**Introduction**

In this experiment, we will use a button and a DC motor together to assemble a small fan. With the button, we can shift the rotational speed or gear selection of the fan.

**Components**

- 1 \* SunFounder Uno board

- 1 \* Breadboard

- 1 \* USB cable

- 1 \* DC motor

- 1 \* Resistor (10KΩ)

- 1 \* Button

- 1 \* Motor drive L293D

- 1 \* 104 Ceramic Capacitor

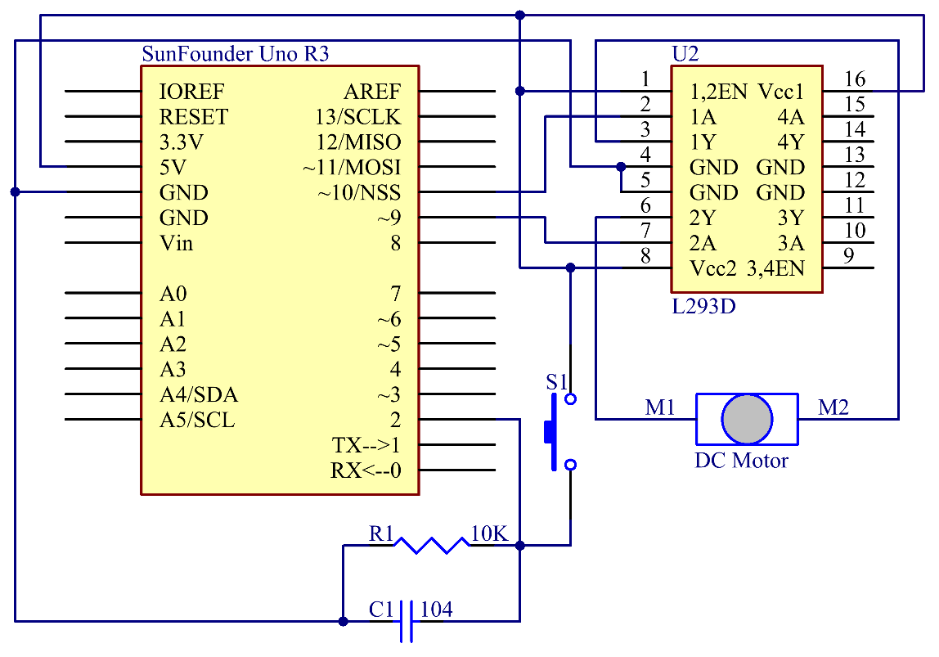
- 1 \* Fan

- Jumper wires

**Principle**

This experiment is to make a small fan based on Lesson 7 by adding a button. The capacitor and resistor here connected to the button in parallel are to eliminate jitters and output a steady low level, so as to protect the motor.

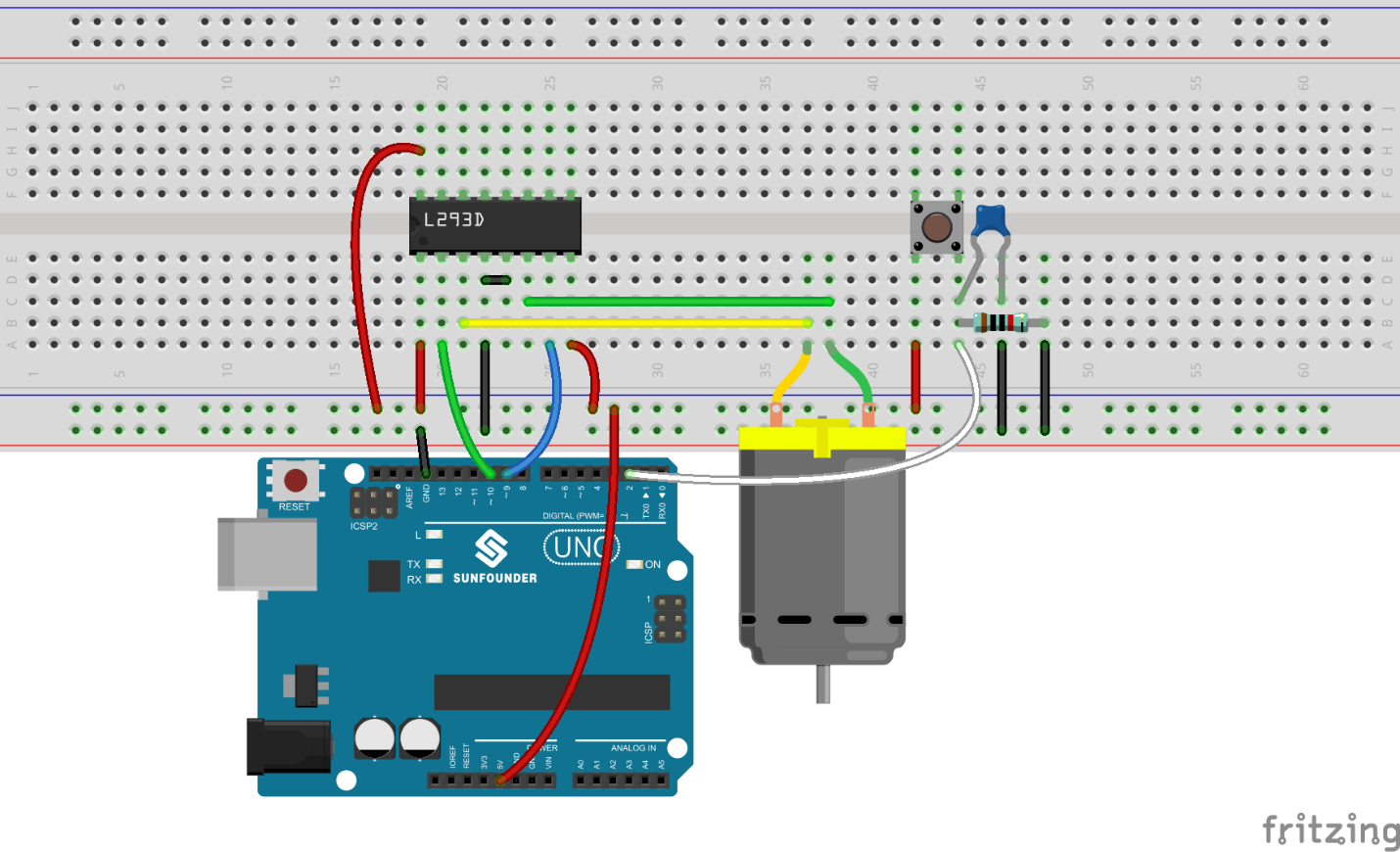
The schematic diagram



Connect one button to pin 2. Count by programming how many times the button is pressed.  If it is more than 4, the number will be cleared. Press the button for once, and the rotational speed of the motor is 150. Press it for the second time, and the speed will be 200. Press the button again, and the speed is 250. When it comes to the fourth pressing, the motor will stop and the number will be cleared. When the button is pressed again, it will start from 1 again. So that makes a small fan with levels.

**Experimental Procedures**

**Step 1:** Build the circuit

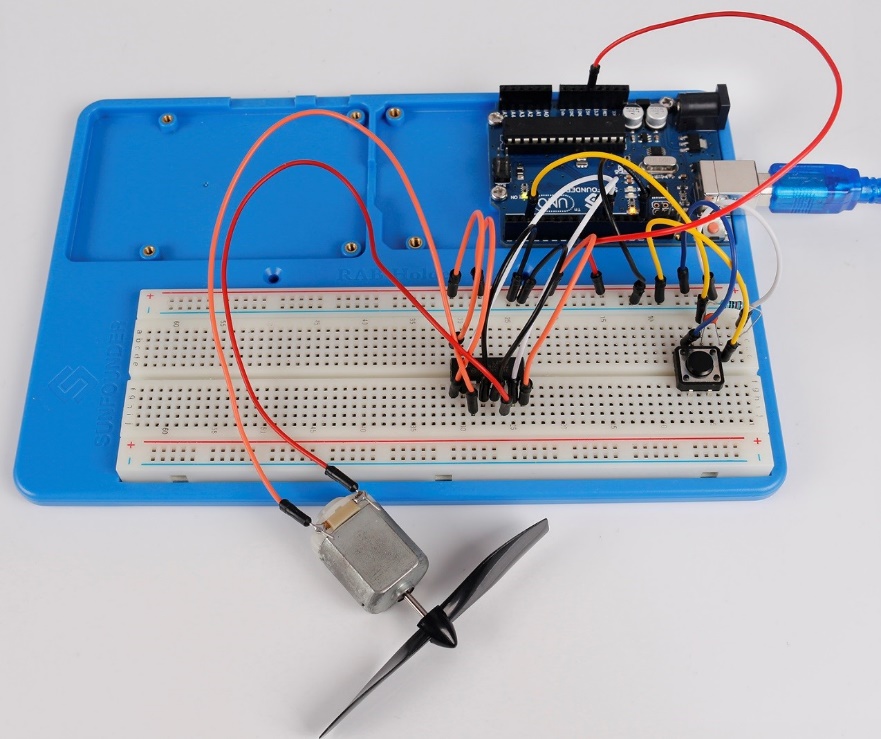


**Step 2**: Open the code file

**Step 3:** Select correct Board and Port

**Step 4:** Upload the sketch to the SunFounder Uno board

As mentioned above, the times you press the button will change the rotational speed of the fan. Pressing it once will cause it to rotate slowly, while at the third time it will rotate rather quickly, and pressing it the fourth time will cause it to stop.



**Code**

|  |
| --- |
| //Small Fan  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* As explained above, the amount of times you press the button should change the rotation speed of the fan.  \* Pressing it once will cause it to rotate slowly,  \* while pressing it three times will cause it to rotate quickly, and pressing it four times will cause it to stop.  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  //Email:support@sunfounder.com  //Website:www.sunfounder.com  //2015.5.7  // constants won't change. They're used here to  // set pin numbers:  const int buttonPin = 2; // the number of the pushbutton pin  const int ledPin = 13; // the number of the LED pin  const int motorIn1 = 9;  const int motorIn2 = 10;  int stat = 0;  #define rank1 150  #define rank2 200  #define rank3 250  // Variables will change:  int buttonState; // the current reading from the input pin  int lastButtonState = LOW; // the previous reading from the input pin  // the following variables are long's because the time, measured in miliseconds,  // will quickly become a bigger number than can be stored in an int.  long lastDebounceTime = 0; // the last time the output pin was toggled  long debounceDelay = 50; // the debounce time; increase if the output flickers  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void setup()  {  //set theled,motors as OUTPUT,button as INPUT  pinMode(buttonPin, INPUT);  pinMode(ledPin, OUTPUT);  pinMode(motorIn1,OUTPUT);  pinMode(motorIn2,OUTPUT);  Serial.begin(9600);  }  void loop() {  // read the state of the switch into a local variable:  int reading = digitalRead(buttonPin);  /  if (reading != lastButtonState)// If the button state is different from last time  {  lastDebounceTime = millis();// reset the debouncing timer  }  if ((millis() - lastDebounceTime) > debounceDelay)  {  if (reading != buttonState)  {  buttonState = reading; // Store the state of button in buttonState  // only toggle the LED if the new button state is HIGH  if (buttonState == HIGH)  {  digitalWrite(ledPin, HIGH); //turn on the LED  stat = stat + 1;  if(stat >= 4)// When stat>=4, set it as 0.  {  stat = 0;  }  }  else  digitalWrite(ledPin, LOW);  }  }  switch(stat)  {  case 1:  clockwise(rank1);// When stat=1, set the rotate speed of the motor as rank1=150  break;  case 2:  clockwise(rank2);// When stat=2, set the rotate speed of the motor as rank1=200  break;  case 3:  clockwise(rank3);// When stat=3, set the rotate speed of the motor as rank1=250  break;  default:  clockwise(0);// else, set the rotate speed of the motor as rank1=150  }  // save the reading. Next time through the loop,  // it'll be the lastButtonState:  lastButtonState = reading;  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void clockwise(int Speed)//  {  analogWrite(motorIn1,0);  analogWrite(motorIn2,Speed);  }  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/ |