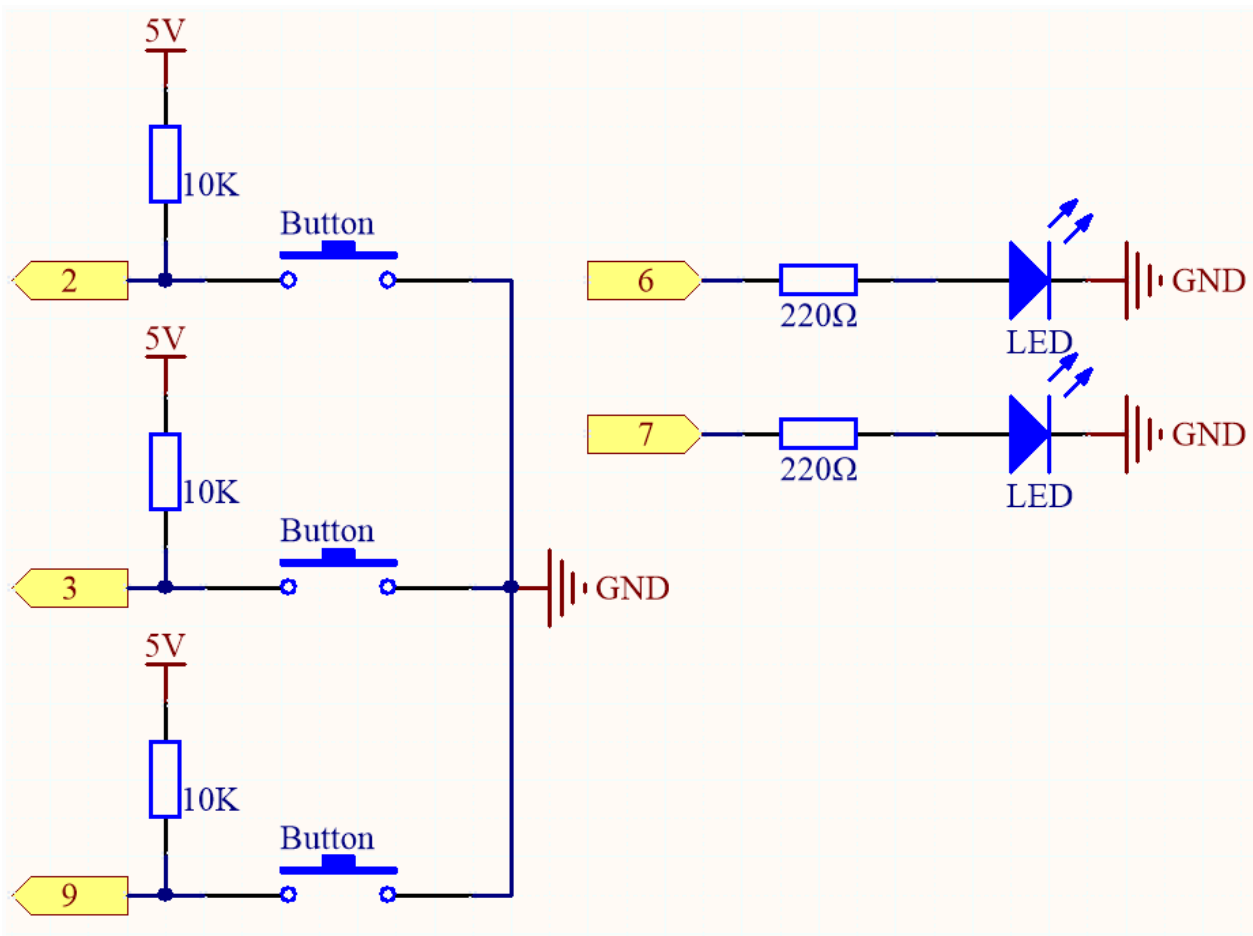


Reaction Game

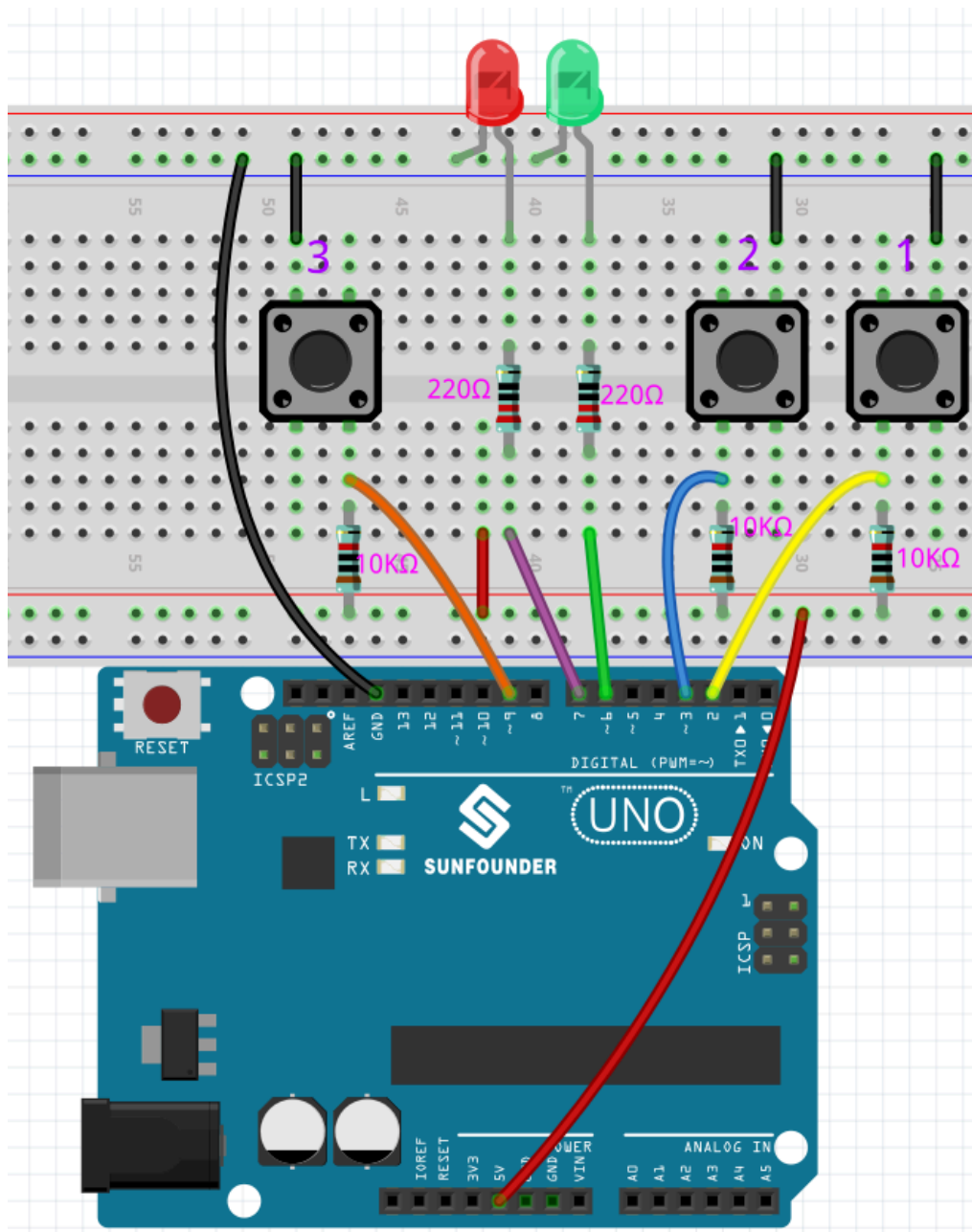
Components Needed:

- ☐ 1 SunFounder R3 Board
- ☐ 1 Breadboard
- ☐ Jumper Wires
- ☐ 2 220 Resistors
- ☐ 3 10K Resistors
- ☐ 1 Red LED
- ☐ 1 Green LED
- ☐ 3 Button

Schematic:



Wiring:



Description Of Hardware:

1. **Arduino Uno:** This is the main controller board for the game, executing the uploaded program to manage the LEDs and buttons.
2. **Buttons (1, 2, and 3):** These are the input devices used by the player. Typically, one of these will be the "Ready" button to start the game, and the other two will correspond to the red and green LEDs for the reaction test.
3. **LEDs (red and green):** These are the output devices that provide a visual signal to the player. When one of these lights up, the player has to press the corresponding button as quickly as possible.
4. **Resistors (220 Ω and 10k Ω):** The 220 Ω resistors are current-limiting resistors for the LEDs to prevent them from burning out by limiting the amount of current passing through them. The 10k Ω resistors are likely used for pull-down or pull-up purposes on the buttons to ensure a stable LOW or HIGH signal when the button is not pressed.
5. **Wires:** These are used to connect the components to the Arduino Uno. Each wire serves a specific purpose:
 - The black wires are ground connections for the common ground circuit.
 - The red, green, and other colored wires are used to connect the button pins and LED pins to their respective digital input/output pins on the Arduino Uno.

Each wire is connected to a specific port to interface the buttons and LEDs with the digital I/O pins of the Arduino, following the logic defined in the code. When the game is played, pressing a button sends a signal to the corresponding digital pin, triggering an interrupt in the code to measure reaction time, while the LEDs are controlled by the output signals from other digital pins based on the game logic.

Code:

~Must Install DHT sensor library 1.4.4~

```
#include <DHT.h>
#include <DHT_U.h>

// set pin numbers:
const int buttonPin1 = 2;  // the number of the pushbutton1 pin
const int buttonPin2 = 3;  // the number of the pushbutton2 pin
const int rstBtn = 9;      // the number of the rst button pin
const int ledPin1 = 5;     // the number of the LED1 anode(+) pin
const int ledPin2 = 7;     // the number of the LED2 anode(+) pin

int flag = -1;
int timer = 0;

void setup()
{
  pinMode(buttonPin1, INPUT);
  pinMode(buttonPin2, INPUT);
  pinMode(rstBtn, INPUT);
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);

  //enabling serial communication
  Serial.begin(115200);

  // set initial LED state
  digitalWrite(ledPin1, LOW);
  digitalWrite(ledPin2, LOW);

  attachInterrupt(digitalPinToInterrupt(buttonPin1), pressed1, FALLING);
  attachInterrupt(digitalPinToInterrupt(buttonPin2), pressed2, FALLING);

  Serial.println("GAME START...");
  Serial.println("Click the rst button to play...");
}

void loop()
{
  if (flag == -1 && digitalRead(rstBtn) == LOW) {
    digitalWrite(ledPin1, LOW);
    digitalWrite(ledPin2, LOW);
    Serial.println("Waiting...");
    int randomTime = random(2000, 5000);
    delay(randomTime);

    timer = millis();
    flag = randomTime % 2;
    Serial.println("Light!");
  }
}
```

```

    if (flag == 0) {
        digitalWrite(ledPin1, HIGH);
    } else if (flag == 1) {
        digitalWrite(ledPin2, HIGH);
    }
}
delay(200);
}

void pressed1() {
    if (flag == -1) {
        return;
    }
    if (flag == 0) {
        int currentTime = millis();
        Serial.print("Correct! You reaction time is : ");
        Serial.print(currentTime - timer);
        Serial.println(" ms");
    } else if (flag == 1) {
        Serial.println("Wrong Click!");
    }
    flag = -1;
}

void pressed2() {
    if (flag == -1) {
        return;
    }
    if (flag == 1) {
        int currentTime = millis();
        Serial.print("Correct! You reaction time is : ");
        Serial.print(currentTime - timer);
        Serial.println(" ms");
    } else if (flag == 0) {
        Serial.println("Wrong Click!");
    }
    flag = -1;
}

```

How to Play:

The Reaction Game on the SunFounder 3in1 Kit involves using three buttons and two LEDs to measure your visual reaction time. Here's how you play:

1. **Setup:** The game uses the Arduino's serial monitor to display messages. Initially, the monitor will show "waiting..."
2. **Starting the Game:** You begin the game by pressing the Ready button.
3. **Random LED Lighting:** After pressing the Ready button (Button 3), one of the two LEDs will light up at a random time. This time interval is randomly chosen by the Arduino.
4. **Your Reaction:** As soon as an LED lights up, you need to press the corresponding button as quickly as possible.
5. **Measuring Reaction Time:** The Arduino measures the time difference between the LED lighting up and your button press. This time difference is your reaction time.
6. **Result Display:** The Arduino then prints your measured reaction time on the serial monitor.