

Λειτουργία έξυπνων φαναριών:

Ένα σημαντικό ζήτημα για τα σύγχρονα αστικά κέντρα είναι αυτός της σωστής διαχείρισης του κυκλοφοριακού φόρτου, ο οποίος απαιτεί δυναμική προσέγγιση. Είναι συνηθισμένο φαινόμενο για παράδειγμα σε μία διασταύρωση να δημιουργείται ουρά αυτοκινήτων μπροστά σε ένα φανάρι, όταν η κατεύθυνση προς την οποία το φανάρι είναι πράσινο έχει μηδενικό φόρτο προκαλώντας άσκοπη καθυστέρηση στην κυκλοφορία των οχημάτων. Οι συνηθισμένοι φωτεινοί σηματοδότες λειτουργούν με χρονοδιακόπτη, έχοντας σταθερά ρυθμισμένο πόσο χρόνο θα είναι αναμμένη η κάθε ένδειξη (πράσινο, πορτοκαλί, κόκκινο). Σημαντική βελτίωση θα μπορούσε να υπάρξει αν αντικαταστήσουμε τους σηματοδότες που λειτουργούν με χρόνο, από έξυπνους σηματοδότες, οι οποίοι θα καταλαβαίνουν αν υπάρχουν αυτοκίνητα που περιμένουν στην ουρά ή αν ένας δρόμος είναι “άδειος” αυξάνοντας δυναμικά τον χρόνο της Πράσινης ένδειξης στους δρόμους που έχουν περισσότερη κίνηση και μειώνοντας τον χρόνο στους δρόμους που είναι άδειοι.

Παρακάτω δίνεται ένα παράδειγμα λειτουργίας 2 φαναριών με τον κλασικό τρόπο προγραμματισμένο με το mBLOCK:



Στην συγκεκριμένη υλοποίηση όπως φαίνεται και από την φωτογραφία που ακολουθεί, έχουν συνδεθεί 2 φωτεινοί σηματοδότες:

| | |
|--------------------|--------------------|
| Φανάρι 1-Κόκκινο | Υποδοχή D2 Arduino |
| Φανάρι1 -Πορτοκαλί | Υποδοχή D3 Arduino |
| Φανάρι 1 Πράσινο | Υποδοχή D4 Arduino |
| Φανάρι 2-Κόκκινο | Υποδοχή D5 Arduino |
| Φανάρι2 -Πορτοκαλί | Υποδοχή D6 Arduino |
| Φανάρι 2 Πράσινο | Υποδοχή D7 Arduino |

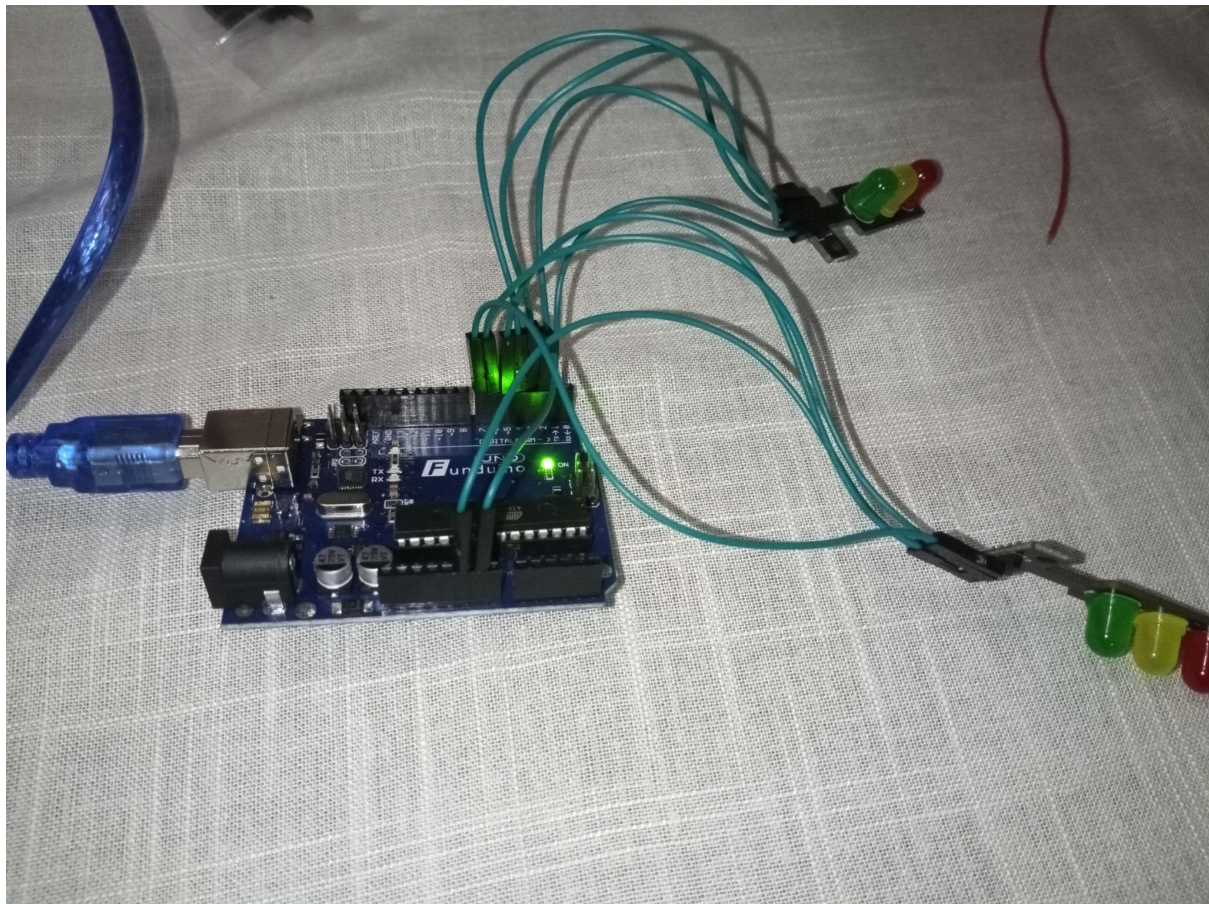
Έχουν γραφτεί δύο διαφορετικά κομμάτια εντολών, ένα για κάθε φανάρι. Προσέξτε ότι το συνολικό άθροισμα του χρόνου στα δυο φανάρια είναι το ίδιο:

Φανάρι 1: $8+8+3 = 19$

Φανάρι 2: $6+2+11 = 19$.

Επίσης προσέξτε ότι πρέπει να αποφεύγουμε οπωσδήποτε:

1. Δεν επιτρέπεται να είναι ταυτόχρονα και τα δύο φανάρια πράσινα.
2. Όταν ένα φανάρι παύει να είναι πράσινο (γίνεται πορτοκαλί) ΔΕΝ πρέπει το άλλο να γίνει κατευθείαν πράσινο, αλλά πρέπει να παραμένει κόκκινο όσο το άλλο φανάρι είναι πορτοκαλί.



Βελτίωση της κυκλοφορίας με τη βοήθεια ενός έξυπνου φαναριού

Με τη βοήθεια ενός αισθητήρα υπερήχων HC-SR04 μπορούμε να κάνουμε το φανάρι μας “έξυπνο” και να δίνει προτεραιότητα σε κάποιον δρόμο όταν αυτός έχει κυκλοφοριακή συμφόρηση.

Αισθητήρας HC-SR04

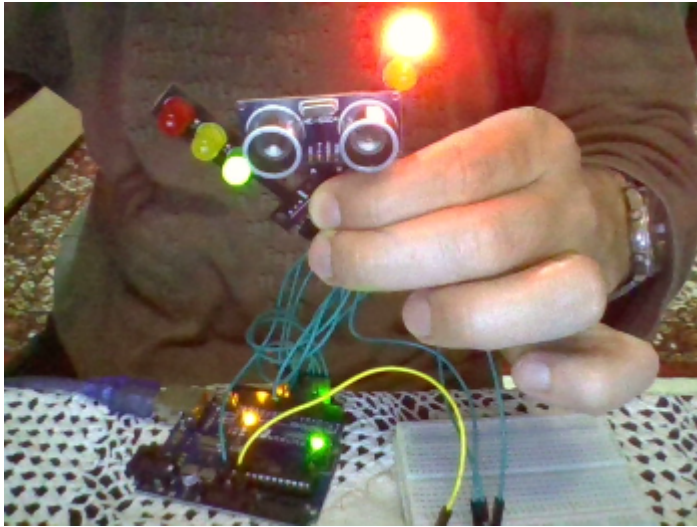
Ο αισθητήρας HC-SR04 είναι ένας αισθητήρας υπερήχων, ο οποίος χρησιμοποιείται για την μέτρηση απόστασης από το επόμενο αντικείμενο που βρίσκεται στην ευθεία μπροστά από τα “μάτια” του. Ο αισθητήρας μας έχει την ικανότητα να αντιλαμβάνεται ότι υπάρχει μπροστά του κάποιο “εμπόδιο και να μετράει τις αποστάσεις απο΄2 έως 400 εκατοστά. Οι αισθητήρες αυτοί στην πραγματική ζωή χρησιμοποιούνται στα αυτοκίνητα και τους ξέρουμε ως “αισθητήρες παρκαρίσματος”. Αντίθετα στα έξυπνα φανάρια χρησιμοποιούνται περισσότερο πολύπλοκοι αισθητήρες, όμως για την δική μας κατασκευή θα χρησιμοποιήσουμε έναν τέτοιο αισθητήρα σε έναν από τους δύο δρόμους.



Ο αισθητήρας θα τοποθετηθεί δίπλα από τον φωτεινό σηματοδότη και θα εντοπίζει την απόσταση από το κοντινότερο αντικείμενο που υποτίθεται ότι είναι τα αυτοκίνητα που κινούνται στο δρόμο. Αν υπάρχουν αυτοκίνητα στον κεντρικό δρόμο σε κοντινή απόσταση, τότε τα φανάρια θα παραμένει πράσινο για πάντα, διαφορετικά θα ενεργοποιείται το σύστημα με τον χρόνο για τη λειτουργία του φαναριού. Φυσικά το ιδανικό είναι να προσθέσουμε και 2ο αισθητήρα που θα βρίσκεται στο 2ο φανάρι, ώστε αν υπάρχουν αυτοκίνητα στο 2ο φανάρι να μην είναι αυτό για πάντα κόκκινο, επειδή στον κεντρικό δρόμο υπάρχει διαρκώς κίνηση.

Ας δούμε αρχικά την υλοποίηση με 1 αισθητήρα: Στον πίνακα που ακολουθεί φαίνονται οι συνδέσεις του αισθητήρα με το Arduino. Οι συνδέσεις μπορούν να γίνουν

με τη “μεσολάβηση ενός Breadboard.



| | |
|--------------------|---------------------|
| Αισθητήρας 1-GND | Υποδοχή GND Arduino |
| Αισθητήρας1 -VCC | Υποδοχή 5V Arduino |
| Αισθητήρας 1-Trig | Υποδοχή D8 Arduino |
| Αισθητήρας 1- Echo | Υποδοχή D9 Arduino |

Το πρόγραμμα μας για τη λειτουργία του συστήματος των φαναριών με έναν αισθητήρα βρίσκεται στην επόμενη σελίδα.

Ανάλυση του προγράμματος:

1. Έχει δημιουργηθεί μία μεταβλητή με το όνομα APOSTASI, στην οποία αποθηκεύεται η απόσταση του κοντινότερου αυτοκινήτου που κινείται προς το φανάρι.
2. Όταν ξεκινάει το πρόγραμμα για πάντα ο αισθητήρας μετράει σε πόση απόσταση βρίσκεται το κοντινότερο αυτοκίνητο και αποθηκεύει την απόσταση στην μεταβλητή.
3. Αν η απόσταση είναι μικρότερη από 20 εκ (αυτό θα το αποφασίσετε εσείς βλέποντας πάνω στο σκηνικό αν η απόσταση είναι πολύ μικρή ή μεγάλη), τότε το φανάρι αυτό θα είναι πάντα πράσινο και το άλλο πάντα κόκκινο.
4. Αν η απόσταση είναι μεγαλύτερη από 20 εκ. τότε ενεργοποιείται η λειτουργία του φαναριού με τον χρόνο.

when Arduino Uno starts up

forever

set APOSTASI to ∞ read ultrasonic sensor trig pin 8 echo pin 9

if APOSTASI < 20 then

set digital pin 2 output as low
set digital pin 3 output as low
set digital pin 4 output as high
set digital pin 5 output as high
set digital pin 6 output as low
set digital pin 7 output as low

wait 8 seconds

else

set digital pin 2 output as high
set digital pin 3 output as low
set digital pin 4 output as low
set digital pin 5 output as low
set digital pin 6 output as low
set digital pin 7 output as high

wait 6 seconds

set digital pin 2 output as high
set digital pin 3 output as low
set digital pin 4 output as low
set digital pin 5 output as low
set digital pin 6 output as high
set digital pin 7 output as low

wait 2 seconds

set digital pin 2 output as low
set digital pin 3 output as low
set digital pin 4 output as high
set digital pin 5 output as high
set digital pin 6 output as low
set digital pin 7 output as low

wait 8 seconds

set digital pin 2 output as low
set digital pin 3 output as high
set digital pin 4 output as low
set digital pin 5 output as high
set digital pin 6 output as low
set digital pin 7 output as low

wait 3 seconds

set digital pin 2 output as high
set digital pin 3 output as low
set digital pin 4 output as low
set digital pin 5 output as low
set digital pin 6 output as low
set digital pin 7 output as high

wait 6 seconds

set digital pin 2 output as high
set digital pin 3 output as low
set digital pin 4 output as low
set digital pin 5 output as low
set digital pin 6 output as high
set digital pin 7 output as low

wait 2 seconds

Για την υλοποίηση του φαναριού με 2 αισθητήρες (1 αισθητήρας σε κάθε φανάρι) θα πρέπει:

1. Να κάνουμε ακόμα μία μεταβλητή (πχ με το όνομα APOSTASI2).
2. Όταν ξεκινάει το πρόγραμμα να γίνεται έλεγχος και στα δυο φανάρια σε ποια απόσταση υπάρχει κάποιο αυτοκίνητο.
3. Θα πρέπει να υπάρχουν 4 διαφορετικές περιπτώσεις για το τι θα γίνεται με τα φανάρια.

Περίπτωση 1η: Και στα δύο φανάρια σε κοντινή απόσταση υπάρχουν αυτοκίνητα. Ενεργοποιείται η λειτουργία με χρόνο.

Περίπτωση 2η: Στο πρώτο φανάρι υπάρχουν αυτοκίνητα στο 2ο όχι. Ανάβει πράσινο στο 1ο φανάρι και κόκκινο στο 2ο.

Περίπτωση 3η: Στο πρώτο φανάρι δεν υπάρχουν αυτοκίνητα, υπάρχουν στο 2ο. Ανάβει πράσινο στο 2ο φανάρι και κόκκινο στο 1ο.

Περίπτωση 4η: Ενεργοποιείται η λειτουργία με το χρόνο.

