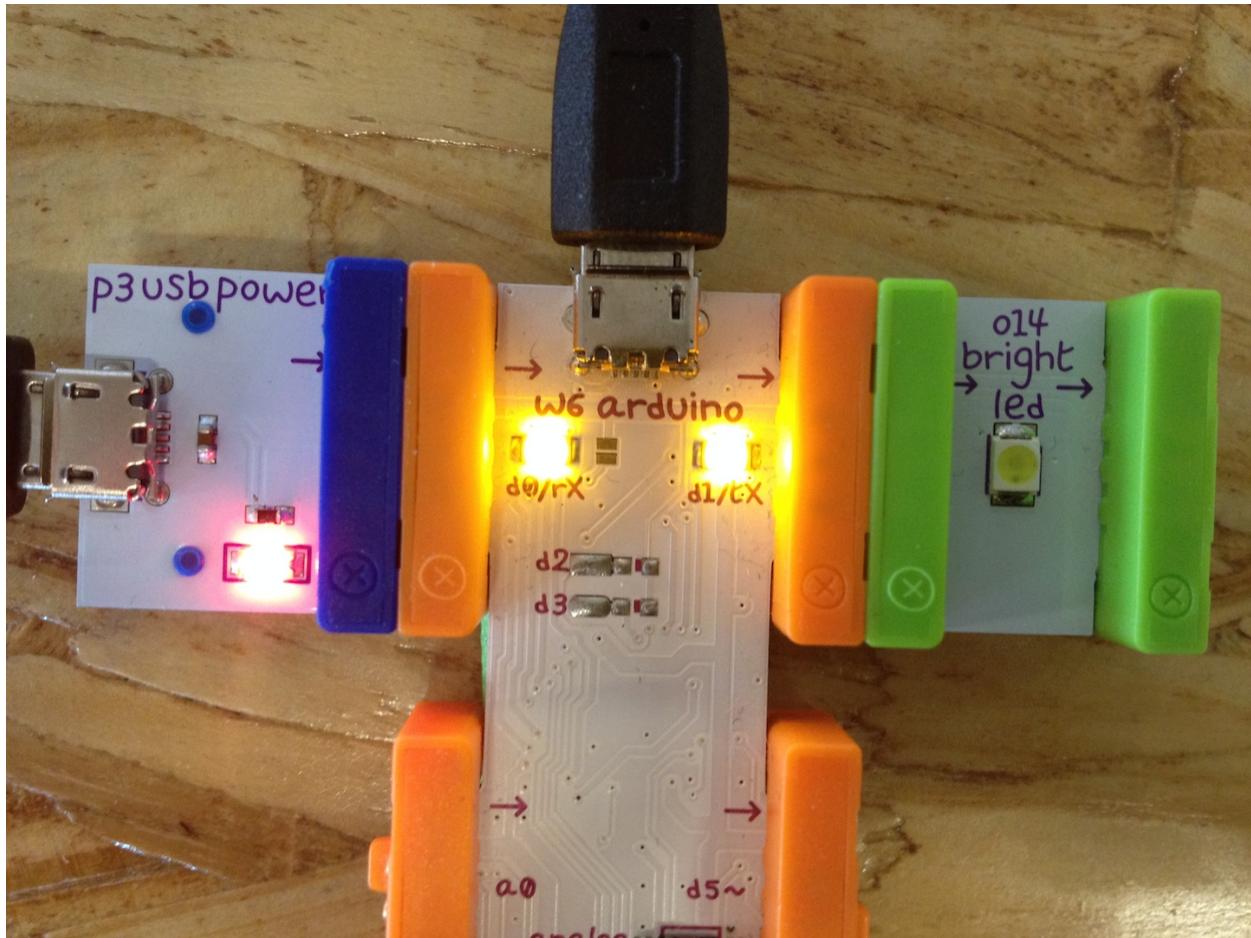
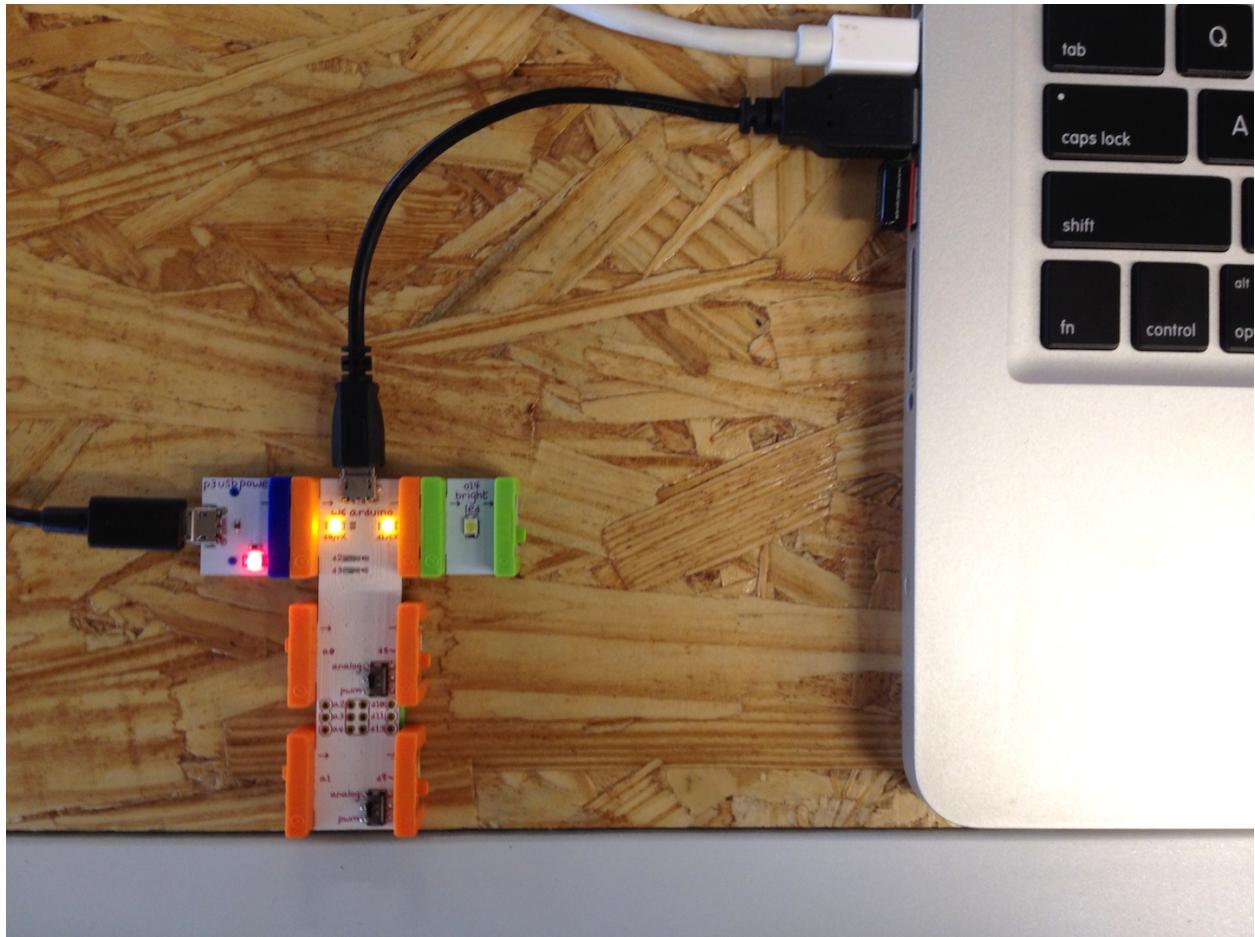


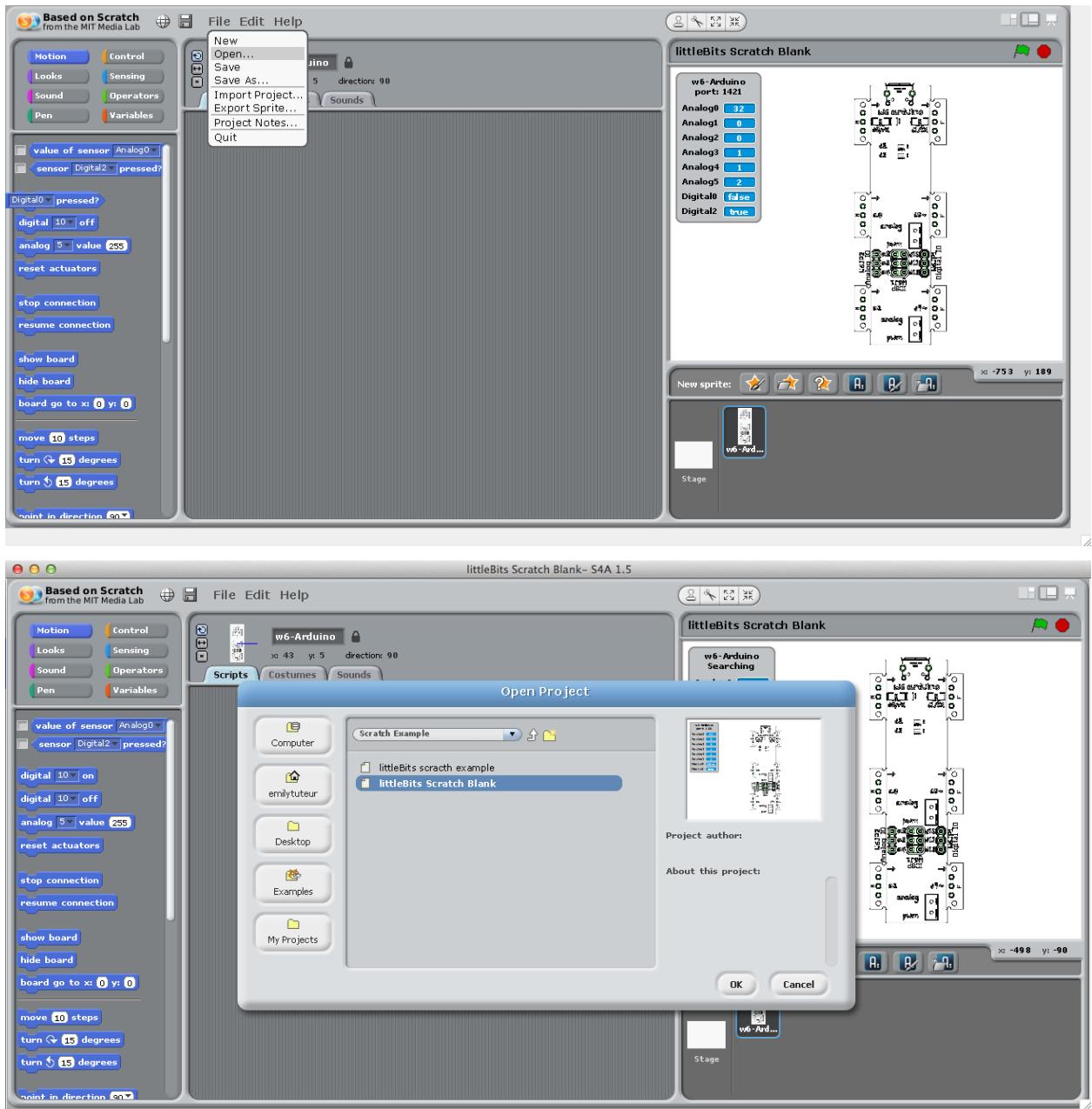
1. To begin, you will need to load our Scratch firmware onto your Arduino module.
2. Set up your circuit: power + Arduino + LED. The power module should be connected to the d0 input on the Arduino, and the LED should be connected to the d1 output.



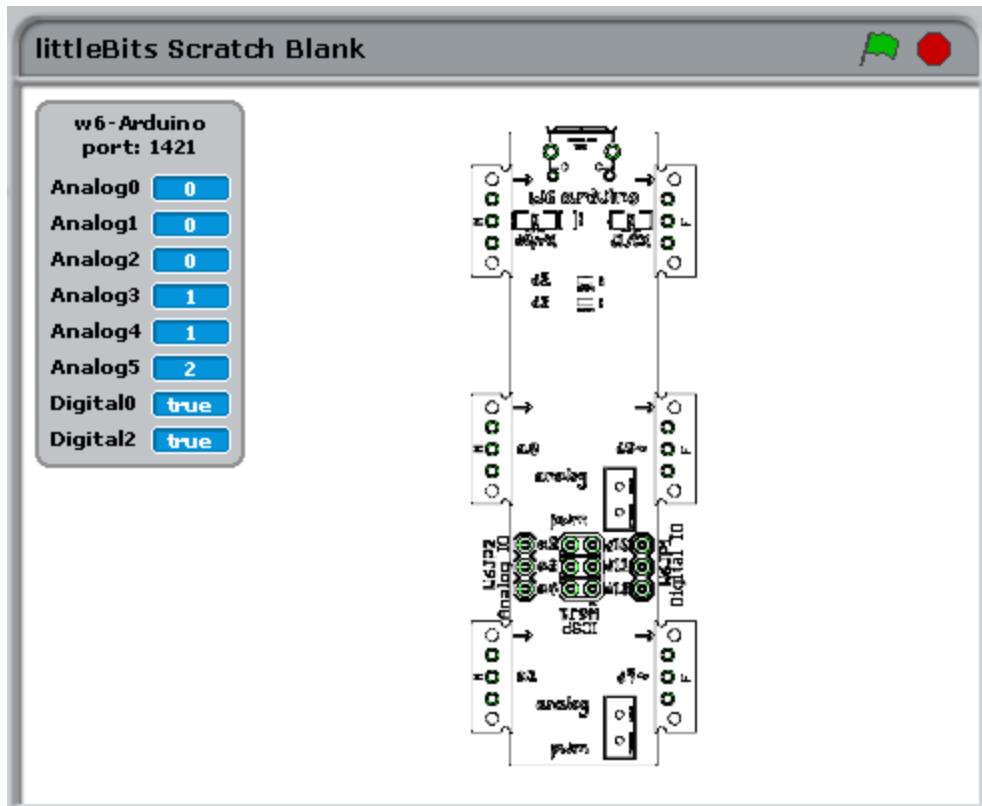
Make sure your Arduino module is connected to your computer with the microUSB cable.



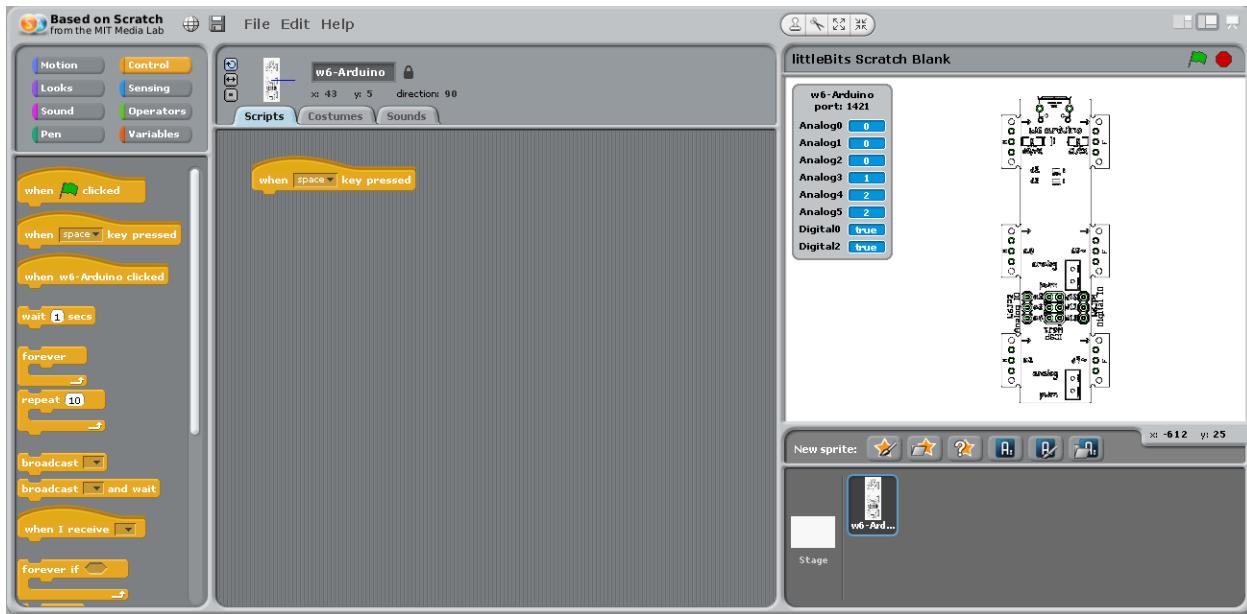
3. Using the Arduino IDE, upload the file provided in the dmg package: LB_S4AFirmware.ino. [Click here for full instructions for downloading the Arduino IDE and uploading files to your Arduino module](#). Note: rather than uploading the sketch listed, use the file we provide.
4. Open Scratch S4A
5. To start playing around, you will need to open the file “littleBits Scratch Blank.sb”. Click “yes” when asked if you want to close dialog. Then click “no” when asked if you would like to save the current project.



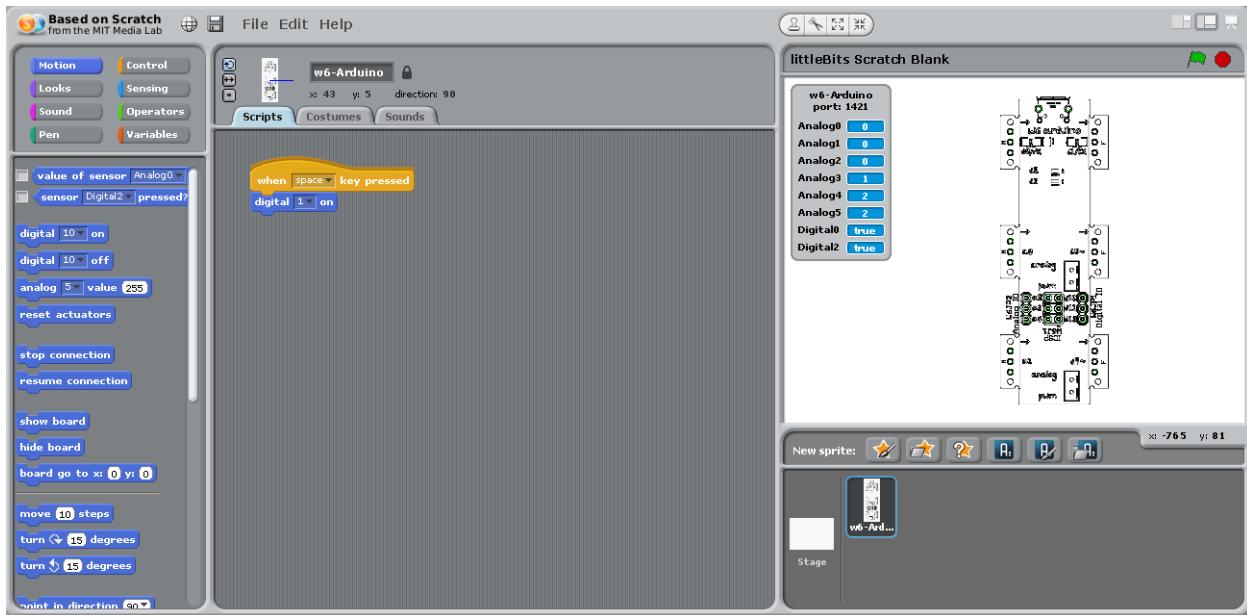
- On the right side of the Scratch environment, you will notice that there is a small version of your Arduino module pictured. You also may see a popup that says “searching for board”. If everything is connected properly, this popup will go away. If it doesn’t, try unplugging and replugging in the USB cable. You can also try powering your circuit on/off and checking that all your bitsnap connections are secure.

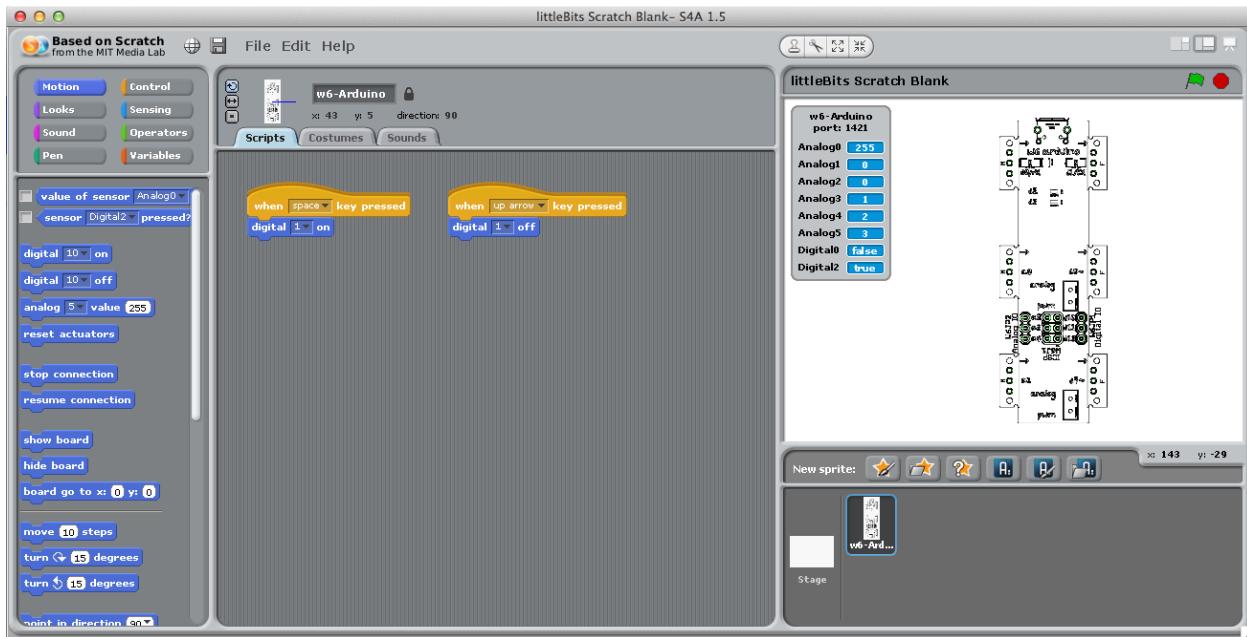


7. Now you are ready to start experimenting. We will walk you through a few basics. To begin, click on the “control” category on the left hand side of the screen. You will see a number of orange functions listed below. Scratch calls these “hats”. You always need to start with a hat that has a rounded top. Drag “when space is pressed” from the left column into the central square. This is setting up a rule that something will happen when the spacebar is pressed. You can also change the spacebar to arrows or letters by clicking another option in the dropdown menu.

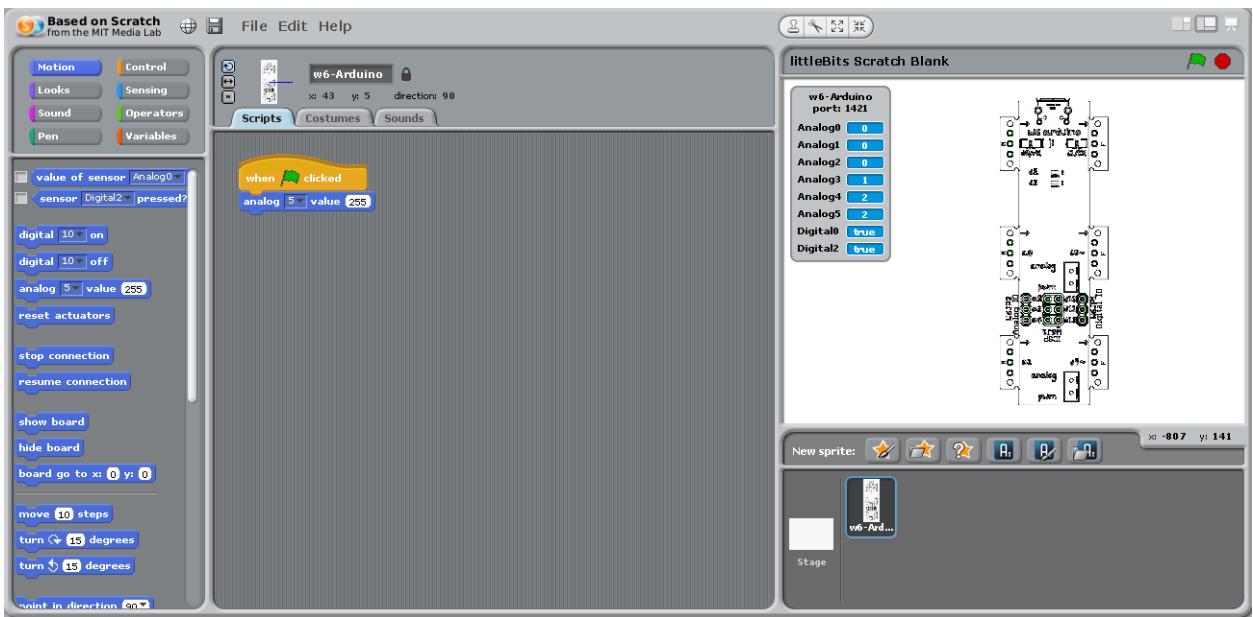
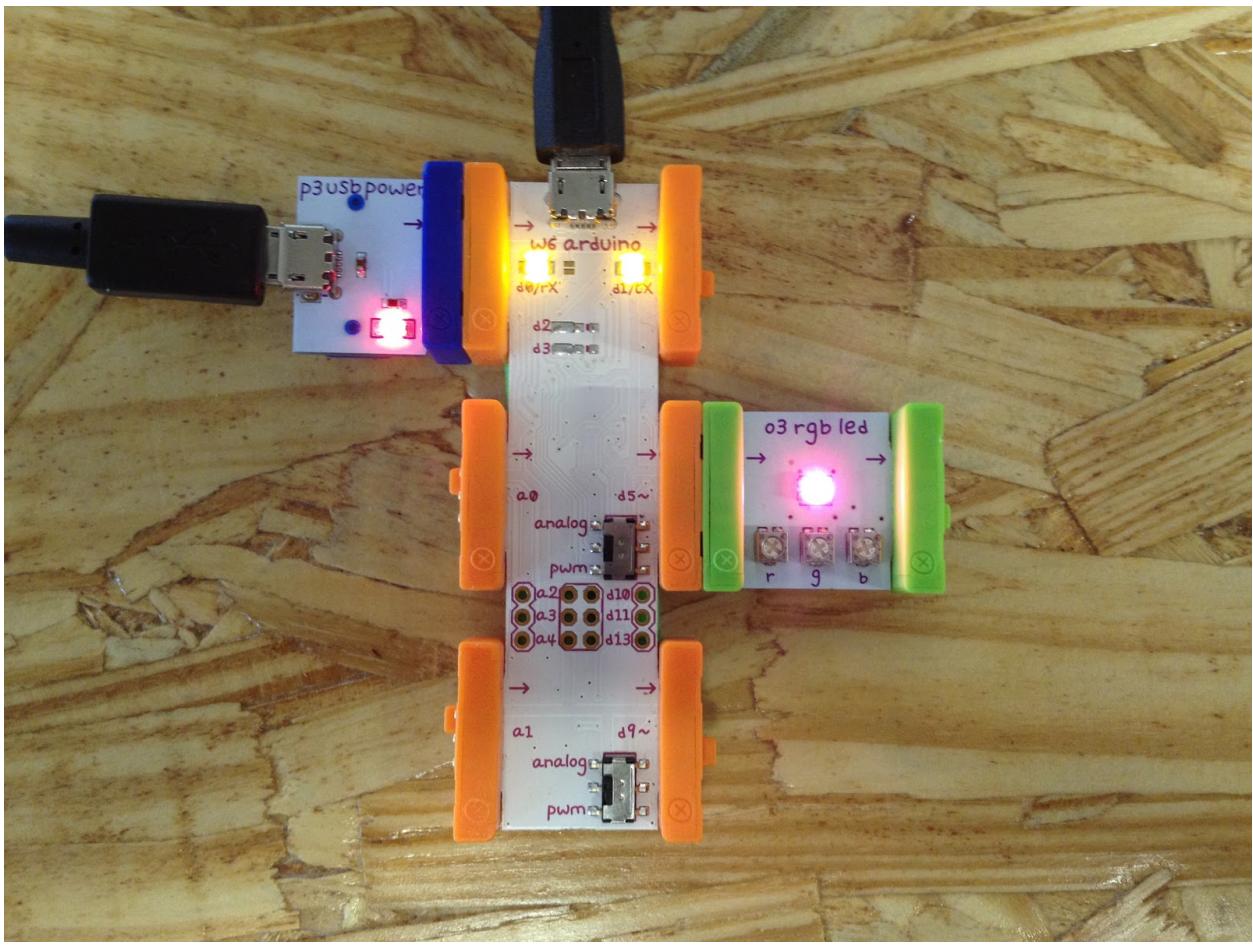


8. Now let's tell our circuit to do something in response to the spacebar being pressed. Click on the "motion" category and drag the hat that says "digital 10 on" to the central space. You will be able to snap this motion below your control. Because the LED is connected to the digital output bitsnap on the Arduino module (d1), change the number on the dropdown menu to "1". Now when you press the spacebar on your keyboard, your LED will light up. You can set up another function to turn your LED off as well using "digital off".

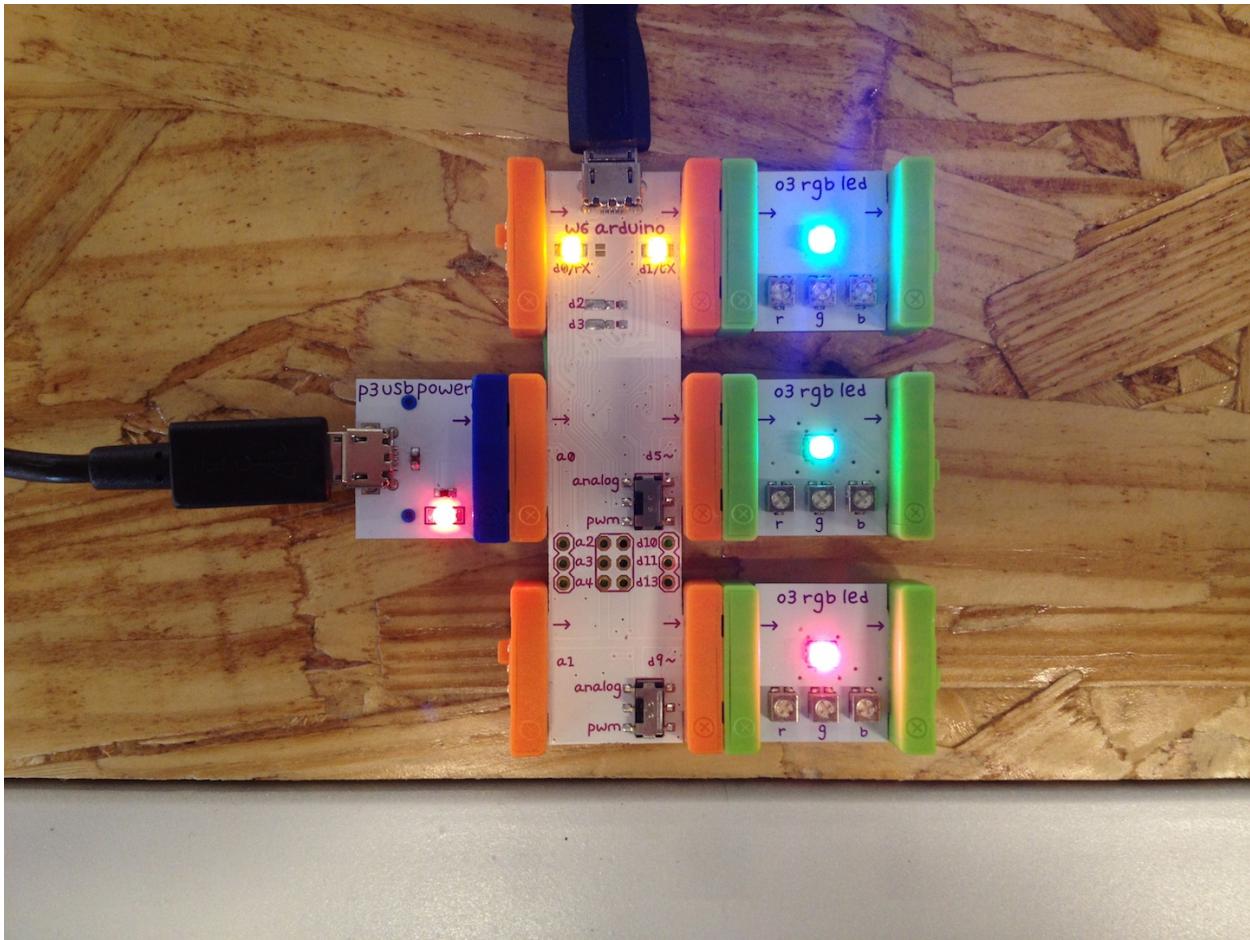


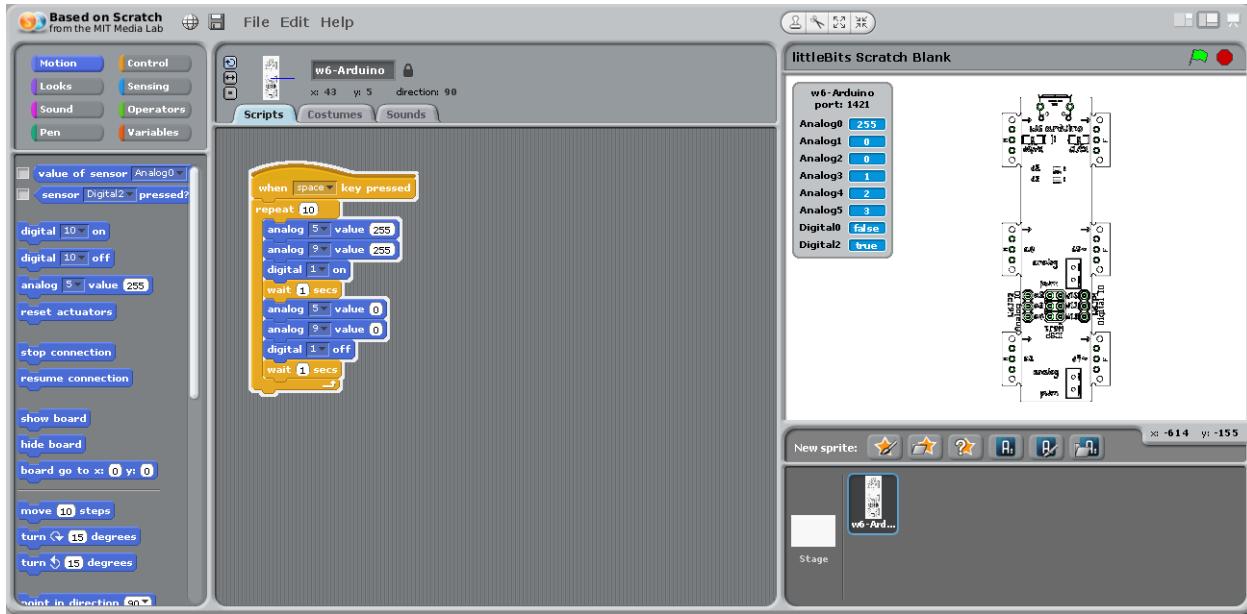


- Let's try another setup using the analog output. Select analog "5" as the output on the arduino module is d5. This is similar to the digital output, but this time you can change the value of the output. 255 is equivalent to 100% voltage, and 0 is equivalent to 0% voltage. Play around with changing the number.

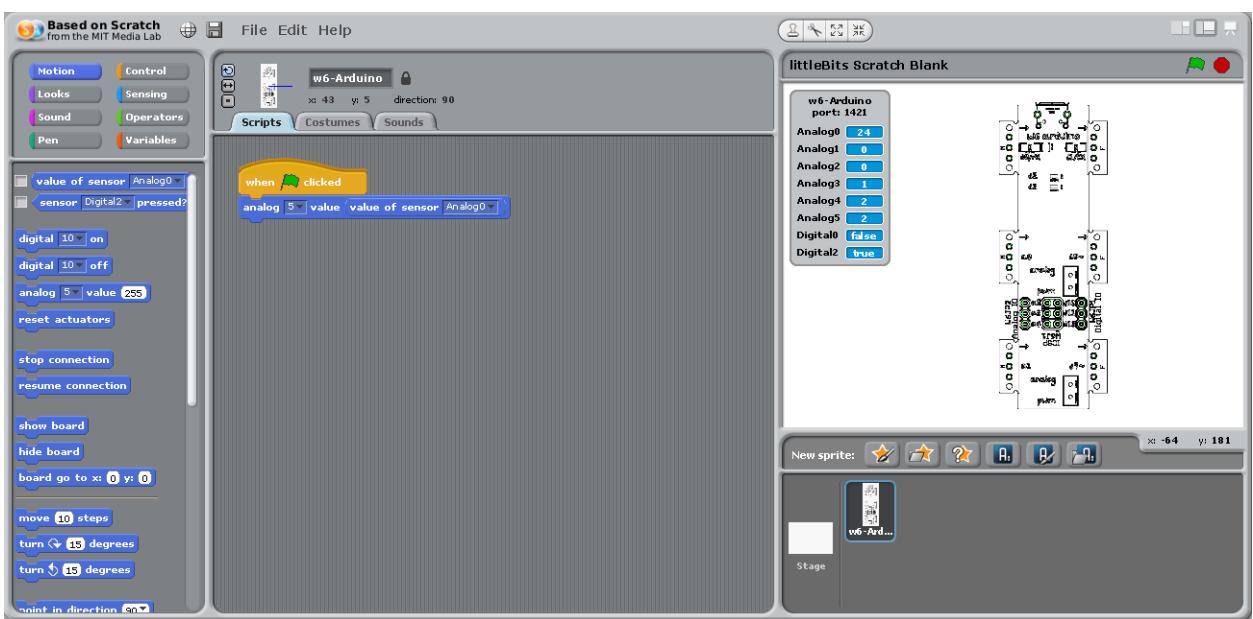
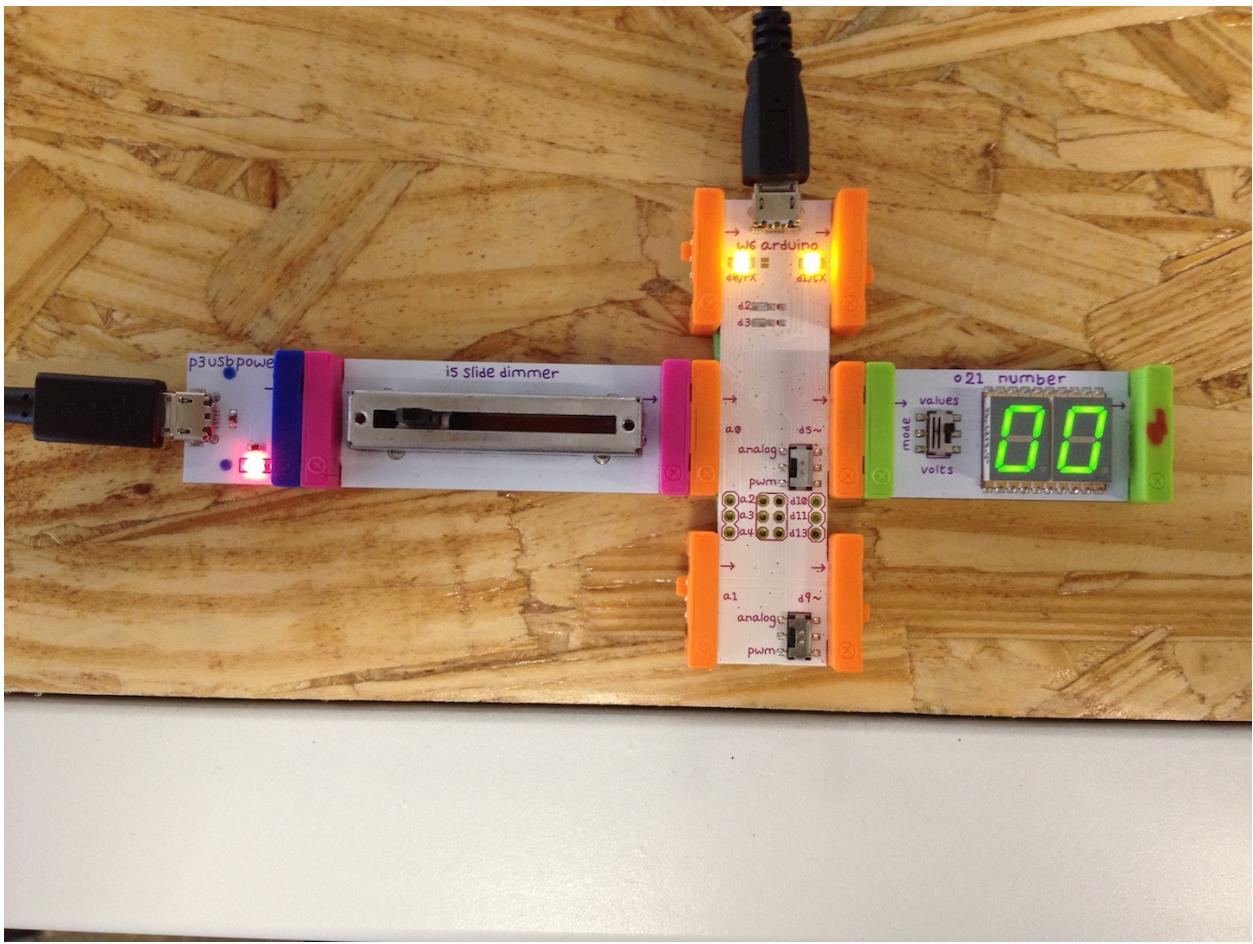


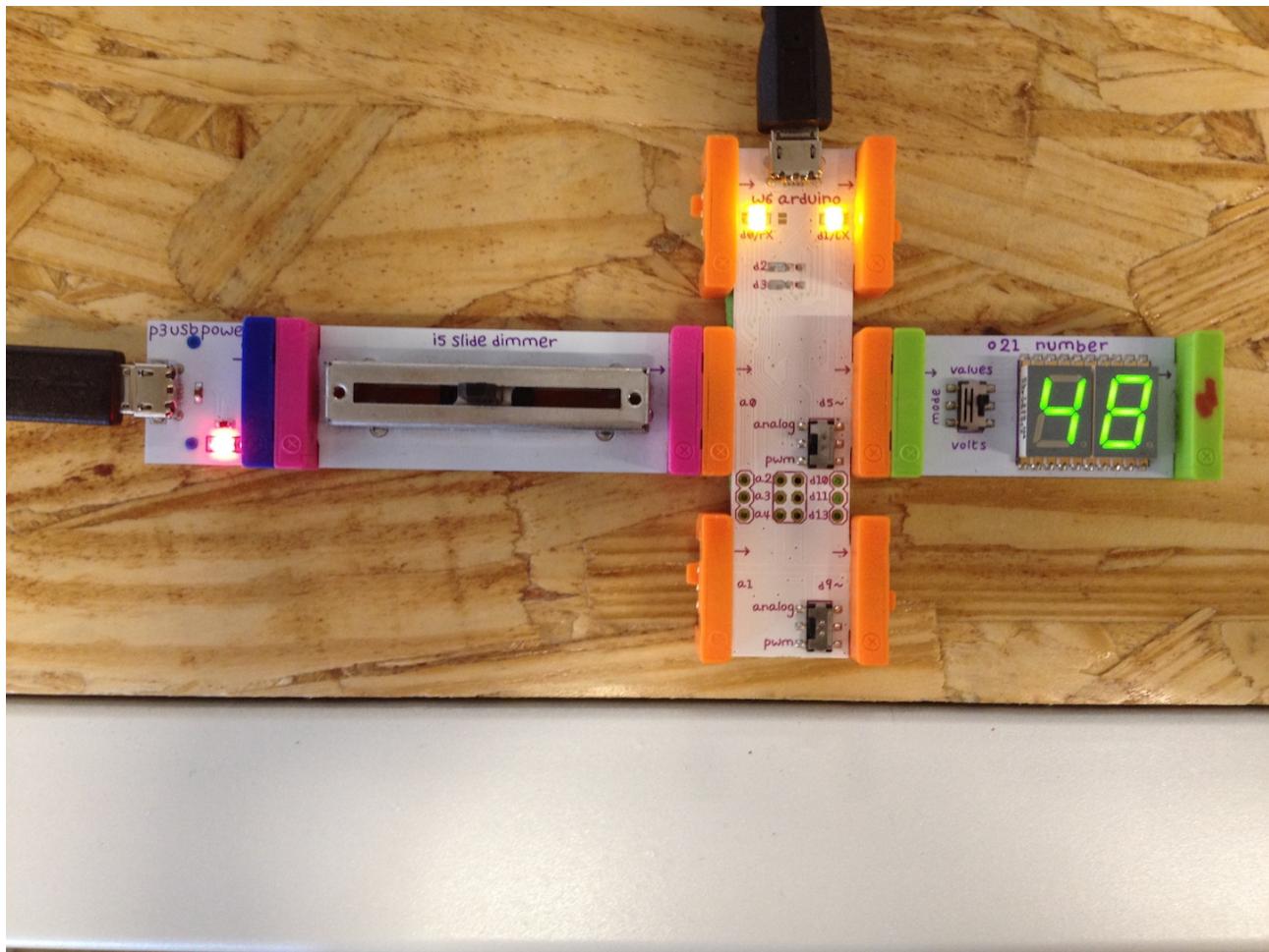
10. Add more outputs and try this set up. You will see your 3 LEDs (or other output modules) blink when you press the spacebar. Play around with the values and timing or add more actions to the sequence by adding more hats. Some examples and setups below.





11. If you want to add littleBits input modules to the equation, you can do that too! In the following setup, we are asking our number module to display an output based on the input voltage when prompted by clicking on the green flag.





12. These are just some basics but there are some very cool things you can do with littleBits and Scratch, like preprogramming the movements of a remote control car, using Scratch programming and wireless modules. You could also control a robotic hand using your keyboard. We can't wait to see what YOU come up with!