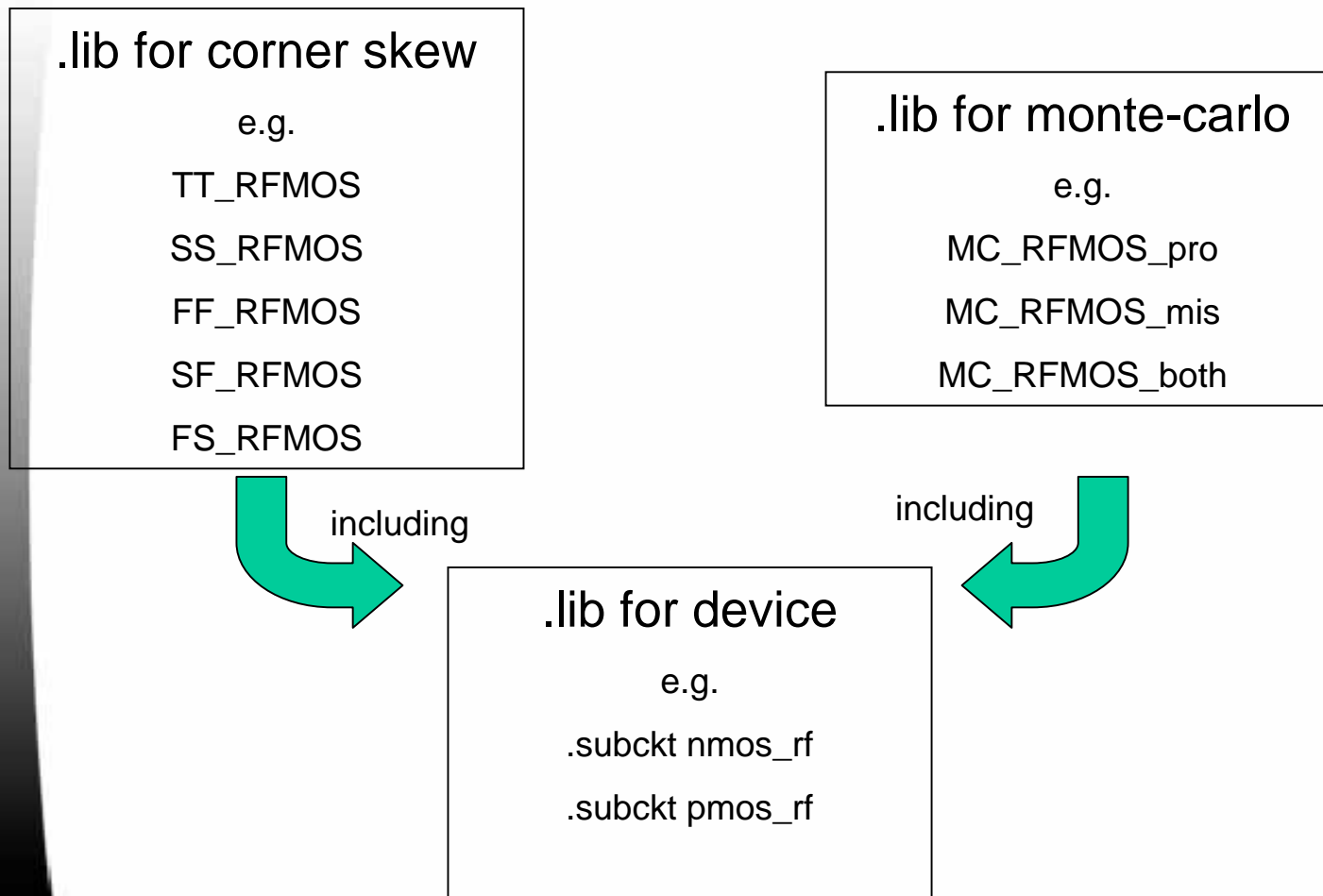


Model cards structure and usage



Original structure for RF CMOS model



Discrepancy: Monte-Carlo all the MOS in a circuit when calling .lib MC_RFMOS_mis, in spite they are not in the mismatch pair.

Corrected Structure for RF CMOS model

.lib for corner skew

TT_RFMOS

SS_RFMOS

FF_RFMOS

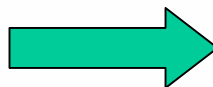
SF_RFMOS

FS_RFMOS

.lib for monte-carlo
(process)

MC_RFMOS

including



.lib for normal device

.subckt nmos_rf

.subckt pmos_rf

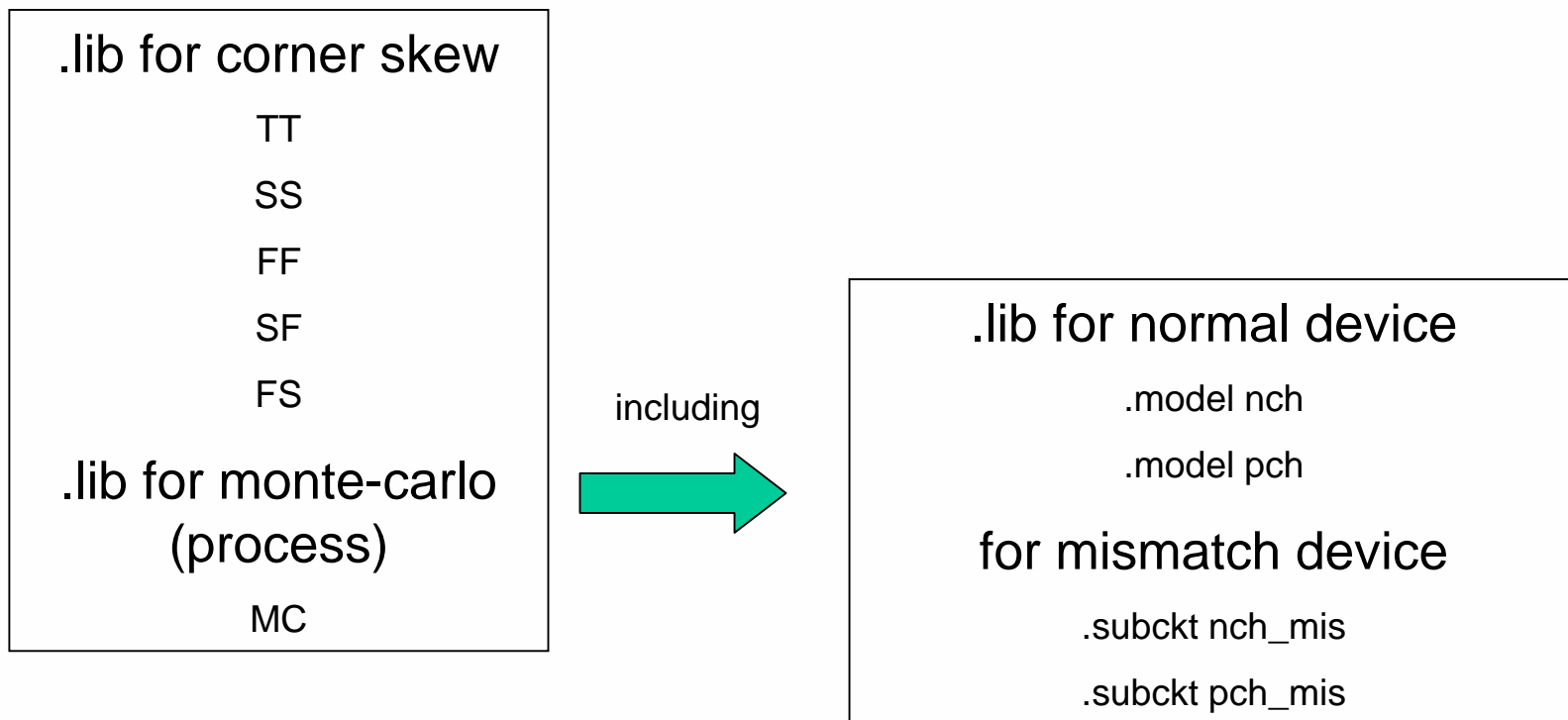
for mismatch device

.subckt nmos_rf_mis

.subckt pmos_rf_mis

Specify the mismatch devices to distinguish from the normal devices. Monte-Carlo the mismatch devices only and keep normal devices unchanged while performing MC simulation .

Structure for baseband CMOS model



Specify the mismatch devices to distinguish from the normal devices. Monte-Carlo the mismatch devices only and keep normal devices unchanged while performing MC simulation .

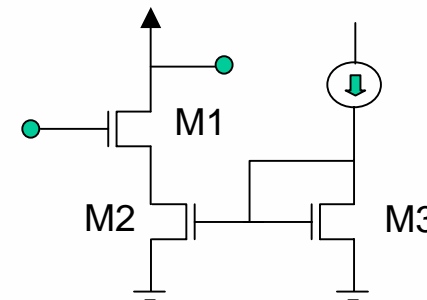


Device selection table

library	device	description
TT	nch	Typical baseband NMOS
TT	nch_mis	Typical baseband NMOS with mismatch
FF	nch	FF corner baseband NMOS
FF	nch_mis	FF corner baseband NMOS with mismatch
MC	nch	statistical baseband NMOS (die-to-die variation)
MC	nch_mis	statistical baseband NMOS (die-to-die variation) with mismatch
TT_RFMOS	nmos_rf	Typical RF NMOS
TT_RFMOS	nmos_rf_mis	Typical RF NMOS with mismatch
FF_RFMOS	nmos_rf	FF corner RF NMOS
FF_RFMOS	nmos_rf_mis	FF corner RF NMOS with mismatch
MC_RFMOS	nmos_rf	statistical RF NMOS (die-to-die variation)
MC_RFMOS	nmos_rf_mis	statistical RF NMOS (die-to-die variation) with mismatch

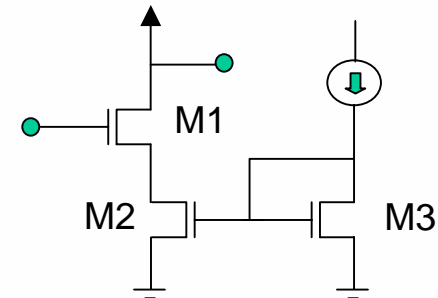
Example in HSPICE, ELDO

- M1(main NMOS): RF NMOS => chose normal device model: nmos_rf
- M2, M3(biasing ckt): baseband NMOS with mismatch => chose device model with mismatch: nch_mis
- for typical case simulation (Monte-Carlo simulation not performed)=>choose .lib TT and TT_RFMOS
- for typical case with M2 and M3 mismatch (monte-carlo simulation) => choose .lib TT and TT_RFMOS alone with monte-carlo simulation (only M2 and M3 are changed in each MC test, M1 keep its typical case value for M1 is a normal device)
- for die-to-die (process) Monte-Carlo simulation
=> choose .lib MC and MC_RFMOS (M1, M2 and M3 all are changed in each MC test)
- if mismatch in M2 and M3 is not to be simulated in the MC tests
=> change M2 and M3 to normal devices (nch instead of nch_mis)



Example in SPECTRE

- M1(main NMOS): RF NMOS => chose normal device model: nmos_rf
- M2, M3(biasing ckt): baseband NMOS with mismatch => chose device model with mismatch: nch_mis
- for typical case simulation (Monte-Carlo simulation not performed)=>choose .lib TT and TT_RFMOS
- for typical case with M2 and M3 mismatch (performing monte-carlo simulation and select the “mismatch” button) => choose .lib TT and TT_RFMOS alone with monte-carlo simulation (only M2 and M3 are changed in each MC test, M1 keep its typical case value for M1 is a normal device)
- for die-to-die (process) Monte-Carlo simulation (performing monte-carlo simulation and select the “process” button)
=> choose .lib MC and MC_RFMOS (M1, M2 and M3 all are changed in each MC test, however mismatch in M2 and M3 is not simulated in the MC tests)
- for die-to-die (process) and device-to-device (mismatch) Monte-Carlo simulation (performing monte-carlo simulation and select the “both” button)
=> choose .lib MC and MC_RFMOS (M1, M2 and M3 all are changed in each MC test and mismatch in M2 and M3 is simulated in the MC tests)





Usage and function in HSPICE, ELDO

.library baseband (RF)	not performing Monte-Carlo	performing Monte-Carlo
TT(TT_RFMOS)	nch (nmos_rf): typical case nch_mis(nmos_rf_mis): same as nch	nch (nmos_rf) : typical case (not changed) nch_mis (nmos_rf_mis) : MC test for mismatch with their mean equal to nch (typical)
FF(FF_RFMOS) and other corners	nch (nmos_rf): fast fast corner nch_mis(nmos_rf_mis): same as nch	nch (nmos_rf) : fast fast corner(not changed) nch_mis (nmos_rf_mis) : MC test for mismatch with their mean equal to nch (fast-fast corner)
MC(MC_RFMOS)	nch (nmos_rf): typical case nch_mis(nmos_rf_mis): same as nch	nch (nmos_rf) : MC test for die-to-die(process) variation nch_mis (nmos_rf_mis) : MC test for both die-to-die(process) and device-to-device variation (mismatch)



Usage and function in SPECTRE

.library baseband (RF)	not performing Monte- Carlo	performing Monte- Carlo of process	performing Monte- Carlo of mismatch	performing Monte-Carlo of process of both
TT(TT_RFMOS)	nch (nmos_rf): typical case nch_mis(nmos_rf_mis): same as nch	nch (nmos_rf): typical case nch_mis(nmos_rf_mis): same as nch MC test not functioning	nch (nmos_rf) : typical case (not changed) nch_mis (nmos_rf_mis) : MC test for mismatch with their mean equal to nch (typical)	nch (nmos_rf) : typical case (not changed) nch_mis (nmos_rf_mis) : MC test for mismatch with their mean equal to nch (typical)
FF(FF_RFMOS) and other corners	nch (nmos_rf): fast fast corner nch_mis(nmos_rf_mis): same as nch	nch (nmos_rf): fast fast corner nch_mis(nmos_rf_mis): same as nch MC test not functioning	nch (nmos_rf) : fast fast corner(not changed) nch_mis (nmos_rf_mis) : MC test for mismatch with their mean equal to nch (fast-fast corner)	nch (nmos_rf) : fast fast corner(not changed) nch_mis (nmos_rf_mis) : MC test for mismatch with their mean equal to nch (fast-fast corner)
MC(MC_RFMOS)	nch (nmos_rf): typical case nch_mis(nmos_rf_mis): same as nch	nch (nmos_rf): performing die-to- die(process) MC test nch_mis(nmos_rf_mis): performing die-to- die(process) MC test (same as nch)	nch (nmos_rf) : typical case (not changed) nch_mis (nmos_rf_mis) : MC test for mismatch with mean equal to nch (typical)	nch (nmos_rf) : MC test for die- to-die(process) variation nch_mis (nmos_rf_mis) : MC test for both die-to-die(process) and device-to-device variation (mismatch)