

Μάθημα: Αισθητήρες-Ενεργοποιητές, Μηχατρονική: **Ανιχνευτής Αντικειμένων-Βοηθός Παρκαρίσματος**

ΒΙΒΛΙΟΓΡΑΦΙΑ:

Το φύλλο έργου στηρίχτηκε σε πληροφορίες

1. από την ηλεκτρονική διεύθυνση: <https://tuixte.wordpress.com/2013/03/05/how-to-ultrasonic-sensor-hc-sr04/> στις 09/02/2016 11:33 π.μ
2. στο βιβλίο «Ανάπτυξη Εφαρμογών με το Arduino» Π.Παπάζογλου και Σ.Π Λιωνής, Εκδόσεις ΤΖΙΟΛΑ 2015 (Κεφάλαιο 4 και Κεφάλαιο 6: κώδικας 6.3)

Στόχοι: κατανόηση της λειτουργίας:

α) του **αισθητήρα απόστασης HC-SR04**

β) τη ψηφιακή οδήγηση βομβητή (buzzer) μέσω τρανζίστορ

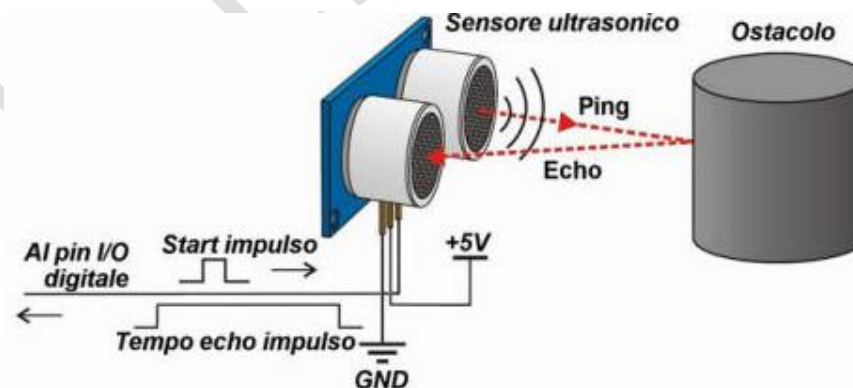
γ) τη κατανόηση λειτουργίας των συναρτήσεων στη γλώσσα Wiring C

Τα Υλικά που θα χρειαστούμε:

1. Πλακέτα Arduino UNO,
2. 1 αισθητήρα HC-SR04 με το ηλεκτρονικό του κύκλωμα ώστε να συμβατός με την πλακέτα Arduino UNO
3. 1 βομβητή (buzzer) 6 Volts
4. 1 αντίσταση 220 Ω
5. 1 δίοδο 1N4004
6. 1 NPN τρανζίστορ BC140 ή 2N2222A ή PN2222A (TO-92)

Τι πρέπει να γνωρίζουμε:

Ο αισθητήρας HC-SR04 χρησιμοποιείται για μετρήσεις αποστάσεων. Όπως φαίνεται στην παρακάτω εικόνα έχει δύο 'μάτια' που στην πραγματικότητα είναι ένας πομπός που στέλνει υπερήχους προς τα γύρω αντικείμενα και ένας δέκτης υπερήχων που δέχεται τους υπερήχους μετά την ανάκλασή τους στα αντικείμενα που βρίσκει μέσα στο 'οπτικό' του πεδίο. Μπορεί να μετρήσει αποστάσεις από 2cm μέχρι 400cm και με ακρίβεια 0,3cm.



<https://tuixte.wordpress.com/2013/03/05/how-to-ultrasonic-sensor-hc-sr04/>

Πως λειτουργεί ο αισθητήρας

Περιλαμβάνει τέσσερις ακροδέκτες από τους οποίους :

- οι δύο εξωτερικοί είναι για την τροφοδοσία του **Vcc** (5 Volts) και **GND** που μπορούν να τροφοδοτηθούν από την πλακέτα Arduino UNO. Από τους δύο κεντρικούς ακροδέκτες
- ο **Trig** είναι ακροδέκτης εισόδου στον αισθητήρα (άρα θα συνδεθεί σε ακροδέκτη εξόδου του Arduino UNO) και χρησιμεύει για την ενεργοποίηση (σκανδαλισμό, triggering). Θα πρέπει να λάβει από το Arduino UNO έναν παλμό που θα έχει διάρκεια του μετώπου HIGH τουλάχιστον 10μs (microseconds).
- ο **Echo** είναι ακροδέκτης εξόδου του αισθητήρα (άρα θα συνδεθεί σε ακροδέκτη εισόδου του Arduino UNO) και στον οποίο όταν φτάνει ο ανακλώμενος από το αντικείμενο υπέρηχος , παράγει ένα παλμό HIGH του οποίου η διάρκεια σε microseconds ταυτίζεται με το χρόνο που μεσολαβεί από τη στιγμή που έφυγε ο υπέρηχος από τον σκανδαλισμό μέχρι να γυρίσει πίσω στον ακροδέκτη Echo. Στην πραγματικότητα μετράει δύο φορές την απόσταση (μια για να πάει και μια για να γυρίσει).



Πως παράγεται ο παλμός triggering

Τα βήματα που ακολουθούμε είναι τα παρακάτω βήματα:

- βάζουμε τον ακροδέκτη **Trig** σε κατάσταση LOW για 2μs
`digitalWrite(trig, LOW);`
`delayMicroseconds(2);`
- βάζουμε τον ακροδέκτη **Trig** σε κατάσταση HIGH για 10μs
`digitalWrite(trig, HIGH);`
`delayMicroseconds(10);`
- βάζουμε τον ακροδέκτη **Trig** σε κατάσταση LOW για 2μs
`digitalWrite(trig, LOW);`
`delayMicroseconds(2);`

Πως διαβάζεται ο ανακλώμενος υπέρηχος στον ακροδέκτη **Echo** και υπολογίζεται η απόσταση

Χρησιμοποιούμε την εσωτερική συνάρτηση **pulseIn(ακροδέκτης, HIGH/LOW)** της γλώσσας Wiring C.βιβλιοθήκη. Η συνάρτηση μετράει το χρόνο ενός μετώπου (HIGH ή LOW) ενός παλμού που έρχεται σε ένα δηλωμένο ακροδέκτη. Με την παρακάτω εντολή μετράμε τη διάρκεια ενός παλμού HIGH που έρχεται στον ακροδέκτη Echo και τον αποδίδουμε στη μεταβλητή duration σε μονάδες microseconds.

```
duration = pulseIn(echo, HIGH);
```

Στη συνέχεια παίρνουμε υπόψη τον τύπο της ευθύγραμμης ομαλής κίνησης $s=u*t$ όπου u είναι η ταχύτητα του ήχου. Είναι γνωστό πως $u=340m/s$ και αν μετατρέψουμε τα μέτρα σε cm ($*100$) και το χρόνο σε $microseconds$ ($*10^6$) τότε η ταχύτητα του ήχου γίνεται:

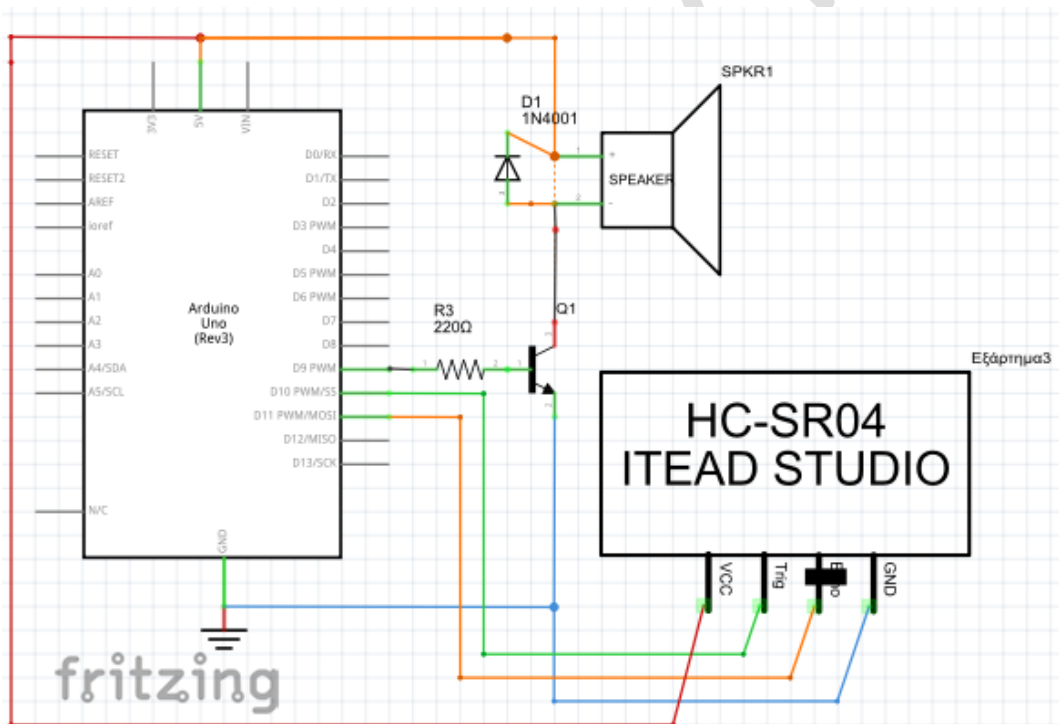
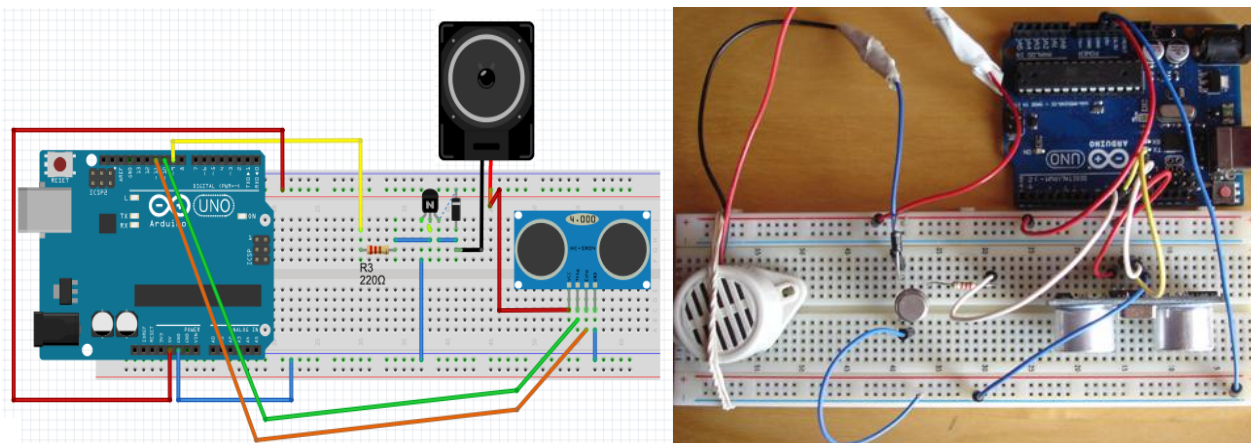
$$u=340*10^2/10^6=340/10000=1/29 \text{ cm}/\mu\text{s}$$

οπότε η υπολογιζόμενη απόσταση είναι $s=t*u$ (χρόνος * ταχύτητα του ήχου):

$$cm=duration / 29 / 2;$$

Η επιπλέον διαίρεση με το 2 προκύπτει γιατί μετράμε δύο φορές την απόσταση (μία για να πάει και μία για να γυρίσει ο υπέρηχος) όπως αναφέραμε και παραπάνω.

Το κύκλωμα της εφαρμογής μας φαίνεται στις παρακάτω εικόνες:



Ένας ενδεικτικός κώδικας σε γλώσσα Wiring είναι ο παρακάτω:

```
int trig = 10;
int echo = 11;
int buzz=9;
long duration, cm;
```

```

void setup()
{
  pinMode(trig,OUTPUT);
  pinMode(echo,INPUT);
  pinMode(buzz, OUTPUT);
  Serial.begin(9600);
}

void loop()
{
  //δημιουργία παλμού triggering
  digitalWrite(trig, LOW);
  delayMicroseconds(2);
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);
  delayMicroseconds(2);

  // μέτρηση της διάρκειας του μετώπου HIGH στον παλμό ECHO
  duration = pulseIn(echo, HIGH);

  //υπολογισμός της απόστασης σε cm και αποστολή της μέτρησης στην οθόνη
  cm=duration / 29 /2;
  Serial.println(cm);

  //παραγωγή ήχου: ενεργοποίηση του βομβητή
  beep(cm);
  delay(50); // είναι απαραίτητο για να προλαβαίνει να γίνεται η εκτίμηση της απόστασης
}

//με μεταλλικό αντικείμενο να κάνω ανακλάσεις και με το χέρι
// ηχητικός εντοπισμός αντικειμένου

void beep(long cm){
  // στη συνάρτηση αυτή παράγω ήχο στον οποίο το LOW μέτωπο είναι αντίστοιχο της
  // απόστασης. Δηλαδή μεγάλη απόσταση δίνει αργό ρυθμό ήχου ενώ μικρή απόσταση δίνει
  //γρήγορο ρυθμό ήχου
  long del=0;
  del=10*cm;
  if (cm < 120) // ΑΝ ΠΛΗΣΙΑΖΕΙ ΕΝΑ ΑΝΤΙΚΕΙΜΕΝΟ:
  {
    digitalWrite(buzz, HIGH); // ON buzzer
    delay(200); // 400 ms ON
    digitalWrite(buzz, LOW); // OFF buzzer
    delay(del); // stay OFF αντίστοιχα με την απόσταση
  }
  else

```

```
{  
digitalWrite(buzz, LOW); // AN TO ANTIKEIMENO EINAI MAKRIA:OXI HXO  
}  
}
```

ΚΑΛΗ ΕΠΙΤΥΧΙΑ

10 ΕΡΓΑΙΑ ΣΥΚΕΩΝ