HICUM in commercial simulators

important differences between V2.0 and V2.1

- model parameters
  - **mCf** (and **mCr**), **hjei** were added for III-V and SiGe HBTs
  - **IS** is now officially an alternative to **c10**
  - **zetacx**: temperature dependence of the transit time of parasitic substrate transistor
  - some vendors did not have **hfc** implemented although it was in V2.0
  - **krbi** (noise): was never in DEVICE, but in the documentation, should be dropped generally
- documentation
  - default values have been defined for a *working* and *implementation* set (cf. www)
  - definition of operating point variables (although not directly related to model code)
- Comment from Yo-Chien, Agilent (ADS), regarding element value limitations

  "... other vendors may have done something different and customers will see different results and will think that something is not right..."

  => EDA companies to define unified limitations or model has to carry its own limitations

  => the latter will result in more code related burden on model developers

DEVICE (and stand-alone) code

- certain vendors seemed to have gathered different "versions" over the time (from mid nineties)
  => the HICUM model code in DEVICE was cleaned up (adding tunneling current & substrate transistor, changing CJCx implementation corresp. to the doc on www)
- KB, QEL were extended from 4 to 6 digits to obtain close agreement to commercial simulators
- emitter-base breakdown: turning of the calculation above Vbep > 0 (at forward bias) saves operations and time. Note: a process that shows any non-negligible BE tunneling current of the implemented type will not be used for design.
- transfer current implementation of parasitic substrate transistor:
  the following coding error was found (which is of little practical relevance though):
  The controlling node potential **V_BxCi** should be changed to **V_BpCi**, consistent with the nodes.
- in HICCR: avoid arithmetic underflow for *T_f0*I_Tf* by assigning finite default values to

```
I_TF=1.0E-30
I_TR=1.0E-30
```

- internal base resistance modification due to emitter perimeter minority charge
  - -*QjCi* > *QjEi* at low forward bias VBE is rare but physically possible (e.g. caused by bad parameter combination)
    => the pole in f_p (and negative r_Bi) needs to be avoided under any circumstances.
    => decision: drop QjCi in the f_p calculation (will not have an impact in practical cases where rB is of importance)
  - ***modified*** formulation in procedure HICRBI:

```
IF(Q_f.GT.0.0D0) THEN
  Q_fi=Q_f*f_qi
  f_p=(Q_jEi+Q_fi)/(Q_jEi+Q_f)
```

  - for consistency reasons (frequency & time domain simulation), CHICRBI needs to be modified accordingly:

```
CCCCC   operating point dependent quantities
        L_l=3
        CALL GETLMLB(LM,LB,L_l,L_M,L_B)
        K1=L_B+LMODS(L_M,3)+LMODS(L_M+1,3)
        C_j=PACIR(L_B+2)
```

frequently asked questions and related code fixes

- certain vendor codes
  - Tf equation at high JC: normalization term, (*1+sqrt(1+A_hc*), was implemented as just the factor "2"
  - *Qds = T_sf *its* to be replaced by *Qds = T_sf * itsf* (corresp. to (2.1.12-3)on page 64 of manual)

- Internal base resistance
  - Q: Missing derivatives: dIrbi/dV... due to Grbi = f(V)
    A: the dependence of rBi vs V within a practical bias range is small and can be neglected. We had done many years ago even simulations o circuits (looking at the small signal input impedance) and did not see any appreciable difference between including and not including the derivatives.
  - CjCi - punch-through voltage
    - The present equation is numerically not suitable for non-physical punch-through voltages outside the range 0 > (VPTCi,VPTCx) <= 500
    - the upper limit depends on the implementation (single or double precision): MAX_EXP=80 limitation is too low

frequently asked questions and related code fixes (cont'd)

- understanding DEVICE matrix setup for MNA
  - example for inserting a voltage controlled transfer current source in DEVICE

```
CALL CMLDTSG(NBP,NS, NBP,NCI,NS, I_Tsu,-S_suB-S_suS,S_suS)
```

  - NBP is the reference node for the controlled current source (any node can be chosen);
  - NCI, NS are the nodes, with respect to which the conductances have been calculated.
  - I_Tsu contains two exponentials that depend on the potential V_Ci, which is the first node to take the derivate for.
  - The derivative terms in the DEVICE code below are more conveniently calculated with respect to nodes Bx and S => derivative w.r.t. node Ci is the sum of the above. Please take a careful look at the following DEVICE code and the definition of the auxiliary variables S_suB and S_suS:

```
        a=DEXP(V_BpCi/UM)
        b=DEXP(V_SCi/UM)
        I_Tsu=IS*(a-b)
CCCCC   derivative dI_Tsu/dV_Bx @ VSi, VCi = const
        S_suB=IS*a/UM
CCCCC   derivative dI_Tsu/dV_S @ VBx, VCi = const
        S_suS=-IS*b/UM
```