**Πείραμα για Proximity (PIR Sensor)**

1. **Περιγραφή:**

Στο παρακάτω πείραμα χρησιμοποιώ το Proximity Sensor Module (PIR) για να ανιχνεύσω κίνηση στο χώρο.Με τη χρήση του κατάλληλου Python κώδικα έχω ένδειξη για πιθανή κίνηση.

1. **Υλικά:**

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

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **1\* Breadboard** | **Jumper Wires** | **1\* Proximity Sensor Module** |

1. **Γνώση υλικού:**

Ο αισθητήρας PIR (παθητική υπέρυθρη ακτινοβολία) μπορεί να ανιχνεύσει τις υπέρυθρες ακτίνες που εκπέμπονται από ανθρώπους ή ζώα και στη συνέχεια να εξάγει σήματα On/Off. Ο παραδοσιακός πυροηλεκτρικός αισθητήρας PIR χρειάζεται πυροηλεκτρικό ανιχνευτή υπερύθρων, ειδικό τσιπ και πολύπλοκα κυκλώματα για την επίτευξη του αποτελέσματος. Σε αυτή την περίπτωση, ο αισθητήρας είναι μεγάλος με πολύπλοκα κυκλώματα και συγκριτικά λιγότερο αξιόπιστος.

Αυτός ο πυροηλεκτρικός αισθητήρας κίνησης υπερύθρων υιοθετεί την ψηφιακή ολοκλήρωση του αισθητήρα, έτσι ώστε να έχει υψηλή αξιοπιστία, χαμηλή κατανάλωση ενέργειας και απλό εξωτερικό κύκλωμα με μικρό μέγεθος. Μπορεί να εφαρμοστεί σε οποιεσδήποτε περιπτώσεις ανίχνευσης κινούμενων ανθρώπων ή ζώων.

**Πειραματική συσκευή:**

■ Πίνακας ελέγχου Raspberry Pi

■ Μονάδα αισθητήρα Proximity

■ Γραμμές σύνδεσης

**Πειραματική σύνδεση:**

|  |  |
| --- | --- |
| The Proximity module | Raspberry Pi |
| Vcc | 5V |
| Gd | GND |
| OUT | 17 |

1. **Πειραματικό συμπέρασμα:**

Στο συγκεκριμμένο σενάριο διαβάζει την έξοδο του αισθητήρα PIR και εκτυπώνει το "Motion detected" όταν ανιχνεύεται κίνηση και το "No motion" όταν δεν ανιχνεύεται κίνηση. Βεβαιωθείτε ότι έχετε προσαρμόσει τον αριθμό των ακροδεκτών GPIO στον κώδικα ώστε να ταιριάζει με την πραγματική σας καλωδίωση**.**

**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**

1. **Κώδικας Python:**

import RPi.GPIO as GPIO

import time

# Set the GPIO mode

GPIO.setmode(GPIO.BCM)

# Define the GPIO pin for the PIR sensor's output

pir\_sensor\_pin = 17  # Replace with the actual GPIO pin number

# Initialize the sensor pin as an input

GPIO.setup(pir\_sensor\_pin, GPIO.IN)

try:

    while True:

        sensor\_state = GPIO.input(pir\_sensor\_pin)

        if sensor\_state == GPIO.HIGH:

            print("Motion detected")

        else:

            print("No motion")

        time.sleep(0.1)  # Delay between readings

except KeyboardInterrupt:

    pass

finally:

    GPIO.cleanup()  # Clean up GPIO on script exit

**Παραλλαγή Πειράματος:**

Σε αυτό το σενάριο διαβάζω την έξοδο (OUT) από τον αισθητήρα Proximity και ενεργοποιώ τον ενεργό βομβητή όταν ανιχνεύεται μαγνητικό πεδίο.

|  |  |
| --- | --- |
| The Proximity module | Raspberry Pi |
| Vcc | 5V |
| Gd | GND |
| OUT | 17 |

|  |  |
| --- | --- |
| The Buzzer module | Raspberry Pi |
| Vcc | 5V |
| Gd | GND |
| S | 18 |

**Κώδικας Python:**

import RPi.GPIO as GPIO

import time

# Set the GPIO mode

GPIO.setmode(GPIO.BCM)

# Define the GPIO pin for the PIR sensor's output

pir\_sensor\_pin = 17  # Replace with the actual GPIO pin number

# Define the GPIO pin for the buzzer module's signal input

buzzer\_pin = 18  # Replace with the actual GPIO pin number

# Initialize the sensor pin as an input

GPIO.setup(pir\_sensor\_pin, GPIO.IN)

# Initialize the buzzer pin as an output

GPIO.setup(buzzer\_pin, GPIO.OUT)

GPIO.output(buzzer\_pin, GPIO.LOW)  # Turn off the buzzer initially

try:

    while True:

        sensor\_state = GPIO.input(pir\_sensor\_pin)

        if sensor\_state == GPIO.HIGH:

            print("Motion detected")

            GPIO.output(buzzer\_pin, GPIO.HIGH)  # Turn on the buzzer

        else:

            print("No motion")

            GPIO.output(buzzer\_pin, GPIO.LOW)  # Turn off the buzzer

        time.sleep(0.1)  # Delay between readings

except KeyboardInterrupt:

    pass

finally:

    GPIO.cleanup()  # Clean up GPIO on script exit