

Έξυπνη πόλη –έξυπνος φωτισμός

Μέχρι το τέλος αυτού του αιώνα περισσότερο από το 80% του παγκόσμιου πληθυσμού θα έχει συγκεντρωθεί σε μεγάλα αστικά κέντρα. Καθώς λοιπόν όλο και περισσότεροι άνθρωποι μετακινούνται στις πόλεις, η βιωσιμότητά τους τίθεται υπό αμφισβήτηση.

Η πρόβλεψη αυτή διαμορφώνει από μόνη της μια ανάγκη για έξυπνο σχεδιασμό και έξυπνους τρόπους οργάνωσης της καθημερινότητας στον αστικό χώρο με σκοπό τη βελτίωση της ποιότητας ζωής. Σημαντικό ρόλο στο πρόβλημα αυτό θα παίξει η τεχνολογία και τα επιτεύγματά της. Στην έρευνά μας για την βιωσιμότητα των πόλεων, συναντήσαμε τον όρο «Έξυπνη πόλη». Ο όρος αυτός αφορά το «τεχνολογικό τμήμα» μιας πόλης όπου προγραμματιστές και μεγάλες επιχειρήσεις, συνεργαζόμενοι με δήμους χρησιμοποιούν το Διαδίκτυο των Πραγμάτων (Νέες Τεχνολογίες – αισθητήρες και το διαδίκτυο) για να βελτιώσουν τους πόρους τους, τις υποδομές, και γενικά τη ζωή των πολιτών τους.

Μια τέτοια δράση θα είναι και ο έξυπνος φωτισμός των δρόμων με το οποίο και ασχοληθήκαμε.

Με τον φωτισμό των οδών για πολλές ώρες με την ίδια ένταση, ανεξαρτήτως του φωτός από το περιβάλλον, από τη βιβλιογραφική μας έρευνα βρήκαμε ότι έχουμε:

Σπατάλη οικονομικών πόρων, Ατμοσφαιρική ρύπανση και Φωτορρύπανση

Οι χώρες ξοδεύουν πάνω από 10 δισ. ευρώ το χρόνο για τον φωτισμό των δρόμων, ποσό που αντιστοιχεί στο 40% των δαπανών της κάθε χώρας για την ενέργεια. Επιπλέον, η ενέργεια αυτή μεταφράζεται σε 40 εκατ. τόνους εκπομπών διοξειδίου του άνθρακα το χρόνο.

Στην εργασία μας:

Προσπαθήσαμε να προσομοιώσουμε ένα Οδικό Σύστημα στο οποίο:

Τα φώτα του δρόμου θα ανάβουν όταν περνάει όχημα μπροστά από μία κολώνα στην οποία έχει τοποθετηθεί ένας αισθητήρας απόστασης.

Το καλύτερο είναι ο φωτισμός να υπάρχει συνεχώς χαμηλός και να ενισχύεται με το πέρασμα των αυτοκινήτων ή των ποδηλάτων ή των ανθρώπων.

Μάλιστα θα μπορούσε να αναπτυχθούν εφαρμογές (ήδη υπάρχουν) που να ρυθμίζουν την ένταση του φωτισμού ανάλογα με την ώρα και τον φυσικό φωτισμό

Επίσης θα μπορούσε να «ειδοποιείται» η επόμενη κολώνα από την προηγούμενη με αισθητήρες ώστε να ανάβει τη λάμπα της λίγο πριν περάσει το αυτοκίνητο ή ο πεζός.

Θα μπορούσε ακόμα η κάθε λάμπα Led να τροφοδοτείται από ένα ατομικό σύστημα συλλογής ηλιακής ενέργειας, πράγμα που θα ελάττωνε συντριπτικά το κόστος, και δεν θα απαιτούνταν καλωδιώσεις και συντήρηση.

Έτσι θα γίνεται εξοικονόμηση ενέργειας, φυσικών πόρων, ελάττωση παγίων εξόδων αλλά και μείωση της ατμοσφαιρικής ρύπανσης.

Είναι μια τεχνολογία που χρησιμοποιούν ήδη Γαλλία, η Νορβηγία, η Κίνα και η Αμερική.

Η ΔΕΗ επίσης, έχει ξεκινήσει συνεργασία με δήμους για τον εκσυγχρονισμό του οδοφωτισμού, το οποίο εκτός από την αντικατάσταση των λαμπτήρων με νέους, χαμηλής κατανάλωσης που από μόνη της οδηγεί σε σημαντική (πάνω από 50 %) μείωση της κατανάλωσης ενέργειας και αντίστοιχη εξοικονόμηση πόρων, μπορεί να επεκταθεί σε νέες καινοτόμες χρήσεις.

Οι κολώνες του οδοφωτισμού της ΔΕΗ μπορεί σύντομα να λένε στους οδηγούς που υπάρχουν κενές θέσεις στάθμευσης, να μετρούν την ηχορύπανση και την ατμοσφαιρική ρύπανση, ενώ οι λάμπες θα σβήνουν όταν δεν υπάρχει κίνηση και θα ειδοποιούν αυτόματα τους τεχνικούς όταν έχουν βλάβη. Αισθητήρες κίνησης μπορούν ακόμη να ειδοποιούν το κέντρο ελέγχου όταν ο δρόμος είναι κενός, ώστε να σβήνουν οι λάμπες με τηλεχειρισμό και να επιτυγχάνεται επιπλέον εξοικονόμηση, ενώ οι βλάβες θα δηλώνονται επίσης αυτόματα στο κέντρο.

Όσο αφορά την φωτορύπανση, δηλαδή το αποτέλεσμα των ανθρωπογενών πηγών φωτός μέσα στη νύχτα, απόρροια του τρόπου ζωής μας, απειλεί να αφανίσει τα έντομα με δραματικές συνέπειες για τα οικοσυστήματα και τον άνθρωπο, γιατί διαταράσσεται ο κύκλος δραστηριότητας των εντόμων και αυτό έχει σοβαρή επίδραση στην υγεία τους. Το φως που πειράζει λιγότερο τα έντομα είναι το πορτοκαλί, γι' αυτό και στην κατασκευή μας χρησιμοποιήσαμε πορτοκαλί led.

Οι κατασκευές μας.

Κατασκευάσαμε ένα δρόμο στις άκρες του οποίου υπάρχουν 6 κολώνες φωτισμού. Σε κάθε κολώνα υπάρχει ένας λαμπτήρας led και ένας αισθητήρας απόστασης ultrasonic. Οι αισθητήρες ενεργοποιούν τον φωτισμό των led με προγραμματισμό δύο επεξεργαστών Arduino UNO, όταν υπάρξει κίνηση σε απόσταση 20 cm από τις κολώνες.

Ένα ρομποτικό όχημα EV3, που κινείται συνεχώς σε μαύρη κλειστή γραμμή, προσομοιώνει το αυτοκίνητο που περνά κάτω από τους λαμπτήρες, (με πρόγραμμα line follower proportional).

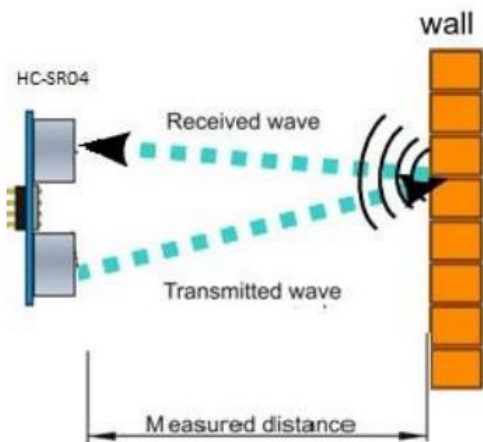
Ενδιάμεσα υπάρχουν κολώνες με λαμπτήρες που τροφοδοούνται από ηλιακά πάνελ. Η ενέργεια του ήλιου αποθηκεύεται και ανάβουν οι λαμπτήρες κάποια ώρα το βράδυ. Αυτές δίνουν ασθενή φωτισμό συνεχώς, ενώ γίνεται έντονος με το άναμμα των λαμπτήρων που είναι συνδεδεμένοι με τους αισθητήρες, όταν υπάρχει κίνηση. Αλλιώς υπάρχει μόνο των λαμπτήρων με τα φωτοβολταϊκά, τα οποία δεν χρειάζονται συντήρηση, καλώδια και δεν έχουν κόστος ενεργειακό. Στην κατασκευή μας το έχουμε κατασκευάσει με προσομοίωση και θα το διευρενήσουμε σε επόμενο στάδιο.



Στον σύνδεσμο [εδώ](#) μπορείτε να δείτε το παράδειγμά μας

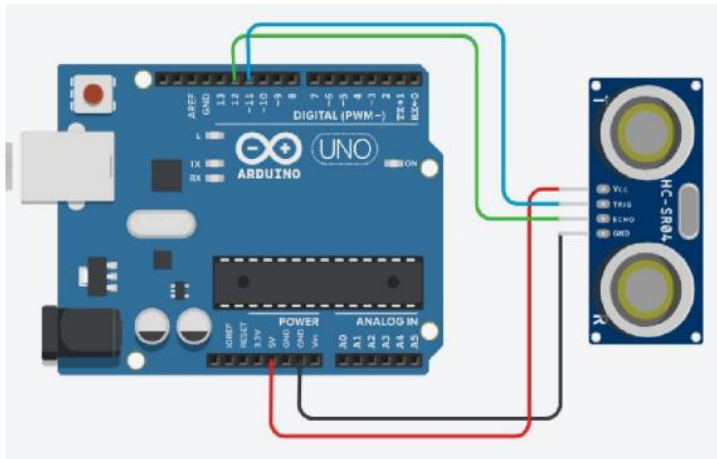
Λίγα λόγια για τον αισθητήρα υπερήχων ultrasonic

Χρησιμοποιήσαμε 6 αισθητήρες υπερήχων HC-SR04 . Ο ultrasonic HC-SR04 έχει ένα πομπό και έναν δέκτη . Ο πομπός εκπέμπει ένα σήμα υψηλής συχνότητας Το κύμα με ταχύτητα $v = 344\text{m/sec}$, ανακλάται πάνω στο εμπόδιο και επιστρέφει. Ο δέκτης του λαμβάνει το ανακλώμενο σήμα μετά από χρόνο t . Με την εξίσωση κίνησης ($s=v*t$) υπολογίζουμε την απόσταση ως εξής. $S=v*t/2$ Απόσταση = ταχύτητα ήχου * (απαιτούμενος χρόνος από την εκπομπή μέχρι τη λήψη)/2). Διαιρούμε με 2 γιατί η απόσταση s διανύεται 2 φορές. Ο αισθητήρας έχει περιοχή μέτρησης από 2 cm έως 400 cm , γωνία μέτρησης 15° και λειτουργεί στα 40 KHz.

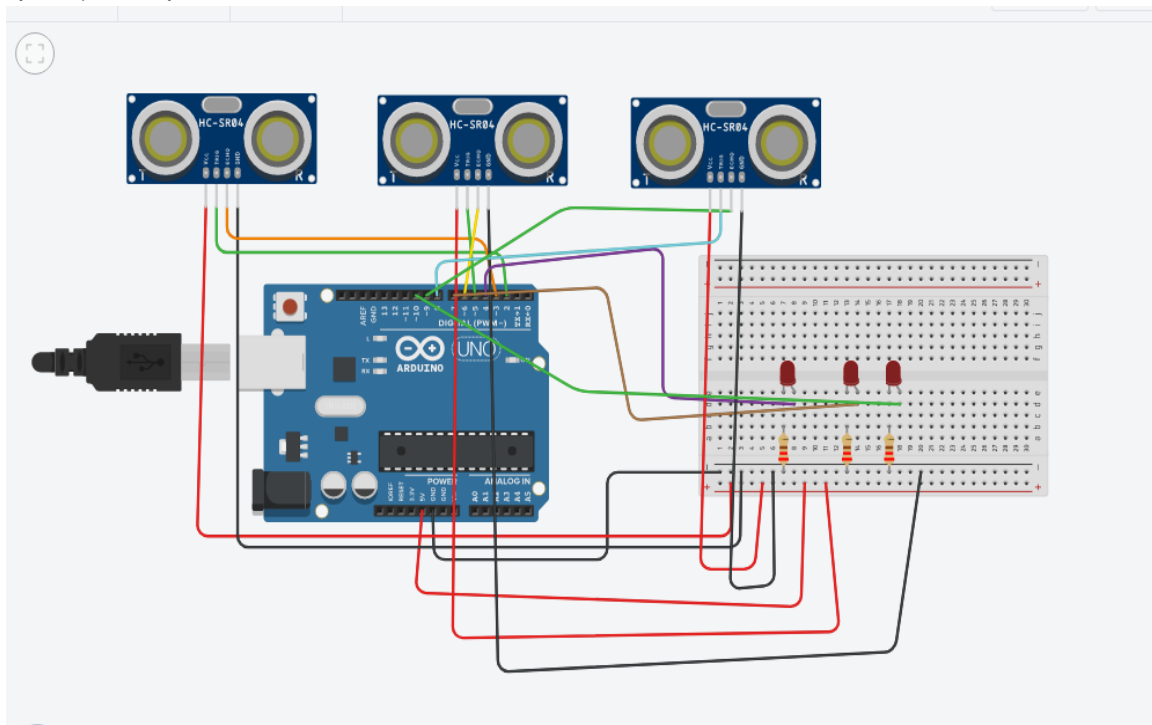


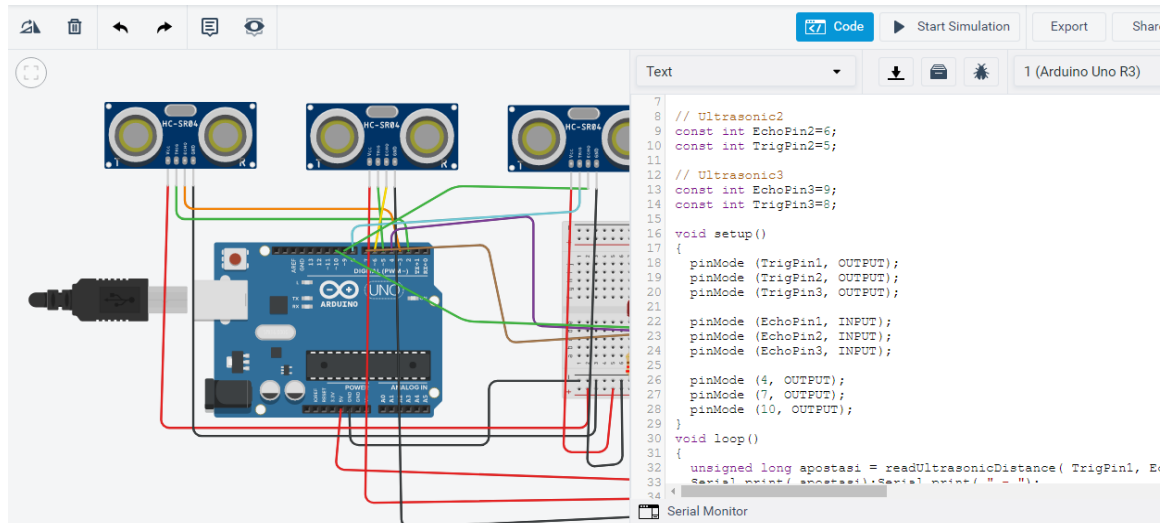
- Sensor Pin 4 (Vcc) -> Arduino +5V
- Sensor Pin 3 (Trig) εκπομπή -> Arduino Pin εξόδου
- Sensor Pin 2 (Echo) λήψη -> Arduino Pin εισόδου
- Sensor Pin1 (Gnd) -> Arduino Ground





Στην κατασκευή μας δημιουργήσαμε για πρακτικούς δύο κυκλώματα με ένα επεξεργαστή Arduino-uno, και τρεις αισθητήρες ultrasonic. Πρώτα σχεδιάσαμε το κύκλωμα στο περιβάλλον Tinkercard και δοκιμάσαμε εκεί την προσομοίωση.





Το πρόγραμμα για το Arduino για τους τρεις αισθητήρες

```
// Ultrasonic1
int EchoPin1=3;
int TrigPin1=2;
```

```
// Ultrasonic2
int EchoPin2=6;
int TrigPin2=5;
```

```
// Ultrasonic3
int EchoPin3=9;
int TrigPin3=8;
```

```
void setup()
{
  pinMode (TrigPin1, OUTPUT);
  pinMode (TrigPin2, OUTPUT);
  pinMode (TrigPin3, OUTPUT);

  pinMode (EchoPin1, INPUT);
  pinMode (EchoPin2, INPUT);
  pinMode (EchoPin3, INPUT);

  pinMode (4, OUTPUT);
  pinMode (7, OUTPUT);
  pinMode (10, OUTPUT);
}
void loop()
{
```

```
float apostasi = readUltrasonicDistance( TrigPin1, EchoPin1) / 58.14;
Serial.print( apostasi);Serial.print( " - ");
if( apostasi > 0) {
  if( apostasi < 20) {
    digitalWrite (4, HIGH);
  } else {
    digitalWrite (4, LOW);
  }
}
delay(100);
```

```
apostasi = readUltrasonicDistance( TrigPin2, EchoPin2) / 58.14;
Serial.print( apostasi);Serial.print( " - ");
if( apostasi > 0) {
  if( apostasi < 20) {
    digitalWrite (7, HIGH);
  } else {
    digitalWrite (7, LOW);
  }
}
delay(100);
```

```
apostasi = readUltrasonicDistance( TrigPin3, EchoPin3) / 58.14;
Serial.print( apostasi);Serial.println( " ");
if( apostasi > 0) {
  if( apostasi < 20) {
    digitalWrite (10, HIGH);
  } else {
    digitalWrite (10, LOW);
  }
}
delay(100);
```

```
/*
digitalWrite (TrigPin1,LOW);
delayMicroseconds(2);
digitalWrite (TrigPin1, HIGH);
delayMicroseconds(10);
digitalWrite (TrigPin1, LOW);
int metrisi1 = pulseIn(EchoPin1, HIGH);
float apostasi1 = metrisi1/58.14;
if (apostasi1 < 20) {
  digitalWrite (4, HIGH);
```

```

digitalWrite (7, LOW);
digitalWrite (10, LOW);
delay(5000);
}

digitalWrite(TrigPin2,LOW);
delayMicroseconds(2);
digitalWrite (TrigPin2, HIGH);
delayMicroseconds(10);
digitalWrite (TrigPin2, LOW);
int metrisi2 = pulseIn(EchoPin2, HIGH);
float apostasi2 = metrisi2/58.14;
if (apostasi2 < 20) {
    digitalWrite (7, HIGH);
    digitalWrite(4, LOW);
    digitalWrite(10, LOW);
    delay(5000);
}
digitalWrite (4, LOW);
digitalWrite (7, LOW);
digitalWrite (TrigPin3, LOW);
delayMicroseconds(2);
digitalWrite (TrigPin3, HIGH);
delayMicroseconds(10);
digitalWrite (TrigPin3, LOW);
int metrisi3 = pulseIn(EchoPin3, HIGH);
float apostasi3 = metrisi3 / 58.14 ;
if (apostasi3 < 20){
    digitalWrite (10, HIGH);
    digitalWrite (4, LOW);
    digitalWrite(7, LOW);
    delay(5000);
}
digitalWrite(TrigPin3, LOW);
delayMicroseconds(2); /*
//delay(100);
}

long readUltrasonicDistance(int triggerPin, int echoPin)
{
    pinMode(triggerPin, OUTPUT); // Clear the trigger
    digitalWrite(triggerPin, LOW);
    delayMicroseconds(2);

```

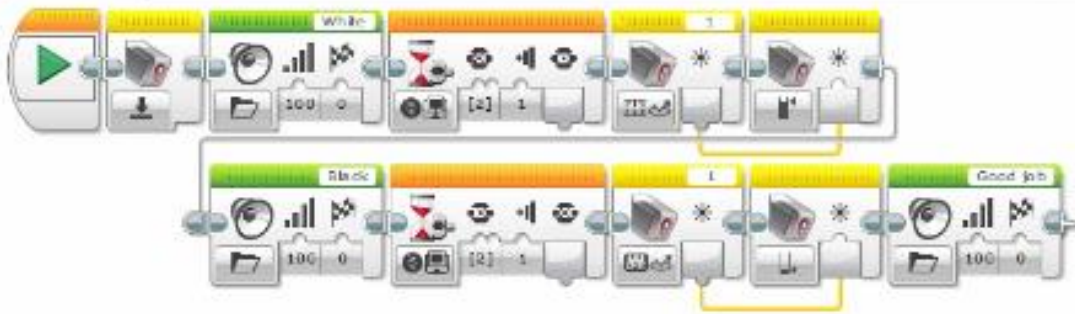
```

// Sets the trigger pin to HIGH state for 10 microseconds
digitalWrite(triggerPin, HIGH);
delayMicroseconds(10);
digitalWrite(triggerPin, LOW);
pinMode(echoPin, INPUT);
// Reads the echo pin, and returns the sound wave travel time in microseconds
return pulseIn(echoPin, HIGH);
}

```

Το ρομποτικό όχημα που κινείται σε μαύρη γραμμή εφαρμόζει ένα πρόγραμμα ακολουθίας γραμμής **Line follower**

Πρώτα κάνουμε την βαθμονόμηση, δηλαδή ρυθμίζουμε τη μέγιστη και την ελάχιστη τιμή του φωτός που έχουμε στον συγκεκριμένο χώρο που βρίσκεται το ρομπότ μας



Το πρόγραμμα που «τρέχει» στο ρομπότ- αυτοκίνητο : Line follower proportional

