

Uputa: Kako koristiti tipkalo (push-button) na Arduino UNO

1. Što je tipkalo?

Tipkalo (push-button) je jednostavna elektronička komponenta koja se **zatvara kad se pritisne**, a **otvara kad se otpusti**.

Drugim riječima:

- pritisnuto → strujni krug je zatvoren
- otpušteno → krug je otvoren

Tipkalo samo po sebi **nema polaritet** – može se okrenuti kako god želimo..

2. Kako se tipkalo ponaša u strujnom krugu?

Tipkalo radi kao **prekidač**:

- dok nije pritisnuto → prekida struju
- kad se pritisne → povezuje dva izvoda, ili jedan izvod i jedan pol napajanja.

U Arduino projektima tipkala se obično koriste za:

- pokretanje nekog događaja
- promjenu načina rada
- okidanje mjerenja
- "resetiranje" vrijednosti

3. Problem "lebdećeg" ulaza (floating input)

Ako digitalni ulaz Arduina nije ni na +5 V ni na GND, nego "visi", on može očitavati:

- slučajne 0 i 1
- smetnje iz zraka
- nestabilne vrijednosti

To se zove **floating input**.

Zbog toga se koriste **pull-up** ili **pull-down otpornici**.

4. Pull-up i pull-down otpornici

Pull-up otpornik

Ulaz je inače **HIGH** (1), a kad pritisneš tipkalo → spoji se na GND → **LOW** (0)

✓ Pull-down otpornik

Ulaz je inače **LOW**, a pritisak ga vodi na +5 V → **HIGH**

Na Arduino UNO najčešće koristimo internal pull-up – ugrađeni otpornik

Arduino ima ugrađeni pull-up od ~20–50 kΩ.

To znači da **ne trebamo koristiti vanjski otpornik** – sve radi samo uz tipkalo i vodiče.

5. Kako spojiti tipkalo na Arduino? (preporučeni način)

Shema (s internal pull-up):

Tipkalo jedna nožica → GND

Tipkalo druga nožica → digitalni izvod (npr. D2)

```
pinMode(D2, INPUT_PULLUP);
```

Kad tipkalo nije pritisnuto → izvod = HIGH

Kad se pritisne → izvod = LOW

Ovaj način je najstabilniji i najjednostavniji.

6. Primjer koda – očitavanje tipkala

```
const int buttonPin = 2;
```

```
void setup() {  
  pinMode(buttonPin, INPUT_PULLUP); // uključi pull-up otpornik  
  Serial.begin(9600);  
}
```

```
void loop() {  
  int state = digitalRead(buttonPin);
```

```
  if (state == LOW) {  
    Serial.println("Pritisnuto!");  
  } else {  
    Serial.println("Otpušteno");  
  }  
}
```

```
  delay(100);  
}
```

7. Debounce – zašto tipkalo ponekad “drhti”?

Mehaničko tipkalo pri pritisku **poskakuje** (bouncing):
u vrlo kratkom vremenu (2–10 ms) brzo otvara i zatvara kontakt.

Arduino to vidi kao:
0001111010001110 → iako je tipkalo pritisnuto samo jednom.

To se rješava:

- softverskim “debounce” kodom
- delay-em oko 20 ms
- ili boljim algoritmom (npr. merenjem stabilnog stanja)

Jednostavan debounce:

```
bool readButton(int pin) {
  static unsigned long lastTime = 0;
  static bool lastState = HIGH;

  bool current = digitalRead(pin);
  if (current != lastState) {
    lastTime = millis();
  }
  if (millis() - lastTime > 20) {
    lastState = current;
  }
  return lastState;
}
```

8. Primjer zadatka za (jednostavan)

Zadatak:

Napiši program koji pali LED kada je tipkalo pritisnuto, a gasi je kad nije.

Rješenje:

```
const int buttonPin = 2;
const int ledPin = 8;

void setup() {
  pinMode(buttonPin, INPUT_PULLUP);
  pinMode(ledPin, OUTPUT);
}

void loop() {
  int state = digitalRead(buttonPin);

  if(state == LOW) digitalWrite(ledPin, HIGH);
  else      digitalWrite(ledPin, LOW);
}
```

9. Česte greške učenika

- ✘ nisu uključili INPUT_PULLUP → ulaz “pleše”
- ✘ spojili tipkalo na +5 V umjesto GND → logika se obrne
- ✘ zaboravili staviti pinMode
- ✘ spojili LED bez otpornika
- ✘ brkanje tipkala i prekidača (tipkalo se vraća samo)

10. Zaključak

Tipkalo je ključna komponenta za učenje:

- digitalnog očitavanja
- rada s ulazima
- korištenja pull-up otpornika
- debounce logike
- logičkih stanja HIGH i LOW

Odlično se nadovezuje na LED, jer pokazuje **kako ulaz upravlja izlazom.**