1 Overview

1. calculation of electrode as a bulk
   output → data of the electrode
   e.g. % openmx au3.input
2. SCF of the central region
   output → H and S of the junction
   e.g. % openmx pol_c.input
3. calculation of transmission
   e.g. % MAIN_TRAN_Calc_Transmission pol_c.input

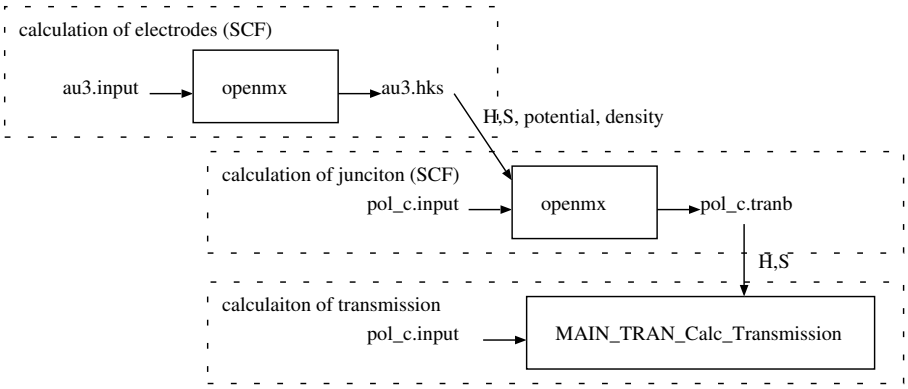

Figure 1:

To calculate I-V.

1. set bias voltages, V = left-right, or right-left
2. calculate current
3. and set next bias voltages

2 electrode

2.1 Atoms and Cell

First calculate the electronic structure of electrode as 'band'

```
Atoms.Number         3
Atoms.SpeciesAndCoordinates.Unit   Ang # Ang|AU
<Atoms.SpeciesAndCoordinates
 1   Au    0.00000    4.40243    0.84724    5.50000    5.50000
 2   Au    2.93496    4.40243    0.84724    5.50000    5.50000
 3   Au    5.86992    4.40243    0.84724    5.50000    5.50000
Atoms.SpeciesAndCoordinates>
Atoms.UnitVectors.Unit             Ang # Ang|AU
<Atoms.UnitVectors
   8.80488    0.00000    0.00000
    0.00000    4.40243    7.62524
```

```
    0.00000    8.80486    0.00000
Atoms.UnitVectors>
scf.EigenvalueSolver       band        # Recursion|Cluster|Band|NEGF
```

2.2 Output

Also write data for the next transport calculation.

```
tran.output_hks true                   # yes | no
tran.filename.hks   au3.hks            # filename
```

The output file is 'au3.hks'

3 transport

3.1 Data of electrode

Define the files which have data of the electrode

```
Tran.filename.hks.l     au3.hks
Tran.filename.hks.r     au3.hks
```

3.2 Atoms and Cell

Define atomic positions and cells

```
LeftLeadAtoms.Number         3
<LeftLeadAtoms.SpeciesAndCoordinates
 1   Au    0.00000    4.40243    0.84724    5.50000    5.50000
 2   Au    2.93496    4.40243    0.84724    5.50000    5.50000
 3   Au    5.86992    4.40243    0.84724    5.50000    5.50000
LeftLeadAtoms.SpeciesAndCoordinates>
<LeftLeadAtoms.UnitVectors
   -8.80488    0.00000    0.00000
    0.00000    4.40243    7.62524
    0.00000    8.80486    0.00000
LeftLeadAtoms.UnitVectors>

RightLeadAtoms.Number        3
<RightLeadAtoms.SpeciesAndCoordinates
 1   Au   28.95062    4.40243    0.84724    5.50000    5.50000
 2   Au   31.88558    4.40243    0.84724    5.50000    5.50000
 3   Au   34.82054    4.40243    0.84724    5.50000    5.50000
RightLeadAtoms.SpeciesAndCoordinates>
<RightLeadAtoms.UnitVectors
    8.80488    0.00000    0.00000
    0.00000    4.40243    7.62524
    0.00000    8.80486    0.00000
RightLeadAtoms.UnitVectors>

Atoms.Number         16
Atoms.SpeciesAndCoordinates.Unit   Ang # Ang|AU
<Atoms.SpeciesAndCoordinates
 1   Au    8.80488    4.40243    0.84724    5.50000    5.50000
 2   Au   11.73984    4.40243    0.84724    5.50000    5.50000
```

```
 3    S   14.24600    4.40243    0.84724    3.00000    3.00000
 4    C   16.00421    4.40243    0.84724    2.00000    2.00000
 5    C   16.71447    5.61653    0.84724    2.00000    2.00000
 6    C   16.71447    3.18833    0.84724    2.00000    2.00000
 7    C   18.10607    5.61653    0.84724    2.00000    2.00000
 8    C   18.10607    3.18833    0.84724    2.00000    2.00000
 9    H   16.16108    2.24682    0.84724    0.50000    0.50000
10    H   16.16108    6.55804    0.84724    0.50000    0.50000
11    H   18.67061    2.24682    0.84724    0.50000    0.50000
12    H   18.67061    6.55804    0.84724    0.50000    0.50000
13    C   18.81633    4.40243    0.84724    2.00000    2.00000
14    S   20.57454    4.40243    0.84724    3.00000    3.00000
15   Au   23.08070    4.40243    0.84724    5.50000    5.50000
16   Au   26.01566    4.40243    0.84724    5.50000    5.50000
Atoms.SpeciesAndCoordinates>
Atoms.UnitVectors.Unit             Ang # Ang|AU
<Atoms.UnitVectors
   20.14574    0.00000    0.00000
    0.00000    4.40243    7.62524
    0.00000    8.80486    0.00000
Atoms.UnitVectors>
```

The unit cell of the calculation is LeftLeadAtoms.UnitVectors + RightLeadAtoms.UnitVectors + Atoms.UnitVectors.

3.3 Solver

Define solver

```
scf.EigenvalueSolver       NEGF        # Recursion|Cluster|Band|NEGF
```

3.4 Integration path

Define path of integration1 in A.U.

```
Tran.Integ.Pathtype       square
Tran.squarePath.div       10 20 30     #   n1 n2 n3
Tran.squarePath.energies  -1.0 a auto a 1.0e-6 0.1  # e1 (a|r) e2 (a|r) E3 E4
```

$E1 = e1$ if a.
$E1 = e1$+min_of_eigenvalue if r. (not supported now)
$E2 = E_F$ if $e2$ =auto $E2 = e2$+ chemical_potential if r.

3.5 Bias voltage

source-drain bias

```
Tran.bias.apply      on     (on|off)
Tran.bias.voltage    0.01 0.0    # V1  V2
Tran.bias.div      20
```

$V1$ =bias of the left electrode, $V2$ =bias of the right electrode. the number of energy mesh is 'Tran.squarePath.bias.div'

3.6 Output

Output = 'System.Name'.tranb

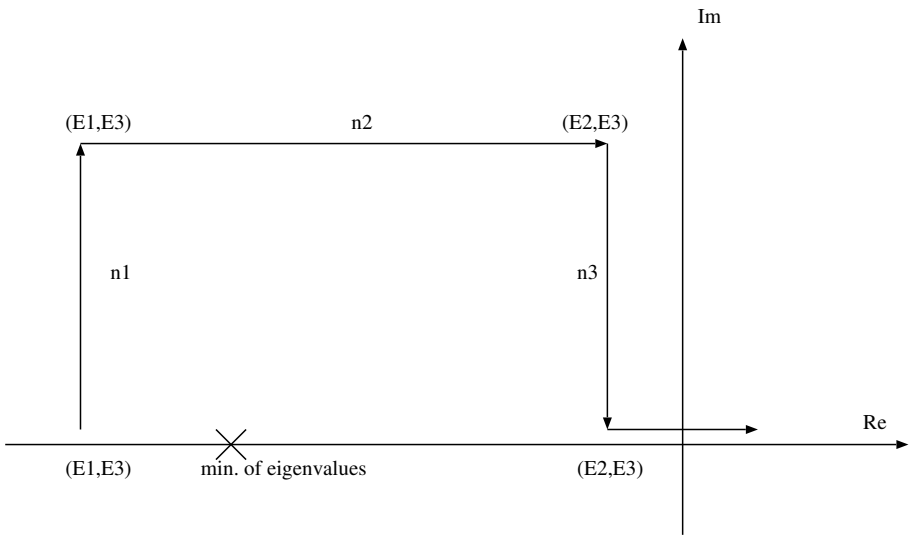

Figure 2:

4 transmission

4.1 Transmission

```
tran.transmission.on   on
Tran.transmission.energydiv  101
Tran.transmission.energyrange  -0.5 0.5 1.0e-6  # E1, E2, E3
```

transmission coeffecient is calculate in the range [E1+i E3, E1+i E4]. The number of energy mesh is 'Tran.transmission.energydiv'.

4.2 current

```
tran.transmission.iv.on  on
Tran.transmission.iv.energydiv 100
```

The number of energy mesh is 'Tran.transmission.iv.energydiv'. It is calculated in the range of 'Tran.bias.voltage'. The imaginary part is define in E3 of 'Tran.squarePath.energies'.

Please integrate the transmission!

4.3 current path

```
tran.currentpath      off     # on | off
```

not supported now.

5 TO DO LIST

1. complete the case of 'auto' in tran.squarepath.energyes.
2. integration in 'tran.transmission.iv'
3. tran.currentpath

4. other integration path
5. calculation of orbital energies and optimization