The NIST 60-Millimeter Diameter Cylindrical Cavity Resonator: Performance Evaluation for Permittivity Measurements

Eric J. Vanzura
Richard G. Geyer
Michael D. Janezic

Electromagnetic Fields Division
Electronics and Electrical Engineering Laboratory
National Institute of Standards and Technology
325 Broadway
Boulder, Colorado 80303-3328

August 1993

U.S. DEPARTMENT OF COMMERCE, Ronald H. Brown, Secretary
TECHNOLOGY ADMINISTRATION, Mary L. Good, Under Secretary for Technology
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY, Arati Prabhakar, Director

National Institute of Standards and Technology Technical Note
Natl. Inst. Stand. Technol., Tech. Note 1354, 236 pages (August 1993)
CODEN:NTNOEF

U.S. GOVERNMENT PRINTING OFFICE
WASHINGTON: 1993

For sale by the Superintendent of Documents, U.S. Government Printing Office, Washington, DC 20402-9

Contents

List of Figures

List of Tables

The NIST 60-millimeter Diameter Cylindrical Cavity Resonator: Performance Evaluation for Permittivity Measurements

Eric J. Vanzura
Richard G. Geyer
Michael D. Janezic

Antenna and Materials Metrology Group
Electromagnetic Fields Division

Abstract

Uncertainty estimates are developed for dielectric permittivity calculations made using the NIST 60-mm diameter cylindrical resonator. A mode-filtering helical waveguide makes up the cavity's cylindrical wall, which permits the generation of high-purity TE_{01p} resonant modes for high accuracy permittivity measurements. The cavity's length can be varied from 408 to 433 mm. Fixed-length and fixed-frequency measurements in the X-band frequency range are evaluated with particular emphasis on 10 GHz. Resonator theory and design, measurement tolerances, and software are included..

Keywords: cylindrical cavity, dielectric permittivity, measurement software, resonator theory and design, TE_{01} helical waveguide, uncertainty estimates, x-band 10 GHz microwave.

Chapter 1

Introduction

1.1 Definition of Permittivity

Permittivity is a physical quantity that describes the relation between electric field $\overline{E}$ in volts per meter and electric displacement $\overline{D}$ in coulombs per square meter. In a vacuum, this relation is given by

$$\overline{D} = \epsilon_0 \overline{E} \; , \tag{1.1}$$

where $\epsilon_0 = 8.854\,187\,82 \times 10^{-12}$ farads per meter is the permittivity of free space. In matter, electric displacement is affected by the material's electric susceptibility, χ_e^*, which is a measure of the material's polarizability. In an isotropic, linear media the polarization vector $\overline{P}$ is parallel and proportional to the electric field,

$$\overline{P} = \epsilon_0 \chi_e^* \overline{E} \; , \tag{1.2}$$

and the electric displacement vector then becomes

$$\begin{aligned} \overline{D} &= \epsilon_0 \overline{E} + \overline{P} \\ &= \epsilon_0 (1 + \chi_e^*) \overline{E} \\ &= \epsilon^* \overline{E} \; , \end{aligned} \tag{1.3}$$

where

$$\begin{aligned} \epsilon^* &= \epsilon_0 (1 + \chi_e^*) \; , \\ &= \epsilon_0 \epsilon_R^* \end{aligned} \tag{1.4}$$

is the permittivity of the material, and $\epsilon_R^* = (1 + \chi_e^*)$ is the material's relative permittivity or dielectric constant. In general, electric susceptibility has real (i.e., dispersive) and imaginary (i.e., absorptive) parts which makes relative permittivity a complex quantity. We define relative permittivity as

$$\begin{aligned} \epsilon_R^* &= 1+\chi_e^* \\ &= (1+\Re\{\chi_e^*\}) - j\Im\{\chi_e^*\} \\ &= \epsilon_R' - j\epsilon_R'' \,. \end{aligned} \tag{1.5}$$

For low-loss materials ϵ_R' is often loosely called the relative permittivity, with the distinction between ϵ_R^* and ϵ_R' implicitly understood. The term ϵ_R'' is called the dielectric loss factor.

If the induced conduction current is directly proportional to the electric field, the curl of the magnetic field $\overline{H}$ is

$$\begin{aligned} \nabla \times \overline{H} &= (j\omega\epsilon^* + \sigma)\overline{E} \,, \\ &= j\omega\epsilon_0\epsilon_R' \left(1 - j\frac{\epsilon_R''}{\epsilon_R'} - j\frac{\sigma}{\omega\epsilon_0\epsilon_R'}\right)\overline{E} \,, \\ &= j\omega\epsilon_0\epsilon_R'(1 - j\tan\delta)\overline{E} \,. \end{aligned} \tag{1.6}$$

This relation for loss tangent $\tan\delta$ now explicitly separates dielectric and conductive losses. In most discussions dielectric loss of nonconductive materials at microwave and millimeter wave frequencies is described by a frequency dependent conductivity of the material at the measurement frequency. In this case loss tangent $\epsilon_R'' = \sigma/\omega\epsilon_0$, and loss tangent becomes

$$\tan\delta = \frac{\epsilon_R''}{\epsilon_R'} = \frac{\sigma}{\omega\epsilon_0\epsilon_R'} \,. \tag{1.7}$$

1.2 Operation of Mode-Filtered Cylindrical Resonators

The cylindrical cavity resonator described in this Technical Note was built by the NIST Electromagnetic Properties of Materials program to accurately measure the complex permittivity of dielectric materials. The application of cavity resonator methods to measure the complex permittivity of dielectric samples is not new [1]. However, resonator methods are inherently accurate and to this day prove to be the most accurate measurement method available. NIST chose to build a cavity resonator to accurately characterize dielectric materials that in turn can be used as check standards to test the validity of other measurement methods. Special test services are performed regularly for those outside NIST, and, since its introduction, the NIST resonator has characterized a large number of materials both for internal and external use.

Cavity resonators can measure permittivity only at fixed-frequencies and over a narrow range of frequency. This is the case especially for the shorter, solid-walled cavities because higher-order modes occur very easily. Measurement of permittivity over a broader range of frequencies is partially solved by the use of helical waveguide cavities [2, 3]. The mode filtering capability of helical waveguides allows construction of cavity resonators that can support several highly pure modes in a limited frequency range.

The NIST resonator has a 60-mm diameter helical wire-wound cylindrical wall, with a variable length of approximately 408 to 433 mm. The cavity's helical winding allows only circumferential currents to flow, which permits propagation of only the TE_{0n} family of circular transmission line modes, where n describes the number of radial variations. Mode-filtering capability permits the use of a long cavity that would otherwise be highly overmoded. This allows permittivity measurements at several TE_{01} resonant frequencies throughout X-band (8.2 to 12.4 GHz) rather than at only one or two specific frequencies.

The well defined geometry of the NIST cylindrical resonator allows standing waves to occur at certain frequencies. Power enters through a coupling iris located in one of the cavity's endplates and excites the TE_{01} waveguide mode. The present coupling endplate connects to X-band waveguide feeds. The cylindrical TE_{01} waveguide mode has an easily calculable wavelength that depends on waveguide diameter and the permittivity of the medium inside the waveguide (7.1). Power propagates in this mode from the coupling endplate to the other endplate and reflects back. At certain frequencies the cavity length will be an integer number p of half-wavelengths, resulting in constructive interference. The integer p is called the axial mode number. At these frequencies a standing wave is set up, and a resonance condition occurs. For the NIST cavity, the first TE_{01} resonant mode occurs near 6.1 GHz, and the first TE_{02} resonant mode occurs near 11.2 GHz.

The NIST cavity resonator is ideally suited for accurate permittivity measurements because the resonator's quality factor Q is very high (60 000 to 80 000). Q is sometimes defined as the number of radians a wave propagates before the wave's power decays to e^{-1}, or approximately 37%, of its initial power. For a high-Q resonator this means that the excitation wave reflects off the cavity endplates and travels the length of the cavity several times before the wave's power is significantly reduced. For example, if $p = 25$ and the Q of the $TE_{01(25)}$ mode is 80 000, then an excitation wave travels the length of the cavity approximately $Q/(25\pi) \approx 1000$ times before decaying to e^{-1} of its initial power. If this high-Q cavity contains a sample, the excitation wave also passes through the material more than 1000 times. In effect this increases the effective thickness of the sample and makes cavity resonator measurements sensitive to small changes in ϵ'_R and $\tan\delta$.

1.3 Methods of Measurement

To determine the permittivity of a material, two measurement methods can be used, as shown graphically in Fig. 1.1. One method involves reducing the sample-loaded cavity's length so that it resonates at the same frequency as the empty cavity. The change in length is a measure of permittivity, and the change in Q is a measure of the sample's dielectric loss (Sec. 2.2.3). The second method holds the cavity length fixed. The change in resonant frequency determines ϵ'_R, and the change in Q determines dielectric loss.

The theory from which ϵ'_R is calculated assumes that the cavity and sample have zero loss (infinite Q), so field orthogonality can be assumed. In practice, this assumption causes little problem because of the cavity's low surface impedance and because dielectric samples are usually low loss. The cavity resonator is so sensitive, however, that it is possible to determine and make corrections to frequency measurements that improve ϵ'_R measurement accuracy. These corrections are discussed in Chapter 4. In a sense, the concept of having two methods to measure the real part of permittivity is artificial. All that is needed to calculate ϵ'_R are cavity length and diameter, sample thickness and axial mode number. The empty cavity needs to be measured only because we have to determine the cavity's dimensions, where cavity length and diameter are calculated from the frequencies and axial mode numbers p of the TE_{01p} resonance spectrum.

The fixed-frequency method is more accurate than the fixed-length method for calculating dielectric loss. Cavity Q is frequency dependent. Skin depth (resistive losses) of the cavity walls decrease and coupling port losses increase as frequency increases. Presently, the measured empty-cavity Q is used as the reference that determines the cavity's conductor and port losses. When we measure the Q of the empty cavity, we determine the cavity loss at that resonant frequency. When a sample is inserted the resonant frequency is lowered. If we use the Q of the empty cavity resonating at a different frequency as our reference, our loss tangent calculation will be in error. This is especially true when the sample is very low loss ($\tan\delta < 0.0003$), because small differences in cavity Q become important. When sample loss is greater than $\tan\delta > 0.0008$ the differences between empty-cavity losses and the cavity losses of the sample-loaded cavity are small enough that the fixed-length method can be used with nearly the same accuracy as the fixed-frequency method.

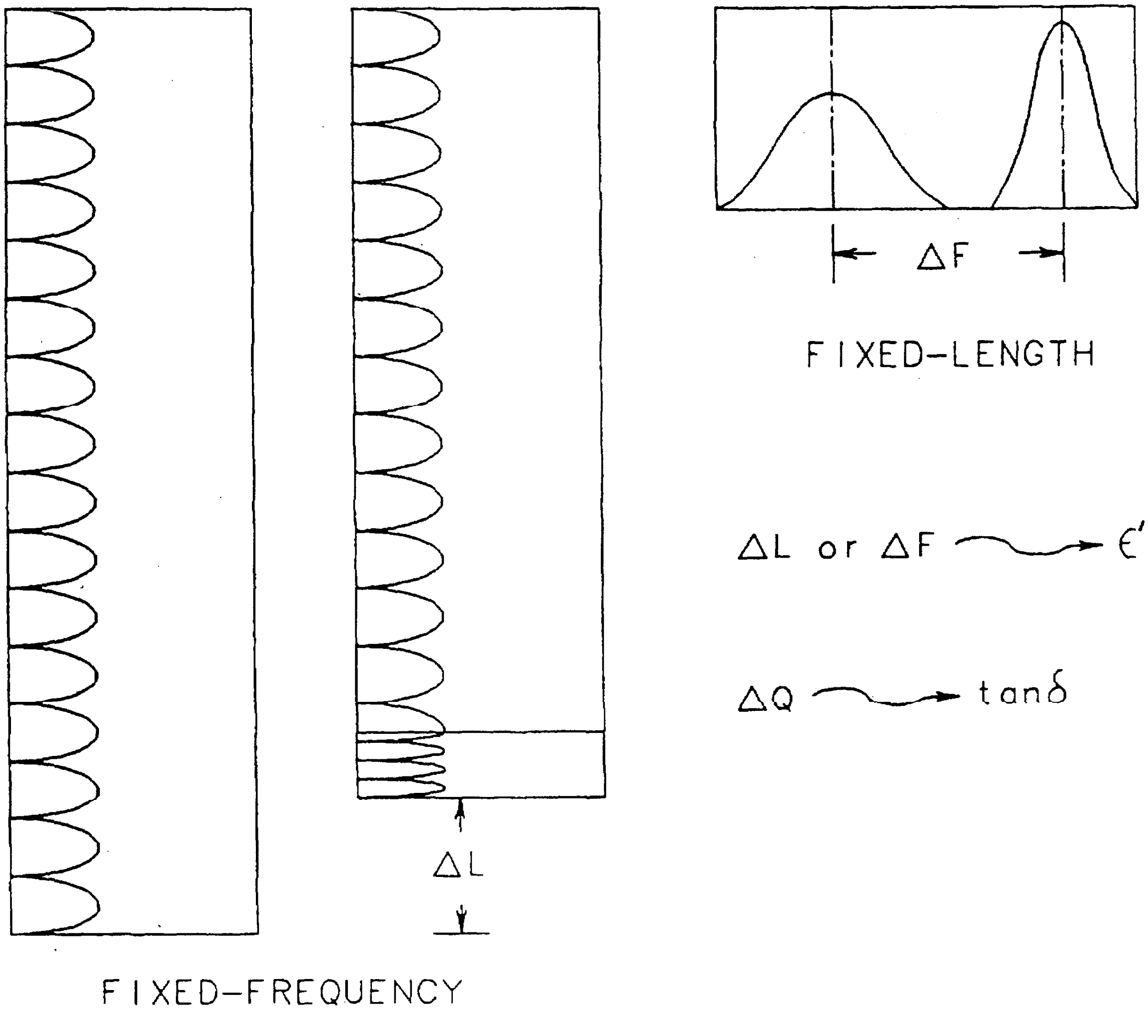


Figure 1.1: The fixed-frequency/variable-length and fixed-length/variable-frequency permittivity measurement methods rely on the empty-cavity dimensions and Q as a reference.

Chapter 2

Theory of Microwave Resonators

2.1 Introduction

Resonators create, filter, and select frequencies in oscillators, amplifiers, and tuners. Recent advances in the miniaturization of microwave circuits have spawned the development of low-loss, temperature-stable dielectric resonators. These resonators replace waveguide filters in many communication systems where microstrip and stripline resonators cannot be used due to their intrinsic high loss. Dielectric resonators are also very important fixtures for measuring the electrical properties of low-loss solids in the microwave region. An important resonator circuit at microwave frequencies is the closed cylindrical cavity operated in a resonant TE_{01p} mode. The magnetic field for this mode is sketched in Fig. 2.1. For a distant observer this mode appears as a magnetic dipole, and for this reason some authors call it a "magnetic dipole mode." The electric field lines are simply circles concentric with the axis of the cylinder (Fig. 2.2).

For a TE mode in a waveguide, the electric field is everywhere *transverse* to the propagation direction or to the axial z-direction of the guide. Coordinate systems for circular and rectangular waveguides are sketched in Figs. 2.3 and 2.4, respectively. For a TE_{mn} mode in a rectangular guide, m simply represents the number of antinodes occurring in the electric field E in the x-direction and n represents the number of antinodes in E in the y-direction. For a TE_{mnp} mode in a finite circular guide, m represents the number of antinodes occurring in E in the ϕ-direction, n the number of antinodes in the radial r-direction, and p the number in the z-direction. The same notation is used for transverse magnetic or TM_{mn} modes, with the m and n indices representing the antinodes occurring

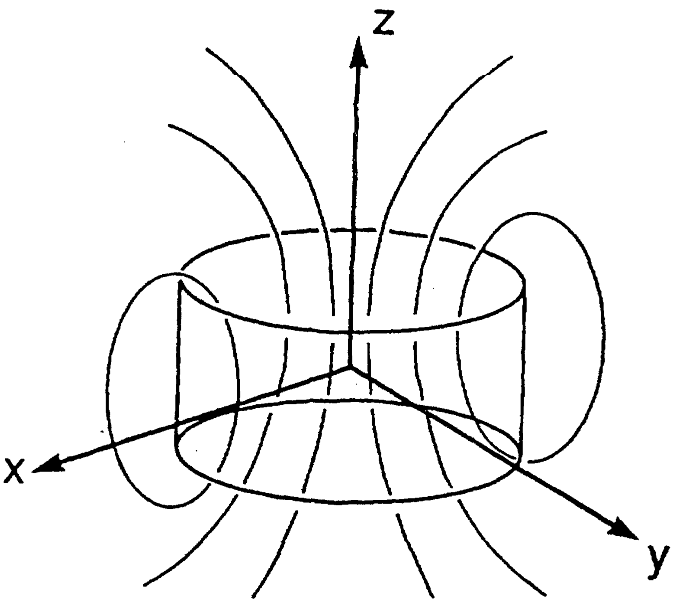


Figure 2.1: Magnetic field lines of the resonant mode TE_{01p} in an isolated dielectric resonator.

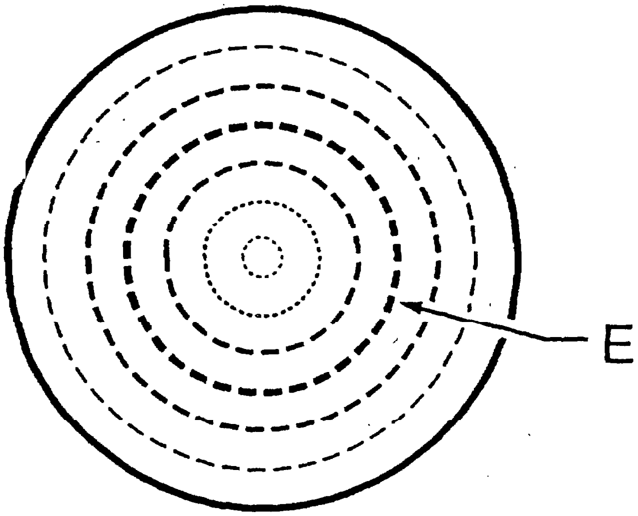


Figure 2.2: Electric field distribution in equatorial plane for TE_{01p} mode. Breadth of dashed lines is proportional to the electric field.

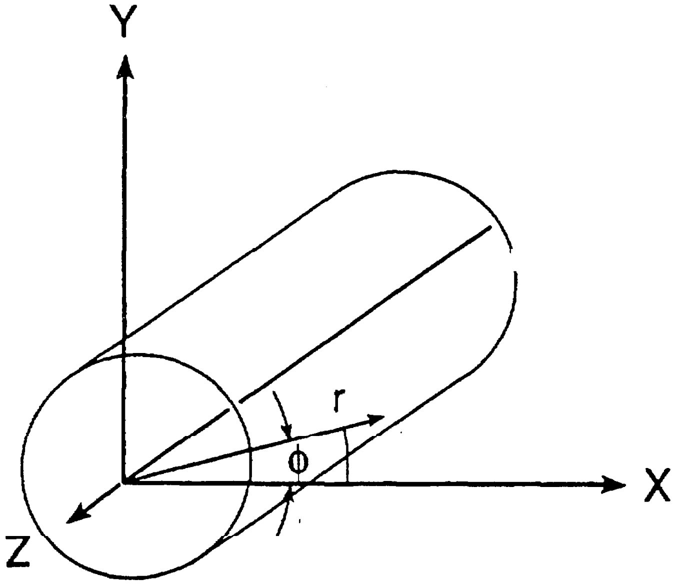


Figure 2.3: Cylindrical coordinate system for waveguides.

in the magnetic field H for a given coordinate direction. Generally, the choice of cavity to be used in dielectric metrology is influenced by the shape and size of the sample to be measured. Insertion and removal of the sample from the cavity should be convenient and not significantly alter the Q [4] of the air-filled cavity or increase the radiation from the cavity. Hollow circular cylindrical waveguide resonators, which are terminated by two short-circuit endplates, are the most commonly used for dielectric measurements of low-loss solids. The reasons for this are the relative ease of fabrication and the very high Q-factors and accompanying narrow bandwidths that can be obtained with this fixture in the microwave region [1]. If the cavity is made from helically wound waveguide, it also acts as a mode filter in that all other waveguide modes except the TE_{01p} mode [3] are greatly attenuated.

In general, microwave energy is coupled to the cavity through transmission line probes, as illustrated in Fig. 2.5. As in resonant transmission lines where resonance occurs at many frequencies, the hollow cylindrical waveguide has many resonance frequencies and associated field distributions or modes. The field of the mode with the lowest or dominant resonant frequency is termed the dominant mode of that fixture. Only enough energy is provided by the input excitation to match cavity losses. Once the internal fields at a resonant frequency are determined, an equivalent lumped parameter circuit of the fixture may be ascertained. In addition, the internal power dissipation, stored energies, and energy flow out of the cavity can be determined.

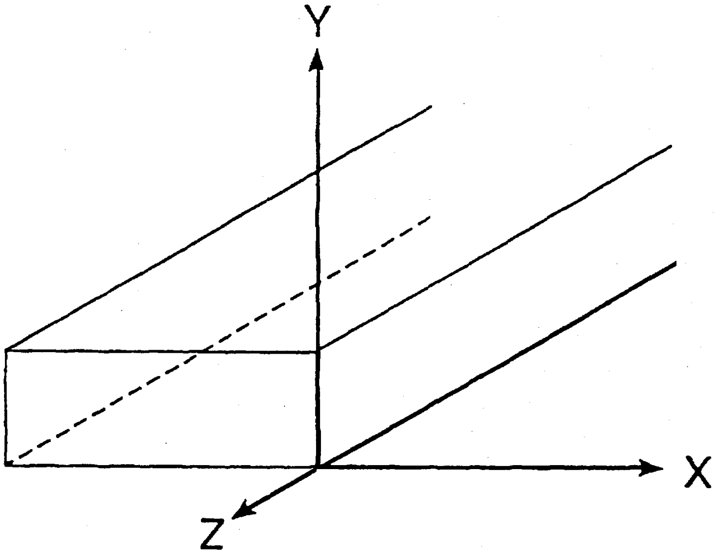


Figure 2.4: Rectangular coordinate system for waveguides.

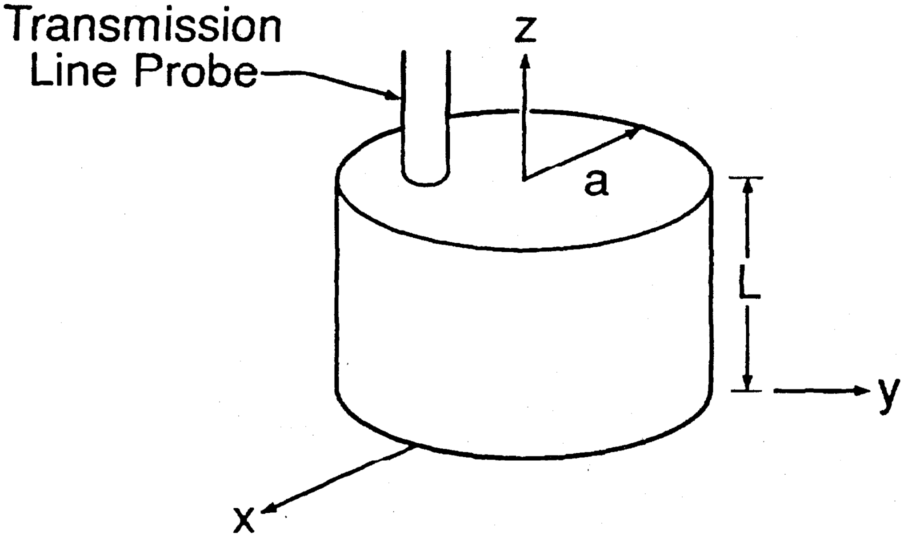


Figure 2.5: Circular cylindrical hollow cavity of length L and radius a.

As expected, the different microwave modes that can be set up in a hollow right circular cylinder depend on both the dimension of the cylinder and the microwave frequency. A hollow right circular cylindrical cavity is supportive of both TE- and TM-mode field structures. For steady-state $\exp(j\omega t)$ field time dependence, the equations for the axial components of the electric and magnetic field interior to a waveguide of arbitrary cross-section and filled with material of complex permittivity ϵ^* and complex permeability μ^* satisfy the homogeneous Helmholtz equation

$$\nabla^2 \begin{pmatrix} E_z \\ H_z \end{pmatrix} + k^2 \begin{pmatrix} E_z \\ H_z \end{pmatrix} = 0 , \tag{2.1}$$

where z is the direction of propagation and E_z, H_z are taken for TM or TE modes, respectively. The differential operator ∇^2 is the Laplacian, which can be expressed in any curvilinear coordinate system by evaluation of the metric. All other field components can be expressed in terms of E_z and H_z. In general, any field in a hollow waveguide, however complicated, may be represented by a combination of TE and TM modes. The wavenumbers

$$k^2 = k_c^2 - \gamma^2 = \omega^2 \epsilon^* \mu^* , \tag{2.2}$$

are the characteristic eigenvalues of (2.1). In other words, to each value of k^2 there will correspond a function E_z or H_z (for TM or TE modes) which is a *characteristic* function from which may be derived the other components of the fields. In general ϵ^* and μ^* in (2.2) can be complex dyadics, but for this discussion the material is assumed isotropic and each dyadic becomes a single complex value multiplied by the unit dyadic. As defined in (1.7), the conductivity of the medium is contained in the imaginary part of the dielectric constant, $\epsilon^* = \epsilon' - j\epsilon''$. For a plane-wave field $H_z(z) = H_z e^{-\gamma z}$ in which the electric field is entirely transverse (TE waves),

$$\gamma^2 = \left(\frac{2\pi}{\lambda}\right)^2 = k_c^2 - \omega^2 \epsilon^* \mu^* , \tag{2.3}$$

where the value of λ is the wavelength of a plane TE wave in the medium that fills the hollow pipe and k_c is the cutoff wavenumber of the TE mode. In order to have propagation down the pipe, γ^2 must be negative. The quantity k_c^2 is always real because, for a circular waveguide of radius a,

$$(k_c)_{mn} = \frac{t'_{mn}}{a} , \tag{2.4}$$

where t'_{mn} is the nth root of the first derivative of the Bessel function $J'_m(k_c a) = 0$ for a TE_{mn} mode. When γ^2 is positive or $\Re\left(\omega^2 \epsilon^* \mu^*\right) \leq k_c^2$ there is no propagation of energy through the waveguide. This condition is termed cutoff. The cutoff or critical frequency is given by

$$\omega_c^2 = \Re\left\{\frac{1}{\epsilon^*\mu^*}\right\} k_c^2 \,. \tag{2.5}$$

Substituting (2.4) into (2.5) gives the cutoff frequency for TE_{mn} modes in a circular waveguide:

$$(f_c)^2_{TE_{mn}} = \left(\frac{1}{2\pi}\right)^2 \Re\left\{\frac{1}{\epsilon^*\mu^*}\right\} \left(\frac{t'_{mn}}{a}\right)^2 , \tag{2.6}$$

or

$$(\lambda_c)_{TE_{mn}} = \frac{c}{(f_c)_{TE_{mn}}} = \frac{c \cdot 2\pi a \cdot \sqrt{\Re\left(\epsilon^*\mu^*\right)}}{t'_{mn}} , \tag{2.7}$$

where c is the velocity of light in the medium.

Similarly, for a TM_{mn} mode in circular waveguide,

$$(k_c)_{mn} = \frac{t_{mn}}{a} , \tag{2.8}$$

where t_{mn} is the nth root of $J_m(k_c a) = 0$.

As before, the cutoff wavelength for TM_{mn} modes is given by:

$$(\lambda_c)_{TM_{mn}} = \frac{c \cdot 2\pi a \cdot \sqrt{\Re\left(\epsilon^*\mu^*\right)}}{t_{mn}} \tag{2.9}$$

If λ_g is the wavelength in the guide and k_g the guide wavenumber, then

$$\gamma = jk_g = j\frac{2\pi}{\lambda_g} , \tag{2.10}$$

or, from (2.3),

$$\left(\frac{2\pi}{\lambda_g}\right)^2 = \left(\frac{2\pi}{\lambda}\right)^2 - \left(\frac{2\pi}{\lambda_c}\right)^2 , \tag{2.11}$$

from which we derive the axial wavelength in the waveguide,

$$\lambda_g = \frac{\lambda}{\sqrt{1-\left(\frac{\lambda}{\lambda_c}\right)^2}} , \tag{2.12}$$

where λ is the wavelength of a uniform plane wave in the medium:

$$\begin{aligned} \lambda &= \frac{\lambda_0}{\sqrt{\frac{\Re\{\epsilon^*\mu^*\}}{\epsilon_0\mu_0}}} \\ &\approx \frac{\lambda_0}{\sqrt{\frac{\epsilon'\mu'}{\epsilon_0\mu_0}}} \end{aligned} \tag{2.13}$$

where λ_0 is the free-space wavelength and we assume that $\Re\{\epsilon^*\mu^*\} \approx \epsilon'\mu'$. Equation (2.12) then becomes

$$\lambda_g = \frac{\lambda_0}{\sqrt{\frac{\epsilon'\mu'}{\epsilon_0\mu_0} - \left(\frac{\lambda_0}{\lambda_c}\right)^2}} = \frac{\lambda_0}{\sqrt{\epsilon'_R\mu'_R - \left(\frac{\lambda_0}{\lambda_c}\right)^2}} . \tag{2.14}$$

Equation (2.14) shows that when the free-space wavelength is much less than the cutoff wavelength, that waves propagate in the guide with a guide wavelength normalized by the square root of the product of relative permittivity and relative permeability of the material filling the guide. Of course, in the empty cavity situation, this means that the guide wavelength is essentially that of air. Similarly, when wavelengths are much greater than the cutoff wavelength for a given mode, the guide wavelength becomes pure imaginary or, stated physically, *nonpropagating* evanescent modes are set up.

2.1.1 Fields in a Right Circular Cylindrical Cavity

For the problem at hand, we are dealing with a cylindrical structure having two metallic end plates. In other words, it is bounded in axial extent and the associated internal fields may have variations in *both* the transverse and axial directions. In the case of TE waves, the wave equation (2.1) for H_z expressed in cylindrical coordinates becomes

$$\frac{1}{r}\frac{\partial}{\partial r}\left[r\frac{\partial}{\partial r}H_z\right] + \frac{1}{r^2}\frac{\partial^2}{\partial \phi^2}H_z + \frac{\partial^2}{\partial z^2}H_z + k^2 H_z = 0 , \tag{2.15}$$

where

$$k_c^2 = \gamma^2 + k^2 , \tag{2.16}$$

and from (1.7)

$$\begin{aligned} k^2 &= \omega^2\mu^*\epsilon^* = \omega^2\mu^*\left(\epsilon' - j\epsilon''\right) \\ &= \omega^2\mu^*\epsilon_0\epsilon'_R\left(1 - j\tan\delta\right) . \end{aligned} \tag{2.17}$$

The wavenumber is complex for lossy dielectric materials; that is,

$$k = k_{real} - j k_{imaginary} , \tag{2.18}$$

where it is easily shown that

$$k^2_{real} = \frac{\omega^2 \mu \epsilon}{2} \left[\sqrt{\left(\frac{\sigma}{\omega \epsilon}\right)^2 + 1} + 1 \right] , \tag{2.19}$$

and

$$k^2_{imaginary} = \frac{\omega^2 \mu \epsilon}{2} \left[\sqrt{\left(\frac{\sigma}{\omega \epsilon}\right)^2 + 1} - 1 \right] . \tag{2.20}$$

In (2.19) and (2.20) $\mu = \mu'_R \mu_0$ and $\epsilon = \epsilon'_R \epsilon_0$, where $\epsilon_0 \approx 8.854 \times 10^{-12}$ F/m and $\mu_0 = 4\pi \times 10^{-7}$ H/m. For lossless dielectrics, $\sigma = 0$ and $\epsilon''_R = 0$ with $k_{real} = \omega\sqrt{\mu\epsilon}$ and $k_{imaginary} = 0$.

The method of separation of variables [5] yields a solution to (2.15) of the form

$$H_z(r, \phi, z) = R(r)\Phi(\phi)Z(z) . \tag{2.21}$$

Substitution of (2.21) into (2.15) and then division by $H_z(r, \phi, z)$ yields

$$\frac{1}{rR}\frac{d}{dr}\left[r\frac{d}{dr}R\right] + \frac{1}{r^2\Phi}\frac{d^2\Phi}{d\phi^2} + \frac{1}{Z}\frac{d^2Z}{dz^2} + k^2 = 0 . \tag{2.22}$$

The third term in (2.22) is explicitly independent of r and ϕ. It is also necessarily independent of z if (2.22) is to sum to zero for all (r, ϕ, z). Therefore,

$$\frac{1}{Z}\frac{d^2Z}{dz^2} = -\beta^2 , \tag{2.23}$$

where β is a constant. Substitution of (2.23) into (2.22) and multiplication by r^2 results in

$$r\frac{1}{R}\frac{d}{dr}\left[r\frac{d}{dr}R\right] + \frac{1}{\Phi}\frac{d^2\Phi}{d\phi^2} + \left(k^2 - \beta^2\right) r^2 = 0 . \tag{2.24}$$

The second term in (2.24) is a function of ϕ only, whereas the rest of the equation is a function of r only. By the same argument the azimuthal component of H_z obeys the relation,

$$\frac{1}{\Phi}\frac{d^2\Phi}{d\phi^2} = -m^2 , \tag{2.25}$$

where m is a constant. Substitution of (2.25) into (2.24) and multiplication throughout by $R(r)$ yields

$$r\frac{d}{dr}\left[r\frac{d}{dr}R\right] + \left[\left(k^2 - \beta^2\right) r^2 - m^2\right] R = 0 , \tag{2.26}$$

where β is the axial waveguide propagation constant and k is the medium wavenumber. Now $k^2-\beta^2$ represents the square of the transverse radial wavenumber k_c, where

$$k_c = \sqrt{k^2 - \beta^2} \; , \tag{2.27}$$

so (2.26) can be written

$$r\frac{d}{dr}\left[r\frac{dR}{dr}\right] + \left[(k_c r)^2 - m^2\right] R = 0 \; . \tag{2.28}$$

The original Helmholtz equation (2.1) is now separated into three equations, each of which determines only one of the functions $R(r)$, $\Phi(\phi)$ or $Z(z)$. Equations (2.23) and (2.25), are harmonic equations, whose solutions are harmonic functions, or linear combinations of sines and cosines. Equation (2.28) is a Bessel equation of the mth order with independent solutions $J_m(k_c r)$ and $N_m(k_c r)$, where J_m and N_m represent Bessel and Newmann functions, respectively, of order m. Because $N_m(k_c r)$ is not finite at $r = 0$, the solution for $R(r)$ is

$$R(r) = J_m(k_c r) \; . \tag{2.29}$$

The choice of the constants β and m, as well as the solutions for (2.23) and (2.25), depends on the physical geometry of the fixture, conditions at the boundaries, and the type of field to be supported by the fixture. For nonevanescent modes in a right circular cylindrical cavity, k_c can take on only *characteristic* or certain discrete values that correspond to different modes of propagation. In the case of the cylindrical cavity, the constant m must be an integer if the solution for E_z is to be single-valued in ϕ (periodic). The radical that defines the transverse radial wavenumber k_c calls for some comment. The branch of the square root is usually chosen such that $k_c \rightarrow k$ as $|\beta^2| \rightarrow 0$ and $k_c \rightarrow |\beta|$ as $|k^2| \rightarrow 0$.

2.1.2 *TE* Modes

For *TE* mode structure ($E_z = 0$) we need solve for only H_z. All other field components interior to the cavity are derived from H_z by Maxwell's equations. The complete solution for $H_z(r, \phi, z)$ is

$$H_z(r, \phi, z) = J_m\left(k_c r\right)\left[A \cos m\phi + B \sin m\phi\right] \sin \beta z \; , \tag{2.30}$$

which satisfies the boundary conditions at the cavity's cylindrical walls $r = a$ and at the cavity endplate $z = 0$. This includes the case where the cavity is filled with dielectric materials such that the transverse wavenumber k_c is complex. A and B are constants in (2.30) which determine the phase of the azimuthal field

Table 2.1: Zeroes t'_{mn} of the first derivative of the Bessel function of first kind and order m.

m	n		
	1	2	3
0	3.8317	7.0156	10.1735
1	1.8412	5.3314	8.5363
2	3.0542	6.7061	9.9695
3	4.2012	8.0152	11.3459

orientation relative to the coupling port(s). We are currently considering only an empty cylindrical cavity with $k = \omega\sqrt{\mu_0\epsilon_0}$ and arbitrary azimuthal phase. In order to enforce the boundary condition on E_ϕ, derivable from H_z from Maxwell's equations, we must have

$$E_\phi(a, \phi, z) = 0 \ , \tag{2.31}$$

or

$$J'_m(k_c a) = 0 \ , \tag{2.32}$$

where J'_m represents the first derivative of the mth-order Bessel function. If the zeroes of J'_m are denoted by t'_{mn}, where $n = 1, 2, 3, \ldots$ represents the zero-crossing number, then k_c must be chosen to take only certain discrete values; that is,

$$k_c = \frac{t'_{mn}}{a} \ . \tag{2.33}$$

Table 2.1 gives some representative values for t'_{mn}.

The boundary condition on the azimuthal electric field at $z = L$ results in certain allowable values for the longitudinal propagation wavenumber:

$$\sin \beta L = 0 \ , \tag{2.34}$$

or

$$\beta = p\frac{\pi}{L} \ , \tag{2.35}$$

where $p = 1, 2, 3, \ldots$. The final result for the axial magnetic field of the TE mode within the cavity is:

$$H_z(TE_{mnp}) = J_m\left(\frac{t'_{mn}r}{a}\right)\left[A\cos m\phi + B\sin m\phi\right]\sin\left[\frac{p\pi z}{L}\right] \ , \tag{2.36}$$

where m, n, p are integers describing the TE mode. An identical analysis can be performed to derive the axial electric field for the TM case

The characteristic equation used to find the resonant frequencies for TE modes in the empty cavity is given by Harrington [5] pp. 213–216 as,

$$f_0\left(TE_{mnp}\right) = \frac{c_{air}}{2\pi}\left[\left(\frac{t'_{mn}}{a}\right)^2 + \left(\frac{p\pi}{L}\right)^2\right]^{\frac{1}{2}}, \tag{2.37}$$

where c_{air} is the speed of light under testing conditions in air. When $f = f_0(TE_{mnp})$, we have a TE-mode solution to Maxwell's equations. From (2.2), the resonant frequency for the cylindrical cavity filled with material of complex permittivity ϵ^* and real permeability μ' is

$$f_0\left(TE_{mnp}\right) = \frac{1}{2\pi\sqrt{\mu'\epsilon'\left(\tan^2\delta + 1\right)^{\frac{1}{2}}}}\left[\left(\frac{t'_{mn}}{a}\right)^2 + \left(\frac{p\pi}{L}\right)^2\right]^{\frac{1}{2}}. \tag{2.38}$$

As expected, when the length or diameter of a cylindrical cavity increases, the resonant frequency decreases for any given mode.

The figure of merit for assessing the performance or quality of a cavity resonator is the quality factor Q which is a measure or energy stored in the fields inside the resonator compared to the energy loss or dissipation per cycle. The Q-factor is defined by

$$\begin{aligned} Q &= 2\pi\frac{maximum\ energy\ stored\ during\ a\ cycle}{average\ energy\ dissipated\ per\ cycle}, \\ &= \frac{2\pi W}{PT} = \frac{\omega_0 W}{P}, \end{aligned} \tag{2.39}$$

where W is stored energy, P is power dissipation, ω_0 is resonant radian frequency, and T is period $= \frac{2\pi}{\omega_0}$. The higher the axial TE-mode number for any given azimuthal and radial mode number, the greater the cavity quality factor, for any given cavity diameter or length. The quality factor Q will be discussed in more detail in connection with the evaluation of dielectric loss measurements of materials. Some examples of normalized cavity Q values are shown in Fig. 2.6 for some representative TM_{mnp} modes and in Fig. 2.7 for some TE_{0np} modes as a function of cavity diameter-to-length ratio. In Fig. 2.7, the optimal (highest) Q value is obtained for TE_{mnp} modes when the diameter of the cavity equals its length. Our 60-mm diameter resonator's length is roughly seven times longer than its diameter. Resonance Q-value is compromised somewhat by this length, but the cavity's helical windings yield better than 30-dB mode purity. The length of the resonator allows for the measurement of several TE_{01p} modes in the X-band frequency range

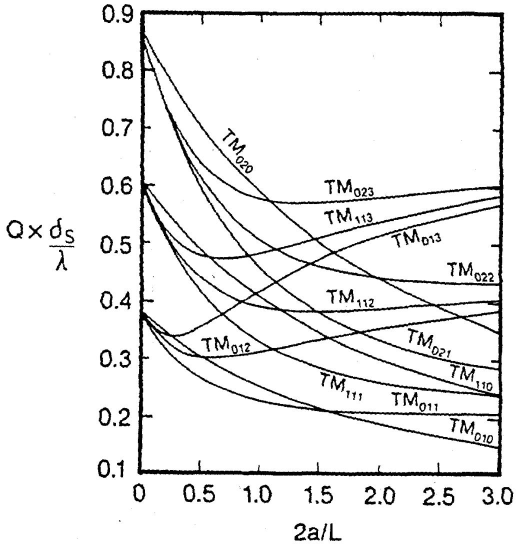


Figure 2.6: Normalized cavity Q versus diameter-to-length ratio for a right circular cylinder for TM_{mnp} modes. δ_s is the skin depth of the cavity wall material and λ is the free-space wavelength.

2.2 TE_{01p} Mode-Filter Cylindrical Cavity

One very useful cavity resonator for microwave dielectric property measurements on low-loss materials is constructed to filter all modes resulting from current other than that flowing circumferentially about the cavity wall. This yields a very high-Q cavity with a very pure TE-mode structure for precision electrical property measurements.

Consider, for example, the wall and endplate currents flowing in the cylindrical cavity for TE_{01p} and TM_{11p} modes as shown in Fig. 2.8. For the TM_{11p} mode there are both azimuthal and radial currents, whereas in the case of the TE_{01p} mode there are only azimuthal currents flowing in the wall and end plates. As Cook [2] notes, the presence of unwanted modes produces larger data scatter of loss tangent and permittivity with change in specimen length than when there is only one effective mode propagating, such as the TE_{01p}.

If the currents associated with the TM_{11p} mode can be interrupted while at the same time leaving those associated with the TE_{01p} mode unchanged, then signif-

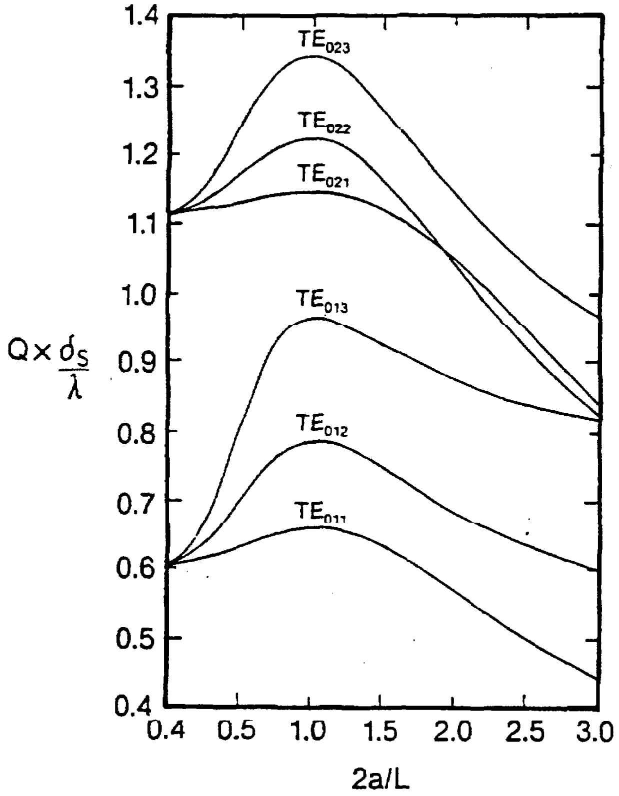


Figure 2.7: Normalized cavity Q versus diameter-to-length ratio for a right circular cylinder for TE_{0np} modes. δ_s is the skin depth of the cavity wall material and λ is the free-space wavelength.

icant attenuation of the TM_{11p} mode will take place. This can be accomplished by constructing a cavity wall in which the conductivity is discontinuous along the length of the cylinder but continuous around its circumference. This type of cavity becomes a mode filter. One approach to the construction of a mode-filter cavity is to construct a cylinder of annular copper rings electrically insulated from each other [2] to form a continuous cylinder. In this case, the currents flowing from ring to ring will be greatly impeded while those flowing around the rings are unimpeded.

Another approach is to make a helical waveguide by winding fine enameled wire into a precision cylindrical former (Fig. 2.9). This type of waveguide has been discussed in the literature by Morgan and Young [6]; Unger [7]; Young [8]; Cook

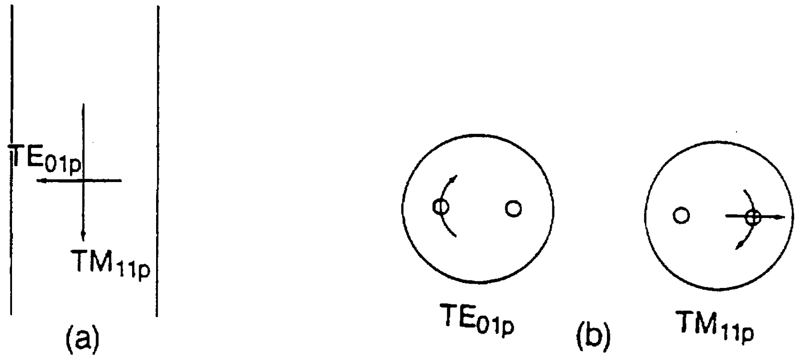


Figure 2.8: Currents flowing in (a) cylinder wall and (b) cavity end plates for TE_{01p} and TM_{11p} modes.

and Jones [9]; Cook [2]; Waldron [10] [11]; Waldron and Bowe [12]; Shimba [13]; Kazantsev, Kaznacheev and Meriakri [14]; Mikoshiba [15]; Piefke [16]; Waldron, Bowe, Wackrill and Wescott [17]; and Noda, Yamaguchi and Suzuki [18]. In a helical waveguide the modes whose wall currents follow the conducting helix possess attenuation constants which are essentially the same as for copper pipe. All other modes, however, have a high transmission loss. In other words, all modes other than the TE_{01p} have very large attenuation constants in a helical waveguide. The exact calculation of the attenuation constants for modes other than TE_{01p} depends on the helix pitch angle and the electrical properties of the thick lossy dielectric jacket surrounding the helix; the fields of those modes penetrate into the jacket and are attenuated, whereas that of the TE_{01p} mode does not and is therefore minimally attenuated. The helical waveguide, then, is the equivalent of an anisotropically conducting cylinder. Whether we use helically wound wire as opposed to annular copper rings is a question of convenience in manufacture.

Waldron [10], and Waldron and Bowe [12] have analyzed the azimuthal electric field dependence as a function of the radius of the helical winding, the complex permittivity of the jacket material, and the pitch angle of the helix. To a first-order approximation the TE_{01p} mode is independent of the pitch angle. Waldron and Bowe [12] also treat the effect of pitch for the TE_{01p} mode to the second order and find that the effect of a finite pitch does indeed cause a slight attenuation of the TE_{01p} mode. Cook [2] compares the attenuation of the TE_{01p} mode to other modes in a 50-mm diameter helix at 35 GHz; these results are given in Table 2.2. Note that the modes other than TE_{01p} have attenuation constants larger by several orders of magnitude than that for TE_{01p}. Piefke [16] demonstrates that

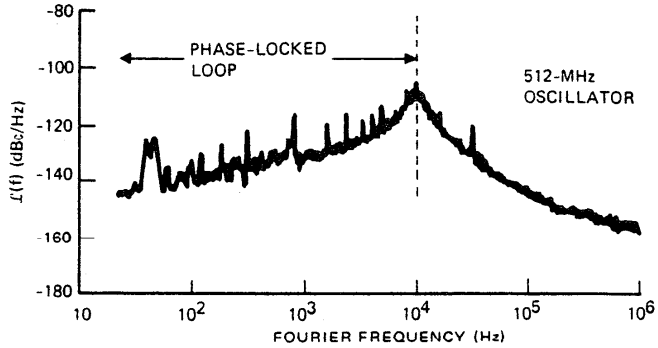


FIG. 11 Phase-locked-loop characteristics of the H.P. 8640B signal generator, showing the normalized phase-noise sideband power spectral density.

4. *Measurements, Calculations, and Data Plots*

The measurement sequence is automated except for the case where manual adjustments are required to maintain phase quadrature of the signals. After phase quadrature of the signals into the mixer is established, the IF attenuator is returned to the zero-decibel reference setting. This attenuator is set to a high value [assumed to be 50 dB in Eq. (65)] to prevent saturation of the spectrum analyzer during the calibration process.

The automated measurements are executed, and the direct measurement and data plot of $\mathscr{L}(f)$ is obtained in decibels (carrier) per hertz using the equation

$$\mathscr{L}(f) = -\,[\text{carrier power level} - (\text{noise power level} - 6 + 2.5 - 10\log B - 3)]. \tag{69}$$

The noise power (dBm) is measured relative to the carrier-power level (dBm), and the remaining terms of the equation represent corrections that must be applied because of the type of measurement and the characteristics of the measurement equipment, as follows.

(1) The measurement of noise sidebands with the signals in phase quadrature requires the −6-dB correction that is noted in Eq. (69).

(2) The nonlinearity of the spectrum analyzer's logarithmic IF amplifier results in compression of the noise peaks which, when average-detected, require the 2.5-dB correction.

(3) The bandwidth correction is required because the spectrum analyzer measurements of random or white noise are a function of the particular bandwidth used in the measurement.

Figure 8.5 gives ϵ'_R results for these three samples. Samples 1 and 2 yield nearly identical results, while sample 3 has a noticeably lower permittivity. Figure. 8.6 gives loss tangent results from these same three samples. All three samples have a noticeably different loss. and, surprisingly, the fused quartz sample has lower loss than the fused silica. We normally expect impurities to cause greater loss. The discontinuity in the results near 11.7 GHz are due to mode interference by TE_{02p} modes.

These results indicate that the permittivity and loss of high-purity, high homogeneity fused silica are consistent from manufacturer to manufacturer. However there are repeatable differences in permittivity and loss between the two fused silica samples. The fused-quartz sample has a noticeably lower permittivity and loss than the fused-silica samples.. This could be because the fused-quartz sample is less dense and has more seed crystals and inclusions, or because of impurities. Also, the fixed-frequency and fixed-length results for sample 2 are nearly identical.

Uncertainty estimates for sample number 1 are given in Figs. 8.7 and 8.8.

8.2.3 Alumina

Eleven samples of 99.9% pure Al_2O_3 are under evaluation. Samples 8–11 were made separately from the first seven samples, and permittivity results for samples 1–7 are given in Figs. 8.9 and 8.10. A demonstrable difference between samples from the same batch can be seen. This is not due to measurement variations because results for Sample 6 are repeatable. Although further investigation is needed, the higher permittivity results at the lower frequencies are most likely due to our having used an effective cavity length that was too long for those frequencies (Sec. 4.1). Loss tangent for this high-purity alumina is the lowest of any material we have measured. Also, the fixed-frequency method gives stable and repeatable loss tangent results for most of the frequency range, while the fixed-length method varies much more. This is due to the limitations in the loss tangent model which assumes an equivalence between the empty cavity and a cavity containing a hypothetical lossless sample of the same permittivity as the real sample (Sec. 4.2). The negative loss tangent values near 11.5 GHz are due to the interfering TE_{021} mode. Uncertainty estimates for samples with $\epsilon'_R \approx 10$ are given in Chapter 9, so no uncertainty results are given here.

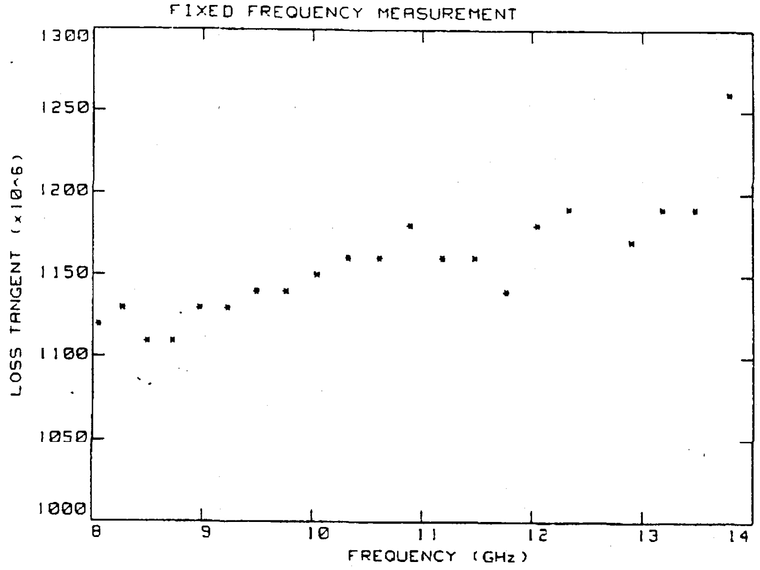


Figure 8.4: Loss tangent tanδ of cross-linked polystyrene.

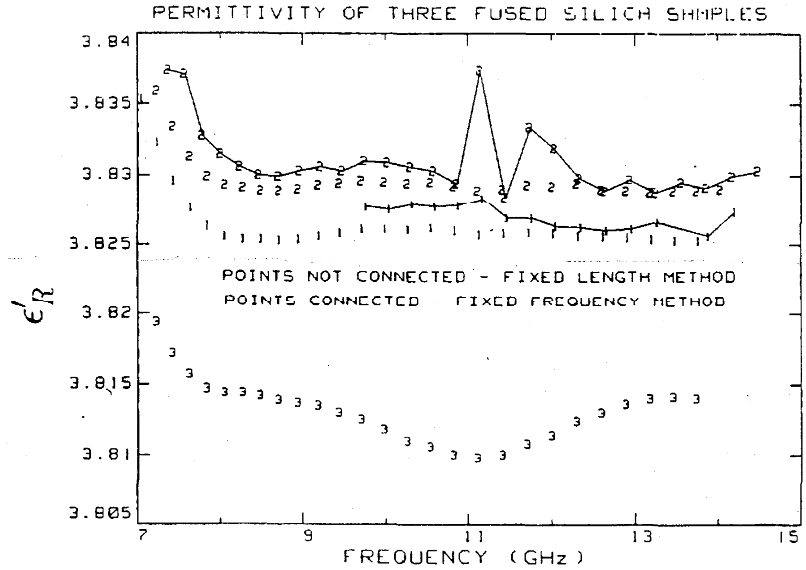


Figure 8.5: Permittivity ϵ'_R of two fused silica samples (1 and 2) and one fused quartz sample (3).

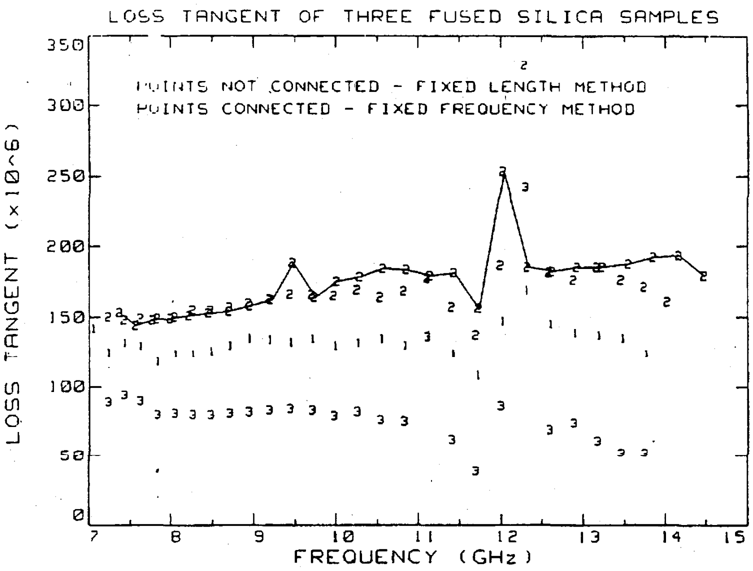


Figure 8.6: Loss tangent tanδ of two fused silica samples (1 and 2) and one fused quartz sample (3). Note effect of interference by a TE_{02} mode between 11 and 13 GHz.

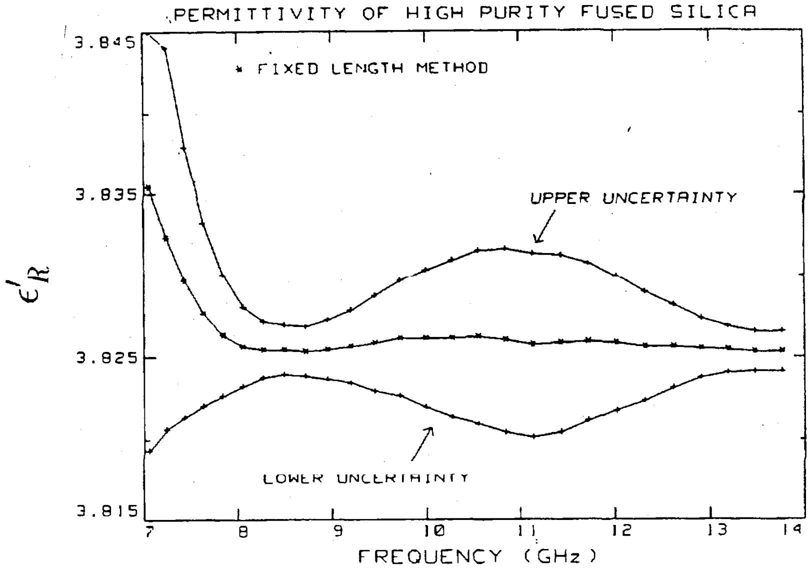


Figure 8.7: Permittivity ϵ'_R and estimated uncertainty of fused silica sample # 1.

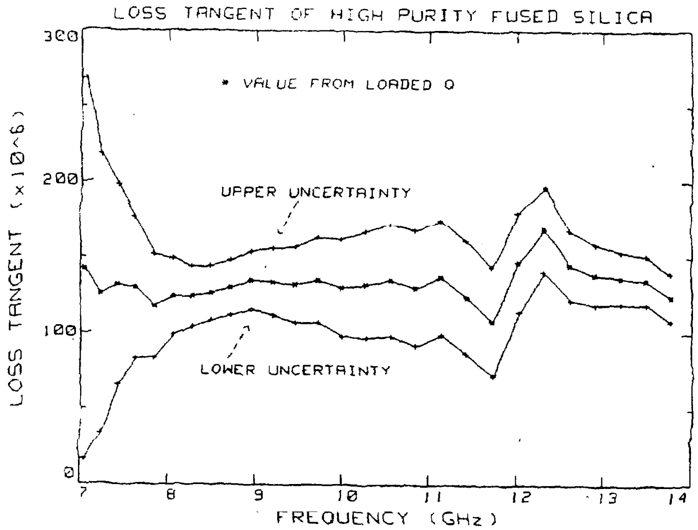


Figure 8.8: Loss tangent tanδ and estimated uncertainty of fused silica sample # 1.

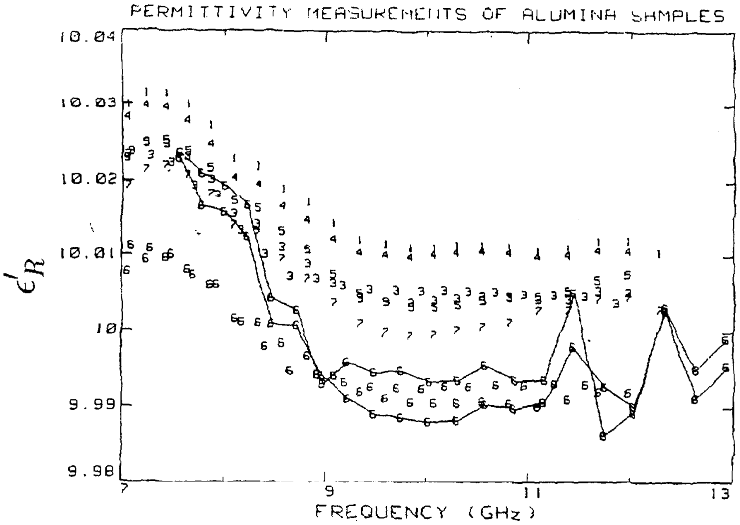


Figure 8.9: Measured permittivity ϵ'_R of several alumina samples from the same batch.

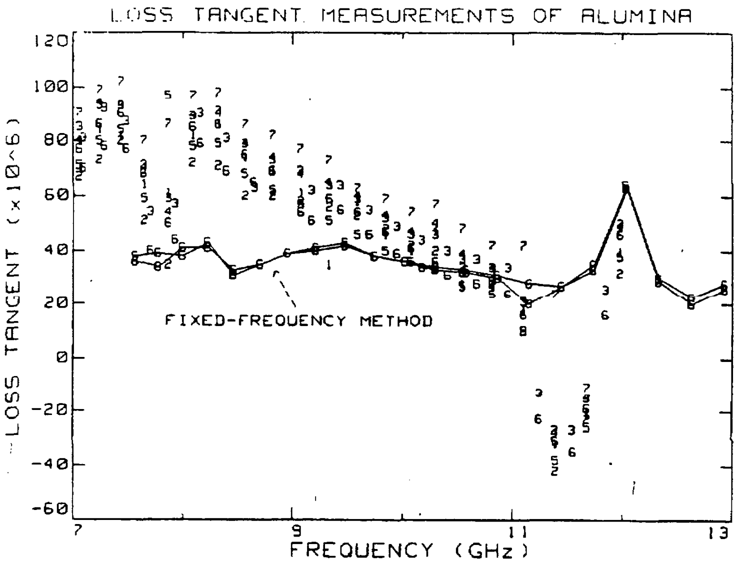


Figure 8.10: Measured loss tangent tanδ of several alumina samples from the same batch.

Chapter 9

Uncertainty Analysis

This chapter describes how uncertainty estimates are obtained from the computer program `CAVITYPROG`. Typical uncertainty estimates are given for a 10-mm thick sample with a relative permittivity value of $10 - j0.001$. `CAVITYPROG` generates uncertainty tables for ϵ'_R and tanδ due to individual parameter uncertainties. The uncertainties in cavity diameter and length, resonant frequency, micrometer reading, and resonance bandwidth were found in Chapter 4 and are summarized in Table 9.1.

The ϵ'_R and tanδ results for the samples presented in Chapter 8 show estimated total uncertainty. This chapter presents estimated ϵ'_R and tanδ uncertainties due to uncertainties in individual parameters. Because there can be any number of combinations of sample permittivity, loss, thickness, cavity length and resonant frequency, for the sake of brevity, we present results for a hypothetical sample with a relative permittivity of $10 - j0.001$. Uncertainty estimates are given for several frequencies in the X-band range. The uncertainty estimates are similar

Table 9.1: Estimated parameter uncertainties for the NIST resonator.

Cavity diameter	a	$\pm 2\,\mu$m
Cavity length	L_r	$\pm 9\,\mu$m
Micrometer reading	ΔL_r	$\pm 1\,\mu$m
Sample thickness	b	$\pm 3\,\mu$m
Resonance frequency	F_{sample}	±1000 Hz
Speed of light	c_{air}	±1000 m/s
Empty cavity Q	Q_0	±3%
With-sample Q	Q_s	±3%

for other permittivities, and must be evaluated on a case by case basis. For example, the measurement uncertainties for a thin sample with low dielectric constant can be much larger than one might expect (on the order of 20%), whereas a thicker sample or one with higher permittivity would have much smaller estimated measurement uncertainty.

9.1 Permittivity Uncertainty ($\Delta\epsilon'_R$)

To find uncertainties in ϵ'_R, CAVITYPROG calculates the change in ϵ'_R when one parameter is changed by its given uncertainty, while all other parameters are held fixed. This is repeated for each parameter to find the estimated uncertainty $\Delta\epsilon'_R$ due to individual parameters. CAVITYPROG then finds worst-case total uncertainty by summing the individual uncertainties. This variational method is equivalent to taking the total differential of ϵ'_R with respect to all parameters.

Figure 9.1 shows the estimated uncertainty $\Delta\epsilon'_R$ due to the individual measured parameter uncertainties given in Table 9.1.

Cavity diameter and length uncertainties are the greatest sources of uncertainty for ϵ'_R permittivity calculation. As shown previously in Fig. 4.3, the calculated cavity diameter changed less than 0.002 mm when the tuner endplate varied the cavity length by 24 mm. This result is corroborated at all frequencies in X-band as shown in Fig. 4.7 of Sec. 4.1.1 where the cavity dimensions were calculated from different subsets of the 7–14 GHz mode spectrum.

When we measure samples using the fixed-frequency method, the cavity length can be shortened by its maximum 24 mm range. From dimensional evaluation experiments described in Chapter 4, the uncertainty in diameter is less than ±0.002 mm and the uncertainty in length is no greater than ±0.009 mm. The overall uncertainty in cavity length is taken as the root-sum-square of the cavity length and the micrometer reading. These uncertainties are primarily due to endplate-travel accuracy (Sec. 4.1.2) and the agreement of calculated cavity dimensions from subsets of the 7–14 GHz resonance spectra (Sec. 4.1.1).

Sample thickness uncertainty is another significant error source that tends to have a maximum contribution at the frequencies at which the other error sources are minimized, as shown in Fig. 9.1. If the sample is thin < 2mm, this estimated error can become very significant especially if the sample surfaces are rough.

The speed of light in Boulder, Colorado is approximately $c_{air} = 2.99709 \times 10^8$ m/s at 10 GHz, which corresponds to a relative velocity factor of 0.99972. At standard temperature and pressure (23°C, 1.013×10^5 Pa (1013 mbar)) and 50% relative humidity the speed of light is approximately 2.99695×10^8 m/s with

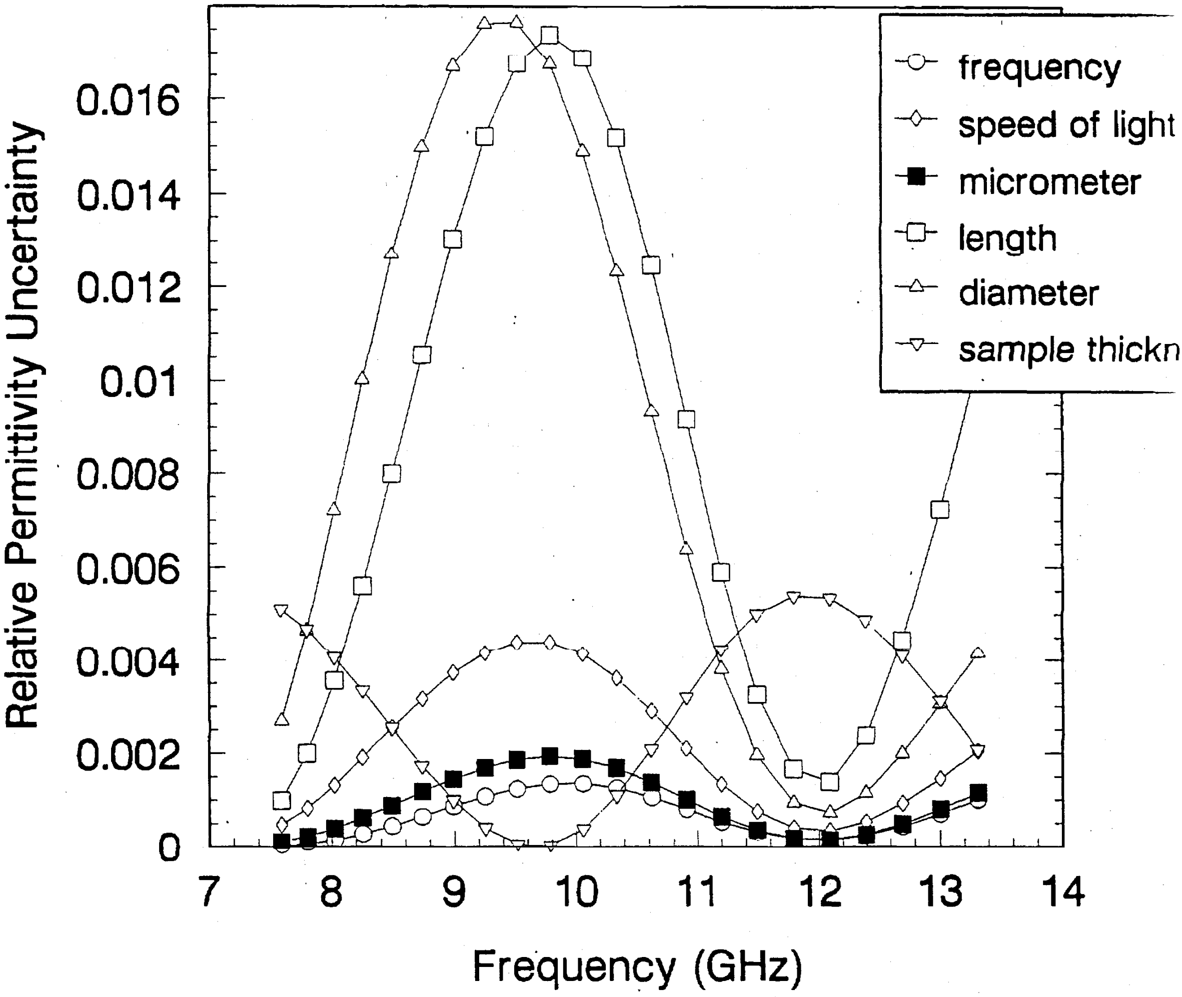


Figure 9.1: Permittivity uncertainty ($\Delta\epsilon_R'$) versus frequency due to the uncertainties in various measured parameters for a 10-mm thick sample with $\epsilon_R' = 10 - j0.001$.

relative velocity factor 0.99967. In the calculations for ϵ_R', erroneously setting the speed of light in the air-filled portion of the cavity to the speed of light in a vacuum gives a significant upward bias to ϵ_R' results. If we use the proper value for the speed of light is calculations, the residual uncertainty in c_{air} is approximately ±300 m/s. To accommodate for miscellaneous atmospheric disturbances we have specified the uncertainty in the speed of light to be approximately three times greater ($\approx \pm 1000$ m/s). Figure 9.1 shows that this uncertainty in the speed of light can indeed contribute an appreciable error to ϵ_R'.

Resonant frequency can be very accurately determined to within a few hundred hertz. However, if the resonant frequency is set and observed for one hour the resonant frequency drifts no more than 1 kHz due to changing cavity length caused by settling of the endplate drive micrometer or because of slight changes in cavity temperature. The reading of the sensing micrometer changes less than 0.001 mm to reflect this change in length. Because this problem occurs when the resonator is left alone for a period of time that is approximately the same as the time the computer takes to measure the X-band resonance spectrum, we use a 1 kHz uncertainty for the empty and sample-filled resonant frequencies. Fig. 9.1 shows the uncertainty in ϵ_R' due to a resonant frequency uncertainty of ±1 kHz. There are cases in which the Q of a resonance can be low (≈ 2000) because of a sample's extreme thickness or lossiness. In this case, resonant frequency and Q can be difficult to define because of asymmetries that may occur in the resonance response. When this is the case, one must use special precautions in defining the uncertainties in resonance frequency and Q because systematic errors having to do with the resonator's frequency dependent characteristics may be occuring.

9.2 Loss Tangent Uncertainty

Loss tangent uncertainty is calculated from the total differential of the loss tangent equation given by Cook [2]. We have verified that this method is identical to the variational method used in calculating $\Delta\epsilon_R'$ in which the uncertainties are added to individual parameters, one at a time, to find the change in tanδ. The resulting individual estimates for Δ tanδ are added together to get a worst-case uncertainty estimate. The total differential method can be used to estimate Δ tanδ because we have an explicit equation for tanδ. The equation is given by

$$\tan\delta = \frac{p(2b-s) + (1/\epsilon_R')[2(L_r - b) - q]}{p(2b-s)} \left(\frac{1}{Q_d} - \frac{1}{Q'}\right), \qquad (9.1)$$

where

$$p = \frac{\sin^2 \beta_0 (L_r - b)}{\sin^2 \beta_1 b}, \tag{9.2}$$

$$q = \frac{\sin 2\beta_0 (L_r - b)}{\beta_0}, \tag{9.3}$$

$$s = \frac{\sin 2\beta_1 b}{\beta_1}. \tag{9.4}$$

The total differential is found from

$$\begin{aligned}
\partial \tan\delta = {} & \partial L_r \left\{ \frac{\partial \tan\delta}{\partial L_r} + \frac{\partial \tan\delta}{\partial p}\frac{\partial p}{\partial L_r} + \frac{\partial \tan\delta}{\partial q}\frac{\partial q}{\partial L_r} \right\} \\
& + \partial b \left\{ \frac{\partial \tan\delta}{\partial b} + \frac{\partial \tan\delta}{\partial p}\frac{\partial p}{\partial b} + \frac{\partial \tan\delta}{\partial q}\frac{\partial q}{\partial b} + \frac{\partial \tan\delta}{\partial s}\frac{\partial s}{\partial b} \right\} \\
& + \partial \epsilon'_R \left\{ \frac{\partial \tan\delta}{\partial \epsilon'_R} + \frac{\partial \tan\delta}{\partial p}\frac{\partial p}{\partial \beta_\epsilon}\frac{\partial \beta_\epsilon}{\partial \epsilon'_R} + \frac{\partial \tan\delta}{\partial s}\frac{\partial s}{\partial \beta_\epsilon}\frac{\partial \beta_\epsilon}{\partial \epsilon'_R} \right\} \\
& + \partial Q_{\text{sample}} \left\{ \frac{\partial \tan\delta}{\partial Q_{\text{sample}}} \right\} \\
& + \partial Q_{\text{empty}} \left\{ \frac{\partial \tan\delta}{\partial Q_{\text{empty}}} \right\} \\
& + \partial c_{air} \left\{ \frac{\partial \tan\delta}{\partial p}\frac{\partial p}{\partial \beta_0}\frac{\partial \beta_0}{\partial c_{air}} + \frac{\partial \tan\delta}{\partial q}\frac{\partial q}{\partial \beta_0}\frac{\partial \beta_0}{\partial c_{air}} \right\} \\
& + \partial a \left\{ \frac{\partial \tan\delta}{\partial p}\frac{\partial p}{\partial \beta_0}\frac{\partial \beta_0}{\partial a} + \frac{\partial \tan\delta}{\partial p}\frac{\partial p}{\partial \beta_\epsilon}\frac{\partial \beta_\epsilon}{\partial a} + \frac{\partial \tan\delta}{\partial q}\frac{\partial q}{\partial \beta_0}\frac{\partial \beta_0}{\partial a} + \frac{\partial \tan\delta}{\partial s}\frac{\partial s}{\partial \beta_\epsilon}\frac{\partial \beta_\epsilon}{\partial a} \right\} \\
& + \partial f \left\{ \frac{\partial \tan\delta}{\partial p}\frac{\partial p}{\partial \beta_0}\frac{\partial \beta_0}{\partial f} + \frac{\partial \tan\delta}{\partial p}\frac{\partial p}{\partial \beta_\epsilon}\frac{\partial \beta_\epsilon}{\partial f} + \frac{\partial \tan\delta}{\partial q}\frac{\partial q}{\partial \beta_0}\frac{\partial \beta_0}{\partial f} + \frac{\partial \tan\delta}{\partial s}\frac{\partial s}{\partial \beta_\epsilon}\frac{\partial \beta_\epsilon}{\partial f} \right\}
\end{aligned} \tag{9.5}$$

Uncertainty is found by replacing the partials ∂L_r, ∂b, $\partial \epsilon'_R$, $\partial Q_{\text{sample}}$, $\partial Q_{\text{empty}}$, ∂c_{air}, ∂a and ∂f with their respective estimated uncertainties ΔL_r, Δb, $\Delta \epsilon'_R$, ΔQ_{sample}, ΔQ_{empty}, Δc_{air}, Δa and Δf. The partial derivatives are given by

$$\frac{\partial \tan\delta}{\partial L_r} = \frac{2\left(Q_{\text{empty}} - Q_{\text{sample}}\right)}{\epsilon'_R p Q_{\text{empty}} Q_{\text{sample}} (2b - s)}, \tag{9.6}$$

$$\frac{\partial \tan\delta}{\partial b} = \frac{-2\left(Q_{\text{empty}} - Q_{\text{sample}}\right)(2L_r - q - s)}{\epsilon'_R p Q_{\text{empty}} Q_{\text{sample}} (2b - s)^2}, \tag{9.7}$$

$$\frac{\partial \tan\delta}{\partial \epsilon'_R} = \frac{(Q_{\text{empty}} - Q_{\text{sample}})(2b - 2L_r + q)}{(\epsilon'_R)^2 p Q_{\text{empty}} Q_{\text{sample}} (2b - s)}, \tag{9.8}$$

$$\frac{\partial \tan\delta}{\partial Q_{\text{sample}}} = -\frac{2b(\epsilon'_R p - 1) - \epsilon'_R p s + 2L_r - q}{\epsilon'_R p Q^2_{\text{sample}}(2b - s)} , \tag{9.9}$$

$$\frac{\partial \tan\delta}{\partial Q_{\text{empty}}} = \frac{2b(\epsilon'_R p - 1) - \epsilon'_R p s + 2L_r - q}{\epsilon'_R p Q^2_{\text{empty}}(2b - s)} . \tag{9.10}$$

The partial derivatives of p, q and s are given by

$$\frac{\partial p}{\partial \beta_0} = \frac{2(b - L_r)\sin(\beta_0(b - L_r))\cos(\beta_0(b - L_r))}{\sin^2(b\beta_\epsilon)} , \tag{9.11}$$

$$\frac{\partial p}{\partial \beta_\epsilon} = -\frac{2b\cos(b\beta_\epsilon)\sin^2(\beta_0(b - L_r))}{\sin^3(b\beta_\epsilon)} , \tag{9.12}$$

$$\begin{aligned}\frac{\partial p}{\partial b} = {} & \frac{2\beta_0 \sin(\beta_0(b - L_r))\cos(\beta_0(b - L_r))}{\sin^2(b\beta_\epsilon)} \\ & -\frac{2\beta_\epsilon \cos(b\beta_\epsilon)\sin^2(\beta_0(b - L_r))}{\sin^3(b\beta_\epsilon)} ,\end{aligned} \tag{9.13}$$

$$\frac{\partial p}{\partial L_r} = -\frac{2\beta_0 \sin(\beta_0(b - L_r))\cos(\beta_0(b - L_r))}{\sin^2(b\beta_\epsilon)} , \tag{9.14}$$

$$\frac{\partial q}{\partial \beta_0} = \frac{\sin(2\beta_0(b - L_r))}{\beta_0^2} - \frac{2(b - L_r)\cos(2\beta_0(b - L_r))}{\beta_0} , \tag{9.15}$$

$$\frac{\partial q}{\partial b} = -2\cos(2\beta_0(b - L_r)) , \tag{9.16}$$

$$\frac{\partial q}{\partial L_r} = 2\cos(2\beta_0(b - L_r)) \text{ and} \tag{9.17}$$

$$\frac{\partial s}{\partial \beta_\epsilon} = \frac{2b\cos(2b\beta_\epsilon)}{\beta_\epsilon} - \frac{2b\beta_\epsilon}{\beta_\epsilon^2} , \tag{9.18}$$

$$\frac{\partial s}{\partial b} = 2\cos 2b\beta_\epsilon . \tag{9.19}$$

The the wavenumbers by the guide wavelength defined as

$$\beta = \sqrt{(2\pi f)^2 \mu\epsilon - \left(\frac{t'_{01}}{a}\right)^2} , \tag{9.20}$$

$$\beta_0 = \tfrac{2\pi}{\lambda_g} \sqrt{\left(\frac{2\pi f}{c_{air}}\right)^2 - \left(\frac{t'_{01}}{a}\right)^2} , \tag{9.21}$$

$$\beta_\epsilon = \tfrac{2\pi}{\lambda_{g\epsilon}} \sqrt{\left(\frac{2\pi f}{c_0}\right)^2 \epsilon'_R - \left(\frac{t'_{01}}{a}\right)^2} , \tag{9.22}$$

where $\epsilon'' \ll \epsilon'$ so that $\epsilon^* \approx \epsilon' - j\epsilon''$. The uncertainties in the wavenumbers are then found from the total differentials

$$
\begin{aligned}
\Delta\beta_0 &= \frac{\partial\beta_0}{\partial c_{air}}\Delta c_{air} + \frac{\partial\beta_0}{\partial a}\Delta a + \frac{\partial\beta_0}{\partial f}\Delta f \text{ and} \qquad (9.23)\\
\Delta\beta_\epsilon &= \frac{\partial\beta_\epsilon}{\partial\epsilon'_R}\Delta c_{air} + \frac{\partial\beta_0}{\partial a}\Delta a + \frac{\partial\beta_0}{\partial f}\Delta f\ , \qquad (9.24)
\end{aligned}
$$

where the partial derivatives are

$$
\begin{aligned}
\frac{\partial\beta_\epsilon}{\partial f} &= \frac{4\pi^2 f\epsilon'_R}{c_0^2}\frac{1}{\sqrt{\left(\frac{2\pi f}{c_0}\right)^2\epsilon'_R - \left(\frac{t'_{01}}{a}\right)^2}}\ , \qquad (9.25)\\
\frac{\partial\beta_\epsilon}{\partial a} &= \frac{(t'_{01})^2}{a^3}\frac{1}{\sqrt{\left(\frac{2\pi f}{c_0}\right)^2\epsilon'_R - \left(\frac{t'_{01}}{a}\right)^2}}\ , \qquad (9.26)\\
\frac{\partial\beta_\epsilon}{\partial\epsilon'_R} &= \frac{(2\pi f)^2}{2c_0^2}\frac{1}{\sqrt{\left(\frac{2\pi f}{c_0}\right)^2\epsilon'_R - \left(\frac{t'_{01}}{a}\right)^2}}\ , \qquad (9.27)
\end{aligned}
$$

and

$$
\begin{aligned}
\frac{\partial\beta_0}{\partial f} &= \frac{4\pi^2 f}{c_{air}^2}\frac{1}{\sqrt{\left(\frac{2\pi f}{c_{air}}\right)^2 - \left(\frac{t'_{01}}{a}\right)^2}}\ , \qquad (9.28)\\
\frac{\partial\beta_0}{\partial a} &= \frac{(t'_{01})^2}{a^3}\frac{1}{\sqrt{\left(\frac{2\pi f}{c_{air}}\right)^2 - \left(\frac{t'_{01}}{a}\right)^2}}\ , \qquad (9.29)\\
\frac{\partial\beta_0}{\partial c_{air}} &= -\frac{(2\pi f)^2}{c_{air}^3}\frac{1}{\sqrt{\left(\frac{2\pi f}{c_{air}}\right)^2 - \left(\frac{t'_{01}}{a}\right)^2}} \qquad (9.30)
\end{aligned}
$$

Loss tangent uncertainty due to the uncertainties in the measured parameters is shown in Fig. 9.2. We can observe a highly frequency-dependent behavior in loss tangent uncertainty due to cavity dimensions uncertainties. Typically we measure samples with much lower loss than our hypothetical 10-mm thick sample with permittivity $10 - j0.001$, and loss tangent uncertainty is usually dominated by uncertainties in the measurement of resonance Q.

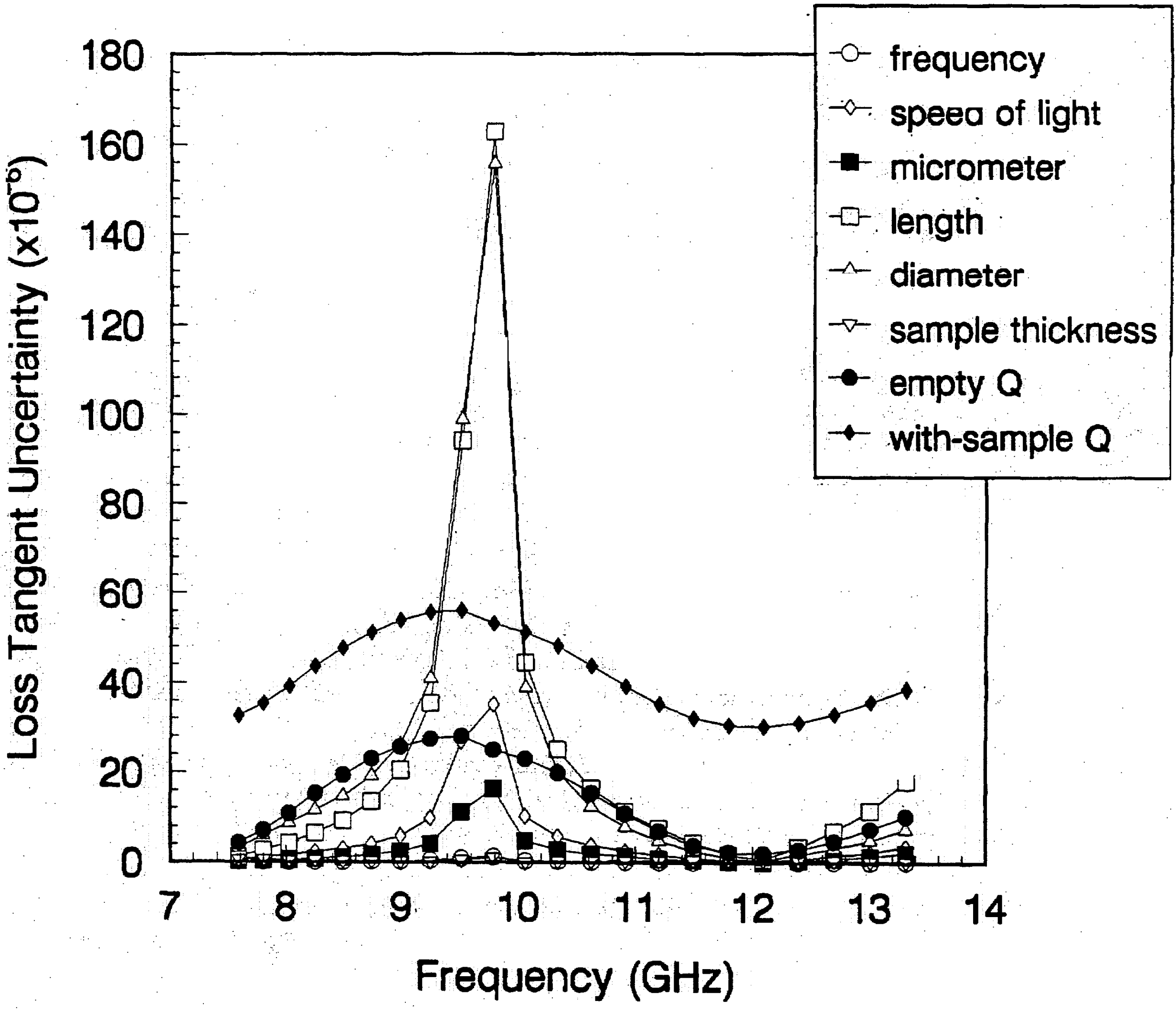


Figure 9.2: Loss tangent uncertainty $\Delta(\tan\delta)$ versus frequency due to uncertainties in various measured parameters for a 10-mm thick sample with $\epsilon_R' = 10 - j0.001$.

Chapter 10

Future Work

The change in length of the cavity is monitored by the sensing micrometer. As described in Sec. 4.1.2, the sensing micrometer's tip gradually wears a small pit into the extended finger of the tuner-endplate yoke. The yoke is made of aluminum and is therefore softer than the sensing micrometer's tip. To improve the repeatability of cavity length measurement, we will press fit a hardened steel plug into the yoke for the sensing micrometer to press against.

The uncertainty estimates given in Chapter 9 assume that systematic errors have been accounted for. To determine whether our accuracy estimates are truly correct, we must compare our results with the results of other instruments of similar accuracy. The glass results given in Sec. 8.1 compare favorably with results from an international intercomparison completed in 1974. NIST plans to engage in future international comparisons not only to qualify the NIST resonator for calibration service; but also to qualify our stock of cross-linked polystyrene, fused silica, and alumina as a Standard Reference Material.

Acknowledgements

The authors express their sincere appreciation to Hans J. Liebe of the U.S. Department of Commerce Institute for Telecommunication Services for his valuable contributions in the design and evaluation of the NIST cavity resonator. Thanks also go to Ulrich Stumper of the German Physikalisch Technische Bendesanstalt for his consultations on the initial design of the NIST resonator. We also express our appreciation to William A. Kissick. His leadership of the NIST Electromagnetic Properties of Materials Program and of the Broadband Microwave Metrology Group has contributed a great deal to the progress and successes of many individuals. Thanks also go to Howard E. Bussey, whose many contributions to the field of electromagnetic properties of materials measurements continue to be standards by which new techniques are judged.

The authors thank Lydell L. Frasch of Boeing Defense and Space Group, Charles Wilker and Ron Riegert of E.I. du Pont de Nemours & Company, and Mark D.A. Rosen and Mark Lizza of Grumman Corporate Research Center for their careful review of this Technical Note. Critical technical review of NIST publications is an important component of NIST manuscript review, and their contribution is greatly appreciated.

References Cited

[1] F. Horner, T. Taylor, R. Dunsmuir, J. Lamb, and W. Jackson, "Resonance methods of dielectric measurement at centimetre wavelengths," *J. IEE*, vol. 93, Pt. III, pp. 53–68, 1946.

[2] R.J. Cook, "Microwave cavity methods," in *High frequency dielectric measurement*, (J. Chamberlain and G.W. Chantry, ed.), pp. 12–27, Guildford, U.K., IPC Science and Technology Press, 1973, Mar. 1972.

[3] E. Ni and U. Stumper, "Permittivity measurements using a frequency-tuned microwave TE_{01} cavity resonator," *IEE Proceedings*, vol. 132, Pt. H, pp. 27–32, Feb. 1985.

[4] R. G. Geyer, "Dielectric Characterization and Reference Materials," NIST-TN 1338, Natl. Inst. Stand. Technol., 1990.

[5] R.F. Harrington, *Time-Harmonic Electromagnetic Fields*, New York: McGraw-Hill Publishing Co., 1961.

[6] S. Morgan and J. Young, "Helix waveguide," *Bell System Tech. J.*, vol. 35, pp. 1347–1384, 1956.

[7] H.G. Unger, "Helix waveguide theory and application," *Bell System Tech. J.*, vol. 37, pp. 1599–1647, 1958.

[8] D. Young, "Measured TE_{01} attenuation in helix waveguide with controlled straightness deviations," *Bell System Tech. J.*, vol. 44, pp. 273–282, 1965.

[9] R.J. Cook and R.G. Jones, "Precise Dielectric Measurement Techniques for the Frequency Range 10 GHz to 150 GHz," pp. 528–531, Proc. 8th European Microwave Conf., 1978.

[10] R. Waldron, *Theory of Guided Electromagnetic Waves*, New York: Van Nostrand Reinhold Co., 1970.

[11] R. Waldron, "A helical coordinate system and its applications in electromagnetic theory," *Quart. J. Mech. Appl. Math*, vol. 11, pp. 438–461, 1958.

[12] R. Waldron and D. Bowe, "Normal modes of helix waveguide.," *Marconi Rev.*, vol. 28, pp. 29–64, 1965.

[13] M. Shimba, "Measurement of spurious mode losses in helix waveguide," *Rev. Electron. Commun. Lab., Tokyo*, vol. 11, pp. 111–120, 1962.

[14] Y. Kazantsev, I. Kaznacheev, and V. Meriakri, "Investigation of helix waveguide," *Radio Eng. Electron. (translation of Radiotekh. Elektron.)*, vol. 4, pp. 88–94, 1959.

[15] K. Mikoshiba, "TE_{01} loss characteristics in multimode waveguide with random imperfections," *J. Inst. Electron. Comm. Eng. of Japan*, vol. 4, pp. 2–3, 1964.

[16] G. Piefke, "The influence of helix wire diameter on the modes in a helix waveguide," *IEE paper No. 2858*, vol. E, pp. 110–118, Sep. 1959.

[17] R. Waldron, D. Bowe, P. Wackrill, and B. Wescott, "Helix waveguide theory," in *Proc. IEE Conf. Transmission Aspects of Communication Networks*, pp. 210–215, 1964.

[18] K. Noda, K. Yamaguchi, and N. Suzuki, "Circular electric wave transmission through helix waveguide," *Rev. Electron. Commun. Lab., Tokyo*, vol. 10, pp. 49–70, 1962

[19] D. Kajfez, "Correction for measured resonant frequency of unloaded cavity," *Electron. Lett.*, vol. 20, no. 2, pp. 81–82, 1984.

[20] E.L. Ginzton, *Microwave Measurements*, *International Series in Pure and Applied Physics*, New York: McGraw-Hill, 1957.

[21] M. Sucher and J. Fox, *Handbook of Microwave Measurements*, Vol. II Brooklyn, N.Y.: Polytechnic Press, 1963.

[22] D. Kajfez and E. Hwan, "Q-factor measurement with a network analyzer," *IEEE Trans. Microwave Theory Tech.*, vol. 32, no. 7, pp. 666–670, 1984.

[23] A.J. Estin, "Resonator characterization and measurement," Cyberlink Corporation, Boulder, CO, Final Report to Sponsor, July 1990.

[24] A.J. Estin and M.D. Janezic, "Improvements in dielectric measurements with a resonant cavity," in *IEEE IMTC-S Conf. Proc.*, Atlanta, GA, May 1991.

[25] D. Kajfez, "Incremental frequency rule for computing the Q-factor of a shielded TE_{0mp} dielectric resonator," *IEEE Trans. Microwave Theory Tech.*, vol. 32, pp. 941–943, Aug. 1984.

[26] H.A. Wheeler, "Formulas for the skin effect," *Proc. IRE*, vol. 30, pp. 412–424, Sep. 1942.

[27] H.E. Bussey, J.E. Gray, E.C. Bamberger, E. Rushton, G. Russell, B.W. Petley, and D. Morris, "International comparison of dielectric measurements," *IEEE Trans. Instrum. Meas. Technol.*, vol. IM-13, pp. 305–311, Dec. 1964.

[28] J.C. Gallop and W.J. Radcliffe, "Dimensional measurement by microwave resonances," *J. Phys. E: Sci. Instrum.*, vol. 14, pp. 461–463, 1981.

[29] J.C. Slater, *Microwave Electronics*, New York: D. Van Nostrand Company, Inc., 1950, p. 75.

[30] H. Bussey, "Standards and measurements of microwave surface impedance, skin depth, conductivity and Q," *IRE Trans. Instrum.*, vol. I-9, pp. 171–175, Sep. 1960.

[31] H. Bethe and J. Schwinger, "Perturbation theory for cavities," Tech. Rep. D1 117, Natl. Defense Res. Committee, Washington, DC, Mar. 1943.

[32] P. Boggs, R. Byrd, J. Donaldson, and R. Schnabel, "Algorithm 676 — ODRPACK: software for weighted orthogonal distance regression," *ACM Trans. Mathematical Software*, vol. 15, no. 4, pp. 348–364, 1989.

[33] C. Montgomery, R. Dicke, and E. Purcell, *Principles of Microwave Circuits.* Vol. 8 of *MIT Radiation Laboratory Series*, McGraw-Hill Book Co., Inc., 1948.

[34] E. Vanzura and J. Rogers, "Resonant circuit model evaluation using reflected S-parameters," in *IEEE IMTC-S Conf. Proc.*, Atlanta, GA, May 1991.

[35] H. Nussenzvieg, *Causality and Dispersion Relations*, Vol. 95 of *Mathematics in Science and Engineering*, New York: Academic Press, Inc., 1972.

[36] Von G. Schulten, "Resonatoren fur millimeterwellen und ihre verwendung zur beobachtung von gasresonanzen," *Frequenz*, vol. 20, no. 1, pp. 10–22, 1966.

[37] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, *Numerical Recipes: The Art of Scientific Computing*, Cambridge: Cambridge University Press, 1986.

[38] E.J. Vanzura, "Creating CSUBs written in FORTRAN that run in BASIC," in *INTEREX Proceedings of the 1988 Conference of HP Technical Computer Users*, INTEREX International Association of Hewlett-Packard Computer Users, Paper 20, 18 pages, Aug. 1988.

[39] E.J. Vanzura, "Creating CSUBs in BASIC," *HP Design and Automation*, Oct. and Nov. 1988, Austin: Wilson Publications, Inc.

[40] H. Liebe, "An updated model for millimeter wave propagation in moist air," *Radio Science*, vol. 20, pp. 1069–1089, Sep.–Oct. 1985.

[41] N.R. Draper and H. Smith, *Applied Regression Analysis*. J. Wiley and Sons, Inc., second ed., 1981.

[42] J.D. Jackson, *Classical Electrodynamics*. New York: J. Wiley and Sons, Inc., second ed., 1975.

[43] H. E. Bussey, D. Morris, and E. Zal'tsman, "International comparison of complex permittivity measurement at 9 GHz," *IEEE Trans. Instrum. Meas. Technol.*, vol. IM-23, pp. 235–239, Sep. 1974.

[44] P.C.R. Grivet *Acad. Sci. (Paris)*, vol. 218, p. 71, 1944.

[45] H. Bussey and J. Gray, "Measurement and standardization of dielectric samples," *IRE Trans. Instrum.*, vol. I-11, pp. 162–165, Dec. 1962.

Appendix A

Partial Filling Factor for Loss Determination

Previously we showed that the unloaded quality factor of the cavity with sample inserted is

$$\frac{1}{Q_{0,\text{ sample}}} = \frac{1}{Q_C} + \frac{1}{Q_S}, \tag{A.1}$$

where $\tan\delta = 1/Q_S$. The empty unloaded cavity quality factor is

$$\frac{1}{Q_{0,\text{ empty}}} = \frac{1}{Q_C}. \tag{A.2}$$

In practice, a loaded cavity quality factor is always measured, which includes coupling port losses Q_E

$$\frac{1}{Q_L} = \frac{1}{Q_0} + \frac{1}{Q_E}. \tag{A.3}$$

where Q_0 is either with or without sample insertion. The loaded quality factor is related to the unloaded quality factor by (2.80).

The analysis in Secs. 2.2.5 and 2.2.6 derived the theoretical Q_C and the measurement approach for evaluating total power loss in the cavity walls, as well as a technique for separating sidewall from end plate losses. However, the loss in the sidewall of the cavity is different when the sample is inserted as opposed to when the cavity is empty (air filled). Therefore, it is necessary to determine the energy stored and power dissipated both (1) in the sample of arbitrary thickness having relative complex permittivity $\epsilon_R' - j\epsilon_R''$ and (2) in the air-filled portion of the cavity when the sample is inserted. This allows us to account for the fact that the cavity is only partially filled with dielectric.

Reference [2] shows that the theoretical quality factor Q_0 of a cylindrical cavity of radius a that is partially filled with a sample of thickness b having a relative complex permittivity $\epsilon_R^* = \epsilon'_R - j\epsilon''_R$, is given by

$$Q_0 = \frac{\{F(2b-G) + [2(L-b) - U]/\epsilon'_R\}\, a\omega^2 \mu'_R \epsilon'_R}{\{\delta_s c^2 \left(k_c^2 \left[(2b-G)F + 2(L-b) - U\right] + 2a \left[F\beta_1^2 + \beta_0^2\right)\right. + a\omega^2 \mu'_R \epsilon'_R F(2b-G) \tan\delta\}}\,, \tag{A.4}$$

where c is the speed of light in air, ω is angular frequency for any TE_{01p} resonant mode, δ_s is the effective penetration depth given by (2.123) of the cavity's wall and end plates, and

$$F = \frac{\sin^2[\beta_0(L-b)]}{\sin^2[\beta_1 b]}\,, \tag{A.5}$$

$$G = \frac{\sin[2\beta_1 b]}{\beta_1}\,, \tag{A.6}$$

$$U = \frac{\sin[2\beta_0(L-b)]}{\beta_0}\,. \tag{A.7}$$

Equation (A.4) is valid for a dielectrically lossy sample and takes into account the effect of finite wall and end plate loss intrinsic to the cavity. The measured unloaded quality factor $Q_{0,m}$ with the dielectric sample is generally somewhat lower than the theoretical Q_0 in (A.4). From (2.123) and (A.4) Cook [2] has further shown that the loss tangent may be precisely calculated from

$$\tan\delta = \left\{\frac{F(2b-G) + [2(L-b) - U]/\epsilon'_R}{F(2b-G)}\right\} \left[\frac{1}{Q_{0,m}} - \frac{1}{\bar{Q}}\right]\,. \tag{A.8}$$

where

$$\bar{Q} = \frac{\{F(2b-G) + [2(L-b) - U]/\epsilon'_R\}\, a\omega^2 \mu'_R \epsilon'_R}{\delta_{s,m} c^2 \left[k_c^2 \left[F(2b-G) + 2(L-b) - U\right] + 2a\left(F\beta_1^2 + \beta_0^2\right)\right]}\,. \tag{A.9}$$

Equation (A.8) is similar to previous analysis except it now contains a filling factor because the specimen only partly fills the cavity.

Appendix B

Radial Air Gap Correction

One possible source of error in dielectric property measurements at microwave frequencies is that due to the air gap between the sample under test and the wall of the waveguide or resonator fixture. The solid dielectric under test is usually machined into the shape of a parallel-sided disk for a right-circular cylindrical resonator, and clearance must be left between the sides of the disk and the wall of the resonator so that the disk may move freely in the resonator. Exactly how much clearance is permissible before the error in dielectric characterization of the sample becomes significant is a problem that must be examined.

Consider two situations for our cavity resonator in which the walls are modelled as perfect electrical conductors. The first case (Fig. B.1) illustrates a dielectric sample in a cylindrical resonator with no air gap. The second case (Fig. B.2) shows the actual situation in which an air gap exists. The presence of this gap results in an increased length of $L_1 + \Delta L$ at resonance, where $L_1 = L - b$ for the ideal case. This, in turn, leads to a measured permittivity of the sample under test that is too small.

One approach to correct for the presence of an air gap is to consider the change in resonant frequency due to the gap with perturbation theory, where the ideal situation shown in Fig. B.1 is perturbed. Since we know the measured length $L_1 + \Delta L$ at measured resonant frequency f_m and the radii of both the sample under test and dielectric resonator, we can estimate the resonant frequency for the resonator with no air gap but of length $L_1 + \Delta L$ for use in the transcendental equation for permittivity calculation. Here a method described by Grivet [44] and Bussey [45] is outlined.

We noted that at resonance the total energy interior to the cavity is constant, as a function of time and that the average values of the energy stored in the

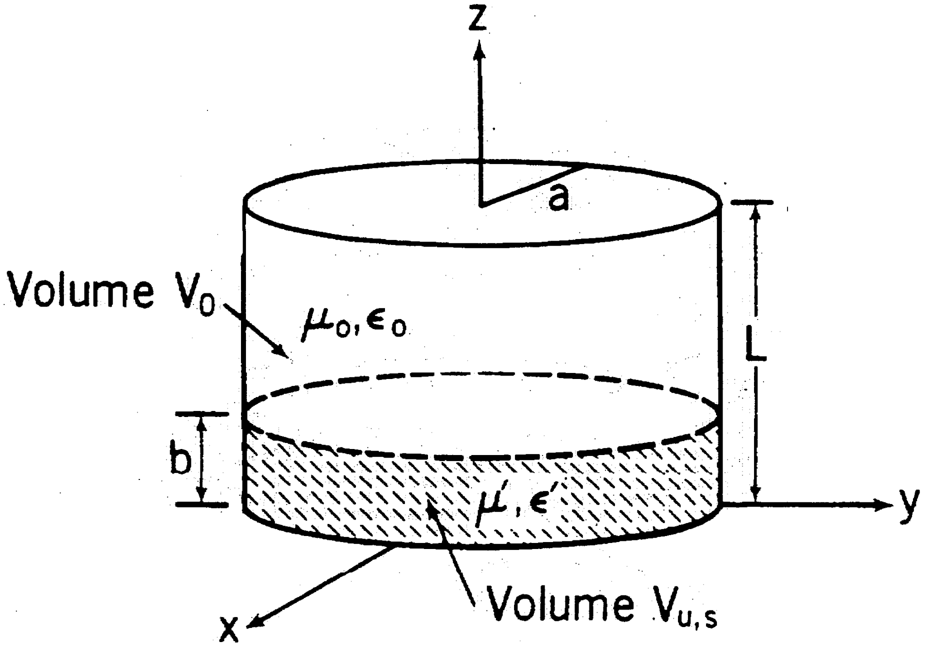


Figure B.1: Unperturbed cylindrical cavity resonator with no air gap present between sample under test and resonator side wall.

electric or magnetic field are equal. Hence, for TE_{01p} mode structure,

$$\iiint_{\text{cavity}} \epsilon' |E_o E_o^*| dV = \iiint_{\text{cavity}} \mu |\vec{H} \vec{H}^*| dV , \tag{B.1}$$

where the * denotes complex conjugate and the integrals are taken over the entire volume of the cavity. For a magnetically impermeable sample under test, μ may be taken outside of the above integral and since, by Faraday's law,

$$\nabla \times \vec{E} = -j2\pi f \mu \vec{H} . \tag{B.2}$$

(B.1) can be written

$$\iiint_{\text{cavity}} \epsilon' \vec{E}|^2 dV = -\frac{1}{4\pi^2 f^2 \mu} \iiint_{\text{cavity}} \left[\nabla \times \vec{E}\right]^2 dV . \tag{B.3}$$

If we now make the assumption that the electric field in the perturbed cavity is approximately that of the unperturbed, we may write

$$\vec{E} \approx \vec{E}_u . \tag{B.4}$$

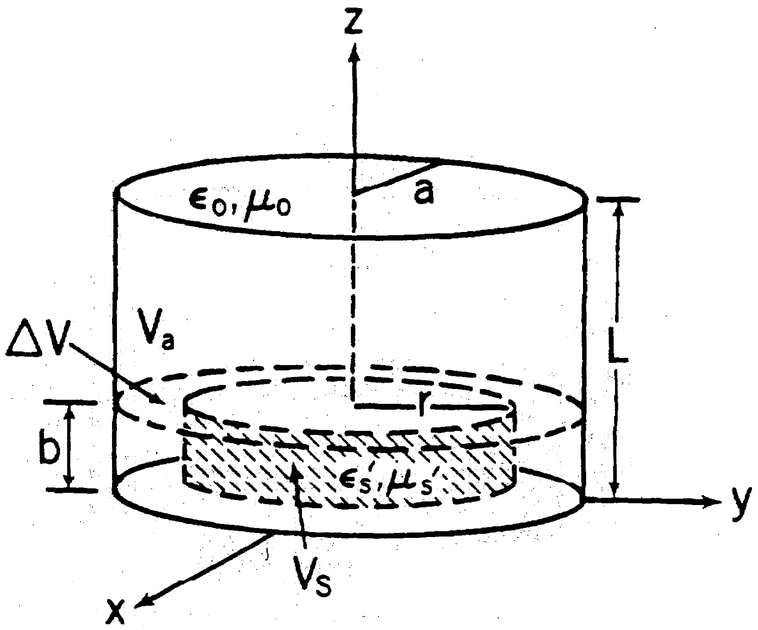


Figure B.2: Perturbed cylindrical cavity with uniform air gap between sample and resonator wall.

since we know that $\vec{E}_u$ satisfies the correct boundary conditions at the conducting walls of the cavity and is continuous at the boundary of the sample under test. This approximation is valid only if the air gap is not too large relative to λ_g and if the air gap is uniform, so azimuthal symmetry is maintained. From (B.4),

$$\nabla \times \vec{E}_u = -j2\pi f_u \mu \vec{H}_u \ , \tag{B.5}$$

where f_u denotes the unperturbed resonant frequency. Hence,

$$\begin{aligned} & f_m^2 \left\{ \iiint_{V_0} \epsilon_0 |\vec{E}_{u,\ \mathrm{air}}|^2 dV + \iiint_{V_{u,s}-\Delta V} \epsilon_s' |\vec{E}_{u,s}|^2 dV \right. \\ & \left. + \iiint_{\Delta V} \epsilon_0 |\vec{E}_{u.s}|^2 dV \right\} = \\ & \qquad f_u^2 \left\{ \iiint_{V_0} \mu_0 |\vec{H}_{u,\ \mathrm{air}}|^2 dV + \iiint_{V_{u,s}-\Delta V} \mu_0 |\vec{H}_{u,s}|^2 dV \right. \\ & \qquad \left. + \iiint_{\Delta V} \mu_0 |\vec{H}_{u,s}|^2 dV \right\} \ , \end{aligned} \tag{B.6}$$

where f_m represents the measured resonant frequency, the subscripts u and s denote unperturbed and sample, and where the integration over ΔV is an integration over the volume of the air gap which is

$$\Delta V = \pi a^2 b - \pi r^2 b = \pi b \left(a^2 - r^2 \right) \ , \tag{B.7}$$

where a and r are the respective radii of the resonator and the sample under test. Equation (B.6) may be simplified as follows:

$$\begin{aligned} & f_m^2 \left\{ \iiint_{V_0} \epsilon_0 |\vec{E}_{u,\text{ air}}|^2 dV + \iiint_{V_{u,s}} \epsilon_s' |\vec{E}_{u,s}|^2 dV \right. \\ & \left. + \iiint_{\Delta V} (\epsilon_0 - \epsilon_s') |\vec{E}_{u,s}|^2 dV \right\} \\ = \; & f_u^2 \left\{ \iiint_{V_0} \mu |\vec{H}_{u,\text{ air}}|^2 dV + \iiint_{V_{u,s}} \mu |\vec{H}_{u,s}|^2 dV \right\} . \end{aligned} \tag{B.8}$$

We now note that

$$\begin{aligned} & f_u^2 \left\{ \iiint_{V_0} \mu |\vec{H}_{u,\text{ air}}|^2 dV + \iiint_{V_{u,s}} \mu |\vec{H}_{u,s}|^2 dV \right\} \\ \equiv \; & f_m^2 \left\{ \iiint_{V_0} \epsilon_0 |\vec{E}_{u,\text{ air}}|^2 dV + \iiint_{V_{u,s}} \epsilon_s' |\vec{E}_{u,s}|^2 dV \right\} , \end{aligned} \tag{B.9}$$

from (B.1), so

$$\begin{aligned} & f_m^2 \left\{ \iiint_{V_0} \epsilon_0 |\vec{E}_{u,\text{ air}}|^2 dV + \iiint_{V_{u,s}} \epsilon_s' |\vec{E}_{u,s}|^2 dV \right. \\ & \left. - \iiint_{\Delta V} (\epsilon_s' - \epsilon_0) |\vec{E}_{u,s}|^2 dV \right\} \\ = \; & f_u^2 \left\{ \iiint_{V_0} \epsilon_0 |\vec{E}_{u,\text{ air}}|^2 dV + \iiint_{V_{u,s}} \epsilon_s' |\vec{E}_{u,s}|^2 dV \right\} . \end{aligned} \tag{B.10}$$

We now have a correction formula for the unperturbed resonant frequency f_u in terms of the measured resonant frequency f_m in the presence of an air gap of volume ΔV; that is,

$$f_u = f_m \left\{ 1 - \frac{\iiint_{\Delta V} (\epsilon_s' - \epsilon_0) |\vec{E}_{u,s}|^2 dV}{\iiint_{V_0} \epsilon_0 |\vec{E}_{u,\text{ air}}|^2 dV + \iiint_{V_{u,s}} \epsilon_s' |\vec{E}_{u,s}|^2 dV} \right\}^{1/2} \tag{B.11}$$

Of course, when there is no air gap $\Delta V = 0$ and $f_u = f_m$. For the mode-filtered TE_{01p} cavity of concern here, we have only an azimuthal component of the electric field that is of concern in the above integrations.

If an air gap is present between the sample under test and wall of the waveguide of the cavity resonator, the cavity will be at resonance at an increased length (compared to the situation where no gap exists). This leads to a measured dielectric constant that is too small. The procedure from the analysis above, which is valid if the air gap is not too large relative to the guide wavelength and if the air gap is uniform so that azimuthal symmetry is maintained, is as follows:

1. Measure the resonant frequency f_m and measured length of the cavity at resonance.
2. Compute the unperturbed resonant frequency f_u for a resonator with no air gap but with the length measured in step 1 above.
3. Use the unperturbed frequency derived in step 2 for ϵ'_R determination in (2.46).

An alternate form of (B.11) for computing the correction to the measured relative dielectric constant in the presence of a radial air gap for TE_{01p} mode is

$$\Delta\epsilon'_R = (\epsilon'_R - 1)\frac{\int_r^a \int_0^{2\pi} [J_1(k_c r_1)]^2 \sin^2(\beta_1 z)\, r_1 dr_1 d\phi}{\int_0^a \int_0^{2\pi} [J_1(k_c r_1)]^2 \sin^2(\beta_1 z)\, r_1 dr_1 d\phi}\,, \tag{B.12}$$

which reduces to

$$\Delta\epsilon'_R = \frac{(\epsilon'_R - 1)}{2}\left\{1 - \left(\frac{r}{a}\right)^2 \frac{\Psi}{J_0^2(k_c a)}\right\}\,, \tag{B.13}$$

with

$$\Psi = J_0^2(k_c r) - \frac{2}{k_c r} J_0(k_c r) J_1(k_c r) + J_1^2(k_c r)\,. \tag{B.14}$$

Appendix C

60-mm Cavity Drawings

To be consistent with shop tooling, dimensions for resonator parts were specified in inches. Apologies are offered for our deviation from Standard International units. In the following drawings, dimensions are in inches unless otherwise specified.

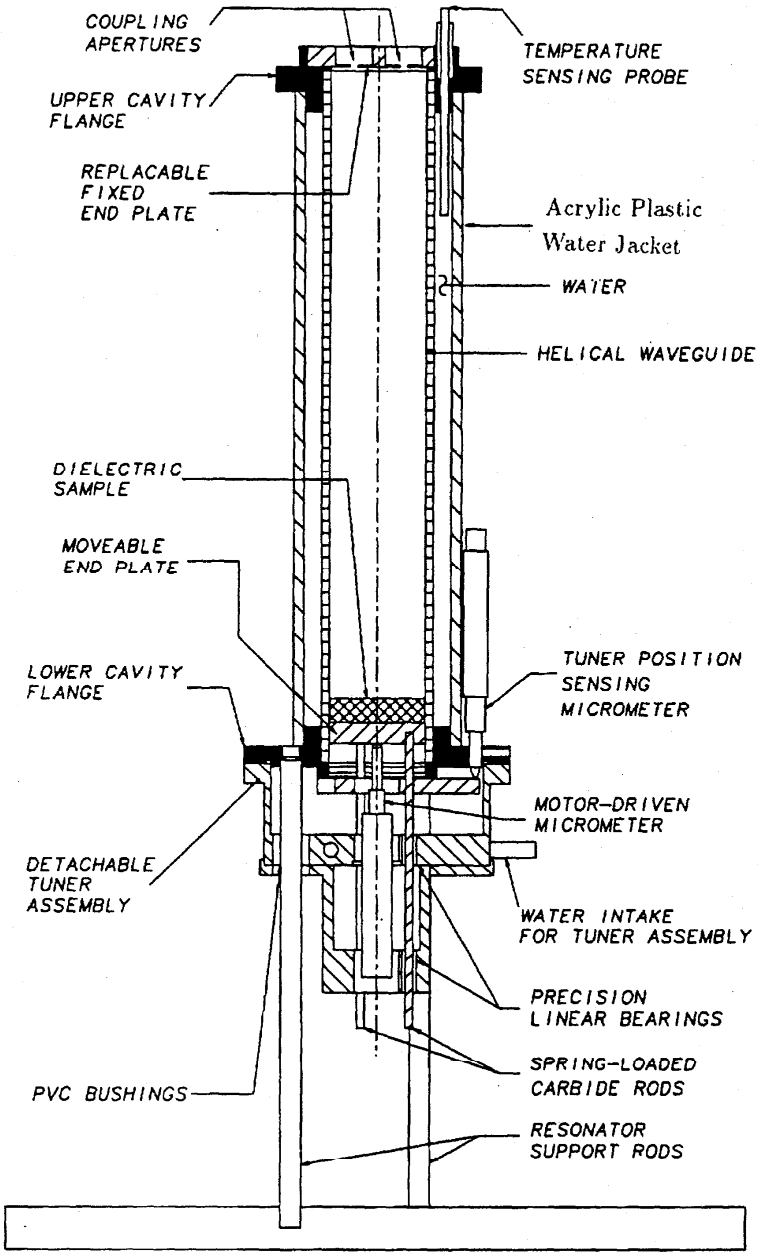


Figure C.1: Cutaway assembly drawing of NIST cavity resonator.

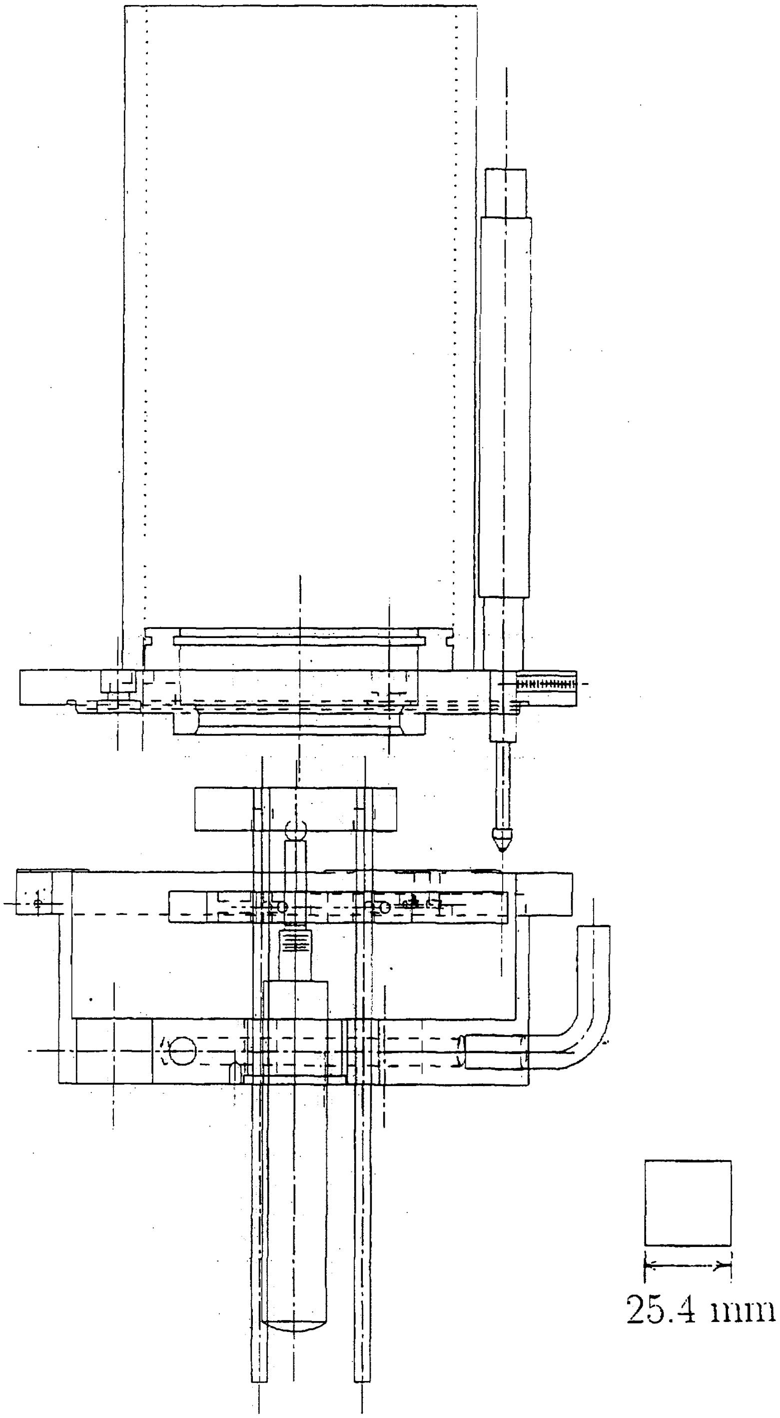


Figure C.2: Detail of tuner assembly and flange.

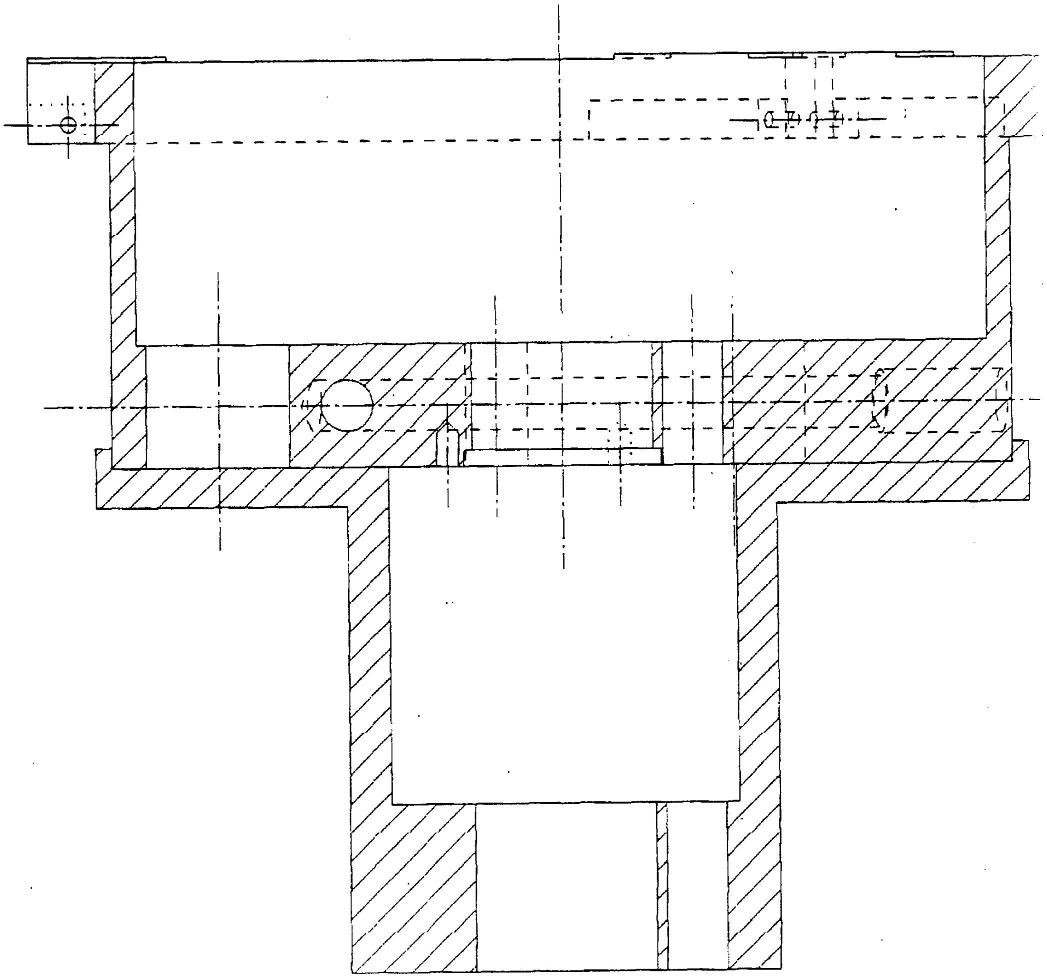


Figure C.3: Cutaway side view of tuner-base and extension.

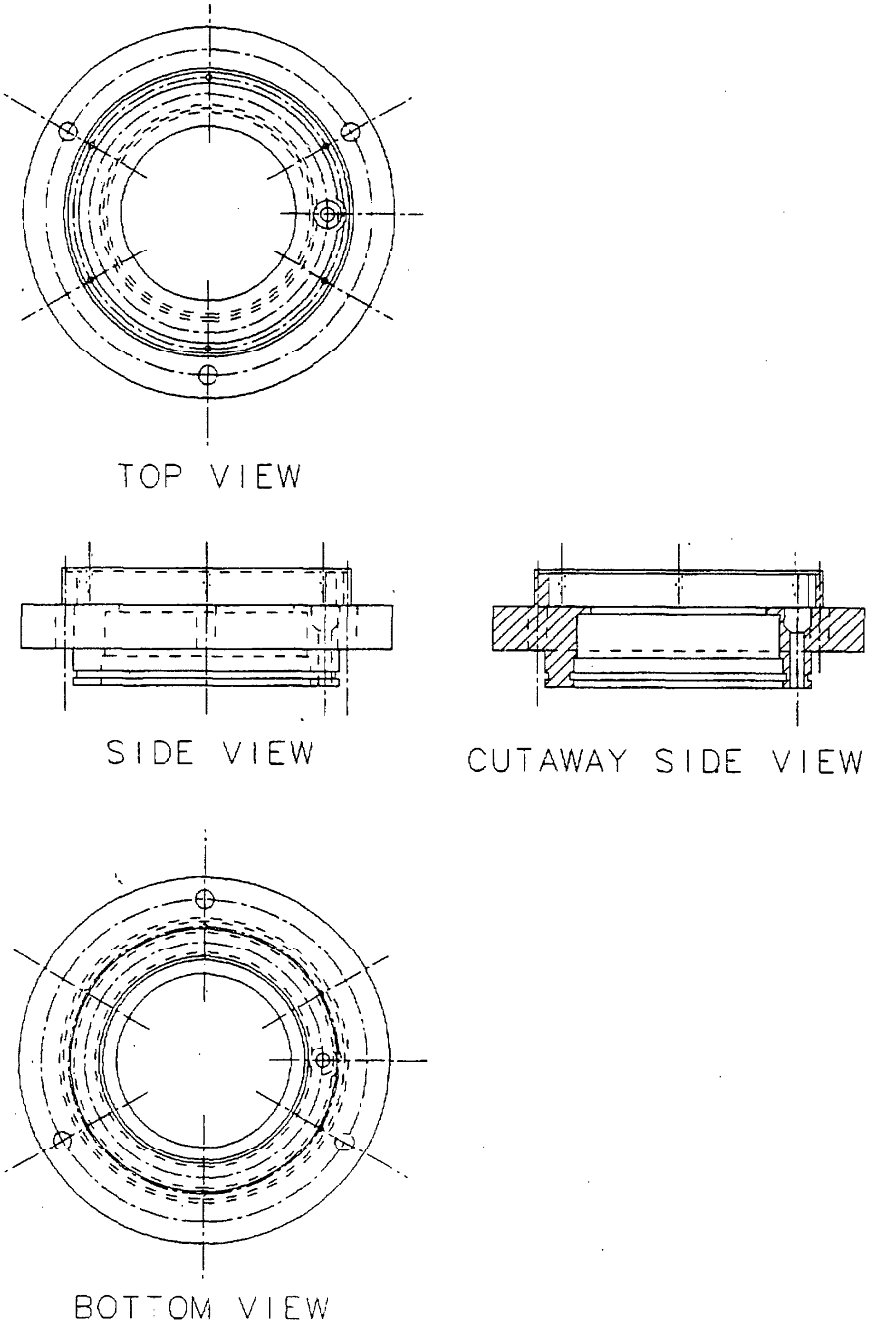


Figure C.4: Orthogonal views of coupling flange.

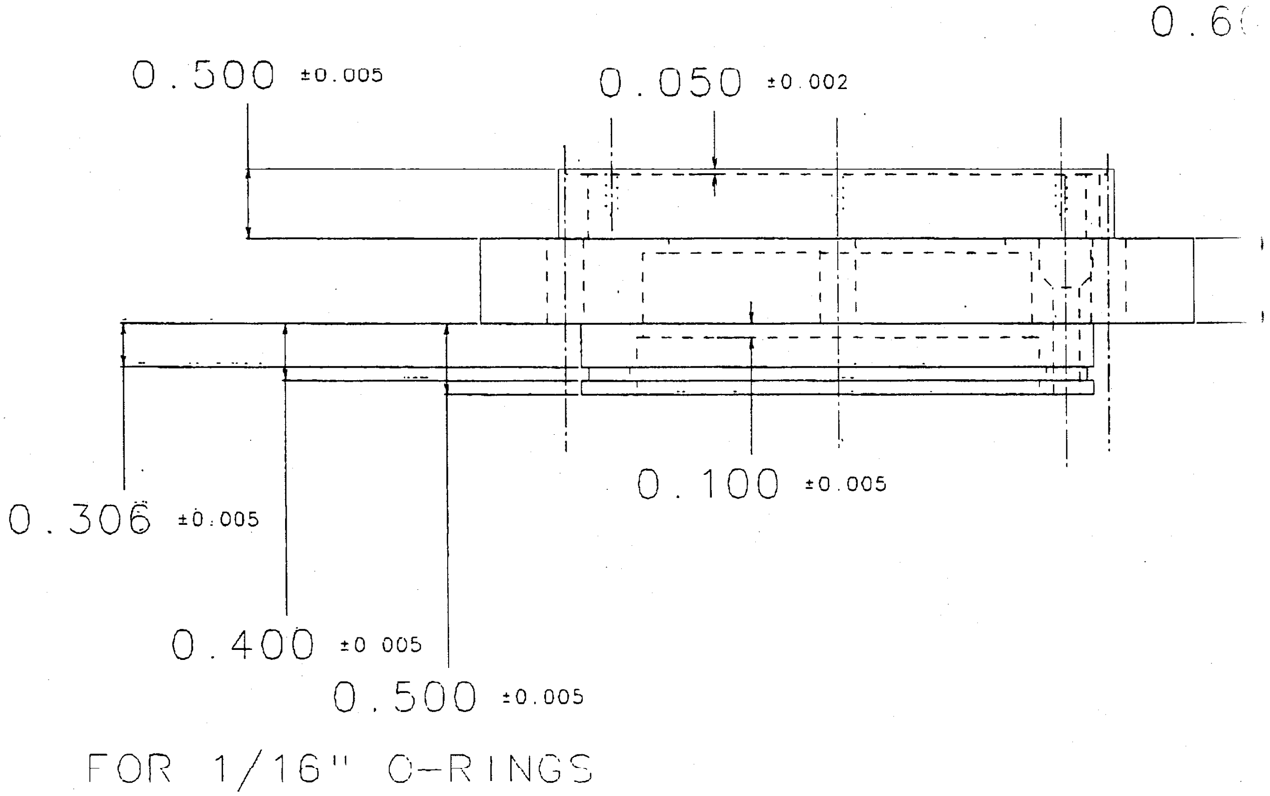


Figure C.5: Side view of coupling flange.

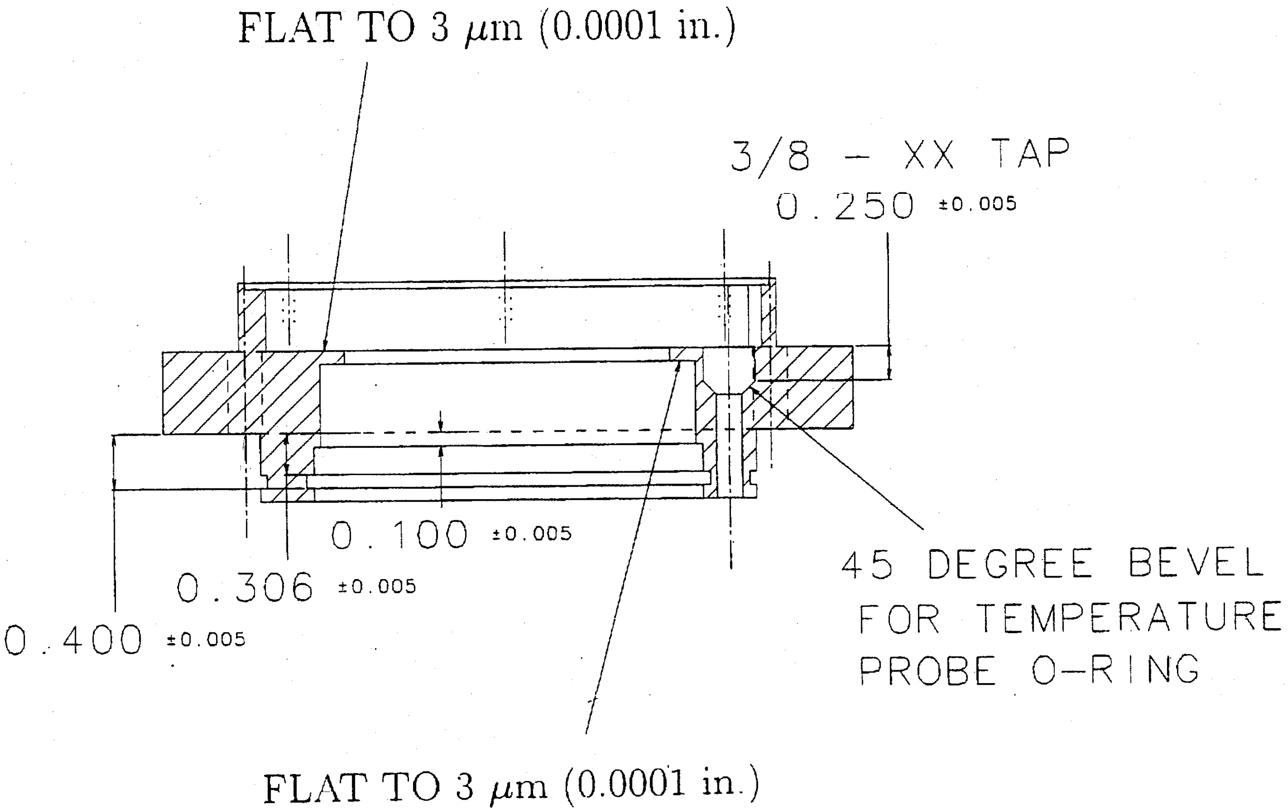


Figure C.6: Cutaway side view of coupling flange.

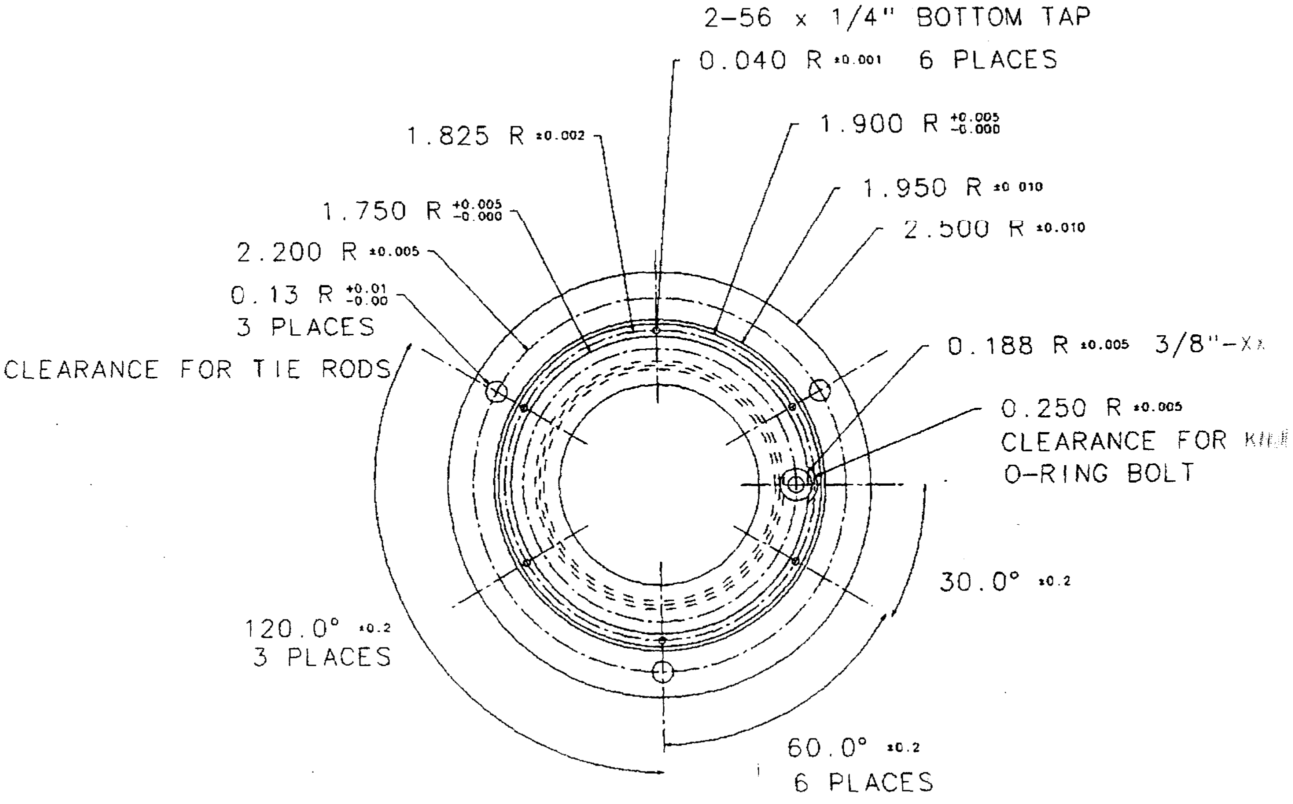


Figure C.7: Top view of coupling flange.

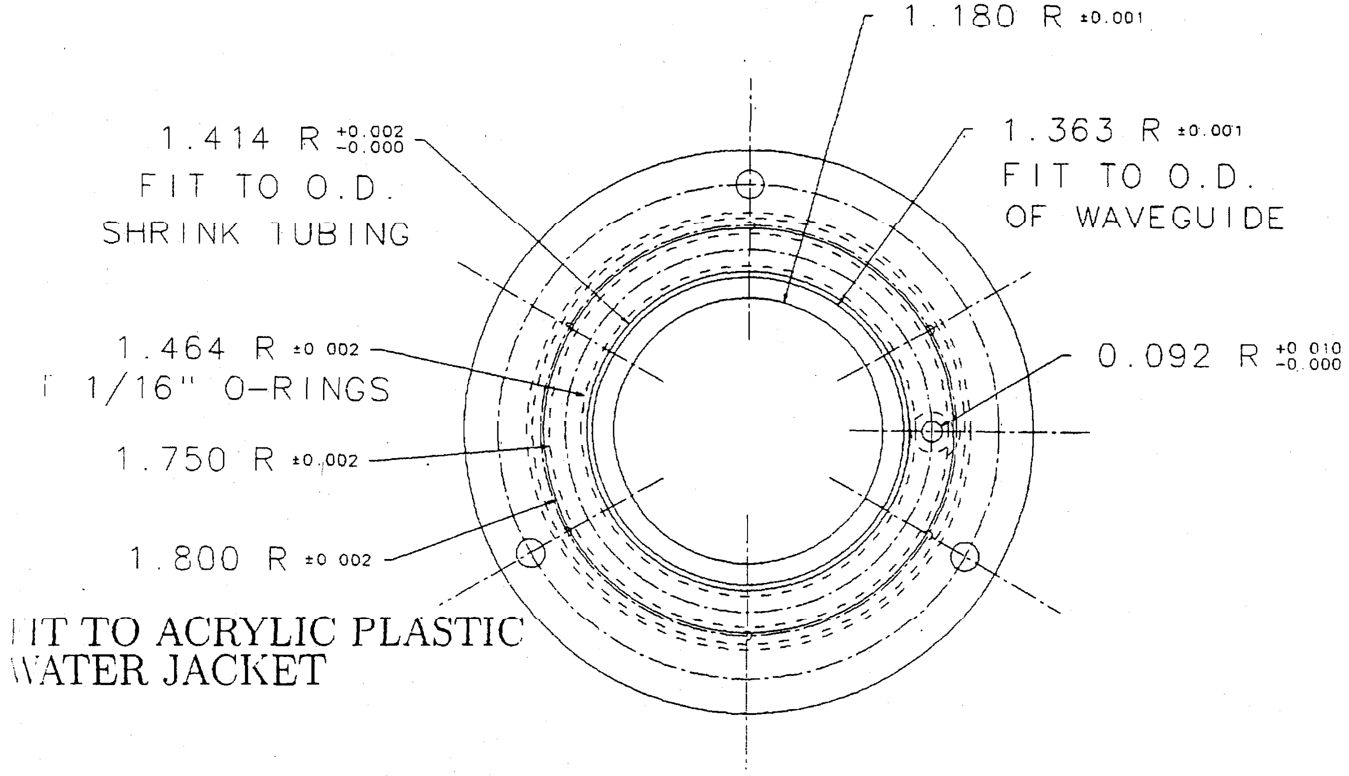
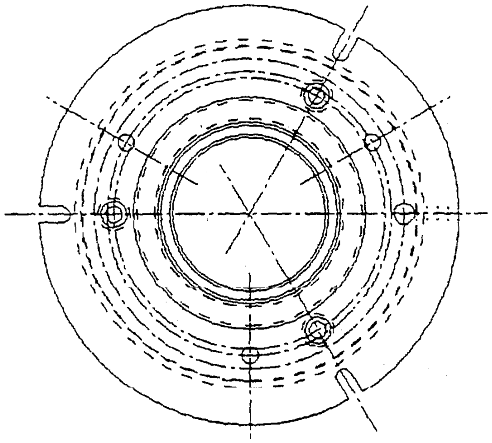
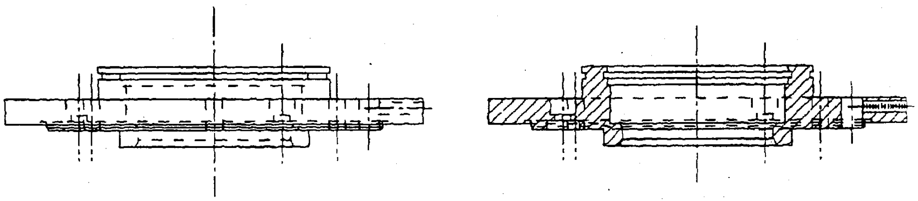


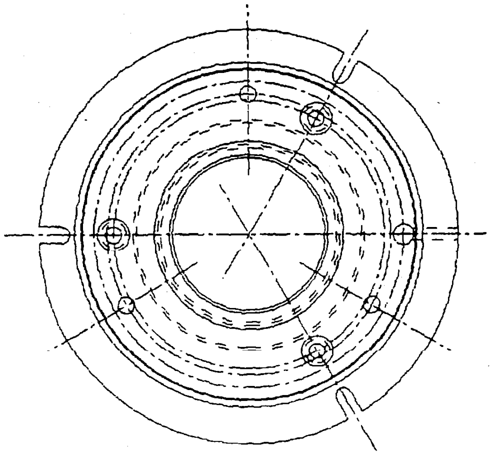
Figure C.8: Bottom view of coupling flange.



TOP VIEW



SIDE VIEW CUTAWAY SIDE VIEW



BOTTOM VIEW

Figure C.9: Orthogonal views of tuner flange.

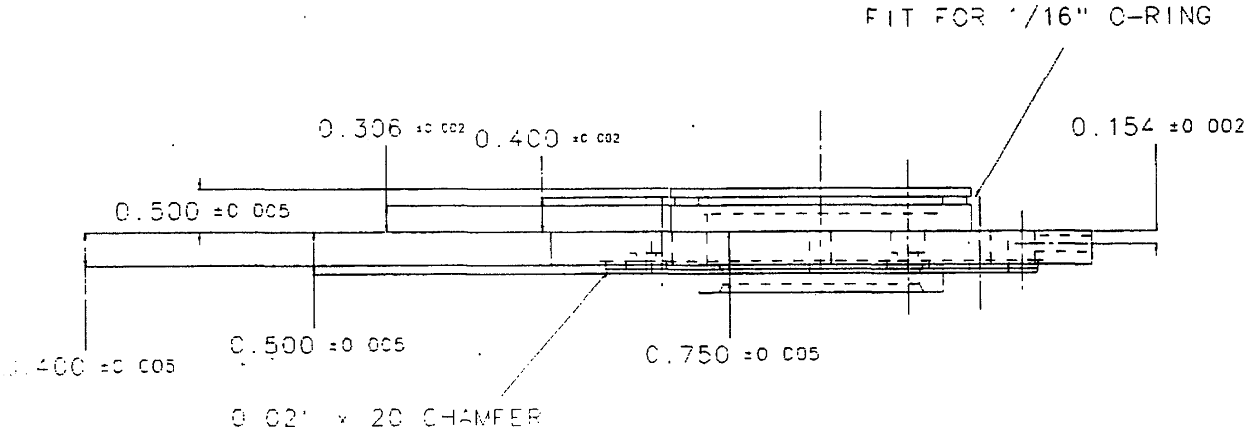


Figure C.10: Side view of coupling flange.

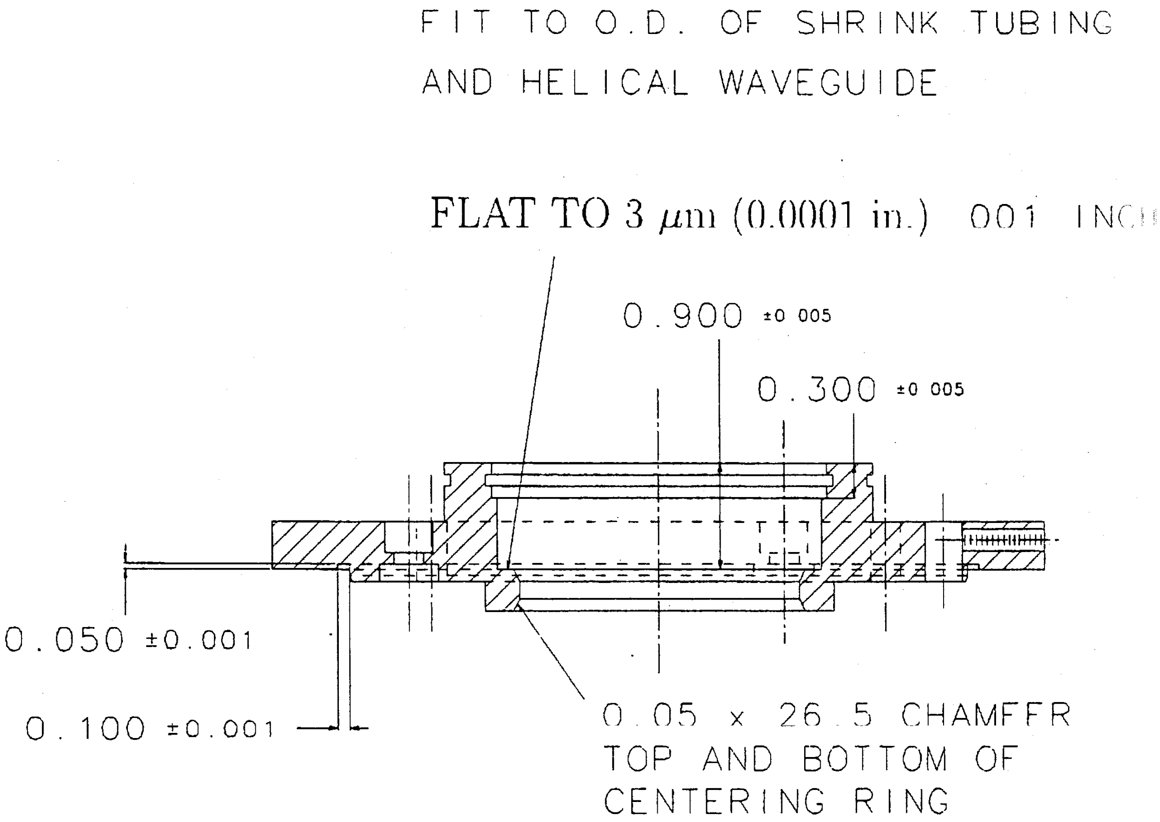


Figure C.11: Cutaway side view of coupling flange.

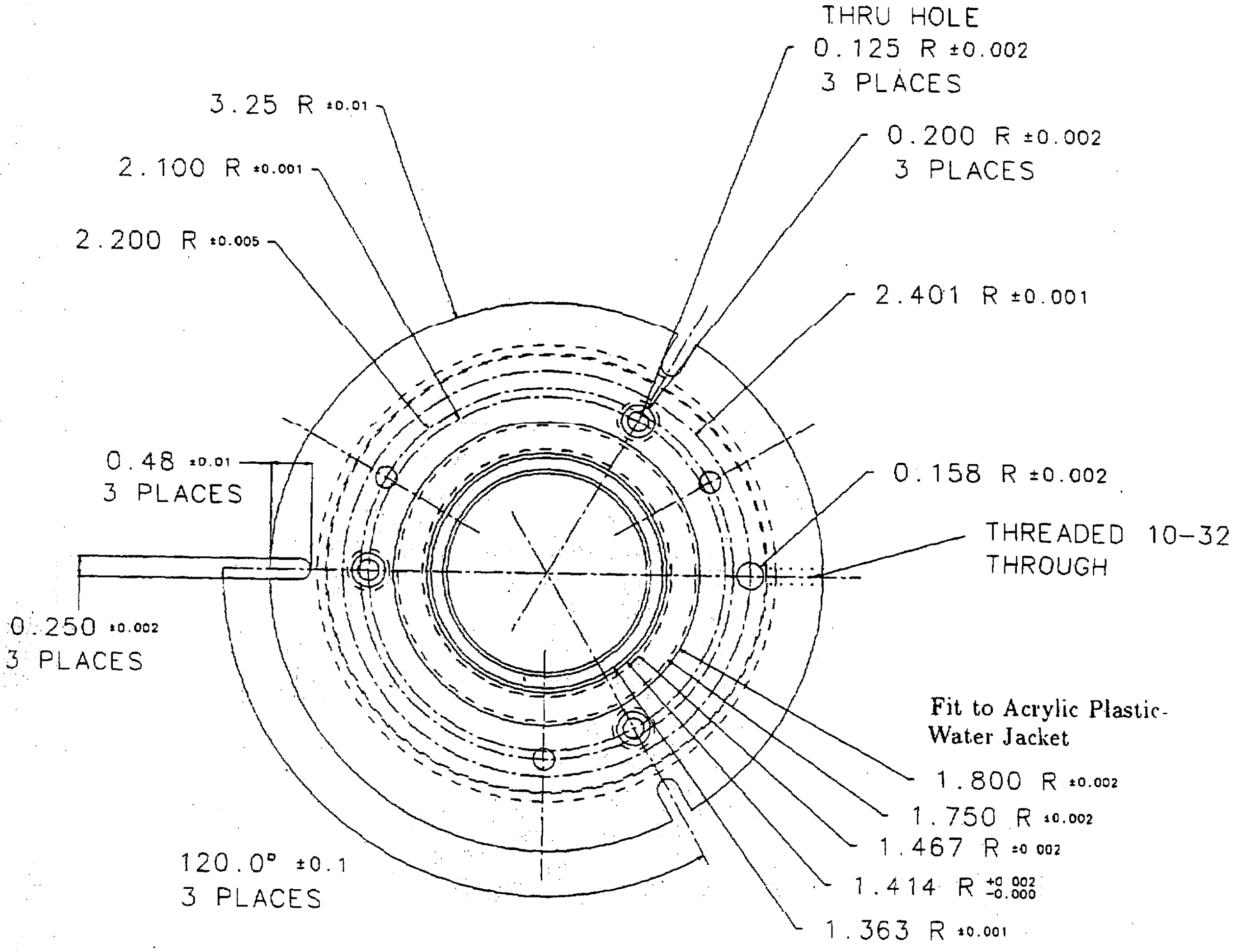


Figure C.12: Top view of tuner flange.

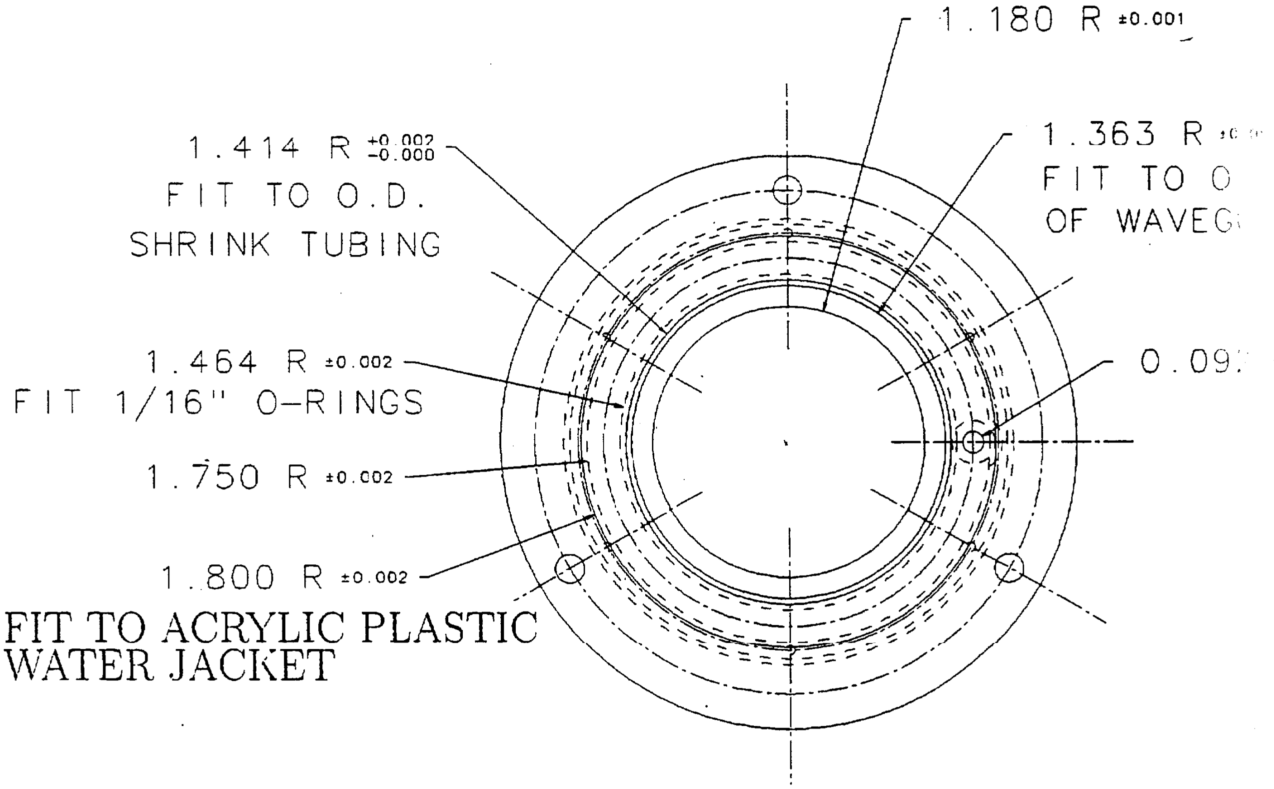
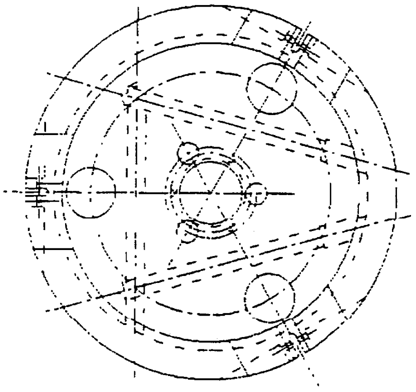
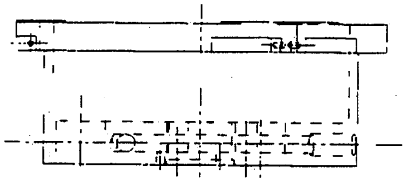


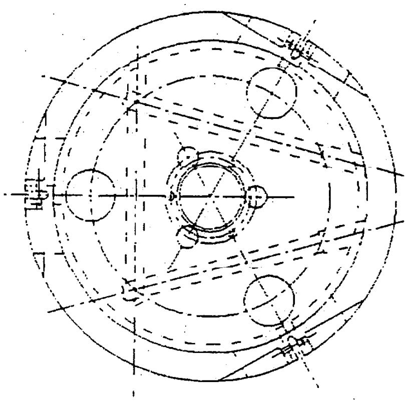
Figure C.13: Bottom view of tuner flange.



TOP VIEW



SIDE VIEW



BOTTOM VIEW

Figure C.14: Orthogonal views of tuner base.

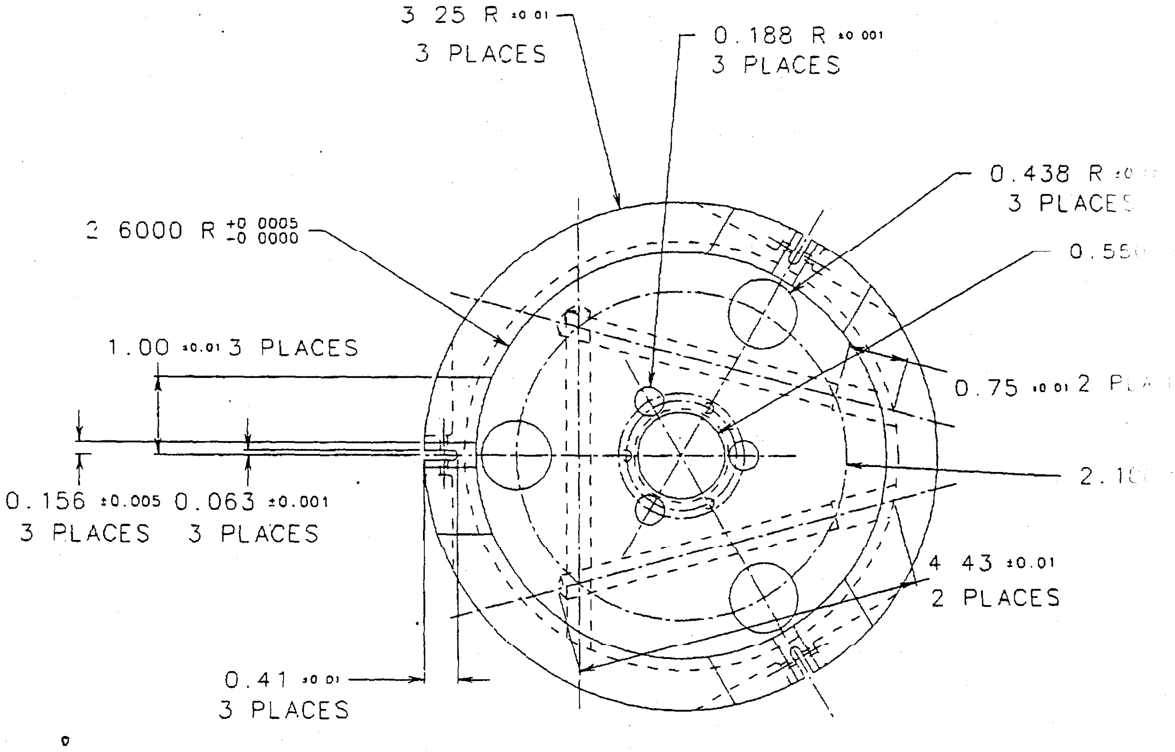


Figure C.15: Top view of tuner base.

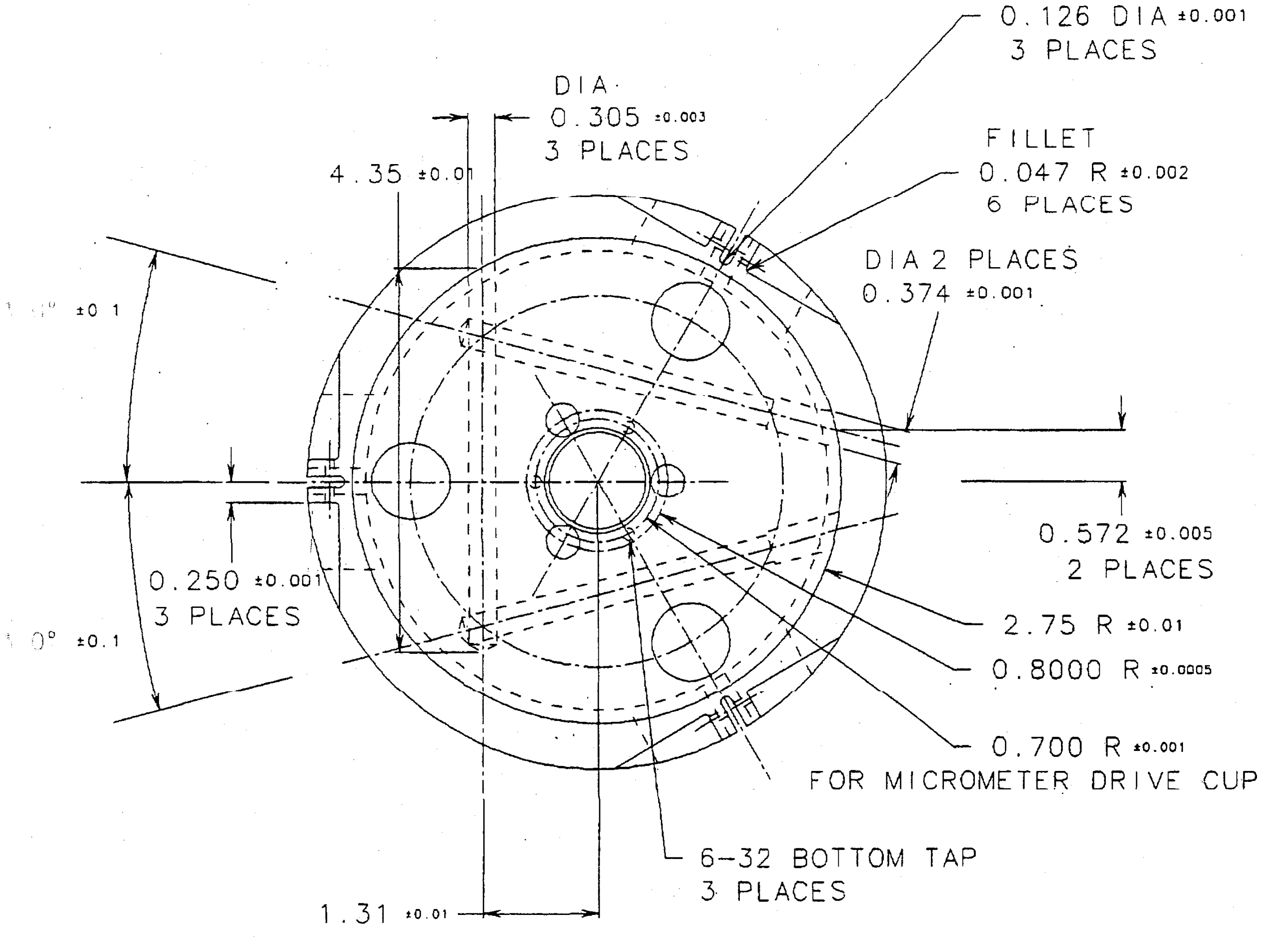


Figure C.16: Bottom view of tuner base.

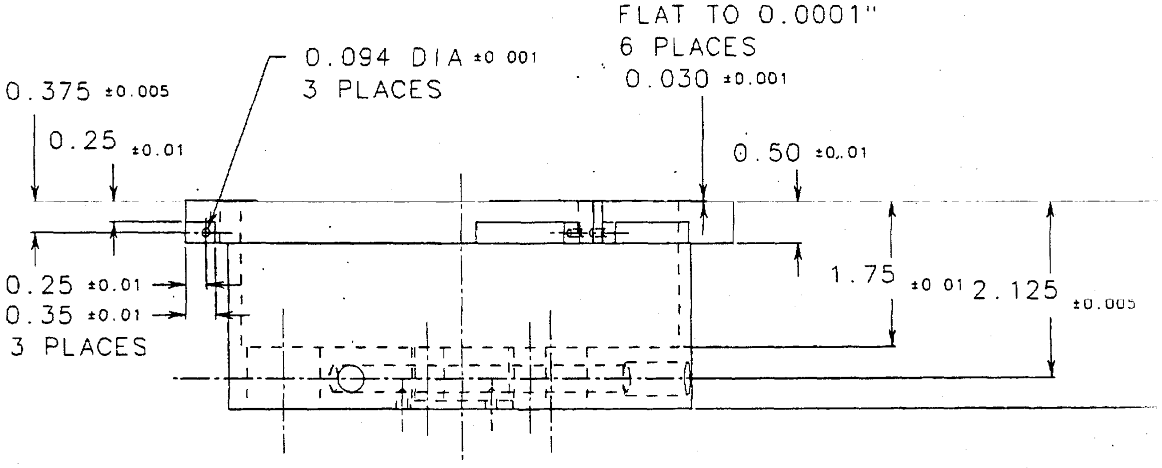


Figure C.17: Side view of tuner base.

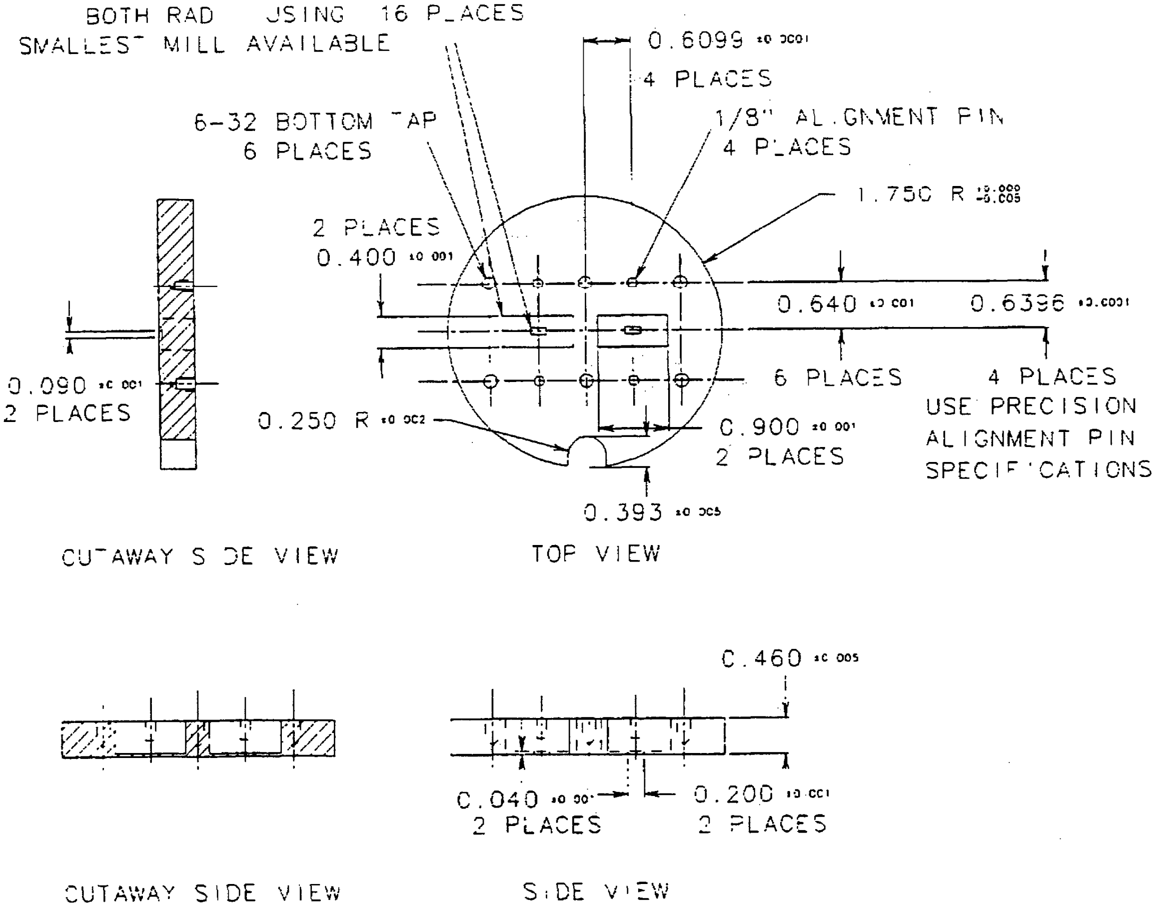
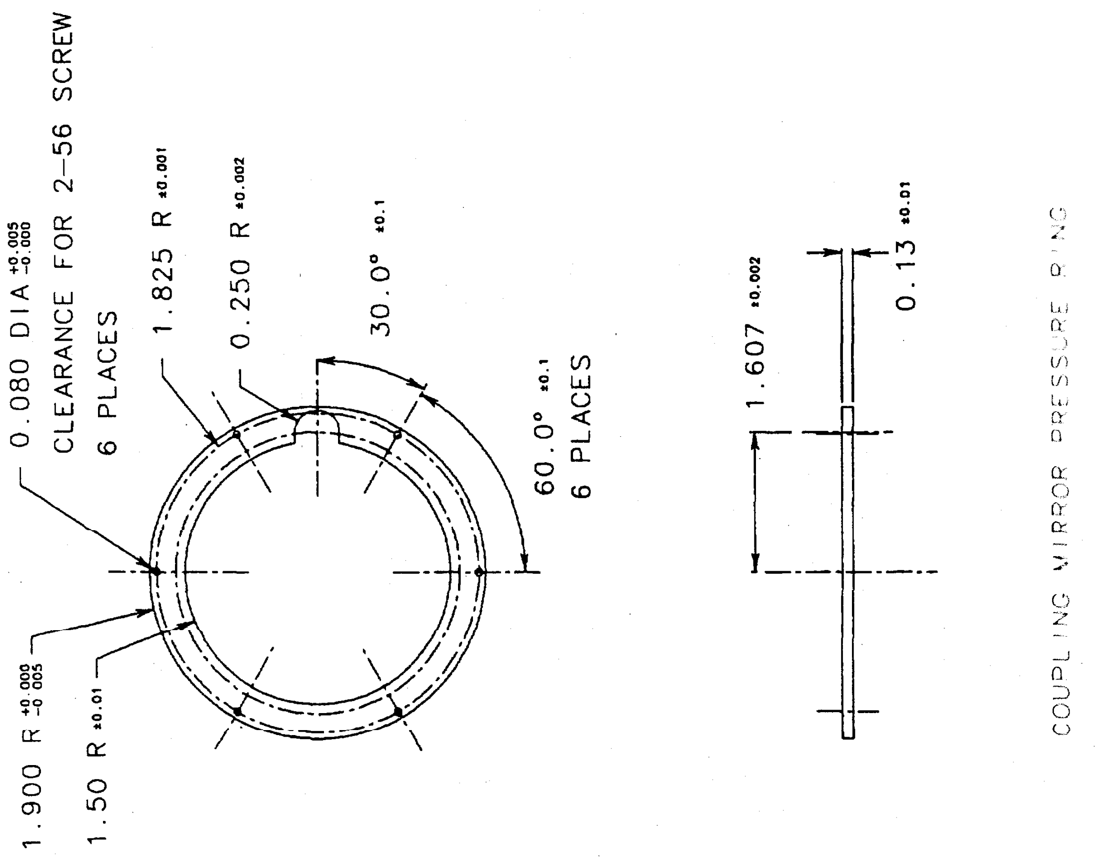


Figure C.18: Coupling endplate.



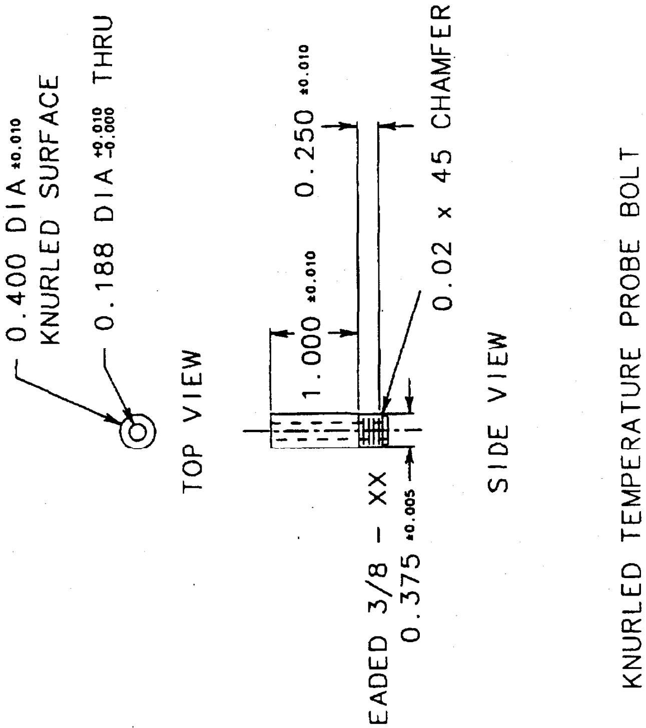


Figure C.19: Temperature-probe bolt and coupling-endplate pressure ring.

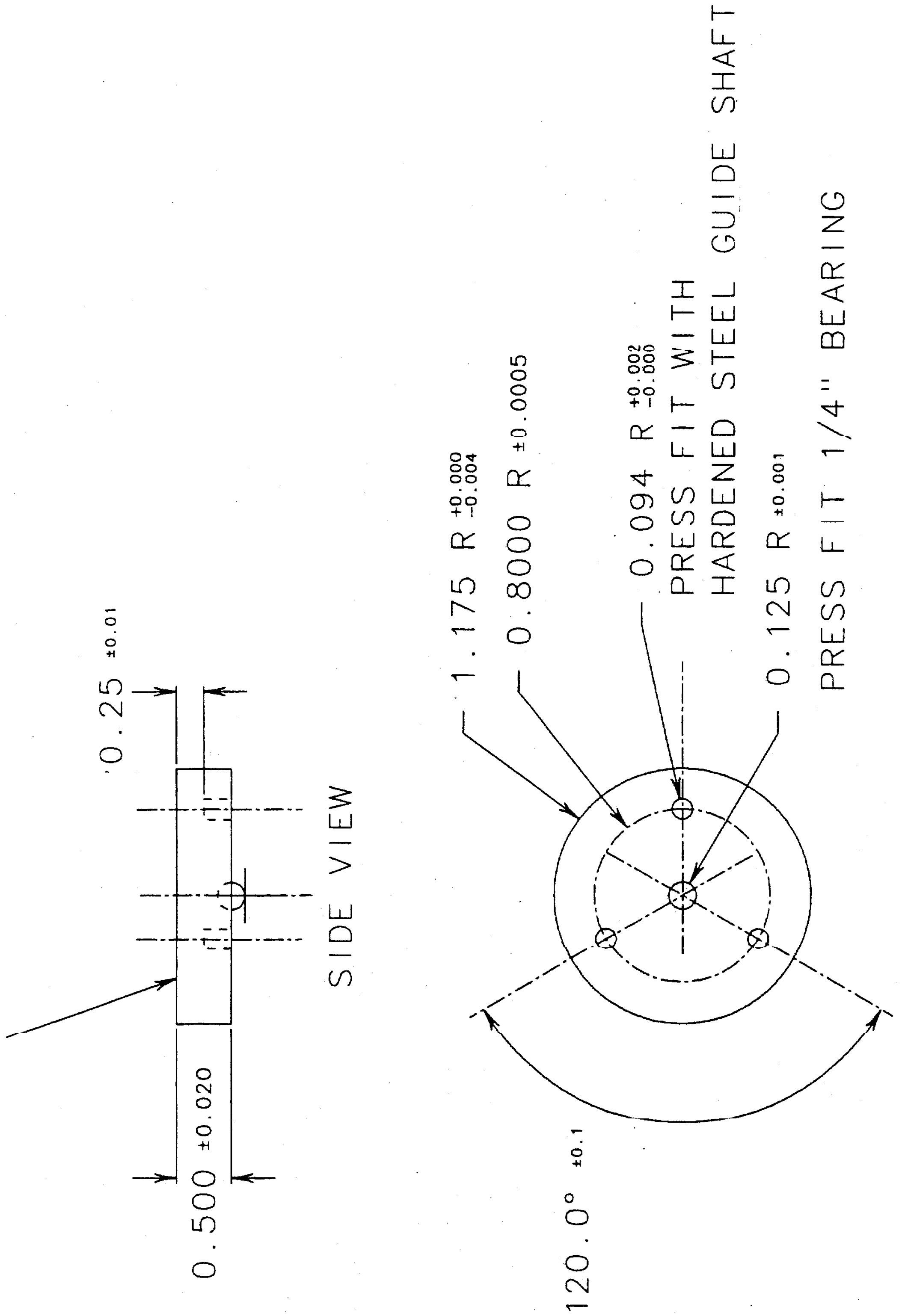
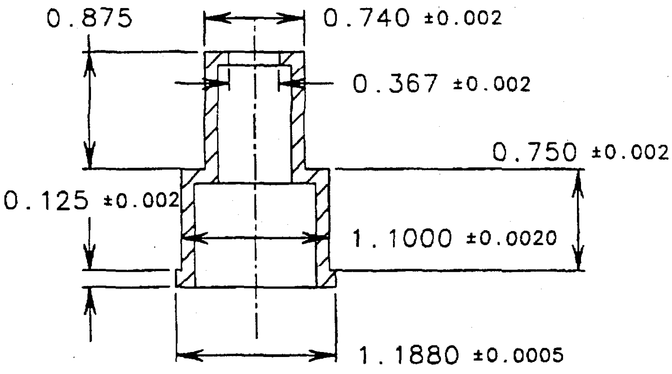
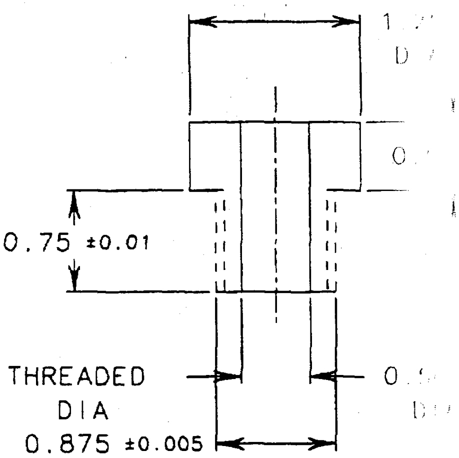


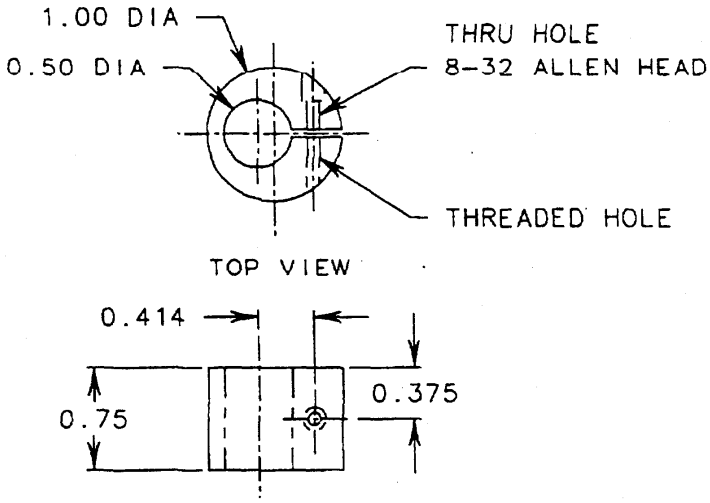
Figure C.20: Tuner-endplate.



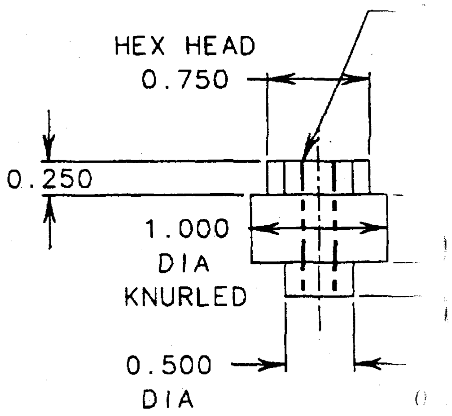
TYPICAL DESIGN FOR
MICROMETER DRIVE HOLDER



BUSHING FOR TUNER

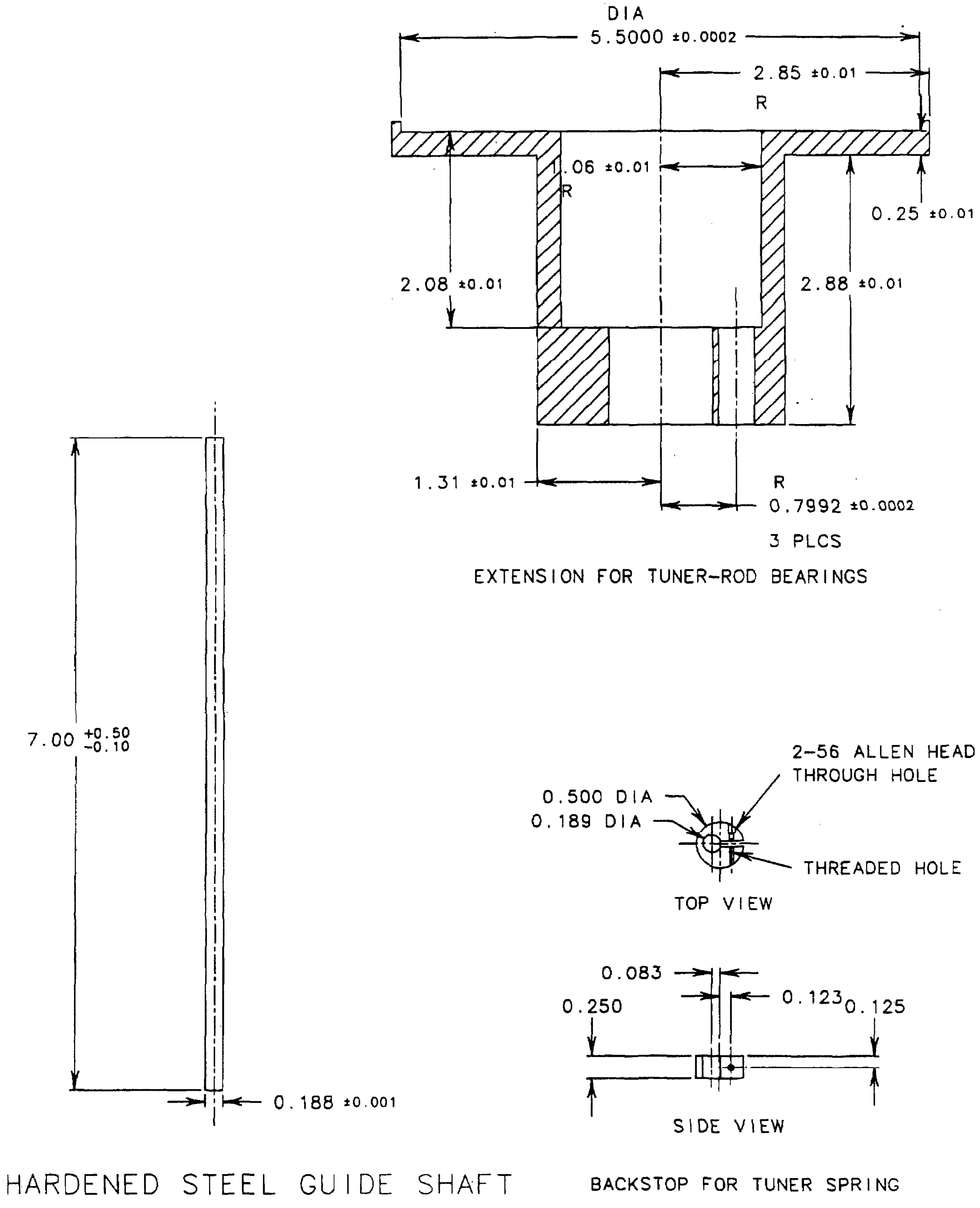


BACKSTOP FOR TUNER ASSEMBLY



TUNER BASE TORQU

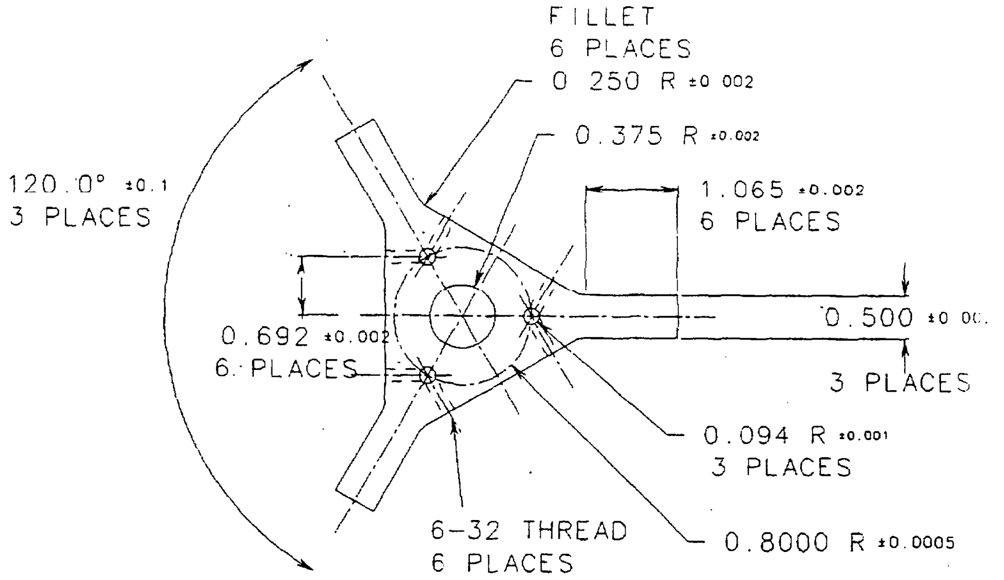
Figure C.21: Tuner assembly backstop, torque nut, micrometer-drive holder and sliding bushing.


EXTENSION FOR TUNER-ROD BEARINGS

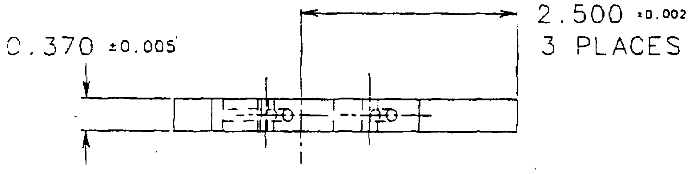
HARDENED STEEL GUIDE SHAFT

BACKSTOP FOR TUNER SPRING

Figure C.22: Tuner guide shaft, spring backstop and extension.



TOP VIEW



SIDE VIEW

Figure C.23: Measurement-micrometer reference yoke.

Other resonator parts not shown are:

1. 1 ea. 60-mm diameter helical waveguide.

   faced off perpendicular to axis.

   outside diameter turned to be concentric with inner diameter.

   ends flat and perpendicular to axis to within 3 μm (0.0001 in).

2. 1 ea. 4 in outside-diameter, 3.5 in inside-diameter acrylic plastic tubing with tangential inlet and outlet ducts at opposite ends.
3. 1 ea. plug for tuner-base water channel.
4. 6 ea. 2-56 machine screws for coupling-endplate pressure ring.
5. 3 ea. 2-56 set-screws for tuner-endplate.
6. 3 ea. Allen-head $\frac{1}{4} - 20$ bolts to attach stand to tuner flange.
7. 1 ea. 10-32 knurled screw for micrometer attachment.
8. 2 ea. $\frac{1}{16}$-in inside flange o-rings.
9. 2 ea. $\frac{1}{16}$-in outside flange o-rings.
10. 1 ea. temperature probe o-ring.
11. 3 ea. $\frac{1}{4}$-in diameter tie rods. Both ends threaded $\frac{1}{4} - 20$. For attaching coupling and tuner flanges.
12. 3 ea. washers and $\frac{1}{4} - 20$ tie rods.
13. 1 ea. base for resonator stand. 18-in diameter, 1-in thick aluminum with 3-in diameter center-hole and 3-$\frac{1}{2}$-in diameter holes located 120° apart at a 2.1-in radius.
14. 3 ea. stands to connect from base to tuner flange. 12-in long, $\frac{1}{2} \pm 0.01$ in diameter stainless steel rods.

Appendix D

MEAS_RES Program Listing

```
105!            PURGE "MEAS_RES"
110!            RE-STORE "MEAS_RES"
115     GOSUB Init_com
120     GOSUB Whatkindofmeas
125 !       Whatkindofmeas$="LIST","SWEEP", or "TRIGGER","DISKDATA"
130     SELECT Whatkindofmeas$
135     CASE "LIST"
140         GOSUB Measurelist
145     CASE "SWEEP"
150         GOSUB Measuresweep
155     CASE "TRIGGER"
160         GOSUB Measuretrigger
165     CASE "LISTTRIGGER"
170         GOSUB Measurelistrig
175     CASE "DUMMY"
180         GOSUB Measuredumdata
185     CASE "DISKDATA"
190         GOSUB Measurediskdata
195     CASE "BAILOUT"
200     CASE ELSE
205         DISP "Hey! ";Whatkindofmeas$;" isn't a measurement option. PROGRAM ST
 => OPPED."
210         BEEP
215         STOP
220     END SELECT
225     GOTO End_program
230  !
235  ! ///////////////////////////////////////////////////////////
240  !
245 Whatkindofmeas: !
250     Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
255     CLEAR SCREEN
260     PRINT "This program allows you to read resonance data from a network anal
 => yzer".
265     PRINT "There are three ways for you to do this:"
270     PRINT
275     PRINT "1) Measure resonances at frequencies given in DATA statements."
280     PRINT "   Edit line label 'Resonance_data' and enter the frequencies befo
 => re running."
285     PRINT "2) Sweep a band of frequencies, searching for resonances."
290     PRINT "   Edit line label 'Init_sweep' to change the sweep parameters."
295     PRINT "3) Measure resonances that you set up manually on the network anal
 => yzer."
300     PRINT "   Simply find a resonance on the screen and press a softkey to me
 => asure."
301 Menu: Reset_keys=0
303     IF Printer_on THEN
304     PRINT TABXY(1,12);"Printer is ON "
305     ELSE
306     PRINT TABXY(1,12);"Printer is OFF"
307     END IF
308     DISP "Which kind of measurement do you want to perform?"
310     ON KEY 1 LABEL "End    Program",Prty GOSUB Bailout
315     ON KEY 2 LABEL "DATA   LIST",Prty GOSUB Datalist
320     ON KEY 3 LABEL "SWEEP  NWA",Prty GOSUB Sweep
325     ON KEY 4 LABEL "MEAS ONTRIGGER",Prty GOSUB Trigger
330     ON KEY 5 LABEL "DUMMY  DATA",Prty GOSUB Dummydata
335     ON KEY 5 LABEL "DISK   DATA",Prty GOSUB Diskdata
340     ON KEY 6 LABEL "LIST ->TRIGGER",Prty GOSUB Measurelisttrig
345     ON KEY 7 LABEL "DELSUB MODESUB",Prty GOSUB Delsub_modesub
346     IF NOT Printer_on THEN
348     ON KEY 8 LABEL "Turn ONPrinter",Prty GOSUB Toggle_printer
349     ELSE
350     ON KEY 8 LABEL "TurnOFFPrinter",Prty GOSUB Toggle_printer
351     END IF
352     Selected=0
```

```
355     LOOP
360     EXIT IF Selected
361     IF Reset_keys THEN GOTO Menu
365     END LOOP
370     OFF KEY
375     CLEAR SCREEN
380     RETURN
385  !
390  ! ///////////////////////////////////////////////////////////
395  !
396 Abort: Abort=1
397         RETURN
400 Bailout:Selected=1
405     Whatkindofmeas$="BAILOUT"
410     RETURN
415  !
420 Datalist:Selected=1
425     Whatkindofmeas$="LIST"
430     RETURN
435  !
440 Sweep:Selected=1
445     Whatkindofmeas$="SWEEP"
450     RETURN
455  !
460 Trigger:Selected=1
465     Whatkindofmeas$="TRIGGE
470     RETURN
475  !
480 Dummydata:Selected=1
485     Whatkindofmeas$="DUMMY"
490     RETURN
495 Diskdata:Selected=1
500     Whatkindofmeas$="DISKDATA"
505     RETURN
510 Delsub_modesub: !
515     DISP "PLEASE WAIT WHILE THE ROUTINES ARE DELETED....."
520     DELSUB Resonance_data
525     DELSUB Empty_cavity
530     DELSUB Loaded_cavity
535     DELSUB Var_freq_empty
540     DELSUB Var_freq_loaded
545     DELSUB Var_len_empty
550     DELSUB Var_len_loaded
555     DELSUB Calc_c
560     DELSUB Calc_mode_freq
565     DISP
570     CLEAR SCREEN
575     RETURN
576  !
577  ! ///////////////////////////////////////////////////////////
578  !
579 Toggle_printer: OFF KEY 8
580     IF Printer_on THEN
581       PRINT TABXY(1,12);"Printer is OFF"
582       Printer_on=0
584       ELSE
585       PRINT TABXY(1,12);"Printer is ON "
586       Printer_on=1
588       END IF
589       Reset_keys=1
590       RETURN
591  !
592  ! ///////////////////////////////////////////////////////////
593  !
595 Measurelist:OFF KEY
600     Sss=1
```

```
605     Speed=0
610     MAT Te01= (0)
615     MAT Te02= (0)
620     MAT Te12= (0)
625     MAT Te13= (0)
630     MAT Freq= (0)
635     GOSUB Resonance_data
640     FOR I=1 TO Num_peaks
645         GOSUB Get_resonance
650         CALL Zoom_on_peak
655         GOSUB Measure_sparms
660         CALL Optimizedisplay
665         GOSUB Set_startstop
670         CALL Fit_sparms
675         CALL Phase_delay
680         CALL Fit_sparms
685     NEXT I
690     IF Printer_on THEN OUTPUT Print_addr;CHR$(12);
695     RETURN
700  !
705  ! ////////////////////////////////////////////////////////////
710  !
715 Measurelisttrig:OFF KEY
720     Sss=1
725     RESTORE Resonance_data
730     READ Num_peaks
735     FOR I=1 TO Num_peaks
740         READ Freq(I)
745     NEXT I
750     FOR I=1 TO Num_peaks
755         GOSUB Give_display
760         CALL Zoom_on_peak
765         GOSUB Measure_sparms
770         CALL Optimizedisplay
775         GOSUB Set_startstop
780         CALL Fit_sparms
785     NEXT I
790     IF Printer_on THEN OUTPUT Print_addr;CHR$(12);
795     RETURN
800  !
805  ! ////////////////////////////////////////////////////////////
810  !
815 Give_display: !
820     CALL Set_nwa("SWEEP=RAMP","S=21")
825     CALL Set_nwa("SPAN=1E5","SCALE=10","CENTER="&VAL$(Freq(I)))
830     CALL Set_nwa("REFV=-40","AVER=OFF","NUMG=1")
835     CALL Optimizedisplay
840     CALL Set_nwa("SWEEP=CONT")
845     Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
850     ON KEY 6 LABEL "READY  TO MEAS",Prty GOTO Get_back
855 Sitnspin:GOTO Sitnspin
860 Get_back:OFF KEY
865     CALL Set_nwa("SPAN=3E7")
870     RETURN
875  !
880  ! ////////////////////////////////////////////////////////////
885  !
890 Get_resonance: !
895     CALL Set_nwa("SWEEP=CONT","SWEEP=RAMP","S=21")
900     CALL Set_nwa("SPAN=3E7","SCALE=5","CENTER="&VAL$(Freq(I)))
905     CALL Set_nwa("REFV=-40","AVER=16","NUMG=17")
910     RETURN
915  ! ////////////////////////////////////////////////////////////
920  !
925 Find_resonance: !
930     CALL Set_nwa("SWEEP=CONT","SWEEP=RAMP","S=21")
935     CALL Set_nwa("SPAN=3E7","SCALE=5","CENTER="&VAL$(Freq(I)))
940     CALL Set_nwa("REFV=-40","AVER=16","NUMG=17")
945     Set_nwa("MARK=CONT","MARK=1","MARK=MAX")
950     Read_nwa("MARK=1")
955     Delta=-3
960     Set_nwa("DEL REF=1","MARKER TARGET="&VAL$(Delta))
965     Set_nwa("MARK=2","MARK=MAX","MARKER=TARGET","MARKER=SEARCH LEFT")
970     Set_nwa("MARK=3","MARK=MAX","MARKER=TARGET")
975     Read_nwa("MARK=2","MARK=3")
980     Set_nwa("DELTA OFF=")
985     IF ABS(Marker(2,1))>5000 AND ABS(Marker(3,1))>5000 THEN
990         IF ABS(Marker(2,1))<ABS(.9*Marker(3,1)) THEN
995             Span=2*ABS(Marker(3,1))
1000        ELSE
1005            Span=2*ABS(Marker(2,1))
1010        END IF
1015        CALL Zoom_on_peak
1020    ELSE
1025        Span=999999
1030        BEEP
1035        PRINT "Span cannot be determined from marker information."
1040        PRINT "No resonance was found at ";Freq(I)/1.E+9;" GHz."
1045        IF Printer_on THEN OUTPUT Print_addr;"No resonance was found at ";Fre
 => q(I)/1.E+9;" GHz."
1050        DISP "No resonance was found at ";Freq(I)/1.E+9;" GHz."
1055!           PAUSE
1060    END IF
1065    RETURN
1070 !
1075 ! ////////////////////////////////////////////////////////////
1080 !
1085 Measuresweep:OFF KEY
1090    CALL Sweep_nwa
1095    RETURN
1100 !
1105 ! ////////////////////////////////////////////////////////////
1110 !
1115 Measuretrigger:OFF KEY
1120    Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
1125    DISP "When you have a resonance on the network analyzer CRT, press 'READ
 => PEAK'."
1130    ON KEY 0 LABEL "END    PROG",Prty GOTO Triggerdone
1135    ON KEY 1 LABEL "READ   S11",Prty GOSUB Read_s11res
1140    ON KEY 2 LABEL "READ   S21",Prty GOSUB Read_s21res
1145    ON KEY 3 LABEL "READ   S12",Prty GOSUB Read_s12res
1150    ON KEY 4 LABEL "READ   S22",Prty GOSUB Read_s22res
1151    ON KEY 5 LABEL "READ   S11/S22",Prty GOSUB Read_s11s22res
1152    ON KEY 15,Prty GOSUB Read_allsparms
1155 !
1160    ON KEY 6 LABEL "ZOOM   S11",Prty GOSUB Zoomread_s11res
1165    ON KEY 7 LABEL "ZOOM   S21",Prty GOSUB Zoomread_s21res
1170    ON KEY 8 LABEL "ZOOM   S12",Prty GOSUB Zoomread_s12res
1175    ON KEY 9 LABEL "ZOOM   S22",Prty GOSUB Zoomread_s22res
1180    Reset_keys=0
1185    LOOP
1190        IF Reset_keys THEN GOTO Measuretrigger
1195    END LOOP
1200 Triggerdone:OFF KEY
1205    RETURN
1210 !
1215 ! ////////////////////////////////////////////////////////////
1220 !
1225 Measurediskdata:OFF KEY
1230 Get_data: !
1235    DISP "Diskdata is in new or old format?"
1240    Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
```

```
1245   ON KEY 0 LABEL "ABORT THIS",Prty GOSUB Done_return
1250   ON KEY 2 LABEL "OLD   SPARMS",Prty GOSUB Get_old_sparms
1255   ON KEY 3 LABEL "NEW   SPARMS",Prty GOSUB Get_new_sparms
1260   Done=0
1265   Reset_keys=0
1270   REPEAT
1275       IF Reset_keys THEN GOTO Get_data
1280   UNTIL Done
1285   RETURN
1290 !
1295 Done_return:Done=1
1300   RETURN
1305 !
1310 ! ////////////////////////////////////////////////////////////
1315 !
1320 Get_old_sparms:OFF KEY
1325   CALL Load_sparms
1330 ! CALL Phase_delay
1335   RESTORE Diskdatas
1340   Speed=0
1345   Sselect=1   !S21
1350   Start_index=1
1355   Stop_index=Dcount
1360   GOSUB Diskdata_keys
1365   Done=1
1370   RETURN
1375 !
1380 ! ////////////////////////////////////////////////////////////
1385 !
1390 Get_new_sparms:OFF KEY
1395 ! ALLOCATE Desc$[160]
1400 ! REPEAT
1405 !     LINPUT "Enter a description of this data. When finished enter a null
  => string.",Desc$
1410 !     PRINT Desc$
1415 !     IF Printer_on THEN OUTPUT Print_addr;Desc$
1420 ! UNTIL Desc$=""
1425 ! DEALLOCATE Desc$
1430 ! N=6
1435 ! OUTPUT 2 USING "K,#";N
1440 ! INPUT "Please enter the number of files",N
1441   ON ERROR GOSUB Change_disk
1443   RESTORE Mode_numbers
1444   READ Num_settings
1445   FOR J=1 TO Num_settings
1446       READ Micr,Nummodes
1447       IF Printer_on THEN OUTPUT Print_addr;"Micr:";Micr;"  Nummodes:";Nummo
  => des;"  30 Nov 1990 Test of micrometer travel accuracy"
1448       PRINT "Micr:";Micr;"  Nummodes:";Nummodes;"  30 Nov 1990 Test of micr
  => ometer travel accuracy"
1449       FOR I=1 TO Nummodes
1453           Filename$=VAL$(Micr)&"30Nov90"&VAL$(I)
1455           Diskdrive$=":,1400,1"
1460           CALL New_load_sparms
1465           Sselect=1          !S21 is hardcoded
1470           Start_index=1
1475           Stop_index=Dcount
1480           CALL Fit_sparms
1485           Peak(I,1)=A(4)
1490           Peak(I,2)=SQRT(A(1)^2+A(2)^2)
1495           Peak(I,3)=A(4)/A(3)
1500       NEXT I
1505       IF Printer_on THEN OUTPUT Print_addr;CHR$(12);
1506       N=Nummodes
1510       ALLOCATE Basket(N,2),Id$[40]
1515       MAT Basket(*,1)= Peak(1:N,1)
```

```
1520       MAT Basket(*,2)= Peak(1:N,3)
1525       CLEAR SCREEN
1526       Filename$=VAL$(Micr)&"30Nov90"&VAL$(J)
1527       Id$=Filename$
1528       Diskdrive$=":,1404"
1530 ! CALL Select_disk
1535 ! IF Diskdrive$="NO DISK" THEN GOTO Skip_peak_save
1540 ! CALL Enterfilename("ABORT","SAVING resonance parameters in a file")
1545 ! IF Filename$="" THEN GOTO Skip_peak_save
1550 ! CALL Enter_id(Id$,"resonance parameter file")
1555 ! IF Id$="" THEN GOTO Skip_peak_save
1560       CALL Save_file(Basket(*),Nummodes,Id$)
1565 Skip_peak_save:DEALLOCATE Id$,Basket(*)
1566   NEXT J
1567   OFF ERROR
1570   Reset_keys=1
1575   RETURN
1580 !
1581 Change_disk:PRINT "please insert next disk"
1582   DISP ERRM$
1583   BEEP
1585   PAUSE
1586   RETURN
1587 ! ////////////////////////////////////////////////////////////
1590 !
1595 Diskdata_keys:OFF KEY
1600   GOSUB Diskdata_menu
1605   DISP "Which S-parameter would you like to fit?"
1610   Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
1615   ON KEY 1 LABEL "Bailout",Prty GOSUB Bailout
1620   ON KEY 6 LABEL "FIT    S11",Prty GOSUB Fit_s11
1625   ON KEY 7 LABEL "FIT    S12",Prty GOSUB Fit_s12
1630   ON KEY 8 LABEL "FIT    S21",Prty GOSUB Fit_s21
1635   ON KEY 9 LABEL "FIT    S22",Prty GOSUB Fit_s22
1636   ON KEY 3 LABEL "SAVE   SPARMS",Prty GOSUB Save_new_sparms
1640   ON KEY 4 LABEL "START  INDEX",Prty GOSUB Fix_start_index
1645   ON KEY 5 LABEL "STOP   INDEX",Prty GOSUB Fix_stop_index
1650   Done=0
1655   Selected=0
1660   LOOP
1665       IF Done THEN GOTO Diskdata_keys
1670   EXIT IF Selected
1675   END LOOP
1680   OFF KEY
1685   CLEAR SCREEN
1690   RETURN
1695 !
1700 Fit_s11: !
1705   Sss=2
1710   Sparm$="S11"
1715   MAT Sparm= S11
1720   MAT Frequency= Freq
1725   CALL Fit_sparms
1730   Done=1
1735   RETURN
1740 Fit_s12: !
1745   Sss=1
1750   Sparm$="S12"
1755   MAT Sparm= S12
1760   MAT Frequency= Freq
1765   CALL Fit_sparms
1770   Done=1
1775   RETURN
1780 Fit_s21: !
1785   Sss=1
1790   Sparm$="S21"
```

```
1795   MAT Sparm= S21
1800   MAT Frequency= Freq
1805   CALL Fit_sparms
1810   Done=1
1815   RETURN
1820 Fit_s22: !
1825   Sss=2
1830   Sparm$="S22"
1835   MAT Sparm= S22
1840   MAT Frequency= Freq
1845   CALL Fit_sparms
1850   Done=1
1855   RETURN
1860 Fix_start_index:Si=Start_index
1865   CALL Test_real(1,Stop_index-1.,"Start_index",Si)
1870   Start_index=Si
1875   Done=1
1880   RETURN
1885 Fix_stop_index:Si=Stop_index
1890   CALL Enter_real(Start_index+1.,Dcount,"Stop_index",Si)
1895   Done=1
1900   RETURN
1905 Diskdatas:DATA 1,33,34,65,67,99,100,132,133,165,166,198,199,231,232,264,265
 => ,297,298,330,331,363,364,396
1910   RETURN
1915 !
1916 Save_new_sparms:OFF KEY
1917   CLEAR SCREEN
1918   Ssave=Svalid
1920   Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
1921 Setup_save:DISP "Which S-parameters do you want to save?"
1922   IF NOT BIT(Svalid,0) THEN
1923       PRINT TABXY(0,1);"S11 is not valid, cannot be saved"
1924   ELSE
1925       IF BIT(Ssave,0) THEN
1926           PRINT TABXY(0,1);CHR$(129);"S11 will be saved    ";CHR$(128)
1927           ON KEY 6 LABEL "noSave S11",Prty GOSUB Toggle_s11
1928       ELSE
1929           PRINT TABXY(0,1);"S11 will NOT be saved"
1930           ON KEY 6 LABEL "  Save S11",Prty GOSUB Toggle_s11
1931       END IF
1932   END IF
1933 !
1934   IF NOT BIT(Svalid,1) THEN
1935       PRINT TABXY(0,2);"S21 is not valid, cannot be saved"
1936   ELSE
1937       IF BIT(Ssave,1) THEN
1939           PRINT TABXY(0,2);CHR$(129);"S21 will be saved    ";CHR$(128)
1940           ON KEY 7 LABEL "noSave S21",Prty GOSUB Toggle_s21
1941       ELSE
1942           PRINT TABXY(0,2);"S21 will NOT be saved"
1943           ON KEY 7 LABEL "  Save S21",Prty GOSUB Toggle_s21
1944       END IF
1945   END IF
1947 !
1948   IF NOT BIT(Svalid,2) THEN
1949       PRINT TABXY(0,3);"S12 is not valid, cannot be saved"
1950   ELSE
1952       IF BIT(Ssave,2) THEN
1953           PRINT TABXY(0,3);CHR$(129);"S12 will be saved    ";CHR$(128)
1954           ON KEY 8 LABEL "noSave S12",Prty GOSUB Toggle_s12
1955       ELSE
1956           PRINT TABXY(0,3);"S12 will NOT be saved"
1957           ON KEY 8 LABEL "  Save S12",Prty GOSUB Toggle_s12
1958       END IF
1959   END IF
```

```
1961 !
1962   IF NOT BIT(Svalid,3) THEN
1963       PRINT TABXY(0,4);"S22 is not valid, cannot be saved"
1964   ELSE
1965       IF BIT(Ssave,3) THEN
1967           PRINT TABXY(0,4);CHR$(129);"S22 will be saved    ";CHR$(128)
1968           ON KEY 9 LABEL "noSave S22",Prty GOSUB Toggle_s22
1969       ELSE
1970           PRINT TABXY(0,4);"S22 will NOT be saved"
1971           ON KEY 9 LABEL "  Save S22",Prty GOSUB Toggle_s22
1972       END IF
1973   END IF
1974   ON KEY 1 LABEL "ABORT",Prty GOSUB Abort
1975   ON KEY 2 LABEL "ALL OK",Prty GOSUB Setup_done
1977 !
1978   Reset_keys=0
1979   Done=0
1980   LOOP
1981       IF Reset_keys THEN GOTO Setup_save
1982   EXIT IF Done
1983   END LOOP
1984   Done=0
1985   Diskdrive$=":,1400,1"
1986   CALL Enterfilename("VALID")
1988   CALL New_save_sparms(Ssave)
1989   Done=1
1990   RETURN
1991 !
1992 Setup_done:OFF KEY
1993           Done=1
1994           RETURN
1995 !
1996 ! ///////////////////////////////////////////////////////////////
1997 !
1998 Toggle_s11:!
1999       IF BIT(Ssave,0) THEN
2000       Ssave=Ssave-1
2001       ELSE
2002       Ssave=Ssave+1
2003       END IF
2004       Reset_keys=1
2005       RETURN
2006 !
2007 Toggle_s21:!
2008       IF BIT(Ssave,1) THEN
2009       Ssave=Ssave-2
2010       ELSE
2011       Ssave=Ssave+2
2012       END IF
2013       Reset_keys=1
2014       RETURN
2015 !
2016 Toggle_s12:!
2017       IF BIT(Ssave,2) THEN
2018       Ssave=Ssave-4
2019       ELSE
2020       Ssave=Ssave+4
2021       END IF
2022       Reset_keys=1
2023       RETURN
2024 !
2025 Toggle_s22:!
2026       IF BIT(Ssave,3) THEN
2027       Ssave=Ssave-8
2028       ELSE
2029       Ssave=Ssave+8
```

```
2030        END IF
2031        Reset_keys=1
2032        RETURN
2033 !
2034 ! ////////////////////////////////////////////////////////////
2035 !
2036 Diskdata_menu:PRINT TABXY(1,1)
2037              PRINT "S PARAMETER FITTING FROM DISK DATA"
2038              PRINT
2039              PRINT USING Image1;Filename$&Diskdrive$
2040 ! PRINT USING Image2;Title$
2041 ! PRINT USING Image6;DATE$(DATE(Measure_time$))
2042              PRINT
2043              PRINT USING Image3;Dcount
2044              PRINT
2045              PRINT USING Image4;Start_index
2046              PRINT USING Image5;Stop_index
2047 Image1:      IMAGE"FILENAME          : ",K
2048 Image2:      IMAGE"DESCRIPTION       : ",K
2049 Image3:      IMAGE"DATACOUNT         : ",3D
2050 Image4:      IMAGE"START  INDEX      : ",3D
2051 Image5:      IMAGE"STOP   INDEX      : ",3D
2052 Image6:      IMAGE"DATE              : ",K
2053              RETURN
2054 ! ////////////////////////////////////////////////////////////
2055 !
2056 Measuredumdata:OFF KEY
2057              Span=2.E+5
2058              F0=(1.200E+10)
2059              Q0=90000
2060              Ql=79000
2061              G0=F0/Q0
2062              Gl=F0/Ql
2063              Degrees=38
2064              Mag=.7
2065              Num_points=50
2066              Start_index=1
2067              Stop_index=50
2068 !        ALLOCATE X(Num_points),COMPLEX S21(Num_points),
2069              FOR I=1 TO Num_points
2070                  X(I)=Span*I/Num_points+F0-(Span/2)
2071                  Frequency(I)=X(I)
2072                  Ql2f=2*Ql*(X(I)-F0)/F0
2073                  P=CMPLX(COS(Degrees),SIN(Degrees))
2074 !            S21(I)=Mag*(1-G0/Gl)*P/CMPLX(1,2*(X(I)-F0)/Gl)
2075 !            S11(I)=Mag*CMPLX(Ql/Q0+Ql2f*Ql2f,Ql2f*(1-Ql/Q0))*P/(1+Ql2f*Ql2f)
2076 !        S21(I)=(1-(Ql/Q0))/CMPLX(1,2*Ql*(F(I)-F0)/F0)
2077 !        S11(I)=Mag*CMPLX(Ql/Q0,2*Ql*(F(I)-F0)/F0)/CMPLX(1,2*Ql*(F(I)-F0)/F0)*
  => CMPLX(COS(Degrees),SIN(Degrees))
2078                  Rndmag=.0001*(RND-.5)
2079                  Rndphase=.001*(RND-.5)
2080                  Sparm(I)=S21(I)+Rndmag*(CMPLX(COS(Rndphase),SIN(Rndphase)))
2081              NEXT I
2082              CALL Fit_sparms
2083 ! PRINT USING "2(K,MD.3DE,2X),K,MD.9DE";"Q0=",A(1),"Ql=",A(2),"F0=",A(3)
2084 ! PRINT USING "2(K,MD.3DE,2X),K,MD.9DE";"UQ0=",Ua,"UQl=",Ub,"UF0=",Uc
2085              RETURN
2086 !
2087 ! ////////////////////////////////////////////////////////////
2088 !
2089 Read_s11s22res:OFF KEY
2090                  INPUT "ENTER THE NUMBER OF AVERAGES",Avs
2091                  OUTPUT 716;"AVERON "&VAL$(Avs)
2092 !
2093                  OUTPUT 716;"S11;"
2095                  OUTPUT 716;"CORRON;CALS4;"
```

```
2096                  GOSUB Read_s11res
2097 !
2098 !                OUTPUT 716;"CORRON;CALS4;"
2099 !                GOSUB Read_s21res
2100 !
2101                  OUTPUT 716;"S22;"
2103                  OUTPUT 716;"CORRON;CALS5;"
2104                  GOSUB Read_s22res
2105 !
2106 !                OUTPUT 716;"CORRON;CALS4;"
2107 !                GOSUB Read_s12res
2108 !
2110                  GOSUB Save_new_sparms
2111                  Reset_keys=1
2112                  RETURN
2113 !
2114 ! ////////////////////////////////////////////////////////////
2115 !
2116 Read_allsparms:OFF KEY
2117                  GOSUB Read_s11res
2118                  GOSUB Read_s21res
2119                  GOSUB Read_s12res
2120                  GOSUB Read_s22res
2121                  GOSUB Save_new_sparms
2122                  Reset_keys=1
2123                  RETURN
2124 !
2125 ! ////////////////////////////////////////////////////////////
2126 !
2127 Zoomread_s11res:OFF KEY
2128             CALL Zoom_on_peak
2129 Read_s11res:OFF KEY
2130             Sparm$="S11"
2131             Sss=3
2132             GOSUB Measure_sparms
2133             CALL Optimizedisplay
2134             GOSUB Set_startstop
2135             CALL Fit_sparms
2136             GOSUB Give_results
2137             Reset_keys=1
2138             RETURN
2139 !
2140 ! ////////////////////////////////////////////////////////////
2141 !
2142 Zoomread_s21res:OFF KEY
2143             CALL Zoom_on_peak
2144 Read_s21res:OFF KEY
2145             Sparm$="S21"
2146             Sss=1
2147             GOSUB Measure_sparms
2148             CALL Optimizedisplay
2149 !           GOSUB Set_startstop
2150 !           CALL Set_nwa("SPAN=100MHZ","SWEEP=RAMP","AVER=OFF","SWEEP=SING")
2151 !           CALL Set_nwa("SWEEP=CONT","AUTOSCALE=","LOCAL=")
2152 !           CALL Fit_sparms
2153 !           GOSUB Give_results
2154             Reset_keys=1
2155             RETURN
2156 !
2157 ! ////////////////////////////////////////////////////////////
2158 !
2159 Zoomread_s12res:OFF KEY
2160             CALL Zoom_on_peak
2161 Read_s12res:OFF KEY
2162             Sparm$="S12"
2163             Sss=1
```

```
2164            GOSUB Measure_sparms
2165            CALL Optimizedisplay
2166 !          CALL Fit_sparms
2167 !          GOSUB Set_startstop
2168 !          GOSUB Give_results
2169            Reset_keys=1
2170            RETURN
2171 !
2172 ! ////////////////////////////////////////////////////////////
2173 !
2174 Zoomread_s22res:OFF KEY
2175            CALL Zoom_on_peak
2176 Read_s22res:OFF KEY
2177            Sparm$="S22"
2178            Sss=3
2179            GOSUB Measure_sparms
2180            CALL Optimizedisplay
2181 !          CALL Fit_sparms
2182 !          GOSUB Give_results
2183            Reset_keys=1
2184            RETURN
2185 !
2186 ! ////////////////////////////////////////////////////////////
2187 !
2188 Measure_sparms:!
2189            SELECT Sparm$
2190            CASE "S11"
2191                Set_nwa("S=11","LOG MAG=")
2192            CASE "S21"
2193                Set_nwa("S=21","LOG MAG=")
2194            CASE "S12"
2195                Set_nwa("S=12","LOG MAG=")
2196            CASE "S22"
2197                Set_nwa("S=22","LOG MAG=")
2198            END SELECT
2199 !          CALL Set_nwa("SWEEP=STEP")
2200            CALL Set_nwa("NUMBER OF GROUPS=1")
2201            CALL Read_nwa("S=")
2202            RETURN
2203 !
2204 ! ////////////////////////////////////////////////////////////
2205 !
2206 Set_startstop:!
2207            Start_index=1
2208            Stop_index=Dcount
2209            RETURN
2210 !
2211 ! ////////////////////////////////////////////////////////////
2212 !
2213 !
2214 Give_results:!
2215            SELECT Sss
2216            CASE 1       !S21 RESONANCE
2217                IF Printer_on THEN
2218                    PRINTER IS Print_addr
2219                    PRINT USING "K,D.5DESZ,K,D.9DE";"Resonance 3dB Span: ",A(
  => 4)/A(3),"   Unloaded resonant frequency: ",A(4)+A(4)/(A(3)*2)
2220                    PRINT
2221                    PRINTER IS 1
2222                END IF
2223            END SELECT
2224            RETURN
2225 ! ////////////////////////////////////////////////////////////
2226 !
2227 Send_to_local:!
2228            Set_nwa("SWEEP=CONT","AVER=OFF","SWEEP=RAMP","LOCAL=")
2229            RETURN
2230 !
2231 ! ////////////////////////////////////////////////////////////
2232 !
2233 Init_com:  !
2234 !
2235            OPTION BASE 1
2236            COM /S_array/ INTEGER Dcount,Svalid,REAL S(801,5,2)
2237            COM /Fittype/ INTEGER Sselect,Start_index,Stop_index
2238! COM /Sparm/ Frequency(801),COMPLEX Sparm(801),Sparm$[3],INTEGER Num_points
2239            COM /Cdata/ INTEGER Mfit(2),Lista(5,2),Ma(2),REAL A(5),Ua(5),INTE
  => GER Ndata(2)
2240            COM /Cstats/ Alamda,Chisq,INTEGER Nca(2),REAL Alpha(5,5),Covar(5,
  => 5)
2241            COM /Files/ Diskdrive$[20],Filename$[10]
2242            COM /Bugs/ INTEGER Bug1,Bug2,Bug3,Printer
2243            COM /Nwa_data/ Start_freq,Stop_freq,Marker(5,3)
2244            COM /Peak_data/ Peak(100,3),Upeak(100,3),INTEGER Peak_index,Num_p
  => eaks
2245            COM /Resonance/ Kx,Ky,Q,F0,Ukx,Uky,Uq,Uf0
2246 !
2247            COM /History/ Status$[1],Time_orgn$[8],Date_orgn$[11]
2248            COM /History/ Time_chng$[8],Date_chng$[11],Description$[160]
2249            COM /Labels/ Labels$(30)[60],INTEGER Lbl_count,REAL Lbl_addr(30,6
  => )
2250            COM /Data_param/ INTEGER Data_count,Filesize,Curvecount,Roster(17
  => ,4)
2251            COM /Data_param/ REAL Sym_size,Symbol$(17)[2],Curve_id$(17)[40]
2252            COM /Data_param/ REAL Xmin_data,Xmax_data
2253            COM /Data_param/ REAL Ymin_data,Ymax_data
2254            COM /Ana_data/ INTEGER Prog_id,Sweep_type,Sweep_mode,Datacount
2255            COM /Ana_data/ INTEGER S11_valid,S21_valid,S12_valid,S22_valid,Po
  => rts
2256            COM /Ana_data/ REAL Z0,Start,Stop
2257            COM /Ana_data/ Title$[55],Company$[30],Operator_name$[30],Measure
  => _time$[30]
2258            COM /Sparms/ REAL Freq(801),COMPLEX S11(801),S21(801),S12(801),S2
  => 2(801)
2259            COM /Graphics/ INTEGER Speed
2260            COM /Background/ Graphtype$[12],Margins$(2)[10],Papersize$[1]
2261            COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
2262            COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
2263            COM /Background/ Xgrid_tick$[4],INTEGER Xmajor,Xminor
2264            COM /Background/ Ygrid_tick$[4],INTEGER Ymajor,Yminor
2265            COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
2266            COM /Axes_labels/ Print_xlabels$[3],Print_ylabels$[3]
2267            COM /Axes_labels/ Sig_digits$[2],REAL Xlcsize,Ylcsize
2268            COM /Windowspace/ REAL Xmin,Xmid,Xmax,Ymin,Ymid,Ymax! graph edges
  =>  UDU
2269            COM /Windowspace/ REAL Xleft,Xright,Ybottom,Ytop! paper edges UDU
  => s
2270            COM /Log_scale/ REAL Xcycles,Xbegin,Ycycles,Ybegin
2271            COM /Plot_device/ Plot_lang$[10],INTEGER Plot_addr
2272            COM /Hard_space/ REAL Xview_lft,Xview_rt,Yview_btm,Yview_top
2273            COM /Hard_space/ REAL Viewscale,Aspect_ratio
2274            COM /Frame_onoff/ INTEGER Frame_flag
2275            COM /Clear_space/ REAL Space_lft,Space_rt,Space_btm,Space_top
2276   ! MODEHELPER COM
2277            COM /Frequency_info/ REAL Start_frequency,Stop_frequency,Incremen
  => t_freq
2278            COM /Length_info/ REAL Start_length,Stop_length,Increment_len
2279            COM /Cavity_info/ REAL Diameter,Length
2280            COM /Lab_info/ REAL Rel_humidity,Temperature,Pressure,C
2281            COM /Sample_info/ Sample_desc$[40],REAL Thickness,Epsilon
2282            COM /Bessel_root/ REAL Bessel_root
2283            COM /Modes/ Mode$[10]
```

```
2284           COM /Modes/ REAL Te01(50,2),Te02(50,2)
2285           COM /Modes/ REAL Te12(50,2),Te13(50,2)
2286           COM /Output/ INTEGER Print_addr,Plotter,Printer_on,Plotter_on
2287           INTEGER Num_settings,Micr,Nummodes,Ssave
2288 !     !
2289 Init_var:!
2290           Speed=1
2291           Printer=701
2292           Print_addr=Printer
2293           Plotter=705
2294           Printer_on=1
2295           Plotter_on=0
2296           COMPLEX Cnum,P
2297           Svalid=0
2298           Ssave=0
2299           RETURN
2300 !
2301 Data:!
2302 Resonance_data:!
2303           DISP "PLEASE WAIT WHILE ROUTINES ARE LOADED....."
2304           LOADSUB Resonance_data FROM "MODESUB:,1400,0,3"
2305           LOADSUB Empty_cavity FROM "MODESUB:,1400,0,3"
2306           LOADSUB Loaded_cavity FROM "MODESUB:,1400,0,3"
2307           LOADSUB Var_freq_empty FROM "MODESUB:,1400,0,3"
2308           LOADSUB Var_freq_loaded FROM "MODESUB:,1400,0,3"
2309           LOADSUB Var_len_empty FROM "MODESUB:,1400,0,3"
2310           LOADSUB Var_len_loaded FROM "MODESUB:,1400,0,3"
2311           LOADSUB Calc_c FROM "MODESUB:,1400,0,3"
2312           LOADSUB Calc_mode_freq FROM "MODESUB:,1400,0,3"
2313           DISP
2314           CALL Resonance_data
2315           I=1
2316           J=0
2317           WHILE Te01(I,1)=0
2318               I=I+1
2319           END WHILE
2320           WHILE Te01(I,1)<>0
2321               J=J+1
2322               Freq(J)=Te01(I,1)
2323               I=I+1
2324           END WHILE
2325           Num_peaks=J
2326           DISP "PLEASE WAIT WHILE ROUTINES ARE DELETED....."
2327           DELSUB Resonance_data
2328           DELSUB Empty_cavity
2329           DELSUB Loaded_cavity
2330           DELSUB Var_freq_empty
2331           DELSUB Var_freq_loaded
2332           DELSUB Var_len_empty
2333           DELSUB Var_len_loaded
2334           DELSUB Calc_c
2335           DELSUB Calc_mode_freq
2336           DISP
2337           RETURN
2338 !
2339 ! ////////////////////////////////////////////////////////////////
2340 !
2341 End_program:  !
2342           DISP "The program has ended."
2343 !
2344 Mode_numbers:DATA 29,0,16,2,16,4,16,6,16,8,17,10,16,12,15,14,16,16,17,18,14
  => ,20,14,22,14,24,15,23,15,21,15,19,14,17,16,15,15,13,16,11,16
2345           DATA 9,15,7,15,5,16,3,18,1,17,2,16,4,16,6,16,8,16
2346           END
2347 !
2348 ! ****************************************************************
```

```
2349 !
2350   SUB Set_nwa(Cmd1$,OPTIONAL Cmd2$,Cmd3$,Cmd4$,Cmd5$,Cmd6$)
2351 Set_nwa:  !
2352        OPTION BASE 1
2353        ASSIGN @Nwa TO 716
2354        ALLOCATE C$(1:6)[35],Hp_ib$[35]
2355        ON NPAR GOTO C1,C2,C3,C4,C5,C6
2356 C6:    C$(6)=Cmd6$
2357 C5:    C$(5)=Cmd5$
2358 C4:    C$(4)=Cmd4$
2359 C3:    C$(3)=Cmd3$
2360 C2:    C$(2)=Cmd2$
2361 C1:    C$(1)=Cmd1$
2362!
2363        FOR I=1 TO NPAR
2364            Function$=C$(I)[1,POS(C$(I),"=")-1]
2365            Value$=C$(I)[POS(C$(I),"=")+1]
2366!
2367            SELECT Function$
2368            CASE "PRES","PRESET"
2369                Hp_ib$="PRES;"
2370            CASE "START","START FREQ"
2371                Hp_ib$="STAR"&Value$&";"
2372            CASE "STOP","STOP FREQ"
2373                Hp_ib$="STOP"&Value$&";"
2374            CASE "CENTER","CF"
2375                SELECT Value$
2376                CASE "MARK","MARKER"
2377                    Hp_ib$="CENT;EQUA;"
2378                CASE ELSE
2379                    Hp_ib$="CENT"&Value$&";"
2380                END SELECT
2381            CASE "SPAN","SP"
2382                SELECT Value$
2383                CASE "MARK","MARKER"
2384                    Hp_ib$="SPAN;EQUA;"
2385                CASE ELSE
2386                    Hp_ib$="SPAN"&Value$&";"
2387                END SELECT
2388 Marker:!
2389            CASE "MARKER","MARK"
2390                SELECT Value$
2391                CASE "MAX","MAXIMUM","MAXI"
2392                    Hp_ib$="MARKMAXI;"
2393                CASE "MIN","MINIMUM","MINI"
2394                    Hp_ib$="MARKMINI;"
2395                CASE "TARG","TARGET"
2396                    Hp_ib$="MARKTARG;"
2397                CASE "CONT","CONTINUOUS"
2398                    Hp_ib$="MARKCONT;"
2399                CASE "DISC","DISCRETE"
2400                    Hp_ib$="MARKDISC;"
2401                CASE "SEARCH RIGHT"
2402                    Hp_ib$="SEAR;"
2403                CASE "SEARCH LEFT"
2404                    Hp_ib$="SEAL;"
2405                CASE ELSE
2406                    Hp_ib$="MARK"&Value$&";"
2407                END SELECT
2408            CASE "MARKER TARGET","TARGET VALUE","TARV"
2409                Hp_ib$="TARV"&Value$&";"
2410            CASE "DELTA REF","DELTA REFERENCE","DEL REF","DELR"
2411                Hp_ib$="DELR"&Value$&";"
2412            CASE "DEL OFF","DELTA OFF","DELO"
2413                Hp_ib$="DELO;"
2414            CASE "AVER","AVERAGING","AVERAGE","AVG"
```

```
2415              IF Value$="OFF" THEN
2416                  Hp_ib$="AVEROFF;"
2417              ELSE
2418                  Hp_ib$="AVERON"&Value$&";"
2419              END IF
2420          CASE "NUMG","NUM GROUPS","NUMBER OF GROUPS"
2421              Hp_ib$="NUMG"&Value$&";"
2422          CASE "REF POS","REF POSN","REFERENCE POSITION","REFP"
2423              Hp_ib$="REFP"&Value$&";"
2424          CASE "REF VAL","REFERENCE VALUE","REF VALUE","REFV"
2425              Hp_ib$="REFV"&Value$&";"
2426          CASE "WAIT"
2427              Hp_ib$="WAIT;"
2428          CASE "UP","STEP UP"
2429              Hp_ib$="UP;"
2430          CASE "DOWN","STEP DOWN"
2431              Hp_ib$="DOWN;"
2432          CASE "LOGM","LOG MAG"
2433              Hp_ib$="LOGM;"
2434              IF Value$<>"" THEN Hp_ib$=Hp_ib$&" SCAL"&Value$&";"
2435          CASE "IMAG","CARTESIAN","X-jY"
2436              Hp_ib$="IMAG;"
2437          CASE "PHASE"
2438              Hp_ib$="PHAS;"
2439          CASE "SMITH CHART","SMITH"
2440              Hp_ib$="SMIC;"
2441          CASE "SWEEP"
2442              SELECT Value$
2443              CASE "CONT","CONTINUOUS"
2444                  Hp_ib$="CONT;"
2445              CASE "SING","SINGLE"
2446                  Hp_ib$="SING;"
2447              CASE "STEP"
2448                  Hp_ib$="STEP;"
2449              CASE "RAMP"
2450                  Hp_ib$="RAMP;"
2451              CASE "HOLD"
2452                  Hp_ib$="HOLD;"
2453              END SELECT
2454          CASE "S","s"
2455              Hp_ib$=Function$&Value$&";"
2456          CASE "SCALE","SCAL"
2457              Hp_ib$="SCAL"&Value$&";"
2458          CASE "AUTOSCALE","AUTO"
2459              Hp_ib$="AUTO;"
2460          CASE "LOCAL"
2461              Hp_ib$=""
2462              LOCAL @Nwa
2463          CASE ELSE
2464              PRINT Function$;" IS NOT DEFINED IN SET_NWA"
2465          END SELECT
2466          IF Hp_ib$<>"" THEN OUTPUT @Nwa;Hp_ib$
2467      NEXT I
2468      ASSIGN @Nwa TO *
2469  SUBEND
2470 !
2471 ! ******************************************************************
2472 !
2473  SUB Read_nwa(Cmd1$,OPTIONAL Cmd2$,Cmd3$,Cmd4$,Cmd5$,Cmd6$)
2474      OPTION BASE 1
2475      COM /Nwa_data/ Start_freq,Stop_freq,Marker(*)
2476      COM /S_array/ INTEGER Dcount,Svalid,REAL S(801,5,2)
2477      DIM Sparm$[5],Swem$[20]
2478 Read_nwa:  !
2479      ASSIGN @Nwa TO 716
2480      ASSIGN @Nwa_data1 TO 716;FORMAT ON
2481      ASSIGN @Nwa_data2 TO 716;FORMAT OFF
2482      REAL Freq
2483      INTEGER Preamble,Bytes
2484      ALLOCATE C$(1:6)[35],Hp_ib$[35]
2485      ON NPAR GOTO C1,C2,C3,C4,C5,C6
2486 C6:  C$(6)=Cmd6$
2487 C5:  C$(5)=Cmd5$
2488 C4:  C$(4)=Cmd4$
2489 C3:  C$(3)=Cmd3$
2490 C2:  C$(2)=Cmd2$
2491 C1:  C$(1)=Cmd1$
2492!
2493      FOR I=1 TO NPAR
2494          Function$=C$(I)[1,POS(C$(I),"=")-1]
2495          Value$=C$(I)[POS(C$(I),"=")+1]
2496!
2497          SELECT Function$
2498          CASE "MARKER","MARK"
2499              Hp_ib$="MARK"&Value$&";"
2500              OUTPUT @Nwa;Hp_ib$
2501              OUTPUT @Nwa;"OUTPACTI;"
2502              ENTER @Nwa_data1;Marker(VAL(Value$),1)
2503              OUTPUT @Nwa;"OUTPMARK;"
2504              ENTER @Nwa_data1;Marker(VAL(Value$),2),Marker(VAL(Value$),3)
2505          CASE "START"
2506              OUTPUT @Nwa;"STAR; OUTPACTI;"
2507              ENTER @Nwa_data1;Start_freq
2508          CASE "STOP"
2509              OUTPUT @Nwa;"STOP; OUTPACTI;"
2510              ENTER @Nwa_data1;Stop_freq
2511          CASE "S","s"
2512              GOSUB Get_sparm
2513          CASE ELSE
2514              PRINT Function$;" NOT DEFINED IN READ ANA"
2515          END SELECT
2516      NEXT I
2517      ASSIGN @Nwa TO *
2518      ASSIGN @Nwa_data1 TO *
2519      ASSIGN @Nwa_data2 TO *
2520      SUBEXIT
2521 !
2522 ! //////////////////////////////////////////////////////////////
2523 !
2524 Get_sparm:!This subroutine gets the S-parameter data that is on the NWA CRT
2525      OUTPUT @Nwa;"PARA?"            !Find out what Sparm is on the CRT
2526      ENTER @Nwa;Sparm$
2527      Sparm$=Sparm$[2,4]
2528      OUTPUT @Nwa;"SWEM?"           !What is the sweep mode?
2529      ENTER @Nwa;Swem$              !RAMP, STEP or FREQUENCY LIST
2530      Swem$=Swem$[2,LEN(Swem$)-1]
2531      SELECT Sparm$
2532      CASE "S11"
2533          N=1                   !Set index for the S(*) array
2534      CASE "S21"
2535          N=2
2536      CASE "S12"
2537          N=3
2538      CASE "S22"
2539          N=4
2540      CASE ELSE
2541          BEEP
2542          DISP "Sparm$ is invalid in Read_nwa"
2543          PAUSE
2544      END SELECT
2545 !
2546      OUTPUT @Nwa;"FORM3; OUTPDATA;"
```

```
2547        ENTER @Nwa_data2;Preamble;Bytes
2548        Dcount=Bytes/16
2549        ALLOCATE S1(1:Dcount,2),F(1:Dcount)
2550        ENTER @Nwa_data2;S1(*)
2551        SELECT Swem$
2552        CASE "RAMP","STEP"
2553            CALL Read_nwa("START=","STOP=")
2554            FOR J=1 TO Dcount
2555                F(J)=Start_freq+(J-1)*(Stop_freq-Start_freq)/(Dcount-1)
2556            NEXT J
2557        CASE "FREQUENCY LIST"
2558            OUTPUT @Nwa;"FORM3;OUTPFREL;"
2559            ENTER @Nwa;Preamble,Bytes,F(*)
2560        CASE ELSE
2561            BEEP
2562            DISP "Swem$ has not been properly selected in Read_nwa"
2563            PAUSE
2564        END SELECT
2565        MAT S(1:Dcount,N,1)= S1(1:Dcount,1)
2566        MAT S(1:Dcount,N,2)= S1(1:Dcount,2)
2567        MAT S(1:Dcount,5,1)= F
2568        DEALLOCATE F(*),S1(*)
2569        IF NOT BIT(Svalid,N-1) THEN Svalid=Svalid+2^(N-1)
2570        RETURN
2571    SUBEND
2572 !
2573 ! ****************************************************************
2574 !
2575    SUB Zoom_on_peak
2576 !This subprogram assumes there is a resonance peak on the network analyzer
2577 !display.  The routine will zoom in on the center frequency by reducing the
2578 !span until the resonance "Delta" dB points are shown.  Averaging depends
2579 !on signal level.  Step frequency mode is used only for the last sweep.
2580 !The S(I,J,K) array is dimensioned (801,5,2) (1:Dcount,Ns,REAL/IMAG)
2581 !Ns= 1:S11 2:S21 3:S12 4:S22 5:Frequency(for K=1)
2582 !Written by Eric J. Vanzura NIST 723.02
2583        OPTION BASE 1
2584        ASSIGN @Nwa TO 716
2585        COM /Nwa_data/ Start_freq,Stop_freq,Marker(*)
2586 !      COM /Sparm/ Frequency(801),COMPLEX Sparm(801),Sparm$[3],INTEGER Num_p
  => oints
2587        COM /S_array/ INTEGER Dcount,Svalid,REAL S(801,5,2)
2588 Zoom_on_peak:      !
2589        Lastscale=.5
2590        Delta=-3
2591        Spanratio=1.5
2592 !
2593        Scale=20*Lastscale
2594        Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
2595        DISP "Zooming in on peak"
2596 !  Resonance data already exists on NWA CRT.
2597        Averages=4
2598        Nwa_span=5.E+6
2599        GOSUB Take_sweeps
2600        CALL Optimizedisplay
2601        GOSUB Find_deltaspan
2602        GOSUB Print_peak_info
2603        Scale=Lastscale*10
2604        GOSUB Set_new_nwaspan
2605        GOSUB Select_averages
2606        Set_nwa("REF VAL="&VAL$(Level+Scale),"SCALE="&VAL$(Scale))
2607        GOSUB Take_sweeps
2608        CALL Optimizedisplay
2609        GOSUB Find_deltaspan
2610        GOSUB Print_peak_info
2611        Scale=Lastscale
2612        GOSUB Set_new_nwaspan
2613        GOSUB Select_averages
2614        Set_nwa("REF VAL="&VAL$(Level+Scale),"SCALE="&VAL$(Scale))
2615        GOSUB Take_sweeps
2616        CALL Optimizedisplay
2617        GOSUB Find_deltaspan
2618        GOSUB Print_peak_info
2619        SUBEXIT
2620 !
2621 ! ////////////////////////////////////////////////////////////////
2622 !
2623 Set_new_nwaspan:!
2624        SELECT Scale
2625        CASE >Lastscale
2626            IF Span<5000 THEN
2627                Span=3.E+6
2628                Nwa_span=Span
2629            ELSE
2630                Nwa_span=Span*3
2631            END IF
2632        CASE =Lastscale
2633            Nwa_span=Spanratio*Span
2634        END SELECT
2635        RETURN
2636 !
2637 ! //////////////////////////////////////////////////////////////
2638 !
2639 Find_deltaspan:!
2640        Set_nwa("MARK=CONT","MARK=1","MARK=MAX")
2641        Read_nwa("MARK=1")
2642        Freq=Marker(1,1)
2643        Level=Marker(1,2)
2644        Set_nwa("DEL REF=1","MARKER TARGET="&VAL$(Delta))
2645        Set_nwa("MARK=2","MARK=MAX","MARKER=TARGET","MARKER=SEARCH LEFT")
2646        Set_nwa("MARK=3","MARK=MAX","MARKER=TARGET")
2647        Read_nwa("MARK=2","MARK=3")
2648        Set_nwa("DELTA OFF=")
2649        IF ABS(Marker(2,1))>1000 AND ABS(Marker(3,1))>1000 THEN
2650            IF ABS(Marker(2,1))<ABS(.9*Marker(3,1)) THEN
2651 !              Span=2*ABS(Marker(3,1))
2652                Span=ABS(Marker(2,1))+ABS(Marker(3,1))
2653            ELSE
2654 !              Span=2*ABS(Marker(2,1))
2655                Span=ABS(Marker(2,1))+ABS(Marker(3,1))
2656            END IF
2657        ELSE
2658            Span=999999
2659            BEEP
2660            PRINT "Span cannot be determined from marker information";
2661            PRINT " in Find_deltaspan."
2662            PAUSE
2663        END IF
2664        RETURN
2665 !
2666 ! //////////////////////////////////////////////////////////////
2667 !
2668 Print_peak_info:!
2669        IF Freq<1.E+10 THEN
2670            PRINT USING "K,X,D.9DE,2X,K,S2D.2D,2X,K,8D,K,X,D.9DE";"Freq=",Fre
  => q/1.E+9,"Level=",Level,"Span=",Span,"Freq+(Span/2)=",(Freq+Span/2)/1.E+9
2671        ELSE
2672            PRINT USING "K,X,DD.9DE,2X,K,S2D.2D,2X,K,8D,K,X,DD.9DE";"Freq=",F
  => req/1.E+9,"Level=",Level,"Span=",Span,"Freq+(Span/2)=",(Freq+Span/2)/1.E+9
2673        END IF
2674        RETURN
2675 !
```

```
2676 ! ////////////////////////////////////////////////////////////
2677 !
2678 Take_sweeps:!
2679       Set_nwa("MARK=1","MARK=MAX","CENTER=MARKER","SPAN="&VAL$(Nwa_span))
2680       Set_nwa("AVER="&VAL$(Averages))
2681       IF Scale=Lastscale THEN
2682           Set_nwa("SWEEP=STEP","NUMG=1")
2683       ELSE
2684           Set_nwa("SWEEP=RAMP","NUMG="&VAL$(Averages+1))
2685       END IF
2686       RETURN
2687 !
2688 ! ////////////////////////////////////////////////////////////
2689 !
2690 Select_averages:!         !
2691       SELECT Level        !Level of marker 1
2692       CASE >-30
2693           Averages=2
2694       CASE >-42
2695           Averages=4
2696       CASE >-50
2697           Averages=8
2698       CASE >-60
2699           Averages=16
2700       CASE <=-60
2701           Averages=32
2702       CASE ELSE
2703           PRINT "Select_averages IS BAD"
2704       END SELECT
2705       IF Scale=Lastscale THEN Averages=Averages*4
2706       PRINT "LEVEL=";Level,"AVERAGES=";Averages
2707       RETURN
2708  SUBEND
2709  !
2710  ! ************************************************************
2711  !
2712  SUB Sweep_nwa
2713 Sweep_nwa:  !
2714       GOSUB Init_sweep_nwa
2715       GOSUB Print_info
2716       CALL Set_nwa("PRES=","WAIT=","S=21","REF POS=10")
2717       Startwind=Startfreq
2718       WHILE Startwind<Stopfreq
2719           DISP "Searching for a peak ";Startwind/1.E+9;" to ";(Startwind+St
  => epfreq)/1.E+9;" GHz"
2720           Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
2721           ON KEY 1 LABEL "ZOOM ONPEAK",Prty GOTO Zoomonpeak
2722           ON KEY 2 LABEL "SKIP   WINDOW",Prty+1 RECOVER Next_step
2723           CALL Set_nwa("SWEEP=CONT","SWEEP=RAMP")
2724           CALL Set_nwa("START FREQ="&VAL$(Startwind),"STOP FREQ="&VAL$(Star
  => twind+Stepfreq))
2725           CALL Set_nwa("S=21","REFV=0","SCALE=10","AVER=16","NUMG=17")
2726           CALL Set_nwa("MARK=1","MARK=MAX","MARK=2","MARK=MIN")
2727           CALL Read_nwa("MARK=1","MARK=2")
2728           IF Marker(1,2)<Min_peak_val THEN GOTO Next_step
2729           IF Marker(1,2)-Marker(2,2)<Min_sig_to_noi THEN GOTO Next_step
2730 Zoomonpeak:OFF KEY 1
2731           CALL Zoom_on_peak  !Sets up S21 for data aquisition
2732           CALL Read_nwa("S=")
2733           Num_peaks=Num_peaks+1
2734           Sselect=1       ! 0:S11 1:S21 2:S12 3:S22
2735           IF Save_fo_later THEN
2736               GOSUB Save_fo_later
2737           ELSE
2738               Start_index=1
2739               Stop_index=Dcount
2740               CALL Fit_sparms
2741               GOSUB Print_3dbspan
2742               PRINT
2743               IF Printer_on THEN
2744                   PRINTER IS Print_addr
2745                   GOSUB Print_3dbspan
2746                   PRINT
2747                   PRINTER IS 1
2748               END IF
2749               GOSUB Add_peak_2_list
2750           END IF
2751 Next_step:OFF KEY
2752           Startwind=Startwind+Stepfreq
2753       END WHILE
2754       GOTO End_sweep_nwa
2755 !
2756 ! ////////////////////////////////////////////////////////////
2757 !
2758 Print_3dbspan:!
2759       SELECT Sss
2760       CASE 1
2761           PRINT USING "K,D.5DESZ,K,D.9DE";"3dB span:",A(4)/A(3)," F0+(Span/
  => 2):",A(4)+A(4)/(A(3)*2)/1.E+9
2762       CASE ELSE
2763           DISP "You need to add a PRINT statement in Sweep_nwa"
2764       END SELECT
2765       RETURN
2766 !
2767 ! ////////////////////////////////////////////////////////////
2768 !
2769 Save_fo_later:!
2770       SELECT TRIM$(SYSTEM$("SYSTEM ID"))
2771       CASE "S300:20"     ! The Fast HPIB
2772           Diskdrive$=":,1400,1"
2773       CASE "9836C"       ! The Wiper card
2774           Diskdrive$=":,1500,2"
2775       CASE ELSE
2776           BEEP
2777           DISP "Save_fo_later ain't set up fo yo computer!"
2778           PAUSE
2779       END SELECT
2780       A$=DATE$(TIMEDATE)
2781       Filename$=TRIM$(A$[1,2])&A$[4,6]&A$[10,11]&VAL$(Num_peaks)
2782       Ssave=2^Sselect
2783 !     CALL New_save_sparms(Ssave)
2784       WAIT 5
2786       PRINT Filename$&Diskdrive$;" has been saved"
2787       DISP Filename$&Diskdrive$;" has been saved"
2788       RETURN
2789!
2790! ////////////////////////////////////////////////////////////
2791!
2792 Init_sweep_nwa:OPTION BASE 1
2793       INTEGER Done,Reset_keys,Bad_number,Ssave
2794       REAL Startfreq,Stopfreq,Stepfreq,Minfreq,Maxfreq
2795       REAL Min_peak_val,Min_sig_to_noi
2796       REAL Freq,Level,Span
2797       DIM Id$[40]
2798 Init_com:DEG
2799       COM /History/ Status$[1],Time_orgn$[8],Date_orgn$[11]
2800       COM /History/ Time_chng$[8],Date_chng$[11],Description$[160]
2801       COM /Labels/ Labels$(30)[60],INTEGER Lbl_count,REAL Lbl_addr(30,6)
2802       COM /Data_param/ INTEGER Data_count,Filesize,Curvecount,Roster(17,4)
2803       COM /Data_param/ REAL Sym_size,Symbol$(17)[2],Curve_id$(17)[40]
2804       COM /Data_param/ REAL Xmin_data,Xmax_data
2805       COM /Data_param/ REAL Ymin_data,Ymax_data
```

```
2806       COM /Background/ Graphtype$[12],Margins$(2)[10],Papersize$[1]
2807       COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
2808       COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
2809       COM /Background/ Xgrid_tick$[4],INTEGER Xmajor,Xminor
2810       COM /Background/ Ygrid_tick$[4],INTEGER Ymajor,Yminor
2811       COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
2812       COM /Axes_labels/ Print_xlabels$[3],Print_ylabels$[3]
2813       COM /Axes_labels/ Sig_digits$[2],REAL Xlcsize,Ylcsize
2814       COM /Windowspace/ REAL Xmin,Xmid,Xmax,Ymin,Ymid,Ymax! graph edges UDU
2815       COM /Windowspace/ REAL Xleft,Xright,Ybottom,Ytop! paper edges UDUs
2816       COM /Log_scale/ REAL Xcycles,Xbegin,Ycycles,Ybegin
2817       COM /Plot_device/ Plot_lang$[10],INTEGER Plot_addr
2818       COM /Hard_space/ REAL Xview_lft,Xview_rt,Yview_btm,Yview_top
2819       COM /Hard_space/ REAL Viewscale,Aspect_ratio
2820       COM /Frame_onoff/ INTEGER Frame_flag
2821       COM /Clear_space/ REAL Space_lft,Space_rt,Space_btm,Space_top
2822 !     !
2823       COM /S_array/ INTEGER Dcount,Svalid,REAL S(801,5,2)
2824       COM /Fittype/ INTEGER Sselect,Start_index,Stop_index
2825       COM /Cdata/ INTEGER Mfit(2),Lista(5,2),Ma(2),REAL A(5),Ua(5),INTEGER
  => Ndata(2)
2826       COM /Cstats/ Alamda,Chisq,INTEGER Nca(2),REAL Alpha(5,5),Covar(5,5)
2827       COM /Files/ Diskdrive$[20],Filename$[10]
2828       COM /Bugs/ INTEGER Bug1,Bug2,Bug3,Printer
2829       COM /Nwa_data/ Start_freq,Stop_freq,Marker(*)
2830       COM /Peak_data/ Peak(*),Upeak(*),INTEGER Peak_index,Num_peaks
2831 !     COM /Resonance/ Kx,Ky,Q,F0,Ukx,Uky,Uq,Uf0
2832       COM /Output/ INTEGER Print_addr,Plotter,Printer_on,Plotter_on
2833       ASSIGN @Nwa TO 716
2834       ASSIGN @Nwa_data TO 716;FORMAT ON
2835 Init_vals:!
2836       Minfreq=6.55E+9
2837       Maxfreq=1.35E+10
2838       Startfreq=8.2E+9
2839       Stopfreq=1.24E+10
2840       Stepfreq=8.0E+7
2841       Min_peak_val=-55
2842       Min_sig_to_noi=10
2843       Num_peaks=0
2844       Peak_index=0
2845       Save_fo_later=1
2846 Init_sweep:OFF KEY
2847       GOSUB Sweep_menu
2848       Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
2849       ON KEY 0 LABEL "OK/DONE",Prty GOSUB Done_return
2850       ON KEY 1 LABEL "Startfreq",Prty GOSUB Startfreq
2851       ON KEY 2 LABEL "Stopfreq",Prty GOSUB Stopfreq
2852       ON KEY 3 LABEL "Stepfreq",Prty GOSUB Stepfreq
2853       ON KEY 6 LABEL "Min_peak_val",Prty GOSUB Min_peak_val
2854       ON KEY 7 LABEL "Min_sig_to_noi",Prty GOSUB Min_sig_to_noi
2855       IF Printer_on THEN
2856           ON KEY 8 LABEL "No printer",Prty GOSUB Tog_printer
2857       ELSE
2858           ON KEY 8 LABEL "Printer ON",Prty GOSUB Tog_printer
2859       END IF
2860       IF Save_fo_later THEN
2861           ON KEY 9 LABEL "Compute Sparms",Prty GOSUB Tog_savefolater
2862       ELSE
2863           ON KEY 9 LABEL "Save Sparms",Prty GOSUB Tog_savefolater
2864       END IF
2865       Done=0
2866       Reset_keys=0
2867       REPEAT
2868           IF Reset_keys THEN GOTO Init_sweep
2869       UNTIL Done
2870       OFF KEY
2871       RETURN
2872 !
2873 ! /////////////////////////////////////////////////////////''''''
2874 !
2875 Tog_printer:CFF KEY 8
2876       IF Printer_on THEN
2877           Printer_on=0
2878       ELSE
2879           Printer_on=1
2880       END IF
2881       Reset_keys=1
2882       RETURN
2883 !
2884 Done_return:Done=1
2885       RETURN
2886 !
2887 Startfreq:CALL Enter_real(Minfreq,Stopfreq,"Sweep start frequency",Startfre
  => q)
2888       Reset_keys=1
2889       RETURN
2890 !
2891 Stopfreq:CALL Enter_real(Startfreq,Maxfreq,"Sweep stop frequency",Stopfreq)
2892       Reset_keys=1
2893       RETURN
2894 !
2895 Stepfreq:CALL Enter_real(1.E+4,3.E+8,"Sweep step span (Hz)",Stepfreq)
2896       Reset_keys=1
2897       RETURN
2898 !
2899 Min_peak_val:CALL Enter_real(-120,20,"Minimum resonance peak value (dB)",Mi
  => n_peak_val)
2900       Reset_keys=1
2901       RETURN
2902 !
2903 Min_sig_to_noi:CALL Enter_real(3,50,"Minimum signal-to-noise ratio",Min_sig
  => _to_noi)
2904       Reset_keys=1
2905       RETURN
2906 !
2907 Tog_savefolater:Reset_keys=1
2908       IF Save_fo_later THEN
2909           Save_fo_later=0
2910       ELSE
2911           Save_fo_later=1
2912       END IF
2913       RETURN
2914 !
2915 Sweep_menu:PRINT TABXY(1,1);
2916       IF Startfreq>1.E+10 THEN
2917           PRINT USING "K,2D.6DE,K";"Startfreq=",Startfreq," Hz"
2918       ELSE
2919           PRINT USING "K,D.6DE,K";"Startfreq=",Startfreq," Hz"
2920       END IF
2921       IF Stopfreq>1.E+10 THEN
2922           PRINT USING "K,2D.6DE,K";"Stopfreq=",Stopfreq," Hz"
2923       ELSE
2924           PRINT USING "K,D.6DE,K";"Stopfreq=",Stopfreq," Hz"
2925       END IF
2926       PRINT USING "K,D.2DE,K";"Stepfreq=",Stepfreq," Hz"
2927       PRINT
2928       PRINT USING "K,M3D,K";"Min_peak_val=",Min_peak_val," dB"
2929       PRINT USING "K,M3D,K";"Min_sig_to_noi=",Min_sig_to_noi," dB"
2930       PRINT
2931       IF Printer_on THEN
2932           PRINT "Printer is ON                    Print_addr=";Print_addr;"   "
2933       ELSE
```

```
2934            PRINT "Printer is OFF              Print_addr=";Print_addr;"   "
2935        END IF
2936        PRINT
2937        IF Save_fo_later THEN
2938            PRINT "The S-parameters will be saved on disk. Resonance paramete
 => rs won't be computed."
2939        ELSE
2940            PRINT "The resonance parameters will be computed.
 =>                     "
2941        END IF
2942        RETURN
2943 !
2944 ! ////////////////////////////////////////////////////////////
2945 !
2946 Print_info:!
2947        ALLOCATE Desc$[160]
2948        REPEAT
2949            LINPUT "Enter a description of this sweep. When finished enter a
 => null string.",Desc$
2950            PRINT Desc$
2951            IF Printer_on THEN OUTPUT Print_addr;Desc$
2952        UNTIL Desc$=""
2953        DEALLOCATE Desc$
2954        RETURN
2955 !
2956 ! ////////////////////////////////////////////////////////////
2957 !
2958 Add_peak_2_list:!
2959        SELECT Sselect
2960        CASE 1,2
2961            Peak(Num_peaks,1)=A(3)
2962            Peak(Num_peaks,2)=SQRT(A(1)^2+A(2)^2)
2963            Peak(Num_peaks,3)=A(3)/A(4)
2964        CASE ELSE
2965            BEEP
2966            DISP "Sweep_nwa needs to have S11 measurements added"
2967            PAUSE
2968        END SELECT
2969        RETURN
2970 !
2971 ! ////////////////////////////////////////////////////////////
2972 !
2973 Save_sweepfreqs:OFF KEY
2974        IF Save_fo_later THEN RETURN
2975        Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
2976        ON KEY 0 LABEL "NO",Prty GOSUB Done_return
2977        ON KEY 2 LABEL "YES",Prty GOSUB Freqs_to_disk
2978        DISP "Do you want to save the resonant frequency list to disk?"
2979        Done=0
2980        LOOP
2981        EXIT IF Done
2982        END LOOP
2983        RETURN
2984 !
2985 ! ////////////////////////////////////////////////////////////
2986 !
2987 Freqs_to_disk:OFF KEY
2988        ALLOCATE Basket(Num_peaks,2)
2989        MAT Basket(1:Num_peaks,1)= Peak(1:Num_peaks,1)
2990        MAT Basket(1:Num_peaks,2)= Peak(1:Num_peaks,3)
2991        CALL Enter_id(Id$,"the resonant frequency list")
2992        IF Id$="" THEN GOTO Abort_savefreqs
2993        CALL Select_disk
2994        IF Diskdrive$="NO DISK" THEN GOTO Abort_savefreqs
2995        CALL Enterfilename("ABORT"," resonant frequency list")
2996        IF Filename$="" THEN GOTO Abort_savefreqs
2997        CALL Save_file(Basket(*),Num_peaks,Id$)
2998 Abort_savefreqs:!
2999        Done=1
3000        RETURN
3001 !
3002 ! ////////////////////////////////////////////////////////////
3003 !
3004 End_sweep_nwa:!
3005 !      CALL Set_nwa("PRES=")
3006        IF Save_fo_later THEN      !This array hasn't been used
3007            Peak(Num_peaks,1)=1.063E+10
3008            Peak(Num_peaks,3)=2.E+7
3009            Peak(Num_peaks,2)=-25
3010        END IF
3011        Peak_index=1
3012        CALL Set_nwa("CENTER="&VAL$(Peak(Peak_index,1)))
3013        CALL Set_nwa("SPAN="&VAL$(Peak(Peak_index,3)*5))
3014        CALL Set_nwa("REF VAL="&VAL$(Peak(Peak_index,2)+5),"SCALE=5")
3015        CALL Set_nwa("S=21","REF POS=10")
3016        LOCAL @Nwa
3017        IF Printer_on THEN OUTPUT Print_addr;CHR$(12);
3018        BEEP
3019!       GOSUB Save_sweepfreqs
3020        DISP "I am finished with the experiment oh great master."
3021   SUBEND
3022 !
3023 ! ************************************************************
3024 !
3025   SUB Pack_data(File(*),OPTIONAL REAL F1(*),INTEGER Data_cnt,Pen,Id$)
3026 Pack_data:  ! Original: 01 Jun 1987    G. Koepke
3027             ! Revision: 07 Aug 1987
3028        ! This routine will take up to 17 independent data files and pack
3029        ! the information into File(*) using GRAPH_DATA master file format.
3030        ! The Roster(*) information will be generated for use by Plot_all.
3031        !
3032        ! Roster(i,1) = Curve number 1,2,3,...,17.
3033        ! Roster(i,2) = Start address in File(x,*); = x
3034        ! Roster(i,3) = Datacount for curve i
3035        ! Roster(i,4) = PEN number for this curve
3036        !
3037        ! Symbol$(i)="" or "Y" => no symbol, connect pts
3038        ! Symbol$(i)="*Y" =>  * symbol, connect pts
3039        ! Symbol$(i)="*N" =>  * symbol, do not connect pts
3040        !
3041        OPTION BASE 1
3042        COM /Data_param/ INTEGER Datacount,Filesize,Curvecount,Roster(*)
3043        COM /Data_param/ REAL Sym_size,Symbol$(*),Curve_id$(*)
3044        COM /Data_param/ REAL Xmin_data,Xmax_data
3045        COM /Data_param/ REAL Ymin_data,Ymax_data
3046        COM /Bugs/ INTEGER Bug1,Bug2,Bug3,Printer
3047        INTEGER I,J,C
3048        Filesize=SIZE(File,1)
3049        IF NPAR=1 THEN   ! Clear data parameters and exit
3050            Datacount=0
3051            Curvecount=0
3052            MAT Roster= (0)
3053            MAT Symbol$= ("")
3054            MAT Curve_id$= ("")
3055            SUBEXIT
3056        END IF
3057        IF Data_cnt<1 THEN
3058            DISP " NO data in this file  (packing aborted...): ";Id$
3059            BEEP
3060            PAUSE
3061            SUBEXIT
3062        END IF
```

```
3063      !
3064      ! Add new file to the data being plotted.
3065      !
3066      Timer=TIMEDATE
3067      DISP " Packing ... ";Id$
3068      Pen=MIN(MAX(Pen,1),8)
3069      SELECT Curvecount
3070      CASE 0
3071          Curvecount=1
3072          I=Curvecount
3073          Roster(I,1)=1
3074          Roster(I,2)=1
3075      CASE <17
3076          Curvecount=Curvecount+1
3077          I=Curvecount
3078          Roster(I,1)=I
3079          Roster(I,2)=Roster(I-1,2)+Roster(I-1,3)
3080      CASE ELSE
3081          DISP " CURVE limit has been reached, new data discarded. ";
3082          DISP " (continue) "
3083          BEEP
3084          PAUSE
3085          DISP CHR$(12);
3086          SUBEXIT
3087      END SELECT
3088      Roster(I,3)=Data_cnt
3089      Roster(I,4)=Pen
3090      Symbol$(I)=""
3091      Curve_id$(I)=Id$
3092      !
3093      IF Roster(I,2)+Roster(I,3)-1>Filesize THEN
3094          DISP " DATA FILE overflow, new data discarded. ";
3095          DISP " (continue) "
3096          BEEP
3097          PAUSE
3098          Curvecount=Curvecount-1
3099          DISP CHR$(12)
3100          SUBEXIT
3101      END IF
3102      !
3103      ! Copy data into File(*)
3104      !
3105      C=1
3106      FOR J=Roster(I,2) TO Roster(I,2)+Roster(I,3)-1
3107          File(J,1)=F1(C,1)
3108          File(J,2)=F1(C,2)
3109          C=C+1
3110      NEXT J
3111      LOOP
3112      EXIT IF TIMEDATE-Timer>1
3113      END LOOP
3114      DISP CHR$(12)
3115      SUBEXIT
3116      !
3117  SUBEND
3118  !
3119  ! ***********************************************************
3120  !
3121  SUB Init_graphics(Label1$,Label2$,Label3$)
3122  !
3123 Init_graphics:  ! Original: 01 Jun 1987   G. Koepke
3124                 ! Revision: 07 Aug 1987
3125      ! Roster(i,1) = Curve number 1,2,3,...,17
3126      ! Roster(i,2) = Start address in File(x,*); = x
3127      ! Roster(i,3) = Datacount for curve i
3128      ! Roster(i,4) = PEN number for this curve
```

```
3129      !
3130      ! Symbol$(i)="" or "Y" => no symbol, connect pts
3131      ! Symbol$(i)="*Y" =>  * symbol, connect pts
3132      ! Symbol$(i)="*N" =>  * symbol, do not connect pts
3133      ! Lbl_addr: x, y, pen, size, LDIR, LORG
3134      !
3135      COM /Labels/ Labels$(*),INTEGER Lbl_count,REAL Lbl_addr(*)
3136      COM /Data_param/ INTEGER Datacount,Filesize,Curvecount,Roster(*)
3137      COM /Data_param/ REAL Sym_size,Symbol$(*),Curve_id$(*)
3138      COM /Data_param/ REAL Xmin_data,Xmax_data
3139      COM /Data_param/ REAL Ymin_data,Ymax_data
3140      COM /Background/ Graphtype$,Margins$(*),Papersize$
3141      COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
3142      COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
3143      COM /Background/ Xgrid_tick$,INTEGER Xmajor,Xminor
3144      COM /Background/ Ygrid_tick$,INTEGER Ymajor,Yminor
3145      COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
3146      !
3147      COM /Axes_labels/ Print_xlabels$,Print_ylabels$
3148      COM /Axes_labels/ Sig_digits$,REAL Xlcsize,Ylcsize
3149      !
3150      COM /Windowspace/ REAL Xmin,Xmid,Xmax,Ymin,Ymid,Ymax!graph edges UDU
3151      COM /Windowspace/ REAL Xleft,Xright,Ybottom,Ytop! paper edges UDUs
3152      COM /Plot_device/ Plot_lang$,INTEGER Plot_addr
3153      COM /Hard_space/ REAL Xview_lft,Xview_rt,Yview_btm,Yview_top
3154      COM /Hard_space/ REAL Viewscale,Aspect_ratio
3155      COM /Frame_onoff/ INTEGER Frame_flag
3156      COM /Clear_space/ REAL Space_lft,Space_rt,Space_btm,Space_top
3157      COM /Bugs/ INTEGER Bug1,Bug2,Bug3,Printer
3158      !
3159      ! ///////////////// INITIAL VALUES ///////////////////////////////
3160      !
3161 Ginit_clear:GINIT  ! Clear all graphics
3162      GCLEAR
3163      OUTPUT 2 USING "#,K";" K"   ! Clear the screen
3164 Initial_values:  .   !DEFINE ALL INITIAL VALUES
3165      !
3166      Frame_flag=1          ! Completely frame the data area.
3167      Print_xlabels$="YES"! AND print labels.
3168      Print_ylabels$="YES"
3169      Sig_digits$="FF"     ! Select free format for X & Y labels.
3170      Xlcsize=.035          ! Label size factor for axes labels.
3171      Ylcsize=.035          ! graph_data default size = .032
3172 !    Plot_lang$="INTERNAL"! OR "HPGL"
3173 !    Plot_addr=3           ! OR 705 (HPIB ADDR OF PLOTTER)
3174      Margins$(1)="HORIZONTAL"! OR "VERTICAL" . for plotter only
3175      Margins$(2)="FULL"   ! OR "BOUND LEFT"
3176                           ! OR "BOUND TOP"
3177                           ! OR "SQUARE"
3178                           ! OR "USER"
3179      Papersize$="4"       ! 8.5x11
3180                           ! OR "3", 11x17
3181      Backgnd_pen=1         ! 1 to 8
3182      Auto_time=1           ! Automatically label time/date on graphs.
3183      Auto_file=0           ! Automatically label last file name.
3184      Pen_speed=20.0        ! .38 cm/s TO 38.1 cm/s
3185      !
3186      ! SET VIEWPORT PARAMETERS
3187      !
3188      Xview_lft=34.4
3189      Xview_rt=100*RATIO
3190      Yview_btm=24
3191      Yview_top=100
3192      ! Clear space around labels in terms of frac of scale
3193      Space_lft=.21
3194      Space_rt=.02
```

```
3195        Space_btm=.11
3196        Space_top=.10
3197        !
3198        ! Background parameters...........................
3199        !
3200        Graphtype$="LINEAR"
3201        Xgrid_tick$="TICK"
3202        Ygrid_tick$="TICK"
3203        Xmajor=6
3204        Ymajor=1
3205        Xminor=3
3206        Yminor=1
3207        Xmin_graph=0
3208        Xmax_graph=1.8E+10
3209        Ymin_graph=0
3210        Ymax_graph=1
3211        X_cross_y=Ymin_graph
3212        Y_cross_x=Xmin_graph
3213        !
3214        Xmin=Xmin_graph
3215        Xmax=Xmax_graph
3216        Xmid=Xmin+((Xmax-Xmin)/2)
3217        Ymin=Ymin_graph
3218        Ymax=Ymax_graph
3219        Ymid=Ymin+((Ymax-Ymin)/2)
3220        !
3221        ! Label parameters................................
3222        !
3223        Lbl_count=0
3224        MAT Labels$= ("")
3225        MAT Lbl_addr= (0)
3226        !
3227        ! Insert desired labels here using UDUs defined above.
3228        !
3229        Lbl_count=3
3230 !      Labels$(1)="Rotation angle (degrees)"
3231        Labels$(1)=Label1$
3232        Lbl_addr(1,1)=9.E+9 ! X location UDUs
3233        Lbl_addr(1,2)=-.05  ! Y location UDUs
3234        Lbl_addr(1,3)=5     ! PEN
3235        Lbl_addr(1,4)=.035  ! SIZE factor
3236        Lbl_addr(1,5)=0     ! LDIR
3237        Lbl_addr(1,6)=6     ! LORG
3238        !
3239 !      Labels$(2)="Relative amplitude (dB)"
3240        Labels$(2)=Label2$
3241        Lbl_addr(2,1)=-2.E+9! X location UDUs
3242        Lbl_addr(2,2)=.5    ! Y location UDUs
3243        Lbl_addr(2,3)=5     ! PEN
3244        Lbl_addr(2,4)=.035  ! SIZE factor
3245        Lbl_addr(2,5)=90    ! LDIR
3246        Lbl_addr(2,6)=4     ! LORG
3247        !
3248 !      Labels$(3)="Probe Calibration Data"
3249        Labels$(3)=Label3$
3250        Lbl_addr(3,1)=9.E+9 ! X location UDUs
3251        Lbl_addr(3,2)=1.02  ! Y location UDUs
3252        Lbl_addr(3,3)=5     ! PEN
3253        Lbl_addr(3,4)=.040  ! SIZE factor
3254        Lbl_addr(3,5)=0     ! LDIR
3255        Lbl_addr(3,6)=4     ! LORG
3256        !
3257        SUBEXIT
3258    SUBEND
3259    !
3260    ! ****************************************************************
3261    !
3262    SUB Plot_all(File(*))
3263 Plot_all:  ! Original: 13 Nov 1984
3264            ! Revision: 06 Aug 1987
3265        OPTION BASE 1
3266        DEG
3267        !
3268        COM /Sys/ Sys_id$[10]
3269        COM /Labels/ Labels$(*),INTEGER Lbl_count,REAL Lbl_addr(*)
3270        COM /Data_param/ INTEGER Datacount,Filesize,Curvecount,Roster(*)
3271        COM /Data_param/ REAL Sym_size,Symbol$(*),Curve_id$(*)
3272        COM /Data_param/ REAL Xmin_data,Xmax_data
3273        COM /Data_param/ REAL Ymin_data,Ymax_data
3274        COM /Background/ Graphtype$,Margins$(*),Papersize$
3275        COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
3276        COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
3277        COM /Background/ Xgrid_tick$,INTEGER Xmajor,Xminor
3278        COM /Background/ Ygrid_tick$,INTEGER Ymajor,Yminor
3279        COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
3280        !
3281        COM /Axes_labels/ Print_xlabels$,Print_ylabels$
3282        COM /Axes_labels/ Sig_digits$,REAL Xlcsize,Ylcsize
3283        !
3284        COM /Windowspace/ REAL Xmin,Xmid,Xmax,Ymin,Ymid,Ymax!graph edges UDU
3285        COM /Windowspace/ REAL Xleft,Xright,Ybottom,Ytop! paper edges UDUs
3286        COM /Plot_device/ Plot_lang$,INTEGER Plot_addr
3287        COM /Hard_space/ REAL Xview_lft,Xview_rt,Yview_btm,Yview_top
3288        COM /Hard_space/ REAL Viewscale,Aspect_ratio
3289        COM /Frame_onoff/ INTEGER Frame_flag
3290        COM /Clear_space/ REAL Space_lft,Space_rt,Space_btm,Space_top
3291        COM /Bugs/ INTEGER Bug1,Bug2,Bug3,Printer
3292        COM /Files/ Diskdrive$,Filename$
3293        INTEGER I,J,Local_prty,Suspended,Outofbounds
3294        !
3295        Aspect_ratio=(Xview_rt-Xview_lft)/((Yview_top-Yview_btm)*2)
3296        Viewscale=MIN((Xview_rt-Xview_lft),(Yview_top-Yview_btm))
3297        Aspect_ratio=MIN(MAX(Aspect_ratio,.35),.65)
3298        !
3299        Local_prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
3300        IF Lbl_count>0 THEN
3301            ALLOCATE Lbl_ratio(Lbl_count,2)
3302            GOSUB Update_lbl_addr     ! includes Find_mid_pt
3303            DEALLOCATE Lbl_ratio(*)
3304        ELSE
3305            GOSUB Find_mid_pt
3306        END IF
3307        CALL Background(Suspended)
3308        IF Suspended THEN GOTO Nomoregraph
3309        IF Plot_addr<>3 THEN
3310            ON KEY 0 LABEL "ABORT  GRAPH",Local_prty+1 G(    Nomoregraph
3311        END IF
3312        CLIP OFF
3313        GOSUB Label_labels
3314        GOSUB Plot_data
3315 Nomoregraph:IF Plot_addr<>3 THEN OFF KEY
3316        PENUP
3317        PEN 0
3318        CLIP OFF
3319        SUBEXIT
3320        !
3321        ! /////////////////////////////////////////////////////////////////
3322        !
3323 Update_lbl_addr:! Make sure that the label addresses are current.
3324        FOR I=1 TO Lbl_count
3325            Lbl_ratio(I,1)=(Lbl_addr(I,1)-Xmin)/(Xmax-Xmin)
3326            Lbl_ratio(I,2)=(Lbl_addr(I,2)-Ymin)/(Ymax-Ymin)
```

```
3327        NEXT I
3328        GOSUB Find_mid_pt             ! define new max and min
3329        FOR I=1 TO Lbl_count
3330            Lbl_addr(I,1)=Lbl_ratio(I,1)*(Xmax-Xmin)+Xmin
3331            Lbl_addr(I,2)=Lbl_ratio(I,2)*(Ymax-Ymin)+Ymin
3332        NEXT I
3333        RETURN
3334        !
3335        ! ////////////////////////////////////////////////////////////////
3336        !
3337 Find_mid_pt:      !Interpret the graph type and the scaling done.
3338        SELECT Graphtype$
3339        CASE "LINEAR"
3340            Xmin=Xmin_graph
3341            Xmax=Xmax_graph
3342            Xmid=((Xmax-Xmin)/2)+Xmin
3343            Ymin=Ymin_graph
3344            Ymax=Ymax_graph
3345            Ymid=((Ymax-Ymin)/2)+Ymin
3346        CASE "SEMILOG X"
3347            Xmin=0
3348            Xmax=100
3349            Xmid=50
3350            Ymin=Ymin_graph
3351            Ymax=Ymax_graph
3352            Ymid=((Ymax-Ymin)/2)+Ymin
3353        CASE "SEMILOG Y"
3354            Xmin=Xmin_graph
3355            Xmax=Xmax_graph
3356            Xmid=((Xmax-Xmin)/2)+Xmin
3357            Ymin=0
3358            Ymax=100
3359            Ymid=50
3360        CASE "LOG LOG"
3361            Xmin=0
3362            Xmax=100
3363            Xmid=50
3364            Ymin=0
3365            Ymax=100
3366            Ymid=50
3367        CASE "POLAR"
3368            !
3369            BEEP
3370            DISP "POLAR PARAMETERS ARE NOT YET IMPLEMENTED!!!!"
3371            PAUSE
3372        END SELECT
3373        RETURN
3374        !
3375        ! ////////////////////////////////////////////////////////////////
3376        !
3377 Label_labels:        !ALL LABELS ARE APPLIED
3378        PEN Backgnd_pen
3379        CSIZE DROUND(.025*Viewscale,3),Aspect_ratio
3380        IF Auto_time THEN
3381            LORG 1
3382            MOVE Xleft,Ybottom
3383            LABEL " "&DATE$(TIMEDATE)&", "&TIME$(TIMEDATE)
3384            PENUP
3385        END IF
3386        IF Auto_file THEN
3387            LORG 7
3388            MOVE Xright,Ybottom
3389            IF LEN(Filename$)>0 THEN
3390                LABEL "File: "&Filename$&"  "
3391            END IF
3392            PENUP
3393        END IF
3394        FOR I=1 TO Lbl_count
3395            LDIR Lbl_addr(I,5)
3396            IF Lbl_addr(I,6)>0 THEN
3397                LORG Lbl_addr(I,6)
3398            ELSE
3399                LORG 1
3400            END IF
3401            CSIZE DROUND(Lbl_addr(I,4)*Viewscale,3),Aspect_ratio
3402            PEN DROUND(Lbl_addr(I,3),1)
3403            MOVE Lbl_addr(I,1),Lbl_addr(I,2)
3404            LABEL USING "#,K";Labels$(I)
3405            PENUP
3406        NEXT I
3407        RETURN
3408        !
3409        ! ////////////////////////////////////////////////////////////////
3410        !
3411 Plot_data:             !PLOT ALL CURVES
3412        Disptime=0.
3413        CLIP Xmin,Xmax,Ymin,Ymax
3414        LORG 5
3415        LDIR 0
3416        CSIZE DROUND(.025*Viewscale,3),Aspect_ratio
3417        FOR I=1 TO Curvecount
3418            Outofbounds=0
3419            PENUP
3420            PEN Roster(I,4)
3421            Xpos=File(Roster(I,2),1)
3422            Ypos=File(Roster(I,2),2)
3423            GOSUB Adjust_xy_pos
3424            IF Xpos>Xmax THEN
3425                GOSUB Skipcurvei
3426                GOTO Skipcurvej
3427            END IF
3428            Xpos=MAX(MIN(Xpos,Xmax),Xmin)
3429            Ypos=MAX(MIN(Ypos,Ymax),Ymin)
3430            MOVE Xpos,Ypos
3431            SELECT LEN(Symbol$(Roster(I,1)))
3432            CASE 2
3433                Mark$=Symbol$(Roster(I,1))[1,1]
3434                Connect$=Symbol$(Roster(I,1))[2,2]
3435            CASE 1
3436                Mark$=Symbol$(Roster(I,1))[1,1]
3437                Connect$="Y"
3438            CASE ELSE
3439                Mark$=""
3440                Connect$="Y"
3441            END SELECT
3442            !
3443            IF Mark$="" AND Connect$="Y" THEN
3444                !No symbol
3445                FOR J=Roster(I,2) TO Roster(I,2)+Roster(I,3)-1
3446                    Xpos=File(J,1)
3447                    Ypos=File(J,2)
3448                    GOSUB Adjust_xy_pos
3449                    IF Xpos>Xmax THEN
3450                        GOSUB Skipcurvei
3451                        GOTO Skipcurvej
3452                    END IF
3453                    IF NOT Outofbounds THEN
3454                        IF Xpos<Xmin OR Ypos<Ymin OR Ypos>Ymax THEN
3455                            Outofbounds=1
3456                            GOSUB Skipcurvei
3457                        END IF
3458                    END IF
```

```
3459                   Xpos=MAX(MIN(Xpos,Xmax),Xmin)
3460                   Ypos=MAX(MIN(Ypos,Ymax),Ymin)
3461                   DRAW Xpos,Ypos
3462                 NEXT J
3463               ELSE
3464                 !Put in symbol
3465                 SELECT Connect$
3466                 CASE "Y"
3467                   FOR J=Roster(I,2) TO Roster(I,2)+Roster(I,3)-1
3468                     Xpos=File(J,1)
3469                     Ypos=File(J,2)
3470                     GOSUB Adjust_xy_pos
3471                     IF Xpos>Xmax THEN
3472                       GOSUB Skipcurvei
3473                       GOTO Skipcurvej
3474                     END IF
3475                     IF NOT Outofbounds THEN
3476                       IF Xpos<Xmin OR Ypos<Ymin OR Ypos>Ymax THEN
3477                         Outofbounds=1
3478                         GOSUB Skipcurvei
3479                       END IF
3480                     END IF
3481                     Xpos=MAX(MIN(Xpos,Xmax),Xmin)
3482                     Ypos=MAX(MIN(Ypos,Ymax),Ymin)
3483                     DRAW Xpos,Ypos
3484                     LABEL USING "#,K";Mark$
3485                     PENUP
3486                     MOVE Xpos,Ypos
3487                   NEXT J
3488                 CASE "N"
3489                   FOR J=Roster(I,2) TO Roster(I,2)+Roster(I,3)-1
3490                     Xpos=File(J,1)
3491                     Ypos=File(J,2)
3492                     GOSUB Adjust_xy_pos
3493                     IF Xpos>Xmax THEN
3494                       GOSUB Skipcurvei
3495                       GOTO Skipcurvej
3496                     END IF
3497                     IF NOT Outofbounds THEN
3498                       IF Xpos<Xmin OR Ypos<Ymin OR Ypos>Ymax THEN
3499                         Outofbounds=1
3500                         GOSUB Skipcurvei
3501                       END IF
3502                     END IF
3503                     Xpos=MAX(MIN(Xpos,Xmax),Xmin)
3504                     Ypos=MAX(MIN(Ypos,Ymax),Ymin)
3505                     MOVE Xpos,Ypos
3506                     LABEL USING "#,K";Mark$
3507                     PENUP
3508                   NEXT J
3509                 END SELECT
3510               END IF
3511 Skipcurvej: !
3512               PENUP
3513             NEXT I
3514             IF Disptime>0 THEN
3515               LOOP
3516               EXIT IF TIMEDATE-Disptime>1.2
3517               END LOOP
3518               DISP CHR$(12)
3519             END IF
3520             RETURN
3521             !
3522             ! ////////////////////////////////////////////////////////////
3523             !
3524 Skipcurvei: !
```

```
3525       DISP " DATA OUT OF RANGE ... NOT PLOTTED "
3526       IF Disptime<1.0E-10 THEN
3527         Disptime=TIMEDATE
3528       END IF
3529       RETURN
3530       !
3531       ! ////////////////////////////////////////////////////////////
3532       !
3533 Adjust_xy_pos:     !CORRECT Xpos, Ypos FOR VARIOUS AXES
3534       SELECT Graphtype$
3535       CASE "LINEAR"
3536         !NO CHANGE
3537       CASE "SEMILOG X"
3538         Xpos=FNLin_map_logx(Xpos)
3539         IF Xpos<0 THEN Xpos=0
3540       CASE "SEMILOG Y"
3541         Ypos=FNLin_map_logy(Ypos)
3542         IF Ypos<0 THEN Ypos=0
3543       CASE "LOG LOG"
3544         Xpos=FNLin_map_logx(Xpos)
3545         Ypos=FNLin_map_logy(Ypos)
3546         IF Xpos<0 THEN Xpos=0
3547         IF Ypos<0 THEN Ypos=0
3548       CASE "POLAR"
3549         !NO CHANGE yet.
3550       END SELECT
3551       RETURN
3552       !
3553       ! ////////////////////////////////////////////////////////////
3554       !
3555       SUBEXIT
3556  SUBEND
3557  !
3558  ! ************************************************************
3559  !
3560  SUB Backgrcund(INTEGER Suspended)
3561 Background:  ! Original: 13 Nov 1984
3562              ! Revision: 06 Aug 1987
3563      !This SUB program is written to draw the background for the PLOT
3564      !program.  It draws LINEAR, SEMILOG, LOG LOG, and POLAR backgrounds.
3565      !Parameters are determined in the main program.
3566      !
3567      !Due to the complexity of LOG and POLAR coordinates, the TICKing is
3568      !done with MOVEs and DRAWs.  Also avoided are device dependent codes.
3569      !
3570       OPTION BASE 1
3571       DEG
3572       COM /Sys/ Sys_id$
3573       COM /Background/ Graphtype$,Margins$(*),Papersize$
3574       COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
3575       COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
3576       COM /Background/ Xgrid_tick$,INTEGER Xmajor,Xminor
3577       COM /Background/ Ygrid_tick$,INTEGER Ymajor,Yminor
3578       COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
3579       COM /Windowspace/ REAL Xmin,Xmid,Xmax,Ymin,Ymid,Ymax! graph edges
3580       COM /Windowspace/ REAL Edge_lft,Edge_rt,Edge_btm,Edge_top
3581       COM /Plot_device/ Plot_lang$[10],INTEGER Plot_addr
3582       COM /Hard_space/ REAL Xview_lft,Xview_rt,Yview_btm,Yview_top
3583       COM /Hard_space/ REAL Viewscale,Aspect_ratio
3584       COM /Log_scale/ REAL Xcycles,Xbegin,Ycycles,Ybegin
3585       COM /Axes_labels/ Print_xlabels$,Print_ylabels$
3586       COM /Axes_labels/ Sig_digits$,REAL Xlcsize,Ylcsize
3587       COM /Frame_onoff/ INTEGER Frame_flag
3588       COM /Clear_space/ REAL Space_lft,Space_rt,Space_btm,Space_top
3589       !
3590       COM /Bugs/ INTEGER Bug1,Bug2,Bug3,Printer
```

```
3591      DIM Digies$[1]
3592      INTEGER Tic_size,Toggle,Log_label_limit,Log_label_step
3593      INTEGER Txminor,Tyminor
3594      REAL Xcrossy,Ycrossx,Xming,Xmaxg,Yming,Ymaxg
3595      !
3596      GRAPHICS ON
3597      IF Print_xlabels$="" THEN Print_xlabels$="YES"
3598      IF Print_ylabels$="" THEN Print_ylabels$="YES"
3599      IF Plot_addr<>3 THEN
3600          Suspended=0
3601          CALL Hp7475a_setup(Suspended)
3602          IF Suspended THEN GOTO End_plotting
3603          ON KEY 0 LABEL "ABORT  GRAPH",Loacal_prty+1 GOTO End_plotting
3604      ELSE
3605          VIEWPORT Xview_lft,Xview_rt,Yview_btm,Yview_top
3606          IF Frame_flag THEN FRAME
3607      END IF
3608      Tic_size=DROUND(.04*Viewscale,3)
3609      LDIR 0
3610      Log_tic_size=.04   !Per cent of total range.
3611      Tic_ratio=(Yview_top-Yview_btm)/(Xview_rt-Xview_lft)
3612      Xming=Xmin_graph  !copy for use here
3613      Xmaxg=Xmax_graph
3614      Yming=Ymin_graph
3615      Ymaxg=Ymax_graph
3616      Xcrossy=X_cross_y
3617      Ycrossx=Y_cross_x
3618      IF Ymajor<1 THEN Ymajor=1
3619      IF Yminor<1 THEN Yminor=1
3620      IF Xmajor<1 THEN Xmajor=1
3621      IF Xminor<1 THEN Xminor=1
3622      IF Xgrid_tick$="GRID" AND Xminor<>1 THEN
3623          Txminor=1
3624      ELSE
3625          Txminor=Xminor
3626      END IF
3627      IF Ygrid_tick$="GRID" AND Yminor<>1 THEN
3628          Tyminor=1
3629      ELSE
3630          Tyminor=Yminor
3631      END IF
3632      !
3633      Log_label_limit=4
3634      !
3635      PEN Backgnd_pen
3636      SELECT Graphtype$
3637      CASE "LINEAR"
3638          ! Set up user units.  Outside edges
3639          Xtics=(Xmaxg-Xming)/(Xmajor*Txminor)
3640          Ytics=(Ymaxg-Yming)/(Ymajor*Tyminor)
3641          GOSUB Window_space
3642          GOSUB Draw_x_linear
3643          IF Print_xlabels$="YES" THEN GOSUB Label_x_linear
3644          GOSUB Draw_y_linear
3645          IF Print_ylabels$="YES" THEN GOSUB Label_y_linear
3646          !
3647      CASE "SEMILOG X"
3648          ! Set up user units.  Outside edges
3649          Xming=0.        !All log operations are mapped into
3650          Xmaxg=100.0     !this range.  Y axis range is unchanged.
3651          Xtic_size=Log_tic_size*(Ymaxg-Yming)
3652          Find_log_cycles(Xmin_graph,Xmax_graph,Xcycles,Xbegin,Xend)
3653          IF Y_cross_x<2*Xbegin THEN
3654              Ycrossx=0.
3655          ELSE
3656              Ycrossx=FNLin_map_logx(Y_cross_x)
3657          END IF
3658          Ytics=(Ymaxg-Yming)/(Ymajor*Tyminor)
3659          GOSUB Window_space
3660          GOSUB Draw_x_log
3661          IF Print_xlabels$="YES" THEN GOSUB Label_x_log
3662          GOSUB Draw_y_linear
3663          IF Print_ylabels$="YES" THEN GOSUB Label_y_linear
3664          !
3665      CASE "SEMILOG Y"
3666          ! Set up user units.  Outside edges
3667          Yming=0.        !All log operations are mapped into this
3668          Ymaxg=100.      !range.  X axis is unchanged.
3669          Ytic_size=Log_tic_size*(Xmaxg-Xming)*Tic_ratio
3670          Find_log_cycles(Ymin_graph,Ymax_graph,Ycycles,Ybegin,Yend)
3671          IF X_cross_y<2*Ybegin THEN
3672              Xcrossy=0.
3673          ELSE
3674              Xcrossy=FNLin_map_logy(X_cross_y)
3675          END IF
3676          Xtics=(Xmaxg-Xming)/(Xmajor*Txminor)
3677          GOSUB Window_space
3678          GOSUB Draw_x_linear
3679          IF Print_xlabels$="YES" THEN GOSUB Label_x_linear
3680          GOSUB Draw_y_log
3681          IF Print_ylabels$="YES" THEN GOSUB Label_y_log
3682          !
3683      CASE "LOG LOG"
3684          ! Set up user units.  Outside edges
3685          Xming=0.      !All log operations are mapped into this
3686          Xmaxg=100.    !range.   Both axes.
3687          Yming=0.
3688          Ymaxg=100.
3689          Xtic_size=Log_tic_size*(Ymaxg-Yming)
3690          Ytic_size=Log_tic_size*(Xmaxg-Xming)*Tic_ratio
3691          Find_log_cycles(Xmin_graph,Xmax_graph,Xcycles,Xbegin,Xend)
3692          Find_log_cycles(Ymin_graph,Ymax_graph,Ycycles,Ybegin,Yend)
3693          IF X_cross_y<2*Ybegin THEN
3694              Xcrossy=0.
3695          ELSE
3696              Xcrossy=FNLin_map_logy(X_cross_y)
3697          END IF
3698          IF Y_cross_x<2*Xbegin THEN
3699              Ycrossx=0.
3700          ELSE
3701              Ycrossx=FNLin_map_logx(Y_cross_x)
3702          END IF
3703          GOSUB Window_space
3704          GOSUB Draw_x_log
3705          IF Print_xlabels$="YES" THEN GOSUB Label_x_log
3706          GOSUB Draw_y_log
3707          IF Print_ylabels$="YES" THEN GOSUB Label_y_log
3708          !
3709      CASE "POLAR"
3710          !??????????
3711      END SELECT
3712      GOTO Back_done
3713 End_plotting:Suspended=1
3714 Back_done:   !
3715      IF Plot_addr<>3 THEN OFF KEY
3716      PEN 0
3717      SUBEXIT
3718    !
3719    ! ////////////////////////////////////////////////////////////
3720    !
3721 Window_space:     !Allow room outside of curve box for labels.
3722                   !Also scale to curve box and turn on CLIP to these
```

```
3723                    !limits.
3724        IF Margins$(2)="SQUARE" THEN
3725            Edge_lft=Xming-.21*(Xmaxg-Xming)
3726            Edge_rt=Xmaxg+.09*(Xmaxg-Xming)
3727            Edge_btm=Yming-.15*(Ymaxg-Yming)
3728            Edge_top=Ymaxg+.15*(Ymaxg-Yming)
3729        ELSE
3730            Edge_lft=Xming-Space_lft*(Xmaxg-Xming)
3731            Edge_rt=Xmaxg+Space_rt*(Xmaxg-Xming)
3732            Edge_btm=Yming-Space_btm*(Ymaxg-Yming)
3733            Edge_top=Ymaxg+Space_top*(Ymaxg-Yming)
3734        END IF
3735        WINDOW Edge_lft,Edge_rt,Edge_btm,Edge_top
3736        CLIP Xming,Xmaxg,Yming,Ymaxg
3737        IF Frame_flag THEN FRAME
3738        RETURN
3739       !
3740       ! ///////////////////////////////////////////////////////////////
3741       !
3742 Draw_x_linear:     !
3743            !Draw X (horizontal) axes.
3744        SELECT Xgrid_tick$
3745        CASE "TICK"
3746            AXES Xtics,0,Xming-Xtics*Txminor,Xcrossy,Txminor,Tyminor,Tic_size
3747            !Select opposite axis if necessary
3748            SELECT Xcrossy
3749            CASE <=Yming+1.0E-10*(Ymaxg-Yming)
3750                AXES Xtics,0,Xming-Xtics*Txminor,Ymaxg,Txminor,Tyminor,Tic_si
  => ze
3751            CASE >=Ymaxg-1.0E-10*(Ymaxg-Yming)
3752                AXES Xtics,0,Xming-Xtics*Txminor,Yming,Txminor,Tyminor,Tic_si
  => ze
3753            END SELECT
3754        CASE "GRID"
3755            GRID Xtics,0,Ycrossx,Xcrossy,Txminor,Tyminor,Tic_size
3756        END SELECT
3757        RETURN
3758       !
3759       ! ///////////////////////////////////////////////////////////////
3760       !
3761 Draw_y_linear:     !
3762            !Draw Y (vertical) axes.
3763        SELECT Ygrid_tick$
3764        CASE "TICK"
3765            AXES 0,Ytics,Ycrossx,Yming-Ytics*Tyminor,Txminor,Tyminor,Tic_size
3766            !Select the opposite axis if necessary.
3767            SELECT Ycrossx
3768            CASE <=Xming+1.0E-10*(Xmaxg-Xming)
3769                AXES 0,Ytics,Xmaxg,Yming-Ytics*Tyminor,Txminor,Tyminor,Tic_si
  => ze
3770            CASE >=Xmaxg-1.0E-10*(Xmaxg-Xming)
3771                AXES 0,Ytics,Xming,Yming-Ytics*Tyminor,Txminor,Tyminor,Tic_si
  => ze
3772            END SELECT
3773        CASE "GRID"
3774            GRID 0,Ytics,Ycrossx,Xcrossy,Txminor,Tyminor,Tic_size
3775        END SELECT
3776        RETURN
3777       !
3778       ! ///////////////////////////////////////////////////////////////
3779       !
3780 Draw_x_log:      !Draw log axis according to x parameters.
3781        Full_cycles=INT(Xcycles)
3782        Div_beyond=10*FRACT(Xcycles)
3783        SELECT Xgrid_tick$
3784        CASE "TICK"
3785            MOVE Xming,Xcrossy-(Xtic_size/2)
3786            RPLOT 0,Xtic_size,-1
3787            FOR C=1 TO Full_cycles
3788                FOR D=2 TO 10
3789                    Xloc=FNLin_map_logx(D*Xbegin*(10^(C-1)))
3790                    MOVE Xloc,Xcrossy-(Xtic_size/2)
3791                    RPLOT 0,Xtic_size,-1
3792                NEXT D
3793            NEXT C
3794            FOR D=1 TO Div_beyond
3795                Xloc=FNLin_map_logx((D+1)*Xbegin*(10^(Full_cycles)))
3796                MOVE Xloc,Xcrossy-(Xtic_size/2)
3797                RPLOT 0,Xtic_size,-1
3798            NEXT D
3799            MOVE Xmaxg,Xcrossy-(Xtic_size/2)
3800            RPLOT 0,Xtic_size,-1
3801            MOVE Xmaxg,Xcrossy
3802            DRAW Xming,Xcrossy
3803            PENUP
3804            !
3805            !CHECK FOR OPPOSITE AXIS.
3806            !
3807            IF Xcrossy=Yming OR Xcrossy=Ymaxg THEN
3808                !Repeat for the opposite axis
3809                IF Xcrossy=Yming THEN
3810                    Second_axis=Ymaxg
3811                ELSE
3812                    Second_axis=Yming
3813                END IF
3814                MOVE Xming,Second_axis-(Xtic_size/2)
3815                RPLOT 0,Xtic_size,-1
3816                FOR C=1 TO Full_cycles
3817                    FOR D=2 TO 10
3818                        Xloc=FNLin_map_logx(D*Xbegin*(10^(C-1)))
3819                        MOVE Xloc,Second_axis-(Xtic_size/2)
3820                        RPLOT 0,Xtic_size,-1
3821                    NEXT D
3822                NEXT C
3823                FOR D=1 TO Div_beyond
3824                    Xloc=FNLin_map_logx((D+1)*Xbegin*(10^(Full_cycles)))
3825                    MOVE Xloc,Second_axis-(Xtic_size/2)
3826                    RPLOT 0,Xtic_size,-1
3827                NEXT D
3828                MOVE Xmaxg,Second_axis-(Xtic_size/2)
3829                RPLOT 0,Xtic_size,-1
3830                MOVE Xmaxg,Second_axis
3831                DRAW Xming,Second_axis
3832                PENUP
3833            END IF
3834        CASE "GRID"
3835            MOVE Xming,Yming
3836            DRAW Xming,Ymaxg
3837            Toggle=+1      !Toggles between Ymaxg and Yming.
3838                           !+1=Ymaxg,  -1=Yming
3839            FOR C=1 TO Full_cycles
3840                FOR D=2 TO 10
3841                    Xloc=FNLin_map_logx(D*Xbegin*(10^(C-1)))
3842                    SELECT Toggle
3843                    CASE +1
3844                        MOVE Xloc,Ymaxg
3845                        DRAW Xloc,Yming
3846                        Toggle=-Toggle
3847                    CASE -1
3848                        MOVE Xloc,Yming
3849                        DRAW Xloc,Ymaxg
3850                        Toggle=-Toggle
```

```
3851                END SELECT
3852            NEXT D
3853        NEXT C
3854        FOR D=1 TO Div_beyond+1
3855            Xloc=FNLin_map_logx((D+1)*Xbegin*(10^(Full_cycles)))
3856            SELECT Toggle
3857            CASE +1
3858                MOVE Xloc,Ymaxg
3859                DRAW Xloc,Yming
3860                Toggle=-Toggle
3861            CASE -1
3862                MOVE Xloc,Yming
3863                DRAW Xloc,Ymaxg
3864                Toggle=-Toggle
3865            END SELECT
3866        NEXT D
3867    END SELECT
3868    RETURN
3869    !
3870    ! //////////////////////////////////////////////////////////
3871    !
3872 Draw_y_log:      !Draw log axis according to Y parameters.
3873    Full_cycles=INT(Ycycles)
3874    Div_beyond=10*FRACT(Ycycles)
3875    SELECT Ygrid_tick$
3876    CASE "TICK"
3877        MOVE Ycrossx-(Ytic_size/2),Yming
3878        RPLOT Ytic_size,0,-1
3879        FOR C=1 TO Full_cycles
3880            FOR D=2 TO 10
3881                Yloc=FNLin_map_logy(D*Ybegin*(10^(C-1)))
3882                MOVE Ycrossx-(Ytic_size/2),Yloc
3883                RPLOT Ytic_size,0,-1
3884            NEXT D
3885        NEXT C
3886        FOR D=1 TO Div_beyond
3887            Yloc=FNLin_map_logy((D+1)*Ybegin*(10^(Full_cycles)))
3888            MOVE Ycrossx-(Ytic_size/2),Yloc
3889            RPLOT Ytic_size,0,-1
3890        NEXT D
3891        MOVE Ycrossx-(Ytic_size/2),Ymaxg
3892        RPLOT Xtic_size,0,-1
3893        MOVE Ycrossx,Ymaxg
3894        DRAW Ycrossx,Yming
3895        PENUP
3896        !
3897        !CHECK FOR OPPOSITE AXIS.
3898        !
3899        IF Ycrossx=Xming OR Ycrossx=Xmaxg THEN
3900            !Repeat for the opposite axis
3901            IF Ycrossx=Xming THEN
3902                Second_axis=Xmaxg
3903            ELSE
3904                Second_axis=Xming
3905            END IF
3906            MOVE Second_axis-(Ytic_size/2),Yming
3907            RPLOT Ytic_size,0,-1
3908            FOR C=1 TO Full_cycles
3909                FOR D=2 TO 10
3910                    Yloc=FNLin_map_logy(D*Ybegin*(10^(C-1)))
3911                    MOVE Second_axis-(Ytic_size/2),Yloc
3912                    RPLOT Ytic_size,0,-1
3913                NEXT D
3914            NEXT C
3915            FOR D=1 TO Div_beyond
3916                Yloc=FNLin_map_logy((D+1)*Ybegin*(10^(Full_cycles)))
3917                MOVE Second_axis-(Ytic_size/2),Yloc
3918                RPLOT Ytic_size,0,-1
3919            NEXT D
3920            MOVE Second_axis-(Ytic_size/2),Ymaxg
3921            RPLOT Ytic_size,0,-1
3922            MOVE Second_axis,Ymaxg
3923            DRAW Second_axis,Yming
3924            PENUP
3925        END IF
3926    CASE "GRID"
3927        MOVE Xming,Yming
3928        DRAW Xmaxg,Yming
3929        Toggle=+1      !Toggles between Ymaxg and Yming.
3930                       !+1=Ymaxg,  -1=Yming
3931        FOR C=1 TO Full_cycles
3932            FOR D=2 TO 10
3933                Yloc=FNLin_map_logy(D*Ybegin*(10^(C-1)))
3934                SELECT Toggle
3935                CASE +1
3936                    MOVE Xmaxg,Yloc
3937                    DRAW Xming,Yloc
3938                    Toggle=-Toggle
3939                CASE -1
3940                    MOVE Xming,Yloc
3941                    DRAW Xmaxg,Yloc
3942                    Toggle=-Toggle
3943                END SELECT
3944            NEXT D
3945        NEXT C
3946        FOR D=1 TO Div_beyond+1
3947            Yloc=FNLin_map_logy((D+1)*Ybegin*(10^(Full_cycles)))
3948            SELECT Toggle
3949            CASE +1
3950                MOVE Xmaxg,Yloc
3951                DRAW Xming,Yloc
3952                Toggle=-Toggle
3953            CASE -1
3954                MOVE Xming,Yloc
3955                DRAW Xmaxg,Yloc
3956                Toggle=-Toggle
3957            END SELECT
3958        NEXT D
3959    END SELECT
3960    RETURN
3961    !
3962    ! //////////////////////////////////////////////////////////
3963    !
3964 Fixed_sig_digit: ! Assure equal significant digits to right of .
3965                  ! Sig_digits$ = 1, 2, 3, or 4
3966                  !
3967    SELECT ABS(Numeric_label)
3968    CASE 1.0 TO 9.99
3969        SELECT Numeric_label
3970        CASE PROUND(Numeric_label,-INT(VAL(Digies$)))
3971            SELECT Digies$
3972            CASE "1"
3973                LABEL USING "MD.D";Numeric_label
3974            CASE "2"
3975                LABEL USING "MD.DD";Numeric_label
3976            CASE "3"
3977                LABEL USING "MD.3D";Numeric_label
3978            CASE "4"
3979                LABEL USING "MD.4D";Numeric_label
3980            END SELECT
3981        CASE ELSE
3982            LABEL USING "MD.DDESZ";Numeric_label
```

```
3983            END SELECT
3984            !
3985        CASE 9.99 TO 99.99
3986            SELECT Numeric_label
3987            CASE PROUND(Numeric_label,-INT(VAL(Digies$)))
3988                SELECT Digies$
3989                CASE "1"
3990                    LABEL USING "M2D.D";Numeric_label
3991                CASE "2"
3992                    LABEL USING "M2D.DD";Numeric_label
3993                CASE "3"
3994                    LABEL USING "M2D.3D";Numeric_label
3995                CASE "4"
3996                    LABEL USING "M2D.4D";Numeric_label
3997                END SELECT
3998            CASE ELSE
3999                LABEL USING "MD.DDESZ";Numeric_label
4000            END SELECT
4001            !
4002        CASE 99.99 TO 999.99
4003            SELECT Numeric_label
4004            CASE PROUND(Numeric_label,-INT(VAL(Digies$)))
4005                SELECT Digies$
4006                CASE "1"
4007                    LABEL USING "M3D.D";Numeric_label
4008                CASE "2"
4009                    LABEL USING "M3D.DD";Numeric_label
4010                CASE "3"
4011                    LABEL USING "M3D.3D";Numeric_label
4012                CASE "4"
4013                    LABEL USING "M3D.4D";Numeric_label
4014                END SELECT
4015            CASE ELSE
4016                LABEL USING "MD.DDESZ";Numeric_label
4017            END SELECT
4018            !
4019        CASE 999.99 TO 9999.99
4020            SELECT Numeric_label
4021            CASE PROUND(Numeric_label,-INT(VAL(Digies$)))
4022                SELECT Digies$
4023                CASE "1"
4024                    LABEL USING "M4D.D";Numeric_label
4025                CASE "2"
4026                    LABEL USING "M4D.DD";Numeric_label
4027                CASE "3"
4028                    LABEL USING "M4D.3D";Numeric_label
4029                CASE "4"
4030                    LABEL USING "M4D.4D";Numeric_label
4031                END SELECT
4032            CASE ELSE
4033                LABEL USING "MD.DDESZ";Numeric_label
4034            END SELECT
4035            !
4036        CASE >9999.99
4037            SELECT ABS(Numeric_label)
4038            CASE <1.0E+10
4039                LABEL USING "MD.DDESZ";Numeric_label
4040            CASE <1.0E+100
4041                LABEL USING "MD.DDESZZ";Numeric_label
4042            CASE ELSE
4043                LABEL USING "MD.DDESZZZ";Numeric_label
4044            END SELECT
4045            !
4046            !++++++All values less that 1.0 ++++++++++
4047            !
4048        CASE .0001 TO 1.0
4049            SELECT Numeric_label
4050            CASE PROUND(Numeric_label,-INT(VAL(Digies$)))
4051                SELECT Digies$
4052                CASE "1"
4053                    LABEL USING "MZ.D";Numeric_label
4054                CASE "2"
4055                    LABEL USING "MZ.DD";Numeric_label
4056                CASE "3"
4057                    LABEL USING "MZ.3D";Numeric_label
4058                CASE "4"
4059                    LABEL USING "MZ.4D";Numeric_label
4060                END SELECT
4061            CASE ELSE
4062                LABEL USING "MD.DDESZ";Numeric_label
4063            END SELECT
4064            !
4065        CASE ELSE
4066            IF ABS(Numeric_label)>1.0E-99 THEN
4067                IF PROUND(Numeric_label,-9)=Numeric_label THEN
4068                    LABEL USING "MD.DDESZ";Numeric_label
4069                ELSE
4070                    LABEL USING "MD.DDESZZ";Numeric_label
4071                END IF
4072            ELSE
4073                SELECT Digies$
4074                CASE "1"
4075                    LABEL USING "Z.D";ABS(Numeric_label)
4076                CASE "2"
4077                    LABEL USING "Z.DD";ABS(Numeric_label)
4078                CASE "3"
4079                    LABEL USING "Z.3D";ABS(Numeric_label)
4080                CASE "4"
4081                    LABEL USING "Z.4D";ABS(Numeric_label)
4082                END SELECT
4083            END IF
4084        END SELECT
4085        RETURN
4086        !
4087        ! ////////////////////////////////////////////////////////////
4088        !
4089 Label_format:          !Select the MINIMUM number of digits for label.
4090                        !Then label the graph at the current location.
4091                        !
4092        Numeric_label=DROUND(Numeric_label,8)
4093        IF Digies$<>"F" THEN   ! Not minimum digits
4094            GOSUB Fixed_sig_digit  ! Fill zeros to right of .
4095            RETURN
4096        END IF
4097        SELECT ABS(Numeric_label)
4098        CASE 1.0 TO 9.99
4099            IF INT(Numeric_label)=Numeric_label THEN
4100                LABEL USING "MD";Numeric_label
4101            ELSE
4102                SELECT Numeric_label
4103                CASE PROUND(Numeric_label,-1)
4104                    LABEL USING "MD.D";Numeric_label
4105                CASE PROUND(Numeric_label,-2)
4106                    LABEL USING "MD.DD";Numeric_label
4107                CASE PROUND(Numeric_label,-3)
4108                    LABEL USING "MD.DDD";Numeric_label
4109                CASE ELSE
4110                    LABEL USING "MD.DDESZ";Numeric_label
4111                END SELECT
4112            END IF
4113            !
4114        CASE 9.99 TO 99.99
```

```
4115           IF INT(Numeric_label)=DROUND(Numeric_label,5) THEN
4116               LABEL USING "MDD";Numeric_label
4117           ELSE
4118               SELECT Numeric_label
4119               CASE PROUND(Numeric_label,-1)
4120                   LABEL USING "M2D.D";Numeric_label
4121               CASE PROUND(Numeric_label,-2)
4122                   LABEL USING "M2D.DD";Numeric_label
4123               CASE PROUND(Numeric_label,-3)
4124                   LABEL USING "M2D.DDD";Numeric_label
4125               CASE ELSE
4126                   LABEL USING "MD.DDESZ";Numeric_label
4127               END SELECT
4128           END IF
4129           !
4130       CASE 99.99 TO 999.99
4131           IF INT(Numeric_label)=DROUND(Numeric_label,6) THEN
4132               LABEL USING "MDDD";Numeric_label
4133           ELSE
4134               SELECT Numeric_label
4135               CASE PROUND(Numeric_label,-1)
4136                   LABEL USING "M3D.D";Numeric_label
4137               CASE PROUND(Numeric_label,-2)
4138                   LABEL USING "M3D.DD";Numeric_label
4139               CASE ELSE
4140                   LABEL USING "MD.DDESZ";Numeric_label
4141               END SELECT
4142           END IF
4143           !
4144       CASE 999.99 TO 9999.99
4145           IF INT(Numeric_label)=DROUND(Numeric_label,7) THEN
4146               LABEL USING "M4D";Numeric_label
4147           ELSE
4148               SELECT Numeric_label
4149               CASE PROUND(Numeric_label,-1)
4150                   LABEL USING "M4D.D";Numeric_label
4151               CASE PROUND(Numeric_label,-2)
4152                   LABEL USING "M4D.DD";Numeric_label
4153               CASE ELSE
4154                   LABEL USING "MD.DDESZ";Numeric_label
4155               END SELECT
4156           END IF
4157           !
4158       CASE >9999.99
4159           SELECT ABS(Numeric_label)
4160           CASE <1.0E+10
4161               LABEL USING "MD.DDESZ";Numeric_label
4162           CASE <1.0E+100
4163               LABEL USING "MD.DDESZZ";Numeric_label
4164           CASE ELSE
4165               LABEL USING "MD.DDESZZZ";Numeric_label
4166           END SELECT
4167           !
4168           !++++++All values less that 1.0 ++++++++++
4169           !
4170       CASE .0001 TO 1.0
4171           SELECT Numeric_label
4172           CASE PROUND(Numeric_label,-1)
4173               LABEL USING "MZ.D";Numeric_label
4174           CASE PROUND(Numeric_label,-2)
4175               LABEL USING "MZ.2D";Numeric_label
4176           CASE PROUND(Numeric_label,-3)
4177               LABEL USING "MZ.3D";Numeric_label
4178           CASE PROUND(Numeric_label,-4)
4179               LABEL USING "MZ.4D";Numeric_label
4180           CASE ELSE
4181               LABEL USING "MD.DDESZ";Numeric_label
4182           END SELECT
4183           !
4184       CASE ELSE
4185           IF ABS(Numeric_label)>1.0E-99 THEN
4186               IF PROUND(Numeric_label,-9)=Numeric_label THEN
4187                   LABEL USING "MD.DDESZ";Numeric_label
4188               ELSE
4189                   LABEL USING "MD.DDESZZ";Numeric_label
4190               END IF
4191           ELSE
4192               LABEL USING "D";Numeric_label
4193           END IF
4194       END SELECT
4195       RETURN
4196       !
4197       ! ///////////////////////////////////////////////////////////////
4198       !
4199 Label_x_linear:        !Put numeric labels at every MAJOR tick mark.
4200       Digies$=Sig_digits$[1;1]  ! Select X significant digits
4201       CSIZE DROUND(Viewscale*Xlcsize,3),Aspect_ratio
4202       CLIP OFF
4203       LDIR 0
4204       Tick=Xtics*Txminor  !Divisions for labeling
4205       LORG 6
4206       FOR Numeric_label=Xming TO Xmaxg-.5*Tick STEP Tick
4207           MOVE Numeric_label,Yming-.005*(Ymaxg-Yming)
4208           GOSUB Label_format
4209       NEXT Numeric_label
4210       MOVE Xmaxg,Yming-.005*(Ymaxg-Yming)
4211       LORG 9
4212       Numeric_label=Xmaxg
4213       GOSUB Label_format
4214       PENUP
4215       CLIP ON
4216       RETURN
4217       !
4218       ! ///////////////////////////////////////////////////////////////
4219       !
4220 Label_y_linear:        !Put numeric labels at every MAJOR tick mark.
4221       Digies$=Sig_digits$[2;1]  ! Select Y significant digits
4222       CSIZE DROUND(Viewscale*Ylcsize,3),Aspect_ratio
4223       CLIP OFF
4224       LDIR 0
4225       MOVE Xming-.005*(Xmaxg-Xming),Yming
4226       LORG 7
4227       Numeric_label=Yming
4228       GOSUB Label_format
4229       Tick=Ytics*Tyminor  !Divisions for labeling
4230       LORG 8
4231       FOR Numeric_label=Yming+Tick TO Ymaxg-.5*Tick STEP Tick
4232           MOVE Xming-.005*(Xmaxg-Xming),Numeric_label
4233           GOSUB Label_format
4234       NEXT Numeric_label
4235       MOVE Xming-.005*(Xmaxg-Xming),Ymaxg
4236       LORG 9
4237       Numeric_label=Ymaxg
4238       GOSUB Label_format
4239       PENUP
4240       CLIP ON
4241       RETURN
4242       !
4243       ! ///////////////////////////////////////////////////////////////
4244       !
4245 Label_x_log:           !Put numeric labels at every log cycle and end.
4246                        !If more that 4 log cycles then thin the labels.
```

```
4247                      !where Log_label_limit=4.
4248         Digies$=Sig_digits$[1;1]  ! Select X significant digits
4249         CSIZE DROUND(Viewscale*Xlcsize,3),Aspect_ratio
4250         CLIP OFF
4251         LORG 6
4252         Yloc=Yming-.005*(Ymaxg-Yming)
4253         !Label left corner
4254         Numeric_label=Xbegin
4255         Xloc=FNLin_map_logx(Numeric_label)
4256         MOVE Xloc,Yloc
4257         GOSUB Label_format
4258         !
4259         IF INT(Xcycles)<Log_label_limit THEN
4260             FOR C=1 TO INT(Xcycles)
4261                 Numeric_label=Xbegin*(10^C)
4262                 Xloc=FNLin_map_logx(Numeric_label)
4263                 MOVE Xloc,Yloc
4264                 IF C=INT(Xcycles) AND INT(Xcycles)=Xcycles THEN LORG 9
4265                 GOSUB Label_format
4266             NEXT C
4267         ELSE
4268             Log_label_step=INT(DROUND((Xcycles/(Log_label_limit-1)+.5),1))
4269             IF Log_label_step<1 THEN Log_label_step=1
4270             FOR C=0 TO INT(Xcycles) STEP Log_label_step
4271                 IF C>0 THEN
4272                     Numeric_label=Xbegin*(10^C)
4273                     Xloc=FNLin_map_logx(Numeric_label)
4274                     MOVE Xloc,Yloc
4275                     IF C=INT(Xcycles) AND INT(Xcycles)=Xcycles THEN LORG 9
4276                     GOSUB Label_format
4277                 END IF
4278             NEXT C
4279         END IF
4280         IF INT(Xcycles)<>Xcycles AND INT(Xcycles)<Log_label_limit-1 THEN
4281             IF FRACT(Xcycles)>.31 THEN !Only label end point if >3/10 cycle
4282                 LORG 9
4283                 MOVE Xmaxg,Yloc
4284                 Numeric_label=Xend
4285                 GOSUB Label_format
4286             END IF
4287         END IF
4288         CLIP ON
4289         RETURN
4290         !
4291         ! ////////////////////////////////////////////////////////////
4292         !
4293 Label_y_log:           !Put numeric labels at every log cycle and end.
4294         Digies$=Sig_digits$[2;1]  ! Select Y significant digits
4295         CSIZE DROUND(Viewscale*Ylcsize,3),Aspect_ratio
4296         CLIP OFF
4297         LORG 7
4298         Xloc=Xming-.005*(Xmaxg-Xming)
4299         !Label lower corner
4300         MOVE Xloc,Yming
4301         Numeric_label=Ybegin
4302         GOSUB Label_format
4303         LORG 8
4304         IF INT(Ycycles)<Log_label_limit THEN
4305             FOR C=1 TO INT(Ycycles)
4306                 Numeric_label=Ybegin*(10^C)
4307                 Yloc=FNLin_map_logy(Numeric_label)
4308                 MOVE Xloc,Yloc
4309                 IF C=INT(Ycycles) AND INT(Ycycles)=Ycycles THEN LORG 9
4310                 GOSUB Label_format
4311             NEXT C
4312         ELSE
4313             Log_label_step=INT(DROUND((Ycycles/(Log_label_limit-1)+.5),1))
4314             IF Log_label_step<1 THEN Log_label_step=1
4315             FOR C=0 TO INT(Ycycles) STEP Log_label_step
4316                 IF C>0 THEN
4317                     Numeric_label=Ybegin*(10^C)
4318                     Yloc=FNLin_map_logy(Numeric_label)
4319                     MOVE Xloc,Yloc
4320                     IF C=INT(Ycycles) AND INT(Ycycles)=Ycycles THEN LORG 9
4321                     GOSUB Label_format
4322                 END IF
4323             NEXT C
4324         END IF
4325         IF INT(Ycycles)<>Ycycles AND INT(Ycycles)<Log_label_limit-1 THEN
4326             LORG 9
4327             MOVE Xloc,Ymaxg
4328             Numeric_label=Yend
4329             GOSUB Label_format
4330         END IF
4331         CLIP ON
4332         RETURN
4333         !
4334         ! ////////////////////////////////////////////////////////////
4335         !
4336     SUBEND
4337   !
4338   ! *****************************************************************
4339   !
4340     SUB Find_log_cycles(Low,High,Cycles,New_low,New_high)
4341 Find_log_cycles:! Original: 13 Nov 1984   G. Koepke
4342                 ! Revision: 06 Aug 1987
4343                 ! Determine the number of LOG cycles that will cover
4344                 ! the range of MIN (>0) to MAX for either axis.  There will
4345                 ! be at least ONE cycle and may be stopped at the first
4346                 ! 1/10 cycle above the scale MAX.
4347                 ! The variable 'Cycles' has as the integer part the number
4348                 ! of FULL cycles and as the fractional part the number of
4349                 ! 1/10 cycle divisions beyond the last FULL cycle.
4350                 ! 'New_low' give a lower value with one digit.
4351                 ! 'New_high' gives the new upper limit value.
4352                 !
4353         COM /Bugs/ INTEGER Bug1,Bug2,Bug3,Printer
4354         INTEGER Exponent,M,N
4355         Exponent=0
4356         IF Low<1.0E-50 THEN
4357             DISP " Range error for LOG plot, begin point too small.";
4358             DISP ".. (continue). "
4359             BEEP
4360             PAUSE
4361             DISP CHR$(12)
4362             Low=1.0E-50
4363         END IF
4364         N=0
4365         SELECT Low
4366         CASE <1.0
4367             REPEAT
4368                 N=N+1
4369                 Test=Low*10.0^(N)
4370                 IF Test>=1.0 THEN
4371                     Exponent=-N
4372                     Test=DROUND(Test,12)
4373                     New_low=INT(Test)*10^(Exponent)
4374                 END IF
4375             UNTIL Exponent<>0
4376         CASE >=10.
4377             REPEAT
4378                 N=N+1
```

```
4379              Test=Low*10.0^(-N)
4380              IF Test<10. THEN
4381                  Exponent=N
4382                  Test=DROUND(Test,12)
4383                  New_low=INT(Test)*10^(Exponent)
4384                  IF Bug3 THEN
4385                      PRINT USING "3(MD.16DE)";New_low,Test,INT(Test)
4386                  END IF
4387              END IF
4388          UNTIL Exponent<>0
4389      END SELECT
4390      !
4391      IF Low>=1.0 AND Low<10.0 THEN
4392          New_low=INT(Low)
4393      END IF
4394      New_low=DROUND(New_low,1)
4395      !
4396      !
4397      IF High<=10*New_low THEN
4398          Cycles=1.0
4399      ELSE
4400          N=1
4401          LOOP
4402              Test=DROUND(New_low*10.0^(N),3)
4403              IF Bug3 THEN
4404                  PRINT "New_low=";New_low;" Full Cycles=";N;
4405                  PRINT " Test=";Test;" Max=";High
4406              END IF
4407          EXIT IF High<=Test
4408              N=N+1
4409          END LOOP
4410          IF Test=High THEN Find_range
4411          IF Bug3 THEN PRINT
4412          !
4413          M=0    ! 1/10 Cycles
4414          LOOP   !Find the number of divisions above full cycles - 1
4415                 !necessary to cover the range.
4416                 !
4417              M=M+1
4418              Test=DROUND((M+1)*New_low*10.0^(N-1),3)
4419              IF Bug3 THEN
4420                  PRINT "Cycles=";N-1;"1/10s=";M;" Test=";Test;" Max=";High
4421              END IF
4422          EXIT IF High<=Test
4423          EXIT IF M>=9
4424          END LOOP
4425 Find_range:!
4426          SELECT M
4427          CASE 0,>=9
4428              Cycles=N
4429          CASE <9
4430              Cycles=(N-1)+(M/10)
4431          END SELECT
4432      END IF
4433      New_high=New_low*(10.0*FRACT(Cycles)+1)*10.0^(INT(Cycles))
4434      SUBEXIT
4435  SUBEND
4436  !
4437  ! ***************************************************************
4438  !
4439  DEF FNLin_map_logx(V)
4440      COM /Log_scale/ REAL Xcycles,Xbegin,Ycycles,Ybegin
4441      !
4442      !P = position, in linear units on that axis, of the LGT(V)
4443      !V = value to be mapped on the LOG scale.
4444      !B = begin LOG value at linear 0.0, ie. 0.1, 0.003, 10, etc.
4445      !R = range of linear scale = 100.0
4446      !C = number of cycles nn.m, nn = whole cycles, m = divisions
4447      !    beyond last whole cycle.
4448      !
4449      R=100.0
4450      B=Xbegin
4451      C=Xcycles
4452      IF V<B THEN V=B/10
4453      P=(R/(INT(C)+LGT(10*FRACT(C)+1)))*LGT(V/B)
4454      RETURN P
4455  FNEND
4456  !
4457  ! ***************************************************************
4458  !
4459  DEF FNLin_map_logy(V)
4460      COM /Log_scale/ REAL Xcycles,Xbegin,Ycycles,Ybegin
4461      !
4462      !P = position, in linear units on that axis, of the LGT(V)
4463      !V = value to be mapped on the LOG scale.
4464      !B = begin LOG value at linear 0.0, ie. 0.1, 0.003, 10, etc.
4465      !R = range of linear scale = 100.0
4466      !C = number of cycles nn.m, nn = whole cycles, m = divisions
4467      !    beyond last whole cycle.
4468      !
4469      R=100.0
4470      B=Ybegin
4471      C=Ycycles
4472      IF V<B THEN V=B/10
4473      P=(R/(INT(C)+LGT(10*FRACT(C)+1)))*LGT(V/B)
4474      RETURN P
4475      !
4476  FNEND
4477  !
4478  ! ***************************************************************
4479  !
4480  SUB Hp7475a_setup(INTEGER Suspended)
4481 Hp7475a_setup:! Original: 13 Nov 1984
4482               ! Revision: 06 Aug 1987
4483      !Optimize use of the HP 7475a plotter to draw
4484      !various axes types.  This is the first step to
4485      !draw the background for the graph.
4486      !
4487      OPTION BASE 1
4488      DEG
4489      COM /Background/ Graphtype$,Margins$(*),Papersize$
4490      COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
4491      COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
4492      COM /Background/ Xgrid_tick$,INTEGER Xmajor,Xminor
4493      COM /Background/ Ygrid_tick$,INTEGER Ymajor,Yminor
4494      COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
4495      COM /Hard_space/ REAL Xview_lft,Xview_rt,Yview_btm,Yview_top
4496      COM /Hard_space/ REAL Viewscale,Aspect_ratio
4497      COM /Plot_device/ Plot_lang$,INTEGER Plot_addr
4498      !
4499      INTEGER Hllx,Hlly,Hurx,Hury
4500      DIM Output$[80]
4501      ASSIGN @Plotter TO Plot_addr
4502      !
4503      GOSUB Initial_7475a
4504      IF NOT Suspended THEN GOSUB Select_margin
4505      SUBEXIT
4506      !
4507      ! ////////////////////////////////////////////////////////////
4508      !
4509 Select_margin:      !Set margins and VIEWPORT.
4510      IF Graphtype$="POLAR" THEN   !Must define polar parameters
```

```
4511        !    Margins$(2)="SQUARE"      !through out program.
4512        !END IF
4513        IF Margins$(2)="USER DEFN" THEN
4514            GOSUB User
4515            GOTO Skipmargins
4516        END IF
4517        SELECT Margins$(1)
4518        CASE "HORIZONTAL"
4519            GOSUB Horizon_setup
4520            SELECT Margins$(2)
4521            CASE "BOUND TOP"
4522                GOSUB H_boundtop
4523            CASE "BOUND LEFT"
4524                GOSUB H_boundleft
4525            CASE "FULL"
4526                GOSUB H_full
4527            CASE "SQUARE"
4528                GOSUB H_square
4529            CASE ELSE
4530                GOSUB User
4531            END SELECT
4532        CASE "VERTICAL"
4533            GOSUB Vertical_setup
4534            SELECT Margins$(2)
4535            CASE "BOUND TOP"
4536                GOSUB V_boundtop
4537            CASE "BOUND LEFT"
4538                GOSUB V_boundleft
4539            CASE "FULL"
4540                GOSUB V_full
4541            CASE "SQUARE"
4542                GOSUB V_square
4543            CASE ELSE
4544                GOSUB User
4545            END SELECT
4546        END SELECT
4547 Skipmargins:!
4548        !
4549        !Scale the Viewscale and Aspect_ratio for the lettering.
4550        !
4551        Xview_lft=Left_mar
4552        Xview_rt=Right_mar
4553        Yview_btm=Bottom_mar
4554        Yview_top=Top_mar
4555        SELECT Margins$(1)
4556        CASE "HORIZONTAL"
4557            Aspect_ratio=(Xview_rt-Xview_lft)/((Yview_top-Yview_btm)*2)
4558        CASE "VERTICAL"
4559            Aspect_ratio=(Yview_top-Yview_btm)/((Xview_rt-Xview_lft)*2)
4560        END SELECT
4561        Aspect_ratio=MIN(MAX(Aspect_ratio,.3),1.5)
4562        Viewscale=MIN((Xview_rt-Xview_lft),(Yview_top-Yview_btm))
4563        VIEWPORT Left_mar,Right_mar,Bottom_mar,Top_mar
4564        IF Bug2 THEN
4565            PRINTER IS Printer
4566            PRINT "P1, P2 coordinates are (x,y): ";Hllx,Hlly,Hurx,Hury
4567            PRINT "Viewport (Xl,Xr,Yb,Yt);";
4568            PRINT Left_mar,Right_mar,Bottom_mar,Top_mar
4569            PRINT USING "5/"
4570            PRINTER IS CRT
4571            BEEP
4572            PAUSE
4573        END IF
4574        RETURN
4575        !
4576        ! //////////////////////////////////////////////////////////////
```

```
4577        !
4578 Initial_7475a:!
4579        !Initialize 7475A plotter, and expand P1 & P2 to include maximum
4580        !plotting area.
4581        !
4582        ON TIMEOUT 7,12 GOTO Plotter_dead   !Plotter hangs bus!
4583        OUTPUT @Plotter;"IN"                 !Initialize 7475A plotter.
4584        OUTPUT @Plotter;"PS"&Papersize$      !Select paper size
4585        OUTPUT @Plotter;"VS"&VAL$(Pen_speed) !Set the pen to Pen_speed
4586                                             !and slow acceleration to .2
4587        IF Margins$(1)="VERTICAL" THEN
4588            OUTPUT @Plotter;"RO90;IP;IW"
4589        END IF
4590        IF Margins$(2)="USER DEFN" THEN
4591            OUTPUT @Plotter;"OP"          !Send the P1,P2 coordinates.
4592            ENTER @Plotter;Hllx,Hlly,Hurx,Hury
4593            OUTPUT @Plotter;"PU"
4594            OUTPUT @Plotter;"SP1"
4595 Test_p1_p2:!
4596            Output$="PA"&VAL$(Hllx)&","&VAL$(Hlly)
4597            OUTPUT @Plotter;Output$
4598            BEEP
4599            DISP " MOVE PEN TO DESIRED LOCATION for LOWER LEFT CORNER.";
4600            DISP "  ...press CONTINUE. "
4601            PAUSE
4602            OUTPUT @Plotter;"OA"
4603            ENTER @Plotter;Hllx,Hlly,Pen_status
4604            !
4605            Output$="PA"&VAL$(Hurx)&","&VAL$(Hury)
4606            OUTPUT @Plotter;Output$
4607            BEEP
4608            DISP " MOVE PEN TO DESIRED LOCATION for UPPER RIGHT CORNER.";
4609            DISP "  ...press CONTINUE. "
4610            PAUSE
4611            OUTPUT @Plotter;"OA"
4612            ENTER @Plotter;Hurx,Hury,Pen_status
4613            IF Hurx<=Hllx OR Hury<=Hlly THEN
4614                DISP " BAD LIMITS....TRY AGAIN "
4615                BEEP 400,.3
4616                GOTO Test_p1_p2
4617                WAIT 1.8
4618            END IF
4619            DISP "Generating GRAPH on HP7475A plotter."
4620        ELSE
4621            OUTPUT @Plotter;"OH"          !Send the HARD CLIP limits.
4622            ENTER @Plotter;Hllx,Hlly,Hurx,Hury
4623        END IF
4624        !
4625        SELECT Margins$(2)  ! SELECT QUADRANTS
4626        CASE "LOW LEFT"
4627            Hurx=Hurx-(Hurx-Hllx)/2
4628            Hury=Hury-(Hury-Hlly)/2
4629        CASE "UP RIGHT"
4630            Hllx=Hllx+(Hurx-Hllx)/2
4631            Hlly=Hlly+(Hury-Hlly)/2
4632        CASE "UP LEFT"
4633            Hlly=Hlly+(Hury-Hlly)/2
4634            Hurx=Hurx-(Hurx-Hllx)/2
4635        CASE "LOW RIGHT"
4636            Hllx=Hllx+(Hurx-Hllx)/2
4637            Hury=Hury-(Hury-Hlly)/2
4638        CASE ELSE
4639            ! NO CHANGE
4640        END SELECT
4641        Output$="IP"&VAL$(Hllx)&","&VAL$(Hlly)&","&VAL$(Hurx)&","&VAL$(Hury)
4642        OUTPUT @Plotter;Output$              !Set P1 & P2 to defined limits.
```

```
4643        !
4644        !RESET all graphics operations, read in HARD CLIP limits and
4645        !scale the result to the same dimensions as the screen, which
4646        !is (X longer) 0,100*RATIO,0,100  or  (Y longer) 0,100,0,100/RATIO.
4647        !Both VIEWPORT and WINDOW operations are performed.
4648        !
4649         GINIT
4650         PLOTTER IS Plot_addr,Plot_lang$
4651         PENUP
4652         OFF TIMEOUT 7
4653         RETURN
4654        !+++++++++++++++++++++++++++++++++++
4655 Plotter_dead:      !The bus is hung.
4656         DISP "The plotter is NOT responding!"
4657         BEEP
4658         WAIT 1.8
4659         DISP CHR$(12)
4660         Suspended=1
4661         OFF TIMEOUT 7
4662         RETURN
4663        !
4664        !////////////////////////////////////////////////////////////////
4665        !
4666 User:!This special margins routine will handle the USER DEFN hard
4667        !clip limits.
4668        !The shorter of Hurx-Hllx and Hury-Hlly is scaled to 100 units
4669        !by the PLOTTER IS command.  The longer is scaled to 100*RATIO units.
4670        !
4671         Left_mar=0
4672         Bottom_mar=0
4673         IF Hurx-Hllx>Hury-Hlly THEN
4674             Right_mar=100*RATIO
4675             Top_mar=100
4676         ELSE
4677             Right_mar=100
4678             Top_mar=100/RATIO
4679         END IF
4680         RETURN
4681        !
4682        ! ////////////////////////////////////////////////////////////
4683        !
4684 Horizon_setup:       !Determine the GDUs/cm for the appropriate paper.
4685         IF Papersize$="3" THEN
4686                 ! 11x17 inch paper.
4687             Xlength=100*RATIO
4688             Xgdu_cm=100*RATIO/41.45
4689             Ylength=100
4690             Ygdu_cm=100/25.85
4691         ELSE          ! 8.5x11 inch.
4692             Xlength=100*RATIO
4693             Xgdu_cm=100*RATIO/25.82
4694             Ylength=100
4695             Ygdu_cm=100/19.85
4696         END IF
4697         RETURN
4698    !
4699    !/////////////////////////////////////////////////////////////////////
4700    !
4701 H_boundtop:     !Set margins for top binding.
4702         IF Papersize$="3" THEN
4703                 ! 11x17 inch paper.
4704             Left_mar=Xgdu_cm*1.17
4705             Right_mar=Xlength-Xgdu_cm*2.25
4706             Bottom_mar=Ygdu_cm*1.65
4707             Top_mar=Ylength-Ygdu_cm*2.75
4708                 !
4709         ELSE
4710                 ! 8.5x11 inch paper.
4711             Left_mar=Xgdu_cm*1.6
4712             Right_mar=Xlength-Xgdu_cm*1.4
4713             Bottom_mar=Ygdu_cm*2.25
4714             Top_mar=Ylength-Ygdu_cm*2.39
4715                 !
4716         END IF
4717         RETURN
4718    !
4719    !//////////////////////////////////////////////////////////////////////
4720    !
4721 H_boundleft:   !Set margins for left side binding.
4722         IF Papersize$="3" THEN
4723                 ! 11x17 inch paper.
4724             Left_mar=Xgdu_cm*2.4
4725             Right_mar=Xlength-Xgdu_cm*2.25
4726             Bottom_mar=Ygdu_cm*1.65
4727             Top_mar=Ylength-Ygdu_cm*1.45
4728                 !
4729         ELSE
4730                 ! 8.5x11 inch paper.
4731             Left_mar=Xgdu_cm*2.89
4732             Right_mar=Xlength-Xgdu_cm*1.4
4733             Bottom_mar=Ygdu_cm*2.25
4734             Top_mar=Ylength-Ygdu_cm*1.15
4735                 !
4736         END IF
4737         RETURN
4738    !
4739    !///////////////////////////////////////////////////////////////
4740    !
4741 H_full:    !Set margins to fullest dimensions.
4742         IF Papersize$="3" THEN
4743                 ! 11x17 inch paper.
4744             Left_mar=Xgdu_cm*0.
4745             Right_mar=Xlength-Xgdu_cm*1.22
4746             Bottom_mar=Ygdu_cm*.15
4747             Top_mar=Ylength-Ygdu_cm*0.
4748                 !
4749         ELSE
4750                 ! 8.5x11 inch paper.
4751             Left_mar=Xgdu_cm*.25
4752             Right_mar=Xlength-Xgdu_cm*0.
4753             Bottom_mar=Ygdu_cm*1.12
4754             Top_mar=Ylength-Ygdu_cm*0.
4755                 !
4756         END IF
4757         RETURN
4758    !
4759    !/////////////////////////////////////////////////////////////////
4760    !
4761 H_square:      !Set margins for square centered on paper.
4762         IF Papersize$="3" THEN
4763                 ! 11x17 inch paper.
4764             Left_mar=Xgdu_cm*7.35
4765             Right_mar=Xlength-Xgdu_cm*8.45
4766             Bottom_mar=Ygdu_cm*.15
4767             Top_mar=Ylength-Ygdu_cm*0.
4768                 !
4769         ELSE
4770                 ! 8.5x11 inch paper.
4771             Left_mar=Xgdu_cm*3.65
4772             Right_mar=Xlength-Xgdu_cm*3.4
4773             Bottom_mar=Ygdu_cm*1.12
4774             Top_mar=Ylength-Ygdu_cm*0.
```

```
4775
4776      END IF
4777      RETURN
4778    !
4779    !////////////////////////////////////////////////////////
4780    !
4781 Vertical_setup:         !Setup the vertical dimensions uniformly.
4782      IF Papersize$="3" THEN
4783             ! 11x17 inch paper.
4784          Xlength=100
4785          Xgdu_cm=100/25.85
4786          Ylength=100/RATIO
4787          Ygdu_cm=100/(RATIO*41.45)
4788      ELSE            ! 8.5x17 inch.
4789          Xlength=100
4790          Xgdu_cm=100/19.85
4791          Ylength=100/RATIO
4792          Ygdu_cm=100/(RATIO*25.82)
4793      END IF
4794      RETURN
4795    !
4796    !////////////////////////////////////////////////////////
4797    !
4798      !
4799 V_boundtop:      !Set margins for top binding.
4800      IF Papersize$="3" THEN
4801                ! 11x17 inch paper.
4802          Left_mar=Xgdu_cm*1.60
4803          Right_mar=Xlength-Xgdu_cm*1.45
4804          Bottom_mar=Ygdu_cm*2.30
4805          Top_mar=Ylength-Ygdu_cm*2.40
4806                !
4807      ELSE
4808                ! 8.5x11 inch paper.
4809          Left_mar=Xgdu_cm*1.11
4810          Right_mar=Xlength-Xgdu_cm*2.24
4811          Bottom_mar=Ygdu_cm*1.60
4812          Top_mar=Ylength-Ygdu_cm*2.67
4813                !
4814      END IF
4815      RETURN
4816     !
4817     !////////////////////////////////////////////////////////////
4818     !
4819 V_boundleft:          !Setup margins for left binding.
4820      IF Papersize$="3" THEN
4821                ! 11x17 inch paper.
4822          Left_mar=Xgdu_cm*2.38
4823          Right_mar=Xlength-Xgdu_cm*1.45
4824          Bottom_mar=Ygdu_cm*2.30
4825          Top_mar=Ylength-Ygdu_cm*1.12
4826                !
4827      ELSE
4828                ! 8.5x11 inch paper.
4829          Left_mar=Xgdu_cm*2.37
4830          Right_mar=Xlength-Xgdu_cm*2.24
4831          Bottom_mar=Ygdu_cm*1.60
4832          Top_mar=Ylength-Ygdu_cm*1.40
4833                !
4834      END IF
4835      RETURN
4836     !
4837     !////////////////////////////////////////////////////////////
4838     !
4839 V_full:          !Set margins for fullest dimensions.
4840      IF Papersize$="3" THEN
4841                ! 11x17 inch paper.
4842          Left_mar=Xgdu_cm*.15
4843          Right_mar=Xlength-Xgdu_cm*0.
4844          Bottom_mar=Ygdu_cm*1.15
4845          Top_mar=Ylength-Ygdu_cm*0.
4846                !
4847      ELSE
4848                ! 8.5x11 inch paper.
4849          Left_mar=Xgdu_cm*0.
4850          Right_mar=Xlength-Xgdu_cm*1.07
4851          Bottom_mar=Ygdu_cm*.20
4852          Top_mar=Ylength-Ygdu_cm*0.
4853                !
4854      END IF
4855      RETURN
4856    !
4857    !////////////////////////////////////////////////////////
4858    !
4859 V_square:        !Set margins for square centered on paper.
4860      IF Papersize$="3" THEN
4861                ! 11x17 inch paper.
4862          Left_mar=Xgdu_cm*.15
4863          Right_mar=Xlength-Xgdu_cm*0.
4864          Bottom_mar=Ygdu_cm*8.6
4865          Top_mar=Ylength-Ygdu_cm*7.25
4866                !
4867      ELSE
4868                ! 8.5x11 inch paper.
4869          Left_mar=Xgdu_cm*0.
4870          Right_mar=Xlength-Xgdu_cm*1.07
4871          Bottom_mar=Ygdu_cm*3.67
4872          Top_mar=Ylength-Ygdu_cm*3.39
4873                !
4874      END IF
4875      RETURN
4876      SUBEXIT
4877   SUBEND
4878   !
4879   ! ****************************************************
4880   !
4881   SUB Wipe_clean
4882 Wipe_clean:  ! Original: 13 Nov 1984
4883             ! Revision: 06 Aug 1987
4884             ! Clear the CRT and home the cursor.
4885      PRINTER IS CRT                   !Interact with CRT
4886      CONTROL KBD,1;0                    !PRT ALL off.
4887      CONTROL CRT,4;0                    !DISPLAY FCTNS off.
4888      OUTPUT 2 USING "#,4A";" K T"     !Clear the display, home the cursor
4889      GINIT
4890      GCLEAR
4891      ALPHA ON
4892      GRAPHICS OFF
4893      SUBEXIT
4894   SUBEND
4895   !
4896   !************************************************************
4897   !
4898   SUB Autoscale(File(*),Ax$)
4899 Autoscale: !
4900   !
4901   ! Original: 01 Oct 1985, Eric Vanzura
4902   ! Revision: 13 Aug 1987, 11:00    by G. Koepke
4903   !
4904   ! This subprogram finds the max and min of X and/or Y axis data
4905   ! then selects the graphtype with it's starting and stopping points
4906   ! File(*) contains packed data of one or more curves.
```

```
4907    ! Ax$ = "X" if only autoscale X axis.
4908    ! Ax$ = "Y" if only autoscale Y axis.
4909    ! Ax$ = "XY" for both.
4910    ! IF Ax$ is not "XY" then Graphtype$ must be set to a valid choice.
4911    ! IF Ax$ is "NO" then autoscaling is disabled
4912    !
4913    ! No SUB Programs are CALLed
4914         !
4915         ! Roster(i,1) = Curve number 1,2,3,...,17
4916         ! Roster(i,2) = Start address in File(x,*); = x
4917         ! Roster(i,3) = Datacount for curve i
4918         ! Roster(i,4) = PEN number for this curve
4919         !
4920         ! Symbol$(i)="" or "Y".=> no symbol, connect pts
4921         ! Symbol$(i)="*Y" =>  * symbol, connect pts
4922         ! Symbol$(i)="*N" =>  * symbol, do not connect pts
4923         ! Lbl_addr: x, y, pen, size, LDIR, LORG
4924         !
4925         COM /Labels/ Labels$(*),INTEGER Lbl_count,REAL Lbl_addr(*)
4926         COM /Data_param/ INTEGER Datacount,Filesize,Curvecount,Roster(*)
4927         COM /Data_param/ REAL Sym_size,Symbol$(*),Curve_id$(*)
4928         COM /Data_param/ REAL Xmin_data,Xmax_data
4929         COM /Data_param/ REAL Ymin_data,Ymax_data
4930         COM /Background/ Graphtype$,Margins$(*),Papersize$
4931         COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
4932         COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
4933         COM /Background/ Xgrid_tick$,INTEGER Xmajor,Xminor
4934         COM /Background/ Ygrid_tick$,INTEGER Ymajor,Yminor
4935         COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
4936         !
4937         COM /Axes_labels/ Print_xlabels$,Print_ylabels$
4938         COM /Axes_labels/ Sig_digits$,REAL Xlcsize,Ylcsize
4939         !
4940         COM /Windowspace/ REAL Xmin,Xmid,Xmax,Ymin,Ymid,Ymax!graph edges UDU
4941         COM /Windowspace/ REAL Xleft,Xright,Ybottom,Ytop! paper edges UDUs
4942         COM /Plot_device/ Plot_lang$,INTEGER Plot_addr
4943         COM /Hard_space/ REAL Xview_lft,Xview_rt,Yview_btm,Yview_top
4944         COM /Hard_space/ REAL Viewscale,Aspect_ratio
4945         COM /Frame_onoff/ INTEGER Frame_flag
4946         COM /Clear_space/ REAL Space_lft,Space_rt,Space_btm,Space_top
4947         COM /Bugs/ INTEGER Bug1,Bug2,Bug3,Printer
4948         !
4949         REAL Max,Min            !dummy variables used for both axes
4950         REAL Range,Factor,Base
4951         INTEGER Major,Minor
4952         INTEGER I,J,Axis        ! Axis =1 for X and 2 for Y.
4953         INTEGER Total_data_pts,Number_below
4954         DIM Xscale$[6],Scale$[6]
4955         !
4956         IF Curvecount<1 OR Ax$="NO" THEN
4957             GRAPHICS OFF
4958             BEEP
4959             IF Ax$="NO" THEN
4960                 DISP " Auto scale is disabled ... see BACKGROUND editor."
4961             ELSE
4962                 DISP " NO DATA AVAILABLE FOR AUTO SCALING "
4963             END IF
4964             WAIT 2.0
4965             DISP CHR$(12)
4966             GRAPHICS ON
4967             SUBEXIT
4968         END IF
4969         IF Ax$<>"XY" THEN
4970             SELECT Graphtype$
4971             CASE "LINEAR"
4972                 Xscale$="LINEAR"
4973                 Scale$="LINEAR"
4974             CASE "SEMILOG X"
4975                 Xscale$="LOG"
4976                 Scale$="LINEAR"
4977             CASE "SEMILOG Y"
4978                 Xscale$="LINEAR"
4979                 Scale$="LOG"
4980             CASE "LOG LOG"
4981                 Xscale$="LOG"
4982                 Scale$="LOG"
4983             CASE ELSE
4984                 DISP "Graph type improperly chosen! ";
4985                 DISP " will be set to LINEAR ...continue"
4986                 BEEP
4987                 PAUSE
4988                 DISP CHR$(12)
4989                 Xscale$="LINEAR"
4990                 Scale$="LINEAR"
4991             END SELECT
4992         END IF
4993         !
4994 Selective_scale:!
4995         IF Ax$="X" OR Ax$="Y" OR Ax$="XY" THEN
4996             IF Ax$="X" OR Ax$="XY" THEN
4997                 Axis=1          ! X-axis
4998                 GOSUB Fird_max_min
4999                 Xmin_data=Min
5000                 Xmax_data=Max
5001                 GOSUB Choose_scale! Either log or linear parameters
5002                 Xscale$=Scale$
5003                 GOSUB Set_x_initial
5004             END IF
5005             IF Ax$="Y" OR Ax$="XY" THEN
5006                 Axis=2          ! Y-axis
5007                 GOSUB Find_max_min
5008                 Ymin_data=Min
5009                 Ymax_data=Max
5010                 GOSUB Choose_scale! Either log or linear parameters -> Scale$
5011                 GOSUB Set_y_initial
5012             END IF
5013         ELSE
5014             DISP " ERROR is setting axis selector for autoscale. ";
5015             DISP " will be set to XY ... continue "
5016             BEEP
5017             PAUSE
5018             DISP CHR$(12)
5019             Ax$="XY"
5020             GOTO Selective_scale
5021         END IF
5022         !
5023         ! Choose Graphtype$
5024         !
5025         SELECT Xscale$
5026         CASE "LOG"
5027             SELECT Scale$
5028             CASE "LOG"
5029                 Graphtype$="LOG LOG"
5030             CASE "LINEAR"
5031                 Graphtype$="SEMILOG X"
5032             CASE ELSE
5033                 DISP "Auto select graph type error."
5034                 BEEP
5035                 PAUSE
5036             END SELECT
5037         CASE "LINEAR"
5038             SELECT Scale$
```

```
5039          CASE "LOG"
5040              Graphtype$="SEMILOG Y"
5041          CASE "LINEAR"
5042              Graphtype$="LINEAR"
5043          CASE ELSE
5044              DISP "Auto select graph type error."
5045              BEEP
5046              PAUSE
5047          END SELECT
5048      CASE ELSE
5049          DISP "Auto select graph type error."
5050          BEEP
5051          PAUSE
5052      END SELECT
5053      SUBEXIT
5054      !
5055      ! ////////////////////////////////////////////////////////////
5056      !
5057 Find_max_min: !
5058      Min=File(Roster(1,2),Axis)      ! 1st curve,1st data point
5059      Max=File(Roster(1,2),Axis)
5060      FOR I=1 TO Curvecount
5061          FOR J=Roster(I,2) TO Roster(I,2)+Roster(I,3)-1
5062              Max=MAX(Max,File(J,Axis))
5063              Min=MIN(Min,File(J,Axis))
5064          NEXT J
5065      NEXT I
5066      IF Max<=Min THEN Max=Min
5067      IF ABS(Max-Min)<1.0E-50 THEN
5068          IF ABS(Max)>1.0E-50 THEN
5069              Max=Max+.1*ABS(Max)
5070          ELSE
5071              Max=.1
5072          END IF
5073          IF ABS(Min)>1.0E-50 THEN
5074              Min=Min-.1*ABS(Min)
5075          ELSE
5076              Min=-.1
5077          END IF
5078      END IF
5079      RETURN
5080      !
5081      ! ////////////////////////////////////////////////////////////
5082      !
5083 Choose_scale: !
5084      Log_cutoff=(Max-Min)*.4+Min
5085      Number_below=0
5086      Total_data_pts=0
5087      FOR I=1 TO Curvecount
5088          Total_data_pts=Total_data_pts+Roster(I,3) !'Datacount'
5089          FOR J=Roster(I,2) TO Roster(I,2)+Roster(I,3)-1
5090              IF File(J,Axis)<Log_cutoff THEN
5091                  Number_below=Number_below+1
5092              ELSE
5093                  GOTO Next_crv
5094              END IF
5095          NEXT J
5096 Next_crv: !
5097      NEXT I
5098      IF Number_below/Total_data_pts<.5 OR Min<1.E-25 THEN
5099          Scale$="LINEAR"
5100      ELSE
5101          Scale$="LOG"
5102          RETURN
5103      END IF
5104      !   Find parameters for a linear scale (ideas by Wilber [illegible]
5105      Range=Max-Min
5106      IF Range>1.0E-50 THEN
5107          Base=10^INT(LGT(Range))
5108          SELECT Range
5109          CASE <=2*Base
5110              Factor=Base/5
5111          CASE <=5*Base
5112              Factor=Base/2
5113          CASE <=10*Base
5114              Factor=Base
5115          END SELECT
5116      ELSE
5117          Factor=1
5118      END IF
5119      Min=Factor*INT(Min/Factor)
5120      Max=Factor*(INT(Max/Factor)+(Max/Factor<>INT(Max/Factor)))
5121      Major=INT((Max-Min)/Factor)
5122      Major=Major+(INT((Max-Min)/Factor)<>DROUND((Max-Min)/Factor,4))
5123      IF (INT(Major/2)=Major/2) THEN
5124          Major=Major/2
5125          Minor=2
5126      ELSE
5127          Minor=1
5128      END IF
5129      RETURN
5130      !
5131      ! ////////////////////////////////////////////////////////////
5132      !
5133 Set_x_initial: !
5134      Xmajor=Major
5135      Xminor=Minor
5136      Xmin_graph=Min
5137      Xmax_graph=Max
5138      IF Bug2 THEN
5139          PRINTER IS Printer
5140          PRINT "XMIN_GRAPH=";Xmin_graph,"XMAX_GRAPH=";Xmax_graph
5141          PRINT
5142          PRINTER IS CRT
5143      END IF
5144      Y_cross_x=Xmin_graph
5145      RETURN
5146      !
5147      ! ////////////////////////////////////////////////////////////
5148      !
5149 Set_y_initial: !
5150      Ymajor=Major
5151      Yminor=Minor
5152      Ymin_graph=Min
5153      Ymax_graph=Max
5154      IF Bug2 THEN
5155          PRINTER IS Printer
5156          PRINT "YMIN_GRAPH=";Ymin_graph,"YMAX_GRAPH=";Ymax_graph
5157          PRINT
5158          PRINTER IS CRT
5159      END IF
5160      X_cross_y=Ymin_graph
5161      RETURN
5162  SUBEND
5163  !
5164  ! ******************************************************
5165  !
5166  SUB Refresh_labels
5167      ! Original: 18 Aug 1987, G. Koepke (drawn from Graph Data)
5168      ! Revision: 18 Aug 1987
5169      ! Use this routine after Autoscale and before making [illegible]
[illegible]
```

```
5171        !
5172        OPTION BASE 1
5173        DEG
5174        COM /Sys/ Sys_id$[10]
5175        COM /Labels/ Labels$(*),INTEGER Lbl_count,REAL Lbl_addr(*)
5176        COM /Data_param/ INTEGER Datacount,Filesize,Curvecount,Roster(*)
5177        COM /Data_param/ Real_sym_size,Symbol$(*),Curve_id$(*)
5178        COM /Data_param/ REAL Xmin_data,Xmax_data
5179        COM /Data_param/ REAL Ymin_data,Ymax_data
5180        COM /Background/ Graphtype$,Margins$(*),Papersize$
5181        COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
5182        COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
5183        COM /Background/ Xgrid_tick$,INTEGER Xmajor,Xminor
5184        COM /Background/ Ygrid_tick$,INTEGER Ymajor,Yminor
5185        COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
5186        !
5187        COM /Windowspace/ REAL Xmin,Xmid,Xmax,Ymin,Ymid,Ymax!graph edges UDU
5188        COM /Windowspace/ REAL Xleft,Xright,Ybottom,Ytop! paper edges UDUs
5189        IF Lbl_count>0 THEN
5190            ALLOCATE Lbl_ratio(Lbl_count,2)
5191            GOSUB Update_lbl_addr    ! includes Find_mid_pt
5192            DEALLOCATE Lbl_ratio(*)
5193        ELSE
5194            GOSUB Find_mid_pt
5195        END IF
5196        SUBEXIT
5197        !
5198        ! ///////////////////////////////////////////////////////////////
5199        !
5200 Update_lbl_addr:! Make sure that the label addresses are current.
5201        FOR I=1 TO Lbl_count
5202            Lbl_ratio(I,1)=(Lbl_addr(I,1)-Xmin)/(Xmax-Xmin)
5203            Lbl_ratio(I,2)=(Lbl_addr(I,2)-Ymin)/(Ymax-Ymin)
5204        NEXT I
5205        GOSUB Find_mid_pt               ! define new max and min
5206        FOR I=1 TO Lbl_count
5207            Lbl_addr(I,1)=Lbl_ratio(I,1)*(Xmax-Xmin)+Xmin
5208            Lbl_addr(I,2)=Lbl_ratio(I,2)*(Ymax-Ymin)+Ymin
5209        NEXT I
5210        RETURN
5211        !
5212        ! ///////////////////////////////////////////////////////////////
5213        !
5214 Find_mid_pt:       !Interpret the graph type and the scaling done.
5215        SELECT Graphtype$
5216        CASE "LINEAR"
5217            Xmin=Xmin_graph
5218            Xmax=Xmax_graph
5219            Xmid=((Xmax-Xmin)/2)+Xmin
5220            Ymin=Ymin_graph
5221            Ymax=Ymax_graph
5222            Ymid=((Ymax-Ymin)/2)+Ymin
5223        CASE "SEMILOG X"
5224            Xmin=0
5225            Xmax=100
5226            Xmid=50
5227            Ymin=Ymin_graph
5228            Ymax=Ymax_graph
5229            Ymid=((Ymax-Ymin)/2)+Ymin
5230        CASE "SEMILOG Y"
5231            Xmin=Xmin_graph
5232            Xmax=Xmax_graph
5233            Xmid=((Xmax-Xmin)/2)+Xmin
5234            Ymin=0
5235            Ymax=100
5236            Ymid=50
5237        CASE "LOG LOG"
5238            Xmin=0
5239            Xmax=100
5240            Xmid=50
5241            Ymin=0
5242            Ymax=100
5243            Ymid=50
5244        CASE "POLAR" ..
5245            !
5246            BEEP
5247            DISP "POLAR PARAMETERS ARE NOT YET IMPLEMENTED!!!!"
5248            PAUSE
5249        END SELECT
5250        RETURN
5251        !
5252        ! ///////////////////////////////////////////////////////////////
5253        !
5254   SUBEND
5255 !
5256  ! ***************************************************************
5257 !
5258   SUB Optimizedisplay
5259  !This subroutine makes the network analyzer display look nice.
5260        OPTION BASE 1
5261        COM /Nwa_data/ Start_freq,Stop_freq,Marker(5,3)
5262 Optimizedisplay: !
5263        Set_nwa("MARK=4","MARK=MIN")
5264        Read_nwa("MARK=4")
5265        Min=Marker(4,2)
5266        Set_nwa("MARK=5","MARK=MAX")
5267        Read_nwa("MARK=5")
5268        Max=Marker(5,2)
5269        GOSUB Select_scale
5270        Set_nwa("SCALE="&VAL$(Scale))
5271        GOSUB Select_ref_val
5272        Set_nwa("REF_POS=10","REF VAL="&VAL$(Ref_val))
5273        SUBEXIT
5274 !
5275  ! //////////////////////////////////////////
5276 !
5277 Select_scale: !
5278        SELECT Max-Min
5279        CASE <.5
5280            Scale=.05
5281        CASE <1
5282            Scale=.1
5283        CASE <2
5284            Scale=.2
5285        CASE <5
5286            Scale=.5
5287        CASE <10
5288            Scale=1
5289        CASE <20
5290            Scale=2
5291        CASE <50
5292            Scale=5
5293        CASE <100
5294            Scale=10
5295        CASE ELSE
5296            Scale=10
5297            BEEP
5298            PRINT "Proper scale was not found."
5299        END SELECT
5300        RETURN
5301  ! ////////////////////////////////////////////////////////////
5302 Select_ref_val: !
```

```
5303        Ref_val=PROUND(((Max+Min)/2)+(Scale*5),-2)
5304        RETURN
5305   SUBEND
5306 !
5307  ! ************************************************************
5308 !
5309   CSUB Mrqmin(X(*),Y(*),Sig(*),INTEGER Ndata(*),REAL A(*),INTEGER Ma(*),Lis
  => ta(*),Mfit(*),REAL Covar(*),Alpha(*),INTEGER Nca(*),REAL Chisq,Alamda,Sss,D
  => ebug)
5310   SUB Load_sparms
5311        OPTION BASE 1
5312        COM /History/ Status$[1],Time_orgn$[8],Date_orgn$[11]
5313        COM /History/ Time_chng$[8],Date_chng$[11],Description$[160]
5314        COM /Files/ Diskdrive$[20],Filename$[10]
5315        COM /Ana_data/ INTEGER Prog_id,Sweep_type,Sweep_mode,Datacount
5316        COM /Ana_data/ INTEGER S11_valid,S21_valid,S12_valid,S22_valid,Ports
5317        COM /Ana_data/ REAL Z0,Start,Stop
5318        COM /Ana_data/ Title$[55],Company$[30],Operator_name$[30],Measure_tim
  => e$[30]
5319        COM /S_array/ INTEGER Dcount,Svalid,REAL S(801,5,2)
5320 ! Sweep_type: 1=start/stop , 2=center/span
5321 ! Sweep_mode: 1=ramp , 2=step , 3=frequency list
5322        DIM Prompt$[30]
5323        INTEGER I,Num_sparms
5324 Load_sparms:!
5325        Prompt$="8510 S-PARAMETER DATA"
5326 !      CALL Select_disk(Prompt$)
5327        CALL Enterfilename("CAT",Prompt$)
5328        IF Filename$="" THEN
5329            CAT Diskdrive$
5330            CALL Enterfilename("ABORT",Prompt$)
5331            IF Filename$="" THEN
5332                BEEP
5333                DISP "PROGRAM ABORTED"
5334                STOP
5335            END IF
5336        END IF
5337        DISP CHR$(129);" Reading S-parameter data: ";CHR$(128);Filename$;Disk
  => drive$
5338        ON ERROR CALL Errortrap
5339        ASSIGN @Infile TO Filename$&Diskdrive$;FORMAT OFF
5340        OFF ERROR
5341        ON ERROR GOSUB Read_sparm_err
5342        ENTER @Infile;Prog_id
5343        ENTER @Infile;Title$
5344        ENTER @Infile;Company$
5345        ENTER @Infile;User_name$
5346        ENTER @Infile;Measure_time$
5347        ENTER @Infile;Z0
5348        ENTER @Infile;Sweep_type
5349        ENTER @Infile;Sweep_mode
5350        ENTER @Infile;Datacount,Start,Stop,S11_valid,S21_valid,S12_valid,S22_
  => valid,Ports
5351        Dcount=Datacount
5352        Num_sparms=S11_valid+S21_valid+S12_valid+S22_valid
5353        Svalid=0
5354        IF S11_valid THEN Svalid=Svalid+1
5355        IF S21_valid THEN Svalid=Svalid+2
5356        IF S12_valid THEN Svalid=Svalid+4
5357        IF S22_valid THEN Svalid=Svalid+8
5358        ALLOCATE REAL Local_array(Datacount,Num_sparms+1,2)
5359        ENTER @Infile;Local_array(*)
5360        OFF ERROR
5361        ASSIGN @Infile TO *
5362        Description$=Title$
5363        Time_orgn$=Measure_time$[15,22]
5364        Date_orgn$=Measure_time$[1,11]
5365        MAT S(1:Dcount,5,1)= Local_array(1:Dcount,5,1)
5366        FOR I=1 TO 4
5367            IF BIT(Svalid,I-1) THEN MAT S(1:Dcount,I,*)= Local_array(*,I,*)
5368        NEXT I
5369        DEALLOCATE Local_array(*)
5370        SUBEXIT !Load_sparms
5371!
5372! ///////////////////////////////////////////////////////////////
5373!
5374 Read_sparm_err:!
5375        BEEP
5376        PRINT "Error number:";ERRN;ERRM$
5377        DISP "something is wrong with S-parameter read"
5378        PAUSE
5379        RETURN
5380!
5381! ///////////////////////////////////////////////////////////////
5382        GCLEAR
5383   SUBEND   !Load_sparms
5384 !
5385 ! ************************************************************
5386 !
5387   SUB Select_disk(OPTIONAL Prompt$)
5388 Select_disk: ! Original: 04 Jul 1987
5389              ! Revision: 06 Aug 1987
5390        COM /Files/ Diskdrive$[20],Filename$[10]
5391        INTEGER Local_prty,Dd,Pt,Choose(1)
5392        DIM Disc$(30)[60],Title$[40],Displ$[80]
5393        Local_prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
5394        OFF KEY
5395        !
5396        ! Define the disk drives available for this system, reserve the
5397        ! first characters for the drive address and the characters after
5398        ! the - for a description of the drive.
5399        !
5400        ! Example:
5401        ! Disc$(1)=":,700,0,0      HP 9133H HARD disk, volume 0."
5402        !
5403        !
5404        IF NPAR THEN
5405            Displ$=" SELECT DISK DRIVE "&Prompt$&" ... Abort will cancel. "
5406            Displ$=Displ$[1,80]
5407        ELSE
5408            Displ$=" SELECT DISK DRIVE  ... Abort will cancel. "
5409        END IF
5410        Title$=" Available disk drives for this system. "
5411        Pt=1   ! allow only one select
5412        !
5413        IF Diskdrive$[1,1]<>":" THEN Diskdrive$=""
5414        IF Msi_id$[1,1]<>":" THEN Msi_id$=SYSTEM$("MSI")
5415        Diskdrive$=TRIM$(Diskdrive$)
5416        Msi_id$=TRIM$(Msi_id$)
5417        IF LEN(Diskdrive$)>0 AND LEN(Msi_id$)>0 THEN
5418            Disc$(1)=Diskdrive$&RPT$(" ",17-LEN(Diskdrive$))
5419            Disc$(1)=Disc$(1)&"- Last selected disk drive."
5420            Dd=1
5421            IF Diskdrive$<>Msi_id$ THEN
5422                Disc$(2)=Msi_id$&RPT$(" ",17-LEN(Msi_id$))
5423                Disc$(2)=Disc$(2)&"- Start-up mass storage unit specifier."
5424                Dd=Dd+1
5425            ELSE
5426                Disc$(1)=Disc$(1)&"  Start-up MSUS."
5427            END IF
5428        ELSE
5429            IF LEN(Msi_id$)>0 THEN
```

```
5430              Disc$(1)=Msi_id$&RPT$(" ",17-LEN(Msi_id$))
5431              Disc$(1)=Disc$(1)&"- Start-up mass storage unit specifier."
5432              Dd=1
5433          ELSE
5434              Dd=0
5435          END IF
5436      END IF
5437 Disk: !
5438      ! ................ customize system drives here ..................
5439      ! Follow format with - after unit specifier, description is
5440      ! optional but recommended.
5441      ! ...............................................................
5442      !
5443      Disc$(Dd+1)=":,707,0          - HP 9122 dual microfloppy left drive"
5444      Disc$(Dd+2)=":,707,1          - HP 9122 dual microfloppy right drive"
5445      Disc$(Dd+3)=":,1404,0         - HP 9127 single 5.25 floppy drive"
5446      Disc$(Dd+4)=":,1400,1         - HP 9153B single microfloppy"
5447      Disc$(Dd+5)=":,1400,0,2       - HP 9153H hard disk ANA volume"
5448      Disc$(Dd+6)=":,1400,0,3       - HP 9153B hard disk PROG volume"
5449      !
5450      Dd=Dd+6   ! add the number of drive specifiers above
5451      !
5452      IF Sys_id$[1,4]<>"S300" THEN
5453          Disc$(Dd+1)=":,4,1            - LEFT internal series 200"
5454          Disc$(Dd+2)=":,4,0            - RIGHT internal series 200"
5455          Dd=Dd+2
5456      END IF
5457      ! ...............................................................
5458      !
5459      CALL Menu_scroll(Displ$,Title$,Disc$(*),Dd,Pt,Choose(*))
5460      IF Pt=0 THEN
5461          Diskdrive$="NO DISK"
5462      ELSE
5463          Dd=POS(Disc$(Choose(Pt)),"-")-1 ! find -
5464          IF Dd>5 THEN  !  valid msus
5465              Diskdrive$=TRIM$(Disc$(Choose(Pt))[1,Dd])
5466          ELSE
5467              DISP " ERROR in reading MSUS from string, - chr not found. "
5468              BEEP
5469              PAUSE
5470              Diskdrive$="NO DISK"
5471          END IF
5472      END IF
5473 Diskselected:OFF KEY
5474      SUBEXIT
5475  SUBEND
5476  !
5477  ! ***************************************************
5478  !
5479  SUB Enterfilename(Ac$,OPTIONAL Prompt$)
5480 Enterfilename:      !
5481      COM /Files/ Diskdrive$,Filename$
5482      INTEGER I,Ascii_num
5483      DIM Test$[160]
5484      SELECT NPAR
5485      CASE 1
5486          DISP " ENTER the FILE NAME ";
5487      CASE 2
5488          DISP " ENTER the FILE NAME for ";Prompt$;
5489      END SELECT
5490      SELECT Ac$
5491      CASE "CAT"
5492          DISP " ... (ENTER alone to CAT) ";
5493      CASE "ABORT"
5494          DISP " ... (ENTER alone to ABORT) ";
5495      CASE "VALID"
```

```
5496      END SELECT
5497      LINPUT Test$
5498      Test$=TRIM$(Test$)
5499      IF LEN(Test$)=0 THEN
5500          SELECT Ac$
5501          CASE "VALID"
5502              DISP "You MUST enter the FILE NAME now."
5503              BEEP
5504              WAIT 1.8
5505              GOTO Enterfilename
5506          CASE "ABORT","CAT"
5507              GOTO Abortline
5508          CASE ELSE
5509              DISP "Ac$=";Ac$;" in SUB Enterfilename"
5510              BEEP
5511              PAUSE
5512          END SELECT
5513      END IF
5514      IF LEN(Test$)>10 THEN
5515          BEEP
5516          DISP "ERROR in NAME ENTRY--up to 10 chars, you have ";
5517          DISP LEN(Test$);" "
5518          WAIT 1.8
5519          OUTPUT 2 USING "#,K,K";" #";Test$
5520          GOTO Enterfilename
5521      END IF
5522      Filename$=Test$
5523      FOR I=1 TO LEN(Filename$)
5524          Ascii_num=NUM(Filename$[I])
5525          SELECT Ascii num
5526          CASE 65 TO 90,95,97 TO 122,48 TO 57
5527              !Allowed characters
5528          CASE ELSE
5529              BEEP
5530              DISP "ERROR in NAME ENTRY--ILLEGAL CHARACTERS, TRY AGAIN."
5531              WAIT 1.8
5532              OUTPUT 2 USING "#,K,K";" #";Filename$
5533              GOTO Enterfilename
5534          END SELECT
5535      NEXT I
5536      SUBEXIT
5537 Abortline:Filename$=""
5538      SUBEXIT
5539  SUBEND
5540  !
5541  ! ***************************************************************
5542  !
5543  SUB Errortrap
5544 Errortrap:  ! Original: 13 Nov 1984
5545             ! Revision: 02 Dec 1987
5546      ! Trap most errors here
5547      COM /Files/ Diskdrive$[20],Filename$[10]
5548      DIM File$[20],Test$[160],What$[20],Ac$[5]
5549      SELECT ERRN
5550      CASE 54
5551          DISP "DUPLICATE FILE NAME:  ";Filename$;
5552          DISP "....PURGE old one? (Y/N)";
5553          LINPUT What$
5554          What$=TRIM$(What$)
5555          SELECT What$[1,1]
5556          CASE "Y","y"
5557              PURGE Filename$&Diskdrive$
5558          CASE ELSE
5559              Ac$="VALID"
5560              CALL Enterfilename(Ac$)
5561          END SELECT
```

```
5562        CASE 52,53
5563            DISP "Improper FILE NAME --- ENTER NEW FILE NAME";
5564            OUTPUT 2 USING "#,K,K";" #";Filename$
5565            LINPUT Filename$
5566            Filename$=TRIM$(Filename$)
5567        CASE 56
5568            DISP "FILE:  ";Filename$;" is not on this disk, please insert";
5569            DISP " correct disk"
5570            CALL Pause_key_on
5571        CASE 64
5572            DISP "This disk is full, PLEASE insert clean disk"
5573            CALL Pause_key_on
5574        CASE 56
5575            DISP "DATA INPUT disk must be in drive!!  ";
5576            DISP "...CONTINUE when ready."
5577            CALL Pause_key_on
5578        CASE 72,73,76
5579            DISP Diskdrive$;
5580            DISP " is not available, type correct";
5581            DISP " unit specifier (ie. ':,707,0').";
5582            OUTPUT 2 USING "K,#";Diskdrive$
5583            LINPUT Diskdrive$
5584        CASE 80
5585            DISP "CHECK DISK drive door!"
5586            CALL Pause_key_on
5587        CASE ELSE
5588            DISP ERRM$;" 'CONTINUE' when fixed"
5589            CALL Pause_key_on
5590        END SELECT
5591        DISP CHR$(12)
5592        SUBEXIT
5593    SUBEND
5594    !
5595    ! ****************************************************************
5596    !
5597    SUB Menu_scroll(D$,I$,Items$(*),INTEGER Item_cnt,To_select,Choose(*))
5598 Menu_scroll: ! Original: 22 Jun 1987, Galen Koepke, NBS 723.04
5599            ! Revision: 30 Jun 1987, 13:35
5600            !
5601            ! A general purpose menu utility for scrolling items and
5602            ! selecting a given number of them.
5603            ! The items are arranged in screens of 15 items each and
5604            ! the user may access screens via softkeys. There may be
5605            ! up to 10 screens or 150 items to choose from.
5606            ! Items$(*) contains the item descriptions
5607            ! Item_cnt is the number of items in Items$(*)
5608            ! Choose(*) is dimensioned to the number of required choices
5609            !           and will be filled with the item numbers chosen.
5610            ! To_select is the number of required choices.
5611            !
5612        OPTION BASE 1
5613        PRINTER IS CRT
5614        DEG
5615        GOSUB Def_variables
5616        GOSUB Define_screens
5617        GOSUB Make_selections
5618        IF Null_file THEN  ! reset to zero
5619            Item_cnt=0
5620            Items$(1)=""
5621            To_select=0    ! no valid selections
5622        END IF
5623        SUBEXIT
5624        !
5625        ! ////////////////////////////////////////////////////////
5626        !
5627 Def_variables:!
5628        !
5629        INTEGER Screen_cnt,Items_per_scn,First_item(10),Last_item(10)
5630        INTEGER I,J,K,First_line,Last_line,Active_screen,Pointer,Last_pt
5631        INTEGER Local_prty,Skips,Knobcount,Pointeractive,K0,Null_file
5632        INTEGER Exit_flag
5633        DIM Marker$[8],Test$[160]
5634        !
5635        ! initialize parameters
5636        !
5637        Local_prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
5638        IF Local_prty<1 THEN Local_prty=10
5639        IF LEN(Sys_id$)=0 THEN Sys_id$=SYSTEM$("SYSTEM ID")
5640        IF Item_cnt<1 THEN
5641            Null_file=1
5642            Item_cnt=1
5643            To_select=0
5644            Items$(1)="*** Empty ***"
5645        ELSE
5646            Null_file=0
5647        END IF
5648        IF To_select>Item_cnt THEN To_select=Item_cnt
5649        Skips=0
5650        Knobcount=0
5651        Doneflag=0
5652        Marker$="===>"&RPT$(CHR$(8),4)
5653        RETURN
5654        !
5655        ! ////////////////////////////////////////////////////////
5656        !
5657 Define_screens:! Set up screens of 15 items each.
5658                 !
5659        Items_per_scn=15    ! Maximum number of displayable items
5660        IF INT(Item_cnt/Items_per_scn)=Item_cnt/Items_per_scn THEN
5661            Screen_cnt=INT(Item_cnt/Items_per_scn)
5662        ELSE
5663            Screen_cnt=INT(Item_cnt/Items_per_scn)+1
5664        END IF
5665        J=1
5666        FOR I=1 TO Screen_cnt   ! set up each screen
5667            First_item(I)=J
5668            IF J+Items_per_scn-1<Item_cnt THEN
5669                Last_item(I)=J+Items_per_scn-1
5670                J=J+Items_per_scn
5671            ELSE
5672                Last_item(I)=Item_cnt
5673            END IF
5674        NEXT I
5675        RETURN
5676        !
5677        ! ////////////////////////////////////////////////////////
5678        !
5679 Make_selections:! MENU setup and use.
5680        Active_screen=1   ! first screen is active
5681        First_line=2      ! first printed line on screen = 2 or greater.
5682        GOSUB Write_screen  ! activate screen at Active_screen
5683                            ! and set First_line and Last_line for Pointer
5684                            ! write Marker$ to first non-selected line.
5685        K0=0                ! Keys start at zero
5686        Exit_flag=0         ! allow ENTER key to exit when selections filled.
5687 Key_loop: !
5688        DISP D$
5689        ON KBD,Local_prty GOSUB Process_kbd
5690        ON KNOB .01,Local_prty GOSUB Move_pointer
5691        IF Skips<To_select THEN
5692            IF To_select>1 THEN
5693                Test$=" Select  "&VAL$(Skips+1)&" of "&VAL$(To_select)
```

```
5694         ELSE
5695             Test$=" Select"
5696         END IF
5697         ON KEY K0 LABEL Test$,Local_prty GOSUB Select_item
5698     ELSE
5699         ON KEY K0 LABEL " Exit",Local_prty GOTO Exit_line
5700     END IF
5701     IF Active_screen<Screen_cnt THEN
5702         ON KEY K0+1 LABEL " Next   Screen",Local_prty GOSUB Next_screen
5703     ELSE
5704         OFF KEY K0+1
5705     END IF
5706     IF Active_screen>1 THEN
5707         ON KEY K0+2 LABEL " Last   Screen",Local_prty GOSUB Last_screen
5708     ELSE
5709         OFF KEY K0+2
5710     END IF
5711     IF Skips>0 THEN
5712         ON KEY K0+3 LABEL " Select  Reset",Local_prty GOSUB Select_reset
5713     ELSE
5714         OFF KEY K0+3
5715     END IF
5716     ON KEY K0+4 LABEL " ABORT ",Local_prty GOTO Escape_line
5717     IF Exit_flag THEN Exit_line
5718     GOTO Key_loop
5719 Escape_line:Skips=0
5720     MAT Choose= (0)
5721     To_select=0
5722 Exit_line:OFF KEY
5723     OFF KNOB
5724     OFF KBD
5725     OUTPUT KBD;CHR$(255)&CHR$(75);
5726     PRINT CHR$(128);
5727     ! everything cleared, now go back to work.
5728     RETURN
5729     !
5730     ! //////////////////////////////////////////////////
5731     !
5732 Next_screen:     !
5733     OFF KBD
5734     OFF KNOB
5735     OFF KEY
5736     IF Active_screen=Screen_cnt THEN RETURN
5737     Active_screen=Active_screen+1
5738     GOSUB Write_screen
5739     RETURN
5740     !
5741     ! //////////////////////////////////////////////////
5742     !
5743 Last_screen:     !
5744     OFF KBD
5745     OFF KNOB
5746     OFF KEY
5747     IF Active_screen=1 THEN RETURN
5748     Active_screen=Active_screen-1
5749     GOSUB Write_screen
5750     RETURN
5751     !
5752     ! //////////////////////////////////////////////////
5753     !
5754 Select_item:!
5755     OFF KBD
5756     OFF KNOB
5757     OFF KEY
5758     IF NOT Pointeractive THEN
5759         DISP "NO additional selections for this screen."
5760         BEEP
5761         WAIT 2
5762         DISP CHR$(12);
5763         RETURN
5764     END IF
5765     IF Skips=To_select THEN
5766         IF To_select=0 THEN
5767             DISP "This menu is for information only,";
5768             DISP " no selection allowed."
5769         ELSE
5770             DISP "All selections have been filled,";
5771             DISP " 'Select Reset' to repeat."
5772         END IF
5773         BEEP
5774         WAIT 2
5775         DISP CHR$(12);
5776         RETURN
5777     END IF
5778     Skips=Skips+1
5779     Choose(Skips)=First_item(Active_screen)+Pointer-First_line
5780     PRINT CHR$(129);  ! inverse video
5781     PRINT TABXY(10,Pointer);Items$(Choose(Skips))
5782     PRINT CHR$(128);
5783     PRINT TABXY(1,Pointer);
5784     SELECT Pointer
5785     CASE First_line
5786         GOSUB Point_forward
5787     CASE Last_line
5788         GOSUB Point_backward
5789     CASE ELSE
5790         ! move forward unless it requires wrapping to beginning.
5791         IF Skips-1>0 THEN  ! check for selected items.
5792             I=Pointer-First_line
5793             LOOP
5794                 K=0
5795                 FOR J=1 TO Skips
5796                     IF First_item(Active_screen)+I=Choose(J) THEN K=1
5797                 NEXT J
5798             EXIT IF K=0
5799                 I=I+1
5800                 IF I+First_line>Last_line THEN K=-1
5801             EXIT IF K=-1
5802             END LOOP
5803             IF K=0 THEN
5804                 GOSUB Point_forward
5805             ELSE
5806                 GOSUB Point_backward
5807             END IF
5808         ELSE
5809             GOSUB Point_forward
5810         END IF
5811     END SELECT
5812     RETURN
5813     !
5814     ! //////////////////////////////////////////////////
5815     !
5816 Select_reset:     !Clear Choose file
5817     OFF KBD
5818     OFF KNOB
5819     OFF KEY
5820     Skips=0
5821     MAT Choose= (0)
5822     GOSUB Write_screen
5823     RETURN
5824     !
5825     ! //////////////////////////////////////////////////
```

```
5826       !
5827 Process_kbd:! Allow use of arrows and enter key in addition to soft.
5828       Test$=KBD$
5829       IF LEN(Test$)=1 AND Test$[1,1]<>CHR$(32) THEN
5830           BEEP 80.,.1
5831           RETURN
5832       END IF
5833       IF Test$[1,1]=CHR$(32) THEN GOSUB Point_forward
5834       IF Test$[1,1]<>CHR$(255) THEN RETURN
5835       SELECT Test$[2,2]
5836       CASE CHR$(255)
5837                  ! do nothing
5838       CASE "V","T"
5839           GOSUB Point_forward
5840       CASE "^","W"
5841           GOSUB Point_backward
5842       CASE "E"
5843           IF Skips<To_select THE
5844               GOSUB Select_item
5845           ELSE
5846               ! exit routine
5847               Exit_flag=1
5848           END IF
5849       CASE ELSE
5850           BEEP 80.,.1
5851       END SELECT
5852       Test$=""
5853       RETURN
5854       !
5855       ! //////////////////////////////////////////////////
5856       !
5857 Point_forward:Knobcount=5
5858       GOSUB Move_pointer
5859       RETURN
5860 Point_backward:Knobcount=-5
5861       GOSUB Move_pointer
5862       RETURN
5863       !
5864       ! //////////////////////////////////////////////////
5865       !
5866 Jog_pointer:! Move the selection pointer on the active screen.
5867             ! without regard to selected values
5868       IF Knobcount>0 THEN  ! Move forward
5869           Pointer=Pointer+1
5870       ELSE                ! Move backward
5871           Pointer=Pointer-1
5872       END IF
5873       IF Pointer<First_line THEN Pointer=Last_line
5874       IF Pointer>Last_line THEN Pointer=First_line
5875       RETURN
5876       !
5877       ! /////////////////////////////////////////////////////////
5878       !
5879 Move_pointer:! Control pointer to avoid re-selection of items
5880       IF NOT Pointeractive THEN RETURN  ! No selecticns to be made.
5881       Knobcount=Knobcount+KNOBX+KNOBY
5882       IF ABS(Knobcount)<4 THEN RETURN
5883       Last_pt=Pointer
5884       GOSUB Jog_pointer
5885       IF Skips>0 THEN
5886           LOOP
5887               J=Pointer-First_line
5888               FOR I=1 TO Skips
5889                   IF First_item(Active_screen)+J=Choose(I) THEN J=999
5890               NEXT I
5891               IF J=999 AND Pointer=Last_pt THEN Pointeractive=0
5892           EXIT IF Pointeractive=0
5893               IF J=999 THEN GOSUB Jog_pointer
5894           EXIT IF J<>999
5895           END LOOP
5896       END IF
5897       Knobcount=0
5898       OUTPUT KBD;CHR$(255)&CHR$(84);  ! Bring screen home
5899       IF Last_pt=Last_line THEN PRINT CHR$(132);
5900       PRINT "    ";
5901       IF Pointeractive THEN ! Pointer active
5902           IF Pointer=Last_line THEN
5903               PRINT CHR$(132);
5904           ELSE
5905               PRINT CHR$(128);
5906           END IF
5907           PRINT TABXY(1,Pointer);Marker$;CHR$(128);
5908       END IF
5909       RETURN
5910       !
5911       ! //////////////////////////////////////////////////
5912       !
5913 Write_screen:! Write the screen pointed to by Active_screen
5914       ! home and clear screen
5915       OUTPUT KBD;CHR$(255)&CHR$(84)&CHR$(255)&CHR$(75);
5916       !
5917       PRINT TABXY(1,First_line-1);CHR$(132);" Item #| Screen #";
5918       PRINT USING "#,2D,3A,2D,3A";Active_screen," of ";Screen_cnt;" | "
5919       PRINT T$;RPT$(" ",52-LEN(T$));CHR$(128);
5920       J=0
5921       REPEAT
5922           IF J=Last_item(Active_screen)-First_item(Active_screen) THEN
5923               PRINT CHR$(132);
5924               PRINT TABXY(1,First_line+J);RPT$(" ",80)
5925           ELSE
5926               PRINT CHR$(128);
5927           END IF
5928           PRINT TABXY(5,First_line+J);
5929           PRINT USING "3D,A;#";First_item(Active_screen)+J,"|"
5930           IF Skips>0 THEN  ! make this line inverse video
5931               FOR I=1 TO Skips
5932                   IF First_item(Active_screen)+J=Choose(I) THEN
5933                       PRINT CHR$(129);
5934                   END IF
5935               NEXT I
5936           END IF
5937           PRINT TABXY(10,First_line+J);Items$(First_item(Active_screen)+J)
5938           J=J+1
5939       UNTIL J>=(Last_item(Active_screen)-First_item(Active_screen)+1)
5940       Last_line=Last_item(Active_screen)-First_item(Active_screen)
5941       Last_line=Last_line+First_line
5942       !
5943       ! set marker to first non-selected item.
5944       !
5945       Pointeractive=0
5946       IF To_select>0 THEN Pointeractive=1
5947       IF Skips>0 AND Pointeractive=1 THEN ! find first non-selected item
5948           J=0
5949           LOOP
5950               Pointer=First_line+J
5951               FOR I=1 TO Skips
5952                   IF First_item(Active_screen)+J=Choose(I) THEN Pointer=0
5953               NEXT I
5954           EXIT IF Pointer<>0
5955               J=J+1
5956               IF First_line+J>Last_line THEN
5957                   Pointeractive=0
```

```
5958                  Pointer=First_line
5959              END IF
5960          EXIT IF Pointer<>0
5961          END LOOP
5962      ELSE
5963          Pointer=First_line
5964      END IF
5965      IF Pointeractive THEN
5966          IF Pointe-=Last_line THEN
5967              PRINT CHR$(132);
5968          ELSE
5969              PRINT CHR$(128);
5970          END IF
5971          PRINT TABXY(1,Pointer);Marker$;CHR$(128);
5972      END IF
5973      RETURN
5974   SUBEND
5975   !
5976   ! ************************************************************
5977   !
5978   SUB Edit_data(Prompt$,Variable,OPTIONAL Multiplier,Uvariable)
5979 Edit_data:OFF KEY
5980      IF NPAR>2 THEN
5981          Test=Variable*Multiplier
5982          IF NPAR=4 THEN Uvariable=Uvariable*Multiplier
5983      ELSE
5984          Test=Variable
5985      END IF
5986      ON ERROR GOTO Test_again
5987 Test_again:!
5988      OUTPUT 2 USING "K,#";Test
5989      DISP "Enter the value of ";Prompt$;
5990      INPUT Variable
5991      OFF ERROR
5992      IF NPAR=4 THEN
5993          Utest=Uvariable
5994          ON ERROR GOTO Utest_again
5995 Utest_again:!
5996          OUTPUT 2 USING "K,#";Utest
5997          DISP "Enter the uncertainty in ";Prompt$;
5998          INPUT Uvariable
5999          OFF ERROR
6000      END IF
6001      IF NPAR>2 THEN
6002          Variable=Variable/Multiplier
6003          IF NPAR=4 THEN Uvariable=Uvariable/Multiplier
6004      END IF
6005   SUBEND
6006   !
6007   ! ***************************************************************
6008   !
6009   SUB Phase_delay
6010 Phase_delay:  !
6011 Init_com:  !
6012      OPTION BASE 1
6013      COM /Lab_info/ REAL Rel_humidity,Temperature,Pressure,C
6014 !  COM /Sparms7 REAL Freq(801),COMPLEX S11(801),S21(801),S12(801),S22(801)
6015 !  COM /Sparms/ Mag_s11_id$[40],Ang_s11_id$[40],Mag_s21_id$[40],Ang_s21_id$
  => [40]
6016 !  COM /Sparms/ Mag_s22_id$[40],Ang_s22_id$[40],Mag_s12_id$[40],Ang_s12_id$
  => [40]
6017 !  COM /Sparm/ Frequency(801),COMPLEX Sparm(801),Sparm$[3],INTEGER Num_poin
  => ts
6018      COM /S_array/ INTEGER Dcount,Svalid,REAL S(801,5,2)
6019 Init_variables:  !
6020      RAD
```

```
6021      B=.02286
6022      Coaxial_len=1.15
6023      Waveguide_len=.36
6024 Calc_delay:  !
6025      Cutoff_freq=C/(2*B)  ! X-BAND WAVEGUIDE
6026      FOR J=1 TO Num_points
6027          Lambda=C/Frequency(J)
6028          Lambda_wg=Lambda/SQR(1-(Cutoff_freq/Frequency(J))^2)
6029          Phase_delay_wg=4*PI*Waveguide_len/Lambda_wg
6030          Phase_delay_cx=4*PI*Coaxial_len/Lambda
6031          Total_delay=Phase_delay_wg+Phase_delay_cx
6032          Real_delay=COS(Total_delay)
6033          Imag_delay=-SIN(Total_delay)
6034          IF BIT(Svalid,0) THEN S(J,1,1)=S(J,1,1)*Real_delay
6035          IF BIT(Svalid,0) THEN S(J,1,2)=S(J,1,2)*Imag_delay
6036          IF BIT(Svalid,1) THEN S(J,2,1)=S(J,2,1)*Real_delay
6037          IF BIT(Svalid,1) THEN S(J,2,2)=S(J,2,2)*Imag_delay
6038          IF BIT(Svalid,2) THEN S(J,3,1)=S(J,3,1)*Real_delay
6039          IF BIT(Svalid,2) THEN S(J,3,2)=S(J,3,2)*Imag_delay
6040          IF BIT(Svalid,3) THEN S(J,4,1)=S(J,4,1)*Real_delay
6041          IF BIT(Svalid,3) THEN S(J,4,2)=S(J,4,2)*Imag_delay
6042      NEXT J
6043      DEG
6044   SUBEND
6045 !
6046  ! ***************************************************************
6047 !
6048   SUB Enter_id(Id$,OPTIONAL Return_test$)
6049 Enter_id:!
6050   !
6051  !LAST REVISION  30/SEPT/86
6052      OPTION BASE 1
6053      COM /Bugs/ INTEGER Bug1,Bug2,Bug3,Printer
6054   !
6055      DIM Test$[160]
6056      INTEGER N
6057      N=LEN(Id$)
6058      Test$=Id$
6059      SELECT Id$
6060      CASE ""
6061          !OUTPUT NOTHING
6062      CASE ELSE
6063          OUTPUT 2 USING "K,#";Test$
6064      END SELECT
6065      SELECT NPAR
6066      CASE 1    !NO Return_test$ given
6067          DISP CHR$(129);"Please ENTER a description (<= 40 chrs).";
6068          DISP CHR$(128);
6069      CASE ELSE
6070          DISP CHR$(129);"Please ENTER a description (<= 40 chrs) ";
6071          DISP CHR$(128);
6072          SELECT Return_test$
6073          CASE Id$
6074              DISP " for THIS ID";
6075          CASE "ABORT"
6076              DISP " CLR LN/ ENTER to ABORT."
6077          CASE ELSE
6078              DISP " for ";Return_test$;
6079          END SELECT
6080      END SELECT
6081      LINPUT Test$
6082      DISP ""
6083      Test$=TRIM$(Test$)
6084      N=LEN(Test$)
6085      SELECT N
6086      CASE >40
```

```
6087          DISP "Length of data_id$ too long. You entered ";N;
6088          DISP " characters. Try again."
6089          BEEP
6090          WAIT 1.5
6091          IF NPAR=2 THEN
6092              IF Id$<>Return_test$ THEN
6093                  OUTPUT 2 USING "#,K";Test$
6094              END IF
6095          END IF
6096          GOTO Enter_id
6097      CASE =0
6098          IF NPAR>1 THEN
6099              IF Return_test$="ABORT" THEN
6100                  Id$=Test$   !=""
6101                  SUBEXIT
6102              END IF
6103          END IF
6104          DISP "You must ENTER SOMETHING or you'll ";
6105          DISP "never get out of this."
6106          BEEP
6107          WAIT 1.8
6108          GOTO Enter_id
6109      CASE ELSE
6110          !Everything ok
6111      END SELECT
6112      Id$=Test$
6113  SUBEND    !Enter_id
6114 !
6115 ! ***********************************************************
6116 !
6117  SUB Save_file(T_f(*),INTEGER Datacount,Id$)
6118 Save_file:  !
6119      OPTION BASE 1
6120      COM /Files/ Diskdrive$,Filename$
6121      ON ERROR CALL Errortrap
6122      Diskspace=INT((3500+(Datacount*16))/256)+1
6123      CREATE BDAT Filename$&Diskdrive$,Diskspace,256
6124!     CREATE ASCII Filename$&Diskdrive$,Diskspace*2
6125      ASSIGN @Datapath TO Filename$&Diskdrive$
6126      OUTPUT @Datapath;"N"
6127      OUTPUT @Datapath;TRIM$(Id$)
6128      OUTPUT @Datapath;Datacount
6129      OUTPUT @Datapath;Datacount
6130      OUTPUT @Datapath;T_f(*)
6131      ASSIGN @Datapath TO *
6132      OFF ERROR
6133  SUBEND     !Save_file
6134!
6135! *************************************************************
6136!
6137  SUB Enter_real(Low,High,Prompt$,Result)
6138      OPTION BASE 1
6139      INTEGER Bad_number
6140      DIM Test$[160]
6141      REAL Test_real
6142 Enter_real:!
6143      DISP "Please ENTER the desired value for";Prompt$;
6144      OUTPUT 2 USING "K,#";Result
6145      LINPUT Test$
6146      Bad_number=0
6147      ON ERROR GOSUB Trap_bad_number
6148      Test_real=VAL(TRIM$(Test$))
6149      OFF ERROR
6150      IF Bad_number THEN
6151          IF Test$="" THEN
6152              SUBEXIT
6153          ELSE
6154              GOTO Enter_real
6155          END IF
6156      END IF
6157      SELECT Test_real
6158      CASE <Low
6159          BEEP 1000,.3
6160          DISP " Number entered is TOO LOW. ";
6161          DISP " LOWEST allowable number is ";Low
6162          WAIT 2.1
6163          GOTO Enter_real
6164      CASE >High
6165          BEEP 1000,.3
6166          DISP " Number entered is TOO HIGH. ";
6167          DISP " HIGHEST allowable number is ";High
6168          WAIT 2.1
6169          GOTO Enter_real
6170      CASE ELSE
6171          Result=Test_real
6172          ! Number within limits
6173      END SELECT
6174      SUBEXIT
6175!
6176! ////////////////////////////////////////////////////////////
6177!
6178 Trap_bad_number:!
6179      SELECT ERRN
6180      CASE 15,32
6181          DISP CHR$(129);" What you ENTERED is not a number! Try again. ";
6182          DISP CHR$(128)
6183          Bad_number=1
6184          WAIT 1.7
6185          LINPUT "Please ENTER the number you wish for",Test$
6186      CASE ELSE
6187          DISP ERRN,ERRM$
6188          BEEP 850,.5
6189          Bad_number=1
6190          PAUSE
6191      END SELECT
6192      RETURN
6193 !
6194 ! ////////////////////////////////////////////////////////////
6195 !
6196  SUBEND
6197 !
6198 ! **************************************************************
6199 !
6200  SUB New_save_sparms(INTEGER Ssave)
6201 ! This subprogram saves the chosen S-parameters that are passed in common
6202 ! blocks.  The BITs 0 through 3 of the pass parameter Ssave are used to
6203 ! select which S-parameters are to be saved.
6204 ! The S(I,J,K) array is (801,5,2) big.  I=1:Dcount .. K=REAL/IMAG
6205 ! J=1:S11 .. J=2:S21 .. J=3:S12 .. J=4:S22 .. J=5,K=1:Frequency
6206 ! This array construct is the same as the NWA automated measurement program
6207 ! To save disk space when only interested in one S-parameter, other
6208 ! programs save only the frequency and the S-parameter.  That is why the
6209 ! Basket array is a variable size that is determined by Num_sparms.
6210 ! If Basket contained all four S-parameters, its dimension would be
6211 ! (I,J)=(Dcount,9), where J=1:Freqeuncy, 2/3:REAL/IMAG(S11), 4/5:REAL/IMAG(
  => S21), 6/7:REAL/IMAG(S12), 8/9:REAL/IMAG(S22)
6212 ! If Basket contained just S22 then (I,J)=(Dcount,3) where J=1:Frequency,
  =>  J=2/3:REAL/IMAG(S22), which saves diskspace.
6213 !
6214      OPTION BASE 1
6215      COM /Files/ Diskdrive$[20],Filename$[10]
6216      COM /S_array/ INTEGER Dcount,Svalid,REAL S(801,5,2)
```

```
6217        INTEGER Num_sparms,I
6218        Num_sparms=BIT(Ssave,0)+BIT(Ssave,1)+BIT(Ssave,2)+BIT(Ssave,3)
6219        ALLOCATE Basket(1:Dcount,Num_sparms*2+1)
6220 New_save_sparms:OFF KEY
6221        MAT Basket(1:Dcount,1)= S(1:Dcount,5,1)    !Frequency
6222        Column=2
6223        FOR I=1 TO 4
6224            IF BIT(Ssave,I-1) THEN          !SAVE S11,S21,S12,S22
6225                MAT Basket(1:Dcount,Column)= S(1:Dcount,I,1)
6226                MAT Basket(1:Dcount,Column+1)= S(1:Dcount,I,2)
6227                Column=Column+2
6228            END IF
6229        NEXT I
6230        DISP CHR$(131);" Saving: ";CHR$(128);Filename$;Diskdrive$
6231        Diskspace=INT(Dcount*16.*(Num_sparms*2+1)/256)+20
6232        ON ERROR GOSUB Trap_error
6233 !      CREATE BDAT Filename$&Diskdrive$,Diskspace,256
6234        CREATE ASCII Filename$&Diskdrive$,Diskspace
6235        OFF ERROR
6236        ASSIGN @Datapath TO Filename$&Diskdrive$
6237        OUTPUT @Datapath;Dcount,Ssave
6238        OUTPUT @Datapath;Basket(*)
6239        ASSIGN @Datapath TO *
6240        DEALLOCATE Basket(*)
6241        DISP
6242        SUBEXIT
6243 Trap_error:        DISP ERRM$
6244                    BEEP
6245                    PAUSE
6246                    RETURN
6247 Purge_old:PURGE Filename$&Diskdrive$
6248        BEEP
6249        PRINT "I just purged ";Filename$;Diskdrive$
6250        RETURN
6251    SUBEND
6252 !
6253 ! ***********************************************************
6254 !
6255    SUB Fit_sparms
6256 !      Fit_lorentz(Kx,Ky,Q,F0,Ukx,Uky,Uq,Uf0)             Sss=1
6257 ! or   Fit_lorentz(Q0,Ql,F0,Phase,Uq0,Uql,Uf0,Uphase)    Sss=2 or 3
6258  !Written by Eric J. Vanzura
6259  !Levenberg-Marquart algorithm CSUB taken from:
6260  !Numerical Recipes : the art of Scientific Computing
6261  !W. H. Press, B. P. Flannery, S. A. Teukolsky, W. T. Vetterling
6262  !Cambridge Press 1987
6263 Fit_sparms:OFF KEY
6264        GOSUB Init
6265        GOSUB Init_est
6266        GOSUB Initial_plot            !Plot meas'd S-parms & initial estimate
6267        REPEAT
6268 Do_mrq:!
6269            GOSUB Printlatestvals
6270            Ochisq=Chisq
6271            MAT Old_a= A
6272            DISP "Performing Levenberg-Marquart Least-square-fit"
6273            OFF KEY
6274            CALL Mrqmin(X(*),Y(*),Sig(*),Ndata(*),A(*),Ma(*),Lista(*),Mfit(*)
  => ,Covar(*),Alpha(*),Nca(*),Chisq,Alamda,Sss,Debug)
6275            GOSUB Convergencekeys
6276            MAT Old_est= Sgen
6277            GOSUB Gen_estimate
6278            IF NOT Speed THEN
6280                GOSUB Plot_estimate
6281            END IF
6282            GOSUB Ask_converge
```

```
6283        UNTIL Itst>=Itst_limit OR Force_converge
6284 Mrq_out:OFF KEY
6285        Alamda=0
6286        CALL Mrqmin(X(*),Y(*),Sig(*),Ndata(*),A(*),Ma(*),Lista(*),Mfit(*),Cov
  => ar(*),Alpha(*),Nca(*),Chisq,Alamda,Sss,Debug)
6287        IF NOT Speed THEN
6288        GOSUB Plot_residuals
6289        END IF
6291        GOSUB Calc_sig2
6292        GOSUB Calc_uncert
6293        GOSUB Print_results
6294        GOTO Subend
6295 !
6296 ! ///////////////////////////////////////////////////////////
6297 !
6298 Calc_uncert:!
6299        SELECT Sss
6300        CASE =1
6301            Ua(1)=SQRT(Sig2*Covar(1,1)) !Kx
6302            Ua(2)=SQRT(Sig2*Covar(2,2)) !Ky
6303            Ua(3)=SQRT(Sig2*Covar(3,3)) !Q
6304            Ua(4)=SQRT(Sig2*Covar(4,4)) !F0
6305            Ua(5)=SQRT(Sig2*Covar(5,5)) !
6306        CASE =2,=3
6307            Ua(1)=SQRT(Sig2*Covar(1,1)) !Uq0
6308            Ua(2)=SQRT(Sig2*Covar(2,2)) !Uql
6309            Ua(3)=SQRT(Sig2*Covar(3,3)) !Uf0
6310            Ua(4)=SQRT(Sig2*Covar(4,4)) !Uphase
6311            Ua(4)=Ua(4)*360/2/PI             !in degrees
6312            Ua(5)=SQRT(Sig2*Covar(5,5)) !Umag (beta)
6313        CASE ELSE
6314            BEEP
6315            DISP "Sss=";Sss;" in Calc_uncert. PAUSED"
6316            PAUSE
6317            GOTO Calc_uncert
6318        END SELECT
6319        RETURN
6320 !
6321 ! ///////////////////////////////////////////////////////////
6322 !
6323 Calc_sig2:!
6324        FOR I=1 TO Fitcount
6325            GOSUB Gen_cnum
6326            N=Start_index+I-1
6327            Sig2=Sig2+(ABS(CMPLX(S(N,Ns,1),S(N,Ns,2))-Cnum))^2
6328        NEXT I
6329        Sig2=Sig2/(Fitcount-Mfit(2))
6330        RETURN
6331 !
6332 ! ///////////////////////////////////////////////////////////
6333 !
6334 Print_results:!
6335        PRINTER IS 1
6336        PRINT Description$;" fit to ";Sparm$;" values."
6337        PRINT USING "K,3D,2(3X,K,MD.3DE)";"Iteration #",K,"Chi_squared=",Chis
  => q,"Alamda=",Alamda
6338        PRINT USING "3D,K,3D,K,3D";Dcount;" data points in file. Start_index=
  => ",Start_index,"  Stop_index=",Stop_index
6339        GOSUB Print_a
6340        GOSUB Print_ua
6341        PRINT USING "K,D.4DE";"Chi-squared=",Chisq
6342        IF Printer_on THEN
6343            PRINTER IS Print_addr
6344            GOSUB Print_a
6345            GOSUB Print_ua
6346            PRINT USING "K,D.4DE";"Chi-squared=",Chisq
```

```
6347          PRINTER IS 1
6348       END IF
6349       Reset_keys=1
6350       RETURN
6351 !
6352 ! ////////////////////////////////////////////////////////////
6353 !
6354 Initial_plot:!
6355       ALLOCATE File(Fitcount*3,2),Old_est(Fitcount,2),Orig_sparm(Fitcount,2
  => )
6356       ALLOCATE Sgen(Fitcount,2)
6357       MAT Orig_sparm(*,1:2)= S(Start_index:Stop_index,Ns,1:2)
6358       GOSUB Gen_estimate
6359       MAT Old_est= Sgen
6360       IF NOT Speed THEN GOSUB Plot_estimate
6361       RETURN
6362 !
6363 ! ////////////////////////////////////////////////////////////
6364 !
6365 Gen_cnum:!
6366       SELECT Sss
6367       CASE 1   !S21
6368          Cnum=-X(I)^2*CMPLX(A(1),-A(2))/CMPLX(X(I)^2-A(4)^2,-X(I)*A(4)/A(3
  => ))
6369       CASE 2   !S11
6370          Degrees=360*A(4)/2/PI
6371          P=CMPLX(COS(Degrees),SIN(Degrees))
6372          Ql2f=2*A(2)*(X(I)-A(3))/A(3)
6373          Cnum=A(5)*CMPLX(A(2)/A(1)+Ql2f*Ql2f,Ql2f*(1-A(2)/A(1)))*P/(1+Ql2f
  => *Ql2f)
6374       CASE 3   !S21
6375          Degrees=360*A(4)/2/PI
6376          P=CMPLX(COS(Degrees),SIN(Degrees))
6377          GOgl=1-A(1)/A(2)
6378          Dum=CMPLX(1,2*(X(I)-A(3))/A(2))
6379          Cnum=A(5)*GOgl*P/Dum
6380       CASE ELSE
6381          BEEP
6382          DISP "Sss=";Sss;" is not valid in Gen_cnum."
6383          PAUSE
6384       END SELECT
6385       RETURN
6386 !
6387 ! ////////////////////////////////////////////////////////////
6388 !
6389 Gen_estimate:!
6390       DISP "Generating latest estimate values"
6391       MAT Sgen= (0)
6392       SELECT Sss
6393       CASE =1  !Nussenzveig Lorentzian
6394          FOR I=1 TO Fitcount
6395             GOSUB Gen_cnum
6396             Sgen(I,1)=REAL(Cnum)
6397             Sgen(I,2)=IMAG(Cnum)
6398          NEXT I
6399       CASE =2  !Schulten S11
6400          FOR I=1 TO Fitcount
6401             GOSUB Gen_cnum
6402             Sgen(I,1)=REAL(Cnum)
6403             Sgen(I,2)=IMAG(Cnum)
6404          NEXT I
6405       CASE =3  !Schulten S21
6406          FOR I=1 TO Fitcount
6407             GOSUB Gen_cnum
6408             Sgen(I,1)=REAL(Cnum)
6409             Sgen(I,2)=IMAG(Cnum)
```

```
6410          NEXT I
6411       CASE ELSE
6412          DISP "Sss=";Sss;" is not valid in Gen_estimate."
6413          BEEP
6414          PAUSE
6415          GOTO Gen_estimate
6416       END SELECT
6417       RETURN
6418 !
6419 ! ////////////////////////////////////////////////////////////
6420 !
6421 Plot_estimate:!
6422       Pack_data(File(*))
6423       Id$=Sparm$
6424       Pen=2
6425       Pack_data(File(*),Orig_sparm(*),Fitcount,Pen,Id$)
6426 !
6427       Id$="Old estimate"
6428       Pen=4
6429       Pack_data(File(*),Old_est(*),Fitcount,Pen,Id$)
6430 !
6431       Id$="Latest estimate"
6432       Pen=6
6433       Pack_data(File(*),Sgen(*),Fitcount,Pen,Id$)
6434 !
6435       Test$="Real Part"
6436       Prompt$="Imaginary Part"
6437       Id$="Results from file:"&Rawdatafile$
6438       Init_graphics(Test$,Prompt$,Id$)
6439       Ax$="XY"
6440       Autoscale(File(*),Ax$)
6441       Refresh_labels
6442       DISP "S-parameters are: Red=measured data, Green=Old estimate, Blue=L
  => atest Estimate"
6443       Plot_all(File(*))
6444       RETURN
6445 !
6446 ! ////////////////////////////////////////////////////////////
6447 !
6450 Plot_residuals:!
6451       Pack_data(File(*))
6452       ALLOCATE Reres(Fitcount,2),Imres(Fitcount,2)
6453       FOR I=1 TO Fitcount
6455       Reres(I,1)=S(Start_index+I-1,5,1)/1.E+9
6456       Imres(I,1)=S(Start_index+I-1,5,1)/1.E+9
6458       Reres(I,2)=Sgen(I,1)-Orig_sparm(I,1)
6459       Imres(I,2)=Sgen(I,2)-Orig_sparm(I,2)
6460       NEXT I
6461       Id$="real residuals"
6462       Pen=2
6463       Pack_data(File(*),Reres(*),Fitcount,Pen,Id$)
6464 !
6465       Id$="imag residuals"
6466       Pen=4
6467       Pack_data(File(*),Imres(*),Fitcount,Pen,Id$)
6468 !
6472       Test$="Frequency (GHz)"
6473       Prompt$="residual"
6474       Id$="red=real green=imaginary"
6475       Init_graphics(Test$,Prompt$,Id$)
6476       Ax$="XY"
6477       Autoscale(File(*),Ax$)
6478       Refresh_labels
6480       Plot_all(File(*))
6481       RETURN
6482 !
```

```
6483 Init:!
6484        DEG
6485        OPTION BASE 1
6486        INTEGER I,Prty,Itst,Itst_limit,Ns
6487        INTEGER Pen,Done,Fitcount
6488        COMPLEX Cnum,P,Dum
6489        REAL Debug
6490        DIM Test$[160],Prompt$[40],Id$[40],Rawdatafile$[10],Sparm$[3]
6491        COM /History/ Status$[1],Time_orgn$[8],Date_orgn$[11]
6492        COM /History/ Time_chng$[8],Date_chng$[11],Description$[160]
6493        COM /Labels/ Labels$(30)[60],INTEGER Lbl_count,REAL Lbl_addr(30,6)
6494        COM /Data_param/ INTEGER Data_count,Filesize,Curvecount,Roster(17,4)
6495        COM /Data_param/ REAL Sym_size,Symbol$(17)[2],Curve_id$(17)[40]
6496        COM /Data_param/ REAL Xmin_data,Xmax_data
6497        COM /Data_param/ REAL Ymin_data,Ymax_data
6498        COM /Graphics/ INTEGER Speed
6499        COM /Background/ Graphtype$[12],Margins$(2)[10],Papersize$[1]
6500        COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
6501        COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
6502        COM /Background/ Xgrid_tick$[4],INTEGER Xmajor,Xminor
6503        COM /Background/ Ygrid_tick$[4],INTEGER Ymajor,Yminor
6504        COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
6505        COM /Axes_labels/ Print_xlabels$[3],Print_ylabels$[3]
6506        COM /Axes_labels/ Sig_digits$[2],REAL Xlcsize,Ylcsize
6507        COM /Windowspace/ REAL Xmin,Xmid,Xmax,Ymin,Ymid,Ymax! graph edges UDU
6508        COM /Windowspace/ REAL Xleft,Xright,Ybottom,Ytop! paper edges UDUs
6509        COM /Log_scale/ REAL Xcycles,Xbegin,Ycycles,Ybegin
6510        COM /Plot_device/ Plot_lang$[10],INTEGER Plot_addr
6511        COM /Hard_space/ REAL Xview_lft,Xview_rt,Yview_btm,Yview_top
6512        COM /Hard_space/ REAL Viewscale,Aspect_ratio
6513        COM /Frame_onoff/ INTEGER Frame_flag
6514        COM /Clear_space/ REAL Space_lft,Space_rt,Space_btm,Space_top
6515 !      !
6516 Init_com:  !
6517        COM /Fittype/ INTEGER Sselect,Start_index,Stop_index
6518        COM /S_array/ INTEGER Dcount,Svalid,REAL S(801,5,2)
6519        COM /Cdata/ INTEGER Mfit(2),Lista(5,2),Ma(2),REAL A(5),Ua(5),INTEGER
  => Ndata(2)
6520        COM /Cstats/ Alamda,Chisq,INTEGER Nca(2),REAL Alpha(5,5),Covar(5,5)
6521        COM /Output/ INTEGER Print_addr,Plotter,Printer_on,Plotter_on
6522 Init_vals:!
6523           Speed=0
6524        K=0
6525        Itst=0
6526        Itst_limit=3
6527        Plot_addr=3
6528        Plot_lang$="INTERNAL"
6529        M=5       !Dimension of Covar,Alpha,Lista & A matricies
6530        Debug=0
6531 !
6532        IF NOT BIT(Svalid,Sselect) THEN
6533            BEEP
6534            DISP "The S(*) array for ";Sparm$;" is not valid (doesn't have da
  => ta?) PAUSED"
6535            PRINT "The S(*) array for ";Sparm$;" is not valid (doesn't have d
  => ata?) PAUSED"
6536            PAUSE
6537        END IF
6538 !Sss=1 S21-Nussenzvieg    Sss=2 S11-Schulten    Sss=3 S21-Schulten
6539        SELECT Sselect
6540        CASE 0       !S11
6541            Sss=2
6542            Sparm$="S11"
6543        CASE 1       !S21
6544            Sss=1
6545 !          Sss=3
```

```
6546            Sparm$="S21"
6547        CASE 2       !S12
6548            Sss=1
6549            Sss=3
6550            Sparm$="S12"
6551        CASE 3       !S22
6552            Sss=2
6553            Sparm$="S22"
6554        CASE ELSE
6555            BEEP
6556            DISP "Sselect is not valid in Fit_sparms"
6557            PAUSE
6558        END SELECT
6559        Ns=Sselect+1
6560        RETURN
6561 !
6562 ! ////////////////////////////////////////////////////////////////
6563 !
6564 Init_est:  !
6565        Fitcount=Stop_index-Start_index+1
6566        Start_freq=S(Start_index,5,1)
6567        Stop_freq=S(Stop_index,5,1)
6568        SELECT Sss
6569        CASE =1  ! Nussenzveig Lorentzian
6570            Ix=INT(Start_index+Fitcount/2)
6571            Angle=ARG(CMPLX(S(Ix,Ns,1),S(Ix,Ns,2)))
6572            Mag=ABS(CMPLX(S(Ix,Ns,1),S(Ix,Ns,2)))
6573            A(1)=-2*10^(-5)*Mag*SIN(Angle)
6574            A(2)=-2*10^(-5)*Mag*COS(Angle)
6575            A(3)=1.5*S(Ix,5,1)/(Stop_freq-Start_freq)
6576            A(4)=S(Ix,5,1)
6577        CASE =2,=3  ! Schulten S11 or S21
6578            Ix=INT(Start_index+Fitcount/2)
6579 !          A(1)=S(Ix,5,1)/(Stop_freq-Start_freq)*1.8
6580 !          A(2)=S(Ix,5,1)/(Stop_freq-Start_freq)*2.0
6584            A(1)=S(Ix,5,1)/90000
6585            A(2)=S(Ix,5,1)/79000
6586            A(3)=S(Ix,5,1)
6590            A(4)=ARG(CMPLX(S(Ix,Ns,1),S(Ix,Ns,2)))*2*PI/360
6591 !          A(4)=0
6592 !          A(5)=ABS(CMPLX(S(Ix,Ns,1),S(Ix,Ns,2)))
6593            A(5)=1
6594        CASE ELSE
6595            DISP "Sss=";Sss;" is not valid in Init_est"
6596            BEEP
6597            PAUSE
6598            GOTO Init_est
6599        END SELECT
6600        Ndata(1)=0
6601        Ndata(2)=Fitcount*2
6602        Ma(1)=0              !size of A array (unknowns)
6603        Ma(2)=M
6604        Chisq=1              !measure of fit
6605        Mfit(1)=0            !number of unknowns to be fitted
6606        Mfit(2)=M
6607        IF Sss=1 THEN Mfit(2)=M-1
6608        MAT Lista= (0)
6609        Lista(1,2)=1         !A array index of the unknowns to be fitted
6610        Lista(2,2)=2
6611        Lista(3,2)=3
6612        Lista(4,2)=4
6613        Lista(5,2)=5
6614        IF Sss=1 THEN Lista(5,2)=0
6615        MAT Covar= (0)
6616        MAT Alpha= (0)
6617        Nca(1)=0             !Size of Alpha & Covar arrays >= Ma
```

```
6618       Nca(2)=M
6619       Alamda=-2
6620       ALLOCATE Old_a(M)
6621       ALLOCATE REAL X(Fitcount*2),Y(Fitcount*2),Sig(Fitcount*2)
6622       MAT Sig= (1)
6623       MAT X(1:Fitcount)= S(Start_index:Stop_index,5,1)
6624       MAT X(Fitcount+1:Fitcount*2)= S(Start_index:Stop_index,5,1) '
6625       MAT Y(1:Fitcount)= S(Start_index:Stop_index,Ns,1)
6626       MAT Y(Fitcount+1:Fitcount*2)= S(Start_index:Stop_index,Ns,2)
6627       RETURN !Init_est
6628 !
6629 ! //////////////////////////////////////////////////////////
6630 !
6631 Done_return:OFF KEY
6632       Done=1
6633       RETURN
6634 !
6635 ! //////////////////////////////////////////////////////////
6636 !
6637 Force_converge:PRINT "Convergence will be forced."
6638       Force_converge=1
6639       RETURN
6640 !
6641 ! //////////////////////////////////////////////////////////
6642 !
6643 Preventconverge:PRINT "Convergence will be prevented."
6644       Itst=0
6645       RETURN
6646 !
6647 ! //////////////////////////////////////////////////////////
6648 !
6649 Graph_on: !
6650       Speed=0
6651       OFF KEY 4
6652 !     ON KEY 4 LABEL "GRAPH  OFF",Prty GOSUB Graph_off
6653       RETURN
6654 !
6655 ! //////////////////////////////////////////////////////////
6656 !
6657 Graph_off: !
6658       Speed=1
6659       OFF KEY 4
6660 !     ON KEY 4 LABEL "GRAPH  ON",Prty GOSUB Graph_on
6661       RETURN
6662 !
6663 ! //////////////////////////////////////////////////////////
6664 !
6666 Debug_on:PRINT "Debug flag is turned on."
6667       Debug=1
6668       OFF KEY 6
6669       ON KEY 6 LABEL "DEBUG  OFF",Prty GOSUB Debug_off
6670       RETURN
6671 !
6672 ! ///////////////////////////////////////////''///////////////
6673 !
6674 Debug_off:PRINT "Debug flag is turned off."
6675       Debug=0
6676       OFF KEY 6
6677       ON KEY 6 LABEL "DEBUG  ON",Prty GOSUB Debug_on
6678       RETURN
6679 !
6680 ! //////////////////////////////////////////////////////////
6681 !
6682 Ask_converge:!
6683       SELECT Chisq
6684       CASE >Ochisq
6685             Itst=0
6686             PRINT USING "K,D.DE";"New Chisq is greater than old Chisq by:";Ch
  => isq-Ochisq
6687       CASE =Ochisq
6688             PRINT "Chisq is equal to old Chisq"
6689       CASE <Ochisq
6690             PRINT USING "K,D.DE";"New Chisq is LESS than old Chisq by:";Ochis
  => q-Chisq
6691             IF ABS((Ochisq-Chisq)/Chisq)<.003 THEN Itst=Itst+1
6692             IF ABS((Ochisq-Chisq)/Chisq)<.00003 THEN Itst=Itst+1
6693             PRINT USING "K,D";"Itst=";Itst
6694             PRINT USING "K,D.DE";"ABS((Ochisq-Chisq)/Chisq)=";ABS((Ochisq-Chi
  => sq)/Chisq)
6695       END SELECT
6696       RETURN
6697 !
6698 ! /////////////////////////////////////////////////////////
6699 !
6700 Convergencekeys:!
6701       Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
6702       ON KEY 0 LABEL "FORCE  CONVERG",Prty GOSUB Force_converge
6703       ON KEY 2 LABEL "PREVENTCONVERG",Prty GOSUB Preventconverge
6704       IF Speed THEN
6705             ON KEY 4 LABEL "GRAPH  ON",Prty GOSUB Graph_on
6706       ELSE
6707             ON KEY 4 LABEL "GRAPH  OFF",Prty GOSUB Graph_off
6708       END IF
6709       IF Debug THEN
6710             ON KEY 6 LABEL "DEBUG  OFF",Prty GOSUB Debug_off
6711       ELSE
6712             ON KEY 6 LABEL "DEBUG  ON",Prty GOSUB Debug_on
6713       END IF
6714       RETURN
6715 !
6716 ! /////////////////////////////////////////////////////////
6717 !
6718 Printlatestvals:!
6719       PRINT USING "K,MD.3DE";"Chisq=",Chisq
6720       K=K+1
6721       PRINT USING "K,3D,L,K,MD.3DE";"Iteration #",K,"Alamda=",Alamda
6722       GOSUB Print_a
6723       RETURN
6724 !
6725 ! /////////////////////////////////////////////////////////
6726 !
6727 Print_a:!
6728       SELECT Sss
6729       CASE =1
6730             IF A(4)<1.E+10 THEN
6731                   PRINT USING A_flt10g;"QL=",A(3),"F0 =",A(4)/1.E+9,"F0+(Span/2
  => )=",(A(4)+A(4)/2/A(3))/1.E+9,"Span=",A(4)/A(3)/1000
6732             ELSE
6733                   PRINT USING A_fgt10g;"QL=",A(3),"F0 =",A(4)/1.E+9,"F0+(Span/2
  => )=",(A(4)+A(4)/2.E+9/A(3))/1.E+9,"Span=",A(4)/A(3)/1000
6734             END IF
6735       CASE =2,=3
6736             PRINT USING "2(K,MD.4DE,2X),K,MD.9DE,2X,K,M3D.DD,2X,K,MD.3DE";"Q0
  => =",A(1),"Ql=",A(2),"F0=",A(3),"Degrees=",A(4)*360/2/PI,"Mag=",A(5)
6737       CASE ELSE
6738             DISP "Sss=";Sss;" is not valid in print_a. program paused"
6739             BEEP
6740             PAUSE
6741       END SELECT
6742       RETURN
6743 A_flt10g:IMAGE K,6D,X,K,X,D.9D,X,K,X,D.9D,X,K,3D.3D
6744 A_fgt10g:IMAGE K,6D,X,K,DD.9D,X,K,DD.9D,X,K,3D.3D
```

```
6745 !
6746 ! ////////////////////////////////////////////////////////////
6747 !
6748 Print_ua:!
6749      SELECT Sss
6750      CASE =1
6751           PRINT USING "2(K,MD.3DE,4X),K,M5D,3X,K,3X,MD.3DE";"Ukx=",Ua(1),"U
  => ky=",Ua(2),"UQ=",Ua(3),"Uf0=",Ua(4)
6752      CASE =2,=3
6753           PRINT USING "2(K,MD.3DE,2X),K,MD.9DE,2X,K,M3D.3D,2X,K,MD.3DE";"UQ
  => 0=",Ua(1),"UQl=",Ua(2),"UF0=",Ua(3),"Udegrees=",Ua(4),"Umag=",Ua(5)
6754      CASE ELSE
6755           DISP "Sss=";Sss;" is not valid in Print_ua. program paused"
6756           BEEP
6757           PAUSE
6758      END SELECT
6759      RETURN
6760 Subend:!
6761   SUBEND
6762 !
6763 ! ************************************************************
6764 !
6765   SUB New_load_sparms
6766 ! This subprogram loads S-parameters into an allocated basket which then
6767 ! puts the S-parameters into the appropriate S_parm according to the
6768 ! 0 through 3 BIT of the parameter Ssave.
6769 ! The S(I,J,K) array is (801,5,2) big.  I=1:Dcount .. K=REAL/IMAG
6770 ! J=1:S11 .. J=2:S21 .. J=3:S12 .. J=4:S22 .. J=5,K=1:Frequency
6771 ! This array construct is the same as the NWA automated measurement program
6772 ! To save disk space when only interested in one S-parameter, other
6773 ! programs save only the frequency and the S-parameter.  That is why the
6774 ! Basket array is a variable size that is determined by Num_sparms.
6775 ! If Basket contained all four S-parameters, its dimension would be
6776 ! (I,J)=(Dcount,9), where J=1:Freqeuncy, 2/3:REAL/IMAG(S11), 4/5:REAL/IMAG(
  => S21), 6/7:REAL/IMAG(S12), 8/9:REAL/IMAG(S22)
6777 ! If Basket contained just S22 then (I,J)=(Dcount,3) where J=1:Frequency,
  => J=2/3:REAL/IMAG(S22), which saves diskspace.
6778      OPTION BASE 1
6779      COM /Files/ Diskdrive$[20],Filename$[10]
6780      COM /S_array/ INTEGER Dcount,Svalid,REAL S(*)
6781      INTEGER Num_sparms,I
6782 New_load_sparms:OFF KEY
6783      DISP CHR$(131);" Loading: ";CHR$(128);Filename$;Diskdrive$
6784      ASSIGN @Datapath TO Filename$&Diskdrive$
6785      ENTER @Datapath;Dcount,Svalid
6786      Num_sparms=BIT(Svalid,0)+BIT(Svalid,1)+BIT(Svalid,2)+BIT(Svalid,3)
6787      ALLOCATE Basket(Dcount,Num_sparms*2+1)
6788      ENTER @Datapath;Basket(*)
6789      ASSIGN @Datapath TO *
6790      MAT S(1:Dcount,5,1)= Basket(*,1)
6791      Column=2
6792      FOR I=1 TO 4
6793           IF BIT(Svalid,I-1) THEN        !TRANSFER TO S11,S21,S12,S22
6794                MAT S(1:Dcount,I,1)= Basket(1:Dcount,Column)
6795                MAT S(1:Dcount,I,2)= Basket(1:Dcount,Column+1)
6796                Column=Column+2
6797           END IF
6798      NEXT I
6799      DEALLOCATE Basket(*)
6800      DISP
6801   SUBEND
6802 !
6803 ! ************************************************************
6804 !
6805   SUB Pause_key_on
6806 Pause_key_on: ! Make sure that CONTINUE key exists.
6807                ! Original: 02 Dec 1987
6808                ! Revision: 02 Dec 1987
6809      COM /Sys/ Sys_id$[10]
6810      IF Sys_id$[1,4]="S300" THEN ! reset to S300 system keys
6811           CONTROL KBD,15;0
6812           CONTROL CRT,12;2
6813           LOAD KEY
6814      END IF
6815      PAUSE
6816      IF Sys_id$[1,4]="S300" THEN  ! set to S200 compatible keys
6817           OUTPUT KBD USING "K,#";"SCRATCH KEY X"
6818           CONTROL KBD,15;1
6819           CONTROL CRT,12;0
6820      END IF
6821      SUBEXIT
6822 SUBEND
6823 !
6824 ! ************************************************************
6825 !
```

Appendix E

CAVITYPROG Program Listing

```
4  !     PURGE "CAVITYPROG"
6  !     RE-STORE "CAVITYPROG"
8  ! Written by Eric J. Vanzura & Michael D. Janezic
10!
12      GOSUB Init
14      GOSUB Main_menu
16      GOTO End
18    !
20    ! ////////////////////////////////////////////////////////////
22    !
24 Init:  !
26      OPTION BASE 1
28 Init_com: !
30      COM /Environment/ Degc,Mbar,Relh,C
32      COM /Environment/ Udegc,Umbar,Urelh,Uc
34      COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
36      COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
38      COM /Sample/ Erreps(12),Errtand(12)
40      COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
42      COM /Cavity_dims/ Udiameter,Ulength
44      COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
46      COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
48      COM /Constants/ Bessel_root
50      COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,Prom
  => pt$[80]
52      COM /History/ Status$[1],Time_orgn$[8],Date_orgn$[11]
54      COM /History/ Time_chng$[8],Date_chng$[11],Description$[160]
56      COM /Labels/ Labels$(30)[60],INTEGER Lbl_count,REAL Lbl_addr(30,6)
58      COM /Data_param/ INTEGER Data_count,Filesize,Curvecount,Roster(17,4)
60      COM /Data_param/ REAL Sym_size,Symbol$(17)[2],Curve_id$(17)[40]
62      COM /Data_param/ REAL Xmin_data,Xmax_data
64      COM /Data_param/ REAL Ymin_data,Ymax_data
66      COM /Background/ Graphtype$[12],Margins$(2)[10],Papersize$[1]
68      COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
70      COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
72      COM /Background/ Xgrid_tick$[4],INTEGER Xmajor,Xminor
74      COM /Background/ Ygrid_tick$[4],INTEGER Ymajor,Yminor
76      COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
78      COM /Axes_labels/ Print_xlabels$[3],Print_ylabels$[3]
80      COM /Axes_labels/ Sig_digits$[2],REAL Xlcsize,Ylcsize
82      COM /Windowspace/ REAL Xmin,Xmid,Xmax,Ymin,Ymid,Ymax     ! graph edges UDU
84      COM /Windowspace/ REAL Xleft,Xright,Ybottom,Ytop    ! paper edges UDUs
86      COM /Log_scale/ REAL Xcycles,Xbegin,Ycycles,Ybegin
88      COM /Plot_device/ Plot_lang$[10],INTEGER Plot_addr
90      COM /Hard_space/ REAL Xview_lft,Xview_rt,Yview_btm,Yview_top
92      COM /Hard_space/ REAL Viewscale,Aspect_ratio
94      COM /Frame_onoff/ INTEGER Frame_flag
96      COM /Clear_space/ REAL Space_lft,Space_rt,Space_btm,Space_top
98    !      !
100     COM /Ana_data/ INTEGER Prog_id,Sweep_type,Sweep_mode,Datacount
102     COM /Ana_data/ INTEGER S11_valid,S21_valid,S12_valid,S22_valid,Ports
104     COM /Ana_data/ REAL Z0,Start,Stop
106     COM /Ana_data/ Title$[55],Company$[30],Operator_name$[30],Measure_time$[3
  => 0]
108     COM /Fittype/ INTEGER Sselect,Start_index,Stop_index
110     COM /S_array/ INTEGER Dcount,Svalid,REAL S(801,5,2)
112     COM /Cdata/ INTEGER Mfit(2),Lista(5,2),Ma(2),REAL A(5),Ua(5),INTEGER Ndat
  => a(2)
114     COM /Cstats/ Alamda,Chisq,INTEGER Nca(2),REAL Alpha(5,5),Covar(5,5)
116     COM /Output/ INTEGER Print_addr,Plotter,Printer_on,Plotter_on
118     COM /Sys/ Sys_id$[10]
120     COM /Sys_msi/ Msi_id$[20]
122     COM /Files/ Diskdrive$[20],Filename$[10]
124     COM /Enlarge_file/ INTEGER Overflow
126   ! COM /Fitsparm/ COMPLEX Sparm(801),Sparm$[3]
128     INTEGER Prty
130   !
132   ! ////////////////////////////////////////////////////////////
134   !
136 Init_vals: ! All values in SI (MKS) units
138     Bessel_root=3.8317059702
140     IF Diameter=0 OR Length0=0 THEN
142         Udiameter=.002/1.E+3
144         Ulength=.013/1.E+3
146         Diameter=.06002996
148         Length0=.4303531
150     END IF
152     IF Samplelen=0 OR Eps_re_guess=0 THEN
154         Eps_re_guess=3.83
156         Samplelen=.0144854
158         Usamplelen=.0000001
160     END IF
162     IF Freq(2)=0 THEN
164         Freq(2)=1.002389381E+10
166         Ufreq(2)=5.E+3
168     END IF
170     IF Degc=0 OR Relh=0 OR Mbar=0 THEN
172         Degc=23
174         Relh=40
176         Mbar=830
178         Udegc=.05
180         Urelh=1
182         Umbar=1
184     END IF
186     MAT Spectrum= (0)
188     FOR I=1 TO SIZE(Spectrum,1)
190         Spectrum(I,1)=1000
192     NEXT I
194     Num_modes=0
196     CALL Speed_of_light
198     CALL Uncert_c
199     Printer_on=1
200     Print_addr=9
202     RETURN
203   !
204   ! ////////////////////////////////////////////////////////////
206   !
208 Main_menu:!
210     GOSUB Menu_main
212     Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
214     ON KEY 1 LABEL "End    Program",Prty GOSUB End_program
216     ON KEY 2 LABEL "VarableLength",Prty CALL Var_length
218     ON KEY 3 LABEL "VarableFreqncy",Prty CALL Var_freq
220   ! ON KEY 4 LABEL "Calc   Dimens",Prty CALL Calc_dimens
222     ON KEY 4 LABEL "Cavity Dimens",Prty CALL Cavity_dimens
224     ON KEY 5 LABEL "Calc   C",Prty CALL Calc_c
226     ON KEY 6 LABEL "Calc   Modes",Prty CALL Calc_modes
228     ON KEY 7 LABEL "Calc   Freqs",Prty CALL Calc_freqs
230     ON KEY 8 LABEL "Fit    S parms",Prty CALL Fit_sparms
232   ! ON KEY 9 LABEL "Error  Curves",Prty CALL Error_curves
234     Done=0
236     Prior_menu=0
238     LOOP
240         IF Done THEN GOTO Main_menu
242     EXIT IF Prior_menu
244     END LOOP
246     RETURN
248   !
250   ! ////////////////////////////////////////////////////////////
252   !
254 End:OFF KEY
256     DISP "Program finished. Enter 'RUN' to start again."
```

```
257    SELECT TRIM$(Sys_id$)
258    CASE "PC300"
260        LOAD KEY "EDITKEYS:DOS,C"
261    CASE "S300"
262        LOAD KEY "/BASIC/EDITKEYS:,1400,0"
263    END SELECT
264    STOP
265  !
266  ! ////////////////////////////////////////////////////////////
267  !
268 Menu_main: !
270    CLEAR SCREEN
272    PRINT USING Image6;Sample_id$
274    PRINT USING Image7;Samplelen*1.E+3,Usamplelen*1.E+3
276    PRINT USING Image8;Eps_re_guess
278    PRINT USING Image1;Degc,Udegc,Relh,Urelh
280    PRINT USING Image2;Mbar,Umbar
282    PRINT USING Image3;C,Uc
284    PRINT
286    PRINT USING Image4;Diameter*1.E+3,Udiameter*1..E+3
288    PRINT USING Image5;Length0*1.E+3,Ulength*1.E+3
290    DISP "Main program menu (select from options given below)"
292    RETURN
294  !
296 Image1:IMAGE "Temperature = ",2D.2D," +/- ",D.2D," deg C   Relative Humidity
  =>  = ",2D.D," +/- ",D.D," %"
298 Image2:IMAGE "Barometric Pressure = ",3D.D," +/- ",D.D,"   mbar"
300 Image3:IMAGE "Speed of Light      = ",D.7DESZ," +/- ",D.3DESZ," meters/secon
  => d"
302 Image4:IMAGE "Resonator Diameter             =  ",2D.5D," +/- ",D.5D,"  mm"
304 Image5:IMAGE "          Length (micr. zero) = ",3D.5D," +/- ",D.5D,"  mm"
306 Image6:IMAGE "Sample    Description: ",K,"                    "
308 Image7:IMAGE "          Length      = ",2D.4D," +/- ",D.4D," mm"
310 Image8:IMAGE "       Epsilon Guess = ",3D.4D
312 !
314 ! ////////////////////////////////////////////////////////////
316 !
318 End_program:OFF KEY
320    Prior_menu=1
322    RETURN
324    END
326 !
328 ! ************************************************************
330 !
332    SUB Var_length
334 Var_length:OFF KEY
336        OPTION BASE 1
338        COM /Environment/ Degc,Mbar,Relh,C
340        COM /Environment/ Udegc,Umbar,Urelh,Uc
342        COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
344        COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
346        COM /Sample/ Erreps(12),Errtand(12)
348        COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
350        COM /Cavity_dims/ Udiameter,Ulength
352        COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
354        COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
356        COM /Constants/ Bessel_root
358        COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
360    !
362        GOSUB Menu_var_len
364    !
366        DISP CHR$(129);" This is the Fixed-frequency/Variable-length menu ";C
  => HR$(128)
368        Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
370        ON KEY 1 LABEL "Prior  Menu",Prty CALL Prior_menu
372        ON KEY 2 LABEL "Sample Info",Prty CALL Sample_info
374        ON KEY 3 LABEL "Resnce Freq",Prty CALL Resnce_freq
376        ON KEY 4 LABEL "Micr.  Readngs",Prty CALL Micr_readings
378        ON KEY 5 LABEL "Q      Values",Prty CALL Q_values
380        ON KEY 6 LABEL "Band   Widths",Prty CALL Band_widths
382        ON KEY 7 LABEL "Calc   Epsilon",Prty GOSUB Calc_eps_len
384        ON KEY 8 LABEL "Calc   Errors ",Prty GOSUB Calc_errs_len
386        Done=0
388        Prior_menu=0
390        LOOP
392            IF Done THEN GOTO Var_length
394        EXIT IF Prior_menu
396        END LOOP
398        OFF KEY
400        Prior_menu=0
402        Done=T
404        SUBEXIT
406 !
408 Calc_eps_len:OFF KEY
410        Do_var_freq=0
412        CALL Calc_eps_re
414        CALL Calc_tand
416        CALL Calc_eps_im
418        CALL Calc_errors(1)
420        Done=1
422        RETURN
424  !
426  ! ////////////////////////////////////////////////////////////
428  !
430 Calc_errs_len:OFF KEY
432        Do_var_freq=0
434        CALL Calc_errors
436        Done=1
438        RETURN
440  !
442  ! ////////////////////////////////////////////////////////////
444  !
446 Menu_var_len:  !
448        CLEAR SCREEN
450        PRINT USING Image6;Sample_id$
452        PRINT USING Image6a;Eps_re_guess,"Method:Variable Length"
454        PRINT USING Image7;Samplelen*1.E+3,Usamplelen*1.E+3
456        PRINT
458        PRINT USING Image1;Degc,Udegc,Relh,Urelh
460        PRINT USING Image2;Mbar,Umbar
462        PRINT USING Image3;C,Uc
464        PRINT
466        PRINT USING Image4;Diameter*1.E+3,Udiameter*1.E+3
468        PRINT USING Image5;Length0*1.E+3,Ulength*1.E+3
470        PRINT
472        PRINT USING Image10ab
474        PRINT USING Image11b;Freq(1)/1.E+9,Ufreq(1)/1.E+9,Freq(2)/1.E+9,Ufreq
  => (2)/1.E+9
476        PRINT USING Image11a;Micr(1)*1.E+3,Umicr(1)*1.E+3,Micr(2)*1.E+3,Umicr
  => (2)*1.E+3
478        PRINT USING Image12ab;Q(1),Uq(1),Q(2),Uq(2)
480        PRINT USING Image13ab;Bw(1)/1.E+3,Ubw(1)/1.E+3,Bw(2)/1.E+3,Ubw(2)/1.E
  => +3,8
482        PRINT
484        PRINT USING Image15ab;Eps_re,Ueps_re,Eps_im,Ueps_im
486        PRINT USING Image16ab;Tand,Utand
488        RETURN
490 !
492 Image1:IMAGE "Temperature = ",2D.2D," +/- ",D.2D," deg C   Relative Humidity
  =>  = ",2D.D," +/- ",D.D," %"
494 Image2:IMAGE "Barometric Pressure = ",3D.D," +/- ",D.D,"   mbar"
```

```
496 Image3:IMAGE "Speed of Light       = ",D.7DESZ," +/- ",D.3DESZ," meters/secon
  => d"
498 Image4:IMAGE "Resonator Diameter                 =  ",2D.5D," +/- ",D.5D,"  mm"
500 Image5:IMAGE "             Length (micr. zero) = ",3D.5D," +/- ",D.5D,"  mm"
502 Image6:IMAGE "Sample Description: ",K,"                        "
504 Image6a:IMAGE "Initial guess for epsilon':",2D.3D,10X,K
506 Image7:IMAGE "          Length    = ",2D.4D," +/- ",D.4D," mm"
508 Image10ab:IMAGE 28X,"Empty",20X,"with Sample"
510 Image11a:IMAGE "Micrometer reading = ",2(S2D.4D,"   +/- ",D.4D,6X),"mm"
512 Image11b:IMAGE "Resonance frequency= ",2(2D.7D," +/- ",D.7D,3X),"GHz"
514 Image12ab:IMAGE "Q value                =   ",2(5D,"    +/- ",3D,11X)
516 Image13ab:IMAGE "3 dB Bandwidth      =  ",2(5D.2D,"   +/- ",3D.2D,6X),B,"kHz"
518 Image15ab:IMAGE "Epsilon'  = ",3D.4D," +/- ",D.4D,5X,"Epsilon''  = ",2D.5D,"
  =>  +/- ",D.5D
520 Image16ab:IMAGE "Tan delta = ",D.6D," +/- ",D.6D
522 !
524 ! ////////////////////////////////////////////////////////////
526 !
528    SUBEND
530 !
532 ! ************************************************************
534 !
536    SUB Var_freq
538 Var_freq:OFF KEY
540         OPTION BASE 1
542         COM /Environment/ Degc,Mbar,Relh,C
544         COM /Environment/ Udegc,Umbar,Urelh,Uc
546         COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
548         COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
550         COM /Sample/ Erreps(12),Errtand(12)
552         COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
554         COM /Cavity_dims/ Udiameter,Ulength
556         COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
558         COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
560         COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
562         COM /Constants/ Bessel_root
564    !
566         GOSUB Menu_var_freq
568    !
570         DISP CHR$(129);" This is the Fixed-length/Variable-frequency menu ";C
  => HR$(128)
572         Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
574         ON KEY 1 LABEL "Prior  Menu",Prty CALL Prior_menu
576         ON KEY 2 LABEL "Sample Info",Prty CALL Sample_info
578         ON KEY 3 LABEL "Resnce Freqs",Prty CALL Resnce_freqs
580         ON KEY 4 LABEL "Micr.  Readng",Prty CALL Micr_reading
582         ON KEY 5 LABEL "Q      Values",Prty CALL Q_values
584         ON KEY 6 LABEL "Band   Widths",Prty CALL Band_widths
586         ON KEY 7 LABEL "Calc   Epsilon",Prty GOSUB Calc_eps_freq
588         ON KEY 8 LABEL "Calc   Errors ",Prty GOSUB Calc_errs_freq
590         Done=0
592         Prior_menu=0
594         LOOP
596             IF Done THEN GOTO Var_freq
598         EXIT IF Prior_menu
600         END LOOP
602         OFF KEY
604         Prior_menu=0
606         Done=1
608         SUBEXIT
610 !
612 Calc_eps_freq:OFF KEY
614         Do_var_freq=1
616         CALL Calc_eps_re
618         CALL Calc_tand
620         CALL Calc_eps_im
622         CALL Calc_errors(1)
624         CALL Calc_err_anal
626         Done=1
628         RETURN
630  !
632  ! ////////////////////////////////////////////////////////////
634  !
636 Calc_errs_freq:OFF KEY
638         Do_var_freq=1
640         CALL Calc_errors
642         Done=1
644         RETURN
646  !
648 ! ////////////////////////////////////////////////////////////
650 !
652 Menu_var_freq: !
654         CLEAR SCREEN
656         PRINT USING Image6;Sample_id$
658         PRINT USING Image6a;Eps_re_guess,"Method:Variable Frequency"
660         PRINT USING Image7;Samplelen*1.E+3,Usamplelen*1.E+3
662         PRINT
664         PRINT USING Image1;Degc,Udegc,Relh,Urelh
666         PRINT USING Image2;Mbar,Umbar
668         PRINT USING Image3;C,Uc
670         PRINT
672         PRINT USING Image4;Diameter*1.E+3,Udiameter*1.E+3
674         PRINT USING Image5;Length0*1.E+3,Ulength*1.E+3
676         PRINT
678         PRINT USING Image10ab
680         PRINT USING Image11b;Freq(1)/1.E+9,Ufreq(1)/1.E+9,Freq(2)/1.E+9,Ufreq
  => (2)/1.E+9
682         PRINT USING Image11a;Micr(1)*1.E+3,Umicr(1)*1.E+3,Micr(2)*1.E+3,Umicr
  => (2)*1.E+3
684         PRINT USING Image12ab;Q(1),Uq(1),Q(2),Uq(2)
686         PRINT USING Image13ab;Bw(1)/1.E+3,Ubw(1)/1.E+3,Bw(2)/1.E+3,Ubw(2)/1.E
  => +3,8
688         PRINT
690         PRINT USING Image15ab;Eps_re,Ueps_re,Eps_im,Ueps_im
692         PRINT USING Image16ab;Tand,Utand
694         RETURN
696 !
698 Image1:IMAGE "Temperature = ",2D.2D," +/- ",D.2D," deg C   Relative Humidity
  =>  = ",2D.D," +/- ",D.D," %"
700 Image2:IMAGE "Barometric Pressure = ",3D.D," +/- ",D.D,"   mbar"
702 Image3:IMAGE "Speed of Light       = ",D.7DESZ," +/- ",D.3DESZ," meters/secon
  => d"
704 Image4:IMAGE "Resonator Diameter                 =  ",2D.5D," +/- ",D.5D,"  mm"
706 Image5:IMAGE "             Length (micr. zero) = ",3D.5D," +/- ",D.5D,"  mm"
708 Image6:IMAGE "Sample Description: ",K,"                        "
710 Image6a:IMAGE "Initial guess for epsilon':",3D.3D,10X,K
712 Image7:IMAGE "          Length    = ",2D.4D," +/- ",D.4D," mm"
714 Image10ab:IMAGE 28X,"Empty",20X,"with Sample"
716 Image11a:IMAGE "Micrometer reading = ",2(S2D.4D,"   +/- ",D.4D,6X),"mm"
718 Image11b:IMAGE "Resonance frequency= ",2(2D.7D," +/- ",D.7D,3X),"GHz"
720 Image12ab:IMAGE "Q value                =   ",2(5D,"    +/- ",4D,10X)
722 Image13ab:IMAGE "3 dB Bandwidth      =  ",2(5D.2D,"   +/- ",3D.2D,6X),B,"kHz"
724 Image15ab:IMAGE "Epsilon'  = ",3D.4D," +/- ",D.4D,5X,"Epsilon''  = ",2D.5D,"
  =>  +/- ",D.5D
726 Image16ab:IMAGE "Tan delta = ",D.6D," +/- ",D.6D
728    SUBEND
730  !
732  ! ************************************************************
734  !
736    SUB Calc_dimens
738         OPTION BASE 1
```

```
740       COM /Environment/ Degc,Mbar,Relh,C
742       COM /Environment/ Udegc,Umbar,Urelh,Uc
744       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
746       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
748       COM /Sample/ Erreps(12),Errtand(12)
750       COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
752       COM /Cavity_dims/ Udiameter,Ulength
754       COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
756       COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
758       COM /Constants/ Bessel_root
760       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
 => Prompt$[80]
762       COM /Output/ INTEGER Print_addr,Plotter,Printer_on,Plotter_on
764       COM /Sys/ Sys_id$[10]
766       COM /Sys_msi/ Msi_id$[20]
768       COM /Files/ Diskdrive$[20],Filename$[10]
770       INTEGER Xpos,Ypos,N
772       REAL C0
774       C0=2.99792456E+8    !vacuum speed of light in meters/second
776    !
778       Diameter_temp=Diameter
780       Length0_temp=Length0
782 Calc_dimens:OFF KEY
784       GOSUB Menu_calc_dimen
786       DISP CHR$(129);" This is the menu for resonator dimensions calculatio
 => ns ";CHR$(128)
788       Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
790       ON KEY 0 LABEL "Print  Dimens",Prty GOSUB Print_dimens
792       ON KEY 1 LABEL "Cancel Results",Prty GOSUB Cancel_results
794       ON KEY 2 LABEL "Keep   Results",Prty CALL Prior_menu
796       ON KEY 3 LABEL "Add Del Mode",Prty GOSUB Add_mode
798       ON KEY 13,Prty GOSUB Delete_mode
800       ON KEY 4 LABEL "Chng  Tog Mode",Prty GOSUB Change_mode
802       ON KEY 14,Prty GOSUB Toggle_mode
804       ON KEY 6 LABEL "Init   Modes",Prty GOSUB Init_modes
806       ON KEY 7 LABEL "Root/ Micr.",Prty GOSUB Change_root
808       ON KEY 17,Prty GOSUB Micr_reading
810       ON KEY 8 LABEL "Disk   Modes",Prty GOSUB Disk_modes
812       ON KEY 9 LABEL "Find   Dimens",Prty GOSUB Find_dimens
814       Done=0
816       Prior_menu=0
818       LOOP
820           IF Done THEN GOTO Calc_dimens
822       EXIT IF Prior_menu
824       END LOOP
826       OFF KEY
828       Prior_menu=0
830       Done=1
832       SUBEXIT
834  !
836  ! ///////////////////////////////////////////////////////////////
838  !
840 Cancel_results:OFF KEY
842       Diameter=Diameter_temp
844       Length0=Length0_temp
846       Prior_menu=1
848       RETURN
850  !
852  ! ///////////////////////////////////////////////////////////////
854  !
856 Print_dimens:!
858       IF Printer_on THEN
860           PRINTER IS Print_addr
862           GOSUB Print_dimens2
864           PRINTER IS 1
866       END IF
868       GOSUB Print_dimens2
870       RETURN
872  !
874 Print_dimens2: !
876       PRINT "Filename$:";Filename$;
878       First=1
880       FOR I=1 TO Num_modes
882           IF Spectrum(I,5) THEN
884               IF First THEN
886                   PRINT USING "#,K,DD";"Modes:",Spectrum(I,1)
888                   First=0
890               ELSE
892                   PRINT USING "#,K,DD";",",Spectrum(I,1)
894               END IF
896           END IF
898       NEXT I
900       PRINT
902       PRINT USING "K,2D.4D,K,D.4D,K,3D.4D,K,D.4D";"     Diameter:",Diameter
 => *1.E+3,"+/-",Udiameter*1.E+3,"   Length:",Length0*1.E+3,"+/-",Ulength*1.E+3
904       RETURN
906  !
908  ! ///////////////////////////////////////////////////////////////
910  !
912 Menu_calc_dimen: !
914       CLEAR SCREEN
916       PRINT USING Image1;Degc,Udegc,Relh,Urelh
918       PRINT USING Image2;Mbar,Umbar
920       PRINT USING Image3;C,Uc
922       PRINT USING Image4;Diameter*1.E+3,Udiameter*1.E+3
924       PRINT USING Image5;Length0*1.E+3,Ulength*1.E+3
926       PRINT USING Image11a;Micr(2)*1.E+3,Umicr(2)*1.E+3
928       PRINT USING Image7c;Bessel_root
930       PRINT USING Image8c
932 Men_calc_dimen2: !
934       Xpos=0
936       Ypos=8
938       FOR I=1 TO Num_modes
940           Ypos=Ypos+1
942           PRINT TABXY(Xpos,Ypos);
944           IF Spectrum(I,5) THEN PRINT CHR$(129);
946           PRINT USING Image9c;Spectrum(I,1),Spectrum(I,2)/1.E+9
948           PRINT CHR$(128);
950           IF I=9 THEN
952               Xpos=35
954               Ypos=8
956               PRINT TABXY(Xpos,Ypos);
958               PRINT USING Image8c
960           END IF
962       NEXT I
964       RETURN
966 Image1:IMAGE "Temperature = ",2D.2D," +/- ",D.2D," deg C   Relative Humidity
 =>  = ",2D.D," +/- ",D.D," %"
968 Image2:IMAGE "Barometric Pressure = ",3D.D," +/- ",D.D,"   mbar"
970 Image3:IMAGE "Speed of Light      = ",D.7DESZ," +/- ",D.3DESZ," meters/secon
 => d"
972 Image4:IMAGE "Resonator Diameter              =  ",2D.5D," +/- ",D.5D,"  mm"
974 Image5:IMAGE "           Length (micr. zero) = ",3D.5D," +/- ",D.5D,"  mm"
976 Image7c:IMAGE "Bessel function root: ",2D.7D
978 Image8c:IMAGE "Axial mode  Frequency (GHz)"
980 Image9c:IMAGE X,2D,9X,2D.9D
982 Image11a:IMAGE "Micrometer reading = ",2(S2D.4D,"   +/- ",D.4D,6X),"mm"
984  !
986  ! ///////////////////////////////////////////////////////////////
988  !
990 Add_mode:OFF KEY
992       Num_modes=Num_modes+1
```

```
994          INPUT "Enter the value of the axial mode number",Spectrum(Num_modes,1
  => )
996          INPUT "Enter the value of the frequency (GHz).",Spectrum(Num_modes,2)
998          Spectrum(Num_modes,2)=Spectrum(Num_modes,2)*1.E+9
1000         INPUT "Enter the value of resonance Q",Spectrum(Num_modes,3)
1002         Spectrum(Num_modes,4)=.05   !Resonance level
1004         INPUT "Will this mode be included in calculations (y/n)",A$
1006         IF A$[1]="y" OR A$[1]="Y" THEN
1008             Spectrum(Num_modes,5)=1
1010         ELSE
1012             Spectrum(Num_modes,5)=0
1014         END IF
1016 !       Sort_modes(Spectrum(*), integer num_modes)
1018         MAT SORT Spectrum(*,1)
1020         Done=1
1022         RETURN
1024 !
1026 ! ////////////////////////////////////////////////////////////
1028 !
1030 Change_mode:OFF KEY
1032         INPUT "Enter the value of the axial mode you wish to change",Edit_num
1034         J=1
1036         WHILE (Spectrum(J,1)<>Edit_num) AND (J<Num_modes)
1038             J=J+1
1040         END WHILE
1042         Test=Spectrum(J,1)
1044         OUTPUT 2 USING "K,#";Test
1046         INPUT "Enter the value of the axial mode number",Spectrum(J,1)
1048         Test=Spectrum(J,2)/1.E+9
1050         OUTPUT 2 USING "K,#";Test
1052         INPUT "Enter the value of the frequency (GHz).",Spectrum(J,2)
1054         Spectrum(J,2)=Spectrum(J,2)*1.E+9
1056         Test=Spectrum(J,3)
1058         OUTPUT 2 USING "K,#";Test
1060         INPUT "Enter the resonance Q value",Spectrum(J,3)
1062         INPUT "Do you want to include this mode in calculations",A$
1064         IF A$[1]="y" OR A$[1]="Y" OR A$="" THEN
1066             Spectrum(J,5)=1
1068         ELSE
1070             Spectrum(J,5)=0
1072         END IF
1074         MAT SORT Spectrum(*,1)
1076 !       CALL Sort_modes(Spectrum(*), integer num_modes)
1078         Done=1
1080         RETURN
1082 !
1084 ! ////////////////////////////////////////////////////////////
1086 !
1088 Toggle_mode:!
1090         INPUT "Enter the mode number you wish to toggle",J
1092         FOR I=1 TO Num_modes
1094             IF Spectrum(I,1)=J THEN
1096                 IF Spectrum(I,5) THEN
1098                     Spectrum(I,5)=0
1100                 ELSE
1102                     Spectrum(I,5)=1
1104                 END IF
1106             END IF
1108         NEXT I
1110         Done=1
1112         RETURN
1114 !
1116 ! ////////////////////////////////////////////////////////////
1118 !
1120 Delete_mode:OFF KEY
1122         INPUT "Enter the value of the axial mode you wish to delete",Del_num
```

```
1124         FOR J=1 TO Num_modes
1126             IF Spectrum(J,1)=Del_num THEN
1128                 Spectrum(J,1)=1000
1130                 Spectrum(J,2)=0
1132                 Spectrum(J,3)=0
1134                 Spectrum(J,4)=0
1136                 Spectrum(J,5)=0
1138                 Hit=Hit+1
1140             END IF
1142 !               FOR K=J TO Num_modes
1144 !                   Spectrum(K,1)=Spectrum(K+1,1)
1146 !                   Spectrum(K,2)=Spectrum(K+1,2)
1148 !               NEXT K
1150 !               J=J-1
1152 !               Num_modes=Num_modes-1
1154 !           END IF
1156         NEXT J
1158         Num_modes=Num_modes-1
1160         MAT SORT Spectrum(*,1)
1162         Done=1
1164         RETURN
1166 !
1168 ! ////////////////////////////////////////////////////////////
1170 !
1172 Init_modes:OFF KEY
1174         DISP "Do you really wish to initialize the spectrum modes?"
1176         Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
1178         ON KEY 1 LABEL "Yes",Prty GOSUB Continue
1180         ON KEY 2 LABEL "No",Prty CALL Prior_menu
1182         Done=0
1184         Prior_menu=0
1186         LOOP
1188             IF Done THEN Quit
1190         EXIT IF Prior_menu
1192         END LOOP
1194         OFF KEY
1196         Prior_menu=0
1198         Done=1
1200 Quit: !
1202         RETURN
1204 Continue:MAT Spectrum= (0)
1206         FOR I=1 TO SIZE(Spectrum,1)
1208             Spectrum(I,1)=1000
1210         NEXT I
1212         Num_modes=0
1214         Done=1
1216         RETURN
1218 !
1220 ! ////////////////////////////////////////////////////////////
1222 !
1224 Change_root:OFF KEY
1226         DISP "Enter the value of the Bessel function root.";
1228         INPUT Bessel_root
1230         PRINT TABXY(0,7);
1232         PRINT USING Image7c;Bessel_root
1234         Done=1
1236         RETURN
1238 !
1240 ! ////////////////////////////////////////////////////////////
1242 !
1244 Disk_modes:OFF KEY
1246         CALL Select_disk("File with frequency/Q data")
1248         IF Diskdrive$="NO DISK" THEN GOTO Adm
1250         CALL Enterfilename("CAT","File with frequency/Q data")
1252         IF Filename$="" THEN
1254             CAT Diskdrive$
```

```
1256            CALL Enterfilename("ABORT","File with frequency/Q data")
1258        END IF
1260        ALLOCATE A(40,2),Id$[40]
1262        CALL Enter_file(A(*),Num_modes,Id$)
1264        IF A(1,1)>8.23E+9 THEN
1266            Ix=15
1268        ELSE
1270            Ix=14
1272        END IF
1273        Ic=0
1275        FOR I=1 TO Num_modes
1276            IF I>1 THEN
1277                IF PROUND(Spectrum(Ic,2),4)=PROUND(A(I,1),4) THEN GOTO Skip_s
  => pec_fill
1278            END IF
1279            Ic=Ic+1
1280            Spectrum(Ic,1)=Ic+Ix
1281            Spectrum(Ic,2)=A(I,1)
1282            Spectrum(Ic,3)=A(I,2)
1283 !          IF I+Ix=27 THEN
1284 !              Spectrum(I,5)=0
1286 !          ELSE
1288            Spectrum(Ic,5)=1
1290 !          END IF
1291 Skip_spec_fill:!
1293        NEXT I
1294        Num_modes=Ic
1296        DEALLOCATE Id$,A(*)
1297 Adm:   Done=1
1298        RETURN
1300 !
1302 ! ////////////////////////////////////////////////////////////////
1304 !
1306 Find_dimens:OFF KEY
1308        J=0
1310        FOR I=1 TO Num_modes
1312            IF Spectrum(I,5) THEN J=J+1
1314        NEXT I
1316        ALLOCATE Spec(1:J,2)
1318        J=0
1320        FOR I=1 TO Num_modes
1322            IF Spectrum(I,5) THEN J=J+1
1324            IF Spectrum(I,5) THEN MAT Spec(J,*)= Spectrum(I,1:2)
1326        NEXT I
1328        CALL Find_dims(Spec(*),Length0,Diameter,Ulength,Udiameter,Micr(2),C,B
  => essel_root,J)
1330        DEALLOCATE Spec(*)
1332        PRINT TABXY(0,4);
1334        PRINT USING Image4;Diameter*1.E+3,Udiameter*1.E+3
1336        PRINT TABXY(0,5);
1338        PRINT USING Image5;Length0*1.E+3,Ulength*1.E+3
1340 !      PRINT TABXY(0,15);B(1,1);B(2,1)
1342 !      PRINT "COVARIANCE MATRIX"
1344 !      PRINT Cvm(1,1),Cvm(1,2)
1346 !      PRINT Cvm(2,1),Cvm(2,2)
1348        Done=1
1350        RETURN
1352 !
1354 ! ////////////////////////////////////////////////////////////////
1356 !
1358 Micr_reading:OFF KEY
1360        CALL Edit_data("the micrometer reading of the cavity (mm).",Micr(2),1
  => .E+3,Umicr(2))
1362        PRINT TABXY(0,6);
1364        PRINT USING Image11a;Micr(2)*1.E+3,Umicr(2)*1.E+3
1366        Done=1
1368        RETURN
1370   SUBEND
1372 !
1374 ! ******************************************************************
1376 !
1378   SUB Calc_c
1380 !
1382 ! Equations from H. Liebe in "An Updated Model for Millimeter Wave
1384 ! Propagation in Moist Air
1386 ! RADIO SCIENCE, Vol. 20, Num 5, pp.1069-1089, Sept/Oct 1985
1388 !
1390 Calc_c:OFF KEY
1392        COM /Environment/ Degc,Mbar,Relh,C
1394        COM /Environment/ Udegc,Umbar,Urelh,Uc
1396        COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
1398   !
1400        GOSUB Menu_calc_c
1402   !
1404        Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
1406        ON KEY 1 LABEL "Prior  Menu",Prty CALL Prior_menu
1408        ON KEY 2 LABEL "Change Temp",Prty CALL Temp
1410        ON KEY 3 LABEL "Change Rel Hum",Prty CALL Rel_hum
1412        ON KEY 4 LABEL "Change Presure",Prty CALL Pressure
1414        Prior_menu=0
1416        Done=0
1418        LOOP
1420            IF Done THEN GOTO Calc_c
1422        EXIT IF Prior_menu
1424        END LOOP
1426        Prior_menu=0
1428        Done=1
1430        SUBEXIT
1432 !
1434 Menu_calc_c:!
1436        CLEAR SCREEN
1438        PRINT USING Image1;Degc,Udegc,Relh,Urelh
1440        PRINT USING Image2;Mbar,Umbar
1442        PRINT USING Image3;C,Uc
1444        DISP "Menu for speed of light calculation"
1446        RETURN
1448 Image1:IMAGE "Temperature = ",2D.2D," +/- ",D.2D," deg C   Relative Humidit
  => y = ",2D.D," +/- ",D.D," %"
1450 Image2:IMAGE "Barometric Pressure = ",3D.D," +/- ",D.D,"   mbar"
1452 Image3:IMAGE "Speed of Light      = ",D.7DESZ," +/- ",D.3DESZ," meters/seco
  => nd"
1454   SUBEND
1456 !
1458 ! ******************************************************************
1460 !
1462   SUB Calc_modes
1464 Calc_modes:OFF KEY
1466   !
1468        OPTION BASE 1
1470        COM /Environment/ Degc,Mbar,Relh,C
1472        COM /Environment/ Udegc,Umbar,Urelh,Uc
1474        COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
1476        COM /Cavity_dims/ Udiameter,Ulength
1478        COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
1480   !
1482        LOADSUB Modefreqs2 FROM "MODEFREQS2"
1484        CALL Modefreqs2(C,Diameter/2,Length0)
1486        DELSUB Modefreqs2
1488        Prior_menu=0
1490        Done=1
```

```
1492   SUBEND
1494 !
1496 ! ************************************************************
1498 !
1500   SUB Calc_freqs
1502 Calc_freqs:OFF KEY
1504   !
1506       OPTION BASE 1
1508       COM /Environment/ Degc,Mbar,Relh,C
1510       COM /Environment/ Udegc,Umbar,Urelh,Uc
1512       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
1514       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
1516       COM /Sample/ Erreps(12),Errtand(12)
1518       COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
1520       COM /Cavity_dims/ Udiameter,Ulength
1522       COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
1524       COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
1526       COM /Constants/ Bessel_root
1528       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
1530   !
1532       GOSUB Menu_calc_freqs
1534   !
1536       Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
1538       ON KEY 1 LABEL "Prior  Menu",Prty CALL Prior_menu
1540       ON KEY 2 LABEL "Sample Thickns",Prty CALL Sample_thickns
1542       ON KEY 3 LABEL "Sample Epsilon",Prty CALL Epsilon
1544       ON KEY 4 LABEL "Cavity Diameter",Prty CALL Diameter
1546       ON KEY 5 LABEL "Cavity Length ",Prty CALL Length
1548       ON KEY 6 LABEL "Micr.  Reading",Prty CALL Micr_readings
1550       ON KEY 7 LABEL "Freq.  Empty",Prty CALL Freq_empty
1552       ON KEY 8 LABEL "Freq.  Loaded",Prty CALL Freq_loaded
1554       Prior_menu=0
1556       Done=0
1558       LOOP
1560         IF Done THEN GOTO Calc_freqs
1562       EXIT IF Prior_menu
1564       END LOOP
1566       OFF KEY
1568       Prior_menu=0
1570       Done=1
1572       SUBEXIT
1574 !
1576 ! ////////////////////////////////////////////////////////////
1578 !
1580 Menu_calc_freqs: !
1582       CLEAR SCREEN
1584       PRINT USING Image6;S_id$
1586       PRINT USING Image7;Samplelen*1.E+3,Usamplelen*1.E+3
1588       PRINT USING Image17;Eps_re,Ueps_re
1590       PRINT
1592       PRINT USING Image1;Degc,Udegc,Relh,Urelh
1594       PRINT USING Image2;Mbar,Umbar
1596       PRINT USING Image3;C,Uc
1598       PRINT
1600       PRINT USING Image4;Diameter*1.E+3,Udiameter*1.E+3
1602       PRINT USING Image5;Length0*1.E+3,Ulength*1.E+3
1604       PRINT USING Image5a;(Length0+Micr(1))*1.E+3,Ulength
1606       PRINT
1608       PRINT USING Image10ab
1610       PRINT USING Image11a;Micr(1)*1.E+3,Umicr(1)*1.E+3,Micr(2)*1.E+3,Umicr
  => (2)*1.E+3
1612       PRINT
1614       RETURN
1616 Image1:IMAGE "Temperature = ",2D.2D," +/- ",D.2D," deg C   Relative Humidit
  => y = ",2D.D," +/- ",D.D," %"
```

```
1618 Image2:IMAGE "Barometric Pressure = ",3D.D," +/- ",D.D,"   mbar"
1620 Image3:IMAGE "Speed of Light      = ",D.7DESZ," +/- ",D.3DESZ," meters/seco
  => nd"
1622 Image4:IMAGE "Resonator Diameter                = ",2D.5D," +/- ",D.5D,"  mm"
1624 Image5:IMAGE "           Length (micr. zero) = ",3D.5D," +/- ",D.5D,"  mm"
1626 Image5a:IMAGE"           Length (total)      = ",3D.5D," +/- ",D.5D,"  mm"
1628 Image6:IMAGE "Sample Description: ",K,"                    "
1630 Image7:IMAGE "        Length    = ",2D.4D," +/- ",D.4D," mm"
1632 Image10ab:IMAGE 28X,"Empty",20X,"with Sample"
1634 Image11a:IMAGE "Micrometer reading = ",2(3D.4D,"   +/- ",D.4D,6X),"mm"
1636 Image17:IMAGE "        Epsilon'  =",3D.4D," +/- ",D.4D
1638   SUBEND
1640 !
1642 ! **********************************************************
1644 !
1646   SUB Sample_descr
1648 Sample_descr:OFF KEY
1650   !
1652       OPTION BASE 1
1654       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
1656       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
1658       COM /Sample/ Erreps(12),Errtand(12)
1660       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
1662   !
1664       LOADSUB Enter_id FROM "BAS_SUBS:,1400,0,3"
1666       Enter_id(Sample_id$,"Sample")
1668       DELSUB Enter_id
1670       Done=1
1672   SUBEND
1674 !
1676 ! **********************************************************
1678 !
1680   SUB Sample_thickns
1682 Sample_thickns:OFF KEY
1684   !
1686       OPTION BASE 1
1688       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
1690       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
1692       COM /Sample/ Erreps(12),Errtand(12)
1694       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
1696   !
1698       CALL Edit_data("the sample length (mm).",Samplelen,1.E+3,Usamplelen)
1700       Done=1
1702   SUBEND
1704 !
1706 ! **********************************************************
1708 !
1710   SUB Resnce_freq
1712 Resnce_freq:OFF KEY
1714   !
1716       OPTION BASE 1
1718       COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
1720       COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
1722       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
1724   !
1726       CALL Edit_data("the resonance frequency (GHz)",Freq(2),1.E-9)
1728       CALL Edit_data("the uncertainty in the resonance frequency (kHz)",Ufr
  => eq(2),1.E-3)
1730       Freq(1)=Freq(2)
1732       Ufreq(1)=Ufreq(2)
1734       IF Bw(1)>0 THEN
1736         Q(1)=Freq(1)/Bw(1)
1738         Uq(1)=Q(1)-Freq(1)/(Bw(1)+Ubw(1))
```

```
1740        END IF
1742        IF Bw(2)>0 THEN
1744            Q(2)=Freq(2)/Bw(2)
1746            Uq(2)=Q(2)-Freq(2)/(Bw(2)+Ubw(2))
1748        END IF
1750        Done=1
1752    SUBEND
1754 !
1756 ! ************************************************************
1758 !
1760    SUB Micr_readings
1762 Micr_readings:OFF KEY
1764    !
1766        OPTION BASE 1
1768        COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
1770        COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
1772        COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
1774    !
1776        Prompt$="Do you want to change the EMPTY or LOADED cavity micrometer
  => reading?"
1778        CALL Empty_or_loaded
1780        IF Prior_menu THEN
1782            Prior_menu=0
1784            Done=1
1786            SUBEXIT
1788        END IF
1790 !
1792        IF Empty_cavity THEN
1794            CALL Edit_data("the micrometer reading for the empty cavity (mm).
  => ",Micr(1),1.E+3,Umicr(1))
1796        ELSE
1798            CALL Edit_data("the micrometer reading for the loaded cavity (mm)
  => .",Micr(2),1.E+3,Umicr(2))
1800        END IF
1802        GOTO Micr_readings
1804    SUBEND
1806 !
1808 ! ************************************************************
1810 !
1812    SUB Q_values
1814 Q_values:OFF KEY
1816    !
1818        OPTION BASE 1
1820        COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
1822        COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
1824        COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
1826    !
1828        Prompt$="Do you want to change the EMPTY or LOADED cavity Q-value?"
1830        CALL Empty_or_loaded
1832        IF Prior_menu THEN
1834            Prior_menu=0
1836            Done=1
1838            SUBEXIT
1840        END IF
1842        IF Empty_cavity THEN
1844 !          CALL Edit_data("Q for the empty cavity.",Q(1),1,Uq(1))
1845            CALL Edit_data("Q for the empty cavity.",Q(1),1)
1846            Uq(1)=Q(1)/100
1848            Bw(1)=Freq(1)/Q(1)
1849            Ubw(1)=Bw(1)-Freq(1)/(Q(1)+Uq(1))
1850        ELSE
1852 !          CALL Edit_data("Q for the loaded cavity.",Q(2),1,Uq(2))
1853            CALL Edit_data("Q for the loaded cavity.",Q(2),1)
1854            Uq(2)=Q(2)/100
```

```
1856            Bw(2)=Freq(2)/Q(2)
1857            Ubw(2)=Bw(2)-Freq(2)/(Q(2)+Uq(2))
1858        END IF
1860        GOTO Q_values
1862    SUBEND
1864 !
1866 ! ************************************************************
1868 !
1870    SUB Band_widths
1872 Band_widths:OFF KEY
1874    !
1876        OPTION BASE 1
1878        COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
1880        COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
1882        COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
1884    !
1886        Prompt$="Do you want to change the bandwidth of the EMPTY or LOADED c
  => avity?"
1888        CALL Empty_or_loaded
1890        IF Prior_menu THEN
1892            Prior_menu=0
1894            Done=1
1896            SUBEXIT
1898        END IF
1900 !
1902        IF Empty_cavity THEN
1904            CALL Edit_data("of the 3dB bandwidth for the empty cavity (kHz)."
  => ,Bw(1),1.E-3,Ubw(1))
1906            Q(1)=Freq(1)/Bw(1)
1908            Uq(1)=ABS(Q(1)-Freq(1)/(Bw(1)+Ubw(1)))
1910        ELSE
1912            CALL Edit_data("of the 3dB bandwidth for the loaded cavity (kHz).
  => ",Bw(2),1.E-3,Ubw(2))
1914            Q(2)=Freq(2)/Bw(2)
1916            Uq(2)=ABS(Q(2)-Freq(2)/(Bw(2)+Ubw(2)))
1918        END IF
1920        GOTO Band_widths
1922    SUBEND
1924 !
1926 ! ************************************************************
1928 !
1930    SUB Resnce_freqs
1932    !
1934        OPTION BASE 1
1936        COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
1938        COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
1940        COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
1942    !
1944 Resnce_freqs:OFF KEY
1946        Prompt$="Do you want to change the EMPTY or LOADED cavity resonance f
  => requency?"
1948        CALL Empty_or_loaded
1950        IF Prior_menu THEN
1952            Prior_menu=0
1954            Done=1
1956            SUBEXIT
1958        END IF
1960        IF Empty_cavity THEN
1962            CALL Edit_data("the empty cavity resonance frequency (GHz).",Freq
  => (1),1.E-9)
1964            CALL Edit_data("the uncertainty in the empty cavity res. freq. (k
  => Hz)",Ufreq(1),1.E-3)
1966            IF Bw(1)>0 THEN
1968                Q(1)=Freq(1)/Bw(1)
```

```
1970             Uq(1)=Q(1)-Freq(1)/(Bw(1)+Ubw(1))
1972          END IF
1974       ELSE
1976          CALL Edit_data("the loaded cavity resonance frequency (GHz).",Fre
  => q(2),1.E-9)
1978          CALL Edit_data("the uncertainty in the loaded cavity res. freq. (
  => kHz).",Ufreq(2),1.E-3)
1980          IF Bw(2)>0 THEN
1982             Q(2)=Freq(2)/Bw(2)
1984             Uq(2)=Q(2)-Freq(2)/(Bw(2)+Ubw(2))
1986          END IF
1988       END IF
1990       GOTO Resnce_freqs
1992    SUBEND
1994 !
1996 ! ****************************************************************
1998 !
2000    SUB Micr_reading
2002 Micr_reading:OFF KEY
2004    !
2006       OPTION BASE 1
2008       COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
2010       COM /Cavity_vals/ Ufreq(2),Umicr(2),Uc(2),Ubw(2)
2012       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2014    !
2016       CALL Edit_data("the micrometer reading (mm).",Micr(1),1.E+3,Umicr(1))
2018       Micr(2)=Micr(1)
2020       Umicr(2)=Umicr(1)
2022       Done=1
2024    SUBEND
2026 !
2028 ! ****************************************************************
2030 !
2032    SUB Calc_eps_re
2034 !
2036 ! Equations from F. Horner, T.A. Taylor, R. Dunsmuir in "Resonance
2038 ! Methods of Dielectric Measurement at Centimetre Wavelengths"
2040 ! J. IEE, 1946, 93, Pt.III, pp.53-68
2042 !
2044 Calc_eps_re:OFF KEY
2046    !
2048       RAD
2050       OPTION BASE 1
2052       COM /Environment/ Degc,Mbar,Relh,C
2054       COM /Environment/ Udegc,Umbar,Urelh,Uc
2056       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
2058       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
2060       COM /Sample/ Erreps(12),Errtand(12)
2062       COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
2064       COM /Cavity_dims/ Udiameter,Ulength
2066       COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
2068       COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
2070       COM /Constants/ Bessel_root
2072    !
2074       Y=Length0+Micr(2)-Samplelen
2076       Mu=(4*PI)*10^(-7)
2078       Eps_air=1/(Mu*(C^2))
2080       Eps0=8.854188E-12
2082       W=2*PI*Freq(2)
2084       K=Bessel_root/(Diameter/2)
2086       Eps_re=Eps_re_guess
2088       B0=SQRT(W^2*Mu*Eps_air-K^2)
2090       B1=SQRT(W^2*Mu*Eps_re*Eps0-K^2)
2092       REPEAT
2094          I=I+1
2096          Funct_b1=TAN(B1*Samplelen)/B1+TAN(B0*Y)/B0
2098          Funct_b1_p=(B1*Samplelen*(1/COS(B1*Samplelen)^2)-TAN(B1*Samplelen
  => ))/(B1^2)
2100          Deltab1=Funct_b1/Funct_b1_p
2102          IF ABS(Deltab1)<1.E-2 THEN Deltab1=Deltab1/10
2104          B1=B1-Deltab1
2106 !        PRINT Deltab1
2108       UNTIL (I=100) OR (ABS(Deltab1)<1.E-4)
2110       Eps_re=(B1^2+K^2)/(B0^2+K^2)
2112 !     PRINT Eps_re
2114 !     PAUSE
2116       IF I=100 THEN Eps_re=0
2118       DEG
2120    SUBEND
2122    !
2124    ! ****************************************************************
2126    !
2128    SUB Calc_tand
2130 !
2132 ! Equations from E. Ni and U. Stumper "Permittivity Measurements Using a
2134 ! Frequency-tuned Microwave TE01 Cavity Resonator"
2136 ! IEE PROCEEDINGS, Vol. 132, Pt. H, No. 1, February 1985 p27-32.
2138 !
2140 Calc_tand:OFF KEY
2142    !
2144       RAD
2146       OPTION BASE 1
2148       COM /Environment/ Degc,Mbar,Relh,C
2150       COM /Environment/ Udegc,Umbar,Urelh,Uc
2152       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
2154       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
2156       COM /Sample/ Erreps(12),Errtand(12)
2158       COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
2160       COM /Cavity_dims/ Udiameter,Ulength
2162       COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
2164       COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
2166       COM /Constants/ Bessel_root
2168       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2170    !
2172       IF Do_var_freq THEN       !Variable frequency/ fixed length
2174          L0=Length0+Micr(1)
2176       ELSE                      !Variable length/ fixed frequency
2178          L0=Length0+Micr(2)
2180       END IF
2182       L=Length0+Micr(2)
2184       A=Diameter/2
2186       Fc=(Bessel_root*C)/(PI*Diameter)
2188       Q0=Fc/Freq(2)
2190       B0=2*PI*Freq(2)*SQRT(1-Q0^2)/C
2192       Be=2*PI*Freq(2)*SQRT(Eps_re-Q0^2)/C
2194       B=(Be/B0)^2
2196       X=Be*Samplelen
2198       P=(SIN(X)^2)+(B*(COS(X)^2))
2200       G=(2*Be*A*((Eps_re-Q0^2)+(1-Q0^2)*P))/(2*X-SIN(2*X))
2202       F=(Q0^2*(2*X+(B-1)*SIN(X)+(2*Be*L-2*X)*P))/(2*X-SIN(2*X))
2204       E=Eps_re*(Q0^2+2*A*(1-Q0^2)/L0)
2206       D=(F+G)/E
2208       A=(2*X*Eps_re+(B-Eps_re)*SIN(2*X)+(2*Be*L-2*X)*P)/(Eps_re*(2*X-SIN(2*
  => X)))
2210       Tand=(A*Bw(2)-D*Bw(1))/Freq(2)
2212       DEG
2214    SUBEND
2216 !
2218 ! ****************************************************************
2220 !
```

```
2222   SJB Calc_eps_im
2224 Calc_eps_im:OFF KEY
2226   !
2228       RAD
2230       OPTION BASE 1
2232       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_in,Tand
2234       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
2236       COM /Sample/ Erreps(12),Errtand(12)
2238   !
2240       Eps_im=Tand*Eps_re
2242       DEG
2244   SJBEND
2246 !
2248 ! ************************************************************
2250 !
2252   SJB Edit_data(Prompt$,Variable,OPTIONAL Multiplier,Uvariable)
2254 Edit_data:OFF KEY
2256       IF NPAR>2 THEN
2258           Test=Variable*Multiplier
2260           IF NPAR=4 THEN Uvariable=Uvariable*Multiplier
2262       ELSE
2264           Test=Variable
2266       END IF
2268       ON ERROR GOTO Test_again
2270 Test_again:!
2272       OUTPUT 2 USING "K,#";Test
2274       DISP "Enter the value of ";Prompt$;
2276       INPUT Variable
2278       OFF ERROR
2280       IF NPAR=4 THEN
2282           Utest=Uvariable
2284           ON ERROR GOTO Utest_again
2286 Utest_again:!
2288           OUTPUT 2 USING "K,#";Utest
2290           DISP "Enter the uncertainty in ";Prompt$;
2292           INPUT Uvariable
2294           OFF ERROR
2296       END IF
2298       IF NPAR>2 THEN
2300           Variable=Variable/Multiplier
2302           IF NPAR=4 THEN Uvariable=Uvariable/Multiplier
2304       END IF
2306   SJBEND
2308   !
2310   ! ************************************************************
2312   !
2314   SJB Speed_of_light
2316 Speed_of_light:!
2318   !
2320       COM /Environment/ Degc,Mbar,Relh,C
2322       COM /Environment/ Udegc,Umbar,Urelh,Uc
2324   !
2326       Theta=300/(273.15+Degc)
2328       Esaturated=Theta^5/(.4151*10^(9.834*Theta-10))
2330       E=Relh*Esaturated/100
2332       Kpa=Mbar/10
2334       Pdryair=Kpa-E
2336       N=1+((((41.6*Theta)+2.39)*E*Theta)+(2.588*Pdryair*Theta))*10^(-6)
2338       C=2.99792458*10^8/N
2340   SJBEND
2342 !
2344 ! ************************************************************
2346 !
2348   SUB Temp
2350 Temo:   !
2352   !
2354       COM /Environment/ Degc,Mbar,Relh,C
2356       COM /Environment/ Udegc,Umbar,Urelh,Uc
2358       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2360   !
2362       CALL Edit_data("the temperature in the lab (C).",Degc,1,Udegc)
2364       CALL Speed_of_light
2366       CALL Uncert_c
2368       Done=1
2370   SUBEND
2372 !
2374 ! ************************************************************
2376 !
2378   SUB Rel_hum
2380 Rel_hum: !
2382   !
2384       COM /Environment/ Degc,Mbar,Relh,C
2386       COM /Environment/ Udegc,Umbar,Urelh,Uc
2388       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2390   !
2392       CALL Edit_data("the relative humidity in the lab. (%)",Relh,1,Urelh)
2394       CALL Speed_of_light
2396       CALL Uncert_c
2398       Done=1
2400   SUBEND
2402 !
2404 ! ************************************************************
2406 !
2408   SUB Pressure
2410 Pressure: !
2412   !
2414       COM /Environment/ Degc,Mbar,Relh,C
2416       COM /Environment/ Udegc,Umbar,Urelh,Uc
2418       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2420   !
2422       CALL Edit_data("the pressure in the lab (millibars).",Mbar,1,Umbar)
2424       CALL Speed_of_light
2426       CALL Uncert_c
2428       Done=1
2430   SUBEND
2432 !
2434 ! ************************************************************
2436 !
2438   SUB Uncert_c
2440 Uncert_c: !
2442   !
2444       COM /Environment/ Degc,Mbar,Relh,C
2446       COM /Environment/ Udegc,Umbar,Urelh,Uc
2448   !
2450       C0=2.99792458*10^(8)
2452       T=Degc+273.15
2454       P=Mbar/10
2456       H=Relh
2458       Ut=Udegc
2460       Up=Umbar/10
2462       Uh=Urelh
2464       A=(300/T)^7
2466       B=(300/T)^6
2468       D=10^((9.834*300/T)-10)
2470       N=1+10^(-6)*((41.6*A*H+2.39*B*H)/(41.51*D)+2.588*P*(300/T)-((2.588*B*
  => H)/(41.51*D)))
2472       Temp1=41.51*D*9.834*300*LOG(10)*(-1/(T^2))*(41.6*A*H-.198*B*H)
2474       Temp2=41.51*D*(41.6*(300^7)*(-7)*(T^(-8))*H-.198*(300^6)*(-6)*(T^(-7)
  => )*H)
```

```
2476        Temp3=2.588*P*300*(-1/(T^2))
2478        Par_n_res_t=(((Temp1-Temp2)/((41.51*D)^2))+Temp3)*10^(-6)
2480        Par_n_res_p=(300/T)*2.588*10^(-6)
2482        Par_n_res_h=((41.6*A-.198*B)/(41.51*D))*10^(-6)
2484        Uncert_n=SQRT(((Par_n_res_t*Ut)^2)+((Par_n_res_p*Up)^2)+((Par_n_res_h
  => *Uh)^2))
2486        Uc=C0*(1/(N^2))*Uncert_n
2488    SUBEND
2490 !
2492 ! ************************************************************
2494 !
2496    SUB Freq_empty
2498 Freq_empty:OFF KEY
2500    !
2502        OPTION BASE 1
2504        COM /Environment/ Degc,Mbar,Relh,C
2506        COM /Environment/ Udegc,Umbar,Urelh,Uc
2508        COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
2510        COM /Cavity_dims/ Udiameter,Ulength
2512        COM /Constants/ Bessel_root
2514        COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
2516        COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
2518        COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2520    !
2522        Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
2524        ON KEY 1 LABEL "Prior  Menu",Prty CALL Prior_menu
2526        ON KEY 2 LABEL "Start  Freq",Prty GOSUB Start_freq
2528        ON KEY 3 LABEL "Stop   Freq",Prty GOSUB Stop_freq
2530        ON KEY 4 LABEL "Calc   Modes",Prty GOSUB Calc_freq_empty
2532        Prior_menu=0
2534        Done=0
2536        LOOP
2538            IF Done THEN GOTO Freq_empty
2540        EXIT IF Prior_menu
2542        END LOOP
2544        OFF KEY
2546        Prior_menu=0
2548        Done=1
2550        SUBEXIT
2552 !
2554 Start_freq: !
2556        Edit_data("the value of the starting frequency",Start_freq,1.E-9)
2558        Done=1
2560        RETURN
2562 !
2564 Stop_freq: !
2566        Edit_data("the value of the stopping frequency",Stop_freq,1.E-9)
2568        Done=1
2570        RETURN
2572 !
2574 Calc_freq_empty: !
2576        PRINT TABXY(0,16);
2578 Image11b:IMAGE "TE01(",2D,") mode occurs at ",2D.7D,"GHz"
2580        B=Bessel_root/(PI*Diameter)
2582        FOR My=1 TO 50
2584            E=My/(2*(Length0+Micr(1)))
2586            E=E^2+B^2
2588            E=SQR(E)
2590            Frequency=C*E
2592            IF Frequency>Stop_freq THEN GOTO Resend
2594            IF Frequency<Start_freq THEN GOTO Respaths
2596            PRINT USING Image11b;My,Frequency/1.E+9
2598 Respaths: !
2600        NEXT My
2602 Resend:   !
```

```
2604        Done=1
2606        RETURN
2608    SUBEND
2610 !
2612 ! ************************************************************
2614 !
2616    SUB Freq_loaded
2618 Freq_loaded:OFF KEY
2620    !
2622        RAD
2624        OPTION BASE 1
2626        COM /Environment/ Degc,Mbar,Relh,C
2628        COM /Environment/ Udegc,Umbar,Urelh,Uc
2630        COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
2632        COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
2634        COM /Sample/ Erreps(12),Errtand(12)
2636        COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
2638        COM /Cavity_dims/ Udiameter,Ulength
2640        COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
2642        COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
2644        COM /Constants/ Bessel_root
2646    !
2648        Edit_data("the initial guess for the resonance frequency (GHz).",Freq
  => (2),1.E-9)
2650        B=Samplelen
2652        Y=Length0+Micr(2)-B
2654        Mu=(4*PI)*10^(-7)
2656        Eps0=1/(Mu*(C^2))
2658        K=Bessel_root/(Diameter/2)
2660        REPEAT
2662            I=I+1
2664            W=2*PI*Freq(2)
2666            B0=SQRT(W^2*Mu*Eps0-K^2)
2668            B1=SQRT(W^2*Mu*Eps_re*Eps0-K^2)
2670            Fun_f=TAN(B1*B)/B1+TAN(B0*Y)/B0
2672            Fun_f_p=4*PI^2*Mu*Eps0*Freq(2)*(Eps_re*(B*SQR(B1)*(1/COS(B*B1)^2)
  => -TAN(B*B1)*(1/SQR(B1)))/(B1^2)+(Y*SQR(B0)*(1/COS(Y*B0)^2)-TAN(Y*B0)*(1/SQR(
  => B0)))/(B0^2))
2674            Freq(2)=Freq(2)-(Fun_f/Fun_f_p)
2676        UNTIL (I=100000) OR (ABS(Fun_f/Fun_f_p)<1.E-5)
2678        W=2*PI*Freq(2)
2680        B0=SQRT(W^2*Mu*Eps0-K^2)
2682        B1=SQRT(W^2*Mu*Eps_re*Eps0-K^2)
2684        Halfwavelens1=B1*B/PI
2686        Halfwavelens0=B0*Y/PI
2688        Halfwavelens=Halfwavelens0+Halfwavelens1
2690 Image11b:IMAGE "Resonant mode ",2D.2D," occurs at ",2D.7D," +/- ",D.7D,3X,"
  => GHz"
2692        PRINT TABXY(0,16);
2694        PRINT USING Image11b;Halfwavelens,Freq(2)/1.E+9,Ufreq(2)/1.E+9
2696        PRINT "                                                        "
2698        PRINT "                                                        "
2700        PRINT "                                                        "
2702        DEG
2704    SUBEND
2706 !
2708 ! ************************************************************
2710 !
2712    SUB Epsilon
2714 Epsilon:OFF KEY
2716    !
2718        OPTION BASE 1
2720        COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
2722        COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
2724        COM /Sample/ Erreps(12),Errtand(12)
2726        COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
```

```
  => Prompt$[80]
2728   !
2730       CALL Edit_data("the relative epsilon of the sample.",Eps_re,1)
2732       Done=1
2734   SUBEND
2736 !
2738 ! ************************************************************
2740 !
2742   SUB Diameter
2744 Diameter:OFF KEY
2746   !
2748       OPTION BASE 1
2750       COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
2752       COM /Cavity_dims/ Udiameter,Ulength
2754       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2756   !
2758       CALL Edit_data("the diameter of the cavity (mm).",Diameter,1.E+3,Udia
  => meter)
2760       Done=1
2762   SUBEND
2764 !
2766 ! ************************************************************
2768 !
2770   SUB Length
2772 Length:OFF KEY
2774   !
2776       OPTION BASE 1
2778       COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
2780       COM /Cavity_dims/ Udiameter,Ulength
2782       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2784   !
2786       CALL Edit_data("the length of the zeroed cavity (mm).",Length0,1.E+3,
  => Ulength)
2788       Done=1
2790   SUBEND
2792 !
2794 ! ************************************************************
2796 !
2798   SUB Sort_modes(Array(*),N)
2800       IF N=1 THEN SUBEXIT
2802       FOR J=2 TO N
2804           Dummy1=Array(J,1)
2806           Dummy2=Array(J,2)
2808           FOR I=J-1 TO 1 STEP -1
2810               IF Array(I,1)<=Dummy1 THEN GOTO Mark
2812               Array(I+1,1)=Array(I,1)
2814               Array(I+1,2)=Array(I,2)
2816           NEXT I
2818           I=0
2820 Mark:  !
2822           Array(I+1,1)=Dummy1
2824           Array(I+1,2)=Dummy2
2826       NEXT J
2828   SUBEND
2830 !
2832 ! ************************************************************
2834 !
2836   SUB Empty_or_loaded
2838 Empty_or_loaded:OFF KEY
2840   !
2842       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2844   !
2846       DISP Prompt$
2848       Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
2850       ON KEY 1 LABEL "PRIOR  MENU",Prty CALL Prior_menu
2852       ON KEY 2 LABEL "EMPTY  CAVITY",Prty GOSUB Empty
2854       ON KEY 3 LABEL "LOADED CAVITY",Prty GOSUB Loaded
2856       Done=0
2858       Prior_menu=0
2860       LOOP
2862       EXIT IF Prior_menu OR Done
2864       END LOOP
2866       SUBEXIT
2868 !
2870 ! ////////////////////////////////////////////////////////////
2872 !
2874 Empty:OFF KEY
2876       Empty_cavity=1
2878       Done=1
2880       RETURN
2882 !
2884 ! ////////////////////////////////////////////////////////////
2886 !
2888 Loaded:OFF KEY
2890       Empty_cavity=0
2892       Done=1
2894       RETURN
2896   SUBEND
2898 !
2900 ! ************************************************************
2902 !
2904   SUB Prior_menu
2906       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2908       Prior_menu=1
2910   SUBEND
2912 !
2914 ! ************************************************************
2916 !
2918   SUB Calc_errors(OPTIONAL INTEGER Dont_print)
2920 !
2922 !
2924 Calc_errors:OFF KEY
2926   !
2928       OPTION BASE 1
2930       COM /Environment/ Degc,Mbar,Relh,C
2932       COM /Environment/ Udegc,Umbar,Urelh,Uc
2934       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
2936       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
2938       COM /Sample/ Erreps(12),Errtand(12)
2940       COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
2942       COM /Cavity_dims/ Udiameter,Ulength
2944       COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
2946       COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
2948       COM /Constants/ Bessel_root
2950       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
2951       Print_addr=9
2953 ! !
2954       DIM Eps(12),Tandel(12)
2956       Diameter=Diameter+Udiameter
2958       CALL Calc_eps_re
2960       Eps(1)=Eps_re
2962       Diameter=Diameter-Udiameter
2964   !
2966       Length0=Length0+Ulength
2968       CALL Calc_eps_re
2970       Eps(2)=Eps_re
2972       Length0=Length0-Ulength
```

```
2974   !
2976        Samplelen=Samplelen+Usamplele
2978        CALL Calc_eps_re
2980        Eps(3)=Eps_re
2982        Samplelen=Samplelen-Usamplelen
2984   !
2986        Degc=Degc+Udegc
2988        CALL Speed_of_light
2990        CALL Calc_eps_re
2992        Degc=Degc-Udegc
2994        Eps(4)=Eps_re
2996   !
2998        Relh=Relh+Urelh
3000        CALL Speed_of_light
3002        CALL Calc_eps_re
3004        Relh=Relh-Urelh
3006        Eps(5)=Eps_re
3008   !
3010        Mbar=Mbar+Umbar
3012        CALL Speed_of_light
3014        CALL Calc_eps_re
3016        Mbar=Mbar-Umbar
3018        Eps(6)=Eps_re
3020   !
3022        CALL Speed_of_light
3024        C=C+Uc
3026        CALL Calc_eps_re
3028        C=C-Uc
3030        Eps(7)=Eps_re
3032   !
3034        Freq(2)=Freq(2)+Ufreq(2)
3036        CALL Calc_eps_re
3038        Eps(8)=Eps_re
3040        Freq(2)=Freq(2)-Ufreq(2)
3042   !
3044        Micr(1)=Micr(1)+Umicr(1)
3046        CALL Calc_eps_re
3048        Eps(9)=Eps_re
3050        Micr(1)=Micr(1)-Umicr(1)
3052   !
3054        Micr(2)=Micr(2)+Umicr(2)
3056        CALL Calc_eps_re
3058        Eps(10)=Eps_re
3060        Micr(2)=Micr(2)-Umicr(2)
3062   !
3064        CALL Calc_eps_re
3066        Eps(11)=Eps_re
3068        Eps(12)=Eps_re
3070   !
3072        Diameter=Diameter+Udiameter
3074        CALL Calc_tand
3076        Diameter=Diameter-Udiameter
3078        Tandel(1)=Tand
3080   !
3082        Length0=Length0+Ulength
3084        CALL Calc_tand
3086        Length0=Length0-Ulength
3088        Tandel(2)=Tand
3090   !
3092        Samplelen=Samplelen+Usamplelen
3094        CALL Calc_tand
3096        Samplelen=Samplelen-Usamplelen
3098        Tandel(3)=Tand
3100   !
3102        Degc=Degc+Udegc
3104        CALL Speed_of_light
```

```
3106        CALL Calc_tand
3108        Degc=Degc-Udegc
3110        Tandel(4)=Tand
3112   !
3114        Relh=Relh+Urelh
3116        CALL Speed_of_light
3118        CALL Calc_tand
3120        Relh=Relh-Urelh
3122        Tandel(5)=Tand
3124   !
3126        Mbar=Mbar+Umbar
3128        CALL Speed_of_light
3130        CALL Calc_tand
3132        Mbar=Mbar-Umbar
3134        Tandel(6)=Tand
3136   !
3138        CALL Speed_of_light
3140        C=C+Uc
3142        CALL Calc_tand
3144        C=C-Uc
3146        Tandel(7)=Tand
3148   !
3150        Freq(2)=Freq(2)+Ufreq(2)
3152        CALL Calc_tand
3154        Freq(2)=Freq(2)-Ufreq(2)
3156        Tandel(8)=Tand
3158   !
3160        Micr(1)=Micr(1)+Umicr(1)
3162        CALL Calc_tand
3164        Micr(1)=Micr(1)-Umicr(1)
3166        Tandel(9)=Tand
3168   !
3170        Micr(2)=Micr(2)+Umicr(2)
3172        CALL Calc_tand
3174        Micr(2)=Micr(2)-Umicr(2)
3176        Tandel(10)=Tand
3178   !
3180        Bw(1)=Bw(1)+Ubw(1)
3182        CALL Calc_tand
3184        Bw(1)=Bw(1)-Ubw(1)
3186        Tandel(11)=Tand
3188   !
3190        Bw(2)=Bw(2)+Ubw(2)
3192        CALL Calc_tand
3194        Bw(2)=Bw(2)-Ubw(2)
3196        Tandel(12)=Tand
3198   !
3200        CALL Calc_tand
3202   !
3204        MAT Erreps= Eps
3206        MAT Errtand= Tandel
3208        FOR I=1 TO 12
3210            Erreps(I)=Erreps(I)-Eps_re
3212            Errtand(I)=Errtand(I)-Tand
3214        NEXT I
3216        Ueps_re=SQRT(Erreps(1)^2+Erreps(2)^2+Erreps(3)^2+Erreps(7)^2+Erreps(8
  => )^2+Erreps(9)^2+Erreps(10)^2)
3218        Utand=SQRT(Errtand(1)^2+Errtand(2)^2+Errtand(3)^2+Errtand(7)^2+Errtan
  => d(8)^2+Errtand(9)^2+Errtand(10)^2+Errtand(11)^2+Errtand(12)^2)
3220        Ueps_im=Utand*Eps_re
3222 !
3224        IF NPAR<1 THEN !Don't Print
3226 Calc_errs_menu: !
3228            CLEAR SCREEN
3230            PRINT "                UNCERTAINTIES                          DELTA EPS
  =>          DELTA TAND"
```

```
3232          PRINT
3234          PRINT "   CAVITY:                                       "
3236          PRINT USING Image17;"     DIAMETER (mm)               : ",Udiameter
 => *1.E+3,Erreps(1),Errtand(1)
3238          PRINT USING Image17;"     LENGTH (mm)                 : ",Ulength*1
 => .E+3,Erreps(2),Errtand(2)
3240          PRINT "   SAMPLE:                                       "
3242          PRINT USING Image17;"     THICKNESS (mm)              : ",Usamplele
 => n*1.E+3,Erreps(3),Errtand(3)
3244          PRINT "   ENVIRONMENT:                                  "
3246          PRINT USING Image17;"     TEMPERATURE (C)             : ",Udegc,Err
 => eps(4),Errtand(4)
3248          PRINT USING Image17;"     RELATIVE HUMIDITY (%)       : ",Ureln,Err
 => eps(5),Errtand(5)
3250          PRINT USING Image17;"     PRESSURE (mbar)             : ",Umbar,Err
 => eps(6),Errtand(6)
3252          PRINT USING Image17;"     SPEED OF LIGHT (m/s)        : ",Uc,Erreps
 => (7),Errtand(7)
3254          PRINT "   MEASUREMENT:                                  "
3256          PRINT USING Image17;"     LOADED FREQUENCY (GHz)      : ",Ufreq(2)/
 => 1.E+9,Erreps(8),Errtand(8)
3258          PRINT USING Image17;"     EMPTY MICR. READING (mm)    : ",Umicr(1)*
 => 1.E+3,Erreps(9),Errtand(9)
3260          PRINT USING Image17;"     LOADED MICR. READING (mm)   : ",Umicr(2)*
 => 1.E+3,Erreps(10),Errtand(10)
3262          PRINT USING Image17;"     EMPTY Q                     : ",Uq(1),Err
 => eps(11),Errtand(11)
3264          PRINT USING Image17;"     LOADED Q                    : ",Uq(2),Err
 => eps(12),Errtand(12)
3266 !
3268          Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
3270          ON KEY 1 LABEL "Prior  Menu",Prty CALL Prior_menu
3271          ON KEY 2 LABEL "Print  Errors",Prty GOSUB Print_errors
3273          Done=0
3274          Prior_menu=0
3276          LOOP
3278              IF Done THEN GOTO Calc_errs_menu
3280          EXIT IF Prior_menu
3282          END LOOP
3284      END IF
3286      Prior_menu=0
3288      Done=T
3290      OFF KEY
3291      SUBEXIT
3292 Print_errors:!
3293      CLEAR SCREEN
3294      PRINTER IS 1
3295      GOSUB Print_them_errs
3302      PRINTER IS Print_addr
3303      GOSUB Print_them_errs
3304      PRINTER IS T
3305      RETURN
3306 !
3307 ! /////////////////////////////////////////////////////////////
3308 !
3309 Print_them_errs:!
3311      IF Freq(2)<1.E+10 THEN
3312          PRINT USING "K,2D.9D,K,3D.4D,2(K,7D)";"Frequency: ",Freq(2)/1.E+9
 => ,"    Micr:",Micr(2)*1.E+3,"                  Qe:",Q(1),"  Qs:",Q(2)
3313      ELSE
3314          PRINT USING "K,2D.9D,K,3D.4D,2(K,7D)";"Frequency:",Freq(2)/1.E+9,
 => "    Micr:",Micr(2)*1.E+3,"                  Qe:",Q(1),"  Qs:",Q(2)
3315      END IF
3316      PRINT USING Giveus;"UNC:d,l,t,c,f,me,ms,q";Udiameter*1.E+6,Ulength*1.
 => E+6,Usamplelen*1.E+6,Uc,Ufreq(2),Umicr(1)*1.E+6,Umicr(2)*1.E+6,Uq(1),Uq(2)
3317      PRINT USING Imageerreps;Eps_re,Ueps_re,Erreps(1),Erreps(2),Erreps(3),
 => Erreps(7),Erreps(8),Erreps(9),Erreps(10)
3318      ALLOCATE Etand(12)
3319      MAT Etand= Errtand*(1.E+6)
3320      PRINT USING Imageerrtand;Tand*1.E+6,Utand*1.E+6,Etand(1),Etand(2),Eta
 => nd(3),Etand(7),Etand(8),Etand(9),Etand(10),Etand(11),Etand(12)
3321      PRINT
3322      DEALLOCATE Etand(*)
3332      RETURN
5000 Image17:IMAGE K,4D.4D,"      ",D.6D,"           ",D.9D
5001 Imageerreps:IMAGE 3D.4D,"+/-",D.4D,7(X,D.4D)
5002 Imageerrtand:IMAGE 4X,4D,"+/-",4D,2X,7(X,6D),3X,3D,X,3D
5003 Giveus:IMAGE K,3D,4X,3D,4X,D.D,3X,4D,X,6D,4X,D.D,4X,D.D,2X,4D,4D
5005   SUBEND
5006 !
5007 ! *************************************************************
5008 !
5010   SUB T95percent(REAL T,INTEGER V)
5012 !
5014 T95percent:!
5016      SELECT V !find t-distribution value for 95% confidence
5018      CASE 1   !degrees of freedom
5020          T=12.706
5022      CASE 2
5024          T=4.303
5026      CASE 3
5028          T=3.182
5030      CASE 4
5032          T=2.776
5034      CASE 5
5036          T=2.571
5038      CASE 6
5040          T=2.447
5042      CASE 7
5044          T=2.365
5046      CASE 8
5048          T=2.306
5050      CASE 9
5052          T=2.262
5054      CASE 10
5056          T=2.228
5058      CASE 11
5060          T=2.201
5062      CASE 12
5064          T=2.179
5066      CASE 13
5068          T=2.160
5070      CASE 14
5072          T=2.145
5074      CASE 15
5076          T=2.131
5078      CASE 16
5080          T=2.120
5082      CASE 17
5084          T=2.110
5086      CASE 18
5088          T=2.101
5090      CASE 19
5092          T=2.093
5094      CASE 20
5096          T=2.086
5098      CASE 21
5100          T=2.080
5102      CASE 22
5104          T=2.074
5106      CASE 23
5108          T=2.069
```

```
5110       CASE 24
5112           T=2.064
5114       CASE 25
5116           T=2.060
5118       CASE 26
5120           T=2.056
5122       CASE 27
5124           T=2.052
5126       CASE 28
5128           T=2.048
5130       CASE 29
5132           T=2.045
5134       CASE 30
5136           T=2.042
5138       CASE >30,<=40
5140           T=2.021
5142       CASE >40
5144           T=2.
5146       CASE ELSE
5148           BEEP
5150           DISP "CASE SELECT FOR T-DISTRIBUTION DOESNT MATCH"
5152       END SELECT
5154   SUBEND
5156 !
5158 ! ************************************************************
5160 !
5162   SUB Cavity_dimens
5164       OPTION BASE 1
5166       COM /Environment/ Degc,Mbar,Relh,C
5168       COM /Environment/ Udegc,Umbar,Urelh,Uc
5170       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
5172       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
5174       COM /Sample/ Erreps(12),Errtand(12)
5176       COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
5178       COM /Cavity_dims/ Udiameter,Ulength
5180       COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
5182       COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
5184       COM /Constants/ Bessel_root
5186       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
5188 Cavity_dimens:OFF KEY
5190       Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
5192       GOSUB Menu_cav_dimen
5194       DISP "Do you want to CALCULATE or ENTER from keyboard the cavity dime
  => nsions?"
5196       ON KEY 1 LABEL "Prior  Menu",Prty CALL Prior_menu
5198       ON KEY 2 LABEL "CALC   DIMENS",Prty CALL Calc_dimens
5200       ON KEY 3 LABEL "ENTER  DIMENS",Prty GOSUB Enter_dimens
5202       LOOP
5204       EXIT IF Done OR Prior_menu
5206       END LOOP
5208       Prior_menu=0
5210       Done=1
5212       SUBEXIT
5214 !
5216 ! ///////////////////////////////////////////////////////////
5218 !
5220 Enter_dimens:OFF KEY
5222       DISP "Which dimension would you like to change?"
5224       GOSUB Menu_cav_dimen
5226       Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
5228       ON KEY 1 LABEL "Prior  Menu",Prty CALL Prior_menu
5230       ON KEY 2 LABEL "Change Diamter",Prty GOSUB Change_diameter
5232       ON KEY 3 LABEL "Change Length",Prty GOSUB Change_length
5234       Done=0
5236       Prior_menu=0
```

```
5238       LOOP
5240       EXIT IF Prior_menu
5242           IF Done THEN GOTO Enter_dimens
5244       END LOOP
5246       Prior_menu=0
5248       Done=1
5250       RETURN
5252 !
5254 ! ///////////////////////////////////////////////////////////.
5256 !
5258 Menu_cav_dimen:!
5260       CLEAR SCREEN
5262       PRINT USING Image1;Degc,Udegc,Relh,Urelh
5264       PRINT USING Image2;Mbar,Umbar
5266       PRINT USING Image3;C,Uc
5268       PRINT USING Image4;Diameter*1.E+3,Udiameter*1.E+3
5270       PRINT USING Image5;Length0*1.E-3,Ulength*1.E+3
5272       PRINT USING Image11a;Micr(2)*1.E+3,Umicr(2)*1.E+3
5274       PRINT USING Image7c;Bessel_root
5276       PRINT USING Image8c
5278 Image1:IMAGE "Temperature = ",2D.2D," +/- ",D.2D," deg C   Relative Humidit
  => y = ",2D.D," +/- ",D.D," %"
5280 Image2:IMAGE "Barometric Pressure = ",3D.D," +/- ",D.D,"   mbar"
5282 Image3:IMAGE "Speed of Light      = ",D.7DESZ," +/- ",D.3DESZ," meters/seco
  => nd"
5284 Image4:IMAGE "Resonator Diameter            =  ",2D.5D," +/- ",D.5D,"  mm"
5286 Image5:IMAGE "          Length (micr. zero) = ",3D.5D," +/- ",D.5D,"  mm"
5288 Image7c:IMAGE "Bessel function root: ",2D.7D
5290 Image8c:IMAGE "Axial mode  Frequency (GHz)"
5292 Image9c:IMAGE X,2D,9X,2D.7D
5294 Image11a:IMAGE "Micrometer reading = ",2(S2D.4D,"   +/- ",D.4D,6X),"mm"
5296       RETURN
5298 !
5300 ! ///////////////////////////////////////////////////////////
5302 !
5304 Change_diameter:OFF KEY
5306       CALL Edit_data("the cavity diameter (mm)",Diameter,1.E+3,Udiameter)
5308       Done=1
5310       RETURN
5312 !
5314 ! ///////////////////////////////////////////////////////////
5316 !
5318 Change_length:OFF KEY
5320       CALL Edit_data("the cavity length at micrometer=zero (mm)",Length0,1.
  => E+3,Ulength)
5322       Done=1
5324       RETURN
5326   SUBEND
5328 !
5330 ! ************************************************************
5332 !
5334   SUB Find_dims(Spectrum(*),Length0,Diameter,Ulength,Udiameter,Micr,C,Besse
  => l_root,Num_modes)
5336       OFF KEY
5338       OPTION BASE 1
5340       INTEGER N
5342       N=Num_modes
5344       ALLOCATE X(N,2),Y(N,1),B(2,1),Bt(1,2),Xtx(2,2),A(2),Cvm(2,2)
5346       ALLOCATE Xt(2,N),Yt(1,N),Xtxi(2,2),S2(1),Yty(1),Btxty(1),Xty(2,1)
5348 Find_dims: !This routine finds the dimensions of a cylindrical cavity
5350 !using a TEmn frequency spectrum.  The normal application is for the TE01
5352 !spectrum, but by changing the Bessel_root, one may use any other TE mode.
5354 !A matrix approach is used to perform a least squares regression.  The
5356 !uncertainties returned are covariant, (i.e. the confidence interval is
5358 !an ellipse and not a rectangle) so one must be certain to consider the
5360 !length and diameter uncertainties individually and not simultaneously.
```

```
5362 !References: J.D. Jackson "Classical Electrodynamics" 1962 page 255.
5364 !             N. Draper & H. Smith "Applied Regression Analysis, Second
5366 !             Edition" 1981 Chapter 2 and page 532.
5368       FOR I=1 TO Num_modes
5370           X(I,1)=1
5372           X(I,2)=Spectrum(I,1)^2
5374           Y(I,1)=Spectrum(I,2)^2
5376       NEXT I
5378       MAT Xt= TRN(X)
5380       MAT Xtx= Xt*X
5382       MAT Xtxi= INV(Xtx)
5384       MAT Yt= TRN(Y)
5386       MAT Yty= Yt*Y
5388       MAT Xty= Xt*Y
5390       MAT B= Xtxi*Xty
5392       MAT Bt= TRN(B)
5394       MAT Btxty= Bt*Xty
5396       MAT S2= Yty-Btxty
5398       S2(1)=S2(1)/(N-2)
5400       MAT Cvm= Xtxi*(S2(1))
5402       CALL T95percent(T,Num_modes-2)
5404       Length0=C/(2*SQRT(B(2,1)))-Micr
5406       Diameter=Bessel_root*C/(SQRT(B(1,1))*PI)
5408       Ulength=C*(T*SQRT(Cvm(2,2)))/(2*B(2,1)^(1.5))
5410       Udiameter=C*Bessel_root*(T*SQRT(Cvm(1,1)))/(2*PI*B(1,1)^(1.5))
5412       DEALLOCATE X(*),Y(*),B(*),Bt(*),Xtx(*),A(*),Cvm(*)
5414       DEALLOCATE Xt(*),Yt(*),Xtxi(*),S2(*),Yty(*),Btxty(*),Xty(*)
5416   SUBEND
5418   !
5420   ! ***************************************************************
5422   !
5424   SUB Calc_err_anal
5426 Calc_err_anal:OFF KEY
5428   !
5430       OPTION BASE 1
5432       COM /Environment/ Degc,Mbar,Relh,C
5434       COM /Environment/ Udegc,Umbar,Urelh,Uc
5436       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
5438       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
5440       COM /Sample/ Erreps(12),Errtand(12)
5442       COM /Cavity_dims/ Diameter,Length0,Spectrum(40,5),INTEGER Num_modes
5444       COM /Cavity_dims/ Udiameter,Ulength
5446       COM /Cavity_vals/ Freq(2),Micr(2),Q(2),Bw(2)
5448       COM /Cavity_vals/ Ufreq(2),Umicr(2),Uq(2),Ubw(2)
5450       COM /Constants/ Bessel_root
5452       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
5454 Equations: !
5456       F0=Freq(2)
5458       Fe=Freq(2)
5460       L0=Micr(1)
5462       Le=Micr(2)
5464       S=Samplelen
5466       D=Diameter
5468       U=Bessel_root
5470       Eps=Eps_re
5472       Y=Le-S
5474       Fc=(U*C)/(PI*D)
5476       X=Fc/F0
5478       B0=(2*PI*F0*SQRT(1-(X^2)))/C
5480       Be=(2*PI*Fe*SQRT(Eps-(X^2)))/C
5482 Partials:  !
5484       Par_eps_b=1-(X^2)
5486       Par_eps_q=(-2*S*X)+(2*X)
5488       Par_b_be=2*((TAN(Be*S)*(1/(COS(Be*S)^2))*B)/(TAN(B0*Y)^2))
5490       Par_b_b0=2*((-(TAN(Be*S)^2)*(1/(SIN(B0*Y)^2))*Y)/(TAN(B0*Y)))
```

```
5492       Par_b_b=2*((-TAN(Be*S)*(1/(COS(Be*S)^2))*Be)/(TAN(B0*Y)^2))
5494       Par_b_y=2*((-(TAN(Be*S)^2)*(1/(SIN(B0*Y)^2))*B0)/(TAN(B0*Y)))
5496       Par_be_fe=(2*PI*SQRT(Eps_re-(X^2)))/C
5498       Par_be_c=(-2*PI*Fe*SQRT(Eps_re-(X^2)))/(C^2)
5500       Par_be_eps=(PI*Fe)/(C*SQRT(Eps-(X^2)))
5502       Par_be_q=(-2*PI*Fe*X)/(C*SQRT(Eps-(X^2)))
5504       Par_b0_f0=(2*PI*SQRT(1-(X^2)))/C
5506       Par_b0_c=(-2*PI*F0*SQRT(1-X^2))/(C^2)
5508       Par_b0_q=(-2*PI*F0*X)/(C*SQRT(1-(X^2)))
5510       Par_q_fc=1/F0
5512       Par_q_f0=-Fc/(F0^2)
5514       Par_fc_c=U/(PI*D)
5516       Par_fc_d=(-U*C)/(PI*(D^2))
5518       Par_y_le=1
5520       Par_y_l0=-1
5522       Par_y_b=-1
5524 Differentials:!
5526       Del_s=Usamplelen
5528       Del_d=Udiameter
5530       Del_le=Umicr(2)
5532       Del_l0=Umicr(1)
5534       Del_f0=Ufreq(1)
5536       Del_fe=Ufreq(2)
5538       Del_c=Uc
5540       Del_y=(Par_y_le*Del_le)+(Par_y_l0*Del_l0)+(Par_y_b*Del_s)
5542       Del_b0=(Par_b0_f0*Del_f0)+(Par_b0_c*Del_c)+(Par_b0_q*Del_q)
5544       Del_fc=(Par_fc_c*Del_c)+(Par_fc_d*Del_d)
5546       Del_q=(Par_q_fc*Del_fc)+(Par_q_f0*Del_f0)
5548       Del_be=(Par_be_eps*Del_eps)+(Par_be_c*Del_c)+(Par_be_fe*Del_fe)+(Par_
  => be_q*Del_q)
5550       Del_b=(Par_b_be*Del_be)+(Par_b_b0*Del_b0)+(Par_b_b*Del_s)+(Par_b_y*De
  => l_y)
5552       Del_eps=(Par_eps_b*Del_b)+(Par_eps_q*Del_q)
5554   SUBEND
5556 !
5558 ! **************************************************************
5560 !
5562   SUB Sample_info
5564 Sample_info:OFF KEY
5566       OPTION BASE 1
5568       COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
5570       COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
5572       COM /Sample/ Erreps(12),Errtand(12)
5574       COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
5576       DISP CHR$(129);" You may change Sample Information here. ";CHR$(128)
5578       Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
5580       ON KEY 1 LABEL "Prior  Menu",Prty CALL Prior_menu
5582       ON KEY 2 LABEL "Sample Desc",Prty CALL Sample_descr
5584       ON KEY 3 LABEL "Sample Thickns",Prty CALL Sample_thickns
5586       ON KEY 4 LABEL "Sample Init_Eps",Prty CALL Sample_init_eps
5588       Done=0
5590       Prior_menu=0
5592       LOOP
5594           IF Done THEN GOTO Sample_info
5596       EXIT IF Prior_menu
5598       END LOOP
5600       OFF KEY
5602       Prior_menu=0
5604       Done=1
5606       SUBEXIT
5608   SUBEND
5610 !
5612 ! **************************************************************
5614 !
5616   SUB Sample_init_eps
```

```
5618 Sample_init_eps:OFF KEY
5620   !
5622        OPTION BASE 1
5624        COM /Sample/ Sample_id$[55],Samplelen,Eps_re,Eps_im,Tand
5626        COM /Sample/ Usamplelen,Ueps_re,Ueps_im,Utand,Eps_re_guess
5628        COM /Sample/ Erreps(12),Errtand(12)
5630        COM /System_status/ INTEGER Done,Prior_menu,Do_var_freq,Empty_cavity,
  => Prompt$[80]
5632   !
5634        CALL Edit_data("the sample's initial guess for epsilon.",Eps_re_guess
  => ,1)
5636        Done=1
5638   SUBEND
5640 !
5642 ! ****************************************************************
5644 !
5646   SUB Fit_sparms
5648 !      Fit_lorentz(Kx,Ky,Q,F0,Ukx,Uky,Uq,Uf0)           Sss=1
5650 ! or   Fit_lorentz(Q0,Ql,F0,Phase,Uq0,Uql,Uf0,Uphase)  Sss=2 or 3
5652  !Written by Eric J. Vanzura
5654  !Levenberg-Marquart algorithm CSUB taken from:
5656  !Numerical Recipes : the art of Scientific Computing
5658  !W. H. Press, B. P. Flannery, S. A. Teukolsky, W. T. Vetterling
5660  !Cambridge Press 1987
5662 Fit_sparms:OFF KEY
5664        GOSUB Init
5666        GOSUB Init_est
5668        GOSUB Initial_plot            !Plot meas'd S-parms & initial estimate
5670        REPEAT
5672 Do_mrq:!
5674             GOSUB Printlatestvals
5676             Ochisq=Chisq
5678             MAT Old_a= A
5680             DISP "Performing Levenberg-Marquart Least-square-fit"
5682             OFF KEY
5684             CALL Mrqmin(X(*),Y(*),Sig(*),Ndata(*),A(*),Ma(*),Lista(*),Mfit(*)
  => ,Covar(*),Alpha(*),Nca(*),Chisq,Alamda,Sss,Debug)
5686             GOSUB Convergencekeys
5688             MAT Old_est= Sgen
5690             GOSUB Gen_estimate
5692             IF Speed<>1 THEN
5694                 GOSUB Plot_estimate
5696             END IF
5698             GOSUB Ask_converge
5700        UNTIL Itst>=Itst_limit OR Force_converge
5702 Mrq_out:OFF KEY
5704        Alamda=0
5706        CALL Mrqmin(X(*),Y(*),Sig(*),Ndata(*),A(*),Ma(*),Lista(*),Mfit(*),Cov
  => ar(*),Alpha(*),Nca(*),Chisq,Alamda,Sss,Debug)
5708        GOSUB Calc_sig2
5710        GOSUB Calc_uncert
5712        GOSUB Print_results
5714        GOTO Subend
5716 !
5718 ! ////////////////////////////////////////////////////////////////
5720 !
5722 Calc_uncert:!
5724        SELECT Sss
5726        CASE =1
5728            Ua(1)=SQRT(Sig2*Covar(1,1)) !Kx
5730            Ua(2)=SQRT(Sig2*Covar(2,2)) !Ky
5732            Ua(3)=SQRT(Sig2*Covar(3,3)) !Q
5734            Ua(4)=SQRT(Sig2*Covar(4,4)) !F0
5736            Ua(5)=SQRT(Sig2*Covar(5,5)) !
5738        CASE =2,=3
5740            Ua(1)=SQRT(Sig2*Covar(1,1)) !Uq0
5742            Ua(2)=SQRT(Sig2*Covar(2,2)) !Uql
5744            Ua(3)=SQRT(Sig2*Covar(3,3)) !Uf0
5746            Ua(4)=SQRT(Sig2*Covar(4,4)) !Uphase
5748            Ua(4)=Ua(4)*360/2/PI              !in degrees
5750            Ua(5)=SQRT(Sig2*Covar(5,5)) !Umag (beta)
5752        CASE ELSE
5754            BEEP
5756            DISP "Sss=";Sss;" in Calc_uncert. PAUSED"
5758            PAUSE
5760            GOTO Calc_uncert
5762        END SELECT
5764        RETURN
5766 !
5768 ! ////////////////////////////////////////////////////////////////
5770 !
5772 Calc_sig2:!
5774        FOR I=1 TO Fitcount
5776            GOSUB Gen_cnum
5778            N=Start_index+I-1
5780            Sig2=Sig2+(ABS(CMPLX(S(N,Ns,1),S(N,Ns,2))-Cnum))^2
5782        NEXT I
5784        Sig2=Sig2/(Fitcount-Mfit(2))
5786        RETURN
5788 !
5790 ! ////////////////////////////////////////////////////////////////
5792 !
5794 Print_results:!
5796        PRINTER IS 1
5798        PRINT Description$;" fit to ";Sparm$;" values."
5800        PRINT USING "K,3D,2(3X,K,MD.3DE)";"Iteration #",K,"Chi_squared=",Chis
  => q,"Alamda=",Alamda
5802        PRINT USING "3D,K,3D,K,3D";Dcount;" data points in file. Start_index=
  => ",Start_index,"  Stop_index=",Stop_index
5804        GOSUB Print_a
5806        GOSUB Print_ua
5808        PRINT USING "K,D.40E";"Chi-squared=",Chisq
5810        PRINTER IS Print_addr
5812        GOSUB Print_a
5814        GOSUB Print_ua
5816        PRINT USING "K,D.40E";"Chi-squared=",Chisq
5818        PRINTER IS 1
5820        Reset_keys=1
5822        RETURN
5824 !
5826 ! ////////////////////////////////////////////////////////////////
5828 !
5830 Initial_plot:!
5832        ALLOCATE File(Fitcount*3,2),Old_est(Fitcount,2),Orig_sparm(Fitcount,2
  => )
5834        ALLOCATE Sgen(Fitcount,2)
5836        MAT Orig_sparm(*,1:2)= S(Start_index:Stop_index,Ns,1:2)
5838        GOSUB Gen_estimate
5840        MAT Old_est= Sgen
5842        IF NOT Speed THEN GOSUB Plot_estimate
5844        RETURN
5846 !
5848 ! ////////////////////////////////////////////////////////////////
5850 !
5852 Gen_cnum:!
5854        SELECT Sss
5856        CASE 1   !S21
5858            Cnum=-X(I)^2*CMPLX(A(1),-A(2))/CMPLX(X(I)^2-A(4)^2,-X(I)*A(4)/A(3
  => ))
5860        CASE 2  .!S11
5862            Degrees=360*A(4)/2/PI
5864            P=CMPLX(COS(Degrees),SIN(Degrees))
```

```
5866            Ql2f=2*A(2)*(X(I)-A(3))/A(3)
5868            Cnum=A(5)*CMPLX(A(2)/A(1)+Ql2f*Ql2f,Ql2f*(1-A(2)/A(1)))*P/(1+Ql2f
 => *Ql2f)
5870        CASE 3   !S21
5872            Degrees=360*A(4)/2/PI
5874            P=CMPLX(COS(Degrees),SIN(Degrees))
5876            G0gl=1-A(1)/A(2)
5878            Dum=CMPLX(1,2*(X(I)-A(3))/A(2))
5880            Cnum=A(5)*G0gl*P/Dum
5882        CASE ELSE
5884            BEEP
5886            DISP "Sss=";Sss;" is not valid in Gen_cnum."
5888            PAUSE
5890        END SELECT
5892        RETURN
5894 !
5896 ! //////////////////////////////////////////////////////////////
5898 !
5900 Gen_estimate:!
5902        DISP "Generating latest estimate values"
5904        MAT Sgen= (0)
5906        SELECT Sss
5908        CASE =1  !Nussenzveig Lorentzian
5910            FOR I=1 TO Fitcount
5912                GOSUB Gen_cnum
5914                Sgen(I,1)=REAL(Cnum)
5916                Sgen(I,2)=IMAG(Cnum)
5918            NEXT I
5920        CASE =2  !Schulten S11
5922            FOR I=1 TO Fitcount
5924                GOSUB Gen_cnum
5926                Sgen(I,1)=REAL(Cnum)
5928                Sgen(I,2)=IMAG(Cnum)
5930            NEXT I
5932        CASE =3  !Schulten S21
5934            FOR I=1 TO Fitcount
5936                GOSUB Gen_cnum
5938                Sgen(I,1)=REAL(Cnum)
5940                Sgen(I,2)=IMAG(Cnum)
5942            NEXT I
5944        CASE ELSE
5946            DISP "Sss=";Sss;" is not valid in Gen_estimate."
5948            BEEP
5950            PAUSE
5952            GOTO Gen_estimate
5954        END SELECT
5956        RETURN
5958 !
5960 ! //////////////////////////////////////////////////////////////
5962 !
5964 Plot_estimate:!
5966        Pack_data(File(*))
5968        Id$=Sparm$
5970        Pen=2
5972        Pack_data(File(*),Orig_sparm(*),Fitcount,Pen,Id$)
5974 !
5976        Id$="Old estimate"
5978        Pen=4
5980        Pack_data(File(*),Old_est(*),Fitcount,Pen,Id$)
5982 !
5984        Id$="Latest estimate"
5986        Pen=6
5988        Pack_data(File(*),Sgen(*),Fitcount,Pen,Id$)
5990 !
5992        Test$="Real Part"
5994        Prompt$="Imaginary Part"
```

```
5996        Id$="Results from file:"&Rawdatafile$
5998        Init_graphics(Test$,Prompt$,Id$)
6000        Ax$="XY"
6002        Autoscale(File(*),Ax$)
6004        Refresh_labels
6006        DISP "S-parameters are: Red=measured data, Green=Old estimate, Blue=L
 => atest Estimate"
6008        Plot_all(File(*))
6010        RETURN
6012 !
6014 ! ////////////////////////////////////////////////////////////
6016 !
6018 Init:!
6020        DEG
6022        OPTION BASE 1
6024        INTEGER I,Prty,Itst,Itst_limit,Ns
6026        INTEGER Pen,Done,Fitcount
6028        COMPLEX Cnum,P,Dum
6030        REAL Debug
6032        DIM Test$[160],Prompt$[40],Id$[40],Rawdatafile$[10],Sparm$[3]
6034        COM /History/ Status$[1],Time_orgn$[8],Date_orgn$[11]
6036        COM /History/ Time_chng$[8],Date_chng$[11],Description$[160]
6038        COM /Labels/ Labels$(30)[60],INTEGER Lbl_count,REAL Lbl_addr(30,6)
6040        COM /Data_param/ INTEGER Data_count,Filesize,Curvecount,Roster(17,4)
6042        COM /Data_param/ REAL Sym_size,Symbol$(17)[2],Curve_id$(17)[40]
6044        COM /Data_param/ REAL Xmin_data,Xmax_data
6046        COM /Data_param/ REAL Ymin_data,Ymax_data
6048        COM /Graphics/ INTEGER Speed
6050        COM /Background/ Graphtype$[12],Margins$(2)[10],Papersize$[1]
6052        COM /Background/ REAL Pen_speed,INTEGER Backgnd_pen,Auto_time
6054        COM /Background/ INTEGER Auto_file,REAL X_cross_y,Y_cross_x
6056        COM /Background/ Xgrid_tick$[4],INTEGER Xmajor,Xminor
6058        COM /Background/ Ygrid_tick$[4],INTEGER Ymajor,Yminor
6060        COM /Background/ REAL Xmin_graph,Xmax_graph,Ymin_graph,Ymax_graph
6062        COM /Axes_labels/ Print_xlabels$[3],Print_ylabels$[3]
6064        COM /Axes_labels/ Sig_digits$[2],REAL Xlcsize,Ylcsize
6066        COM /Windowspace/ REAL Xmin,Xmid,Xmax,Ymin,Ymid,Ymax! graph edges UDU
6068        COM /Windowspace/ REAL Xleft,Xright,Ybottom,Ytop! paper edges UDUs
6070        COM /Log_scale/ REAL Xcycles,Xbegin,Ycycles,Ybegin
6072        COM /Plot_device/ Plot_lang$[10],INTEGER Plot_addr
6074        COM /Hard_space/ REAL Xview_lft,Xview_rt,Yview_btm,Yview_top
6076        COM /Hard_space/ REAL Viewscale,Aspect_ratio
6078        COM /Frame_onoff/ INTEGER Frame_flag
6080        COM /Clear_space/ REAL Space_lft,Space_rt,Space_btm,Space_top
6082 !      !
6084 Init_com:  !
6086        COM /Fittype/ INTEGER Sselect,Start_index,Stop_index
6088        COM /S_array/ INTEGER Dcount,Svalid,REAL S(801,5,2)
6090        COM /Cdata/ INTEGER Mfit(2),Lista(5,2),Ma(2),REAL A(5),Ua(5),INTEGER
 => Ndata(2)
6092        COM /Cstats/ Alamda,Chisq,INTEGER Nca(2),REAL Alpha(5,5),Covar(5,5)
6094        COM /Output/ INTEGER Print_addr,Plotter,Printer_on,Plotter_on
6096        ASSIGN @Printer TO Print_addr
6098 Init_vals:!
6100        K=0
6102        Itst=0
6104        Itst_limit=3
6106 !      Printer_on=0
6108        Plot_addr=3
6110        Plot_lang$="INTERNAL"
6112        M=5      !Dimension of Covar,Alpha,Lista & A matricies
6114        Debug=0
6116 !
6118        IF NOT BIT(Svalid,Sselect) THEN
6120            BEEP
6122            DISP "The S(*) array for ";Sparm$;" is not valid (doesn't have da
```

```
 => ta?) PAUSED"
6124             PRINT "The S(*) array for ";Sparm$;" is not valid (doesn't have d
 => ata?) PAUSED"
6126             PAUSE
6128         END IF
6130 !Sss=1 S21-Nussenzvieg    Sss=2 S11-Schulten    Sss=3 S21-Schulten
6132         SELECT Sselect
6134         CASE 0       !S11
6136             Sss=2
6138             Sparm$="S11"
6140         CASE 1       !S21
6142             Sss=1
6144 !           Sss=3
6146             Sparm$="S21"
6148         CASE 2       !S12
6150             Sss=1
6152             Sss=3
6154             Sparm$="S12"
6156         CASE 3       !S22
6158             Sss=2
6160             Sparm$="S22"
6162         CASE ELSE
6164             BEEP
6166             DISP "Sselect is not valid in Fit_sparms"
6168             PAUSE
6170         END SELECT
6172         Ns=Sselect+1
6174         RETURN
6176 !
6178 ! ////////////////////////////////////////////////////////////
6180 !
6182 Init_est:  !
6184         Fitcount=Stop_index-Start_index+1
6186         Start_freq=S(Start_index,5,1)
6188         Stop_freq=S(Stop_index,5,1)
6190         SELECT Sss
6192         CASE =1  ! Nussenzveig Lorentzian
6194             Ix=INT(Start_index+Fitcount/2)
6196             Angle=ARG(CMPLX(S(Ix,Ns,1),S(Ix,Ns,2)))
6198             Mag=ABS(CMPLX(S(Ix,Ns,1),S(Ix,Ns,2)))
6200             A(1)=-2*10^(-5)*Mag*SIN(Angle)
6202             A(2)=-2*10^(-5)*Mag*COS(Angle)
6204             A(3)=1.5*S(Ix,5,1)/(Stop_freq-Start_freq)
6206             A(4)=S(Ix,5,1)
6208         CASE =2,=3  ! Schulten S11 or S21
6210             Ix=INT(Start_index+Fitcount/2)
6212             A(1)=S(Ix,5,1)/(Stop_freq-Start_freq)*1.8
6214             A(2)=S(Ix,5,1)/(Stop_freq-Start_freq)*2.0
6216 !           A(1)=S(Ix,5,1)/90000
6218 !           A(2)=S(Ix,5,1)/79000
6220             A(3)=S(Ix,5,1)
6222             A(4)=ARG(CMPLX(S(Ix,Ns,1),S(Ix,Ns,2)))*2*PI/360
6224 !           A(4)=0
6226 !           A(5)=ABS(CMPLX(S(Ix,Ns,1),S(Ix,Ns,2)))
6228             A(5)=.7
6230         CASE ELSE
6232             DISP "Sss=";Sss;" is not valid in Init_est"
6234             BEEP
6236             PAUSE
6238             GOTO Init_est
6240         END SELECT
6242         Ndata(1)=0
6244         Ndata(2)=Fitcount*2
6246         Ma(1)=0             !size of A array (unknowns)
6248         Ma(2)=M
6250         Chisq=1             !measure of fit
6252         Mfit(1)=0           !number of unknowns to be fitted
6254         Mfit(2)=M
6256         IF Sss=1 THEN Mfit(2)=M-1
6258         MAT Lista= (0)
6260         Lista(1,2)=1        !A array index of the unknowns to be fitted
6262         Lista(2,2)=2
6264         Lista(3,2)=3
6266         Lista(4,2)=4
6268         Lista(5,2)=5
6270         IF Sss=1 THEN Lista(5,2)=0
6272         MAT Covar= (0)
6274         MAT Alpha= (0)
6276         Nca(1)=0            !Size of Alpha & Covar arrays >= Ma
6278         Nca(2)=M
6280         Alamda=-2
6282         ALLOCATE Old_a(M)
6284         ALLOCATE REAL X(Fitcount*2),Y(Fitcount*2),Sig(Fitcount*2)
6286         MAT Sig= (1)
6288         MAT X(1:Fitcount)= S(Start_index:Stop_index,5,1)
6290         MAT X(Fitcount+1:Fitcount*2)= S(Start_index:Stop_index,5,1)
6292         MAT Y(1:Fitcount)= S(Start_index:Stop_index,Ns,1)
6294         MAT Y(Fitcount+1:Fitcount*2)= S(Start_index:Stop_index,Ns,2)
6296         RETURN !Init_est
6298 !
6300 ! ////////////////////////////////////////////////////////////
6302 !
6304 Done_return:OFF KEY
6306         Done=1
6308         RETURN
6310 !
6312 ! ////////////////////////////////////////////////////////////
6314 !
6316 Force_converge:PRINT "Convergence will be forced."
6318         Force_converge=1
6320         RETURN
6322 !
6324 ! ////////////////////////////////////////////////////////////
6326 !
6328 Preventconverge:PRINT "Convergence will be prevented."
6330         Itst=0
6332         RETURN
6334 !
6336 ! ////////////////////////////////////////////////////////////
6338 !
6340 Graph_on: !
6342         Speed=1
6344         OFF KEY 4
6346         ON KEY 4 LABEL "GRAPH  OFF",Prty GOSUB Graph_off
6348         RETURN
6350 !
6352 ! ////////////////////////////////////////////////////////////
6354 !
6356 Graph_off: !
6358         Speed=0
6360         OFF KEY 4
6362         ON KEY 4 LABEL "GRAPH  ON",Prty GOSUB Graph_on
6364         RETURN
6366 Debug_on:PRINT "Debug flag is turred on."
6368         Debug=1
6370         OFF KEY 6
6372         ON KEY 6 LABEL "DEBUG  OFF",Prty GOSUB Debug_off
6374         RETURN
6376 !
6378 ! ////////////////////////////////////////////////////////////
6380 !
6382 Debug_off:PRINT "Debug flag is turned off."
```

```
6384       Debug=0
6386       OFF KEY 6
6388       ON KEY 6 LABEL "DEBUG  ON",Prty GOSUB Debug_on
6390       RETURN
6392 !
6394 ! ///////////////////////////////////////////////////////////////
6396 !
6398 Ask_converge:!
6400       SELECT Chisq
6402       CASE >Ochisq
6404           Itst=0
6406           PRINT USING "K,D.DE";"New Chisq is greater than old Chisq by:";Ch
  => isq-Ochisq
6408       CASE =Ochisq
6410           PRINT "Chisq is equal to old Chisq"
6412       CASE <Ochisq
6414           PRINT USING "K,D.DE";"New Chisq is LESS than old Chisq by:";Ochis
  => q-Chisq
6416           IF ABS((Ochisq-Chisq)/Chisq)<.003 THEN Itst=Itst+1
6418           IF ABS((Ochisq-Chisq)/Chisq)<.00003 THEN Itst=Itst+1
6420           PRINT USING "K,D";"Itst=";Itst
6422           PRINT USING "K,D.DE";"ABS((Ochisq-Chisq)/Chisq)=";ABS((Ochisq-Chi
  => sq)/Chisq)
6424       END SELECT
6426       RETURN
6428 !
6430 ! ///////////////////////////////////////////////////////////////
6432 !
6434 Convergencekeys:!
6436       Prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
6438       ON KEY 0 LABEL "FORCE  CONVERG",Prty GOSUB Force_converge
6440       ON KEY 2 LABEL "PREVENTCONVERG",Prty GOSUB Preventconverge
6442       IF Speed THEN
6444           ON KEY 4 LABEL "GRAPH  OFF",Prty GOSUB Graph_on
6446       ELSE
6448           ON KEY 4 LABEL "GRAPH  ON",Prty GOSUB Graph_off
6450       END IF
6452       IF Debug THEN
6454           ON KEY 6 LABEL "DEBUG  OFF",Prty GOSUB Debug_off
6456       ELSE
6458           ON KEY 6 LABEL "DEBUG  ON",Prty GOSUB Debug_on
6460       END IF
6462       RETURN
6464 !
6466 ! ///////////////////////////////////////////////////////////////
6468 !
6470 Printlatestvals:!
6472       PRINT USING "K,MD.3DE";"Chisq=",Chisq
6474       K=K+1
6476       PRINT USING "K,3D,L,K,MD.3DE";"Iteration #",K,"Alamda=",Alamda
6478       GOSUB Print_a
6480       RETURN
6482 !
6484 ! ///////////////////////////////////////////////////////////////
6486 !
6488 Print_a:!
6490       SELECT Sss
6492       CASE =1
6494           IF A(4)<1.E+10 THEN
6496               PRINT USING "2(K,MD.4DE,3X),K,M6D,3X,K,X,MD.9DE";"Kx =",A(1),
  => "Ky =",A(2),"Q =",A(3),"F0 =",A(4)
6498           ELSE
6500               PRINT USING "2(K,MD.4DE,3X),K,M6D,3X,K,M2D.9DE";"Kx =",A(1),"
  => Ky =",A(2),"Q =",A(3),"F0 =",A(4)
6502           END IF
6504       CASE =2,=3
```

```
6506               PRINT USING "2(K,MD.4DE,2X),K,MD.9DE,2X,K,M3D.DD,2X,K,MD.3DE";"Q0
  => =",A(3)/A(1),"Ql=",A(3)/A(2),"F0=",A(3),"Degrees=",A(4)*360/2/PI,"Mag=",A(5
  => )
6508       CASE ELSE
6510           DISP "Sss=";Sss;" is not valid in print_a. program paused"
6512           BEEP
6514           PAUSE
6516       END SELECT
6518       RETURN
6520 !
6522 ! ///////////////////////////////////////////////////////////////
6524 !
6526 Print_ua:!
6528       SELECT Sss
6530       CASE =1
6532           PRINT USING "2(K,MD.3DE,4X),K,M5D,3X,K,3X,MD.3DE";"Ukx=",Ua(1),"U
  => ky=",Ua(2),"UQ=",Ua(3),"Uf0=",Ua(4)
6534       CASE =2,=3
6536           PRINT USING "2(K,MD.3DE,2X),K,MD.9DE,2X,K,M3D.3D,2X,K,MD.3DE";"UQ
  => 0=",Ua(1),"UQl=",Ua(2),"UF0=",Ua(3),"Udegrees=",Ua(4),"Umag=",Ua(5)
6538       CASE ELSE
6540           DISP "Sss=";Sss;" is not valid in Print_ua. program paused"
6542           BEEP
6544           PAUSE
6546       END SELECT
6548       RETURN
6550 !
6552 ! ///////////////////////////////////////////////////////////////
6554 !
6556 Subend:!
6558   SUBEND     !Fit_sparms
6560 !
6562 ! ***************************************************************
6564 !
6566   CSUB Mrqmin(X(*),Y(*),Sig(*),INTEGER Ndata(*),REAL A(*),INTEGER Ma(*),Lis
  => ta(*),Mfit(*),REAL Covar(*),Alpha(*),INTEGER Nca(*),REAL Chisq,Alamda,Sss,D
  => ebug)
6568   SUB Enterfilename(Ac$,OPTIONAL Prompt$)
6570       COM /Files/ Diskdrive$[20],Filename$[10]
6572       INTEGER I,Ascii_num
6574       DIM Test$[160]
6576 Enterfilename:!
6578       IF Filename$<>"" THEN
6580           OUTPUT 2 USING "K,#";Filename$
6582       END IF
6584       SELECT NPAR
6586       CASE 1
6588           DISP " ENTER the FILE NAME ";
6590       CASE 2
6592           DISP " ENTER the FILE NAME for ";Prompt$;
6594       END SELECT
6596       SELECT Ac$
6598       CASE "CAT"
6600           DISP " ... (ENTER alone to CAT) ";
6602       CASE "ABORT"
6604           DISP " ... (ENTER alone to ABORT) ";
6606       CASE "VALID"
6608       END SELECT
6610       LINPUT Test$
6612       Test$=TRIM$(Test$)
6614       IF LEN(Test$)=0 THEN
6616           SELECT Ac$
6618           CASE "VALID"
6620               DISP "You MUST enter the FILE NAME now."
6622               BEEP
6624               WAIT 1.8
```

```
6626              GOTO Enterfilename
6628          CASE "ABORT","CAT"
6630              GOTO Abortline
6632          CASE ELSE
6634              DISP "Ac$=";Ac$;" in SUB Enterfilename"
6636              BEEP
6638              PAUSE
6640          END SELECT
6642      END IF
6644      IF LEN(Test$)>10 THEN
6646          BEEP
6648          DISP "ERROR in NAME ENTRY--up to 10 chars, you have "
6650          DISP LEN(Test$);" "
6652          WAIT 1.8
6654          OUTPUT 2 USING "#,K,K";" #";Test$
6656          GOTO Enterfilename
6658      END IF
6660      Filename$=Test$
6662      FOR I=1 TO LEN(Filename$)
6664          Ascii_num=NUM(Filename$[I])
6666          SELECT Ascii_num
6668          CASE 65 TO 90,95,97 TO 122,48 TO 57
6670              !Allowed characters
6672          CASE ELSE
6674              BEEP
6676              DISP "ERROR in NAME ENTRY--ILLEGAL CHARACTERS, TRY AGAIN."
6678              WAIT 1.8
6680              OUTPUT 2 USING "#,K,K";" #";Filename$
6682              GOTO Enterfilename
6684          END SELECT
6686      NEXT I
6688      SUBEXIT
6690 Abortline:Filename$=""
6692      SUBEXIT
6694   SUBEND
6696   !
6698   ! ****************************************************************
6700   !
6702   SUB Select_disk(OPTIONAL Prompt$)
6704 Select_disk: ! Original: 13 Nov 1984
6706              ! Revision: 02 Dec 1987
6708      COM /Files/ Diskdrive$[20],Filename$[10]
6710      INTEGER Local_prty,Dd,Pt,Choose(1)
6712      DIM Disc$(30)[60],Title$[40],Displ$[80]
6714      Local_prty=VAL(SYSTEM$("SYSTEM PRIORITY"))+1
6716      Sys_id$=SYSTEM$("SYSTEM ID")
6718      OFF KEY
6720      !
6722      ! Define the disk drives available for this system, reserve the
6724      ! first characters for the drive address and the characters after
6726      ! the - for a description of the drive.
6728      !
6730      ! Example:
6732      ! Disc$(1)=":,700,0,0      HP 9133H HARD disk, volume 0."
6734      !
6736      !
6738      IF NPAR>0 THEN
6740          Displ$=" SELECT DISK DRIVE for "&Prompt$&" ... Abort will cancel"
6742          Displ$=Displ$[1,80]
6744      ELSE
6746          Displ$=" SELECT DISK DRIVE  ... Abort will cancel. "
6748      END IF
6750      Title$=" Available disk drives for this system. "
6752      Pt=1   ! allow only one select
6754      !
6756      IF Diskdrive$[1,1]<>":" THEN Diskdrive$=""
6758 !     IF Msi_id$[1,1]<>":" THEN Msi_id$=SYSTEM$("MSI")
6760      Msi_id$=SYSTEM$("MSI")
6762      Msi_id$=Msi_id$[POS(Msi_id$,":"),LEN(Msi_id$)]
6764      Diskdrive$=TRIM$(Diskdrive$)
6766      Msi_id$=TRIM$(Msi_id$)
6768      IF LEN(Diskdrive$)>0 AND LEN(Msi_id$)>0 THEN
6770          Disc$(1)=Diskdrive$&RPT$(" ",17-LEN(Diskdrive$))
6772          Disc$(1)=Disc$(1)&"- Last selected disk drive."
6774          Dd=1
6776          IF Diskdrive$<>Msi_id$ THEN
6778              Disc$(2)=Msi_id$&RPT$(" ",17-LEN(Msi_id$))
6780              Disc$(2)=Disc$(2)&"- Start-up mass storage unit specifier."
6782              Dd=Dd+1
6784          ELSE
6786              Disc$(1)=Disc$(1)&"  Start-up MSUS."
6788          END IF
6790      ELSE
6792          IF LEN(Msi_id$)>0 THEN
6794              Disc$(1)=Msi_id$&RPT$(" ",17-LEN(Msi_id$))
6796              Disc$(1)=Disc$(1)&"- Start-up mass storage unit specifier."
6798              Dd=1
6800          ELSE
6802              Dd=0
6804          END IF
6806      END IF
6808 Disk: !
6810      ! ................ customize system drives here ..................
6812      ! Follow format with - after unit specifier, description is
6814      ! optional but recommended.
6816      ! ...............................................................
6818      !
6820      SELECT TRIM$(Sys_id$)
6822      CASE "S300:20"
6824          Disc$(Dd+1)=":,1404                   - HP 9127 single 5.25 floppy drive"
6826          Disc$(Dd+2)=":,1400,1                 - HP 9153 microfloppy drive"
6828          Disc$(Dd+3)=":,1400,0                 - HP 9153 hard disk"
6830 !    Disc$(Dd+4)=":,1400,0,3              - HP 9153 hard disk PROGS volume"
6832 !    Disc$(Dd+5)=":,700,0,0               - HP 9133H hard disk volume 0"
6834 !    Disc$(Dd+6)=":,700,0,1               - HP 9133H hard disk volume 1"
6836      !
6838          Dd=Dd+3! add the number of drive specifiers above
6840      !
6842      CASE "PC300"
6844          Disc$(Dd+1)=":,1500,0                 - LIF/PC a: drive"
6846          Disc$(Dd+2)=":,1500,1                 - LIF/PC b: drive"
6848          Disc$(Dd+3)=":,1500,2                 - LIF/PC c: drive PROGS volume"
6850          Disc$(Dd+4)=":,1500,3                 - LIF/PC c: drive DATA  volume"
6852          Disc$(Dd+5)=":,1500,4                 - LIF/PC c: drive UTILS volume"
6854          Disc$(Dd+6)=":DOS,A                   - DOS/PC A: drive"
6856          Disc$(Dd+7)=":DOS,B                   - DOS/PC B: drive"
6858          Disc$(Dd+8)=":DOS,C                   - DOS/PC C: drive \BLP subdirectory
  => "
6860          Dd=Dd+8
6862      CASE ELSE
6864          BEEP
6866          DISP "You need to define your diskdrives in Select_disk for:";Sys
  =>  _id$
6868          PAUSE
6870      END SELECT
6872      ! ...............................................................
6874      !
6876      CALL Menu_scroll(Displ$,Title$,Disc$(*),Dd,Pt,Choose(*))
6878      IF Pt=0 THEN
6880          Diskdrive$="NO DISK"
6882      ELSE
6884          Dd=POS(Disc$(Choose(Pt)),"-")-1 ! find -
```

```
6886          IF Dd>5 THEN  !  valid msus
6888              Diskdrive$=TRIM$(Disc$(Choose(Pt))[1,Dd])
6890          ELSE
6892              DISP " ERROR in reading MSUS from string, - chr not found. "
6894              BEEP
6896              CALL Pause_key_on
6898              Diskdrive$="NO DISK"
6900          END IF
6902      END IF
6904 Diskselected:OFF KEY
6906      SUBEXIT
6908  SUBEND
6910  !
6912  ! *****************************************************
6914  !
6916  SUB Enter_file(T_f(*),INTEGER Datacount,Id$)
6918 Enter_file:  !
6920      OPTION BASE 1
6922      COM /Files/ Diskdrive$[20],Filename$[10]
6924      ON ERROR CALL Errortrap
6926      ASSIGN @Datapath TO Filename$&Diskdrive$
6928      ENTER @Datapath;Status$
6930      ENTER @Datapath;Id$
6932      ENTER @Datapath;Datacount
6934      ENTER @Datapath;Datacount
6936      ALLOCATE Temp(Datacount,2)
6938      ENTER @Datapath;Temp(*)
6940      ASSIGN @Datapath TO *
6942      IF SIZE(T_f,1)>=Datacount THEN
6944          FOR I=1 TO Datacount
6946              T_f(I,1)=Temp(I,1)
6948              T_f(I,2)=Temp(I,2)
6950          NEXT I
6952      ELSE
6954          DISP "SIZE OF FILE PASSED TO Enter_file=";SIZE(T_f,1);
6956          DISP " IS TOO SMALL. PROCESS ABORTED."
6958          BEEP
6960          WAIT 1.8
6962          Datacount=0
6964          Id$=""
6966      END IF
6968      DEALLOCATE Temp(*)
6970      OFF ERROR
6972  SUBEND
6974 !
6976 ! ***************************************************************
6978 !
6980  SUB Menu_scroll(D$,T$,Items$(*),INTEGER Item_cnt,To_select,Choose(*))
6982 Menu_scroll:! Original: 22 Jun 1987, Galen Koepke, NBS 723.04
6984              ! Revision: 02 Dec 1987
6986              !
6988              ! A general purpose menu utility for scrolling items and
6990              ! selecting a given number of them.
6992              ! The items are arranged in screens of 15 items each and
6994              ! the user may access screens via softkeys. There may be
6996              ! up to 40 screens or 600 items to choose from.
6998              ! Items$(*) contains the item descriptions
7000              ! Item_cnt is the number of items in Items$(*)
7002              ! Choose(*) is dimensioned to the number of required choices
7004              !           and will be filled with the item numbers chosen.
7006              ! To_select is the number of required choices.
7008              !
7010      OPTION BASE 1
7012      PRINTER IS CRT
7014      DEG
7016      GOSUB Def_variables
```

```
7018      GOSUB Define_screens
7020      GOSUB Make_selections
7022      IF Null_file THEN  ! reset to zero
7024          Item_cnt=0
7026          Items$(1)=""
7028          To_select=0    ! no valid selections
7030      END IF
7032      SUBEXIT
7034      !
7036      ! ///////////////////////////////////////////////////
7038      !
7040 Def_variables:!
7042 !    COM /Interrupts/ INTEGER Intr_prty
7044      COM /Bugs/ INTEGER Bug1,Bug2,Bug3,Printer
7046      COM /Sys/ Sys_id$[10]
7048      !
7050      INTEGER Screen_cnt,Items_per_scn,First_item(40),Last_item(40)
7052      INTEGER I,J,K,First_line,Last_line,Active_screen,Pointer,Last_pt
7054      INTEGER Local_prty,Skips,Knobcount,Pointeractive,K0,Null_file
7056      INTEGER Exit_flag,Temp
7058      DIM Marker$[8],Test$[160]
7060      !
7062      ! initialize parameters
7064      !
7066      Local_prty=Intr_prty
7068      IF Local_prty<1 THEN Local_prty=10
7070      IF LEN(Sys_id$)=0 THEN Sys_id$=SYSTEM$("SYSTEM ID")
7072      IF Item_cnt<1 THEN
7074          Null_file=1
7076          Item_cnt=1
7078          To_select=0
7080          Items$(1)="*** Empty ***"
7082      ELSE
7084          Null_file=0
7086      END IF
7088      IF To_select>Item_cnt THEN To_select=Item_cnt
7090      Skips=0
7092      Knobcount=0
7094      Doneflag=0
7096      Marker$="===>"&RPT$(CHR$(8),4)
7098      RETURN
7100      !
7102      ! ///////////////////////////////////////////////////
7104      !
7106 Define_screens:! Set up screens of 15 items each.
7108               !
7110      Items_per_scn=15    ! Maximum number of displayable items
7112      IF INT(Item_cnt/Items_per_scn)=Item_cnt/Items_per_scn THEN
7114          Screen_cnt=INT(Item_cnt/Items_per_scn)
7116      ELSE
7118          Screen_cnt=INT(Item_cnt/Items_per_scn)+1
7120      END IF
7122      J=1
7124      FOR I=1 TO Screen_cnt    ! set up each screen
7126          First_item(I)=J
7128          IF J+Items_per_scn-1<Item_cnt THEN
7130              Last_item(I)=J+Items_per_scn-1
7132              J=J+Items_per_scn
7134          ELSE
7136              Last_item(I)=Item_cnt
7138          END IF
7140      NEXT I
7142      RETURN
7144      !
7146      ! ///////////////////////////////////////////////////
7148      !
```

```
7150 Make_selections:! MENU setup and use.
7152        Active_screen=1   ! first screen is active
7154        First_line=2      ! first printed line on screen = 2 or greater.
7156        GOSUB Write_screen  ! activate screen at Active_screen
7158                            ! and set First_line and Last_line for Pointer
7160                            ! write Marker$ to first non-selected line.
7162        K0=0                ! Keys start at zero
7164        Exit_flag=0         ! allow ENTER key to exit when selections filled.
7166 Key_loop:  !
7168        ON KBD,Local_prty GOSUB Process_kbd
7170        ON KNOB .01,Local_prty GOSUB Move_pointer
7172        IF Skips<To_select THEN
7174            DISP D$
7176            IF To_select>1 THEN
7178                Test$=" Select  "&VAL$(Skips+1)&" of "&VAL$(To_select)
7180            ELSE
7182                Test$=" Select"
7184            END IF
7186            ON KEY K0 LABEL Test$,Local_prty GOSUB Select_item
7188        ELSE
7190            IF To_select>0 THEN
7192                DISP " Selection process complete ..."
7194            ELSE
7196                DISP " Menu for information only ... "
7198            END IF
7200            ON KEY K0 LABEL "Accept",Local_prty GOTO Exit_line
7202        END IF
7204        IF Active_screen<Screen_cnt THEN
7206            ON KEY K0+1 LABEL " Next   Screen",Local_prty GOSUB Next_screen
7208        ELSE
7210            OFF KEY K0+1
7212        END IF
7214        IF Active_screen>1 THEN
7216            ON KEY K0+2 LABEL " Last   Screen",Local_prty GOSUB Last_screen
7218        ELSE
7220            OFF KEY K0+2
7222        END IF
7224        IF Skips>0 THEN
7226            ON KEY K0+3 LABEL " Reset  Select",Local_prty GOSUB Select_reset
7228        ELSE
7230            OFF KEY K0+3
7232        END IF
7234        IF To_select>0 THEN
7236            ON KEY K0+4 LABEL " Abort ",Local_prty GOTO Escape_line
7238        ELSE
7240            OFF KEY K0+4
7242        END IF
7244        IF Screen_cnt>2 THEN
7246            ON KEY K0+6 LABEL "Jump to Screen",Local_prty GOSUB Jump_to_scn
7248        ELSE
7250            OFF KEY K0+6
7252        END IF
7254        IF Exit_flag THEN Exit_line
7256        GOTO Key_loop
7258 Escape_line:Skips=0
7260        MAT Choose= (0)
7262        To_select=0
7264 Exit_line:OFF KEY
7266        OFF KNOB
7268        OFF KBD
7270        OUTPUT KBD;CHR$(255)&CHR$(75);
7272        PRINT CHR$(128);
7274        ! everything cleared, now go back to work.
7276        RETURN
7278        !
7280        ! //////////////////////////////////////////////////
7282
7284 Next_screen:
7286        OFF KBD
7288        OFF KNOB
7290        OFF KEY
7292        IF Active_screen=Screen_cnt THEN RETURN
7294        Active_screen=Active_screen+1
7296        GOSUB Write_screen
7298        RETURN
7300        !
7302        ! //////////////////////////////////////////////////
7304        !
7306 Last_screen:
7308        OFF KBD
7310        OFF KNOB
7312        OFF KEY
7314        IF Active_screen=1 THEN RETURN
7316        Active_screen=Active_screen-1
7318        GOSUB Write_screen
7320        RETURN
7322        !
7324        ! //////////////////////////////////////////////////
7326        !
7328 Jump_to_errors:DISP " Not a valid screen number ... try again. "
7330        BEEP
7332        WAIT 1.8
7334 Jump_to_scn: !
7336        OFF KBD
7338        OFF KNOB
7340        OFF KEY
7342        DISP " ENTER the screen number desired (1 to ";Screen_cnt;").";
7344        LINPUT Test$
7346        Test$=TRIM$(Test$)
7348        IF LEN(Test$)=0 THEN Jump_to_return
7350        ON ERROR GOTO Jump_to_errors
7352        Temp=INT(VAL(Test$))
7354        OFF ERROR
7356        IF Temp<1 OR Temp>Screen_cnt THEN Jump_to_errors
7358        Active_screen=Temp
7360        GOSUB Write_screen
7362 Jump_to_return: !
7364        DISP CHR$(12)
7366        Test$=""
7368        RETURN
7370        !
7372        ! //////////////////////////////////////////////////
7374        !
7376 Select_item:!
7378        OFF KBD
7380        OFF KNOB
7382        OFF KEY
7384        IF NOT Pointeractive THEN
7386            DISP "NO additional selections for this screen."
7388            BEEP
7390            WAIT 2
7392            DISP CHR$(12);
7394            RETURN
7396        END IF
7398        IF Skips=To_select THEN
7400            IF To_select=0 THEN
7402                DISP "This menu is for information only,";
7404                DISP " no selection allowed."
7406            ELSE
7408                DISP "All selections have been filled,";
7410                DISP " 'Select Reset' to repeat."
7412            END IF
```

```
7414          BEEP
7416          WAIT 2
7418          DISP CHR$(12);
7420          RETURN
7422      END IF
7424      Skips=Skips+1
7426      Choose(Skips)=First_item(Active_screen)+Pointer-First_line
7428      PRINT CHR$(129);  ! inverse video
7430      PRINT TABXY(10,Pointer);Items$(Choose(Skips))
7432      PRINT CHR$(128);
7434      PRINT TABXY(1,Pointer);
7436      SELECT Pointer
7438      CASE First_line
7440          GOSUB Point_forward
7442      CASE Last_line
7444          GOSUB Point_backward
7446      CASE ELSE
7448          ! move forward unless it requires wrapping to beginning.
7450          IF Skips-1>0 THEN  ! check for selected items.
7452              I=Pointer-First_line
7454              LOOP
7456                  K=0
7458                  FOR J=1 TO Skips
7460                      IF First_item(Active_screen)+I=Choose(J) THEN K=1
7462                  NEXT J
7464              EXIT IF K=0
7466                  I=I+1
7468                  IF I+First_line>Last_line THEN K=-1
7470              EXIT IF K=-1
7472              END LOOP
7474              IF K=0 THEN
7476                  GOSUB Point_forward
7478              ELSE
7480                  GOSUB Point_backward
7482              END IF
7484          ELSE
7486              GOSUB Point_forward
7488          END IF
7490      END SELECT
7492      RETURN
7494      !
7496      ! /////////////////////////////////////////////////
7498      !
7500 Select_reset:      !Clear Choose file
7502      OFF KBD
7504      OFF KNOB
7506      OFF KEY
7508      Skips=0
7510      MAT Choose= (0)
7512      GOSUB Write_screen
7514      RETURN
7516      !
7518      ! ////////////////////////////////////////////////
7520      !
7522 Process_kbd:! Allow use of arrows and enter key in addition to soft.
7524      Test$=KBD$
7526      IF LEN(Test$)=1 AND Test$[1,1]<>CHR$(32) THEN
7528          BEEP 80.,.1
7530          RETURN
7532      END IF
7534      IF Test$[1,1]=CHR$(32) THEN GOSUB Point_forward
7536      IF Test$[1,1]<>CHR$(255) THEN RETURN
7538      SELECT Test$[2,2]
7540      CASE CHR$(255)
7542              ! do nothing
7544      CASE "V","T"
7546          GOSUB Point_forward
7548      CASE "^","W"
7550          GOSUB Point_backward
7552      CASE "E","s","t","&"
7554          IF Skips<To_select THEN
7556              GOSUB Select_item
7558          ELSE
7560              ! exit routine
7562              Exit_flag=1
7564          END IF
7566      CASE ELSE
7568          BEEP 80.,.1
7570      END SELECT
7572      Test$=""
7574      RETURN
7576      !
7578      ! /////////////////////////////////////////////////
7580      !
7582 Point_forward:Knobcount=5
7584      GOSUB Move_pointer
7586      RETURN
7588 Point_backward:Knobcount=-5
7590      GOSUB Move_pointer
7592      RETURN
7594      !
7596      ! /////////////////////////////////////////////////
7598      !
7600 Jog_pointer:! Move the selection pointer on the active screen.
7602             ! without regard to selected values
7604      IF Knobcount>0 THEN  ! Move forward
7606          Pointer=Pointer+1
7608      ELSE                 ! Move backward
7610          Pointer=Pointer-1
7612      END IF
7614      IF Pointer<First_line THEN Pointer=Last_line
7616      IF Pointer>Last_line THEN Pointer=First_line
7618      RETURN
7620      !
7622      ! ////////////////////////////////////////////////////////
7624      !
7626 Move_pointer:! Control pointer to avoid re-selection of items
7628      IF NOT Pointeractive THEN RETURN  ! No selections to be made.
7630      Knobcount=Knobcount+KNOBX-KNOBY
7632      IF ABS(Knobcount)<4 THEN RETURN
7634      Last_pt=Pointer
7636      GOSUB Jog_pointer
7638      IF Skips>0 THEN
7640          LOOP
7642              J=Pointer-First_line
7644              FOR I=1 TO Skips
7646                  IF First_item(Active_screen)+J=Choose(I) THEN J=999
7648              NEXT I
7650              IF J=999 AND Pointer=Last_pt THEN Pointeractive=0
7652          EXIT IF Pointeractive=0
7654              IF J=999 THEN GOSUB Jog_pointer
7656          EXIT IF J<>999
7658          END LOOP
7660      END IF
7662      Knobcount=0
7664      OUTPUT KBD;CHR$(255)&CHR$(84);  ! Bring screen home
7666      IF Last_pt=Last_line THEN PRINT CHR$(132);
7668      PRINT "   ";
7670      IF Pointeractive THEN ! Pointer active
7672          IF Pointer=Last_line THEN
7674              PRINT CHR$(132);
7676          ELSE
```

```
7678                PRINT CHR$(128);
7680            END IF
7682            PRINT TABXY(1,Pointer);Marker$;CHR$(128);
7684        END IF
7686        RETURN
7688        !
7690        ! ///////////////////////////////////////////////////
7692        !
7694 Write_screen:! Write the screen pointed to by Active_screen
7696        ! home and clear screen
7698        OUTPUT KBD;CHR$(255)&CHR$(84)&CHR$(255)&CHR$(75);
7700        Knobcount=KNOBX-KNOBY   ! Clear knob and keyboard
7702        Knobcount=0
7704        Test$=KBD$
7706        Test$=""
7708        !
7710        PRINT TABXY(1,First_line-1);CHR$(132);" Item #| Screen #";
7712        PRINT USING "#,2D,4A,2D,3A";Active_screen," of ";Screen_cnt;" | "
7714        PRINT T$;RPT$(" ",51-LEN(T$));CHR$(128);
7716        J=0
7718        REPEAT
7720            IF J=Last_item(Active_screen)-First_item(Active_screen) THEN
7722                PRINT CHR$(132);
7724                PRINT TABXY(1,First_line+J);RPT$(" ",80)
7726            ELSE
7728                PRINT CHR$(128);
7730            END IF
7732            PRINT TABXY(5,First_line+J);
7734            PRINT USING "3D,A,#";First_item(Active_screen)+J,"|"
7736            IF Skips>0 THEN  ! make this line inverse video
7738                FOR I=1 TO Skips
7740                    IF First_item(Active_screen)+J=Choose(I) THEN
7742                        PRINT CHR$(129);
7744                    END IF
7746                NEXT I
7748            END IF
7750            PRINT TABXY(10,First_line+J);Items$(First_item(Active_screen)+J)
7752            J=J+1
7754        UNTIL J>=(Last_item(Active_screen)-First_item(Active_screen)+1)
7756        Last_line=Last_item(Active_screen)-First_item(Active_screen)
7758        Last_line=Last_line+First_line
7760        !
7762        ! set marker to first non-selected item.
7764        !
7766        Pointeractive=0
7768        IF To_select>0 THEN Pointeractive=1
7770        IF Skips>0 AND Pointeractive=1 THEN ! find first non-selected item
7772            J=0
7774            LOOP
7776                Pointer=First_line+J
7778                FOR I=1 TO Skips
7780                    IF First_item(Active_screen)+J=Choose(I) THEN Pointer=0
7782                NEXT I
7784            EXIT IF Pointer<>0
7786                J=J+1
7788                IF First_line+J>Last_line THEN
7790                    Pointeractive=0
7792                    Pointer=First_line
7794                END IF
7796            EXIT IF Pointer<>0
7798            END LOOP
7800        ELSE
7802            Pointer=First_line
7804        END IF
7806        IF Pointeractive THEN
7808            IF Pointer=Last_line THEN
7810                PRINT CHR$(132);
7812            ELSE
7814                PRINT CHR$(128);
7816            END IF
7818            PRINT TABXY(1,Pointer);Marker$;CHR$(128);
7820        END IF
7822        RETURN
7824    SUBEND
7826    !
7828    ! *************************************************************
7830    !
7832    SUB Errortrap
7834 Errortrap: ! Original: 13 Nov 1984
7836            ! Revision: 02 Dec 1987
7838        ! Trap most errors here
7840        COM /Files/ Diskdrive$[20],Filename$[10]
7842        DIM File$[20],Test$[160],What$[20],Ac$[5]
7844        BEEP 400,.6
7846        SELECT ERRN
7848        CASE 54
7850            DISP "DUPLICATE FILE NAME:  ";Filename$;
7852            DISP "....PURGE old one? (Y/N)";
7854            LINPUT What$
7856            What$=TRIM$(What$)
7858            SELECT What$[1,1]
7860            CASE "Y","y"
7862                PURGE Filename$&Diskdrive$
7864            CASE ELSE
7866                Ac$="VALID"
7868                CALL Enterfilename(Ac$)
7870            END SELECT
7872        CASE 52,53
7874            DISP "Improper FILE NAME --- ENTER NEW FILE NAME";
7876            OUTPUT 2 USING "#,K,K";" #";Filename$
7878            LINPUT Filename$
7880            Filename$=TRIM$(Filename$)
7882        CASE 56
7884            DISP "FILE:  ";Filename$;" is not on this disk, please insert";
7886            DISP " correct disk"
7888            CALL Pause_key_on
7890        CASE 64
7892            DISP "This disk is full, PLEASE insert clean disk"
7894            CALL Pause_key_on
7896        CASE 56
7898            DISP "DATA INPUT disk must be in drive!!  ";
7900            DISP "...CONTINUE when ready."
7902            CALL Pause_key_on
7904        CASE 72,73,76
7906            DISP Diskdrive$;
7908            DISP " is not available, type correct";
7910            DISP " unit specifier (ie. ':,707,0').";
7912            OUTPUT 2 USING "K,#";Diskdrive$
7914            LINPUT Diskdrive$
7916        CASE 80
7918            DISP "CHECK DISK drive door!"
7920            CALL Pause_key_on
7922        CASE ELSE
7924            DISP ERRM$;" 'CONTINUE' when fixed"
7926            CALL Pause_key_on
7928        END SELECT
7930        DISP CHR$(12)
7932        SUBEXIT
7934    SUBEND
7936    !
7938    ! *************************************************************
7940    !
```

```
7942   SUB Pause_key_on
7944 Pause_key_on: ! Make sure that CONTINUE key exists.
7946               ! Original: 02 Dec 1987
7948               ! Revision: 02 Dec 1987
7950      COM /Sys/ Sys_id$[10]
7952      IF Sys_id$[1,4]="S300" THEN ! reset to S300 system keys
7954          CONTROL KBD,15;0
7956          CONTROL CRT,12;2
7958          LOAD KEY
7960      END IF
7962      PAUSE
7964      IF Sys_id$[1,4]="S300" THEN  ! set to S200 compatible keys
7966          OUTPUT KBD USING "K,#";"SCRATCH KEY X"
7968          CONTROL KBD,15;1
7970          CONTROL CRT,12;0
7972      END IF
7974      SUBEXIT
7976   SUBEND
7978   !
7980   ! **************************************************************
7982   !
```

*U.S. GOVERNMENT PRINTING OFFICE:1993-774-025/89040

(4) The −3-dB correction is required because this is a direct measure of $\mathscr{L}(f)$ of *two oscillators*, assuming that the oscillators are of a similar type and that the noise contribution is the same for each oscillator. If one oscillator is sufficiently superior to the other, this correction is not required.

Other defined spectral densities can be calculated and plotted as desired. The plotted or stored value of the spectral density of phase fluctuations in decibels relative to one square radian (dBc rad^2/Hz) is calculated as

$$S_{\delta\phi}(f) = \mathscr{L}(f) + 3. \tag{70}$$

The spectral density of phase fluctuations, in radians squared per hertz, is calculated as

$$S_{\delta\phi}(f) = 10 \exp(S_{\delta\phi}(f)/10), \tag{71}$$

The spectral density of frequency fluctuations, in hertz squared per hertz, is

$$S_{\delta\nu}(f) = f^2 S_{\delta\phi}(f). \tag{72}$$

where $S_{\delta\phi}(F)$ is in decibels with respect to 1 radian.

5. *System Noise Floor Verification*

A plot of the system noise floor (sensitivity) is obtained by repeating the automated measurement procedures with the system modified as shown in Fig. 12. Accurate measurements can be obtained using the configuration shown in Fig. 12a. The reference source supplies 10 dBm to one side of the mixer and 0 dBm to the other mixer input through equal path lengths; phase quadrature is maintained with the phase shifter.

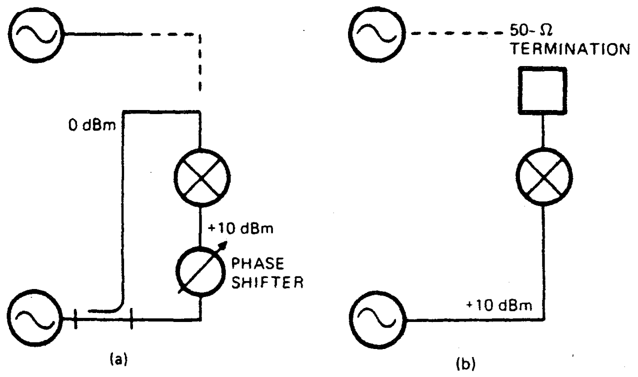


FIG. 12 System configurations for measuring the system noise floor (sensitivity): (a) configuration used for accurate measurements; (b) alternate configuration sometimes used.

The configuration shown in Fig. 12b is sometimes used and does not greatly degrade the noise floor because the reference signal of 10 dBm is larger than the signal frequency. See Sections IV.B and IV.C.4 for additional discussions related to system sensitivity and recommended system evaluation.

Proper selection of drive and output termination of the double-balanced mixer can result in improvement by 15 to 25 dB in the performance of phase-noise measurements, as discussed by Walls *et al.* (1976). The beat frequency between the two oscillators can be a sine wave, as previously mentioned, with proper low drive levels. This requires a proper terminating impedance for the mixer. With high drive levels, the mixer output waveform will be clipped. The slope of the clipped waveform at the zero crossings, illustrated by Walls *et al.* (1976), is twice the slope of the sine wave and therefore improves the noise floor sensitivity by 6 dB, i.e., the output signal, proportional to the phase fluctuations, increases with drive level. This condition of clipping requires characterization over the Fourier frequency range, as previously mentioned for the Hewlett-Packard 3047 phase noise measurement system. An amplifier can be used to increase the mixer drive levels for devices that have insufficient output power to drive the double-balanced mixers.

Lower noise floors can be achieved using high-level mixers when available drive levels are sufficient. A step-up transformer can be used to increase the mixer drive voltage because the signal and noise power increase in the same ratio, and the spectral density of phase of the device under test is unchanged, but the noise floor of the measurement system is reduced.

Walls *et al.* (1976) used a correlation technique that consisted primarily of two phase-noise measurement systems. At TRW the technique is used as shown in Fig. 13. The cross spectrum is obtained with the fast Fourier transform (FFT) analyzer that performs the product of the Fourier transform of one signal and the complex conjugate of the Fourier transform of

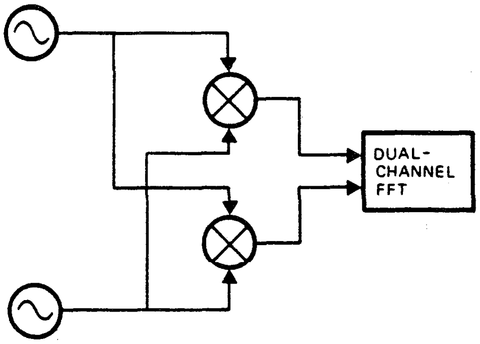


FIG. 13 Cross-spectrum measurement using the two-oscillator technique.

the second signal. This cross spectrum, which is a phase-sensitive characteristic, gives a phase and amplitude sensitivity measure directly. A signal-to-noise enhancement greater than 20 dB can be achieved.

If the double-balanced phase noise measurement system does not provide a noise floor sufficient for measuring a high-quality source, frequency multiplier chains can be used if their inherent noise is 10–20 dB below the measurement system noise. In frequency multiplication the noise increases according to

$$10 \log(\text{final frequency}/\text{original frequency}). \qquad (73) *$$

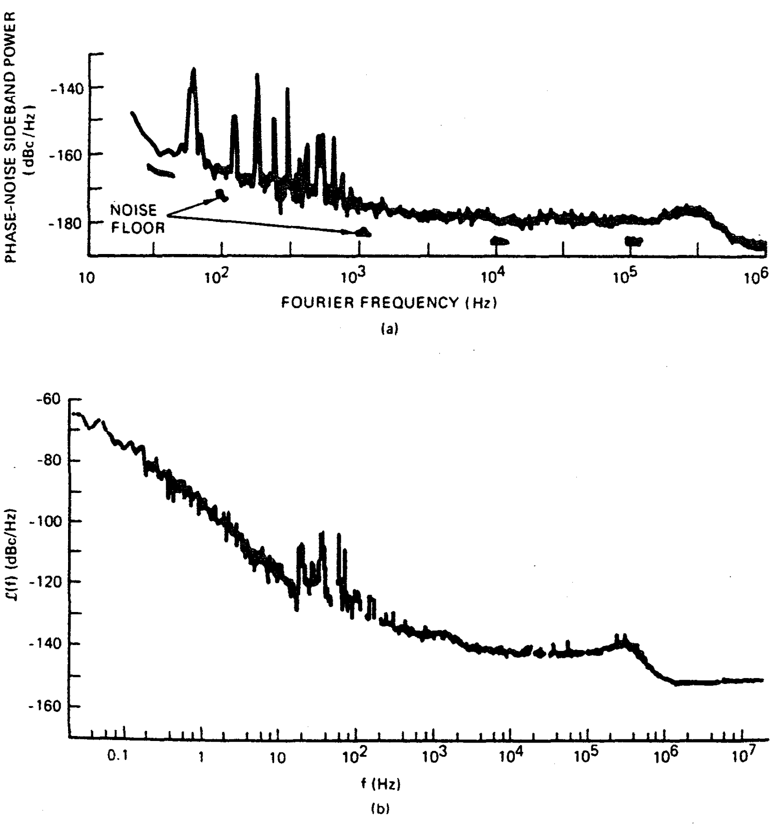


FIG. 14 Data plots of the automated phase-noise measurement system: (a) a high quality 5-MHz quartz oscillator; (b) combined noise of two H.P. 8662A synthesizers (subtract 3 dB for a single unit).

---

* See Appendix Note # 32

The following equation is used to correct for noise-floor contribution P_{nf}, in dBc/Hz, if desired or necessary:

$$\mathscr{L}(f)(\text{corrected}) = -\mathscr{L}(f) + 10\log\left[\frac{P_{\mathscr{L}(f)} - P_{\mathrm{nf}}}{P_{\mathscr{L}(f)}}\right] \tag{74}$$

The correction for noise-floor contribution can also be obtained by using the measurement of $S_{\delta v}(f)$ of Eq. (57). Measurement of $S_{\delta v}(f)$ of the oscillator plus floor is obtained, then $S_{\delta v}(f)$ is obtained for the noise floor only. Then,

$$S_{\delta v}(f)\Big|_{\mathrm{cor}} = S_{\delta v}(f)\Big|_{(\mathrm{osc}+\mathrm{nf})} - S_{\delta v}(f)\Big|_{\mathrm{nf}} \tag{75}$$

Figure 14a shows a phase noise plot of a very high-quality (5-MHz) quartz oscillator, measured by the two-oscillator technique. The sharp peaks below 1000 Hz represent the 60-Hz line frequency of the power supply and its harmonics and are not part of the oscillator phase noise. Figure 14b shows measurements to 0.02 Hz of the carrier at a frequency of 20 MHz.

IV. Single-Oscillator Phase-Noise Measurement Systems and Techniques *

The phase-noise measurements of a single-oscillator are based on the measurement of *frequency fluctuations* using discriminator techniques. The practical discriminator acts as a filter with finite bandwidth that suppresses the carrier and the sidebands on both sides of the carrier. The ideal carrier-suppression filter would provide infinite attentuation of the carrier and zero attenutation of all other frequencies. The effective Q of the practical discriminator determines how much the signals are attenuated.

Frequency discrimination at very high frequencies (VHF) has been obtained using slope detectors and ratio detectors, by use of lumped circuit elements of inductance and capacitance. At ultrahigh frequencies (UHF) between the VHF and microwave regions, measurements can be performed by beating, or heterodyning, the UHF signal with a local oscillator to obtain a VHF signal that is analyzed with a discriminator in the VHF frequency range. Those techniques provide a means for rejecting residual amplitude-modulated (AM) noise on the signal under test. The VHF discriminators usually employ a limiter or ratio detector.

Ashley *et al.* (1968) and Ondria (1968) have discussed the microwave cavity discriminator that rejects AM noise, suppresses the carrier so that the input level can be increased, and provides a high discriminated output to improve the signal-to-noise floor ratio. The delay line used as an FM discriminator has been discussed by Tykulsky (1966), Halford (1975), and

---

* See Appendix Note # 6

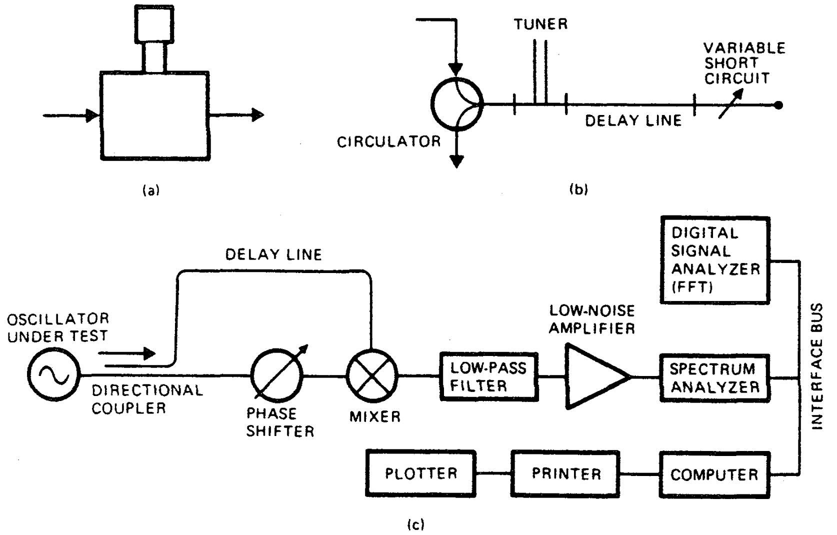


FIG. 15 Single-oscillator phase-noise measurement techniques: (a) cavity discriminator; (b) reflective-type delay-line discriminator; (c) one-way delay line.

Ashley *et al.* (1968). Ashley *et al.* (1968) proposed the reflective-type delay-line discriminator shown in Fig. 15b. The cavity can also be used to replace delay line. The one-way delay line shown in Fig. 15c is implemented in the TRW measurement systems. The theory and applications set forth in this section are based on a system of this particular type.

A. The Delay Line as an FM Discriminator

1. *The Single-Oscillator Measurement System*

The single-oscillator signal is split into two channels in the system shown in Fig. 15. One channel is called the nondelay or reference channel. It is also referred to as the local-oscillator (LO) channel because the signal in this channel drives the mixer at the prescribed impedance level (the usual LO drive). The signal in the second channel arrives at the mixer through a delay line. The two signals are adjusted for phase quadrature with the phase shifter, and the output of the mixer is a fluctuating voltage, analogous to the frequency fluctuations of the source, centered on approximately zero dc volts.

The delay line yields a phase shift by the time the signal arrives at the balanced mixer. The phase shift depends on the instantaneous frequency of the signal. The presence of frequency modulation (FM) on the signal gives rise to differential phase modulation (PM) at the output of the differential delay and its associated (nondelay) reference line. This relationship is linear if the delay τ_d is nondispersive. This is the property that allows the delay line to be used as an FM discriminator. In general, the conversion factors are a function of the delay (τ_d) and the Fourier frequency f but not of the carrier frequency.

The differential phase shift of the nominal frequency ν_0 caused by the delay line is

$$\Delta\phi = 2\pi\nu_0\tau_d, \tag{76}$$

where τ_d is the time delay.

The phase fluctuations at the mixer are related to the frequency fluctuations (at the rate f) by

$$\delta\phi = 2\pi\tau_d\,\delta\nu(f). \tag{77}$$

The spectral density relationships are

$$S_{\delta\phi}(f)\Big|_{\text{mixer}} = (2\pi\tau_d)^2\,S_{\delta\nu}(f)\Big|_{\text{osc}} \tag{78}$$

and

$$S_{\delta\nu}(f) = f^2\,S_{\delta\phi}(f). \tag{79}$$

Then,

$$S_{\delta\phi}(f)\Big|_{\text{dlm}} = (2\pi f\tau_d)^2\,S_{\delta\phi}(f)\Big|_{\text{osc}} \tag{80}$$

where the subscript dlm indicates delay-line method. From Eq. (56), the spectral density of phase for the two-oscillator technique, in radians squared per hertz, is

$$S_{\delta\phi}(f) = 4\,\frac{S_{\delta v}(f)}{(V_{\text{ptp}})^2} = \frac{S_{\delta v}(f)}{2(V_{\text{rms}})^2} = \left[\frac{(\delta v_{\text{rms}})^2}{2(V_{\text{rms}})B}\right] \tag{81}$$

because

$$(V_{\text{ptp}})^2 = 8(v_{\text{rms}})^2 = 4(V_p)^2 = 4[2(v_{\text{rms}})^2]$$

and

$$\mathscr{L}(f) = 2(S_{\delta v}(f)/(V_{\text{ptp}})^2) = (\delta v_{\text{rms}})^2/4(V_{\text{rms}})^2B \tag{82}$$

per hertz.

The sensitivity (noise floor) of the two-oscillator measurement system includes the thermal and shot noise of the mixer and the noise of the base-band preamplifier (referred to its input). This noise floor is measured with the oscillator under test inoperative. The measurement system sensitivity of the two-oscillator system, on a per hertz density basis (dBc/Hz) is

$$\mathscr{L}(f)_{\rm nf} = 10 \log[2(\delta v_n)^2/(V_{\rm ptp})^2], \tag{83}$$

where $\delta v_{\rm n}$ is the rms noise voltage measured in a one-hertz bandwidth.

The two-oscillator system therefore yields the output noise from both oscillators. If the reference oscillator is superior in performance, as assumed in the previous discussions, then one obtains a direct measure of the noise characteristics of the oscillator under test. If the reference and test oscillators are the same type, a useful approximation is to assume that the measured noise power is twice that associated with one noisy oscillator. This approximation is in error by no more than 3 dB for the noisier oscillator. Substituting in Eq. (80) and using the relationships in Eq. (56), we have, per hertz,

$$\mathscr{L}(f)\bigg|_{\rm dlm} = 2[(\delta v_{\rm rms})^2/(V_{\rm ptp})^2](2\pi f \tau_{\rm d})^2 \tag{84}$$

Examination of this equation reveals the following.

(1) The term in the brackets represents the two-oscillator response. *Note that this term represents the noise floor of the two-oscillator method.* Therefore, adoption of the delay-line method results in a higher noise by the factor $(2\pi f \tau_{\rm d})^2$ when compared with the two-oscillator measurement method. The sensitivity (noise floor) for delay lines with different values of time delay are illustrated in Fig. 17.

(2) Equation (84) also indicates that the measured value of $\mathscr{L}(f)$ is periodic in $\omega = 2\pi f$. This is shown in Fig. 21. The first null in the responses is at the Fourier frequency $f = 1/\tau_{\rm d}$. The periodicity indicates that the calibration range of the discriminator is limited and that valid measurements occur only in the indicated range, as verified by the discriminator slope shown in Fig. 16. (See Fig. 23.)

(3) The maximum value of $(2\pi f \tau_{\rm d})^2$ can be greater than unity (it is 4 at $f = 1/2\tau_{\rm d}$). This 6-dB advantage is utilized in the noise-floor measurement. However, it is beyond the valid calibration range of the delay-line system. The 6-dB advantage is offset by the line attenuation at microwave frequencies, as discussed by Halford (1975).

The delay-line discriminator system has been analyzed in terms of a power-limited system (a particular idealized system in which the choice of power oscillator voltage, the attenuator of the delay line, and the conversion loss of the mixer are limited by the capability of the mixer) by Tykulsky (1966), Halford (1975), and Ashley *et al.* (1977). For this particular case, Eq. (83) indicates that an increase in the length of the delay line (to increase τ_d for decorrelation of Fourier frequencies closer to the carrier) results in an increase in attenuation of the line, which causes a corresponding decrease in V_{ptp}. The optimum length occurs where τ_d is such that the decrease in V_{ptp} is approximately compensated by the increase in $(2\pi f\tau_d)$, i.e., where

$$\frac{d}{d\tau_d}\frac{2\pi f\tau_d}{V_{ptp}} = 0. \qquad (85)$$

This condition occurs where the attenuation of the delay line is 1 Np (8.686 dB). However, when the system is not power limited, the attenuation of the delay line is not limited, because the input power to the delay line can be adjusted to maintain V_{ptp} at the desired value. The optimum delay-line length is determined at a particular selectable frequency. However, since the attenuation varies slowly (approximately proportional to the square root of frequency), this characteristic allows near-optimum operation over a considerable frequency range without appreciable degradation in the measurements.

A practical view of the time delay (τ_d) and Fourier-frequency functional relationship can be obtained by reviewing the basic concepts of the dual-channel time-delay measurement system discussed by Lance (1964). If the differential delay between the two channels is zero, there is no phase difference at the detector output when a swept-frequency cw signal is applied to the system. Figure 16 shows the detected output interference display when a swept-frequency cw signal (zero to 4 MHz) is applied to a system that has

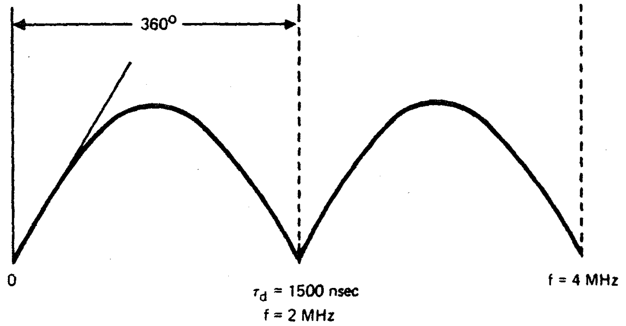


FIG. 16 Swept-frequency interference display at the output of a dual-channel system with a differential delay of 500 nsec.

a differential delay of 500 nsec between the two channels. The signal amplitudes are assumed to be almost equal, thus producing the familiar voltage-standing-wave pattern or interference display. Because this is a two-channel system, there is a null every 360°, as shown.

2. *System Sensitivity (Noise Floor) When Using the Differential Delay-Line Technique*

Halford (1975) has shown that the sensitivity (noise floor) of the single-oscillator differential delay-line technique is reduced relative to the two-oscillator techniques. The sensitivity is modified by the factor

$$S_d = 2(1 - \cos 2\pi f\tau_d). \tag{86}$$

For $\omega\tau_d = 2\pi d\tau_d \ll 1$ a good approximation is

$$S_d^2 = 2(1 - \cos 2\pi f\tau_d) = (\omega\tau_d)^2[1 - \tfrac{1}{12}(\omega\tau_d)^2] = (2\pi f\tau_d)^2 = \theta^2, \tag{87}$$

where θ is the phase delay of the differential delay line evaluated at the frequency f. Figure 17 shows the relative sensitivity (noise floor) of the two-oscillator technique and the single-oscillator technique with different delay-line lengths. The f^{-2} slope is noted at Fourier frequencies beyond

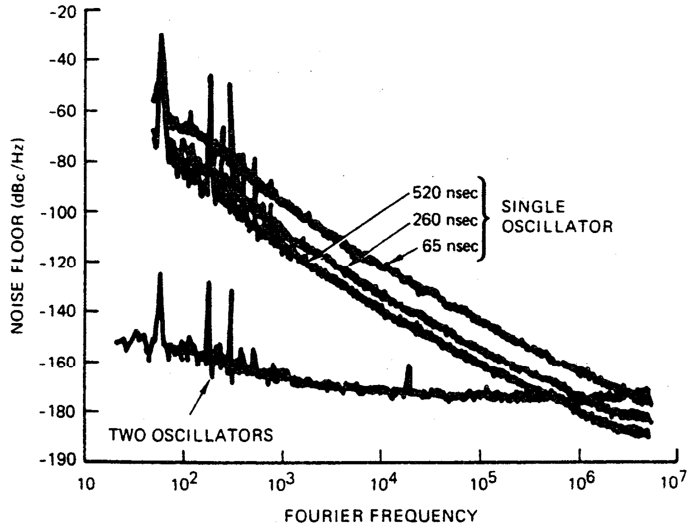
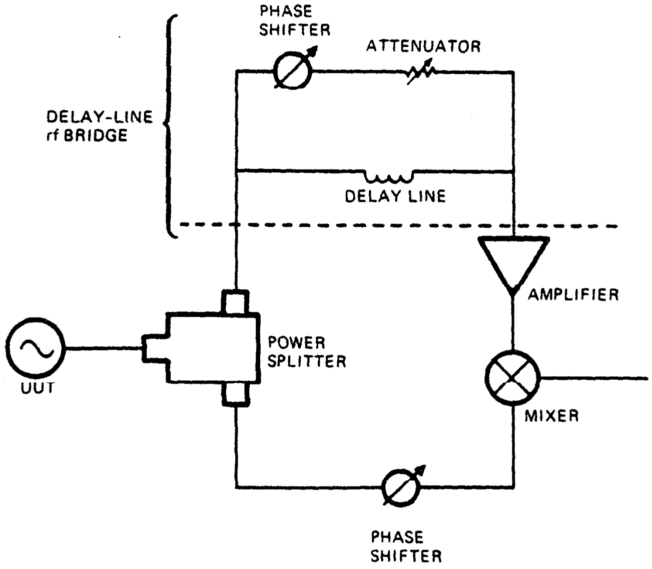


FIG. 17 Relative sensitivity (noise floor) of single-oscillator and two-oscillator phase-noise measurement systems.

about 1 kHz. For Fourier frequencies closer to the carrier, the slope is f^{-3}, i.e., the sum of the f^{-2} slope of Eq. (87) and the f^{-1} flicker noise.

Phase-locked sources have phase-noise characteristics that cannot be measured at close-in Fourier frequencies using this basic system. The relative sensitivity of the system can be improved by using a dual (two-channel) delay-line system and performing cross-spectrum analysis, which will be presented in this chapter.

Labaar (1982) developed the delay-line rf bridge configuration shown in Fig. 18. At microwave frequencies where a high-gain amplifier is available, suppression of the carrier by the rf bridge allows amplification of the noise going into the mixer. A relative sensitivity improvement of 35 dB has been obtained without difficulty. The limitations of the technique depend on the available rf power and the carrier suppression by the bridge. Naturally, if the rf input to the bridge is high one must use the technique with adequate precautions to prevent mixer damage that can occur by an accidental bridge unbalance. Labaar (1982) indicated the added advantage of using the rf bridge carrier-suppression technique when attempting to measure phase noise close to the carrier when AM noise is present. Figure 19 shows the



Fɪɢ. 18 Carrier suppression using an rf bridge to increase relative sensitivity. (Courtesy Instrument Society of America.)

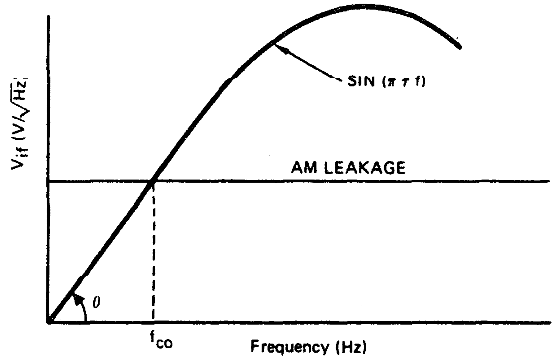


FIG. 19 Phase detector output (AM–PM crossover); τ, delay time.

mixer output for phase (PM) and amplitude (AM) noise in the single-oscillator delay-line FM discriminator system. It is noted that the phase noise and AM noise intersect and that the AM will therefore limit the measurement accuracy near the carrier. Even though AM noise is much lower than phase noise in most sources, and even though the AM is normally suppressed about 20 dB, there is still AM at the mixer output. This output is AM leakage and is caused by the finite isolation between the mixer ports. The two-oscillator technique does not experience this problem to this extent because the phase noise and AM noise maintain their relative relationships at the mixer output independent of the offset frequency from the carrier.

B. CALIBRATION AND MEASUREMENTS USING THE DELAY LINE AS AN FM DISCRIMINATOR

The block diagram of a practical single-oscillator phase noise measurement system is shown in Fig. 20. The signals in the delay-line channel of the system experience the one-way delay of the line. With adequate source power, the system is not limited to the optimum 1 Np (8.686 dB) previously discussed for a power-limited system. Measurements are performed using the following operational procedures.

(1) Measure the tracking spectrum analyzer IF bandwidths as set forth in Section III.C.1.

(2) Establish the system power levels (Section IV.B.1).

(3) Establish the discriminator calibration factor (Section IV.B.2).

(4) Measure and plot the oscillator characteristics in the automatic system used (Section IV.B.3).

(5) Measure the system noise floor (sensitivity) (Section IV.B.4).

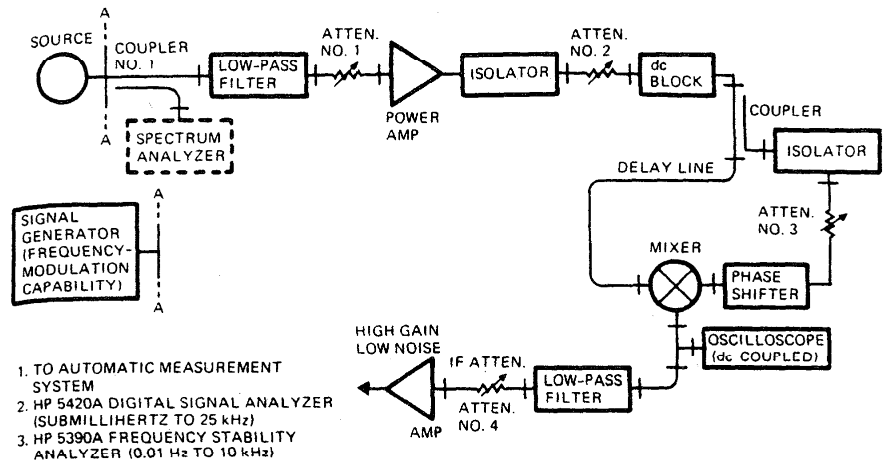


FIG. 20 Single-oscillator phase noise measurement system using the delay line as an FM discriminator. (From Lance *et al.*, 1977a.)

1. *System Power Levels*

The system power levels are set using attenuators, as shown in Fig. 20. Because the characteristic impedance of attenuator No. 4 is 50 Ω, mismatch errors will occur if the mixer output impedance is not 50 Ω. As previously discussed, the mixer drive levels are set so that the mixer output signal, as observed during calibration, is sinusoidal. This has been accomplished in TRW systems with a reference (LO) signal level of 10 dBm and a mixer input level of about 0 dBm from the delay line.

A power amplifier can be used to increase the source signal to the measurement system. This amplifier must not contribute appreciable additional noise to the signal.

2. *Discriminator Calibration*

The discriminator characteristics are measured as a function of frequency and voltage. The hertz-per-volt sensitivity of the discriminator is defined as the *calibration factor* (CF). The calibration process involves measuring the effects of intentional modulation of the source (carrier) frequency. A known modulation index must be obtained to calculate the calibration factors of the discriminator. The modulation index is obtained by using amplitude modulation to establish the carrier-to-sideband ratio when there is considerable instability of the source or when the source cannot be frequency modulated.

It is convenient to consider the system equations and calibration techniques in terms of frequency modulation of stable sources. If the source to be measured cannot be frequency modulated, it must be replaced, during the calibration process, with a modulatable source. The calibration process will be described using a modulatable source and a 20-kHz modulation frequency. However, other modulation frequencies can be used. The calibration factor of this type discriminator has been found to be constant over the usable Fourier frequency range, within the resolution of the measuring technique. The calibration factor of the discriminator is established after the system power levels have been set with the unit under test as the source.

The discriminator calibration procedures are as follows.

(1) Set attenuator No. 4 (Fig. 20) to 50 dB.

(2) Replace the oscillator under test with a signal generator or oscillator that can be frequency modulated. *The power output and operating frequency of the generator must be set to the same precise frequency and amplitude values that the oscillator under test will present to the system during the measurement process.*

(3) Select a modulation frequency of 20 kHz and increase the modulation until the carrier is reduced to the first Bessel null, as indicated on the spectrum analyzer connected to coupler No. 1. This establishes a modulation index ($m = 2.405$).

(4) Adjust the phase shifter for zero volts dc at the output of the mixer, as indicated on the oscilloscope connected as shown in Fig. 20. *This establishes the quadrature condition for the two inputs to the mixer.* This quadrature condition is continuously monitored and is adjusted if necessary.

(5) Tune the tracking spectrum analyzer to the modulation frequency of 20 kHz. The power reading at this frequency is recorded in the program and is corrected for the 50-dB setting of attenuator No. 4, which will be set to zero decibel indication during the automated measurements.

$$P(\text{dBm}) = (-\text{dBm power reading}) + 50\ \text{dB} \qquad (88)$$

This power level is converted to the equivalent rms voltage that the spectrum analyzer would have read if the total signal had been applied:

$$V_{\text{rms}} = \sqrt{10^{P/10}/1000 + R}. \qquad (89)$$

(6) The discriminator calibration factor can now be calculated because this power in dBm can be converted to the corresponding rms voltage using the following equation:

$$V_{\text{rms}} = \sqrt{(10^{P/10}/1000) \times R}, \qquad (90)$$

where $R = 50\ \Omega$ in this system.

(7) The discriminator calibration factor is calculated in hertz per volt as

$$CF = mf_m/\sqrt{2}\,V_{\text{rms}} = 2.405 f_m/\sqrt{2}\,V_{\text{rms}}. \tag{91}$$

The modulation index m for the first Bessel null as used in this technique is 2.405. The modulation frequency is f_m.

3. *Measurement and Data Plotting*

After the discriminator is calibrated, the modulated signal source is replaced with the frequency source to be measured. Quadrature of the signals into the mixer is reestablished, attenuator No. 4 (Fig. 20) is set to 0 dB, and the measurement process can begin.

The measurements, calculations, and data plotting are completely automated. The calculator program selects the Fourier frequency, performs autoranging, and sets the bandwidth, and measurements of Fourier frequency power are performed by the tracking spectrum analyzer. Each Fourier frequency noise-power reading P_n (dBm) is converted to the corresponding rms voltage by

$$v_{1\text{rms}} = \sqrt{10^{(P_n+2.5)/10}/1000 \times R}. \tag{92}$$

The rms frequency fluctuations are calculated as

$$\delta v_{\text{rms}} = v_{1\text{rms}} \times \text{CF}. \tag{93}$$

The spectral density of frequency fluctuations in hertz squared per hertz is calculated as

$$S_{\delta v}(f) = (\delta v_{\text{rms}})^2/B, \tag{94}$$

where B is the measured IF noise-power bandwidth of the spectrum analyzer. The spectral density of phase fluctuations in radians squared per hertz is calculated as

$$S_{\delta\phi}(f) = S_{\delta v}(f)/f^2. \tag{95}$$

The NBS-designated spectral density in decibels (carrier) per hertz is calculated as

$$\mathscr{L}(f)_{\text{dB}} = 10 \log \tfrac{1}{2} S_{\delta\phi}(f). \tag{96}$$

Spectral density is plotted in real time in our program. However, the data can be stored and the desired spectral density can be plotted in other forms. Integrated phase noise can be obtained as desired.

4. *Noise Floor Measurements*

The relative sensitivity (noise floor) of the single-oscillator measurement system is measured as shown in Fig. 12a for the two-oscillator technique. The delay line must be removed and equal channel lengths constructed, as in Fig. (12a). The same power levels used in the original calibration and measurements are reestablished, and the noise floor is measured at specific Fourier frequencies, using the same calibration–measurement technique, or by repeating the automated measurement sequence.

A correction for the noise floor requires a measurement of the rms voltage of the oscillator ($v_{1\mathrm{rms}}$) and a measurement of the noise floor rms voltage ($v_{2\mathrm{rms}}$). These voltages are used in the following equation to obtain the corrected value:

$$v_{\mathrm{rms}} = \sqrt{(v_{1\mathrm{rms}})^2 - (v_{2\mathrm{rms}})^2}. \tag{97}$$

The value v_{rms} is then used in the calculation of frequency fluctuations. If adequate memory is available, each value of $v_{1\mathrm{rms}}$ can be stored and used after the other set of measurements are performed at the same Fourier frequencies.

The following technique was developed by Labaar (1982). Carrier suppression is obtained using the rf bridge illustrated in Fig. 18. One can easily improve sensitivity more than 40 dB. At 2.0 and 3.0 GHz 70-dB carrier suppression was realized. In general, the improvement in sensitivity will depend on the availability of an amplifier or adequate input power.

Figure 21 shows the different noise floors in a delay-line bridge discriminator. It is good measurement discipline to always determine these noise floors; also, the measurements, displayed in Fig. 21, give a quick understanding of the physical process involved. The first trace is obtained by terminating the input of the baseband spectrum analyzer. The measured output noise power is then a direct measure of the spectrum analyzer's noise figure (NF). The input noise is thermal noise and is usually indicated by "KTB," which is short for "the thermal noise power at absolute temperature of T degrees K(elvin) per one hertz bandwidth (B). This KTB number is, at 18°C, about -174 dBm/Hz.

Figure 21a shows that trace number 1 for frequencies above about 1 kHz is level with a value of about -150 dBm $= -(174\text{-}24)$ dBm, which means that the spectrum analyzer has an NF of 24 dB. At 20 Hz the NF has gone up to about 48 dB. To improve the NF, a low-noise (NF, 2dB), low-frequency (10 Hz–10 MHz) amplifier is inserted as a preamplifier. Terminating its input now results in trace number 2. At the high frequency end, the measured power goes up by about 12–13 dB, and the amplifiers gain is 34 dB. This means that the NF is improved by 34 − 12–13 = 21–22 dB, which is an

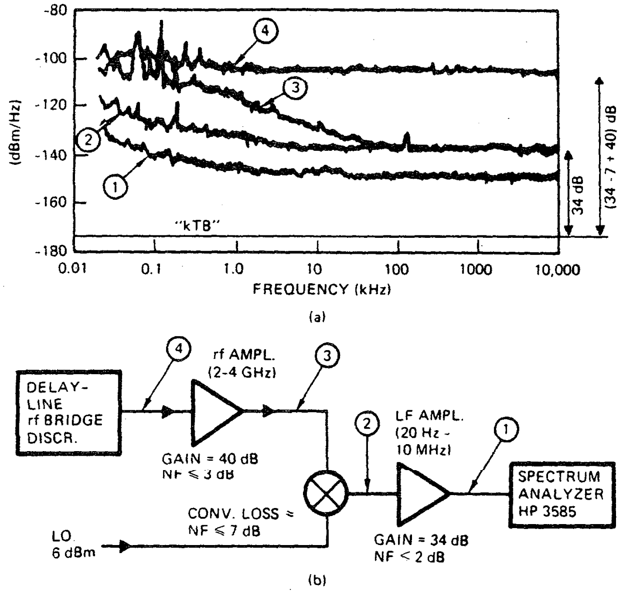


FIG. 21 (a) Noise contribution analysis; (b) phase noise test setup using a delay-line rf bridge discriminator (rf = 2.8 GHz; ○, termination points. NF, noise figure; LF, low frequency. (From Seal and Lance, 1981.)

NF ≃ 2–3 dB as expected, i.e., the first stage noise predominates. The low-frequency end at 20 Hz gives an NF of 26 dB, which overall is quite an improvement.

In trace number 3 the mixer is included with it's rf (signal) port terminated. It is clear from this trace that certainly up to 100 kHz, the noise generated by the mixer diodes being "pumped" by the LO signal dominates. This case represents the "classic" delay-line discriminator. The last trace (number 4) includes the low-noise, high-gain rf amplifier that can be used because the carrier is suppressed in the delay-line rf bridge discriminator, in contrast to the classic delay-line discriminator case. This trace shows that from 1 kHz on up the measured output power is flat, representing a 2–3-dB NF.

At about 20–40 Hz, trace numbers 3 and 4 begin nearing their crossover floor. In this particular case, which is discussed in full by Labaar (1982), the measurement systems noise floor (resolution) has been improved by 40 dB.

Figure 22 shows plots of phase noise as measured at two frequencies using delay lines of different lengths. The delay line used measure at 600 MHz

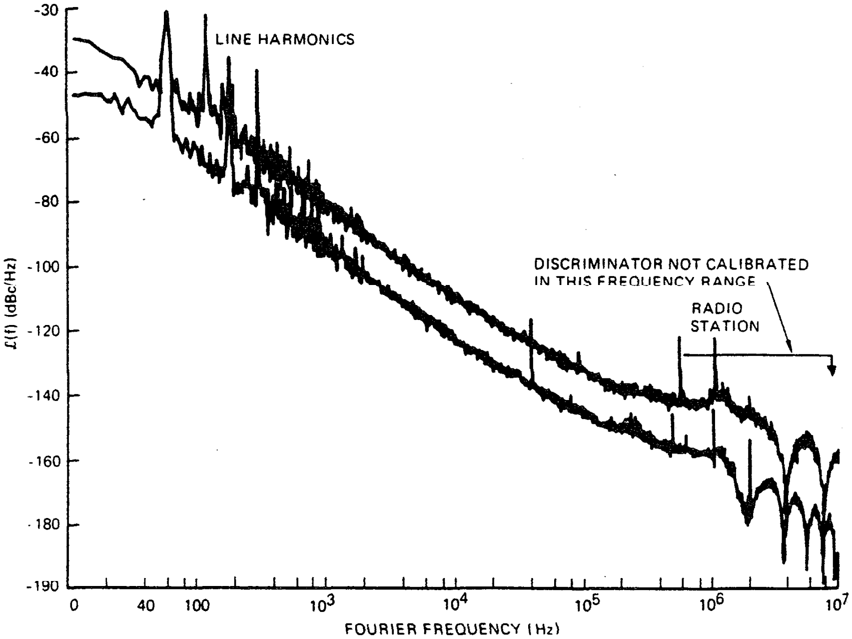


FIG. 22 Phase noise of 600-MHz oscillator multiplied to 2.4 GHz (From Lance *et al.*, 1977a.)

was about 500 nsec long, as noted by the first null, i.e., the reciprocal of the Fourier frequency of 2 MHz is the approximate differential time delay. Note that a shorter delay line (approximately 250 nsec differential) is used to measure the higher frequency because the delay-line discriminator calibration is valid only to a Fourier frequency at approximately 35% of the Fourier frequency at which the first null occurs, if a linear transfer function is assumed.

The actual transfer function of a delay-line discriminator (classic and rf bridge types) is sinusoidal, as shown in Fig. 23a. The baseband spectrum analyzer measures power in a finite bandwidth, and as a consequence it is possible to measure through a transfer-function null if the noise power does not change substantially over a spectrum-analyzer bandwidth. The following power relations then hold:

$$P_{\text{meas}}(\omega) = 1/\Delta\omega \int_{\omega-\Delta\omega/2}^{\omega+\Delta\omega/2} P(\omega')\,d\omega' \simeq \frac{P(\omega)}{\Delta\omega} \int_{\omega-\Delta\omega/2}^{\omega+\Delta\omega/2} d\omega' = P(\omega). \quad (98)$$

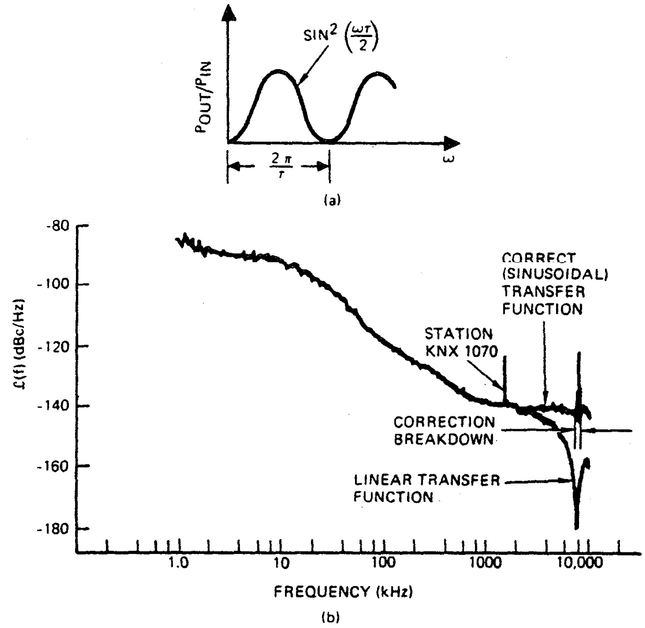


FIG. 23 Transfer functions for a delay-line rf bridge discriminator: (a) actual; (b) approximate (linear) and "correct" (sinusoidal). Phase noise: H.P. 8672A at 2.4 GHz.

Figure 23b shows the results using a linear approximation and the "correct" transfer function for a delay-line rf bridge discriminator. The correct transfer function breaks down close to the null because the signal level drops below the system's noise floor, as explained by Labaar (1982).

Using the sinusoidal transfer function in the calculator software gives correct results barring frequency intervals of 5 to 10 spectrum analyzer's bandwidths (10 × 30 = 300 kHz) centered at the transfer function nulls. These particular data were selected to illustrate the characteristics of the system. Recall that one can easily make the noise floor 40 dB lower using the rf bridge shown in Fig. 18.

C. DUAL DELAY-LINE DISCRIMINATOR

1. *Phase Noise Measurements*

The dual delay-line discriminator is shown in Fig. 24. This system was suggested by Halford (1975) as a technique for lowering the noise floor of the delay-line phase noise measurement system. The system consists of

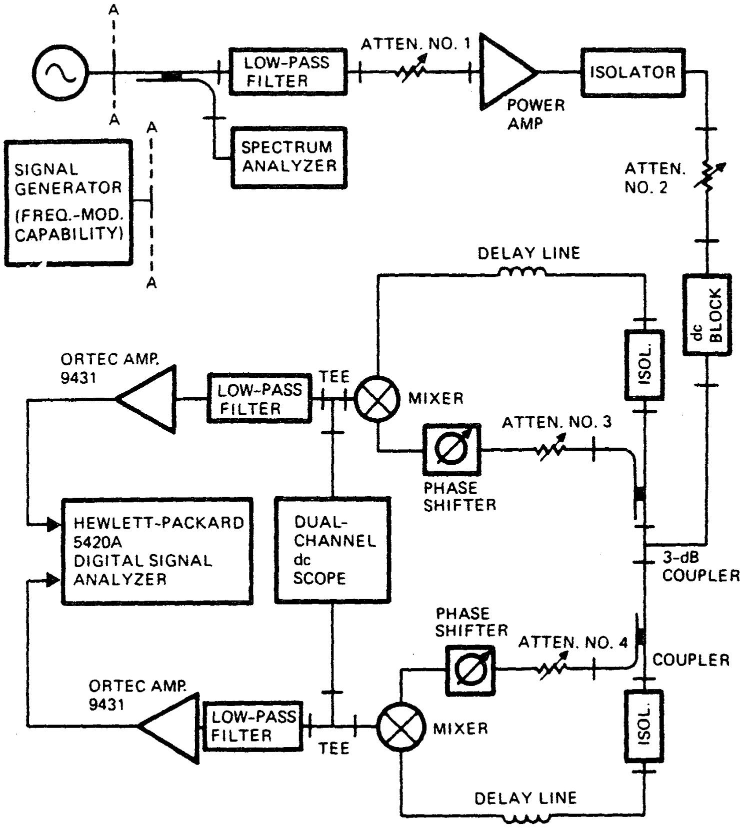


FIG. 24 A dual delay-line phase noise measurement system. (Courtesy Instrument Society of America.)

two differential delay-line systems. The single-oscillator signal is applied to both systems and cross-spectrum analysis is performed on the signal output from the two delay-line systems. Signal processing is performed with the Hewlett-Packard 5420A digital signal analyzer. The *cross spectrum* is obtained by taking the product of the Fourier transform of one signal and the complex conjugate of the Fourier transform of a second signal. It is a phase-sensitive characteristic resulting in a complex product that serves as a measurement of the relative phase of two signals. Cross spectrum gives a phase- and amplitude-sensitive measurement directly. By performing the product $Sy(f) \cdot Sx(f)^*$, a certain signal-to-noise enhancement is achieved.

The low-noise amplifiers preceding the digital signal analyzer are used when performing measurements at Fourier frequencies from 1 Hz to 25 kHz. The amplifiers are not used when performing measurements below the Fourier frequency of 1 Hz.

2. *Calibrating the Dual Delay-Line System*

Each delay line in the system is calibrated separately following the same basic procedure set forth in Section IV.B. The Hewlett-Packard 5420 measures the one-sided spectral density of frequency fluctuations in hertz squared per hertz. The spectral density of phase fluctuation in radians squared per hertz can be calculated as

$$S_{\delta\phi}(f) = S_{\delta v}(f)/f^2, \tag{99}$$

and

$$\mathcal{L}(f) = S_{\delta v}(f)/2f^2, \tag{100}$$

per hertz. The Hewlett-Packard 5420 measurement of $S_{\delta v}(f)$ in Hz^2/Hz must, therefore, be corrected by $1/2f^2$. However, the f^2 correction must be entered in terms of radian frequency ($\omega = 2\pi f$). This conversion is accomplished by

$$\mathcal{L}(f) = S_{\delta v}(f)(1/2f^2)(4\pi^2/4\pi^2) = [2\pi^2 S_{\delta v}(f)]/(\omega)^2 \tag{101}$$

per hertz since Eq. (100) can be stated in the following terms:

$$[2\pi^2 S_{\delta v}(f)]/4\pi^2 f^2.$$

Signal-to-noise enhancement greater than 20 dB has been obtained using the dual-channel delay-line system.

D. MILLIMETER-WAVE PHASE-NOISE MEASUREMENTS

1. *Spectral Density of Phase Fluctuations*

The delay line used as an FM discriminator is based, in principle, on a *nondispersive* delay line. However, a waveguide can be used as the delay line because the Fourier frequency range of interest is a small percentage of the operating bandwidth (seldom over 100 MHz), and the dispersion can be considered negligible.

The calibration and measurement are performed as set forth in Section IV.B. The modulation index m is usually established using the carrier-to-sideband ratio that uses amplitude modulation because millimeter sources are either unstable or cannot be modulated. The two approaches to measurements at millimeter frequencies are shown in Figs. 25 and 26. Figure 25 shows the *direct measurement* using a waveguide delay line. This system offers improved sensitivity if adequate input power is available. The rf bridge and delay-line portion of the system differs from Fig. 18 because pre- and

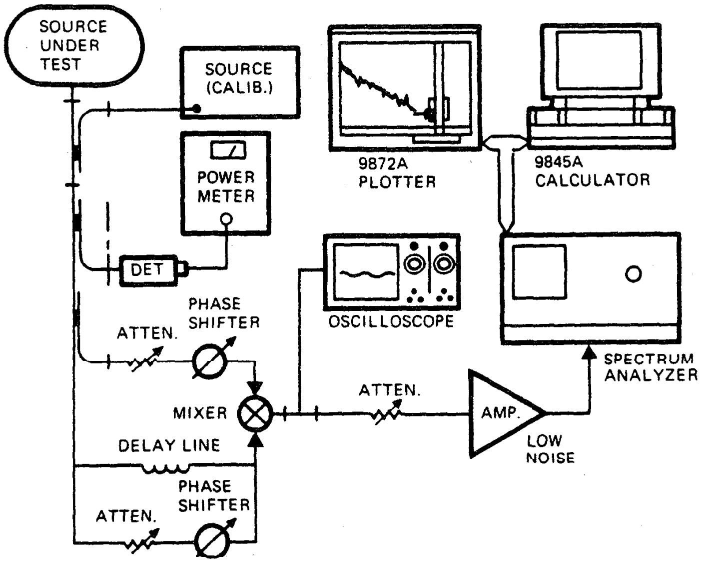


FIG. 25 Millimeter-wave phase noise measurements using a waveguide delay line. (From Seal and Lance, 1981.)

post-bridge amplifiers with appropriate gain are not available, so the sensitivity can equal the amount of carrier suppression.

Figure 26 shows the use of a harmonic mixer to downconvert to the convenient lower frequency where post-bridge amplifiers are available. The relatively low sensitivity to frequency drift that is characteristic of delay-line discriminators becomes an advantage here. A separate calibration generator is required, as shown in Fig. 25, and a power meter is used to assure proper power levels during the calibration process.

2. SPECTRAL DENSITY OF AMPLITUDE FLUCTUATIONS

AM noise measurements require equal electrical length in the two channels that supply the signals to the mixer. The delay line must be replaced with the necessary length of transmission line to establish the equal-length condition when the systems shown in Figs. 25 and 26 are used. The AM noise measurement system is calibrated and the noise measurements are performed directly in units of power for a direct measurement of $m(f)$ in dBc/Hz. $m(f)$ is the spectral density of one modulation sideband divided by the total signal power at a Fourier frequency difference f from the signal's average frequency ν_0. The system calibration establishes the detection characteristics in terms of total power output at the IF port of the mixer (detector).

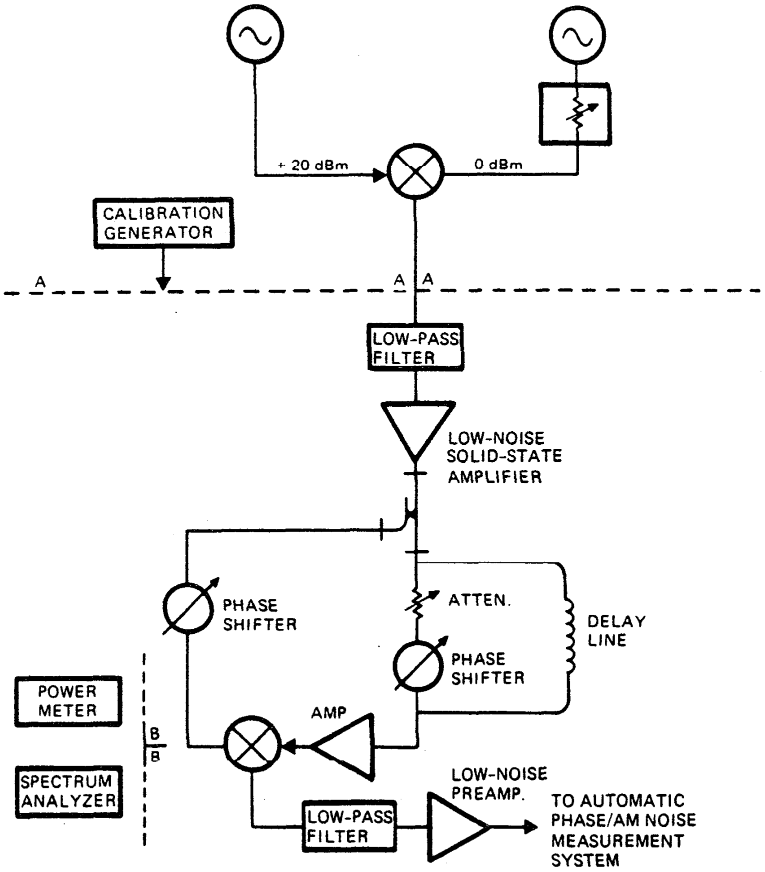


FIG. 26 A millimeter-wave hybrid phase noise measurement system that produces IF frequency and uses a delay-line discriminator at IF frequency. (From Seal and Lance, 1981.)

The AM noise measurements are performed according to the following.

(1) A known AM modulation (carrier–sideband ratio) must be established to calibrate this detector in terms of total power output at the IF port. The modulation must be low enough so that the sidebands are at least 20 dB below the carrier. This is to keep the total added power due to the modulation small enough to cause an insignificant change in the detector characteristics.

(2) The rf power levels are adjusted for levels of approximately 10 dBm at the reference port and 0 dBm at the test port of the mixer.

(3) Approximately 40 dB is set in the precision IF attenuator. The system is adjusted for an *out-of-phase quadrature condition.*

(4) The modulation frequency and power level are measured by the automatic baseband spectrum analyzer. The total carrier-power reference level is measured power, plus the carrier–sideband modulation ratio, plus the IF attenuator setting.

(5) The AM modulation is removed, the IF attenuator set to 0 dB, and the system re-checked to verify the out-of-phase quadrature (maximum dc output from the mixer IF port). Noise (V_n) is measured at the selected Fourier frequencies. A direct calculation of $\mathscr{m}(f)$ in dBc/Hz is

$$\begin{aligned}\mathscr{m}(f) = {} & [(\text{modulation power (dBm)} + \text{carrier–sideband ratio (dB)} \\ & + \text{IF attenuation (dB)} - \text{noise power (dBm)} + 2.5\ \text{dB} \\ & - 10\log(\text{BW})]. \end{aligned} \tag{102}$$

Figure 27 illustrates the measurements of AM and phase noise of two GUNN oscillators that were offset in frequency by 1 GHz. The measurements were performed using the coaxial delay-line system.

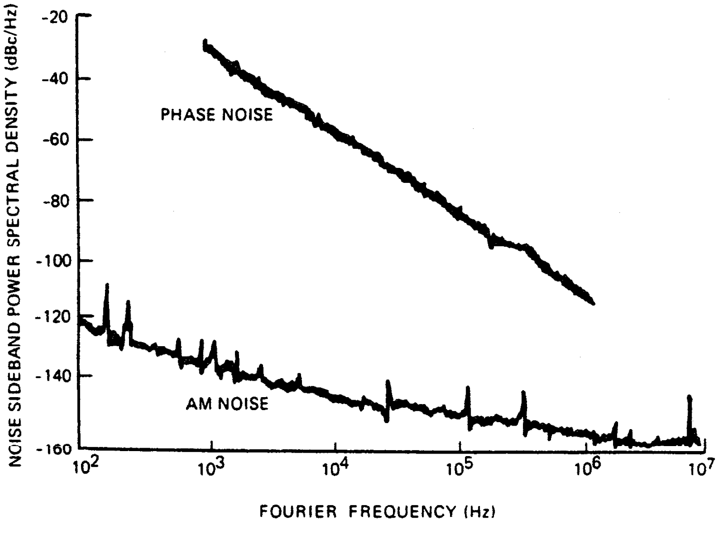


FIG. 27 Phase noise and AM noise of 40- and 41-GHz Gunn oscillators. (From Seal and Lance, 1981.)

V. Conclusion

The fundamentals and techniques for measurement of phase noise have been set forth for two basic systems. The two-oscillator technique provides the capability for measuring high-performance cw sources. The system sensitivity is superior to the single-oscillator technique for measuring phase noise very close to the carrier.

High-stability sources such as those used in frequency standards applications can be measured without using phase-locked loops. However, most microwave sources exhibit frequency instability that requires phase-locked loops to maintain the necessary quadrature conditions. The characteristics of the phase-locked loops must be evaluated to obtain the source phase noise characteristics. Also, in principle, one must have three sources at the same frequency to characterize a given source. If three sources are not available, one must assume that either one source is superior in performance or that they have equal phase noise contributions.

The single-oscillator technique employing the delay line as an FM discriminator has adequate sensitivity for measuring most microwave sources. The economic advantages of using this system include the fact that only one source is required, phase-locked loops are not required, system configuration is relatively inexpensive, and the system is inherently insensitive to oscillator frequency drift.

The single-oscillator technique using the delay-line discriminator can be adapted to measure the phase noise of pulsed sources. Pulsed sources have been measured at 94 GHz by F. Labaar at TRW, Redondo Beach, California.

ACKNOWLEDGMENTS

Our initial preparations for developing a phase noise measurement capability were the result of discussions with Dr. Jorg Raue of TRW. Our first phase noise development effort was assisting in the evaluation of phase noise measurement systems designed and developed by Bill Hook of TRW (Hook, 1973). The efforts of Don Leavitt of TRW were vital in initiating the measurement program.

We are very grateful to Dr. Donald Halford of the National Bureau of Standards in Boulder, Colorado, for his interest, consultations, and valuable suggestions during the development of the phase noise measurement systems at TRW.

We appreciate the measurement cross-checks performed by C. Reynolds, J. Oliverio, and H. Cole of the Hewlett-Packard Company, Dr. J. Robert Ashley of the University of Colorado, Colorado Springs, and G. Rast of the U.S. Army Missile Command, Redstone Arsenal, Huntsville, Alabama.

REFERENCES

Allen, D. W. (1966). *Proc. IEEE* **54**, 221–230.

Ashley, J. R., Searles, C. B., and Palka, F. M. (1968). *IEEE Trans. Microwave Theory Tech.* **MTT-16**(9), 753–760.

Ashley, J. R., Barley, T. A., and Rast, G. J. (1977). *IEEE Trans. Microwave Theory Tech.* **MTT-25**(4), 294–318.

Baghdady, E. J., Lincoln, R. N., and Nelin, B. D. (1965). *Proc. IEEE* **53**, 704–722.

Barnes, J. A. (1969). "Tables of Bias Functions, B_1 and B_2, for Variances Based on Finite Samples of Processes with Power Law Spectral Densities," National Bureau of Standards Technical Note 375.

Barnes, J. A., Chie, A. R., and Cutler, L. S. (1970). "Characterization of Frequency Stability," National Bureau of Standards Technical Note 394; also published in *IEEE Trans. Instrum. Meas.* **IM-20**(2), 105–120 (1971).

Brandenberger, H., Hadorn, F., Halford, D., and Shoaf, J. H. (1971). *Proc. 25th Ann. Symp. Freq. Control*, Fort Monmouth, New Jersey, pp. 226–230.

Culter, L. S., and Searle, C. L. (1966). *Proc. IEEE* **54**, 136–154.

Fisher, M. C. (1978). "Frequency Domain Measurement Systems," Paper presented at the 10th Annual Precise Time and Time Interval Applications and Planning Meeting, Goddard Space Flight Center, Greenbelt, Maryland. *

Halford, D. (1968). *Proc. IEEE* **56**(2), 251–258.

Halford, D. (1975). "The Delay Line Discriminator," National Bureau of Standards Technical Note 10, pp. 19–38.

Halford, D., Wainwright, A. E., and Barnes, J. A. (1968). *Proc. 22nd Ann. Symp. Freq. Control*, Fort Monmouth, New Jersey, 340–341.

Halford, D., Shoaf, J. H., and Risley, A. S. (1973). *Proc. 27th Ann. Symp. Freq. Control*, Cherry Hill, New Jersey, pp. 421–430.

Hook, W. R. (1973). "Phase Noise Measurement Techniques for Sources Having Extremely Low Phase Noise," TRW Defense and Space Systems Group Internal Document, Redondo Beach, California.

Hewlett-Packard Company (1965). "Frequency and Time Standards," Application Note 52. Hewlett-Packard Co., Palo Alto, California.

Hewlett-Packard Company (1970). "Computing Counter Applications Library," Application Notes 7, 22, 27, and 29. Hewlett-Packard Co., Palo Alto, California.

Hewlett-Packard Company (1976). "Understanding and Measuring Phase Noise in the Frequency Domain," Application Note 207. Hewlett-Packard Co., Loveland, Colorado.

Labaar, F. (1982). "New Discriminator Boosts Phase Noise Testing." *Microwaves* **21**(3), 65–69.

Lance, A. L. (1964). "Introduction to Microwave Theory and Measurements." McGraw-Hill, New York.

Lance, A. L., Seal, W. D., Mendoza, F. G., and Hudson, N. W. (1977a). *Microwave J.* **20**(6), 87–103.

Lance, A. L., Seal, W. D., Mendoza, F. G., and Hudson, N. W. (1977b). *Proc. 31st Annu. Symp. Freq. Control*, Atlantic City, New Jersey, pp. 463–483.

Lance, A. L., Seal, W. D., Halford, D., Hudson, N., and Mendoza, F. (1978). "Phase Noise Measurements Using Cross-Spectrum Analysis." Paper presented at the IEEE Conference on Electromagnetic Measurements, Ottawa, Canada.

Lance, A. L., Seal, W. D., and Labaar, F. (1982). *ISA Trans.* **21**(4), 37–44.

Meyer, D. G. (1970). *IEEE Trans.* **IM-19**(4), 215–227.

Ondria, J. (1968). *IEEE Trans. Microwave Theory Tech.* **MTT-16**(9), 767–781.

Ondria, J. G. (1980). *IEEE-MTT-S Int. Microwave Symp. Dig.*, pp. 24–25.

Payne, J. B., III (1976). *Microwave System News* **6**(2), 118–128.

Scherer, D. (1979). "Design Principles and Measurement Low Phase Noise RF and Microwave Sources." Hewlett-Packard Co., Palo Alto, California.

Seal, W. D., and Lance, A. L. (1981). *Microwave System News* **11**(7), 54–61.

Shoaf, J. H., Halford, D., and Risley, A. S. (1973). "Frequency Stability Specifications and Measurement." National Bureau of Standards Technical Note 632.

---

* See Appendix Note # 33

Tykulsky, A. (1966). *Proc. IEEE* **54**(2), 306.

Van Duzer, V. (1965). *Proc. IEEE-NASA Symp. Definition Meas. Short-Term Freq. Stability*, NASA SP-80, 269–272.

Walls, F. L., and Stein, S. R. (1977). *Proc. 31st Ann. Symp. Freq. Control*, Atlantic City, New Jersey, pp. 335–343.

Walls, F. L., Stein, S. R., Gray, J. E., Glaze, D. J., and Allen, D. W. (1976). *Proc. 30th Ann. Symp. Freq. Control*, Atlantic City, New Jersey, p. 269.

36th Annual Frequency Control Symposium - 1982

PERFORMANCE OF AN AUTOMATED HIGH ACCURACY PHASE MEASUREMENT SYSTEM

S. Stein, D. Glaze, J. Levine, J. Gray, D. Hilliard, D. Howe
Time and Frequency Division
National Bureau of Standards
Boulder, Colorado 80303

and

L. Erb
ERBTEC Engineering Inc.
Boulder, Colorado

Summary

A fully automated measurement system has been developed that combines many properties previously realized with separate techniques. This system is an extension of the dual mixer time difference technique, and maintains its important features: zero dead time, absolute phase difference measurement, very high precision, the ability to measure oscillators of equal frequency and the ability to make measurements at the time of the operator's choice.[1] For one set of design parameters, the theoretical resolution is 0.2 ps, the measurement noise is 2 ps rms and measurements may be made within 0.1 s of any selected time. The dual mixer technique has been extended by adding scalers which remove the cycle ambiguity experienced in previous realizations. In this respect, the system functions like a divider plus clock, storing the epoch of each device under test in hardware.

The automation is based on the ANSI/IEEE-583 (CAMAC) interface standard.[2] Each measurement channel consists of a mixer, zero-crossing detector, scaler and time interval counter. Four channels fit in a double width CAMAC module which in turn is installed in a standard CAMAC crate. Controllers are available to interface with a wide variety of computers as well as any IEEE-488 compatible device. Two systems have been in operation for several months. One operates 24 hours a day, taking data from 15 clocks for the NBS time scale, and the other is used for short duration laboratory experiments.

Review of the Dual Mixer Time Difference Technique

It is advantageous to measure time directly rather than time fluctuations, frequency or frequency fluctuationns. These measurements constitute a hierarchy in which the subsequently listed quantities may always be calculated from the previous ones. However, the reverse is not true when there are gaps in the measurements. In the past, frequency was usually not derived from time measurements for short sample times because time interval measurements could not be performed with adequate precision. The dual mixer technique, illustrated in Figure 1, made it possible to realize the precision of the beat frequency technique in time interval measurements.

The signals from two oscillators (clocks) are applied to two ports of a pair of double balanced mixers. Another signal synthesized from one of the oscillators is applied to the remaining two ports of the mixer pair. The input signals may be represented in the usual fashion

$$V_1(t) = V_{10} \sin [2\pi\nu_{10}t + \phi_1(t)],$$

$$V_2(t) = V_{20} \sin [2\pi\nu_{20}t + \phi_2(t)] \text{ and}$$

$$V_s(t) = V_{so} \cos [2\pi\nu_{so}t + \phi_s(t)]$$

where $\nu_{so} = \nu_{10}(1-1/R)$ and R is a constant usually called the heterodyne factor.

The low passed outputs of the two mixers are

$$V_{B1} = V_{B10} \sin [\Phi_1(t) - \Phi_s(t)] \text{ and}$$

$$V_{B2} = V_{B20} \sin [\Phi_2(t) - \Phi_s(t)] \text{ where}$$

$$\Phi(t) = 2\pi\nu_o t + \phi(t).$$

The time interval counter starts at time t_M when V_{B1} crosses zero in the positive direction and stops at time t_N, the time of the very next positive zero crossing of V_{B2}. Thus

$$\Phi_1(t_M) - \Phi_s(t_M) = 2M\pi \text{ and}$$

$$\Phi_2(t_N) - \Phi_s(t_N) = 2N\pi \text{ where}$$

N and M are integers.
Subtracting the two equations in order to compare the phases of oscillators 1 and 2, one obtains

$$\Phi_2(t_N) - \Phi_1(t_M) = \Phi_s(t_N) - \Phi_s(t_M) + 2(N-M)\pi.$$

The phase of an oscillator at time t_N may be written in terms of its phase at t_M and its average frequency over the interval $t_M < t_N$.

$$\Phi(t_N) = \Phi(t_M) + 2\pi[\bar{\nu}(t_M;t_N)](t_N-t_M) \text{ and}$$

when we apply this equation to both Φ_2 and Φ_S we find

$$\Phi_2(t_M)-\Phi_1(t_M) = 2(N-M)\pi$$
$$-2\pi[\bar{\nu}_{B2}(t_M;t_N)](t_N-t_M)$$

where $\nu_{B2} = \nu_2-\nu_s$.

Since M and N are not measureable with the equipment in Figure 1, the dual mixer technique has heretofore only been used to measure the phase difference between two oscillators modulo 2π. We denote the period of the time interval counter time base by τ_c and the number of counts recorded in a measurement by P. Then the phase difference between the two oscillators is given by

$$[\Phi_2(t_M)-\Phi_1(t_M)]\text{mod } 2\pi = -2\pi[\bar{\nu}_{B2}(t_M;t_N)]\tau_c P$$

Figure 2 illustrates the output of the measurement system over a period of time. If a measurement begins and ends without the time interval counter making a transition between zero and its maximum value, e.g., $t_a < t_M < t_N < t_b$, then the phase difference can be calculated from the data. If $t_a < t_M < t_b < t_N < t_c$, then the data must be corrected by 2π to calculate the phase difference. Experience has shown that there are many measurement situations for which the number of transitions of the time interval counter which occur between t_M and t_N cannot be known. For this reason, a modification has been developed which removes the ambiguity by measuring M and N.

Extended Dual Mixer Time Difference Measurement Technique

In order to configure the system to acquire complete phase information, two scalers are added to count the zero crossings of each mixer. Figure 3 is the block diagram of a two channel system. It is constructed from identical circuit modules and therefore contains an unused time interval counter. However, this design permits very straightforward and inexpensive extension to the comparison of an arbitrarily large number of oscillators with no need for switching any signals.

The counter outputs are combined to form the phase difference between oscillators.

$$\Phi_2(t_M)-\Phi_1(t_M) = 2(N_o-M_o)\pi + 2(N-M)\pi$$
$$-2\pi[\bar{\nu}_{B2}(t_M;t_N)]\tau_c P$$

The first term is a constant which represents the choice of the time origin and can be ignored. The last two terms and their sum are plotted in Figure 4.

The average beat frequency $\bar{\nu}_{B2}$ $(t_M;t_N)$ cannot be known exactly. However, it may be estimated with sufficient precision from the previous pair of measurements designated ' and ". The average frequency is approximately

$$\bar{\nu}_{B2}(t_M;t_N) \cong (N''-N')/[R(M''-M')/\nu_{10} + \tau_c(P''-P')]$$

provided that it changes sufficiently slowly compared to the interval $t_M<t_N$. A typical value for this error will be given in the following section.

Hardware Implementation

All measurement channels consist of a mixer, zero-crossing detector, scaler and time interval counter. Four such circuits can be built in a double width CAMAC module. The system is easily expanded to compare many oscillators and a *complete system for making phase* comparisons among four clocks is shown in Figure 5. We have chosen parameters which are reasonable for comparing state-of-the-art atomic standards. Thus, the synthesizer is offset 10Hz below oscillator # 1 and R = 5×10^5. The outputs from both mixers are approximately 10Hz. The noise bandwidth is 100 Hz. The time interval counter is twice the frequency of oscillator #1 or approximately 10 MHz. The quantization error is $1/2R = 10^{-6}$ cycle or 0.2ps which is a factor of ten smaller than the measurement noise. As stated earlier, an error will result from frequency changes which violate the constancy assumption used to estimate ν_{B2}. A change in ν_2 by 10^{-10} during the interval between two measurements will result in a time deviation error of 10ps. *Thus, one must make* more closely spaced measurements for oscillators which have large dynamic frequency changes than for more stable devices. Two other sources of inaccuracy are the sensitivities to the amplitude and phase of the common oscillator. Figure 6 shows the measured value of $x = \phi/2\pi\nu_o$ as a function of the amplitude of the input signal and the phase of the synthesizer.

The new measurement system has many desirable features and properties:

(1) It has very high resolution, limited by the internal counters to 0.2 ps and by noise to approximately 2 ps.

(2) It has much lower noise than divider based measurement systems. However compromises made to achieve low cost, low power, small size and automatic operation degrade the performance compared to state-of-the-art systems for comparing 2 oscillators.

(3) The operation is fully automatic.

(4) NBS has developed a detailed operating manual for the equipment and software.[3]

(5) All oscillators in the range of 5 MHz ± 5 Hz may be compared. Other carrier frequencies such as 1 MHz, 5.115 MHz, 10 MHz and 10.23 MHz are also usable. However, different carrier frequencies may not be mixed on the same system. The system has been successfully tested with an oscillator offset 4.6 Hz from nominal 5MHz. Measurements were made at intervals of 2 hours between which the system had to accumulate approximately $2 \times 10^{6}\pi$. The system has also been tested with an oscillator offset 4×10^{-9}, and no errors were detected during a period of 40 days.

(6) All sampling times in the range of 1 second to 16 days with a resolution of 0.1 second are possible. Measurements may be made on command or in a preprogrammed sequence.

(7) Measurements are synchronized precisely, i.e. at the picosecond level, with the reference clock. They may therefore be synchronized with important user system events, such as the switching times of a FSK or PSK system.

(8) All oscillators are compared synchronously and all measurements are performed within a maximum interval of 0.1 second. As a result, the phase of any oscillator needs to be interpolated to the chosen measurement time for an interval of 0.1 second maximum. This capability, which is not present in either single heterodyne measurement systems or switched measurement systems eliminates a source of "measurement" error which is generally much larger than the noise induced errors. For example, interpolation of the phase of a high performance Cs clock ($\sigma_y \sim 10^{-11}/\tau^{1/2}$) over a period of 3 hours would produce approximately 1.5 ns phase uncertainty. To maintain 4 ps accuracy requires measurements simultaneous to 0.1s.

(9) There are no phase errors due to the switching of rf signals since there is no switching anywhere in the analog measurement system.

(10) No appreciable phase errors are introduced when it is necessary to change the reference clock since, as shown in Figure 6, the peak error due to changes in synthesizer phase is 20 ps.

(11) The measurement system is capable of measuring its own phase noise when the same signal is applied to two input ports. Figure 7 shows the phase deviations between two such channels over a period of 75,000 seconds and Figure 8 is the corresponding Allan variance plot. Figure 9 shows the phase deviations between 2 input channels over a period of 40 days.

(12) Since the IEEE-583 (CAMAC) interface standard has been followed for all the custom hardware, the system may be easily interfaced to almost any instrument controller. NBS has already tested the system using a large minicomputer, a small minicomputer and a desk top calculator. Interfaces between IEEE-583 and IEEE-488 controllers are available and have been used successfully.

(13) The system is capable of camparing a very large number of oscillators at a reasonable cost per device.

There are also disadvantages to this measurement system. The most important are:

(1) The complexity of the hardware is greater than for some systems. It is possible that this will reduce reliability.

(2) A high level of redundancy is difficult to achieve. The system design stresses size, power, convenience and cost, resulting in an increase in the number of possible single point failure mechanisms compared to some other techniques. For example, a CAMAC power supply failure will result in a loss of data for all devices being measured.

(3) A substantial committment is required in both specialized hardware and software.

(4) If an oscillator under test experiences a phase jump which exceeds 1 cycle, the measurement system records a jump with incorrect absolute magnitude. As a result, it may not be applicable to signals which are frequency modulated with discontinuous phase steps larger than 2π.

Conclusions

We have demonstrated a new phase measurement system with very desirable properties: All oscillators in the range of 5MHz ± 5Hz may be measured directly. The sampling times are only restricted by the requirement that they exceed one second. The noise floor is $\sigma_y(2,\tau) = 3 \times 10^{-12}/\tau$ in short term and the time deviations are less than 100 ps. All circuitry is designed as modules which allows expansion at modest cost. Compatibility with a variety of computers is insured through the use of the IEEE-583 interface and adapters are available to permit use with an IEEE-488 controller. The system makes it feasible to make completely automated phase measurements at predetermined times on large numbers of atomic clocks. It's own noise is one-hundred times less than the state-of-the-art in clock performance. It will be used in the near future to make all measurement needed to compute NBS atomic time, but it will also be very valuable for any laboratory which uses three or more atomic clocks.

References

1. D. W. Allan, "The measurement of frequency

and frequency stability of precision oscillators," NBS Tech. Note 669 (1975).
2. "CAMAC instrumentation and interface standards," Institute of Electrical and Electronic Engineers, Inc., 345 E. 47th St. New York, NY 10017.
3. D. J. Glaze and S. R. Stein, "Picosecond time difference measurements utilizing CAMAC based ANSI/IEEE -488 data acquisition hardware", NBS Tech Note 1056 (in preparation).

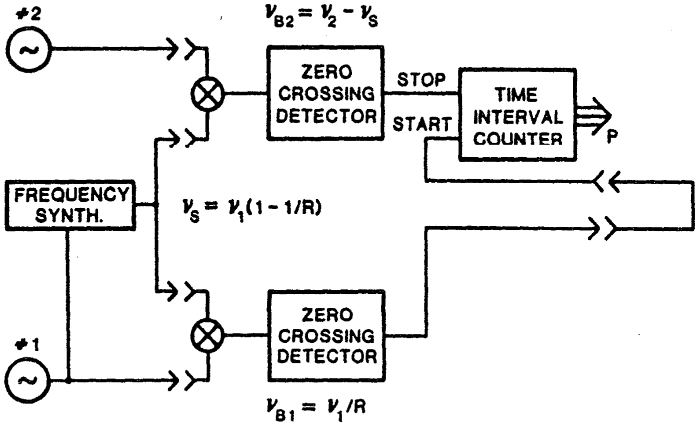


Figure 1. Dual Mixer Time Difference Measurement System

$$[\Phi_2(t_M)-\Phi_1(t_M)]\bmod 2\pi = -2\pi[\bar{\nu}_{B2}(t_M;t_N)]\tau_c P$$

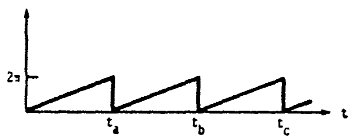


Figure 2. Dual Mixer Data

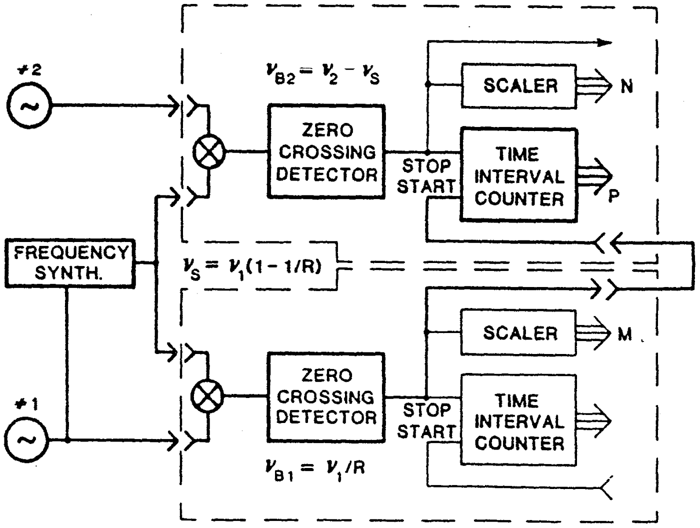


Figure 3. Extended Dual Mixer Time Difference Measurement System

$$\Phi_2(t_M)-\Phi_1(t_M) = 2(N_o-M_o)\pi + 2(N-M)\pi - 2\pi[\bar{\nu}_{B2}(t_M;t_N)]\tau_c P$$

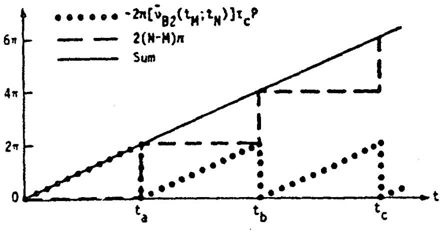
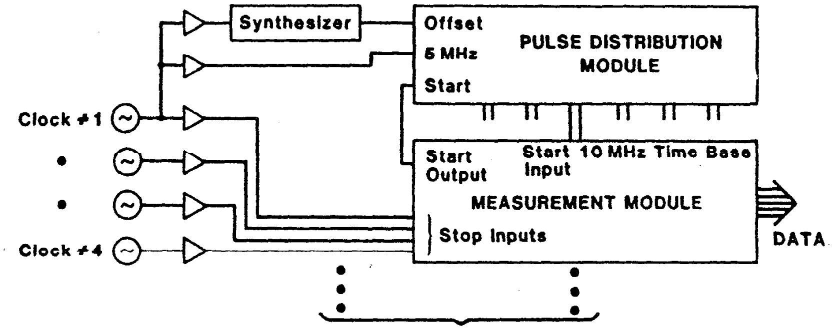


Figure 4. Extended Dual Mixer Data



6 X 4 = 24 clocks maximum requires one pulse distribution module and six measurement modules maximum

Figure 5. System Block Diagram

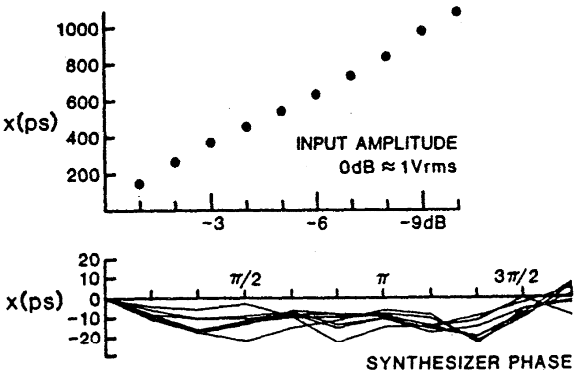


Figure 6. Measured Time Difference vs. Input Amplitude and Synthesizer Phase

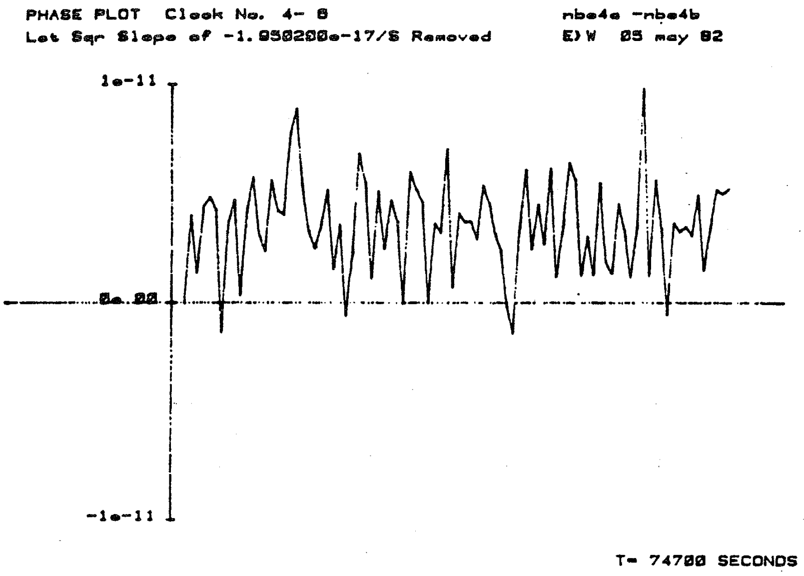


Figure 7. Raw Phase Data for Two Channels Driven from the Same Source

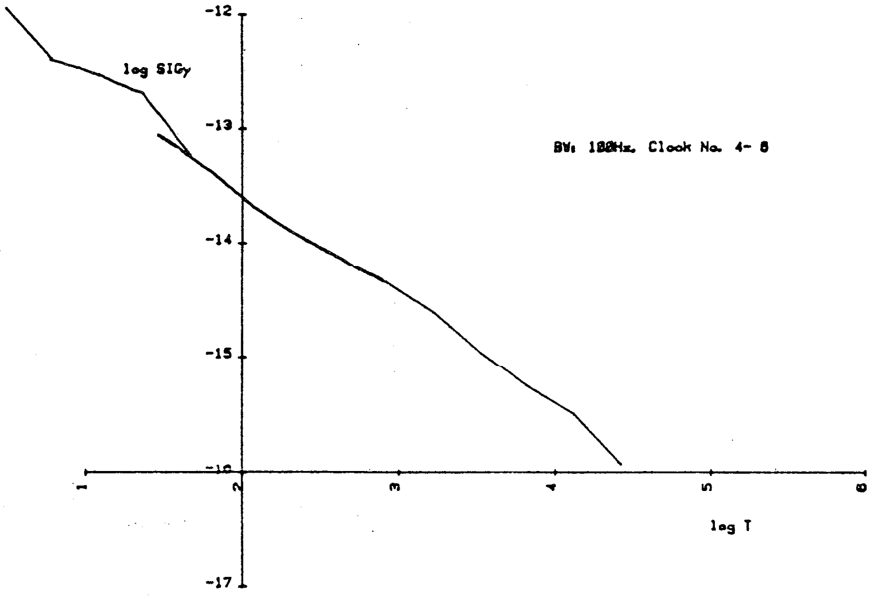


Figure 8. Noise Floor of Measurement System

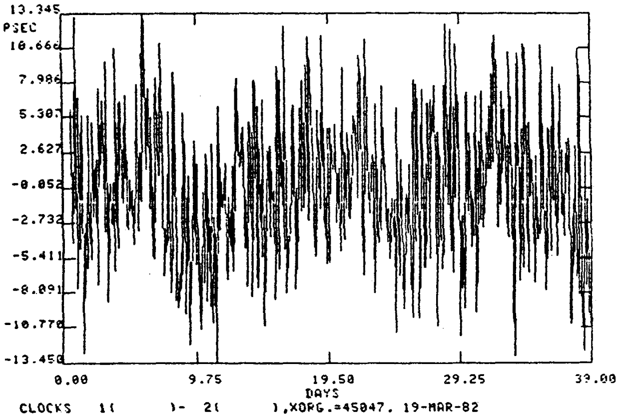


Figure 9. Raw Phase Data for Two Channels Driven from the Same Source

BIASES AND VARIANCES OF SEVERAL FFT SPECTRAL ESTIMATORS AS A FUNCTION OF NOISE TYPE AND NUMBER OF SAMPLES

F. L. Walls, Time and Frequency Division,
National Institute of Standards and Technology, Boulder, CO 80303

D. B. Percival, Applied Physics Laboratory,
University of Washington, Seattle, WA 98106

W. R. Irelan, Irelan Electronics,
412 Janet, Tahlequah, OK 74464

From: *Proceedings of the 43rd Annual Symposium on Frequency Control,* 1989.

Abstract

We theoretically and experimentally investigate the biases and the variances of Fast Fourier transform (FFT) spectral estimates with different windows (data tapers) when used to analyze power-law noise types f^0, f^{-2}, f^{-3} and f^{-4}. There is a wide body of literature for white noise but virtually no investigation of biases and variances of spectral estimates for power-law noise spectra commonly seen in oscillators, amplifiers, mixers, etc. Biases (errors) in some cases exceed 30 dB. The experimental techniques introduced here permit one to analyze the performance of virtually any window for any power-law noise. This makes it possible to determine the level of a particular noise type to a specified statistical accuracy for a particular window.

I. Introduction

Fast Fourier transform (FFT) spectrum analyzers are very commonly used to estimate the spectral density of noise. These instruments often have several different windows (data tapers) available for analyzing different types of spectra. For example, in some applications spectral resolution is important; in others, the precise amplitude of a widely resolved line is important; and in still other applications, noise analysis is important. These diverse applications require different types of windows.

We theoretically and experimentally investigate the biases and variances of FFT spectral estimates with different windows when used to analyze a number of common power-law noise types. There is a wide body of literature for white noise but virtually no investigation of these effects for the types of power-law noise spectra commonly seen in oscillators, amplifiers, mixers, etc. Specifically, we present theoretical results for the biases associated with two common windows — the uniform and Hanning windows — when applied to power-law spectra varying as f^0, f^{-2} and f^{-4}. We then introduce experimental techniques for accurately determining the biases of any window and use them to evaluate the biases of three different windows for power-law spectra varying as f^0, f^{-2}, f^{-3} and f^{-4}. As an example we find with f^{-4} noise that the uniform window can have errors ranging from a few dB to over 30 dB, depending on the length of span of the f^{-4} noise.

We have also theoretically investigated the variances of FFT spectral estimates with the uniform and Hanning windows (confidence of the estimates) as a function of the power-law noise type and as a function of the amount of data. We introduce experimental techniques that make it relatively easy to independently determine the variance of the spectral estimate for virtually any window on any FFT spectrum analyzer. The variance that is realized on a particular instrument depends not only on the window but on the specific implementation in both hardware and software. We find that the variance of the spectral density estimates for white noise, f^0, is very similar for three specific windows available on one instrument and almost identical to that obtained by standard statistical analysis. The variances for spectral density estimates of f^{-4} noise are only 4% higher than that of f^0 noise for two of the windows studied. The third window — the uniform window — does not yield usable results for either f^{-3} or f^{-4} noise.

Based on this work it is now possible to determine the minimum number of samples necessary to determine the level of a particular noise type to a specified statistical accuracy as a function of the window. To our knowledge this was previously possible only for white noise — although the traditional results are generally valid for noise that varied as $f^{-\beta}$, where β was equal to or less than 4.

II. Spectrum Analyzer Basics

The spectrum analyzer which was used in the experimental work reported here is fairly typical of a number of such instruments currently available from various manufacturers. The basic measurement process generally consists of taking a string of $N_s = 1024$ digital samples of the input wave form, which we represent here by $X_1, X_2, \ldots, X_{N_s}$. The basic measurement period was 4 ms. This yields a sampling time $\Delta t = 3.90625\ \mu s$. Associated with the FFT of a time series with N_s data points, there are usually $(N_s/2) + 1 = 513$ frequencies

$$f_j = \frac{j}{N_s \Delta t}, \qquad j = 0, 1, \ldots, N_s/2.$$

The fundamental frequency f_1 is 250 Hz, and the Nyquist frequency $f_{N_s/2}$ is 128 kHz. Since the spectrum analyzer uses an anti-aliasing filter which significantly distorts the high frequency portion of the spectrum, the instrument only displays the measured spectrum for the lowest 400 nonzero frequencies, namely, $f_1 = 250$ Hz, $f_2 = 500$ Hz, …, $f_{400} = 100$ kHz.

The exact details of how the spectrum analyzer estimates the spectrum for $X_1, \ldots, X_{N_s}$ are unfortunately not provided in the documentation supplied by the manufacturer, so the following must be regarded only as a reasonable guess on our part as to its operation (see [1] for a good discussion on the basic ideas behind a spectrum analyzer; two good general references for spectral analysis are [2] and [4]). The sample mean,

$$\bar{X} \equiv \frac{1}{N_s} \sum_{t=1}^{N_s} X_t,$$

is subtracted from each of the samples, and each of these "demeaned" samples is multipled by a window h_t (sometimes called a data taper) to produce

$$X_t^{(h)} = h_t \left(X_t - \bar{X} \right).$$

The spectral estimate,

$$\hat{S}_1(f_j) = \Delta t \left| \sum_{t=1}^{N_s} X_t^{(h)} e^{-i 2\pi f_j t \Delta t} \right|^2, \qquad j = 0, 1, \ldots, N_s/2.$$

is then computed using an FFT algorithm.

The subscript "1" on $\hat{S}_1(f_j)$ indicates that this is the spectral estimate formed from the first block of N_s samples. A similar spectral estimate $\hat{S}_2(f_j)$ is then formed from the second block of contiguous data $X_{N_s+1}, X_{N_s+2}, \ldots, X_{2N_s}$. In all, there are N_b different spectral estimates from N_b contiguous blocks, and the spectrum analyzer averages these together to form

$$\hat{S}(f_j) \equiv \frac{1}{N_b} \sum_{k=1}^{N_b} \hat{S}_k(f_j). \tag{1}$$

It is the statistical properties of $\hat{S}(f_j)$ with which we are concerned in this paper.

Unfortunately some important aspects of the windows are not provided in the documentation for the instrument. One important detail is the manner in which the window is normalized. There are two common normalizations:

$$\sum_{t=1}^{N_s} \left[h_t\left(X_t - \bar{X}\right)\right]^2 = \frac{1}{N_s} \sum_{t=1}^{N_s} \left(X_t - \bar{X}\right)^2$$

and

$$\sum_{t=1}^{N_s} h_t^2 = 1. \tag{2}$$

The first of these is common in engineering applications because it ensures that the power in the windowed samples $X_t^{(h)}$ is the same as in the original demeaned samples; the second is equivalent to the first in expectation and is computationaly more convenient, but it can result in small discrepancies in power levels. Either normalization affects only the level of the spectral estimate and not its shape.

There are three windows built into the spectrum analyzer used here. The first is the uniform (rectangular, default) window $h_t^{(U)} = 1/\sqrt{N_s}$. The second is the Hanning data window, for which there are several slightly different definitions in the literature. In lieu of specific details, we assume the following symmetric definition:

$$\begin{aligned} h_t^{(H)} &= C^{(H)}\left(1 - \cos\frac{2\pi(t-0.5)}{N_s}\right), \quad 1 \le t \le \frac{N_s}{2}, \\ &= h_{N_s-t+1}^{(H)}, \quad \frac{N_s}{2} + 1 \le t \le N_s; \end{aligned}$$

ere $C^{(H)}$ is a constant which forces the normalization in Equation (2). The third window is a proprietary "flattened peak" window, about which little specific information is available (it is evidently designed to accurately measure the heights of peaks in a spectrum).

III. Expected Value and Bias of Spectral Estimates

III.A. Theoretical Analysis

We need to assume a noise model for the X_t's in order to determine the statistical properties of $\hat{S}(f_j)$ in Equation (1). We consider three different models, each of which is represented in terms of a Gaussian white noise process ϵ_t with mean zero and variance σ_ϵ^2. The second-order properties of each model are given by a spectral density function $S(\cdot)$ defined over the interval $[-1/(2\Delta t), 1/(2\Delta t)]$ in cycles/Δt. The first model is a discrete parameter, white noise process (f^0 noise):

$$X_t = \epsilon_t \quad \text{and} \quad S(f) = \sigma_\epsilon^2 \Delta t.$$

The second model is a discrete-parameter, random-walk process (nominally f^{-2} noise):

$$X_t = \sum_{s=1}^{t} \epsilon_s \quad \text{and} \quad S(f) = \frac{\sigma_\epsilon^2 \Delta t}{4\sin^2(\pi f \Delta t)}.$$

The third model is a discrete-parameter, random-run process (nominally f^{-4} noise):

$$X_t = \sum_{r=1}^{t} \sum_{s=1}^{r} \epsilon_s \quad \text{and} \quad S(f) = \frac{\sigma_\epsilon^2 \Delta t}{16\sin^4(\pi f \Delta t)}.$$

Continuous parameter versions of these three models have been used extensively in the literature as models for noise commonly seen in oscillators.

For each of the three models we have derived expressions for $E\{\hat{S}(f)\}$, the expected value of $\hat{S}(f)$. These expressions depend on the window h_t, the number of samples N_s in each block and — in the case of a random-run process — the number of blocks N_b. The details behind these calculations will be reported elsewhere [3]; here we merely summarize our conclusions for the three models in combination with the uniform and Hanning windows and $N_s = 1024$.

First, for a white noise process,

$$E\{\hat{S}(f_j)\} = S(f_j), \qquad j = 1, 2, \ldots, 512,$$

when the uniform window is used. For the Hanning window, the above equality also holds to a very good approximation for $2 \le j \le 511$ and to within 0.8 dB for $j = 1$ and 512 (the latter is of no practical importance since the highest frequency index given by the spectrum analyzer is $j = 400$). These theoretical calculations agree with our experimental data except at f_1 (see Table 1).

Second, for a random-walk process,

$$E\{\hat{S}(f_j)\} = 2S(f_j), \qquad j = 1, 2, \ldots, 512,$$

when the uniform window is used, i.e., the expected value is *twice* what it should be at all frequencies. This theoretical result has been verified by Monte Carlo simulations, but it does *not* agree with our experimental data, which shows no significant level shift in the estimated spectrum. The source of this discrepancy is currently under investigation, but it may be due to either (a) factors in the experimental data which effectively make it band-limited, random-walk noise, i.e., its spectral shape is markedly different from f^{-2} for, say, $0 < f < f_1$ or (b) an incorrect guess on our part as to how the spectral estimate is normalized by the spectrum analyzer. For the Hanning window, we found that

$$E\{\hat{S}(f_j)\} = \begin{cases} 1.08S(f_j) & j = 1; \\ 1.48S(f_j) & j = 2; \\ 1.15S(f_j) & j = 3; \\ 1.07S(f_j) & j = 4; \\ 1.04S(f_j) & j = 5; \\ S(f_j) & 6 \le j \le 511 \text{ to within } 3\%, \end{cases}$$

i.e., $\hat{S}(f_j)$ is essentially an unbiased spectral estimate except for the lowest few frequencies. This theoretical result has been verified by Monte Carlo simulations and also agrees in general with our experimental data.

Third, for a random-run process,

$$E\{\hat{S}(f_j)\} = C_{N_b} f^{-2}, \qquad 1 \le j \le 400,$$

to a good appoximation when the uniform window is used, where C_{N_b} is a constant which depends on the number of blocks N_b and increases as N_b increases. Thus the shape of $E\{\hat{S}(f_j)\}$ follows that of a random-walk process (f^{-2}) rather than that of a random-run process (f^{-4}). This shape has been verified experimentally (see the next subsection), but the dependence of the level on N_b has not. The increase in level of $E\{\hat{S}(f_j)\}$ as N_b increases is due to the fact that the expected value of the sample variance of a block of N_s samples increases with time — by contrast, it is constant with time for the white noise and random-walk cases. For the Hanning window, we found that

$$E\{\hat{S}(f_j)\} = C'_{N_b} S(f_j), \qquad 4 \le j \le 400,$$

to a good approximation, where again C'_{N_b} is a constant — different from C_{N_b} — which depends on the number of blocks N_b and increases as N_b increases. For frequencies less than f_4 the theoretical results indicate significant (greater than 4%) distortion in the shape, but these do not agree in detail with the experimental values reported in Table 1. For $f_j \ge f_4$ the shape has been verified experimentally, but the dependence of the level on N_b has not. The discrepancy in level between the theoretical and experimental results is yet to be resolved, but it is probably due to a mismatch between the assumed random-run model and the true spectrum for the data (possibly band-limited random-run).

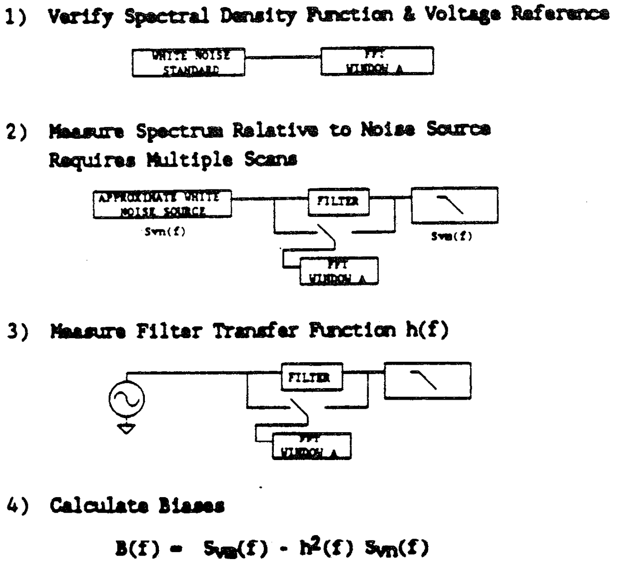


Figure 1. Outline of measurement procedure for determining the biases in spectral estimators.

III.B. Experimental Determination

The following procedure can be used to experimentally determine the bias in the spectral estimate of any noise spectrum using any window in a particular instrument. The basic concept is to implement a filter that, when applied to white noise, mimics the approximate noise spectra of interest and then measures the level of the white noise and the filter transfer function in a way which has high precision and accuracy as illustrated in Figure 1. First, the level of a known white noise is measured over a convenient range. The higher the frequency span the faster that this is accomplished. Obviously, the chosen range must be one over which the noise source is accurate. To obtain a precision of order 0.2 dB generally requires 1000 samples. This measurement verifies that the spectral density function and the internal reference voltage of the FFT are accurately calibrated and working properly. Virtually all of the windows accurately determine the value of white noise if the first few channels are ignored as explained above. Figure 2 shows the measurement of a noise source, which has been independently determined to have a noise spectral density of 99.8 dBV/Hz by the three windows. (Appendix A shows the circuit diagram for this noise source which has an accuracy of better than 0.2 dB for frequencies from 20 to 20 kHz.)

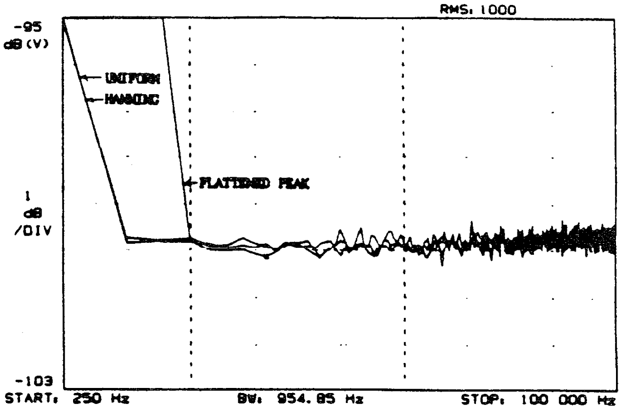
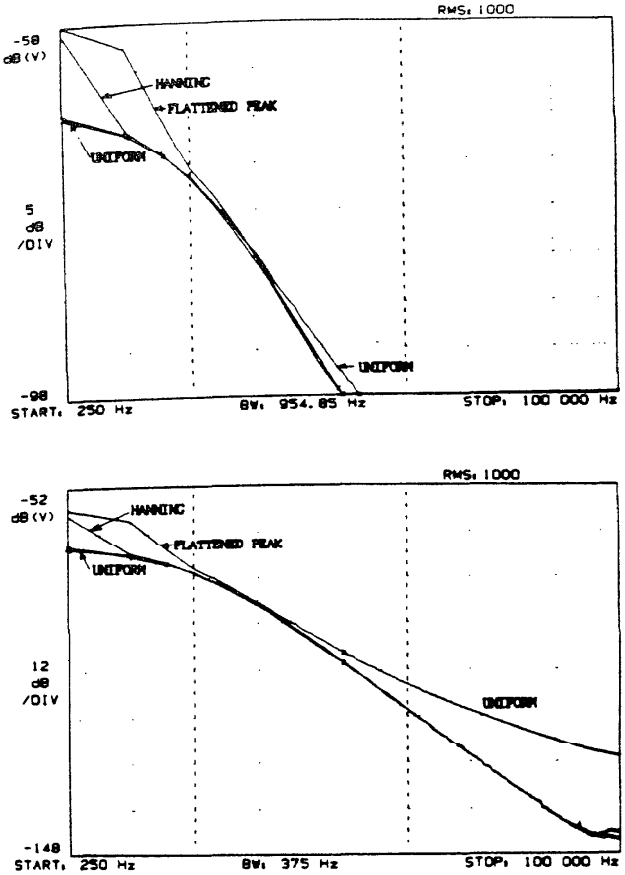


Figure 2. Spectral estimation of a white noise standard using the uniform, Hanning and the proprietary "flattened peak" windows.

Second, an approximately flat spectrum is measured over the frequency range of interest. It is not important if there are small variations in the level that change slowly over the frequency span. Third, the transfer function of the filter is determined for the frequencies of interest using a very narrow spectral source (typically an audio oscillator is sufficient). The very narrow source is accurately measured by the window since there is no problem with either high frequency or low frequency noise biasing the estimate. The use of a window with a flattened peak response is helpful but not necessary if the frequency source is sufficiently stable. This transfer function is then applied to the measured white noise spectrum in step two above. This yields a very accurate value for the "true" spectral density of the white noise source as measured through the filter. This "true value" is then compared to that obtained by the FFT analyzer. The difference between that measured in steps two and three and that measured directly with the FFT is the bias in the spectral estimate for that particular window and noise type. The accuracy of this approach comes from the fact that the calibration has been broken up into steps that can individually be determined with high precision and very small bias. The primary assumption is that the FFT analyzer is linear. Even this assumption can be checked by using precision attenuators. If the known white noise in step one does not extend to the frequencies of interest, then there is an additional assumption that the FFT is flat with frequency. This assumption is nearly always good except perhaps near the last few channels where the effect of the antialiasing filter might cause small inaccuracies.

Figures 3a and 3b show the "true" spectral estimate and the estimates as measured on a particular instrument using the uniform, Hanning, and the instrument's proprietary "flattened peak" windows for noise that varies as f^{-4} over much of the



Figures 3a (top) and 3b (bottom). Difference between the true spectrum (top) which varies approximately as f^{-4} and that estimated by the uniform, Hanning and proprietary "flattened peak" windows (bottom).

range from 1 kHz to 100 kHz. The scan is 0 to 100 kHz, and 1000 samples were taken for all curves. Note the considerable difference between the spectral estimates for channels 1 to 3 for the Hanning and proprietary "flattened peak" windows. These results confirm the theoretical calculations above showing that, for the Hanning window, the first 3 channels should be ignored. For the "flattened peak" window, the first 14 channels should be ignored. For both f^{-3} and f^{-4} noise, the uniform window does not yield usable spectral estimates over *any* portion of the scan. Note in this example that at frequencies above 80 kHz there is a small step in the spectral estimates. This is due to digitizing errors of the signal due to quantization. If the digitizer had more bits, these errors would not occur. This problem of dynamic range is common whenever the spectrum of interest covers many decades. The usual solution is to use filters to divide the spectrum into various frequency range segments which are suitable for the dynamic range of the FFT.

Table 1 summarizes the measured experimental biases in the spectral estimates of a particular instrument with three different windows for power-law noise types varying from f^0 to f^{-4}. This covers most of the random types of noise found in oscillators and signal processing equipment. We do *not* advocate using the biases reported in this table to correct data — they only indicate which channels should not be relied upon for data analysis.

Table 1. Approximate Biases in FFT Spectral Estimates

noise type f^0

channel #	uniform	Hanning	flattened peak
1	19.6 dB	19.6 dB	20.1 dB
2	small	small	16.7 dB
3	↓	↓	7.2 dB
4			small
5			↓

noise type f^{-4}

channel #	uniform	Hanning	flattened peak
1	unusable	8.6 dB	10.0 dB
2		0.4 dB	9.1 dB
3		0.4 dB	4.0 dB
4		small	1.2 dB
5		↓	1.1 dB
6			1.1 dB
7			1.0 dB
8			0.8 dB
9			0.6 dB
10			0.6 dB
11			0.5 dB
12			0.4 dB
13			0.4 dB
14			small
15			↓

IV. Variances of Spectral Estimates

IV.A. Theoretical Analysis

We have derived expressions for $var\{\hat{S}(f)\}$ — the variance of $\hat{S}(f)$ — for each of the three models considered in Section III.A. These expressions depend primarily on the number of blocks N_b. Again, the details behind these calculations will be reported elsewhere [3].

First, for a white noise process, the uniform window yields

$$var\{\hat{S}(f_j)\} = S^2(f_j)/N_b, \qquad 1 \le j \le 511,$$

while the Hanning window yields

$$var\{\hat{S}(f_j)\} = \begin{cases} 0.69S^2(f_j)/N_b, & j = 1; \\ S^2(f_j)/N_b, & 2 \le j \le 510; \\ 1.03S^2(f_j)/N_b, & j = 511. \end{cases}$$

These results are consistent with our experimental results and with standard statistical theory.

Second, for a random-walk process, the uniform window yields

$$var\{\hat{S}(f_j)\} = 5S^2(f_j)/N_b, \quad 1 \le j \le 511.$$

while the Hanning window yields

$$var\{\hat{S}(f_j)\} = \begin{cases} 1.30S^2(f_j)/N_b, & j = 1; \\ 2.20S^2(f_j)/N_b, & j = 2; \\ 1.31S^2(f_j)/N_b, & j = 3; \\ 1.15S^2(f_j)/N_b, & j = 4; \\ 1.09S^2(f_j)/N_b, & j = 5; \\ 1.06S^2(f_j)/N_b, & j = 6; \\ 1.04S^2(f_j)/N_b, & j = 7; \\ S^2(f_j)/N_b, & 8 \leq j \leq 511 \text{ to within } 3\%. \end{cases}$$

Except for the few lowest frequencies, the results for the Hanning window agree with our experimental results and with standard statistical theory; however, the factor of five in the variance for the uniform window disagrees with our experiments and with standard theory (although it has been verified by Monte Carlo techniques). The cause of this discrepancy is under investigation, but we think it is due to the band-limited nature of the experimental data.

Third, for a random-run process, the variance computations are not useful since the variance is dominated by the fact that the expected value of the sample variance for each block of samples increases with time. The agreement which we found between standard statistical theory and our experimental results on the $1/N_b$ rate of decrease of variance is undoubtedly due to the band-limited nature of the experimental data. We will attempt to verify these conclusions in the future using Monte Carlo techniques.

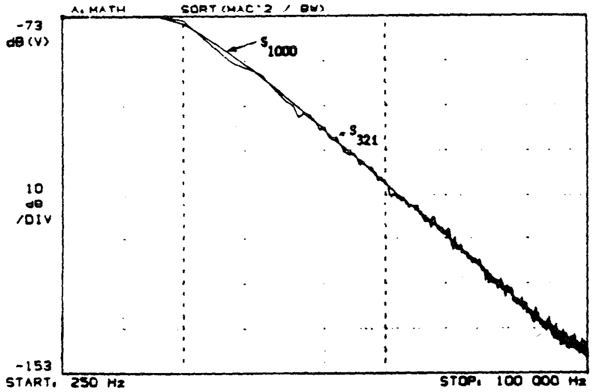


Figure 4. Comparison of the spectral estimate of f^{-4} power-law noise with 1000 samples with that obtained with 32 samples. The text explains how these two curves are used to obtain the fractional RMS confidence of the spectral estimate for 32 samples.

IV.B. Experimental Determination

The following procedure can be used to experimentally determine the variance of the spectral estimates of virtually any type of noise spectrum with any type of window for a particular instrument. Since the spectral density of interest is in general nonwhite, we must determine both the "true value" and a way to normalize the fractional error of the estimate as a function of the number of samples. This can be done by making use of the above theoretical analysis that shows that the variance should decrease as the square root of the number of samples since they are approximately statistically independent (in fact, exactly so in the cases of white and random-walk noise). As an example we have chosen to take $N_b = 1000$ blocks of the various power-law noise types examined in III B above and compare the value of the spectral estimate with that obtained from $N_b = 32$ blocks (see Figure 4). Since the variance of the 1000 block data is about 32 times smaller than that of the 32 block data, it can serve as an accurate estimate of the "true value." Let $\hat{S}_{1000}(f_j)$ represent this quantity at the j-th channel (frequency). By subtracting the 1000 block data from the 32 block data at the j-th channel, we then have one estimate of the error for the 32 block data; by repeating this procedure over N_c different channels and N_r different replications, we can obtain accurate estimates of the variance for the 32 block data. Let $\hat{S}_{32i}(f_j)$ represent the spectral estimate for the 32 block data at the j-th channel and the i-th replication. To compensate for the variation in the level of the spectral estimates with channel, it is necessary to divide the error at the j-th channel by the "true value" $\hat{S}_{1000}(f_j)$. The mean square fractional error of the 32 block data for the noise type under study is given by

$$\sigma_{32}^2 = \frac{1}{N_r N_c} \sum_{i=1}^{N_r} \sum_j \left(\frac{\hat{S}_{32i}(f_j) - \hat{S}_{1000}(f_j)}{\hat{S}_{1000}(f_j)} \right)^2$$
$$\approx \frac{var\{\hat{S}_{32i}(f_j)\}}{S^2(f_j)}.$$

It is assumed that all channels with bias — as indicated in Table 1 — have been excluded in the summation over j. It is also important that the changes in the spectral density not exceed the dynamic range of the digitizer because under this condition the quantization errors — in addition to causing biases in the spectral estimates as discussed earlier — can lead to situations where the variance does not improve as N_b increases. These values can be scaled to any number of blocks N_b if care is taken to avoid these quantization errors. Upper and lower approximate 67% confidence limits for $S(f_j)$ — the true spectral density at channel j — using Hanning, uniform and the proprietary "flattened peak" windows for N_b approximately independent blocks are given by

$$\hat{S}(f_j)(1 \pm V(\alpha, N_b))$$

where $\hat{S}(f_j)$ is the spectral estimate given by Equation (1) and $V(\alpha, N_b)$ is the fractional variance given in Table 2 for f^α and α = 0, -2, -3 and -4 (these results were obtained by averaging over $N_r N_c = 1200$ channels). The variances obtained are very close to those obtained from standard statistical analysis for white noise, i.e.,

$$\hat{S}(f_j)\left(1 \pm \frac{1}{\sqrt{N_b}}\right).$$

Table 2. Confidence Intervals for FFT Spectral Estimates

power law noise type	uniform	window Hanning	flattened peak
f^0	$1.02/\sqrt{N_b}$	$0.98/\sqrt{N_b}$	$0.98/\sqrt{N_b}$
f^{-2}	$1.02/\sqrt{N_b}$	$1.04/\sqrt{N_b}$	$1.04/\sqrt{N_b}$
f^{-3}	unusable	$1.04/\sqrt{N_b}$	$1.04/\sqrt{N_b}$
f^{-4}	unusable	$1.04/\sqrt{N_b}$	$1.04/\sqrt{N_b}$

V. Conclusions

We have introduced experimental techniques to evaluate the statistical properties of FFT spectral estimates for common noise types found in oscillators, amplifiers, mixers and similar devices, and we have compared these with theoretical calculations. We have used these techniques to study the biases and variances of FFT spectral estimates using the uniform, Hanning, and a proprietary "flattened peak" window. The theoretical analysis was greatly hampered because the instrument manufacturer does not disclose the exact form of the "flattened peak" window or the normalization procedure for the other windows. Nevertheless, we obtained fair agreement between the theoretical and the experimental analysis. The variances of the spectral estimation were virtually identical to a few percent for f^0 to f^{-4} noise except for the uniform window which is incapable of measuring noise which falls off faster than f^{-2}. There was a very large difference in the biases of the first few channels for the three windows. The Hanning window showed significant biases in the first 3 channels while the proprietary "flattened peak" window showed large biases for f^{-4} noise even up to channel 13. The Hanning window therefore yields useful information over three times wider frequency range than the proprietary "flattened peak" window. In the particular instrument studied, the proprietary "flattened peak" window is the best choice for estimating the height of a narrow band source, while the Hanning window is by far the best choice for spectral analysis of common noise types found in oscillators, amplifiers, mixers, etc. We have also shown that the 67% confidence levels for spectral estimation as a function of the number of contiguous nonoverlapping blocks, N_b, is approximately given by

$$S = S_m\left(1 \pm \frac{0.98}{\sqrt{N_b}}\right)$$

for white noise (f^0) and by

$$S = S_m\left(1 \pm \frac{1.04}{\sqrt{N_b}}\right)$$

for noise types f^{-2} to f^{-4}. This agrees to within 4% of that found by standard statistical analysis for white noise. Using this data one can now determine the number of samples necessary to estimate — to a given level of statistical uncertainty — the spectrum of the various noise types commonly found in oscillators, amplifiers, mixers, etc.

References

[1] D. Babitch and J. Oliverio, "Phase Noise of Various Oscillators at Very Low Fourier Frequencies," in *Proceedings of the 28th Annual Symposium on Frequency Control*, 1974, pp. 150–159.

[2] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Englewood Cliffs, New Jersey: Prentice Hall, 1989, ch. 11, pp. 695–767.

[3] D. B. Percival and F. L. Walls, "Spectral Analysis of Power-Law Processes with a Spectrum Analyzer," manuscript in preparation, 1989.

[4] M. B. Priestley, *Spectral Analysis and Time Series*. London: Academic Press, 1981, ch. 7, pp. 502–612.

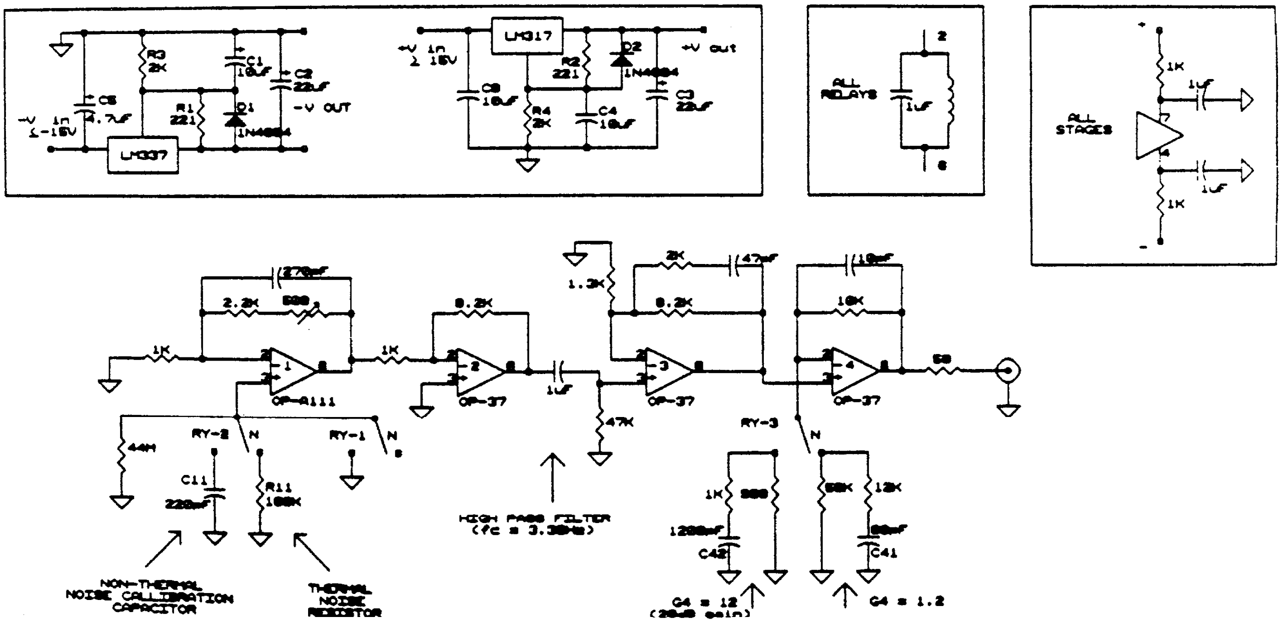


Figure 5. Circuit diagram of a precision noise source.

Appendix. Precision Noise Source

Figure 5 shows the circuit diagram of a precision noise source whose spectral density can be determined from first principles to ±0.2 dB over the frequency range from 20 Hz to 20 kHz. The spectral density is basically given by the Johnson noise of the 10^5 ohm resistor, $V_n^2 = 4kTR$, where T is in Kelvin, and k is Boltzmann's constant. Corrections due to the input noise voltage and noise current of the amplifier amount to about 0.2 dB for the circuit elements shown. All resistors are precision 1% metal film resistors. The output level can be switched from -100 dBV/Hz to -80 dBV/Hz. By adjusting the noise-gain capacitors one can make the noise spectrum flat to within 0.3 dB out to 200 kHz. There is also provision to measure the input noise voltage of the amplifier by shorting the input to ground or the combined noise voltage and noise current by switching a 220 pF capacitor into the input instead of the 10^5 ohm noise resistor.

Proc. 35th Ann. Freq. Control Symposium, USAERADCOM, Ft. Monmouth, NJ 07703, May 1981

A MODIFIED "ALLAN VARIANCE" WITH INCREASED OSCILLATOR CHARACTERIZATION ABILITY

David W. Allan and James A. Barnes

Time and Frequency Division
National Bureau of Standards
Boulder, Colorado 80303

Summary

Heretofore, the "Allan Variance," $\sigma_y^2(\tau)$, has become the de facto standard for measuring oscillator instability in the time-domain. Often oscillator frequency instabilities are resonably modelable with a power law spectrum: $S_y(f) \sim f^{\alpha}$, where y is the normalized frequency, f is the Fourier frequency, and α is a constant over some range of Fourier frequencies. It has been shown that for power law spectrum $\sigma_y^2(\tau) \sim \tau^{\mu}$, and that $\mu = -\alpha-1$ for $-3<\alpha\leq +1$, where τ is the nomimal sample time over which each value of y is measured. The modified "Allan Variance" developed in this paper yields $\mu \cong -\alpha-1$ for all α in the range $-3<\alpha$, which removes the previous ambiguity: $\mu = -2$ for $+1\leq\alpha$. In other words, with the modified "Allan Variance" one can easily distinguish between white phase noise ($\alpha = +2$) and flicker phase noise ($\alpha = +1$) -- commonly occurring for the short term instabilities of quartz crystal oscillators and active hydrogen masers.

Key Words. Flicker Noise; Frequency Stability; Oscillator Noise Modeling; Power Law Spectrum; Time-Domain Stability; White Noise.

Introduction

The random fluctuations in precision oscillators may often be characterized by a power law spectrum:

$$S_y(f) = h_{\alpha} f^{\alpha} , \tag{1}$$

where y is the normalized frequency deviation, f is the Fourier frequency, h_{α} is the intensity of the particular noise process, and α is constant over some range of f. The typical values of α are: +2 (white noise phase modulation, PM); +1 (flicker noise PM); 0 (white noise frequency modulation, FM); -1 (flicker noise FM); and -2 (random walk FM). The Allan variance, as it has come to be known,[1] has been demonstrated as a very useful statistical tool for characterizing these various random processes with the exception that if $\alpha = +1$ or $+2$, the dependence on τ is nominally the same, i.e., $\sim \tau^{-2}$. It is not at all uncommon for white PM and flicker PM to occur in precision oscillators for τ of the order of one second and shorter. The modified Allan variance, as developed in this paper, depends as τ^{-2} for $\alpha = +1$ and as τ^{-3} for $\alpha = +2$. This yields a clear distinction in the time domain between these heretofore somewhat ambiguous processes.

Definition of "Allan Variance" and Related Concepts

Define y, the normalized frequency deviation, as

$$y(t) = \frac{\nu(t) - \nu_0}{\nu_0} \tag{2}$$

where $\nu(t)$ is the output frequency of the oscillator being studied, and ν_0 is nominally the same frequency, but of a reference oscillator assumed for the moment without loss of generality to be better than the test oscillator. The time deviation from some arbitrary origin (t = 0) is the integral of the frequency deviations (from that origin):

$$x(t) = \int_0^t y(t') \cdot dt' \tag{3}$$

The $i^{\underline{th}}$ average frequency deviation over an interval, τ, is

$$\bar{y}_i = \frac{x_{i+1} - x_i}{\tau} \tag{4}$$

where the assumption is made that the time deviation measurements are nominally spaced τ apart.

The "Allan Variance" is defined as:

$$\sigma_y^2(\tau) = \tfrac{1}{2} \langle (\bar{y}_{i+1} - \bar{y}_i)^2 \rangle \ , \tag{5}$$

where the brackets "< >" denote infinite time average. Using equation (4), one may write:

$$\sigma_y^2(\tau) = \frac{1}{2\tau^2} \langle (x_{i+2} - 2x_{i+1} + x_i)^2 \rangle \ . \tag{6}$$

It has been shown that typically $\sigma_y^2(\tau)$ varies as τ^{μ}, and that $\mu = -\alpha-1$ for $-3 \leq \alpha \lesssim +1$.[1,2] Hence, we see one of the dimensions of usefulness of $\sigma_y^2(\tau)$; i.e., ascertaining the dependence on τ allows an estimate of α (the power law spectral type of noise). However, if $\alpha \geq +1$, then $\mu \cong -2$, and the τ dependence becomes somewhat ambiguous as to the type of noise in this region. It is interesting to note that in the region $\alpha \geq +1$, $\sigma_y^2(\tau)$ is bandwidth (f_h) dependent; i.e., the bandwidth of the measurement system will affect the value of $\sigma_y(\tau)$, and furthermore, one may use the bandwidth dependence[3] to determine the value of α (see also Appendix Ref. 2).

Development of the Modified Allan Variance

One may also write $\sigma_y^2(\tau)$ in terms of a generalized autocovariance function:

$$\sigma_y^2(\tau) = \frac{1}{2\tau^2} [4U_x(\tau) - U_x(2\tau)] \ , \tag{7}$$

where

$$U_x(\tau) = 2[R_x(0) - R_x(\tau)], \tag{8}$$

and where

$$R_x(\tau) = \langle x(t+\tau) \cdot x(t) \rangle \ , \tag{9}$$

the classical autocovariance function of $x(t)$. Using the Fourier transforms of generalized functions, one may determine the coefficients relating the power spectral density to $\sigma_y^2(\tau)$. Ref. 1 gives these relationships. It is of interest to note that $U_x(\tau)$ has the following approximate form in the region $\alpha \geq +1$ (see Appendix Ref. 2):

$$U_x(\tau) \sim a(\alpha) \left[\left| \frac{1}{2\pi f_h} \right|^{-\alpha+1} - |\tau|^{-\alpha+1} \right] \tag{10}$$

Hence, one notes that by changing the reciprocal bandwidth as well as τ, one affects $\sigma_y^2(\tau)$ in similar ways, depending on the value of α. From this, one should be able to deduce the value of α, since the bandwith dependence becomes stronger for α moving positive from +1, and the τ dependence becomes stronger as α moves negative from +1. One can change the bandwidth in the hardware or in the software. In the past, it has typically been done in the hardware.[3] James Snyder[4] has shown that it is relatively easy to change the bandwidth in the data processing by a clever technique and we have followed his lead. In particular, we have chosen a new variance analysis scheme which coincides with the Allan variance at the minimum sample time, τ_0, (i.e., minimum data spacing), but which changes the bandwidth in the software as the sample time, τ, is changed.

Each reading of the time deviation, x_i, has associated with it an intrinsic nominal (hardware) measurement system bandwidth, f_h. Define $\tau_h = \frac{1}{2\pi f_h}$; and similarly we may define a software bandwidth, $f_s = f_h/n$, which is 1/n times narrower than the hardware bandwidth. This software bandwidth can be realized by averaging n adjacent x_i's; $\tau_s = n\tau_h$, where $\tau_s = 1/f_s$. We have defined a modified Allan variance which allows the reciprocal software bandwidth to change linearly with the sample time, τ:

$$\text{Mod } \sigma_y^2(\tau) = \frac{1}{2\tau^2} \left\langle \left[\frac{1}{n} \sum_{i=1}^{n} (x_{i+2n} - 2x_{i+n} + x_i) \right]^2 \right\rangle \tag{11}$$

where $\tau = n\tau_0$. Eq. 11 clearly coincides with Eq. 6 for n = 1. One can see that, in general, we have formed a second difference of three time readings with each of the three being an average of n of the x_i's (with non-overlapping averages). As n increases, the (software) bandwidth decreases and this bandwidth varies just as $f_s = f_h/n$.

For a finite data set of N readings of x_i (i = 1 to N), we may write an estimate:

$$\text{Mod}\ \sigma_y^2(\tau) = \frac{1}{2\tau^2 n^2 (N-3n+1)} \cdot \sum_{j=1}^{N-3n+1} \left[\sum_{i=j}^{n+j-1} (x_{i+2n} - 2x_{i+n} + x_i) \right]^2 \tag{12}$$

Eq. 12 is easy to program, but takes more time to compute than for $\sigma_y(\tau)$. This is only of significance for the smaller computer or handheld calculator.

Comparisons, Tests, and Examples of Usage of the Modified Allan Variance

We simulated various power law noise processes, and applied Eq. 12. Shown in Fig. 1 are the resulting τ-dependences of the modified Allan variances for $\alpha = -2, -1, 0, +1,$ and $+2$. The solid lines drawn are the anticipated or theoretical slopes for the particular noise process. One sees excellent agreement for white noise PM and for flicker noise PM, and nominal agreement for the others.

One can express Eq. 11 in terms of the generalized autocovariance function:

$$\begin{aligned}\text{Mod}\ \sigma_y^2(\tau) = \frac{1}{2\tau^2 n^2} \Big\{ & [4U_x(\tau) - U_x(2\tau)] \cdot n \\ & + \sum_{i-1}^{n-1} (n-i)[-6U_x(i\tau_0) + 4U_x((n+i)\tau_0) \\ & + 4U_x((n-i)\tau_0) - U_x((2n+i)\tau_0) \\ & - U_x((2n-i)\tau_0)] \Big\} \ .\end{aligned} \tag{13}$$

In the range $-3 \leq \alpha \leq +1$, one may write:

$$U_x(\tau) = a(\alpha) \cdot \tau^{-\alpha+1} \ , \tag{14}$$

which when substituted in Eq. 13, and using Eq. 7, yields

$$\begin{aligned}\text{Mod}\ \sigma_y^2(\tau) = \sigma_y^2(\tau) \Big\{ & \frac{1}{n} + \frac{1}{n^2\ 4n^{-\alpha+1} - (2n)^{-\alpha+1}} \cdot \\ & \sum_{i=1}^{n-1} (n-i) \cdot \Big[-6i^{-\alpha+1} + 4(n+i)^{-\alpha+1} \\ & -(2n+i)^{-\alpha+1} + 4(n-i)^{-\alpha+1} - (2n-i)^{-\alpha+1} \Big] \Big\}\end{aligned} \tag{15}$$

Since we know that $\sigma_y^2(\tau)$ is well behaved in this range and $\mu = -\alpha - 1$, it is of interest to look at the ratio:

$$R(n) = \text{Mod}\ \sigma_y^2(\tau) / \sigma_y^2(\tau) \ .$$

As stated before, at $n = 1$ ($\tau = \tau_0$) the ratio is unity. One can evaluate Eq. 15 with a computer. A reasonable empirical fit may be formed, which is good to 0.5% or better of Eq. 15:

$$R(n) = \frac{q + pn^E - p}{qn^E} \tag{16}$$

which approaches p/q asymptotically as n approaches infinity, and is within 1% of p/q for $n \geq 8$. Listed in Table 1 are the empirical values of p, q, and E and the quality of fit for the appropriate power law noise processes.

TABLE 1

Noise Type						
Name	α	p	q	E	fit	Mod $\sigma_y(\tau)/\sigma_y(\tau)$ $\tau >> \tau_0$
White FM	0	1	2	2	perfect	.707
Flicker FM	-1	99.9	148	2.35	½%	.821
Random Walk FM	-2	33	40	2.35	<½%	.908
Flicker Walk FM	-3	1	1	--	perfect	1

The results of Table 1 are in reasonable agreement with simulated results of Fig. 1(a) through 1(e). The last row in Table 1, "flicker walk" frequency modulation, is out of the range of applicability of α, but the ratio, R(n), is still convergent.

The $U_x(\tau)$ function for flicker noise PM is extremely complicated and has not been developed, but one can arrive at an empirical value for it. The $U_x(\tau)$ function is derivable for the other power law spectral processes. Table 2 gives the relationships between the time domain measure Mod $\sigma_y^2(\tau)$ and its power law spectral counterpart, given in Eq. 1. Also listed in the right hand column of Table 2 are the asymptotic values of $R(n)$:

TABLE 2

Noise Type		Mod $\sigma_y^2(\tau)$	Comment	$R(n)$ $n \to \infty$
Name	α			
White PM	+2	$h_2 \cdot \frac{3 f_h}{(2\pi)^2 n \tau^2}$	Exact	1
Flicker PM	+1	$h_1 \cdot \frac{1.038 + 3\ell n(\omega_h \tau)}{(2\pi)^2 \tau^2}$	Empirical	1
White FM	0	$h_0 \cdot \frac{R(n)}{2\tau}$	Exact	0.5
Flicker FM	-1	$h_{-1}\, 2\ell n(2) \cdot R(n)$	Empirical; Exact Available	0.674
Random Walk FM	-2	$h_{-2} \cdot \frac{(2\pi)^2 \tau}{6} \cdot R(n)$	Empirical; Exact Available	0.824

It is clear from Table 2 that Mod $\sigma_y^2(\tau)$ is very useful for white PM and flicker noise PM, but for $\alpha < +1$ the conventional Allan variance, $\sigma_y^2(\tau)$, gives both an easier-to-interpret and an easier-to-calculate measure of stability.

It is interesting to make a graph of α versus μ for both the ordinary Allan variance and the modified Allan variance. Shown in Fig. 2 is such a graph. This graph allows one to determine power law spectra for non-interger as well as interger values of α. The dashed line for the modified Allan variance has been intentionally moved to the left in Fig. 2 because for small values of n the value of μ will appear to be slightly more negative that for $\sigma_y^2(\tau)$, even though for large n, they both approach the same slope (i.e., the same values of μ). In fact, in the asymtotic limit, the equation relating μ and α for the modified Allan variance is

$$\alpha = -\mu - 1, \text{ for } -3 < \alpha < +3 . \qquad (17)$$

The value of $\mu = -4$ for $\alpha = +3$ was verified empirically with simulated data, and it appears that for $\alpha > +3$, μ remains at -4.

A direct application for using the modified Allan variance recently arose in the analysis of atomic clock data as received from a Global Positioning System (GPS) satellite. We were interested in knowing the short-term characteristics of the newly developed, high-accuracy NBS/GPS receiver, as well as the propagation fluctuations. Fig. 3 shows both $\sigma_y^2(\tau)$ and Mod $\sigma_y^2(\tau)$ for comparison. Using Mod $\sigma_y^2(\tau)$, we can tell that the fundamental limiting noise process involved in the system is white noise PM with the exciting result that averaging for four minutes can allow one to
* ascertain time difference to better than one nanosecond excluding other systematic effects.

Conclusion

We have developed a supplemental measure, the "Modified Allan Variance" (Mod $\sigma_y^2(\tau)$), which has very useful properties when analyzing oscillator or signal stability in the presence of white noise phase modulation or flicker noise phase modulation. It also works reasonably well as a stability measure for other commonly occuring noise processes in precision oscillators.

We would recommend that for most time domain analysis, $\sigma_y^2(\tau)$ should be the first choice. If $\sigma_y^2(\tau)$ depends on τ as τ^{-1}, then the modified Allan variance can be used as a substitute to help remove the ambiguity as to the noise processes.

Acknowledgments

The authors are deeply grateful for stimulating discussions with Dr. James J. Snyder, and for the useful comments of Dr. Robert Kamper and Mr. David A. Howe.

References

1. James A. Barnes, Andrew R. Chi, Leonard S. Cutler, Daniel J. Healey, David B. Leeson, Tomas E. McGunical, James A. Mullen, Jr., Warren L. Smith, Richard L. Sydnor, Robert F. C. Vessot, and Gernot M. R. Winkler, Proc. IEEE Trans. Instrum. Meas. IM-20, 105 (1971); also published as NBS Tech. Note 394 (1971).
2. David W. Allan, Proc. IEEE 54, 221 (1966).
3. R. F. C. Vessot, Proc. IEEE 54, 199 (1966).
4. James J. Snyder, Proc. 35th Annual Symp. on Freq. Control (1981), to be published.

---

* See Appendix Note # 34

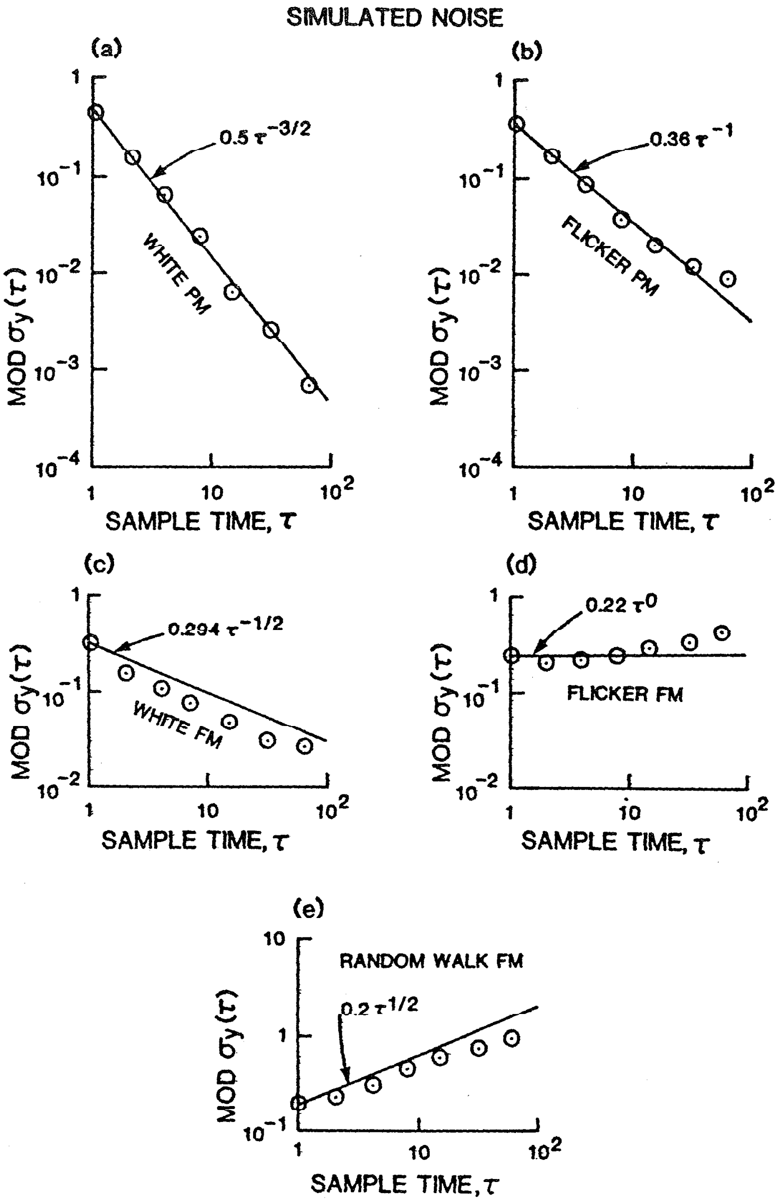


Fig 1a-e. Mod $\sigma_y(\tau)$ using Eq. 12 was calculated for different sample times for independently generated and simulated noise processes, which were white phase noise, flicker phase noise, white frequency noise, flicker frequency noise, and random walk frequency noise, respectively. Mod $\sigma_y(\tau)$ was computed for 399 data points in each case. One sees the excellent fit to the theory for white phase noise and flicker phase noise, an important new contribution in the ability to characterize oscillators having these noise processes.

Characterization of Frequency Stability: Analysis of the Modified Allan Variance and Properties of Its Estimate

PAUL LESAGE AND THÉOPHANE AYI

Abstract—An analytical expression for the modified Allan variance is given for each component of the model usually considered to describe the frequency or phase fluctuations in frequency standards. The relation between the Allan variance and the modified Allan variance is specified and compared with that of a previously published analysis. The uncertainty on the estimate of the modified Allan variance calculated from a finite set of measurement data is discussed.

Manuscript received December 8, 1983; revised February 17, 1984.

The authors are with the Laboratoire de l'Horloge Atomique, Equipe de Recherche du CNRS, associée à l'Université Paris-Sud, 91405 Orsay, France.

I. INTRODUCTION

Many works [1]-[5] have been devoted to the characterization of the frequency stability of ultrastable frequency sources and have shown that the frequency noise of a generator can be easily characterized by means of the "two-sample variance" [2] of frequency fluctuations, which is also known as the "Allan variance" [2] in the special case where the dead time between samples is zero.

An algorithm for frequency measurements has been developed by J. J. Snyder [6], [7]. It increases the resolution of frequency meters, in the presence of white phase noise. It has been considered in detail by D. W. Allan and J. A. Barnes [8]. They have defined a function called the "modified Allan variance" and they have analyzed its properties for the commonly encountered components of phase or frequency fluctuations [3]. For that purpose, the authors of [8] have expressed the modified Allan variance in terms of the autocorrelation of the phase fluctuations. For each noise component, they have computed the modified Allan variance and deduced an empirical expression for the ratio between the modified Allan variance and the Allan variance.

In this paper, we show that the analytical expression of this ratio can be obtained directly, even for the noise components for which the autocorrelation of phase functions is not defined from the mathematical point of view. We give the theoretical expressions and compare them with those published in [8].

The precision of the estimate of the modified Allan variance is discussed and results related to white phase and white frequency noises are presented.

II. BACKGROUND AND DEFINITIONS

In the time domain, the characterization of frequency stability is currently achieved by means of the two-sample variance [2] $\langle\sigma_y^2(2, T, \tau)\rangle$ of fractional frequency fluctuations. It is defined as

$$\langle\sigma_y^2(2, T, \tau)\rangle = \tfrac{1}{2}\langle(\bar{y}_{k+1} - \bar{y}_k)^2\rangle \tag{1}$$

where the quantity $\bar{y}_k$ is the average value of the fractional frequency fluctuations $y(t)$ over the time interval $(t_k, t_k + \tau)$ such that

$$\bar{y}_k = \frac{1}{\tau}\int_{t_k}^{t_k+\tau} y(t)\,dt. \tag{2}$$

In (2), t_k represents the moment at which the kth observation time interval starts. We have

$$t_k = t_0 + kT, \qquad T \geqslant \tau \tag{3}$$

where t_0 is an arbitrary time origin, k is a positive integer, and T is the time interval between the beginning of two successive observations.

In all the following, we assume that the dead time between samples is zero. We then have

$$T = \tau. \tag{4}$$

In this special case, the two-sample variance is well known as the Allan variance $\sigma_y^2(\tau)$

$$\sigma_y^2(\tau) = \langle\sigma_y^2(2, \tau, \tau)\rangle. \tag{5}$$

The relation between the Allan variance and $y(t)$ can be expressed as

$$\sigma_y^2(\tau) = \frac{1}{2\tau^2}\left\langle\left\{\int_{t_k+\tau}^{t_k+2\tau} y(t)\,dt - \int_{t_k}^{t_k+\tau} y(t)\,dt\right\}^2\right\rangle. \tag{6}$$

Equation (6) shows that $\sigma_y^2(\tau)$ is proportional to the true variance of the output of a linear filter with input signal $y(t)$ and impulse response $h_1(t)$ in Fig. 1.

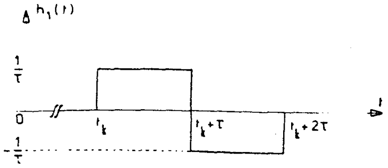


Fig. 1. Variations with time of the linear filter impulse response which represents the signal processing for the Allan variance calculation.

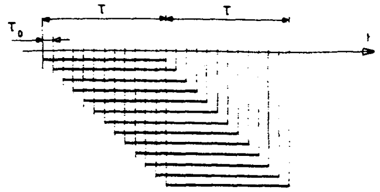


Fig. 2. Illustration of the algorithm considered for the measurement of periodic signal frequency.

The fractional frequency fluctuations $y(t)$ are actually well described by a conventional model which consists of a set of five independent noise processes [2]. Taking into account the finite bandwidth of the processed signal and assuming a single pole filter, the one-sided power spectral density $S_y(f)$ of $y(t)$ can be written as

$$S_y(f) = h_\alpha \frac{f^\alpha}{1 + \left(\dfrac{f}{f_c}\right)^2} \tag{7}$$

where coefficients h_α do not depend on f. The integer α equals 2, 1, 0, -1, and -2. f_c is the 3-dB bandwidth of the hardware filter.

III. THE MODIFIED ALLAN VARIANCE

The main property of the algorithm developed by J. J. Snyder is to increase the precision on the measure of periodic signal frequency, in presence of white phase noise [7]. It consists in dividing a time interval τ into n cycles of clock period τ_0 such as

$$\tau = n\tau_0. \tag{8}$$

Therefore, from a given observation time interval of duration 2τ, n overlapping time intervals of duration τ can be obtained, as depicted in Fig. 2. Another property of this algorithm is to reduce the total observation time by a factor $n/2$.

Following this way, Allan and Barnes have introduced the "modified Allan variance" [8] such as

$$\text{Mod}\,\sigma_y^2(\tau) = \frac{1}{2\tau^2}\left\langle\left[\frac{1}{n}\sum_{i=1}^{n}\left\{\int_{t_0+(i+n)\tau_0}^{t_0+(i+2n)\tau_0} y(t)\,dt - \int_{t_0+i\tau_0}^{t_0+(i+n)\tau_0} y(t)\,dt\right\}\right]^2\right\rangle. \tag{9}$$

It can be easily seen from (9) that the calculation of each statistical sample involved in the definition of $\text{Mod}\,\sigma_y^2(\tau)$ requires a signal observation of duration 3τ.

The impulse response $h_n(t)$ of the equivalent linear filter consists in finite sum of n shifted impulse responses $h_1(t)$. We have

$$h_n(t) = \frac{1}{n}\sum_{i=1}^{n} h_1(t - i\tau_0). \tag{10}$$

TABLE I
ANALYTICAL EXPRESSION FOR THE MODIFIED ALLAN VARIANCE WITHIN CONDITION $2\pi f_c \tau_0 \gg 1$

NOISE TYPE	α	$\text{Mod}\ \sigma_y^2(\tau)$
WHITE P M	2	$\dfrac{3 h_2 f_c}{8 n\pi \tau^2}$
FLICKER P M	1	$\dfrac{h_1}{4 \pi^2 n^2 \tau^2}\left[3n \ln(2\pi f_c \tau) + \sum_{k=1}^{n-1} (n-k)\left\{4 \ln\left(\dfrac{n^2}{k^2} - 1\right) - \ln\left(\dfrac{4n^2}{k^2} - 1\right)\right\}\right]$
WHITE F M	0	$\dfrac{h_0}{2\tau} \times \dfrac{n^2+1}{2n^2}$
FLICKER F M	- 1	$\dfrac{2h_{-1}\ln 2}{n^2}\left[\dfrac{4n^2-3n+1}{2} + \dfrac{1}{n^2 \ln 2} \times \sum_{k=1}^{n-1} (n-k) \times \left\{\dfrac{n}{2}\left[(k+2n)\ln(k+2n) - (k-2n)\ln(2n-k)\right] + \dfrac{1}{2}(k+n)(k-2n)\ln(k+n) + \dfrac{1}{2}(k-n)(k+2n)\ln\lvert k-n\rvert + 3k^2 \ln k - k\left[(n+2k)\ln\left(k+\dfrac{n}{2}\right) - (n-2k)\ln\left\lvert k-\dfrac{n}{2}\right\rvert\right]\right\}\right]$
RANDOM WALK F M	- 2	$\dfrac{33}{40} + \dfrac{1}{8n^2} + \dfrac{1}{20n^4}$

*

In order to illustrate (10), variations with time of the shifted functions $h_1(t - i\tau_0)$ and of the impulse response $h_n(t)$ are represented in Fig. 3(a) and (b), respectively, for $n = 10$.

For $n = 1$, the Allan variance and the modified one are equal. We have

$$\text{Mod}\ \sigma_y^2(\tau) = \sigma_y^2(\tau). \tag{11}$$

One can express (9) in terms of the spectral density $S_y(f)$. We have

$$\text{Mod}\ \sigma_y^2(\tau) = \frac{2}{n^2\pi^2\tau^2} \cdot \left\{ n \int_0^\infty \frac{1}{f^2} S_y(f) \sin^4(\pi f n \tau_0)\, df + 2 \sum_{k=1}^{n-1} (n-k) \int_0^\infty \frac{1}{f^2} S_y(f) \cos(2\pi f k \tau_0) \cdot \sin^4(\pi f n \tau_0)\, df \right\}. \tag{12}$$

It should be noted that the integrals involved in (12) are convergent for each noise component. The analytical expression for the modified Allan variance can therefore be deduced directly from this equation.

In the following, it is assumed that the condition $2\pi f_c \tau_0 \gg 1$ is fulfilled. This means that the hardware bandwidth of the measurement system must be much larger than the reference clock frequency.

We have calculated the modified Allan variance for each noise component. Results are reported in Table I. It appears that the analytical expression for $\text{Mod}\ \sigma_y^2(\tau)$ is relatively simple except for flicker phase and flicker frequency noises where it is given as a finite sum of functions depending on n. In order to compare the Allan variance with the modified one, we

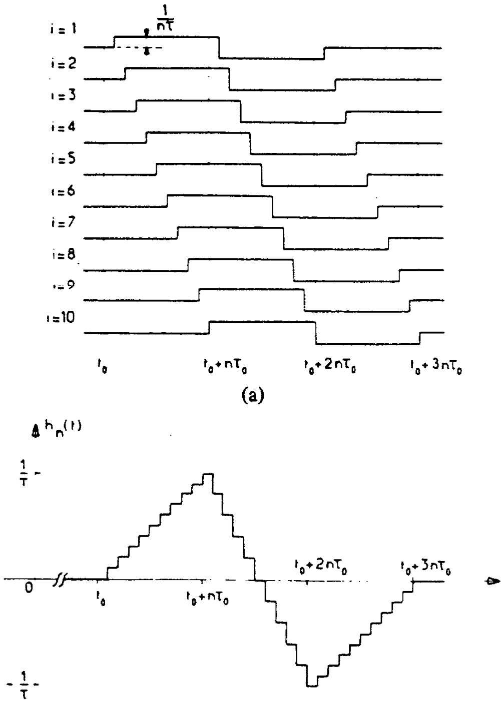


Fig. 3. (a) The impulse response $h_n(t)$, associated with the modified Allan variance calculation, represented as a sum of n shifted impulse response $h_1(t)$. It is assumed $n = 10$. (b) Variations with time of the impulse response $h_n(t)$, in the special case where $n = 10$.

---

* See Appendix Note # 35

TABLE II
ANALYTICAL EXPRESSIONS AND ASYMPTOTICAL VALUS FOR $R(n)$
(Results Are Valid within Condition $2\pi f_c\tau_0 >> 1$)

α	$R(n)$	$\lim_{n\to\infty} R(n)$
2	$\frac{1}{n}$	0
1	$\frac{1}{n^2}\left[n + \frac{1}{3\,\ell n(2\pi f_c n\tau_0)} \times \sum_{k=1}^{n-1}(n-k)\left\{4\,\ell n\left(\frac{n^2}{k^2}-1\right) - \ell n\left(\frac{4n^2}{k^2}-1\right)\right\}\right]$	0
0	$\frac{n^2+1}{2n^2}$	0.5
-1	$\frac{1}{n^2}\left[\frac{4n^2-3n+1}{2} + \frac{1}{n^2\ell n 2} \times \sum_{k=1}^{n-1}(n-k) \times \left\{\frac{n}{2}\left[(k+2n)\ell n(k+2n)-(k-2n)\ell n(2n-k)\right] + \frac{1}{2}(k+n)(k-2n)\ell n(k+n) + \frac{1}{2}(k-n)(k+2n)\ell n\lvert k-n\rvert + 3k^2\,\ell n k - k\left[(n+2k)\ell n\left(k+\frac{n}{2}\right) - (n-2k)\ell n\left\lvert k-\frac{n}{2}\right\rvert\right]\right\}\right]$	0.787
-2	$\frac{33}{40}+\frac{1}{8n^2}+\frac{1}{20n^4}$	0.825

*

consider the ratio $R(n)$ defined in [8] as

$$R(n) = \text{Mod}\,\sigma_y^2(\tau)/\sigma_y^2(\tau). \tag{13}$$

The analytical expressions for $R(n)$, deduced from Table I, are reported in Table II. One can see that $R(n)$ does not depend on the product $f_c\tau_0$, except for flicker phase noise modulation. The asymptotic values of $R(n)$ are also listed in Table II.

Fig. 4 depicts the variations of $R(n)$ with n. It shows that, for large values of n, white phase and flicker phase noise modulations have different dependences. As outlined in [8], this gives a means to easily distinguish these two noise processes, in the time domain. For large n, and for $\alpha = 0, -1, -2$, $R(n)$ remains a constant. Consequently the Allan variance can be deduced from the modified one, for these noise processes.

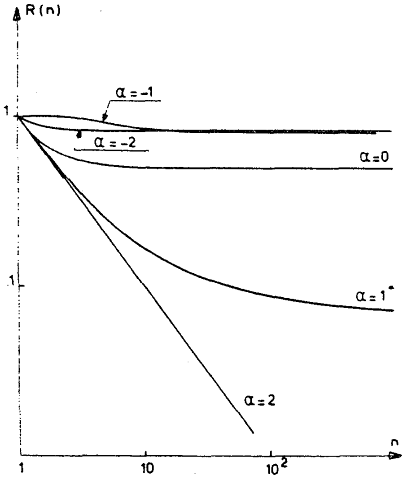
A comparison with results of [8] shows a good agreement for $\alpha = 2$, 0 and -2. But, for $\alpha = 1$ and -1, our expressions for the modified Allan variance and ratio $R(n)$ disagree, especially for flicker phase noise modulation. This discrepancy might be due to the fact that in [8], Mod $\sigma_y^2(\tau)$ is expressed in terms of the autocorrelation function of phase fluctuations which is not defined for $\alpha = 1$.

IV. UNCERTAINTY ON THE ESTIMATE OF THE MODIFIED ALLAN VARIANCE

Equation (9) shows that the definition of the modified Allan variance theoretically implies an infinite set of time intervals. Practically, one can only estimate this quantity from a finite set of m successive cycles similar to the one depicted in Fig. 3(b).

Let Mod $\hat{\sigma}_y^2(\tau)$ be the estimated modified Allan variance (EMAV) such as

$$\text{Mod}\,\hat{\sigma}_y^2(\tau) = \frac{1}{2\tau^2 n^2} \times \frac{1}{m}\sum_{k=1}^{m}\left\{\sum_{i=1}^{n} A_{i,k}\right\}^2 \tag{14}$$



*

Fig. 4. Variations with n of the ratio $R(n)$, for fractional frequency fluctuations with power law spectrum $S_y(f) = h_\alpha \cdot (1/1 + (f/f_c)^2)f^\alpha$, within condition $2\pi f_c\tau_0 >> 1$. (*For $\alpha = +1$, $R(n)$ is a function of f_c and τ_0. The reported variations are for $2\pi f_c\tau_0 = 10^4$.)

where

$$A_{i,k} = \int_{t_0+(i+n)\tau_0+(k-1)3n\tau_0}^{t_0+(i+2n)\tau_0+(k-1)3n\tau_0} y(t)\,dt - \int_{t_0+i\tau_0+(k-1)3n\tau_0}^{t_0+(i+n)\tau_0+(k-1)3n\tau_0} y(t)\,dt. \tag{15}$$

* See Appendix Note # 35

The EMAV is a random function of m. Its calculation requires an observation time of duration $3m\tau$.

We consider ϵ, the fractional deviation of the EMAV relative to the modified Allan variance defined as follows:

$$\epsilon = \frac{\text{Mod } \hat{\sigma}_y^2(\tau) - \text{Mod } \sigma_y^2(\tau)}{\text{Mod } \sigma_y^2(\tau)}. \tag{16}$$

The standard deviation $\sigma(\epsilon)$ of ϵ defines the relative uncertainty on the measurement of the modified Allan variance, due to the finite number of averaging cycles. We have

$$\sigma(\epsilon) = \frac{1}{\text{Mod } \sigma_y^2(\tau)} \{\sigma^2 [\text{Mod } \hat{\sigma}_y^2(\tau)]\}^{1/2} \tag{17}$$

where $\sigma^2 [\text{Mod } \hat{\sigma}_y^2(\tau)]$ denotes the true variance of the EMAV such as

$$\sigma^2 [\text{Mod } \hat{\sigma}_y^2(\tau)] = \langle [\text{Mod } \hat{\sigma}_y^2(\tau)]^2 \rangle - [\text{Mod } \sigma_y^2(\tau)]^2. \tag{18}$$

We assume that the fluctuations $y(t)$ are normally distributed [10]. One can therefore express $\langle [\text{Mod } \hat{\sigma}_y^2(\tau)]^2 \rangle$ as

$$\begin{aligned} m^2 \langle [\text{Mod } \hat{\sigma}_y^2(\tau)]^2 \rangle &= (m^2 + 2m) [\text{Mod } \sigma_y^2(\tau)]^2 \\ &\quad + 4 \sum_{p=1}^{m-1} (m-p) \left\{ 2 \sum_{i=1}^{n-1} (n-i) I_n + n I_0 \right\}^2 \end{aligned} \tag{19}$$

where I_n are integrals which depend on n and on the noise process. We have

$$\begin{aligned} 8\pi^2\tau^2 n^2 I_n = \int_0^\infty \frac{S_y(f)}{f^2} \cos 6\pi n p f \tau_0 \\ \times \{6 \cos 2\pi f \tau_0 i - 4 \cos 2\pi f \tau_0 (i+n) \\ - 4 \cos 2\pi f \tau_0 (i-n) + \cos 2\pi f \tau_0 (i+2n) \\ + \cos 2\pi f \tau_0 (i-2n)\} \, df. \end{aligned} \tag{20}$$

For each noise component, the expression for $\sigma(\epsilon)$ can be deduced from the calculation of integrals involved in (20). These expressions are generally lengthy and complicated except for white phase and white frequency noise modulations, where integrals I_n equal zero. We have limited the present analysis to these two noise components. We get for $\sigma(\epsilon)$

$$\sigma(\epsilon) = \frac{2}{m}, \quad \text{for } \alpha = 2 \text{ and } 0. \tag{21}$$

We now compare (21) with previously published results related to the estimate of the Allan variance [5]. For a given time observation of duration $3m\tau$, it can be easily deduced from [5] that the relative uncertainty on the estimate of the Allan variance varies asymptotically as $1.14\, m^{-1/2}$ and $1.0\, m^{-1/2}$ for $\alpha = 2$ and 0, respectively. For these two noise components, the uncertainty on the EMAV is larger than the uncertainty on the estimated Allan variance, but of the same order of magnitude.

V. CONCLUSION

We have calculated the analytical expression for the modified Allan variance for each component of the model usually considered to characterize random frequency fluctuations in precision oscillators. These expressions have been compared with previously published results and the link between the Allan variance and the modified Allan variance has been specified.

The uncertainty on the estimate of the modified Allan variance has been studied and numerical values have been reported for white phase and white frequency noise modulations.

In conclusion, the modified Allan variance appears to be well suited for removing the ambiguity between white and flicker phase noise modulation. Nevertheless, the calculation of the modified Allan variance requires signal processing which is complicated, compared to the Allan variance. In the presence of white or flicker phase noise, the Allan variance cannot be easily deduced from the modified Allan variance. Furthermore, for a given source exhibiting different noise components, the determination of the Allan variance from the modified one is difficult to perform. For most of time-domain measurements, the use of the Allan variance is preferred.

ACKNOWLEDGMENT

The authors would like to express their thanks to Dr. Claude Audoin for constructive discussions and valuable comments on the manuscript.

REFERENCES

[1] D. W. Allan, "Statistics of atomic frequency standards," *Proc. IEEE*, vol. 54, pp. 221-230, Feb. 1966.

[2] J. A. Barnes *et al.*, "Characterization of frequency stability," *IEEE Trans. Instrum. Meas.*, vol. IM-20, pp. 105-120, May 1971.

[3] L. S. Cutler and C. L. Searle, "Some aspects of the theory and measurement of frequency fluctuations in frequency standards," *Proc. IEEE*, vol. 54, pp. 136-154, Feb. 1966.

[4] J. Rutman, "Characterization of phase and frequency instabilities in precision frequency sources: Fifteen years of progress," *Proc. IEEE*, vol. 66, pp. 1048-1075, Sept. 1978.

[5] P. Lesage and C. Audoin, "Effect of dead-time on the estimation of the two-sample variance," *IEEE Trans. Instrum. Meas.*, vol. IM-28, pp. 6-10, Mar. 1979.

[6] J. J. Snyder, "Algorithm for fast digital analysis of interference fringes," *Appl. Opt.*, vol. 19, pp. 1223-1225, Apr. 1980.

[7] —, "An ultra-high resolution frequency meter," in *Proc. 35th Annu. Symp. Frequency Control* (Fort Monmouth, NJ), 1981, pp. 464-469.

[8] D. W. Allan and J. A. Barnes, "A modified Allan variance with increased oscillator characterization ability," in *Proc. 35th Annu. Symp. Frequency Control* (Fort Monmouth, NJ), 1981, pp. 470-474.

[9] B. Picinbono, "Processus de diffusion et stationnarité," *C.R. Acad. Sci.*, vol. 271, pp. 661-664, Oct. 1970.

[10] P. Lesage and C. Audoin, "Characterization of frequency stability: Uncertainty due to the finite number of measurements," *IEEE Trans. Instrum. Meas.*, vol. IM-22, pp. 157-161, June 1973.

From: *Proceedings of the 15th Annual PTTI Meeting*, 1983.

THE MEASUREMENT OF LINEAR FREQUENCY DRIFT IN OSCILLATORS

James A. Barnes
Austron, Inc., Austin, Texas

ABSTRACT

A linear drift in frequency is an important element in most stochastic models of oscillator performance. Quartz crystal oscillators often have drifts in excess of a part in ten to the tenth power per day. Even commercial cesium beam devices often show drifts of a few parts in ten to the thirteenth per year. There are many ways to estimate the drift rates from data samples (e.g., regress the phase on a quadratic; regress the frequency on a linear; compute the simple mean of the first difference of frequency; use Kalman filters with a drift term as one element in the state vector; and others). Although most of these estimators are unbiased, they vary in efficiency (i.e., confidence intervals). Further, the estimation of confidence intervals using the standard analysis of variance (typically associated with the specific estimation technique) can give amazingly optimistic results. The source of these problems is not an error in, say, the regressions techniques, but rather the problems arise from correlations within the residuals. That is, the oscillator model is often not consistent with constraints on the analysis technique or, in other words, some specific analysis techniques are often inappropriate for the task at hand.

The appropriateness of a specific analysis technique is critically dependent on the oscillator model and can often be checked with a simple "whiteness" test on the residuals. Following a brief review of linear regression techniques, the paper provides guidelines for appropriate drift estimation for various oscillator models, including estimation of realistic confidence intervals for the drift.

I. INTRODUCTION

Almost all oscillators display a superposition of random and deterministic variations in frequency and phase. The most typical model used is[1]:

$$X(t) = a + b\cdot t + Dr\cdot t^2/2 + \phi(t) \qquad (1) \qquad *$$

where X(t) is the time (phase) error of the oscillator (or clock) relative to some standard; a, b, and Dr are constants for the particular clock; and $\phi(t)$ is the random part. X(t) is a random variable by virtue of its dependence on $\phi(t)$.

---

* See Appendix Note # 36

Even though one cannot predict future values of X(t) exactly, there are often significant autocorrelations within the random parts of the model. These correlations allow forecasts which can significantly reduce clock errors. Errors in each element of the model (Eq. 1) contribute their own uncertainties to the prediction. These time uncertainties depend on the duration of the forecast interval, τ, as shown below in Table 1:

TABLE 1. GROWTH OF TIME ERRORS

MODEL ELEMENT NAME	CLOCK PARAMETER	RMS TIME ERROR
Initial Time Error	a	Constant
Initial Freq Error	b	$\sim \tau$
Frequency Drift	Dr	$\sim \tau^2$
Random Variations	$\phi(t)$	$\sim \tau^{3/2}$*

*The growth of time uncertainties due to the random component, $\phi(t)$, can have various time dependencies. The three-halves power-law shown here is a "worst case" model.[2]

One of the most significant points provided by Table 1 is that eventually, the linear drift term in the model over-powers all other uncertainties for sufficiently long forecast intervals! While one can certainly measure (i.e., estimate) the drift coefficient, Dr, and make corrections, there must always remain some uncertainty in the value used. That is, the effect of a drift correction based on a measurement of Dr, is to reduce (hopefully!) the magnitude of the drift error, but not remove it. Thus, even with drift corrections, the drift term eventually dominates all time uncertainties in the model.

As with any random process, one wants not only the point estimate of a parameter, but one also wants the confidence interval. For example, one might be happy to know that a particular value (e.g., clock time error) can be estimated without bias, he may still want to know how large an error range he should expect. Clearly, an error in the drift estimate (see Eq. 1) leads directly to a time error and hence the drift confidence interval leads directly to a confidence interval for the forecast time.

II. LEAST SQUARES REGRESSION OF PHASE ON A QUADRATIC

A conventional least squares regression of oscillator phase data on a quadratic function reveals a great deal about the general problems. A slight modification of Eq. 1 provides a conventional model used in regression analysis[3]:

$$X(t) = a + b\cdot t + c\cdot t^2 + \phi(t) \qquad (2)$$

where c = Dr/2. In regression analysis, it is customary to use the symbol "Y" as the dependent variable and "X" as the independent variable. This is in conflict with usage in time and frequency where "X" and "Y" (time error and frequency error, respectively) are dependent on a coarse measure of time, t, the independent variable. This paper will follow the time and frequency custom even though this may cause some confusion in the use of regression analysis text books.

The model given by Eq. 2 is complete if the random component, $\phi(t)$, is a white noise (i.e., random, uncorrelated, normally distributed, zero mean, and finite variance).

III. EXAMPLE

One must emphasize here that ALL results regarding parameter error magnitudes and their distributions are totally dependent on the adequacy of the model. A primary source of errors is often autocorrelation of the residuals (contrary to the explicit model assumptions). While simple visual inspection of the residuals is often sufficient to recognize the autocorrelation problem, "whiteness tests" can be more objective and precise.

This section analyzes a set of 94 hourly values of the time difference between two oscillators. Figure 1 is a plot of the time difference (measured in microseconds) between the two oscillators. The general curve of the data along with the general expectation of frequency drift in crystal oscillators leads one to try the quadratic behavior (Model #1; models #2 and #3 discussed below). While it is not common to find white phase noise on oscillators at levels indicated on the plot, that assumption will be made temporarily. The results of the regression are summarized in a conventional Analysis of Variance, Table 2.

TABLE 2. ANALYSIS OF VARIANCE QUADRATIC FIT TO PHASE
(Units: seconds squared)

SOURCE	SUM OF SQUARES	d.f.	MEAN SQUARE
Regression	2.32E-9	3	
Residuals	4.26E-12	91	4.68E-14
Total	2.329E-9	94	2.48E-11

Coefficient of simple determination 0.99713

Parameters:

$\hat{a}$ = 1.10795E-5 (seconds) t-ratio = 161.98

$\hat{b}$ = 1.4034E-10 (sec/sec) t-ratio = 152.02

$\hat{c}$ = -3.7534E-16 (sec/sec^2) t-ratio = -143.51

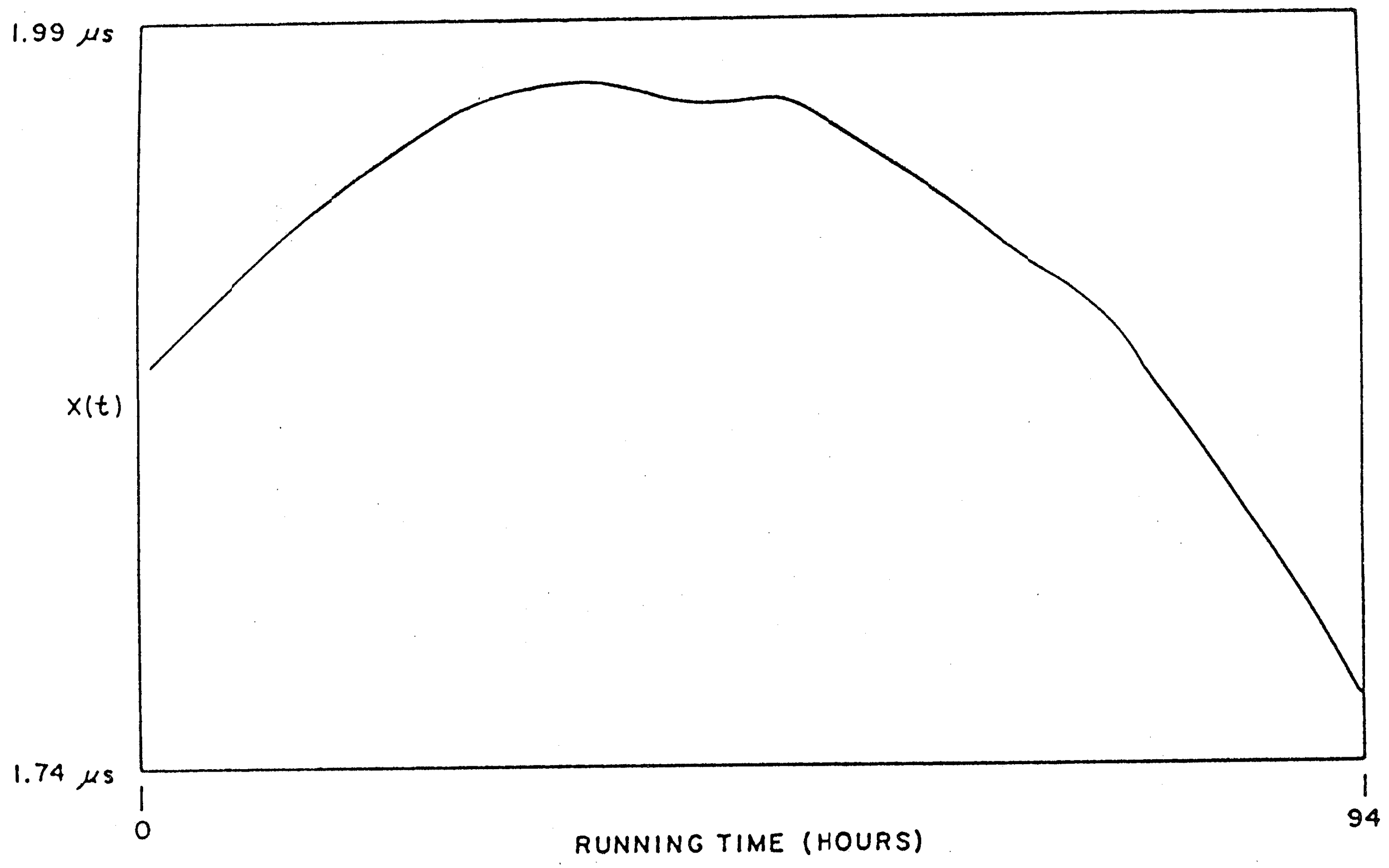


Fig. 1 - Phase difference

$\hat{Dr} = 2\hat{c} = -7.507E-16$ (about -6.5E-11 per day)
Std. Error = 0.05231E-16

(Note: $\hat{a}$ is the value estimated for the a-parameter, etc.)

The Analysis of Variance, Table 2, above suggests an impressive fit of the data to a quadratic function, with 99.71% of the variations in the data "explained" by the regression. The estimated drift coefficient, $\hat{Dr}$, is -7.507E-16 (sec/sec^2) or about -6.5E-11 per day --- 143 times the indicated standard error of the estimate. However, Figure 2, a plot of the residuals, reveals significant autocorrelations even visually and without sensitive tests. (The autocorrelations can be recognized by the essentially smooth variations in the plot. See Fig. 5 as an example of a more nearly white data set.) It is true that the regression reduced the peak-to-peak deviations from about 18 microseconds to less than one microsecond. It is also true that the drift rate is an unbiased estimate of the actual drift rate, but the model assumptions are NOT consistent with the autocorrelation visible in Fig. 2. This means that the confidence intervals for the parameters are not reliable. In fact, the analysis to follow will show just how extremely optimistic these intervals really are.

At this point we can consider at least two other simple analysis schemes which might provide more realistic estimates of the drift rate and its variance. Each of the two analysis schemes has its own implicit model; they are:

(2) Regress the beat frequency on a straight line.
(Model: Linear frequency drift and white FM.)

(3) Remove a simple average from the second difference of the phase.
(Model: Linear frequency drift and random walk FM.)

Continuing with scheme 2, above, the (average) frequency, $\overline{Y}(t)$, is the first difference of the phase data divided by the time interval between successive data points. The regression model is:

$$\overline{Y}(t) = b + Dr \cdot t + \varepsilon(t) \qquad (3) \qquad *$$

where $\varepsilon(t) = [\phi(t + \tau_0) - \phi(t)]/\tau_0$. Following standard regression procedures as before, the results are summarized in another Analysis of Variance Table, Table 3.

TABLE 3. ANALYSIS OF VARIANCE LINEAR FIT TO FREQUENCY
(Units: sec^2/sec^2)

SOURCE	SUM OF SQUARES	d.f.	MEAN SQUARE
Regression	1.879E-18	2	
Residuals	2.084E-20	91	2.29E-22
Total	1.899E-18	93	2.042E-20

(Note: Taking the first differences of the original data set reduces the number of data points from 94 to 93.)

---

* See Appendix Note # 37

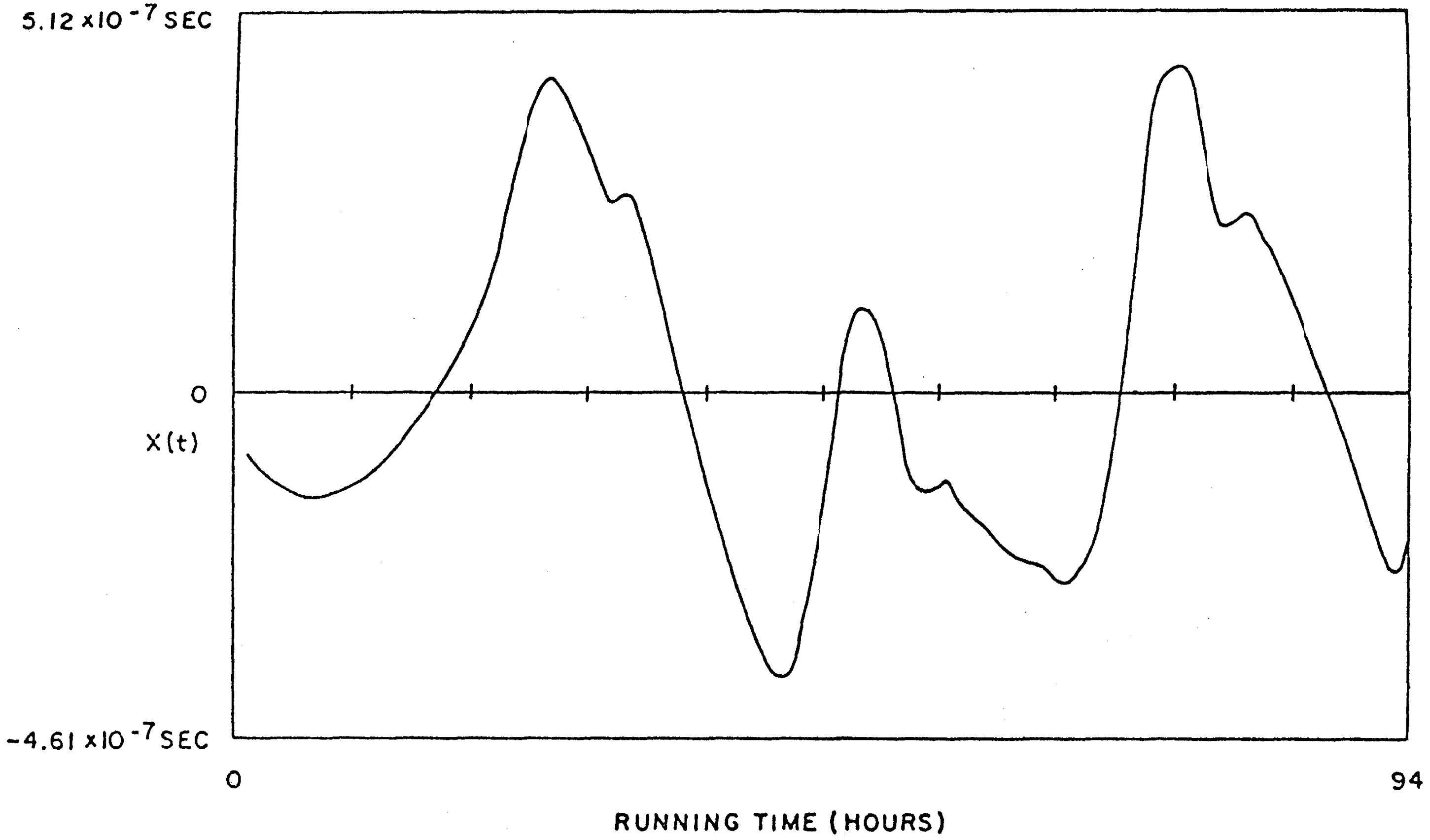


Fig. 2 - Residuals from quadratic fit to phase

Coefficient of simple determination 0.9605

Parameters:

$\hat{b}$ = 1.049E-10 (sec/sec) t-ratio = 3.32

$\hat{D}r$ = -7.635E-16 (sec/sec^2) t-ratio = -4.70
Std. Error = 0.1624E-16 (sec/sec^2)

While the drift rate estimates for the two regressions are comparable in value (-7.507E-16 and -7.635E-16), the standard errors of the drift estimates have gone from 0.052E-16 to 0.162E-16 (a factor of 3). The linear regression's coefficient of simple determination is 96.05% compared to 99.17% for the quadratic fit. Figure 3 shows the residuals from the linear fit and they appear more nearly white. A cumulative periodogram[4] is a more objective test of whiteness, however. The periodogram, Fig. 4, does not find the residuals acceptable at all.

IV. DRIFT AND RANDOM WALK FM

In the absence of noise, the second difference of the phase would be a constant, $Dr\cdot\tau_o^2$. If one assumes that the second difference of the noise part is white, then one has the classic problem of estimating a constant (the drift term), in the presence of white noise (the second difference of the phase noise). Of course, the optimum estimate of the drift term is just the simple mean of the second difference divided by τ_o^2. The results are summarized below, Table 4:

TABLE 4. SIMPLE MEAN OF SECOND DIFFERENCE PHASE

Simple mean $\hat{D}r$ = -6.709E-16 t-ratio = -2.45

Degrees of Freedom = 91

Standard Deviation $\hat{s}$ = 26.2405E-16

Standard Deviation of the Mean = 2.7358E-16

Figure 5 shows the second difference of the phase after the mean was subtracted. Visually, the data appear reasonably white, and the periodogram, Fig. 6, cannot reject the null hypothesis of whiteness. Now the standard error of the drift term is 2.735E-16, 52 times larger than that computed for the quadratic fit! Indeed, the estimated drift term is only 2.45 times its standard error.

V. SUMMARY OF TESTS

The analyses reported above were all performed on a single data set. In order to verify any conclusions, all three analyses used above were performed a total of four times on four different data sets from the same pair of oscillators. Table 5 summarizes the results:

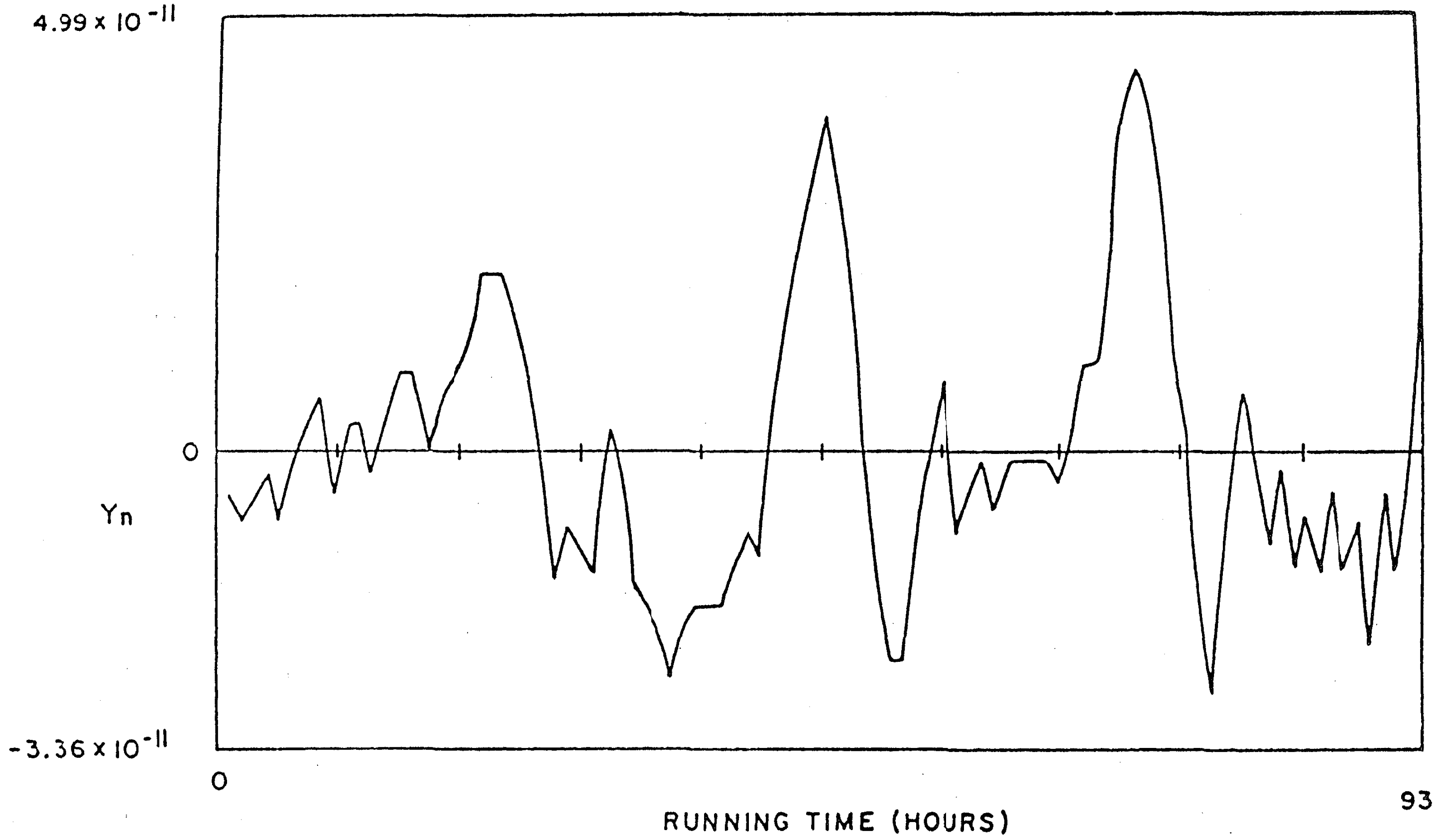


Fig. 3 - Frequency residual
(1st diff-lin)

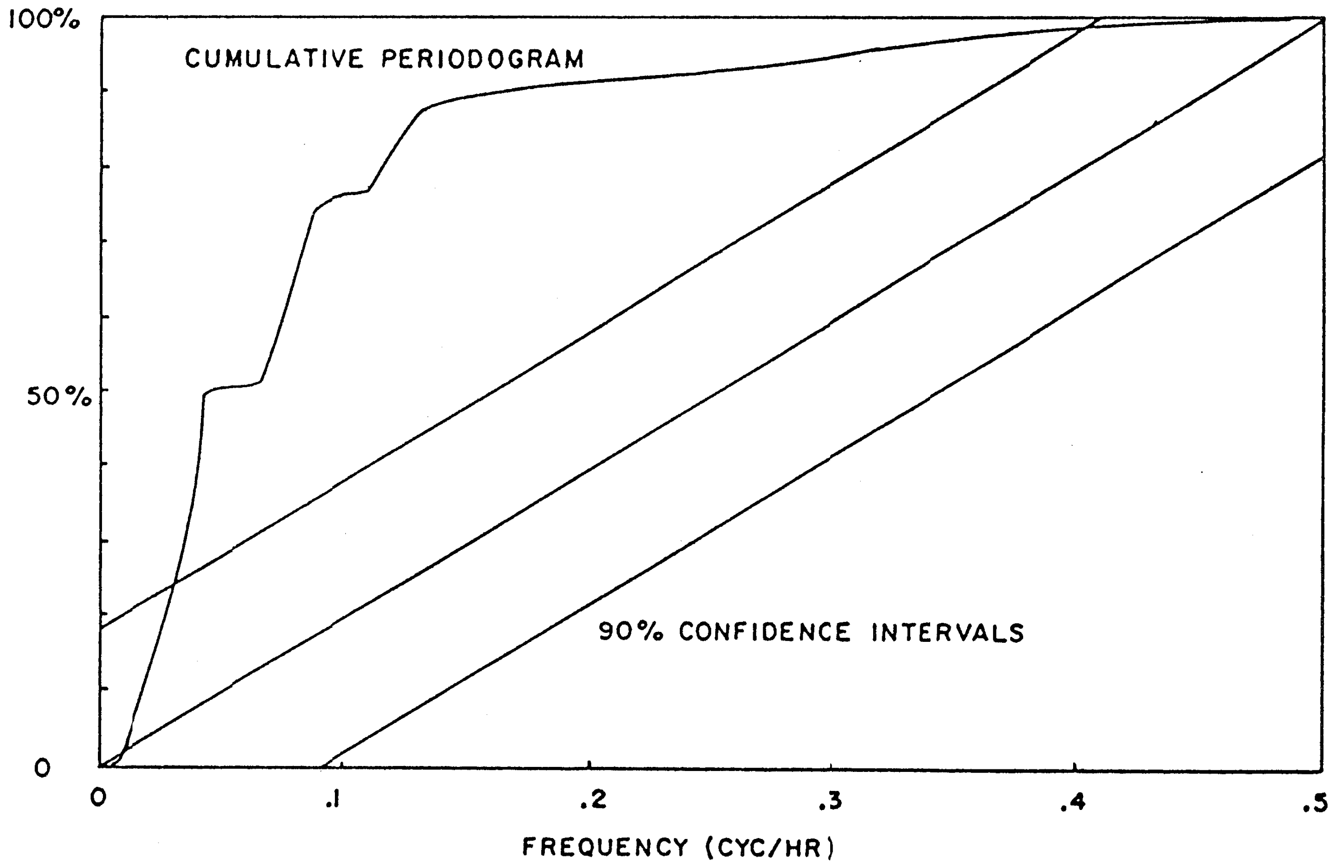


Fig. 4 - First diff - linear

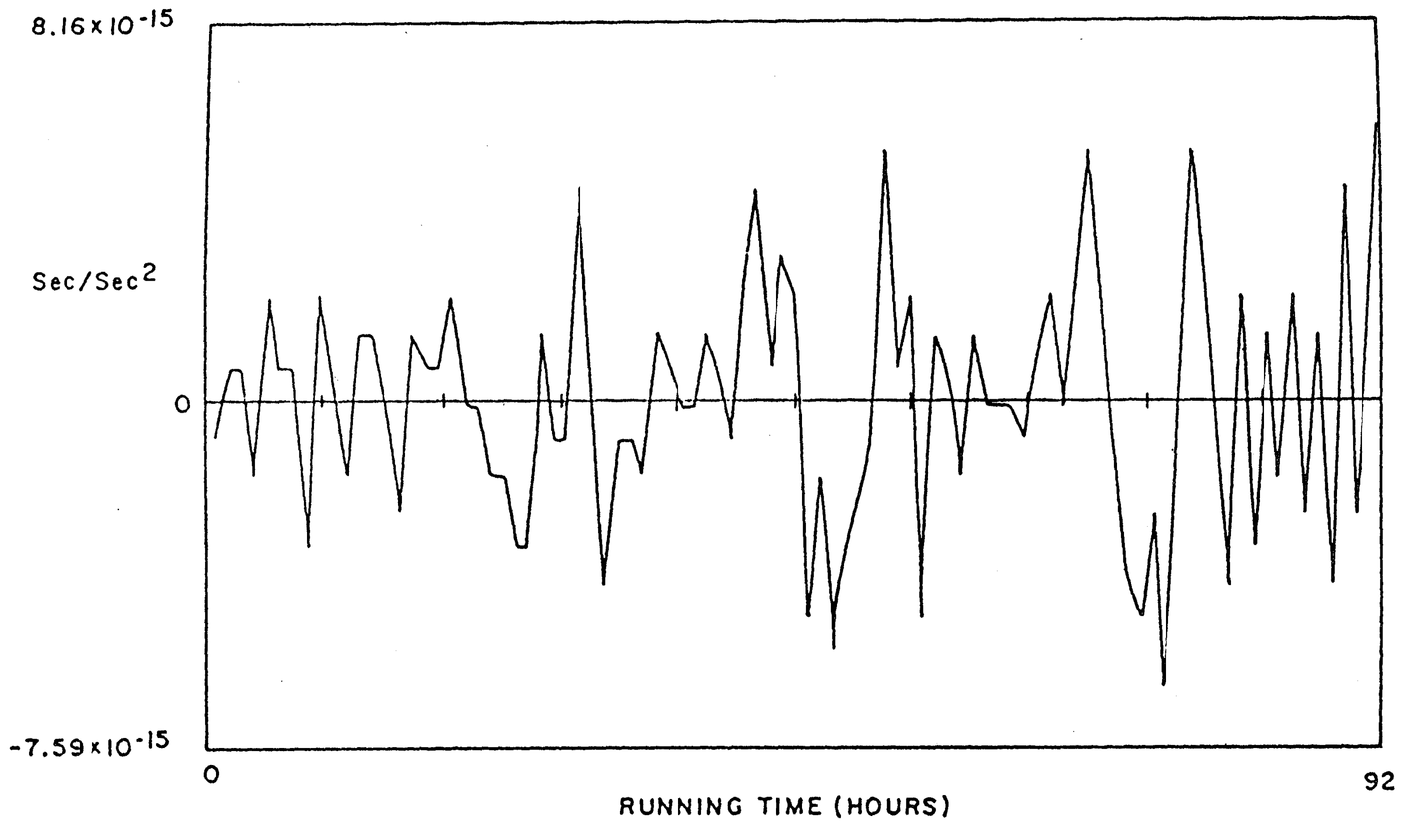


Fig. 5 - Second diff of phase

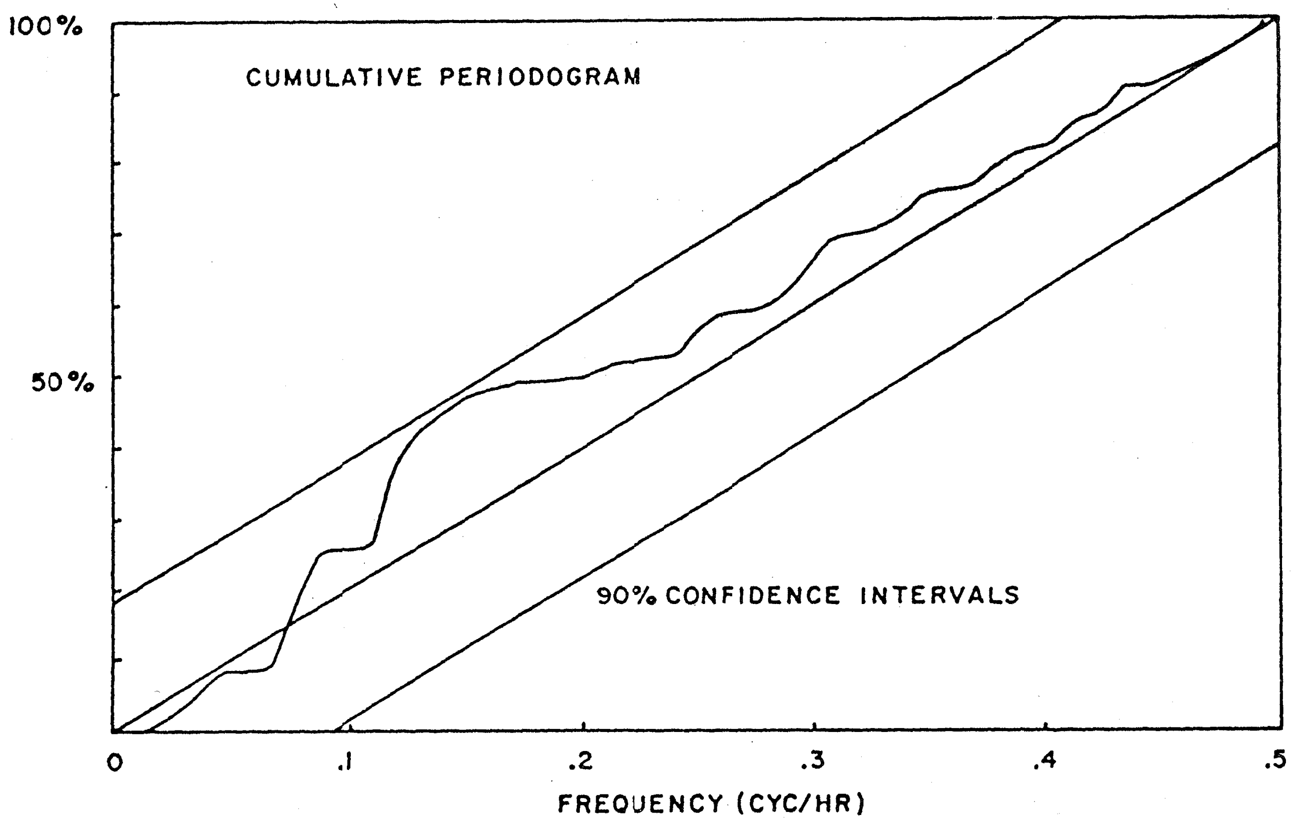


Fig. 6 - Second diff - avg.

TABLE 5. SUMMARY OF DRIFT ESTIMATES
(Units of 1.E-16 sec/sec^2)

ESTIMATION PROCEDURE & MODEL	DRIFT ESTIMATE	COMPUTED STANDARD ERROR	PASS WHITENESS TESTS?
Quad Fit (White PM and Drift)	-7.507	.0523	No
	-8.746	.0493	No
	-6.479	.0645	No
	-6.468	.0880	No
1st Difference Linear Fit (White FM and Drift)	-7.635	.162	No
	-8.558	.206	No
	-6.443	.192	No
	-6.253	.295	No
Second Difference Less Mean (Random Walk FM and Drift)	-6.710	2.736	Yes
	-7.462	9.335	No
	-6.870	3.424	Yes
	-6.412	3.543	Yes

One can calculate the sample means and variances of the drift estimates for each of the three procedures listed in Table 5, and compare these "external" estimates with those values listed in the table under "Computed Standard Error," the "internal" estimates. Of course the sample size is small and we do not expect high precision in the results, but some conclusions can be drawn. The comparisons are shown in Table 6.

It is clear that the quadratic fit to the data displays a very optimistic internal estimate for the standard deviation of the drift rate. Other conclusions are not so clear cut, but still some things can be said. Considering Table 5, the "2nd Diff - Mean" residuals passed the whiteness test three times out of four. The external estimate of the drift standard deviation lies between the internal estimates based on the first and second differences. Since the oscillators under test were crystal oscillators, one expects flicker FM to be present at some level. One also expects the flicker FM behavior to lie between white FM and random walk FM. This may be the explanation of the observed standard deviations, noted in Table 6.

TABLE 6. STANDARD DEVIATIONS

PROCEDURE (Model)	EXTERNAL ESTIMATE (Std. Dev. of Drift Estimates from Col. #2, Table 5)	INTERNAL ESTIMATE (RMS Computed Std. Dev. Col. #3, Table 5)
Quad Fit (White PM)	1.08	0.065
1st Diff - Lin (White FM)	1.08	0.203
2nd Diff - Mean (Rand Wlk FM)	0.44	5.45

VI. DISCUSSION

In all three of the analysis procedures used above, more parameters than just the frequency drift rate were estimated. Indeed, this is generally the case. The estimated parameters included the drift rate, the variance of the random (white) noise component, and other parameters appropriate to the specific model (e.g., the initial frequency offset for the first two models). If these other parameters could be known precisely by some other means, then methods exist to exploit this knowledge and get even better estimates of the drift rate. The real problems, however, seem to require the estimate of several parameters in addition to the drift rate, and it is not appropriate to just ignore unknown model parameters.

To this point, we have considered only three, rather ideal oscillator models, and seldom does one encounter such simplicity. Typical models for commercial cesium beam frequency standards include white FM, random walks FM, and frequency drift. Unfortunately, none of the three estimation routines discussed above are appropriate to such a model. This problem has been solved in some of the recent work of Jones and Tryon[5],[6]. Their estimation routines are based on Kalman Filters and maximum likelihood estimators and these methods are appropriate for the more complex models. For details, the reader is referred to the works of Jones and Tryon.

Still left untreated are the models which, in addition to drift and other noises, incorporate flicker noises, either in PM or FM or both. In principle, the methods of Jones and Tryon could be applied to Kalman Filters which incorporate empirical flicker models[7]. To the author's knowledge, however, no such analyses have been reported.

VII. CONCLUSIONS

There are two primary conclusions to be drawn:

(1) The estimation of the linear frequency drift rates of oscillators and the inclusion of realistic confidence intervals for these

estimates are critically dependent on the adequacy of the model used and, hence the adequacy of the analysis procedures.

(2) The estimation of the drift rate must be carried along with the estimation of any and all other model parameters which are not known precisely from other considerations (e.g., initial frequency and time offsets, phase noise types, etc.)

More and more, scientists and engineers require clocks which can be relied on to maintain accuracy relative to some master clock. Not only is it important to know that on the average the clock runs well, but it is essential to have some measure of time imprecision as the clock ages. For example, the uncertainties might be expressed as, say,, 90% certain that the clock will be within 5 microseconds of the master two weeks after synchronization. Such measures are what statisticians call "interval estimates" (in contrast to point estimates) and their estimations require interval estimates of the clock's model parameters. Clearly, the parameter estimation routines must be reliable and based on sound measurement practices. Some inappropriate estimation routines can be applied to clocks and oscillators and give dangerously optimistic forecasts of performance.

APPENDIX A

REGRESSION ANALYSIS
(Equally Spaced Data)

We begin with the continuous model equation:

$$X(t) = a + b \cdot t + c \cdot t^2 + \phi(t) \tag{A1}$$

We assume that the data is in the form of discrete readings of the dependent variable X(t) at the regular intervals given by:

$$t = n\tau_o \tag{A2}$$

Equation (A1) can then be written in the obvious form:

$$X_n = a + \tau_o\, b \cdot n + \tau_o^2\, c \cdot n^2 + \phi(n\tau_o) \tag{A3}$$

for $n = 1, 2, 3 \ldots, N$.

Next, we define the matrices:

$$\underline{N} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 4 \\ 1 & 3 & 9 \\ '' & '' & '' \\ '' & '' & '' \\ 1 & N & N^2 \end{bmatrix} \qquad \underline{X} = \begin{bmatrix} X_1 \\ X_2 \\ X_3 \\ '' \\ '' \\ X_N \end{bmatrix}$$

$$\underline{T} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \tau_o & 0 \\ 0 & 0 & \tau_o^2 \end{bmatrix}$$

$$(\underline{NT})'\underline{X} = \begin{bmatrix} \sum_1^N X_n \\ \tau_o \sum_1^N X_n\, n \\ \tau_o^2 \sum_1^N X_n\, n^2 \end{bmatrix} = \underline{T} \cdot \begin{bmatrix} \sum_1^N X_n \\ \sum_1^N X_n\, n \\ \sum_1^N X_n\, n^2 \end{bmatrix} = \underline{T} \cdot \begin{bmatrix} S_x \\ S_{nx} \\ S_{nnx} \end{bmatrix}$$

where four quantities must be calculated from the data:

$$S_x = \sum_{n=1}^{N} X_n \qquad S_{nnx} = \sum_{n=1}^{N} X_n \, n^2$$

$$S_{nx} = \sum_{n=1}^{N} X_n \, n \qquad S_{xx} = \sum_{n=1}^{N} X_n{}^2$$

Define

$$\underline{B} = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

With these definitions, Eq. A3 can be rewritten in the matrix form:

$$\underline{X} = \underline{N}\,\underline{T}\,\underline{B} + \underline{\varepsilon} \tag{A4}$$

and the coefficients, B, which minimize the squared errors are given by:

$$\underline{\hat{B}} = \begin{vmatrix} \hat{a} \\ \hat{b} \\ \hat{c} \end{vmatrix} = \underline{T} \cdot (\, \underline{N'N} \,)^{-1} \cdot (\, \underline{N'X} \,) \tag{A5}$$

The advantage of evenly spaced data for these regressions is that, with a bit of algebra, the matrix, $(\, \underline{N'N} \,)^{-1}$, can be written down in closed form:

$$(\, N'N \,)^{-1} = \begin{bmatrix} A & B & C \\ B & D & E \\ C & E & F \end{bmatrix} \cdot 1 / G \tag{A6}$$

where

* $A = 3\,[3\,(N + 1) + 2]$

$B = -18\,(2N + 1)$

$C = 30$

$D = 12\,(2N + 1)\,(8N + 11) \,/\, [(N + 1)\,(N + 2)]$

$E = -180 \,/\, (N + 2)$

$F = 180 \,/\, [(N + 1)\,(N + 2)]$

---

* See Appendix Note # 40

and

$$G = N\,(N - 1)\,(N - 2)$$

Also, the inverse of T is just:

$$\underline{T}^{-1} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/\tau_o & 0 \\ 0 & 0 & 1/\tau_o^2 \end{bmatrix}$$

The complete solution for the regression parameters can be summarized as follows:

There are four quantities which must calculate from the data:

$$S_x = \sum_{n=1}^{N} X_n \qquad S_{nnx} = \sum_{n=1}^{N} X_n\, n^2$$

$$S_{nx} = \sum_{n=1}^{N} X_n\, n \qquad S_{xx} = \sum_{n=1}^{N} X_n^2$$

for n = 1, 2, 3, ..., N. Based on these four quantities, the regression parameters are calculated from the seven following equations:

$$\hat{a} = (A\, S_x + \tau_o\, B\, S_{nx} + \tau_o^2\, C\, S_{nnx}) / G$$

$$\hat{b} = (B\, S_x + \tau_o\, D\, S_{nx} + \tau_o^2\, E\, S_{nnx}) / (G\, \tau_o)$$

$$\hat{c} = (C\, S_x + \tau_o\, E\, S_{nx} + \tau_o^2\, F\, S_{nnx}) / (G\, \tau_o^2)$$

$$\hat{\sigma}^2 = (S_{xx} - \hat{a}\, S_x - \tau_o\, \hat{b}\, S_{nx} - \tau_o^2\, \hat{c}\, S_{nnx}) / (N - 3)$$

$$\hat{\sigma}_a^2 = \hat{\sigma}^2\, A / G$$

$$\hat{\sigma}_b^2 = \hat{\sigma}^2\, D / (G\, \tau_o^2)$$

$$\hat{\sigma}_c^2 = \hat{\sigma}^2\, F / (G\, \tau_o^4)$$

where the coefficients A, B, C, etc., are given by:

$$A = 3\ [3N\ (N + 1) + 2]$$

$$B = -18\ (2N + 1)$$

$$C = 30$$

$$D = 12\ (2N + 1)\ (8N + 11)\ /\ [(N + 1)\ (N + 2)]$$

$$E = -180\ /\ (N + 2)$$

$$F = 180\ /\ [(N + 1)\ (N + 2)]$$

and

$$G = N\ (N - 1)\ (N - 2).$$

In matrix form, the error variance for forecast values is:

$$\text{Var}\ (\hat{X}_k) = \hat{\sigma}^2\ [1 + \underline{N_k}'\ (\underline{N'N})^{-1}\ \underline{N_k}]$$

where $N_k' = [1\ n_o\ n_o]$ and $\tau_o\ n_o$ is the date for the forecast point, $\hat{X}_k$. That is, $n_o = N + K$ and K is the number of lags past the last data point at lag N.

APPENDIX B

REGRESSIONS ON LINEAR AND CUBIC FUNCTIONS

The matrixes $(\underline{N'N})^{-1}$ for the linear fit and cubic fit, which correspond to Eq. A6 in Appendix A are as follows:

For the linear fit:

$$(N'N)^{-1} = \begin{bmatrix} A & B \\ B & C \end{bmatrix} 1 / D$$

where

$A = 2 (2N + 1)$

$B = -6$

$C = 12 / (N + 1)$

and

$D = N (N - 1)$

For the cubic fit:

$$(N'N)^{-1} = \begin{bmatrix} A & B & C & D \\ B & E & F & G \\ C & F & H & I \\ D & G & I & J \end{bmatrix} 1 / K$$

where

$A = 8 (2N + 1) (N + N + 3)$

$B = -20 (6N^2 + 6N + 5)$

$C = 120 (2N + 1)$

$D = -140$

$E = 200 (6N^4 + 27 N^3 + 42 N^2 + 30 N + 11) / L$

$F = -300 (N + 1) (3N + 5) (3N + 2) / L$

$G = 280 (6N^2 + 15N + 11) / L$

$$H = 360\ (2N + 1)\ (9N + 13)\ /\ L$$

$$I = -4200\ (N + 1)\ /\ L$$

$$J = 2800\ /\ L$$

and

$$K = N\ (N - 1)\ (N - 2)\ (N - 3)$$

$$L = (N + 1)\ (N + 2)\ (N + 3).$$

The restrictions on these equations are that the data is evenly spaced beginning with $n = 1$ to $n = N$, and no missing values. For error estimates (and their distributions) to be valid, the residuals must be random, uncorrelated, (i.e., white).

REFERENCES

1. D.W. Allan, "The Measurement of Frequency and Frequency Stability of Precision Oscillators," Proc. 6th PTTI Planning Meeting, p109 (1975).

2. J.A. Barnes, et al., "Characterization of Frequency Stability," Proc IEEE Trans. on Inst. and Meas., IM-20, p105 (1971).

3. N.R. Draper and H. Smith, "Applied Regression Analysis," John Wiley and Sons, N.Y., (1966).

4. G.E.P. Box and G.M. Jenkins, "Time Series Analysis," Holden-Day, San Francisco, p294, (1970).

5. P.V. Tryon and R.H. Jones, "Estimation of Parameters in Models for Cesium Beam Atomic Clocks," NBS Journal of Research, Vol. 88, No. 1, Jan-Feb. 1983.

6. R.H. Jones and P.V. Tryon, "Estimating Time from Atomic Clocks," NBS Journal of Research, Vol. 88, No. 1, Jan-Feb. 1983.

7. J.A. Barnes and S. Jarvis, Jr., "Efficient Numerical and Analog Modeling of Flicker Noise Processes," NBS Tech. Note 604, (1971).

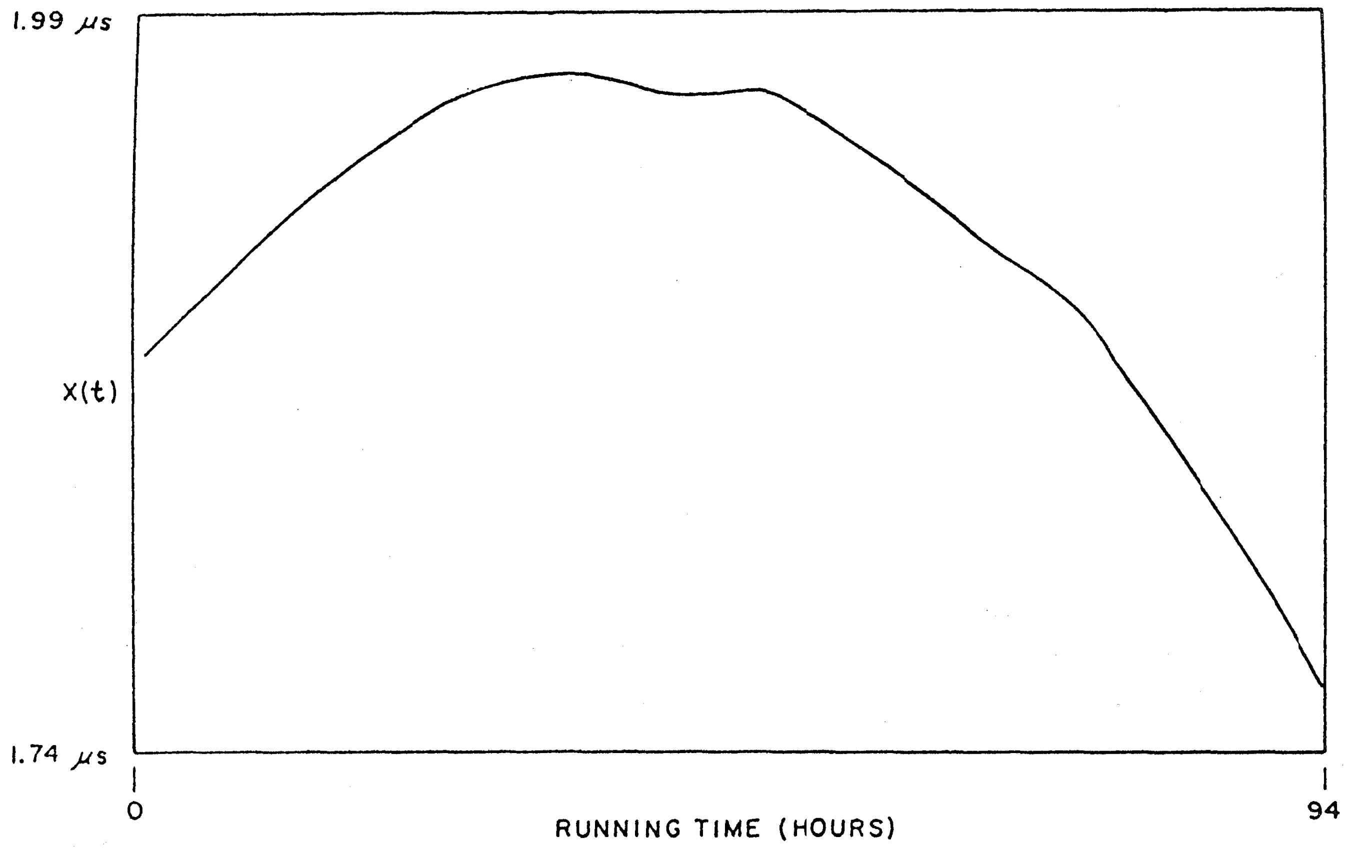


Fig. 1 - Phase difference

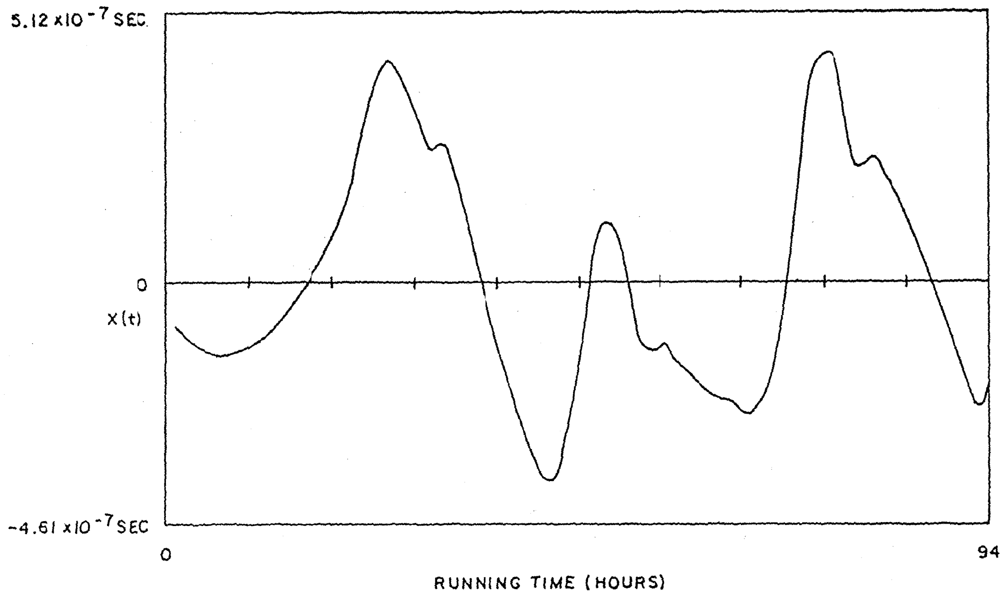


Fig. 2 - Residuals from quadratic fit to phase

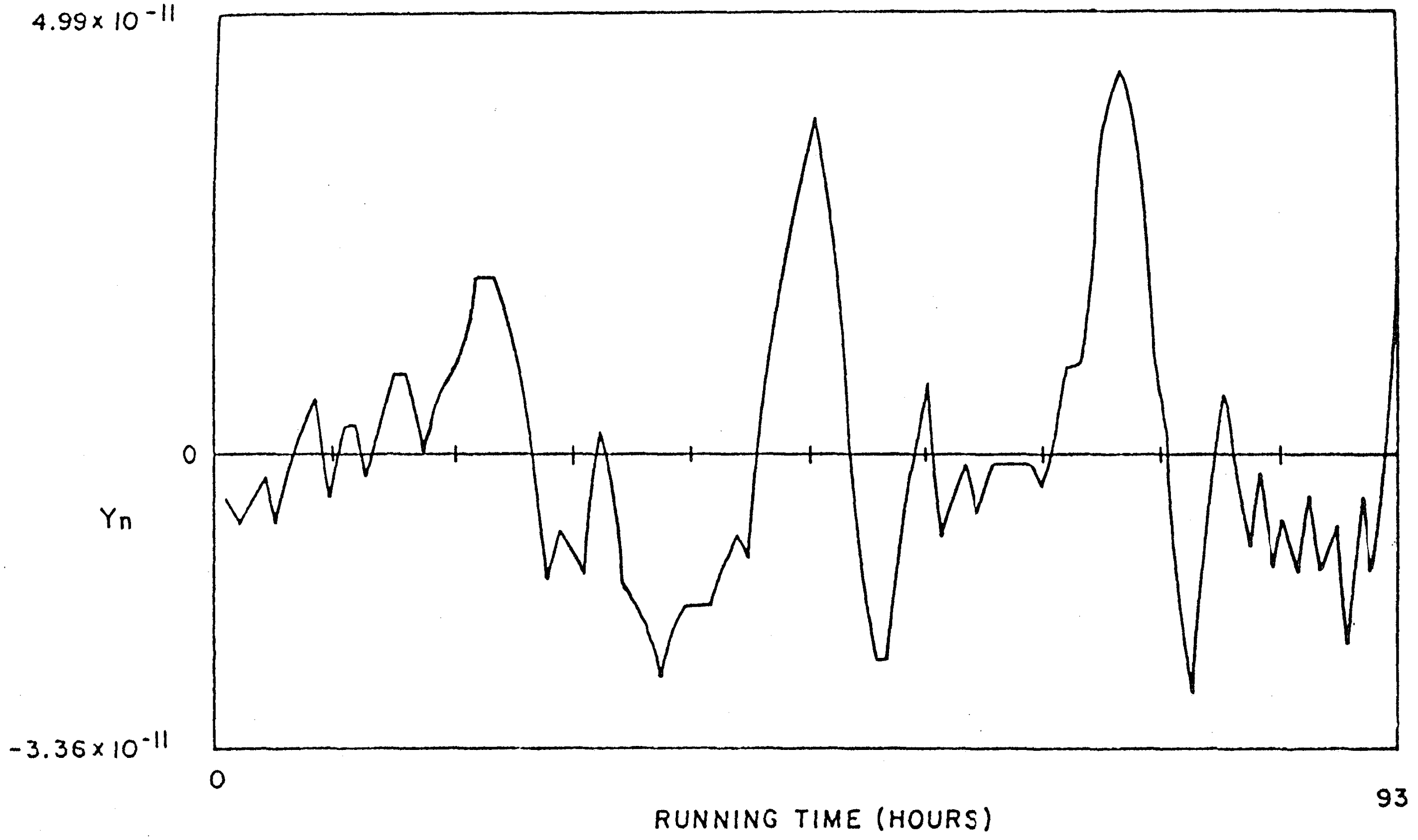


Fig. 3 - Frequency residual
(1st diff-lin)

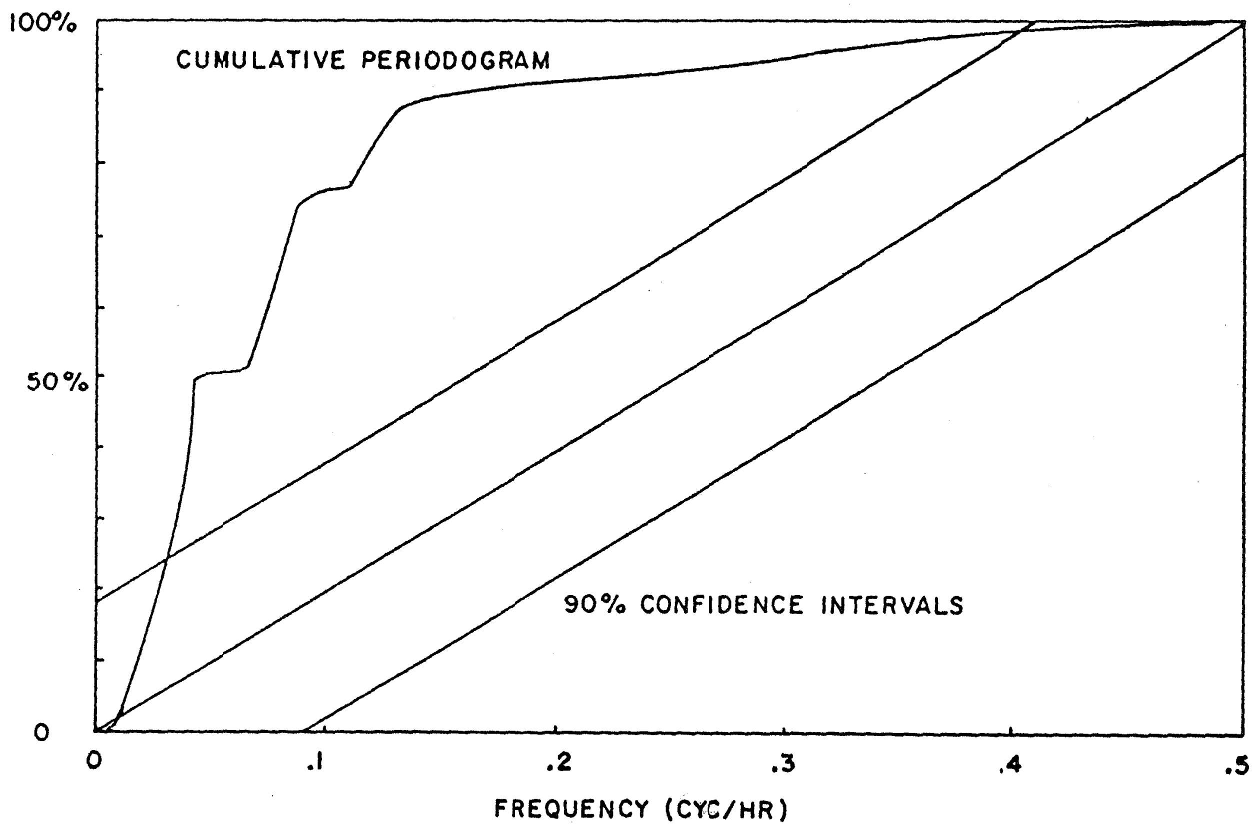


Fig. 4 - First diff - linear

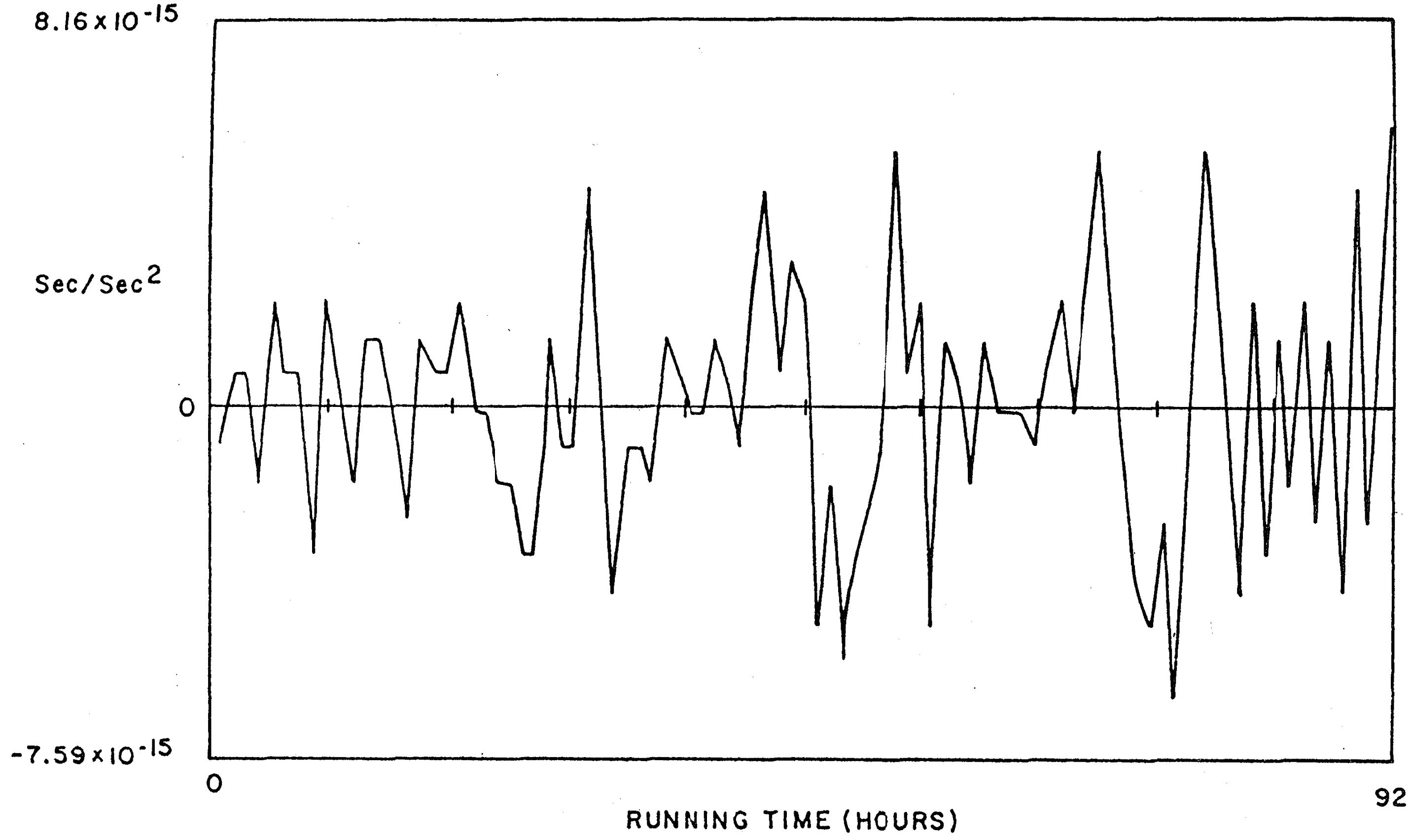


Fig. 5 - Second diff of phase

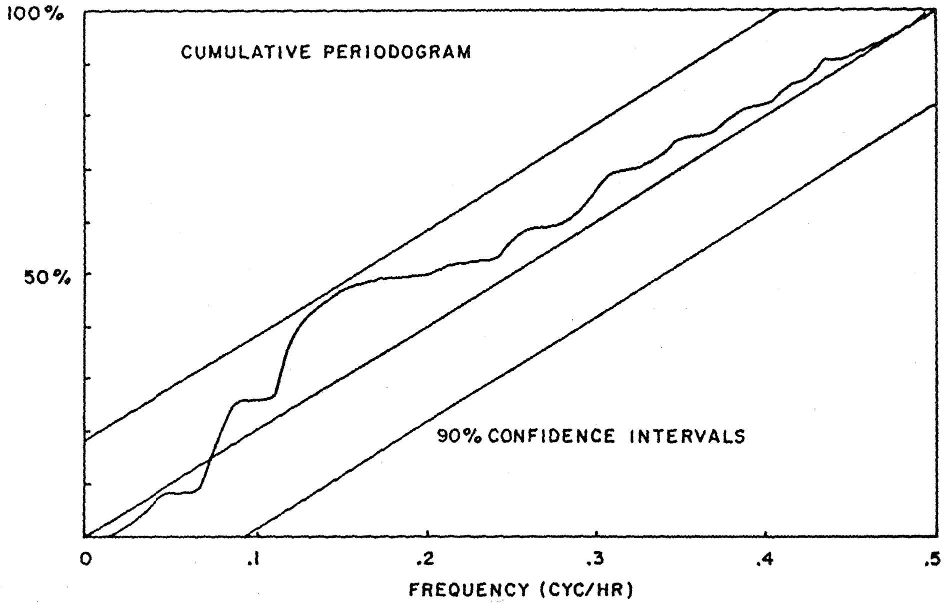
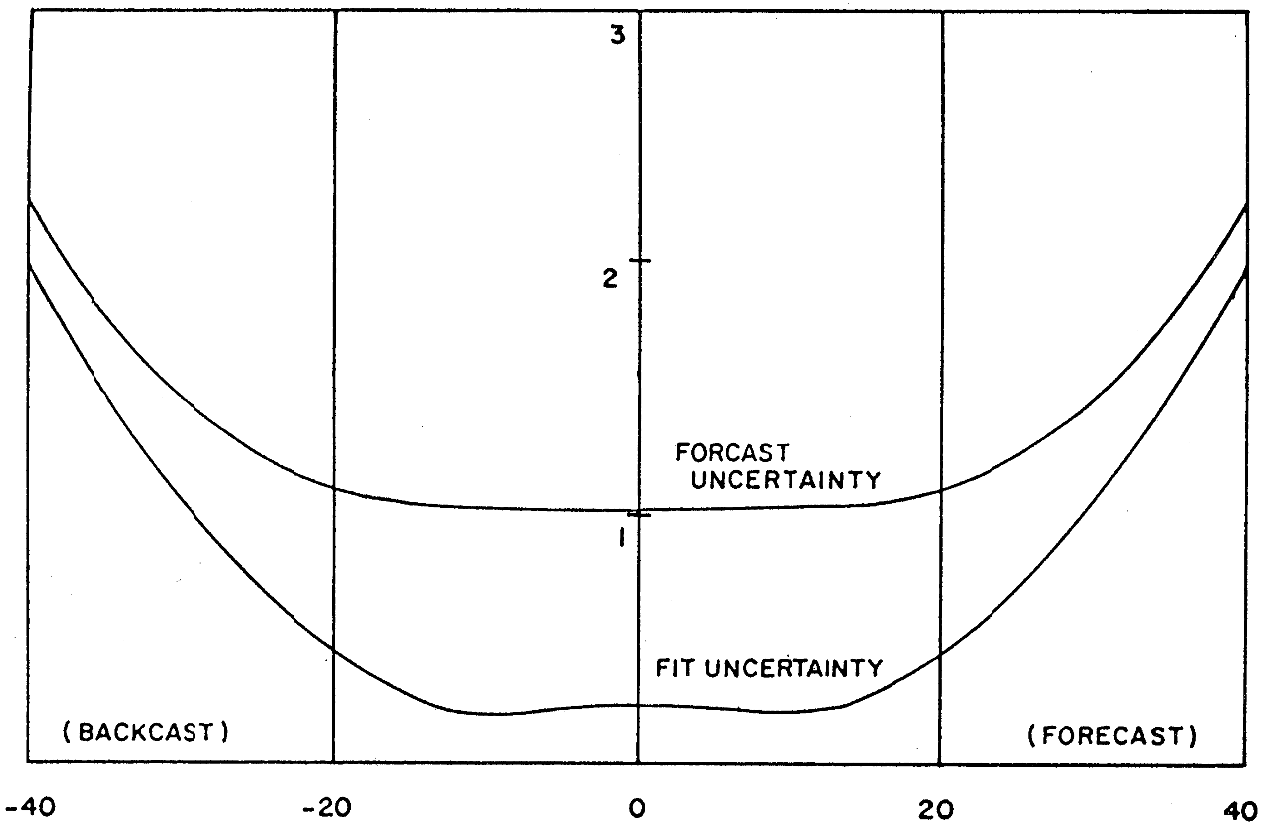
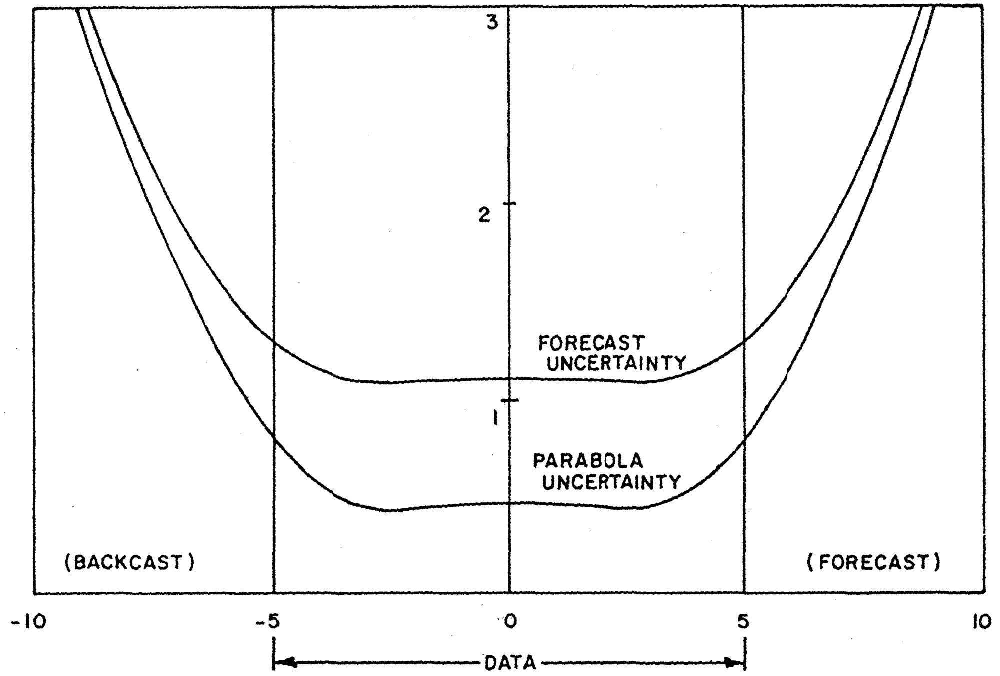


Fig. 6 - Second diff - avg.


FREQUENCY DRIFT AND WHITE PM

STANDARD DEVIATIONS

FREQUENCY DRIFT AND WHITE PM
STANDARD DEVIATIONS



QUESTIONS AND ANSWERS

MR. ALLAN:

We have found the second difference drift estimator to be useful for the random walk FM process. One has to be careful when you apply it, because if you use overlapping second differences, for example, if you have a cesium beam, and you have white noise FM out to several days and then random walk FM beyond that point, and you are making hourly or 2 hour measurements, if you use the mean of the second differences, you can show that all the middle terms cancel, and in fact you are looking at the frequency at the beginning of the run and the frequency at the end of the run to compute the frequency drift and that's very poor.

DR. BARNES:

My comment would be: There again the problem is in the model, and not in the arithmatic. The models applied here were 3 very simple models, very simple, simpler than you will run into in life. It was pure random walk plus a drift or pure frequency noise plus a drift, or pure random walk of frequency noise plus a drift and it did not approach at all any of the noise complex models where you would have both white frequency and random walk frequency and a drift. That's got to be handled separately, I'm not even totally sure how to perform it in all cases at this point.

MR. McCASKILL:

I would like to know if you would comment on the value of the sample time and the reason why, of course, is that the mean second difference does depend on the sample time? For instance, if you wanted to estimate the aging rate or change in linear change of frequency, what value of sample time would you use in order to make that correction. So, really, the question is, how does the sample time enter into your calculations?

DR. BARNES:

At least I will try to answer in part. I don't know in all cases, I'm sure, but if you have a complex noise process where you have at short term different noise behavior than in long term, it may benefit you to take a longer sampling time and effectively not look at the short term, and then one of the simple models might apply. I honestly haven't looked in great detail at how to choose the sample time. It is an interesting question.

MR. McCASKILL:

Well, let me go further, because we had the benefit of being out at NBS and talking with Dr. Allan earlier and he suggested that we use the mean second difference, and the only problem is if we want to calculate or correct for our aging rate at a tau of five days or ten days, and you calculate the mean second difference, you come up with exactly the Allan Variance. What appears is that in order to come up with the number for the aging rate, you have to calculate the mean second difference using a sample time whenever you take your differences of longer than, let's say, ten days sample time. So we use, of course, the regression model, but we use on the order of two or three weeks in order to calculate an aging rate correction for, say, something like the rubidium in the NAVSTAR 3 clock. It looks like in order to come up with a valid value for the Allan Variance of five or ten day sample time you have to calculate the aging rate at a longer, maybe two or three times longer sample time.

DR. BARNES:

Dave Allan, do you think you can answer that?

MR. ALLAN:

Not to go into details, but if you assume that in the longer term you have random walk frequency modulation as the predominant noise process in a clock, which seems to be true for rubidium, cesium and hydrogen, you can do a very simple thing. You can take the full data length and take the time at the beginning, the time in the middle and the time in the end and construct a second difference, and that's your drift.

There is still the issue of the confidence interval on that. If you really want to verify your confidence interval, you have to have enough data to do a regression. You need enough data to test to be sure the model is good.

DR. WINKLER:

That argument is fourteen years old, because we have been critized here. I still believe that for a practical case where you depend on measurements which are contaminated and maybe even contaminated by arbitrarily large errors if they are digital. These, theoretical advantages that you have outlined may not be as important as the benefits which you get when you make a least square regression. In this case you can immediately identify the wrong data. Otherwise, if you put the data into an algorithm you may not know how much your data is contaminated. So for practical applications, the first model even though theoretically it is poor, it still gives reasonable estimates of the drift and you have residuals which let you identify wrong phase values immediately.

DR. BARNES:

I think that's true and you may have different reasons to do regression analysis, if your purpose is to measure a drift and understand the confidence intervals, then I think what has been presented is reasonable, if you have as your purpose to look--to see if there are indications of funny behavior in a curve that has such strong curvature or drift that you can't get it on graph paper without doing that, I think it is a very reasonable thing to do. I think looking at the data is one of the healthiest things any analyst can do.

NIST Technical Note 1318, 1990.

VARIANCES BASED ON DATA WITH DEAD TIME BETWEEN THE MEASUREMENTS

James A. Barnes
Austron, Inc.
Boulder, Colorado 80301

and

David W. Allan
Time and Frequency Division
National Institute of Standards and Technology
Boulder, Colorado 80303

The accepted definition of frequency stability in the time domain is the two-sample variance (or Allan variance). It is based on the measurement of average frequencies over adjacent time intervals, with no "dead time" between the intervals. The primary advantages of the Allan variance are that (1) it is convergent for many encountered noise models for which the conventional variance is divergent; (2) it can distinguish between many important and different spectral noise types; (3) the two-sample approach relates to many practical implementations; for example, the rms change of an oscillator's frequency from one period to the next; and (4) Allan variances can be easily estimated at integer multiples of the sample interval.

In 1974 a table of bias functions which related variance estimates with various configurations of number of samples and dead time to the Allan variance was published [1]. The tables were based on noises with pure power-law spectral densities.

Often situations occur that unavoidably have dead time between measurements, but still the conventional variances are not convergent. Some of these applications are outside of the time-and-frequency field. Also, the dead times are often distributed throughout a given average, and this distributed dead time is not treated in the 1974 tables.

This paper reviews the bias functions $B_1(N,r,\mu)$, and $B_2(r,\mu)$ and introduces a new bias function, $B_3(2,M,r,\mu)$, to handle the commonly occurring cases of the effect of distributed dead time on the computed variances. Some convenient and easy-to-interpret asymptotic limits are reported. A set of tables for the bias functions are included at the end of this paper.

Key words: Allan variance; bias functions; data sampling and dead time; dead time between the measurement; definition of frequency stability; distributed dead time; two-sample variance

1. Introduction

The sample mean and variance indicate respectively the approximate magnitude of a quantity and its uncertainty. For many situations a continuous function of time is sampled, or measured, at fairly regular intervals. Sampling is not always instantaneous. It takes a finite time and provides an "average reading." If the underlying process (or noise) is random and uncorrelated in time, then the fluctuations are said to be "white" noise. In this situation, the sample mean and variance calculated by the conventional formulas,

$$m = \frac{1}{N} \sum_{n=1}^{N} \bar{y}_n ,$$

$$s^2 = \frac{1}{N-1} \sum_{n=1}^{N} (\bar{y}_n - m)^2 , \qquad (1)$$

provide the needed information. The "bar" over the y in eq (1) above denotes the average over a finite time interval. In time and frequency work, y is defined as the average fractional (or normalized) frequency deviation from nominal over an interval τ and at some specified measurement time. As in science generally, the physical model determines the appropriate mathematical model. For the white noise model, the sample mean and variance are the mainstays of most analyses.

Although white noise is a common model for many physical processes, more general noise models are being identified and used. In precise time and frequency measurement, for example, there are two quantities of great interest: instantaneous frequency and phase. These two quantities by definition are exactly related by a differential. (We are NOT considering Fourier frequencies at this point.) That is, the instantaneous frequency is the time rate of change of phase. Thus, if we were employing a model of white frequency-modulation (white FM) noise, then the phase noise is the integral of the white FM noise, commonly called a Brownian motion or random walk. Therefore, depending on whether we are currently interested in phase or frequency, the sample mean and variance may or may not be appropriate.

By definition, white noise has a power spectral density (PSD) that is constant with Fourier frequency. Since random walk noise is the integral of white noise, the power spectral density of a random walk varies as $1/f^2$ (where f is the Fourier frequency) [2]. We encounter noise models whose power spectral densities are various power laws of their Fourier frequencies. Flicker noise is very common and is defined as a noise whose power spectral density varies as $1/f$ over a relevant spectral range. If an oscillator's instantaneous frequency is well modeled by flicker noise, then its phase would be the integral of the flicker noise. It would have a PSD which varied as $1/f^3$.

Noise models whose PSD's are power laws of the Fourier frequency but not integer exponents are possible as well but not as common. This paper considers power-law PSD's of a quantity $y(t)$; $y(t)$ is a continuous sample function which can be measured at regular intervals. For noises whose PSD's vary as f^α with $\alpha < -1$ at low frequencies, the conventional sample mean and variance given in eq (1) do not converge as N gets large [2, 3]. This lack of convergence renders the sample mean and variance ineffective and often misleading in some situations.

Although the sample mean and variance have limitations, other time-domain statistics can be convergent and quite useful. The quantities that we consider in this paper depend significantly on the details of the sampling procedures. Indeed, each sampling scheme has its own bias, and this is the motivation for the bias functions discussed in this paper.

2. The Allan Variance

Recognizing that for particular types of noise, the conventional sample variance fails to converge as the number of samples, N, grows, Allan suggested that we set $N = 2$ and average many of these two-sample variances to get a convergent and stable measure of the spread of the quantity in question [3]. This is what has come to be called the Allan variance.

More specifically let us consider a sample function of time as indicated in figure 1. A measurement consists of averaging y(t) over the interval τ. The next measurement begins at a time T after the beginning of the previous measurement interval. There is no logical reason why T must be as large as τ or larger--if $T < \tau$, then the second measurement begins before the first is completed, which is unusual but possible. When $T = \tau$, there is no dead time between measurements.

The accepted definition of the Allan variance is the expected value of a two-sample variance with no dead time between successive measurements. In symbols, the Allan variance is given by

$$\sigma_y^2(\tau) = \tfrac{1}{2}E[(\bar{y}_{n+1}-\bar{y}_n)^2], \qquad (2)$$

where there is no dead time between the two sample averages for the Allan variance and the E[•] denotes the expectation operator.

3. The Bias Function $B_1(N,r,\mu)$

Define N to be the number of sample averages of y(t) used in eq (1) to estimate a sample variance (N = 2 for an Allan variance). Also define r to be the ratio of T to τ (r = 1 when there is no dead time between measurements). The parameter μ is related to the exponent of the power law of the PSD of the process y(t). If α is the exponent in the power-law spectrum for y(t), then the Allan variance varies as τ raised to the μ power, where α and μ are related as shown in figure 2 [2-4]. We can use estimates of μ to infer α, the spectral type. The ambiguity in α for $\mu = -2$ has been resolved by using a modified $\sigma_y^2(\tau)$ [5-7].

Often data cannot be taken without dead time between sample averages, and it is useful to consider other than two-sample variances. We will define the bias function $B_1(N,r,\mu)$ by the ratio,

$$B_1(N,r,\mu) = \frac{\sigma^2(N,T,\tau)}{\sigma^2(2,T,\tau)}, \qquad (3)$$

where $\sigma^2(N,T,\tau)$ is the expected sample variance given in eq (1) and based on N measurements at intervals T and averaged over a time τ and $r = T/\tau$. In words, $B_1(N,r,\mu)$ is the ratio of the expected variance for N measurements to the expected variance for two samples (everything else held constant). The variances on the right in eq (3) depend implicitly on the noise type even though μ or α are not shown as independent variables. The noise-type parameter, μ, is shown as an independent variable for all of the bias functions in this paper, because the values of the ratio of these variances explicitly depend on μ as will be derived later in the paper. Allan showed that if N and r are held constant, then the α, μ relationship shown in figure 2 is the same; that is, we can still infer the spectral type from the τ dependence using the equation $\alpha = -\mu-1$, $-2 \le \mu < 2$ [3].

4. The Bias Function $B_2(r,\mu)$

The bias function $B_2(r,\mu)$ is defined in [1] by the relation,

$$B_2(r,\mu) = \frac{\sigma^2(2,T,\tau)}{\sigma^2(2,\tau,\tau)} = \frac{\sigma^2(2,T,\tau)}{\sigma_y^2(\tau)} . \qquad (4)$$

In words, $B_2(r,\mu)$ is the ratio of the expected two-sample variance with dead time to that without dead time (with N = 2 and τ the same for both variances). A plot of the $B_2(r,\mu)$ function is shown in figure 3. The bias functions B_1 and B_2 represent biases relative to N = 2 rather than infinity; that is, the ratio of the N sample variance (with or without dead time) to the Allan variance and the ratio of the two-sample dead-time variance to the Allan variance respectively.

5. The Bias Function $B_3(N,M,r,\mu)$

Consider the case where a great many measurements are available with dead time between each pair of measurements ($T_o > \tau_o$). The measurements are averaged over the time interval τ_o, the spacing between the beginning of one measurement to the next is T_0, and it may not be convenient to retake the data. We might want to estimate the Allan variance at, say, multiples M of the averaging time τ_o. If we average groups of the measurements of y(t), then the dead times between the original measurements are distributed periodically throughout the new average measurements (see figure 4). Define

$$\bar{\bar{y}}_i = \frac{1}{M} \sum_{n=i}^{M+i-1} \bar{y}_n , \tag{5}$$

where $\bar{y}_i$ are the raw or original measurements based on dead time $T_0 - \tau_0$.

Also define the two-sample variance with distributed dead time as

$$\sigma^2(2,M,T,\tau) = \tfrac{1}{2}E[(\bar{\bar{y}}_i - \bar{\bar{y}}_{i+M})^2], \tag{6}$$

with $\tau = M\tau_0$ and $T = MT_0$.

We can now define B_3 as the ratio of the N-sample variance with distributed dead time to the N-sample variance with dead time accumulated at the end as in figure 1:

$$B_3(N,M,r,\mu) = \frac{\sigma^2(N,M,T,\tau)}{\sigma^2(N,T,\tau)} . \tag{7}$$

Although $B_3(N,M,r,\mu)$ is defined for general N, the tables in the Appendix confine treatment to the case where N = 2. There is little value in extending the tables to include general N. Though the variances on the right in eq (7) depend explicitly on N, T and τ, the ratio $B_3(N,M,r,\mu)$ depends on the ratio $r = T/\tau$, and on μ as developed later in this paper.

In words, $B_3(2,M,r,\mu)$ is the ratio of the expected two-sample variance with periodically distributed dead time, as shown in figure 4, to the expected two-sample variance with all the dead time grouped together as shown in figure 1. Both the numerator and the denominator have the same total averaging time and dead time, but they are apportioned differently. The product $B_2(r,\mu) \cdot B_3(2,M,r,\mu)$ is the distributed dead-time variance over the Allan variance for a particular T, τ, M and μ.

Some useful asymptotic forms of B_3 can be found. In the case of large M and $M > r$, we may write that

$$B_3 \simeq \frac{1 + \mu}{3}, \qquad 1 \le \mu \le 2,$$
$$B_3 \simeq \frac{4\ \ell n(2)}{2\ \ell n(r) + 3}, \qquad \mu = 0. \tag{8}$$

One simple and important conclusion from these two equations is that for the cases of flicker FM noise and random-walk FM noise, the τ^μ dependence for large τ is the same whether or not there is periodically distributed dead time. The values of the variances differ only by a constant, and in the latter case the constant is 1. This conclusion is also true for white FM noise, and in this case the constant is also 1.

In the cases $r \gg 1$ and $-2 \le \mu \le -1$, we may write for the asymptotic behavior of B_3

$$B_3 \simeq M^\alpha, \ \alpha = -\mu-1 , \tag{9}$$

as was determined empirically. In this region of power-law spectrum the B_3 function has an M^α dependence for an f^α spectrum.

6. The Bias Functions

The bias functions can be written fairly simply by first defining the function,

$$F(A) = 2A^{\mu+2} - (A + 1)^{\mu+2} - |(A - 1)|^{\mu+2}. \tag{10}$$

The bias functions become

$$B_1(N,r,\mu) = \frac{1 + \sum\limits_{n=1}^{N-1} \dfrac{N-n}{N(N-1)} \cdot F(nr)}{1 + \frac{1}{2}\ F(r)}, \tag{11}$$

$$B_2(r,\mu) = \frac{1 + \frac{1}{2}F(r)}{2(1-2^{\mu})}, \qquad (12)$$

as given in [1], and

$$B_3(2,M,r,\mu) = \frac{2M + M\cdot F(Mr) - \sum_{n=1}^{M-1} (M-n)[2F(nr)-F((M+n)r) - F((M-n)r)]}{(M^{\mu+2})[F(r) + 2]}, \qquad (13)$$

as indicated in the appendix.

For $\mu = 0$, eqs (11), (12), and (13) are the indeterminate form 0/0 and must be evaluated by l'Hôpital's rule. Special attention must also be given when expressions of the form 0^0 arise. We verified a random sampling of the table entries using noise simulation and Monte Carlo techniques. No errors were detected. The results in this paper differ some from those in [8], which suggests that there may be some mistakes. Tables for the three bias functions are listed at the end of the paper (note that the computer print-out did not have a symbol for Greek mu = μ).

7. Examples of the Use of the Bias Functions

The spectral type, that is, the value of μ, may be inferred by varying τ, the sample time. However, another useful way of determining the value of μ is by using $B_1(N,r,\mu)$ as follows: calculate an estimate of $\sigma_y^2(N,T,\tau)$ and $\sigma_y^2(2,T,\tau)$ and hence $B_1(N,r,\mu)$; then use the tables to infer the value of μ.

Suppose one has an experimental value for $\sigma_y^2(N_1, T_1, \tau_1)$ and its spectral type is known, that is, μ is known. Suppose also that one wishes to know the variance at some other set of measurement parameters, N_2, T_2, τ_2. An unbiased estimate of $\sigma_y^2(N_2,T_2,\tau_2)$ may be calculated by the equation:

$$\sigma_y^2(N_2,T_2,\tau_2) = \left(\frac{\tau_2}{\tau_1}\right)^{\mu} \left(\frac{B_1(N_2,r_2,\mu)B_2(r_2,\mu)}{B_1(N_1,r_1,\mu)B_2(r_1,\mu)}\right) \sigma_y^2(N_1,T_1,\tau_1) \qquad (14)$$

where $r_1 = T_1/\tau_1$ and $r_2 = T_2/\tau_2$.

Since the time-domain definition for frequency stability is the Allan variance, it behooves us, where possible, to relate other variances to the Allan variance. If we have an N-sample variance on data with dead-time T-τ and we know the power-law spectral type (the value of μ), then we may write

$$\sigma_y^2(\tau) = \frac{\sigma_y^2(N,T,\tau)}{B_1(N,r,\mu)\ B_2(r,\mu)}. \qquad (15)$$

If we have an N-sample variance where each data entry is an average of M samples with distributed dead time, then we may write

$$\sigma_y^2(\tau) = \frac{\sigma_y^2(N,M,T,\tau)}{B_1(N,r,\mu)\ B_2(r,\mu)\ B_3(N,M,r,\mu)}. \qquad (16)$$

8. Conclusion

For some important power-law spectral density models often used in characterizing precision oscillators ($S_y(f) \sim f^\alpha$, $\alpha = -2, -1, 0, +1, +2$), we have studied the effects on variances when there is dead time between the frequency samples, and the frequency samples are averaged to increase the integration time. Since dead time between measurements is a common problem throughout metrology, the analysis here has broader applicability than just to time and frequency. Specifically, this kind of analysis has been used with gage blocks and standard volt cells--showing that the classical variance may be non-convergent in some cases [9].

Heretofore, the Allan variance has been shown to have some convenient theoretical properties in relation to power-law spectra as the integration or sample time is varied (if $\sigma_y{}^2(\tau) \sim \tau^\mu$, then $\alpha = -\mu - 1$, $-2 < \mu \le 2$). Since $\sigma_y(\tau)$, by definition, is estimated from data with no dead time, the sample or integration time can be unambiguously changed to investigate the τ dependence. From our analysis, we have concluded that for the asymptotic limit of several samples being averaged with dead time present in the data, the τ dependence of the variances is the same. The $\alpha = -\mu - 1$ relationship still remains valid for white FM noise ($\mu = -1$, $\alpha = 0$), flicker FM noise ($\mu = 0$, $\alpha = -1$), and for random-walk FM noise ($\mu = +1$, $\alpha = -2$). The asymptotic limit is approached as the product of number of samples averaged and the initial data sample time, τ_0, becomes larger than the dead time ($M > r$). The variances so obtained differ only by a constant, which can be calculated as given in this paper.

A knowledge of the appropriate power-law spectral model is required to translate a distributed dead-time variance to the corresponding value of the Allan variance. In principle, the power-law spectral model can be estimated from the τ^{μ} dependence, using the variance analysis on the data as outlined above.

9. References

[1] J.A. Barnes, "Tables of Bias Functions, B_1 and B_2, for Variances Based on Finite Samples of Processes with Power Law Spectral Densities," NBS Tech. Note 375 (1969).

[2] J.A. Barnes, "Atomic Timekeeping and the Statistics of Precision Signal Generation," IEEE Proc. 54, No. 2, pp. 207-220, Feb. 1966.

[3] D.W. Allan, "Statistics of Atomic Frequency Standards, "IEEE Proc. 54, No. 2, pp. 221-230, Feb. 1966.

[4] RFC Vessot, L. Mueller, and J. Vanier, "The Specification of Oscillator Characteristics from Measurements Made in the Frequency Domain," IEEE Proc. 54, No. 2, pp. 199-207, Feb. 1966.

[5] D.W. Allan and J.A. Barnes, "A Modified "Allan Variance" With Increased Oscillator Characterization Ability," Proc. 35th Annual Frequency Control Symposium, USAERADCOM, Ft. Monmouth, NJ, May 1981, pp. 470-475.

[6] D.W. Allan, "Time and Frequency (Time-Domain) Characterization, Estimation, and Prediction of Precision Clocks and Oscillators," IEEE Transactions on UFFC, November 1987.

[7] P. Lesage and T. Ayi, "Characterization of Frequency Stability: Analysis of the Modified Allan Variance and Properties of its Estimate," IEEE Trans. Instrum. Meas., IM-33, no. 4, pp. 332-336, Dec. 1984.

[8] N.D. Faulkner and E.V.I. Mestre, "Time-Domain Analysis of Frequency

Stability Using Non-zero Dead-Time Counter Techniques," IEEE Trans. on Instrum. & Meas., IM-34, 144-151 (1985).

[9] D.W. Allan, "Should the Classical Variance Be Used as a Basic Measure in Standards Metrology?" IEEE Trans. on Instrumentation and Measurement, IM-36, 646-654, 1987.

Table 1. Table of some bias function identities

$B_1(2,r,\mu) = 1$
$B_1(N,r,2) = (N(N+1))/6$
$B_1(N,1,1) = N/2$
$B_1(N,1,\mu) = (N(1-N^{\mu}))/[(2(N-1)(1-2^{\mu})]$ for $\mu \neq 0$
$= N \ln(N)/[2(N-1) \ln(2)]$ for $\mu = 0$
* $B_1(N,1,\mu) = 1$ for $\mu < 0$

$$= [2/(N(N-1)] \sum_{n=1}^{N-1} (N-n) \cdot n^{\mu} \text{ for } \mu > 0$$

$B_1(N,r,-1) = 1$ if $r \geq 1$
$B_1(N,r,-2) = 1$ if $r \neq 1$ or 0
$B_2(0,\mu) = 0$
$B_2(1,\mu) = 1$
$B_2(r,2) = r^2$
$B_2(r,1) = (3r - 1)/2$ if $r \geq 1$
$B_2(r,-1) = r$ if $0 \leq r \leq 1$
$= 1$ if $r \geq 1$
$B_2(r,-2) = 0$ if $r=0$
$= 1$ if $r=1$
$= 2/3$ otherwise
$B_3(2,M,1,\mu) = 1$
$B_3(2,M,r,-2) = M$
$B_3(2,r,\mu) = 1$
$B_3(2,M,r,2) = 1$
${}_3(2,M,r,-1) = 1$ for $r \geq 1$

---

* See Appendix Note #41

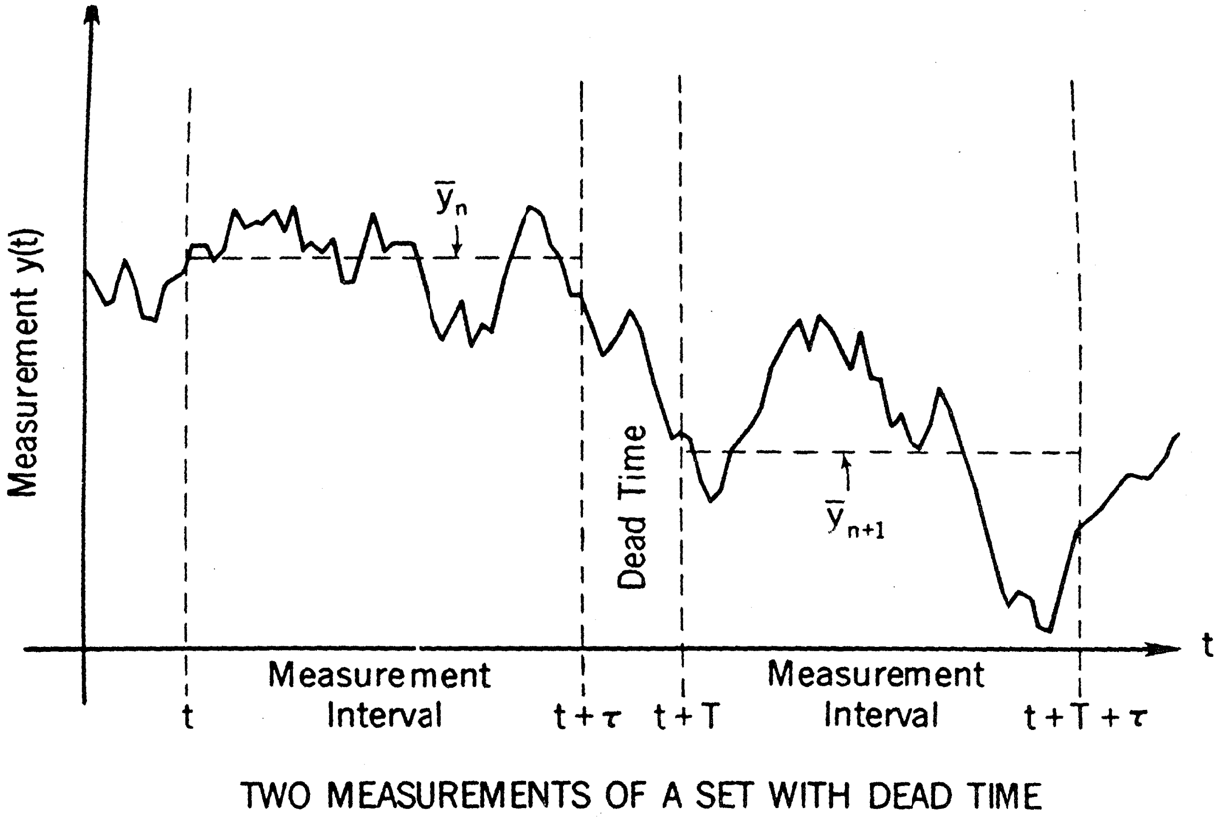


Figure 1. Illustration of two fractional frequency samples with dead time, $T-\tau$, between the samples. This is two of a set of adjacent frequency measurements, each averaged over an interval τ, needed to calculate a two-sample variance from a data set.

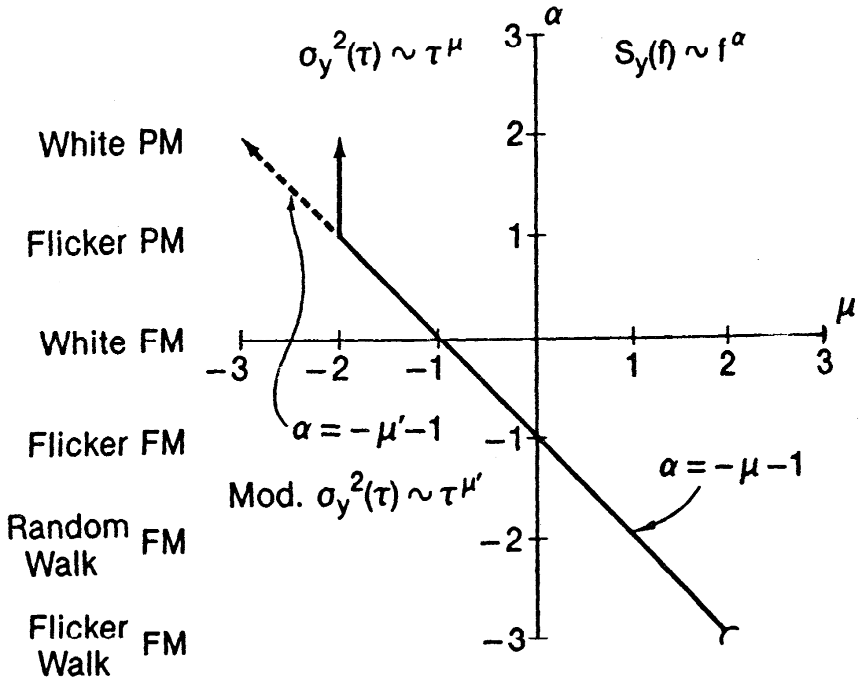


Figure 2. A plot of the relationship between the frequency-domain power-law spectral-density exponent α and the time-domain two-sample Allan variance exponent μ ($\alpha = -\mu-1$, $-2 \le \mu < 2$ and $\alpha \ge 1$ for $\mu = -2$). Also shown is the similar relationship between α and the modified Allan variance with exponent on τ of μ' ($\alpha = -\mu'-1$, $-4 \le \mu' < 2$). The pointing arrows indicate the mu-alpha relationship (α vs. μ or μ') for which the particular variance applies.

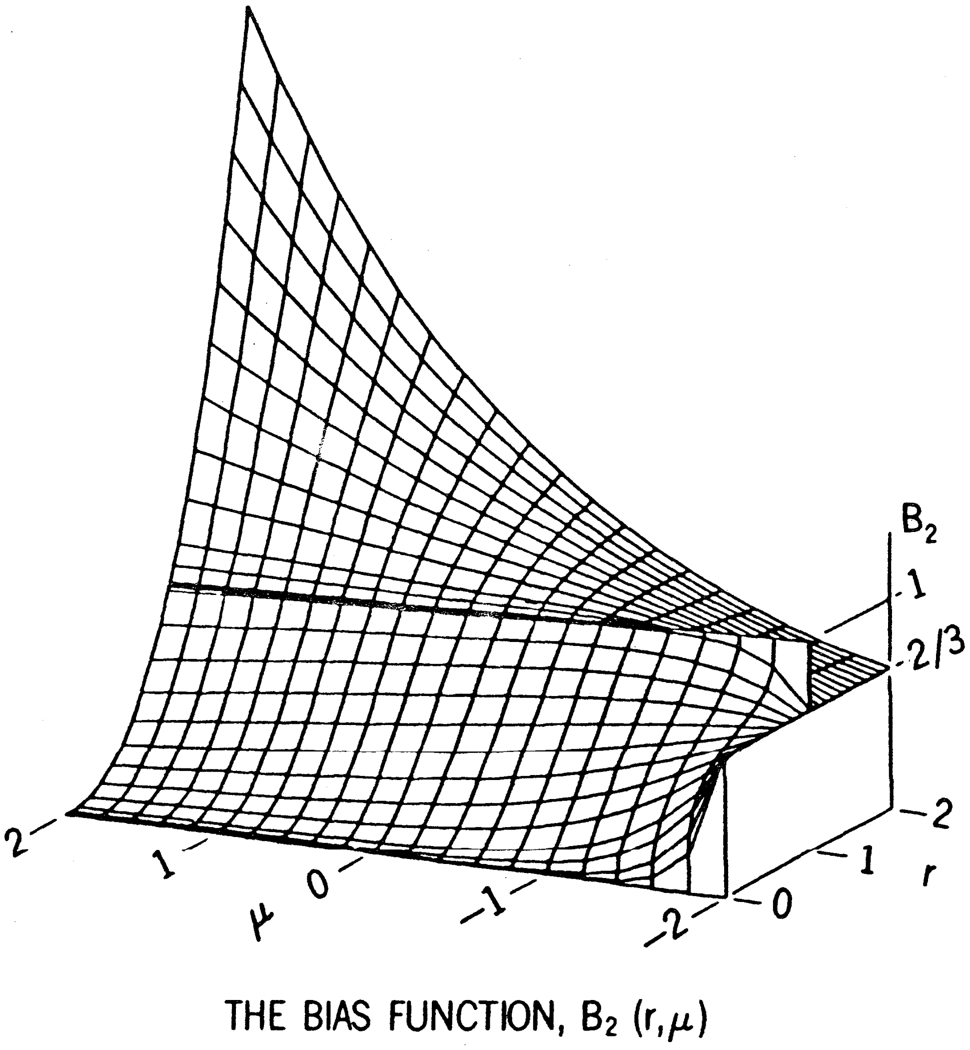


Figure 3. A three dimensional plot of the bias function $B_2(r,\mu)$, where $r = T/\tau$, and the dead time is $T - \tau$. The "fin" at $r = 1$ and $\mu = -2$ approaches zero width as the measurement bandwidth approaches infinity (see appendix ref. [3]).

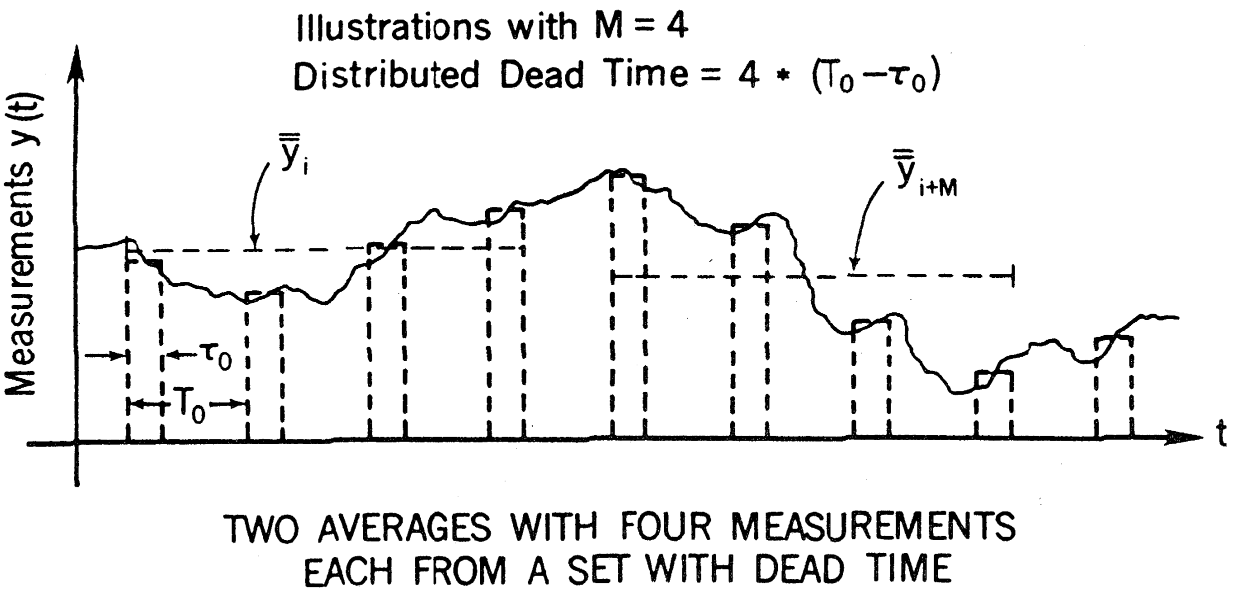


Figure 4. An illustration of determining two averages, each taken over four measurements. In this illustration the original data have a sample time τ_o and a dead time $T_o - \tau_o$. The two averages each have a distributed sample time $4\tau_o = \tau$, and a distributed dead time $4(T_o - \tau_o) = T - \tau$, since in this case $M = 4$. These two averages are two of a set similar averages with distributed sample times and dead times needed to calculate a two-sample distributed sample-time and dead-time variance -- the numerator for the bias function B_3. The denominator for B_3 is a two-sample variance with no distributed sample time or dead time but with $T = MT_o$ and $\tau = M\tau_o$ and dead time $T - \tau$.

Appendix

With reference to figure 1, the frequency sampling window has an equivalent phase sampling window. The intent is to evaluate the variance, S(M), of the sampled phase function in terms of the phase autocorrelation function, $R(\tau)$. The process here is to correctly account for terms and cross-terms coming from squaring and averaging the samples for each M. The $B_3(2,M,r,\mu)$ function can then be obtained from the relation,

$$B_3(2,M,r,\mu) = \frac{S(M)}{S(1)\cdot M^{\mu+2}},$$

for appropriate M, r, and μ. The denominator is just the two-sample variance with dead time for MT and $M\tau$ (in accordance with the definition of $B_3(2,M,r,\mu)$). The factors common to the numerator and denominator are ignored in the following.

For M = 1, the variance S(1) is just

$$S(1) = 4\cdot R(0) - 4\cdot R(\tau) - 4\cdot R(T) + 2\cdot R(T+\tau) + 2R(T-\tau),$$

where use has been made of the definition of the autocorrelation function,

$$R(T) = E[\phi(t) \cdot \phi(t+T)].$$

It is convenient to define a function G(T) as

$$G(T) = 2\cdot R(T) - R(T+\tau) - R(T-\tau).$$

Similarly, S(2) can now be written in the form,

$$S(2) = 8\cdot R(0) - 8\cdot R(\tau) + 2\cdot G(T) - 4\cdot G(2T) \quad 2\cdot G(3T).$$

Following this procedure, we can verify that the general S(M) is just

$$S(M) = 4\cdot M\cdot R(0) - 4\cdot M\cdot R(\tau) - 2\cdot M\cdot G(MT)$$

$$+ 2 \sum_{n=1}^{M-1} (M-n)[2\cdot G(nT) - G((M+n)T) - G((M-n)T)].$$

Following the work of Barnes and Allan [2,3], we can define the function $U(\tau)$ by the relation,

$$U(\tau) = 2\cdot R(0) - 2\cdot R(\tau),$$

and also define

$$F(nr) = G(nT)/U(\tau),$$

where $r = T/\tau$. The function $U(\tau)$ for power-law power spectral densities has the form,

$$U(\tau) = \frac{|\tau|^{\mu+2}}{4-2^{\mu+2}},$$

which yields

$$F(nr) = 2\cdot(nr)^{\mu+2} - (nr+1)^{\mu+2} - |nr-1|^{\mu+2}.$$

Finally, the working relation can be written as

$$B_3(2,M,r,\mu) = \frac{2\cdot M + M\cdot F(Mr) - \sum_{n=1}^{M-1} (M-n)\ [2\cdot F(nr) - F((M+n)r) - F((M-n)r)]}{[2 + F(r)]\ \cdot\ M^{\mu+2}}$$

B1(N,r,mu) for r = .01

Mu \ N=	4	8	16	32	64	128	256	512	1024	INF
-2	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	1.091E+00	1.210E+00	1.360E+00	1.545E+00	1.772E+00	2.089E+00	2.349E+00	2.456E+00	2.494E+00	2.512E+00
-1.6	1.199E+00	1.487E+00	1.893E+00	2.455E+00	3.239E+00	4.440E+00	5.505E+00	6.002E+00	6.198E+00	6.309E+00
-1.4	1.328E+00	1.856E+00	2.688E+00	3.991E+00	6.054E+00	9.500E+00	1.282E+01	1.456E+01	1.532E+01	1.585E+01
-1.2	1.482E+00	2.346E+00	3.880E+00	6.598E+00	1.145E+01	2.026E+01	2.947E+01	3.486E+01	3.754E+01	3.980E+01
-1	1.667E+00	3.000E+00	5.667E+00	1.100E+01	2.167E+01	4.255E+01	6.628E+01	8.190E+01	9.064E+01	1.000E+02
-.8	1.884E+00	3.860E+00	8.303E+00	1.828E+01	4.041E+01	8.691E+01	1.443E+02	1.868E+02	2.137E+02	2.519E+02
-.6	2.134E+00	4.959E+00	1.205E+01	2.977E+01	7.296E+01	1.697E+02	2.995E+02	4.076E+02	4.851E+02	6.423E+02
-.4	2.407E+00	6.279E+00	1.703E+01	4.655E+01	1.247E+02	3.106E+02	5.814E+02	8.357E+02	1.043E+03	1.715E+03
-.2	2.677E+00	7.714E+00	2.296E+01	6.836E+01	1.976E+02	5.220E+02	1.036E+03	1.580E+03	2.084E+03	5.581E+03
0	2.912E+00	9.075E+00	2.909E+01	9.282E+01	2.857E+02	7.951E+02	1.672E+03	2.719E+03	3.826E+03	
.2	3.089E+00	1.018E+01	3.448E+01	1.162E+02	3.763E+02	1.097E+03	2.446E+03	4.263E+03	6.453E+03	
.4	3.203E+00	1.095E+01	3.857E+01	1.354E+02	4.572E+02	1.390E+03	3.287E+03	6.172E+03	1.014E+04	
.6	3.268E+00	1.143E+01	4.133E+01	1.495E+02	5.221E+02	1.649E+03	4.142E+03	8.419E+03	1.514E+04	
.8	3.302E+00	1.170E+01	4.303E+01	1.590E+02	5.708E+02	1.869E+03	4.991E+03	1.104E+04	2.189E+04	
1	3.319E+00	1.185E+01	4.403E+01	1.653E+02	6.066E+02	2.055E+03	5.842E+03	1.412E+04	3.109E+04	
1.2	3.327E+00	1.192E+01	4.461E+01	1.693E+02	6.330E+02	2.215E+03	6.717E+03	1.782E+04	4.383E+04	
1.4	3.330E+00	1.196E+01	4.494E+01	1.720E+02	6.530E+02	2.359E+03	7.640E+03	2.234E+04	6.169E+04	
1.6	3.332E+00	1.198E+01	4.514E+01	1.738E+02	6.688E+02	2.493E+03	8.637E+03	2.793E+04	8.697E+04	
1.8	3.333E+00	1.199E+01	4.526E+01	1.750E+02	6.819E+02	2.623E+03	9.737E+03	3.493E+04	1.231E+05	
2	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = .03

Mu \ N=	4	8	16	32	64	128	256	512	1024	INF
-2	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	1.091E+00	1.211E+00	1.366E+00	1.573E+00	1.827E+00	1.950E+00	1.995E+00	2.010E+00	2.015E+00	2.016E+00
-1.6	1.199E+00	1.491E+00	1.910E+00	2.532E+00	3.341E+00	3.782E+00	3.963E+00	4.030E+00	4.053E+00	4.064E+00
-1.4	1.329E+00	1.863E+00	2.719E+00	4.133E+00	6.080E+00	7.268E+00	7.815E+00	8.044E+00	8.135E+00	8.191E+00
-1.2	1.484E+00	2.355E+00	3.918E+00	6.768E+00	1.093E+01	1.378E+01	1.525E+01	1.595E+01	1.626E+01	1.651E+01
-1	1.667E+00	3.000E+00	5.667E+00	1.100E+01	1.928E+01	2.560E+01	2.929E+01	3.127E+01	3.229E+01	3.333E+01
-.8	1.878E+00	3.825E+00	8.141E+00	1.753E+01	3.304E+01	4.626E+01	5.499E+01	6.029E+01	6.341E+01	6.770E+01
-.6	2.114E+00	4.838E+00	1.147E+01	2.706E+01	5.441E+01	8.049E+01	9.997E+01	1.134E+02	1.225E+02	1.403E+02
-.4	2.362E+00	6.005E+00	1.566E+01	3.993E+01	8.524E+01	1.335E+02	1.744E+02	2.066E+02	2.314E+02	3.099E+02
-.2	2.604E+00	7.242E+00	2.047E+01	5.581E+01	1.259E+02	2.094E+02	2.897E+02	3.619E+02	4.256E+02	8.571E+02
0	2.819E+00	8.430E+00	2.546E+01	7.348E+01	1.748E+02	3.096E+02	4.568E+02	6.082E+02	7.611E+02	
.2	2.992E+00	9.460E+00	3.013E+01	9.127E+01	2.286E+02	4.324E+02	6.856E+02	9.834E+02	1.328E+03	
.4	3.118E+00	1.027E+01	3.412E+01	1.077E+02	2.835E+02	5.750E+02	9.868E+02	1.541E+03	2.277E+03	
.6	3.203E+00	1.087E+01	3.728E+01	1.220E+02	3.372E+02	7.357E+02	1.376E+03	2.364E+03	3.868E+03	
.8	3.257E+00	1.128E+01	3.966E+01	1.339E+02	3.887E+02	9.152E+02	1.879E+03	3.580E+03	6.554E+03	
1	3.290E+00	1.155E+01	4.141E+01	1.437E+02	4.382E+02	1.117E+03	2.533E+03	5.395E+03	1.114E+04	
1.2	3.309E+00	1.172E+01	4.267E+01	1.518E+02	4.867E+02	1.348E+03	3.392E+03	8.128E+03	1.903E+04	
1.4	3.320E+00	1.184E+01	4.360E+01	1.588E+02	5.352E+02	1.615E+03	4.535E+03	1.228E+04	3.276E+04	
1.6	3.327E+00	1.191E+01	4.431E+01	1.650E+02	5.851E+02	1.930E+03	6.068E+03	1.866E+04	5.684E+04	
1.8	3.331E+00	1.196E+01	4.487E+01	1.706E+02	6.374E+02	2.304E+03	8.142E+03	2.849E+04	9.936E+04	
2	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = .1

Mu \	N=	4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	1.093E+00	1.226E+00	1.438E+00	1.547E+00	1.579E+00	1.586E+00	1.586E+00	1.585E+00	1.584E+00	1.583E+00
-1.6	:	1.205E+00	1.522E+00	2.021E+00	2.329E+00	2.449E+00	2.490E+00	2.502E+00	2.505E+00	2.505E+00	2.504E+00
-1.4	:	1.337E+00	1.907E+00	2.832E+00	3.477E+00	3.772E+00	3.892E+00	3.938E+00	3.954E+00	3.960E+00	3.962E+00
-1.2	:	1.491E+00	2.396E+00	3.931E+00	5.125E+00	5.751E+00	6.046E+00	6.177E+00	6.235E+00	6.260E+00	6.278E+00
-1	:	1.667E+00	3.000E+00	5.375E+00	7.429E+00	8.653E+00	9.312E+00	9.652E+00	9.825E+00	9.912E+00	1.000E+01
-.8	:	1.860E+00	3.720E+00	7.204E+00	1.055E+01	1.281E+01	1.419E+01	1.502E+01	1.550E+01	1.577E+01	1.616E+01
-.6	:	2.065E+00	4.540E+00	9.423E+00	1.461E+01	1.859E+01	2.137E+01	2.326E+01	2.452E+01	2.537E+01	2.702E+01
-.4	:	2.273E+00	5.430E+00	1.199E+01	1.971E+01	2.642E+01	3.177E+01	3.592E+01	3.910E+01	4.153E+01	4.922E+01
-.2	:	2.472E+00	6.344E+00	1.481E+01	2.585E+01	3.673E+01	4.664E+01	5.542E+01	6.314E+01	6.989E+01	1.156E+02
0	:	2.653E+00	7.236E+00	1.779E+01	3.300E+01	5.003E+01	6.771E+01	8.566E+01	1.037E+02	1.219E+02	
.2	:	2.810E+00	8.065E+00	2.080E+01	4.111E+01	6.690E+01	9.750E+01	1.331E+02	1.741E+02	2.214E+02	
.4	:	2.940E+00	8.804E+00	2.376E+01	5.015E+01	8.816E+01	1.397E+02	2.084E+02	2.993E+02	4.194E+02	
.6	:	3.043E+00	9.445E+00	2.663E+01	6.014E+01	1.149E+02	1.999E+02	3.295E+02	5.265E+02	8.254E+02	
.8	:	3.123E+00	9.989E+00	2.938E+01	7.118E+01	1.487E+02	2.862E+02	5.269E+02	9.466E+02	1.678E+03	
1	:	3.184E+00	1.045E+01	3.204E+01	8.347E+01	1.918E+02	4.114E+02	8.523E+02	1.735E+03	3.500E+03	
1.2	:	3.230E+00	1.084E+01	3.463E+01	9.728E+01	2.470E+02	5.943E+02	1.394E+03	3.232E+03	7.453E+03	
1.4	:	3.265E+00	1.118E+01	3.721E+01	1.130E+02	3.183E+02	8.636E+02	2.304E+03	6.103E+03	1.613E+04	
1.6	:	3.293E+00	1.147E+01	3.982E+01	1.309E+02	4.113E+02	1.263E+03	3.843E+03	1.166E+04	3.535E+04	
1.8	:	3.315E+00	1.174E+01	4.252E+01	1.517E+02	5.330E+02	1.858E+03	6.467E+03	2.250E+04	7.830E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = .3

Mu \	N=	4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	1.124E+00	1.226E+00	1.259E+00	1.265E+00	1.265E+00	1.263E+00	1.262E+00	1.261E+00	1.261E+00	1.260E+00
-1.6	:	1.254E+00	1.479E+00	1.564E+00	1.589E+00	1.594E+00	1.594E+00	1.593E+00	1.591E+00	1.591E+00	1.590E+00
-1.4	:	1.388E+00	1.759E+00	1.925E+00	1.988E+00	2.009E+00	2.014E+00	2.015E+00	2.015E+00	2.014E+00	2.013E+00
-1.2	:	1.527E+00	2.068E+00	2.351E+00	2.479E+00	2.534E+00	2.556E+00	2.565E+00	2.569E+00	2.570E+00	2.571E+00
-1	:	1.667E+00	2.405E+00	2.850E+00	3.087E+00	3.209E+00	3.271E+00	3.302E+00	3.318E+00	3.326E+00	3.333E+00
-.8	:	1.806E+00	2.769E+00	3.435E+00	3.847E+00	4.093E+00	4.239E+00	4.325E+00	4.376E+00	4.405E+00	4.447E+00
-.6	:	1.943E+00	3.160E+00	4.122E+00	4.804E+00	5.275E+00	5.595E+00	5.812E+00	5.958E+00	6.056E+00	6.252E+00
-.4	:	2.076E+00	3.577E+00	4.927E+00	6.024E+00	6.889E+00	7.564E+00	8.086E+00	8.487E+00	8.795E+00	9.776E+00
-.2	:	2.203E+00	4.021E+00	5.877E+00	7.597E+00	9.147E+00	1.053E+01	1.175E+01	1.282E+01	1.377E+01	2.017E+01
0	:	2.325E+00	4.492E+00	7.000E+00	9.651E+00	1.238E+01	1.515E+01	1.796E+01	2.079E+01	2.363E+01	
.2	:	2.440E+00	4.994E+00	8.338E+00	1.236E+01	1.710E+01	2.260E+01	2.897E+01	3.632E+01	4.479E+01	
.4	:	2.550E+00	5.530E+00	9.940E+00	1.599E+01	2.411E+01	3.493E+01	4.926E+01	6.823E+01	9.330E+01	
.6	:	2.654E+00	6.106E+00	1.187E+01	2.089E+01	3.472E+01	5.581E+01	8.786E+01	1.365E+02	2.103E+02	
.8	:	2.754E+00	6.729E+00	1.422E+01	2.756E+01	5.098E+01	9.187E+01	1.632E+02	2.874E+02	5.037E+02	
1	:	2.852E+00	7.406E+00	1.708E+01	3.673E+01	7.618E+01	1.552E+02	3.132E+02	6.292E+02	1.261E+03	
1.2	:	2.947E+00	8.147E+00	2.059E+01	4.942E+01	1.157E+02	2.678E+02	6.170E+02	1.419E+03	3.261E+03	
1.4	:	3.042E+00	8.963E+00	2.493E+01	6.709E+01	1.781E+02	4.703E+02	1.241E+03	3.272E+03	8.631E+03	
1.6	:	3.137E+00	9.868E+00	3.030E+01	9.185E+01	2.774E+02	8.381E+02	2.535E+03	7.673E+03	2.324E+04	
1.8	:	3.234E+00	1.087E+01	3.699E+01	1.267E+02	4.366E+02	1.511E+03	5.245E+03	1.823E+04	6.343E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 1

Mu \	N=	4	8	16	32	64	128	256	512	1024	INF
-2	:	8.333E-01	7.500E-01	7.083E-01	6.875E-01	6.771E-01	6.719E-01	6.693E-01	6.680E-01	6.673E-01	6.667E-01
-1.8	:	8.581E-01	7.827E-01	7.431E-01	7.226E-01	7.122E-01	7.068E-01	7.042E-01	7.028E-01	7.021E-01	7.014E-01
-1.6	:	8.866E-01	8.221E-01	7.864E-01	7.672E-01	7.570E-01	7.517E-01	7.490E-01	7.476E-01	7.468E-01	7.461E-01
-1.4	:	9.193E-01	8.700E-01	8.410E-01	8.245E-01	8.154E-01	8.105E-01	8.079E-01	8.065E-01	8.058E-01	8.051E-01
-1.2	:	9.569E-01	9.284E-01	9.105E-01	8.997E-01	8.933E-01	8.897E-01	8.877E-01	8.866E-01	8.860E-01	8.854E-01
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.050E+00	1.088E+00	1.117E+00	1.137E+00	1.150E+00	1.160E+00	1.165E+00	1.169E+00	1.171E+00	1.175E+00
-.6	:	1.107E+00	1.197E+00	1.271E+00	1.327E+00	1.370E+00	1.401E+00	1.422E+00	1.438E+00	1.448E+00	1.470E+00
-.4	:	1.172E+00	1.333E+00	1.476E+00	1.599E+00	1.700E+00	1.782E+00	1.847E+00	1.898E+00	1.938E+00	2.065E+00
-.2	:	1.247E+00	1.502E+00	1.754E+00	1.994E+00	2.216E+00	2.418E+00	2.599E+00	2.759E+00	2.900E+00	3.863E+00
0	:	1.333E+00	1.714E+00	2.133E+00	2.581E+00	3.048E+00	3.528E+00	4.016E+00	4.509E+00	5.005E+00	
.2	:	1.432E+00	1.982E+00	2.658E+00	3.471E+00	4.432E+00	5.555E+00	6.858E+00	8.363E+00	1.010E+01	
.4	:	1.546E+00	2.320E+00	3.391E+00	4.846E+00	6.801E+00	9.407E+00	1.287E+01	1.744E+01	2.350E+01	
.6	:	1.677E+00	2.750E+00	4.424E+00	7.006E+00	1.096E+01	1.698E+01	2.614E+01	4.005E+01	6.114E+01	
.8	:	1.827E+00	3.299E+00	5.894E+00	1.045E+01	1.841E+01	3.230E+01	5.652E+01	9.872E+01	1.722E+02	
1	:	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	
1.2	:	2.198E+00	4.900E+00	1.104E+01	2.506E+01	5.717E+01	1.308E+02	2.999E+02	6.881E+02	1.580E+03	
1.4	:	2.426E+00	6.059E+00	1.546E+01	3.999E+01	1.044E+02	2.738E+02	7.202E+02	1.897E+03	5.003E+03	
1.6	:	2.688E+00	7.555E+00	2.191E+01	6.479E+01	1.938E+02	5.833E+02	1.762E+03	5.331E+03	1.615E+04	
1.8	:	2.988E+00	9.490E+00	3.138E+01	1.063E+02	3.646E+02	1.260E+03	4.372E+03	1.519E+04	5.286E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 1.01

Mu \	N=	4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	9.156E-01	8.682E-01	8.425E-01	8.288E-01	8.217E-01	8.181E-01	8.162E-01	8.153E-01	8.148E-01	8.143E-01
-1.6	:	9.098E-01	8.566E-01	8.264E-01	8.098E-01	8.009E-01	7.963E-01	7.938E-01	7.926E-01	7.920E-01	7.913E-01
-1.4	:	9.286E-01	8.840E-01	8.573E-01	8.419E-01	8.333E-01	8.287E-01	8.262E-01	8.249E-01	8.242E-01	8.235E-01
-1.2	:	9.599E-01	9.331E-01	9.160E-01	9.056E-01	8.995E-01	8.960E-01	8.940E-01	8.929E-01	8.924E-01	8.917E-01
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.048E+00	1.085E+00	1.113E+00	1.133E+00	1.147E+00	1.155E+00	1.161E+00	1.165E+00	1.167E+00	1.170E+00
-.6	:	1.104E+00	1.193E+00	1.265E+00	1.321E+00	1.362E+00	1.393E+00	1.414E+00	1.429E+00	1.440E+00	1.461E+00
-.4	:	1.168E+00	1.327E+00	1.468E+00	1.589E+00	1.689E+00	1.770E+00	1.835E+00	1.885E+00	1.924E+00	2.050E+00
-.2	:	1.243E+00	1.495E+00	1.743E+00	1.980E+00	2.200E+00	2.400E+00	2.579E+00	2.737E+00	2.877E+00	3.829E+00
0	:	1.329E+00	1.706E+00	2.120E+00	2.563E+00	3.025E+00	3.500E+00	3.984E+00	4.472E+00	4.963E+00	
.2	:	1.428E+00	1.972E+00	2.642E+00	3.447E+00	4.400E+00	5.512E+00	6.804E+00	8.296E+00	1.002E+01	
.4	:	1.541E+00	2.309E+00	3.371E+00	4.814E+00	6.754E+00	9.340E+00	1.277E+01	1.731E+01	2.332E+01	
.6	:	1.672E+00	2.738E+00	4.400E+00	6.963E+00	1.089E+01	1.687E+01	2.597E+01	3.978E+01	6.073E+01	
.8	:	1.822E+00	3.285E+00	5.864E+00	1.039E+01	1.830E+01	3.212E+01	5.619E+01	9.814E+01	1.712E+02	
1	:	1.995E+00	3.985E+00	7.966E+00	1.593E+01	3.185E+01	6.369E+01	1.274E+02	2.547E+02	5.095E+02	
1.2	:	2.194E+00	4.885E+00	1.100E+01	2.497E+01	5.695E+01	1.303E+02	2.987E+02	6.854E+02	1.573E+03	
1.4	:	2.422E+00	6.045E+00	1.542E+01	3.988E+01	1.041E+02	2.730E+02	7.180E+02	1.892E+03	4.988E+03	
1.6	:	2.685E+00	7.542E+00	2.187E+01	6.466E+01	1.934E+02	5.822E+02	1.758E+03	5.321E+03	1.611E+04	
1.8	:	2.986E+00	9.482E+00	3.135E+01	1.062E+02	3.643E+02	1.259E+03	4.367E+03	1.518E+04	5.280E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 1.1

Mu \ N=	4	8	16	32	64	128	256	512	1024	INF
-2	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	9.527E-01	9.243E-01	9.082E-01	8.994E-01	8.948E-01	8.924E-01	8.911E-01	8.905E-01	8.902E-01	8.898E-01
-1.6	9.415E-01	9.048E-01	8.831E-01	8.709E-01	8.642E-01	8.607E-01	8.588E-01	8.578E-01	8.574E-01	8.569E-01
-1.4	9.493E-01	9.157E-01	8.948E-01	8.825E-01	8.755E-01	8.717E-01	8.696E-01	8.685E-01	8.680E-01	8.674E-01
-1.2	9.696E-01	9.482E-01	9.341E-01	9.254E-01	9.201E-01	9.171E-01	9.154E-01	9.145E-01	9.140E-01	9.134E-01
-1	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	1.040E+00	1.072E+00	1.096E+00	1.114E+00	1.126E+00	1.134E+00	1.139E+00	1.142E+00	1.144E+00	1.147E+00
-.6	1.088E+00	1.166E+00	1.230E+00	1.281E+00	1.319E+00	1.346E+00	1.366E+00	1.379E+00	1.389E+00	1.408E+00
-.4	1.147E+00	1.288E+00	1.416E+00	1.526E+00	1.617E+00	1.692E+00	1.751E+00	1.797E+00	1.832E+00	1.948E+00
-.2	1.217E+00	1.444E+00	1.671E+00	1.889E+00	2.091E+00	2.275E+00	2.440E+00	2.586E+00	2.714E+00	3.593E+00
0	1.298E+00	1.643E+00	2.026E+00	2.435E+00	2.863E+00	3.304E+00	3.752E+00	4.205E+00	4.661E+00	
.2	1.393E+00	1.898E+00	2.521E+00	3.272E+00	4.160E+00	5.199E+00	6.404E+00	7.797E+00	9.402E+00	
.4	1.504E+00	2.223E+00	3.219E+00	4.575E+00	6.398E+00	8.828E+00	1.205E+01	1.632E+01	2.197E+01	
.6	1.633E+00	2.640E+00	4.213E+00	6.639E+00	1.035E+01	1.602E+01	2.463E+01	3.771E+01	5.754E+01	
.8	1.783E+00	3.177E+00	5.637E+00	9.954E+00	1.750E+01	3.068E+01	5.365E+01	9.366E+01	1.634E+02	
1	1.957E+00	3.870E+00	7.696E+00	1.535E+01	3.065E+01	6.126E+01	1.225E+02	2.449E+02	4.898E+02	
1.2	2.158E+00	4.766E+00	1.069E+01	2.422E+01	5.521E+01	1.263E+02	2.894E+02	6.640E+02	1.524E+03	
1.4	2.391E+00	5.929E+00	1.508E+01	3.897E+01	1.016E+02	2.666E+02	7.011E+02	1.847E+03	4.870E+03	
1.6	2.660E+00	7.443E+00	2.154E+01	6.366E+01	1.904E+02	5.730E+02	1.731E+03	5.237E+03	1.586E+04	
1.8	2.972E+00	9.418E+00	3.111E+01	1.053E+02	3.614E+02	1.249E+03	4.333E+03	1.506E+04	5.238E+04	
2	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 2

Mu \ N=	4	8	16	32	64	128	256	512	1024	INF
-2	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	9.901E-01	9.836E-01	9.796E-01	9.774E-01	9.761E-01	9.755E-01	9.752E-01	9.750E-01	9.749E-01	9.748E-01
-1.6	9.845E-01	9.737E-01	9.669E-01	9.630E-01	9.607E-01	9.595E-01	9.589E-01	9.586E-01	9.584E-01	9.582E-01
-1.4	9.837E-01	9.719E-01	9.641E-01	9.593E-01	9.565E-01	9.550E-01	9.541E-01	9.537E-01	9.534E-01	9.532E-01
-1.2	9.886E-01	9.799E-01	9.738E-01	9.699E-01	9.675E-01	9.661E-01	9.653E-01	9.648E-01	9.646E-01	9.643E-01
-1	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	1.019E+00	1.035E+00	1.048E+00	1.058E+00	1.064E+00	1.069E+00	1.072E+00	1.074E+00	1.075E+00	1.077E+00
-.6	1.046E+00	1.090E+00	1.126E+00	1.156E+00	1.178E+00	1.195E+00	1.207E+00	1.215E+00	1.221E+00	1.233E+00
-.4	1.084E+00	1.168E+00	1.246E+00	1.315E+00	1.373E+00	1.420E+00	1.457E+00	1.487E+00	1.509E+00	1.583E+00
-.2	1.133E+00	1.277E+00	1.425E+00	1.568E+00	1.702E+00	1.824E+00	1.934E+00	2.031E+00	2.117E+00	2.704E+00
0	1.195E+00	1.427E+00	1.688E+00	1.971E+00	2.267E+00	2.573E+00	2.884E+00	3.198E+00	3.515E+00	
.2	1.273E+00	1.629E+00	2.075E+00	2.615E+00	3.256E+00	4.005E+00	4.876E+00	5.882E+00	7.042E+00	
.4	1.369E+00	1.901E+00	2.644E+00	3.659E+00	5.025E+00	6.847E+00	9.267E+00	1.247E+01	1.670E+01	
.6	1.486E+00	2.264E+00	3.485E+00	5.371E+00	8.262E+00	1.267E+01	1.937E+01	2.955E+01	4.498E+01	
.8	1.628E+00	2.750E+00	4.733E+00	8.215E+00	1.431E+01	2.494E+01	4.347E+01	7.576E+01	1.320E+02	
1	1.800E+00	3.400E+00	6.600E+00	1.300E+01	2.580E+01	5.140E+01	1.026E+02	2.050E+02	4.098E+02	
1.2	2.007E+00	4.271E+00	9.409E+00	2.114E+01	4.800E+01	1.096E+02	2.510E+02	5.756E+02	1.321E+03	
1.4	2.255E+00	5.439E+00	1.366E+01	3.512E+01	9.142E+01	2.395E+02	6.299E+02	1.659E+03	4.374E+03	
1.6	2.552E+00	7.012E+00	2.014E+01	5.935E+01	1.773E+02	5.334E+02	1.611E+03	4.874E+03	1.476E+04	
1.8	2.908E+00	9.132E+00	3.006E+01	1.017E+02	3.487E+02	1.205E+03	4.179E+03	1.453E+04	5.053E+04	
2	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 4

Mu \ N=		4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	9.974E-01	9.957E-01	9.946E-01	9.940E-01	9.936E-01	9.935E-01	9.934E-01	9.933E-01	9.933E-01	9.933E-01
-1.6	:	9.952E-01	9.918E-01	9.896E-01	9.884E-01	9.876E-01	9.873E-01	9.870E-01	9.869E-01	9.869E-01	9.868E-01
-1.4	:	9.941E-01	9.898E-01	9.869E-01	9.851E-01	9.840E-01	9.834E-01	9.831E-01	9.829E-01	9.829E-01	9.828E-01
-1.2	:	9.953E-01	9.916E-01	9.890E-01	9.873E-01	9.862E-01	9.856E-01	9.853E-01	9.851E-01	9.850E-01	9.849E-01
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.010E+00	1.019E+00	1.026E+00	1.031E+00	1.035E+00	1.037E+00	1.039E+00	1.040E+00	1.041E+00	1.042E+00
-.6	:	1.027E+00	1.053E+00	1.075E+00	1.093E+00	1.107E+00	1.117E+00	1.124E+00	1.129E+00	1.132E+00	1.140E+00
-.4	:	1.054E+00	1.109E+00	1.160E+00	1.205E+00	1.243E+00	1.274E+00	1.299E+00	1.318E+00	1.333E+00	1.382E+00
-.2	:	1.092E+00	1.194E+00	1.298E+00	1.399E+00	1.494E+00	1.580E+00	1.658E+00	1.727E+00	1.788E+00	2.204E+00
0	:	1.144E+00	1.318E+00	1.514E+00	1.726E+00	1.949E+00	2.179E+00	2.414E+00	2.651E+00	2.890E+00	
.2	:	1.214E+00	1.495E+00	1.848E+00	2.275E+00	2.783E+00	3.377E+00	4.067E+00	4.865E+00	5.784E+00	
.4	:	1.303E+00	1.742E+00	2.356E+00	3.196E+00	4.326E+00	5.834E+00	7.837E+00	1.049E+01	1.399E+01	
.6	:	1.415E+00	2.082E+00	3.129E+00	4.748E+00	7.229E+00	1.101E+01	1.676E+01	2.550E+01	3.875E+01	
.8	:	1.555E+00	2.548E+00	4.303E+00	7.385E+00	1.278E+01	2.219E+01	3.860E+01	6.718E+01	1.170E+02	
1	:	1.727E+00	3.182E+00	6.091E+00	1.191E+01	2.355E+01	4.682E+01	9.336E+01	1.865E+02	3.726E+02	
1.2	:	1.938E+00	4.045E+00	8.826E+00	1.974E+01	4.472E+01	1.020E+02	2.336E+02	5.356E+02	1.229E+03	
1.4	:	2.194E+00	5.219E+00	1.303E+01	3.340E+01	8.686E+01	2.275E+02	5.981E+02	1.575E+03	4.154E+03	
1.6	:	2.504E+00	6.819E+00	1.952E+01	5.745E+01	1.715E+02	5.160E+02	1.558E+03	4.714E+03	1.428E+04	
1.8	:	2.879E+00	9.006E+00	2.960E+01	1.000E+02	3.431E+02	1.186E+03	4.112E+03	1.429E+04	4.972E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 8

Mu \ N=		4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	9.993E-01	9.988E-01	9.985E-01	9.983E-01	9.982E-01	9.981E-01	9.981E-01	9.981E-01	9.981E-01	9.981E-01
-1.6	:	9.984E-01	9.973E-01	9.966E-01	9.962E-01	9.960E-01	9.958E-01	9.958E-01	9.957E-01	9.957E-01	9.957E-01
-1.4	:	9.978E-01	9.961E-01	9.950E-01	9.944E-01	9.940E-01	9.937E-01	9.936E-01	9.936E-01	9.935E-01	9.935E-01
-1.2	:	9.980E-01	9.963E-01	9.952E-01	9.945E-01	9.940E-01	9.938E-01	9.936E-01	9.935E-01	9.935E-01	9.934E-01
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.006E+00	1.011E+00	1.014E+00	1.017E+00	1.020E+00	1.021E+00	1.022E+00	1.022E+00	1.023E+00	1.023E+00
-.6	:	1.017E+00	1.033E+00	1.047E+00	1.058E+00	1.067E+00	1.073E+00	1.078E+00	1.081E+00	1.083E+00	1.088E+00
-.4	:	1.037E+00	1.075E+00	1.110E+00	1.142E+00	1.168E+00	1.189E+00	1.207E+00	1.220E+00	1.231E+00	1.264E+00
-.2	:	1.069E+00	1.145E+00	1.223E+00	1.299E+00	1.371E+00	1.436E+00	1.494E+00	1.546E+00	1.592E+00	1.905E+00
0	:	1.116E+00	1.255E+00	1.413E+00	1.584E+00	1.763E+00	1.949E+00	2.137E+00	2.328E+00	2.520E+00	
.2	:	1.181E+00	1.420E+00	1.720E+00	2.083E+00	2.514E+00	3.019E+00	3.606E+00	4.284E+00	5.066E+00	
.4	:	1.268E+00	1.657E+00	2.202E+00	2.946E+00	3.949E+00	5.286E+00	7.062E+00	9.414E+00	1.252E+01	
.6	:	1.380E+00	1.991E+00	2.950E+00	4.433E+00	6.706E+00	1.017E+01	1.544E+01	2.345E+01	3.559E+01	
.8	:	1.521E+00	2.453E+00	4.101E+00	6.996E+00	1.206E+01	2.090E+01	3.630E+01	6.315E+01	1.099E+02	
1	:	1.696E+00	3.087E+00	5.870E+00	1.143E+01	2.257E+01	4.483E+01	8.935E+01	1.784E+02	3.565E+02	
1.2	:	1.910E+00	3.953E+00	8.590E+00	1.917E+01	4.341E+01	9.898E+01	2.265E+02	5.195E+02	1.192E+03	
1.4	:	2.170E+00	5.135E+00	1.279E+01	3.276E+01	8.515E+01	2.230E+02	5.862E+02	1.544E+03	4.071E+03	
1.6	:	2.486E+00	6.751E+00	1.930E+01	5.678E+01	1.695E+02	5.099E+02	1.540E+03	4.658E+03	1.411E+04	
1.8	:	2.870E+00	8.963E+00	2.945E+01	9.951E+01	3.413E+02	1.179E+03	4.090E+03	1.421E+04	4.945E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 16

Mu \	N=	4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	9.998E-01	9.997E-01	9.996E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01
-1.6	:	9.995E-01	9.991E-01	9.989E-01	9.987E-01	9.987E-01	9.986E-01	9.986E-01	9.986E-01	9.986E-01	9.986E-01
-1.4	:	9.992E-01	9.985E-01	9.981E-01	9.979E-01	9.977E-01	9.976E-01	9.976E-01	9.976E-01	9.975E-01	9.975E-01
-1.2	:	9.991E-01	9.984E-01	9.979E-01	9.976E-01	9.974E-01	9.973E-01	9.972E-01	9.972E-01	9.972E-01	9.971E-01
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.003E+00	1.006E+00	1.008E+00	1.010E+00	1.011E+00	1.012E+00	1.012E+00	1.013E+00	1.013E+00	1.013E+00
-.6	:	1.011E+00	1.021E+00	1.030E+00	1.037E+00	1.043E+00	1.047E+00	1.050E+00	1.052E+00	1.053E+00	1.056E+00
-.4	:	1.026E+00	1.053E+00	1.078E+00	1.101E+00	1.119E+00	1.135E+00	1.147E+00	1.157E+00	1.164E+00	1.188E+00
-.2	:	1.054E+00	1.113E+00	1.174E+00	1.233E+00	1.289E+00	1.339E+00	1.385E+00	1.425E+00	1.461E+00	1.705E+00
0	:	1.097E+00	1.214E+00	1.346E+00	1.489E+00	1.639E+00	1.794E+00	1.952E+00	2.112E+00	2.273E+00	
.2	:	1.160E+00	1.372E+00	1.637E+00	1.958E+00	2.340E+00	2.787E+00	3.307E+00	3.907E+00	4.599E+00	
.4	:	1.247E+00	1.606E+00	2.108E+00	2.794E+00	3.717E+00	4.950E+00	6.587E+00	8.754E+00	1.162E+01	
.6	:	1.360E+00	1.939E+00	2.848E+00	4.254E+00	6.409E+00	9.696E+00	1.469E+01	2.228E+01	3.379E+01	
.8	:	1.504E+00	2.404E+00	3.997E+00	6.794E+00	1.169E+01	2.023E+01	3.512E+01	6.106E+01	1.062E+02	
1	:	1.681E+00	3.043E+00	5.766E+00	1.121E+01	2.211E+01	4.389E+01	8.747E+01	1.746E+02	3.489E+02	
1.2	:	1.890E+00	3.914E+00	8.491E+00	1.894E+01	4.285E+01	9.769E+01	2.236E+02	5.127E+02	1.177E+03	
1.4	:	2.161E+00	5.104E+00	1.270E+01	3.251E+01	8.450E+01	2.213E+02	5.817E+02	1.532E+03	4.039E+03	
1.6	:	2.480E+00	6.727E+00	1.923E+01	5.654E+01	1.688E+02	5.077E+02	1.533E+03	4.639E+03	1.405E+04	
1.8	:	2.866E+00	8.949E+00	2.940E+01	9.934E+01	3.406E+02	1.177E+03	4.083E+03	1.419E+04	4.936E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 32

Mu \	N=	4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.998E-01	9.998E-01	9.998E-01	9.998E-01	9.998E-01
-1.6	:	9.998E-01	9.997E-01	9.996E-01	9.996E-01	9.996E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01
-1.4	:	9.997E-01	9.994E-01	9.993E-01	9.992E-01	9.991E-01	9.991E-01	9.991E-01	9.991E-01	9.991E-01	9.991E-01
-1.2	:	9.996E-01	9.993E-01	9.991E-01	9.990E-01	9.989E-01	9.988E-01	9.988E-01	9.988E-01	9.988E-01	9.988E-01
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.002E+00	1.003E+00	1.005E+00	1.006E+00	1.006E+00	1.007E+00	1.007E+00	1.007E+00	1.007E+00	1.008E+00
-.6	:	1.007E+00	1.014E+00	1.019E+00	1.024E+00	1.028E+00	1.030E+00	1.032E+00	1.033E+00	1.034E+00	1.036E+00
-.4	:	1.019E+00	1.039E+00	1.057E+00	1.073E+00	1.087E+00	1.098E+00	1.107E+00	1.114E+00	1.119E+00	1.136E+00
-.2	:	1.043E+00	1.090E+00	1.139E+00	1.186E+00	1.230E+00	1.271E+00	1.307E+00	1.339E+00	1.368E+00	1.563E+00
0	:	1.083E+00	1.184E+00	1.297E+00	1.420E+00	1.550E+00	1.683E+00	1.819E+00	1.957E+00	2.095E+00	
.2	:	1.146E+00	1.338E+00	1.579E+00	1.871E+00	2.218E+00	2.625E+00	3.097E+00	3.643E+00	4.272E+00	
.4	:	1.233E+00	1.572E+00	2.046E+00	2.693E+00	3.565E+00	4.729E+00	6.275E+00	8.320E+00	1.102E+01	
.6	:	1.348E+00	1.908E+00	2.787E+00	4.147E+00	6.231E+00	9.408E+00	1.424E+01	2.158E+01	3.270E+01	
.8	:	1.494E+00	2.378E+00	3.940E+00	6.685E+00	1.149E+01	1.986E+01	3.447E+01	5.992E+01	1.042E+02	
1	:	1.674E+00	3.021E+00	5.716E+00	1.111E+01	2.188E+01	4.344E+01	8.656E+01	1.728E+02	3.453E+02	
1.2	:	1.893E+00	3.898E+00	8.448E+00	1.883E+01	4.261E+01	9.714E+01	2.223E+02	5.097E+02	1.170E+03	
1.4	:	2.158E+00	5.091E+00	1.266E+01	3.242E+01	8.425E+01	2.206E+02	5.799E+02	1.527E+03	4.027E+03	
1.6	:	2.478E+00	6.719E+00	1.920E+01	5.646E+01	1.685E+02	5.070E+02	1.531E+03	4.632E+03	1.403E+04	
1.8	:	2.865E+00	8.945E+00	2.938E+01	9.928E+01	3.405E+02	1.176E+03	4.081E+03	1.418E+04	4.934E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 64

Mu \ N=		4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.6	:	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.998E-01	9.998E-01	9.998E-01	9.998E-01
-1.4	:	9.999E-01	9.998E-01	9.997E-01	9.997E-01	9.997E-01	9.997E-01	9.997E-01	9.996E-01	9.996E-01	9.996E-01
-1.2	:	9.998E-01	9.997E-01	9.996E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01	9.995E-01
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.001E+00	1.002E+00	1.003E+00	1.003E+00	1.004E+00	1.004E+00	1.004E+00	1.004E+00	1.004E+00	1.004E+00
-.6	:	1.005E+00	1.009E+00	1.013E+00	1.016E+00	1.018E+00	1.020E+00	1.021E+00	1.022E+00	1.022E+00	1.024E+00
-.4	:	1.014E+00	1.028E+00	1.042E+00	1.054E+00	1.064E+00	1.072E+00	1.078E+00	1.083E+00	1.087E+00	1.100E+00
-.2	:	1.035E+00	1.073E+00	1.112E+00	1.151E+00	1.187E+00	1.220E+00	1.249E+00	1.275E+00	1.298E+00	1.456E+00
0	:	1.073E+00	1.161E+00	1.261E+00	1.369E+00	1.482E+00	1.600E+00	1.719E+00	1.840E+00	1.961E+00	
.2	:	1.135E+00	1.313E+00	1.536E+00	1.808E+00	2.129E+00	2.506E+00	2.944E+00	3.450E+00	4.033E+00	
.4	:	1.224E+00	1.548E+00	2.003E+00	2.624E+00	3.461E+00	4.578E+00	6.060E+00	8.023E+00	1.062E+01	
.6	:	1.341E+00	1.889E+00	2.749E+00	4.080E+00	6.119E+00	9.229E+00	1.396E+01	2.114E+01	3.203E+01	
.8	:	1.489E+00	2.363E+00	3.909E+00	6.624E+00	1.137E+01	1.966E+01	3.411E+01	5.929E+01	1.031E+02	
1	:	1.670E+00	3.010E+00	5.691E+00	1.105E+01	2.177E+01	4.322E+01	8.611E+01	1.719E+02	3.435E+02	
1.2	:	1.890E+00	3.891E+00	8.429E+00	1.879E+01	4.251E+01	9.690E+01	2.218E+02	5.085E+02	1.167E+03	
1.4	:	2.156E+00	5.087E+00	1.265E+01	3.238E+01	8.415E+01	2.204E+02	5.793E+02	1.526E+03	4.022E+03	
1.6	:	2.477E+00	6.716E+00	1.919E+01	5.644E+01	1.685E+02	5.067E+02	1.530E+03	4.630E+03	1.402E+04	
1.8	:	2.865E+00	8.943E+00	2.938E+01	9.927E+01	3.404E+02	1.176E+03	4.080E+03	1.418E+04	4.933E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 128

Mu \ N=		4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.6	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	9.999E-01	9.999E-01	9.999E-01	9.999E-01
-1.4	:	1.000E+00	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01
-1.2	:	9.999E-01	9.999E-01	9.998E-01	9.998E-01	9.998E-01	9.998E-01	9.998E-01	9.998E-01	9.998E-01	9.998E-01
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.001E+00	1.001E+00	1.002E+00	1.002E+00	1.002E+00	1.002E+00	1.002E+00	1.002E+00	1.002E+00	1.002E+00
-.6	:	1.003E+00	1.006E+00	1.008E+00	1.010E+00	1.012E+00	1.013E+00	1.014E+00	1.014E+00	1.015E+00	1.015E+00
-.4	:	1.010E+00	1.021E+00	1.031E+00	1.040E+00	1.047E+00	1.053E+00	1.058E+00	1.062E+00	1.065E+00	1.074E+00
-.2	:	1.029E+00	1.060E+00	1.092E+00	1.124E+00	1.153E+00	1.181E+00	1.205E+00	1.226E+00	1.245E+00	1.375E+00
0	:	1.065E+00	1.144E+00	1.232E+00	1.329E+00	1.430E+00	1.534E+00	1.640E+00	1.748E+00	1.856E+00	
.2	:	1.127E+00	1.294E+00	1.504E+00	1.759E+00	2.062E+00	2.416E+00	2.827E+00	3.303E+00	3.851E+00	
.4	:	1.217E+00	1.532E+00	1.973E+00	2.576E+00	3.388E+00	4.471E+00	5.909E+00	7.813E+00	1.033E+01	
.6	:	1.336E+00	1.877E+00	2.725E+00	4.037E+00	6.048E+00	9.115E+00	1.378E+01	2.086E+01	3.160E+01	
.8	:	1.486E+00	2.354E+00	3.891E+00	6.589E+00	1.131E+01	1.955E+01	3.391E+01	5.893E+01	1.025E+02	
1	:	1.668E+00	3.005E+00	5.679E+00	1.103E+01	2.172E+01	4.311E+01	8.589E+01	1.714E+02	3.426E+02	
1.2	:	1.889E+00	3.887E+00	8.421E+00	1.877E+01	4.246E+01	9.680E+01	2.215E+02	5.079E+02	1.166E+03	
1.4	:	2.156E+00	5.085E+00	1.265E+01	3.237E+01	8.411E+01	2.203E+02	5.790E+02	1.525E+03	4.021E+03	
1.6	:	2.477E+00	6.715E+00	1.919E+01	5.643E+01	1.684E+02	5.067E+02	1.530E+03	4.629E+03	1.402E+04	
1.8	:	2.865E+00	8.943E+00	2.938E+01	9.926E+01	3.404E+02	1.176E+03	4.080E+03	1.418E+04	4.932E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 256

Mu \ N=		4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.6	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.4	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	9.999E-01	9.999E-01	9.999E-01
-1.2	:	1.000E+00	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01	9.999E-01
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.000E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00
-.6	:	1.002E+00	1.004E+00	1.005E+00	1.007E+00	1.008E+00	1.008E+00	1.009E+00	1.009E+00	1.010E+00	1.010E+00
-.4	:	1.008E+00	1.016E+00	1.023E+00	1.029E+00	1.035E+00	1.039E+00	1.043E+00	1.046E+00	1.048E+00	1.055E+00
-.2	:	1.024E+00	1.050E+00	1.077E+00	1.103E+00	1.127E+00	1.150E+00	1.170E+00	1.188E+00	1.204E+00	1.311E+00
0	:	1.059E+00	1.130E+00	1.210E+00	1.296E+00	1.388E+00	1.482E+00	1.577E+00	1.674E+00	1.772E+00	
.2	:	1.121E+00	1.280E+00	1.479E+00	1.722E+00	2.009E+00	2.346E+00	2.737E+00	3.189E+00	3.710E+00	
.4	:	1.212E+00	1.520E+00	1.952E+00	2.541E+00	3.335E+00	4.394E+00	5.800E+00	7.662E+00	1.012E+01	
.6	:	1.333E+00	1.869E+00	2.709E+00	4.010E+00	6.003E+00	9.042E+00	1.366E+01	2.068E+01	3.132E+01	
.8	:	1.484E+00	2.350E+00	3.881E+00	6.570E+00	1.127E+01	1.948E+01	3.380E+01	5.873E+01	1.022E+02	
1	:	1.668E+00	3.003E+00	5.673E+00	1.101E+01	2.169E+01	4.305E+01	8.578E+01	1.712E+02	3.421E+02	
1.2	:	1.889E+00	3.886E+00	8.418E+00	1.876E+01	4.244E+01	9.675E+01	2.214E+02	5.077E+02	1.165E+03	
1.4	:	2.156E+00	5.084E+00	1.264E+01	3.236E+01	8.410E+01	2.202E+02	5.789E+02	1.525E+03	4.020E+03	
1.6	:	2.477E+00	6.715E+00	1.919E+01	5.642E+01	1.684E+02	5.066E+02	1.530E+03	4.629E+03	1.402E+04	
1.8	:	2.865E+00	8.943E+00	2.938E+01	9.926E+01	3.404E+02	1.176E+03	4.080E+03	1.418E+04	4.932E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 512

Mu \ N=		4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.6	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.4	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.000E+00	1.000E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00	1.001E+00
-.6	:	1.001E+00	1.003E+00	1.004E+00	1.004E+00	1.005E+00	1.006E+00	1.006E+00	1.006E+00	1.006E+00	1.007E+00
-.4	:	1.006E+00	1.012E+00	1.017E+00	1.022E+00	1.026E+00	1.030E+00	1.032E+00	1.034E+00	1.036E+00	1.041E+00
-.2	:	1.020E+00	1.042E+00	1.064E+00	1.086E+00	1.107E+00	1.125E+00	1.142E+00	1.157E+00	1.170E+00	1.261E+00
0	:	1.054E+00	1.118E+00	1.191E+00	1.270E+00	1.353E+00	1.438E+00	1.526E+00	1.614E+00	1.703E+00	
.2	:	1.116E+00	1.268E+00	1.460E+00	1.692E+00	1.967E+00	2.290E+00	2.665E+00	3.098E+00	3.598E+00	
.4	:	1.209E+00	1.512E+00	1.936E+00	2.516E+00	3.296E+00	4.338E+00	5.721E+00	7.553E+00	9.973E+00	
.6	:	1.331E+00	1.864E+00	2.699E+00	3.992E+00	5.973E+00	8.994E+00	1.359E+01	2.056E+01	3.114E+01	
.8	:	1.483E+00	2.347E+00	3.875E+00	6.558E+00	1.125E+01	1.945E+01	3.373E+01	5.862E+01	1.020E+02	
1	:	1.667E+00	3.001E+00	5.670E+00	1.101E+01	2.168E+01	4.303E+01	8.572E+01	1.711E+02	3.419E+02	
1.2	:	1.889E+00	3.885E+00	8.416E+00	1.876E+01	4.243E+01	9.673E+01	2.214E+02	5.076E+02	1.165E+03	
1.4	:	2.156E+00	5.084E+00	1.264E+01	3.236E+01	8.410E+01	2.202E+02	5.789E+02	1.525E+03	4.020E+03	
1.6	:	2.477E+00	6.714E+00	1.919E+01	5.642E+01	1.684E+02	5.066E+02	1.530E+03	4.628E+03	1.402E+04	
1.8	:	2.865E+00	8.943E+00	2.938E+01	9.926E+01	3.404E+02	1.176E+03	4.080E+03	1.418E+04	4.932E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 1024

Mu \ N=		4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.6	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.4	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.6	:	1.001E+00	1.002E+00	1.002E+00	1.003E+00	1.003E+00	1.004E+00	1.004E+00	1.004E+00	1.004E+00	1.004E+00
-.4	:	1.004E+00	1.009E+00	1.013E+00	1.017E+00	1.020E+00	1.022E+00	1.024E+00	1.026E+00	1.027E+00	1.031E+00
-.2	:	1.017E+00	1.035E+00	1.054E+00	1.073E+00	1.090E+00	1.106E+00	1.120E+00	1.132E+00	1.144E+00	1.220E+00
0	:	1.049E+00	1.108E+00	1.175E+00	1.248E+00	1.324E+00	1.402E+00	1.483E+00	1.564E+00	1.645E+00	
.2	:	1.112E+00	1.259E+00	1.444E+00	1.668E+00	1.934E+00	2.245E+00	2.607E+00	3.025E+00	3.507E+00	
.4	:	1.206E+00	1.505E+00	1.924E+00	2.497E+00	3.268E+00	4.297E+00	5.663E+00	7.472E+00	9.862E+00	
.6	:	1.330E+00	1.860E+00	2.693E+00	3.980E+00	5.953E+00	8.963E+00	1.354E+01	2.049E+01	3.102E+01	
.8	:	1.482E+00	2.345E+00	3.872E+00	6.552E+00	1.124E+01	1.942E+01	3.369E+01	5.855E+01	1.018E+02	
1	:	1.667E+00	3.001E+00	5.668E+00	1.100E+01	2.167E+01	4.301E+01	8.569E+01	1.711E+02	3.418E+02	
1.2	:	1.889E+00	3.885E+00	8.416E+00	1.876E+01	4.243E+01	9.672E+01	2.213E+02	5.075E+02	1.165E+03	
1.4	:	2.156E+00	5.084E+00	1.264E+01	3.236E+01	8.409E+01	2.202E+02	5.789E+02	1.525E+03	4.020E+03	
1.6	:	2.477E+00	6.714E+00	1.919E+01	5.642E+01	1.684E+02	5.066E+02	1.530E+03	4.628E+03	1.402E+04	
1.8	:	2.865E+00	8.943E+00	2.938E+01	9.926E+01	3.404E+02	1.176E+03	4.080E+03	1.418E+04	4.932E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B1(N,r,mu) for r = 2048

Mu \ N=		4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.6	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.4	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.6	:	1.001E+00	1.001E+00	1.002E+00	1.002E+00	1.002E+00	1.002E+00	1.003E+00	1.003E+00	1.003E+00	1.003E+00
-.4	:	1.003E+00	1.007E+00	1.010E+00	1.012E+00	1.015E+00	1.017E+00	1.018E+00	1.019E+00	1.020E+00	1.023E+00
-.2	:	1.014E+00	1.030E+00	1.046E+00	1.061E+00	1.076E+00	1.089E+00	1.101E+00	1.112E+00	1.122E+00	1.186E+00
0	:	1.045E+00	1.100E+00	1.162E+00	1.229E+00	1.299E+00	1.372E+00	1.446E+00	1.521E+00	1.596E+00	
.2	:	1.108E+00	1.251E+00	1.431E+00	1.648E+00	1.906E+00	2.209E+00	2.560E+00	2.966E+00	3.434E+00	
.4	:	1.204E+00	1.501E+00	1.916E+00	2.483E+00	3.247E+00	4.266E+00	5.620E+00	7.412E+00	9.780E+00	
.6	:	1.329E+00	1.858E+00	2.688E+00	3.972E+00	5.941E+00	8.942E+00	1.351E+01	2.044E+01	3.095E+01	
.8	:	1.482E+00	2.345E+00	3.870E+00	6.548E+00	1.123E+01	1.941E+01	3.367E+01	5.851E+01	1.018E+02	
1	:	1.667E+00	3.000E+00	5.667E+00	1.100E+01	2.167E+01	4.301E+01	8.568E+01	1.710E+02	3.417E+02	
1.2	:	1.889E+00	3.885E+00	8.415E+00	1.875E+01	4.243E+01	9.672E+01	2.213E+02	5.075E+02	1.165E+03	
1.4	:	2.156E+00	5.084E+00	1.264E+01	3.236E+01	8.409E+01	2.202E+02	5.789E+02	1.525E+03	4.020E+03	
1.6	:	2.477E+00	6.714E+00	1.919E+01	5.642E+01	1.684E+02	5.066E+02	1.530E+03	4.628E+03	1.402E+04	
1.8	:	2.865E+00	8.943E+00	2.938E+01	9.926E+01	3.404E+02	1.176E+03	4.080E+03	1.418E+04	4.932E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

BI(N,r,mu) for r = INFINITY

Mu \ N=		4	8	16	32	64	128	256	512	1024	INF
-2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.6	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.4	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.6	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.4	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
0	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	
.2	:	1.091E+00	1.210E+00	1.360E+00	1.541E+00	1.757E+00	2.009E+00	2.303E+00	2.642E+00	3.033E+00	
.4	:	1.198E+00	1.487E+00	1.890E+00	2.441E+00	3.184E+00	4.174E+00	5.489E+00	7.231E+00	9.533E+00	
.6	:	1.327E+00	1.854E+00	2.680E+00	3.958E+00	5.916E+00	8.903E+00	1.344E+01	2.034E+01	3.080E+01	
.8	:	1.482E+00	2.343E+00	3.867E+00	6.543E+00	1.123E+01	1.940E+01	3.364E+01	5.846E+01	1.017E+02	
1	:	1.667E+00	3.000E+00	5.667E+00	1.100E+01	2.167E+01	4.300E+01	8.567E+01	1.710E+02	3.417E+02	
1.2	:	1.889E+00	3.885E+00	8.415E+00	1.875E+01	4.243E+01	9.672E+01	2.213E+02	5.075E+02	1.165E+03	
1.4	:	2.156E+00	5.084E+00	1.264E+01	3.236E+01	8.409E+01	2.202E+02	5.789E+02	1.525E+03	4.020E+03	
1.6	:	2.477E+00	6.714E+00	1.919E+01	5.642E+01	1.684E+02	5.066E+02	1.530E+03	4.628E+03	1.402E+04	
1.8	:	2.865E+00	8.943E+00	2.938E+01	9.926E+01	3.404E+02	1.176E+03	4.080E+03	1.418E+04	4.932E+04	
2	:	3.333E+00	1.200E+01	4.533E+01	1.760E+02	6.933E+02	2.752E+03	1.097E+04	4.378E+04	1.749E+05	

B2(r,mu)

MU \ r =		.0001	.0003	.001	.003	.01	.03	.1	.3	.5	.7
-2	:	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01
-1.8	:	1.112E-01	1.385E-01	1.762E-01	2.195E-01	2.793E-01	3.479E-01	4.431E-01	5.566E-01	6.264E-01	6.889E-01
-1.6	:	1.874E-02	2.908E-02	4.708E-02	7.306E-02	1.183E-01	1.836E-01	2.979E-01	4.693E-01	5.901E-01	7.013E-01
-1.4	:	3.205E-03	6.196E-03	1.276E-02	2.467E-02	5.081E-02	9.828E-02	2.032E-01	3.999E-01	5.572E-01	7.061E-01
-1.2	:	5.586E-04	1.345E-03	3.525E-03	8.489E-03	2.225E-02	5.362E-02	1.410E-01	3.444E-01	5.273E-01	7.052E-01
-1	:	1.000E-04	3.000E-04	1.000E-03	3.000E-03	1.000E-02	3.000E-02	1.000E-01	3.000E-01	5.000E-01	7.000E-01
-.8	:	1.862E-05	6.956E-05	2.949E-04	1.101E-03	4.662E-03	1.735E-02	7.271E-02	2.642E-01	4.749E-01	6.916E-01
-.6	:	3.687E-06	1.715E-05	9.231E-05	4.280E-04	2.288E-03	1.047E-02	5.438E-02	2.351E-01	4.517E-01	6.808E-01
-.4	:	8.121E-07	4.678E-06	3.174E-05	1.809E-04	1.204E-03	6.664E-03	4.195E-02	2.112E-01	4.303E-01	6.683E-01
-.2	:	2.159E-07	1.510E-06	1.260E-05	8.606E-05	6.921E-04	4.507E-03	3.340E-02	1.915E-01	4.103E-01	6.546E-01
0	:	7.726E-08	6.240E-07	6.065E-06	4.745E-05	4.404E-04	3.250E-03	2.742E-02	1.750E-01	3.915E-01	6.401E-01
.2	:	3.906E-08	3.397E-07	3.594E-06	3.048E-05	3.100E-04	2.494E-03	2.316E-02	1.611E-01	3.740E-01	6.251E-01
.4	:	2.590E-08	2.311E-07	2.530E-06	2.228E-05	2.381E-04	2.020E-03	2.006E-02	1.492E-01	3.574E-01	6.097E-01
.6	:	2.013E-08	1.808E-07	2.001E-06	1.788E-05	1.955E-04	1.708E-03	1.773E-02	1.388E-01	3.416E-01	5.943E-01
.8	:	1.700E-08	1.529E-07	1.697E-06	1.524E-05	1.683E-04	1.493E-03	1.593E-02	1.297E-01	3.267E-01	5.789E-01
1	:	1.500E-08	1.350E-07	1.500E-06	1.349E-05	1.495E-04	1.337E-03	1.450E-02	1.215E-01	3.125E-01	5.635E-01
1.2	:	1.357E-08	1.221E-07	1.356E-06	1.221E-05	1.355E-04	1.216E-03	1.333E-02	1.141E-01	2.989E-01	5.483E-01
1.4	:	1.245E-08	1.120E-07	1.245E-06	1.120E-05	1.244E-04	1.118E-03	1.233E-02	1.074E-01	2.859E-01	5.333E-01
1.6	:	1.152E-08	1.037E-07	1.152E-06	1.037E-05	1.152E-04	1.036E-03	1.147E-02	1.012E-01	2.735E-01	5.186E-01
1.8	:	1.072E-08	9.645E-08	1.072E-06	9.645E-06	1.072E-04	9.642E-04	1.070E-02	9.541E-02	2.615E-01	5.042E-01
2	:	1.000E-08	9.000E-08	1.000E-06	9.000E-06	1.000E-04	9.000E-04	1.000E-02	9.000E-02	2.500E-01	4.900E-01

B2(r,mu)

MU \ r =	1	1.01	1.1	2	4	8	16	32	64	128
-2 :	1.000E+00	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01
-1.8 :	1.000E+00	8.614E-01	7.883E-01	7.196E-01	7.062E-01	7.028E-01	7.018E-01	7.015E-01	7.015E-01	7.014E-01
-1.6 :	1.000E+00	9.429E-01	8.708E-01	7.787E-01	7.561E-01	7.494E-01	7.472E-01	7.465E-01	7.462E-01	7.462E-01
-1.4 :	1.000E+00	9.776E-01	9.281E-01	8.446E-01	8.192E-01	8.103E-01	8.071E-01	8.058E-01	8.053E-01	8.052E-01
-1.2 :	1.000E+00	9.929E-01	9.693E-01	9.181E-01	8.990E-01	8.912E-01	8.879E-01	8.865E-01	8.859E-01	8.856E-01
-1 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8 :	1.000E+00	1.004E+00	1.024E+00	1.091E+00	1.128E+00	1.148E+00	1.159E+00	1.166E+00	1.170E+00	1.172E+00
-.6 :	1.000E+00	1.006E+00	1.043E+00	1.192E+00	1.290E+00	1.351E+00	1.392E+00	1.418E+00	1.436E+00	1.447E+00
-.4 :	1.000E+00	1.007E+00	1.060E+00	1.304E+00	1.494E+00	1.633E+00	1.738E+00	1.817E+00	1.877E+00	1.923E+00
-.2 :	1.000E+00	1.009E+00	1.075E+00	1.429E+00	1.752E+00	2.027E+00	2.265E+00	2.472E+00	2.652E+00	2.809E+00
0 :	1.000E+00	1.010E+00	1.089E+00	1.566E+00	2.078E+00	2.581E+00	3.082E+00	3.582E+00	4.082E+00	4.582E+00
.2 :	1.000E+00	1.011E+00	1.102E+00	1.718E+00	2.489E+00	3.364E+00	4.365E+00	5.514E+00	6.834E+00	8.351E+00
.4 :	1.000E+00	1.012E+00	1.114E+00	1.886E+00	3.007E+00	4.473E+00	6.404E+00	8.951E+00	1.231E+01	1.674E+01
.6 :	1.000E+00	1.013E+00	1.126E+00	2.071E+00	3.658E+00	6.051E+00	9.673E+00	1.516E+01	2.348E+01	3.609E+01
.8 :	1.000E+00	1.014E+00	1.138E+00	2.275E+00	4.475E+00	8.297E+00	1.495E+01	2.653E+01	4.669E+01	8.179E+01
1 :	1.000E+00	1.015E+00	1.150E+00	2.500E+00	5.500E+00	1.150E+01	2.350E+01	4.750E+01	9.550E+01	1.915E+02
1.2 :	1.000E+00	1.016E+00	1.162E+00	2.747E+00	6.784E+00	1.607E+01	3.741E+01	8.644E+01	1.991E+02	4.579E+02
1.4 :	1.000E+00	1.017E+00	1.174E+00	3.018E+00	8.389E+00	2.259E+01	6.008E+01	1.590E+02	4.201E+02	1.109E+03
1.6 :	1.000E+00	1.018E+00	1.186E+00	3.316E+00	1.039E+01	3.188E+01	9.706E+01	2.947E+02	8.937E+02	2.710E+03
1.8 :	1.000E+00	1.019E+00	1.198E+00	3.642E+00	1.289E+01	4.513E+01	1.574E+02	5.485E+02	1.910E+03	6.653E+03
2 :	1.000E+00	1.020E+00	1.210E+00	4.000E+00	1.600E+01	6.400E+01	2.560E+02	1.024E+03	4.096E+03	1.638E+04

MU \ r =	256	512	1024	2048	4096	INF
-2 :	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01	6.667E-01
-1.8 :	7.014E-01	7.014E-01	7.014E-01	7.014E-01	7.014E-01	7.014E-01
-1.6 :	7.461E-01	7.461E-01	7.461E-01	7.461E-01	7.461E-01	7.461E-01
-1.4 :	8.051E-01	8.051E-01	8.051E-01	8.051E-01	8.051E-01	8.051E-01
-1.2 :	8.855E-01	8.854E-01	8.854E-01	8.854E-01	8.854E-01	8.854E-01
-1 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8 :	1.173E+00	1.174E+00	1.174E+00	1.174E+00	1.174E+00	1.175E+00
-.6 :	1.455E+00	1.460E+00	1.463E+00	1.465E+00	1.467E+00	1.470E+00
-.4 :	1.957E+00	1.983E+00	2.003E+00	2.018E+00	2.029E+00	2.065E+00
-.2 :	2.945E+00	3.064E+00	3.167E+00	3.257E+00	3.336E+00	3.863E+00
0 :	5.082E+00	5.582E+00	6.082E+00	6.582E+00	7.082E+00	
.2 :	1.009E+01	1.209E+01	1.439E+01	1.703E+01	2.006E+01	
.4 :	2.259E+01	3.031E+01	4.050E+01	5.394E+01	7.167E+01	
.6 :	5.521E+01	8.418E+01	1.281E+02	1.947E+02	2.955E+02	
.8 :	1.429E+02	2.493E+02	4.346E+02	7.571E+02	1.319E+03	
1 :	3.835E+02	7.675E+02	1.536E+03	3.072E+03	6.144E+03	
1.2 :	1.052E+03	2.418E+03	5.556E+03	1.277E+04	2.933E+04	
1.4 :	2.928E+03	7.727E+03	2.039E+04	5.382E+04	1.420E+05	
1.6 :	8.215E+03	2.490E+04	7.549E+04	2.288E+05	6.937E+05	
1.8 :	2.317E+04	8.067E+04	2.809E+05	9.782E+05	3.406E+06	
2 :	6.554E+04	2.621E+05	1.049E+06	4.194E+06	1.678E+07	

B3(2,M,r,mu) for r = .01

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8	1.000E+00	2.214E+00	5.842E+00	1.829E+01	6.502E+01	2.536E+02	1.173E+03	3.308E+03	4.836E+03	6.080E+03	7.145E+03
-1.6	1.000E+00	2.419E+00	7.398E+00	2.638E+01	1.018E+02	4.151E+02	1.888E+03	5.075E+03	7.222E+03	8.748E+03	9.882E+03
-1.4	1.000E+00	2.617E+00	8.745E+00	3.295E+01	1.306E+02	5.358E+02	2.342E+03	5.964E+03	8.263E+03	9.700E+03	1.063E+04
-1.2	1.000E+00	2.811E+00	9.937E+00	3.843E+01	1.534E+02	6.247E+02	2.595E+03	6.253E+03	8.449E+03	9.671E+03	1.036E+04
-1	1.000E+00	3.000E+00	1.100E+01	4.300E+01	1.710E+02	6.830E+02	2.674E+03	6.094E+03	8.047E+03	9.024E+03	9.512E+03
-.8	1.000E+00	3.180E+00	1.191E+01	4.651E+01	1.823E+02	7.065E+02	2.588E+03	5.578E+03	7.208E+03	7.950E+03	8.280E+03
-.6	1.000E+00	3.337E+00	1.258E+01	4.848E+01	1.851E+02	6.877E+02	2.342E+03	4.775E+03	6.045E+03	6.577E+03	6.791E+03
-.4	1.000E+00	3.444E+00	1.285E+01	4.812E+01	1.766E+02	6.215E+02	1.957E+03	3.774E+03	4.686E+03	5.041E+03	5.171E+03
-.2	1.000E+00	3.463E+00	1.252E+01	4.480E+01	1.557E+02	5.131E+02	1.486E+03	2.712E+03	3.304E+03	3.521E+03	3.594E+03
0	1.000E+00	3.368E+00	1.151E+01	3.866E+01	1.253E+02	3.826E+02	1.015E+03	1.753E+03	2.097E+03	2.216E+03	2.254E+03
.2	1.000E+00	3.163E+00	9.992E+00	3.090E+01	9.213E+01	2.579E+02	6.253E+02	1.021E+03	1.200E+03	1.259E+03	1.277E+03
.4	1.000E+00	2.883E+00	8.251E+00	2.314E+01	6.266E+01	1.595E+02	3.521E+02	5.442E+02	6.280E+02	6.549E+02	6.627E+02
.6	1.000E+00	2.574E+00	6.575E+00	1.651E+01	4.018E+01	9.228E+01	1.851E+02	2.709E+02	3.071E+02	3.183E+02	3.215E+02
.8	1.000E+00	2.270E+00	5.123E+00	1.141E+01	2.475E+01	5.098E+01	9.280E+01	1.285E+02	1.431E+02	1.476E+02	1.488E+02
1	1.000E+00	1.989E+00	3.940E+00	7.728E+00	1.486E+01	2.731E+01	4.505E+01	5.908E+01	6.464E+01	6.629E+01	6.673E+01
1.2	1.000E+00	1.737E+00	3.008E+00	5.177E+00	8.775E+00	1.435E+01	2.142E+01	2.660E+01	2.859E+01	2.917E+01	2.933E+01
1.4	1.000E+00	1.514E+00	2.289E+00	3.446E+00	5.133E+00	7.440E+00	1.005E+01	1.182E+01	1.248E+01	1.267E+01	1.272E+01
1.6	1.000E+00	1.319E+00	1.738E+00	2.285E+00	2.985E+00	3.828E+00	4.678E+00	5.211E+00	5.405E+00	5.459E+00	5.473E+00
1.8	1.000E+00	1.149E+00	1.319E+00	1.513E+00	1.729E+00	1.960E+00	2.166E+00	2.286E+00	2.328E+00	2.340E+00	2.343E+00
2	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = .03

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8	1.000E+00	2.216E+00	5.866E+00	1.865E+01	7.138E+01	2.812E+02	4.880E+02	6.534E+02	8.075E+02	9.647E+02	1.143E+03
-1.6	1.000E+00	2.422E+00	7.449E+00	2.705E+01	1.119E+02	4.279E+02	7.195E+02	9.170E+02	1.065E+03	1.180E+03	1.274E+03
-1.4	1.000E+00	2.623E+00	8.817E+00	3.378E+01	1.411E+02	5.073E+02	8.285E+02	1.022E+03	1.147E+03	1.229E+03	1.283E+03
-1.2	1.000E+00	2.816E+00	1.000E+01	3.910E+01	1.607E+02	5.386E+02	8.548E+02	1.027E+03	1.125E+03	1.180E+03	1.212E+03
-1	1.000E+00	3.000E+00	1.100E+01	4.300E+01	1.710E+02	5.328E+02	8.221E+02	9.667E+02	1.039E+03	1.075E+03	1.093E+03
-.8	1.000E+00	3.164E+00	1.175E+01	4.523E+01	1.719E+02	4.971E+02	7.455E+02	8.606E+02	9.122E+02	9.350E+02	9.451E+02
-.6	1.000E+00	3.290E+00	1.217E+01	4.543E+01	1.632E+02	4.376E+02	6.378E+02	7.247E+02	7.600E+02	7.740E+02	7.794E+02
-.4	1.000E+00	3.356E+00	1.213E+01	4.331E+01	1.455E+02	3.618E+02	5.126E+02	5.743E+02	5.974E+02	6.057E+02	6.086E+02
-.2	1.000E+00	3.339E+00	1.157E+01	3.896E+01	1.214E+02	2.797E+02	3.851E+02	4.262E+02	4.405E+02	4.452E+02	4.468E+02
0	1.000E+00	3.229E+00	1.052E+01	3.298E+01	9.462E+01	2.019E+02	2.702E+02	2.957E+02	3.040E+02	3.066E+02	3.074E+02
.2	1.000E+00	3.036E+00	9.146E+00	2.635E+01	6.912E+01	1.366E+02	1.776E+02	1.924E+02	1.970E+02	1.983E+02	1.987E+02
.4	1.000E+00	2.787E+00	7.640E+00	2.002E+01	4.774E+01	8.737E+01	1.104E+02	1.184E+02	1.209E+02	1.215E+02	1.217E+02
.6	1.000E+00	2.510E+00	6.186E+00	1.461E+01	3.153E+01	5.345E+01	6.564E+01	6.975E+01	7.095E+01	7.128E+01	7.136E+01
.8	1.000E+00	2.232E+00	4.897E+00	1.035E+01	2.013E+01	3.162E+01	3.772E+01	3.973E+01	4.030E+01	4.045E+01	4.049E+01
1	1.000E+00	1.967E+00	3.818E+00	7.179E+00	1.254E+01	1.826E+01	2.116E+01	2.210E+01	2.236E+01	2.242E+01	2.244E+01
1.2	1.000E+00	1.725E+00	2.946E+00	4.907E+00	7.688E+00	1.037E+01	1.168E+01	1.209E+01	1.220E+01	1.223E+01	1.224E+01
1.4	1.000E+00	1.508E+00	2.259E+00	3.322E+00	4.658E+00	5.829E+00	6.371E+00	6.538E+00	6.583E+00	6.595E+00	6.598E+00
1.6	1.000E+00	1.316E+00	1.725E+00	2.234E+00	2.801E+00	3.251E+00	3.450E+00	3.511E+00	3.527E+00	3.531E+00	3.532E+00
1.8	1.000E+00	1.148E+00	1.314E+00	1.497E+00	1.676E+00	1.805E+00	1.860E+00	1.876E+00	1.881E+00	1.882E+00	1.882E+00
2	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = .1

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2 :	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8 :	1.000E+00	2.229E+00	6.118E+00	2.327E+01	4.419E+01	5.757E+01	6.871E+01	7.831E+01	8.665E+01	9.393E+01	1.003E+02
-1.6 :	1.000E+00	2.448E+00	7.866E+00	3.138E+01	6.097E+01	7.934E+01	9.267E+01	1.026E+02	1.101E+02	1.158E+02	1.201E+02
-1.4 :	1.000E+00	2.654E+00	9.270E+00	3.626E+01	6.919E+01	8.832E+01	1.005E+02	1.084E+02	1.136E+02	1.170E+02	1.193E+02
-1.2 :	1.000E+00	2.842E+00	1.032E+01	3.853E+01	7.133E+01	8.911E+01	9.909E+01	1.048E+02	1.080E+02	1.099E+02	1.109E+02
-1 :	1.000E+00	3.000E+00	1.100E+01	3.863E+01	6.906E+01	8.453E+01	9.227E+01	9.613E+01	9.807E+01	9.903E+01	9.952E+01
-.8 :	1.000E+00	3.117E+00	1.127E+01	3.694E+01	6.365E+01	7.645E+01	8.222E+01	8.478E+01	8.591E+01	8.640E+01	8.662E+01
-.6 :	1.000E+00	3.180E+00	1.114E+01	3.385E+01	5.617E+01	6.632E+01	7.049E+01	7.215E+01	7.280E+01	7.305E+01	7.314E+01
-.4 :	1.000E+00	3.179E+00	1.061E+01	2.981E+01	4.760E+01	5.532E+01	5.825E+01	5.931E+01	5.968E+01	5.981E+01	5.985E+01
-.2 :	1.000E+00	3.111E+00	9.757E+00	2.528E+01	3.884E+01	4.447E+01	4.647E+01	4.714E+01	4.735E+01	4.742E+01	4.744E+01
0 :	1.000E+00	2.980E+00	8.683E+00	2.069E+01	3.059E+01	3.455E+01	3.587E+01	3.629E+01	3.641E+01	3.645E+01	3.646E+01
.2 :	1.000E+00	2.799E+00	7.502E+00	1.642E+01	2.335E+01	2.603E+01	2.689E+01	2.714E+01	2.721E+01	2.723E+01	2.724E+01
.4 :	1.000E+00	2.584E+00	6.317E+00	1.269E+01	1.735E+01	1.910E+01	1.964E+01	1.979E+01	1.984E+01	1.985E+01	1.985E+01
.6 :	1.000E+00	2.352E+00	5.208E+00	9.591E+00	1.262E+01	1.372E+01	1.405E+01	1.414E+01	1.416E+01	1.417E+01	1.417E+01
.8 :	1.000E+00	2.116E+00	4.221E+00	7.125E+00	9.013E+00	9.683E+00	9.878E+00	9.931E+00	9.944E+00	9.948E+00	9.949E+00
1 :	1.000E+00	1.888E+00	3.377E+00	5.223E+00	6.354E+00	6.745E+00	6.857E+00	6.886E+00	6.894E+00	6.896E+00	6.896E+00
1.2 :	1.000E+00	1.674E+00	2.676E+00	3.791E+00	4.435E+00	4.653E+00	4.714E+00	4.730E+00	4.734E+00	4.735E+00	4.735E+00
1.4 :	1.000E+00	1.477E+00	2.105E+00	2.732E+00	3.074E+00	3.187E+00	3.218E+00	3.226E+00	3.228E+00	3.228E+00	3.229E+00
1.6 :	1.000E+00	1.299E+00	1.647E+00	1.960E+00	2.120E+00	2.172E+00	2.186E+00	2.189E+00	2.190E+00	2.191E+00	2.191E+00
1.8 :	1.000E+00	1.141E+00	1.285E+00	1.401E+00	1.457E+00	1.475E+00	1.480E+00	1.481E+00	1.481E+00	1.482E+00	1.482E+00
2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = .3

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2 :	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8 :	1.000E+00	2.388E+00	5.582E+00	9.024E+00	1.341E+01	1.980E+01	2.987E+01	4.649E+01	7.464E+01	1.230E+02	2.065E+02
-1.6 :	1.000E+00	2.669E+00	6.503E+00	9.614E+00	1.237E+01	1.521E+01	1.854E+01	2.285E+01	2.882E+01	3.746E+01	5.023E+01
-1.4 :	1.000E+00	2.857E+00	6.937E+00	9.770E+00	1.171E+01	1.317E+01	1.440E+01	1.557E+01	1.681E+01	1.825E+01	2.002E+01
-1.2 :	1.000E+00	2.964E+00	7.013E+00	9.562E+00	1.102E+01	1.188E+01	1.242E+01	1.278E+01	1.305E+01	1.328E+01	1.348E+01
-1 :	1.000E+00	3.000E+00	6.833E+00	9.083E+00	1.021E+01	1.077E+01	1.105E+01	1.119E+01	1.126E+01	1.130E+01	1.132E+01
-.8 :	1.000E+00	2.976E+00	6.477E+00	8.422E+00	9.300E+00	9.686E+00	9.850E+00	9.918E+00	9.944E+00	9.952E+00	9.953E+00
-.6 :	1.000E+00	2.902E+00	6.008E+00	7.655E+00	8.339E+00	8.610E+00	8.713E+00	8.750E+00	8.763E+00	8.766E+00	8.766E+00
-.4 :	1.000E+00	2.789E+00	5.478E+00	6.844E+00	7.374E+00	7.566E+00	7.633E+00	7.655E+00	7.662E+00	7.664E+00	7.664E+00
-.2 :	1.000E+00	2.648E+00	4.923E+00	6.036E+00	6.443E+00	6.580E+00	6.624E+00	6.638E+00	6.642E+00	6.643E+00	6.643E+00
0 :	1.000E+00	2.487E+00	4.374E+00	5.264E+00	5.573E+00	5.671E+00	5.701E+00	5.709E+00	5.712E+00	5.712E+00	5.713E+00
.2 :	1.000E+00	2.316E+00	3.850E+00	4.548E+00	4.781E+00	4.851E+00	4.871E+00	4.876E+00	4.878E+00	4.878E+00	4.879E+00
.4 :	1.000E+00	2.140E+00	3.362E+00	3.900E+00	4.073E+00	4.123E+00	4.137E+00	4.140E+00	4.141E+00	4.141E+00	4.142E+00
.6 :	1.000E+00	1.967E+00	2.919E+00	3.324E+00	3.450E+00	3.486E+00	3.495E+00	3.498E+00	3.498E+00	3.499E+00	3.499E+00
.8 :	1.000E+00	1.799E+00	2.522E+00	2.820E+00	2.910E+00	2.935E+00	2.942E+00	2.943E+00	2.944E+00	2.944E+00	2.944E+00
1 :	1.000E+00	1.639E+00	2.171E+00	2.383E+00	2.446E+00	2.463E+00	2.468E+00	2.469E+00	2.469E+00	2.469E+00	2.469E+00
1.2 :	1.000E+00	1.489E+00	1.864E+00	2.009E+00	2.051E+00	2.062E+00	2.065E+00	2.066E+00	2.066E+00	2.066E+00	2.066E+00
1.4 :	1.000E+00	1.350E+00	1.597E+00	1.690E+00	1.716E+00	1.723E+00	1.725E+00	1.726E+00	1.726E+00	1.726E+00	1.726E+00
1.6 :	1.000E+00	1.223E+00	1.367E+00	1.420E+00	1.434E+00	1.438E+00	1.439E+00	1.440E+00	1.440E+00	1.440E+00	1.440E+00
1.8 :	1.000E+00	1.106E+00	1.169E+00	1.192E+00	1.198E+00	1.200E+00	1.200E+00	1.200E+00	1.200E+00	1.200E+00	1.200E+00
2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 1

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.8 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.6 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.4 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1.2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-1 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.6 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.4 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
0 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
.2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
.4 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
.6 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
.8 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
1 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
1.2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
1.4 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
1.6 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
1.8 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 1.01

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2 :	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8 :	1.000E+00	1.300E+00	1.771E+00	2.545E+00	3.852E+00	6.095E+00	9.968E+00	1.669E+01	2.836E+01	4.867E+01	8.401E+01
-1.6 :	1.000E+00	1.105E+00	1.234E+00	1.407E+00	1.650E+00	2.004E+00	2.532E+00	3.324E+00	4.518E+00	6.324E+00	9.057E+00
-1.4 :	1.000E+00	1.037E+00	1.073E+00	1.110E+00	1.155E+00	1.209E+00	1.278E+00	1.367E+00	1.484E+00	1.637E+00	1.838E+00
-1.2 :	1.000E+00	1.011E+00	1.019E+00	1.026E+00	1.032E+00	1.039E+00	1.046E+00	1.054E+00	1.063E+00	1.073E+00	1.084E+00
-1 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8 :	1.000E+00	9.956E-01	9.933E-01	9.921E-01	9.913E-01	9.908E-01	9.905E-01	9.902E-01	9.899E-01	9.897E-01	9.896E-01
-.6 :	1.000E+00	9.938E-01	9.910E-01	9.898E-01	9.892E-01	9.889E-01	9.887E-01	9.886E-01	9.885E-01	9.884E-01	9.884E-01
-.4 :	1.000E+00	9.932E-01	9.905E-01	9.895E-01	9.891E-01	9.889E-01	9.888E-01	9.888E-01	9.887E-01	9.887E-01	9.887E-01
-.2 :	1.000E+00	9.931E-01	9.907E-01	9.899E-01	9.896E-01	9.895E-01	9.895E-01	9.894E-01	9.894E-01	9.894E-01	9.894E-01
0 :	1.000E+00	9.934E-01	9.913E-01	9.906E-01	9.904E-01	9.903E-01	9.903E-01	9.903E-01	9.903E-01	9.903E-01	9.903E-01
.2 :	1.000E+00	9.938E-01	9.920E-01	9.914E-01	9.913E-01	9.912E-01	9.912E-01	9.912E-01	9.912E-01	9.912E-01	9.912E-01
.4 :	1.000E+00	9.944E-01	9.928E-01	9.923E-01	9.922E-01	9.922E-01	9.922E-01	9.922E-01	9.922E-01	9.922E-01	9.922E-01
.6 :	1.000E+00	9.950E-01	9.936E-01	9.932E-01	9.932E-01	9.931E-01	9.931E-01	9.931E-01	9.931E-01	9.931E-01	9.931E-01
.8 :	1.000E+00	9.956E-01	9.945E-01	9.942E-01	9.941E-01	9.941E-01	9.941E-01	9.941E-01	9.941E-01	9.941E-01	9.941E-01
1 :	1.000E+00	9.963E-01	9.954E-01	9.952E-01	9.951E-01	9.951E-01	9.951E-01	9.951E-01	9.951E-01	9.951E-01	9.951E-01
1.2 :	1.000E+00	9.970E-01	9.963E-01	9.961E-01	9.961E-01	9.961E-01	9.961E-01	9.961E-01	9.961E-01	9.961E-01	9.961E-01
1.4 :	1.000E+00	9.978E-01	9.972E-01	9.971E-01	9.970E-01	9.970E-01	9.970E-01	9.970E-01	9.970E-01	9.970E-01	9.970E-01
1.6 :	1.000E+00	9.985E-01	9.981E-01	9.981E-01	9.980E-01	9.980E-01	9.980E-01	9.980E-01	9.980E-01	9.980E-01	9.980E-01
1.8 :	1.000E+00	9.992E-01	9.991E-01	9.990E-01	9.990E-01	9.990E-01	9.990E-01	9.990E-01	9.990E-01	9.990E-01	9.990E-01
2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 1.1

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2 :	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8 :	1.000E+00	1.494E+00	2.286E+00	3.604E+00	5.845E+00	9.700E+00	1.637E+01	2.795E+01	4.808E+01	8.311E+01	1.441E+02
-1.6 :	1.000E+00	1.250E+00	1.573E+00	2.017E+00	2.657E+00	3.601E+00	5.011E+00	7.134E+00	1.034E+01	1.519E+01	2.254E+01
-1.4 :	1.000E+00	1.119E+00	1.243E+00	1.385E+00	1.556E+00	1.772E+00	2.050E+00	2.412E+00	2.887E+00	3.512E+00	4.336E+00
-1.2 :	1.000E+00	1.044E+00	1.082E+00	1.117E+00	1.153E+00	1.191E+00	1.234E+00	1.282E+00	1.336E+00	1.398E+00	1.470E+00
-1 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8 :	1.000E+00	9.740E-01	9.584E-01	9.484E-01	9.414E-01	9.361E-01	9.318E-01	9.282E-01	9.251E-01	9.225E-01	9.202E-01
-.6 :	1.000E+00	9.590E-01	9.380E-01	9.268E-01	9.202E-01	9.160E-01	9.132E-01	9.111E-01	9.096E-01	9.085E-01	9.077E-01
-.4 :	1.000E+00	9.510E-01	9.292E-01	9.193E-01	9.145E-01	9.119E-01	9.105E-01	9.096E-01	9.090E-01	9.087E-01	9.085E-01
-.2 :	1.000E+00	9.476E-01	9.270E-01	9.189E-01	9.156E-01	9.142E-01	9.135E-01	9.131E-01	9.129E-01	9.128E-01	9.128E-01
0 :	1.000E+00	9.472E-01	9.287E-01	9.223E-01	9.200E-01	9.192E-01	9.189E-01	9.187E-01	9.187E-01	9.186E-01	9.186E-01
.2 :	1.000E+00	9.490E-01	9.327E-01	9.277E-01	9.261E-01	9.256E-01	9.254E-01	9.254E-01	9.253E-01	9.253E-01	9.253E-01
.4 :	1.000E+00	9.523E-01	9.382E-01	9.342E-01	9.331E-01	9.327E-01	9.326E-01	9.326E-01	9.326E-01	9.326E-01	9.326E-01
.6 :	1.000E+00	9.566E-01	9.447E-01	9.415E-01	9.406E-01	9.404E-01	9.403E-01	9.403E-01	9.403E-01	9.403E-01	9.403E-01
.8 :	1.000E+00	9.617E-01	9.517E-01	9.492E-01	9.485E-01	9.483E-01	9.483E-01	9.483E-01	9.483E-01	9.483E-01	9.483E-01
1 :	1.000E+00	9.674E-01	9.592E-01	9.572E-01	9.567E-01	9.566E-01	9.565E-01	9.565E-01	9.565E-01	9.565E-01	9.565E-01
1.2 :	1.000E+00	9.735E-01	9.670E-01	9.655E-01	9.651E-01	9.650E-01	9.649E-01	9.649E-01	9.649E-01	9.649E-01	9.649E-01
1.4 :	1.000E+00	9.798E-01	9.751E-01	9.739E-01	9.736E-01	9.735E-01	9.735E-01	9.735E-01	9.735E-01	9.735E-01	9.735E-01
1.6 :	1.000E+00	9.864E-01	9.832E-01	9.825E-01	9.823E-01	9.822E-01	9.822E-01	9.822E-01	9.822E-01	9.822E-01	9.822E-01
1.8 :	1.000E+00	9.931E-01	9.916E-01	9.912E-01	9.911E-01	9.911E-01	9.911E-01	9.911E-01	9.911E-01	9.911E-01	9.911E-01
2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 2

M=	1	2	4	8	16	32	64	128	256	512	1024
-2 :	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8 :	1.000E+00	1.690E+00	2.862E+00	4.877E+00	8.362E+00	1.441E+01	2.491E+01	4.319E+01	7.500E+01	1.304E+02	2.267E+02
-1.6 :	1.000E+00	1.445E+00	2.089E+00	3.037E+00	4.453E+00	6.583E+00	9.798E+00	1.466E+01	2.203E+01	3.319E+01	5.010E+01
-1.4 :	1.000E+00	1.255E+00	1.568E+00	1.964E+00	2.473E+00	3.137E+00	4.006E+00	5.150E+00	6.657E+00	8.644E+00	1.126E+01
-1.2 :	1.000E+00	1.109E+00	1.224E+00	1.347E+00	1.484E+00	1.638E+00	1.813E+00	2.014E+00	2.244E+00	2.507E+00	2.810E+00
-1 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8 :	1.000E+00	9.198E-01	8.591E-01	8.113E-01	7.723E-01	7.395E-01	7.115E-01	6.873E-01	6.664E-01	6.483E-01	6.325E-01
-.6 :	1.000E+00	8.632E-01	7.746E-01	7.151E-01	6.735E-01	6.435E-01	6.213E-01	6.047E-01	5.922E-01	5.828E-01	5.757E-01
-.4 :	1.000E+00	8.255E-01	7.283E-01	6.727E-01	6.394E-01	6.188E-01	6.057E-01	5.972E-01	5.917E-01	5.881E-01	5.857E-01
-.2 :	1.000E+00	8.028E-01	7.080E-01	6.614E-01	6.377E-01	6.251E-01	6.183E-01	6.144E-01	6.123E-01	6.110E-01	6.103E-01
0 :	1.000E+00	7.922E-01	7.052E-01	6.684E-01	6.523E-01	6.451E-01	6.417E-01	6.401E-01	6.393E-01	6.389E-01	6.387E-01
.2 :	1.000E+00	7.912E-01	7.143E-01	6.860E-01	6.754E-01	6.713E-01	6.697E-01	6.690E-01	6.687E-01	6.686E-01	6.686E-01
.4 :	1.000E+00	7.979E-01	7.316E-01	7.101E-01	7.031E-01	7.008E-01	7.000E-01	6.997E-01	6.996E-01	6.996E-01	6.996E-01
.6 :	1.000E+00	8.106E-01	7.547E-01	7.384E-01	7.337E-01	7.323E-01	7.319E-01	7.318E-01	7.318E-01	7.318E-01	7.318E-01
.8 :	1.000E+00	8.283E-01	7.820E-01	7.696E-01	7.664E-01	7.655E-01	7.653E-01	7.652E-01	7.652E-01	7.652E-01	7.652E-01
1 :	1.000E+00	8.500E-01	8.125E-01	8.031E-01	8.008E-01	8.002E-01	8.000E-01	8.000E-01	8.000E-01	8.000E-01	8.000E-01
1.2 :	1.000E+00	8.750E-01	8.456E-01	8.386E-01	8.369E-01	8.365E-01	8.364E-01	8.364E-01	8.363E-01	8.363E-01	8.363E-01
1.4 :	1.000E+00	9.029E-01	8.811E-01	8.760E-01	8.747E-01	8.745E-01	8.744E-01	8.744E-01	8.744E-01	8.744E-01	8.744E-01
1.6 :	1.000E+00	9.331E-01	9.186E-01	9.153E-01	9.145E-01	9.143E-01	9.142E-01	9.142E-01	9.142E-01	9.142E-01	9.142E-01
1.8 :	1.000E+00	9.656E-01	9.582E-01	9.566E-01	9.562E-01	9.561E-01	9.560E-01	9.560E-01	9.560E-01	9.560E-01	9.560E-01
2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

```
                 B3(2,M,r,mu)  for  r =  4

Mu \  M=  1          2          4          8          16         32         64         128        256        512        1024
-------------------------------------------------------------------------------------------------------------------------------
-2    :   1.000E+00  2.000E+00  4.000E+00  8.000E+00  1.600E+01  3.200E+01  6.400E+01  1.280E+02  2.560E+02  5.120E+02  1.024E+03
-1.8  :   1.000E+00  1.728E+00  2.986E+00  5.170E+00  8.964E+00  1.556E+01  2.705E+01  4.704E+01  8.185E+01  1.425E+02  2.480E+02
-1.6  :   1.000E+00  1.494E+00  2.232E+00  3.341E+00  5.014E+00  7.544E+00  1.137E+01  1.718E+01  2.597E+01  3.929E+01  5.948E+01
-1.4  :   1.000E+00  1.296E+00  1.678E+00  2.174E+00  2.823E+00  3.676E+00  4.799E+00  6.279E+00  8.232E+00  1.081E+01  1.420E+01
-1.2  :   1.000E+00  1.132E+00  1.279E+00  1.444E+00  1.631E+00  1.844E+00  2.088E+00  2.367E+00  2.688E+00  3.057E+00  3.480E+00
-1    :   1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00
-.8   :   1.000E+00  8.967E-01  8.123E-01  7.420E-01  6.824E-01  6.313E-01  5.872E-01  5.489E-01  5.157E-01  4.868E-01  4.616E-01
-.6   :   1.000E+00  8.198E-01  6.934E-01  6.031E-01  5.370E-01  4.880E-01  4.513E-01  4.237E-01  4.028E-01  3.870E-01  3.750E-01
-.4   :   1.000E+00  7.663E-01  6.256E-01  5.392E-01  4.849E-01  4.502E-01  4.276E-01  4.128E-01  4.032E-01  3.968E-01  3.926E-01
-.2   :   1.000E+00  7.330E-01  5.946E-01  5.217E-01  4.824E-01  4.607E-01  4.486E-01  4.417E-01  4.378E-01  4.355E-01  4.342E-01
 0    :   1.000E+00  7.167E-01  5.898E-01  5.323E-01  5.057E-01  4.932E-01  4.871E-01  4.841E-01  4.826E-01  4.819E-01  4.815E-01
  .2  :   1.000E+00  7.144E-01  6.030E-01  5.595E-01  5.422E-01  5.351E-01  5.322E-01  5.310E-01  5.305E-01  5.303E-01  5.302E-01
  .4  :   1.000E+00  7.232E-01  6.285E-01  5.962E-01  5.851E-01  5.812E-01  5.798E-01  5.794E-01  5.792E-01  5.791E-01  5.791E-01
  .6  :   1.000E+00  7.409E-01  6.621E-01  6.384E-01  6.312E-01  6.291E-01  6.284E-01  6.282E-01  6.281E-01  6.281E-01  6.281E-01
  .8  :   1.000E+00  7.654E-01  7.013E-01  6.839E-01  6.792E-01  6.779E-01  6.775E-01  6.775E-01  6.774E-01  6.774E-01  6.774E-01
  1   :   1.000E+00  7.955E-01  7.443E-01  7.315E-01  7.283E-01  7.275E-01  7.273E-01  7.273E-01  7.273E-01  7.273E-01  7.273E-01
  1.2 :   1.000E+00  8.298E-01  7.903E-01  7.810E-01  7.788E-01  7.782E-01  7.781E-01  7.781E-01  7.781E-01  7.781E-01  7.781E-01
  1.4 :   1.000E+00  8.679E-01  8.389E-01  8.323E-01  8.307E-01  8.303E-01  8.303E-01  8.302E-01  8.302E-01  8.302E-01  8.302E-01
  1.6 :   1.000E+00  9.090E-01  8.899E-01  8.856E-01  8.846E-01  8.843E-01  8.843E-01  8.843E-01  8.843E-01  8.843E-01  8.843E-01
  1.8 :   1.000E+00  9.531E-01  9.435E-01  9.414E-01  9.408E-01  9.407E-01  9.407E-01  9.407E-01  9.407E-01  9.407E-01  9.407E-01
  2   :   1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00
```

```
                 B3(2,M,r,mu)  for  r =  8

Mu \  M=  1          2          4          8          16         32         64         128        256        512        1024
-------------------------------------------------------------------------------------------------------------------------------
-2    :   1.000E+00  2.000E+00  4.000E+00  8.000E+00  1.600E+01  3.200E+01  6.400E+01  1.280E+02  2.560E+02  5.120E+02  1.024E+03
-1.8  :   1.000E+00  1.737E+00  3.019E+00  5.247E+00  9.126E+00  1.588E+01  2.763E+01  4.809E+01  8.371E+01  1.457E+02  2.537E+02
-1.6  :   1.000E+00  1.509E+00  2.276E+00  3.436E+00  5.191E+00  7.850E+00  1.188E+01  1.798E+01  2.723E+01  4.126E+01  6.251E+01
-1.4  :   1.000E+00  1.311E+00  1.717E+00  2.251E+00  2.952E+00  3.877E+00  5.096E+00  6.704E+00  8.825E+00  1.162E+01  1.532E+01
-1.2  :   1.000E+00  1.142E+00  1.302E+00  1.484E+00  1.693E+00  1.932E+00  2.206E+00  2.520E+00  2.882E+00  3.296E+00  3.773E+00
-1    :   1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00
-.8   :   1.000E+00  8.851E-01  7.883E-01  7.059E-01  6.351E-01  5.739E-01  5.208E-01  4.747E-01  4.346E-01  3.997E-01  3.693E-01
-.6   :   1.000E+00  7.965E-01  6.489E-01  5.405E-01  4.600E-01  3.996E-01  3.541E-01  3.198E-01  2.938E-01  2.741E-01  2.592E-01
-.4   :   1.000E+00  7.331E-01  5.666E-01  4.614E-01  3.939E-01  3.502E-01  3.216E-01  3.028E-01  2.904E-01  2.823E-01  2.769E-01
-.2   :   1.000E+00  6.931E-01  5.284E-01  4.390E-01  3.896E-01  3.620E-01  3.463E-01  3.374E-01  3.323E-01  3.294E-01  3.277E-01
 0    :   1.000E+00  6.738E-01  5.230E-01  4.526E-01  4.191E-01  4.030E-01  3.952E-01  3.913E-01  3.893E-01  3.884E-01  3.879E-01
  .2  :   1.000E+00  6.719E-01  5.405E-01  4.876E-01  4.661E-01  4.572E-01  4.534E-01  4.518E-01  4.511E-01  4.508E-01  4.507E-01
  .4  :   1.000E+00  6.838E-01  5.732E-01  5.347E-01  5.212E-01  5.164E-01  5.146E-01  5.140E-01  5.137E-01  5.136E-01  5.136E-01
  .6  :   1.000E+00  7.061E-01  6.155E-01  5.878E-01  5.793E-01  5.767E-01  5.759E-01  5.756E-01  5.756E-01  5.755E-01  5.755E-01
  .8  :   1.000E+00  7.361E-01  6.634E-01  6.436E-01  6.382E-01  6.367E-01  6.363E-01  6.362E-01  6.361E-01  6.361E-01  6.361E-01
  1   :   1.000E+00  7.717E-01  7.147E-01  7.004E-01  6.968E-01  6.959E-01  6.957E-01  6.957E-01  6.957E-01  6.957E-01  6.957E-01
  1.2 :   1.000E+00  8.115E-01  7.680E-01  7.578E-01  7.554E-01  7.548E-01  7.546E-01  7.546E-01  7.546E-01  7.546E-01  7.546E-01
  1.4 :   1.000E+00  8.546E-01  8.230E-01  8.159E-01  8.142E-01  8.138E-01  8.137E-01  8.137E-01  8.137E-01  8.137E-01  8.137E-01
  1.6 :   1.000E+00  9.005E-01  8.797E-01  8.751E-01  8.741E-01  8.738E-01  8.738E-01  8.737E-01  8.737E-01  8.737E-01  8.737E-01
  1.8 :   1.000E+00  9.489E-01  9.386E-01  9.363E-01  9.357E-01  9.356E-01  9.356E-01  9.356E-01  9.356E-01  9.356E-01  9.356E-01
  2   :   1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00
```

B3(2,M,r,mu) for r = 16

Mu \ M=		1	2	4	8	16	32	64	128	256	512	1024
-2	:	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8	:	1.000E+00	1.740E+00	3.028E+00	5.269E+00	9.171E+00	1.596E+01	2.779E+01	4.838E+01	8.424E+01	1.467E+02	2.553E+02
-1.6	:	1.000E+00	1.513E+00	2.290E+00	3.467E+00	5.249E+00	7.950E+00	1.204E+01	1.825E+01	2.765E+01	4.191E+01	6.351E+01
-1.4	:	1.000E+00	1.316E+00	1.732E+00	2.280E+00	3.001E+00	3.953E+00	5.209E+00	6.865E+00	9.051E+00	1.194E+01	1.574E+01
-1.2	:	1.000E+00	1.146E+00	1.312E+00	1.502E+00	1.720E+00	1.970E+00	2.257E+00	2.587E+00	2.966E+00	3.401E+00	3.901E+00
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.000E+00	8.788E-01	7.752E-01	6.860E-01	6.089E-01	5.420E-01	4.839E-01	4.334E-01	3.894E-01	3.511E-01	3.178E-01
-.6	:	1.000E+00	7.826E-01	6.220E-01	5.026E-01	4.132E-01	3.458E-01	2.949E-01	2.564E-01	2.273E-01	2.052E-01	1.885E-01
-.4	:	1.000E+00	7.119E-01	5.287E-01	4.112E-01	3.350E-01	2.853E-01	2.527E-01	2.313E-01	2.172E-01	2.078E-01	2.017E-01
-.2	:	1.000E+00	6.668E-01	4.845E-01	3.838E-01	3.275E-01	2.958E-01	2.777E-01	2.674E-01	2.615E-01	2.581E-01	2.562E-01
0	:	1.000E+00	6.455E-01	4.785E-01	3.992E-01	3.612E-01	3.426E-01	3.335E-01	3.290E-01	3.267E-01	3.256E-01	3.250E-01
.2	:	1.000E+00	6.446E-01	5.001E-01	4.410E-01	4.167E-01	4.065E-01	4.021E-01	4.003E-01	3.995E-01	3.991E-01	3.990E-01
.4	:	1.000E+00	6.597E-01	5.395E-01	4.970E-01	4.819E-01	4.765E-01	4.745E-01	4.738E-01	4.735E-01	4.734E-01	4.734E-01
.6	:	1.000E+00	6.865E-01	5.891E-01	5.591E-01	5.498E-01	5.469E-01	5.460E-01	5.458E-01	5.457E-01	5.456E-01	5.456E-01
.8	:	1.000E+00	7.210E-01	6.439E-01	6.227E-01	6.169E-01	6.154E-01	6.149E-01	6.148E-01	6.148E-01	6.148E-01	6.148E-01
1	:	1.000E+00	7.606E-01	7.008E-01	6.858E-01	6.821E-01	6.812E-01	6.809E-01	6.809E-01	6.809E-01	6.809E-01	6.809E-01
1.2	:	1.000E+00	8.038E-01	7.586E-01	7.480E-01	7.455E-01	7.449E-01	7.447E-01	7.447E-01	7.447E-01	7.447E-01	7.447E-01
1.4	:	1.000E+00	8.495E-01	8.169E-01	8.096E-01	8.079E-01	8.075E-01	8.074E-01	8.074E-01	8.074E-01	8.074E-01	8.074E-01
1.6	:	1.000E+00	8.974E-01	8.762E-01	8.715E-01	8.704E-01	8.701E-01	8.701E-01	8.701E-01	8.701E-01	8.701E-01	8.701E-01
1.8	:	1.000E+00	9.475E-01	9.370E-01	9.346E-01	9.341E-01	9.340E-01	9.339E-01	9.339E-01	9.339E-01	9.339E-01	9.339E-01
2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 32

Mu \ M=		1	2	4	8	16	32	64	128	256	512	1024
-2	:	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8	:	1.000E+00	1.741E+00	3.030E+00	5.276E+00	9.184E+00	1.599E+01	2.784E+01	4.847E+01	8.439E+01	1.469E+02	2.558E+02
-1.6	:	1.000E+00	1.515E+00	2.295E+00	3.477E+00	5.269E+00	7.984E+00	1.210E+01	1.834E+01	2.779E+01	4.212E+01	6.384E+01
-1.4	:	1.000E+00	1.318E+00	1.738E+00	2.291E+00	3.020E+00	3.982E+00	5.252E+00	6.927E+00	9.137E+00	1.205E+01	1.590E+01
-1.2	:	1.000E+00	1.147E+00	1.316E+00	1.510E+00	1.732E+00	1.987E+00	2.280E+00	2.616E+00	3.003E+00	3.447E+00	3.957E+00
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.000E+00	8.753E-01	7.677E-01	6.747E-01	5.941E-01	5.240E-01	4.630E-01	4.100E-01	3.639E-01	3.237E-01	2.887E-01
-.6	:	1.000E+00	7.738E-01	6.052E-01	4.789E-01	3.838E-01	3.120E-01	2.577E-01	2.166E-01	1.855E-01	1.619E-01	1.441E-01
-.4	:	1.000E+00	6.975E-01	5.030E-01	3.770E-01	2.950E-01	2.412E-01	2.059E-01	1.826E-01	1.673E-01	1.572E-01	1.505E-01
-.2	:	1.000E+00	6.480E-01	4.532E-01	3.445E-01	2.833E-01	2.486E-01	2.288E-01	2.175E-01	2.110E-01	2.073E-01	2.051E-01
0	:	1.000E+00	6.251E-01	4.465E-01	3.609E-01	3.194E-01	2.991E-01	2.891E-01	2.841E-01	2.816E-01	2.804E-01	2.798E-01
.2	:	1.000E+00	6.256E-01	4.718E-01	4.085E-01	3.821E-01	3.710E-01	3.663E-01	3.642E-01	3.634E-01	3.630E-01	3.628E-01
.4	:	1.000E+00	6.440E-01	5.173E-01	4.723E-01	4.561E-01	4.503E-01	4.481E-01	4.473E-01	4.471E-01	4.469E-01	4.469E-01
.6	:	1.000E+00	6.747E-01	5.733E-01	5.418E-01	5.321E-01	5.290E-01	5.280E-01	5.277E-01	5.276E-01	5.276E-01	5.276E-01
.8	:	1.000E+00	7.128E-01	6.333E-01	6.114E-01	6.054E-01	6.038E-01	6.033E-01	6.032E-01	6.032E-01	6.031E-01	6.031E-01
1	:	1.000E+00	7.553E-01	6.941E-01	6.788E-01	6.750E-01	6.740E-01	6.738E-01	6.737E-01	6.737E-01	6.737E-01	6.737E-01
1.2	:	1.000E+00	8.004E-01	7.545E-01	7.438E-01	7.412E-01	7.406E-01	7.405E-01	7.404E-01	7.404E-01	7.404E-01	7.404E-01
1.4	:	1.000E+00	8.475E-01	8.146E-01	8.072E-01	8.055E-01	8.051E-01	8.050E-01	8.050E-01	8.049E-01	8.049E-01	8.049E-01
1.6	:	1.000E+00	8.964E-01	8.750E-01	8.702E-01	8.691E-01	8.689E-01	8.688E-01	8.688E-01	8.688E-01	8.688E-01	8.688E-01
1.8	:	1.000E+00	9.471E-01	9.365E-01	9.341E-01	9.336E-01	9.334E-01	9.334E-01	9.334E-01	9.334E-01	9.334E-01	9.334E-01
2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 64

Mu \ M=		1	2	4	8	16	32	64	128	256	512	1024
-2	:	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8	:	1.000E+00	1.741E+00	3.031E+00	5.277E+00	9.188E+00	1.600E+01	2.785E+01	4.849E+01	8.443E+01	1.470E+02	2.559E+02
-1.6	:	1.000E+00	1.515E+00	2.297E+00	3.481E+00	5.275E+00	7.995E+00	1.212E+01	1.836E+01	2.784E+01	4.219E+01	6.395E+01
-1.4	:	1.000E+00	1.319E+00	1.740E+00	2.295E+00	3.027E+00	3.993E+00	5.268E+00	6.950E+00	9.170E+00	1.210E+01	1.596E+01
-1.2	:	1.000E+00	1.148E+00	1.318E+00	1.513E+00	1.737E+00	1.994E+00	2.290E+00	2.629E+00	3.019E+00	3.467E+00	3.981E+00
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.000E+00	8.733E-01	7.635E-01	6.683E-01	5.856E-01	5.137E-01	4.512E-01	3.967E-01	3.493E-01	3.081E-01	2.722E-01
-.6	:	1.000E+00	7.683E-01	5.945E-01	4.637E-01	3.650E-01	2.904E-01	2.340E-01	1.912E-01	1.588E-01	1.343E-01	1.157E-01
-.4	:	1.000E+00	6.875E-01	4.849E-01	3.531E-01	2.668E-01	2.102E-01	1.730E-01	1.485E-01	1.323E-01	1.216E-01	1.146E-01
-.2	:	1.000E+00	6.341E-01	4.299E-01	3.152E-01	2.504E-01	2.135E-01	1.924E-01	1.804E-01	1.734E-01	1.695E-01	1.672E-01
0	:	1.000E+00	6.098E-01	4.225E-01	3.320E-01	2.880E-01	2.663E-01	2.556E-01	2.503E-01	2.476E-01	2.463E-01	2.456E-01
.2	:	1.000E+00	6.116E-01	4.512E-01	3.846E-01	3.568E-01	3.450E-01	3.400E-01	3.378E-01	3.369E-01	3.365E-01	3.363E-01
.4	:	1.000E+00	6.332E-01	5.021E-01	4.553E-01	4.384E-01	4.323E-01	4.300E-01	4.292E-01	4.289E-01	4.288E-01	4.287E-01
.6	:	1.000E+00	6.673E-01	5.634E-01	5.310E-01	5.210E-01	5.178E-01	5.168E-01	5.165E-01	5.164E-01	5.164E-01	5.164E-01
.8	:	1.000E+00	7.082E-01	6.274E-01	6.051E-01	5.990E-01	5.973E-01	5.969E-01	5.967E-01	5.967E-01	5.967E-01	5.967E-01
1	:	1.000E+00	7.526E-01	6.908E-01	6.753E-01	6.714E-01	6.705E-01	6.702E-01	6.702E-01	6.702E-01	6.702E-01	6.702E-01
1.2	:	1.000E+00	7.990E-01	7.527E-01	7.419E-01	7.394E-01	7.388E-01	7.386E-01	7.386E-01	7.386E-01	7.386E-01	7.386E-01
1.4	:	1.000E+00	8.468E-01	8.137E-01	8.063E-01	8.045E-01	8.041E-01	8.040E-01	8.040E-01	8.040E-01	8.040E-01	8.040E-01
1.6	:	1.000E+00	8.960E-01	8.745E-01	8.698E-01	8.687E-01	8.684E-01	8.684E-01	8.684E-01	8.684E-01	8.684E-01	8.684E-01
1.8	:	1.000E+00	9.470E-01	9.363E-01	9.340E-01	9.334E-01	9.333E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01
2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 128

Mu \ M=		1	2	4	8	16	32	64	128	256	512	1024
-2	:	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8	:	1.000E+00	1.741E+00	3.031E+00	5.278E+00	9.189E+00	1.600E+01	2.786E+01	4.850E+01	8.444E+01	1.470E+02	2.560E+02
-1.6	:	1.000E+00	1.516E+00	2.297E+00	3.482E+00	5.277E+00	7.998E+00	1.212E+01	1.837E+01	2.785E+01	4.221E+01	6.398E+01
-1.4	:	1.000E+00	1.319E+00	1.741E+00	2.296E+00	3.030E+00	3.997E+00	5.274E+00	6.959E+00	9.182E+00	1.212E+01	1.599E+01
-1.2	:	1.000E+00	1.148E+00	1.319E+00	1.515E+00	1.739E+00	1.998E+00	2.294E+00	2.635E+00	3.026E+00	3.475E+00	3.992E+00
-1	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	:	1.000E+00	8.721E-01	7.611E-01	6.647E-01	5.808E-01	5.079E-01	4.444E-01	3.891E-01	3.410E-01	2.992E-01	2.627E-01
-.6	:	1.000E+00	7.647E-01	5.875E-01	4.539E-01	3.529E-01	2.764E-01	2.186E-01	1.747E-01	1.415E-01	1.164E-01	9.729E-02
-.4	:	1.000E+00	6.803E-01	4.720E-01	3.359E-01	2.467E-01	1.881E-01	1.494E-01	1.240E-01	1.072E-01	9.614E-02	8.884E-02
-.2	:	1.000E+00	6.235E-01	4.121E-01	2.928E-01	2.252E-01	1.866E-01	1.646E-01	1.519E-01	1.447E-01	1.405E-01	1.381E-01
0	:	1.000E+00	5.978E-01	4.036E-01	3.094E-01	2.634E-01	2.407E-01	2.294E-01	2.238E-01	2.210E-01	2.196E-01	2.189E-01
.2	:	1.000E+00	6.011E-01	4.355E-01	3.665E-01	3.376E-01	3.253E-01	3.200E-01	3.177E-01	3.168E-01	3.163E-01	3.162E-01
.4	:	1.000E+00	6.256E-01	4.914E-01	4.433E-01	4.259E-01	4.196E-01	4.173E-01	4.164E-01	4.161E-01	4.160E-01	4.159E-01
.6	:	1.000E+00	6.626E-01	5.571E-01	5.242E-01	5.139E-01	5.107E-01	5.097E-01	5.094E-01	5.093E-01	5.092E-01	5.092E-01
.8	:	1.000E+00	7.056E-01	6.240E-01	6.016E-01	5.954E-01	5.937E-01	5.932E-01	5.931E-01	5.930E-01	5.930E-01	5.930E-01
1	:	1.000E+00	7.513E-01	6.891E-01	6.736E-01	6.697E-01	6.687E-01	6.685E-01	6.684E-01	6.684E-01	6.684E-01	6.684E-01
1.2	:	1.000E+00	7.983E-01	7.520E-01	7.411E-01	7.386E-01	7.380E-01	7.378E-01	7.378E-01	7.378E-01	7.378E-01	7.378E-01
1.4	:	1.000E+00	8.465E-01	8.133E-01	8.059E-01	8.042E-01	8.038E-01	8.037E-01	8.037E-01	8.037E-01	8.037E-01	8.037E-01
1.6	:	1.000E+00	8.959E-01	8.744E-01	8.696E-01	8.686E-01	8.683E-01	8.682E-01	8.682E-01	8.682E-01	8.682E-01	8.682E-01
1.8	:	1.000E+00	9.469E-01	9.363E-01	9.339E-01	9.334E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01
2	:	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 256

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8	1.000E+00	1.741E+00	3.031E+00	5.278E+00	9.189E+00	1.600E+01	2.786E+01	4.850E+01	8.445E+01	1.470E+02	2.560E+02
-1.6	1.000E+00	1.516E+00	2.297E+00	3.482E+00	5.278E+00	7.999E+00	1.212E+01	1.838E+01	2.786E+01	4.222E+01	6.399E+01
-1.4	1.000E+00	1.319E+00	1.741E+00	2.297E+00	3.031E+00	3.999E+00	5.277E+00	6.962E+00	9.187E+00	1.212E+01	1.599E+01
-1.2	1.000E+00	1.149E+00	1.319E+00	1.515E+00	1.740E+00	1.999E+00	2.296E+00	2.637E+00	3.029E+00	3.479E+00	3.996E+00
-1	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	1.000E+00	8.714E-01	7.597E-01	6.626E-01	5.781E-01	5.045E-01	4.405E-01	3.848E-01	3.363E-01	2.941E-01	2.573E-01
-.6	1.000E+00	7.623E-01	5.830E-01	4.475E-01	3.450E-01	2.674E-01	2.086E-01	1.640E-01	1.303E-01	1.047E-01	8.533E-02
-.4	1.000E+00	6.750E-01	4.626E-01	3.235E-01	2.321E-01	1.720E-01	1.323E-01	1.062E-01	8.899E-02	7.763E-02	7.014E-02
-.2	1.000E+00	6.152E-01	3.981E-01	2.753E-01	2.054E-01	1.655E-01	1.427E-01	1.296E-01	1.221E-01	1.178E-01	1.153E-01
0	1.000E+00	5.882E-01	3.885E-01	2.913E-01	2.436E-01	2.201E-01	2.084E-01	2.026E-01	1.997E-01	1.982E-01	1.975E-01
.2	1.000E+00	5.929E-01	4.233E-01	3.525E-01	3.226E-01	3.099E-01	3.045E-01	3.022E-01	3.011E-01	3.007E-01	3.005E-01
.4	1.000E+00	6.202E-01	4.837E-01	4.347E-01	4.170E-01	4.105E-01	4.081E-01	4.072E-01	4.069E-01	4.068E-01	4.067E-01
.6	1.000E+00	6.596E-01	5.530E-01	5.198E-01	5.094E-01	5.061E-01	5.051E-01	5.047E-01	5.046E-01	5.046E-01	5.046E-01
.8	1.000E+00	7.042E-01	6.221E-01	5.995E-01	5.933E-01	5.916E-01	5.911E-01	5.910E-01	5.910E-01	5.910E-01	5.910E-01
1	1.000E+00	7.507E-01	6.883E-01	6.727E-01	6.688E-01	6.679E-01	6.676E-01	6.676E-01	6.675E-01	6.675E-01	6.675E-01
1.2	1.000E+00	7.981E-01	7.516E-01	7.408E-01	7.382E-01	7.376E-01	7.375E-01	7.374E-01	7.374E-01	7.374E-01	7.374E-01
1.4	1.000E+00	8.464E-01	8.132E-01	8.058E-01	8.041E-01	8.036E-01	8.036E-01	8.035E-01	8.035E-01	8.035E-01	8.035E-01
1.6	1.000E+00	8.958E-01	8.743E-01	8.696E-01	8.685E-01	8.682E-01	8.682E-01	8.682E-01	8.682E-01	8.682E-01	8.682E-01
1.8	1.000E+00	9.469E-01	9.362E-01	9.339E-01	9.333E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01
2	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 512

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8	1.000E+00	1.741E+00	3.031E+00	5.278E+00	9.190E+00	1.600E+01	2.786E+01	4.850E+01	8.445E+01	1.470E+02	2.560E+02
-1.6	1.000E+00	1.516E+00	2.297E+00	3.482E+00	5.278E+00	8.000E+00	1.213E+01	1.838E+01	2.786E+01	4.222E+01	6.400E+01
-1.4	1.000E+00	1.319E+00	1.741E+00	2.297E+00	3.031E+00	4.000E+00	5.277E+00	6.964E+00	9.189E+00	1.212E+01	1.600E+01
-1.2	1.000E+00	1.149E+00	1.319E+00	1.516E+00	1.741E+00	2.000E+00	2.297E+00	2.638E+00	3.030E+00	3.481E+00	3.998E+00
-1	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8	1.000E+00	8.711E-01	7.589E-01	6.614E-01	5.765E-01	5.026E-01	4.383E-01	3.823E-01	3.336E-01	2.911E-01	2.542E-01
-.6	1.000E+00	7.608E-01	5.800E-01	4.433E-01	3.398E-01	2.614E-01	2.020E-01	1.570E-01	1.229E-01	9.709E-02	7.751E-02
-.4	1.000E+00	6.712E-01	4.557E-01	3.143E-01	2.213E-01	1.601E-01	1.198E-01	9.316E-02	7.561E-02	6.403E-02	5.639E-02
-.2	1.000E+00	6.085E-01	3.870E-01	2.613E-01	1.897E-01	1.487E-01	1.253E-01	1.118E-01	1.041E-01	9.970E-02	9.716E-02
0	1.000E+00	5.803E-01	3.761E-01	2.764E-01	2.274E-01	2.031E-01	1.911E-01	1.851E-01	1.821E-01	1.806E-01	1.799E-01
.2	1.000E+00	5.863E-01	4.136E-01	3.413E-01	3.108E-01	2.978E-01	2.922E-01	2.898E-01	2.887E-01	2.883E-01	2.881E-01
.4	1.000E+00	6.162E-01	4.781E-01	4.284E-01	4.104E-01	4.039E-01	4.014E-01	4.005E-01	4.002E-01	4.001E-01	4.000E-01
.6	1.000E+00	6.576E-01	5.504E-01	5.169E-01	5.064E-01	5.031E-01	5.021E-01	5.018E-01	5.016E-01	5.016E-01	5.016E-01
.8	1.000E+00	7.033E-01	6.211E-01	5.984E-01	5.921E-01	5.904E-01	5.899E-01	5.898E-01	5.898E-01	5.898E-01	5.898E-01
1	1.000E+00	7.503E-01	6.879E-01	6.723E-01	6.684E-01	6.674E-01	6.672E-01	6.671E-01	6.671E-01	6.671E-01	6.671E-01
1.2	1.000E+00	7.980E-01	7.515E-01	7.406E-01	7.381E-01	7.375E-01	7.373E-01	7.373E-01	7.373E-01	7.373E-01	7.373E-01
1.4	1.000E+00	8.463E-01	8.132E-01	8.057E-01	8.040E-01	8.036E-01	8.035E-01	8.035E-01	8.035E-01	8.035E-01	8.035E-01
1.6	1.000E+00	8.958E-01	8.743E-01	8.696E-01	8.685E-01	8.682E-01	8.682E-01	8.681E-01	8.681E-01	8.681E-01	8.681E-01
1.8	1.000E+00	9.469E-01	9.362E-01	9.339E-01	9.333E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01
2	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 1024

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2 :	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8 :	1.000E+00	1.741E+00	3.031E+00	5.278E+00	9.190E+00	1.600E+01	2.786E+01	4.850E+01	8.445E+01	1.470E+02	2.560E+02
-1.6 :	1.000E+00	1.516E+00	2.297E+00	3.482E+00	5.278E+00	8.000E+00	1.213E+01	1.838E+01	2.786E+01	4.222E+01	6.400E+01
-1.4 :	1.000E+00	1.319E+00	1.741E+00	2.297E+00	3.031E+00	4.000E+00	5.278E+00	6.964E+00	9.189E+00	1.213E+01	1.600E+01
-1.2 :	1.000E+00	1.149E+00	1.319E+00	1.516E+00	1.741E+00	2.000E+00	2.297E+00	2.639E+00	3.031E+00	3.482E+00	3.999E+00
-1 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8 :	1.000E+00	8.708E-01	7.585E-01	6.607E-01	5.756E-01	5.015E-01	4.370E-01	3.809E-01	3.320E-01	2.894E-01	2.524E-01
-.6 :	1.000E+00	7.598E-01	5.781E-01	4.406E-01	3.364E-01	2.575E-01	1.977E-01	1.524E-01	1.181E-01	9.210E-02	7.238E-02
-.4 :	1.000E+00	6.683E-01	4.506E-01	3.076E-01	2.134E-01	1.514E-01	1.105E-01	8.348E-02	6.569E-02	5.396E-02	4.622E-02
-.2 :	1.000E+00	6.031E-01	3.780E-01	2.499E-01	1.769E-01	1.351E-01	1.112E-01	9.744E-02	8.956E-02	8.504E-02	8.244E-02
0 :	1.000E+00	5.737E-01	3.658E-01	2.639E-01	2.138E-01	1.890E-01	1.767E-01	1.705E-01	1.675E-01	1.659E-01	1.652E-01
.2 :	1.000E+00	5.811E-01	4.059E-01	3.323E-01	3.012E-01	2.880E-01	2.823E-01	2.798E-01	2.788E-01	2.783E-01	2.781E-01
.4 :	1.000E+00	6.133E-01	4.740E-01	4.238E-01	4.056E-01	3.990E-01	3.965E-01	3.956E-01	3.953E-01	3.951E-01	3.951E-01
.6 :	1.000E+00	6.564E-01	5.486E-01	5.150E-01	5.045E-01	5.012E-01	5.001E-01	4.998E-01	4.997E-01	4.997E-01	4.996E-01
.8 :	1.000E+00	7.028E-01	6.204E-01	5.977E-01	5.915E-01	5.897E-01	5.893E-01	5.891E-01	5.891E-01	5.891E-01	5.891E-01
1 :	1.000E+00	7.502E-01	6.877E-01	6.721E-01	6.682E-01	6.672E-01	6.670E-01	6.669E-01	6.669E-01	6.669E-01	6.669E-01
1.2 :	1.000E+00	7.979E-01	7.514E-01	7.406E-01	7.380E-01	7.374E-01	7.373E-01	7.372E-01	7.372E-01	7.372E-01	7.372E-01
1.4 :	1.000E+00	8.463E-01	8.131E-01	8.057E-01	8.040E-01	8.036E-01	8.035E-01	8.035E-01	8.035E-01	8.035E-01	8.035E-01
1.6 :	1.000E+00	8.958E-01	8.743E-01	8.696E-01	8.685E-01	8.682E-01	8.682E-01	8.681E-01	8.681E-01	8.681E-01	8.681E-01
1.8 :	1.000E+00	9.469E-01	9.362E-01	9.339E-01	9.333E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01
2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.008E+00	1.000E+00	1.000E+00	1.000E+00

B3(2,M,r,mu) for r = 2048

Mu \ M=	1	2	4	8	16	32	64	128	256	512	1024
-2 :	1.000E+00	2.000E+00	4.000E+00	8.000E+00	1.600E+01	3.200E+01	6.400E+01	1.280E+02	2.560E+02	5.120E+02	1.024E+03
-1.8 :	1.000E+00	1.741E+00	3.031E+00	5.278E+00	9.190E+00	1.600E+01	2.786E+01	4.850E+01	8.445E+01	1.470E+02	2.560E+02
-1.6 :	1.000E+00	1.516E+00	2.297E+00	3.482E+00	5.278E+00	8.000E+00	1.213E+01	1.838E+01	2.786E+01	4.222E+01	6.400E+01
-1.4 :	1.000E+00	1.320E+00	1.741E+00	2.297E+00	3.031E+00	4.000E+00	5.278E+00	6.964E+00	9.189E+00	1.213E+01	1.600E+01
-1.2 :	1.000E+00	1.149E+00	1.319E+00	1.516E+00	1.741E+00	2.000E+00	2.297E+00	2.639E+00	3.031E+00	3.482E+00	4.000E+00
-1 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00
-.8 :	1.000E+00	8.707E-01	7.582E-01	6.603E-01	5.751E-01	5.009E-01	4.363E-01	3.800E-01	3.311E-01	2.885E-01	2.514E-01
-.6 :	1.000E+00	7.591E-01	5.768E-01	4.388E-01	3.342E-01	2.549E-01	1.949E-01	1.494E-01	1.149E-01	8.881E-02	6.901E-02
-.4 :	1.000E+00	6.662E-01	4.468E-01	3.025E-01	2.075E-01	1.448E-01	1.035E-01	7.628E-02	5.831E-02	4.646E-02	3.864E-02
-.2 :	1.000E+00	5.987E-01	3.706E-01	2.407E-01	1.664E-01	1.240E-01	9.962E-02	8.565E-02	7.763E-02	7.303E-02	7.038E-02
0 :	1.000E+00	5.681E-01	3.570E-01	2.534E-01	2.023E-01	1.770E-01	1.644E-01	1.582E-01	1.551E-01	1.535E-01	1.527E-01
.2 :	1.000E+00	5.768E-01	3.995E-01	3.250E-01	2.934E-01	2.800E-01	2.742E-01	2.717E-01	2.706E-01	2.701E-01	2.699E-01
.4 :	1.000E+00	6.111E-01	4.709E-01	4.204E-01	4.021E-01	3.953E-01	3.929E-01	3.920E-01	3.916E-01	3.915E-01	3.914E-01
.6 :	1.000E+00	6.555E-01	5.475E-01	5.138E-01	5.032E-01	4.999E-01	4.988E-01	4.985E-01	4.984E-01	4.984E-01	4.984E-01
.8 :	1.000E+00	7.026E-01	6.201E-01	5.973E-01	5.911E-01	5.894E-01	5.889E-01	5.888E-01	5.887E-01	5.887E-01	5.887E-01
1 :	1.000E+00	7.501E-01	6.876E-01	6.720E-01	6.681E-01	6.671E-01	6.669E-01	6.668E-01	6.668E-01	6.668E-01	6.668E-01
1.2 :	1.000E+00	7.979E-01	7.514E-01	7.405E-01	7.380E-01	7.374E-01	7.372E-01	7.372E-01	7.372E-01	7.372E-01	7.372E-01
1.4 :	1.000E+00	8.463E-01	8.131E-01	8.057E-01	8.040E-01	8.036E-01	8.035E-01	8.035E-01	8.034E-01	8.034E-01	8.034E-01
1.6 :	1.000E+00	8.958E-01	8.743E-01	8.696E-01	8.685E-01	8.682E-01	8.682E-01	8.681E-01	8.681E-01	8.681E-01	8.681E-01
1.8 :	1.000E+00	9.469E-01	9.362E-01	9.339E-01	9.333E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01	9.332E-01
2 :	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00	1.000E+00

```
                B3(2,M,r,mu)  for  r =  4096

Mu \  M=  1          2          4          8          16         32         64         128        256        512        1024
-------------------------------------------------------------------------------------------------------------------------------
-2    :  1.000E+00  2.000E+00  4.000E+00  8.000E+00  1.600E+01  3.200E+01  6.400E+01  1.280E+02  2.560E+02  5.120E+02  1.024E+03
-1.8  :  1.000E+00  1.741E+00  3.031E+00  5.278E+00  9.190E+00  1.600E+01  2.786E+01  4.850E+01  8.445E+01  1.470E+02  2.560E+02
-1.6  :  1.000E+00  1.516E+00  2.297E+00  3.482E+00  5.278E+00  8.000E+00  1.213E+01  1.838E+01  2.786E+01  4.222E+01  6.400E+01
-1.4  :  1.000E+00  1.320E+00  1.741E+00  2.297E+00  3.031E+00  4.000E+00  5.278E+00  6.964E+00  9.190E+00  1.213E+01  1.600E+01
-1.2  :  1.000E+00  1.149E+00  1.319E+00  1.516E+00  1.741E+00  2.000E+00  2.297E+00  2.639E+00  3.031E+00  3.482E+00  4.000E+00
-1    :  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00
-.8   :  1.000E+00  8.706E-01  7.581E-01  6.601E-01  5.748E-01  5.005E-01  4.358E-01  3.796E-01  3.306E-01  2.879E-01  2.508E-01
-.6   :  1.000E+00  7.587E-01  5.760E-01  4.376E-01  3.327E-01  2.533E-01  1.931E-01  1.474E-01  1.129E-01  8.665E-02  6.679E-02
-.4   :  1.000E+00  6.646E-01  4.440E-01  2.987E-01  2.030E-01  1.399E-01  9.833E-02  7.089E-02  5.279E-02  4.085E-02  3.297E-02
-.2   :  1.000E+00  5.950E-01  3.645E-01  2.330E-01  1.578E-01  1.148E-01  9.007E-02  7.590E-02  6.777E-02  6.310E-02  6.042E-02
 0    :  1.000E+00  5.633E-01  3.494E-01  2.443E-01  1.925E-01  1.667E-01  1.539E-01  1.476E-01  1.444E-01  1.428E-01  1.420E-01
  .2  :  1.000E+00  5.733E-01  3.943E-01  3.189E-01  2.870E-01  2.734E-01  2.675E-01  2.650E-01  2.639E-01  2.634E-01  2.632E-01
  .4  :  1.000E+00  6.095E-01  4.686E-01  4.178E-01  3.994E-01  3.926E-01  3.902E-01  3.892E-01  3.889E-01  3.887E-01  3.887E-01
  .6  :  1.000E+00  6.550E-01  5.468E-01  5.130E-01  5.024E-01  4.991E-01  4.980E-01  4.977E-01  4.976E-01  4.975E-01  4.975E-01
  .8  :  1.000E+00  7.024E-01  6.199E-01  5.971E-01  5.909E-01  5.891E-01  5.887E-01  5.885E-01  5.885E-01  5.885E-01  5.885E-01
  1   :  1.000E+00  7.500E-01  6.876E-01  6.719E-01  6.680E-01  6.670E-01  6.668E-01  6.667E-01  6.667E-01  6.667E-01  6.667E-01
  1.2 :  1.000E+00  7.979E-01  7.514E-01  7.405E-01  7.380E-01  7.374E-01  7.372E-01  7.372E-01  7.372E-01  7.372E-01  7.372E-01
  1.4 :  1.000E+00  8.463E-01  8.131E-01  8.057E-01  8.040E-01  8.036E-01  8.035E-01  8.034E-01  8.034E-01  8.034E-01  8.034E-01
  1.6 :  1.000E+00  8.958E-01  8.743E-01  8.696E-01  8.685E-01  8.682E-01  8.682E-01  8.681E-01  8.681E-01  8.681E-01  8.681E-01
  1.8 :  1.000E+00  9.469E-01  9.362E-01  9.339E-01  9.333E-01  9.332E-01  9.332E-01  9.332E-01  9.332E-01  9.332E-01  9.332E-01
  2   :  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00
```

```
                B3(2,M,r,mu)  for  r =  8192

Mu \  M=  1          2          4          8          16         32         64         128        256        512        1024
-------------------------------------------------------------------------------------------------------------------------------
-2    :  1.000E+00  2.000E+00  4.000E+00  8.000E+00  1.600E+01  3.200E+01  6.400E+01  1.280E+02  2.560E+02  5.120E+02  1.024E+03
-1.8  :  1.000E+00  1.741E+00  3.031E+00  5.278E+00  9.190E+00  1.600E+01  2.786E+01  4.850E+01  8.445E+01  1.470E+02  2.560E+02
-1.6  :  1.000E+00  1.516E+00  2.297E+00  3.482E+00  5.278E+00  8.000E+00  1.213E+01  1.838E+01  2.786E+01  4.222E+01  6.400E+01
-1.4  :  1.000E+00  1.320E+00  1.741E+00  2.297E+00  3.031E+00  4.000E+00  5.278E+00  6.964E+00  9.190E+00  1.213E+01  1.600E+01
-1.2  :  1.000E+00  1.149E+00  1.320E+00  1.516E+00  1.741E+00  2.000E+00  2.297E+00  2.639E+00  3.031E+00  3.482E+00  4.000E+00
-1    :  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00
-.8   :  1.000E+00  8.706E-01  7.580E-01  6.599E-01  5.746E-01  5.003E-01  4.356E-01  3.793E-01  3.303E-01  2.876E-01  2.505E-01
-.6   :  1.000E+00  7.584E-01  5.754E-01  4.368E-01  3.317E-01  2.522E-01  1.918E-01  1.461E-01  1.115E-01  8.523E-02  6.533E-02
-.4   :  1.000E+00  6.634E-01  4.418E-01  2.959E-01  1.997E-01  1.363E-01  9.444E-02  6.685E-02  4.864E-02  3.663E-02  2.871E-02
-.2   :  1.000E+00  5.920E-01  3.594E-01  2.266E-01  1.506E-01  1.071E-01  8.211E-02  6.778E-02  5.955E-02  5.483E-02  5.212E-02
 0    :  1.000E+00  5.591E-01  3.429E-01  2.365E-01  1.839E-01  1.578E-01  1.448E-01  1.383E-01  1.351E-01  1.335E-01  1.327E-01
  .2  :  1.000E+00  5.704E-01  3.899E-01  3.139E-01  2.817E-01  2.679E-01  2.620E-01  2.594E-01  2.583E-01  2.578E-01  2.576E-01
  .4  :  1.000E+00  6.083E-01  4.669E-01  4.159E-01  3.974E-01  3.906E-01  3.881E-01  3.872E-01  3.868E-01  3.867E-01  3.867E-01
  .6  :  1.000E+00  6.546E-01  5.463E-01  5.124E-01  5.018E-01  4.985E-01  4.975E-01  4.971E-01  4.970E-01  4.970E-01  4.970E-01
  .8  :  1.000E+00  7.023E-01  6.198E-01  5.970E-01  5.907E-01  5.890E-01  5.885E-01  5.884E-01  5.884E-01  5.884E-01  5.884E-01
  1   :  1.000E+00  7.500E-01  6.875E-01  6.719E-01  6.680E-01  6.670E-01  6.668E-01  6.667E-01  6.667E-01  6.667E-01  6.667E-01
  1.2 :  1.000E+00  7.979E-01  7.514E-01  7.405E-01  7.380E-01  7.374E-01  7.372E-01  7.372E-01  7.372E-01  7.372E-01  7.372E-01
  1.4 :  1.000E+00  8.463E-01  8.131E-01  8.057E-01  8.040E-01  8.036E-01  8.035E-01  8.034E-01  8.034E-01  8.034E-01  8.034E-01
  1.6 :  1.000E+00  8.958E-01  8.743E-01  8.696E-01  8.685E-01  8.682E-01  8.682E-01  8.681E-01  8.681E-01  8.681E-01  8.681E-01
  1.8 :  1.000E+00  9.469E-01  9.362E-01  9.339E-01  9.333E-01  9.332E-01  9.332E-01  9.332E-01  9.332E-01  9.332E-01  9.332E-01
  2   :  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00  1.000E+00
```

E. APPENDIX - Notes and Errata

The notes listed below are included to bring attention to notation and other problems. The page number provides reference to the location of the problem or comment.

1. page TN-4

AM noise is especially important in the measurement of the (residual) phase noise added by amplifiers and other signal handling components. Often the driving source for the measurements is a frequency synthesizer that has phase and/or amplitude noise that is comparable or larger than the added noise of the component under test. Most measurement systems are configured so that the phase noise of the source cancels out to a large degree (at high Fourier frequencies decorrelation effects limit the cancellation). In such measurement systems, the AM to PM conversion factor and the AM noise of the source may then set the noise floor. Discussion of the effect of AM noise and AM to PM conversion factors on the accuracy and precision of phase noise measurements is found in "Residual Phase Noise Measurements of vhf, uhf, and Microwave Components" by G. K. Montress, T. E. Parker, and M. J. Loboda Proc. *43rd Annual Symposium on Frequency Control*, pp. 349-359 (1989) and in "Accuracy Model for Phase Noise Measurements by F. L. Walls, C. M. Felton, and A. J. D. Clements, *21st Annual Precision Time and Time Interval Meeting* (1989). The notation in these papers as well as that in other parts of the literature differs from that given below. Our notation below is drawn from a modest level of consensus among individuals responsible for setting standards. We expect that it will gradually be adopted within standards committees. The following comments are directed specifically at the specification of noise performance.

The total power spectral density in a signal can be approximated by expanding eq 12-5 of paper B.2 (by Stein) and extending it to include the spectral density of relative amplitude fluctuations, $S_a(f)$. The double-sideband density written in single-sideband form is given by

$$S_V(f) \simeq \frac{V_0^2}{2}\left[e^{-I(f_c)}\delta(f) + e^{-I(f)}S_\phi(f) + S_a(f)\right] \qquad 0 < f < \infty,$$

where

$$I(f) \simeq \int_f^\infty S_\phi(f)df.$$

I(f) is the integrated phase modulation due to the pedestal and $\delta(f)$ represents the carrier with frequency width $\pm f_c$. The effect of large $S_a(f)$ on power in the carrier has not, to our knowledge, been explored. The power spectral density of relative phase fluctuations, $S_\phi(f)$, is normalized to one rad^2/Hz and $S_a(f)$ is normalized to the carrier voltage, but the total power spectral density, $S_V(f)$ is not normalized and has the units of V^2/Hz. All of these are single-sided spectral densities. For most measurement purposes, we can disregard the carrier and find that, away from the carrier, the above expression simplifies to

$$S_V(f) = \frac{V_0^2}{2}\left[S_\phi(f) + S_a(f)\right] \qquad for \qquad 0 \leq f \leq \infty.$$

The more general expression is important only very near the carrier and in certain types of frequency multiplication. The single-sideband, amplitude noise, normalized to the total signal power is given simply by $S_a(f)$ for $0 \leq f \leq \infty$. The measurement of added phase and amplitude noise for amplifiers and other signal handling components should specify the signal level since the AM noise level and the contribution due to AM-to-PM conversion depend on the signal level.

2. page TN-6
   For a direct measurement, time accuracy only has meaning when the phase of the time-base oscillator of the frequency counter is known with respect to some time standard. Either it is phase locked or is calibrated with respect to that standard at the time of measurement. The phase of the time-base oscillator can then be measured with respect to the phase of the frequency standard being calibrated (accounting for cable delays, etc.). Except for the cycle ambiguity of the carrier, the phase of the frequency standard being measured can carry time information and have time accuracy. This technique is not common, but is very useful and eliminates divider noise that typically occurs in going from 5 or 10 Mhz to 1 pulse per second. Caution must be exercised to assure that the phase point measured in a sine wave is at a reproducible voltage and impedance so that the cycle ambiguity is an exact integer.

3. page TN-35
   A second-order servo loop provides substantially enhanced performance. See, for example, F.L. Walls and S.R. Stein, "Servo techniques in oscillators and measurement systems," *NBS Tech. Note 692* (1976).

4. page TN-35
   This error can be identified and corrected using the phase modulation scheme described in paper B.4 on page TN-136.

5. page TN-36
   Low-noise DC amplifiers have been substantially improved since publication of this paper.

6. pages TN-37, TN-91, TN-130, TN-174, TN-206 and TN-218
   The reader is reminded that the discussions of frequency-domain measurements assume incoherent noise processes. Often the phase noise spectrum of a signal will contain bright spectral features (spurious lines) other than the carrier. Frequency-domain measurements are often useful in identifying such features. But if these spurious lines are narrow compared to the measurement bandwidth, statistical measures such as $S_\phi(f)$ and $\mathcal{L}(f)$ are not appropriate. It is better to specify the phase deviation in terms of the rms value of the phase deviations, ϕ_{rms}(rms radians), without reference to bandwidth. This specification in rms radians can be related to the Allan variance (see note # 8 below).

7. page TN-51
   Humidity is often an important environmental factor. See, for example, J.E. Gray, H.E. Machlan and D.W. Allan, "The Effect of humidity on commercial cesium beam atomic clocks," *42nd Annual Symp. on Frequency Control*, pp. 514-518 (1988) and F.L. Walls, "The

Influence of Pressure and Humidity on the Medium and Long-Term Frequency Stability of Quartz Oscillators," *42nd Annual Symp. on Frequency Control*, pp. 279-283 (1988).

8. page TN-74

For a bright line (one which is narrow compared with the measurement bandwidth), the solution of eq 12-27 simplifies to

$$\sigma_y(\tau) = \frac{\sqrt{2}\,\phi_{rms}}{\pi \nu_0 \tau} \sin^2(\pi f \tau),$$

where $\phi_{rms}$is the rms value of the phase deviations. The above relationship may be useful where one is trying to determine the effect of a bright line in the time domain. If the bright line is the dominant factor, the plot of $\sigma_y(\tau)$ versus τ has strong $\sin^2(\pi f \tau)$ oscillations and it can be ambiguous. In that case, it is better to provide a specification in terms of ϕ_{rms}without reference to bandwidth. Statistical measures such as $\sigma_y(\tau)$ and $S_\phi(f)$ are not meant to be used to describe coherent signals. For further discussion see paper B.1, section 12.2 (page TN-51).

9. page TN-75

A set of brackets, [ ], are missing in eq 12-29. The equation should read

$$\mathrm{mod}\sigma_y^2(\tau) \simeq \frac{1}{2\tau^2 N^2 (N-3n+1)} \sum_{j=1}^{N-3n+1} \left[\sum_{i=j}^{n+j-1} (x_{i+2n} - 2x_{i+n} + x_i) \right]^2 .$$

10. page TN-119

The reference (Walls and DeMarchi, 1975) is listed as being on pages 310-317. The page numbers should be 210-217.

11. page TN-121

Most of the literature uses the expression $[V_0 + \epsilon(t)]$ instead of V_0 as in eq (2). V_0 is the peak voltage amplitude and $\epsilon(t)$ is the voltage deviation of the amplitude from nominal.

12. page TN-122

In eq (5), most of the literature uses $x_1(t)$ instead of $\epsilon(t)$. $\epsilon(t)$ is usually the voltage deviation as described in note 11 above.

13. page TN-123

There is an error in the caption for figure 7. The last portion of that caption should read: "where f is Fourier frequency (($\omega = 2\pi f$) and $S_y(f) = \omega^2 S_x(f)$)."

14. page TN-123

In the right-hand column, last paragraph, 5th line, there is an extraneous minus sign. The quantity $\overline{y}_i^{-\tau_0}$ should read $\overline{y}_i^{\tau_0}$. Also, note that the use of supercript τ and τ_0 with $\overline{y}$ is not consistent with the new IEEE standard definitions (see paper C.1, page TN-139).

15. page TN-124

In eq 9 the subscript, k - n, should read k + n. That is, the equation should read

$$\bar{y}_k^{\tau} = \frac{1}{n}\sum_{i=k}^{k+n-1} \bar{y}_i^{\tau_0} = \frac{x_{k+n} - x_k}{\tau}.$$

16. page TN-125

The quantity $\bar{\sigma}_y^2(\tau)$ is now commonly known as mod$\sigma_y^2(\tau)$. This latter form has been recently adopted by IEEE as the standard terminology.

17. page TN-139

IEEE Std 1139-1988, *IEEE Standard Definitions of Physical Quantities for Fundamental Frequency and Time Metrology*, is an almost exact replica of this paper (D.1). The paper was published during the latter period of the development of the standard. The only substantial difference is that, wherever it occurs, the word "departure" in the paper is replaced in the IEEE standard by "deviation."

18. page TN-146

This widely cited paper provided the *de facto* standards for terminology and oscillator characterization until the recent adoption of the IEEE standard presented in paper C.1 (page TN-139). For terminology, the latest IEEE standard should always take precedence. This paper (C.2) is fairly consistent with the IEEE standard. One exception is that, in this paper, N denotes the number of frequency measurements. The symbol M in the IEEE standard is the same as N in this paper. In the standard, the equation relating M and N is M = N - 1.

19. page TN-151

Equation (23) can be substantially simplified as shown, for example, by Stein (page TN-74, eq (12-27)), which is

$$\sigma_y^2(\tau) = \frac{2}{(\pi\nu_0\tau)^2}\int_0^{\infty} S_{\phi}(f)\sin^4(\pi f\tau)\,df.$$

20. page TN-154

In eq 36 the T outside the brackets should be a τ. The equation should read

$$\hat{x}(t_0+\tau) = x(t_0) + \tau\left[\frac{x(t_0)-x(t_0-T)}{T}\right].$$

21. page TN-160

The 2 expressions listed as eq (95) are in error. They should read

$$\begin{array}{ll} \sim h_{-1}\tau^2[3 + 2\ln r - 1/(6r^2)] & r > 1 \\ \sim h_{-1}T^2[3 - 2\ln r] & r < 1. \end{array}$$

22. page TN-160
In eqs (101), (102), (103), (104), and (105), a Greek γ was mistakenly replaced by the number 2. In each of these equations the quantity $[2 + \ln(2\pi f_h \tau)]$ should be replaced by the quantity $[\gamma + \ln(2\pi f_h \tau)]$. γ, Euler's constant, has the value 0.5772156649........

23. page TN-162
This paper is included in this collection because it presents the internationally accepted terminology and definitions. There are no substantial inconsistencies with the new IEEE standard (paper C.1, page TN-139), but the latest IEEE standard should is considered to be the most up-to-date authority. A new version of this international report should be issued by the CCIR in 1990.

24. page TN-171
The definitions for symbols used in this paper are fairly consistent with those adopted in the recent IEEE standard (C.1). One exception is that, in this paper, N denotes the number of frequency measurements. The symbol M in the IEEE standard is the same as N in this paper. Another is that, while this paper uses μ as the exponent of τ in describing the power-law noise processes, the paper adopts the opposite sign convention for μ. v(t) is used where many other papers use V(t) for instantaneous voltage. Some confusion is generated when this small v is typeset in the equations to look almost identical to the Greek ν, a symbol which is used exclusively to represent frequency. For example, In equation (1) the left-hand quantity is voltage, whereas the $\nu(t)$ and ν_0 in equation (4) are clearly frequencies. Finally, the authors of this paper, in equation (2), define $\epsilon(t)$ as the normalized amplitude fluctuations, a very sound choice, but the reader should note that most other papers have not normalized it.

25. page TN-175
For consistency with figure 12 and the text, $\nu(t)$, the left-hand member of eq (11) should probably be u(t).

26. page TN-177
Walls, Percival and Irelan (D.4) have recently addressed the more accurate specification of the quantity p in eq (12).

27. page TN-179
It is important to note that the expressions in Table 2 in this paper are derived assuming use of a single-pole filter. The calculations can also be done using an ideal (infinitely sharp) filter. The solutions in these two limits are useful because they define the range of practical values (using n-pole filters) for the expressions. Table I in this section is an expansion of Table 2 of Lesage and Audoin providing both the single-pole results as well as the results for an infinitely sharp filter. There are discrepencies in several of the coefficients between terms in Table 2 in the paper and those in Table I on the next page.

28. page TN-180
Barnes and Allan (paper D.8) have recently completed further analysis of the effect of dead time on measurements.

Table I. Asymptotic forms of $\sigma_y^2(\tau)$ for various power-law types and two filter types. Note: $\omega_h/2\pi = f_h$ is the measurement system bandwidth, often called the high-frequency cutoff. $\ln \equiv \log e$.

Name of Noise	α	$S_y(f)$	$\sigma_y^2(\tau)$			
			$\omega_h\tau >> 1$ Infinite Sharp Filter	$\omega_h\tau >> 1$ Single Pole Filter	$\omega_h\tau << 1$ Infinite Sharp Filter	$\omega_h\tau << 1$ Single Pole Filter
White Phase	2	h_2f^2	$\frac{3f_hh_2}{(2\pi)^2\tau^2}$	$\frac{3f_hh_2}{(2\pi)^2\tau^2}$	$\frac{2\pi^2f_h^5\tau^2h_2}{5}$	$\frac{f_h^2h_2}{2\tau}$
Flicker Phase	1	h_1f	$\frac{(1.038 + 3\ln(\omega_h\tau))h_1}{(2\pi)^2\tau^2}$	$\frac{(3\ln(\omega_h\tau))h_1}{(2\pi)^2\tau^2}$	$\frac{\pi^2f_h^4\tau^2h_1}{2}$	$2f_h^2(\ln(2))h_1$
White Frequency	0	h_o	$\frac{h_o}{2\tau}$	$\frac{h_o}{2\tau}$	$\frac{2\pi^2f_h^3\tau^2h_0}{3}$	$\frac{2\pi^2f_h^2\tau h_0}{3}$
Flicker Frequency	-1	$h_{-1}f^{1}$	$2(\ln(2))h_{-1}$	$2(\ln(2))h_{-1}$	$\pi^2f_h^2\tau^2h_{-1}$	$8\pi^2f_h^2\tau^2h_{-1}$
Random-Walk Frequency	-2	$h_{-2}f^{-2}$	$\frac{2\pi^2\tau h_{-2}}{3}$	$\frac{2\pi^2\tau h_{-2}}{3}$	$2\pi^2f_h\tau^2h_{-2}$	$2\pi^2f_h\tau^2h_{-2}$

29. page TN-180
If the ratio of T/τ is constant and greater than 1 (the usual case), the problem described is eliminated. However, in taking data for a plot of $\sigma_y(\tau)$ versus τ, it is difficult to achieve this in the hardware and not possible to do it with software processing alone. For further discussion see paper D.8 (page TN-296).

30. page TN-197
The most recent definitions and concepts for spectral density are given in a new IEEE standard (paper C.1 on page TN-138). This new standard should be consulted as the latest authority on definitions and terminology.

31. page TN-198
The newly accepted definition of $\mathcal{L}(f)$ is given in paper C.1. This new definition, $\mathcal{L}(f) \equiv \frac{1}{2}S_\phi(f)$, was always valid for Fourier frequencies far from the carrier. It has now been extended to cover all frequencies.

32. page TN-217
Equation (73) should read 20 log(final frequency/original frequency).

33. page TN-239
On page TN-198 the authors refer to a paper by Glaze (1970). The reference, apparently lost in printing, is: Glaze, D.J. (1970). "Improvements in Atomic Beam Frequency Standards at the National Bureau of Standards, "*IEEE Trans. Instrum. Meas.* **IM-19**(3), 156-160.

34. page TN-257
There are two errors in Table 2. Under R(n) the first entry should be 1/n rather than 1. The second item in the same column is not single valued (1), but takes on different values for different measurement bandwidths. The reader is referred to section A.6 (page TN-9) for a discussion of this topic.

35. pages TN-261 and TN-262
Subsequent work on $\mathrm{mod}\sigma_y^2(\tau)$ and R(n) is reported in section A.6 (page TN-9) of this report. There are some differences between the results in A.6 and the ones reported in Tables I and II and Figure 4 in this paper.

36. page TN-264
To be consistent with other papers in the literature, $\phi(t)$ in eq (1) should probably be written as $x_1(t)$.

37. page TN-268
The term $\epsilon(t)$ in eq (3) is normally used to represent the amplitude fluctuations in the output voltage of an oscillator. This term might be better designated $Y_1(t)$.

38. Page TN-30

There are two errors in eq 6.6. The equation for Flicker PM is missing a square root in the exponential. It should read

$$\text{Flicker PM} \qquad \text{d.f.} \approx \exp\left(\ln\frac{N-1}{2n}\ \ln\frac{(2n+1)(N-1)}{4}\right)^{1/2}.$$

In the equation for Flicker FM, the numerator of the upper term should read $2(N - 2)^2$ instead of $2(N - 2)$. The equation should read

$$\text{Flicker FM} \qquad \text{d.f.} \approx \begin{cases} \dfrac{2(N-2)^2}{2.3N-4.9}, & \text{for } n=1 \\[2ex] \dfrac{5N^2}{4n(N+3n)}, & \text{for } n\geq 2 \end{cases}$$

39. Page TN-85

There are errors in two of the terms of Table 12-4. The small n in the denominator of the first logarithmic term for flicker phase should be an m, and the whole quantity in the square bracket should have an exponent of ½. The equation should read

$$\text{Flicker phase} \qquad \exp\left[\ln\left(\frac{N-1}{2m}\right)\ln\left(\frac{(2m+1)(N-1)}{4}\right)\right]^{1/2}.$$

The (N - 2) in the flicker frequency term should be replaced by $(N - 2)^2$ so that df is

$$\text{Flicker frequency} \qquad \frac{2(N-2)^2}{2.3N-4.9} \quad \text{for } m=1\,.$$

40. Page TN-279

There is a factor N missing in the first expression under eq A6. It should read

$$A = 3\,[3N(N+1)+2]\,.$$

41. Page TN-307

The fifth identity for B_1, $B_1(N,1,\mu) = 1$ for $\mu < 0$ and $= [2/N(N - 1)]$..... for $\mu > 0$, should be deleted. The identity for $\mu < 0$ is incorrect and that for $\mu > 0$ is superfluous considering the identity above it which covers all cases for $\mu \neq 0$.