

Arduino Board and Simulators

Lê Thê Dũng, Ph.D.

Dep. of Computing Fundamentals, FPT University, Viet Nam

January 2024



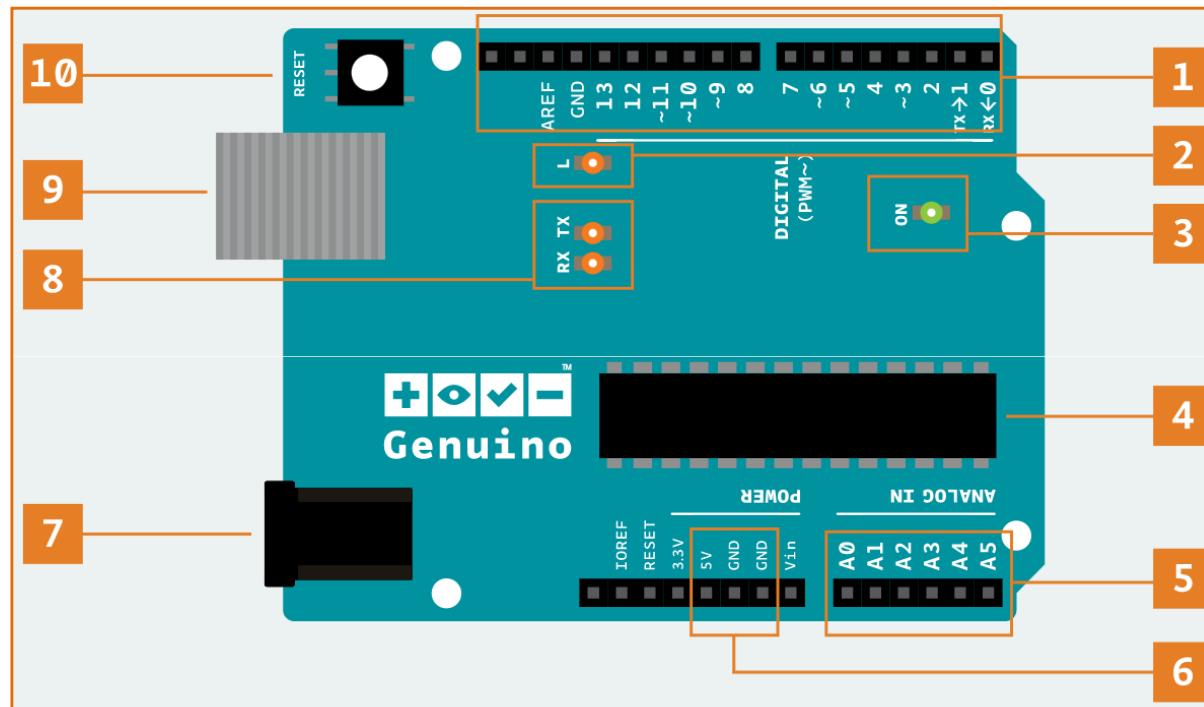
Outlines

- Anatomy of Arduino Board
- Fritzing Simulator
 - Interface
 - Starting a New Project
 - Arranging the Environment
 - Building a Circuit
 - Editing Properties
- Tinkercad Simulator
 - Create an Account
 - Create a New Circuit
 - Build a New Circuit
 - Program with Arduino
 - Simulate the Built Circuit

ARDUINO BOARD

Arduino Board - Anatomy

- Arduino boards sense the environment by receiving inputs from many sensors, and affects their surroundings by controlling lights, motors, and other actuators.
- Arduino boards are the microcontroller development platform that will be at the heart of your projects.



The anatomy of Arduino Uno.

Arduino Board - Anatomy (cont')

No.	Component	Function
1	Digital pins	Use these pins with digitalRead(), digitalWrite(), and analogWrite(). analogWrite() works only on the pins with the PWM symbol.
2	Pin 13 LED	The only actuator built-in to your board. Besides being a handy target for your first blink sketch, this LED is very useful for debugging.
3	Power LED	Indicates that your Arduino is receiving power. Useful for debugging.
4	ATmega microcontroller	The heart of your board.
5	Analog in	Use these pins with analogRead().
6	GND and 5V pins	Use these pins to provide +5V power and ground to your circuits.
7	Power connector	This is how you power your Arduino when it's not plugged into a USB port for power. Can accept voltages between 7-12V.

Arduino Board – Anatomy (cont.)

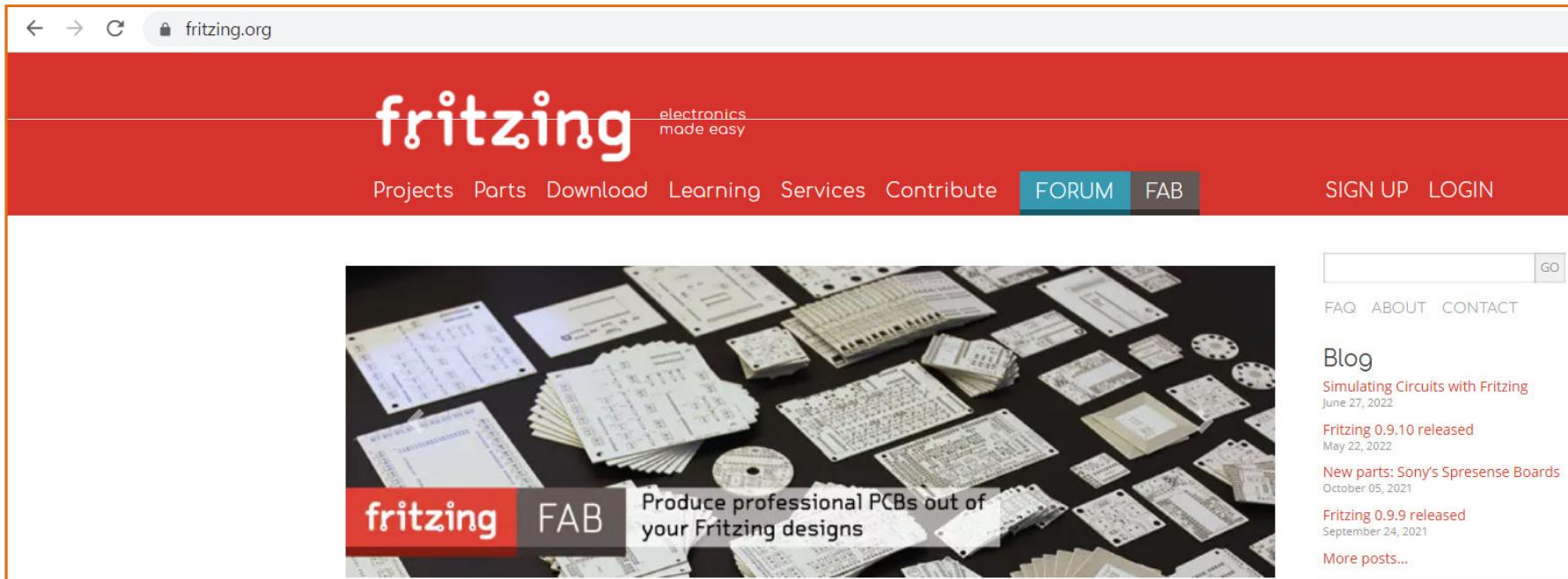
No.	Component	Function
8	TX and RX LEDs	These LEDs indicate communication between your Arduino and your computer. Expect them to flicker rapidly during sketch upload as well as during serial communication. Useful for debugging.
9	USB port	Used for powering your Arduino Uno, uploading your sketches to your Arduino, and for communicating with your Arduino sketch (via Serial.println() etc.).
10	Reset button	Resets the ATmega microcontroller.

ARDUINO BOARD SIMULATORS

1. Fritzing

Fritzing - Introduction

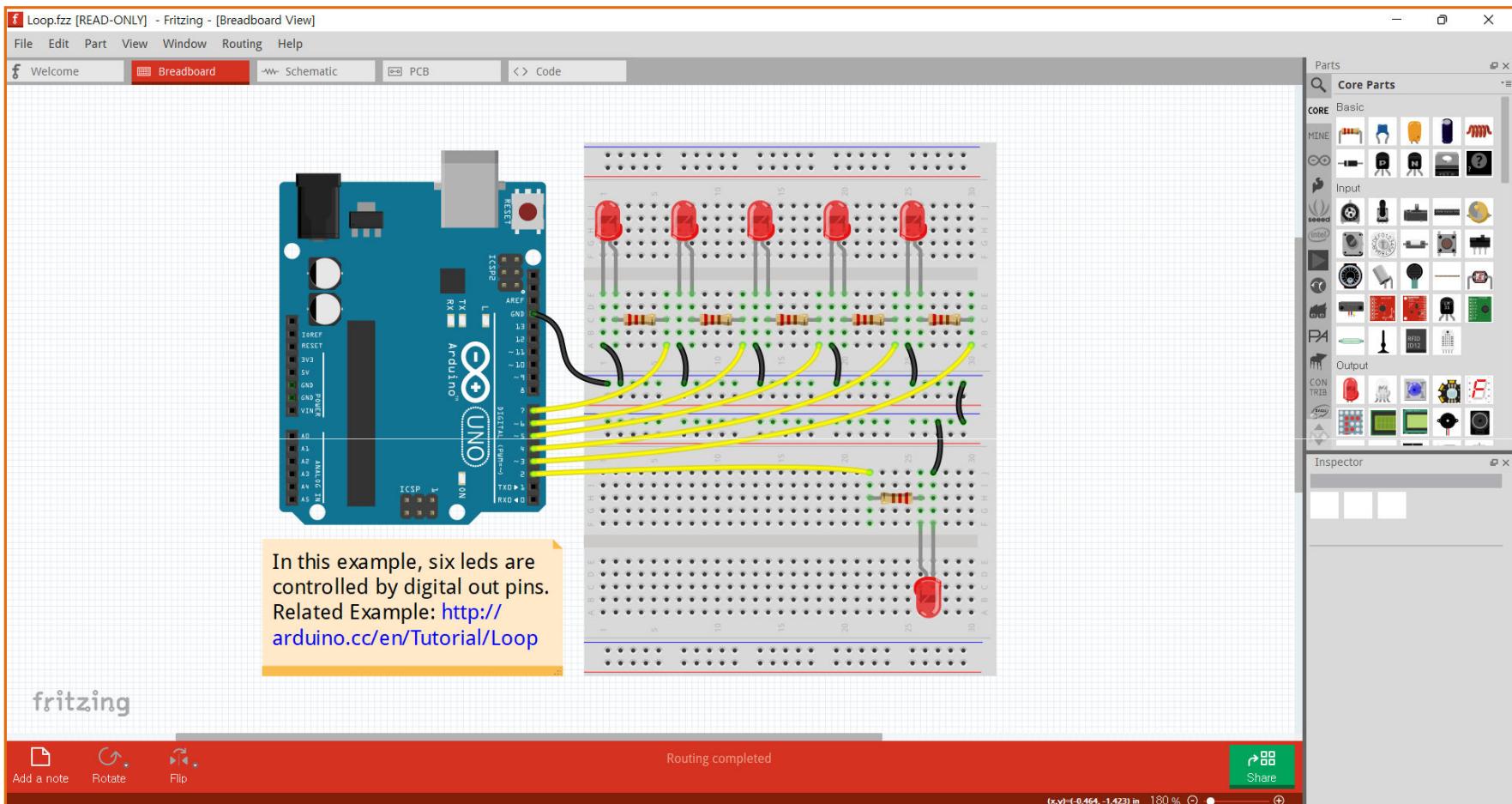
- Fritzing (<https://fritzing.org>) is an *open-source hardware initiative* that makes electronics accessible as a creative material for anyone.



- It lets you develop a virtual test board, utilize an auto-router in automatic mode or manual mode, and create a circuit diagram.
- You can add notes which are sharable with other users
- There is an online project gallery which is accessible to see the work.

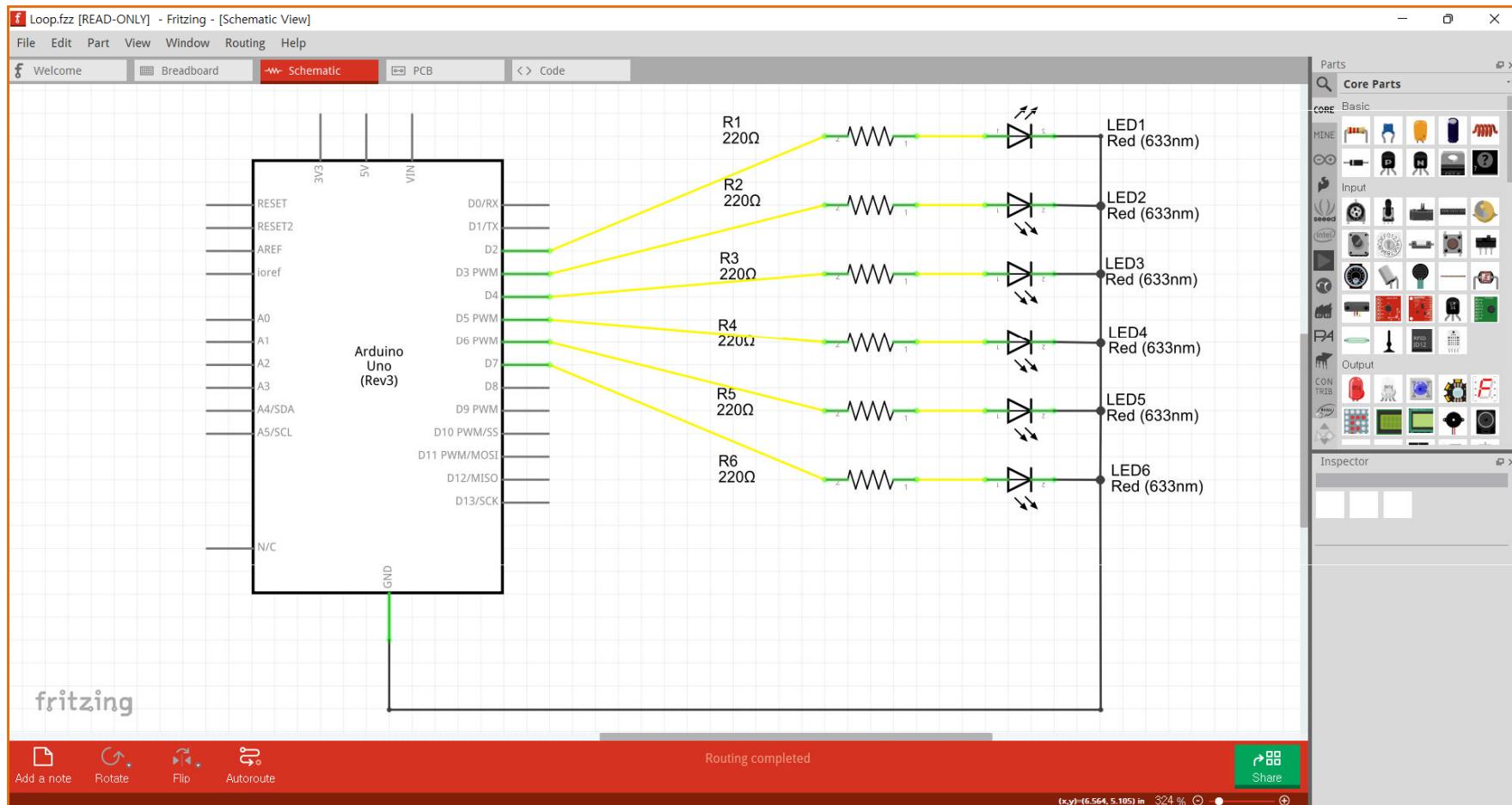
Fritzing – Interface (cont.)

- The interface of the tool has three accessible views available for starting a new project. They are as follows:
 - Protoboard View – Gives the test board view, so you can sketch the way you want the components to connect on the real board.



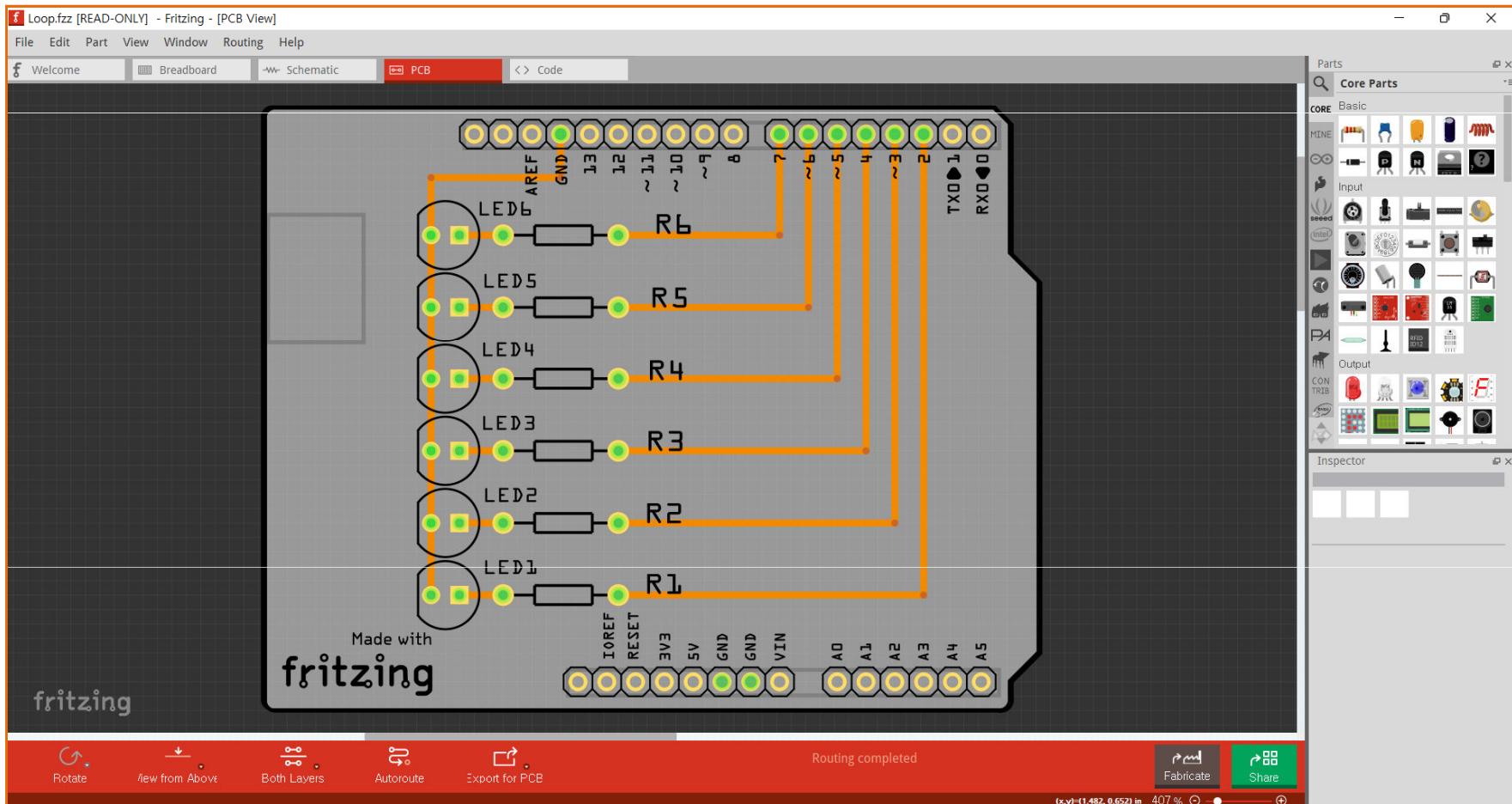
Fritzing – Interface (cont.)

- Schematic View – Uses symbols to represent the connections and components.



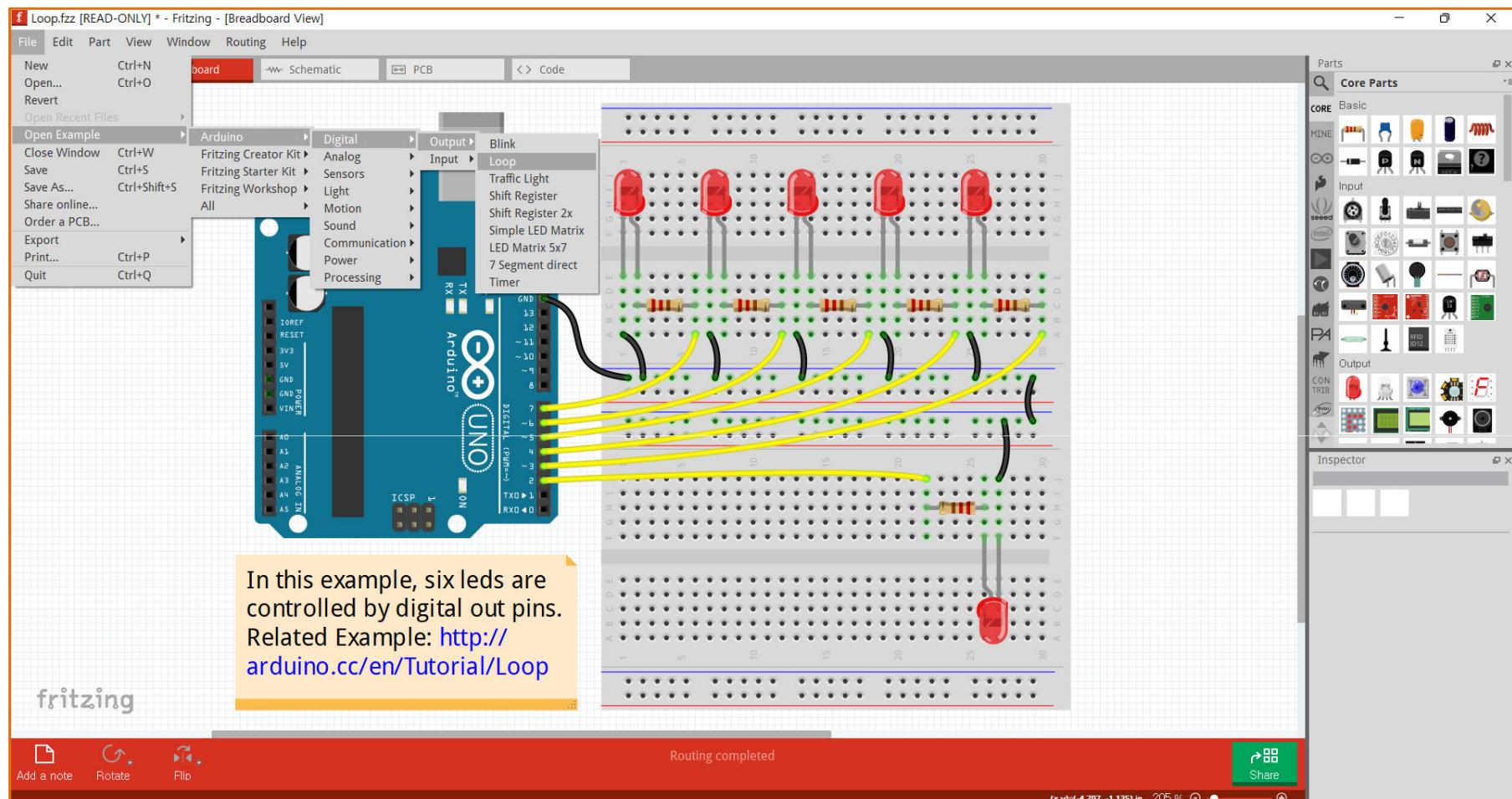
Fritzing – Interface (cont.)

- Printed Circuit Board – Lets you preview the way components get distributed on a PCB. Here you select the positioning and connections which are best for the components.



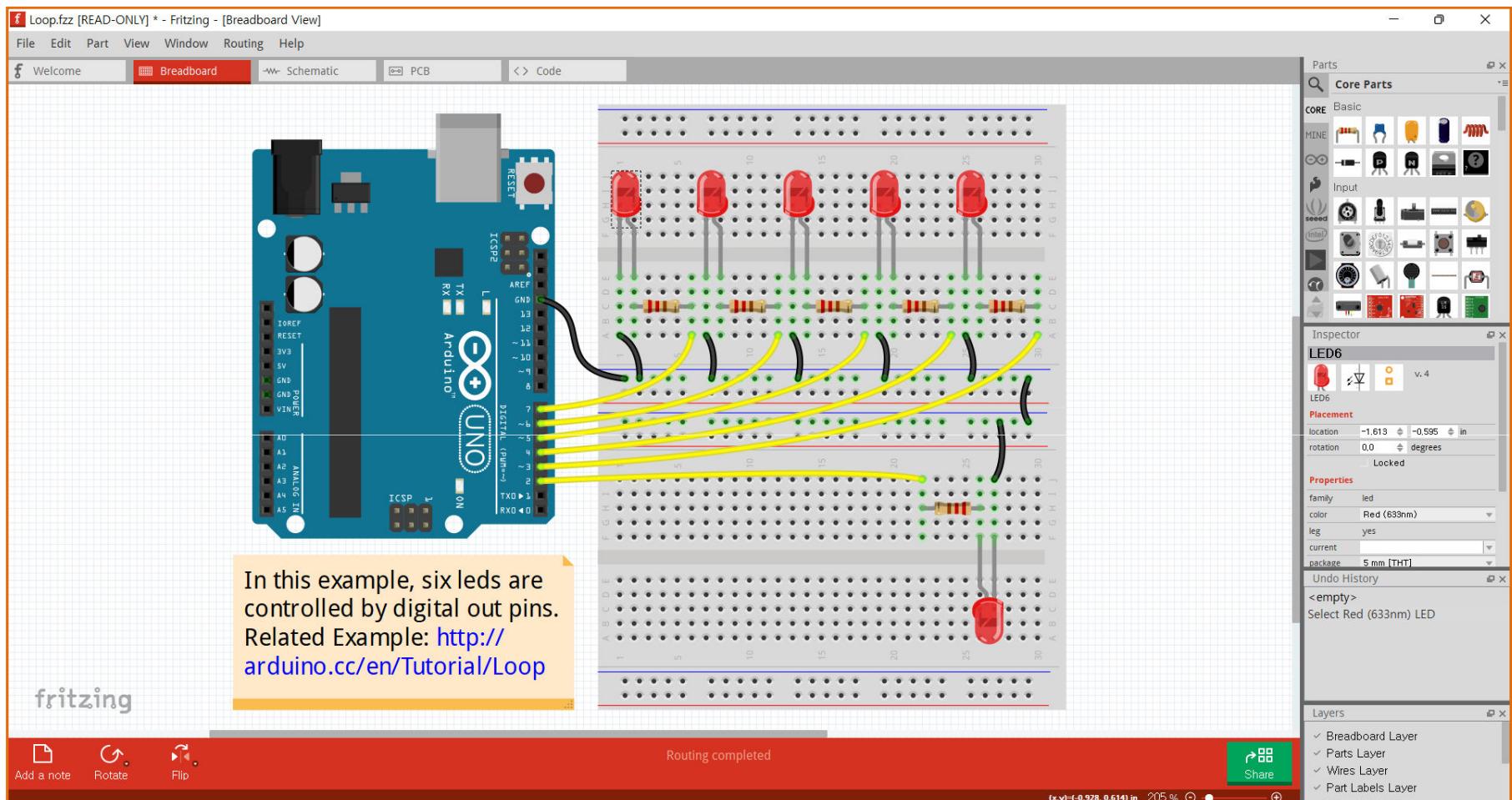
Fritzing – Starting a New Project

- Create a new sketch: File > New
- Open a Fritzing sketch: File > Open
- Open an example: File > Open Example > ... > ...



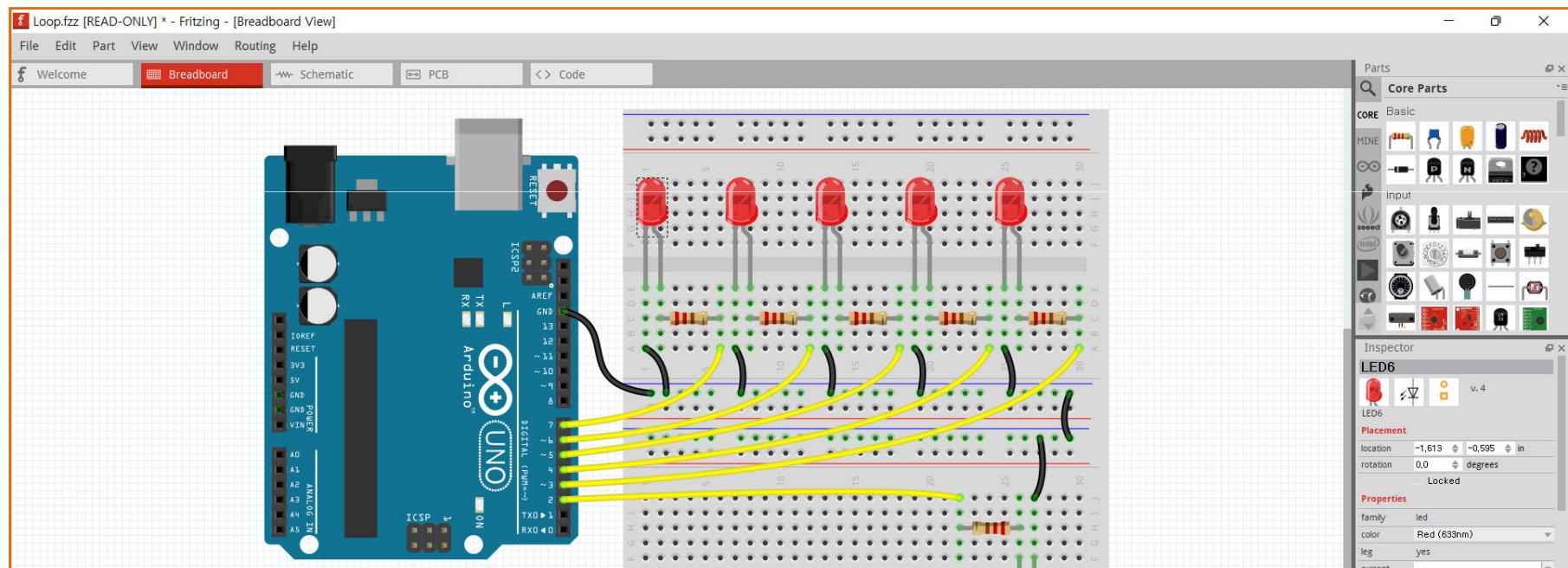
Fritzing – Arranging the Environment

- Show the Parts window: Window > Parts
- Show the Inspector window: Window > Inspector
- Show the Layers window: Window > Layers



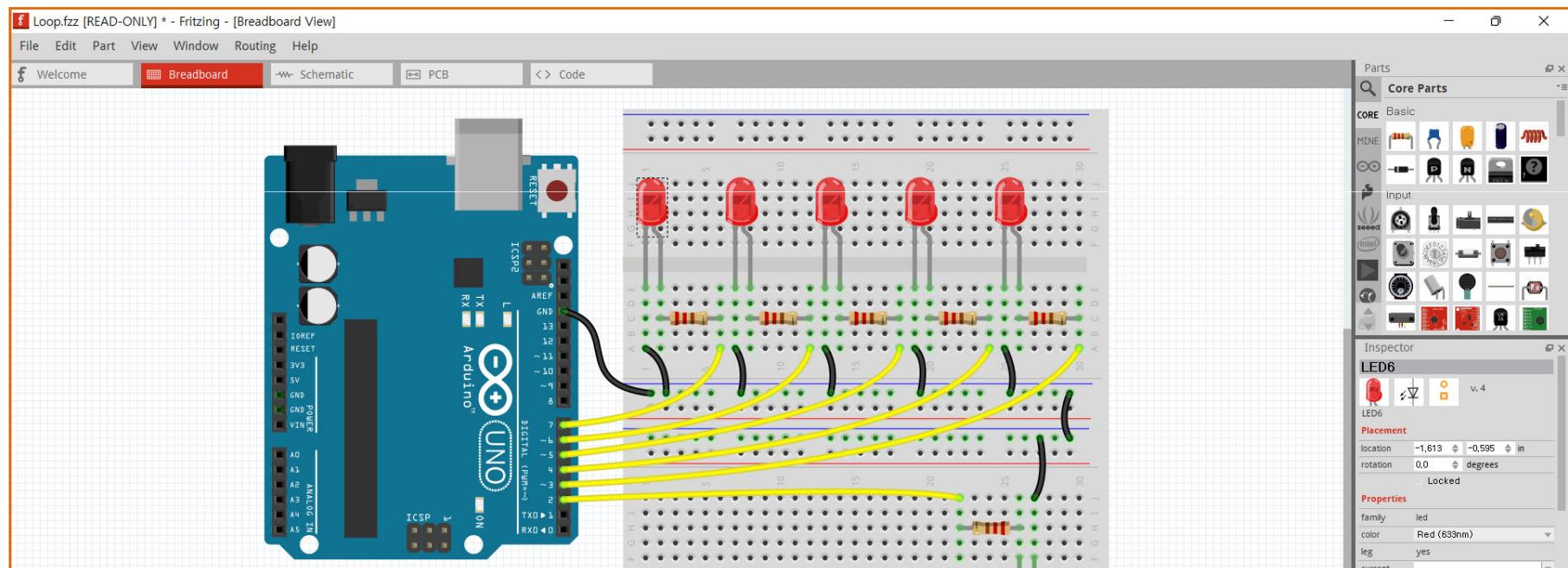
Fritzing – Building a Circuit

- Drag and drop an Arduino from the Parts window. Do the same with the breadboard and all other parts of your circuit.
- Arrange parts by selecting, dragging and dropping, or by using the functions in the menu bar.
- To delete a part, simply select and press DELETE.



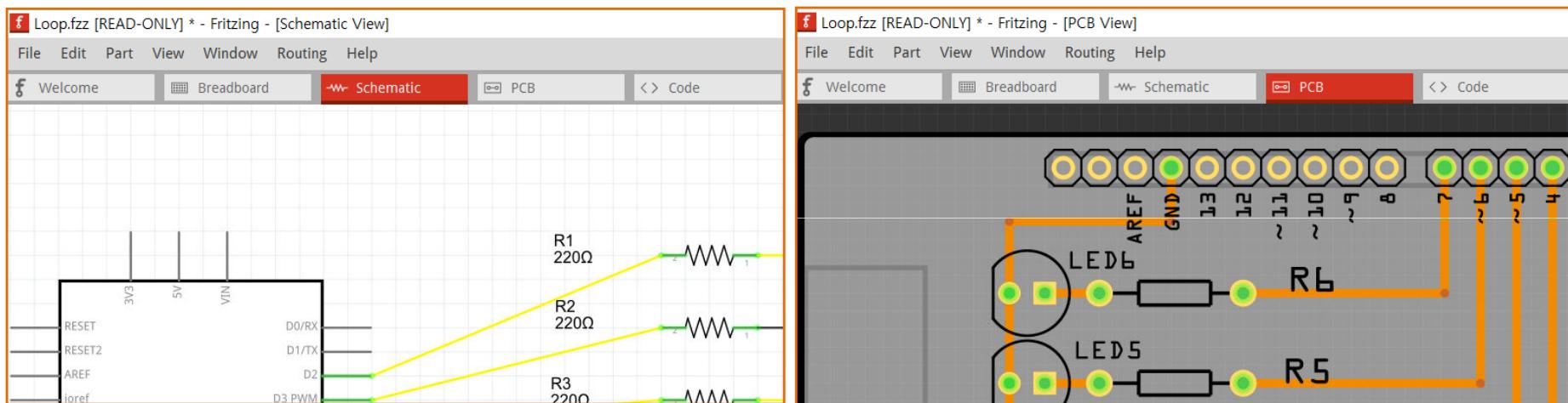
Fritzing – Building a Circuit (cont.)

- Click & drag the Arduino +5V connector. This should create a wire. Drop the wire on one of the breadboard's connectors. The connection is confirmed by a small green circle or square.
- Connect all parts until the circuit looks exactly like your circuit in the real world. Notice that connectors that are not properly connected are painted red.



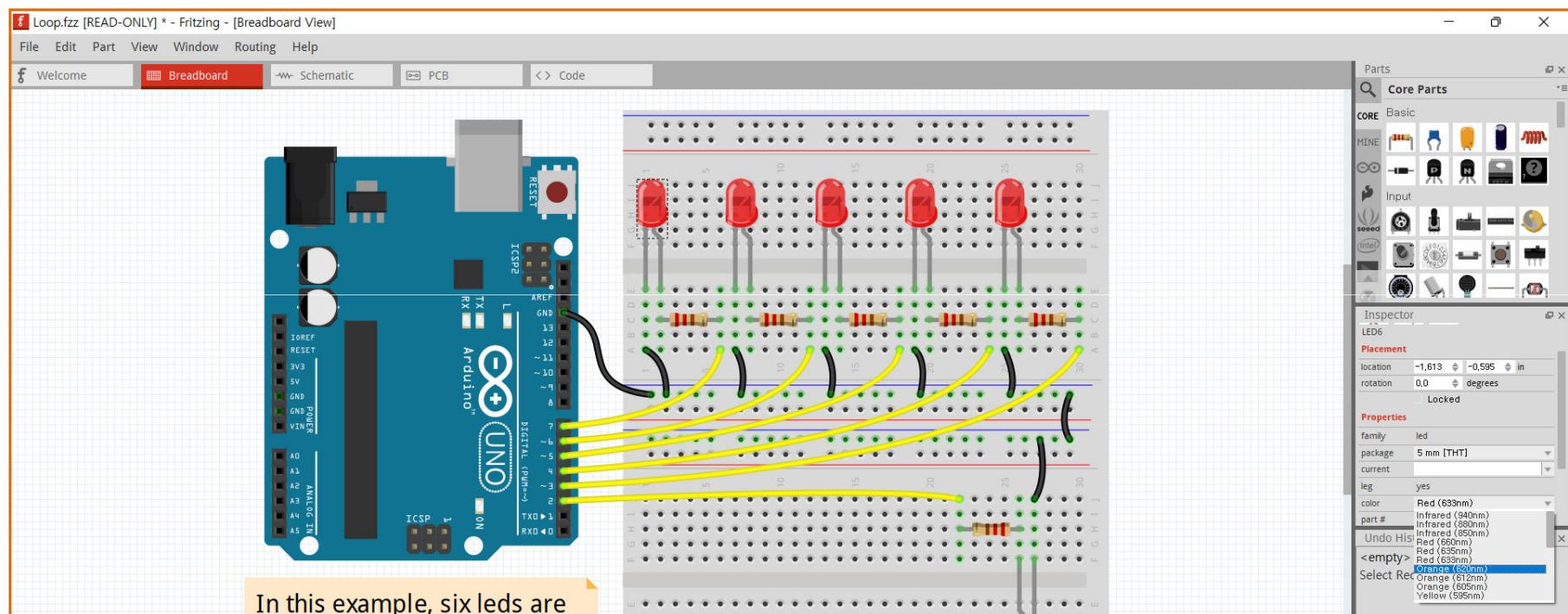
Fritzing – Building a Circuit (cont.)

- If you click and hold on a connector, Fritzing will highlight all equipotential connectors. This can really be useful if you want to see the whole set of connections attached to this particular connection.
- You can bend wires by adding bend-points. Just drag them out of a wire.
- Select the schematic and PCB tabs to watch or edit your circuit in these views.



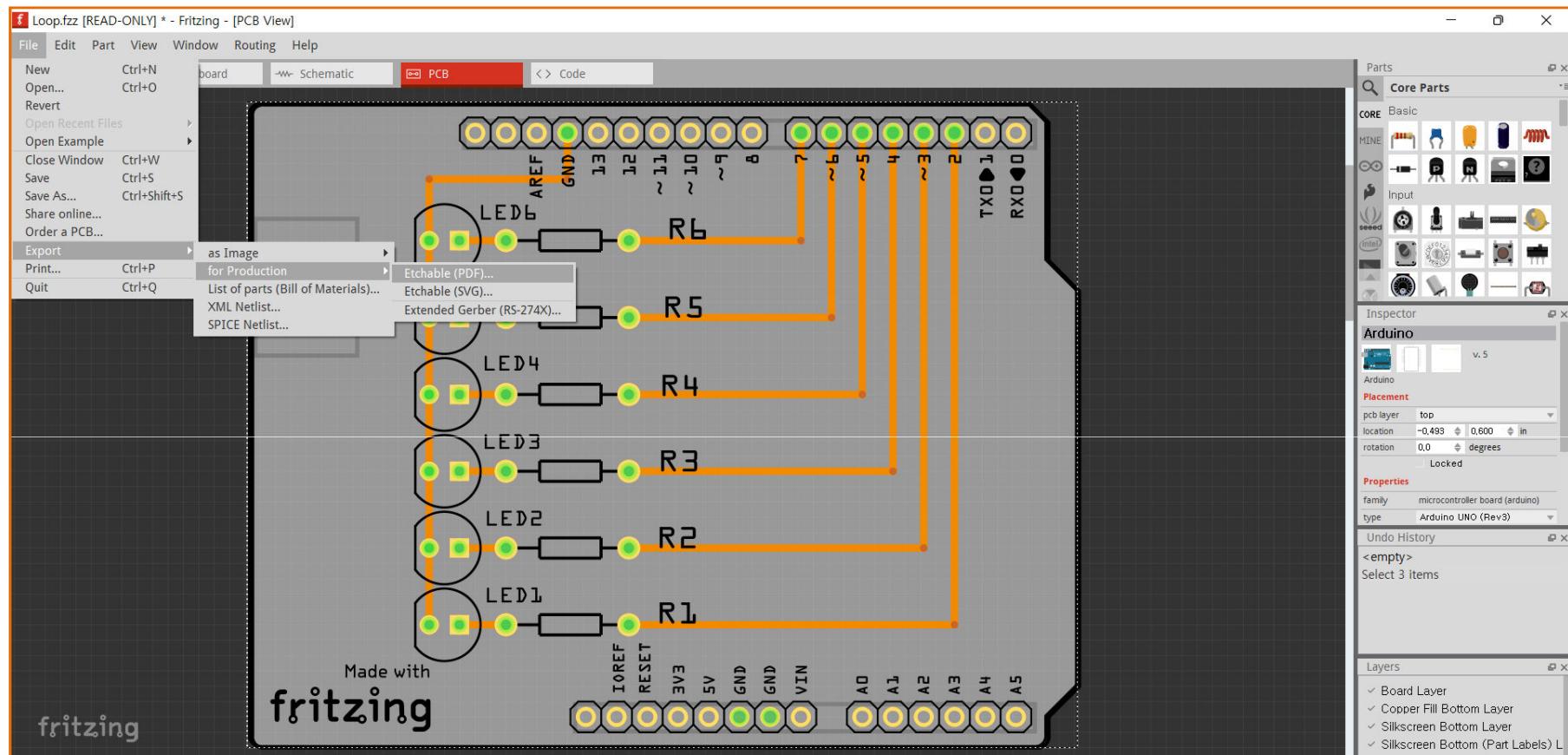
Fritzing – Editing Properties

- Select any of your circuit's parts and have a look at the Part Inspector palette window.
- Click on the part's name and rename it. This is useful when you want to distinguish between similar parts.
- You can also change properties of parts in the PCB View.



Fritzing – Exporting a Circuit

- Select the desired Project View to be exported (breadboard, schematic or pcb).
- From the Fritzing menu bar, select File > Export > and the desired format.

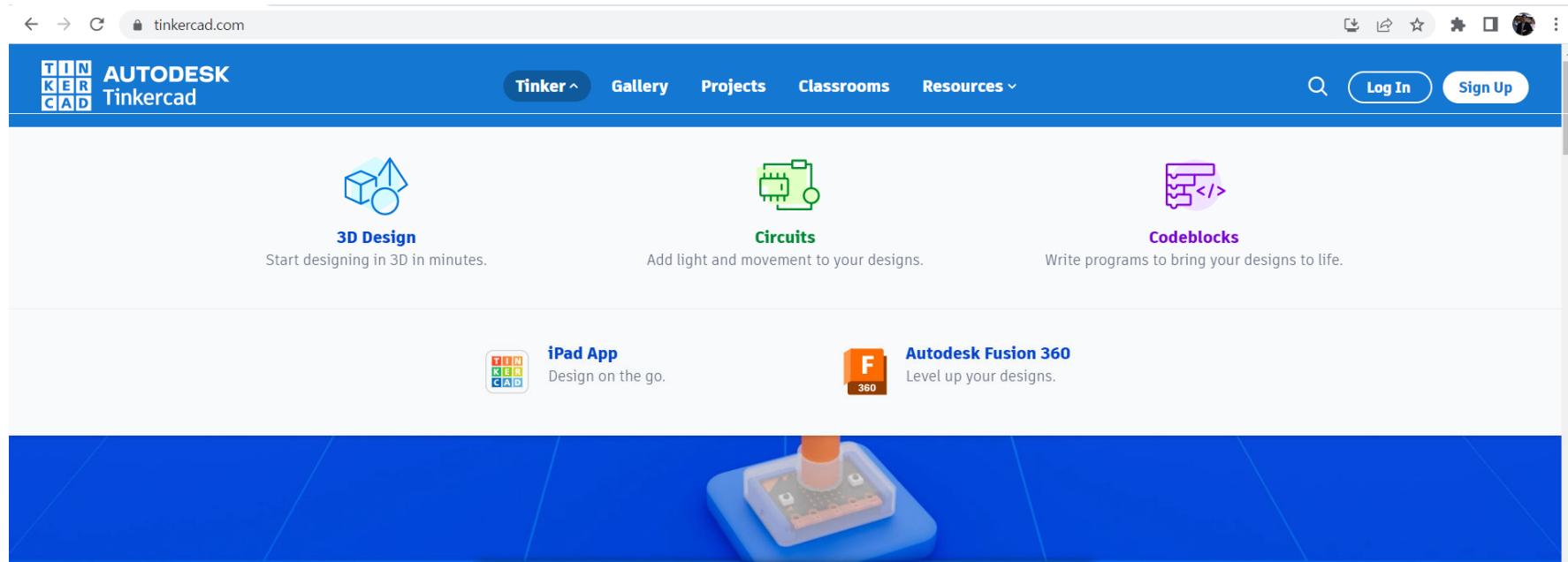


ARDUINO BOARD SIMULATORS

2. Tinkercad

Tinkercad - Introduction

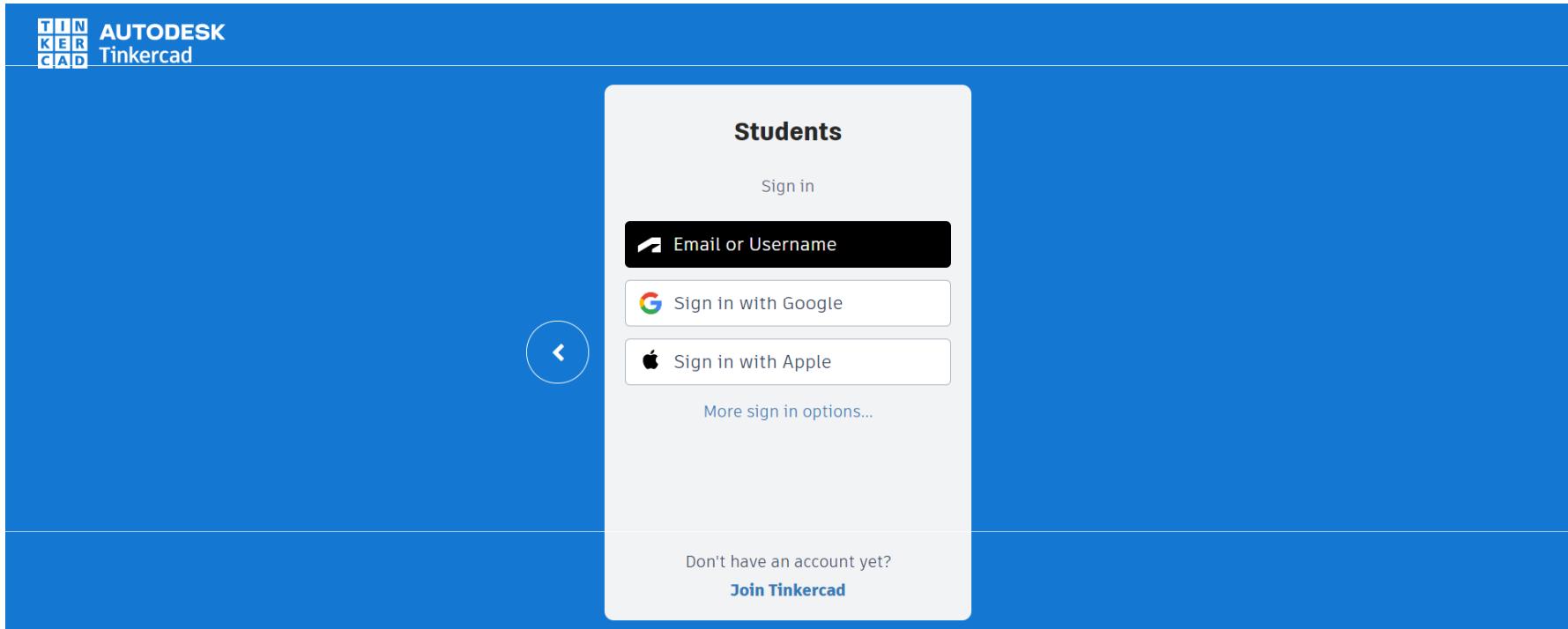
- Tinkercad (<https://www.tinkercad.com>) is a free and easy-to-use application for 3D design, electronics, and coding.



- No need to buy any physical components, sensors, Arduino boards or modules.
- No destruction of any physical components if circuit is incorrect.
- Make your presentation clean and professional.

Tinkercad – Create an Account

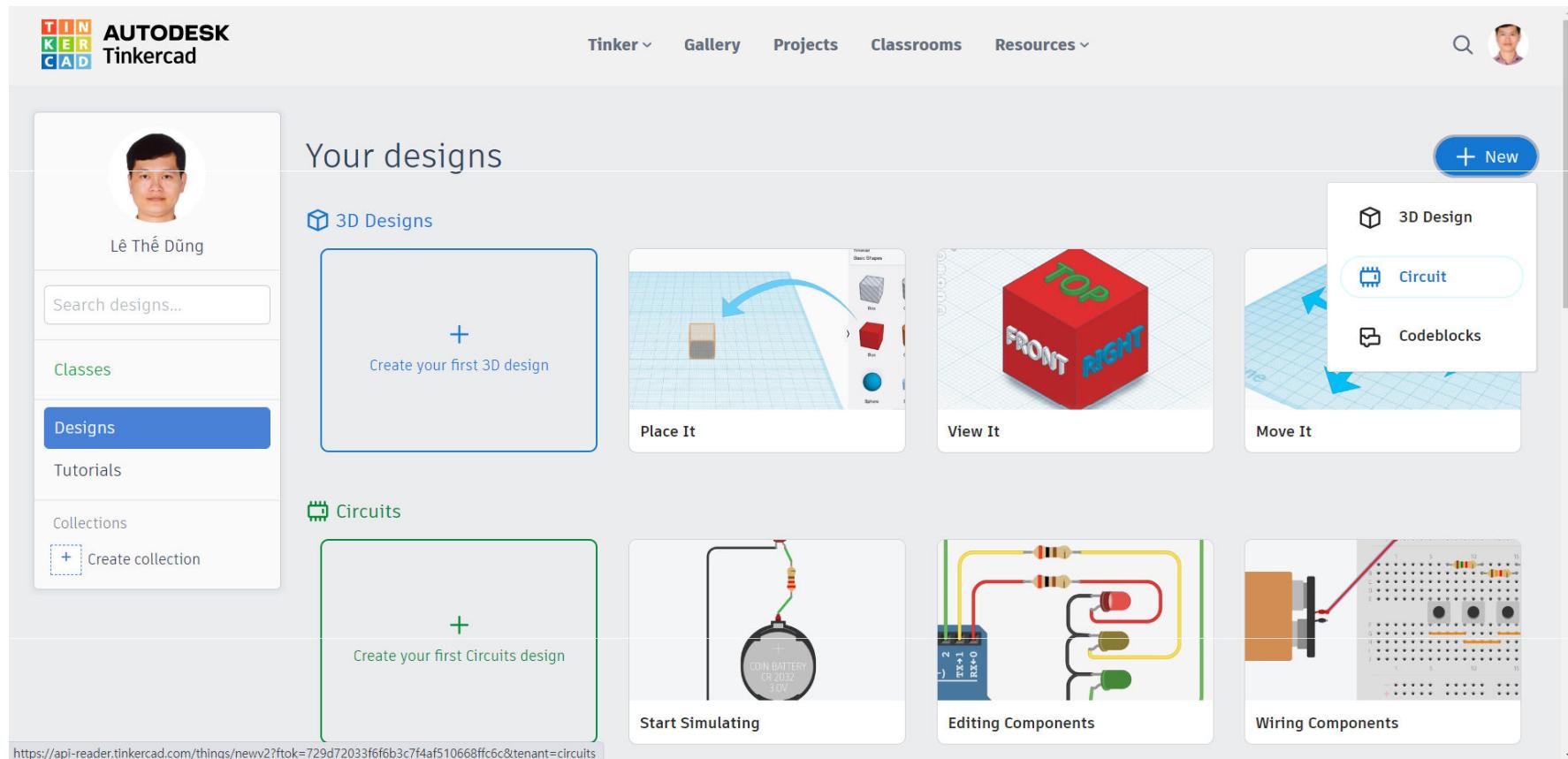
- Access to <https://www.tinkercad.com>.
- Select Login > Student accounts.



- If given a Class Code from teacher, select Student with Class Code.

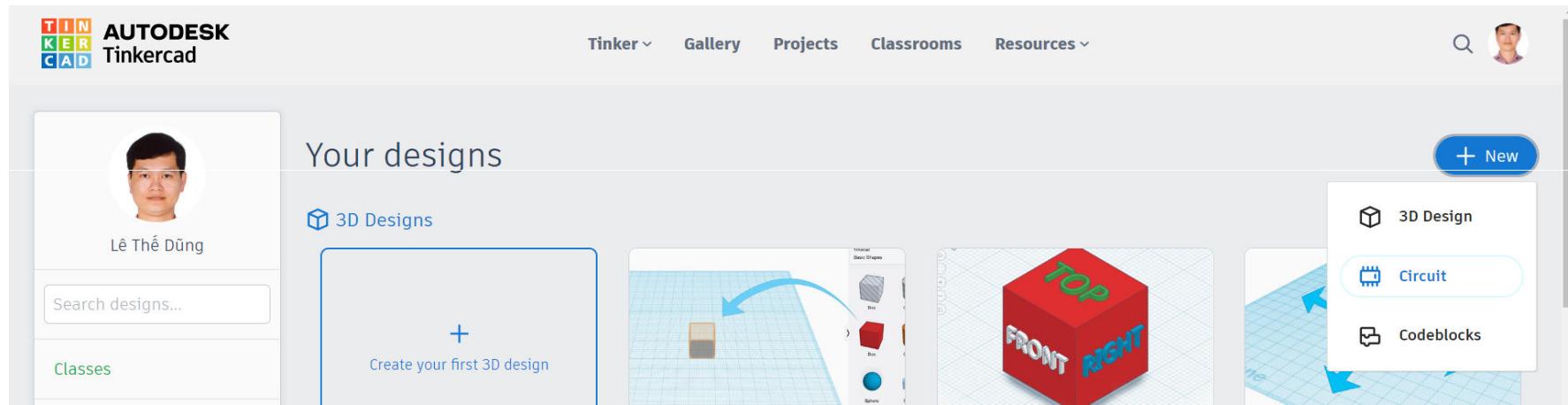
Tinkercad – Create a New Circuit

- Select Design > New > Circuit



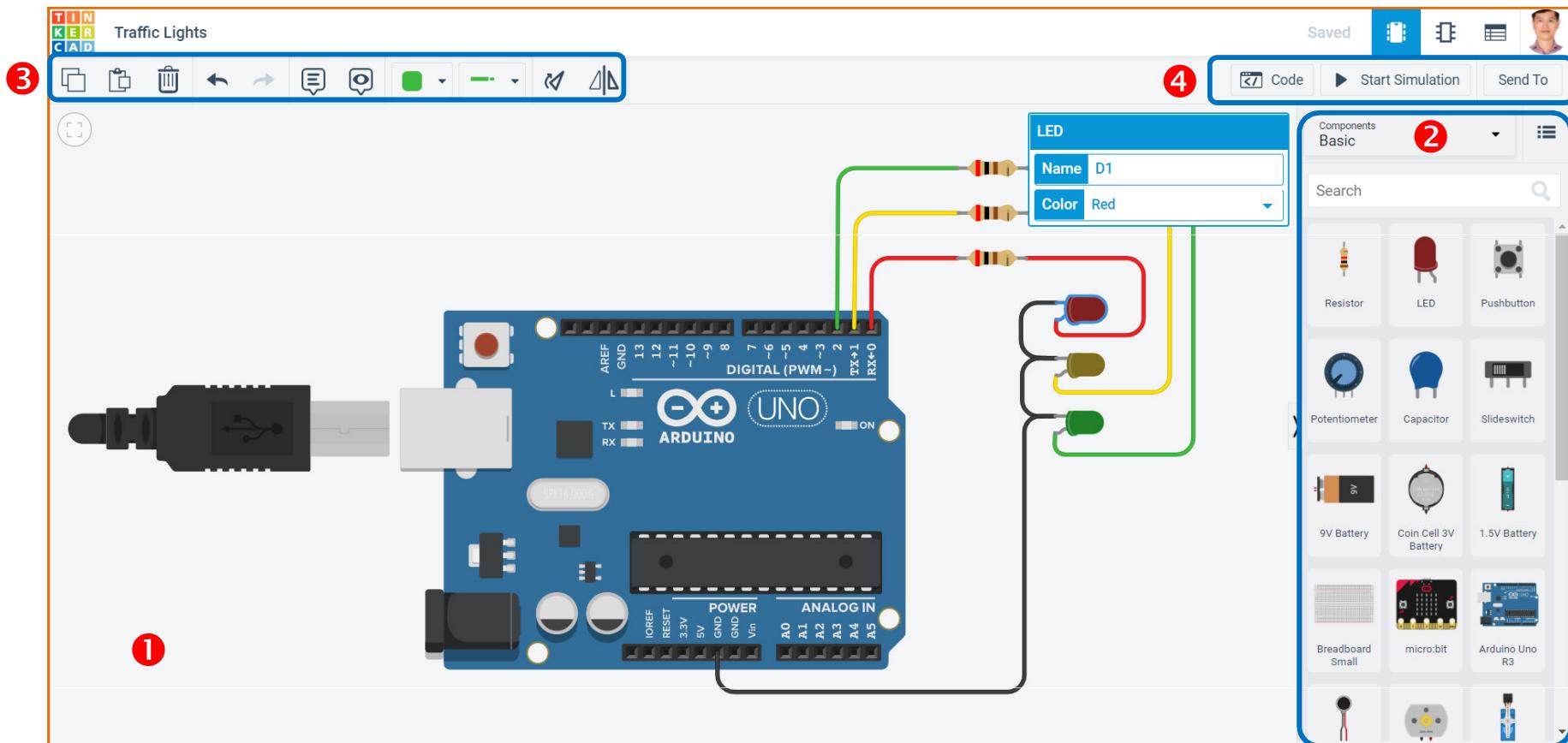
Tinkercad – Create a New Circuit (cont.)

- Select Design > New > Circuit



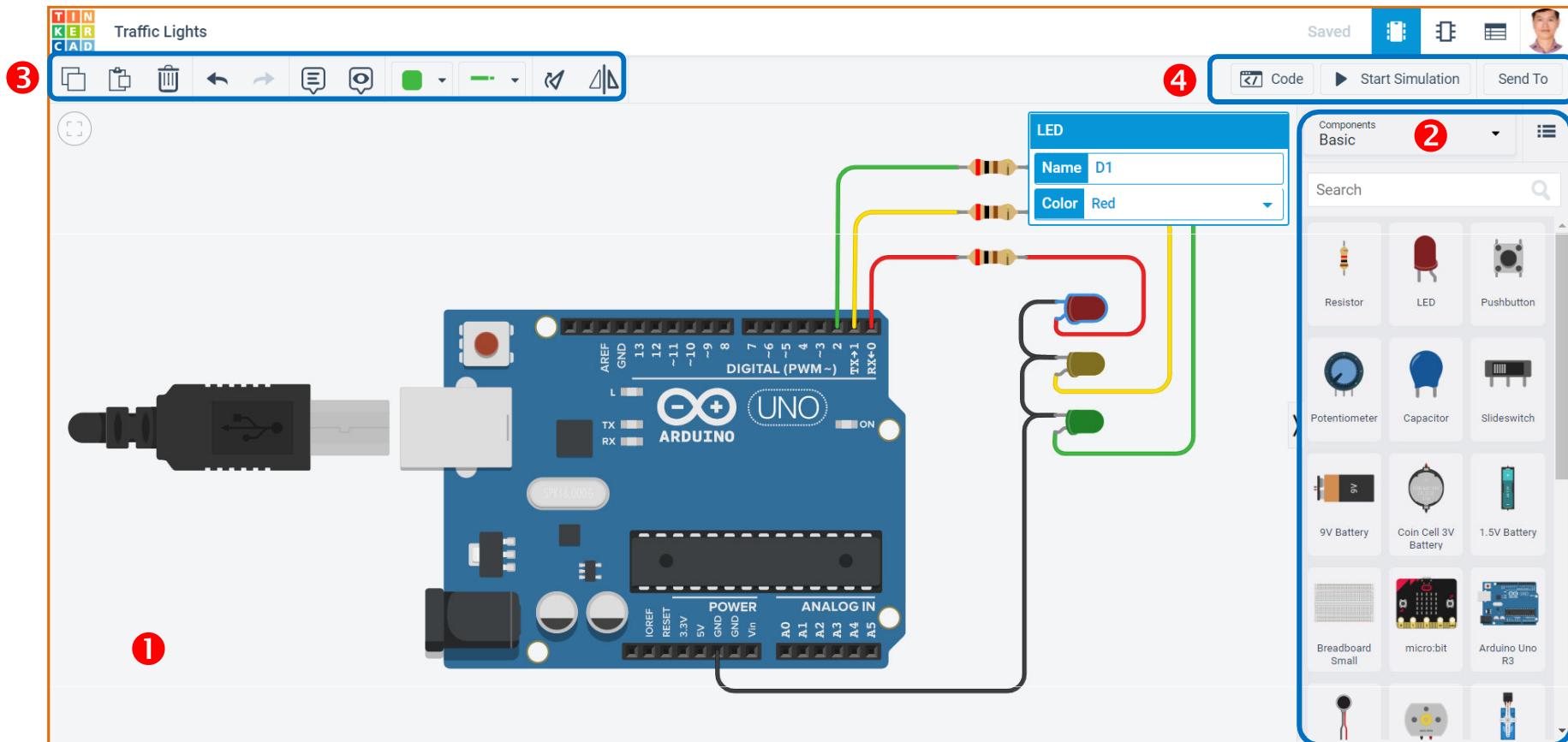
- **3D Design:** Tinkercad is similar to a CAD software where you can design 3D models for 3d printing. The CAD software is based on constructive solid geometry (CSG), which allows users to create complex models by combining simpler objects together.
- **Circuits:** This option helps create a virtual circuit, program it, and test it in real time.
- **Codeblocks:** This new feature helps you create a block programming tree, where the 3D models are formed step by step following the tree instructions.

Tinkercad – Create a New Circuit (cont.)



- ① This space is where we will place all the components. The components can be moved around, edited, and wired together.
- ② This section holds all the components. Scroll down to access component types.

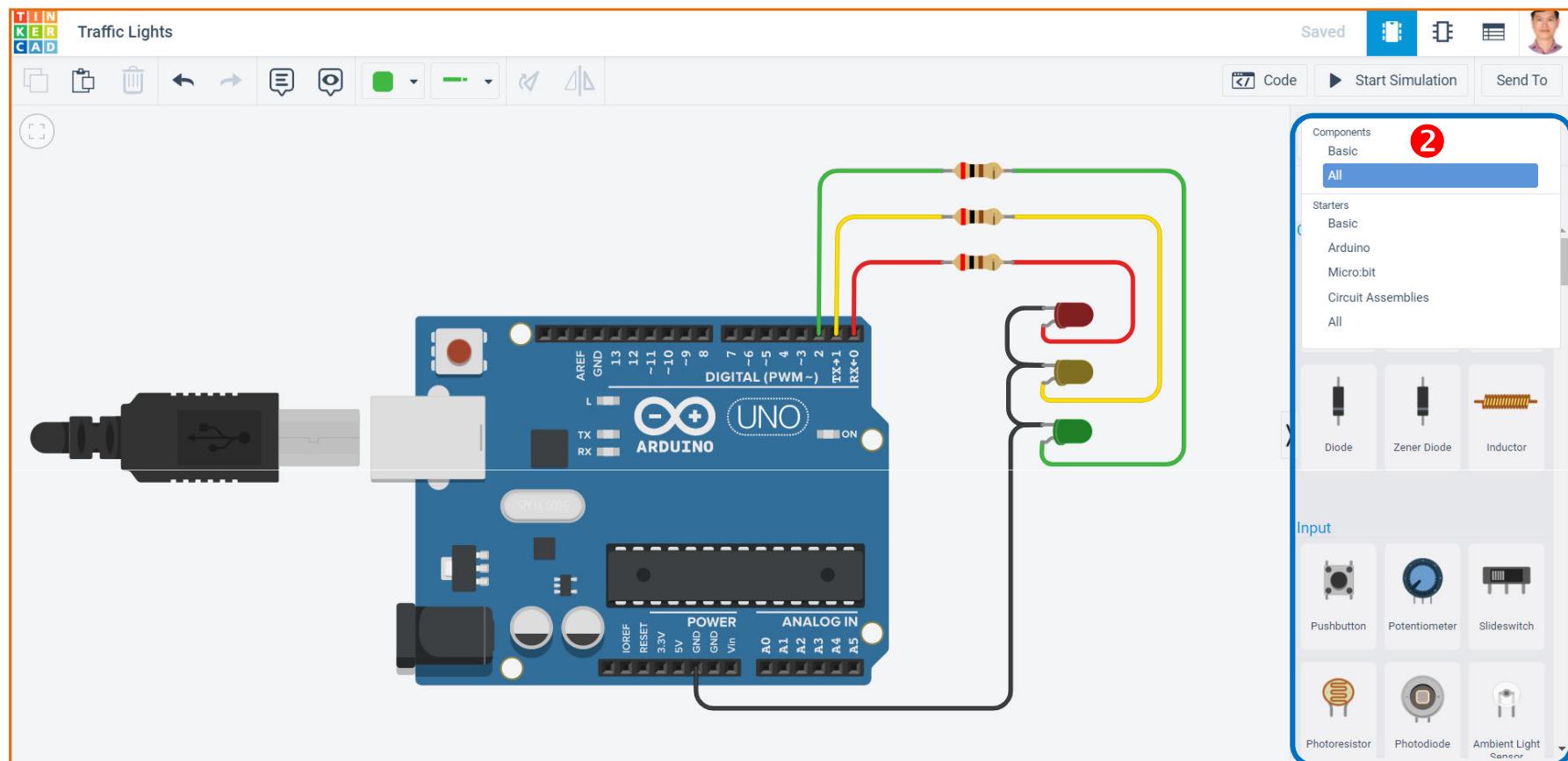
Tinkercad – Create a New Circuit (cont.)



- ③ Use this tab to rotate, delete, undo or redo. It also helps users to create and name labels for components.
- ④ This option helps you program Arduino, use serial monitor, start real time simulation, export code, and share your projects.

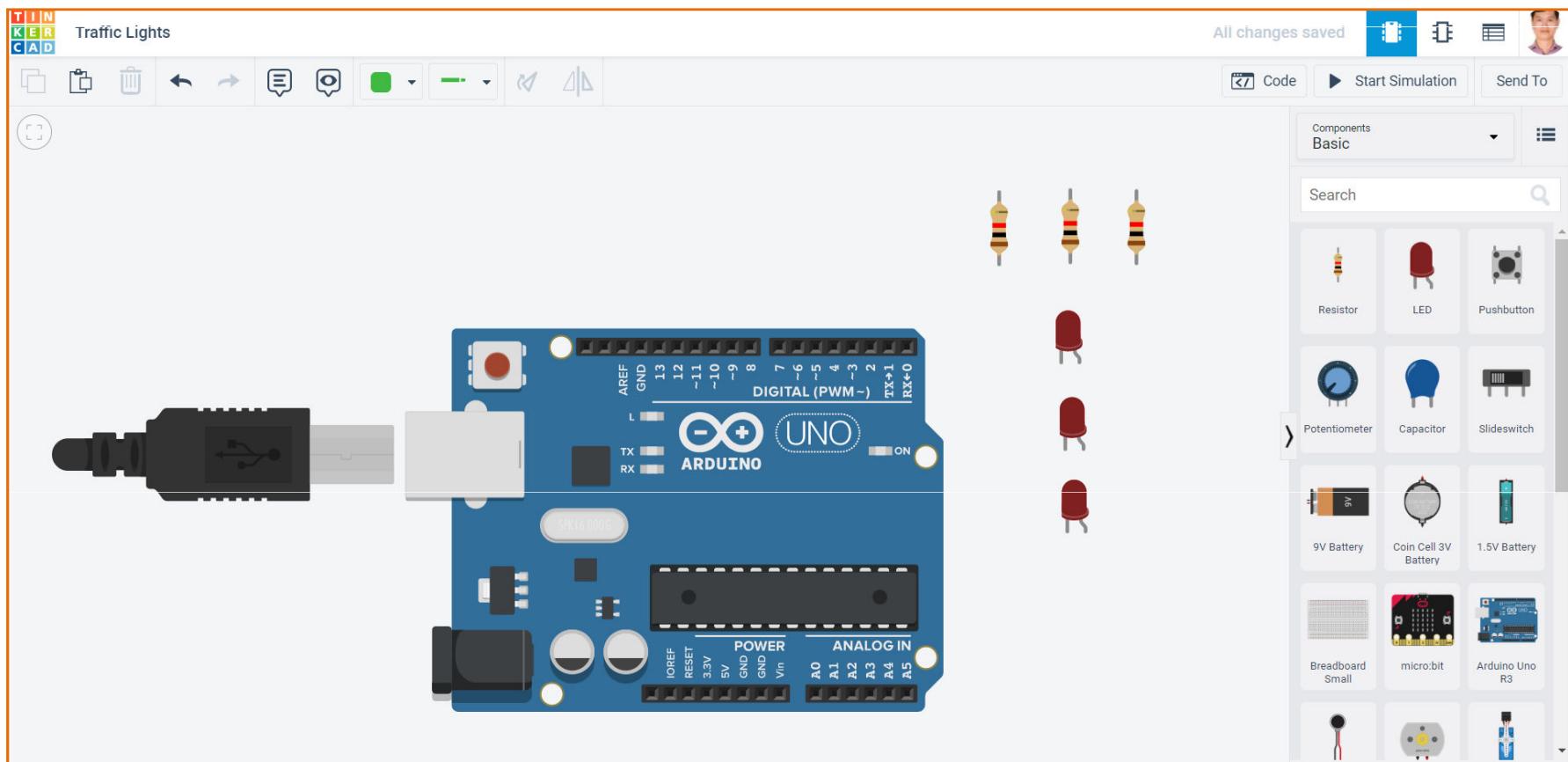
Tinkercad – Create a New Circuit (cont.)

- Tinkercad has many components and modules which can be used with Arduino UNO.
- To get access to more components select All option in ②.



Tinkercad – Build a New Circuit

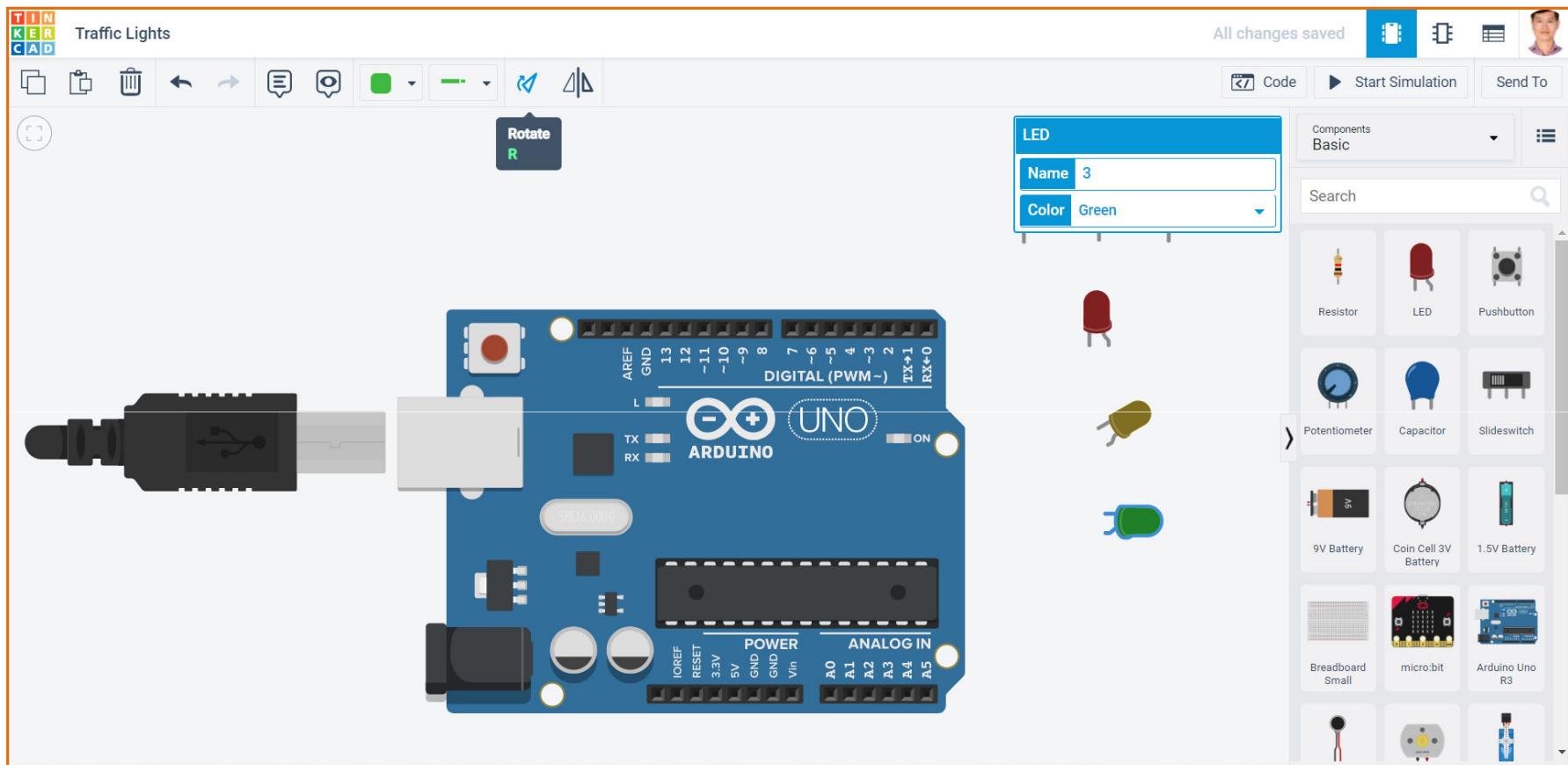
- Place all the components as image below by selecting them from the components section on right side. Click on the component to select and click again anywhere on workspace to place.



- 1 Arduino Uno R3; 3 resistors; 3 LEDs.

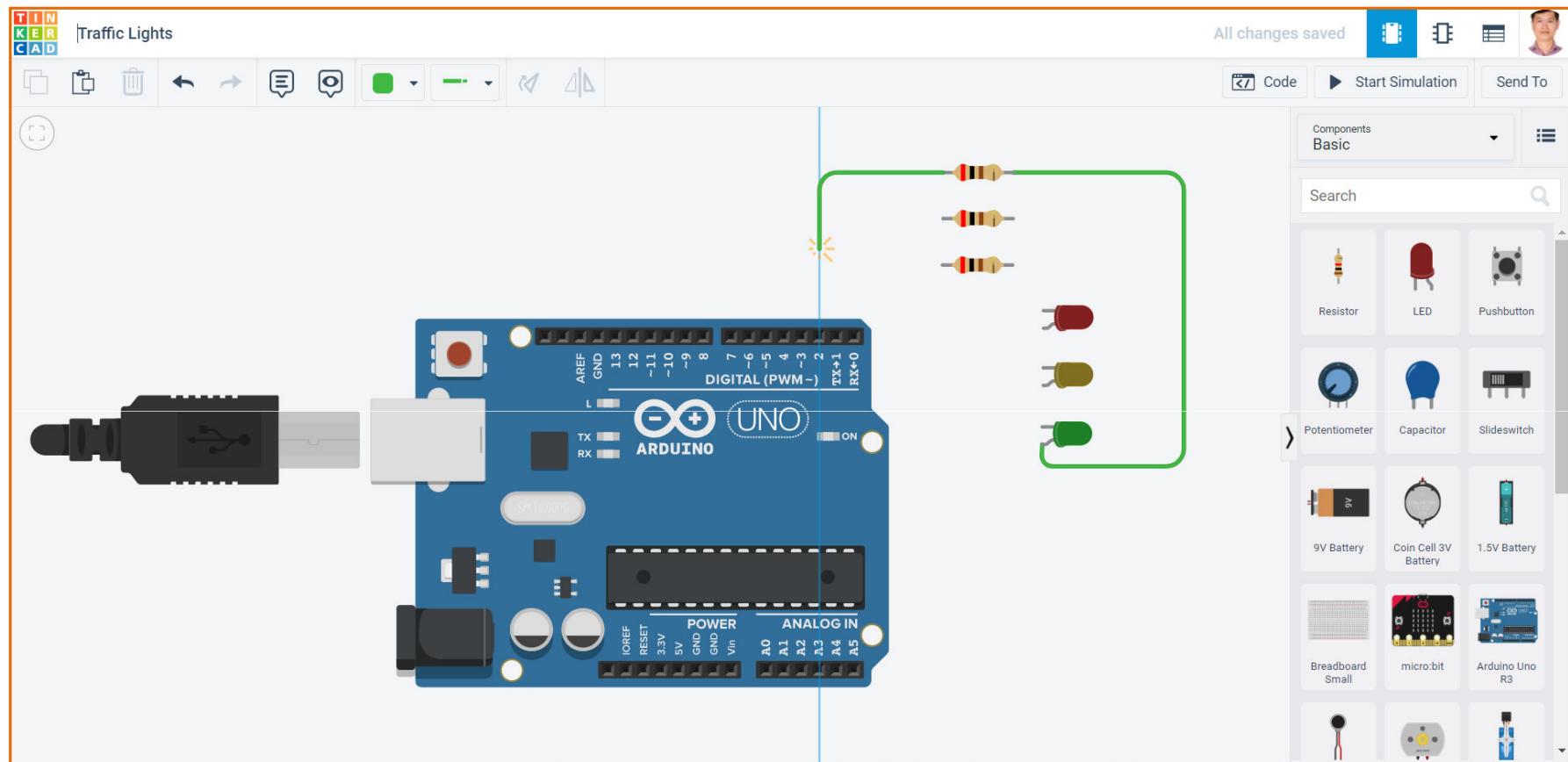
Tinkercad – Build a New Circuit (cont.)

- For rotating the component's direction, click the Rotate icon.
- For changing the component's value, select the component and change its value.



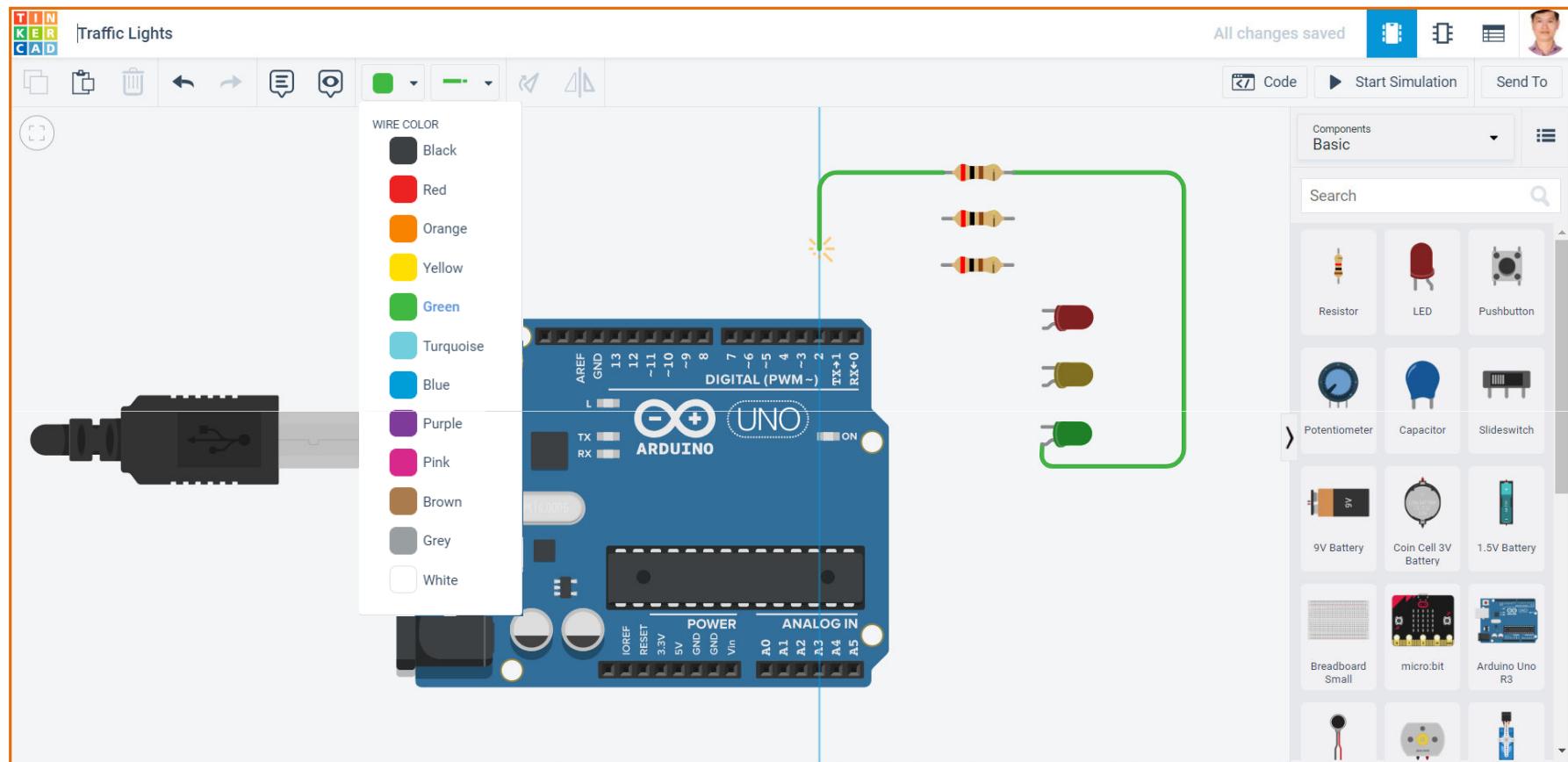
Tinkercad – Build a New Circuit (cont.)

- Move the components to the right positions by dragging, dropping, and rotating.
- Hover over the points of the components to know the terminal name and click it to start wiring the components.



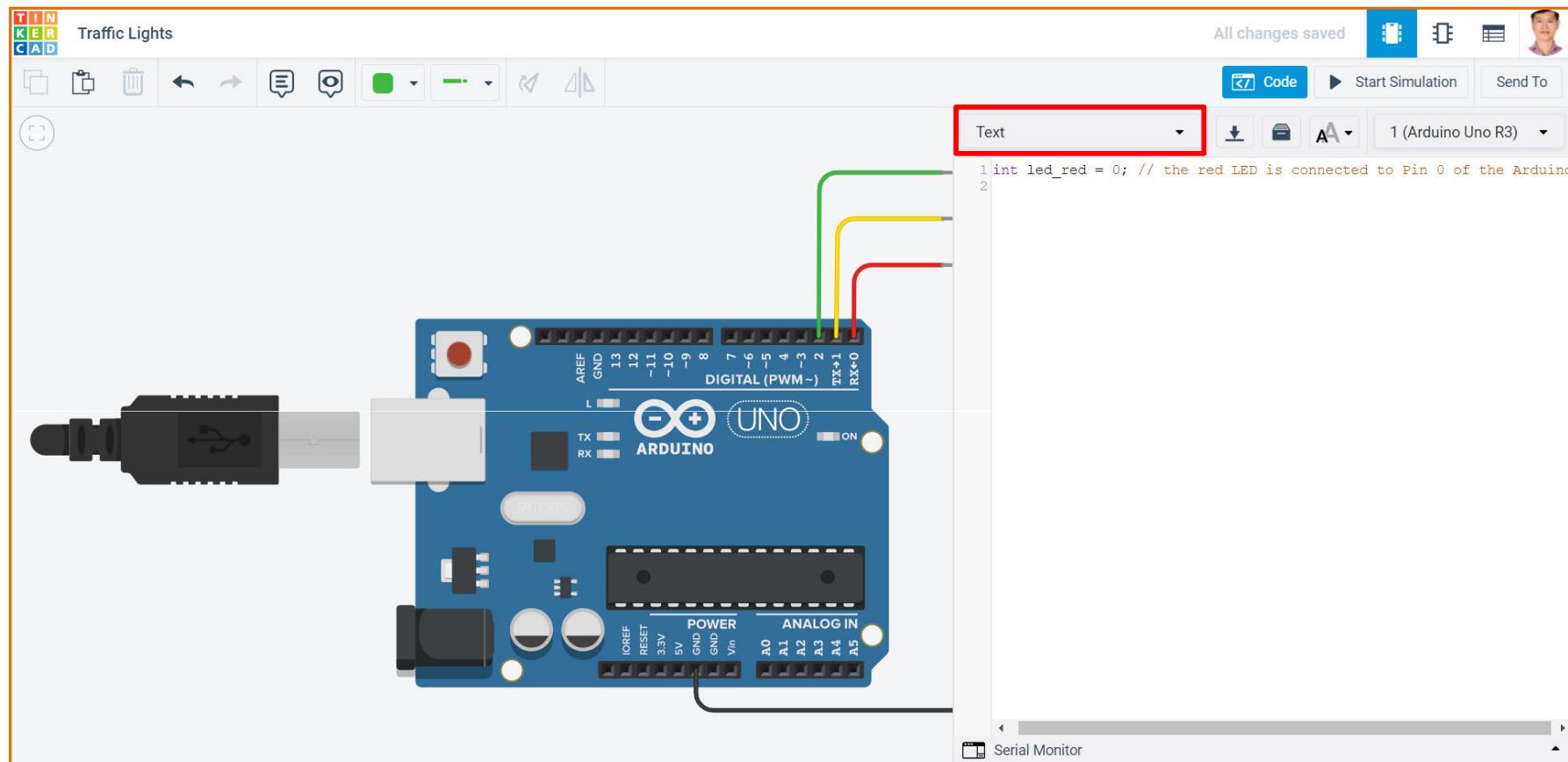
Tinkercad – Build a New Circuit (cont.)

- Click on wire to select the wire color. This helps in differentiating between wire use.
- Wire all the components as shown in the image below.



Tinkercad – Program with Arduino

- Once wiring is complete, select the code option to start coding.
- Use the Text option to write the code for this tutorial. This is where all code can be written. Copy and paste the code shown on next page.



Tinkercad – Program with Arduino (cont.)

```
int led_red = 0; // the red LED is connected to Pin 0 of the Arduino
int led_yellow = 1; // the yellow LED is connected to Pin 1 of the Arduino
int led_green = 2; // the green LED is connected to Pin 2 of the Arduino

void setup() {
    // set up all the LEDs as OUTPUT
    pinMode(led_red, OUTPUT);
    pinMode(led_yellow, OUTPUT);
    pinMode(led_green, OUTPUT);
}

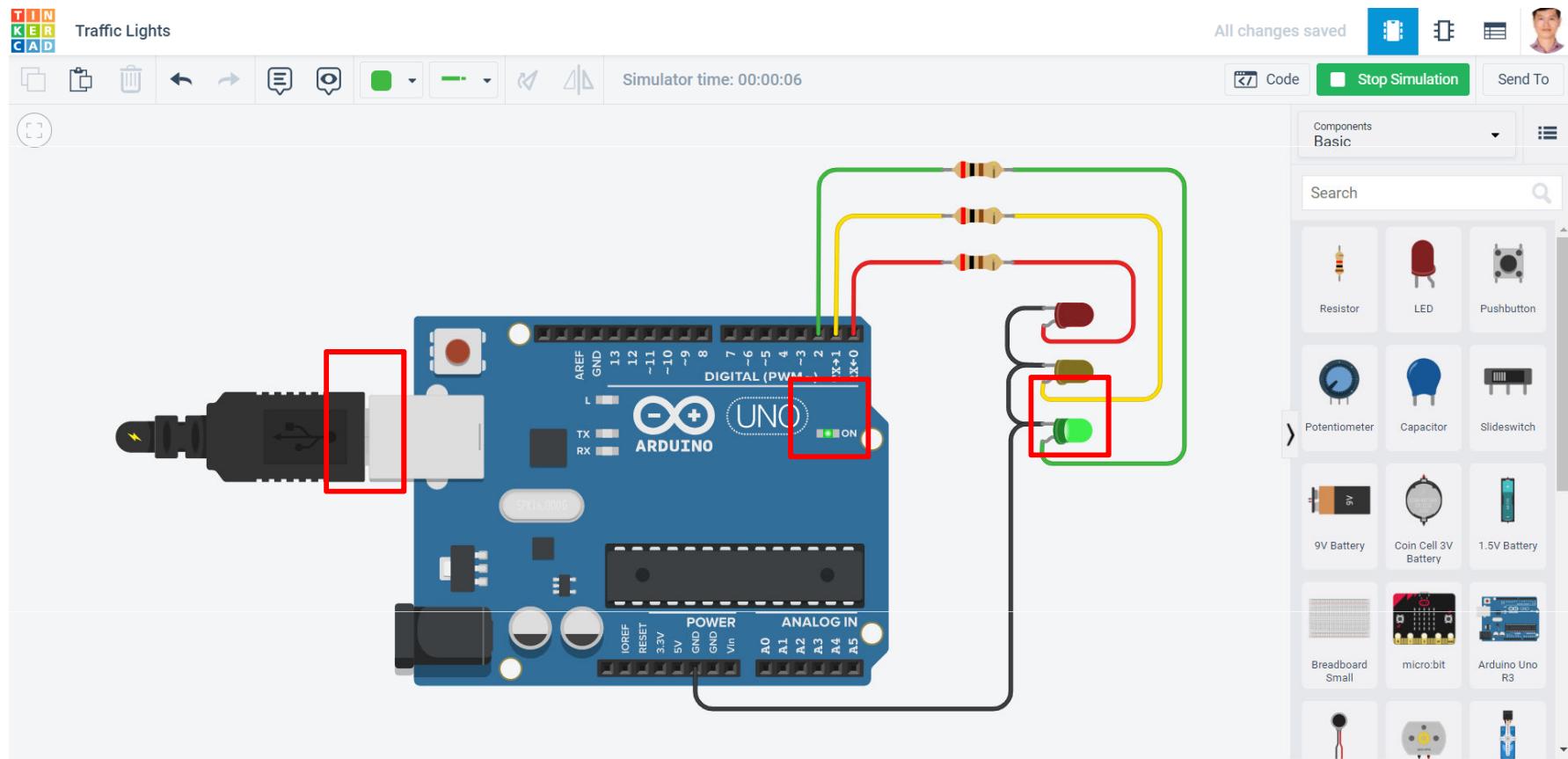
void loop() {
    // turn the green LED on and the other LEDs off
    digitalWrite(led_red, LOW);
    digitalWrite(led_yellow, LOW);
    digitalWrite(led_green, HIGH);
    delay(2000); // wait 2 seconds

    // turn the yellow LED on and the other LEDs off
    digitalWrite(led_red, LOW);
    digitalWrite(led_yellow, HIGH);
    digitalWrite(led_green, LOW);
    delay(1000); // wait 1 second

    // turn the red LED on and the other LEDs off
    digitalWrite(led_red, HIGH);
    digitalWrite(led_yellow, LOW);
    digitalWrite(led_green, LOW);
    delay(3000); // wait 3 seconds
}
```

Tinkercad – Simulate the Built Circuit

- Click Start Simulation after writing the code.



THANK YOU ALL FOR LISTENING



QUESTIONS AND ANSWERS