

NASA-CR-176395
19860004987

The Telecommunications and Data Acquisition Progress Report 42-83

July–September 1985

E. C. Posner
Editor

November 15, 1985



National Aeronautics and
Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

DEC 1 1985

RESEARCH CENTER
LIBRARY, NASA
WASHINGTON, VIRGINIA



NF00442

The Telecommunications and Data Acquisition Progress Report 42-83

July–September 1985

E. C. Posner
Editor

November 15, 1985



National Aeronautics and
Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

N86-14457#
THRU
N86-14476#

The research described in this publication was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology.

Preface

This quarterly publication provides archival reports on developments in programs managed by JPL's Office of Telecommunications and Data Acquisition (TDA). In space communications, radio navigation, radio science, and ground-based radio astronomy, it reports on activities of the Deep Space Network (DSN) and its associated Ground Communications Facility (GCF) in planning, in supporting research and technology, in implementation, and in operations. Also included is TDA-funded activity at JPL on data and information systems and reimbursable DSN work performed for other space agencies through NASA. The preceding work is all performed for NASA's Office of Space Tracking and Data Systems (OSTDS).

In geodynamics, the publication reports on the application of radio interferometry at microwave frequencies for geodynamic measurements. In the search for extraterrestrial intelligence (SETI), it reports on implementation and operations for searching the microwave spectrum. The latter two programs are performed for NASA's Office of Space Science and Applications (OSSA).

Finally, tasks funded under the JPL Director's Discretionary Fund and the Caltech President's Fund which involve the TDA Office are included.

This and each succeeding issue of the TDA Progress Report will present material in some, but not necessarily all, of the following categories:

OSTDS Tasks:

- DSN Advanced Systems
 - Tracking and Ground-Based Navigation
 - Communications, Spacecraft-Ground
 - Station Control and System Technology
 - Network Data Processing and Productivity
- DSN Systems Implementation
 - Capabilities for New Projects
 - Networks Consolidation Program
 - New Initiatives
 - Network Sustaining
- DSN Operations
 - Network Operations and Operations Support
 - Mission Interface and Support
 - TDA Program Management and Analysis
- GCF Implementation and Operations
- Data and Information Systems

OSSA Tasks:

- Search for Extraterrestrial Intelligence
- Geodynamics
 - Geodetic Instrument Development
 - Geodynamic Science

Discretionary Funded Tasks

Contents

OSTDS TASKS DSN Advanced Systems COMMUNICATIONS, SPACECRAFT-GROUND

Coupled Translations of the 64-Meter Antenna Subreflector Supports	1-D1
M. S. Katow	
NASA Code 310-20-65-62-00	

Seismic Analysis of the Large 70-Meter Antenna, Part II: General Dynamic Response and a Seismic Safety Check	12 D2
K. Kiedron and C. T. Chian	
NASA Code 310-20-65-64-00	

STATION CONTROL AND SYSTEM TECHNOLOGY

In Search of a 2-dB Coding Gain	26 D3
J. H. Yuen and Q. D. Vo	
NASA Code 310-30-71-83-02	

Decoding Convolutionally Encoded Images	34 D4
G. H. Pitt, III, and L. Swanson	
NASA Code 310-30-71-84-06	

Erasur Information for a Reed-Solomon Decoder	39 D5
G. H. Pitt, III, and L. Swanson	
NASA Code 310-30-71-84-06	

A VLSI Single Chip 8-Bit Finite Field Multiplier	45 D6
I. S. Hsu, L. J. Deutsch, T. K. Truong, and H. M. Shao	
NASA Code 310-30-70-84-08	

A VLSI Single Chip (255, 223) Reed-Solomon Encoder	51 D7
I. S. Hsu, L. J. Deutsch, T. K. Truong, and I. S. Reed	
NASA Code 310-30-70-84-08	

Carrier Tracking by Smoothing Filter Can Improve Symbol SNR	57 D8
C. A. Pomalaza-Raez and W. J. Hurd	
NASA Code 310-30-70-84-02	

A Class of Optimum Digital Phase Locked Loops for the DSN Advanced Receiver	63 D9
R. Kumar and W. J. Hurd	
NASA Code 310-30-70-84-02	

NETWORK DATA PROCESSING AND PRODUCTIVITY

A Laser Plotting System for VLSI Chip Layouts	81 D10
J. A. Harding and L. J. Deutsch	
NASA Code 310-40-72-15-00	

DSN Systems Implementation CAPABILITIES FOR NEW PROJECTS

Mark IV-A DSCC Telemetry System Description	92 D11
D. L. Ross	
NASA Code 314-40-41-81-13	

Comparison of GSFC and JPL VLBI Modeling Software: Benchmark 101 *D12*
 O. J. Sovers and C. Ma
 NASA Code 314-40-51-36-05

**Controller and Interface Module for the High-Speed Data Acquisition
 System Correlator/Accumulator** 113 *D13*
 S. S. Brokl
 NASA Code 314-40-22-60-12

**A General Monitor and Control Interface to the VAX UNIBUS by Way of the
 DR11-C I/O Port** 125 *D14*
 S. S. Brokl
 NASA Code 314-40-22-10-12

DSN Operations TDA PROGRAM MANAGEMENT AND ANALYSIS

Mark IVA Project Training Evaluation 134 *D15*
 S. N. Stephenson
 NASA Code 314-30-32-20-13

Error and Erasure Probabilities for Galileo Uplink Code 165 *D16*
 J. B. Berner, R. J. McEliece, and E. C. Posner
 NASA Code 314-40-31-81-12

OSSA TASKS Search for Extraterrestrial Intelligence

A Wide-Band, High-Resolution Spectrum Analyzer 180 *D17*
 M. P. Quirk, H. S. Wilck, and M. J. Grimm
 NASA Code 199-50-62-11-06

A Signal Detection Strategy for the SETI All Sky Survey 191 *D18*
 J. Solomon, W. Lawton, M. P. Quirk, and E. T. Olsen
 NASA Code 199-50-62-09-06

DISCRETIONARY FUNDED TASKS

The Number of Stable Points of an Infinite-Range Spin Glass Memory 209 *D19*
 R. J. McEliece and E. C. Posner
 NASA Code 404-00-73-03-27

Coupled Translations of the 64-Meter Antenna Subreflector Supports

M. S. Katow

Ground Antenna and Facilities Engineering Section

The Tricone subreflector assembly of the 64-m diameter antenna is supported from the apex of the quadripod by six flexure rods. The subreflector can be translated in the three orthogonal axes by jackscrews at the ends of each rod. The arrangement of the six rods is described and analysis of the position errors introduced by coupled translations is presented. The results indicate that negligible RF gain losses caused by axis-coupled translations (less than 0.05 dB) are possible at X-band (8.45 GHz) operation.

I. Introduction

The existing subreflector of the 64-m antenna, employing the Tricone RF feed system, is supported from the quadripod apex by six rods, each with flexures at both ends as shown in Figs. 1 and 2. A jackscrew is located at one end of each rod. Three rods are used for the *A*-axis motion, two rods for the *Y*-axis motion and one rod for the *X*-axis motion. The subreflector can be translated in the three orthogonal directions *X*, *Y* and *Z* by changes in the lengths of these rods by motorized jackscrews using worm gears. The RF feed cone-to-RF feed cone rotation of the subreflector is provided by a rotational drive mechanism.

In the Tricone RF feed system, the sloped axis of the subreflector is rotated about the symmetric axis of the reflector to one of the three main RF feed positions *A*, *B* or *C* (Fig. 3).

Because the reflective surfaces (the main reflector and the subreflector) are rigged or set at 45 degrees elevation angle, the rotation of the antenna to other elevation angles introduces more gravity loading distortions that result in RF phase center offsets at the paraboloid's prime focus, which in turn

produce RF gain losses. Accordingly, the subreflector can be translated with respect to its supporting quadripod by length changes of the supporting rods to minimize these offsets and reduce the antenna's resulting RF gain losses.

The arrangement of the six supporting rods results in a statically determinate type truss where a small length change of any rod produces a minimal change in the axial force. The use of flexures has minimized any moments or bending stresses in the rods.

The geometrical arrangement of the subreflector support rods is described in this article together with the analysis and computation of the cross couplings of the corrective translations and the resulting gain losses. The goal was to estimate how much the gain losses can be caused by these axis-coupling position errors.

II. Analysis

The coordinate system of the supporting rods was chosen parallel to the symmetric axis of the main reflector with the

angular difference of 4 deg and 31 min between the axis of the main reflector and the subreflector axis. In the isometric view of Fig. 4, the angles between the hyperboloid axis and the two planes parallel to the symmetric axis of the main reflector for the subreflector in position *A* are shown.

The subreflector is supported in the axial-*Z* direction by three flexure rods as shown in Figs. 1 and 2. These rods also constrain the rotational moments about the *X* and *Y* axes to account for three degrees of freedom. Three jackscrews driven in unison control the *Z*-axis translation.

In the lateral-*Y* direction, two rods driven in unison are used to account for the rotation constraint about the *Z*-axis as well as the *Y* translation. These two rods account for two more degrees of freedom.

By arranging the design to connect the sliding anchor on the subreflector end of the *X*-lateral constraint to the end of a *Y*-lateral strut, as shown in Fig. 1, it was possible to eliminate the axis-coupled translations in the *X*-direction resulting from the large *Y*-lateral corrections. The gain loss due to the induced azimuth pointing (or boresight) error would not have satisfied operational requirements without correction. The *X*-lateral constraint adds to the six degrees required for stability.

Starting with the subreflector aligned with the main reflector's symmetric axis at the 45 deg elevation angle, raising the 64-m antenna to 84 deg elevation generates 7.62 cm (3.0 in.) lateral offset between the focus of the best fit paraboloid of the main reflector and the deflected virtual focus of the subreflector system. This lateral *Y* defocus can be continuously corrected by jackscrews adjusting the lengths of the *Y* lateral rods through the change in the elevation angles. The cross-coupled axial *Z* defocus resulting from the lateral *Y* corrections is described in Fig. 5 and defined in Table 1.

During the elevation angle change, axial *Z* defocus also occurs. From the set 45-deg position, an axial *Z* defocus of +0.76 cm occurs close to 80-deg elevation angle and of -2.03 cm

at 10-deg elevation angle, as changes from the 45-deg elevation angle.

When the axial *Z* defocus is corrected by the jackscrews on the *Z* axial rods, four coupled sources of gain losses are generated. Figure 6 shows the lateral cross-coupled defocus in the *Y*-direction (view-AA) and in the *X* direction (view-BB) defining two of the sources of gain losses with calculated lateral offsets in Table 2. The third lateral offset occurs because the axis of the subreflector is tilted 4 deg and 31 min to the axial *Z* motion. Figure 7 describes the lateral offsets from this source with the numerical offsets shown in Table 3.

The vector sum of the lateral offsets of Tables 2 and 3 are added and shown in Table 4 for the three basic RF feed cone positions only for the maximum -2.03 cm *Z* focus position.

The subreflector's lateral offsets can be redefined as the offsets of the vertex, and the lateral defocus of Fig. 9 can be calculated. This defocus results in a gain loss as calculated in Fig. 8. The gain loss calculations of Fig. 8 are described in Ref. 1. Additional gain loss occurs due to boresight errors as shown in Fig. 9. This loss should be added to the gain loss described in Table 4.

III. Summary

The existing six supporting rods of the subreflector assembly generate axis-coupled translations which, when combined with the translations required to correct the phase centers offsets, result in less than 0.05 dB RF gain loss at X-band (8.45 GHz).

For a higher RF frequency operation the corrective translations will be significant and should be added to the new microprocessor now used to control the subreflector. New subreflector rotations about the *X* and *Y* axes can also be superimposed, if required later on, by adding differential gear boxes to the shafts between the jackscrews driving the three *Z*-axis supporting rods.

Reference

1. Katow, M. S. "34-Meter Antenna-Subreflector Translations to Maximize RF Gain," *TDA Progress Report 42-62*, Jan. and Feb. 1981, pp. 112-120.

**Table 1. Gain loss from Z-axis defocus resulting from lateral-Y motion
(see Fig. 5, cone positions A, B, or C)**

Sequence	Y-Offset (YOF), cm (in.)	Angle (α), deg ^a	A, cm (in.) ^b	B Defocus, cm (in.) ^c	Gain Loss, dB ^d
1	2.54 (1.00)	0.4668	311.78 (122.748)	0.01 (0.004)	<0.0001
2	5.08 (2.00)	0.9336	311.75 (122.736)	0.04 (0.016)	0.0004
3	7.62 (3.00)	1.4004	311.70 (122.715)	0.09 (0.035)	0.0023
4	10.16 (4.00)	1.8674	311.62 (122.687)	0.17 (0.067)	0.0079

^a $\alpha = \sin^{-1} (YOF/311.79)$

^b $A = \sqrt{(311.79)^2 + (YOF)^2}$

^c $B = 311.79 - A$

^dSee Fig. 8

Table 2. Lateral X (Fig. 6, view BB) or Y (Fig. 6, view AA) travel cross-coupled to $\pm Z$ axial focus motion for cone positions A, B, or C

Axial-Z Focus Motion, cm (in.)	H, cm (in.) ^a	I, cm (in.) ^b	J, cm (in.) ^c	K, cm (in.) ^d
+0.76 (+0.3)	253.371 (99.752)	+0.0010 (+0.0004)	251.4589 (99.000)	+0.0011 (+0.0004)
-2.03 (-0.8)	253.362 (99.749)	+0.0080 (+0.0032)	251.4518 (98.9968)	+0.0082 (+0.0032)

^a $H = \sqrt{253.370^2 - Z^2}$

^b $I = 253.370 - H$

^c $J = \sqrt{251.460^2 - Z^2}$

^d $K = 251.460 - I$

Table 3. Lateral X and Y offsets for $\pm Z$ axial focus motions. Angular offset to subreflector axis is 4 deg. 31 min. (See Fig. 7 for lateral Y offset and lateral X offset.)

Cone Position	A, B	A	B	C
Axial-Z Focus Motion, cm (in.)	L , cm (in.) ^a	M , cm (in.) ^b	N , cm (in.) ^c	P , cm (in.) ^d
+0.76 (+0.3)	0.030 (0.012)	-0.052 (-0.020)	0.052 (0.020)	0.060 (0.236)
-2.03 (-0.8)	-0.080 (0.031)	0.139 (0.055)	-0.139 (-0.055)	0.060 (-0.063)
^a L = 0.76 tan (2.2566); L = 2.03 tan (2.2566) ^b M = 0.76 tan (3.9106); M = 2.03 tan (3.9106) ^c N = 0.76 tan (3.9106); N = 2.03 tan (3.9106) ^d P = 0.76 tan (4.5167); P = 2.03 tan (4.5167)				

Table 4. X-band (8.45 GHz) gain loss from cross-coupled lateral and boresight error for -2.03 cm Z focus (see Figs. 7 and 9)

Parameter	Cone Position		
	A	B	C
Lateral-Y Travel, cm	I = +0.008	I = +0.001	I = +0.001
Lateral-X Travel, cm	K = +0.008	K = +0.008	K = +0.008
Lateral-Y Offset, cm	L = -0.080	L = -0.080	P = -0.160
Lateral-X Offset, cm	M = 0.139	N = -0.139	0
Subreflector Vertex Vector Sum Offset, cm	0.164	0.153	0.159
Lateral Defocus, cm	0.130	0.122	0.126
Lateral Defocus Gain Loss, dB	-0.00022	-0.00019	-0.00021
Boresight Error, deg	0.0023	0.0022	0.0023
Boresight Error Gain Loss, dB	-0.049	-0.045	-0.049
Total Gain Loss, dB	-0.0492	-0.0452	-0.0491

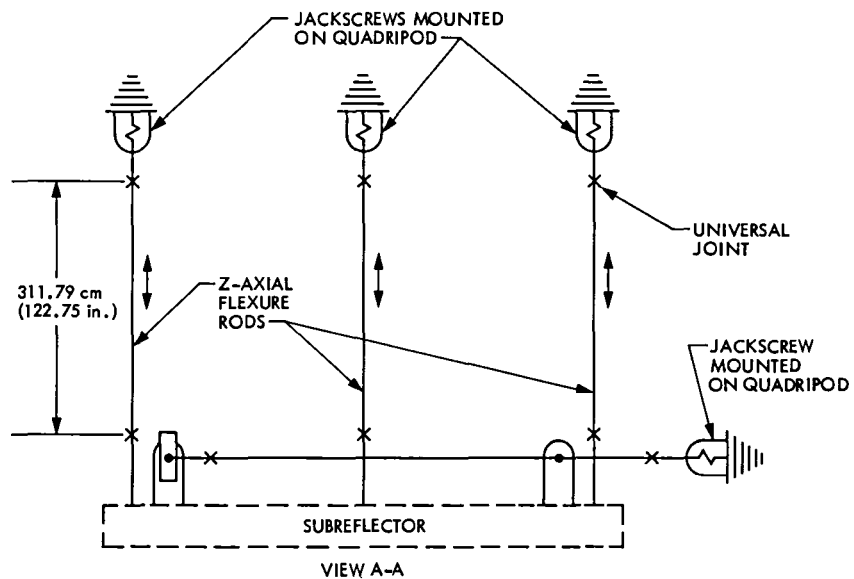
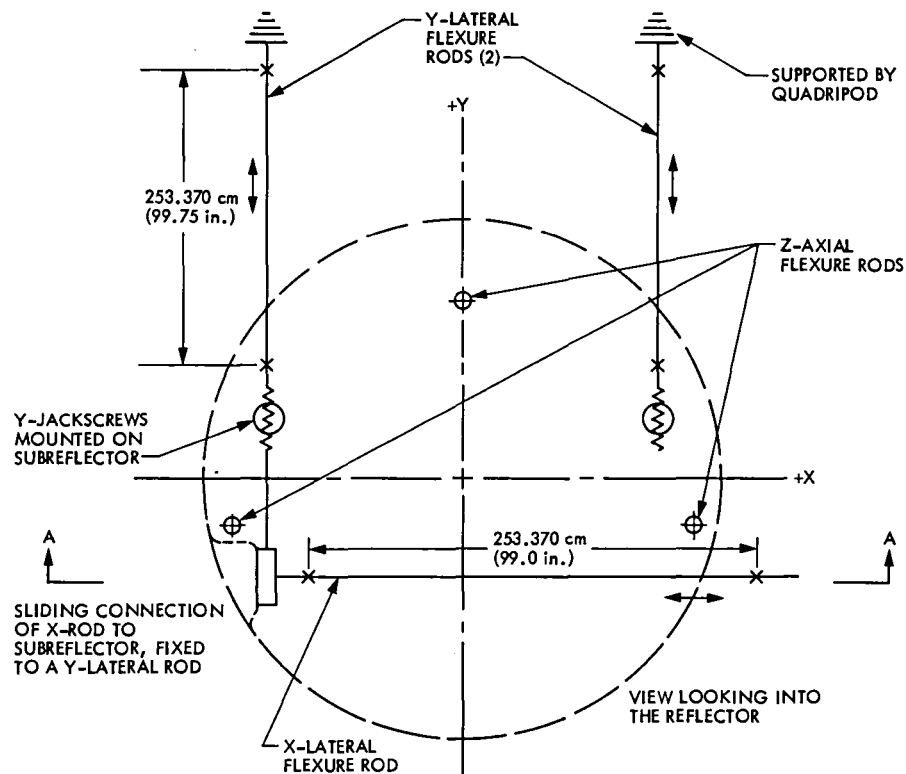


Fig. 1. Flexure supports for the 64-m antenna subreflector

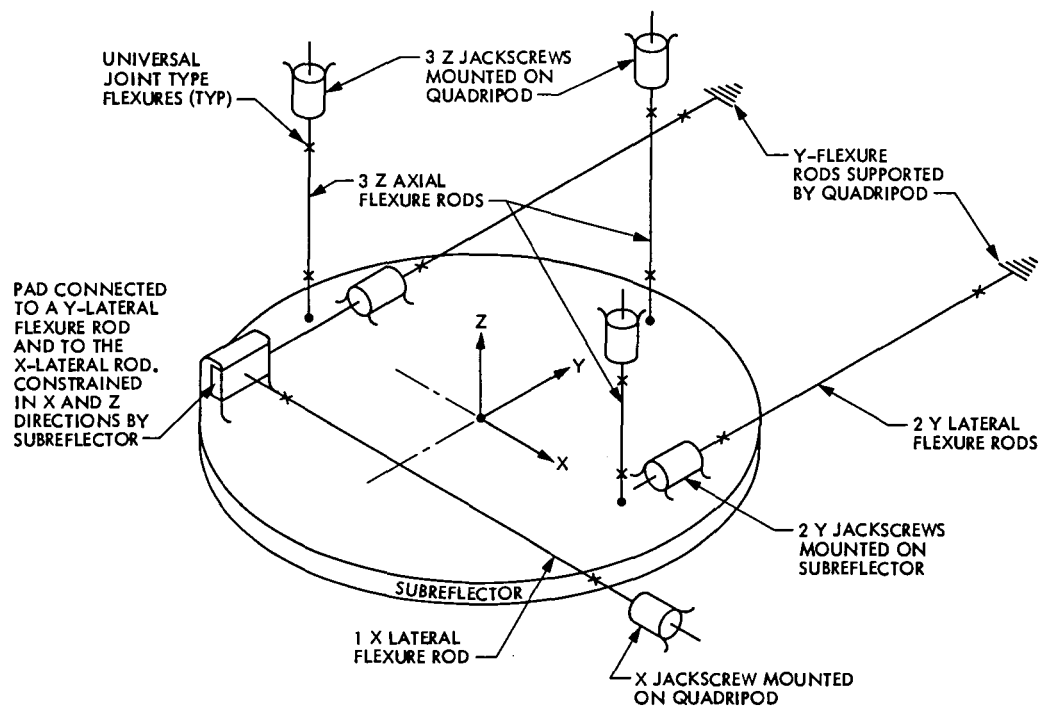


Fig. 2. Isometric view of the 64-m subreflector supports

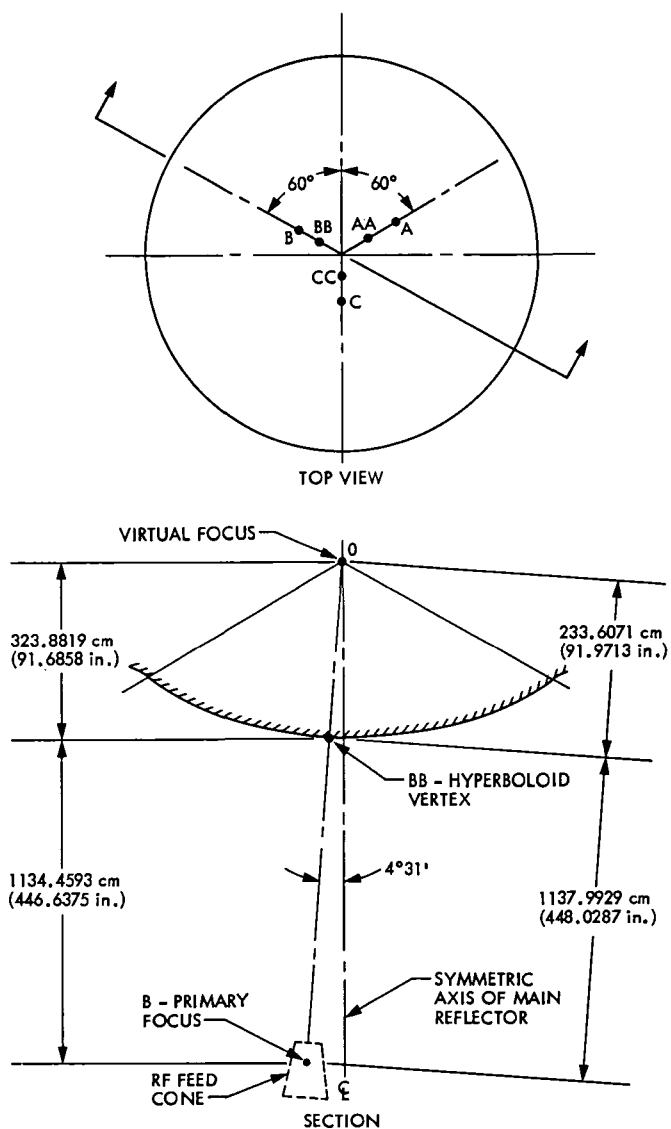


Fig. 3. The 64-m Tricone subreflector geometry

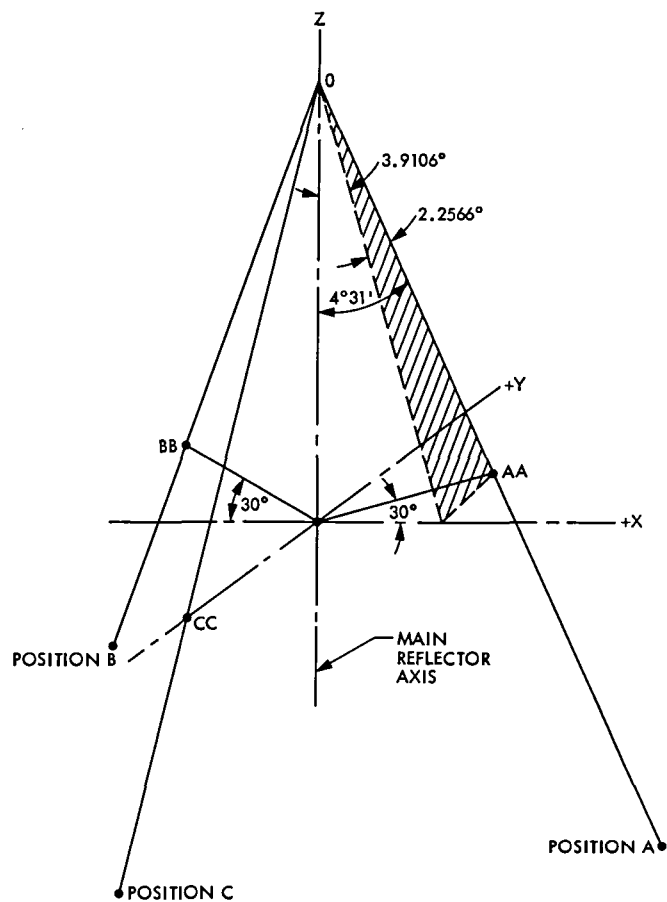


Fig. 4. The 64-m subreflector axis orientation positions

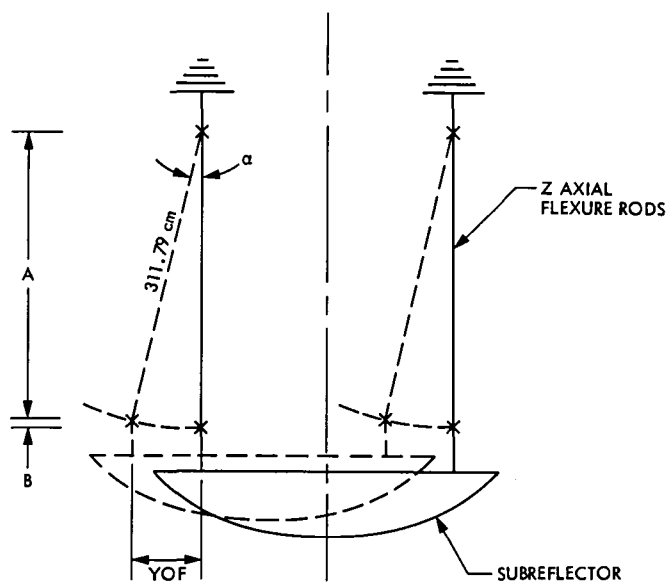


Fig. 5. The 64-m subreflector axial-Z defocus resulting from lateral-Y motion or offset

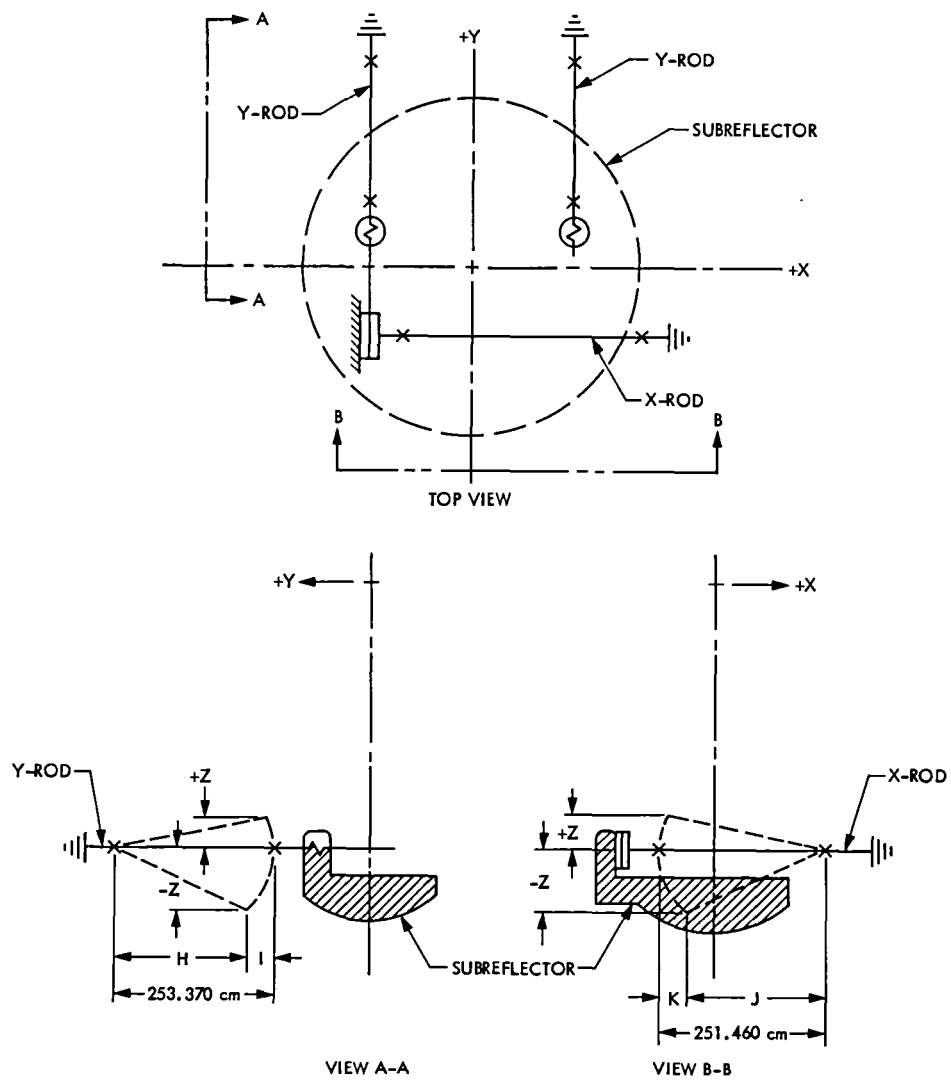


Fig. 6. The 64-m subreflector X and Y Support Rods. The X and Y lateral defocus is cross-coupled from axis-Z motion.

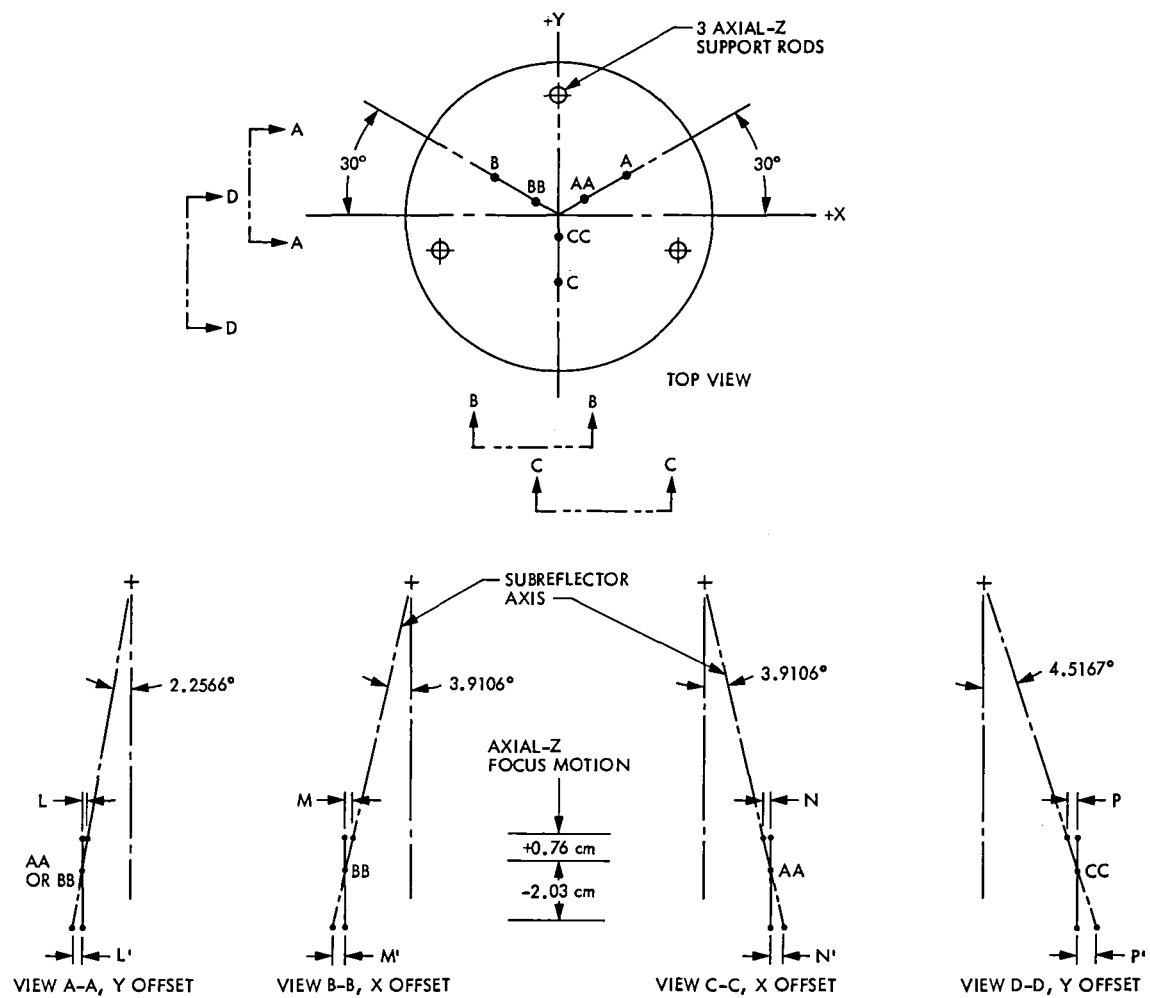
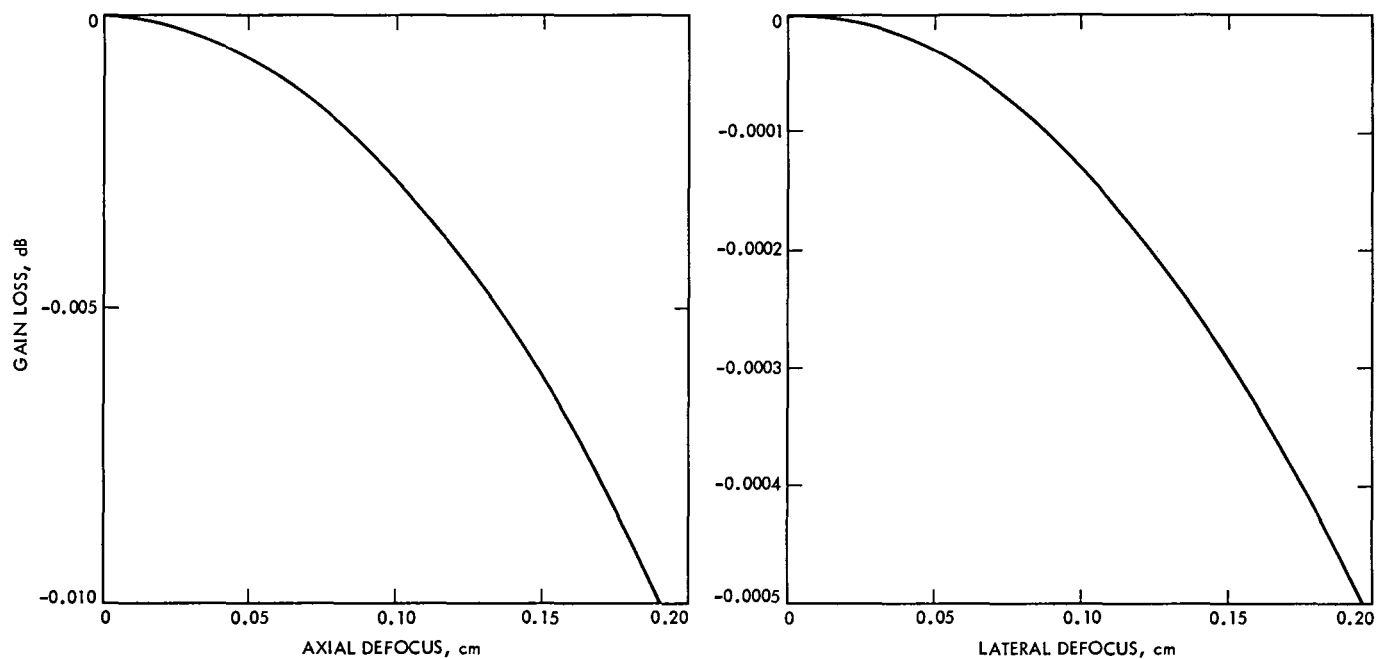


Fig. 7. The 64-m subreflector X and Y lateral offset from axial-Z focus motions



$$\begin{aligned}
 \text{GAIN LOSS, dB (RUZE)} &= 10 \text{ LOG}_{10} \left(\exp \left[-16 \pi^2 \left(\frac{\text{RMS}}{\lambda} \right)^2 \right] \right) \\
 &= 10 \text{ LOG}_{10} \left(\exp \left[-16 \pi^2 \left(\frac{\text{OFFSET-CM} \times \text{SLOPE}}{\lambda - \text{mm}} - \frac{\text{mm}}{\text{cm}} \right)^2 \right] \right) \\
 \left. \begin{aligned}
 \text{AXIAL OFFSET SLOPE} &= 0.7111 \frac{\text{mm}}{\text{cm}} \\
 \text{LATERAL OFFSET SLOPE} &= 0.1553 \frac{\text{mm}}{\text{cm}}
 \end{aligned} \right\} \text{ FOR } F/D = 0.4235
 \end{aligned}$$

Fig. 8. The 64-m antenna ($F/D = 0.4235$) gain loss vs axial and lateral defocus

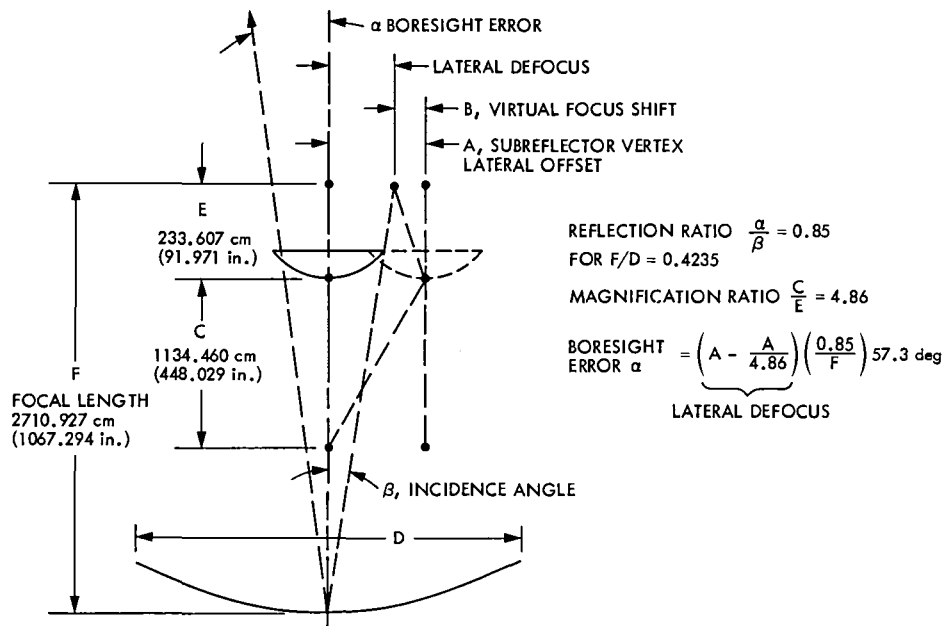


Fig. 9. The 64-m antenna boresight error from subreflector lateral shift

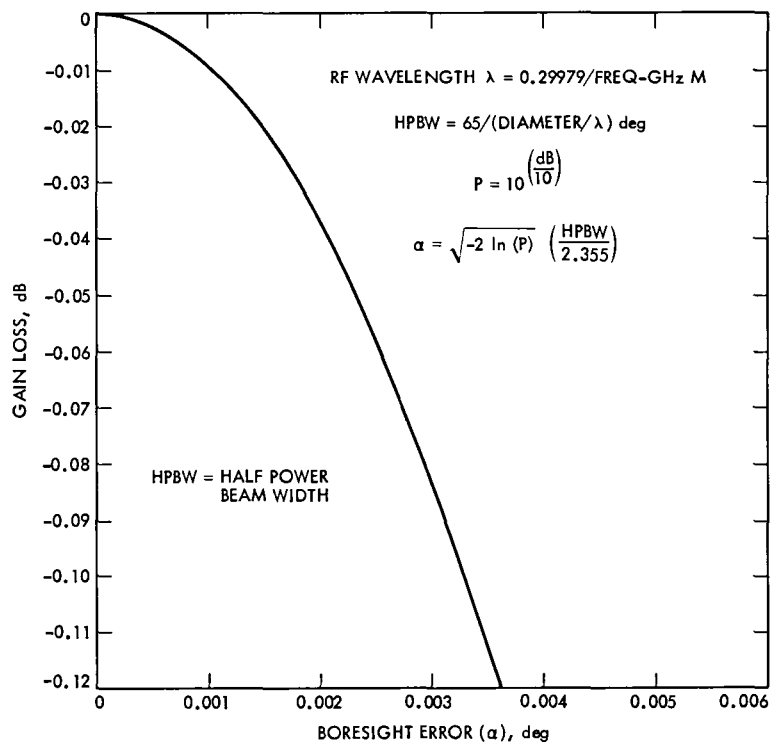


Fig. 10. The 64-m antenna gain loss vs boresight errors. Uniform amplitude illumination assumed.

Seismic Analysis of the Large 70-Meter Antenna, Part II: General Dynamic Response and a Seismic Safety Check

K. Kiedron and C. T. Chian

Ground Antennas and Facilities Engineering Section

An extensive dynamic analysis for the new JPL 70-meter antenna structure is presented. Analytical procedures are based on the normal mode decomposition, which include damping and special forcing functions. The dynamic response can be obtained for any arbitrarily selected point on the structure.

A new computer program for computing the time-dependent, resultant structural displacement, summing the effects of all participating modes, was developed also. Program compatibility with natural frequency analysis output was verified. The program was applied to the JPL 70-meter antenna structure and the dynamic response for several specially selected points was computed.

Seismic analysis of structures, a special application of the general dynamic analysis, is based also on the normal modal decomposition. Strength specification of the antenna, with respect to the earthquake excitation, is done by using the common response spectra. The results indicated basically a safe design under an assumed 5% or more damping coefficient. However, for the antenna located at Goldstone, with more active seismic environment, this study strongly recommends an experimental program that determines the true damping coefficient for a more reliable safety check.

I. Introduction

The development of a general dynamic analysis for an RF antenna which leads to a better understanding of the full range of pointing error sources is important from the point of view of evaluating the antenna performance and preparing it for higher frequency operation. Normal modal decomposition is suitable for an antenna structure represented by a finite ele-

ment model with multi-thousand degrees of freedom. The theoretical background of such an approach is given in Refs. 1, 2, 3, 4. The primary purpose of this study was to get the time history of the dynamic response of the antenna subject to different forcing functions. Another important task was related to the upgrading and rehabilitation of the present JPL 64-/70-meter antenna to ensure that the antenna structure design satisfies the local earthquake safety criteria. Because

seismic analysis is also based on normal modal decomposition, the dynamic analysis was treated more generally and expanded to give the general dynamic response.

The seismicity map of the world shown in Fig. 1 illustrates the strong epicenter locations as obtained from different U.S.A. and foreign seismological stations. In particular, the West Mediterranean area is of interest to the JPL antenna located in Madrid (Spain). Southeastern Australia is important for the antenna located in Canberra, and the southwestern U.S.A. for the antenna located at Goldstone, California. Fig. 1 together with a previous study¹ have indicated that the seismicity of the overseas JPL stations in Spain and Australia do not pose serious problems. However, the Goldstone antenna is subjected occasionally to substantial seismic activity and hence, requires a careful consideration of earthquake-induced loads.

The method of seismic analysis based on normal modal analysis was presented earlier in Ref. 5. However, for most engineering design safety purposes the simplified approach based on the design response spectra is usually adequate. The approach describes the structure earthquake characteristics in a statistical manner, encompassing a number of earthquake motions appropriately selected to the given excitation magnitude and the distance of object from the center of earthquake. The specification of the structure strength can be evaluated on a statistical basis.

II. Governing Equations

The dynamic equations of motion for a multi-degree of freedom system subjected to a ground forcing function can be summarized from Part I of the study (Ref. 5) as:

$$[M] \{\ddot{y}\} + [C] \{\dot{y}\} + [K] \{y\} = -[M] \{a\} \quad (1)$$

where $[M]$, $[C]$, $[K]$ are mass, damping and stiffness matrices, respectively, $\{y\}$ is the displacement vector relative to the ground displacement of modal mass and $\{a\}$ is the horizontal component of ground acceleration.

Equation (1) represents a system of N linear differential equations of second order which are transformed into a set of N uncoupled equations in normal coordinates, $\{z\}$. The mode equation is written as:

$$\ddot{z}_r + 2 \xi_r \omega_r \dot{z}_r + \omega_r^2 z_r = \alpha_r a(t) \quad (2)$$

where ω_r , ξ_r are the natural circular frequency and damping ratio, respectively and α_r is the r th modal participation factor, which determines to what extent each mode participates in the final response. The solution of Eq. (2) is given by:

$$z_r = \frac{\alpha_r}{\omega_r} \int_0^t a(\tau) \exp [-\xi_r \omega_r (t - \tau)] \sin \omega_r (t - \tau) d\tau \quad (3)$$

where τ is the time variable for Duhamel integral.

The resultant relative nodal displacements, after determining z_r are computed by:

$$\{y\} = \sum_{r=1}^N \{y_{or}\} \{z_r\} = [y_o] \{z\} \quad (4)$$

where y_{or} is the r th eigenvector and N is the number of modes. Any numerical integration of Eq. (3) can be employed. When it is desirable to know more precisely what the actual structure dynamic response will be, the complete dynamic analysis of the structure is performed. This requires the knowledge of the natural frequencies ω , the eigenvectors and the weighting or participation factors. The participation factors are obtained from the antenna eigenvalue analysis program.

A simplified approximate approach, adequate for common design purposes can be performed too where the upper bound of the response spectra is estimated. Such approximate analysis, based on ground-motion response spectra is an attractive alternative which avoids a major computation task of numerical integration of the coupled Eq. (1) or their transformation to a system of normal (modal) coordinates. Various earthquake motion responses for a single degree-of-freedom system, represented by Eq. (2), have been evaluated (Ref. 2) and are used in the form of design response spectra, as shown in Fig. 2.

III. Antenna Finite-Element Model

The 70-m antenna-structure components are shown in Fig. 3. For an economical processing on the UNIVAC 1100/81 computer, only half of the structure was modeled. The Y - Z plane is a plane of symmetry and the Z -axis is the focal axis as in Fig. 4. The X -axis is the axis of the reflector elevation rotation. The finite element model of only half of the structure contains about 2000 nodes, 7000 finite element members and 6000 degrees of freedom (Ref. 6). The JPL IDEAS structure

¹"Antenna (210 ft) Facilities in Southeast Australia and Southern Europe," JPL Preliminary Engineering Report, Vol. 3 and 4 (internal document), JPL Contract No. 951281 with Holms and Narver, Inc., Jet Propulsion Laboratory, Pasadena, Calif., August 1965.

analysis computer program was used for performing the eigenvalue analysis. In order to obtain a good convergence between the multi-thousand degree of freedom finite element model of the antenna and its condensed representation to suit the dynamic analysis, two methods were applied: the inverse power (Stodola Method) and the Subspace-Iteration method.

Although the natural frequency (eigenvalue) analysis of the antenna structure does not require the presence of any external loading, the approach employed in this study was to make use of two antenna loading decompositions. These two loading decompositions were made possible in general to include the arbitrarily oriented acceleration vector in a 3-D space.

To illustrate the two loading deflection decompositions further, Figs. 5(a) and 5(b) give a general geometrically symmetrical antenna structure with asymmetric loading. By resolving the loading (forces or moments) as shown in Fig. 5(a) into two decompositions, the loading in decomposition 2 is antisymmetrical (equal but directionally opposite forces). In each loading decomposition pattern, only half the antenna structure is modeled and different boundary conditions are applied in each case which affect the stiffness matrix $[K]$ and may generate different eigenvalues (frequencies). Generally, in order to obtain the whole antenna response, superposition of the two decompositions of the half-antenna models is necessary. Ten modes for each of the two decompositions were obtained as in Table 1 together with participation factors. In our study only loading decomposition 2 was chosen since it resulted in larger participation factors relative to decomposition 1.

It is also chosen because it closely represents a no-external-load situation when the antenna is at zenith position and under gravity loading. Furthermore, a complete eigenvector analysis was made at decomposition 2, and the results are listed in Table 2 for five selected points on the antenna as shown in Fig. 4.

IV. Design Response Spectra

The specification of the strength of 64-meter antenna located at Goldstone, with respect to the appropriate earthquake loading, has been based on the site design response spectra, as shown in Fig. 2. In fact, the same method was proposed by Prof. G. W. Housner of Caltech in 1959, recommended by U.S. Atomic Commission in 1963, and was employed in checking the 64-m antenna in 1964. The new 70-m antenna structure is characterized by twenty, closely spaced, natural frequencies, which are computed from structure programs and listed in Table 1. For design convenience, three different damping ratios (2, 5 and 10%) were selected and the response spectra for each mode were obtained and

plotted in Fig. 6. The dotted band in Fig. 6 indicates the safety margin given by Prof. Housner circa 1960 for the 64-m antenna which ranges between 0.25 g and 0.35 g.

Figure 6 indicates also that viscous damping coefficients play an important role in determining the structure safety. Prior seismic analysis work has not provided the true damping coefficients of the antenna because of air resistance or bearing friction. If the coefficient was to be greater than 5%, then the safety margin previously used circa 1960 is suitable. If the damping coefficient obtained from a future experiment is to be less than 5%, then the safety margin should be increased for a safer design.

V. Results of General Dynamic Analysis

Five points, *A*, *B*, *C*, *D* and *E*, for the whole antenna structure were selected as representative from the dynamic point of view shown in Fig. 4. A FORTRAN program TRANSIENT was developed to perform the operations represented by Eq. (4), which give the time-dependent resultant nodal displacements, compounded from all participating modes. The values of natural frequencies, eigenvectors and participation factors were taken from the JPL IDEAS eigenvalue analysis program. These are listed in Tables 1 and 2. With the El Centro (1940) earthquake excitation (shown in Fig. 7) as a sample forcing input, the results of five time-dependent responses, each for a particular point on the antenna structure, are presented in Figs. 8, through 12, respectively. The 5% damping ratio was taken as a computational parameter.

The maximum displacements were determined at five antenna points for illustration. The comparison of the five peak displacements is shown in Fig. 13. Point *A* (top of the apex) has the largest displacements (11 cm) due to the more flexible quadripod supports compared to the back-up reflector structure. Point *B*, at the rim of the reflector, has a maximum displacement of 2 cm. Point *C* (bottom of the reflector, dish) and Point *D* (elevation drive gear) have similar maximum displacements (8 mm and 7 mm, respectively). Point *E* (at the bottom of the alidade) experiences the smallest peak displacements of 2 mm. The results show that the level of the maximum displacements is not excessive.

VI. Summary

A new dynamic technique for computing the time-dependent multi-mode resultant displacement of a large structure was developed. This new tool, which includes damping and special forcing functions, is important in solving many engineering problems, such as those dealing with structure dynamics under nonsteady wind loading, control system design and pointing

error analyses. The seismic analysis of the antenna structure was performed as a special application of this general dynamic technique and shows, according to the obliging seismic criteria, basically a safe design.

In the past it was recommended that the 64-m antenna structure be designed to withstand an equivalent, constant, lateral acceleration of 25% G for good earthquake resistance.

Those early seismic analyses were based on rough estimates of the fundamental mode frequency. With the first 20 natural frequencies obtained from this mathematical model for the 70-m antenna, more extensive dynamic studies can be performed in general and on seismic responses in particular. Our seismic study results suggest an experimental program that determines the true damping coefficients for a more reliable safety check.

References

1. Clough, R. W., and Penzien, L., *Dynamics of Structures*, New York, McGraw-Hill, 1975.
2. Wiegel, R. L., *Earthquake Engineering*, New York, Prentice-Hall, 1970.
3. Zienkiewicz, O. C., *The Finite Element Method in Engineering Science*, New York, McGraw-Hill, 1977.
4. Levy, S., and Wilkinson, J. P. D., *The Component Element Method in Dynamics with Application to Earthquake and Vehicle Engineering*, New York, McGraw-Hill, 1976.
5. Kiedron, K., and Chian, C. T., Seismic Analysis of the Large 70-Meter Antenna, Part I: Earthquake Response Spectra Versus Full Transient Analysis, *TDA Progress Report 42-81*, pp. 31-42, Jet Propulsion Laboratory, Pasadena, Calif., March 1985.
6. Levy, R., and Strain, D., "Design Optimization of 70M Antenna with Reflective Symmetry," Proc. of 21st Annual Meeting Society of Engineering Science, Inc., Virginia Polytechnic Institute and State University, Blacksburg, VA, Oct. 15-17, 1984.

Table 1. Antenna dynamic characteristics

Mode No.	Circular Frequency (ω), Radian/s	Frequency (f), Hz	Participation Factors (α)
Decomposition 1			
1	9.99	1.59	-0.056
2	16.64	2.65	0.156
3	17.20	2.74	1.213
4	21.15	3.37	-1.410
5	21.29	3.39	1.379
6	22.73	3.62	1.059
7	23.52	3.74	-0.153
8	24.40	3.88	0.281
9	25.37	4.04	0.885
10	26.16	4.16	-0.154
Decomposition 2			
1	7.89	1.26	2.703
2	8.61	1.37	-1.518
3	10.13	1.61	3.280
4	13.39	2.13	-3.732
5	16.68	2.65	0.333
6	18.13	2.88	3.287
7	18.43	2.93	-0.443
8	18.70	2.98	-0.340
9	19.13	3.04	2.301
10	19.60	3.12	0.033

Table 2. Values of eigenvectors for decomposition 2

Mode No.	Node Identification				
	A	B	C	D	E
1	0.693	0.103	0.066	0.058	0.012
2	0.598	-0.020	-0.015	-0.013	-0.003
3	0.460	-0.068	0.018	0.014	-0.004
4	0.395	-0.236	-0.108	-0.082	-0.002
5	0.067	-0.187	0.006	-0.011	0.013
6	-0.096	-0.023	0.023	0.017	0.005
7	-0.104	0.030	-0.005	-0.003	-0.003
8	-0.014	0.043	-0.009	0.000	0.001
9	0.108	-0.117	0.053	0.036	0.013
10	0.071	0.010	0.000	0.000	-0.001

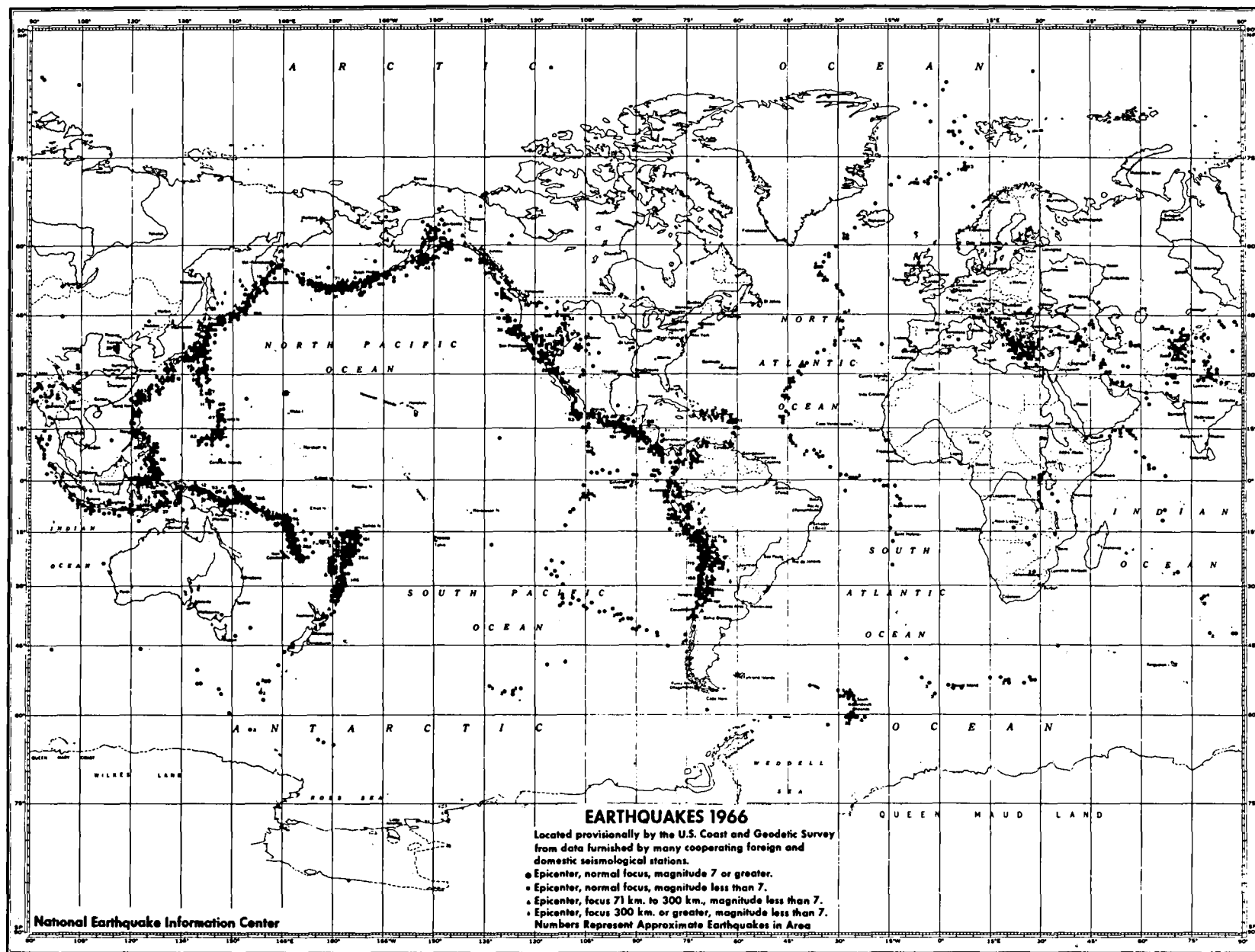


Fig. 1. Seismicity map of the world (from Ref. 2, p. 23)

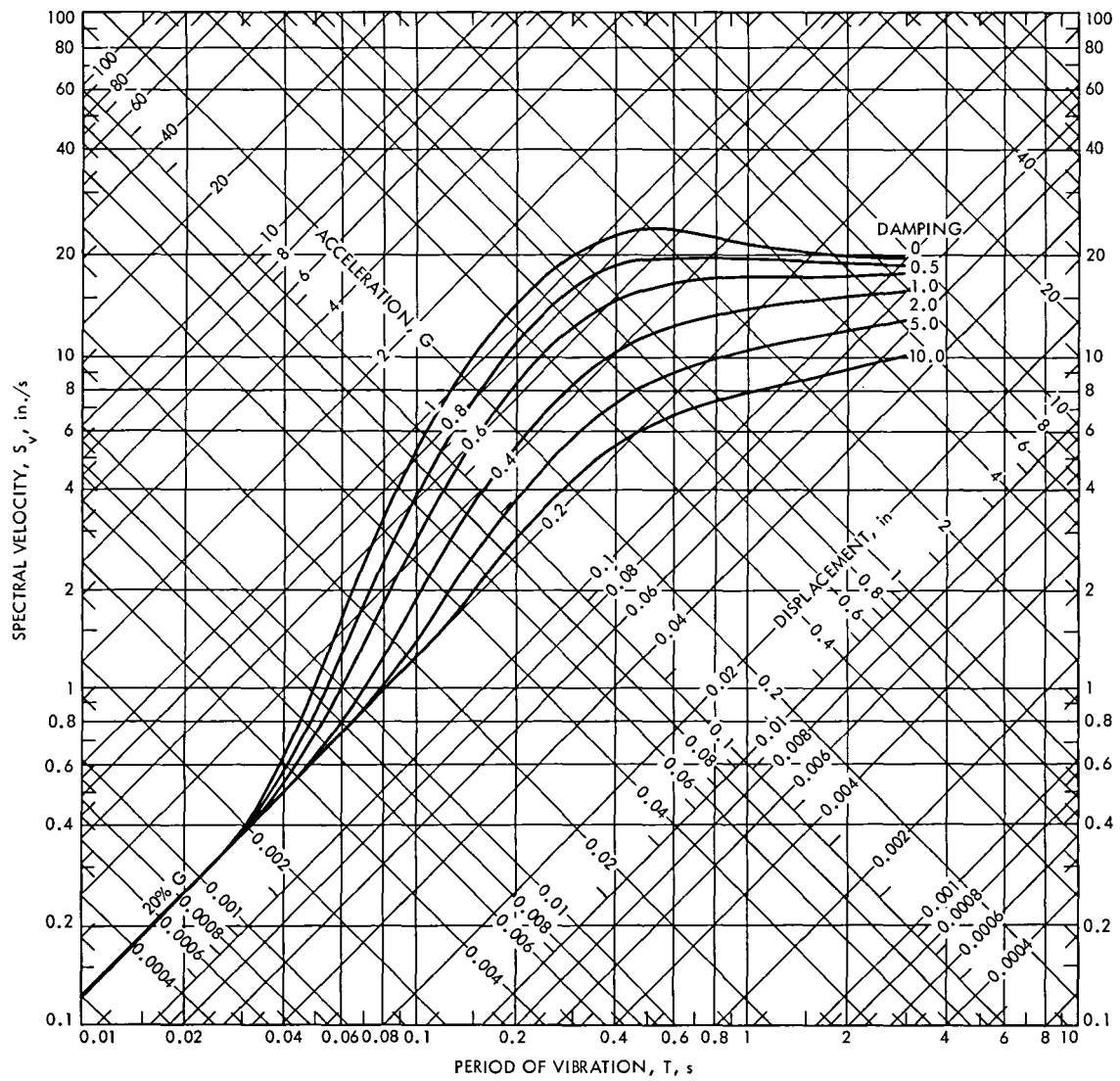


Fig. 2. Design response spectra

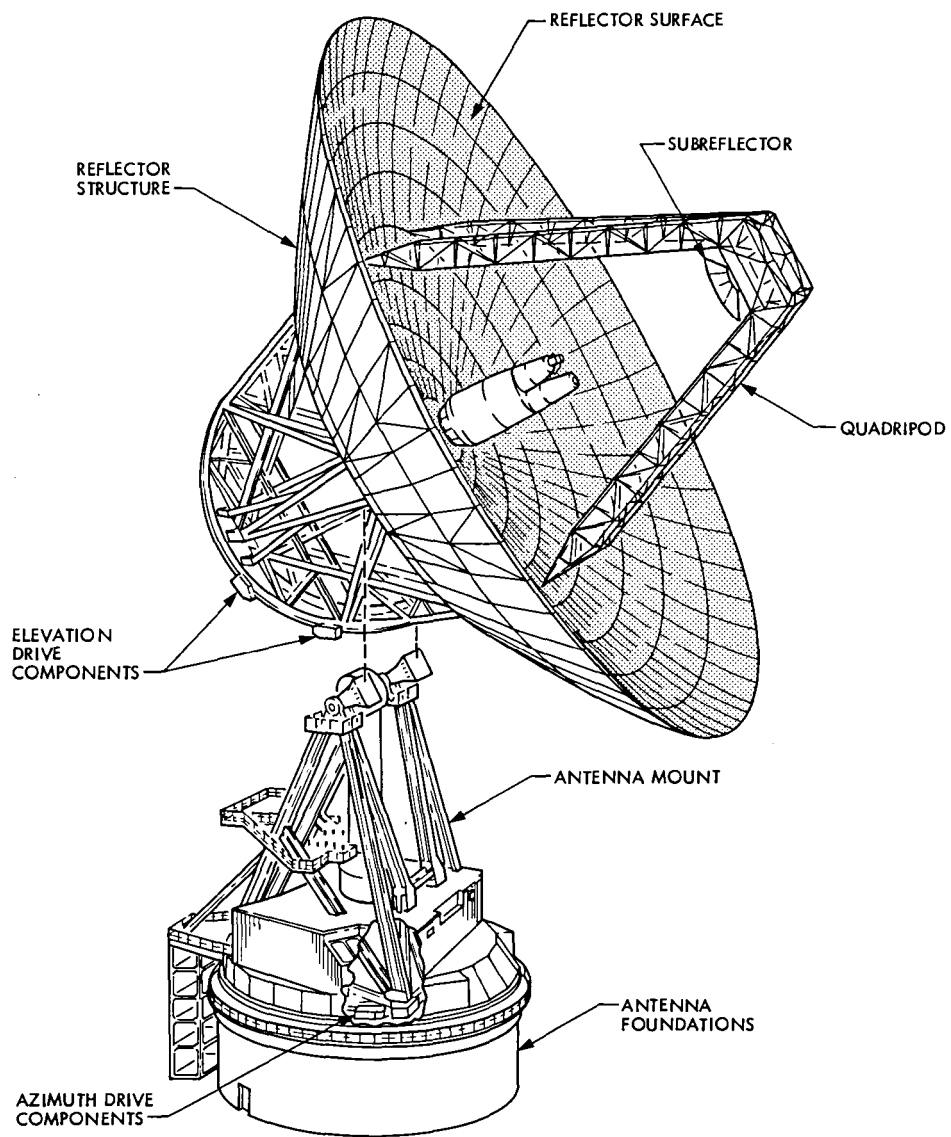


Fig. 3. Antenna components

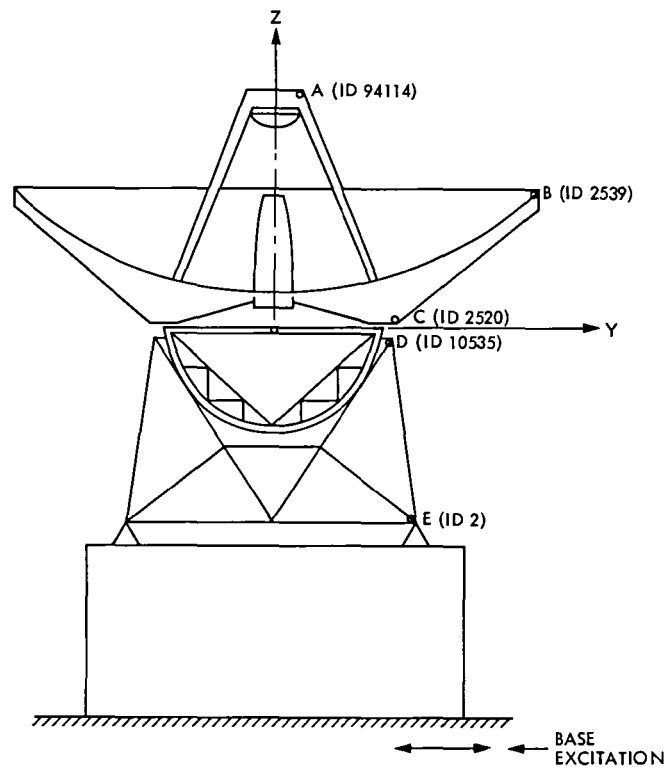


Fig. 4. Locations of selected dynamic response points

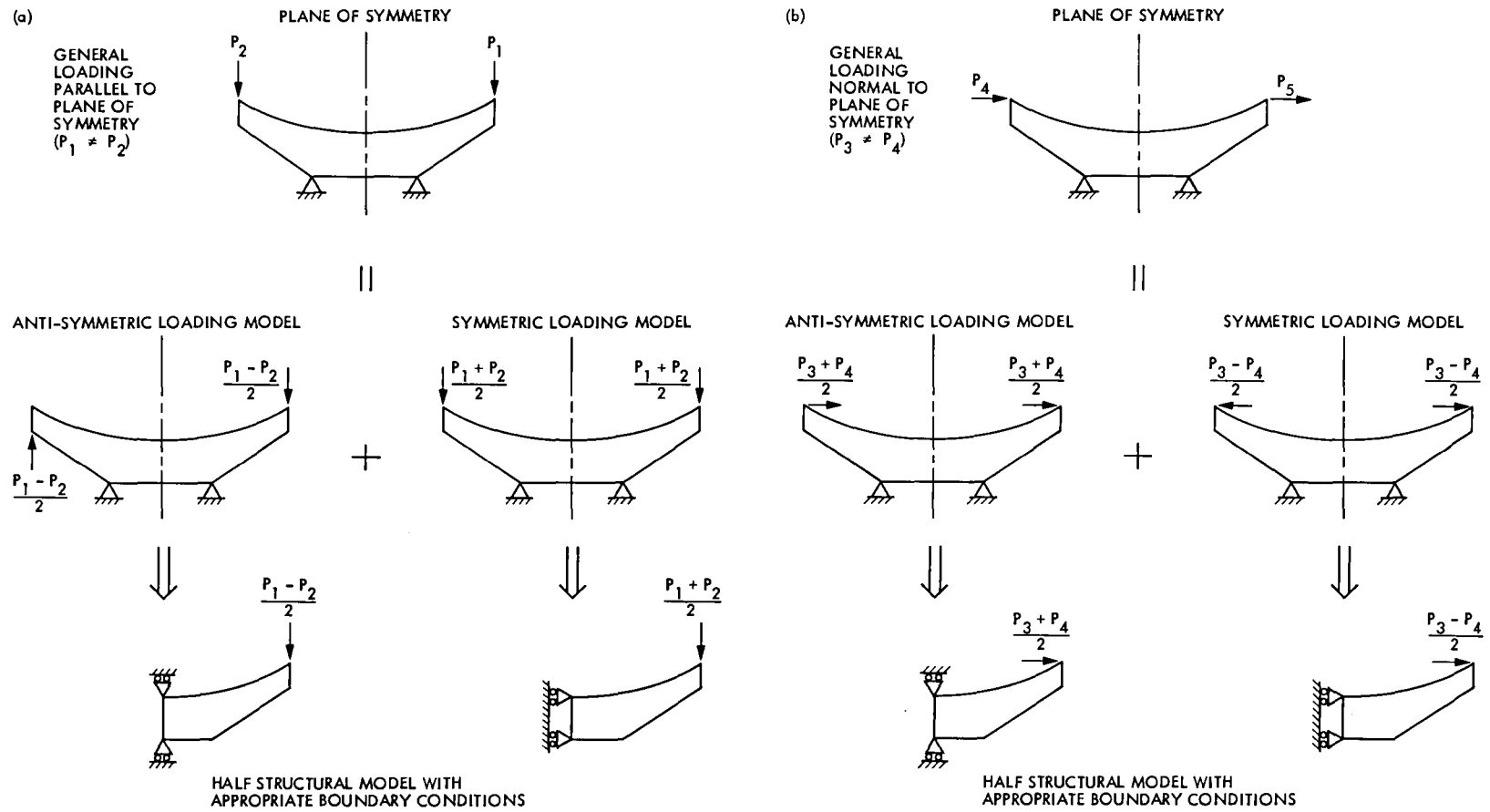


Fig. 5. Load decomposition on a symmetric antenna

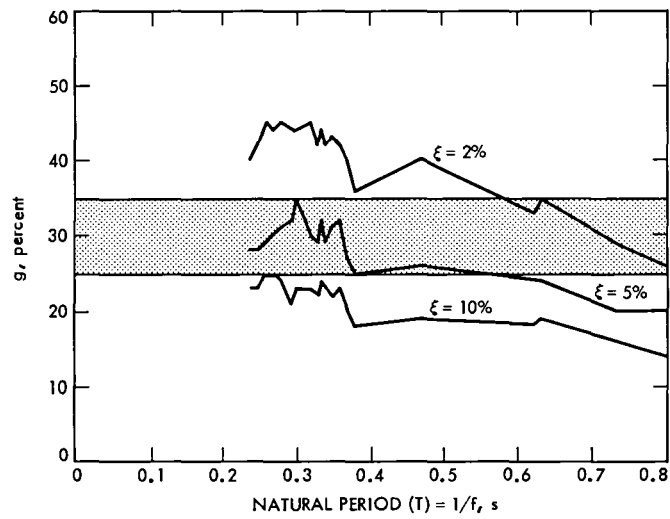


Fig. 6. Antenna response spectra

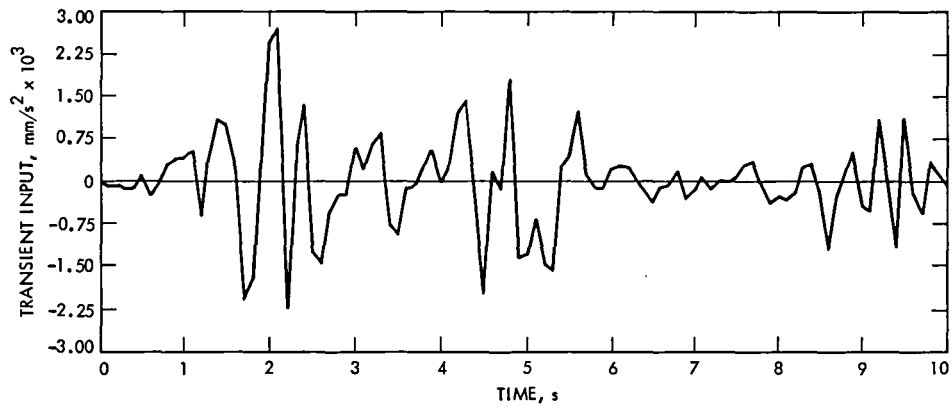


Fig. 7. Dynamic El Centro input

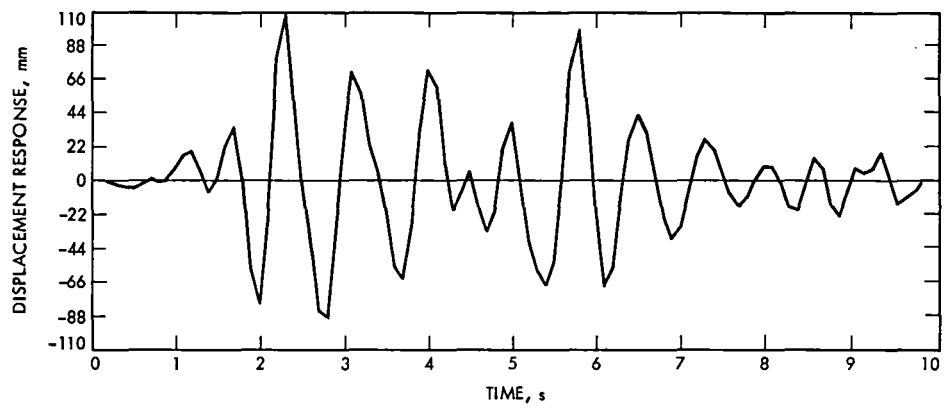


Fig. 8. Dynamic response for point A

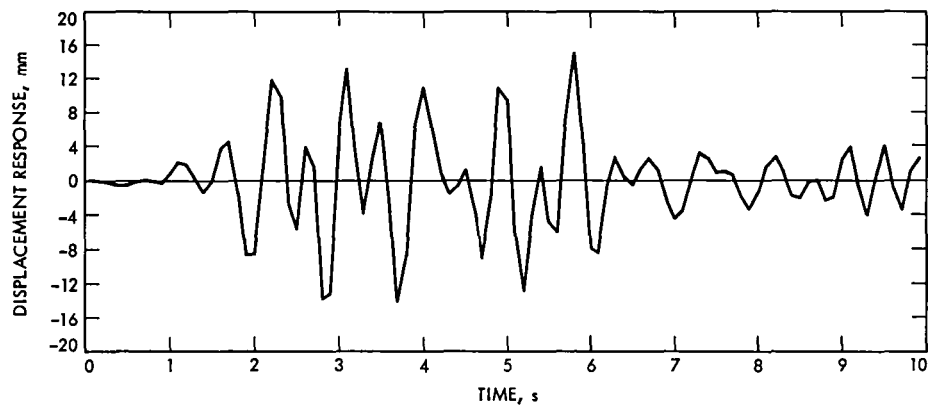


Fig. 9. Dynamic response for point B

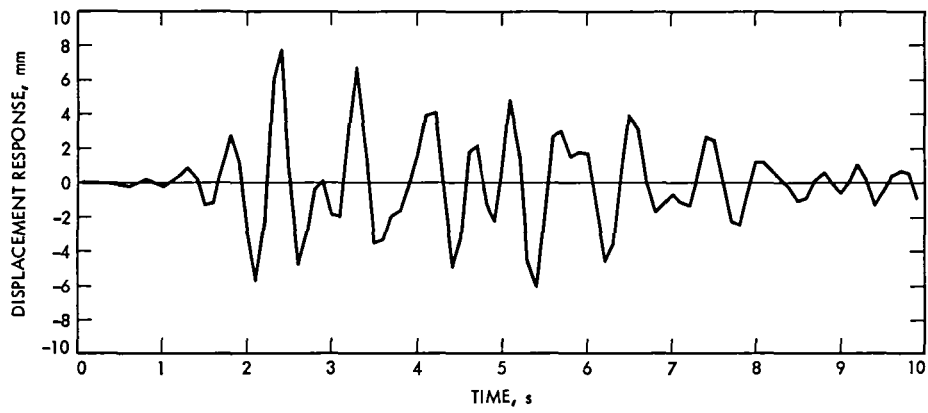


Fig. 10. Dynamic response for point C

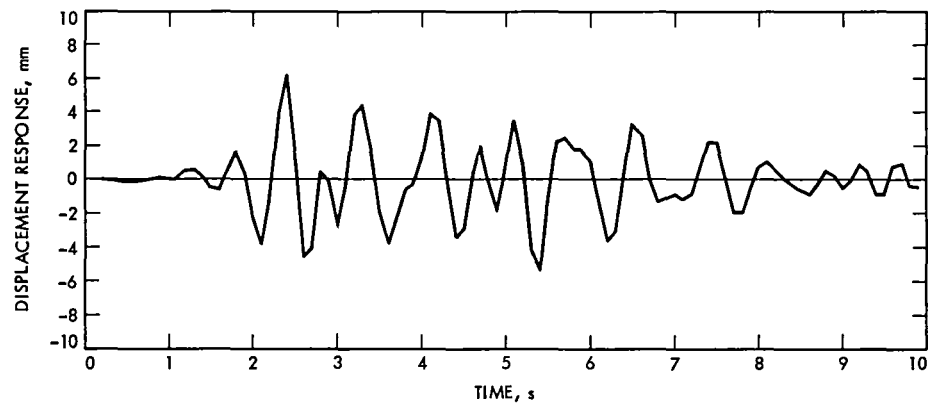


Fig. 11. Dynamic response for point D

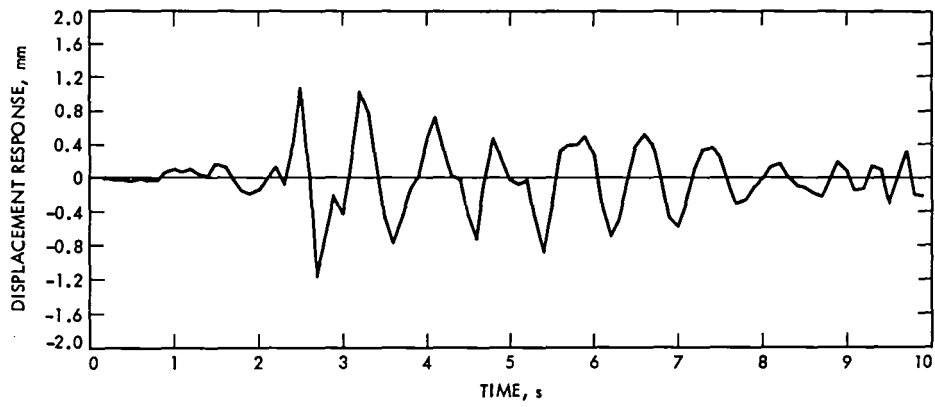


Fig. 12. Dynamic response for point E

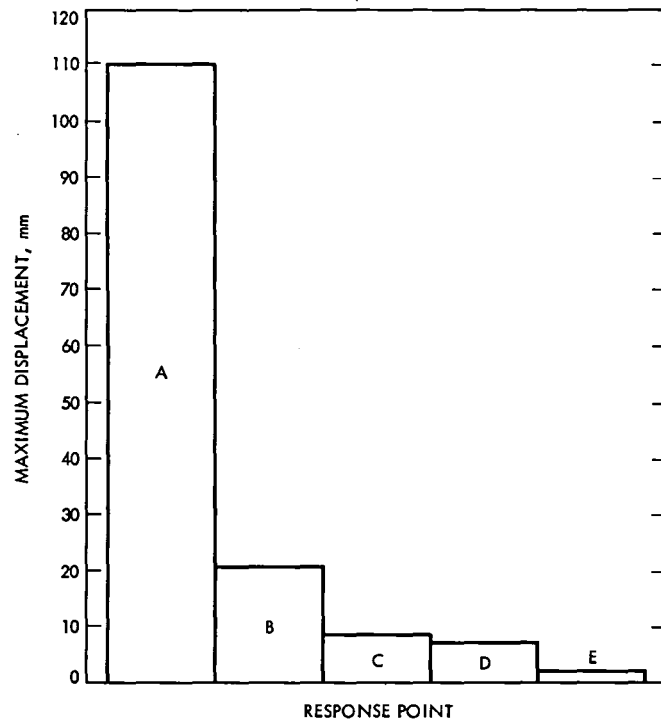


Fig. 13. Maximum displacements at the five selected antenna points

In Search of a 2-dB Coding Gain

J. H. Yuen and Q. D. Vo

Communications Systems Research Section

A recent code search found a (15, 1/5), a (14, 1/6), and a (15, 1/6) convolutional code which, when concatenated with a 10-bit (1023, 959) Reed-Solomon (RS) code, achieves a bit-error rate (BER) of 10^{-6} at a bit signal-to-noise ratio (SNR) of 0.50 dB, 0.47 dB and 0.42 dB, respectively. All of these three codes outperform the Voyager communication system, our baseline, which achieves a BER of 10^{-6} at bit SNR of 2.53 dB, by more than 2 dB. Our 2 dB coding improvement goal has been exceeded.

I. Introduction

To enable future planetary missions to provide today's equivalent or better information return, higher communication efficiencies have to be achieved. More efficient communication links will help to lower mission cost.

To achieve reliable communications over a noisy channel with acceptable coding complexity, concatenated coding systems with a convolutional code as the inner code and a Reed-Solomon (RS) code as the outer code are used. NASA's Voyager employs a communication system which uses a (7, 1/2) convolutional code as its inner code and an 8-bit (255, 223) RS code as its outer code. This system, which achieves a bit-error rate (BER) of 10^{-6} at a bit signal-to-noise ratio (SNR) of 2.53 dB, is the proposed NASA standard and our baseline. Communication system performance can be improved by many means, including increasing the spacecraft transmitter power, the antenna size, and the coding complexity.

Spacecraft transmitter power is at a premium; larger antennas are also very costly. Fortunately, with the continued progress in VLSI technology, it is possible to implement codes with much higher complexity than our baseline system. Our

goal in this search was to find codes which will provide a 2 dB improvement over the baseline system, i.e., the new code needs to achieve a BER of 10^{-6} at a bit SNR of 0.53 dB. This 2 dB gain of bit SNR by coding improvement is likely to be a much more cost effective means than increasing spacecraft transmitter power, antenna size, etc.

II. Code Search Results

Instead of using the criterion of maximum free distance, d_f , we searched for good convolutional codes using the criterion of minimizing required bit SNR, E_b/N_0 , for a given value of desired BER, for the goodness of code. The channel considered is the deep space channel which is the additive white Gaussian noise channel and binary-phase-shift-keying (BPSK) modulation (Ref. 1). Partial results for good convolutional codes of constraint length $K < 13$, and rate $1/N$, $2 < N < 8$, have been reported (Refs. 2 and 3). For $(K, 1/N)$ convolutional codes, the number of all possible codes is 2^{KN} , which is extremely large for large K and N . It is prohibitively time consuming to perform an exhaustive search except for small K and N . Using educated guesses combined with the idea that good codes generate good codes (Ref. 2), we performed selective searches for good low rate $1/N$ convolutional codes of

constraint length $K = 13, 14$, and 15 . The codes found were first measured by using the free distance and then the transfer function bound (Ref. 3), which is a fast, efficient algorithm developed for our code search effort. Relatively good code resulted. The symbol-error probabilities were determined by computer simulations using 4-bit channel output quantization, and the performance of these relatively good convolutional codes when concatenated with various RS codes was determined.

The convolutional code generators (in Octal) for several relatively good codes of rate $1/4, 1/5$, and $1/6$ are listed in Table 1, 2, and 3, respectively. The minimum free distances of these codes are also listed. The 10-bit symbol error probabilities, obtained by long computer simulations, are plotted in Fig. 1 for some of these codes (codes marked with an asterisk (*) in Tables 1, 2, and 3). These convolutional codes are concatenated with various 10-bit RS codes. The required E_b/N_0 (i.e., outer code SNR) to achieve a BER of 10^{-6} is determined (Ref. 5) and shown in Table 4 for concatenated systems with optimal RS code rates.

Similarly, performance results have also been obtained when these convolutional codes are concatenated with various 8-bit RS codes, including the (255, 223) RS code of our baseline system. Results are included in Table 4. It is found that the performance of the 8-bit RS code is only about 0.25 dB less than the 10-bit RS code.

We have found three convolutional codes which, when concatenated with the 10-bit (1023, 959) RS code, exceed our 2 dB coding improvement goal. These are the (15, 1/5), (14, 1/6), and (15, 1/6) convolutional codes whose code generators are given in Tables 2 and 3. The performance of these codes is also illustrated in Fig. 2 along with some historically significant codes used for deep space communications.

Our 2 dB coding gain in SNR comes from increased decoder complexity and expanded bandwidth. Since implementation complexity may be an issue, let us examine the complexity of these codes. The encoders are relatively simple. For example, the best convolutional code (15, 1/6) that we have found is only slightly more complicated than the current Voyager baseline (7, 1/2) convolutional code as depicted in Fig. 3. They are indeed simple logic circuits. An assessment of the increased decoder complexity and expanded bandwidth of the two most powerful codes that we have found is summarized in Table 5 by comparing them to our baseline system. Replacing the 8-bit RS code in our baseline system by the 10-bit RS code contributes only about 0.25 dB in coding gain, but the increase in decoder complexity is rather large. Since most of the coding gain comes from the long constraint length, low rate convolutional codes, we should concentrate on their implementation. We feel that the realization of our newly found good convolutional codes is within reach by VLSI technology.

III. Conclusion

Our selected search of the astronomical number of possible convolutional codes of constraint length $K = 13, 14$, and 15 , and rate $1/2$ to $1/6$ was guided by educated guesses and the idea that good codes generate good codes. We used the practical criterion of minimizing the required bit SNR for a given BER for the goodness of code. By using a specially developed fast efficient transfer function bound algorithm and computer simulation programs, we obtained many good convolutional codes. In particular, we found a (15, 1/5), a (14, 1/6), and a (15, 1/6) convolutional code. Each, when concatenated with a 10 bit (1023, 959) RS code, outperforms our baseline system by 2.03 dB, 2.06 dB, and 2.11 dB respectively. They exceed the 2 dB coding improvement goal. Considerations have been given to the decoding complexity. We feel that VLSI technology will enable the realization of our newly found convolutional codes.

References

1. Yuen, J. H., Ed., *Deep Space Telecommunications Systems Engineering*, Plenum, N.Y., October 1983.
2. Lee, P. J., "New Short Constraint Length, Rate $1/N$ Convolutional Codes Which Minimize Required SNR for Given Bit Error Rate," *TDA Progress Report 42-77*, pp. 41-56, Jet Propulsion Laboratory, Pasadena, CA., May 15, 1984; also in *IEEE Transactions on Communications*, Volume COM-33, No. 2, pp. 171-177, February 1985.
3. Lee, P. J., "Further Results on Rate $1/N$ Convolutional Code Constructions with Minimum Required SNR Criterion," *TDA Progress Report 42-80*, pp. 97-102, Jet Propulsion Laboratory, Pasadena, CA., February 15, 1985; also to appear in *IEEE Transactions on Communications*, November 1985.
4. Larsen, K. J., "Short Convolutional Codes with Maximal Free Distance for Rates $1/2$, $1/3$, and $1/4$," *IEEE Transactions on Information Theory*, Vol. IT-19, pp. 371-372, May 1973.
5. Vo, Q. D., "Simulations for Full Unit Memory and Partial Unit Memory Convolutional Codes with Real-Time Minimal-Byte-Error Probability Decoding Algorithm," *TDA Progress Report 42-76*, pp. 77-81, Jet Propulsion Laboratory, Pasadena, California, February 15, 1984.

Table 1. Rate 1/4 convolutional codes

K	Generators (Octal)				d_{free}
13	11145	12477	15573	16727	33* [†]
14	20553	25271	33447	37515	34
	21113	23175	35557	36527	36
15	46321	51271	63667	73257	37
	46321	51271	63667	73277	38

[†] This code was found by Larsen (Ref. 4).

Table 2. Rate 1/5 convolutional codes

K	Generators (Octal)					d_{free}
13	10661	11145	12477	15573	16727	41*
	10671	11145	12477	15573	16727	42
14	21113	23175	27621	35557	36527	44*
15	46321	51271	63667	70535	73277	47
	46321	51271	63667	70565	73277	47*

Table 3. Rate 1/6 convolutional codes

K	Generators (Octal)						d_{free}
14	21113	23175	27621	33465	35557	36527	53*
	21113	23175	27631	33465	35557	36527	54
15	46321	51271	70535	63667	73277	76513	56*

Table 4. Required E_b/N_0 to achieve a BER of 10^{-6}

Inner Convolutional Codes	Outer RS Codes	Required E_b/N_0 for $P_b = 10^{-6}$, dB	Improvement Over Baseline System, dB
(13, 1/4)	(1023, 895)	0.84	1.69
(13, 1/5)	(1023, 927)	0.68	1.85
(14, 1/5)	(1023, 927)	0.57	1.96
(15, 1/5)	(1023, 959)	0.50	2.03
(14, 1/6)	(1023, 959)	0.47	2.06
(15, 1/6)	(1023, 959)	0.42	2.11
(13, 1/5)	(255, 223)	0.93	1.60
	(255, 229)	0.91	1.62
(14, 1/5)	(255, 223)	0.82	1.71
	(255, 231)	0.79	1.74
(14, 1/6)	(255, 223)	0.71	1.82
	(255, 233)	0.63	1.90

Table 5. Decoder complexity considerations

Performance Parameter	Voyager's Code (7, 1/2) + (1023, 959) RS	New Code No. 1		New Code No. 2		New Code No. 3	
		(15, 1/5) + (1023, 959) RS	Adv./Disadv. Relative to Voyager's Code	(14, 1/6) + (1023, 959) RS	Adv./Disadv. Relative to Voyager's Code	(15, 1/6) + (1023, 959) RS	Adv./Disadv. Relative to Voyager's Code
SNR Required for BER = 10^{-6}	2.53 dB	0.50 dB	2.03 dB Improvement	0.47 dB	2.06 dB Improvement	0.42 dB	2.11 dB Improvement
Inner Code							
Memory	64 States	16384 States	256 Times	8192 States	128 Times	16384 States	256 Times
Computations	1 (Normalized)	640 (Normalized)	640 Times	384 (Normalized)	384 Times	768 (Normalized)	768 Times
Outer Code							
Memory	255 × 8	1023 × 10	5 Times	1023 × 10	5 Times	1023 × 10	5 Times
Computations	1 (Normalized)	20 (Normalized)	20 Times	20 (Normalized)	20 Times	20 (Normalized)	20 Times
Overall Bandwidth Expansion	2.2	5.3	2.5 Times	6.4	2.9 Times	6.4	2.9 Times

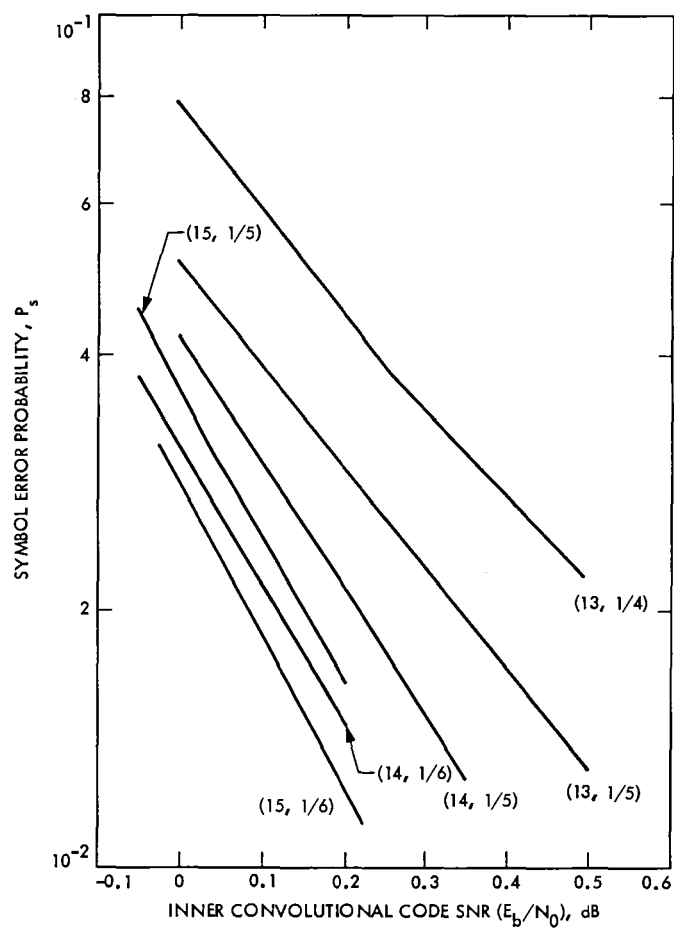
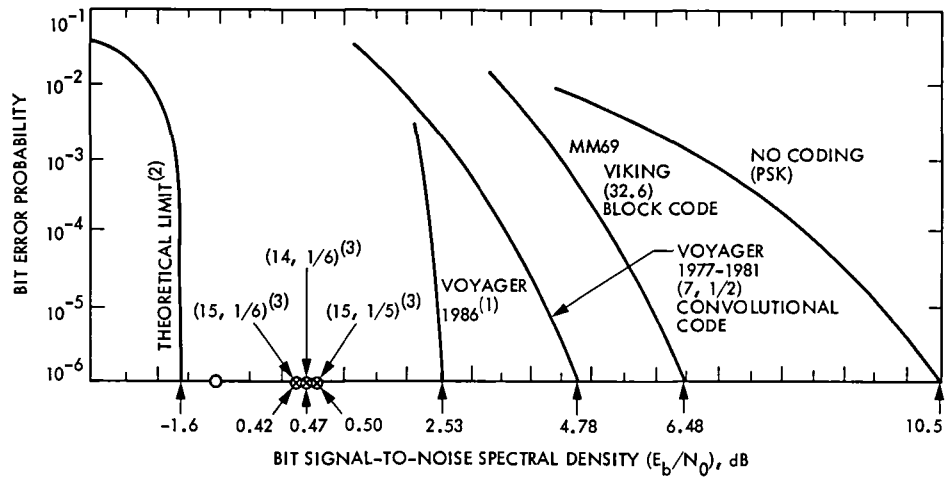


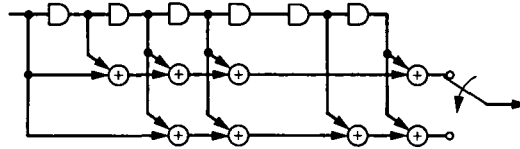
Fig. 1. Simulated 10-bit symbol error probability



- (1) (7, 1/2) CONVOLUTIONAL CODE (VITERBI DECODING) CONCATENATED WITH A (255, 223) REED-SOLOMON OUTER CODE
- (2) INFINITE BANDWIDTH EXPANSION
- (3) CONCATENATED WITH A (1023, 959) REED-SOLOMON CODE
- O THEORETICAL LIMIT FOR RATE 1/5 CODES IS -1.01 dB AND FOR RATE 1/6 CODES IS -1.10 dB

Fig. 2. Deep space telemetry performance

PRESENTLY USED (7, 1/2) CONVOLUTIONAL ENCODER ON VOYAGER



NEWLY FOUND (15, 1/6) CONVOLUTIONAL ENCODER

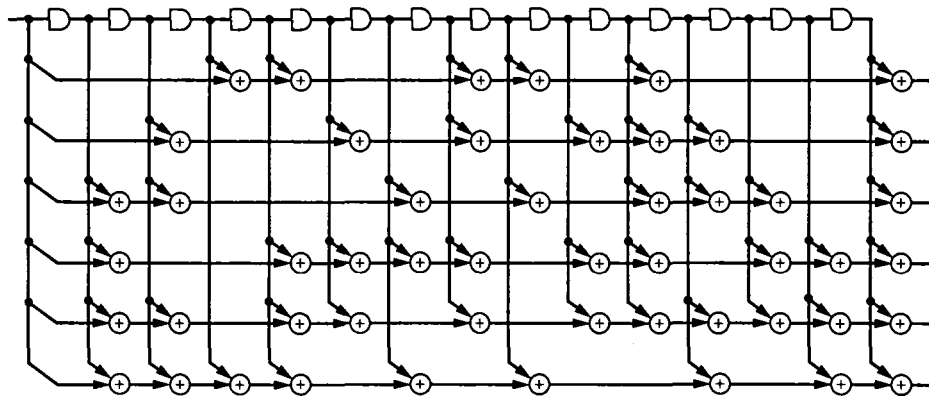


Fig. 3. Encoder complexity considerations

Decoding Convolutionally Encoded Images

G. H. Pitt, III, and L. Swanson
Communications Systems Research Section

Maximum Likelihood Convolutional Decoding, which is used by the Deep Space Network for short constraint-length convolutional codes, assumes that all strings of information bits are equally likely. In some cases, like image data, this is not the case. We examine the use of information about an adjacent pixel in decoding convolutionally encoded Voyager images, and discover that, in a region of interest, as much as 2 dB may be gained.

From the standpoint of digital data, the coding system for Voyager (and an international coding standard) consists of a Reed-Solomon encoder (optional), a convolutional encoder, a noisy channel, a maximum likelihood convolutional decoder (Viterbi decoder), and, if necessary, a Reed-Solomon decoder (see Fig. 1). With bit signal-to-noise ratio (SNR) as low as 2.3 dB, the concatenated scheme produces a bit error rate of 10^{-5} , and the convolutionally coded only scheme produces a bit error rate of 3×10^{-3} , while an uncoded scheme produces a bit error rate of 3×10^{-2} at this signal-to-noise ratio.

The Viterbi decoder finds the codeword which is closest to the received string. This is called maximum likelihood decoding because, under the assumption that all codewords are *a priori* equally likely to be transmitted, this decoding scheme retrieves the most likely sent codeword. In some cases, though, codewords are not all equally likely to be transmitted. In Voyager images, for example, pixel to pixel variations are not completely random: They are much more likely to be small than large. In this case, a decoder which makes use of the source statistics should perform better than a Viterbi decoder. (Image compression uses these statistics to lower the transmission rate and thus raise symbol SNR, but some Voyager

images are sent uncompressed because of spacecraft limitations; also, an alternative would be valuable in the unlikely event of a data compressor failure before Neptune encounter in 1989.)

In Ref. 1, Korwar investigates the use of source statistics in convolutional decoding hard-quantized images, i.e., images in which each pixel is represented by one bit. Using hypothetical data, she found that using source statistics makes substantial improvement in decoder performance. Our work is different in two ways: We use 8-bit quantized pixels and real Voyager imaging data. Using 8-bit quantized pixels requires a byte-oriented rather than a bit-oriented decoder, which is more complicated and runs much more slowly.

We obtained a tape of image data from Voyager project. It consisted of several pictures, each 800 by 800 8-bit quantized pixels. A C-language program was written on a VAX running UNIX to extract the frequency of various absolute differences in value between adjacent pixels. That is, thinking of each pixel as an integer X between 0 and 255, the values of $|X_i - X_{i-1}|$ were tabulated. We were not surprised to discover that small values for this difference were much more common

than would be expected in random (independent) data (see Fig. 3).

With the frequencies of $|X_i - X_{i-1}|$ obtained from one picture, a new decoder based on these statistics was written, again using the UNIX VAX (see Appendix A for the algorithm). Another picture was then convolutionally encoded, Gaussian noise was added, and the picture was decoded with the new decoder. The performance of this decoder on this picture is shown in Figs. 4 and 5. These figures show that performance improves substantially for signal-to-noise ratios below 2.0 dB. This is a region of interest for images which have been Reed-Solomon encoded. (Byte error rates are shown in Fig. 5 because this is of interest for data which is Reed-Solomon encoded.)

The curves show that our decoder does not perform as well as a conventional Viterbi decoder for signal-to-noise ratios much above 2.0 dB. At least part of this can be attributed to the edges of pictures, which our tape shows as dark on one edge and white on the other. These strips may be the result of

data compression, in which case they would not occur in images being sent uncompressed (the only kind of images for which the new decoder is useful). Our lower performance may also be partly a result of "rizzomarks" which are added for calibration and whose effects could be discounted in a scheme like ours.

We have not yet examined other possible schemes, such as considering both the pixel to the left of the current pixel (which the new decoder uses) and the pixel above it. Such a slightly more complicated scheme may slightly improve performance at the cost of slightly slower decoding. Using a software decoder on a VAX 750, decoding is very slow (640,000 bytes/month); if our scheme were to be used for real transmitted data, we would need big gains from hardware and algorithm improvements.

During Voyager's Uranus encounter, some undecoded channel symbols may be saved for the symbol stream combining experiment. We hope to use these to test our decoder under "real life" conditions.

Reference

1. Korwar, V., Viterbi Decoding Modified for Sources with Memory, *DSN Progress Report 42-55*, pp. 97-110, Jet Propulsion Laboratory, Pasadena, Calif., November and December 1979.

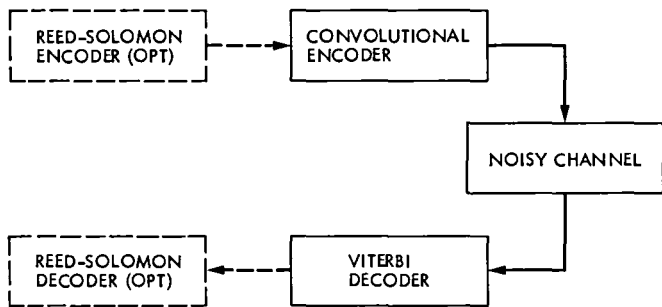


Fig. 1. Digital data coding scheme

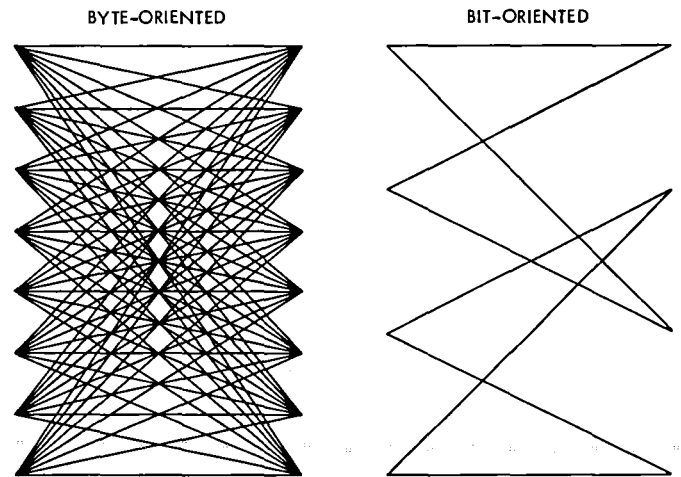


Fig. 2. An illustration of the difference in complexity of a byte-oriented decoder vs a bit-oriented decoder for 3-bit bytes and a constraint length 3 code

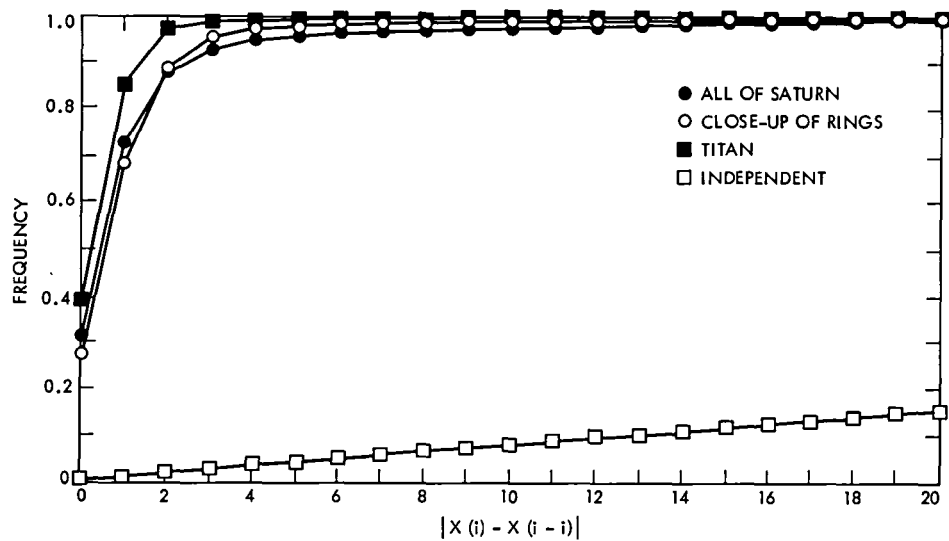


Fig. 3. A graph of the observed frequency of different values of $|X_i - X_{i-1}|$ for several Voyager images compared with the predicted frequency assuming independence (randomness) between adjacent pixels

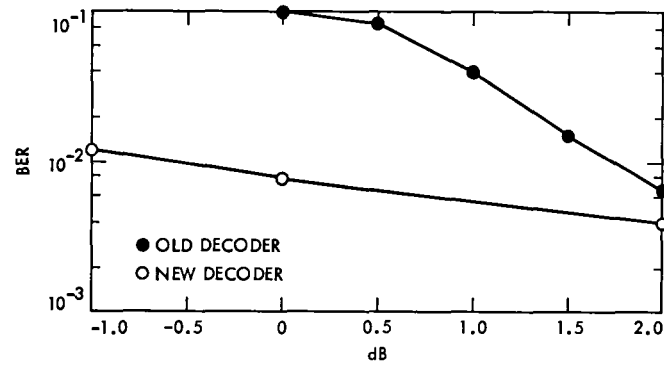


Fig. 4. A graph of bit error rates for the conventional (old) and the modified (new) maximum likelihood decoder

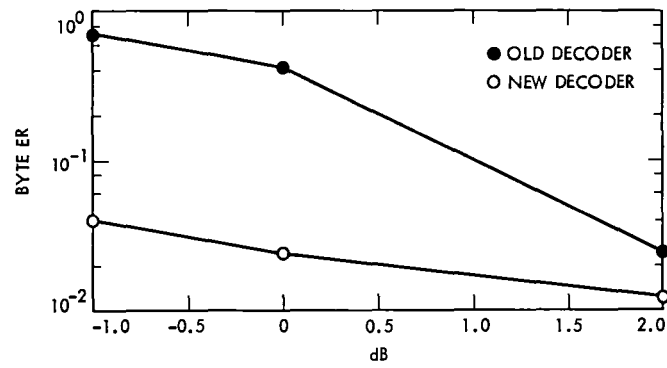


Fig. 5. A graph of byte error rates for the conventional (old) and the modified (new) maximum likelihood decoder

Appendix

We assume that symbols ± 1 are transmitted over an independent Gaussian channel. That is, if $C = (c_1, c_2, \dots, c_m)$ is transmitted, then $R = (r_1, r_2, \dots, r_m)$ is received with probability density

$$P(R|C) = \frac{1}{(\sigma\sqrt{2\pi})^m} \prod_{i=1}^m \exp[-(r_i - c_i)^2/2\sigma^2] \quad (\text{A-1})$$

where $\sigma^2 = 1/[2(E_s/N_0)]$ and E_s/N_0 is the symbol signal-to-noise ratio.

When a stream R is received, we wish to determine which was the most likely sent codeword C , or equivalently, which bit stream $X = (x_1, x_2, \dots, x_k)$ was encoded to C , which was then transmitted ($2k = m$ in the case of our rate 1/2 convolutional code). We want to maximize $P(C|R)$ over all possible codewords C .

$$P(C|R) = \frac{P(R|C)P(C)}{P(R)} \quad (\text{A-2})$$

so if all codewords are equally likely, we may equivalently maximize $P(R|C)$ over C . This in turn is the same as minimizing

$$\sum_{i=1}^m (r_i - c_i)^2 \quad (\text{A-3})$$

since nothing else depends on C .

Current decoders use Viterbi's algorithm, which finds the string which minimizes Eq. (3) without searching every possible codeword.

If it is not the case that all codewords are equally likely, the analysis above is only valid through Eq. (2). Instead, when we maximize $P(C|R)$, we must maximize

$$\frac{1}{(\sigma\sqrt{2\pi})^m} \prod_{i=1}^m \exp[-(r_i - c_i)^2/2\sigma^2] P(C) \quad (\text{A-4})$$

which is the same as minimizing

$$\sum_{i=1}^m (r_i - c_i)^2 - 2\sigma^2 \log(P(C)) \quad (\text{A-5})$$

since nothing else depends on C .

As the signal-to-noise ratio decreases, σ^2 increases (for fixed signal energy), so the choice of model for the probabilities of various codewords ("source statistics") becomes more important.

We wish to add as little complexity as possible to the Viterbi algorithm, so we would like to assume that the source is Markov, meaning the probability distribution of x_i given x_1, x_2, \dots, x_{i-1} really only depends on x_{i-r}, \dots, x_{i-1} for some small r . Unfortunately, since each pixel is represented by one 8-bit byte, the natural statistics (of one bit to the next) are not even stationary, so we are forced to do all computations one byte at a time instead of one bit at a time. We have modeled the source statistics as byte-by-byte Markov; that is, if bytes are labeled X_i , with $X_i = x_{8(i-1)+1}, \dots, x_{8i}$, we have modeled

$$P(X) = P(X_1, X_2, \dots, X_\ell) = P(X_1) \prod_{i=2}^{\ell} P(X_i|X_{i-1}) \quad (\text{A-6})$$

In addition, we have modeled $P(X_i|X_{i-1})$ as a function of the absolute differences in grey level (pixel value).

Each step of our convolutional decoder decodes one byte or pixel, and the process minimizes

$$\sum_{i=1}^{\ell} \left[\sum_{j=1}^8 (r_{8(i-1)+j} - c_{8(i-1)+j})^2 - 2\sigma^2 \log(P(X_i|X_{i-1})) \right] \quad (\text{A-7})$$

where we assume $P(X_i|X_{i-1})$ is actually distributed as modeled above.

Erasure Information for a Reed-Solomon Decoder

G. H. Pitt, III, and L. Swanson
Communications Systems Research Section

Many Reed-Solomon decoders, including the one decoding the outer code for Voyager data from Uranus, assume that all symbols have the same chance of being correct or incorrect. In some cases, like in a burst of incorrect symbols, this is not the case, and a Reed-Solomon decoder could make use of this. We examine the use of information about bit quality sent to the Reed-Solomon from an (inner) Viterbi decoder and information about the error status of adjacent symbols in decoding interleaved Reed-Solomon encoded symbols, and discover that, in a region of interest, only about 0.04 dB can be gained.

I. Introduction

A digital coding system, used on Voyager and as an international coding standard, is shown in Fig. 1. The decoding for this system, as implemented in the deep space network and at JPL, is shown in Fig. 2. The Reed-Solomon code is able to make use of soft-quantized data, however, especially in the form of symbol erasures. This article considers the value of passing "erasure information" to the Reed-Solomon decoder.

Our Reed-Solomon code is an 8-bit (255, 223) code. This means that each word consists of 255 symbols of 8 bits each. Of these symbols, 223 are information and 32 are parity. The code allows any 16 *symbol* errors to be corrected. But the code in fact has greater erasure correction capability. If some symbols are lost, or if there is reason to believe that some symbols are in error, they can be declared "erasures." The code can correct any word in which $2e + E \leq 32$, where e is the number of errors in the word and E is the number of erasures.

From time to time, methods have been discussed to make use of this erasure correction capacity. One is to develop a

method of determining quality information for Viterbi decoded bits from the rate of decoder metric renormalization, and to erase those symbols which contain bits of bad quality. Another is to make use of the interleaving of Reed-Solomon words (Fig. 3) to erase symbols which are adjacent to incorrect symbols. (This uses the fact that errors in a Viterbi decoded stream fall in "bursts"; Ref. 1.) We have used a software Viterbi decoder simulator written by Fabrizio Pollara to simulate a number of possible methods for the Viterbi decoder to pass quality information bits; these are described below, but none of them is satisfactory. We have studied the method of using information from interleaved words; this gains only about 0.04 dB, but at some signal-to-noise ratios this improves the bit error rate by about 50%. (This method may be used by the European Space Agency during its Giotto mission.)

II. Erasure Information from the Viterbi Decoder

A problem with determining quality information for Viterbi decoded bits from the rate of metric renormalization is

that we don't know how the metric renormalization is implemented in the Viterbi decoder, but we can simulate it by examining the change in the best (smallest) metric over time. We assumed that we could find the exact best metric after any bit and compare the observed byte errors to the difference in the best metric over 8 consecutive bits. This method should give us results at least as good as in any kind of renormalization scheme.

It is unclear whether the highest correlation will occur by comparing the difference in metrics from the beginning of a byte to the end of it, or from the n th bit in the byte to the n th in the next, or even by comparing one byte's metrics to another byte, so we used programs that would allow the user to try all of these schemes (Fig. 4). The difference in best metric (BM) was computed from the n th bit in one byte to the n th bit in the next and compared to the m th byte away. Then m and n were varied to achieve the highest correlation.

After shifting bits and metrics to find the highest correlation on several simulations, we found very little correlation at all. We used a program that assumes that all bytes with a corresponding metric above a certain threshold (set by the user) will be erased, and calculates the improvement gained in bytes. For example, an improvement of 2 bytes means that the Reed-Solomon decoder will react (decode or not) as though there were 2 fewer byte errors than before. Of course, this number means nothing unless compared with the number of bytes observed.

With 896 bytes observed at 0 dB, the best gain found was 15.5 bytes over the 373 bytes in error using conventional means, a gain in byte error rate from 0.364 to 0.349. Reed-Solomon frames essentially never decode at either of these error rates (less than 0.001% of the time), so there is no gain. There can be gain from erasures only if the Reed-Solomon word decodes after erasures.

With 3968 bytes observed at 1.5 dB (an interesting area for Reed-Solomon encoded data), the best gain found was 1 byte (2 predicted errors and 2 occurrences) over the conventional 201 bytes in error, a gain in byte error rate from 0.0507 to 0.0504. Considering the fact that there would almost never be more than one erasure in a frame (less than 1% of the time), this is no real gain at all. There can be gain from erasures only if there are at least two of them.

III. Erasure Information From Interleaved Reed-Solomon Words

Examining erasure based on interleaving of Reed-Solomon codewords proved more profitable. We ran the simulation for a long time and made a graph of the symbol error bursts (Fig.

5). Then, given an undecoded word W_i , we found the probability that the previous word W_{i-1} decoded and erased all symbols in W_i following those symbols which were corrected in W_{i-1} . Possibly an even better scheme is erasing only symbols in W_i surrounded by two detected in error (in W_{i-1} and W_{i+1}), which works only if the words on both sides of W_i decode.

Taking into account the fact that not all undecoded words have an adjacent decoded word (only 77% do at 1.7 dB), and even given that an adjacent word decodes the probability of decoding is not 100% (37% will then decode), only 29% of the previously undecoded words will decode with this scheme (Fig. 6).

The gain may now be calculated by making a graph of the probability of an undecoded word for the regular decoder and for the new one (Fig. 7). The calculated estimate of the probability of decoding previously undecoded words for 2 dB was about the same as for 1.7 dB, so the gain for this one-sided scheme is about 0.02 dB.

At 1.7 dB, the statistics in Fig. 5 show that if the symbol before a given symbol is in error but the one after it is not, the symbol in between is more likely to be correct than incorrect. Therefore, if an undecoded word is surrounded by two decoded ones, erasing all symbols which are adjacent to exactly one detected in error loses more than it gains. However, if only symbols surrounded on both sides by symbols are erased, 10 times as many symbols in error as correct ones will be erased.

We calculated that 61% of the undecoded words are surrounded by two decoded words. Even those 61% will not decode with probability 1, but a good guess (derived from the assumptions in the appendix) is that about 81% will now decode. Therefore, about 50% of the previously undecoded words will now decode at 1.7 dB (Fig. 6).

The gain may be calculated by making a graph of the probability of an undecoded word for the regular decoder and for the new one (Fig. 7). We assume that the estimated probability of decoding previously undecoded words for 2 dB is about the same as for 1.7 dB, so the curve may be drawn. The gain for this two-sided scheme is about 0.03 dB.

From the above information, we may naively estimate the probability of decoding previously undecoded words if we try the two-sided scheme when both W_{i-1} and W_{i+1} decode and the one-sided scheme if only one of them decodes.

The probability that exactly one side decodes is $2 \times (77\% - 61\%) = 32\%$ and 29% of those will now decode, so we get $50\% + (32\% \times 29\%) = 59\%$ of the previously undecoded words to now decode (Fig. 6).

The gain may be calculated by making a graph of the probability of an undecoded word for the regular decoder and for the new one (Fig. 7). We assume that the estimated probability of decoding previously undecoded words for 2 dB is about the same as for 1.7 dB, so the curve may be drawn. The gain for this combination of the two schemes is about 0.04 dB.

Figures 6 and 7 show the probability of not decoding a word instead of the probability of not decoding a single

symbol. To find that, one must multiply by the fraction of symbols in error given that a word doesn't decode (the mean number of symbols in error given that a word doesn't decode divided by 255). These new schemes decode most of the undecoded words with a low number of errors, so the fraction of errors given that a word doesn't decode will increase and the symbol error rate gain may not be quite as good. (The change in mean number of symbols in error given that a word doesn't decode should be about from 20 to 21.)

Reference

1. Miller, R. L., Deutsch, L. J., and Butman, S. A., *On the Error Statistics of Viterbi Decoding and the Performance of Concatenated Codes*, JPL Publication 81-9, Jet Propulsion Laboratory, Pasadena, Calif., September 1, 1981.

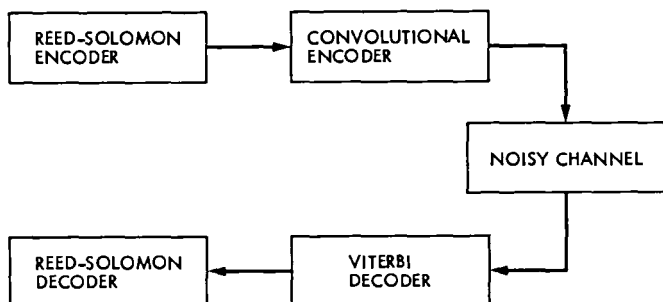


Fig. 1. A digital data coding scheme

WORD 1	WORD 2	WORD 3	WORD 4	WORD 5
SYMBOL 1	SYMBOL 2	SYMBOL 3	SYMBOL 4	SYMBOL 5
SYMBOL 6	SYMBOL 7	SYMBOL 8	SYMBOL 9	SYMBOL 10
SYMBOL 11	SYMBOL 12	SYMBOL 13	SYMBOL 14	SYMBOL 15
•	•	•	•	•
•	•	•	•	•
•	•	•	•	•
SYMBOL 1271	SYMBOL 1272	SYMBOL 1273	SYMBOL 1274	SYMBOL 1275

Fig. 3. The 5 interleaved (vertical) Reed-Solomon codewords of 255 symbols each. Notice the sequence of the symbols.

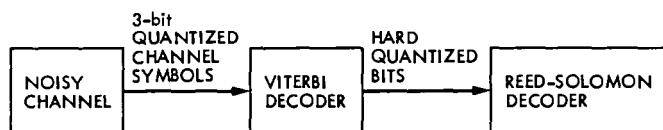


Fig. 2. The current decoding method

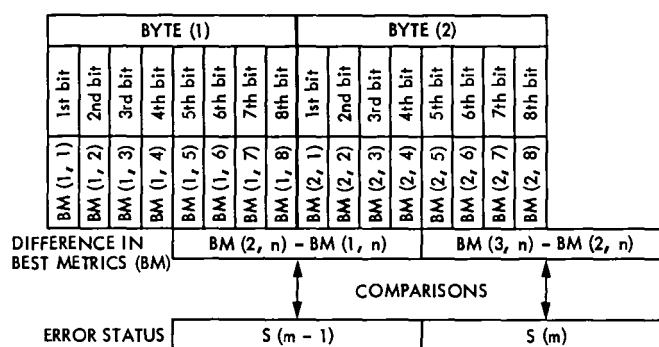


Fig. 4. Difference in best metric (BM) vs byte error status (S) comparisons. Variable n is the bit in the byte where you want to start computing and m is the byte shift.

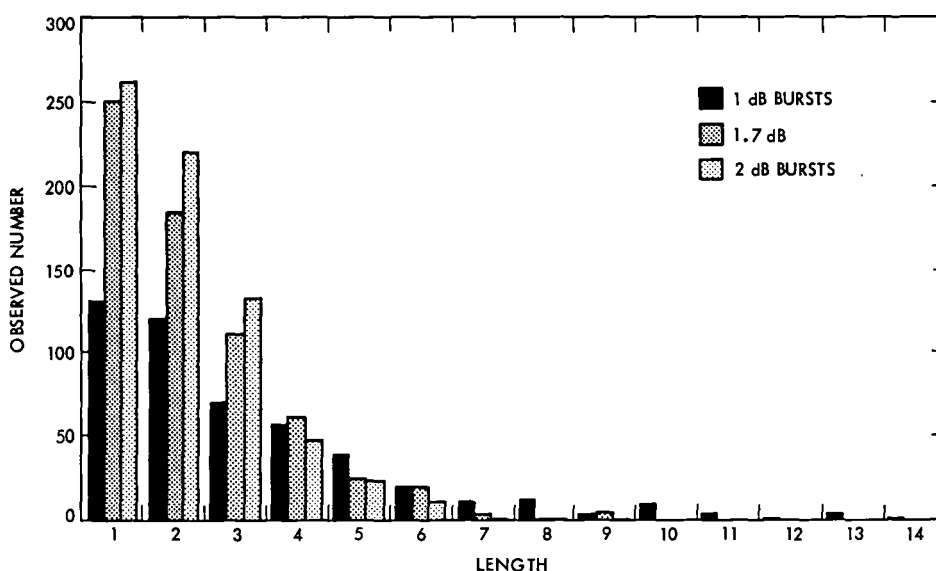


Fig. 5. A graph of the length of bursts for 1, 1.7, and 2 dB as observed in simulation and normalized to 1500 errors at each E_p/N_0

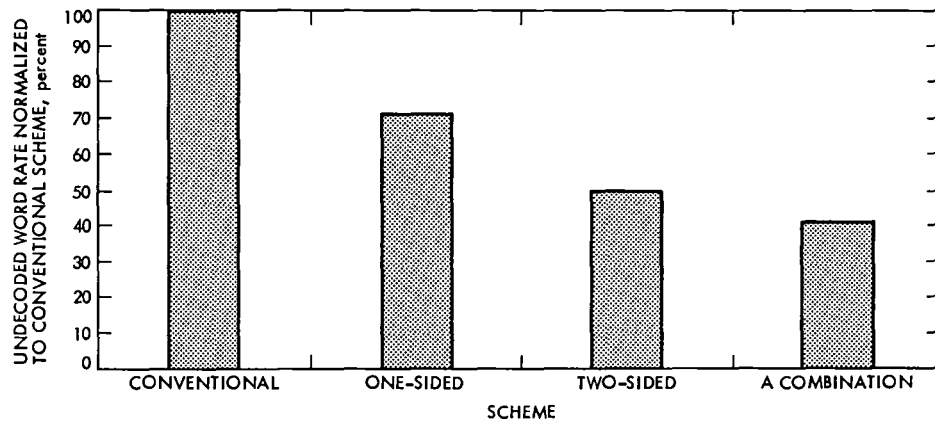


Fig. 6. A graph comparing decoded word rates for the old scheme and the three new ones

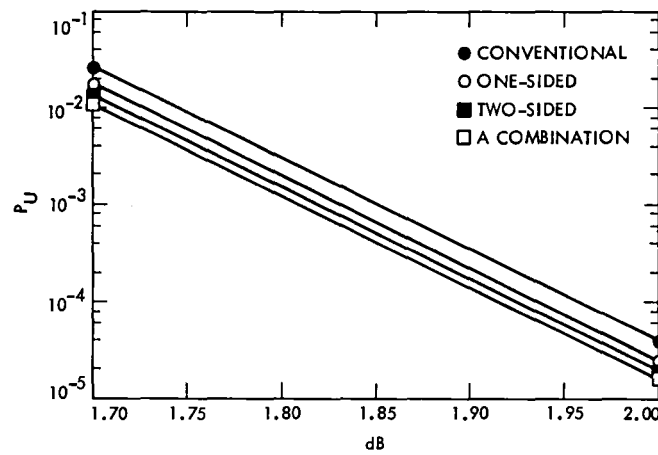


Fig. 7. A graph of the probability of an undecoded word for the old scheme and the three new ones

Appendix

The above analyses may all be derived from the following assumptions (Viterbi decoder errors occur in “bursts”; Ref. 1) and observations using the software Viterbi decoder at 1.7 dB.

Total number of symbols = 38528

Total number of errors = 1497

Total number of bursts = 659

Number of correct symbols with two adjacent incorrect symbols = 33

Total number of triple errors = 430

If W_i = word i and E_i = the number of errors in W_i then we assume:

$P(W_i \text{ now decodes} | W_{i-1}, W_{i+1} \text{ decoded and } E_i = 17) = 100\%$

$P(W_i \text{ now decodes} | W_{i-1}, W_{i+1} \text{ decoded and } E_i = 18) = 90\%$

$P(W_i \text{ now decodes} | W_{i-1}, W_{i+1} \text{ decoded and } E_i = 19) = 55\%$

$P(W_i \text{ now decodes} | W_{i-1}, W_{i+1} \text{ decoded and } E_i = 20) = 10\%$

$P(W_i \text{ now decodes} | W_{i-1}, W_{i+1} \text{ decoded and } E_i = 21 \text{ to } 255) = 0\%$

In the one-sided scheme, we assumed that only symbols that were adjacent to a symbol that decoded with the normal decoder may now be affected by the new new scheme. This makes the estimate a little low since we could decode one word with the new scheme and then use that information to decode the word next to it, but this would seem to happen so infrequently that we didn't try to estimate its probability.

A VLSI Single Chip 8-Bit Finite Field Multiplier

I. S. Hsu, L. J. Deutsch, T. K. Truong, and H. M. Shao
Communications Systems Research Section

This article describes a VLSI architecture and layout for an 8-bit finite field multiplier. The algorithm used in this design was developed by Massey and Omura (Ref. 1). A normal basis representation of finite field elements is used to reduce the multiplication complexity. It is shown in this article that a drastic improvement has been achieved in this design. This multiplier will be used intensively in the implementation of an 8-bit Reed-Solomon decoder and in many other related projects.

I. Introduction

The era of VLSI digital signal processors has arrived and its impact is evident in many areas of research. The trend is to put more and more elements on a single silicon chip in order to enhance the performance and reliability of the system. These techniques are currently being used in the VLSI-based Reed-Solomon decoder using a small number of chips. Conventional decoders use several hundred chips. Our research has shown that it is possible to fabricate an entire decoder on a single chip.

Recently, finite field theory has found widespread applications. Examples include cryptography, coding theory and computer arithmetic. What is more, finite field arithmetic is the central part in the implementation of Reed-Solomon coders. Because there are quite a few finite field multiplications in the Reed-Solomon decoding algorithm, a small finite field multiplier is urgently needed for the implementation of a single VLSI chip Reed-Solomon decoder.

The major problem encountered in designing a small multiplier with the conventional method is the large quantity of hardware required. The conventional methods for implementing a finite field multiplier use either full parallel or

table look-up algorithms. These methods lead to architectures that are not easily realized in a VLSI circuit. Massey and Omura (Ref. 1) recently developed a new multiplication algorithm for Galois fields based on a normal basis representation of field elements. In this article, a pipeline structure based on Massey and Omura's algorithm, developed in Ref. 2, is used to realize an 8-bit finite field multiplier. It is shown in this article that the chip area of this 8-bit multiplier is only about 1.3 times larger than the 4-bit multiplier designed and fabricated previously.

II. A Functional Description of the 8-Bit Multiplication Chip

The function f as described in Ref. 2 is chosen to be the following expression:

$$\begin{aligned} f(a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7; b_0, b_1, b_2, b_3, b_4, b_5, b_6, b_7) \\ = a_5b_0 + a_6b_0 + a_3b_1 + a_5b_1 + a_4b_2 + a_5b_2 + a_6b_2 + a_7b_2 \\ + a_1b_3 + a_4b_3 + a_2b_4 + a_3b_4 + a_0b_5 + a_1b_5 + a_2b_5 + a_6b_5 \\ + a_0b_6 + a_2b_6 + a_5b_6 + a_6b_6 + a_2b_7 \end{aligned} \quad (1)$$

where the generator polynomial of this finite field is

$$g(X) = X^8 + X^5 + X^3 + X + 1 \quad (2)$$

There are a variety of different possible expressions of the function f ; however, the one above was chosen because it has the least number (21) of terms (Ref. 3). Since each term in the expression represents a conduction line in the AND-plane of a PLA (programmable logic array), then obviously, the fewer the terms are, the less area will be spent in implementation.

An overall block diagram of the chip is shown in Fig. 1. In Fig. 1, Vdd and GND are power pins. The signals PHI-1 and PHI-2 are two non-overlapping phases of a system clock. The input bits of the multiplicand and multiplier are fed into the chip serially through the data input pins, DATA-IN1 and DATA-IN2. Similarly, the product of these two elements is transmitted out of the chip sequentially from the data output pin, DATA-OUT. The control signal LOAD is high for one bit time every 8 bits. The signal N-LOAD is just the logical complement of signal LOAD. Both the LOAD and N-LOAD signals are used for converting input data from serial type to parallel type.

Figure 2 shows the block diagram of an 8-bit finite field multiplier using Massey-Omura's normal basis algorithm. The circuit is divided into three units which are discussed in the following:

(1) *Serial-to-Parallel Unit.* This unit performs the serial to parallel conversion of input data sequence. In Fig. 3, R_i 's and Q_i 's, for $1 \leq i \leq 2$, represent 7-bit and 8-bit shift registers with parallel load, respectively. The input bits of the multiplicand and multiplier are first stored in the R_i registers. These data will then be loaded into the Q_i registers for every eight clock cycles. To reiterate, when signal LOAD is low and N-LOAD is high, data come in bit by bit. At the eighth clock cycle, signal LOAD is high and signal N-LOAD becomes low. Data in the R_i registers will then be loaded into Q_i registers. The output of Q_i registers is the input of the AND-generation unit.

(2) *AND-Generation Unit.* This unit generates the ANDed terms of the input bits of the multiplicand and multiplier. The AND functions are configured in a structure of AND-plane of PLA because of the regularity and modularity that a PLA possesses. The inputs to this unit are the complemented values of the input bits of the multiplicand and multiplier. The output of this unit will be the ANDed terms of the input. These ANDed terms will be sent to the exclusive-or unit.

(3) *Exclusive-Or Unit.* In finite field arithmetic, if the ground field is $GF(2)$, addition is just the exclusive-or operation. Since there are twenty additions in Eq. (1), twenty exclusive-or cells are required for the conventional method

to implement these exclusive-or operations. Because each exclusive-or cell needs a substantial amount of chip area, 20 of these will consume so much area such that a small multiplier will not be possible. Consequently, this will prohibit the implementation of a single chip 8-bit Reed-Solomon decoder. In our design, an alternate way for implementing the exclusive-or operation is used. This new technique was developed and used in the fabrication of the Multicode Convolutional Encoder chip (Ref. 5).

Figure 4 shows the block diagram of the conceptual exclusive-or array. This array comprises twenty-one subcells. Each subcell performs a switching operation. As shown in Fig. 5, if input A is a "one," i.e., it is high, then its complement signal \bar{A} is low and signals $S1$ and $S2$ will be switched. On the other hand, if signal \bar{A} is low, then A is high and signals $S1$ and $S2$ will not be switched at this moment.

There are two signal paths through each row of the subcell. At the extreme left of the array, a "one" is connected to one of these and a "zero" to the other. Each time this pair of signals passes through a subcell, they exchange places if the corresponding output of AND-generation unit is "one." In this way, the pair of signals has gone through a number of path exchanges equal to the number of ones in the AND-generation unit output. If the number of ones of the AND-generation unit output is even, then the signals are in the same places as they started. If the number of ones is odd, they will come out reversed.

III. The Estimated Performance of the 8-Bit Multiplier

The 8-bit multiplier chip was designed using the UNIX-based integrated CAD system (Ref. 6). The entire chip was simulated on a general purpose computer using ESIM (a logic level simulation program; Ref. 7) and SPICE (a transistor level simulation program; Ref. 8). The layout of the multiplier was accomplished using the program CAESAR (Ref. 9). LYRA (Ref. 10) was used to check the resulting layout against a set of geometric rules supplied by the fabricator. Timing simulation was done using CRYSTAL (Ref. 11). The circuit comprises about 2000 transistors.

The chip described here was sent out for fabrication through the MOSIS service (Ref. 12). The technology used was 3 μm NMOS. When the completed chips return they will be evaluated and tested. The layout of the multiplier is shown in Fig. 6. The area of this chip is $1200 \times 900 \mu\text{m}^2$ while the chip area of a 4-bit finite field multiplier designed previously is $1000 \times 800 \mu\text{m}^2$. The 8-bit one is only 1.3 times larger than the 4-bit. The estimated operation speed of this chip is 10 MHz and the estimated power consumption at this frequency is 30 mW.

References

1. J.L. Massey and J.K. Omura, U.S. Patent Application of "Computational Method and Apparatus for Finite Field Arithmetic," submitted in 1981.
2. C.C. Wang, T.K. Truong, H.M. Shao, L.J. Deutsch, J.K. Omura and I.S. Reed, "VLSI Architecture of Computing Multiplications and Inverse in $GF(2^m)$," *TDA Progress Report 42-75*, pp. 52-64, 1983. Jet Propulsion Laboratory, Pasadena, California.
3. C.C. Wang, "Computer Simulation of Finite Field Multiplications Based on Massey-Omura's Normal Basis Representation of Field Elements," private communication, 1985.
4. C.M. Mead and L. Conway, *Introduction to VLSI Systems*, New York, Addison-Wesley Publishing Company, 1980.
5. L.J. Deutsch, "A VLSI Implementation of a Multicode Convolutional Encoder," *TDA Progress Report 42-72*, pp. 61-69, 1985. Jet Propulsion Laboratory, Pasadena, CA.
6. L.J. Deutsch, "An UNIX-Based CAD System for the Design and Testing of Custom VLSI Chips," *TDA Progress Report 42-81*, pp. 51-62, 1985. Jet Propulsion Laboratory, Pasadena, CA.
7. C. Terman, "ESIM - An Event Driven Simulator," Technical Memorandum, Electrical Engineering Department, Massachusetts Institute of Technology, 1977.
8. L.W. Nagel and D.O. Pederson, "SPICE - Simulation Program with Integrated Circuit Emphasis," *Memorandum No. ERL-M382*, Electronics Research Laboratory, University of California, Berkeley.
9. J. Ousterhout, "CAESAR - Editing VLSI Circuits with CAESAR," Computer Science Division, Technical Memorandum, Electrical Engineering and Computer Science Department, University of California, Berkeley, April 21, 1982.
10. J. Ousterhout, "LYRA - A Design Rule Checker," Computer Science Division, Technical Memorandum, Electrical Engineering and Computer Science Department, University of California, Berkeley, April 21, 1982.
11. J. Ousterhout, "Using Crystal for Timing Analysis," Computer Science Division, Technical Memorandum, Electrical Engineering and Computer Science Department, University of California, Berkeley, March, 1983.
12. The MOSIS Project, The MOSIS System (What It Is and How to Use It). Information Science Institute, University of Southern California, *Publication ISI/TM-84-128*, Marina Del Rey, CA.

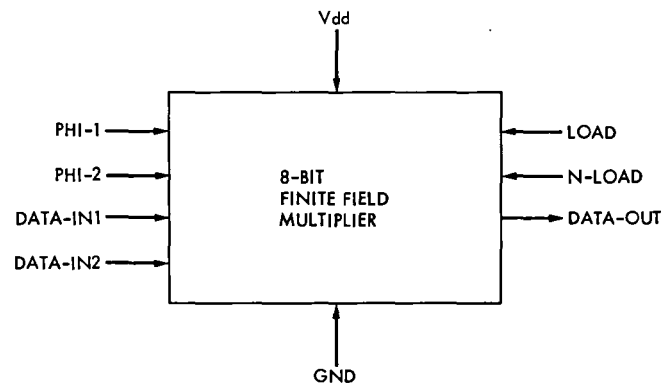


Fig. 1. The symbolic diagram of an 8-bit finite field multiplier

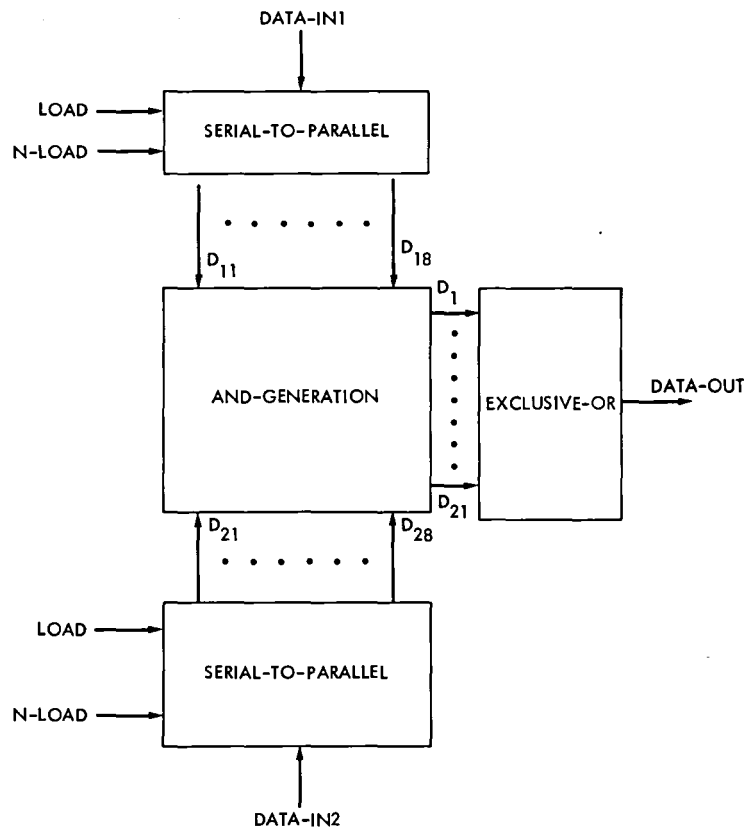


Fig. 2. The block diagram of an 8-bit finite field multiplier

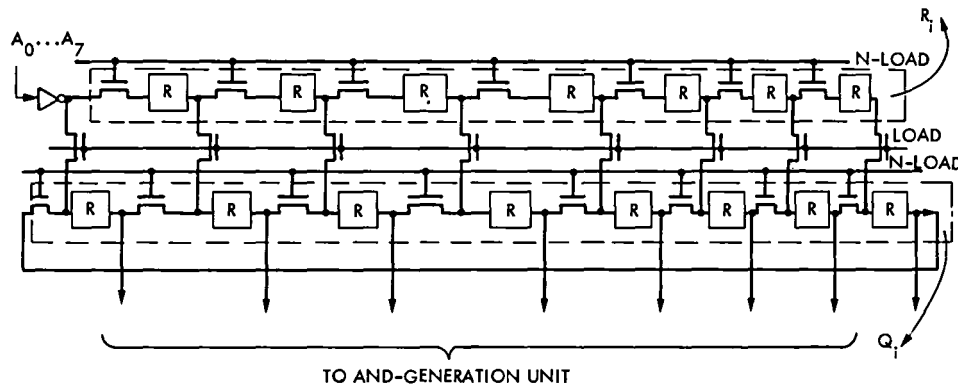


Fig. 3. The logic diagram of R_i 's and Q_i 's registers

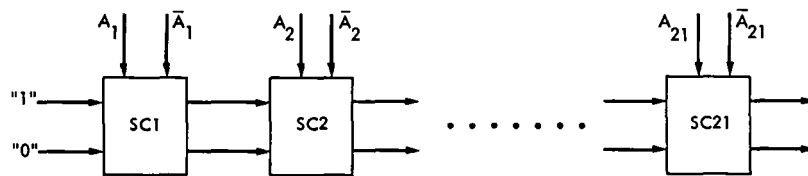


Fig. 4. The block diagram of conceptual exclusive-or array

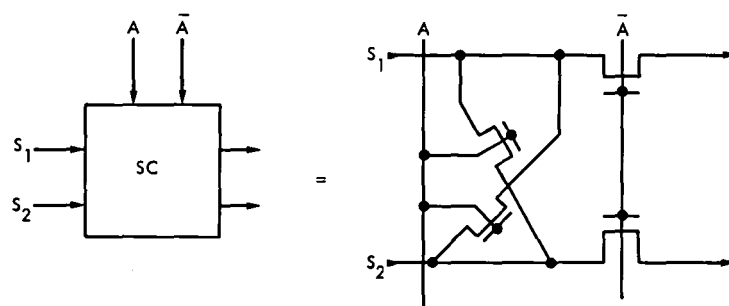


Fig. 5. The circuit diagram of a subcell in the conceptual exclusive-or arrays

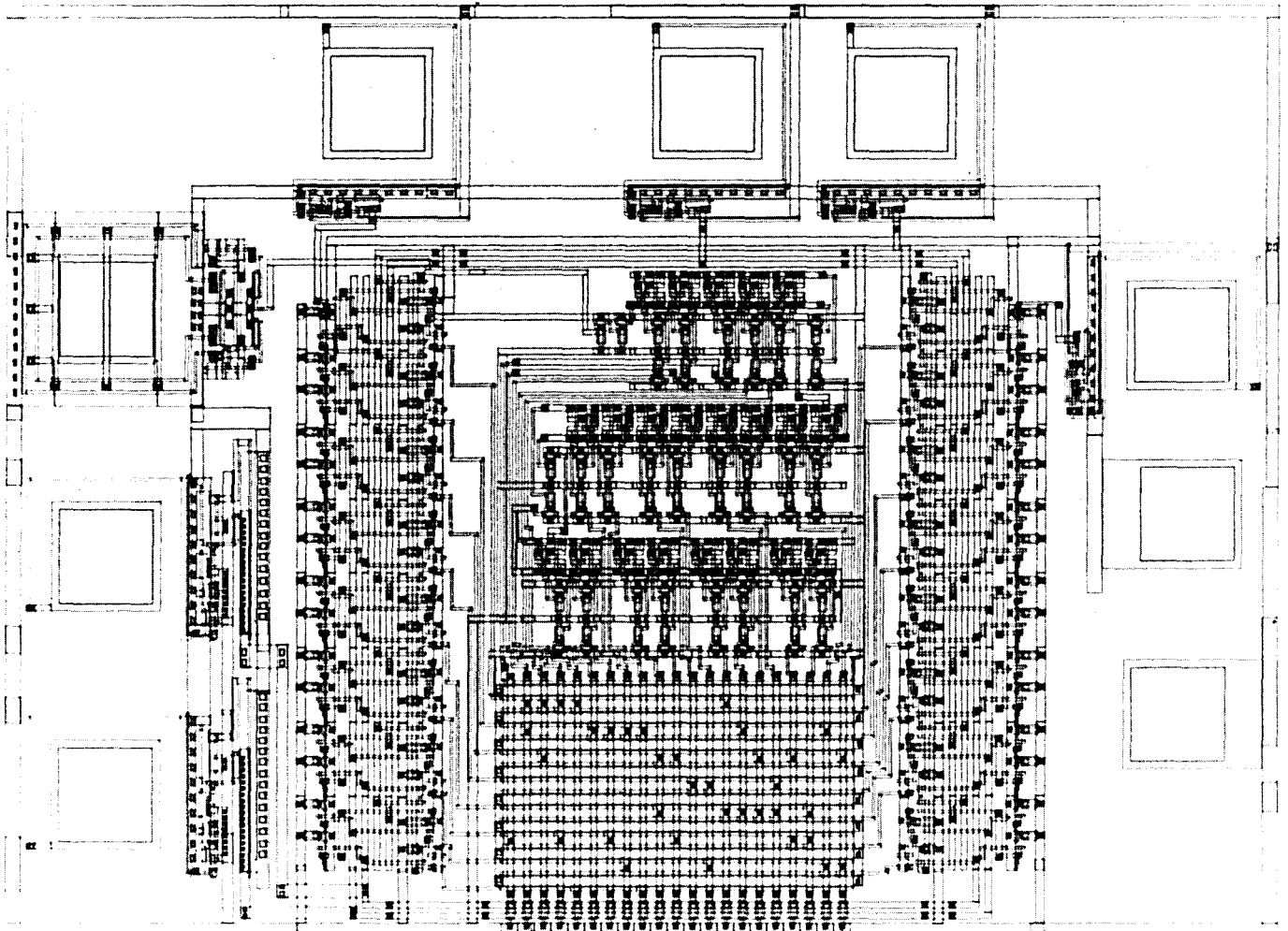


Fig. 6. The layout of an 8-bit finite field multiplier

D7-33

A VLSI Single Chip (255, 223) Reed-Solomon Encoder

I. S. Hsu, L. J. Deutsch, T. K. Truong
Communication Systems Research Section

I. S. Reed
University of Southern California

This article presents a description of a working single chip implementation of a Reed-Solomon encoder. The code used is the CCSDS (Consultative Committee on Space Data Systems) standard (255, 223) code. The architecture that leads to this single VLSI chip design makes use of a bit-serial finite field multiplication algorithm of E. R. Berlekamp.

I. Introduction

A concatenated coding system consisting of a convolutional inner code and a Reed-Solomon outer code has been adopted as a guideline for downlink telemetry for future space missions by CCSDS (Consultative Committee for Space Data Systems) (Ref. 1). The participants in this committee include the European Space Agency (ESA) and NASA as well as space agencies from many other nations. The convolutional inner code is the same (7, 1/2) code used by NASA's Voyager project. The outer Reed-Solomon code is a (255, 223) block code on 8-bit symbols and it is capable of correcting up to 16 symbol errors. The performance of such schemes is investigated in Ref. 2 where it is shown that this concatenated channel provides a coding gain of almost 2 dB over the convolutional-only channel at a decoded bit error rate of 10^{-5} . One of the benefits of concatenated coding, and one of the main motivations for its acceptance as a standard system, is that it provides for a nearly error-free communications link at fairly low signal power levels. This means that source data compression techniques (Ref. 3) can be used to help increase channel throughput without a substantial change in overall error rate. An end-to-end study of a system using concatenated coding with data compression can be found in Ref. 4.

A Reed-Solomon encoder is basically a circuit which performs polynomial division in a finite field. Such circuits are well known (Ref. 5) and their implementation is straightforward. The major problem in designing a small encoder is the large quantity of hardware that is necessary to perform the finite field multiplications. Due to the limited weight, space and power that can be allotted to a spacecraft instrument, the equipment used must be as light and as small as possible. Therefore, a single chip Reed-Solomon encoder can be a significant advantage in deep-space probe missions. This encoder represents a considerable space, weight and power savings over the smallest existing encoder (about 30 chips).

A conventional encoder described in Ref. 5 for the (255, 223) RS code, requires 32 finite field multipliers. These multipliers are usually implemented as full parallel multipliers or table look-up multipliers. The use of either of these multiplication algorithms prohibits the implementation of the encoder on a single medium density VLSI chip.

Fortunately, E. R. Berlekamp (Ref. 6) developed a serial algorithm for finite field multiplication. Berlekamp's algorithm requires only shifting and exclusive-or operations. Recently, it

was shown (Ref. 7) that this multiplication algorithm has enabled the design of a workable VLSI architecture, and that this new dual-basis (255, 223) RS encoder can be realized readily on a single VLSI chip with NMOS technology. This article presents the results of the implementation and testing of the RS encoder.

II. A Single VLSI Chip of a (255, 223) RS Encoder

Berlekamp's bit-serial multiplication algorithm for a (255, 223) RS-encoder over $GF(2^8)$ is presented in Refs. 6 and 7.¹ A VLSI architecture for implementing this encoder using Berlekamp's multiplication algorithm is presented in Ref. 7. The block diagram of the (255, 223) RS encoder is exhibited here in Fig. 1.² In Fig. 1, one observes that the circuit is divided into five units: the Product unit, Remainder unit, Quotient unit, I/O unit, and Control unit. The use of each unit is explained in detail in Ref. 7. An overall block diagram of the implemented chip is shown in Fig. 2. In Fig. 2, VDD and GND are power pins. The signals ϕ_1 and ϕ_2 are the two phases of a system clock. The information symbols are fed into the chip serially through the data-in pin, DIN. Similarly, the encoded codeword is transmitted out of the chip sequentially from the data-out pin, DOUT. The control signal SL is set to 1 (logic 1) when the information symbols are loaded into the chip. After this, SL is set to 0. The control signal "START" resets a 3-bit word counter in this chip before the encoding process begins.

The entire chip was simulated on a general purpose computer using ESIM (a logic-level simulation program; see Ref. 8) and SPICE (a transistor level circuit simulation program; see Ref. 9). The layout of the encoder was accomplished using the program CAESAR (Ref. 10). LYRA (Ref. 11) was used to check the resulting layout against a set of geometric rules supplied by the fabricator. Timing simulation was done using CRYSTAL (Ref. 13). The circuit comprises about 3000

transistors. It was fabricated using the MOSIS service (Ref. 12) and the technology used was 4 μ m NMOS. The layout of the encoder³ was modified and the new version is shown in Fig. 3(a). A photograph of a fabricated chip is shown in Fig. 3(b).

III. The Procedure for Testing a (255, 223) RS Encoder

The testing of this chip was accomplished at the Jet Propulsion Laboratory (JPL). The JPL testing system consists of a custom-built general purpose hardware tester called the Digital Microcircuit Functionality Tester (DMFT) (Ref. 14) and a VAX 11/750 computer. A program called "Logic" (Ref. 14) acts as an interface between these two subsystems. Logic is a program which allows a chip designer to create a set of test vectors for the functional simulation and testing of a circuit. Two types of tests are supported currently. The first is a link to the ESIM logic simulator. The second is an interface to the DMFT tester. Logic can read and edit files that contain test vector and tester configuration information. Once a set of test vectors is generated, they may be saved or applied to either ESIM or the DMFT tester. The results are displayed on the terminal screen in a logic analyzer format. The DMFT, at present, is capable of supplying 8 inputs to a chip and monitoring 8 outputs. Each input and output sequence may be as many as 4096 bits long.

Since a codeword for this RS code contains 255 symbols, the computation of a complete codeword requires 255 "symbol cycles." A symbol cycle is the time interval required for executing a complete cycle of Berlekamp's algorithm. Since a symbol consists of 8 bits, a symbol cycle contains 8 "bit cycles." A bit cycle is the time interval for executing one step in Berlekamp's algorithm.

In this design, a bit cycle corresponds to one period of the clock. The total number of clock cycles required to encode a single RS codeword is therefore 255×2040 .

The chip was successfully tested at up to 3.2 MHz using the DMFT. At the maximum clock rate, the chip consumed 100 mW of power.

¹See also M. Perlman and J. J. Lee, "Reed-Solomon Encoders - Conventional Versus Berlekamp's Architecture," Interoffice Memo No. 3610-81-119 ISPM (internal document), Jet Propulsion Laboratory, Pasadena, CA.

²*Ibid.*, Figure 3.

³*Ibid.*; see Figure 6 for the original layout of the encoder.

Acknowledgements

The authors wish to acknowledge M. Perlman and J. J. Lee who brought the Berlekamp algorithm to our attention and worked with Berlekamp to produce the prototype discrete IC encoder upon which this chip is modeled. Additionally, the authors wish to thank Erlend Olson who designed the DMFT system in response to our particular testing needs.

References

1. "Recommendation for Space Data System Standards: Telemetry Channel Coding," (Blue Book), Consultative Committee for Space Data System, CCSDS Secretariat, Communications and Data Systems Division, Code TS, NASA, Washington, D.C., May 1984.
2. Miller, R. L., Deutsch, L. J., and Butman, S. A., "On the Error Statistics of Viterbi Decoding and the Performance of Concatenated Codes," *JPL Publication 81-9*, Jet Propulsion Laboratory, Pasadena, Calif., September 1, 1981.
3. Rice, R. F., "Some Practical Universal Noiseless Coding Techniques," *JPL Publication 79-22*, Jet Propulsion Laboratory, Pasadena, Calif., March 15, 1979.
4. Rice, R. F., "End-to-End Imaging Information Rate Advantages of Various Alternative Communication Systems," *JPL Publication 82-61*, Jet Propulsion Laboratory, Pasadena, Calif., September 1, 1982.
5. Peterson, W. W., and Weldon, E. J., *Error-Correcting Codes*, Cambridge, MIT Press, 1972.
6. Berlekamp, E. R., "Bit-Serial Reed-Solomon Encoders," *IEEE Trans. Inform. Theory*, Vol. IT-28, No. 6, pp. 869-874, November 1982.
7. Hsu, I. S., Reed, I. S., Truong, T. K., Wang, K., Yeh, C. S., and Deutsch, L. J., "The VLSI Implementation of a Reed-Solomon Encoder Using Berlekamp's Bit-Serial Multiplier Algorithm," *IEEE Trans. on Computers*, Vol. C-33, No. 10, October 1984.
8. Terman, C. "ESIM - An Event Driven Simulator," Electrical Engineering Department, Massachusetts Institute of Technology, 1977.
9. Negal, L. W., and Pederson, D. O., "SPICE - Simulation Program with Integrated Circuit Emphasis," No. ERL-M382, Electronic Research Laboratory, University of California, Berkeley.
10. Ousterhout, J., "CAESAR - Editing VLSI Circuits with CAESAR," Computer Science Division, Electrical Engineering and Computer Science Department, University of California, Berkeley, April 12, 1982.
11. Ousterhout, J. "LYRA - A Design Rule Checker," Computer Science Division, Electrical Engineering and Computer Science Department, University of California, Berkeley, April 21, 1982.
12. The MOSIS Project, *The MOSIS System (What It Is and How to Use It)*, Information Science Institute, Publication ISI/TM-84-128, Marina Del Rey, CA, 90292.

13. Ousterhout, J. "Using Crystal for Timing Analysis," Computer Science Division, Electrical Engineering and Computer Science Department, University of California, Berkeley, March 1983.
14. Olson, E. M., and Deutsch, L. J., "A System for the Functional Testing and Simulation of Custom and Semicustom VLSI Chips," *TDA Progress Report*, Jet Propulsion Laboratory, Pasadena, Calif. (in preparation).

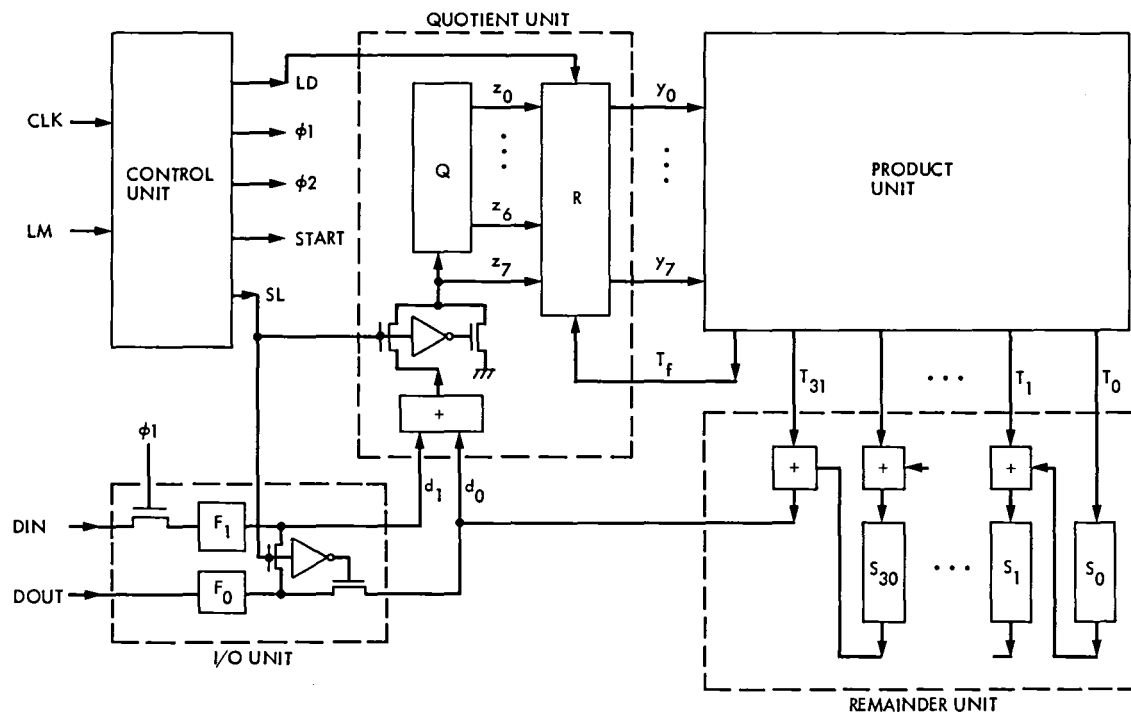


Fig. 1. Block Diagram of the (255, 223) RS encoder

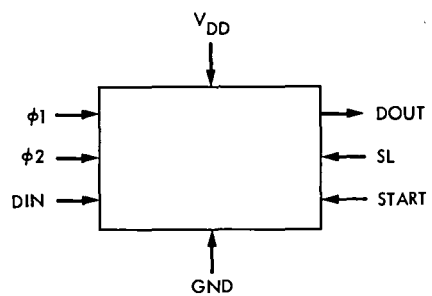


Fig. 2. The symbolic diagram of a (255, 223) RS encoder chip

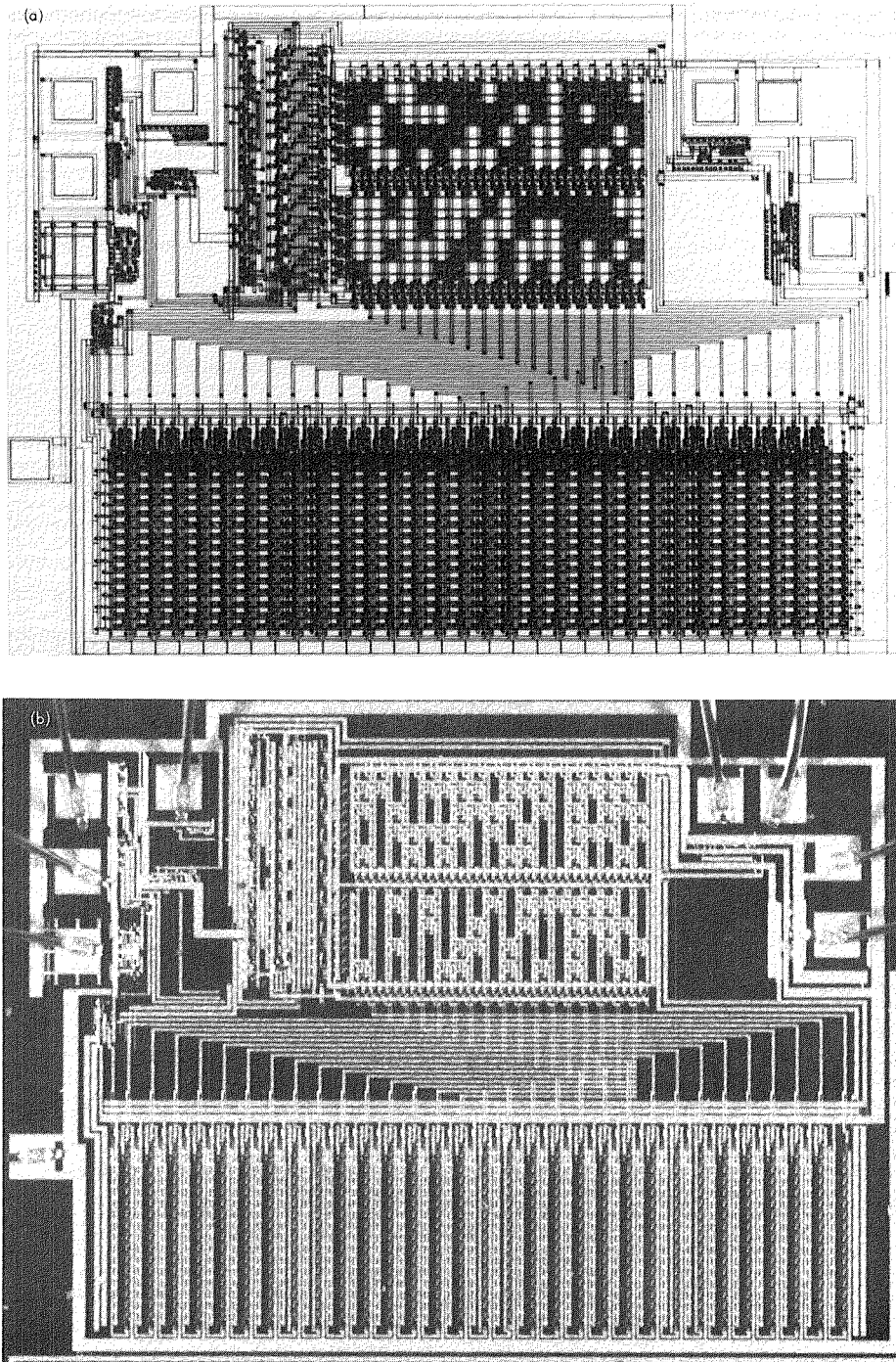


Fig. 3. The (255, 223) RS encoder chip: (a) layout, (b) photograph of working chip

Carrier Tracking by Smoothing Filter Can Improve Symbol SNR

C. A. Pomalaza-Raez and W. J. Hurd
Communications Systems Research Section

The potential benefit of using a smoothing filter to estimate carrier phase over use of phase locked loops (PLL) is determined. Numerical results are presented for the performance of three possible configurations of the DSN Advanced Receiver. These are Residual Carrier PLL, Sideband Aided Residual Carrier PLL, and finally Sideband Aiding with a Kalman Smoother. The average symbol SNR after losses due to carrier phase estimation error is computed for different total power SNRs, symbol rates and symbol SNRs. It is found that smoothing is most beneficial for low symbol SNRs and low symbol rates. Smoothing gains up to 0.4 dB over a Sideband Aided Residual Carrier PLL, and the combined benefit of Smoothing and Sideband Aiding relative to a Residual Carrier Loop is often in excess of 1 dB.

I. Introduction

Smoothing as a way to improve carrier tracking has been proposed and analyzed (Ref. 1). It has been shown that by cascading an optimum Kalman smoother with a carrier tracking loop, up to a 6-dB reduction in the phase error variance can be obtained with a second order smoother. This work presents the potential improvement in the effective data signal-to-noise ratio (SNR) by using a smoother in the DSN Advanced Receiver. Optimum modulation indexes and sideband aiding are used.

The average symbol SNR, after degradation due to carrier phase estimation error (radio loss), is used as a measure of

performance when comparing the various possible configurations for the receiver. Also, conditions such as the minimum SNR needed to ensure acquisition and lock of the loops are satisfied in each case.

II. Analysis

The analysis is carried out for three possible configurations of the DSN Advanced Receiver. They are:

- (1) Residual Carrier tracking only (RC)
- (2) Residual Carrier plus Sideband Aiding (SA)
- (3) SA plus Smoothing (SM)

The received signal for the DSN Advanced Receiver is assumed to be of the form (Ref. 2)

$$r(t) = \sqrt{2P} \sin(\omega_i t + \Delta D(t) + \Theta_c) + n(t) \quad (1)$$

where

$r(t)$ = received signal (v)

P = average signal power (v^2)

Δ = modulation index (rad)

$D(t) = d(t) \text{sgn}(\sin(\omega_{sc} t + \Theta_{sc}))$

$d(t) = \sum_{\ell=-\infty}^{\infty} a_{\ell} p(t - \ell T)$, $a_{\ell} = \pm 1$ with equal probability

$p(t) = 1$ for $0 < t \leq T$, 0 elsewhere

ω_i = received IF frequency (rad/s)

Θ_c = carrier phase (rad)

ω_{sc} = subcarrier frequency (rad/s)

Θ_{sc} = subcarrier phase (rad)

$n(t) = \sqrt{2} n_c(t) \cos(\omega_i t + \Theta_c) - \sqrt{2} n_s(t) \sin(\omega_i t + \Theta_c)$
is a noise process with $n_c(t)$ and $n_s(t)$ being statistically independent, stationary, band-limited white Gaussian noise processes with one-sided spectral density N_0 (v^2/Hz) and one sided bandwidth W .

T = symbol time (s)

The carrier phase estimate is $\hat{\Theta}_c$, and the resulting carrier phase error is $\phi = \hat{\Theta}_c - \Theta_c$. It is assumed that the phase error process $\phi(t)$ is essentially constant during one symbol time. Thus, for example, in the RC case the demodulated waveform is

$$z(t) = \sqrt{P} \sin \Delta D(t) \cos \phi(t) + n'(t) \quad (2)$$

where $n'(t)$ is a narrow-band Gaussian process with one-sided spectral density N_0 .

Conditioned on $\phi(t)$, the symbol SNR at the output of the symbol integration is

$$\frac{E_s(\phi)}{N_0} = \frac{PT \sin^2 \Delta \cos^2 \phi}{N_0} \quad (3)$$

and the average symbol SNR after losses due to ϕ , $[E_s/N_0]_{\text{ave}}$, can be approximated by

$$\left[\frac{E_s}{N_0} \right]_{\text{ave}} = \frac{PT \sin^2 \Delta}{N_0} (1 - \sigma_{\phi}^2) \quad (4)$$

where σ_{ϕ}^2 is the phase error variance. In Eq. (4) it is assumed that σ_{ϕ} is small ($\ll 1$ rad). This is valid for all the cases discussed here; thus, for residual carrier tracking only, from Ref. 2

$$\sigma_{\phi_1}^2 = \frac{N_0 B_L}{P \cos^2 \Delta} \quad (5)$$

where B_L is the single-sided loop bandwidth.

For residual carrier tracking the modulation index which maximizes the average symbol SNR is independent of the symbol SNR and satisfies $\cos^4 \Delta = N_0 B_L / P$, under the above conditions. This follows from substituting Eq. (5) into Eq. (4) and maximizing.

For sideband aiding,

$$\sigma_{\phi_2}^2 = \frac{1}{\frac{P}{N_0 B_L} \left[\cos^2 \Delta + \frac{\sin^2 \Delta}{1 + \frac{1}{(2E_s/N_0)}} \right]} \quad (6)$$

where

$$E_s = PT \sin^2 \Delta \quad (7)$$

Finally, for a smoothing estimator following a loop with RC plus SA, the best possible result with a second order smoother is (Ref. 1)

$$\sigma_{\phi_3}^2 = \sigma_{\phi_2}^2 / 4 \quad (8)$$

The average symbol SNR depends on the modulation index Δ . Therefore, it is necessary to find the best Δ for each case. This best Δ is that which maximizes $[E_s/N_0]_{\text{ave}}$ subject to the constraint of sufficient energy in the unsmoothed carrier loop to ensure acquisition and lock, without "too many" cycle slips.

One possible way to restrict the modulation index would be to assume that there is sufficient carrier power to lock on the residual carrier only loop. This restriction leads to an easy acquisition procedure, but not to the best effective SNR. We therefore use a criterion for RC plus SA locking, which is

described as follows. The maximum rms phase error which can be tolerated without too many cycle slips is related to the average phase detector output versus ϕ , which is called the S -curve. For sideband aiding, the S -curve can be approximated (Ref. 2)

$$S(\phi) = \sqrt{2P} \left[\cos \Delta \sin \phi + \frac{\sin \Delta}{4} \sin^2 \phi \right] \quad (9)$$

The value of ϕ for which $S(\phi)$ is maximum can be found to be

$$\phi_{\max} = \cos^{-1} \left[\frac{-\cos \Delta + \sqrt{1 + \sin^2 \Delta}}{2 \sin \Delta} \right] \quad (10)$$

In a conventional PLL, acceptable cycle slipping and acquisition is usually achieved with $\sigma_\phi^2 \leq 0.2$ (loop SNR ≥ 7 dB). Since the corresponding S -curve peaks at $\phi_{\max} = \pi/2$, this roughly amounts to

$$\sigma_\phi \leq 0.285 \phi_{\max} \quad (11)$$

We use as this criterion for sideband aided loops with ϕ_{\max} according to Eq. (10). A numerical analysis is then carried out to maximize $[E_s/N_0]_{\text{ave}}$ from Eqs. (4) and (6) subject to the constraint of Eq. (11).

III. Results

The maximum (over Δ) average symbol SNR, $[E_s/N_0]_{\text{ave}}$, was evaluated for the various tracking methods as a function of the ratio of total signal power to noise in the loop bandwidth, $P/N_0 B_L$. The results are shown in Fig. 1 for E_s/N_0 of 0 dB, -3 dB, -10 dB, and -20 dB, respectively. Also shown on the curves are the modulation indexes which maximize the average SNR. Note that, using Eq. (7), the symbol rate for any point on the curves can be determined by:

$$R_s = \frac{1}{T} = \frac{P}{N_0 B_L} \frac{B_L \sin^2 \Delta}{E_s/N_0} \quad (12)$$

Thus, symbol rate, R_s , is almost proportional to the abscissa value in Fig. 1.

For the higher symbol SNRs, Figs. 1(a) and 1(b), the use of sideband aiding with optimized modulation index results in almost 100 percent efficient utilization of received power, except at very low symbol rates. There is little to gain by use of smoothing, except at low symbol rates. At $E_s/N_0 = -20$ dB, Fig. 1(d), sideband aiding does not improve performance very much, but smoothing helps significantly. Both SA and SM are significant at -10 dB, Fig. 1(c).

The benefits of sideband aiding relative to residual carrier tracking only are shown versus $P/N_0 B_L$, in Fig. 2, for the four different symbol SNRs. For the higher SNRs, 0 dB to -3 dB, the benefit is often greater than 1 dB and sometimes as much as 2.5 dB, at low $P/N_0 B_L$. The benefit is almost 0.4 dB at the highest $P/N_0 B_L$ shown, 500, which corresponds to a symbols rate of $500 B_L$ and $1000 B_L$ at symbol SNRs of 0 dB and -3 dB, respectively. At a symbol SNR of -10 dB, the maximum benefit of sideband aiding is approximately 0.65 dB, and occurs approximately at $P/N_0 B_L = 100$. At a symbol SNR of -20 dB, sideband aiding gains less than 0.1 dB for all $P/N_0 B_L$ shown.

The benefits of smoothing over sideband aiding are shown in Fig. 3. At the lowest $P/N_0 B_L$ shown, 20, the benefits are 0.38 dB, 0.37 dB, 0.29 dB and 0.25 dB at symbol SNRs of -20 dB, -10 dB, -3 dB and 0 dB, respectively. The corresponding symbol rates are $1100 B_L$, $124 B_L$, $33 B_L$ and $18 B_L$. Looking at the results from another viewpoint, smoothing gains at least 0.25 dB over sideband aiding at all symbol SNRs, provided that the symbol rate is low enough, i.e., less than $18 B_L$, $52 B_L$, $920 B_L$ and $1400 B_L$ for symbol SNRs of 0 dB, -3 dB, -10 dB and -20 dB.

IV. Discussion of Other Conditions

Results have been presented only for cases in which the modulation index is optimized, subject to a lock constraint, and in which sideband aiding is used. For any conditions, smoothing can reduce the carrier phase error variance, and hence the radio loss, by a factor of four. Thus, smoothing is a valuable tool whenever the radio loss is large. For example, suppose that sideband aiding is not used and the modulation index and loop bandwidth are such that the carrier loop SNR is 7 dB. Then the radio loss is approximately 0.97 dB without smoothing, and 0.22 dB with smoothing, a gain of 0.75 dB. On the other hand, if the modulation index for a spacecraft is always low enough so that the residual carrier loop SNR is high and the radio loss is low, then there is minimal potential benefit to either sideband aiding or smoothing.

Smoothing is also beneficial in reducing the effects of oscillator noise and spacecraft dynamics, as shown in Ref. 1. This might lead to reduction in the loop bandwidth, and thus further reduction in radio loss. The impacts on symbol SNR have not yet been evaluated.

V. Implementation, Cost and Remaining Problem

A block diagram for carrier smoothing is given in Ref. 1. It is estimated that this can be implemented with one special

purpose digital board, plus one single board computer, added to the Advanced Receiver. The cost is estimated at \$5K to \$10K per receiver.

The main problem remaining is the possible effect of nonlinearities in the phase detectors on the smoothing filter phase solution. This needs to be evaluated by simulation before a breadboard is developed.

VI. Conclusions

Smoothing is a useful tool for carrier tracking at low symbol SNRs and low data rates. For optimum modulation indexes and using sideband aiding, smoothing virtually eliminates radio loss and gains up to 0.4 dB in average effective symbol SNR. Larger gains in SNR are possible in cases in which sideband aiding is not used, or in which the modulation index is not optimized.

References

1. Pomalaza-Raez, C. A., and Hurd, W. J., "Improved Carrier Tracking by Smoothing Estimators," *TDA Progress Report 42-79*, pp. 96-106 November 15, 1984, Jet Propulsion Laboratory, Pasadena, Calif.; also in *IEEE Trans. Aerospace and Electronic Systems*, Vol. AES-21, No. 5, September 1985, pp. 610-618.
2. Sfeir, R., Aguirre, S. and Hurd, W. J., "Coherent Digital Demodulation of a Residual Carrier Signal Using IF Sampling," *TDA Progress Report 42-78*, pp. 135-142 August 15, 1984, Jet Propulsion Laboratory, Pasadena, Calif.

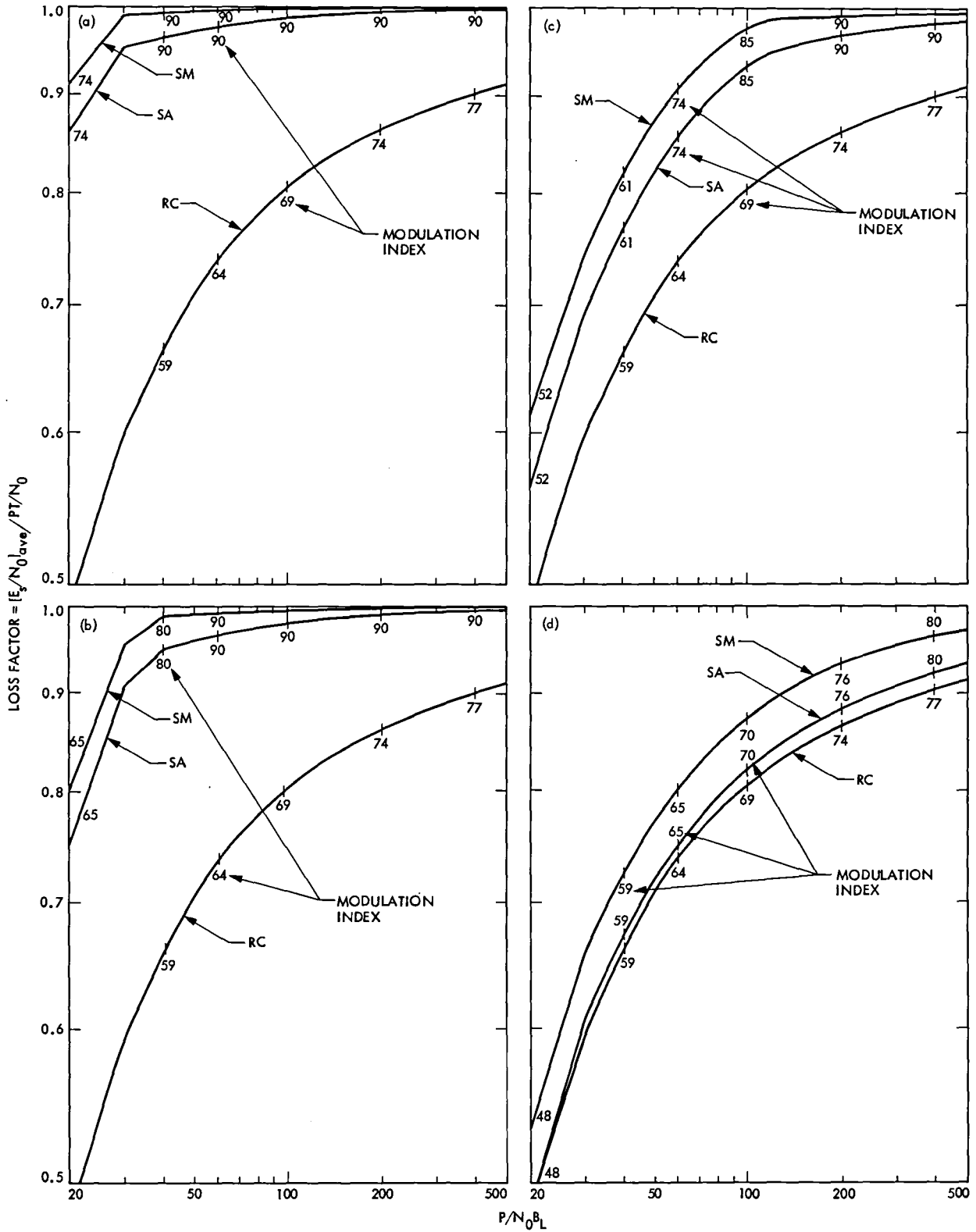


Fig. 1. Comparison of carrier phase tracking loss factor vs total power SNR in loop bandwidth. The symbol SNR is (a) 0 dB, (b) -3 dB, (c) -10 dB, and (d) -20 dB.

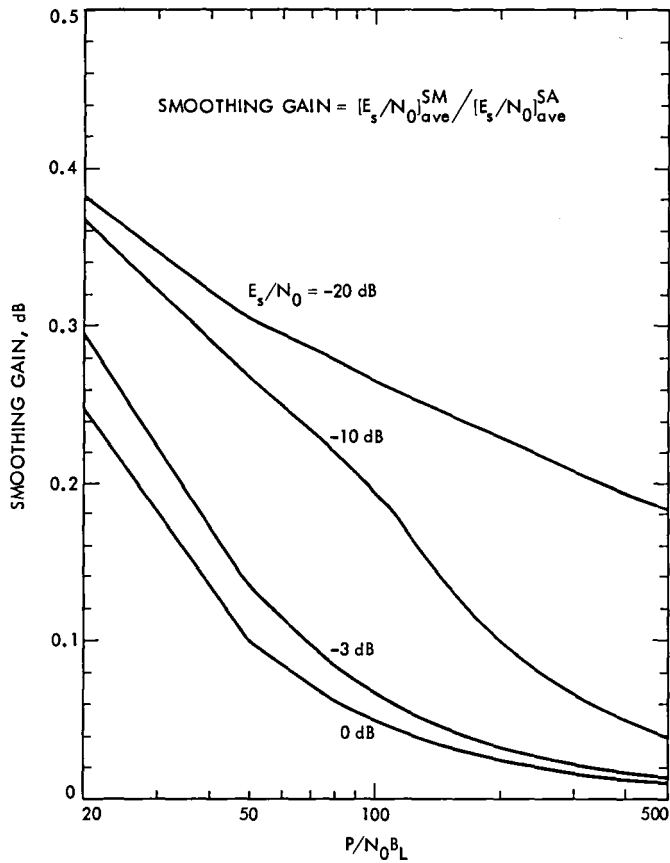


Fig. 2. The gain of sideband aiding over residual carrier tracking

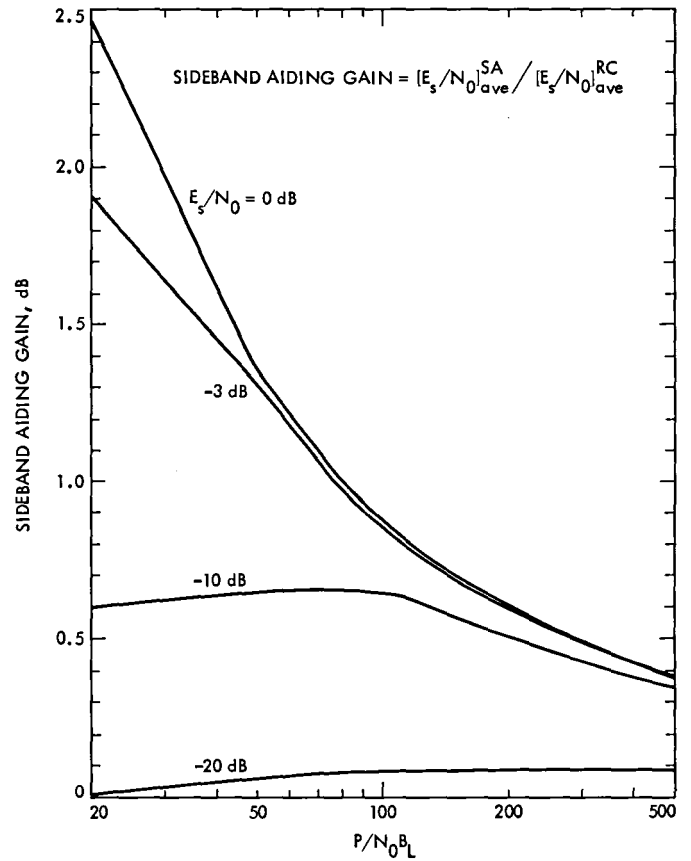


Fig. 3. The gain of smoothing over sideband aiding

A Class of Optimum Digital Phase Locked Loops for the DSN Advanced Receiver

R. Kumar¹ and W. J. Hurd

Communication Systems Research Section

This article presents a class of optimum digital filters for digital phase locked loops of the DSN Advanced Receiver. The filter minimizes a weighted combination of the variance of the random component of the phase error and the sum square of the deterministic dynamic component of phase error at the output of the numerically controlled oscillator (NCO). By varying the weighting coefficient over a suitable range of values, a wide set of filters are obtained such that, for any specified value of the equivalent loop-noise bandwidth, there corresponds a unique filter in this class. This filter thus has the property of having the best transient response over all possible filters of the same bandwidth and type. The optimum filters are also evaluated in terms of their gain margin for stability and their steady-state error performance.

I. Introduction

There has been an increasing interest (Refs. 1–4) in the study of digital phase locked loops. Such an interest emerges in part from the capability of digital technology which makes it possible to control the loop parameters accurately and even make these programmable and/or adaptive.

This article derives optimum filters for the digital phase locked loops for carrier, subcarrier and symbol timing recovery in the DSN Advanced Receiver. The loop is analyzed in the z-transform domain so as to arrive at a set of optimum digital filters for various input dynamics.

The filter minimizes a weighted sum of the variance of the random component of the phase error and the sum square of the deterministic dynamic component of the phase error at the output of the numerically controlled oscillator (NCO). By varying the weighting coefficient over a suitable range of values, a wide set of filters is obtained such that, for any specified value of the equivalent loop noise bandwidth, there corresponds a unique filter in this class. This filter thus has the property of having the best transient response over all possible filters of the same bandwidth and type.

Three specific filter sets optimum for a phase step, phase ramp and the phase acceleration inputs are considered in some detail. The corresponding optimum filters are type I, II, and III, respectively. Due to the specific optimization index under

¹Also with the faculty of Electrical Engineering Department at California State University, Long Beach

consideration, the filter design also ensures that the deterministic dynamic component of the steady-state phase error is zero for all input phase dynamics with the order of the highest order nonzero derivative term smaller than the filter type.

The open loop transfer function (the product of the transfer functions of the filter and the NCO) possesses a multiple pole at $z = 1$, with its multiplicity equal to the order of the nonzero phase derivative plus one. The type of the filter is the multiplicity of the pole at $z = 1$.

II. Optimum Filter

Figure 1 depicts the block diagram of the basic digital phase locked loop (DPLL) and the z domain model of its linearized version. In the figure, $F(z)$ represents the transfer function of the digital filter and $N(z)$ is the transfer function of the NCO and the transport lag in the loop, due to the specific implementation. We use the $N(z)$ appropriate for the DSN Advanced Receiver (Ref. 2),

$$N(z) = \frac{KT(z+1)}{2z^2(z-1)} \quad (1)$$

where K is some constant and T is the sampling period. Such an NCO transfer function results because of the limited loop update rate $1/T$. Similar results are possible for other $N(z)$.

The input noise $\xi(t)$ is assumed to be a zero mean white Gaussian noise with two-sided spectral density $N_0/2$. The noise $\bar{n}_i(t)$ is the real part of the complex envelope of $\xi(t)$ and is also white Gaussian with spectral density N_0 .

In Fig. 1(b) is given a linearized and discrete-time version of the DPLL wherein $\{n_i(k)\}$ represents a zero mean white Gaussian sequence with variance (N_0/A^2T) . In this model $\hat{\Theta}$ may be decomposed into a sum of $\psi(k)$ and $n_0(k)$ where $\psi(k)$ represents the deterministic part of $\hat{\Theta}$ while $n_0(k)$ is the stochastic component. The optimum filter is derived by minimization of the following index (Refs. 4, 5, 6):

$$Q = E[n_0^2(k)] + \lambda \sum_k e^2(k) \quad (2)$$

with $e(k) = \Theta(k) - \psi(k)$ denoting the deterministic component of the phase error. The parameter λ is selected from the consideration of the loop noise bandwidth and the transient performance of the loop, which are functions of λ for the optimum solution. In the sequel, the parameter λ is expressed in terms of \bar{N}_0 , T and an appropriate normalized parameter. Thus, for example, when the input phase is a step function of

time, $\lambda = \bar{N}_0 T q$ where q is the normalized parameter. The first term in the index Q can be expressed in terms of the loop filter $F(z)$ as

$$E[n_0^2(k)] = \frac{1}{2\pi j} \int_{\Gamma} \Phi_{n_0 n_0}(z) \frac{dz}{z} \quad (3)$$

with $\Phi_{n_0 n_0}(z)$ denoting the noise spectral density of $n_0(k)$. Now

$$\Phi_{n_0 n_0}(z) = H(z)H(z^{-1})\Phi_{n_i n_i}(z);$$

$$H(z) = \frac{F(z)N(z)}{1 + F(z)N(z)}$$

$$\Phi_{n_i n_i}(z) = \frac{N_0}{A^2 T} = \frac{\bar{N}_0}{T}; \quad \bar{N}_0 = N_0/A^2$$

Similarly, the second term in Eq. (2) can be evaluated in terms of the following contour integral (Γ denotes the unit circle),

$$\sum_k e^2(k) = \frac{1}{2\pi j} \int_{\Gamma} E(z)E(z^{-1}) \frac{dz}{z}; \quad E(z) = Z[e(k)]$$

where Z denotes z -transform. As $E(z) = (1 - H(z))\Theta(z)$ (where $\Theta(z)$ denotes z -transform of $\theta(k)$), an equivalent expression for the sum of squared errors is

$$\sum_k e^2(k) = \frac{1}{2\pi j} \int_{\Gamma} (1 - H(z))(1 - H(z^{-1}))\Phi_{\theta\theta}(z) \frac{dz}{z};$$

$$\Phi_{\theta\theta}(z) = \Theta(z)\Theta(z^{-1}) \quad (4)$$

From Eqs. (3) and (4), the optimization index Q may be rewritten as,

$$Q = \frac{1}{2\pi j} \int_{\Gamma} [\lambda \Phi_{\theta\theta}(z) + P(z)W(z)W(z^{-1}) - \lambda W(z)N(z)\Phi_{\theta\theta}(z) - \lambda W(z^{-1})N(z^{-1})\Phi_{\theta\theta}(z)] \frac{dz}{z};$$

$$H(z) = W(z)N(z); \quad P(z) = [\bar{N}_0 T + \lambda \Phi_{\theta\theta}(z)]N(z)N(z^{-1}) \quad (5)$$

The optimum solution for $W(z)$ and thus $H(z)$ can then be obtained from the spectral factorization (Refs. 4, 5, 6) as

$$W_0(z) = \frac{z \left[\frac{\lambda N(z^{-1}) \Phi_{\theta\theta}(z)}{z P^-(z)} \right]_+}{P^+(z)}; \quad P(z) = P^+(z) P^-(z) \quad (6a)$$

In the above $P^+(z)$ denotes that factor of $P(z)$ which has all its poles and zeros inside the unit circle, and $[C(z)]_+$, for any rational function $C(z)$, represents that part of the partial fraction expansion of $C(z)$ whose poles are inside the unit circle.

The optimum loop filter $F(z)$ can then be obtained from $W_0(z)$ as

$$F(z) = \frac{W_0(z)}{1 - W_0(z) N(z)} \quad (6b)$$

III. Performance Considerations of Optimum Filters

The optimum filters of the previous section are evaluated in terms of various parameters of interest. The most significant parameters are the loop noise bandwidth, transient error performance, stability analysis and gain margin, and the steady state phase error for an input having a nonzero derivative of order equal to the filter type.

A. Loop Noise Bandwidth

In terms of the filter performance, it is of interest to evaluate the phase noise variance $E[n_0^2(k)]$ as the weighting parameter λ or its normalized version is varied. This variance is given by Eq. (3) and is usually expressed in terms of a normalized parameter B given by

$$B = \frac{1}{2\pi j} \int_{\Gamma} H(z) H(z^{-1}) \frac{dz}{z} \quad (7)$$

The parameter B is termed normalized loop noise bandwidth and in terms of B , the oscillator phase noise variance is given by $(N_0 B/T)$. In the PLL literature, a loop noise bandwidth B_L is also defined and equals (B/T) Hz.

B. Computation of B

From Ref. 7, the integral for B can be expressed in terms of the coefficients of $H(z)$. Let

$$H(z) = \frac{b_0 z^n + b_1 z^{n-1} + \cdots + b_n}{a_0 z^n + a_1 z^{n-1} + \cdots + a_n}; \quad a_0 \neq 0$$

where the coefficients a_r , $0 < r \leq n$ and b_r , $0 \leq r \leq n$ are not necessarily nonzero. Then

$$\Omega v = u \quad (8)$$

where Ω is an $n \times n$ matrix and v and u are n -dimensional vectors with the elements of Ω and u being functions of the coefficients a 's and b 's. Moreover, the first element of v equals $a_0 B$; thus, B can be computed by solving the set of n linear equations (8). In this regards, it is of interest to note that the matrix Ω can be decomposed as,

$$\Omega = \Omega_{UT} + \Omega_{UH} - \bar{\Omega}$$

where Ω_{UT} and Ω_{UH} are upper triangular Toeplitz and "upper triangular" Hankel matrices respectively, i.e.,

$$\Omega_{UT} = \begin{bmatrix} a_0 & a_1 & a_2 & a_3 & \cdots & a_n \\ 0 & a_0 & a_1 & a_2 & \cdots & a_{n-1} \\ 0 & 0 & a_0 & a_1 & \cdots & a_{n-2} \\ \vdots & & & & & \\ 0 & 0 & 0 & 0 & \cdots & a_0 \end{bmatrix} \quad (9)$$

$$\Omega_{UH} = \begin{bmatrix} a_0 & a_1 & a_2 & a_3 & \cdots & a_n \\ a_1 & a_2 & a_3 & a_4 & \cdots & 0 \\ a_2 & a_3 & a_4 & a_5 & \cdots & 0 \\ \vdots & & & & & \\ a_n & 0 & 0 & 0 & \cdots & 0 \end{bmatrix}$$

and $\bar{\Omega}$ has its first row equal to $[a_0 \ a_1 \ a_2 \ \cdots \ a_n]$ with the remaining rows identically equal to zero. The elements of the vector u are expressed in terms of the autocorrelation function of the sequence b_i , i.e.,

$$u^T = [R_{bb}(0) \ 2R_{bb}(1) \ \cdots \ 2R_{bb}(n)]; \quad R_{bb}(k) \triangleq \sum_{i=0}^n b_i b_{i+k}$$

With a slight modification, the solution for B can be expressed in the following form (possessing a rich structure for computational purposes),

$$B = \frac{2}{a_0} \times \text{first element of } \{(\Omega_{UT} + \Omega_{UH})^{-1} \bar{u}\} \quad (10)$$

$$\bar{u} \triangleq [R_{bb}(0)R_{bb}(1) \cdots R_{bb}(n)]^T$$

The specific structure of the matrix in Eq. (10) can be exploited in the fast computation of the bandwidth. Thus, when the dimension of the matrix $(\Omega_{UT} + \Omega_{UH})$ is high, sophisticated algorithms requiring order $n \log_2 n \log_2 n$ operations can be used for solving Eq. (10).

C. Transient Performance

The transient performance of the filter is evaluated in terms of the index

$$\sum_k e^2(k)$$

where $e(k)$ is the deterministic component of the phase error. A smaller value of this index also implies a faster settling of the dynamic component of the phase error to zero. In principle, the value of this index can be reduced to an arbitrarily small value by choosing the value of the parameter λ sufficiently high. However, this would have the adverse effect of increasing the loop noise bandwidth and degrading the noise performance of the loop. The objective is thus to arrive at a compromise solution, as in the following section.

D. Stability Analysis and Gain Margin

In the preceding derivation of the optimum filter, it is assumed that the input signal amplitude A is constant (implicitly assumed to be 1). If the gain A is known then it can be taken into account by dividing the constant of $F(z)$ by A . However, in actual practice, there may be some uncertainty associated with A or A may be a slowly varying function of time. In such situations it is essential that the loop remain stable for a sufficiently large range of A .

In the subsequent section, using Jury's criterion, the upper and lower gain margins are evaluated for the closed loop stability of the loop. Such an evaluation is particularly important, since the filter derivation does not explicitly take into account such an index.

E. Steady State Error Due to Higher Order Dynamics

In practice, the signal may possess a component with the order of its highest nonzero derivative equal to or greater than

the type of the filter. For example, the phase locked loop with a type III filter may have a nonzero phase jerk at its input. In this case, the deterministic component of the steady state phase error is nonzero and it may be of interest to evaluate the derived filter in terms of such steady state phase error ϕ_{ss} .

For analog phase locked loops (Refs. 5, 6) ϕ_{ss} is inversely proportional to B_L^i , where i denotes the filter type and B_L is the actual loop noise bandwidth in Hz. As $B_L = B/T$, a suitably normalized parameter $C_{\phi i}$ is given by $C_{\phi i} = (\phi_{ss})^{1/i} B/T$, where the order of the highest nonzero derivative of the input signal is assumed to be equal to i . The variation of $C_{\phi i}$ is studied in a subsequent section, as the normalized loop noise bandwidth B varies over the range of interest.

IV. Performance Evaluation of the Specific Filter Classes

In the following, we evaluate the parameters of the optimum filter $W_0(z)$ and the loop filter $F(z)$ as the parameter λ is varied for the NCO transfer function given by Eq. (1). The design of three specific classes of filters is considered to correspond to a phase step, frequency step and a frequency ramp input to the PLL. These filters are type I, II and III, respectively. For each class of filters, various design curves are obtained for the parameters and performance including the normalized design parameter, the optimum loop gain, the pole and zero locations of $F(z)$, the upper and lower gain margin for stability, and the steady state error constant $C_{\phi i}$, as functions of the normalized loop noise bandwidth, B .

A. Optimum Filter for Phase Step (Type I)

In this case, the input phase function $\theta(t) = u(t)$, the unit step function, and thus $\Theta(z) = (1 - z^{-1})^{-1}$. As shown in the appendix, the optimum filter $W_0(z)$ in this case is given by

$$W_0(z) = \frac{a + b + c}{KT} \left[\frac{(z - 1)z}{(az^2 + bz + c)} \right]$$

where a , b and c must satisfy the set of following equations

$$\left. \begin{aligned} ac &= -1 \\ ab + bc &= q; \quad q = \lambda/\bar{N}_0 T \\ a^2 + b^2 + c^2 &= 2(1 + q) \end{aligned} \right\} \quad (11)$$

The optimum loop filter $F(z)$ can then be obtained from $W_0(z)$ in a straightforward manner and after a few manipulations can be written as

$$F(z) = \frac{2(a+b+c)}{KT}$$

$$\times \left\{ \frac{z^2}{2az^2 + 2(b+a)z + (a+b+c)} \right\} \quad (12)$$

From the computations of the pole-zero locations of $F(z)$, it turns out that the filter $F(z)$ possesses a pole at $z = -1$, i.e.,

$$az^2 + bz + c = (\bar{a}z + \bar{b})(z + 1) \quad (13a)$$

where \bar{a} , \bar{b} satisfy the following

$$\begin{aligned} \bar{a}\bar{b} &= -1 \\ \bar{a}^2 + \bar{b}^2 &= 2 + q \end{aligned} \quad (13b)$$

Even though the pole at -1 is optimal, it is desirable to avoid a pole-zero cancellation at $z = -1$ so as to avoid the practical problem of imperfect pole cancellation and hence instability. Therefore, the filter coefficients are modified so as to shift the pole at the unit circle to $z = -0.9$ (say). Thus the factor $(z + 1)$ in Eq. (13a) is replaced by $(z + 0.9)$ with corresponding changes in Eq. (13b).

From Eqs. (1) and (12), the open loop transfer function for the type I PLL is given by,

$$F(z)N(z) = \bar{K}_{\text{opt}} \frac{(z + 1)}{(z - 1) \left\{ z^2 + \left(\frac{b+a}{a} \right) z + \left(\frac{a+b+c}{2a} \right) \right\}};$$

$$\bar{K}_{\text{opt}} = 0.95 \left(\frac{\bar{a} + \bar{b}}{\bar{a}} \right) \quad (14)$$

Moreover, the corresponding closed loop transfer function of the PLL is given by

$$H(z) = \frac{a+b+c}{2} \frac{z+1}{z(az^2 + bz + c)}$$

In this case, the normalized loop noise bandwidth B as evaluated from Eq. (7) has the following closed-form expression.

$$B = \frac{a+b+c}{2(a-c)} \quad (15)$$

1. Filter parameters. In the practice of phase locked receiver design, it is customary to treat the normalized loop noise bandwidth B as an independent parameter and then express various other loop parameters and the performance indices as functions of B . In the case of digital PLL, closed form expressions of this form cannot possibly be derived. Thus to obtain such relationships numerically, we evaluate \bar{a} , \bar{b} first as a function of q from the modified version of Eq. (13b). (Actually for the case of computations, b and q are calculated as a function of \bar{a} which can have values strictly greater than 1.) In this case, from the modified version of Eq. (13a), we also note that $a = \bar{a}$, $b = \bar{b} + 0.9\bar{a}$, $c = 0.9\bar{b}$.

In Fig. 2 is plotted the value of the parameter q which is required to obtain the normalized bandwidth B for the type I PLL. As is apparent from the figure, corresponding to the range of B between 0.001 and 0.95, the parameter q varies over approximately 7 orders of magnitudes (10^{-5} to 10^2). To design an optimum type I filter of bandwidth B , one can determine the loop gain from Fig. 3 and the pole locations from Fig. 4.

2. Stability. From stability theory, the closed loop system corresponding to the open loop transfer function of Eq. (14) remains stable if \bar{K}_{opt} is replaced by any positive $\bar{K} \leq \bar{K}_{\text{opt}}$. However, if \bar{K} is greater than \bar{K}_{opt} , then the closed loop system may become unstable. The maximum value of \bar{K} for closed loop stability, denoted \bar{K}_{max} is expressed in terms of the upper gain margin $G_u \triangleq 20 \log (\bar{K}_{\text{max}} / \bar{K}_{\text{opt}})$ (dB) and is plotted versus B in Fig. 5. As may be inferred from the figure, for a normalized loop noise bandwidth B less than or equal to 0.1, the optimum filter yields an upper gain margin of 16 dB or higher.

In Fig. 6(a) is plotted the root locus diagram for a typical filter of type 1, corresponding to a nominal bandwidth of 0.0965. As the loop gain is varied around the optimum gain \bar{K}_{opt} , the normalized bandwidth B also varies around its nominal value as depicted in Fig. 6(b).

3. Transient error performance. Figure 7 plots the integral square error, given by Eq. (4) as a function of normalized bandwidth. Over the bandwidth range of interest, the error varies approximately by two orders of magnitude. Note that some suboptimality is introduced due to modification of the pole at $z = -1$.

4. **Steady state error due to phase ramp.** The steady state phase error is given by

$$\phi_{ss} = \lim_{z \rightarrow 1} \left(\frac{z-1}{z} \right) \frac{1}{1+F(z)N(z)} \Theta(z)$$

$$\Theta(z) = T \frac{z}{(z-1)^2}$$

or,

$$\phi_{ss} = \lim_{z \rightarrow 1} \frac{T}{(z-1)F(z)N(z)}$$

For the optimum filter (Eq. [14]) ϕ_{ss} is given by

$$\phi_{ss} = T \frac{5a+3b+c}{2(a+b+c)} = C_{\phi 1}/B_L$$

Figure 8 plots the normalized value of ϕ_{ss} i.e., $C_{\phi 1}$ as a function of B . The radian phase error for radian center frequency ω_0 and the spacecraft velocity v is $(\omega_0 v/c)$ ($C_{\phi 1}/B_L$) where c is the speed of light.

In Fig. 9 is plotted the location of the closed loop pole as a function of B . As can be inferred from the figure, increasing the value of B results in the movement of the pole towards the origin, thus resulting in faster transient response.

5. **Type I summary.** Table 1 lists some of the parameters of six typical filter corresponding to different values of q for quick reference.

B. Optimum Filter for Frequency Step (Type II)

In this case, the input phase function $\theta(t) = t u(t)$ and thus

$$\Theta(z) = \frac{Tz}{(z-1)^2}$$

The optimum filter $W_0(z)$ is derived in the appendix and is given by

$$W_0(z) = \frac{2}{KT} \frac{(h_0 z - h_1)(z-1)z}{4(az^3 + bz^2 + cz + d)}$$

where

$$h_0 = (7a + 5b + 3c + d)$$

$$h_1 = (5a + 3b + c - d)$$

and the parameters a, b, c, d are the solutions of the following set of nonlinear equations.

$$\left. \begin{aligned} ad &= 1 \\ ac + bd &= -2 \\ ab + bc + cd &= r - 1 \\ a^2 + b^2 + c^2 + d^2 &= 2r + 4 \\ r &= \frac{\lambda T}{N_0} \end{aligned} \right\} \quad (16)$$

Using the identity that $(a + b + c + d)^2 = 4r$, and after a few algebraic manipulations, the corresponding loop filter is given by

$$F(z) = \frac{2}{KT} \times \frac{(h_0 z - h_1)z^2}{\{4az^2 + (8a + 4b)z + (5a + 3b + c - d)\}(z-1)} \quad (17)$$

As in the case of phase-step, there is a pole-zero cancellation at $z = -1$ in the filter $F(z)$, i.e., the coefficients a, b, c, d satisfy the following identity

$$az^3 + bz^2 + cz + d = (z+1)(\bar{a}z^2 + \bar{b}z + \bar{c})$$

where \bar{a}, \bar{b} , and \bar{c} satisfy the following set of equations

$$\left. \begin{aligned} \bar{a}\bar{c} &= 1 \\ \bar{a}\bar{b} + \bar{b}\bar{c} &= -4 \\ \bar{a}^2 + \bar{b}^2 + \bar{c}^2 &= 6 + r \end{aligned} \right\} \quad (18)$$

The modified coefficients a, b, c, d are then obtained according to the following relation

$$(az^3 + bz^2 + cz + d) = (z + 0.9)(\bar{a}z^2 + \bar{b}z + \bar{c})$$

1. **Filter parameters.** Figure 10 plots the normalized bandwidth B versus the optimizations parameter r . Figures 11, 12, and 13 plot the loop gain and those zeros and poles of the optimum filter whose locations depend upon B . A comparison of these plots with the corresponding plots of type I filter indicates that for low values of B , the pole locations are nearly the same for both types. Similar remarks apply with respect to loop gain.

2. **Stability.** The upper and lower gain margins are plotted versus B in Fig. 14. An upper gain margin of 10 dB or higher is obtained for all B less than 0.5, and the lower gain margin is very large for all B less than 1.

3. **Transient response.** The transient response is given in Fig. 15. Compared to the type I filter, the transient error varies over a much higher range – approximately six orders of magnitude.

4. **Steady state error to frequency ramp.** In Fig. 16 is plotted the normalized steady state phase error constant $C_{\phi 2} = (\phi_{ss}^{1/2} B/T)$ for a unit frequency ramp input to the PLL. A limiting value of just over 1 is approached for B less than 0.1. The radian phase error for radian center frequency ω_0 is $(\omega_0 a/c) (C_{\phi 2}/B_L)^2$ where a is acceleration (in m/s^2 for c in m/s).

C. Optimum Filter for Frequency Ramp (Type III)

In this case

$$\Theta(z) = \frac{T^2 z(z+1)}{(z-1)^3}$$

and the optimum filter $W_0(z)$ is given by (the details similar to the first two cases and omitted):

$$W_0(z) = \frac{2}{KT} \frac{\{\tilde{C}z^2 + (\tilde{B} - 2\tilde{C})z + (\tilde{A} + \tilde{C} - \tilde{B})\}z(z-1)}{16(az^4 + bz^3 + cz^2 + dz + e)} \quad (19)$$

and the loop filter transfer function $F(z)$ is given by

$$F(z) = \frac{2}{KT} \times \frac{\{\tilde{C}z^2 + (\tilde{B} - 2\tilde{C})z + (\tilde{A} + \tilde{C} - \tilde{B})\}z^2}{(z-1)^2 \{16az^2 + 16(3a+b)z + [16(6a+3b+c) - \tilde{C}]\}}$$

where

$$\left. \begin{aligned} \tilde{A} &= 8(a+b+c+d+e) \\ \tilde{B} &= 4(9a+7b+5c+3d+e) \\ \tilde{C} &= 2 \left\{ Q_2 - \frac{2\bar{Q}_1(b+2c+3d+4e)}{(a+b+c+d+e)} \right\} \\ \bar{Q}_1 &= (9a+7b+5c+3d+e) \\ \bar{Q}_2 &= (33a+33b+29c+21d+9e) \end{aligned} \right\} \quad (20)$$

The coefficients a, b, c, d and e are obtained by equating the coefficients of various powers of z on both sides of

$$az^4 + bz^3 + cz^2 + dz + e = (z+0.9)(\bar{a}z^3 + \bar{b}z^2 + \bar{c}z + \bar{d}) \quad (21)$$

and $\bar{a}, \bar{b}, \bar{c}, \bar{d}$ are obtained as a solution of the

$$\left. \begin{aligned} \bar{a}\bar{d} &= -1 \\ \bar{a}\bar{c} + \bar{b}\bar{d} &= 6 \\ \bar{a}\bar{b} + \bar{b}\bar{c} + \bar{c}\bar{d} &= -(15-s) \\ \bar{a}^2 + \bar{b}^2 + \bar{c}^2 + \bar{d}^2 &= 20 + 2s; \quad s = \frac{\lambda T^3}{N_0} \end{aligned} \right\} \quad (22)$$

1. **Parameters of type III filter.** The parameters of the optimum filter, i.e., the loop gain, location of zeros and poles (those depending upon B) are quite close to those for type I and II filters for $B \leq 0.5$. For $B > 0.5$ however, there is significant difference in the location of poles. For space limitations the plots of these parameters versus B are omitted.

2. **Performance.** The performance indices of the filters are plotted in Figs. 17 through 19. As may be inferred from Fig. 17, both upper and lower gain margins of 10 dB or higher are achieved for normal regions of operation. The dynamic error (Fig. 18) has an extremely large range (about 12 orders of magnitude) for the range of B of interest. This suggests a programmable implementation of the PLL for fast acquisition as is discussed below. In Fig. 19 is plotted the normalized steady state phase error constant $C_{\phi 3} = \phi_{ss}^{1/3} B/T$. For low values of B , this has a value of approximately 1.5. Radian phase error is $(\omega_0 j/c) (C_{\phi 3}/B_L)^3$ where j is jerk (m/s^3).

V. Programmable and Adaptive Implementations

From the performance analysis of the optimum filters the following adaptive implementations are suggested.

The phase noise variance at the output of the NCO is given by $(N_0 B/A^2 T)$. Thus in those situations where (A^2/N_0) (the input signal power to noise spectral density) is slowly varying with time, in order to maintain the phase noise variance within specified limit, say V_{\max} , a value is obtained from $B = V_{\max} T(A^2/N_0)$. Then from the plots of the filter parameters (stored in the memory in the form of tables versus B) an optimum filter corresponding to B is obtained. This filter then also has the best transient response consistent with the desired phase variance.

Because the transient error varies over an extremely large range (about 12 orders of magnitude for type III filter), a rapid acquisition can be attained in the following manner. Starting with a sufficiently high value of B (thus very rapid settling of the loop), the loop parameters are adjusted, at an interval of several times the dominant time constant of the filter, to the values corresponding to successive lower values of B until the desired value of B is achieved.

As the filter poles location are not sensitive to the input dynamics, a serial implementation of the filter is suggested. Acquisition can be accomplished with a type II loop, which has a good transient response, and then the type can be increased by type III by adding a pole at $z = 1$. The feasibility of acquisition with the type II loop depends upon an estimate (upper bound) of the magnitude of the acceleration.

VI. Conclusions

The performance of a class of optimum filters for three different input phase dynamics has been evaluated. The filters achieve minimum transient error for any loop bandwidth. The optimum filter is such that the open loop transfer func-

tion has multiple poles at $z = 1$ with the multiplicity m equal to one plus the number of nonzero derivatives ($t > 0$) terms of the input. In addition the filter has an optimum compensator with its denominator polynomial of degree 2 and the degree of numerator polynomial equal to m .

The parameters of the optimum compensator have been obtained as a function of an optimizing parameter q . Increasing the value of q has the effect of weighing more heavily the deterministic component of the phase error. This has the effect of achieving smaller transient error by increasing the loop gain, placing the open-loop poles closer to the unit circle and the filter zeros close to the origin. Alternatively, higher values of q bring the poles of the closed-loop system closer to the origin. Such a behavior of the system may be very desirable during the acquisition phase of the loop. Thus, in an adaptive implementation, a loop filter corresponding to high values of q may be used during the acquisition phase. During the tracking phase then a filter corresponding to a lower value of q may be switched in.

The filters designed on this basis also have good margin against possible variations in the received signal power level.

References

1. Lindsey, W. C., and Chie, C. M., "A Survey of Digital Phase-Locked Loops," *Proceedings of the IEEE*, Vol. 69, No. 4, pp. 410-431, April 1981.
2. Aguirre, S., and Hurd, W. J., "Design and Performance of a Sampled Data Loops for Subcarrier and Carrier Tracking," *TDA Progress Report 42-79*, pp. 81-95, Jet Propulsion Laboratory, Pasadena, CA, July-September 1984.
3. Tausworthe, R. C., Theory and Practical Design of Phase-Locked Receivers, *JPL Technical Report No. 32-819*, Jet Propulsion Laboratory, Pasadena, CA, April 27, 1971.
4. Gupta, S. C., "On Optimum Digital Phase-Locked Loop," *IEEE Transactions on Communication Technology*, Vol. 16, No. 2, pp. 340-344, April 1968.
5. Jaffe, R. and Rechtin, E., "Design and Performance of Phase-Lock Circuits Capable of Near-Optimum Performance Over a Wide Range of Input Signal and Noise Level," *IRE Trans. Information Theory*, Vol. IT-1, pp. 66-76, March 1955.
6. Chang, S. S. L., *Synthesis of Optimum Control Systems*, New York: McGraw-Hill, 1961.
7. Jury, E. I., *Theory and Application of the z-Transform Method*, New York: John Wiley & Sons, 1964.

Table 1. Performance of optimum filters (type I) corresponding to different values of the parameter q

q	Optimum Loop Gain \bar{K} Optimum	Optimum Loop Noise BW	Loop Transfer Function	Range of \bar{K} for Stability	Gain Margin in dB	Bandwidth Over ± 12 dB Gain Variation
0.0364	0.1649	0.0965	$\frac{z+1}{(z-1)(z^2+1.0736z+0.1649)}$	0.002 – 1.091	–38, 16.4	0.018 – 0.945
0.1344	0.2903	0.1786	$\frac{z+1}{(z-1)(z^2+1.2056z+0.2903)}$	0.002 – 1.204	–43, 12.35	0.034 – 1.2
0.4702	0.4653	0.3189	$\frac{z+1}{(z-1)(z^2+1.390z+0.4653)}$	0.002 – 1.380	–47, 9.66	0.11 – 1.35*
1.548	0.6568	0.5140	$\frac{z+1}{(z-1)(z^2+1.5914z+0.6568)}$	0.002 – 1.60	–50, 7.73	0.17 – 3.8*
10.33	0.8724	0.8127	$\frac{z+1}{(z-1)(z^2+1.8184z+0.8724)}$	0.002 – 1.805	–52.8, 6.31	0.25 – 18.*
180.2	0.9448	0.9401	$\frac{z+1}{(z-1)(z^2+1.8945z+0.9448)}$	0.002 – 1.88	–53.5, 6.0	0.30 – 38.*

*Entries correspond to ± 6 dB gain variation.

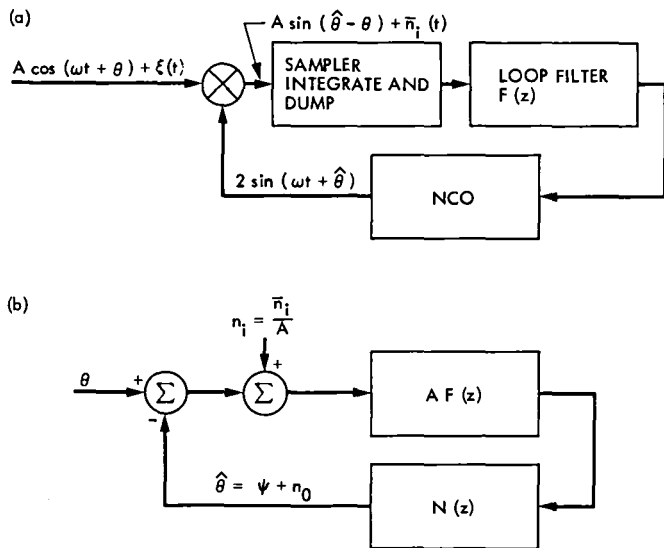


Fig. 1. Digital phase locked loop: (a) basic phase locked loop, (b) linearized model

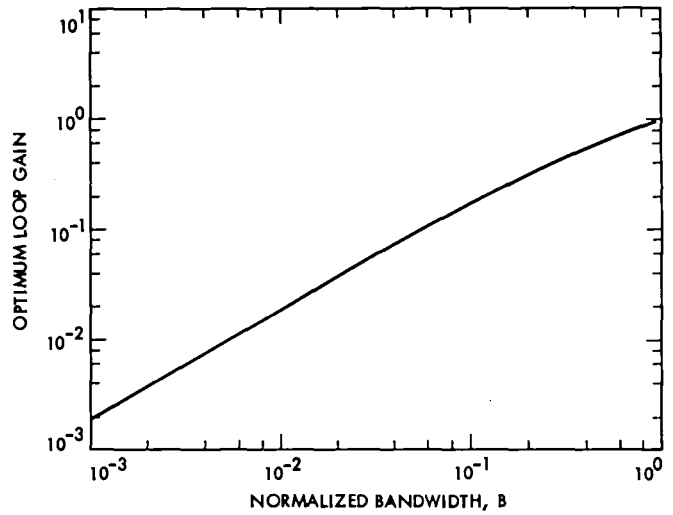


Fig. 3. Optimum loop gain vs normalized bandwidth for type I filter

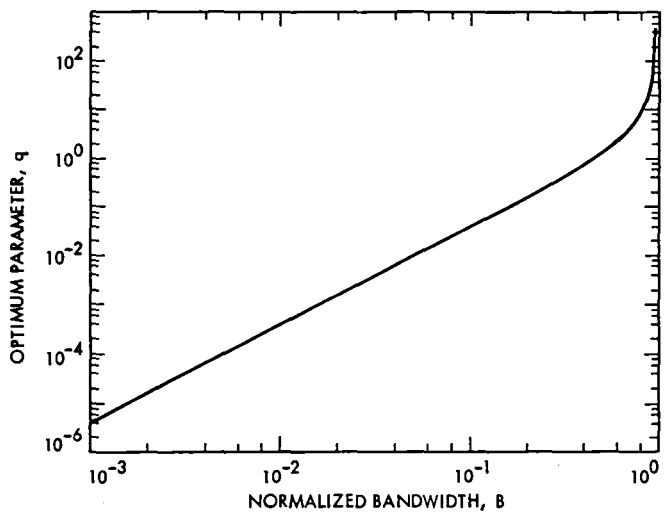


Fig. 2. Optimum parameter q for loop filter of type I

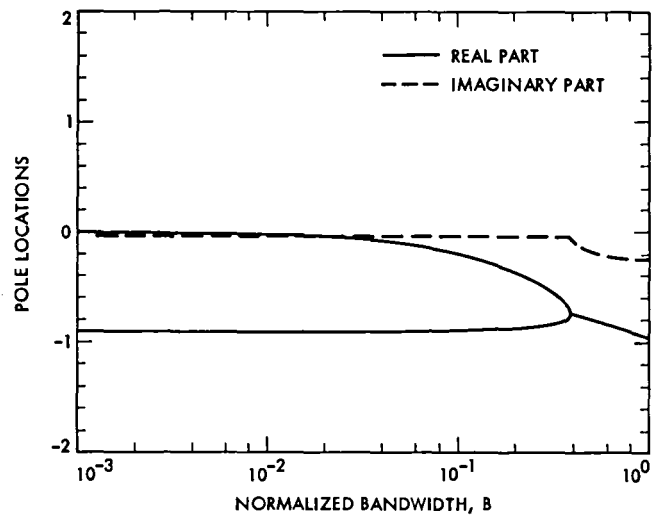


Fig. 4. Pole locations of the optimum loop filter of type I

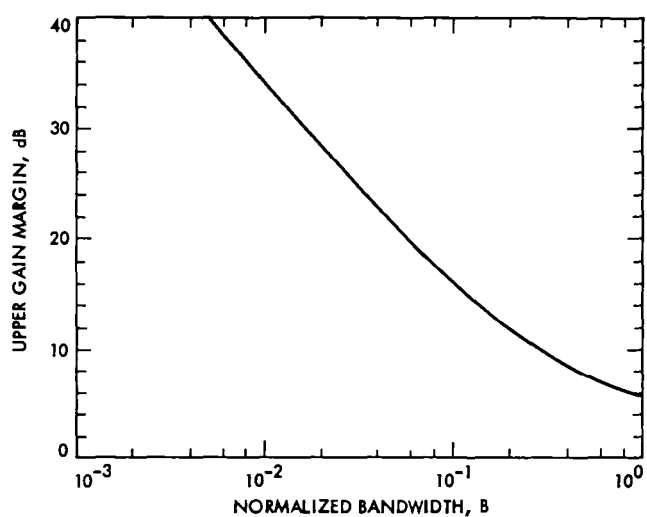


Fig. 5. Upper gain margin vs normalized bandwidth

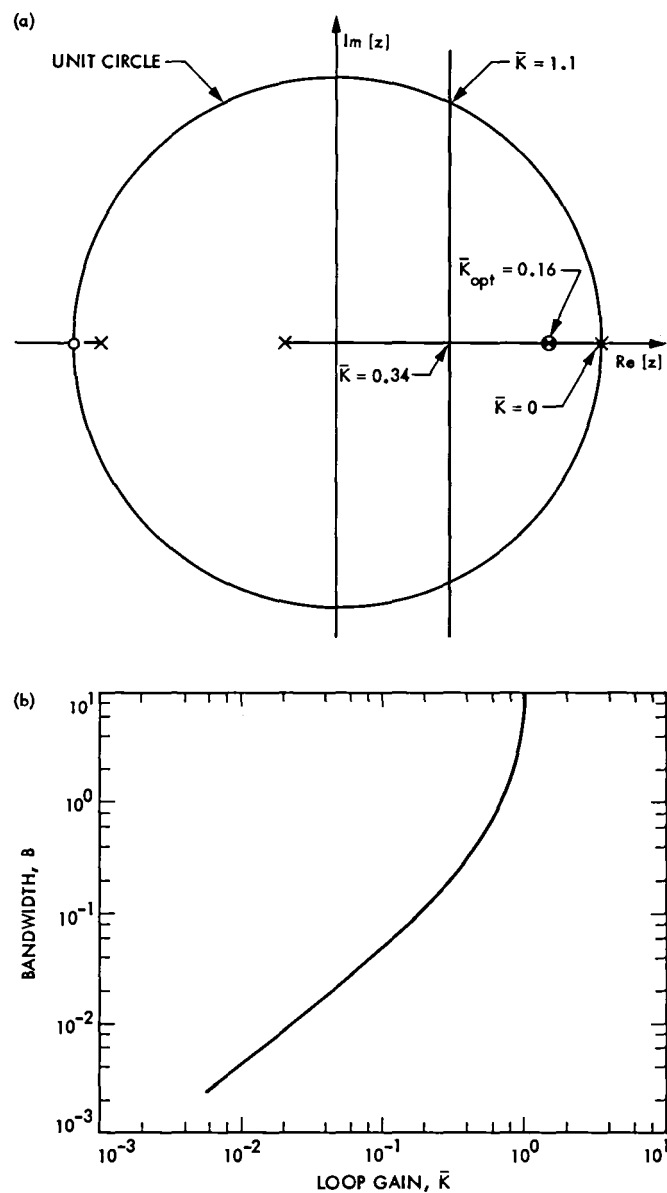


Fig. 6. Performance of type I filter vs loop gain: (a) root locus diagram for type I PLL, (b) loop noise bandwidth vs loop gain \bar{K} for type I filter (nominal BW = 0.0965)

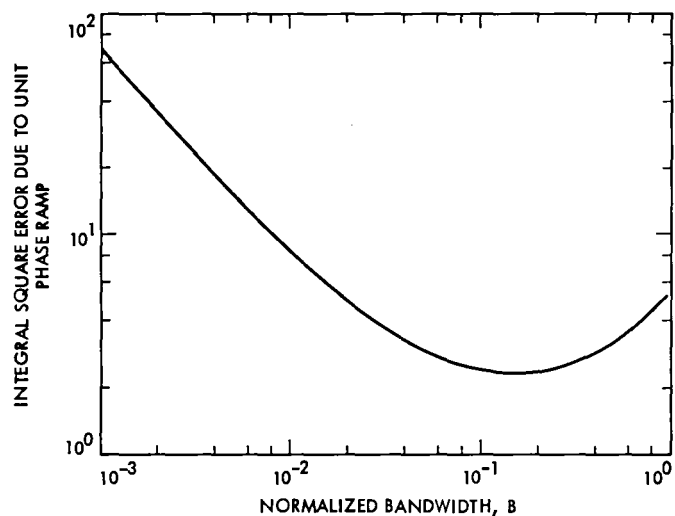


Fig. 7. Transient error performance of type I filter

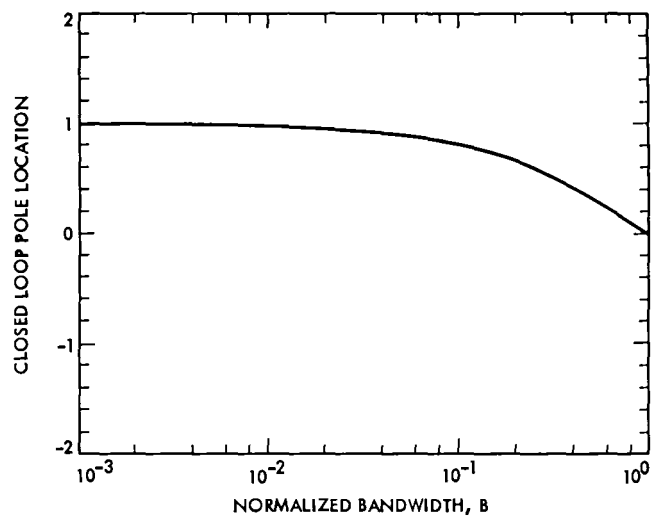


Fig. 9. Closed loop pole location of type I PLL

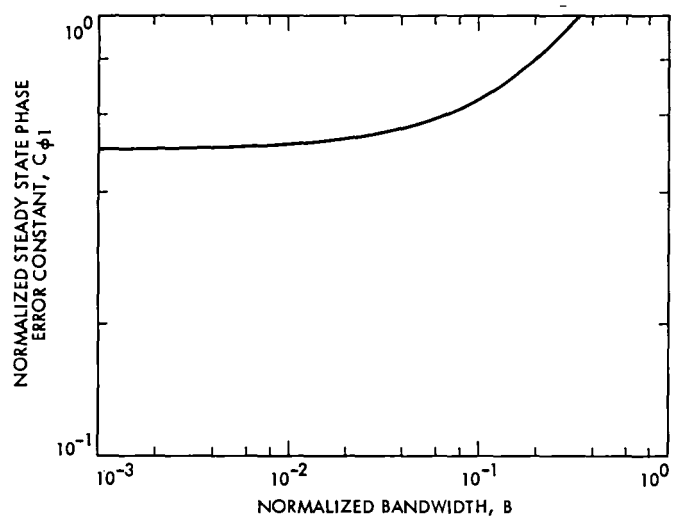


Fig. 8. Steady state error performance of type I filter

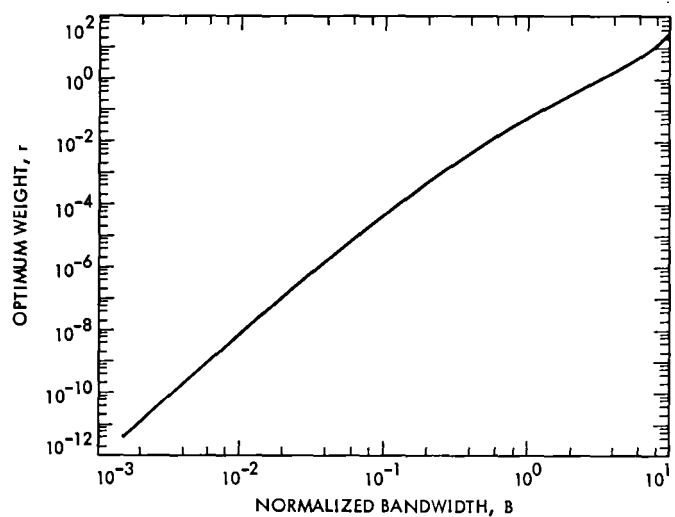


Fig. 10. Optimum weight r for loop filter of type II

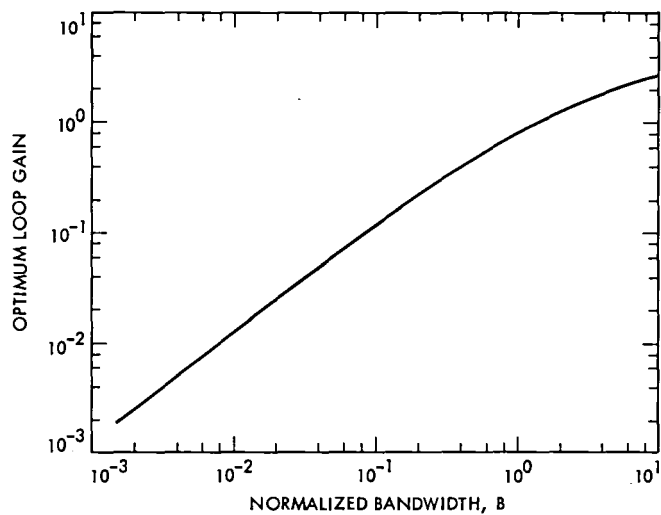


Fig. 11. Optimum loop gain vs normalized bandwidth for type II filter

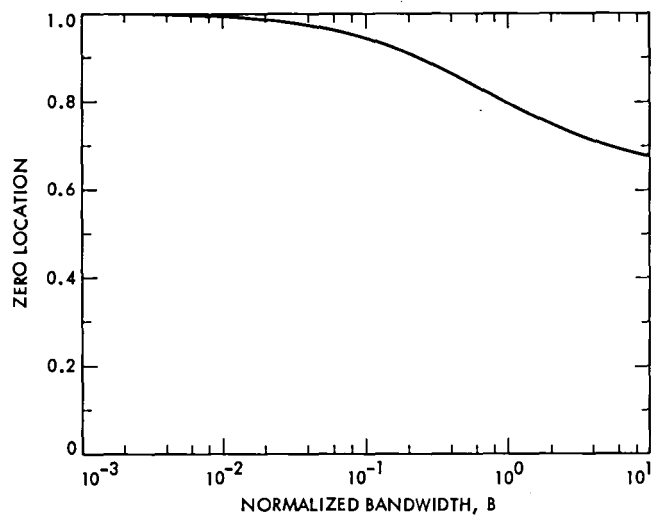


Fig. 12. Zero location of the optimum loop filter of type II

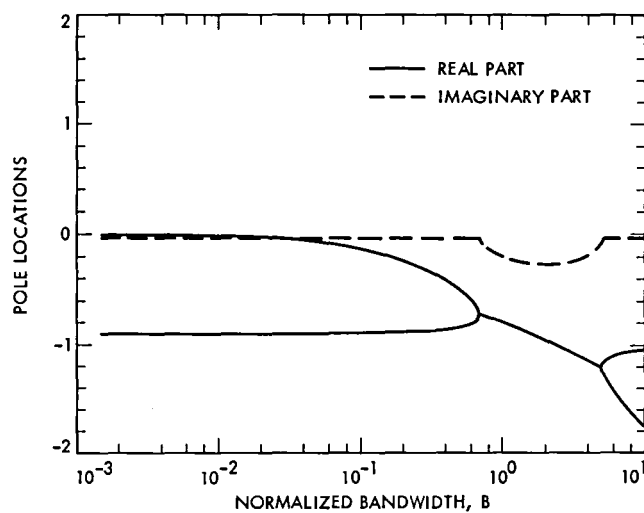


Fig. 13. Pole locations of the optimum loop filter of type II

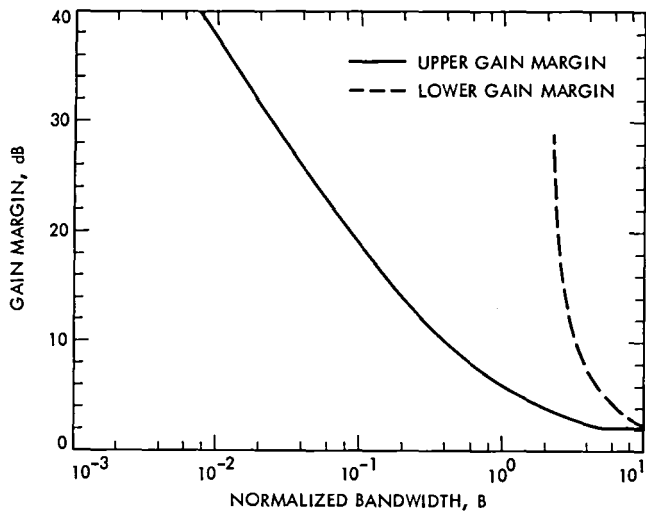


Fig. 14. Gain margin vs normalized bandwidth for type II filter

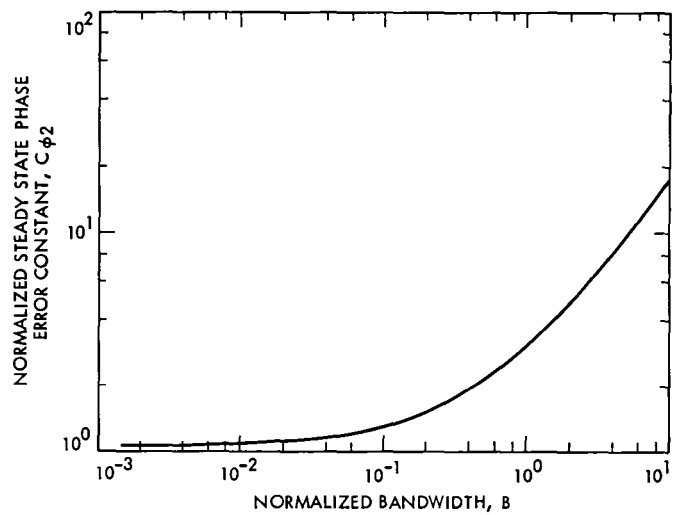


Fig. 16. Steady state error performance of type II filter

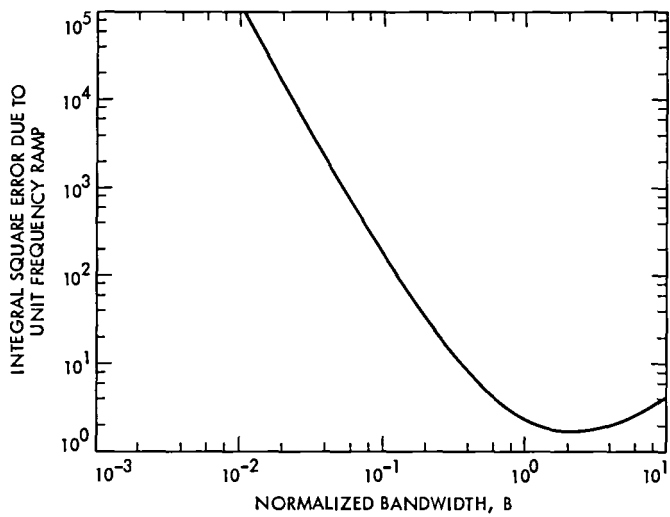


Fig. 15. Transient error performance of type II filter

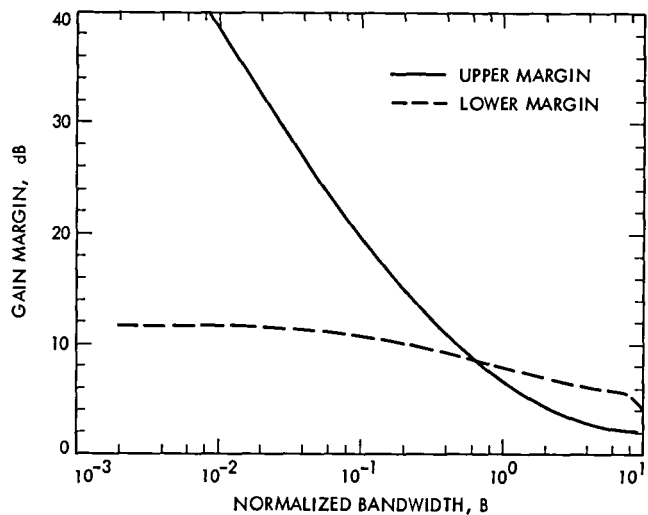


Fig. 17. Gain margin vs normalized bandwidth for type III filter

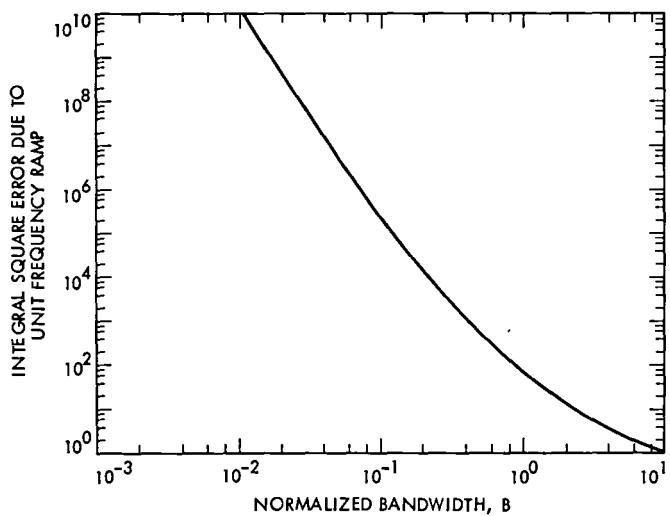


Fig. 18. Transient error performance of type III filter

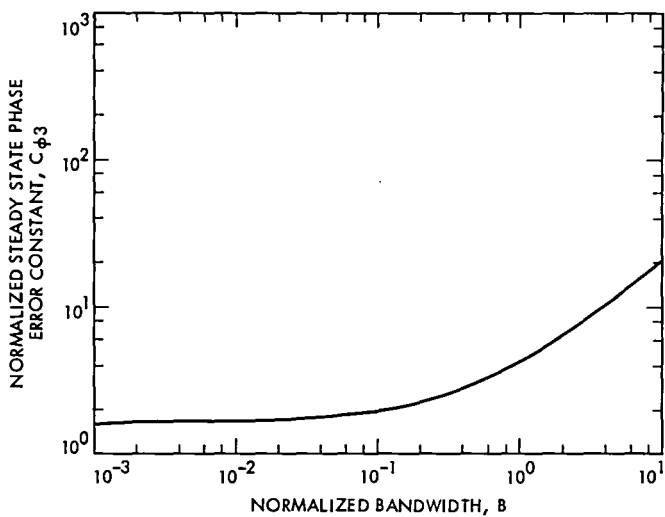


Fig. 19. Steady state error performance of type III filter

Appendix

In the following, we derive the transfer functions of the optimum filters for various phase dynamics considered in the article.

I. Case 1: Phase Step

In this case $\Theta(z) = (1 - z^{-1})^{-1}$, and the expression for $P(z)$ can be easily evaluated as,

$$\begin{aligned} P(z) &= -\frac{K^2 \bar{N}_0 T^3}{4} \frac{(z+1)^2}{(z-1)^2} \left\{ 1 - \frac{qz}{(z-1)^2} \right\} \\ &= -\frac{K^2 \bar{N}_0 T^3}{4} \frac{z^4 - qz^3 - 2(1+q)z - qz + 1}{(z-1)^4} \end{aligned} \quad (\text{A-1})$$

Factorization of $P(z)$ yields,

$$P(z) = \left\{ \frac{K^2 \bar{N}_0 T^3}{4} \frac{(az^2 + bz + c)}{(z-1)^2} \right\} \times \left\{ \frac{az^{-2} + bz^{-1} + c}{(z^{-1} - 1)^2} \right\} \quad (\text{A-2})$$

where the two bracketed terms in the above represent $P^+(z)$ and $P^-(z)$ respectively. Comparing the coefficients of equal powers of z in the equivalent expressions (A-1) and (A-2), the following set of equations (referred to above as Eq. [11]) are obtained.

$$\begin{aligned} ac &= -1 \\ ab + bc &= q \\ a^2 + b^2 + c^2 &= 2(1+q) \end{aligned}$$

The above set of equations can then be solved for unknown a, b, c .

Now,

$$\frac{\lambda N(z^{-1}) \Phi_{\Theta\Theta}(z)}{zP^-(z)} = \frac{\lambda KT}{2} \frac{(z+1)z^2}{(z-1)(a+bz+cz^2)}$$

Writing down the partial fraction expansion for the above

$$\left[\frac{\lambda N(z^{-1}) \Phi_{\Theta\Theta}(z)}{zP^-(z)} \right]_+ = \frac{\lambda KT}{2} \frac{2/(a+b+c)}{(z-1)} \quad (\text{A-3})$$

This is in view of definition of $[]_+$ and the fact that the roots of $a + bz + cz^2 = 0$ are outside the unit circle. Substitution of Eq. (A-3) and the expression for $P^-(z)$ from Eq. (A-2) into Eq. (6) yields the following expression for $W_0(z)$,

$$W_0(z) = \frac{4q}{(a+b+c)} \frac{1}{KT} \frac{z(z-1)}{(az^2 + bz + c)}$$

From Eq. (7) and the expansion for $(a+b+c)^2$, it follows that $(a+b+c)^2 = 4q$, and the expression for $W_0(z)$ can be simplified to the one given in Section IV.

Straightforward substitution yields that,

$$\begin{aligned} 1 - W_0(z)N(z) &= \frac{2az^3 + 2bz^2 + (c-a-b)z - (a+b+c)}{2z(az^2 + bz + c)} \end{aligned}$$

and the optimum loop filter is given by

$$\begin{aligned} F(z) &= \frac{W_0(z)}{1 - W_0(z)N(z)} \\ &= \frac{2(a+b+c)}{KT} \times \frac{z^2(z-1)}{2az^3 + 2bz^2 + (c-a-b)z - (a+b+c)} \end{aligned}$$

Cancellation of the factor $(z-1)$ common to both the numerator and denominator yields Eq. (12).

II. Case 2. Frequency Step

In this case $\Theta(z) = Tz/(z-1)^2$, and the $P(z)$ is given by

$$\begin{aligned} P(z) &= -\frac{K^2 \bar{N}_0 T^3}{4} \frac{(z+1)^2}{(z-1)^2} \left\{ 1 + r \frac{z^2}{(z-1)^4} \right\}; \quad r = \frac{\lambda T}{\bar{N}_0} \\ &= -\frac{K^2 \bar{N}_0 T^3}{4} \\ &\quad \times \frac{\{z^6 - 2z^5 + (r-1)z^4 + (2r+4)z^3 + (r-1)z^2 - 2z + 1\}}{(z-1)^6} \end{aligned}$$

Factorization of $P(z)$ results in

$$P(z) = \left\{ \frac{K^2 \bar{N}_0 T^3}{4} \frac{az^3 + bz^2 + cz + d}{(z-1)^3} \right\} \times \left\{ \frac{az^{-3} + bz^{-2} + cz^{-1} + d}{(z^{-1}-1)^3} \right\} \quad (A-4)$$

where the factors of $P(z)$ above represent $P^+(z)$ and $P^-(z)$ respectively. Comparison of the two alternative expressions yields the following set of equations for the solution of a , b , c and d .

$$\left. \begin{aligned} ad &= 1 \\ ac + bd &= -2 \\ ab + bc + cd &= (r-1) \\ a^2 + b^2 + c^2 + d^2 &= (2r+4) \end{aligned} \right\} \quad (A-5)$$

Now,

$$Y(z) \triangleq \frac{\lambda N(z^{-1}) \Phi_{\Theta\Theta}(z)}{zP^-(z)} = \frac{(z+1)z^3}{(z-1)^2} \frac{1}{(a+bz+cz^2+dz^3)} \quad (A-6)$$

Writing down the partial fraction expansion of the above rational function of z and recognizing that the roots of $a+bz+cz^2+dz^3=0$ lie outside the unit circle,

$$\left[\frac{\lambda N(z^{-1}) \Phi_{\Theta\Theta}(z)}{zP^-(z)} \right]_+ = \frac{A}{(z-1)^2} + \frac{B}{(z-1)};$$

$$A = (z-1)^2 Y(z) \Big|_{z=1} = \frac{\lambda K T^3}{(a+b+c+d)}$$

$$B = \frac{d}{dz} (z-1)^2 Y(z) \Big|_{z=1} = \frac{\lambda K T^3}{2} \frac{(7a+5b+3c+d)}{(a+b+c+d)^2}$$

Substitution of A and B , and a little simplification yields,

$$[Y(z)]_+ = \frac{\lambda K T^3}{2} \frac{1}{(a+b+c+d)^2} \times \frac{\{(7a+5b+3c+d)z - (5a+3b+c-d)\}}{(z-1)^2} \quad (A-7)$$

Substitution of Eq. (A-7) and the expression for $P^-(z)$ from Eq. (A-4) into Eq. (6) provides an expression for the optimum close loop filter $W_0(z)$,

$$W_0(z) = \left(\frac{\lambda T}{\bar{N}_0} \right) \left(\frac{2}{KT} \right) \times \frac{1}{(a+b+c+d)^2} \frac{(h_0 z - h_1)(z-1)z}{(az^3 + bz^2 + cz + d)};$$

$$h_0 = (7a+5b+3c+d)$$

$$h_1 = (5a+3b+c-d)$$

From Eq. (A-5) and the expansion for $(a+b+c+d)^2$, it follows that $(a+b+c+d)^2 = 4r$. Substitution of this in the above leads to the expression for $W_0(z)$ given in Section IV.

Straightforward substitution yields the optimum loop filter $F(z)$ as,

$$F(z) = \frac{2}{KT} \times \frac{(h_0 z - h_1)(z-1)z^2}{4az^4 + 4bz^3 + (4c - h_0)z^2 + (4d - h_0 + h_1)z + h_1}$$

Factorization of the denominator polynomial and cancelling the factor $(z-1)$ common to both the numerator and denominator,

$$F(z) = \frac{2}{KT} \times \frac{(h_0 z - h_1)z^2}{(z-1)\{4az^2 + (8a+4b)z + (5a+3b+c-d)\}}$$

Solution of the Filter Coefficients a, b, c, d . For any given value of r , the system of simultaneous non-linear equations, Eq. (A-5) must be solved for a, b, c and d . In fact, as discussed earlier, it is sufficient to solve the set (18) for $\bar{a}, \bar{b}, \bar{c}$ first and then obtain a, b, c , and d from

$$a = \bar{a}$$

$$b = \bar{b} + 0.9\bar{a}$$

$$c = \bar{c} + 0.9\bar{c}$$

$$d = 0.9\bar{c}$$

It turns out from the stability considerations that $|\bar{a}| > |\bar{c}|$ and without loss of any generality we may assume $a > 0$. Thus, to satisfy the first equation of (18), $\bar{a} > 1$. Since we are interested in solving the set (18) for a range of values of r , an equivalent but simpler approach is to assume a range of \bar{a} and then solve for \bar{b}, \bar{c} , and r from the following:

$$\bar{c} = 1/\bar{a}$$

$$\bar{b} = -4/(\bar{a} + \bar{c})$$

$$r = (\bar{a}^2 + \bar{b}^2 + \bar{c}^2 - 6)^{1/2}$$

A Laser Plotting System for VLSI Chip Layouts

J. A. Harding and L. J. Deutsch
Communications Systems Research Section

One of the most time consuming facets of custom Very Large Scale Integration (VLSI) design is obtaining hardcopy plots of the mask geometries of cells and chips. The traditional method of generating these plots is to use a multicolor pen plotter. Pen plotters are inherently slow and the plotting speed increases linearly with the number of edges that must be plotted. A moderate custom chip design at JPL now consists of more than 200,000 such edges and can take as much as eight hours to plot using a pen plotter. This paper describes software that was written at JPL to produce similar plots using a laser printer. It is shown that, for rather small layouts, the laser printer can provide nearly instantaneous turnaround. For moderate to large chip designs, the laser printer provides a factor of five or more improvement in speed over pen plotting.

I. Introduction

The design of a custom or semicustom Very Large Scale Integrated (VLSI) chip involves many Computer Aided Design (CAD) tools. A UNIX-based CAD system for VLSI design was described in Ref. 1. The process is centered around the creation of a computer file that describes the geometry for each mask layer in the fabrication process. This data is often referred to as a *layout*. Caltech Intermediate Format (CIF) (Ref. 2) is simply a standardized syntax for the description of such geometry.

There are several stages in the design process when it is preferable that the designer have a hardcopy plot of a layout file. One of these stages is called floorplanning. In floorplanning, layouts of subcells are arranged into a larger system. This involves a great deal of rearranging of the subcells. Hardcopy plots help the designer understand the relative sizes, interconnection requirements, and shapes of the subcells.

Hardcopy plots are also useful in the chip simulation process. If the plots have node labels on them, then the designer can use them to follow a circuit's behavior during simulation.

Plots are sometimes helpful even while performing layout editing — even when using a graphical layout editor such as *caesar* (Ref. 3). These plots help the designer see areas of the chip that are too large to be viewed on the graphics screen at one time.

All of these applications of hardcopy *cifplots* are only feasible if the plots can be generated in a relatively short time. Currently, *cifplots* are generated on mechanical pen plotters. These devices are inherently slow and are only useful in the design process for very small cells. The turnaround time for generating pen plots is, as a rule, only acceptable for layouts of cells with up to around 50 transistors. Larger cells require more time to plot than is practical for applications other than producing documents.

The time that it takes a pen plotter to produce a cifplot is roughly proportional to the number of lines that it must plot multiplied by the length of these lines (this is the arclength of the CIF file). Most pen plotting programs for cifplotting use a different color pen for each mask layer. This has definite advantages. However, only the outline of each geometric shape is plotted. This means that if the plot is even moderately dense, it is often difficult to tell where the interior and exterior regions of these shapes really are. Many pen plotting programs are not capable of placing text on the plot. This means that electrical nodes cannot be labeled and the plot will be of limited use during the chip simulation process. Finally, most existing software for cifplotting — regardless of the output device — is not technology independent. This means that the programs are limited to a small number of fabrication technologies and therefore they will always become obsolete.

With the complexity of JPL in-house chip designs increasing by about a factor of five every year, it became necessary this year to address the hardcopy plotting issue. In an attempt to solve most of the above problems, a laser printer was purchased. The particular printer is a Quality Micro Systems (QMS) Lasergrafix 2400. The QMS printer is capable of producing high quality text at a rate of 24 pages each minute. It also comes with special hardware and firmware for performing many complex graphics functions. It may be operated in a pure vector mode similar to that of a pen plotter only many times faster. The resolution of the QMS printer is 300 points per inch in both axes. It uses 8.5 in. X 11 in. plain paper.

A laser printer is a device that uses a plain paper copying process to produce rasterized hardcopy output. In the case of the QMS printer, the method is the Xerographic copying process. In fact, the QMS is based on a standard Xerox brand copier. A laser is used to scan input pixel information onto a drum. A sheet of plain paper is passed over the drum and the pixels that will be black are heated with respect to the rest of the paper. A dry toner is then applied to the paper. It fixes to the paper only where the paper has been heated by the drum. The remaining toner is then removed resulting in a finished page.

Software was written to produce cifplots on the QMS printer. The software is called *Laser*. Plots generated by *Laser* are produced in black and white. This is a disadvantage over the pen plotters. However, the geometric shapes may be filled in with different hatching styles. This gives the same functionality as the different colors, and it also solves the problem of distinguishing interior and exterior regions in complex chip plots. *Laser* is completely technology independent. The user may use a technology description file in a standard system library or one of original design. By using the

QMS printer's text capability, *Laser* can also label electrical nodes. Finally, *Laser* works much faster than programs that use pen plotters. It is now possible to obtain hardcopy cifplots of chips with hundreds of transistors in a matter of minutes. In fact, since the time required to produce *Laser* plots is roughly proportional to the number of geometric shapes rather than arc length or number of edges, the *Laser* printer has a distinct speed advantage over pen plotters.

Since all the designs that are produced on the UNIX-based CAD system are of *Manhattan* geometry (i.e., they consist of rectangles that are all aligned with the *x* and *y* axes) the ability to plot non-*Manhattan* CIF was not programmed into *Laser*. Only a minor modification would be required to allow *Laser* to plot non-*Manhattan* (sometimes called *Boston*) geometry.

II. The Laser Cifplotting Software

Laser actually consists of several individual programs that are controlled through a command file (called a *shell script* in UNIX). The command file is written in the UNIX C-shell. It parses the command line and routes the appropriate data through the various programs. The user invokes *Laser* by typing the word *Laser*. In addition, the user may specify certain options on the command line.

There are currently three options supported on the command line. First, a technology description file may be specified. The default technology is currently NMOS. The *Laser* command file first searches the system's technology library for the requested file. If it is not found there, then the user's current directory is searched. In this way, each user may have a set of unique technology description files. If a designer were working with an NMOS design, for example, but wished to use a hatching style for the layers that is different than the one in the system library, then a technology description file with a name such as *nmos.my-own* could be created and placed in the user's own directory.

Another option allows the user to specify which layers should be filled in and which should be plotted as outlines only. This option is provided to allow even quicker turnaround time for simple cell layouts. Since node labels are treated like a mask layer in the *Laser* program, the user could use this option to eliminate the plotting of labels. This might be necessary on very complex plots in which the labels would overlap each other. *Laser* creates a legend on the finished plot that indicates the hatch styles that are used for each non-deleted layer.

The last option that the user may request on the command line is the *title* option. The user may use this option to pass a

line of text to the Laser program for use as a large headline-style title that will be placed at the top of the plot.

The first program that is called by the Laser command file is a University of California at Berkeley program called *cifflatten*. CIF is a hierarchical description language. This means that cells that are used many times in a chip design need only be described once at a geometric level. Further references to that cell may then be specified by calls to this definition. *Cifflatten* takes a CIF file as an input and produces a CIF file that describes the same design but with all the hierarchy removed. This "simple" CIF is easier to process in subsequent parts of the program. *Cifflatten* was slightly modified to allow the calculation of the overall geometric dimensions of the chip design at the same time as the flattening operation. This information is needed to scale the plot for final output on a hardcopy page.

After flattening, the CIF information is processed by a program called *qmscif*. *Qmscif* is a Pascal program that was written here at JPL. It produces a file that contains the control sequences for the QMS printer that are needed to produce the plot. QMS calls this set of codes the *QUIC* programming language. The *QUIC* language has graphics primitives for vector plotting, box drawing, and labeling. It even has a simple looping ability that is used in *qmscif* to produce the various hatch styles for different VLSI layers.

The *QUIC* file is placed in the UNIX spooling system for subsequent downloading to the QMS printer. Laser is fully integrated onto the UNIX spooling environment. This means that *cifplotting*, laser text printing, and standard (dot matrix) text printing are all under one set of system controls.

When the *QUIC* file reaches the top of the spooler's queue, it is sent to the QMS printer. A plot is then generated.

III. Some Benchmark Tests

In order to determine the functionality and relative performance improvement over pen plotting, a set of CIF files that covers a wide range of typical designs was assembled. Many of these CIF files come from designs that were performed for projects within the DSN Advanced Systems Program. Table 1 shows some of the important statistics of these designs.

The names of the designs are shown in the first column of the table. *Shiftcell* is a single half-bit stage in a dynamic NMOS shift register. It is the simplest of the designs that were tested. *Fa* is a full adder. It is typical of the complexity of a single low-level subcell in a chip design. *D* is a d-type flip-flop. It comes from a library of standard cells that are

based on the Sandia National Laboratory radiation-hardened CMOS fabrication process. *Lag* is a bit slice through a three-level complex correlator with phase rotation. It is typical of an intermediate level subcell, and it contains several cells of the complexity of *fa*. *Conv2* is a small complete chip. It is the layout of the Multicode Convolutional Encoder chip (Ref. 4). *Rsint* is the layout of an 8-bit (255, 223) Reed-Solomon encoder with interleaving (Ref. 5). It is an example of a chip of moderate complexity. No chips of higher complexity than *rsint* were benchmarked because the time for pen plotting would have been prohibitive.

The third column of the table shows the number of transistors in each design. It gives a good indication of the relative complexity of the designs. It is important to note that the increase in complexity between *conv2* and *rsint* is typical of the trend over the two-year time period that this represents here at JPL. A typical large chip that is designed in-house today contains between 30,000 and 60,000 transistors.

The fourth column indicates the number of rectangles in the design. Since all these designs are of Manhattan geometry, this is actually the number of individual shapes in the design. The Laser plotting speed should be roughly proportional to this number.

The next column is labeled *Normalized Arc Length*. This number, when divided by 1,000, is actually the number of millimeters of perimeter for all the rectangles that would be measured on the fabricated chip using a standard fabrication scaling ($\lambda = 2.0\mu$). The time for pen plotting should be proportional to this number. Notice that ratio of the largest to the smallest number of rectangles is about 3,200 while the similar ratio in arc length is about 4,300. This already indicates a probable speed advantage for Laser plotting, even if the time to plot a shape is the same on the two devices.

The last column in the table indicates the *regularity ratio*. This is a measure of the hierarchical complexity of the design. It is actually the number of rectangles in the design divided by the number of rectangles defined explicitly (rather than by calls to cell definitions) in the CIF file. A value of 1.0 means that there is no hierarchy in the CIF. The higher the ratio, the more regular the design is. *Rsint* is a very regular design that contains large blocks of identical cells. The regularity ratio is an indication of the amount of computation needed to perform the CIF flattening operation.

These designs were plotted using both the pen plotter and the QMS Laser printer. The plots were performed on a Digital Equipment Corporation VAX 11/750 computer with a load average of approximately 3.0 (moderately heavy usage). The plotting and processing times (actual elapsed time) were both

recorded. The processing time for the pen plotter is not important since it is negligible compared to the mechanical plotting time. Also, since the pen plotter is not a spooled device, it is run directly by the software and the processing time cannot be distinguished.

The results of these timing tests are recorded in Table 2. The pen plots of the six CIF files appear in Figs. 1 through 6. The Laser plots for the same files appear in Figs. 7 through 12.

IV. Conclusions

The first thing that is evident from the figures is that the Laser plots are much easier to read. This is because the QMS printer fills in the various layers. The layer types and intersections of layers are all easily discernible. Even though the pen plots were originally in color, the color effect is much less pleasing to the eye than the solid regions produced by the Laser printer.

Table 2 shows the processor and plotting times required for both the pen and Laser plots. Notice that in every case, the Laser plot took less time to generate than the corresponding pen plot. Even in the smallest CIF file (shiftcell) there was an almost three-fold increase in performance. In fact, the ratio

of Laser plotting time to pen plotting time increases with the complexity of the CIF file as expected.

V. Future Work

Some improvements will be made in Laser during the next year. One of these will be to add the ability to produce mosaic plots. Since the QMS printer is limited to a paper size of 8.5 in. \times 11 in., large CIF files for which the designer wishes to see a reasonable amount of detail will have to be generated on multiple pages. These could then be pasted together to produce one large plot. This technique is also used with the current pen plotter which is limited to 11 in. \times 14 in. paper.

In order to make the software run faster, the code in qmcsif should be merged with ciffatten so that multiple interrogations of very large data files need not occur during processing. Another advantage of doing this would be to allow some cells of a CIF file to have only their outlines plotted.

As costs become less prohibitive, a color Laser printer could be acquired. This would enhance the plots even more and allow many more layer types to be plotted on a page without confusion.

References

1. Deutsch, L. J., "An Integrated UNIX-Based CAD System for the Design and Testing of Custom VLSI Chips," *TDA Progress Report 42-81*, Jet Propulsion Laboratory, Pasadena, California, January-March 1985, pp. 51-62.
2. Mead, C. and Conway, L., *Introduction to VLSI Systems*, Addison-Wesley Publishing Company, Menlo Park, California, 1980, pp. 115-127.
3. Ousterhout, J., *Editing VLSI Circuits with Caesar*, Computer Science Division, University of California, Berkeley, California, 1983.
4. Deutsch, L. J., "A VLSI Implementation of a Multicode Convolutional Encoder," *TDA Progress Report 42-72*, Jet Propulsion Laboratory, Pasadena, California, February 15, 1983, pp. 61-69.
5. Truong, T. K., Deutsch, L. J., Reed, I. S., Hsu, I. S., Wang, K., Yeh, C. S., "The VLSI Design of a Reed-Solomon Encoder Using Berlekamp's Bit Serial Algorithm," *Proceedings of the Third Caltech Conference on VLSI*, California Institute of Technology, Pasadena, California, 1983, pp. 303-330.

Table 1. Statistics for benchmark test layouts

Cell Name	Fabrication Technology	Number of Transistors	Number of Rectangles	Normalized Arc Length	Regularity Ratio
shiftcell	NMOS	31	17	324	1.0
fa	NMOS	18	114	2,378	1.0
d	CMOS	25	288	11,364	1.0
lag	NMOS	184	1676	44,956	2.6
conv2	NMOS	453	5646	186,632	4.0
rsint	NMOS	8918	54182	1,405,182	27.1

Table 2. Results of benchmark timing tests for pen and laser CIF plotting

Cell Name	Time for CIF Processing	Time to Download to Laser Printer	Total Time for Laser Plotting	Time for Pen Improvement	Relative Speed
shiftcell	0:13	0:16	0:39	1:05	2.7
fa	0:23	0:38	1:01	3:46	3.7
d	0:46	1:12	1:58	6:58	3.5
lag	1:48	1:28	2:36	13:24	5.2
conv2	11:10	3:38	14:48	42:40	2.9
rsint	37.40	26.36	1:03:56	5:51:00	5.5

Note: All times are in the form *hours:minutes:seconds*

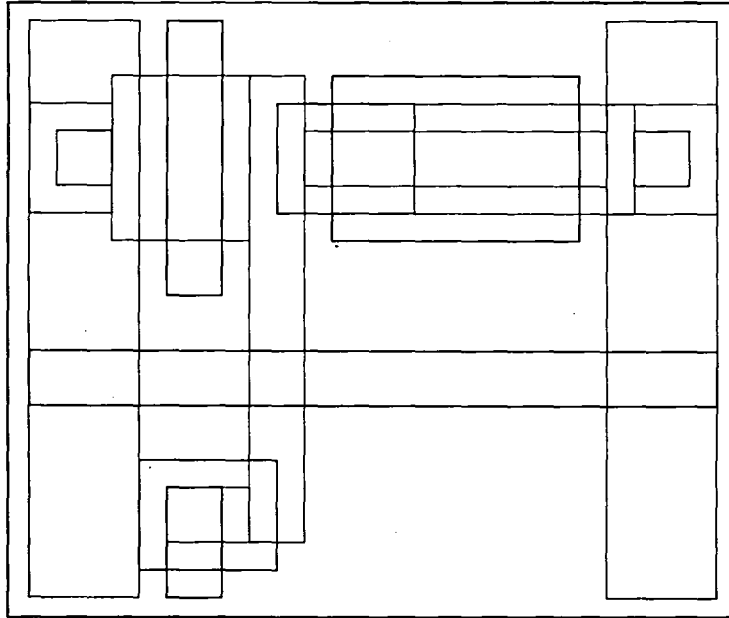


Fig. 1. Pen cilplot of *shiftcell*

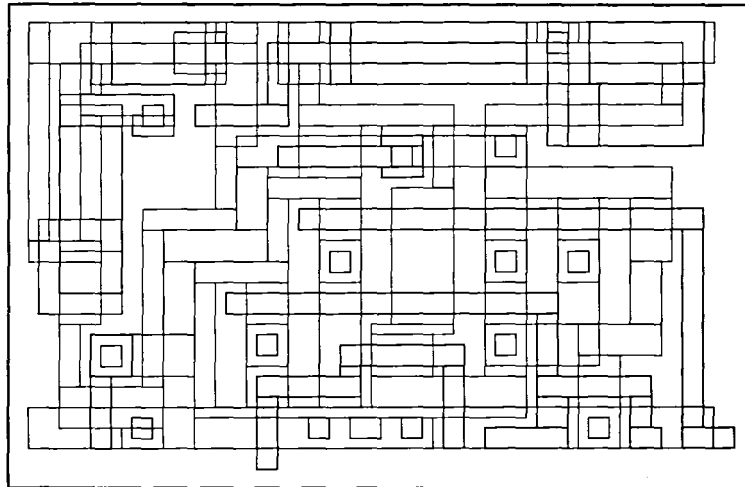


Fig. 2. Pen cilplot of *fa*

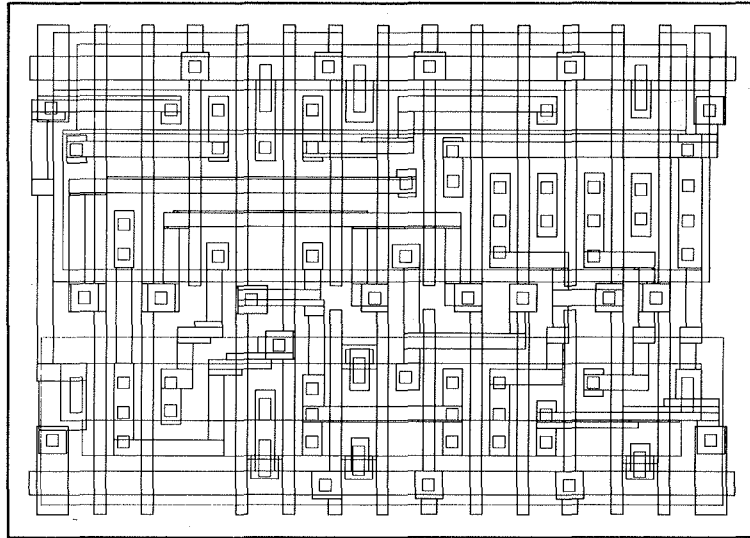


Fig. 3. Pen cifplot of d

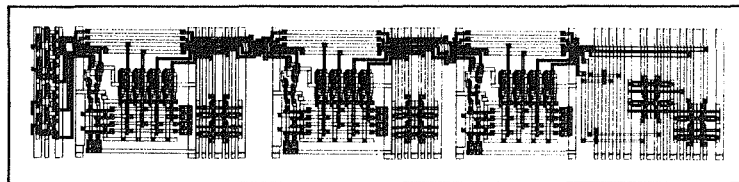


Fig. 4. Pen cifplot of lag

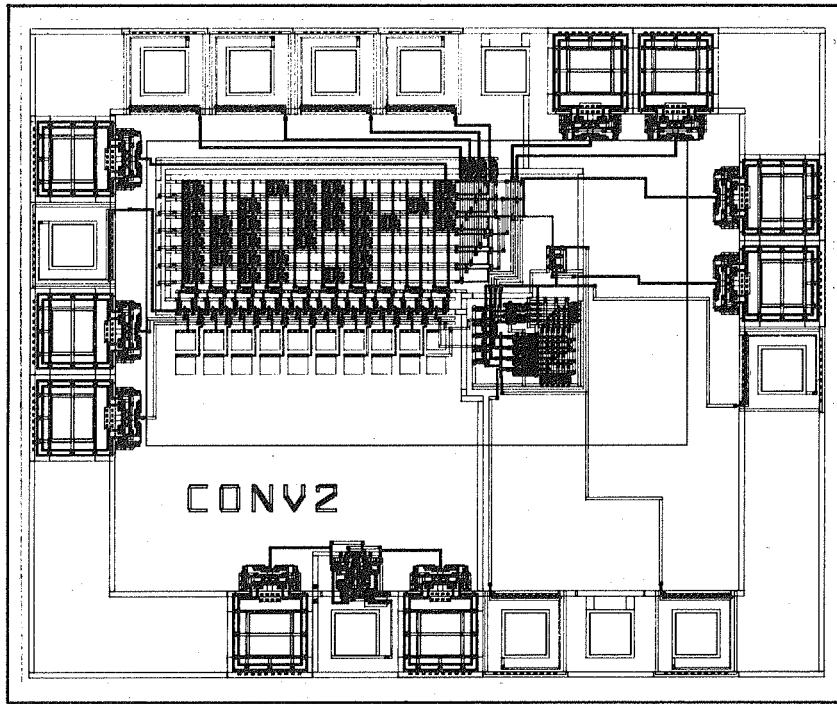


Fig. 5. Pen cifplot of *conv2*

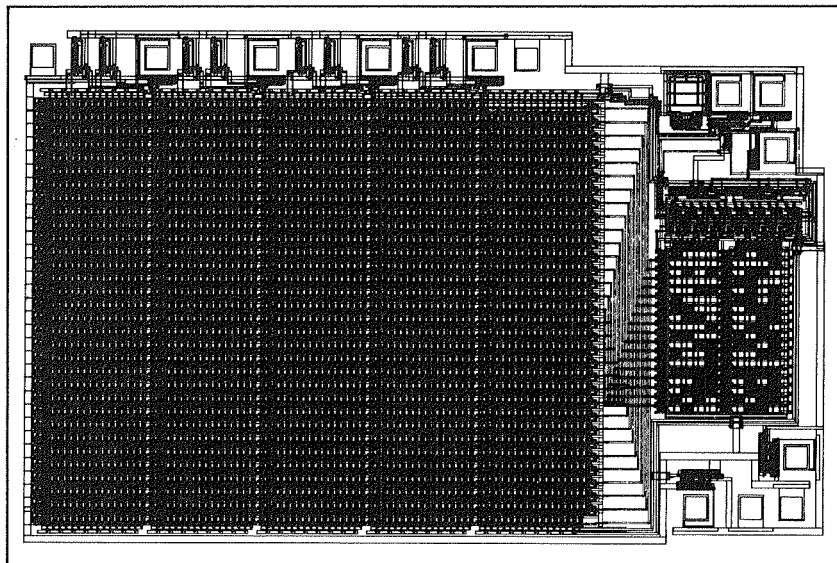


Fig. 6. Pen cifplot of *rsint*

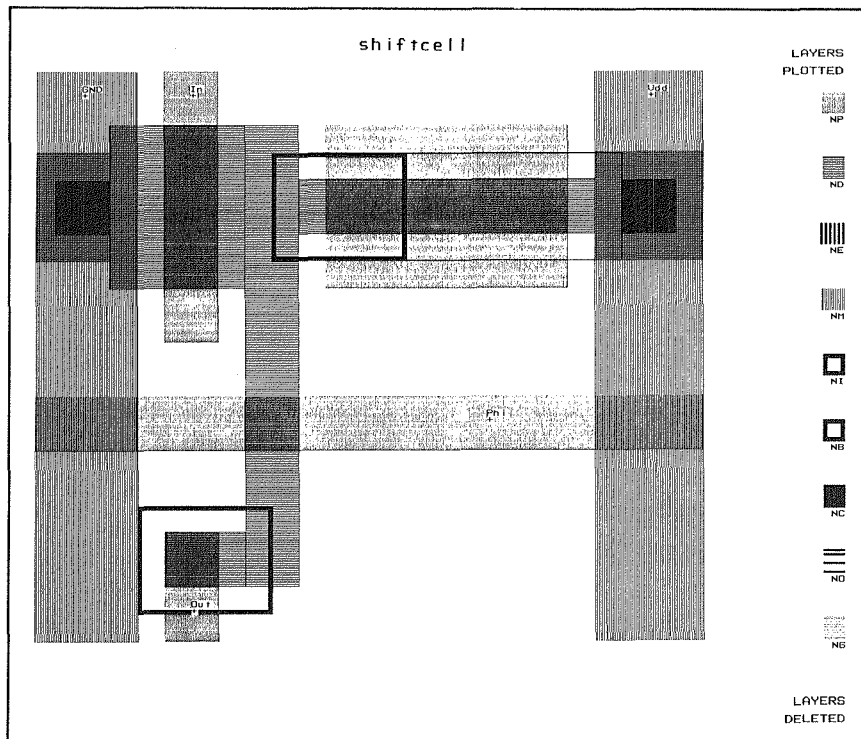


Fig. 7. Laser citplot of *shiftcell*

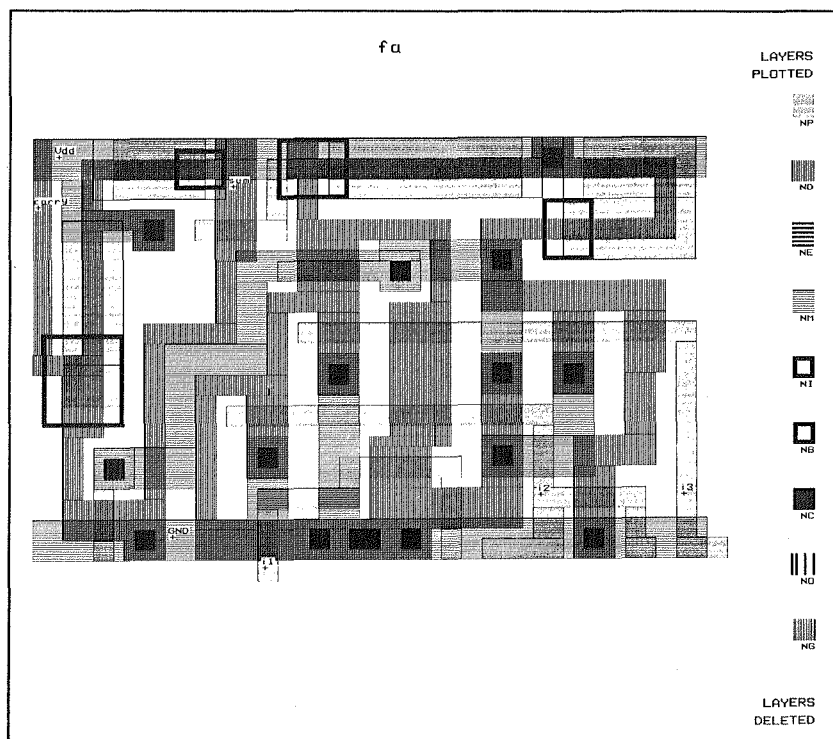


Fig. 8. Laser citplot of *fa*

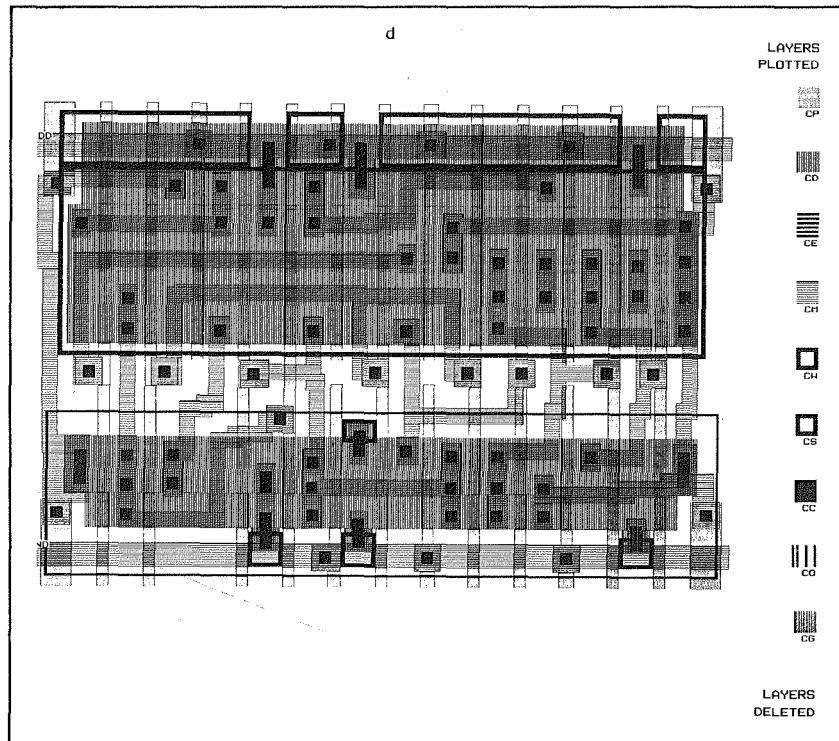


Fig. 9. Laser ctfplot of *d*

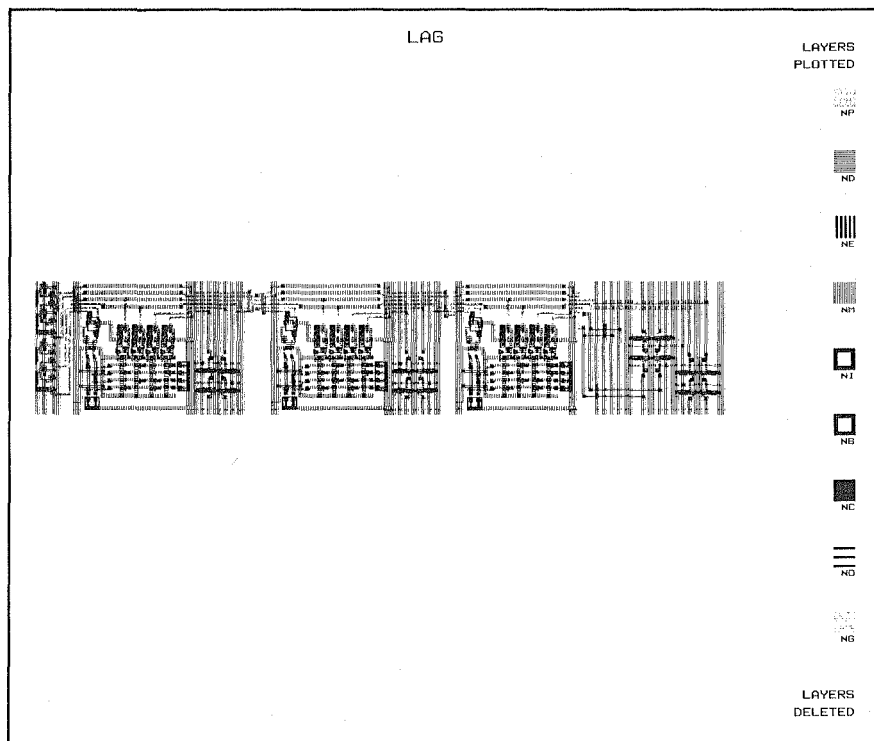


Fig. 10. Laser ctfplot of *lag*

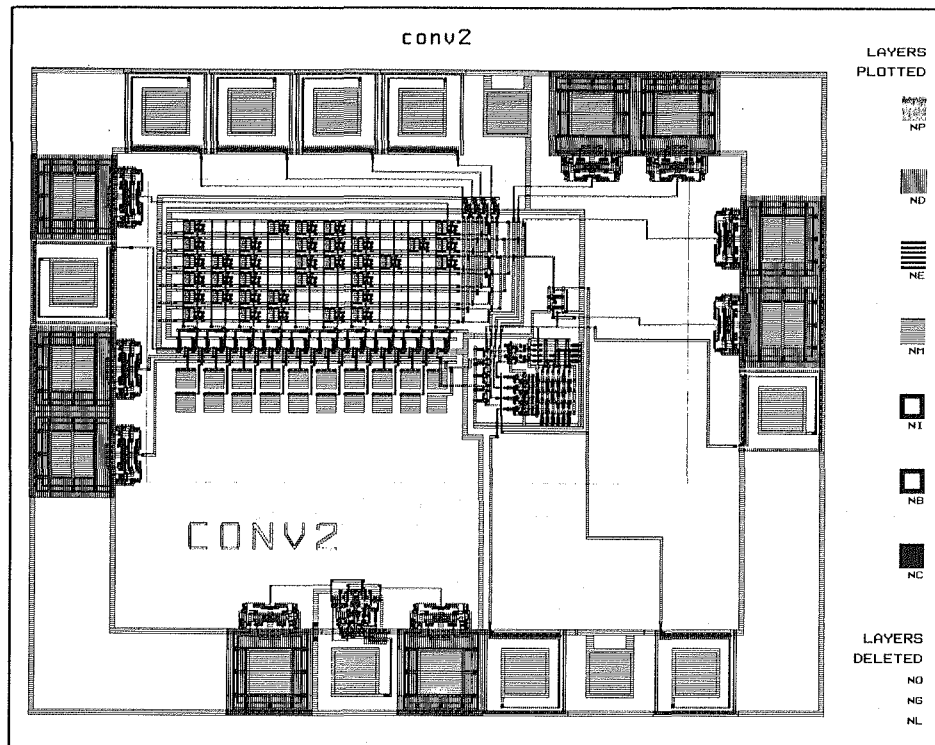


Fig. 11. Laser cifplot of conv2

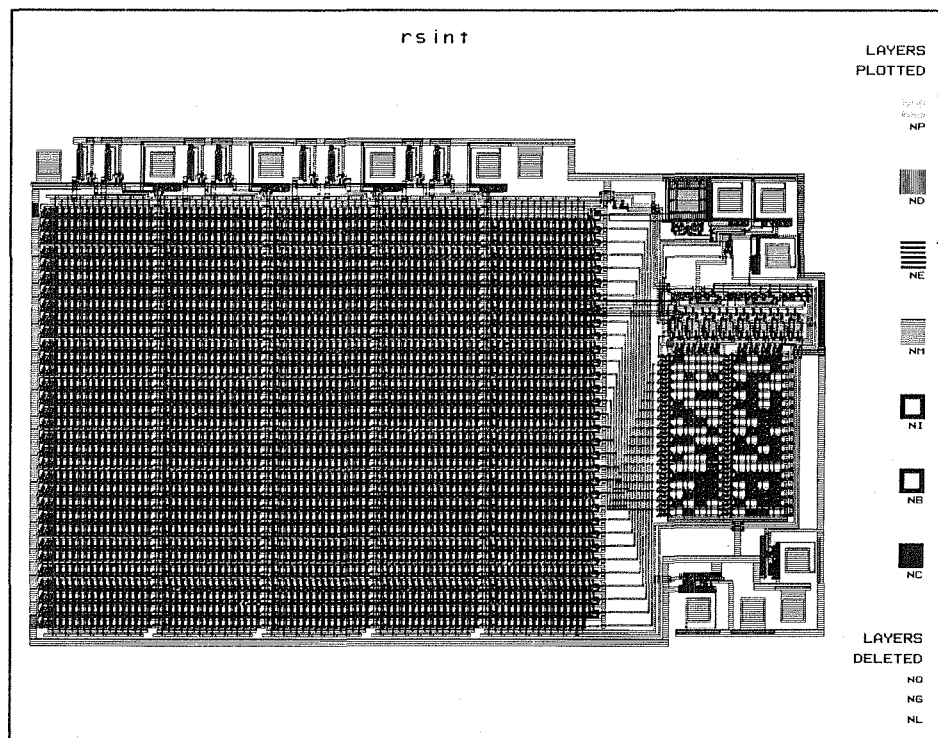


Fig. 12. Laser cifplot of rsint

Mark IV-A DSCC Telemetry System Description

D. L. Ross
DSN Systems Engineering

This article provides an update to the description of the Deep Space Communications Complex (DSCC) portion of the MK IV-A Telemetry System. This system is currently being implemented at all signal processing centers. The Mark III Telemetry System is the predecessor of the Mark IV-A System.

I. Introduction

Implementation of the DSN MK IV-A Telemetry System (Ref. 1) has been proceeding for the past two years and is nearly complete at all signal processing centers. The Deep Space Communications Complex (DSCC) portion of the DSN Telemetry System has been changed in two major ways as a result of the Mark IV-A DSN implementation. The Mark III Telemetry System is the predecessor of the Mark IV-A System and is described in Ref. 2.

- (1) One 34-meter antenna has been added to each DSCC at Goldstone and Canberra. A 34-meter antenna will also be added at the Madrid Complex in June 1987. The arraying of this additional antenna with the existing 64- and 34-meter antennas at X-band will provide a relative gain of 1.84 dB (assuming a 0.2 dB array loss) as compared to a single 64-meter antenna, or a relative gain of 1.07 dB as compared to the arraying of the existing 64- and 34-meter antennas. To accomplish this, equipment has been added to provide baseband combining of signals from the three antennas.
- (2) The telemetry system has been configured to support three missions simultaneously, with one or two of them being highly elliptical earth orbiters. Highly elliptical orbiter (HEO) missions will have coded data

at rates up to 211.2 ksps, manchester coded, modulated directly on the carrier.

Implementation of these changes, combined with previous capabilities, has prepared the network to support Deep Space and HEO missions.

The Deep Space missions are the following:

Pioneers 6 through 9
Pioneers 10 and 11
Pioneer Venus
Voyager
Galileo
Ulysses
Giotto

The HEO missions are the following:

ICE (International Cometary Explorer)
AMPTE (Active Magnetospheric Particle Tracer Experiment)
CCE (charge composition Explorer)

IRM (Ion Release Module)

UKS (United Kingdom Spacecraft)

II. Key Characteristics

The key characteristics of the DSCC portion of the Mark IV-A Telemetry System are:

- (1) Coded data at rates up to 250 kilobits per second (kbps).
- (2) Baseband combining for up to 7 antennas.
- (3) Four complete groups of telemetry equipment at each complex, each with the capacity to support one of the above missions.
- (4) Demodulation of Manchester coded (Bi ϕ -L) or NRZ-L data modulated directly on the carrier.
- (5) Maximum likelihood decoding of short-constraint-length convolutional codes and sequential decoding of long-constraint-length convolutional codes.
- (6) Precise measurement of received signal level and system noise temperature.
- (7) Centralized control by (and real-time reporting to) the Monitor and Control Subsystem.
- (8) Production of digital telemetry Original Data Record (ODR) at each telemetry group with playback via local manual control or in automatic response to GCF inputs; reduced playback rate for higher data rates as required.

The handling of increased data rates and demodulation of NRZ or biphase data modulated directly on the carrier responds to a requirement for telemetry support of HEO spacecraft. Baseband combining provides for improved sensitivity to high data rate X-band signals in support of deep space telemetry and is driven by the Voyager project requirement for support of 29.9 kbps at Uranus encounter. The ICE mission also requires three-input baseband combining; two signals from the 64-m antenna (one right circularly polarized, one left) and one from the standard 34-m antenna. This will enable the data rate to be maintained at 1024 bps during encounter with the Giacobini-Zinner comet. New equipment for baseband combining allows a data rate increase to 250 kbps from the previous peak of 135 kbps. The provision for four groups of telemetry equipment at each SPC responds to the requirement to provide telemetry support to three projects concurrently.

III. Project Data Rate and Coding Requirements

HEO missions are compatible with the existing DSN capabilities. This is illustrated in Table 1, which defines the single link data handling requirements for the HEO projects included in the Mark IV-A mission set. A telemetry single link can be defined as all of the functional elements, from the antenna(s) through an SPC telemetry group, that have been selected for support of a project. The requirement to detect Biphase (Manchester coding) modulation directly on the carrier was met during the Mark IV-A implementation.

Requirements for new and existing deep space missions are listed in Table 2. The new missions to be supported during Mark IV-A are Galileo, Ulysses and Giotto. The next section shows how HEO and deep space mission requirements will be met.

IV. DSCC Conceptual Description

The DSCC block diagram in Fig. 1 provides a conceptual description of the portion of the Mark IV-A Telemetry System to be located at the DSCC. At each complex there will be one 64-meter antenna and two 34-meter antennas (with the exception of the second Madrid 34-m antenna as noted earlier). The 64-meter and 34-meter standard antennas will be able to receive an S-band plus an X-band carrier simultaneously. The 34-meter High-Efficiency antennas will receive one X-band carrier; they are not useful for receiving S-band telemetry carriers for deep space missions because of high system noise temperature (115 K). Table 3 gives the RF reception characteristics for these antennas and indicates the distribution of masers and FETs. The 64-meter antennas will be equipped exclusively with masers whereas the 34-meter antennas will also have FETs. At the 34-meter antennas the masers will provide deep space support, and the S-band FETs will support HEOs with their broader (2200-2300 MHz) reception bandwidth requirements. New Block II-A X-band masers have been provided for the 64-m antennas at Goldstone and Canberra, giving lower X-band system temperatures than in the Mark III DSN. Also, the new 34-meter high-efficiency antennas will provide increased X-band gain compared with the existing 34-meter standard antennas.

Existing Block III and Block IV receivers will be used to receive and detect baseband signals. The Block III receivers were modified to support HEO frequencies and provide a broad (4 MHz) baseband output to the Telemetry Subsystem. The Telemetry Subsystem is arranged to provide four telemetry groups, any of which can process data from any receiver and therefore from either an HEO or deep space spacecraft.

All groups will include the Mark III Maximum Likelihood Convolutional Decoder (MCD) and Telemetry Processor Assembly (TPA). The MCDs were modified to change the node synchronization algorithm to achieve better performance at low signal-to-noise ratio for Voyager Uranus encounter. Telemetry Groups 3 and 4 are equipped with a new Baseband Assembly (BBA) which will include the functions of baseband combining, subcarrier demodulation and symbol synchronization. Figure 2 is a functional block diagram of the BBA. Any combination of receiver outputs can be input to either subcarrier demodulator; or any single receiver output can be routed to either subcarrier demodulator, bypassing the combining function. The monitor and control function is performed from the TPA with no manual intervention required. The BBA is designed to accommodate NRZ-L, NRZ-M or Bi ϕ -L symbol formats, subcarriers up to 2 MHz and data rates from 4 s/s to 1 Ms/s with subcarrier, up to 4 Ms/s (NRZ) without subcarrier or up to 2 Ms/s for Bi ϕ -L. Operation with the BBA, including combining the 64-m and two 34-m antenna basebands, will result in a nominal system degradation at the highest data rates of about 0.3 dB. This includes an allowance for waveform, spectrum correlation and symbol timing losses and represents an improvement over the Mark III system of several tenths of a dB. Telemetry Groups 1 and 2 include modified Mark III Subcarrier Demodulator Assemblies (SDA) and Symbol Synchronizer Assemblies (SSA) as well as an MCD and TPA. Therefore, hardware in Group 1 and Group 2 closely resembles the Mark III Telemetry Subsystem. The SSAs were modified to decode Bi ϕ -L (Manchester coded) data so that Groups 1 and 2 can support HEO missions without

using Spaceflight Tracking and Data Network (STDN) bit synchronizers as proposed in Ref. 3. New software was provided for the TPAs. The 64- and 34-meter antennas can be arrayed by combining baseband signals and performing subcarrier demodulation and symbol synchronization in the BBA in either Telemetry Group 3 or Telemetry Group 4. The combined signal is then decoded in the Maximum Likelihood Convolutional Decoder and formatted for transmission to JPL in the Telemetry Processor Assembly. When combining is not required, outputs from an antenna may also be routed to a Subcarrier Demodulator Assembly (Groups 1 and 2) or to either Baseband Assemblies (Groups 3 and 4).

Any of the telemetry equipment groups can accept two data streams. In any group, one data stream is processed by Channel 1 and one by Channel 2. The performance parameters for Channels 1 and 2 are listed in Tables 4 and 5, respectively. When comparing Table 1 with Tables 4 and 5, note that Data Stream 1 in Table 1 is processed by Channel 1, while Data Stream 2 is processed in Channel 2. Also note that Groups 3 and 4 provide higher data rate capability (for example, for convolutionally coded data: 250 kb/s vs 135 kb/s) and higher subcarrier frequency capability (2 MHz vs 1 MHz). This improvement is due to the BBA and wider receiver passband. However, the overall system will not support the highest BBA/receiver data rate capability until further improvements are made, including a wider bandwidth in the receivers, MCD replacement, provision for high rate recording, and quick look. Future planned missions will require these additional improvements, as well as adding a Reed-Solomon Decoder.

References

1. Burt, R. W., "Mark IV-A DSCC Telemetry System Description," in *TDA Progress Report 42-68*, January and February 1982, Jet Propulsion Laboratory, Pasadena, Calif., April 15, 1982, pp. 130-138.
2. Gatz, E. C., "DSN Telemetry System Mark III-77," in *DSN Progress Report 42-49*, Jet Propulsion Laboratory, Pasadena, Calif., Feb. 15, 1979, pp. 4-9.
3. Burt, R. W., "Mark IV-A DSCC Telemetry System Description," in *TDA Progress Report 42-63*, March and April 1981, Jet Propulsion Laboratory, Pasadena, Calif., June 15, 1981, pp. 3-7.

Table 1. Single link requirements for HEO missions

Mission	Data Stream 1	Data Stream 2
ICE	Convolutionally coded; $K=24, R=1/2$; NRZ-L; 64 b/s; Subcarrier: 1024 Hz or Convolutionally coded; $K=24, R=1/2$; 512 to 2048 b/s; Bi ϕ -L	
AMPTE-CCE	Convolutionally coded; $K=7, R=1/2$; Bi ϕ -L 105,600 b/s	Uncoded NRZ-L; 3300 b/s; subcarrier: 429 kHz
AMPTE-IRM	Convolutionally coded; $K=7, R=1/2$; NRZ-L; 1.024 kb/s; 2.048 kb/s, 4.096 kb/s, or 8.192 kb/s; subcarrier: 131,072 Hz	
AMPTE-UKS	Convolutionally coded; $K=7, R=1/2$, or uncoded; Bi ϕ -L; 1024 b/s, 2048 b/s, 4096 b/s, 8192 b/s, 16384 b/s, 32768 b/s	

Table 2. Single link requirements for deep space missions

Mission	Data Stream 1	Data Stream 2
Pioneers 6-9	Uncoded; NRZ-M; 8 to 512 b/s; subcarrier: 512 Hz for 8 to 64 b/s, 2048 Hz for 128 b/s; S-band	
Pioneers 10/11	Uncoded; NRZ-L; 8 to 2048 b/s; subcarrier: 32.768 kHz; S-band	
	or	
	Convolutionally coded; $K=32$, $R=1/2$, NRZ-L; 8 to 2048 b/s; subcarrier: 32.768 kHz; S-band	
Pioneer Venus	Uncoded; NRZ-L; 8 to 4096 b/s; subcarrier: 16 kHz; S-band	
	or	
	Convolutionally coded; $K=32$, $R=1/2$; NRZ-L; 8 to 2048 b/s; subcarrier: 16 kHz; S-band	
Voyager	Convolutionally coded; $K=7$, $R=1/2$; NRZ-L; 10 to 115,000 b/s; combined X-band; subcarrier: 360 kHz	Uncoded; NRZ-L; 46.667 b/s; subcarrier: 360 kHz; combined X-band
		or
		Uncoded; NRZ-L; 46.667 b/s; subcarrier: 22.5 kHz; S-band
Galileo	Convolutionally coded; $K=7$, $R=1/2$; NRZ-L up to 134.4 kb/s; subcarrier: 360 kHz; combined X-band	Uncoded; NRZ-L 40 b/s; subcarrier: 22.5 kHz; S-band
	or	
	Convolutionally coded; $K=7$, $R=1/2$; NRZ-L; up to 40 kb/s; subcarrier: 22.5 kHz for data rates up to 7.68 kb/s, 360 kHz for rates ≥ 7.68 kb/s; S-band	
Ulysses	Convolutionally coded; NRZ-L; $K=7$, $R=1/2$; 128 b/s to 8,192 b/s; subcarrier: 65,536 Hz for rates up to 1024 b/s, 131,072 Hz for rates of 2,048 b/s or greater; X-band	
Giotto*	Convolutionally coded; NRZ-L; $K=7$, $R=1/2$; Subcarrier: 46.080 kHz for data rate of 360 b/s, 276.480 kHz for data rates of 5,760 b/s, 23,040 b/s and 46,080 b/s; S- or X-band	Uncoded; NRZ-L; Subcarrier: 46.080 kHz for data rate of 360 b/s; S- or X-band
	or	or
	Reed-Solomon and convolutionally coded; NRZ-L; $K=7$, $R=1/2$; Subcarrier: 276.480 kHz for data rates of 5,760 b/s, 23,040 b/s, and 46,080 b/s; S- or X-band	Reed-Solomon coded; NRZ-L; Subcarrier: 276.480 kHz for data rates of 5,760 b/s, 23,040 b/s, and 46,080 b/s; S- or X-band

*Reed-Solomon data are not presently decoded by the DSN; only convolutionally encoded data are decoded by the DSN.

Table 3. RF reception characteristics

Parameter	Antenna		
	64-meter	34-meter Standard	34-meter High-Efficiency
Frequency Range, MHz			
S-band	2270-2300	2200-2300	2200-2300
X-band	8400-8440	8400-8500	8400-8500
Gain, dBi			
S-band	61.7 ^{+0.3} _{-0.4}	56.1 ^{+0.3} _{-0.7}	55.8 ^{+0.0} _{-0.5}
X-band	72.1 ^{+0.6} _{-0.6}	66.2 ^{+0.6} _{-0.6}	67.3 ^{+0.5} _{-0.8}
System Noise Temperature, K Zenith			
S-band with maser			
Diplex	18.5 ±3	27.5 ±2.5	
Listen-only	14.5 ±2	21.5 ±2.5	
S-band with FET			
Diplex		130 ±10	
Listen only			115 ±10
X-band with maser	20 ±3	25.0 ±2.5	18.5 ±2

Table 4. DSCC telemetry s/s channel capabilities (telemetry groups 1 and 2)

Functions	Channel 1	Channel 2
Baseband Combining	N/A	N/A
Subcarrier Demodulation	100 Hz to 1 MHz, squarewave or sine wave	100 Hz to 1 MHz, squarewave or sine wave
Symbol Synchronization	6 s/s to 268.8 ks/s	6 s/s to 268.8 ks/s
Data Format	NRZ-L, NRZ-M, Biφ-L	NRZ-L, NRZ-M, Biφ-L
Sequential Decoding	K=24 or 32; R=1/2; frame length selectable; 16 s/s to 20 ks/s	N/A
Maximum Likelihood Convolutional Decoding	K=7; R=1/2 or 1/3 10 b/s to 134.4 kb/s	N/A
Uncoded	6 b/s to 268.8 kb/s*	6 b/s to 268.8 kb/s*

*Record only with non-real-time playback above 250 kbps

Table 5. DSCC telemetry s/s channel capabilities (telemetry groups 3 and 4)

Functions	Channel 1	Channel 2
Baseband Combining	Up to seven basebands	N/A
Subcarrier Demodulation	10 kHz to 2 MHz, squarewave or sine wave	10 kHz to 2 MHz, squarewave or sine wave
Symbol Synchronization	4 s/s to 4 Ms/s	4 s/s to 4 Ms/s
Data Format	NRZ-L, NRZ-M, Bi ϕ -L	NRZ-L, NRZ-M, Bi ϕ -L
Sequential Decoding	$K=24$ or 32 ; $R=1/2$; frame length selectable; 16 s/s to 20 ks/s	N/A
Maximum Likelihood Convolutional Decoding	$K=7$; $R=1/2$ or $1/3$ 10 bps to 250 kbps*	N/A
Uncoded	4 bps to 500 kbps*	4 bps to 500 kbps*
*Record only with non-real-time playback above 250 kbps		

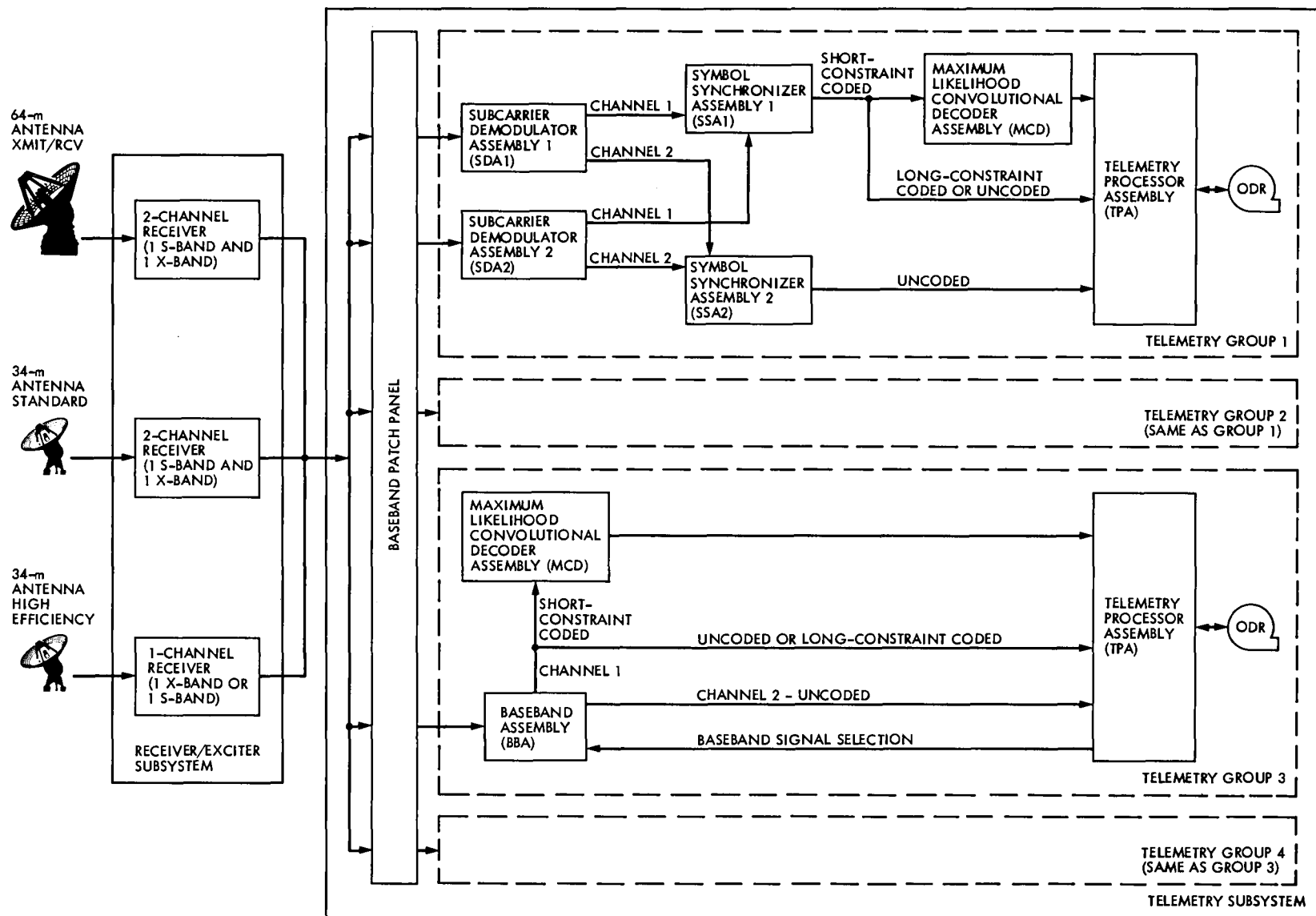


Fig. 1. Telemetry System: DSCC block diagram

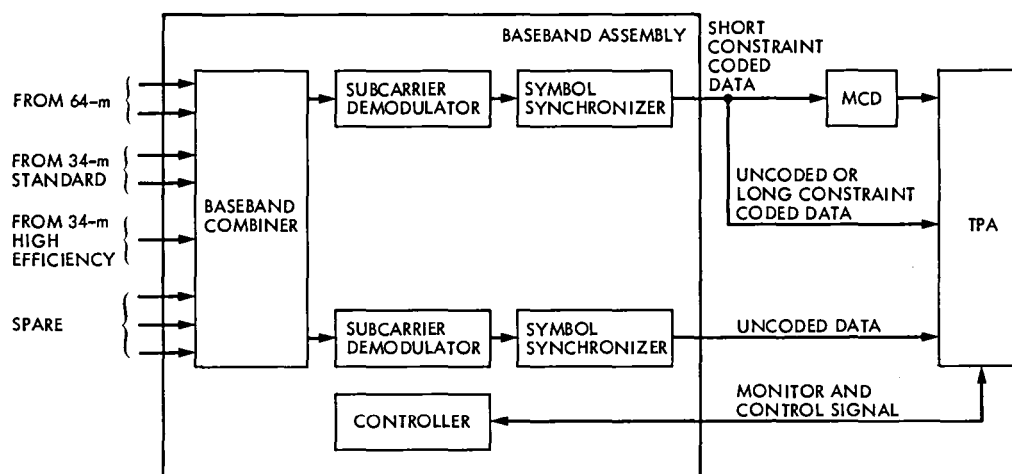


Fig. 2. Baseband Assembly functional block diagram

Comparison of GSFC and JPL VLBI Modeling Software: Benchmark

O. J. Sovers

Tracking Systems and Applications Section

C. Ma

Goddard Space Flight Center

The VLBI modeling software packages CALC 6.0 (GSFC) and MASTERFIT (JPL) are compared in some detail. Theoretical model delays are calculated for a set of 120 fictitious observations which involve a variety of sources, baselines, and antennas. Discrepancies between the total delays given by the two programs are of the order of 2 cm (RMS). The modeling of antenna offsets appears to account for approximately half of this difference. Relativistic bending contributions to the delay differ by 3 cm (RMS), and there appears to be some mutual cancellation of errors involving antenna offsets, bending, and the effects of the two different Solar System ephemerides employed by CALC and MASTERFIT. This cancellation has not been completely characterized in the present study.

I. Introduction

During the past decade, parallel and independent development of the VLBI technique for geodesy and astrometry has taken place on the U.S. East and West coasts. The respective focal points for software development are the Goddard Space Flight Center (GSFC) and Jet Propulsion Laboratory (JPL). The corresponding modeling software packages are known by the names of CALC (Refs. 1, 2) and MASTERFIT (Ref. 3). Station location estimates have reached the point of 1-cm formal errors. In order to gain assurance that there are no significant differences between the modeling software at the centimeter level, we undertook a detailed comparison. Previous efforts (Ref. 4; M. G. Roth, private communication, 1981) were not comprehensive enough to provide detailed

assessments of model differences at this level. At the time that this study was initiated (late 1984), there still existed discrepancies as large as 10 cm. Most of these appear to have been eliminated by the implementation of the Yoder short-period tidal UT1 corrections and the Shapiro relativistic corrections in going from version 5.0 to 6.0 of CALC.

The method we employ entails comparison of model delays yielded by CALC and MASTERFIT. Both are the versions in use on the GSFC HP 1000 and JPL VAX 11/785 computers in the Spring of 1985, identified as CALC 6.0 and MFIT.EXE;333, respectively. The MASTERFIT models, with minor exceptions, correspond to those described by Fanselow and Sovers (Ref. 3). A set of 120 fictitious ("benchmark")

observations was generated, employing hourly alternating observations of three sources with five baselines on April 1, 1982.

If care is taken to make the two models identical to the extent permitted by the existing code, differences of calculated delays are found to be 9–50 ps (3–15 mm), even for baselines nearly the length of an Earth diameter. To some extent, this agreement results from partial cancellation of larger discrepancies in parts of the models. In particular, apparent differences as large as 250 ps (8 cm) in the relativistic part of the delay remain unexplained. The comparison process uncovered one centimeter-level bug in MASTERFIT, and slight inadequacies of two models in CALC and MASTERFIT. The former involved inconsistent coordinate systems used to estimate the pole tide, while the latter concerned a singularity in the correction of the elevation angle for estimating the antenna axis offset effect in CALC, and corrections for a nonspherical Earth in the local to geocentric transformations of tidal displacements in MASTERFIT.

It is emphasized that the present benchmark study does not address delay rates, observable partials, or least-squares parameter estimation. We hope to compare all of these aspects of the software in the future. Indirect evidence from a variety of fits to real data, however, implies that no serious discrepancies exist. For example, universal time and polar motion values derived from independent experiments by the two groups are in good agreement with each other, as well as with the BIH compilation (Ref. 5); a transcontinental baseline measurement agrees at the centimeter level (Refs. 6, 7); both groups obtain a value close to 1.0 for the parametrized post-Newtonian theory gamma factor (Refs. 8, 7); positions of extragalactic radio sources derived from independent measurements by GSFC and JPL agree essentially within formal error estimates (Ref. 9).

Previous attempts at software comparison were not strictly analogous to the present study. Fliegel (Ref. 4) did an overall comparison of baseline results for Mobile VLBI experiments, but did not isolate components of the model. M. G. Roth (private communication, 1981) did investigate details of the JPL and GSFC models, but abandoned the comparison at the nanosecond level because both sets of code were in a state of flux at the time. In particular, MASTERFIT was being translated from an IBM to a VAX computer, while simultaneously incorporating the IAU 1984 resolutions.

II. Benchmark Observations

Three fictitious sources and five idealized stations were chosen for a set of benchmark observations on five baselines at hourly intervals over a 24-hour period: 1 April 1984 0 hr to

23 hr UT. The J2000 source coordinates are given in Table 1. They were chosen to span a wide range of right ascensions and declinations. The time of year provides a high probability of observing sources near the Sun, and thus magnifies discrepancies in gravitational bending models.

Station cartesian coordinates, antenna types, and axis offsets are listed in Table 2. Two of the stations (Haystack and Sweden) are routinely employed in the IRIS/POLARIS experiments (Ref. 10) while another two (Goldmars and Canberra) correspond to the approximate positions of the Deep Space Network American and Australian stations, and the last station (Pole) is entirely fictitious. The antenna characteristics were arbitrarily chosen to include a variety of types and large axis offsets to expose and magnify discrepancies in these aspects of modeling.

Table 3 shows the five baselines employed for hourly alternating observations of the sources of Table 1. Their lengths range from 4000 to 12,000 km: the Canberra-Sweden baseline is nearly equal to an Earth diameter.

Table 4 gives the observing schedule for each of the five baselines, making a total of 120 benchmark observations. It should be noted that approximately half of these observations force one or both stations to observe at negative elevation angles. Since the tropospheric delay model is turned off in both programs, and both assume a transparent Earth, this presents no problem. Magnitudes of the calculated delays range between 0.6 and 38.8 ms for these 120 observations.

III. Comparison of Model Delays

We first enumerate the known discrepancies in CALC and MASTERFIT models, in order to be able to present more systematic comparisons. These are as follows:

- (1) The Solar System ephemerides are different: CALC uses a version of the MIT PEP ephemeris rotated to J2000, while MASTERFIT uses the JPL ephemeris DE 200 (Ref. 11). Major effects of these differences are expected to be manifested in the solid Earth tide and gravitational bending contributions to the delay. Differences in the magnitudes of Earth-Sun vectors are of the order of 800 m, for example.
- (2) The treatment of the general relativistic bending of the ray path is totally different. In addition, MASTERFIT includes the Earth and all planets, while CALC only takes account of the Sun. The bending effect of the Earth is as large as 155 ps for the benchmark observations, while other planets contribute a maximum of 0.3 ps.

- (3) CALC corrects the source position employed in modeling antenna offset contributions for tropospheric bending with a simplified (tangent of the elevation angle) function. This correction uses nominal zenith troposphere values, and is present even when the tropospheric delay modeling is turned off. The standard MASTERFIT model ignores this, since it amounts to only a few mm at a 6-deg elevation for a 10-m offset.

A. Input Model Parameters

Care was taken to ensure that all parameters entering the two models were identical. In addition to the source and station parameters of Tables 1 and 2, these include the following:

Velocity of light	= 299792458 m/s
Relativistic gamma factor	= 1.0
Earth equatorial radius	= 6378.14 km
Flattening factor	= 298.257
Solid Earth tide Love numbers	= 0.609, 0.0852
Zenith tropospheric delay	= 0 (all stations)

Both CALC and MASTERFIT obtain pole position and universal time values for each observation by interpolating input tables. Since the interpolating functions are different (cubic and linear, respectively), we made sure that identical values were obtained by preparing idealized linear input tables in both cases. Table 5 shows the values of polar motion and UT1-UTC employed for the benchmark calculations. Differences in the interpolation algorithms are expected to contribute no more than a few mm to theoretical delay differences in unfavorable cases.

For the sake of completeness, Table 6 gives the parameters (amplitudes and phases) in the 11-component ocean loading model used in the benchmark calculations. Only the Haystack and Goldmars stations are assigned ocean loading displacements, and these are solely radial, since CALC 6.0 does not treat the longitudinal components. The absence of horizontal ocean loading displacements is expected to contribute only a few mm to calculated delays.

B. Comparison of Calculated Delays

Details of the observation-by-observation calculated delays are relegated to the figure in the Appendix. Note that the delays for one observation (No. 85) differ by approximately 130 ns. This is due to the above mentioned CALC correction of source position for tropospheric refraction, which is unreasonably large for the North Pole station, whose elevation angle is -0.012 deg. This observation has been excluded from all RMS quantities quoted below. We note that the overall discrepancy (excluding negative elevations) is 67 ps (20 mm). The

CALC-MASTERFIT difference does not appear to be correlated with baseline length, with the worst agreement (92 ps = 28 mm RMS) occurring for one of the shortest baselines (Goldstone to the North Pole).

In view of the above mentioned known model differences, we tried to isolate the major factor responsible for the 2-cm delay discrepancies. Three further comparisons were made, in which the effects of differences in tidal, relativistic bending, and antenna offset models were eliminated in turn. For tides, the MASTERFIT contributions were subtracted from the MASTERFIT model delays and the CALC contributions added; for bending, all contributions except that of the Sun were subtracted from MASTERFIT; for antenna offsets, the axis offsets of Table 2 were set to zero in both programs. A further comparison included all three modifications simultaneously. The results are tabulated in Table 7 ("all" and "+ only" indicate all 120 observations, and only those with positive elevation angles at both stations, respectively). It is clear that discrepancies in tidal and bending models make little if any contribution to the raw differences.

The identical treatment of antenna offsets, on the other hand, is seen to reduce the RMS difference to 40% of its raw value, and the maximum difference by nearly a factor of 5. It must be stressed that the benchmark comparison is a stringent test, involving a majority of observations at elevations and antenna offsets which would never be encountered in real data. In view of this, the 26 ps RMS and 50 ps maximum discrepancies (approximately 8 and 15 mm) are gratifyingly small.

Further numerical quantification of CALC-MASTERFIT differences was performed in three areas: tidal, gravitational bending, and tropospheric contributions. Table 8 presents a comparison of delay contributions and station (baseline) displacements due to the various tidal components, for all 120 observations. In contrast to Table 7, which exhibits only the total model delays, Table 8 is concerned with component parts of the delays. It is seen that only the solid Earth tide differences are substantial, while there is nearly perfect agreement for the K1 correction (Ref. 12), pole tide, and ocean loading. At present it is not clear whether the solid tide differences are due to ephemeris differences or to computational algorithms.

For relativistic bending, the understanding and reconciliation of the two models will require more work than we were prepared to perform in this study. We attempted, however, to isolate the components of the delay that depend solely on the Parametrized Post-Newtonian (PPN) parameter gamma, namely the relativistic bending. When this is done, comparison shows that the RMS relativistic delay difference is 92 ps (3 cm), with a maximum of 254 ps (8 cm). Contrast this with the

corresponding numbers for agreement of total delay: 21 ps (6 mm) and 49 ps (15 mm).

While tropospheric delay is outside the scope of this comparison, and is not included in any of the comparisons discussed above, we took the opportunity to compare this aspect of modeling as well. Both CALC and MASTERFIT were set to employ the model of Chao (Ref. 13), which is now known to be inadequate for VLBI purposes. The RMS CALC-MASTERFIT tropospheric delay difference for the benchmark set was 28 ps, with a maximum of 106 ps (8 and 32 mm, respectively). Only observations at positive elevation angles were included. These results are surprisingly large for a formula which is algebraically simple; at present, no explanation is evident. Naturally a large contribution to the discrepancy is made by the substantial fraction of benchmark observations that involve very low elevation angles. We hope to perform a more thorough comparison of the new Lanyi

(Ref. 14) and CFA (Ref. 15) tropospheric mapping functions in the near future.

IV. Conclusions

Comparison of GSFC and JPL VLBI software shows agreement at the 1–2 cm level. Included are all components of the model in common use at the beginning of 1985, with the exception of ionospheric and tropospheric delays. Three areas of remaining discrepancy are the use of different Solar System ephemerides, different methods of estimating gravitational bending, and different corrections for tropospheric refraction in calculating antenna offset effects. Millimeter-size discrepancies could also potentially arise due to the difference in interpolating algorithms for UT1 and polar motion *a priori* values. It is hoped to reconcile these differences during the coming year, and to perform similar comparisons of the partial derivative and parameter estimation sections of the respective codes.

References

1. Ryan, J., D. Gordon and C. Ma, *CALC 6.0 Released*, Goddard Space Flight Center, Technical Memorandum, Jan. 15, 1985.
2. GSFC VLBI Group, *CALC, Version 6.0*, Goddard Space Flight Center, Greenbelt, Maryland, 30 Jan. 1985.
3. Fanselow, J. L. and O. J. Sovers, Observation Model and Parameter Partial for the JPL VLBI Parameter Estimation Software "MASTERFIT" – V2.0, *JPL Publication 83-39, Rev. 1*, Jet Propulsion Laboratory, Pasadena, Calif., 1985.
4. Fliegel, H. F., Session III of the VLBI/Laser Intercomparison Task of the NASA Crustal Dynamics Project, *JPL Publication 81-96*, Jet Propulsion Laboratory, Pasadena, Calif., 1981.
5. Eubanks, T. M., M. A. Spieth and J. A. Steppe, Accuracy of Radio Interferometric Estimates of Earth Rotation, *TDA Progress Report 42-80*, pp. 229–235, Jet Propulsion Laboratory, Pasadena, Calif., 1985.
6. Rogers, A. E. E., R. J. Cappallo, H. F. Hinteregger, J. I. Levine, E. F. Nesman, J. C. Webber, A. R. Whitney, T. A. Clark, C. Ma, J. Ryan, B. E. Corey, C. C. Counselman, T. A. Herring, I. I. Shapiro, C. A. Knight, D. B. Shaffer, N. R. Vandenberg, R. Lacasse, R. Mauzy, B. Rayhrer, B. R. Schupler, and J. C. Pigg, Very-Long-Baseline Interferometry: The Mark III System for Geodesy, Astrometry and Aperture Synthesis, *Science* 219, 51–54, 1983.
7. Sovers, O. J., J. B. Thomas, J. L. Fanselow, E. J. Cohen, G. H. Purcell, Jr., D. H. Rogstad, L. J. Skjerve and D. J. Spitzmesser, Radio Interferometric Determination of Intercontinental Baselines and Earth Orientation Utilizing Deep Space Network Antennas – 1971 to 1980, *J. Geophys. Res.* 89, 7597–7607, 1984.

8. Robertson, D. S. and W. E. Carter, Relativistic Deflection of Radio Signals in the Solar Gravitational Field Measured with VLBI, *Nature* 310, 572-574, 1984.
9. Fanselow, J. L., O. J. Sovers, J. B. Thomas, G. H. Purcell, Jr., E. J. Cohen, D. H. Rogstad, L. J. Skjerve and D. J. Spitzmesser, Radio Interferometric Determination of Source Positions Utilizing Deep Space Network Antennas – 1971 to 1980, *A. J.* 89, 987-998, 1984.
10. Carter, W. E., and D. S. Robertson, IRIS Earth Rotation and Polar Motion Results, presented at the International Symposium on Space Techniques for Geodesy, Sopron, Hungary, 1984.
11. Standish, E. M., Orientation of the JPL Ephemeris, DE200/LE200, to the Dynamical Equinox of J2000, *Astron. Astrophys.* 114, 79-106, 1982.
12. Melbourne, W. G., R. Anderle, M. Feissel, R. King, D. McCarthy, D. Smith, B. Tapley and R. Vicente, Project MERIT Standards, *USNO Circular No. 167*, U.S. Naval Observatory, Washington, D. C., Dec. 27, 1983.
13. Chao, C. C., The Tropospheric Calibration Model for Mariner Mars 1971, *Tech. Rep.* 32-1587, 61-76, Jet Propulsion Laboratory, Pasadena, Calif., 1974.
14. Lanyi, G. E., Tropospheric Delay Effects in Radio Interferometry, *TDA Progress Report* 42-78, pp. 152-159, Jet Propulsion Laboratory, Pasadena, Calif., April 1984.
15. Davis, J. and T. Herring, *New Atmospheric Mapping Function*, Harvard College Observatory, Smithsonian Astrophysical Observatory, Technical Memorandum, July 30, 1984.

Table 1. Benchmark source positions

Source	RA, hms			Dec, dms		
T1	0	2	33.76405	0	16	42.2454
T2	6	3	12.33782	29	59	52.9881
T3	12	2	33.12210	59	43	17.7545

Table 2. Benchmark station coordinates, antenna types and offsets

Station	x, km	y, km	z, km	Type	Offset, m
HAYSTACK	1492.407	-4457.267	4296.882	EQU	10
GOLDMARS	-2353.620	-4641.343	3677.053	X-Y	20
CANBERRA	-4446.247	2684.627	-3691.949	X-Y	40
SWEDEN	3370.968	711.466	5349.664	AZEL	2.15
POLE	-3796.424	-104.306	6360.000	AZEL	50

Table 3. Benchmark baselines and lengths

Baseline	Length, km
HAY-GOL	3900
GOL-CAN	10599
GOL-SWE	8014
GOL-POL	5465
CAN-SWE	12114

Table 4. Benchmark observations

Time		Source	Time		Source
82/4/1	0	T1	82/4/1	12	T1
	1	T2		13	T2
	2	T3		14	T3
	3	T1		15	T1
	4	T2		16	T2
	5	T3		17	T3
	6	T1		18	T1
	7	T2		19	T2
	8	T3		20	T3
	9	T1		21	T1
	10	T2		22	T2
	11	T3		23	T3

Table 5. Polar motion and UT1 values for benchmark

Julian Date	Polar Motion, mas		UT1-UTC, msec
	x	y	
2445054	78.0	436.0	-170.0
2445059	90.0	432.0	-182.5
2445064	102.0	428.0	-195.0
2445069	114.0	424.0	-207.5

Table 6. Ocean loading model for benchmark

Component	Station	Amplitude, cm	Phase, deg	Station	Amplitude, cm	Phase, deg
M2	HAYST	0.96	183.2	GOLDM	0.23	351.5
S2		0.26	205.2		0.14	250.3
N2		0.21	165.2		0.12	305.7
K2		0.07	198.4		0.05	276.9
K1		0.39	354.6		0.99	43.8
O1		0.26	355.2		0.62	28.9
P1		0.12	356.8		0.31	43.1
Q1		0.05	358.3		0.12	20.0
MF		0.04	10.5		0.01	295.6
MM		0.02	59.2		0.01	104.9
SSA		0.04	264.0		0.06	74.2

Table 7. Total model delay differences between CALC and MASTERFIT

Type of Comparison	RMS, ps		Maximum, ps	
	All	+ Only	All	+ Only
Raw	73.2	67.4	374.1	237.8
Identical tides, (a)	81.8	64.7	374.4	238.7
Solar bending only, (b)	75.6	67.0	357.8	244.5
Zero antenna offsets, (c)	21.2	26.4	48.7	48.7
(a) + (b) + (c)	38.8	25.5	157.0	50.2

Table 8. Tidal delay and displacement differences between CALC and MASTERFIT

Quantity	RMS difference, ps		Maximum difference, ps		
			Delay	Displacement	
	Delay	Displacement		Local	E-Cent.
Solid tides	7.6	6.9	25.2	22.1	17.2
K1 correction	...	0.12	0.18
Pole tide	0.05	0.05	0.10	0.13	0.10
Ocean loading	0.001	0.001	0.004	0.001	0.001

Appendix

Figure A-1 is a tabulation of “raw” model delays calculated with CALC 6.0 and MFIT.EXE; 333 and their differences. Listed are the observation number, baseline, source, time (ymdh), CALC and MASTERFIT total delays, and their difference (CALC minus MASTERFIT). An asterisk (*) following the source name indicates an observation for which one or both elevation angles are zero or negative. For each baseline, the root mean square difference between CALC and MASTERFIT is calculated for all observations as well as for only those at positive elevation angles.

				Model delay (psec)		DIF.
Observation				CALC 6.0	MFIT	
1	HAY-GOL	T1*	82/4/1 0	-12585612701.157	-12585612711.389	10.232
2	HAY-GOL	T2	82/4/1 1	-3893892543.696	-3893892530.870	-12.827
3	HAY-GOL	T3	82/4/1 2	6627974849.523	6627974868.641	-19.118
4	HAY-GOL	T1*	82/4/1 3	-7085396676.466	-7085396673.550	-2.916
5	HAY-GOL	T2	82/4/1 4	-9509538985.199	-9509538967.877	-17.322
6	HAY-GOL	T3	82/4/1 5	2175372246.348	2175372227.878	18.471
7	HAY-GOL	T1*	82/4/1 6	2590632122.327	2590632144.998	-22.672
8	HAY-GOL	T2*	82/4/1 7	-8916992949.671	-8916992987.427	37.756
9	HAY-GOL	T3	82/4/1 8	-2505749430.424	-2505749424.881	-5.544
10	HAY-GOL	T1*	82/4/1 9	10744906794.568	10744906779.750	14.818
11	HAY-GOL	T2*	82/4/1 10	-2465150182.063	-2465150183.234	1.171
12	HAY-GOL	T3	82/4/1 11	-4659003876.422	-4659003817.997	-58.425
13	HAY-GOL	T1*	82/4/1 12	12575954242.330	12575954240.011	2.319
14	HAY-GOL	T2*	82/4/1 13	6046931392.195	6046931385.644	6.551
15	HAY-GOL	T3	82/4/1 14	-3016481033.448	-3016480965.839	-67.608
16	HAY-GOL	T1	82/4/1 15	7005622254.772	7005622271.324	-16.552
17	HAY-GOL	T2*	82/4/1 16	11607049844.123	11607049889.674	-45.551
18	HAY-GOL	T3	82/4/1 17	1454657529.508	1454657622.209	-92.700
19	HAY-GOL	T1	82/4/1 18	-2686119354.696	-2686119383.565	28.869
20	HAY-GOL	T2	82/4/1 19	10941265333.850	10941265312.359	21.491
21	HAY-GOL	T3	82/4/1 20	6121656289.089	6121656434.588	-145.499
22	HAY-GOL	T1	82/4/1 21	-10792502104.886	-10792502137.607	32.721
23	HAY-GOL	T2	82/4/1 22	4441622393.791	4441622351.781	42.010
24	HAY-GOL	T3	82/4/1 23	8236427202.930	8236427232.913	-29.983
RMS (no *; all)					54.316	45.023
				Model delay (psec)		DIF.
Observation				CALC 6.0	MFIT	
25	GOL-C	T1	82/4/1 0	-10489285192.723	-10489285177.502	-15.221
26	GOL-C	T2*	82/4/1 1	29273449598.709	29273449578.182	20.527
27	GOL-C	T3*	82/4/1 2	31666616616.733	31666616598.619	18.114
28	GOL-C	T1*	82/4/1 3	-23758218846.926	-23758218840.067	-6.859
29	GOL-C	T2	82/4/1 4	14353178847.841	14353178834.335	13.506
30	GOL-C	T3*	82/4/1 5	33839550962.713	33839550954.473	8.240
31	GOL-C	T1*	82/4/1 6	-22993320249.619	-22993320252.856	3.237
32	GOL-C	T2	82/4/1 7	-1782046094.129	-1782046038.673	-55.456
33	GOL-C	T3*	82/4/1 8	28598744840.469	28598744842.616	-2.147
34	GOL-C	T1*	82/4/1 9	-8644924563.615	-8644924540.795	-22.820
35	GOL-C	T2*	82/4/1 10	-9631363846.549	-9631363822.950	-23.598
36	GOL-C	T3*	82/4/1 11	19030165408.978	19030165470.940	-61.962
37	GOL-C	T1*	82/4/1 12	10838168545.026	10838168579.929	-34.903
38	GOL-C	T2*	82/4/1 13	-4572797071.631	-4572797084.982	13.350
39	GOL-C	T3*	82/4/1 14	10768069531.403	10768069868.862	-337.459
40	GOL-C	T1*	82/4/1 15	23983661117.362	23983661126.302	-8.940
41	GOL-C	T2*	82/4/1 16	10415056242.877	10415056224.272	18.606
42	GOL-C	T3*	82/4/1 17	8677378465.377	8677378510.634	-45.257
43	GOL-C	T1*	82/4/1 18	23051112702.293	23051112723.516	-21.223
44	GOL-C	T2*	82/4/1 19	26506825493.578	26506825517.828	-24.250
45	GOL-C	T3*	82/4/1 20	13989229081.622	13989228994.326	87.296
46	GOL-C	T1	82/4/1 21	8589695859.210	8589695891.955	-32.746
47	GOL-C	T2*	82/4/1 22	34227153040.586	34227153078.053	-37.467
48	GOL-C	T3*	82/4/1 23	23575839561.394	23575839553.200	8.194
RMS (no *; all)					33.770	75.894

Fig. A-1. "Raw" model delays

				Model delay (psec)		
Observation				CALC 6.0	MFIT	DIF.
49	GOL-SWE	T1*	82/4/1 0	16185720822.098	16185720828.309	-6.211
50	GOL-SWE	T2	82/4/1 1	17993984333.845	17993984317.336	16.509
51	GOL-SWE	T3	82/4/1 2	-6704573201.099	-6704573227.812	26.714
52	GOL-SWE	T1*	82/4/1 3	-3122544321.856	-3122544319.901	-1.955
53	GOL-SWE	T2*	82/4/1 4	18237985299.639	18237985268.271	31.368
54	GOL-SWE	T3	82/4/1 5	3065860092.861	3065860066.385	26.476
55	GOL-SWE	T1*	82/4/1 6	-20602185597.935	-20602185597.695	-0.240
56	GOL-SWE	T2	82/4/1 7	6100368835.672	6100368799.121	36.551
57	GOL-SWE	T3	82/4/1 8	8191086555.686	8191086515.249	40.436
58	GOL-SWE	T1*	82/4/1 9	-25960663261.192	-25960663247.461	-13.731
59	GOL-SWE	T2*	82/4/1 10	-11271861472.590	-11271861505.598	33.008
60	GOL-SWE	T3	82/4/1 11	5653219083.324	5653219027.937	55.386
61	GOL-SWE	T1*	82/4/1 12	-16042735574.646	-16042735569.644	-5.003
62	GOL-SWE	T2*	82/4/1 13	-23649410272.859	-23649410314.626	41.767
63	GOL-SWE	T3	82/4/1 14	-3053364888.251	-3053364924.789	36.538
64	GOL-SWE	T1	82/4/1 15	3311626032.342	3311626024.955	7.387
65	GOL-SWE	T2*	82/4/1 16	-23743982434.421	-23743982452.398	17.977
66	GOL-SWE	T3	82/4/1 17	-12801967736.361	-12801967783.033	46.672
67	GOL-SWE	T1*	82/4/1 18	20705974196.957	20705974220.002	-23.045
68	GOL-SWE	T2	82/4/1 19	-11499896861.510	-11499896828.546	-32.964
69	GOL-SWE	T3	82/4/1 20	-17852311291.384	-17852311418.687	127.303
70	GOL-SWE	T1*	82/4/1 21	25897978015.240	25897978051.196	-35.956
71	GOL-SWE	T2	82/4/1 22	5873154425.261	5873154471.908	-46.647
72	GOL-SWE	T3	82/4/1 23	-15230584157.899	-15230584183.491	25.592
				RMS (no *; all)	49.160	39.601
				Model delay (psec)		
Observation				CALC 6.0	MFIT	DIF.
73	GOL-P	T1	82/4/1 0	5340718523.284	5340718285.470	237.814
74	GOL-P	T2	82/4/1 1	10213082801.243	10213082739.583	61.660
75	GOL-P	T3	82/4/1 2	-6011503170.623	-6011503231.485	60.862
76	GOL-P	T1*	82/4/1 3	-7663353656.411	-7663353892.195	235.784
77	GOL-P	T2	82/4/1 4	6522352926.267	6522352851.542	74.725
78	GOL-P	T3	82/4/1 5	-591502606.599	-591502666.631	60.032
79	GOL-P	T1*	82/4/1 6	-16171214562.317	-16171214649.323	87.006
80	GOL-P	T2	82/4/1 7	-3643744283.646	-3643744350.026	66.380
81	GOL-P	T3	82/4/1 8	621339818.869	621339750.440	68.430
82	GOL-P	T1*	82/4/1 9	-15173163478.887	-15173163572.408	93.521
83	GOL-P	T2*	82/4/1 10	-14299105695.405	-14299105764.133	68.728
84	GOL-P	T3	82/4/1 11	-3087131113.445	-3087131192.689	79.244
85	GOL-P	T1*	82/4/1 12	-5256757234.665	-5256887060.910	129826.244
86	GOL-P	T2*	82/4/1 13	-19169522762.562	-19169522843.099	80.537
87	GOL-P	T3	82/4/1 14	-9533257528.390	-9533257592.090	63.700
88	GOL-P	T1	82/4/1 15	7738596413.005	7738596312.038	100.968
89	GOL-P	T2*	82/4/1 16	-15387123092.829	-15387123162.407	69.579
90	GOL-P	T3	82/4/1 17	-14921370529.279	-14921370612.200	82.921
91	GOL-P	T1	82/4/1 18	16161150625.774	16161150583.615	42.159
92	GOL-P	T2	82/4/1 19	-5179119029.039	-5179119053.382	24.343
93	GOL-P	T3	82/4/1 20	-16078778204.958	-16078778369.709	164.751
94	GOL-P	T1	82/4/1 21	15051330462.976	15051330421.856	41.119
95	GOL-P	T2	82/4/1 22	5443695609.533	5443695604.304	5.229
96	GOL-P	T3	82/4/1 23	-12323948800.114	-12323948858.500	58.386
				RMS (no *; all)	92.159	100.509

Fig. A-1 (contd)

Observation				Model delay (psec)		DIF.
				CALC 6.0	MFIT	
97 C	-SWE T1*	82/4/1	0	26675008002.348	26675007993.396	8.952
98 C	-SWE T2*	82/4/1	1	-11279514315.827	-11279514311.919	-3.908
99 C	-SWE T3*	82/4/1	2	-38371202775.555	-38371202784.196	8.641
100 C	-SWE T1*	82/4/1	3	20635644839.992	20635644835.129	4.864
101 C	-SWE T2*	82/4/1	4	3884792300.511	3884792282.556	17.956
102 C	-SWE T3*	82/4/1	5	-30773722146.810	-30773722165.147	18.337
103 C	-SWE T1	82/4/1	6	2391089752.881	2391089756.316	-3.435
104 C	-SWE T2	82/4/1	7	7882414423.181	7882414331.183	91.998
105 C	-SWE T3*	82/4/1	8	-20407683883.853	-20407683926.588	42.734
106 C	-SWE T1*	82/4/1	9	-17315751717.739	-17315751726.858	9.119
107 C	-SWE T2	82/4/1	10	-1640510985.784	-1640511042.366	56.582
108 C	-SWE T3*	82/4/1	11	-13376952774.638	-13376952892.106	117.468
109 C	-SWE T1*	82/4/1	12	-26880902248.027	-26880902277.886	29.859
110 C	-SWE T2*	82/4/1	13	-19076620852.110	-19076620880.523	28.413
111 C	-SWE T3*	82/4/1	14	-13821429930.569	-13821430304.624	374.054
112 C	-SWE T1*	82/4/1	15	-20672065362.659	-20672065378.922	16.263
113 C	-SWE T2*	82/4/1	16	-34159028532.921	-34159028532.285	-0.636
114 C	-SWE T3*	82/4/1	17	-21479338156.173	-21479338248.132	91.959
115 C	-SWE T1*	82/4/1	18	-2345183551.590	-2345183549.819	-1.771
116 C	-SWE T2*	82/4/1	19	-38006730272.234	-38006730263.484	-8.751
117 C	-SWE T3*	82/4/1	20	-31841527901.186	-31841527941.220	40.035
118 C	-SWE T1*	82/4/1	21	17308269312.116	17308269315.262	-3.146
119 C	-SWE T2*	82/4/1	22	-28354046379.223	-28354046370.044	-9.180
120 C	-SWE T3*	82/4/1	23	-38806415917.829	-38806415935.253	17.424
RMS (no *; all)					62.389	86.760
RMST (no *; all)					67.410	73.194
MAX (no *; all)					237.814	374.054

Fig. A-1 (contd)

Controller and Interface Module for the High-Speed Data Acquisition System Correlator/Accumulator

S. S. Brokl

Communications Systems Research Section

One complex channel of the High-Speed Data Acquisition System (a subsystem used in the Goldstone solar system radar), consisting of two correlator modules and one accumulator module, is operated by the controller and interface module. Interfaces are provided to the VAX UNIBUS for computer control, monitor, and test of the controller and correlator/accumulator. The correlator and accumulator modules controlled by this module are the key digital signal processing elements of the Goldstone High-Speed Data Acquisition System. This fully programmable unit provides for a wide variety of correlation and filtering functions operating on a three megaword/second data flow. Data flow is to the VAX by way of the I/O port of a FPS 5210 array processor.

I. Introduction

A new digital signal processing and data acquisition system is installed in the pedestal room at DSS-14 at Goldstone in support of radio astronomy and solar system radar experiments. Figure 1 is an overall block diagram for this system. This article is about the correlator assembly; the complete system is described in *TDA Progress Report 42-77* (Ref. 1).

The Controller and Interface board (CI) ties together four major components of the High Speed Data Acquisition System (HSDAS): the VAX 11/780 computer (by way of the UNIBUS), the correlators, the accumulators, and the array processor (Ref. 1). Additionally, the controller provides the timing and distribution of clock and control signals to the two correlators and one accumulator board, and provides the interface path to the array processor. Together the four modules comprise two data channels (or one complex data channel) which make up one Complex System Unit (CSU).

Control paths are provided from the VAX 11/780 computer to the correlators and accumulators (Ref. 2), and data paths into the Floating-Point System array processor (FPS 5210) (by way of the General Purpose Programmable I/O Processor [GPIOP]) are also provided. Each correlator can be programmed to compute as many as 256 lags of four-bit cross-correlation and autocorrelation data using a single-bit reference function and single-bit mask function. Also, each correlator is usable as a digital finite impulse response (FIR) filter. The CI is contained on one standard universal wire wrapping board type 8136-UG6 manufactured by Augat Inc. and holds the 150 integrated circuits (Fig. 8) used.

Figure 2 is a block diagram showing the control and data paths to and from the CI, and therein we see its relation to the correlators and accumulators in the system. The major control path is from the VAX 11/780 by way of a DR11C I/O port. Through this port, one configures and runs the

correlators and accumulators in the desired modes. A discussion of the operating modes is in Section II below. An additional port from the GPIOP is used to monitor the status of the accumulator memories. Through this port GPIOP software commands the accumulator memory contents to be transferred using direct memory access (DMA) to the array processor memory. Each accumulator is double buffered and has a FULL flag and an OVERFLOW flag. The FULL flag is monitored by the GPIOP and indicates when data is ready to be unloaded from the accumulator. The OVERFLOW flag indicates that the accumulator memory was not unloaded in time and has been at least partially overwritten with new data. The operation of the DMA and flags is explained in Section IV, below.

II. Computer Control Paths

The major function of the CI is to distribute data and system clock signals to the correlators and accumulators and to control their operation independently of the computer (see Fig. 3, a block diagram for the CI). A 256×12 bit memory is included for downloading of test data and filter and mask functions into the correlators from the VAX. Control functions are downloaded from the VAX by way of a DR11C I/O port (a UNIBUS compatible interface used with several devices in the radar system, Ref. 3). The Polynomial Driven Time Base and PN Generator (Ref. 3) provide the time base and coding signals for the CI. Figures 4 and 5 illustrate the Control and Status Register (CSR) and the internal function registers accessible through the DR11C port to the CI. Note that one complex channel is serviced by one controller (see Figs. 9 and 10). Figure 3 illustrates the lag counters and registers and the accumulation counters and registers.

The Correlator/Accumulator (Cor/Acc) control and status register (CSR) (Fig. 4) is a broadcast register which is used to communicate individually with all function registers in the selected Cor/Acc, and is therefore received and decoded on all CI boards. Bits 6 through 8 determine which CI board is selected. The word pointer bits 0 through 2 determine which function register is selected. Bits 4 and 5 set the automatic increment of the word pointer and are used for multiple reads or writes from the computer. The CLEAR bit (position 10) is broadcast to all CI boards and is used to initialize all internal control functions and flags to a known inactive state throughout the system. MASTER RUN (bit 15) is broadcast to all CI boards throughout the system and starts the Cor/Acc operation at a known 1 second epoch.

The 256×12 CI memory is used to hold correlator test data and filter and mask coefficients. Although only 12 bits are used, it is organized as if it were two bytes using only 6 bits each and numbered as if all 16 bits were present. Bits 0

through 3 of the first byte contain data for correlator A, bit 4 is the mask, and bit 5 is the code for correlator A. Bits 8 through 11 of the second byte contain data for correlator B, with bit 12 as the mask and bit 13 as the code. The memory has two address generators. Each is multiplexed and split so that each byte can be independently downloaded into the correlator. The addresses are organized so that address 00 corresponds to the last data downloaded into the correlator. The address generators are strobed by the system clock during downloading. The computer reads and writes both bytes and is not split in addressing. It uses the function register 0 (Fig. 5) as a pointer to the buffer memory location to be written. Function register 1 is a window into the buffer memory. The computer address generator location may be incremented automatically while writing to the window permitting a block data transfer from the computer. The mask bits allow programming any length of correlation function up to 256 lags. The reference bits allow any single bit function up to 256 lags to be loaded into the correlator. By using the mask and reference bits the correlator can run also as a binary FIR filter.

The function register 2 controls mode and data downloading for correlator A from the buffer memory. Function register 3 controls correlator B in the same way as A. Each correlator can run in three separate modes and can accept data from two separate sources selected by bit 15. In the LONG CODE AUTO mode (bit 14 on), the correlator runs continuously, updating the reference input modulo 256. In the SHORT CODE mode the correlator runs continuously modulo the number set as maximum in function register 4. In both modes the data and reference are continuously streaming. In the SHORT CODE mode the mask (bit 13) must be loaded for the selected length set in register 4. In the FIR FILTER mode the reference and mask are downloaded from the buffer prior to running, and bits 14 and 13 are not set. In this mode the reference is not updated and the correlators run as an FIR filter. The ZERO BUFFER ADDRESS bit 12 is used prior to downloading reference data or masks into the correlator from the buffer memory. The AUTO LOAD FROM CODER bit (3) is used for testing and for streaming data into the correlator reference directly from the coder between word detects. Each of the bits 0 through 2 allows for selectively downloading segments of the buffer memory into the correlator without altering the contents of an unselected segment. Function register 4 defines the maximum lag that will be used by the correlator/accumulator pairs. The first byte controls accumulator A and correlator A. Similarly, the second byte controls accumulator B and correlator B. In the LONG CODE AUTO mode both bytes would be loaded with FF Hex (i.e., all 256 lags are used).

Function registers 5 and 6 specify the number (N) of coherent sums computed for each lag in accumulator A and B

respectively. Function register 7 enables and clears the accumulator's internal address generators and must be set properly prior to running.

III. Data Paths

Figure 2 best illustrates the data path connections into and out of the CI. The input multiplexer allows for choosing one of two inputs to each correlator. The input word consists of four bits of data, one bit for reference, and one bit for external loading of the mask. The number system is offset binary (Ref. 2), and the size of each word out of each correlator lag is 12 bits. This means that the accumulator can add up to 65,536 samples of 12-bit wide data, giving a maximum result 28 bits wide. The GPIOP accepts up to 38 bits in word width. Each correlator yields a 256 lag by 6-bit output (4 data bits, one reference bit, and one mask bit) which can be easily passed to other CSU modules to extend the number of lags.

IV. GPIOP Array Processor Interface

This CI module provides high-speed DMA transfer of data from the accumulator memories to the FPS 5210 Array Processor via the GPIOP I/O channel. The burst transfer rate is 3.0 million samples/sec with no reformatting of data to floating point in the GPIOP. With reformatting, the burst transfer rate is 1.5 million samples/sec. The burst length is dependent upon the size of the array processor memory (now 256K words deep).

Figure 6 illustrates the GPIOP Command and Device Control words and Data Format words. Bits 17 through 19 select

which CI module is accessed and bit 16 selects accumulator A or B for DMA read. Bits 13 through 15 are command bits that define the following functions:

NULL	No request
REPORT	Tests response of device status; Ready = 1 or Overflow = 1
SEND	Sets selected device for DMA transfers
CLEAR	Clears Ready and Overflow bits to false
INIT	Zeros address register in the Accumulator

It is important to note that the NULL command must precede each of the other commands to prevent spurious transient "commands" in the CI interface.

V. Summary

The CI modules control individual CI units, providing the node which ties the VAX, in its controlling and monitoring tasks, to the remainder of the high-speed data acquisition system. These modules provide for distribution of A/D converter input, accumulation control (by way of the GPIOP), system clock distribution, and system control. The complete HSDAS will consist of 8 CSU modules configured with a multiplexer as shown in Fig. 7. Three channels have been completed and tested. They were used during the April/May Venus inferior conjunction radar experiment at Goldstone. That series of measurements confirmed that system operation is correct. Completion of the remaining 5 CSU modules is in progress.

References

1. Deutsch, L. J., Jurgens, R. F., Brokl, S. S., "Goldstone R/D High Speed Data Acquisition System." *The Telecommunications and Data Acquisition Progress Report 42-77*, Jet Propulsion Laboratory, Pasadena, Calif., Jan.-Mar. 1984, pp. 87-96.
2. Brokl, S. S., "Demodulator and Accumulator for the High Speed Data Acquisition System." *The Telecommunications and Data Acquisition Progress Report 42-77*, Jet Propulsion Laboratory, Pasadena, Calif., Jan.-Mar. 1984, pp. 97-103.
3. Brokl, S. S., "Polynomial Driven Time Base and PN Generator," *The Telecommunications and Data Acquisition Progress Report 42-75*, Jet Propulsion Laboratory, Pasadena, Calif., July-Sept. 1983, pp. 84-90.

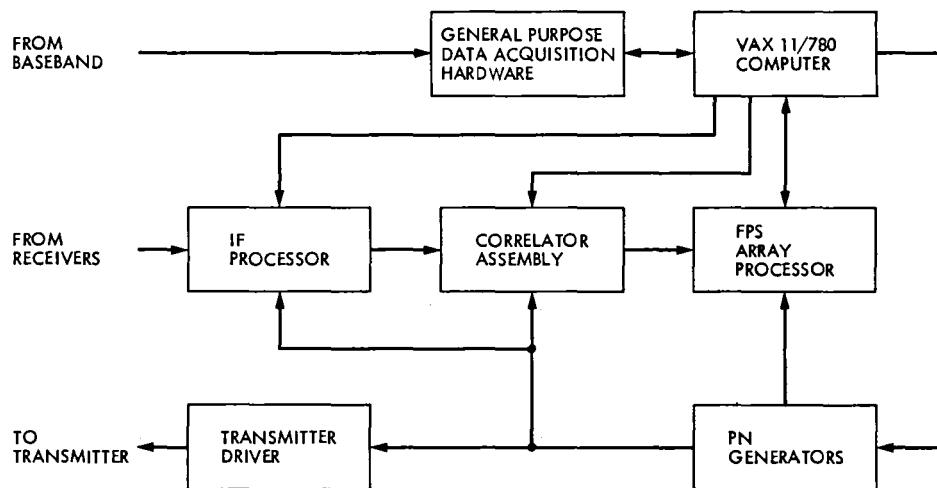


Fig. 1. High Speed Data Acquisition System overall block diagram

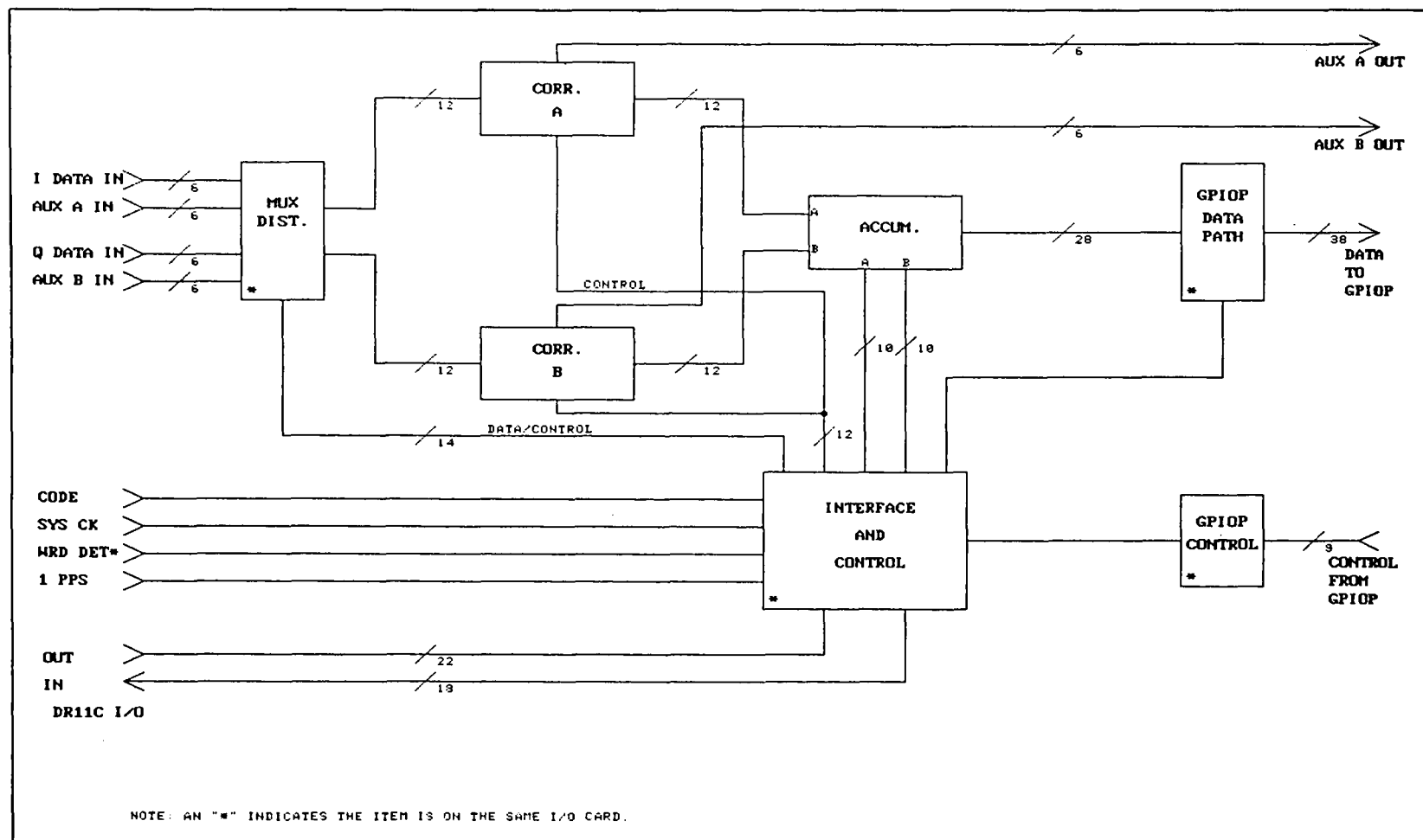


Fig. 2. Correlator/Accumulator block diagram

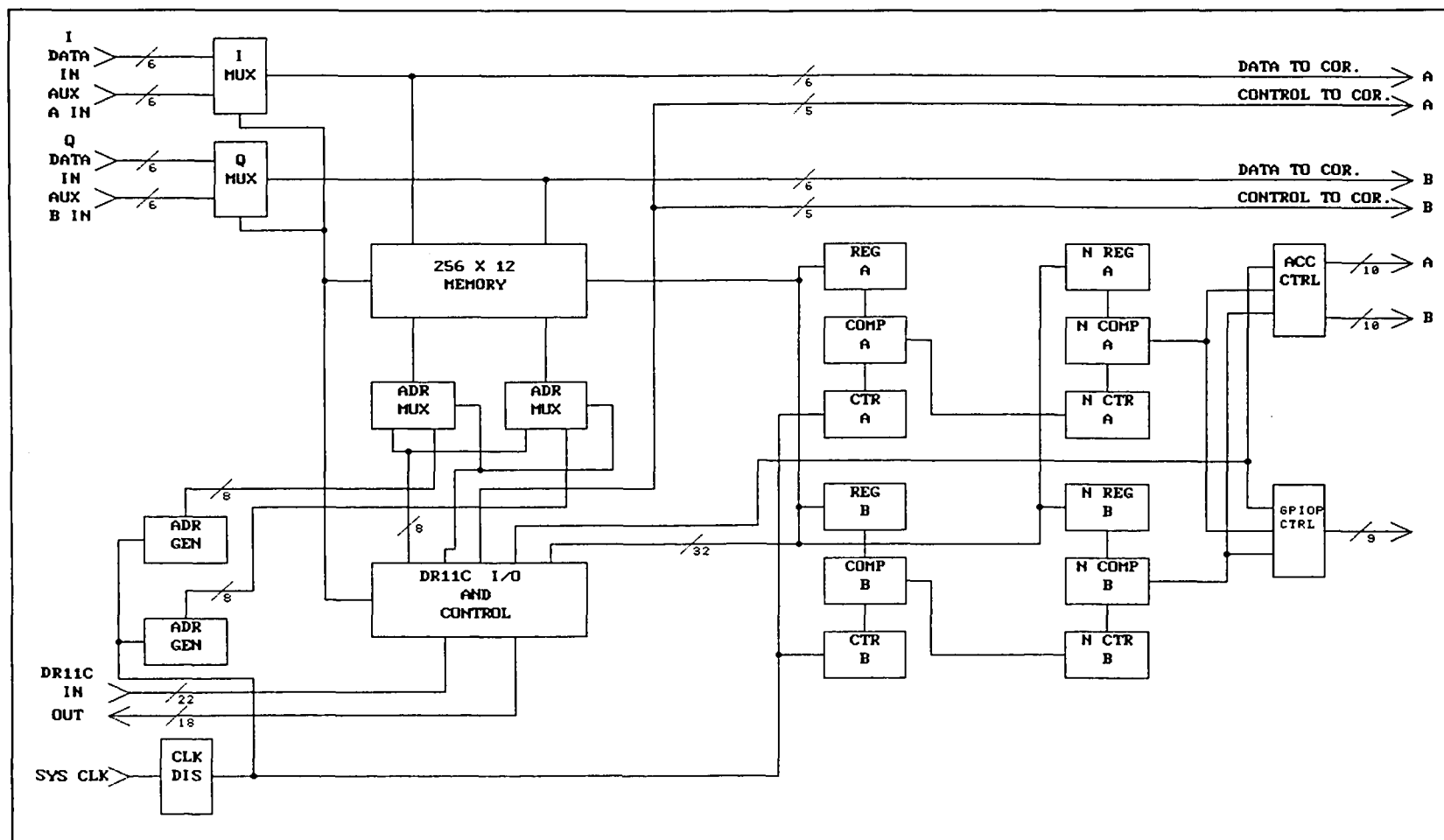
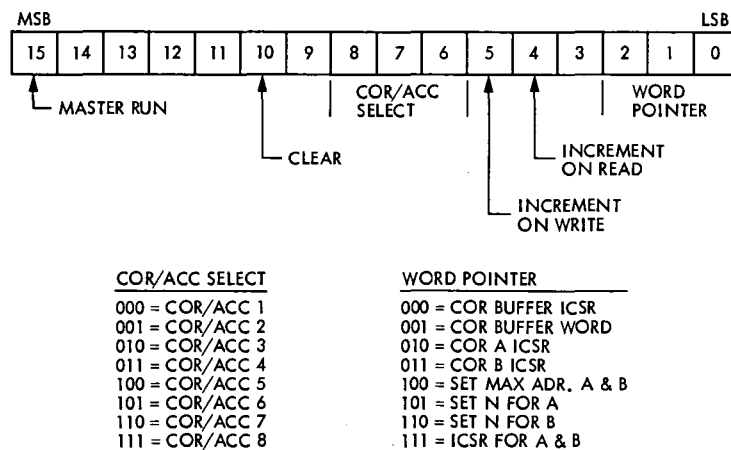


Fig. 3. Controller and Interface block diagram



NOTE:
 DR11C MODE CONTROL BITS

CSR1	CSR0	
0	0	= READ/WRITE ACCUM CSR
0	1	= READ/WRITE FUNCTION REGISTERS POINTED TO IN THE ACCUM. SELECT AND WORD POINTER.
1	0	= RESERVED FOR FUTURE USE
1	1	= RESERVED FOR FUTURE USE

Fig. 4. Correlator/Accumulator control and status register format

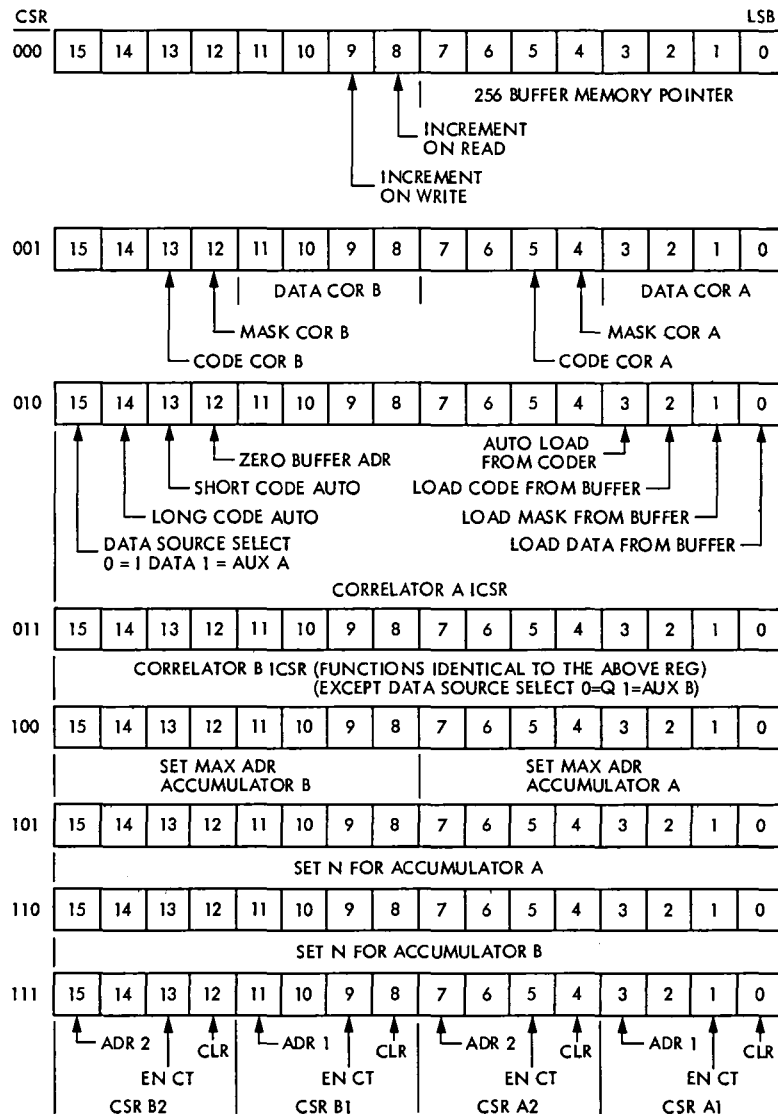


Fig. 5. Correlator/Accumulator function register formats

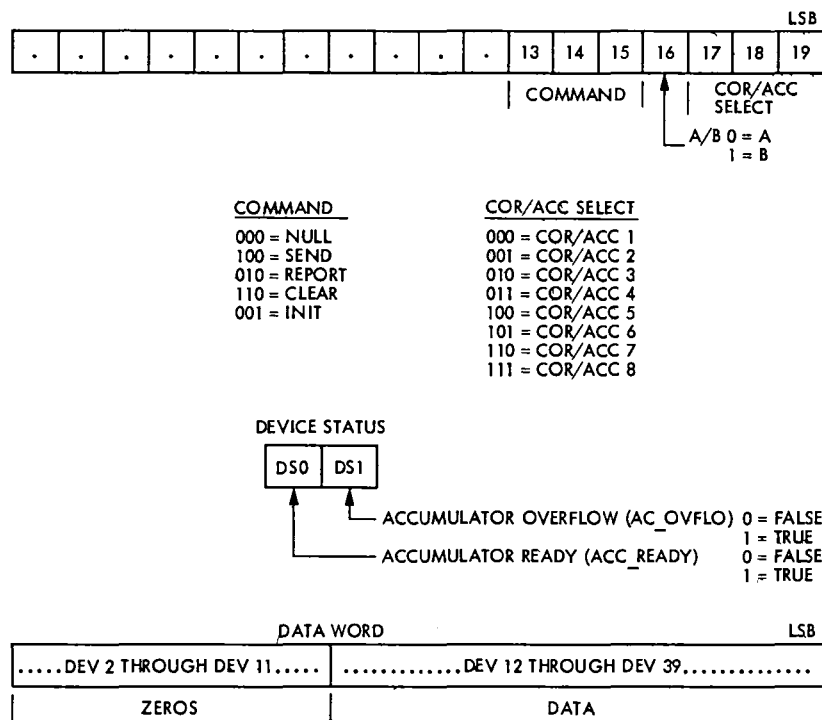


Fig. 6. GPIOP command and device word formats

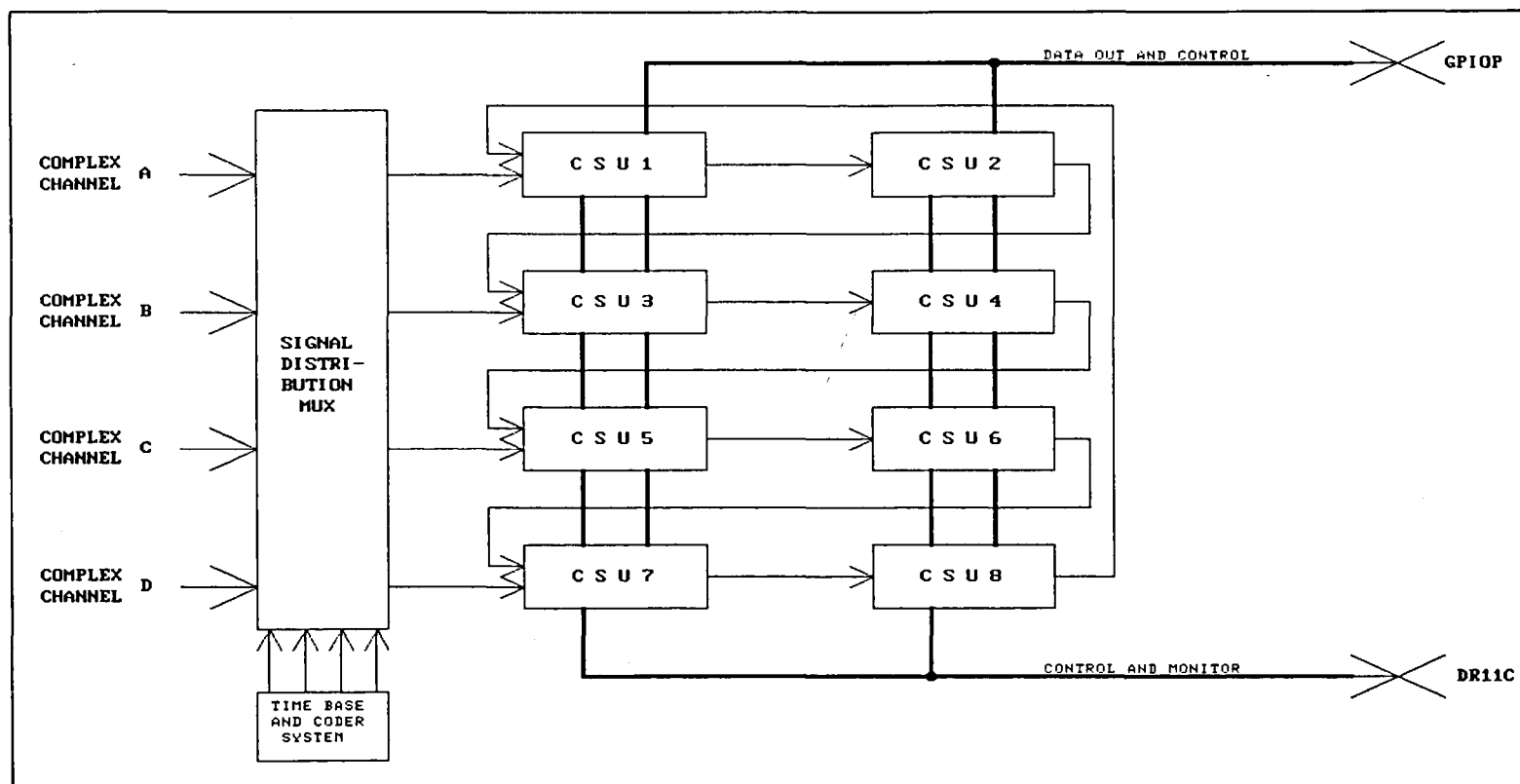


Fig. 7. Signal distribution diagram

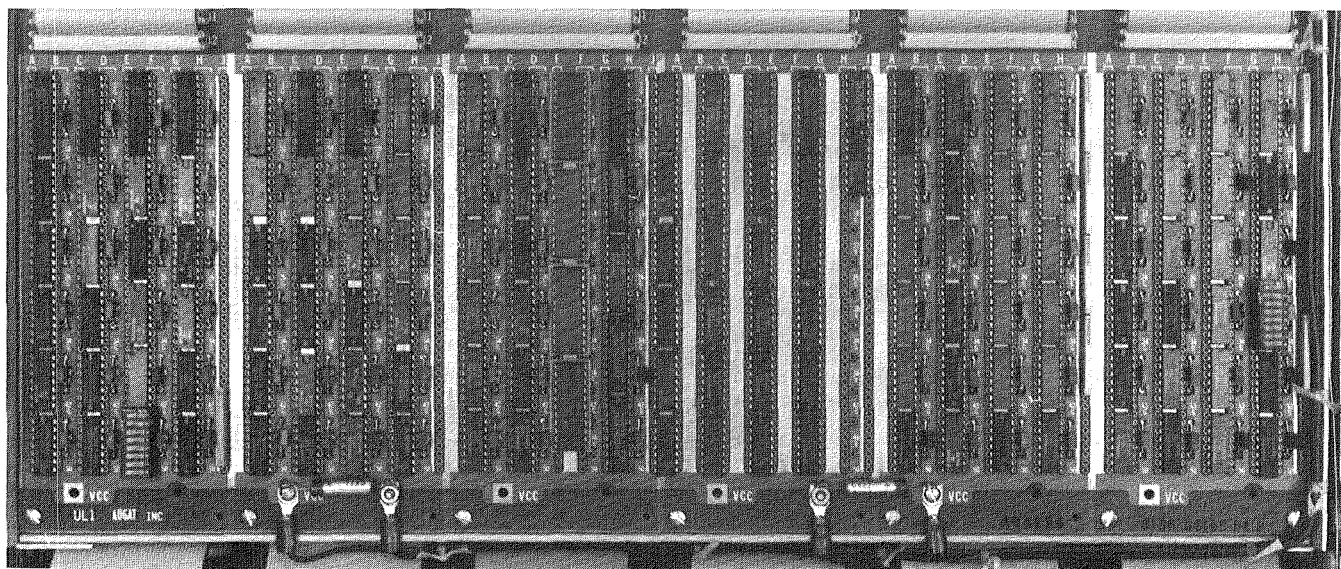


Fig. 8. Controller and Interface board

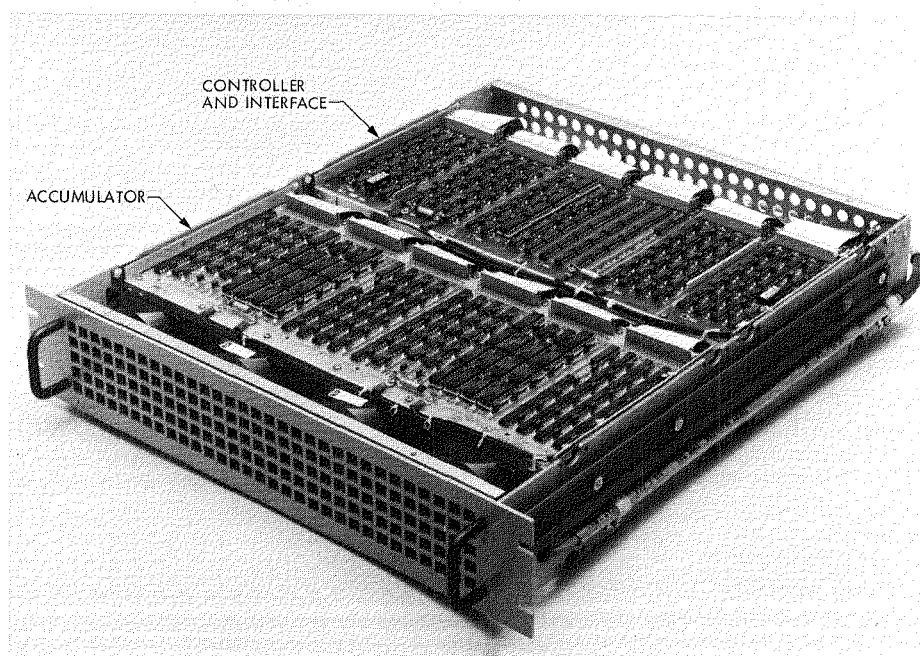


Fig. 9. Correlator/Accumulator System Module, top view

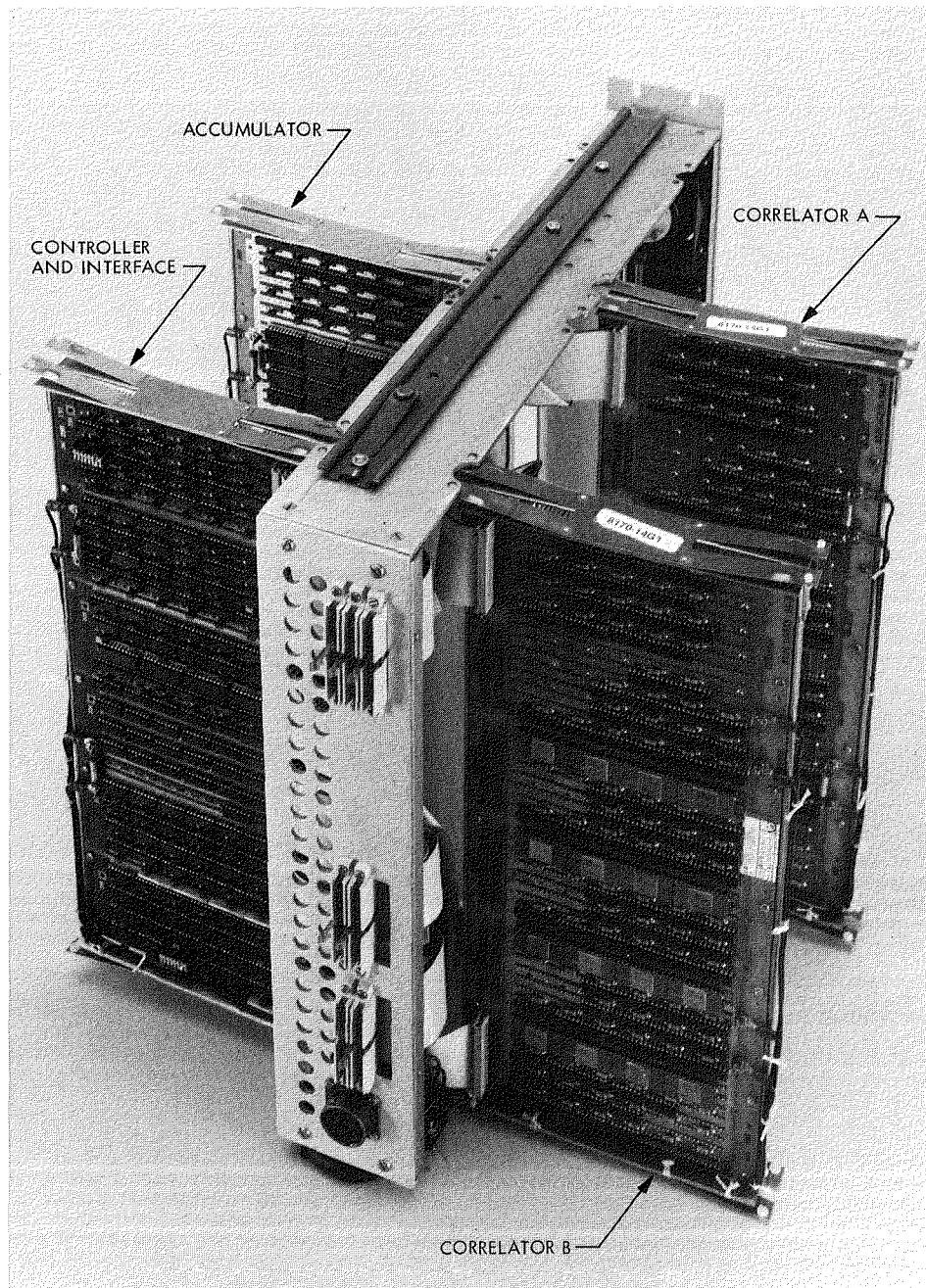


Fig. 10. Correlator/Accumulator System Module, open view

A General Monitor and Control Interface to the VAX UNIBUS by Way of the DR11-C I/O Port

S. S. Brokl

Communications Systems Research Section

The High Speed Data Acquisition System for the Goldstone Solar System Radar requires a number of interfaces for monitor and control functions for the special purpose data acquisition hardware. One common interface design is used. A large number of external control and monitor registers may be addressed and observed with this interface design, by using only three address locations on the UNIBUS and a standard DEC supplied input/output port. A broadcast control and status register is introduced which addresses any number of different units and registers.

I. Introduction

Digital Equipment Corp. supplies several types of interface ports (DR11-C, DR11-WC, and others); each one is designed for a limited number of address entries onto the UNIBUS. The UNIBUS is a single common path that connects the processor and all peripherals. The bus contains 56 lines for address, data and control information. In order to access a large number of peripheral device registers, a special interface is required for each different application. Presented here is a simple computer controlled general I/O structure designed for monitor and control functions using a DR11-C I/O port (Ref. 1) with an external Broadcast Control and Status Register (BCSR) and multiple function registers. It is designed to be used with the DR11-C but the idea lends itself well to any addressed bus with limited I/O capability. The DR11-C uses three address locations on the UNIBUS; one for communication with the I/O port internal registers, one for transferring data to an external register, and one for receiving data from an external register. By assigning different values to itself, the internal register may be used as a pointer to the external registers and allows broadcasting an external BCSR for further coding and communications to external function registers. A type of

indirect addressing is generated so that, with only three address locations used on the UNIBUS, any number of external registers may be addressed, read, or written.

II. Limitations with the DR11-C Interface

Figure 1 shows the register assignments for the DR11-C interface. The DRCSR register has only two control bits (CSR0, CSR1) for conditioning what the DROUTBUF and DRINBUF registers read from or write to an external device. A typical interface could therefore read or write only four external registers. This is far too few for many monitor and control functions for the High Speed Data Acquisition System and most other general applications. The DR11-WC has a direct-memory-access controller which allows a large number of external memory locations to be read or written but costs about six times more than the DR11-C. Also, the DR11-WC has more functions than are required for monitor and control applications. Since the monitor and control applications for the Goldstone Solar System Radar do not require high data throughput, the following solution using the DR11-C I/O port and external controller was adopted.

III. The General Monitor and Control Interface, a Solution to This Problem

Using the CSR0 and CSR1 bits in the DRCSR register as pointers that tag what area DROUTBUF and DRINBUF read and write enables a large number of external registers to be addressed. The following example will illustrate how this can be done using an external broadcast control and status register. A large number of external devices and registers can be communicated with through one port by using this technique. In simple terms this is a form of conditional indirect addressing.

As a simple example, assume the broadcast register format is as shown in Figs. 2 and 3. Both CSR0 and CSR1 are zero and a write operation is requested. This write is to all broadcast registers in all units. (The subsequent read operation will now only come from the unit selected that was written in the above write.) If no unit was selected, all 1's will appear on the data bus indicating that nothing was activated during read-back. Now, in the simple case where CSR0 = 1 and CSR1 = 0 we now read and write only to the registers pointed to by the broadcast register pointer and unit selected. Unit selection is achieved with a simple decoder shown in Fig. 4. In this example, only eight units are possible. This could be extended, but consideration of bus loading and bus length to other units reduces the maximum number to no more than ten. In this example, eight switches, pull up resistors, and an 8-to-1 multiplexer (74LS151) are all that are needed to select a unit. When a large number of identical units are being used, each one can now be made to look like any other without changing the software. Only the switch positions need be changed. With this logic, a unit can be unselected by having all switches on. This allows for unselecting bad units without removing the hardware. The "on" position for each switch generates ground to the inputs D0, D1, . . . , D7 on the 8-to-1 multiplexer. To decode a module and generate the MODEN TRUE signal, only one switch should be turned off per unit. Switch one signifies unit 0 and switch position eight signifies unit 7 when the data select bits A, B, C on the multiplexer decode D0 through D7. That is when MODEN is TRUE. Table 1 illustrates the unit decoder operation; XXX in the Address column indicates "don't care" conditions.

The Broadcast Control and Status Register (BCSR) (see Fig. 5) addresses the decoded unit using bits ICSR06, ICSR07, and ICSR08. Bits ICSR00, ICSR01, ICSR02, and ICSR03 comprise the function register pointer. Bits ICSR04 and

ICSR05 can be used to automatically increment the register pointer. The bit ICSR04 set to 1 will increment the function register pointer by one each time a function register is written to. With ICSR05 set to a 1 every time a function register is read, the pointer is incremented. Table 2 illustrates the decoding for reading and writing the function registers.

The broadcast register can be read only when MODEN is true. The tri-state buffer required on the BCSR is not shown. Bits ICSR09 through ICSR15 can be used individually for control and monitoring of needed internal tasks that must occur simultaneously on a number of units.

Figure 6 illustrates the buffers required for communication into and out of the DR11-C board. Additionally, the control decoding for reading and writing the registers and incrementing the function register pointer is illustrated. Figure 7 shows one type of function register that can be used. The clocks for reading and writing shown in Fig. 5 could control any type of register or control function. Each AM25S18 has both a register output plus a tri-state output to monitor its internal condition. This ensures that the busses used in the interface are functional. Only the first two 16 bit register files and the last register file are shown in Fig. 7. In Fig. 6, NDRDY* provides the clock when decoded in the 74138 for CKFR00, CKFR01, . . . , CKFR15. The function DTRAN* provides the read signal by way of the 74138 decoder for RDFR00, RDFR01, . . . , RDFR15. The function CKCSR loads data into the BCSR when CSR0 and CSR1 are both zero. The function FRDET is true when CSR0 is one and CSR1 is zero. The function CKPTR* increments the function register pointer every time a function register is read or written if the appropriate ICSR04* or ICSR05* bits are set. The signals DI00 through DI15 are data inputs to the function registers. DO00 through DO15 are function register data outputs.

IV. Conclusion

This simple interface is easily constructed and can be changed to suit a wide variety of interactive applications using the UNIBUS on a VAX computer. FORTRAN callable software drivers have also been developed at JPL to run under the VAX/VMS operating system. This type of interface was used by three different engineers while developing the Goldstone Solar System Radar and reduced three different types of hardware and software interfaces to only one.

Reference

1. "DR11-C General Device Interface User's Manual," Digital Equipment Corp., Maynard, Mass., 1978.

Table 1. MODEN decoding

Switch Positions								Address			MODEN
D0	D1	D2	D3	D4	D5	D6	D7	A	B	C	Y
0	0	0	0	0	0	0	0	X	X	X	False
1	0	0	0	0	0	0	0	0	0	0	True
0	1	0	0	0	0	0	0	1	0	0	True
0	0	1	0	0	0	0	0	0	1	0	True
0	0	0	1	0	0	0	0	1	1	0	True
0	0	0	0	1	0	0	0	0	0	1	True
0	0	0	0	0	1	0	0	1	0	1	True
0	0	0	0	0	0	1	0	0	1	1	True
0	0	0	0	0	0	0	1	1	1	1	True
0	0	0	0	0	0	0	1	0	0	0	False

- Notes: 1. All eight positions can be individually decoded true and will not decode on all zeros or another switch position not selected.
2. Dip Switch Position: 0 = switch on, 1 = switch off.

Table 2. Function register write and read decoding

ICSR03	ICSR02	ICSR01	ICSR00	NDRDY*	DTRAN*	FRDET	Output Function True
0	0	0	0	1	1	0	All Outputs False
0	0	0	0	0	1	1	CKFR00
0	0	0	0	1	0	1	RDFR00
0	0	0	1	0	1	1	CKFR01
0	0	0	1	1	0	1	RDFR01
0	0	1	0	0	1	1	CKFR02
0	0	1	0	1	0	1	RDFR02
0	0	1	1	0	1	1	CKFR03
0	0	1	1	1	0	1	RDFR03
0	1	0	0	0	1	1	CKFR04
0	1	0	0	1	0	1	RDFR04
0	1	0	1	0	1	1	CKFR05
0	1	0	1	1	0	1	RDFR05
0	1	1	0	0	1	1	CKFR06
0	1	1	0	1	0	1	RDFR06
0	1	1	1	0	1	1	CKFR07
0	1	1	1	1	0	1	RDFR07
1	0	0	0	0	1	1	CKFR08
1	0	0	0	1	0	1	RDFR08
1	0	0	1	0	1	1	CKFR09
1	0	0	1	1	0	1	RDFR09
1	0	1	0	0	1	1	CKFR10
1	0	1	0	1	0	1	RDFR10
1	0	1	1	0	1	1	CKFR11
1	0	1	1	1	0	1	RDFR11
1	1	0	0	0	1	1	CKFR12
1	1	0	0	1	0	1	RDFR12
1	1	0	1	0	1	1	CKFR13
1	1	0	1	1	0	1	RDFR13
1	1	1	0	0	1	1	CKFR14
1	1	1	0	1	0	1	RDFR14
1	1	1	1	0	1	1	CKFR15
1	1	1	1	1	0	1	RDFR15

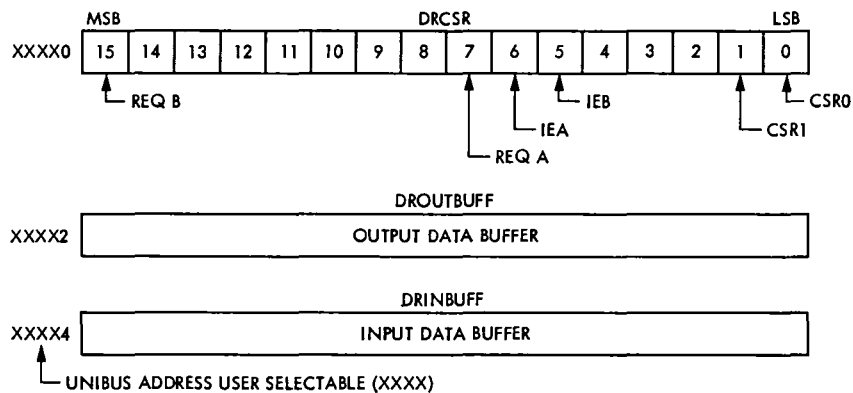
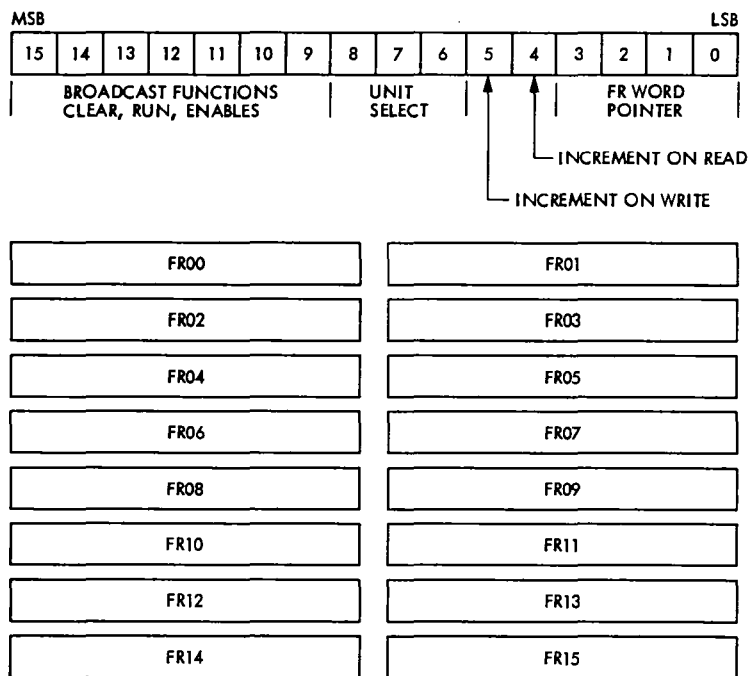


Fig. 1. DR11-C register assignments



NOTE: EACH FUNCTION REGISTER FR00 THROUGH FR15 CAN BE UP TO 16 BITS

Fig. 2. Broadcast control and status register with function register formats

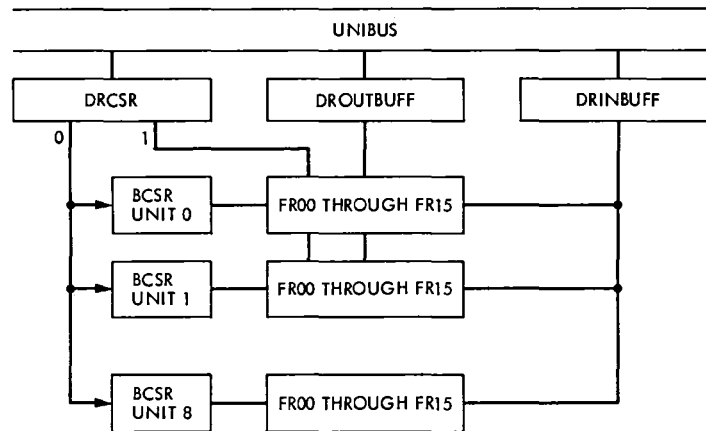


Fig. 3. Control and data flow diagram for register pointing

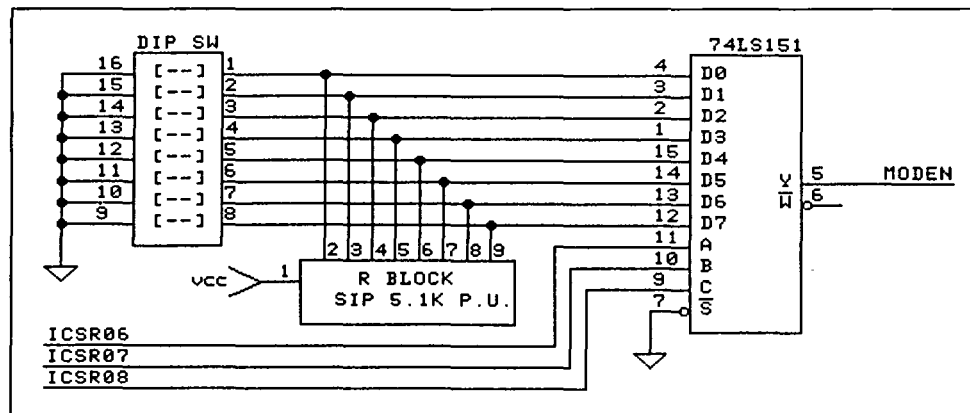


Fig. 4. Unit decoding

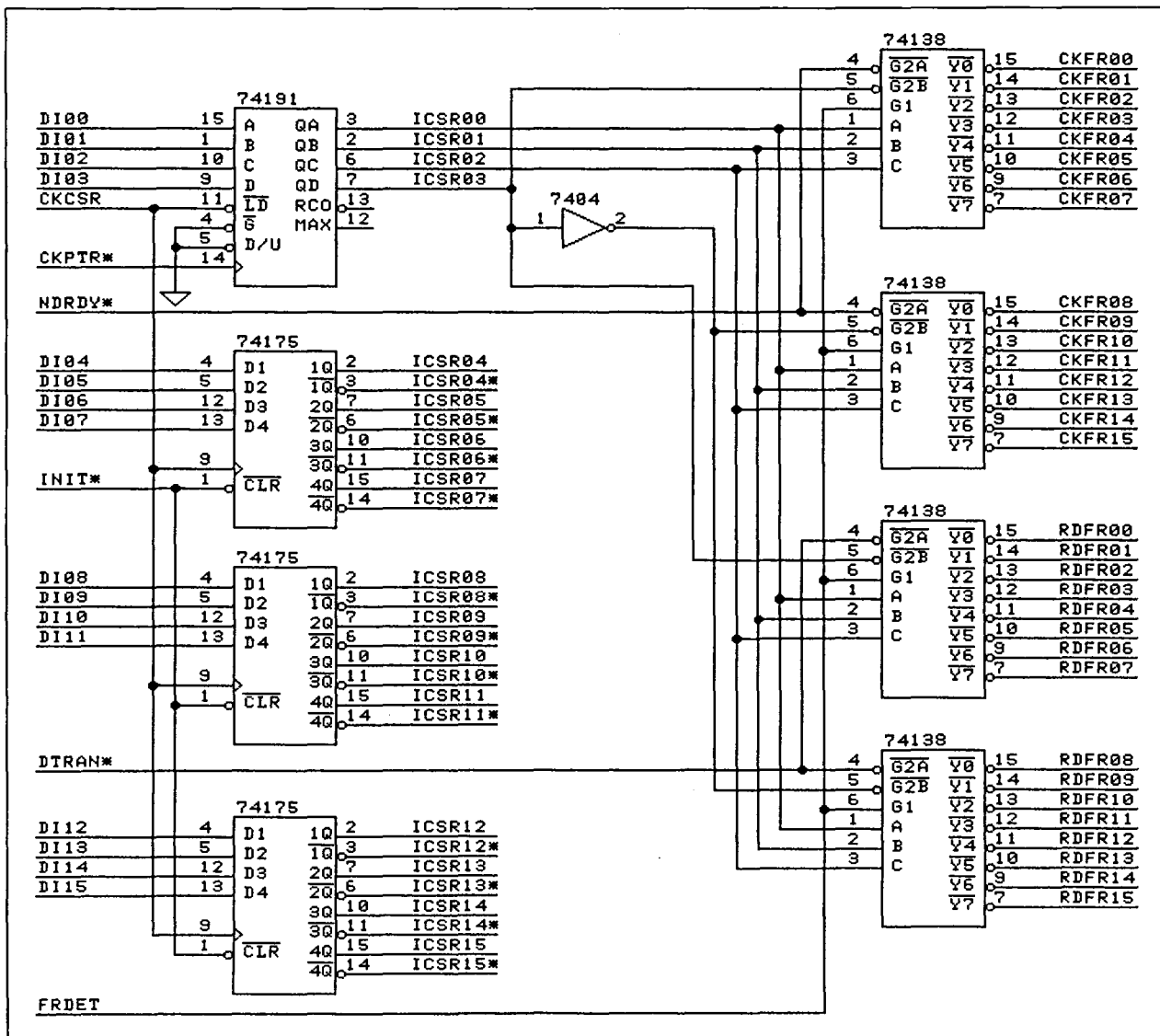


Fig. 5. Broadcast control and status register (BCSR)

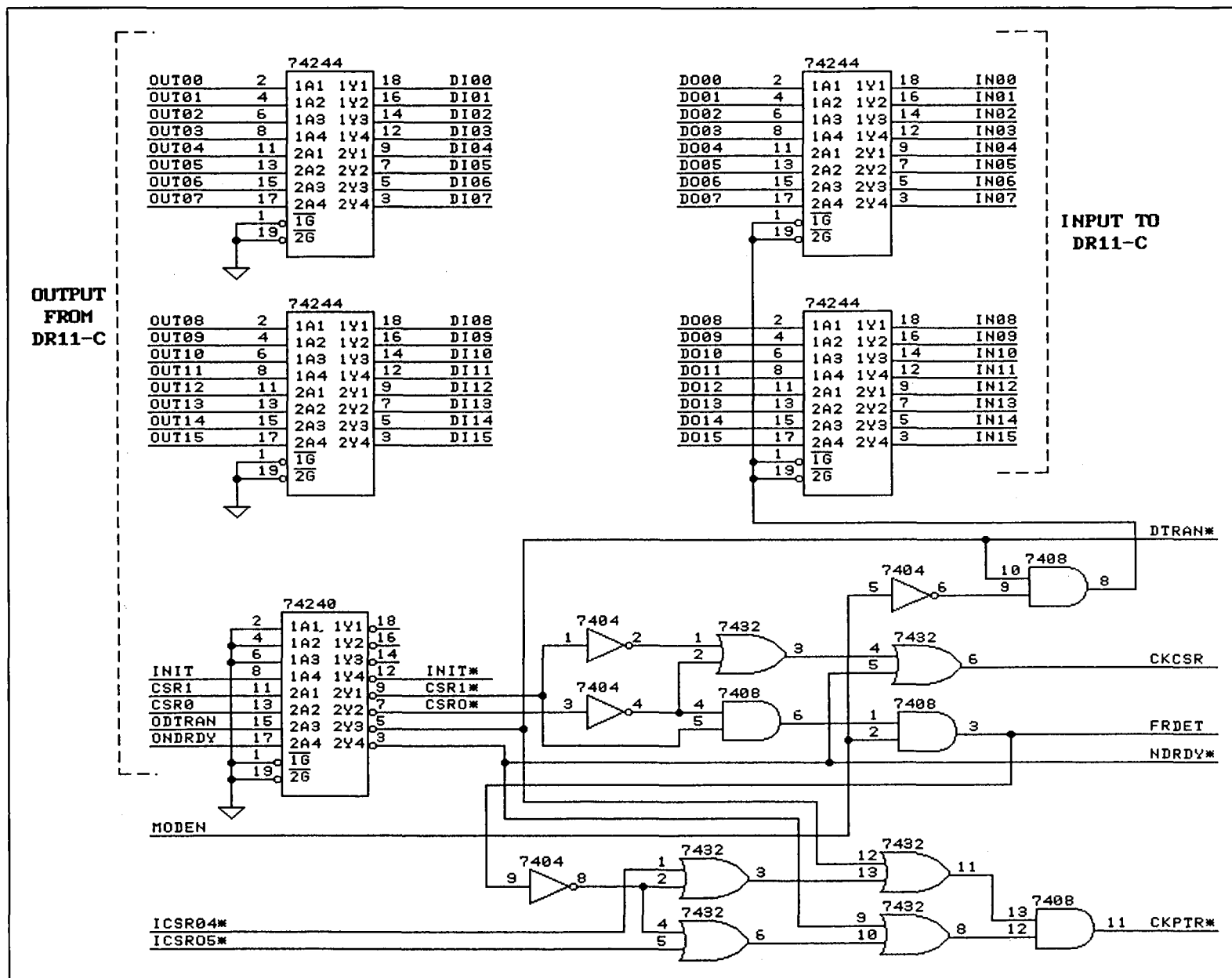


Fig. 6. DR11-C input/output buffers

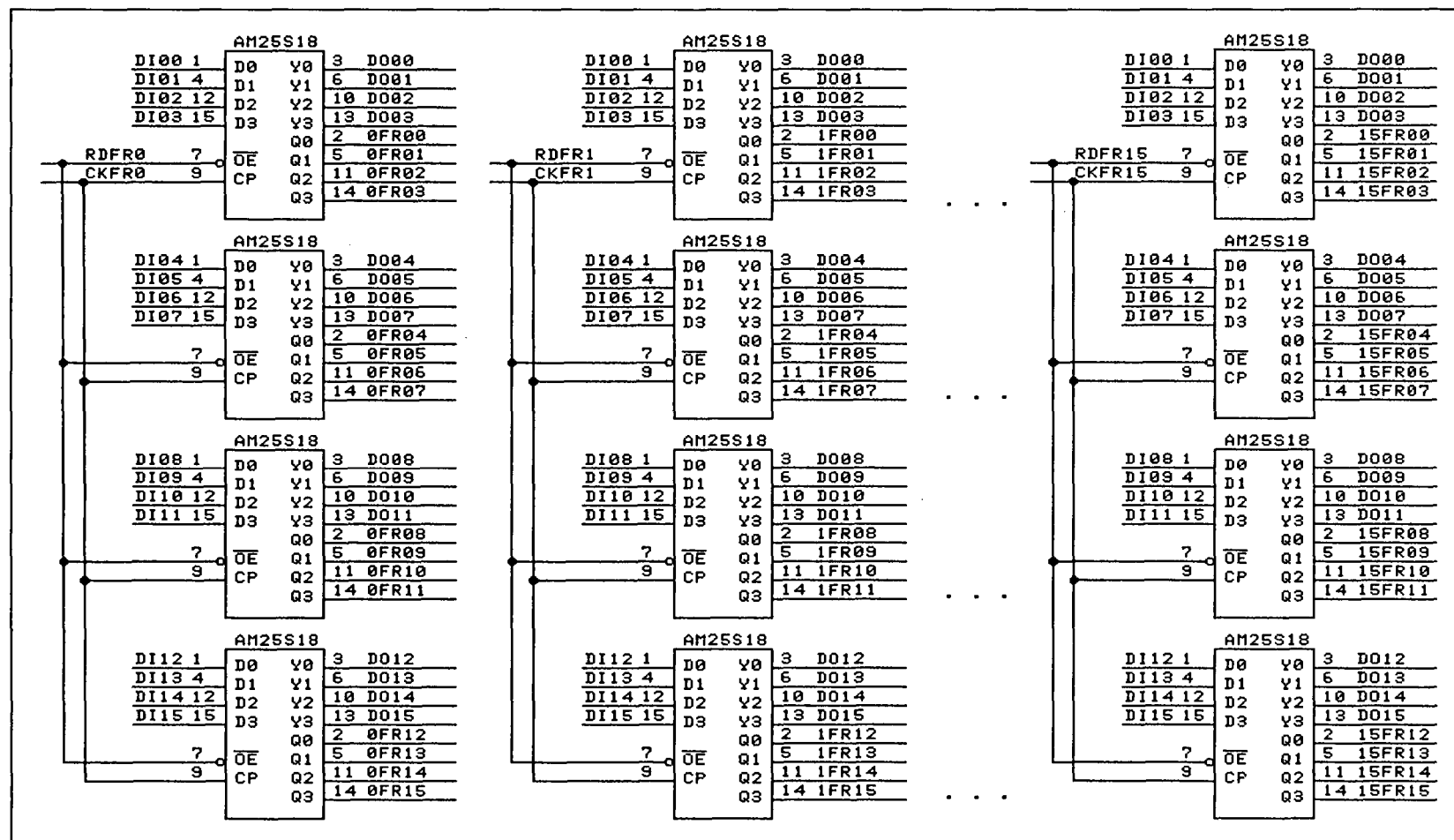


Fig. 7. Function registers

Mark IVA Project Training Evaluation

S. N. Stephenson

TDA Mission Support and DSN Operations Office

The article describes a participant evaluation of a DSN technical training program. The Mark IVA project is an implementation to upgrade the tracking and data acquisition systems of the DSN. Approximately six hundred DSN operations and engineering maintenance personnel were surveyed. The survey obtained a convenience sample including trained people within the population in order to learn what training had taken place and to what effect. The survey questionnaire used modifications of standard rating scales to evaluate over one hundred items in four training dimensions. The scope of the evaluation included Mark IVA vendor training, a systems familiarization training seminar, engineering training classes, and on-the-job training. Measures of central tendency were made from participant rating responses. Chi square tests of statistical significance were performed on the data. The evaluation results indicated that the effects of different Mark IVA training methods could be measured according to certain ratings of technical training effectiveness, and that the Mark IVA technical training has exhibited positive effects on the abilities of DSN personnel to operate and maintain new Mark IVA equipment systems.

I. Introduction

The DSN Mark IVA Project is a multiyear, multimillion dollar implementation effort to upgrade telecommunications and data acquisition systems of the DSN (Ref. 1). The actions taken to establish training requirements and budgets, organize training resources, plan and supervise training activities, and evaluate training results are a Mark IVA project responsibility. The scope of Mark IVA project training includes four types of training: 1) vendor training, 2) a systems familiarization training seminar, 3) engineering training classes, and 4) on-the-job training (OJT) experiences. Vendor training is associated with generic units of equipment which form basic components of Network systems, such as a minicomputer or a microprocessor. Seminar training is oriented primarily to description and discussion of new and modified Network subsystem groups of equipment, and the engineering classes are focused

upon individual major assemblies within these subsystem equipment groups. OJT training occurs for all Network personnel who use elements of Network systems to perform telecommunications and data acquisition functions.

The trainers are customer engineers of equipment manufacturers, JPL system designers of hardware and software capabilities, and the JPL development engineers and computer programmers who have built or modified the DSN data systems. The trainees are field engineers, technicians, and operators who work at the DSN Deep Space Communications Complexes (DSCCs) located in Goldstone, California; Canberra, Australia; and Madrid, Spain; and at the DSN Network Operations Control Center (NOCC) in Pasadena, California. The training instruction consists of theoretical concepts associated with each major equipment assembly, subsystem,

and system, plus a variety of techniques and procedures needed for installing, configuring, testing, operating, and maintaining Mark IVA hardware and software capabilities.

The research objective was to complete a participant evaluation of the effects of Mark IVA project training in rapidly changing DSN work environments through a training survey designed to identify the outcomes of various Mark IVA training experiences. Key terms are defined below.

Vendor training is any commercially available class or learning device employed to communicate structured product information or to teach well-defined skills associated with maintenance of generic equipment units.

Seminar training refers specifically to a five-week DSCC Implementation Seminar held at JPL in November 1983 to inform and familiarize DSN personnel with Mark IVA subsystems.

Cognizant development engineer (CDE) training refers to the theoretical and practical instruction provided on Mark IVA equipment assemblies by JPL design and development engineers during or after initial installation and testing periods at the DSCC.

On-the-job training (OJT) is the unregulated learning activities that take place through self study programs under actual field engineering, test, operation, or maintenance conditions at the Goldstone, Canberra, and Madrid DSCC; at the NOCC; and within other DSN facilities and groups.

System refers to the interacting parts of a functional whole consisting of Mark IVA equipment assemblies and computer programs grouped according to purpose for deep-space telecommunications and data acquisition.

Operations is configuring and activating DSN equipment systems to produce end-to-end data flow from the DSCC to the NOCC.

Maintenance is testing, isolating, and correcting faults in equipment systems in order to restore malfunctioning hardware or software to operational service.

Effect refers to a consequence or a result of some action or process attributable to Mark IVA training experience; e.g., the ability to load and initialize a new computer program.

Evaluation means assessing a Mark IVA training experience to determine whether training has been technically accurate, clearly presented, and applicable to participant needs.

The research questions were a) whether the effects of different Mark IVA training methods can be measured according to certain ratings of technical training effectiveness; and

b) whether the Mark IVA training, inclusive of vendor, seminar, CDE and OJT training experiences, has affected the abilities of DSN personnel to operate and maintain new Mark IVA equipment systems. The significance of the study is that it has been of value in assessing the impact of Mark IVA training in terms of decisions about additional expenditure and use of dedicated project resources and funds for training purposes. The training evaluation was wanted to characterize Mark IVA training results for JPL DSN managers who are presently working to resolve major issues of Mark IVA operability. A study of training effects was useful to uncover discrepancies between intended and actual training outcomes so that corrective action can be taken to modify training design or delivery problems.

As evaluation research, the chief limitation of the study is that the research is retrospective. The participant response data are based upon self report and subjective opinion measures. In order to employ quasi-experimental pretest or post-test measures, or even to make timely trainee performance appraisals, formal training evaluation methodology would have to have been funded and incorporated as part of the original Mark IVA training program design. The original training plan made the assumption that the value of Mark IVA training would be inherent in achieving levels of acceptable system performance by trained people at the DSCC and the NOCC. It was because system performance issues had arisen that it was appropriate to conduct a formal retrospective participant evaluation of Mark IVA training. Results of the Mark IVA training evaluation were presented at the biennial DSN operations and engineering conference of JPL managers held on 20-24 May 1985 in Palm Springs, California, where it was reported that the effects of different Mark IVA training methods had been measured, and that positive effects had been identified for the Mark IVA technical training on the abilities of DSN personnel to operate and maintain new Mark IVA equipment systems.

II. Method

The approach of the study was to evaluate the impact of Mark IVA training in the context of the individual participant's perceptions of the training program. A Mark IVA Network survey was designed to gather four kinds of data: basic biographical information, training evaluation ratings, training needs, and general comments about training. The Mark IVA training program consisted of four types of training presentations: vendor schools, seminar sessions, engineering tutorials, and OJT training. Data were collected from a sample of trained and untrained DSN operations and engineering maintenance personnel. The survey obtained a large quantity of individual participant training evaluations, which have been

descriptively analyzed, plus information about training experiences and respondent views, which has been qualitatively analyzed.

A. Network Survey Plan

There is a population of approximately six hundred operations and engineering maintenance personnel in the DSN and an estimated one-quarter of this population has received some Mark IVA project training in the form of vendor classes, seminar sessions, and engineering tutorials. An additional number of individuals have watched video tapes of the vendor, seminar, and engineering training, and have received on-the-job training (OJT) in the use of Mark IVA hardware and software. There was no simple way to identify the trained people or to obtain a random sample. Accordingly, the population of DSN operations and engineering maintenance personnel was surveyed to obtain a convenience sample. Individual self reports identified which people were trained, what training had taken place, and to what effect. The Network survey which was developed also provided an opportunity for poorly trained or untrained personnel to express training concerns and preferences, and for training deficiencies and omissions to be detected.

The Network survey was released throughout the Network in late February 1985. Questionnaire returns were requested by the first week in April 1985. The timing of the survey was important relative to Mark IVA project implementation progress. The DSN could be viewed as being two-thirds implemented in March 1985. Training transfer was well underway in most facilities but not concluded. The training process was sensitive to problem solving and corrective action while project training resources could still be made available to the DSN operations organization. The Network survey was accompanied by an internal delivery return envelope and was bulk mailed through the JPL and overseas organizational mail system to DSN facility and section managers, who were asked to assure that copies of the questionnaire were given to all technical persons employed in the target facilities and groups. A survey cover letter was signed by the DSN operations organization office manager requesting but not requiring the cooperation of DSN operations and engineering maintenance personnel in completing the survey. Questionnaire response was anonymous and entirely voluntary, which was made clear in the letter and on the survey.

The Network survey measured participant reaction to Mark IVA project training. The respondent was identified only by his or her facility, the nature of his or her job, and whether he or she had participated in any of the Mark IVA training activities. If not, the respondent was invited to respond to a series of items about interest in training. If the respondent

had received training, he or she was asked to evaluate each type of training: 1) vendor classes, 2) seminar sessions, 3) engineering tutorials, and 4) OJT training. Each respondent was asked to identify his or her training experiences and to rate these experiences on a scale of positive to negative effects according to the degree to which the training received was technically accurate, clearly presented, and applicable to the respondent's needs. Each dimension contained a different array of training items. Many of the training subjects were identical. The respondents who had participated in training activities were asked on a different scale to agree or disagree as to whether Mark IVA training had helped respondents to perform Mark IVA related work. Finally, each respondent was asked if more training was wanted, and was invited to make comments relative to training.

B. Mode of Inquiry

The Network survey was conducted using questions that employ a standard opinion response rating scale, and a five-category, single-column response format with all choice points defined was developed for the Mark IVA training evaluation questionnaire (Refs. 2, 3). The rationale guiding the scale format decision was related to the necessity for respondents to assign ratings from among more than one hundred items in four separate training dimensions. The questionnaire had to minimize ambiguity and reduce the effort required by respondents to complete the survey, even at the cost of some discriminating power, in order to maximize the number of survey questionnaires returned. Five categories were felt to be easier to choose from than six categories. Single-column responses were felt to be less demanding than double-column responses when numerous items were to be read, considered, and rated. Defining all choice points instead of requiring abstract thought about possible alternatives avoided time consuming conceptual dilemmas for respondents.

The Network survey used ordinal response measures in four dimensional groups to obtain scores in five response categories which are identical across dimensions. There were 20 to 30 items for each dimension. The same technical subjects occurred in more than one training dimension. The units of analysis were the rating responses. Every completed survey from a person who had been trained produced rating responses in one or more groups of training items. Questionnaire rating scores in each response category were accumulated to arrive at frequency counts and arithmetic means for individual items, and a composite mean of the scores for each group of training items. Selected score frequencies of like training items in each training dimension were compared, and comparisons provided a useful check on the reliability of the measurement procedure. Since most of the respondents who received training supplied ratings in more than one training dimension, no

group of training items obtained an independent N for use in correlating rating scores of one training dimension with those of another dimension, or with training effectiveness scores. Although association between training dimension rating response scores and opinions about training effectiveness in Mark IVA job performance could not be statistically established with correlation coefficients, a relationship was inferred using nonparametric techniques.

Data presentations included tables and figures showing measures of central tendency, including the calculation of item score frequencies, arithmetic means, frequency curve and percentage distributions of response scores. Results of chi square tests of statistical significance were computed. The chi square test of statistical significance was performed on Network survey data, given the first null hypothesis (H_0) that there is no difference in types of Mark IVA training received by DSN technical personnel, and the first alternative hypothesis (H_1) that there is a difference in types of Mark IVA training received by DSN technical personnel; and the second null hypothesis (H_0) that there is no effect of Mark IVA technical training on the ability of DSN personnel to perform Mark IVA related work, and the second alternative hypothesis (H_1) that there is an effect of Mark IVA technical training on the ability of DSN personnel to perform Mark IVA related work. Replies to an open-ended (comments) question were compiled for content analysis. Examples of respondent comment are presented and interpreted in connection with Mark IVA project training effectiveness or resource allocation issues.

The Network survey did not measure psychological state-change variables (Refs. 4, 5) suggested by the participant reaction to stimulus other than training, such as experiences of flawed Mark IVA equipment system performance. Variables associated primarily with OJT training may have interfered with training effectiveness ratings, which are vulnerable to mixed stimuli. Some people found it difficult to separate their OJT training experiences and their training opinions from their problematic experiences of Mark IVA equipment performance. Unrelated psychological state-change variables may have biased the training evaluation and influenced training effectiveness opinions, with a possible result that some OJT training responses may have reflected change which would be difficult to connect with training or to compare with ratings and opinions associated with other training dimensions. Another source of potential bias in the evaluation research is an unknown response-shift effect (Ref. 6) which would have to be measured using pretest and post-test research designs. Although response-shift bias may have occurred, it has not been possible under these retrospective evaluation conditions to identify a response-shift effect in the statistical analysis of Mark IVA training survey data.

III. Data Analysis

Six hundred twenty survey questionnaires and return mailing envelopes were sent out in lot shipments to managers of target facilities and groups during the last week in February 1985. The survey sample interval lasted six weeks until April 5, 1985 to allow time for responses from internationals, vacationers, and rotating shift workers. It was assumed that there are 620 DSN operations and engineering maintenance personnel, that each individual received a survey questionnaire and return mailing envelope, and that every person was given an equal chance to answer the survey questions. The 221 survey respondents provided a 35.6% return rate of the survey questionnaires distributed. Table 1 and Fig. 1 show training survey responses from DSN facilities and groups. The distribution of respondents by training received is indicated for each DSN facility and group in Table 2 and Fig. 2.

The actual number of DSN operations and engineering maintenance personnel in the population, including JPL employees, contractors, and internationals, changed several times during the long survey sampling interval. Also, the definition of a technical person was not precisely specified in the letter accompanying the survey. The technical state was interpreted somewhat differently from one facility or group to another, and in some instances, the appropriate individuals may not have received questionnaires. Other members of the population may have had no opportunity to participate. One of the managers decided to administer the survey only to technical persons who had attended training classes, possibly excluding from the survey individuals who might have seen training video tapes, or who might have had OJT training experience to rate, or those untrained technical personnel who might have been numbered among survey respondents. The sampling procedure irregularities mean that the 221 survey respondents could not represent true DSN population parameters. The survey obtained a sample of convenience which approximates certain population characteristics that may be verified by other means. Proportions of survey questionnaire returns from different facilities and groups can not be directly related to actual numbers of technical personnel employed in these DSN facilities and groups, but appear to be representative for training evaluation purposes.

The Deep Space Communications Complex (DSCC) percentages seen in Figs. 1 and 2 are larger than those from all other DSN organizations. Mark IVA implementation training was focused on the hardware and software capabilities which have been designed for tracking stations. Most new Mark IVA physical equipment system elements are installed, configured, operated, tested, and maintained by the DSCC personnel. The three DSCCs have been target facilities of first importance in Mark IVA implementation training and to the Mark IVA training evaluation although the technical training of other

DSN facilities and support groups is certainly of equivalent programmatic significance.

Mark IVA project training evaluation was directed at the concentration of Mark IVA training resource investments reflecting new hardware and software maintenance and operation functions and priorities. The survey did not thoroughly evaluate training which may have occurred in connection with Mark IVA induced changes in DSN personnel job duties at all points in DSN organizations. Key DSN facilities and groups were part of the Mark IVA training survey because selective Mark IVA project training access and training materials have been offered to and utilized by most if not all DSN organizations. Seminar and OJT training ratings, for example, were received from survey respondents all over the Network. Composite comments provided by the respondents from each facility and group are valuable Network training status indicators.

The Ground Communications Facility (GCF), System Cognizant Operations Engineers (SCOE) group, and Performance Analysis Group (PAG) data presented in Tables 1 and 2 indicate that these organizations may contain a larger proportion of untrained people than of trained people. Network Operations Control Center (NOCC) facility data indicate that almost equal proportions of trained and untrained individuals may be present in that facility. While these numbers can not be accurately used to make representations about population conditions, they may be interpreted to suggest limited classroom training access, low utilization of available Mark IVA training materials (e.g., seminar video tapes), or a range of OJT training experiences which may not have been reported adequately.

The comments of respondents from these organizations point toward a pattern of unstructured reactive Mark IVA training accomplished through documentation self study programs, which is discussed further in this section. By contrast, the data associated with the Complex Maintenance Facility (CMF) and Compatability Test Area (CTA-21) facilities and with the Cognizant Operations Engineers (COE) group indicate that larger proportions of trained personnel may exist within these organizations. These personnel have participated as students in vendor classes, seminar sessions, and engineering tutorials, and their training has included hands-on equipment OJT training experiences. (CMF survey statistics contain questionnaire returns from only two of the three complex maintenance facilities.)

The Network survey returns produced a convenience sample consisting of nearly twice as many trained people (143) as untrained people (78) among the survey respondents. The data suggest that people who received some Mark IVA training

are twice as likely to respond to a survey that evaluates Mark IVA training than people who have not received Mark IVA training. Another interpretation of survey sample proportions could be that fewer technical personnel may presently require Mark IVA training than have already been trained. This is an attractive way to look at the data from an organization manager's perspective, but does not account for almost one hundred comments from trained and untrained survey respondents that suggest some, or more, or better Mark IVA training.

Following the presentation and discussion of Mark IVA training evaluation results at the May 1985 DSN operations and engineering conference of JPL managers, \$115K in TDA implementation organization funds were allocated for supplemental microprocessor and minicomputer vendor schools to be attended by DSN engineering maintenance personnel. Assurances were given that JPL technical division engineering resources would also be committed to the DSN operations organization in support of additional JPL unique equipment technical training which was required and appropriately funded through the TDA implementation organization. Conference minutes record a formal action item assigned to the TDA implementation organization to develop and fund an ongoing training program that involves the support of the Deep Space Communications Complexes by the cognizant development engineers in technical divisions.

The central focus of the Network survey was to obtain participants' assessments of the Mark IVA project training which had been accomplished and to learn whether Mark IVA training requirements had been met or if additional Mark IVA maintenance and operations training in DSN facilities and groups was needed. Figure 3 is a graph of the types of work performed by survey respondents, some of whom are working in more than one kind of job (e.g., in both engineering and supervision categories, or in both operations and analysis categories). Note that "operations" and "maintenance" were dominant DSN job categories of the survey respondents. This information was collected to assure that training assessments obtained from the survey sample arose from authentic DSN maintenance and operations work experiences. The following paragraphs review dimensions of Mark IVA training which have been evaluated.

A. Vendor Training

DSN training on the Mark IVA generic equipment units was purchased almost entirely from the commercial certification training course offerings of the manufacturer and equipment suppliers. Many product-based companies, such as Control Data Corporation (CDC), Intel Corp., and Modular Computer Systems (Modcomp), build up professionally staffed training facilities to support user maintenance and operations programs for the company's main product lines. During the Mark IVA

project, it is estimated that \$300K in tuition fees were invested in commercial training (maintenance) courses on microprocessors and minicomputers, clocks, discs, peripherals, programming, receivers, synthesizers, tape units, and television display systems for DSN engineers and technicians.

Generic equipment units were procured in lot quantities for installing and sparing at a dozen DSN locations, so it was necessary to train individuals from several DSN facilities and groups. Student transportation and sustenance costs became a significant factor in DSN organization training budgets for major equipment procurements because international travel expenses are involved. DSN organizations preferred to send the students to vendor schools rather than bring commercial equipment instructors to DSN locations because it was cheaper. Vendors usually have superior laboratory facilities and test equipment for student hands-on maintenance training practice.

Table 3 shows the participant evaluations of vendor training classes by the respondents who either attended the vendor schools or watched commercial course video tapes. Of the twenty-four vendor classes, seven classes were rated "good" with mean values ranging from 4.40 to 4.00. The most successful vendor training classes, in descending order of student ratings, were high-density magnetic tape unit class, high-density disc class, Classic minicomputer central processing unit (CPU) and memory class, frequency and timing master clock class, IAPX 86, 88, 186 microprocessor class, synthesizer class, and microsystems equipment troubleshooting class. The survey evaluations of these vendor training classes confirm earlier participant feedback received in interviews with individual students returning from vendor schools.

The seven "good" vendor classes are followed by fourteen vendor classes that were rated in the 3.85 to 3.00 range, which indicates that these vendor training classes have been evaluated as technically accurate, clearly presented, and applicable to respondent needs at the "satisfactory" level. Refer to Table 3 for vendor school subjects, levels of participation, and positive mean values associated with these successful vendor training experiences. The Lark disc class and the microbus structures (video tape) class received "poor" ratings, and the minicomputer 13.5, 256 Mbyte discs class was evaluated as "terrible."

It is interesting to note that the negative participant feedback about the minicomputer discs class caused a top priority purchase of two disc classes from the disc manufacturer, only one of which appears to have been an unqualified training success. Both disc manufacturer classes were presented one after the other during a two-week time period at the same place to the same student group. The Lark disc class followed the high-density disc class, and the Lark disc class participant

evaluation may have suffered by comparison with high-density disc class participant evaluation. If the Lark disc class had been the single vendor disc training experience that occurred, it is possible that ratings of the Lark disc training would have been more favorable. Participant feedback about the Lark disc class indicated that JPL obtained some valuable information leading to a disc design change, which may have been worth the price of the Lark disc training regardless of participant ratings. The microbus structure video tapes were a recent purchase and more participant feedback about these training materials would be desirable. Students described the problems with the original minicomputer discs class to be inadequate instruction plus a lack of proper equipment and materials at the vendor's facility. Survey evaluations confirmed the original negative participant feedback.

Some characteristics that successful DSN vendor maintenance training class experiences appear to share are instructor expertise in the product line, well-prepared technical documentation, suitable laboratory test fixtures, and training environments with controlled access to practice equipment units. DSCC maintenance engineers have formed the majority of the DSN attendees at vendor schools. They were hand-picked for commercial course opportunities, and they tend to be observant of vendor shortcomings in meeting the conditions for successful training. There seemed to be less concern about vendor aptitude for polished training performances than about the extent to which hands-on equipment learning experiences are supported by the commercial course design. It may be as important for DSCC maintenance engineers to be able to learn effectively by themselves within the vendor environment as it is for the vendor to provide the appropriate level of competent instruction.

Survey respondents' comments include descriptions of the need for additional microprocessor maintenance vendor training, and it seems important to fund a 1985-1986 DSN microprocessor training program. Based on participant feedback, emphasis and priority have been placed on vendor microprocessor training for DSCC personnel assigned to antenna and RF subsystem maintenance, CMF personnel repairing lowest replaceable elements (LREs) with microelectronic components and modern test fixtures, CTA-21 personnel maintaining new microprocessor based RF and digital equipment groups, and COE personnel specializing in microprocessor equipment maintenance program support.

DSCC maintenance engineers have asked for Classic minicomputer maintenance training in asynchronous data communication controllers which are part of the Classic minicomputer assemblies procured by JPL and for which maintenance training has never been provided. Assembly language programming with the Classic operating system and maintenance

troubleshooting with diagnostics applications executive software also have been suggested as candidate classes for 1985-1986 vendor minicomputer training. Operating system and diagnostics applications capability were an integral part of the JPL DSN Classic minicomputer assembly procurements, but vendor training classes were not provided. DSCC, CMF, CTA-21, and COE equipment maintenance engineers are key candidates for Classic programming schools.

B. Seminar Training

The DSCC Implementation Seminar was characterized by the Mark IVA implementation manager as the centerpiece of Mark IVA project training. This program of subsystem familiarization training events was created to respond to Mark IVA operations and maintenance training needs of engineers, technicians, and operators at Goldstone, Canberra, and Madrid DSCC, and to the Mark IVA technical information needs of seminar participants from local JPL DSN operational facilities and support groups. Subsystem training presented by Mark IVA designers, development engineers and programmers from implementation organizations was video taped at JPL's von Karman auditorium television studio annex as a permanent training resource for the DSCC and other DSN facilities. Video tape sets were also supplied to the engineering groups for archival purposes and for new employee training.

The seminar had Mark III era precedents as a training vehicle. The original Mark IVA project training plan envisioned a 1983 training seminar that would provide videotaped high-level technical introductions to Mark IVA systems. Funding allocated on the basis of Mark III cost records was possibly an order of magnitude off in accuracy for estimating the cost of a Mark IVA seminar involving ten times as many hardware and software training issues. Engineering organizations absorbed large charges as unestimated seminar training expenditures, including the significant cost of engineering man-hours invested in seminar technical presentations which had not been a Mark IVA line item called training in 1983-1984 engineering budgets.

If the hidden cost of man-hours which were expended by the engineering groups plus the travel and sustenance expenses billed to DSCC budgets had actually been debited to the Mark IVA training account, half a million dollars might not have been enough to pay for the DSCC Implementation Seminar. The training account and engineering account overruns associated with the training seminar support were a seminar training evaluation issue to the extent that the miscalculations reflected a notion that training comes free, that using DSN internal resources for technical training represents just another one of the intangible costs associated with a major implementation. There are no free, or invisible, training

lunches. Engineering organization budgets have to acknowledge the requirements for technical training in order to forecast the internal resource obligations needed to support technical training on the same basis as any other DSN implementation requirement.

Training should be assigned a priority that gives the function enough visibility to compete for available resources. DSN organization managers ought to be able to assess a training priority in all of their decisions affecting hardware and software deliverables. Future Mark IVA training account allocations should be based on provable estimates. Past expenditures traceable in the organization's records may be taken as guidelines, but technical training cost estimates, like other expensive organization investments, ought to start with current figures and comparisons of training delivery methods, comparisons available in the literature (Refs. 7, 8).

Over one hundred engineering presenters, fifty DSCC engineers and another fifty to sixty local DSN attendees were committed to a seminar program of training events held at the von Karman auditorium television studio from October 31 through December 2, 1983. Implementation organizations chose individual presenters based on their technical design expertise or range of developmental responsibilities. Many designers and development engineers elected to offer material in team presentation formats which gave more people the opportunity to contribute to a seminar knowledge base, to gain organization visibility and recognition through television appearances, and to stimulate discussions between speakers and the audiences. DSCC managers chose the DSCC personnel who traveled to JPL for the five-week seminar. Mark IVA complex implementation coordinators led station teams of mature experienced engineers with staying power for five weeks of Mark IVA technical presentations and with the ability to ask important, interesting questions for video tape training purposes.

Table 4 contains the evaluations of nineteen seminar training events which include the technical presentations and the dialogues that occurred among the engineering presenters and the seminar participants. In examining the frequency and mean values associated with each subsystem training item, recall that seminar events which were recorded on unedited video tapes were personal human experiences, not glossy packaged training products. The seminar was a series of formal structured presentations interleaved with question and answer exchanges punctuated by video tape breaks, coffee breaks, equipment crashes, anecdotes and arguments from the floor. It was intended to be like that. Dialogue and discussion were training objectives, and the survey statistics should not be interpreted to mean that presentations did not achieve the intended technical training goal of the seminar.

Twelve seminar sessions were rated as technically accurate, clearly presented, and applicable to respondent needs at the "satisfactory" level. These sessions were considered successful seminar training experiences. The highest frequency and mean values in seminar events responses were assigned to the local area network (LAN) training session. The LAN training was given by an exceptional presenter who brought terminal equipment to the seminar and by chance or by design, swept the seminar participants into exciting exercises with LAN programs. Skilled camera work captured this unusual seminar training experience on the video tapes. In addition to high task competence, the LAN presenter also exhibited strong leader position behavior which may have influenced the survey respondent perceptions. That combination of trainer qualities may have enhanced LAN seminar training effectiveness (Ref. 9).

Seven seminar sessions have mean values ranging from 2.95 to 2.42, meaning that survey respondents did not give full marks for technical accuracy, clarity, or fit with respondent needs to these seminar training experiences. Baseband assembly (BBA) seminar training, for example, got off to a poor start when presenters appeared to be unwilling to respond to BBA questions, which could have had some effect on ratings for applicability of training to respondent needs. Although the BBA seminar training was an engineering development team event, most of the technical material was prepared and presented by a Madrid DSCC engineer working at JPL on a Mark IVA project implementation assignment. Survey respondent ratings for clarity may have been affected by differences in English language enunciation which are natural for this individual and for the two other engineers from the Madrid DSCC who presented the precision power monitor (PPM) material. The Mark IVA systems presentations were made by the DSN system engineers. In contrast with the presenters from implementation organizations, the system engineers did not invest as much time or effort in preparing for the seminar. Network support subsystem (NSS) presentations were given as brief instructive talks and a panel discussion by several teams of individuals. Limited DSCC audience participation during the NSS presentations may point to an effect on the ratings for applicability of training to respondent needs.

The less than satisfactory mean value assigned to the receiver subsystem seminar training was a surprise. Considerable effort was invested in the receiver subsystem seminar presentations, which were led by a person with teaching credentials, one of the rare presenters with a background and skills in adult education. Test support subsystem seminar presentations included speakers for the maintenance support assembly (MSA) and for the telemetry simulation assembly (TSA). By accident, the TSA presentation master video tapes were reloaded into the television studio video tape decks and overwritten during a

subsequent seminar event. When the tape problem was discovered, the TSA presentation was reconvened, but the second session fell short in technical information content and audience involvement through no fault of the presenters. The disservice done to TSA seminar training by the tape accident may have affected the survey respondent ratings and mean value assigned to test support subsystem.

One problem with the DSCC Implementation Seminar video tapes as a permanent DSN training resource is the technical changes which have occurred in Mark IVA subsystems since the November 1983 presentations given by the implementation organizations. Several survey respondents commented about large differences between seminar training information and prevailing Mark IVA equipment subsystem realities. While it is not likely or even desirable that another Mark IVA seminar will be organized for training purposes, the 1983 seminar training presentations can be selectively updated with 1985-1986 technical status information. Based on participant feedback, some candidates for updated or amended technical presentations on changed equipment are the antenna, command, monitor and control, network support, telemetry (including baseband assembly), and test support subsystems. The cognizant development engineers for antenna firmware and the monitor and control software have experimented with updated seminar format video tape training presentations, and have gotten good results with informal videotaping sessions held at the Goldstone DSCC.

C. Cognizant Development Engineer (CDE) Training

CDE training is a traditional JPL engineering activity associated with first installations of JPL unique hardware assemblies or major software packages. It is customary for a JPL designer or implementer to accompany the equipment into the field when it makes its first appearance in a DSN facility system environment. Development engineers have responsibility for overseeing the new equipment integration into the operational subsystems within which it will be expected to function. The training idea is that while the development engineers are in the field, they can show the facility engineers what to do with the equipment, assuming that it works when it arrives and that it has been properly supported with spares and documentation.

Formal CDE training for the Mark IVA equipment was negotiated with the engineering organizations in January 1984 following the DSCC Implementation Seminar. Mark IVA CDE training classes took place during the initial Mark IVA implementation phase at the Goldstone DSCC, where field engineers from all three DSCCs were gathered to assist with the first equipment installations. Selected CDE training classes were repeated at overseas DSCCs. Mark IVA capabilities at other DSN facilities were also supported with training

by CDEs, but not with formal classes like those which were planned and organized for the DSCC.

The initial training expectations of DSN facility personnel are often thwarted because JPL designers and implementers tend to view first installations as another milestone in the development cycle rather than as completed equipment deliveries. Protracted cycles of phased Mark IVA hardware and software deliveries meant that the CDE training was replete with technical information inconsistencies which added confusion to learning experiences for DSN maintenance and operations personnel. The CDE training process lacks instructional rigor. Classes were defined and scheduled according to the characteristics of what was being delivered, not by any rational application of training design standards to DSN instructional needs. Requirements for the CDE training were that CDEs would provide, to the best of their individual abilities, the technical instruction needed to install, operate, and maintain the delivered Mark IVA equipment capability, including theory of operations information.

The goal of the CDE training was to bridge the gap between the seminar training and on-the-job training experiences of the DSN technical personnel. When the Goldstone DSCC CDE training classes began, the decision was made to restrict CDE training access to facility engineers and technicians who were responsible for installing and maintaining the new equipment. Operations personnel were represented in the CDE training classes by DSCC operations individuals who were committed to the crew training for rotating shift operations. There were reasons for the decision to hold down operator participation, reasons associated with effective utilization of CDE training time during the Goldstone equipment installation phase lasting from March to November 1984.

CDEs were delivering unfinished hardware and incomplete software packages. Engineers and technicians had to learn everything that they could absorb from the CDEs quickly in order to help make the Mark IVA equipment systems operable. Operations was considered to be something that engineers would learn as they discovered how to make equipment work. Also, operations information would change each time the equipment capabilities changed. The CDE training classes were recorded on video tapes for later viewing by station personnel. Station trainers and engineers rather than CDEs appeared to be the logical choice for distilling accurate data and teaching station operators how to use the new equipment. CDE training classes were to address urgent operability issues for the engineers and technicians. The operator training was to be deferred until a later time in the Mark IVA integration and test phase when properly functioning hardware and software could be dedicated to operator training use.

The catch for the Goldstone DSCC and the Canberra DSCC was that these complexes had to be committed to spacecraft tracking operations before that time arrived. A similar outcome was in store for the Madrid DSCC. All of the DSCC operations crews have had to take the equipment as they find it. They have had to learn to operate the hard way through the OJT training experiences. Although CDE training class video tapes are available and have been used by the DSCC operators, these tapes require engineering interpretation and do not always represent the actual state of the Mark IVA hardware and software. DSCC engineers appear to have been too busy to give training to DSCC operators.

Pre-survey participant feedback indicated that operator crews were unhappy about exclusion as a group from the CDE training experiences. The operators' attitudes appeared to have been negatively influenced by lost opportunities for CDE training participation. Aversive approach behaviors toward the Mark IVA equipment (Ref. 10) may have been formed among DSCC operators at least partly as a consequence of CDE training access restrictions which meant that the operator learning process had to occur through OJT training augmented by seminar or CDE training class video tapes. More operator participation in CDE training classes probably should have been tolerated despite the danger of sidetracking CDEs from engineering operability issues and the risk of technical misinformation.

Table 5 contains the frequency and mean values assigned by survey respondents to the CDE training classes. Low frequency values for some CDE tutorials may mean that a number of trainees have not responded to the survey. Alternatively, class sizes were determined by Mark IVA equipment delivery conditions, restricted participation, and crowded work spaces. Of the twenty-five CDE training classes held, the mean values for fifteen classes range from 4.40 to 3.00. Most of the fifteen tutorials were assessed as technically accurate, clearly presented, and applicable to the respondents' needs. These classes were considered successful engineering training experiences at the "satisfactory" level. One CDE tutorial was rated as "good." The digital spectrum processor (DSP) software training class which achieved the highest CDE training mean score of 4.40 was eulogized in survey respondents' comments, where the CDE was praised as an "excellent" and "pragmatic" instructor.

According to survey respondents, ten CDE training classes did not fully measure up to the evaluation criteria for successful Mark IVA project training. One of the ten CDE tutorials was rated very close to a Mark IVA training failure attributed to the embryonic state of antenna-pointing software prematurely introduced into the field. In contrast to mean values of the DSCC Implementation Seminar training, survey respondent ratings were lower for LAN, maser, and telemetry CDE train-

ing experiences, while CDE training scores were higher for antenna drive assembly equipment, DSCC monitor and control (DMC), frequency and timing, microwave, receivers, transmitter, MSA and TSA test support equipment, and very long baseline interferometry (VLBI) receiver training experiences.

One interpretation of the increases in mean values was supported by participant feedback which suggested that seminar training had been an authentic learning experience for the CDEs. As cited in Carlsson *et al.* (Ref. 11, p. 2) individual experiential learning has been defined by the Kolb learning model in these terms: "Immediate concrete experience is the basis for observation and reflection. These observations are assimilated . . . then serve as guides in acting to create new experiences." Higher means for CDE training than were achieved for seminar training may be related to the efforts of the individual CDEs to adapt their Mark IVA technical instruction presentations to the perceived needs of real maintenance and operations people encountered at the seminar and in the field. The Mark IVA CDE learning experience might be considered as an encouraging step toward a better communications network (Ref. 12) between engineering and operations that would enhance Mark IV follow on development work and Mark V project designs and training.

D. On The Job Training

The OJT training process of learning how to do a job by actually performing the work is a time honored way to conduct DSN technical training. Survey responses from DSN people who have rated Mark IVA OJT training experiences describe perceptions of Mark IVA equipment environments and job training tools available within these work settings. Technical documents are the principal resource that supports learning in Mark IVA environments under OJT training conditions. Engineers, technicians, programmers, operators, analysts, supervisors, and support personnel obtain OJT training through extensive use of the written word in self study programs leading to job or equipment knowledge of a major implementation.

Technical documentation makes it possible to enter the hands-on equipment operation and maintenance work places with a set of meaningful actions to perform. If an individual does what the book says to do and produces results that are consistent with the job or equipment performance specified by the book, OJT learning processes have been reinforced and the books become significant elements of OJT training experiences. Successful Mark IVA OJT training means that the technical reference documents delivered with the Mark IVA equipment systems achieved standards comparable to other technical information presentations used for training DSN people. The information contained in all Mark IVA document deliverables

should be technically accurate, clearly presented, and applicable to respondents' needs.

Document deliverables that accompany new hardware and software are key source materials used to prepare abridged training presentations for different DSN groups and to develop equipment procedures for operations and maintenance activity in DSN environments. System procedures were rarely available in advance of hardware or software deliveries, even with good preliminary source materials, because systems-level procedure instructions have to be designed to mirror the actual behavior of interfaced equipment under program control. DSN technical personnel have had to rely on Mark IVA engineering source documents for firsthand information, particularly in real-time operations, about ways of performing Mark IVA related work.

Table 6 contains frequency and mean values provided by respondents with OJT training experiences. Means range from 3.33 to 1.66 and many of these values indicated trouble with Mark IVA OJT training. Participant feedback described the OJT training as a trial and error learning process of trying things that may not lead to repeatable results, suggesting that Mark IVA documentation insufficiencies may account for some of the poor OJT training results. The symbiotic relationship between Mark IVA technical documentation and OJT training may be new to some readers, and to illustrate the perceptions of association, typical comments are reproduced in these excerpts from Network survey data:

Documentation is only now starting to catch up to the changes that have been implemented in the hardware and software for Mark IVA. The first six months of OJT training were made difficult because hardware and software were changing on an almost daily basis. Operator OCIs could work one day and not the next, and there was little or no documentation of what was being changed. There are still multiple areas of software and hardware with no documentation, and what little is available is extremely confusing and difficult to interpret or understand.

Training and documentation are so poor that theory of operation is a word of mouth learning experience with everybody expressing their own theories.

I needed proper training. Proper means training with good procedures that work right. You can't tell whether the procedure is bad or the equipment is bad. Things should be standard. So far, nothing is standard. Mark IVA needs to be standardized with proven procedures.

Documentation is poor. There really isn't anything resembling training available.

The most beneficial training has been from reading SOMs and hands-on experience.

Of the thirty-three kinds of OJT training experiences rated, the OJT training area with the highest mean score of 3.33 was GCF software training. Eight OJT training areas with low frequencies and "poor" to "terrible" mean scores were NOCC software training experiences. Twenty-four OJT training areas were rated by DSCC, CMF, CTA-21, COE, and SCOE respondents. High frequency and mean values of 3.27 for link monitor console (LMC) software training, 3.13 for complex monitor console (CMC) software training, and 3.00 for frequency and timing (FTS) hardware training represented successful OJT training experiences that were assumed to have been supported in the DSCC and CTA-21 work environments with documented Mark IVA information which was technically accurate, clearly presented, and applicable to respondents' needs. Twenty of these OJT training areas have high frequency values with somewhat lower means ranging from 2.93 to 2.00, raising the question of how successfully the Mark IVA documentation associated with each of these OJT training experiences supported the learning process in DSCC and CTA-21 work settings. Five OJT training areas (four NOCC software package training experiences and the DSCC phase calibrator training experience) have means below 2.00 and were considered Mark IVA OJT training failures that were assumed to be partly attributable to Mark IVA technical document deficiencies.

It would be inappropriate to ignore the unstable problem-ridden Mark IVA system performance as a locus for variables other than technical documents which may affect OJT training. The difficulty is to isolate and quantify these variables, or their effects, for Mark IVA training evaluation purposes. Inconsistencies between Mark IVA equipment performance and most of the technical information training presentations appear to have influenced Mark IVA training evaluation, particularly in the OJT training dimension, where people were learning real-time operations by trial and error and the Mark IVA documentation became a critical element of Mark IVA OJT training experiences. Participant feedback invites speculation about possible gamma change effects for DSN trainees who appeared to have been experiencing "quantum shift in ways of conceptualizing salient dimensions of reality," versus alpha or beta change effects "occurring along relatively stable dimensions of reality" (Ref. 5, pp. 135-138). However, the Network training evaluation survey research design does not support distinctive interpretations of psychological state change effects which require controlled experimental research designs. Respondents' ratings and comments offer observational evidence, not proof (Ref. 13) that the presence of interfering variables may have been producing psychological change related to Mark IVA

work in DSN environments, at least for the participants who have supplied Mark IVA training survey data.

Low mean values associated with Mark IVA OJT training experiences should improve as Mark IVA systems performance stabilizes and operability problems are solved. Better Mark IVA technical document ought to be delivered to the DSN work force as expeditiously as possible. Engineering implementation organizations need to consider the OJT learning aspects of all Mark IVA engineering source documentation, especially the need for a close match between the written word and equipment reality. Mark IVA funds and higher development priorities should be assigned in order to deliver the necessary complement of Mark IVA technical documents for support of real-time operations OJT training on Mark IVA equipment systems in NOCC and DSCC environments.

E. Differences

Table 7 shows clear statistical differences between the four types of Mark IVA training which have just been discussed. One hundred forty-three survey respondents who had received training have supplied the composite rating frequencies and means displayed in Table 7 for vendor schools, seminar sessions, engineering tutorials, and OJT training experiences. Survey data indicate that there is one full rating point's difference between mean values assigned to the vendor schools (3.55) and to the OJT training experiences (2.54). Half of a rating point separates the vendor schools' score from mean values assigned to the seminar sessions (3.00) and the CDE tutorials (3.08). The same difference exists between seminar session and CDE tutorial scores and the mean value for OJT training experiences.

According to the stated evaluation criteria of technical accuracy, clarity of presentation, and applicability to respondent need, vendor schools were the most successful Mark IVA training venture. CDE tutorials and DSCC Implementation Seminar sessions have also met the survey training evaluation criteria for successful Mark IVA training. OJT training experiences, with four exceptions, were not very successful. Training method comparison data graphs for seminar session, CDE tutorial, and OJT training experience rating scores have been prepared on Mark IVA technical subjects which have been covered by all three types of training. Frequency curve comparisons of training survey mean values are provided to assist DSN implementation organizations who want to look at graphic representations of the survey data in connection with their particular areas of technical training responsibility. Refer to Figs. 4 through 21 for these graphs.

The training experience could perhaps represent a million dollar Mark IVA project investment if Mark IVA implementa-

tion organization training man-hours had been calculated and charged above the line. Even larger sums could probably be allotted to Mark IVA technical drawing and document costs. As suggested by the Network survey OJT training results, future DSN implementation project training and documentation efforts could be enhanced by priorities and project development resource allocations rigorously maintained from design to first installation stages of the implementation project for successful results in these key DSN support functions.

F. Effects

Survey respondents who had received training were asked to rate the effectiveness of Mark IVA training by expressing opinions as to whether they agreed or disagreed with the statement that Mark IVA training had helped them to perform Mark IVA related work in their jobs. One hundred forty-three people in the survey were qualified to determine whether their job performance abilities were affected by training and to state their opinions about Mark IVA training effectiveness. Table 8 is a breakdown of opinion responses by DSN facility and group. Figure 22 displays the same data as a percentage distribution of rating totals across the five opinion categories of strongly agree, agree, no opinion, disagree, and strongly disagree. A glance at Fig. 2 serves as a reminder that the DSCC respondents are the largest group of trained people in the survey. In Fig. 22, the DSCC led other DSN organizations in Mark IVA training effectiveness ratings.

The percentage distribution in Fig. 22 reveals that 78% of the survey respondents who had received training perceived Mark IVA training to have affected their abilities to perform Mark IVA related work while 22% of survey respondents who had received training had no opinion. Equivalent percentages (7% and 6%) of respondents strongly agreed and strongly disagreed about the effects of Mark IVA training. A larger percentage (55%) consisting of people who strongly agreed and agreed felt that the training had been helpful. A smaller percentage (23%) consisting of people who strongly disagreed and disagreed felt that the training had not been helpful. The fact that thirty one people expressed no opinions is interpreted to mean that these individuals were ambivalent about Mark IVA training effectiveness. They may not be able to perceive that training helps, yet they may not be willing to assert that training does not help.

The effects of Mark IVA training as perceived by survey respondents are more obvious in Fig. 23, where opinion response ratings were combined by DSN facility and group into positive and negative effect classifications, separating these data from the statistics of survey respondents who had no opinion. Positive effect responses consist of all of the

ratings in the "strongly agree" and "agree" opinion categories. Negative effect responses consist of all of the ratings in the "strongly disagree" and "disagree" opinion categories. Positive effect rating percentages, particularly for the DSCC, are greater than negative effect rating percentages, except for the percentages reported by NOCC, CTA-21, and PAG respondents. Survey findings suggest that NOCC and PAG respondents had not experienced adequate or satisfactory Mark IVA training.

By contrast, the Mark IVA training which has been made available to the CTA-21 facility was equal in most respects to the DSCC Mark IVA training because CTA-21 functionally resembles a tracking station. Survey findings do not suggest a reason why a larger percentage of the ratings reported by trained CTA-21 respondents indicates negative training effects. Figure 23 percentages derived from one hundred twelve decisive opinion response ratings (excluding the "no opinion" response ratings) indicate that the Mark IVA training has had positive effects for a majority (70%) of the trained survey respondents who have expressed decisive opinions. A minority (30%) of the trained survey respondents who have expressed decisive opinions appear to have felt that the Mark IVA training had not been helpful in its effect on their abilities to perform Mark IVA related work.

G. Significance

The nonparametric chi square test of statistical significance for categorical variables and enumerative data was selected to determine whether measures of association for variables reported in this study can be considered significant. A nonparametric choice acknowledges that conventional probability assumptions have not been satisfied by the sampling design of this research, although the magnitude of χ^2 values obtained by applying chi square distributions to the survey data clearly points toward the operation of nonchance factors at an appropriately conservative level of significance. Table 9 presents the results of a two variable chi square test of the relationship between types of Mark IVA training and training survey rating responses, and Table 10 presents the results of a single variable goodness of fit test of observed and theoretical survey opinion response distribution (Refs. 14, 15). In formal statistical terms, the null hypotheses (H_0) have been established that there is no difference in types of Mark IVA training received by DSN technical personnel, and there is no effect of Mark IVA technical training on the ability of DSN personnel to perform Mark IVA related work (see Section II: Method). Alternative hypotheses (H_1) are that there is a difference in types of Mark IVA training received by DSN technical personnel, and there is an effect of Mark IVA technical training on the ability of DSN personnel to perform Mark IVA related work. The significance level for both tests is $\alpha = 0.01$.

The value of χ^2 is calculated for the two variable case from the formula

$$\chi^2 = \sum_{r=1}^r \sum_{c=1}^c \frac{(f_o - f_e)^2}{f_e}$$

where

f_o = the observed value

f_e = the expected value

$$\sum_{r=1}^r \sum_{c=1}^c = \text{sum of the ratio over both rows and columns}$$

The value of χ^2 is calculated for the one variable case from the formula

$$\chi^2 = \sum_{i=1}^k \frac{(f_o - f_e)^2}{f_e}$$

where

f_o = the observed value

f_e = the expected value

$$\sum_{i=1}^k = \text{sum of the ratio over } k \text{ categories}$$

Sampling distributions are the chi square distributions with degrees of freedom (df) equal to $(r - 1)(c - 1)$ for the two variable test. For the one variable test, $df = k - 1$. The two variable test has $df = 12$, and at $\alpha = 0.01$, the critical region consists of all values of $\chi^2 \geq 26.217$. The one variable test has $df = 4$, and at $\alpha = 0.01$; the critical region consists of all values of $\chi^2 \geq 13.277$. Table 9 shows the obtained χ^2 value of 209.817 for the two variable test case to be greater than the critical value of 26.217 which is required for significance at the 0.01 level. This result means that H_0 may be rejected and H_1 may be asserted: there is a statistically significant difference in types of Mark IVA training received by DSN technical personnel. Table 10 shows the obtained χ^2 value of 80.460 for the one variable test case to be greater than the critical value of 13.277 which is required for significance at the 0.01 level. This result means that H_0 may be rejected and H_1 may be asserted: there is a statistically significant effect of Mark IVA technical training on the ability of DSN personnel to perform Mark IVA related work.

H. Training Needs

The discussion of survey information up to this point has been concerned with data from survey respondents who received training. The needs of survey respondents who did not receive Mark IVA training or who felt that they had been poorly trained are important to the Mark IVA training evaluation and were not as precisely measured as the dimensions and effects of accomplished training. Survey respondents who received no training make up 35% of the survey sample. They have contributed to the evaluation database and provided many of the survey comments which indicate where Mark IVA training deficiencies and omissions may be found. NOCC and PAG survey respondents, for example, have already been identified in other parts of the evaluation as the sources of information which suggested that these organizations may need attention and more resource assistance to accomplish Mark IVA training.

Quantified Mark IVA training needs are displayed in Figs. 24 and 25, which note the types of training requested by untrained and poorly trained survey respondents and the subjects for training that interested these respondents. The figures show selection frequencies that occurred in training method and subject categories. Those respondents who were untrained and poorly trained and who made the choices usually selected more than one training method or subject. The selection frequencies for the OJT training method and for the subjects of JPL unique equipment and software are larger than the numbers for any other selection categories. The respondents evidently felt that the needs for Mark IVA OJT training and for classes in JPL unique hardware and software outweighed other needs, which is not surprising for respondents who may have been working at their jobs without training in the Mark IVA environments. Commercial equipment and software training classes also have high selection frequencies in the subject categories. Frequencies associated with vendor school, seminar session, and engineering tutorial types of training are on a par with one another in the method categories.

Although Mark IVA technical documents have been characterized earlier in this section as the principal resource for Mark IVA OJT training, the survey comments of the untrained and poorly trained respondents highlighted experiences of being shown what to do and told how to do it as a meaningful (if not always comfortable) form of OJT training. The challenge for managers interested in cultivating this method of training is to identify and assign a sufficient number of trained people who know how to instruct others to serve as resources for individuals who need productive or timely show-and-tell OJT training. It is in every DSN organization's interest to encourage trained people to share Mark IVA knowledge by helping coworkers with no Mark IVA background to develop

the capability required for getting the job done. Managers could stimulate some informal Mark IVA training activity by rewarding people who are qualified and willing to teach others. The survey comments suggested that there may be a number of DSN people who would probably do better work if they received some simple show-and-tell training support from knowledgeable members of the organization. The alternative to rewarding competent in-house instructors is to force people to learn the hard way by trial and error, which lowers organization productivity and morale.

Survey comments cited hands-on equipment experience as significant Mark IVA OJT training which would be available only in facility environments under maintenance or operations supervision and special procedural controls. Section and facility managers should be aware of individuals who need or want hands-on Mark IVA experience and would be in the best position to take the appropriate action for providing this form of OJT training. In "other" categories noted on Figs. 24 and 25, the untrained and poorly trained respondents wrote twelve training requests: DSCC-NOCC-MCCC data flow, computer-aided instruction on NSS software, Mark IVA system capabilities, limitations, workarounds, reliable reference materials, antenna control, DSN system flow and configuration, qualified instructors, more classroom work, and better documentation (three items repeated prior statements). The item suggesting NSS software computer-aided instruction may be a useful Mark IV follow on software design input. The remaining items would be local classroom or OJT training options where appropriate resources are available.

There seem to be pressing local needs for sustained Mark IVA training within DSN organizations, at least as represented by the comments of the respondents who were untrained and poorly trained. Facility and section managers are responsible for establishing structured in-house training programs that report regularly and utilize formal record keeping practices, such as training hours checkoff sheets, lesson plan outlines (LPOs), and posted lists of technical documents, library books, video tapes, commercial course opportunities, resident expert lecture schedules, and the like. Several DSN facilities already have individuals who are accountable for training. Such attention to the training function may help to assure that DSN facility training occurs according to local training plans and schedules of training events.

IV. Conclusions and Recommendations

The research shows that training method differences can be measured through ordinal response ratings of Mark IVA training experiences. Mark IVA technical training is effective for DSN personnel who operate and maintain new Mark IVA equipment systems. Survey data indicate that on balance

Mark IVA training helps people perform Mark IVA related operations and maintenance jobs in Mark IVA work environments. Vendor classes have received favorable ratings. Participant feedback for this training dimension indicates good return on commercial course investments. Scores for seminar training sessions did not produce frequent values at positive or negative extremes. The engineering tutorial ratings were mixed and on-the-job training experiences produced values in the satisfactory-to-poor evaluation range because this dimension is where DSN respondents have had difficulty keeping their Mark IVA training experiences separated from their experiences of flawed Mark IVA equipment performance.

Rating mean values associated with successful training experiences are consistent with, but are not necessarily related to, the positive opinion response scores indicating agreement that Mark IVA training helps to perform Mark IVA work. Rating mean values associated with unsuccessful training experiences are consistent with, but are not necessarily related to, the negative opinion response scores indicating disagreement that Mark IVA training helps to perform Mark IVA work. Statistical significance of these data has been tested with chi square distributions. Obtained values permit rejection of the null hypotheses, asserting the alternative hypotheses that there is a difference in types of Mark IVA training, and there is an effect of Mark IVA technical training on the ability of DSN personnel to perform Mark IVA related work.

Recommendations resulting from the Mark IVA project training evaluation are summarized as follows: supplementary Mark IVA training resource allocations including more commercial courses in minicomputer and microprocessor vendor schools; seminar session technical presentation video tape updates; engineering review of unsuccessful CDE tutorial presentations to help determine whether these tutorials should be repeated in the field; higher project priorities and funds for technical training and document delivery improvements; and operations management review of DSN organization local training plans, resources, and schedules in DSN facilities and support groups.

Some future issues in the area of training design and evaluation were suggested by review of related literature as being worthy of DSN research. These issues include simulation training applications (Ref. 16); designing new matches between training materials and technical documents (Refs. 17, 18); isolation of psychological state change variables which affect training (Refs. 4, 5); and use of learning models (Refs. 10, 11) to characterize future training designs and to build better intraproject communication networks (Ref. 12) in anticipation of Mark IV follow on development work, and Mark V engineering implementation projects.

Acknowledgment

Portions of the material presented in this article have been previously submitted to and accepted by the University of San Francisco as graduate research. Assistance provided during the training evaluation study by Duncan L. Dieterly, Ph.D., psychologist and University of San Francisco senior faculty member, is gratefully acknowledged. Management support provided during Mark IVA project training by Jet Propulsion Laboratory Deep Space Network organization managers Raymond J. Amorose, Louis E. Butcher, Thomas Reid, Robertson Stevens, Jose Urech, Robert J. Wallace, and Paul T. Westmoreland is appreciated. Special thanks are extended to Jet Propulsion Laboratory technical division managers, supervisors, and Mark IVA cognizant development engineers, without whom there would have been little Mark IVA training to evaluate.

References

1. Gatz, E. C., "Networks consolidation program system design," *Telecommunications Data Acquisition Progress Report 42-63*, March-April 1981, Jet Propulsion Laboratory, Pasadena, CA, pp. 150-153.
2. Johnson, W. L., and Dixon, P. N., "Response alternatives in Likert scaling," *Education and Psychological Measurement*, Vol. 44, No. 3, Autumn 1984, pp. 563-567.
3. Dixon, P. N., Bobo, M., and Stevick, R. A., "Response differences and preferences for all-category-defined and end-defined Likert formats," *Educational and Psychological Measurement*, Vol. 44, No. 1, Spring 1984, pp. 61-66.
4. Putnam, A., "Designing from the logic of people," *Training and Development Journal*, Vol. 35, No. 5, May 1981, pp. 125-131.
5. Golembiewski, R. T., Billingsley, K., and Yeager, S., "Measuring change and persistence in human affairs: types of OD designs," *Journal of Applied Behavioral Science*, Vol. 12, No. 2, 1976, pp. 132-157.
6. Bray, J. H., Maxwell, S. E., Howard, G. S., "Methods of analysis with response shift bias," *Educational and Psychological Measurement*, Vol. 44, No. 4, 1984, pp. 56-61.
7. Holden, S. J., "The science of developing technical training," *Training and Development Journal*, Vol. 38, No. 9, September 1984, pp. 35-41.
8. Zemke, R., "Development and delivery of training: media, methods, and means," *Training*, Vol. 20, No. 10, October 1983, pp. 47-53.
9. Justis, R. T., Kedia, B. L., and Stephens, D. B., "The effect of position power and perceived task competence on trainer effectiveness: a partial utilization of Fiedler's contingency model of leadership," *Personnel Psychology*, Vol. 31, No. 1, Spring 1978, pp. 83-93.
10. Mager, R. F., *Developing Attitude Toward Learning* (Belmont, CA: Fearon Publishers, 1968).

11. Carlsson, B., Keane, P., and Martin, J. B., "R&D organizations as learning systems," *Sloan Management Review*, Vol. 17, No. 3, Spring 1976, pp. 1-16.
12. Tushman, M. L., "Managing communication networks in R&D laboratories," *Sloan Management Review*, Vol. 20, No. 2, Winter 1979, pp. 37-49.
13. Kirkpatrick, D. L., "Evaluating training: evidence vs proof," *Training and Development Journal*, Vol. 31, No. 11, November 1977, pp. 9-12.
14. Runyon, Richard P., and Haber, Audrey, *Fundamentals of Behavioral Statistics* (Reading, Massachusetts: Addison-Wesley Publishing Co., 1984), pp. 362-370, p. 418.
15. Babbie, Earl, *Practice of Social Research* (Belmont, California: Wadsworth Publishing Co., 1983), pp. 424-426, 498-499.
16. Simon, C. W., and Roscoe, S. N., "Application of a multifactor approach to training research," *Human Factors*, Vol. 26, No. 5, October 1984, pp. 591-612.
17. Klema, C., Casey, H. B., and Caple, D., "Technical and training manuals can be one and the same," *Training and Development Journal*, Vol. 38, No. 1, January 1984, pp. 74-75.
18. Jons, J. A. R., "Standard procedures: a quick path to technical training," *Training/HRD*, Vol. 19, No. 8, August 1982, pp. 52-55.

Table 1. Survey sample

Facility/Group	Number	Percent
DSCC	102	46.2
CMF	16	7.2
NOCC	28	12.7
GCF	16	7.2
CTA-21	14	6.3
COE	15	6.8
SCOE	19	8.6
PAG	11	5.0
Total	221	100.0

Table 2. Respondents of training survey

Facility/Group	Trained		Untrained	
	Number	Percent	Number	Percent
DSCC	78	35.4	24	10.8
CMF	12	5.4	4	1.8
NOCC	15	6.8	13	5.9
GCF	5	2.2	11	5.0
CTA-21	9	4.0	5	2.3
COE	13	5.9	2	0.9
SCOE	7	3.2	12	5.4
PAG	4	1.8	7	3.2
Total	143	64.7	78	35.3

Table 3. Vendor training: frequency and mean values of respondents' ratings^a

Vendor Classes/Video Tapes	Frequency (<i>f</i>)	Mean (<i>X</i>)
High-Density Disc	11	4.36
Lark Disc	11	2.81
Surveillance TV	8	3.75
IAPX 86, 88, 186 Microprocessors	13	4.00
IRMS 88, 80 Microprocessor Operating System	4	3.50
IRMX 86 Microprocessor Operating System	7	3.71
Introduction to Microprocessors	7	3.57
MCS 80/85 Microprocessors	11	3.45
Microbus Structures	4	2.75
PL/M Programming	9	3.66
RMX Programming	1	3.00
Synthesizers	8	4.00
Microsystems Equipment/Troubleshooting	7	4.00
Classic Minicomputer CPU and Memory	14	4.21
Classic Minicomputer Extended Arithmetic Unit	13	3.76
Classic Minicomputer Disc Controller	9	3.44
Classic Minicomputer Bus I/O Processor	14	3.85
Minicomputer Diagnostics Applications Executive	4	3.25
Minicomputer Introduction to Microprocessors	10	3.10
Minicomputer 13.5, 256 Mb Discs	8	1.50
Receivers	4	3.50
Console Displays	13	3.53
High-Density Magnetic Tape Units	5	4.40
FTS Master Clock	7	4.14
^a Excellent = 5.00		
Good = 4.00		
Satisfactory = 3.00		
Poor = 2.00		
Terrible = 1.00		

Table 4. Seminar training: frequency and mean values of respondents' ratings^a

Seminar Sessions/Video Tapes	Frequency (f)	Mean (X)
Antenna Subsystem	44	3.04
Baseband Assembly	25	2.92
Block IIA, V Masers	14	3.28
Command Subsystem	25	3.00
DSCC Monitor and Control Subsystem	49	3.10
Digital Spectrum Processing Subsystem	19	3.05
Frequency and Timing Subsystem	30	3.13
Ground Communications Facility Subsystem	29	3.03
Local Area Network	56	3.35
Mark IVA Systems	35	2.82
Microwave Subsystem	15	3.13
Network Support Subsystem	23	2.82
Precision Power Monitor	28	2.89
Receiver Subsystem	36	2.80
Telemetry Subsystem	28	3.00
Test Support Subsystem	21	2.42
Tracking Subsystem	22	2.95
Transmitter Subsystem	22	3.27
VLBI Receiver	25	3.00
^a Excellent = 5.00		
Good = 4.00		
Satisfactory = 3.00		
Poor = 2.00		
Terrible = 1.00		

Table 5. CDE training: frequency and mean values of respondents' ratings^a

CDE Tutorials/Video Tapes	Frequency (f)	Mean (X)
Antenna Drive Assembly	13	3.23
Antenna Pointing Software	15	2.06
Area Routing Assembly Software	9	2.77
BLK III RCVR/EXC/Controllers	23	3.13
BLK IV RCVR/EXC/Controllers	24	3.04
Command Modulator Assembly/Switch	18	2.83
CPA Software	8	2.87
Common Computational Modules	5	3.00
Data Channel Filter	4	2.50
DMC Software	13	3.53
DSP Software	5	4.40
Frequency and Timing	16	3.43
GCF Software	4	3.50
Local Area Network	29	3.13
Maintenance Support Assembly	20	2.85
Masers/S-Band FET/LNA	7	2.57
Microwave/Controller	14	3.71
MDA Software	6	2.66
NSS Software	3	3.00
Phase Calibrator	8	2.75
Portable Development System	8	3.12
Telemetry Simulation Assembly	15	3.26
TPA Software	7	2.71
Transmitter/Controller	13	3.61
VLBI Receiver/Controller	13	3.38
^a Excellent = 5.00		
Good = 4.00		
Satisfactory = 3.00		
Poor = 2.00		
Terrible = 1.00		

Table 6. OJT training: frequency and mean values of respondents' ratings^a

Hardware and Software OJT	Frequency (f)	Mean (X)
Antenna Control Subassembly (ACS)	29	2.89
Antenna Drive Assembly (ADA)	29	2.93
Antenna Pointing Assembly (APA)	25	2.72
Area Routing Assembly (ARA)	26	2.73
Baseband Assembly (BBA)	11	2.00
Block III and Block IV Receivers	31	2.90
Command Modulator Assembly (CMA)	26	2.76
Complex Monitor Console (CMC)	29	3.13
CPA Software	21	2.57
DDP Software	9	2.11
Digital Spectrum Processing (DSP)	9	2.77
Frequency and Timing (FTS)	25	3.00
GCF Software	9	3.33
Local Area Network (LAN)	30	2.86
Link Monitor Console (LMC)	33	3.27
Maintenance Support Assembly (MSA)	16	2.18
Masers	15	2.66
Microwave	21	2.76
MDA Software	21	2.71
NCD Software	6	1.66
NMC Software	7	2.14
NRS Software	7	1.85
NSS Software	10	2.10
NTK Software	9	2.11
NTM Software	7	1.85
Phase Calibrator	9	1.66
Precision Power Monitor (PPM)	22	2.72
Portable Development System (PDS)	11	2.27
Telemetry Simulation Assembly (TSA)	25	2.84
TPA Software	23	2.91
Transmitters	21	2.80
VAP Software	7	1.85
VLBI Receiver	12	2.83
^a Excellent = 5.00		
Good = 4.00		
Satisfactory = 3.00		
Poor = 2.00		
Terrible = 1.00		

Table 7. Composite frequency and mean values of respondents' ratings^a

Training Method	Frequency (<i>f</i>)	Mean (<i>X</i>)
Vendor Schools	202	3.55
Seminar Sessions	546	3.00
CDE Tutorials	300	3.08
OJT Training	591	2.54
^a Excellent = 5.00 Good = 4.00 Satisfactory = 3.00 Poor = 2.00 Terrible = 1.00		

Table 8. Survey opinions of trained people responding to the statement "Mark IVA training has helped me to perform Mark IVA related work in my job"

Facility/ Group	Strongly Agree	Agree	No Opinion	Disagree	Strongly Disagree
DSCC	8	40	16	12	2
CMF		7	3	2	
NOCC		5	2	4	4
GCF		3	1	1	
CTA-21	1	2	2	2	2
COE	1	7	4	1	
SCOE		4	1	2	
PAG			2	1	1
Total	10	68	31	25	9

Table 9. Chi square two-variable test of significance: relationship between types of training and rating responses

Types of Training	Rating Responses ^a					Total
	Excellent	Good	Satisfactory	Poor	Terrible	
Vendor	10 (11.092)	75 (48.189)	62 (83.807)	20 (46.833)	5 (12.078)	202
Seminar	11 (29.981)	134 (130.253)	268 (226.528)	120 (126.589)	13 (32.646)	546
CDE	18 (16.473)	92 (71.568)	110 (124.466)	60 (69.554)	20 (17.937)	300
OJT	21 (32.452)	90 (140.989)	240 (245.198)	180 (137.022)	60 (35.337)	591
Totals	90	391	680	380	98	1,639

^aValues in parentheses denote expected frequencies; other values denote observed frequencies.

$$\chi^2 = \sum_{r=1}^r \sum_{c=1}^c \frac{(f_o - f_e)^2}{f_e}$$

$$\chi^2 = 209.817$$

where

f_o = the observed value

f_e = the expected value

$$\sum_{r=1}^r \sum_{c=1}^c = \text{sum of the ratio over both rows and columns}$$

degrees of freedom (df) = $(r - 1)(c - 1)$

Test Result: At a significance level of $\alpha = 0.01$, with $df = 12$, the obtained χ^2 value of 209.817 exceeds the critical value of 26.217 and the null hypothesis (H_0) may be rejected. There is a statistically significant difference in types of Mark IVA training received by DSN technical personnel.

Table 10. Chi square one-variable test of significance: goodness of fit between observed and theoretical opinion response distribution. Statement of opinion: "Mark IVA training has helped me to perform Mark IVA related work in my job."

Trained Respondents	Opinion Responses ^a					Total
	Strongly Agree	Agree	No Opinion	Disagree	Strongly Disagree	
	10 (28.600)	68 (28.600)	31 (28.600)	25 (28.600)	9 (28.600)	143

^aValues in parentheses denote expected frequencies; other values denote observed frequencies.

$$\chi^2 = 80.460$$

$$\chi^2 = \sum_{i=1}^k \frac{(f_o - f_e)^2}{f_e}$$

where

f_o = the observed value

f_e = the expected value

$\sum_{i=1}^k$ = sum of the ratio over k categories

degrees of freedom (df) = ($k - 1$)

Test Result: At a significance level of $\alpha = 0.01$, with $df = 4$, the obtained χ^2 value of 80.460 exceeds the critical value of 13.277 and the null hypothesis (H_0) may be rejected. There is a statistically significant effect of Mark IVA technical training on DSN personnel ability to perform Mark IVA related work.

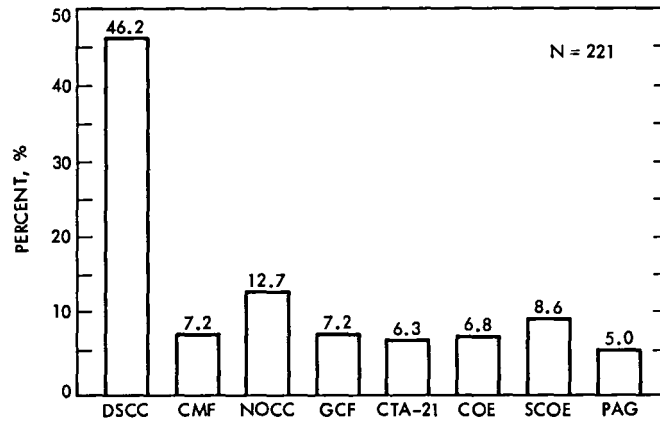


Fig. 1. Percentage distribution of training survey responses: facility/group

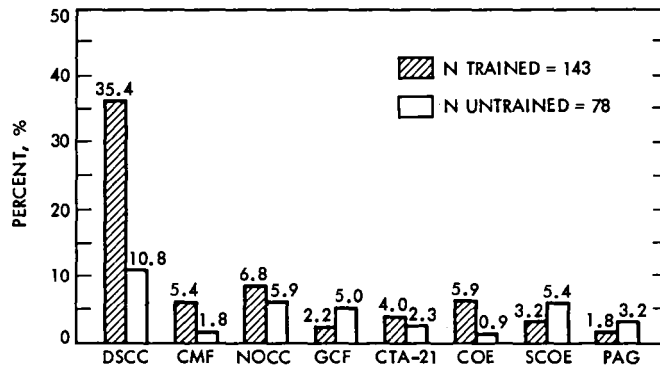


Fig. 2. Percentage distribution of trained vs untrained survey respondents: facility/group

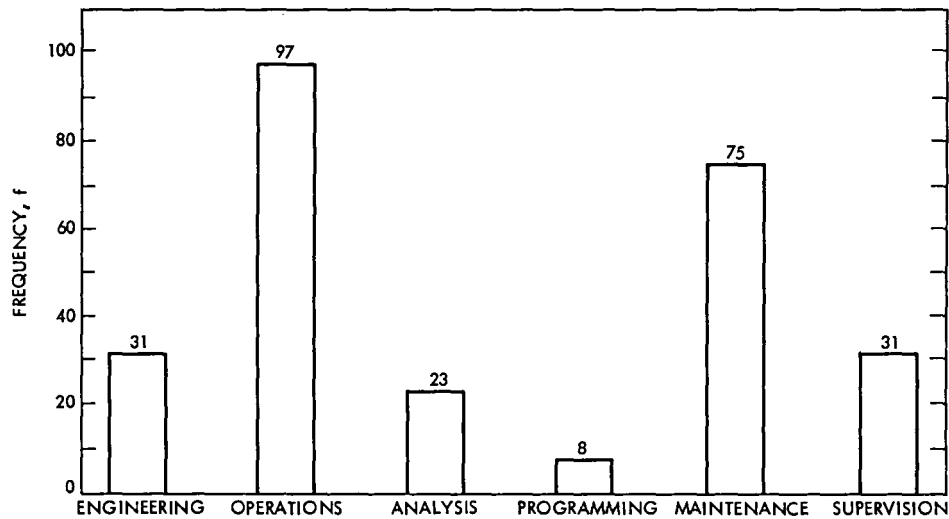


Fig. 3. Types of work performed by survey respondents: job categories

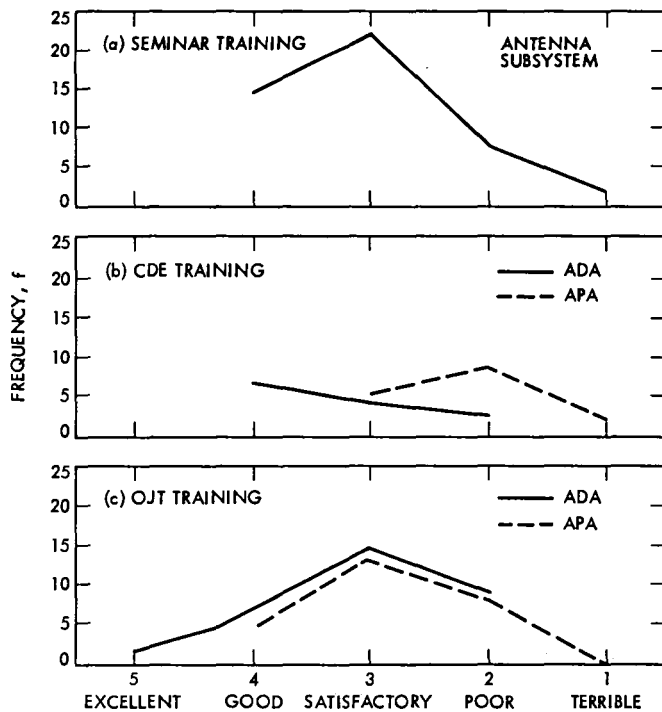


Fig. 4. Frequency curve comparison of antenna rating scores

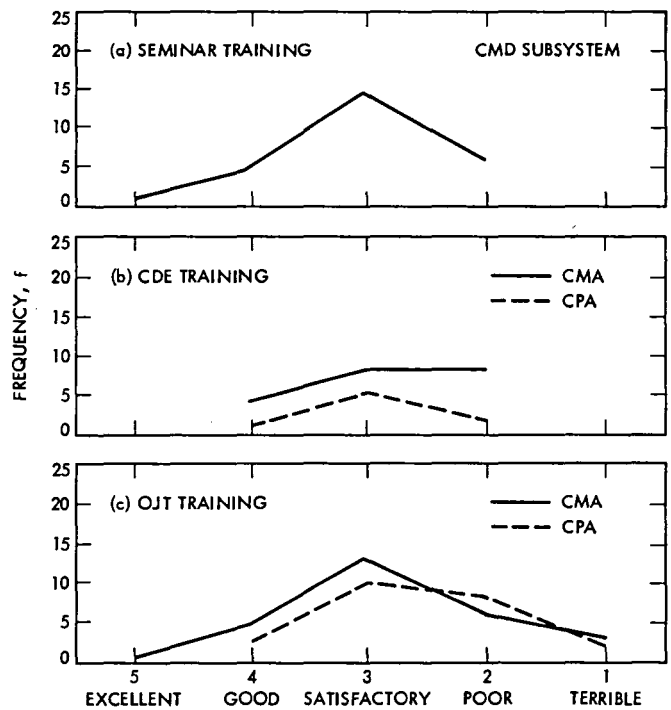


Fig. 6. Frequency curve comparison of command rating scores

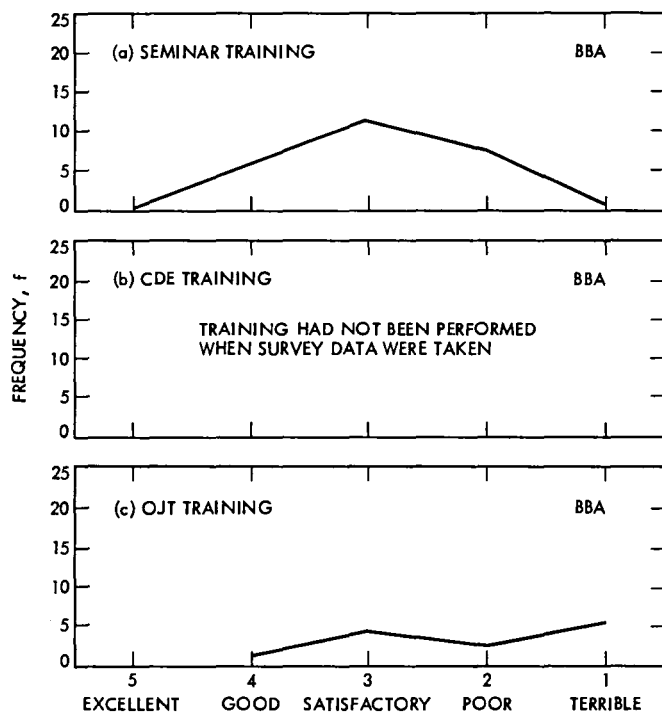


Fig. 5. Frequency curve comparison of baseband assembly rating scores

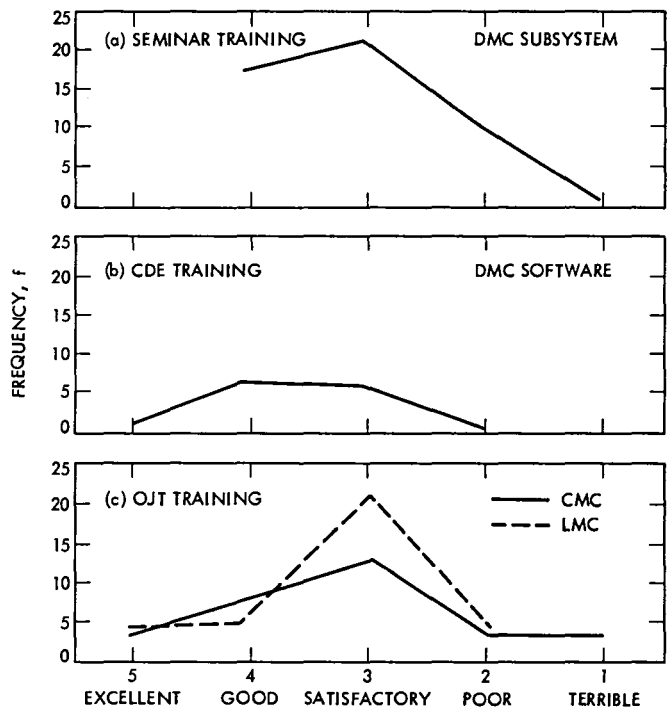


Fig. 7. Frequency curve comparison of monitor and control rating scores

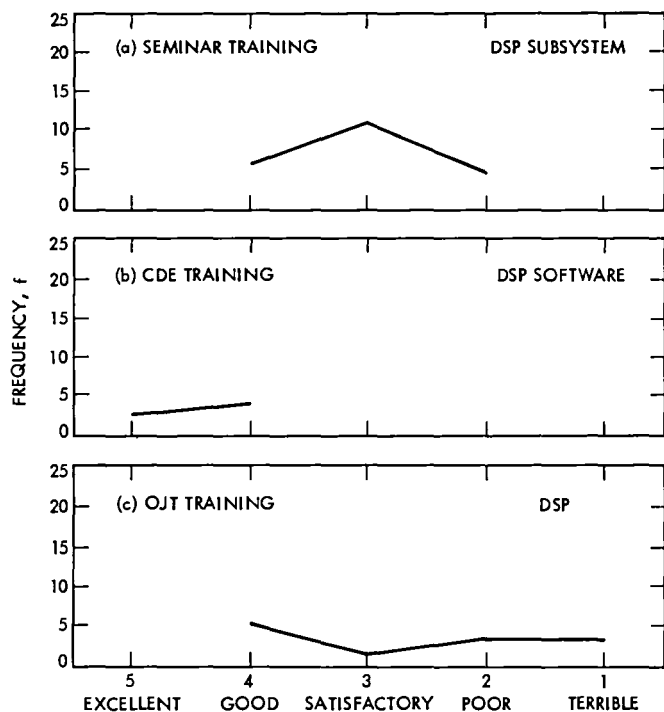


Fig. 8. Frequency curve comparison of digital spectrum processing rating scores

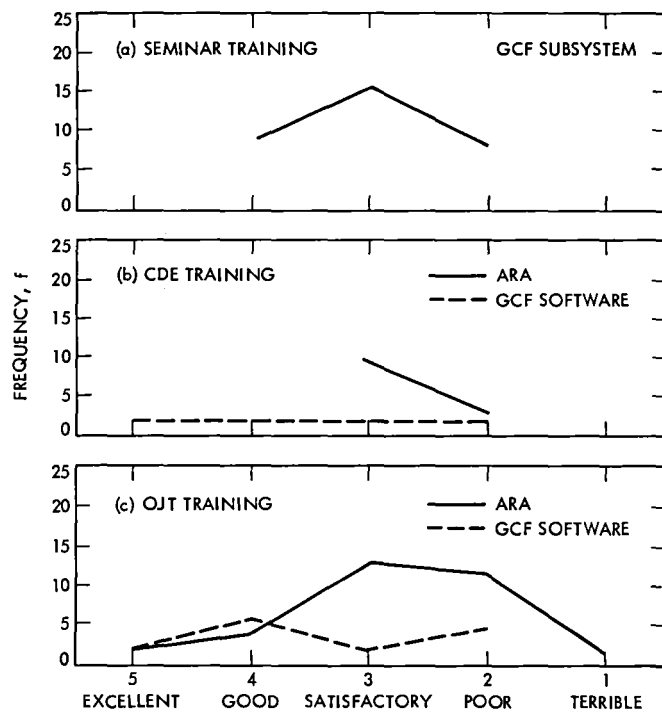


Fig. 10. Frequency curve comparison of ground communications rating scores

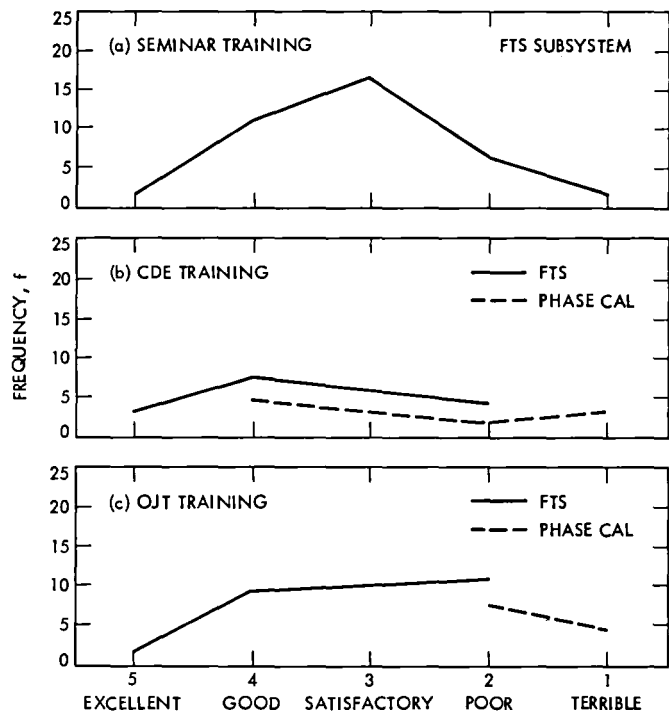


Fig. 9. Frequency curve comparison of frequency and timing rating scores

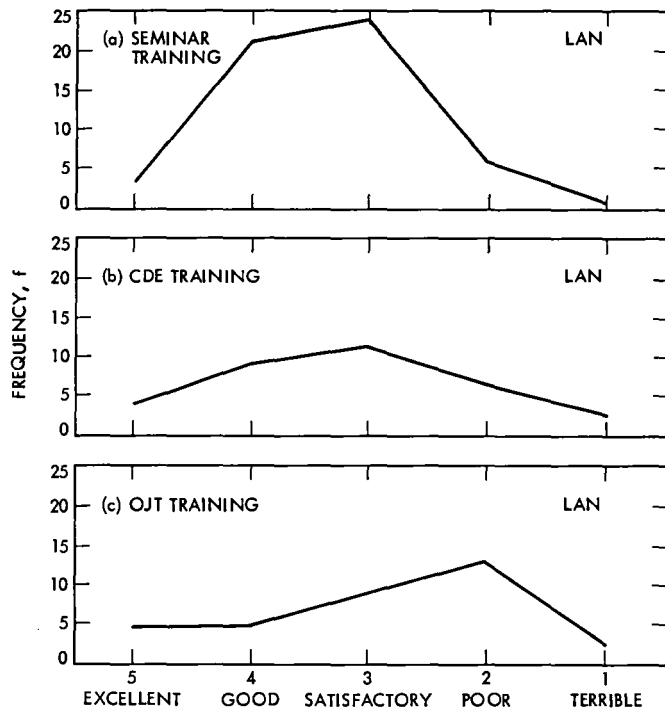


Fig. 11. Frequency curve comparison of local area network rating scores

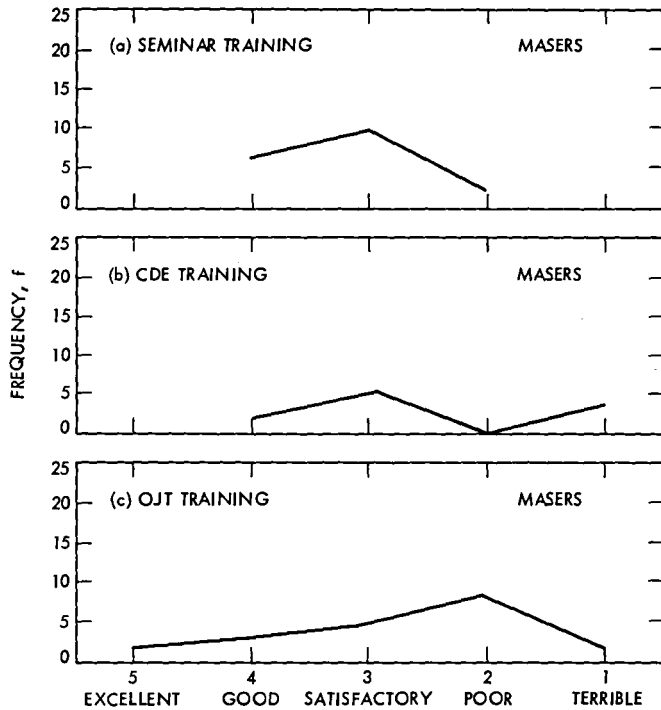


Fig. 12. Frequency curve comparison of maser rating scores

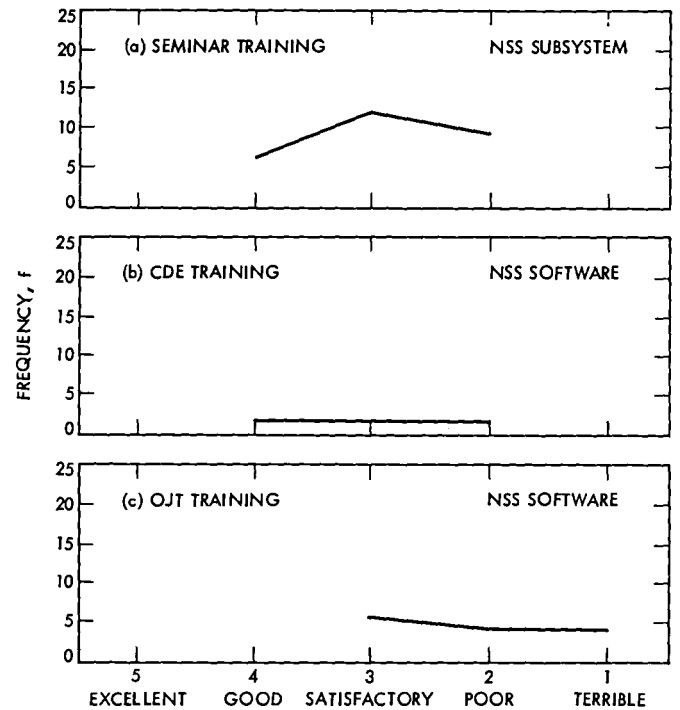


Fig. 14. Frequency curve comparison of network support subsystem rating scores

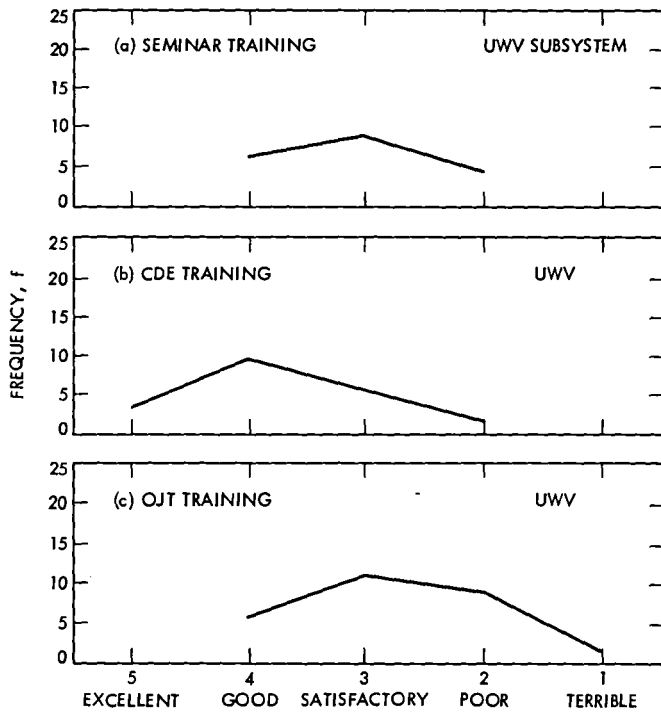


Fig. 13. Frequency curve comparison of microwave rating scores

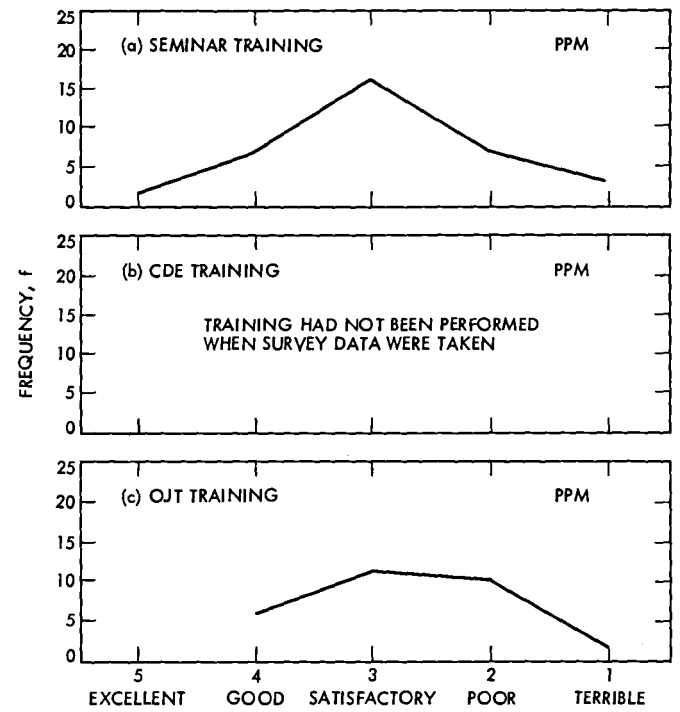


Fig. 15. Frequency curve comparison of precision power monitor rating scores

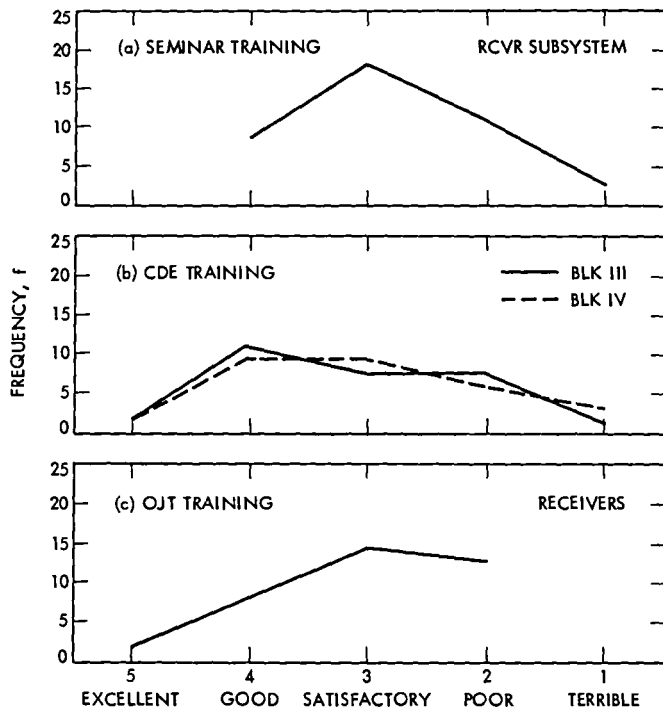


Fig. 16. Frequency curve comparison of receiver rating scores

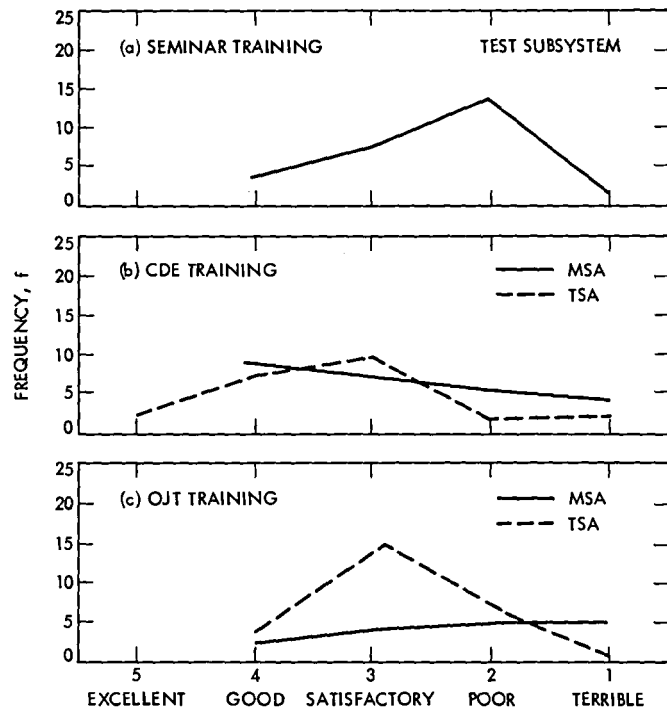


Fig. 18. Frequency curve comparison of test support subsystem rating scores

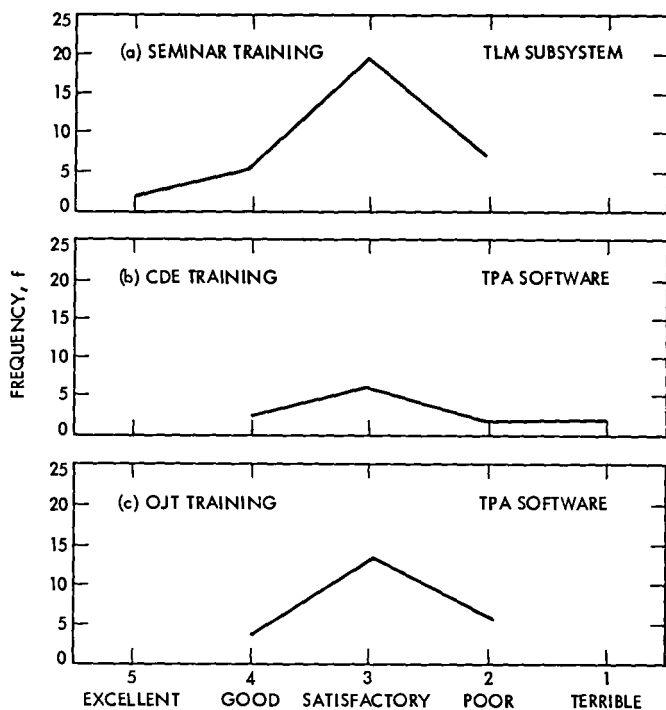


Fig. 17. Frequency curve comparison of telemetry rating scores

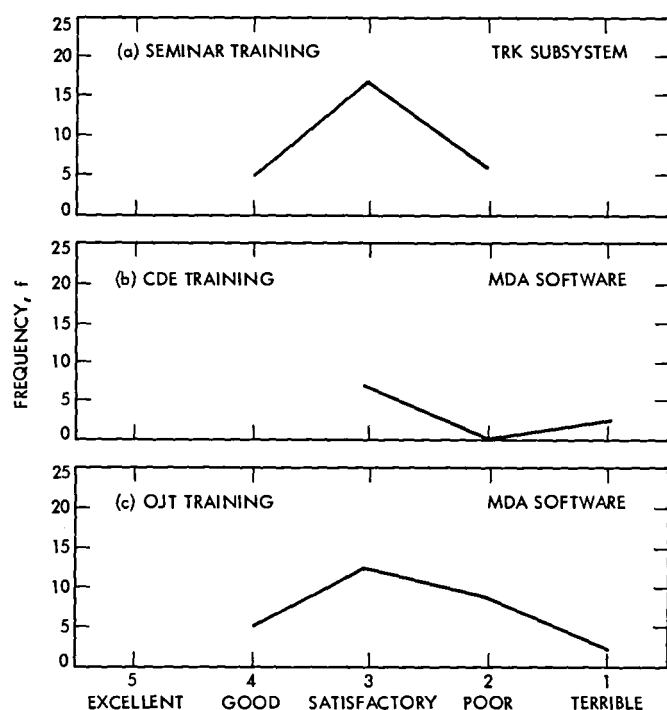


Fig. 19. Frequency curve comparison of tracking rating scores

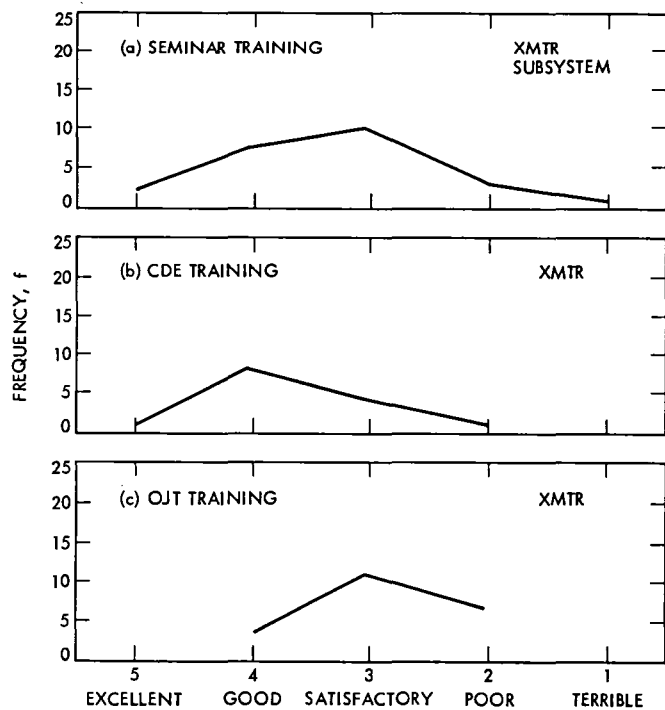


Fig. 20. Frequency curve comparison of transmitter rating scores

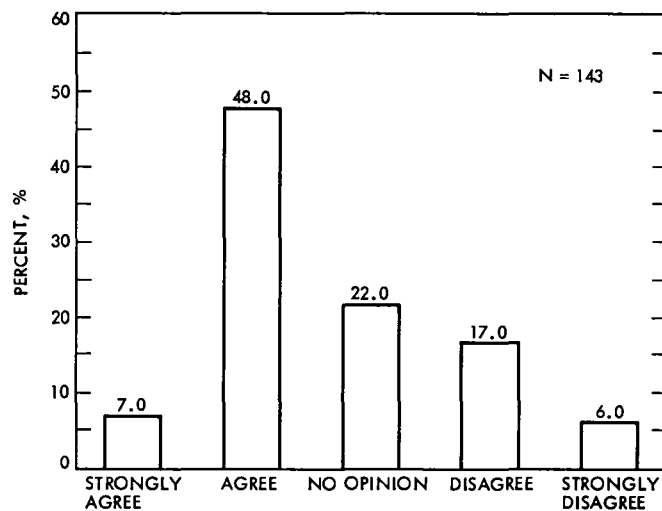


Fig. 22. Percentage distribution of training effectiveness opinion scores: opinion categories

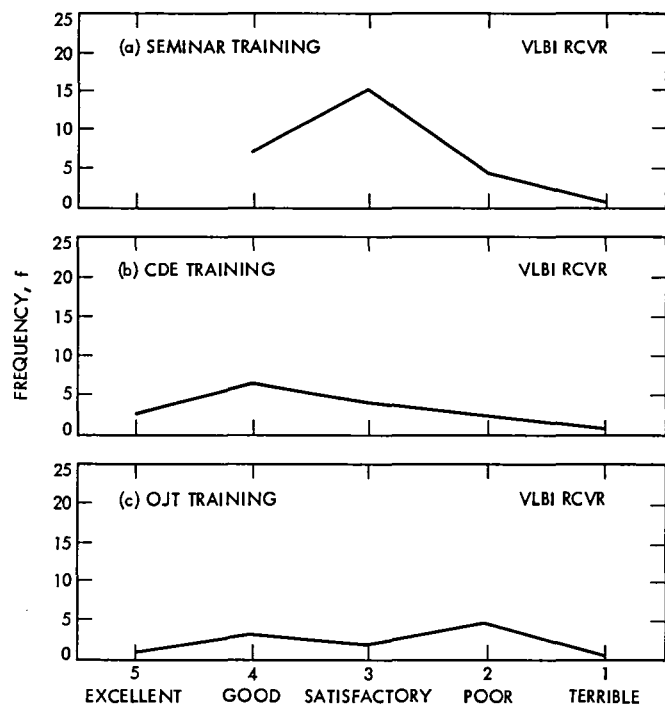


Fig. 21. Frequency curve comparison of VLBI receiver rating scores

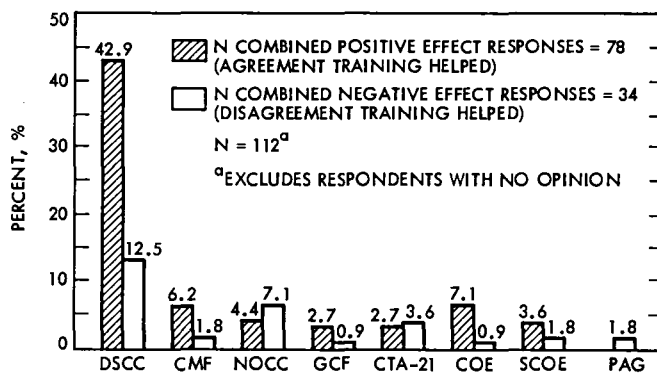


Fig. 23. Percentage distribution of positive/negative training effect responses

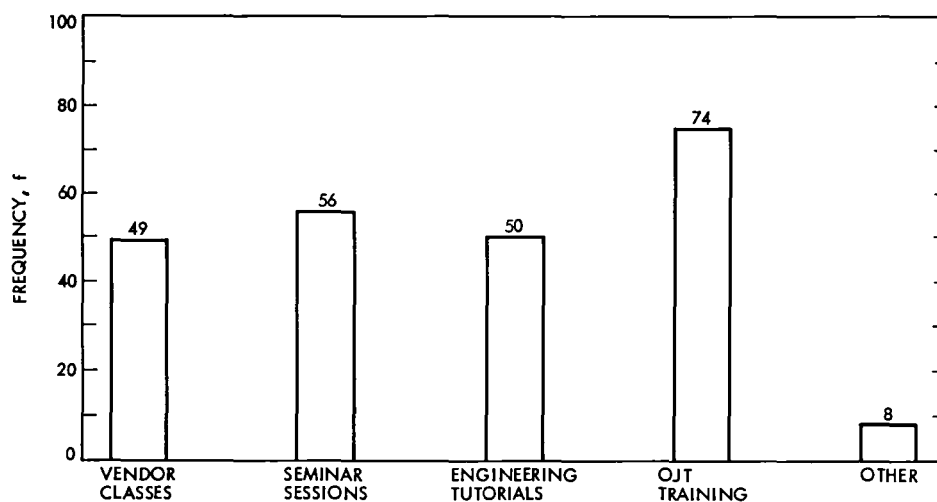


Fig. 24. Types of training requested by untrained/poorly trained respondents: method categories

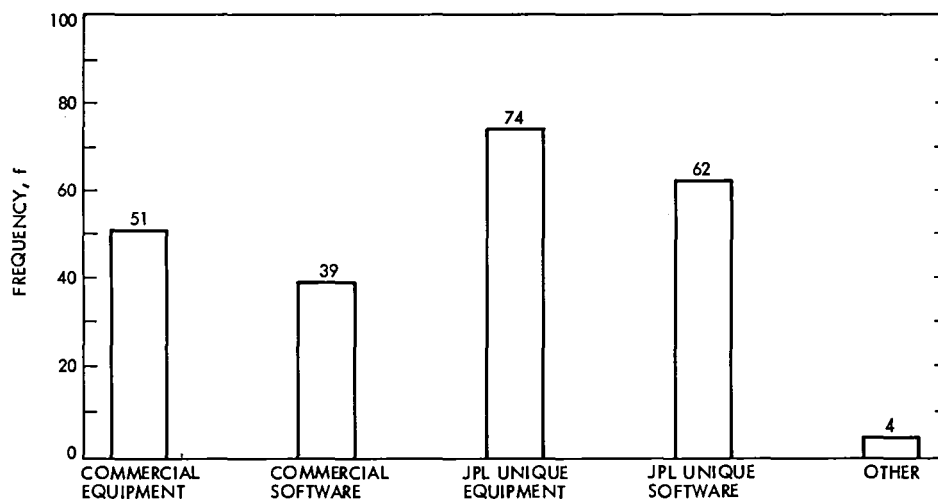


Fig. 25. Training classes requested by untrained/poorly trained respondents: subject categories

Error and Erasure Probabilities for Galileo Uplink Code

J. B. Berner

Telecommunications Systems Section

R. J. McEliece

California Institute of Technology

E. C. Posner

Office of Telecommunications and Data Acquisition

The Galileo uplink Frame Erasure probability and Undetected Frame Error probability are derived. The performance meets desired specifications under normal operations. The Galileo command system will work well even in an emergency condition, where the bit error rate into the command decoder is 1.00×10^{-3} (although Galileo's command threshold error rate is 1.00×10^{-5}).

I. Introduction

Galileo has a requirement for an undetected error rate for the End-to-End Command System of less than one undetected command error in 1×10^6 48-bit-long frames when the bit error rate into the CDU is 1×10^{-5} (Ref. 1). This translates into a bit error rate of 2.08×10^{-8} . While Galileo does not have any requirements on frame erasure rates, the draft Consultative Committee For Space Data Systems (CCSDS) command link coding recommendation is to have a 10^{-3} frame erasure probability rate and a 10^{-9} frame undetected error probability (Ref. 4).

These requirements are achieved with two forms of error detection/correction coding. All command frames are encoded with a block code, and messages spanning more than one frame are protected with an additional parity byte, called the *checksum byte*. If, due to the uplink coding, these requirements could be met at a higher input error probability, then the requirements on the signal-to-noise ratio (SNR) could be

relaxed, which would allow more link margin. One purpose of this article is to analyze the effect that the uplink coding has on the channel error and erasure probabilities and to discover if any SNR savings can be found. A second purpose is to derive command error and erasure rates at the command threshold for use in making emergency commanding decisions.

II. The Code

The code that the Galileo uplink channel is using is a shortened cyclic code with the following generator polynomial (Ref. 1):

$$g(x) = x^7 + x^6 + x^2 + 1 = (x + 1)(x^6 + x + 1) \quad (1)$$

This generator polynomial gives a (63, 56) code. The version that Galileo uses is shortened to (47, 40) by filling the first sixteen bits of the longer code with 0's.

The second factor of the polynomial, $(x^6 + x + 1)$, is the generator polynomial for the (63, 57) Hamming code. The first term, $(x + 1)$, expurgates the code, i.e., it lessens the number of information bits by one and increases the number of parity bits by one. Thus, although the code can be thought of as a shortened BCH code (Ref. 1), the more useful way of describing it is as a shortened, expurgated Hamming code.

A by-product of expurgating the code is that all codewords are even weight. This fact will be used later in determining the code's error properties.

There are several ways of viewing the code. The two most useful both use the parity check matrix, H . The first way looks at the Hamming code parity check matrix and the second way looks at the cyclic properties of the unshortened code. Both ways will be used to demonstrate various properties and to obtain results.

It can be shown (Ref. 2) that the parity check matrix of the unshortened, expurgated code is:

$$H' = \begin{bmatrix} H \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix} \quad (2)$$

where H is the parity check matrix of the (63, 57) Hamming code. Also, here the Hamming code is a linear code, of the form

$$H = [BI_{n-k}] \quad (3)$$

where I is the $(n - k) \times (n - k)$ identity matrix and B is a $(n - k) \times k$ matrix. Then, shortening H by t gives the following:

$$H'' = [B_t I_{n-k}] \quad (4)$$

where B_t is B minus the first t columns (Ref. 2). This result is easy to justify, since shortening a code is just replacing the first t information bits in the long code with 0's (and not sending these 0's), which zeros out the first t columns of the parity check matrix.

So the first goal is to get the shortened code into the form of Eq. (4). We do this by looking at the code in the second way, this time as a cyclic code.

The parity check matrix of a cyclic Hamming code can be written as:

$$H = [a^0, a^1, \dots, a^{62}] \quad (5)$$

where the row vector $[0100000]$ is equal to the primitive element a , which is a zero of the polynomial $x^6 + x + 1$ (Ref. 3). The i th column of H is equal to $g(a^i)$. Since the unshortened code is cyclic, we can rotate the matrix and get:

$$H = [a^7, a^8, \dots, a^{62}, a^0, \dots, a^6] \quad (6)$$

This is now in the form of Eq. (3). So we shorten and get:

$$H'' = [a^{23}, \dots, a^{62}, a^0, \dots, a^6] \quad (7)$$

Thus, we can generate the necessary parity check matrix from the Galois field elements a^i . Note that the two parity check matrices H' and H'' are equivalent under row reduction operations.

III. Error Properties of the Code

Let us look at the code's parity check matrix H' (Eq. [2]) and analyze the error properties. The code will correct one error and detect two errors when used as an error correcting code; it will detect three errors when operating as an error detecting code.

The received code word, Y , is equal to the transmitted codeword, X , plus the error pattern, Z . The syndrome of the received codeword is equal to the parity check matrix, H' , times the transpose of the received codeword. By definition of the parity check matrix, the syndrome of an unaltered codeword is the all 0 vector. Thus (Ref. 2),

$$\begin{aligned} \text{Syndrome} &= H' Y^T \\ &= H' (X^T + Z^T) \\ &= H' X^T + H' Z^T \\ &= H' Z^T \end{aligned}$$

So, we are only interested in the error patterns; the transmitted codeword does not affect the error properties. So from here on, for ease of calculation, we will assume that the all zero codeword is sent.

There are two distinct flavors of error patterns: odd weight and even weight, where weight is defined as the number of 1's in the codeword or error pattern. For more than two errors and the parity check matrix H' , the following is true:

Even Weight Error Pattern:

$$H'Z^T = \begin{array}{cc} (a) & (b) \\ x & 0 \\ x & 0 \\ x & 0 \\ x & 0 \\ x & 0 \\ x & 0 \\ 0 & 0 \end{array}$$

where x is either a 0 or a 1, Z^T is the transpose of the received codeword, and (a) and (b) are the only possible even weight error pattern syndromes.

The possible syndrome values for the Hamming code are the columns of the parity check matrix and the all 0 vector (the syndrome of a codeword) (Ref. 2). Since the parity check matrix H' has all 1's in its last row, the only even weight error patterns that will escape detection are the ones that are codewords. Thus, the code will detect error patterns with syndromes of pattern (a) and will not detect error patterns with syndromes of pattern (b).

Odd Weight Error Pattern:

$$H'Z^T = \begin{array}{cc} (c) & (d) \\ x & 0 \\ x & 0 \\ x & 0 \\ x & 0 \\ x & 0 \\ x & 0 \\ 1 & 1 \end{array}$$

If an odd weight error pattern occurs, the above syndromes are the only possible. The code will detect error patterns with syndromes of pattern (d), since the parity check matrix has no columns of this form, and will not detect error patterns with syndromes of pattern (c), unless the code is shortened. If the code is shortened, error patterns with syndromes equal to the zeroed out parity check matrix columns will be detected.

So, to determine the error statistics (how many detected and how many not detected for weight I), we must take a different approach for the even and odd weight errors. For the even weight errors, all we need is the weight enumerator (the number of words of weight I for all I) of the code; for the odd weight errors, we will have to do more work.

A. Even Weight Errors

There are 2^{40} possible code words; thus, counting the weights of the codewords borders on the impossible. However, using the dual code, the weight enumerator of the code can easily be found.

The dual code of a code is the code produced when using the parity check matrix as the generating matrix. This gives a code with $2^7 = 128$ codewords, easily countable with a computer. Then one can use the MacWilliams Identities (Ref. 2) to get the weight enumerator of the code from the weight enumerator of the dual code. The weight enumerator then gives the number of undetectable errors of weight I . Subtract the undetectable errors from the total number of errors of weight I , $\binom{47}{I}$, and we have the number of detectable even weight errors. The work is thus half done.

B. Odd Weight Errors

In the Galileo uplink, the code is used in two modes, error-detect (ED) and error-detect-and-correct (EDC). In error-detect mode, the code is used as an error detecting code; it does not try to correct the errors. This allows the code to detect all error patterns of weight three or less. In error-detect-and-correct mode, which is used only on data frames, the code is used as an error correcting code; thus, it can correct all one-error patterns and detect all two-error patterns.

In the error-detect mode, only even weight errors with the all 0 syndrome (patterns that are codewords) escape detection. Since the code has no odd weight codewords, if the code is operated in error detect mode, there are no undetectable odd weight errors.

However, if the code is operated in error-detect-and-correct mode, the situation is very different. The code will make an error if the error pattern causes a syndrome that is a column in the parity check matrix, i.e., the code assumes a single error has been made. When the code assumes a single error has been made, it will try to correct the error. The correction attempt changes one bit in the error pattern; this new bit may cancel out one error bit or it just adds one error bit. Thus, to generate an error pattern of weight $2K + 1$, there are $2K + 2$ ways of canceling a bit of a codeword of weight $2K + 2$ and there are $47 - 2K$ ways of adding a bit to a codeword of weight $2K$. So, the formula for the undetectable errors of weight $2K + 1$ is:

$$U = (47 - 2K) \times A_{2K} + (2K + 2) \times A_{2K+2} \quad (8)$$

where A_i is the number of codewords of weight i .

Once again, the number of detectable errors is the number of error patterns of weight I minus the number of undetectable errors.

The weight enumerator, the number of detectable and undetectable errors, and the details of the calculations are given in Appendix A.

The next task is to compute for a given input (to the code) symbol error probability, p , the output frame error and erasure probabilities. This is straightforward:

For the error-detect-and-correct mode,

$$P_{\text{error}} = \sum_{i=2}^{47} p^i (1-p)^{47-i} U_i \quad (9)$$

$$P_{\text{erase}} = \sum_{i=2}^{47} p^i (1-p)^{47-i} D_i \quad (10)$$

where U_i and D_i are the error-detect-and-correct mode undetectable and detectable errors of weight i , respectively.

For the error-detect mode,

$$P_{\text{error}} = \sum_{i=1}^{47} p^i (1-p)^{47-i} U'_i \quad (11)$$

$$P_{\text{error}} = \sum_{i=1}^{47} p^i (1-p)^{47-i} D'_i \quad (12)$$

where U'_i and D'_i are the error-detect mode undetected and detected errors of weight i , respectively, as described above.

For an input symbol error of 1.00×10^{-5} , the output probabilities are as follows. For error-detect-and-correct,

$$P_{\text{error}} = 1.170 \times 10^{-11}$$

$$P_{\text{erase}} = 1.081 \times 10^{-7}$$

For error-detect,

$$P_{\text{error}} = 2.926 \times 10^{-17}$$

$$P_{\text{erase}} = 4.699 \times 10^{-4}$$

Figures 1 and 2 are plots of the error and erasure probabilities versus the input channel error probability for the error-

detect-and-correct mode (Fig. 1) and the error-detect mode (Fig. 2). We see that the error-detect-and-correct mode provides error rates of 10^{-5} to 10^{-14} for $p = 10^{-3}$ to 10^{-6} , while the error-detect mode provides error rates of 10^{-9} to 10^{-21} for the same range of p .

IV. The Command Frame

A Galileo command frame is 48 bits long. It is equal to the 47-bit codeword, with a 0 fill bit appended to the end to make the frame length be a multiple of eight bits. The frame format (Ref. 1) is shown in Fig. 3.

There are two types of command frames, Single Frames and Multiple Frames (Ref. 1):

- (1) Single Frame — A single command frame, decoded in error-detect mode.
- (2) Multiple Frames — A header frame, called the *Block Command frame*, which is decoded in error-detect (ED) mode, followed by 1 to 31 Data frames, decoded in error-detect-and-correct (EDC) mode. The last information byte of the last data frame is the *checksum byte*, which is the result of exclusive-ORing all the previous information bytes. After decoding the data frames, the parity bits are discarded and the information bytes are exclusive-ORed. If the result is nonzero, an erasure is declared.

Thus, there will be two sets of error and erasure probabilities, one for the Single Frames and one for the Multiple Frames.

A. Single Frames

The Single Frame is just a codeword decoded in the error-detect mode. The resulting error and erasure probabilities are shown in Fig. 2. For an input symbol error probability of 1.00×10^{-5} , the Single Frame probabilities are:

$$P_{\text{error}} = 2.926 \times 10^{-17}$$

$$P_{\text{erase}} = 4.699 \times 10^{-4}$$

From Fig. 2, we see that the coding provides error rates that are at least 10^{-8} below the erasure rates.

B. Multiple Frames

Before finding the required probabilities, we must understand how the overall decoding system (Hamming Code decoder and *checksum byte*) works.

The *Block Command frame* is decoded. If errors are detected, processing of the Multiple Frame stops and an erasure

is declared. If no errors are detected, the Data frames are decoded. If an uncorrectable error is detected, processing of the Multiple Frame stops and an erasure is declared. If all frames are decoded without an erasure, the data bytes (the 40 information bits per Data frame) are exclusive-ORed, and if the result is nonzero, the processing is stopped and an erasure is declared. The last data byte of the last Data frame is called the *checksum byte*, since this byte is appended onto the data stream to make the exclusive-ORing of the data bytes be zero (Ref. 1).

Since the code and checksum are linear, any frame can be used for the analysis. So we will assume that the all zero frame is sent; thus, a 1 denotes an error.

Assume that the codeword has passed through the decoder. We now drop the 7 parity bits and only consider the 40 information bits (five information bytes). If we break each byte into its 8 bits and look at each bit position (or slot) in the byte, two things are obvious:

- (1) If there is an odd number of 1's in any slot, the exclusive-ORing process will result in a 1 being in that slot; thus, an erasure will be declared.
- (2) If there is an even number of 1's in each slot, the exclusive-ORing process will result in a 0 being in that slot; so, no error will be detected.

Thus, for an error pattern to escape detection, there must be an even number of 1's in each slot.

Now, the statistics of the shortened, expurgated Hamming code are such that if the decoder makes an error, it outputs even weight error patterns of weight four or more. And, as was shown above, if there is an odd number of channel errors that the decoder cannot correct or detect, the decoder outputs either one less error or one more error. Thus, the minimum number of channel errors at the input to the decoder to cause an error that is undetectable (or uncorrectable) by the decoder is three. So the probability of an error that is undetectable by the checksum is proportional to p^3 , for small p .

Since we have multiple data frames, the question is, do we worry about multiple frame errors, or do we just look at a single frame in error? The probability of a multiple undetected frame error would be equal to $C \times p^6 + O(7)$ (since p^3 is the minimum for a single frame), where C is a constant. Since p is less than 10^{-3} , the probability of a double frame error is, at the least, 10^{-9} less than the probability of a single frame error. Thus, we can ignore all multiple data frame errors and concentrate on undetected single data frame errors. And, since we are dealing with a low p , the number of undetectable combinations we have to look at is small.

The undetectable error patterns are those with an even number of 1's in each slot. This only occurs in error patterns with an even number of 1's in the information bits.

There are 80 possible undetectable weight 2 information-bit error patterns. The corresponding codeword weights are (see Appendix B for the derivation):

37 weight 4

32 weight 6

11 weight 8

There are 2840 possible undetectable weight 4 information-bit error patterns. The corresponding codeword weights are:

37 weight 4

892 weight 6

1591 weight 8

320 weight 10

There are 58800 possible undetectable weight 6 information bit error patterns. The corresponding codeword weights are:

924 weight 6

19253 weight 8

32244 weight 10

6379 weight 12

Before we begin, some symbols that we will use:

P_R^{ED} = Probability that the decoder declares an erasure in ED mode

P_R^{EDC} = Probability that the decoder declares an erasure in EDC mode.

P_E^{ED} = Probability that the decoder makes an error in ED mode

P_E^{EDC} = Probability that the decoder makes an error in EDC mode

With the above numbers in hand, and with the knowledge that the code converts odd weight error patterns into even weight patterns, we can calculate the probability that one frame will have an error pattern that is undetectable by the *checksum* (called P_u), given that the decoder made an error.

$$\begin{aligned}
 P_u &= \left(\text{sum of (the number of patterns of weight } i \text{ that are undetectable)} \right. \\
 &\quad \times \left. \frac{p^i \times (1-p)^{n-i}}{P_E^{EDC}} \right) \\
 P_u &= \frac{(45 \times 0 + 4 \times (37 + 37)) \times p^3 \times (1-p)^{44}}{P_E^{EDC}} \\
 &\quad + \frac{(37 + 37) \times p^4 \times (1-p)^{43}}{P_E^{EDC}} \\
 &\quad + \frac{(43 \times (37 + 37) + 6 \times (32 + 892 + 924))}{P_E^{EDC}} \\
 &\quad \times p^5 \times (1-p)^{42} \\
 &\quad + \frac{0(p^6)}{P_E^{EDC}} \\
 P_u &= \frac{(296p^3(1-p)^{44} + 74p^4(1-p)^{43} + 14270p^5(1-p)^{42})}{P_E^{EDC}} \quad (14)
 \end{aligned}$$

And the probability of a frame having a detectable error by the checksum (P_d), given that the decoder made an error, is:

$$P_d = 1 - P_u \quad (15)$$

And we will assume that we have n data frames, where n is between 1 and 31.

Now let's calculate the erasure probability of the system, remembering that an erasure stops the processing:

$$\begin{aligned}
 P_{\text{erase}} &= \text{Prob (erasure in Block Command frame)} \\
 &\quad + \text{Prob (erasure in one of the } n \text{ data frames)} \\
 &\quad + \text{Prob (checksum declares erasure)} \\
 P_{\text{erase}} &= P_R^{ED} + (1 - P_R^{ED}) \times P_R^{EDC} \\
 &\quad \times \sum_{i=1}^{n-1} (1 - P_R^{EDC})^{i-1} \\
 &\quad + \binom{n}{1} \times P_d \times (1 - P_R^{EDC} - P_E^{EDC})^{n-1} \\
 &\quad \times (1 - P_R^{ED}) \times P_E^{EDC} \quad (16)
 \end{aligned}$$

And, the error probability (again assuming one undetectable error):

$$\begin{aligned}
 P_{\text{error}} &= \text{Prob (error in Block Command frame)} \\
 &\quad + \text{Prob (undetectable error in data frame)} \\
 P_{\text{error}} &= P_E^{ED} \times (1 - P_E^{EDC} - P_R^{EDC})^n \\
 &\quad + (1 - P_E^{ED} - P_R^{ED}) \times \binom{n}{1} \times P_u \\
 &\quad \times (1 - P_E^{EDC} - P_R^{EDC})^{n-1} \times P_E^{EDC} \quad (17)
 \end{aligned}$$

The resulting probabilities are plotted in Figs. 4, 5, and 6, for $n = 5, 10$, and 31 . For a channel error probability of $p = 10^{-5}$, the results are:

For $n = 5$,

$$P_{\text{error}} = 1.48 \times 10^{-12}$$

$$P_{\text{erase}} = 4.70 \times 10^{-4}$$

For $n = 10$,

$$P_{\text{error}} = 2.96 \times 10^{-12}$$

$$P_{\text{erase}} = 4.71 \times 10^{-4}$$

For $n = 31$,

$$P_{\text{error}} = 9.17 \times 10^{-12}$$

$$P_{\text{erase}} = 4.73 \times 10^{-4}$$

V. Examining the Results

Let us assume the CCSDS requirements (10^{-3} frame erasure probability and 10^{-9} frame undetected error probability). At least for now, we will assume these requirements for the Multiple Frame. So, using Figs. 2, 4, and 5, we get the following undetected error probabilities at an erasure probability of 10^{-3} :

For $n = 5$,

$$P_{\text{error}} = 1.00 \times 10^{-11}$$

For $n = 10$,

$$P_{\text{error}} = 2.50 \times 10^{-11}$$

For $n = 31$,

$$P_{\text{error}} = 8.00 \times 10^{-11}$$

These values occur at a channel error probability of 2.00×10^{-5} or $1.00 \times 10^{-4.70}$, which gives an E_b/N_0 improvement of 0.33 dB over the E_b/N_0 required for a 1.00×10^{-5} channel error probability.

Now, let us consider the $n = 31$ case (this has the highest error probability). The error probability is equal to $1.00 \times 10^{-10.09691}$. This means we have one frame in error for every $10^{10.09691}$ frames. At 32 bits per second, and 48 bits per frame, that means that it will take 594 years for an error to occur. Or, to put it another way, we could have started transmitting commands continuously when Galileo was alive in the mid 1600's and still would not expect an error until sometime in the twenty-third century (Jim Taylor, private communication).

VI. Command Threshold

We now want to look at what happens at the command threshold. The command threshold is assumed to be the point where the input symbol error rate, p , is 1.00×10^{-3} .

In Section IV, we made an assumption that since the symbol error rate would be less than 10^{-3} , we only had to worry about one frame in error when we have a multiple frame command. Since we are at $p = 10^{-3}$, we should look at this again:

For the probability of erasure, P_{erasure} , the only term affected is Prob (*checksum* declares erasure). This term is equal to the sum of Prob (*checksum* declares erasure | number of frame

errors = i) \times Prob (number of frame errors = i). For each i , the summed terms are bounded by:

$$\binom{n}{i} \times (P_{\text{E}}^{\text{EDC}})^i$$

At $p = 10^{-3}$, $P_{\text{E}}^{\text{EDC}} = 1.1 \times 10^{-5}$. The later terms in the summation are thus insignificant when compared to the first term (i equals 1). Thus the equation for P_{erasure} (Eq. [16]) holds at $p = 10^{-3}$.

The probability of error is even easier. The probability, P_u , of an undetectable error, given that a frame error has occurred, is the summation over i of $D \times (p^3)^i$, where i is the number of frames that have errors and D is a constant that depends on i . With $p = 10^{-3}$, only the first term (i equals 1) is significant. So the equation for P_{error} (Eq. [17]) holds.

The equations give the following numbers for $p = 10^{-3}$:

Single Frame:

$$P_{\text{erasure}} = 4.6 \times 10^{-2}$$

$$P_{\text{error}} = 2.8 \times 10^{-9}$$

Multiple Frame:

$$n = 5, \quad P_{\text{erasure}} = 5.0 \times 10^{-2}$$

$$P_{\text{error}} = 1.3 \times 10^{-6}$$

$$n = 10, \quad P_{\text{erasure}} = 5.5 \times 10^{-2}$$

$$P_{\text{error}} = 2.7 \times 10^{-6}$$

$$n = 31, \quad P_{\text{erasure}} = 7.6 \times 10^{-2}$$

$$P_{\text{error}} = 8.1 \times 10^{-6}$$

So, even in the worst case ($n = 31$), in an emergency, only 7.6% of the frames would get erased and more importantly, those that get through the decoder have an error probability of 8.1×10^{-6} . This means that it is safe to blind command (the sending of commands without any hope of the spacecraft verifying them) Galileo in case of a spacecraft emergency.

VII. Summary

The Galileo uplink frame erasure and undetectable frame error probabilities have been derived. It has also been shown that the Expurgated Hamming code with *checksum byte* gives better than required error protection for a given erasure probability.

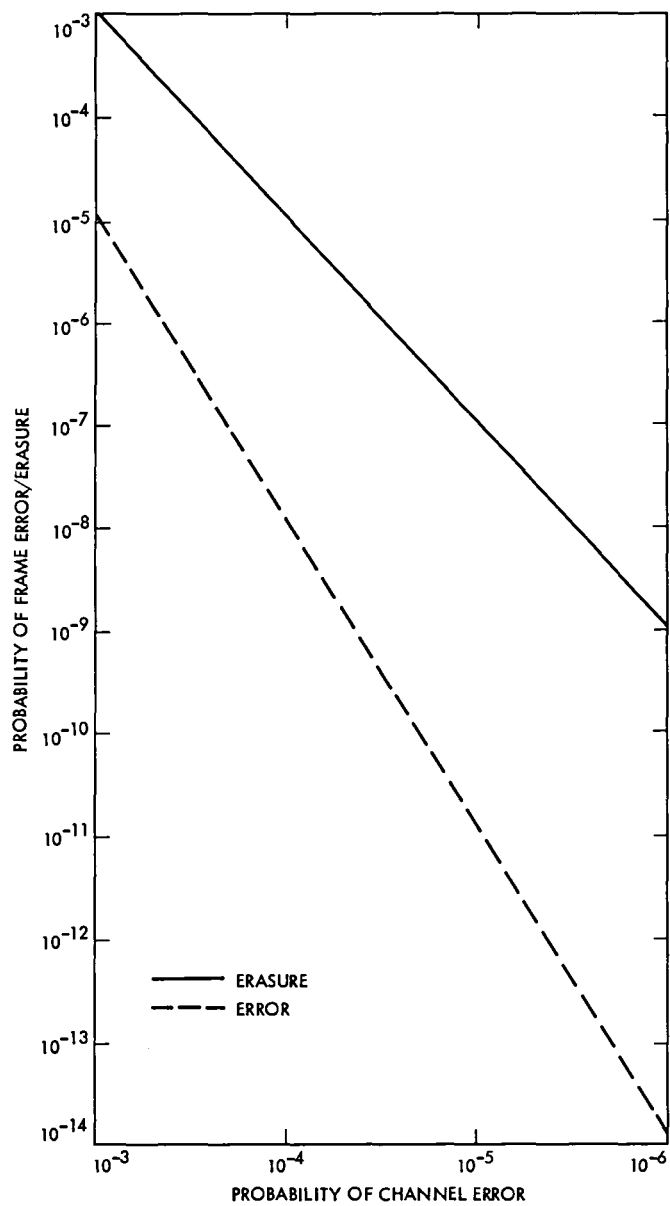


Fig. 1. Probabilities for error correct mode

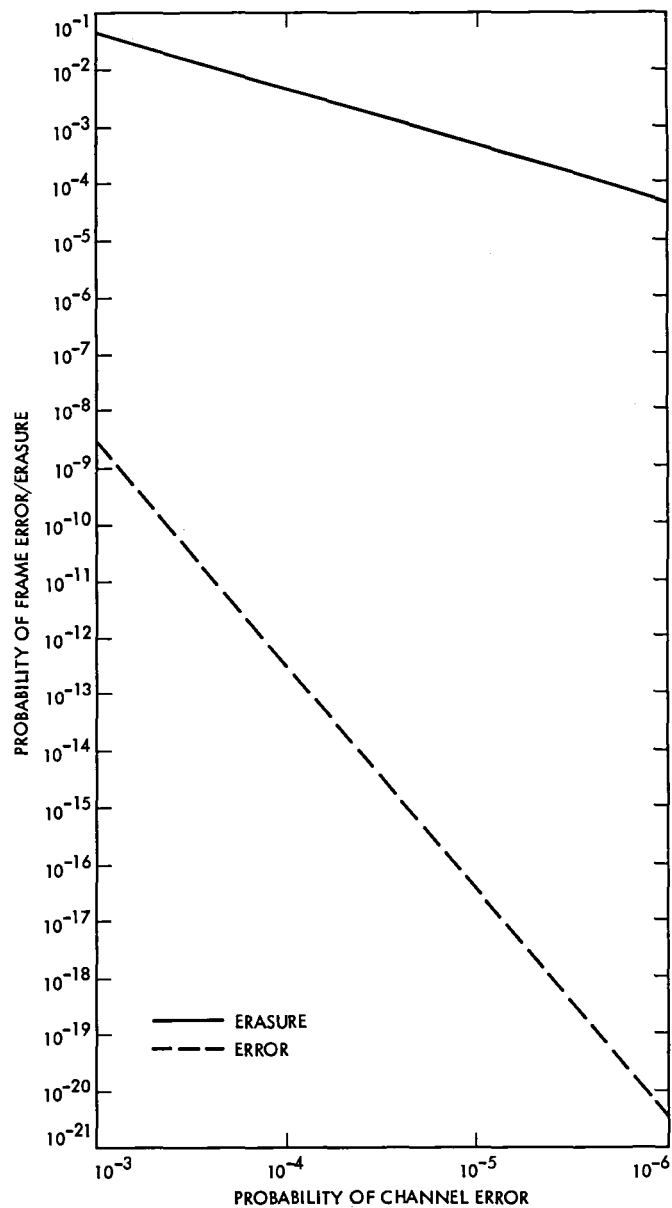


Fig. 2. Probabilities for error detect mode

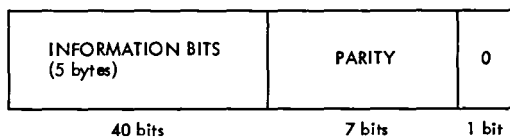


Fig. 3. Galileo command frame format

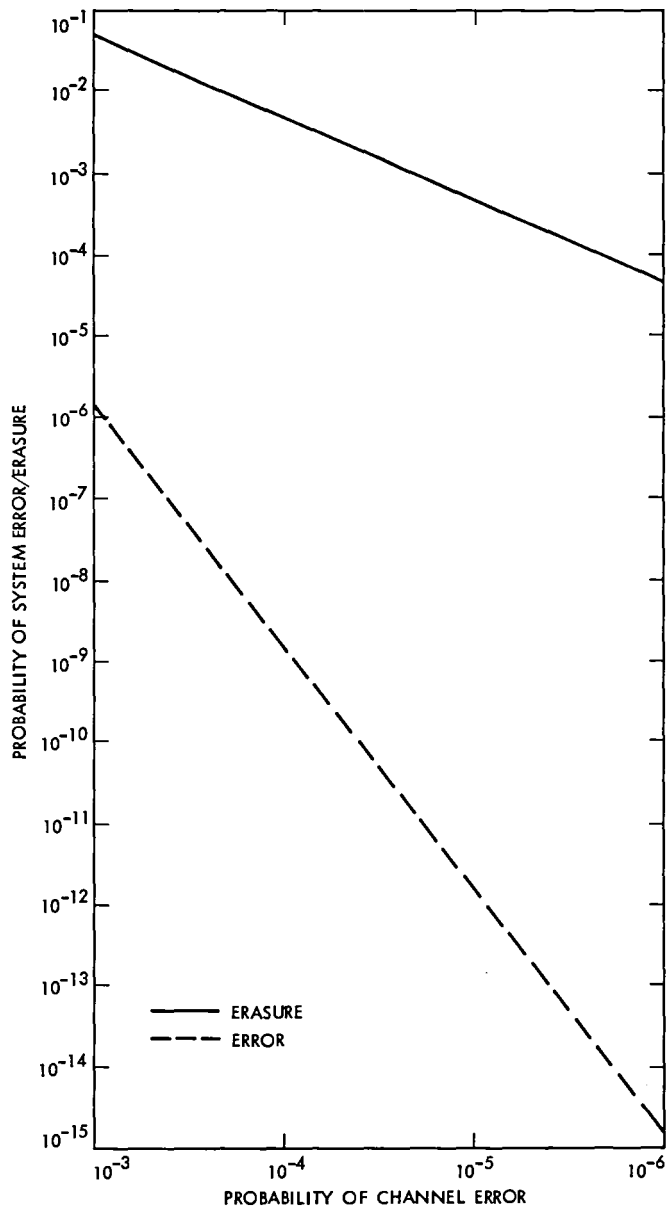


Fig. 4. Probabilities for multiple frames where $n = 5$

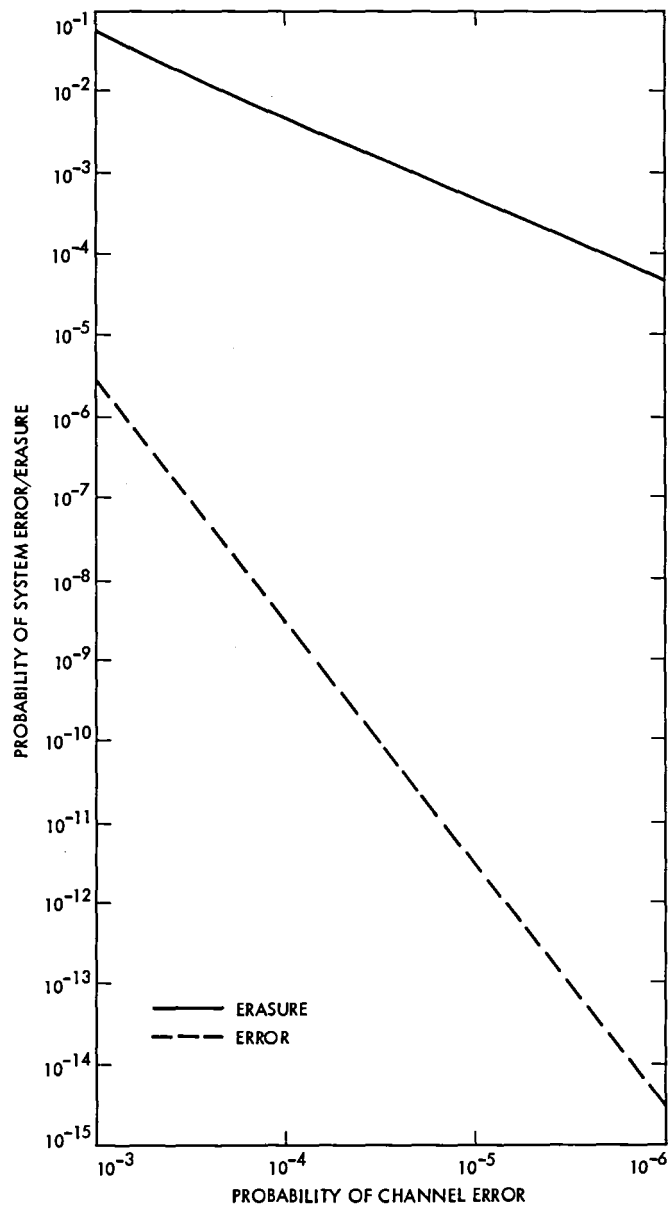


Fig. 5. Probabilities for multiple frames where $n = 10$

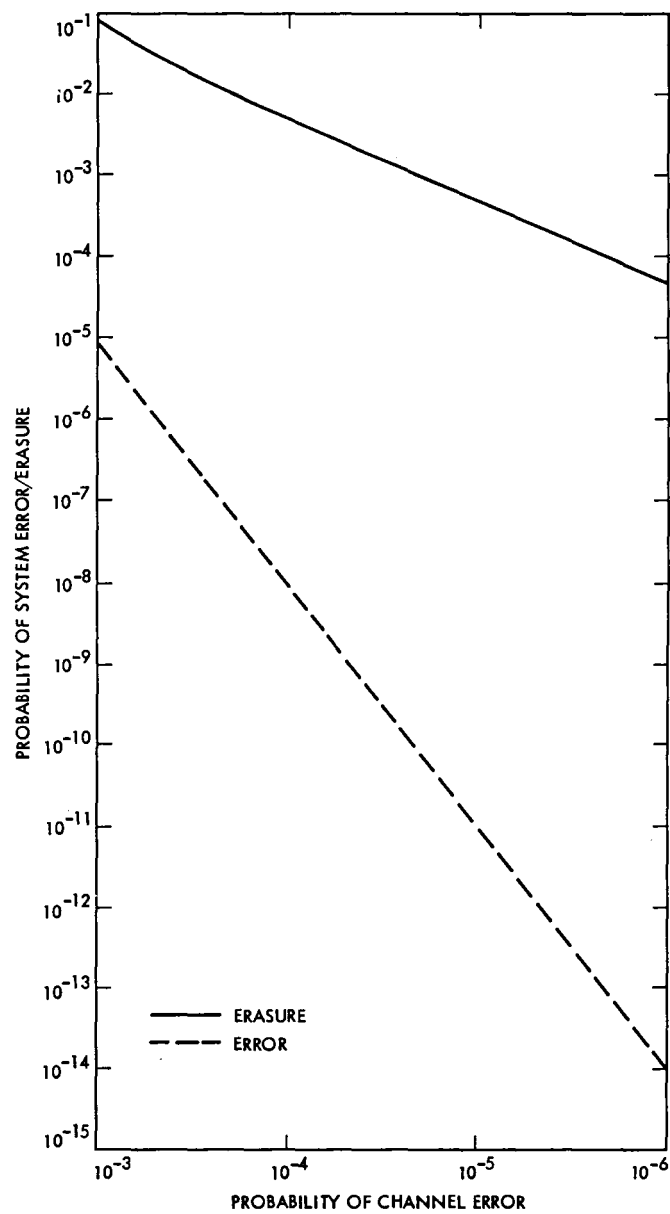


Fig. 6. Probabilities for multiple frames where $n = 31$

Appendix A

Code Statistics

The goal is to find, for each error pattern weight, the number of detectable and undetectable error patterns. To do this, we must first find the weight enumerator of the code.

The weight enumerator is found via the dual code of the code. The dual code is the code created by using the parity check matrix of our code as a generator matrix. This creates a (47, 7) code, which has only $2^7 = 128$ codewords. The weight enumerator of this code is easily found by computer by generating the 128 7-bit sequences, encoding them with the parity check matrix, and then counting the 1's in each codeword. The results of this are in Table A-1.

The next step is to use the MacWilliams identities to calculate the weight enumerator of the code. The MacWilliams identities state (Ref. 2) that if $B(z)$ is the weight enumerator of an (n, k) linear code and $A(z)$ is the weight enumerator of its dual code, then:

$$A(z) = 2^{-k} \sum_{i=0}^n B_i (1-z)^i (1+z)^{n-i} \quad (\text{A-1})$$

In our case, $k = 7$, $n = 47$, $B(z)$ is the weight enumerator of the dual code, and $A(z)$ is the weight enumerator of the code. So, again using a computer, the weight enumerator can be calculated. The results are shown in Table A-2. Now we can calculate the number of detectable and undetectable error patterns of weight i . First, let's do it for the EDC mode.

As was discussed in Section III, the only undetectable even weight error patterns are those that are codewords. Thus, for errors of weight $2i$, the number of undetectable and detectable patterns are:

$$\text{Undetectable} = A(2i) \quad (\text{A-2})$$

$$\text{Detectable} = \binom{47}{2i} - A(2i) \quad (\text{A-3})$$

And, as was shown in Section III, the number of odd weight detectable and undetectable error patterns is:

$$\text{Undetectable} = (47 - 2i) \times A(2i) + (2i + 2) \times A(2i + 2) \quad (\text{A-4})$$

$$\begin{aligned} \text{Detectable} = & \binom{47}{2i+1} - (47 - 2i) \times A(2i) \\ & - (2i + 2) \times A(2i + 2) \end{aligned} \quad (\text{A-5})$$

The results are shown in Table A-3.

For the ED mode, the even weight error pattern breakdown is the same. The difference is in the odd weight error patterns. Since we are in ED mode, errors that look like single errors are detected, not corrected. So, there are no undetectable error patterns of odd weight. These results are in Table A-4.

Table A-1. Dual code weight enumerator

Weight	Number of Words
0	1
19	3
20	4
21	5
22	23
23	28
24	28
25	23
26	5
27	4
28	3
47	1

Table A-2. Weight enumerator

Weight	Number of Words
0	1
2	0
4	2927
6	167017
8	4916447
10	80897478
12	816454377
14	5338125069
16	23488120074
18	71384955784
20	152538959670
22	231779439554
24	251934559006
26	196121173412
28	108956243986
30	42831113394
32	11743967013
34	2198099208
36	272132459
38	21294613
40	981971
42	24070
44	245
46	1

Table A-3. Error statistics (EDC mode)

Weight	Detectable	Undetectable
1	47	0
2	1081	0
3	4507	11708
4	175438	2927
5	405976	1127963
6	10570556	167017
7	16712226	46179273
8	309541048	4916447
9	361932932	1000716213
10	5097169273	80897478
11	4626474407	12790659210
12	51434946474	816454377
13	37367194284	103309654161
14	336305649726	5338125069
15	199648256088	551968048461
16	1479744489024	23488120074
17	728127949008	2013060926406
18	4497263169906	71384955784
19	1852256859654	5120942911136
20	9609940719436	152538959670
21	3334060006144	9217699581278
22	14602118254672	231779439554
23	4282886436556	11840915404994
24	15871867282544	251934559006
25	3940252328376	10893645365850
26	12355638414010	196121173412
27	2593160205846	7169319473260
28	6864243526804	108956243986
29	1213546088136	3355102037554
30	2698357762020	42831113394
31	399296736984	1103935872114
32	739872337536	11743967013
33	90748896528	250894878267
34	138478749237	2198099208
35	13879342623	38372058228
36	17145001158	272132459
37	1375414408	3802652343
38	1341354532	21294613
39	83527138	230930357
40	61909528	981971
41	2852836	7884737
42	1509869	24070
43	47235	131130
44	15970	245
45	300	781
46	46	1
47	0	1

Table A-4. Error statistics (ED mode)

Weight	Detectable	Undetectable
1	47	0
2	1081	0
3	16215	0
4	175438	2927
5	1533939	0
6	10570556	167017
7	62891499	0
8	309541048	4916447
9	1362649145	0
10	5097169273	80897478
11	17417133617	0
12	51434946474	816454377
13	140676848445	0
14	336305649726	5338125069
15	751616304549	0
16	1479744489024	23488120074
17	2741188875414	0
18	4497263169906	71384955784
19	6973199770790	0
20	9609940719436	152538959670
21	12551759587422	0
22	14602118254672	231779439554
23	16123801841550	0
24	15871867282544	251934559006
25	14833897694226	0
26	12355638414010	196121173412
27	9762479679106	0
28	6864243526804	108956243986
29	4568648125690	0
30	2698357762020	42831113394
31	1503232609098	0
32	739872337536	11743967013
33	341643774795	0
34	138478749237	2198099208
35	52251400851	0
36	17145001158	272132459
37	5178066751	0
38	1341354532	21294613
39	314457495	0
40	61909528	981971
41	10737573	0
42	1509869	24070
43	178365	0
44	15970	245
45	1081	0
46	46	1
47	1	0

Appendix B

Undetectable Information-Bit Error Patterns

The only way for errors to escape detection by the *checksum byte* is to have an even number of errors in each bit slot. Since there are 5 information bytes per Data frame, there are 5 different bits per bit slot. We are only interested in the undetectable patterns with 2, 4, and 6 information bits in error.

I. Two Information Bit Errors

The only way to get an undetectable error pattern with two information bits in error is to have both bits in the same slot. That means that there are $\binom{5}{2}$ ways to do this in each slot and that there are $\binom{8}{1}$ slots to do it in. So:

$$\binom{5}{2} \times \binom{8}{1} = 80$$

There are 80 possible two information-bit error patterns that will escape detection. To find the codeword weights, a computer was used; each pattern was generated, encoded, and then "weighed" (the number of 1's counted). The results are:

37	weight 4
32	weight 6
11	weight 8

II. Four Information Bit Errors

There are two types of four information-bit error patterns that escape detection. The first is all four errors in one slot. There are $\binom{5}{4}$ ways per slot to do this and $\binom{8}{1}$ slots to do it in. Secondly, there could be two sets of slots with two errors in each. There are $\binom{5}{2}$ ways per slot and $\binom{8}{2}$ slots to do it in. Thus:

$$\binom{8}{1} \times \binom{5}{4} + \binom{8}{2} \times \binom{5}{2} \times \binom{5}{2} = 2840$$

So, there are 2840 possible four information bit patterns. Again a computer was used to generate the codeword weights, using the same method as before. The results are:

37	weight 4
892	weight 6
1591	weight 8
320	weight 10

III. Six Information Bit Errors

There are two types of six information-bit error patterns that escape detection. The first has three slots with two errors in each slot. There are $\binom{5}{2}$ ways per slot and $\binom{8}{3}$ slots to do it in. The second way has four errors in one slot and two errors in another. There are $\binom{5}{4}$ ways to do it in the four error slot and $\binom{5}{2}$ ways to do it in the two error slot; there are 8×7 slots to do it in. So:

$$8 \times 7 \times \binom{5}{4} \times \binom{5}{2} + \binom{8}{3} \times \binom{5}{2} \times \binom{5}{2} \times \binom{5}{2} = 58800$$

Thus, there are 58800 undetectable six information-bit error patterns. Using the same methods as above, the codeword weights were calculated. The results are:

924	weight 6
19253	weight 8
32244	weight 10
6379	weight 12

References

1. *Project Galileo Orbiter Functional Requirements Book, Vol. I*, Document GLL-3-290, Rev. B, "Command Structure and Assignment" (internal document), Jet Propulsion Laboratory, Pasadena, California.
2. McEliece, Robert J., *The Theory of Information and Coding*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1977.
3. Blahut, Richard E., *Theory and Practice of Error Control Codes*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1983.
4. *Telecommand, Part-1: Channel Service*, CCSDS Draft Recommendation, Red Book, Issue-1, Consultative Committee for Space Data Systems, Washington, D.C., April 1985.

A Wide-Band, High-Resolution Spectrum Analyzer

M. P. Quirk, H. C. Wilck, and M. J. Grimm
Communications Systems Research Section

This article describes a million-channel, 20 MHz-bandwidth, digital spectrum analyzer under development at JPL for use in the SETI Sky Survey and other applications in the Deep Space Network. The analyzer digitizes an analog input, performs a 2^{20} -point, Radix-2, Fast Fourier Transform, accumulates the output power, and normalizes the output to remove frequency-dependent gain. The effective speed of the real-time hardware is 2.2 GigaFLOPS.

I. System Overview

A. Spectrum Analyzer Description

This article describes a wide-band, high-resolution, digital spectrum analyzer design for use in the Search for Extraterrestrial Intelligence (SETI), spacecraft telemetry, radio astronomy, planetary radar, Radio Frequency Interference (RFI) monitoring, and frequency analysis of wide-band systems in the Jet Propulsion Laboratory's Deep Space Network. The primary radio astronomy user will be the Search for Extraterrestrial Intelligence (SETI). The analyzer samples an input signal with up to a 20 MHz IF bandwidth, performs contiguous real-time Fast Fourier Transforms (FFTs) of up to $1,048,576 = 2^{20}$ points corresponding to a resolution of 20 Hz. Special-purpose digital hardware provides real-time processing of the output spectra, including power accumulation, baseline rectification, matched filtering, and thresholding.

The output spectra can be recorded, displayed, and sent to a processing computer over a high-speed data bus. The system is microprocessor controlled, software reconfigurable, and includes built-in module and system-level diagnostics. The unit will be one seven-foot rack for the high-speed digital hard-

ware plus a half rack for the analog components, and a workstation computer for system control, spectrum displays, and post-processing.

The system contains two RF input channels, enabling dual-polarization analysis or the simultaneous analysis of two analog input channels.

The various components of the system are described below. Table 1 lists all the modules in the spectrum analyzer system.

B. System Block Diagram

Figure 1 is a functional block diagram of the spectrum analyzer system. The Intermediate Frequency (IF) signals come into the IF Input Signal Conditioner where they are mixed to baseband and sampled at the desired sampling frequency. The resultant digital samples are complex numbers with a total of 16 bits, 8 bits real and 8 bits imaginary.

The Input and Timing module collects sufficient data for a spectrum in one of the input buffers, implements the delay for the first stage of an N -point FFT, then sends the delay pairs to the window module. The Input and Timing module also gener-

ates all the synchronization signals for the spectrum analyzer system. The Window module applies a real, symmetric window to the data to broaden the main lobe of the frequency response and suppress the sidelobes. An FFT is then performed on the windowed data. Each FFT stage performs an FFT butterfly operation and then implements the delay for the next stage. It is also possible to transform two channels simultaneously, by putting one real channel into the real components and another real channel into the imaginary components of the complex FFT. The two channels are sorted out in the Real-Adjust board of the FFT processor. An unscrambler module bit-reverses the frequency bin indices of the FFT output so that the output data are in natural order.

The final FFT data then goes to the power accumulator which calculates the power of the complex spectrum and samples and accumulates them on a frequency-by-frequency basis. There are actually two independent accumulators so that two separate data streams can be accumulated. The accumulated power spectra are then baselined to remove frequency-dependent system gain. The baseline is derived from an exponentially updated accumulation for each frequency. Again two baseline accumulation memories are needed for the separate accumulations.

The final processing involves combining successive spectra using a five-coefficient filter which operates on a frequency-by-frequency basis. This spectrum filter can be a matched filter to any desired single-frequency time signature of duration equal to five accumulation intervals. For example, as a source transits across an antenna beam, its amplitude traces the shape of the antenna beam response as shown in Fig. 2. If the coefficients of the spectrum filter are chosen to match the beam shape, the filter acts as a matched filter for transit sources. The output of the spectrum which passes a threshold test is then sent to the Spectrum Processor.

An output processor is used to select data for display on the system graphics display. The data can be from the spectrum filter or from the baseline module. The output processor can be used for test output by bypassing all the modules between it and the module under test. The spectrum analyzer system is controlled by the microprocessor computer system whose primary function is data processing and graphics handling.

C. Configuration, Control and Post Processing

The system controller and post processor will be a microprocessor-based computer system with high-resolution graphics, and sufficient storage and processing power to control and configure the spectrum analyzer, and perform additional data processing on the output spectra. The computer system will generate graphics displays, archive data, provide antenna monitoring and control, and perform special-purpose processing.

Examples of special-purpose processing include signal identification, signal tracking, and the removal of undesired interfering signals.

The spectrum analyzer can be configured by a few dozen control bytes in the memory address space of the control microprocessor. Configuration parameters which can be changed include the sampling rate, whether the input is dual-channel or single-channel, real or complex, the length of the FFT, accumulation frame lengths, the baseline decay factor, the length of the spectrum filter, and the processing of the output data. The RAM coefficient memories for the window, matched filter, or initial baseline can be read in from the control microprocessor. Any module can be bypassed during operation or testing.

Each module of the analyzer is designed to support self-tests and diagnostics. Data vectors can be read into or out of any module for testing. The test output vectors are compared to the expected response to check the performance of the board. Since any module can be bypassed as the controller chooses, each module can be tested in isolation or with any combination of other modules. The executive board is controlled to generate test synchronization signals, and transmit and receive test vectors through the executive board buffers. The test software will reside in the control computer.

D. Hardware Description

The hardware design of the megachannel analyzer is based on design experience with previous instruments of a similar nature. The 65,536-channel spectrum analyzer (Refs. 2, 3) is used for radio spectroscopy, Radio Frequency Interference (RFI) surveillance, to aid the DSN in trouble-shooting, and for the SETI All-Sky Survey (Refs. 4, 5). Recently a digital convolver was developed to process Synthetic Aperture Radar (SAR) data. The SAR processor contains two forward FFTs and two inverse FFTs, each 16,384 points long.

The new megachannel analyzer prototype will use TTL logic on large 16-inch by 17-inch wirewrap boards with eight boards in a cage. The entire system will be housed in a single 7-foot rack. Extensive use is made of the TRW TDC1022 22-bit floating-point adders and AMD MPY016-type 16-bit multipliers. The machine processes 20 complex megasamples per second and performs more than 2.2 GigaFLOPS.

II. Module Descriptions

Table 1 lists all the modules in the spectrum analyzer system. Except for the Output Buffer and Interface Board, all the modules are described below.

A. Intermediate Frequency Input Signal Conditioner

The IF Input Signal Conditioner filters and samples the analog input data to provide digital samples for the spectrum analyzer. Figure 3 shows the block diagram for one of the two IF sections. The input is bandpass filtered to a 20 MHz bandwidth and 55 MHz center frequency. An attenuator is used to adjust the IF level. This signal is downconverted to baseband by the complex mixer. The 3 dB points of the inphase and quadrature low-pass filters are at 10 MHz. These filters are matched in gain and phase to provide image rejection of greater than 30 dB. The complex baseband signal is sampled at 20 MHz by the 8-bit Analog-to-Digital Converters (ADCs). Synthesizers generate the sampling frequency and center frequency reference tones.

Two IF Channels can be processed simultaneously by the spectrum analyzer, but the maximum bandwidth is 10 MHz, half the maximum for a single input. Each signal is applied to the input of a complex mixer and the ADCs sample one of the arms of each mixer. Thus one of the real sampled signals is in the real components and the other is in the imaginary components of the complex FFT input. Since the spectrum of the real component is conjugate symmetric and the spectrum of the imaginary component is conjugate antisymmetric, the two signals can be separated by forming the conjugate symmetric and antisymmetric components of the output spectra. This operation is performed in the Real-Adjust Module.

For sampling frequencies F_s other than 10 or 20 MHz bandwidth, filters with adjustable bandwidths are used after the matched low-pass filters. The current design does not allow for sampling rates greater than 20 MHz, but this could be implemented by adding a higher bandwidth IF section, faster ADCs, and a fast input buffer which would read into the current input buffers at 20 MHz. Contiguous spectra could not be processed at a rate higher than 20 MHz, so the duty cycle would be the ratio of the fast sampling rate to the 20 MHz processing rate.

B. Input and Timing

The Input and Timing module triple buffers the input data using three, 2-Mbyte RAM buffers, implements the delay operation for the first stage of the FFT, and generates all the synchronization signals for the spectrum analyzer. The synchronization signals, explained below, include the start-of-spectrum indicator, valid spectrum indicators, and frame triggers. The synchronization signals are sent down the processing pipe on two serial lines. Each module delays the synchronization signals by an amount equal to the module pipe delay before sending the signals to the next module in the pipe.

The current buffer is read out by the delay commutator which outputs pairs of complex data samples, $x(k)$ and $x(k + N/2)$, $0 \leq k \leq N/2 - 1$ which are to be combined in the first butterfly stage of an N -point FFT. The start-of-spectrum indicator coincides with $k = 0$ and occurs every N samples, or every $N/2$ sample pairs. Because of the periodicity of the delay commutator, the beginning of a new spectrum can occur only every N samples times. Thus, if the sampling rate is less than the 20-MHz processing rate, and the next buffer is not full when the commutator is reset to $k = 0$, N ticks of the 20-MHz processor clock must be skipped before the buffer can begin to be processed. The first two time lines in Fig. 4 show the write and read times for a 16 MHz sampling rate. Since the sampling rate is four-fifths the processing rate, every fifth spectrum does not contain valid data, and a valid spectrum indicator does not appear in the third time line. Since the processing pipe cannot be stopped to wait for the buffer to fill, the modules continue cycling through their processing, but the absence of the valid spectrum indicator prevents the data from being added into the accumulation, used to update the baseline, or entered into the spectrum filter.

The frame trigger signals control the lengths of the spectrum accumulations. The bottom time line in Fig. 4 is for accumulation frames of 3 spectra. Since two input channels can be processed simultaneously there are two separate accumulator memories. The valid spectrum indicator identifies which accumulation each spectrum belongs to, and there are two frame triggers, one for the α accumulator and one for the β accumulator so that they can be controlled independently. For dual-channel input, the Real-Adjust module interleaves a spectrum of the real input with a spectrum of the imaginary input. Thus the even-numbered spectra belong to the real channel and are accumulated separately from the odd spectra which belong to the imaginary channel.

Dicke switching also makes use of the two separate accumulators. In Dicke switching the antenna switches from looking at a source for K spectra, to looking at the noise for K spectra. The switching is driven in synchronization with the spectrum analyzer. A counter can be set to discard a specified number of samples to allow synchronization with a pulsing signal, or settling time for the Dicke switch.

C. Executive Board

The Executive board generates the clock signals for the high-speed digital components. It also contains a test input buffer and response buffer which are read only and write only respectively to the microprocessor bus. The buffers are in the memory address space of the microprocessor. Test data vectors are generated by the microprocessor and written to the test input buffer at microprocessor speed and then transferred to the Input and Timing module at the full hardware processing

speed. The Executive board generates the appropriate test synchronization signals as commanded by the microprocessor. Test output vectors are returned to the microprocessor via the Response Bus and Response Buffer. The Response buffer is a First-In-First-Out (FIFO) buffer. The data streams through until a programmed stop signal freezes the contents. The contents can then be read out and analyzed by the microprocessor at its slower processing rate.

Test vectors can be applied to any module or group of modules in the spectrum analyzer by bypassing all preceding boards in the pipe so that the test vectors are passed on unchanged to the input of the first board under test. Every module has the ability to send output on the Response Bus to the Response Buffer. The bypass data path is 46-bits wide to accommodate 44-bit data vectors plus two synchronization lines.

D. Window Module

The window module applies a real, symmetric window to the fixed-point, complex input pairs for each spectrum. The operation is a simple real times complex multiply:

$$y_m(n) = w(n)x_m(n) \quad 0 \leq n \leq N-1, 0 \leq m \quad (1)$$

The subscript will be used to index successive spectra, or equivalently, segments of N samples. Thus $x_m(n)$ refers to the $(mN + n)$ th time sample.

Since a window is a point-by-point multiplication in frequency, by the convolution theorem of the Discrete Fourier Transform (DFT) (Ref. 1; p. 59) the DFT of a windowed signal is the convolution of the spectrum of the unwindowed signal with the frequency response of the window. Thus the window $\{w(n)\}$ shapes the frequency response of the FFT. If no window is applied, performing an FFT on a finite data segment of length N is equivalent to using a rectangular window. The frequency response of the rectangular window is shown in Fig. 5. Since the FFT samples the spectrum at frequency points kF_s/N , $0 \leq k \leq N-1$, the FFT of a frequency component depends on how close it is to a multiple of F_s/N . The FFT for a frequency which is an exact multiple of F_s/N is shown in Fig. 6(a). Figure 6(b) shows the FFT of a frequency exactly half way between two sample points. For Fig. 6(b) the peak magnitude is 3.9 dB less than for Fig. 6(a), and the first sidelobe at -13.5 dB is only 9.5 dB down from the peak. The frequency response can be improved by using a window like the Hanning window shown in Fig. 7. This window broadens the main lobe and suppresses the sidelobes. A frequency halfway between the FFT sample point would have a peak magnitude

only 1.4 dB down from the peak, and the first sidelobe is 31.5 dB down from the peak. Frequency responses of many different windows are given in Ref. 1.

The hardware for this module consists of an arithmetic unit for the window multiply, and a RAM coefficient memory which is loaded in from the control microprocessor during system setup. The window length is the same as the FFT length, and can be any power of two from $2^4 = 16$ to $2^{20} = 1,048,576$.

E. FFT Module

The FFT module transforms the time domain signal into the frequency domain by means of an FFT. The FFT of the input signal is:

$$Y_m(k) = \frac{1}{N} \sum_{n=0}^{N-1} y_m(n) \exp(-j2\pi nk/N), \quad 0 \leq k \leq N-1 \quad (2)$$

where $\{y_m(n)\}$ is the output of the window module. The length of the FFT, N , can be any power of two from $2^4 = 16$ to $2^{20} = 1,048,576$.

The FFT is implemented using the Radix-2, Decimation-In-Frequency (DIF), pipelined FFT algorithm. This algorithm is derived in detail in Ref. 1 (pp. 368-371). The Radix-2 form was chosen because it has the least complexity for addressing data and coefficients. The post-multiply butterfly of the DIF algorithm was preferred by the hardware designers.

In the Radix-2, DIF algorithm the FFT is computed in $M = \log_2 N$ stages. Each stage consists of a "butterfly" operation on $N/2$ pairs of complex points. The butterfly, shown in Fig. 8, takes the sum and difference of the complex pairs and then multiplies the difference by the appropriate "twiddle" factor. The twiddle factors are complex roots of unity $W_N^{nk} = \exp(-j2\pi nk/N)$. At the i th stage, $1 \leq i \leq M$, the pairs for each butterfly operation are the results from the previous stage taken 2^{M-i} apart, $z_{i-1}(n)$ and $z_{i-1}(n + 2^{M-i})$, and the twiddle factors are addressed by $n2^{i-1}$. The butterfly operation is:

$$\left. \begin{aligned} z_i(n) &= \{z_{i-1}(n) + z_{i-1}(n + 2^{M-i})\} \\ 0 &\leq n \leq N-1 \\ z_i(n + 2^{M-i}) &= \{z_{i-1}(n) - z_{i-1}(n + 2^{M-i})\} W_N^{n2^{i-1}} \\ 1 &\leq i \leq M \end{aligned} \right\} \quad (3)$$

with

$$z_0(n) = y_m(n)$$

and

$$Y_m(U(n)) = z_M(n)$$

where $U(n)$ is the bit reversal of n . The bit reversal is performed in the Unscrambler.

Each FFT board, except the final board, contains two butterfly stages of a pipe-lined, Radix-2 FFT. The final board contains four butterfly stages. Each stage consists of an arithmetic unit (AU) and a PROM coefficient memory followed by a RAM delay memory with a cross switch. The cross switch is programmed to choose the correct data pairs for the next stage. The first six stages of the FFT are implemented using fixed-point arithmetic and data. The final fourteen stages use floating-point arithmetic and data. The last four stages are implemented on one board. There are no delay memories since only pipelined delays are used. In addition, the last two stages require no complex multiplies. The third card contains Response Bus connection for the first cage. The FFT module can be configured to produce any power-of-2, complex FFT from a 16-point up to a 1,048,576-point (2^{20}).

F. FFT Unscrambler

The FFT Unscrambler puts the complex, floating-point FFT data stream in natural order. The ordering involves bit reversing the addresses such that the input data with address:

$$n = n_0 + 2n_1 + 2^2n_2 + \dots + 2^kn_k + \dots + 2^{M-2}n_{M-2} + 2^{M-1}n_{M-1} \quad (4)$$

becomes the output address:

$$U(n) = n_{M-1} + 2n_{M-2} + 2^2n_{M-3} + \dots + 2^kn_{M-k-1} + \dots + 2^{M-2}n_1 + 2^{M-1}n_0$$

The bit reversal is performed by writing the data into a double-buffered memory in order n , and reading them out in order $U(n)$. Two parallel output lines concurrently carry the upper and lower halves of the spectrum in sequential order. The module consists of two 1,048,576 by 44-bit memories and a crossbar switch. It also contains the control bus and response bus interfaces for the FFT cage. The board can unscramble any transform length from 16 to $1,048,576 = 2^{20}$.

G. FFT Real-Adjust Stage

The FFT Real-Adjust stage is used when the input is one or two real channels, $a_m(n)$ and $b_m(n)$, $0 \leq n \leq N-1$, instead

of the normal input of a single complex channel. Two real channels can be transformed simultaneously if one channel is put into the real components and another channel into the imaginary components of the complex FFT input:

$$x_m(n) = a_m(n) + j b_m(n), \quad 0 \leq n \leq N-1 \quad (5)$$

Since the spectrum of a real signal is conjugate symmetric and the spectrum of an imaginary signal is conjugate antisymmetric, the two channels can be separated by forming the conjugate symmetric and antisymmetric components:

$$A_{m(k)} = \frac{1}{2} \{Y_m(k) + Y_m^*(N-k)\}, \quad 1 \leq k \leq N-1 \quad (6)$$

and

$$B_{m(k)} = -\frac{j}{2} \{Y_m(k) - Y_m^*(N-k)\}, \quad 1 \leq k \leq N-1$$

with

$$A_m(0) = \text{Re} \{Y_m(0)\}$$

$$B_m(0) = \text{Im} \{Y_m(0)\}$$

H. Power Accumulator

The power accumulator module computes the power of each of the complex points output by the FFT and accumulates the power values into one of the channel accumulation memories. The output, $P_k(n)$, is

$$P_k(n) = \sum_{m=L(k-1)}^{Lk-1} |Y_m(n)|^2, \quad k \geq 0 \quad (7)$$

where L is the length of the accumulation. The average power in each spectrum is computed and accumulated in place of the zero-frequency component. The two accumulation memories, A and B , are controlled independently, and can be used to accumulate two different channels, or to switch the accumulators as in Dicke switching. Every valid spectrum entering the Power Accumulator module is identified as a type A or B by the frame synchronization indicator. The maximum accumulation length is 2^{16} , and is controlled by frame trigger synchronization signals generated by the Input and Control module.

The power is computed in floating-point arithmetic and then converted to fixed-point in order to provide sufficient precision during the accumulation. The accumulations are performed with fixed-point arithmetic and then converted to floating-point before being output to the baseline module. The

module consists of two arithmetic units, one for the power calculation and one for the accumulation operation, two barrel shifters to fix the input and float the output, and a RAM memory for the 60-bit wide, 1,048,576-point accumulated spectra. The power accumulator board also provides the multi-bus connection for the signal processing cage. The input and output are 22-bit floating-point numbers.

I. Baseline Module

The baseline module removes the frequency-dependent system gain by multiplying each spectrum channel by the inverse of a baseline value. The floating-point baseline can be loaded in by the system controller during system setup, or computed by inverting a baseline formed from the input channels. The baseline is formed by applying an exponentially decaying update filter to the frequency components of the power accumulator. Components greater than a specified threshold value are not used to update the baseline. A local spectrum average is used instead. The baseline algorithm is:

$$S_k(n) = P_k(n)/B_{k-1}(n) \quad (8)$$

$$B_k(n) = \begin{cases} (1 - \alpha)B_{k-1}(n) + \alpha P_k(n) & \text{if } S_k(n) \leq T \\ (1 - \alpha)B_{k-1}(n) + \alpha \bar{P}_k(n-1) & \text{if } S_k(n) > T \end{cases} \quad (9)$$

where $\bar{P}_k(n)$ is a local average of the spectrum values formed by another exponentially updated filter:

$$\bar{P}_k(n) = (1 - \beta)\bar{P}_k(n-1) + \beta P_k(n), \quad \text{if } S_k(n) \leq T \quad (10)$$

If the baseline spectrum value $S_k(n)$ is above the threshold, the spectrum average is not updated. Since there can be two distinct sequences of spectra from the Power Accumulator, there are two baseline memories to accommodate them.

The inversion of the floating-point baseline is performed by table lookup on the mantissa values, and negation of the exponents. The baseline module consists of an arithmetic unit (AU) to implement the inverse baseline multiply and update filter, a floating-point table-lookup inverter, and RAM memories for the baseline.

J. Spectrum Filter Module

The Spectrum Filter module contains a five-coefficient Finite Impulse Response filter which filters each baselined frequency component across successive spectra. The spectrum filter can be a matched filter to any desired single-frequency

time signature of duration less than or equal to five accumulation intervals. The example in Fig. 2 was for a source transiting an antenna beam. Another example is needed. The filtering operation is:

$$F_k(n) = \sum_{i=0}^4 \alpha_i S_{k-i}(n) \quad (11)$$

where the α_i , $0 \leq i \leq 4$ are input parameters. To implement a filter of length less than 5, simply set the end coefficients to zero.

The filtered output which passes a threshold test is sent to the Spectrum Processor via a high-speed data bus for additional processing. The output can also be sent to the Output processor. A bit mask suppresses specified components so that they do not pass the threshold test. The mask is input from the control computer. The spectrum filtering operation requires an arithmetic unit and four 1,048,576 by 22-bit memories.

K. Output Processor

The Output Processor controls the amount and form of the spectrum analyzer output. A full buffer of data is collected and then sent to the system Response Buffer on the Executive board for transfer to the processing computer via a microprocessor bus. The output data is used for the spectrum displays and can also be used for further processing. The output data can be a snapshot of a full spectrum, or a selected window of data from successive spectra. The Output Processor can divide a spectrum into sections and choose the k th value, select the peak value, or compute the average value for each section. The Output Processor consists of an arithmetic unit and a 1,048,576 by 50-bit buffer memory. It also contains the Control Bus and Response Bus connections for the signal processing cage.

III. Scaling

The 22-bit floating-point format represents numbers with exponents from -32 to 31, and mantissas from -1 to $1-2^{-15}$. Numbers between $(-2^{-33} + 2^{-47})$ and 2^{-33} will all be represented as zero. Negative overflow occurs for numbers less than -2^{31} , and positive overflow occurs for numbers greater than $2^{31} - 2^{16}$. Scaling is used to keep the numbers computed in the spectrum analyzer within the dynamic range of the floating-point format.

To determine the range of numbers the analyzer will encounter, we consider two different types of input signals, a pure sinusoid, and Gaussian white noise.

The samples of the complex sinusoid with amplitude A , initial phase ϕ , and frequency f are:

$$x(n) = A \exp(j\phi) \exp(j2\pi fn/F_s) \quad (12)$$

where F_s is the sampling frequency. The FFT of this signal, as defined in Eq. (2), using a rectangular window is:

$$Y(k) = \frac{A \sin \pi N f / F_s}{N \sin \pi (f / F_s - k / N)} \quad (13)$$

with power A^2 at $k/N = f/F_s$ and less than A^2 elsewhere. If we accumulate over L spectra, the maximum power will be LA^2 .

For complex Gaussian white noise, the real and imaginary components are independently, identically distributed Normal ($0, \sigma^2$):

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp[-x^2/(2\sigma^2)] \quad (14)$$

The frequency bins of the FFT from Eq. (2) will be identically distributed Normal ($0, \sigma^2/N$). Taking the sum of the squares

of the imaginary and real components, and accumulating L of these power values results in a chi-square distribution of channels with $2L$ degrees of freedom:

$$p(x) = \frac{1}{2^L (\sigma^2/N)^L \Gamma(L)} x^{L-1} \exp[-xN/(2\sigma^2)] \quad (15)$$

The expected value of this distribution is $2L\sigma^2/N$ with variance $4L(L+1)\sigma^2/N$.

Since the spectrum analyzer input consists of 8-bit two's complement numbers, the range for both A and σ is 2^{-7} to 1. As seen from the above discussion, the FFT output range is much larger than the input range, and the power of a sinusoidal signal in a frequency bin is approximately N times as large as the expected power of Gaussian white noise. In order to keep the accumulated output from overflowing the floating-point range, and at the same time keep the noise power from underflowing, we provide a scale factor in the form of an exponent bias applied at the input to the first floating-point FFT stage. The bias exponent, B , will be based on the length of the FFT, N , and the number of accumulations, L , in order to retain as much precision as possible while avoiding overflow.

References

1. Rabiner, L. R., and Gold, B., *Theory and Application of Digital Signal Processing*, Prentice-Hall, Inc., Englewood Cliffs, N.J., 1975.
2. Emerson, R. F., "Biplex Pipelined FFT," *The Deep Space Network Progress Report 42-34*, pp. 54-59, Jet Propulsion Laboratory, Pasadena, Calif., 1976.
3. Morris, G. A., and Wilck, H. C., "JPL 2²⁰ Channel 300 MHz Bandwidth Digital Spectrum Analyzer," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Tulsa, Oklahoma, pp. 808-811, 1978.
4. Olsen, E. T., Lokshin, A., and Gulkis, S., "An Analysis of the Elements of an All Sky Survey," *Proc. IAU Symposium No. 112*, Boston, MA, June 1984 (in press).
5. Solomon, J., Quirk, M. P., Olsen, E. T., and Lawton, W., "A Signal Detection Strategy for The SETI All Sky Survey," *The Telecommunications and Data Acquisition Progress Report 42-83*, Jet Propulsion Laboratory, Pasadena, Calif., 1985.

Table 1. Spectrum analyzer modules

Module	Quantity
IF Signal Input	2
Executive Board	1
Input and Timing	2
Window	1
Fixed-Point FFT	3
Floating-Point FFT	5
4-stage FFT	1
Real-Adjust Board	1
Unscrambler	1
Power Accumulator	2
Baseline	2
Spectrum Filter	2
Output Processor	2
Output Buffer	1
Interface Board	1
Power Supplies	2
Cages	5
Rack	1
Microprocessor System and Software	1

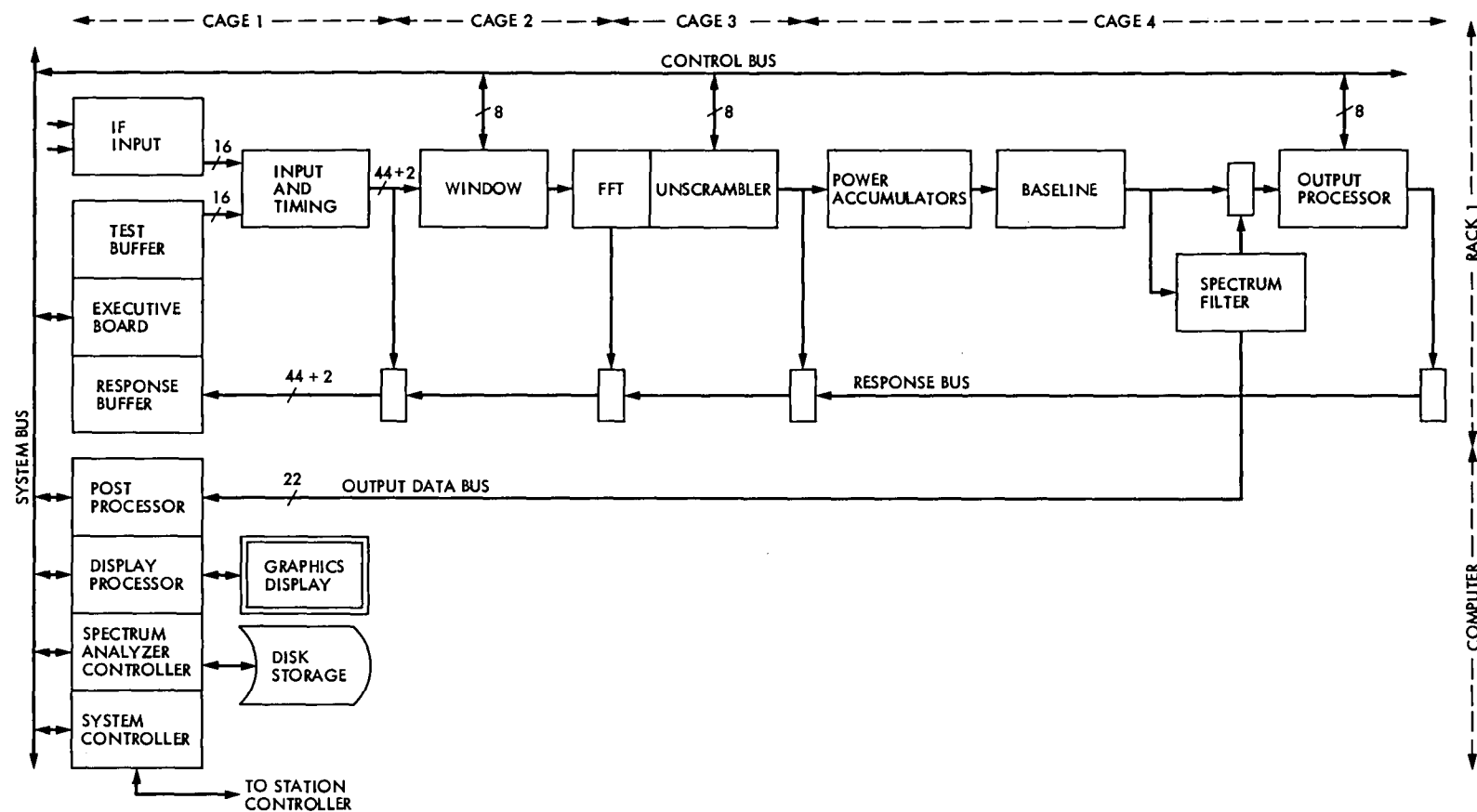


Fig. 1. Spectrum analyzer system

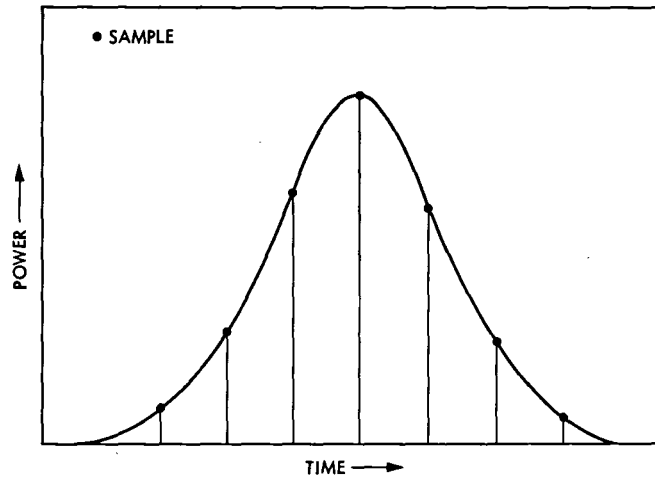


Fig. 2. Frequency response of a source transiting an antenna beam

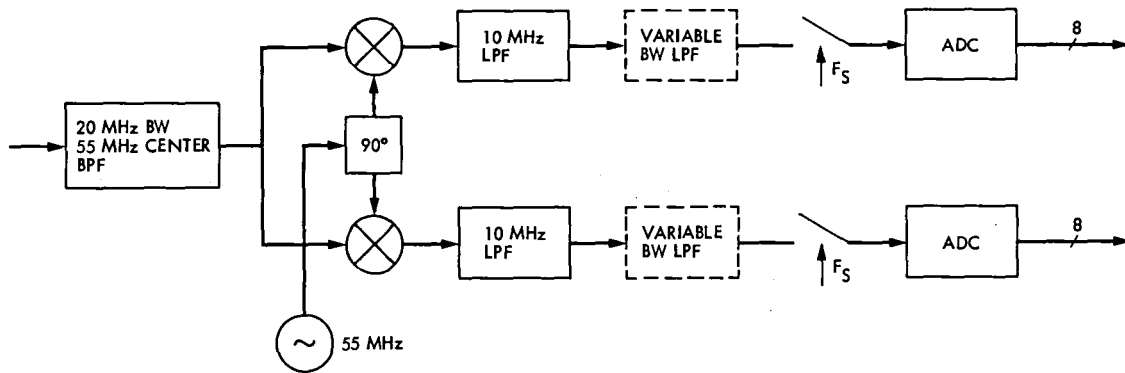


Fig. 3. Intermediate frequency signal conditioner

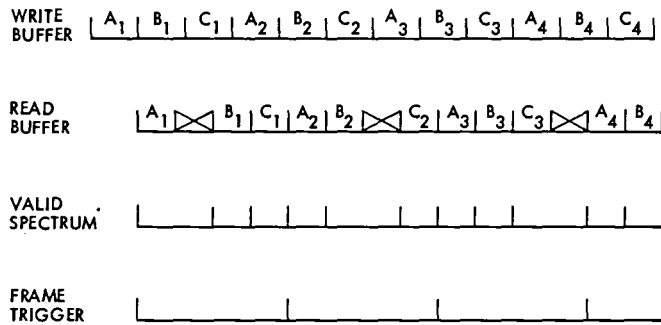


Fig. 4. Timing diagram

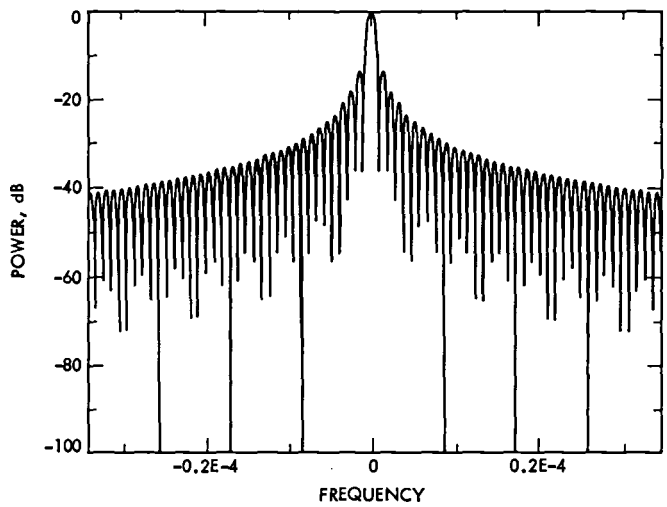


Fig. 5. Rectangular window response

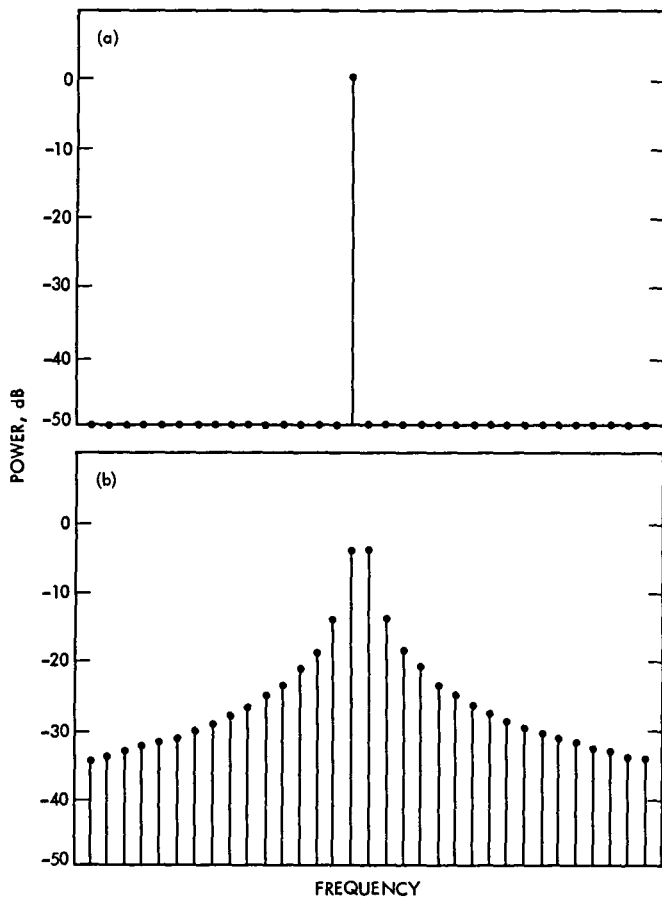


Fig. 6. Frequency responses: (a) a frequency halfway between two FFT frequencies; (b) a signal with frequency coinciding with an FFT frequency

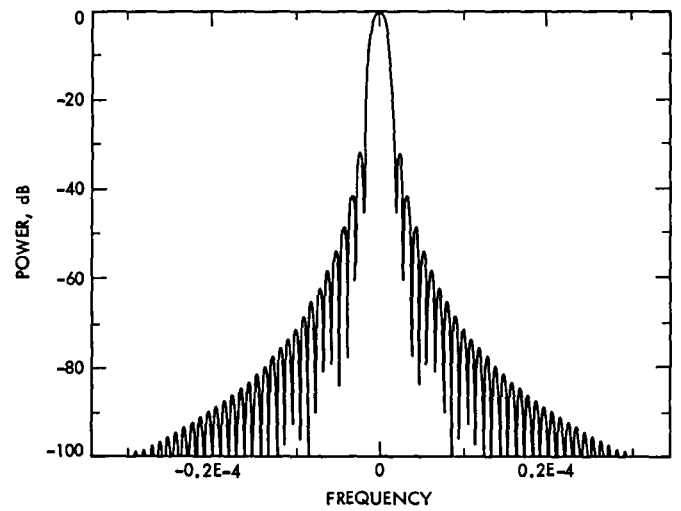
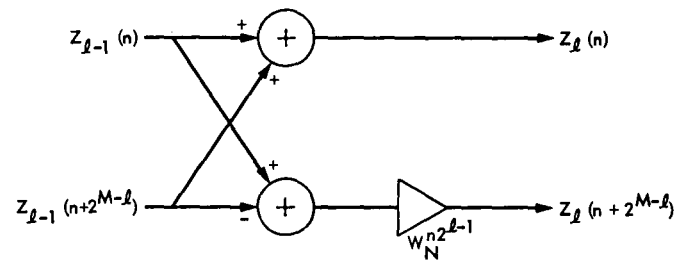


Fig. 7. Hanning window response



NOTE: ADDITIONS AND MULTIPLICATIONS ARE COMPLEX

Fig. 8. Decimation-in-frequency butterfly stage

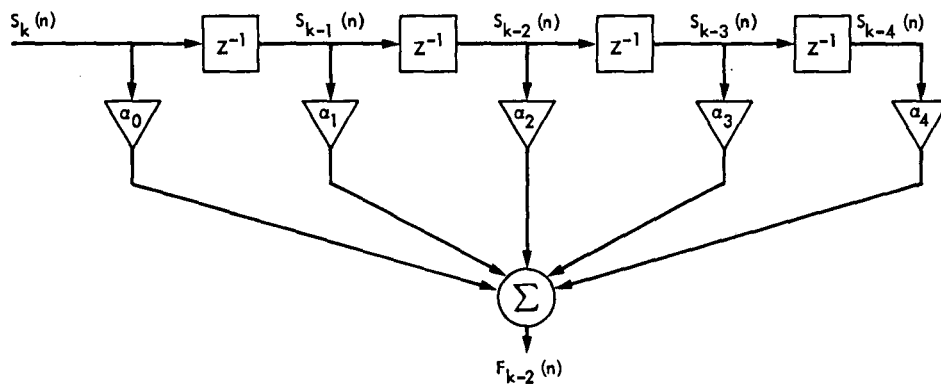


Fig. 9. The 5-stage finite impulse response filter

A Signal Detection Strategy for the SETI All Sky Survey

J. Solomon and W. Lawton

Image Processing Applications and Development Section

M. P. Quirk

Communications Systems Research Section

E. T. Olsen

Space Physics and Astrophysics Section

This article describes a source detection strategy for the SETI All Sky Survey. The method is designed to detect continuous wave (or very narrowband) sources transiting an antenna beam. The short-time spectra of the received signal are accumulated, and candidate extraterrestrial sources are recognized by the presence of narrowband power exceeding a threshold function. The threshold function is derived using a Neyman-Pearson hypothesis test.

I. Introduction

The NASA program for the Search for Extraterrestrial Intelligence (SETI) is currently developing prototype signal processing hardware and software, search strategies, and observing procedures for an ambitious search to be conducted in the early 1990's. The long range NASA SETI plan calls for an All Sky Survey covering the 1-10 GHz frequency range and a Targeted Search covering the 1-3 GHz frequency range. In the All Sky Survey, all 4π steradians of the celestial sphere will be surveyed with approximately uniform sensitivity for the presence of narrowband signals of extraterrestrial origin. In the Targeted Search, a thousand candidate sources will be observed, including nearby solar-type stars and interesting sources identified by the All Sky Survey.

The JPL SETI team has the responsibility for designing, developing, and implementing the All Sky Survey. To detect

ETI signals, which are assumed to be narrowband, we divide the input bandwidth into narrowband channels using a Fast Fourier Transform (FFT). The FFT output power in each channel is accumulated and convolved with a filter matched to the antenna beam shape. A thresholding operation is performed on the matched-filter output to locate channels with excess power. These events are stored for a confirmation test on the return scan. The processing up to and including the matched filter is performed in a high-speed digital spectrum analyzer such as that described in Ref. 1. A digital computer system will detect ETI signals from the spectrum analyzer output.

This report presents the current design for the signal detection strategy. In the next section we describe the All Sky Survey. Section III describes the signal characteristics, and in Section IV we derive the performance characteristics for a Neyman-Pearson detection system applied to these signals.

II. The SETI All Sky Survey

The characteristics of the All Sky Survey are shown in Table 1. The basic premise of the Sky Survey is that we do not know where to look. There are over a million solar-type stars within a thousand light years of the earth, and their density per unit steradian is approximately uniform. Since we also do not know how strong a signal might be, there is no preferred direction within this volume of stars.

To satisfy the first three characteristics, the antenna beam must be swept across the sky at 20 to 50 times the sidereal rate. The sensitivity of the survey is thereby established (within an order of magnitude), and the class of signal types is restricted to Continuous Wave (CW) and narrowband signals with bandwidths from 10 to 30 Hz.

The requirement of uniform sensitivity affects the signal detection strategy and the scan strategy. The specified variation of sensitivity with frequency results from a compromise among frequency coverage, survey duration, and sensitivity. It assumes that the angular rate at which the main beam of the antenna is swept across the celestial sphere is independent of the survey frequency. Because the beam area is inversely proportional to frequency squared, the duration of the survey is approximately proportional to the square of the highest frequency observed. If it were required that the sensitivity be independent of frequency, the scan rate would increase with frequency, and the duration of the survey would then be proportional to the cube of the highest frequency observed.

A survey of the sky to a uniform limiting flux at any given frequency requires that the main beam of the antenna be swept across the celestial sphere at a rate that varies inversely as the square of the system temperature. The scan strategy is therefore designed to minimize the system temperature and its fluctuations, caused, for example, by changing sidelobe pickup and sky temperature variations. It must also utilize the antenna time efficiently, minimizing the time spent accelerating and decelerating and the time lost waiting for mechanical oscillations to damp out. Since SETI will not be the sole user of the antenna, it is necessary to subdivide the celestial sphere into elements which can be observed over the course of one to four hours and which are easily incorporated into a complete sky map with minimum overlap. Finally, the scan strategy must satisfy the requirements of the signal detector for signal confirmation and RFI rejection. In order to satisfy these conflicting requirements, we choose a reasonable survey time and design our strategy to achieve maximum sensitivity and uniformity. As shown below, our strategy requires that the scans be of moderate length and that successive scans be adjacent to one another.

Figure 1 is a schematic representation of one scan strategy which can satisfy these requirements using an alt-azimuth antenna. The pixels visible to the observatory are mapped by scanning the beam along lines of constant declination near transit. The maximum rate at which the antenna may be driven places an upper bound to the declination range for which this strategy may be employed and still yield a uniformly sensitive survey, since the required azimuth rate for this strategy varies inversely with the cosine of the declination. A different scan strategy must be employed at higher declinations. In general, a pixel will be mapped using a boustrophedonic scan pattern designed to optimize the system temperature by minimizing changes in elevation. A representative three-scan pattern is shown in the center of Fig. 1.

Although we desire to minimize the time lost reversing the direction of motion, care must be taken to avoid exciting the normal modes of oscillation of the mechanical system. Antenna dynamics and servo control time constants determine the duration of the turnaround period at the end of each scan. The scan length must not be so short that a large part of the time is wasted on the antenna turnaround.

III. Signal Characteristics

If the number of independent samples included in the accumulation is large, the limiting sensitivity to a CW source achieved by accumulating over time τ as the antenna beam is swept through its position is:

$$\Phi(t, \tau) = \frac{4\alpha kT}{\pi\eta D^2} \sqrt{b/\tau} \frac{1}{\frac{1}{\tau} \int_t^{t+\tau} G(\psi) d\psi} \text{ W/m}^2 \quad (1)$$

where α is the signal to noise ratio (SNR) for detection, T is the system temperature (K), k is Boltzmann's constant (Watt-sec/K), η is the aperture efficiency, D is the diameter of the circular equivalent aperture (meters), and b is the spectral resolution (Hz). The effective gain during the integration time is represented by the dimensionless function, $G(\psi)$, which is defined here to be unity when the source is on the beam axis. If the antenna beam tracks the source position rather than scanning through it, the integral becomes τ and the limiting sensitivity decreases as $1/\sqrt{\tau}$.

The time required to move the antenna beam through one half power beam width (HPBW) is approximately:

$$t_B = \frac{70c}{\nu D \omega} \quad (2)$$

in seconds where c is the speed of light (meter/sec), ω is rate in degrees/second at which the beam is swept across the sky (deg/sec), and ν is the frequency at which the survey is being carried out (Hz). The length of time that the source will be in the beam is approximately equal to $2t_B$, but the effective gain will vary greatly over that time interval.

Figure 2(a) shows the expected response within a frequency bin as the antenna beam is swept across the source position. For simplicity only a one dimensional slice of the circularly symmetric beam is shown. If the frequency response is measured as a source transits the beam, then the matched-filter detector would convolve the response with the beam shape shown. The ideal matched-filter would be many HPBW's long in order to use all the power from the source. It is desirable to find a practical filter length on the order of a HPBW. Figure 2(b) shows the SNR loss versus filter length in fractions of a HPBW. For lengths greater than 0.9 HPBW, the loss is less than 0.25 dB.

To keep the processing hardware cost-affordable, we accumulate spectra until there are only 3 to 5 spectra per HPBW. Figure 3(a) illustrates this coarser resolution and shows that the best SNR is realized when a source transits the center of the beam at the center of an accumulation interval. The worst case is when the maximum occurs at the transition between two accumulations. The uneven response due to this difference in source position is called scalloping. Figure 3(b) shows the SNR loss between the worst case and the best case for filter lengths of 1 HPBW and 0.6 HPBW. In both cases scalloping decreases as the number of accumulations increases.

IV. The Signal Detection Strategy

The purpose of this section is to provide the background and theoretical basis for SETI sky survey signal detector performance evaluations. The problem is formulated as a classical binary decision problem of detecting a signal with random parameters in noise (Refs. 2, 3). Our treatment will rely, where applicable, on results to be found in the radar signal detection literature (Refs. 4, 5).

Figure 4 illustrates the basic front-end processing which is to be performed on the received signal. The ETI signal is assumed to be narrowband Continuous Wave (CW) of the form:

$$s(t) = A \cos(2\pi f_s t + \phi) \quad (3)$$

and the spectrum analyzer is assumed to have a resolution of 20 Hz. The detection system is designed to test for the two possibilities:

$$\begin{aligned} r(t) &= A \cos(2\pi f_s t + \phi), \quad \{H_1\} \\ r(t) &= n(t), \quad \{H_0\} \end{aligned} \quad (3a)$$

where $n(t)$ is a zero-mean Gaussian noise process whose variance, σ_0^2 , is determined by the effective system temperature. Denoting by $x_j(f_\alpha)$ the j th sample of the α th frequency bin, the accumulator output is given by:

$$\xi_j(f_\alpha) = \sum_{n=0}^{N-1} |x_{Nj-n}(f_\alpha)|^2 \quad (4)$$

The probability density function (PDF) of the test statistic ξ under the null hypothesis $\{H_0\}$ is the gamma distribution with $2N$ degrees of freedom and $E\{\xi\} = 2N$, $\sigma_\xi^2 = 4N\sigma_0^2$. The PDF of ξ under the alternative hypothesis $\{H_1\}$ is the non-central chi-square distribution. The parameters of the PDF need to be explicitly related to the SETI observation parameters. In the receiver input with signal present, Eq. (3), if A were constant, then the ξ_j would have the form:

$$q' = \sum_{i=1}^N (A + x'_i)^2$$

where the x'_i are identically distributed Gaussian random variables with zero mean and variance σ_0^2 (the system noise). However, under the conditions of the survey, the antenna scans at a relatively rapid rate and thus during the time required to accumulate N samples ($T = N\tau$, where τ is the analyzer sample time), the received signal is modulated by the antenna beam pattern. We thus rewrite Eq. (3) as:

$$r(t) = A(t) \cos(2\pi f_s t + \phi) + n(t), \quad (5)$$

and note that the actual form of the ξ_i is now:

$$Q' = \sum_{i=1}^N (A_i + x'_i)^2$$

Thus, from the results of Whalen (Ref. 2), the expression for $p_0(\cdot)$ is the chi-square distribution with $2N$ degrees of freedom:

$$p_0(\xi) = \frac{1}{(\sigma_0^2)^N 2^N \Gamma(N)} \xi^{N-1} e^{-\xi/(2\sigma_0^2)} \quad (6)$$

and for $p_1(\cdot)$ is:

$$p_1(\xi) = \frac{1}{2\sigma_0^2} \left(\frac{\xi}{\lambda} \right)^{(N-1)/2} \exp \left[-(\xi + \lambda)/2\sigma_0^2 \right] I_{N-1}(\sqrt{\lambda\xi}/\sigma_0^2) \quad (7)$$

with $2N$ degrees of freedom, where:

$$\lambda = 2 \sum_{i=1}^N A_i^2$$

In Eq. (7), $I_N(\cdot)$ is the modified Bessel Function of the first kind and order N . These distributions are discussed in detail in Refs. 2 and 6.

We are now in a position to examine explicitly the effects of antenna scanning on $p_1(\cdot)$. The antenna scan rate is ω rad/sec, and the main lobe half-power beam width (HPBW) is Ω rad. The scan rate in HPBW/sec is:

$$\nu \equiv \omega/\Omega \quad (8)$$

If S_0 represents the ETI signal flux at the surface of the earth in W/m^2 , then

$$A(t) = \left(\frac{\pi D^2}{4} \epsilon S_0 \right)^2 \exp \left(-\frac{1}{2} [\nu^2 (t - t_0)^2 + (\xi_2^0)^2/\Omega^2] \right) \quad (9)$$

where D is the antenna diameter in meters, and ϵ is the antenna efficiency. A Gaussian main lobe shape has been assumed, and the coordinate system chosen, (ξ_1, ξ_2) , is centered on the moving bore-sight axis of the antenna, and (ξ_1^0, ξ_2^0) denotes the actual source position. Thus, our expression for the A_i , which determines the non-central parameter, λ , is:

$$A_i^2 = \frac{\pi D^2}{4} \epsilon S_0 \times \frac{1}{\tau} \int_{t_i - \tau/2}^{t_i + \tau/2} \exp \left(-[\nu^2 (t - t_0)^2 + (\xi_2^0)^2/\Omega^2] \right) dt \quad (10)$$

This expression is rather cumbersome, so we make the simplifying assumption that $\tau \ll 1/\nu$, i.e., point-wise sampling, and rewrite Eq. (10) as:

$$A_i^2 = \frac{\pi D^2}{4} \epsilon S_0 \exp \left(-[\nu^2 (t - t_0)^2 + (\xi_2^0)^2/\Omega^2] \right) \quad (11)$$

Note that the A_i have units of energy, so that the signal-to-noise ratio (SNR) for a single sample observation is:

$$\text{SNR}_i = \frac{A_i^2}{\sigma_0^2} \quad (12)$$

If we re-examine the expression for ξ_i , Eq. (4), we see that the effect of antenna scanning is to produce weighted accumulation of individual samples, where the weight function, w_i , is just:

$$w_i = \exp \{ -[\nu^2 (t_i - t_0) + (\xi_2^0)^2/\Omega^2] \} \quad (13)$$

From the above discussion, and our expression for w_i , it is apparent that for fixed (τ, Ω, N) increasing the antenna scan rate above some value, ω_{\max} , reduces the accumulator performance to no better than single sample detection. In fact, an upper limit on scan rate may be set by choosing some minimum weight for the N th sample as the antenna scans through a source. To illustrate the point, consider the case where the source position is (ξ_1^0, ξ_2^0) , N is odd, and the center sample is taken with the source on beam-axis, i.e., $n = (N+1)/2$ at ξ_1 . The weight function is rewritten with time in units of $n\tau$ as:

$$w_n = \exp \{ -\nu^2 \tau^2 (n - (N+1)/2)^2 \}, \quad n = 1, 2, \dots, N \quad (14)$$

Letting β denote the minimum desired weight, from Eq. (14) we find

$$\omega_{\max} = \frac{2\Omega}{\tau N} [\ln(1/\beta)]^{1/2} \quad (15)$$

or

$$\omega_{\max} = \frac{0.6}{f_0 D \pi N} [\ln(1/\beta)]^{1/2} \quad (16)$$

where f_0 is the system operating frequency in GHz. Thus, for $D = 34$ meters, $f_0 = 1$ GHz, $\tau = 50$ ms, and $N = 8$, if we require $\beta = 0.5$, then $\omega_{\max} = 10$ mrad/sec. On the other hand, at the upper end of the SETI survey frequency range, $f_0 = 10$ GHz, we have $\omega_{\max} = 1.0$ mrad/sec which is significantly less than scan rate capabilities of the antenna system. Since τ is fixed and ω_{\max} is only very weakly dependent on β , the dominating factor is our choice of N . For the time being we will assume that ω and N may be suitably chosen so as to yield approximately uniform accumulator weighting; thus:

$$A^2 = \frac{\pi D^2}{4} \epsilon S_0 \quad (17)$$

and the non-central parameter appropriate for $p_1(\cdot)$, Eq. (7) is given by:

$$\lambda = 2NA^2 \quad (18)$$

We conclude our discussion of the statistics of ξ_i by casting the results obtained above in a form more amenable to computations described in Section IV. To accomplish this we introduce the "normalized" test statistics

$$q_0 = \sum_{i=1}^N \left(\frac{x'_i}{\sigma_0} \right)^2 \quad (19)$$

$$q_1 = \sum_{i=1}^N \left(\frac{A}{\sigma_0} + \frac{x'_i}{\sigma_0} \right)^2 \quad (20)$$

where the random variable (x'_i/σ_0) is now zero-mean Gaussian with unit variance. The resulting PDFs under H_0 and H_1 are now given by (Ref. 2)

$$p_0(q) = \frac{1}{2^N \Gamma(N)} q^{N-1} \exp(-q/2) \quad (21)$$

with $2N$ degrees of freedom, $E\{q\} = 2N$, $\text{Var}\{q\} = 4N$; and

$$p_1(q) = \frac{1}{2} \left(\frac{q}{\lambda} \right)^{(N-1)/2} \exp[-(\lambda + q)/2] I_{N-1}(\sqrt{\lambda q}), \quad (22)$$

$$\lambda = \frac{2NA^2}{\sigma_0^2}$$

with $2N$ degrees of freedom, $E\{q\} = \lambda + 2N$, $\text{Var}\{q\} = 4(\lambda + N)$. Notice that, unlike the strictly Gaussian assumption sometimes used in detection problems, the presence of signal affects both the mean and variance of $p_1(\cdot)$.

For the purpose of performance prediction calculations we assume the ETI signal bandwidth to be much less than an analyzer bin-width, and that, if present, the signal is centered in a bin. These assumptions correspond roughly to Swirling's Case I model (Ref. 5). We will not discuss the effects of moving the signal within a bin in this article. We wish to examine the detection performance of several processing scenarios which cover a side range of cost and implementation complexity. The cases in this work are summarized in Fig. 5, and illustrated with data flow diagrams in Figs. 6-9.

A. Case I

Referring to Fig. 6, we see that detection is carried out by applying a single threshold test to the individual $q_i(f_\alpha)$. This case corresponds most closely to the classical radar detection problem with noncoherent integration (Refs. 4, 5, 8). Since *a priori* probabilities cannot be realistically assigned for H_0 and H_1 , we choose to use the Neyman-Pearson criterion for choosing our decision threshold. With the usual notation we define the probability of false alarm as:

$$P_{FA} = P(D_1 | H_0) = \int_{\gamma}^{\infty} p_0(q) dq \quad (23)$$

or

$$P_{FA} = 1 - F_0(\gamma; q) \quad (24)$$

where $F_0(\gamma; q)$ is the cumulative distribution function describing H_0 , evaluated for $q = \gamma$; thus

$$F_0(\gamma; q) = \int_{\gamma}^{\infty} p_0(q) dq, \quad q \geq 0 \quad (25)$$

This later form is more useful for numerical computations (Refs. 4, 8, 9). Similarly the probability of detection is:

$$P_D = P(D_1 | H_1) = \int_{\gamma}^{\infty} p_1(q) dq \quad (26)$$

or, as above

$$P_D = 1 - F_1(\gamma; q) \quad (27)$$

Using the Neyman-Pearson criterion our threshold value, γ , is found from the required false alarm rate. Thus, we solve

$$\int_0^{\gamma} p_0(q) dq = 1 - \beta \quad (28)$$

where β is the desired false alarm rate.

B. Case II

Here we return to a discussion of the effects of antenna beam and shape and scan rate on our ability to detect an ETI source. A preliminary discussion of these effects for the case $N = 0$ (no accumulation) is described in Lockshin and Olsen (Ref. 10). The discussion leading up to Eq. (17) assumed that a scan strategy could be devised so that $N\tau \ll 1/\nu$. Thus, the

accumulator output gives us a sky sample every $N\tau$ seconds, or spatial sampling at "points" separated by $N\tau\omega$ radians. We will call this spacing the "intra-scan" separation.

$$\delta = N\tau\omega \text{ radians} \quad (29)$$

Intuitively it is clear that, given a finite Ω , and assuming a uniform PDF for the spatial distribution of ETI sources, the probability of detection should be a monotonically decreasing function of δ using the test statistic ξ_i (or equivalently, q_i). Looked at from another viewpoint, for a given choice of δ , the scanning operation cannot be accomplished at uniform sensitivity, an effect referred to as scalloping in Ref. 10. This means that for a source located between ξ_i and ξ_{i+1} the probability of detection will be less than if the source were at either ξ_i or ξ_{i+1} .

In order to provide more uniform scan sensitivity, the accumulator output, after baseline correction, is passed through a convolutional filter as shown in Fig. 7. We thus have a new test statistic given by:

$$u_i = \sum_{m=0}^{M-1} a_m q_{i-m} \quad (30)$$

where the filter weights, a_m , are chosen to optimize P_D over the sky area covered by the M sample points. Due to the weighted sum in Eq. (30) the PDFs under H_0 and H_1 are no longer simply chi-square or non-central chi-square. Writing the test statistic as:

$$u = \sum_{i=1}^M a_i q_i \quad (31)$$

a single component, $u_j = a_j q_j$, has a PDF given by:

$$p(u_j) = \frac{1}{|a_j|} p(q_j/a_j)$$

Using the Fourier transform scaling property and the known characteristic functions for $p_0(q)$ and $p_1(q)$, the characteristic functions for $p_0(u)$ and $p_1(u)$ are found to be:

$$C_0(j\omega; u) = \prod_{i=1}^M \frac{1}{(1 - j 2a_i \omega)^N} \quad (32)$$

and

$$\begin{aligned} C_1(j\omega; u) &= \exp\left(-\frac{NMA^2}{2\sigma_0^2}\right) \\ &\times \exp\left(\sum_i^M \frac{NA^2 \sigma_0^2/2}{[1 - j 2a_i \sigma_0^2 \omega]}\right) \\ &\times \prod_{i=1}^M \frac{1}{(1 - j 2a_i \sigma_0^2 \omega)^N} \end{aligned} \quad (33)$$

As will be seen in Section IV it is not necessary to invert Eqs. (32) and (33) to obtain explicit expressions for $p_0(u)$ and $p_1(u)$. A discussion of weighted signal summation may be found in Di Franco and Rubin (Ref. 7). Since the test statistic u is composed of M samples covering a particular area of the sky, we actually need to compute:

$$p_1(u) = \iint_{\Gamma} p_1(u | \xi_1^0, \xi_2^0) p(\xi_1^0) p(\xi_2^0) d\xi_1^0 d\xi_2^0 \quad (34)$$

where (ξ_1^0, ξ_2^0) describes the source position. Here, $\{p(\xi_1^0) p(\xi_2^0)\}$ is the joint PDF describing the spatial distribution of sources, and is assumed to be uniform. The dependence of $p(u)$ on (ξ_1^0, ξ_2^0) enters through the parameter A , which depends on intrascan separation, δ , HPBW, and relative source position. The required detection probabilities can be computed without an explicit evaluation of Eq. (34).

C. Case III

Further improvements in system performance can be obtained by combining observations from adjacent scan lines, i.e., "inter-scan line" signal combination. Case III represents an implementation of inter-scan line combination with a means for drastically reducing the amount of data storage required. Data decimation is achieved by "pre-thresholding" the convolutional filter output so as to allow only a certain percentage (on average) of the data to pass through for further processing. The pre-threshold level is set according to:

$$\int_{\gamma_p}^{\infty} p_0(u) du = \eta \quad (35)$$

where η is the fraction of data allowed to pass threshold. By introducing inter-scan signal combination, we now have a test statistic which is bivariate, and in which each of the

marginal distributions is a thresholded version of $p_0(y)$, and $p_1(y)$. If we denote the test statistic after prethresholding by u' , then (Ref. 11),

$$p_i(u') = \begin{cases} p_i(u), & u > \gamma_p \\ p_i(\gamma_p) \delta(u - \gamma_p), & u = \gamma_p \\ 0, & u < \gamma_p \end{cases}, \quad i = 1, 2 \quad (36)$$

The behavior of this thresholding operation is illustrated in Fig. 10. As noted in Fig. 8, our test statistic, u , is a two-dimensional vector whose components are $\{u_k, u_{k+K}\}$. For notational convenience we will simply write $\vec{u} = (u'_1, u'_2)$ and remember that u'_1 and u'_2 refer to sample observations from adjacent scan lines. Now in order to determine the average system response to an ETI signal located somewhere in the area defined by the two vertically adjacent beam patterns, we assume a uniform spatial distribution of sources all of the same strength. Note that this result depends on vertical scan separation and beam profile, but not on scan rate.

If we denote by ξ_1^0 and ξ_2^0 the horizontal and vertical coordinates describing the location of an ETI source (see Fig. 11), then the joint PDF describing \vec{u} is given by:

$$p_1(u'_1, u'_2) = \iint_{\Gamma} p_1(u_1 | \xi_1, \xi_2) p_1(u_2 | \xi_1, \xi_2) \times p(\xi_1) p(\xi_2) d\xi_1 d\xi_2 \quad (37)$$

where the integration is carried out over the region Γ , as shown in Fig. 11. Since we are assuming uniform spatial distribution of source locations,

$$p(\xi_1) p(\xi_2) = 1/\Gamma$$

The explicit dependance of $p_1(u_i | \xi_1, \xi_2)$, $i = 1, 2$, on (ξ_1, ξ_2) enters through the system response dependence on beam profile in a manner analogous to that shown by Eq. (7). Again assuming a Gaussian beam profile, the expression for $A(\xi_1, \xi_2)$ is given by:

$$A(\xi_1, \xi_2) = \frac{\pi D^2}{4} \epsilon S_0 \exp \left\{ \frac{-2}{\Omega^2} [(\xi_1 - \xi_1^0)^2 (\xi_2 - \xi_2^0)^2] \right\} \quad (38)$$

The situation described above will be recognized as a problem in composite hypothesis testing for a bivariate random

variable (Refs. 2, 12). Specifically, the null hypothesis is simple, while the alternative hypothesis is composite, being conditioned on the ETI source location parameters. Furthermore, since we are now dealing with a bivariate distribution, the decision regions, R_0 and R_1 , are no longer divided by a single threshold value, but in fact by a family of such values which define a decision curve on the (u'_1, u'_2) -plane. The situation for a simple case is shown in Fig. 12, where the Neyman-Pearson curve is taken to be a quarter-circle; the region excluded by prethresholding is shaded.

D. Case IV

For analysis purposes this case is identical to Case III except that we remove the prethresholding operation and allow the data to continue on to be combinatorially processed. A practical implementation of this case would of course require a very large amount of high-speed bulk storage and significantly increase processor speed requirements.

As can be seen from the above discussions, we have assumed a signal processing implementation which defers RFI and "natural source" identification operations to the very last processing step. This is not only useful for analysis purposes, but is in fact probably the only practical manner in which to implement a sky survey processing strategy. As a final note, we point out that all of our analyses assume that the "noise" is stationary, i.e., position independent. If this turns out to be a poor assumption, then the detection performance must be re-evaluated to incorporate adaptive threshold methods (Refs. 13, 18). The following section is devoted to describing in some detail the actual implementation of performance calculations carried out based on the problem formulation described in this section.

V. Performance Analysis Calculations

In this section we wish to describe the methods used to carry out the required computations based on the discussion of Section III. The objective of these calculations is to produce a set of receiver operating characteristic (ROC) curves for each of the processing Cases (I-IV). These ROC curves will then be used to choose the signal processing strategy that yields the best performance/cost ratio. As has been pointed out in the literature (Refs. 8, 9, 14), numerical implementation of detection probability analyses encounters several practical difficulties. Some of these difficulties will be discussed before we proceed to describe the computational implementations actually used.

For our purposes, there are basically two computational issues to be addressed: (1) the use of Gaussian versus true PDFs; and (2) the choice of a numerical computation proce-

dures for the required probabilities. At first sight one might suspect, from the Central Limit Theorem, that Gaussian PDFs would provide an adequate approximation for our cases. However, we need to keep in mind that because of the small false alarm rate requirements ($<10^{-8}$), we are working very far out on the tails of distributions. Due to the relatively small number of samples involved (<50) one cannot expect the Gaussian to provide a very good approximation except near the "core" of the distribution. When using Gaussians to model the problem, we will use the means and variances computed from the actual non-central chi-square distributions as the Gaussian parameters. Regardless of the actual PDFs used, the choice of a numerical evaluation algorithm is critical in determining both solution accuracy, precision requirements, and computational time. Direct numerical integration of the PDFs invariably requires the use of extended computation precision and dynamic range and can result in excessively long CPU times (Refs. 8, 9) when one demands small false alarm rates.

We shall begin our discussion by setting out those elements of the computation that are common to all four cases. A general computation flow diagram is shown in Fig. 13. As pointed out in Section III, all variables are normalized by σ_0^2 so that the system noise PDF is zero-mean Gaussian with unit variance. The quantity σ_0^2 is determined solely by the effective system temperature and thus represents the noise energy at the receiver front-end. The SNRs used in computing the ROC curves are referred to the receiver front-end, and are defined as

$$S \equiv \text{SNR} = \frac{\pi D^2 \epsilon S_0}{4\sigma_0^2}, \quad (39)$$

$$\sigma_0^2 = kT_s/\tau$$

where k is Boltzmann's constant, and T_s is the system temperature in Kelvin. In terms of the system parameters then:

$$S = \frac{\pi D^2 \epsilon \tau S_0}{4kT_s} \quad (40)$$

In the numerical implementation of calculations described in this section all expressions for P_{FA} and P_D are parameterized in terms of S .

One of the more elegant and powerful methods for efficient numerical evaluation of detection probabilities has been described by Helstrom (Refs. 14, 15). Recalling the expressions for P_{FA} and P_D from Section III,

$$P_{FA} = 1 - F_0(\gamma; q) = 1 - \int_{-\infty}^{\gamma} p_0(q) dq \quad (41)$$

$$P_D = 1 - F_1(\gamma; q) = 1 - \int_{-\infty}^{\gamma} p_1(q) dq \quad (42)$$

and that the moment generating function (MGF) for $p(x)$ is defined by

$$\mu(\nu) = E \{e^{-\nu x}\} = \int_0^{\infty} p(x) e^{-\nu x} dx, \quad x \geq 0 \quad (43)$$

we may write a general expression for the required quantities using the inverse Laplace transform,

$$1 - F(\gamma; x) = L^{-1} \{ \nu^{-1} [1 - \mu(\nu)] \} \quad (44)$$

$$= \int_{c-i\infty}^{c+i\infty} \nu^{-1} [1 - \mu(\nu)] e^{\nu x} \frac{d\nu}{2\pi i}$$

where the contour of integration is parallel to the imaginary axis and lies to the right of all poles in the integrand (Ref. 16). Helstrom's approach uses a saddlepoint integration (Ref. 17) technique to obtain an efficient approximation to the contour integral in Eq. (44). Taking the contour to the left of the origin, but still to the right of all singularities in the integrand, Eq. (44) may be expressed as

$$1 - F(\gamma; x) = \int_{c-i\infty}^{c+i\infty} -\nu^{-1} \mu(\nu) e^{\nu x} \frac{d\nu}{2\pi i}, \quad c < 0 \quad (45)$$

Using the fact that the integral in Eq. (45) is convex over $(-\infty < \nu < 0)$ the contour is placed so as to pass through $c = \nu_0$, i.e., $\nu = \nu_0 + iy$ $(-\infty < y < \infty)$, the point ν_0 being the saddlepoint. Writing

$$\nu^{-1} \mu(\nu) e^{\nu x} = e^{\Phi(\nu)} \quad (46)$$

and expanding $\Phi(\nu)$ as a Taylor series about ν_0 ,

$$\begin{aligned} \Phi(\nu) &= \Phi(\nu_0) + \frac{1}{2} \Phi''(\nu_0) (\nu - \nu_0)^2 + \dots \\ &+ \frac{1}{k!} \Phi^{(k)}(\nu_0) (\nu - \nu_0)^k + \dots \end{aligned} \quad (47)$$

the expression for $[1 - F(\gamma; x)]$ becomes

$$1 - F(\gamma; x) = -\nu_0^{-1} [2\pi\Phi''(\nu_0)]^{-1/2} \times \mu(\nu_0) e^{\nu_0\gamma} (1 + R), \quad \nu_0 < 0 \quad (48)$$

where R is the residual term given by Helstrom (Ref. 14). The saddlepoint is found simply by solving

$$\Phi'(\nu) = \frac{d}{d\nu} \ln [\mu(\nu)] + \gamma - \frac{1}{\nu} = 0 \quad (49)$$

numerically and noting that $\Phi(\nu)$ possesses only a single minimum ($\nu = \nu_0$) on the interval $(-\infty < \nu < 0)$. As Helstrom points out, for $|E\{q\} - \gamma| \gg \sigma_q$, the residual term may be neglected and we shall use the zero-order approximation,

$$1 - F(\gamma; x) = -\nu_0^{-1} [2\pi\Phi''(\nu_0)]^{-1/2} \mu(\nu_0) e^{\nu_0\gamma}, \quad \nu_0 < 0 \quad (50)$$

The threshold, γ , for a given choice of P_{FA} is determined by an iterative solution of Eq. (50).

The treatment of Case II and Cases III and IV requires evaluation of conditional probabilities. As can be seen from Eqs. (21) and (36), direct evaluation of the integrals involved is quite tedious and such an approach would not allow us to take advantage of Helstrom's approximation technique. We therefore propose to treat these cases in the framework of a fluctuating target model (Ref. 5). The "fluctuation" is being induced by the antenna scan parameters and the assumption of a uniform spatial distribution for the probability of finding the source at a particular location within the area covered. Another view of Eqs. (31) and (37) is that the source position (ξ_1^0, ξ_2^0) results in a conditioning of $p_1(\cdot)$ on λ , the non-central parameter.

Utilizing the above interpretation of the conditional probabilities encountered in Cases II-IV, we rewrite Eq. (43) as

$$\mu(\nu | \lambda) = \int_0^\infty p_1(q) e^{-\nu q} dq \quad (51)$$

where $p_1(q)$ is our non-central chi-square PDF with N degrees of freedom and non-central parameter λ . Recall that λ is in fact the total signal-to-noise ratio at the output of the convolutional filter. For the given PDF, Eq. (51) yields (Ref. 14)

$$\mu(\nu | \lambda) = (1 + \nu)^{-N} \exp [-\lambda\nu/(1 + \nu)] \quad (52)$$

Denoting by $p_\lambda(\lambda | \bar{\lambda})$ the PDF of λ which describes the fluctuation model, the unconditional MGF is obtained from

$$\mu(\nu) = \int_0^\infty \mu(\nu | \lambda) p_\lambda(\lambda | \bar{\lambda}) d\lambda \quad (53)$$

With suitable change of variable, Eq. (53) is seen to be the Laplace transform of $p_\lambda(\cdot)$ and thus the unconditional MGF is

$$\mu(\nu) = (1 + \nu)^{-N} \mu_\lambda[\nu/(1 + \nu)] \quad (54)$$

where $\mu_\lambda[\cdot]$ is the MGF of the distribution describing our fluctuation model. Therefore, given a suitable fluctuation model which yields $\mu_\lambda[\cdot]$ in analytic form, we may utilize the saddlepoint method to compute approximate detection probabilities for Cases II-IV.

Recalling the discussion and assumptions leading up to Eq. (17) in Section III, our Case II appears to formally correspond to the Swerling model I. In practice the situation actually lies somewhere between Swerling I and II. For simplicity we take our fluctuation model PDF to be (Ref. 5)

$$p_\lambda(\lambda | \bar{\lambda}) = \frac{1}{\bar{\lambda}} \exp(-\lambda/\bar{\lambda}) \quad (55)$$

where $\bar{\lambda}$ is computed by averaging the antenna response over the area covered by our M -point convolution of Eq. (28). Note that this averaging process includes the convolutional weights for each of the M samples. Cases III and IV are treated in the same way, the obvious extension to bivariate distributions.

Finally we note that the MGF describing the test statistics used in Cases II-IV, can be derived in a manner similar to that used in finding the characteristic functions, Eqs. (32) and (33). Thus, for Case II the MGF's under $\{H_0\}$ and $\{H_1\}$ are given by

$$\mu_0(\nu) = \prod_{i=1}^M (1 + a_i \nu)^{-N} \quad (56)$$

and

$$\mu_1(\nu) = \exp \left[-\sum_{i=1}^M \frac{a_i \lambda \nu}{(1 + a_i \nu)} \right] \prod_{i=1}^M (1 + a_i \nu)^{-N} \quad (57)$$

where, as before, the a_i denote the convolutional filter weights. Here it is most convenient to use the cumulant generating function, $h(\nu) = \ln \mu(\nu)$ and we then have

$$h_0(\nu) = -N \sum_{i=1}^M \ln(1 + a_i \nu) \quad (58)$$

and

$$h_1(\nu) = - \sum_{i=1}^M \frac{a_i \lambda \nu}{(1 + a_i \nu)} - N \sum_{i=1}^M \ln(1 + a_i \nu) \quad (59)$$

Results of our performance calculations for the Gaussian model are shown in Fig. 14 and 15, for P_{FA} of 10^{-12} and

10^{-8} respectively. For an assumed antenna temperature of 25 K, an SNR ratio of 1 is equivalent to a received flux of approximately 2×10^{-23} W/m² at the antenna. The relative performance of the four cases studied is easily seen from these Receiver Operating Characteristic (ROC) curves. The largest increase in performance comes in going from Case I to Case II, where we add intra-scan combination processing. As is to be expected, there is little difference in performance between Cases III and IV; both cases yielded a performance increase of about a factor of two over that for Case II. A similar set of ROC calculations is currently being carried out using the actual chi-square distributions which describe the problem and will be reported in a later paper.

References

1. Wilck, H. C., Quirk, M. P., Grimm, M. J., "A Wide-Band, High-Resolution Spectrum Analyzer," *The Telecommunications and Data Acquisition Progress Report 42-83*, The Jet Propulsion Laboratory, Pasadena, CA, 1985.
2. Whalen, A. D., *Detection of Signals in Noise*, Academic Press, New York, 1971.
3. Helstrom, C. W., *Statistical Theory of Signal Detection*, Pergamon Press, New York, 1968.
4. Marcum, J. I., and Swerling, P. A., "A Statistical Theory of Target Detection by Pulsed Radar," *IRE Trans. Inform. Theory*, IT-6, 59-267 (April 1960).
5. Swerling, P. A., "Probability of Detection for Fluctuating Targets," *IRE Trans. Inform. Theory*, IT-6, 269-308 (April, 1960).
6. Patnaik, P. B., "The Non-Central Chi-Square and F-distributions and their Applications," *Biometrika*, 36, 202 (1949).
7. DiFranco, J. V., and Rubin, W. L., *Radar Detection*, ARTECH House, Inc., Dedham, MA, 1980.
8. Bird, J. S., "Calculating Detection Probabilities for Systems Employing Noncoherent Integration," *IEEE Trans. on Aerospace and Electronic Systems*, AES-18, 401-409 (July, 1982).
9. Helstrom, C. W., and Ritcey, J. A., "Evaluating Radar Detection Probabilities by Steepest Descent Integration," *IEEE Trans. Aerospace and Electronic Systems*, AES-20, 624-633 (September, 1984).
10. Loksins, A. and Olsen, E. T., "An Investigation of the Effects of Scan Separation on the Sensitivity of the SETI All Sky Survey for the Case of Gaussian Noise," *TDA Progress Report 42-77*, pp. 151-158, Jet Propulsion Laboratory, Pasadena, CA, Mar. 1984.
11. Papoulis, A., *Probability, Random Variables, and Stochastic Processes*, Chap. 5, McGraw-Hill, New York, 1965.

12. Lehman, E. L., *Testing Statistical Hypotheses*, Wiley, New York, 1959.
13. Bird, J. S., "Calculating Detection Probabilities for Adaptive Thresholds," *IEEE Trans. Aerospace and Electronic Systems*, AES-19, 506-512 (July 1983).
14. Helstrom, C. W., "Approximate Evaluation of Detection Probabilities in Radar and Optical Communications," *IEEE Trans. Aerospace and Electronic Systems*, AES-14, 630-640 (July 1978).
15. Helstrom, C. W., "Evaluating the Detectability of Gaussian Stochastic Signals by Steepest Descent Integration," *IEEE Trans. Aerospace and Electronic Systems*, AES-19, 428-437, (May 1983).
16. Feller, W., *An Introduction to Probability Theory and its Applications*, Vol. II, Chap. XIII, (John Wiley, New York, 1971).
17. Jeffreys, H., *Asymptotic Approximations*, Oxford Univ. Press., London, 1962.
18. Nitzberg, R., "Constant-False-Alarm-Rate Signal Processors for Several Types of Interference," *IEEE Trans. Aerospace and Electronic Systems*, AES-8, 27-34, (Jan. 1972).

**Table 1. Characteristics of the All Sky Survey
(for a 34-m parabolic antenna)**

Spatial Coverage	4π steradians
Frequency Coverage	1-10 GHz and higher spot bands
Duration of Survey	5-10 years
Dwell Time on Source	0.5-10 seconds
Frequency Resolution	10-30 Hz
Sensitivity	$\approx 10^{-23}$ Watts/meter ²
Spatial Uniformity	12% peak-to-peak
Variation with Frequency	$\approx \nu^{1/2}$
Polarization	Simultaneous Dual Circular
Signal Type	Continuous Wave (CW)

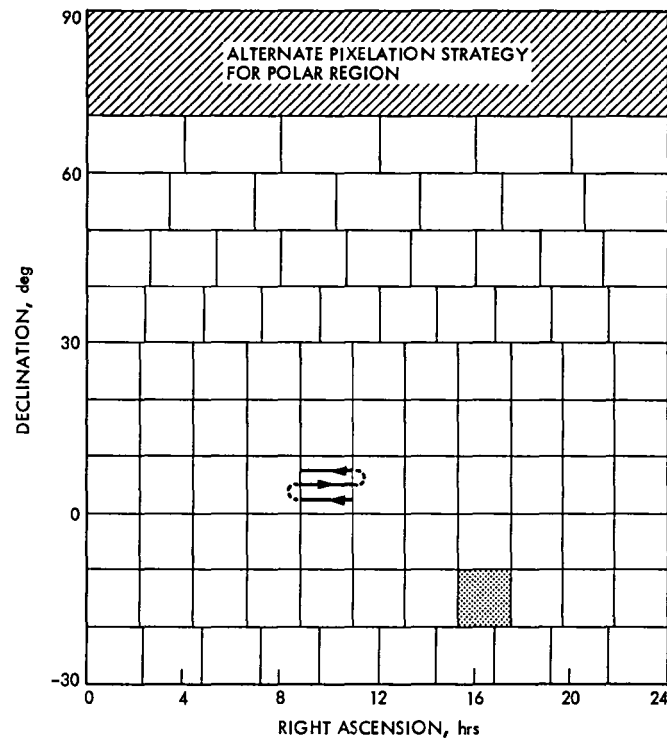


Fig. 1. Schematic representation of sky pixelation and scan strategy

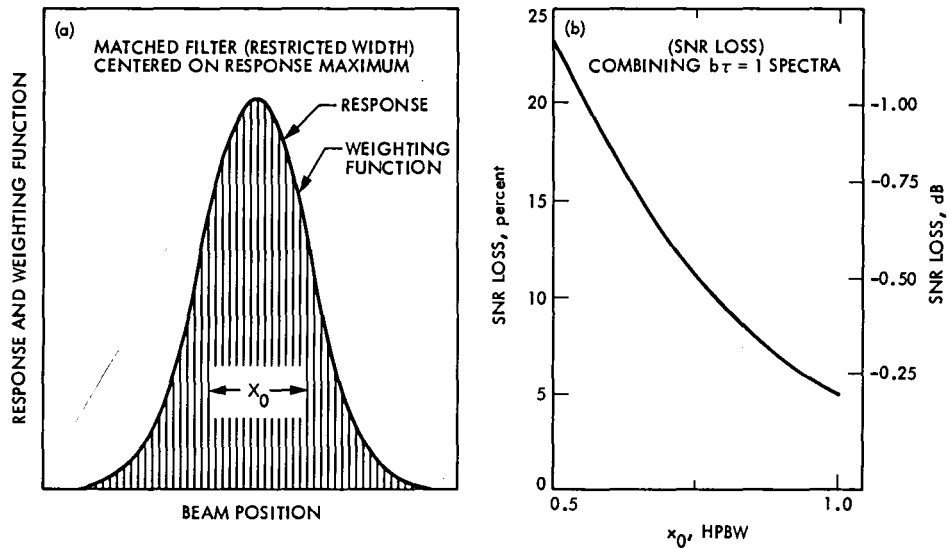


Fig. 2. Antenna scan effects: (a) the expected response within a frequency bin as the antenna beam is swept across a source; (b) the SNR loss as a function of filter length, X_0 , in HPBWs

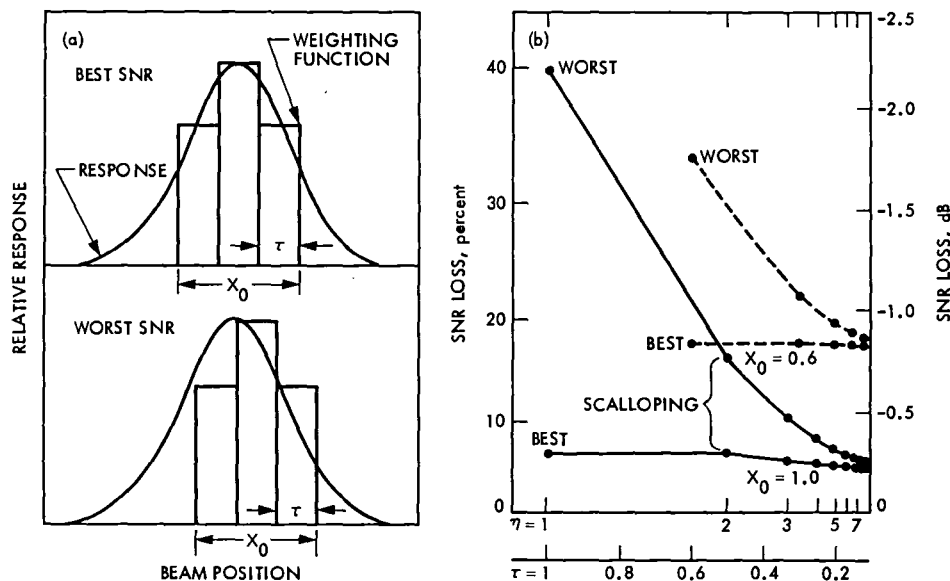


Fig. 3. Scalloping effects: (a) Schematic representation of the source of scalloping; (b) variation of SNR loss

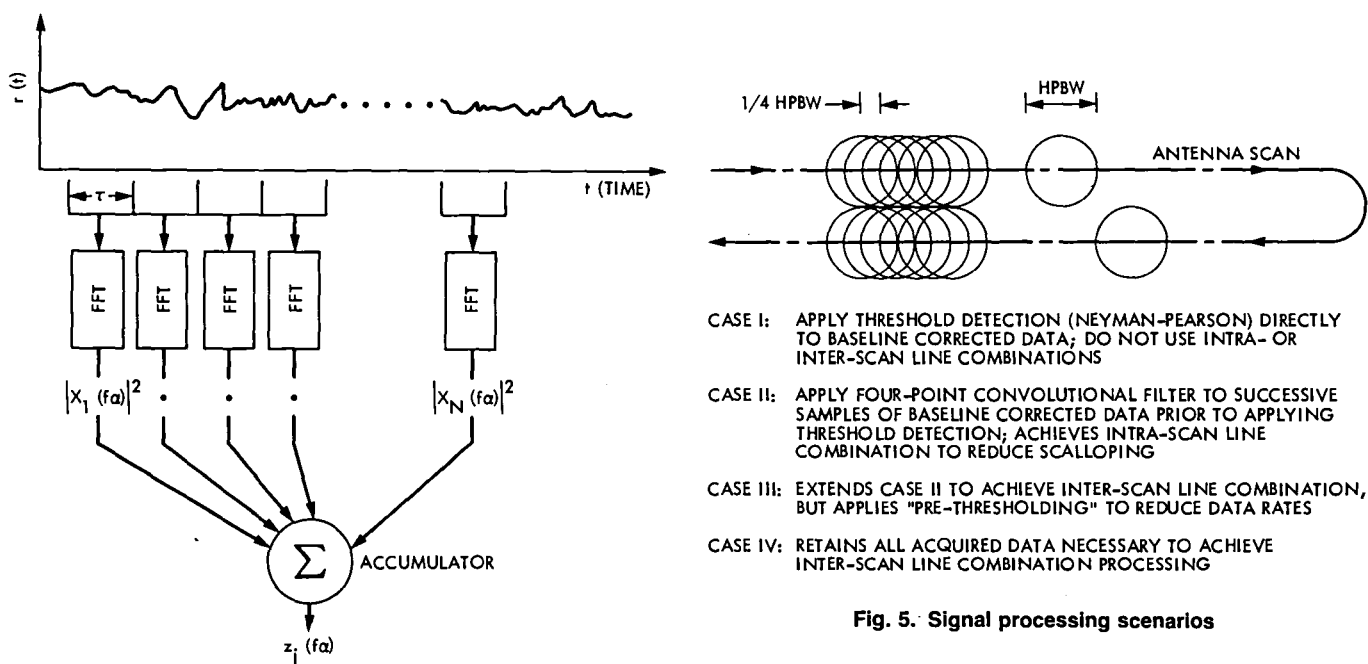


Fig. 4. Front-end signal processing

Fig. 5. Signal processing scenarios

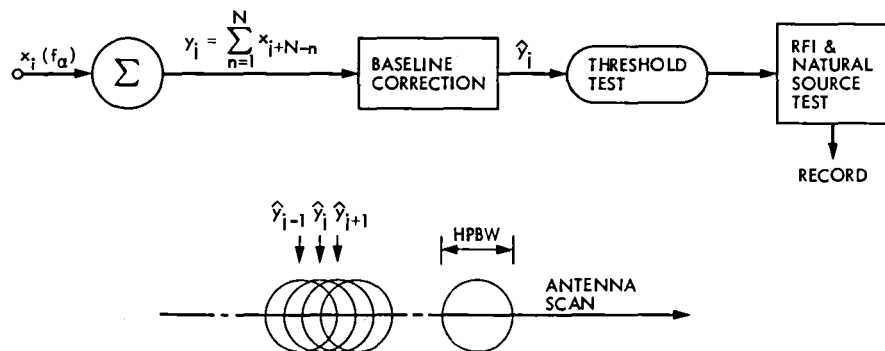


Fig. 6. Signal processing, Case I

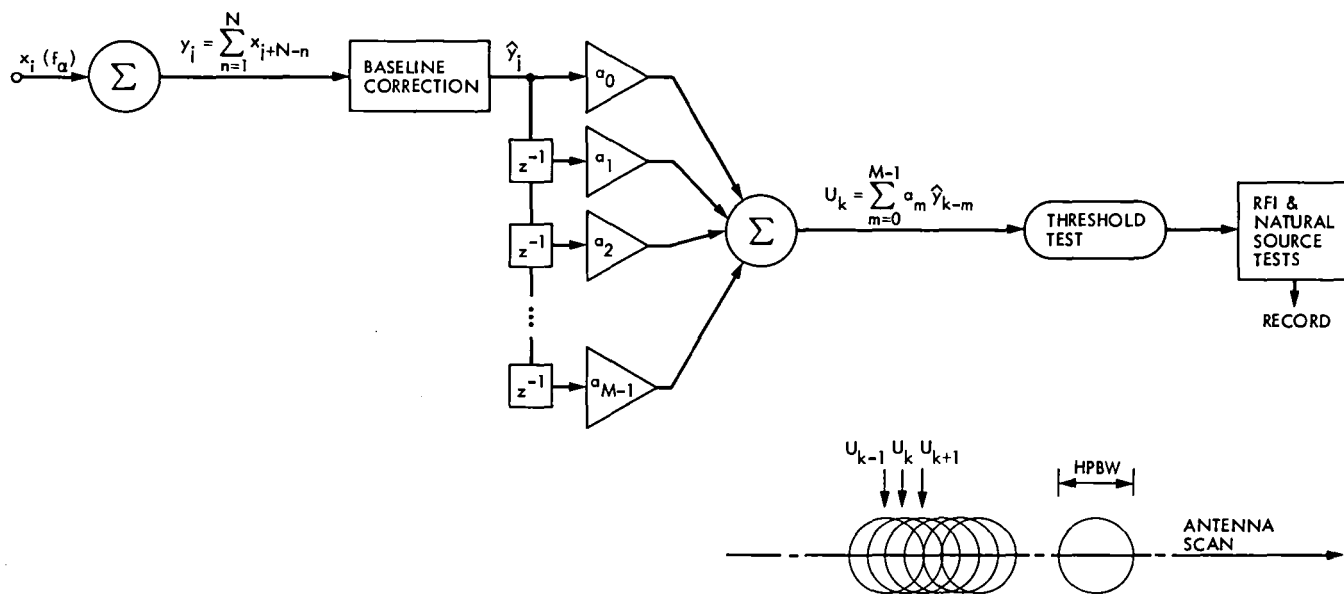


Fig. 7. Signal processing, Case II

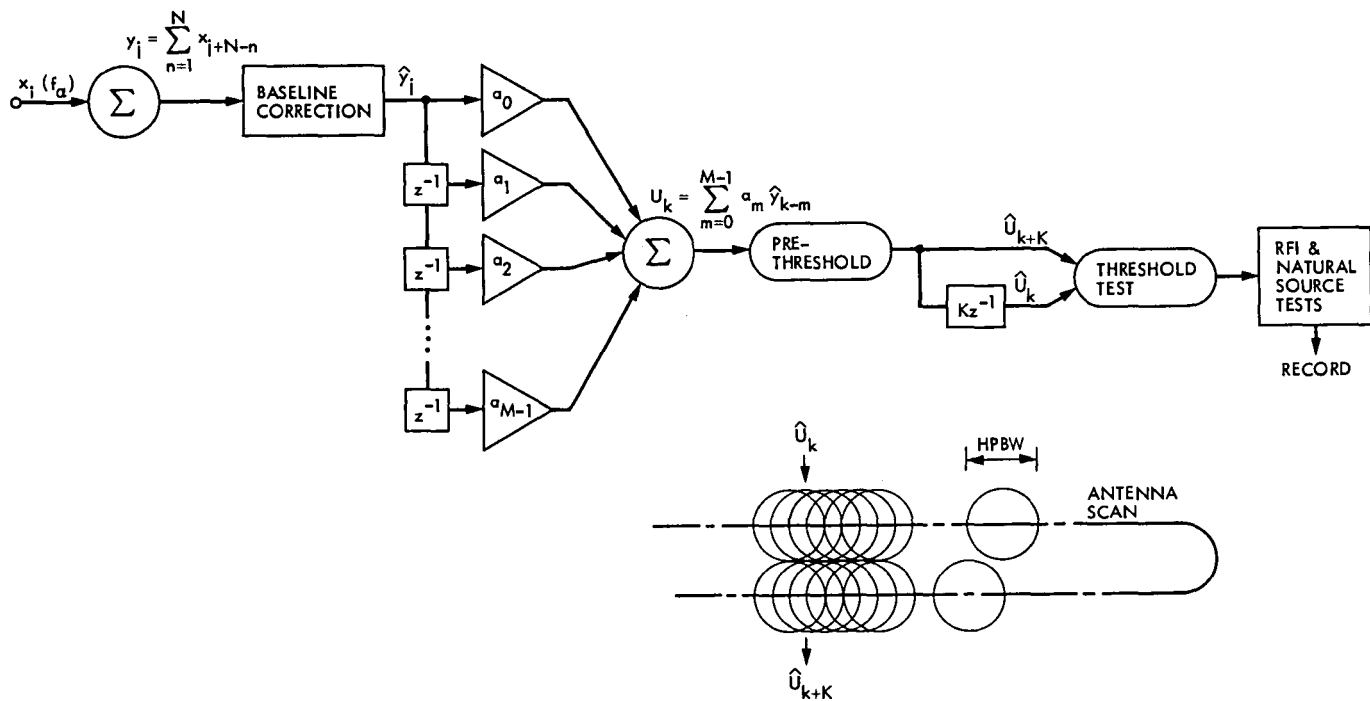


Fig. 8. Signal processing, Case III

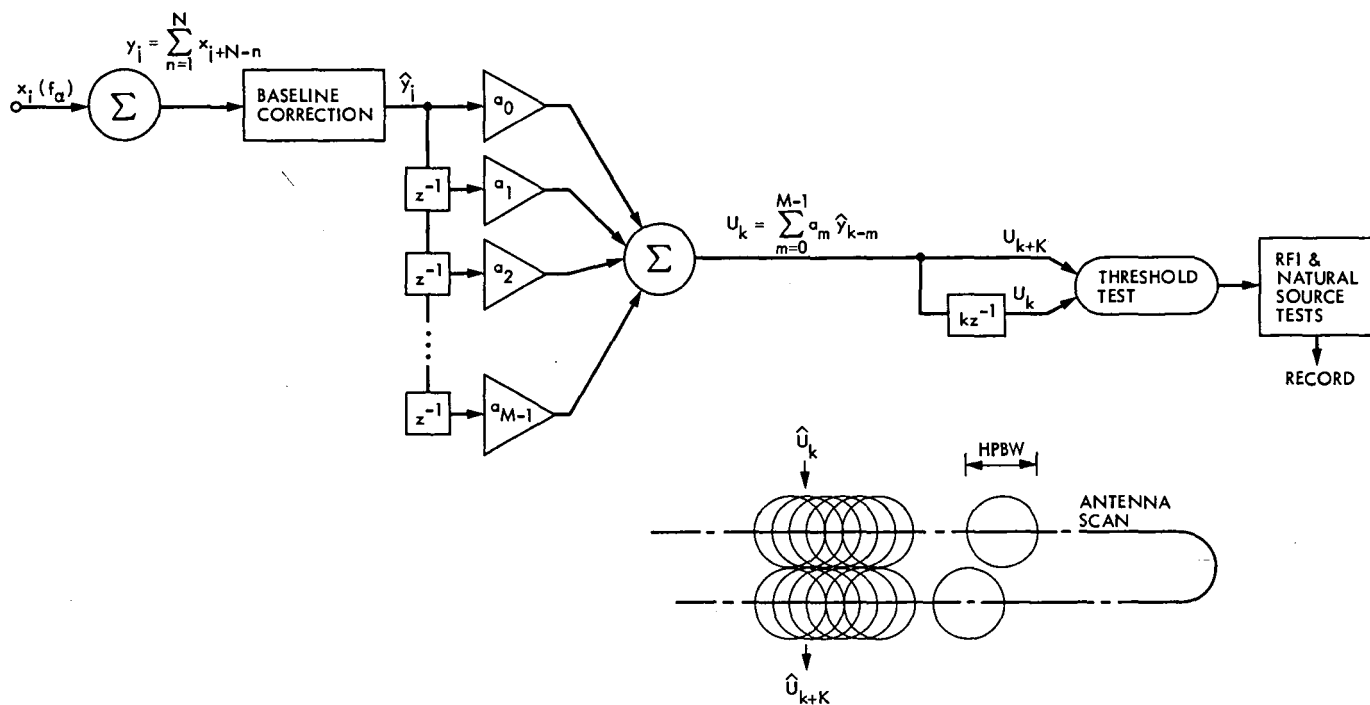


Fig. 9. Signal processing, Case IV

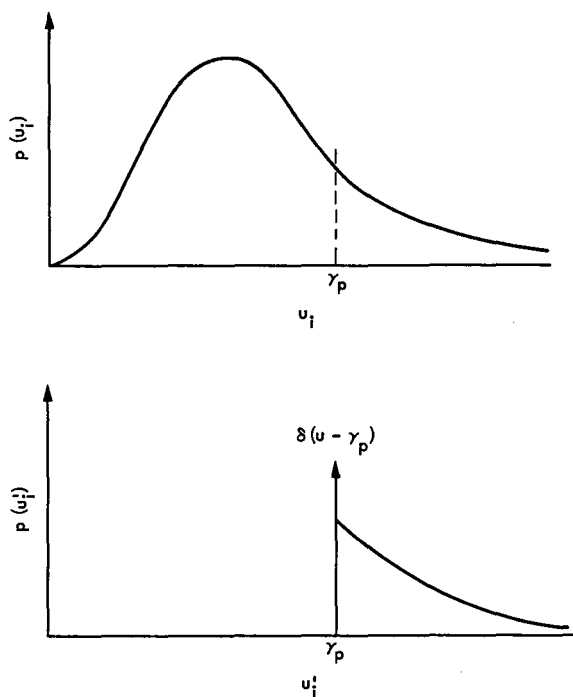
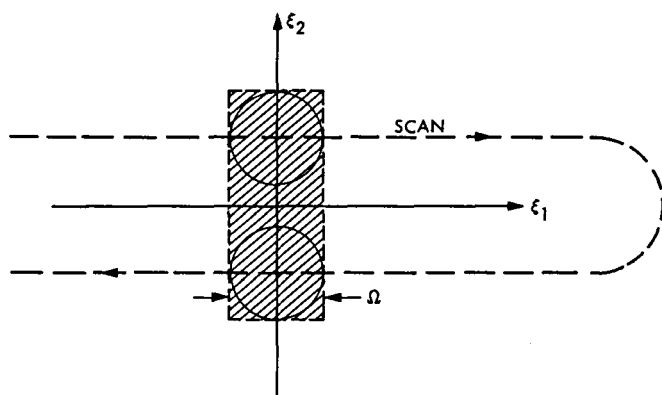


Fig. 10. Effects of the thresholding operation



AREA OF INTEGRATION, Γ , SHOWN SHADED

Fig. 11. Inter-scan combination geometry

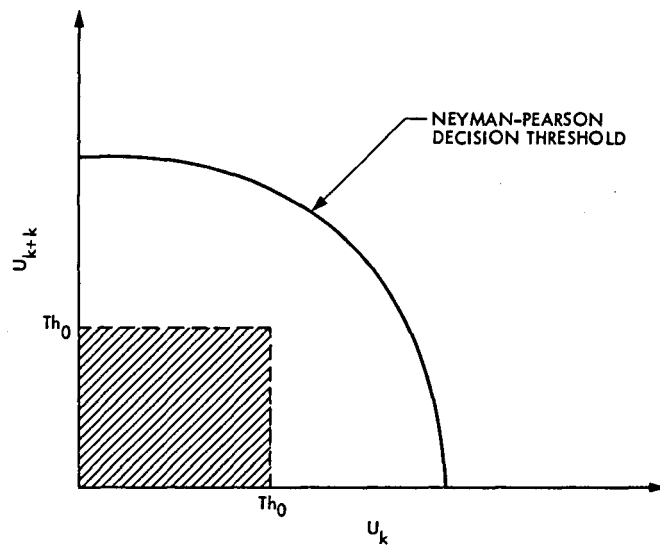


Fig. 12. Inter-scan combination case

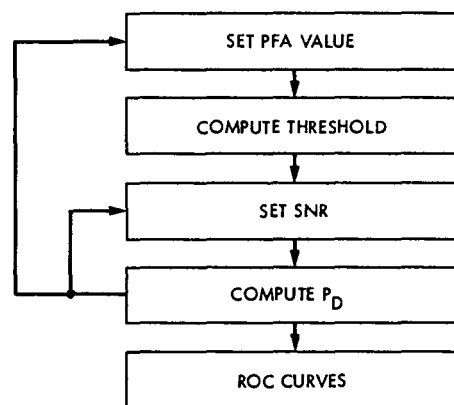


Fig. 13. Computational flow

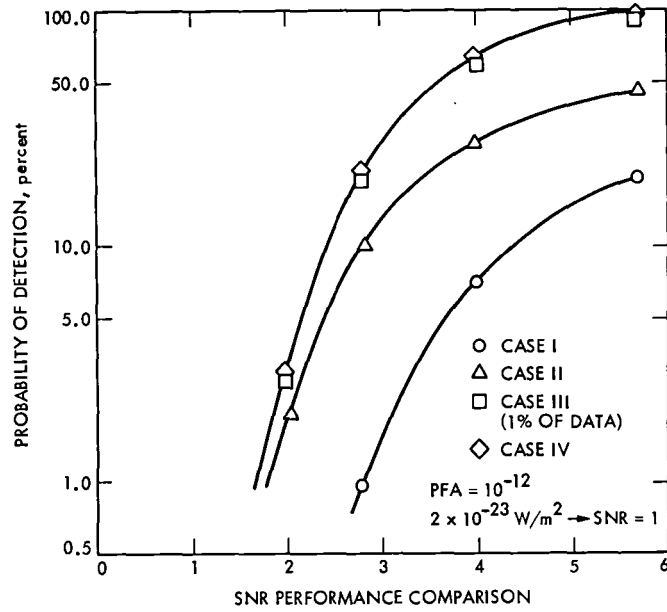


Fig. 14. Receiver operating characteristic curves for $PFA = 10^{-12}$

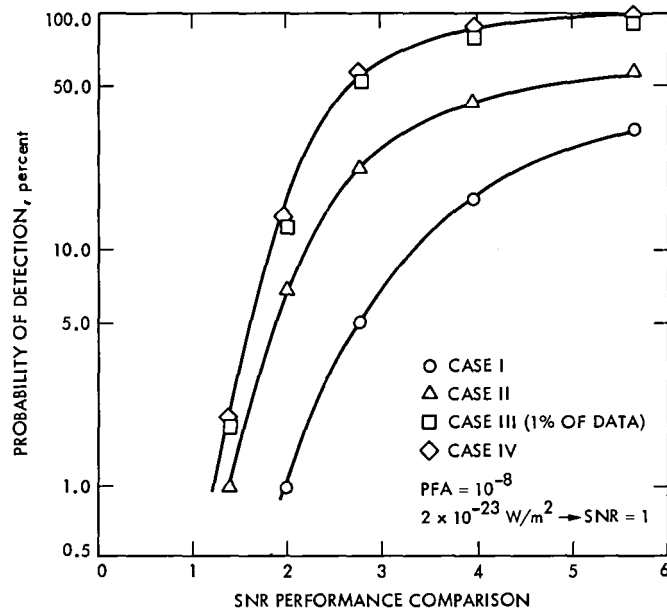


Fig. 15. Receiver operating characteristic curves for $PFA = 10^{-8}$

The Number of Stable Points of an Infinite-Range Spin Glass Memory

R. J. McEliece

California Institute of Technology

E. C. Posner

Telecommunications and Data Acquisition Office

This paper finds a rigorous asymptotic expression for the number of stable points of an infinite-range spin glass with independently identically distributed (i.i.d.) zero-mean gaussian exchange interactions. The result also applies to the number of stable points of a Hopfield Memory (a kind of associative memory) when the memory connections are i.i.d. zero-mean gaussians. The result is that the number of stable points is asymptotic to a constant slightly larger than 1 times 2 to a power slightly larger than $n/4$, where n is the number of spins in the glass, or the length of the n -tuples to be remembered by the memory. The answer is easily derived using simple asymptotic techniques from an exact expression for the probability that an arbitrary ± 1 n -tuple of spins is a fixed point. This expression is obtained from the fact that any distribution of joint zero-mean gaussians of given covariances is specified solely by these covariances. This is a far shorter derivation of the result than those existing.

I. Introduction

A spin glass (Ref. 1) can be identified with n -tuples of ± 1 's, called *spins* σ_i , $1 \leq i \leq n$. Spins i and j ($i \neq j$) interact via an *exchange interaction* symmetric matrix $J = (J_{ij})$. Spin i "influences" spin j via J_{ij} , and j influences i likewise via $J_{ji} = J_{ij}$; $J_{ii} = 0$, $1 \leq i \leq n$. This means that, in the absence of other interactions, spin value σ_i changes to (or remains equal to) the value

$$\sigma'_i = \text{Sgn} \left(\sum_{j=1}^n J_{ij} \sigma_j \right) \quad (1)$$

where the Sgn of a nonzero real number is its sign. We ignore the possibility that the sum in Eq. (1) is 0, for this event will have probability 0 in our model, where the J_{ij} , $i \neq j$, are independent identically distributed (i.i.d.), zero-mean gaussian, $n(n-1)/2$ in number. Thus $\sigma'_i = \pm 1$ accordingly as $\sum_{j=1}^n J_{ij} \sigma_j$ is positive or negative. An n -tuple of ± 1 's,

$$\sigma = (\sigma_1, \sigma_2, \dots, \sigma_n)$$

is then a fixed point of the spin glass specified by the $n \times n$ symmetric matrix

$$J = (J_{ij})$$

provided

$$\sigma' = \sigma$$

i.e.,

$$\sigma'_i = \sigma_i, 1 \leq i \leq n \quad (2)$$

We ask, if the J_{ij} , $i \neq j$, are zero-mean i.i.d. gaussian (of positive variance), what is the expected number of fixed points in Eq. (2)?

Another formulation is to find the number of stable points of the Hopfield Memory (Ref. 2). In this, n -tuples σ of ± 1 's are input to an electronic device that connects each input to each other input via a conductance J_{ij} , where $J_{ij} = J_{ji}$, and all J_{ii} are zero. The n resulting voltages

$$\sum_j J_{ij} \sigma_j$$

are each hard-limited to become the new σ n -tuple, but again the changes occur at random rather than at synchronous times. See Ref. 3 for further details on the Hopfield Memory and its potential capacity as an associative or content-addressable memory in processing systems.

From Eqs. (1) and (2), σ is a fixed point if and only if

$$\text{Sgn} \left(\sum_j J_{ij} \sigma_j \right) = \sigma_i, 1 \leq i \leq n \quad (3)$$

This is also the fixed point formula if all components change at once; the difference between the two concepts of fixed points is not relevant in what follows.

II. Reduction to a Positivity Condition

We claim that the probability that σ be a fixed point is the same for all σ . This is because all the probabilities are equal to the one for $\sigma = (1, 1, \dots, 1)$. Merely replace J_{ij} by $J_{ij} \text{Sgn}(\sigma_i) \cdot \text{Sgn}(\sigma_j)$ so that Eq. (3) can be rewritten

$$\text{Sgn} \left(\left(\sum_j J_{ij} \text{Sgn}(\sigma_i) \text{Sgn}(\sigma_j) \frac{\sigma_j}{\text{Sgn}(\sigma_j)} \right) / \text{Sgn}(\sigma_i) \right) = \sigma_i \quad (4)$$

In Eq. (4), $\sigma_j / \text{Sgn}(\sigma_j) = 1$, $1 \leq j \leq n$, and $\sigma_i \text{Sgn}(\sigma_i) = 1$. So Eq. (4) can be written

$$\text{Sgn} \left(\sum_j J_{ij} \text{Sgn}(\sigma_i) \text{Sgn}(\sigma_j) \right) = 1, 1 \leq j \leq n \quad (5)$$

This is Eq. (3) with all $\sigma_i = +1$ and J_{ij} replaced by the $n(n-1)/2$ random variables

$$J_{ij} \text{Sgn}(\sigma_i) \text{Sgn}(\sigma_j)$$

identically distributed with the J_{ij} .

The expected number of fixed points F_n is then the probability that the *all* 1's n -tuple is fixed, times the number of n -tuples 2^n :

$$F_n = 2^n \text{pr} \left(\sum_{j=1}^n J_{ij} \geq 0, 1 \leq j \leq n \right) \quad (6)$$

Here we recall that the $J_{ij} = J_{ji}$ are $n(n-1)/2$ independent zero-mean gaussians of the same positive variance, which we take as 1; $J_{ii} = 0$, $1 \leq i \leq n$.

III. Covariances

What is the Covariance Matrix of two row sums of J ,

$$X_i = \sum_{j=1}^n J_{ij}$$

and

$$X_k = \sum_{j=1}^n J_{kj} \quad ?$$

The means are 0, because each J_{ij} has mean 0. The variances ($k = i$) are

$$E(X_i^2) = n - 1 \quad (7)$$

because there are $n-1$ independent random variables being added, each of variance 1 (recall $J_{ii} = 0$, all i).

What is $\text{Cov}(X_i, X_k)$, $i \neq k$? We claim that any two rows of the symmetric matrix J of exchange interactions has exactly one J -random variable in common, so that the rest are inde-

pendent. If we write the J_{xy} always with $x < y$, row i (with the 0 being in position i) is

$$(J_{1i} J_{2i} \cdots J_{i-1,i} 0 J_{i,i+1} \cdots J_{in})$$

while row k is

$$(J_{1k} J_{2k} \cdots J_{k-1,k} 0 J_{k,k+1} \cdots J_{kn})$$

We can let $i < k$ without loss of generality. Then the only random variable in common between the two rows is J_{ik} , by inspection. So

$$\text{Cov}(X_i, X_k) = E(X_i X_k) = 1, i \neq k \quad (8)$$

This is because the cross terms

$$E(J_{hi} J_{mk}) = E(J_{hi} J_{kn}) = E(J_{ip} J_{qk}) = E(J_{ip} J_{kr}) = 0$$

in every other case, because they involve mean-0 independent random variables.

IV. Equivalent Gaussians

Any n zero-mean gaussians of the same variances (Eq. (7)) and covariances (Eq. (8)) will do just as well (Ref. 4, Sec. 9.3) in order to find the probability

$$p_n = \text{pr}(X_i > 0, 1 \leq i \leq n)$$

This, from Eq. (6), is 2^{-n} times the desired answer F_n for the expected number of fixed points of the random gaussian spin glass:

$$F_n = 2^n \text{pr}(X_i > 0, 1 \leq i \leq n) = 2^n p_n \quad (9)$$

Here is another way of getting zero-mean gaussians of the same covariances (Eqs. (7) and (8)). Let Y_0, Y_1, \dots, Y_n be $n+1$ independent mean-0 variance-1 gaussians. Let

$$S_i = \sqrt{n-2} Y_i - Y_0, 1 \leq i \leq n \quad (10)$$

These are, of course, zero-mean joint gaussians (for $n > 2$). What are their covariances?

We have

$$E(S_i^2) = (n-2)E(Y_i^2) + 1$$

the cross-terms $\sqrt{n-2} E(Y_i Y_0)$ vanishing. Thus

$$E(S_i^2) = n-1, 1 \leq i \leq n \quad (11)$$

Likewise,

$$E(S_i S_k) = 1, 1 \leq i, k \leq n, i \neq k \quad (12)$$

From Eqs. (7) and (8), the n gaussian random variables S_i have the same covariance matrix as the n gaussian random variables X_i , and so we may use them instead of the X_i to calculate F_n in Eq. (9). This we do in the next section.

V. Exact Answer

From Eq. (9), we want

$$F_n = 2^n \text{pr}(S_i > 0, 1 \leq i \leq n) = 2^n p_n \quad (13)$$

From Eq. (10), this can be written (for $n > 2$) as

$$F_n = 2^n \text{pr}(Y_i > Y_0/\sqrt{n-2}, 1 \leq i \leq n) \quad (14)$$

Since the $n+1$ Y_i 's are jointly independent, the n random variables Y_1, Y_2, \dots, Y_n are *conditionally* independent given that $Y_0 = y$. So

$$\text{pr}(Y_1 > Y_0/\sqrt{n-2}, 1 \leq i \leq n | Y_0 = y) = [Q(y/\sqrt{n-2})]^n \quad (15)$$

by the conditional independence. Here Q is the righthand tail of the standard gaussian:

$$Q(z) = \frac{1}{\sqrt{2\pi}} \int_{t=z}^{\infty} e^{-t^2/2} dt \quad (16)$$

We can average Eq. (15) over y , which has the standard gaussian distribution, to get the unconditional probability that $Y_i \geq Y_0/\sqrt{n-2}, 1 \leq i \leq n$. The result (for $n > 2$) is

$$p_n = \text{pr}(Y_i \geq Y_0/\sqrt{n-2}, 1 \leq i \leq n)$$

$$= \frac{1}{\sqrt{2\pi}} \int_{t=-\infty}^{\infty} Q^n\left(\frac{t}{\sqrt{n-2}}\right) e^{-t^2/2} dt$$

$$= - \int_{t=-\infty}^{\infty} Q^n \left(\frac{t}{\sqrt{n-2}} \right) dQ(t) \quad (17)$$

This is exact. For $n = 2$, $p_n = 1/2 = \text{pr}(J_{12} > 0, J_{21} > 0)$. (This follows without this method.) We conclude that for $n = 2$, there are $F_2 = 2$ fixed points on the average. For $n = 3$, we use $n - 2 = 1$ in Eq. (17) to find

$$p_3 = - \int_{t=-\infty}^{\infty} Q^3(t) dQ(t) = - \frac{Q^4(t)}{4} \Big|_{t=-\infty}^{\infty} = \frac{1}{4}$$

Thus $F_3 = 8p_3 = 2$ also.

VI. Asymptotic Form

To find F_n for large n , let

$$s = t/\sqrt{n-2}$$

in Eq. (17). The result is

$$p_n = \sqrt{\frac{n-2}{2\pi}} \int_{s=-\infty}^{\infty} Q^n(s) e^{-(n-2)s^2/2} ds$$

or

$$p_n = \sqrt{\frac{n-2}{2\pi}} \int_{s=-\infty}^{\infty} \left[Q(s) e^{-s^2/2} \right]^n e^{s^2} ds \quad (18)$$

This is now in a form ripe for the Laplace or saddle point method (Ref. 5, Sec. 4.2). The method implies that for suitable functions g and h (the conditions are satisfied here with $e^{h(s)} = Q(s) e^{-s^2/2}$, $g(s) = e^{s^2}$), we have the asymptotic equality

$$\int_{s=-\infty}^{\infty} e^{nh(s)} g(s) ds \sim \frac{e^{nh(s_0)}}{\sqrt{n}} \left\{ g(s_0) \sqrt{\frac{2\pi}{-h''(s_0)}} \right\} \quad (19)$$

In this,

$$s_0 = \arg \max h(s) = \arg \max e^{h(s)} \quad (20)$$

is to be the unique maximum, and we must have

$$h''(s_0) < 0 \quad (21)$$

conditions which we will check shortly.

Assuming all this, we have from Eqs. (18) and (19) for n large

$$p_n \sim \left(Q(s_0) e^{-s_0^2/2} \right)^n e^{s_0^2} \left/ \left(\frac{d^2}{ds^2} (\log Q(s) e^{-s^2/2}) \right) \right|_{s=s_0} \quad (22)$$

The next section shows

$$s_0 = -0.50605$$

$$Q(s_0) e^{-s_0^2/2} = 0.61023$$

$$e^{s_0^2} \left/ \left(\frac{d^2}{ds^2} (\log Q(s) e^{-s^2/2}) \right) \right|_{s=s_0} = 1.0505$$

and so

$$p_n \sim (1.0505)(0.6102)^n \quad (23)$$

From Eq. (13), then,

$$F_n = 2^n p_n \sim (1.0505)(1.22046)^n$$

or, in more familiar form,

$$F_n \sim (1.0505)2^{0.2874n} \quad (24)$$

the desired result.

We note here that the exponent in Eq. (24) agrees with that in Ref. 1. But Ref. 1 had at best a logarithmically asymptotic answer, not a true asymptotic one, due both to minor errors and to the method there. However, Ref. 6 has the correct result, although the constants are not specifically worked out and the proof is much more complicated (although more generalizable) than the one given here.

VII. Calculation Details

This last section gives some details of the calculations of the preceding section and presents numerical results based on evaluating Eq. (18). Note that there is a finite maximum for h because $Q(-\infty) = 1$, $Q(\infty) = 0$, and $h(\pm\infty) = -\infty$. We have

$$h(s) = \log Q(s) - s^2/2, g(s) = s^2 \quad (25)$$

Thus,

$$h'(s) = -\frac{Z(s)}{Q(s)} - s \quad (26)$$

where

$$Z(s) = \frac{1}{\sqrt{2\pi}} e^{-s^2/2} \quad (27)$$

is the unit normal density function. From Eq. (26),

$$h''(s) = \frac{sZ(s)}{Q(s)} - \left(\frac{Z(s)}{Q(s)}\right)^2 - 1 \quad (28)$$

To search for a maximum of $h(s)$, we set $h'(s)$ equal to 0 in Eq. (26) to obtain

$$\frac{Z(s_0)}{Q(s_0)} + s_0 = 0 \quad (29)$$

as the condition on s_0 for the derivative $h'(s_0)$ to be zero. We see then that $s_0 < 0$. Continuing, from Eq. (28),

$$h''(s_0) = -2s_0^2 - 1 < 0 \quad (30)$$

Thus, there is a *unique* maximum on $(-\infty, \infty)$ and the conditions for Eq. (19) to hold are satisfied.

From Eqs. (18), (19), (29), (30), and (9), we now know that

$$F_n \sim \left(\sqrt{\frac{2}{\pi}} e^{-s_0^2/2} / |s_0| \right)^n e^{s_0^2/2} / \sqrt{1 + 2s_0^2} \quad (31)$$

So, what is s_0 ? Calculations on a good personal computer show

$$\begin{aligned} s_0 &= -0.50605 \\ \sqrt{\frac{2}{\pi}} e^{-s_0^2/2} / |s_0| &= 1.22046 \\ e^{s_0^2/2} / \sqrt{1 + 2s_0^2} &= 1.0505 \end{aligned} \quad (32)$$

This completes the derivation of Eq. (24).

Table 1 gives comparisons between a numerical evaluation of exact expression Eq. (17) times 2^n with the asymptotic formula Eq. (24). For $n = 2$ and 3, the exact answers, as we saw in Sec. V, are both 2. Note how good the asymptotic values for the expected number of fixed points are even for small n .

References

1. Tanaka, F., and Edwards, S. F., "Analytic theory of the ground state properties of a spin glass: I. Ising spin glass," *Jour. Phys. F: Metal Physics*, Vol. 10 (1980), pp. 2769-2778.
2. Hopfield, J. J., "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat'l. Acad. Sci. USA*, Vol. 79 (1982), pp. 2554-2558.
3. McEliece, R. J., Posner, E. C., Rodemich, E. R., and Venkatesh, S., "The Capacity of the Hopfield Associative Memory," to be submitted to the *IEEE Trans. on Information Theory*, 1985.
4. Pierce, J. R., and Posner, E. C., *Introduction to Communication Science and Systems*, Plenum, New York, 1980.
5. De Bruijn, N. G., *Asymptotic Methods in Analysis*, Second Edition, North-Holland, Amsterdam, 1961.
6. Gross, D. J., and Mezard, M., "The Simplest Spin Glass," *Nuclear Physics*, Vol. B 246 [FS12] (1984), pp. 431-452.

Table 1. Comparison of asymptotic and exact values of F_n

F_n		
n	Exact	Asymptotic
2	2	1.56
3	2	1.91
4	2.40	2.33
5	2.90	2.84
6	3.53	3.47
7	4.29	4.24
8	5.23	5.17
9	6.37	6.31
10	7.76	7.70
20	56.69	56.48
100	4.72E8	4.72E8

End of Document