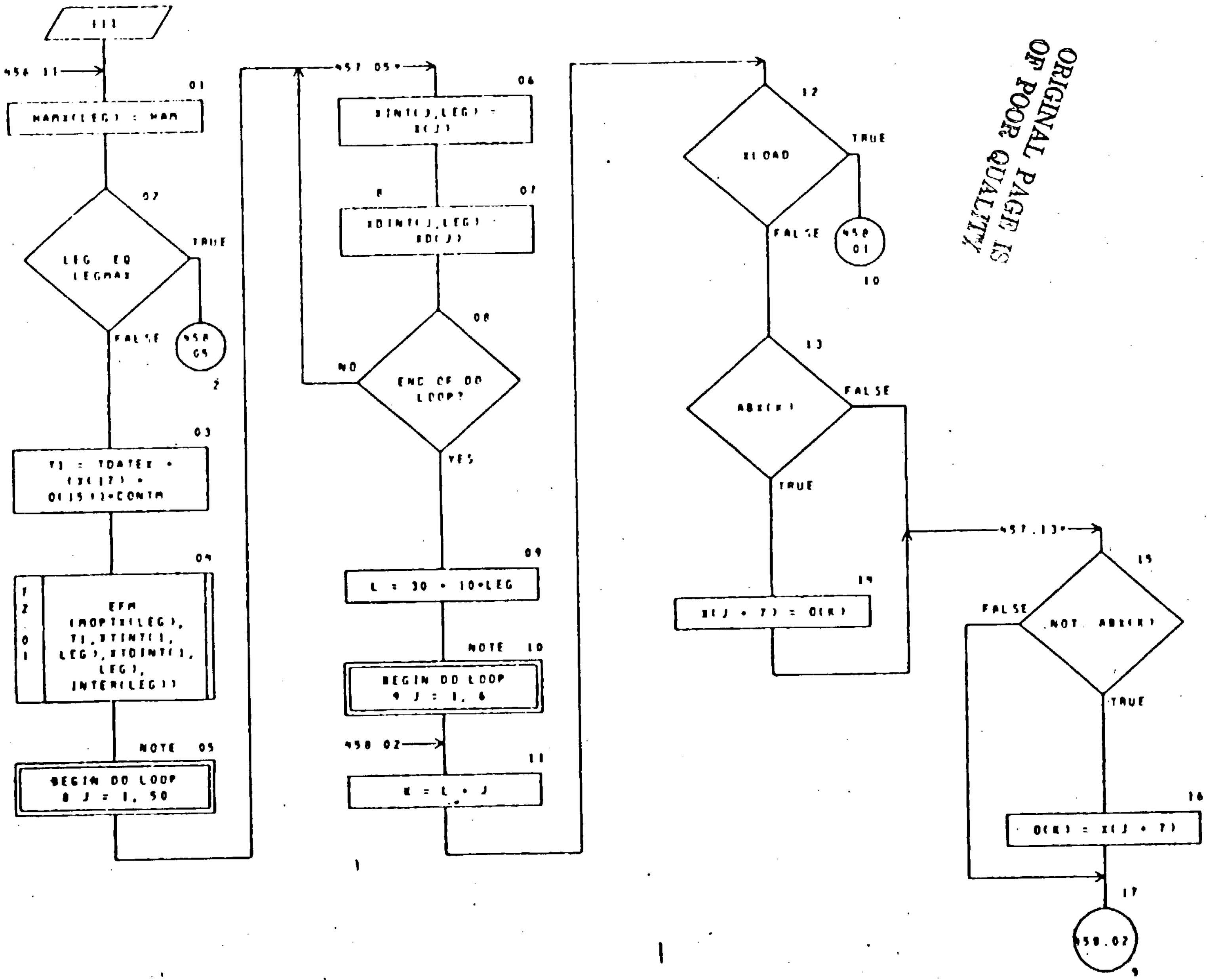


CHART TITLE - SUBROUTINE TRAJ(ERROR)



ORIGINAL PAGE IS OF POOR QUALITY

CHART TITLE - SUBROUTINE TRAJ(ERROR)

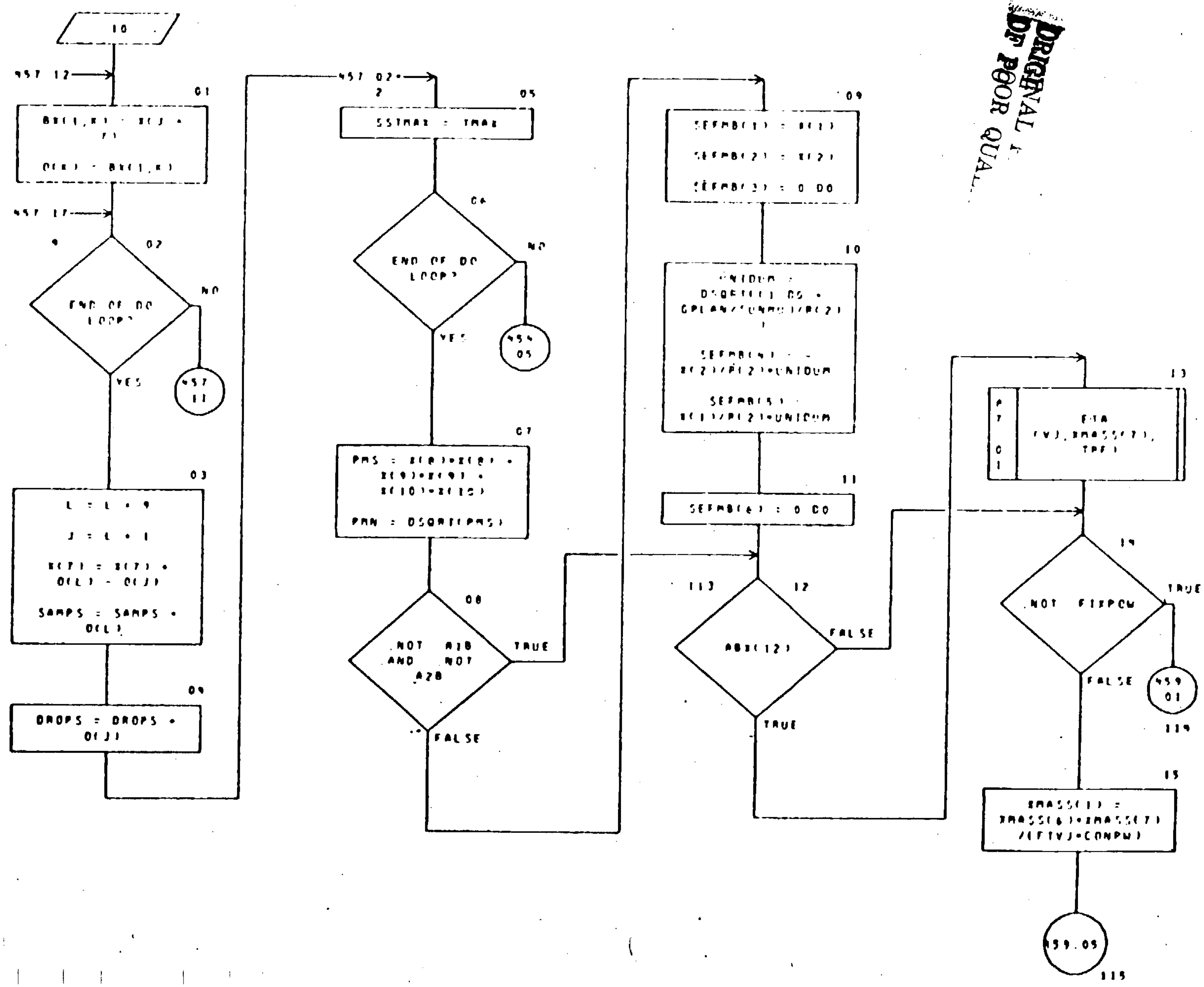


CHART TITLE - SUBROUTINE TRAJ(ERROR)

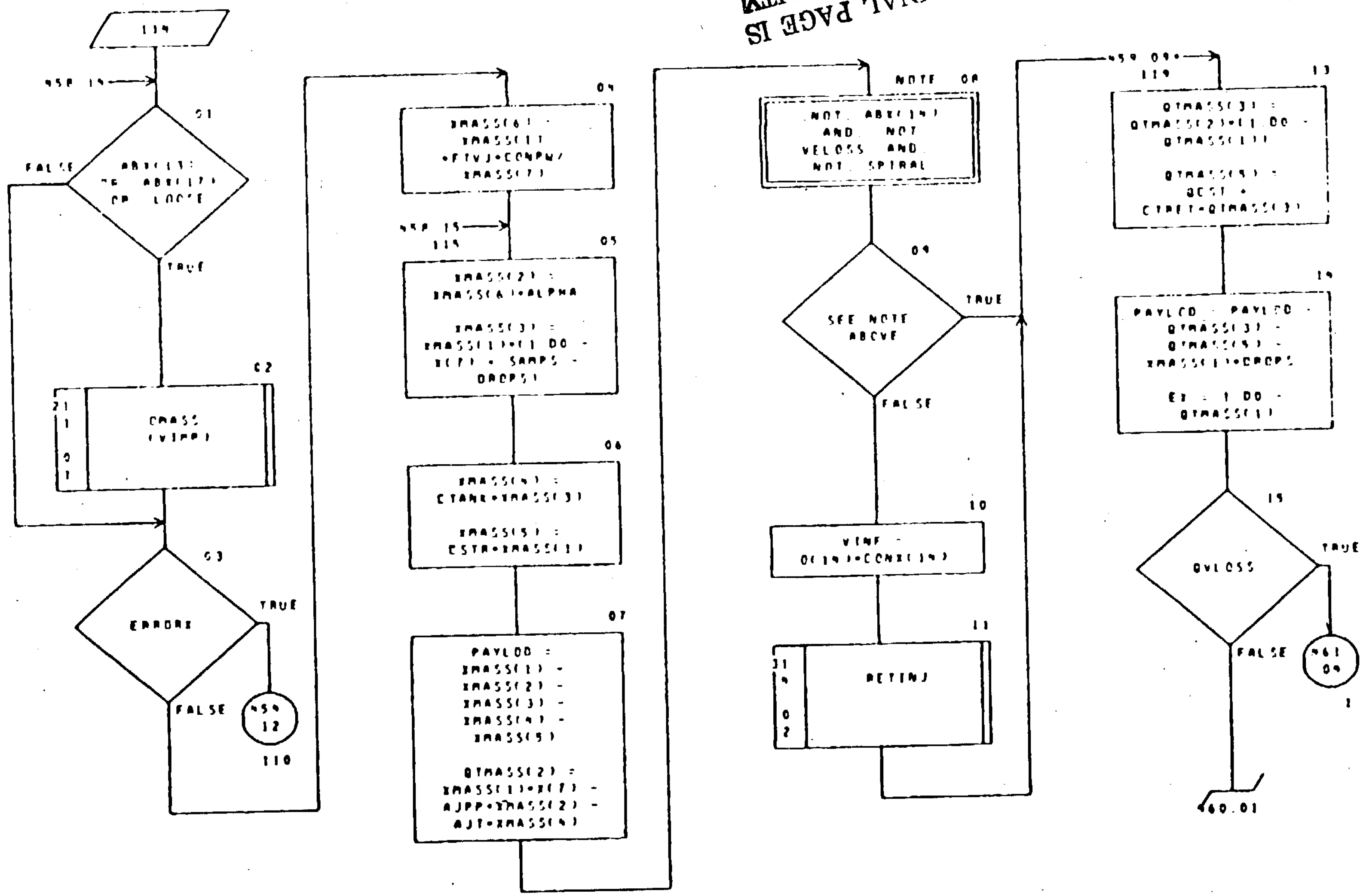


CHART TITLE - SUBROUTINE TRAJ(ERROR)

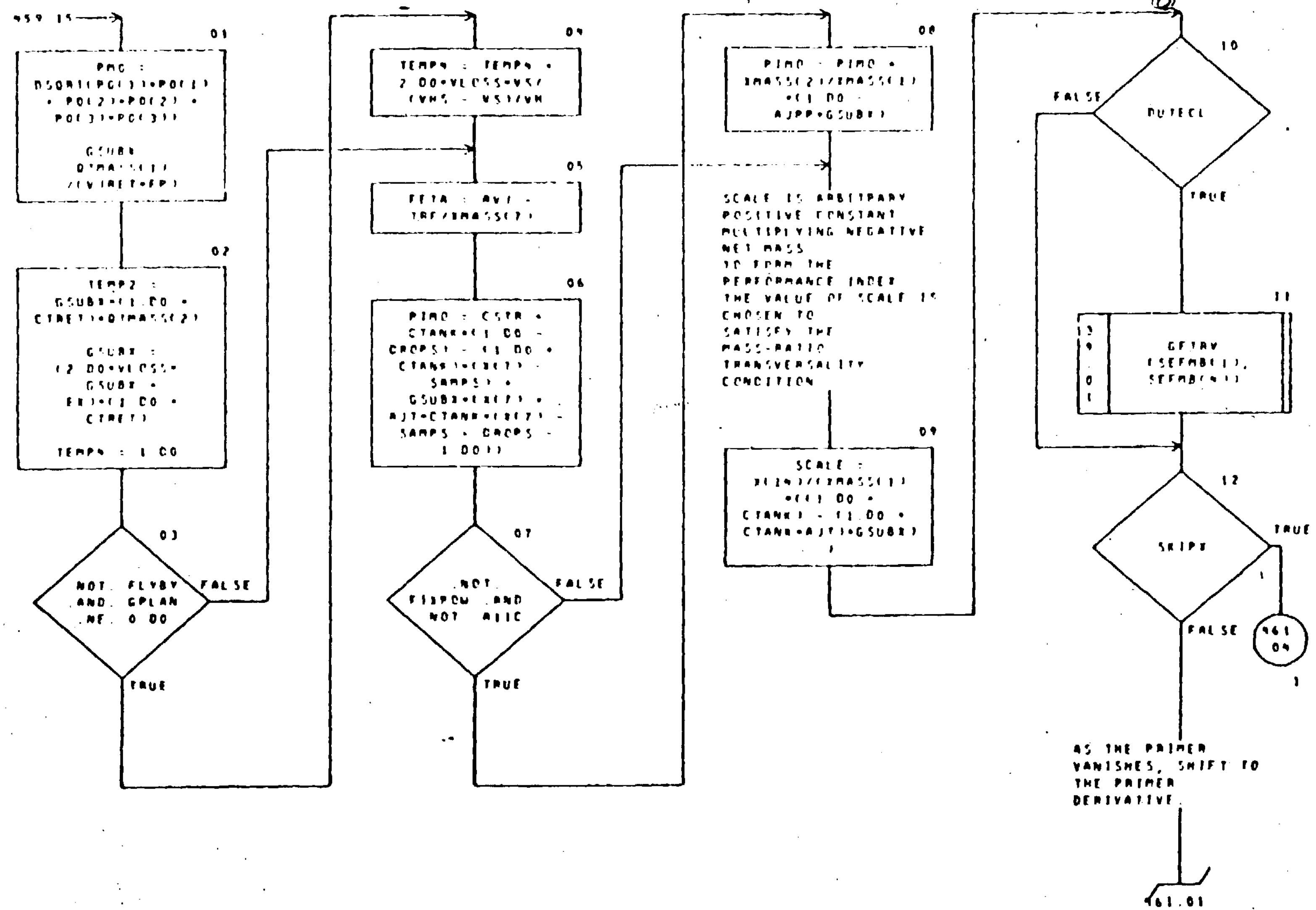
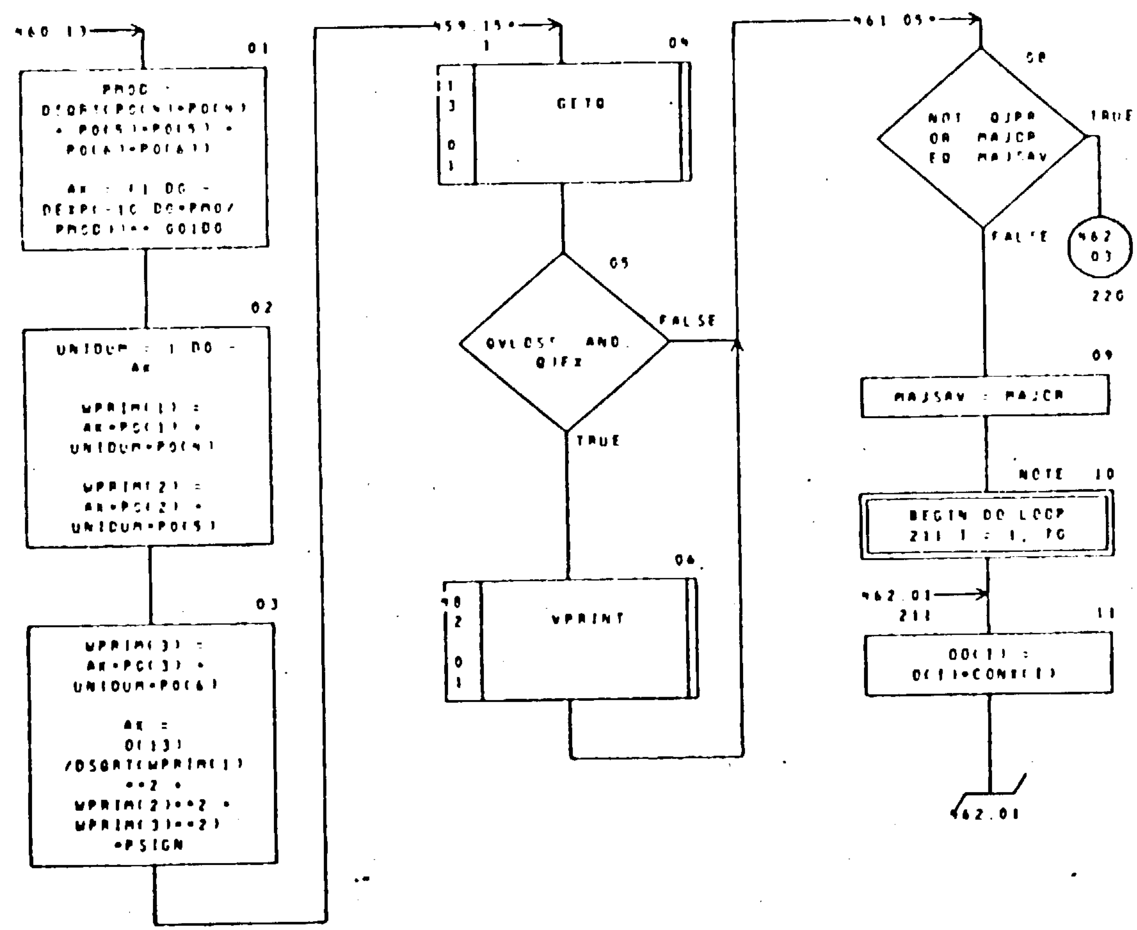


CHART TITLE - SUBROUTINE TRAJ(ERROR)



TRAJ-23

CHART TITLE - SUBROUTINE TRAJ(ERROR)

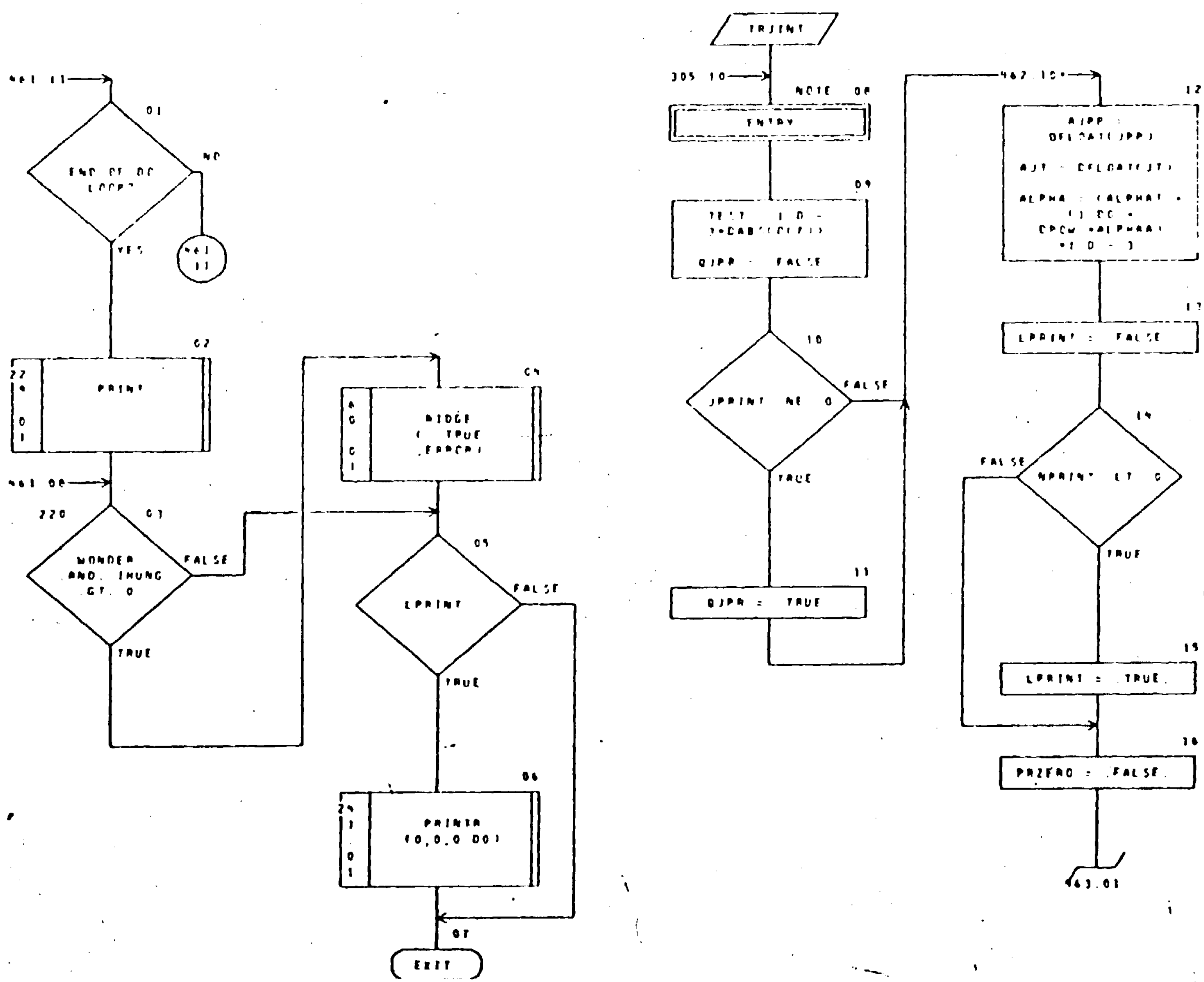
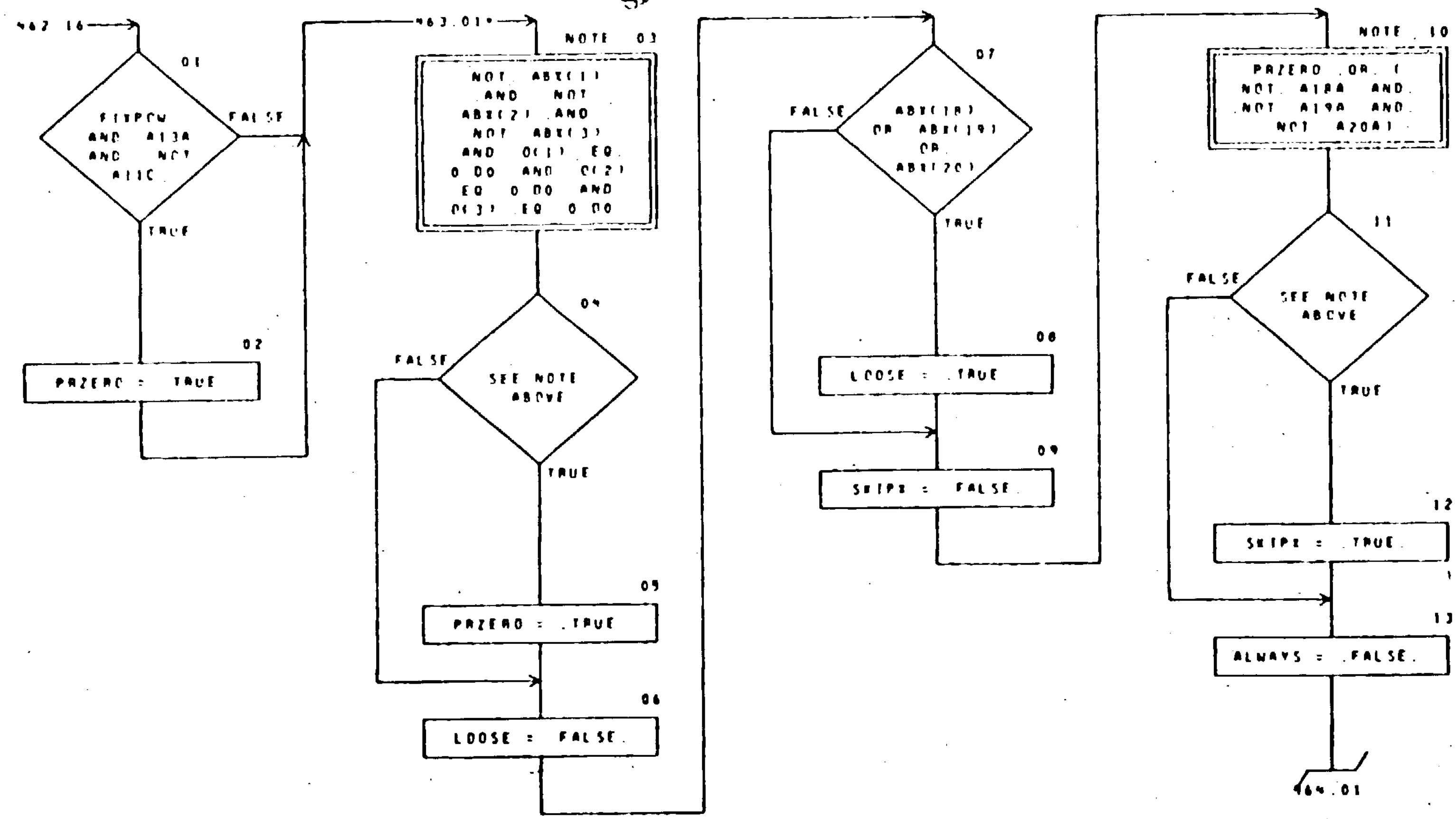


CHART TITLE - SUBROUTINE TRAJ(ERROR)



ORIGINAL PAGE IS
OF POOR QUALITY

01/08/75

CHART TITLE - SUBROUTINE TRAJERROR(

AUTOFLOW CHART SET - G.S.F.C. MILTOP DECEMBER 1974

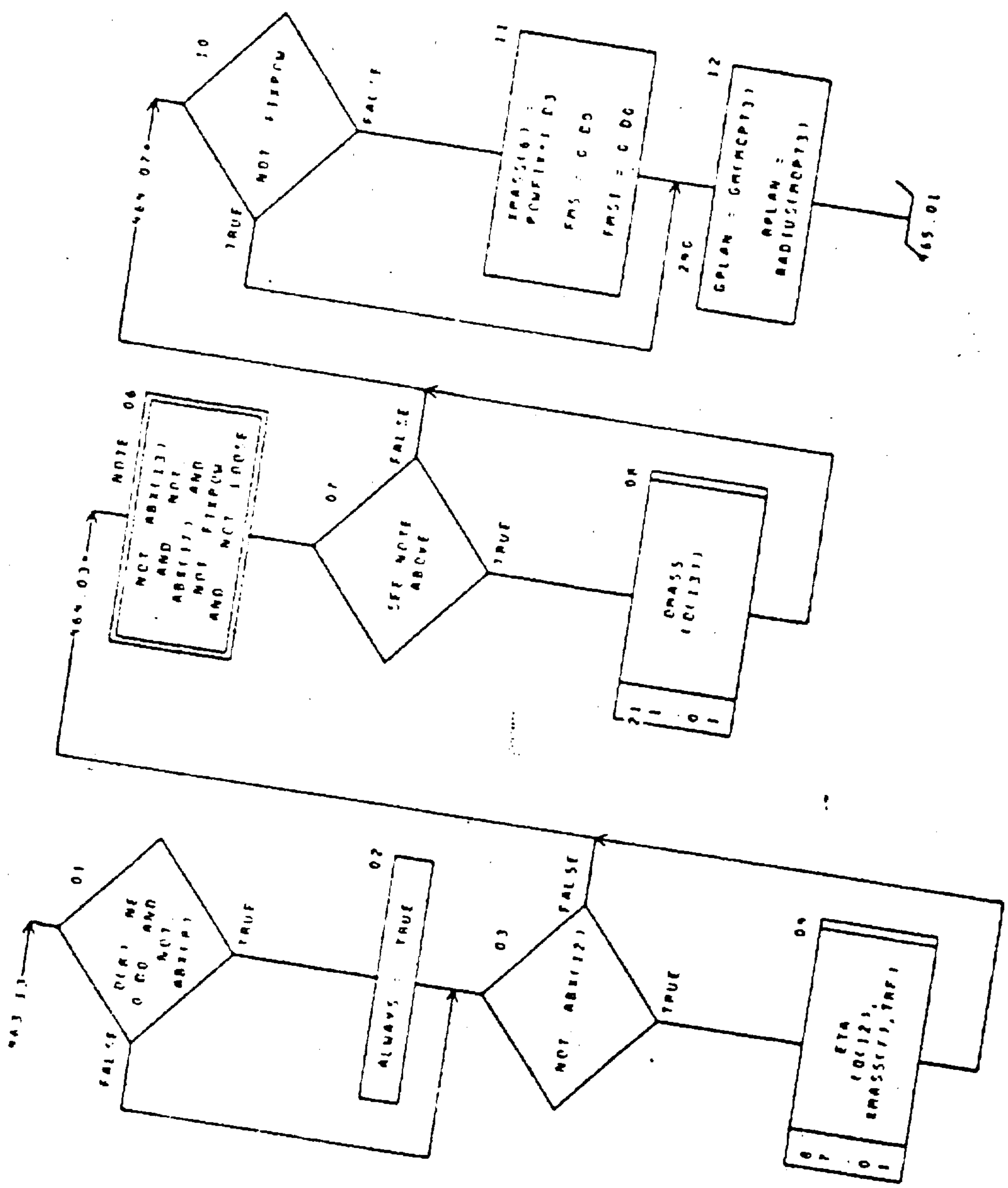


CHART TITLE - SUBROUTINE TRAJ(ERROR)

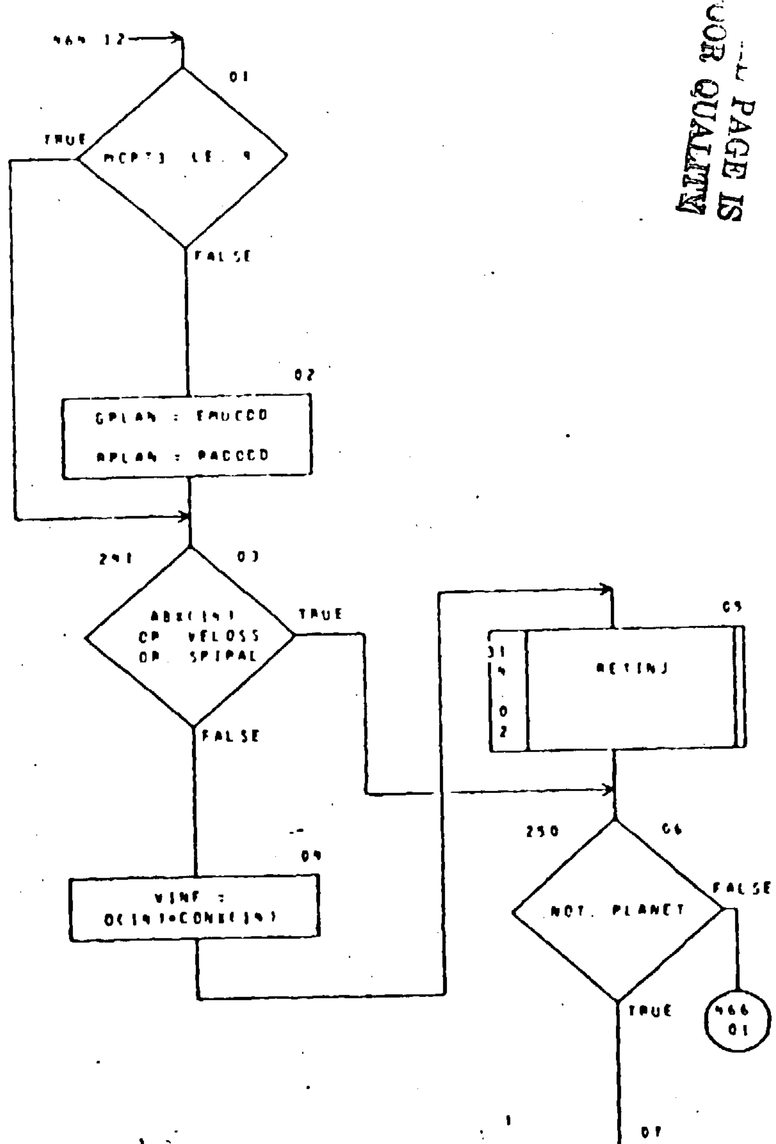


CHART TITLE - SUBROUTINE TRAJ(ERROR)

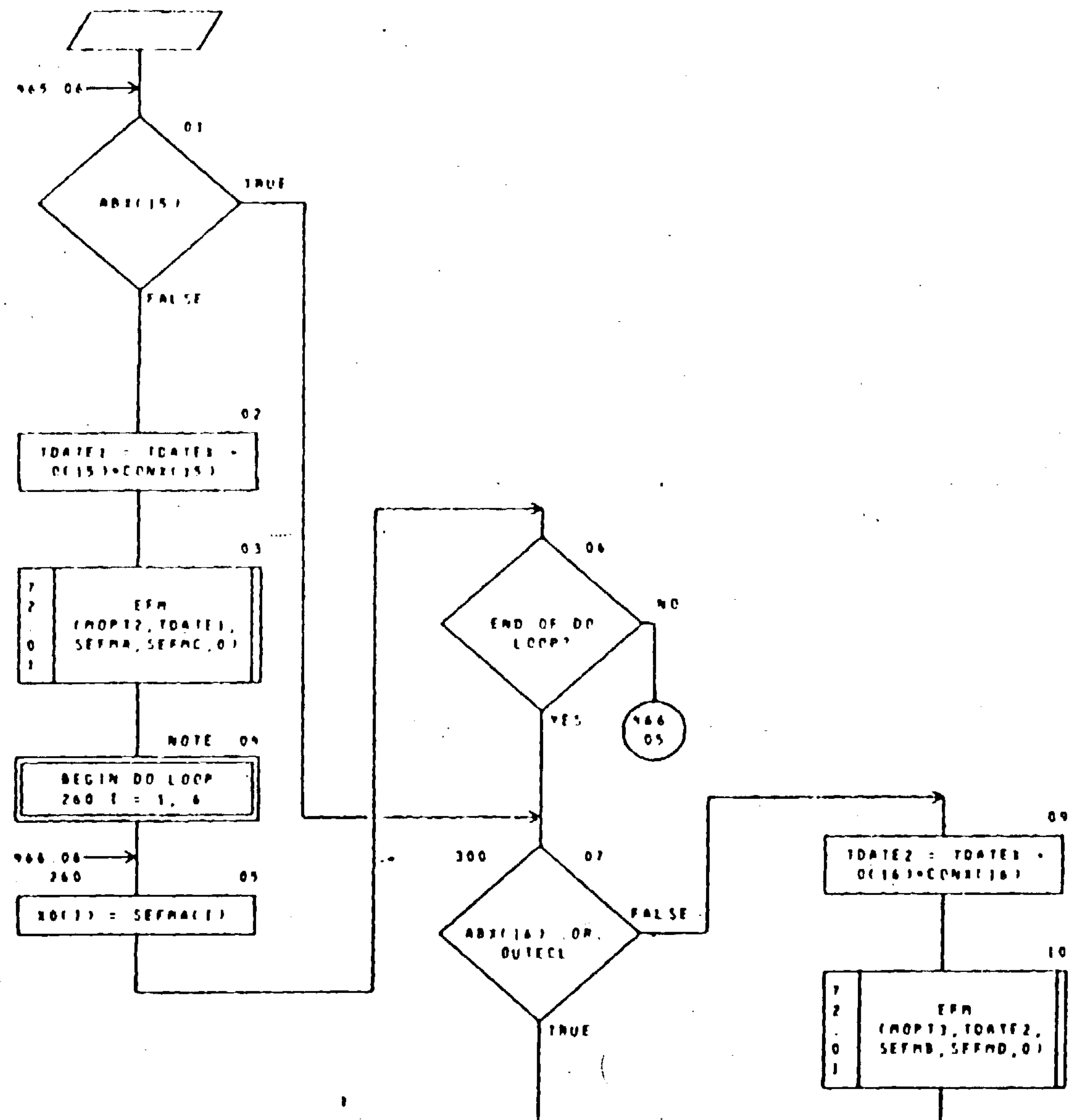


CHART TITLE - NON-PROCEDURAL STATEMENTS


```
IMPLICIT REAL*8 (A-H,O-Z)
LOGICAL ERROR,ERRORX,FLYBY,WONDER,OUTFCL,QVLOSS,FIXPOW,QJFX,PRZERO
,ABX,PLANET,SPIRAL,VELOSS,LOOSE,LPRINT,QJPR,XLOAD,SKIPX,ALWAYS,
A1B,A2B,A11C,A13A,A18A,A19A,A20A
COMMON /REAL8/ PAYLOD,XMASS(7),R01,CTANK,CSTR,R02,VJ,VIMP,
QIMASS(5),R03,CTRET,R04(5),VLOSS,R05(23),XO(7),PO(7),
R07(6),POWFIX,R08(12),EMUCOD,RADCOD,R09(3),TMAX,R10(36),SEFMA(7),
SEFMB(7),SEFMC(7),SEFMD(7),R11(2),TDATE1,TDATE2,TDATEX,R12(28),
GPLAN,PPLAN,R13(12),P(2),R14(63),ALPHAA,ALPHAT,R06(39),
PIMC,PSIGN,R33(6),                       SUNMU,CONTM,R15(2),CONPW,
                        R16(256),TOFF(20),TCOAST(20),TCHECK(41),R32(239)
,HAM,HAMX(4),R20(282),CPOW,R30(7),SAMPS,DROPS,R31(3)
,                          PMN,PMS,R18(4),FMS,R19(8)            ,EX,FP,QCST,
VH,VHS,VINF,VJRET,VS,R21,AVJ,FTVJ,R22,FMST,R23(2),AH,R24,PMC,PMSD,
TEST,GSUBX,TEMP2,R30   ,TEMP4,AJT,AJPP,FETA,SCALE,R25(6),WPRIM(3),
R26(53),X(50),XD(50),R27(25),XINT(50,5),XDINT(50,5),XTINT(6,5),
XTDINT(6,5),R29(65)
COMMON /INTGR4/    I01(5),JPRINT,I02(15),JPP,JT,MOPT2,MOPT3,
I04(16),NPRINT,I05(8),IHUNG,I06(15),MAJOR,I07(59),LEG,LEGMAX,
MOPTX(5),INTER(5),I08(193),JC,JCMAX,I03(667)
COMMON /LOGIC4/ ERRORX,L01,FIXPOW,L02,WONDER,L03(4),OUTFCL,L07,
QVLOSS,FLYBY,L05(3),SPIRAL,VELOSS,L04(7),PLANET,QJFX,L06(2),
LOOSE,L08(3),PRZERO,XLOAD,L17(6),ALWAYS,L18(109),A1B,L09(2),
A2B,L10(19),A11C,L11(2),
A13A,L12(31),A18A,L13,A19A,L14,A20A,L15(11),ABX(70),L16(205)
```

TRAJ-29

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
          COMMON /ITERAT/ BXCS,70),BO1(210),CONX(70),BO2(70),D(70),DD(70),
          BO3(190)
          COMMON /SOLSYS/ GM(70),RADIUS(70),SOLOI(INC)
          DATA MAJSAV /0/
 5        FORMAT(1H ,52HINTERMEDIATE TIME GREATER THAN FINAL TIME /1H ,
          16HCASE TERMINATED )
 7        FORMAT(1H ,35HTARGET PASSAGE TIMES NOT ASCENDING /1H ,
          16HCASE TERMINATED )
 109      FORMAT(1H0,19HFINAL TIME VIOLATED2D24.16)
```

ORIGINAL PAGE IS
OF POOR QUALITY

<u>Name</u>: TRAJI

<u>Calling Argument</u>: ERROR

<u>Referenced Sub-programs</u>: CHECKI, EFM, RADAR, RIDGE, SOLAR

<u>Referenced Commons</u>: INTGR4, ITERAT, ITER2, LOGIC4, REAL8

<u>Entry Points</u>: None

<u>Referencing Sub-programs</u>: TRAJ

<u>Discussion</u>: Subroutine TRAJI basically consists of logic originally broken out of the top portion of subroutine TRAJ for the purpose of reducing the size of TRAJ.

TRAJI performs the initialization of the trajectory-starting quantities; in particular, the active iterator independent variables are employed at this point to initialize the applicable trajectory starting parameters, especially the array of integrated trajectory dependent-variables X(i).

If the iterator is allowed to vary the time of launch or the time of arrival at the primary target, then the analytic ephemeris routine is invoked to yield updated values for the state vectors of the pertinent celestial bodies.

Initialization is performed within subroutine RADAR, which computes communication distance and angle, if the current trajectory is the final, case-summary trajectory of a given case. Entry point CHECKI of subroutine CHECK is called, to initialize CHECK for the current trajectory. When power degradation is simulated, subroutine SOLAR is called in order to determine if the optimum solar array tilt-angle corresponds to operating on the power-function curve γ or below it.

<u>Messages and printouts</u>: If the iterator drives the initial primer vector Λ_o toward the origin of primer-space, i.e., toward the zero-vector, which is a singularity of the optimal rocket problem and which causes severe numerical difficulty when the thrust switch function is positive, preventing the two-point boundary value problem from converging and wasting machine time, then the program will detect this condition via a simple test of the magnitude of Λ_o, and will declare an error condition

and exit the subroutine after printing the diagnostic message on units 6 and 12:

INITIAL PRIMER CONVERGING TO ZERO

The iterator may decide to try several such trajectories before giving up, and so the message may appear several times. The analyst should consider reversing the value of program input quantity PSIGN when this problem occurs.

If the iterator drives the magnitude of the launch hyperbolic excess velocity $v_{\infty o}$ = O(13) to become negative, which is physically meaningless, then the program will declare an error condition and exit the subroutine after printing the diagnostic message on units 6 and 12:

DEPARTURE V00 IS NEGATIVE (X13)

The iterator may decide to try several such trajectories before giving up, and so the message may appear several times. The analyst should consider trying a significantly different set of values as initial guesses for the iterator independent variables when this problem occurs.

TRAJI EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
B(35)	U	ITER2	Array of active iterator independent-variables.
O(70)	SU	ITERAT	Array of iterator independent-variables, both passive and active.
X(50)	SU	REAL8	Array of trajectory dependent-variables, as described in subroutine RKSTEP.
CE	U	REAL8	Cosine of obliquity of ecliptic, cos ϵ.
FT	SU	REAL8	Reference thrust acceleration, g, in AU/tau^2.
LL(70)	U	INTGR4	Index set of active iterator independent-variables.

TRAJI EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
P0(7)	SU	REAL8	Array of initial adjoint variables Λ_o, $\dot{\Lambda}_o$, $\lambda_{\nu o}$.
RT	S	REAL8	Spacecraft solar distance, r, in AU.
SE	U	REAL8	Sine of obliquity of ecliptic, sin ϵ.
VJ	SU	REAL8	Jet exhaust speed, c, in EMOS.
X0(7)	SU	REAL8	Initial launch planet state P_o, $\dot{P}_o$; and initial mass ratio ν_o.
ABX(70)	U	LOGIC4	Array of iterator independent-variable indicators, which selects the active variables.
AVJ	SU	REAL8	Inverse of jet exhaust speed, 1/c, in $EMOS^{-1}$.
LXX	U	INTGR4	The number of (active) iterator independent-variables.
NSW	S	INTGR4	Counter of the number of thrust switching points encountered along the trajectory.
PMN	SU	REAL8	Magnitude of primer vector, λ.
PMS	SU	REAL8	Square of magnitude of primer vector, λ^2.
TAU	S	REAL8	Propulsion time, τ, in tau.
ANG1	U	REAL8	Launch site latitude, in radians.
BETA	S	REAL8	Trajectory independent-variable, β.
FTVJ	S	REAL8	Auxiliary parameter gc, in AU^2/tau^3.
HEAT	U	LOGIC4	Indicator that the solar panels are maintained normal to the sun line at all times.

TRAJI EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
PM0D	SU	REAL8	Magnitude of initial primer derivative, $\lvert\dot{\lambda}_o\rvert$.
QJEX	U	LOGIC4	Detailed printout indicator.
TEST	U	REAL8	Tolerance value in test for zero initial primer vector.
TMAX	S	REAL8	Integration stopping condition: time of flight from launch to primary target, in tau.
VIMP	SU	REAL8	Launch hyperbolic excess speed, $v_{\infty o}$, in AU/tau (= EMOS).
V00D(3)	SU	REAL8	Launch hyperbolic excess velocity, $V_{\infty o}$, in AU/tau.
ALTAU	SU	REAL8	Propulsion-time adjoint variable, λ_{τ}.
ANGLE	S	REAL8	Travel angle, θ_t, in radians.
CHFNC	U	REAL8	A function whose roots determine the switch points of the array tilt angle to and from the power-curve boundary, f_{ch1}.
CONTM	U	REAL8	Time conversion factor, tau to days.
ERODE	U	LOGIC4	Power degradation option indicator.
ERROR	SAX		Error-condition indicator.
FTOVJ	S	REAL8	Auxiliary parameter g/c, in tau^{-1}.
LDECL	SU	LOGIC4	Indicator for the condition in which the magnitude of the departure asymptote declination exceeds the parking orbit inclination.

TRAJI EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
LOOSE	U	LOGIC4	Indicates that the initial heliocentric spacecraft velocity is included in the (active) iterator independent variables.
MOPT2	A	INTGR4	Launch planet number.
MOPT3	A	INTGR4	Planet-number of primary target.
NSPEC	S	INTGR4	Counter for storage arrays associated with the Extremum Table of Selected Functions.
PCURV	S	LOGIC4	Indicator for condition in which solar panels are in position to receive maximum power, when degradation option is invoked.
PSIGN	U	REAL8	Coefficient defining the sense of the launch hyperbolic excess velocity relative to the initial primer vector.
QDECL	U	LOGIC4	Non-coplanar launch maneuver indicator.
SEFMA(7)	SUA	REAL8	Array containing position and velocity of launch planet at launch time, P_o and $\dot{P}_o$, in AU and EMOS, respectively.
SEFMB(7)	SA	REAL8	Array containing position and velocity of primary target at time of target intercept, P_n and $\dot{P}_n$, in AU and EMOS, respectively.
SEFMC(7)	A	REAL8	Time derivative of SEFMA array (w.r.t. tau^{-1}).
SEFMD(7)	A	REAL8	Time derivative of SEFMB array (w.r.t. tau^{-1}).
DECLAM	SU	REAL8	Geocentric declination of initial primer vector, δ_λ, in radians.
ERRORX	S	LOGIC4	Program master error indicator.

TRAJI EXTERNAL VARIABLES TABLE (cont)

Variable	Use	Common	Description
FIXTAU	S	LOGIC4	Indicator for non-zero λ_τ.
FIXTHR	U	LOGIC4	Indicator for fixed (i.e., constant) thrust cone angle.
LIMPHI	U	INTGR4	Highest index-value of all non-zero (multiple) fixed thrust-cone-angles; highest permissible value is currently one.
MAXPOW	U	LOGIC4	Indicator for mode of operation in which solar panels are maintained in orientation to receive maximum permissible power, when degradation option is invoked.
NSTEP1	S	INTGR4	Number of thrust computation steps associated with the current trajectory.
NSTEP2	S	INTGR4	Number of coast computation steps associated with the current trajectory.
OUTECL	U	LOGIC4	Indicator for out-of-ecliptic mission.
PLANET	U	LOGIC4	Ephemeris-option indicator.
PRZERO	U	LOGIC4	Indicator that zero initial primer vector is the desired condition.
TANGLE	S	REAL8	Elapsed ecliptic longitude, θ_ϵ, in radians.
TDATEX	U	REAL8	Reference Julian date.
TDATE1	SA	REAL8	Julian date at time of launch.
TDATE2	SA	REAL8	Julian date at time of primary target intercept.

CHART TITLE - SUBROUTINE TRAJI(ERROR)

ORIGINAL PAGE IS OF POOR QUALITY

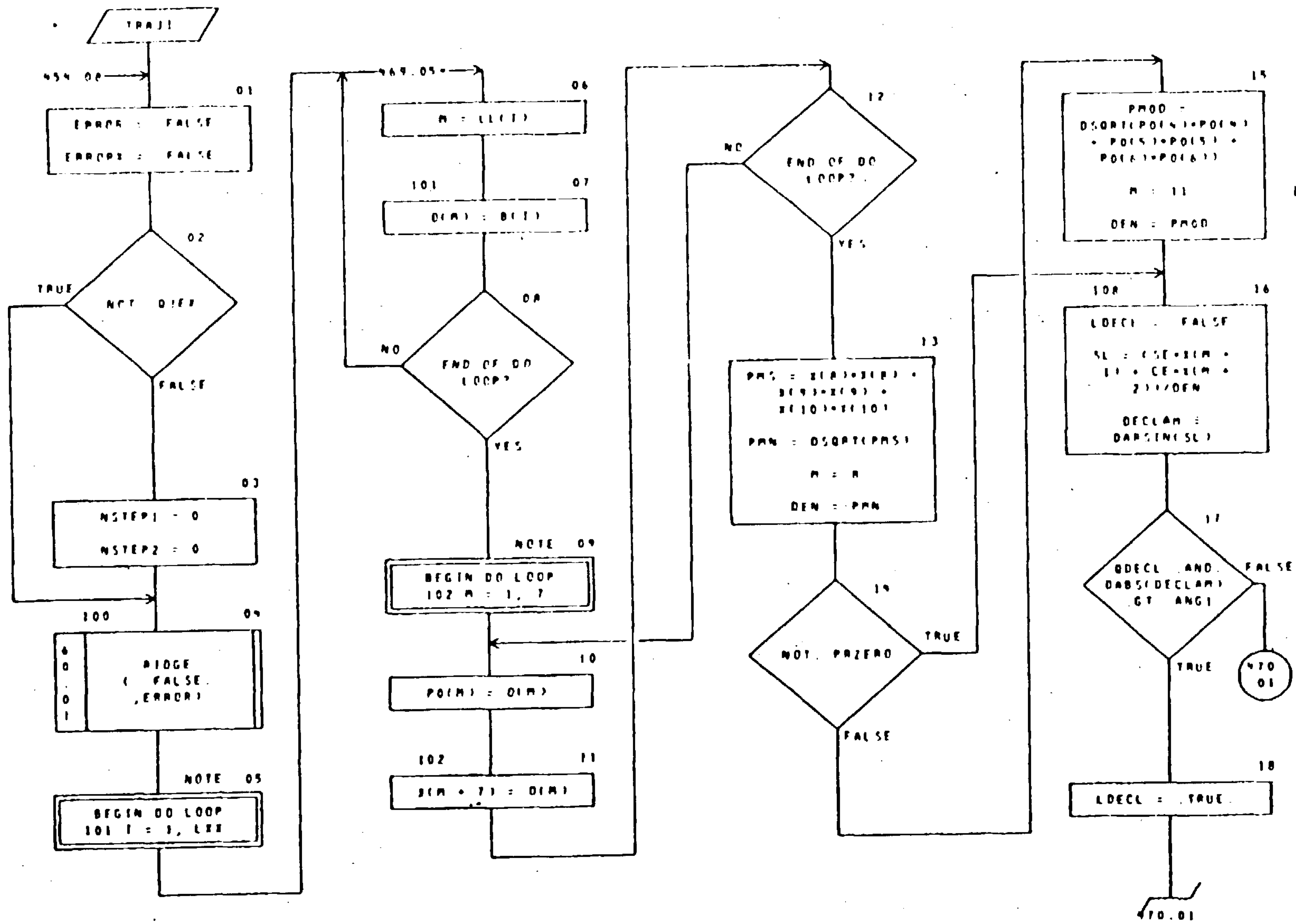


CHART TITLE - SUBROUTINE TRAJ(ERROR)

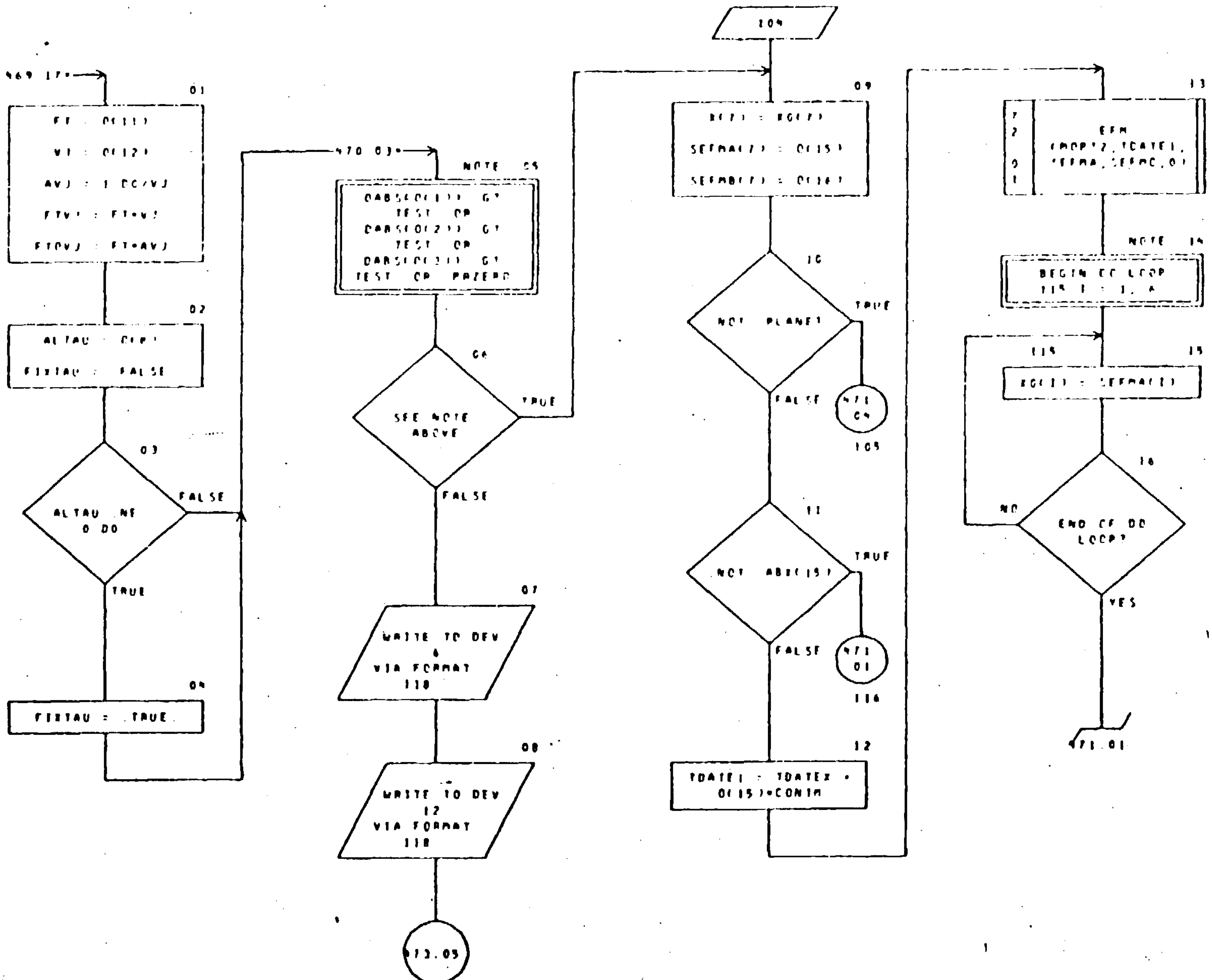


CHART TITLE - SUBROUTINE TRAJ(ERROR)

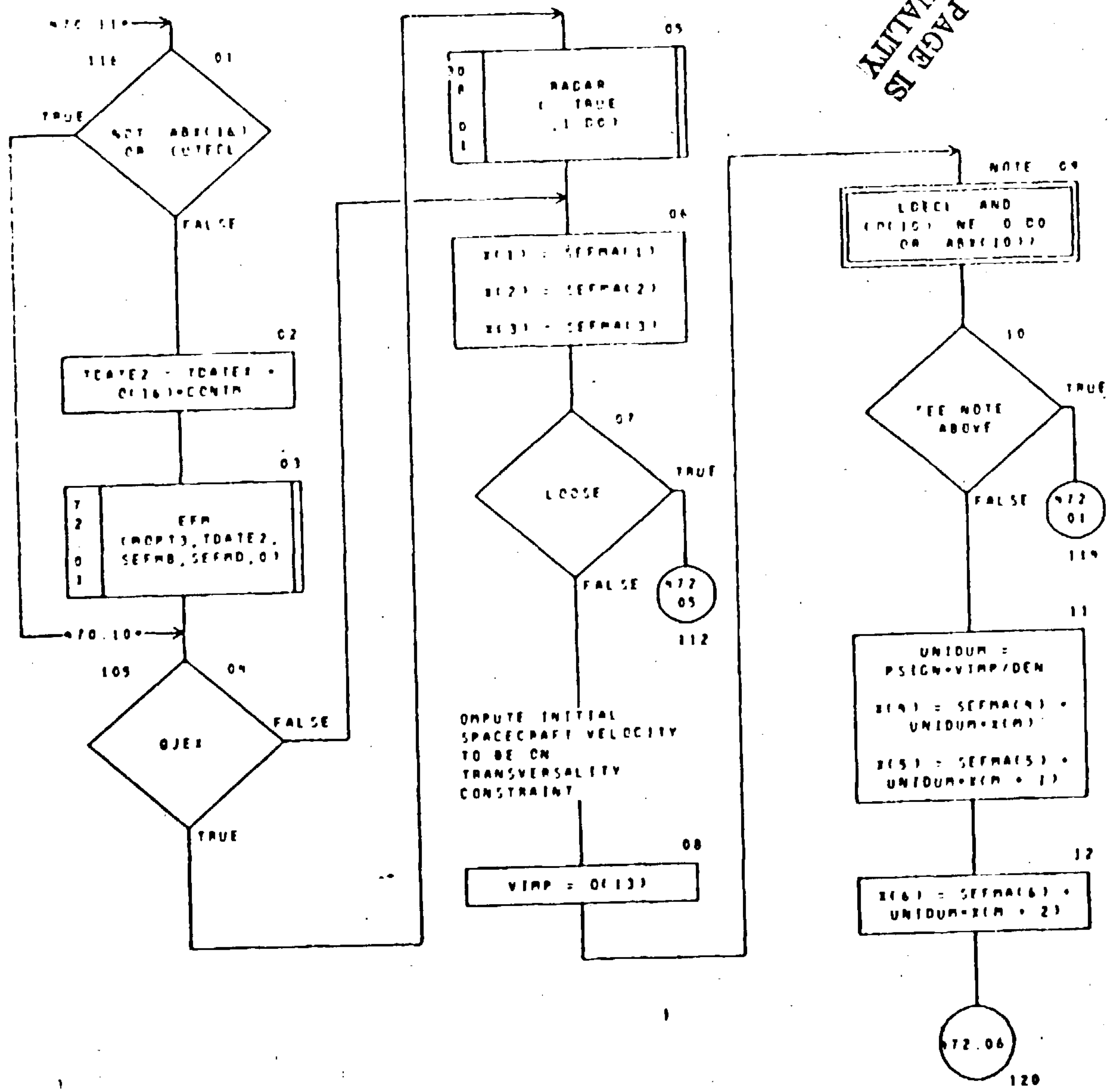


CHART TITLE - SUBROUTINE TRAJ(ERROR)

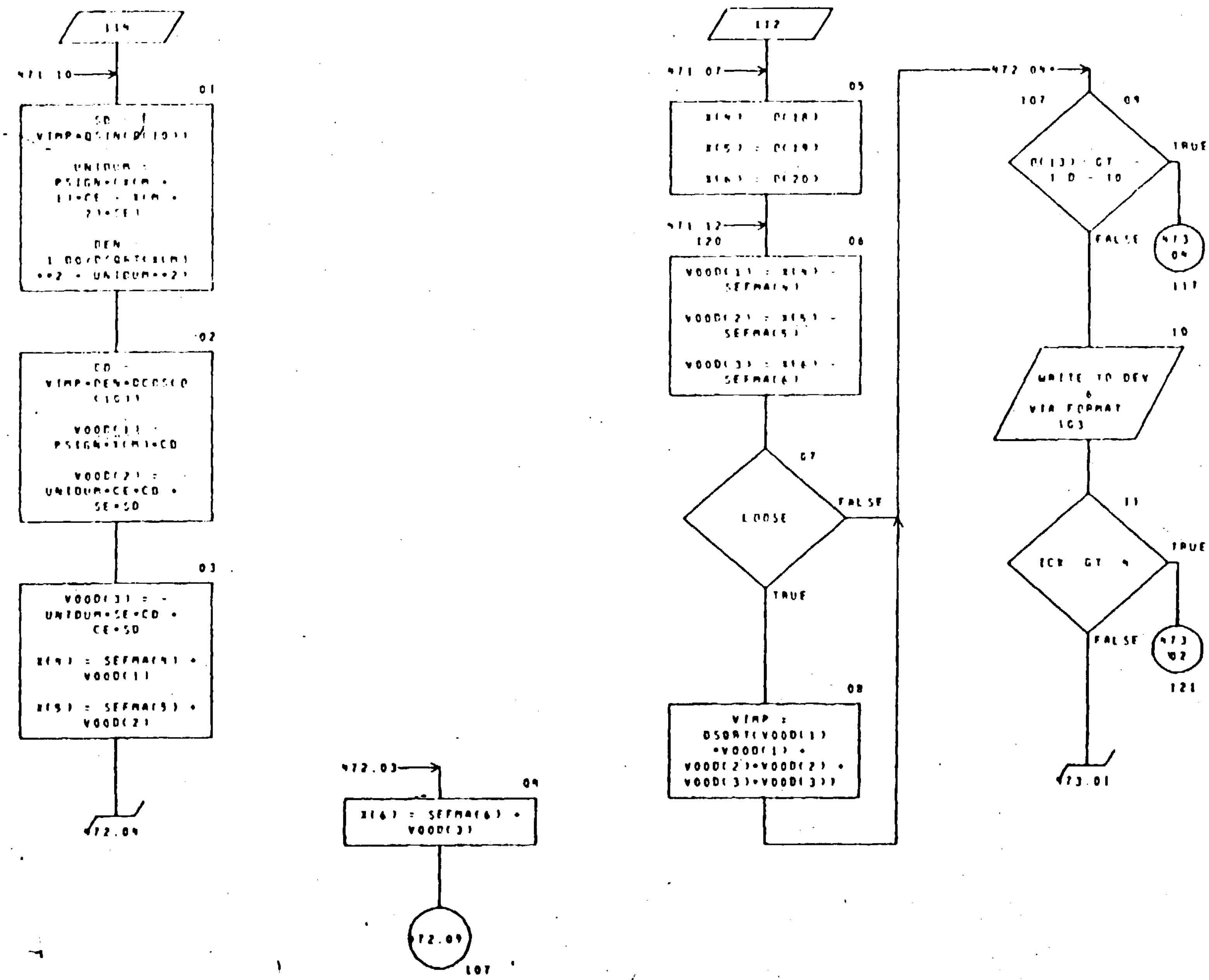
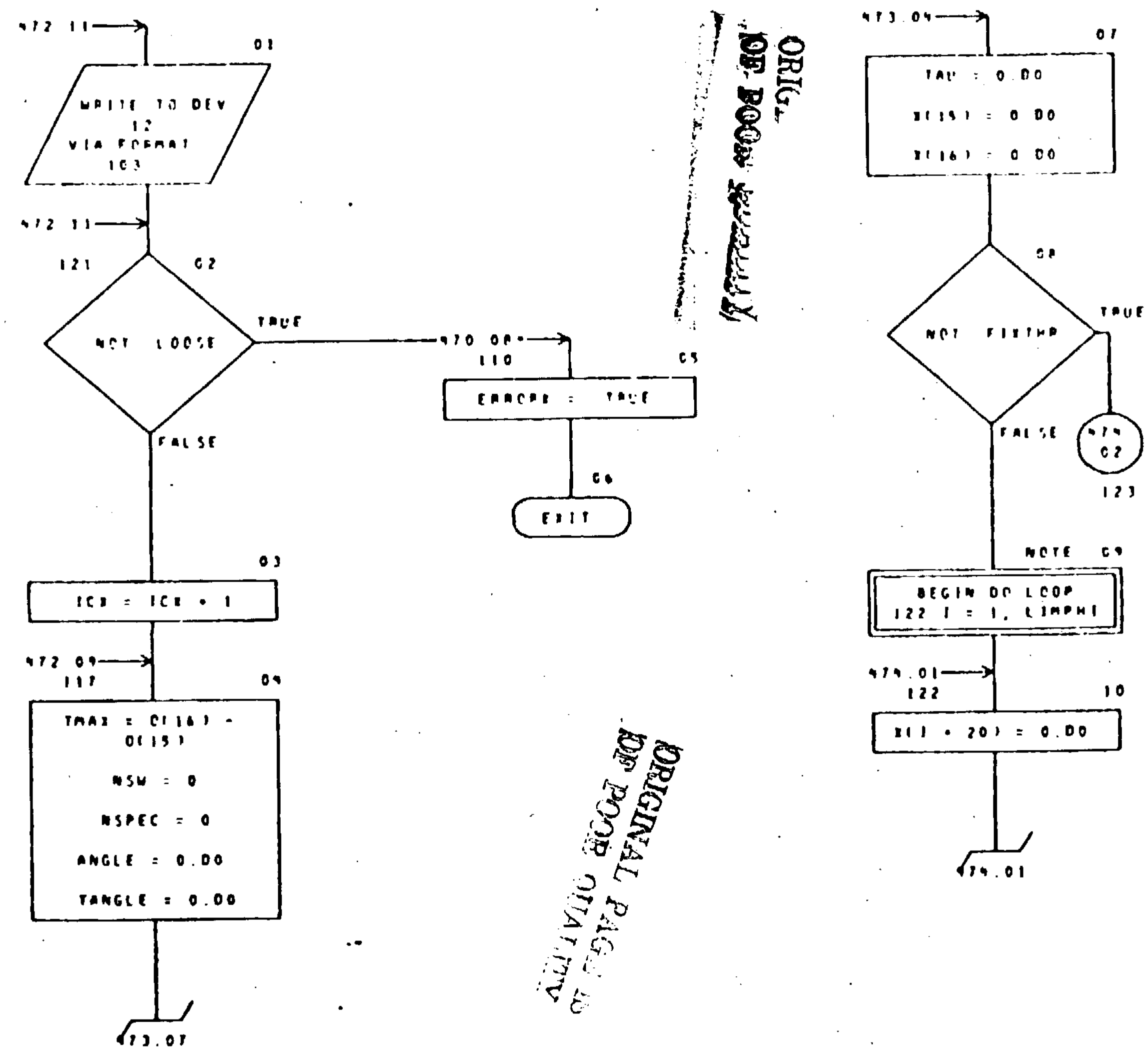


CHART TITLE - SUBROUTINE TRAJI(ERROR)



ORIGINAL PAGE IS
OF POOR QUALITY

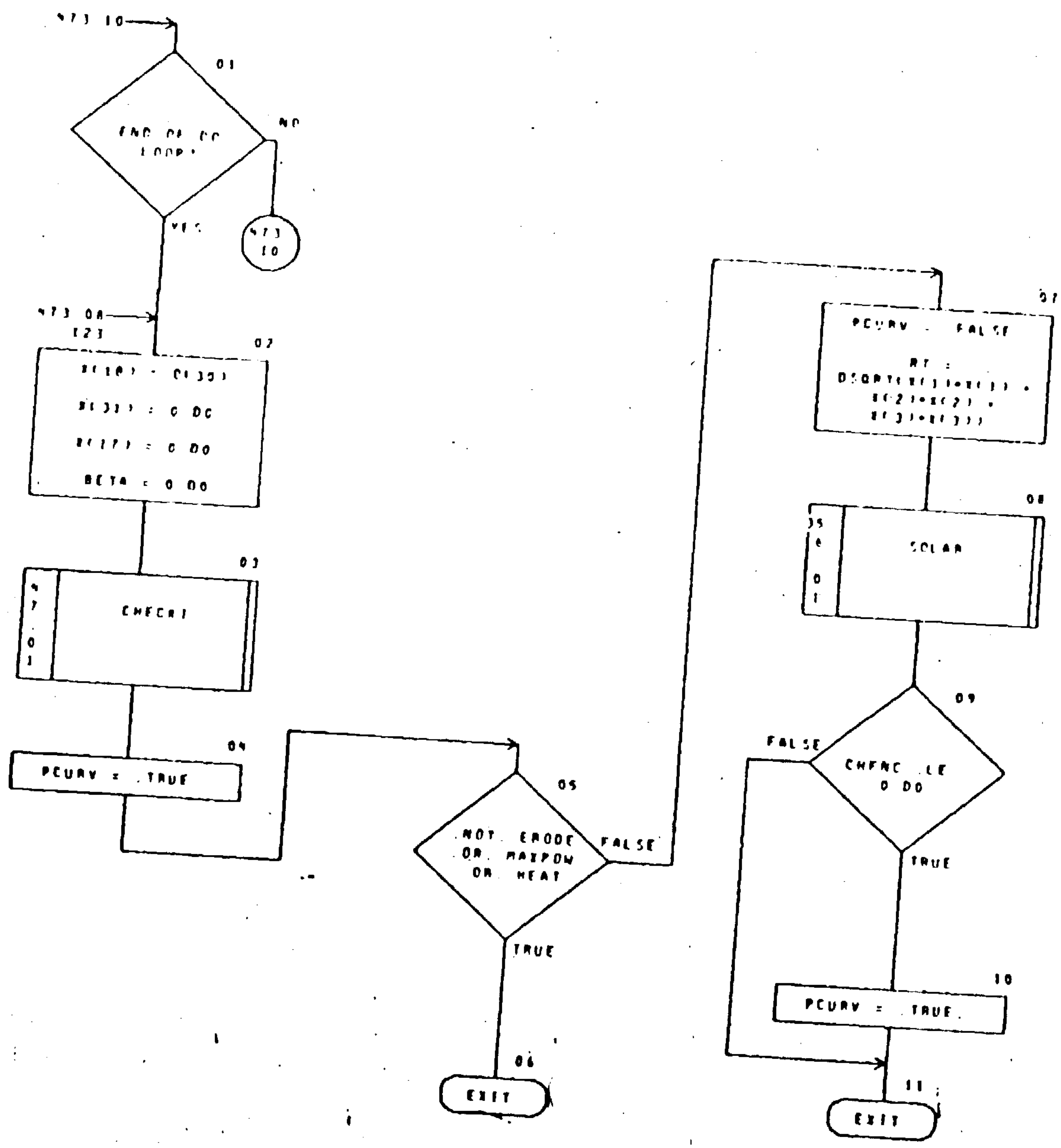


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL ERROR,ERRORI,OJEX,PRZERO,ABT,PLANET,FIXTHR,FIXTAU,LOOSE,
      ERODE,MAXPOW,HEAT,PCURV,LDECL,ODECL,OUTECL
      COMMON /REAL8/ RO1(11),FT,VJ,VIMP,RO2(36),XO(7),PO(7),RO3(24),TMAX
      ,RO4(36),SEFMA(7),SEFMB(7),SEFMC(7),SEFMD(7),ANGLE,RO5,TDATE1,
      TDATE2,TCATEX,RO6(5),TANGLE,RO7(3),ANG1,R17(2),SE,CE,RIN(3),
      ALTAU,RO8(127),VOGO(3),RO9,PSIGN,R22,DECLAM,R19(5),CONTM,
      R21(823),CHFNC,R16(5),PMN,PMS,RT,R10(8),TAUPOW,R20(12),
                               AVJ,FTVJ,FTOVJ,R11(6),
          PMOD,TEST,R12(70),
      X(50),R13(51),BETA,RIN,TAU,R15(696)
      COMMON /INTGRN/ IO1(23),MOPT2,MOPT3,IO7(12),NSPEC,IO2(10),LIMPHI,
      IO3(12),LXX,IO4(12),NSW,IO5(110),LL(70),IO6(74),NSTEP1,NSTEP2,
      IO8(669)
      COMMON /LOGIC4/ ERRORI,LO1(6),ODECL,LO9,OUTECL,
                                         FIXTHR,LO2(8),ERODE,LO3(5),PLANET,
      OJEX,FIXTAU,HEAT,LOOSE,LO5(3),PRZERO,LO6(2),PCURV,MAXPOW,
      LO4(4),LDECL,LO8(102),ABT(70),LO7(205)
      COMMON /ITERAT/ BO1(700),O(70),BO2(210)
      COMMON /ITER2/ B(35),PO1(1505)
      DATA ICX /0/
 110  FORMAT(1HO,33HINITIAL PRIMER CONVERGING TO ZERO)
 103  FORMAT(1H ,33HDEPARTURE VOO IS NEGATIVE (X13))
```

Name: TRAVEL

Calling Argument: X, Y, XM, YM, ANGLE

Referenced Sub-programs: None

Referenced Commons: REAL8

Entry Points: None

Referencing Sub-programs: LOAD, TAP

Discussion: TRAVEL is used in the approximate integration of two auxiliary angles calculated for printout use. The first angle, termed travel angle, is defined

$$\theta_t = \Sigma_i \, \theta_i$$

where

$$\theta_i = \cos^{-1}\left(\frac{X \cdot Y}{xy}\right)$$

On each call to TRAVEL the θ_i associated with the two state vectors X and Y is calculated and subsequently added to the previous value of θ_t.

The second angle, termed ecliptic longitude, is accumulated similarly

$$\theta_\epsilon = \Sigma_i \, \theta_{\epsilon i}$$

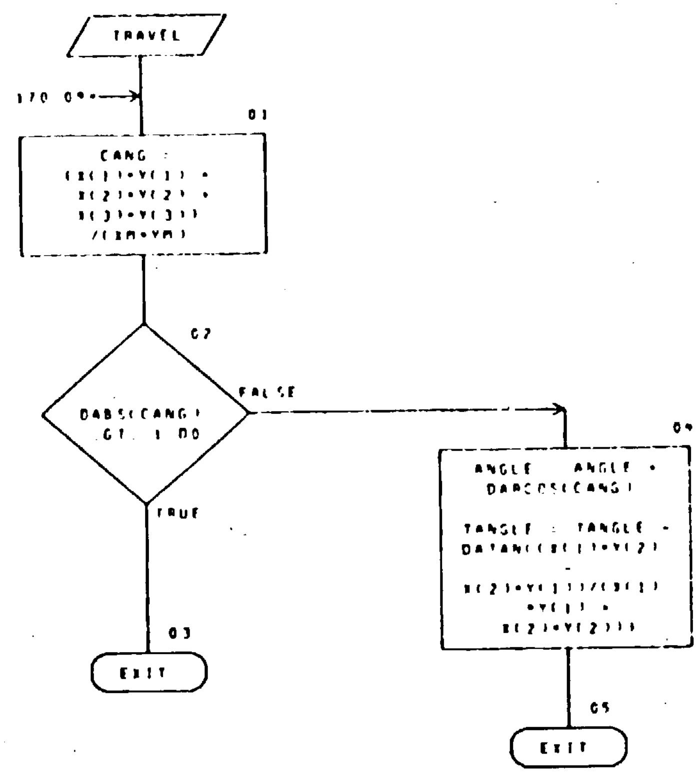
where

$$\theta_{\epsilon i} = -\tan^{-1}\left(\frac{x_1 y_2 - x_2 y_1}{x_1 y_1 + x_2 y_2}\right)$$

TRAVEL EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
X(3)	UX		Current position vector, X, AU.
Y(3)	UX		Previous position vector, Y, AU.
XM	UX		Magnitude of X, x, AU.
YM	UX		Magnitude of Y, y, AU.
ANGLE	SUX		Travel angle, θ_t, radians.
TANGLE	SU	REAL8	Ecliptic longitude, θ_ϵ, radians.

CHART TITLE - SUBROUTINE TRAVEL(X,Y,XM,YM,ANGLE)



ORIGINAL PAGE IS
OF POOR QUALITY

TRAVEL-3

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION X(3), Y(3)
COMMON /REAL8/ ROI(3,3),TANGLE,RS2(18,3)
```

<u>Name:</u> TWINKL

<u>Calling Argument:</u> None

<u>Referenced Sub-programs:</u> None

<u>Referenced Commons:</u> REAL8

<u>Entry Points:</u> None

<u>Referencing Sub-programs:</u> QSTART

<u>Discussion:</u> TWINKL is used to calculate a Cartesian pointing vector, $\bar{s}$, in the direction of Canopus. The right ascension, α, and declination, δ, of Canopus in the equatorial system are built-in to the sub-program.

$$\alpha = 95^{o}.799167$$

$$\delta = -52^{o}.67666$$

The obliquity of the ecliptic, ϵ, is used to calculate the vector in the ecliptic system.

$$\bar{s} = \begin{bmatrix} \cos\delta\cos\alpha \\ \cos\epsilon\cos\delta\sin\alpha + \sin\epsilon\sin\delta \\ -\sin\epsilon\cos\delta\sin\alpha + \cos\epsilon\sin\delta \end{bmatrix}$$

TWINKL EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
SRA	SU	REAL8	Right ascension of Canopus, α, deg.
SDC	SU	REAL8	Declination of Canopus, δ, deg.
SPV(3)	S	REAL8	Star unit vector, $\bar{s}$.
SE	U	REAL8	Sine of obliquity, $\sin\epsilon$.
CE	U	REAL8	Cosine of obliquity, $\cos\epsilon$.
DEG	U	REAL8	Degrees per radian .

CHART TITLE - SUBROUTINE TWINKL

ORIGINAL PAGE IS OF POOR QUALITY

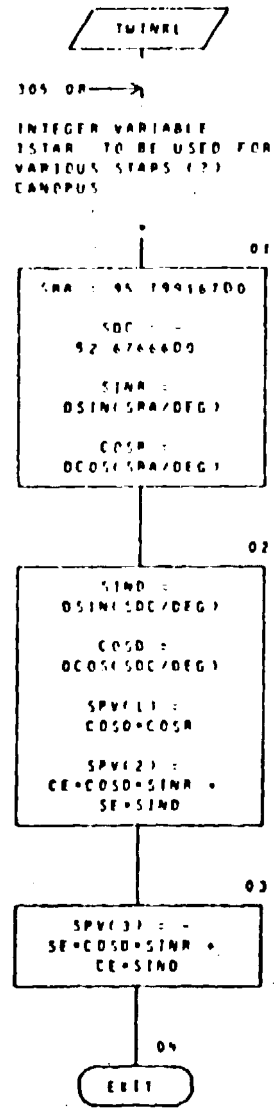


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8 (A-H,O-Z)
COMMON /REAL8/ RO1(150),SRA,SDC,SPV(3),RO2(7),SE,CE,RO3(196),
DEG,RO4(1687)
```

Name:	VMAG
Calling Argument:	V for VMAG; V, W for VDOT
Referenced Sub-programs:	None
Referenced Commons:	None
Entry Points:	VDOT
Referencing Sub-programs:	ALBEDO, CARKEP, CDERIV, QPRINT, RADAR, SPRINT, SWING for VMAG; ALBEDO, CARKEP, CDERIV, RADAR, SPRINT, SWING for VDOT

Discussion: This is basically a package of two FUNCTION subroutines, each of which performs an elementary operation in three dimensional space as follows:

Subroutine VMAG computes the magnitude of a vector;

$$VMAG = \sqrt{v_1^2 + v_2^2 + v_3^2}$$

Entry point VDOT computes the dot product of two vectors;

$$VDOT = v_1 w_1 + v_2 w_2 + v_3 w_3$$

VMAG EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
V(3)	UX		General input vector.
W(3)	UX		General input vector.
VDOT	SX		Output dot product of V and W.
VMAG	SX		Output magnitude of V.

CHART TITLE - FUNCTION VMAG(V)

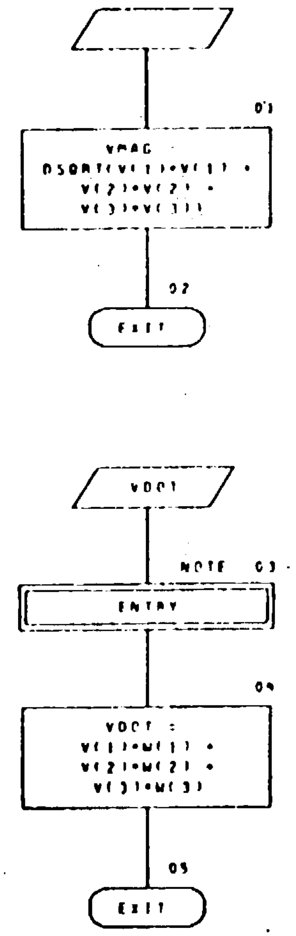


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION U(1),V(1),W(1)
```


VMAG-3

Name: VPRINT

Calling Argument: None

Referenced Sub-programs: None

Referenced Commons: ITERAT, REAL8

Entry Points: None

Referencing Sub-programs: TRAJ

Discussion: Historically, this routine was used briefly in 1970 in a related study involving optimum departure trajectories of the NERVA rocket. It has been retained in the program due to its potential use as a general print routine concerning optimum departure of rockets from circular Earth parking orbits. (The routine is called after the trajectory endpoint is reached). The related transversality conditions are found in subroutine GETQ. The related program option, invoked by the program input quantity ALTITU, could possibly be somewhat obsolete, i.e., in error, due to the many program alterations which have occurred over the years since 1970.

The following consists of a brief exposition summarizing the program option of simulating optimum Earth departure trajectories. It is assumed that the initial mass ratio and associated adjoint variable are initialized to unity; $\nu_o = 1$ and $\lambda_{\nu o} = 1$. The problem is restricted, without loss of generality, to two-dimensional motion in the xy-plane. The two-point boundary problem is 5x5, as follows:

Independent Variables Dependent Variables

$$\begin{bmatrix} \lambda_x \\ \lambda_y \\ \dot{\lambda}_x \\ \dot{\lambda}_y \\ t_f \end{bmatrix} \qquad \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \\ v_\infty \end{bmatrix}$$

The independent variables consist of the primer vector and its derivative and the transfer time (which is optimized), and the dependent variables consist of four transversality conditions and specified departure excess speed. The transversality conditions, all evaluated at the trajectory endpoint, are:

$$T_1 = \sigma = 0$$

$$T_2 = |R \times \dot{\Lambda}| = 0$$

$$T_3 = |\dot{R} \times \Lambda| = 0$$

$$T_4 = \dot{\Lambda} \cdot \dot{R} + \frac{\mu}{r\nu^2} = 0$$

where σ is the thrust switching function, R is the spacecraft position vector, Λ is the primer, μ is the Earth's gravity constant, $r = |R|$, and ν is the mass ratio. These correspond to optimum transfer time, optimum travel angle, optimum final eccentricity, and optimum propellant expenditure. Thrust switch points are optimized; however, in the class of solutions of primary interest, $\sigma_o = \sigma_f = 0$ and $\sigma > 0$ otherwise (between t_o and t_f). $\sigma_o = 0$ because the spacecraft is (conceptually) optimally coasting in the circular parking orbit at time of thrust initiation. This subroutine, which is a print subroutine, contains a test and related diagnostic printout to ensure that $\dot{\sigma}_f < 0$, which is a desired characteristic of condition T_1 for the class of solutions considered.

The circular parking orbit radius is given by

$$r_o = r_{Earth} + h ,$$

where r_{Earth} is the Earth's mean radius and h is the altitude (program input quantity). The initial speed is then

$$v_o = \sqrt{\mu/r_o} \quad .$$

The (final) semi-major axis, which is related to the specified v_∞, is given by

$$a = -(v^2 - 2/r)^{-1},$$

in which the gravity parameter is normalized; the actual departure excess speed achieved on the trial trajectory is then computed as

$$v_\infty = \pm v_o / \sqrt{|a|} \quad ,$$

where the minus sign indicates that escape was not achieved. The impulsive speed required to achieve the actual excess speed, starting from the parking orbit, is computed as

$$\Delta v_{IMP} = v_o \left(\sqrt{2 + 1/|a|} - 1 \right).$$

The characteristic speed of the departure trajectory is

$$v_c = -c\, v_o \ln \nu_f \quad ,$$

and the velocity loss due to gravity is

$$v_{loss} = v_c - \Delta v_{IMP} \quad .$$

The initial thrust-to-weight ratio is

$$T/W = g v_o^2 / r_o G \, ,$$

and the specific impulse is

$$I_{sp} = \frac{c\, v_o}{G} \quad .$$

In the above, G is the acceleration of gravity at the Earth's surface, g is the reference thrust acceleration in EMOS/tau, and c is the jet exhaust speed in EMOS.

Messages and printouts: The normal informative output is

OPTIMUM EARTH DEPARTURE

T/W	(T/W)	
ISP	(I_{sp})	SEC
V00	(v_{∞})	FPS
DVI	(Δv_{IMP})	FPS
VCHAR	(v_c)	FPS
VLOSS	(v_{loss})	FPS
TIME	$(t_f - t_o)$	HOURS

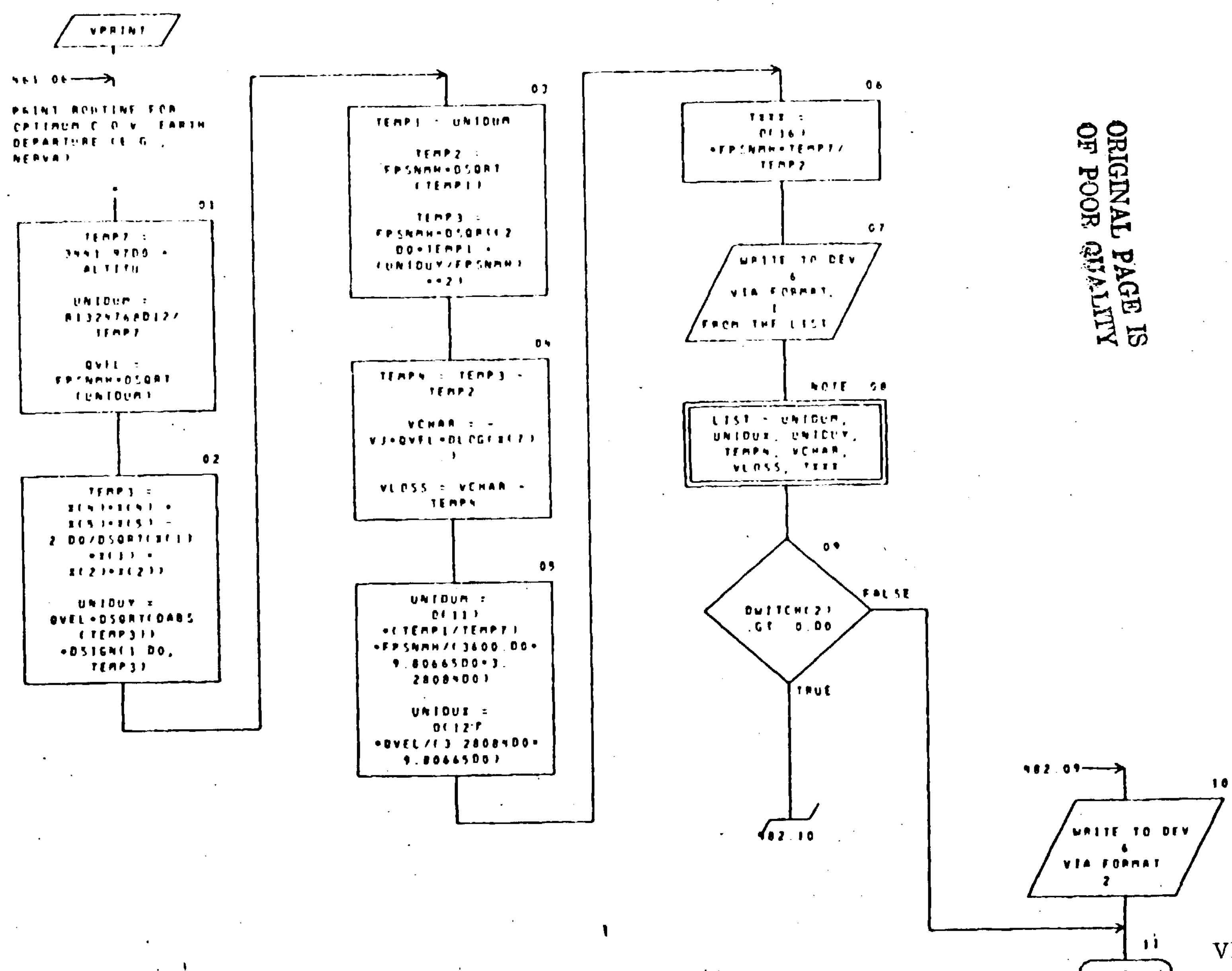
and, if $\dot{\sigma}_f > 0$, the diagnostic message is output:

ERROR BAD SWITCH FUNCTION ROOT

VPRINT EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
O(70)	U	ITERAT	Array of iterator independent variables.
X(50)	U	REAL8	Array of trajectory integrated functions.
VJ	U	REAL8	Jet exhaust speed, c, in EMOS.
ALTITU	U	REAL8	Circular orbit altitude, h, in nautical miles.
DWITCH(2)	U	REAL8	Thrust switching function derivative, $\dot{\sigma}$.
FPSNMH	U	REAL8	Conversion factor from nautical miles per hour to feet per second.

CHART TITLE - SUBROUTINE VPRINT



ORIGINAL PAGE IS OF POOR QUALITY

VPRINT-5

CHART TITLE - NON-PROCEDURAL STATEMENTS

```
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /REAL8/ RO1(12),VJ,RO2(193),SWITCH(2),RO3(105),EPSNMM,
     RO4(385),ALTITU,RO5(550),X(50),RO6(700)
      COMMON /ITERAT/ BO1(700),C(70),BO2(210)
1     FORMAT(1H1,23HOPTIMUM EARTH DEPARTURE//1H ,
     3HT/W,7XD16.8/1H ,3HISP,7XD16.8,4H SEC/1H ,
     3HVGO,7XD16.8,4H FPS/1H ,3HDVI,7XD16.8,4H FPS/1H ,
     5HVCHAR,5XD16.8,4H FPS/1H ,5HVLOSS,5XD16.8,4H FPS/1H ,
     4HTIME,6XD16.8,6H HOURS/1H )
2     FORMAT(1HO,11H***ERROR***,8X24HBAD SWITCH FUNCTION ROOT)
```

ORIGINAL PAGE IS
OF POOR QUALITY

<u>Name:</u> VSCAL

<u>Calling Argument:</u> A, V, W for VSCAL;
U, V, W for VCROSS, VADD, VSUB, and UNITD;
U, T, V for TFORM

<u>Referenced Sub-programs:</u> None

<u>Referenced Commons:</u> None

<u>Entry Points:</u> VCROSS, VADD, VSUB, UNITD, TFORM

<u>Referencing Sub-programs:</u> CDERIV, INCOND, SWING, TAP for VSCAL;
CARKEP, CDERIV, INCOND, QPRINT, SPRINT, SWING for VCROSS;
CDERIV for VADD;
ALBEDO, CDERIV, SWING for VSUB;
CDERIV, THANGD for UNITD;
SPRINT for TFORM

<u>Discussion:</u> This is basically a package of six subroutines, each of which performs an elementary operation in three dimensional space as follows:

Subroutine VSCAL magnifies vector V by the scalar a;

$$W = a\,V$$

Entry point VCROSS performs a vector cross-product;

$$W = U \times V$$

Entry point VADD performs a vector addition;

$$W = U + V$$

Entry point VSUB performs a vector subtraction;

$$W = U - V$$

Entry point UNITD computes the derivative vector W of the unit vector associated with any given vector U, where $V = \dot{U}$ is also given;

$$W = (\dot{\hat{U}}) = (U \cdot U)^{-1/2} [\dot{U} - (U \cdot U)^{-1} (U \cdot \dot{U})\, U]$$

Entry point TFORM multiplies vector U by matrix T to form vector V, in general. Specifically, if $\hat{i} = (i_x, i_y, i_z)$, $\hat{j} = (j_x, j_y, j_z)$, and $\hat{k} = (k_x, k_y, k_z)$ form an orthonormal set of vectors representing the axes of another coordinate system, then

$$V = T\,U$$

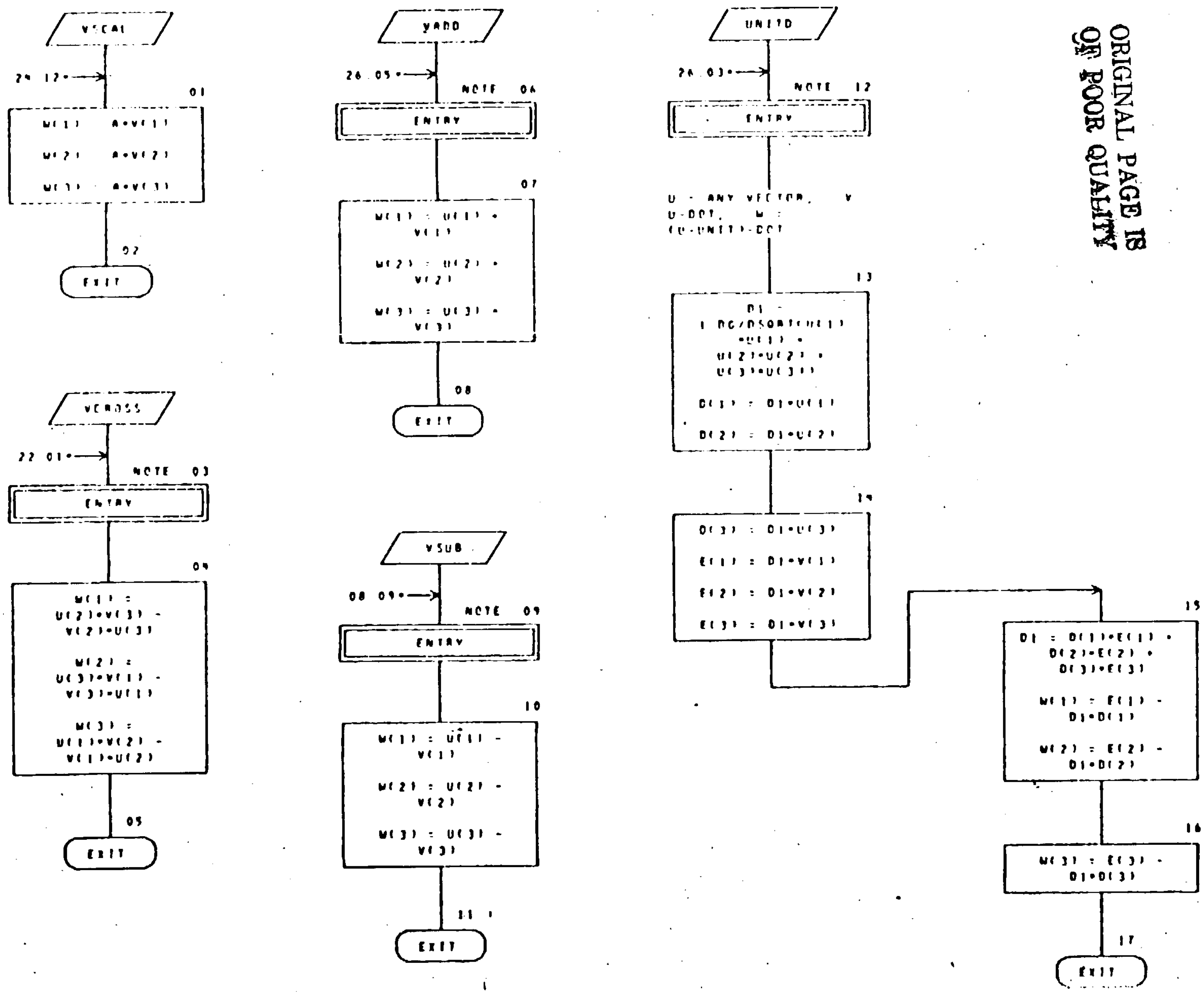
generates the representation of U in that other coordinate system, where

$$T = \begin{bmatrix} i_x & i_y & i_z \\ j_x & j_y & j_z \\ k_x & k_y & k_z \end{bmatrix}.$$

VSCAL EXTERNAL VARIABLES TABLE

Variable	Use	Common	Description
A	UX		Scalar quantity, a, which multiplies input vector to form output vector.
T (9)	UX		Transformation matrix, T, which is input to entry point TFORM.
U (3)	UX		General input vector.
V (3)	SUX		General input vector, except in entry point TFORM where it is the output vector.
W (3)	SX		General output vector.

CHART TITLE - SUBROUTINE VSCAL(A,V,W)



VSCAL-3

CHART TITLE - SUBROUTINE VSCAL(A,V,W)

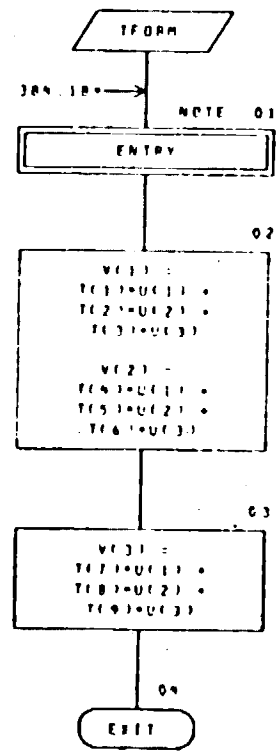


CHART TITLE - NON-PROCEDURAL STATEMENTS

```
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION U(3),V(3),W(3),D(3),F(3),T(9)
```


VSCAL-5