GSP

("Graphics to Spice")

Netlist Compiler

User Manual & Description

OVERVIEW

GSP is a tool which generates a Berkeley SPICE netlist from a SCALD format drawing, like that produced by the GEX schematic editor, or the Cadence/Valid GED editor.

GSP is extremely fast.

It requires no command files or knowledge of the Unix operating system or of its associated text editors. However, it has the capability of accepting a command file for paths to libraries, if these are located somewhere other than under a common root.

GSP has an inherent understanding of SPICE2G6 and Spice3D2 primitives, thus making unnecessary the duplication of body drawings and spice models for generic parts totally described by one primitive. It means that libraries of resistors, capacitors, transistors, diodes etc, need only contain physical data files for use by PCB layout tools etc, the part being defined on the schematic by a common body from the existing spice library, and by body properties.

It works directly from the files created by GEX, and produces a netlist which is completely compatible with SPICE2G6 or Spice3d2. A cross-reference file ("spice.xref") is also produced, under the drawing directory, which contains SPICE node numbers and their corresponding signal-names as defined within GEX.

Hierarchical designs are handled by recursively traversing the hierarchy, and creating a .SUBCKT netlist for each level. Up to 32 levels can be compiled..

Multiple drawing pages on the same level are also supported.

GSP offers certain enhancements currently unavailable on many netlisters, as follows:

- **Interpretation of polynomial expressions for SPICE2G6**
  GSP correctly interprets polynomial expressions describing the transfer functions of non-linear dependent sources and thus permits high-level modelling of such functions as A to D and D to A converters, log and power-law amplifiers, modulators etc.

- **Interpretation of transfer function equations for Spice3D2**

GSP correctly interprets mathematical equations describing the transfer functions of non-linear dependent sources and thus permits high-level modelling of such functions as A to D and D to A converters, log and power-law amplifiers, modulators etc.

- **Compilation of multi-winding transformers**
  Inductors may be coupled on the schematic, without the need for special bodies, in any number and any order. The correct coupling coefficient is calculated from the turns ratio and inductances of coupled windings.

- **Automatic inclusion of semiconductor model parameters**
  A search is performed in all libraries through the 'models' files and all necessary model parameters are included at the head of the netlist.

- **Spice3D2 advanced primitives**
  In addition to all the standard SPICE2G6 primitives, the advanced primitives of Spice3D2 are included, such as switches, GaAs FET's etc.

OPERATION

Environment Variables

GSP expects to see the following environment variables:

VSPICE_HOME
The location of the GEX, Vspice3 and GSP executables

VSPICE_LIB
The location of all libraries used by GEX and GSP

Invocation

GSP is invoked in the following ways:

GSP <name>

GSP will generate a netlist in file called 'name.ckt' and a cross-reference of node numbers and GEX node names in a file 'name/spice.xref', where 'name' is the directory containing the drawing files.

A dummy ".tran" command, together with a dummy ".print" statement are inserted in the netlist after the title line. The assumption is that the correct simulator commands and print/plot statements will be inserted from Vspice3, or by hand, by the user.

The file "name" must be a standard GEX drawing having either a spice_cn.1.1 or logic_cn.1.1 connectivity file.

GSP will default to a search of the current directory, and under the root $VSPICE_LIB for all libraries described in the above connectivity file. It also expects

to find a file named 'models' directly under each library, or in the current directory, containing semiconductor model definitions, as described below under 'Models'.

GSP <name> <path/to/root/of/libraries>
As above, but the search path will now assume that libraries also exist under the root given in the second argument.

GSP <name> -f <filename>
If libraries are widely scattered around the file systems, the paths to them may be included in a file, one per line, containing no other information. An example of a complete file is as follows:

/usr4/lib
/usr2/lib/analog
/usr6/natsemi/linear/opamps
/usr/local/analog/lib/signetics/linear/opamps

GSP -s <any of above three formats>
If the first argument is -s, GSP assumes that a complete SPICE netlist is required, and generates the file 'name.netlist' instead of 'name.ckt' and the cross-reference file 'name/spice.xref'.

As a convenience, if Vspice3 is not going to be used, and a file 'name.header' exists, its contents will be inserted at the head of the netlist. The file should contain the SPICE analysis cards for simulating the netlist, together with print/plot directives. All .PLOT and .PRINT commands should in the format:

.PLOT DC/AC/TRAN/etc V(%NET_NAME1) V(%NET_NAME2).....

where *NET_NAME1, NET_NAME2 etc* are GEX net names.

GSP will substitute SPICE node numbers from 'name/spice.xref' for the net names.

The -s should be followed by any of the formats used for SPICE typed in exactly the same way, e.g: **GSP -s videoamp -f path.cmd**

COMPILATION RULES
GSP observes the following procedures when compiling:

PRIMITIVE DEVICES

If a component has any of the following names, it is assumed to be a SPICE primitive, whose characteristics are known to GSP, and no library search is performed:

Cadence/Valid GED-compatible components:

Resistor:	*'resistor'*
Capacitor:	*'capacitor'*

Inductor:	*'inductor'*
Transmission Line:	*'transmission'*
Voltage source:	*'vsource'*
Current source:	*'isource'*
Voltage-controlled voltage source:	*'vsourcev'*
Voltage-controlled current source:	*'isourcev'*
Current-controlled voltage source:	*'vsourcei'*
Current-controlled current source:	*'isourcei'*
Ground symbol:	*'ground' 'gnd'*
Bipolar transistor:	*'npn' 'pnp'*
Mosfet:	*'pmos' 'nmos'*
Jfet:	*'njfet' 'pjfet'*
Diode:	*'diode'*

Alternative component names:

Resistor:	*'res' 'r'*
Capacitor:	*'cap' 'c'*
Inductor:	*'l'*
Transmission Line:	*'txline'*
Voltage source:	*'vsource' 'vsrc'*
Current source:	*'isource' 'isrc'*
Voltage-controlled voltage source:	*'vsrcv'*
Voltage-controlled current source:	*'isrcv'*
Current-controlled voltage source:	*'vsrci'*
Current-controlled current source:	*'isrci'*
Bipolar transistor:	*'tnpn' 'tpnp'*
Mosfet:	*'pmf' 'nmf'*
Jfet:	*'pfet' 'nfet'*
Diode:	*'d'*

Spice3D2 additional components:

GaAs FET:	*'nmf' 'pmf'*
Voltage-controlled switch:	*'vswitch' 'vsw'*
Current-controlled switch:	*'iswitch' 'isw'*
Distributed (Lossy) RC Line:	*'rcline'*

The absence of a 'VALUE' property on a resistor, capacitor, or inductor, the absence of a 'VALUE' or 'PARAMS' property on a voltage or current source, or the absence of a 'MODEL' property on a semiconductor device, switch or rc line is considered a fatal error, and GSP flags this and aborts.

SEMICONDUCTOR DEVICES

All semiconductor devices must have the property 'MODEL=XYZ', where 'XYZ' is a device name, as it appears in the 'models' file. If a component is hierarchical (e.g power device, where the spice.1.1 drawing contains lead inductances etc) then the 'MODEL' property should be on the primitive in the spice.1.1 drawing, NOT on the

top level body. The additional placing of a 'MODEL' property on the body will cause two identical .MODEL entries to be placed in the output file.

If any component has a 'MODEL' property, it is assumed to be a semiconductor device, switch or rcline. If there is no spice or logic_cn.1.1, or no corresponding entry in the 'models' file, this is reported as a warning, but GSP completes the netlist.

MODELS

GSP is compatible with all SPICE macro models generated in GEX, whether hierarchical or not, with the proviso that there are only two levels of hierarchy. The top level may be a logic.1.1 or a spice.1.1 drawing, containing bodies, each of which has a no more than one spice.1.1 or logic.1.1 drawing beneath it.

A single-level drawing may be either a logic.1.1 or a spice.1.1 drawing, provided that it contains only the primitives named above.

All semiconductor, switch and rc line SPICE model descriptions should be in a file called 'models', situated either in the user's local directory, or directly under the corresponding library subdirectory. The file should contain semiconductor .MODEL cards, one per line, with each continuation line starting with a '+' as in the following example:

.MODEL BC108 NPN BF=120 IKF=30E-3 ISC=1E-8
+TF=1.5N TR=6N RB=100 VAF=30 CJE=10PF VJE=0.6 CJC=2PF

Model files may also be in the local directory, which is searched by default, or $VSPICE_LIB may contain a master model file.

The search always follows the same order (i.e adt/lib, local directory, user-defined search paths) so if speed is critical, some advantage may be gained by putting a hard or soft link from $VSPICE_LIB to a master file located in a convenient directory.

MULTI-WINDING TRANSFORMERS

A transformer is constructed in GEX in the following manner:

- All windings are inductors, whose inductance is specified by the property *VALUE=n*
- All windings of a transformer must have the property 'LOCATION=Lx', where Lx is a name unique to that transformer, and 'x' is not a digit.
- One winding, conveniently the primary, must have the property *COUPLED=Lp Lq Lr Ls*
- Each winding must include the property *PARMS=n* where 'n' is the number of turns on the winding

Lp Lq Lr and Ls are the location properties of the secondary windings, specified as above, and separated by spaces. The 'PARMS' property specifies the coupling coefficients, again separated by spaces, each expressed as a number less than unity, (optionally floating point) of all windings with respect to one another according to the following rule, for a five-winding transformer:

Lx to Lp = a
Lx to Lq = b
Lx to Lr = c
Lx to Ls = d
Lp to Lq = e
Lp to Lr = f
Lp to Ls = g
Lq to Lr = h
Lq to Ls = i
Lr to Ls = j

GSP will translate all inductor location properties to a linearly-numbered set of SPICE device names, and produce coupling statements in the output file of the following form:

K1   L1 L2 a
K2   L1 L3 b
K3   L1 L4 c
K4   L1 L5 d
K5   L2 L3 e
K6   L2 L4 f
K7   L2 L5 g
K8   L3 L4 h
K9   L3 L5 i
K10 L4 L5 j

The coupling coefficients a-j are calculated as follows:
K12 = N2/N1 * sqrt(L1/L2)
This occasionally leads to a coefficient greater than 1.0, in which case, we recalculate it as
K21 = N1/N2 * sqrt(L2/L1)
where **N2, N1** are the numbers of turns on each winding, and **L1, L2** are their inductances.

SOURCES WITH EQUATION-DEFINED TRANSFER FUNCTIONS (Spice3 only)

Non-linear dependent sources, both current and voltage controlled, are constructed from the SPICE primitives 'vsource' and 'isource', 'vsrc' and 'isrc', or any compatible name. They are defined in GEX in the following manner:

- Each output source must have a property 'LOCATION=x' where 'x' is a name, whose first character MUST be 'B'.
  Spice3 makes no distinction as to whether the source is voltage or current controlled, the equation being the key.
  The equation must contain a combination of symbolic voltage source names or net names and the following mathematical operators:

+   -   *   /   ^
abs   acos   acosh   asin   asinh

atan	atanh	cos	cosh	exp
ln	log	sin	sinh	sqrt
tan				

GSP does not parse the equation, since the simulator will do this at run time, but merely substitutes node numbers for symbolic node names, and voltage source names for their symbolic equivalent. For example, if it is desired to make a current a function of the 12th power of the difference of two voltages, a current source would be added in GEX, and given a location property of, perhaps, 'BDIFF'. It would then be given a PARAMS property of:

 (v(vplus) - v(vminus))^12

where 'vplus' and 'vminus' are net names assigned with the **NAME NET** command in GEX.

In the netlist, this might appear as:

 B1 1 0 I=(V(3) - V(4))^12

In a case where a voltage is required to be, say, a function of the product of two currents, a voltage source would be given the location property 'BMULT', and the PARAMS property:

 I(VCURRENT1) * I(VCURRENT2)

This might appear in the netlist as:

 B2 2 0 V=I(V3) * I(V9)

Note that net names MUST begin with a character, not a digit, since GSP assumes that numeric entries in the equation are node numbers, supplied by the user, and does nothing to them.

SOURCES WITH POLYNOMIAL-DEFINED TRANSFER FUNCTIONS (Spice2 only)

Non-linear dependent sources, both current and voltage controlled, are constructed from the SPICE primitives 'vsource' and 'isource', 'vsrc' and 'isrc', or any compatible name. They are defined in GEX in the following manner:

- Each output source must have a property 'LOCATION=x' where 'x' is a name, whose first character determines the type of source, acccording to the following table:

    E = Voltage-controlled voltage source
    G = Voltage-controlled current source

    F = Current-controlled current source
    H = Current-controlled voltage source

No check is performed for this latter, since it is not possible to predict the type of source required, and if the polynomial is incompatible, no name substitution is performed.

- Voltage-controlled sources are associated with the netnames of the wires corresponding to the controlling voltage(s), so these wires should be given net names of no more than eight characters.

- Current-controlled sources are associated with current-measuring voltage sources, each of which has the VALUE property '0' or 'DC 0', and passes the controlling current. Each current-measuring voltage source must have the property 'LOCATION=Vn', which must be identical with that used in the polynomial definition. Again, if the names do not correspond, no substitution is performed.

- Each voltage-controlled output source must have a polynomial definition, defined by the property
  PARMS=POLY(n) s1 s0 t1 t0 u1 u0...z1 z0 p1 p2...pn

  The factor 'n' states the number of voltages (net name pairs) of which the output is a function, and s1 s0 ... z1 z0 are pairs of net names defining the controlling voltage(s). The net names s1, t1 etc are the positive nodes, and s0, t0 etc are the negative nodes. If any voltage is referred to ground, the net name corresponding to the ground node is specified as '0'. If 'n' is incorrect, no substitution is performed.

  The factors p1 ... pn are the coefficients of the polynomial defining the transfer function, as outlined in the SPICE 2G.6 manual.

- Each current-controlled output source must have a polynomial definition, defined by the property
  PARMS=POLY(n) V1 V2 V3 ...Vn p1 p2...pn
  where V1 ... Vn are the location properties of the current-measuring zero-valued voltage sources.

  The factor 'n' states the number of currents (zero-valued voltage sources) of which the output is a function.

  The factors p1 ... pn are the coefficients of the polynomial defining the transfer function, as outlined in the SPICE 2G.6 manual.

OUTPUT FILES

The output files generated are 'name.ckt' and 'name/spice.xref', where 'name' is the drawing name.

The .ckt file is a standard Berkeley netlist, but contains a dummy simulation command (".tran") and a dummy print command (".print"). These may either be

edited by hand, or can be replaced by the correct commands from the SETUP menu in Vspice3, and the file is then renamed, 'name.netlist'.

The xref file has two sections, both of which is a two column list. The first is a list of symbolic names assigned in GEX, and the corresponding numbers which appear in the SPICE netlist. The second is a list of all the components of the drawing, and their corresponding PATH properties. The file is used by vspice3 to relate mouse-selected items on the drawing to the netlist.

BUGS and OMISSIONS

1. Cross-reference only includes top level net names

    Although GSP has at its disposal a full cross-reference of all signal names in the hierarchy, these are not included because each body may be a library component, to which the user may not have write permission. .

3. Only thirty-two levels of hierarchy are compiled

    This will probably not change.

4. No facility for subcircuit definitions to be read from a file.

    This may be implemented, using the Mentor Graphics nomenclature for the pin names of the subcircuit. A body needs to be created with pins named N1, N2.. Nn, which will be related in numerical order to the subcircuit's pin numbers.
    GSP will then assign circuit node numbers to the body in this order.