Application Note 64-3

Accurate and Automatic Noise Figure Measurements

HEWLETT PACKARD

Application Note 64-3

Accurate and Automatic Noise Figure Measurements

June 1980

HEWLETT PACKARD

1501 Page Mill Rd., Palo Alto CA 94304

Contents

DESIGNED FOR HP-IB SYSTEMS

HP-IB: Not just IEEE-488, but the hardware, documentation and support that delivers the shortest path to a measurement system.

I. Introduction

Modern receiving systems must often process very weak signals, but the noise added by the system components tends to obscure those very weak signals. Sensitivity, SINAD, and noise figure are popular system parameters that characterize ability to process low level signals. Noise figure is a unique parameter in that it is suitable not only for characterizing the entire system but also the system components such as the RF amplifier, mixer, and IF amplifier that make up the system. By controlling the noise figure and gain of system components, the manufacturer directly controls the noise figure of the overall system. Once the noise figure is known, system sensitivity and SINAD can be easily estimated from system bandwidth. Noise figure is often the key parameter that differentiates one system from another, one amplifier from another, and one transistor from another. Such widespread application of noise figure specifications implies that highly repeatable and accurate measurements between vendors and their customers are very important.

Accurate noise figure measurements are valuable during design and manufacturing. Low signal level applications require accurate selection of the best noise performance components and accurate tuning adjustments during both design and fabrication. The system described here, although assembled from standard instruments, is very accurate. When measuring devices or components with an input SWR of 2.0, the root-sum-of-squares measurement uncertainty with this system, considering all effects, at most frequencies is only about ± 0.22 dB—a factor of two better than traditional systems assembled with standard noise figure meters. Measurements described toward the end of this chapter indicate that the repeatability of noise figure measurements from one system to another is within ± 0.1 dB.

This system uses a desktop computer to automate measurements, process data, and account for many small effects that, in manual noise measurement systems, are very bothersome to correct and are often accepted as measurement errors. The total error from the many small effects can easily be very significant. Such effects include ambient temperature variations, the variation with frequency of the excess noise output from the noise source, and correction for noise contributions by the receiver components of the measurement system. Correction for these effects is now so routine and simple that noise figure measurements will likely become an often-used tool to replace old, seriously questioned, often postponed manual routines. These routines were usually dreaded, especially by the occasional user who felt he had to reeducate himself at each new encounter with those many small effects.

The computer also provides data in the form desired. It can list and plot the results without tedious manual tabulation of data requiring correction charts and graphs. Permanent records are available for use by vendor and customer, labs and production, quality assurance and other functional areas. Simplified system operation with the required output format means that highly trained personnel are unnecessary for testing many components at a fast rate.

This noise figure measurement system is very flexible in several different aspects. First, the IF frequency can be anywhere in the 10 MHz to 18 GHz frequency range and the measurement bandwidth can be chosen to suit the application. Second, most of the equipment in the system is of a general purpose nature, likely to be available in many facilities and useful for other purposes besides measuring noise figure. This general purpose property also means that other equipment can often be substituted for the suggested equipment. This application note cites several such cases of alternative equipment. A third aspect of flexibility pertains to the software. The programs are written in such a way that the measurement results, for example, may be printed on a special form or special measurement sequences may be programmed. The modular construction of the programs and the detailed description in chapter three of the application note, allows easy translation to other capable computers.

This application note describes the details of this accurate and automatic system. The high accuracy arises from the HP 346B Noise Source, which has low SWR and a stable, calibrated excess noise ratio (ENR), and the HP 436A Power Meter, which can read noise power ratios to an accuracy of ± 0.04 dB. This first chapter describes the features of the system, discusses its accuracy, and presents results of measurements. Chapter two describes the equipment required, how to assemble the equipment to form a system, and how to operate the system. The third chapter discusses the computer programs so that they can be tailored to individual applications. The major example covered in this application note is the measurement of a low or medium gain preamplifier at a large number of frequencies. This preamp measurement requirement is felt to be very general and to require many special program features. Thus measurements of other kinds of components or other parts of a receiver system are of lesser complexity and may require some modification or simplification of the software. A second program to show noise measurement of mixers is also listed.

Noise Figure Principles

The purpose of noise measurements is to characterize the noise added to the signal by the unit under test (UUT). There are two popular definitions of noise figure that yield the same result even though they seem quite different at the outset.

The first definition describes noise figure as the degradation in the signal-to-noise ratio between a signal source and the UUT output. If, for example, the device is excited with a generator that is measured to have an available signal-to-noise ratio of 10 dB, but the signal-to-noise ratio at the output of the device is 7 dB, the device is said to have a noise figure of 3 dB. The standard definition of noise figure refers to a generator whose available noise power output is identical to that of an input termination at a temperature of 290 K. It should be realized that in progressing through the UUT the signal level increases (or decreases) due to device gain (or loss). Similarly, in progressing through the device, the input noise increases (or decreases) due to device gain (or loss). The noise output, however, also has components that are due to noise generation mechanisms that exist within the device. It is those components of noise that cause the degradation in the signal-to-noise ratio.

Figure 1-1(a) leads to the second definition of noise figure. The total output noise power (N_1) from a unit under test is composed of two components. One is the noise added by the unit under test (N_a) and the second is the amplified noise from the input termination when the termination is at the standard temperature of 290 K. The noise power available from the source termination is kT_oB where k is Boltzmann's constant (1.38×10^{-23}J/K), T_o is the source temperature (290 K), and B is the bandwidth over which the available noise power is measured. When the input termination is connected to the unit under test, it engenders available output noise kT_oBG_a where G_a is the available gain of the unit under test. The total available noise power output is

$$N_1 = N_a + kT_oBG_a \tag{1-1}$$

Noise figure is defined as the ratio of the total available noise power output (N_1) to the available noise power output engendered by the source (kT_oBG_a). The ratio, however, is usually expressed in dB.

$$F(dB) = 10 \log \frac{N_1}{kT_oBG_a} \tag{1-2}$$

For perspective, note that if the total noise output is twice the amplified input noise, the noise figure of the UUT is 3 dB. Small noise added leads to small noise figures.

Measuring noise figure by directly measuring the ratio has uncertainties on the order of 2 dB because the denominator, kT_oBG_a, is so difficult to determine. A more accurate measurement technique, used in almost all noise figure measurements, is to measure not only the output noise power (N_1) of **Figure 1-1(a)** but also the output noise power (N_2) of **Figure 1-1(b)** which includes extra input noise from a noise source. **Figure 1-1(b)** shows the available output noise power when the input termination temperature is raised by the excess noise temperature T_{ex} to $T_{ex} + T_o = T_{hot}$.

$$N_2 = N_a + kBG_a(T_{ex} + T_o) \tag{1-3}$$

If equations (1-1) and (1-3) are appropriately substituted into (1-2), the result is

$$F(dB) = 10 \log \frac{T_{ex}}{T_o} - 10 \log \left(\frac{N_2}{N_1} - 1\right) \tag{1-4}$$

Equation (1-4) is the foundation of noise figure measurement. The first term on the right side of (1-4) is a property of the noise source. This term is almost identical to the specified excess noise ratio (ENR) of the noise source.* The second term is found by measuring the ratio of the output powers (N_2/N_1) corresponding to the noise source at the input being hot (turned ON) and to being cold (turned OFF). The ratio N_2/N_1 is often called the Y factor.

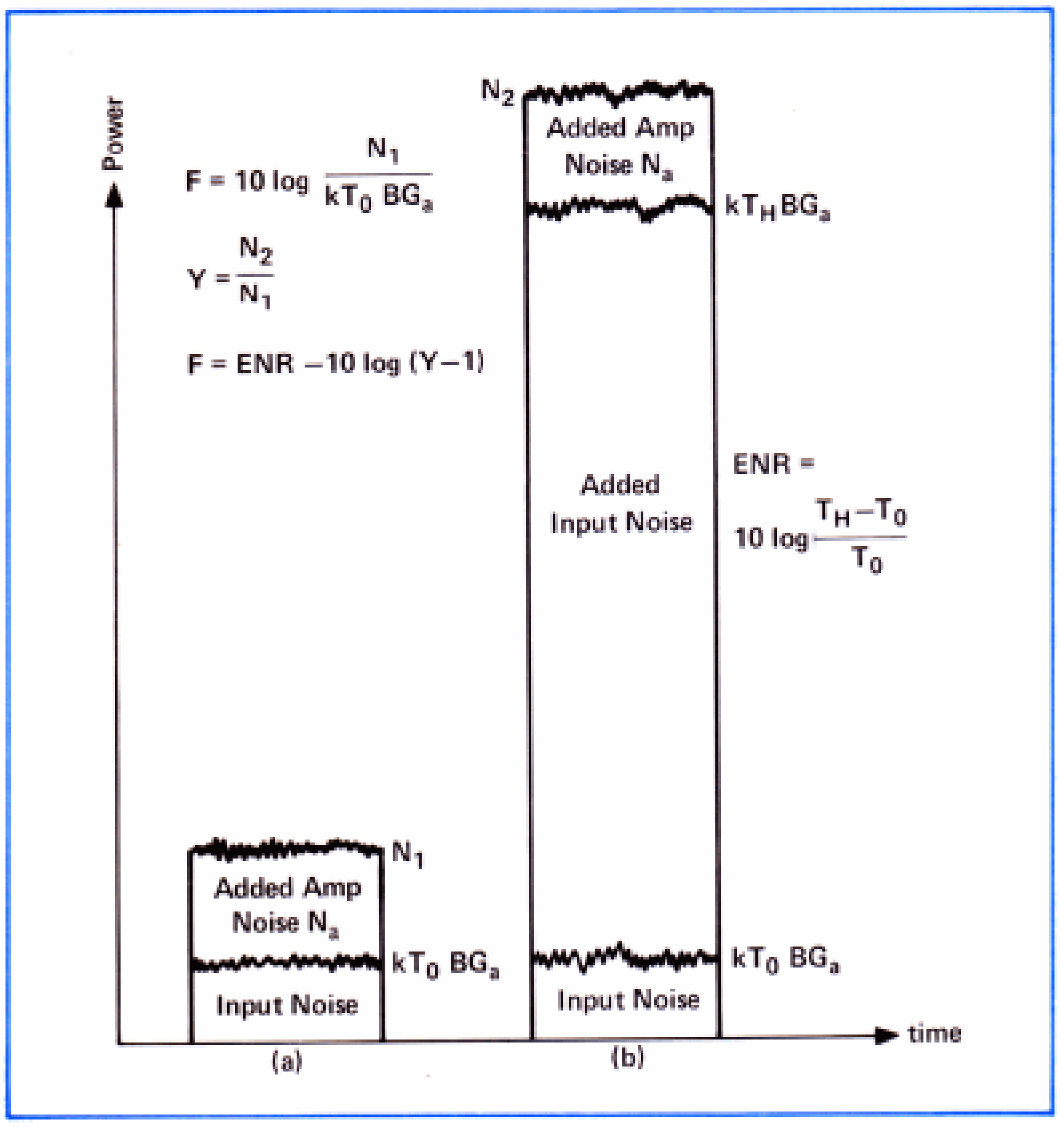


Figure 1-1. Bar graphs representing the total available noise power output from a unit under test and its component parts for (a) a T_o (290 K) source termination and (b) a source with an equivalent temperature of T_H.

*The temperatures in the first term on the right of (1-4) are associated with the power available from the source termination, that is, the power dissipated in a load that is the complex conjugate of the source impedance. ENR, however, refers to the power dissipated in a reflectionless load (usually 50Ω for coaxial systems) rather than the complex conjugate.

The Noise Figure Measurement System

Figure 1-2 is a block diagram for a general-purpose noise figure measurement system that is the subject of this application note. The unit under test (UUT) can be all of or any part of the large block shown as the UUT/receiver.

Consider first the case where the UUT is the preamplifier alone and the mixer, local oscillator (LO), and IF amplifier are part of the measurement system. The first part of the measurement process, called the calibration run, is to measure the noise characteristics of the mixer, LO, and IF amplifier at each measurement frequency and store the results in the computer memory. During the calibration run the preamplifier (UUT) is removed and the noise source is connected directly to the mixer. After the calibration run the UUT is inserted between the noise source and mixer, and the system measures the noise contribution of the entire UUT/receiver at each frequency. At each frequency the desktop computer calculates the noise figure of the UUT alone by removing the noise contribution of the measurement system from the overall measured data. The computer does this at almost no increase in measurement time. The corrected measurement also yields the gain of the UUT. This application of characterizing a preamplifier by itself is quite simple with this system but very difficult with most other systems. A program for this case is listed at the end of this application note.

Consider a second case where the UUT is the RF front end consisting of mixer, local oscillator, and perhaps a preamplifier. In this case the IF amplifier is part of the system and its noise characteristics must be determined. To do this the noise source is connected directly to the IF amplifier and calibration run data is taken at the IF frequency. Note that the broad frequency range on the 346B allows it to also be a calibrated noise source for measuring IF amplifiers.

The next step is to insert the UUT so the system can make measurements at each of the desired frequencies. In this case the local oscillator is probably different from the HP 8672A Signal Generator or 8620C/86290B Sweep Oscillator for which these programs were written. If so, a small section of the computer program will have to be modified to properly set the local oscillator frequency. If the receiver is of the hand-tuned type, the computer software will continually output results as the LO is manually tuned. The software could also be modified to prompt the operator to tune to desired frequencies.

If the UUT is a mixer alone, the IF amplifier is again part of the system and the calibration process is the same as for the second case above. The 8672A or 8620C/86290B can still be used as the local oscillator with no change made to the frequency programming part of the software. Software for this mixer application is also listed at the end of this application note.

Although other noise sources can be used with this system, the HP 346B is highly recommended because it yields such accurate measurements and covers such a broad frequency range. The power supply that biases the noise source ON and OFF must be capable of supplying 28 volts and of being turned ON and OFF by the computer. The HP 6002A Power Supply or the HP 6205B Power Supply with

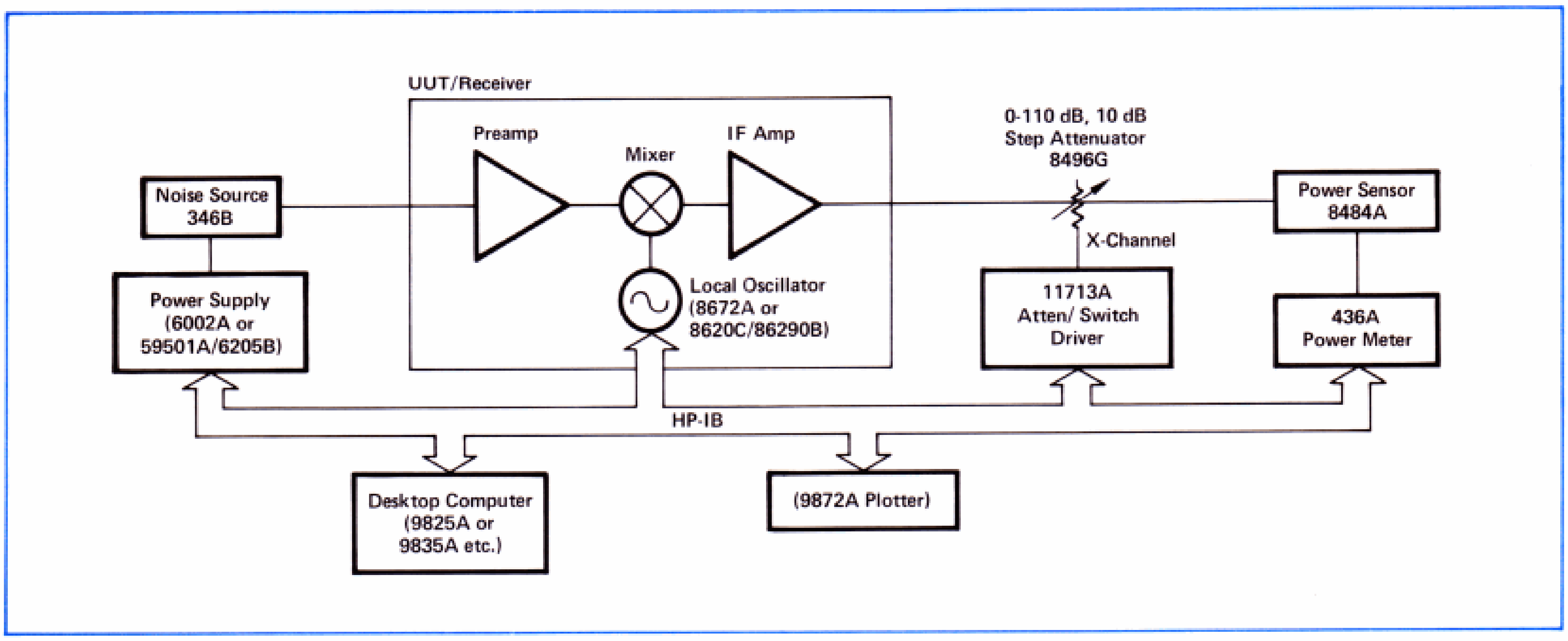

UUT/Receiver
Preamp
Mixer
IF Amp
Noise Source 346B
Power Supply (6002A or 59501A/6205B)
Local Oscillator (8672A or 8620C/86290B)
0-110 dB, 10 dB Step Attenuator 8496G
X-Channel
Power Sensor 8484A
11713A Atten/ Switch Driver
436A Power Meter
HP-IB
Desktop Computer (9825A or 9835A etc.)
(9872A Plotter)

Figure 1-2. Block diagram of the general purpose noise figure measurement system. The unit under test can be all of or any part of the UUT/Receiver portion of the block diagram. The remainder of the UUT/Receiver block is then considered part of the measurement system and its noise contribution can be calibrated out of the measurement results by the computer.

the HP 59501A Power Supply Programmer are two alternatives for driving the noise source.

The output noise power is measured with the 8484A Power Sensor and the 436A Power Meter. This combination can measure the output ON-to-OFF noise power output ratio to an accuracy of ± 0.04 dB. The 8496G Programmable Step Attenuator shown in front of the 8484A Power Sensor of **Figure 1-2** adjusts the power level to be in the −20 to −50 dBm range where the 8484A/436A measures most quickly and accurately.

The desktop computer coordinates and controls the operation of the measurement equipment. It sends out commands on the interface bus (HP-IB) to turn the noise source ON and OFF, set the local oscillator frequency, vary the attenuator to adjust the range of the powers to be measured, and to read the power from the power meter. The computer processes the measurement data and sends the results to a printer and/or plotter.

Besides the operation of equipment, the processing of data, and the presentation of results, the computer also corrects for several effects that are so complicated in manual systems that they are usually accepted as measurement errors. First the computer corrects for noise source output variations with frequency. The computer memory contains measured values of the ENR for the particular 346B Noise Source being used at 20 frequencies in the 10 MHz to 18 GHz range. For each measurement frequency the computer interpolates the stored data to find the proper ENR. This process alone usually reduces uncertainty by about 0.2 dB, but it could be as much as 0.5 dB.

The computer also corrects for the ambient temperature. Equation (1-2) only applies when the source-off temperature is 290 K (17° C). For temperatures different than 290 K the measurement must be corrected so the results are presented as if the source were at 290 K. For measured noise figures of about 3 dB, this effect, if not accounted for, can cause errors on the order of ± 0.1 dB.

The third type of computer correction is the automatic correction for the system noise contributions. **Figure 1-3** shows how much correction is applied to the measured overall noise figure (F_{12}) when calculating the UUT noise figure (F_1). To use the chart consider that the measurement system with a noise figure of 8 dB (F_2) measures the noise figure when the UUT is attached to the system and the result is 5 dB (F_{12}). Further, consider the UUT to have a gain (G_1) of 10 dB that is also measured by the system. To find the correction to F_{12}, move along the horizontal axis to $F_{12} + G_1 = 15$ dB. Then move vertically to intercept the $F_2 = 8$ dB curve. Read the correction of 0.8 dB off the vertical axis. Thus the UUT noise figure (F_1) is $5 - 0.8 = 4.2$ dB. This entire process is done automatically by the computer.

Accuracy

To evaluate the accuracy of this system, the errors are categorized into four groups: mismatch uncertainty, ENR uncertainty, instrumentation uncertainty including receiver errors, and variation of the UUT noise figure with the noise source impedance.

Mismatch Uncertainty

Mismatch uncertainty is caused by rereflections between the UUT and the noise source. The rereflections can cause the noise power emerging from the source to be different than it would be when facing a reflectionless load. The mismatch occurs under two conditions, one when the noise source is ON and again when it is OFF, and the rereflections could have opposite effects on those two occasions. **Figure 1-4** shows the maximum possible error in the measured noise figure due to mismatch. Each of the four graphs is for a different value of SWR for the UUT. Consider the example of a noise source with a SWR of 1.1 and a UUT with a SWR of 2.0. **Figure 1-4** (the graph for UUT's with a SWR of 2.0) shows that a source SWR of 1.1 yields a maximum measure-

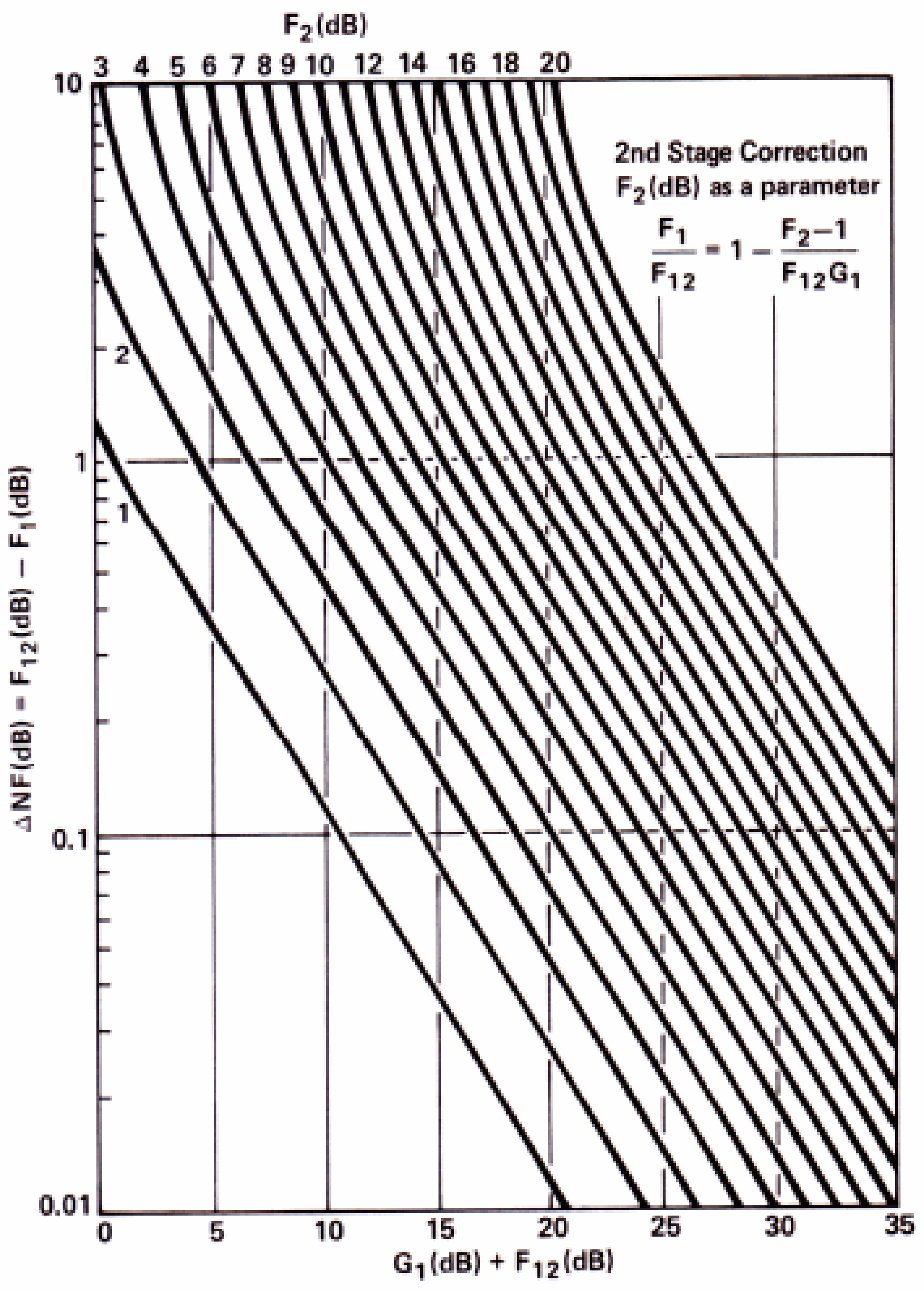


Figure 1-3. The correction due to the system's contribution of noise to measured data. After the calibration run, the computer performs this correction at each measurement frequency.

ment error of about ± 0.15 dB for measured noise figures larger than 10 dB. For purposes of calculating an overall uncertainty, a mismatch uncertainty of ± 0.15 dB is taken as typical of measurement systems that use the 346B.

ENR Uncertainty

The ENR of the noise source has a root-sum-of-squares (RSS) uncertainty of ± 0.1 dB for frequencies below 8 GHz and increasing to ± 0.19 dB at 18 GHz. At the time of this writing the 346B ENR calibration is traceable to the U.S. National Bureau of Standards (NBS) through waveguide gas discharge type noise sources measured at NBS from 9 to 18 GHz and at 3 GHz. At other frequencies the 346B calibration is referenced to a hot termination (390 K) and a cold termination (78 K) in Hewlett-Packard's standards laboratory. Justification of the 346B ENR calibration accuracy is the subject of a paper by Pratt.*

*Pratt, R.E., to be published.

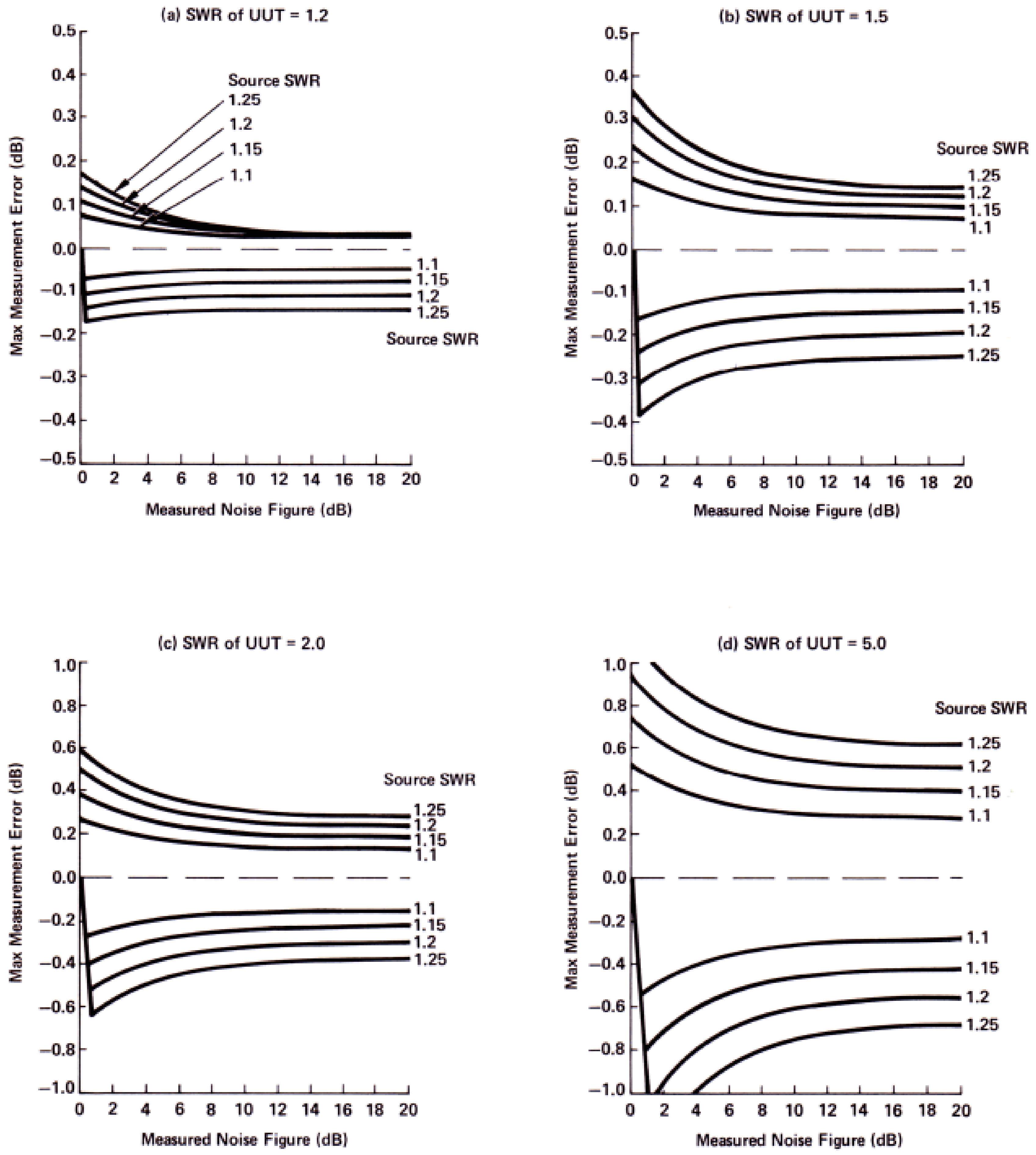


Figure 1-4. The worst case errors in the measured noise figure caused by rereflections between the noise source and UUT. Each graph is for a different UUT input SWR.

The RSS method of calculating the uncertainty is based on the fact that the many sources of error are independent of each other and, even though they might be systematic, combine like random variables. The overall RSS uncertainty of the ENR, including an allowance for interpolation at arbitrary frequencies, is typically ± 0.15 dB.

Instrumentation Uncertainty

The Power Sensor/Power Meter combination measures the Y factor, N_2/N_1, to an accuracy of ± 0.04 dB, including the non-linearities of the power sensor. This compares to an uncertainty of traditional noise figure meters of ± 0.15 dB or higher. Receiver errors include mixer and IF amplifier non-linearities, i.e., different response for the noise source ON and OFF conditions. The estimated uncertainty from this cause is an additional ± 0.05 dB.

Noise Figure vs. Source Impedance

The noise figure of a UUT generally varies with source impedance in an undetermined way. This means that if the noise source impedance is different than 50Ω, the noise figure of the UUT is probably different than its value when connected to a 50Ω source. This variation is very difficult to determine and is not considered here. The effect is minimized, however, by using a noise source with a low SWR.

Overall Measurement Uncertainty

An RSS combination of the four remaining uncertainties (± 0.15 dB for mismatch uncertainty, ± 0.15 dB for ENR uncertainty, ± 0.05 dB for receiver nonlinearity, and ± 0.04 dB for power meter instrumentation) yields an overall uncertainty of ± 0.22 dB.

The correction for system noise contribution **(Figure 1-3)** could have its own errors because of the measurement errors in system noise figure (F_2), overall noise figure (F_{12}), and UUT gain (G_1). For a set of measurement data, the effect can be calculated from Figure 1-3. Consider, for example, that F_2 is 8 ± 0.25 dB, F_{12} is 5 ± 0.25 dB, and G_1 is 10 ± 0.25 dB. Using the 7.75, 5.25, and 10.25 dB extremes for F_2, F_{12}, and G_1, the correction (ΔNF) from Figure 1-3 is 0.65 dB for an F_1 of 4.6 dB. Using the 8, 5, and 10 dB values for F_2, F_{12}, and G_1, the correction is 0.8 dB for an F_1 of 4.2 dB. Thus 0.25 dB of uncertainty in both noise figure and gain measurements could cause a 0.4 dB peak change in the final noise figure. It is usually best to minimize the correction by selecting system measurement components that yield a small system noise figure.

Demonstrated Performance

The repeatability of noise figure results from one measurement system to another is shown in **Figure 1-5**. That figure shows three different swept noise figure measurements from 2 to 12 GHz at 100 MHz frequency increments. Each swept measurement used a different 346B Noise Source, a different 8484A Power Sensor, and a different 436A Power Meter. Each of the three sets of equipment, furthermore, were from different production runs. Yet the measurement results at each frequency show a maximum variation of ± 0.1 dB. The excellent repeatability of results when all the measuring instruments are changed means that now data measured at one location can be verified at a different location. This repeatability also demonstrates that the measuring equipment is stable with time.

The actual accuracy of the system is very difficult to demonstrate because present day traceability of excess noise

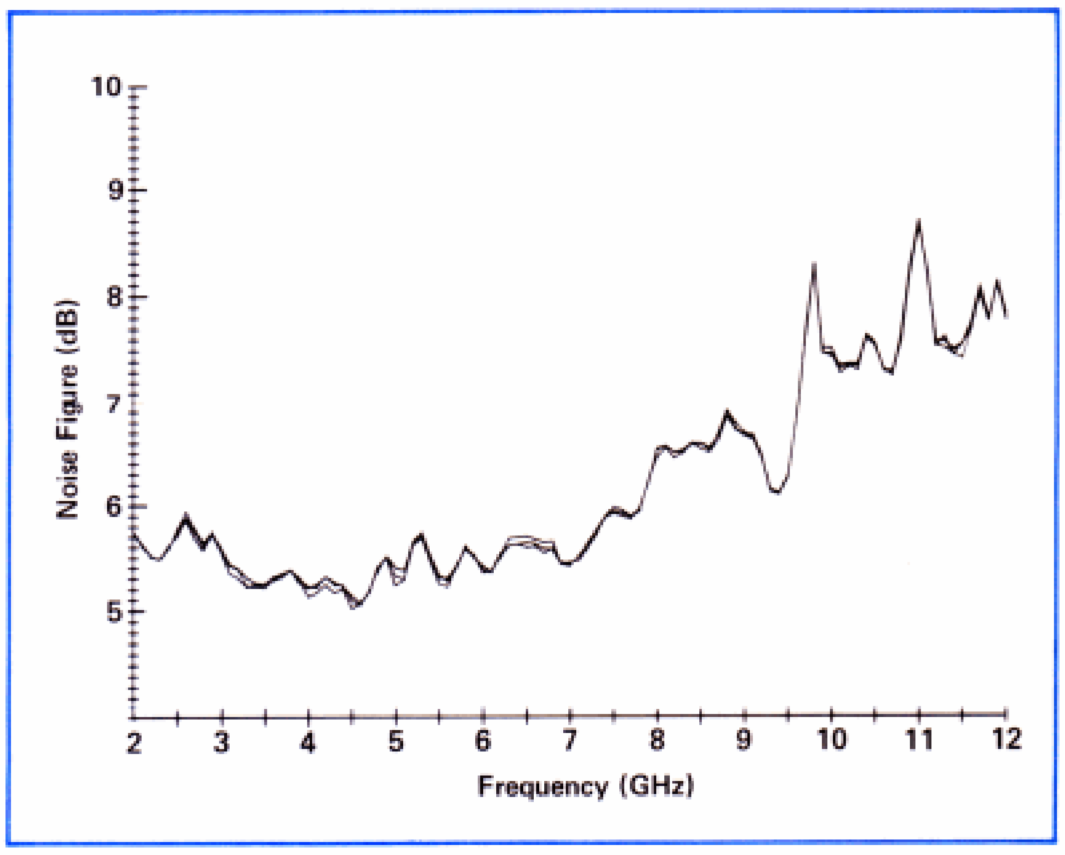


Figure 1-5. Three plots of the noise figure of a broadband receiver measured by three different sets of measuring equipment are within 0.1 dB at most frequencies.

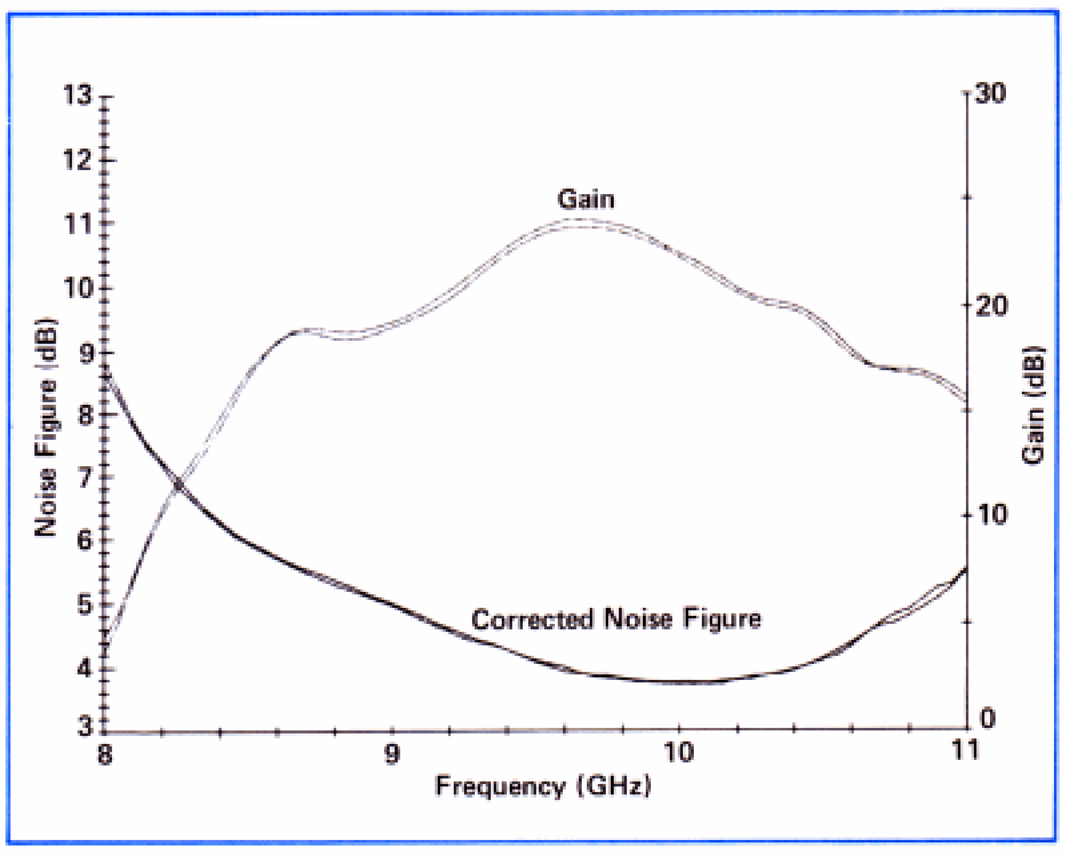


Figure 1-6. The corrected noise figure and gain of an X-band amplifier, as measured by this system using two different mixers, shows good repeatability.

ratio measurements to national standards is tenuous with many possible sources of error. This system measured the noise figure of an amplifier (corrected for the receiver noise contribution) to be 3.14, 3.04 and 3.01 dB at 9.55, 9.75, and 9.95 GHz respectively. Careful, exacting, and more time-consuming measurements with a different type of system with completely different equipment yielded 3.01, 3.03 and 3.14 dB noise figures at the same frequencies.

Figure 1-6 shows plots of the corrected noise figure and gain of an amplifier with two different mixers. Rereflection between the mixer and UUT were minimized by including a 3 dB attenuator at the mixer input. This added loss would traditionally degrade accuracy by making the system noise contribution higher. With this system, however, the system noise contribution is automatically removed from the final result, leading to the good repeatability of **Figure 1-6**. The 9872A Plotter actually makes the noise figure and gain plots in different colors.

The noise figure and conversion loss of mixers are often assumed to be the same, but the plot in **Figure 1-7** shows them to be different. Although it is beyond the scope of this note to study the intricacies of mixer operation, the conversion loss and noise figure each vary differently with local oscillator drive power, the match on the various ports, etc. Such measurements, easy to do with this system, enable optimization of the local oscillator and mixer combination. The measurement and plotting time for the 101 frequencies and two curves of **Figure 1-7** were about 70 seconds.

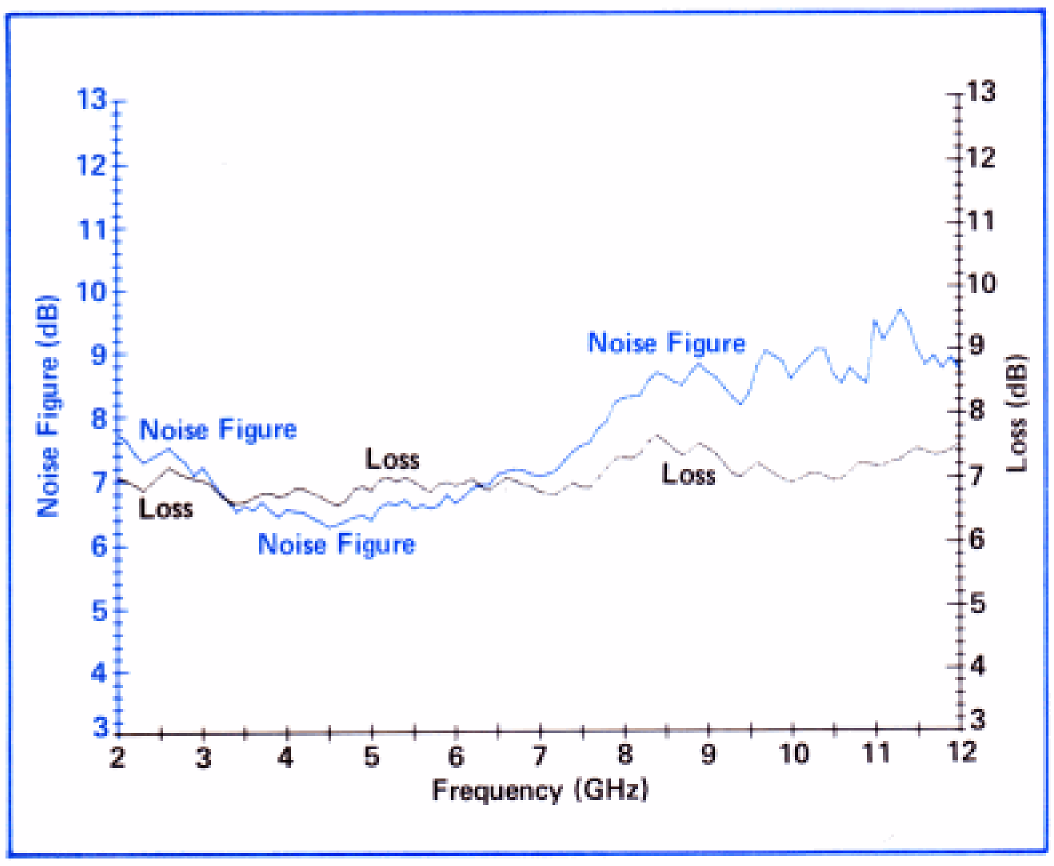


Figure 1-7. Corrected noise figure and insertion loss measurements of a double balanced mixer.

The noise figure measurement system of this application note, although it mostly consists of general purpose equipment, can be easily tailored to specific requirements. The desktop computer has the ability to easily and rapidly correct for many noise measurement effects that were previously often considered as errors. This ability, when combined with the accurate noise source and power meter of this system, gives measurement accuracy and repeatability that were previously possible only with great effort by highly skilled people and usually at only a few frequencies.

II. Assembly and Operation

This chapter serves as a guide for assembling and operating the noise figure measurement system discussed in the first chapter. The goals for selecting equipment are that the system be accurate, be convenient to use, require a minimum of special wiring, and be easy to adapt to other needs with minor modifications. Most of the equipment is of a general-purpose nature and likely to be available in many organizations, allowing this system to be assembled for trial use at minimum expense. Although the equipment used in the system is listed under "Suggested Measurement Equipment", many alternative components may be chosen from available supplies. The 346B Noise Source, the 8484A Power Sensor, and the 436A Power Meter are the principal accuracy-determining elements of the system. This application note emphasizes testing RF preamplifiers because this application demonstrates most of the features of the system and is of interest to a large number of readers. Testing other receiver components will generally require less measurement equipment and perhaps minor modification to the software. This chapter, on constructing the system, and the next, on describing the software, should facilitate tailoring the measurement system to other applications.

Suggested Measurement Equipment

The following list shows the equipment that has been extensively used to demonstrate the system capability and accuracy.

Desktop Computer	HP 9825T
(Other 9825's need 23K bytes memory, String-Advanced Programming ROM, 9872A Plotter—General I/O—Extended I/O ROM*)	
HP-IB Interface Card	HP 98034A
Thermal Printer	HP 9866B Opt 025
(with interface to 9825A Calculator)	
Plotter	HP 9872A Opt 25
HP-IB Cables (1m, 4 each)	HP 10631A
Tape Cartridge for 9825A	HP11719A
Noise Source (with male APC-3.5 Connector)	HP 346B
DC Power Supply	HP 6002A Opt 001
Power Sensor	HP 8484A
Power Meter with HP-IB	HP 436A Opt 022
Programmable Step Attenuator	HP 8496G Opt 001
Attenuator/Switch Driver	HP 11713A
Synthesized Signal Generator	HP 8672A
Double Balanced Mixer	HP HMXR-5001
Dual Power Supply	HP 6205B
IF Amplifier	RHG # ET3010
50Ω Coaxial Cable	HP 11500B
with Type N (m) connectors (24 in. long)	
50Ω Coaxial Cable	HP 10503A
with BNC (m) connector (48 in. long)	
50Ω Coaxial Cable	HP 11001A
with one BNC (m) connector and one dual banana plug	
Adapters as required to insert unit under test and interconnect mixer, local oscillator, etc.	

Figure 2-1 is an example diagram showing the various adapters and cables for interconnecting components. A similar diagram should be prepared for each application to help pinpoint critical, long-delivery cables and adapters.

Alternative Equipment

The equipment listed above is suggested for the system. Except for only a few items, other equipment may be substituted. The items that are important for maintaining the accuracy of the system are the 346B Noise Source and the 436A Power Meter with the 8484A Power Sensor. Some of the alternatives for the rest of the equipment will now be discussed.

The most important requirement of a mixer is that it cover the proper frequency range. The one mentioned above is designed for the 2 to 12.4 GHz frequency range. A second desirable characteristic is that the mixer be of the balanced or double-balanced type. Double-balanced and balanced mixers help prevent phase noise from the receiver local oscillator from increasing the noise power output of the receiver. Other suppliers of broadband double-balanced mixers

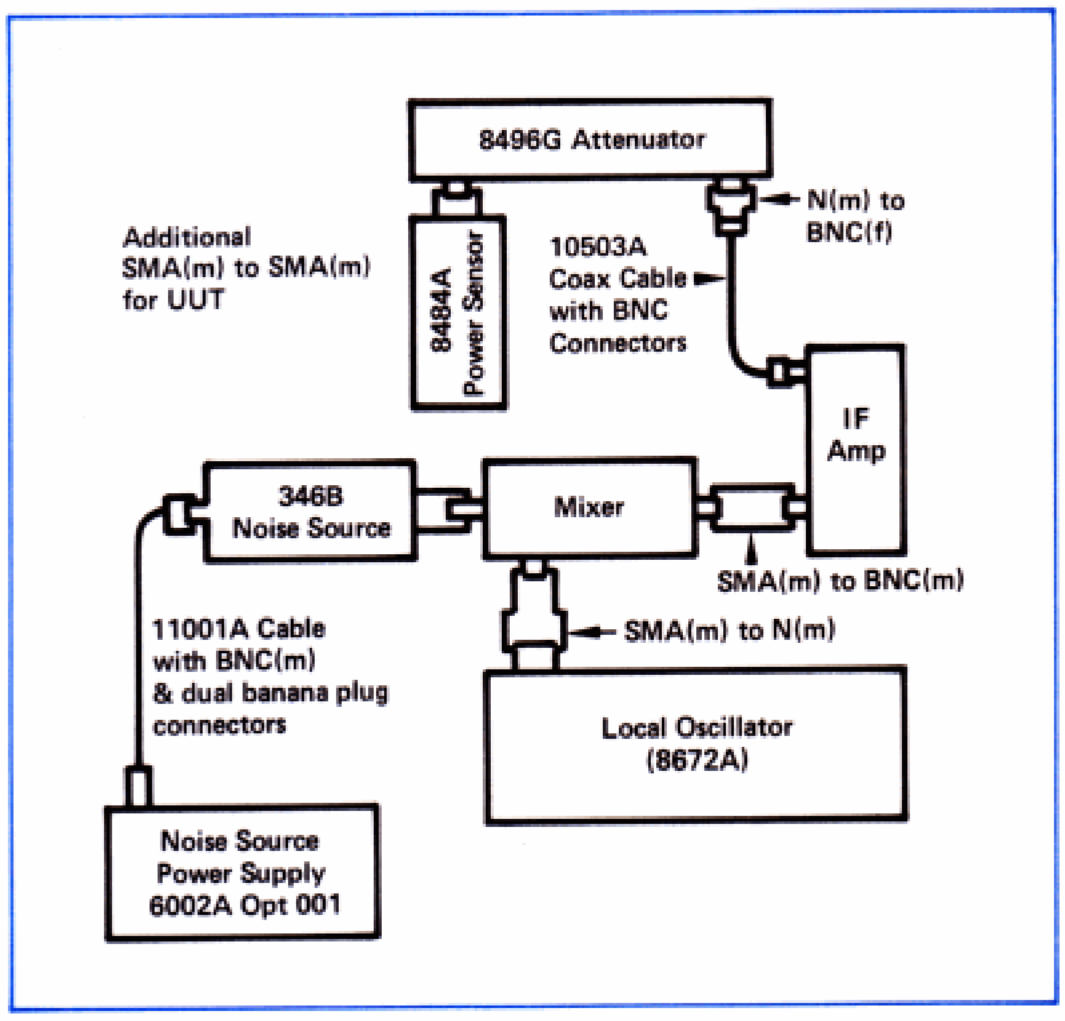

8496G Attenuator
N(m) to BNC(f)
Additional SMA(m) to SMA(m) for UUT
8484A Power Sensor
10503A Coax Cable with BNC Connectors
IF Amp
346B Noise Source
Mixer
SMA(m) to BNC(m)
11001A Cable with BNC(m) & dual banana plug connectors
SMA(m) to N(m)
Local Oscillator (8672A)
Noise Source Power Supply 6002A Opt 001

Figure 2-1. Example block diagram of the signal path used showing interconnecting cables and likely adapters.

*Note that this ROM is necessary with the software described in this Application Note even if the 9872B Plotter is not used.

include RHG Electronics Laboratory and Watkins-Johnson Co.

The local oscillator shown for this system is the 8672A Synthesized Signal Generator covering the 2-18 GHz frequency range. The HP 8620C/86290B Sweep Oscillator covers the same frequency range and works quite well for most applications. For applications such as mixer characterization of conversion loss and noise figure versus LO drive level, or for certain gain saturation measurements to be made on the same system, the 8672A is preferred because of its programmable output level. Programmable local oscillators for other frequency ranges include the HP 8660C Synthesized Signal Generator with the 86603A RF Section (to 2.6 GHz), and the HP 8662A Synthesized Signal Generator (to 1.28 GHz). The critical properties of the local oscillator are that it be programmable for automatic receiver tuning, that it have enough power to drive the mixer, and that its phase noise and other spurious signals offset by the IF frequency from the carrier be low. Some double-balanced mixers require +7 dBm local oscillator power to achieve a good noise figure. The noise and spurious output from the local oscillator, if it causes larger mixer output than engendered by thermal noise at the mixer input, will raise the system noise figure. The best way to tell whether a local oscillator is free enough of noise is to try it. YIG-tuned oscillators and high Q cavity type oscillators, when used with balanced and double-balanced mixers, tend to have low enough noise.

In some cases, such as testing a television tuner, the unit under test consists of an RF preamplifier, the mixer, and a local oscillator. If the tuner is electronically tuned, an HP 59501A Power Supply Programmer can likely tune through the many VHF and UHF channels under program control.

One advantage to the system described in this note is that the IF frequency can be anywhere in the 10 MHz to 18 GHz range of the power sensor. Furthermore, the IF bandwidth is also flexible. Most noise figure measurements are double sideband measurements, meaning that the noise figure is measured in two bands, one at $f_{LO} + f_{IF}$ and the other at $f_{LO} - f_{IF}$. As the local oscillator frequency is tuned across the operating frequency range, the measured noise figure is a type of running average of the noise figure in each band. For broadband components like preamplifiers, where the noise figure and gain tend to vary slowly with frequency, the running average is commonly accepted as the proper noise figure with little error. For less averaging effect, choose an IF frequency that is small compared to the expected variations of the UUT noise figure and gain with frequency. As the IF frequency is reduced, however, local oscillator noise might significantly raise the system noise contribution.

Single sideband noise figure measurements with receivers require either image rejection mixers or filters to eliminate one sideband ahead of the mixer. In both cases the noise power at the receiver output due to thermal noise power in one of the sidebands is removed. Filter design is greatly simplified by using a high IF frequency so that the desired RF sideband is greatly separated from the image sideband to be rejected.

The bandwidth of the IF amplifier for noise figure measurements is usually in the 1 to 10 MHz range. If the bandwidth of the IF is excessive compared to the bandwidth of the receiver components ahead of it, noise power contributed by the IF amplifier over the excessive bandwidth may obscure the measured noise figure of the unit under test. The safest practice is to have progressively narrower bandwidths from the receiver input to the IF output.

The gain and bandwidth of the IF amplifier must be sufficient so the amplified input noise lies within the dynamic range of the power measurement equipment at the output. For the HP 8484A Power Sensor the power output should be in the –49 to –20 dBm range for fast and accurate power measurements. The power output while the noise source is OFF is given in dBm by

$$N_{OFF} = kTB(dBm) + G + F \qquad (2\text{-}1)$$

where F is the overall noise figure in dB, G is the overall gain in dB from the input of the receiver to IF output, and kTB is the input noise power (–114 dBm for a 1 MHz bandwidth and –104 dBm for a 10 MHz bandwidth for single sideband measurement while –111 and –101 dBm apply for double sideband measurements). The maximum power output while the noise source is ON is limited by

$$N_{ON\ MAX} = kTB + G + 3 + \max(F, ENR) \qquad (2\text{-}2)$$

For the system being described, the IF bandwidth is 10 MHz and the lowest noise figure of the test system is about 6 dB. That means the IF gain must be about 50 dB to ensure a noise power output at the sensor of –49 dBm.

The IF amplifier on the "Suggested Equipment List" has about 80 dB gain, 10 MHz bandwidth, and 3 dB noise figure. The mixer has about 7 dB loss and 7 dB noise figure. With the attenuator set to 10 dB and for the noise source ON, the measured power level is about –20 dBm—just low enough for the power meter to read. If, however, because of high gain UUT's the signal becomes so large that more than 30 dB attenuation is needed to keep the power meter reading lower than –20 dBm, then the IF output level is above about +10 dBm and probably too close to gain saturation for accurate measurement. The limits of the attenuator settings for this program are therefore 10 to 30 dB. If the program requires attenuation outside these limits to make proper power meter readings, an error message will occur and the program will

At certain times the program requires the operator to perform a function such as to connect a new device to be measured. The program pauses with a prompt message in the computer display for the operator. After performing the required operation, press the CONTINUE key.

For certain functions, the program prompts the operator to key in new values of some variable. Examples of such numbers are the measurement frequencies or the limits to be plotted on a graph. If the operator presses the CONTINUE key without entering a new value with the keyboard, the program will retain the old value for that variable. To enter a new value, the operator enters the value and then presses the CONTINUE key.

The RUN button on the computer should only be used when starting the program for the first time. If the program is stopped, for whatever reason, the CONT O special function key should be used to restart the program. The reason for this is that RUN initializes all variables of the program to zero and such variables as calibration data and parameters for the plotter must be re-entered into the computer. The CONT O key starts the program again but retains the old variables.

When the program is finished doing what it was ordered to do, it displays AWAITING MEASUREMENT COMMAND until the operator presses a new special function key.

The special function keys, pictured in **Figure 2-3,** operate as follows:

ENT S,S,S: Interrupts whatever measurements are being made and begins prompting the operator for new start, stop, and step frequencies for swept noise figure measurements. Enter the new variables and press CONTINUE after each.

SWEEP: Interrupts whatever measurements are being made. Begins making noise figure measurements in the swept mode according to the start, stop, and step frequencies in effect. If no frequencies are yet in effect, the program first prompts entry of start, stop, and step frequencies.

ENT CW: Interrupts whatever measurements are being made to prompt for a new fixed frequency of measurement. Enter the new frequency and press CONTINUE.

CW: Interrupts whatever measurements are being made and measures noise figure at the CW frequency in effect. The program continues making that measurement over and over until interrupted by pressing some special function key. Plotting is disabled in CW operation.

EXT FREQ: Interrupts whatever measurement is being made and places the local oscillator in the manual mode of operation. The program continuously makes noise figure measurements while the operator is free to tune the receiver manually. Using this command also has the effect of pressing the NF ONLY key. This is because of lack of calibration data at the frequency that is manually tuned.

NEW ENR: Interrupts whatever measurement is being made and prompts the operator for a new value of noise source ENR to be used during EXT FREQ operation. Since the computer has no way of knowing the receiver frequency, it cannot use its stored table of ENR data so it needs a manually input value of ENR. If no special value of ENR is ever entered, the program uses the default value of 15.2 dB.

NF ONLY: The system measures and displays the total noise figure of the equipment between the 346B Noise Source and the 8484A Power Sensor.

NF & G: The system measures the overall noise figure of the UUT combined with the system components, but corrects the measured data to display the noise figure and gain of the UUT by itself, without the system noise contribution. To perform this correction, the system must first make calibration noise measurements on the system without the UUT.

NEW CAL: The system initializes new noise calibration

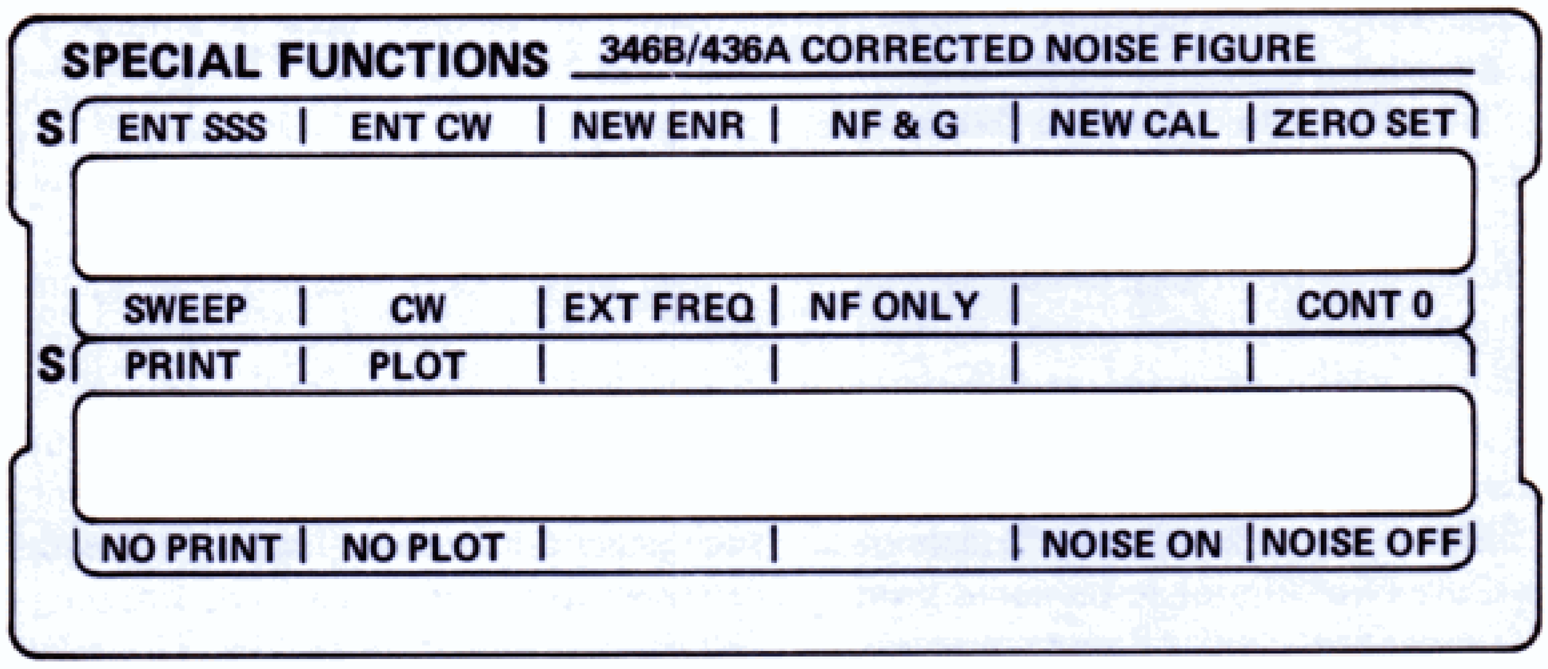


Figure 2-3. Overlay for the special function keys of the computer. These are used as controls for operating the system.

measurements for use in future corrected measurements that display the noise figure and gain of the UUT by itself.

ZERO-SET: Interrupts whatever measurement is being made. Turns the noise source OFF, inserts maximum attenuation at the output of the IF, and zero-sets the power meter. Resets the attenuator to its original value and returns to the AWAITING MEASUREMENT COMMAND state.

CONT 0: For use in restarting the program once it is stopped for any reason. This key is used instead of RUN because RUN initializes all variables to zero, making it necessary to re-enter ENR data about the source, plotting parameters, calibration data, etc.

PRINT: Prints the measurement results on the external printer of the system. Results are always displayed on the LED readout of the computer.

NO PRINT: Aborts the printing of measurement results on the external printer of the system. Results are still displayed on the LED readout of the computer.

PLOT: Plots the measurement results versus frequency on the 9872A Plotter for swept measurements. CW measurements are not plotted. Just before plotting the measurement result at the start frequency, the computer prompts the operator about printing the axes for plotting.

NOISE ON: Interrupts the measurement being made, turns ON the noise source power supply, and places other system instruments in manual operation.

NOISE OFF: Interrupts the measurement being made, turns OFF the noise source power supply, and places other system instruments in manual operation. This control is very useful for stopping a measurement and getting back to the AWAITING MEASUREMENT COMMAND state of the system.

Turn-on Procedure

(1) Set up the measurement system as shown in Figure 1-2. Zero-set and set the CAL ADJ of the 436A/8484A according to the 436A Operating and Service Manual. Be sure to connect the 11708A Reference Attenuator to the 436A Power Reference Output when setting the CAL ADJ. Connect the 8484A to the system output as shown in Figure 1-2.

(2) Before turning ON the 6205B Power Supply, remove the power output leads. Check the voltage output before reconnecting to insure that no changes have been made since the last time the system was operated. Significant overvoltage will likely damage the IF amplifier. Connect the power supply output.

(3) Insert the data cartridge into the computer, load the noise measurement program, and press RUN. The program loads the special function keys.

(4) The program then asks for the file number where the ENR data for the 346B Noise Source is stored. Type in the file number and press CONTINUE.

(5) Next the program asks for the ambient temperature in kelvins. If CONTINUE is pressed the system will use the default value of 290 K. This value tells the computer the true OFF noise power output from the 346B. If the 346B ambient temperature is more than 10 K different than 290 K, enter the new value to avoid significant errors.

(6) The program displays its AWAITING MEASUREMENT COMMAND message. Remove the lead from the 6002A Power Supply to the 346B Noise Source. Press the NOISE ON special function key and check the 6002A meter for +28 V from the noise source power supply. Then connect the 346B to the power supply.

(7) Press the ZERO SET key on the calculator. This should also be done every several minutes while the AWAITING MEASUREMENT COMMAND message is displayed. Frequent zero-setting overcomes possible drift in the 8484A.

(8) Begin measurement by using the special function keys.

Using the Plotter

The program assumes there is a plotter included in the system. If there is no plotter, line 6 of the program must have the **psc 705** changed to **psc 0** for the program to run. Even if there is no plotter, the 98216A ROM for the plotter must still be installed in the computer.

The prompt messages that pertain to the plotter are concerned with printing the scales. Before running the program the plotter should be adjusted so that the P_1 and P_2 buttons move the plotter pen holder to the lower-left and upper-right corners of the plotting area to be used. The graphical area is somewhat smaller than P_1 and P_2 to allow room for labels in the margin. The frequency scale is plotted along the bottom of the paper, the noise figure scale along the left edge and the gain scale along the right edge.

When the plotting routine is entered for the first time, default min and max values are chosen by the program for each axis. Default values are also chosen for spacing between tic marks for each scale and for the number labels that are printed near cardinal tic marks.

The plotting routine works as follows:

(1) After measurements are made at the start frequency, the program pauses for an answer to "NEW PAPER (Y OR N)?". The default answer, achieved by pressing CONTINUE without entering Y or N, is "N". For an "N" answer, the computer assumes the axes are already printed and plots the measured noise figure at each frequency. A "Y" answer triggers questions about the existing axes parameters. If the plotting routine is being used for the first time, the paper is assumed to be new and this question is skipped.

(2) If the paper is new, the program asks, "ARE THE OLD AXES OK (Y OR N)?". The default answer is "Y" and the axes are replotted just as they were on the previous occasion. Initial operation of the plotting routine skips this question.

(3) The computer asks about the minimum and maximum value of each scale, the interval between tic marks, and how often the tic marks should be labeled. As each question is asked, the LED readout also displays the current value that answers the question. If that value is still desired, press CONTINUE; if not, enter the new value before pressing CONTINUE.

(4) After a plot of the corrected noise figure the computer asks whether overall noise figure should also be plotted (default answer is "N"), and whether the gain should be plotted (default answer is "Y").

Noise Source Data File Construction Program

This program is used to construct, modify, print, or copy the calibration data file of the 346B Noise Source. Operation of the program is as follows:

(1) Load the program and press RUN.

(2) Answer the prompt "OLD CALIBRATION FILE # =?" with the data file number and press CONTINUE. If a completely new file is to be constructed simply press CONTINUE without entering any number.

- If a totally new file is being constructed, the program loads all variables with value zero.
- If a totally new file is not being constructed, the program loads the old data. Then answer the prompt "CHANGE DATA (Y or N)?" with a Y for yes or N for no and press CONTINUE. If "N" is entered, proceed to step 4 below.

(3) As the program prompts the operator through the various pieces of data the old data is displayed. If no change is to be made to the old data item, simply press CONTINUE to retain the old data item. The next data item is then prompted for entry. If a new value is to be entered, type in the value and press CONTINUE.

- The first item is an alphanumeric description of the noise source of up to 72 characters. Serial number and the date of the calibration data are likely entries.
- The remainder of the prompts proceed through the array of calibration data, A [20,6], column by column from columns 1 through 6. Column 1 is for frequency in MHz. Column 2 is for ENR. Columns 3 through 6, for reflection coefficient data, are optional because the current noise measurement program does not use the data. Column 3 is normally for the magnitude of the ON reflection coefficient, column 4 for the angle of the ON reflection coefficient, column 5 for the magnitude of the OFF reflection coefficient, and column 6 for the angle of the OFF reflection coefficient. If a mistake is made, write it down and proceed as if there were no mistake. There is an opportunity for corrections in step 5 below.

(4) The next question is "PRINT CALIBRATION DATA (Y or N)?" If "Y" is answered, all the data items are printed.

(5) The next question is "ANY OTHER CHANGES (Y or N)?" If "Y" is answered, the program goes back to the beginning of step 3.

(6) The last prompt is "FILE # TO STORE NEW DATA." The program will record data on whatever magnetic cartridge file is specified.

The above discussion should be enough to construct and operate this noise figure measurement system. Operators should realize, however, that accurate, repeatable results require good general measurement technique. Be sure to allow enough warm-up time for the measurement equipment and the units under test (usually 1 hour suffices). Connectors should be cleaned frequently—weekly is usually adequate. The most frequently used cleaner is isopropyl alcohol applied with a cotton swab. Make sure that all fibers from the swab are removed. It is good practice to frequently test connections by gently wiggling them while observing the signal level.

This chapter suggested which equipment to use to build a broadband microwave noise figure measurement system. Other equipment, however, may be used in most parts of the system because of different frequency ranges or because the other equipment is readily available. Assembling the equipment into a system was also discussed along with operating the system by using the special function keys. The text also described how to construct a cassette data file for storing the ENR calibration data for a particular noise source.

III. Software Description

The programs listed at the end of this chapter were constructed to demonstrate the flexibility of the system. With them the operator can select, for example, whether to sweep frequency or to measure at only a single frequency or whether to make a hard copy of the measurement results or not. The special function keys of the computer give the required flexibility by directing the main program through one or several modular subroutines that accomplish the desired tasks.

Many users of this system will find it best to modify the program. Consider, for example, a production test application where only a few measurement situations are encountered. In such a case the program could be changed, for example, so that it automatically loads the noise source calibration data into the computer memory without forcing the operator to answer with the file number where the data is stored. The various scales for plotting results can also be written into the program so the operator doesn't have to answer the same computer prompts over and over. Furthermore, the tasks performed by the special function keys may also be changed. Several of the keys, for example, might each direct the program to a specific CW measurement frequency for making tuning adjustments while observing the computer output of noise figure. Another key might be programmed to make a final broadband sweep at preprogrammed frequencies while both plotting the results and printing the measurement results at each frequency. The operator would not need to be asked the several questions about frequencies and plotting limits.

The purpose of this chapter is to describe the noise figure measurement software written for the 9825 Desktop Computer so that changes can be made for such special applications as described above or for different equipment. The first section explains the meaning of each of the variables used in the programs. Next comes a description of the subroutine interrelationships and then of the mechanism for directing measurements with the special function keys. The principal equations used to process the measured data are described. These are followed by an annotated listing of the programs.

Variables

The variables here do not include p variables because their use differs in each subroutine. Each use in a subroutine is localized to so few steps, that the local use can be deciphered from the program code at the end of this chapter.

Global Variables

A Programmed level of attenuation in dB.
B Start frequency in MHz.
C Stop frequency in MHz.
D Size of frequency steps in MHz.
E Current value of noise source ENR in dB.
F Current frequency in MHz.
G CW frequency in MHz.
I Counter for the frequency number being measured.
J Row index for ENR data being accessed.
K Counter for the frequency number being plotted.
L Desired local oscillator level in dBm.
M Mask word used to change bits of the state word.
N The number of frequencies being tested (0 for CW tests).
P Bit pattern used to change bits of the state word.
Q Desired state of the system.
S State of the system being executed.
T Effective input noise temperature in K.
U Wait time before proceeding after turning the noise source ON.
V Wait time before proceeding after turning the noise source OFF.
W Receiver noise figure in dB.
X (1) File number of noise source data to be loaded into memory. (2) Overall noise figure in dB of UUT/receiver. (3) A general variable used in measurement and calculation.
Y (1) Y factor just measured. (2) Gain of the unit under test (usually in dB). (3) A general variable used in measurement and calculation.
Z Corrected noise figure of the unit under test.

Flags

A flag value of 1 implies the condition stated below. Value 0 implies the negative (opposite) of the stated condition.

2 Calibration of the receiver at sweep frequencies is valid.
3 Calibration of the receiver at the CW frequencies is valid.
4 The headings of the columns for printed data are already printed.
5 The axes and labels for plotting noise figure versus frequency have been plotted.
6 The axes and labels for plotting gain versus frequency have been plotted.
7 A special function key that requires interrupting the present state of the system has been pressed.
9 Noise source ON and OFF power meter readings in dBm are to be printed on the strip printer for troubleshooting. This flag is set to 1 and back to zero by the live keyboard.
13 The CONTINUE key has been pressed without entering new data on the keyboard. This retains the old value of the data or uses a default answer.
14 The computer will ignore math errors, such as attempting to take the logarithm of negative numbers or to

divide by zero, and use the closest value it can without stopping the program.

15 A math error has occurred even if flag 14 was set.

r Variables

r0 ENR for external (manual) frequency tuning.

r1 Ambient temperature of the noise source in K.

r2 Hot noise source temperature in K.

r3 Left hand frequency of plotting scale in GHz.

r4 Right hand frequency of plotting scale in GHz.

r5 Tic interval in GHz for frequency axis.

r6 Number of tics per label on frequency axis.

r7 Bottom noise figure on plotting scale in dB.

r8 Top noise figure on plotting scale in dB.

r9 Tic interval in dB for noise figure axis.

r10 Number of tics per label on the noise figure axis.

r11 The number of frequencies at which the noise source is calibrated.

r12 The X coordinate of the lower-left plotting limit, P1, of the 9872A plotter.

r13 The Y coordinate of the lower-left plotting limit, P1, of the 9872A plotter.

r14 The X coordinate of the upper-right plotting limit, P2, of the 9872A plotter.

r15 The Y coordinate of the upper-right plotting limit, P2, of the 9872A plotter.

r16 Bottom gain on plotting scale in dB.

r17 Top gain on plotting scale in dB.

r18 Tic interval in dB for the gain axis.

r19 Number of tics per label on the gain axis.

r20 IF frequency in MHz (for mixer measurement program.)

r21 Character height as a percentage of the total height of the plotting area.

r22 Number of character heights for each margin.

r23 Character height in dB of noise figure units.

r24 Character height in dB of gain units.

r25 Character height in GHz of frequency units.

r26 Hot noise source temperature at the IF frequency (for mixer measurement program).

r27 Device number of the line printer for outputting data.

r28 Minimum attenuation allowed for proper measurements.

r29 Maximum attenuation allowed for proper measurements.

Arrays

A [20,6] Array of noise source data. Each row corresponds to a frequency. Column 1 contains the frequency in MHz. Column 2 contains the excess noise ratio in dB. Columns 3 and 4 contain the magnitude and phase of the noise source reflection coefficient when turned ON. Columns 5 and 6 contain the magnitude and phase of the noise source reflection coefficient when turned OFF.

C [0:101,3] Array for storing calibration run data. Column 1 contains the difference in watts between the noise output powers of the receiver for the noise source turned ON and OFF (N_2-N_1). Column 2 contains the effective input noise temperature of the receiver in Kelvins. Column 3 holds the value of P[A/10] that was in effect during the calibration run at each frequency. Each row of data corresponds to a measurement frequency with row 0 for the CW frequency, row 1 for the start frequency, row 2 for the second frequency in the sweep, etc. This array is C[2] for the mixer measurement program.

G [0:101] Array for storing the difference between the noise output power in watts of the UUT/receiver for the noise source turned ON and OFF ($N_2 - N_1$). During the calculation of corrected noise figure data for display on the computer LED readout, this array is changed to the gain of the UUT in dB. The elements correspond to the various frequencies of operation—element 0 for CW, element 1 for the start frequency, element 2 for the second frequency, etc.

P [0:8] An array for storing the noise power output in watts from the receiver for the allowed attenuator settings at the beginning of the calibration run. This effectively calibrates the attenuator at the IF frequency. Element 0 corresponds to the 0 dB setting, element 1 to 10 dB, element 2 to 20 dB, etc. P[8], used during mixer measurement, holds the value of P[A/10] that was in effect during the calibration run.

T [0:101] Array for storing the effective input noise temperature in Kelvins of the UUT/receiver, T_e. Element 0 is for the CW frequency, element 1 for the start frequency, element 2 for the second frequency, etc.

Strings

A$ [1] A string variable for storing “Y” or “N” entered from the keyboard in answer to computer prompts.

C$ [72] A string for reading the description of the noise source from the cassette data file.

Program Structure

An examination of the software at the end of this chapter shows that the programs are highly modularized into subroutines that are usually shorter than ten lines of program code. Each subroutine performs at least one function, but it may call another subroutine to help perform that function or to perform the next function. **Figure 3-1** shows which routines call which subroutines. After finishing its task, the subroutine is programmed to either repeat that task or return to the routine that called it. Some routines frequently examine whether one of the special function keys has been pressed by testing flag 7. If flag 7 is set to one, the subroutine goes back to the routine entitled START.

The program finds its way through the several desired tasks by examining the state word S. This is a 16 bit integer number where each of the bits from 0 to 11 represents a specific task. The task is to be performed if the bit is 1 and is not to be performed if the bit is 0. **Figure 3-2** shows the meaning of each bit and the effect each special function key has on each of the bits of S. The special function keys not only initialize tasks by setting the required bits to 1, but also assure that conflicting tasks are not performed by setting certain other bits to zero. The various parts of the program then only need to examine various bits of S to see how to proceed.

In actual fact, most of the special function keys operate on the 16 bit number Q as an intermediate variable to save the desired state until the program completes its current task. Then the program transfers the bits from Q to S. Otherwise, pressing a special function key might change the bits of S in the middle of a task and the program would get confused about what it was doing.

The method used to change only certain bits of the number S or Q is to use a mask number, M, which indicates

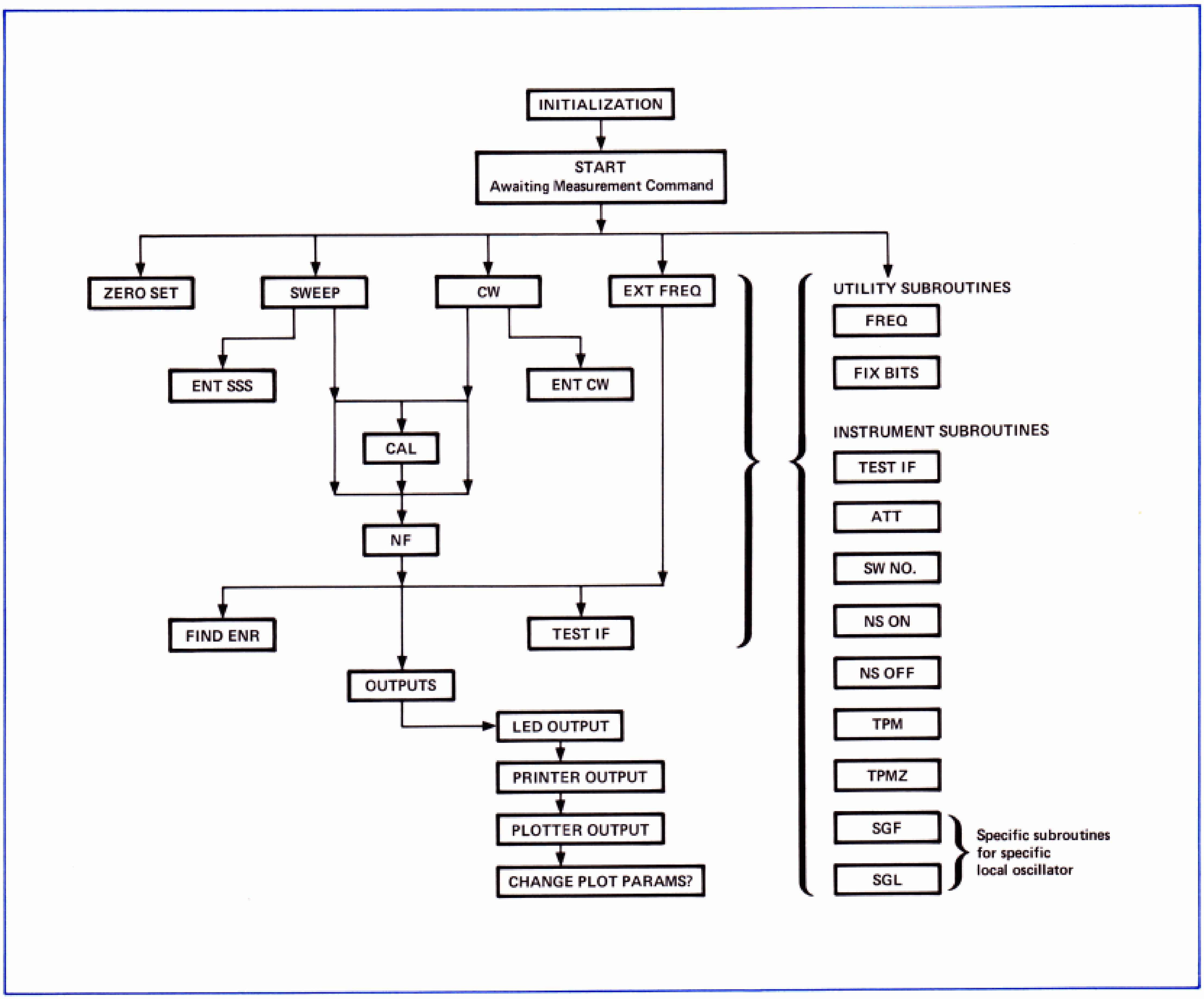


Figure 3-1. Chart of the subroutine structure for the measurement programs.

with 1's which bits are to be set to 1 or 0, and to use a bit pattern, P, which contains the actual pattern of 1's and 0's that are to be inserted into the bits indicated by M. Thus wherever M has a 1, the new value of S or Q will contain the same bit value as P; wherever M has a zero, the value of S or Q will retain its old value.

While following the listing at the end of this chapter, consider the case that the program has been measuring at the CW frequency and printing results on the line printer. S therefore contains the number 000 010 000 001 (= 201_8) and Q contains the number 000 000 000 001 (= 1_8). The program keeps circulating through the CW and NF subroutines. Now consider that the operator pushes the special function labeled ENTER SSS. Then M becomes 111 000 011 111 (= 7037_8) and P becomes 000 000 000 110 (= 6_8). After the binary operations of the special function program are performed, Q is changed to 000 000 000 110 (= 6_8) and flag 7 is set. The program finishes the CW measurement in progress and prints the results because S hasn't changed yet. After the CW results are printed the NF subroutine finds that flag 7 is set and therefore returns to the START routine. In the first line of START, with the aid of subroutine FIX BITS, most of the bits of S are changed to be just like the bits of Q. Bits 7 and 8 of S, which indicate whether or not to plot results and/or print them on the line printer, are not changed. Therefore S has the value 000 010 000 110 (= 206_8). The START routine sees that bit #1 of S is 1 so it calls the SWEEP subroutine. The SWEEP subroutine sees that bit #2 of S is 1 so it calls the ENT SSS subroutine. At the end of subroutine ENT SSS, new values of M and P along with subroutine FIX BITS operate on S to change S to be 000 010 000 010 (= 202_8). Then the SWEEP subroutine executes, measuring and printing the noise figure at each of the sweep

	TASK PERFORMED IF BIT IS 1													
	NEW ENR FOR EXT FREQ	STOP WITH SOURCE OFF	STOP WITH SOURCE ON	PLOT (1) or NO PLOT (0)	PRINT (1) or NO PRINT (0)	NEW CAL	NF & G (1) or NF ONLY (0)	ZERO SET	EXT FREQ	ENTER NEW FREQS	SWEEP	CW	MASK M_8	BIT PATTERN P_8
BIT NUMBER	11	10	9	8	7	6	5	4	3	2	1	0		
SPECIAL FUNCTIONS														
SWEEP	0	0	0	X	X	X	X	0	0	0	1	0	7037	2
CW	0	0	0	X	X	X	X	0	0	0	0	1	7037	1
EXT FREQ	X	0	0	X	X	X	0	0	1	0	0	0	3077	10
NF ONLY	X	X	X	X	X	X	0	X	X	X	0	0	43	0
CONT.0 (ALSO RUN)	X	1	0	X	X	0	X	0	0	0	0	0	3137	2000
NO PRINT	X	X	X	X	0	X	X	X	X	X	X	X	200	0
NO PLOT	X	X	X	0	X	X	X	X	X	X	X	X	400	0
NOISE ON	0	0	1	X	X	X	X	0	0	0	0	0	7037	1000
NOISE OFF (STOP)	0	1	0	X	X	X	X	0	0	0	0	0	7037	2000
ENT SSS	0	0	0	X	X	X	X	0	0	1	1	0	7037	6
ENT CW	0	0	0	X	X	X	X	0	0	1	0	1	7037	5
ENT EXT ENR	1	0	0	X	X	X	X	0	1	0	0	0	7037	4010
NF & G	0	X	X	X	X	X	1	X	X	X	0	0	4043	40
NEW CAL	0	X	X	X	X	1	X	X	X	X	X	X	4100	100
ZERO SET	X	0	0	X	X	X	X	1	0	X	0	0	3033	20
PRINT	X	X	X	X	1	X	X	X	X	X	X	X	200	200
PLOT	X	X	X	1	X	X	X	X	X	X	X	X	400	400

Figure 3-2. Key for setting the various bits of the state number S and its preliminary storage number Q. Those bits shown as X are not changed by that special function operation and retain their old value.

frequencies. After the sweep, M, P, and subroutine FIX BITS change S to be 000 010 000 000 (= 200_8). Subroutine SWEEP then returns to the START routine. START examines certain bits of S but finds all of those bits are zero so that START displays the AWAITING MEASUREMENT COMMAND message on the computer.

Mathematical Relations

This section describes the equations used in the programs of this application note. The hot (turned ON) temperature of the noise source in kelvins, T_H, can be expressed in terms of its excess noise ratio ENR, in dB by

$$T_H = (10^{\frac{ENR}{10}} + 1)\,290 \qquad (3\text{-}1)$$

The Y factor is the ratio of the measured noise power output for the hot noise source, N_2, to the noise power output for the cold noise source, N_1.

$$Y = \frac{N_2}{N_1} \qquad (3\text{-}2)$$

The effective input noise temperature, T_e, is calculated from the Y factor, hot noise source temperature, T_H, and cold (turned OFF) noise source temperature, T_C according to:

$$T_e = \frac{T_H - T_C Y}{Y - 1} \qquad (3\text{-}3)$$

where all temperatures are in kelvins.

The following relates the noise figure, F, in dB to the effective input noise temperature

$$F = 10 \log \left(\frac{T_e}{290} + 1 \right) \qquad (3\text{-}4)$$

Correcting the measured noise figure for the noise contributed by the measurement system involves measuring the system noise contribution and the gain of the unit under test. The effective input noise temperature of the measurement system is measured the same way as when the unit under test is connected. Thus the equations above also apply during the calibration run, but more relationships are needed. The T_e of the unit under test, denoted as T_{e1}, is related to the overall T_e, denoted by T_{e12}, and the T_e of the measurement system, denoted by T_{e2}, by

$$T_{e1} = T_{e12} - \frac{T_{e2}}{G_1} \qquad (3\text{-}5)$$

where G_1 is the available gain of the unit under test expressed as a ratio (not in dB). In terms of noise figure expressed as a ratio, equation (3-5) becomes

$$F_1 = F_{12} - \frac{F_2 - 1}{G_1} \qquad (3\text{-}6)$$

Gain is found by measuring the difference between the noise source ON and noise source OFF noise power outputs. During calibration this difference is

$$(N_2 - N_1)|_{Cal} = kBG_{2NS}\,(T_H - T_C)|_{Cal} \qquad (3\text{-}7)$$

where k is Boltzmann's constant, B is the measurement bandwidth, G_{2NS} is the available gain of the measurement equipment up to the power detecting apparatus, and T_H and T_C correspond to the hot and cold noise source temperatures respectively. G_{2NS} refers to the available gain for a source impedance corresponding to the noise source impedance. The noise added by the measurement equipment disappears in the process of taking the difference of the noise powers. The difference in noise powers is measured again for the unit under test (UUT) with the measurement equipment. Now

$$(N_2 - N_1)|_{Meas} = kBG_1\,G_{2UUT}\,(T_H - T_C)|_{Meas} \qquad (3\text{-}8)$$

where G_1 is the available gain of the UUT for a source impedance corresponding to the noise source impedance and G_{2UUT} is the available gain of the measurement equipment for a source impedance corresponding to the output impedance of the UUT.

Taking the ratio of the two noise power differences yields

$$\frac{(N_2 - N_1)|_{Meas}}{(N_2 - N_1)|_{Cal}} = \frac{G_1 G_{2UUT}\,(T_H - T_C)|_{Meas}}{G_{2NS}\quad (T_H - T_C)|_{Cal}} \qquad (3\text{-}9)$$

Wherever the calibration run and measurement frequency are the same, such as for a preamplifier, the numerator and denominator values of T_H and T_C are the same so they cancel. Furthermore, the program assumes that both the noise source and the output impedance of the UUT are identical (usually 50 ohms). This means G_{2UUT} is equal to G_{2NS} and they cancel. The ratio of the power differences indicated on the left side of equation (3-9) then yields the available gain G_1 of the unit under test. That value of G_1 can then be used in equation (3-5) to give the effective input noise temperature of the UUT alone.

The noise figure measurements discussed so far are all double sideband measurements. The measured result is an average of the noise figure at two frequencies given by the plus and minus in

$$f = f_{lo} \pm f_{if}$$

where f_{lo} is the local oscillator frequency and f_{if} is the frequency of the IF amplifier. The double sideband noise figure is not the proper one whenever the final application of the device allows (1) desired signals in only one sideband and (2) amplified input noise in both sidebands. Since input noise is processed from both sidebands, but signals from only one, the ability to process low-level signals is poorer and the noise figure should be higher. This situation often occurs with mixers and receivers that do not have some mechanism for rejecting noise in the undesired sideband.

A traditional treatment for characterizing mixers has been to assume that the noise figure and loss of a mixer are equal and that the single sideband noise figure of the mixer is 3 dB poorer than the double sideband noise figure. If these assumptions are used in equation (3-6), the result is

$$\begin{aligned} F_{1SSB}\,(\mathrm{dB}) &= F_{1DSB}\,(\mathrm{dB}) + 3\ \mathrm{dB} \\ &= F_{12}\,(\mathrm{dB}) - F_2\,(\mathrm{dB}) + 3\ \mathrm{dB} \end{aligned} \qquad (3\text{-}10)$$

F_{1SSB} and F_{1DSB} are the single and double sideband noise figures of the mixer, F_{12} is the measured overall double sideband noise figure, and F_2 is the noise figure of what follows the mixer, usually the IF amplifier. This treatment is often erroneous because the noise figure and loss of a mixer are generally not equal.

With this noise figure measurement system the noise figure and loss of the mixer may be measured independently. In this case, a slight modification is made to the program because (1) the calibration of the system takes place at one frequency only (the IF frequency) and (2) the final noise figure and loss (rather than gain) calculations are slightly different. During calibration the following measurements are stored (see equations (3-3) and (3-5))

$$T_{e2} = \frac{T_H - T_C Y_{Cal}}{Y_{Cal} - 1} \qquad (3\text{-}11)$$

$$(N_2 - N_1)|_{Cal} = kBG_{2NS}(T_H - T_C)|_{Cal} \qquad (3\text{-}12)$$

During measurement let G_1 be the available gain of UUT in a single sideband and T_e the effective input noise temperature of the UUT. The measurement is double sideband so that the following apply:

$$N_2 = kT_H 2BG_1G_2 + kT_e 2BG_1G_2 + kT_{e2}BG_2 \qquad (3\text{-}13)$$

$$N_1 = kT_C 2BG_1G_2 + kT_e 2BG_1G_2 + kT_{e2}BG_2 \qquad (3\text{-}14)$$

where G_2 refers to the available gain of the IF for the UUT as its source impedance. Therefore,

$$\left.\frac{N_2}{N_1}\right|_{Meas} = \frac{2T_HG_1 + 2T_eG_1 + T_{e2}}{2T_CG_1 + 2T_eG_1 + T_{e2}} = Y_{Meas} \qquad (3\text{-}15)$$

$$T_e = \frac{T_H - T_C Y_{Meas}}{Y_{Meas} - 1} - \frac{T_{e2}}{2G_1} \qquad (3\text{-}16)$$

$$(N_2 - N_1)|_{Meas} = k2BG_1G_{2UUT}(T_H - T_C)|_{Meas} \qquad (3\text{-}17)$$

$$\frac{(N_2 - N_1)|_{Meas}}{(N_2 - N_1)|_{Cal}} = \frac{G_{2UUT}}{G_{2NS}}\,2G_1\,\frac{(T_H - T_C)|_{Meas}}{(T_H - T_C)|_{Cal}} \qquad (3\text{-}18)$$

The T_H during measurement and calibration is usually not the same because the noise source output is generally different at the IF and signal frequencies. If the output impedance of the UUT is equal to the impedance of the noise source, $G_{2NS} = G_{2UUT}$. Solving equation (3-18) for conversion loss yields

$$L_1 = \frac{1}{G_1} = 2\,\frac{(N_2 - N_1)|_{Cal}}{(N_2 - N_1)|_{Meas}}\,\frac{(T_H - T_C)|_{Meas}}{(T_H - T_C)|_{Cal}} \qquad (3\text{-}19)$$

In order to apply the double sideband noise figure of a UUT during single sideband applications, a slight modification of the noise figure definition of equation (1-2) in Chapter 1 is necessary. The numerator of F, representing the total noise power output, should use the actual noise bandwidth of the UUT. The denominator of F, however, representing the ideal noise power output, should use only the bandwidth over which signals will come into the UUT. Thus, expressing F as a ratio rather than in dB,

$$\begin{aligned} F &= \frac{kT_eB_{total}G_a + kT_oB_{total}G_a}{kT_oB_{sig}G_a} \\ &= \left(\frac{T_e}{T_o} + 1\right)\frac{B_{total}}{B_{sig}} \end{aligned} \qquad (3\text{-}20)$$

For this case of a mixer the total bandwidth includes both sidebands and the signal bandwidth includes only one. This means

$$B_{total} = 2B_{sig} \qquad (3\text{-}21)$$

Now equation (3-20) can be expressed as

$$F(\mathrm{dB}) = 3\ \mathrm{dB} + 10\log\left(\frac{T_e}{T_o} + 1\right) \qquad (3\text{-}22)$$

These are the equations used for mixers in the accompanying program.

Annotated listing of Noise Figure and Gain Program.

```
0: "346B/436A AUTOMATIC CORRECTED NOISE FIGURE & GAIN PROGRAM;800425":
1:
2: "INITIALIZATION":time 5000;cli 7;clr 7;lcl 7
3: otd3137→M;otd2000→P;cll 'FIX BITS'(M,P,S);S→Q ———— Initialize system state
4: if r0;gto "START" ———— Senses whether all variables need to be initialized
5: dev "tpm",713,"lo",719,"ns",707,"att",728,"pl",705
6: psc 705;cfg ;6→r27 ———— If no plotter, change to "psc φ;"; device number of printer
7: dim A$[1],C$[72],A[20,6],C[0:101,3],G[0:101],T[0:101],P[0:7];20→r11
8: "noise source wait times":50→V;0→U ———— Power supply response times
9: "IF output atten limits":10→r28;30→r29;r28→A ———— Change limits to suit IF amp
10: ldk 1;if r0=0;15.2→r0 ———— Loads special function keys from file #1; default ENR
11: ent "FILE # OF ENR DATA?",X;ldf X,C$,A[*] ———— Load ENR data
12: 290→r1;ent "AMBIENT TEMP IN K",r1
13:
14: "START":if flg7;otd7177→M;cll 'FIX BITS'(M,Q,S);cfg 7;pen  ┐
15: if bit(4,S);dsp ;cll 'ZERO SET'                            │
16: if bit(1,S);dsp ;cll 'SWEEP'                               │
17: if bit(0,S);dsp ;cll 'CW'                                  │
18: if bit(3,S);dsp ;lcl 7;cll 'EXT FREQ'                      │  Branch to subroutines
19: if bit(9,S);cll 'NS ON';lcl 7                              ├— according to special
20: otd1000→M;0→P;cll 'FIX BITS'(M,P,S)                        │  function key command
21: if bit(10,S);cll 'NS OFF';lcl 7                            │
22: otd2000→M;0→P;cll 'FIX BITS'(M,P,S)                        │
23: dsp "AWAITING MEASUREMENT COMMAND"                          │
24: gto "START"                                                │
25: end                                                        ┘
26:
27: "ZERO SET":rem 7;cll 'NS OFF';cll 'ATT'(110);cll 'TPMZ';cll 'ATT'(A)
28: 20→M;0→P;cll 'FIX BITS'(M,P,S);'TPM'→X;ret ———— Zero-set completed
29:
30: "SWEEP":
31: if bit(2,S);cll 'ENTER SSS' ———— Go get new freqs if desired
32: if B=0;cll 'ENTER SSS' ———— If freqs were never entered, go get freqs
33: if bit(5,S);if flg2=0;otd100→M→P;cll 'FIX BITS'(M,P,S) ———— If cal'n is necessary, set bit 6
34: int((C-B)/D)+1→N ———— Calculate number of freqs to be tested
35: gto +2;if bit(6,S);cll 'CAL';gto +1 ———— Go calibrate if needed
36: dsp "CONNECT DEVICE UNDER TEST";lcl 7;stp
37: rem 7;cll 'SGF'(B);10→L;cll 'SGL'(L);cll 'ATT'(A) ———— Initialize signal freq and level
38: cll 'NF' ———— Go measure noise figure at all freqs
39: 2→M;0→P;cll 'FIX BITS'(M,P,S);ret ———— Measurements completed; return to initialize state
40:
41: "ENTER SSS":cfg 13;ent "START FREQ(MHz)?",B;if flg13=0;cfg 2
42: ent "STOP FREQ(MHz)?",C;if flg13=0;cfg 2
43: ent "FREQ STEP(MHz)?",D;if flg13=0;cfg 2
44: int((C-B)/D)+1→N;if N>101;gto "ENTER SSS" ———— Assure not too many freqs for array size
45: 4→M;0→P;cll 'FIX BITS'(M,P,S);ret ———— Freq entry completed
46:
47: "CW":if bit(2,S);cll 'ENTER CW' ———— Go get new freq if desired
48: if G=0;cll 'ENTER CW' ———— If CW freq was never entered, go get freq
49: if bit(5,S);if flg3=0;otd100→M→P;cll 'FIX BITS'(M,P,S) ———— If cal'n is necessary, set bit 6
50: 0→N
51: gto +2;if bit(6,S);cll 'CAL';gto +1 ———— Go calibrate if needed
52: dsp "CONNECT DEVICE UNDER TEST";lcl 7;stp
53: rem 7;cll 'SGF'(G);10→L;cll 'SGL'(L);cll 'ATT'(A) ———— Initialize signal freq and level
54: cll 'NF';gto "CW" ———— Measure noise figure and repeat CW measurement
55:
56: "ENTER CW":cfg 13;ent "CW FREQ(MHz)",G;if flg13=0;cfg 3
57: 4→M;0→P;cll 'FIX BITS'(M,P,S);ret ———— Freq entry completed
58:
59: "EXT FREQ":rem 7
60: if bit(11,S);cll 'NEW EXT ENR' ———— Go get new ENR if desired
61: cll 'TEST IF'(X,Y) ———— Go adjust and measure IF output for source ON and OFF
62: if flg9;fmt f2.0,2f7.2;wrt 16,A,10log(X1e3),10log(Y1e3) ———— If flag 9, print outputs in dBm
63: "hot noise source temp":(tn↑(r0/10)+1)290→r2
64: "Y factor":X/Y→Y;max(Y,1.0033113)→Y ———— Limit max NF to 40 dB (Y to 0.14 dB)
65: "equiv input noise temp":(r2-r1Y)/(Y-1)→T
66: "noise figure":10log(T/290+1)→X
67: "outputs":fmt "NF=",f5.2,"  ENR=",f4.1;wrt 0,X,r0 ———— LED output
68: if bit(7,S);wrt r27,X,r0 ———— If desired, print result
69: if flg7;gto "START" ———— Test for special function key interrupt
70: gto "EXT FREQ" ———— Repeat external frequency measurement
```

```
71:
72: "NEW EXT ENR":fmt "ENR (now=",f5.2,")?";wrt 0,r0;ent "",r0
73: 4000→M;0→P;cll 'FIX BITS'(M,P,S);ret ——— ENR entry completed
74:
75: "CAL":dsp "REMOVE DUT (CONNECT NS TO RCVR)";stp
76: cll 'ZERO SET';if N=0;cfg 3 ——— If CW freq, clear CW cal'n valid flag
77: if N;cfg 2 ——— If SWEEP, clear sweep cal'n valid flag; next initialize freq and level
78: bit(1,S)→I;rem 7;cll 'SGF'('FREQ'(I));10→L;cll 'SGL'(L);cll 'NS ON'
79: for A=r28 to r29 by 10;cll 'ATT'(A);'TPM'→X→P[A/10] ——— Step atten and store power readings
80: if A=r28;if X>1e-5;beep;dsp "HI GAIN OR LO ATT! PROG STOPPED!";stp
81: if flg9;fmt f2.0,f10.2;wrt 16,A,10log(X)+30 ——— If flag 9, print output in dBm
82: next A
83: r28→A;cll 'ATT'(A);cll 'NF' ——— Go measure system noise contribution at each freq
84: if N=0;sfg 3 } ——— Calibration is now valid; set flag
85: if N;sfg 2
86: otd100→M;0→P;cll 'FIX BITS'(M,P,S);cll 'FIX BITS'(M,P,0);beep;ret ——— Cal completed
87:
88: "NF":for I=bit(1,S) to N;'FREQ'(I)→F;cll 'SGF'(F) ——— Set freq
89: cll 'TEST IF'(X,Y) ——— Go adjust and measure IF output for source ON and OFF
90: if flg9;fmt f2.0,2f7.2;wrt 16,A,10log(X1e3),10log(Y1e3) ——— If flag 9, print output in dBm
91: X-Y→G[I];cll 'FIND ENR'(E) ——— Store power output difference; go find proper ENR
92: "hot noise source temp":(tn↑(E/10)+1)290→r2
93: "Y factor":X/Y→Y;max(Y,1.0033113)→Y ——— Limit max NF to 40 dB (Y to 0.14 dB)
94: "equiv input noise temp":(r2-r1Y)/(Y-1)→T→T[I]
95: cll 'OUTPUTS' ——— Output results
96: if bit(6,S);G[I]→C[I,1];T→C[I,2];P[A/10]→C[I,3] ——— If needed, store calibration data
97: if flg7;gto "START" ——— Test for special function key interrupt
98: next I ——— Repeat at next freq
99: ret ——— Measurements completed
100:
101: "FIND ENR":
102: max(J,1)→J;if F<A[J,1];J-1→J;gto +0;if J=0;A[1,2]→p1;ret
103: if J=r11;A[J,2]→p1;ret
104: if F>A[J+1,1];J+1→J;gto -2
105: "interpolate":(F-A[J,1])(A[J+1,2]-A[J,2])/(A[J+1,1]-A[J,1])+A[J,2]→p1
106: ret
107:
108: "-------------------------------------------------------------":
109: "UTILITY SUBROUTINES":
110: "FREQ":if bit(0,S);G→p2;ret p2
111: B+(p1-1)D→p2;ret p2
112:
113: "FIX BITS":ior(band(cmpp1,p3),band(p1,p2))→p3;ret
114:
115: "-------------------------------------------------------------":
116:
117: "OUTPUTS":
118: "LED output":
119: "noise fig":10log(T/290+1)→X
120: if bit(6,S);fmt f5.0," MHz;NF=",f6.2," dB";wrt 0,F,10log(T/290+1);ret ——— Display during cal'n
121: if bit(5,S);gto "LED NF&G" ——— Different output display for noise figure and gain
122: fmt f5.0," MHz   NF=",f6.2," dB"
123: wrt 0,F,X;gto "Printer output" ——— Display freq and NF; go see about printing result
124: "LED NF&G":sfg 14;G[I]C[I,3]/C[I,1]/P[A/10]→Y;10log(C[I,2]/290+1)→W
125: 10log((T-C[I,2]/Y)/290+1)→Z;10log(Y)→Y→G[I];cfg 14
126: fmt f5.0,"MHz;NF=",f5.2,"dB;G=",f5.2,"dB"
127: wrt 0,F,Z,Y ——— Display freq, noise figure, and gain
128:
129: "Printer output":if bit(7,S)=0;gto "Plotter output" – If no printing, go see about plotting
130: if I=1;cfg 4 ——— At first freq get ready to print heading
131: if bit(5,S);gto "NF&G Printout" ——— Different printout for noise figure and gain
132: if flg4;gto "Print NF" ——— If heading is already printed, skip to print result
133: fmt "Freq(MHz)   NF(dB)",/;wrt r27;sfg 4 ——— Print heading
134: "Print NF":fmt f7.0,f10.2;wrt r27,F,X ——— Print freq and noise figure
135: gto "Check end of print"
136: "NF&G Printout":if flg4;gto "Print NF&G" ——— If heading is already printed, print results
137: fmt "Freq(MHz)  NF(dB)   G(dB)   Tot NF   rcvrNF",/;wrt r27;sfg 4 ——— Print heading
138: "Print NF&G":fmt f5.0,4x,8f8.2;wrt r27,F,Z,Y,X,W ——— Print results
139: "Check end of print":if I#N;gto "Plotter output" } ——— Leave blank lines at last freq
140: if I#0;fmt 3/;wrt r27
141:
```

```
142: "Plotter output":if bit(1,S)=0;ret ———————— No plots unless sweeping
143: if bit(8,S)=0;ret ———————— If no plot is wanted, return
144: if I=1;cll 'CHANGE PLOT PARAMS?'      }
145: if flg5=0;cll 'CHANGE PLOT PARAMS?'   }———— At first freq check about plotting axes
146: scl r3-r22r25,r4+r22r25,r7-r22r23,r8+2r23 }
147: lim r3,r4,r7,r8+2r23                      }———— Scale for noise figure vs. freq
148: pen# 4;line ;if bit(5,S);gto "Plot corrected NF" ———— Plot solid line
149: plt F/1000,X ———————— Plot measured noise figure vs. freq
150: if I=N;pen ———————— At last freq raise pen
151: ret ———————— Plot completed
152:
153: "Plot corrected NF":plt F/1000,Z ———————— Plot corrected noise figure vs. freq
154: if I#N;ret ———————— Return unless this is the last freq (at last freq maybe more plots)
155: if flg7;gto "START" ———————— Test for special function key interrupt
156:
157: pen;cfg 13;ent "UNCORRECTED NF PLOT (Y or N)?",A$
158: if flg13;"N"→A$ ———————— Default answer is "N"
159: if cap(A$)="N";gto "Gain Plot?"
160: if cap(A$)#"Y";gto -3
161: pen# 4;line 2,1 ———————— Plot with dashed line
162: for K=1 to N;'FREQ'(K)→F ———————— Find freq
163: plt F/1000,10log(T[K]/290+1);next K ———————— Plot measured noise figure vs. freq
164: pen;line ———————— Next plots with solid line
165: if flg7;gto "START" ———————— Test for special function key interrupt
166:
167: "Gain Plot?":cfg 13;ent "PLOT GAIN (Y or N)?",A$
168: if flg13;"Y"→A$ ———————— Default answer is "Y"
169: if cap(A$)="N";ret
170: if cap(A$)#"Y";gto -3
171: "Plot Gain":if flg6=0;cll 'Gain axis' ———————— If there is no gain axis, go plot one
172: scl r3-r22r25,r4+r22r25,r16-r22r24,r17+2r24 }
173: pen# 3;lim r3,r4,r16,r17+2r24               }———— Scale for gain vs. freq
174: for K=1 to N;'FREQ'(K)→F ———————— Find freq
175: plt F/1000,G[K];next K ———————— Plot gain vs freq
176: pen;ret ———————— Plotting completed
177:
178: "CHANGE PLOT PARAMS?":if r21=0;gto "Default freq axis"
179: cfg 13;ent "NEW PAPER (Y or N)?",A$
180: if flg13;"N"→A$ ———————— Default answer is "N"
181: if cap(A$)="N";sfg 5;ret
182: if cap(A$)#"Y";gto -3
183: cfg 5;cfg 6;gto "Old axes OK?" ———————— Clear axes OK flag; go check on desired scales
184:
185: "Default freq axis":int(B/1000)→r3
186: int(C/1000)→r4;if C/1000#r4;1+r4→r4
187: .5→r5;if r4-r3<7;.2→r5
188: 1/r5→r6
189: "Default NF axis":3→r7;13→r8;.2→r9;5→r10
190: "Default gain axis":0→r16;30→r17;1→r18;5→r19;gto "Check each axis"
191:
192: "Old axes OK?":cfg 13;ent "ARE OLD AXES OK (Y or N)?",A$
193: if flg13;"Y"→A$ ———————— Default answer is "Y"
194: if cap(A$[1])="Y";gto "Label axes"
195: if cap(A$[1])#"N";gto -3
196:
197: "Check each axis":fmt "Min freq in GHz (now=",f5.2,")?"
198: wrt 0,r3;ent "",r3
199: fmt "Max freq in GHz (now=",f5.2,")?";wrt 0,r4;ent "",r4
200: fmt "Tic interval in GHz (now=",f5.2,")?";wrt 0,r5;ent "",r5
201: fmt "#tics/label (now=",f2.0,")?";wrt 0,r6;ent "",r6
202: fmt "Min Noise Figure in dB (now=",f2.0,")?";wrt 0,r7;ent "",r7
203: fmt "Max Noise Figure in dB (now=",f2.0,")?";wrt 0,r8;ent "",r8
204: fmt "Tic interval in dB (now=",f5.2,")?";wrt 0,r9;ent "",r9
205: fmt "#tics/label (now=",f5.2,")?";wrt 0,r10;ent "",r10
206: fmt "Min Gain in dB (now=",f2.0,")?";wrt 0,r16;ent "",r16
207: fmt "Max Gain in dB (now=",f2.0,")?";wrt 0,r17;ent "",r17
208: fmt "Tic interval in dB (now=",f2.0,")?";wrt 0,r18;ent "",r18
209: fmt "#tics/label (now=",f5.2,")?";wrt 0,r19;ent "",r19
210:
```

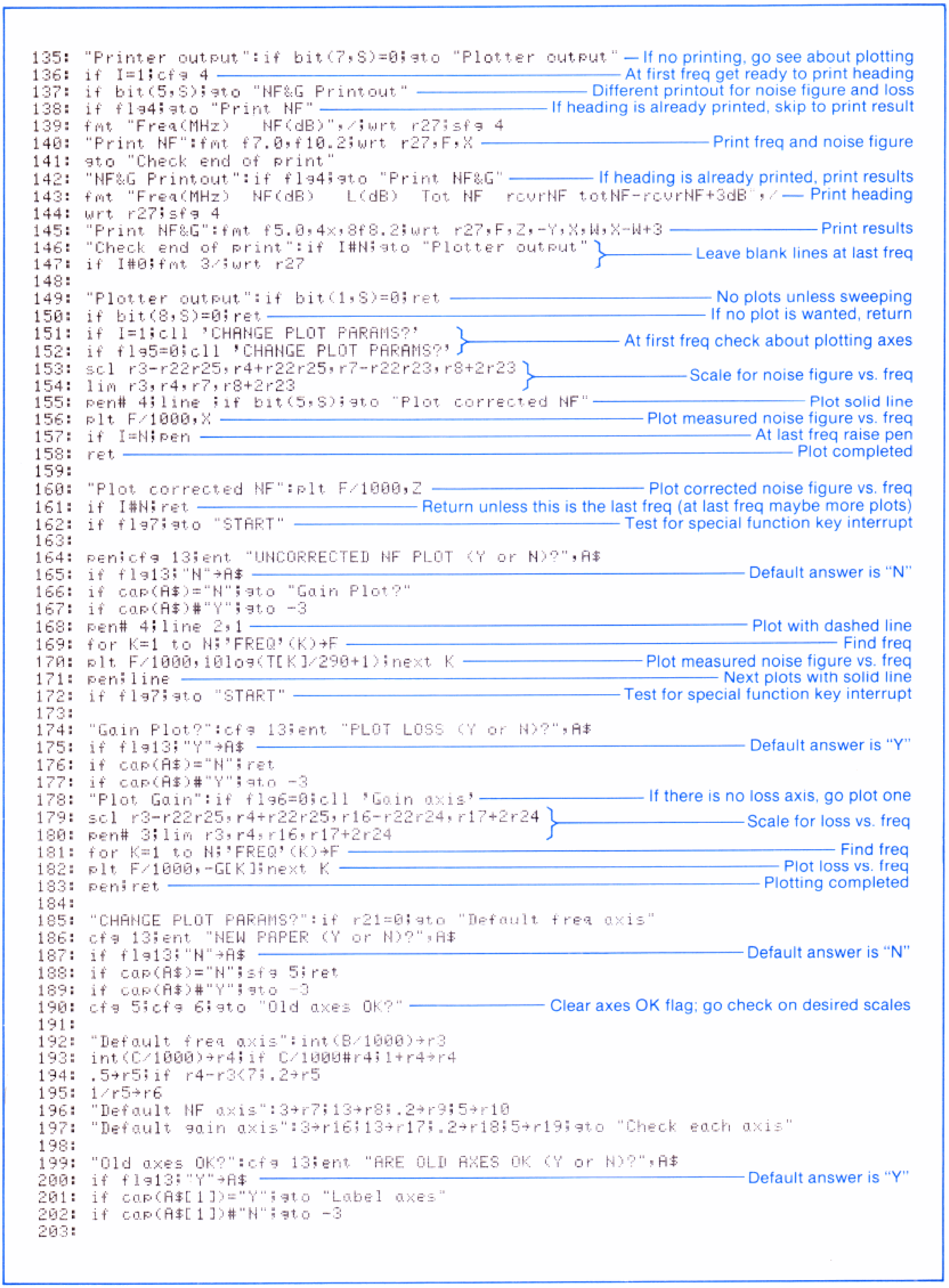
```
211: "Label axes":pclr;fmt ;wrt "pl","OP";red "pl",r12,r13,r14,r15 —— Find corners of plot
212: "# of character heights per margin":5→r22;1.5→r21
213: "Character height in NF units":(r8-r7)/(100/r21-r22)→r23
214: "Character height in Gain units":(r17-r16)/(100/r21-r22)→r24
215: "Character height in Freq units":100(r14-r12)/r21(r15-r13)-2r22→r25
216: (r4-r3)/r25→r25
217:
218: "Freq axis":scl r12,r14,r13,r15;lim r12,r14,r13,r15
219: csiz r21,2,(r15-r13)/(r14-r12),0
220: pen# 2;plt (r14+r12)/2,r13,1;cplt -8,.5;lbl "Frequency (GHz)"
221: scl r3-r22r25,r4+r22r25,r7-r22r23,r8+2r23
222: xax r7,r5,r3,r4,r6
223:
224: "NF axis":scl r12,r14,r13,r15;csiz r21,2,(r15-r13)/(r14-r12),90
225: pen# 4;plt r12,(r15+r13)/2,1;cplt -9,-1.1;lbl "Noise Figure (dB)"
226: scl r3-r22r25,r4+r22r25,r7-r22r23,r8+2r23
227: yax r3,r9,r7,r8,r10;sfg 5;ret
228:
229: "Gain axis":scl r12,r14,r13,r15;lim r12,r14,r13,r15
230: csiz r21,2,(r15-r13)/(r14-r12),90
231: pen# 3;plt r14,(r15+r13)/2,1;cplt -5,.5;lbl "Gain (dB)"
232: scl r3-r22r25,r4+r22r25,r16-r22r24,r17+2r24
233: yax r4,r18,r16,r17
234: yax r4+3r25,r18,r16,r17,-r19;sfg 6;ret
235:
236: "------------------------------------------------------------------------":
237:
238: "INSTRUMENT SUBROUTINES":
239: "TEST IF":cll 'NS ON';'TPM'→p1 ———————— Go measure ON noise power
240: if p1<1e-5;jmp 3 ———————— OK if pl < -20 dBm, so go measure OFF power
241: if A=r29;beep;dsp "I-F SATURATED! PROGRAM STOPPED";lcl 7;end ———— Error
242: A+10→A;cll 'ATT'(A);gto "TEST IF" ———————— Raise atten and repeat measurement
243: cll 'NS OFF';'TPM'→p2 ———————— Measure OFF noise power
244: if p2>1.258e-8;ret ———————— OK if > -49 dBm so return to calling subroutine
245: if A=r28;jmp 3 ———————— Do not lower atten beyond min allowed value
246: A+10int((10log(p1*1e3)+22)/10)→A   }
247: max(r28,A)→A;cll 'ATT'(A);gto "TEST IF"   } ———— Lower atten and repeat measurements
248: if p2<5e-10;beep;dsp "LOW GAIN! PROGRAM STOPPED";lcl 7;end ———— Error
249: fmt "SYSTEM GAIN MARGINAL AT",f5.0,"MHz" ———————— Warning
250: wrt 0,F;beep;wait 1000 ———————— Display freq of marginal gain
251: ret ———————— Measurement complete
252:
253: "ATT":int(p1)→p1 ———————— General subroutine for Model 11713A
254: min(p1,121)→p1;max(0,p1)→p1
255: int(p1/10)→p2;if p2=12;11→p2
256: fxd 0;cll 'swt#'(p2,1) ———————— 10 dB step atten connected to "X" output
257: p1-10p2→p3
258: cll 'swt#'(p3,5);wait 20;ret
259:
260: "swt#":p1→p3;if p1>7;p1+4(p3>3)→p3
261: if bit(p6,p3);p2+p6+10p5→p5;jmp 2
262: p2+p6+10p4→p4
263: p6+1→p6;if p6<4;jmp -2
264: if p4;if p5;fmt "A",f4.0,"B",f4.0;wrt "att",p5,p4;ret
265: fmt "A",f4.0;if p5;wrt "att",p5;ret
266: fmt "B",f4.0;wrt "att",p4;ret
267:
268: "NS ON":rem 7;fmt c,z;wrt "ns","2560";wait U;ret ———— Subroutine for Model 6002A-001
269:
270: "NS OFF":rem 7;fmt c,z;wrt "ns","2000";wait V;ret ———— Subroutine for Model 6002A-001
271:
272: "TPM":fmt ;wrt "tpm","9A+T";fmt 3b,e9.0;red "tpm",p1,p2,p3,p4 ———— Read power
273: fmt ;wrt "tpm","9A+T";fmt 3b,e9.0;red "tpm",p1,p2,p3,p4            subroutine
274: if p2>73;if p3=65;ret p4
275: if p3#65;gto -3
276: p4→p5;wrt "tpm","T";red "tpm",p1,p2,p3,p4
277: gto -1;if abs(1-p5/p4)<.012;ret p4
```

```
278:
279: "TPMZ":fmt ;wrt "tpm","Z1I" ——————————— Zero-set power meter subroutine
280: fmt b,3x,f4.0;red "tpm",p1,p2
281: if p2>2;jmp -2 ———————————————————— Assure reading is close to zero
282: wrt "tpm","9A+I" ——————————————————— Program to read absolute watts
283: red "tpm",p1,p2
284: gto -2;if p1<84;ret ——————————— Repeat reading until no longer in auto-zero
285:
286: "SGF":cll '8672A FRE'(p1);ret ——————— Calls proper LO subroutine for freq
287:
288: "SGL":cll '8672A LEV'(p1);ret ——————— Calls proper LO subroutine for level
289:
290: "8620C FRE":(p1-2000)*10/16→p2 ——————————— Model 8620C freq subroutine
291: fmt "M1B4V",f4.0,"E";wrt "lo",p2;ret
292:
293: "8672A Device Subroutines": ——————————————— Model 8672A subroutines
294: "8672A FRE":max(p1,2000)→p1;min(p1,18000)→p1 ————————— Freq subroutine
295: fmt "P",fz9.3,"Z9";wrt "lo",p1;ret
296: "8672A LEV":fmt "K",2b,"07",b;if p1>13;13→p1 ————————— Level subroutine
297: if p1<-120;wrt "lo",59,61,48;ret
298: if p1>3;wrt "lo",48,61-p1,51;ret
299: int(abs(p1/10))→p2;wrt "lo",p2+48,51-10p2-p1,49;ret
300:
301: "8660 Device Subroutines":
302: "8660C FRE":fmt f.0,z;wrt "lo",'Inv'(1e6p1,10),"(";ret ——— 8660C freq subroutine
303: "8660A FRE":fmt b,f.0,z;p1/(1+(p1)=1300→p0))→p2 ————— 8660A freq subroutine
304: wrt "lo",73-2p0,'Inv'(1e6p2,10),"(";ret
305: "8660 LEV":fmt f.0,z;wrt "lo",'Inv'(13-p1,3),"C";ret ———— 8660 level subroutine
306: "Inv":p1→p4
307: gto +0;int(p4)/10→p4;10(frc(p4)+p3)→p3;if (p5+1→p5)>=p2;ret p3
308:
309: "8662A Device Subroutines":
310: "8662A FRE":fmt "fr",f12.7,"mz";wrt "lo",p1;ret ——— 8662A fundamental freq subroutine
311: "8662A LEV":fmt "ap",f6.1,"dm";wrt "lo",p1;ret ——————8662A level subroutine
*23195
```

Annotated listing of Mixer Noise Figure and Loss Program.

```
0: "346B/436A AUTOMATIC DSB MIXER NOISE FIGURE & LOSS PROGRAM;800414":
1:
2: "INITIALIZATION":time 5000;cli 7;clr 7;lcl 7
3: otd3137→M;otd2000→P;cll 'FIX BITS'(M,P,S);S→Q ———— Initialize system state
4: if r0;gto "START" ———— Senses whether all variables need to be initialized
5: dev "tpm",713,"lo",719,"ns",707,"att",728,"pl",705
6: psc 705;cfg ;6→r27 ———— If no plotter, change to "psc φ"; device number of printer
7: dim A$[1],C$[72],A[20,6],C[2],G[0:101],T[0:101],P[0:8];20→r11
8: "noise source wait times":50→V;0→U ———— Power supply response times
9: "IF output atten limits":10→r28;30→r29;r28→A ———— Change limits to suit IF amp
10: ldk 1;if r0=0;15.2→r0 ———— Loads special functions from file #1; default ENR
11: ent "FILE # OF ENR DATA?",X;ldf X,C$,A[*] ———— Load ENR data
12: 290→r1;ent "AMBIENT TEMP IN K",r1
13:
14: "START":if flg7;otd7177→M;cll 'FIX BITS'(M,Q,S);cfg 7;pen
15: if bit(4,S);dsp ;cll 'ZERO SET'
16: if bit(1,S);dsp ;cll 'SWEEP'
17: if bit(0,S);dsp ;cll 'CW'
18: if bit(3,S);dsp ;lcl 7;cll 'EXT FREQ'          Branch to subroutines
19: if bit(9,S);cll 'NS ON';lcl 7           ———— according to special
20: otd1000→M;0→P;cll 'FIX BITS'(M,P,S)              function key command
21: if bit(10,S);cll 'NS OFF';lcl 7
22: otd2000→M;0→P;cll 'FIX BITS'(M,P,S)
23: dsp "AWAITING MEASUREMENT COMMAND"
24: gto "START"
25: end
26:
27: "ZERO SET":rem 7;cll 'NS OFF';cll 'ATT'(110);cll 'TPMZ';cll 'ATT'(A)
28: 20→M;0→P;cll 'FIX BITS'(M,P,S);'TPM'→X;ret ———— Zero-set completed
29:
30: "SWEEP":
31: if bit(2,S);cll 'ENTER SSS' ———— Go get new freqs if desired
32: if B=0;cll 'ENTER SSS' ———— If freqs were never entered, go get freqs
33: if bit(5,S);if flg2=0;otd100→M→P;cll 'FIX BITS'(M,P,S) ———— If cal'n is necessary, set bit 6
34: int((C-B)/D)+1→N ———— Calculate number of freqs to be tested
35: gto +2;if bit(6,S);cll 'CAL';gto +1 ———— Go calibrate if needed
36: dsp "CONNECT DEVICE UNDER TEST";lcl 7;stp
37: rem 7;cll 'SGF'(B);13→L;cll 'SGL'(L);cll 'ATT'(A) ———— Initialize signal freq and level
38: cll 'NF' ———— Go measure noise figure at all freqs
39: 2→M;0→P;cll 'FIX BITS'(M,P,S);ret ———— Measurements completed; return to initial state
40:
41: "ENTER SSS":cfg 13;ent "START FREQ(MHz)?",B
42: ent "STOP FREQ(MHz)?",C
43: ent "FREQ STEP(MHz)?",D
44: int((C-B)/D)+1→N;if N>101;gto "ENTER SSS" ———— Assure not too many freqs for array size
45: 4→M;0→P;cll 'FIX BITS'(M,P,S);ret ———— Freq entry completed
46:
47: "CW":if bit(2,S);cll 'ENTER CW' ———— Go get new freq if desired
48: if G=0;cll 'ENTER CW' ———— If CW freq was never entered, go get freq
49: if bit(5,S);if flg3=0;otd100→M→P;cll 'FIX BITS'(M,P,S) ———— If cal'n is necessary set bit 6
50: 0→N
51: gto +2;if bit(6,S);cll 'CAL';gto +1 ———— Go calibrate if needed
52: dsp "CONNECT DEVICE UNDER TEST";lcl 7;stp
53: rem 7;cll 'SGF'(G);13→L;cll 'SGL'(L);cll 'ATT'(A) ———— Initialize signal freq and level
54: cll 'NF';gto "CW" ———— Measure noise figure and repeat CW measurement
55:
56: "ENTER CW":cfg 13;ent "CW FREQ(MHz)",G
57: 4→M;0→P;cll 'FIX BITS'(M,P,S);ret ———— Freq entry completed
58:
59: "EXT FREQ":rem 7
60: if bit(11,S);cll 'NEW EXT ENR' ———— Go get new ENR if desired
61: cll 'TEST IF'(X,Y) ———— Go adjust and measure IF output for source ON and OFF
62: if flg9;fmt f2.0,2f7.2;wrt 16,A,10log(X1e3),10log(Y1e3) ———— If flag 9, print outputs in dBm
63: "hot noise source temp":(tn↑(r0/10)+1)290→r2
64: "Y factor":X/Y→Y;max(Y,1.0033113)→Y ———— Limit max NF to 40 dB (Y to 0.14 dB)
65: "equiv input noise temp":(r2-r1Y)/(Y-1)→T
66: "noise figure":10log(T/290+1)→X
67: "outputs":fmt "NF=",f5.2," ENR=",f4.1;wrt 0,X,r0
68: if bit(7,S);wrt r27,X,r0 ———— If desired, print result
69: if flg7;gto "START" ———— Test for special function key interrupt
70: gto "EXT FREQ" ———— Repeat external frequency measurement
```

```
71:
72: "NEW EXT ENR":fmt "ENR (now=",f5.2,")?";wrt 0,r0;ent "",r0
73: 4000→M;0→P;cll 'FIX BITS'(M,P,S);ret ———————————— ENR entry completed
74:
75: "CAL":dsp "REMOVE DUT (CONNECT NS TO RCVR)";stp
76: "enter IF freq":if r20=0;30→r20 ———————————————— Default value is 30 MHz
77: cfg 13;fmt "IF FREQ (now=",f7.2,")?";wrt 0,r20;ent "",r20
78: cfg 2;cfg 3;rem 7;120→L;cll 'SGL'(L);cll 'ZERO SET';cll 'NS ON'
79: for A=r28 to r29 by 10;cll 'ATT'(A);'TPM'→X→P[A/10] — Step atten and store power readings
80: if A=r28;if X>1e-5;beep;dsp "HI GAIN OR LO ATT! PROG STOPPED!";stp
81: if flg9;fmt f2.0,f10.2;wrt 16,A,10log(X)+30 ——————— If flag 9, print output in dBm
82: next A
83: r28→A;cll 'ATT'(A)
84: r20→F;cll 'FIND ENR'(E) ————————————————————— Go find proper ENR
85: cll 'TEST IF'(X,Y);X-Y→C[1];P[A/10]→P[8] —— Store power difference and atten ref power level
86: "hot noise source temp":(tn↑(E/10)+1)290→r2→r26
87: "Y factor":X/Y→Y;max(Y,1.0033113)→Y —————— Limit max NF to 40 dB (Y to 0.14 dB)
88: "equiv input noise temp":(r2-r1Y)/(Y-1)→C[2]→T
89: fmt f5.0," MHz;IF Amp NF=",f6.2," dB";wrt 0,r20,10log(T/290+1) –Display IF noise figure
90: wait 2000;sfg 2;sfg 3 ———————————————————————— Set calibration valid flags
91: if flg7;gto "START" ——————————————————— Test for special function key interrupt
92: otd100→M;0→P;cll 'FIX BITS'(M,P,S);cll 'FIX BITS'(M,P,Q);ret ——— Cal completed
93:
94: "NF":for I=bit(1,S) to N;'FREQ'(I)→F;cll 'SGF'(F) —————————— Set freq
95: cll 'TEST IF'(X,Y) ——————————— Go adjust and measure IF output for source ON and OFF
96: if flg9;fmt f2.0,2f7.2;wrt 16,A,10log(X1e3),10log(Y1e3) —— If flag 9, print output in dBm
97: X-Y→G[I];cll 'FIND ENR'(E) ———————————— Store power difference; go find proper ENR
98: "hot noise source temp":(tn↑(E/10)+1)290→r2
99: "Y factor":X/Y→Y;max(Y,1.0033113)→Y —————— Limit max NF to 40 dB (Y to 0.14 dB)
100: "equiv input noise temp":(r2-r1Y)/(Y-1)→T→T[I]
101: cll 'OUTPUTS' ——————————————————————————— Output results
102: if flg7;gto "START" ——————————————————— Test for special function key interrupt
103: next I ————————————————————————————————— Repeat at next freq
104: ret ——————————————————————————————— Measurements completed
105:
106: "FIND ENR":
107: max(J,1)→J;if F<A[J,1];J-1→J;gto +0;if J=0;A[1,2]→p1;ret
108: if J=r11;A[J,2]→p1;ret
109: if F>A[J+1,1];J+1→J;gto -2
110: "interpolate":(F-A[J,1])(A[J+1,2]-A[J,2])/(A[J+1,1]-A[J,1])+A[J,2]→p1
111: ret
112:
113: "-----------------------------------------------------------------------------------";
114:
115: "UTILITY SUBROUTINES":
116: "FREQ":if bit(0,S);G→p2;ret p2
117: B+(p1-1)D→p2;ret p2
118:
119: "FIX BITS":ior(band(cmpp1,p3),band(p1,p2))→p3;ret
120:
121: "-----------------------------------------------------------------------------------";
122:
123: "OUTPUTS":
124: "LED output":
125: "noise fig":10log(T/290+1)→X
126: if bit(5,S);gto "LED NF&G" ————————————— Different output display for noise figure and loss
127: fmt f5.0," MHz   NF=",f6.2," dB"
128: wrt 0,F,X;gto "Printer output" ———————— Display freq and NF; go see about printing result
129: "LED NF&G":sfg 14;G[I]P[8](r26-r1)/C[1]/P[A/10](r2-r1)→Y
130: 10log(C[2]/290+1)→W
131: 10log((T-C[2]/Y)/290+1)+3→Z;10log(Y)-3→Y→G[I];cfg 14
132: fmt f5.0,"MHz;NF=",f5.2,"dB;L=",f5.2,"dB"
133: wrt 0,F,Z,-Y ——————————————————————————— Display freq, noise figure, and loss
134:
```

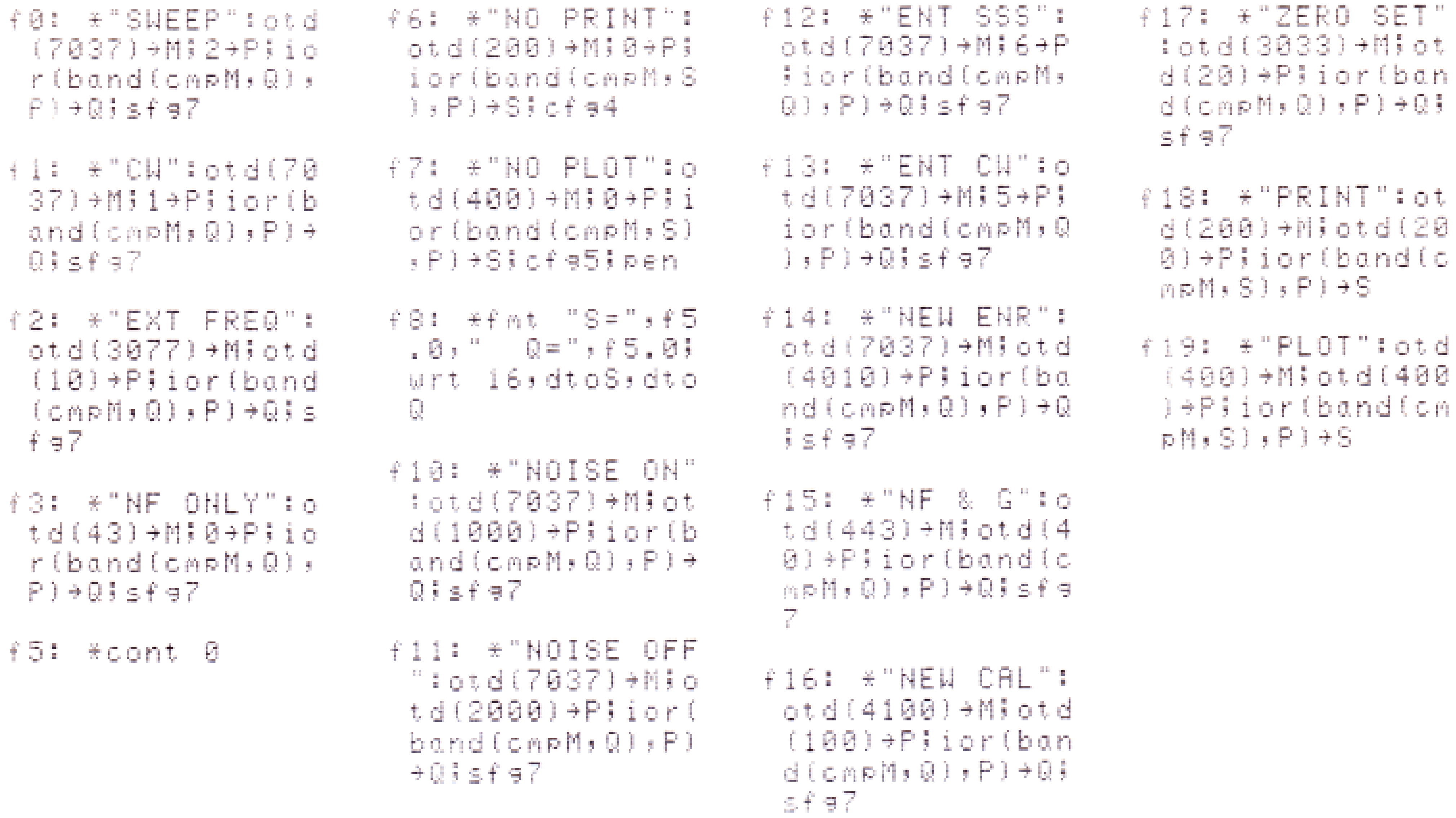


```
135: "Printer output":if bit(7,S)=0;gto "Plotter output" — If no printing, go see about plotting
136: if I=1;cfg 4 ——————————————— At first freq get ready to print heading
137: if bit(5,S);gto "NF&G Printout" ——————— Different printout for noise figure and loss
138: if flg4;gto "Print NF" ——————————— If heading is already printed, skip to print result
139: fmt "Freq(MHz)   NF(dB)",/;wrt r27;sfg 4
140: "Print NF":fmt f7.0,f10.2;wrt r27,F,X ———— Print freq and noise figure
141: gto "Check end of print"
142: "NF&G Printout":if flg4;gto "Print NF&G" ——— If heading is already printed, print results
143: fmt "Freq(MHz)  NF(dB)   L(dB)  Tot NF  rcvrNF totNF-rcvrNF+3dB",/ — Print heading
144: wrt r27;sfg 4
145: "Print NF&G":fmt f5.0,4x,8f8.2;wrt r27,F,Z,-Y,X,W,X-W+3 ——— Print results
146: "Check end of print":if I#N;gto "Plotter output" }
147: if I#0;fmt 3/;wrt r27                            }——— Leave blank lines at last freq
148:
149: "Plotter output":if bit(1,S)=0;ret ——————————— No plots unless sweeping
150: if bit(8,S)=0;ret ——————————————————— If no plot is wanted, return
151: if I=1;cll 'CHANGE PLOT PARAMS?'  }
152: if flg5=0;cll 'CHANGE PLOT PARAMS?' }——— At first freq check about plotting axes
153: scl r3-r22r25,r4+r22r25,r7-r22r23,r8+2r23 }
154: lim r3,r4,r7,r8+2r23                     }——— Scale for noise figure vs. freq
155: pen# 4;line ;if bit(5,S);gto "Plot corrected NF" ——— Plot solid line
156: plt F/1000,X ——————————————————— Plot measured noise figure vs. freq
157: if I=N;pen ——————————————————————— At last freq raise pen
158: ret ——————————————————————————— Plot completed
159:
160: "Plot corrected NF":plt F/1000,Z ——————— Plot corrected noise figure vs. freq
161: if I#N;ret ———————— Return unless this is the last freq (at last freq maybe more plots)
162: if flg7;gto "START" ——————————— Test for special function key interrupt
163:
164: pen;cfg 13;ent "UNCORRECTED NF PLOT (Y or N)?",A$
165: if flg13;"N"→A$ ————————————————————— Default answer is "N"
166: if cap(A$)="N";gto "Gain Plot?"
167: if cap(A$)#"Y";gto -3
168: pen# 4;line 2,1 ————————————————————— Plot with dashed line
169: for K=1 to N;'FREQ'(K)→F ——————————————————— Find freq
170: plt F/1000,10log(T[K]/290+1);next K ——————— Plot measured noise figure vs. freq
171: pen;line —————————————————————————— Next plots with solid line
172: if flg7;gto "START" ——————————————— Test for special function key interrupt
173:
174: "Gain Plot?":cfg 13;ent "PLOT LOSS (Y or N)?",A$
175: if flg13;"Y"→A$ ————————————————————— Default answer is "Y"
176: if cap(A$)="N";ret
177: if cap(A$)#"Y";gto -3
178: "Plot Gain":if flg6=0;cll 'Gain axis' ——————— If there is no loss axis, go plot one
179: scl r3-r22r25,r4+r22r25,r16-r22r24,r17+2r24 }
180: pen# 3;lim r3,r4,r16,r17+2r24              }——— Scale for loss vs. freq
181: for K=1 to N;'FREQ'(K)→F ——————————————————— Find freq
182: plt F/1000,-G[K];next K ————————————————— Plot loss vs. freq
183: pen;ret ————————————————————————— Plotting completed
184:
185: "CHANGE PLOT PARAMS?":if r21=0;gto "Default freq axis"
186: cfg 13;ent "NEW PAPER (Y or N)?",A$
187: if flg13;"N"→A$ ————————————————————— Default answer is "N"
188: if cap(A$)="N";sfg 5;ret
189: if cap(A$)#"Y";gto -3
190: cfg 5;cfg 6;gto "Old axes OK?" ——————— Clear axes OK flag; go check on desired scales
191:
192: "Default freq axis":int(B/1000)→r3
193: int(C/1000)→r4;if C/1000#r4;1+r4→r4
194: .5→r5;if r4-r3<7;.2→r5
195: 1/r5→r6
196: "Default NF axis":3→r7;13→r8;.2→r9;5→r10
197: "Default gain axis":3→r16;13→r17;.2→r18;5→r19;gto "Check each axis"
198:
199: "Old axes OK?":cfg 13;ent "ARE OLD AXES OK (Y or N)?",A$
200: if flg13;"Y"→A$ ————————————————————— Default answer is "Y"
201: if cap(A$[1])="Y";gto "Label axes"
202: if cap(A$[1])#"N";gto -3
203:
```

```
204: "Check each axis":fmt "Min freq in GHz (now=",f5.2,")?"
205: wrt 0,r3;ent "",r3
206: fmt "Max freq in GHz (now=",f5.2,")?";wrt 0,r4;ent "",r4
207: fmt "Tic interval in GHz (now=",f5.2,")?";wrt 0,r5;ent "",r5
208: fmt "#tics/label (now=",f2.0,")?";wrt 0,r6;ent "",r6
209: fmt "Min Noise Figure in dB (now=",f2.0,")?";wrt 0,r7;ent "",r7
210: fmt "Max Noise Figure in dB (now=",f2.0,")?";wrt 0,r8;ent "",r8
211: fmt "Tic interval in dB (now=",f5.2,")?";wrt 0,r9;ent "",r9
212: fmt "#tics/label (now=",f5.2,")?";wrt 0,r10;ent "",r10
213: fmt "Min Loss in dB (now=",f2.0,")?";wrt 0,r16;ent "",r16
214: fmt "Max Loss in dB (now=",f2.0,")?";wrt 0,r17;ent "",r17
215: fmt "Tic interval in dB (now=",f2.0,")?";wrt 0,r18;ent "",r18
216: fmt "#tics/label (now=",f5.2,")?";wrt 0,r19;ent "",r19
217:
218: "Label axes":pclr;fmt ;wrt "pl","OP";red "pl",r12,r13,r14,r15 — Find corners of plot
219: "# of character heights per margin":5→r22;1.5→r21
220: "Character height in NF units":(r8-r7)/(100/r21-r22)→r23
221: "Character height in Gain units":(r17-r16)/(100/r21-r22)→r24
222: "Character height in Freq units":100(r14-r12)/r21(r15-r13)-2r22→r25
223: (r4-r3)/r25→r25
224:
225: "Freq axis":scl r12,r14,r13,r15;lim r12,r14,r13,r15
226: csiz r21,2,(r15-r13)/(r14-r12),0
227: pen# 2;plt (r14+r12)/2,r13,1;cplt -8,.5;lbl "Frequency (GHz)"
228: scl r3-r22r25,r4+r22r25,r7-r22r23,r8+2r23
229: xax r7,r5,r3,r4,r6
230:
231: "NF axis":scl r12,r14,r13,r15;csiz r21,2,(r15-r13)/(r14-r12),90
232: pen# 4;plt r12,(r15+r13)/2,1;cplt -9,-1.1;lbl "Noise Figure (dB)"
233: scl r3-r22r25,r4+r22r25,r7-r22r23,r8+2r23
234: yax r3,r9,r7,r8,r10;sfg 5;ret
235:
236: "Gain axis":scl r12,r14,r13,r15;lim r12,r14,r13,r15
237: csiz r21,2,(r15-r13)/(r14-r12),90
238: pen# 3;plt r14,(r15+r13)/2,1;cplt -5,.5;lbl "Loss (dB)"
239: scl r3-r22r25,r4+r22r25,r16-r22r24,r17+2r24
240: yax r4,r18,r16,r17
241: yax r4+3r25,r18,r16,r17,-r19;sfg 6;ret
242:
243: "-----------------------------------------------------------------------";
244:
245: "INSTRUMENT SUBROUTINES":
246: "TEST IF":cll 'NS ON';'TPM'→p1 ——————————— Go measure ON noise power
247: if p1<1e-5;jmp 3 ——————————— OK if pl < -20 dBm so go measure OFF noise power
248: if A=r29;beep;dsp "I-F SATURATED! PROGRAM STOPPED";lcl 7;end ——————— Error
249: A+10→A;cll 'ATT'(A);gto "TEST IF" ——————— Raise atten and repeat measurement
250: cll 'NS OFF';'TPM'→p2 ——————————— Measure OFF noise power
251: if p2>1.258e-8;ret ——————— OK if > -49 dBm so return to calling subroutine
252: if A=r28;jmp 3 ——————— Do not lower atten beyond min allowed value
253: A+10int((10log(p1*1e3)+22)/10)→A       }
254: max(r28,A)→A;cll 'ATT'(A);gto "TEST IF" }——— Lower atten and repeat measurements
255: if p2<5e-10;beep;dsp "LOW GAIN! PROGRAM STOPPED";lcl 7;end ——————— Error
256: fmt "SYSTEM GAIN MARGINAL AT",f5.0,"MHz" ——————————— Warning
257: wrt 0,F;beep;wait 1000 ——————————— Display freq of marginal gain
258: ret ——————————— Measurement complete
259:
260: "ATT":int(p1)→p1 ——————————— General subroutine for Model 11713A
261: min(p1,121)→p1;max(0,p1)→p1
262: int(p1/10)→p2;if p2=12;11→p2
263: fxd 0;cll 'swt#'(p2,1) ——————————— 10 dB step atten connected to "X" output
264: p1-10p2→p3
265: cll 'swt#'(p3,5);wait 20;ret
266:
267: "swt#":p1→p3;if p1>7;p1+4(p3>3)→p3
268: if bit(p6,p3);p2+p6+10p5→p5;jmp 2
269: p2+p6+10p4→p4
270: p6+1→p6;if p6<4;jmp -2
271: if p4;if p5;fmt "A",f4.0,"B",f4.0;wrt "att",p5,p4;ret
272: fmt "A",f4.0;if p5;wrt "att",p5;ret
273: fmt "B",f4.0;wrt "att",p4;ret
```

```
274:
275: "NS ON":rem 7;fmt c,z;wrt "ns","2560";wait U;ret ——— Subroutine for Model 6002A-001
276:
277: "NS OFF":rem 7;fmt c,z;wrt "ns","2000";wait V;ret ——— Subroutine for Model 6002A-001
278:
279: "TPM":fmt ;wrt "tpm","9A+T";fmt 3b,e9.0;red "tpm",p1,p2,p3,p4 ——— Read power
280: fmt ;wrt "tpm","9A+T";fmt 3b,e9.0;red "tpm",p1,p2,p3,p4               subroutine
281: if p2>73;if p3=65;ret p4
282: if p3#65;gto -3
283: p4→p5;wrt "tpm","T";red "tpm",p1,p2,p3,p4
284: gto -1;if abs(1-p5/p4)<.012;ret p4
285:
286: "TPMZ":fmt ;wrt "tpm","Z1I" ——— Zero-set power meter subroutine
287: fmt b,3x,f4.0;red "tpm",p1,p2
288: if p2>2;jmp -2 ——— Assure reading is close to zero
289: wrt "tpm","9A+I" ——— Program to read absolute watts
290: red "tpm",p1,p2
291: gto -2;if p1<84;ret ——— Repeat reading until no longer in auto-zero
292:
293: "SGF":cll '8672A FRE'(p1);ret ——— Calls proper LO subroutine for freq
294:
295: "SGL":cll '8672A LEV'(p1);ret ——— Calls proper LO subroutine for level
296:
297: "8620C FRE":(p1-2000)*10/16→p2 ——— Model 8620C freq subroutine
298: fmt "M1B4V",f4.0,"E";wrt "lo",p2;ret
299:
300: "8672A Device Subroutines": ——— Model 8672A subroutines
301: "8672A FRE":max(p1,2000)→p1;min(p1,18000)→p1 ——— Frequency subroutine
302: fmt "P",fz9.3,"Z9";wrt "lo",p1;ret
303: "8672A LEV":fmt "K",2b,"07",b;if p1>13;13→p1 ——— Level subroutine
304: if p1<-120;wrt "lo",59,61,48;ret
305: if p1>3;wrt "lo",48,61-p1,51;ret
306: int(abs(p1/10))→p2;wrt "lo",p2+48,51-10p2-p1,49;ret
307:
308: "8660 Device Subroutines":
309: "8660C FRE":fmt f.0,z;wrt "lo",'Inv'(1e6p1,10),"(";ret ——— 8660C freq subroutine
310: "8660A FRE":fmt b,f.0,z;p1/(1+(p1>=1300→p0))→p2 ——— 8660A freq subroutine
311: wrt "lo",73-2p0,'Inv'(1e6p2,10),"(";ret
312: "8660 LEV":fmt f.0,z;wrt "lo",'Inv'(13-p1,3),"C";ret ——— 8660 level subroutine
313: "Inv":p1→p4
314: gto +0;int(p4)/10→p4;10(frc(p4)+p3)→p3;if (p5+1→p5)>=p2;ret p3
315:
316: "8662A Device Subroutines":
317: "8662A FRE":fmt "fr",f12.7,"mz";wrt "lo",p1;ret ——— 8662A fundamental freq subroutine
318: "8662A LEV":fmt "ap",f6.1,"dm";wrt "lo",p1;ret ——— 8662A level subroutine
*26079
```

Listing of special function keys for noise figure programs.



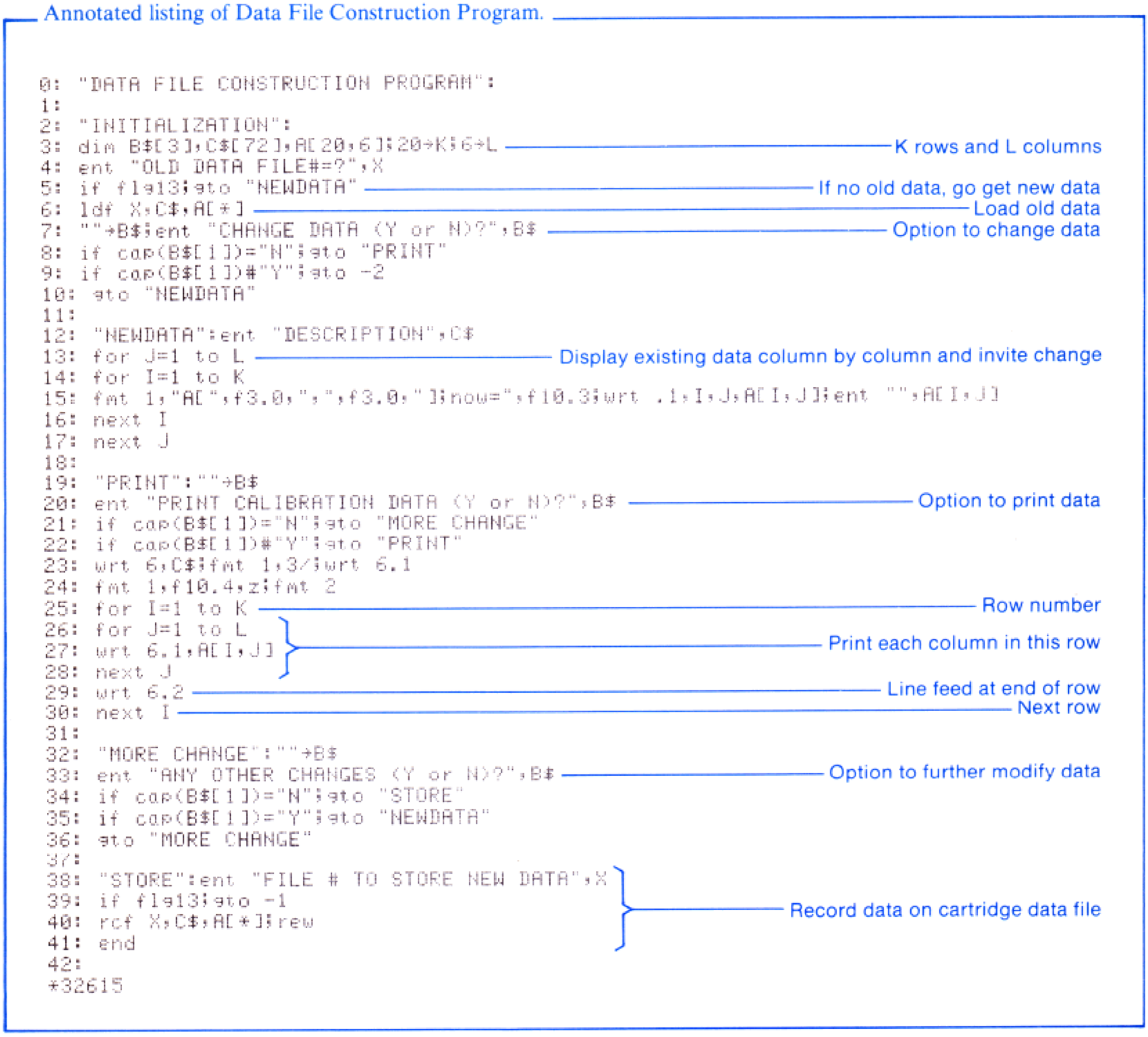
```
f0: *"SWEEP":otd
 (7037)→M;2→P;io
 r(band(cmpM,Q),
 P)→Q;sfg7

f1: *"CW":otd(70
 37)→M;1→P;ior(b
 and(cmpM,Q),P)→
 Q;sfg7

f2: *"EXT FREQ":
 otd(3077)→M;otd
 (10)→P;ior(band
 (cmpM,Q),P)→Q;s
 fg7

f3: *"NF ONLY":o
 td(43)→M;0→P;io
 r(band(cmpM,Q),
 P)→Q;sfg7

f5: *cont 0
```

```
f6: *"NO PRINT":
 otd(200)→M;0→P;
 ior(band(cmpM,S
 ),P)→S;cfg4

f7: *"NO PLOT":o
 td(400)→M;0→P;i
 or(band(cmpM,S)
 ,P)→S;cfg5;pen

f8: *fmt "S=",f5
 .0,"  Q=",f5.0;
 wrt 16,dtoS,dto
 Q

f10: *"NOISE ON"
 :otd(7037)→M;ot
 d(1000)→P;ior(b
 and(cmpM,Q),P)→
 Q;sfg7

f11: *"NOISE OFF
 ":otd(7037)→M;o
 td(2000)→P;ior(
 band(cmpM,Q),P)
 →Q;sfg7
```

```
f12: *"ENT SSS":
 otd(7037)→M;6→P
 ;ior(band(cmpM,
 Q),P)→Q;sfg7

f13: *"ENT CW":o
 td(7037)→M;5→P;
 ior(band(cmpM,Q
 ),P)→Q;sfg7

f14: *"NEW ENR":
 otd(7037)→M;otd
 (4010)→P;ior(ba
 nd(cmpM,Q),P)→Q
 ;sfg7

f15: *"NF & G":o
 td(443)→M;otd(4
 0)→P;ior(band(c
 mpM,Q),P)→Q;sfg
 7

f16: *"NEW CAL":
 otd(4100)→M;otd
 (100)→P;ior(ban
 d(cmpM,Q),P)→Q;
 sfg7
```

```
f17: *"ZERO SET"
 :otd(3033)→M;ot
 d(20)→P;ior(ban
 d(cmpM,Q),P)→Q;
 sfg7

f18: *"PRINT":ot
 d(200)→M;otd(20
 0)→P;ior(band(c
 mpM,S),P)→S

f19: *"PLOT":otd
 (400)→M;otd(400
 )→P;ior(band(cm
 pM,S),P)→S
```

Annotated listing of Data File Construction Program.



```
0: "DATA FILE CONSTRUCTION PROGRAM":
1:
2: "INITIALIZATION":
3: dim B$[3],C$[72],A[20,6];20→K;6→L
4: ent "OLD DATA FILE#=?",X
5: if flg13;gto "NEWDATA"
6: ldf X,C$,A[*]
7: ""→B$;ent "CHANGE DATA (Y or N)?",B$
8: if cap(B$[1])="N";gto "PRINT"
9: if cap(B$[1])#"Y";gto -2
10: gto "NEWDATA"
11:
12: "NEWDATA":ent "DESCRIPTION",C$
13: for J=1 to L
14: for I=1 to K
15: fmt 1,"A[",f3.0,",",f3.0,"]now=",f10.3;wrt .1,I,J,A[I,J];ent "",A[I,J]
16: next I
17: next J
18:
19: "PRINT":""→B$
20: ent "PRINT CALIBRATION DATA (Y or N)?",B$
21: if cap(B$[1])="N";gto "MORE CHANGE"
22: if cap(B$[1])#"Y";gto "PRINT"
23: wrt 6,C$;fmt 1,3/;wrt 6.1
24: fmt 1,f10.4,z;fmt 2
25: for I=1 to K
26: for J=1 to L
27: wrt 6.1,A[I,J]
28: next J
29: wrt 6.2
30: next I
31:
32: "MORE CHANGE":""→B$
33: ent "ANY OTHER CHANGES (Y or N)?",B$
34: if cap(B$[1])="N";gto "STORE"
35: if cap(B$[1])="Y";gto "NEWDATA"
36: gto "MORE CHANGE"
37:
38: "STORE":ent "FILE # TO STORE NEW DATA",X
39: if flg13;gto -1
40: rcf X,C$,A[*];rew
41: end
42:
*32615
```

K rows and L columns

If no old data, go get new data

Load old data

Option to change data

Display existing data column by column and invite change

Option to print data

Row number

Print each column in this row

Line feed at end of row

Next row

Option to further modify data

Record data on cartridge data file

HEWLETT
PACKARD

5952-8218

Printed in U.S.A.