This code is an **Arduino program for a Raspberry Pi Pico** that uses **capacitive touch sensing** to control an **Adafruit NeoPixel LED**. Below is a breakdown of its components:

**Key Libraries and Hardware Setup**

* **#include <Arduino.h>**
  + Standard Arduino library for compatibility.
* **#include <Adafruit\_NeoPixel.h>**
  + Controls NeoPixel (RGB LED) on PIN 12 with NUMPIXELS = 1.
* **#include <pico/stdlib.h> and #include <hardware/gpio.h>**
  + Provides Raspberry Pi Pico-specific functions (e.g., GPIO and timing).

**Pin Definitions**

* Power = 11 → Powers an external circuit.
* PIN = 12 → Controls the **NeoPixel LED**.
* LED\_PIN = 25 → Built-in LED on the Pico.
* THRESHOLD = 80 → Defines sensitivity for capacitive touch.

**Interrupt Variables**

These are used to measure time differences when sensing touch:

* **volatile bool flag2int = false;** → Indicates an interrupt occurred.
* **volatile uint32\_t start\_moment, trigger\_moment;** → Stores timestamps.
* **volatile int32\_t t\_diff = 0;** → Holds the measured time difference.

**NeoPixel Initialization**

cpp

CopyEdit

Adafruit\_NeoPixel pixels(NUMPIXELS, PIN, NEO\_GRB + NEO\_KHZ800);

* Creates a single RGB LED (NUMPIXELS = 1) on **pin 12**.
* Uses **NEO\_GRB** color order and a **800kHz protocol**.

**Color and Pin Mode Enums**

cpp

CopyEdit

enum RGB\_color { CLEAR, R, G, B };

enum pinmode { Analog, Digital };

* Defines RGB colors for LED control.
* Defines modes for capacitive touch sensing.

**Calibration Function**

cpp

CopyEdit

int calibration(int pin) {

int sum = 0;

for (int i = 0; i < CALI\_CNT; i++) {

pinMode(pin, OUTPUT);

digitalWrite(pin, HIGH);

delayMicroseconds(10);

pinMode(pin, INPUT);

delayMicroseconds(100);

sum += analogRead(pin);

delay(50);

}

return (sum/CALI\_CNT);

}

* Measures the baseline capacitance of a pin over CALI\_CNT = 30 cycles.
* **Stores this value for touch detection.**

**RGB Flashing Function**

cpp

CopyEdit

void flash\_RGB(RGB\_color color) {

switch (color) {

case R: pixels.setPixelColor(0, pixels.Color(220, 0, 0)); break;

case G: pixels.setPixelColor(0, pixels.Color(0, 220, 0)); break;

case B: pixels.setPixelColor(0, pixels.Color(0, 0, 220)); break;

default: pixels.clear(); break;

}

pixels.show();

}

* **Lights up the NeoPixel** in **red, green, or blue**.
* **Clears the LED** when there's no touch.

**Touch Sensing Using Time Differences**

cpp

CopyEdit

RGB\_color sense\_touch(int pin, int cali, RGB\_color color, pinmode mode) {

if (mode == Analog) {

pinMode(pin, OUTPUT);

digitalWrite(pin, HIGH);

delayMicroseconds(10);

pinMode(pin, INPUT);

delayMicroseconds(100);

if (abs(analogRead(pin) - cali) > THRESHOLD) {

return color;

}

else {

return CLEAR;

}

}

else {

pinMode(pin, OUTPUT);

digitalWrite(pin, HIGH);

delayMicroseconds(10);

gpio\_init(pin);

start\_moment = time\_us\_32();

gpio\_set\_irq\_enabled\_with\_callback(pin, GPIO\_IRQ\_EDGE\_FALL, true, &timber\_counter\_call\_back);

delay(1);

if (t\_diff > 5) return color;

else return CLEAR;

}

}

* **Analog mode**: Compares capacitance reading to cali baseline.
* **Digital mode**: Uses an **interrupt** to detect rapid capacitance changes.

**Interrupt Handler for Touch Sensing**

cpp

CopyEdit

void timber\_counter\_call\_back(uint gpio, uint32\_t event\_mask) {

flag2int = true;

t\_diff = time\_us\_32() - start\_moment;

}

* **Captures time difference (t\_diff) when a touch is detected.**

**Setup Function (setup())**

cpp

CopyEdit

void setup() {

Serial.begin(115200);

pixels.begin();

pinMode(Power, OUTPUT);

digitalWrite(Power, HIGH);

pinMode(LED\_PIN, OUTPUT);

digitalWrite(LED\_PIN, LOW);

delay(3000);

cali\_1 = calibration(D0);

cali\_2 = calibration(D1);

cali\_3 = calibration(D2);

digitalWrite(LED\_PIN, HIGH);

}

* **Initializes NeoPixel LED**.
* **Calibrates touch-sensitive pins (D0, D1, D2)**.

**Main Loop (loop())**

cpp

CopyEdit

void loop() {

flash\_r = CLEAR;

flash\_g = CLEAR;

flash\_b = CLEAR;

RGB\_color c1 = R;

RGB\_color c2 = G;

RGB\_color c3 = B;

flash\_r = sense\_touch(D0, cali\_1, c1, Analog); flash\_RGB(flash\_r);

flash\_g = sense\_touch(D1, cali\_2, c2, Analog); flash\_RGB(flash\_g);

flash\_b = sense\_touch(D2, cali\_3, c3, Analog); flash\_RGB(flash\_b);

}

* **Checks for touches on D0, D1, D2.**
* **Lights up the NeoPixel accordingly**:
  + D0 → Red
  + D1 → Green
  + D2 → Blue

**Summary**

* **Detects capacitive touch on D0, D1, D2**.
* **Lights up a NeoPixel LED in red, green, or blue based on touch.**
* **Uses both analog and digital methods for touch detection.**
* **Uses an interrupt-based approach to improve response time.**