NATIONAL RADIO ASTRONOMY OBSERVATORY
Charlottesville, Virginia

ELECTRONICS DIVISION TECHNICAL NOTE NO. 150

Title: **CHANGES TO FARANT**

Author(s):   John Granlund

Date: March 20, 1989

CHANGES TO FARANT

John Granlund

March 20, 1989

At this time, the BASIC 5.1 operating system has been installed in the HP 320 and HP 9816 computers at Ivy Road, and the Infotek 5.0 compiler, version F, is available to users of these computers. Both the revised operating system and the revised compiler seem to be free of error and ready for a long period of use without further modification. FARANT 5.1 has been modified to accommodate the associated HP 9122 dual disc drives and HP 2225A Thinkjet printers.

Over the last fraction of a year, three complaints have been received about FARANT from its users. The associated problems are described below, together with the changes to FARANT that have been chosen to solve them. This note is not intended to be a complete list of changes to Dan Fenstermacher's FARANT [1]. Previous changes are noted in [2,3,4].

<u>Is the FARANT Mainline Compiled or Not</u>?

The original mainline was

```
180  OPTION BASE 1
185  COM Zo,F,Dat(51,18),INTEGER Nogo,Count
190  Farstart
195  END
```

When COMPILEd with SOURCE OFF, it read

```
185  COM Zo,F,Dat(51,18),INTEGER Nogo,Count
190  Farstart
195  end
```

and when UNCOMPILEd, it then read

```
185  COM Zo,F,Dat(51,18),INTEGER Nogo,Count
190  Farstart
195  END
```

Rerunning FARANT after this UNCOMPILE operation, with the default OPTION BASE 0 now in effect, produced errors in Dat(*). To correct this difficulty, the mainline has been changed to read

```
175  GOTO 180                    !PREVENTS IMPROPER UNCOMPILE
180  OPTION BASE 1
185  COM Zo,F,Dat(51,18),INTEGER Nogo,Count
190  Farstart
195  END
```

When COMPILEd with SOURCE OFF, this reads

```
175  *
185  COM Zo,F,Dat(51,18),INTEGER Nogo,Count
190  Farstart
195  end
```

Since this COMPILEd version cannot be UNCOMPILEd, the difficulty with the mainline has been removed. None of FARANT's subprograms can be UNCOMPILEd either, once they have been COMPILEd with SOURCE OFF.

The Gains Calculated by Nperformance

Following pass parameter Gtype, SUB Nperformance can be instructed to calculate one of four gains for the two-port described by X(*), which is also passed to Nperformance. Maximum available gain, called for by Gtype = 4, does not exist when X(*) is only conditionally stable, but Nperformance already deals adequately with this potential difficulty by printing a message to the user and calculating maximum <u>stable</u> gain instead.

The other three -- decibel -- gains are calculated as

$$Gain = 10 * LGT(N/D)$$

after N and D have been separately evaluated, checked for positiveness, and revised if not positive. If N is not positive, it is set equal to 10^{-49}; if D is not positive, it is set to 10^{-99}. This results in the following bogus Gains:

$N \le 0$	Gain $\approx$ -490 dB
$D \le 0$	Gain $\approx$ +990 dB
$N \le 0$ and $D \le 0$	Gain $\equiv$ +500 dB

The program is then not stopped by an error message for the attempted calculation of the log of a negative number, but the user can make himself aware of the difficulty by looking for unexpectedly large or small values of Gain in his results. Gain is stored by Nperformance in Dat(Count,15), and it is also passed back to the calling subprogram.

The current, changed version of Nperformance has been described. Previously, N and D for available gain, Gtype = 3, were not checked for positiveness, and a negative real part of output impedance made D negative, resulting in the error message noted above. In the revision process, it was noted that if both non-positive N and non-positive D were reset to 10^{-99}, the resulting Gain = 0 dB might not carry a strong enough message to the user. So the case of non-positive N has been revised to reset N to 10^{-49}.

2

Calling Ntrans with No Noise

A user loaded his two-port matrix X(*) with signal parameters, but did not load noise parameters. This left Nset = X(5,2) = 0 and all of the elements in the sixth row of X(*) zero. Then he called SUB Flop with X(*). Flop called Ntrans with X(*), and an error resulted: Ntrans was not designed to deal with noiseless two-ports.

In seeking a solution to this problem, it was determined that Ntrans is called by no less than 11 FARANT subprograms, but prior to the fix about to be described, only two of these subprograms did not avoid the call of Ntrans with a two-port descriptor having no noise: SUB Flop and SUB Web in FET_LIB. It was further determined that Flop would properly calculate signal parameters without error messages if its call to Ntrans were omitted, and Web would properly make its plot -- with errors in the locations of Z_{opt} and Z^*_{out} with $Z_s = Z_{opt}$ -- with no call to Ntrans.

The solution to the problem that has been implemented has two parts: SUB Flop and SUB Web have been revised to avoid calling Ntrans without noise, and SUB Ntrans has been revised to avoid making any calculations on noiseless two-ports.

REFERENCES

[1]   Dan L. Fenstermacher, "A Computer-Aided Analysis Routine Including Optimization for Microwave Circuits and Their Noise," NRAO Electronics Division Internal Report No. 217, July 1981.

[2]   John Granlund, "FARANT on the HP 9816 Computer," NRAO Electronics Division Internal Report No. 250, July 1984.

[3]   John Granlund, "FARANT 3.0 on the HP 9816 Computer," NRAO Electronics Division Technical Note No. 124, November 1984.

[4]   John Granlund, "FARANT 5.0 on the HP 320 and HP 9816 Computers," NRAO Electronics Division Technical Note No. 145, May 1988.