

Finger Count on 7-Segment Display using Python and Arduino Uno

What You Need :

- Arduino UNO
- Common Anode 7-Segment Display
- Jumper Wires
- Breadboard
- USB Cable for Arduino
- A laptop with Python and OpenCV

Software Setup :

1. **Install Arduino IDE** from <https://www.arduino.cc/en/software>
2. **Install Python** from <https://www.python.org/downloads>
3. Open CMD/Terminal and run:

```
pip install opencv-python mediapipe pyserial
```

Arduino to 7-Segment Display (Common Anode) Wiring :

Arduino Pin	Segment	Display Pin	Note
2	A	a	Resistor in between (Optional)
3	B	b	Resistor in between (Optional)
4	C	c	Resistor in between (Optional)
5	D	d	Resistor in between (Optional)
6	E	e	Resistor in between (Optional)
7	F	f	Resistor in between (Optional)
8	G	g	Resistor in between (Optional)
5V (or VCC)	Common	Common pins	Connect both common pins to 5V
GND			Common ground

Arduino Code :

```

const int segments[] = {2, 3, 4, 5, 6, 7, 8};
const byte digits[10][7] = {
    {0,0,0,0,0,0,1}, // 0
    {1,0,0,1,1,1,1}, // 1
    {0,0,1,0,0,1,0}, // 2
    {0,0,0,0,1,1,0}, // 3
    {1,0,0,1,1,0,0}, // 4
    {0,1,0,0,1,0,0}, // 5
    {0,1,0,0,0,0,0}, // 6
    {0,0,0,1,1,1,1}, // 7
    {0,0,0,0,0,0,0}, // 8
    {0,0,0,1,1,0,0} // 9
};

void setup() {
    for (int i = 0; i < 7; i++) pinMode(segments[i], OUTPUT);
    Serial.begin(9600);
}

void displayDigit(int digit) {
    for (int i = 0; i < 7; i++) digitalWrite(segments[i], digits[digit][i]);
}

void loop() {
    if (Serial.available()) {
        int val = Serial.read() - '0';
        if (val >= 0 && val <= 9) displayDigit(val);
    }
}

```

Python Code for Finger Counting :

```
import cv2
import mediapipe as mp
import serial
import time

arduino = serial.Serial('COM7', 9600)
time.sleep(2)

mp_hands = mp.solutions.hands
hands = mp_hands.Hands(max_num_hands=1, min_detection_confidence=0.7,
min_tracking_confidence=0.7)
mp_draw = mp.solutions.drawing_utils

finger_tips = [4, 8, 12, 16, 20]
cap = cv2.VideoCapture(0)
prev_count = -1

while True:
    ret, frame = cap.read()
    if not ret:
        break

    frame = cv2.flip(frame, 1)
    h, w, _ = frame.shape
    rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    results = hands.process(rgb)

    finger_count = 0

    if results.multi_hand_landmarks:
        for hand_landmarks in results.multi_hand_landmarks:
```

```

lm_list = [(int(lm.x * w), int(lm.y * h)) for lm in hand_landmarks.landmark]

if lm_list[finger_tips[0]][0] > lm_list[finger_tips[0] - 1][0]:
    finger_count += 1

for id in finger_tips[1:]:
    if lm_list[id][1] < lm_list[id - 2][1]:
        finger_count += 1

mp_draw.draw_landmarks(frame, hand_landmarks, mp_hands.HAND_CONNECTIONS)

if finger_count != prev_count:
    arduino.write(str(finger_count).encode())
    prev_count = finger_count

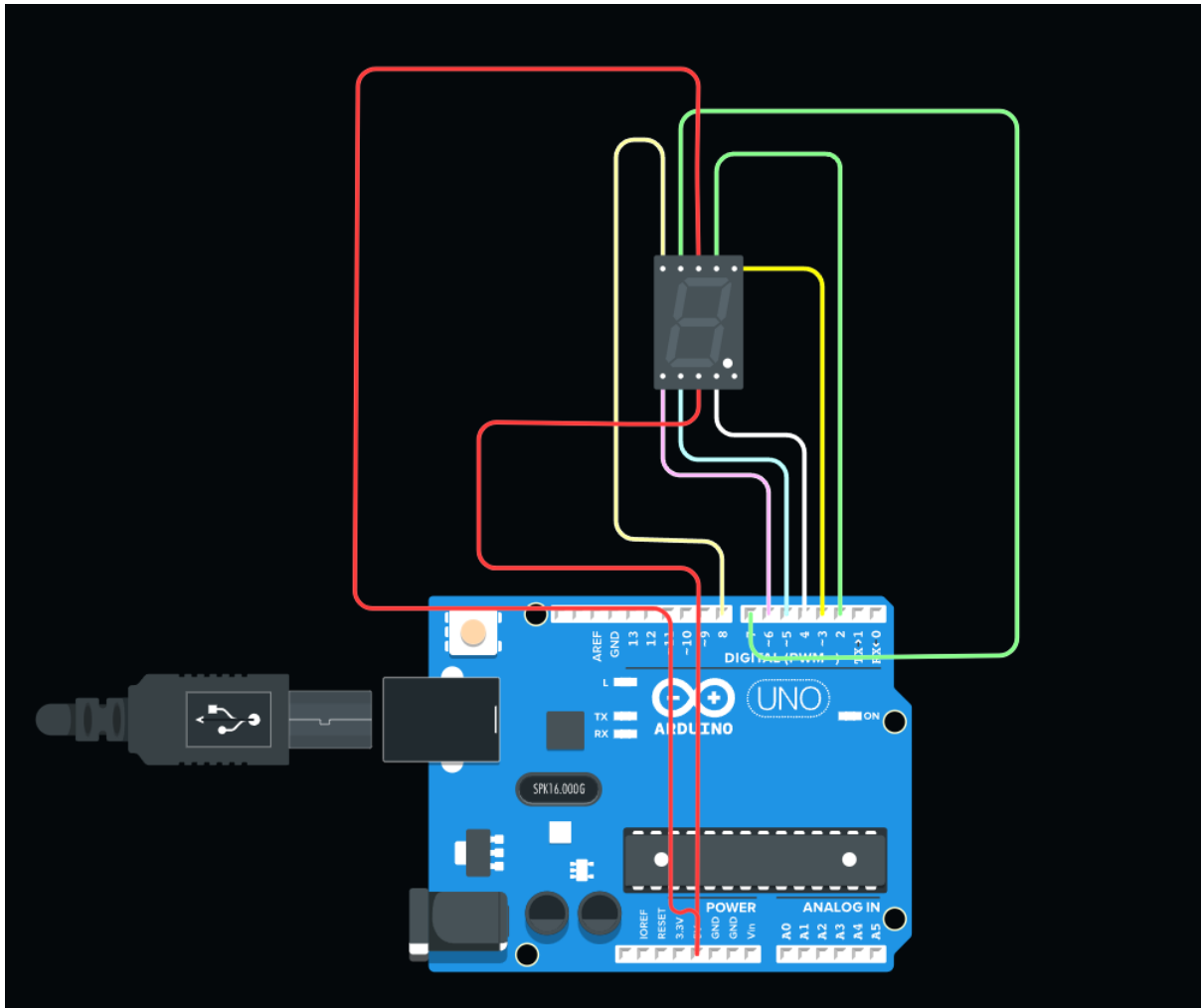
cv2.putText(frame, f"Fingers: {finger_count}", (10, 60),
            cv2.FONT_HERSHEY_SIMPLEX, 1.5, (0, 255, 0), 3)

cv2.imshow("Hand Gesture Counter", frame)
if cv2.waitKey(1) & 0xFF == ord('q'):
    break

cap.release()
cv2.destroyAllWindows()
arduino.close()

```

Circuit Diagram :





Follow these simple steps to run the project:

1. Connect the Hardware

- Assemble the components as shown in the Circuit Diagram.
- Double-check jumper wires and resistor placement.
- Plug in the Arduino to your laptop via USB.

2. Upload Arduino Code

- Open the Arduino IDE.
- Copy and paste the Arduino code from this guide.
- Select your correct COM port under *Tools > Port*.
- Select the Arduino Uno board under *Tools > Board*.
- Click the  checkmark to compile and then  Upload.

3. Run Python Code

- Open a terminal or CMD window.
- Navigate to the folder containing your .py file.
- Type and run:

4. Interact and Observe

- Raise your fingers in front of the webcam.
 - You'll see the number of fingers displayed on the 7-segment display!
 - Press q to quit the Python program safely.
-