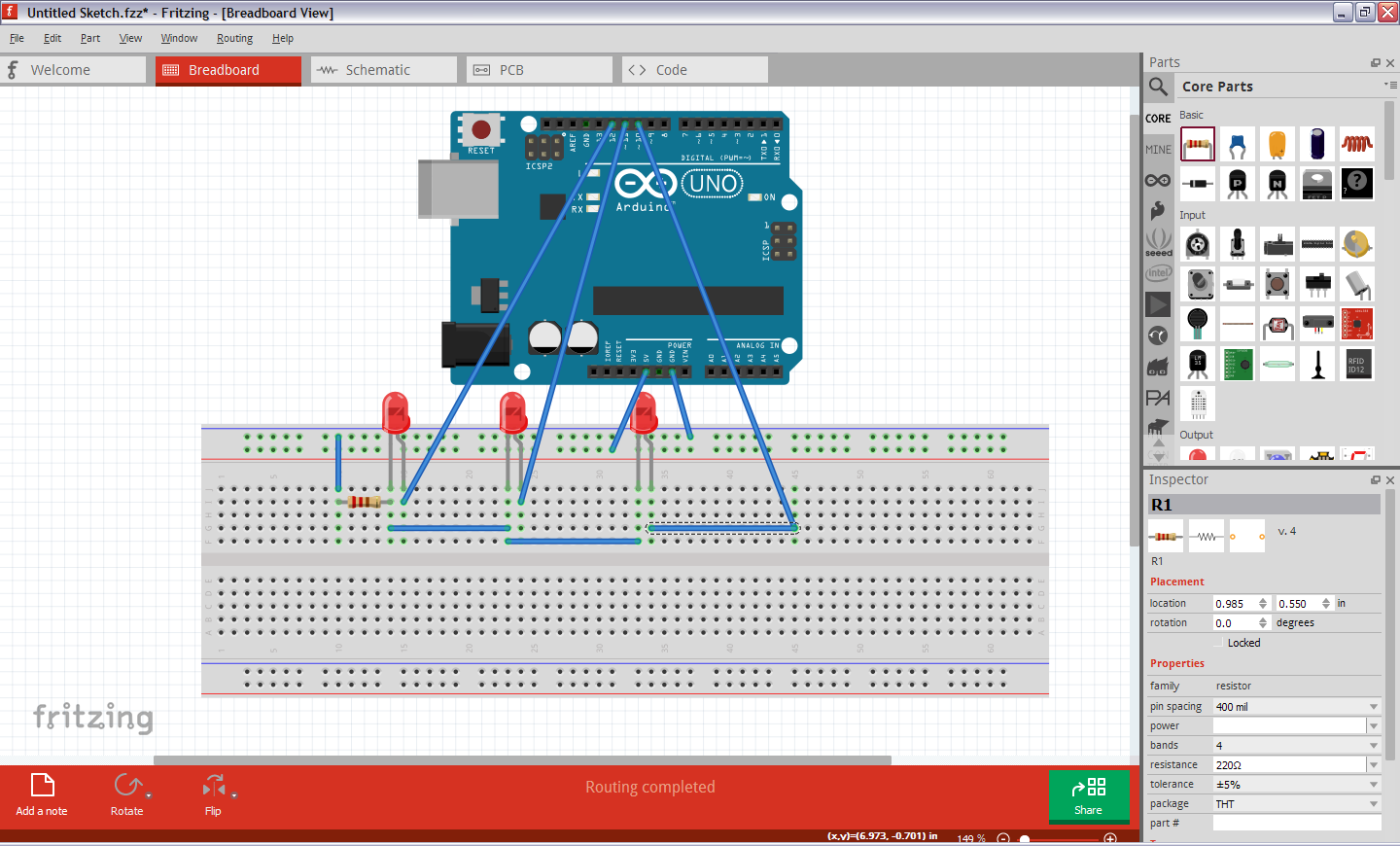
**Binary Dice**

In this activity, you will require the following:

* Arduino
* Breadboard
* 3 LED’s (polarity sensitive)
* 200 Ohm resistor
* Hookup wire
* USB A-B cable



Type in and run the following code:

const unsigned int LED\_BIT0 = 12;

const unsigned int LED\_BIT1 = 11;

const unsigned int LED\_BIT2 = 10;

void setup() {

// put your setup code here, to run once:

pinMode(LED\_BIT0, OUTPUT);

pinMode(LED\_BIT1, OUTPUT);

pinMode(LED\_BIT2, OUTPUT);

randomSeed(analogRead(A0));

long result = random(1, 7);

output\_result(result);

}

void loop() {

//To roll dice again, press the reset button on the Arduino

}

void output\_result(const long result){

digitalWrite(LED\_BIT0, result & B001);

digitalWrite(LED\_BIT1, result & B010);

digitalWrite(LED\_BIT2, result & B100);

}

**Now that you have this working, modify it to be a counter from 0 (000) to 7 (111).**

In case you are not aware, a 3 bit binary counter has 8 states (23 conditions). Specifically, 000, 001, 010, 011, 100, 101, 110, 111 where 0 represents an off LED and 1 represents an on LED. Hint: instead of sending a random number from 1 to 6 to the output result function, send it the numbers from 0 to 7 in that order.

FYI: result & B001 is an example of binary masking where only the numbers 1 (001), 3 (011), and 5 (101) will result in LED\_BIT0 being on. The same can be applied to result & B010 and result & B100.