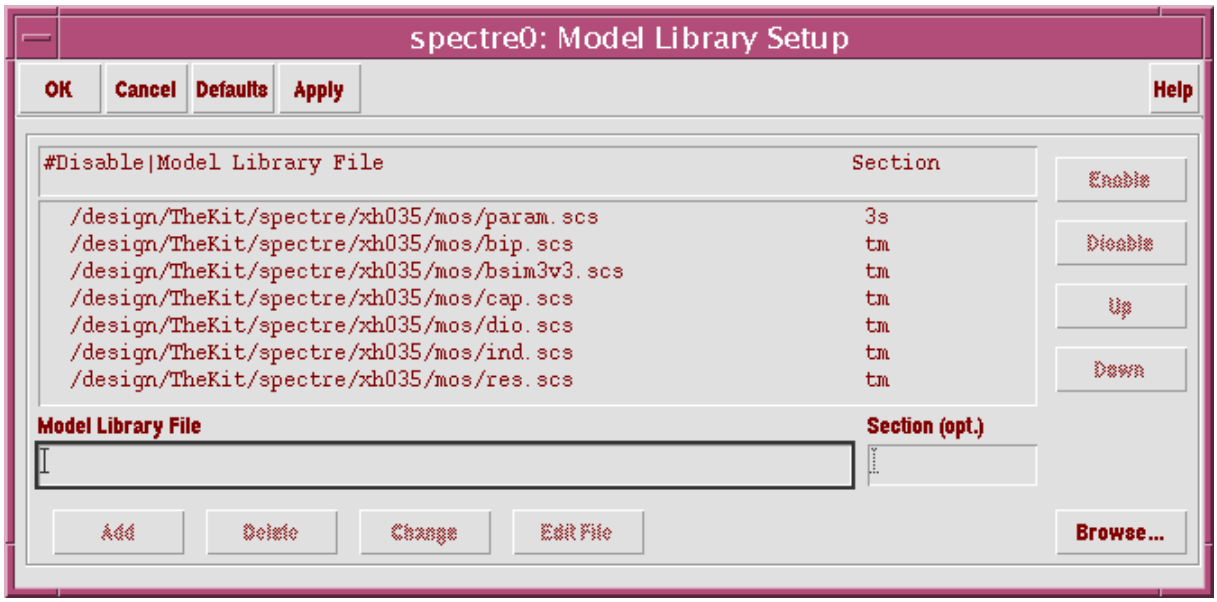
OCC Flow for simulation

How to perform an OCC (Operating Condition Check) – simulation

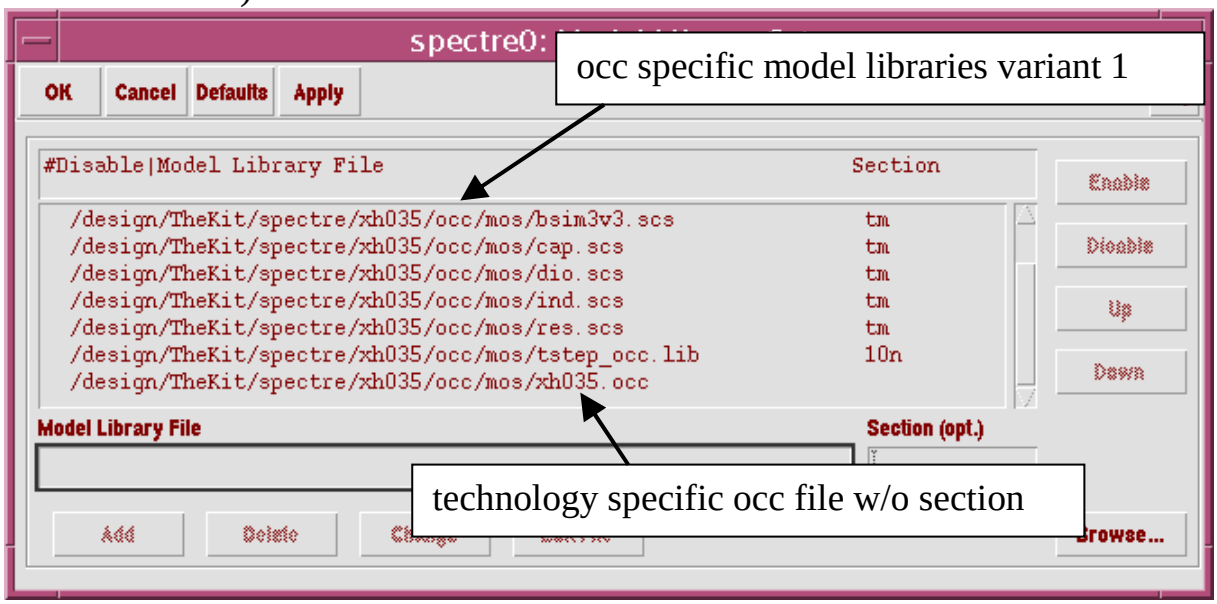
1. Spectre

- Invoke “Cadence Analog Design Environment
- Choosing “Simulator/Directory-..” -> Spectre as preferred simulator
- Choosing “Setup” -> Model Libraries …” Picture below shows an example with the normal model path.



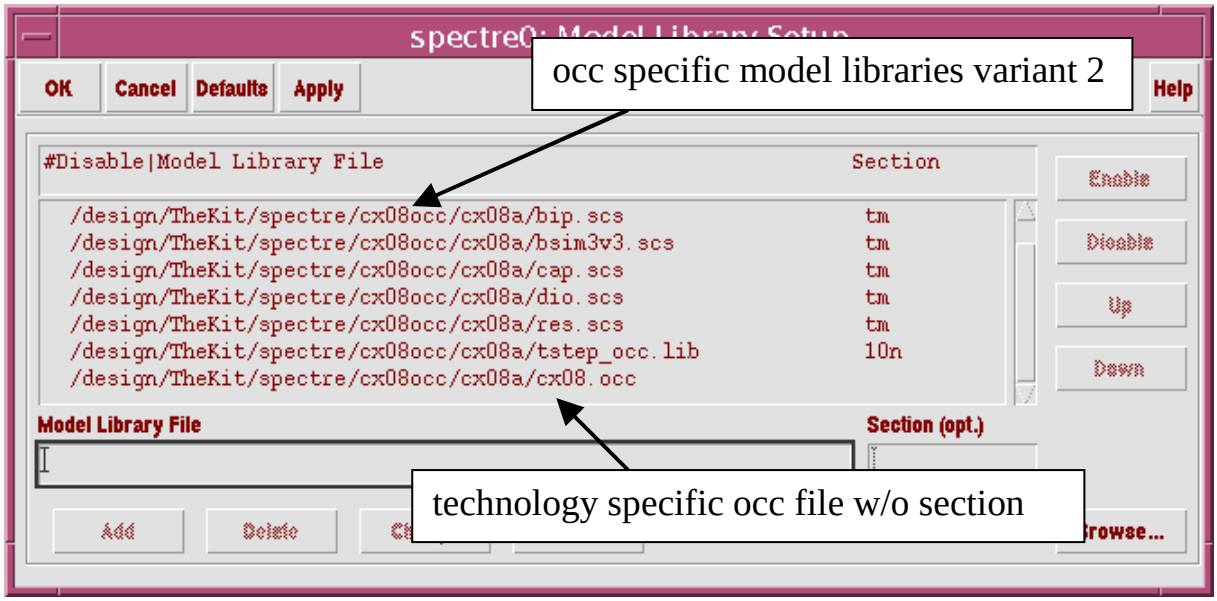
Depend on the model directory structure there are two different methods to declare the model paths. See table below which method must be used for the specific technology.

In the first method the “occ” subpath is included in the model path and the two model libraries “tstep_occ.lib” and tech.occ (in the example below “xh035.occ”) must be added to the model libraries.



The first five lines show the normal model libraries with the “occ” extensions. In the tstep_occ library is defined a simulation variable which influence the display of the occ event. The library *.occ (last line in the picture) is the library with the VerilogA modules. This library must not have a section entry

The second method to define the model paths shows the picture below. In this case a common directory is included (in the picture “cx08occ”) before the technology name.



The table shows the assignment of the technologies to the path modification variants

Techmology	Path w/o OCC	Path modification	Path-variant	Note
xc10		---		n.a.
xi10		---		n.a.
xdm10		---		n.a.
xdh10		---		n.a.
cx08a	…/cx08a/…	…/cx08occ/cx08a/…	2	
cx08h	…/cx08h/…	…/cx08occ/cx08h/…	2	
cx08n	…/cx08n/…	…/cx08occ/cx08n/…	2	
cx06		---		n.a.
xc06	…/xc06/…	…/xc06occ/xc06/…	2	
xc06pin	…/xc06pin…	…/xc06occ/xc06pin/…	2	
xb06	…/xb06/…	…/xb06occ/xb06/…	2	
xb06pin	…/xb06pin/…	…/xb06occ/xb06pin/…	2	
xt06	…/xt06/…	…/xt06occ/xt06	2	
xh035	…/xh035/…	…/xh035/occ/…	1	
xc018	…/xc018/…	…/xc018/occ/…	1	
xh018				n.a.

Start transient simulation within the "Cadence Analog Design Environment" windows

- evaluate simulation results of the created "occ.err" log file which is located in the simulation project directory eg. "/simulation/…/occ.err"
- the "occ.err" text file contains all violations regarding voltage limitations ("OCA") of each primitive device with the associated time step related to operating conditions define within the process specification

Device	OC-Para	Event	Limit	Time	Instance
cpoly	Vpm	leaving	OCA (> 5.5 V)	at 0.000000e+00	I3.C0.occ_m1
pmos4	VDpsub	leaving	OCA (> 5.5 V)	at 0.000000e+00	I2.M18.occ_m1
pghv	VDB	leaving	OCA (> 0.5 V)	at 1.590018e-06	I3.M3.occ_m1
cpoly	Vpm	entering	OCA (-5.5 V .. 5.5 V)	at 3.679900e-06	I3.C0. occ_m1
nhv	VSB	leaving	OCA (> 11.0 V)	at 4.654617e-06	I3.M1.occ_m1
nhv	VSB	leaving	OCA (> 11.0 V)	at 4.654617e-06	I3.M2.occ_m1
pghv	VDB	entering	OCA (-25.0 V .. 0.5 V)	at 6.545138e-06	I3.M3.occ_m1
nhv	VSB	entering	OCA (-0.5 V .. 11.0 V)	at 2.006150e-03	I3.M1.occ_m1
nhv	VSB	entering	OCA (-0.5 V .. 11.0 V)	at 2.006150e-03	I3.M2.occ_m1
cpoly	Vpm	leaving	OCA (> 5.5 V)	at 2.009185e-03	I3.C0.occ_m1
pghv	VDB	leaving	OCA (> 0.5 V)	at 5.001485e-03	I3.M3.occ_m1
cpoly	Vpm	entering	OCA (-5.5 V .. 5.5 V)	at 5.003387e-03	I3.C0.occ_m1
nhv	VSB	leaving	OCA (> 11.0 V)	at 5.004651e-03	I3.M1.occ_m1
nhv	VSB	leaving	OCA (> 11.0 V)	at 5.004651e-03	I3.M2.occ_m1
pghv	VDB	entering	OCA (-25.0 V .. 0.5 V)	at 5.006209e-03	I3.M3.occ_m1

- the "occ.err" file indicates "**leaving**" and "**entering**" of the OCA related to the operating conditions of all terminal voltages defined in the process specification
- but the circuit designer has the complete **"responsibility" to evaluate and to interpret** the existing events / violations
- the "OCC" simulation method is only a kind of terminal voltage monitor of primitive devices within a circuitry
- experiences show, problem devices are mostly not devices which are directly connected with the power supply rails but primitive devices of intermediate circuit nodes are more jeopardized (eg. compensation capacitors or capacitive coupling)

2. Eldo and HSpice

Preparing Netlist

Change the model path to the occ models. The rules how to change to the occ models are the same as described above in the Spectre section. Next add the library tstep_occ.lib with the .lib command and <tech>.occ file with the .inc command to the net list (see example).

In the examples <sim> stands for “eldo” or “hSpiceS”

Example for Variant 1

Change model path from

```
.lib    “…/<sim>/xh035/mos/bsim3v3.lib” tm
…
```

to

```
.lib    "…/<sim>/xh035/occ/mos/bsim3v3.lib" tm
…
```

Add new statements

```
.lib    "…/<sim>/xh035/occ/mos/tstep_occ.lib" 10n
.inc    "…/<sim>/xh035/occ/mos/xh035.occ"
```

Example for Variant 2

Change model path from

```
.lib    “…/<sim>/xc06/bsim3v3.lib” tm
…
```

to

```
.lib    "…/<sim>/ xc06occ/xc06/bsim3v3.lib" tm
…
```

Add new statements

```
.lib    "…/<sim>/ xc06occ/xc06/tstep_occ.lib" 10n
.inc    "…/<sim>/ xc06occ/xc06/xh035.occ"
```

The simulation run creates the log file “occ.err” in the run directory. The content of this file is the same as described in the spectre section above. The format is lightly different depend on the used simulator.