Contents

1 Multiband quantum transport in tunnel devices

In this section, we will describe a tight-binding (TB)-based transfer matrix approach used to calculate the proper scattering states of an heterostructure system [1]. TB models for the materials constituting the heterostructure allow to obtain a complete full-band description of the whole Brillouin zone of the interface plane and to relax all the envelope function approximations usually made for the treatment of tunneling problems in heterostructures.

1.1 Quantun transmitting boundary method

In this approach, the transmission matrix is determined in terms of an extended transfer-matrix method within the tight-binding framework [2] [5] [8] [9] [10] [11].

This method, called (multiband) Quantum Transmitting Boundary Method (***QTBM***) is a numerically stable and efficient method for computing transmission coefficients in semiconductor heterostructures using (multiband) band structure models. It has been successfully implemented for the empirical tight-binding model, the effective bond orbital, the $k \cdot p$ model, and the pseudopotential method and has been used extensively in the investigation of band structure effects such as valley mixing and band mixing.

The flexibility of the method allows to examine a variety of physical phenomena relevant to quantum transport, including alloy disorder, interface roughness, defect impurities, and 0D, 1D, and 2D quantum confinement, in device geometries ranging from double barrier heterostructures to quantum wire electron waveguides and MOS tunnel structures with non-uniform ultra-thin oxide layers.

Fig. 1 shows the energy band diagram of a typical heterostructure on which we wish to perform scattering calculations. The device structure consists of a central active region sandwiched between flat-band electrode regions to the left and right. The scattering

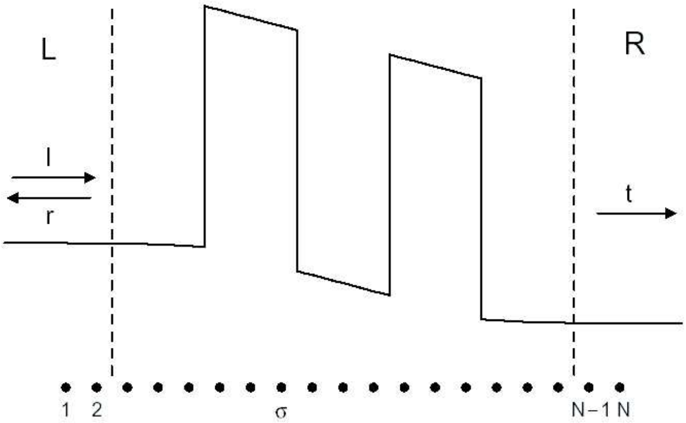

Figure 1: *Scheme of the energy band diagram of a generic double barrier heterostructure.*

problem just described is traditionally solved using the transfer matrix method.

The major drawback of the transfer matrix method lies in its inherent numerical instability. While this is typically not an issue in 1D single-band implementations, it can be a major concern for multiband and 3D applications.

The quantum transmitting boundary method dispenses with transfer matrices entirely. QTBM is a local basis method, suitable for tight-binding band structure models as well as discretized (finite difference or finite element) plane wave basis models.

1.2 Multiband method for the empirical tight-binding model

The multiband quantum transmitting boundary method (***MQTBM***) is a multiband generalization of the one-band quantum transmitting boundary method. Compared to the transfer matrix method, the MQTBM is equal in numerical efficiency, and superior in numerical stability and ease of implementation. Here is a description of MQTBM for the empirical tight-binding model [8].

Let's consider a generic tunnel heterostructure device. For convenience we vertically subdivide the whole device into three sections: a central region of interest made by N monolayers ($s = 1, 2, ..., N$) and two semi-infinite flat-band regions on the left and on the right, corresponding to the incoming and outgoing electrons. These are considered sufficiently far from the heterointerface so that the potential and compositions are constant and the wavefunction can be described in terms of bulklike plane-wave states. In general, the transfer-matrix procedure consists in relating the amplitudes of the incoming plane wave states to the transmitted ones. In the tight-binding framework, the amplitudes are translated, with an opportune basis transformation, into coefficients of tight-binding orbitals. Let M be the number of orbitals per unit cell in the chosen tight-binding basis set. The basis orbitals may be written in the form $\left|\mathbf{R}_{\|} s\alpha\right\rangle$, where s is the monolayer label,$\mathbf{R}_{\|}$ is the in plane component of unit cell coordinate and α labels the orbitals within a unit cell. Let's consider now a planar orbital formed by taking Bloch sums of tight-binding orbitals over the $N_{\|}$ unit cells in a generical monolayer :

$$\left|s\alpha, \mathbf{k}_{\|}\right\rangle = \frac{1}{\sqrt{N_{\|}}} \sum_{\mathbf{R}_{\|}} \exp(i\mathbf{k}_{\|} \cdot \mathbf{R}_{\|}) \left|\mathbf{R}_{\|} s\alpha\right\rangle \tag{1}$$

Then, since $\mathbf{k}_{\|}$ is a good quantum number, the wave-function can be expanded as a linear combination of these planar orbitals:

$$|\psi\rangle = \sum_{s,\alpha} C_{s\alpha} \left|s\alpha, \mathbf{k}_{\|}\right\rangle \tag{2}$$

The coefficients of this expansion must satisfy the Schrödinger equation, written in the planar orbital basis :

$$\left(H_{s,s} - E\right) C_s + H_{s,s-1} C_{s-1} + H_{s,s+1} C_{s+1} = 0 \tag{3}$$

where $H_{s,s}$ and $H_{s,s-1}$, $H_{s,s+1}$ are respectively the intra- and interslice bulk Hamiltonian matrices for the left bulk. The dimension M of the vector C_s is determined by the number of orbitals within the lateral unit cell . Analogous equations hold for the right bulk. Equation 3 can be written in the transfer matrix form :

$$\begin{pmatrix} -H_{s,s-1}^{-1}(H_{s,s}-E) & -H_{s,s-1}^{-1}H_{s,s+1} \\ 1 & 0 \end{pmatrix} \begin{pmatrix} C_s \\ C_{s+1} \end{pmatrix} = \Gamma_s \begin{pmatrix} C_s \\ C_{s+1} \end{pmatrix} = \begin{pmatrix} C_{s-1} \\ C_s \end{pmatrix} \tag{4}$$

where Γ_s is the transfer matrix for a single layer. The boundary conditions can be described in the bulk-state basis by choosing the correct form for the wave function in the left and right regions, that is such that there is a known incoming plane-wave state from the left region, no incoming states from the right, and unknown outgoing transmitted and reflected states in the right and left regions. These conditions can be translated into the tight-binding basis by means of a basis transformation which relates the coefficients of the in-coming, reflected, and transmitted plane-waves, respectively **I**, **r**,and **t**, to the tight-binding coefficients at the boundaries of the central region

$$\begin{pmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \end{pmatrix} = \mathbf{D}^L \begin{pmatrix} \mathbf{I} \\ \mathbf{r} \end{pmatrix} = \begin{pmatrix} \mathbf{D}_{11}^L & \mathbf{D}_{12}^L \\ \mathbf{D}_{21}^L & \mathbf{D}_{22}^L \end{pmatrix} \begin{pmatrix} \mathbf{I} \\ \mathbf{r} \end{pmatrix} \tag{5}$$

$$\begin{pmatrix} \mathbf{C}_{N-1} \\ \mathbf{C}_N \end{pmatrix} = \mathbf{D}^R \begin{pmatrix} \mathbf{t} \\ 0 \end{pmatrix} = \begin{pmatrix} \mathbf{D}_{11}^R & \mathbf{D}_{12}^R \\ \mathbf{D}_{21}^R & \mathbf{D}_{22}^R \end{pmatrix} \begin{pmatrix} \mathbf{t} \\ 0 \end{pmatrix} \tag{6}$$

$\mathbf{D^L}$ and $\mathbf{D^R}$ are 2M x 2M matrices whose column vectors are the eigenvectors obtained by diagonalizing the bulk transfer matrices in the left and right electrodes. In this bulk regions, in fact, the TB coefficients must satisfy the Bloch condition for near slices :

$$C_s = e^{ik_z d_z} C_{s-1} \tag{7}$$

where d is the distance between monolayers and k_z is the component of the crystal momentum along the growth direction.

Applying the condition (7) to Eq.4 , it is possible to set up a generalized eigenvalue equation of dimension $2N \times 2N$ that does not involve inverse of $\mathbf{H}_{s,s+1}$ [3, 4] :

$$\begin{bmatrix} (\mathbf{H}_{s,s} - E) & \mathbf{H}_{s,s+1} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{C}_s \\ \mathbf{C}_{s+1} \end{bmatrix} = e^{-ik_\perp \cdot \mathbf{d}_\perp} \begin{bmatrix} -\mathbf{H}_{s,s-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{C}_s \\ \mathbf{C}_{s+1} \end{bmatrix} . \tag{8}$$

whose solutions are a set of $2M$ complex wave vectors. They correspond to bulk states which propagate or decay to the right and to the left. In the left and right regions the wave function can be expressed, respectively , as:

$$|\psi; L\rangle = \sum_{j=1}^{M} \left(I_j \, |k_{\perp,j}; L\rangle + r_j \, |k_{\perp,j+M}; L\rangle \right) \tag{9}$$

$$|\psi; R\rangle = \sum_{j=1}^{M} t_j \, |k_{\perp,j}; R\rangle \tag{10}$$

where $|k_{\perp,j}; L\rangle$ and $|k_{\perp,j}; R\rangle$ are the bulk complex band states in the left and right regions, respectively , and the coefficients $\{I_j\}$, $\{r_j\}$ and $\{t_j\}$ represent the known incoming states, the reflected and the transmitted components.

Since the eigenvectors of Eq.8 occur in pairs $(k_\perp, -k_\perp)$, we have exactly N in-coming solutions and N out-going solutions.

In Fig. 2 the GaAs complex band structure in the [001] direction is calculated by solving the eigenvalue equation (8) with the spin-sp^3s^* ETB model. We notice that the light hole band is connected with the lower conduction band, while heavy and split–off hole bands are connected with higher energy conduction bands. The curvature in the imaginary plane defines the decay constant in a typical tunneling problem. Thus, a precise description over the whole complex $\mathbf{k}$ space is crucial for tunneling based devices.

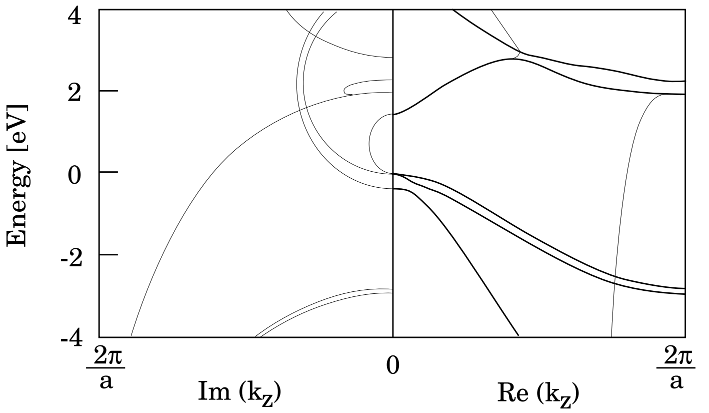

Figure 2: *Tight–binding complex band structure of GaAs. The thick lines refer to real bands, while the thin lines are related to complex values of the k_z vector.*

In the classical transfer-matrix method, by means of the repeated application of transfer matrices (4), one can relate the TB coefficients of left and right bulks to each other , and so obtain the transmission amplitude t. Using this method with realistic multiband band-structure models arises numerical instability problems, when large devices are considered.

To avoid these instabilities, one can bypass transfer matrices completely, by eliminating the unknowns **r**,and **t** from the two equations. In this way, we obtain the quantum transmitting boundary condition and the problem is reduced to a system of linear equations with only the TB coefficients as the unknowns.

After the inversion of matrix **D**, the boundary conditions can be written in this way :

$$\left[\left(\mathbf{D}_L^{-1}\right)_{11}\left(\mathbf{D}_L^{-1}\right)_{12}\right]\begin{pmatrix}\mathbf{C}_1\\\mathbf{C}_2\end{pmatrix}=\mathbf{I} \tag{11}$$

$$\left[\left(\mathbf{D}_R^{-1}\right)_{21}\left(\mathbf{D}_R^{-1}\right)_{22}\right]\begin{pmatrix}\mathbf{C}_{N-1}\\\mathbf{C}_N\end{pmatrix}=\mathbf{0} \tag{12}$$

These two boundary equations, together with Eq.3, constitute a tridiagonal system of

MN linear equations which can be written in the matrix form as :

$$\begin{bmatrix} (\mathbf{D}_L^{-1})_{11} & (\mathbf{D}_L^{-1})_{12} & 0 & \cdots & \cdots & \cdots & 0 \\ H_{2,1} & \bar{H}_{2,2} & H_{2,3} & 0 & \cdots & \cdots & 0 \\ 0 & H_{3,2} & \bar{H}_{3,3} & H_{3,4} & 0 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & & & 0 & H_{N-1,N-2} & \bar{H}_{N-1,N-1} & H_{N-1,N} \\ 0 & \cdots & \cdots & \cdots & 0 & (\mathbf{D}_R^{-1})_{21} & (\mathbf{D}_R^{-1})_{22} \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ \vdots \\ C_{N-1} \\ C_N \end{bmatrix} = \begin{bmatrix} \mathbf{I} \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \tag{13}$$

Solving the system one gets the TB coefficients and from these the amplitude of transmitted and reflected states are easily calculated :

$$\mathbf{t} = \left[\left(\mathbf{D}_R^{-1}\right)_{11} \left(\mathbf{D}_R^{-1}\right)_{12} \right] \begin{pmatrix} \mathbf{C}_{N-1} \\ \mathbf{C}_N \end{pmatrix} \tag{14}$$

$$\mathbf{r} = \left[\left(\mathbf{D}_L^{-1}\right)_{21} \left(\mathbf{D}_L^{-1}\right)_{22} \right] \begin{pmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \end{pmatrix} \tag{15}$$

In turn, from $\mathbf{t}$ the transmission coefficient can be computed:

$$T(E, \mathbf{k}_{\|}) = \sum_{j=1}^{M} \left| t_j(E, \mathbf{k}_{\|}) \right|^2 \frac{\left| v_j(E, \mathbf{k}_{\|}; R) \right|}{\left| v_I(E, \mathbf{k}_{\|}; L) \right|} \tag{16}$$

where $v_I(E, \mathbf{k}_{\|}; L)$ and $v_j(E, \mathbf{k}_{\|}; R)$ are the group velocities of the incident and the transmitted bulk plane-wave states, respectively.

The current density is then calculated integrating for $\mathbf{k}_{\|}$ in the projected Brillouin zone and for all the energy values, with the Landauer-Büttiker expression [6] [7] :

$$J = \frac{-e}{4\pi^3 \hbar} \int\limits_{BZ_{//}} d\mathbf{k}_{//} \int T\left(E, \mathbf{k}_{//}\right) \left[f_R\left(E, E_{FR}\right) - f_L\left(E, E_{FL}\right) \right] dE \tag{17}$$

where $T(E, k_{||})$ is the transmission coefficient, $BZ_{||}$ is the two dimensional Brillouin zone, E_{FL} the quasi-Fermi level in the left bulk and E_{FR} the quasi-Fermi level in the right bulk .

Note that MQTBM (or QTBM in general) does not specify how the system of the linear equations should be solved; it merely formulates the physical problem into a set of mathematical equations which can be solved using readily available numerical routines. This is in contrast to the transfer matrix method which not only provides the formulation, but also specifies the solution algorithm, which, unfortunately, is numerically unstable.

In conclusion, QTBM has the following advantages:

(1) ***Simplicity***. QTBM is extremely simple to implement: the basic QTBM equations consist simply of the effective mass Schrödinger equation in tight-binding or finite-element form for the active region, plus a pair of boundary equations specifying the scattering boundary condition.

(2) ***Numerical stability***. QTBM allows the user to choose the most stable and efficient numerical algorithm. Note that a poor choice such as back substitution, can of course lead to instabilities.

(3) ***Numerical efficiency***. At first glance, the transfer matrix method would appear to be more efficient, dealing with 2 x 2 matrices rather than the large N x N matrix in QTBM. However, the QTBM matrix is extremely sparse, and the use of sparse matrix algorithms leads to computational and storage efficiencies equaling that of the transfer matrix method.

2 HETERO simulation code for calculation of tunneling properties in heterostructures

2.1 Introduction

HETERO simulation code is an implementation of Quantun Transmitting Boundary Method for the calculation of multiband quantum transport in tunnel devices.
It can be applied to Resonant Tunnel Diodes (RTD) as well as to other structures implying some kind of tunnel transport, such as Metal-Oxide-Semiconductor (MOS) diodes, provided that a complete atomistic and tight-binding (TB) description of the device or structure is given. Only coherent tunneling is taken in account in these calculations, so no phonon scattering can be considered. On the other hand, defect or trap assisted tunneling can be studied, by introducing purposely some dangling bonds in the atomic basis of the structure. Besides, it must be noted that the code allows the integration of transmission coefficient on all the 2D Brillouin zone and it's not limited to the Gamma point contribution; this is of fundamental importance for the Tunneling Current calculation.

Two main kinds of information are required in input by the code:

1. An atomic structure description of the heterostructure under exam.
2. A set of TB parameters valid for the constituent materials of the heterostructure.

The program is executed with this command:

```
hetero   input_file_name
```

where *input_file_name* is the name of the input file: a text file including all the relevant information for the code to run.
In the following, we will see in details how to write the input file which is read by ***hetero*** and will discuss how to run a simulation in the appropriate way.

2.2 Atomic basis definition

The atomic structure must consist of a list of atoms, each followed by their *xyz* coordinates. This representation has to be consistent with a set of three primitive lattice vectors. Two of them define the lateral cell which corresponds to the periodicity of the structure in the *xy* plane. In the *z* direction, which is taken as the growth direction of the heterostructure, and so also as the direction along which tunneling takes place, there is of course no periodicity. The third vector is therefore to be considered meaningful only for the two bulk materials placed at the two contact regions of the device.
In the *lattice* section of the input file, first the primitive lattice vectors are defined (in nm), with the following format: *primII* is the couple of vectors which lie in the plane of the heterojunction; the next two (*prim3L* and *prim3R*) define the third primitive lattice vectors corresponding respectively to the left and right bulk.

```
$lattice
 primII =    0.5430     0.00000    0.00000
             0.00000      0.5430    0.00000
 prim3L =    0.00000     0.00000     0.543000
 prim3R =    0.00000     0.00000     0.543000
```

Then, after the keyword ”*device_basis*”, the complete atom basis of the heterostructure is defined, with this format:

Atom name, Atom number, Plane, Insulator, Position(3).

Plane is the order number of the crystal layer: it must be noted that the first and the last plane should be a basis of each bulk material;

Insulator is a flag (*.false.* or *.true.*) which determines whether that atom belongs or not to an ” *Insulator*” material. This label will be explained in the following section about current calculation.

Finally, the atom coordinates are entered, in the format *x* , *y, z* (in nm)

```
device_basis =
  "Sil"  1  1 .false. 0.4764296 0.2057682 -0.50050
  "Sil"  2  1 .false. 0.2049289 0.4772689 -0.50050
  "Sil"  3  1 .false. 0.0691927 0.0700178 -0.364750
  "Sil"  4  1 .false. 0.3406793 0.3415185 -0.364750
  "Sil"  5  1 .false. 0.2049289 0.2057682 -0.229000
  "Sil"  6  1 .false. 0.4764296 0.4772548 -0.229000
  ..................................................
 $end
```

The device should in fact be designed in this way:

- One left contact region, constituted by a bulk material (usually with a supercell basis)
- The ”*device*” region, where tunneling takes place
- One right contact region, constituted by a bulk material

Since the two bulk contact layers are seen by the code as infinite reservoirs, it’s not important their thickness, which usually corresponds to that of a supercell of the given material. Of course, most important is the thicknesses of the layer(s) in the device region, which mainly determine the tunneling.

2.3 TB parameters

Depending on the TB parameterization which has been chosen, the related states and couplings have to be declared, for each atom species, in the section *states_and_used_couplings*. Here, after the keyword *atom_states* each atom name is followed by the list of used couplings in the order s, p, d, s^* (use a dot for unused couplings).

```
$states_and_used_couplings


 atom_states =
 "Sil"  ’s’    ’p’    ’d’   ’se’
 "Si"   ’s’    ’p’    ’.’   ’.’
 "Ox"   ’s’    ’p’    ’.’   ’.’


 used_couplings =
     "sss"    "sps"     "pss"     "pps"      "ppp"   "s*ps"      "ps*s"
      "s*s*s"  "s*ss"   "ss*s"  "sds"    "dss"    "s*ds"    "ds*s"      "pds"
       "dps"      "pdp"     "dpp"   "dds"     "ddp"    "ddd"


 max_nr_stars = 1
$end
```

In this example, we declare the states used for a heterostructure Si/SiO_2, where Silicon ("Sil") has a sp^3s^*d representation, while SiO_2 (atoms "Si" and "Ox") has a

sp^3 basis. In the following, the relevant couplings for our heterostructure are listed (*used_couplings*) and the maximum number of stars (*max_nr_stars*) is put to 1, which means that only first neighbours interactions will be considered.

As for the TB parameters, two sets of values have to be entered: the on-site energies, required for each atom species present in the basis, and the TB couplings between different atom types. On-site energies are included in the related section , *onsite_energy* , as in the following example:

```
$onsite_energy
"Sil"   -2.0196    4.5448 14.1836    19.6748
"Si"   1.397     4.497      99.00        99.00
"Ox"   -20.503    -5.703   99.00        99.00
$end
```

where the atom name is entered , followed by the onsite energies list, in the order s,p,d,s^* and in eV (every state must be given, even if it is not used).
It must be noted that, for a heterostructure, the conduction (or valence) band discontinuity between different material layers must be taken in account in the correct way. Since TB parameters are usually given for a bulk material, the on-site energies must be shifted of the appropriate quantity, in order to get the correct band offset at the heterointerfaces.

TB couplings are listed in the section *tb_couplings*, with the following format: Atom name1, Atom name2, Star, max coupling distance, list of tb couplings.

```
$tb_couplings
"Sil"  "Sil"   1  0.25 -1.9413   2.7836    2.7836   4.1068    -1.5934  2.8428
   2.8428   -3.3081  -1.6933  -1.6933   -2.7998  -2.7998  -0.7003 -0.7003
```

```
     -2.1073  -2.1073  1.9977  1.9977  -1.2327  2.5145   -2.4734
"Ox"   "Si"   1    0.20 -3.0000   5.500    7.700   6.0000    -1.4000   0.000
    0.000     0.00      0.00      0.00      0.00       0.00   0.00   0.00    0.00
      0.00      0.00   0.00    0.00      0.00 0.00
"Ox"     "Ox"   1    0.28  0.0000    0.0000     0.0000   0.56    -0.13       0.000
     0.000     0.00      0.00      0.00      0.00       0.00   0.00   0.00    0.00
       0.00      0.00   0.00    0.00      0.00 0.00
$end
```

Star: A *star* is defined as the set of all atoms that have the same distance d from a given basis atom. Here, the parameter *star* defines how many stars are calculated: if we specify "1" , all the first nearest neighbours are considered;
max coupling distance defines the maximum distance which can separate two atoms for them to be still considered as first neighbours (can be useful in case of strain or lattice mismatch) ;
list of tb couplings must be given in the same order as in the section *states_and_couplings*

Finally, in the section *distance_dependence*, it is possible to enter the dependence of TB coupling parameters on bond length, according to a generalization of Harrison scaling law $V_{i,i'}(d_{i,i'}) = V_{i,i'}(d^0_{i,i'}) \cdot (\frac{d^0_{i,i'}}{d_{i,i'}})^{\eta}$ where $V_{i,i'}$ is the coupling integral between atoms i and i' and $d_{i,i'}(d^0_{i,i'})$ is the strained (unstrained) interatomic distance.
The parameters η are orbital-dependent exponents. They can be included in this section:

```
$distance_dependence
$end
```

according to the following format:

Atom name 1, Atom name 2, Star, reference distance[nm], power.

"*reference distance*" is the unstrained interatomic distance for that couple of atoms; "*power*" is the value of η for that coupling.

2.4 Simulation Control

In this paragraph we will examine the input parameters required to control the simulation and specify the kind of calculations to be performed.

After a "*subject*" section, dedicated to give a name and some comments to the input file, in the *general* section it is possible to set some very general simulation parameters.

```
$general
 program_version         = 1.32
 fuzzyequal              = .true.
 fuzzyness_in_percent    = 20.0
 calculate_lattice_only  = .false.
 calculate_bandstructure = .false.
 chatty                  = 2
$end
```

fuzzyequal and *fuzzyness_in_percent* are used to set a tolerance in the lattice structure. Namely, if *fuzzyequal* is *.true.*, all the atoms with a displacement within the percentage specified in *fuzzyness_in_percent* from the unstrained position, are considered still belonging to the same star. This allows to consider the crystal lattice as periodical, even in case of strain and displacements in the lattice positions.

calculate_lattice_only , if *.true.*, limits the simulation to calculate only the lattice structure. This can be very useful at the beginning of a device design, since it allows to determine the presence of dangling bonds or errors in the description of the basis.

calculate_bandstructure : if *.true.*, the program calculates the **bulk** bandstructure for the first or last plane, as specified in *bandstructure* section.

chatty : 0,1,2,3,4 specifies the level of information in output during execution.

In the section *general_heterostructure*, the control parameters for the tunneling calculation can be set up.

```
$general_heterostructure
 calculate_bound_states      = .false.
 calculate_current           = .true.
 calculate_special_initial_k = .false.
 calculate_I_U_character     = .true.
 calculate_kii_scan          = .false.
 calculate_E_scan            = .false.
 integrated_E                = .false.
 integrated_kII              = .false.
 BEEM_experiment             = .false.
 special_initial_energy      = 4.0
 special_initial_kII         = 0.0 0.0001 0.0
$end
```

In general, the simulation code allows one to calculate Transmission Coefficient and Tunneling current through the heterostructure. Regarding calculation of Transmission Coefficient T, three different modes of operation are available:

1. Calculation of T as a function of energy, for a given single $k_{||}$ point in the 2D Brillouin zone.

2. Calculation of T as a function of energy, with T being integrated on the whole 2D Brillouin zone.

3. Calculation of T for a fixed energy value and for a grid of k-points: in this way a picture of the transmission in the first 2D Brillouin zone can be obtained.

As for tunneling current, I/V characteristic is obtained in output, depending on the applied voltage between the two terminals of the system.

The previous options can be selected by switching the following flags in section *general_heterostructure*:

- *calculate_current* : determines if a $k_{||}$-space and energy integration is performed, as specified in the *tunneling_current* section;

- *calculate_I_U_character*: If *.true.* and *calculate_current* is *.true.*, the **current densities** for the range of voltage given in *I_U_character* will be calculated.

- *calculate_special_initial_k* : Calculates the transmission amplitude for a special $k_{||}$ as given by *special_initial_kII* (format : k_x , k_y, k_z $[nm^{-1}]$).

- *calculate_E_scan* : performs calculation of T, scanning an energy range given by *special_initial_energy* (starting from quasi Fermi level energy, see *tunneling_current* section).

- *calculate_kii_scan*: performs the calculation of T for a fixed energy value, given by *special_initial_energy* , and for a grid of k-points, given by *k_II_steps* in section *tunneling_current*. The result is a scan of T in the 2D Brillouin zone.

- *integrated_kII* : If *calculate_E_scan* is *.true.*, performs the calculation of integrated Transmission coefficient, where integration is made on the 2D Brillouin zone, with a number of k-points given by *k_II_steps* in section *tunneling_current*.

- *integrated_E* , *calculate_bound_states*, *BEEM_experiment* : not implemented.

The *bandstructure* section allows the calculation of the band structure of the left or the right bulk (only if *calculate_bandstructure* in the *general* section is *.true.*).
If *left_bulk* is *.true.*, the program calculates the band-structure of plane nr. 1 using primII,prim3L as primitive lattice vectors.
If *right_bulk* is *.true.*, the program calculates the band-structure of the last plane using primII,prim3R as primitive lattice vectors.
lower_energy_limit, *upper_energy_limit*: define the range of energy for the band-structure.
k3_file is the name of the file containing the k-vectors at which the band-structure should

be calculated. Format of the file :

total nr of k-points

nr-k-point k1 k2 k3

.....

where k1 and k2 are in plane wavevectors , k3 is the orthogonal wavevector i.e. in growth direction. All wavevectors are in units of reciprocal primitive lattice vectors.

2.5 Tunneling current calculation

The *tunneling_current* section allows to control the tunneling current calculation.

```
$tunneling_current
 calculate_three_pole_device = .false.
 basis_emitter_voltage       = 0.0
 basis_atom_basis            = 38
 quasi_fermi_energy_L        = 1.241
 quasi_fermi_energy_B        = 1.241
 quasi_fermi_energy_R        = 0.30
 base_fermi_cutoff           = .false.
 plane_symmetry              = 9
 temperature                 = 300.0
 cutoff_energy_diff          = 0.180
 energy_step                 = 0.005
 energy_min_step             = 0.0001
 k_II_steps                  = 50
 adaptive_kII_grid           = .true.
 nr_subgrids                 = 6
 focus_point_only            = .false.
 focus_point                 = 0.0 0.0
 focus_point_radius      = 0.12
$end
```

To calculate the current density in the device (only performed for *calculate_current* = *.true.*), an integration over the energy interval from (*quasi_fermi_energy_R*-*cutoff_energy_diff*) to (*quasi_fermi_energy_L*+ *cutoff_energy_diff*) is performed. That is, the energy interval is defined by quasi-Fermi energy (in eV) in left (L) and right (R) bulk layers, and by a cutoff energy.

energy_step is the stepsize for energy integration; the stepsize will be decreased to *energy_min_step*/2 (sometimes down to *energy_min_steps*/4) if the transmission amplitude is close to a peak.

For each energy a 2D integration over $k_{||}$ is performed with *k_II_steps* in each direction. *k_II_steps* defines the equivalent nr. of points in each direction of the projection of the BZ over which the (integrated) transmission is sampled. The actual nr. of calculated points can be well below (*k_II_steps*)^2 if the BZ-projection posesses high symmetry. A possible C_n symmetry along the growth axis may speed up the last summation significantly.

plane_symmetry gives the 2D-symmetry of the heterostructure in growth direction. Possible values:

0 : No symmetry, full BZ-integration

2 : C2, i.e. a 180-degree rotation symmetry along growth axis. The in plane primitive lattice vectors must be of equal length and enclose 180deg.

3 : C3, i.e. a 120-degree rotation symmetry along growth axis. The in plane prim latt. vec must be of equal length and enclose 120deg. e.g. (1,0,-1) and (-1,1,0)

4 : C4, i.e. a 90-degree rotation symmetry along growth axis. The in plane prim latt. vec must be of equal length and enclose 90deg.

6 : C6, i.e. a 60-degree rotation symmetry along growth axis. The in plane prim latt. vec must be of equal length and enclose 120deg.

7 : C3v, a 120-degree rotation symmetry and three mirror planes that contain the basis vectors.

8 : C6v, a 60-degree rotation symmetry and six mirror planes that contain the basis vectors.

9 : C2v, a 180-degree rotation symmetry and one mirror plane that lies in the middle between two primitive translations (Diamond-(100))

10 : C4v, a 90-degree rotation symmetry and a mirror plane that lies in the middle between two primitive translations (Flourite-(100)), fcc-(100).

adaptive_kII_grid: If this parameter is set to *true*, a denser grid is created with additional gridpoints between those given by *k_II_steps*. These are only calculated if a high transmission is nearby.

nr_subgrids: If *adaptive_kII_grid* is true, between two coarse gridpoints given by *k_II_steps* there will be $\sqrt{2}^{(nr\_subgrids)}$ additional points, but only some of them will be calculated. The use of these parameters cause an exponential growth in the computational time, so it is advisable not to choose too high values (suggested value: 3) *temperature*: gives the system temperature; it affects only the Fermi-Distribution calculation.

Calculate_three_pole_device specifies if the device to be calculated has three and not only two leads. If this flag is *true*, the two following parameters *basis_atom_basis* and *quasi_fermi_energy_B* are meaningful.

basis_atom_basis determines a basis atom that belongs to the base terminal of the three pole device (eg. Transistor)

quasi_fermi_energy_B gives the quasi-fermi energy of the transistor base.

focus_point_only : if *true*, scanning of k-space in the 2D Brillouin zone, if requested, is limited to a specific region, defined by *focus_point* (k_x , k_y in nm^{-1}), that is the center of the region, and by *focus_point_radius*, that is the radius of the region.

In the section *I_U_character*, one can define parameters for IV characteristic :
voltage range (starting voltage , end voltage) and voltage step. If *calculate_I_U_character* is *.true.*, the current density as a function of the applied voltage is calculated for the heterostructure.

```
$I_U_character
 start_voltage = 0.80
 end_voltage   = 4.0
 voltage_step  = 0.1
```

```
$end
```

The calculation of current is performed in the following way.
A voltage drop, defined by the voltage step in *I_U_character*, is applied to the device structure described in *lattice* section. It is assumed that all the applied voltage drops in the "*device*" region, which is defined by the basis atoms whose label *insulator* , in the section *device_basis*, is set to *true.* No voltage drop occurs in the regions whose basis atoms have *insulator* set to *false*; usually, the left and right bulk contact regions are defined in this way.
In the region considered as insulator, a linear potential drop is assumed, which is implemented with a shift of TB on-site energies of the related atoms. Then, transmission coefficient $T(E, k_{||})$ is calculated with the new TB parameters and tunneling current is found integrating T on energy and $k_{||}$. This procedure is repeated for all the voltage steps, giving the final IV characteristic.

2.6 HETERO implementation for ICODE

For usage in the ***ICODE*** web site, a few modifications have been made to the I/O section of the code. Basically, the simulation control sections have been separated from the atom basis and TB parameters definition, in order to let the user run a simulation on a RTD or MOS structure in a quickly way, with no need to define manually the basis structure. On ***ICODE***, in fact, the basis structure is generated automatically by the code, according to the device geometrical parameters in user's input.
The user has, anyway, the option to edit his own input file for HETERO, which can be run from ICODE simulator applet.

In this case the input file for HETERO, which must have the name ***hetero.in***, has the following format (see also the example in Appendix A).

The sections *subject*, *general*, *general_heterostructure*, *I_U_character*, *bandstructure* are exactly as described above. In *tunneling_current* section, *quasi_fermi_energy* values are now considered with respect to Conduction Band bottom energy. New sections are *device_structure* and *ext_structure*:

```
$device_structure
 MOS                = .true.
 RTD_GaAs           = .false.
 RTD_GaN           = .false.
 t_ox    = 1.25
 well_RTD   = 3.4d0
 barr_RTD           = 3.4d0
$end

$ext_structure
  include_ext_structure_file = .true.
  name_ext_structure_file = 'prova_struct'
$end
```

In the first one , the user can select one of three predefined device structures, namely a Si/SiO_2 MOS diode, a GaAs-based RTD , a GaN-based RTD. Depending on the device selected, a relevant structure file will be read by the code, containing TB parameters and information about lattice. Then, geometrical parameters are entered, namely oxide thickness *t_ox* for the MOS or well (*well_RTD*) and barrier (*barr_RTD*) widths (all in nm) for RTDs. On the ground of these data, the code generates internally the appropriate atomic basis.

Instead of making use of predefined structures, the user has the option to write and include his own structure file: in section *ext_structure*, set *include_ext_structure_file* to *true* and specify, after *name_ext_structure_file*, the name of the user-defined external structure file. With these settings, *device_structure* section is overridden, and the structure basis and TB parameters are completely defined by the user through the file *name_ext_structure_file*.

2.6.1 External structure file

With the option *include_ext_structure_file*, the user has to provide, in the *ICODE* working directory, also the structure file defined by *name_ext_structure_file*, together with the file *hetero.in*.
The structure file (see also Appendix B) should include the sections related to atom basis (*lattice*, *device_basis*) and to TB parameters (*onsite_energy*, *states*, *used_TB_couplings*, *tb_couplings*, *distance_dependence*), exactly as described before. The only difference is that *states* and *used_TB_couplings* are now defined in two distinct sections, however with the same syntax as before. Finally, in the new section *top_CB_energy*, the value *CB_energy* (Conduction Band bottom energy) must be entered. This energy (in eV), will be the reference for energy scan calculations and should refer, usually, to left and right bulk layers material.

```
$top_CB_energy
  CB_energy   = 1.171
$end
```

A Example of *hetero.in* file

```
$subject
 name    = 'tunneling in a Si/SiO2 system'
 comment = 'sp3 and sp3s*d TB  basis'
$end

# General
#
# Fuzzyequal : if true deviations in distance up to "fuzzyness_in_percent"
#              are accepted without begining a new star
# calculate_lattice_only : stop after lattice was read in
# calculate_bandstructure : Calculates the bulk (!) bandstructure for the
#              first plane. Desired k-points must be given in a file whose
#              name is given in the second program argument
# chatty     : 0,1,2,3,4 Specifies how much the program talks to you. Ranges
#              from 0 (only what must be said) to 4 (tells you all secrets)

$general
 program_version         = 1.32
 fuzzyequal              = .true.
 fuzzyness_in_percent    = 20.0
 calculate_lattice_only  = .false.
 calculate_bandstructure = .false.
 chatty                  = 4
$end

# device_structure
#
# MOS  :  if  true   MOS  structure
#   RTD_GaAs    :  if  true   RTD  structure  GaAs/AlGaAs
#  RTD_GaN  :  if  true   RTD  structure  GaN/AlgaN
#  t_ox        :    oxide thickness for MOS  SiO2  [nm]
#   well_RTD   :    well   length  in  RTD    [nm]
#    barr_RTD  :   barrier  length  in  RTD   [nm]
#
#

$ext_structure
  include_ext_structure_file = .true.
  name_ext_structure_file = 'test_struct'
$end


$device_structure
 MOS                = .false.
  RTD_GaAs         = .false.
 RTD_GaN           = .false.
   t_ox  = 1.25
 well_RTD   = 3.4d0
 barr_RTD            = 3.4d0
$end

# General features of the heterostructure:
#
# calculate_bound_states :
#   If true, possible bound states of the system are calculated. (Not yet
#   implemented)
# calculate_current:
#   Determines if a kII-space and energy integration is performed as
#   specified in the tunneling_current structure.
# calculate_special_initial_k:
#   Calculates the transmission amplitude for a special kII, and energy,
#   as given by special_initial_kII and special_initial_energy.
#   If calculate_I_U_character or calculate_current is also true and
#   k_II_steps is 1 or zero instead of performing a kII - space integration
#   only the kII value as given by special_initial_kII is taken
# calculate_I_U_character:
#   If true and calculate_current true the current densities for the range
#   of voltage drops given in I_U_character will be calculate. Note that since
#   a quadruple integration has to be performed ( dU,dkII^2,dE ) this may
#   take a while!
#   The kII and energy integration are defined in the tunneling_current
#   structure. Note that if k_II_steps <= 1 the kII integration is skipped

$general_heterostructure
```

```
 calculate_bound_states      = .false.
 calculate_current           = .false.
 calculate_special_initial_k = .true.
 calculate_I_U_character     = .false.
 calculate_kii_scan          = .false.
 calculate_E_scan            = .true.
 integrated_E                = .false.
 integrated_kII              = .false.
 BEEM_experiment             = .false.
 special_initial_energy      = 4.0
 special_initial_kII         = 0.0 0.0001 0.0
$end

# Structure for current - voltage characteristics
#
# If calculate_I_U_character is true (above) the program reads in
# this structure and the current densitiy as a function of the applied
# voltage is calculated for the heterostructure.
# The quasi_fermi_energies given in the tunneling_current structure
# below are assumed to be equal on both bulk sides

$I_U_character
 start_voltage = 0.02
 end_voltage   = 2.5
 voltage_step  = 0.1
$end
```

```
# Informations required to calculate the band structure of the left
# or the right bulk.
#
# left_bulk     : If true the program calculates the band-structure of
#                 plane nr. 1 using primII,prim3L as primitive latt. vec.
# right_bulk    : If true the program calculates the band-structure of
#                 the last plane using primII,prim3R as primitive latt. vec.
# lower_energy_limit, upper_energy_limit : The nr. of bands at gamma-
#                 point determines the nr. of bands calculated at all
#                 points of the Brillouin Zone. i.e. if at gamma point
#                 lower_energy_limit <= E_n1 < E_n2 <= upper_energy_limit
#                 The bands n1 to n2 are printed at all points of the
#                 BZ
# k3_file       : file containing the k-vectors at which the band-structre
#                 should be calculated. Format of the file
#                 nr_of_entries*
#                 nr_entry* k1 k2 k3
#                 nr_entry* k1 k2 k3
#                 .....
#                 * = not evaluated
#                 k1,k2 = In plane wavevectors
#                 k3 = Orthagonal wavevector i.e. in growth direction
#                 All wavevectors in units of reciprocal prim. latt. vec.
```

```
$bandstructure
 left_bulk          = .true.
 right_bulk         = .false.
 lower_energy_limit = -5.0
 upper_energy_limit = 5.0
 k3_file            = 'k_GZ_Si.sl8_001.in'
$end


# To calculate the current density in the device (only performed for
# calculate_current == .true.) an integration over the energy interval
# from (quasi_fermi_energy_L-cutoff_energy) to (quasi_fermi_energy_L+
# cutoff_energy) is being performed weighting it the fermi function.
# For each energy an 2D summation over k_parallel is performed with
# k_II_steps in each direction. A possible C_n symmetry along the
# growth axis given by symmetry_around_growth_axis may speed up the
# last summation significantly.
# calculate_current determines if the energy and kII integration is
#                   performed to get the current density.
# Calculate_three_pole_device specifies if the device to be calcu-
#                   lated has three and not only two leads. If this
#                   flag is true, the two variables basis_atom_basis
#                   and quasi_fermi_energy_B are important.
# basis_atom_basis determines a basis atom that belongs to the
#                   basis of the three pole device (eg. Transistor)
# quasi_fermi_energy_B gives the fermi energy of the basis (center)
# plane_symmetry:
#   gives the 2D-symmetry the heterostructure posses in growth
#   direction. Possible values:
#   0 : No symmetry, full BZ-integration
#   2 : C2, i.e. a 180-degree rotation symmetry along growth axis. The
#       in plane prim latt. vec must be of equal length and enclose 180deg.
#   3 : C3, i.e. a 120-degree rotation symmetry along growth axis. The
#       in plane prim latt. vec must be of equal length and enclose 120deg.
#       e.g. (1,0,-1) and (-1,1,0)
#   4 : C4, i.e. a 90-degree rotation symmetry along growth axis. The
#       in plane prim latt. vec must be of equal length and enclose 90deg.
#   6 : C6, i.e. a 60-degree rotation symmetry along growth axis. The
#       in plane prim latt. vec must be of equal length and enclose 120deg.
#   7 : C3v, a 120-degree rotation symmetry and three mirror planes that
#       contain the basis vectors.
#   8 : C6v, a  60-degree rotation symmetry and six mirror planes that
#       contain the basis vectors.
#   9 : C2v, a 180-degree rotation symmetry and one mirror plane that
#       lies in the middle between two primitive translations (Diamond-(100) )
#  10 : C4v, a 90-degree rotation symmetry and a mirror plane that
#       lies in the middle between two primitive translations (Flourite-(100)),
#       fcc-(100);

# temperature:     gives the system temperature; It enters only over the
#                  Fermi-Distribution; Decoherence effects are neglected!
# energy_step:     Stepsize for energy integration; the stepsize will be de-
#                  crease down to energy_min_step/2 (sometimes down to
#                  energy_min_steps/4) if the transmissionamplitude is close
#                  to a peak.
# energy_min_step: Gives the energy difference below which a peak in the
#             energy integration is fitted to the lorentz function.
# k_II_steps:      Defines the equivalent nr. of points in each direction
#                  of the projection of the BZ over which the (integrated)
#                  transmission is sampled. The actual nr. of calculated
#                  nr. of points can be well below (k_II_steps)^2 if the
#                  BZ-projection posesses high symmetry.
#                  sugg. value: 50  [#12]
# adaptive_kII_grid: If set to true. there will be additional gridpoints
#                  between those given by k_II_steps, which are only
#                  calculated if a high transmission is nearby.
#                  sugg. value: .true. (for beasty problems)
# nr_subgrids:     Between two coarse gridpoints given by k_II_steps there
#                  will be sqrt(2)^(nr_subgrids) additional points, but only
#                  some of them will be calculated. (Only for adaptive_kII
#                  _grid == .true.) Exponential growth choose not too high!
#                  sugg. value: 3 (4) [#12  (mio = 3 )]

# Si    VB Top :         E = 0.0     eV
# Si    BC Bottom  :   E = 1.171     eV  (a T = 0 K )
# SiO2   CB Bottom :        E = 8.824 eV
# Si-SiO2 CB-offset:   dE = 3.15     eV
#
```

```
$tunneling_current
 calculate_three_pole_device = .false.
 basis_emitter_voltage       = 0.0
 basis_atom_basis            = 38

 quasi_fermi_energy_L        = 0.0
 quasi_fermi_energy_B        = 0.0
 quasi_fermi_energy_R        = 0.0
 base_fermi_cutoff           = .false.
 plane_symmetry              = 9
 temperature                 = 300.0
 cutoff_energy_diff          = 0.180
 energy_step                 = 0.3
 energy_min_step             = 0.0001
 k_II_steps                  = 1
 adaptive_kII_grid           = .false.
 nr_subgrids                 = 6
 focus_point_only            = .false.
 focus_point                 = 0.0 0.0
 focus_point_radius      = 0.12
$end
```

B Example of *structure file*

```
$top_CB_energy
  CB_energy   = 1.171
$end

# onsite_energies_list (in the order {s,p,d,se} and in eV (every state
# must be given even if unused)

$onsite_energy
"Sil"   -2.0196    4.5448 14.1836    19.6748
"SiD"   -2.0196    4.5448 14.1836    19.6748
"OxBr"   -20.503    -5.703   99.00       99.00
"OxIF"   -20.503    -5.703   99.00       99.00
"SiIF"   1.397     4.497     99.00       99.00
"Oxl22"   -20.503    -5.703   99.00       99.00
"Oxl23"   -20.503    -5.703   99.00       99.00
"Si"   1.397     4.497     99.00       99.00
"Ox"   -20.503    -5.703   99.00       99.00
"Oxl34"   -20.503    -5.703   99.00       99.00
"Oxl35"   -20.503    -5.703   99.00       99.00
$end

# The first two lattice vectors must lie in the plane of the heterojunction.
# The next two define the third primitive lattice vector corresponding to the
# left and right bulk. The first and the last plane are a basis of each bulk
# material

# Structure of device basis:
# Atom_name Atom_number Plane Insulator? Position(3)

$lattice
 primII =    0.5430    0.00000   0.00000
            0.00000     0.5430   0.00000
 prim3L =    0.00000    0.00000    0.543000
 prim3R =    0.00000    0.00000    0.543000

device_basis =
  "Sil"  1  1 .false. 0.4764296 0.2057682 -0.50050
  "Sil"  2  1 .false. 0.2049289 0.4772689 -0.50050
  "Sil"  3  1 .false. 0.0691927 0.0700178 -0.364750
  "Sil"  4  1 .false. 0.3406793 0.3415185 -0.364750
  "Sil"  5  1 .false. 0.2049289 0.2057682 -0.229000
  "Sil"  6  1 .false. 0.4764296 0.4772548 -0.229000
  "Sil"  7  1 .false. 0.3406793 0.0700178 -0.093250
  "Sil"  8  1 .false. 0.0691927 0.3415185 -0.093250
  "Sil"  9  2 .false. 0.4764296 0.2057682 0.0425
  "Sil"  10  2 .false. 0.2049289 0.4772689 0.0425
  "Sil"  11  2 .false. 0.0691927 0.0700178 0.17825
  "Sil"  12  2 .false. 0.3406793 0.3415185 0.17825
  "Sil"  13  2 .false. 0.2049289 0.2057682 0.314
```

```
"Sil"  14  2 .false. 0.4764296 0.4772548 0.314

"Sil"  15  2 .false. 0.3406793 0.0700178 0.44975

"Sil"  16  2 .false. 0.0691927 0.3415185 0.44975

"Sil"  17  3 .false. 0.4764296 0.2057823 0.5855

"SiD"  18  3 .false. 0.2049431 0.4772689 0.5855

"OxBr"  19  3  .true. 0.3406934 0.3415185 0.6055

"OxIF"  20  3  .true. 0.0691998 0.2057753 0.6595

"SiIF"  21  3 .true. 0.1907938 0.191619 0.6995

"Oxl22" 22  3  .true. 0.2049289 0.0848246 0.788

"Oxl23" 23  3  .true. 0.2049289 0.3464542 0.788

"Si"  24  4 .true. 0.2049289 0.4772689 0.8805

"Ox"  25  4  .true. 0.3357437 0.4772689 0.973

"Ox"  26  4  .true. 0.0543859 0.4772689 0.973

"Si"  27   4 .true. 0.4665584 0.4772689 1.0655

"Ox"  28  4  .true. 0.4665584 0.0650964 1.158

"Ox"  29  4  .true. 0.4665584 0.3464542 1.158

"Si"  30   4 .true. 0.4665584 0.2156394 1.2505

"Ox"  31  4  .true. 0.3357437 0.2156394 1.343

"Ox"  32  4  .true. 0.0543859 0.2156394 1.343

"Si"  33   4 .true. 0.2049289 0.2156394 1.4355

"Oxl34" 34  4  .true. 0.2049289 0.0848246 1.528

"Oxl35" 35  4  .true. 0.2049289 0.3464542 1.528

"SiIF"  36  5 .true. 0.219064 0.4729768 1.6165

"OxIF" 37  5  .true. 0.3406651 0.487126 1.6565

"OxBr"  38  5  .true. 0.0691573 0.079882 1.7105

"SiD"  39  6 .false. 0.2049148 0.2156394 1.7305
```

```
  "Sil"  40  6 .false. 0.4764155 0.4871401 1.7305
  "Sil"  41  6 .false. 0.3406793 0.0700178 1.86625
  "Sil"  42  6 .false. 0.0691927 0.3415185 1.86625
  "Sil"  43  7 .false. 0.4764296 0.2057682 2.002
  "Sil"  44  7 .false. 0.2049289 0.4772689 2.002
  "Sil"  45  7 .false. 0.0691927 0.0700178 2.13775
  "Sil"  46  7 .false. 0.3406793 0.3415185 2.13775
  "Sil"  47  7 .false. 0.2049289 0.2057682 2.2735
  "Sil"  48  7 .false. 0.4764296 0.4772548 2.2735
  "Sil"  49  7 .false. 0.3406793 0.0700178 2.40925
  "Sil"  50  7 .false. 0.0691927 0.3415185 2.40925
  "Sil"  51  8 .false. 0.4764296 0.2057682 2.54500
  "Sil"  52  8 .false. 0.2049289 0.4772689 2.54500
  "Sil"  53  8 .false. 0.0691927 0.0700178 2.68075
  "Sil"  54  8 .false. 0.3406793 0.3415185 2.68075
  "Sil"  55  8 .false. 0.2049289 0.2057682 2.81650
  "Sil"  56  8 .false. 0.4764296 0.4772548 2.81650
  "Sil"  57  8 .false. 0.3406793 0.0700178 2.952250
  "Sil"  58  8 .false. 0.0691927 0.3415185 2.952250
 $end


# STATES_AND_COUPLINGS
# Structure of states: Atom_name, used_couplings_list in the the order
# {s p d se}, use a dot for unused couplings
```

```
$states

 atom_states =
 "Sil"  ’s’  ’p’   ’d’      ’se’
 "SiD"  ’s ’     ’p ’    ’d ’   ’se’
 "Si"   ’s’     ’p’    ’.’   ’.’
 "SiIF" ’s’     ’p’    ’.’   ’se’
 "Ox"   ’s’     ’p’    ’.’   ’.’
 "Oxl22"  ’s’     ’p’    ’.’   ’.’
 "Oxl23"  ’s’     ’p’    ’.’   ’.’
 "Oxl34"  ’s’     ’p’    ’.’   ’.’
 "Oxl35"  ’s’     ’p’    ’.’   ’.’
 "OxIF" ’s’     ’p’    ’.’   ’.’
 "OxBr" ’s’     ’p’    ’.’   ’.’
$end_states


$used_TB_couplings

 used_couplings =
 "sss"     "sps"      "pss"      "pps"        "ppp"    "s*ps"
 "ps*s"    "s*s*s"  "s*ss"   "ss*s"  "sds"    "dss"    "s*ds"
 "ds*s"       "pds"   "dps"       "pdp"      "dpp"    "dds"     "ddp"     "ddd"

 max_nr_stars = 1
$end_used_TB_couplings
```

```
# Structure of couplings:
# Atom_name1 Atom_name2 Star max_coupling_distance tb_elements_of_used_couplings

# Atom1 Atom2      Star   mx_d        "sss"     "sps"
  "pss"      " pps"       "ppp"    "seps"       "pses"
   "seses"  "sess"   "sses"  "sds"    "dss"    "seds"
     "dses"       "pds"   "dps"       "pdp"      "dpp"
      "dds"     "ddp"     "ddd"

$tb_couplings
 "Sil"  "Sil"   1  0.25 -1.9413   2.7836    2.7836   4.1068
 -1.5934  2.8428  2.8428   -3.3081  -1.6933  -1.6933   -2.7998
   -2.7998  -0.7003   -0.7003   -2.1073
     -2.1073  1.9977  1.9977  -1.2327  2.5145   -2.4734
 "Sil"  "SiD"   1  0.25 -1.9413   2.7836    2.7836   4.1068
   -1.5934  2.8428   2.8428   -3.3081  -1.6933  -1.6933
   -2.7998  -2.7998  -0.7003 -0.7003   -2.1073
    -2.1073  1.9977  1.9977  -1.2327  2.5145   -2.4734
 "Ox"   "Si"   1    0.20 -3.0000   5.500    7.700   6.0000    -1.4000   0.000
   0.000     0.00     0.00     0.00     0.00      0.00   0.00   0.00    0.00
      0.00     0.00   0.00    0.00     0.00 0.00
 "Oxl22"   "Si"   1    0.18 -3.0000   5.500    7.700   6.0000    -1.4000   0.000
    0.000     0.00     0.00     0.00     0.00      0.00   0.00   0.00    0.00
      0.00     0.00   0.00    0.00     0.00 0.00
 "Ox123"   "Si"   1    0.18 -3.0000   5.500    7.700   6.0000    -1.4000   0.000
     0.000     0.00     0.00     0.00     0.00      0.00   0.00   0.00    0.00
        0.00     0.00   0.00    0.00     0.00 0.00
 "Ox134"   "Si"   1    0.18 -3.0000   5.500    7.700   6.0000    -1.4000   0.000
    0.000     0.00     0.00     0.00     0.00      0.00   0.00   0.00    0.00
     0.00     0.00   0.00    0.00     0.00 0.00
 "Ox135"   "Si"   1    0.18 -3.0000   5.500    7.700   6.0000    -1.4000   0.000
    0.000     0.00     0.00     0.00     0.00      0.00   0.00   0.00    0.00
      0.00     0.00   0.00    0.00     0.00 0.00
 "Ox"     "Ox"   1    0.28  0.0000   0.0000    0.0000   0.56    -0.13      0.000
   0.000     0.00     0.00     0.00     0.00      0.00   0.00   0.00    0.00
     0.00     0.00   0.00    0.00     0.00 0.00
 "OxIF"    "SiIF"   1   0.130 -3.0000   5.500    7.700   6.0000    -1.4000  0.000
   0.000     0.00     0.00     0.00     0.00      0.00   0.00   0.00    0.00
     0.00     0.00   0.00    0.00     0.00 0.00
 "Ox122"    "SiIF"   1   0.165 -3.0000   5.500    7.700   6.0000
   -1.4000  0.000 0.000     0.00     0.00     0.00     0.00      0.00   0.00
      0.00    0.00   0.00     0.00   0.00    0.00     0.00 0.00
 "Ox123"    "SiIF"   1   0.190 -3.0000   5.500    7.700   6.0000
   -1.4000  0.000    0.000     0.00     0.00     0.00     0.00      0.00
    0.00    0.00    0.00    0.00     0.00   0.00    0.00     0.00 0.00
 "Ox134"    "SiIF"   1   0.190 -3.0000   5.500    7.700   6.0000
    -1.4000  0.000   0.000     0.00     0.00     0.00     0.00
     0.00   0.00   0.00    0.00      0.00     0.00   0.00    0.00     0.00
0.00
 "Ox135"    "SiIF"   1   0.170 -3.0000   5.500    7.700   6.0000
```

```
  -1.4000  0.000 0.000     0.00     0.00     0.00     0.00      0.00   0.00
    0.00    0.00    0.00     0.00   0.00    0.00     0.00 0.00
 "OxIF"    "Sil"   1    0.17 -3.0000   5.500    7.700   6.0000
   -1.4000  0.000   0.000     0.00     0.00     0.00     0.00      0.00
    0.00   0.00    0.00     0.00     0.00   0.00    0.00     0.00 0.00
 "OxBr"    "SiD"   1   0.21 -3.0000   5.500    7.700   6.0000    -1.4000  0.000
  0.000     0.00     0.00     0.00     0.00      0.00   0.00   0.00    0.00
  0.00     0.00   0.00    0.00     0.00 0.00
 "OxBr"    "Sil"   1    0.21 -3.0000   5.500    7.700   6.0000
   -1.4000  0.000   0.000     0.00     0.00     0.00     0.00      0.00
    0.00   0.00    0.00    0.00     0.00   0.00    0.00     0.00 0.00
 "SiIF"   "SiD"    1  0.29 -1.9413   2.7836    2.7836   4.1068
   -1.5934  2.8428   2.8428   -3.3081  -1.6933  -1.6933   -2.7998  -2.7998
   -0.7003 -0.7003   -2.1073 -2.1073   1.9977  1.9977  -1.2327  2.5145   -2.4734
$end


# Structure of distance dependence:
# Atom_name_1 Atom_name_2 Star# reference_distance[nm] power

$distance_dependence
$end
```

References

[1] F. Sacconi, A. Di Carlo, P. Lugli, M. Staedele, J.M. Jancu ”Full-band approach to tunneling in MOS structures” IEEE Trans. on Electron Devices, 51, (2004)

[2] A. Di Carlo, Semicond. Sci. Technol., 18,R1, (2003)

[3] T. B. Boykin, Phys. Rev. B 54, 8107 (1996).

[4] T. B. Boykin, Phys. Rev. B 54, 7670 (1996).

[5] C. Strahberger and P. Vogl, Phys. Rev. B 62, 7289 (2000).

[6] R. Landauer, IBM J. Res. Dev. 32, 306 ,(1988)

[7] M. Buttiker, IBM J. Res. Dev. 32, 317 ,(1988)

[8] D.Z.-Y. Ting, E.T. Yu, and T.C. McGill, Phys. Rev. B 45, 3583 (1992)

[9] T.B. Boykin, Phys. Rev. B 54, 7670 (1996)

[10] T.B. Boykin, Phys. Rev. B 54, 8107 (1996)

[11] M. Staedele, B. R. Tuttle, and K. Hess, J. Appl. Phys. 89, 348 (2001)