Creating a traffic light controller in Tinkercad is a great way to learn how to simulate a simple Arduino project. Here's a step-by-step guide to creating a traffic light system using three LEDs (red, yellow, and green):

Step 1: Create a New Tinkercad Project

1. Log in to Tinkercad or create an account if you haven't already.
2. Click on "Create New Circuit" to start a new project.

Step 2: Add Components

1. In the Components panel on the right, search for and add the following components to your workspace:
   * Arduino UNO
   * 3 LEDs (Red, Yellow, and Green)
   * 3 220-ohm resistors (one for each LED)
   * 3 push-button switches
2. Arrange the components on your workspace and wire them as follows:
   * Connect the longer leg (anode) of each LED to a 220-ohm resistor. Connect the other end of each resistor to a different digital pin on the Arduino (e.g., 2, 3, and 4).
   * Connect the shorter leg (cathode) of each LED to a common ground rail on your breadboard.
   * Connect one terminal of each push-button switch to a different digital pin on the Arduino (e.g., 5, 6, and 7).
   * Connect the other terminal of each push-button switch to a common ground rail on your breadboard.

Your setup should look something like this:

Step 3: Write the Arduino Code

1. Click on the Arduino UNO board to open the code editor.
2. Copy and paste the following Arduino code:

arduinoCopy code

// Define the pin numbers for the LEDs

int redLED = 2;

int yellowLED = 3;

int greenLED = 4;

// Define the pin numbers for the push-button switches

int redButton = 5;

int yellowButton = 6;

int greenButton = 7;

void setup() {

// Set LED pins as OUTPUT

pinMode(redLED, OUTPUT);

pinMode(yellowLED, OUTPUT);

pinMode(greenLED, OUTPUT);

// Set button pins as INPUT\_PULLUP

pinMode(redButton, INPUT\_PULLUP);

pinMode(yellowButton, INPUT\_PULLUP);

pinMode(greenButton, INPUT\_PULLUP);

}

void loop() {

// Check the state of each button

int redState = digitalRead(redButton);

int yellowState = digitalRead(yellowButton);

int greenState = digitalRead(greenButton);

// Traffic light control logic

if (redState == LOW) {

// Red light

digitalWrite(redLED, HIGH);

digitalWrite(yellowLED, LOW);

digitalWrite(greenLED, LOW);

} else if (yellowState == LOW) {

// Yellow light

digitalWrite(redLED, LOW);

digitalWrite(yellowLED, HIGH);

digitalWrite(greenLED, LOW);

} else if (greenState == LOW) {

// Green light

digitalWrite(redLED, LOW);

digitalWrite(yellowLED, LOW);

digitalWrite(greenLED, HIGH);

} else {

// All lights off

digitalWrite(redLED, LOW);

digitalWrite(yellowLED, LOW);

digitalWrite(greenLED, LOW);

}

}

Step 4: Simulate and Test

1. Click the "Start Simulation" button to run your simulation.
2. Press the push-button switches to change the traffic light sequence. Each button corresponds to a different traffic light state.
3. Observe how the LEDs change their states based on the button presses. You should see the traffic light sequence (Red -> Yellow -> Green) in response to the button presses.

That's it! You've successfully created a traffic light controller in Tinkercad. This project helps you understand the basics of input and output control using Arduino and simulates a simple traffic light system. You can further expand on this project by adding features like pedestrian crossing signals or a timer.