

# Φύλλο Εργασίας 1 (Λειτουργία και έλεγχος κινητήρων)

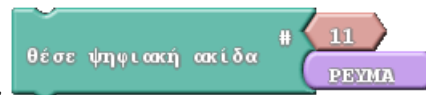
Στις παρακάτω δραστηριότητες θα δημιουργήσουμε το πρώτο πρόγραμμα για τον έλεγχο της κίνησης των κινητήρων στο περιβάλλον του Ardublock.

## Δραστηριότητα 1 (Έλεγχος του αριστερού κινητήρα)

Ο αριστερός κινητήρας του ρομποτικού οχήματος ελέγχεται από τις ψηφιακές εξόδους 10, 11 (digital pins 10, 11) του arduino.

Με την εντολή **θέσε ψηφιακή ακίδα..** μπορούμε να θέσουμε μια ψηφιακή ακίδα στην κατάσταση:


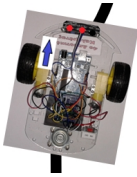
- PEYMA (High: +5V) ή στην κατάσταση
- OXI PEYMA (Low: 0 V).



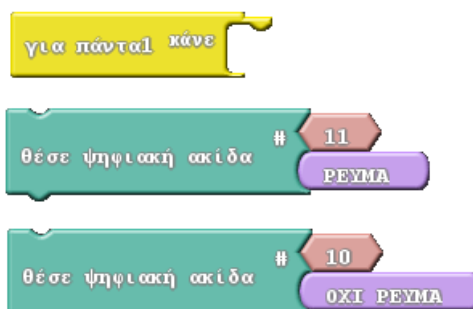
Όταν θέσουμε τις ψηφιακές ακίδες (10,11) στην κατάσταση Low τότε ο κινητήρας δεν παίρνει ρεύμα και άρα **σταματά**.

Όταν θέσουμε τη μία ψηφιακή ακίδα (π.χ. pin 11) στην κατάσταση High και την άλλη (pin 10) στην κατάσταση Low, τότε τροφοδοτείται με ρεύμα ο κινητήρας και τίθεται σε **κίνηση** ο αντίστοιχος τροχός.

1) Συμπληρώστε τον παρακάτω πίνακα τις τιμές που θα πρέπει να έχουν οι ψηφιακές ακίδες (pins 10, 11) που ελέγχουν την κίνηση του αριστερού κινητήρα του ρομποτικού οχήματος;

	Κατάσταση αριστερού κινητήρα	Τιμή Pin 10	Τιμή Pin 11
	Σταματημένος		
	Κινείται		

2) Χρησιμοποιώντας το περιβάλλον του Ardublock δημιουργήστε ένα πρόγραμμα που θα θέτει σε λειτουργία συνεχώς τον αριστερό κινητήρα. Θα χρειαστείτε τις εντολές:



3) **Αποθηκεύστε** το πρόγραμμά στο φάκελό σας με όνομα Motion1.

4) **Ανεβάστε** στο Arduino το πρόγραμμα που δημιουργήσατε. Μόλις γίνει η φόρτωση στο Arduino του προγράμματος, αποσυνδέστε από το καλώδιο USB το ρομποτικό όχημα και θέστε το σε λειτουργία σε ασφαλή χώρο για να δοκιμάσετε το πρόγραμμά σας.

## Δραστηριότητα 2 (Υπορουτίνα LeftMotorOn)

### 1) Δημιουργία της υπορουτίνας LeftMotorOn

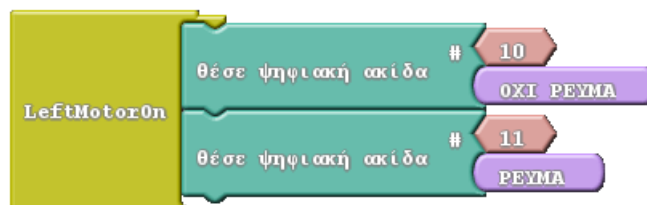
Ξαναγυρίστε στο περιβάλλον του Ardublock (διακόψτε τη λειτουργία του οχήματος και ξανασυνδέστε το στον Η/Υ). Τις 2 εντολές (θέσε ψηφιακή ακίδα ...) θα τις χρησιμοποιήσουμε πολλές φορές παρακάτω για να θέτουμε σε λειτουργία τον αριστερό κινητήρα. Για να μην τις επαναλαμβάνουμε συνέχεια, θα δημιουργήσουμε μια δική μας εντολή (**υπορουτίνα**) η οποία θα θέτει σε λειτουργία τον αριστερό κινητήρα όποτε τη χρησιμοποιούμε στο πρόγραμμά μας. Την εντολή αυτή θα την ονομάσουμε **LeftMotorOn** για να μας “θυμίζει” ότι θέτει τον αριστερό κινητήρα σε κίνηση.

Θα χρειαστείτε την εντολή από την καρτέλα Έλεγχος:

Αλλάξτε το όνομα της σε



Η εντολή **LeftMotorOn** θέλουμε όταν εκτελείται να θέτει σε λειτουργία τον αριστερό κινητήρα. Άρα στην αντίστοιχη εσοχή θα βάλουμε τις 2 εντολές που χρησιμοποιήσαμε στη δραστηριότητα 1 για να θέτουμε τις ψηφιακές ακίδες 10, 11 σε κατάλληλη τιμή:



### 2) Χρήση της εντολής (υπορουτίνας) LeftMotorOn

Από εδώ και στο εξής, όταν θέλουμε να θέσουμε σε λειτουργία τον αριστερό κινητήρα, δεν έχουμε παρά να εκτελέσουμε την εντολή (υπορουτίνα) που δημιουργήσαμε. Για να το κάνουμε αυτό θα χρειαστούμε από την καρτέλα Έλεγχος την εντολή:

Φυσικά θα πρέπει να βάλουμε το όνομα της δικής μας υπορουτίνας, δηλαδή:



**Τροποποιήστε** τώρα το πρόγραμμα που δημιουργήσατε στη δραστηριότητα 1 ως εξής:



**Ανεβάστε** το πρόγραμμα στο Arduino και δοκιμάστε ξανά τη λειτουργία του οχήματος.  
**Αποθηκεύστε** το πρόγραμμα με όνομα Motion2.

### Συμπέρασμα

Από τις δοκιμές λειτουργίας που κάνατε στα 2 προγράμματα που δημιουργήσατε στις δραστηριότητες 1 και 2 πιστεύετε ότι είναι τελικά ισοδύναμα (κάνουν ακριβώς το ίδιο);



Πρόγραμμα δραστηριότητας 1



Πρόγραμμα δραστηριότητας 2

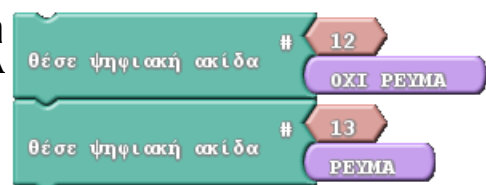
Απάντηση: \_\_\_\_\_

### Δραστηριότητα 3 (Έλεγχος του δεξιού κινητήρα)

Ο δεξιός κινητήρας του ρομποτικού οχήματος ελέγχεται από τις ψηφιακές εξόδους 12, 13 (digital pins 12, 13) του Arduino.

Με την εντολή **θέσε ψηφιακή ακίδα..**

μπορούμε να θέσουμε μια ψηφιακή ακίδα στην κατάσταση ΡΕΥΜΑ (High: +5V) ή στην κατάσταση ΟΧΙ ΡΕΥΜΑ (Low: 0 V).



Όταν θέσουμε τις ψηφιακές ακίδες (12,13) στην κατάσταση Low τότε ο κινητήρας δεν παίρνει ρεύμα και άρα **σταματά**.

Όταν θέσουμε τη μία ψηφιακή ακίδα (π.χ. pin 12) στην κατάσταση High και την άλλη (pin 13) στην κατάσταση Low, τότε τροφοδοτείται με ρεύμα ο κινητήρας και τίθεται σε **κίνηση** ο αντίστοιχος τροχός.

1) Συμπληρώστε τον παρακάτω πίνακα τις τιμές που θα πρέπει να έχουν οι ψηφιακές ακίδες (pins 12, 13) που ελέγχουν την κίνηση του αριστερού κινητήρα του ρομποτικού οχήματος;

	Κατάσταση δεξιού κινητήρα	Τιμή Pin 12	Τιμή Pin 13
	Σταματημένος		
	Κινείται		

2) Δημιουργία της υπορουτίνας *RightMotorOn*

Ξαναγυρίστε στο περιβάλλον του Ardublock (διακόψτε τη λειτουργία του οχήματος και ξανασυνδέστε το στον Η/Υ). Τις 2 εντολές (θέσε ψηφιακή ακίδα ...) θα τις χρησιμοποιήσουμε πολλές φορές παρακάτω για να θέτουμε σε λειτουργία τον δεξιό κινητήρα.

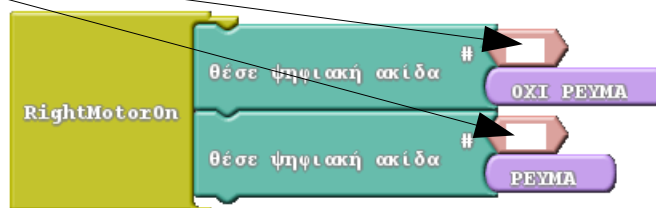
Με παρόμοιο τρόπο όπως στη δραστηριότητα 2, θα δημιουργήσουμε μια δική μας εντολή (υπορουτίνα) η οποία θα θέτει σε λειτουργία τον δεξιό κινητήρα όταν τη χρησιμοποιούμε στο πρόγραμμά μας. Την εντολή αυτή θα την ονομάσουμε **RightMotorOn** για να μας “θυμίζει” ότι θέτει τον δεξιό κινητήρα σε κίνηση.

Θα χρειαστείτε την εντολή από την καρτέλα Έλεγχος:



Αλλάξτε το όνομα της σε **RightMotorOn**.

Συμπληρώστε στην παρακάτω εικόνα τις τιμές των ψηφιακών ακίδων:



Δημιουργήστε την υπορουτίνα RightMotorOn και αποθηκεύστε το αρχείο σας με όνομα Motion3.

### 3) Χρήση της εντολής (υπορουτίνας) RightMotorOn

Από εδώ και στο εξής, όταν θέλουμε να θέσουμε σε λειτουργία τον δεξιό κινητήρα, δεν έχουμε παρά να εκτελέσουμε την εντολή (υπορουτίνα) που δημιουργήσαμε. Για να το κάνουμε αυτό θα χρειαστούμε από την καρτέλα Έλεγχος την εντολή:

υπορουτίνα

Φυσικά θα πρέπει να βάλουμε το όνομα της δικής μας υπορουτίνας, δηλαδή:

RightMotorOn

Τροποποιήστε τώρα το πρόγραμμα που δημιουργήσατε στη δραστηριότητα 2 ώστε να κινείται διαρκώς ο δεξιός κινητήρας. Ποια υπορουτίνα θα χρησιμοποιήσετε (συμπληρώστε το κενό);



Ανεβάστε το πρόγραμμα στο Arduino και δοκιμάστε ξανά τη λειτουργία του οχήματος. Αποθηκεύστε το πρόγραμμά σας.

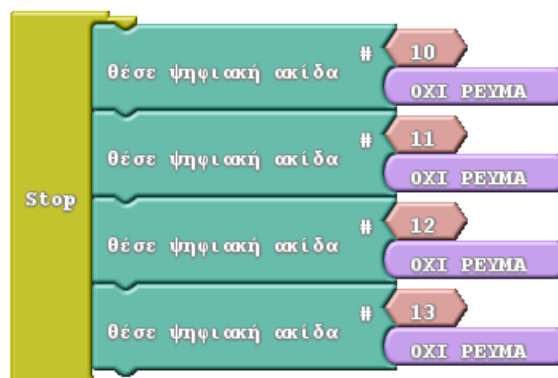
### Συμπέρασμα

Ολοκληρώνοντας τις δραστηριότητες, θα πρέπει να έχετε δημιουργήσει τις υπορουτίνες LeftMotorOn και RightMotorOn που θέτουν σε λειτουργία τον αριστερό και δεξιό κινητήρα αντίστοιχα.

## ΕΠΕΚΤΑΣΕΙΣ

### Δραστηριότητα 4 (Υπορουτίνα Stop για σταμάτημα οχήματος)

Δημιουργήστε μια ακόμα υπορουτίνα με όνομα Stop που θα σταματά εντελώς το όχημα. Για να γίνει αυτό θα πρέπει να σταματήσετε και τον αριστερό και το δεξιό κινητήρα. Αυτό σημαίνει ότι όλες οι ψηφιακές έξοδοι (10, 11, 12, 13) θα πρέπει με κατάλληλες εντολές να γίνουν Low (OXI ΡΕΥΜΑ).



### Δραστηριότητα 5 (Υπορουτίνες LeftMotorOff & RightMotorOff για σταμάτημα κινητήρων)

Δημιουργήστε δύο ακόμα υπορουτίνες με όνομα LeftMotorOff και RightMotorOff που θα σταματά τον αριστερό και δεξιό κινητήρα αντίστοιχα.

Υπόδειξη: Μπορείτε να

- αντιγράψετε τις LeftMotorOn και RightMotorOff
- να τους αλλάξετε τα ονόματα σε LeftMotorOff και RightMotorOff αντίστοιχα και
- να αλλάξετε τις τιμές των ψηφιακών ακίδων 10, 11, 12, 13 σε κατάλληλη τιμή.

### Σύνοψη

Θα πρέπει μέχρι τώρα να έχετε δημιουργήσει τις παρακάτω υπορουτίνες που θα χρησιμοποιήσουμε στις επόμενες εφαρμογές.

<b>Υπορουτίνα</b>	<b>Λειτουργία</b>
<b>LeftMotorOn</b>	Θέτει σε λειτουργία τον αριστερό κινητήρα
<b>RightMotorOn</b>	Θέτει σε λειτουργία τον δεξιό κινητήρα
<b>LeftMotorOff</b>	Σταματά τον αριστερό κινητήρα
<b>RightMotorOff</b>	Σταματά τον δεξιό κινητήρα
<b>Stop</b>	Σταματά και του δύο κινητήρες