**Πείραμα με 3 toggle switches**

1. **Περιγραφή:**

Στο πείραμα που ακολουθεί χρησιμοποιώ διακόπτες ενεργοποίησης(toggle switch) για να δω πώς λειτουργούν συνδεδεμένοι σε πλακέτα Raspberry Pi και τί συμπεριφορά έχουν ανάλογα το είδος του καθένα.

1. **Υλικά:**

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **1\* Raspberry pi** | **1\* GPIO Extension board** | **1\* 40 pin colorful jumper wires** |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **1\* Breadboard** | **Jumper Wires** | **1\*Three (3) toggle switches** |

1. **Γνώση υλικού:**

Οι διακόπτες εναλλαγής είναι μηχανικές συσκευές που δημιουργούν ή διακόπτουν ηλεκτρικές συνδέσεις σε ένα κύκλωμα με το κλείσιμο ή το άνοιγμα ηλεκτρικών επαφών όταν ο προεξέχων ενεργοποιητής μετακινείται χειροκίνητα (εναλλαγή) από τη μία θέση στην άλλη. Οι εναλλασσόμενοι διακόπτες ορίζονται από τον αριθμό των πόλων και των διαδρομών τους.

**Πειραματική συσκευή:**

■ Πίνακας ελέγχου Raspberry Pi

■ Μονάδα διακοπτών

■ Γραμμές σύνδεσης

**Πειραματική σύνδεση:**

|  |  |
| --- | --- |
| Three Toggle-switch module | Raspberry Pi |
| Vcc | 5V |
| Gd | GND |
| S0 , S1 ,S2 , S3 , S4 ,S5 | 17 , 12 , 27 , 22 , 23 , 24 |

1. **Πειραματικό συμπέρασμα:**

Στο πείραμα ο κώδικας διαβάζει τις καταστάσεις των τριών εναλλασσόμενων διακοπτών που είναι συνδεδεμένοι σε ακίδες GPIO και εξάγει τις καταστάσεις με τη μορφή που έχετε καθορίσει (s0, s1, s2, s3, s4, s5). Μπορείτε να χρησιμοποιήσετε αυτές τις πληροφορίες για να ελέγξετε διάφορες πτυχές του έργου σας Raspberry Pi.

**Βεβαιωθείτε ότι έχετε εγκαταστήσει τη βιβλιοθήκη RPi.GPIO στο Raspberry Pi σας, αν δεν είναι ήδη εγκατεστημένη. Μπορείτε να το κάνετε αυτό με την ακόλουθη εντολή στη γραμμή εντολών(command line):**

**pip install RPi.GPIO**

Ανεβάστε τον κώδικα στην αναπτυξιακή πλακέτα και εκτελέστε τον.

1. **Κώδικας Python:**

import RPi.GPIO as GPIO

import time

# Define GPIO pins for toggle switches

S0\_PIN = 17

S1\_PIN = 12

S2\_PIN = 27

S3\_PIN = 22

S4\_PIN = 23

S5\_PIN = 24

# Set the GPIO mode and setup the pins as inputs

GPIO.setmode(GPIO.BCM)

GPIO.setup([S0\_PIN, S1\_PIN, S2\_PIN, S3\_PIN, S4\_PIN, S5\_PIN], GPIO.IN,GPIO.PUD\_DOWN)

try:

    while True:

        # Read the state of each toggle switch

        state\_s0 = GPIO.input(S0\_PIN)

        state\_s1 = GPIO.input(S1\_PIN)

        state\_s2 = GPIO.input(S2\_PIN)

        state\_s3 = GPIO.input(S3\_PIN)

        state\_s4 = GPIO.input(S4\_PIN)

        state\_s5 = GPIO.input(S5\_PIN)

        # Print the state of each toggle switch

        print(f"S0-S1: {state\_s0}-{state\_s1}, S2-S3: {state\_s2}-{state\_s3}, S4-S5: {state\_s4}-{state\_s5}")

        # Add your code here to perform actions based on the switch states

        # For example, you can check each switch's state and take appropriate actions

        # Wait for a short duration to avoid continuous printing

        time.sleep(0.1)

except KeyboardInterrupt:

    print("Exiting program")

finally:

    # Cleanup GPIO settings on program exit

    GPIO.cleanup()