Introduction to Multisim & Ultiboard

(Lab by Wayne Stanton)

In this lab you will be utilizing your understanding of circuit generation/testing in Multisim in order to create the final project (figure 1), a TinyMatrix pendant. From this schematic you will be able to import your diagram into Ultiboard in order to design a Printed Circuit Board (PCB).

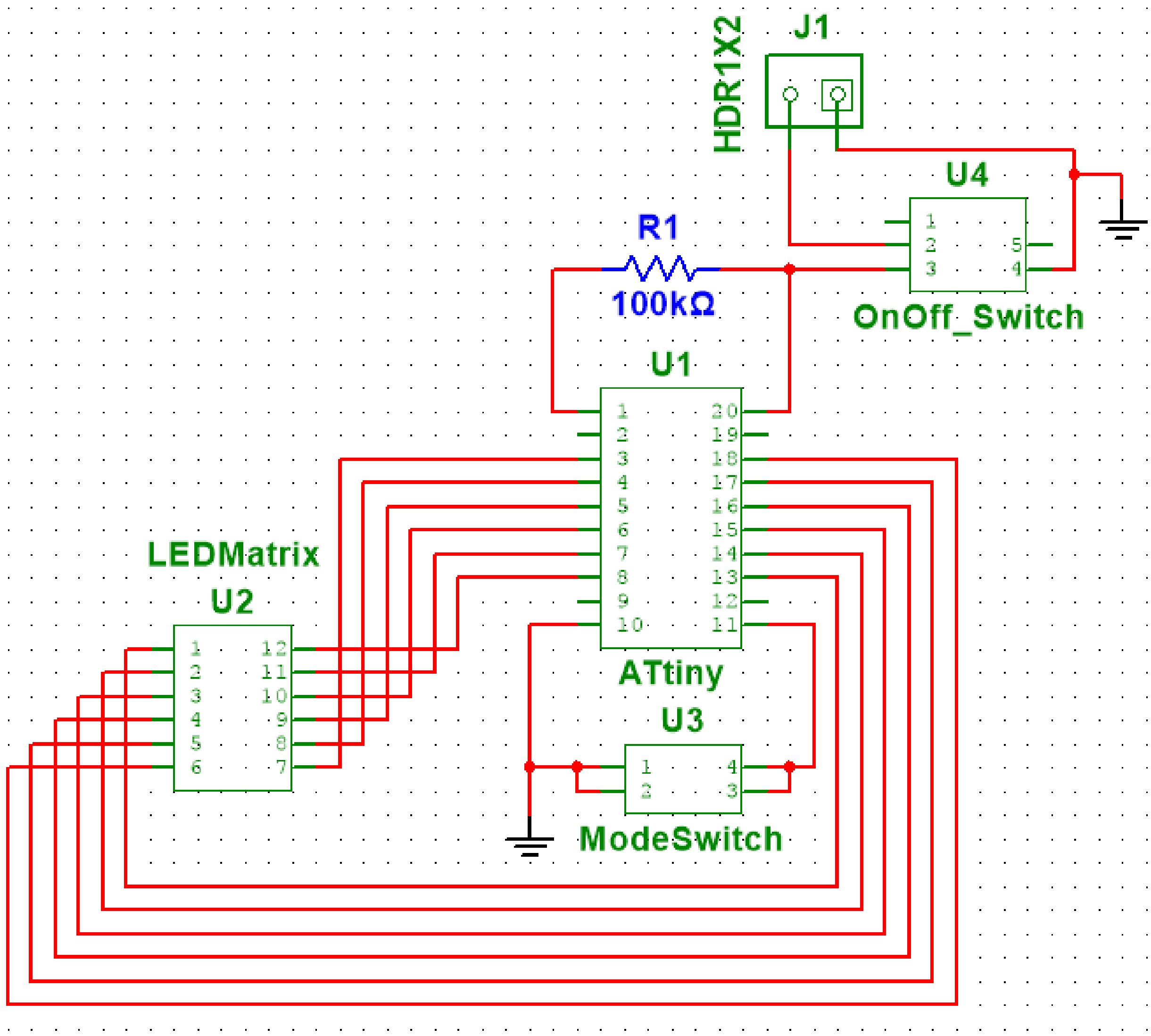

Figure 1: TinyMatrix Pendant

Lab Exercises

Part 1: Revenge of the Multisim

1. Once you have Multisim open you'll notice that when you try to place the components shown in figure 1 that they're not in the master database. This is often times the case when working on real world applications. For instances like this we can "cheat" by looking up the datasheet and finding the footprint for the chip. This footprint represents the physical layout, or pin placement, of the chip that will be placed on the PCB. For this reason it is very important that we have the correct footprint for the chip otherwise we cannot solder the chip to the board after it is made.

2. Since the components and their footprints are not in the Multisim database we must create each component and it's footprint to add to the database. We'll start with the microprocessor, the ATtiny.

    a. Go to Tools > Component Wizard. This is how we will create the chip and define it's footprint. For the *component name* we will begin with the ATtiny. Under *Author name* simply put your name. Finally click on *Layout only (footprint)*. *Next >*

    b. Now if we look at the ATtiny datasheet we find that the chip has a DIP-20 package type. For this case we're just going to simply select that footprint since it is already in the Multisim database. We *Select a footprint* and make sure we are in the *Master Database*. From here we can select the DIP20 footprint. *Hint:* clicking the Footprint or pin tab sorts them in alphabetical order or by number of pins which may make your life easier. After you've selected the footprint you should change the Number of pins in the component wizard window to 20 and press next.

    c. Press Next to skip through Component Wizard steps 3 & 4.

    d. Once on Component Wizard – Step 5 of 6 we must set the mapping information between the symbol and the footprint. This will be the step that links the pin numbering in Multisim to the pin placement in Ultiboard. We start by clicking *Footprint pins* for *Symbol pin 1* and will see a numbered checklist will show up. Simply click 1 and this assigns symbol pin 1 to footprint pin 1. Do this for all 20 pins. Symbol pin 2 goes to footprint pin 2, 3 to 3, etc. Once all the pins are done the Next button should become available.

    e. On step 6, the final step unless you did something incorrectly, we will be saving our component to the database. Here we click on the User Database on the left then press the *Add family* button. Here I chose *Family group: Misc* and *Family name: ATtiny.* Press Finish and place your component. SAVE! (Avoid spaces in your file name, Multisim and Ultiboard and temperamental)

3. Repeat the process in step 2 of the lab with the following information:

a. *Component Name:* LEDMatrix

   Author Name: Your Name

   Footprint: DIP12

   Number of pins: 12

   Family Name: LEDMatrix

   SAVE!

b. *Component Name:* ModeSwitch

   Author Name: Your Name

   Footprint: SKHH_1

   Number of pins: 4

   Family Name: ModeSwitch

   SAVE!

4. Finally, place the 100kΩ resistor, Ground, and the HDR1X2 (found under the connectors tab) as you have done in previously labs and SAVE! because we will be moving on to Ultiboard for the final On/Off switch. This switch has strangely placed leads so it's a good learning experience for you to understand how to not only link other standard footprints to your components but also how to physically create your own if the need arises (it will).

Part 2: Attack of the Ultiboard

5. Open Ultiboard either through searching through the start menu or *searching* through the start menu. Once it is open, create a new project, name it whatever you would like, and set the location to your personal space on the U: drive. Now you should see the vast black emptiness that is Ultiboard, scattered with white dots that will soon haunt your nightmares. Here we will begin making our On/Off switch.

6. Select Tools > Part Wizard and:

   a. Step 1: select *THT (through hole)* to begin making the switch footprint.

   b. Step 2: Select a *SIP (single in line package)*.

   c. Step 3: Select *Units:mm* and adjust the Package Dimensions to the following:

      i. *X: 8.60mm*

      ii. *Y: 4.40mm*

   d. Step 4: *Next >*

   e. Step 5: Set the *Drill hole Diameter (D): 1.00mm*

   f. Step 6: Set the *Number of pins: 5* and *Distances Between Pins (A): 2.00mm*

   g. Step 7: *Finish*

7. Assuming that the previous steps were followed correctly you should see a footprint that looks something like figure 2. Now that the footprint is finished we can go to *File >Save to Database > User Database* and name your component *OnOff_Switch* (you name it by typing the name into the *Existing parts:* box.. yes it is strange). Press *OK*. TAKE NOTE OF PIN NUMBERS, IT WILL BE IMPORTANT LATER!

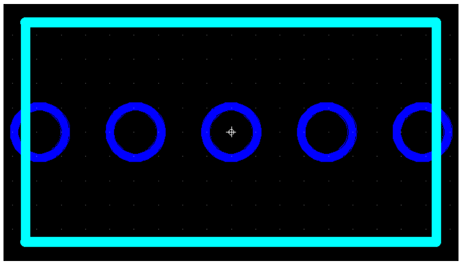

Figure 2: On/Off Switch Footprint

Part 3: Multisim Strikes Back

8. Now that we've created the footprint in Ultiboard we may now return to Multisim create and place the final component. Now things become a little trickier than before because if we take a look at the datasheet from Digikey we'll find that there are 5 pins but only the center 3 contain some electrical connection while the other two are to solder the switch housing structure.

   a. We set the name to *OnOff_Switch* as well as the author name and *layout only*

   b. We *Select a footprint* and now we will use the *User database* since this is where we saved our switch footprint. Select your newly made footprint to apply it to your new component and complete the component wizard as normal.

   c. The next thing that we must be wary of is the way the footprint was created in Ultiboard. If we remember correctly, you were supposed to take note of the pin numbers. When comparing the datasheet and the pin labeling in Ultiboard we find that pins 1 and 5 are the housing pins (not connected to anything) and pins 2 - 4 are the internal switch connections therefore on our schematic we should have nothing connected to pins 1 or 5 for the *OnOff_Switch.*

9. Now you can refer to figure 1 to draw the lines between components until your schematic matches. Once you've completed your schematic please have a lab TA check over your schematic before proceeding.

10. Now that we've completed the schematic and told Multisim what the components look like and what connections we want, we're finally ready to import it into Ultiboard. To do so select *Transfer > Transfer to Ultiboard 12.0.* Save your file and you should see Ultiboard appear.

Part 4: Return of the Ultiboard

11. When Ultiboard appears it will display the *Import Netlist* window that lists all the components and the connections between them. You shouldn't have to worry about this screen if everything was done correctly so simply press *OK.*

12. You should see your design as a jumbled mess of components on the screen with yellow lines going every direction denoting the connections between the chips (figure 3). We call this the "rat's nest". The yellow box is the keep-in boundary and represents the actual dimension of the finished PCB. For this case we want to make a small pendant so we will want to resize the board.

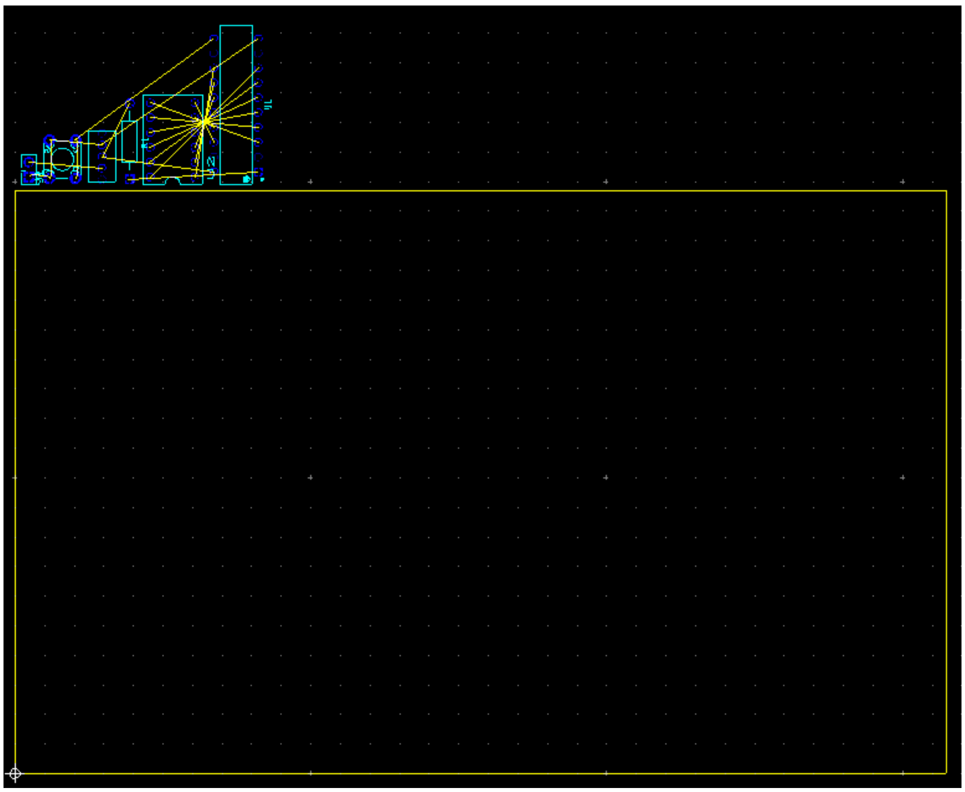

Figure 3: Rat's Nest

13. To resize the board simply right click on the boundary line and select *Properties > Rectangle* and adjust the board to *Width: 20mm Height: 75mm*. This will resize our board but ultimately give us less room to work with.

14. Next, we use the selection enable icons located in the top left corner so that we only have the part selection enabled, see figure 4. This allows us to turn on and off what components we can actually select. This is extremely helpful in more complicated circuits when the board becomes extremely cluttered.

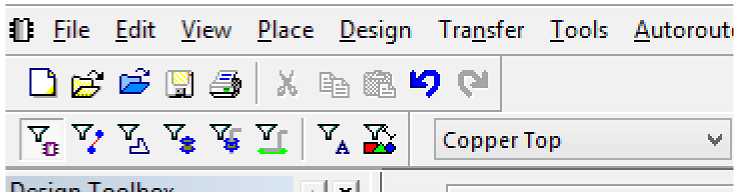

Figure 4: Part Selection

15. Move the components inside the yellow board outline and use CTRL+R to help you rotate them until you reduce the amount of crossing “rat’s nest” wires. Yours should look something similar to figure 5 although feel free to move components around however you would like. Make sure not to get the pins too close to the edge otherwise you won’t have room to route the traces in the upcoming steps.

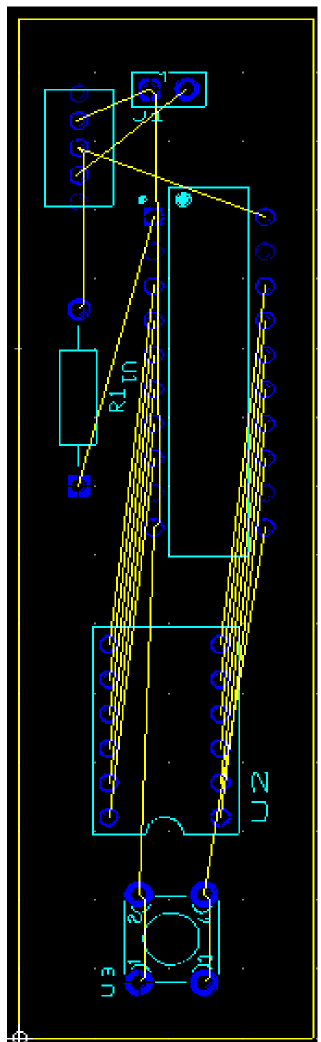

Figure 5: Untangled Rat's Nest on Board

16. Once the parts are arranged on the board, each yellow connection indicator must be replaced by a copper trace line (a conductive path on the surface of the PCB) to provide the electrical connection for each node. We can do this manually but this is only possible in a reasonable time on a relatively simple board. It can be done with this board but most likely in the future you'll be designing boards that are too complex for that. Instead, we will let the software route them automatically for us. Something to keep in mind first... the PCB is built of many separate layers. The board has a fiberglass base material (substrate) which provides the mechanical structure for the other layers. On each side of the substrate is a layer of copper which is where the conductive trace lines will be. These are sometimes called solder layers because they are typically coated with a thin layer of solder. Yes, there are two separate copper layers which is what allows trace lines to cross each other since they can run on two different planes with the substrate as an insulating layer between (more complex boards will have multiple substrates layered as insulation between additional copper layers). On top of both copper layers is a solder mask. This is the (typically) green colored layer that gives circuit boards their characteristic look. It looks like paint and insulates the copper from the outside world (except for where there are pads which are locations where parts will be soldered on). Finally, on top each solder mask there is a silkscreen layer on which we print labels, component outlines, and other useful indicators.

17. From the *Autoroute* menu we can select *Start/Resume Autorouter* and the yellow lines that designated the rat's nest will be replaced with beautiful green and red lines. Inspect how well the autoroute function worked and note that you can redraw the trace lines if you find that one looks awkward or strange.

18. Next we should start labeling the actual components on the board so that they can easily be replaced if need be and to make our lives easier when we construct the final product. First we will add some labels to identify the board itself, such as the name of the device and the designer's name (you). Then we'll add some labels to assist the user such as the component values and part numbers.

    a. Name labels - start by adding your name to the bottom layer.
        - We need to be sure we're working on the "Silkscreen Top" layer, so look in the drop-down box just above the workspace and confirm that's the layer displayed. If it displays anything else, switch it before you proceed.
        - Click the *Place Text* "A" button on the toolbar. The Text window opens.
        - Type your name in the *Value* box. After your name, it would also be good to add something to identify the class such as " EE 101" or similar. The default size in 'normal' font should be adequate.
        - Click *OK* and it will put the text on the cursor for you to place. Put it on the edge that you left open for labels.
        -

    b. User interface labels:
        - Check to make sure the "Silkscreen Top" layer is selected in order to add the component labels next to the chips.
        - Label the ATtiny 4313, the Mode Switch, the LED Matrix, the On/Off Switch, and the header. See figure 6 as an example.

c. After labeling, remove the part labels such as U2, U1, etc.
   - To do so, right click the label and go to *Properties* > *Attribute* and select *Invisible.* This ensures that only the relevant/important information that you added is visible (see figure 6).

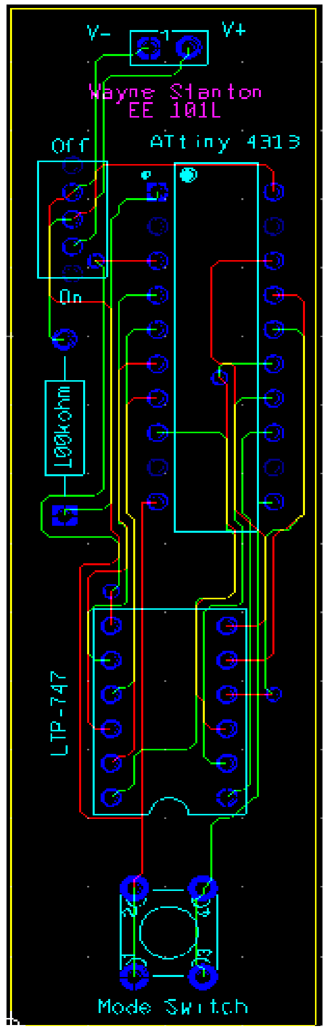

Figure 6: Final Board

19. PCB creation software is generally capable of creating an industry standard set of files called Gerber files. These are universally recognized by other PCB software such that designs can be exchanged regardless of platform. Each file in the set defines the layout of each layer, plus there's another for where the holes get drilled (for inserting the components, one hole per pad). Normally you would export your project to an array of gerber files for upload to your PCB manufacturer. However, since in this case your instructor will first create a master board containing the entire group's designs, the Gerber files will not be created until after that has been done.

    a. After you have completed your PCB please attach the file with the “.ewprj” extension in an email it and email it to your lab instructor.