



2023

1. Εφεδρικός αισθητήρας αυτοκινήτου DIY

Αρ. έργου: **2021-1-FR01-KA220-SCH-000031617**



 **Co-funded by
the European Union**

The European Commission's support for the production of this publication does not constitute an endorsement of the contents, which reflect the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

SCRAPY Partnership
31/05/2023

Πίνακας περιεχομένων

Πείραμα 1: Αισθητήρας εφεδρικού αυτοκινήτου DIY	2
Σύντομη περιγραφή	2
Εκτεταμένη περιγραφή	2
Στόχοι:	3
Υλικά που θα χρησιμοποιηθούν:	3
Βήματα που πρέπει να ακολουθηθούν:	4
Διάγραμμα συνδεσμολογίας	5
Κώδικας	7
Παραδείγματα εικόνων	8
Συμπέρασμα	9

Πείραμα 1: Αισθητήρας εφεδρικού αυτοκινήτου DIY

Σύντομη περιγραφή

Δημιουργήστε ένα σύστημα οπισθοπορείας αυτοκινήτου με το Raspberry Pi Pico και έναν αισθητήρα υπερήχων.

Εκτεταμένη περιγραφή

Θεωρείτε λίγο δύσκολο το να κάνετε όπισθεν; Ανησυχείτε μήπως χτυπήσετε κάτι ή κάποιον κατά το παρκάρισμα του αυτοκινήτου; Λοιπόν, μην ανησυχείτε άλλο! Με τη βοήθεια της πλακέτας Raspberry Pi Pico και ενός αισθητήρα υπερήχων HC-SR04, μπορείτε να δημιουργήσετε το δικό σας σύστημα ραντάρ οπισθοπορείας που θα κάνει το παρκάρισμα του αυτοκινήτου σας παιχνιδάκι.

- Αυτό το DIY πρότζεκτ είναι τέλειο για όποιον ενδιαφέρεται για τα ηλεκτρονικά και τη μηχανική και είναι κατάλληλο για όλα τα επίπεδα δεξιοτήτων. Αυτό το ραντάρ οπισθοπορείας είναι βέβαιο ότι θα σας προκαλέσει και θα σας ενθουσιάσει.
- Εκτός από τον αισθητήρα Raspberry Pi Pico και HC-SR04, αυτό το έργο απαιτεί επίσης ένα βομβητή, τρεις λυχνίες LED (πράσινο, κίτρινο και κόκκινο), τρεις αντιστάσεις 220 ohm, μια πλακέτα ταχείας πρωτοτυποποίησης και καλώδια σύνδεσης. Με αυτά τα εξαρτήματα, θα μπορείτε να δημιουργήσετε ένα αξιόπιστο και ακριβές ραντάρ οπισθοπορείας που θα σας βοηθήσει να παρκάρετε το αυτοκίνητό σας με ευκολία.
- Ακολουθώντας τις οδηγίες βήμα προς βήμα, θα μάθετε πώς να συνδέσετε τα εξαρτήματα μεταξύ τους, να προγραμματίσετε την πλακέτα Raspberry Pi Pico και να ελέγξετε το σύστημα για να βεβαιωθείτε ότι λειτουργεί σωστά. Θα μάθετε επίσης για τη φυσική των αισθητήρων υπερήχων και πώς μπορούν να χρησιμοποιηθούν για τη μέτρηση της απόστασης.

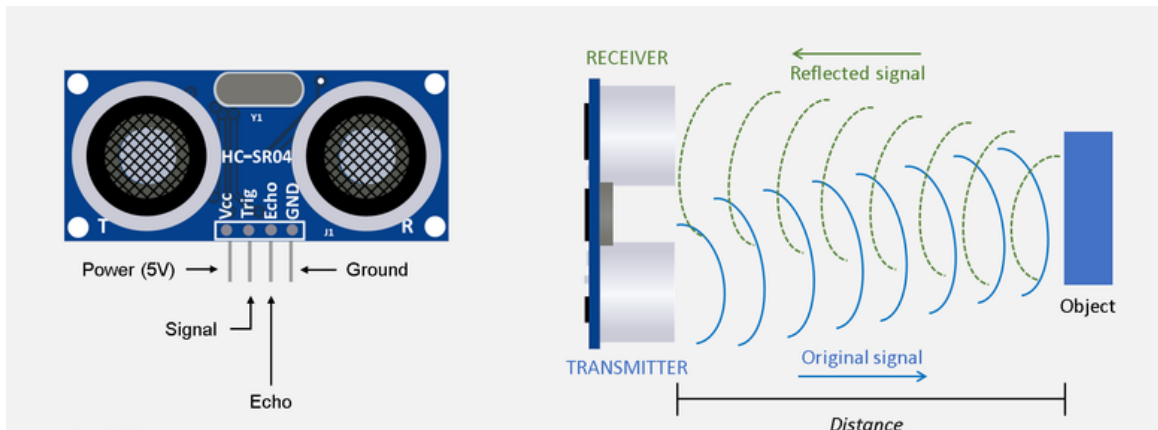
Οι αρχές λειτουργίας του αισθητήρα υπερήχων HC-SR04;

Ο αισθητήρας υπερήχων HC-SR04 λειτουργεί εκπέμποντας ηχητικά κύματα υψηλής συχνότητας που δεν ακούγονται στο ανθρώπινο αυτί. Αυτά τα ηχητικά κύματα ταξιδεύουν στον αέρα και αναπηδούν από αντικείμενα στο πέρασμά τους. Όταν τα ηχητικά κύματα χτυπούν ένα αντικείμενο, αντανακλώνονται πίσω στον αισθητήρα, ο οποίος μετρά τον χρόνο που χρειάζεται για να αναπηδήσουν τα κύματα. Γνωρίζοντας την ταχύτητα του ήχου και το χρόνο που χρειάζονται για να ταξιδέψουν τα κύματα προς το αντικείμενο και πίσω, ο αισθητήρας μπορεί να υπολογίσει την απόσταση από το αντικείμενο.

Ακολουθούν τα κύρια βήματα στις αρχές λειτουργίας του αισθητήρα υπερήχων HC-SR04:

- Ο αισθητήρας στέλνει ένα ηχητικό κύμα υψηλής συχνότητας (συνήθως περίπου 40 kHz) από τον πομπό του.
- Το ηχητικό κύμα ταξιδεύει στον αέρα και αναπηδά από ένα αντικείμενο.
- Το ανακλώμενο ηχητικό κύμα ανιχνεύεται από τον δέκτη του αισθητήρα.

- Ο αισθητήρας μετρά το χρόνο που χρειάζεται για να ταξιδέψει το ηχητικό κύμα στο αντικείμενο και πίσω.
- Ο αισθητήρας υπολογίζει την απόσταση από το αντικείμενο με βάση το χρόνο που χρειάστηκε για να ταξιδέψει το ηχητικό κύμα προς το αντικείμενο και πίσω.



Συνολικά, ο αισθητήρας υπερήχων HC-SR04 είναι ένας αξιόπιστος και ακριβής τρόπος μέτρησης αποστάσεων και χρησιμοποιείται συνήθως σε εφαρμογές όπως η ρομποτική, ο αυτοματισμός και η ανίχνευση απόστασης.

Στόχοι:

Μέσω αυτής της δραστηριότητας, ο χρήστης θα πειραματιστεί στην κατασκευή ενός συστήματος ραντάρ αναστροφής χρησιμοποιώντας την πλακέτα Raspberry Pi Pico και έναν αισθητήρα υπερήχων HC-SR04. Ο χρήστης θα αποκτήσει γνώσεις σχετικά με:

- Την φυσική των υπερηχητικών κυμάτων και πώς μπορούν να χρησιμοποιηθούν για τη μέτρηση της απόστασης.
- Τα βασικά του προγραμματισμού στην Python και πώς να γράψει κώδικα για τον έλεγχο του πίνακα Raspberry Pi Pico.
- Τις αρχές του σχεδιασμού του κυκλώματος και τον τρόπο σύνδεσης εξαρτημάτων σε μια πλακέτα ταχείας πρωτοτυποποίησης για τη δημιουργία ενός λειτουργικού συστήματος ραντάρ οπισθοπορείας.

Ολοκληρώνοντας αυτό το πρότζεκτ, ο χρήστης θα αποκτήσει μια βαθύτερη κατανόηση των ηλεκτρονικών, της μηχανικής και του προγραμματισμού. Θα έχει επίσης μια πρακτική και χρήσιμη συσκευή που μπορεί να χρησιμοποιήσει για να κάνει το παρκάρισμα του αυτοκινήτου ασφαλέστερο και πιο βολικό.

Υλικά που θα χρησιμοποιηθούν:

- 1 x Raspberry Pi Pico
- 1 x κιτ breadboard Pico
- 1 x breadboard πλήρους μεγέθους

- 1 x HC-SR04 αισθητήρας υπερήχων
- 1 x Βομβητής
- 3 x LED (πράσινα, κίτρινα και κόκκινα)
- Αντιστάσεις 3 x 220 ohm
- Καλώδια βραχυκυκλωτήρα

Βήματα που πρέπει να ακολουθηθούν:

Τα κύρια βήματα για να πραγματοποιήσετε το πείραμα ραντάρ αναστροφής με την πλακέτα Raspberry Pi Pico και τον αισθητήρα υπερήχων HC-SR04:

1. Συνδέστε τον αισθητήρα υπερήχων HC-SR04 στην πλακέτα Raspberry Pi Pico χρησιμοποιώντας καλώδια σύνδεσης.
2. Συνδέστε το βομβητή και τις λυχνίες LED στην πλακέτα Raspberry Pi Pico χρησιμοποιώντας καλώδια σύνδεσης και τις αντιστάσεις 220 ohm για να περιορίσετε τη ροή ρεύματος.
3. Γράψτε ένα πρόγραμμα Python για τον έλεγχο της πλακέτας Raspberry Pi Pico και χρησιμοποιήστε τον αισθητήρα HC-SR04 για να μετρήσετε αποστάσεις.
4. Προγραμματίστε την πλακέτα Raspberry Pi Pico να ανάβει την πράσινη λυχνία LED όταν δεν υπάρχει εμπόδιο, την κίτρινη λυχνία LED όταν το εμπόδιο βρίσκεται εντός ορισμένου εύρους και την κόκκινη λυχνία LED όταν το εμπόδιο είναι πολύ κοντά.
5. Προγραμματίστε την πλακέτα Raspberry Pi Pico ώστε να ενεργοποιεί το βομβητή όταν το εμπόδιο είναι πολύ κοντά.
6. Δοκιμάστε το σύστημα ραντάρ οπισθοπορείας τοποθετώντας εμπόδια σε διάφορες αποστάσεις και γωνίες πίσω από το ραντάρ και βεβαιωθείτε ότι τα LED και ο βομβητής παρέχουν την κατάλληλη ανάδραση.

Για το Βήμα 4: Για να μελετήσετε τη λειτουργία του συστήματος, είναι απαραίτητο να προγραμματίσετε τη συμπεριφορά του. Οι τιμές απόστασης μπορούν να ρυθμιστούν σύμφωνα με συγκεκριμένες απαιτήσεις. Ο προγραμματισμός πρέπει να ακολουθεί τους ακόλουθους κανόνες:

- Το πράσινο LED θα πρέπει να είναι συνεχώς αναμμένο εάν η απόσταση μεταξύ του αισθητήρα και οποιουδήποτε εμποδίου είναι μεγαλύτερη από 20 cm (1 μέτρο).
- Το πορτοκαλί LED πρέπει να είναι αναμμένο εάν η απόσταση μεταξύ του αισθητήρα και οποιουδήποτε εμποδίου είναι μεταξύ 20 cm και 5 cm.
- Το κόκκινο LED πρέπει να είναι αναμμένο εάν η απόσταση μεταξύ του αισθητήρα και οποιουδήποτε εμποδίου είναι μικρότερη από 5 cm.

Προγραμματίζοντας το σύστημα να ακολουθεί αυτούς τους κανόνες, θα παρέχει σαφή οπτική ανάδραση της απόστασης μεταξύ του αισθητήρα και οποιουδήποτε εμποδίου.

Αυτοί οι κανόνες μπορούν να προσαρμοστούν όπως απαιτείται για να ταιριάζουν σε συγκεκριμένες απαιτήσεις ή περιπτώσεις χρήσης.

Διάγραμμα συνδεσμολογίας

Ακολουθεί ένα σχέδιο καλωδίωσης για το πρότζεκτ με τη χρήση της πλακέτας Raspberry Pi Pico, τον αισθητήρα υπερήχων HC-SR04, τον βομβητή και τρεις λυχνίες LED:

Raspberry Pi Pico Board:

- GP15: Ακίδα ενεργοποίησης του αισθητήρα HC-SR04
- GP14: Ακίδα ήχου του αισθητήρα HC-SR04
- GP10: Θετική ακίδα του πράσινου LED
- GP11: Θετική ακίδα του πορτοκαλί LED
- GP12: Θετική ακίδα του κόκκινου LED
- GP2: Θετική ακίδα του βομβητή
- GND: Ακίδα γείωσης της πλακέτας

Αισθητήρας HC-SR04:

- VCC: Σύνδεση σε πηγή ρεύματος 5V
- GND: Σύνδεση στο GND της πλακέτας Raspberry Pi Pico
- Trig: Σύνδεση στο GP15 της πλακέτας Raspberry Pi Pico
- Echo: Σύνδεση στο GP14 της πλακέτας Raspberry Pi Pico

Πράσινο LED:

- Θετικό πόδι: Σύνδεση στο GP10 της πλακέτας Raspberry Pi Pico μέσω αντίστασης 220 ohm
- Αρνητικό πόδι: Σύνδεση στο GND της πλακέτας Raspberry Pi Pico

Πορτοκαλί LED:

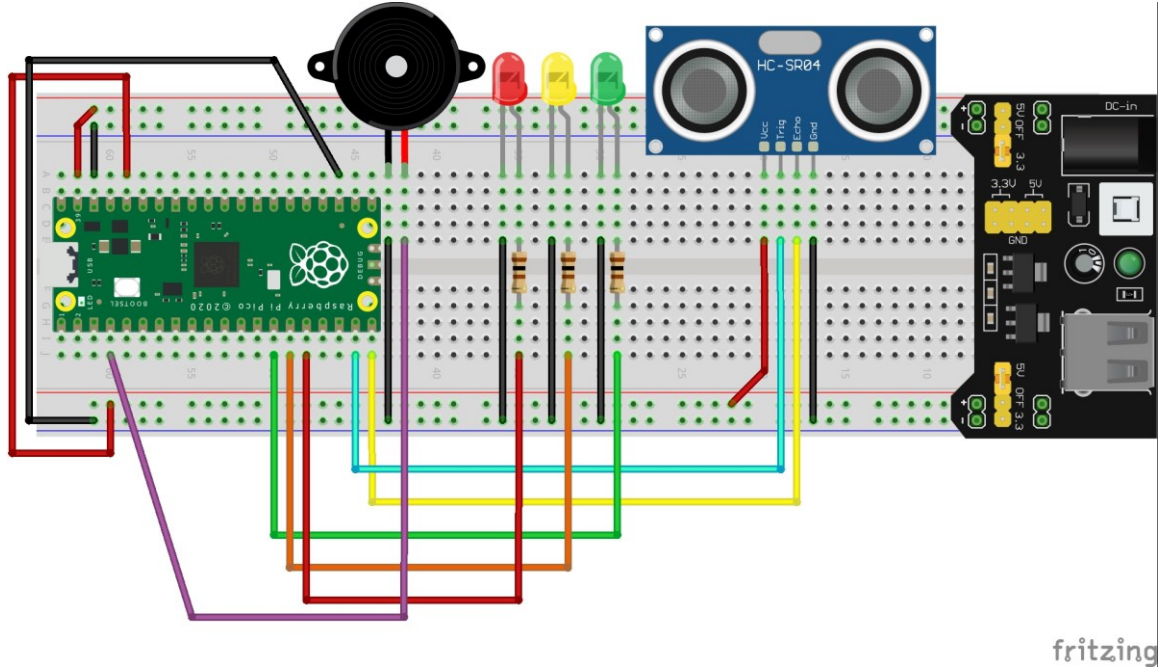
- Θετικό πόδι: Σύνδεση στο GP11 της πλακέτας Raspberry Pi Pico μέσω αντίστασης 220 ohm
- Αρνητικό πόδι: Σύνδεση στο GND της πλακέτας Raspberry Pi Pico

Κόκκινο LED:

- Θετικό πόδι: Σύνδεση στο GP12 της πλακέτας Raspberry Pi Pico μέσω αντίστασης 220 ohm
- Αρνητικό πόδι: Σύνδεση στο GND της πλακέτας Raspberry Pi Pico

Βομβητής:

- Θετικό πόδι: Σύνδεση στο GP2 της πλακέτας Raspberry Pi Pico
- Αρνητικό πόδι: Σύνδεση στο GND της πλακέτας Raspberry Pi Pico



Κώδικας

```
# Import required libraries
from machine import Pin, time_pulse_us
import time

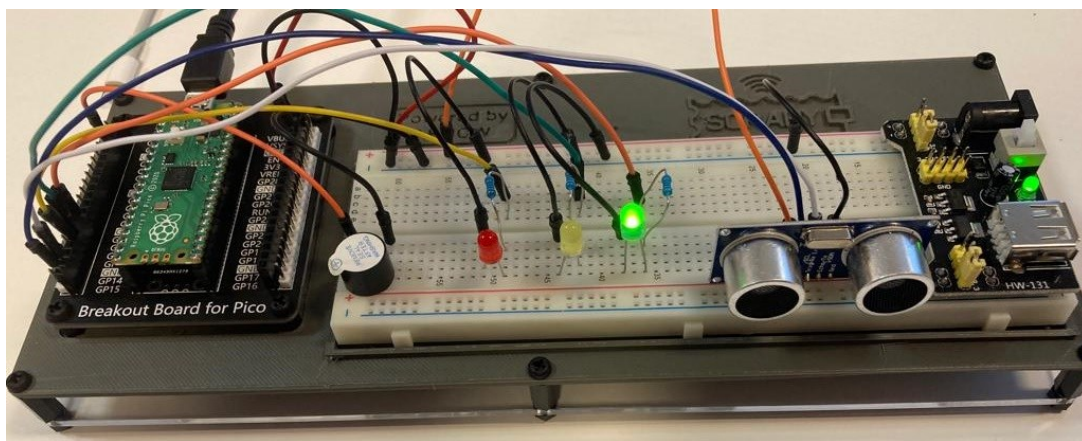
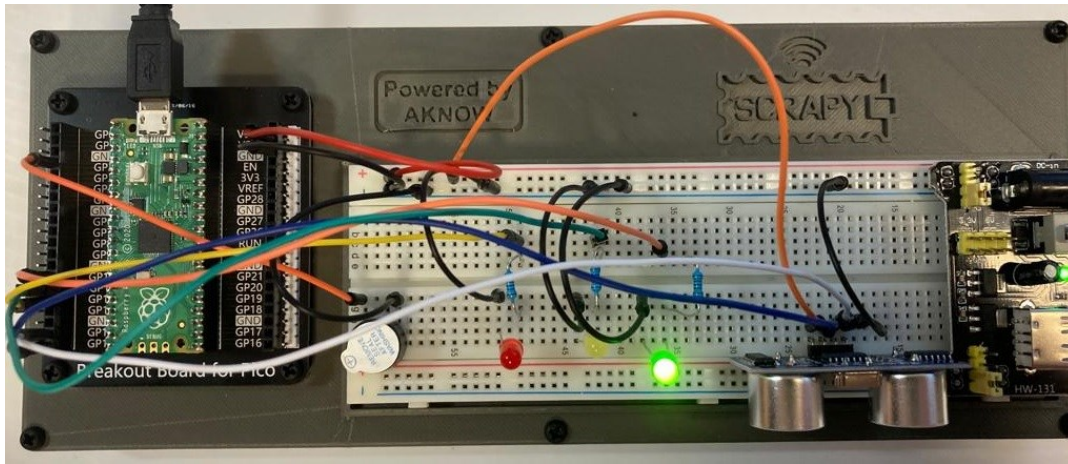
# Define pins for the components
trigger_pin = Pin(15, Pin.OUT)
echo_pin = Pin(14, Pin.IN)
green_led = Pin(10, Pin.OUT)
orange_led = Pin(11, Pin.OUT)
red_led = Pin(12, Pin.OUT)
buzzer = Pin(2, Pin.OUT)

# Define the distance thresholds for each LED
green_threshold = 20 # cm
orange_threshold = 5 # cm

# Define the function to calculate the distance from the HC-SR04 sensor
def get_distance():
    # Send a 10us pulse to trigger the sensor
    trigger_pin.low()
    time.sleep_us(2)
    trigger_pin.high()
    time.sleep_us(10)
    trigger_pin.low()
    # Measure the duration of the echo signal
    duration = time_pulse_us(echo_pin, 1, 10000)
    # Calculate the distance from the duration using the speed of sound (343
    m/s)
    distance = duration / 2 / 1000000 * 343 * 100
    return distance

# Define the main loop to read the distance and control the LEDs and buzzer
while True:
    # Get the distance from the sensor and print it
    distance = get_distance()
    print(f"Distance : {distance} cm")
    # Turn on the green LED if the distance is greater than the threshold
    if distance > green_threshold:
        green_led.on()
        orange_led.off()
        red_led.off()
        buzzer.off()
    # Turn on the orange LED if the distance is between the thresholds
    elif distance > orange_threshold:
        green_led.off()
        orange_led.on()
        red_led.off()
        buzzer.off()
    # Turn on the red LED and buzzer if the distance is less than the threshold
    else:
        green_led.off()
        orange_led.off()
        red_led.on()
        buzzer.on()
        time.sleep(0.5) # Buzz for 0.5 seconds
    # Wait for 0.1 second before the next measurement
    time.sleep(0.1)
```


Παραδείγματα εικόνων



Συμπέρασμα

Συμπερασματικά, αυτό το έργο περιλάμβανε τη χρήση μιας πλακέτας Raspberry Pi Pico, ενός αισθητήρα υπερήχων HC-SR04 και ορισμένων πρόσθετων εξαρτημάτων για τη δημιουργία ενός ραντάρ οπισθοπορείας για αυτοκίνητα. Μάθαμε για τις αρχές λειτουργίας του αισθητήρα HC-SR04, πώς να συνδέσουμε τα εξαρτήματα μεταξύ τους και πώς να προγραμματίσουμε τη συμπεριφορά του συστήματος χρησιμοποιώντας το MicroPython. Ακολουθώντας τα βήματα σε αυτό το σεμινάριο, μπορέσαμε να δημιουργήσουμε ένα σύστημα που μπορεί να ανιχνεύει εμπόδια και να παρέχει οπτική και ακουστική ανατροφοδότηση στον οδηγό.

Όσον αφορά την περαιτέρω εξερεύνηση, υπάρχουν πολλοί τρόποι επέκτασης αυτού του έργου. Μερικές προτάσεις περιλαμβάνουν:

- Προσθήκη περισσότερων LED ή οθόνης για πιο λεπτομερή ανάδραση απόστασης.
- Χρήση αλγορίθμων μηχανικής εκμάθησης για τη βελτίωση της ανίχνευσης και της ακρίβειας εμποδίων.
- Δημιουργία ασύρματης διεπαφής που θα επιτρέπει στο σύστημα να επικοινωνεί με μια φορητή συσκευή ή άλλη εξωτερική συσκευή.
- Ενσωμάτωση πρόσθετων αισθητήρων ή εξαρτημάτων για τη δημιουργία ενός πιο ολοκληρωμένου συστήματος ασφάλειας του οχήματος.

Συνολικά, αυτό το έργο παρέχει μια μεγάλη εισαγωγή στον κόσμο της ηλεκτρονικής και του προγραμματισμού και χρησιμεύει ως αφετηρία για περαιτέρω εξερεύνηση και πειραματισμό.