

SPICE2G6

RELEASE NOTICE 2-01

Published by:

Acorn Computers Limited
Fulbourn Road
Cherry Hinton
Cambridge
CB1 4JN

© Acorn Computers Limited 1986

Part No 0449,041
Issue No 1
August 1986

1. Release Summary

The following are the files included with this release:

File names	Description
DISC 1	
guide-txt	SPICE2G6 Reference Manual
install-cmd	Installation Command File
setup-f77	One of 8 SPICE2G6 source modules
test-log	Expected output of test program test-sim
readin-f77	One of 8 SPICE2G6 source modules
root-f77	One of 8 SPICE2G6 source modules
DISC 2	
acan-f77	One of 8 SPICE2G6 source modules
dcop-f77	One of 8 SPICE2G6 source modules
errchk-f77	One of 8 SPICE2G6 source modules
install-cmd	Installation Command File
makesp-cmd	Command File to link 8 modules and assembler module into executable SPICE2G6
misc-asm	
benchmk-sim	Berkeley Test file
test-sim	Test file
dctran-f77	One of 8 SPICE2G6 source modules
ovtpvt-f77	One of 8 SPICE2G6 source modules
DISC 3	
Spice1-rif	First part of executable version of SPICE2G6
Spice2-rif	Second part of executable version of SPICE2G6
Loadsp-cmd	Command file to load spice2g6 from DFS disc

2. Installation

2.1 Installing the program

To install the executable program, SPICE2G6-rif on DISC 3, from within Panos simply copy the two parts of the file onto a Winchester disc or onto a file server and then configure your system for use (see below). For example:

```
-> Copy dfs::0.spice1-rif,dfs::2.spice2-rif
      -to <fs>:$.PanosLib.Spice2g6-rif
```

where <fs> stands for adfs or nfs.

If re-installing append `-force`.

2.1.1 Initialising for use with DFS floppy discs

It is not necessary to install SPICE2G6 when running from a DFS floppy disc. Simply use DISC 3 as provided (but take a back-up copy first). However it does need to be initialised each time the system is switched on or reset.

If running from DFS or copying the test files from the release disc, a very small amount of configuration is necessary before running SPICE2G6. Two file extensions need to be set, and this can be done by typing:

```
-> .Set File$dfs:-sim u.-
-> .Set File$dfs:-log v.-
```

The best way of achieving this is by adding these commands to the *Panos* start-up command file, or by creating a new start-up file, say *Spice-cmd*.

Panos version 1.4 (or later) is necessary to run SPICE2G6 from a DFS disc, as the executable image needs to be loaded into the store filing system. A command file *loadsp* has been provided to do this. Having started up Panos from your DFS Start-Up disc, place DISC 3 in the top drive and type:

```
-> Loadsp
```

The program is now copied into the store filing system and is ready for

1.1 Documentation

Documentation supplied with this release: SPICE2G6 User's Guide formatted for 132 column, 66 line line-printer paper, both hardcopy and on disc.

1.2 Pre-requisites

Memory size : 1 Mb

Panos version : 1.4

use.

2.2 Note for Econet users

Before installing any software onto the Econet, you must have logged-on as System Manager, as the installation command file copies files onto the '\$' level.

3. Running SPICE2G6 under Panos

The command to run SPICE2G6 under Panos is:

```
-> SPICE2G6 <INfile>-sim 50=<OUTfile>-log
```

The first file given is the input file, which will send its output to the default output device (usually the screen). To specify a file for the output, type *50=* immediately before the output file name.

For example:

```
-> SPICE2G6 test-sim  
-> SPICE2G6 test-sim 50=test-log
```

3.1 Testing

After SPICE2G6 has been installed and/or built, you can test the program using the file *test-sim* supplied on disc 2 as input. The output should be identical to the supplied file *test-log* (except for times and dates).

4. Installing and re-building the source

Those using DFS only cannot re-build SPICE2G6 from source. Instructions are given for NFS and ADFS only.

For those who wish to re-build SPICE2G6, the following steps will install the source and then the command file *makesp* will carry out the procedure. Note that the 32000 FORTRAN 77 compiler, assembler, and linker are necessary pre-requisites.

(1) Place a copy of distribution DISC 1 in the top drive (if present).

(2) Type:

```
-> dfs:install <fs>
```

where <fs> is either ADFS or NFS.

(3) Replace DISC 1 with DISC 2 and repeat the above command.

(4) To re-build SPICE2G6:

(5) Enter the 'spice' directory and type:

```
-> makesp
```

and the command file will compile and link the necessary files. This takes some time; about 40 minutes on the ADFS. Remember that the global string variable *link\$lib:f77* must contain the path name of the FORTRAN 77 library, and *link\$lib_list* must also contain *F77*, as is normal for all linking of FORTRAN modules.

If SPICE2G6 has already been installed, append *-force* to your command and any existing files of the same name will be overwritten. The keyword *-abandon* will cause the command file to abort if any error has occurred during the process.

For example:

```
-> dfs:install ADFS -force
```

During installation and building of the source, files were confined to the directory *Spice*. You may prefer to place the executable program in another directory.

5. Release Details

The Berkeley distribution source of SPICE2G6 was split into sections:

root : top to SUBROUTINE READIN (0%-18% of file)
readin: SUBROUTINE READIN to SUBROUTINE ERRCHK (18%-29% of file)
errchk: SUBROUTINE ERRCHK to SUBROUTINE SETUP (29%-43% of file)
setup : SUBROUTINE SETUP to SUBROUTINE DCTRAN (43%-47% of file)
dctran: SUBROUTINE DCTRAN to SUBROUTINE DCOP (47%-75% of file)
dcop : SUBROUTINE DCOP to SUBROUTINE ACAN (75%-82% of file)
acan : SUBROUTINE ACAN to SUBROUTINE OVTPVT (82%-94% of file)
ovtpvt: SUBROUTINE OVTPVT to end (94%-100% of file)

5.1 Compiling problems

Incompatibility in LOCF calls in 'root'. Two are real parameters, two complex and the rest integers:

SUBROUTINE USHIFT in 'readin' has an INTEGER*2 declaration.
SUBROUTINE PHEADR in 'dctran' has an INTEGER*2 declaration.
SUBROUTINE DISTO in 'acan' has lots of type illegalities with complex number arithmetic and equivalence.

LOCF only applies to integer variables. Two new INTEGER FUNCTIONS have been added; LOCFR(V) for a real LOCF, and LOFC(C) for complex LOCF. Both are simple equivalence functions. USHIFT takes lower-case letters and transforms them to upper case. This has to been modified in 'readin'. We believe that the PHEADR is for the binary data file output option:

```
SPICE2G6 -r datafile < file-sim > file-log
```

and can therefore be commented out, as long as IPOSTP=0 at all times (no postprocessor option). PHEADR has been replaced with a message saying post-processor not implemented in this version. The subroutine DISTO causes a number of problems with complex number manipulation. In particular, ComplexNo*0.5D0 is illegal, and should be replaced by ComplexNo*(0.5,0.0). A further problem with real-complex conversion was found with real variables which had not been converted to complex by CMPLX(realno). About 80 changes of this type were necessary.

5.2 Changes made to this SPICE2G6 conversion

All comments are identified by C^

COMMON /BLOCK/ VALUE(200000) changed to VALUE(50000) everywhere.

root:

IOFILE changed to 50 (was 6).

MAXMEM set to 2*50000 (was 2*200000) for 50000.

double-precision words:

Two new INTEGER FUNCTIONS added; LOCFR(V) for a real LOCF, and LOCFC(C) for complex LOCF. Both are simple equivalence functions.

The calls (two of each type) are also edited in the SETMEM section.

SUBROUTINE SECOND commented out (will be assembler).

COMMON /BLOCK/ VALUE(200000) changed to VALUE(50000).

readin:

READ(5,6,.. changed to READ(1,6,.. (input unit number).

SUBROUTINE USHIFT modified.

COMMON /BLOCK/ VALUE(200000) changed to VALUE(50000).

errchk:

COMMON /BLOCK/ VALUE(200000) changed to VALUE(50000).

setup:

COMMON /BLOCK/ VALUE(200000) changed to VALUE(50000).

dctran:

Subroutine PHEADR commented out and replaced by a dummy routine.

COMMON /BLOCK/ VALUE(200000) changed to VALUE(50000).

dcop:

COMMON /BLOCK/ VALUE(200000) changed to VALUE(50000).

acan:

SUBROUTINE DISTO modified.

COMMON /BLOCK/ VALUE(200000) changed to VALUE(50000).

ovtpvt:

COMMON /BLOCK/ VALUE(200000) changed to VALUE(50000).