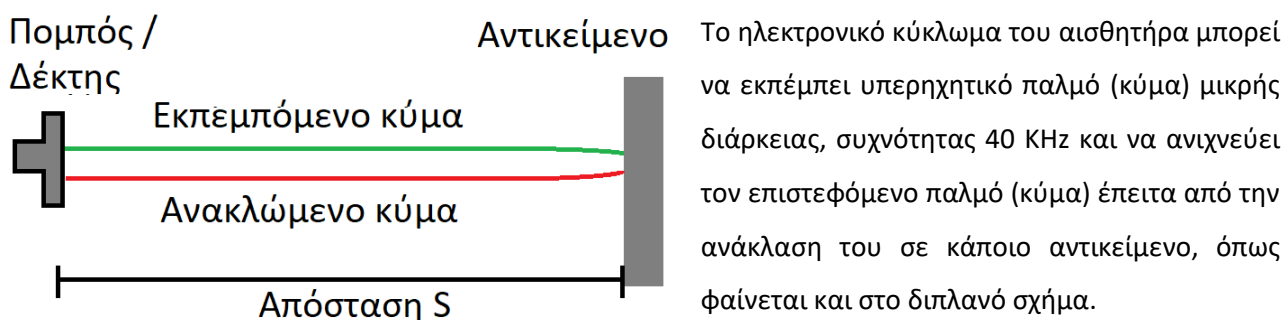


Φύλλο εργασίας 1: Μέτρηση απόστασης με τη βοήθεια κυμάτων

Ενότητα «Ήχος - Κυματικά φαινόμενα του ήχου» Φυσική Γ' Γυμνασίου

Σε ένα υπολογιστικό σύστημα μετρήσεων για διαχείριση και έλεγχο ηλεκτρικής ενέργειας, απαιτείται η συνεχής μέτρηση της απόστασης αντικειμένου (άνθρωπος) από ένα γνωστό σημείο. Για αυτό τον σκοπό χρησιμοποιείται ένας αισθητήρας υπερήχων (σύστημα εκπομπής και λήψης υπερηχητικών κυμάτων).



Αν η ταχύτητα του υπερηχητικού κύματος στον αέρα είναι u , η απόσταση ενός αντικειμένου από το σημείο εκπομπής-λήψης είναι S και ο χρόνος διάδοσης του κύματος t_{tof} (time of flight) για τη διαδρομή (σημείο_εκπομπής) – (αντικείμενο) – (σημείο_λήψης), τότε θεωρώντας ότι η ταχύτητα του υπερήχου παραμένει σταθερή στον αέρα, ισχύει από την ευθύγραμμη ομαλή κίνηση:

$$v = \frac{2S}{t_{tof}}$$

Εαν ο αισθητήρας μπορεί να παρέχει τον χρόνο t_{tof} σε μs , και την απόσταση S την εκφράζουμε σε cm τότε η σχέση που δίνει την ταχύτητα του υπερήχου στο μέσο διάδοσης του αέρα με μονάδες m/s υπολογίζεται από την έκφραση:

$$v = \frac{2 \cdot S / 100}{t_{tof} / 1000000} \Rightarrow v = \frac{2 \cdot S}{t_{tof} / 10000} \Rightarrow v = \frac{200000 \cdot S}{t_{tof}} \text{ σε } m/s$$

Υπολογισμός ταχύτητας υπερήχου στον αέρα

Τοποθετήστε ένα αντικείμενο σε κάποια απόσταση S από τον αισθητήρα. ($10 \text{ cm} < S < 350 \text{ cm}$)

Μετρήστε με ένα μέτρο ή χάρακα την απόσταση σε cm , με όσο μεγαλύτερη ακρίβεια μπορείτε.

Καταγράψτε την μέτρηση:

S = cm

Ανοίξτε την **εφαρμογή Arduino IDE** αντιγράψτε τον παρακάτω κώδικα και εισάγετε τον στην εφαρμογή.

Εντοπίστε την εντολή **const long s =**; και γράψτε την παραπάνω τιμή S , ως τιμή αντιστοίχησης στην σταθερά **long s** χωρίς το κόμμα και τυχόν δεκαδικά ψηφία.

```

const int triggerPin = 8;           // ακροδέκτης 8 για ενεργοποίηση του ηχητικού
                                     // παλμού στον αισθητήρα υπερήχων
const int echoPin = 7;             // ακροδέκτης 7 για ανίχνευση του επιστρεφόμενου
                                     // ηχητικού παλμού

void setup()                       // Συνάρτηση βασικών αρχικοποιήσεων
{
    Serial.begin(9600);            // άνοιγμα σειριακής θύρας για ανάγνωση μετρήσεων
    pinMode(triggerPin, OUTPUT);
    pinMode(echoPin, INPUT);
}

void loop()                        // Κυρίως πρόγραμμα
{
    distance_check();
}

void distance_check()              // Συνάρτηση αισθητήρα απόστασης
{
    const long s = .....;         // Απόσταση αντικειμένου που μετρήσατε σε cm
    long tof = 0;                 // Χρόνος που ταξιδεύει ο ηχητικός παλμός
                                   // (tof = time of flight)
    long u = 0;                   // Ταχύτητα υπερήχου

    // Ενεργοποίηση αισθητήρα υπερήχων
    digitalWrite(triggerPin, LOW); // Παλμός LOW για 2us
    delayMicroseconds(2);
    digitalWrite(triggerPin, HIGH); // Παλμός HIGH για 10us
    delayMicroseconds(10);
    digitalWrite(triggerPin, LOW);  // Παλμός LOW για 2us
    delayMicroseconds(10);

    tof = pulseIn(echoPin, HIGH);   // Χρόνος του ηχητικού παλμού από τον αισθητήρα έως
    // το πιο κοντινό αντικείμενο και επιστροφή
    u = (200000 * s) / tof;         // Ταχύτητα σε m/s

    Serial.print("u = ");
    Serial.println(u);             // Εκτύπωση ταχύτητας στη σειριακή
}

```

Τρέξτε τον κώδικα, ανοίξτε το παράθυρο της σειριακής και καταγράψτε την ταχύτητα του υπερήχου που εμφανίζεται στην οθόνη.

Καταγράψτε την μέτρηση:

u = m/s

Υπολογισμός απόστασης αντικειμένου από τον αισθητήρα

Στην **εφαρμογή Arduino IDE** ανοίξτε νέο αρχείο και αντιγράψτε τον παρακάτω κώδικα.

Εντοπίστε την εντολή **const long u =;** και γράψτε την παραπάνω τιμή u, ως τιμή αντιστοίχησης στην σταθερά **long u** χωρίς το κόμμα και τυχόν δεκαδικά ψηφία.

```

const int triggerPin = 8;           // ακροδέκτης 8 για ενεργοποίηση του ηχητικού παλμού
                                     // στον αισθητήρα υπερήχων
const int echoPin = 7;             // ακροδέκτης 7 για ανίχνευση του επιστρεφόμενου
                                     // ηχητικού παλμού
void setup()                       // Συνάρτηση βασικών αρχικοποιήσεων
{
    Serial.begin(9600);            // άνοιγμα σειριακής θύρας για ανάγνωση μετρήσεων
    pinMode(triggerPin, OUTPUT);
    pinMode(echoPin, INPUT);
}

void loop()                        // Κυρίως πρόγραμμα
{
    distance_check();
}

```

```

void distance_check()                                // Συνάρτηση αισθητήρα απόστασης
{
    long s;                                           // Απόσταση αντικειμένου
    long tof = 0;                                     // Χρόνος που ταξιδεύει ο ηχητικός
                                                    // παλμός (tof = time of flight)
    const long u = .....;                           // ταχύτητα υπερήχου που υπολογίσατε

    // Ενεργοποίηση αισθητήρα υπερήχων
    digitalWrite(trigerPin, LOW);                    // Παλμός LOW για 2us
    delayMicroseconds(2);
    digitalWrite(trigerPin, HIGH);                   // Παλμός HIGH για 10us
    delayMicroseconds(10);
    digitalWrite(trigerPin, LOW);                    // Παλμός LOW για 2us
    delayMicroseconds(10);

    tof = pulseIn(echoPin, HIGH);                    // Χρόνος του ηχητικού παλμού από τον αισθητήρα
                                                    // έως το πιο κοντινό αντικείμενο και επιστροφή

    s = (100 * u * tof) / (2*1000000);               // Απόσταση σε εκατοστά (ταχύτητα σε cm/μs)

    Serial.print("s = ");                            // Εκτύπωση απόστασης στη σειριακή
    Serial.println(s);
}
    
```

Τρέξτε τον κώδικα, ανοίξτε το παράθυρο της σειριακής και καταγράψτε την απόσταση του αντικειμένου που εμφανίζεται στην οθόνη.

Καταγράψτε την απόσταση του αντικειμένου:

S = cm

Μετά το πέρας της δραστηριότητας

Η δραστηριότητα προσεγγίζει ένα μέρος της ανάπτυξης του συστήματος εξοικονόμηση ενέργειας A.li.S. σε μια σχολική αίθουσα.

Τώρα μπορείτε μέσω της **εφαρμογής Arduino IDE** να «ανεβάσετε» ολόκληρο τον κώδικα στον μικροελεγκτή του συστήματος και να δοκιμάσετε τη λειτουργία του.

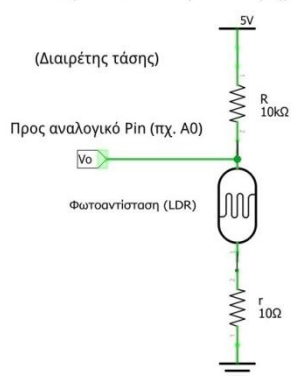
Ο κώδικας βρίσκεται στην ηλεκτρονική διεύθυνση: https://github.com/konsk/ALiS/tree/master/code_files

Φύλλο εργασίας 2: Μέτρηση τάσης με τη βοήθεια αντιστατών**Ενότητα « Εφαρμογές αρχών διατήρησης στη μελέτη απλών ηλεκτρικών κυκλωμάτων – Σύνδεση αντιστατών σε σειρά » Φυσική Γ' Γυμνασίου**

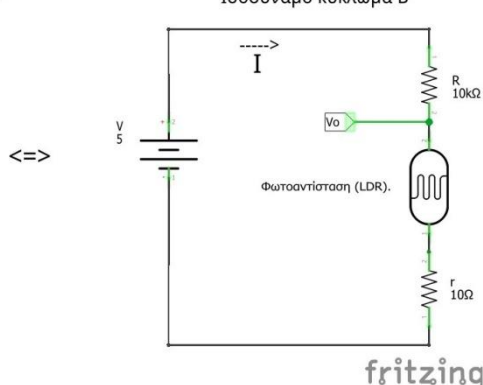
Σε ένα υπολογιστικό σύστημα μετρήσεων για διαχείριση και έλεγχο ηλεκτρικής ενέργειας, απαιτείται η συνεχής μέτρηση της φωτεινότητας σε διάφορες περιοχές. Για τον σκοπό αυτό χρησιμοποιείται ως αισθητήρας φωτός μια φωτοαντίσταση δηλ ένα φωτοστοιχείο που μεταβάλλει την αντίστασή του όταν μεταβάλλεται η ένταση της φωτεινής ακτινοβολίας που προσπίπτει επάνω του.



Ηλεκτρονικό κύκλωμα αισθητήρα φωτός Α



Ισοδύναμο κύκλωμα Β



Το ηλεκτρονικό κύκλωμα του αισθητήρα φωτός δίνεται δίπλα. Τρεις αντιστάτες R , R_{ldr} , r συνδέονται σε σειρά.

Η φωτοαντίσταση R_{ldr} μεταβάλλεται από λίγα Ω (πχ 600Ω) σε συνθήκες έντονης φωτεινότητας, έως και πάνω από $40K\Omega$ σε απειροελάχιστη φωτεινότητα και σε πολύ μεγαλύτερες τιμές σε πλήρες σκοτάδι.

Η εκτίμηση της φωτεινότητας γίνεται με μέτρηση της τάσης V_0 στα άκρα της συνδεσμολογίας R_{ldr} και r .

Στόχος:

Να υπολογιστεί το εύρος τάσεων V_0 (που μπορεί να μετρηθεί από ένα υπολογιστικό σύστημα με τάση λειτουργίας 5 V, για $1K\Omega \leq R_{ldr} \leq 40K\Omega$, και να αντιστοιχιστούν οι τιμές σε ένα εύρος αριθμών από 0 έως 255 (πίνακας 1), τις οποίες μπορεί να επεξεργαστεί το υπολογιστικό σύστημα. Κατά τη διαδικασία της μέτρησης το υπολογιστικό σύστημα, αντιστοιχίζει 4,9mV σε κάθε αριθμό από 0-1023. Παράδειγμα:

Πίνακας 1				
Μετρούμενες τιμές τάσεων	Ελάχιστη τιμή V_{min} (mVolt)		Μέγιστη τιμή V_{max} (mVolt)	
	$1,5 \cdot 1000 = 1500$		$4 \cdot 1000 = 4000$	
Τιμή που αντιστοιχίζει ο υπολογιστής στην κλίμακα 0-1023 (4,9mV/αριθμό)	Ελάχιστη		Μέγιστη	
	$1500 / 4,9 = 306$		$4000 / 4,9 = 816$	
Ακραίες τιμές αριθμών (κλίμακα 0-1023)	Αριθμός κοντά στον παραπάνω αλλά προς τα κάτω		Αριθμός κοντά στον παραπάνω αλλά προς τα άνω	
	Amin	280	Amax	850
Ακραίες τιμές (κλίμακα 0 – 255)	Bmin	0	Bmax	255

Καταγράψτε από το ηλεκτρονικό κύκλωμα τις τιμές των αντιστάσεων για τους αντιστάτες R και r, αντίστοιχα, καθώς και την ελάχιστη και μέγιστη τιμή της R_{ldr} όπως φαίνεται από την παραπάνω ανισότητα:

Αντιστάτης	R	r	$R_{ldr,min}$	$R_{ldr,max}$
Αντίσταση σε (Ω)				

α) Υπολογίστε την ελάχιστη ισοδύναμη αντίσταση του κυκλώματος $R_{min} = R + R_{ldr,min} + r$

$$R_{min} = \dots\dots\dots$$

β) Υπολογίστε την μέγιστη ισοδύναμη αντίσταση του κυκλώματος $R_{max} = R + R_{ldr,max} + r$

$$R_{max} = \dots\dots\dots$$

γ) Με βάση R_{min} υπολογίστε την μέγιστη ένταση του ρεύματος που διαρρέει όλο το κύκλωμα I_{max}

$$I_{max} = \frac{V}{R_{min}} = \frac{5}{\dots\dots\dots} = \dots\dots\dots A$$

δ) Με βάση R_{max} υπολογίστε την ελάχιστη ένταση του ρεύματος που διαρρέει όλο το κύκλωμα I_{min}

$$I_{min} = \frac{V}{R_{max}} = \frac{5}{\dots\dots\dots} = \dots\dots\dots A$$

ε) Με βάση την I_{max} υπολογίστε την ελάχιστη τάση στα άκρα της $R_{ldr} + r$

$$V_{min} = I_{max} \cdot (R_{ldr,min} + r) = \dots\dots\dots V$$

ζ) Με βάση την I_{min} υπολογίστε την μέγιστη τάση στα άκρα της $R_{ldr} + r$

$$V_{max} = I_{min} \cdot (R_{ldr,max} + r) = \dots\dots\dots V$$

Εισάγετε τις τιμές ελάχιστης και μέγιστης τάσης (V_{min} , V_{max}) που υπολογίσατε στον πίνακα 1, σε mV.

Ορίστε την ελάχιστη και μέγιστη τιμή στις θέσεις μνήμης, που θα αντιστοιχούν στην ελάχιστη και μέγιστη μετρούμενη τάση αντίστοιχα.

Πίνακας 1				
Μετρούμενες τιμές τάσεων	Ελάχιστη τιμή V_{min} (mVolt)		Μέγιστη τιμή V_{max} (mVolt)	
Τιμή που αντιστοιχίζει ο υπολογιστής στην κλίμακα 0-1023 (4,9mV/αριθμό)	Ελάχιστη		Μέγιστη	
Ακραίες τιμές (κλίμακα 0-1023)	Αριθμός κοντά στον παραπάνω αλλά προς τα κάτω		Αριθμός κοντά στον παραπάνω αλλά προς τα άνω	
	Amin		Amax	
Ακραίες τιμές (κλίμακα 0 – 255)	Bmin		Bmax	

Στην **εφαρμογή Arduino IDE** ανοίξτε νέο αρχείο και αντιγράψτε τον παρακάτω κώδικα. Εντοπίστε την εντολή **avg = map(avg, Amin, Amax, Bmin, Bmax);** και γράψτε τις τιμές Amin, Amax, Bmin, Bmax του Πίνακα 1 στις αντίστοιχες θέσεις της εντολής.

```
const int ldrPin11 = A0; // ακροδέκτης A0 για ανάγνωση αναλογικής
                          // τάσης από τον αισθητήρα φωτός (ldr) 11
void setup()             // Συνάρτηση βασικών αρχικοποιήσεων
{
  Serial.begin(9600);
  pinMode(ldrPin11, INPUT);
}
void loop()              // Κυρίως πρόγραμμα
{
  measure_ldr(ldrPin);
}
/* Συνάρτηση μετρήσεων με μέσο όρο τιμών */
int measure_ldr(int ldrPin) // ldrPin, ο ακροδέκτης σύνδεσης του
                          // μικροελεγκτή με την φωτοαντίσταση
{
  const int num_measure = 10; // Αριθμός τιμών για υπολογισμό μέσου όρου
  int lev = 0;                // Τρέχουσα μέτρηση
  int avg = 0;                // Μέσος όρος (ακέρια μορφή)
  float average = 0.0;        // Μέσος όρος (δεκαδική μορφή)

  for(int i=1; i<=num_measure; i++)
  {
    lev = analogRead(ldrPin); // Ανάγνωση τιμής
                                // αισθητήρα
    average = float(lev)/float(i) + average*float(i-1)/float(i); // Υπολογισμός μέσου
                                                                    // όρου από προηγούμενη
                                                                    // τιμή του και από
                                                                    // τρέχουσα μέτρηση)
  }
  avg = int(average); // Μετατροπή μέσου όρου σε ακέραιο
  avg = map(avg, Amin, Amax, Bmin, Bmax); // Προσαρμογή εύρους τιμών από
                                          // Amin-Amax σε Bmin-Bmax
  Serial.print("level_11 = ");
  Serial.println(level_11); // Εκτύπωση απόστασης στη σειριακή

  return avg;
}
```

Τρέξτε τον κώδικα, ανοίξτε το παράθυρο της σειριακής και καταγράψτε την τιμή που αντιστοιχεί στην φωτεινότητα του χώρου και εμφανίζεται στην οθόνη (διάστημα τιμών 0-255).

Καταγράψτε την τιμή που αντιστοιχεί στην φωτεινότητα του χώρου:

level_11 =

Μετά το πέρας της δραστηριότητας

Η δραστηριότητα προσεγγίζει ένα μέρος της ανάπτυξης του συστήματος εξοικονόμηση ενέργειας A.li.S. σε μια σχολική αίθουσα.

Τώρα μπορείτε μέσω της **εφαρμογής Arduino IDE** να «ανεβάσετε» ολόκληρο τον κώδικα στον μικροελεγκτή του συστήματος και να δοκιμάσετε τη λειτουργία του.

Ο κώδικας βρίσκεται στην ηλεκτρονική διεύθυνση: https://github.com/konsk/ALiS/tree/master/code_files