Navid Paydavosi (navidp@eecs.berkeley.edu)

b4ld.c

- BSIM4mobMod=4, 5, and 6 were introduced as modified [and variability friendly] forms of mobMod=0, 1, and 2, respectively: (~line 1458) (Synopsys)

```
else if (model->BSIM4mobMod == 4) /* Synopsys 08/30/2013 add */
        {
      T0 = Vgsteff + here->BSIM4vtfbphi1 - T14;
      T2 = pParam->BSIM4ua + pParam->BSIM4uc * Vbseff;
      T3 = T0 / toxe;
      T12 = sqrt(here->BSIM4vtfbphi1*here->BSIM4vtfbphi1 + 0.0001);
      T9 = 1.0/(Vgsteff + 2*T12);
      T10 = T9*toxe;
      T8 = pParam->BSIM4ud * T10 * T10 * here->BSIM4vtfbphi1;
      T6 = T8 * here->BSIM4vtfbphi1;
      T5 = T3 * (T2 + pParam->BSIM4ub * T3) + T6;
      T7 = - 2.0 * T6 * T9;
      dDenomi_dVg = (T2 + 2.0 * pParam->BSIM4ub * T3) / toxe;
      dDenomi_dVd = 0.0;
      dDenomi_dVb = pParam->BSIM4uc * T3;
      dDenomi_dVg+= T7;
        }
else if (model->BSIM4mobMod == 5) /* Synopsys 08/30/2013 add */
        {
      T0 = Vgsteff + here->BSIM4vtfbphi1 - T14;
      T2 = 1.0 + pParam->BSIM4uc * Vbseff;
      T3 = T0 / toxe;
      T4 = T3 * (pParam->BSIM4ua + pParam->BSIM4ub * T3);
      T12 = sqrt(here->BSIM4vtfbphi1 * here->BSIM4vtfbphi1 + 0.0001);
      T9 = 1.0/(Vgsteff + 2*T12);
      T10 = T9*toxe;
      T8 = pParam->BSIM4ud * T10 * T10 * here->BSIM4vtfbphi1;
      T6 = T8 * here->BSIM4vtfbphi1;
      T5 = T4 * T2 + T6;
      T7 = - 2.0 * T6 * T9;
      dDenomi_dVg = (pParam->BSIM4ua + 2.0 * pParam->BSIM4ub * T3) * T2/ toxe;
      dDenomi_dVd = 0.0;
      dDenomi_dVb = pParam->BSIM4uc * T4;
      dDenomi_dVg+= T7;
        }
else if (model->BSIM4mobMod == 6) /* Synopsys 08/30/2013 modify */
    {   T0 = (Vgsteff + here->BSIM4vtfbphi1) / toxe;
      T1 = exp(pParam->BSIM4eu * log(T0));
      dT1_dVg = T1 * pParam->BSIM4eu / T0 / toxe;
      T2 = pParam->BSIM4ua + pParam->BSIM4uc * Vbseff;
      T12 = sqrt(here->BSIM4vtfbphi1 * here->BSIM4vtfbphi1 + 0.0001);
      T9 = 1.0/(Vgsteff + 2*T12);
      T10 = T9*toxe;
      T8 = pParam->BSIM4ud * T10 * T10 * here->BSIM4vtfbphi1;
      T6 = T8 * here->BSIM4vtfbphi1;
      T5 = T1 * T2 + T6;
      T7 = - 2.0 * T6 * T9;
      dDenomi_dVg = T2 * dT1_dVg + T7;
      dDenomi_dVd = 0;
      dDenomi_dVb = T1 * pParam->BSIM4uc;
    }
```

- Bug Fix: The following derivative issues in igcMod=2 were fixed. (~line 2548) (Synopsys)

```
if(model->BSIM4igcMod == 1) {
        dVaux_dVd = 0.0;
        dVaux_dVb = 0.0;
} else if (model->BSIM4igcMod == 2) {
        - dVaux_dVd = -dVgs_eff_dVg * dVth_dVd;
        - dVaux_dVb = -dVgs_eff_dVg * dVth_dVb;
        + dVaux_dVd = -dVaux_dVg* dVth_dVd;
```

```
+ dVaux_dVb = -dVaux_dVg* dVth_dVb;  }
```

- Change references of "model->BSIM4toxp" to "here->BSIM4toxp" and "model->BSIM4coxp" to "here->BSIM4coxp" on lines 1772, 1782-83, 3583-84, 3605 (ADI)

- ~Line2610 that is

  ```
  dT7_dVb *= dVbseff_dVb;
  ```

  was commented out because "dVbseff_dVb" is already considered in Lines 2660 and 2664. (Synopsys)

b4set.c

- mobmod checking criteria was changed to allow mobmod=4/5/6. (~ line 79) (Synopsys)
- Default value of **version** was set to 4.80. (~line 202)
- Default value of **fgidl** was set to 1.0. (~line 454) (Mentor Graphics)
- Default values of *GISL* parameters **agisl**, **bgisl**, **cgisl**, **egisl, rgisl, kgisl,** and **fgisl** were set to those of *GIDL's.* (~line 456 to 508) (ADI)
- Parameters **lk1**, **lk2, wk1**, **wk2**, **pk1**, and **pk2** were initialized (default set to zero). (~lines 801, 809, 1177, 1185, 1553, 1559) (ADI)
- Default values of *GISL* parameters **lagisl**, **lbgisl**, **lcgisl**, **legisl, lrgisl, lkgisl,** and **lfgisl** were set to those of *GIDL's.* (~line 951 to 1008) (ADI)
- Default values of *GISL* parameters **wagisl**, **wbgisl**, **wcgisl**, **wegisl, wrgisl, wkgisl,** and **wfgisl** were set to those of *GIDL's.* (~line 1327 to 1385) (ADI)
- Default values of *GISL* parameters **pagisl**, **pbgisl**, **pcgisl**, **pegisl, prgisl, pkgisl,** and **pfgisl** were set to those of *GIDL's.* (~line 1704 to 1763) (ADI)

b4check.c

- The following check on **BSIM4version**

  ```
  if (strcmp(model->BSIM4version, "4.7.0") != 0)
              {  fprintf(fplog, "Warning: This model is BSIM4.7.0; you specified a wrong version number.\n");
              printf("Warning: This model is BSIM4.7.0; you specified a wrong version number.\n");       }
  ```

  was changed to (ADI):

  ```
  if (fabs(model->BSIM4version - 4.80) > 0.0001)
        {  fprintf(fplog, "Warning: This model is BSIM4.8.0; you specified a wrong version number.\n");
           printf("Warning: This model is BSIM4.8.0; you specified a wrong version number.\n");       }
  ```

- Change references of "model->BSIM4toxp" to "here->BSIM4toxp" on lines 74-77 and 455-458. (ADI)
- The following checks on body resistances were added. (ADI)

  ```
  /* Check body resistance parameters */
  if (model->BSIM4rbps0 <= 0.0)
              {   fprintf(fplog, "Fatal: RBPS0 = %g is not positive.\n", model->BSIM4rbps0 );
                  printf("Fatal: RBPS0 = %g is not positive.\n", model->BSIM4rbps0);
                  Fatal_Flag = 1;
              }
  if (model->BSIM4rbpd0 <= 0.0)
              {   fprintf(fplog, "Fatal: RBPD0 = %g is not positive.\n", model->BSIM4rbpd0 );
                  printf("Fatal: RBPD0 = %g is not positive.\n", model->BSIM4rbpd0);
  ```

```
                Fatal_Flag = 1;
            }
if (model->BSIM4rbpbx0 <= 0.0)
            {   fprintf(fplog, "Fatal: RBPBX0 = %g is not positive.\n", model->BSIM4rbpbx0);
                printf("Fatal: RBPBX0 = %g is not positive.\n", model->BSIM4rbpbx0);
                Fatal_Flag = 1;
            }
if (model->BSIM4rbpby0 <= 0.0)
            {   fprintf(fplog, "Fatal: RBPBY0 = %g is not positive.\n", model->BSIM4rbpby0);
                printf("Fatal: RBPBY0 = %g is not positive.\n", model->BSIM4rbpby0);
                Fatal_Flag = 1;
            }
 if (model->BSIM4rbdbx0 <= 0.0)
            {   fprintf(fplog, "Fatal: RBDBX0 = %g is not positive.\n", model->BSIM4rbdbx0);
                printf("Fatal: RBDBX0 = %g is not positive.\n", model->BSIM4rbdbx0);
                Fatal_Flag = 1;
            }
if (model->BSIM4rbdby0 <= 0.0)
            {   fprintf(fplog, "Fatal: RBDBY0 = %g is not positive.\n", model->BSIM4rbdby0);
                printf("Fatal: RBDBY0 = %g is not positive.\n", model->BSIM4rbdby0);
                Fatal_Flag = 1;
            }
if (model->BSIM4rbsbx0 <= 0.0)
            {   fprintf(fplog, "Fatal: RBSBX0 = %g is not positive.\n", model->BSIM4rbsbx0);
                printf("Fatal: RBSBX0 = %g is not positive.\n", model->BSIM4rbsbx0);
                Fatal_Flag = 1;
            }
if (model->BSIM4rbsby0 <= 0.0)
            {   fprintf(fplog, "Fatal: RBSBY0 = %g is not positive.\n", model->BSIM4rbsby0);
                printf("Fatal: RBSBY0 = %g is not positive.\n", model->BSIM4rbsby0);
                Fatal_Flag = 1;
            }
```

b4temp.c

- For computational efficiency, the code was improved for computation of **RBSBX, RBSBY, RBDBX, RBDBY, RBPBX, RBPBY, RBPS0,** and **RBPD0**. (~line 1780 to 1829) (ADI). For example:

BSIM4.7.0

```
rbpbx = exp( log(model->BSIM4rbpbx0) + model->BSIM4rbpbxl * lnl + model->BSIM4rbpbxw * lnw + model->BSIM4rbpbxnf * lnnf );
```

BSIMS4.8.0

```
rbpbx = model->BSIM4rbpbx0 *exp( model->BSIM4rbpbxl * lnl +  model->BSIM4rbpbxw * lnw + model->BSIM4rbpbxnf * lnnf);
```

- Missing binning equations for **TVOFFCV**, **AGISD**, **BGISD**, and **CGISD** were added. (~lines 869 and 1005-16) (Cadence, Synopsys, Mentor Graphics & ADI)

```
pParam->BSIM4tvoffcv = model->BSIM4tvoffcv + model->BSIM4ltvoffcv * Inv_L + model->BSIM4wtvoffcv * Inv_W + model->BSIM4ptvoffcv * Inv_LW;

 pParam->BSIM4aigsd = model->BSIM4aigsd + model->BSIM4laigsd * Inv_L + model->BSIM4waigsd * Inv_W + model->BSIM4paigsd * Inv_LW;

pParam->BSIM4bigsd = model->BSIM4bigsd + model->BSIM4lbigsd * Inv_L + model->BSIM4wbigsd * Inv_W + model->BSIM4pbigsd * Inv_LW;

pParam->BSIM4cigsd = model->BSIM4cigsd + model->BSIM4lcigsd * Inv_L + model->BSIM4wcigsd * Inv_W + model->BSIM4pcigsd * Inv_LW;
```

- Bug fixes: (~line 1720)

```
- fprintf(stderr, "Warning: ku0we = %g is negatively too high. Negative mobility! \n");
```

+ fprintf(stderr, "Warning: ku0we = %g is negatively too high. Negative mobility! \n",ku0we);

- Changes made to correct "BSIM4toxp" and "BSIM4coxp": these values may be instance-specific (**here->BSIM4toxp** instead of **model->BSIM4toxp**);
  in b4temp.c, line 2233-2353,
  if(model->BSIM4mtrlMod != 0 && model->BSIM4mtrlCompatMod == 0)
  the computations involve here->BSIM4vfb and here->BSIM4vth0, and thus it is an error to assign values to model->BSIM4toxp and model->BSIM4coxp (ADI)

  1. if mtrlMod != 0 or mtrlCompatMod == 0, make assignments to "**here->BSIM4toxp**" and "**here->BSIM4coxp**";
     else copy the model values to the instance (lines 2351-56)
  2. delete computations of **here->BSIM4nstar** (2 locations); it is properly computed in b4ld.c and is not used in b4temp.c:
     - here->BSIM4nstar = Vtmeot / Charge_q * (model->BSIM4coxe + tmp1 + pParam->BSIM4cit);
     - here->BSIM4nstar = model->BSIM4vtm / Charge_q * (model->BSIM4coxe + tmp1 + pParam->BSIM4cit);
  3. move the calculations **of pParam->BSIM4VgsteffVth** and **pParam->BSIM4dvtp2factor** inside the "Size_Not_Found" block so that they are not recomputed for every instance.

b4noi.c

- Unused temporary variable was removed (~line 400)
  - T5 = here->BSIM4Vgsteff / here->BSIM4EsatL;
  - T5 = T5 * T5;

b4.c

- Based on a request from Cadence, model parameter **version** changed into a real number instead of a string:

  - IOP("version", BSIM4_MOD_VERSION, IF_STRING, "parameter for model version"),

  + IOP("version", BSIM4_MOD_VERSION, IF_REAL, "parameter for model version"),

b4mask.c

- Based on a request from Cadence, model parameter **version** changed into a real number instead of a string:

  case BSIM4_MOD_VERSION :

  - value->sValue = model->BSIM4version;

  + value->rValue = model->BSIM4version;

b4mpar.c

- Based on a request from Cadence, model parameter **version** changed into a real number instead of a string:

  case BSIM4_MOD_VERSION :

  - mod->BSIM4version = value->sValue;

  + mod->BSIM4version = value->rValue;

- Missing "break" statement for the following "case" commands were added: (~Lines 1118, 1644X) (ADI)

```
case  BSIM4_MOD_JTWEFF :
                mod->BSIM4jtweff = value->rValue;
                mod->BSIM4jtweffGiven = TRUE;
+           break;


case  BSIM4_MOD_LDVTP5:
                 mod->BSIM4ldvtp5 = value->rValue;
                 mod->BSIM4ldvtp5Given = TRUE;
+         break;
```

bsim4def.h

- **BSIM4toxp** and **BSIM4coxp** were added to the structure sBSIM4instance (ADI)
- **BSIM4aigsd**, **BSIM4bigsd**, and **BSIM4cigsd** were added to the struct bsim4SizeDependParam
- Based on a request from Cadence, model parameter **version** changed into a real number instead of a string:

```
- char   *BSIM4version;
+ double BSIM4version;
```