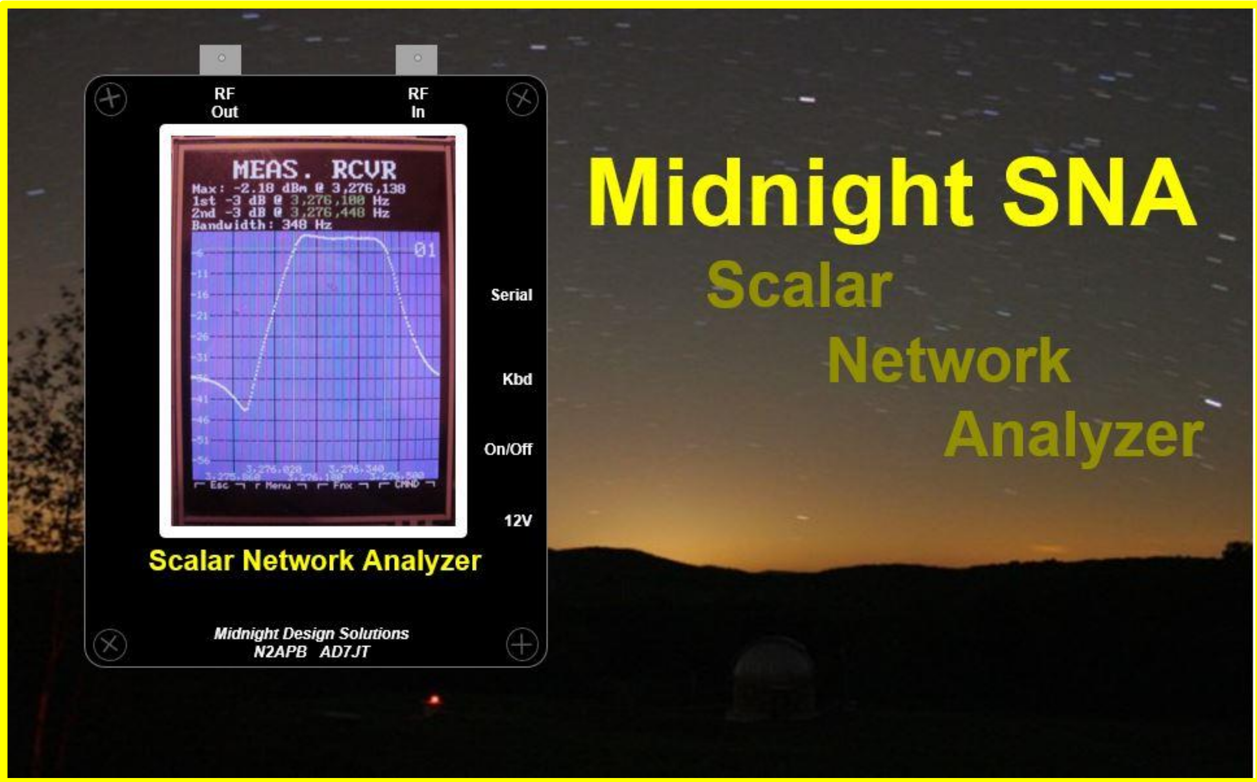
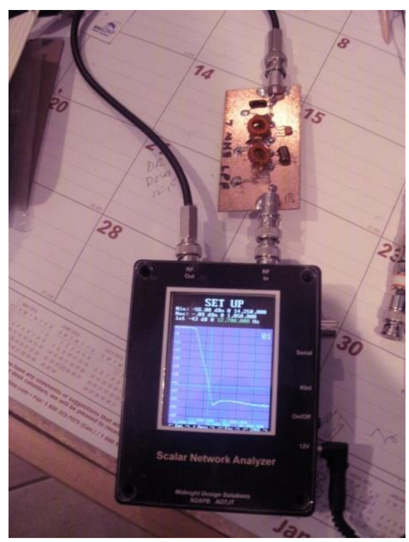


Midnight Scalar Network Analyzer (MSNA)

By Dave Collins, AD7JT and George Heron, N2APB

Version 4 of the MSNA firmware completes the evolution of the Network Analyzer Terminal (NAT) to a full-featured, self-contained, hand-held Scalar Network Analyzer. This paper briefly describes the MSNA development project and many applications for this tool in and around your ham shack and in the field. The resulting SNA can be purchased fully assembled, can be assembled from scratch using a kit from Midnight Design Solutions[1], or can be assembled incrementally by applying simple-to-do hardware enhancements to a basic NAT. The process is further simplified by the ability to field upgrade the NAT-SNA firmware from a file on the SD card mass storage device included in the basic design.



Midnight Scalar Network Analyzer testing a prototype 40-Meter Low Pass Filter (LPF).

INTRODUCTION

The MSNA is based on what we call "QVGA16". QVGA16 is a (mostly) C language library written for 16-bit Microchip PIC and dsPIC micro controllers. The library contains a set of drivers, support functions, and utility routines developed to support the functionality of a display assembly largely designated "TFT_320QVT" and available from many suppliers including several on eBay for less than $20, quantity one.

The firmware for this project adds the SNA functions on top of the QVGA16 foundation plus drivers for a serial interface, a PS2 keyboard, and a DDS-60 digitally controlled RF source. In addition to the microcontroller and a couple support ICs, the hardware also includes a logarithmic power meter and a precision signal source for calibration.

GENERAL SPECS and FEATURES

Physical specifications:

- PCB: 4.47"x3.31"
- Enclosure: 4.82"x3.77"x1.39"
- Power: 8-12V DC @ 330ma (typ)
- Weight: 8 oz (approx)

Features:

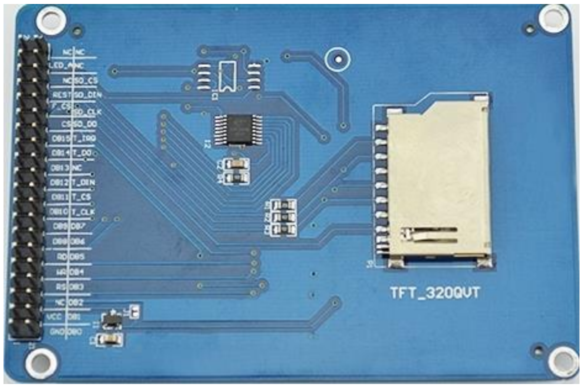
- Handheld graphic terminal
- 3.2", 240x320, 64K color graphic display
- User-friendly operator interface
- Keyboard and touch screen for human input
- Field upgradable firmware
- Serial port with data rates of 1,200 to 115,200 baud, user selectable
- Basic ASCII terminal emulation
- 32KB EEPROM for persistent storage (settings, options, and macros)
- 14 macros for storing operating parameters
- Simplified calibration (no curve fitting)
- Measurement & Plot Capabilities
    - Testing and evaluating filters
    - Sorting and matching crystals
    - Antenna return loss and VSWR
    - Insertion loss measurements
    - QRP power meter (current, average, and peak power)
- Operating frequency range from <1 to 60+ MHz
- Wide dynamic range: over 95 dB
- SD Card mass storage up to 1GB
    - FAT16 file system compatibility
    - Subdirectory support
    - Data spooling and playback
    - Storage format compatible with Windows and Linux applications
    - File transfers to PCs using serial interface
    - Easy firmware upgrades
    - DOS-like commands to manage and playback data files

Detailed descriptions of the MSNA hardware design including schematics and extensive user documentation can be found on the NAT-SNA and MSNA web pages[2]. This paper primarily concentrates on describing the MSNA functions and applications. However, a brief look at the basic functions provided by the display assembly will help clarify later description of MSNA operations.

DISPLAY ASSEMBLY

The heart of the display assembly is, of course, the 3.2" (diagonal) TFT LCD display. It is a quarter VGA (QVGA) display which means it has a 240x320 pixel array. It supports 16-bit color (5R-6G-5B) and is covered with a slightly oversized, 4-wire, resistive touch screen. The oversize allows for a row of buttons below the display area.

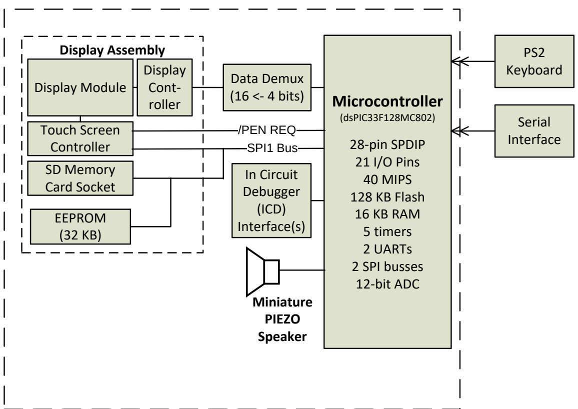
In addition to the LCD display, the display assembly has, on the back side of the PCBA, a touch screen controller, an SD card socket, and the pads for mounting an EEPROM. The display controller is integral to the LCD display and presents a 16-bit parallel data interface. A Small Peripheral Interface (SPI) bus is used to access the touch screen controller, SD card, and EEPROM.



All display module device interfaces are presented on a single 40-pin (2x20), male connector.

NAT BASICS

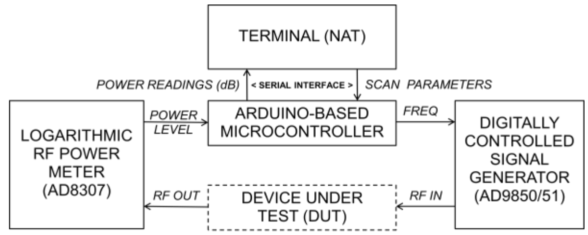
The basic NAT architecture is shown in the following block diagram:



We wanted a pluggable microcontroller so we selected one in a 28-pin DIP package. This gave us only 21 I/O pins to access all the peripheral devices. To relieve some of the pressure, a couple ICs were added to de-multiplex the display data interface. The demultiplexor enables the microcontroller to access the 16-bit parallel data interface plus one strobe using only seven I/O pins on the microcontroller, a savings of 10 I/O pins. To further reduce the I/O pin count requirement, three of these seven I/O pins double as chip select signals for devices on the SPI bus.

POOR HAM'S SNA

Our initial objective was to replace the PC in the Poor Ham's Scalar Network Analyzer (PHSNA)[3]. The following block diagram shows the basic PHSNA architecture with the NAT replacing the usual PC.



In this architecture, the terminal specifies the SNA operation to be performed and provides any required operating parameters. The SNA functions are performed by the PHSNA controller and the results are returned to the terminal. The operating parameters generally include the starting frequency, ending frequency, and the frequency change-per-step.

Once the SNA operation starts, the PHSNA controller sets the DDS frequency to the specified starting value, reads the power meter output, and converts the reading to dBm. The power level is sent to the terminal and the DDS frequency is incremented by the specified change-per-step value and the new power level is sent to the terminal. This process continues until the DDS frequency has been advanced beyond the specified ending frequency.

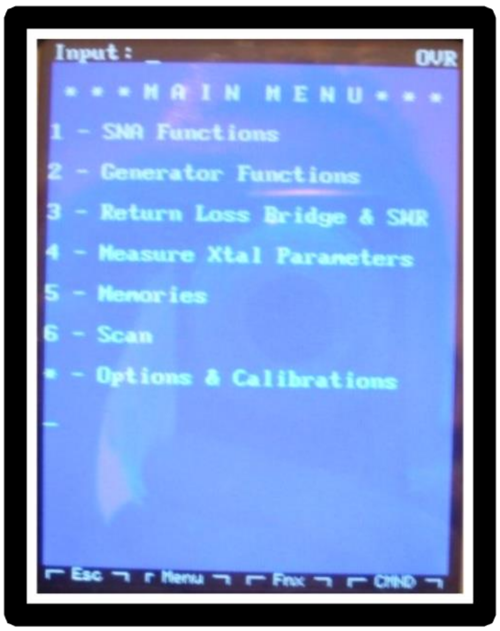
The PHSNA controller operates in one of two modes as described below.

MENU MODE

In Menu mode the PHSNA controller generates menus and prompts for display on the terminal. You select menu items and enter responses to prompts. When the terminal is a PC, the PC is normally running a terminal emulator application. During the frequency scan, results are returned to the PC and displayed on the PC monitor. The results are also stored in a buffer allowing you to scroll through them. You can then copy data from the buffer and paste them in another PC application such as Excel for plotting and further analysis.

When the terminal is the NAT the NAT is operating in Terminal mode which emulates the PC terminal emulator application. It is important to note that the PHSNA controller is generating all menus and prompts, the NAT mostly operates as a dumb terminal. The NAT will work with the standard PHSNA controller firmware but that version assumes the terminal is an 80-column window on the PC. Since the NAT's display is only 30 columns wide, the standard PHSNA menus and prompts are a little hard to read and follow. For this reason, the standard PHSNA firmware has been modified to look better on the NAT display and, in some cases, to accommodate touch screen operations. This version is functionally equivalent to the standard version.

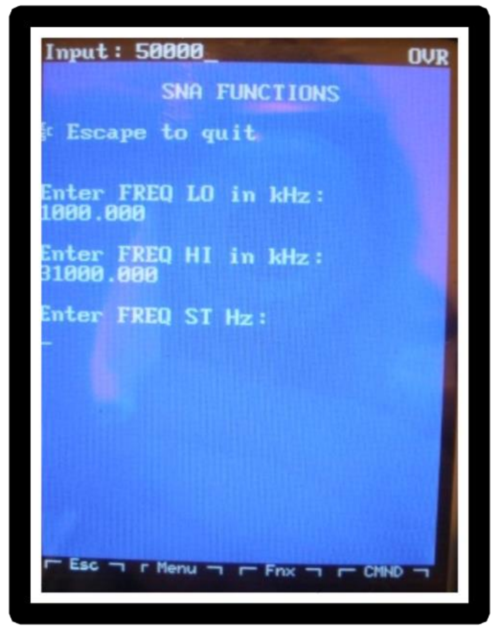
When the PHSNA controller is reset it displays the main menu which looks like this on the NAT display:



In Terminal mode, the NAT displays all received text in the blue area of the display. All menus are double spaced to simplify touch screen operation. Touching a blank line will select the line above it. The top line is the input line. Text entered here will be sent to the PHSNA controller when Enter is pressed. Entries may be edited using the arrow, backspace, and delete keys before entering Enter. Editing may operate in either overprint or insert mode as indicated in the upper right corner of the display.

The following example demonstrates Menu mode operation using the PHSNA to scan and the NAT to plot the frequency response of a 40 meter, Low Pass Filter (LPF) which is the Device Under Test (DUT).

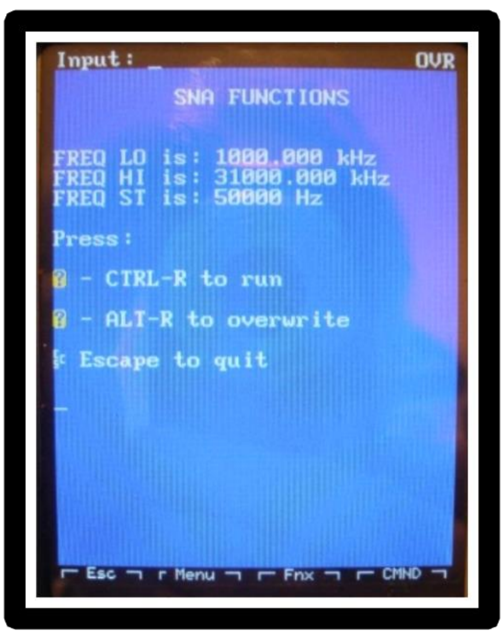
For this example we will use the basic SNA Functions (menu item 1). Other options support special test fixtures and operations. Documentation is available online[2] describing the PHSNA menus so they will not be covered here. We select menu option 1 by either keying '1' followed by Enter or by momentarily touching the menu line or the one below it. After the selection is made, the PHSNA firmware generates a number of prompts directing you to enter the operating parameters for the frequency sweep.



Here we have already entered the starting frequency (FREQ LO: 1 MHZ) and the ending frequency (FREQ HI: 31 MHz) and are in the process of entering the frequency change-per-step (FREQ ST: 50000 Hz). Note that there is a numeric pad (NUM PAD) display we can call up to enter numeric values using the touch screen. The NUM PAD is called up by touching the input line.

The NUM PAD will close when Enter (on the NUM PAD) or Escape (Esc on the permanent buttons along the bottom of the display) is pressed or touched.

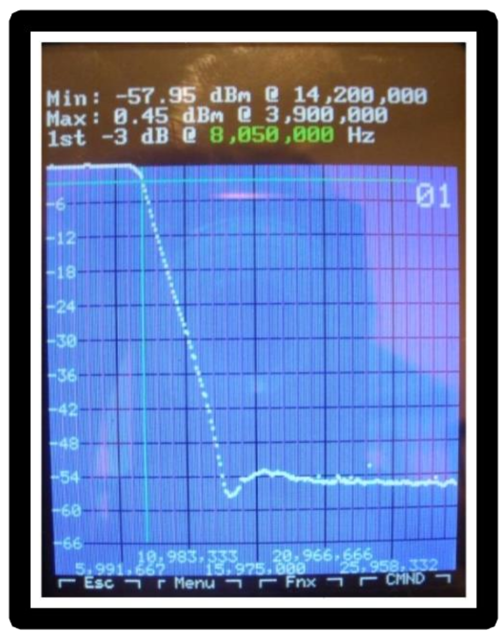
After the FREQ ST has been entered, the PHSNA firmware displays a summary of the operation and gives us three choices.



Touching or entering Escape will abort the operation and return to the main menu. The other two options have to do with data logging. The NAT allows us to specify a file on the SD card to spool the results to. This file is referred to as the "log file" and is shared by several functions. We can enter and/or edit the log file name using a NAT command.

When a log file has been named, the NAT firmware will attempt to open it in write mode. When the operation is started with CTRL-R and the named log file does not exist, a new file is created and the returned results spooled to it. In this case, if the file already exists, a warning is displayed and the results are not spooled but the scan is performed without spooling. When the operation is started with ALT-R and the named log file does exist, it is deleted and a new file is created.

During the frequency scan, the power level for each step is returned to and recorded by the NAT firmware. During the transfer, data is shown on the NAT display but scrolling is turned off because it is both time consuming and distracting to have the screen scrolling through what could be several hundred rows of data. With scrolling turned off, the data line is continuously overwritten which serves as a progress display. After the data transfer is complete, the PHSNA sends a special directive and the NAT firmware displays some basic information about the data set, sets up scaling factors, and plots the data.



Here we see the frequency response for our 40M LPF over a frequency range of 1 MHz to 31 MHz. At the top of the display we see some information about the overall data set. The third line shows the frequency at which the "roll-off power level" (-3 dB in this case) first occurs. The -3 dB point is often used to characterize filters. Some filters use -6 dB and sometimes we are interested in other points along the curve. For these reasons, the roll-off power level can be set by the operator.

The NAT firmware recognizes five curve types as follows:

- Low Pass Filter (LPF)
- High Pass Filter (HPF)
- Band Pass Filter (BPF)
- Notch Filter (NF)
- None of the above

The LPF and HPF plots will display only one roll-off power level, the BPF and NF plots will display two. These points are shown on the plot by the horizontal green line and one or two vertical green lines.

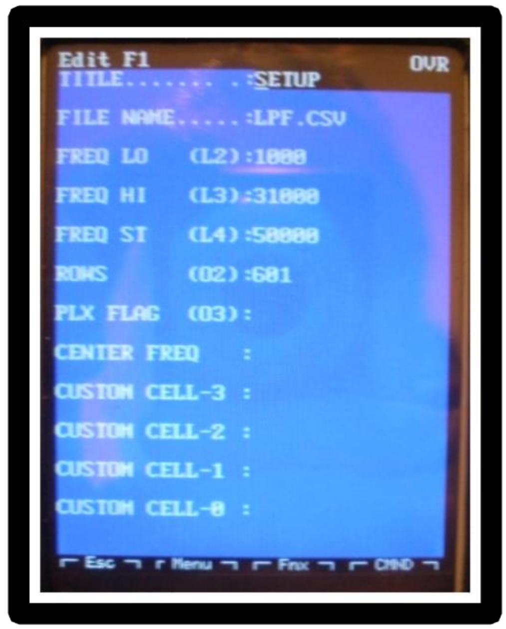
PLX MODE

The second PHSNA operating mode is called "PLX Mode". In this mode a PC terminal would be running Excel with a special VB macro installed. This macro allows external devices to access spreadsheet cells over a PC port. The macro was developed by Parallax Corporation and is named "PLX-DAQ". The macro is freeware and is available for down loading from the Parallax web site[4].

The PHSNA designers defined a standard spreadsheet with specific locations assigned to the SNA operating parameters and result buffer. After a reset, the PHSNA firmware attempts to access one of these parameter cells and, if successful, changes to PLX mode otherwise it remains in Menu mode. In PLX mode the PHSNA firmware collects the scan parameters (FREQ LO, FREQ HI, and FREQ ST) and initiates the scan. The results are sent back using a special directive causing them to be loaded in the first few columns of the spreadsheet starting at the first row.

When the terminal is the NAT the NAT firmware is operating in PLX mode. In PLX mode the NAT emulates the standard spreadsheet with the PLX DAQ macro installed. The NAT does not actually have a spreadsheet to work with. Instead, the NAT uses forms (macros) with certain entries assigned to cells in the standard spreadsheet. The NAT firmware receives commands (directives) from the PHSNA firmware and responds as the PLX DAQ would respond from the standard spreadsheet.

Macros are accessed using function keys F1 through F7 and ALT-F1 through ALT-F7 giving a total of fourteen possible macros. Macros are stored in the EEPROM so they will persist through a power cycle. We edit macros using the following form:



In this case we are editing the macro assigned to F1 as indicated by the top line of the display. The currently active field is highlighted with a black background. The up and down arrow keys move from one field to the next. We can also tap the line we wish to edit to select it. The NUM PAD is available for numeric entry.

The first line of the macro is the title. Titles are used to identify macros and are displayed when the macros are listed. The title is also used to identify the macro on a plot display.

The second line specifies the name of the spool file where the results will be saved. Each macro can specify a different file name. If no spool file name is specified, the results will be plotted but not spooled.

The next three lines define the starting and ending scan frequencies and the frequency change-per-step. The designators in parentheses are the spreadsheet coordinates of the corresponding cells in the standard spreadsheet. We can edit these coordinates if a different spreadsheet layout is used.

The fourth line is the numbers of rows in the data set. Each row generally contains the frequency and the power level for that step plus the raw ADC count from the analog to digital converter. The PHSNA controller uses a 10-bit A to D conversion. The ADC cont is of little interest to us and we suspect it is left over from early firmware testing. The data is formatted so that each line ends with a new line character pair and fields are separated by commas. This format is commonly known as "comma separated value" and generally has a file extension of ".CSV". Many graphical applications, including Excel, can read and load CSV files directly.

After all the data rows have been transferred, the PHSNA firmware indicates the end of data with a special directive that includes the number of rows transferred. The NAT firmware ignores the received row count but does recognize the directive as the end of data indicator.

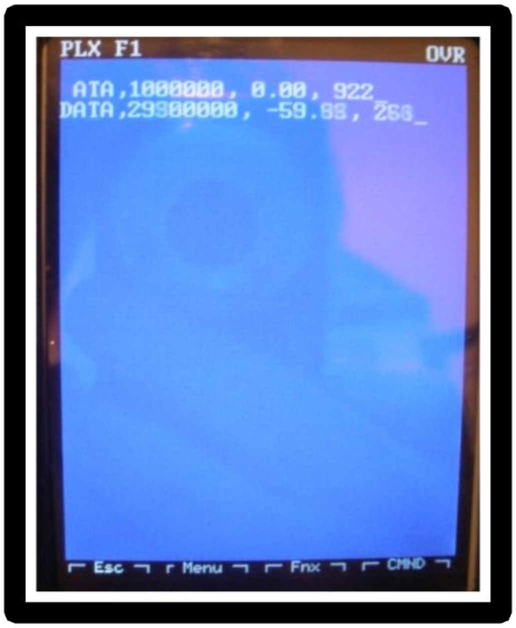
The next line is a flag which, when set to "100" tells the PHSNA firmware not perform the scan and return to Menu mode. Any other value in this field tells the PHSNA firmware to perform the scan. The modified PHSNA firmware uses a PLX FLAG value of 1 to request a shortened version of the data row. This format significantly speeds up the scan. The NAT firmware will accept either data row format so if the standard PHSNA firmware is being used, the old data format will be accepted and properly interpreted.

The next line (CENTER FREQ) gives us an alternate way to specify the frequency range. If we make an entry in this field, the NAT firmware interprets that value as the mid-point of the frequency scan and uses the FREQ ST and ROWS entries to calculate the FREQ LO and FREQ HI values. This feature is provided strictly as a convenience to simplify the setup for some types of scans.

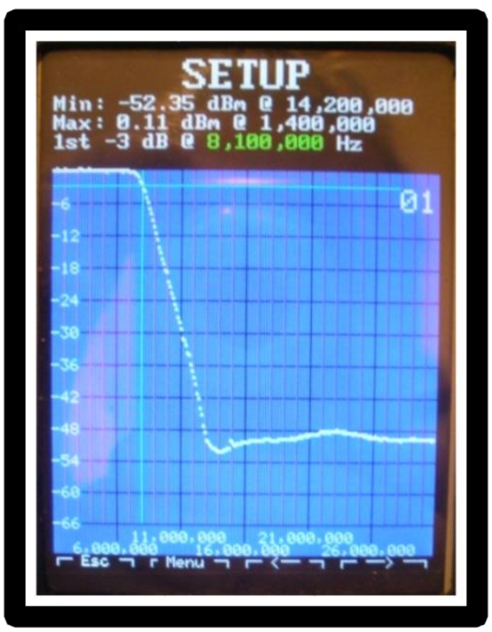
The remaining rows on the form are reserved for future use. The NAT firmware provides the ability to edit the form and change the parameter names (first fifteen columns) and specify alternate spreadsheet cell locations. If you have control over the PHSNA firmware and want to add features requiring additional parameters, this can be done with editing of the master form, NAT firmware does not have to be changed. The only restriction is that the first six and the eighth rows have specific meanings to the NAT firmware. The row names and spread sheet cell references can be changed but their meaning to the NAT firmware cannot be changed.

Once the macro form has been filled in, entering Enter will save the macro to EEPROM and switch back to Terminal mode. Pressing one of the fourteen function keys in Terminal mode will activate the associated macro. With the F1 macro we have been working with here we will see the following display:

The NAT forces a PHSNA firmware reset. The PHSNA firmware initializes itself and then displays some splash screen information. It then tries to access spreadsheet L2. If that is successful, it announces "Mode changed to PLX", reads the parameter cells, and does the scan. Just before the data transfer starts, the NAT clears the display and disables scrolling resulting in a display something like this:



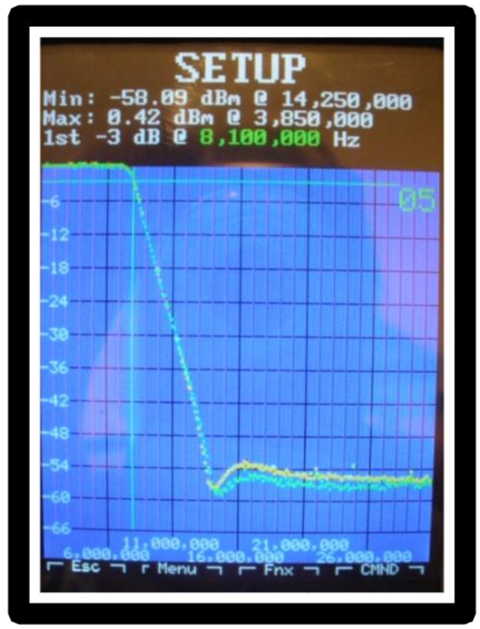
Here each data row after the first is overwritten on the second line to give a progress indication. When all data rows have been transferred and the row count directive is received, the NAT firmware plots the data.



This should look a lot like our Menu mode example with one exception. This plot display has a title ("SETUP") which is taken from the first parameter of the MACRO form.

Also, note the number in the upper right corner of the plot area ("01" in this case). It is possible to over-plot additional traces on the same plot screen. Each time a trace is added, the plot number will be incremented. Eight colors are used in sequence for each plot and the plot number is rendered in the color of the last plot.

Being able to over-plot is a big advantage for using PLX mode over menu mode. Each time a trace is plotted, entering a Space or tapping the plot area will initiate another scan and plot the results. The following plot display illustrates this feature:



Here we see we have plotted five times, the original and four over-plots. The color of the fifth plot is green. Since all plots were perfectly overwriting the previous ones, when the fourth plot was done, I wrapped my hand around the LPF to try to change the plot enough to show a difference. The yellow trace is the result. The heading information (Min, Max, ...) is not updated, it only applies to the first plot. The plot scaling is also computed from the first data set so if there is a significant change in power levels, the plot may run off the plot display area.

Over-plotting in PLX mode can be very handy when testing a tunable DUT. First, we don't have to go through the menu select and prompt response needed for each plot in Menu mode. Second, we can see all plots on the same display and easily see changes while Menu mode clears the plot area and only plots one at a time. It is also possible for us to specify an automatic start mode where the firmware will repeatedly execute the macro with about one-quarter second between scans.

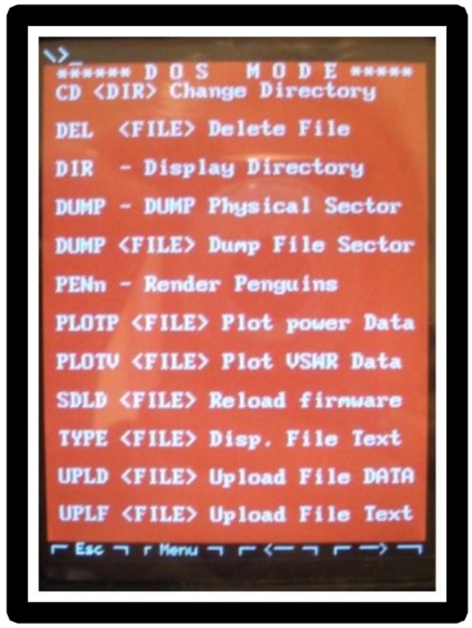
DOS MODE

The third and last basic NAT mode is DOS mode. DOS mode gives us a set of DOS-like commands to help manage and play back spool files on the SD card. To enter DOS mode, we must first switch to Command mode. This is done by pressing and releasing the Scroll Lock key or by tapping the CMND permanent button at the lower right corner of the display. DOS mode is one of the Command Mode menu options. DOS mode starts with a Directory (DIR) display.

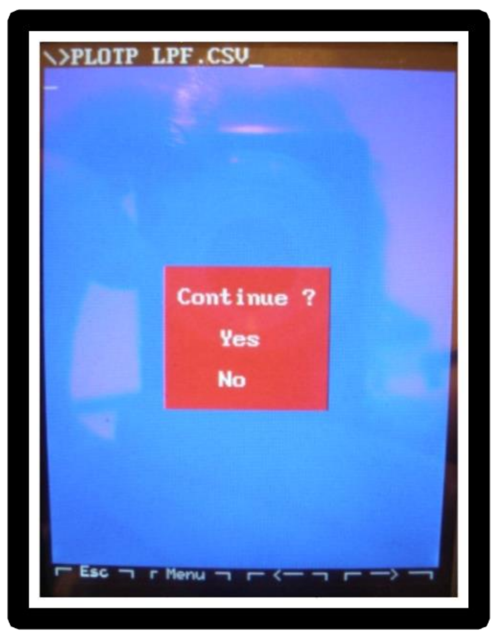
The DIR display lists the contents of the first sector of the current directory. Each directory sector holds up to sixteen entries. Here we see the first sector is not completely full, there are only 11 entries plus the directory end flag. Note the two permanent buttons at the bottom right of the display have changed to left and right arrows. Entering these will navigate to the previous or next sector in the directory. The current directory sector number is shown in the top right corner of the blue area of the display. The current volume name is shown in the top left ("DEMO 1" in this case)t.

The top line of the display is the DOS command line. The command line prompt follows DOS standards and will show the total path to the current directory. If the current directory is a sub directory the prompt will contain the directory name of the current directory and any parent directories. The back slash represents the root directory.

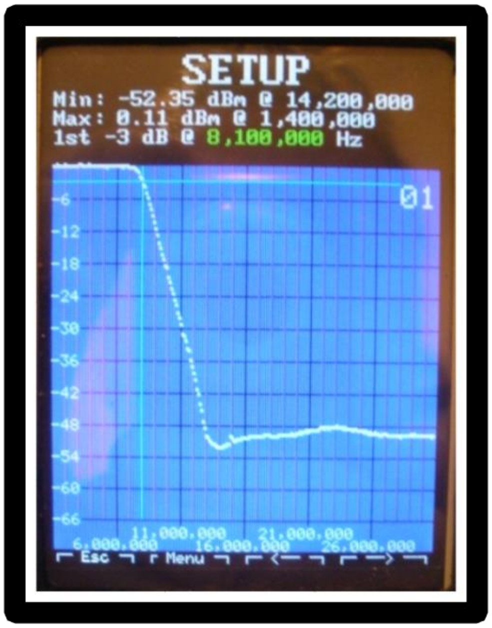
Touching a file name on the display will bring up a context menu containing the available DOS commands.



Touching a line in the context menu will select the associated command and display the confirmation display.



Note here that the command line has been filled in with the selected command and the selected file name. This information could have been entered from the keyboard when the DIR listing was first displayed. In this case, the conformation screen is skipped and the data from the specified file is plotted immediately.



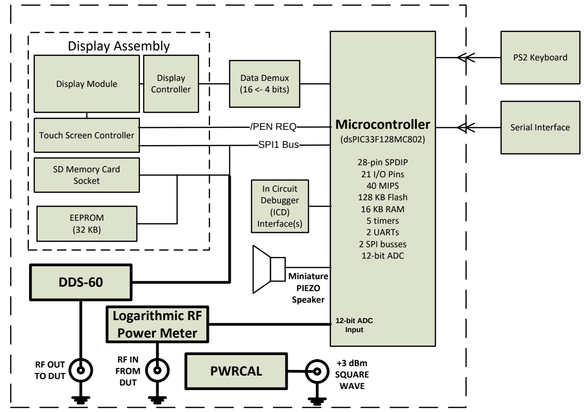
Look familiar? In PLX mode, spooled data is always preceded with the data from the macro form. This is where the DOS mode firmware got the plot title.

This completes the basic description of the NAT and sets the stage for the Midnight Scalar Network Analyzer (MSNA) discussion.

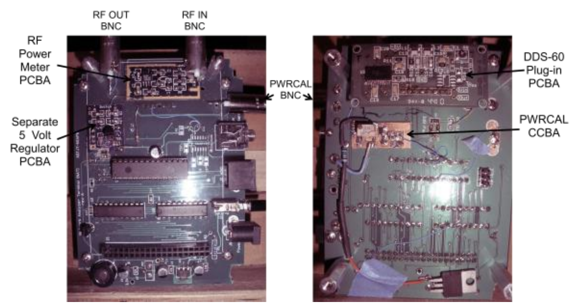
MSNA BASICS

At this point in our project, we had all the PHSNA functionality and then some available with the NAT and still had lots of program memory and PCB space available. We then set out on a two step process. The first step was to incorporate the PHSNA firmware and DDS functions into the NAT thus creating the Network Analyzer Controlling Terminal (NACT). Since George (N2APB) had wisely added a DDS-60 connector to the original design, the only other hardware enhancements were the addition of two BNC connectors and the wire to hook them up. The NACT still required an external power meter.

The second step was to add an integral RF Power Meter to the NACT giving us a fully-featured, self contained SNA we named the "Midnight Scalar Network Analyzer" or "MSNA for short. We also took the opportunity to add a simple circuit to aid the calibration of the RF Power Meter.



Here we see the hardware enhancements shown with heavy lines and bold print. The following pictures show how the hardware enhancements can be implemented on a standard NAT PCBA.



The RF Power Meter is a very sensitive to noise so it was decided that it has to be implemented on a PCB with special attention to grounding and noise suppression. fortunately, the RF Power Meter design used in the PHSNA has been around a while and there have been countless implementations. The original design was done by Wes Hayward, W7ZOI and is described in detail in several places[5]. We have used two sources for the PCBs and both layouts are about the same size and easily modified to fit our needs. The one shown in the picture above is an OSH Park project by Dick Faust, K9IVB[6]. A similar board is available from time-to-time from the PHSNA project[3]. Detailed instructions are available for adding the RF Power Meter board from either of these sources[7]. A separate voltage regulator is recommended to minimize noise pickup.

We now had integrated the RF Power Meter and were able to take advantage of the internal PCB location and simplify output circuitry. Since we still had some board space available, we decided to add a simple circuit to generate a known power level RF signal that could be used to calibrate the RF Power Meter which could then be used to set the DDS-60 output to a known power level. We named this circuit "PWRCAL". It is driven by the firmware, generates a 1 MHz square wave at a power level of +3 dBm, and can be set up using only a digital volt meter capable of measuring dc voltages in millivolts. The PWRCAL design is based on a couple articles in QEX written by Bob Kopski, K3NHI[8].

ADDED FEATURES

Once we had the hardware enhancements defined and the supporting firmware written and running, we discovered we still had some program memory space available, even after leaving all the PHSNA support functions in. We needed more features, of course.

Since we were reasonably confident we could calibrate the RF Power Meter, we added a QRP power meter function. We also added a signal generator mode that could be controlled either from the keyboard, the touch screen, or remotely over the serial interface by a PC application[9]. Since the serial port connection was no longer required, we added the ability to transfer files back and forth between the SD card and a PC application over the serial interface thus eliminating the need to remove the SD card and insert it in a PC's card reader.

QRP RF POWER METER

The nominal range of the RF Power Meter is from about +20 dBm to about -80 dBm. You lose a small part of this range due to roll-offs at the two ends but still wind up with over 95 dBm of dynamic range. To minimize the noise floor as much as possible, the firmware turns off the DDS-60 in power meter mode except for a scaling operation described later.

There are actually three power meters displayed: current, average, and peak. In power meter mode, the input power level is sampled approximately 200 times per second (every 5 ms). The current power reading (top meter) is a 10-point running average of the power meter readings. The average power reading (second meter) is a 200-point running average of the power readings. The peak power meter (bottom meter) displays the maximum power level measured over the last second. The current and average readings are updated every 5 ms, the peak reading is updated once a second.

Each power meter displays the power level three ways. The upper number is the power level in dBm and the lower number is the power level in watts scaled to picowatts, nanowatts, microwatts,, milliwatts, and watts. The power level is also shown graphically by the horizontal green bar. The scale for the bar is shown above the bar in dBm and below the bar in milliwatts. The white vertical line in the graphic meter is the 0.00 dBm (1 milliwatt) point.

To change the scale, connect the DDS output to the RF Power Meter through an attenuator. A hot key (CTRL-G) tells the firmware to take a reading to adjust the scale. When this happens, the firmware momentarily turns on the DDS, pauses for everything to stabilize, records the power level, and then turns the DDS off again. The power reading is then displayed in the button at the bottom of the display and the power scales are adjusted to show the power levels with the attenuator in the RF Power Meter input.

Here we see the results of doing the calibration with a 10 dBm attenuator (obviously a cheap one). The Gain button now shows the power reading that was used to shift the meter scales and scale the readings. Note that the power level shown in the above picture is about 10 dBm higher than the first picture and the 0.00 dBm vertical bar has shifted some to the right. The power meter scale now runs from -70 dBm to +30 dBm raising the top level from about 0.1 watt to about 1.0 watt.

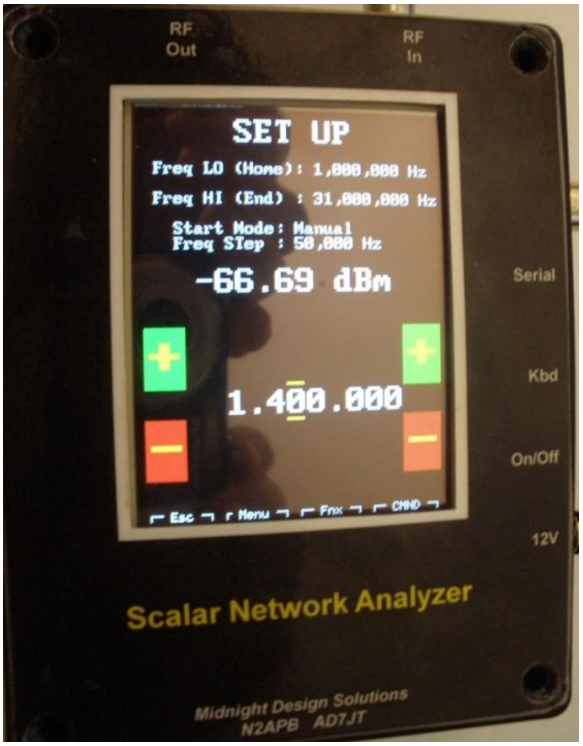
Note that the power reading illustrated in the above pictures is with no input to the power meter except any radiation picked up by the attenuator and coax cable acting like an antenna. When we disconnected from the DDS and connected the coax to an RF source with a power level of about 0.25 watts we got the following readings:

The 30 dBm upper limit on the scale is probably not usable. Since the power curve rolls off at either end we should not use anything within 5 dBm of the end. Thirty dBm is only one watt so the upper limit would be something less, say, 0.75 watts (28 dBm). A better choice for the attenuator would be 20 dBm which would slide the scale to cover from -60 dBm to +40 dBm. This puts the nominal upper limit at 10 watts, a more practical range for testing the output of our QRP rigs.

A word of caution is in order here. The attenuator you use must be able to handle the power and the attenuated power level should not significantly exceed the upper limit of the RF Power Meter (+20 dBm). If you need to measure power levels above 5-10 watts, we recommend you use another type of power meter. Please don't try to measure your legal-limit amplifier output with the MSNA and a 70 dBm attenuator!

SIGNAL GENERATOR

In signal generator (sig-gen) mode we can control the output frequency manually using either the keyboard of the touch screen. Signal generator mode is specified and certain limits are controlled by a macro form just like in PLX mode. The FREQ LO and FREQ HI parameters limit the frequency range. This can be handy keeping us within a specific range such as an amateur radio band while testing a receiver or driving a transmitter as a VFO. The sig-gen display is as follows:



The top line is from the title parameter in the macro form. The frequency limits are also from the macro form and will limit the range of the signal generator. Entering Home will force the sig-gen frequency to FREQ LO and entering End will force the sig-gen frequency to FREQ HI.

Touching the green and red buttons will increment and decrement the frequency by the FREQ ST value. If we try to move through FREQ HI or FREQ LO, you will be BEEPed and the frequency will be set to the limit value. The number between the buttons is the current frequency. The yellow bars above and below one digit position define the most significant digit position of FREQ ST. The yellow bars can be moved by touching another digit position or with the left and right arrow keys. Each digit position the bars are moved will change FREQ ST by a factor of 10.

Normally power level readings are only taken when we tap the touch screen just below the frequency display or press the Space bar on the keyboard. This is "Manual" Start Mode as shown above. Tapping the Start Mode line or pressing the 'A' key will change to automatic mode. In automatic mode, a power reading is taken approximately every 0.25 seconds.

REMOTE CONTROL

The PHSNA group has one or more PC applications that remotely control the PHSNA. A set of basic commands has been implemented and is used with special PHSNA firmware running in the PHSNA controller. When in sig-gen mode, the MSNA monitors the serial port for remote control commands. When a command is received, the MSNA responds as the PHSNA firmware would. The following remote control commands are recognized:

- ENQ 0x05 ENQ Check for PHSNA/MSNA
- ACK 0x06 ACK I'm here response to ENQ
- FQU 0x11 DC1 Set DDS frequency
- FRQ 0x91 Set DDS frequency and return power level
- ADC 0x12 DC2 Request ADC count (10-bit)
- REV 0x13 DC3 Request current firmware revision
- MOR 0x14 DC4 Send character in Morse code
- RFY 0x0E SO Turn DDS on
- RFN 0x0F SI Turn DDS off

In the above list, the first column is the PHSNA mnemonic, the second column is the hex value of the command character, the third column is the standard ASCII character name (if any), and the last column is a description of the requested action. The power level is returned as an integer equal to the power level in dBm times 100. This allows two decimal place accuracy.

FILE TRANSFER

File transfers between SD card files and a PC over the serial interface require a PC application to be running. Most terminal emulator programs have a file transfer function that transfers file data directly between the serial port and a storage device on the PC (e.g., the HDD or a thumb drive). Transfers from the SD card to the PC are performed with a DOS command, transfers from the PC to the SD card are performed in terminal mode with a hot-key command.

Since most modern PCs today do not come with RS232 ports and the MSNA hardware does not generally support RS232 signal levels, some sort of adapter will probably be required to interface the serial port to the PC. There are many sources (e.g., eBay) for these adaptors at very low costs (usually less than $5).

File Input (Upload to the PC)

There are two DOS command for transferring files to the serial interface. The first, Upload File Data (UPLD), transfers data exactly as read from the SD card and will handle binary data (even though binary data is not currently used by the MSNA firmware). The second, Upload File Text (UPLF), assumes the file is an ASCII text file and scans the data to assure that all new lines consist of a Carriage Return (CR) and a Line Feed (LF) as is required by most PC applications. In either case, a command line argument names the file to be transferred. There are no restrictions on file names other than all SD card file names must adhere to the 8.3 file naming standard. To initiate the transfer, start the file transfer function in the PC application and then execute the DOS command specifying the file to be transferred.

File Output (Download from the PC)

File data is received from the serial interface in Terminal mode. The received file data is stored in the current log file. To initiate the transfer:

1. Confirm the serial interface baud rates are set properly and the proper PC port number has been assigned to the terminal emulator application by keying data from both keyboards and verifying correct transmission.
2. Have the MSNA idling in Terminal mode.
3. Select the file transfer option in the PC application and select the file to be transferred **but do not yet start the transfer**.
4. Enter the file transfer hot key (see below) to start the file transfer function in the MSNA.
5. Within three seconds of entering the hot key, start the file transfer function in the PC.

There are two hotkeys to start a file transfer:

- CTRL-Z will not overwrite an existing file with the current log file name. In that case, a warning will be displayed and the transfer will be aborted.
- ALT-Z will delete an existing file with the current log file name and write a new file with the same name.

Once the hot key is entered, the firmware will wait approximately three seconds for the transfer to start. If it does not start, the MSNA firmware will abort the transfer and return to normal Terminal mode. Once the transfer has started, it will continue until the MSNA detects a pause in the data stream of over about one-half second. At this point the MSNA firmware assumes the transfer has completed and it closes the file and returns to normal terminal mode.

File download is mainly used to load firmware updates to the SD card for later loading with a Reload Firmware (SDLD) DOS command. It is possible, however, to down load a file to be plotted on the MSNA using one of the DOS PLOT commands.

The .HEX file used to update the firmware is, in fact, an ASCII text file. It is also quite large normally being nearly 500KB. We have found some terminal emulators have trouble handling file transfers this large. The one we use is term232[10] which is free, very easy to use, and does not seem to have a problem with large files.

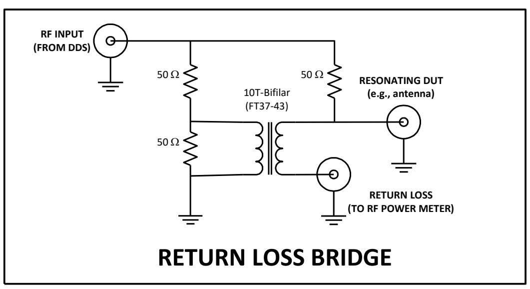
TEST FIXTURES

Up to now we have been discussing MSNA applications where the DUT connects direct to the RF out and RF in connectors on the MSNA. RF is fed direct to the DUT input and the power readings are taken direct from the DUT output. Some applications may, however, require an adaptor or test fixture to, for example, separate RF signal components or match impedances. Functions requiring test fixtures generally require special support in the firmware. Two such fixtures are currently supported by the MSNA firmware: the return loss bridge (RLB) for antenna analysis and the crystal test fixture (CTF) for grading and matching crystals usually for use in crystal filters.

RETURN LOSS BRIDGE (RLB)

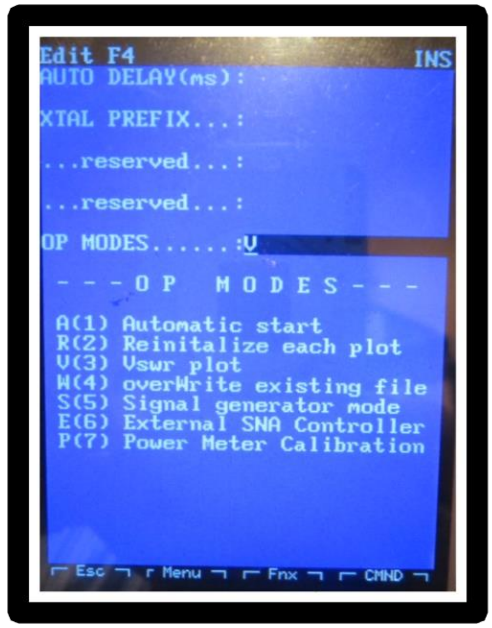
Antenna analysis and tuning usually involves feeding the antenna with an RF source of varying frequency and measuring the amount of energy (power) radiated by the antenna. The portion of the RF energy that is not radiated is returned or "reflected" back to the source (e.g., your transmitter output stage). Returned power is generally referred to as "return loss" and the comparison of the return loss with the power fed to the antenna is a measure of the inefficiency of the antenna. The difference between the power radiated and the power fed is a measure of the efficiency of the antenna and is generally expressed as the Standing Wave Ratio (SWR). The efficiency can also be expressed as a relation between signal voltage levels or Voltage Standing Wave Ratio (VSWR).

To measure the return loss in an antenna, a test fixture is needed to separate the power fed in from the returned power. A simple device to do this is the Return Loss Bridge (RLB). The RLB shown here is from the pages of Experimental Methods in RF Design[11].

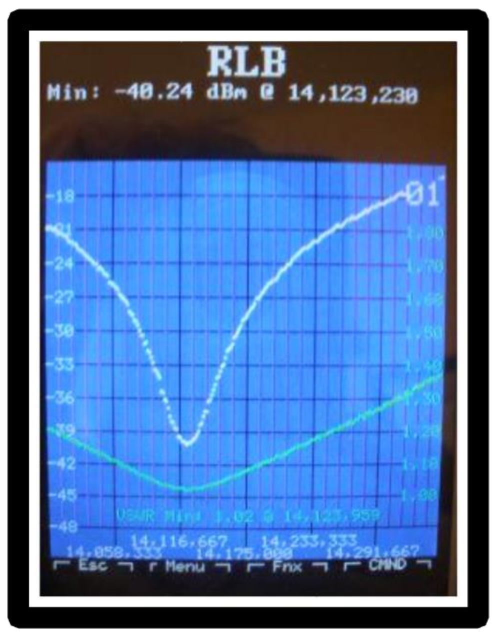


This RLB consists of a single, bi-filer wound toroid, three resistors, and three connectors. The RF output from the DDS is fed to the RF input connector of the RLB. The RF is fed in turn to the resonating DUT (aka "the antenna"). The toroid and resistors form a bridge which separates the power fed to the antenna from the return loss from the antenna. The return loss portion of the antenna is routed to the third connector which connects to the MSNA RF input connector.

The bridge is used in PLX mode using a macro form to define the frequency range etc. A second page has been added to the form which includes a couple new parameters plus an operating (or "op") mode entry.



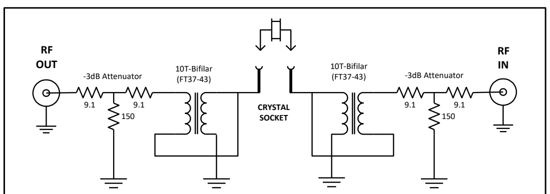
One of the op modes (C) directs the firmware to use the on-board DDS-60 and to not communicate with the PHSNA over the serial interface. Another op mode (V) directs the firmware to compute and plot the VSWR in addition to the normal power curve.



The example shown here is a sweep of a 20M vertical antenna. The sweep covers from 14MHz to 14.35MHz with 481 steps. The white trace is the return loss expressed as a negative value. The green trace is the VSWR with the VSWR scale on the right side of the plot also in green. The minimum VSWR and the frequency at that point are displayed in green just below the VSWR plot. Entering Space or tapping the plot area will repeat the sweep and over plot the results. The VSWR reading below the VSWR plot will be updated too.

CRYSTAL TEST FIXTURE (CTF)

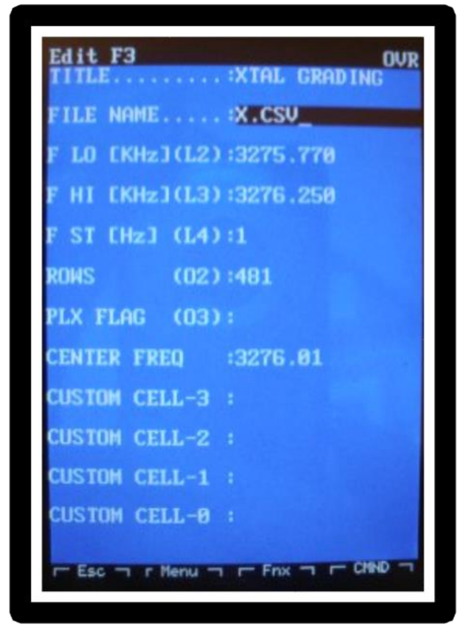
In a crystal filter, the crystals used must be graded and matched for the filter to meet its design objectives. Several parameters may be considered such as the motional parameters, the quality factor, and the series resonant frequency. Assuming the crystals meet the basic requirements it important that the resonant frequencies of all crystals be close. A common rule of thumb calls for the frequency spread among the crystals in the filter be less than 10% of the desired filter bandwidth. A Crystal Test Fixture (CTF) allows the MSNA to measure the crystal parameters and very accurately determine the frequency spread in a group of crystals.



CRYSTAL TEST FIXTURE

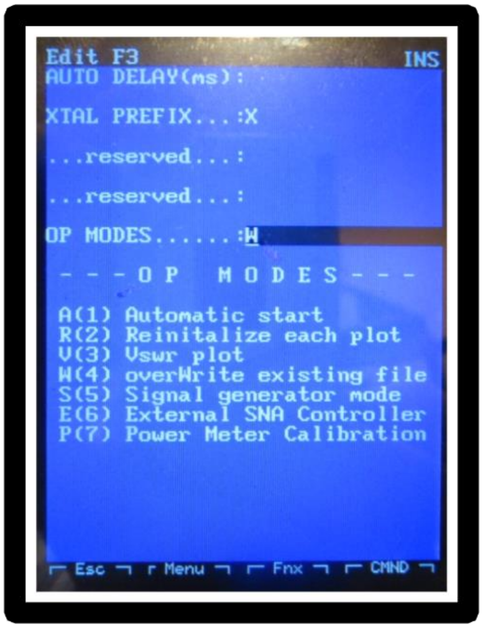
The purpose of the CTF is two-fold. First, crystals respond best to very low signal levels and low loading so the CTF input and output are attenuated. Second, crystals generally have a natural impedance other than the 50 ohms the MSNA DDS and RF Power Meter are designed for. Crystal impedance is generally about 12.5 ohms so the CTF includes bifilar wound toroids to convert from 50 ohms to 12.5 ohms and back again. Note that the crystal impedance may be another value, an MSNA editing option allows us to specify most any value.

Since we are working with very narrow bandwidths here, the frequency sweeps need to be done with minimum size steps, usually no more than 1 Hz. Expect a fair amount of frequency variation in any batch of crystals so some pre screening may be necessary to get the batch results displayed on one plot screen. This is where the CENTER FREQ parameter in page one of the macro form comes in handy.



The example used here was to find six crystals for use in the PHSNA Measurement Receiver's filter section. The goal was a pass-band filter with a 400 Hz pass band and center frequency around 3.27 MHz. This means we need to find six crystals with a frequency spread of less than 40 Hz. One approach would be to special order the crystals and have the manufacturer grade and match them. This can be very expensive. A better approach is to do it ourselves.

To accomplish this, 30 3.2768 crystals were ordered from Mouser for $0.50 apiece, $15.00 for the lot. After testing a number of crystals to locate a midpoint where all would be within the plot limits, the crystals were organized so we could later match crystals to results (empty egg cartons were used for this high-tech function). As the above screen shot shows, the row count was set to 481 and the FREQ ST was set to 1 Hz. The located center point of 3276.01 Hz was entered into the CENTER FREQ field and the firmware calculated the start and end frequencies. We also gave the macro a title and entered the spool file name as "X.CSV".

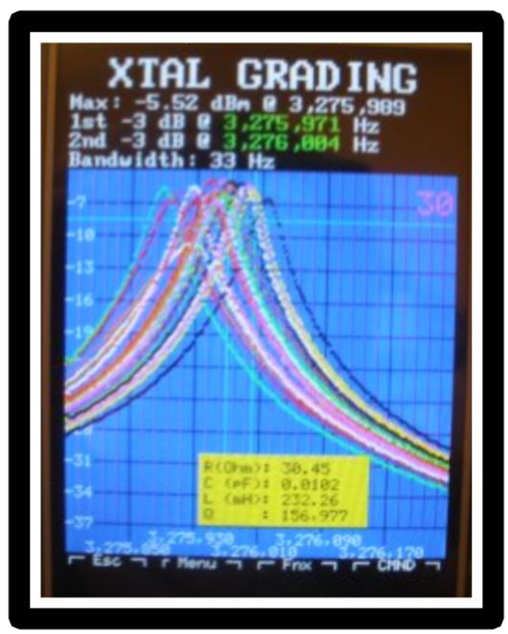


The second page of the macro form was edited to include the C and W op modes. The W op mode directs the firmware to overwrite the spool file every time the macro is activated. This is advisable since it is easy to make a few false starts when grading crystals and the W op mode avoids having to delete the spool file or change its name for every restart.

There is another important parameter to set on the second page. The XTAL PREFIX parameter defines a prefix for naming the crystals and marking the result set for each crystal. As each crystal is tested and the results (over-) plotted, each data set is prefixed (ID'd) with the XTAL PREFIX concatenated to the plot number (in this case "X1", "X2", "X3", etc.). This parameter, when not empty, also directs the firmware to spool, not the plot data but, the crystal ID, the series resonant frequency, the motional parameters, and the quality (or "Q") factor as a single data row for each plot/crystal.

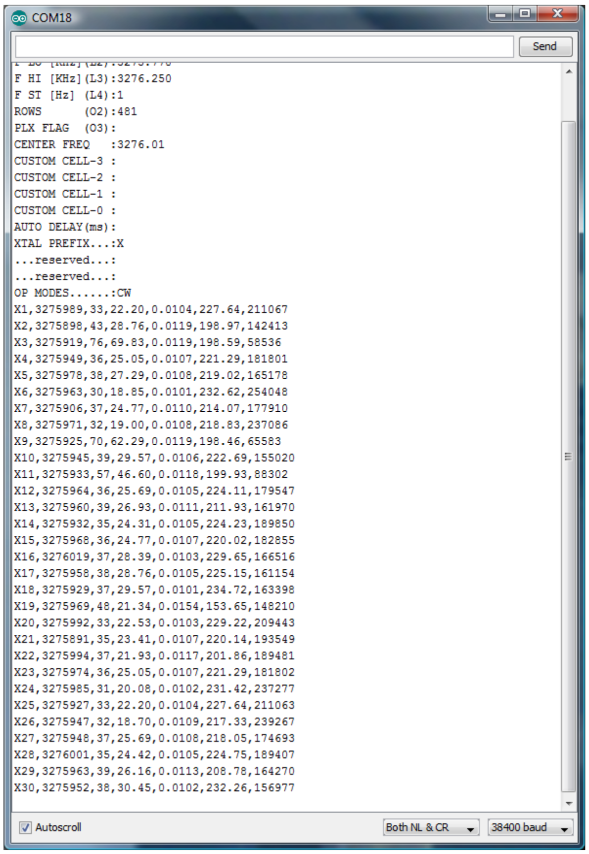
Once the Crystal impedance is specified, the CTF is calibrated (described later), and everything is set up, the first crystal is placed in the CTF and the macro is started. Here's what our first plot looked like:

The firmware recognizes the plot as a pass band filter so there are two -3 dB points. Note the extremely narrow band width of only 33 Hz. The peak and bandwidth are used to calculate the motional parameters and the Q of the crystal. These values are displayed in the yellow window.



The second crystal is then put in the CTF and entering Space or tapping the plot screen will initiate another scan and the results over-plotted. The data in the yellow window is updated with each plot. We then repeated this process for all 30 crystals. A new data row was spooled for each crystal. The above screen shot shows the results after all 30 crystals had been processed. Note the variation in series resonant frequencies and insertion losses. These crystals were all marked with the same frequency and, presumably, came from the same production lot.

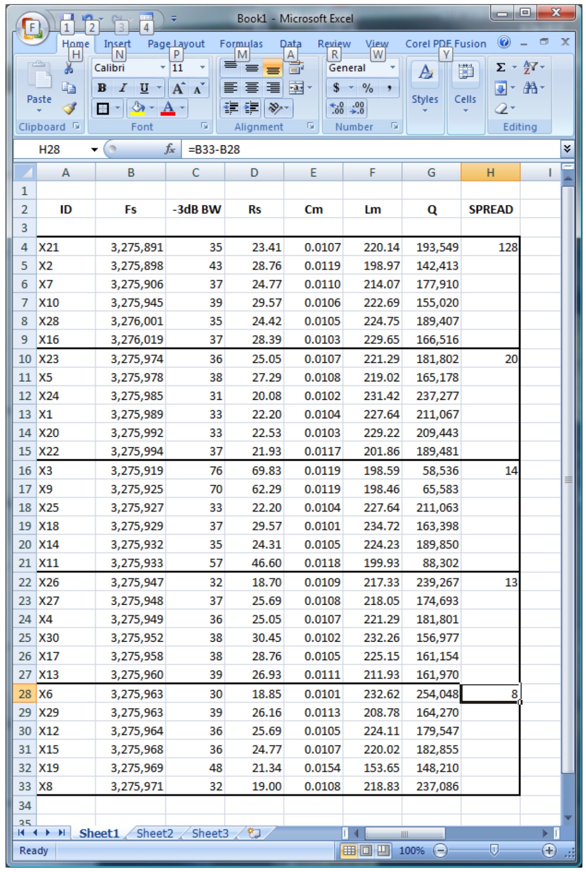
Now that we have the data spooled, what do we do with it? That's the easy part. We could play back the data and write everything down and then manually sort and match the parameters or we could import the data to an Excel spread sheet and let Excel do the drudge work. We chose the latter approach. We used the UPload File Text (UPLF) DOS command to input the data set to a terminal emulator on our PC.



This is a screen shot from the terminal emulator screen and buffer. As usual, the file starts with the contents of the macro form used to generate the data set. This is followed by the data set itself. As you can see, the data set consists of 30 rows of seven fields each, one for each crystal in the batch. The first field in each row is the crystal ID ("X1" through "X30").

Next, we copied and pasted the 30 rows of crystal data into an Excel spreadsheet and used Excel's data-to-columns function to separate the parameters. We then sorted the 30-row data set by frequency and added an equation in the eighth column to compute the difference between the frequency in the data row five down from this row and the frequency in this row. This equation gives us the frequency range for each possible group of six crystals in the sorted set. Scanning this column for the smallest value gave us the best group of six crystals for our filter.

After isolating this group of six from the remaining 24 data rows, we repeated the process until we had identified the five groups of six crystals with the smallest frequency spread. Here are the results:

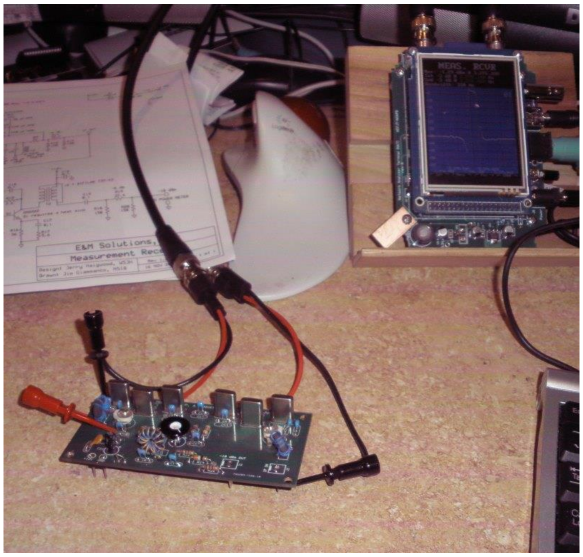


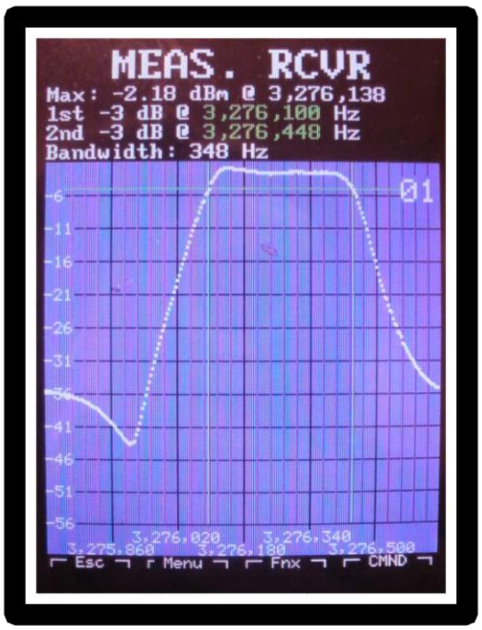
H28 =B33-B28

ID	Fs	-3dB BW	Rs	Cm	Lm	Q	SPREAD
X21	3,275,891	35	23.41	0.0107	220.14	193,549	128
X2	3,275,898	43	28.76	0.0119	198.97	142,413	
X7	3,275,906	37	24.77	0.0110	214.07	177,910	
X10	3,275,945	39	29.57	0.0106	222.69	155,020	
X28	3,276,001	35	24.42	0.0105	224.75	189,407	
X16	3,276,019	37	28.39	0.0103	229.65	166,516	
X23	3,275,974	36	25.05	0.0107	221.29	181,802	20
X5	3,275,978	38	27.29	0.0108	219.02	165,178	
X24	3,275,985	31	20.08	0.0102	231.42	237,277	
X1	3,275,989	33	22.20	0.0104	227.64	211,067	
X20	3,275,992	33	22.53	0.0103	229.22	209,443	
X22	3,275,994	37	21.93	0.0117	201.86	189,481	
X3	3,275,919	76	69.83	0.0119	198.59	58,536	14
X9	3,275,925	70	62.29	0.0119	198.46	65,583	
X25	3,275,927	33	22.20	0.0104	227.64	211,063	
X18	3,275,929	37	29.57	0.0101	234.72	163,398	
X14	3,275,932	35	24.31	0.0105	224.23	189,850	
X11	3,275,933	57	46.60	0.0118	199.93	88,302	
X26	3,275,947	32	18.70	0.0109	217.33	239,267	13
X27	3,275,948	37	25.69	0.0108	218.05	174,693	
X4	3,275,949	36	25.05	0.0107	221.29	181,801	
X30	3,275,952	38	30.45	0.0102	232.26	156,977	
X17	3,275,958	38	28.76	0.0105	225.15	161,154	
X13	3,275,960	39	26.93	0.0111	211.93	161,970	
X6	3,275,963	30	18.85	0.0101	232.62	254,048	8
X29	3,275,963	39	26.16	0.0113	208.78	164,270	
X12	3,275,964	36	25.69	0.0105	224.11	179,547	
X15	3,275,968	36	24.77	0.0107	220.02	182,855	
X19	3,275,969	48	21.34	0.0154	153.65	148,210	
X8	3,275,971	32	19.00	0.0108	218.83	237,086	

As you can see, the best group of six crystals had a frequency spread of only 8 Hz. Three of the other groups met our criteria with spreads of 13, 14, and 20 Hz. The last group with a spread of 128 Hz did not meet our requirements

To test our results, we built up the filter section of the PHSNA Measurement Receiver and checked its band pass characteristics.





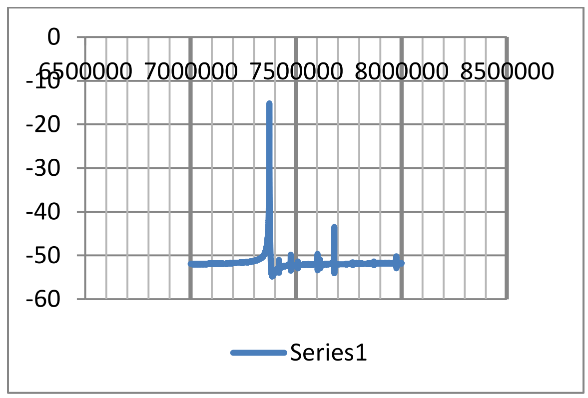
As you can see, we have a -3 dB pass band of 348 Hz with a center frequency of 3,276,274 Hz. This is without any tuning of the filter. The PCB design has lots of extra pads for tweaking capacitor values.

OTHER TEST FIXTURES

There are a number of other test fixtures that could further expand the MSNA's utility in the shack and in the shop. These will generally require firmware enhancements to support them. We are currently against the wall as far as available, unused program memory space in the MSNA. But fear not, the ability to easily download new firmware from the SD card makes possible firmware overlays that can be loaded as needed or complete new, dedicated applications to support additional test fixtures and new functions.

Prime candidates for new test fixture support includes the PHSNA Measurement Receiver (aka the "Poor Ham's Spectrum Analyzer" or simply the "PHSA"). The problem with the PHSA is the frequency range to be analyzed and displayed at the highest resolution possible. The current planning is for the MSNA firmware to record peaks and spool the data during the frequency scan. The peaks would then be shown (plotted?) on the display. The resulting data set could be read back and displayed a section at a time in a graphical scrolling window.

We can illustrate the problem by scanning a 40 M crystal over a 1 MHz frequency range at a resolution of 50 Hz per step. The 20,000 point data set was imported to Excel and plotted.



The exercise was to locate the resonant frequency of an unmarked crystal. We were able to reduce the size of the data set because we suspected it was a 40 M crystal. The MSNA will only display the first 920 data points of this plot because that's the size of the data buffer. It will, however, spool all the data points the scan. Currently, the data has to be imported to a PC application for analysis as we did here. It would definitely be an advantage to be able to play back the spooled data and plot it on the MSNA screen without needing the PC.

Another test fixture candidate is one for measuring impedance. This would be most valuable for checking the impedance of inductors wound for use in filters. One way to do this would be to put a known precision capacitance in series with the inductor and use the MSNA to determine its resonant frequency and compute the inductance of the inductor. Of course, this could be done now with us doing all the calculations but it would be nicer for the firmware to do them.

CALIBRATION

No discussion of an SNA would be complete without considering calibration. In many instances, the things we hams do with an SNA do not require extensive calibration. We are mostly interested in RF power ratios, not absolute power levels. The following equation is what the MSNA (and most any SNA) uses:

$$\text{Power ratio in dB} = 10 \log(P_{out}/P_{in})$$

Where P_{in} is the input power to the DUT and P_{out} is the output power from the DUT. Since we are generally dealing with passive DUTs, we can assume that any scaling applied to P_{in} will result in the same scaling of P_{out}.

The MSNA RF Power Meter uses the AD8307 logarithmic power amplifier/meter. This device measures absolute power levels and compares them to a standard power level of 0.001 watt. This ratio is the definition of a dBm (the "m" stands for milliwatt). In other words, the RF Power Meter reading (PM) assumes that P_{in} is one milliwatt and

$$PM = 10 \log(P_{out}/P_{in})$$

$$= 10 \log(P_{out}/.001) = dBm$$

For the rest of our discussion assume P_{out} is the DUT output power level when the input power level is one milliwatt. If, for example, the input power is two milliwatts, the output power will also double and we will have the same power ratio:

$$10 \log(2P_{out}/.002) = 10 \log(P_{out}/.001)$$

But the AD8307 will still measure the output power level against one milliwatt (it doesn't know what the real input power level is):

$$PM = 10 \log(2P_{out}/.001)$$

$$= 10 \log(P_{out}/0.001) + 10 \log(2)$$

$$= dBm + 10 \log(2)$$

The RF Power Meter will measure twice the DUT output power level as it would with one milliwatt of input power to the DUT. In this case we can get the desired reading by subtracting 10 log(2) from the RF Power Meter reading.

$$dBm = PM - 10 \log(2)$$

But wait a minute, we said that our new DUT input power level is 2 milliwatts instead of the standard of 1 milliwatt. If we feed a 2 milliwatt signal to the AD8307 we should get the following:

$$dB = 10 \log(.002/.001) = 10 \log(2)$$

If we subtract this value from the RF Power Meter reading with our DUT in the circuit we will get the same dB value as if we were feeding 1 milliwatt into the DUT. This means that we can calibrate our power readings by subtracting the power readings read when the DDS output is fed directly into the power meter with no DUT. (FYI, 10 log(2) is about 3 dB.)

The MSNA can do a sweep across a frequency range and we can save the results in a calibration buffer. When the calibration buffer is not empty, the buffer content closest the current frequency will be subtracted from the current RF Power Meter reading in all future sweeps. You don't need to know the DDS output power level to get an accurate picture of your LPF response.

When a test fixture is used, it must be calibrated to compensate for any fixture insertion loss. In the case of the RLB the calibration sweep can be made with the antenna connection either open or shorted to ground. In the case of the CTF, the calibration run is made with the crystal terminals shorted together.

However, and there is always a "however", there are some other considerations here. The AD8307 generates an analog output at a voltage level that is a function of the input power level. The transfer curve is very linear and does not change much from dc to 500 Hz. Here is the transfer function shown as a graph:

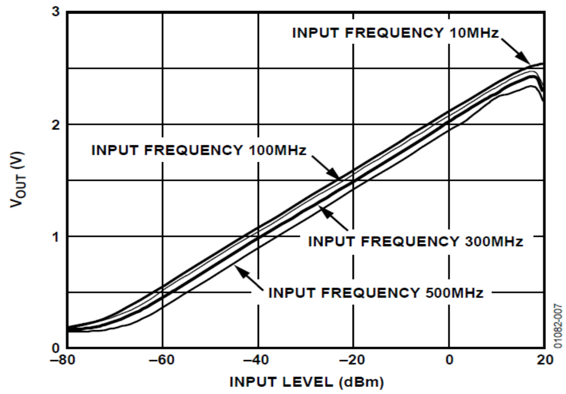


Figure 7. V_{OUT} vs. Input Level (dBm) at Various Frequencies

The nominal range is +20 dBm to -80 dBm however there is some roll off at both ends so the really usable dynamic range is about 95 dBm. If we drive our DUT with a low power level, say 40 dBm, and use the above method to calibrate the power readings, we will be limiting the lower range of the power meter to -40 dBm instead of -80 dBm. We will however have gained the 40 dBm of dynamic range at the high end and still have the 95 dBm of dynamic range.

We normally recommend setting the DDS output level to 0.00 dBm (one milliwatt). "But how can I do that?" you ask. There are a number of methods but most of them require lab-grade test equipment that most of us will never see let alone have access to. To help us calibrate the power meter, the MSNA includes a simple circuit that generates a 3 dBm square wave that can be used for this purpose. The circuit is the PWRCAL circuit mentioned earlier.

To calibrate the RF Power Meter, the PWRCAL output is connected directly to the RF Power Meter input and a reading is taken. This is the +3 dBm reading. A 20 dBm attenuator is then inserted in the connection and a second reading is taken. This is the -17 dBm reading. Two readings are enough for the firmware to determine the slope and zero intercept point of the above power curve. This essentially calibrates the RF Power Meter and its output circuit including the ADC input to the dsPIC microcontroller.

Once the RF Power Meter is calibrated, it can be used to set the DDS output power level. Then the calibration buffer data will only have to compensate for variations in DDS power output levels due to changing frequency. This variation is typically about 4-5 dBm from one to 60 MHz.

CONCLUSION

The basic Network Analyzer Terminal (NAT) started out as a replacement for the PC in a Poor Ham's Scalar Network Analyzer. The NAT's small size saves bench top space and allows hand-held operation of the "terminal." The QVGA full color display gives an immediate visual view of the results of a frequency scan without requiring data to be transferred to a PC application. Field upgradable firmware allows new features to be added and installed by the user.

Hardware enhancements and new firmware versions have made possible the evolution of the NAT into a hand-held, full-featured, stand-alone Scalar Network Analyzer SNA). The original NAT can be user upgraded to an SNA with the addition of relatively few new components. The enhanced firmware retains all of the original NAT functionality plus much more including signal generator and QRP power meter modes. Firmware support is included for specialized test fixtures such as Run Loss Bridges and Crystal Test Fixtures.

For new orders, Midnight Design Solutions now supplies MSNA kits built around a new PCB layout that includes all the enhancements that evolved the NAT into a complete SNA. Work is being done on new firmware to provide new functions and to provide support for additional test fixtures.

The MSNA would be a very useful and valuable tool in any ham's shack and workshop. Its small size and battery operation makes a practical, portable antenna tester and analyzer for remote operation for special events like Field Day and for very remote operations like DXpedtions or for just a day of operating in the park or on a local mountain top.

REFERENCES

[1] http://midnightdesignsolutions.com/

[2] Links to NAT, NAT-SNA, and MSNA documentation may be found on line at: http://midnightdesignsolutions.com/nat/#Documentation

[3] All PHSNA documentation and support are available on line at: https://groups.yahoo.com/neo/groups/PHSNA/info

[4] Parallax Data Acquisition tool (PLX-DAQ) is a software add-in for Microsoft Excel which provides access to spreadsheet cells through a port on the PC. It is available for download from: https://www.parallax.com/downloads/plx-daq

[5] Wes Hayward, W7ZOI; Bob Larkin, W7PUA; "Simple RF-Power Measurement," QST, June 2001, pp 38-43, available on line at: http://www.qsl.net/sz1a/download/build an rf power meter.pdf

[6] Dick Faust, K9IVB, "RF Power Meter" available on line at: http://www.k9ivb.net/RF_Power_Meter/

[7] Dave Collins, AD7JT, "NAT USER GUIDE EXTENSION for Firmware Version 4 with NACT to SNA Hardware Upgrade Instructions," Appendices A and B, pp 14-18, found on-line at: http://midnightdesignsolutions.com/nat/doc/NAT USER GUIDE EXTENSION FOR V4.pdf

[8] Bob Kopski, K3NHI, "A Simple RF Power Calibrator," QEX, Jan/Feb 2004, pp 51-54 and Bob Kopski, K3NHI, "Tech Notes," QEX, May/June 2010, pp 44-43.

[9] Nick Kennedy, WABDU, "PHSNA User's Guide" and other support documents and source code found on line at: https://groups.yahoo.com/neo/groups/PHSNA/files/Windows PHSNA/

[10] term232 is a free utility program by IndustroLogic(r) and available for down load from their web site http://www.industrologic.com/resource.htm#software

[11] Wes Hayward, W7ZOI; Rick Campbell, KK7B; and Bob Larkin, W7PUA; "Bridges and Impedance Measurement" in Experimental Methods in RF Design published by the ARRL, Fig 7.41, page 7.23.