1 Getting Started

Before you get started, you will need a PC or a virtual machine running Windows. Get an educational license from Altium from here:

https://www.altium.com/solutions/academic-programs/student-licenses

The Altium website will send you a verification email to your school address containing a link that you must click. It took about 5 minutes for Altium to verify my email when I signed up for a student license. After you verify your school address, they will send you a second email with instructions for activating your Altium account and downloading the installer. Follow the instructions in that email to download the Altium installer and install the program.

2 Creating a Project

2.1 Setting up Your License

Open Altium Designer. You will need to sign in to your account using the same email and password that you used to get your academic license on the Altium website. Once you’re signed in, you can activate your license by right-clicking on the license and selecting “Use” (see Fig. 1).

2.2 Creating a Project

On the top menu, click File → New → Project. This should open a dialog box where you can enter details about the project you want to create (See Fig. 2).

- The location should be set to “Local Projects” on the left side of the dialog.
- The Project Type should be set to PCB in the middle.
- The Project Name can be whatever you want. I’ve called mine ironon because I’m designing a board for the iron-on PCB lab. I have also created a new directory for my project where all the files will be stored (there will be at least 3-5 files associated with each Altium PCB project).
- Don’t worry about the Parameters.

After you’ve entered all the info for your new project, click Create in the bottom right corner of the dialog. This will create a new empty project. Each PCB project consists of one schematic file and one PCB layout file, both of which we need to create next.
On the top menu, click File → New → Schematic. This will create a new schematic file and put it into your project. Then click File → New → PCB to create a new PCB file. Now click File → SaveAll to save your newly-created schematic and PCB files in the same directory as your project.

3 Drawing a Schematic

The next step is to draw our schematic by placing components like resistors, capacitors, and chips in the schematic document and connecting them with wires. On the left side of the Altium window, select your schematic document. When you click on it, you should see a white schematic document with grid lines in the main part of the screen (see Fig. 3).

3.1 Panning and Zooming

You might want to move the schematic document around in the window. There are three ways to do this:

- Move your mouse’s scroll wheel up and down to pan the schematic window vertically.
- Hold down the Shift key on the keyboard and move the mouse’s scroll wheel to pan the window horizontally.
- Hold down the CTRL key on the keyboard and move the scroll wheel to zoom in and out.

3.2 Placing Parts

Once we’re in schematic mode, we can start placing parts. Click on the “Place Part” button in the top center of the screen. This will bring up the part search dialog on the right side of the screen. In the Place Part search box, type in ATTINY12L-4PU (the part number of the microcontroller we are going to use for this project). Below the search box, click “Manufacturer Part Search”.

The first result that comes up should have a picture of an 8-pin chip. Click and drag that picture onto your schematic document. It will show up as a yellow schematic symbol (see Fig. 4).

Next, we will place our voltage regulator, the UA78L05AILP. Use the same procedure as before to search for the part and place it on the schematic.

3.3 Setting Up Power and Ground

We usually don’t draw individual power wires to ever chip on our schematic document because that would make the schematic hard to read. Instead, we use power (\(\oplus\)) and ground (\(\ominus\)) symbols to indicate where power should be connected to each chip. Let’s connect the voltage regulator’s pins to schematic power and ground symbols.
The voltage regulator has three pins:

1. **IN** is the input voltage. It must be higher than the regulated output voltage, usually 9V or 12V.
2. **COM** is ground.
3. **OUT** is the regulated output 5V.

On the top toolbar next to the “Place Part” button, there should be a symbol for GND ($\downarrow$). Click on the ground symbol and place it right below pin 2 of your voltage regulator. After you’ve placed a ground symbol, you can press the escape key to stop placing ground symbols. If you don’t press escape, it will keep placing grounds every time you click on the schematic.

Now connect the voltage regulator’s pin 2 to the ground symbol by clicking on “Place Wire” ( $\rightarrow$ ) from the toolbar at the top of the screen. Draw a wire from pin 2 on the voltage regulator to the ground symbol. **Important:** when you draw your wire, make sure that you click on the tip of the pin, not in the middle of the pin. Press the ESC key to get out of wire drawing mode.

Use the same procedure to place 5V and 12V power ports next to the **OUT** and **IN** terminals of the voltage regulator and connect them with wires. You can access power ports by right-clicking on the GND symbol in the top toolbar. After you’ve placed a power port, you can double-click on it to change its name, say from +5 to +12. Your power supply should look like the one in Fig. 5 when you’re finished.

### 3.4 Hooking Up an LED

We are now going to place an LED and a resistor in our schematic. Both are generic components without specific part numbers, and symbols for them are available in the “Miscellaneous Devices” library. When you open up the Place Part dialog, you can select the Miscellaneous Devices library at the top (see Fig. 6). Place a component called Res1 and LED0.

Connect them in series to one of the port pins on the microcontroller as shown in Fig. 6. We will also place a capacitor (Cap in Miscellaneous Devices) to the power bus. **Pro tip:** to rotate a component, click and hold the left mouse button on it, then press the space bar.

Lastly, you might want to add a power jack to your power supply so you can plug your board in to a wall supply. You can find the power jack in the “Miscellaneous Connectors” library. It’s called PWR2.5. The final circuit layout is shown in Fig. 6. Make sure yours is connected the same way, and ask Neil to verify before moving on.

### 4 Laying out a PCB

After we have designed our circuit in the schematic, Altium can automatically transfer our design to a PCB. To transfer your design, click Design $\rightarrow$ Update PCB Document PCB.PcbDoc in the menu on the top of the screen. That will bring up a dialog box shown in Fig. 7. On the bottom left corner of that dialog box, click “Validate Changes” then click “Execute Changes.” Then close the dialog box.

Your main Altium window should show the PCB components for your design (pan the window around if you can’t see them). Click and drag your components into the black area of the screen. Electrical connections between components are indicated with thin lines. Try to arrange components so the electrical connections don’t cross each other. You can rotate components in the same way as in schematic mode.
• **Layers**: On the bottom of the screen, you can see a bunch of tabs that allow you to select which layer you are working on. We will have all of our components on the top layer, and all of our routing will be done on the bottom layer.

• **Units**: You can use either metric or SAE units in your board design. Metric units are millimeters (mm), and SAE are in *mils* (1 mil = 1/1000th inch). You can switch between units by pressing the Q key.

---

**4.1 Defining your Board Shape**

You need to define the borders of your board so the manufacturer knows where to cut it. You can do this using the Keepout Layer, which is one of the layers in the bottom tabs. Select the keepout layer at the bottom of the screen, then click *Place → Line*. Now click to draw a rectangle around your components. Make sure your keepout layer rectangle is a fully closed path (no holes). Press the Escape key to quit drawing lines in the keepout layer. You can see my board shape (drawn in purple keepout layer lines) in Fig. 8.

To define the board shape, select all four of the rectangle in your keepout layer by shift-clicking on them. Once they’re all selected, click *Design → Define Board Shape From Selected Objects*.

**Important**: there is some purple colored text next to the power jack that needs to be removed. Click on that text and delete all of it. We don’t want it showing up on our board.

**4.2 Design Rules**

Before we start routing our board, we need to adjust some design rules. These are limits on sizes and spacing between features on the board which are enforced by Altium. We are going to be making this board ourselves with the iron-on method, which is not able to produce fine details on the board. The traces should be wide and far apart from each other to make sure it can be transferred to the board correctly. For this design, it’s a good idea to keep your traces wider than 20 mils. We will define a design rule to allow wide traces.

In the top menu, click *Design → Rules...*. A dialog box like the one in Fig. 9 should show up. Navigate to the Routing Width (see the figure) and change the max width to 50 mils. Also change the preferred width to 20 mils. If you don’t do this, Altium won’t let you make traces any wider than 10 mils, which won’t show up on our boards.

You should also disable the “Silk to Solder Mask Clearance” rule by unchecking the box (see Fig. 10).
4.3 Routing Traces

Now we need to actually draw wires between pads on our parts. To make your first trace:

1. Select the bottom layer in the tabs at the bottom of the screen. It should be the second tab from the left, with a blue square next to it.

2. In the top toolbar, select “Interactively Route Connections” ( ). Once you click the button, your mouse cursor should turn into a crosshair.

3. Choose one of the pads in your design to start routing from, and hover your mouse crosshair over that pad. When your mouse is centered on the pad, the crosshair should show a small circle around the center of the pad. When your mouse is centered, click on the pad to start routing a trace.

4. You can now drag your mouse around, and the trace should follow your mouse. Click the mouse to make a corner. Route your trace to whatever pad it needs to go to.

5. Press the ESC key to get out of routing mode.

You can play around with the routing on your board. If you make a mistake, you can always select your trace and delete it. Make sure you route all of your traces on the bottom layer—the traces should be blue. My final routed board is shown in Fig. 11.

4.4 Checking Your Design
Before you manufacture a board, you always should run a design rule check to make sure there are no errors. This will check manufacturability (like trace widths, hole sizes, etc.) as well as verify that all connections have been made. Click Tools → Design Rule Check, then in the dialog that comes up, click Run Design Rule Check in the lower left corner. Altium will generate a report. Check to make sure there are no errors. If there are errors, fix them and re-run the design rule check.

5 Generating Masks

Now we are going to generate masks to make our board. These are basically just PDF files that will get printed onto a piece of glossy paper then transferred using a clothes iron onto the board. The mask forms a barrier between the copper on the board and our etchant solution, which prevents the etchant from dissolving copper underneath the mask. Only the places on the board where we want wires and pads will be masked. The rest of the board will be unmasked, and the etchant solution will dissolve the copper in those areas, leaving only our traces behind.
Figure 14: Scaled printing setup in Page Setup.

We need to generate a PDF file that is 1:1 scale. It’s important to make sure the scale of our print is correct because if the scale is wrong, our components won’t fit on the board. To configure scaled printing, click File → Page Setup and change the Scaling option to Scaled Print (see Fig. 12). Make sure the scale is set to 1.00. Also, make sure your Color Set is set to Mono in the Page Setup dialog.

On the bottom of the Page Setup dialog, click the Advanced button. This will bring up a window that allows you to specify which layers are printed. By default, Altium prints all layers on the same sheet, but we want them to be separated out. Make the following changes in the PCB Printout Properties dialog.

- Right-click on the Multilayer Composite Print and select Create Final (see Fig. 14).
- Check the “Mirror” checkbox for all layers in the list.
- On the bottom layer printout, right-click and delete all of the mechanical layers.
- Finally, click on Preferences at the bottom of that dialog, and de-select all mechanical layers (see Fig. 15).

Click OK to close out all of the dialog boxes. Next go to File → Print to print your design to PDF. This is the mask that will get transferred to your PCB. Show Neil your mask to make sure it’s right. Use a laser printer to print out your mask on a sheet of glossy magazine paper.

6 Making a PCB

7 Programming the AVR

7.1 Getting the Tools

On Mac OS, you can get the AVR C compiler through homebrew. If you don’t have Homebrew on your computer, there are installation instructions at https://brew.sh. Basically, Homebrew is just a package manager for Mac. Once you have homebrew, you can use it to install the AVR gcc:

neil@Neils-MacBook-Pro ~ $ brew tap osx-cross/avr
...
neil@Neils-MacBook-Pro ~ $ brew install avr-gcc
7.2 Writing A Simple Program in Assembly

A simple assembly program for AVR has a vector table and a program. The vector table must start at address 0, and the program can be located anywhere else. The ATTINY12's vector table is really simple (see Table 7.2). Each entry in the vector table contains one instruction that jumps to the interrupt vector handler. The only one you need is the reset vector, which should jump to the beginning of your program.

```
.org 0 /* Starting at address 0 */
rjmp main /* Reset Vector */
reti /* INTO Vector (Unimplemented) */
reti /* PIN_CHANGE Vector (Unimplemented) */
reti /* TIM0_OVF Vector (Unimplemented) */
reti /* EE_RDY Vector (Unimplemented) */
reti /* ANA_COMP Vector (Unimplemented) */

main:
    rjmp main /* Infinite loop */
```

<table>
<thead>
<tr>
<th>Vector Number</th>
<th>Program Address</th>
<th>Vector Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0x0000</td>
<td>Reset</td>
</tr>
<tr>
<td>2</td>
<td>0x0001</td>
<td>INTO</td>
</tr>
<tr>
<td>3</td>
<td>0x0002</td>
<td>PIN_CHANGE</td>
</tr>
<tr>
<td>4</td>
<td>0x0003</td>
<td>TIM0_OVF</td>
</tr>
<tr>
<td>5</td>
<td>0x0004</td>
<td>EE_RDY</td>
</tr>
<tr>
<td>6</td>
<td>0x0005</td>
<td>ANA_COMP</td>
</tr>
</tbody>
</table>