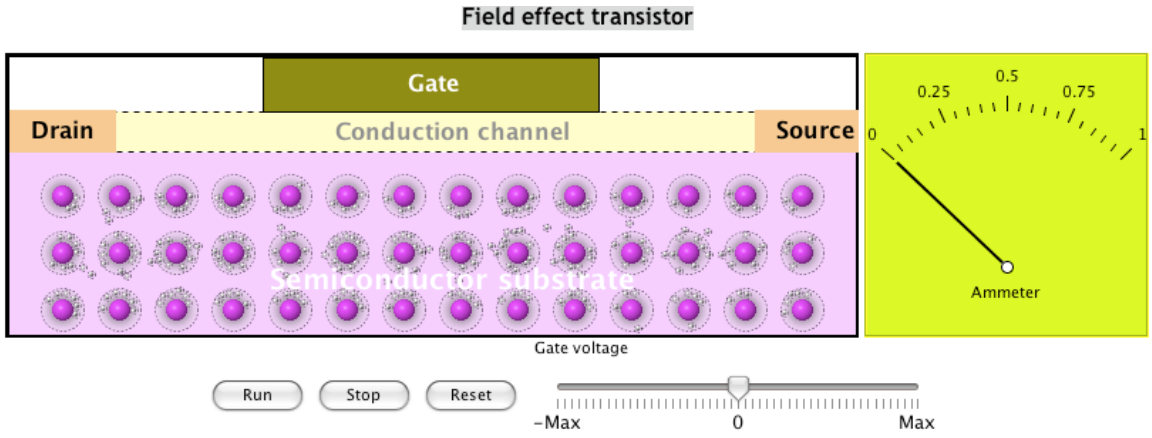
# Molecular Workbench

Model creation by and for beginners by Ravi Tavakley

## FIELD EFFECT TRANSISTOR (FET)




### Summary

A picture is worth a thousand words. Molecular Workbench (MW) brings complex molecular interactions to life on our desktops. MW models move and dance, they interact, and change to inform and entertain us. Learning complex ideas is as easy as playing a video game. The models are more than mere pictures. How are these models created? What follows is my experience, as a new user, of how to build a MW model.

### Introduction

Computer based modeling is vital to Nanotechnology. Macro level physical modeling is a low cost substitute for the real thing. At nano-scale there is no such substitute. As we study increasingly smaller events in electronics, physics, chemistry, biology, and genetics the use of computer modeling becomes increasingly more important. I am studying modeling as part of a Nanotechnology program at Dakota County Technical College, Rosemount, MN. My assignment in the simulation class is to become familiar with computer based modeling and to build a model. The tool of choice is the Molecular Workbench. I installed the Workbench on my Mac so that I could use it from anywhere. I played with many self directed

2-D, 3-D and Jmol models from the gallery. Then it was time to build my own. Although MW supports 2-D, 3-D, and Jmol and my eventual goal is to build any model type but this paper is based on my experience using the 2-D model.

## Discussion

I found the *Quickstart Guide for Authors* very helpful. It is a small part of a comprehensive set of MW documentation which includes the User's Manual, the Scripting language, Tips, Essays, and Tutorials. MW is patterned after the web browser technology. It uses a combination of Java scripts, Java applets, HTML and XML. MW also contains built-in tools for model builders. The combination of built-ins and WEB technology makes it possible to build models of any level of complexity.

The science behind MW and how laws of physics, chemistry, biology, genetics, nanotechnology, and material science are integrated into a flexible model building environment is the real "intellectual property". An expert model builder will combine the science, the full power of the tools, the development platform and the documentation. On the other hand a new user with little knowledge of the science can start with the built-in tools to build a simple model first and then add complexity to it. I chose the reverse engineering approach to learning as opposed to the more traditional approach of mastering the documentation first.

After looking at many models in the library I picked one that has everything I need. It has an input device, an output gauge (presentation & interaction), and directional flow of elements (science). Above all it shows physics in action through a device we take for granted in everyday use of electronics equipment. This 2-D FET model allows you to apply voltage to a gate, watch electrons flow from source to drain, and use ammeter to measure the resulting current. This model is simple by comparison to others in the library but sufficient as a starting point for me.

Elements used in the model, their behavior and look and feel is the first thing to strike you and therefore must be done right at the outset. The model has two visible elements. (Once made invisible an object disappears becoming impossible to find specially if it moves. Why I mention this here will become evident later). Selecting the right particles and setting their properties is straightforward. It is the bread-and-butter operation of MW accomplished by many of the built-in tools. The larger atoms representing silicon were made stationary. Smaller ones representing electrons were free to move about. Now comes the look and feel part of it. This is done by selecting the *Show view options* under the *Options* menu. This can also be reached by right clicking anywhere in the model and selecting *Show view options* from the

pop-up. There are at least a dozen options to choose from. Finding the right combination of options is a challenge. After some trial and error I got the right combination which included the *Ball and Stick* option among others. The silicon atoms appear with electrons hovering at various energy levels around a nucleus. Although somewhat lengthy this turned out to be the easy part.

I attacked the gate and the gate voltage next. This is when I learned the most important lesson of model building. Models are not true representations of real life but require some trickery and sleight of hand to make them appear real. Hidden behind an invisible box I found a *heatbath*. Due to lack of any other visible clues I assumed this to be the *Gate*. Association between the *Slider* marked *Gate Charge* and *heatbath* became my focus. This code in the slider holds the key:

*select element 2;*
*charge %val;*
*select element none;*

This code applies the charge value of the *Slider* tool to element# 2 in the model. The *heatbath* had to be element #2. I couldn't confirm that. This is when I had to call for help. I found that *heatbath* isn't an element at all and that I should look under the box marked *Gate* for element# 2. Makes sense. Under the cover; invisible? So the *Gate* turned out to be a set of eight #2 elements hidden behind the box marked Gate. I used the *What is this* or the *Help* tool to look at element information. I found the tool somewhat annoying as the information displayed vanishes too fast. There is also an option to display element index in the *View options* menu. The element # and element index are not the same. Element # is the type of element while element index is a unique number assigned to an atom in the model.

Adding an ammeter was easy. Attaching the current source to it was not. Clue to this association is the use of variable# 0 in the ammeter. The ammeter is a *Gauge* found under the *Insert* menu's *Model Output* option. Setting the gauge attributes is done through the gauge dialog. The model interchanges information with the outside world through eight channels and predefined variables. Channel variables are numbered from 0 to 7. Where and how is the channel variable set? This job belongs to Tasks. A Task Manager (available under the *Options* menu) can view, add, and modify tasks to the model. These tasks are executed at specified intervals when the model runs. They can access model's state information and pass them on to the external devices with or without computations. A task appropriately named "Current" calculates the *current* in the FET model. Here is how it does it:

*set %current count(0, 0, %rectangle[0].y, %width, %rectangle[0].height);*

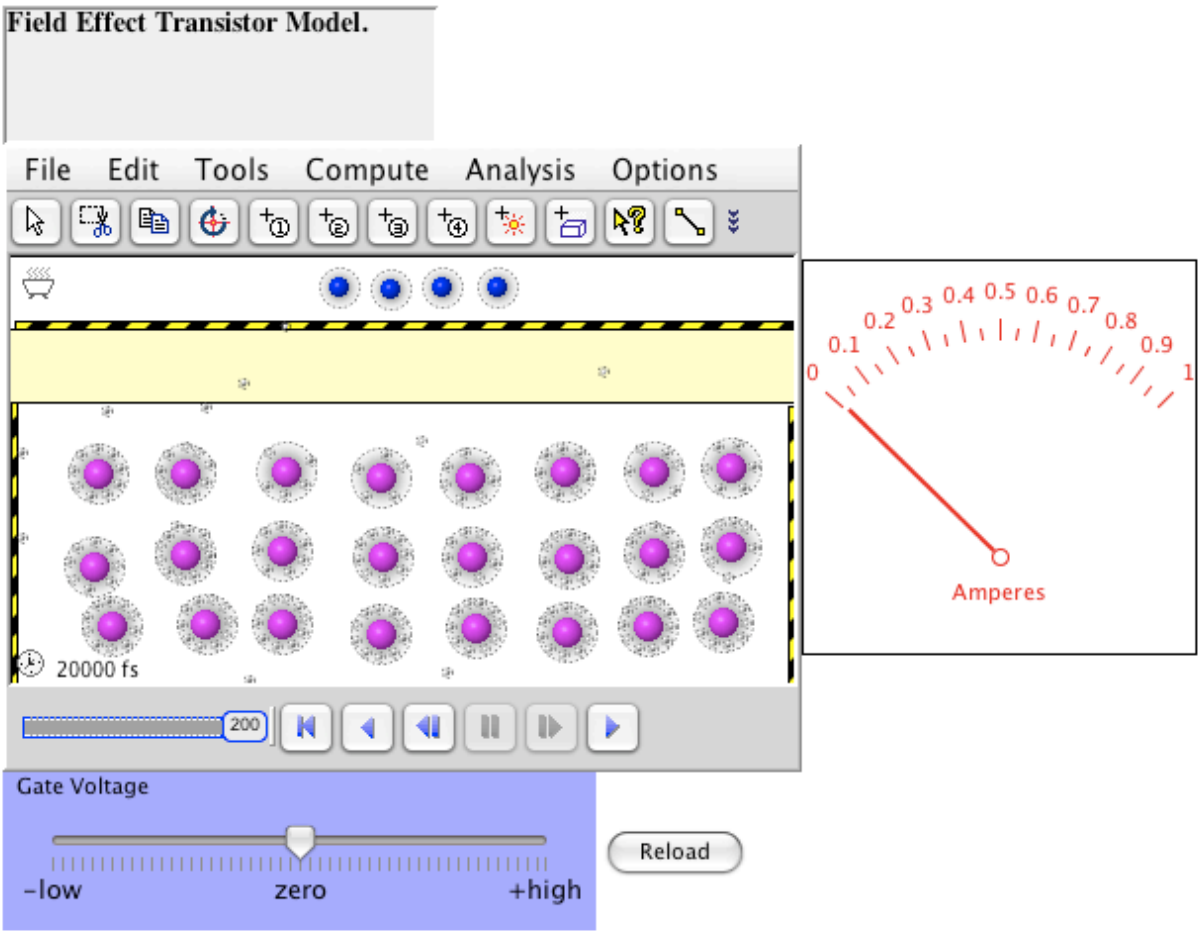
*set %current 2*%current/%number_of_atoms;*

*store 0 %current;*

As you see from this code *current* is the number of atoms in a box with given coordinates and size. This is the box labeled *Channel* in this model.

The last attribute to complete this model is the directional (Source to Drain) flow of atoms. This motion has to be restricted to the channel area. One would assume that the Source and Drain would have something to do with this but they don't. Source and Drain are inert boxes with labels but no active role. The directionality comes from the *Boundary conditions* of the model. *Boundary* option is available in the *Model Properties* menu. Some reading and some help from the support team was required to clear this up. *Boundary condition* in this model is set to X-periodic Y-reflectory. The boundary encloses the entire model area. Boundary outline is not easy to see either. It is made visible when you leave the menu after making a Boundary selection. This is the only time when a Boundary outline can be grabbed and resized. Since the Boundary covers the entire area of the model, what restricts the directional flow to the channel area? This is another thing that is not obvious at first glance. Perseverance, trial and error, and a little intuition paid off. As it turns out boxes in MW have some properties grouped under the *Physics* option. They can be *Reflective*, *Magnetic*, *Electric*, or *Magnetic*. Setting the Channel box *Physics* to *Electric* did the trick. I am not sure how or why this combination works this way but it works. Confinement borders were also added to prevent elements from escaping through the gate and the sides.

Now I have a working model. See Figure below. Rather crude looking but functionally complete. I can change the gate charge and watch the electron flow and ammeter needle swing to my delight, Missing cosmetics are deliberate but easy to add. It took me just a few days to go from nothing to a working model. In the end it was very satisfying.




## Conclusions

MW is a wonderful modeling tool. I didn't have to learn any equation in physics, chemistry or materials science to build the field effect transistor. The model does not use n- or p- type dopant nor energy bands. However, knowledge gained about model building is invaluable. There is no denying the educational potential of MW. All you have to do is watch any model in the library or the museum. Those with student activities are exceptional. I selected the reverse engineering approach of learning over the read-first approach. It took some trial and error, some reading, and two calls for help to complete the model. If I had taken the read-first approach I would still be reading. I would know a lot more but would have no model to show for it. Tradeoffs we all make. Even in this simple FET model some things such as those I mentioned in text remain for further examination and study, Finally I thank Charles (Qian) Xie for providing excellent support.