

General Disclaimer

One or more of the Following Statements may affect this Document

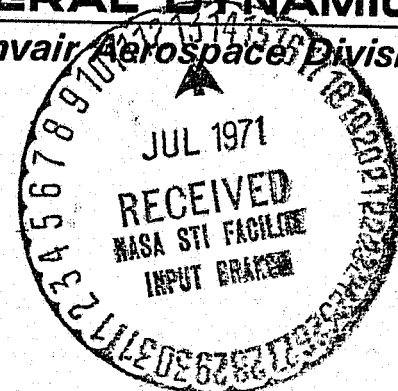
- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.
- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.
- This document is paginated as submitted by the original source.
- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

CR-119890
REPORT NO. GDC-DCF70-002
CONTRACT NAS 8-21036

TELEVISION BROADCAST SATELLITE STUDY

VOLUME V + SYNTHESIS/OPTIMIZATION
COMPUTER PROGRAM

GENERAL DYNAMICS
Convair Aerospace Division



FACILITY FORM 602

N71-31410	(ACCESSION NUMBER)
258	(PAGES)
CR-119890	(NASA CR OR TMX OR AD NUMBER)
G3	(THRU)
31	(CODE)
	(CATEGORY)

REPORT NO. GDC-DCF70-002

TELEVISION BROADCAST SATELLITE STUDY

TVBS SYSTEM

**SYNTHESIS/OPTIMIZATION
COMPUTER PROGRAM**

VOLUME V

CONTRACT NAS 8-21036

31 December 1970

Prepared for
E. C. Hamilton, COR
National Aeronautics and Space Administration
George C. Marshall Space Flight Center
Huntsville, Alabama

Prepared by
CONVAIR AEROSPACE DIVISION OF GENERAL DYNAMICS
San Diego, California

PRECEDING PAGE BLANK NOT FILMED

FOREWORD

This report was prepared by the San Diego operation of the Convair Aerospace Division of General Dynamics under Contract NAS8-21036, Television Broadcast Satellite Study, for the George C. Marshall Space Flight Center of the National Aeronautics and Space Administration. The work was administered under the technical direction of the Astrionics Laboratory of the George C. Marshall Space Flight Center.

TABLE OF CONTENTS

	<u>Page No.</u>
List of Figures	vi
List of Tables	vii
Subroutine Listings	viii
1.0 INTRODUCTION	1-1
2.0 TECHNICAL DISCUSSION	2-1
2.1 Program Description	2-1
2.2 Mathematical Design Optimization Technique	2-4
2.3 Parametric Subsystem Models	2-7
2.4 Cost Model	2-8
2.5 Area of Coverage Model	2-19
2.6 Signal Attenuation/Noise Models	2-22
2.7 Redundancy Model	2-28
3.0 INPUT VARIABLES	3-1
3.1 Namelist NAME1	3-1
3.2 Namelist NAMEP1	3-23
3.3 Namelist NAME2	3-26
4.0 SAMPLE COMPUTER RUN	4-1
5.0 SUBROUTINE DESCRIPTION	5-1
5.1 TVBS DRIVER	5-1
5.2 Subroutine INIT	5-15
5.3 Subroutine CONV	5-21
5.4 Subroutine SYSEQ	5-28
5.5 Subroutine PLACE	5-53
5.6 Subroutine PRTIAL	5-58
5.7 Subroutine BCHECK	5-67
5.8 Subroutine ERROR	5-70
5.9 Subroutine OUT and OUTPTX	5-72
Appendix A General Purpose System Subroutines	A-1
Appendix B Derivation of Steepest Descent Iterative Algorithm	B-1
Appendix C Derivation of Partial Derivatives	C-1
Appendix D Parametric Relationships	D-1
Appendix E Spacecraft Constraint Relationships	E-1

LIST OF FIGURES

Figure	Page No.
2.1 TVBS Deck Setup	2-2
2.2 TVBS General Flow Diagram	2-3
2.3 TVBS Cost Model	2-8
2.4 Beamwidth Geometry - minor axis	2-21
2.5 Beamwidth Geometry - major axis	2-21
4.1 Sample Input	4-2
4.2 Sample Output	4-11
5.1 Flow Diagram - TVBS Driver	5-7
5.2 Flow Diagram - TVBS Do-loop System	5-8
5.3 Flow Diagram - Subroutine INIT	5-16
5.4 Flow Diagram - Subroutine CONV	5-23
5.5 Flow Diagram - Subroutine SYSEQ	5-35
5.6 Flow Diagram - Subroutine PLACE	5-54
5.7 Flow Diagram - Subroutine PRTIAL	5-59
5.8 Flow Diagram - Subroutine BCHECK	5-68
D-1 Power Model	D-67
E-1 Concepts A and B Geometrical Parameters	E-3
E-2 Concept C Configuration Geometry	E-6

LIST OF TABLES

<u>Table No.</u>	<u>Page No.</u>
2-1 Table of Cost Elements	2-9
2-2 Attenuation Due to Rain	2-23
2-3 Attenuation Due to Clouds	2-24
2-4 Attenuation Due to Atmosphere	2-24
2-5 TBVS Noise and Propagation Models	2-25
2-6 Loss Model	2-27
5-1 SYSEQ Parameter List	5-29
D-1 Thruput Cost Elements	D-29
D-2 TVBS Cost Model Launch Venicle Costs	D-30
E-1 Concept Summary	E-2
E-2 Geometry, Weight, and Volume Constraints - Concept A	E-4
E-3 Geometry, Weight, and Volume Constraints - Concept B	E-5
E-4 Geometry, Weight, and Volume Constraints - Concept C	E-7
E-5 Weight, Volume, and Geometry Constraints - Concept D	E-8
E-6 Weight, Volume, and Geometry Constraints - Concept E	E-9
E-7 Weight, Volume, and Geometry Constraints - Concept F	E-10
E-8 Concept G Constraints	E-11
E-9 Nomenclature	E-12

SUBROUTINE LISTINGS

<u>Subroutine</u>	<u>Page No.</u>
5.1 TVBS DRIVER	5-9
5.2 Subroutine INIT	5-17
5.3 Subroutine CONV	5-25
5.4 Subroutine SYSEQ	5-37
5.5 Subroutine PLACE	5-55
5.6 Subroutine PRTIAL	5-60
5.7 Subroutine BCHECK	5-69
5.8 Subroutine ERROR	5-71
5.9 Subroutine OUT and OUTPTX	5-73, 74

1

1.0 INTRODUCTION

The TVBS system synthesis/optimization program developed by Convair under contract No. NAS8-21036 provides the user with a flexible tool for the analysis of TVBS system requirements. While obtaining minimum cost design points, the program enables the user to perform studies over a wide range of user requirements and spacecraft configurations. The program utilizes a total systems approach in which the ground uplink, spacecraft, launch vehicle, and ground downlink systems are simultaneously synthesized. A steepest-descent algorithm is employed to determine the minimum total system cost design for each configuration considered. In the process of converging to the solution, the pertinent subsystem trade-offs are resolved.

The program provides the system designer with the capability to consider a wide choice of system configurations. By appropriate specification of the pertinent input variables, the program can be made to cycle through the various types of subsystem elements and user requirements to be considered. The minimum cost solutions obtained for each configuration can be compared to determine that configuration which has the minimum cost.

The program can also be used as a tool in performing sensitivity analyses. For example, it can be employed to determine the change in the total system cost resulting from a perturbation of the design parameters, the configuration sizing constraints, and the subsystem parametric relationships, as well as the user requirements (S/N, frequency, etc.).

This report documents the TVBS computer program. Section 2 provides a general technical discussion. Section 3 provides definitions of the variables in the input lists. Section 5 provides a description of the driver and each subroutine comprising the program. A verbal description of a computer listing, and a flow diagram where applicable is included for each. A sample of the input used by the program is included in Section 4 together with a sample output listing. Appendix A is included to present the general purpose system subroutines required by the program. Appendix B provides a derivation of the equations used in the convergence algorithm.

Appendix C presents the method implemented for the solution of the partial derivatives of the parametric relations. Appendix D provides the subsystem parametric relations which were used to obtain the analytic model coefficient values listed in Section 4.1. Appendix E provides the payload sizing constraints imposed by the satellite configurations which were considered.

2

2.0 TECHNICAL DISCUSSION

2.1 PROGRAM DESCRIPTION

The TVBS computer program is comprised of a driver and nine sub-routines (see Figure 2.1). The components are as listed:

TVBS DRIVER	- initializes and controls execution
SUBROUTINE INIT	- initializes and sets boundary constraints
SUBROUTINE CONV	- controls execution of the convergence technique
SUBROUTINE SYSEQ	- computes value of dependent parameters
SUBROUTINE PLACE	- sets coefficients for transmitter curves
SUBROUTINE PRTIAL	- computes required partial derivatives
SUBROUTINE BCHECK	- checks system parameters for boundary violations
SUBROUTINE ERROR	- controls output of diagnostic messages under certain conditions
SUBROUTINE OUT	- provides intermediate output of iteration results
SUBROUTINE OUTPIX	- provides final summary output

These do not include the required general purpose system routines which are listed in Appendix A. A detailed description of each of the above sub-programs is given in Section 3.

The general logic flow of the computer program is given in Figure 2.2. The program was specifically designed for use on Convair's CDC 6400 computer system using FORTRAN IV, CDC Version 2.1 programming language.

On the basis of 100 cases run, the average required central processor (CP) time per run was 4.6 seconds. Core requirement for execution is approximately 47K₈ including routines required by the operating system.

It is essential that before executing the program, core be initialized to zero.

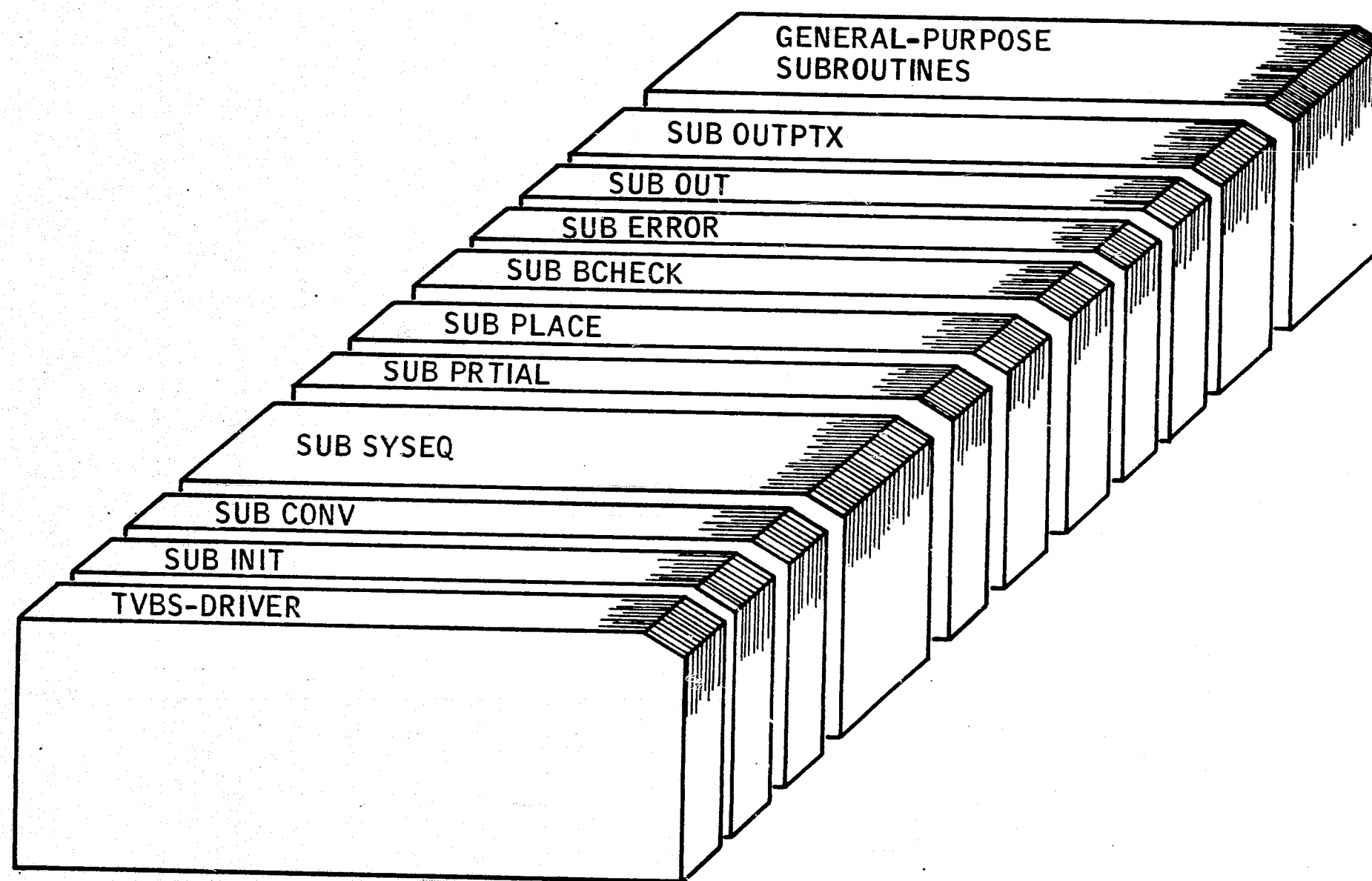


Figure 2.1 TVBS Deck Setup

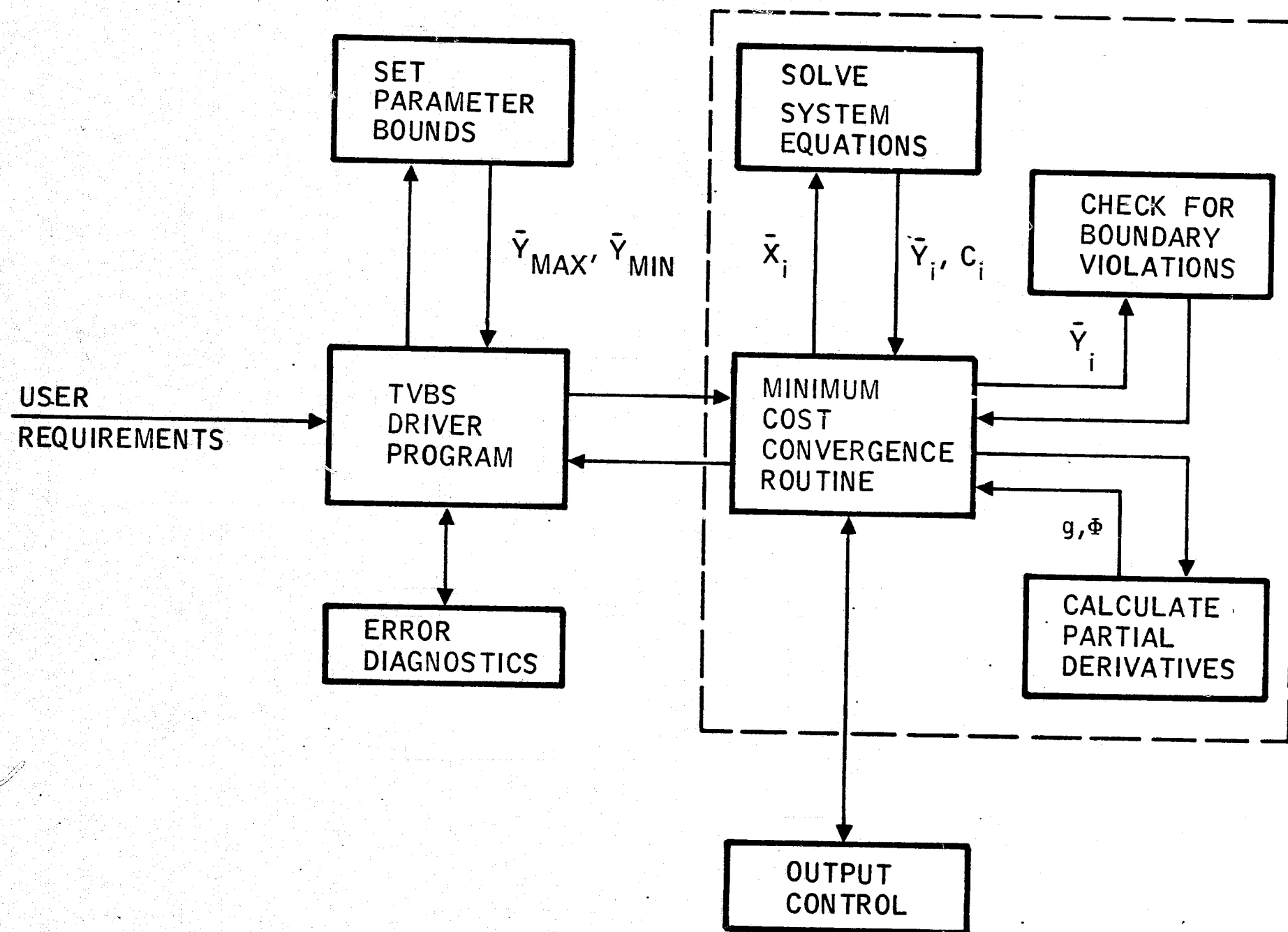


Figure 2.2 TVBS General Flow Diagram

2.2 MATHEMATICAL DESIGN OPTIMIZATION TECHNIQUE

The Convair TVBS system design optimization program uses a steepest descent iterative process, which is a modified gradient technique to determine the minimum cost design point for a given system configuration. At each iteration the nominal design point is perturbed in such a way as to cause the maximum rate of decrease in total system cost to be consistent with the design constraints.

If Δy_j is the amount by which the j^{th} boundary is violated and g represents the gradient of total system cost, then the perturbation in $\bar{X}$, the vector of independent design parameters, is given by

$$\Delta \bar{X} = -K_c \left[I - \Phi (\Phi^T \Phi)^{-1} \Phi^T \right] g - K_y \Phi (\Phi^T \Phi)^{-1} \Delta y_j$$

where Φ is the gradient of the constraint relationship. The derivation of this equation is given in Appendix B.

If K_y is set equal to unity, the second term represents, to first order terms, the amount by which $\bar{X}$ must be changed to reduce Δy_j to zero. If K_y is less than one, the boundary violation is partially reduced. The first term represents the component of the cost gradient parallel to the constraint boundary and is orthogonal to the second term. For small ΔX_i (so that the linear approximation is valid), the first term reduces the cost and contributes nothing directly to the boundary violation. If no boundaries are violated, then Δy_j and Φ are zero, so that the equation reduces to

$$\Delta X = -K_c g$$

which represents a perturbation in the direction of the negative of the cost gradient.

Since K_c is arbitrary some control over the step-size must be introduced. The principle used in determining the magnitude of the step is that if successive steps have been in the same direction it is desirable

to increase the step-size in order to speed up the convergence process. On the other hand, if successive steps have been in opposite directions, it is desirable to decrease the step-size since the optimum design point has been overshoot. Furthermore, it appears that larger step-sizes are most desirable in the early iterations to hasten arrival to the vicinity of the optimum design point. Once in the vicinity of the optimum, smaller steps would prove more practical. With this in mind, the following step-size control algorithm has been implemented. Let α be the angle between two successive step vectors, ΔX_{i-1} and ΔX_{i-2} , and let S_i be the magnitude of the i^{th} step, $|\Delta X_i|$.

$$\begin{aligned} S_i &= (.75 + .25 \cos \alpha) S_{i-1} && \text{if } \cos \alpha < 0 \\ S_i &= (.5 + 1.5 \cos \alpha) S_{i-1} && \text{if } \cos \alpha \geq 0 \text{ and } i \leq 10 \\ S_i &= (.55 + .45 \cos \alpha) S_{i-1} && \text{if } \cos \alpha \geq 0 \text{ and } i > 10 \\ S_i &= 20 && \text{if } S_i > 20 \end{aligned}$$

where i = iteration index.

With the step-size, S_i , thus specified, the appropriate value of K_c can be determined from

$$K_c = \sqrt{\frac{S_i^2 - K_y^2 \cdot \Delta y_j^T (\Phi^T \Phi)^{-1} \Delta y_j}{H^T H}}, \quad H = \left[I - \Phi(\Phi^T \Phi)^{-1} \Phi^T \right] g$$

in the event of a boundary violation. If the solution is unconstrained K_c is determined from

$$K_c = \frac{S_i}{|g|}$$

These are the equations used by the TVBS convergence routine. It is important to keep S_i from growing to a point at which a large number of iterations would be required to bring it down. Therefore the upper bound is placed on S_i .

The rate of convergence to the minimum cost design point is dependent upon the relative sensitivities of the cost to each of the design parameters. Since the algorithm operates on the most sensitive parameters, convergence to the optimum value of the less sensitive parameters can be slow. For this reason it is often desirable to rescale the units of the design parameters to equalize these sensitivities.

In this application, the vector of independent variables $\bar{X}$ is given as

$$\bar{X} = (x_1, x_2, x_3) = (D_1, D_4, T_R)$$

where

D_1 = diameter of the ground uplink antenna,

D_4 = diameter of the ground downlink antenna, and

T_R = temperature of the receiver on the ground,

$\bar{Y}$, the vector of dependent parameters, is given as

$$\bar{Y} = (y_1, \dots, y_6) \text{ where}$$

$y_1 = P_{TR}$ = spacecraft transmitter power output/channel

$y_2 = W_{TOT}$ = total weight of spacecraft payload

$y_3 = V_{EM}$ = total volume of equipment module

$y_4 = V_{TOT}$ = total volume of spacecraft payload

$y_5 = P_G$ = ground uplink power output/channel

$y_6 = V_{PP}$ = total volume of power supply

2.3 PARAMETRIC SUBSYSTEM MODELS

The TVBS subsystem models include the pertinent elements required to obtain a complete synthesis of the total system. The cost, weight, and volume curves of the various subsystems modeled in the program are given in Appendix D. The coefficients may be found in Section 4 and the general forms of the equations representing the curves are found in Section 3.

In general, many of the curves were modeled using a log-quadratic approximation. The equations are of the form

$$Z = 10^{\frac{y}{10}}$$

$$\text{where } y = a + bx + cx^2$$

$$x = 10 \log X, \text{ and}$$

X represents the value of the parameter determining Z.

The curves used during the study are applicable over a wide range of parametric values. Where it was impossible to obtain an accurate fit with a single equation, the curve was fit in more than one piece. The breakpoint, in most cases, is variable and included in the input list. Curves not fit with equations of this form are either linear or exponential and are indicated in Section 3.

The system synthesis incorporates a loss model for both the uplink and downlink, a noise model, and scale factors to account for redundancy. The loss model and noise model are discussed in Section 2.6 while the redundancy scale factors are found in Section 2.7.

The cost elements employed in the subsystem cost models are located in tabular form in Section 2.4.

The list of dependent system parameters includes transmitter output power per channel (ground uplink and spacecraft), total spacecraft weight and volume, volume of the spacecraft equipment module, and, when constrained, the volume of the power supply. The list of independent design parameters includes the diameter of the ground uplink and downlink antenna and the noise temperature of the ground receiver.

2.4 COST MODEL

The TVBS cost model, summarized in Figure 2.3, has a unique feature of a unit cost submodel which uses mission parameters to calculate unit hardware costs in

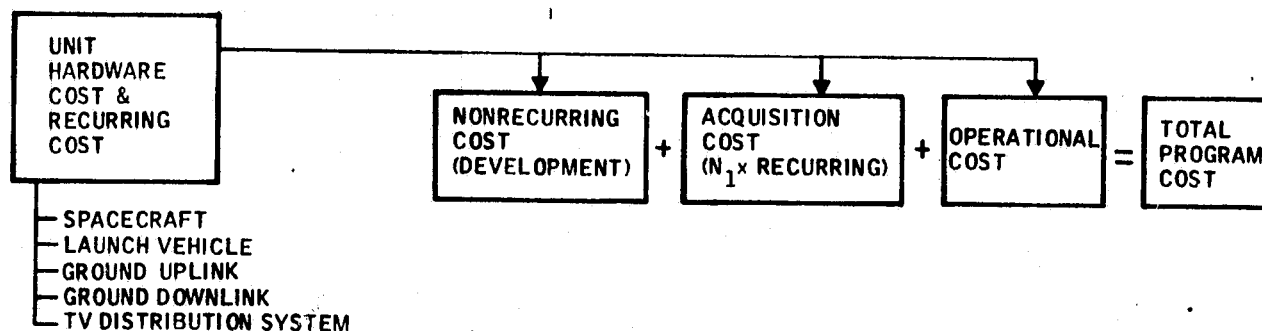


Figure 2.3 TVBS Cost Model

the various subsystems. The model then computes the non-recurring costs (development), acquisition costs, and operational costs to form the total program cost. The non-recurring costs include all design, development, and analysis costs necessary to bring the program to the point of first operational configuration. The acquisition submodel accumulates the costs associated with the fabrication and launching necessary to establish the initial operational system. Operational costs include all costs associated with the operational phase of the mission including replacement satellites.

Table 2.1 is a tabulation of the cost model as implemented in the computer program. The cost elements or cost estimating relationships which have been defined are presented in Appendix D of this report. As discussed in Section 2.3 these parametric relationships, in the majority of the cases, are modeled analytically with log-quadratic approximations.

TABLE 2.1

SUBSYSTEM COST MODEL

Table of Cost Elements

I GROUND UPLINK SYSTEMVALUE

A. Unit Recurring Cost

1. UREC1(1) - Facilities	Curve
2. UREC1(2) - Programming, Recording and Control Equipment	Curve
3. UREC1(3) - Transmitter Subsystem	Curve
4. UREC1(4) - Antenna Subsystem	Curve
5. UREC1(5) - Satellite Operations and Control Equipment	Thruput
6. UREC1(6) - Standby Power Subsystem	Curve
7. UREC1(7) - Test Equipment	Thruput
8. TUREC (1)- Total Ground Uplink Unit Recurring Cost	Σ (1) thru (7)

B. Non-recurring Cost (Research and Development)

1. NREC1(1) - Facilities Architecture and Engineering	10% x UREC1(1)
2. NREC1(2) - Programming, Recording and Control Equipment	0% x UREC1(2)
3. NREC1(3) - Transmitter Subsystem	0% x UREC1(3)
4. NREC1(4) - Antenna Subsystem	0% x UREC1(4)
5. NREC1(5) - Satellite Operations and Control Equipment	100% x UREC1(5)
6. NREC1(6) - System Design, Integration, and Management	Σ (3) thru (6)
7. TNREC(1) - Total Ground Uplink Non-recurring Cost	Σ (1) thru (6)

TABLE 2.1(cont)

C. Acquisition Cost

1. ACQC1(1) - Hardware Unit Recurring
2. ACQC1(2) - Installation, Activation, and Checkout
3. ACQC1(3) - Ground Uplink Acquisition Cost
4. TACQC(1) - Total Ground Uplink Acquisition Cost

$$\begin{aligned} & \text{TUREC}(1) \\ & 20\% \times \Sigma \text{UREC1}(2) \text{ thru } \text{UREC1}(6) \\ & (1) + (2) \\ & N_u \times \text{ACQC1}(3) \end{aligned}$$

D. Operations Cost

1. OPC1(1) - Facility Maintenance
2. OPC1(2) - Programming, Recording, and Control Maintenance
3. OPC1(3) - Programming, Recording, and Control Personnel
4. OPC1(4) - Transmitter Subsystem Maintenance
5. OPC1(5) - Antenna Subsystem Maintenance
6. OPC1(6) - Satellite Operations and Control Maintenance
7. OPC1(7) - Satellite Operations and Control Personnel
8. OPC1(8) - Standby Power Maintenance
9. OPC1(9) - Systems Operation Management
10. OPC1(10) - Ground Uplink System Annual Operating Cost
11. TOPC(1) - Total Ground Uplink Operating Cost

$$\begin{aligned} & 2\% \times \text{UREC1}(1) \\ & 15\% \times \text{UREC1}(2) \\ & \text{Curve} \\ & 15\% \times \text{UREC1}(3) \\ & 8\% \times \text{UREC1}(4) \\ & 15\% \times \text{UREC1}(5) \\ & \text{Thruput} \\ & 5\% \times \text{UREC1}(6) \\ & 50\% \times [(3) + (7)] \\ & \Sigma (1) \text{ thru } (9) \\ & N_u \times N_L \times \text{OPC1}(10) \end{aligned}$$

TABLE 2.1 (cont)

II SPACECRAFT SYSTEM**A. Unit Recurring Cost**

	<u>VALUE</u>
1. UREC2(1) - Structure	Curve
2. UREC2(2) - Electrical Power Subsystem	Curve
2a UREC2(3) - Batteries	Curve
3. UREC2(4) - Attitude Control Subsystem	Curve
4. UREC2(5) - Stationkeeping	Curve
5. UREC2(6) - Thermal Control	Curve
6. UREC2(7) - Life Support Subsystem	Thruput
7. UREC2(8) - Instrumentation and Communication	Thruput
8. UREC2(9) - Uplink Receiver Subsystem	Curve
8a UREC2(10) - Uplink Antenna	Curve
9. UREC2(11) - Downlink Transmitter Subsystem	Curve
9a UREC2(12) - Multiplexer	Curve
10. UREC2(13) - Downlink Antenna Subsystem	Curve
11. UREC2(14) - Spacecraft Unit Hardware	Σ (1) thru (13)
12. UREC2(15) - Integration, Assembly, and Checkout	$7.5\% \times \text{UREC2}(14)$
13. TUREC(2) - Total Spacecraft Unit Hardware	$(14) + (15)$

B. Non-recurring Cost (Research and Development)

1. NREC2(1) - Structure	$350\% \times \text{UREC2}(1)$ (conventional);
2. NREC2(2) - Electrical Power Subsystem	Curve (thruput otherwise)
2a NREC2(3) - Batteries	Curve

TABLE 2.1 (cont)

3. NREC2(4) - Attitude Control Subsystem	Curve
4. NREC2(5) - Stationkeeping	Curve
5. NREC2(6) - Thermal Control	Curve
6. NREC2(7) - Life Support Subsystem	Thruput
7. NREC2(8) - Instrumentation and Communication	Thruput
8. NREC2(9) - Uplink Receiver	Curve
8a NREC2(10) - Uplink Antenna	Curve
9. NREC2(11) - Downlink Transmitter Subsystem	Curve
9a NREC2(12) - Multiplexer	Curve
10. NREC2(13) - Downlink Antenna Subsystem	Curve
11. NREC2(14) - Spacecraft Subsystems Total	$\Sigma (1) \text{ thru } (13)$
12. NREC2(15) - Prototype	$N_{pt} \times [\text{UREC2}(14) - .85 \text{ prime power}]$
13. NREC2(16) - Subsystem Design, Integration, and Management	$50\% \times (14)$
14. NREC2(17) - Center Support and Development	$5\% \times (14 + (16))$
15. NREC2(18) - GSE Design, Development and Testing	$50\% \text{ TUREC}(2)$
16. NREC2(19) - GSE Hardware	Thruput
17. NREC2(20) - Facilities and S/C Test Equipment	Thruput
18. TNREC(2) - Total Spacecraft Non-recurring Cost	$\Sigma (1) \text{ thru } (20)$
C. Acquisition Cost	
1. ACQC2(1) - Hardware Unit Recurring Cost	TUREC(2)
2. ACQC2(2) - Center Support/Unit	$5\% \times (1)$
3. ACQC2(3) - Launch Operations, Support and Spare	$10\% \times (1)$
4. ACQC2(4) - Unit Spacecraft Acquisition Cost	$\Sigma (1) \text{ thru } (3)$
5. TACQC(2) - Total Spacecraft Acquisition Cost	$N_S \times (4)$

TABLE 2.1 (cont)

D. Operations Cost

- | | | |
|------------|---------------------------------------|----------|
| 1. OPC2(1) | - Satellite Hardware Replacement Cost | TACQC(2) |
| 2. TOPC(2) | - Total Hardware Replacement Cost | Nr x (1) |

III LAUNCH VEHICLE SYSTEM

A. Unit Recurring Cost

- | | | |
|-------------|--------------------------------|-----------|
| 1. UREC3(1) | - Launch Vehicle | Thruput |
| 2. UREC3(2) | - Adapter and Fairings | Thruput |
| 3. TUREC(3) | - Launch Vehicle Hardware Cost | (1) + (2) |

B. Non-recurring Cost (Research and Development)

- | | | |
|-------------|--|-----------------------|
| 1. NREC3(1) | - Launch Vehicle, Adapter and Fairings | Thruput |
| 2. NREC3(2) | - Launch Complex Modifications | Thruput |
| 3. NREC3(3) | - Peculiar GSE Hardware | Thruput |
| 4. NREC3(4) | - Peculiar GSE Design and Development | $O\%$ x NREC3(3) |
| 5. NREC3(5) | - Center Support and Development | Thruput |
| 6. TNREC(3) | - Launch Vehicle Non-recurring Cost | Σ (1) thru (5) |

C. Acquisition Cost

- | | | |
|-------------|--|-----------------------|
| 1. ACQC3(1) | - Launch Vehicle Hardware Cost | TUREC(3) |
| 2. ACQC3(2) | - Center Support/Unit | Thruput |
| 3. ACQC3(3) | - Launch Operations, Support and Spares | Thruput |
| 4. ACQC4(4) | --Launch Vehicle Unit Acquisition Cost | Σ (1) thru (3) |
| 5. TACQC(3) | - Total System Launch Vehicle Acquisition Cost | N_{LV} x ACQC3(4) |

TABLE 2.1 (cont)

D. Operations Cost

1. OPC3(1) - Launch Vehicle Unit Cost
2. TOPC(3) - Total Launch Vehicle Costs for Replacements

TACQC(3)

$N_{LVR} \times OPC3(1)$

IV GROUND DOWNLINK SYSTEM Direct-to-Home

A. Unit Recurring Cost

1. UREC41(1) - Downlink Receiving Antenna
2. UREC41(2) - Downlink Receiver/Converter
3. TUREC(4) - Total Direct-to-Home Unit Cost

Curve

Curve

(1) + (2)

B. Non-recurring Cost (Research and Development)

1. NREC41(1) - Downlink Receiving Antenna
2. NREC41(2) - Downlink Receiver/Converter
3. NREC41(3) - Testing and Management
4. TNREC(4) - Total Direct-to-Home Non-recurring Cost

Included in UREC41(1)

Included in UREC41(2)

(1) + (2)

Σ (1) thru (3)

C. Acquisition Cost

1. ACQC41(1) - Direct-to-Home Unit Hardware
2. ACQC41(2) - Unit Installation
3. TACQC(4) - Total Direct-to-Home Acquisition Cost

TUREC(4)

Curve

$N_{dh} \left[(1) + (2) \right]$

TABLE 2.1 (cont)

D. Operations Cost

1. OPC41(1) - Direct-to-Home Annual Maintenance Cost
2. TOPC(4) - Total Direct-to-Home Operations Cost

$$10\% \times ACQC41(1)$$

$$N_{dh} \times N_L \times OPC41(1)$$

GROUND DOWNLINK SYSTEM - Retransmission System

A. Unit Recurring Cost

1. UREC42(1) - Retransmission Facility
2. UREC42(2) - Retransmission Antenna System
3. UREC42(3) - Retransmission Transmitter
4. UREC42(4) - Downlink Receiving Antenna
5. UREC42(5) - Downlink Receiver/Converter
6. UREC42(6) - Standby Power
7. UREC42(7) - Test Equipment
8. TUREC(4) - Total Retransmission Unit Hardware

Thruput



$$\Sigma (1) \text{ thru } (7)$$

B. Non-recurring Cost

1. NREC42(1) - Retransmission Facility Architecture and Engineering
2. NREC42(2) - Retransmission Antenna Subsystem
3. NREC42(3) - Retransmission Transmitter Subsystem
4. NREC42(4) - Downlink Receiving Antenna
5. NREC42(5) - Downlink Receiver/Converter
6. NREC42(6) - Retransmission Subsystem Development
7. NREC42(7) - System Design, Integration and Management
8. TNREC(4) - Total Retransmission Non-recurring Cost

Thruput



$$\Sigma (1) \text{ thru } (5)$$

$$200\% \times TUREC(4)$$

$$(6) + (7)$$

TABLE 2.1 (cont)

C. Acquisition Cost

1. ACQC42(1) - Retransmission Unit Hardware
2. ACQC42(2) - Installation, Activation, and Checkout
3. TACQC(4) - Total Retransmission System Acquisition

$$\begin{aligned} &T_{UREC}(4) \\ &20\% \times ACQC42(1) \\ &N_{rt} [(1) + (2)] \end{aligned}$$

D. Operations Cost

1. OPC42(1) - Retransmission Facility Maintenance
2. OPC42(2) - Receiving Subsystem Maintenance
3. OPC42(3) - Retransmission Antenna
4. OPC42(4) - Retransmission Transmitter Maintenance
5. OPC42(5) - Standby Power Maintenance
6. OPC42(6) - Retransmission Operations Personnel
7. OPC42(7) - Retransmission System Management
8. OPC42(8) - Retransmission Annual Operating Cost
9. TOPC(4) - Total Retransmission Operating Cost

$$\begin{aligned} &2\% \times UREC42(1) \\ &15\% \times [UREC42(4) + UREC42(5)] \\ &8\% \times UREC42(2) \\ &15\% \times UREC42(3) \\ &5\% \times UREC42(6) \\ &\text{Thruput} \\ &50\% \times (6) \\ &\Sigma (1) \text{ thru } (7) \\ &N_L \times N_{rt} \times (8) \end{aligned}$$

TABLE 2. 1 (cont)

GROUND DOWNLINK SYSTEM - CATV System

A. Unit Recurring Cost

- | | | |
|--------------|---|---------------------------|
| 1. UREC43(1) | - CATV Distribution Downlink Antenna | 89% - 95% learning, curve |
| 2. UREC43(2) | - CATV Downlink Receiver/Converter | 85% learning, curve |
| 3. UREC43(3) | - Receiving System Support Facilities | Thruput |
| 4. UREC43(4) | - Receiving System Equipment (Excl. antenna & rec.) | Thruput |
| 5. UREC43(5) | - Cable | Curve |
| 6. UREC43(6) | - Transmission Line | Curve |
| 7. UREC43(7) | - Hardware Cost (excl. antenna, receiver & trans. line) | Σ (3) thru (5) |
| 8. TUREC(4) | - Total CATV Unit Hardware Cost | Σ (1) thru (7) |

B. Non-recurring Cost

1. NREC43(1),...NREC43(5) - not used, assumed to be included in unit receiving cost

C. Acquisition Cost

- | | | |
|--------------|--|-----------------------------|
| 1. ACQC43(1) | - CATV Unit Hardware | TUREC(4) |
| 2. ACQC43(2) | - Installation, Activation, and Checkout | Curve |
| 3. TACQC(4) | - Total CATV System Acquisition | $N_{ca} \times [(1) + (2)]$ |

D. Operations Cost

- | | | |
|-------------|--|-----------------|
| 1. OPC43(1) | - Antenna Maintenance | 15% x UREC43(1) |
| 2. OPC43(2) | - Receiver Maintenance | 10% x UREC43(2) |
| 3. OPC43(3) | - Support Facilities Maintenance | 2% x UREC43(3) |
| 4. OPC43(4) | - Receiving System Equipment Maintenance | 15% x UREC43(4) |
| 5. OPC43(5) | - Cable Repairs, Parts, Pole Rental, Lab & Personnel | Curve |

TABLE 2.1(cont)

6. OPC43(6)	- Support Equipment Annual Operating Cost	Σ (3) thru (5)
7. OPC43(7)	- CATV Annual Operating Cost	Σ (1), (2), (6)
8. TOPC(4)	- Total CATV System Operating Cost	$N_L \times N_{ca} \times (7)$

MANNED PROVISIONS/STRUCTURE

ACQMP, ENGMP	- acquisition/engineering cost for manned provisions
URCBSM, ENGBSM	- unit recurring/engineering cost for BSM
URCMDA, ENGMDA	- unit recurring/engineering cost for MDA

NOTES:

N_u	= Number of Uplinks in System	N_{LVR}	= Number of Launch Vehicles for Replacement
N_L	= System Lifetime in years	N_{dh}	= Number of Direct-to-Home Receivers
N_{pt}	= Number of Prototypes fabricated	N_{rt}	= Number of Retransmission Systems
N_s	= Number of Satellites in System	N_{ca}	= Number of CATV Systems
N_r	= Number of Replacement Systems		

2.5 AREA OF COVERAGE MODEL

To describe the calculation of beamwidth and area of coverage, consider the following parameter definitions.

a = area of coverage major axis (statute miles)

b = area of coverage minor axis (statute miles)

R_E = radius of earth (3960 statute miles)

R_S = radius of synchronous orbit (26210 statute miles)

R_{SL} = slant range, satellite to beam center

ω = earth central angle, subsatellite point to beam center

γ = satellite antenna beam tilt

lat = beam center latitude

long = beam center relative longitude

β = antenna beamwidth subtending minor axis of pattern

α = antenna beamwidth subtending major axis of pattern

Then

$$\omega = \cos^{-1} [\cos(\text{lat}) \cos(\text{long})],$$

$$R_{SL} = (R_E^2 + R_S^2 - 2 R_E R_S \cos \omega)^{\frac{1}{2}},$$

$$\gamma = \sin^{-1} \left(\frac{R_E}{R_{SL}} \sin \omega \right), \text{ and}$$

$$\text{area of coverage} = \frac{\pi}{4} a b.$$

$$\text{Since } \omega + \gamma + \frac{\pi}{2} + El = \pi,$$

$$El = \frac{\pi}{2} - \gamma - \omega. \text{ Furthermore,}$$

$$R_L = R_S \cos \gamma - R_{SL}$$

$$\psi = \frac{b}{R_L}$$

$$\beta = 2 \sin^{-1} \left\{ \frac{R_L \sin \frac{\psi}{2}}{[(R_{SL} + R_L)^2 + R_L^2 - 2R_L(R_{SL} + R_L) \cos \frac{\psi}{2}]^{\frac{1}{2}}} \right\}$$

For the major axis, let

$$\zeta = \frac{a}{R_E} \quad \text{and}$$

$$\zeta' = \sin^{-1} \left[\frac{R_S}{R_E} \sin \left(\gamma - \frac{\alpha}{2} \right) \right] - \sin^{-1} \left[\frac{R_S}{R_E} \sin \left(\gamma + \frac{\alpha}{2} \right) \right] - \alpha$$

Because of the complexity of the expression for ζ' it is more practical to iterate for a solution to α rather than directly solve for it. This approach is as follows:

If after calculating β it is found that

$$\beta \geq 2 \left(\sin^{-1} \frac{R_E}{R_S} - \gamma \right),$$

β is adequate for coverage in both directions and can be accepted as the beamwidth. However, if this is not the case, it is necessary to iterate to the solution.

Let $\alpha'_0 = \beta$ and obtain ζ'_0 using $\alpha = \alpha'_0$. If ζ'_0 is sufficiently close to ζ , α'_0 can be accepted. Otherwise, let

$$\alpha'_i = \alpha'_{i-1} + \frac{(\zeta - \zeta'_{i-1})}{\left(\frac{d\zeta'}{d\alpha} \right)} \quad i = 1, 2, \dots, n$$

and replace α by α'_i to obtain ζ'_i . Iterate in this manner until the solution is acceptable or until the maximum number of desired iterations is performed. If no solution is achieved, β is used as the beamwidth.

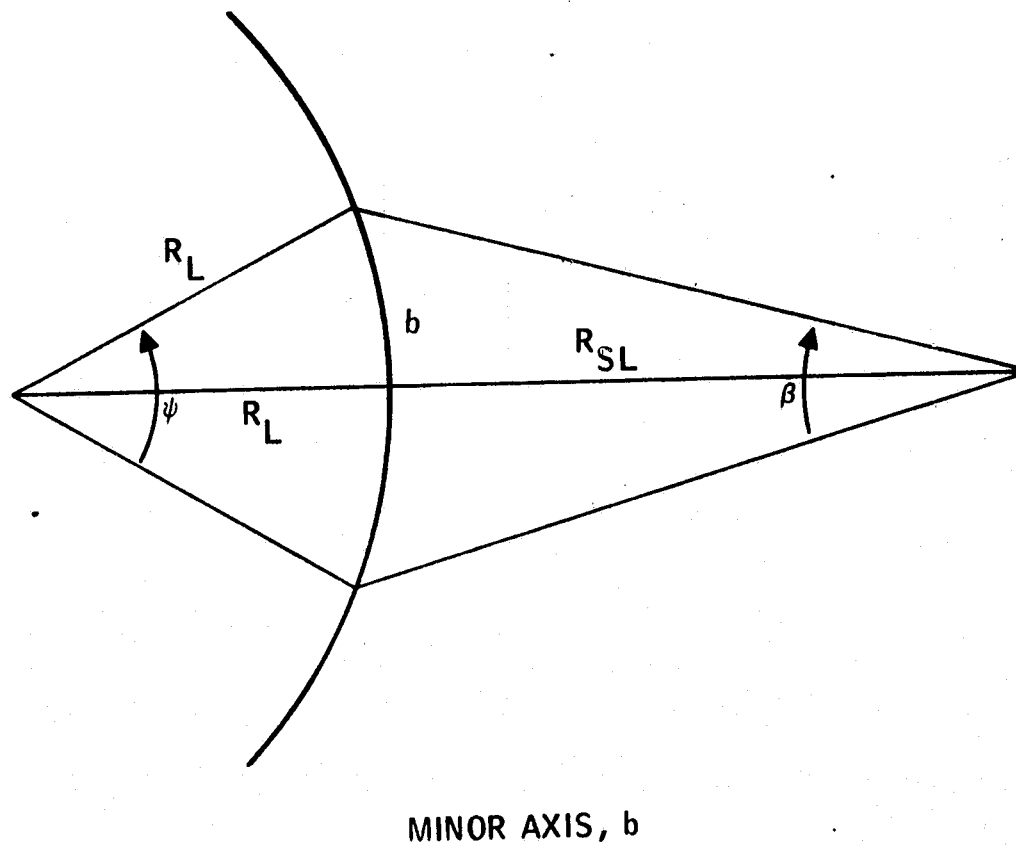


Figure 2.4 Beamwidth Geometry - Minor Axis

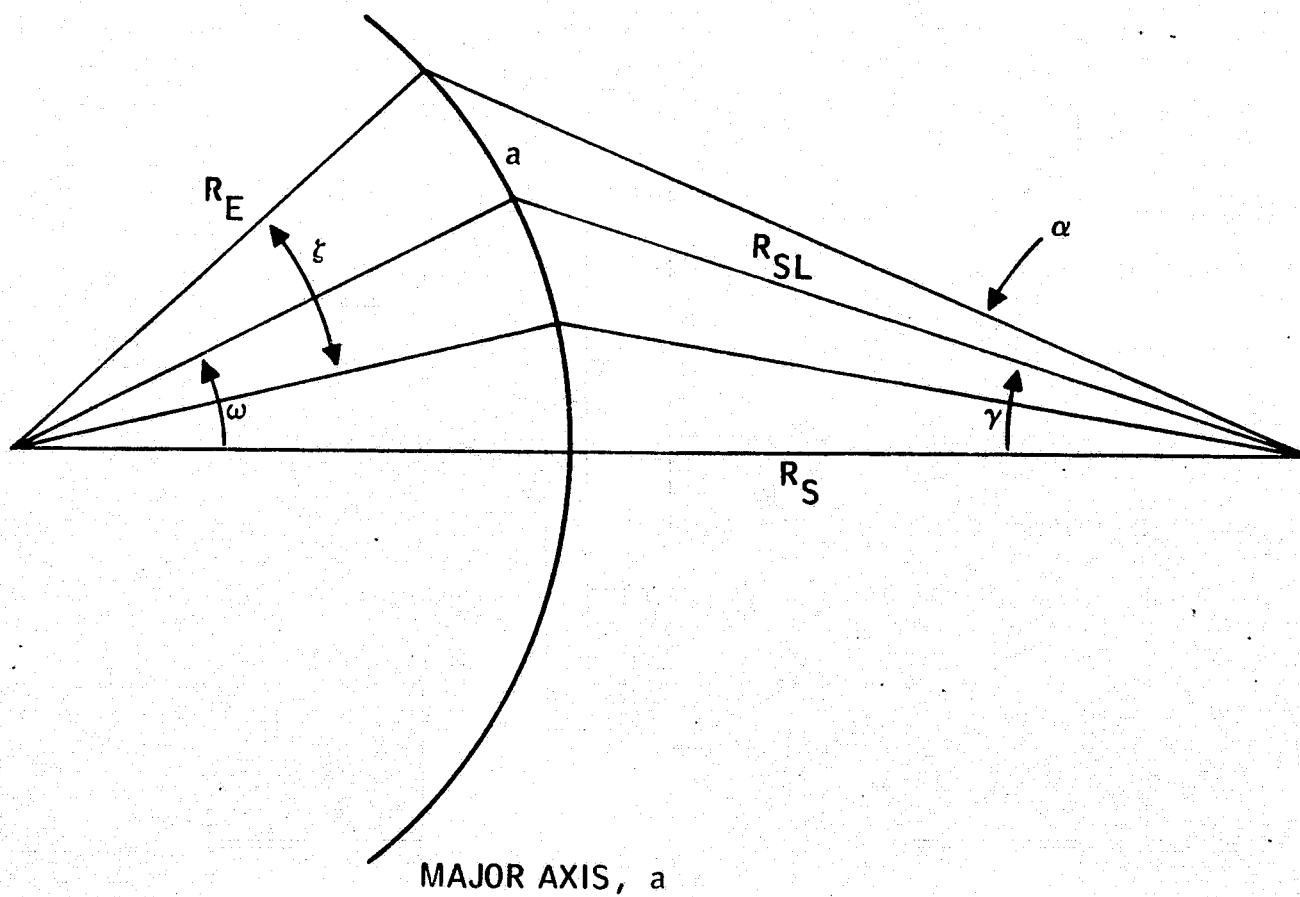


Figure 2-5 Beamwidth Geometry - Major Axis

2.6 SIGNAL ATTENUATION/NOISE MODELS

This section presents the various attenuation and noise factors which affect the propagation between the ground uplink station, the satellite and the ground receiving station and which are incorporated into the program. Associated curves may be found in Appendix D.

TABLE 2.2 ATTENUATION DUE TO RAIN

Horizontal Direction:

$$L_{RH} = (B) (R_p^A) \left[41.4 - 23.5 \log R_p \right] \text{ dB}$$

where R_p = precipitation rate in millimeters/hr.

A and B are given below as a function of frequency.

	FREQUENCY (GHZ)			
	.8	2.5	8.4	12.2
A	0	0.906	1.20	1.22
B	0	2.1×10^{-4}	8.5×10^{-3}	2.4×10^{-2}

In Direction of Satellite

For $\epsilon \geq \tan^{-1} \left[\frac{2}{41.4 - 23.5 \log R_p} \right]$, ϵ = elevation angle

$$L_R = \frac{(2)(B)(R_p^A)}{\sin \epsilon} \text{ dB}$$

For $\epsilon < \tan^{-1} \left[\frac{2}{41.4 - 23.5 \log R_p} \right]$

$$L_R = (B)(R_p^A) \left[\frac{41.4 - 23.5 \log R_p}{\cos \epsilon} \right] \text{ dB}$$

Vertical Extent = 2 km.

TABLE 2.3 ATTENUATION DUE TO CLOUDS

Vertical Extent = k 3m.

Water Vapor Density = 1 gm/m³

Visibility = 55 mi

In Direction of Satellite:

$$L_c = K_{ATTEN} \left(\frac{3}{\sin \epsilon} \right) \text{ dB}$$

	FREQUENCY (GHZ)			
	.8	2.5	8.4	12.2
K_{ATTEN}	5×10^{-3}	1.6×10^{-2}	7.0×10^{-2}	1.6×10^{-1}

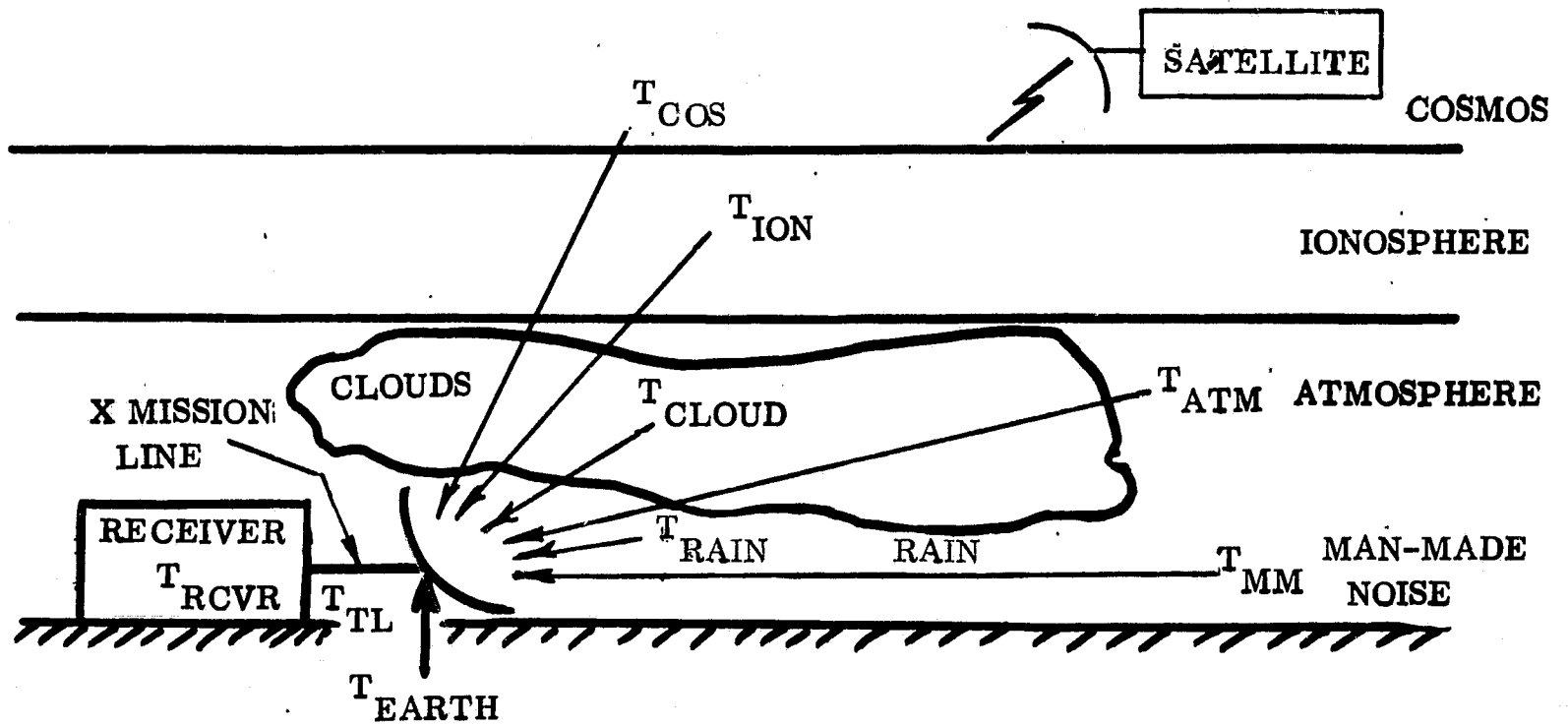
TABLE 2.4 ATTENUATION DUE TO ATMOSPHERE
(Water Vapor & Oxygen)

In Direction of Satellite:

$$L_A = 10 \log \left[\left(1 - \frac{T_Z}{290} \right)^{\csc \epsilon} \right] \text{ dB}$$

	FREQUENCY (GHZ)			
	.8	2.5	8.4	12.2
T_Z	1.8	2.6	3.5	4.8

TABLE 2.5 TVBS NOISE AND PROPAGATION MODELS



Region	Attenuation	Absorption	Ambient Temperature	Brightness Temperature Contributions
Cosmos	-	-	T_{COS}	$T_{COS} = L_I L_A L_C L_R T_{COS}$
Ionosphere	L_I	$1-L_I$	T_{ION}	$T_I = L_A L_C L_R (1-L_I) T_{ION}$
Atmosphere	L_A	$1-L_A$	T_{ATM}	$T_A = L_C L_R (1-L_A) T_{ATM}$
Clouds	L_C	$1-L_C$	T_{CLOUD}	$T_C = L_R (1-L_C) T_{CLOUD}$
Rain	L_R	$1-L_R$	T_{RAIN}	$T_R = (1-L_R) T_{RAIN}$
Horizontal Path	L_{RH}	$1-L_{RH}$	T_{RAIN}	
Transmission Line	L_{TL}	$1-L_{TL}$	T_{TL}	$T_T = (1-L_{TL}) T_{TL}$
Man-Made	-	-	T_{MM}	$T_M = L_{RH} T_{MM}$
Earth	-	-	T_{EARTH}	$T_E = T_{EARTH}$
Receiver	-	-	T_{RCVR}	$T_{RCVR} = T_{RCVR}$

The effective received noise temperature is

$$T_{\text{EFF}} = L_{\text{TL}} \left[(T_{\text{CO}} + T_{\text{I}} + T_{\text{A}} + T_{\text{C}} + T_{\text{R}}) \bar{G}_{\text{S}} + T_{\text{M}} \bar{G}_{\text{M}} + T_{\text{E}} \bar{G}_{\text{E}} \right] + T_{\text{T}} + T_{\text{RCVR}}$$

where $\bar{G}_{\text{S}}$ = Relative discrimination of antenna to cosmic and sky noise

$\bar{G}_{\text{M}}$ = Relative discrimination of antenna to man-made noise

$\bar{G}_{\text{E}}$ = Relative discrimination of antenna to hot earth

In this study, we will assume that since L_{I} is negligible above 100 MHz,

$$L_{\text{I}} = 1.$$

In addition the following ambient temperatures will be used:

$$T_{\text{EARTH}} = T_{\text{TL}} = T_{\text{ATM}} = 290^{\circ}\text{K}$$

$$T_{\text{RAIN}} = T_{\text{CLOUD}} = 273^{\circ}\text{K}$$

T_{MM} and T_{COS} are given by curves, and

$$T_{\text{RCVR}} = 290^{\circ}\text{K} \left[10^{\frac{\text{NF}}{10}} - 1 \right]$$

where NF is receiver noise figure in DB.

The total signal attenuation for the downlink is

$$L_{\text{TOT}} = L_{\text{I}} L_{\text{A}} L_{\text{C}} L_{\text{R}} L_{\text{FS}}$$

where L_{FS} in the free space attenuation = $\left(\frac{2}{4\pi R} \right)^2$.

TABLE 2.6 LOSS MODEL

The loss terms in the computer program consist of:

$$\text{free space} = \left(\frac{4 \pi R}{\lambda} \right)^2$$

$$\text{atmospheric} = L_I L_A L_C L_R$$

$$\text{circuit (XMIT)} = L_{TL3} L_{MUX} \quad (\text{downlink})$$

$$\text{circuit (XMIT)} = L_{TL1} L_{MUX} \quad (\text{uplink})$$

$$\text{circuit (RCVR)} = L_{TL4} \quad (\text{downlink})$$

$$\text{circuit (RCVR)} = L_{TL2} \quad (\text{uplink})$$

$$\text{pointing (XMIT)} = 0.1 \text{ db} \quad (\text{uplink and downlink})$$

$$\text{pointing (RCVR)} = 0.1 \text{ db} \quad (\text{uplink})$$

$$\text{pointing (RCVR)} = 4.0 \text{ db} \quad \begin{cases} 2.0\text{db pointing} \\ 2.0\text{db pattern contour} \end{cases} \quad (\text{downlink})$$

$$\text{polarization} = 0 \text{ db} \quad (\text{uplink})$$

$$\text{polarization} = 3.0 \text{ db} \quad (\text{downlink})$$

2.7 REDUNDANCY MODEL

The program computes factors which scale the affected weight, volume and cost terms to reflect the redundant elements required for additional spacecraft life.

Let

l_S = life of the satellite

l_I = life of the equipment in question

ρ = a scale factor such that $0 \leq \rho \leq 1$

If $d = \frac{l_S - l_I}{l_I}$, then

the scaling factor S is given by

$$S = 1 + \rho \cdot d.$$

A different scale factor is determined for the weight, volume, and cost terms.

The list of equipment for which redundancy is provided includes the following:

- a) downlink transmitter
- b) spacecraft receiver
- c) power transfer unit
- d) power conditioning unit
- e) power control unit
- f) L/c filter

Each requires that the lifetime l_I and the scaling factor ρ be input.

3

3.0 INPUT VARIABLES

This section provides definitions of all program input variables. The input list is comprised of three parts, NAME1, NAMEP1, and NAME2, respectively.

In the following list of definitions, the vectors containing the analytic coefficients for the models of the parametric relationships are represented by

$$\bar{V} = (\alpha, \beta) \quad \text{or}$$

$$\bar{V} = (\alpha, \beta, \gamma) *$$

depending on the required dimension and are stored as columns when part of an array (see Sections 3.1 and 5.5). The form of the equation is also given together with the number of pieces in which the curve was fit and the associated variable for the breakpoint. In general, the variables are in the order in which they appear in the data list in the program, and any subscripts are intended to indicate the maximum dimension the variable has.

3.1 NAMELIST/NAME1/

The following parameters and relations are defined for the spacecraft transmitter:

P_{TR}	= power output/channel
W_{TR}	= transmitter weight
V_{TR}	= transmitter volume
C_E	= engineering cost
C_M	= manufacturing cost
η	= transmitter efficiency

* The analytic models are, in the majority of cases, of the quadratic form:

$$y = \alpha + \beta x + \gamma x^2$$

$$\begin{aligned}
 p &= 10 \log P_{TR} \\
 w &= 10 \log W_{TR}; \quad W_{TR} = 10^{\frac{w}{10}} \\
 v &= 10 \log V_{TR}; \quad V_{TR} = 10^{\frac{v}{10}} \\
 C_e &= 10 \log C_E; \quad C_E = 10^{\frac{C_e}{10}} \\
 C_m &= 10 \log C_M; \quad C_M = 10^{\frac{C_m}{10}}
 \end{aligned}$$

where w , v , C_e , and C_m are related to p by equations of the form

$$w = \alpha + \beta p + \gamma p^2.$$

AWGHT(3,4,3)* - AM weight coefficients

FWGHT(3,4,3) - FM weight coefficients

AVOL1(3,4,3), AVOL2(3,4,3) - AM volume coefficients; fit in two parts:

AVOL1 , $p \leq 30$

AVOL2 , $30 < p \leq 50$

FVOL1(3,4,3) , FVOL2(3,4,3) - FM volume coefficients

ACENG1(3,4,3), ACENG2(3,4,3),

ACENG3(3,4,3) - AM cost of engineering coefficients;
fit in 3 parts:

ACENG1 , $10 \leq p \leq 21.14$

ACENG2 , $21.14 < p \leq 30$

ACENG3 , $30 < p \leq 43.01$

FCENG1(3,4,3), FCENG2(3,4,3),

FCENG3(3,4,3) - FM cost of engineering coefficients

* The triple subscripts on the transmitter coefficients are described in Section 5.5.

ACMFG1(3,4,3), ACMFG2(3,4,3) - AM cost of manufacturing coefficients;
fit in two parts:

$$\text{ACMFG1} \quad , \quad p \leq 21.14$$

$$\text{ACMFG2} \quad , \quad 21.14 < p$$

FCMFG1(3,4,3) FCMFG2(3,4,3) - FM cost of manufacturing coefficients

$$\eta = \alpha + \beta p + \gamma p^2$$

AETA(3,4,3) - AM efficiency coefficients

FETA(3,4,3) - FM efficiency coefficients

PSPPLY(4,2) - coefficients for weight and volume equations of power supply

Col. 1 - solar array

Col. 2 - nuclear reactor*

Solar array: $W_{pp} = \alpha + \beta P_{ps}$

$$V_{pp} = \gamma + \zeta P_{ps}$$

where P_{ps} = required prime power

For weight computations:

$$\alpha = \text{PSPPLY}(1, I)$$

$$\beta = \text{PSPPLY}(2, I)$$

For volume computations:

$$\gamma = \text{PSPPLY}(3, I)$$

$$\zeta = \text{PSPPLY}(4, I)$$

where I = power supply type

ANT(4,2) - coefficients and exponents for weight and volume equations of spacecraft antenna

Col. 1 - PETA

Col. 2 - Rigid

$$W_A = \alpha_1 \cdot D_3^{\beta_1}$$

$$V_A = \alpha_2 \cdot D_3^{\beta_2}$$

where D_3 = downlink antenna diameter and each column of the array is of the form: $V = (\alpha_1, \beta_1, \alpha_2, \beta_2)$.

For some satellite concepts the volume equation has been modified.

*Not implemented.

FBOOM(4,3) - coefficients and exponents for weight and volume of feed boom.

Col. 1 - folding

Col. 2 - expandable

Col. 3 - rigid

$$W_{FB} = \alpha_1 D_3^{\beta_1}$$

$$V_{FB} = \alpha_2 D_3^{\beta_2}$$

where

D_3 = downlink antenna diameter and each column is of the form:

$$V = (\alpha_1, \beta_1, \alpha_2, \beta_2).$$

BDWDTH(4) - frequency bandwidth allocation (GHZ)

DABD(8) - upper bound on downlink antenna diameter as a function of configuration type (ft.).

BLNGTH(8) - upper bound on equipment module length as function of configuration (ft.).

BVOL(8) - upper bound on payload volume capability as function of configuration (cu. ft.).

BWGHT(8) - upper bound on payload weight capability as function of configuration (lbs.).

DEM(8) - diameter of equipment module as function of configuration (ft.).

FREQ(4) - downlink frequency (GHZ)

EPSLON - ϵ used in convergence check on components of vector of independent design parameters, i.e., $\frac{|\Delta X_i|}{X_i} \leq \epsilon$

MCONV - number of successive iterations for which all components of independent parameters vector $\bar{X}$ must pass convergence check in order to identify convergent solution.

- ITMAX - maximum number of iterations of steepest-descent algorithm at which run is terminated (unless convergence test is satisfied at an earlier iteration).
- PSIK - K_y coefficient in convergence technique; factor approximating the amount by which the boundary violation is reduced. (See Section 2.2).
- SNRGI(7) - vector of signal to noise ratios (db) as a function of desired TASO grade
- DCYCLE - duty cycle (hrs.)
- WGHTB(3,3) - coefficients for battery weight equation (stored as columns)
- Col. 1 - nickel cadmium
- Col. 2 - silver cadmium
- Col. 3 - silver zinc

$$W_{BATT} = 10^{\beta_1/10}$$

where β_1 = quadratic in w_h , and $w_h = 10 \log$ (watt hrs)

- VOLB(3,3) - coefficients for battery volume equation, stored as columns;
- Col. 1 - nickel cadmium
- Col. 2 - silver cadmium
- Col. 3 - silver zinc

$$V_{BATT} = 10^{\beta_2/10}$$

where β_2 = quadratic in w_h

XTR(3) - unit manufacturing cost coefficients for ground uplink transmitter equation

$$C_{XTR} = 10^{C/10}$$

where C = quadratic in p_g ,

$$p_g = 10 \log P_g ;$$

P_g = power output/channel

ANTC(3) - unit manufacturing cost coefficients for ground uplink antenna equation

$$C_{ANT} = 10^{C/10}$$

where C = quadratic in d_1 ,

$$d_1 = 10 \log D_1 ;$$

D_1 = diameter of antenna (feet)

CSTR(3,3) - unit manufacturing cost coefficients for satellite structure; stored as columns.

Col. 1 - conventional

Col. 2 - (not used)

Col. 3 - (not used)

$$C_{STR} = 10^{S/10}$$

where S = quadratic in w_s ,

$$w_s = 10 \log W_{STRCT} ;$$

W_{STRCT} = weight of the structure

CP1(3,2), CP2(3,2) - unit manufacturing and R&D cost coefficients for power supply, stored as columns

Col.1- solar array

Col. 2 - nuclear reactor*

The form of the cost equations is:

* Not implemented.

$$\text{COST} = 10^{C/10}$$

where C = quadratic in p_s ,

$$p_s = 10 \log P_S;$$

P_S = prime power supply

RAD1(3,2), RAD2(3,2)

- unit manufacturing and R&D cost coefficients for thermal control, stored as columns:

Col. 1 - passive

Col. 2 - active

$$\text{COST} = 10^{C/10}$$

C = quadratic in p_r ,

$$p_r = 10 \log P_{RD};$$

P_{RD} = power radiated

SKLS1(2,2,3), SKLS2(2,2,3)

SK1(2,2,3), SK2(2,2,3)

- coefficients to determine engineering costs and manufacturing costs of the station-keeping.

The third subscript represents type:

array 1 - ammonia resistojet

array 2 - ion propulsion

array 3 - H_2O electrolysis

The cost equation for both engineering and manufacturing cost is of the form

$$\text{COST} = 10^{\frac{A + B \cdot \log l_s + C \cdot \log W_s + D \cdot (\log W_s)^2}{10}}$$

where l_s = life of the satellite and

W_s = total satellite weight - station keeping.

The term $A + B \cdot \log l_s$ represents a scaling factor based on lifetime. The 2X2 arrays represent the 2X1 coefficient vectors for the equation with a break point at 5000 lbs for W_s .

SKLS1, SKLS2:

Col. 1 = (A,B) for $W_s \leq 5000$

Col. 2 = (A,B) for $W_s > 5000$

SK1, SK2:

Col. 1 = (C,D) for $W_s \leq 5000$

Col. 2 = (C,D) for $W_s > 5000$

ATLS1(2,2,3), ATLS2(2,2,3)

ATT1(2,2,3), ATT2(2,2,3)

- coefficients to determine engineering costs and manufacturing costs of the attitude control. The third subscript represents type:

Array 1 - ammonia resistojet

Array 2 - ion propulsion

Array 3 - H_2O electrolysis

The cost equation for both the engineering and manufacturing cost is of the form:

$$COST = 10 \frac{A + B \cdot \log l_s + C \cdot \log AML + D(\log AML)^2}{10}$$

where l_s = life of the satellite and

$$AML = A \cdot \frac{M}{L} ;$$

A = area of solar array

M, L = variables determined by the concept type
(see Appendix D)

The 2X2 arrays represent the coefficients with a break point at 5000 for AML.

ATLS1, ATLS2:

Col. 1 = (A,B) for $AML \leq 5000$

Col. 2 = (A,B) for $AML > 5000$

ATT1, ATT2:

Col. 1 (C,D) for $AML \leq 5000$

Col. 2 (C,D) for $AML > 5000$

PLX1(4,3), PLX2(4,3)

- unit manufacturing and R&D cost coefficients for multiplexer

Col. 1 - coax

Col. 2 - waveguide

Col. 3 - stripline

The fourth element in each column is the cost for one channel.

In the case of multiple channels, the 3x1 column vector is used.

If the number of channels exceeds one,

$$COST = 10^{C/10}$$

where C = quadratic in N_C ,
 N_C = number of channels.

If there is only one channel ,

$$C_{mfg} = PLX1(4,I)$$

$$C_{R\&D} = PLX2(4,I)$$

where $I = 1,2,3$ depending on the multiplexer type.

DNLA, ANTD1(3,4), ANTD2(3,4)

- downlink antenna cost coefficients and break point.

ANTD1 is the array of manufacturing cost coefficients;

ANTD2 is the array of R&D cost coefficients. The curves may be fit in two parts with DNLA as the break point.

Col. 1 - TRUSS, diameter $\leq$ DNLA

Col. 2 - TRUSS, diameter $>$ DNLA

Col. 3 - Rigid, diameter $\leq$ DNLA

Col. 4 - Rigid, diameter $>$ DNLA

$$\text{COST} = 10^{C/10}$$

where

C = quadratic in d_3 ,

d_3 = $10 \log D_3$;

D_3 = diameter (in feet) of downlink antenna

CBT1(3,3), CBT2(3,3) - unit manufacturing cost and R&D cost coefficients
for battery

Col. 1 - nickel cadmium

Col. 2 - silver cadmium

Col. 3 - silver zinc

$$\text{COST} = 10^{C/10}$$

where

C = quadratic in w_h ,

w_h = $10 \log$ (watt hours)

RECA1, RECA2, ANTH(3,4) - cost coefficients for ground downlink antenna.

RECA1, RECA2 are break points allowing the curves to be
fit in 2 pieces.

Col. 1 - wideband antenna, diameter $\leq$ RECA1

Col. 2 - wideband antenna, diameter $>$ RECA1

Col. 3 - parabolic antenna, diameter $\leq$ RECA2

Col. 4 - parabolic antenna, diameter $>$ RECA2

$$C_{\text{mfg}} = 10^{C/10}$$

where C = quadratic in d_4 ,

d_4 = $10 \log D_4$;

D_4 = antenna diameter

XINST1, XINST2, CINST(3,4) - cost coefficients for installation of ground downlink antenna.

XINST1, XINST2 are the breakpoints allowing the curve to be fit in two pieces:

Col. 1 - wideband, diameter $\leq$ XINST1

Col. 2 - wideband, diameter $>$ XINST1

Col. 3 - parabolic, diameter $\leq$ XINST2

Col. 4 - parabolic, diameter $>$ XINST2

$$C_{in} = 10^{C/10} = \text{installation cost}$$

where

C = quadratic in d_4

CRCUR(3,2), CRCNR(3,2) - unit manufacturing and R&D cost coefficients for uplink receiver

Col. 1 - linear translator

Col. 2 - modulation converter

$$\text{COST} = \alpha + \beta N_c + \gamma M_{Nc}$$

where

N_c = number of channels

$$M_{Nc} = (N_c + 3)/4$$

NOANT(8) - number of spacecraft antennas as a function of concept

NCNFIG - maximum number of concept types to be considered

NMOD1 - maximum number of uplink modulations to be considered

NMOD2 - maximum number of downlink modulations to be considered

NFBAND - maximum number of frequency bands to be considered

NPTYPE - maximum number of power supply types to be considered

NXMTR - maximum number of transmitter types to be considered
 NRAD - maximum number of thermal control systems to be considered
 NSTK - maximum number of station keeping types to be considered
 NATT - maximum number of attitude control types to be considered
 NBATT - maximum number of battery types to be considered

The above values set the upper index bound on the do-loop parameter values.

M - number of dependent design parameters
 N - number of independent design parameters

RECH1(3,3,4), TRBD1(2,4), RECH2(3,3), TRBD2(2) - ground downlink
 unit manufacturing cost coefficients and breakpoints for the
 ground downlink receiver (cable)

RECH1 is the array containing the manufacturing cost coefficients
 for AM and TRBD1 is the array of breakpoints enabling the
 curve to be fit in as many as three pieces. The subscript 1-4
 indicates one of the 4 frequency operating points and the 3X3
 array is described as follows:

Col. 1 - $TR \leq TRBD(1,I)$

Col. 2 - $TRBD(1,I) < TR$

Col. 3 - $TRBD(2,I) \geq TR$

where TR = temperature of the receiver

$I = 1,2,3,4$

RECH2 is the array containing the cost coefficients for FM. It is
 assumed that the costs are not dependent on frequency. TRBD2
 are the breakpoints to fit the curve as above. The column
 designations are the same as for RECH1.

$$COST = 10^{C/10}$$

where $C = \alpha + \beta t + \gamma t^2$

$t = 10 \log_{10} T;$

T = receiver temperature

BVEST - vestigial bandwidth required/channel

BBB(2) - base bandwidth required/channel

ZMOD(2) - factor used to compute improvement factor index

ZMOD(1) = 1 (AM)

ZMOD(2) = modulation index used to compute the FM improvement factor and bandwidth $I_{\text{mprv}} = 3 \cdot \text{ZMOD}(2)^2 \cdot (1 + \text{ZMOD}(2))$

UPFREQ(2) - uplink frequency operating points (GHZ)

RPP(3) - array of precipitation rates (mm/hr)

RP = RPP(1) if $0^\circ \leq \text{lat} < 20^\circ$

RP = RPP(2) if $20^\circ \leq \text{lat} < 40^\circ$

RP = RPP(3) if $40^\circ \leq \text{lat}$

AT1(2), AT2(4),

BT1(2), BT2(4) - arrays containing the coefficients and exponents for calculating attenuations due to rain in the horizontal direction

$$L_{\text{RH}} = B * R_p^A \left[41.4 - 23.5 \log_{10} R_p \right] \text{ db}$$

and the direction of the satellite: $L_{\text{RH}} / \cos(E_\ell)$

where

E_ℓ = elevation angle

BT1 and AT1 contain the values for the uplink frequency operating points.

BT2 and AT2 contain the values for the downlink frequency operating points.

The coefficient values are in BT1, BT2; the exponents in AT1, AT2.

KATTN1(2), KATTN2(4) - coefficients for calculating attenuation due to clouds. Subscripts denote operating frequency points. KATTN1 for uplink; KATTN2 for downlink.

$$L_c = K_{\text{atten}} \cdot \frac{3}{\sin E_\ell} \quad (\text{db})$$

where

E_ℓ = elevation angle

TZ1(2), TZ2(4) - temperature values as function of frequency operating points to calculate attenuation due to atmosphere (water vapor and oxygen).

TZ1 - uplink

TZ2 - downlink

$$L_A = 10 \log \left[\left(1 - \frac{T_Z}{290^\circ} \right)^{\csc E_\ell} \right] \quad (\text{db})$$

where E_ℓ = elevation angle

- XLI - ionosphere attenuation
 - XLT1 - ground uplink
 - XLT2 - spacecraft receiving
 - XLT3 - spacecraft transmitting
 - XLT4 - ground downlink
- Transmission line losses
- XPT1 - transmitter pointing loss (ground uplink)
 - XPT2 - transmitter pointing loss (spacecraft)
 - RPT1 - receiver pointing loss (spacecraft)
 - RPT2 - receiver pointing loss (ground downlink)
 - P1 - polarization loss (uplink)
 - P2 - polarization loss (downlink)

The above losses are input in db.

- COSNT(4) - cosmic ambient temperature as function of operating frequency
- TION - ionospheric ambient temperature
- TATM - atmospheric ambient temperature
- TCLD - cloud ambient temperature

TRAIN - rain ambient temperature
 TTL - transmission line ambient temperature
 AMMT(3,4) - coefficients and exponents to calculate man-made ambient temperature as a function of frequency operating point

$$TMM = \alpha + \beta p^{\gamma}$$

p = population density

TE - ambient earth temperature
 TRCVRS - ambient temperature at satellite receiver

GE1, GE2, GEX(3,3) - coefficients to calculate relative discrimination of antenna to hot earth; GE1, GE2 are break points allowing the curve to be fit in 3 pieces.

Col. 1 - $g \leq GE1$

Col. 2 - $GE1 < g \leq GE2$

Col. 3 - $GE2 < g$

where

g = gain of ground downlink antenna in db. GE1, GE2 are in db.

$$G_E = \alpha + \beta g + \gamma g^2$$

GSX(3,3), GS1,GS2 - coefficients and breakpoints to calculate relative discrimination of antenna to cosmic and sky noise (same as above).

GMX(3,3), GM1,GM2 - coefficients and breakpoints to calculate relative discrimination of antenna to man-made noise (same as above).

AMTPLX(4), BMTPLX(4) - coefficients and exponents to calculate multiplexer loss as function of frequency

$$LOSS = A \cdot N_0^B$$

D3BD - lower diameter bound on downlink antenna for concept E.

ARCVR(4) - coefficients used in calculating cost of receiver feed line as function of frequency

$$C_{TL} = A \cdot L_{TL}^B \quad \text{where } L_{TL} = \text{transmission line attenuation in db/ft.}$$

EXPR - exponent in cost of receiver feed line equation

FDLINE - length of receiver feed line in feet

ETAPT - efficiency of power transfer unit

ETAPC - efficiency of power conditioning unit

ETAB - efficiency of battery and charge control unit

ETAPCU - efficiency of power control unit

ETALCF - efficiency of L/c Filter unit

RHOPC - $(1 - \rho)$ where $\rho = \%$ darktime used to calculate power input to the power conditioning unit: $P_{PT} = P_B \left(1 + \frac{P_C}{\eta_B}\right)$

where P_B = power out of the battery and η_B = efficiency of the battery and charge control unit

DGRD(2) - coefficients used to calculate degradation factor on prime power supply

$$\rho = \alpha - \beta \log_{10} (t_{sat})$$

where t_{sat} = life of the satellite

then prime power supply = PPS/ ρ

WPTR(3) - coefficients to calculate weight of the power transfer unit

$$W_{PT} = \alpha + \beta P_{PT} + \gamma P_{PT}^2$$

where P_{PT} = power out of the power transfer unit

WCU(3,3), WCU1, WCU2 - coefficients to calculate weight of the power control unit. WCU1 and WCU2 are breakpoints allowing the curve to be fit in three pieces.

$$\text{Col. 1} - P_{\text{pcu}} \leq \text{WCU1}$$

$$\text{Col. 2} - \text{WCU1} < P_{\text{pcu}} \leq \text{WCU2}$$

$$\text{Col. 3} - \text{WCU2} < P_{\text{pcu}}$$

where P_{pcu} = power out of the power control unit

$$W_{\text{pcu}} = 10^{W/10}$$

where W = quadratic in p ,
 $p = 10 \log P_{\text{pcu}}$

WPCN(3) - coefficients to calculate weight of the power conditioning unit

$$W_{\text{pc}} = 10^{W/10}$$

where W = quadratic in p ,
 $p = 10 \log P_{\text{pc}}$

P_{pc} = power out of the power conditioning unit

WLFCF(2,2), WLFCF1 - coefficients to calculate weight of ℓ/c filter.

WLFCF1 is a breakpoint allowing the curve to be fit in two pieces.

$$W_{\ell c} = \alpha \cdot P_{\ell c}^{\beta} \cdot N_c$$

where N_c = number of channels
 $P_{\ell c}$ = power out of ℓ/c filter.

VPTR(3,2), VPTR1 - coefficients to calculate volume of power transfer unit. VPTR1 is a breakpoint allowing the curve to be fit by two pieces.

$$\text{Col. 1} - P_{pt} \leq VPTR1$$

$$\text{Col. 2} - VPTR1 < P_{pt}$$

$$V_{pt} = 10^{v/10}$$

where $v = \text{quadratic in } p$,

$$p = 10 \log P_{pt}$$

VCU(3,2), VCUL - coefficients to calculate volume of power control unit. VCUL is the breakpoint to allow the curve to be fit as two pieces.

$$\text{Col. 1} - P_{pcu} \leq VCUL$$

$$\text{Col. 2} - VCUL < P_{pcu}$$

$$V_{cu} = 10^{v/10}$$

where $v = \text{quadratic in } p$,

$$p = 10 \log P_{pcu}$$

VPCN(3,2), VPCN1 - coefficients to calculate volume of power conditioning unit. VPCN1 is a breakpoint to allow the curve to be fit in two pieces.

$$\text{Col. 1} - P_{pc} \leq VPCN1$$

$$\text{Col. 2} - VPCN1 < P_{pc}$$

$$V_{pc} = 10^{v/10}$$

where $v = \text{quadratic in } p$,

$$p = 10 \log P_{pc}$$

VLCFP(2,2), VLCFPI - coefficients to calculate volume of l/c filter.

VLCFPI is the breakpoint allowing the curve to be fit in two pieces.

$$\text{Col. 1} - P_{LC} \leq VLCFPI$$

$$\text{Col. 2} - VLCFPI < P_{LC}$$

$$V_{LC} = \alpha \cdot P_{LC}^{\beta} N^{\gamma} C^{\delta}$$

CENGPC(3,3), CENGPI, CENGP2 - coefficients to calculate engineering costs of the power transfer unit, the power control unit, and the power conditioning unit. CENGPI, CENGP2 are breakpoints allowing the curve to be fit in three pieces.

Col.1- power $\leq$ CENGPI

Col.2- CENGPI < power $\leq$ CENGP2

Col.3- CENGP2 < power

where power is the power out of one of the three units mentioned above;

$$\text{then } C_{\text{eng}} = C_1 \cdot 10^{C_2/10}$$

where C_2 = quadratic in p

p = 10 log power;

C_1 = function of which unit under consideration

CMFGPC(3,2), CMFGPI - coefficients to calculate manufacturing costs of the power transfer unit, the power control unit, and the power conditioning unit. CMFGPI is a breakpoint allowing the curve to be fit in two pieces.

Col. 1 - power $\leq$ CMFGPI

Col. 2 - CMFGPI < power

where power is one of the three mentioned above.

$$\text{then } C_{\text{mfg}} = C_1 \cdot 10^{C_2/10}$$

where C_2 = quadratic in p ,

p = 10 log power ;

C_1 = function of the unit under consideration

CMFGLC(2,3), CENGLC(2,3), CLC1, CLC2 - manufacturing and R&D cost coefficients for ℓ/c filter. CLC1 and CLC2 are two breakpoints to allow the curve to be fit in three pieces.

Col. 1 - $P_{LC} \leq CLC1$

Col. 2 - $CLC1 < P_{LC} \leq CLC2$

$$\text{Col. 3} - \text{CLC2} < P_{LC}$$

$$\text{then } C = \alpha + \beta P_{LC}$$

TSTRCT - thickness of the structure (feet)

- LTR - life (in years) of the downlink transmitters
- LRCVR - life (in years) of the spacecraft receiver
- LPT - life (in years) of the power transfer unit
- LBATT(3) - life (in years) of the battery

subscript designates battery type

- 1 - nickel cadmium
- 2 - silver cadmium
- 3 - silver zinc

- LPC - life (in years) of power conditioning unit
- LPCU - life (in years) of power control unit
- LLCF - life (in years) of LC filter

- RRWTR - scaling factor ρ used to determine redundancy factor for weight of the transmitter (See Section 2.7).
- RRWRVR - scaling factor ρ used to determine redundancy factor for weight of the receiver
- RRWPT - scaling factor ρ used to determine redundancy factor for weight of the power transfer unit
- RRWBTT - scaling factor ρ used to determine redundancy factor for weight of the battery
- RRWPC - scaling factor ρ used to determine redundancy factor for weight of the power supply
- RRWPCU - scaling factor ρ used to determine redundancy factor for weight of the power control unit
- RRWLCF - scaling factor ρ used to determine redundancy factor for weight of the l/c filter
- RRVTR - scaling factor used to determine redundancy factor for volume of the transmitter
- RRVRVR - scaling factor used to determine redundancy factor for volume of the receiver

RRVPT - scaling factor used to determine redundancy factor for the volume of the power transfer unit

RRVBTT - scaling factor used to determine redundancy factor for the volume of the battery

RRVPC - scaling factor used to determine redundancy factor for the volume of the power control unit

RRVPCU - scaling factor used to determine redundancy factor for the volume of the power conditioning unit

RRVLCF - scaling factor used to determine redundancy factor for the volume of the l/c filter

RRCTR - scaling factor used to determine redundancy factor for the cost of the transmitter

RRCRVR - scaling factor used to determine redundancy factor for the cost of the receiver

RRCPT - scaling factor used to determine redundancy factor for the cost of the power transfer unit

RRCBTT - scaling factor used to determine redundancy factor for the cost of the battery

RRCPC - scaling factor used to determine redundancy factor for the cost of the power control unit

RRCPCU - scaling factor used to determine redundancy factor for the cost of the power conditioning unit

RRCLCF - scaling factor used to determine redundancy factor for the cost of the l/c filter

VRHO - contingency factor for volume on spacecraft (≥ 1)

POP1, POP2 - population density cut off points for environment

Rural - population density $\leq$ POP1

Suburban - POP1 < population density $\leq$ POP2

Urban - POP2 < population density

POP(4) - array of population density to be considered

WMUX(3,3) - coefficients and exponents to determine weight of the multiplexer, stored by columns

Col. 1 - coax

Col. 2 - waveguide

Col. 3 - stripline

$$W_{mplx} = WMUX(3,I) \cdot NBS, \text{ if } NCB = 1$$

$$W_{mplx} = (\alpha \cdot 10^{\beta \cdot NCB}) \cdot NBS, \text{ if } NCB > 1$$

where

NBS = number of beams/satellite

NCB = number of channels/beam

$$(\alpha, \beta) = [WMUX(1,I), WMUX(2,I)]$$

and I = 1, 2, 3 depending on multiplexer type

WHARR(3,8) - coefficients to determine weight of the harnessing, stored in columns (dependent on concept type)

$$W_{HAR} = \alpha + \beta P_{pt} + \gamma P_{pt}^2$$

where

P_{pt} = power out of the power transfer unit

NPROTO - number of satellite prototypes

IMO, IDA, IYR - calendar date of run

IMO = month (1, ..., 12)

IDA = day (1, ..., 31)

IYR = (69, 70, ...)

XNGRMX - maximum value for the total population in computing ground facility cost learning curve

3.2 NAMELIST/NAMEP1/

NAMELIST/NAMEP1/ is an extension of NAME1 and contains the following parameters.

XLIFE(6)	- array of satellite lifetimes (to be used in do-loop system)
PGMIN,PGMAX	- minimum and maximum value for power output/channel on ground uplink transmitters
DIMIN,DIMAX	- minimum and maximum value for ground uplink antenna diameter
D4MIN,D4MAX	- minimum and maximum value for ground downlink antenna diameter
TRMIN(2), TRMAX(2)	- minimum and maximum value for ground downlink receiver temperature (1 - UHF; 2 - other than UHF)
XLMIN,XLMAX	- minimum and maximum value for spacecraft lifetime (not used)
DFMREC	- FM cost increment to be added to ground downlink receiver (cable)
PARANT(4)	- scale factor (relative to UHF) to be applied to parabolic antenna cost, as function of frequency
FREQ	.8GHz 2.5GHz 8.2GHz 12.5GHz
PARANT	1. 1.05 1.10 1.15
BWUPL	- uplink beamwidth (deg)
ACQLV(8)	- array of acquisition costs for launch vehicles ($\$10^6$)
ENGLV(8)	- array of engineering costs for launch vehicles ($\$10^6$)
VSABD	- boundary on the volume of the solar array (cu.ft)
ACOMP(8)	- array of acquisition costs for manned provisions as a function of concept ($\$10^6$)
ENGMP(8)	- array of engineering costs for manned provisions as a function of concept ($\$10^6$)

URCBSM	- unit recurring cost for BSM structure
URCMDA	- unit recurring cost for MDA
ENGBSM	- engineering cost for BSM structure
ENGMDA	- engineering cost for MDA
PTRMIN	- minimum value for power output/channel on spacecraft transmitter
PTRMX1, PTRMX2, PTRMX3	- maximum value for power output/channel on spacecraft transmitter
	PTRMX2 - freq ≤ 2.5 , multiplexer waveguide
	PTRMX3 - freq ≤ 2.5 , solid state
	PTRMX1 - all others
BMWDTH	- downlink beamwidth (deg) if input
IBWC	- flag indicator on whether BMWDTH is computed or input
	IBWC = 0 input
	IBWC $\neq 0$ compute

GROUND RECEIVER COMPUTATIONS FOR DIRECT TO HOME

RCVR1(2)	- curve fitting breakpoints for UHF in terms of receiver noise temperatures ($^{\circ}\text{K}$) - breakpoint one for AM; breakpoint two for FM
RCVR2(1)	- curve fitting breakpoint if not UHF
HRCVR1(3,4)	- coefficients for quadratic exponential fit on cost of home receiver for UHF. Col. 1 and 2 for two piece fit on AM receiver. Col. 3 and 4 for FM. The curves are fit as a function of $10 \log T$ where T = noise temperature ($^{\circ}\text{K}$)
HRCVR2(3,2)	- coefficients for two piece quadratic exponential fit on cost of home receiver for frequencies greater than UHF. The curves are fit as a function of $10 \log T$ where T = noise temperature ($^{\circ}\text{K}$)
XLC1(3,2)	- learning curve coefficients for basic home receiver unit recurring cost as a function of $10 \log X$ where X = total no. of receivers/3 based on annual production.

Col. 1 - UHF

Col. 2 - Not UHF

- XLC2(3,1) - learning curve coefficients for the first additional block of 3 channels for AM and 1 channel for FM
- XLC3(3,1) - learning curve coefficients for each additional such block
- RCVC2 - constant to compute first additional block (see equation below)
- RCVC3 - constant to compute each additional block

Additional blocks are computed using

$$\Delta C = \text{CONST} * R$$

where $R = 10^{A/10}$

and $A = \alpha + \beta X + \gamma X^2$

X = total no. of receivers on annual production basis

The (α, β, γ) are given by XLC2 or XLC3

3.3 NAMELIST/NAME2/

NAMELIST/NAME2/contains the variables which specify the user requirements.

NB	- number of beams
NCB	- number of channels/beam
NF	- number of facilities/satellite system
NRS	- number of reserve satellites
NRLV	- number of reserve launch vehicles
NRTSTA	- number of retransmission stations for retransmission system
LSYS	- life of the system (years)
LSAT	- life of the satellite (years)
NTRF	- number of transmitters/uplink facility
NCF	- number of channels/uplink facility
NCTRF	- number of channels/transmitter/uplink facility
NBRQD	- number of beams required/system
ICNFIG(8)	- array of flags to determine which concept types are to be considered
ICNFIG(I) = 0, do not consider concept I	
ICNFIG(I) = 1, do consider concept I	
I = 1, ..., number of concept types	
IMOD1(2)	- array of flags to determine which uplink modulations are to be considered (see ICNFIG)
IMOD2(2)	- array of flags to determine which downlink modulations are to be considered (see ICNFIG)
IFBAND(4)	- array of to determine frequency bands which are to be considered (see ICNFIG).

IXMTR(4) - array of flags to determine which transmitter types are to be considered (see ICNFIG)
 IRAD(2) - array of flags to determine which thermal control systems are to be considered (see ICNFIG)
 ISTK(3) - array of flags to determine which station-keeping types are to be considered (see ICNFIG)
 IATT(3) - array of flags to determine which altitude control types are to be considered (see ICNFIG)
 IBATT(3) - array of flags to determine which battery types are to be considered
 IPTYPE(2) - array of flags to determine which power supply system is to be considered
 XLAT,XLONG- latitude and relative longitude of beam center (degrees)
 AMAJR,BMINR- major and minor axes of area of coverage (statute miles)
 NAPV - number of audio channels/video
 I11 - uplink antenna type
 I19 - broadcast type
 I21 - type of downlink system
 1- Direct
 2- Retransmission (not implemented)
 3- Cable
 RHO - ratio of satellite noise to ground noise
 IBW - set to zero whenever latitude and longitude are re-initialized; thereafter it is set to one. This eliminates some unnecessary initialization in the driver.
 NSAT - number of satellite systems
 ITASOI(7) - array of flags to determine which TASO grades are to be considered
 IPDNSE(4) - array of flags to determine which population densities are to be considered
 INGR(8) - array of flags to determine which ground receiver population/beam are to be considered
 TITLE(6) - array to provide a descriptive title on the run summary of the case being run. (Maximum of 60 Hollerith characters).
 ICASE - identification number of last case run
 XNGRI(8) - array of ground receiver population/beam values to be considered.

- XO(3,8) - array of initial values for $\bar{X}$; stored by columns and determined by the configuration
- JIT - number of desired iterations through system equations for iterative solution of implicit equations. (Nominal value of 2 is adequate).
- SETDNS - TV set density for cable distribution system (sets/sq. mi).

Other variables contained in this list which also appear in either NAME1 or NAMEP1 include:

SNRGI, PGMIN, PGMAX, DIMIN, DIMAX, D4MIN, D4MAX, TRMIN, TRMAX, XLMIN, XLMA
XLMAX, ILIFE, NRGR, BWUPL, ACQLV, ENGLV, VSABD, ACQMP, ENGMP, URCBSM,
URCMDA, ENGBSM, ENGMDA, PTRMIN, PTRMX1, PTRMX2, PTRMX3, BWIDTH, IBWC,
NOANT

4

4.0 SAMPLE COMPUTER RUN

This section presents a sample of the nominal input parameter values which are defined in Section 3 and an example of a typical run summary output.

The input values are presented in Figure 4.1; the sample output in Figure 4.2.

4-2

GDC-DCF70-002

FIG. 4.1 TVBS INPUT DATA (Cont)

FCENG3	= 0.549E+02, -0.52E-01, 0.54F-02, 0.5875E+02, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.5875E+02, 0.0, 0.0, 0.5192E+02, 0.189E+00, 0.16E-02, 0.0, 0.0, 0.0, 0.0, 0.0, 0.189E+00, 0.16E-02,	0.5384E+02, 0.98E-01, 0.25E-02, 0.5672E+02, 0.585E-01, 0.285E-02, 0.0, 0.0, 0.0, 0.0, 0.5192F+02,
FCMG1	= 0.4468E+02, 0.33E-01, 0.25E-02, 0.4321E+02, 0.11F+00, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.4321E+02, 0.11E+00, 0.0, 0.4615E+02, -0.1485E+00, 0.625E-02, 0.0, 0.0, 0.0, 0.0, 0.0, -0.1485E+00, 0.625E-02,	0.4721E+02, -0.45E-02, 0.235E-02, 0.5124E+02, -0.2895E+00, 0.895E-02, 0.0, 0.0, 0.0, 0.0, 0.4615F+02,
FCMG2	= 0.4468E+02, 0.33E-01, 0.25E-02, 0.2766E+02, 0.845E+00, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.2766E+02, 0.845E+00, 0.0, 0.4615E+02, -0.1485E+00, 0.625E-02, 0.0, 0.0, 0.0, 0.0, 0.0, -0.1485E+00, 0.625E-02,	0.4721E+02, -0.45E-02, 0.235E-02, 0.5124E+02, -0.2895E+00, 0.895E-02, 0.0, 0.0, 0.0, 0.0, 0.4615F+02,
FETA	= -0.45E-01, 0.35E-01, -0.5F-03, 0.3825E+00, -0.25E-03, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.2967E+00, -0.67E-03, 0.0, 0.85E-01, 0.273E-01, -0.325E-03, 0.0, 0.0, 0.0, 0.0, 0.0, 0.255E-01, -0.325E-03,	0.17E+00, 0.2575E-01, -0.325E-03, -0.375E-01, 0.231E-01, -0.313F-03, 0.0, 0.0, 0.0, 0.0, 0.975E-01,
PSPLY	= 0.1E+00, 0.0, 0.167E-01, 0.0, 0.0, 0.0, 0.0, 0.0,	
ANT	= 0.1E+00, 0.2E+01, 0.28E-02, 0.3E+01, 0.64E+01, 0.5E+00,	0.3E+00, 0.3E+01,
FROM	= 0.32E+00, 0.169E+01, 0.39E-01, 0.2E+01, 0.33E+00, 0.132E+01, 0.169E+01, 0.41E-01, 0.7E+01,	0.16E-03, 0.3E+01, 0.32E+00,
XIMPRV	= 0.1E+01, 0.108E+03,	
ROWDTH	= 0.42E+00, 0.1E+01, 0.2E+00, 0.1E+01,	
DARD	= 0.15E+02, 0.3E+02, 0.56E+02, 0.15E+02, 0.9E+02, 0.33E+02,	0.0, 0.1E+03,
BLNGTH	= 0.5E+01, 0.75E+01, 0.26F+02, 0.0, 0.2E+02, 0.0, 0.0,	0.35E+02,
RVOL	= 0.98E+02, 0.262E+03, 0.81F+03, 0.6E+04, 0.45E+04, 0.6E+04,	0.0, 0.6E+04,
RWHT	= 0.2E+04, 0.52E+04, 0.9E+04, 0.42E+05, 0.98E+04, 0.42E+05,	0.0, 0.42E+05,
DEM	= 0.4E+01, 0.567E+01, 0.567E+01, 0.0, 0.7E+00, 0.0, 0.0,	0.1E+02,
FREQ	= 0.8E+00, 0.25E+01, 0.84E+01, 0.122E+02,	
EPSLON	= 0.1E-02,	
MCONV	= 4,	
ITMAX	= 40,	
PSIK	= 0.1E+01,	
SNRGI	= 0.445E+02, 0.335E+02, 0.27E+02, 0.23E+02, 0.17E+02, 0.1E+02,	0.53E+02,
DCYCLE	= 0.24E+02,	
WGMTR	= -0.10778E+02, 0.942E+00, 0.0, -0.1365E+02, 0.935E+00, 0.0,	-0.1642E+02, 0.93E+00, 0.0,
VOLB	= -0.3231E+02, 0.937E+00, 0.0, -0.354E+02, 0.915E+00, 0.0,	-0.3674E+02, 0.918E+00, 0.0,
XTR	= -0.1285E+02, 0.175E-01, 0.375E-02,	
ANYC	= -0.392E+02, 0.185E+01, 0.0,	

FIG. 4.1 TVBS INPUT DATA (Cont)

CSTR = -0.2332E+02, 0.723E+00, 0.0, -0.6E+02, 0.0, 0.0, 0.0, 0.0, 0.0.
 CP1 = -0.31511E+02, 0.8681E+00, 0.4245E-03, 0.0, 0.0, 0.0.
 CP2 = -0.336534E+02, 0.825E+00, 0.4075E-03, 0.0, 0.0, 0.0.
 RAD1 = -0.33071E+02, 0.683E+00, 0.0, -0.36676E+02, 0.683E+00, 0.0.
 RAD2 = -0.24023E+02, 0.675E+00, 0.0, -0.2752E+02, 0.675E+00, 0.0.
 SK1 = 0.124E+00, 0.0, 0.547E+00, 0.0, 0.136E+00, 0.0, 0.136E+00, 0.0, 0.0, 0.0, 0.0.
 SK2 = 0.19E+00, 0.0, 0.49E-01, 0.0, 0.108E+00, 0.0, 0.108E+00, 0.0, 0.0, 0.0, 0.0.
 PLX1 = -0.1517E+01, 0.913E-01, -0.308E-02, 0.15E-01, -0.1566E+01, 0.9575E-01, -0.3367E-02, 0.14E-01, -0.1647E+01, 0.925E-01, -0.322E-02, 0.11E-01.
 PLX2 = -0.782E+00, 0.106E+00, -0.38E-02, 0.85E-01, -0.841E+00, 0.1095E+00, -0.4E-02, 0.8E-01, -0.1308E+01, 0.322E-01, -0.373E-02, 0.7E-01.
 DNLA = 0.5E+01.
 ANTD1 = -0.9733E+01, 0.2408E+00, 0.179E-01, -0.9733E+01, 0.2408E+00, 0.179E-01, -0.3523E+02, 0.0, 0.0, -0.3679E+02, 0.7633E-01, 0.1107E-01.
 ANTD2 = -0.7033E+01, 0.2369E+00, 0.178E-01, -0.7033E+01, 0.2369E+00, 0.178E-01, -0.1523E+02, 0.0, 0.0, -0.1438E+02, -0.3496E+00, 0.3462E-01.
 CRT1 = -0.5328E+02, 0.354E+00, 0.85E-02, -0.4946E+02, 0.215E+00, 0.144E-01, -0.5235E+02, 0.345E+00, 0.1E-01.
 CRT2 = -0.269E+02, 0.4E+00, 0.0, -0.269E+02, 0.4E+00, 0.0, -0.269E+02, 0.4E+00, 0.0.
 RECA1 = 0.4E+01.
 RECA2 = 0.1E+02.
 ANTH = -0.4722E+02, -0.136E-01, 0.9E-01, -0.4641E+02, -0.448E+00, 0.137E+00, -0.392E+02, -0.18E+00, 0.1E+00, -0.694E+02, 0.384E+01, 0.0.
 XINST1 = 0.4E+01.
 XINST2 = 0.8E+01.
 CINST = -0.46E+02, -0.631E-01, 0.74E-01, -0.439E+02, -0.944E+00, 0.162E+00, -0.46E+02, -0.27E+00, 0.12E+00, -0.605E+02, 0.257E+01, 0.0.
 CRCUR = 0.5E-02, 0.333E-02, 0.3167E-01, 0.5E-02, 0.333E-02, 0.3667E-01.
 CRCNR = 0.14E+00, 0.667E-02, 0.5333E-01, 0.17E+00, 0.8333E-01, 0.6167E-01.
 NOANT = 0.4E+01.
 NCNFIG = 8.
 NMOD1 = 2.
 NMOD2 = 2.
 NFRAND = 4.
 NPTYPE = 2.
 NXMTR = 4.

FIG. 4.1 TVBS INPUT DATA (Cont)

NMTPLEX = 2,
 NRAD = 2,
 NSTK = 3,
 NATT = 3,
 NRATT = 3,
 M = 4,
 N = 3,
 RECH1 = 0.1005E+03, -0.685E+01, 0.0, 0.1055E+03, -0.12F+02, 0.252E+00, -0.351F+02, -0.45F+01, 0.0,
 0.588E+01, -0.131E+01, 0.0, 0.588F+01, -0.131E+01, 0.0, -0.7374E+02, -0.76F+01, 0.0,
 0.272E+01, -0.786E+00, 0.0, 0.272F+01, -0.786E+00, 0.0, -0.149F+02, -0.13E+00, 0.0,
 0.555E+01, -0.81F+00, 0.0, 0.555E+01, -0.81E+00, 0.0, -0.154E+02, -0.702F+00, 0.0,
 RECH2 = 0.783E+02, -0.48E+01, 0.0, 0.797E+02, -0.816F+01, 0.141E+00, -0.3561F+02, -0.48F+01, 0.0,
 TORO1 = 0.9E+02, 0.3E+03, 0.0, 0.25E+03, 0.0, 0.28F+04, 0.0, 0.28E+04,
 TORO2 = 0.22E+03, 0.7E+03,
 RVFST = 0.75E-03,
 RRR = 0.425E-02, 0.42F-02,
 ZMOD = 0.0, 0.3E+01,
 UPFRE0 = 0.62E+01, 0.82E+01,
 RPP = 0.16E+02, 0.4E+01, 0.1E+01,
 RT1 = 0.32E-02, 0.85E-02,
 RT2 = 0.0, 0.21E-03, 0.85E-02, 0.24E-01,
 AT1 = 0.115E+01, 0.12F+01,
 AT2 = 0.0, 0.906E+00, 0.12E+01, 0.122E+01,
 KATTN1 = 0.46E-01, 0.7E-01,
 KATTN2 = 0.5E-02, 0.16E-01, 0.7E-01, 0.16F+00,
 T71 = 0.32E+01, 0.35E+01,
 T72 = 0.18E+01, 0.26E+01, 0.35E+01, 0.48E+01,
 XLT = 0.0,
 XLT1 = 0.0,
 XLT2 = 0.0,
 XLT3 = -0.1E+01,
 XLT4 = -0.1E+00,
 XPT1 = -0.1E+00.

FIG. 4.1 TVBS INPUT DATA (Cont)

XPT2 = -0.1E+00.
 RPT1 = -0.1E+00.
 RPT2 = -0.4E+01.
 P1 = 0.0.
 P2 = -0.3E+01.
 COSNT = 0.46E+02, 0.3E+01, 0.0, 0.0.
 TION = 0.29E+03.
 TATM = 0.29E+03.
 TLOUD = 0.273E+03.
 TRAIN = 0.273E+03.
 TTL = 0.29E+03.
 AMMT = 0.48E+01, 0.8E-03, 0.166E+01, 0.0, 0.45E-04, 0.166E+01, 0.0, 0.23E-05, 0.166E+01, 0.0, 0.8E-06,
 0.166E+01.
 TE = 0.29E+03.
 TRCVRS = 0.17E+03.
 GF1 = 0.24E+02.
 GF2 = 0.44E+02.
 GS1 = 0.22E+02.
 GS2 = 0.34E+02.
 GW1 = 0.18E+02.
 GW2 = 0.44E+02.
 GSX = 0.114E+01, 0.0, 0.0, -0.267E+01, 0.173E+00, 0.0, 0.322E+01, 0.0, 0.0.
 GFY = -0.214E+01, 0.0, 0.0, 0.157E+02, -0.743E+00, 0.0, -0.17E+02, 0.0, 0.0.
 GWX = -0.38E+01, 0.0, 0.0, 0.5E+00, -0.242E+00, 0.0, -0.101E+02, 0.0, 0.0.
 AMTPLX = 0.935E+00, 0.98E+00, 0.95E+00, 0.89E+00.
 RMTPLX = -0.6E-01, -0.22E-01, -0.35E-01, -0.63E-01.
 D3RD = 0.25E+02.
 ARCVR = 0.275E-02, 0.975E-02, 0.297E-02, 0.372E-01.
 EXPR = -0.146E+01.
 FOLINE = 0.1E+01.
 ETAPT = 0.95E+00.
 ETAPC = 0.7E+00.
 ETAB = 0.83E+00.

FIG. 4.1 TVBS INPUT DATA (Cont)

ETARCU = 0.98E+00,
 ETALCF = 0.95E+00,
 RHOPC = 0.526E-01,
 DGRD = 0.71E+00, -0.3E-01,
 WPTR = 0.0, 0.4E-02, 0.0,
 WCU1 = 0.36E+03,
 WCU2 = 0.26E+05,
 WCU = 0.623E+01, 0.0, 0.0, -0.106E+02, 0.658E+00, 0.0, 0.3E+00, 0.41E+00, 0.0,
 WPCN = -0.3E+01, 0.77E+00, 0.0,
 WLCFP1 = 0.3E+03,
 WLCFP = 0.161E+00, 0.953E+00, 0.514E+00, 0.75E+00,
 VPTR1 = 0.6E+04,
 VPTR = -0.1754E+02, 0.587E+00, 0.0, -0.295E+02, 0.905E+00, 0.0,
 VCU1 = 0.6E+04,
 VCU = -0.2054E+02, 0.587E+00, 0.0, -0.325E+02, 0.905E+00, 0.0,
 VPCN1 = 0.4E+04,
 VPCN = -0.1444E+02, 0.54E+00, 0.0, -0.2516E+02, 0.835E+00, 0.0,
 VLCFP1 = 0.35E+03,
 VLCFP = 0.247E-02, 0.869E+00, 0.25E-03, 0.1098E+01,
 CENGP1 = 0.16E+04,
 CFNGP2 = 0.8E+04,
 CENGPC = -0.3472E+02, 0.824E+00, 0.0, -0.2234E+02, 0.438E+00, 0.0, -0.523E+01, 0.0, 0.0,
 CMFGP1 = 0.12E+04,
 CMFGPC = -0.5254E+02, 0.1326E+01, 0.0, -0.2866E+02, 0.552E+00, 0.0,
 CLC1 = 0.1E+04,
 CLC2 = 0.2E+04,
 CMFGLC = 0.25E-02, 0.0, 0.0, 0.25E-05, 0.263E-02, 0.1185E-05,
 CENGLC = 0.125E-02, 0.0, 0.0, 0.125E-05, 0.0, 0.125E-05,
 TSTRCT = 0.25E+00,
 LTR = 0.2E+01,
 LRCVR = 0.2E+01,

FIG. 4.1 TVBS INPUT DATA (Cont)

```

LPT      = 0.2E+01,
LRATT    = 0.5E+01, 0.2E+01, 0.1E+01,
LPC      = 0.2E+01,
LPCU     = 0.1E+02,
LLCF     = 0.1E+02,
RRWTR    = 0.1E+01,
RRWRVR   = 0.1E+01,
RRWPT    = 0.1E+01,
RRWATT   = 0.1E+01,
RRWPC    = 0.1E+01,
RRWPCU   = 0.1E+01,
RRWLCF   = 0.1E+01,
RRVTR    = 0.1E+01,
RRVRVR   = 0.1E+01,
RRVPT    = 0.1E+01,
RRVATT   = 0.1E+01,
RRVPC    = 0.1E+01,
RRVPCU   = 0.1E+01,
RRVLCF   = 0.1E+01,
RRCTR    = 0.1E+01,
RRCRVR   = 0.1E+01,
RRCPPT   = 0.1E+01,
RRCRATT  = 0.1E+01,
RRCPPC   = 0.1E+01,
RRCPCU   = 0.1E+01,
RRCLCF   = 0.1E+01,
VRHO     = 0.1E+01,
POP1     = 0.1E+03,
POP2     = 0.4E+04,
POP      = 0.5E+02, 0.1E+04, 0.7E+04, 0.25E+05,
IMD      = 6,
IDA      = 13,
IYR      = 69,
WMUX     = 0.188E+02, 0.787E-01, 0.9F+01, 0.242F+02, 0.77E-01, 0.115E+02, 0.128E+01, 0.797E-01, 0.35E+00,
WHARR    = 0.0, 0.3E-02, 0.0, 0.0, 0.3E-02, 0.0, 0.0, 0.3E-02, 0.0, 0.0, 0.1E-01, 0.0, 0.0, 0.15E-01, 0.0, 0.0, 0.1E-01, 0.0,
NPROTO   = 0.1E+01,
SKLS1    = -0.117E+02, 0.181E+00, -0.2478E+02, 0.181E+00, -0.807E+01, 0.0, -0.807E+01, 0.0, 0.0, 0.0, 0.0,
SKLS2    = -0.5E-01, 0.0, 0.518E+01, 0.0, 0.22E+00, 0.365E+00, 0.22E+00, 0.365E+00, 0.0, 0.0, 0.0, 0.0, 0.0,
ATT1     = 0.401E-01, 0.0, -0.196E+01, 0.257F-01, 0.219E+00, 0.0, 0.219E+00, 0.0, 0.0, 0.0, 0.0,
ATT2     = 0.11E+00, 0.0, 0.11E+00, 0.0, 0.17E+00, 0.0, 0.17E+00, 0.0, 0.0, 0.0, 0.0, 0.0,
ATLS1    = -0.836E+01, 0.0, 0.2931E+02, 0.0, -0.1062E+02, 0.0, -0.1062E+02, 0.0, 0.0, 0.0, 0.0, 0.0,
ATLS2    = 0.175E+01, 0.528E-01, 0.175E+01, 0.528E-01, -0.11E+00, 0.116F+00, -0.11E+00, 0.116F+00,
SEND

```

6-4
P\$NAMEP1

XLIFE = 2., 5., 10., 3., 4., 6.,
PGMIN = 1., PGMAX = 1.E+05, D1MIN = .5, D1MAX = 100.,
D4MIN = 1., D4MAX = 100.,
XLMIN = 1., XLMAX = 10.,
PTRMIN = 1., PTRMX1 = 1.0E+05, PTRMX2 = 1.0E+04, PTRMX3 = 2.0E+03,
DFMREC = .000625,
PARANT(1) = 1., 1.05, 1.10, 1.15,
BWUPL = 17.5,
ACQLV = 4*211.1, 257.57, 3*211.1,
ENGLV = 4*36.7, 14.6, 3*36.7,
ACQMP = 3*0., 21.08, 0., 21.17, 2*0.,
ENGMP = 3*0., 60.6, 0., 66.63, 2*0.,
URCBSM = 1.72, URCMDA = 1.63, ENGBSM = 15.4, ENGMDA = 11.37,
VSABD = 170.,
TRMIN = 150., 760.,
TRMAX = 1100., 2200.,
RCVR1 = 0., 0., RCVR2 = 865.,
HRCVR1 = -24.41, -1.014, .0108, -24.41, -1.014, .0108,
-26.46, -.884, .0106, -26.46, -.884, .0106,
HRCVR2 = 27.52, -2.27, 0., -28.90, -.349, 0.,
XLC1 = 8.25, -.371, .00322, 6.34, -.279, .00227,
XLC2 = 7.38, -.322, .00255, XLC3 = 5.52, -.232, .00160,
RCVC2 = 9.7E-06, RCVC3 = 6.34E-06,

\$END

FIG. 4.1 TVBS INPUT DATA (Cont)

```

SNAME2
NR      = 6,
NCR     = 2,
NF      = 7,
NRS     = 0,
NRLV    = 0,
NRTSTA  = 0,
NC      = 0,
LSYS    = 0.1E+02,
LSAT    = 0.5E+01,
NTRF    = 1,
NCF     = 0.1E+01,
NCTRF   = 0,
NARQD   = 0.6E+01,
ICNFIG  = 0, 2, 0, 0, 0, 0, 0, 0,
IMOD1   = 0, 2,
IMOD2   = 0, 2,
IFRAND  = 0, 2, 0, 0,
IXMTR   = 0, 0, 0, 4,
IRAD    = 0, 2,
ISTK    = 0, 2, 0,
IATT    = 0, 2, 0,
IRATT   = 1, 0, 0,
IPTYPE  = 1, 0,
XLAT    = 0.35E+02,
XLONG   = 0.25E+02,
AMAJR   = 0.16E+04,
BMINR   = 0.1E+04,
NAPV    = 0.1E+01,
I11     = 1,
I19     = 2,
I21     = 1,
I22     = 1,
RMO     = 0.1E+00,
TRW     = 0,
NSAT    = 0.1E+01,
ITASOI  = 0, 0, 3, 0, 0, 0, 0,
SNRGI   = 0.445E+02, 0.335E+02, 0.27E+02, 0.23E+02, 0.17E+02, 0.1E+02, 0.53E+02,
IPDNSE  = 0, 2, 0, 0,
INBR    = 1, 0, 0, 0, 0, 0, 0, 0,
XNGRI   = 0.1E+08, 0.2E+05, 0.4E+05, 0.1E+06, 0.2E+06, 0.4E+06, 0.1E+07, 0.2E+07,
TITLE   = 160H  UNITED STATES NATIONAL FTV / OPERATIONAL DIRECT SYSTEM
ICASE   = 0,
X0      = 0.0, 0.0, 0.0, 0.10718E+02, 0.1E+02, 0.29E+03, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
JIT     = 2,
SETDMS  = 0.0,
SFND

```

FIG. 4.2 TVBS OUTPUT SUMMARY

CASE NO. 1

UNITED STATES NATIONAL ETV / OPERATIONAL DIRECT SYSTEM

6/13/69

USER REQUIREMENTS		LINK PARAMETERS		
		PARAMETER	UPLINK	DOWNLINK
AREA OF COVERAGE		FREQUENCY (GHZ)	8.20	2.50
MAJOR AXIS (MI)	1600.000	FRP (DBW)	46.3	62.6
MINOR AXIS (MI)	1000.000	RF BANDWIDTH (MHZ)	33.6	33.6
AOC (MILLIONS OF SQUARE MILES)	1.257	MODULATION INDEX	3.0	3.0
RECEIVER POP/BEAM (THOUSANDS)	10000.000	ANTENNA BEAMWIDTH (DEG)	.7	2.4
TOTAL RECEIVER POP (THOUSANDS)	60000.000			
LATITUDE (DEGREES)	35.000	LOSS (DB)		
RELATIVE LONGITUDE (DEGREES)	25.000	FREESPACE	-201.8	-191.5
TASO GRADE	3	ATMOSPHERIC	-.5	-.1
SIGNAL TO NOISE RATIO (DB)	27.000	XMIT CIRCUIT	-2.0	-1.3
NUMBER OF UPLINK FAC/SAT SYSTEM	7	RCVR CIRCUIT	0.0	-.1
NUMBER OF CHANNELS/FACILITY	1	XMIT POINTING	-.1	-.1
UPLINK MODULATION	FM	RCVR POINTING	-.1	-4.0
DOWNLINK MODULATION	FM	POLARIZATION	0.0	-3.0
TOTAL SYSTEM OPERATION LIFE (YRS)	10			
SATELLITE LIFE (YRS)	5	NOISE TEMPERATURES (DEG K)		
DUTY CYCLE (HRS/DAY)	24.0	SKY	-	11.4
NUMBER OF VIDEO CHANNELS/BEAM	2	ENVIRONMENTAL	-	4.3
NUMBER OF AUDIO/VIDEO CHANNELS	1	EARTH	290.0	290.0
NOISE ENVIRONMENT	SURR	RECEIVER	170.0	603.1
NUMBER OF BEAMS/SYSTEM	6	TRANSMISSION LINE	0.0	6.6
NUMBER OF SAT SYSTEMS	1			
GROUND DOWNLINK SYSTEM	DIRECT			

MINIMUM COST SOLUTION

EXCLUDING LAUNCH VEHICLE COST
INCLUDING LAUNCH VEHICLE COST

TOTAL COST/YR
(MILLIONS)
309.257
348.592

COST/YR/RCVR
(DOLLARS)
5.15
5.81

OPTIMUM DESIGN PARAMETERS - CONCEPT R

PARAMETER	PTH-SAT (WATTS)	WT-SAT (LBS)	VOL-EM (CU FT)	VOL-SAT (CU FT)	PTR-GND (WATTS)	VOL-SA (CU FT)	DA-G UP (FT)	DA-G DN (FT)	TR-GND (DEG K)
DESIGN VALUE	548.3	2603.6	45.6	262.0	3.7	117.1	7.2	1.6	603.1
ON BOUNDARY	NO	NO	NO	YES	NO	NO	NO	NO	NO
UPPER BOUND	100000.0	5200.0	123.2	262.0	100000.0	262.0	100.0	100.0	50000.0
LOWER BOUND	1.0	0.0	0.0	0.0	1.0	0.0	.5	1.0	50.0

FIG. 4.2 TVBS OUTPUT SUMMARY (Cont)

CASE NO. 1	UNITED STATES NATIONAL ETV / OPERATIONAL DIRECT SYSTEM						6/13/69	
SPACECRAFT CONCEPT B		COST (MILLIONS)		TOTAL COST	WEIGHT (LBS.)	VOLUME (CU. FT.)	TYPE	DESCRIPTIVE PARAMETERS
		ACQUISITION COST	R/D COST					
POWER SUBSYSTEM								
POWER SOURCE	2.199	3.897	8.296	701.22	117.10	SOL ARRAY	PRIME PWR = 4832 WATTS	
BATTERIES	.004	.134	.141	506.30	3.40	NICKEL CAD	WATT HRS = 10353.4	
CONDITIONING	.898	3.186	4.981	675.83	13.99			
ANTENNA (DOWN-LINK)	.374	1.342	2.090	32.47	91.76	TRUSS	GAIN = 36.6 DB, DIA = 11.4 FT	
ANTENNA (UP-LINK)	0.000	0.000	0.000	0.00	0.00	TRUSS	GAIN = 46.7 DB, DIA = 11.4 FT	
TRANSMITTER	.511	2.284	3.306	130.90	5.60	TWT	PWR/CHAN = 54R, EFF = .589	
MULTIPLEXER	.898	2.477	4.273	34.50	.10	WAVEGUIDE		
RECEIVER	.134	.492	.760	15.00	.63	LIN TRANS	NOISE FIG = 2.0	
STRUCTURE	.260	2.805	3.325	182.28	18.18	CONV		
THERMAL CONTROL	.068	.754	.890	37.81	7.56	HYBRID	PWR RAD = 3361, ARFA = 30.2	
STATION KEEPING	.559	7.645	8.764	75.85	.26	ION PROP		
ATTITUDE CONTROL	2.627	28.360	33.614	121.45	1.42	ION PROP		
TLM AND CMD	1.051	1.575	3.676	70.00	2.00			
TEST EQUIPMENT	-	.750	.750	-	-			
	-----	-----	-----	-----	-----			
TOTAL/SAT	9.583	55.700	74.865	2603.61	262.00			
USER SYSTEM TOTAL	114.990	55.700	170.690					
GROUND UP-LINK SYSTEM		COST (MILLIONS)		TOTAL COST	DESCRIPTIVE PARAMETERS			
		ACQUISITION COST	R/D COST					
FACILITIES	.048	.005	.001	.062	BROADCAST = COLOR PWR/CHAN = 3.67 GAIN = 42.6 DB, DIA = 7.2 FT			
PROG. REC. + CONTR EQUIP	.144	0.000	.113	1.274				
TRANSMITTER	.065	0.000	.008	.147				
ANTENNA	.006	0.000	.000	.009				
STANDBY POWER	.040	-	.002	.057				
TEST EQUIPMENT	.200	-	-	.200				
SYST DES. INTEG. + MGMT	-	.127	-	.127				
SYST OPS MGMT	-	-	.121	1.210				
	-----	-----	-----	-----				
TOTAL/FAC	.503	.132	.245	3.087				
TLM AND CMD EQUIP	.290	.195	1.065	11.138				
SYSTEM TOTAL	3.812	.327	2.782	31.955				
GROUND DOWN-LINK SYSTEM		UNIT ACQ COST (DOLLARS)	UNIT INST COST (DOLLARS)	UNIT MAINT COST (DOLLARS/YR)	TOTAL COST (MILLIONS)	TYPE	DESCRIPTIVE PARAMETERS	
ANTENNA	7.15	24.68	.72	389.805	PARABOLIC	GAIN = 19.0 DB, EFFECTIVE DIA = 1.6 FT NOISE FIG = 4.9, NOISE TEMP = 603.1		
RECEIVER	3.06	3.06	.31	91.874				
	-----	-----	-----	-----				
TOTAL/BEAM	10.21	27.74	1.02	481.679				
SYSTEM TOTAL				2890.073				
LAUNCH VEHICLE		COST	WEIGHT LIMIT	VOLUME LIMIT	COMMENTS			
SATURN V (UNMANNED)	458.900	42000.000	6000.000		INCLUDES RELAUNCHES AND R/D PROHATED = 7 SATELLITE CONCEPT			
USER TOTAL	393.343	5200.000	262.000					

5

5.0 SUBROUTINE DESCRIPTION

5.1 TVBS DRIVER

The function of the TVBS DRIVER is to

- a) read the input lists,
- b) initialize parameters,
- c) set the flags describing the cases to be run, and
- d) control the execution of the program.

Input of data is performed through the use of the namelist feature which eases the task of preparing data cards by the program user. The input data is comprised of three separate lists identified by NAME1, NAMEP1, and NAME2, respectively. NAME1 and NAMEP1 consist mainly of fixed data which is not liable to change during the variation of user requirements. It contains such parameters as the coefficients for the weight, volume, and cost curves of the system elements, losses, and noise temperatures. NAME2 is used to input quantities which describe the specific user requirements. A more detailed description of the elements in the lists is given in Section 3.

The driver also initializes certain parameters not initialized in the other subroutines, INIT and SYSEQ.

The TVBS driver converts the input values of the ionospheric and transmission line losses (XLI, XLT1, XLT2, XLT3, XLT4, respectively) from db to dimensionless units.

It also calculates the redundancy factors to be used in the system equations. (See Section 2.7). It sets the precipitation rate as a function of latitude and calculates beamwidth (BMWDTH) and area of coverage (AOC) as determined by the major and minor axes (AMAJR, BMINR) specified by the user requirements. The TVBS driver also computes the FM improvement factor:

$$I_m = 3 M^2 (1 + M) \text{ where } M = 3;$$

$$I_m = 1 \text{ for AM.}$$

The main function of the driver, however, is to set the descriptive flags for the desired configuration. This is done through the use of a series of do-loops which iterate over the various types of system elements and other relevant parameters. The elements which are variable in the do-loop system consist of the following:

- a) Taso grade - 0, 1, 2, ..., 6
- b) Population density - p_1, p_2, p_3, p_4 (input)
- c) Receiver population - $r_1, r_2, \dots, r_8$ (input)
- d) Concept types - A, B, ..., H *
- e) Uplink modulation - AM and FM
- f) Downlink modulation - AM and FM (currently constrained to be the same as the uplink)
- g) Frequency band operating point - f_1, f_2, f_3, f_4 (input)
- h) Power supply type - solar array and nuclear reactor **
- i) Downlink transmitter type - grid, solid state, CFA, TWT
- j) Thermal control system - passive and hybrid
- k) Station-keeping system - ammonia resistojet, ion propulsion, H_2O electrolysis
- l) Attitude control system - ammonia resistojet, ion propulsion, H_2O electrolysis
- m) Battery type - nickel cadmium, silver cadmium, silver zinc.

By executing these loops, the various combinations can be considered. Vector arrays have been provided which permit the user to bypass any specified value of a given index parameter. These arrays are dimensioned according to the maximum value the index parameters can have. Each time the index parameter is updated, it is checked against the corresponding component of the vector. If the value of the index parameter is not equal to the value of that component, that value of the index parameter is by-passed. For example, assume that it is desired that both AM and FM are to be considered

* Concept G has not been implemented.

** Nuclear reactor has not been implemented.

for downlink modulation. Let the associated 2x1 vector be IMOD2. Then

$$\text{IMOD2} = (1,2) .$$

If only AM is to be considered,

$$\text{IMOD2} = (1,0) ,$$

and if only FM is to be considered,

$$\text{IMOD2} = (0,2) .$$

Note that for a case to be considered, the value of the index parameter must be equal to its associated component value.

The index parameters used in the do-loop system and the flags used to describe the cases being run are given below with their maximum values.

J1 Taso grade (7)

A vector of ground signal-to-noise ratios (SNGRI) is input and is used to set the signal to noise ratio as a function of desired taso grade.

J2 Population density (4)

An associated vector of population densities (POP) is input and set during the execution of the do-loop system.

J3 Receiver population (8)

A vector of receiver populations (per beam) to be considered (XNGRI) is input and set as a function of the value of J3.

I1 Concept type (8)

1 - A	5 - E
2 - B	6 - F
3 - C	7 - G
4 - D	8 - H

I2 Downlink modulation (2)

1 - AM
2 - FM

I3 Frequency band operating point (4)

1 - FREQ(1)
2 - FREQ(2)
3 - FREQ(3)
4 - FREQ(4)

FREQ is input in GHZ

- I4 Downlink Antenna type *(2)
 - 1 - TRUSS
 - 2 - RIGID
- I5 Downlink feed boom *(3)
 - 1 - folding
 - 2 - expandable
 - 3 - rigid
- I6 Power supply type (2)
 - 1 - solar array
 - 2 -- nuclear reactor **
- I7 Transmitter type (spacecraft) (4)
 - 1 - grid tube
 - 2 - solid state
 - 3 - CFA
 - 4 - TWT
- I8 Receiver type (spacecraft) * (2)
 - 1 - linear translator
 - 2 - modulation converter **
- I9 Multiplexer type * (3)
 - 1 - coax
 - 2 - waveguide
 - 3 - stripline
- II0 Uplink modulation (2)
 - 1 - AM
 - 2 - FM
- II1 Ground uplink antenna * (1)
 - 1 - minimum motion antenna
- II2 Structure * (2)
 - 1 - conventional
 - 2 - BSM

* flag not used as index parameter in do-loop system
** not implemented.

- I13 Thermal control system (2)
 - 1 - passive
 - 2 - hybrid
- I14 Station keeping system (3)
 - 1 - ammonia resistojet
 - 2 - ion propulsion
 - 3 - H₂O electrolysis
- I15 Attitude control system (3)
 - 1 - ammonia resistojet
 - 2 - ion propulsion
 - 3 - H₂O electrolysis
- I16 Ground downlink antenna type * (2)
 - 1 - wideband
 - 2 - parabolic
- I17 (not used)
- I18 Launch vehicle type * (2)
 - 1 - Saturn V, unmanned
 - 2 - Saturn V, manned
- I19 Type of broadcast * (2)
 - 1 - monochrome
 - 2 - color
- I20 Battery type (3)
 - 1 - nickel cadmium
 - 2 - silver cadmium
 - 3 - silver zinc
- I21 Type of downlink system (3)
 - 1 - direct to home
 - 2 - retransmission **
 - 3 - CATV
- I22 (not used)
- I23 Uplink antenna (satellite) (2)
 - 1 - truss
 - 2 - rigid

** not implemented

I24 Uplink frequency operating point (2)

1 - UPFREQ (1)

2 - UPFREQ (2)

UPFREQ is input in GHZ.

I25 (not used)

During the execution of the do-loops, certain operations are performed. These are enumerated in the accompanying flow diagram (see Figure 5.1).

They include:

- a) calculating required bandwidth
- b) setting multiplexer type as function of frequency band:

UHF - coax

S-band - waveguide

X-band - stripline

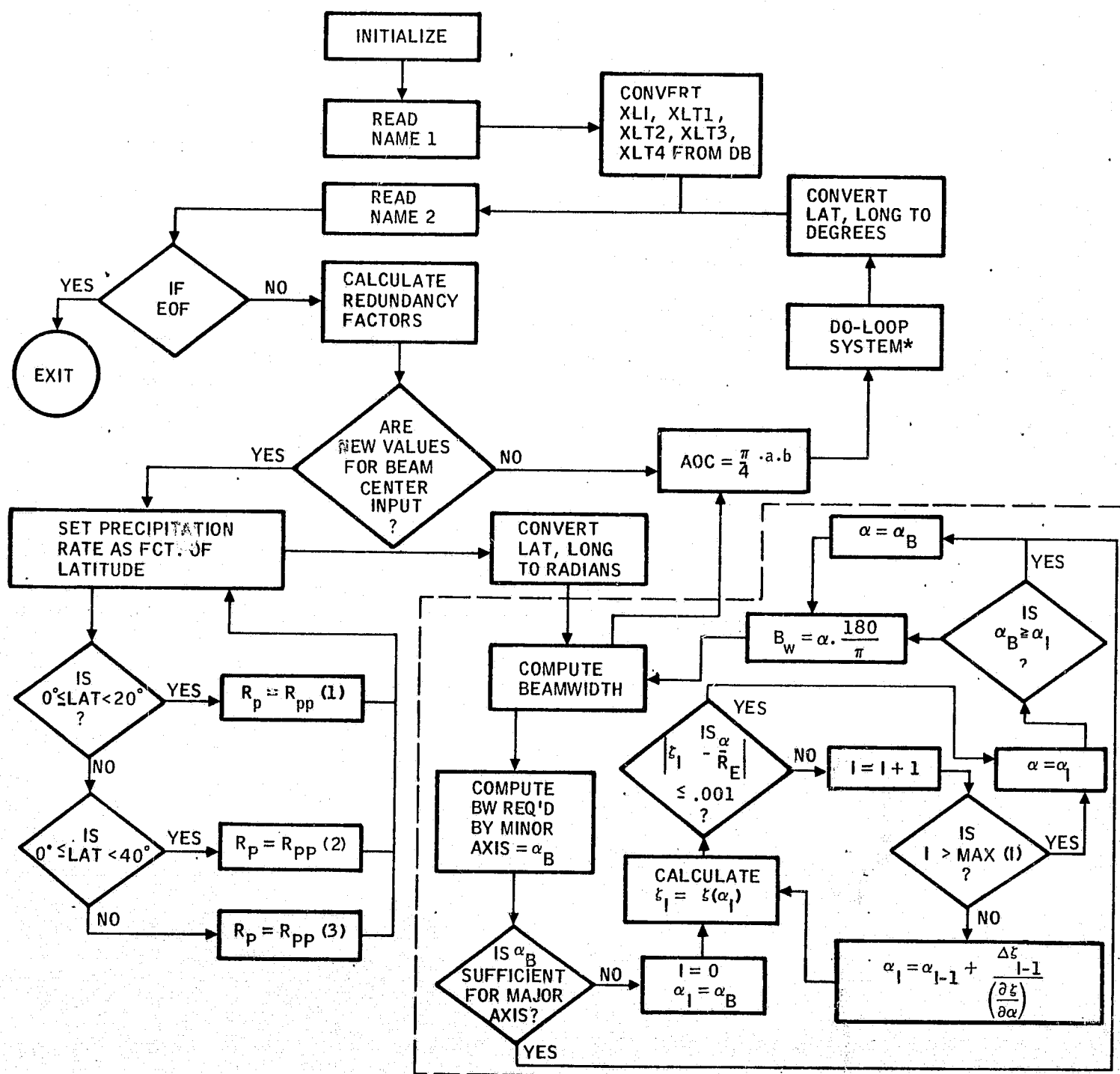
- c) calculating spacecraft uplink antenna diameter D_2 and downlink antenna diameter D_3 .

$$D_i = \frac{70 \cdot C}{BW_i \cdot f_i} \quad i = 2, 3 \quad \text{where}$$

C is the velocity of light (ft/sec)

BW - beamwidth (degree)

f - frequency (kc)



* See Figure 5.2

Figure 5.1 Flow Diagram - TVBS Driver

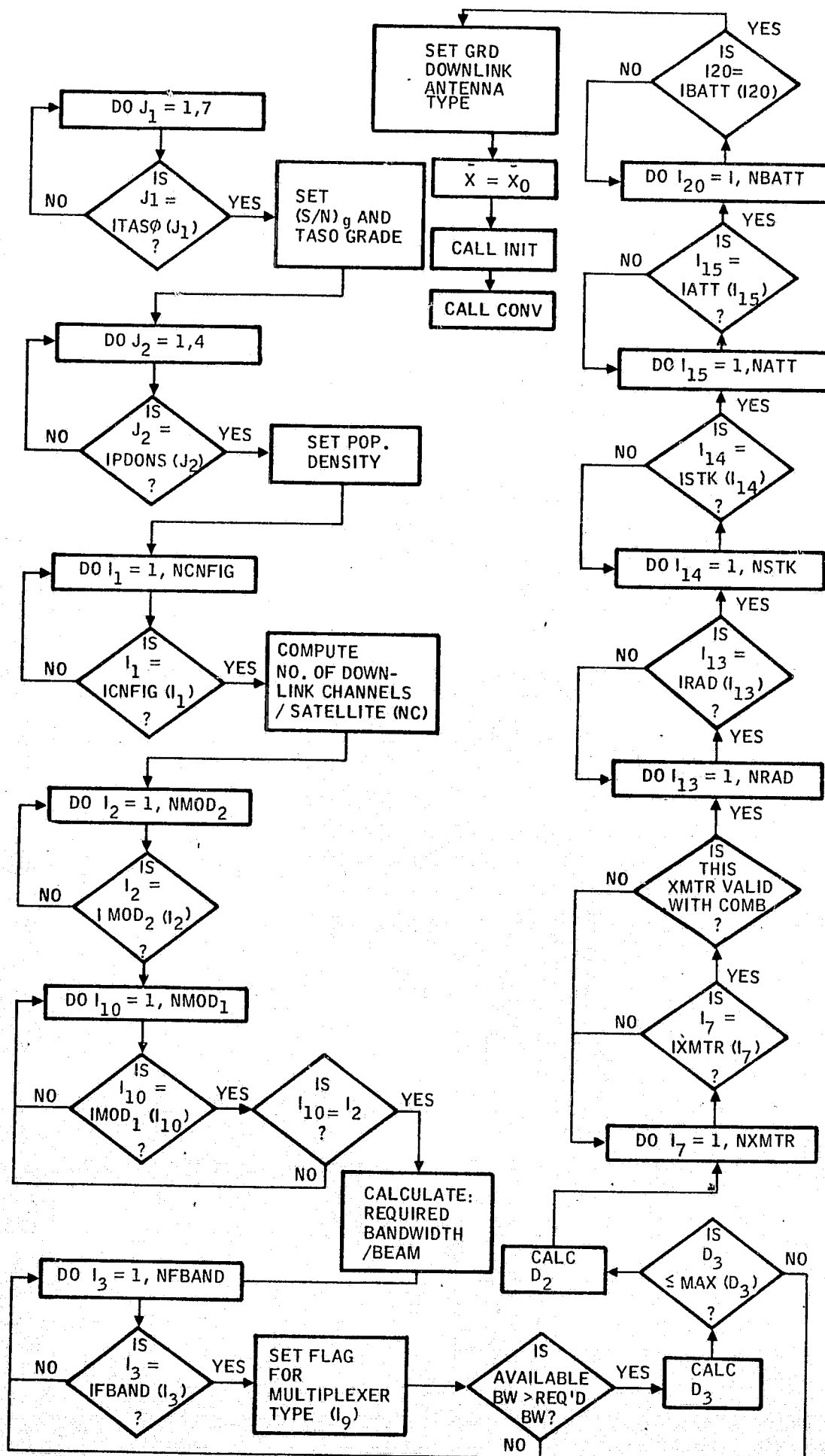


Figure 5.2 TVBS Do-loop System

```

PROGRAM TVBS(INPUT,TAPE5=INPUT,OUTPUT,TAPE6=OUTPUT)
COMMON/ATTN/ BVEST,BBB(2),ZMOD(2),XLAT,XLONG,RP,BT1(2),BT2(4)
* ,AT1(2),AT2(4),KATTN1(2),KATTN2(4),TZ1(2),TZ2(4),XLI,XLT1,XLT2
* ,XLT3,XLT4,XPT1,XPT2,RPT1,RPT2,P1,P2,COSNT(4),TION,TATM,TCLOUD
* ,TRAIN,TTL,AMMT(3,4),TE,TRCVRS,GE1,GE2,GS1,GS2,GM1,GM2,GSX(3,3)
* ,GEX(3,3),GMX(3,3),AMTPLX(4),BMTPLX(4),TCO,TI,TA,TC,TM,TR,TT
* ,TEFF,FSL12,FSL34,ATL1,ATL2,XCIRC1,XCIRC2,RCIRC1,RCIRC2
* ,XL12,XL34,KGS,KGM,KGE,RPP(3),BETA,GAMMA,EL,RSL
COMMON/BOUND/IARRAY(15),JARRAY(15),YMIN(15),YMAX(15)
COMMON/CNST/ PI,R,CVEL,BOLZ,RSAT,RE
COMMON/CNVRGE/ EPSLON,MCONV,ITMAX,PSIK,GRAD(5)
COMMON/DATE/ IMO,IDA,IYR,POP1,POP2,ICASE, TITLE(6)
COMMON /DIAM / D(4),G(4)
COMMON/MAIN/ I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15
* ,I16,I17,I18,I19,I20,I21,I22,I23,I24,I25
* ,XQ,XJ,XK,XL,XP1,XP2,XP3,XP4
COMMON/MISC/ BLEM,BVEM,XLF1,XLF2,XLF,XLA,SNRG,XTA,XIMPRV(2),BVID12
* ,BVID34,TSTRCT,AMAJR,BMINR
COMMON/NUMB/ NTR,NCTR,NF,LSYS,NSL,NRS,NLV,NRLV,NRSTA,NB,NCB,NC
* ,NCTRF,NTRF,NCF,LSAT,FREQ1,GR85,GR89,GR95,GR,XNGR,XNGRP
* ,NAPV,NAUDC,PDNSE,ITASO,NBRQD,AOC,NSAT,NBS,SETDNS,JIT
COMMON/PARA/ PLOG,ETA,ARD,VTR,WTR,FREQ4,XNS,RHO,AKP,XNG
* ,CE,CMFG,WEM,WLOG,PGLOG,CSTPYR,G4LOG,TLOG
* ,VEM,WSTLOG,PSLOG,WHLOG,PRLOG,D1LOG
* ,D4LOG,PTALOG,PCULOG,PPCLOG,PPTLOG,CEPP,CMFGPP
COMMON/PARA3/ YM,YL,AML,AREA,WHRS,WS,XLEM
COMMON/PARBD/PGMIN,PGMAX,D1MIN,D1MAX,D4MIN,D4MAX,TRMIN(2),TRMAX(2)
* ,XLMIN,XLMAX,XNGRMX,VSABD,PTRMIN,PTRMX1,PTRMX2
* ,PTRMX3
COMMON/REDUND/ LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF,RRWTR,RRWRVR
* ,RRWPT,RRWBTT,RRWPC,RRWPCU,RRWLCF,RRVTR,RRVRVR,RRVPT,RRVBTT
* ,RRVPC,RRVPCU,RRVLCF,RRCTR,RRCRVR,RRCPTR,RRCBTT,RRPCPC,RRPCPCU
* ,RRCLCF,VRHO,WRTR,WRCVR,WRPT,WRBTT,WRPC,WRPCU,WRLCF,VRTR,VRCVR
* ,VRPT,VRBTT,VRPC,VRPCU,VRLCF,CRTR,CRCVR,CRPT,CRBTT,CRPC,CRPCU
* ,CRLCF
COMMON/SYS/ WMP,VMP,WGHTB(3,3),VOLB(3,3),XTR(3),ANTC(3),CSTR(3,3)
* ,CP1(3,2),CP2(3,2),RAD1(3,2),RAD2(3,2),SK1(2,2,3),SK2(2,2,3)
* ,PLX1(4,3),PLX2(4,3),DNLA,ANTD1(3,4),ANTD2(3,4),CBT1(3,3)
* ,CBT2(3,3),RECH(3),RECH1(3,3,4),TRBD1(2,4),RECH2(3,3),TRBD2(2)
* ,RECA1,RECA2,ANTH(3,4),XINST1,XINST2,CINST(3,4),CRCUR(3,2)
* ,CRCNR(3,2),NOANT(8),DCYCLE,ARCV(4),ETAPT,ETAPC,ETAB,ETAPCU
* ,ETALCF,RHOPC,DGRD(2),WPTR(3) ,WCU1,WCU2,WCU(3,3),WPCN(3)
* ,WLCFP1,WLCFP(2,2),VPTR1,VPTR(3,2),VCU1,VCU(3,2),VPCN1
* ,VPCN(3,2),VLCFP1,VLCFP(2,2),CENGP1,CENGP2,CMFGP1,CENGPC(3,3)
* ,CMFGPC(3,2),CLC1,CLC2,CENGLC(2,3),CMFGLC(2,3),XCNST
* ,DEGRAD,CPTUR(3),CPTNR(3),CPCUR(3),CPCNR(3),CPCUR(3),CPCNR(3)
* ,CBETA,SBETA,IETBD1,IETBD2,WMUX(3,3),WHARR(3,8),CHARUR(3)
* ,CHARNR(3),NPROTO,SKLS1(2,2,3),SKLS2(2,2,3),ATT1(2,2,3)
* ,ATT2(2,2,3),ATLS1(2,2,3),ATLS2(2,2,3),IAML,IWS,AMLLOG
* ,DFMREC,PARANT(4),NRGR,ACQLV(8),ENGLV(8),ACQMP(8),ENGMP(8)
* ,URCBSM,ENGBSM,URCMDA,ENGMDA
* ,RCVR1(2),RCVR2(1),HRCVR1(3,4),HRCVR2(3,2),XLC1(3,2),XLC2(3,1)
* ,XLC3(3,1),RCVC2,RCVC3,DELC1,DELC2
COMMON/SYSTEM/ AW(3),AV(3),AET(3),ACE(3),ACMFG(3),AWP(3),AVP(3)
* ,AETP(3),ACEP(3),ACMFGP(3),IVLEV(4),ICLEV(4)

```

```

COMMON/TABLE/ AWGHT(3,4,3),AVOL1(3,4,3),AVOL2(3,4,3),ACENG1(3,4,3)
*      ,ACENG2(3,4,3),ACENG3(3,4,3),ACMFG1(3,4,3)
*      ,ACMFG2(3,4,3),AETA(3,4,3),FWGHT(3,4,3),FVOL1(3,4,3)
*      ,FVOL2(3,4,3),FCENG1(3,4,3),FCENG2(3,4,3)
*      ,FCENG3(3,4,3),FCMFG1(3,4,3),FCMFG2(3,4,3)
*      ,FETA(3,4,3),PSPPLY(4,2),ANT(4,2)
*      ,FBOOM(4,3),BCHANN(2),DABD(8),BLNGTH(8)
*      ,BVOL(8),BWGHT(8),DEM(8),FREQ(4),BDWDTH(4)
*      ,UPFREQ(2),EXPR,FDLIN
REAL NREC1,NREC2,NREC3,NREC41,NREC42,NREC43,K1,K2,NOANT
REAL KATTN1,KATTN2,LSYS,LSAT,NBRQD,NLV,NTR,NAPV,NAUDC,NCF
REAL LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF
REAL NSAT,NBS,NPROTO
REAL NRGR
DIMENSION XO(3,8),X(7),LBATT(3),POP(4)
*      ,ITASOI(7),SNRGI(7),IPDNSE(4),INGR(8),XNGRI(8)
DIMENSION ICNFIG(8),IMOD1(2),IMOD2(2),IFBAND(4),IXMTR(4),IRAD(2)
*      ,ISTK(3),IATT(3),IBATT(3),IPTYPE(2),IMTPLX(2)
DIMENSION XLIFE(6),ILIFE(6)
NAMELIST/NAME1/ AWGHT,AVOL1,AVOL2,ACENG1,ACENG2,ACENG3,ACMFG1
*      ,ACMFG2,AETA,FWGHT,FVOL1,FVOL2,FCENG1,FCENG2,FCENG3,FCMFG1
*      ,FCMFG2,FETA,PSPPLY,ANT,FBOOM,XIMPRV,BDWDTH,DABD,BLNGTH,BVOL
*      ,BWGHT,DEM,FREQ,EPSLON,MCONV,ITMAX,PSIK,SNRGI,DCYCLE, WGHTB
*      ,VOLB,XTR,ANTC,CSTR,CP1,CP2,RAD1,RAD2,SK1,SK2,PLX1,PLX2,DNLA
*      ,ANTD1,ANTD2,CBT1,CBT2,RECA1,RECA2,ANTH,XINST1,XINST2,CINST
*      ,CRCUR,CRCNR,NOANT,NCNFIG,NMOD1,NMOD2,NFBAND,NPTYPE,NXMTR
*      ,NMTPLX,NRAD,NSTK,NATT,NBATT,M,N,RECH1,RECH2,TRBD1,TRBD2,BVEST
*      ,BBB,ZMOD,UPFREQ,RPP,BT1,BT2,AT1,AT2,KATTN1,KATTN2,TZ1,TZ2,XLI
*      ,XLT1,XLT2,XLT3,XLT4,XPT1,XPT2,RPT1,RPT2,P1,P2,COSNT,TION
*      ,TATM,TCLD,TRAIN,TTL,AMMT,TE,TRCVRS,GE1,GE2,GS1,GS2,GM1,GM2
*      ,GSX,GEX,GMX,AMTPLX,BMTPLX,D3BD,ARCV,EXPR,FDLIN,ETAPT,ETAPC
*      ,ETAB,ETAPCU,ETALCF,RHOPC,DGRD,WPTR,WCU1,WCU2,WCU,WPCN,WLCFP1
*      ,WLCFP,VPTR1,VPTR,VCU1,VCU,VPCN1,VPCN,VLCFP1,VLCFP,CENGP1
*      ,CENGP2,CENGPC,CMFGP1,CMFGPC,CLC1,CLC2,CMFGLC,CENGLC,TSTRCT
*      ,LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF,RRWTR,RRWRVR,RRWPT,RRWBTT
*      ,RRWPC,RRWPCU,RRWLCF,RRVTR,RRVRVR,RRVPT,RRVBTT,RRVPC,RRVPCU
*      ,RRVLCF,RRCTR,RRCRVR,RRCP,RRCBTT,RRCP,RRCPCU,RRCLCF,VRHO
*      ,WMUX,WHARR,NPROTO,SKLS1,SKLS2,ATT1,ATT2,ATLS1,ATLS2
*      ,POP1,POP2,POP,IMO,IDA,IYR,XNGRMX
NAMELIST/NAMEP1/ XLIFE,PGMIN,PGMAX,D1MIN,D1MAX,D4MIN,D4MAX,TRMIN
*      ,TRMAX,XLMIN,XLMAX,DFMREC,PARANT,BWUPL
*      ,ACQLV,ENGLV,VSABD,ACQMP,ENGMP,URCBM,URCMDA
*      ,ENGBM,ENGMDA,PTRMIN,PTRMX1,PTRMX2,PTRMX3
*      ,BMWDTH,IBWC
*      ,RCVR1,RCVR2,HRCVR1,HRCVR2,XLC1,XLC2,XLC3,RCVC2
*      ,RCVC3
NAMELIST/NAME2/ NB,NCB,NF,NRS,NRLV,NRTSTA,NC,LSYS,LSAT,NTRF,NCF
*      ,NCTRF,NBRQD,ICNFIG,IMOD1,IMOD2,IFBAND,IXMTR,IRAD,ISTK,IATT
*      ,IBATT,IPTYPE,XLAT,XLONG,AMAJR,BMINR,NAPV,I11,I19,I21,I22,RHO
*      ,IBW,NSAT,ITASOI,SNRGI,IPDNSE,INGR,XNGRI,TITLE,ICASE,XO,JIT
*      ,PGMIN,PGMAX,D1MIN,D1MAX,D4MIN,D4MAX,TRMIN,TRMAX,XLMIN,XLMAX
*      ,SETDNS,ILIFE,NRGR,BWUPL,ACQLV,ENGLV,VSABD,ACQMP,ENGMP
*      ,URCBM,URCMDA,ENGBM,ENGMDA,PTRMIN,PTRMX1,PTRMX2,PTRMX3
*      ,BMWDTH,IBWC,NOANT
DATA PI,R,CVEL,BOL2,RSAT,RE/3.1415927, 22250., .98358, 1.38E-23,

```

26213., 3963./

```

*
DO 10 I=1,6
10 TITLE(I) = 10H
   NCNFIG = 8
   NMOD1 = 2
   NMOD2 = 2
   NFBAND = 4
   NPTYPE = 2
   NXMTR = 4
   NRAD = 2
   NSTK = 3
   NATT = 3
   NBATT = 3
   NOANT = 1
   NCTR = 1
   READ(5,NAME1)
   WRITE(6,NAME1)
   READ(5,NAMEP1)
   WRITE(6,NAMEP1)
   XLI = 10.**(XLI/10.)
   XLT1 = 10.**(XLT1/10.)
   XLT2 = 10.**(XLT2/10.)
   XLT3 = 10.**(XLT3/10.)
   XLT4 = 10.**(XLT4/10.)
9999 READ(5,NAME2)
   WRITE(6,NAME2)
   XIMPRV(1) = 1.
   XIMPRV(2) = 1.
   IF(IBW.EQ.1) GO TO 13
   IF((XLAT.GE.0.).AND.(XLAT.LT.20.)) RP = RPP(1)
   IF((XLAT.GE.20.).AND.(XLAT.LT.40.)) RP = RPP(2)
   IF(XLAT.GE.40.) RP = RPP(3)
   XLAT = XLAT * PI / 180.
   XLONG = XLONG * PI / 180.
   BETA = ACOS(COS(XLAT) * COS(XLONG))
   CBETA = COS(BETA)
   SBETA = SIN(BETA)
   RSL = (RSAT**2 + RE**2 - 2.*RE*RSAT*CBETA)**.5
   GAMMA = ASIN(RE * SBETA/RSL)
   EL = PI/2. - GAMMA - BETA
   IF(IBWC.EQ.0) GO TO 18
   RL = RSAT * COS(GAMMA) - RSL
   PSI = BMINR / RL
   ALPHAB = 2. * ASIN((RL*SIN(PSI/2.)) / ((RSL+RL)**2 + RL**2 -
*2. * RL * (RL + RSL) * COS(PSI/2.))**.5)
   ICOUNT = 0
   ZETA = AMAJR / RE
   ANGMAX = ASIN(RE/RSAT)
   COEF = RSAT / RE
   IF(ALPHAB.GE.(2.*(ANGMAX - GAMMA))) GO TO 12
   ALPHAP = ALPHAB
11 ZETAP = ASIN(COEF*SIN(GAMMA - ALPHAP/2.)) - ASIN(COEF*SIN(GAMMA +
* ALPHAP/2.)) - ALPHAP
   IF(ABS(ZETAP - ZETA).LE..001) GO TO 14
   ICOUNT = ICOUNT + 1

```



```

IF(ICOUNT .GT.20.) GO TO 15
DZETDA = -(COEF/2.) * (COS(GAMMA-ALPHAP/2.) / (1. - (COEF *
* SIN(GAMMA-ALPHAP/2.))**2 ) **.5 + COS(GAMMA + ALPHAP/2.) /
* (1. - (COEF * SIN(GAMMA + ALPHAP/2.))**2 ) **.5) - 1.
ALPHAP = ALPHAP + (ZETA - ZETAP) / DZETDA
GO TO 11
12 ALPHA = ALPHAB
GO TO 13
14 ALPHA = ALPHAP
IF(ALPHAB.GE.ALPHAP) ALPHA = ALPHAB
GO TO 13
15 WRITE(6,16)
16 FORMAT(///25X,55H*BEAMWIDTH REQUIRED BY MAJOR AXIS CANNOT BE DETERMINED*/25X,48H*CONTINUE WITH BEAMWIDTH REQUIRED BY MINOR AXIS*//)
GO TO 12
13 BMWDTH = ALPHA * 180. / PI
18 AOC = PI * AMAJR * BMINR / 4.
DO 3004 J4 = 1,6
IF(J4.NE.ILIFE(J4)) GO TO 3004
LSAT = XLIFE(J4)
DLTR = (LSAT - LTR) / LTR
DLRCVR = (LSAT - LRCVR) / LRCVR
DLPT = (LSAT - LPT) / LPT
DLPC = (LSAT - LPC) / LPC
DLPCU = (LSAT - LPCU) / LPCU
DLLCF = (LSAT - LLCF) / LLCF
IF(DLTR.LE.0.) DLTR = 0.
IF(DLRCVR.LE.0.) DLRCVR = 0.
IF(DLPT.LE.0.) DLPT = 0.
IF(DLPC.LE.0.) DLPC = 0.
IF(DLPCU.LE.0.) DLPCU = 0.
IF(DLLCF.LE.0.) DLLCF = 0.
WRTR = 1. + RRWTR * DLTR
WRCVR = 1. + RRWVR * DLRCVR
WRPT = 1. + RRWPT * DLPT
WRPC = 1. + RRWPC * DLPC
WRPCU = 1. + RRWPCU * DLPCU
WRLCF = 1. + RRWLCF * DLLCF
VRTR = 1. + RRVTR * DLTR
VRCVR = 1. + RRVVR * DLRCVR
VRPT = 1. + RRVPT * DLPT
VRPC = 1. + RRVPC * DLPC
VRPCU = 1. + RRVPCU * DLPCU
VRLCF = 1. + RRVLCF * DLLCF
CRTR = 1. + RRCTR * DLTR
CRCVR = 1. + RRCVR * DLRCVR
CRPT = 1. + RRCPT * DLPT
CRPC = 1. + RRCPC * DLPC
CRPCU = 1. + RRCPCU * DLPCU
CRLCF = 1. + RRCLCF * DLLCF
IFLAG = 0
L = M+N
DO 3001 J1=1,7
IF(J1.NE.ITASOI(J1)) GO TO 3001
SNRG=SNRGI(J1)

```

```

SNRG=10.**(SNRG/10.)
ITASO=ITASOI(J1)
IF(ITASO.EQ.7) ITASO=0
DO 3002 J2=1,4
IF(J2.NE.IPDNSE(J2)) GO TO 3002
PDNSE=POP(J2)
DO 3003 J3=1,8
IF(J3.NE.INGR(J3)) GO TO 3003
XNGR=XNGRI(J3)
DO 2000 I1=1,NCNFIG
IF(I1.NE.ICNFIG(I1)) GO TO 2000
NC = NCB * NOANT(I1)
DO 1900 I2=1,NMOD2
IF(I2.EQ.2) XIMPRV(2) = 3.*ZMOD(2)**2 * (ZMOD(2) + 1.)
IF(I2.NE.IMOD2(I2)) GO TO 1900
DO 1800 I10=1,NMOD1
IF(I10.NE.IMOD1(I10)) GO TO 1800
IF(I2.NE.I10) GO TO 1800
IF(I2.EQ.I10) I8 = 1
IF(I2.NE.I10) I8 = 2
IF(I10.EQ.2) GO TO 20
BVID12 = BVEST + BBB(1)
GO TO 22
20 BVID12 = 2. * BBB(2) * (1. + ZMOD(2))
22 IF(I2.EQ.2) GO TO 24
BVID34 = BVEST + BBB(1)
GO TO 25
24 BVID34 = 2. * BBB(2) * (1. + ZMOD(2))
25 BCHANN(1) = NCB * (BVID12 + 1.0E-03 + NAPV*0.15E-03)
BCHANN(2) = NCB * (BVID34 + 1.0E-03 + NAPV*0.15E-03)
DO 1700 I3=1,NFBAND
IF(I3.NE.IFBAND(I3)) GO TO 1700
I9=1
IF(I3.EQ.2) I9=2
IF(I3.GE.3) I9=3
IF(BDWIDTH(I3).GE.BCHANN(2)) GO TO 50
IFLAG = 1
CALL ERROR(IFLAG)
GO TO 1700
50 D(3) = 70. * CVEL / (BMWIDTH * FREQ(I3))
IF (D(3).LE.DABD(I1)) GO TO 60
IFLAG = 2
CALL ERROR(IFLAG)
GO TO 1700
60 CONTINUE
IF((I1.EQ.5).AND.(D(3).LT.D3BD)) GO TO 65
GO TO 66
65 IFLAG = 5
CALL ERROR(IFLAG)
GO TO 1700
66 CONTINUE
I24 = 2
IF(I3.EQ.3) I24 = 1
FREQ1 = UPFREQ(I24)
D(2) = 70. * CVEL / (BMWIDTH * FREQ1)

```

```

IF((I1.EQ.4).OR.(I1.EQ.6)) D(2) = 70. * CVEL / (FREQ1 * BWUPL)
MPTYPE = NPTYPE
IF(I1.NE.5) MPTYPE = 1
DO 1600 I6=1,MPTYPE
IF(I6.NE.IPTYPE(I6)) GO TO 1600
DO 1500 I7=1,NXMTR
IF(I7.NE.IXMTR(I7)) GO TO 1500
IF((I2.EQ.1).AND.(I3.GT.2)) GO TO 250
GO TO (1,2,3,4),I7
1 IF(I3.EQ.1) GO TO 650
GO TO 1500
2 IF(I3.LE.2) GO TO 650
GO TO 1500
3 IF(((I2.EQ.1).AND.(I3.LE.2)).OR.((I2.EQ.2).AND.(I3.LE.2))) GO TO
* 650
GO TO 1500
4 IF(I2.EQ.1) GO TO 250
IF(I3.GT.1) GO TO 650
GO TO 1700
650 CONTINUE
DO 1300 I13=1,NRAD
IF(I13.NE.IRAD(I13)) GO TO 1300
DO 1200 I14=1,NSTK
IF(I14.NE.ISTK(I14)) GO TO 1200
DO 1100 I15=1,NATT
IF(I15.NE.IATT(I15)) GO TO 1100
DO 1000 I20=1,NBATT
IF(I20.NE.IBATT(I20)) GO TO 1000
DLBTT = (LSAT - LBATT(I20)) / LBATT(I20)
IF(DLBTT.LE.0.) DLBTT = 0.
WRBTT = 1. + RRWBTT * DLBTT
VRBTT = 1. + RRVBTT * DLBTT
CRBTT = 1. + RRCBTT * DLBTT
I16 = 1
IF(I21.GT.1) I16 = 2
IF(I3.GT.1) I16 = 2
970 CONTINUE
CALL MADD(N,1,X,0.,XO(1,I1),1.,X)
CALL INIT (X,N,M,L)
CALL CONV (X,N,M,L)
IF(I3.GT.1) GO TO 1000
IF(G(4).LE.100.) GO TO 1000
IF(I16.EQ.2) GO TO 1000
I16 = 2
GO TO 970
1000 CONTINUE
1100 CONTINUE
1200 CONTINUE
1300 CONTINUE
1500 CONTINUE
1600 CONTINUE
1700 CONTINUE
1800 CONTINUE
250 CONTINUE
1900 CONTINUE

```

2000 CONTINUE
6000 CONTINUE
3003 CONTINUE
3002 CONTINUE
3001 CONTINUE
3004 CONTINUE
XLAT = XLAT * 57.29577951
XLONG = XLONG * 57.29577951
GO TO 9999
END

5.2 SUBROUTINE INIT

The main function of subroutine INIT is to set boundaries and initialize certain parameters which are configuration dependent. The subroutine also computes the learning curve coefficients to be used in determining the cost of the ground downlink system. The flow diagram is given in Figure 5.2.

This routine calculates the number of audio channels (NAUDC) from the number of audio channels/video (NAPV) and the number of video channels (NC). It sets the flags for launch vehicle type (I18) and for structure (I12). The learning curve coefficients are calculated from the total population (XNGRP) over all of the satellite systems (NSAT) and beams required/system (NBRQD) as specified by the user. XNGR is required population per beam.

The subroutine also sets boundary values on the dependent parameters. It computes the feed length(s) (XLF1 and XLF2), the packaged length of the antenna (XLA), the upper boundary on the length of the equipment module (BLEM), and determines the upper boundary on the volume of the equipment module (BVEM) as a function of the concept being considered. It also sets the flags for the downlink antenna and feed boom type (I4, I5, respectively). For concept E, it computes the diameter of the equipment module (DEM) which, in this case, is dependent on the diameter of the downlink antenna (D_3).

The initial value of the vector of independent parameters, $\bar{X}_0$, is then compared against the boundary values, $\bar{X}_B$, to see whether any are exceeded. If a boundary is exceeded, the initial value x_{0j} is replaced by its boundary value x_{bj} .

The coefficients for calculating weight and volume of the downlink antenna and feed boom are then set as a function of type.

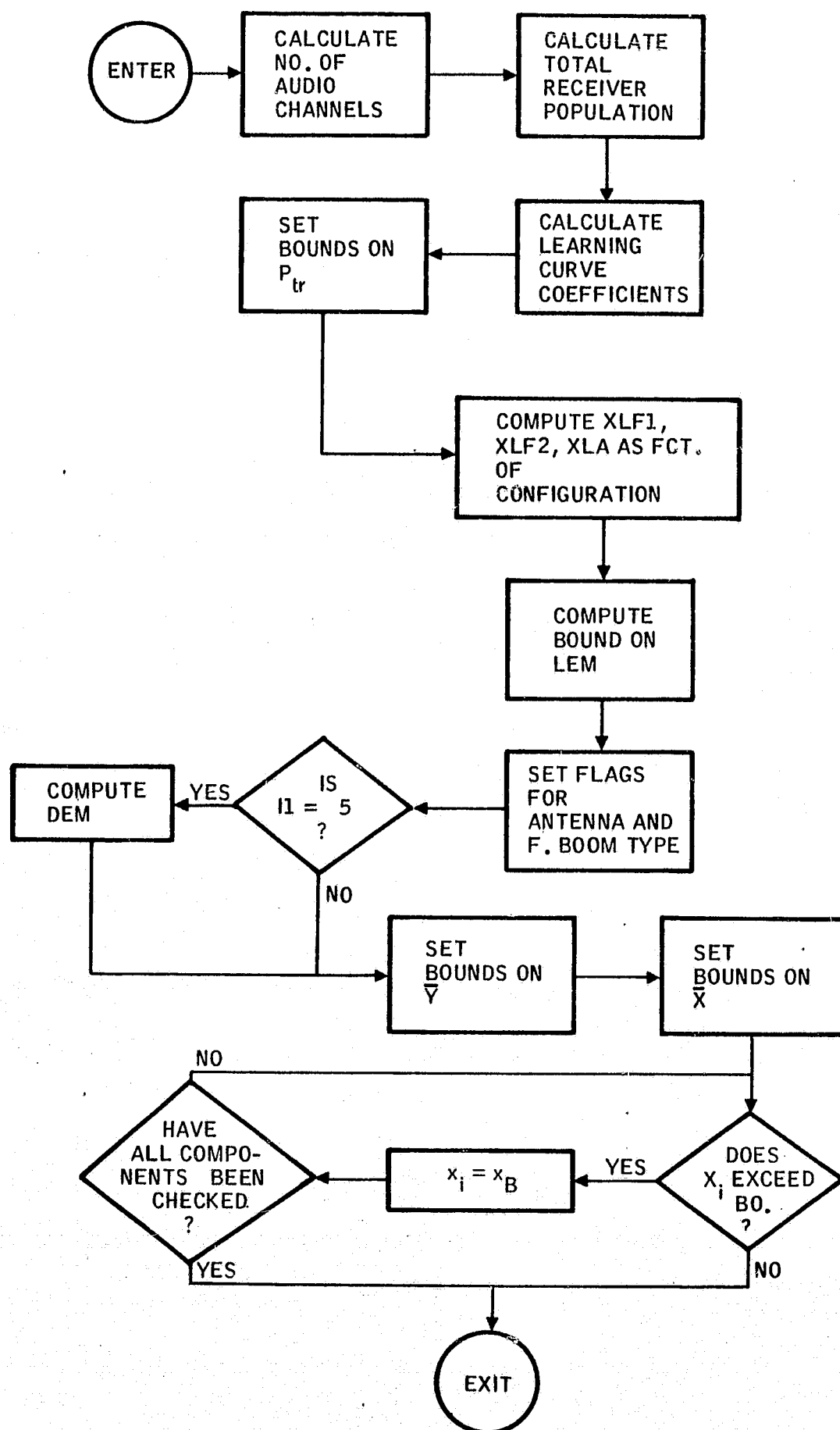


Figure 5.3 Flow Diagram = Subroutine INIT

```

SUBROUTINE INIT (X,N,M,L)
COMMON/BOUND/IARRAY(15),JARRAY(15),YMIN(15),YMAX(15)
COMMON/DIAM/ D(4),G(4)
COMMON/MAIN/ I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15
*           ,I16,I17,I18,I19,I20,I21,I22,I23,I24,I25
*           ,XQ,XJ,XK,XL,XP1,XP2,XP3,XP4
COMMON/MISC/ BLEM,BVEM,XLF1,XLF2,XLF,XLA,SNRG,XTA,XIMPRV(2),BVID12
*           ,BVID34,TSTRCT,AMAJR,BMINR
COMMON/NUMB/ NTR,NCTR,NF,LSYS,NSL,NRS,NLV,NRLV,NRSTA,NB,NCB,NC
*           ,NCTRF,NTRF,NCF,LSAT,FREQ1,GR85,GR89,GR95,GR,XNGR,XNGRP
*           ,NAPV,NAUDC,PDNSE,ITASO,NBRQD,AOC,NSAT,NBS,SETDNS,JIT
COMMON/PARA/ PLOG,ETA,ARD,VTR,WTR,FREQ4,XNS,RHO,AKP,XNG
*           ,CE,CMFG,WEM,WLOG,PLOG,CSTPYR,G4LOG,TLOG
*           ,VEM,WSTLOG,PSLOG,WHLOG,PRLOG,D1LOG
*           ,D4LOG,PTALOG,PCULOG,PPCLOG,PPTLOG,CEPP,CMFGPP
COMMON/PARBD/PGMIN,PGMAX,D1MIN,D1MAX,D4MIN,D4MAX,TRMIN(2),TRMAX(2)
*           ,XLMIN,XLMAX,XNGRMX,VSABD,PTRMIN,PTRMX1,PTRMX2
*           ,PTRMX3
COMMON/SYS/ WMP,VMP,WGHTB(3,3),VOLB(3,3),XTR(3),ANTC(3),CSTR(3,3)
*           ,CP1(3,2),CP2(3,2),RAD1(3,2),RAD2(3,2),SK1(2,2,3),SK2(2,2,3)
*           ,PLX1(4,3),PLX2(4,3),DNLA,ANTD1(3,4),ANTD2(3,4),CBT1(3,3)
*           ,CBT2(3,3),RECH(3),RECH1(3,3,4),TRBD1(2,4),RECH2(3,3),TRBD2(2)
*           ,RECA1,RECA2,ANTH(3,4),XINST1,XINST2,CINST(3,4),CRCUR(3,2)
*           ,CRCNR(3,2),NOANT(8),DCYCLE,ARCV(4),ETAPT,ETAPC,ETAB,ETAPCU
*           ,ETALCF,RHOPC,DGRD(2),WPTR(3),WCU1,WCU2,WCU(3,3),WPCN(3)
*           ,WLCFP1,WLCFP(2,2),VPTR1,VPTR(3,2),VCU1,VCU(3,2),VPCN1
*           ,VPCN(3,2),VLCFP1,VLCFP(2,2),CENGP1,CENGP2,CMFGP1,CENGPC(3,3)
*           ,CMFGPC(3,2),CLC1,CLC2,CENGLC(2,3),CMFGLC(2,3),XCNST
*           ,DEGRAD,CPTUR(3),CPTNR(3),CPCUUR(3),CPCUNR(3),CPCUR(3),CPCNR(3)
*           ,CBETA,SBETA,IETBD1,IETBD2,WMUX(3,3),WHARR(3,8),CHARUR(3)
*           ,CHARNR(3),NPROTO,SKLS1(2,2,3),SKLS2(2,2,3),ATT1(2,2,3)
*           ,ATT2(2,2,3),ATLS1(2,2,3),ATLS2(2,2,3),IAML,IWS,AMLLOG
*           ,DFMREC,PARANT(4),NRGR,ACQLV(8),ENGLV(8),ACQMP(8),ENGMP(8)
*           ,URCBSM,ENGBSM,URCMDA,ENGMDA
*           ,RCVR1(2),RCVR2(1),HRCVR1(3,4),HRCVR2(3,2),XLC1(3,2),XLC2(3,1)
*           ,XLC3(3,1),RCVC2,RCVC3,DELC1,DELC2
COMMON/TABLE/ AWGHT(3,4,3),AVOL1(3,4,3),AVOL2(3,4,3),ACENG1(3,4,3)
*           ,ACENG2(3,4,3),ACENG3(3,4,3),ACMFG1(3,4,3)
*           ,ACMFG2(3,4,3),AETA(3,4,3),FWGHT(3,4,3),FVOL1(3,4,3)
*           ,FVOL2(3,4,3),FCENG1(3,4,3),FCENG2(3,4,3)
*           ,FCENG3(3,4,3),FCMFG1(3,4,3),FCMFG2(3,4,3)
*           ,FETA(3,4,3),PSPPLY(4,2),ANT(4,2)
*           ,FBOOM(4,3),BCHANN(2),DABD(8),BLNGTH(8)
*           ,BVOL(8),BWGHT(8),DEM(8),FREQ(4),BDWDTH(4)
*           ,UPFREQ(2),EXPR,FDLIN
REAL NREC1,NREC2,NREC3,NREC41,NREC42,NREC43,K1,K2,NOANT
REAL KATTN1,KATTN2,LSYS,LSAT,NBRQD,NLV,NTR,NAPV,NAUDC,NCF
REAL LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF
REAL NSAT,NBS
REAL NRGR
DIMENSION X(1)
NAUDC = NAPV * NC
NTR = NC
NLV = LSYS / LSAT
I18 = 1

```

```

IF(I1,EQ.5) I18 = 2
I12 = 1
IF(I1,EQ.4) I12 = 2
DA = D(3)
XNGRP = XNGR * NSAT * NBRQD
XNGRPP = XNGRP
IF(XNGRP.GE.XNGRMX) XNGRPP = XNGRMX
GR89 = (XNGRPP** .83188 - .16812) / (XNGRPP * .83188)
GR95 = (XNGRPP** .92599 - .07401) / (XNGRPP * .92599)
XNGRPP = XNGRPP * NRGR
IF(XNGRPP.GT.XNGRMX) XNGRPP = XNGRMX
GR85 = (XNGRPP** .76554 - .23446) / (XNGRPP * .76554)

```

```

... SET BOUNDS ...
... DEPENDENT PARAMETERS ...
... PTR ...

```

```

C
C
C
YMIN(1) = PTRMIN
YMAX(1) = PTRMX1
IF(I3.GT.2) GO TO 5
IF(I9,EQ.1) YMAX(1) = PTRMX2
5 CONTINUE
IF(I7,EQ.2) YMAX(1) = PTRMX3
XLF1 = 0.
XLF2 = 0.
GO TO (10,20,30,40,50,60,80,30), I1
10 IF (DA.GT.5) GO TO 15
XLF1 = .2 * DA + .25
XLA = .5
I4 = 2
I5 = 3
BLEM = BLNGTH(I1) - XLF1 - XLA
GO TO 80
15 XLF1 = (.2 * DA + .25)
XLA = .12 * DA
I4 = 1
I5 = 3
BLEM = BLNGTH(I1) - XLF1 - XLA
GO TO 80
20 IF (DA.GT. 6.67) GO TO 25
XLF1 = .2 * DA + .25
XLA = .5
I4=2
I5=3
BLEM= BLNGTH(I1) -XLF1-XLA
GO TO 80
25 XLF1=.5*(.2*DA+.25)
XLA = .12 * DA
I4=1
I5=3
BLEM=BLNGTH(I1) -XLF1-XLA
GO TO 80
30 IF(DA.GT.25.)GO TO 35.
XLF1 = .2 * DA + .5
XLF2 = .2 * D(2) + .25
XLA = .11 * DA
I4=1

```



```

      I5=3
      BLEM = BLNGTH(I1) - XLF1 - XLF2 - XLA
      GO TO 80
35  XLF1 = .10 * DA + .5
      XLF2 = .2 * D(2) + .25
      XLA = .11 * DA
      I4=1.
      I5=1
      BLEM = BLNGTH(I1) - XLF1 - XLF2 - XLA
      GO TO 80
40  CONTINUE
      XLF1 = .2 * DA + .25
      I4 = 2
      I5 = 3
      GO TO 80
50  CONTINUE
      XLF1 = .02 * DA + 3.0
      XLF2 = .2 * D(2) + .25
      XLA = .11 * DA
      I4 = 1
      I5 = 2
      BLEM = BLNGTH(I1) - XLA - XLF1 - XLF2
      DEM(5) = .10 * DA
      GO TO 80
60  CONTINUE
      I4 = 1
      I5 = 3
      80 GO TO (110,130,130,140,150,160,190,180), I1
110 BVEM=13.3*BLEM
      GO TO 190
130 CONTINUE
      BVEM = 25.3 * BLEM
      GO TO 190
140 CONTINUE
      BVEM = 882.
      IF(I22.EQ.0) BVEM = 1600.
      GO TO 190
150 CONTINUE
      BVEM = (3.14 / 4.) * BLEM * DEM(I1) **2
      GO TO 190
160 CONTINUE
      BVEM = 563.
      BVEM = 900.
      IF(I22.EQ.0) BVEM = 1300.
      GO TO 190
180 BVEM = 78.5 * BLEM
190 CONTINUE
      YMIN(2)=0.
      YMAX(2)=BWGHT(I1)
      YMIN(3)=0.
      YMAX(3)=BVEM
      YMIN(4)=0.
      YMAX(4)=BVOL(I1)
      YMIN(5) = PGMIN

```

C

... PG ...

YMAX(5) = PGMAX

... VPP ...

YMIN(6) = 0.

YMAX(6) = BVOL(I1)

IF(I1.EQ.3) YMAX(6) = VSABD

IF(I1.EQ.8) YMAX(6) = 4000.

XLF=XLF1+XLF2

... INDEPENDENT PARAMETERS ...

YMIN(M+1) = D1MIN

...D(1)

YMAX(M+1) = D1MAX

YMIN(M+2) = D4MIN

...D(4)

YMAX(M+2) = D4MAX

K = 1

IF(I3.GT.1) K = 2

YMIN(M+3) = TRMIN(K)

YMAX(M+3) = TRMAX(K)

... CHECK INDEPENDENT PARAMETERS BDS ...

DO 200 I=1,N

J = I + M

IF(X(I).LT.YMIN(J)) X(I) = YMIN(J)

IF(X(I).GT.YMAX(J)) X(I) = YMAX(J)

200 CONTINUE

... WEIGHT ...

XL = ANT(1,I4)

XP1 = ANT(2,I4)

... VOLUME ...

XM = ANT(3,I4)

XP2 = ANT(4,I4)

... WEIGHT ...

XN = FB00M(1,I5)

XP3 = FB00M(2,I5)

... VOLUME ...

XY = FB00M(3,I5)

XP4 = FB00M(4,I5)

RETURN

END

5.3 SUBROUTINE CONV

When control is returned from subroutine INIT, the driver calls subroutine CONV. The function of CONV is to control execution of the iterative optimization technique. A flow diagram is given in Figure 5.4.

The iteration counter (ITER) is initialized to zero and the flag MBPRV is initialized.

MBPRV = 0 indicates that there was no boundary violation on the previous iteration

MVPRV = 1 indicates that there did exist a boundary violation on the previous iteration.

The value of the iteration index is compared to the maximum number of desired iterations (ITMAX). If the two values are equal, an exit is made indicating that the process has not yet met the convergence test requirements. If they are not equal, the algorithm continues. Subroutine SYSEQ is called to calculate the value of $\bar{Y}_i$ for the current iteration i ($i = \text{ITER} + 1$). Then subroutine BCHECK is called to determine whether or not any boundaries are exceeded. The routine returns the number of boundary violations (MB). If there are no violations, $\bar{X}_i$ is accepted and the process continues to call the partial subroutines. However, if the number of boundary violations is not zero and there were no boundary violations on the previous iteration, $\bar{X}$ is reset to its previous value and the iteration counter is reduced. The step-size control factor (S) is reset to its previous value (SP). $\bar{Y}_i$ ($i = \text{ITER}$) is then recalculated. A new perturbation is then computed which takes the boundary condition into account.

Now the partial routine, subroutine PRTIAL, is called to calculate the cost gradient (GRAD) and the gradient of the component exceeding the boundary by the greatest degree (P^T). A scaling factor is then applied to the partials with respect to the effective noise temperature of the receiver.

If there is no boundary violation,

$$\Delta X_i = -K_c g.$$

If there exists a boundary violation,

$$\Delta X_i = -K_c g \left[I - \Phi(\Phi^T \Phi)^{-1} \Phi^T \right] g - K_y \Phi (\Phi^T \Phi)^{-1} \Delta y_j$$

The angle between the previous two perturbations is calculated and used to compute the step-size control factor (as explained in Section 2.2).

The step is applied and $\bar{X}' = \bar{X}_i + \Delta \bar{X}_i$ is checked against its boundary values. If boundaries are violated, the appropriate components are placed on the boundary and $\Delta \bar{X}_i$ is adjusted to reflect this change.

The convergence test is then applied to $\Delta \bar{X}_i$. If convergence has been achieved, an exit is performed; if not, control is transferred to perform an additional iteration.

The convergence test requires that $\frac{|\Delta X_j|}{|\bar{X}_j|} \leq \epsilon$ (ϵ input) for all components $j = 1, 2, 3$ for a preselected number of consecutive iterations (MCONV).

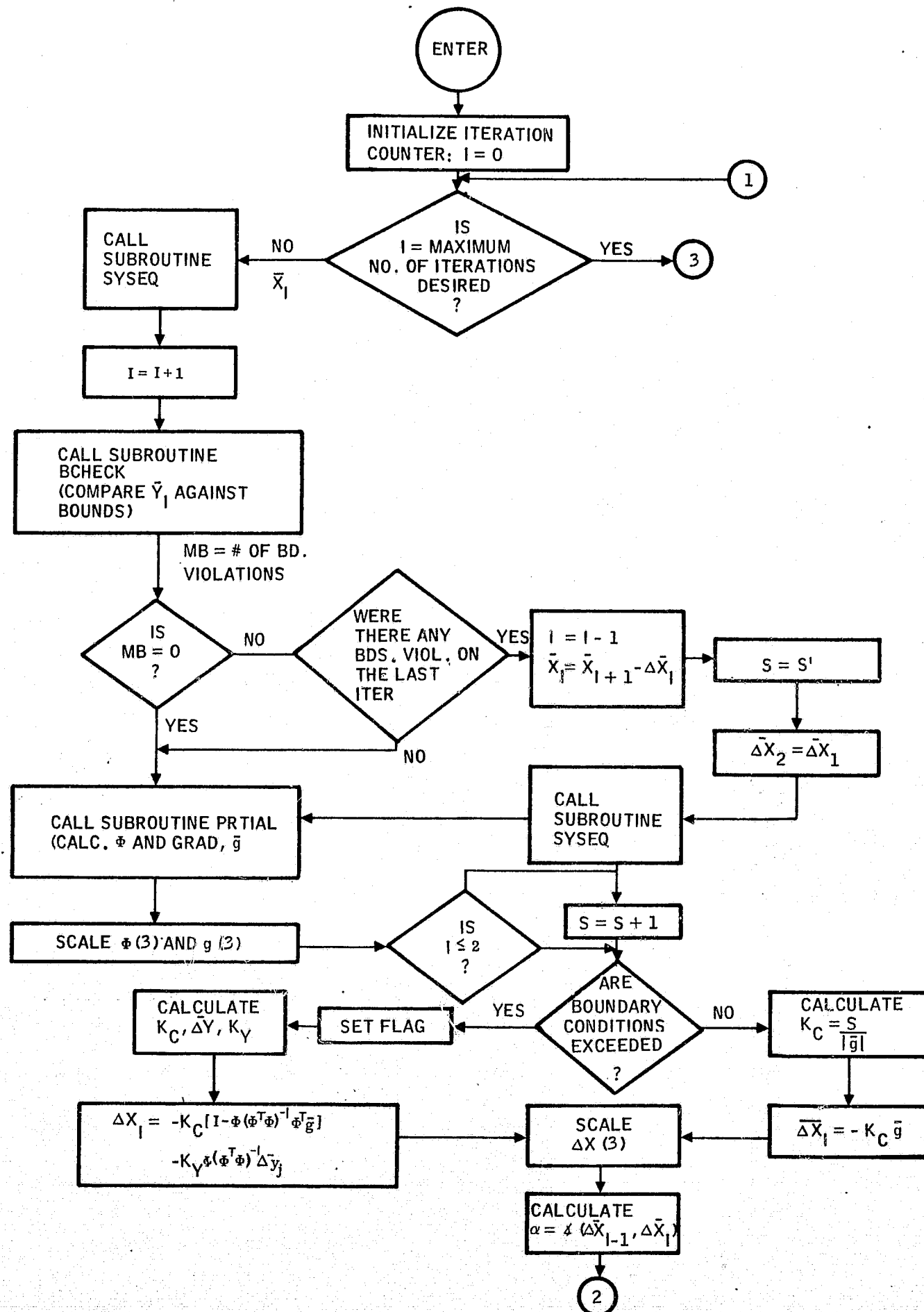
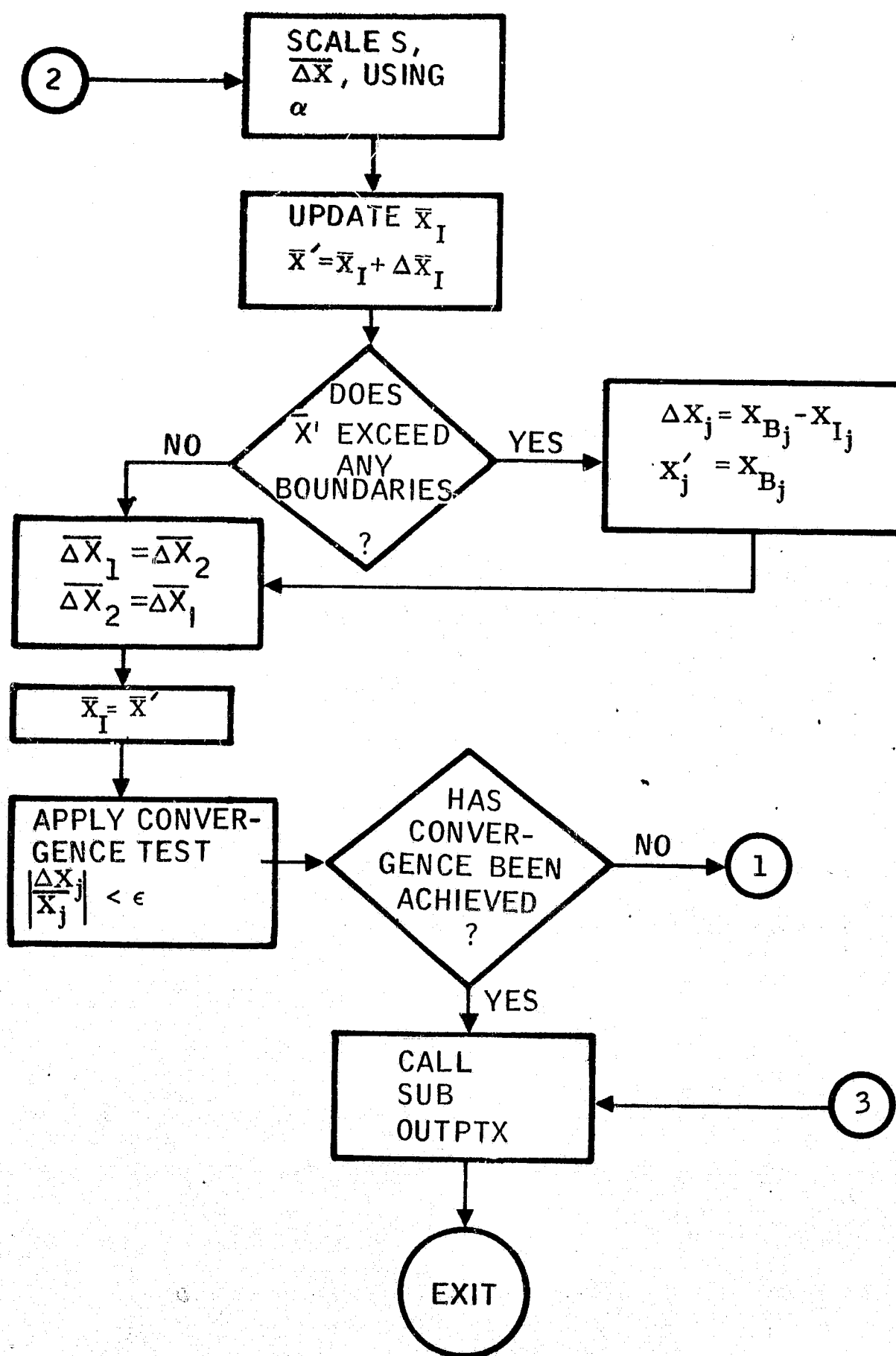


Figure 5.4 Flow Diagram - Subroutine CONV



* j indicates component
violating the boundary

Figure 5.4 Flow Diagram - Subroutine CONV-Continued

```

SUBROUTINE CONV (X,N,M,L)
COMMON/BOUND/IARRAY(15),JARRAY(15),YMIN(15),YMAX(15)
COMMON / CNVRGE / EPSLON,MCONV,ITMAX,PSIK,GRAD(5)
COMMON/PARA/ PLOG,ETA,ARD,VTR,WTR,FREQ4,XNS,RHO,AKP,XNG
*          ,CE,CMFG,WEM,WLOG,PLOG,CSTPYR,G4LOG,TLOG
*          ,VEM,WSTLOG,PSLOG,WHLOG,PRLOG,D1LOG
*          ,D4LOG,PTALOG,PCULOG,PPCLOG,PPTLOG,CEPP,CMFGPP
REAL NREC1,NREC2,NREC3,NREC41,NREC42,NREC43,K1,K2,NOANT
REAL KATTN1,KATTN2,LSYS,LSAT,NBRQD,NLV,NTR,NAPV,NAUDC,NCF
REAL LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF
DIMENSION L1(10),L2(10)
DIMENSION PT(5,10), P(10,5),PPT(10,10),PPTI(10,10),PSI(10),
*          STOR(10),PTPPTI(5,10),H(5,5),HG(5),DX1(5),DX2(5),DX(5),
*          X(5),Y(20),STOR1(10),XP(5),DXP(3),DX2P(3)
DIMENSION IC2(5),IC3(5)
ITER = 0
DO 400 I=1,5
400 IC2(I) = IC3(I) = 0
R1 = 10.
MBPRV = 1
100 IF(ITER.EQ.ITMAX) GO TO 350
CALL SYSEQ (X,Y,ITER,N,M,L)
ITER = ITER + 1
CALL BCHECK(Y,MB,KS,MBS,N,M,L)
IF((MBPRV.EQ.0).AND.(MB.NE.0)) GO TO 101
DO 401 I=1,4
IC2(6-I) = IC2(5-I)
401 IC3(6-I) = IC3(5-I)
IC2(1) = IC3(1) = 0
IF(DX(2) * DX1(2).LT.0.) IC2(1) = 1
IF(DX(3) * DX1(3) .LT.0.) IC3(1) = 1
ISUM2 = ISUM3 = 0
DO 402 I=1,5
ISUM2 = ISUM2 + IC2(I)
402 ISUM3 = ISUM3 + IC3(I)
IF(ISUM2.LE.ISUM3 + 2) GO TO 403
R1 = 2. * R1
404 DO 405 I=1,5
405 IC2(I) = IC3(I) = 0
GO TO 406
403 IF(ISUM3.LE.ISUM2 + 2) GO TO 406
R1 = R1 / 2.
GO TO 404
406 CONTINUE
GO TO 105
101 CALL MADD(N,1,X,1.,DX,-1.,X)
ITER = ITER - 1
S = SP
CALL MADD(N,1,DX2,0.,DX1,1.,DX2)
CALL SYSEQ(X,Y,ITER,N,M,L)
105 CALL PRTIAL(Y,X,PT,MBS,N,M,L,ITER,MB)
R2 = 1.
GRAD(1) = R2 * GRAD(1)
GRAD(3) = R1*GRAD(3)
PT(3,1) = R1*PT(3,1)

```

```

PT(1,1) = R2 * PT(1,1)
IF(ITER.LE.2) S = 1.
IF(MB.EQ.0) GO TO 200
MBPRV = 1.
MBP = 1
CALL TRS(PT,P,N,MBP)
CALL MTMPY(P,PT,PPT,MBP,N,MBP)
CALL INVL(PPT,PPTI,MBP,MBP,L1,L2,DET,IDFLAG)
IF(IDFLAG.EQ.1) GO TO 1000
IFLAG = 3
CALL ERROR(IFLAG)
GO TO 2000
1000 CONTINUE
I = 1
J = IARRAY(MBS)
K = JARRAY(MBS)
IF(K.EQ.1) GO TO 178
PSI(I) = Y(J) - YMAX(J)
GO TO 180
178 PSI(I) = Y(J) - YMIN(J)
180 CONTINUE
PSIK = 1.
CALL MTMPY(PPTI,PSI,STOR,MBP,MBP,1)
CALL MTMPY(PSI,STOR,VALU1,1,MBP,1)
CALL MTMPY(PT,PPTI,PTPPTI,N,MBP,MBP)
CALL MTMPY(PTPPTI,P,H,N,MBP,N)
CALL MTMPY(H,GRAD,HG,N,N,1)
CALL MADD(N,1,GRAD,1.,HG,-1.,HG)
CALL MTMPY(HG,HG,HTH,1,N,1)
XK = (S**2 - PSIK * VALU1) / HTH
IF(XK.LT.0.) GO TO 190
XK = XK**.5
GO TO 195
190 XK=0.
195 CALL MTMPY(PT,STOR,STOR1,N,MBP,1)
CALL MADD(N,1,HG,-XK,STOR1,-PSIK,DX)
DX(3) = R1 * DX(3)
DX(1) = R2 * DX(1)
GO TO 250
200 GRADM = VMAG(GRAD,N)
XK = S / GRADM
MBPRV = 0
DO 225 I=1,N
DX(I) = -XK * GRAD(I)
225 CONTINUE
DX(1) = R2 * DX(1)
DX(3) = R1 * DX(3)
250 CONTINUE
IF(ITER.EQ.1) GO TO 252
DXP(1) = 0.
DXP(2) = DX(2)
DXP(3) = DX(3)
DX2P(1) = 0.
DX2P(2) = DX2(2)
DX2P(3) = DX2(3)

```



```

VDX = VDOT(DXP,DXP,N)
IF(VDX.EQ.0.) GO TO 350
VDX = SQRT(VDX)
CALL MTMPY(DXP,DX2P,VALUE,1,N,1)
CALPHA = VALUE / (VDX * VMAG(DX2P,N))
DO 251 I=1,N
DX(I) = DX(I) * (.75 + .25 * CALPHA)
251 CONTINUE
SP = S
IF(CALPHA.LT.0.) S = (.75 + .25*CALPHA) * S
IF(CALPHA.GE.0..AND.ITER.LE.10) S=(.5 + 1.5*CALPHA)* S
IF(CALPHA.GE.0..AND.ITER.GT.10) S=(.55 + .45*CALPHA) * S
IF(S.GE.20.) S = 20.
252 CONTINUE
IF (ITER .EQ.1) DXPP = 10.
IF (GRADP*GRAD(1) .LT. 0.) DXPP = DXPP/2.
DX(1) = -DXPP *GRAD(1)/ABS(GRAD(1))
GRADP = GRAD(1)
CALL MADD(N,1,X,1.,DX,1.,XP)
DO 230 I=1,N
IF(XP(I).LE.YMAX(M+I)) GO TO 231
DX(I) = YMAX(M+I) - X(I)
XP(I) = YMAX(M+I)
GO TO 232
231 IF(XP(I).GE.YMIN(M+I)) GO TO 232
DX(I) = YMIN(M+I) - X(I)
XP(I) = YMIN(M+I)
232 CONTINUE
230 CONTINUE
VDX = VDOT(DX,DX,N)
IF(VDX.EQ.0.) GO TO 350
DO 291 I=1,N
DX1(I) = DX2(I)
DX2(I) = DX(I)
291 CONTINUE
JFLAG = 0
JCONV = 0
DO 300 K=1,N
RATIO = ABS(DX(K)/ X(K))
IF(RATIO.LE.EPSLON) JCONV = JCONV + 1
X(K) = XP(K)
300 CONTINUE
CALL OUT(S,XK,HG,STOR1,DX,Y,ITER,MB,MBS,KS,JCONV,ICONV,N,M,L,R1)
IF(JCONV.LT.N) GO TO 325
ICONV = ICONV + 1
IF(ICONV.LT. MCONV) GO TO 100
JFLAG= 1
GO TO 350
325 ICONV = 0
GO TO 100
350 CONTINUE
2000 CONTINUE
CALL ZERO(PT,5,10)
CALL OUTPTX (Y,N,M,L)
RETURN
END

```

5.4 SUBROUTINE SYSEQ

The subroutine SYSEQ is used to calculate the values of the system parametric relations, $\bar{Y}_i(\bar{X}_i)$, and the cost elements of the system at iteration i .

Those calculations not dependent on the value of $\bar{X}_i$ are performed only at the first call to subroutine SYSEQ (ITER=0). These include:

- a) the terms for the attenuation model,
- b) the temperatures for the noise model, and
- c) the weight, volume, power, and cost terms which are not a function of $\bar{X}_i$.

Subroutine PLACE is called by SYSEQ to set the coefficients to be used in determining the weight, volume, cost, and efficiency terms of the downlink transmitter.

The system equations for the spacecraft system include implicit functions. As a result, it is necessary to solve the equations iteratively. The number of iterations required for this solution is specified by a fixed constant (JIT). The optimization of the total system cost is determined on a cost/year/beam basis.

The unit recurring, non-recurring, acquisition, and operations cost elements are numbered in order to coincide as closely as possible with the order presented in Table 5.1.

Relevant parameters found in SYSEQ are identified in the following list (more or less in the order in which they appear).

TABLE 5.1 SYSEQ PARAMETER LIST

NBS	Number of downlink beams (antennas)/satellite
NTR	Number of downlink transmitters
NC	Number of downlink channels/satellite
XPT1,XPT2	Transmitter pointing losses (uplink, downlink)
RPT1,RPT2	Receiver pointing losses (uplink, downlink)
P1, P2	Polarization losses (uplink, downlink)
EL	Elevation angle of satellite beam
XLR12, XLR34	Attenuation due to rain
XLRH34	Attenuation due to rain (horizontal) (downlink)
XLC12, XLC34	Attenuation due to clouds (uplink, downlink)
XLA12, XLA34	Atmospheric attenuation (Oxygen and water vapor) (uplink,downlink)
FSL12, FSL34	Freespace loss (uplink, downlink)
XLMUX	Multiplexer loss (downlink)
XLMUX1	Multiplexer loss (uplink)
ATL1, ATL2	Total atmospheric attenuation (uplink, downlink)
XCIRC1 XCIRC2	Transmitter circuit loss (uplink, downlink)
RCIRC1 RCIRC2	Receiver circuit loss (uplink, downlink)
XL12, XL34	Total loss (uplink, downlink)
XNS	Effective temperature noise at the satellite
TCOS	Ambient cosmic temperature
TCO	Brightness temperature contribution, cosmic
TI	Brightness temperature contribution, ionosphere
TA	Brightness temperature contribution, atmosphere
TC	Brightness temperature contribution, cloud

TR	Brightness temperature contribution, rain
TT	Brightness temperature contribution, transmission line
TMM	Ambient noise temperature, man-made
TM	Brightness temperature contribution, man-made
WMP,VMP,MP	Weight, volume, power for manned provisions
WCOND,VCOND	Weight and volume of power conditioning equipment
DEGRAD	Power supply degradation factor
PAC,WAC,VAC	Power, weight, volume of attitude control
PCS,WCS,VCS	Power, weight, volume of station-keeping
PINC	Power required for instrumentation and communication
ADEGRD	Degradation factor for antenna
D(1),G(1)	Ground uplink antenna diameter and gain
D(2),G(2)	Satellite uplink antenna diameter and gain
D(3),G(3)	Satellite downlink antenna diameter and gain
D(4),G(4)	Ground downlink antenna diameter and gain
NCB	Number of channels/beam
WINC,VINC	Weight and volume of instrumentation and communication hardware
XJ,XJC	Coefficients for equations of weight of power supply
XK,XKC	Coefficients for equations of volume of power supply
ETALC	Efficiency of λ/c filter
XCNST	Conversion factor from peak to peak to rms
WREC,VREC,PR	Weight, volume, and power of spacecraft receiver
WMPLX,VMPLX	Weight and volume of multiplexer
WUPLA,VUPLA	Weight and volume of spacecraft uplink antenna
WA,VA	Weight and volume of downlink antenna
WFB	Weight of feed boom for downlink antenna
CREC1,CREC2	Unit recurring and non-recurring cost for the spacecraft receiver
CMPLX1, CMPLX2	Unit recurring and non-recurring cost for the spacecraft multi- plexer
CANTD1, CANTD2	Unit recurring and non-recurring cost for the spacecraft downlink antenna

CUPL1, CUPL2	Unit recurring and non-recurring cost for the spacecraft uplink antenna
NSL	Number of satellite/launch
GS	Relative discrimination of antenna to cosmic and sky noise
GM	Relative discrimination of antenna to man-made noise
GE	Relative discrimination of antenna to earth noise
TEFF	Effective noise temperature
XNG	Ambient ground noise
Y(1)	Downlink transmitter power output/video channel
PTA	Downlink transmitter power output/audio channel
IVLEV	Vector indicating level of power for setting the coefficients to determine the volume of the downlink transmitters (video and audio)
ICLEV	Vector indicating level of power for setting the coefficients to determine cost of the downlink transmitters (video and audio)
ETA	Efficiency of downlink video transmitters
ETAP	Efficiency of downlink audio transmitters
PLCF	ℓ/c filter power output
PPCU	Power control unit power output
PPC	Power conditioning unit power output
PB	Power out of the battery
PPT	Input power to power conditioning unit
PPSD	Prime power supply requirement (degraded)
PPS	Prime power supply (before degradation)
WHRS	Battery watt hours requirement
WPT	Weight of the power transfer unit
WHAR	Weight of the harnessing
WPCU	Weight of the power control unit
WPC	Weight of the power conditioning unit
WLCF	Weight of the ℓ/c filter
WCOND	Total weight of the power subsystem
VPT	Volume of the power transfer unit
VPCU	Volume of the power control unit

VPC	Volume of the power conditioning unit
VLCF	Volume of the ℓ/c filter
VCOND	Total volume of the power subsystem
CPT1D	Unit recurring cost of power transfer unit
CHAR1D	Unit recurring cost of harnessing equipment
CPCU1D	Unit recurring cost of power control unit
CPC1D	Unit recurring cost of power conditioning unit
CLCF1D	Unit recurring cost of ℓ/c filter
CLCF2D	Non-recurring cost of ℓ/c filter
CPT2D	Non-recurring cost of power transfer unit
CHAR2D	Non-recurring cost of harnessing
CPCU2D	Non-recurring cost of power control unit
CPC2D	Non-recurring cost of power conditioning unit
COND1D	Total unit recurring cost of power subsystem
COND2D	Total non-recurring cost of power subsystem
AREA	Area required by power supply
PRDT	Total radiated power
WTR	Weight of downlink video transmitters
WTRP	Weight of downlink audio transmitters
VTR	Volume of downlink video transmitters
VTPR	Volume of downlink audio transmitters
WPP, VPP	Weight and volume of power supply
WBATT, VBATT	Weight and volume of batteries
VEM	Total equipment module volume
XLEM	Length of the equipment module
VSTRCT	Structure volume
WSTRCT	Structure weight
AEFF	Effective area
PRDEM	Radiated power of equipment module
PRD	Power radiated
ARD	Area required
WRD	Weight of thermal control system
VRD	Volume of thermal control system

YM, YL	Variable used in determining attitude control elements, function of concept
WEM	Weight of the equipment module
WTOT	Total weight requirement
VTOT	Total volume requirement
CE	Non-recurring cost for the downlink video transmitters
CEPP	Non-recurring cost for the downlink audio transmitters
CMFG	Unit recurring cost for the downlink video transmitters
CMFGPP	Unit recurring cost for the downlink audio transmitters
Y(5)	Power output/channel (ground uplink)
CTL	Cost of receiver transmission line (ground downlink)
GR	Learning curve coefficients
SETDNS	Density of TV sets (sets/mi. ²) (for CATV distribution)
CINREC	Receiver installation cost
CINTL	Transmission line installation cost
CINDIS	Installation cost of cable distribution equipment
CINANT	Antenna installation cost
CSTPYR	Total cost/year/beam

The individual cost elements for the four subsystems are given by the following storage vectors:

UREC1	Unit recurring costs, ground uplink system
NREC1	Non-recurring costs, ground uplink system
ACQC1	Acquisition costs, ground uplink system
OPC1	Operations costs, ground uplink system
UREC2	Unit recurring costs, spacecraft system
NREC2	Non-recurring costs, spacecraft system
ACQC2	Acquisition costs, spacecraft system
OPC2	Operations costs, spacecraft system
UREC3	Unit recurring costs, launch vehicle
NREC3	Non-recurring costs, launch vehicle
ACQC3	Acquisition costs, launch vehicle
OPC3	Operations costs, launch vehicle
UREC4	Unit recurring costs, ground downlink system
NREC4	Non-recurring costs, ground downlink system
ACQC4	Acquisition costs, ground downlink system
OPC4	Operations costs, ground downlink system
TUREC(1),..., TUREC(4)	Total unit recurring cost of each subsystem
TNREC(1),..., TNREC(4)	Total non-recurring cost of each subsystem
TACQC(1),..., TACQC(4)	Total acquisition cost of each subsystem
TOPC(1),..., TOPC(4)	Total operations cost of each subsystem

See Appendix D for cost model.

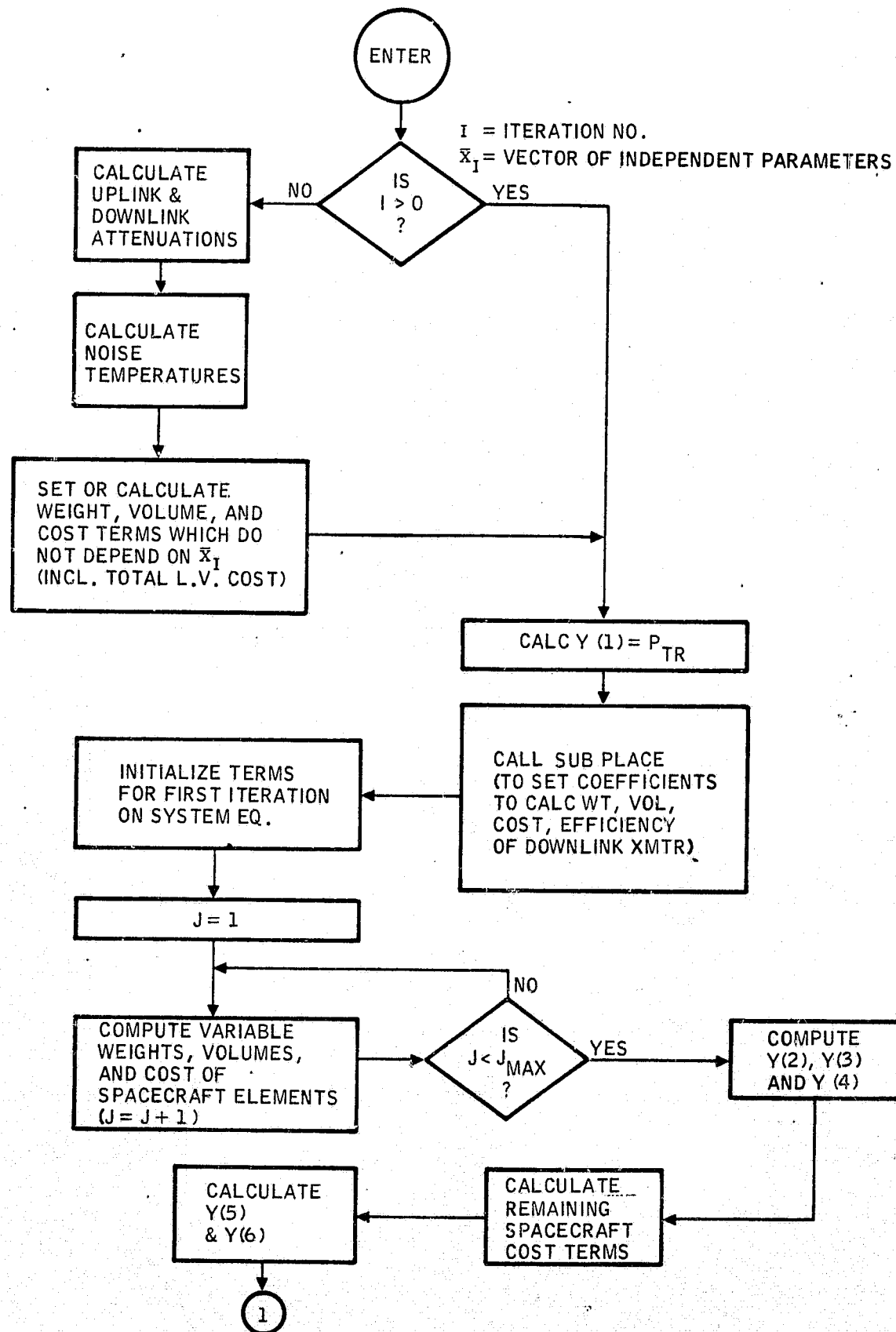


Figure 5.5 Flow Diagram - Subroutine SYSEQ

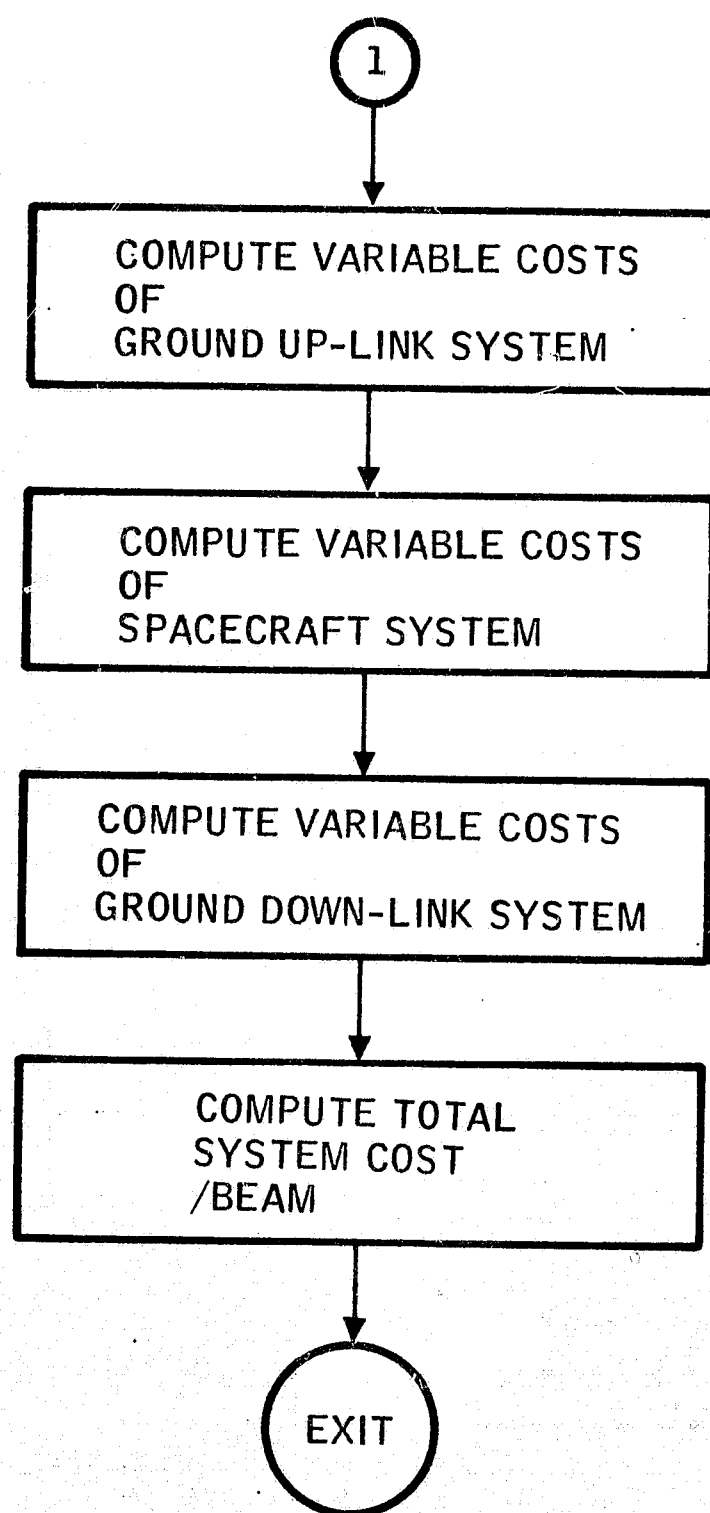


Figure 5.5 Flow Diagram - Subroutine SYSEQ - Continued

```

SUBROUTINE SYSEQ (X,Y,ITER,N,M,L)
COMMON/ATTN/ BVEST,BBB(2),ZMOD(2),XLAT,XLONG,RP,BT1(2),BT2(4)
* ,AT1(2),AT2(4),KATTN1(2),KATTN2(4),TZ1(2),TZ2(4),XLI,XLT1,XLT2
* ,XLT3,XLT4,XPT1,XPT2,RPT1,RPT2,P1,P2,COSNT(4),TION,TATM,TCLOUD
* ,TRAIN,TTL,AMMT(3,4),TE,TRCVRS,GE1,GE2,GS1,GS2,GM1,GM2,GSX(3,3)
* ,GEX(3,3),GMX(3,3),AMTPLX(4),BMTPLX(4),TCO,TI,TA,TC,TM,TR,TT
* ,TEFF,FSL12,FSL34,ATL1,ATL2,XCIRC1,XCIRC2,RCIRC1,RCIRC2
* ,XL12,XL34,KGS,KGM,KGE,RPP(3),BETA,GAMMA,EL,RSL
COMMON/BOUND/IARRAY(15),JARRAY(15),YMIN(15),YMAX(15)
COMMON/CNST/ PI,R,CVEL,BOLZ,RSAT,RE
COMMON/ CNVRGE/ EPSLON,MCONV,ITMAX,PSIK,GRAD(5)
COMMON/CONST/KRA,KIA,KUA,KA,KL,KCLE,KWPCU,KVPLCF,KVPT,KPCN,KVCU
* ,KWPLCF
COMMON/COST/ UREC1(7),NREC1(6),ACQC1(3),OPC1(10),UREC2(15)
* ,NREC2(20),ACQC2(4),OPC2(1),UREC3(2),NREC3(5)
* ,ACQC3(4),OPC3(1),UREC41(2),NREC41(3),ACQC41(2)
* ,OPC41(1),UREC42(7),NREC42(7),ACQC42(2),OPC42(8)
* ,UREC43(7),NREC43(5),ACQC43(2),OPC43(7),TUREC(4)
* ,TNREC(4),TACQC(4),TOPC(4)
COMMON/ DIAM/ D(4),G(4)
COMMON/DISCR/ GE,GS,GM
COMMON/MAIN/ I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15
* ,I16,I17,I18,I19,I20,I21,I22,I23,I24,I25
* ,XQ,XJ,XK,XL,XP1,XM,XP2,XN,XP3,XY,XP4
COMMON/MISC/ BLEM,BVEM,XLF1,XLF2,XLF,XLA,SNRG,XTA,XIMPRV(2),BVID12
* ,BVID34,TSTRCT,AMAJR,BMINR
COMMON/NUMB/ NTR,NCTR,NF,LSYS,NSL,NRS,NLV,NRLV,NRSTA,NB,NCB,NC
* ,NCTRF,NTRF,NCF,LSAT,FREQ1,GR85,GR89,GR95,GR,XNGR,XNGRP
* ,NAPV,NAUDC,PDNSE,ITASO,NBRQD,AOC,NSAT,NBS,SETDNS,JIT
COMMON/PARA/ PLOG,ETA,ARD,VTR,WTR,FREQ4,XNS,RHO,AKP,XNG
* ,CE,CMFG,WEM,WLOG,PGLOG,CSTPYR,G4LOG,TLOG
* ,VEM,WSTLOG,PSLOG,WHLOG,PRLOG,D1LOG
* ,D4LOG,PTALOG,PCULOG,PPCLOG,PPTLOG,CEPP,CMFGPP
COMMON/PARA2/WPP,VPP,PPS,WA,VA,WUPLA,VUPLA,WREC,VREC,WSTRCT,VSTRCT
* ,WMPLX,VMPLX,WRD,VRD,WCS,VCS,WAC,VAC,PRD,WINC,VINC
* ,WBATT,VBATT,CMPLX,CREC,CPP1D,CPT1D,CPCU1D,CLCF1D
* ,CPC1D,CPP2D,CPT2D,CPCU2D,CLCF2D,CPC2D,CBTT1D
* ,CBTT2D,PTA,PPT,PPCU,PPC,WTRP,VTRP,VPCU,WPCU,VPC,WPC
* ,ETAP,ETALC,CTL,UCOSTR,COND1D,COND2D,WCOND,VCOND,PLCF
* ,VPT,CHAR1D,CHAR2D,CINANT,CINREC,CINTL,CINDIS,CABL
COMMON/PARA3/ YM,YL,AML,AREA,WHRS,WS,XLEM
COMMON/REDUND/ LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF,RRWTR,RRWVR
* ,RRWPT,RRWBTT,RRWPC,RRWPCU,RRWLCF,RRVTR,RRVVR,RRVPT,RRVBTT
* ,RRVPC,RRVPCU,RRVLCF,RRCTR,RRCRVR,RRCP,RRCBTT,RRPC,RRPCU
* ,RRCLCF,VRHO,WRTR,WRCVR,WRPT,WRBTT,WRPC,WRPCU,WRLCF,VRTR,VRCVR
* ,VRPT,VRBTT,VRPC,VRPCU,VRLCF,CRTR,CRCVR,CRPT,CRBTT,CRPC,CRPCU
* ,CRLCF
COMMON/SYS/ WMP,VMP,WGHTB(3,3),VOLB(3,3),XTR(3),ANTC(3),CSTR(3,3)
* ,CP1(3,2),CP2(3,2),RAD1(3,2),RAD2(3,2),SK1(2,2,3),SK2(2,2,3)
* ,PLX1(4,3),PLX2(4,3),DNLA,ANTD1(3,4),ANTD2(3,4),CBT1(3,3)
* ,CBT2(3,3),RECH(3),RECH1(3,3,4),TRBD1(2,4),RECH2(3,3),TRBD2(2)
* ,RECA1,RECA2,ANTH(3,4),XINST1,XINST2,CINST(3,4),CRCUR(3,2)
* ,CRCNR(3,2),NOANT(8),DCYCLE,ARCV(4),ETAPT,ETAPC,ETAB,ETAPCU
* ,ETALCF,RHOPC,DGRD(2),WPTR(3) ,WCU1,WCU2,WCU(3,3),WPCN(3)
* ,WLCFP1,WLCFP(2,2),VPTR1,VPTR(3,2),VCU1,VCU(3,2),VPCN1

```

```

* ,VPCN(3,2),VLCFP1,VLCFP(2,2),CENGP1,CENGP2,CMFGP1,CENGPC(3,3)
* ,CMFGPC(3,2),CLC1,CLC2,CENGLC(2,3),CMFGLC(2,3),XCNST
* ,DEGRAD,CPTUR(3),CPTNR(3),CPCUUR(3),CPCUNR(3),CPCUR(3),CPCNR(3)
* ,CBETA,SBETA,IETBD1,IETBD2,WMUX(3,3),WHARR(3,8),CHARUR(3)
* ,CHARNR(3),NPROTO,SKLS1(2,2,3),SKLS2(2,2,3),ATT1(2,2,3)
* ,ATT2(2,2,3),ATLS1(2,2,3),ATLS2(2,2,3),IAML,IWS,AMLLOG
* ,DFMREC,PARANT(4),NRGR,ACQLV(8),ENGLV(8),ACQMP(8),ENGMP(8)
* ,URCBSM,ENGBSM,URCMDA,ENGMDA
* ,RCVR1(2),RCVR2(1),HRCVR1(3,4),HRCVR2(3,2),XLC1(3,2),XLC2(3,1)
* ,XLC3(3,1),RCVC2,RCVC3,DELC1,DELC2
COMMON/SYSTEM/ AW(3),AV(3),AET(3),ACE(3),ACMFG(3),AWP(3),AVP(3)
* ,AETP(3),ACEP(3),ACMFGP(3),IVLEV(4),ICLEV(4)
COMMON/TABLE/ AWGHT(3,4,3),AVOL1(3,4,3),AVOL2(3,4,3),ACENG1(3,4,3)
* ,ACENG2(3,4,3),ACENG3(3,4,3),ACMFG1(3,4,3)
* ,ACMFG2(3,4,3),AETA(3,4,3),FWGHT(3,4,3),FVOL1(3,4,3)
* ,FVOL2(3,4,3),FCENG1(3,4,3),FCENG2(3,4,3)
* ,FCENG3(3,4,3),FCMFG1(3,4,3),FCMFG2(3,4,3)
* ,FETA(3,4,3),PSPPLY(4,2),ANT(4,2)
* ,FBOOM(4,3),BCHANN(2),DABD(8),BLNGTH(8)
* ,BVOL(8),BWGHT(8),DEM(8),FREQ(4),BDWDTH(4)
* ,UPFREQ(2),EXPR,FDLIN
COMMON/TRANS/ AEFF,PB,PRDEM,PRDT,XLMUX,PR,PCS,PAC,PINC,PMP
* ,XLA34,XLC34,XLR34,XLRH34
REAL NREC1,NREC2,NREC3,NREC41,NREC42,NREC43,K1,K2,NOANT
REAL KATTN1,KATTN2,LSYS,LSAT,NBRQD,NLV,NTR,NAPV,NAUDC,NCF
REAL LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF
REAL NSAT,NBS,NPROTO
REAL NRGR
DIMENSION X(1),Y(1),LBATT(3),SIGD(2)
DIMENSION PACTMP(2),WACTMP(2),VACTMP(2),PCSTMP(2),WCSTMP(2)
* ,VCSTMP(2)
DATA SIGD(1),SIGD(2) / 2.E-04, 1.E-04/
QUAD(A,B,C,D) = A + B*D + C*D**2
NBS = NOANT(I1)
IF(ITER.GT.0) GO TO 5000
PACTMP(1) = PACTMP(2) = 0.
WACTMP(1) = WACTMP(2) = 0.
VACTMP(1) = VACTMP(2) = 0.
PCSTMP(1) = PCSTMP(2) = 0.
VCSTMP(1) = VCSTMP(2) = 0.
WCSTMP(1) = WCSTMP(2) = 0.
***** ATTENUATION AND NOISE MODEL *****
... ATTENUATIONS ...
XNC = NC
XPT1 = 10.**(XPT1/10.)
XPT2 = 10.**(XPT2/10.)
RPT1 = 10.**(RPT1/10.)
RPT2 = 10.**(RPT2/10.)
P1 = 10.**(P1/10.)
P2 = 10.**(P2/10.)
SEL = SIN(EL)
CEL = COS(EL)
COEF = 41.4 - 23.5 * ALOG10(RP)
EXP1 = AT1(I24)
EXP2 = AT2(I3)

```

C
C

```

ELP = ATAN(2. / COEF)
IF(EL.LT.ELP) GO TO 400
XLR12 = 2. * BT1(I24) * RP**EXP1 / SEL
XLR34 = 2. * BT2(I3) * RP**EXP2 / SEL
GO TO 410
400 CONTINUE
XLR12 = BT1(I24) * COEF * RP**EXP1 / CEL
XLR34 = BT2(I3) * COEF * RP**EXP2 / CEL
410 XLR12 = 10.**(-XLR12/10.)
XLR34 = 10.**(-XLR34/10.)
XLRH34 = BT2(I3) * COEF * RP**EXP2
XLRH34 = 10.**(XLRH34/10.)
XLC12 = KATTN1(I24) * 3./SEL
XLC34 = KATTN2(I3) * 3./SEL
XLC12 = 10.**(-XLC12/10.)
XLC34 = 10.**(-XLC34/10.)
EXP = 1. / SEL
XLA12 = (1. - TZ1(I24)/290.):**EXP
XLA34 = (1. - TZ2(I3)/290.):**EXP
FSL12 = 1. / (4.56*1.E+03*RSL**2*(FREQ1*1.E+03)**2)
FSL34 = 1. / (4.56*1.E+03*RSL**2*(FREQ(I3)*1.E+03)**2)
IF(I3.EQ.1) J = I9
IF(I3.NE.1) J = I3 + 1 - I3/4
XLMUX = AMTPLX(J) * XNC **BMTPLX(J)
XLMUX1 = .631
ATL1 = XLI * XLA12 * XLC12 * XLR12
ATL2 = XLI * XLA34 * XLC34 * XLR34
XCIRC1 = XLT1 * XLMUX1
XCIRC2 = XLT3 * XLMUX
RCIRC1 = XLT2
RCIRC2 = XLT4
XL12 = FSL12 * ATL1 * XCIRC1 * RCIRC1 * XPT1 * RPT1 * P1
XL34 = FSL34 * ATL2 * XCIRC2 * RCIRC2 * XPT2 * RPT2 * P2
... NOISE TEMPERATURES ...
XNS = BOLZ * BVID12 * (TE + TRCVRS) * 1.E+09
TCOS = COSNT(I3)
TCO = XLI * XLA34 * XLC34 * XLR34 * TCOS
TI = XLA34 * XLC34 * XLR34 * (1. - XLI) * TION
TA = XLC34 * XLR34 * (1. - XLA34) * TATM
TC = XLR34 * (1. - XLC34) * TCloud
TR = (1. - XLR34) * TRAIN
TT = (1. - XLT4) * TTL
EXP = AMMT(3,I3)
TMM = AMMT(1,I3) + AMMT(2,I3) * PDNSE**EXP
TM = XLRH34 * TMM
WMP = VMP = PMP = 0.
IF(I22.EQ.0) GO TO 515
GO TO (515,515,515,511,512,513,515,515),I1
511 PMP = 3000.
WMP = 6983.
GO TO 515
512 WMP = 3200.
GO TO 515
513 PMP = 3000.
WMP = 5500.

```

ATMOS

XMTR-CIR

RCVR-CIR

COSMIC

IONOSP
ATMOSPH
CLOUD
RAIN
TRAN LIN
MAN-MADE

515 CONTINUE

WCOND = 0.

VCOND = 0.

DEGRD = DGRD(1) + DGRD(2) * ALOG10(LSAT)

PAC = WAC = VAC = 0.

PCS = WCS = VCS = 0.

ADEGRD = EXP((12.8*SIGD(I4)*D(3)*FREQ(I3))**2)

G(3) = 5.62 * (FREQ(I3) * D(3))**2 / ADEGRD

I26 = 2

IF(I1.GT.2) GO TO 516

D(2) = D(3)

I26 = I4

516 ADEGRD = EXP((12.8*SIGD(I26)*D(2)*FREQ(I1))**2)

G(2) = 5.62*(FREQ(I1)*D(2))**2 / ADEGRD

MNC = (NCB + 3) / 4

C

... INSTRUMENTATION AND COMMUNICATION ...

PINC = 166.

WINC = 70.0

VINC = 2. * VRHO

XJ = PSPPLY(1,I6)

XJC = PSPPLY(2,I6)

XK = PSPPLY(3,I6)

XKC = PSPPLY(4,I6)

ETALC = 1.0

IF(I2.EQ.1) ETALC = ETALCF

XCNST = 1.

IF(I2.EQ.1) XCNST = .32

C

... RECEIVER ...

WREC = 14. * MNC * WRCVR * NBS

VREC = .25 * MNC * VRCVR * VRHO * NBS

PR = 10. * MNC * NBS

C

... MULTIPLEXER ...

IF (NCB.EQ.1) GO TO 1

EXPP= WMUX(2,I9)*NCB

WMPLX=WMUX(1,I9)*10.**EXPP*NBS

GO TO 2

1 WMPLX=WMUX(3,I9)*NBS

2 CONTINUE

VMPLX = .1 * VRHO*NBS

C

... ANTENNA SUBSYSTEMS ...

IF (I1.GT.2) GO TO 4

WUPLA = 0.

VUPLA = 0.

GO TO 6

4 CONTINUE

WUPLA = 6.4 * D(2) **.5

VUPLA = .786 * (.2 * D(2) + .25) * D(2) **2

6 GO TO (10,20,30,40,50,60,80,30), I1

10 CONTINUE

WA = XL * D(3) **XP1

WFB = XN * D(3) **XP3

WA = WA + WFB

VA = 19.65 * (XLF + XLA)

GO TO 80

20 CONTINUE

```

WA = XL * D(3) **XP1
WFB = XN * D(3) **XP3
WA = WA + WFB
VA = 34.9 * (XLF + XLA)
GO TO 80
30 CONTINUE
WA = .1 * D(3) **2 + .32 * D(3) **1.69
VA = .0028 * D(3)**3 + .786 * XLF1 * (.1 * D(3) + 1.)**2
GO TO 80
40 CONTINUE
WA = NOANT(I1) * (6.4*D(3)**.5 + .32*D(3)**1.69)
VA = NOANT(I1) * (.786 * XLF1 * D(3)**2)
GO TO 80
50 CONTINUE
WA = .1 * D(3) **2 + .32 * D(3) **1.69
VA = .0028 * D(3)**3 + .00016 * D(3)**3
GO TO 80
60 CONTINUE
IF(NBS.LT.6.) GO TO 61
WA = NBS * 0.1 * D(3)**2 + 850.
DTOT = 2. * D(3) + 34.
VA = 0.0028 * DTOT**3 + 170.
GO TO 80
61 WA = NBS * (0.1 * D(3)**2 + 0.32 * D(3)**1.96)
VA = NBS * (0.0028 * D(3)**3 + 0.039 * D(3)**2)
80 CONTINUE
UREC1(1) = .0066 * NCF + .041
UREC1(2) = .07 * NCF
IF(I19.EQ.2) UREC1(2) = .12 * NCF
UREC1(5) = 0.242 / NF
UREC1(6) = .000915 * NCF + .03275
UREC1(7) = .2
NREC1(2) = 0.
NREC1(3)=0.
NREC1(4)=0.
NREC1(5) = 0.195 / NF
OPC1(1) = .02 * UREC1(1)
OPC1(2) = .15 * UREC1(2)
OPC1(3)=(60.+35.*NCF) * 1.E-03
OPC1(6) = .15 * UREC1(5)
OPC1(7)=.147
OPC1(8) = .05 * UREC1(6)
OPC1(9)=.5*(OPC1(3)+OPC1(7))
KL = 1
IF(I8.GT.3) KL = 2
CREC1 = CRCUR(1,KL) + CRCUR(2,KL)*NCB + CRCUR(3,KL) * MNC
CREC2 = CRCNR(1,KL) + CRCNR(2,KL)*NCB + CRCNR(3,KL) * MNC
D2LOG = 10. * ALOG10(D(2))
D3LOG = 10. * ALOG10(D(3))
IF (NCB.EQ.1) GO TO 710
J = I9
XNCB = NCB
XPLX1 = QUAD(PLX1(1,J),PLX1(2,J),PLX1(3,J),XNCB)
XPLX2 = QUAD(PLX2(1,J),PLX2(2,J),PLX2(3,J),XNCB)
CMPLX1 = 10. *(XPLX1 / 10.) *NBS

```

FACIL
PRC-EQ

STBY-PWR
TEST-EC

```

CMPLX2 = 10. *(XPLX2 / 10.)
GO TO 715
710 CMPLX1 = PLX1(4,J) *NBS
CMPLX2 = PLX2(4,J)
715 CONTINUE
KA = 2 * I4 - 1
IF(D(3).GT.DNLA) KA = KA + 1
CANT1 = QUAD(ANTD1(1,KA),ANTD1(2,KA),ANTD1(3,KA),D3LOG)
CANT2 = QUAD(ANTD2(1,KA),ANTD2(2,KA),ANTD2(3,KA),D3LOG)
CANTD1 = 10. *(CANT1 / 10.) * NBS
CANTD2 = 10. *(CANT2 / 10.)
I23 = I4
CUPL1 = CUPL2 = 0.
IF(I1.LE.2) GO TO 900
I23 = 2
KUA = 3
IF(D(2).GT.DNLA) KUA = 4
CUP1 = QUAD(ANTD1(1,KUA),ANTD1(2,KUA),ANTD1(3,KUA),D2LOG)
CUP2 = QUAD(ANTD2(1,KUA),ANTD2(2,KUA),ANTD2(3,KUA),D2LOG)
CUPL1 = 10. *(CUP1/10.)
CUPL2 = 10. *(CUP2/10.)

```

```

900 CONTINUE
UREC2(7) = 0.
UREC2(8) = 0.850
UREC2(9) = CREC1 * CRCVR * NBS
UREC2(10) = CUPL1
UREC2(12) = CMPLX1
UREC2(13) = CANTD1
NREC2(7) = 0.
NREC2(8) = 0.170
NREC2(9) = CREC2
NREC2(10) = CUPL2
NREC2(13) = CANTD2
NREC2(19) = 0.
NREC2(20) = .75

```

LIFE-SI

UPL-REC
UPL-ANT
MTPLX
DNL-ANT

... LAUNCH VEHICLE SYSTEM ...

```

C
ACQC3(4) = ACQLV(I1)
TNREC(3) = ENGLV(I1)
NSL = 1
IF(I1.EQ.1)NSL=13
IF(I1.EQ.2)NSL=7
IF(I1.EQ.3) NSL = 3
TACQC(3)=NLV*ACQC3(4)
TOPC(3) = 0.
5000 CONTINUE
D(1) = X(1)
D(4) = X(2)
ADEGRD = EXP((12.8*SIGD(2)*D(1)*FREQ1)**2)
G(1) = 5.26 * (FREQ1 * D(1)) ** 2 / ADEGRD
ADEGRD = 1.
IF(I16.EQ.2) ADEGRD = EXP((12.8*SIGD(2)*D(4)*FREQ(I3))**2)
G(4) = 5.26 * (FREQ(I3) * D(4))** 2 / ADEGRD
G4LOG = 10. * ALOG10(G(4))
KG = 1
IF(G4LOG.GT.GS1) KG = 2

```



```

IF(G4LOG.GE.GS2) KG = 3
KGS = KG
GS = QUAD(GSX(1,KG),GSX(2,KG),GSX(3,KG),G4LOG)
GS = 10.**(GS/10.)
KG = 1
IF(G4LOG.GT.GM1) KG = 2
IF(G4LOG.GE.GM2) KG = 3
KGM = KG
GM = QUAD(GMX(1,KG),GMX(2,KG),GMX(3,KG),G4LOG)
GM = 10.**(GM/10.)
KG = 1
IF(G4LOG.GT.GE1) KG = 2
IF(G4LOG.GE.GE2) KG = 3
KGE = KG
GE = QUAD(GEX(1,KG),GEX(2,KG),GEX(3,KG),G4LOG)
GE = 10.**(GE/10.)
GS = GS / 2.42
GE = GE / 2.
GM = GM / 11.5
TEFF = XLT4 * ((TCO+TI+TA+TC+TR)*GS + TM*GM + TE*GE) + TT + X(3)
XNG = BOLZ * BVID34 * TEFF * 1.E+09
Y(1) = (1. + RHO) * SNRG / (G(3) * XL34 * XIMPRV(I2)) * XNG/G(4)
PLOG = 10. * ALOG10(Y(1))
PTALOG = PLOG - 10.
IF(ITER.NE.0) GO TO 999
JFLAG1 = 1
JFLAG2 = 1
GO TO 1000

```

... CHECK PTR LEVEL ...

C

```

999 JFLAG1 = 0
JFLAG2 = 0
IVLEV(2) = 1
ICLEV(2) = 1
IVLEV(4) = 1
ICLEV(4) = 1
IF(PLOG.GT.30.) IVLEV(2) = 2
IF(PLOG.GT.21.14) ICLEV(2) = 2
IF(PLOG.GT.30.) ICLEV(2) = 3
IF(IVLEV(1).NE.IVLEV(2)) JFLAG1 = 1
IF(ICLEV(1).NE.ICLEV(2)) JFLAG1 = 1
IF(PTALOG.GT.30.) IVLEV(4) = 2
IF(PTALOG.GT.21.14) ICLEV(4) = 2
IF(PTALOG.GT.30.) ICLEV(4) = 3
IF(IVLEV(3).NE.IVLEV(4)) JFLAG2 = 1
IF(ICLEV(3).NE.ICLEV(4)) JFLAG2 = 1
1000 IF(JFLAG1.NE.0) CALL PLACE(Y,I2,I7,I3,1,N,M,L)
IF(JFLAG2.NE.0) CALL PLACE(Y,2,I7,I3,2,N,M,L)
IF(ITER.NE.ITERP) GO TO 7000
PAC = PACTMP(1)
PCS = PCSTMP(1)
WAC = WACTMP(1)
WCS = WCSTMP(1)
VAC = VACTMP(1)
VCS = VCSTMP(1)
7000 PACTMP(1) = PACTMP(2)

```

```

PACTMP(2) = PAC
WACTMP(1) = WACTMP(2)
WACTMP(2) = WAC
VACTMP(1) = VACTMP(2)
VACTMP(2) = VAC
PCSTMP(1) = PCSTMP(2)
PCSTMP(2) = PCS
WCSTMP(1) = WCSTMP(2)
WCSTMP(2) = WCS
VCSTMP(1) = VCSTMP(2)
VCSTMP(2) = VCS
IETBD1 = 0
IETBD2 = 0
ETA = QUAD(AET(1),AET(2),AET(3),PLOG)
IF(ETA.GE..15) GO TO 99
ETA = .15
IETBD1 = 1

```

```

99 PLCF = XCNST * Y(1) / ETA

```

C

... POWER SUBSYSTEMS ...

```

PTA = .1 * Y(1)
ETAP = QUAD(AETP(1),AETP(2),AETP(3),PTALOG)
JITR = 0
IF(ETAP.GE..15) GO TO 101
ETAP = .15
IETBD2 = 1
101 PPCU = NC * PLCF/ETALC + NC*NAPV*PTA/ETAP+PAC+PCS+PINC+PR
* + PMP
JITR = JITR + 1
PPC = PPCU / ETAPCU
PB = PPC / ETAPC
PPT = PB * (1. + RHOPC / ETAB)
PPSD = PPT / ETAPT
PPS = PPSD / DEGRAD
WHRS = 45.6 * RHOPC * PB
PCULOG = 10. * ALOG10(PPCU)
PPCLOG = 10. * ALOG10(PPC)
PPTLOG = 10. * ALOG10(PPT)

```

... WEIGHTS ...

```

WPT = WPTR(1) + WPTR(2) * PPT + WPTR(3) * PPT**2
WPT = WPT * WRPT
WHAR=QUAD(WHARR(1,I1),WHARR(2,I1),WHARR(3,I1),PPT)
KWPCU = 1
IF(PPCU.GT.WCU1) KWPCU = 2
IF(PPCU.GT.WCU2) KWPCU = 3
WPCU = QUAD(WCU(1,KWPCU),WCU(2,KWPCU),WCU(3,KWPCU),PCULOG)
WPCU = (10.**((WPCU/10.)) * WRPCU
WPC = QUAD(WPCN(1),WPCN(2),WPCN(3),PPCLOG)
WPC = (10.**((WPC/10.)) * WRPC
KWPLCF = 1
IF(PLCF.GT.WLCFP1) KWPLCF = 2
EXP1= WLCFP(2,KWPLCF)
WLCF = WLCFP(1,KWPLCF) * PLCF **EXP1 * XNC
WLCF = WLCF * WRLCF
IF(I2.EQ.2) WLCF = 0.
WCOND = WPT + WPC + WPCU + WLCF+WHAR

```

C

... VOLUMES ...

```

KVPT = 1
IF(PPT.GT.VPTR1) KVPT = 2
VPT = QUAD(VPTR(1,KVPT),VPTR(2,KVPT),VPTR(3,KVPT),PPTLOG)
VPT = (10.** (VPT/10.)) * VRPT * VRHO
KVCU = 1
IF(PPCU.GT.VCU1) KVCU = 2
VPCU = QUAD(VCU(1,KVCU),VCU(2,KVCU),VCU(3,KVCU),PCULOG)
VPCU = (10.** (VPCU/10.)) * VRPCU * VRHO
KPCN = 1
IF(PPC.GT.VPCN1) KPCN = 2
VPC = QUAD(VPCN(1,KPCN),VPCN(2,KPCN),VPCN(3,KPCN),PPCLOG)
VPC = (10.** (VPC/10.)) * VRPC * VRHO
KVPLCF = 1
IF(PLCF.GT.VLCFP1) KVPLCF = 2
EXP1 = VLCFP(2,KVPLCF)
VLCF = VLCFP(1,KVPLCF) * PLCF **EXP1 * XNC
VLCF = VLCF * VRLCF * VRHO
IF(I2.EQ.2) VLCF = 0.
VCOND = VPT + VPC + VPCU + VLCF

```

... COSTS ...

```

KCE = 1
IF(PPT.GT.CENGP1) KCE = 2
IF(PPT.GT.CENGP2) KCE = 3
CALL MADD(3,1,CPTNR,0.,CENGPC(1,KCE),1.,CPTNR)
CALL MADD(3,1,CHARNR,0.,CENGPC(1,KCE),1.,CHARNR)
KCE = 1
IF(PPCU.GT.CENGP1) KCE = 2
IF(PPCU.GT.CENGP2) KCE = 3
CALL MADD(3,1,CPCUNR,0.,CENGPC(1,KCE),1.,CPCUNR)
KCE = 1
IF(PPC.GT.CENGP1) KCE = 2
IF(PPC.GT.CENGP2) KCE = 3
CALL MADD(3,1,CPCNR,0.,CENGPC(1,KCE),1.,CPCNR)
KCMFG = 1
IF(PPT.GT.CMFGP1) KCMFG = 2
CALL MADD(3,1,CPTUR,0.,CMFGPC(1,KCMFG),1.,CPTUR)
CALL MADD(3,1,CHARUR,0.,CMFGPC(1,KCMFG),1.,CHARUR)
KCMFG = 1
IF(PPCU.GT.CMFGP1) KCMFG = 2
CALL MADD(3,1,CPCUUR,0.,CMFGPC(1,KCMFG),1.,CPCUUR)
KCMFG = 1
IF(PPC.GT.CMFGP1) KCMFG = 2
CALL MADD(3,1,CPCUR,0.,CMFGPC(1,KCMFG),1.,CPCUR)
CPT1 = QUAD(CPTUR(1),CPTUR(2),CPTUR(3),PPTLOG)
CPT2 = QUAD(CPTNR(1),CPTNR(2),CPTNR(3),PPTLOG)
CHAR1 = QUAD(CHARUR(1),CHARUR(2),CHARUR(3),PPTLOG)
CHAR2 = QUAD(CHARNR(1),CHARNR(2),CHARNR(3),PPTLOG)
CPCU1 = QUAD(CPCUUR(1),CPCUUR(2),CPCUUR(3),PCULOG)
CPCU2 = QUAD(CPCUNR(1),CPCUNR(2),CPCUNR(3),PCULOG)
CPC1 = QUAD(CPCUR(1),CPCUR(2),CPCUR(3),PPCLOG)
CPC2 = QUAD(CPCNR(1),CPCNR(2),CPCNR(3),PPCLOG)
CPT1D = (1. * 10.** (CPT1/10.)) * CRPT
CHAR1D = 10.** (CHAR1/10.) / 3.
CPCU1D = 10.** (CPCU1/10.) * CRPCU / 3.

```

```

CPC1D = 10.** (CPC1/10.) * CRPC
KCLE = 1
IF (PLCF.GT.CLC1) KCLE = 2
IF (PLCF.GT.CLC2) KCLE = 3
CLCF1D = CMFGLC(1,KCLE) + CMFGLC(2,KCLE) * PLCF
CLCF1D = CLCF1D * CRLCF * XNC
CLCF2D = CENGLC(1,KCLE) + CENGLC(2,KCLE) * PLCF
IF (I2.EQ.2) CLCF1D = CLCF2D = 0.
CPT2D = 10.** (CPT2/10.)
CHAR2D = 10.** (CHAR2/10.) / 3.
CPCU2D = 10.** (CPCU2/10.) / 3.
CPC2D = 10.** (CPC2/10.)
COND1D = CPT1D + CPCU1D + CLCF1D + CPC1D+CHAR1D
COND2D = CPT2D + CPCU2D + CLCF2D + CPC2D+CHAR2D
AREA = .1 * PPS
PRDT= PPSD - Y(1) * (NTR*XCNST + 0.1 * NAUDC)*XCIRC2

```

... WEIGHT AND VOLUME ...

... TRANSMITTER ...

```

WC = QUAD(AW(1),AW(2),AW(3),PLOG)
WTR = (NTR * 10.** (WC/10.)) * WRTR
WCP = QUAD(AWP(1),AWP(2),AWP(3),PTALOG)
WTRP = (NAUDC * 10.** (WCP/10.)) * WRTR
V = QUAD(AV(1),AV(2),AV(3),PLOG)
VTR = VRHO * (NTR * 10.** (V/10.)) * VRTR
VP = QUAD(AVP(1),AVP(2),AVP(3),PTALOG)
VTRP = VRHO * (NAUDC * 10.** (VP/10.)) * VRTR

```

... POWER SUPPLY ...

```

WPP = XJ * PPS + XJC
VPP = XK * PPS + XKC

```

... BATTERIES ...

```

WHLOG = 10. * ALOG10(WHRS)
BATT1 = QUAD(WGHTB(1,I20),WGHTB(2,I20),WGHTB(3,I20),WHLOG)
BATT2 = QUAD(VOLB(1,I20),VOLB(2,I20),VOLB(3,I20),WHLOG)
WBATT = (10.** (BATT1/10.)) * WRBTT
VBATT = VRHO * (10.** (BATT2/10.)) * VRBTT
VEM = VTR + VTRP + VREC + VMPLX + VINC + VAC + VCS + VBATT + VCOND
XLEM = 2. * TSTRCT + (VEM / ((PI/4.) * (DEM(I1) - 2. * TSTRCT)

```

* **2))

```

VSTRCT = (PI/4.) * DEM(I1)**2 * 2. * TSTRCT + PI * DEM(I1) *
* TSTRCT * (XLEM - 2. * TSTRCT) * (1. - TSTRCT/DEM(I1))

```

... STRUCTURE ...

```

IF (I1.EQ.6) GO TO 85
IF (I1.NE.4) GO TO 86
WSTRCT = 5700.
AEFF = 313.
VSTRCT = 442.
GO TO 86
85 WSTRCT = 2500.
AEFF = 301.
VSTRCT = 163.
86 CONTINUE
VEM = VEM + VSTRCT
IF ((I1.EQ.4).OR.(I1.EQ.6)) GO TO 87
WSTRCT = PI * DEM(I1) * XLEM
AEFF = 2.5 * (VEM) **.6667

```

87 IF(I1,EQ.4) VMP = 1600. - VEM
IF(I1,EQ.6) VMP = 1300. - VEM

... RADIATOR ...

C

PRDEM = AEFF / 0.0548

PRD = PRDT - PRDEM

IF(PRD,LE.0.) PRD=PRLOG=CRD1=CRD2=UREC2(6)=NREC2(6)=0.

GO TO(88,89),I13

88 ARD = 0.0548 * PRD

WRD = ARD

VRD = ARD / 6.

GO TO 90

89 ARD = 0.009 * PRD

WRD = 1.25 * ARD

VRD = 0.25 * ARD

... ATTITUDE CONTROL ...

C

90 GO TO (180,180,181,182,183,184,185,181), I1

180 YM = .25 * AREA / XLEM + .5 * DEM(I1)

YL = DEM(I1) / 2.

GO TO 185

181 YM = .5 * D(3) + 5.

YL = DEM(I1) / 2.

GO TO 185

182 YM = 30.

YL = 7.5

GO TO 185

183 YM = .5 * (D(3) + XLF + XLA)

YL = D(3) / 2.

GO TO 185

184 IF(NBS.LT.6.) GO TO 186

YM = DTOT/2. + 11.

YL = DTOT / 2.

GO TO 185

186 YL = 5.

YM = 1.35 * D(3) + 11.

IF(NBS.LE.4.) YM = D(3) + 11.

IF(NBS.LE.2.) YM = D(3) / 2. + 11.

185 CONTINUE

AML = AREA * YM / YL

GO TO (190,191,192),I15

... NH3 RESISTOJET ...

C

190 WAC = 60. + (.706 + 1.36 * LSAT) * AML * 1.E-03

VAC = 1. + 2.E-05 * AML * LSAT

PAC = 30. + 3.7E-03 * AML

GO TO 196

... ION PROPULSION ...

C

191 WAC = 16.9*SQRT(7.14E-04*AML + 1.) + 1.37E-04*AML*LSAT + 60.

VAC = 5.97E-06*AML + 2.22E-07*AML*LSAT + 1.332

PAC = 211. + 1.46E-02*AML

GO TO 196

... H2O ELECTROLYSIS ...

C

192 WAC = 60. + 1.37E-03 * AML * LSAT

VAC = 1. + 1.32E-05 * AML * LSAT

PAC = 30. + 3.7E-03 * AML

196 CONTINUE

VAC = VAC * VRHO

WS = WTR + WPP + WREC + WMPLX + WUPLA + WA + WRD + WSTRCT +
 * WAC + WBATT + WINC + WCOND + WTRP + WMP

C ... STATION KEEPING ...

GO TO (211,212,213),I14

C ... AMMONIA RESISTOJET ...

211 WCS = 0.00354 * LSAT * WS / (1. - 0.00197 * LSAT) + 0.0346 * LSAT
 * * WS / (1. - 0.0192 * LSAT)

VCS = 0.045 * WCS

PCS = 4.E-03 * WS

GO TO 217

C ... ION PROPULSION ...

212 WFU = 2.384E-03 * WS * LSAT

WTK = 2. * (3.61 + 0.028 * WFU)**0.667

WCS = 2.82*SQRT(1.+1.43E-03*WS) + 0.658*SQRT(WS) + WFU + WTK

* + 9.16E-05*WS*LSAT

VCS = 2.08E-05*WS + 0.1662 + WFU/984. + WTK/450.

PCS = 181. + 4.469E-02*WS

GO TO 217

C ... H2O ELECTROLYSIS ...

213 WCS = .00216 * LSAT * WS / (1. - .00138 * LSAT) + .0211 * LSAT * WS
 * / (1. - .0135 * LSAT)

VCS = 2.* (.015 * WCS)

PCS = 5.E-03 * WS

217 CONTINUE

VCS = VCS * VRHO

9000 FORMAT(/6H PLCF ,F8.0,3X,5HPAC ,F8.0,3X,5HPCS ,F8.0,3X,5HPPCU ,
 * F8.0,3X,5HPPC ,F8.0/6H PB ,F8.0,3X,5HPPT ,F8.0,3X,5HPPSD ,
 * F8.0,3X,5HPPS ,F8.0,3X,5HJITER, I3)

IF(JITR.LE.JIT) GO TO 101

WEM = WTR + WTRP + WREC + WMPLX + WINC + WAC + WCS + WSTRCT + WBATT

* + WCOND

VEM = VTR + VTRP + VREC + VMPLX + VINC + VAC + VCS + VBATT + VCOND

* + VSTRCT

Y(3) = VEM

WTOT = WEM + WA + WPP + WRD + WUPLA + WMP

Y(2) = WTOT

VTOT = VEM + VA + VPP + VRD + VUPLA + VMP

Y(4) = VTOT

CEP = QUAD(ACE(1),ACE(2),ACE(3),PLOG)

CMFGP = QUAD(ACMFG(1),ACMFG(2),ACMFG(3),PLOG)

CE = 10.**(CEP/10.) * 1.E-06

CMFG=NTR*10.**(CMFGP/10.) * 1.E-06

CEPP = QUAD(ACEP(1),ACEP(2),ACEP(3),PTALOG)

CMFGPP = QUAD(ACMFGP(1),ACMFGP(2),ACMFGP(3),PTALOG)

CEPP= 10.**(CEPP/10.) * 1.E-06

CMFGPP = NAUDC * 10. ** (CMFGPP/10.) * 1.E-06

Y(5) = (1. + RH0) * SNRG/(RH0 * XL12 * XIMPRV(I2)) * XNS /

* (G(1) * G(2))

Y(6) = VPP

C ... GROUND UP-LINK SYSTEM ...

PGLOG = 10. * ALOG10(Y(5))

D1LOG = 10. * ALOG10(X(1))

CXTR = QUAD(XTR(1),XTR(2),XTR(3),PGLOG)

CANT = QUAD(ANTC(1),ANTC(2),ANTC(3),D1LOG)

UREC1(3) = NCF * 10. ** (CXTR/10.)

XMTR

ANTENNA

```

UREC1(4) = 10. ** (CANT / 10.)
TUREC(1) = UREC1(1)+UREC1(2)+UREC1(3)+UREC1(4)+
* UREC1(5)+UREC1(6)+UREC1(7)
NREC1(1) = .10 * UREC1(1)
NREC1(6) = UREC1(3) + UREC1(4) + UREC1(5) + UREC1(6)
TNREC(1) = NREC1(1) + NREC1(2) + NREC1(3) +
* NREC1(4) + NREC1(5) + NREC1(6)
ACQC1(1) = TUREC(1)
ACQC1(2) = .2 * (UREC1(2) + UREC1(3) + UREC1(4) + UREC1(5) +
* UREC1(6))
ACQC1(3) = ACQC1(1) + ACQC1(2)
TACQC(1) = ACQC1(3) * NF
OPC1(4) = .15 * UREC1(3)
OPC1(5) = .08 * UREC1(4)
OPC1(10) = OPC1(1) + OPC1(2) + OPC1(3) + OPC1(4) + OPC1(5) +
* OPC1(6) + OPC1(7) + OPC1(8) + OPC1(9)
TOPC(1) = OPC1(10) * NF * LSYS

```

... SPACECRAFT SYSTEM ...

C

```

WLOG = 10. * ALOG10(WS)
WSTLOG = 10. * ALOG10(WSTRCT)
PSLOG = 10. * ALOG10(PPS)
IF (PRD.EQ.0.) GO TO 218
PRLOG = 10. * ALOG10(PRD)
CRD1 = QUAD(RAD1(1,I13),RAD1(2,I13),RAD1(3,I13),PRLOG)
CRD2 = QUAD(RAD2(1,I13),RAD2(2,I13),RAD2(3,I13),PRLOG)
UREC2(6) = 10. ** (CRD1 / 10.)
NREC2(6) = 10. ** (CRD2 / 10.)
218 STRUC = QUAD(CSTR(1,I12),CSTR(2,I12),CSTR(3,I12),WSTLOG)
CPP1 = QUAD(CP1(1,I6),CP1(2,I6),CP1(3,I6),PSLOG)
CPP2 = QUAD(CP2(1,I6),CP2(2,I6),CP2(3,I6),PSLOG)
ALSTLG = 10. * ALOG10(LSAT)
AMLLOG = 10. * ALOG10(AML)
IAML = 1
IF(AML.GT.5000.) IAML = 2
COEFL1 = ATLS1(1,IAML,I15) + ATLS1(2,IAML,I15) * ALSTLG
COEFL2 = ATLS2(1,IAML,I15) + ATLS2(2,IAML,I15) * ALSTLG
CAC1 = QUAD(COEFL1,ATT1(1,IAML,I15),ATT1(2,IAML,I15),AMLLOG)
CAC2 = QUAD(COEFL2,ATT2(1,IAML,I15),ATT2(2,IAML,I15),AMLLOG)
IWS = 1
IF(WS.GT.5000.) IWS = 2
COEFL1 = SKLS1(1,IWS,I14) + SKLS1(2,IWS,I14)*ALSTLG
COEFL2 = SKLS2(1,IWS,I14) + SKLS2(2,IWS,I14)*ALSTLG
CSK1 = QUAD(COEFL1,SK1(1,IWS,I14),SK1(2,IWS,I14),WLOG)
CSK2 = QUAD(COEFL2,SK2(1,IWS,I14),SK2(2,IWS,I14),WLOG)
CBATT1 = QUAD(CBT1(1,I20),CBT1(2,I20),CBT1(3,I20),WHLOG)
CBATT2 = QUAD(CBT2(1,I20),CBT2(2,I20),CBT2(3,I20),WHLOG)
CPP1D = 10.**(CPP1/10.)
CPP2D = 10.**(CPP2/10.)
CBTT1D = (10.**(CBATT1/10.)) * CRBTT
CBTT2D = 10.**(CBATT2/10.)
UREC2(1) = 10. ** (STRUC / 10.)
IF(I1.EQ.4) UREC2(1) = URCBSM
IF(I1.EQ.6) UREC2(1) = URCMDA
UREC2(2) = CPP1D + COND1D
UREC2(3) = CBTT1D

```

STRUCT

PWR
BATT

```

UREC2(4) = 10.** (CAC1/10.) + 0.2886*LSAT
UREC2(5) = 10.** (CSK1/10.)
UREC2(11) = (CMFG + CMFGPP) * CRTR
NREC2(1) = 3.5 * UREC2(1)
IF(I1,EQ.4) NREC2(1) = ENGBSM
IF(I1,EQ.6) NREC2(1) = ENGMDA
NREC2(2) = CPP2D + COND2D
NREC2(3) = CBTT2D
NREC2(4) = 10.** (CAC2/10.) + 0.85
NREC2(5) = 10.** (CSK2/10.)
NREC2(11) = CE + CEPP
NREC2(12) = CMPLX2
UREC2(14) = 0,
NREC2(14) = 0,
DO 615 I = 1,13
UREC2(14) = UREC2(14) + UREC2(I)
NREC2(14) = NREC2(14) + NREC2(I)

```

615 CONTINUE

```

UREC2(15) = .075 * UREC2(14)
TUREC(2) = UREC2(14) + UREC2(15)
NREC2(15) = NPROTO * (UREC2(14) - 0.85 * CPP1D)
NREC2(16) = .5 * NREC2(14)
NREC2(17) = .05 * (NREC2(14) + NREC2(16))
NREC2(18) = .5 * TUREC(2)
TNREC(2) = 0.

```

DO 175 I = 14,20

TNREC(2) = TNREC(2) + NREC2(I)

175 CONTINUE

```

ACQC2(1) = TUREC(2)
ACQC2(2) = .05 * ACQC2(1)
ACQC2(3) = .10 * ACQC2(1)
ACQC2(4) = ACQC2(1) + ACQC2(2) + ACQC2(3)
TACQC(2) = ACQC2(4) * NSAT
OPC2(1) = TACQC(2)
TOPC(2) = ACQC2(4) * (LSYS / LSAT - 1) * NSAT

```

... GROUND DOWN-LINK SYSTEM ...

D4LOG = 10. * ALOG10(D(4))

... DIRECT-TO-HOME ...

```

TEMR = ABS(10.*ALOG10(XLT4))/FDLINE
CTL = ARCVR(I3) * TEMR ** EXPR * FDLINE * 1.E-06
TLOG = 10.*ALOG10(X(3))
KIA = 2*I16 - 1
KRA = KIA
IF(I16,EQ.2) GO TO 2100

```

... WIDEBAND

```

IF(D(4).GT.RECA1) KRA = KRA + 1
IF(D(4).GT.XINST1) KIA = KIA + 1
GR = GR95
GO TO 2200

```

... PARABOLIC

```

2100 IF(D(4).GT.RECA2) KRA = KRA + 1
IF(D(4).GT.XINST2) KIA = KIA + 1
GR = GR89

```

```

2200 HANT = QUAD(ANTH(1,KRA),ANTH(2,KRA),ANTH(3,KRA),D4LOG)
CPARX = QUAD(CINST(1,KIA),CINST(2,KIA),CINST(3,KIA),D4LOG)

```



```

CINANT = 10.**(CPARX/10.)
IF(I21.GT.1) GO TO 2000
UREC41(1) = 10.**(HANT/10.)
IF(I16.EQ.2) UREC41(1) = PARANT(I3) * UREC41(1)
AXNGR = (XNGR * NSAT * NBRQD) / 3.
IF(AXNGR.LT.1.E+03) AXNGR = 1.E+03
IF(AXNGR.GT.1.E+06) AXNGR = 1.E+06
AGRLOG = 10. * ALOG10(AXNGR)
K = 1
IF(I3.GT.1) GO TO 1045
KREC = 2 * I2 - 1
IF(X(3).GT.RCVR1(I2)) KREC = KREC + 1
CALL MADD(3,1,RECH,0.,HRCVR1(1,KREC),1.,RECH)
GO TO 1060
1045 K = 2
KREC = 1
IF(X(3).GT.RCVR2(1)) KREC = 2
CALL MADD(3,1,RECH,0.,HRCVR2(1,KREC),1.,RECH)
1060 HREC = QUAD(RECH(1),RECH(2),RECH(3),TLOG)
DELC1 = 0.
DELC2 = 0.
IF(((NCB.LE.3).AND.(I2.EQ.1)).OR.((NCB.LE.1).AND.(I2.EQ.2)))
* GO TO 1065
DELC1 = RCVC2 * 10.**(QUAD(XLC2(1),XLC2(2),XLC2(3),AGRLOG)/10.)
IF(((NCB.LE.6).AND.(I2.EQ.1)).OR.((NCB.LE.2).AND.(I2.EQ.2)))
* GO TO 1065
IF(I2.EQ.2) GO TO 1061
IF(MOD(NCB,3).NE.0) XKP = NCB / 3 - 2 + 1
IF(MOD(NCB,3).EQ.0) XKP = NCB / 3 - 2
GO TO 1062
1061 XKP = NCB - 2
1062 CONTINUE
DELC2 = XKP * RCVC3 * 10.**(QUAD(XLC3(1),XLC3(2),XLC3(3),AGRLOG)
* / 10.)
1065 XLC = 10.**(QUAD(XLC1(1,K),XLC1(2,K),XLC1(3,K),AGRLOG) / 10.)
UREC41(2) = 10.**(HREC/10.) * XLC + DELC1 + DELC2
CINREC = 0.
CTL = 0.
TUREC(4) = UREC41(1) * GR + UREC41(2) + CTL * GR95.
UCOSTR = UREC41(2) + CTL * GR95
NREC41(1) = 0.
NREC41(2) = 0.
NREC41(3) = NREC41(1) + NREC41(2)
TNREC(4) = NREC41(1) + NREC41(2) + NREC41(3)
CINTL = CTL * GR95
ACQC41(1) = TUREC(4)
ACQC41(2) = CINANT + CINREC + CINTL
TACQC(4) = (ACQC41(1) + ACQC41(2)) * XNGR
OPC41(1) = .10 * TUREC(4)
TOPC(4)=LSYS*OPC41(1) *XNGR
GO TO 3500
2000 IF(I21.EQ.3) GO TO 3000
C
UREC42(1) = .03
UREC42(2) = 0.

```

...ANT

... RETRANSMISSION. ...

```

UREC42(3) = 0.
UREC42(4) = 0.
UREC42(5) = 0.
UREC42(6) = .05
UREC42(7) = .05
TUREC(4) = UREC42(1) + UREC42(2) + UREC42(3) + UREC42(4) +
* UREC42(5) + UREC42(6) + UREC42(7)
NREC42(1) = .10 * UREC42(1)
NREC42(2) = 0.
NREC42(3) = 0.
NREC42(4) = 0.
NREC42(5) = 0.
NREC42(6) = NREC42(1) + NREC42(2) + NREC42(3) + NREC42(4)
* + NREC42(5)
NREC42(7) = 2. * TUREC(4)
TNREC(4) = NREC42(6) + NREC42(7)
ACQC42(1) = TUREC(4)
ACQC42(2) = .20 * ACQC42(1)
TACQC(4) = (ACQC42(1) + ACQC42(2)) * XNRT
OPC42(1) = .02 * UREC42(1)
OPC42(2) = .15 * (UREC42(4) + UREC42(5))
OPC42(3) = .08 * UREC42(2)
OPC42(4) = .15 * UREC42(3)
OPC42(5) = .05 * UREC42(5)
OPC42(6) = 0.
OPC42(7) = .5 * OPC42(6)
OPC42(8) = OPC42(1) + OPC42(2) + OPC42(3) + OPC42(4) +
* OPC42(5) + OPC42(6) + OPC42(7)
TOPC(4) = OPC42(8) * LSYS * XNRT
GO TO 3500

```

C ... CATV ...

```

3000 CONTINUE
NRGR = NCB
CABL = 2.*SETDNS

```

C ... AM ...
C ... FM ...

```

IF(X(3).GT.TRBD1(1,I3)) GO TO 1070
KREC = 1
GO TO 1075
1070 IF(X(3).GE.TRBD1(2,I3)) GO TO 1072
KREC = 2
GO TO 1075.
1072 KREC = 3
1075 CALL MADD(3,1,RECH,0.,RECH1(1,KREC,I3),1.,RECH)
HREC = QUAD(RECH(1),RECH(2),RECH(3),TLOG)
UREC43(1) = 10.*(HANT/10.)
IF(I16.EQ.2) UREC43(1) = PARANT(I3) * UREC43(1)
UREC43(2) = 10.*(HREC/10.)
IF(I2.EQ.2)UREC43(2) = UREC43(2) + DFMREC
UREC43(2) = UREC43(2) * NRGR
UREC43(3) = 0.0542
UREC43(4) = 0.0015
UREC43(5) = 0.004*CABL
UREC43(6) = CTL
UREC43(7) = UREC43(3) + UREC43(4) + UREC43(5)

```

```

TUREC(4) = UREC43(1) * GR + UREC43(2) * GR85 + UREC43(6) * GR95
* + UREC43(7)
CINREC = UREC43(2) * GR85
CINTL = UREC43(6) * GR95
CINDIS = 0.004 + 2.0E-05*SETDNS*30.
TACQC(4) = (TUREC(4) + CINANT + CINREC + CINTL + CINDIS) * XNGR
OPC43(1) = .15 * UREC43(1) * GR
OPC43(2) = .10 * UREC43(2) * GR85
OPC43(3) = 0.02*UREC43(3)
OPC43(4) = .15 * UREC43(4)
STDNLG = 10.*ALOG10(SETDNS)
EXPN = QUAD(-22.81,0.429,0.00815,STDNLG)
OPC43(5) = 10.** (EXPN/10.)
OPC43(6) = OPC43(3) + OPC43(4) + OPC43(5)
OPC43(7) = OPC43(1) + OPC43(2) + OPC43(6)
TOPC(4) = XNGR * LSYS * OPC43(7)

```

... TOTAL COST ...

C

```

3500 R3 = NBS
IF(NBRQD.GT.NBS) R3 = NBRQD
R1=NBRQD/NSL
R2=NBRQD/NBS
IF (R1.GE.1.) R1=1.
IF (R2.GE.1.) R2=1.
HNREC = TNREC(1) / (NBRQD * NSAT) + TNREC(2) / (NSAT * R3)
* + TNREC(3)*R1*R2/(NSAT*NBRQD) + TNREC(4)
HACQ = TACQC(1) / NBRQD + TACQC(2) / (NBS * NSAT) + TACQC(3) /
* (NBS*NSL) + TACQC(4)
HOPER = TOPC(1) / NBRQD + TOPC(2) / (NBS * NSAT) + TOPC(3) /
* (NBS*NSL) + TOPC(4)
IF(I22.EQ.0) GO TO 299
HNREC = HNREC + ENGMP(I1) / (R3 * NSAT)
HACQ = HACQ + ACQMP(I1) / NBS
HOPER = HOPER + ACQMP(I1)*NSAT*(LSYS/LSAT - 1) / NBS
299 CSTPYR = (HNREC + HACQ + HOPER) / LSYS
DO 300 I = 1,N

Y(M+I) = X(I)
300 CONTINUE
ITERP = ITER
6373 CONTINUE
RETURN
END

```

5.5 SUBROUTINE PLACE

The function of subroutine PLACE is to set the coefficients required to calculate the weight, volume, cost, and efficiency of the downlink transmitters. This is done as a function of transmitter type, modulation, frequency band, and power level.

The method is straightforward and is readily clarified by observing the flow diagram (Figure 5.6) and the listing. A leading character of "A" indicates AM and a leading character of "F" indicates the FM coefficients.

The following explanation will clarify the subscripts notation. Consider for example

AWGHT(I,J,K)

The subscript J indicates a transmitter type and may assume any value from 1 to 4. K indicates frequency band and may assume any value from 1 to 3. (Frequency operating points 3 and 4 use the same coefficients). I assumes values from 1 to 3 and represents the components of the 3 x 1 vector of analytic coefficients for each transmitter frequency combination.

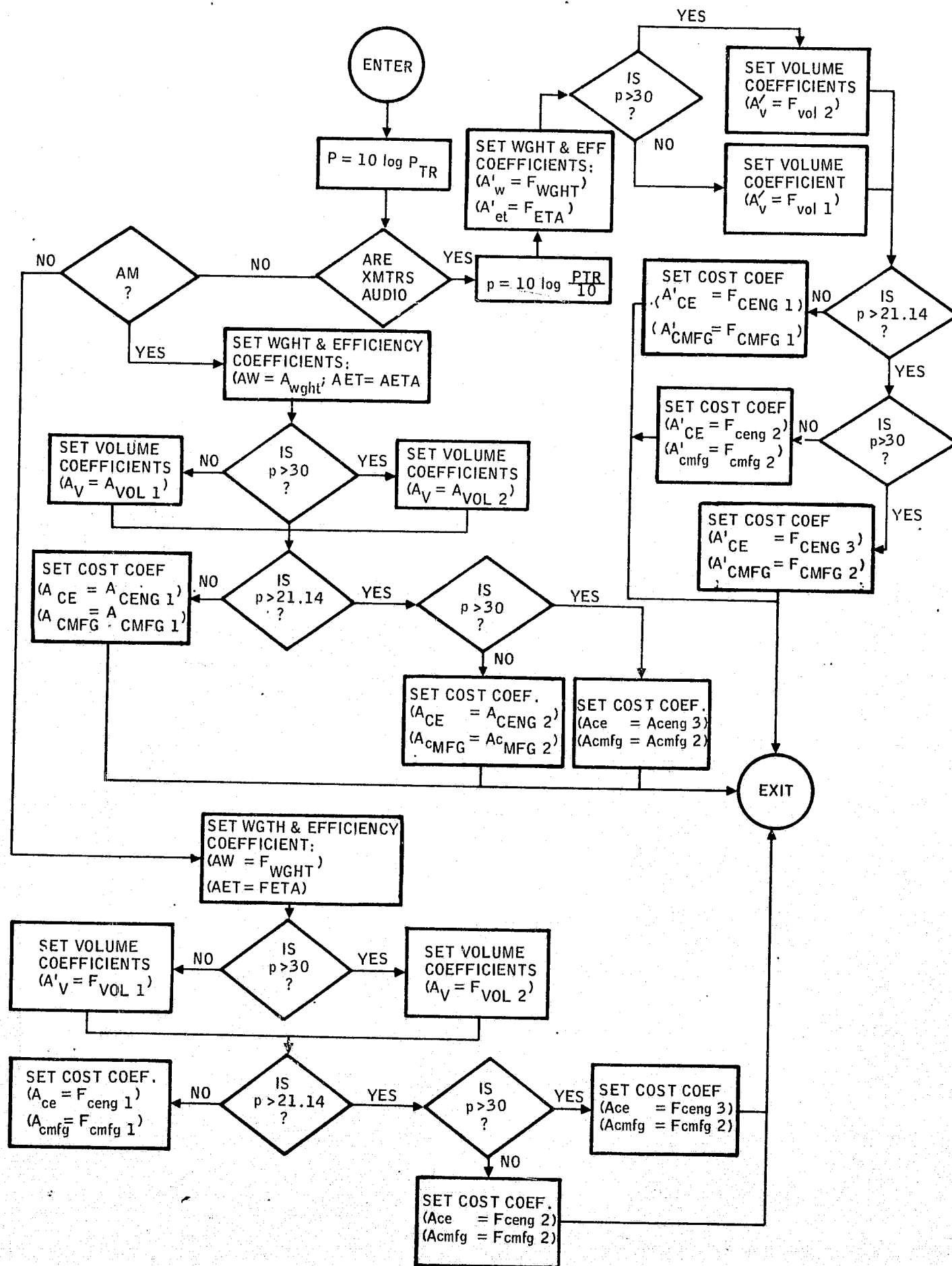


Figure 5.6 Flow Diagram - Subroutine PLACE

```

SUBROUTINE PLACE(Y,IM,IN,IJ,ID,N,M,L)
COMMON/NUMB/ NTR,NCTR,NF,LSYS,NSL,NRS,NLV,NRLV,NRSTA,NB,NCB,NC
* ,NCTRF,NTRF,NCF,LSAT,FREQ1,GR85,GR89,GR95,GR,XNGR,XNGRP
* ,NAPV,NAUDC,PDNSE,ITASO,NBRQD,AOC,NSAT,NBS,SETDNS,JIT
COMMON/SYSTEM/ AW(3),AV(3),AET(3),ACE(3),ACMFG(3),AWP(3),AVP(3)
* ,AETP(3),ACEP(3),ACMFGP(3),IVLEV(4),ICLEV(4)
COMMON/TABLE/ AWGHT(3,4,3),AVOL1(3,4,3),AVOL2(3,4,3),ACENG1(3,4,3)
* ,ACENG2(3,4,3),ACENG3(3,4,3),ACMFG1(3,4,3)
* ,ACMFG2(3,4,3),AETA(3,4,3),FWGHT(3,4,3),FVOL1(3,4,3)
* ,FVOL2(3,4,3),FCENG1(3,4,3),FCENG2(3,4,3)
* ,FCENG3(3,4,3),FCMFG1(3,4,3),FCMFG2(3,4,3)
* ,FETA(3,4,3),PSPPLY(4,2),ANT(4,2)
* ,FBOOM(4,3),BCHANN(2),DABD(8),BLNGTH(8)
* ,BVOL(8),BWGHT(8),DEM(8),FREQ(4),BDWDTH(4)
* ,UPFREQ(2),EXPR,FDLIN
REAL NREC1,NREC2,NREC3,NREC41,NREC42,NREC43,K1,K2,NOANT
REAL KATTN1,KATTN2,LSYS,LSAT,NBRQD,NLV,NTR,NAPV,NAUDC,NCF
REAL LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF
REAL NSAT,NBS
DIMENSION Y(1)
I1 = IM
I2 = IN
I3 = IJ
IF(IJ.EQ.4) I3 = I3 - 1
PLOG = 10. * ALOG10(Y(1))
IF(ID.EQ.2) GO TO 599
IF(I1.EQ.2) GO TO 399

```

... AM ...
... WEIGHT AND STA ...

```

DO 300 I=1,3
AW(I) = AWGHT(I,I2,I3)
AET(I) = AETA(I,I2,I3)
300 CONTINUE

```

... VOLUME ...

```

IF(PLOG.GT.30.) GO TO 310
DO 305 I=1,3
AV(I) = AVOL1(I,I2,I3)
IVLEV(1) = 1
305 CONTINUE
GO TO 320
310 DO 315 I=1,3
AV(I) = AVOL2(I,I2,I3)
IVLEV(1) = 2
315 CONTINUE

```

... MFG. AND ENGINEERING COST ...

```

320 IF(PLOG.GT.21.14) GO TO 330
DO 325 I=1,3
ACE(I) = ACENG1(I,I2,I3)
ACMFG(I) = ACMFG1(I,I2,I3)
325 CONTINUE
ICLEV(1) = 1
GO TO 500
330 IF(PLOG.GT.30.) GO TO 350
DO 335 I=1,3
ACE(I) = ACENG2(I,I2,I3)

```

```

    ACMFG(I) = ACMFG2(I,I2,I3)
335 CONTINUE
    ICLEV(1) = 2
    GO TO 500
350 DO 355 I=1,3
    ACE(I) = ACENG3(I,I2,I3)
    ACMFG(I) = ACMFG2(I,I2,I3)
355 CONTINUE
    ICLEV(1) = 3
    GO TO 500

```

C

... FM ...

```

399 CONTINUE
    DO 400 I=1,3
    AW(I) = FWGHT(I,I2,I3)
    AET(I) = FETA(I,I2,I3)
400 CONTINUE

```

C

... VOLUME ...

```

    IF(PLOG.GT.30.) GO TO 410
    DO 405 I=1,3
    AV(I) = FVOL1(I,I2,I3)
    IVLEV(1) = 1
405 CONTINUE
    GO TO 420.
410 DO 415 I=1,3
    AV(I) = FVOL2(I,I2,I3)
    IVLEV(1) = 2
415 CONTINUE

```

C

... MFG. AND ENGINEERING COST ...

```

420 IF(PLOG.GT.21.14) GO TO 430
    DO 425 I=1,3
    ACE(I) = FCENG1(I,I2,I3)
    ACMFG(I) = FCMFG1(I,I2,I3)
425 CONTINUE
    ICLEV(1) = 1
    GO TO 500
430 IF(PLOG.GT.30.) GO TO 450
    DO 435 I=1,3
    ACE(I) = FCENG2(I,I2,I3)
    ACMFG(I) = FCMFG2(I,I2,I3)
435 CONTINUE
    ICLEV(1) = 2
    GO TO 500
450 DO 455 I=1,3
    ACE(I) = FCENG3(I,I2,I3)
    ACMFG(I) = FCMFG2(I,I2,I3)
455 CONTINUE
    ICLEV(1) = 3
500 CONTINUE
    GO TO 700
599 CONTINUE
    PTALOG = PLOG - 10.

```

C

... FM-AUDIO ...

```

    DO 600 I = 1,3
    AWP(I) = FWGHT(I,I2,I3)
    AETP(I) = FETA(I,I2,I3)

```

600 CONTINUE

C

... VOLUME ...

IF(PTALOG.GT.30.) GO TO 610

DO 605 I=1,3

AVP(I) = FVOL1(I,I2,I3)

605 CONTINUE

IVLEV(3) = 1

GO TO 620

610 DO 615 I=1,3

AVP(I) = FVOL2(I,I2,I3)

615 CONTINUE

IVLEV(3) = 2

C

... MFG. AND ENG. COSTS ...

620 IF(PTALOG.GT.21.14) GO TO 630

DO 625 I=1,3

ACEP(I) = FCENG1(I,I2,I3)

ACMFGP(I) = FCMFG1(I,I2,I3)

625 CONTINUE

ICLEV(3) = 1

GO TO 700

630 IF(PTALOG.GT.30.) GO TO 650

DO 635 I=1,3

ACEP(I) = FCENG2(I,I2,I3)

ACMFGP(I) = FCMFG2(I,I2,I3)

635 CONTINUE

ICLEV(3) = 2

GO TO 700

650 DO 655 I=1,3

ACEP(I) = FCENG3(I,I2,I3)

ACMFGP(I) = FCMFG2(I,I2,I3)

655 CONTINUE

ICLEV(3) = 3

700 CONTINUE

RETURN

END

5.6 SUBROUTINE PRTIAL

The function of subroutine PRTIAL is to compute all of the partial derivatives required by the optimization technique, i.e., the cost gradient and the gradient of the design parameter which violates a boundary (if any).

In computing the partial derivatives, due to the appearance of implicit functions in the system equations, it was necessary to solve two linear equations simultaneously to obtain the values of the derivatives. The solution of these equations is discussed in more detail in Appendix C.

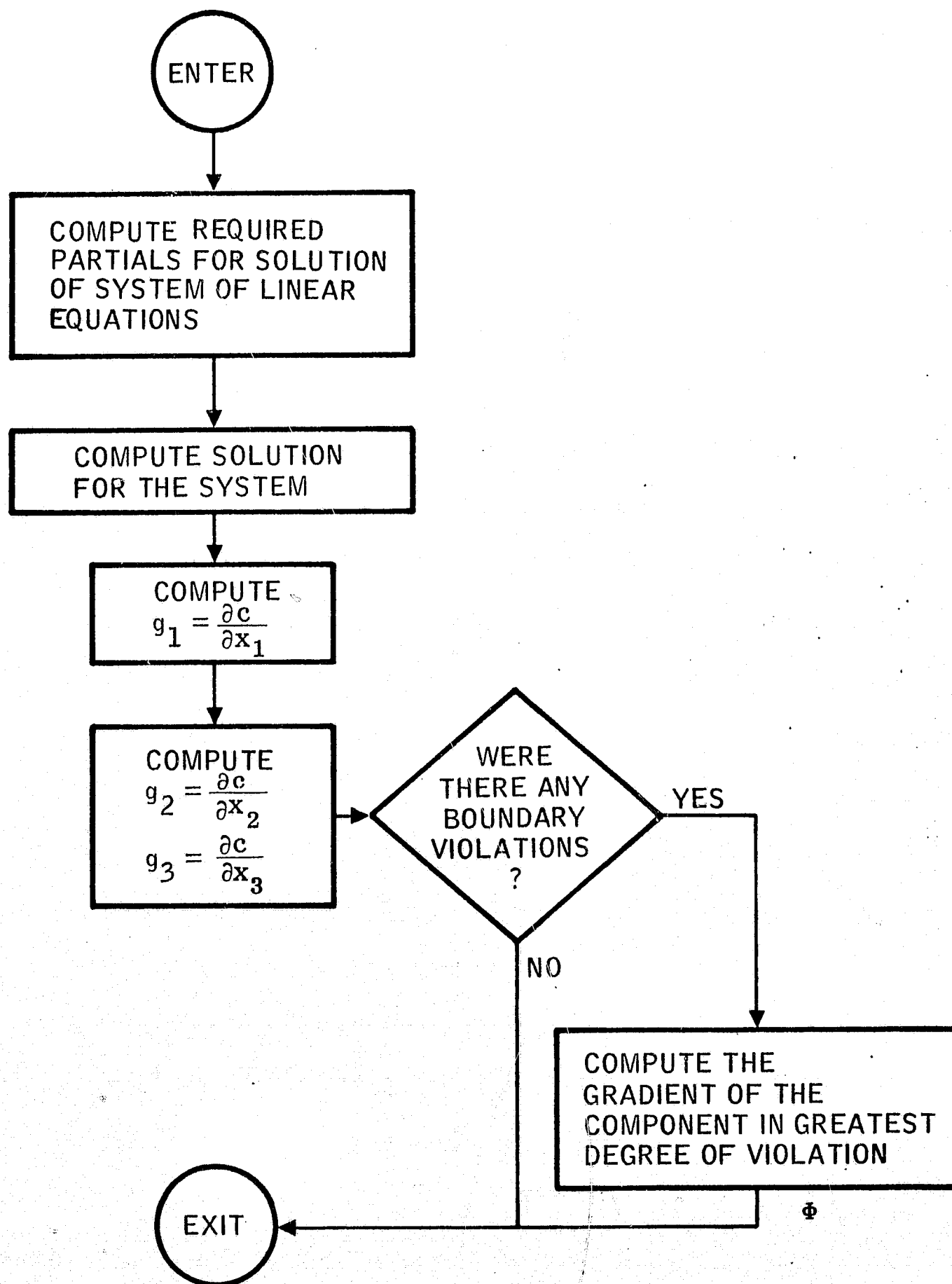


Figure 5.7 Flow Diagram - Subroutine PRTIAL

```

SUBROUTINE PRTIAL(Y,X,PT,MBS,N,M,L,ITER,MB)
COMMON/ATTN/ BVEST,BBB(2),ZMOD(2),XLAT,XLONG,RP,BT1(2),BT2(4)
* ,AT1(2),AT2(4),KATTN1(2),KATTN2(4),TZ1(2),TZ2(4),XLI,XLT1,XLT2
* ,XLT3,XLT4,XPT1,XPT2,RPT1,RPT2,P1,P2,COSNT(4),TION,TATM,TCLOUD
* ,TRAIN,TTL,AMMT(3,4),TE,TRCVRS,GE1,GE2,GS1,GS2,GM1,GM2,GSX(3,3)
* ,GEX(3,3),GMX(3,3),AMTPLX(4),BMTPLX(4),TCO,TI,TA,TC,TM,TR,TT
* ,TEFF,FSL12,FSL34,ATL1,ATL2,XCIRC1,XCIRC2,RCIRC1,RCIRC2
* ,XL12,XL34,KGS,KGM,KGE,RPP(3),BETA,GAMMA,EL,RSL
COMMON/BOUND/IARRAY(15),JARRAY(15),YMIN(15),YMAX(15)
COMMON/CNST/ PI,R,CVEL,BOLZ,RSAT,RE
COMMON / CNVRGE / EPSLON,MCONV,ITMAX,PSIK,GRAD(5)
COMMON/CONST/KRA,KIA,KUA,KA,KL,KCLE,KWPCU,KVPLCF,KVPT,KPCN,KVCU
* ,KWPLCF
COMMON/COST/ UREC1(7),NREC1(6),ACQC1(3),OPC1(10),UREC2(15)
* ,NREC2(20),ACQC2(4),OPC2(1),UREC3(2),NREC3(5)
* ,ACQC3(4),OPC3(1),UREC41(2),NREC41(3),ACQC41(2)
* ,OPC41(1),UREC42(7),NREC42(7),ACQC42(2),OPC42(8)
* ,UREC43(7),NREC43(5),ACQC43(2),OPC43(7),TUREC(4)
* ,TNREC(4),TACQC(4),TOPC(4)
COMMON /DIAM / D(4),G(4)
COMMON/DISCR/ GE,GS,GM
COMMON/MAIN/ I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15
* ,I16,I17,I18,I19,I20,I21,I22,I23,I24,I25
* ,XQ,XJ,XK,XL,XP1,XP2,XP3,XP4
COMMON/MISC/ BLEM,BVEM,XLF1,XLF2,XLF,XLA,SNRG,XTA,XIMPRV(2),BVID12
* ,BVID34,TSTRCT,AMAJR,BMINR
COMMON/NUMB/ NTR,NCTR,NF,LSYS,NSL,NRS,NLV,NRLV,NRSTA,NB,NCB,NC
* ,NCTRF,NTRF,NCF,LSAT,FREQ1,GR85,GR89,GR95,GR,XNGR,XNGRP
* ,NAPV,NAUDC,PDNSE,ITASO,NBRQD,AOC,NSAT,NBS,SETDNS,JIT
COMMON/PARA/ PLOG,ETA,ARD,VTR,WTR,FREQ4,XNS,RHO,AKP,XNG
* ,CE,CMFG,WEM,WLOG,PLOG,CSTPYR,G4LOG,TLOG
* ,VEM,WSTLOG,PSLOG,WHLOG,PRLOG,D1LOG
* ,D4LOG,PTALOG,PCULOG,PPCLOG,PPTLOG,CEPP,CMFGPP
COMMON/PARA2/WPP,VPP,PPS,WA,VA,WAP,VAP,WREC,VREC,WSTRCT,VSTRCT
* ,WMPLX,VMPLX,WRD,VRD,WCS,VCS,WAC,VAC,PRD,WINC,VINC
* ,WBATT,VBATT,CMPLX,CREC,CPP1D,CPT1D,CPCU1D,CLCF1D
* ,CPC1D,CPP2D,CPT2D,CPCU2D,CLCF2D,CPC2D,CBTT1D
* ,CBTT2D,PTA,PPT,PPCU,PPC,WTRP,VTRP,VPCU,WPCU,VPC,WPC
* ,ETAP,ETALC,CTL,UCOSTR,COND1D,COND2D,WCOND,VCOND,PLCF
* ,VPT,CHAR1D,CHAR2D,CINANT,CINREC,CINTL,CINDIS,CABL
COMMON/PARA3/ YM,YL,AML,AREA,WHRS,WS,XLEM
COMMON/REDUND/ LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF,RRWTR,RRWVR
* ,RRWPT,RRWBTT,RRWPC,RRWPCU,RRWLCF,RRVTR,RRVRVR,RRVPT,RRVBTT
* ,RRVPC,RRVPCU,RRVLCF,RRCTR,RRCRVR,RRCP,RRCBTT,RRPC,RRPCU
* ,RRCLCF,VRHO,WRTR,WRCVR,WRPT,WRBTT,WRPC,WPCU,WRLCF,VRTR,VRCVR
* ,VRPT,VRBTT,VRPC,VRPCU,VRLCF,CRTR,CRCVR,CRPT,CRBTT,CRPC,CRPCU
* ,CRLCF
COMMON/SYS/ WMP,VMP,WGHTB(3,3),VOLB(3,3),XTR(3),ANTC(3),CSTR(3,3)
* ,CP1(3,2),CP2(3,2),RAD1(3,2),RAD2(3,2),SK1(2,2,3),SK2(2,2,3)
* ,PLX1(4,3),PLX2(4,3),DNLA,ANTD1(3,4),ANTD2(3,4),CBT1(3,3)
* ,CBT2(3,3),RECH(3),RECH1(3,3,4),TRBD1(2,4),RECH2(3,3),TRBD2(2)
* ,RECA1,RECA2,ANTH(3,4),XINST1,XINST2,CINST(3,4),CRCUR(3,2)
* ,CRCNR(3,2),NOANT(8),DCYCLE,ARCV(4),ETAPT,ETAPC,ETAB,ETAPCU
* ,ETALCF,RHOPC,DGRD(2),WPTR(3) ,WCU1,WCU2,WCU(3,3),WPCN(3)
* ,WLCFP1,WLCFP(2,2),VPTR1,VPTR(3,2),VCU1,VCU(3,2),VPCN1

```

```

* ,VPCN(3,2),VLCFP1,VLCFP(2,2),CENGP1,CENGP2,CMFGP1,CENGPC(3,3)
* ,CMFGPC(3,2),CLC1,CLC2,CENGLC(2,3),CMFGLC(2,3),XCNST
* ,DEGRAD,CPTUR(3),CPTNR(3),CPCUUR(3),CPCUNR(3),CPCUR(3),CPCNR(3)
* ,CBETA,SBETA,IETBD1,IETBD2,WMUX(3,3),WHARR(3,8),CHARUR(3)
* ,CHARNR(3),NPROTO,SKLS1(2,2,3),SKLS2(2,2,3),ATT1(2,2,3)
* ,ATT2(2,2,3),ATLS1(2,2,3),ATLS2(2,2,3),IAML,IWS,AMLLOG
* ,DFMREC,PARANT(4),NRGR,ACQLV(8),ENGLV(8),ACQMP(8),ENGMP(8)
* ,URCBSM,ENGBSM,URCMDA,ENGMDA
* ,RCVR1(2),RCVR2(1),HRCVR1(3,4),HRCVR2(3,2),XLC1(3,2),XLC2(3,1)
* ,XLC3(3,1),RCVC2,RCVC3,DELC1,DELC2
COMMON/SYSTEM/ AW(3),AV(3),AET(3),ACE(3),ACMFG(3),AWP(3),AVP(3)
* ,AETP(3),ACEP(3),ACMFGP(3),IVLEV(4),ICLEV(4)
COMMON/TABLE/ AWGHT(3,4,3),AVOL1(3,4,3),AVOL2(3,4,3),ACENG1(3,4,3)
* ,ACENG2(3,4,3),ACENG3(3,4,3),ACMFG1(3,4,3)
* ,ACMFG2(3,4,3),AETA(3,4,3),FWGHT(3,4,3),FVOL1(3,4,3)
* ,FVOL2(3,4,3),FCENG1(3,4,3),FCENG2(3,4,3)
* ,FCENG3(3,4,3),FCMFG1(3,4,3),FCMFG2(3,4,3)
* ,FETA(3,4,3),PSPPLY(4,2),ANT(4,2)
* ,FBOOM(4,3),BCHANN(2),DABD(8),BLNGTH(8)
* ,BVOL(8),BWGHT(8),DEM(8),FREQ(4),BDWDTH(4)
* ,UPFREQ(2),EXPR,FDLIN
DIMENSION DPTRX(2),DVTRX(2),DWTRX(2),DVLCFX(2),DPLCX(2),DWLCFX(2)
* ,DPTAX(2),DETPX(2),XKS(2),XKP3(2),XK3(2),DPPSX(2),DVEMX(2)
* ,DWSTX(2),DPACX(2),DWSX(2),DPCUX(2),DPPCX(2),DPBX(2),DPPTX(2)
* ,DPRDX(2),DUCSTX(2),DUCPRX(2),DUCBTX(2),DUCSKX(2),DUCRDY(2)
* ,DUCTRX(2),DUCPTX(2),DUCPCX(2),DUCPNX(2),DUCLCX(2),DNCSTX(2)
* ,DNCPRX(2),DNCBTX(2),DNCSTX(2),DNCRDY(2),DNCTRX(2),DNCPTX(2)
* ,DNCPCX(2),DNCPNX(2),DNCLCX(2),DVMPX(2),DWCSX(2),DVCSX(2)
* ,DPCSX(2),DWRDX(2),DPRDMX(2),DUCACX(2),DNCACX(2)
DIMENSION X(1),Y(1),PT(N,M),LBATT(3),SIGD(2),DNV(2),RAD(2)
DATA SIGD(1),SIGD(2) / 2.0E-04, 1.0E-04 /
DATA RAD(1),RAD(2),DNV(1),DNV(2) / 0.0548, 0.01125, 6., 5./
REAL NREC1,NREC2,NREC3,NREC41,NREC42,NREC43,K1,K2,NOANT
REAL KATTN1,KATTN2,LSYS,LSAT,NBRQD,NLV,NTR,NAPV,NAUDC,NCF
REAL LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF
REAL NSAT,NBS,NPROTO
REAL NRGR
XNC = NC
IF((I1.EQ.4).OR.(I1.EQ.6)) GO TO 5000
XSTR = (4. * DEM(I1) * TSTRCT) / (DEM(I1) - 2. * TSTRCT)**2
* * (1. - TSTRCT/DEM(I1))
GO TO 5001
5000 XSTR = 0.
5001 CONTINUE
AKP = 1.
FREQ4 = FREQ(I3)

```

C

```

... PARTIAL OF COST WRT X ...
K1 = NF * (1.2 + .15 * LSYS) + 1.
K2 = NF * (1.2 + .08 * LSYS) + 1.
XNUM = (1. + RHO) * SNRG
PHI1 = XNUM / (G(3) * XL34 * XIMPRV(I2))
PHI2 = XNUM / (RHO * XL12 * XIMPRV(I2))
DG4D4 = 2.0 * G(4) / D(4)
IF(I16.EQ.2) DG4D4 = DG4D4*(1.-(12.8*SIGD(2)*FREQ4*D(4))**2)
DGS64 = GS/G(4) * (GSX(2,KGS) + 2. * GSX(3,KGS) * G4LOG)

```

```

DGMG4 = GM/G(4) * (GMX(2,KGM) + 2. * GMX(3,KGM) * G4LOG)
DGE4 = GE/G(4) * (GEX(2,KGE) + 2. * GEX(3,KGE) * G4LOG)
DTEFG4 = XLT4 * ((TCO+TI+TA+TC+TR)*DGS4 + TM*DGMG4 + TE*DGE4)
DXNGG4 = BOLZ * BVID34 * DTEFG4 * 1.E+09
DG1D1 = 2.*G(1)*(1.-(12.8*SIGD(2)*FREQ1*D(1))**2)/D(1)
DPGD1 = - PHI2*XNS / (G(1)**2 * G(2)) * DG1D1
DPTRX(1) = PHI1 / G(4) * (-XNG/G(4) + DXNGG4) * DG4D4
DPTRX(2) = PHI1 * BOLZ * BVID34 * 1.E+09 / G(4)
IF (I1.GT.2) GO TO 101
DAMLPS = 1. / YL * (.005 * PPS / XLEM + .05 * DEM(I1))
DAMLVM = -.25/YL * (AREA/XLEM)**2 * 4./(PI*(DEM(I1)-2.*TSTRCT)**2)
GO TO 103
101 DAMLPS = .1 * YM / YL
DAMLVM = 0.
103 GO TO (105,106,107) ,I15
105 C1 = (.706 + 1.36 * LSAT) * 1.E-03
C2 = 2.E-05 * LSAT
C3 = 3.7E-03
GO TO 125
106 C1 = 6.0333E-03/SQRT(1. + 7.14E-04*AML) + 1.37E-04*LSAT
C2 = 5.97E-06 + 2.22E-07*LSAT
C3 = 1.46E-02
GO TO 125
107 C1 = 1.37E-03 * LSAT
C2 = 1.32 E-05 * LSAT
C3 = 3.7E-03
125 CONTINUE
C2 = VRHO * C2
DWACPS = C1 * DAMLPS
DWACVM = C1 * DAMLVM
DVACPS = C2 * DAMLPS
DVACVM = C2 * DAMLVM
DPACPS = C3 * DAMLPS
DPACVM = C3 * DAMLVM
DPRDPT = - (NTR*XCNST + 0.1*NAUDC) * XCIRC2
DWRDPS = 0.
DWRDVM = 0.
DWRDPT = 0.
IF (ARD.LE.0.) GO TO 130
DWRDPS = RADDC(I13) * DEGRAD
DWRDVM = - (RADDC(I13)/0.0548) * (5./3.) * VEM**-0.333
DWRDPT = RADDC(I13) * DPRDPT
IF((I1.EQ.4).OR.(I1.EQ.6)) DWRDVM = 0.
130 CONTINUE
DWBTPB = 45.6*RHOPC*WBATT/WHRS * (WGHTB(2,I20) + 2.*WGHTB(3,I20)
* * WHLOG)
DVBTPB = 45.6*RHOPC*VBATT/WHRS * (VOLB(2,I20) + 2. * VOLB(3,I20)
* * WHLOG)
TEMP = (VTR/Y(1)) * (AV(2) + 2. * AV(3) * PLOG)
TEMPP = .1 * (VTRP/PTA) * (AVP(2) + 2. * AVP(3)*PTALOG)
TEMPX = (WTR / Y(1)) * (AW(2) + 2. * AW(3) * PLOG)
TEMPXX = .1 * (WTRP/PTA) * (AWP(2) + 2. * AWP(3) * PTALOG)
DWSTVM = 4. / DEM(I1)
IF((I1.EQ.4).OR.(I1.EQ.6)) DWSTVM = 0.
DVSTVM = XSTR / (1. + XSTR)

```

```

DWPPPS = XJ
DETPTR = (4.3429 / Y(1)) * (AET(2) + 2. * AET(3) * PLOG)
IF(IETBD1.EQ.1) DETPTR = 0.
GO TO (155,156,157),I14
155 CCS1 = 0.00354 * LSAT / (1. - 0.00197 * LSAT) + 0.0346 * LSAT / (1.
* - 0.0192 * LSAT)
CCS2 = 0.045 * CCS1
CCS3 = 4.E-03
GO TO 165
156 WFU = 2.384E-03 * WS * LSAT
DWFUWS = 2.384E-03 * LSAT
DWTKWS = 0.112 / (3.*(3.61 + 0.028*WFU)**0.333) * DWFUWS
CCS1 = 2.0163E-03/SQRT(1.43E-03*WS + 1.) + 0.329/SQRT(WS)
* 9.16E-05 * LSAT + DWFUWS + DWTKWS
CCS2 = 2.08E-05 + DWFUWS/984. + DWTKWS/450.
CCS3 = 4.469E-02
GO TO 165
157 CCS1 = .00216 * LSAT / (1. - .00138 * LSAT) + .0211 * LSAT /
* (1. - .0135 * LSAT)
CCS2 = .03 * CCS1
CCS3 = 5.E-03
165 CONTINUE
CCS2 = VRHO * CCS2

```

C

```

... COST ...
GRAD(1) = K2 * UREC1(4)/D(1) * (ANTC(2) + 2.*ANTC(3)*D1LOG)
* +K1 * UREC1(3)/Y(5) * (XTR(2)+2.*XTR(3)*PGLOG) * DPGD1
GRAD(1) = GRAD(1) / NBRQD
DVPCUP = VPCU/PPCU * (VCU(2,KVCU) + 2. * VCU(3,KVCU) * PCULOG)
DWPCUP = WPCU/PPCU * (WCU(2,KWPCU) + 2. * WCU(3,KWPCU) * PCULOG)
DVPCPC = VPC/PPC * (VPCN(2,KPCN) + 2. * VPCN(3,KPCN) * PPCLOG)
DWPCPC = WPC/PPC * (WPCN(2) + 2. * WPCN(3) * PPCLOG)
DVPTPT = VPT / PPT * (VPTR(2,KVPT) + 2. * VPTR(3,KVPT) * PPTLOG)
DWPTPT = WPTR(2) + 2. * WPTR(3) * PPT
DWPTPT = DWPTPT * WRPT
DWHARP = WHARR(2,I1) + 2. * WHARR(3,I1)*PPT
DWPTPT = DWPTPT + DWHARP
XKT = DWBTPB / (ETAPC*ETAPCU) + DWPTPT * (1.+RHOPC/ETAB)/
* (ETAPC*ETAPCU) + DWPCPC/ETAPCU + DWPCUP
XKP = DVPTPT * (1.+RHOPC/ETAB)/(ETAPC*ETAPCU) + DVPCPC/ETAPCU
* + DVBTPB/(ETAPC*ETAPCU) + DVPCUP
TCNST = (1. + RHOPC /ETAB) / (ETAPT * DEGRAD * ETAPC * ETAPCU)
DWSPPS = DWPPPS + DWACPS + DWRDPS + XKT / TCNST
DWSVM = DWACVM + DWSTVM + DWRDVM
XK1 = 1. - TCNST * DPACPS - CCS3 * TCNST * DWSPPS
XK2 = -TCNST * (DPACVM + CCS3 * DWSVM)
XKP1 = - (DVACPS + CCS2 * DWSPPS + XKP / TCNST)
XKP2 = 1. - (DVACVM + CCS2 * DWSVM + DVSTVM)
DEN = XK1 * XKP2 - XKP1 * XK2
DO 135 I = 1,2
DVTRX(I) = (TEMP + TEMPP) * DPTRX(I)
DWTRX(I) = (TEMPX + TEMPXX) * DPTRX(I)
DPLCX(I) = XCNST/ETA * (1. - DETPTR * Y(1) / ETA) * DPTRX(I)
EXP = VLCFP(2,KVPLCF) - 1.
DVLCFX(I) = XNC * VLCFP(1,KVPLCF) * VLCFP(2,KVPLCF) * PLCF**EXP
* * DPLCX(I)

```

```

DVLCFX(I) = DVLCFX(I) * VRLCF * VRHO
EXP = WLCFP(2,KWPLCF) - 1.
DWLCFX(I) = XNC * WLCFP(2,KWPLCF) * WLCFP(1,KWPLCF) * PLCF**EXP
*      * DPLCX(I)
DWLCFX(I) = DWLCFX(I) * WRLCF
IF(I2.EQ.2) DWLCFX(I) = DVLCFX(I) = 0.0
DPTAX(I) = .1 * DPTRX(I)
DETPX(I) = 4.3429/Y(1)*(AETP(2) + 2. * AETP(3) * PTALOG) *DPTRX(I)
IF(IETBD2.EQ.1) DETPX(I) = 0.
XKS(I) = DPLCX(I)/ETALC * XNC + XNC * NAPV * (DPTAX(I)/ETAP
*      - PTA/ETAP**2 * DETPX(I))
XKP3(I) = DVTRX(I) + CCS2 * (DWTRX(I) + DWRDPT * DPTRX(I)
*      + DWLCFX(I)) + DVLCFX(I)
XK3(I) = TCNST * (XKS(I) + CCS3 * (DWTRX(I) + DWLCFX(I) + DWRDPT
*      * DPTRX(I)))
DPPSX(I) = (XK3(I) * XKP2 - XKP3(I) * XK2) / DEN
DVEMX(I) = - (XK3(I) * XKP1 - XKP3(I) * XK1) / DEN
DVMPX(I) = 0.
IF((I1.EQ.4).OR.(I1.EQ.6))DVMPX(I) = -DVEMX(I)
DWSTX(I) = DWSTVM * DVEMX(I)
DWSX(I) = DWSPPS * DPPSX(I) + DWSVM * DVEMX(I) + DWTRX(I)
DPACX(I) = DPACPS * DPPSX(I) + DPACVM * DVEMX(I)
*      + DWRDPT * DPTRX(I) + DWLCFX(I)
DPCUX(I) = XKS(I) + DPACX(I) + CCS3 * DWSX(I)
DPPCX(I) = DPCUX(I) / ETAPCU
DPBX(I) = DPPCX(I) / ETAPC
DPPTX(I) = (1. + RHOPC/ETAB) * DPBX(I)
DPRDMX(I) = 0.
IF((I1.EQ.4).OR.(I1.EQ.6)) GO TO 134
DPRDMX(I) = (5./3.) * VEM ** -0.333 / 0.0548 * DVEMX(I)
134 DPRDX(I) = DPPSX(I) * DEGRAD - (XNC*XCNST + .1 * NAUDC) * DPTRX(I)
*      * XCIRC2 - DPRDMX(I)

```

C

```

... UNIT RECURRING COSTS ...
DUCSTX(I) = UREC2(1)/WSTRCT *(CSTR(2,I12) + 2.*CSTR(3,I12)*WSTLOG)
*      * DWSTX(I)
DUCPRX(I) = CPP1D/PPS * (CP1(2,I6) + 2.*CP1(3,I6) * PSLOG)
*      * DPPSX(I)
DUCBTX(I) = CBT1D/WHRS * (CBT1(2,I20) + 2.*CBT1(3,I20)*WHLOG)
*      * 45.6 * RHOPC * DPBX(I)
DUCSKX(I) = UREC2(5)/WS *(SK1(1,IWS,I14)+2.*SK1(2,IWS,I14)
*      * WLOG) * DWSX(I)
DUCACX(I) = (UREC2(4)-0.2886*LSAT)/AML*(ATT1(1,IAML,I15)
*      +2.*ATT1(2,IAML,I15)*AMLLOG)*(DAMLVM*DVEMX(I)+DAMLPS*DPPSX(I))
DUCRDX(I) = DNCRDX(I) = 0.0
IF(PRD.LE.0.) GO TO 136
DUCRDX(I) = UREC2(6) / PRD * (RAD1(2,I13) + 2. * RAD1(3,I13) *
*      PRLOG) * DPRDX(I)
DNCRDX(I) = NREC2(6) / PRD * (RAD2(2,I13) + 2. * RAD2(3,I13) *
*      PRLOG) * DPRDX(I)
136 DUCTRX(I) = CMFG/Y(1) * (ACMFG(2) + 2.*ACMFG(3)*PLOG) * DPTRX(I)
*      + .1 * CMFGPP/PTA * (ACMFGP(2) + 2.*ACMFGP(3)*PTALOG)*DPTRX(I)
DUCPTX(I) = CPT1D/PPT * (CPTUR(2) + 2.*CPTUR(3)*PPTLOG) * DPPTX(I)
*      +CHAR1D/PPT*(CHARUR(2)+2.*CHARUR(3)*PPTLOG)*DPPTX(I)
DUCPCX(I) = CPCU1D/PCU * (CPCUUR(2)+2.*CPCUUR(3)*PCULOG)*DPCUX(I)
DUCPNX(I) = CPC1D/PPC * (CPCUR(2) + 2.*CPCUR(3)*PPCLOG)*DPPCX(I)

```

```

DUCLCX(I) = CMFGLC(2,KCLE) * DPLCX(I) * CRLCF * XNC
... NON-RECURRING COSTS ...
DNCSTX(I) = 3.5 * DUCSTX(I)
DNCPRX(I) = CPP2D/PPS*(CP2(2,I6) + 2.*CP2(3,I6)*PSLOG)*DPPSX(I)
DNCBTX(I) = CBT2D/WHRS * (CBT2(2,I20) + 2.*CBT2(3,I20)*WHLOG)
*      * 45.6 * RHOPC * DPBX(I)
DNCSKX(I) = NREC2(5)/WS *(SK2(1,IWS,I14)+2.*SK2(2,IWS,I14)
*      * WLOG) * DWSX(I)
DNCACX(I) = (NREC2(4)-0.85)/AML*(ATT2(1,IAML,I15)
*      +2.*ATT2(2,IAML,I15)*AMLLLOG)*(DAMLVM*DVEMX(I)+DAMLPS*DPPSX(I))
DNCTRX(I) = CE/Y(1)*(ACE(2) + 2.*ACE(3)*PLOG) * DPTRX(I)
*      + .1*CEPP/PTA * (ACEP(2) + 2.*ACEP(3)*PTALOG)*DPTRX(I)
DNCPTX(I) = CPT2D/PPT * (CPTNR(2) + 2. * CPTNR(3)*PPTLOG)
*      * DPPTX(I)
*      +CHAR2D/PPT*(CHARNR(2)+2.*CHARNR(3)*PPTLOG)*DPPTX(I)
DNCPCX(I) = CPCU2D/PPCU * (CPCUNR(2) + 2. * CPCUNR(3) * PCULOG)
*      * DPCUX(I)
DNCNPNX(I) = CPC2D/PPC*(CPCNR(2) + 2.*CPCNR(3)*PPCLOG) *DPPCX(I)
DNCLCX(I) = CENGLC(2,KCLE) * DPLCX(I)
IF(I2.EQ.2) DUCLCX(I) = DNCLCX(I) = 0.0
IF((I1.EQ.4).OR.(I1.EQ.6)) DUCSTX(I) = DNCSTX(I) = 0.
PSUM1 = DUCSTX(I) + DUCPRX(I) + DUCBTX(I) + DUCSKX(I)
*      + DUCRDX(I) + DUCTRX(I) + DUCPTX(I) + DUCPCX(I)
*      + DUCPNX(I) + DUCLCX(I) + DUCACX(I)
PSUM2 = DNCSTX(I) + DNCPRX(I) + DNCBTX(I) + DNCSKX(I)
*      + DNCRDX(I) + DNCTRX(I) + DNCPTX(I) + DNCPCX(I)
*      + DNCNPNX(I) + DNCLCX(I) + DNCACX(I)
R3 = NBS
IF (NBRQD .GT. NBS) R3=NBRQD
DUREC2 = 1.236*(LSYS/LSAT)*PSUM1/NBS
DNREC2 = (1.575*PSUM2 + 0.5375*PSUM1 + NPROTO*(PSUM1-
*      0.85*DUCLCX(I))) / (NSAT * R3)
GRAD(I+1) = DNREC2 + DUREC2
135 CONTINUE
C = 0.1
TERM1 = UREC41(1)
TERM2 = UREC41(2) - DELC1 - DELC2
GLC = 1.
XIN = 1.
IF (I21.EQ.1) GO TO 137
C = 0.15
TERM1 = UREC43(1)
TERM2 = UREC43(2)
GLC = GR85
XIN = 2.
137 GRAD(2) = GRAD(2) + TERM1 * XNGR * (1. + C *LSYS) * GR
*      * (ANTH(2,KRA) + 2.*ANTH(3,KRA) * D4LOG) / D(4)
*      + CINANT * XNGR * (CINST(2,KIA) +
*      2. * CINST(3,KIA) * D4LOG) / D(4)
GRAD(3) = GRAD(3) + TERM2 * GLC * XNGR * (XIN + .1 * LSYS) *
*      (RECH(2) + 2. * RECH(3) * TLOG) / X(3)
IF (MB.EQ.0) GO TO 500
DWRDX(1) = DWRDVM * DVEMX(1) + DWRDPS * DPPSX(1) + DWRDPT
*      * DPTRX(1).
DWRDX(2) = DWRDVM * DVEMX(2) + DWRDPS * DPPSX(2) + DWRDPT

```



```

*      * DPTRX(2)
K = IARRAY(MBS)
GO TO (1,2,3,4,5,6),K

C                                     ... DEPENDENT PARAMETERS ...
C                                     ... PARTIALS OF PTR WRT X ...
1 PT(1,1) = 0.
  PT(2,1) = DPTRX(1)
  PT(3,1) = DPTRX(2)
  GO TO 200

C                                     ... PARTIALS OF WEIGHT WRT X ...
2 CONTINUE
  DO 25 I=1,2
    DWCSX(I) = CCS1 * DWSX(I)
    DVCSX(I) = CCS2 * DWSX(I)
    DPCSX(I) = CCS3 * DWSX(I)
25 CONTINUE
  PT(1,1) = 0.
  PT(2,1) = DWSX(1) + DWCSX(1)
  PT(3,1) = DWSX(2) + DWCSX(2)
  GO TO 200

C                                     ... PARTIALS OF VEM WRT X ...
3 PT(1,1) = 0.
  PT(2,1) = DVEMX(1)
  PT(3,1) = DVEMX(2)
  GO TO 200

C                                     ... PARTIALS OF VOLUME WRT X ...
4 PT(1,1) = 0.
  PT(2,1) = DVEMX(1) + XK * DPPSX(1) + (1./DNV(I13))* DWRDX(1)
  *      DVMPX(1)
  PT(3,1) = DVEMX(2) + XK * DPPSX(2) + (1./DNV(I13)) * DWRDX(2)
  *      + DVMPX(2)
  GO TO 200

C                                     ... PARTIALS OF PG WRT X ...
5 CONTINUE
  PT(1,1) = -PHI2 * XNS / (G(1)**2 * G(2)) * DG1D1
  PT(2,1) = 0.
  PT(3,1) = 0.
  GO TO 200

C                                     ... PARTIALS OF VPP WRT X ...
6 PT(1,1) = 0.
  PT(2,1) = XK * DPPSX(1)
  PT(3,1) = XK * DPPSX(2)
200 CONTINUE
500 CONTINUE
  RETURN
  END

```

5.7 SUBROUTINE BCHECK

The function of subroutine BCHECK is to determine whether any of the boundary constraints have been violated and, if more than one are violated, which violates by the greatest degree.

Upon entrance to the routine, the counter of boundary violations (MB) is set to zero.

The array indicating which components violate boundaries (IARRAY) and the array indicating whether a maximum or minimum constraint is violated (JARRAY) are initialized to zero.*

The routine performs three general steps:

- (a) it checks to see if any of the components violate a minimum boundary,
- (b) it checks to see if any of the components violate a maximum boundary, and
- (c) if there is more than one boundary violation, it determines which exceeds its constraint to the largest degree and flags that (KS) as the one which will be used in computing ΔX .

* A value of 1 indicates the particular violation was with respect to a minimum.

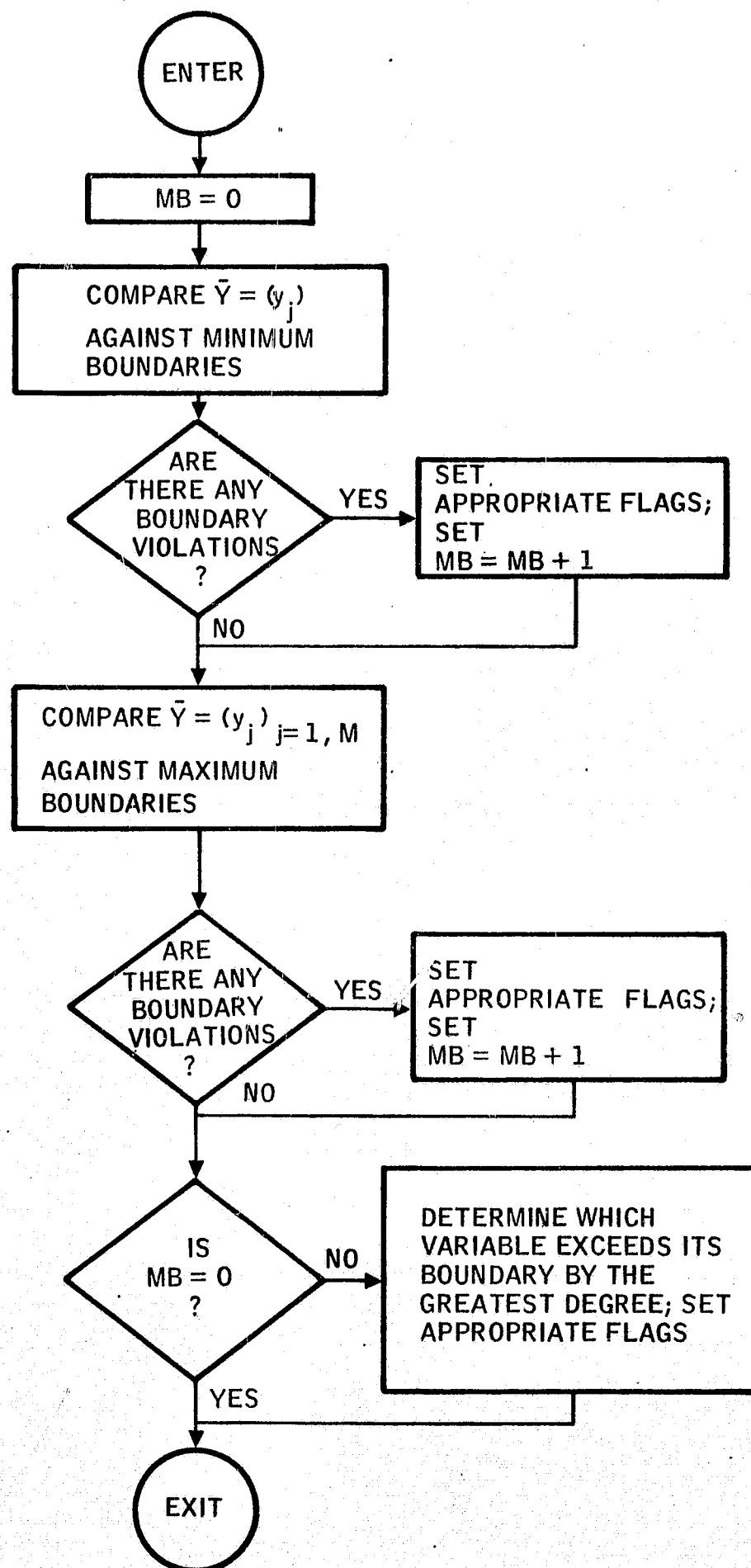


Figure 5.8 Flow Diagram - Subroutine BCHECK

```

SUBROUTINE BCHECK(Y,MB,KS,MBS,N,M,L)
COMMON/BOUND/IARRAY(15),JARRAY(15),YMIN(15),YMAX(15)
DIMENSION Y(15)
NAMELIST/NAME6/ IARRAY,JARRAY,MB
MB = 0
CALL ZERO(IARRAY,15,1)
CALL ZERO(JARRAY,15,1)
DO 300 I=1,M
IF(Y(I).GE.YMIN(I)) GO TO 100
MB = MB + 1
IARRAY(MB) = I
JARRAY(MB) = 1
IF(YMIN(I).EQ.0.) YMIN = 1.E-30
GO TO 300
100 IF(Y(I).LE.YMAX(I)) GO TO 300
MB = MB + 1
IARRAY(MB) = I
300 CONTINUE
DIST1=0.
MBS = 1
KS = IARRAY(1)
IF(MB.LE.1) GO TO 400
DO 325 I = 1,MB
K = IARRAY(I)
IF(JARRAY(I).EQ.1) GO TO 310
DIST2 = (Y(K) - YMAX(K)) / YMAX(K)
GO TO 315
310 DIST2 = (YMIN(K) - Y(K)) / YMIN(K)
315 IF(DIST2.LE.DIST1) GO TO 325
DIST1 = DIST2
KS = K
MBS = I
325 CONTINUE
400 CONTINUE
RETURN
END

```

5.8 SUBROUTINE ERROR

The function of subroutine ERROR is to print diagnostic messages in the event that certain conditions cause a case to be terminated. A diagnostic is printed under the following conditions:

- IFLAG=1: available bandwidth insufficient for the desired number of channels
- IFLAG=2: satellite downlink antenna size exceeds configuration boundary
- IFLAG=5: satellite downlink antenna size for concept(s) below the minimum boundary (D3BD).

```

SUBROUTINE ERROR(IFLAG)
DIMENSION CNCPT(8),YMOD(2)
COMMON /DIAM / D(4),G(4)
COMMON/MAIN/ I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15
*           ,I16,I17,I18,I19,I20,I21,I22,I23,I24,I25
*           ,XQ,XJ,XK,XL,XP1,XP2,XP3,XP4
COMMON/TABLE/ AWGHT(3,4,3),AVOL1(3,4,3),AVOL2(3,4,3),ACENG1(3,4,3)
*           ,ACENG2(3,4,3),ACENG3(3,4,3),ACMFG1(3,4,3)
*           ,ACMFG2(3,4,3),AETA(3,4,3),FWGHT(3,4,3),FVOL1(3,4,3)
*           ,FVOL2(3,4,3),FCENG1(3,4,3),FCENG2(3,4,3)
*           ,FCENG3(3,4,3),FCMFG1(3,4,3),FCMFG2(3,4,3)
*           ,FETA(3,4,3),PSPPLY(4,2),ANT(4,2)
*           ,FBOOM(4,3),BCHANN(2),DABD(8),BLNGTH(8)
*           ,BVOL(8),BWGHT(8),DEM(8),FREQ(4),BDWDTH(4)
*           ,UPFREQ(2),EXPR,FDLIN
DATA CNCPT(1),CNCPT(2),CNCPT(3),CNCPT(4),CNCPT(5),CNCPT(6),
*   CNCPT(7),CNCPT(8)/2H A,2H B,2H C,2H D,2H E,2H F,2H G,2H H/
DATA YMOD(1),YMOD(2)/3H AM,3H FM/
WRITE(6,599) IFLAG
GO TO (1,2,3,4,5,6),IFLAG
1 WRITE (6,600)
GO TO 100
2 WRITE (6,605)
WRITE(6,650) D(3)
GO TO 100

3 WRITE(6,610)
GO TO 100
4 WRITE(6,615)
GO TO 100
5 WRITE (6,620)
WRITE(6,650) D(3)
6 CONTINUE
100 CONTINUE
WRITE(6,900) CNCPT(I1),FREQ(I3),YMOD(I2)
599 FORMAT (1H1,15H ABNORMAL EXIT.,2X,9H IFLAG = I2)
600 FORMAT (52H BANDWIDTH INSUFFICIENT FOR DESIRED NO. OF CHANNELS.)
605 FORMAT (60H SATELLITE DOWN-LINK ANTENNA EXCEEDS CONFIGURATION BOUN
*DARY.)
610 FORMAT(31H MATRIX INVERSION UNSUCCESSFUL.)
615 FORMAT(25H INCONSISTENT BOUNDARIES.)
620 FORMAT(47H DOWNLINK ANTENNA FALLS BELOW MINIMUM BOUNDARY.)
650 FORMAT(21H ANTENNA DIAMETER IS ,F5.1, 6H FEET.)
900 FORMAT(8H CONCEPT,A2,14H, FREQUENCY = F4.1,2H, ,A2,11H MODULATION)
RETURN
END

```

5.9 SUBROUTINE OUT AND OUTPTX

The function of subroutine OUT is to provide intermediate output (per iteration) and subroutine OUTPTX provides the final output. Listings of both subroutines are provided in the following set of figures. A sample listing of a run summary is given in Section 4.

```

SUBROUTINE OUT(S,XK,HG,STOR1,DX,Y,ITER,MB,MBS,KS,JCONV,ICONV,N,M,L
*      ,R1)
COMMON/BOUND/IARRAY(15),JARRAY(15),YMIN(15),YMAX(15)
COMMON/PARA/ PLOG,ETA,ARD,VTR,WTR,FREQ4,XNS,RHO,AKP,XNG
*      ,CE,CMFG,WEM,WLOG,PGLOG,CSTPYR,G4LOG,TLOG
*      ,VEM,WSTLOG,PSLOG,WHLOG,PRLOG,D1LOG
*      ,D4LOG,PTALOG,PCULOG,PPCLOG,PPTLOG,CEPP,CMFGPP
COMMON/CNVRGE/ EPSLON,MCONV,ITMAX,PSIK,GRAD(5)
DIMENSION DX(1),Y(1),HG(1),STOR1(1)
BLANK = 10H      -
IF(ITER.EQ.1) WRITE(6,699)
699 FORMAT(1H1)
IF(ITER.EQ.1) WRITE(6,700)
700 FORMAT(26X,20HDEPENDENT PARAMETERS,39X,22HINDEPENDENT PARAMETERS/
*1X,4HITER,6X,3HPTR,8X,4HWGHT,9X,3HVEM,9X,3HVOL,9X,2HPG,9X,4H VPP,
*8X,4HD(1),8X,4HD(4),8X,2HTR,11X,3HC/Y)
WRITE(6,750) ITER,(Y(J),J=1,9),CSTPYR
750 FORMAT(2X,12,1X,10(1X,E11.4))
WRITE(6,775) S, (DX(I),I=1,N)
775 FORMAT(8X,4HS = E10.3,52X,5HDX = E10.3,2(2X,E10.3))
WRITE(6,790) MB,(IARRAY(I),I=1,9),KS,(JARRAY(I),I=1,9)
790 FORMAT(8X,5HMB = I3,3X,9HIARRAY = ,9(2X,I2) ,8X,5HKS = I3,3X,
* 9HJARRAY = ,9(2X,I2))
WRITE(6,800) JCONV,ICONV
800 FORMAT(8X,8HJCONV = I2,3X,8HICONV = I2)
WRITE(6,810) R1,(GRAD(I),I=1,3)
RETURN
810 FORMAT(4(5X,E11.4))
END

```



```

SUBROUTINE OUTPTX(Y,N,M,L)
COMMON/ATTN/ BVEST,BBB(2),ZMOD(2),XLAT,XLONG,RP,BT1(2),BT2(4)
*   ,AT1(2),AT2(4),KAT1N1(2),KATTN2(4),TZ1(2),TZ2(4),XL1,XLT1,XLT2
*   ,XLT3,XLT4,XPT1,XPT2,RPT1,RPT2,P1,P2,COSNT(4),TION,TATM,TCLOUD
*   ,TRAIN,TTL,AMMT(3,4),TE,TRCVRS,GE1,GE2,GS1,GS2,GM1,GM2,GSX(3,3)
*   ,GEX(3,3),GMX(3,3),AMTPLX(4),BMTPLX(4),TCO,TI,TA,TC,TM,TR,TT
*   ,TEFF,FSL12,FSL34,ATL1,ATL2,XCIRC1,XCIRC2,RCIRC1,RCIRC2
*   ,XL12,XL34,KGS,KGM,KGE,RPP(3),BETA,GAMMA,EL,RSL
COMMON/BOUND/IARRAY(15),JARRAY(15),YMIN(15),YMAX(15)
COMMON/CNST/ PI,R,CVEL,BOLZ,RSAT,RE
COMMON/ CNVRGE/ EPSLON,MCONV,ITMAX,PSIK,GRAD(5)
COMMON/COST/ UREC1(7),NREC1(6),ACQC1(3),OPC1(10),UREC2(15)
*   ,NREC2(20),ACQC2(4),OPC2(1),UREC3(2),NREC3(5)
*   ,ACQC3(4),OPC3(1),UREC41(2),NREC41(3),ACQC41(2)
*   ,OPC41(1),UREC42(7),NREC42(7),ACQC42(2),OPC42(8)
*   ,UREC43(7),NREC43(5),ACQC43(2),OPC43(7),TUREC(4)
*   ,TNREC(4),TACQC(4),TOPC(4)
COMMON/DATE/ IMO,IDA,IYR,POP1,POP2,ICASE, TITLE(6)
COMMON /DIAM / D(4),G(4)
COMMON/DISCR/ GE,GS,GM
COMMON/MAIN/ I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,I13,I14,I15
*   ,I16,I17,I18,I19,I20,I21,I22,I23,I24,I25
*   ,XQ,XJ,XK,XL,XP1,XP2,XN,XP3,XY,XP4
COMMON/MISC/ BLEM,BVEM,XLF1,XLF2,XLF,XLA,SNRG,XTA,XIMPRV(2),BVID12
*   ,BVID34,TSTRCT,AMAJR,BMINR
COMMON/NUMB/ NTR,NCTR,NF,LSYS,NSL,NRS,NLV,NRLV,NRSTA,NB,NCB,NC
*   ,NCTRF,NTRF,NCF,LSAT,FREQ1,GR85,GR89,GR95,GR,XNGR,XNGRP
*   ,NAPV,NAUDC,PDNSE,ITASO,NBRQD,AOC,NSAT,NBS,SETDNS,JIT
COMMON/PARA/ PLOG,ETA,ARD,VTR,WTR,FREQ4,XNS,RHO,AKP,XNG
*   ,CE,CMFG,WEM,WLOG,PGLOG,CSTPYR,G4LOG,TLOG
*   ,VEM,WSTLOG,PSLOG,WHLOG,PRLOG,D1LOG
*   ,D4LOG,PTALOG,PCULOG,PPCLOG,PPTLOG,CEPP,CMFGPP
COMMON/PARA2/WPP,VPP,PPS,WA,VA,WAP,VAP,WREC,VREC,WSTRCT,VSTRCT
*   ,WMPLX,VMPLX,WRD,VRD,WCS,VCS,WAC,VAC,PRD,WINC,VINC
*   ,WBATT,VBATT,CMPLX,CREC,CPP1D,CPT1D,CPCU1D,CLCF1D
*   ,CPC1D,CPP2D,CPT2D,CPCU2D,CLCF2D,CPC2D,CBTT1D
*   ,CBTT2D,PTA,PPT,PPCU,PPC,WTRP,VTRP,VPCU,WPCU,VPC,WPC
*   ,ETAP,ETALC,CTL,UCOSTR,COND1D,COND2D,WCOND,VCOND,PLCF
*   ,VPT,CHAR1D,CHAR2D,CINANT,CINREC,CINTL,CINDIS,CABL
COMMON/PARA3/ YM,YL,AML,AREA,WHRS,WS,XLEM
COMMON/SYS/ WMP,VMP,WGHTB(3,3),VOLB(3,3),XTR(3),ANTC(3),CSTR(3,3)
*   ,CP1(3,2),CP2(3,2),RAD1(3,2),RAD2(3,2),SK1(2,2,3),SK2(2,2,3)
*   ,PLX1(4,3),PLX2(4,3),DNLA,ANTD1(3,4),ANTD2(3,4),CBT1(3,3)
*   ,CBT2(3,3),RECH(3),RECH1(3,3,4),TRBD1(2,4),RECH2(3,3),TRBD2(2)
*   ,RECA1,RECA2,ANTH(3,4),XINST1,XINST2,CINST(3,4),CRCUR(3,2)
*   ,CRCNR(3,2),NOANT(8),DCYCLE,ARCVR(4),ETAPT,ETAPC,ETAB,ETAPCU
*   ,ETALCF,RHOPC,DGRD(2),WPTR(3) ,WCU1,WCU2,WCU(3,3),WPCN(3)
*   ,WLCFP1,WLCFP(2,2),VPTR1,VPTR(3,2),VCU1,VCU(3,2),VPCN1
*   ,VPCN(3,2),VLCFP1,VLCFP(2,2),CENGP1,CENGP2,CMFGP1,CENGPC(3,3)
*   ,CMFGPC(3,2),CLC1,CLC2,CENGLC(2,3),CMFGLC(2,3),XCNST
*   ,DEGRAD,CPTUR(3),CPTNR(3),CPCUUR(3),CPCUNR(3),CPCUR(3),CPCNR(3)
*   ,CBETA,SBETA,IETBD1,IETBD2,WMUX(3,3),WHARR(3,8),CHARUR(3)
*   ,CHARNR(3),NPROTO,SKLS1(2,2,3),SKLS2(2,2,3),ATT1(2,2,3)
*   ,ATT2(2,2,3),ATLS1(2,2,3),ATLS2(2,2,3),IAML,IWS,AMLLG
*   ,DFMREC,PARANT(4),NRGR,ACQLV(8),ENGLV(8),ACQMP(8),ENGMP(8)

```

```

* ,URCBSM,ENGBSM,URCMDA,ENGMDA
* ,RCVR1(2),RCVR2(1),HRCVR1(3,4),HRCVR2(3,2),XLC1(3,2),XLC2(3,1)
* ,XLC3(3,1),RCVC2,RCVC3,DELC1,DELC2
COMMON/TABLE/ AWGHT(3,4,3),AVOL1(3,4,3),AVOL2(3,4,3),ACENG1(3,4,3)
* ,ACENG2(3,4,3),ACENG3(3,4,3),ACMG1(3,4,3)
* ,ACMG2(3,4,3),AETA(3,4,3),FWGHT(3,4,3),FVOL1(3,4,3)
* ,FVOL2(3,4,3),FCENG1(3,4,3),FCENG2(3,4,3)
* ,FCENG3(3,4,3),FCMG1(3,4,3),FCMG2(3,4,3)
* ,FETA(3,4,3),PSPPLY(4,2),ANT(4,2)
* ,FBOOM(4,3),BCHANN(2),DABD(8),BLNGTH(8)
* ,BVOL(8),BWGHT(8),DEM(8),FREQ(4),BDWDTH(4)
* ,UPFREQ(2),EXPR,FDLIN
COMMON/TRANS/ AEFF,PB,PRDEM,PRDT,XLMUX,PR,PCS,PAC,PINC,PMP
* ,XLA34,XLC34,XLR34, XLRH34
DIMENSION BC(2),ANTYPE(2),XMOD(2),TTYPE(5),PTYPE(2)
* ,ATYPE(2),FDTYPE(3),RTYPE(4),XMULT(3),CNCPT(8),STRCT(2)
* ,RDTYPE(2),SKTYPE(5),ATTTYPE(3),RCTYPE(2),VTYPE(2),Y(15)
DIMENSION CST1(15),CST2(15),CST3(15),CST4(15),TOTC(4),BTYPE(3)
DIMENSION TYPEM(3), ENV(3), LBATT(3), YBOUND(9), YESNO(2),
* TTLE(9), UNITS(9),YVIOL(9), TVSYS(3)
REAL NREC1,NREC2,NREC3,NREC41,NREC42,NREC43,K1,K2,NOANT
REAL KATTN1,KATTN2,LSYS,LSAT,NBRQD,NLV,NTR,NAPV,NAUDC,NCF
REAL LTR,LRCVR,LPT,LBATT,LPC,LPCU,LLCF
REAL NSAT,NBS, NPROTO
REAL NRGR
DATA PI/3.1415927/
DATA XC1,XC2/42000., 6000./
DATA TYPEM(1),TYPEM(2),TYPEM(3) /10HBSM ,10H ,
* 10HMDA /
DATA YESNO(1), YESNO(2) / 10H NO , 10H YES /
DATA TTLE(1), TTLE(2), TTLE(3), TTLE(4), TTLE(5), TTLE(6),
* TTLE(7), TTLE(8), TTLE(9) / 10H PTR-SAT , 10H WT-SAT ,
* 10H VOL-EM , 10H VOL-SAT , 10H PTR-GND , 10H VOL-SA ,
* 10H DA-G UP , 10H DA-G DN , 10H TR-GND /
DATA UNITS(1), UNITS(2), UNITS(3), UNITS(4), UNITS(5),
* UNITS(6), UNITS(7), UNITS(8), UNITS(9) / 10H (WATTS) ,
* 10H (LBS) , 10H (CU FT) ,
* 10H (CU FT) , 10H (WATTS) , 10H (CU FT) ,
* 10H (FT) , 10H (FT) , 10H (DEG K) /
DATA ANTYPE(1),ANTYPE(2)/10HWIDEBAND ,10HPARABOLIC /
DATA XMOD(1),XMOD(2)/2HAM, 2HFM/
DATA BTYPE(1),BTYPE(2),BTYPE(3)/10HNICKEL CAD,10HSILVER CAD,
* 10HSILVR ZINC/
DATA ATTYPE(1),ATTTYPE(2),ATTTYPE(3)/10HNH3 RESIST,10HION PROP ,
* 10HH2O ELECTR/
DATA TTYPE(1),TTYPE(2),TTYPE(3),TTYPE(4),TTYPE(5)
* /10HGRID TUBE ,10HSOL STATE ,10HCFA ,10HTWT ,
* 10HKLYSTRON /
DATA PTYPE(1),PTYPE(2) /10HSOL ARRAY , 10HNCLR REACT/
DATA ATYPE(1),ATYPE(2)/10HTRUSS ,10HRIGID DISH/
DATA FDTYPE(1),FDTYPE(2),FDTYPE(3)/10HFOLDING ,10HEXPANDABLE,
* 10HRIGID /
DATA RTYPE(1),RTYPE(2)/10HLIN TRANS ,10HDMOD CNVRT/
DATA ENV(1),ENV(3),ENV(2)/10H RURAL ,10H URBAN ,10HSUBURB
* /

```

```

DATA BC(1),BC(2) /10HMONOCHROME,10HCOLOR /
DATA XMULT(1),XMULT(2),XMULT(3)/10HCOAX ,10HWAVEGUIDE ,
* 10HWAVEGUIDE /
DATA CNCPT(1),CNCPT(2),CNCPT(3),CNCPT(4),CNCPT(5),CNCPT(6)
*,CNCPT(7),CNCPT(8)/10HA ,10HB ,10HC
*10HD ,10HE ,10HF ,10HG
*10HH /
DATA STRCT(1),STRCT(2)/10HCONV ,10HBSM /
DATA RDTYPE(1),RDTYPE(2)/10HPASSIVE ,10HHYBRID /
DATA SKTYPE(1),SKTYPE(2),SKTYPE(3),SKTYPE(4),SKTYPE(5)
*/10HMH3 RESIST,10HION PROP ,10HH2O ELECTR,10HMH2 H4
* 10HCOLD GAS /
DATA VTYPE(1),VTYPE(2)/10HSV (UNMAN),10HSV(MANNED)/
DATA TVSYS(1),TVSYS(2),TVSYS(3) / 10HDIRECT
* 10HRETRANS , 10H CATV /
NAMELIST/OUTD/ PPS,PPSD,DEGRAD,AREA,PB,PPT,PPC,PPCU,PLCF,PRDEM
* ,PRD,PRDT,AEFF,ETAP,PTA,WTRP,VTRP,CEPP,CMFGPP,WTR
* ,VTR,CE,CMFG,XLMUX,PR,WS,PCS,AML,YM,YL,PAC,PINC,PMP
* ,GS,GE,GM,TCO,TI,TA,TC,TR,TT,TM,XLA34,XLC34,XLR34
* ,XLI,XLRH34,XLT4,EL,COST1,COST2,CTL,GR85,GR89,GR95
* ,Y,UREC1,NREC1,OPC1,ACQC1,UREC2,NREC2,ACQC2,OPC2,UREC41,NREC41
* ,ACQC41,OPC41,UREC43,NREC43,ACQC43,OPC43,UREC3,NREC3,ACQC3
* ,OPC3,TUREC,TNREC,TOPC,TACQC,CSTPYR,VEM,WEM
BLANK = 7H -
XSS = LSYS/LSAT
ICASE = ICASE + 1
WRITE(6,1) ICASE,TITLE,IMO,IDA,IYR
1 FORMAT(1H1,9HCASE NO. I3,15X,6A10,15X,I2,1H/,I2,1H/,I2//)
WRITE(6,600)
600 FORMAT(1H ,29X,17HUSER REQUIREMENTS)
WRITE(6,500)
500 FORMAT(1H+,89X,15HLINK PARAMETERS/77X, 9HPARAMETER ,16X,6HUPLINK,
*5X,8HDOWNLINK)
WRITE(6,699)
699 FORMAT ( 15X,16HAREA OF COVERAGE )
WRITE(6,501)FREQ1,FREQ(I3)
501 FORMAT(1H+, 73X,14HFREQUENCY(GHZ),10X,F7.2,5X,F7.2)
AOC = AOC * 1.E-06
WRITE(6,601) AMAJR
601 FORMAT(18X,14HMAJOR AXIS(MI),18X,F9.3)
ERP1 = 10. * ALOG10(Y(5) * G(1) * XCIRC1)
ERP2 = 10. * ALOG10(Y(1) * G(3) * XCIRC2)
WRITE(6,502)ERP1,ERP2
502 FORMAT(1H+,73X,8HERP(DBW),16X,F6.1,6X,F6.1)
WRITE(6,615) BMINR
615 FORMAT(18X,14HMINOR AXIS(MI),18X,F9.3)
BVID1P = BVID12 * 1.E+03
BVID2P = BVID34 * 1.E+03
WRITE(6,512) BVID1P,BVID2P
512 FORMAT( 1H+,73X,17HRF BANDWIDTH(MHZ),7X,F6.1,6X,F6.1)
WRITE(6,616) AOC
616 FORMAT(18X,29HAOC(MILLIONS OF SQUARE MILES),3X,F9.3)
XMOD11 = 1.
XMOD12 = 1.
IF (I10.EQ.2) XMOD11 = 3.

```

```

IF (I2.EQ.2) XMODI2 =3.
WRITE(6,513) XMODI1, XMODI2
513 FORMAT(1H+,73X,16HMODULATION INDEX,8X,F6.1,6X,F6.1)
XNGR1 = XNGR * 1.E-03
IF (I21.EQ.3) XNGR1 = XNGR1*SETDNS*30.
WRITE(6,602) XNGR1
602 FORMAT(15X,28HRECEIVER POP/BEAM(THOUSANDS),7X,F9.3)
ANTBW1 = 70. * CVEL / (D(2) * FREQ1)
ANTBW2 = 70. * CVEL / (D(3) * FREQ(I3))
WRITE(6,514) ANTBW1,ANTBW2
514 FORMAT(1H+,73X,22HANTENNA BEAMWIDTH(DEG),2X,F6.1,6X,F6.1)
XNGR2 = XNGR * 1.E-03
IF (I21.EQ.3) XNGR2 = XNGR2*SETDNS*30.
WRITE(6,698) XNGR2
698 FORMAT(15X,29HTOTAL RECEIVER POP(THOUSANDS),5X,F10.3)
XLATP = XLAT * 180. / PI
WRITE(6,603) XLATP
603 FORMAT(15X,17HLATITUDE(DEGREES),20X,F7.3)
WRITE(6,503)
503 FORMAT(1H+,73X,8HLOSS(DB))
XLONGP = XLONG * 180. / PI
WRITE(6,631) XLONGP
631 FORMAT(15X,27HRELATIVE LONGITUDE(DEGREES),10X,F7.3)
FSL12 = 10. * ALOG10(FSL12)
FSL34 = 10. * ALOG10(FSL34)
WRITE(6,525) FSL12,FSL34
525 FORMAT(1H+,75X,9HFREESPACE,13X,F6.1,6X,F6.1)
WRITE(6,604) ITASO
604 FORMAT(15X,10HTASO GRADE,29X,I3)
ATL1 = 10. * ALOG10(ATL1)
ATL2 = 10. * ALOG10(ATL2)
WRITE(6,504) ATL1,ATL2
504 FORMAT(1H+,75X,11HATMOSPHERIC,11X,F6.1,6X,F6.1)
SNRGP = 10. * ALOG10(SNRG)
WRITE(6,620) SNRGP
620 FORMAT(15X,25HSIGNAL TO NOISE RATIO(DB),10X,F9.3)
XCIRC1 = 10. * ALOG10(XCIRC1)
XCIRC2 = 10. * ALOG10(XCIRC2)
WRITE(6,516) XCIRC1,XCIRC2
516 FORMAT(1H+,75X,12HXMIT CIRCUIT,10X,F6.1,6X,F6.1)
WRITE(6,605) NF
605 FORMAT(15X,31HNUMBER OF UPLINK FAC/SAT SYSTEM,8X,I3)
RCIRC1 = 10. * ALOG10(RCIRC1)
RCIRC2 = 10. * ALOG10(RCIRC2)
WRITE(6,517) RCIRC1,RCIRC2
517 FORMAT(1H+,75X,12HRCVR CIRCUIT,10X,F6.1,6X,F6.1)
WRITE(6,606) NCF
606 FORMAT(15X,27HNUMBER OF CHANNELS/FACILITY,12X,F3.0)
XPT2 = 10. * ALOG10(XPT2)
XPT1 = 10. * ALOG10(XPT1)
WRITE(6,505) XPT1,XPT2
505 FORMAT(1H+,75X,13HXMIT POINTING,9X,F6.1,6X,F6.1)
WRITE(6,607) XMOD(I10)
607 FORMAT(15X,17HUPLINK MODULATION,23X,A2)
RPT1 = 10. * ALOG10(RPT1)

```

```

RPT2 = 10. * ALOG10(RPT2)
WRITE(6,506)RPT1,RPT2
506 FORMAT(1H+,75X,13HRCVR POINTING,9X,F6.1,6X,F6.1)
WRITE(6,612) XMOD(12)
612 FORMAT(15X,19HDOWNLINK MODULATION,21X,A2)
P1 = 10. * ALOG10(P1)
P2 = 10. * ALOG10(P2)
WRITE(6,507) P1,P2
507 FORMAT(1H+,75X,12HPOLARIZATION,10X,F6.1,6X,F6.1)
WRITE(6,608) LSYS
608 FORMAT(15X,32HTOTAL SYSTEM OPERATION LIFE(YRS) ,7X,F3.0)
WRITE(6,609) LSAT
609 FORMAT(15X,19HSATELLITE LIFE(YRS) ,20X,F3.0)
WRITE(6,508)
508 FORMAT( 1H+,73X,25HNOISE TEMPERATURES(DEG K))
WRITE(6,610) DCYCLE
610 FORMAT(15X,19HDUTY CYCLE(HRS/DAY),19X,F4.1)
TSKY = TCO + TI + TA + TC + TR
WRITE(6,509) BLANK,TSKY
509 FORMAT(1H+,75X,3HISKY,19X, A7 ,5X,F6.1)
WRITE(6,611) NCB
611 FORMAT(15X,29HNUMBER OF VIDEO CHANNELS/BEAM,10X,I3)
WRITE(6,510) BLANK,TM
510 FORMAT(1H+,75X,13HENVIRONMENTAL,9X, A7 ,5X,F6.1)
WRITE(6,650) NAPV
650 FORMAT(15X,30HNUMBER OF AUDIO/VIDEO CHANNELS,9X,F3.0)
WRITE(6,511) TE,TE
511 FORMAT(1H+,75X,5HEARTH,17X,F6.1,6X,F6.1)
IJ = 1
IF (PDNSE.GT.POP1) IJ=2
IF (PDNSE.GT.POP2) IJ=3
WRITE(6,675) ENV(IJ)
675 FORMAT(15X,17HNOISE ENVIRONMENT,21X,A10)
RCV = Y(M+3)
WRITE(6,515) TRCVRS,RCV
515 FORMAT(1H+,75X,8HRECEIVER,14X,F6.1,5X,F7.1)
WRITE(6,541) NBRQD
541 FORMAT(15X,24HNUMBER OF BEAMS/SYSTEM ,15X,F3.0)
ZRO = 0.
WRITE(6,599) ZRO,TT
599 FORMAT(1H+,75X,17HTRANSMISSION LINE,5X,F6.1,6X,F6.1)
WRITE(6,632) NSAT
632 FORMAT(15X,21HNUMBER OF SAT SYSTEMS,18X,F3.0)
WRITE (6,633) TVSYS(I21)
633 FORMAT (15X, 22HGROUND DOWNLINK SYSTEM , 16X, A10)
R1= NBRQD / NSL
R2 = NBRQD / NBS
IF ( R1.GT.1. ) R1 = 1.
IF (R2.GT.1. ) R2 = 1.
X80 = XSS * ACQC3(4) + TNREC(3)
ACST = NBRQD * NSAT * CSTPYR
BCST = ACST / XNGRP * 1.E+06
IF(I21.EQ.3) BCST=BCST/(SETDNS*30.)
ACSTP = ACST - ( NSAT*NBRQD/(NBS*NSL)*TACQC(3)
* + TNREC(3)*R1*R2)/LSYS

```

```

BCSTP = ACSTP / XNGRP * 1.E+06
IF(I21.EQ.3) BCSTP=BCSTP/(SETDNS*30.)
WRITE(6,700) ACSTP,BCSTP,ACST,BCST
700 FORMAT(///1X, 21HMINIMUM COST SOLUTION , 29X, 13HTOTAL COST/YR ,
* 6X, 12HCOST/YR/RCVR / 53X, 10H(MILLIONS) , 9X, 9H(DOLLARS) /
* 16X, 29HEXCLUDING LAUNCH VEHICLE COST , 5X, F11.3, 8X, F11.2/
* 16X, 29HINCLUDING LAUNCH VEHICLE COST , 5X, F11.3, 8X, F11.2)
JJJ = 1
DO 6000 I=1,9
II=1
JJ=1
TMIN = YMIN(I)
IF(YMIN(I).EQ.0.) TMIN = 1.
IF (ABS((Y(I)-YMAX(I))/YMAX(I)).LE. 0.001 .OR. ABS((Y(I)-
* YMIN(I))/TMIN).LE. 0.001 ) II = 2
IF (Y(I).GE.1.001*YMAX(I) .OR. Y(I).LE.0.999*YMIN(I))
* JJ=JJJ=2
YVIOL(I) = YESNO(JJ)
6000 YBOUND(I) = YESNO(II)
IF (JJJ.EQ.2) WRITE (6,6009)
6009 FORMAT (/58X, 22H***** / 58X,
* 22H* * / 58X, 22H* BOUNDARY VIOLATION * /
* 58X, 22H* * / 58X,22H*****))
WRITE (6,6001) CNCPT(II)
6001 FORMAT(/// 1X,36HOPTIMUM DESIGN PARAMETERS - CONCEPT , A10)
WRITE (6,6002) ( TTLE(I), I=1,9)
6002 FORMAT(/5X, 9HPARAMETER, 6X, 9(A10))
WRITE (6,6003) (UNITS(I), I=1,9)
6003 FORMAT (20X, 9(A10))
WRITE (6,6004) (Y(I), I=1,9)
6004 FORMAT(/5X, 12HDESIGN VALUE, 3X, 9(F8.1,2X))
WRITE (6,6005) (YBOUND(I), I=1,9)
6005 FORMAT (5X, 11HON BOUNDARY, 4X, 9(A10))
IF (JJJ.EQ.2) WRITE (6,6008) (YVIOL(I), I=1,9)
6008 FORMAT ( 5X, 13HVIOLATE BOUND , 2X, 9(A10))
WRITE (6,6006) (YMAX(I), I=1,9)
6006 FORMAT (5X, 11HUPPER BOUND, 4X, 9(F8.1,2X))
WRITE (6,6007) (YMIN(I), I=1,9)
6007 FORMAT (5X, 11HLOWER BOUND, 4X, 9(F8.1,2X))
WRITE(6,1) ICASE,TITLE,IMO,IDA,IYR
WRITE(6,702) CNCPT(II)
702 FORMAT(1H , 10HSPACECRAFT,26X,14HCOST(MILLIONS)/10H CONCEPT ,A10,
* 7X, 11HACQUISITION,
*4X,3HR/D,9X,5HTOTAL,7X,6HWEIGHT,6X,6HVOLUME,5X,4HTYPE,7X,11HDESCRI
*PTIVE/30X,4HCOST,8X,4HCOST,8X,4HCOST,8X,6H(LBS.),5X,9H(CU. FT.),14
*X,10HPARAMETERS)
WRITE(6,703)
703 FORMAT( 5X,15HPOWER SUBSYSTEM)
CST1(1) = 1.236 * CPP1D
CST2(1) = 1.575*CPP2D + 0.5375*CPP1D + 0.15*NPROTO*CPP1D
CST3(1) = XSS * CST1(1) + CST2(1)
PPSD = PPS * DEGRAD
WRITE(6,705) CST1(1),CST2(1),CST3(1),WPP,VPP,PTYPE(I6),PPSD
705 FORMAT(7X,12HPOWER SOURCE,9X,F7.3,5X,F7.3,5X,F8.3,4X,F9.2,3X,
* F9.2,3X,A10,2X,11HPRIME PWR =F7.0,6H WATTS)

```

```

CST1(12) = 1.236 * UREC2(3)
CST2(12) = 1.575 * NREC2(3) + (.5375+NPROTO)* UREC2(3)
CST3(12) = XSS * CST1(12) + CST2(12)
WRITE(6,704) CST1(12),CST2(12),CST3(12),WBATT,VBATT,BTYPE(I20),WHRS
704 FORMAT(7X,9HBATTERIES,12X,F7.3,5X,F7.3,5X,F8.3,4X,F9.2,3X,F9.2,3X,
*A10,2X,11HWATT HRS = F8.1)
CST1(13) = 1.236 * COND1D
CST2(13) = 1.575 * COND2D + (.5375+NPROTO)* COND1D
CST3(13) = XSS * CST1(13) + CST2(13)
WRITE(6,725) CST1(13),CST2(13),CST3(13),WCOND,VCOND
725 FORMAT(7X,12HCONDITIONING,9X,F7.3,5X,F7.3,5X,F8.3,4X,F9.2,3X,
*F9.2)
CST1(2) = 1.236 * UREC2(13)
CST2(2) = 1.575 * NREC2(13) + (.5375+NPROTO)* UREC2(13)
CST3(2) = XSS * CST1(2) + CST2(2)
X1 = 10. * ALOG10(G(3))
WRITE(6,706) CST1(2),CST2(2),CST3(2),WA,VA,ATYPE(I4),X1,D(3)
706 FORMAT(5X,18HANTENNA(DOWN-LINK),5X,F7.3,5X,F7.3,5X,F8.3,4X, F9.2,3
*X, F9.2,3X,A10,2X,6HGAIN =F5.1,3H DB,7H, DIA =F5.1,3H FT)
CST1(3) = 1.236 * UREC2(10)
CST2(3) = 1.575 * NREC2(10) + (.5375+NPROTO)* UREC2(10)
CST3(3) = XSS * CST1(3) + CST2(3)
X1 = 10. * ALOG10(G(2))
WRITE(6,707) CST1(3),CST2(3),CST3(3),WAP,VAP,ATYPE(I23),X1,D(2)
707 FORMAT(5X,16HANTENNA(UP-LINK),7X,F7.3,5X,F7.3,5X,F8.3,4X, F9.2,3X,
* F9.2,3X,A10,2X,6HGAIN =F5.1,3H DB,7H, DIA =F5.1,3H FT)
CST1(4) = 1.236 * UREC2(11)
CST2(4) = 1.575 * NREC2(11) + (.5375+NPROTO)* UREC2(11)
CST3(4) = XSS * CST1(4) + CST2(4)
WTRX = WTR + WTRP
VTRX = VTR + VTRP
WRITE(6,708) CST1(4),CST2(4),CST3(4),WTRX,VTRX,TTYE(I7),Y(1),ETA
708 FORMAT(5X,11HTRANSMITTER,12X,F7.3,5X,F7.3,5X,F8.3,4X, F9.2,3X, F9.
*2,3X,A10,2X,10HPWR/CHAN =F6.0,7H, EFF =F5.3)
CST1(7) = 1.236 * UREC2(12)
CST2(7) = 1.575 * NREC2(12) + (.5375+NPROTO)* UREC2(12)
CST3(7) = XSS * CST1(7) + CST2(7)
WRITE(6,711) CST1(7),CST2(7),CST3(7), WMPLX,VMPLX,XMULT(I9)
711 FORMAT(5X,11HMULTIPLEXER,12X,F7.3,5X,F7.3,5X,F8.3,4X, F9.2,3X, F9.
*2,3X,A10,2X)
CST1(5) = 1.236 * UREC2(9)
CST2(5) = 1.575 * NREC2(9) + (.5375+NPROTO)* UREC2(9)
CST3(5) = XSS * CST1(5) + CST2(5)
Q = 170. / 290. + 1.
X1 = 10. * ALOG10(Q)
WRITE(6,709) CST1(5),CST2(5),CST3(5),WREC,VREC,RTYPE(I8),X1
709 FORMAT(5X,8HRECEIVER,15X,F7.3,5X,F7.3,5X,F8.3,4X, F9.2,3X, F9.2,3X
*,A10,2X,11HNOISE FIG =F5.1)
CST1(6) = 1.236 * UREC2(1)
CST2(6) = 1.575 * NREC2(1) + (.5375+NPROTO) * UREC2(1)
CST3(6) = XSS * CST1(6) + CST2(6)
WRITE(6,710) CST1(6),CST2(6),CST3(6),WSTRCT,VSTRCT,STRCT(I12)
710 FORMAT(5X,9HSTRUCTURE,14X,F7.3,5X,F7.3,5X,F8.3,4X, F9.2,3X, F9.2,3
*X,A10,2X)
CST1(8) = 1.236 * UREC2(6)

```



```

CST2(8) = 1.575 * NREC2(6) + (.5375+NPROTO) * UREC2(6)
CST3(8) = XSS * CST1(8) + CST2(8)
WRITE(6,712) CST1(8),CST2(8),CST3(8),WRD,VRD,RDTYPE(I13),PRD,ARD
712 FORMAT(5X,15HTHERMAL CONTROL,8X,F7.3,5X,F7.3,5X,F8.3,4X, F9.2,3X,
* F9.2,3X,A10,2X,9HPWR RAD =,F6.0, 8H, AREA =F6.1)
CST1(9) = 1.236 * UREC2(5)
CST2(9) = 1.575 * NREC2(5) + (.5375+NPROTO) * UREC2(5)
CST3(9) = XSS * CST1(9) + CST2(9)
WRITE (6,715) CST1(9),CST2(9),CST3(9),WCS,VCS,SKTYPE(I14)
715 FORMAT(5X,15HSTATION KEEPING,8X,F7.3,5X,F7.3,5X,F8.3,4X, F9.2,3X,
* F9.2,3X,A10)
CST1(10) = 1.236 * UREC2(4)
CST2(10) = 1.575 * NREC2(4) + (.5375+NPROTO) * UREC2(4)
CST3(10) = XSS * CST1(10) + CST2(10)
WRITE (6,717) CST1(10),CST2(10),CST3(10),WAC,VAC,ATYPE(I15)
717 FORMAT(5X,16HATTITUDE CONTROL,7X,F7.3,5X,F7.3,5X,F8.3,4X, F9.2,3X
*, F9.2,3X,A10)
CST1(11) = 1.236 * UREC2(8)
CST2(11) = 1.575 * NREC2(8) + (.5375+NPROTO) * UREC2(8)
CST3(11) = XSS * CST1(11) + CST2(11)
WRITE(6,720) CST1(11),CST2(11),CST3(11),WINC,VINC
720 FORMAT (5X,14HTLM AND CMD ,9X,F7.3,5X,F7.3,5X,F8.3,4X,F9.2,3X,
*F9.2)
CST1(14) = 0.
CST2(14) = NREC2(20)
CST3(14) = CST2(14)
WRITE(6,729) BLANK,CST2(14),CST3(14),BLANK,BLANK
729 FORMAT (5X,14HTEST EQUIPMENT,9X,A7,5X,F7.3,5X,F8.3,6X,A7,5X,A7)
CST1(15) = CST2(15) = CST3(15) = 0.
IF(I1,LE.3) GO TO 730
IF(I1,EQ.8) GO TO 730
IF(I22,EQ.0) GO TO 726
CST1(15) = ACQMP(I1)
CST2(15) = ENGMP(I1)
CST3(15) = XSS * CST1(15) + CST2(15)
WRITE(6,731) CST1(15),CST2(15),CST3(15),WMP,VMP,TYPEM(I1-3)
731 FORMAT(5X,17HMANNED PROVISIONS,6X,F7.3,5X,F7.3,5X,F8.3,4X,F9.2,
* 3X,F9.2,3X,A10)
GO TO 730
726 WRITE(6,727) CST1(15),CST2(15),CST3(15),WMP,VMP,TYPEM(I1-3)
727 FORMAT(5X,17HMANNED PROVISIONS,6X,F7.3,5X,F7.3,5X,F8.3,4X,F9.2,
* 3X,F9.2,3X,17HUNUSED VOLUME IN , A10)
730 CONTINUE
CALL ZERO (TOTC,3,1)
DO 1000 I=1,15
TOTC(1) = TOTC(1) + CST1(I)
TOTC(2) = TOTC(2) + CST2(I)
TOTC(3) = TOTC(3) + CST3(I)
1000 CONTINUE
WRITE (6,750) TOTC(1),TOTC(2),TOTC(3),Y(2),Y(4)
750 FORMAT(28X,7H-----,5X,7H-----,5X,8H-----,5X, 8H-----,3X
*, 9H-----/15X,9HTOTAL/SAT,3X,F8.3,4X,F8.3,4X,F9.3,3X,F10.2,
* 2X,F10.2)
TOTC(1) = XSS * NSAT * NBRQD / NBS * TOTC(1)
TOTC(2) = TOTC(2)*R2

```



```

      TOTC(3) = TOTC(1) + TOTC(2)
      WRITE(6,785) TOTC(1),TOTC(2),TOTC(3)
785  FORMAT(7X,17HUSER SYSTEM TOTAL,1X,F10.3,2X,F10.3,2X,F11.3)
      WRITE(6,100)
100  FORMAT(/)
      WRITE(6,800)
800  FORMAT(1X,21HGROUND UP-LINK SYSTEM,21X,14HCOST(MILLIONS)/29X,
      *11HACQUISITION,4X,3HR/D,6X,10HOPERATIONS,4X,5HTOTAL,5X,11HDESCRIPT
      *IVE/32X,4HCOST,8X,4HCOST,7X,7HCOST/YR,5X,4HCOST,6X,10HPARAMETERS)
      CST1(1) = UREC1(1)
      CST2(1) = NREC1(1)
      CST3(1) = OPC1(1)
      CST4(1) = (CST1(1) + CST3(1) * LSYS) + CST2(1)
      WRITE (6,803) CST1(1),CST2(1),CST3(1),CST4(1)
803  FORMAT (/5X,10HFACILITIES,15X,F7.3,5X,F7.3,5X,F7.3,4X,F8.3)
      CST1(2) = 1.2 * UREC1(2)
      CST2(2) = NREC1(2)
      CST3(2) = OPC1(2) + OPC1(3)
      CST4(2) = (CST1(2) + CST3(2) * LSYS) + CST2(2)
      WRITE (6,806) CST1(2),CST2(2),CST3(2),CST4(2),BC(I19)
806  FORMAT (5X,24HPROG, REC, + CONTR EQUIP , 1X,
      *F7.3,5X,F7.3,4X,F8.3,4X,12HBROADCAST = A10)
      CST1(3) = 1.2 * UREC1(3)
      CST2(3) = NREC1(3)
      CST3(3) = .15 * UREC1(3)
      CST4(3) = (CST1(3) + CST3(3) * LSYS) + CST2(3)
      WRITE(6,802) CST1(3),CST2(3),CST3(3),CST4(3),Y(5)
802  FORMAT(5X,11HTRANSMITTER,14X,F7.3,5X,F7.3,5X,F7.3,4X,F8.3,4X,
      *10HPWR/CHAN =,F9.2)
      CST1(4) = 1.2 * UREC1(4)
      CST2(4) = NREC1(4)
      CST3(4) = .08 * UREC1(4)
      CST4(4) = (CST1(4) + CST3(4) * LSYS) + CST2(4)
      X1 = 10. * ALOG10(G(1))
      WRITE(6,801) CST1(4),CST2(4),CST3(4),CST4(4),X1,D(1)
801  FORMAT( 5X,7HANTENNA,18X,F7.3,5X,F7.3,5X,F7.3,4X,F8.3,4X,6HGAIN =,
      *F5.1,3H DB,7H, DIA =F5.1,3H FT)
      CST1(5) = CST2(5) = CST3(5) = CST4(5) = 0.0
      CST1(6) = 1.2 * UREC1(6)
      CST2(6) = 0.
      CST3(6) = OPC1(8)
      CST4(6) = (CST1(6) + CST3(6) * LSYS)
      WRITE (6,808) CST1(6),BLANK,CST3(6),CST4(6)
808  FORMAT (5X,13HSTANDBY POWER,12X,F7.3,5X,A5,2X,5X,F7.3,4X,F8.3)
      CST1(7) = UREC1(7)
      CST2(7) = 0.
      CST3(7) = 0.
      CST4(7) = UREC1(7)
      WRITE (6,809) CST1(7),BLANK,BLANK,CST4(7)
809  FORMAT (5X,14HTEST EQUIPMENT,11X,F7.3,5X,A5,2X,5X,A5,2X,4X,F8.3)
      CST1(8) = 0.
      CST2(8) = NREC1(6)
      CST3(8) = 0.
      CST4(8) = NREC1(6)
      WRITE (6,810) BLANK,CST2(8),BLANK,CST4(8)

```

```

810 FORMAT(5X, 23HSYST DES, INTEG, + MGMT
      *2X,A5,2X,5X,F7.3,5X,A5,2X,4X,F8.3)
      CST1(9) = 0.
      CST2(9) = 0.
      CST3(9) = OPC1(9)
      CST4(9) = OPC1(9) * LSYS
      WRITE(6,811) BLANK,BLANK,CST3(9),CST4(9)
811 FORMAT (5X,13HSYST OPS MGMT ,12X,
      *2X,5X,F7.3,4X,F8.3)
      WRITE(6,805)
805 FORMAT(30X,7H-----,5X,7H-----,5X,7H-----,4X,8H-----)
      CALL ZERO (TOTC,4,1)
      DO 2000 I = 1,9
      TOTC(1) = TOTC(1) + CST1(I)
      TOTC(2) = TOTC(2) + CST2(I)
      TOTC(3) = TOTC(3) + CST3(I)
      TOTC(4) = TOTC(4) + CST4(I)
2000 CONTINUE
      WRITE(6,804) TOTC(1),TOTC(2),TOTC(3),TOTC(4)
804 FORMAT(13X,9HTOTAL/FAC,7X,F8.3,4X,F8.3,4X,F8.3,3X,F9.3)
      CST1(5) = 1.2 * UREC1(5) * NF
      CST2(5) = NREC1(5) * NF
      CST3(5) = (OPC1(6) + OPC1(7)) * NF
      CST4(5) = (CST1(5) + CST3(5) * LSYS) + CST2(5)
      WRITE (6,807) CST1(5),CST2(5),CST3(5),CST4(5)
807 FORMAT(5X,21HTLM AND CMD EQUIP ,4X,F7.3,5X,F7.3,5X,F7.3,4X,
      * F8.3)
      TOTC(1) = TOTC(1) * NSAT * NF + CST1(5) * NSAT
      TOTC(2) = TOTC(2) + CST2(5)
      TOTC(3) = TOTC(3) * NSAT * NF + CST3(5) * NSAT
      TOTC(4) = (TOTC(1) + TOTC(3) * LSYS) + TOTC(2)
      WRITE(6,885) TOTC(1),TOTC(2),TOTC(3),TOTC(4)
885 FORMAT(8X,12HSYSTEM TOTAL,7X,F10.3,2X,F10.3,2X,F10.3,1X,F11.3)
      WRITE(6,100)
      WRITE(6,900)
900 FORMAT(1X,23HGROUND DOWN-LINK SYSTEM , 4X,11H UNIT ACQ ,2X,
      *12H UNIT INST ,3X,11H UNIT MAINT,4X,5HTOTAL,4X,4HTYPE,8X,
      *11HDESCRIPTIVE/32X,4HCOST,10X,4HCOST,11X,4HCOST,6X,4HCOST,17X,
      *10HPARAMETERS)
      X1 = 10. * ALOG10(G(4))
      IF(I21.EQ.1) GO TO 916
      WRITE(6,917)
917 FORMAT(28X,10H(THOUSNDS),5X,10H(THOUSNDS),2X,13H(THOUSNDS/YR),
      * 1X,10H(MILLIONS))
      CST1(1) = UREC43(1) * 1.E+03 * GR
      CST2(1) = CINANT * 1.E+03
      CST3(1) = 0.15 * CST1(1)
      CST4(1) = (CST1(1) + CST2(1) + CST3(1)*LSYS)*XNGR*1.E-03
      GO TO 919
916 WRITE(6,915)
915 FORMAT(29X,9H(DOLLARS),6X,9H(DOLLARS),3X,12H(DOLLARS/YR),1X,10H(MI
      *LLIONS))
918 CST1(1) = UREC41(1) * 1.E+06 * GR
      CST2(1) = CINANT * 1.E+06
      CST3(1) = 0.1*CST1(1)

```

```

CST4(1) = (CST1(1) + CST2(1) + CST3(1)*LSYS)*XNGR*1.E-06
919 WRITE(6,901) CST1(1),CST2(1),CST3(1),CST4(1),ANTYPE(I16),X1,D(4)
901 FORMAT(/5X,7HANTENNA,9X,F15.2,5X,F9.2,5X,F9.2,3X,F10.3,3X,A10,2X,6
*HGAIN =F5.1,3H DB,17H, EFFECTIVE DIA =F5.1,3H FT)
Q = Y(M+3) / 290. + 1.
X1 = 10. * ALOG10(Q)
IF(I21.EQ,1) GO TO 922
CST1(2) = (UREC43(2)*GR85 + UREC43(6)*GR95) * 1.E+03
CST2(2) = (CINREC + CINTL) * 1.E+03
CST3(2) = OPC43(2) * 1.E+03
CST4(2) = (CST1(2)+CST2(2)+CST3(2)*LSYS)*1.E-03*XNGR
GO TO 920
922 CST1(2) = UCOSTR * 1.E+06
CST2(2) = (CINREC + CINTL) * 1.E+06
CST3(2) = .1 * CST1(2)
CST4(2) = (CST1(2) + CST2(2) + CST3(2)*LSYS) * XNGR
CST4(2) = CST4(2) * 1.E-06
920 WRITE(6,903) CST1(2),CST2(2),CST3(2),CST4(2),X1,Y(M+3)
903 FORMAT(5X,8HRECEIVER,14X,F9.2,5X,F9.2,5X,F9.2,3X,F10.3,15X,
*HNOISE FIG = F5.1,15H, NOISE TEMP = F7.1)
CST1(3) = CST2(3) = CST3(3) = CST4(3) = 0.0
IF(I21.EQ,1) GO TO 906
CST1(3) = UREC43(7) * 1.E+03
CST2(3) = CINDIS * 1.E+03
CST3(3) = OPC43(6) * 1.E+03
CST4(3) = (CST1(3) + CST2(3) + CST3(3)*LSYS)*XNGR*1.E-03
WRITE(6,921) CST1(3),CST2(3),CST3(3),CST4(3),CABL,XNGR
921 FORMAT(5X,11HCABLE DISTR,10X,F10.2,4X,F10.2,4X,F10.2,2X,F11.3,15X,
* 10HCAB LNTH = , F6.0, 18H MI, CATV SYS/BM = , F6.0)
906 WRITE(6,905)
905 FORMAT(30X,6H-----,8X,6H-----,8X,6H-----,6X,7H-----)
CALL ZERO (TOTC,4,1)
DO 3000 I = 1,3
TOTC(1) = TOTC(1) + CST1(I)
TOTC(2) = TOTC(2) + CST2(I)
TOTC(3) = TOTC(3) + CST3(I)
TOTC(4) = TOTC(4) + CST4(I)
3000 CONTINUE
WRITE(6,910) TOTC(1),TOTC(2),TOTC(3),TOTC(4)
910 FORMAT(12X,10HTOTAL/BEAM,4X,F10.2,4X,F10.2,4X,F10.2,2X,F11.3)
TOTC(4) = TOTC(4) * NBRQD * NSAT
WRITE(6,985) TOTC(4)
985 FORMAT(8X,12HSYSTEM TOTAL,46X,F11.3)
WRITE(6,1010)
1010 FORMAT( //1X, 14HLAUNCH VEHICLE , 15X, 4HCOST , 9X,
* 12HWEIGHT LIMIT , 7X, 12HVOLUME LIMIT , 15X, 8HCOMMENTS )
XNSL = NSL
PCST = XSS*ACQC3(4)*NSAT*NBRQD/(NBS*NSL) + TNREC(3)*R2*R1
X81 = BWGHT(I1)
X82 = BVOL(I1)
WRITE(6,1025) X80,XC1,XC2
1025 FORMAT (/5X, 19HSATURN V (UNMANNED), 3X, F9.3, 9X, F9.3,
* 9X, F9.3, 10X, 27HINCLUDES RELAUNCHES AND R/D)
WRITE (6,1020) PCST, X81, X82,NSL
1020 FORMAT ( 5X,10HUSER TOTAL,12X,
F9.3, 9X,

```

* F9.3, 9X, F9.3,10X,11HPRORATED - ,I2,18H SATELLITE CONCEPT)

```
986  WRITE (6,986)
      FORMAT (1H1)
      XLA34 = 10.* ALOG10(XLA34)
      XLC34 = 10.* ALOG10(XLC34)
      XLR34 = 10.* ALOG10(XLR34)
      XLI = 10.* ALOG10(XLI)
      XLRH34 = 10.* ALOG10(XLRH34)
      XLMUX = 10.* ALOG10(XLMUX)
      XLT4 = 10.* ALOG10(XLT4)
      COST1 = UREC41(2)
      COST2 = UREC41(2)
      WRITE (6,OUTD)
      XLA34 = 10.**(XLA34/10.)
      XLC34 = 10.**(XLC34/10.)
      XLR34 = 10.**(XLR34/10.)
      XLI = 10.**(XLI/10.)
      XLRH34 = 10.**(XLRH34/10.)
      XLMUX = 10.**(XLMUX/10.)
      XLT4 = 10.**(XLT4/10.)
      RETURN
      END
```

A

APPENDIX A

GENERAL-PURPOSE SYSTEM ROUTINES

The TVBS computer program requires the following general purpose system subroutines which are available for users of the CDC 6400 computer at the Convair facility. Listings of each are provided together with a description of their respective calling sequences.

GENERAL PURPOSE SYSTEM SUBROUTINES

- 1) Subroutine MADD - to form the sum of two matrices
- 2) Subroutine TRS - to compute the transpose of an N by M matrix
- 3) Subroutine MTMPY - to form the product of two matrices
- 4) Subroutine INVL - to form the inverse and determinant of a matrix
- 5) Subroutine ZERO - to clear an N by M array

1) SUBROUTINE MADD

Purpose: to form the sum of two matrices.

$$R = F_1 A + F_2 A_2$$

Usage:

Call MADD(N,M,A₁,F₁,A₂,F₂,R)

where A₁ and A₂ are N by M matrices,

F₁ and F₂ are scalar factors of A₁ and A₂,

and R is the resultant N by M matrix.

2) SUBROUTINE TRS

Purpose: to compute the transpose of an N by M matrix

Usage:

Call TRS (A,ATRS,N,M)

where A is the location of the matrix to be transposed,
ATRS is the location to store the resultant transpose, and
N and M are the dimensions of A.

NOTE: A and ATRS cannot be the same array.

3) SUBROUTINE MTMPY

Purpose: to form the product of two matrices

$$C = A * B$$

Usage:

Call MTMPY (A,B,C,N,M,L)

where A is an N by M matrix,
B is an M by L matrix, and
C is the resultant N by L matrix

NOTE: A and B may be the same matrix, but C must be a separate matrix.

4) SUBROUTINE INVL

Purpose: to calculate the inverse and determinant of any non-singular square matrix, which need not fill its dimensional area. The inverse is stored either in the same matrix or in a different matrix. In the latter case, the input matrix remains unchanged.

Usage:

Call INVL (A,AINV,N,NDA,L,M,D,K)

where A is the matrix to be inverted (input)
AINV is the inverted matrix (may be A) (output)

NDA is the dimension of A and AINV as given
in the calling program (input)
L an integer storage vector of dimension NDA (input)
M as for L (input)
D the determinant of A upon successful exit (output)
K an error flag (output)

K is set as follows:

- 0 The inversion was unsuccessful, due either to A
being singular or NDA being less than N
- 1 The inversion was successful.

NOTE: NDA must be equal to or greater than N.

5) SUBROUTINE ZERO

Purpose: to clear an N by M array

Usage:

Call ZERO (A,N,M)

where A is the array to clear, and N and M are the dimensions of A.

The following library functions are referenced.

<u>FUNCTION</u>	<u>DEFINITION</u>
SIN(X)	Sine X (radians)
ASIN(X)	Arcsine
COS(X)	Cosine X (radians)
ACOS(X)	Arccosine X (radians)
ATAN(X)	Arctangent X (radians)
SQRT(X)	Square root of X
ALOG10(X)	Log to the base 10 of X
EXP(X)	e to the Xth power

Also, the following functions to perform vector operations:

- 1) VMAG
- 2) VDOT

1) VMAG

Purpose: to compute the magnitude of an N - dimensional vector V.

Usage:

$$VM = VMAG(V, N)$$

2) VDOT

Purpose: to compute the scalar product of two vectors

$$S = A \cdot B$$

Usage:

$$S = VDOT(A, B, N)$$

where N is the dimension of vectors A and B.


```

SUBROUTINE INVL(A,AINV,N,NDA,L,M,D,IDONLY)
DIMENSION A(1),AINV(N,N),L(N),M(N)

```

```

C
C   INVERSION AND DETERMINANT OF ANY GIVEN MATRIX.
C

```

```

C   THIS ROUTINE WILL INVERT ANY NON-SINGULAR SQUARE MATRIX, WHICH
C   NEED NOT FILL ITS DIMENSIONED AREA.
C   THE ANSWER IS STORED EITHER IN THE SAME MATRIX OR IN A DIFFERENT
C   MATRIX. IN THE LATTER CASE, THE INPUT MATRIX REMAINS UNCHANGED.
C

```

```

C   THE MEANING OF THE ARGUMENTS IS AS FOLLOWS---
C

```

```

C       A       THE MATRIX TO BE INVERTED.
C       AINV     THE ANSWER MATRIX (MAY BE MATRIX A).
C       N       THE SIZE OF MATRIX A(N*N).
C       NDA      THE DIMENSION OF MATRICES A AND AINV(NDA,NDA).
C               NDA MUST BE EQUAL TO OR GREATER THAN N.
C       L       AN INTEGER STORAGE VECTOR WITH DIMENSION NDA.
C       M       AS FOR L
C       D       THE DETERMINANT OF MATRIX A
C       IDONLY   AN ERROR FLAG, SET AS FOLLOWS---
C               0   THE INVERSION WAS UNSUCCESSFUL, DUE EITHER TO
C                   THE INPUT MATRIX (A) BEING SINGULAR, OR
C                   NDA BEING LESS THAN N.
C               1   THE INVERSION WAS SUCCESSFUL.
C

```

```

C       IF(NDA-N)3,4,4
3  IDONLY=0
   RETURN
4  J=0
   DO 8 K=1,N
   DO 5 I=1,N
   J=J+1
   KOP=(K-1)*NDA+I
   AINV(J)=A(KOP)
5  CONTINUE
8  CONTINUE
   D=1.0
   IDONLY = 1
   IF (D.EQ.0.0) IDONLY = 0
   D=1.0
   NK=-N
   DO 80 K=1,N
   NK=NK+N
   L(K)=K
   M(K)=K
   KK=NK+K
   BIGA=AINV(KK)
   DO 20 J=K,N
   IZ=N*(J-1)
   DO 20 I=K,N
   IJ=IZ+I
10  IF ( ABS(BIGA)-ABS(AINV(IJ)))15,20,20
15  BIGA=AINV(IJ)
   L(K)=I
   M(K)=J
20  CONTINUE

```

```

C   INTERCHANGE ROWS
C

```

```

      J=L(K)
      IF(J-K) 35,35,25
25  KI=K-N
      DO 30 I=1,N
      KI=KI+N
      HOLD=-AINV(KI)
      JI=KI-K+J
      AINV(KI)=AINV(JI)
30  AINV(JI)=HOLD
C
C      INTERCHANGE COLUMNS
C
35  I=M(K)
      IF(I-K) 45,45,38
38  JP=N*(I-1)
      DO 40 J=1,N
      JK=NK+J
      JI=JP+J
      HOLD=-AINV(JK)
      AINV(JK)=AINV(JI)
40  AINV(JI)=HOLD
C
C      DIVIDE COLUMN BY MINUS PIVOT
C
45  IF(BIGA)48,46,48
46  D=0.0
      IDONLY=0
      RETURN
48  DO 55 I=1,N
      IF(I-K) 50,55,50
50  IK=NK+I
      AINV(IK)=AINV(IK)/(-BIGA)
55  CONTINUE
C
C      REDUCE MATRIX
      DO 65 I=1,N
      IK=NK+I
      IJ=I-N
      DO 65 J=1,N
      IJ = IJ + N
      IF(I-K) 60,65,60
60  IF(J-K) 62,65,62
62  KJ=IJ-I+K
      AINV(IJ)=AINV(IK)*AINV(KJ)+AINV(IJ)
65  CONTINUE
C
C      DIVIDE ROW BY PIVOT
C
      KJ=K-N
      DO 75 J=1,N
      KJ=KJ+N
      IF(J-K) 70,75,70
70  AINV(KJ)=AINV(KJ)/BIGA

```

```

75 CONTINUE
C
C   PRODUCT OF PIVOTS
C
D=D*BIGA
C
C   REPLACE PIVOT BY RECIPROCAL
C
AINV(KK)=1.0/BIGA
80 CONTINUE
IF (IDONLY.EQ.0) RETURN
C
C   FINAL ROW AND COLUMN INTERCHANGE
C
K=N
100 K=(K-1)
IF (K) 150,150,105
105 I=L(K)
IF (I-K) 120,120,108
108 JQ=N*(K-1)
JR=N*(I-1)
DO 110 J=1,N
JK=JQ+J
HOLD=AINV(JK)
JI=JR+J
AINV(JK)=-AINV(JI)
110 AINV(JI)=HOLD
120 J=M(K)
IF (J-K) 100,100,125
125 KI=K-N
DO 130 I=1,N
KI=KI+N
HOLD=AINV(KI)
JI=KI-K+J
AINV(KI)=-AINV(JI)
130 AINV(JI)=HOLD
GO TO 100
150 IF (NDA-N) 152,235,155
152 IDONLY=0
GO TO 235
155 K2=N
M1=(N-1)*NDA+N+1
K5=M1-1+N*(N-1)
K6=NDA*NDA
IF (K5-K6) 175,165,165
165 K1=K6
GO TO 185
175 K1=K5
185 M3=K1-1
DO 195 K4=M1,K1
K2=K2+1
AINV(K4)=AINV(K2)
195 CONTINUE
IF (K5-K6) 197,197,202

```

```
197 M1=M1-1
    DO 200 K=2,N
    DO 108 J=1,N
    KOP=(K-1)*NDA+J
    M1=M1+1
    AINV(KOP)=AINV(M1)
198 CONTINUE
200 CONTINUE
    GO TO 235
202 DO 225 K=2,N
    DO 215 J=1,N
    KOP=(K-1)*NDA+J
    K2=K2+1
    TEMP=AINV(K2)
    AINV(KOP)=AINV(M1)
    DO 205 I=M1,M3
    AINV(I)=AINV(I+1)
205 CONTINUE
    AINV(K1)=TEMP
215 CONTINUE
225 CONTINUE
235 RETURN
    END
```

```

SUBROUTINE MADD(N,M,A1,F1,A2,F2,R)
C SUBROUTINE TO ADD 2 N BY M MATRICES
C A1 AND A2 ARE N BY M MATRICES
C R IS THE RESULTANT MATRIX
C F1 AND F2 ARE SCALAR FACTORS OF A1 AND A2
DIMENSION A1(1), A2(1), R(1)
M1 = M*N
DO 100 I=1,M1
100 R(I) = F1*A1(I) + F2*A2(I)
RETURN
END

```

```

SUBROUTINE TRS(A,AT,N,M)
C SUBROUTINE TO COMPUTE THE TRANSPOSE OF AN N BY M MATRIX
DIMENSION A(N,M),AT(M,N)
DO 1 I=1,N
DO 1 J=1,M
1 AT(J,I) = A(I,J)
RETURN
END

```

```

SUBROUTINE MTPY(A,B,C,M,N,L)
C SUBROUTINE TO FORM THE PRODUCT OF TWO MATRICES
C A IS AN M BY N MATRIX
C B IS AN N BY L MATRIX
C C IS THE RESULTANT M BY L MATRIX
DIMENSION A(M,N), B(N,L), C(M,L)
DO 20 I=1,M
DO 20 J=1,L
C(I,J) = 0.0
DO 20 K=1,N
20 C(I,J) = C(I,J) + A(I,K)*B(K,J)
RETURN
END

```

```

SUBROUTINE ZERO(A,N,M)
C TO ZERO OUT MATRIX A, DIMENSIONED N BY M
DIMENSION A(1)
M1=N*M
DO 20 I=1,M1
20 A(I) = 0.0
RETURN
END

```

APPENDIX B

DERIVATION OF STEEPEST DESCENT ITERATIVE ALGORITHM

Let X represent a vector of n independent system design parameters, $y(X)$ a vector of m constrained system parameters, and $C(X)$ the total system cost. It is desired to determine the minimum value of C subject to the constraints imposed on y . This solution is to be obtained by iteratively perturbing a nominal design parameter vector X_i until further reduction in C is possible and all constraints are satisfied. At each iteration only those constraints which have been violated need be considered.

Now let y be a k -vector whose elements are those parameters whose boundaries have been violated (a subset of the original y vector). For these parameters the constraint

$$y(X) - y_{\text{BOUND}} = 0 \quad (1)$$

will be imposed. At a given iteration

$$Y_i = y(X_i) - y_{\text{BOUND}} \neq 0 \quad (2)$$

Let g and Φ represent the cost and constraint gradients, respectively. That is,

$$g = \frac{dC}{dX}(X_i) \text{ and } \Phi = \frac{dy}{dX}(X_i) \quad (3)$$

To the first order terms

$$\Delta C(\Delta X) = g^T \cdot \Delta X \quad (4)$$

$$\text{and } \Delta Y = \Phi^T \cdot \Delta X \quad (5)$$

It is desired to determine the vector ΔX which minimizes

$$J = \Delta C(\Delta X) = g^T \cdot \Delta X \quad (6)$$

while reducing Y by some specified amount

$$\Delta Y = -K_y \cdot Y \quad (7)$$

The magnitude of the perturbation is constrained by

$$S^2 = \Delta X^T \cdot \Delta X \quad (8)$$

Applying Lagrange multiplier theory one defines J' to be

$$J' = g^T \cdot \Delta X + \lambda \cdot (\Delta X^T \cdot \Delta X - S^2) + \eta^T \cdot (\Phi^T \cdot \Delta X + K_y \cdot Y) \quad (9)$$

where λ is an arbitrary constant and η is an arbitrary k -vector. From Equations 7 and 8 it is clear that J and J' are equal, so one can proceed by minimizing J' . This minimum occurs where

$$\frac{dJ'}{d\Delta X} = g^T + 2\lambda \cdot \Delta X^T + \eta^T \Phi^T = 0 \quad (10)$$

Solving Equation (1) for ΔX gives

$$\Delta X = -\frac{1}{2\lambda} \cdot (g + \Phi \cdot \eta) \quad (11)$$

For convenience define

$$K_c = -\frac{1}{2\lambda} \text{ and } \lambda = -\frac{1}{2\lambda} \cdot \eta \quad (12)$$

Then

$$\Delta X = -K_c \cdot g - \Phi \cdot \lambda \quad (13)$$

Substituting Equation (13) into Equations (5) and (7) gives

$$\Delta Y = -K_c \Phi^T g - \Phi^T \Phi \gamma = -K_y \cdot Y \quad (14)$$

Solving for γ gives

$$\gamma = -K_c (\Phi^T \Phi)^{-1} \cdot \Phi^T \cdot g + K_y (\Phi^T \Phi)^{-1} \cdot Y \quad (15)$$

Substituting Equation (15) into Equation (13) gives

$$\Delta X = -K_c \left[I - \Phi (\Phi^T \Phi)^{-1} \Phi^T \right] \cdot g - K_y \cdot \Phi (\Phi^T \Phi)^{-1} \cdot Y \quad (16)$$

which is the perturbation equation given in Section 2.2.

The two terms of Equation (16) represent perturbations parallel and perpendicular to the constraint boundaries, respectively. This can be shown by taking the vector inner product (vector projection) of the first term and the constraint gradient

$$\begin{aligned} \Phi^T \cdot \left\{ -K_c \left[I - \Phi (\Phi^T \Phi)^{-1} \Phi^T \right] g \right\} &= -K_c \left[\Phi^T - (\Phi^T \Phi) (\Phi^T \Phi)^{-1} \Phi^T \right] g \\ &= -K_c \left[\Phi^T - \Phi^T \right] g = 0 \end{aligned} \quad (17)$$

The value of K_c can be determined by substituting Equation (16) into Equation (8) and reducing to

$$S^2 = K_y^2 Y^T (\Phi^T \Phi)^{-1} Y + K_c^2 \cdot H^T H \quad (18)$$

$$\text{where } H = \left[I - \Phi (\Phi^T \Phi)^{-1} \Phi^T \right] g \quad (19)$$

Then

$$K_c = \sqrt{\frac{S^2 - K_y^2 \cdot Y^T (\Phi^T \Phi)^{-1} Y}{H^T H}} \quad (20)$$

The step-size S is controlled to insure rapid convergence to the solution. When the nominal value of X is "far" from the solution successive perturbations will tend to be in the same direction. Under this condition it is desirable to increase S for the following iteration. When the nominal X is "close" to the solution such that a given perturbation overshoots the solution, successive perturbations will tend to be in opposite directions. Under this condition it is desirable to reduce S to allow the minimum cost solution to be determined to a finer resolution. The logic for implementing this step-size control is given in Section 2.2.

C

APPENDIX C

DERIVATION OF THE PARTIAL DERIVATIVES

Because of the appearance of implicit functions in the system equations, computation of the partial derivatives of the subsystem parametric relations required the solution of a set of simultaneous linear equations. This was accomplished as follows:

Let $W_A(V_A)$	= weight (volume) of the antennas
$W_{PP}(V_{PP})$	= weight (volume) of the power supply
$W_{RD}(V_{RD})$	= weight (volume) of the thermal control system
$W_{MP}(V_{MP})$	= weight (volume) of the manned provisions
$W_{TR}(V_{TR})$	= weight (volume) of the transmitter
$W_{REC}(V_{REC})$	= weight (volume) of the receiver
$W_{MPLX}(V_{MPLX})$	= weight (volume) of the multiplexer
$W_{INC}(V_{INC})$	= weight (volume) of the instrumentation and communication
$W_{AC}(V_{AC})$	= weight (volume) of the attitude control
$W_{CS}(V_{CS})$	= weight (volume) of the station keeping (control system)
$W_{STR}(V_{STR})$	= weight (volume) of the structure
$W_{BATT}(V_{BATT})$	= weight (volume) of the battery supply
$W_{PT}(V_{PT})$	= weight (volume) of the power transfer unit
$W_{PC}(V_{PC})$	= weight (volume) of the power conditioning unit
$W_{PCU}(V_{PCU})$	= weight (volume) of the power control unit
$W_{LCF}(V_{LCF})$	= weight (volume) of the l/c filter
W_{HAR}	= weight of the harnessing.

Then

$$W_{EM} = \text{weight of the equipment module}$$

$$= W_{TR} + W_{REC} + W_{MPLX} + W_{INC} + W_{AC} + W_{CS} + W_{STR} + W_{BATT} + W_{COND}$$

where W_{COND} = weight of the power subsystem

$$= W_{PT} + W_{PC} + W_{PCU} + W_{LCF} + W_{HAR}$$

and W_T = total satellite payload weight

$$= W_{EM} + W_A + W_{PP} + W_{RD} + W_{MP}.$$

$$W_S = W_T - W_{CS}$$

The expressions for volume are similar.

The following power relations are derived from the power subsystem model.
(See Appendix D).

$$P_{LCF} = \frac{P_{TR}}{\eta_{TR}}$$

$$P_{TA} = .1 P_{TR}$$

$$P_{PCU} = N_C \cdot \frac{P_{LCF}}{\eta_{LCF}} + N_C \cdot N_{APV} \frac{P_{TA}}{\eta_{TA}} + P_{AC} + P_{CS} + P_{INC} + P_R$$

$$P_{PC} = \frac{P_{PCU}}{\eta_{PCU}}$$

$$P_B = \frac{P_{PC}}{\eta_{PC}}$$

$$P_{PT} = P_B + \rho \frac{P_B}{\eta_B}$$

$$P_{PSD} = \frac{P_{PT}}{\eta_{PT}}$$

$$P_{PS} = \frac{P_{PSD}}{\rho'}$$

where ρ' represents the adjustment to the prime power supply as a result of degradation over the span of lifetime of the power supply.

By expanding $\frac{\partial P_{PS}}{\partial X_i}$ and $\frac{\partial V_{EM}}{\partial X_i}$, the following equations result:

$$K_1 \frac{\partial P_{PS}}{\partial X_i} + K_2 \frac{\partial V_{EM}}{\partial X_i} = K_3, \quad i = 2, 3 \quad (1)$$

$$K'_1 \frac{\partial P_{PS}}{\partial X_i} + K'_2 \frac{\partial V_{EM}}{\partial X_i} = K'_3, \quad i = 2, 3 \quad (2)$$

where $X_2 = D_4$ and $X_3 = T_R$

Solving this system all of the remaining partials can be derived.

Solution of this set of equations provide the values of $\frac{\partial P_{PS}}{\partial X_i}$ and $\frac{\partial V_{EM}}{\partial X_i}$

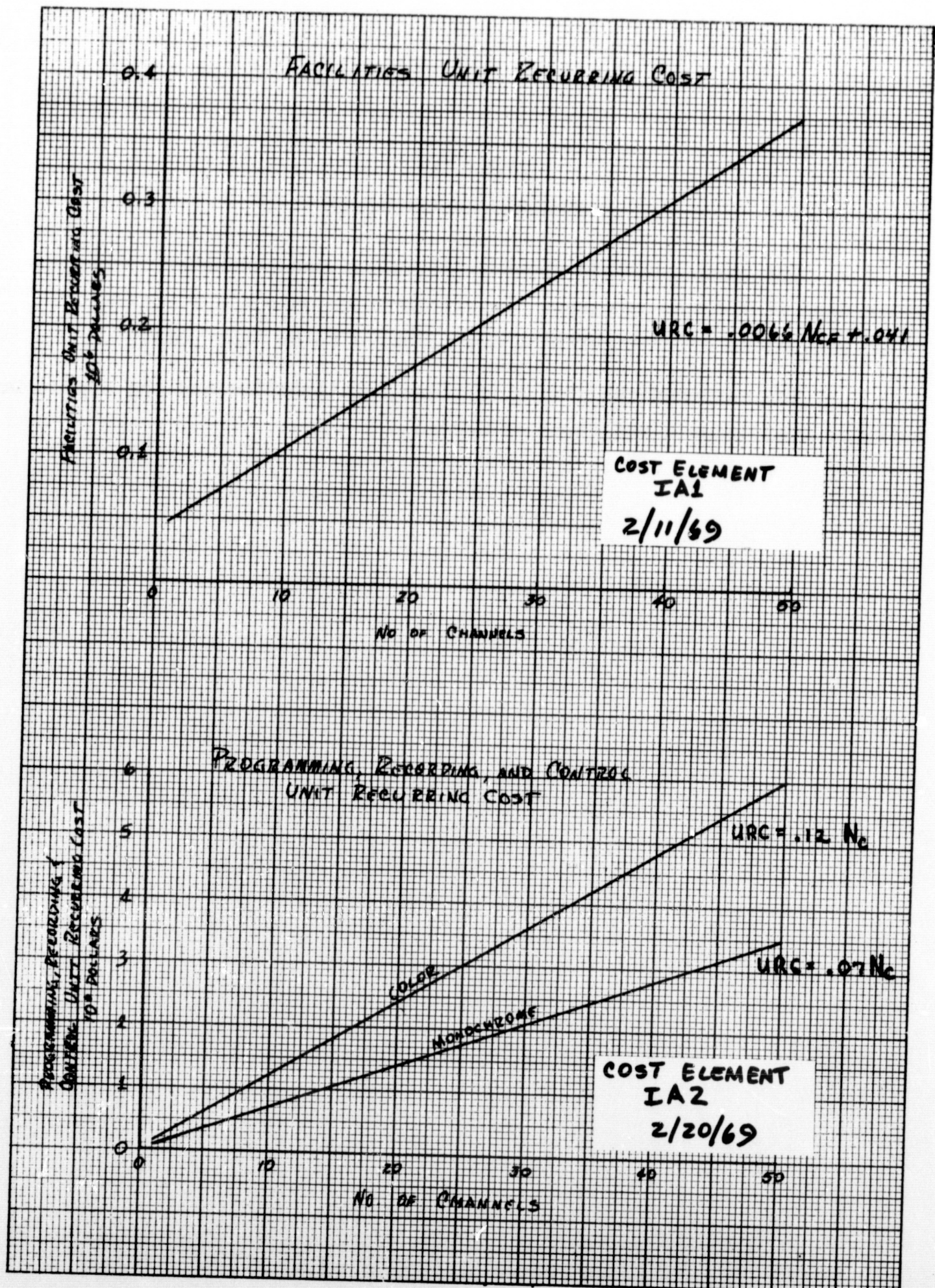
from which all other subsystem partial derivatives can be obtained.

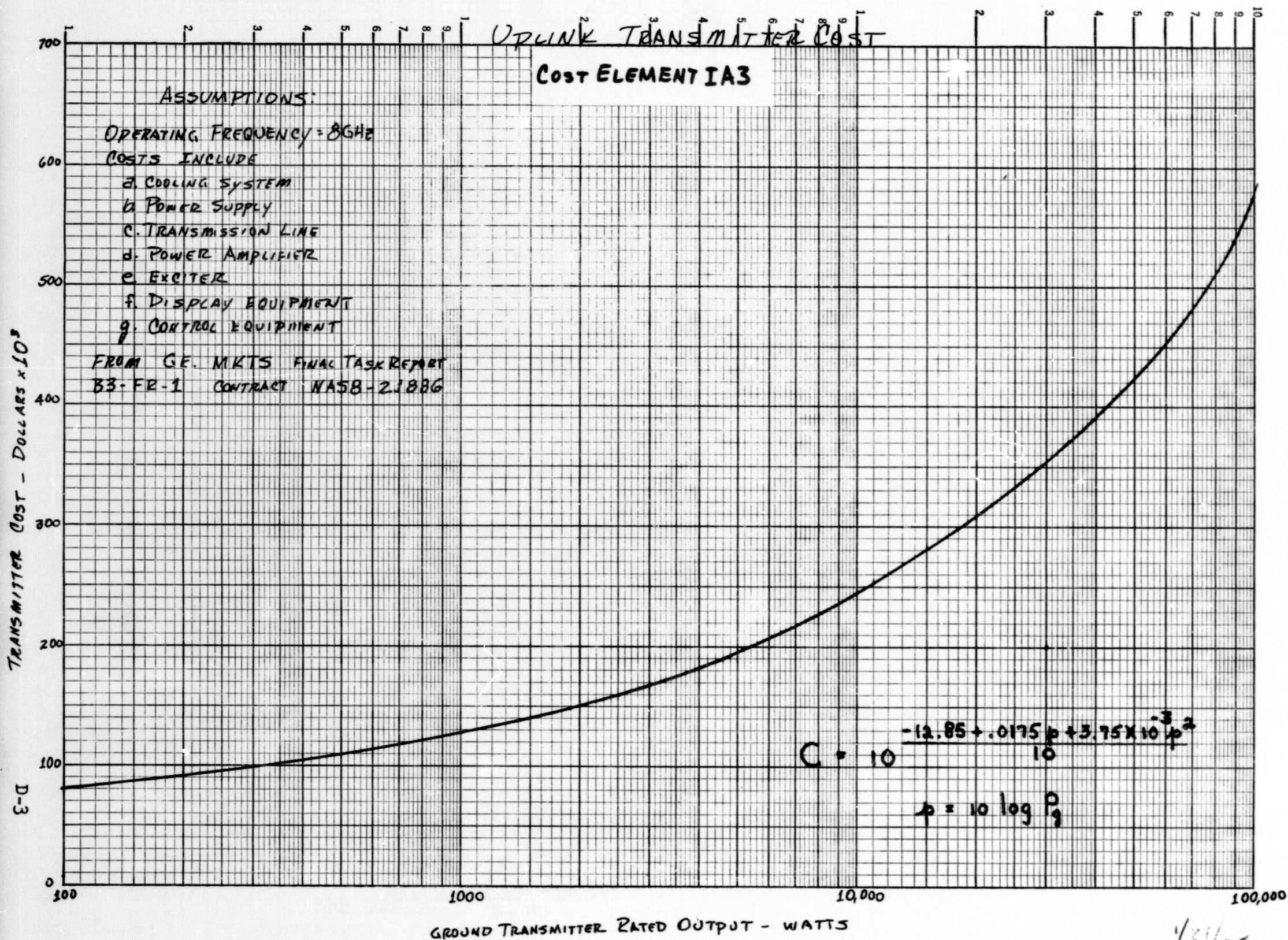
Solution of these equations is performed in subroutine PRTIAL (Section 5.6).

D**APPENDIX D
PARAMETRIC RELATIONSHIP**

The curves in this appendix represent the parametric relationships used in the TVBS program. Functions which maintain a constant value regardless of configuration are referred to as "thruputs."

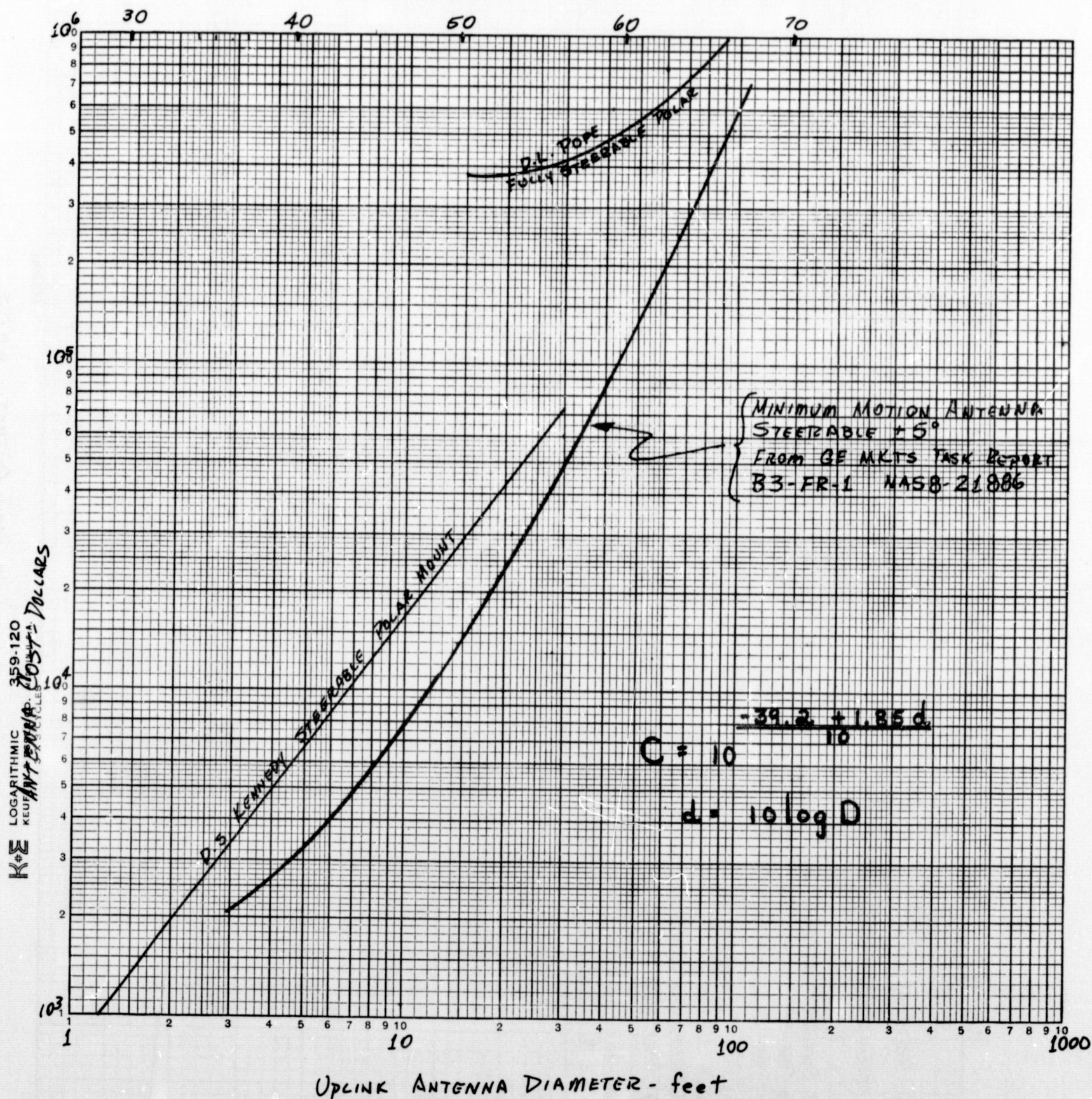
The mathematical models of these relationships and the appropriate analytic coefficients are given in Sections 3 and 4.

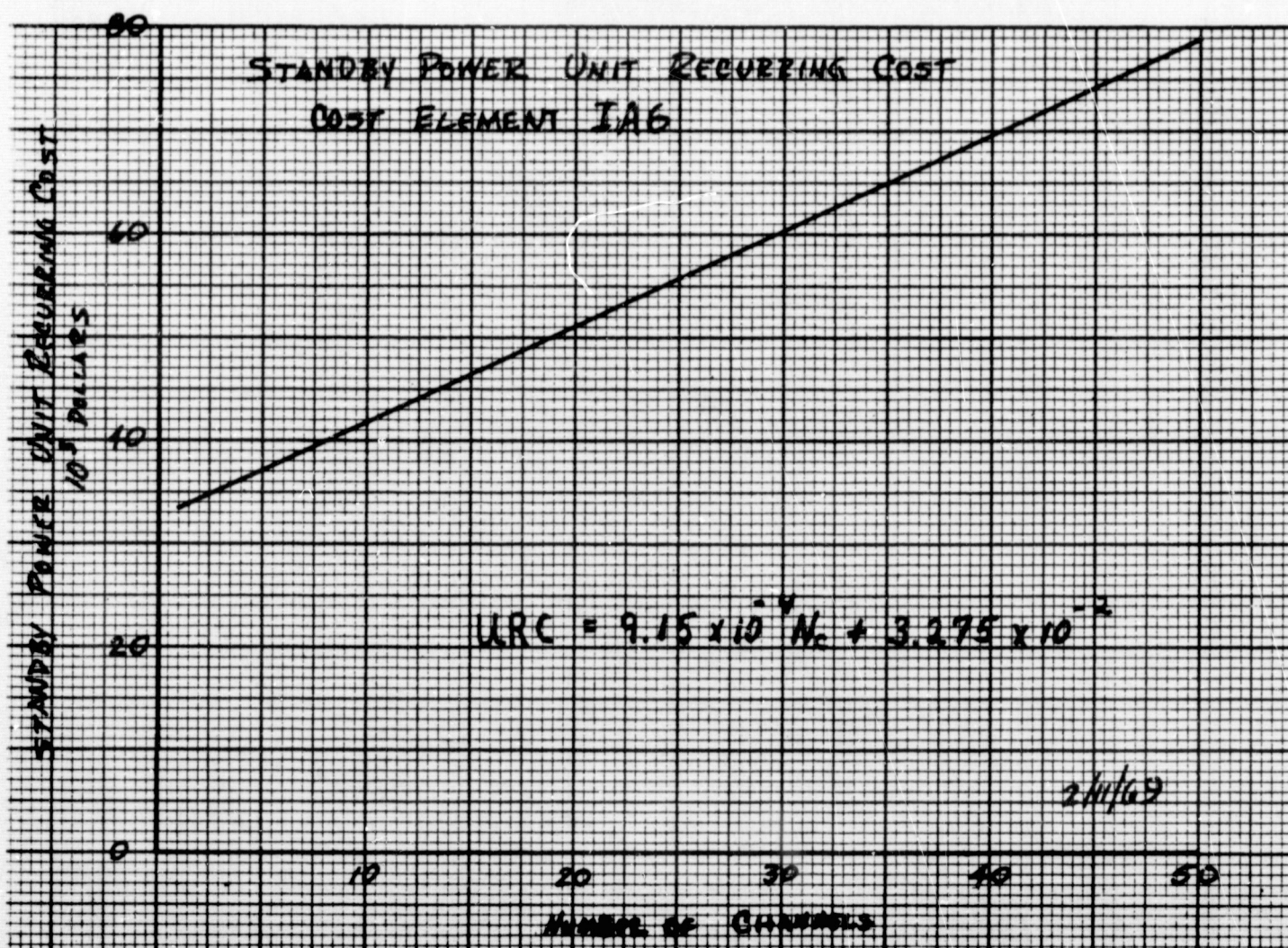


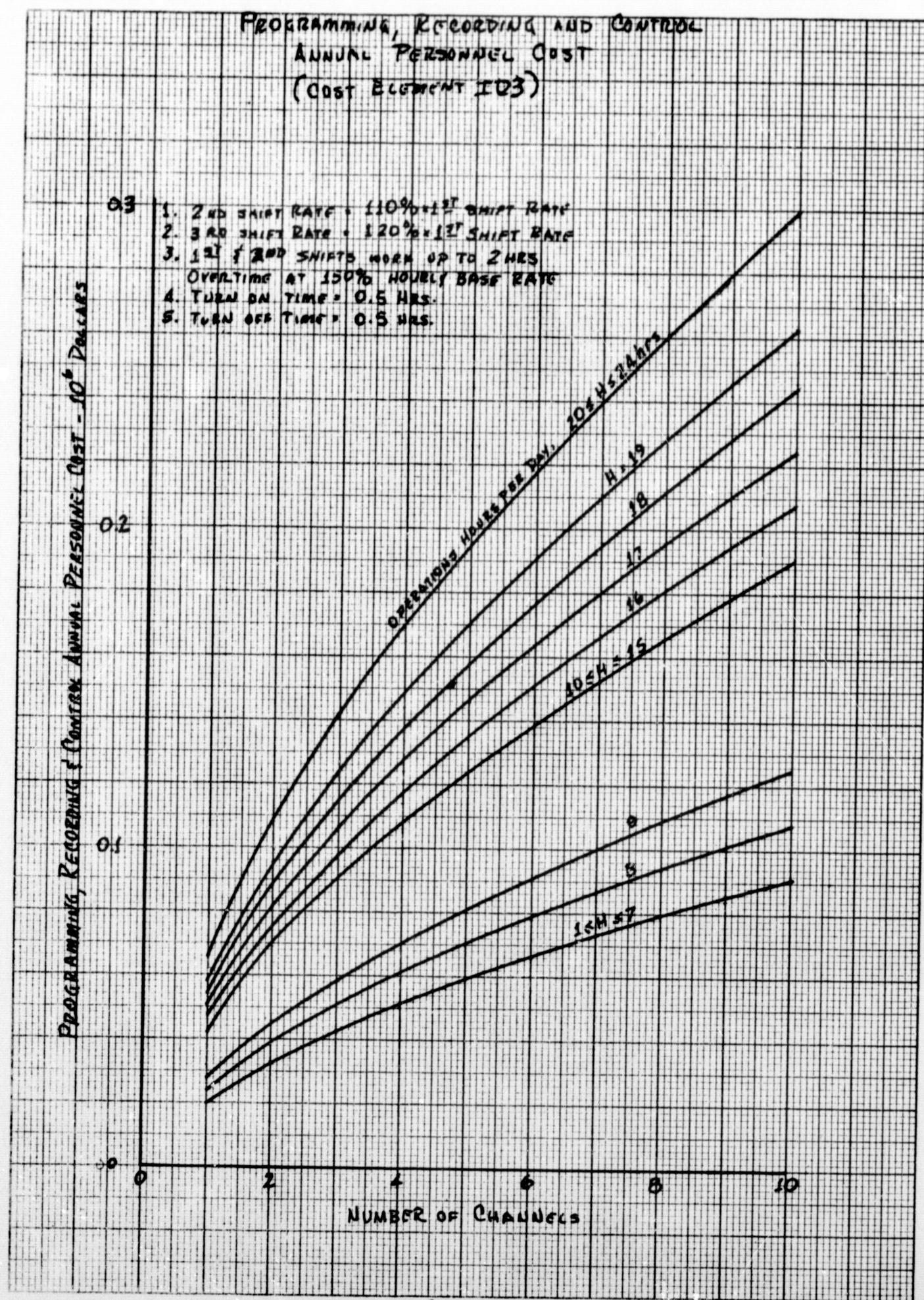


UPLINK ANTENNA COST COST ELEMENT IAA

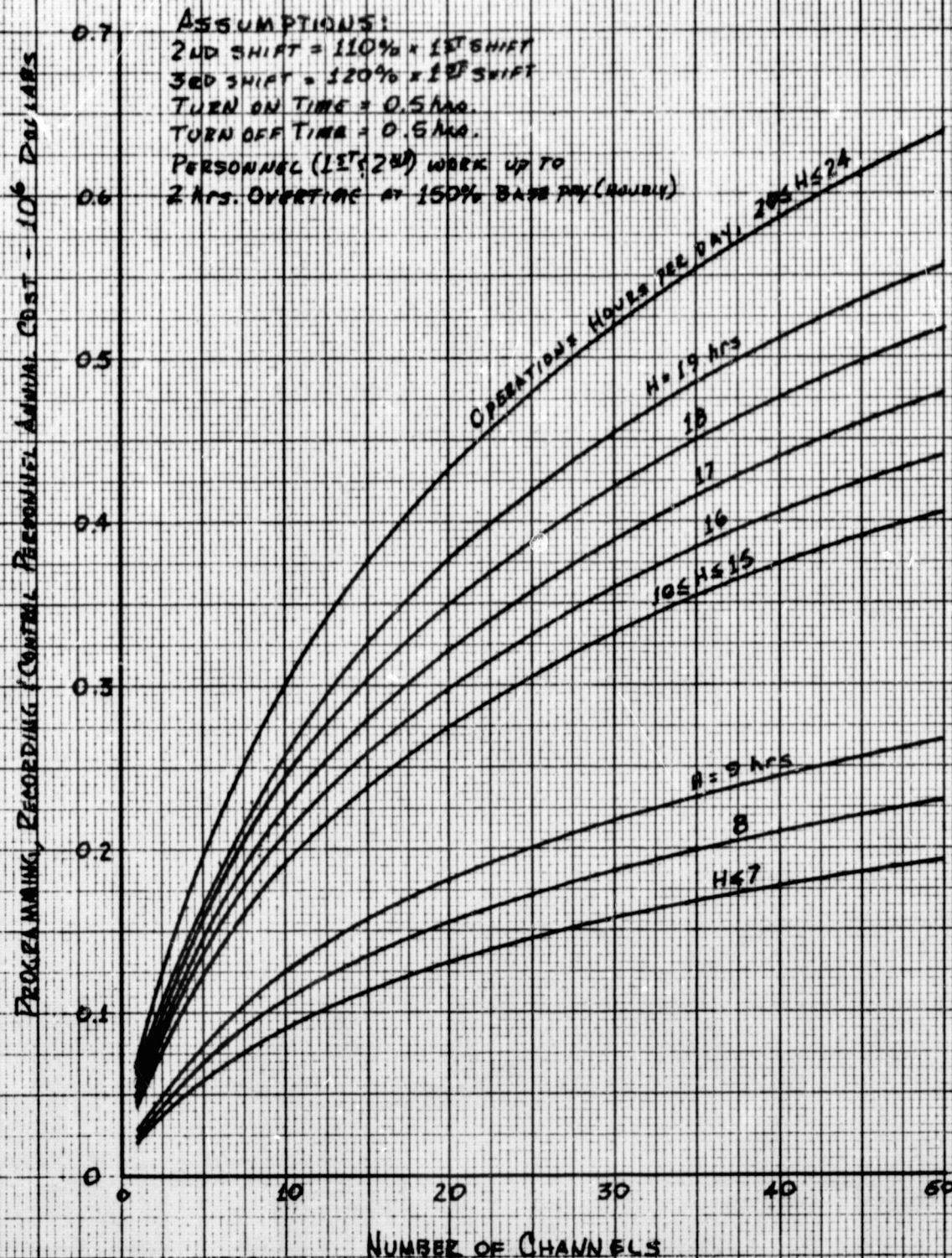
GAIN AT 8 GHz - dB



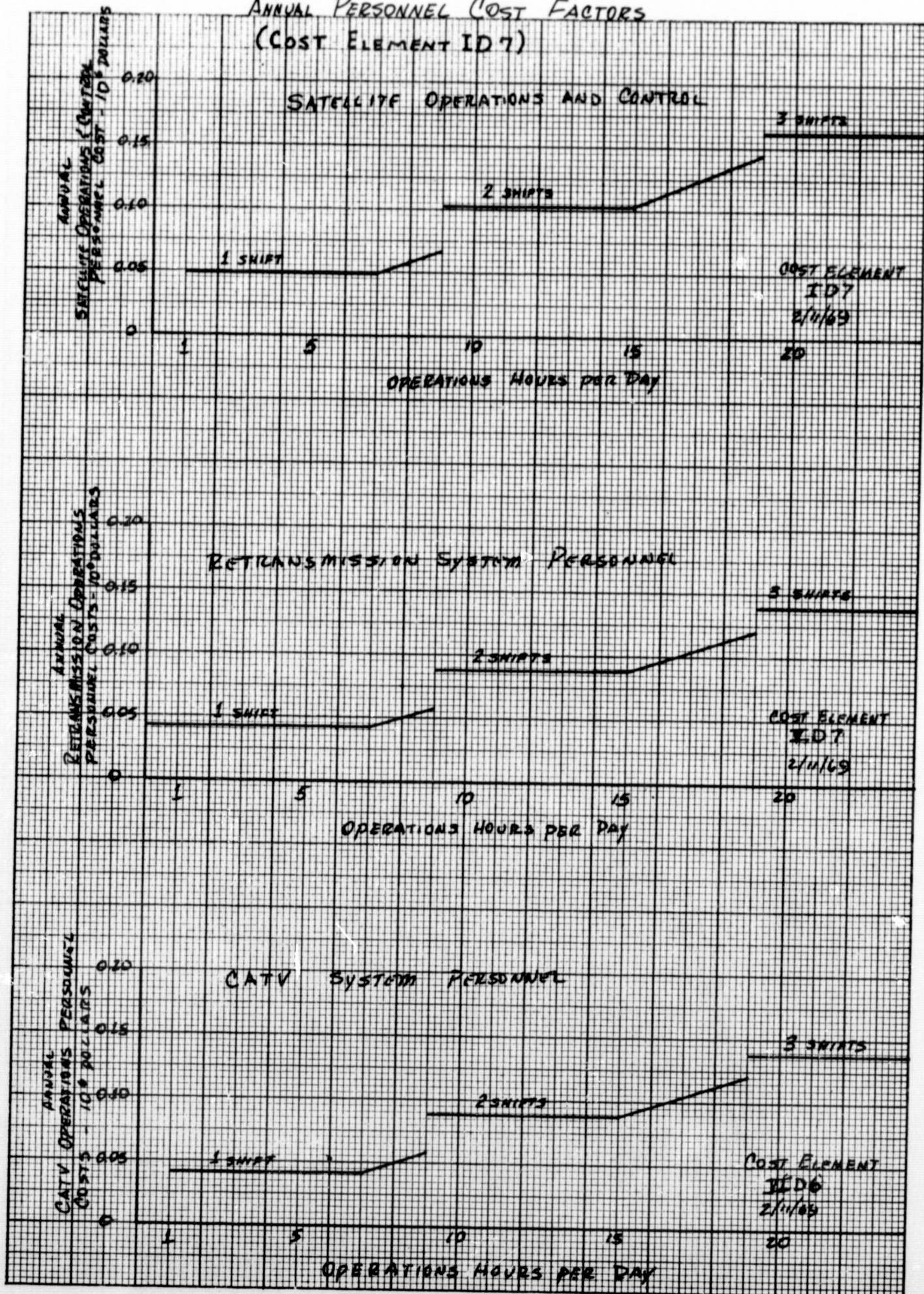




PROGRAMMING, RECORDING & CONTROL ANNUAL PERSONNEL COST (COST ELEMENT ID3)

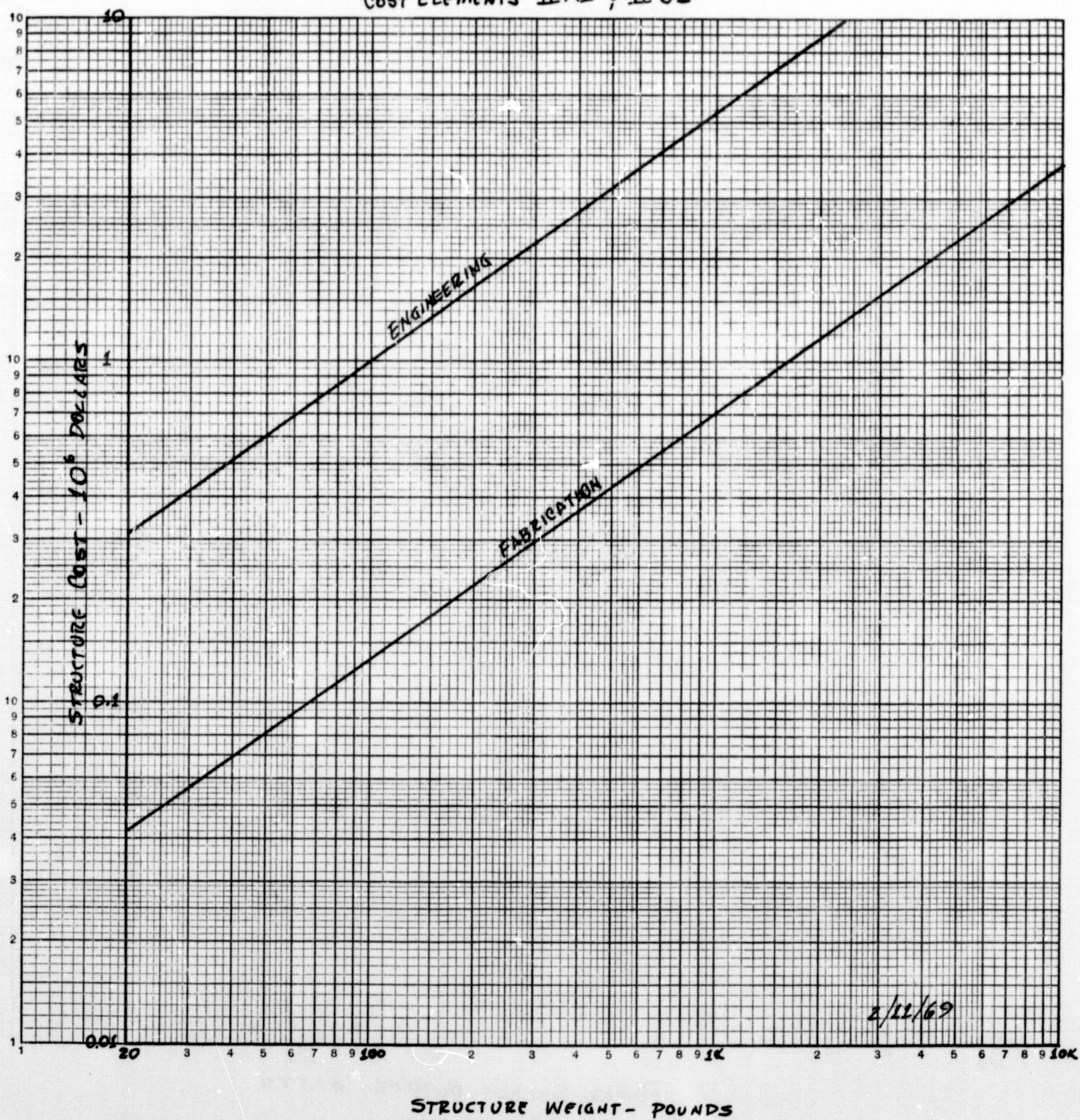


ANNUAL PERSONNEL COST FACTORS (COST ELEMENT ID 7)

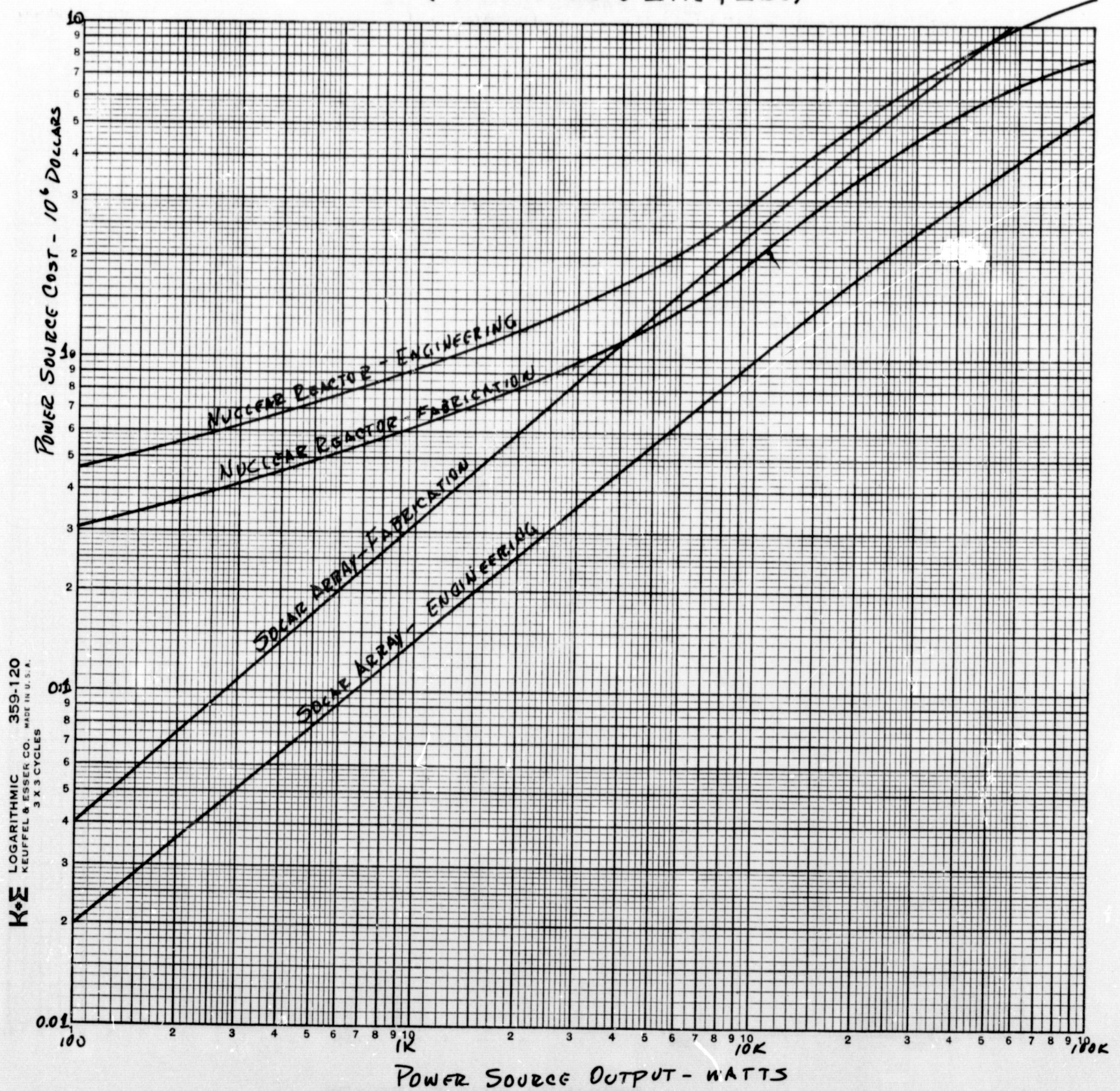


STRUCTURE COSTS

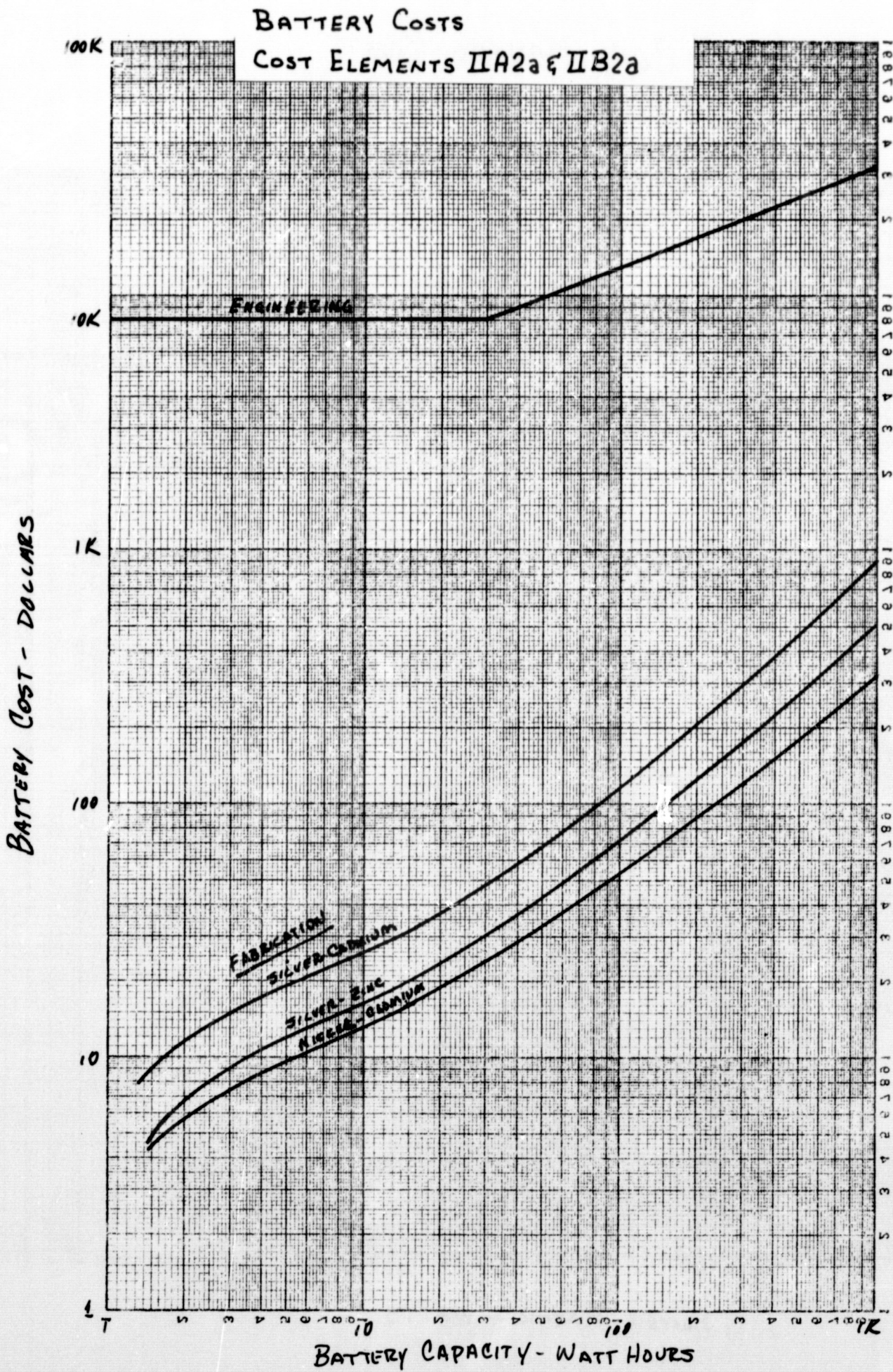
COST ELEMENTS IIA1 & IIB1



POWER SOURCE COST (COST ELEMENTS IIA2 & IIB2)



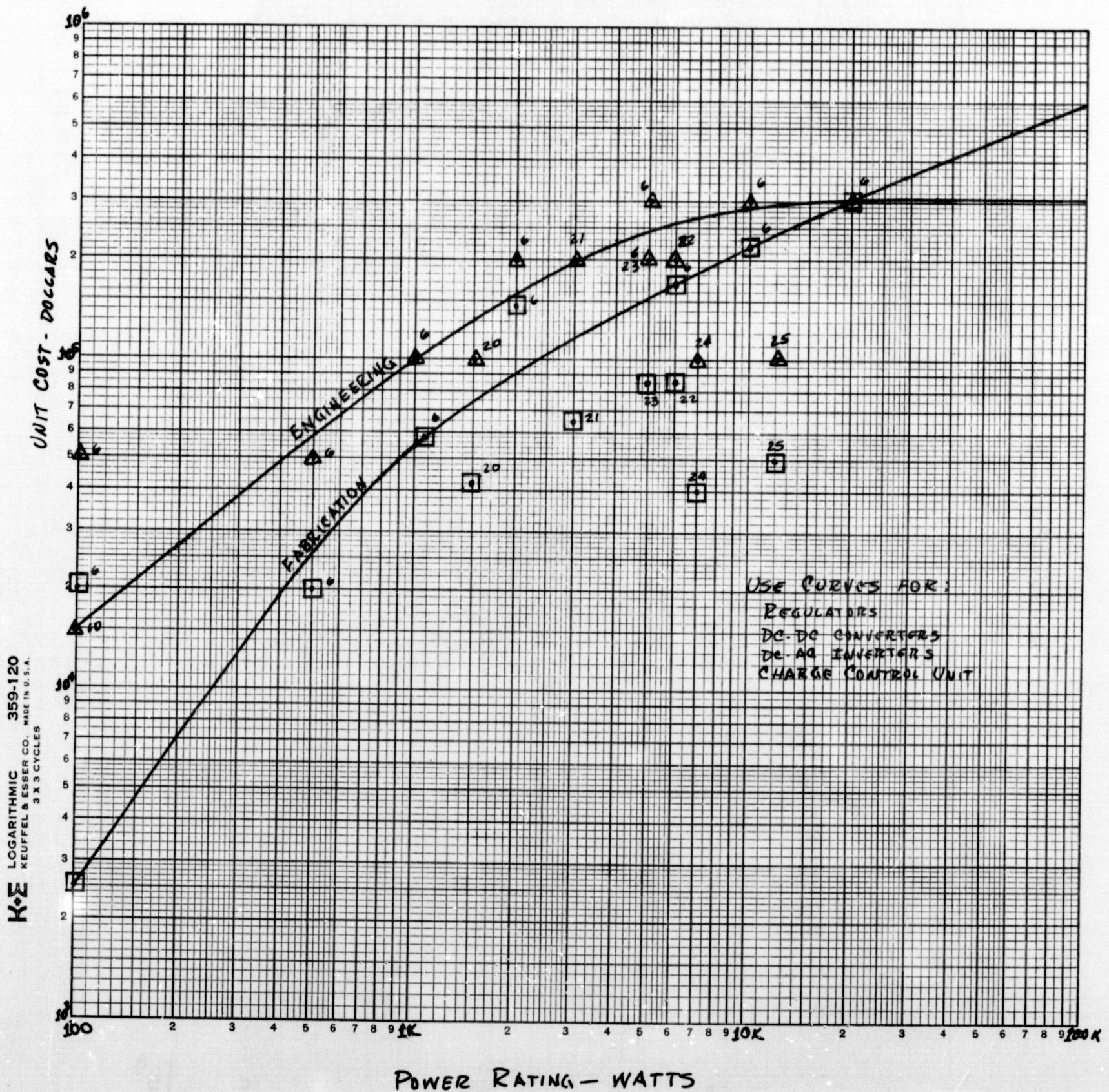
3/31/69



2/26/69

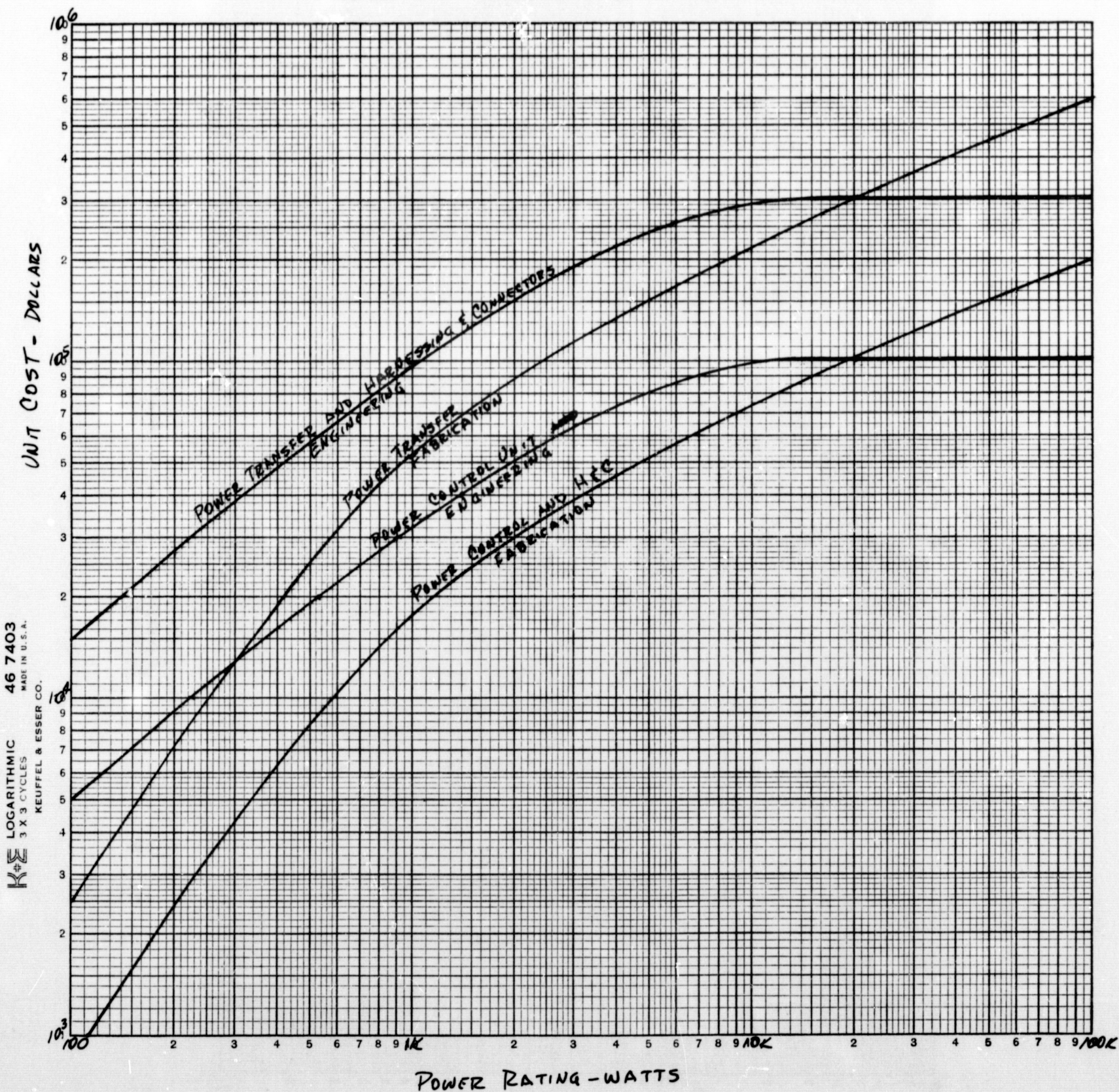
POWER CONDITIONING COST

COST ELEMENTS IIA2 AND IIB2

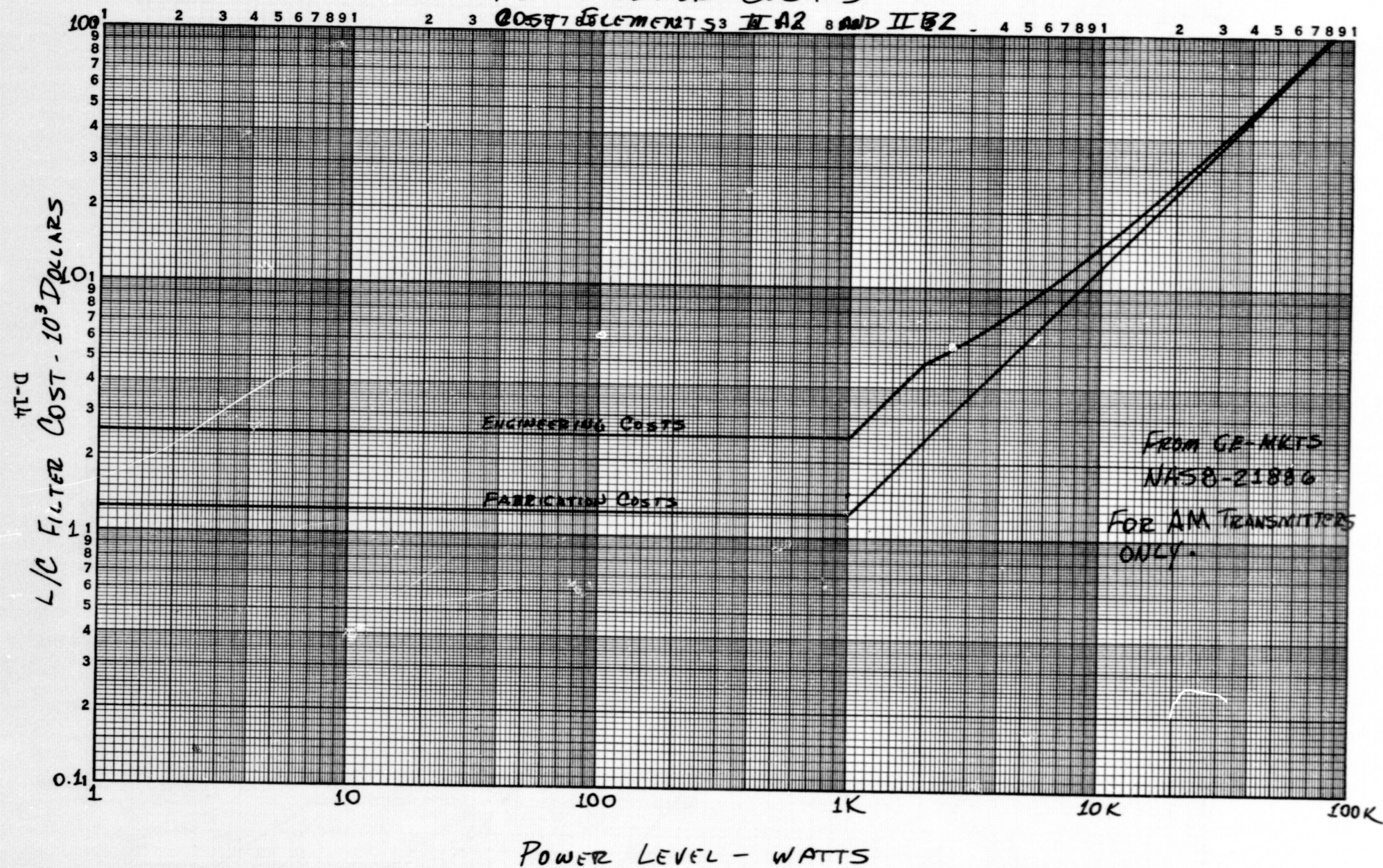


3/20/69

POWER DISTRIBUTION COSTS COST ELEMENTS IIA2 AND IIB2

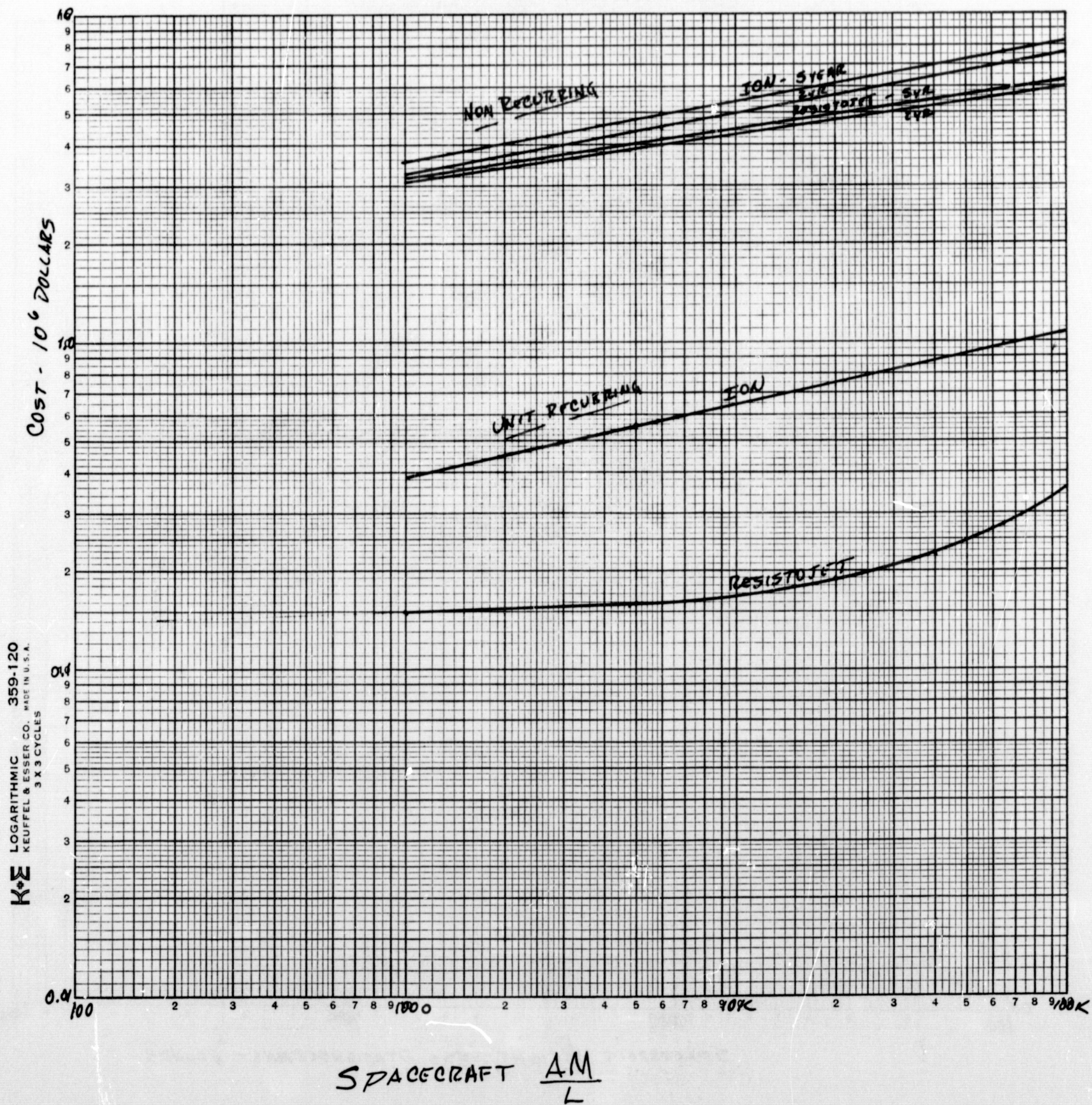


L/C FILTER COSTS

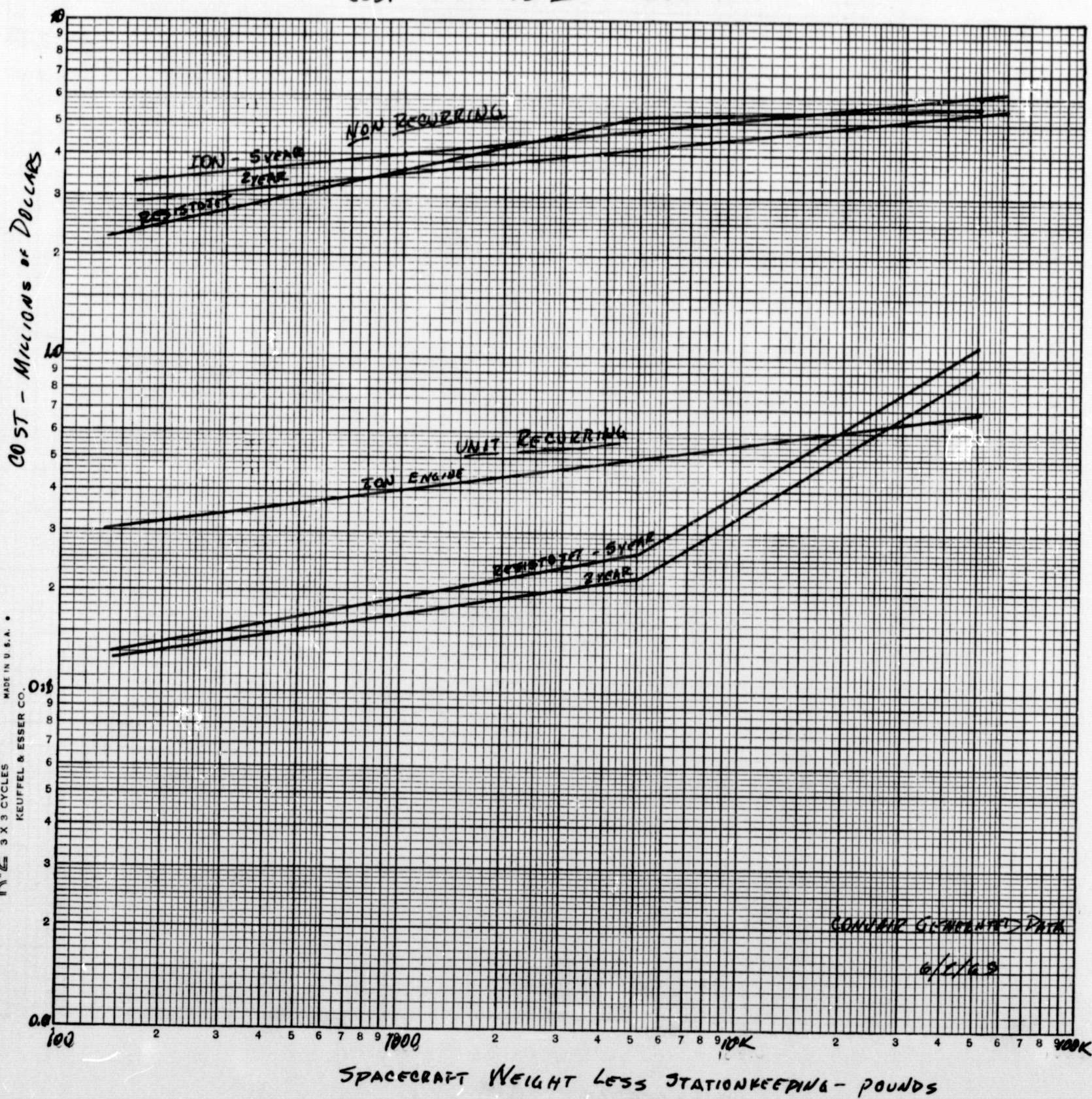


4/7/69

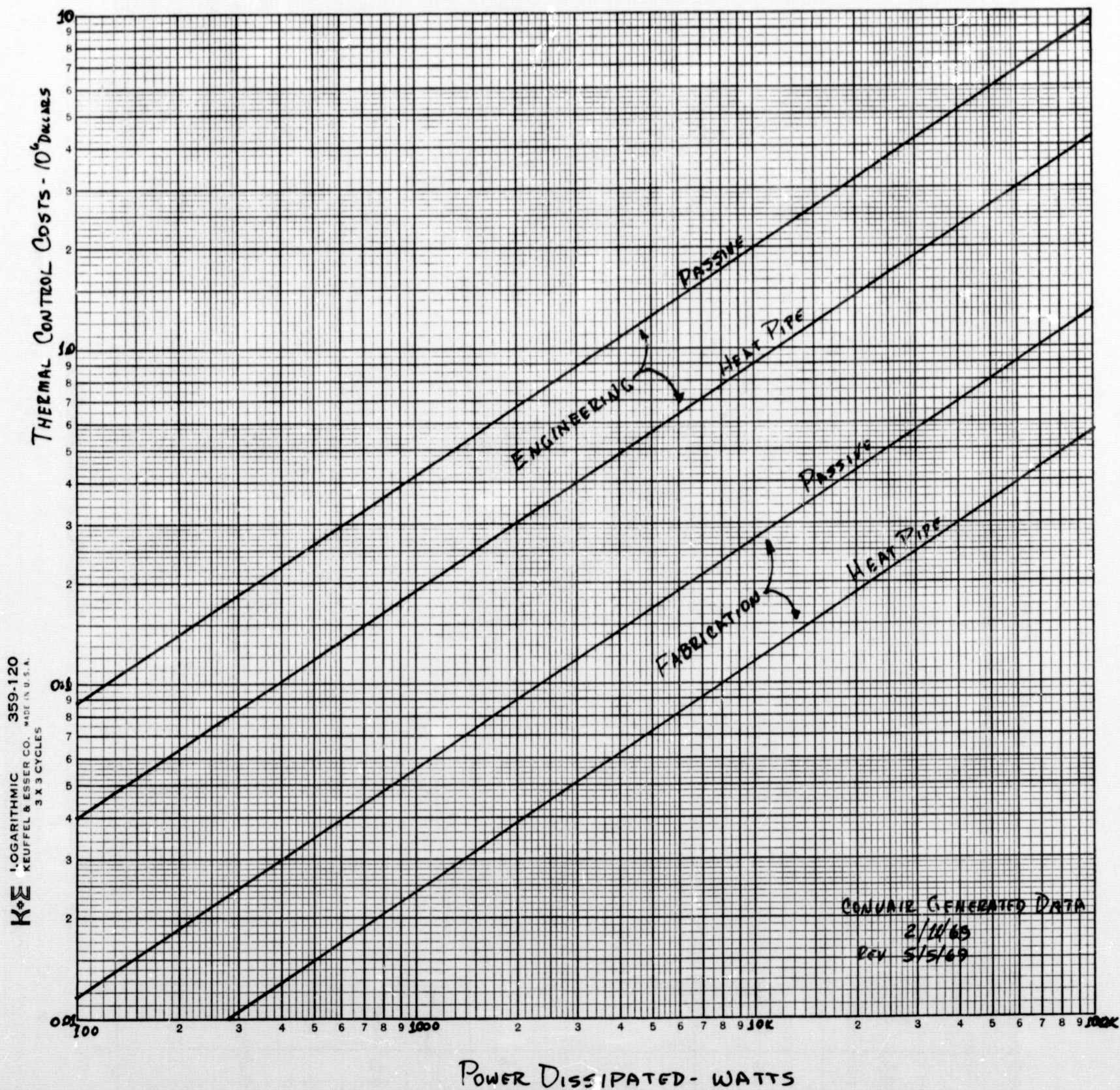
ATTITUDE CONTROL COSTS COST ELEMENTS IIA3 AND IIB3

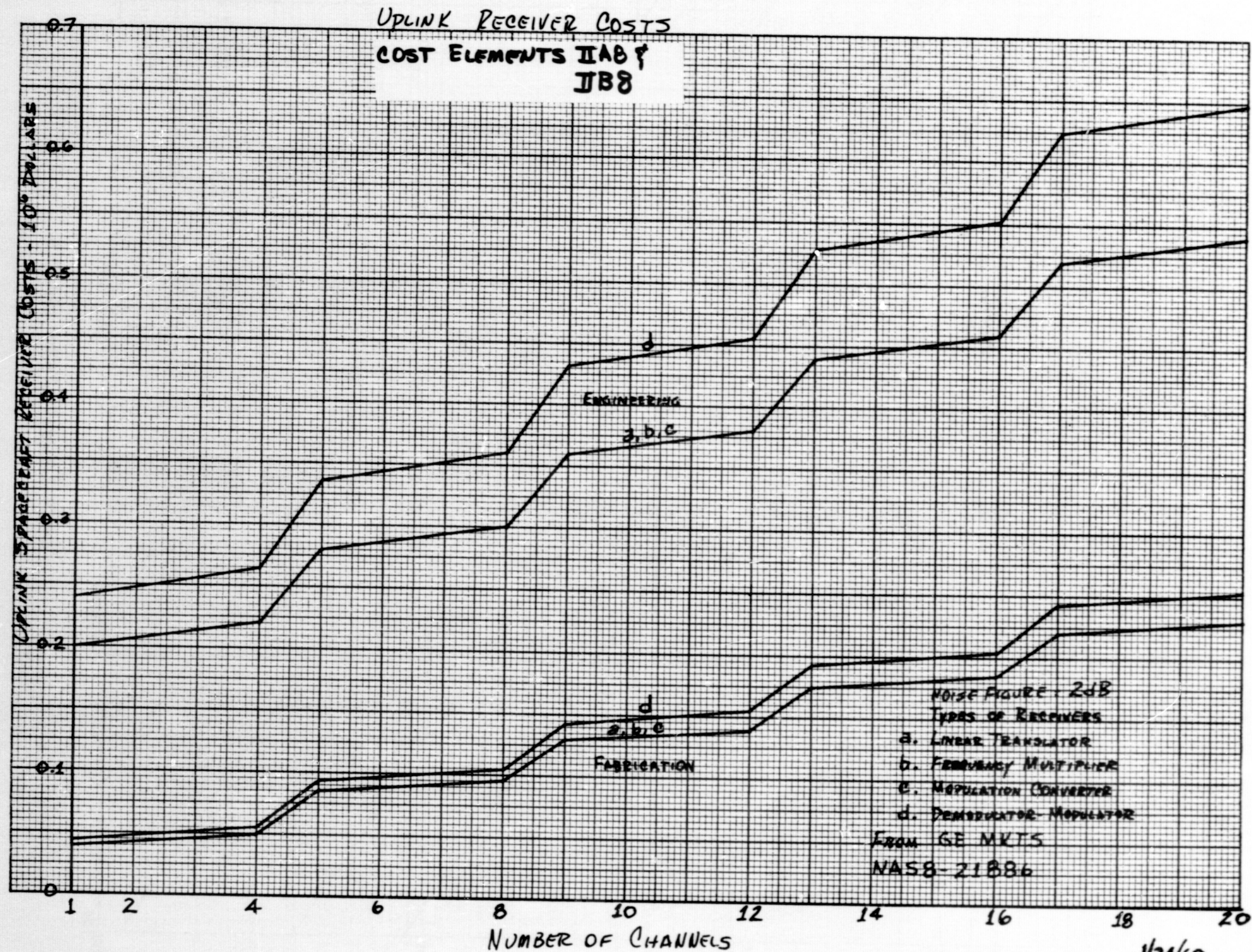


STATIONKEEPING COSTS COST ELEMENTS IIA4 AND IIB4



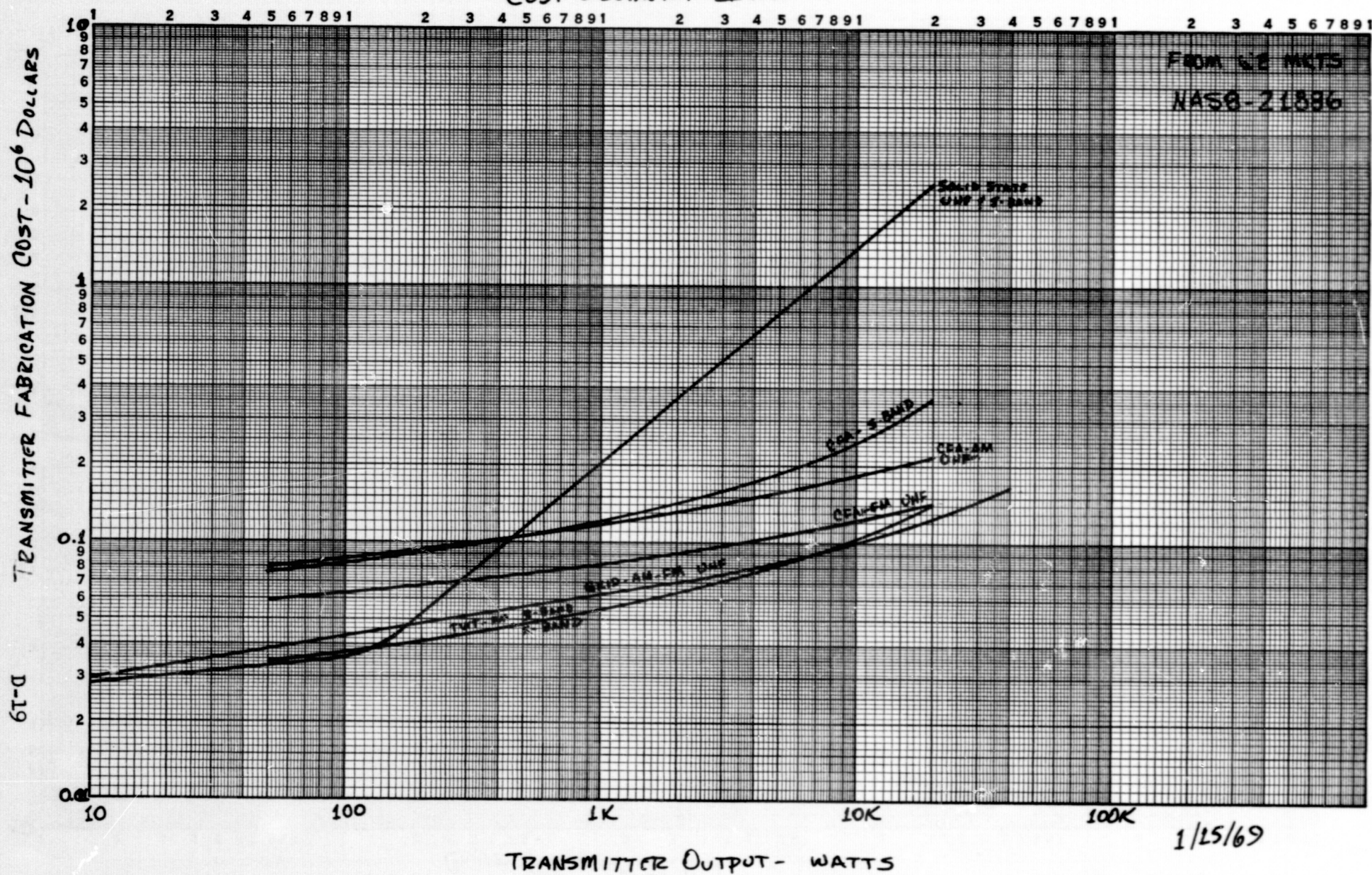
THERMAL CONTROL SUBSYSTEM COSTS COST ELEMENTS IIA5 & IIB5



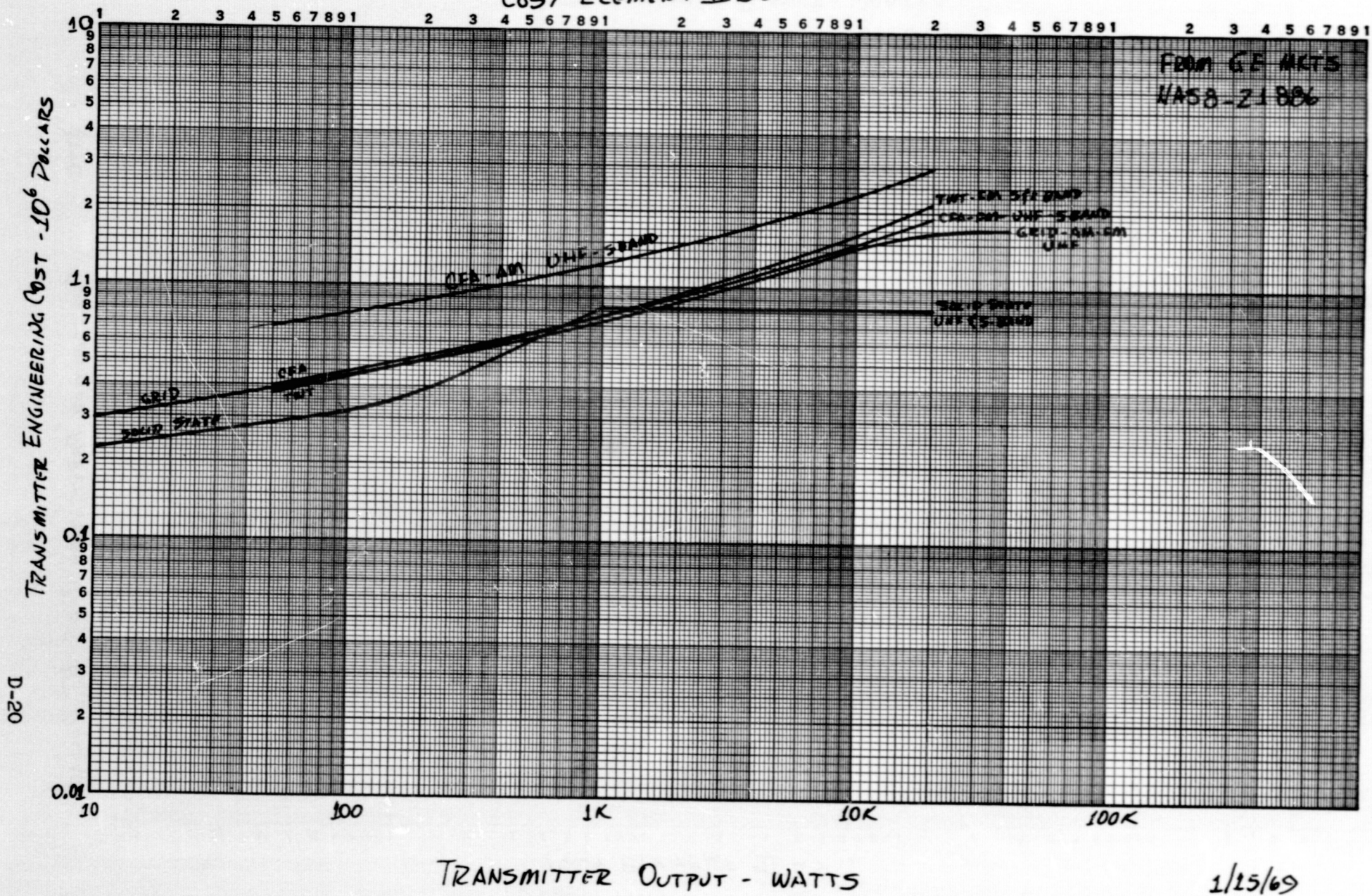


4/24/69

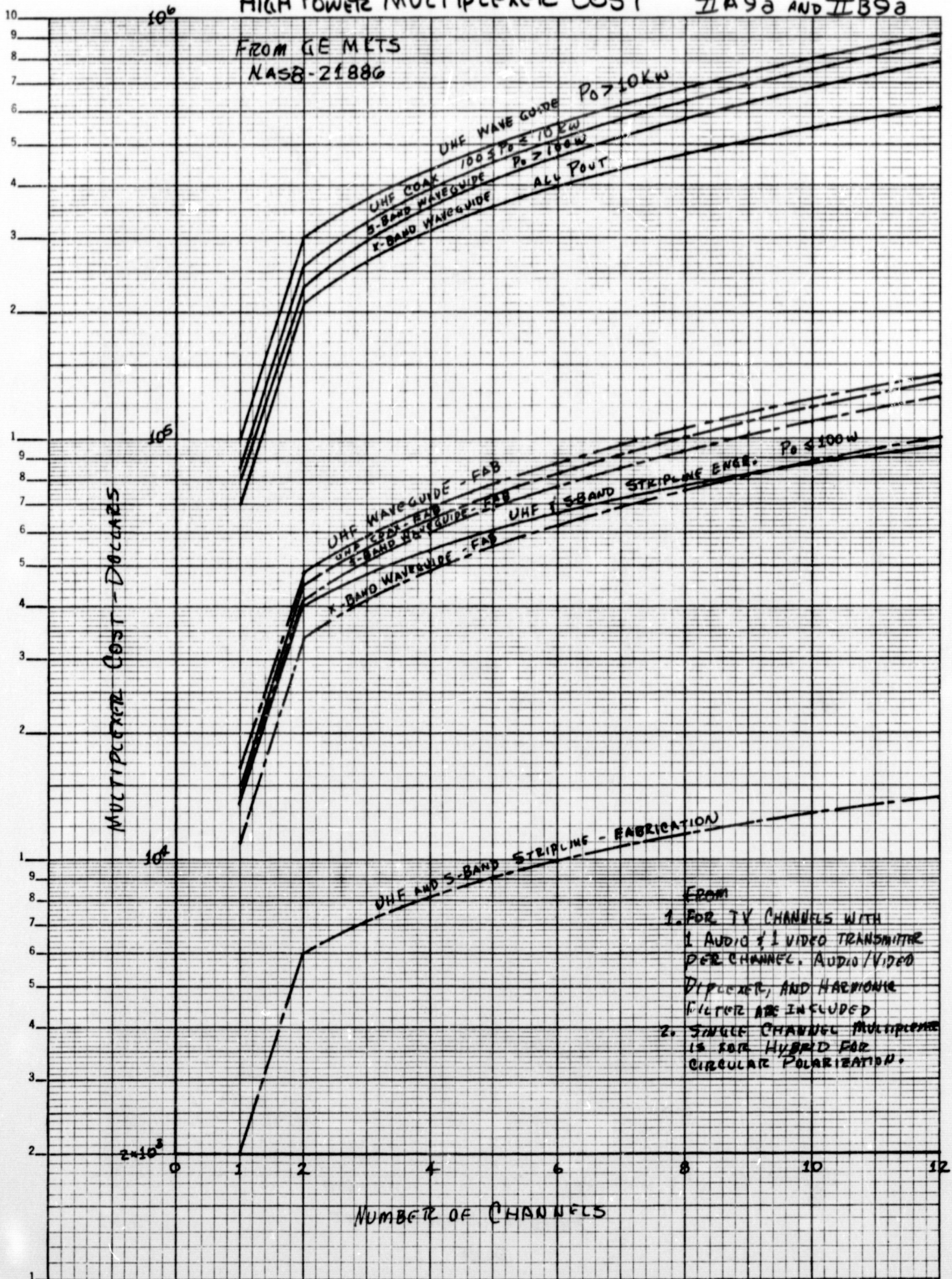
TRANSMITTER FABRICATION COSTS COST ELEMENT II A9



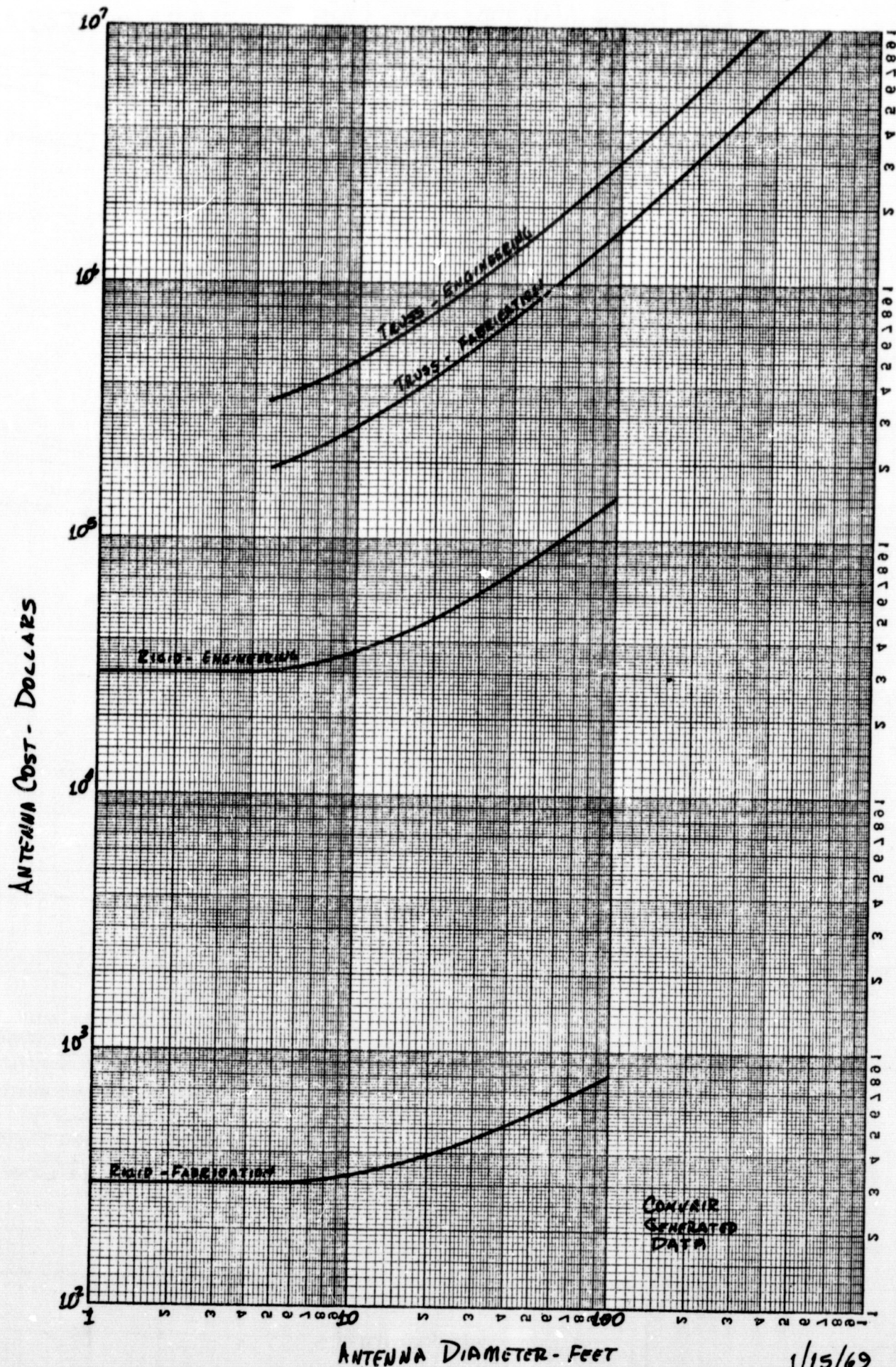
TRANSMITTER ENGINEERING COSTS
COST ELEMENT IIB9

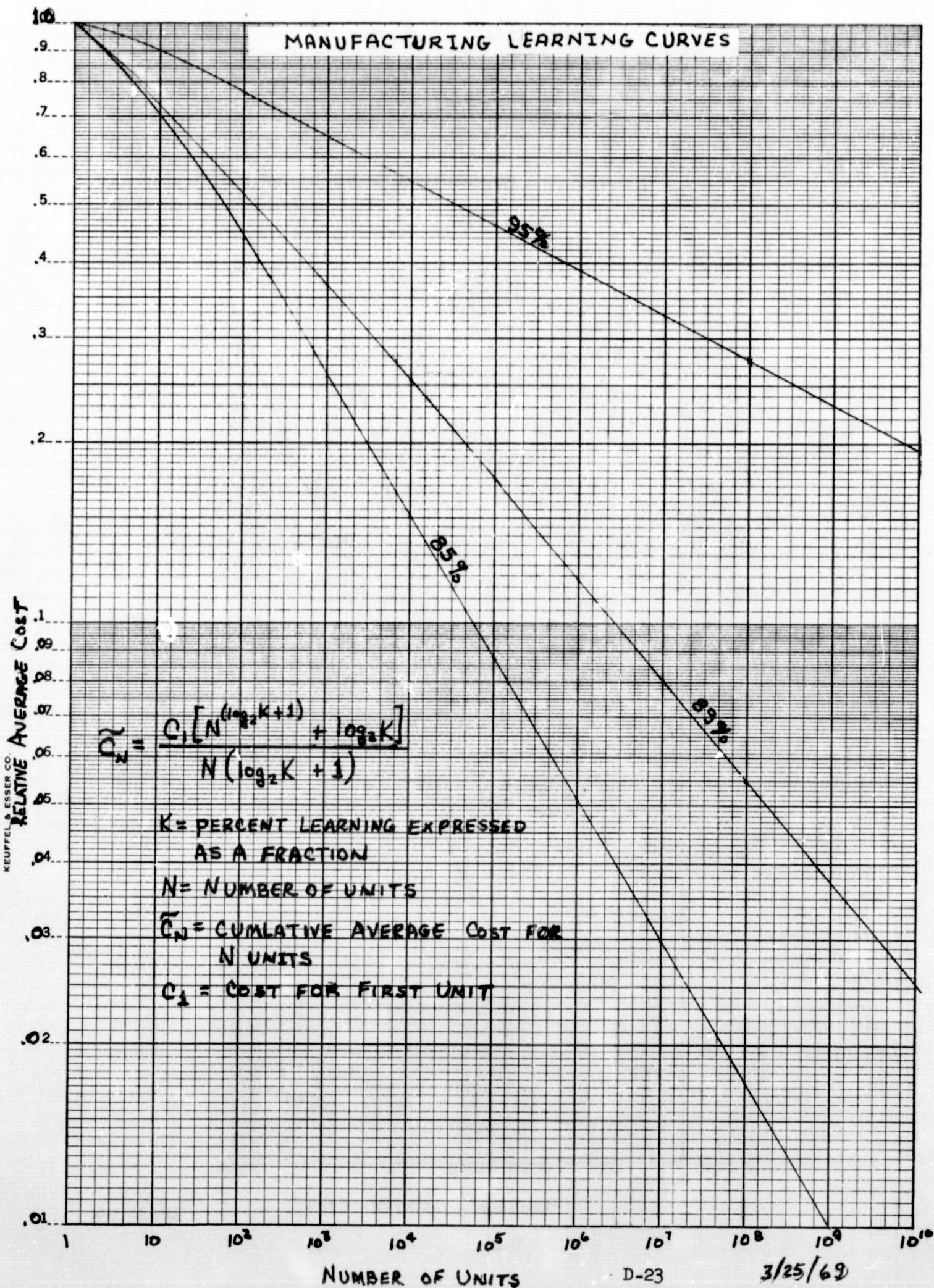


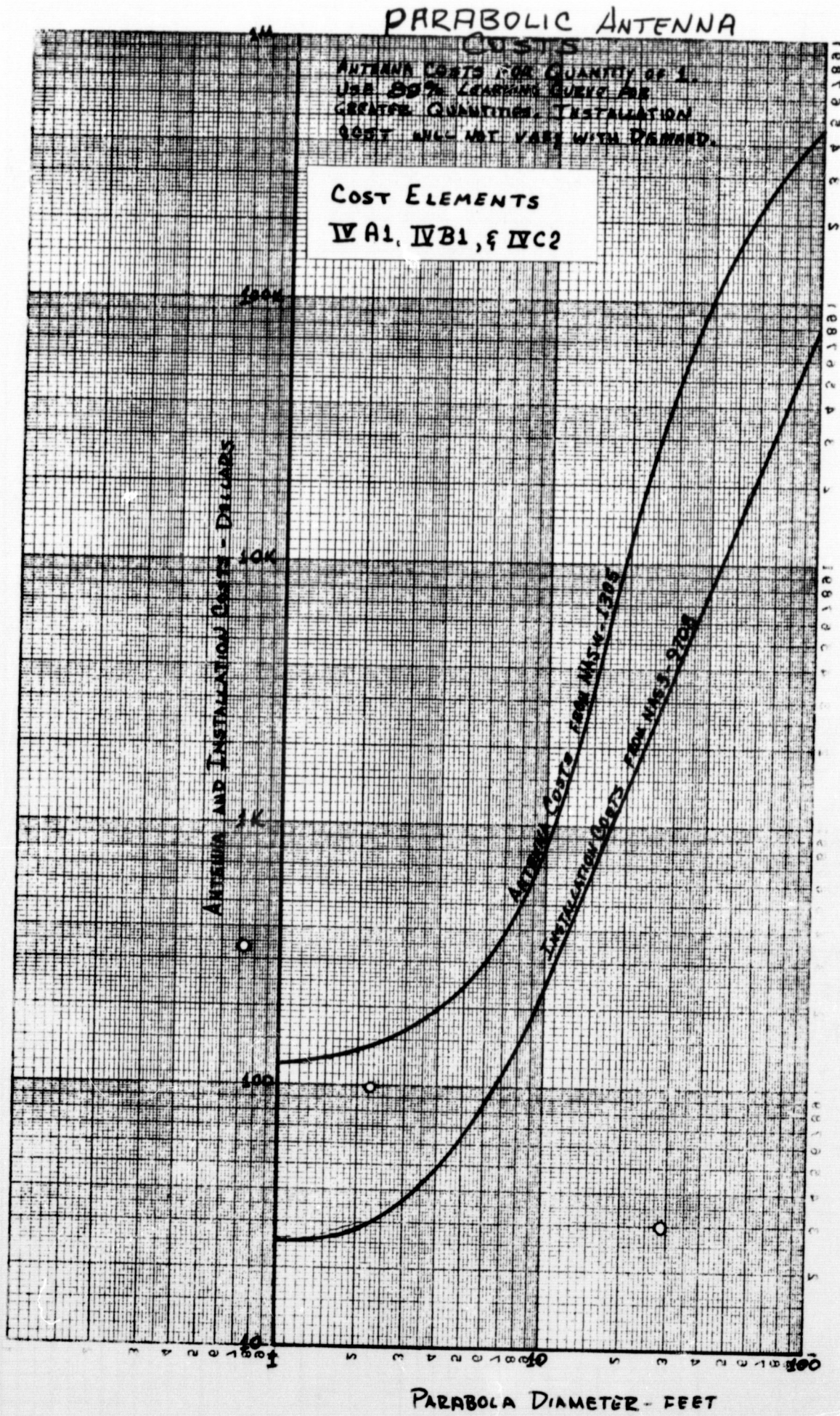
HIGH POWER MULTIPLEXER COST

COST ELEMENTS
IIA9a AND IIB9aFROM GE METS
NASB-21886

SPACECRAFT ANTENNA COSTS COST ELEMENTS IIA8a & IIB8a





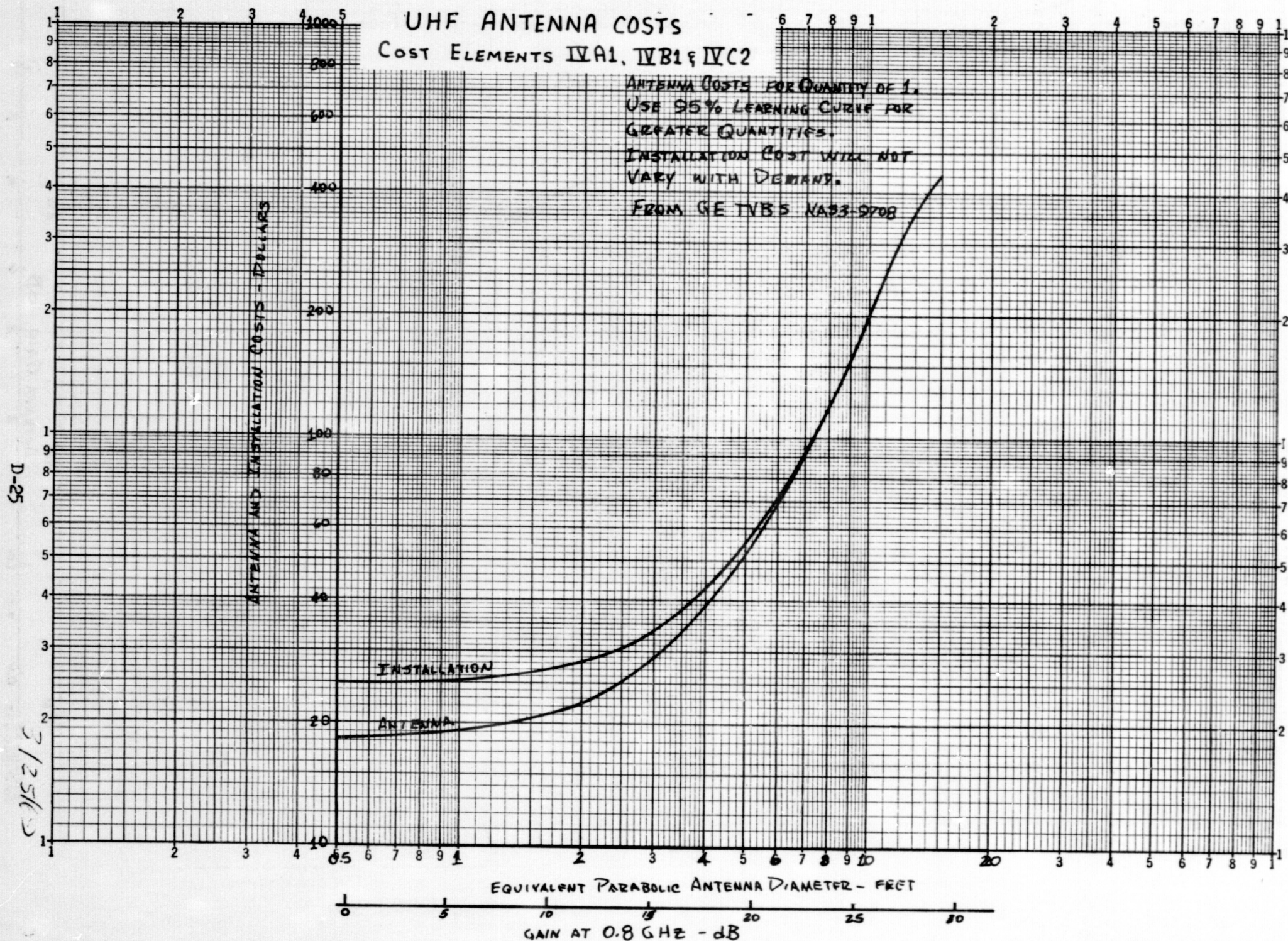


3/25/69

UHF ANTENNA COSTS

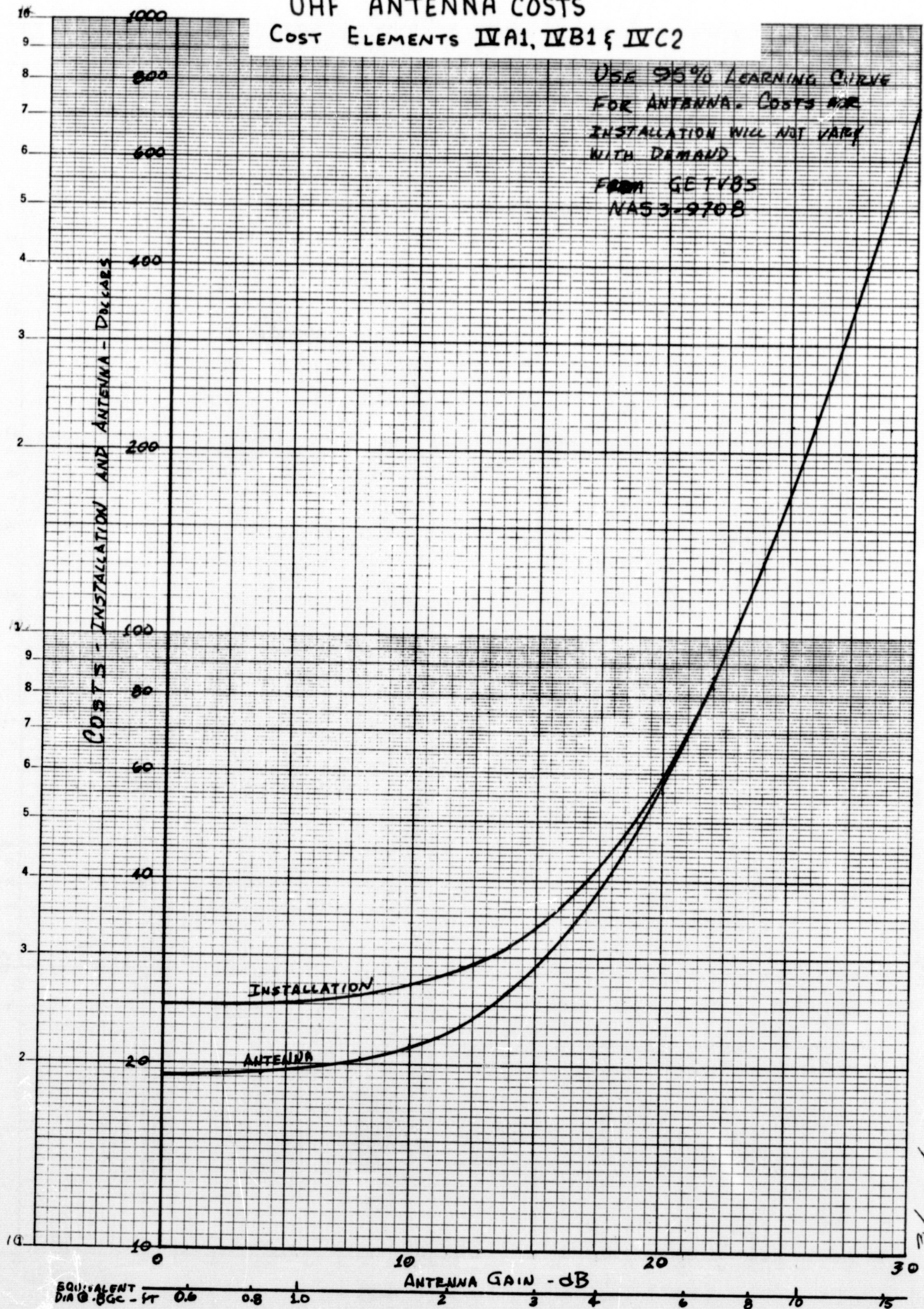
COST ELEMENTS IVA1, IVB1 & IVC2

ANTENNA COSTS FOR QUANTITY OF 1.
USE 95% LEARNING CURVE FOR
GREATER QUANTITIES.
INSTALLATION COST WILL NOT
VARY WITH DEMAND.
FROM GE TVBS NAB3-9708



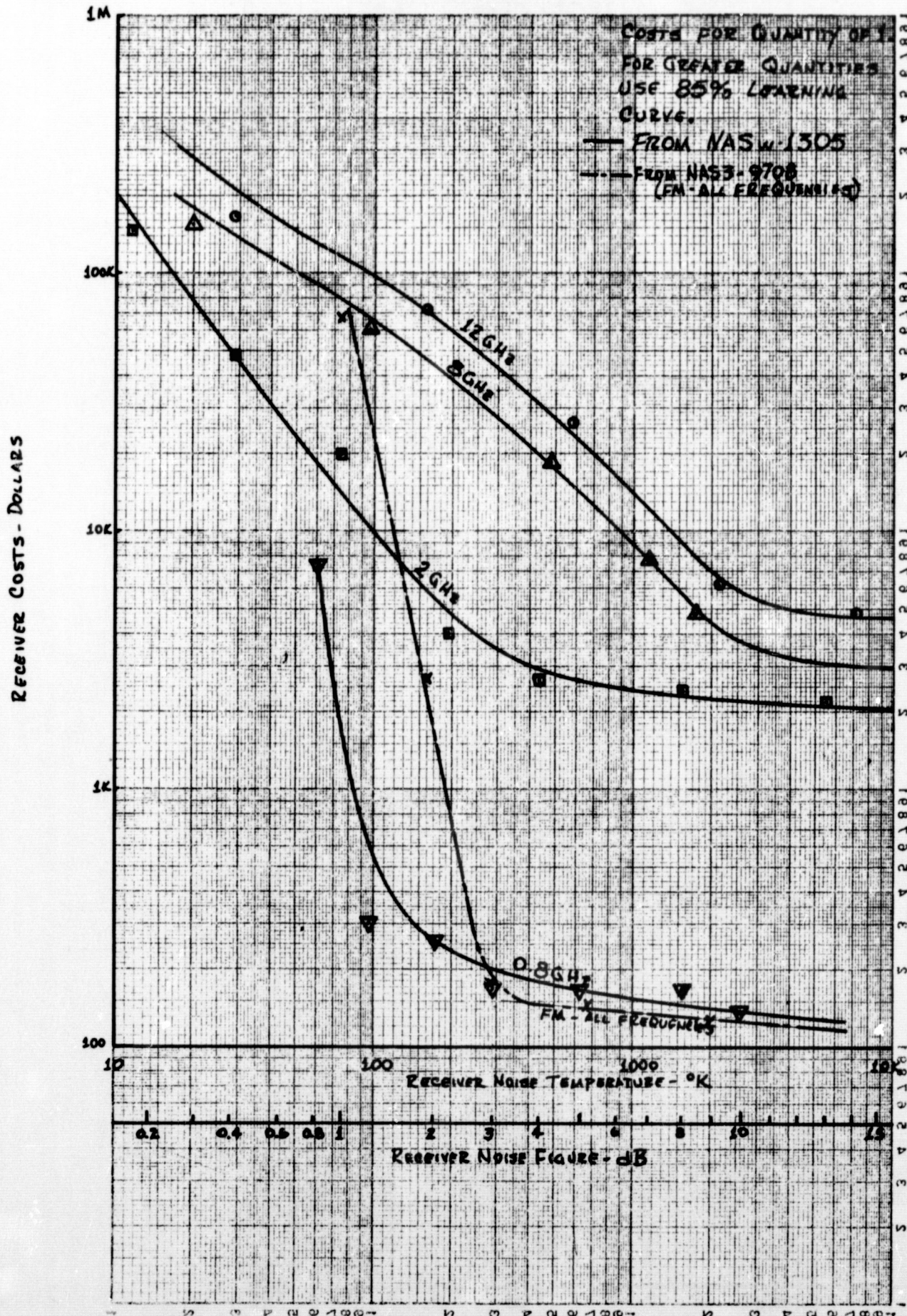
UHF ANTENNA COSTS

COST ELEMENTS IVA1, IVB1 & IVC2



K&E SEMI-LOGARITHMIC 359 61
KEUFFEL & ESSER CO. MADE IN U.S.A.
2 CYCLES X 70 DIVISIONS

GROUND RECEIVER COSTS COST ELEMENTS IIA2, IIB2 & IIC2



HOME RECEIVER TRANSMISSION LINE COST

USE 95% LEARNING CURVE
FOR GREATER LOTS.

FOR 1 RECEIVING SYSTEM.

COST ELEMENT IN A2, IRE2, IRE2

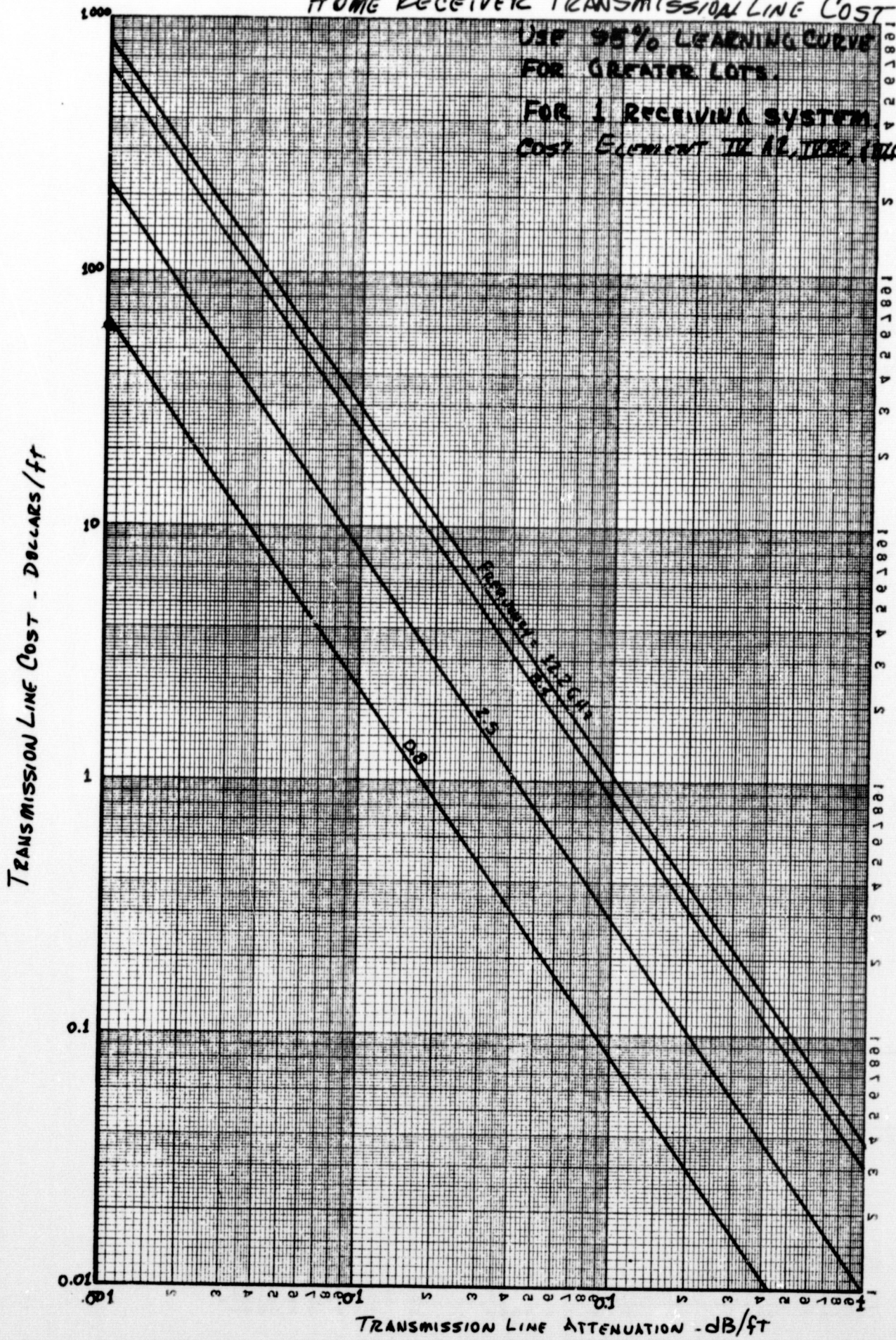


TABLE D.1 THRUPUT COST ELEMENTS

<u>COST ELEMENT</u>	<u>VALUE</u>
I. Ground Uplink System	
A-5 Satellite Operation Unit Recurring	.242
A-7 Test Equipment Unit Recurring	.200
D-7 Satellite Operations and Control Personnel	.147
II. Spacecraft System	
A-7 Life Support Subsystem	0
A-8 Instrumentation and Communication	0
B-7 Life Support Subsystem	0
B-8 Instrumentation and Communication	0
B-19 GSE Hardware	0
B-20 Facilities and S/C Test Equipment	.75
IV. Ground Downlink System - CATV	
A-3 Receiving System Support Facilities	.0542
A-4 Receiving System Equipment (excl. antenna and receiver)	.0015

TABLE D-2. TVBS COST MODEL
LAUNCH VEHICLE COSTS IN 10⁶ DOLLARS

COST ELEMENT	LAUNCH VEHICLE				
	TITAN IIC	TITAN IID/ CENTAUR	SATURN IB/ CENTAUR	SATURN V (UNMANNED)	SATURN V (MANNED)
UNIT RECURRING COST					
68. Launch Vehicle	12.5	14.6	33.5	130.6	166.6
69. Adapters & Fairings	0.4	0.6	0.9	7.2	0.5
NON-RECURRING COST					
71. Launch Vehicle, Adapters & Fairings	2.9	2.9	33.4	29.5	6.0
72. Launch Complex Modifi- cations	0.0	0.0	5.0	1.0	1.0
73. Peculiar GSE Hardware	0.3	0.6	1.5	1.2	0.6
75. Center Support & Development	3.0	3.0	7.0	5.0	6.5
ACQUISITION COST					
78. Center Support per Unit	1.0	1.1	5.4	34.3	41.8
79. Launch Operations, Support & Spares	4.1	6.0	14.6	39.0	48.4

For Concept E, the following costs are incorporated into the launch vehicle costs.

Docking Cone 300 lbs.

- Assume mechanical equipment at \$500/lb.
- Unit Recurring \$150K
- Non-recurring \$150K

Crew Systems 500 lbs

- No Life Support System included - all LSS functions provided by CSM piped in and out.
- Need (1) ducting and circulation for atmosphere
 - (2) miscellaneous furnishings, lights, etc.
 - (3) miscellaneous controls and displays, housekeeping, etc.
- Probably need emergency spacesuit/backpack in module but cost included herein.

Ducting and fans	25K
Misc. furnishings	10K
Controls and displays	<u>50K</u>
Total unit recurring	85K
Non-recurring	255K

Emergency Atmosphere

- Atmosphere pressure bottles, pressure regulator, and miscellaneous - assume two

Unit recurring - $2 \times 10K + 10K + 5K = 35K$

Non-recurring - 100K

Therefore, total unit recurring = $150K + 85K + 35K = 270K$

total non-recurring = $150K + 255K + 100K = 505K$

A. EQUIPMENT MODULE STRUCTURE

$$\text{Weight} = \pi l_{EM} D_{EM}^2 \quad l_{EM} = \text{Length of Equipment Mod.}$$

$$\text{Volume} = \text{Negligible (input = 0)} \quad D_{EM} = \text{Diameter of Equipment Mod.}$$

B. ELECTRICAL POWER SUBSYSTEM

Solar Array

 P_{PRIME} = Prime Power in Watts

$$\text{Weight} = 0.1 P_{PRIME}$$

$$\text{Volume} = 0.0167 P_{PRIME}$$

Nuclear Reactor

$$\text{Weight} = 0.305 P_{PRIME}$$

$$\text{Volume} = 0.001 P_{PRIME}$$

Batteries

$$\text{Weight} = \text{Curve}$$

$$\text{Volume} = \text{Curve}$$

Power Conditioners

$$\text{Efficiency} = \text{Curve}$$

$$\text{Weight} = \text{Curve}$$

$$\text{Volume} = \text{Curve}$$

C. ATTITUDE CONTROL SUBSYSTEM

$$\text{Weight} = 60 + W_{AC} \quad \text{Pounds}$$

$$\text{Volume} = 1 + V_{AC} \quad \text{ft}^3$$

$$\text{Power} = 30 + P_{AC} \quad \text{Watts}$$

Parameters:

N = Satellite life in years

A = Area of solar array — ft^2

CONCEPT	$M =$	$L =$
A, B	$\frac{1}{4} \frac{A}{\ell_{EM}} + \frac{1}{2} D_{EM}$	D_{EM}
C	$\frac{1}{2} D_A + 5$	D_{EM}
D	30 ft	15 ft
E	$\frac{1}{2} (D_A + \ell_f + \ell_A)$	D_A
F	$\frac{1}{2} D_{TOTAL} + 11$	D_{TOTAL}
G	$\frac{A}{48} + 5$	10

ℓ_{EM} = Length of equipment module

D_{EM} = Diameter of equipment module

ℓ_F = Feed length

ℓ_A = Stowed antenna height

D_A = Antenna diameter

1. NH_3 Resistojet

$$W_{AC} = (0.706 + 1.36N) \frac{AM}{L} \times 10^{-3}$$

$$V_{AC} = 2 \times 10^{-5} \cdot \frac{AMN}{L}$$

$$P_{AC} = 3.7 \times 10^{-3} \frac{AM}{L}$$

2. Ion Propulsion

$$W_{AC} = 16.9 (3.12 \times 10^{-3} \frac{AM}{L} + 1.0)^{\frac{1}{2}} + 5(40 + 7.7 \times 10^{-3} \frac{AM}{L})^{\frac{1}{2}} + 2.2 \times 10^{-4} \frac{AMN}{L}$$

$$V_{AC} = (0.025 + 0.37 \times 10^{-3} N) \frac{AM}{L} \times 10^{-3}$$

$$P_{AC} = 160 + 0.031 \frac{AM}{L}$$

3. Water Electrolysis

$$W_{AC} = 1.37 \times 10^{-3} \frac{AMN}{L}$$

$$V_{AC} = 1.32 \times 10^{-5} \frac{AMN}{L}$$

$$P_{AC} = 3.7 \times 10^{-3} \frac{AM}{L}$$

D. STATIONKEEPING

W_S = Weight of spacecraft less stationkeeping

N = Spacecraft life in years

1. Nitrogen Cold Gas (East-West Only)

$$W_{CS} = \frac{0.01782 N W_S}{1 - 0.0133 N}$$

$$V_{CS} = 0.038 W_{CS}$$

$$P_{CS} = 0.$$

2. Water Electrolysis

$$W_{CS} = \frac{0.00216 N W_S}{1 - 0.00138 N} + \frac{0.0211 N W_S}{1 - 0.0135 N}$$

$$V_{CS} = 0.03 W_{CS}$$

$$P_{CS} = 5 \times 10^{-3} W_S$$

3. Ammonia Resistojet

$$W_{CS} = \frac{0.00354 N W_S}{1 - 0.00197 N} + \frac{0.0346 N W_S}{1 - 0.0192 N}$$

$$V_{CS} = 2.12 W_S$$

$$P_{CS} = 4 \times 10^{-3} W_S$$

4. Peroxide

$$W_{CS} = \frac{0.0381 N W_S}{1 - 0.0218 N} + \frac{0.0039 N W_S}{1 - 0.00223 N} + 40$$

$$V_{CS} = 0.026 W_S$$

$$P_{CS} = 0.$$

5. Ion Propulsion

$$W_{CS} = 1.66 + 0.38 (W_S)^{\frac{1}{2}} + 1.26 P^{\frac{1}{2}} + 0.0013 N W_S$$

$$V_{CS} = (0.06 + 4.77 \times 10^{-3} N) \times W_S \times 10^{-3}$$

$$P_{CS} = (20 + 0.0832 W_S)$$

E. THERMAL CONTROL SUBSYSTEM

1. Passive Radiator For Total Spacecraft Heat Rejection

P_{DIS} = Power to be dissipated by thermal control subsystem

P_{PRIME} = Prime power generated by power source

P_{XMTR} = Power transmitted by transmitter (RF)

V_{EM} = Volume of Equipment Module

$P_{DIS} = P_{PRIME} - P_{XMTR}$ Watts

$A_{TC} = 0.0548 P_{DIS}^{2/3} - 2.5 V_{EM}^{2/3}$ ft²

$$W_{TC} = 0.0548 P_{DIS} - 2.5 V_{EM}^{2/3} \quad \text{lbs}$$

$$V_{TC} = 0.0913 P_{DIS} - 0.417 V_{EM}^{2/3} \quad \text{ft}^3$$

2. Heat pipe for transmitter; passive radiator for other subsystems.

P_{DIST} = Transmitter thermal power to be rejected

P_{DISS} = Heat to be rejected from other subsystems

η = Transmitter efficiency

a. Heat Pipe for Transmitter

$$P_{DIST} = \frac{1 - \eta}{\eta} P_{XMTR} \quad \text{Watts}$$

$$A_{HP} = 9 \times 10^{-3} P_{DIST} \quad \text{ft}^2$$

$$W_{TC} = 0.01125 P_{DIST} \quad \text{lbs}$$

$$V_{TC} = 0.00225 P_{DIST} \quad \text{ft}^3$$

b. Passive System For Other Subsystems

$$P_{DISS} = P_{PRIME} - \frac{P_{XMTR}}{\eta} \quad \text{Watts}$$

$$A_{PR} = 0.0548 P_{DISS} \quad \text{ft}^2$$

$$W_{PR} = 0.0548 P_{DISS} \quad \text{lbs}$$

$$V_{PR} = 9.13 \times 10^{-3} P_{DISS} \quad \text{ft}^3$$

For total system

$$P_{DIS} = P_{PRIME} - P_{XMTR}$$

$$A_{TC} = A_{HP} + A_{PR} - 2.5 V_{EM}^{2/3}$$

$$W_{TC} = W_{HP} + W_{PR} - 2.5 V_{EM}^{2/3}$$

$$V_{TC} = V_{HP} + V_{PR} - 0.417 V_{EM}^{2/3}$$

3. In addition to Types 1 and 2, if the power source is a nuclear reaction, the following additional values are required

l_{EM} = Equipment Module Length

AREA = $0.02 P_{PRIME}$ ft^2

WEIGHT = $47.0 l_{EM}$ lbs

VOLUME = 0 (Integral with EM Structure)

NOTE: For $D_{EM} = 10$ ft.

$$l_{EM} = \frac{0.05 P_{PRIME} - 78.6}{25}$$

F. UPLINK RECEIVER SUBSYSTEM

Weight = Curve

Volume = TBD

Power = Curve

G. DOWNLINK TRANSMITTER

Weight = Curve

Volume = Curve

Efficiency = Curve

MULTIPLEXER

Weight = Curve

Volume = TBD

Insertion Loss = Curve

H. ANTENNA SUBSYSTEM

1. Reflector

a. Expandable Truss

$$5 \leq D_A \leq 140 \text{ ft}$$

$$\text{Weight} = 0.10 D_A^2$$

$$\text{Volume} = 0.0028 D_A^3$$

b. Rigid Parabola

$$1 \leq D_A \leq 10 \text{ ft}$$

$$\text{Weight} = 6.4 D_A^{0.5}$$

$$\text{Volume} = 0.30 D_A^3$$

2. Feed Booms

A. Rigid Boom

D_f = Diameter of Feed

$$\text{Weight} = 0.32 D_A^{-0.69}$$

$$\text{Volume} = 0.41 D_A D_f$$

b. Folding Boom

$$\text{Weight} = 0.32 D_A^{1.69}$$

$$\text{Volume} = 0.039 D_A^2$$

C. Expandable Boom

$$\text{Weight} = 0.33 D_A^{1.32}$$

$$\text{Volume} = 0.016 D_A D_f^2$$

Now $\eta_{PT} = 95\%$

$$\eta_{PC} = 70\%$$

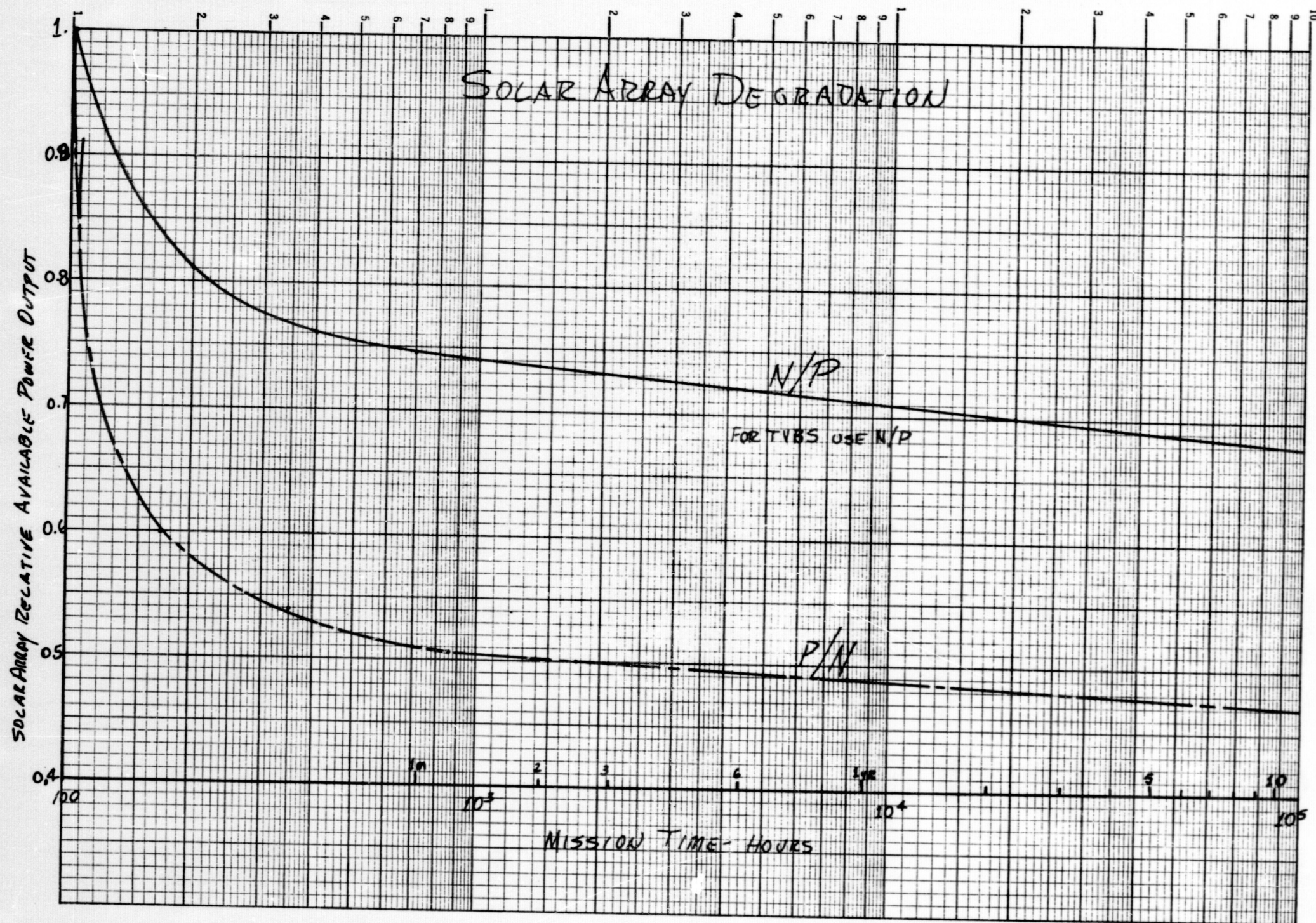
$$\eta_B = 83\%$$

$$\eta_{PCU} = 98\%$$

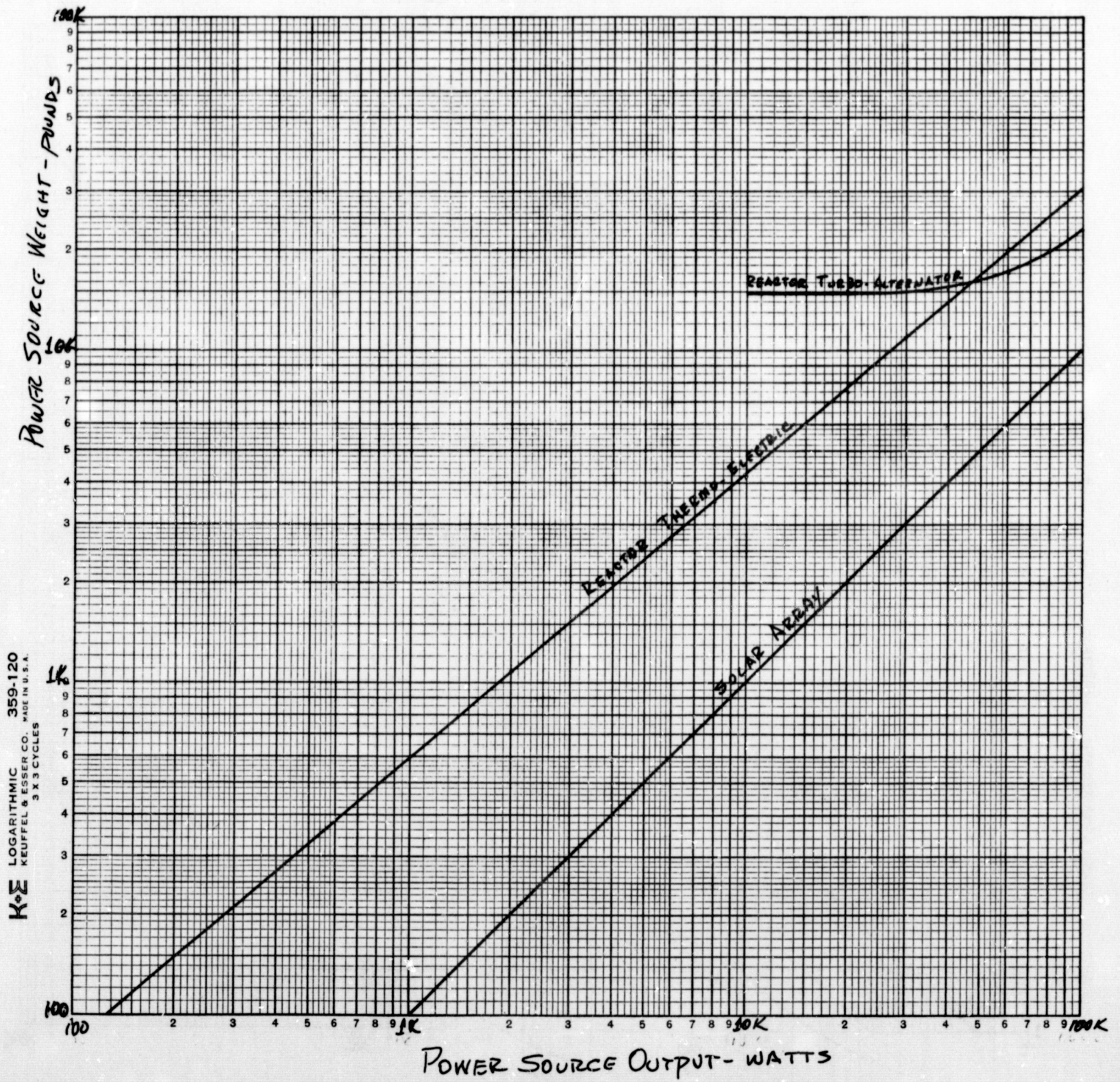
$$\eta_{LCF} = 95\%$$

$$\eta_{TV} \text{ and } \eta_{TA} = \text{funct}(P_T)$$

D-40

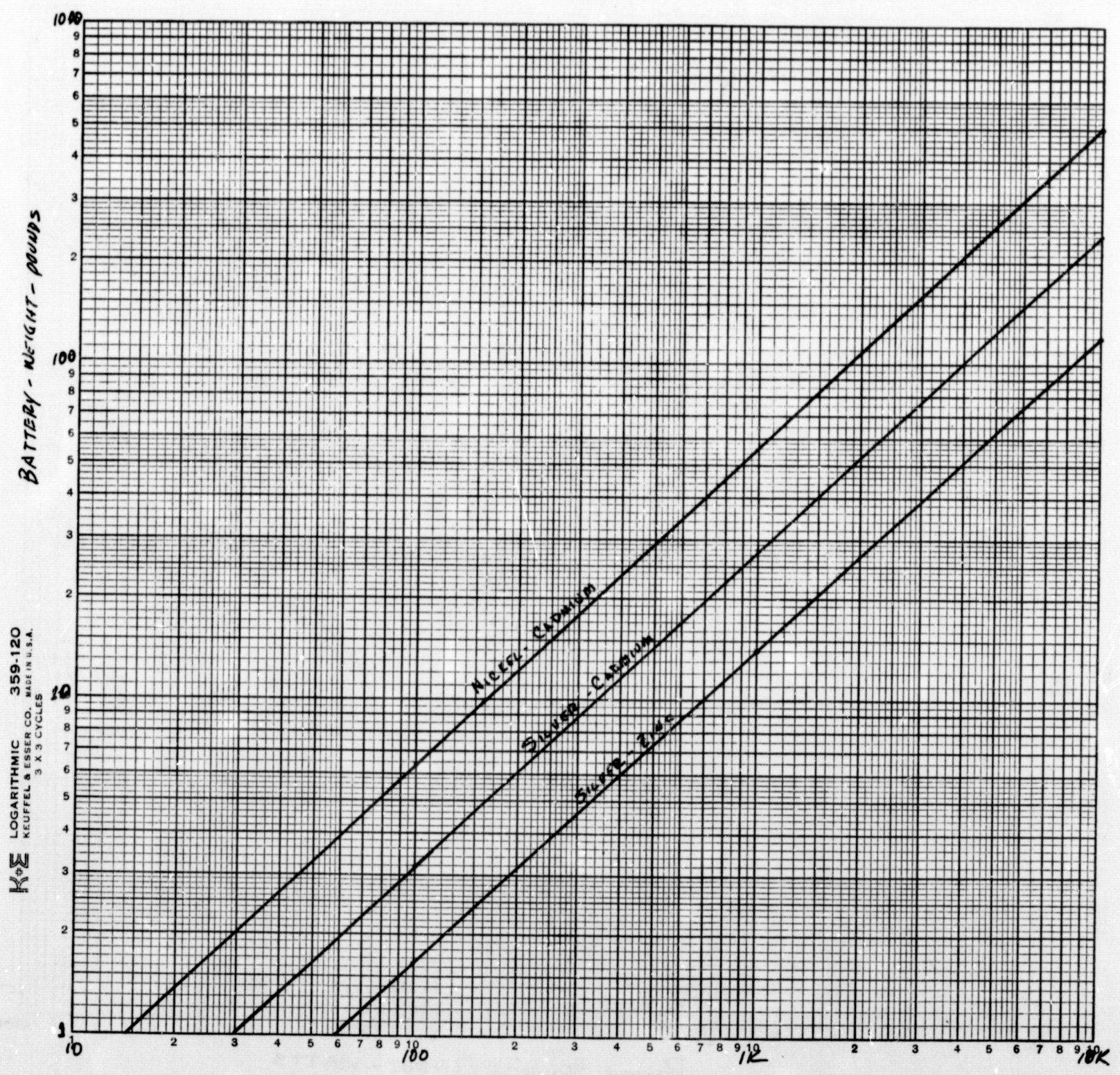


POWER SOURCE WEIGHT



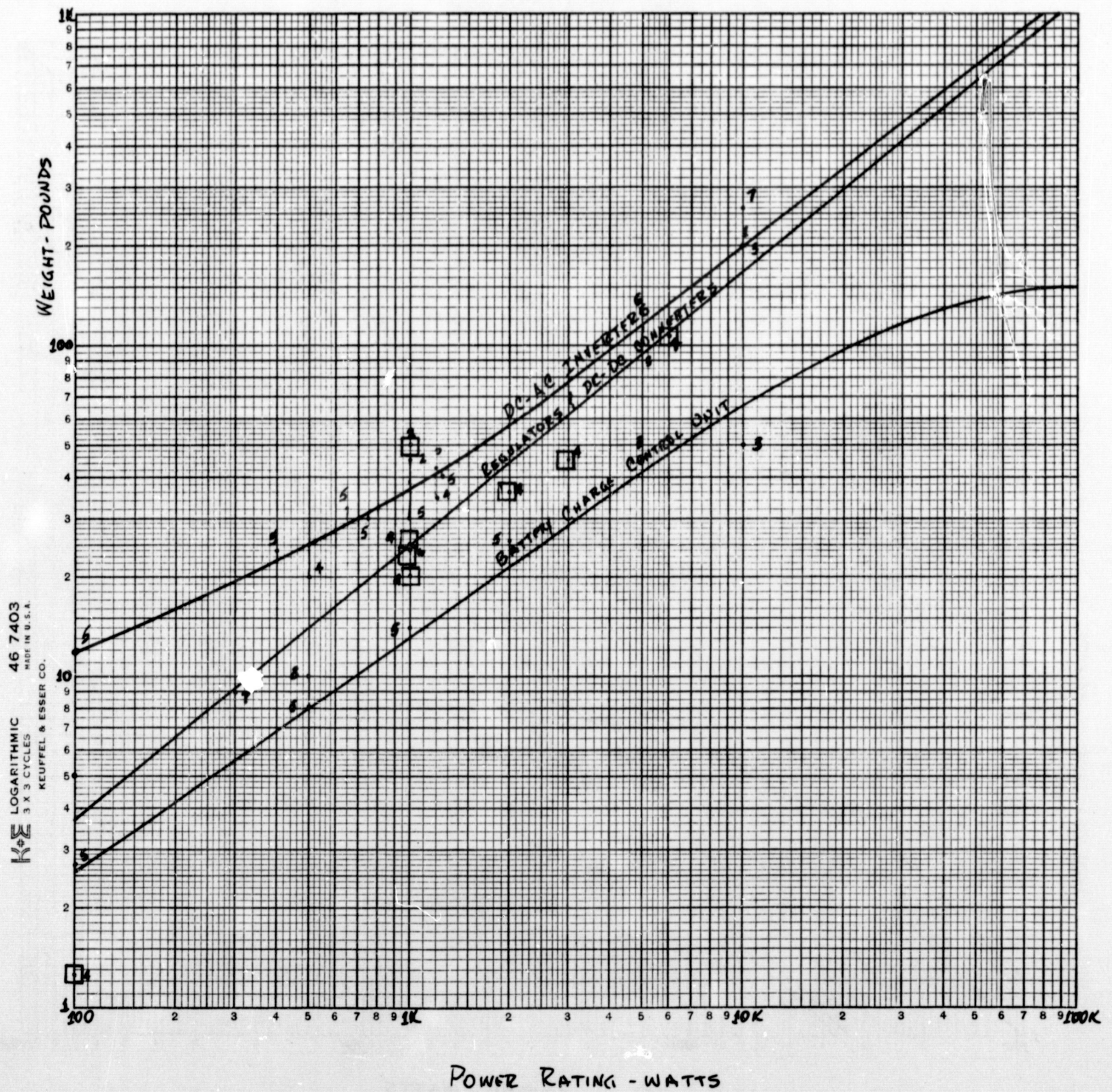
3/31/69

Battery Weight

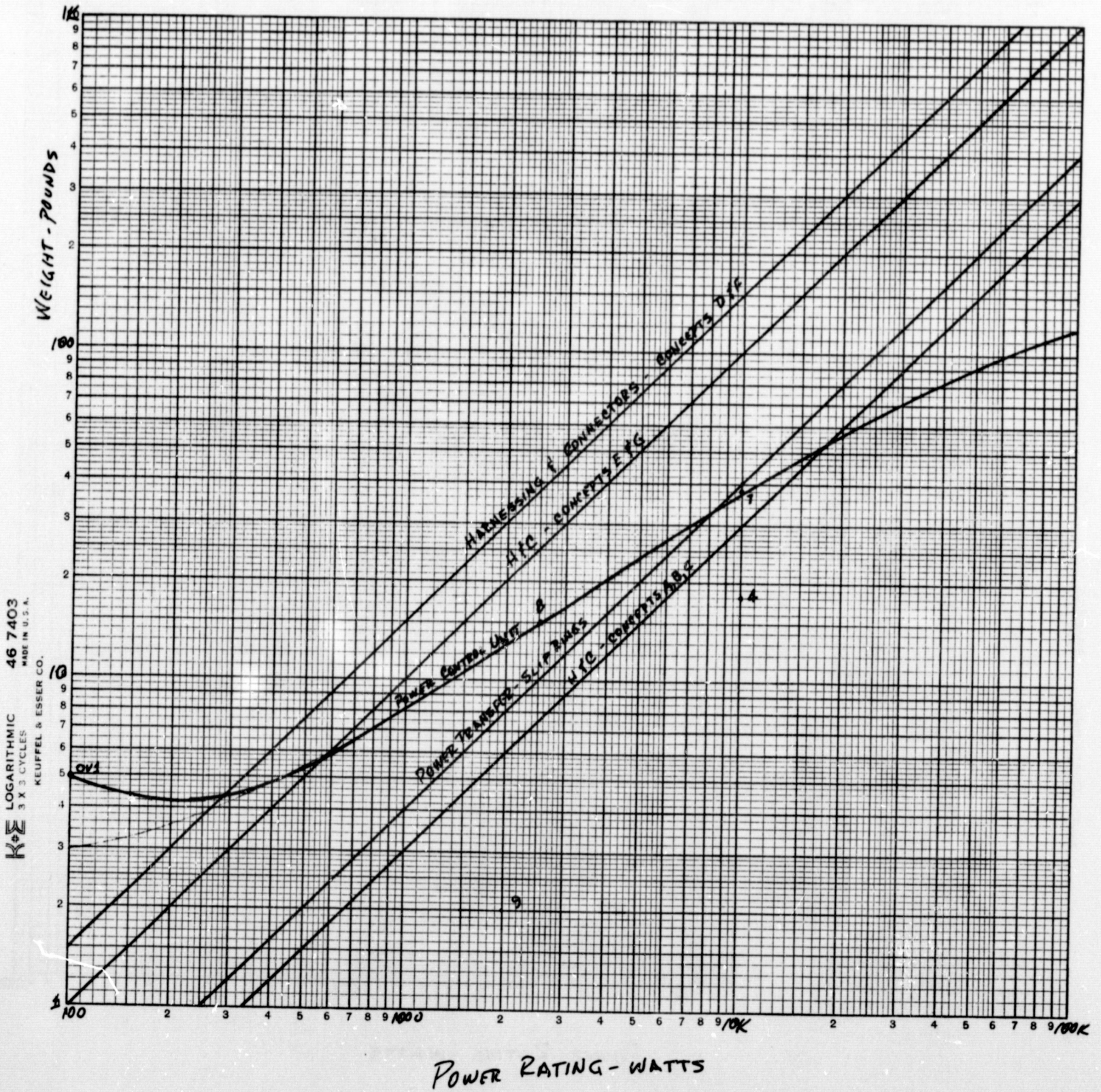


BATTERY CAPACITY - WATT-HOURS

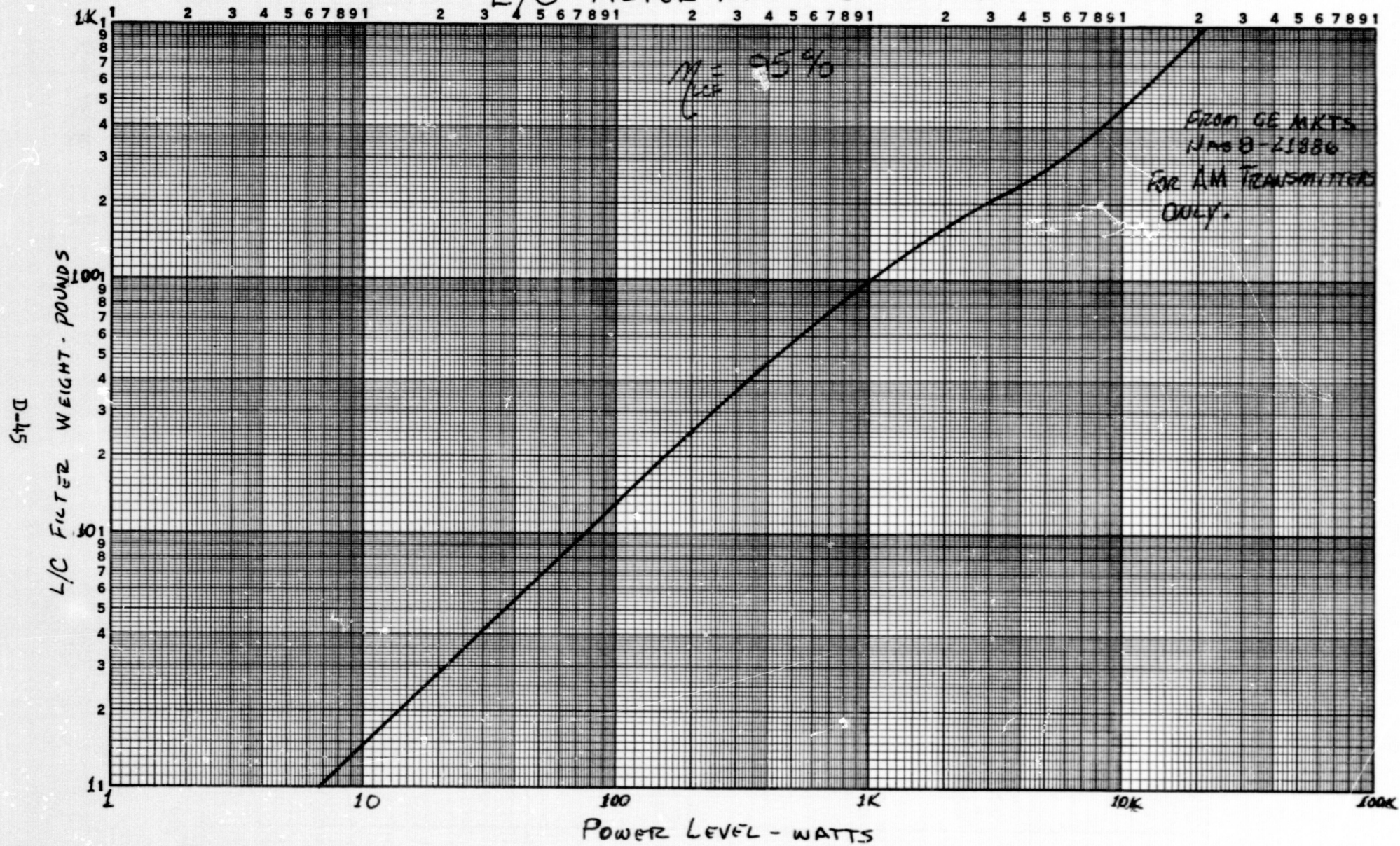
POWER CONDITIONING WEIGHT



POWER DISTRIBUTION WEIGHT

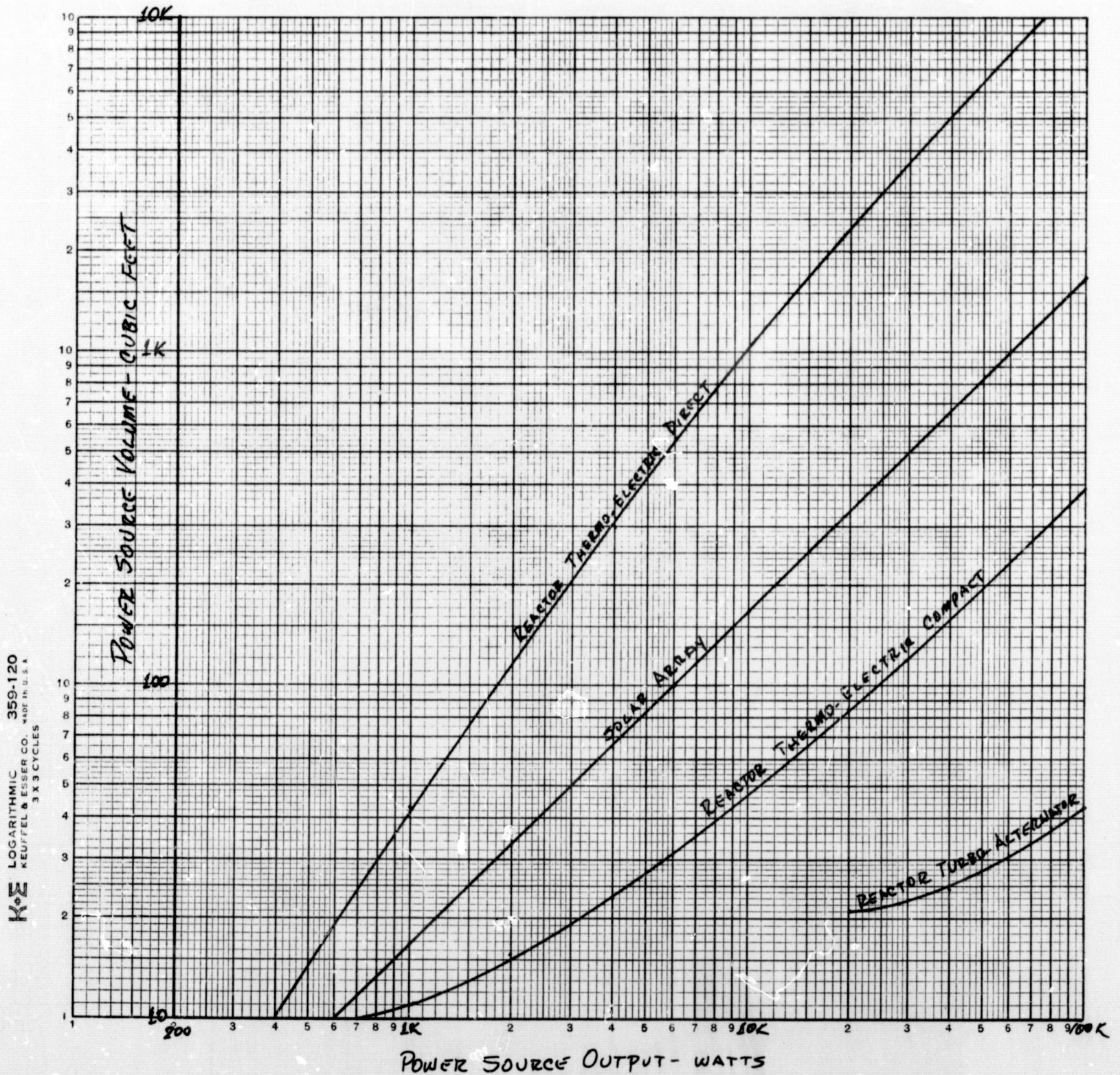


L/C FILTER WEIGHT CHARACTERISTICS



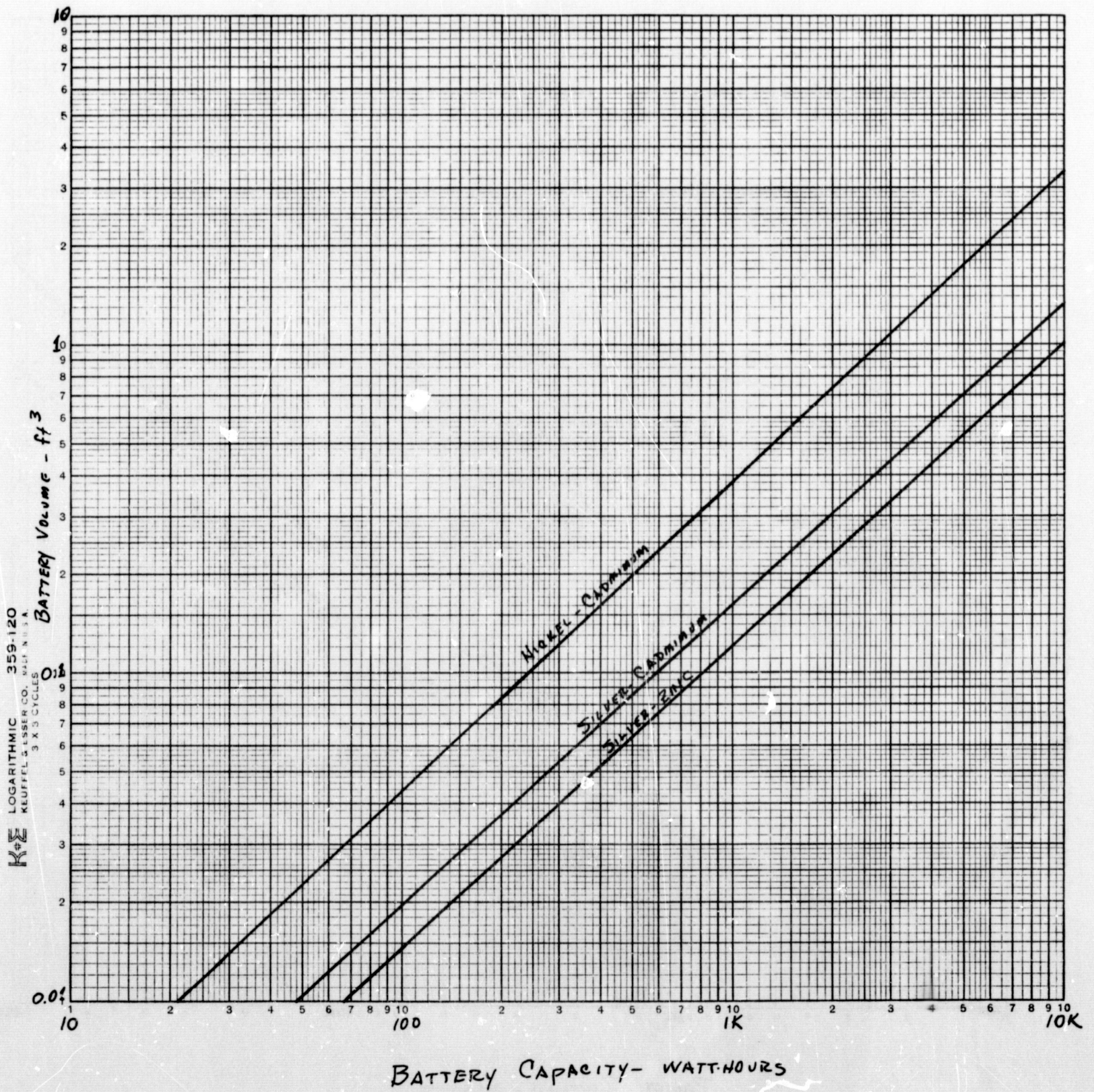
4/7/69

POWER SOURCE VOLUME

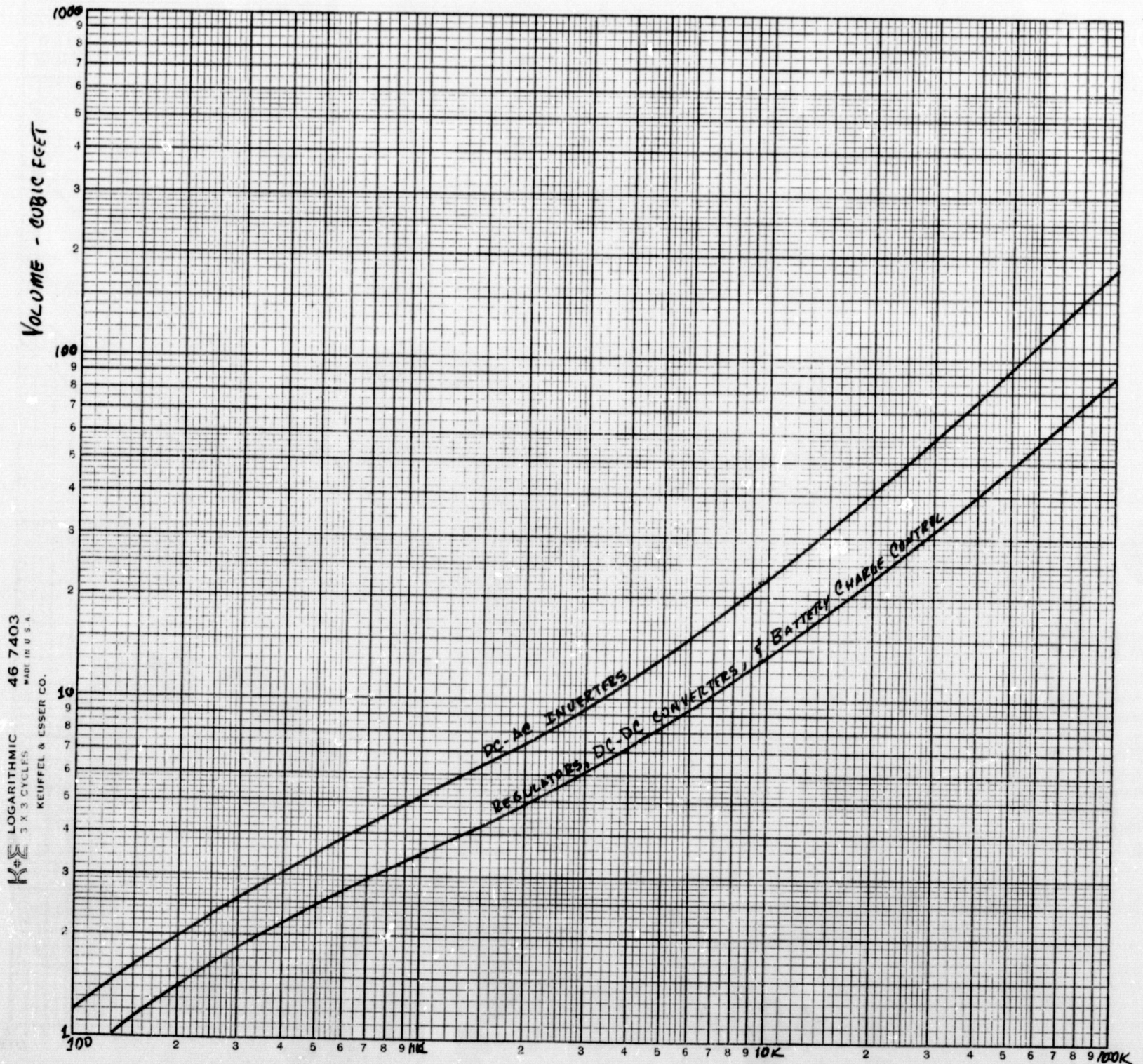


3/31/69

BATTERY VOLUME

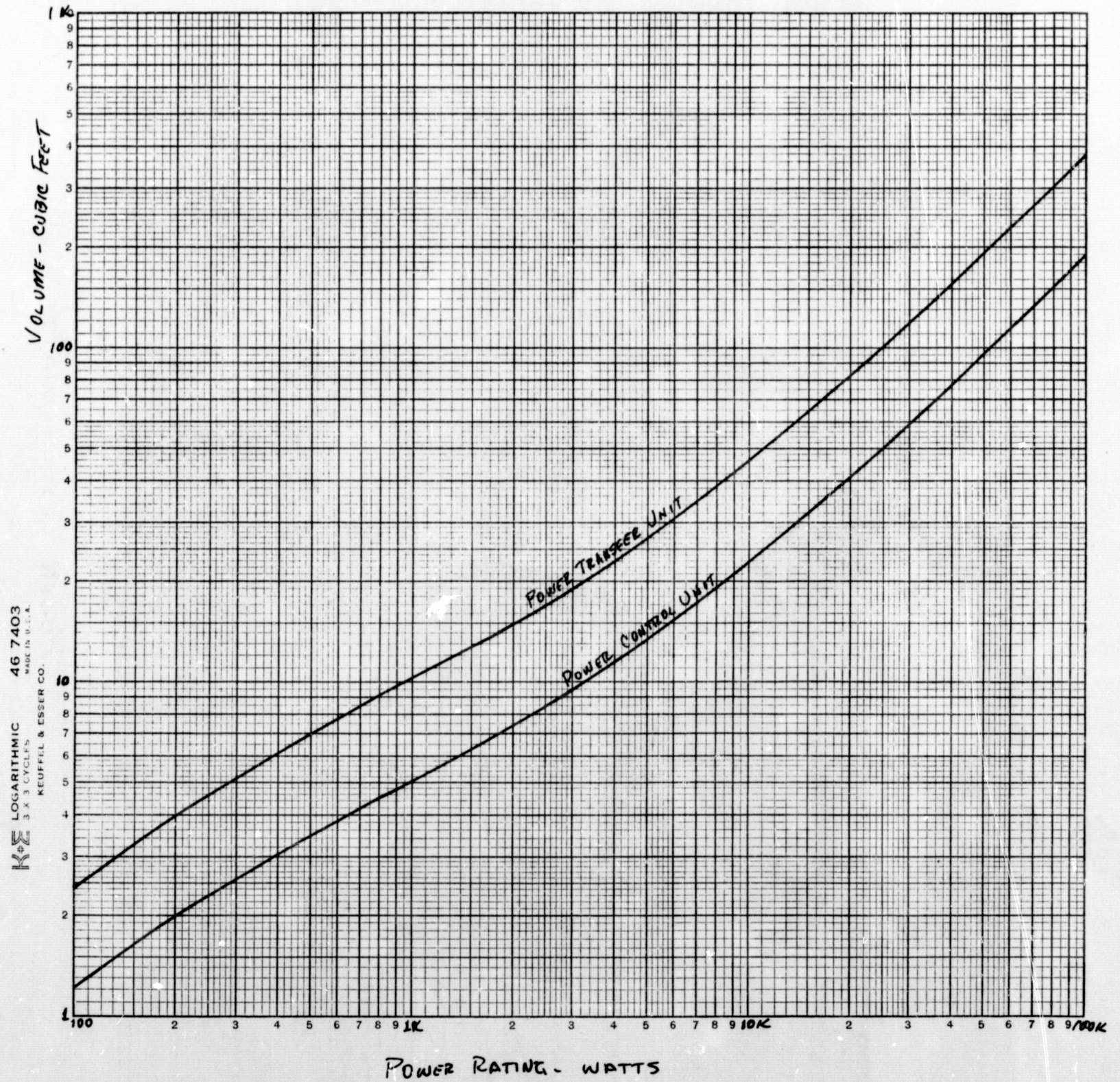


POWER CONDITIONING VOLUME

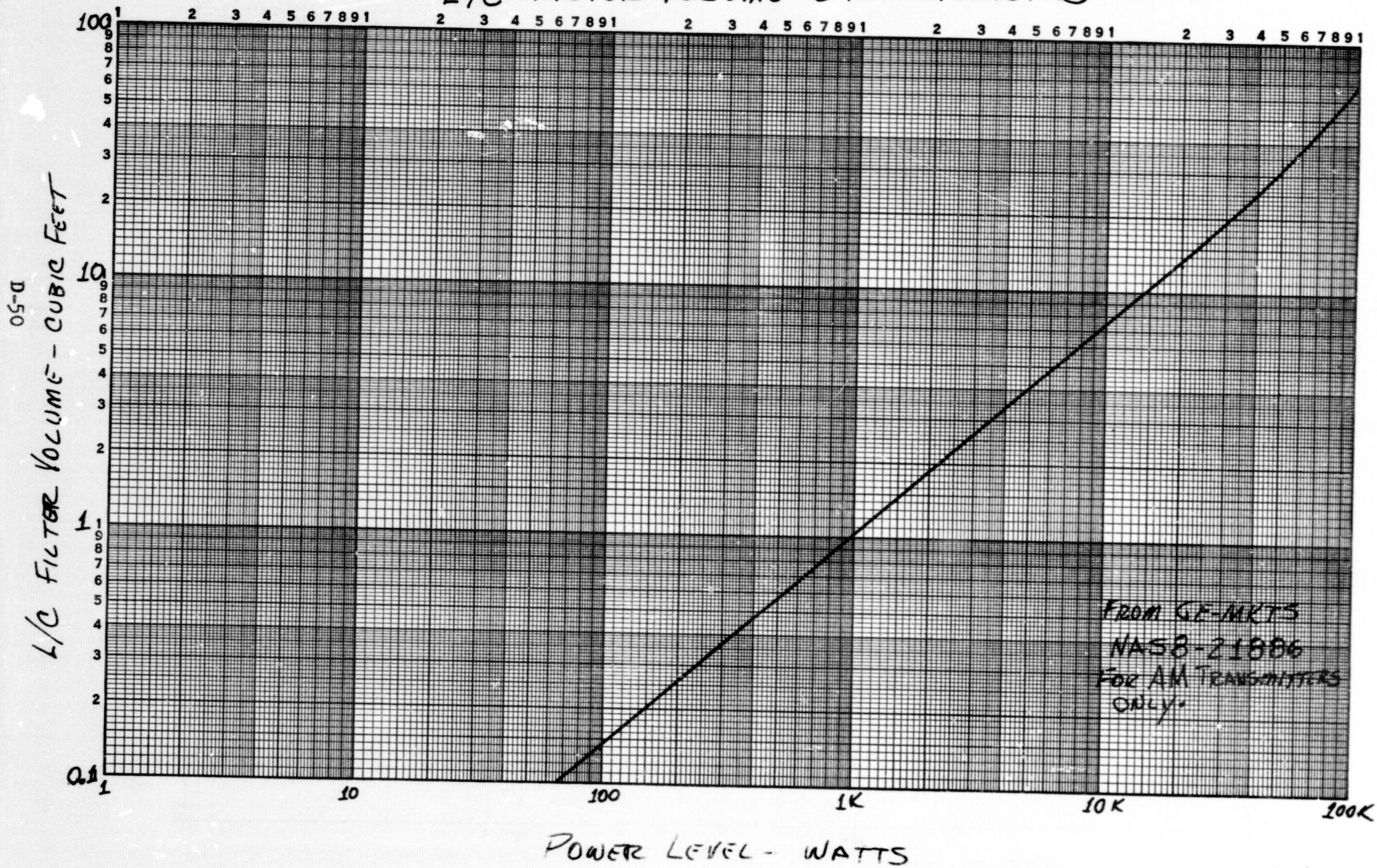


POWER RATING - WATTS

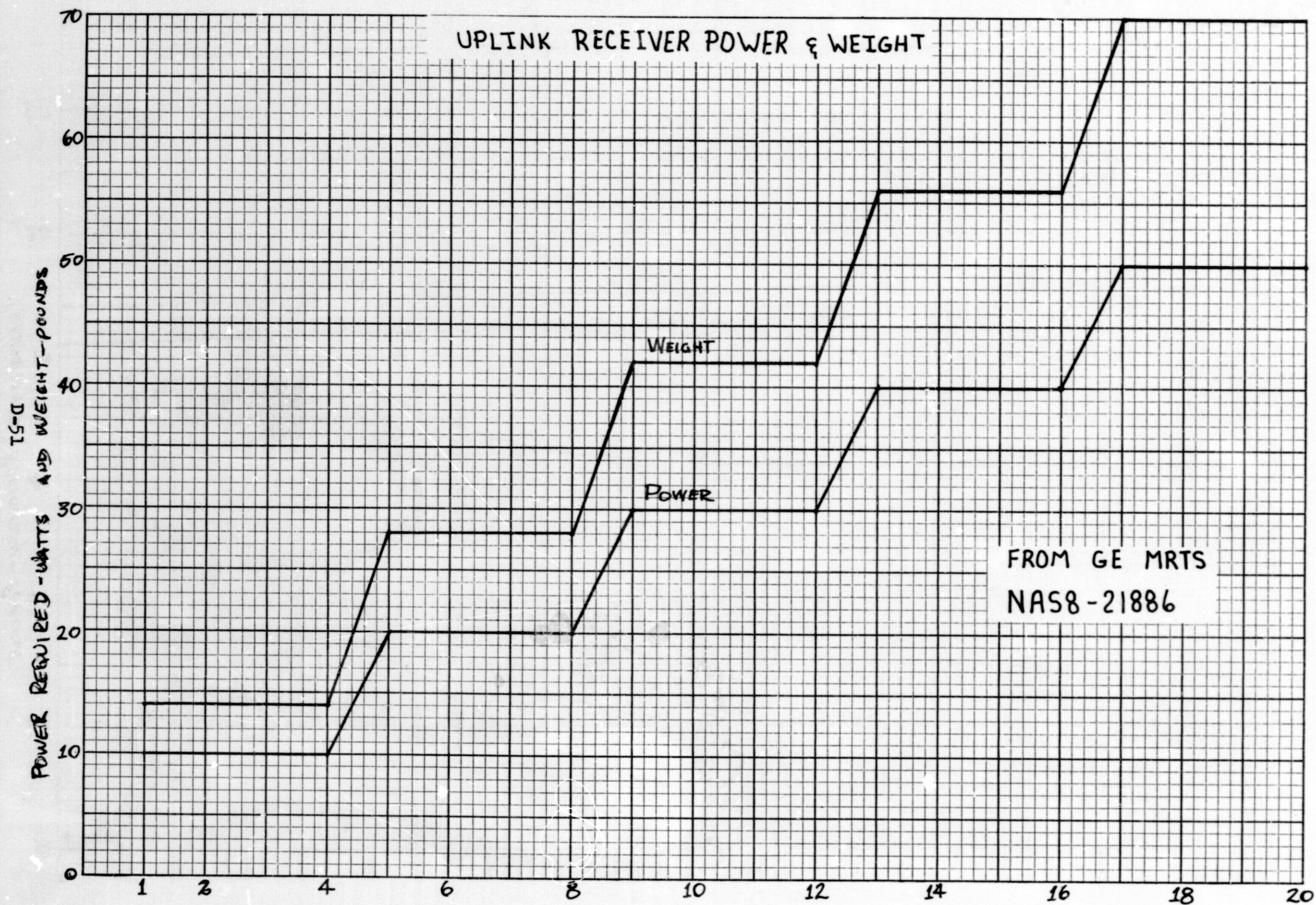
POWER DISTRIBUTION VOLUME



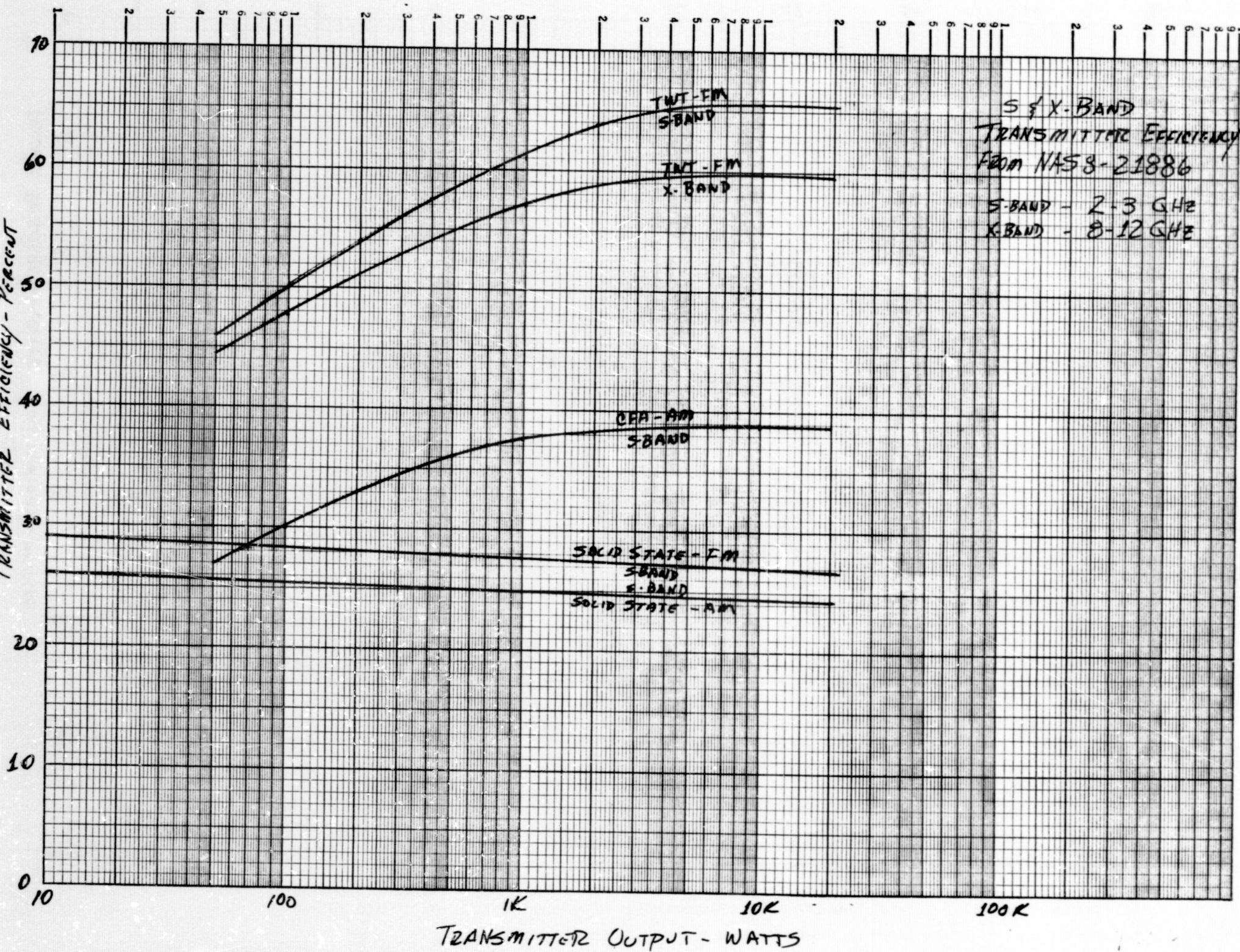
L/C FILTER VOLUME CHARACTERISTICS

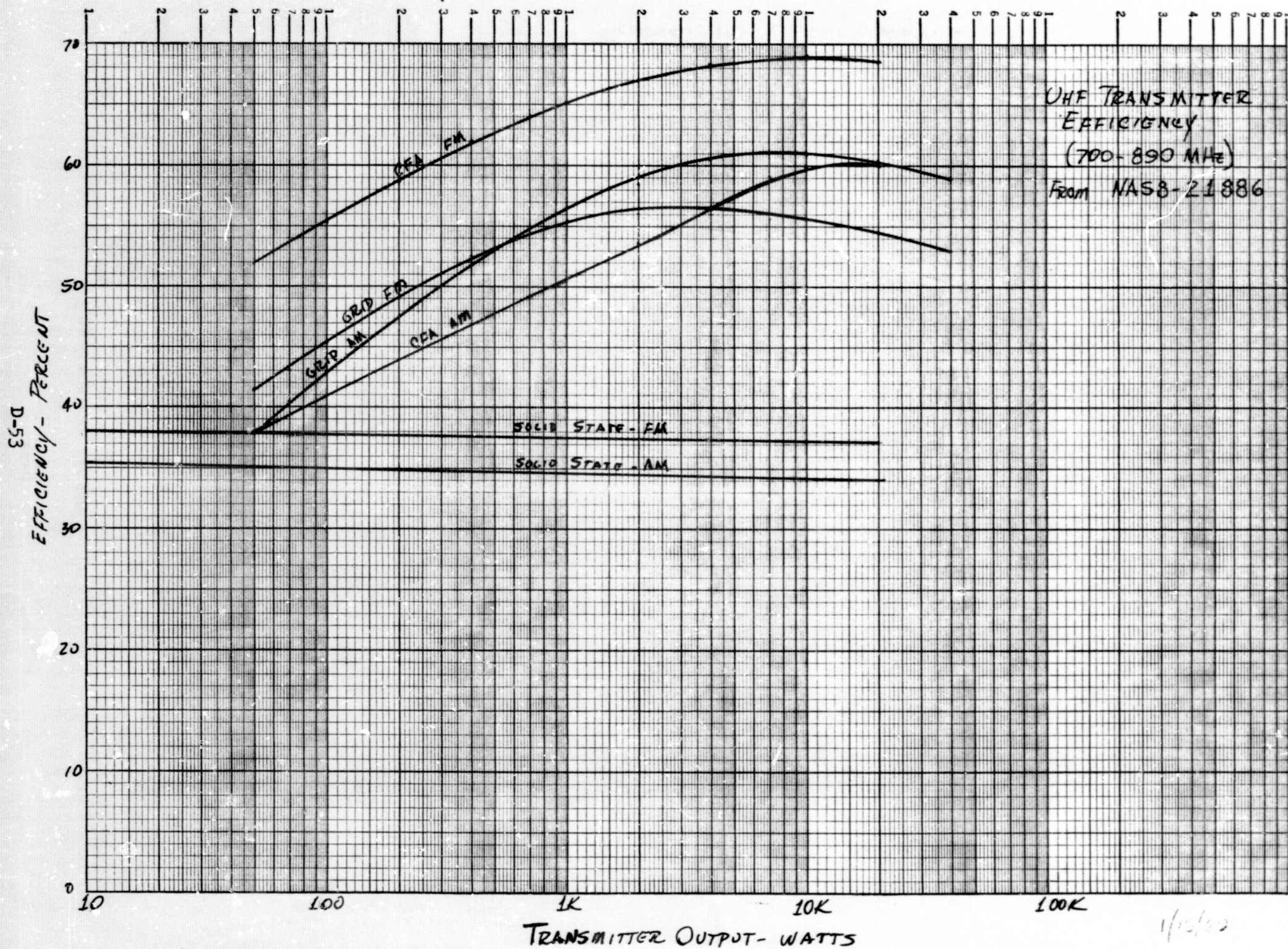


4/7/69



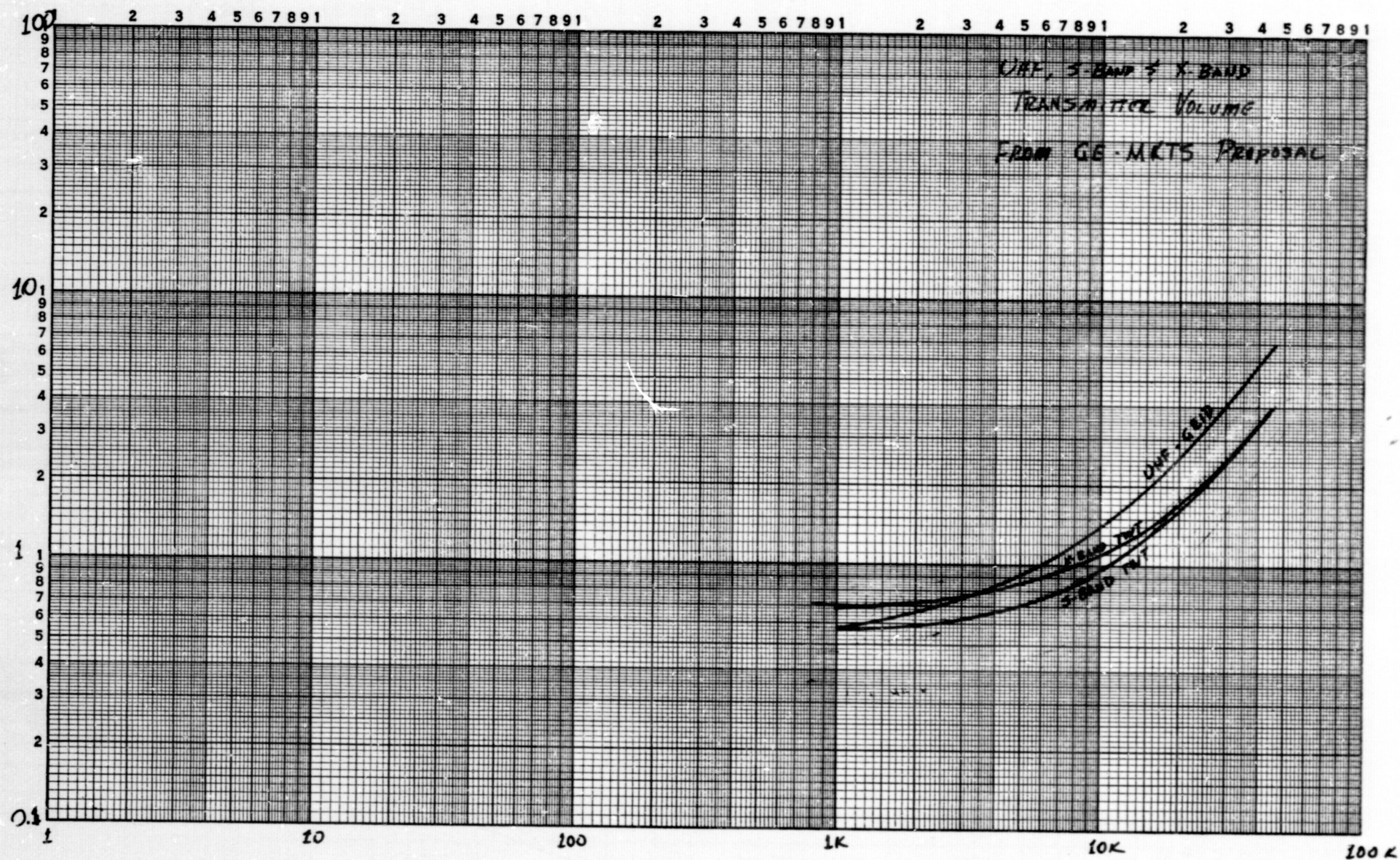
D-52 TRANSMITTER EFFICIENCY - PERCENT





D-54

TRANSMITTER VOLUME - ft³

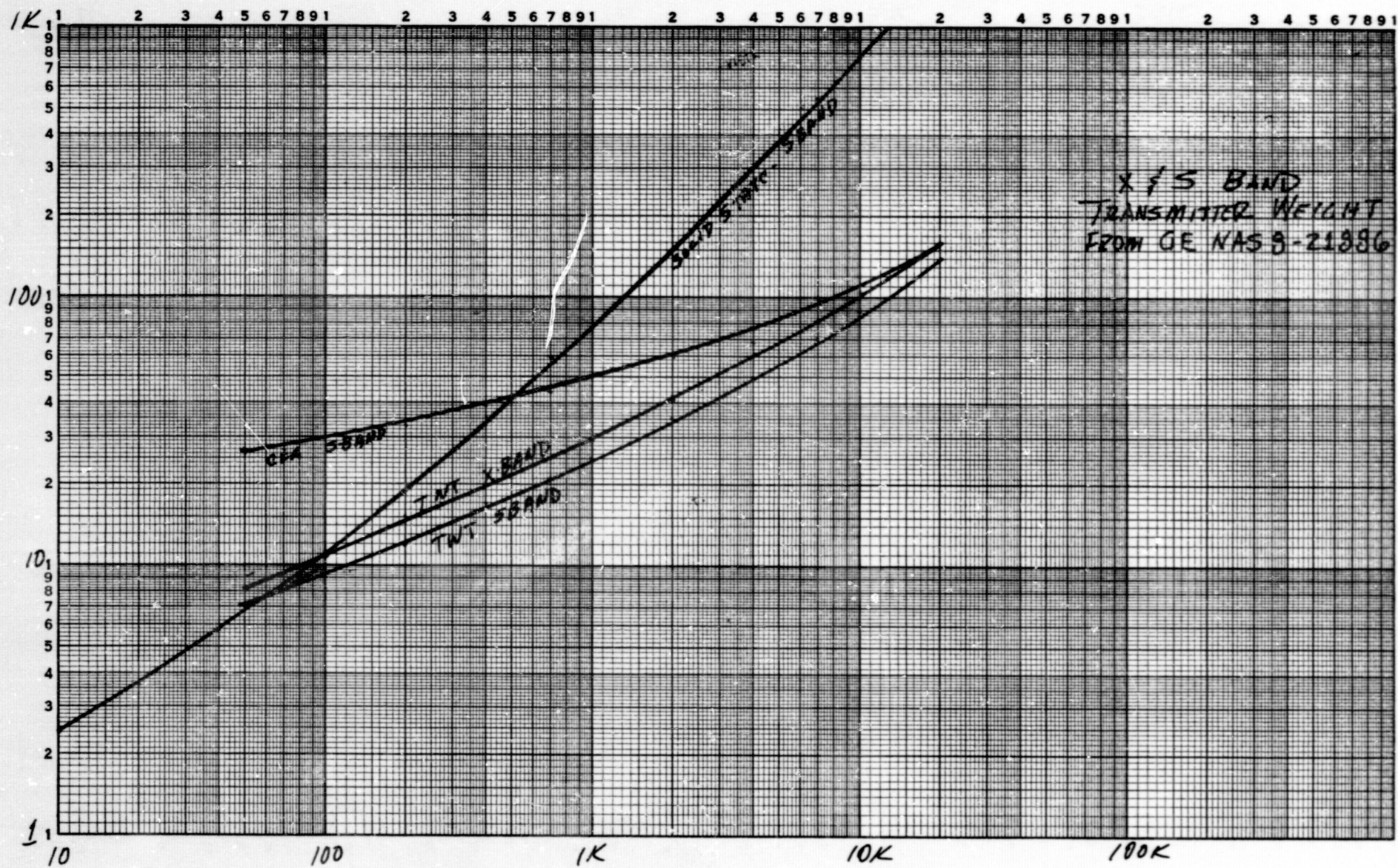


TRANSMITTER OUTPUT - WATTS

1/20/69

GM-10070-002

D-55
TRANSMITTER WEIGHT - POUNDS

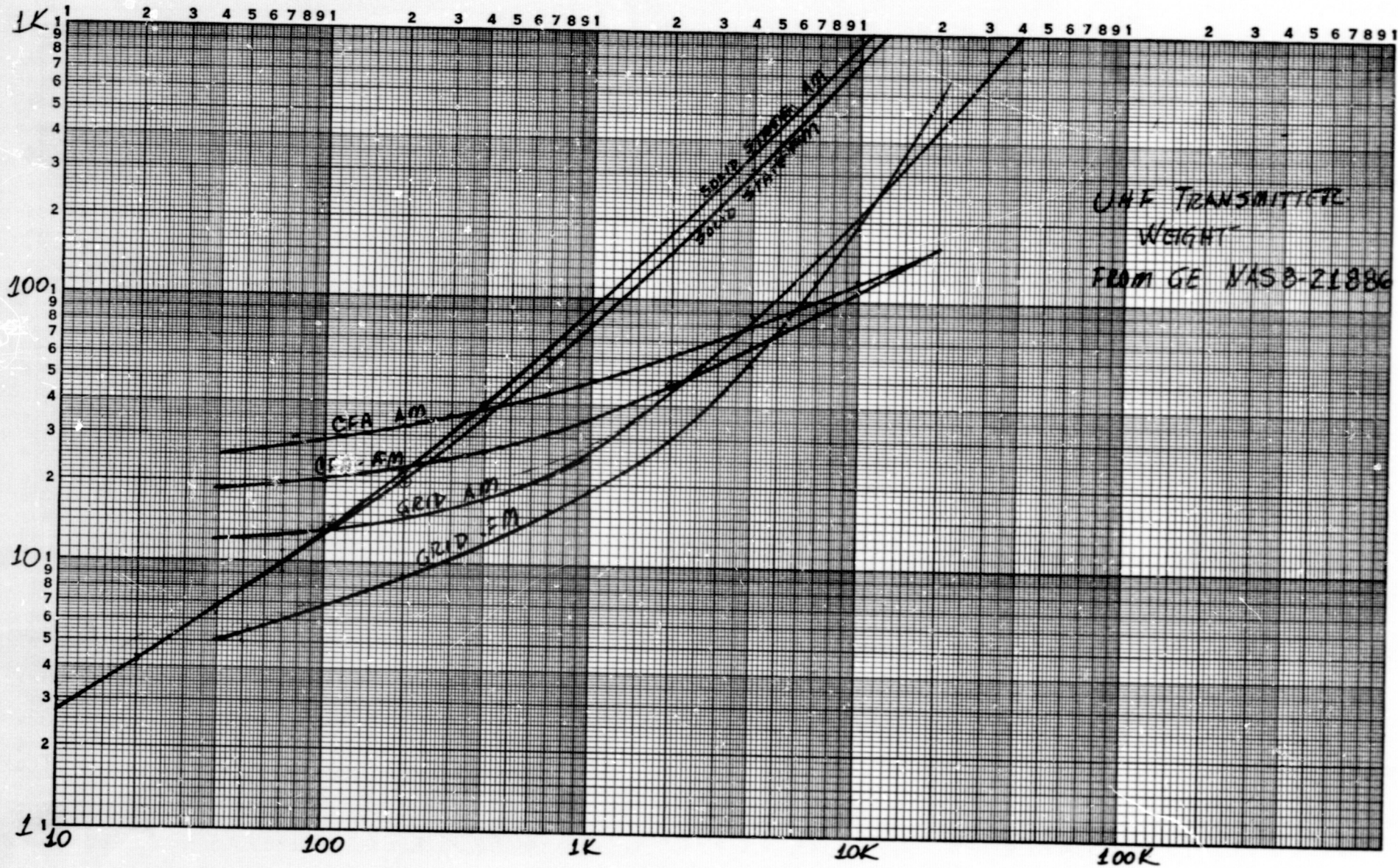


TRANSMITTER OUTPUT - WATTS

1/15/69

D-56

TRANSMITTER WEIGHT - POUNDS

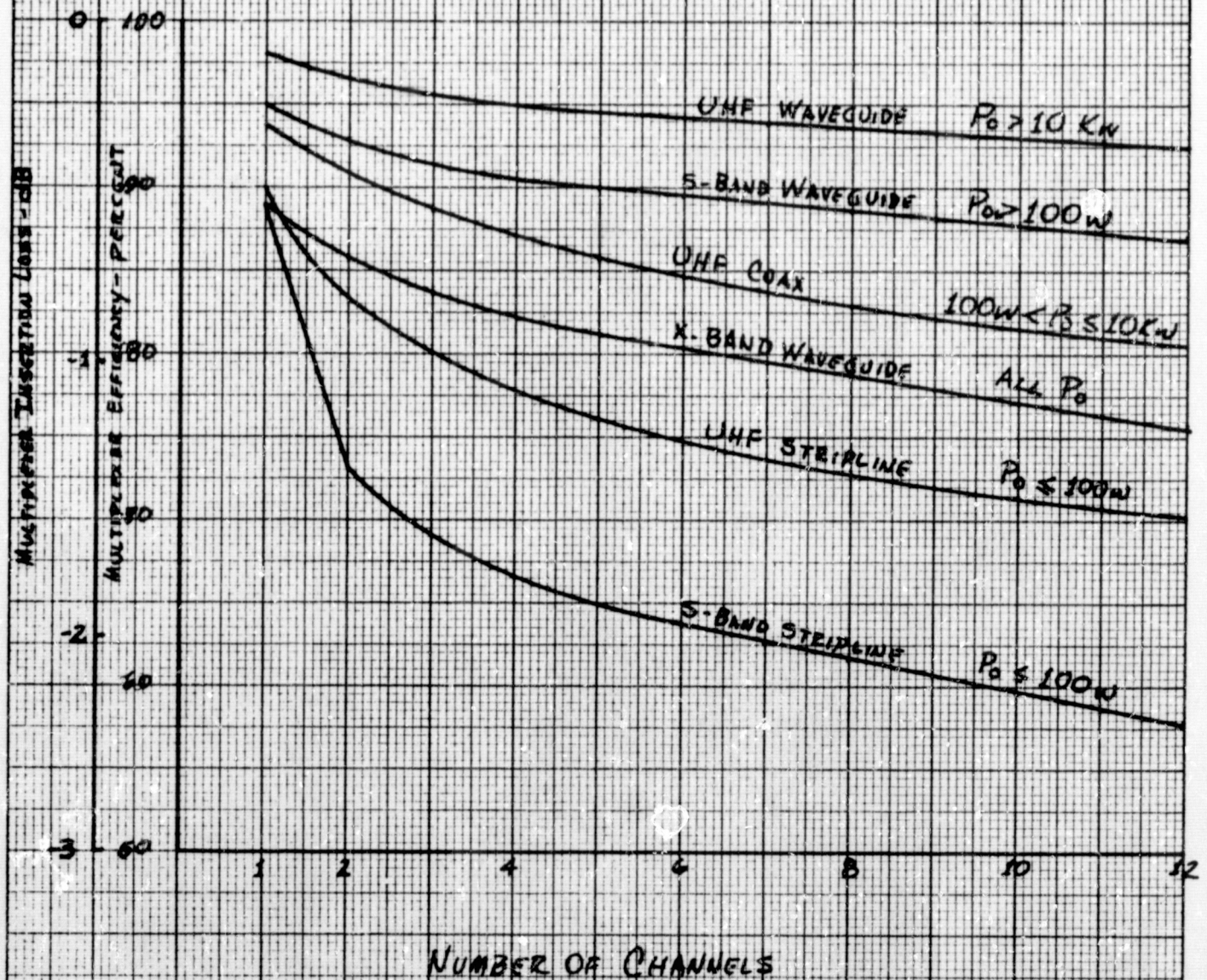


TRANSMITTER OUTPUT - WATTS

1/15/69

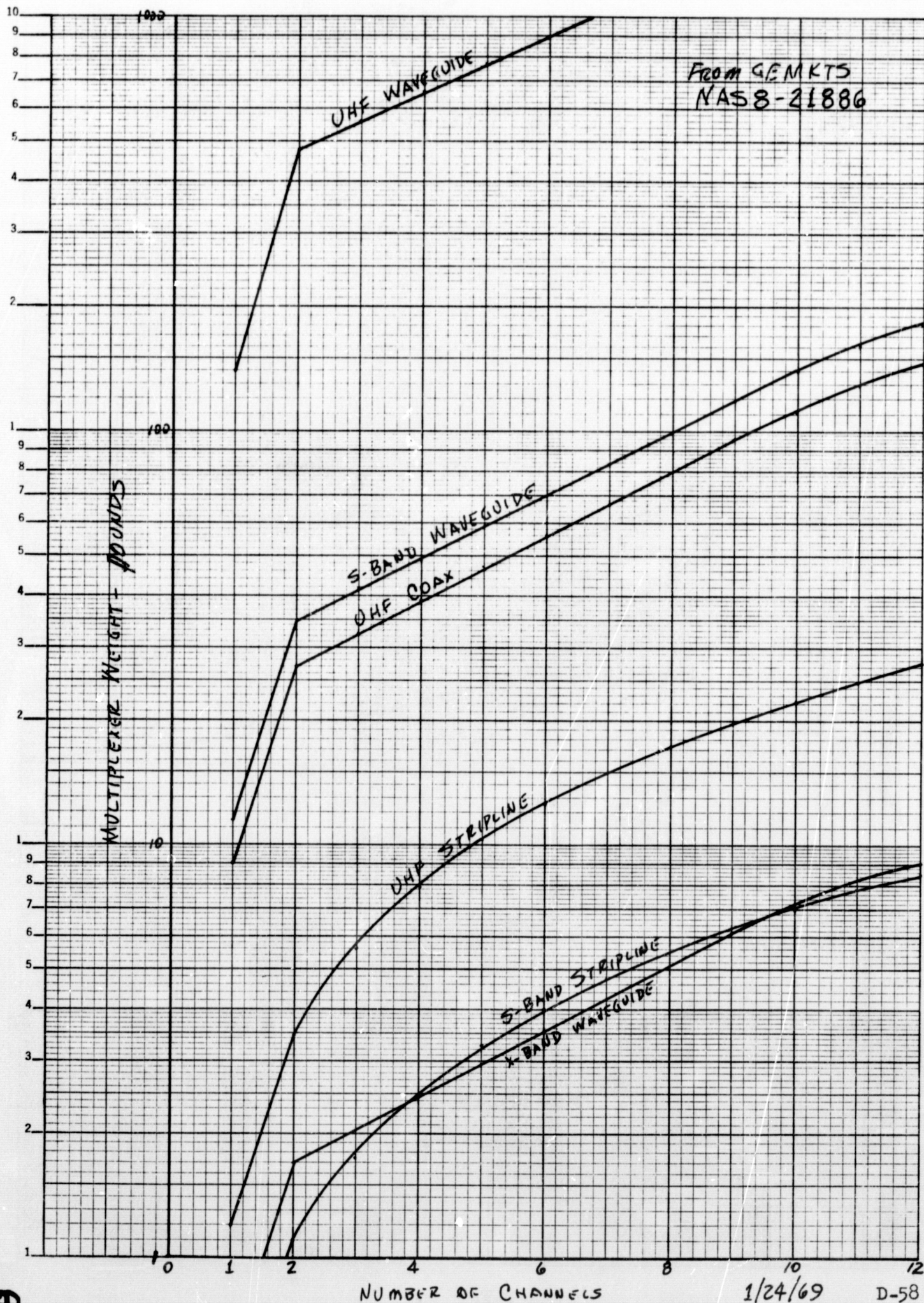
GDC-DCE70-002

EFFICIENCY OF HIGH POWER MULTIPLEXERS FROM GE MRS - NAS 8-21886



124/69

FROM GEMKTS
NAS 8-21886

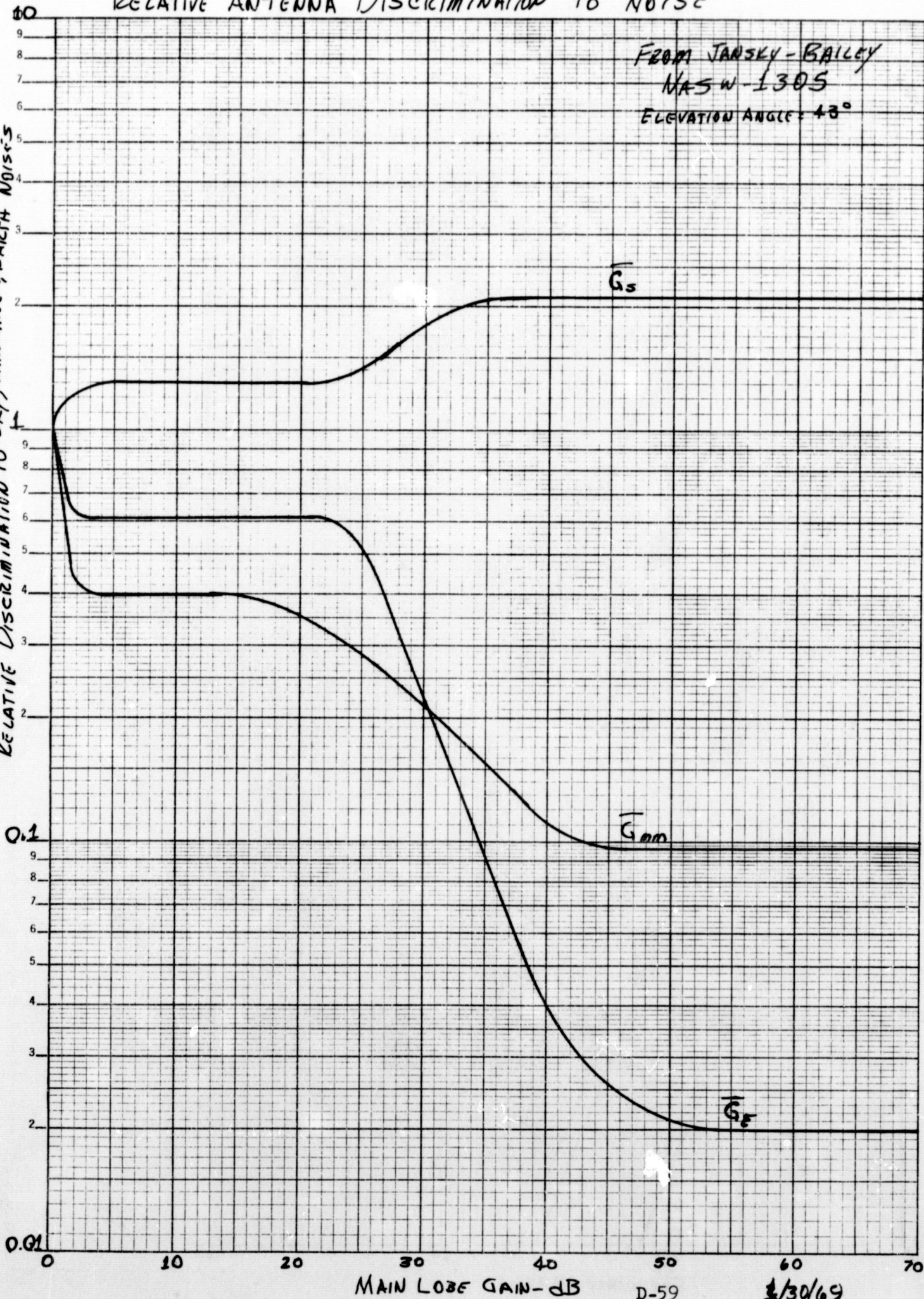


KE SEMI-LOGARITHMIC 359-71
KEUFFEL & ESSER CO. MADE IN U.S.A.
3 CYCLES X 70 DIVISIONS

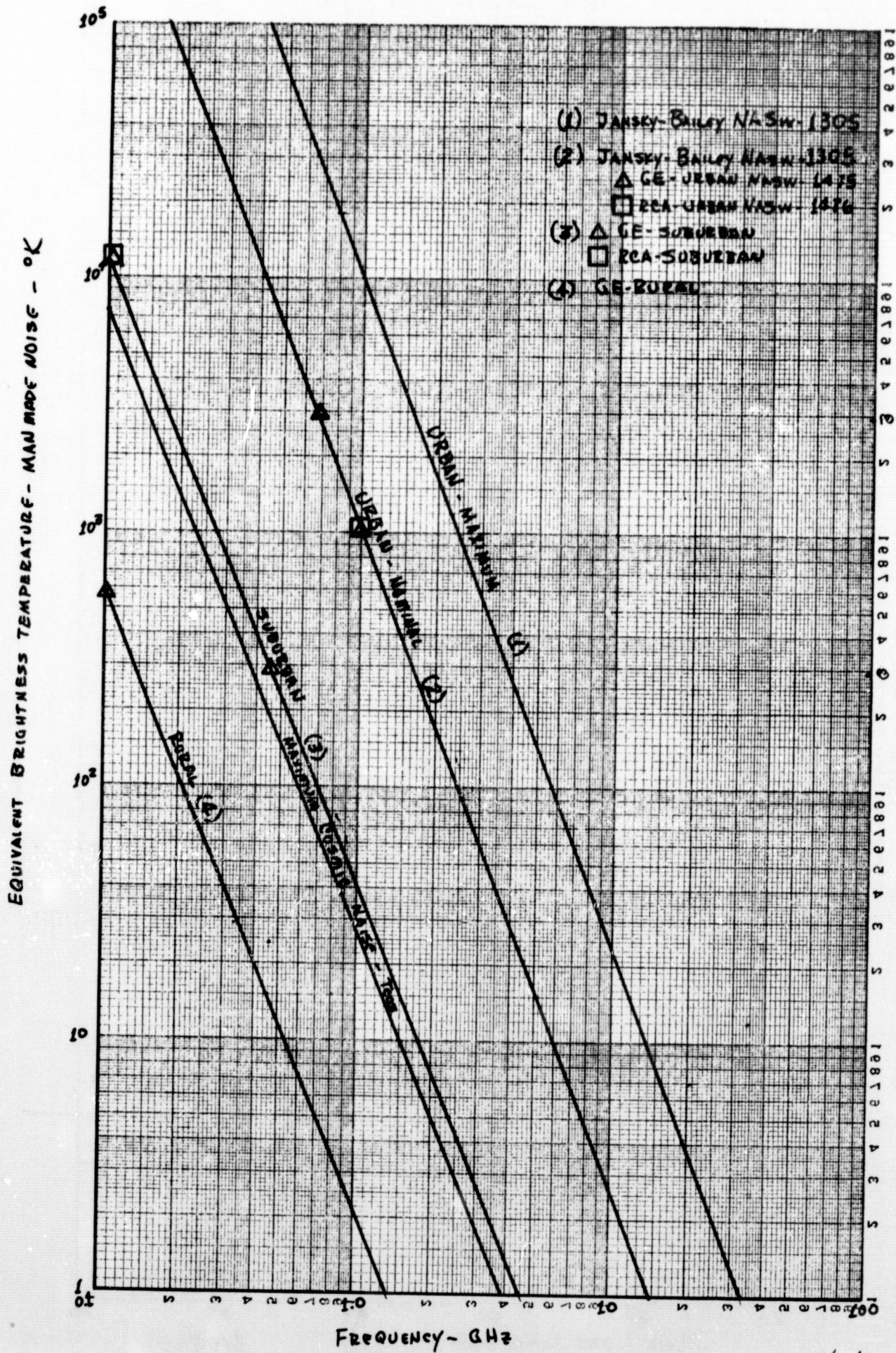
RELATIVE ANTENNA DISCRIMINATION TO NOISE

FROM JANSKY-BAILLY
NAS W-1305
ELEVATION ANGLE = 43°

RELATIVE DISCRIMINATION TO SKY, MAN MADE & EARTH NOISES



EQUIVALENT BRIGHTNESS TEMPERATURE MAN-MADE NOISE



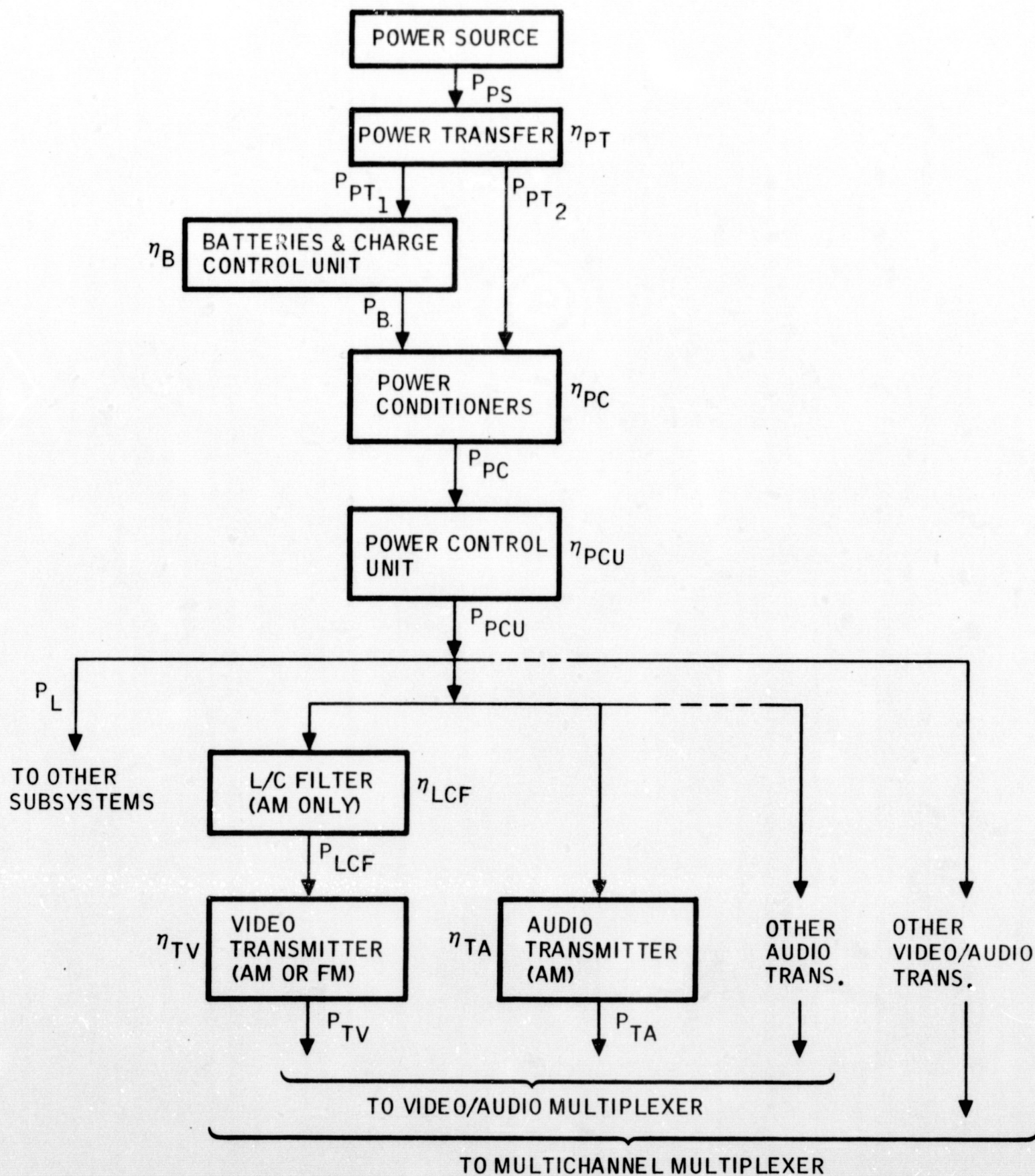


Figure D-1. Power model.

E

APPENDIX E

The following tables and figures present the geometrical constraints for each of the satellite configurations, A through H.*

* G has not yet been implemented in the program

Table E-1. Concept Summary

CONCEPT CODE	ALLOWABLE WEIGHT	ALLOWABLE VOLUME	ANTENNA TYPE	POWER TYPE	MANNED PROVISIONS	BOOSTER TYPE	COMMENTS
A	2,000 lb	98 cf	PETA or Rigid	Solar Cell	None	S-V or THIC	Multiple satellite version, 13 per S-V launch, 1 per THIC launch.
B	5,200	262	Phased Array, PETA, or Rigid Dish	Solar Cell	None	S-V or THID/Centaur	Multiple satellite version, 7 per S-V Launch, 1 per THID-Centaur.
C	9,000	810	PETA	Solar Cell	None	S-V or S-IB/Centaur	Multiple satellite version, 3 per S-V Launch, 1 per S-IB-Centaur
D	29,300	6,000	Rigid Dish Array	Solar Cell	BSM Module	S-V	BSM provides a shirt-sleeve environment for extended manned operation.
E	10,000	4,500	PETA	Solar Cell or Reactor	Dock	S-V with CSM	Dock provided, all operations by EVA or remote control.
F	42,000	6,000	PETA	Solar Cell	Pressurized Feed Module	S-V	MDA Module provides shirtsleeve environment for short periods (14 days).
G	42,000	6,000	Phased Array Dipoles	Solar Cell or Reactor	Optional Dock	S-V	Optional manned support with solar cell power only.

* H - same as C, except allowable WT = 42,000 lbs. and allowable VOL = 6,000

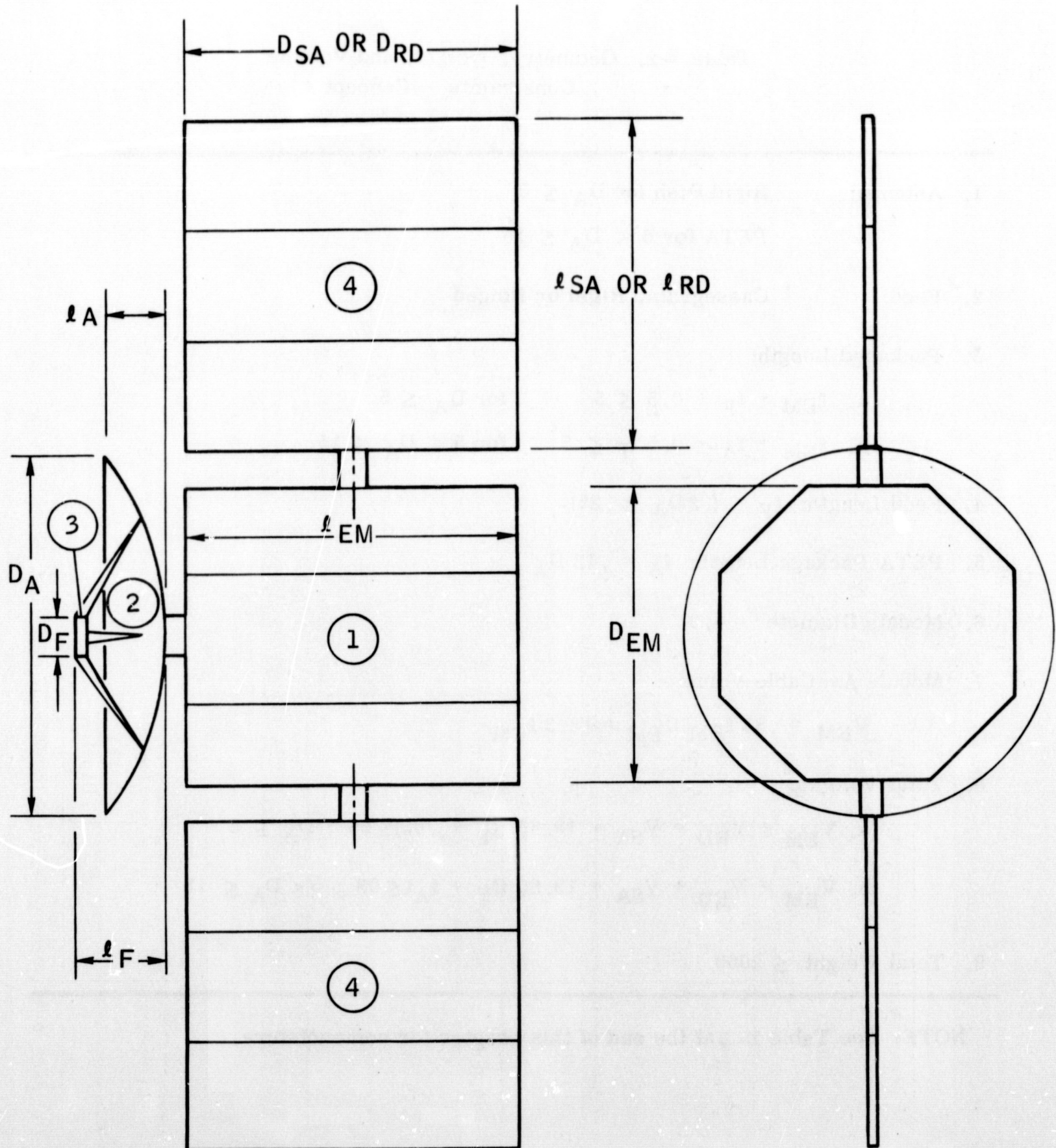


Figure E-1. Concepts A and B geometrical parameters

Table E-2. Geometry, Weight, and Volume
Constraints — Concept A

-
1. Antenna: Rigid Dish for $D_A \leq 5$
 PETA for $5 < D_A \leq 15$
 2. Feed: Cassegrain, Rigid or Hinged
 3. Packaged Length:
 - a. $\ell_{EM} + \ell_F + 0.5 \leq 5$ for $D_A \leq 5$
 - b. $\ell_{EM} + \ell_A + 0.5 \ell_F \leq 5$ for $5 < D_A \leq 15$
 4. Feed Length: $\ell_F = (.2 D_A + .25)$
 5. PETA Package Length: $\ell_A = .12 D_A$
 6. Module Diameter = 4.0
 7. Module Available Volume:

$$V_{EM} = .83 \ell_{EM} D_{EM}^2 = 13.3 \ell_{EM}$$
 8. Total Volume:
 - a. $V_{EM} + V_{RD} + V_{SA} + 19.65 (\ell_F + .05) \leq 98$; $D_A \leq 5$
 - b. $V_{EM} + V_{RD} + V_{SA} + 19.65 (\ell_F + \ell_A) \leq 98$; $5 < D_A \leq 15$
 9. Total Weight ≤ 2000
-

NOTE: See Table E-9 at the end of this chapter for nomenclature.

Table E-3. Geometry, Weight and Volume
Constraints — Concept B

-
1. Antenna: Rigid Dish for $D_A \leq 6.67$
PETA for $6.67 < D_A \leq 30.0$
 2. Feed: Cassegrain, Rigid or Hinged
 3. Packaged Length:
 - a. $l_{EM} + l_F + 0.5 \leq 7.5$ for $D_A \leq 6.67$
 - b. $l_{EM} + l_A + 0.5 l_F \leq 7.5$ for $6.67 < D_A \leq 30$
 4. Feed Length: $l_F = (.2 D_A + .25)$
 5. PETA Package Length: $l_A = .12 D_A$
 6. Module Diameter = 5.67
 7. Module Available Volume:

$$V_{EM} = 25.3 l_{EM}$$
 8. Total Volume:
 - a. $V_{EM} + V_{RD} + V_{SA} + 34.9 (l_F + 0.5) \leq 262$; $D_A \leq 6.67$
 - b. $V_{EM} + V_{RD} + V_{SA} + 34.9 (l_F + l_A) \leq 262$; $6.67 < D_A \leq 30$
 9. Total Weight $\leq 5,200$
-

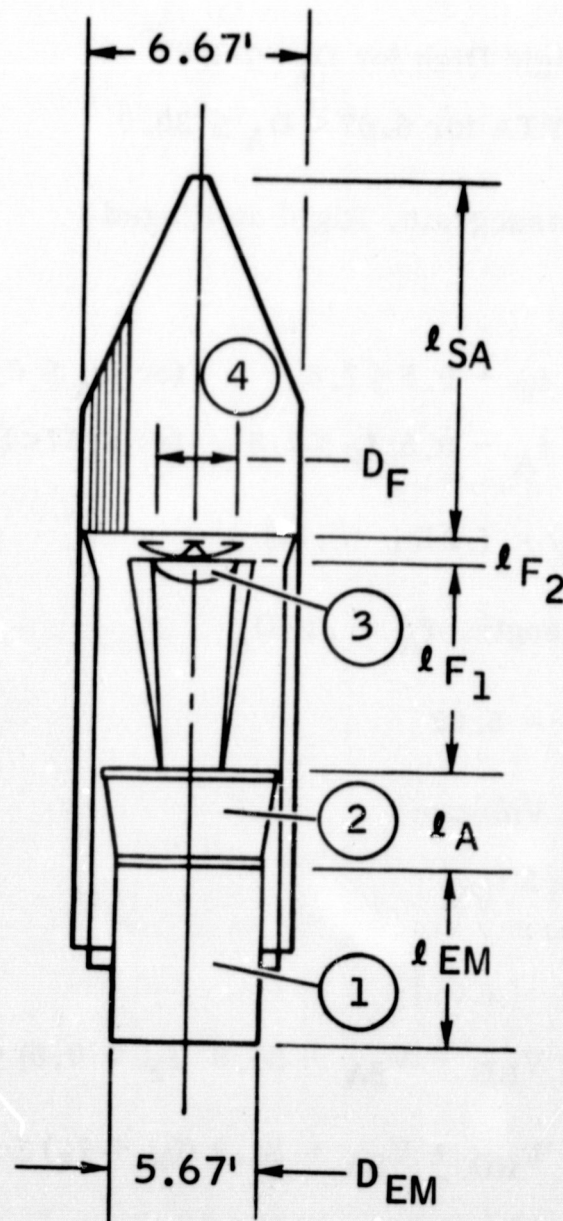


Figure E-2 Concept C configuration geometry.

Table E-4. Geometry, Weight and Volume
Constraints — Concept C (and H)

-
1. Antenna: PETA only $D_A \leq 56$ (100)
 2. Feed: Fixed Cassegrain for $D_A \leq 25$
Folding Cassegrain for $D_A > 25$
 3. Uplink Antenna: Rigid Dish With Cassegrain Feed, $D_U \leq 5.67$ (10)
 4. Package Length:
$$l_{EM} + l_A + l_F + l_{SA} \leq 26 \quad (35)$$
 5. Feed Length:
$$l_F = (.2 D_A + 0.5) + (.2 D_U + .25) \quad \text{for } D_A \leq 25$$

$$l_F = (.10 D_A + 0.5) + (.2 D_U + .25) \quad \text{for } 25 < D_A < 56 \quad (100)$$
 6. PETA Package Length = $0.11 D_A$
 7. Module Diameter = 5.67 (10)
 8. Module Available Volume:
$$V_{EM} = 25.3 l_{EM} \quad (78.5 l_{EM})$$
 9. Maximum Solar Array Volume:
$$V_{SA} = 170 \quad (4000)$$
 10. Total Volume:
 - a. $V_{EM} + V_A + V_{SA} + V_{RD} + V_F + V_U \leq 810 \quad (6000)$
 - b. $V_F = .786 (.2 D_A + 0.5) (.10 D_A + 1.0)^2 \quad \text{for } D_A \leq 25$
 - c. $V_F = .786 (.10 D_A + 0.5) (.10 D_A + 1.0)^2 \quad \text{for } 25 < D_A \leq 56 \quad (100)$
 - d. $V_U = .786 (.2 D_U + .25) \bar{D}_U^2$
 11. Total Weight $\leq 9,000$ (42,000)
-

Table E-5. Weight, Volume, and Geometry
Constraints for Concept D

1. Antenna:	Rigid Dish Array	$D_A \leq 15$
2. Radiator:	Active Boiler/Condenser System with Deployed Radiator/Condenser Units	
3. BSM Volume =	1600	
4. Total Available Volume =	6,000	
5. Total Spacecraft Volume:	$V_{SA} + V_{RD} + \Sigma V_A + \Sigma V_F + .786 (.2D_U + .25) D_U^2 \leq 4400$	
6. Fixed BSM Weights:		
	BSM Structure	3650
	Equip Racks	2050
	Atmosphere Supply	5775
	Water & Waste Management	708
	Crew Systems	<u>500</u>
	Total	12,683
7. Available Payload (with 20% Contingency) =	42,000	
8. Available Experiment Weight =	42,000 - 12,683 = 29,317	

Table E-6. Weight, Volume, and Geometry
Constraints for Concept E

1. Antenna: PETA $D_A \leq 90$

2. Feed Supports: Expandable

3. Uplink Antenna: Rigid Dish, Cassegrain Feed

4. Packaged Length:

$$* \ell_{EM} + \ell_F + \ell_A + \ell_U \leq 20$$

5. Uplink Antenna Length:

$$\ell_U = (.2 D_U + .25)$$

6. Total Spacecraft Volume:

$$V_{EM} + V_A + V_{RD} + V_{SA} + V_U + V_F \leq 4,500$$

7. Equipment Module Diameter = $0.10 D_A$

8. Uplink Antenna Volume = $.786 D_U^2 (.2 D_U + .25)$

9. Fixed Satellite Weights:

Docking Cone	300
Crew Systems	500
Emergency Atmosphere	<u>2400</u>
Total	3200

10. Available Weight = $9800 - 3200 = 6600$

*In this concept, the equipment module is the feed module.

Table E-7. Weight, Volume, and Geometry
Constraints for Concept F

1. Reflector: PETA $D_{\text{Total}} \leq 100$

2. Antenna Beams: 6 Beams

$$D_A \leq 33$$

$$D_{\text{Total}} = 2.00 D_{A(\text{Max})} + 34$$

3. Uplink Antenna: Rigid Dish, Cassgrain Feed

4. Packaged Length — Not Critical

5. Packaged Volume:

Fixed Volumes —

MDA	1300
-----	------

Feed Booms	<u>170</u>
------------	------------

a. Total Fixed	1470
----------------	------

b. Total Fairing Volume	6000
-------------------------	------

c. Available Volume	4530
---------------------	------

d. $*V_{\text{Reflector}} + V_U + V_{\text{SA}} + V_{\text{RD}} \leq$	4530
---	------

e. $V_U = .786 D_U^2 (.2 D_U + .25)$	
--------------------------------------	--

6. Fixed Weights:	MDA	8000
-------------------	-----	------

	Feed Booms	<u>850</u>
--	------------	------------

	Total Fixed =	8850
--	---------------	------

7. Available Weight = 42,000 - 8850 = 33,150

*Use D_{Total} to compute $V_{\text{Reflector}}$

Table E-8. Concept G Constraints

-
1. Antenna: Double Dipole Array
 2. Uplink Antenna: Rigid Dish, Cassegrain Feed
 3. Power Source: Solar Array or Nuclear
 4. Available Volume = 6000

$$V_{EM} + V_A + V_U + V_{PP} \leq 6000$$

5. Available Weight = 42,000
-

Table E-9. Nomenclature.

l_{EM}	=	Length of Equipment Module
D_{EM}	=	Diameter of Equipment Module
P_{PRIME}	=	Power Source Output in Watts
N	=	Satellite Life in Years
A	=	Area of Solar Array in Square Feet
l_F	=	Length of Feed in Feet
l_A	=	Height of Stowed Antenna in Feet
D_A	=	Diameter of Deployed Antenna in Feet
W_S	=	Weight of Spacecraft Less Stationkeeping
P_{DIS}	=	Power Dissipated by Thermal Control Subsystem in Watts
P_{XMTR}	=	RF Power Radiated by Transmitter in Watts
V_{EM}	=	Volume of Equipment Module in Cubic Feet
P_{DIST}	=	Power Generated by Transmitter to be Dissipated in Watts
P_{DISS}	=	Power Generated by Other Subsystems to be Dissipated in Watts
η	=	Transmitter Efficiency
V_{RD}	=	Volume of Radiator
V_{SA}	=	Volume of Stowed Solar Array
D_U	=	Diameter of the Uplink Antenna
V_F	=	Volume of Feed
V_U	=	Volume of Uplink Antenna
l_U	=	Height of Stowed Uplink Antenna
V_{PP}	=	Volume of Prime Power System
