

OLED Simulation LAB

Introduction to Organic LED

In this lab, we will use a simulator to study the organic LEDs. The simulator has been indigenously developed by IIT Kanpur students. We will begin with a single layer device moving on to a bi-layer device and if time permits a tri-layer device.

- We will try and identify the currents in the devices as space charge limited or injection limited. And see the effect of HOMO/LUMO gap and device length on the current.
- We will try to improve the efficiency of a device by varying the mobility of the electrons and/or holes.

Working of an OLED

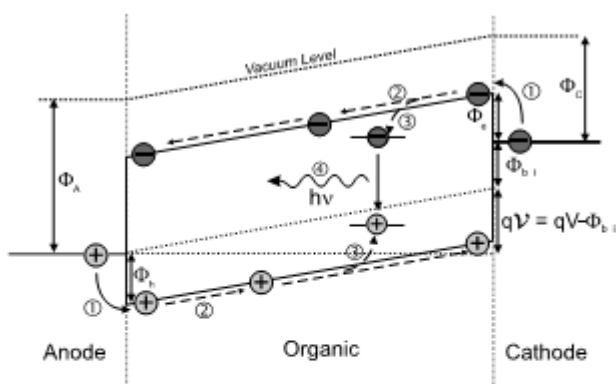


Fig. 1. Basic steps of EL: (1) charge carrier injection, (2) charge carrier transport, (3) exciton formation, (4) radiative exciton decay. Polaronic effects and a distribution of transport states due to disorder are neglected. (Φ_A : anode work function, Φ_C : cathode work function, Φ_h : hole injection barrier, Φ_e : electron injection barrier, Φ_{bi} : built-in potential, V : applied voltage, \mathcal{V} : effective voltage across the organic layer, q : elementary charge).

Figure 1 Brutting W.; Berleb S.; Muckl A.G.

Organic Electronics, Volume 2, Number 1, March 2001 , pp. 1-36(36)

Elsevier

The above figure explains the working of an OLED. The series of steps involved are:

- Charge carrier injection
- Charge carrier transport
- Exciton formation
- Exciton decay(light emission, may or may not)

Bi-layer device

It consists of hole transport layer (HTL) and electron transport layer (ETL).

A bi-layer device helps in blocking the holes in the middle of the device, and thus allows for the recombination of the charge carriers towards the centre of the device rather than near the cathode (or anode).

Materials are chosen so that the mobility of holes is higher in the HTL and less in ETL, and vice versa for the electrons. Thus both holes and electrons tend to accumulate near the interface of the two materials.

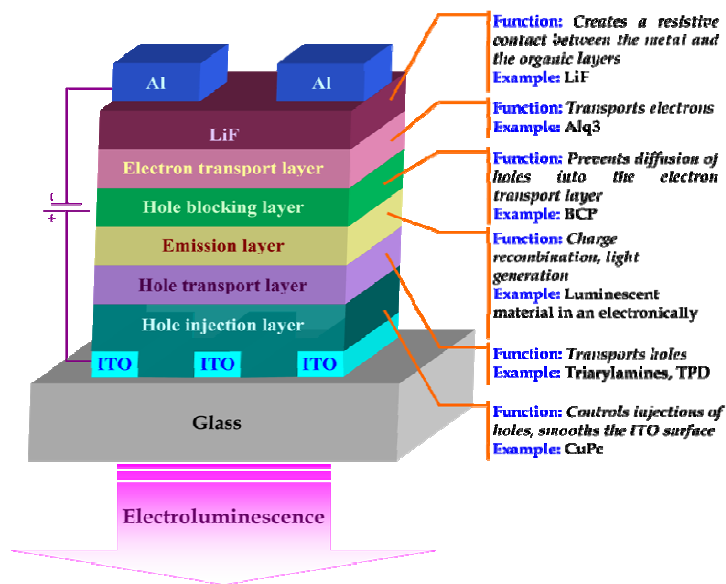
Tri-layer device

It consists of HTL, ETL and a emission layer(EML).

In such a device we try to keep the recombination maximum in the EML, where we would have the material through which we want radiation to take place.

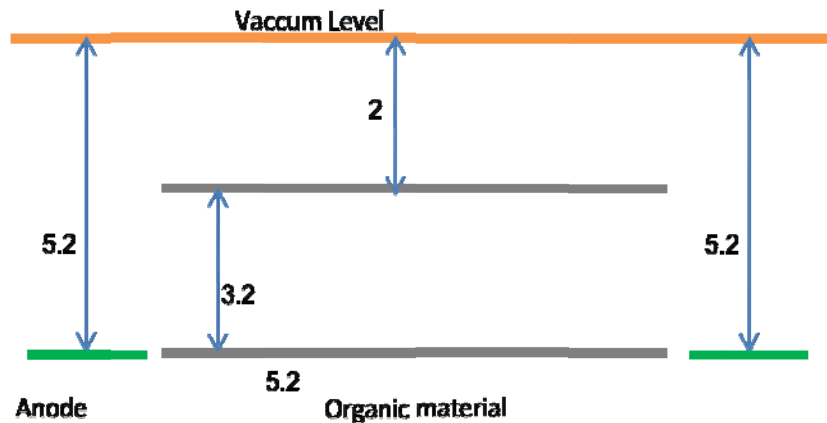
Multi-layer devices

We can keep on increasing the number of layers, and thus, make the recombination happen in a specific layer. But due to the large number of layers, the number of charge carriers which are recombining by themselves will be reduced significantly than in the smaller number of layers devices. Also this will increase the thickness of the device and thus also reducing the field. These trade off needs to be taken care of, while increasing the number of layers.



Schematic illustration of multi layer structure of small molecule based OLED Figure 2

Problem 1(Single layer device):



[The values are in eV]

Length of device = 200nm

Dielectric constant of the material = 3

,
,
,
,

Enter the starting and final voltages for the applied voltages for which you want to simulate. Also mention the voltage step.

Enter the parameters mentioned above in the code, '*simulator.cpp*'. The data only needs to be entered in the beginning of the file. All the values are to be entered as SI units, except the work functions and HOMO/LUMO gap which is to be entered in eV.

Determine whether the device is hole-only, electron-only or both carriers are present.

Problem 2:

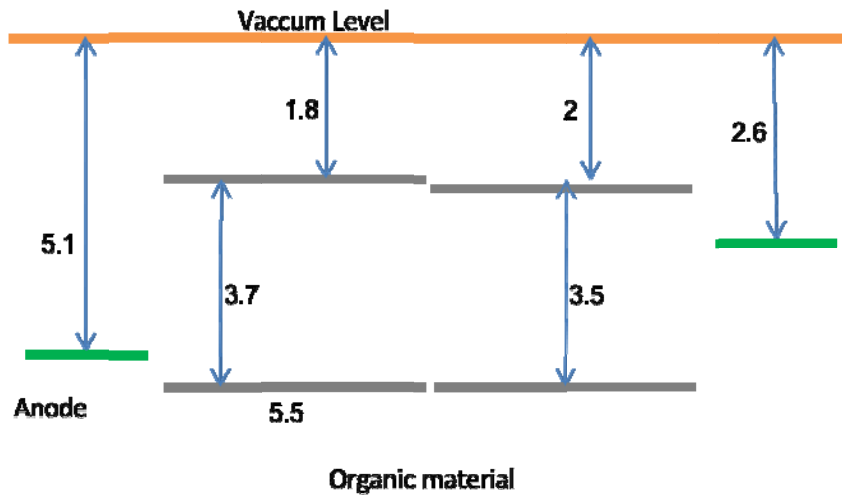
Is the current space charge limited? Why or why not?

If it is space charge limited, then determine the parameters in the expression,

$J = \frac{9}{8} \epsilon_0 \epsilon_r V^2 \mu$, using the expression for space charge limited current for charge carriers whose mobility is field dependent.

— — —

Problem 3(Bi-layer device):



Length of the device = 200nm [100nm,100nm]

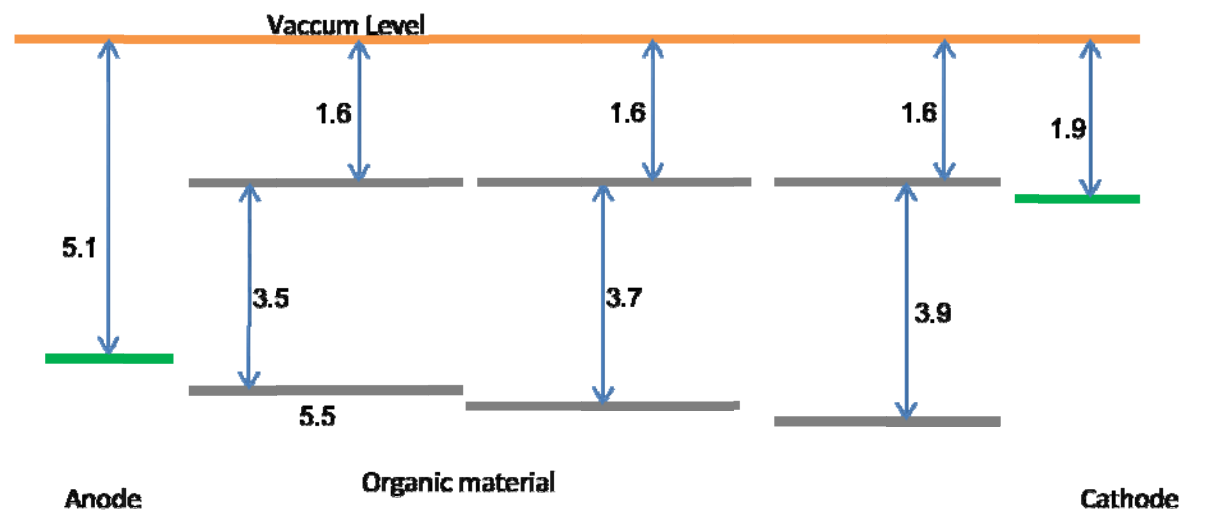
Dielectric constant of the material = 3

HTL	ETL

- Is the electron/hole mobility field dependent? Why or why not?(What parameter above is crucial in determining this dependence)
- Plot the recombination profile (column [N] versus column [J] in Origin) for applied voltage = 5V.
- Can you find the efficiency of the device? [Hint: You know the hole current values in the device] Find the efficiency of the device assuming that emission only takes place from a 10nm thick layer centered at the interface of the two layers. Increase the efficiency of the device by changing the mobility(μ_p and μ_n) and/or length of the layers.

Problem 4:

Tinker with the values given below for a tri-layer device, and try to improve the efficiency of the device in the middle layer, which is expected to be the EML(emission layer).



Length of device = 245nm [10nm,45nm,100nm]

Dielectric constant of the material = 3

HTL	EML	ETL

Instructions to use the OLED Simulator

1. Open '*simulator.cpp*' using Dev-C++.
2. In the beginning of the code you will find the variables which you need to specify. These variables are:
 - a. **wfa, wfc** : Work function of anode and cathode.
 - b. **len_full** : Length of device.
 - c. **epsi_r** : Dielectric constant of the device.
 - d. **layers** : No. of layers.
 - e. Data for individual layers is given in the form of arrays. For details on how to enter data in a array see bottom of the page.
 - i. **ea** : Electron affinity.
 - ii. **eg** : Band gap.
 - iii. **len** : Length of layer.
 - iv. **uN** : Zero field permittivity for electrons.
 - v. **uP** : Zero field permittivity for holes.
 - vi. **gamman** : Factor in Poole-Frenkel type relation of mobility, for electrons.
 - vii. **gammap** : Factor in Poole-Frenkel type relation of mobility, for holes.

$$\mu = \mu_o \exp(\gamma \sqrt{E_o})$$
 - viii. **v_start, v_final** : Applied voltage: starting voltage for simulation, and final voltage.
 - ix. **incr_step** : Step increment voltage.
 - x. **name** : Name of the folder where the files will be stored. Files are stored in C:/Simulation/%name%, where %name% is the name you enter.

How to enter data in a array?

```
double ea[layers] = { 3.0 }; //for a single layer device
double ea[layers] = { 3.0, 2.1 }; //for a two layer device
double ea[layers] = { 3.0,2.5,2.1 }; //for a three layer device
```

Name of the folder should be written within double quotes:

```
char *name = "problem1"; //[Don't use underscore(_) in the name]
//Every statement must end with semicolon(;)

```

3. The parameters are now entered. To run the simulation, press '*F9*', or select '*Compile*' and then '*Run*' from the '*Execute*' menu.

4. If the simulation is successful then the applied voltage and the number of iterations taken for simulation are displayed. In case the simulator diverges for the applied voltage, then “*many iterations, possible nonconvergence*” is displayed.
5. The simulation results are stored in the directory C:/Simulation/%name%. Contents of this folder are:
 - a. Sub-folder, ‘temp’ stores the values of the parameters of the simulation for the last applied voltage. This folder is not of any use.
 - b. ‘device_par.txt’, stores the device parameters, which may be helpful for referring the data at a later stage.
 - c. ‘IV-char.txt’, gives you the result in two columns. The first column is for the voltage, and the second for the current.
 - d. Other files are named as ‘device_char_%voltage applied%.txt’. Where %voltage applied% is the voltage for which the simulation was carried out. The columns from left to right are:
 - A. Position1(for Jn, Jp, J)
 - B. Electron current density(Jn)
 - C. Hole current density(Jp)
 - D. Current density
 - E. Position2(for hole and electron mobility)
 - F. Electron mobility
 - G. Hole mobility
 - H. Position3(for electric field)
 - I. Electric field
 - J. Position4(for Voltage, charge densities, recombination)
 - K. Voltage
 - L. Electron density
 - M. Hole density
 - N. Recombination(rate of exciton formation, assuming 100% exciton formation)