Edited by **David Newkirk, WJ1Z** • Electronic Publications Editor

Frequently Asked Questions About *ARRL Radio Designer* Reports and Report Editor

It doesn't matter whether you're modeling a single resistor or the innards of an op-amp: If you can't make *ARRL Radio Designer*[1] tell you what you want to know about your circuit's performance, you're in quite a fine pickle. And yet this is exactly the situation more than few *ARD* users find themselves in as they arm-wrestle with *ARD*'s Report Editor (Figure 1). Many of us want to ask *ARD* relatively simple questions about circuit performance (what's the gain, what's the impedance?)—and do relatively straightforward things with *ARD* report files (add reports, delete reports, reuse reports with new circuit files). And yet somehow the answers to our questions don't just fall out of *The ARRL Radio Designer Manual*. We therefore present a question-and-answer session aimed at just these issues as they relate to reports and Report Editor in *ARRL Radio Designer 1.5*.

Q1. *The ARRL Radio Designer Manual* says, on page 2-7, that "two predefined reports await us for EXAMPLE1.CKT." Why can't I see those canned reports without first analyzing *example1.ckt*?

A1. *ARD* report files contain only report *definitions*, not report data or report images. *ARD* must therefore analyze an .RP2 file's associated circuit (.CKT) file to generate the data necessary to produce the report(s) defined in the .RP2 file.

If you'd like to be able to view a given report without analyzing its associated circuit file—without running *ARD* at all, in fact—just copy the report to the *Windows* Clipboard (right-click in the report window to pop a menu that includes a **Save to Clipboard** option) and save it to disk as a .CLP file (using *Windows* 3.*x*'s *Clipboard Viewer*, a subset of *Clipbook Viewer* in *Windows* 3.11) or as a bitmap (.BMP) file (via the Clipboard and *Windows* 3.x's *Paintbrush* or *Windows* 95's *Paint*.) (Curiously, *Clipbook Viewer* doesn't install as part of the standard *Windows* 95 install. Its installation file [*clipbk.exe*] is nonetheless present in the Win95 CD-ROM distribution. *Clipbk.exe* doesn't come with Win95's 3.5-inch-floppy distribution, but you can download it via the World Wide Web at **http://www.microsoft.com/windows/software/otherutils.htm**.

[1]Available from ARRL for $150 plus shipping as publication #4882. Contact HQ Publications Sales at (voice) 860-594-0250, (fax) 860-594-0303 or (e-mail) **pubsales@arrl.org**.

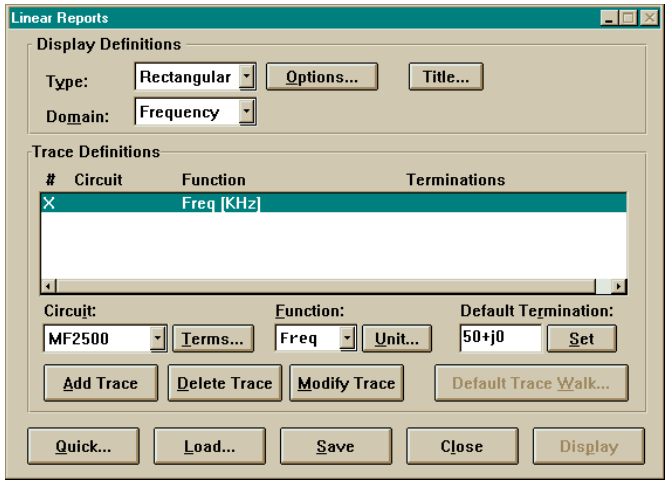

Figure 1—*ARRL Radio Designer 1.5*'s **Linear Reports** dialog, also known as Report Editor. Regardless of whether or not *ARD*'s **Reports | Save Reports** menu option is active, you can save your reports to disk whenever you're looking at this dialog just by clicking **Save**.

Q2. How do I delete unwanted reports from a report file?

A2. Analyze the associated circuit file, and load and tile all of its reports. Close all the reports you want to remove. Once your desktop contains only reports you want to keep, invoke **Save Reports** to rewrite your report file. Hint: Even if the **Reports | Save Reports** option is unavailable ("greyed out," as it appears), you can save reports by opening Report Editor (AKA Linear Reports)—to do so, press F2 or select **Report Editor** on the pull-down **Reports** menu—and clicking its **Save** button (visible near the bottom of Figure 1). After a short pause, you'll see disk activity that confirms the save.

The thing to keep in mind is that saving your reports writes a .RP2 file that defines *only the reports loaded—present on* ARD*'s desktop—at save time*. If, for instance, you've loaded only three reports from an .RP2 file that contains five report definitions, doing a **Save Reports** loses you the two reports you didn't load.

Q3. How can I add reports to a preexisting .RP2 file?

A3. (1) Analyze the circuit file involved. (2) Load all the reports defined in the .RP2 file. (3) *With the preexisting reports still loaded*, define your new reports. Once you've assembled the report suite you want, invoke **Save Reports**. Again: The .RP2 file you save will contain definitions for all reports present—whether you loaded them from the .RP2 file or created them anew—at save time. Any preexisting reports you didn't load are lost.

Q4. I accidentally overwrote .RP2 files for some of the *ARD* example circuits. How can I get them back?

A4. Reinstalling *ARD* is one way. They're also available via *ARD*'s Web page (**http://www.arrl.org/ard/files**) and via *ARD* FTP (**ftp://ftp.arrl.org/ard/**). Giving you the means of uncompressing the *examples.z* file on the *ARD* installation disks isn't an option, by the way. Our license to distribute the *ARD* installation program doesn't extend to handing out copies of its manufacturer's proprietary file compression/decompression software. And, yes, the format of the .Z files decompressible by Nico Mak's excellent *WinZip* archiving utility (**http://www.winzip.com/**) differs from that of *ARD*'s .Z files.

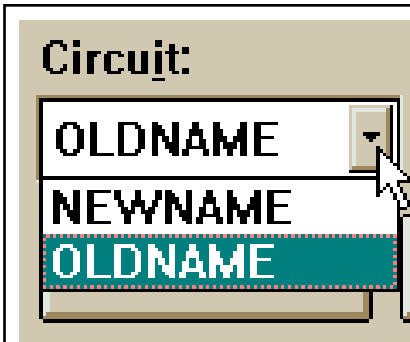

Figure 2—*ARD*'s reports engine doesn't take kindly to trying to graph data it can't find, and merely renaming a circuit will make *ARD* yelp. The text explains how to keep your reports while keeping *ARD* calm.

Q5. Having loaded, analyzed and displayed a report for a circuit file, I renamed one of the file's blocks from OLDNAME to NEWNAME. When I next analyzed the file, I got an error message that said, "REPORT specifies a circuit which does NOT exist: NEWNAME. Graph will be closed...," and the report disappeared from the screen! Have I lost that report definition forever? How can I change a circuit's name to NEWNAME without losing reports that refer to it as OLDNAME?

A5. You won't lose that report unless you do a **Save Reports** while it's absent from your *ARD* desktop. But you also won't be able to load that report—*ARD* will bark that same error message at you again—until you change NEWNAME's name back to OLDNAME. Once you've done that, do this: (1) Use Circuit Editor to make a second copy of the circuit block you want to rename. Insert the second copy right below the original circuit block in your netlist. (2) Rename the new copy NEWNAME. (3) Analyze the circuit. (4) Load all reports. (5) Pop the **Linear Reports (Edit Mode)** dialog by double-clicking in the report you want to modify. (6) Double-click the trace definition you want to rename. (7) Click the **Circuit:** drop down menu (Figure 2) and select NEWNAME. (8) Click the **End Modify** button. Repeat Steps 5 through 7 for all the traces you want to rename. (9) Click the **Display** button to see your modified report. Repeat Steps 5 through 9 for all reports that need fixing. Don't forget to save your modified report(s) and circuit file when you're done.

Q6. Having loaded and analyzed *old.ckt*, and loaded all of the reports in *old.rp2*, I'd like to save *old.ckt* as *new.ckt* but keep working with the reports I defined in *old.rp2*. Can I do this without windowing out of *ARD* and using *File Manager* (*Windows* 3.*x*) or *Explorer* (*Windows* 95) to copy *old.rp2* to *new.rp2*?

A6. Yes. All you have to do is invoke **File | Save As** to save *old.ckt* as *new.ckt*. *ARRL Radio Designer 1.5* automatically copies *old.rp2* to *new.rp2* when you do—even if you haven't yet analyzed *old.ckt* and loaded the reports in *old.rp2*.

Q7. Can I set default report attributes—graph scaling, circuit responses, log versus linear frequency scaling, report title, terminations, and so on—that *ARD* will remember from session to session?

A7. No. Some Report Editor settings hold only as long as you're working with a given circuit file; others (values set in Report Editor's **Default termination:** field, for instance), for an entire *ARD* session (that is, until you exit the program). Report Editor returns to its defaults with every new *ARD* session.

Q8. What about Quick Reporter graphs, and the reports that pop in response to keyboard accelerator commands like Alt+R ("analyze circuit and do rectangular plot")? They all default to reporting MS_{11} or S_{11} plots for the first circuit defined in my netlist. Can I change these defaults—say, make Quick Report default to MS_{21} or S_{21} for its graphs?

A8. No. Although we may sometimes find it useful, Quick Reporter was intended to make *ARD*'s superset program, *Super-Compact*, backward-compatible with an obsolete *Super-Compact* report-file format. (Trivia: The names of report files written in that obsolete form use a .RPT extension.)

Q9. I'm new to circuit modeling and feeling my way into understanding how the S, Y and Z parameters used in *ARD* reports equate to the radio terminology I know. How do I equate *ARD*'s cryptic circuit response identifiers (MS21, RZ11 and so on) to basic circuit characteristics like gain and impedance?

A9. If you want to know a two-port circuit's forward transmission gain (oversimply put, the ratio of its output power to its input power, with impedance factored in), specify MS21. This reports the difference in magnitude (M) of the signal at Port 2 (the output) to the signal at Port 1 (the input). What I find particularly neat about using MS_{21}—that's how it should be written, but *ARD* doesn't display the numbers as subscripts—to report circuit gain is that *ARD* correctly calculates MS_{21} even if the circuit's input and output impedances—the terminations you set in Report Editor—differ. If, for instance, you ask for the MS_{21} of a realistically modeled matching network that transforms 50 Ω to 5 kΩ, you'll see a *negative* MS_{21}—a loss—even though the network steps up the input voltage in transforming 50 Ω to 5 kΩ. Were you to ask *ARD* for the same circuit's MVG3 response—just ratio of the voltage across the circuit's output terminals to the voltage across the input terminals, expressed in dB—you'd see the voltage step-up as gain. Both numbers would be correct in context, but the MVG3 number would be potentially misleading because it would report a voltage gain where there is in fact a power *loss*. (Power loss can be of great practical significance; in circuits that operate at power levels in the kilowatts, unexpectedly high loss can lead one to rapidly seek a fire extinguisher.)

Speaking of loss, *return loss*, expressed in dB, is a highly useful thing to know about a given circuit port because it indicates how closely that port's impedance matches that of its termination or load. An absolutely perfect match results in infinite return loss; that is, all of the applied signal is absorbed by the port and none would reflect back out. If you want *ARD* to tell you a circuit's *input* return loss—an indication of how closely its input impedance matches the impedance of its input termination—ask for MS11. (For the return loss of a two-port circuit's output port, specify MS22.) How much return loss is good enough depends on the application; a return loss of 20 dB or more is a reasonably fine thing.

Relatedly, PS11 (PS22 for the output) indicates the phase difference, in degrees, between the applied signal and the reflected signal. A PS_{11} of 0° indicates a purely resistive input impedance: The reflected signal is inphase with the incident signal. Departures from 0°—leading or lagging phase in the reflected signal—indicate the presence of reactance; that is, that the circuit's impedance is inductive or capacitive and not purely resistive.

That brings us to the subject of a circuit's terminal impedance(s), which *ARD* most directly reports in terms of Z parameters. Determining the impedance seen across a circuit's input or output terminals can be tricky because the load connected to one port can (and usually does) noticeably affect the impedance we measure at the other port(s)—and because, by definition, Z parameters are determined with all circuit ports *unterminated*. To determine a circuit's input impedance with *ARD*, hard-code the circuit's anticipated output load across its output terminals—you can use an IMP or ONE element for a complex load if all you know is your load's impedance in R + *j*X form—and ask for MZ11 (magnitude, in ohms), RZ11 (resistive component, in ohms) and IZ11 (the imaginary [reactive] component, in ohms). To determine a circuit's output impedance, hard-code the appropriate input load into your netlist and ask for the circuit's MZ22, RZ22 and IZ22. The RZ and IZ values returned convey impedance in R + *j*X form.

Hint: The coded-into-the-netlist loads you need to determine port impedance with Z parameters can result in wacky S parameters because they appear in parallel with the terminations set in Report Editor. ASCII is cheap, so I prefer to steer clear of such errors by using multiple netlist copies of the same circuit to achieve these different ends. Wanting to know the MS_{21}, RZ_{11} and IZ_{11} of a matching network, for instance, I'd use two copies of the circuit: One, configured as a two-port network, I'd use to model MS_{21}; the second, set up as a one-port with network's complex load hard-coded across its output port, I'd use for modeling RZ_{11} and IZ_{11}.

You can download circuit and report files exemplifying this two-port/one-port approach, packaged in the archive file *exrf9605.zip*, via *ARRL Radio Designer*'s home page (**http://www.arrl.org/ard/**), a subpage of *ARRLWeb* (**http://www.arrl.org/**), the League's World Wide Web service. The file is also available via the ARRL HQ telephone BBS (860-594-0306) and ARRL FTP site (**ftp://ftp.barc.org/pub/hamradio/arrl/**) and its mirrors. Past editions of Exploring RF are available via the Web, too; just check the *ARD* home page's "Articles About *ARRL Radio Designer*" subpage at **http://www.arrl.org/ard/ardarts.html**.

QST