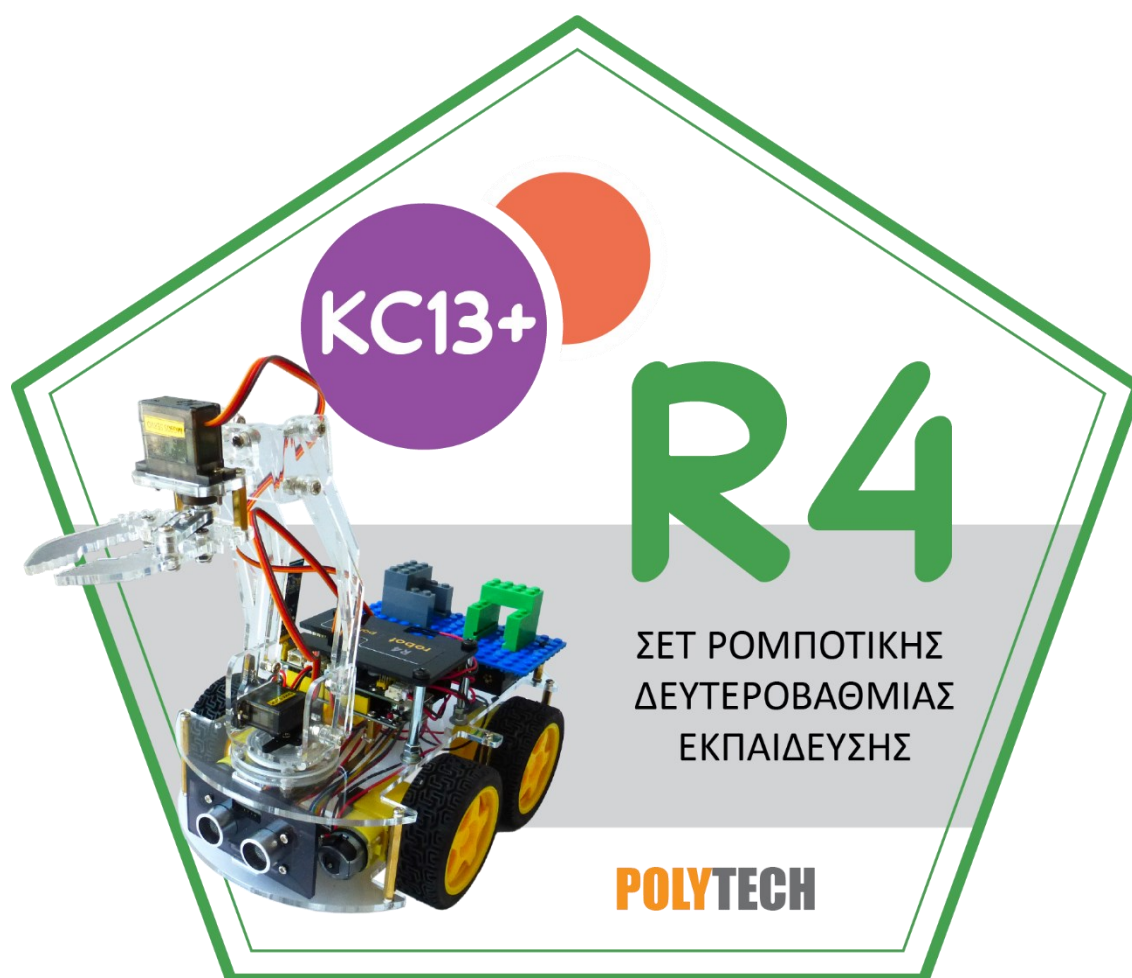


ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ ΚΑΙ ΔΡΑΣΤΗΡΙΟΤΗΤΩΝ ΓΙΑ ΤΟΝ ΕΚΠΑΙΔΕΥΤΙΚΟ



Περιεχόμενα

1.1 Εισαγωγή στο ARD:icon	3
1.1.1 Εγκατάσταση της εφαρμογής ARD:icon	3
1.1.2 Εφαρμογή ARD:icon	7
1.2 Σύνδεση Ελεγκτή και Περιφερειακών μονάδων	11
1.2.1 Σωστή συναρμολόγηση των περιφερειακών συσκευών με τα τουβλάκια	12
1.3 Εντολές Προγραμματισμού	16
1.3.1. Μενού Επιλογής Εντολών Λειτουργίας	17
1.2.2 Ανάλυση Εντολών Λειτουργίας	20
1.3.3. Μενού Επιλογής Βασικών Μαθηματικών Εντολών	26
1.3.4 Αποθήκευση και Επαναφορά Ρουτίνας.....	29
1.4 Εντολές Προγραμματισμού Ρομπότ R4	33
1.4.1 Φόρτωση προγράμματος λειτουργίας Bluetooth για προγραμματισμό με ARD:icon	33
1.4.2 Ασύρματη Σύνδεση Ρομπότ R4 με τον Υπολογιστή (Bluetooth connection)	43
1.4.3 Προγραμματισμός εντολής “R4 Κίνηση κινητήρων”	45
1.4.4 Προγραμματισμός εντολής “R4 Κίνηση βραχίονα”	46
1.5 Δραστηριότητες με ARD:icon	49
Δραστηριότητα 1: Προγραμματισμός συνδυασμού κινήσεων Ρομπότ	49
Δραστηριότητα 2: Προγραμματισμός συνδυασμού κινήσεων Βραχίονα	51
Δραστηριότητα 3: Προγραμματισμός Ταχύτητας Ρομπότ	53
Δραστηριότητα 4: Συνδυασμός κινήσεων κινητήρων και βραχίονα ρομπότ.....	55
Δραστηριότητα 5: Συνδυασμός κινήσεων κινητήρων και βραχίονα ρομπότ II	58
Δραστηριότητα 6: Έναρξη Λειτουργίας Ρομπότ με Ήχο	63
Δραστηριότητα 7: Έναρξη Λειτουργίας Ρομπότ με Ήχο II	67
Δραστηριότητα 8: Έναρξη Λειτουργίας Ρομπότ με βάση το επίπεδο Φωτισμού.....	71
Δραστηριότητα 9: Έναρξη Λειτουργίας Ρομπότ με βάση το επίπεδο Φωτισμού II	74
Δραστηριότητα 10: Προγραμματισμός κινήσεων ρομπότ με τον αισθητήρα γωνίας	79
Δραστηριότητα 11: Χειροκίνητος έλεγχος ταχύτητας ρομπότ.....	82
Δραστηριότητα 12: Χειροκίνητος έλεγχος ταχύτητας και συλλογής αντικειμένου	85
Δραστηριότητα 13: Προγραμματισμός διακόπτη έκτακτης ανάγκης.....	91
Δραστηριότητα 14: Προγραμματισμός διακόπτη έκτακτης ανάγκης II	94
Δραστηριότητα 15: Έναρξη Λειτουργίας Ρομπότ με Ήχο.....	97
Δραστηριότητα 16: Έναρξη Λειτουργίας Ρομπότ με Φως και διακοπή με ήχο	100
Δραστηριότητα 17: Προγραμματισμός συλλογής αντικειμένου ανάλογα με το χρώμα του αντικειμένου.....	103

Δραστηριότητα 18: Προγραμματισμός διακόπτη έκτακτης ανάγκης III	108
---	-----

1.1 Εισαγωγή στο ARD:icon

Το πρόγραμμα ARD:icon, αποτελεί μια εύχρηστη εφαρμογή προγραμματισμού για ελεγκτές Arduino.

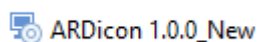
- Διαθέτει ένα διαδραστικό περιβάλλον κωδικοποίησης, που βασίζεται σε εικονίδια και μπλοκ εντολών, που κάνει τον προγραμματισμό εύκολο σαν παιχνίδι έτσι, ώστε ακόμα και τα μικρά παιδιά να μπορέσουν να κατανοήσουν εύκολα τη λογική των υπολογιστών.
- Διαθέτει βιβλιοθήκη συσκευών εισόδου και εξόδου, που βασίζονται σε εικονίδια (έγχρωμες λυχνίες LED, αισθητήρες, κινητήρες, πολλοί τύποι ηλεκτρονικών διακοπών, ροοστάτες κ.λπ.)
- Το σχήμα συνάρτησης-λογικών μπλοκ προσφέρει έναν μοναδικό, ευφυή και απλοποιημένο τρόπο εκπαίδευσης στον προγραμματισμό και την επεξεργασία διαφόρων εισόδων, την εφαρμογή θεμελιωδών εννοιών προγραμματισμού, όπως επανάληψη, δηλώσεις υπό όρους και μεταβλητές, προκειμένου να δημιουργηθούν τα επιθυμητά αποτελέσματα.

Το ARD:icon, είναι μια πλατφόρμα προγραμματισμού, που χρησιμοποιεί πλακίδια προγραμματισμού και εικονίδια δομικών στοιχείων, για να δημιουργήσει μια εύκολη εμπειρία “plug and play” για παιδιά.

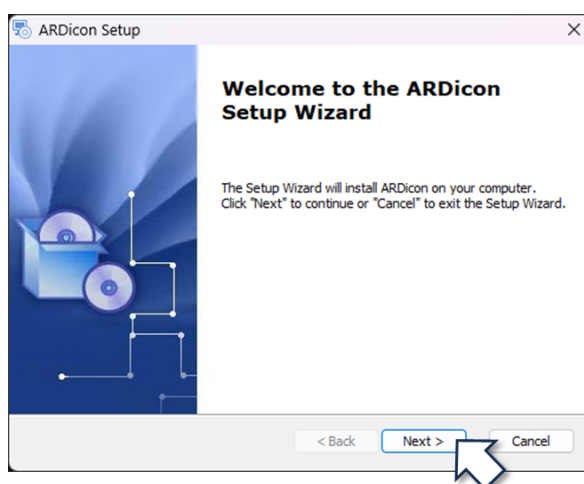
1.1.1 Εγκατάσταση της εφαρμογής ARD:icon

Για να ξεκινήσετε να προγραμματίζετε τις εργασίες με την εφαρμογή ARD:icon, το μόνο που έχετε να κάνετε είναι απλά να εγκαταστήσετε την εφαρμογή στον υπολογιστή σας και να τρέξετε την εφαρμογή.

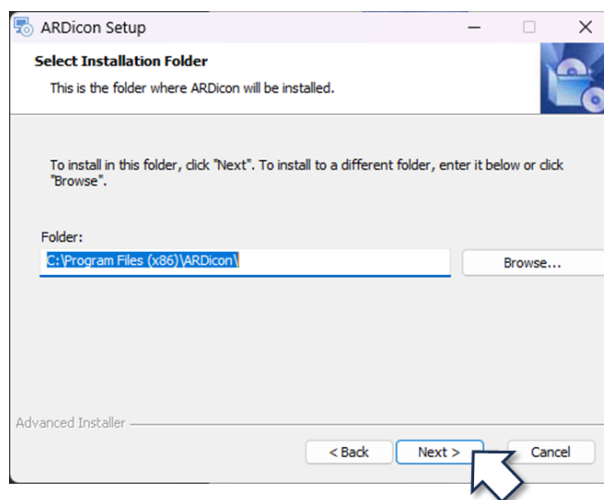
Για να ξεκινήσει η εγκατάσταση της εφαρμογής, κάνετε διπλό κλικ πάνω στο εικονίδιο



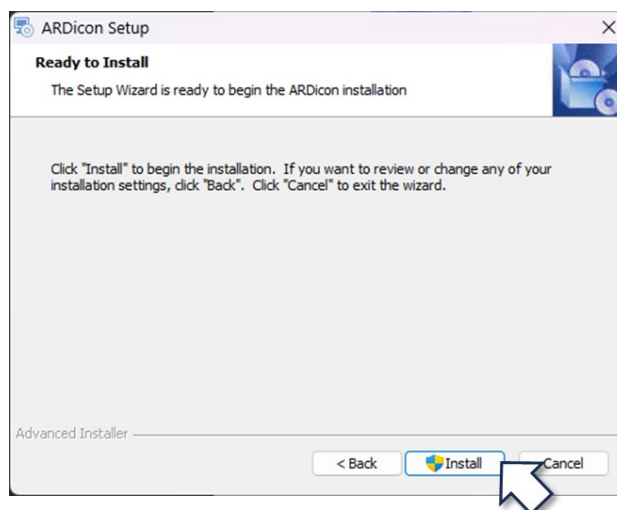
Στο αναδυόμενο παράθυρο, επιλέξτε “Next”



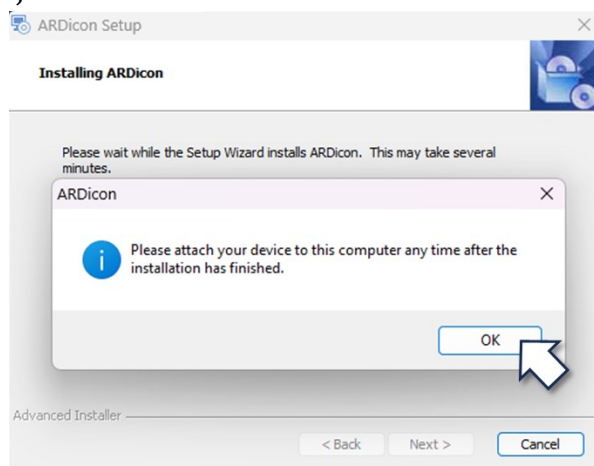
Επιλέξτε έναν φάκελο προορισμού, όπου θέλετε να γίνει η εγκατάσταση της εφαρμογής (ή αφήστε την υπάρχουσα διαδρομή) και πατήστε “Next”.



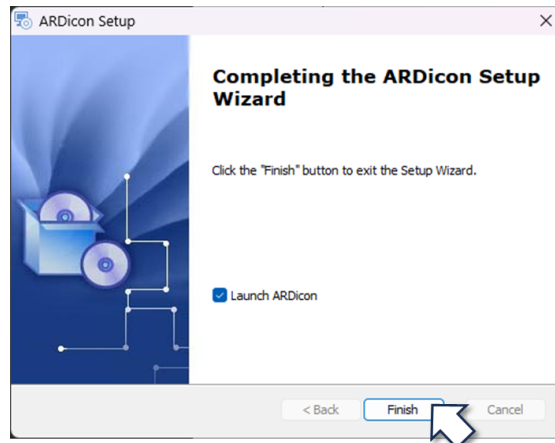
Κάνοντας αριστερό κλικ στο κουμπί “Install” θα ξεκινήσει η διαδικασία της εγκατάστασης.



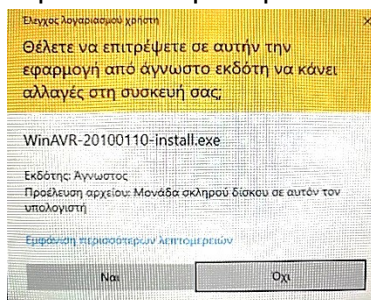
Στο επόμενο παράθυρο επιλέξτε “ok”.



Μόλις ολοκληρωθεί η εγκατάσταση, πατήστε “Finish”.

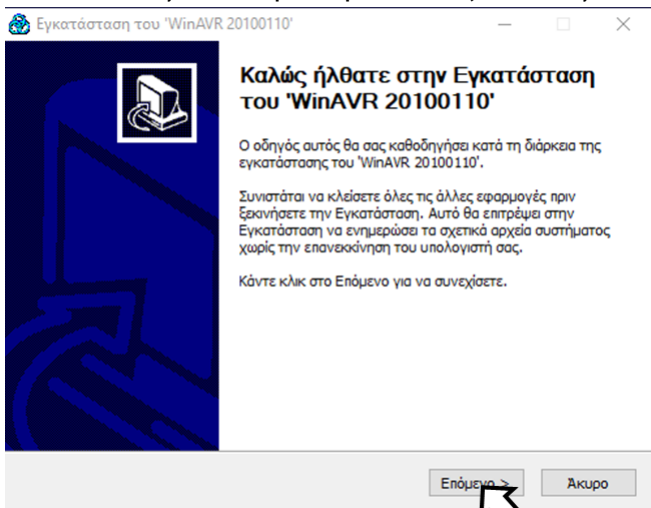


Όταν η εγκατάσταση του λογισμικού ολοκληρωθεί, στην επιφάνεια του υπολογιστή θα εμφανιστεί το παρακάτω παράθυρο.

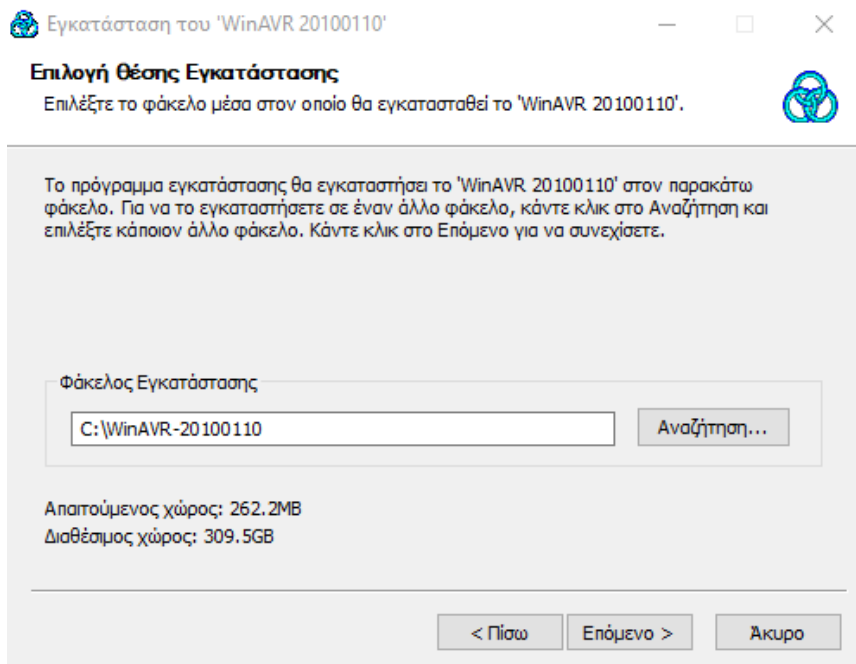


Επιλέξτε «**ΝΑΙ**» και προχωρήστε στην εγκατάσταση του λογισμικού WinAVR.

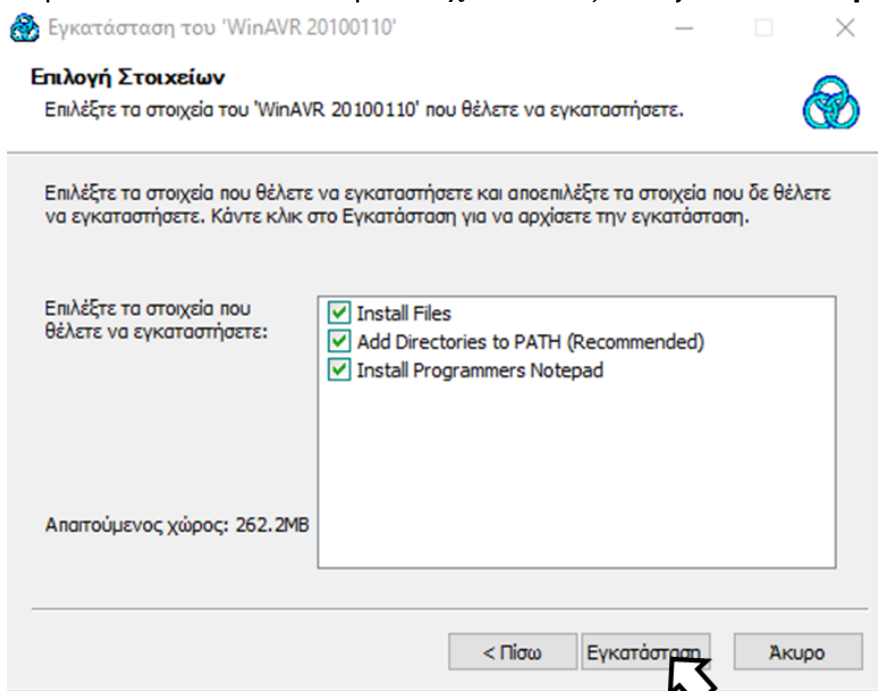
Στο αναδυόμενο παράθυρο, επιλέξτε “Επόμενο”.



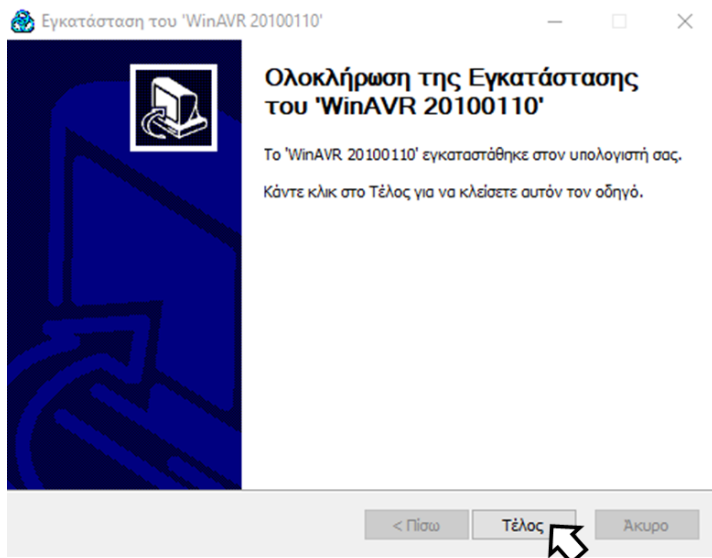
Στο επόμενο παράθυρο επιλέξτε έναν φάκελο προορισμού, όπου θέλετε να γίνει η εγκατάσταση της εφαρμογής επιλέγοντας αναζήτηση (ή αφήστε την υπάρχουσα διαδρομή) και πατήστε **“Επόμενο”**.



Στο επόμενο παράθυρο επιλέξτε τα στοιχεία που θέλετε να εγκαταστήσετε όπως φαίνεται στην παρακάτω εικόνα και στην συνέχεια επιλέξτε **«Εγκατάσταση»**.



Τέλος επιλέξτε «**Τέλος**» για να ολοκληρωθεί η διαδικασία.



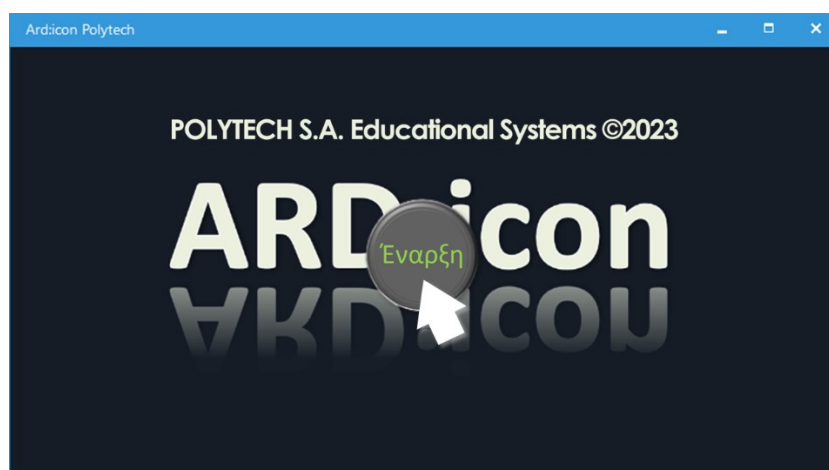
Αφού τελειώσετε και την διαδικασία εγκατάστασης του winAVR (κλείστε την σελίδα του winAVR που ανοίγει στο διαδίκτυο) τότε στην επιφάνεια του υπολογιστή, αυτόματα θα δημιουργηθεί το εικονίδιο συντόμευσης της εφαρμογής. Με διπλό κλικ πάνω στο εικονίδιο, μπορείτε να ανοίξετε την εφαρμογή “ARD:icon”.

1.1.2 Εφαρμογή ARD:icon

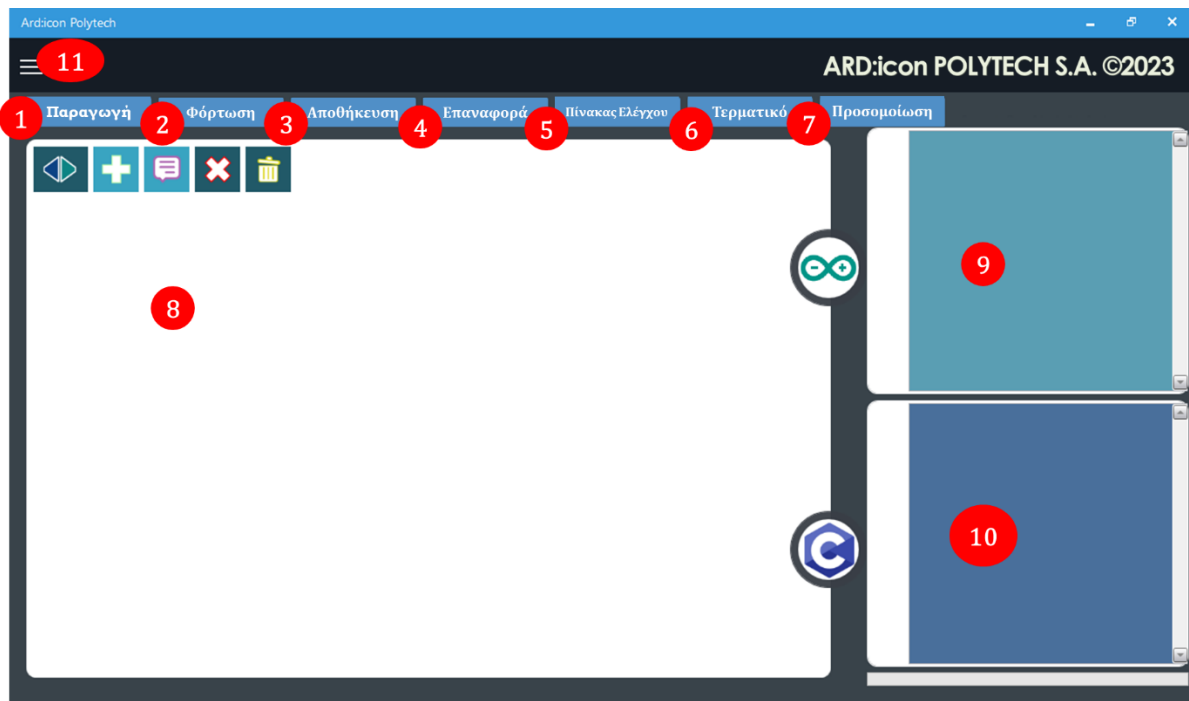
Ανοίξτε την εφαρμογή, επιλέγοντας το εικονίδιο της εφαρμογής



. Στην επόμενη οθόνη, επιλέξτε Start.



Η αρχική οθόνη της εφαρμογής θα εμφανιστεί στην οθόνη του υπολογιστή σας.



Μενού Επιλογών

1. Παραγωγή

Με το κουμπί Παραγωγή, μπορείτε να παράγετε τον κώδικα που προγραμματίσατε σε Κώδικα Arduino και κώδικα C++.

2. Φόρτωση

Με το κουμπί Φόρτωση, μπορείτε να φορτώσετε τον κώδικα στη συσκευή Arduino.

3. Αποθήκευση

Με το κουμπί Αποθήκευση, μπορείτε να αποθηκεύσετε τον κώδικα (ρουτίνα) που προγραμματίσατε, προκειμένου να τον χρησιμοποιήσετε σε κάποια άλλη εργασία.

4. Επαναφορά

Με το κουμπί Επαναφορά, μπορείτε να επαναφέρετε κάποια από τις ρουτίνες που έχετε αποθηκεύσει.

5. Πίνακας Ελέγχου

Με το κουμπί Πίνακας Ελέγχου, μπορείτε να εισέλθετε στην οθόνη ελέγχου των Συσκευών που έχετε προγραμματίσει.

6. Τερματικό

Με το κουμπί Τερματικό, μπορείτε να ανοίξετε την οθόνη του τερματικού, όπου τυπώνονται οι τιμές των μεταβλητών που εσείς προγραμματίζετε (εντολή "[serialprint](#)"). Στην ουσία είναι ένας εικονικός εκτυπωτής.

7. Προσομοίωση

Με το κουμπί Προσομοίωση, μπορείτε να ανοίξετε την οθόνη προσομοίωσης του κώδικα που προγραμματίσατε.

8. Οθόνη προγραμματισμού

Σε αυτή την οθόνη ξεκινάτε να προγραμματίζετε τις εργασίες σας. Όπως βλέπετε πάνω στην περιοχή προγραμματισμού, υπάρχει μια σειρά από κουμπιά. Ας δούμε τη λειτουργία καθενός από αυτά ξεχωριστά:



A. Το πλήκτρο  ορίζει το επίπεδο του κώδικα, που χρησιμοποιείται για ένθετες συναρτήσεις και υπορουτίνες. Διαθέτει 3 επίπεδα. Πιέστε το κουμπί, για να μεταφερθείτε στην γραμμή 2 και 3 αλλά και για να επιστρέψετε στην γραμμή 1.

Πάντα ξεκινάμε μία καινούργια ρουτίνα ή υπορουτίνα από το επίπεδο (γραμμή) 1.

B. Το πλήκτρο  σας εισάγει στο περιβάλλον προγραμματισμού, για το επίπεδο που έχετε επιλέξει.

Γ. Το πλήκτρο  σας δίνει τη δυνατότητα να προσθέσετε κάποιο σχόλιο

Δ. Το πλήκτρο  διαγράφει τις τελευταίες εντολές που έχετε επιλέξει

Ε. Το πλήκτρο  διαγράφει όλες τις εντολές που έχετε επιλέξει

9. Οθόνη εμφάνισης κώδικα, που δημιουργείται αυτόματα από το Arduino/**Arduino Code**

10.Οθόνη εμφάνισης κώδικα C ,που δημιουργείται αυτόματα/ C code

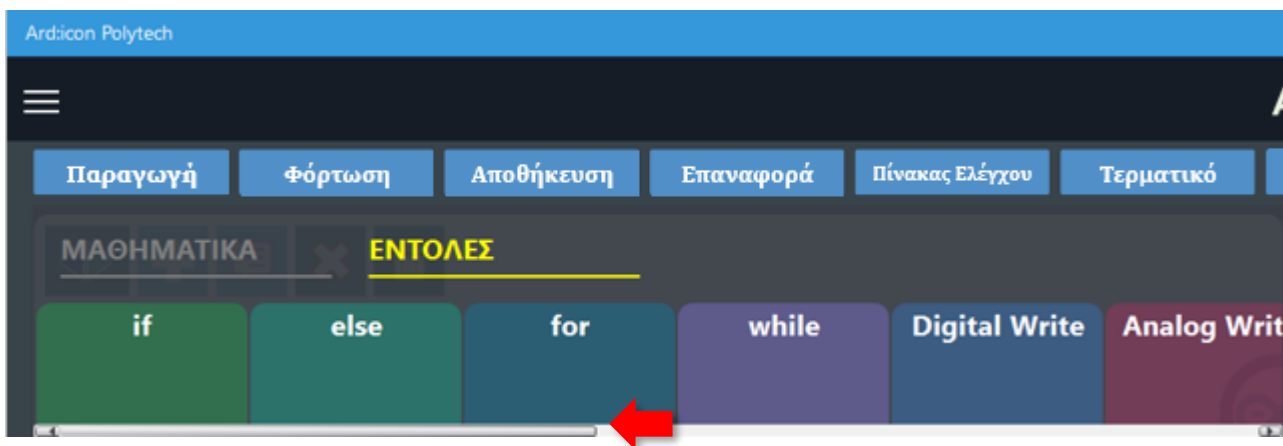
11.Μενού επιπλέον επιλογών. Πατώντας πάνω στο εικονίδιο, θα δείτε τους συνδέσμους, όπου μπορείτε να κατεβάσετε τα εγχειρίδια λειτουργίας των λογισμικών, καθώς και τον Γρήγορο Οδηγό Δραστηριοτήτων.

Για να ξεκινήσετε να προγραμματίζετε ένα έργο, πρώτα επιλέγετε το επίπεδο του κώδικα

χρησιμοποιώντας το κουμπί  στην περιοχή προγραμματισμού. Πάντα ξεκινάτε μια νέα άσκηση από το επίπεδο ένα.

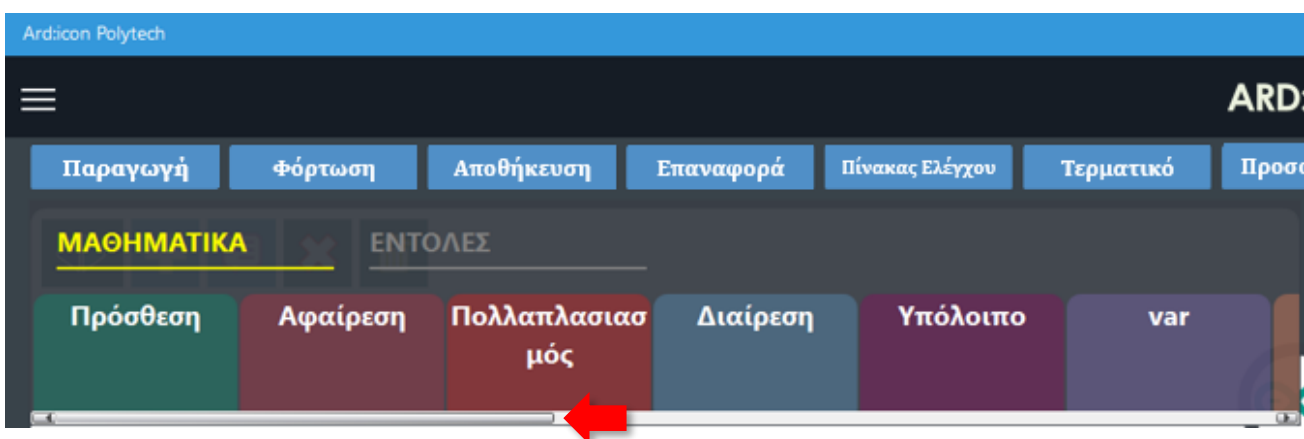
Στη συνέχεια, πατώντας το κουμπί  , εισέρχεστε στο περιβάλλον των εντολών. Εμφανίζονται οι εξής επιλογές,

A. Μενού επιλογών Εντολών Λειτουργίας



Μπορείτε να μεταβείτε στις υπόλοιπες εντολές, μετακινώντας την μπάρα στο κάτω μέρος

B. Μενού επιλογών βασικών μαθηματικών εντολών



Μπορείτε να μεταβείτε στις υπόλοιπες εντολές, μετακινώντας την μπάρα στο κάτω μέρος

1.2 Σύνδεση Ελεγκτή και Περιφερειακών μονάδων

Για να ξεκινήσετε να προγραμματίζετε με την εφαρμογή ARD:icon θα χρειαστείτε τον ελεγκτή ARD:icon τον οποίο πρέπει να συνδέσετε με τον υπολογιστή σας χρησιμοποιώντας το καλώδιο USB.



Συνδέστε την άκρη USB Type A στον υπολογιστή σας και στη συνέχεια την άκρη USB

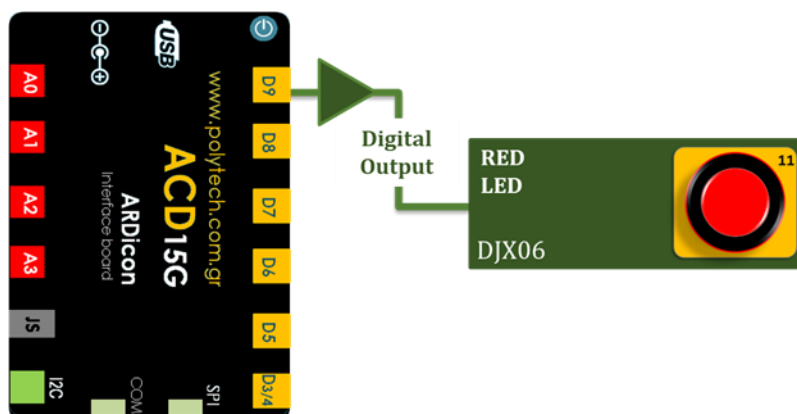


Type B στη θύρα USB του ελεγκτή ARD:icon.

Για να συνδέσετε τις περιφερειακές συσκευές (αισθητήρες, ενεργοποιητές και εξόδους) χρησιμοποιήστε τα καλώδια UTP.

Συνδέστε το ένα άκρο του καλωδίου στη συσκευή και το άλλο άκρο στην ανάλογη θύρα του ελεγκτή σύμφωνα με τις οδηγίες που θα βρείτε σε κάθε δραστηριότητα. Όπως το παρακάτω παράδειγμα:

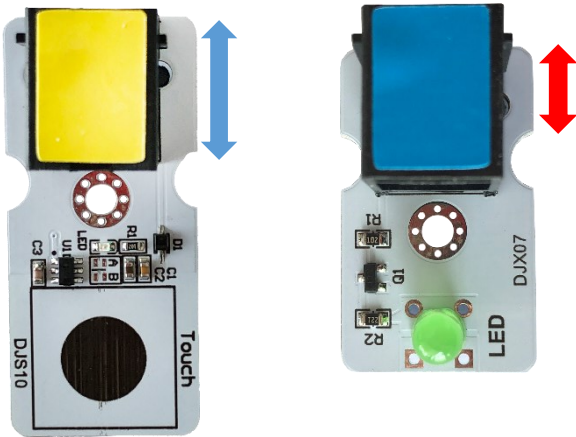
- Κόκκινο LED (DJX06) στη θύρα D9



1.2.1 Σωστή συναρμολόγηση των περιφερειακών συσκευών με τα τουβλάκια

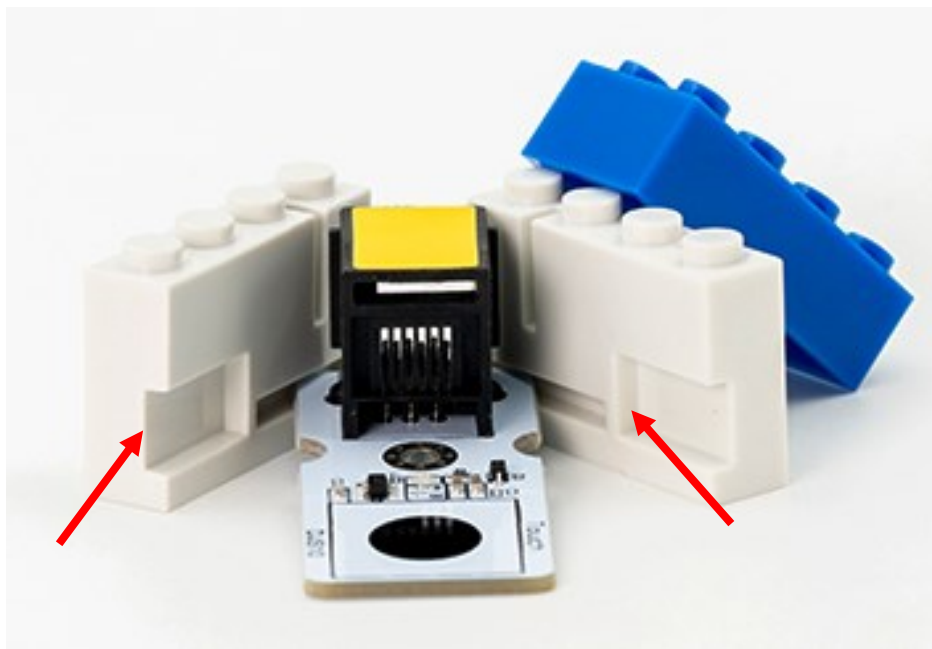
Τα τουβλάκια που συνοδεύουν τις περιφερειακές μονάδες εισόδου και εξόδου μπορούν να συναρμολογηθούν με δύο διαφορετικούς τρόπους ανάλογα με το μέγεθος της πλακέτας στη πλευρά σύνδεσης του καλωδίου RJ11.

Πιο συγκεκριμένα όπως μπορείτε να δείτε στις παρακάτω εικόνες, οι πλακέτες του αισθητήρα αφής και του πράσινου LED διαφέρουν σε μέγεθος. Στον αισθητήρα αφής η πλευρά της σύνδεσης του καλωδίου (**μπλε βέλος**) είναι μεγαλύτερη από αυτή του LED (**κόκκινο βέλος**).

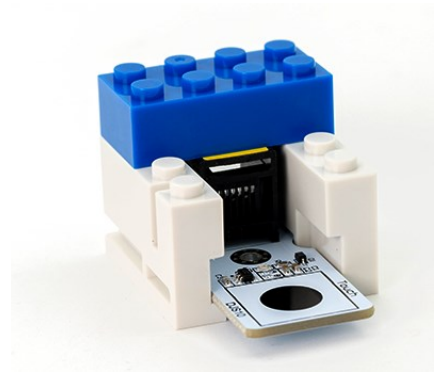


Α. Συσκευή με μέγεθος όπως ο αισθητήρας αφής.

Αρχικά, τοποθετείστε τη συσκευή στα πλαϊνά τουβλάκια. Οι εγκοπές στα τουβλάκια πρέπει να είναι όπως δείχνει η παρακάτω εικόνα, για να μπορέσει να τοποθετηθεί σωστά η συσκευή.

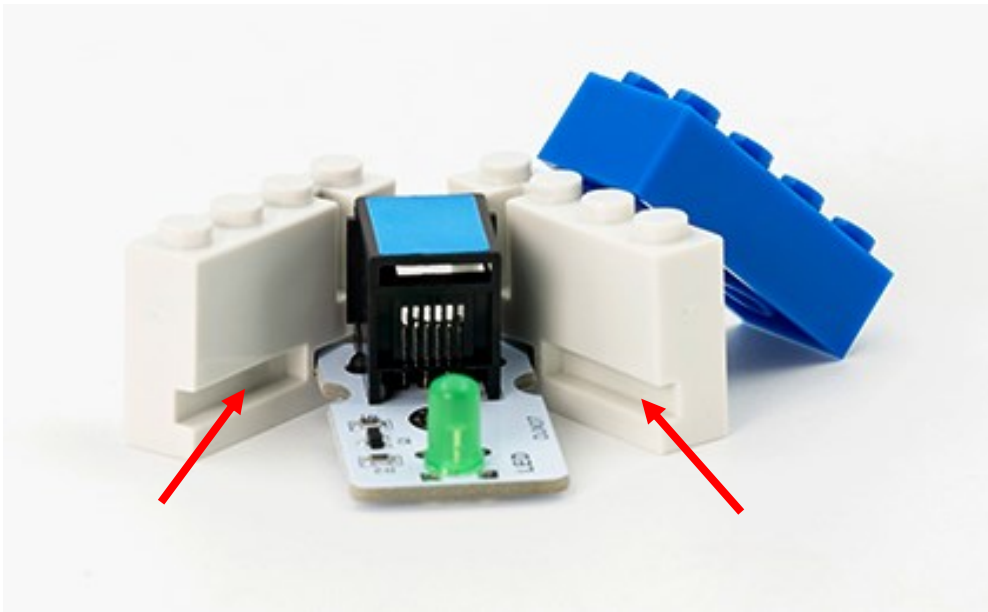


Τέλος, τοποθετείστε το μπλε τουβλάκι στο πάνω μέρος.

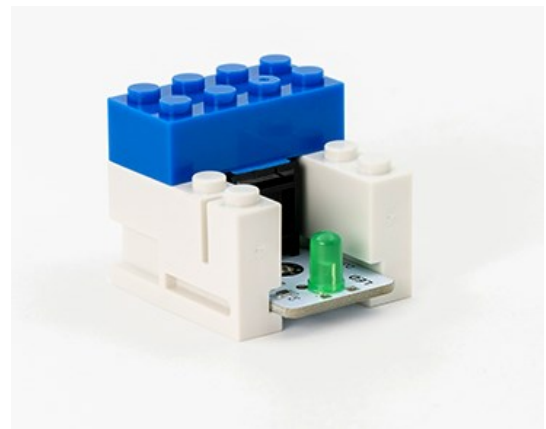


Β. Συσκευή με μέγεθος όπως το πράσινο LED.

Αρχικά, τοποθετείστε τη συσκευή στα πλαϊνά τουβλάκια. Οι εγκοπές στα τουβλάκια πρέπει να είναι όπως δείχνει η παρακάτω εικόνα, για να μπορέσει να τοποθετηθεί σωστά η συσκευή.

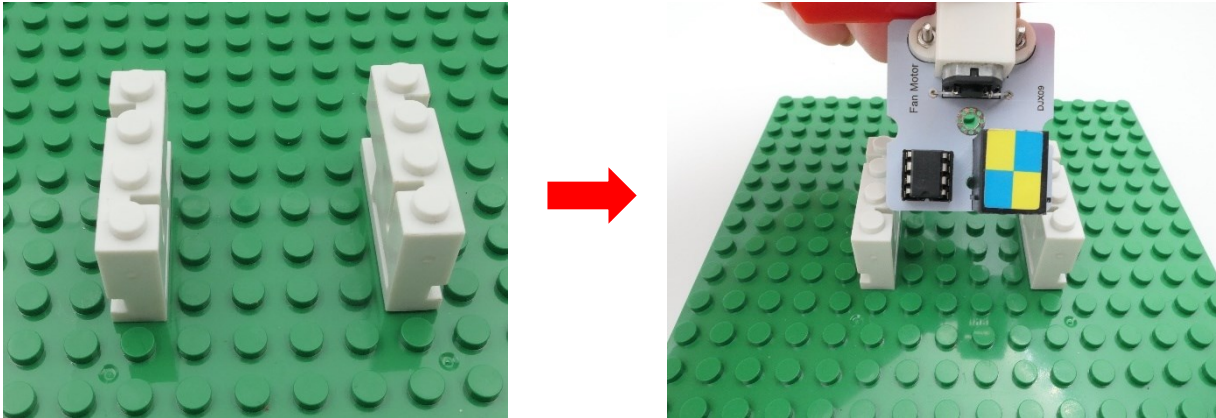


Τέλος, τοποθετείστε το μπλε τουβλάκι στο πάνω μέρος.

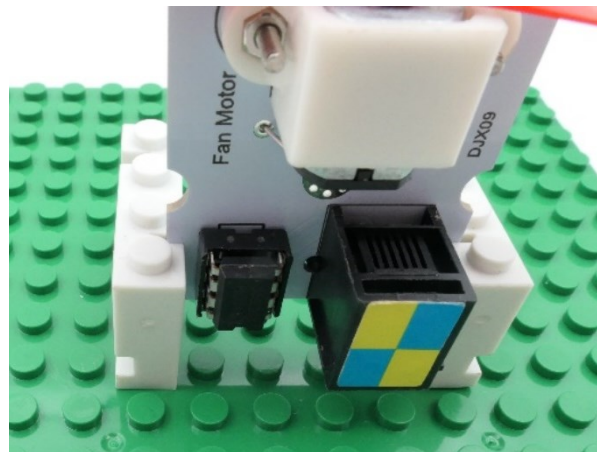


Κάποιες από τις συσκευές έχουν αρκετά μεγαλύτερη πλακέτα σε πλάτος και δεν μπορούν να στερεωθούν με αυτόν το τρόπο στα τουβλάκια. Δείτε παρακάτω στις εικόνες τον τρόπο που μπορείτε να στερεώσετε αυτές τις συσκευές στα τουβλάκια.

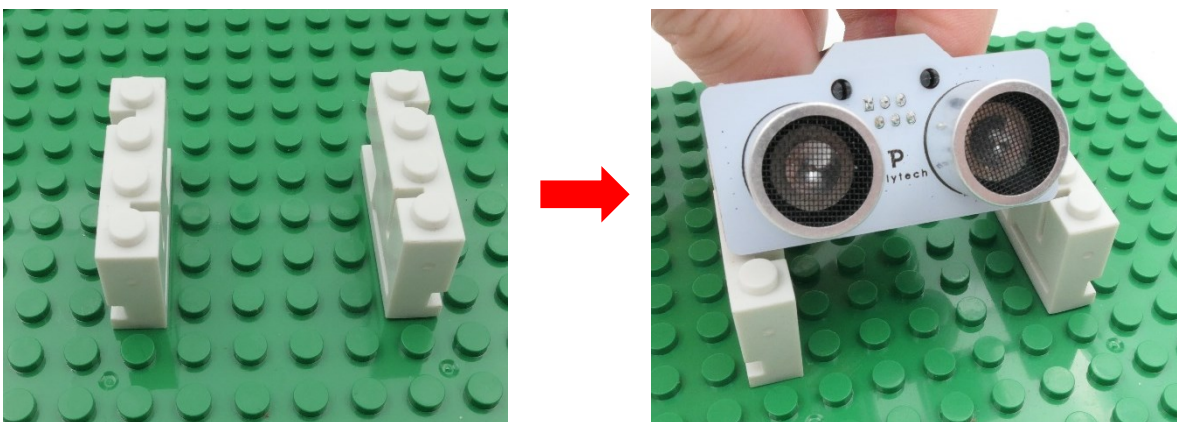
1. Μονάδα Step Motor με Προπέλα – DJX09



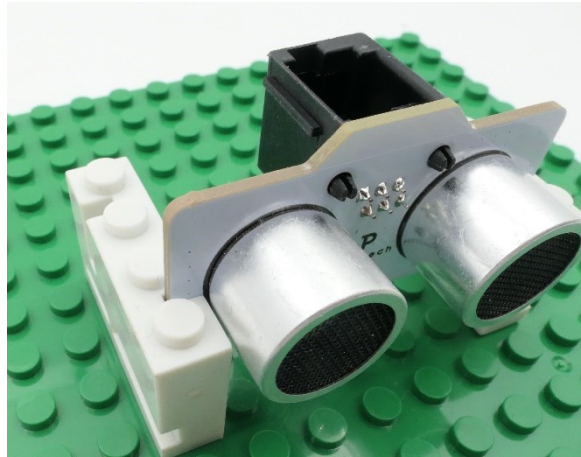
Στερεώστε τη συσκευή στις εγκοπές με το μικρότερο άνοιγμα.



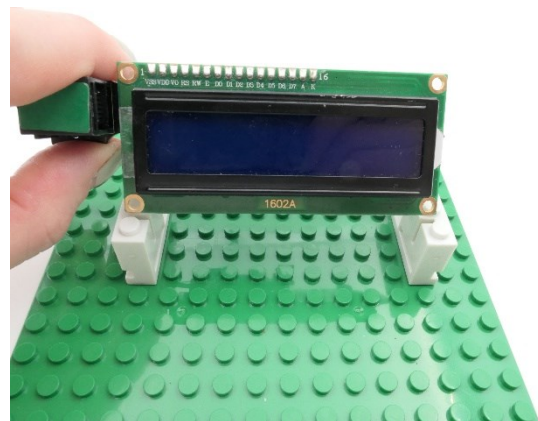
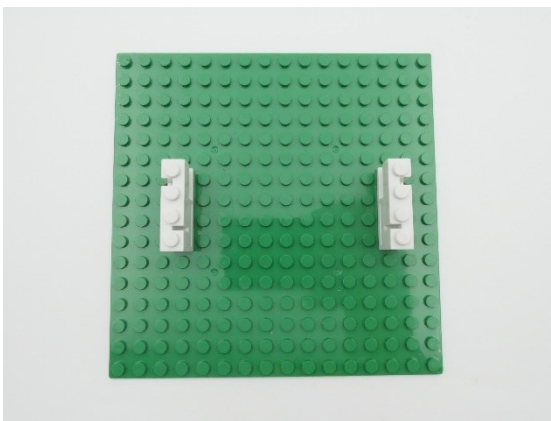
2. Μονάδα Αισθητήρα Απόστασης Υπερήχων – DJS22



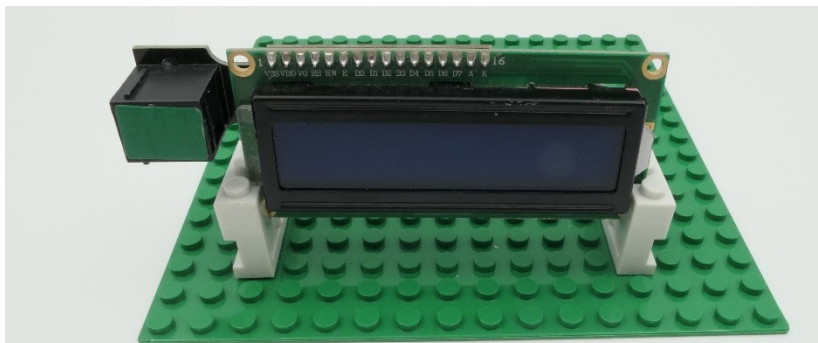
Στερεώστε τη συσκευή στις εγκοπές με το μικρότερο άνοιγμα.



3. Μονάδα Οθόνης LCD – AJX04



Στερεώστε τη συσκευή στις εγκοπές με το μικρότερο άνοιγμα.

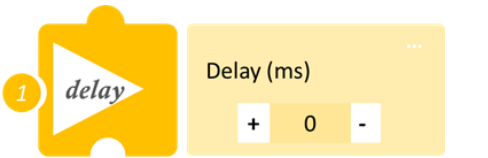
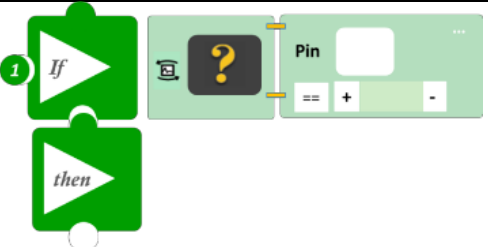


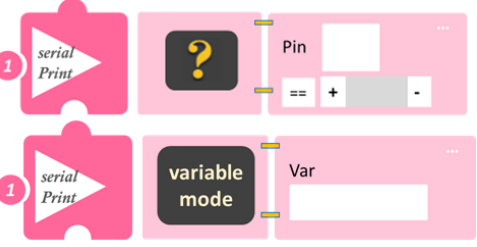
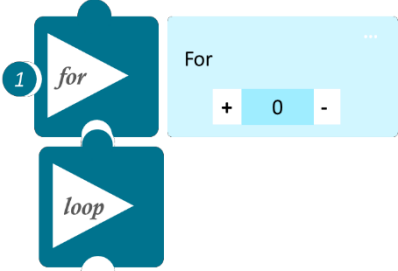
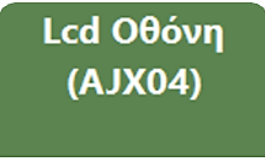
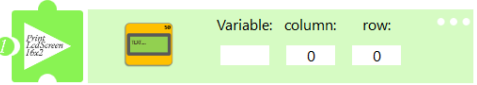
1.3 Εντολές Προγραμματισμού

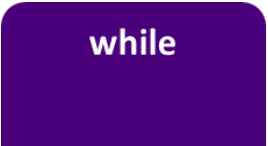
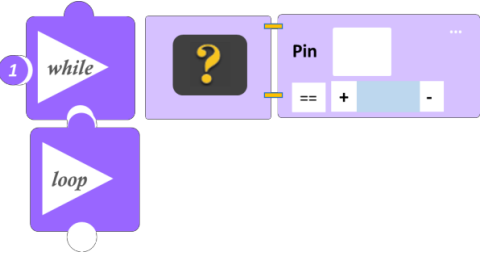
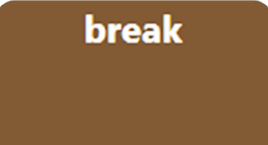
Το μενού επιλογών της εφαρμογής αποτελείται από το:

1. Μενού εντολών λειτουργίας. Την ανάλυση των εντολών του συγκεκριμένου μενού μπορείτε να τη δείτε στις παρακάτω ενότητες.
2. Μενού μαθηματικών εντολών. Την ανάλυση των εντολών του συγκεκριμένου μενού μπορείτε να τη δείτε στις παρακάτω ενότητες.
3. Μενού Εντολών των Ρομπότ R2 και R4. Την ανάλυση των εντολών του συγκεκριμένου μενού στα εγχειρίδια λειτουργίας των ρομπότ.

1.3.1. Μενού Επιλογής Εντολών Λειτουργίας

Εικονίδιο Επιλογής Εντολής	Περιγραφή	Πλακίδιο Εντολής
	Analog Read: Αυτή η εντολή είναι ενσωματωμένη στις εντολές "if", "while" και "variable".	
Analog Write	Analog Write: Αυτή η εντολή προγραμματίζει τις παραμέτρους της αναλογικής συσκευής εξόδου που συνδέετε στον ελεγκτή ARD:icon.	 Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής
	Digital Read: Αυτή η εντολή είναι ενσωματωμένη στις εντολές "if", "while" και "variable".	
Digital Write	Digital Write: Αυτή η εντολή προγραμματίζει τις παραμέτρους της ψηφιακής συσκευής εξόδου που συνδέετε στον ελεγκτή ARD:icon.	 Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής
Delay Time	delay: Επιλέγουμε την παύση του προγράμματος για συγκεκριμένο αριθμό χιλιοστών του δευτερολέπτου.	 Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής
if	if...then: Επιλέγουμε τη συνθήκη if. Αυτή η εντολή προγραμματίζει τις παραμέτρους της συνθήκης. Εάν η συνθήκη είναι αληθής, τότε θα εκτελεστεί η εντολή.	 Δείτε τα 2 σχετικά βίντεο με τις οδηγίες προγραμματισμού της εντολής
	else: Επιλέγουμε την συνθήκη else. Συνδέεται ΠΑΝΤΑ με την δήλωση if. Αν η δήλωση if είναι ψευδής, τότε θα εκτελεστεί η εντολή else.	

	Η εντολή else πάντα συνοδεύεται από τουλάχιστον μία εντολή προγραμματισμού αναλογικής (analog write) ή ψηφιακής εξόδου(digital write)	 <u>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</u>
	Serial print: Επιλέγουμε τις μεταβλητές που θέλουμε να τυπώσει ο ελεγκτής. Μπορεί να είναι οι τιμές κάποιας αναλογικής ή ψηφιακής συσκευής εισόδου ή κάποια συγκεκριμένη φράση ή αριθμός.	 <u>Δείτε τα 2 σχετικά βίντεο με τις οδηγίες προγραμματισμού της εντολής</u>
	for: Επιλέγουμε τη συνθήκη βρόχου “for”, για να προγραμματίσουμε πόσες φορές κάτι πρόκειται να επαναληφθεί για μια δεδομένη συνθήκη if. Η επανάληψη θα εκτελείται, έως ότου η δεδομένη συνθήκη σταματήσει να ισχύει. Η εντολή “for” πάντα συνοδεύεται τουλάχιστον από μία εντολή Προγραμματισμού αναλογικής (analog write) ή ψηφιακής εξόδου(digital write).	 <u>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</u>
	LCD Οθόνη: Επιλέγουμε τις μεταβλητές που θέλουμε να τυπώνονται στη σειρά 1 και 2 της οθόνης LCD.	 <u>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</u>
	RGB LED: Επιλέγουμε το χρώμα που θέλουμε να ανάβει το κάθε LED της συσκευής RGB LED.	 <u>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</u>

Εικονίδιο Επιλογής Εντολής	Περιγραφή	Πλακίδιο Εντολής
	while: Επιλέγουμε τη συνθήκη βρόχου “while”, για να προγραμματίσουμε πόσες φορές κάτι πρόκειται να επαναληφθεί, έως ότου μια δεδομένη συνθήκη σταματήσει να ισχύει και ενώ εκτελείται μια άλλη συνθήκη. Η σύνταξη της εντολής “while” βασίζεται σε μια συνθήκη, παρόμοια με την εντολή «if». Η διαφορά τους είναι ότι η «if» εκτελεί μόνο μια φορά το τμήμα Κώδικα που την αφορά, ενώ η «while» για όσο είναι αληθής η συνθήκη της. Πάντα συνοδεύεται τουλάχιστον από μία εντολή προγραμματισμού αναλογικής (analog write) ή ψηφιακής εξόδου(digital write).	 <i>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</i>
	break: Η εντολή break μας επιτρέπει να εγκαταλείψουμε μια δομή ελέγχου βρόχου (for, while, do) από οποιοδήποτε σημείο της και άσχετα με τη συνθήκη ελέγχου. Μόλις εκτελεστεί η εντολή break ο έλεγχος του προγράμματος περνάει στο τέλος του εσωτέρου βρόχου που την περιέχει.	

1.2.2 Ανάλυση Εντολών Λειτουργίας

1. Η εντολή "digital write":

αναφέρεται σε έναν όρο, που χρησιμοποιείται σε προγραμματιστικά περιβάλλοντα, για τον έλεγχο των ψηφιακών ακροδεκτών/θυρών ενός μικροελεγκτή. Συγκεκριμένα, αναφέρεται σε μια εντολή που γράφει (ή θέτει) μια τιμή σε έναν ψηφιακό ακροδέκτη/θύρα.

Οι ψηφιακοί ακροδέκτες μπορούν να έχουν δύο πιθανές καταστάσεις: "HIGH" (υψηλό) και "LOW" (χαμηλό). Η εντολή "digital write", επιτρέπει στον προγραμματιστή να ορίσει την κατάσταση του ψηφιακού ακροδέκτη σε ένα από αυτά τα δύο σήματα.

Όταν προγραμματίζουμε μια ψηφιακή συσκευή, χρησιμοποιώντας την εντολή "digital write", συνήθως αναφερόμαστε στον έλεγχο ενός ψηφιακού ακροδέκτη για την ενεργοποίηση ή απενεργοποίηση μιας συσκευής ή ενός περιφερειακού, όπως ένα LED ή ένας κινητήρας.

2. Η εντολή "analog write":

χρησιμοποιείται συνήθως σε πλατφόρμες προγραμματισμού, όπως η Arduino, για να ελέγξουμε την ένταση ενός αναλογικού ακροδέκτη/θύρα. Αν και η ονομασία "analog write" μπορεί να είναι παραπλανητική, δεν πρόκειται για πραγματική αναλογική έξοδο, αλλά για μια μορφή παλμοταχύτητας (pulse-width modulation - PWM).

Η τεχνική της PWM σας επιτρέπει να προσομοιώσετε μια αναλογική έξοδο, ρυθμίζοντας την αναλογία μεταξύ της περιόδου, που η έξοδος είναι σε υψηλή κατάσταση (HIGH) και της περιόδου, που είναι σε χαμηλή κατάσταση (LOW). Αυτή η αλλαγή της αναλογίας επιτρέπει τον έλεγχο της φωτεινότητας LED, της ταχύτητας κινητήρων ή άλλων περιφερειακών, που ανταποκρίνονται σε αναλογικές τιμές.

3. Η εντολή "if" (ή "if statement"):

αναφέρεται σε έναν έλεγχο υπό συνθήκη, σε ένα πρόγραμμα. Η εντολή "if", χρησιμοποιείται για να εκτελέσει μια συγκεκριμένη ενέργεια ή ένα κομμάτι κώδικα, αν η συνθήκη που δίνεται είναι αληθής (true).

Η γενική μορφή της εντολής "if" είναι η εξής:

```
if (συνθήκη) {  
    // Κώδικας που εκτελείται, αν η συνθήκη είναι αληθής  
}
```

Η "συνθήκη", είναι μια λογική έκφραση, που αξιολογείται ως αληθής ή ψευδής. Αν η συνθήκη είναι αληθής, τότε ο κώδικας μέσα στις αγκύλες {} εκτελείται. Αν η συνθήκη είναι ψευδής, τότε ο κώδικας μέσα στις αγκύλες παραλείπεται και η εκτέλεση συνεχίζεται μετά το τμήμα "if".

4. Η εντολή "else":

χρησιμοποιείται σε συνδυασμό με την εντολή "if", σε ένα πρόγραμμα για την εκτέλεση ενός διαφορετικού κομματιού κώδικα, όταν η συνθήκη της εντολής "if" είναι ψευδής (false).

Αφού εκτελεστεί ο κώδικας, που βρίσκεται μέσα στο τμήμα "if", όταν η συνθήκη είναι αληθής, μπορείτε να προσθέσετε το τμήμα "else", για να καθορίσετε τι θα γίνει, αν η συνθήκη είναι ψευδής. Ο κώδικας μέσα στο τμήμα "else" θα εκτελεστεί, αν η συνθήκη της εντολής "if" δεν είναι αληθής.

Η γενική μορφή της εντολής "if-else" είναι η εξής:

```
if (συνθήκη) {  
    // Κώδικας που εκτελείται, αν η συνθήκη είναι αληθής
```

```

} else {
    // Κώδικας που εκτελείται, αν η συνθήκη είναι ψευδής
}

```

Αν η συνθήκη της εντολής "if" είναι αληθής, τότε ο κώδικας μέσα στο πρώτο μπλοκ εκτελείται. Αν η συνθήκη είναι ψευδής, τότε ο κώδικας μέσα στο μπλοκ "else" εκτελείται αντί αυτού.

5. Η εντολή "for":

χρησιμοποιείται σε πολλές γλώσσες προγραμματισμού, για την επανάληψη ενός τμήματος κώδικα για έναν συγκεκριμένο αριθμό επαναλήψεων.

Η γενική μορφή της εντολής "for" είναι η εξής:

```

for (αρχικοποίηση; συνθήκη; ενημέρωση) {
    // Κώδικας που εκτελείται κατά τη διάρκεια της επανάληψης
}

```

Στην αρχή της εντολής "for", το τμήμα "αρχικοποίηση" χρησιμοποιείται για να αρχικοποιήσετε μια μεταβλητή ελέγχου ή να εκτελέσετε οποιεσδήποτε άλλες εντολές, που χρειάζονται πριν από την έναρξη της επανάληψης.

Η "συνθήκη", είναι μια λογική έκφραση που ελέγχεται σε κάθε επανάληψη. Αν η συνθήκη είναι αληθής, ο κώδικας μέσα στις αγκύλες εκτελείται. Αν η συνθήκη είναι ψευδής, η εκτέλεση της επανάληψης σταματά και η εκτέλεση του προγράμματος συνεχίζεται από το επόμενο σημείο μετά την εντολή "for".

Η "ενημέρωση", είναι μια εντολή, που εκτελείται μετά από κάθε επανάληψη και χρησιμοποιείται για να αλλάξετε την κατάσταση της μεταβλητής ελέγχου ή να εκτελέσετε άλλες ενέργειες, που απαιτούνται στο τέλος κάθε επανάληψης.

Ας δούμε ένα παράδειγμα: Έστω, ότι θέλουμε να εκτυπώσουμε τους αριθμούς από 1 έως 5. Μπορούμε να το πετύχουμε με την εξής χρήση της εντολής "for" σε γλώσσα όπως η C++:

```

for (int i = 1; i <= 5; i++) {
    cout << i << endl;
}

```

Σε αυτό το παράδειγμα, η μεταβλητή "i" αρχικοποιείται στην τιμή 1. Η συνθήκη ελέγχει αν η "i" είναι μικρότερη ή ίση με 5. Καθώς αυξάνουμε την "i" κατά 1, μετά από κάθε επανάληψη (i++), ο κώδικας μέσα στις αγκύλες εκτελείται και εκτυπώνει την τρέχουσα τιμή της "i". Η διαδικασία επαναλαμβάνεται, μέχρι η συνθήκη να γίνει ψευδής, δηλαδή όταν η "i" γίνει μεγαλύτερη από 5.

6. Η εντολή "while":

χρησιμοποιείται σε πολλές γλώσσες προγραμματισμού, για την επανάληψη ενός τμήματος κώδικα, μέχρις ότου μια συνθήκη γίνει ψευδής (false). Η συνθήκη ελέγχεται στην αρχή κάθε επανάληψης και αν είναι αληθής (true), ο κώδικας εντός του "while" εκτελείται. Η εκτέλεση συνεχίζεται, μέχρι η συνθήκη να γίνει ψευδής, οπότε η εκτέλεση συνεχίζεται από το επόμενο σημείο μετά το τμήμα "while".

Η γενική μορφή της εντολής "while" είναι η εξής:

```

while (συνθήκη) {
    // Κώδικας που εκτελείται κατά τη διάρκεια της επανάληψης
}

```

Ο κώδικας μέσα στο τμήμα "while" εκτελείται, μόνο όσο η συνθήκη είναι αληθής. Αν η συνθήκη είναι αρχικά ψευδής, ο κώδικας μέσα στο τμήμα "while" δεν εκτελείται καθόλου.

Ας δούμε ένα παράδειγμα: Έστω, ότι θέλουμε να εκτυπώσουμε τους αριθμούς από 1 έως 5 χρησιμοποιώντας την εντολή "while" σε γλώσσα όπως η C++:

```
int i = 1;
while (i <= 5) {
    cout << i << endl;
    i++;
}
```

Σε αυτό το παράδειγμα, ξεκινούμε με την αρχική τιμή της μεταβλητής "i" να είναι 1. Η συνθήκη ελέγχει αν η "i" είναι μικρότερη ή ίση του 5. Όσο η συνθήκη είναι αληθής, ο κώδικας μέσα στο "while" εκτελείται, εκτυπώνοντας την τρέχουσα τιμή της "i". Στο τέλος κάθε επανάληψης, αυξάνουμε την "i" κατά 1 (i++). Η διαδικασία συνεχίζεται, μέχρι η συνθήκη να γίνει ψευδής, δηλαδή όταν η "i" γίνει μεγαλύτερη από 5.

7. Η εντολή "delay":

χρησιμοποιείται συνήθως σε προγραμματισμό μικροελεγκτών ή αρχιτεκτονικών, που υποστηρίζουν εικονικό χρονισμό. Ο σκοπός της είναι να προκαλέσει μια παύση ή καθυστέρηση στην εκτέλεση του προγράμματος για ένα συγκεκριμένο χρονικό διάστημα.

Συνήθως, η εντολή "delay", παίρνει ως όρισμα έναν ακέραιο αριθμό, που αναπαριστά το χρονικό διάστημα σε μιλιδευτερόλεπτα (ms) ή μικροδευτερόλεπτα (μs). Κατά τη διάρκεια της καθυστέρησης, η εκτέλεση του προγράμματος παύει, και η συσκευή αναμένει το πέρας της περιόδου καθυστέρησης, προτού συνεχίσει την εκτέλεση του επόμενου κώδικα.

Η εντολή "delay" είναι χρήσιμη σε πολλές περιπτώσεις, όπως για την προσθήκη χρονικών καθυστερήσεων μεταξύ των ενεργειών, τον συγχρονισμό των ενεργειών ή απλά για να προσομοιώσετε χρονικές καθυστερήσεις.

Εδώ είναι ένα παράδειγμα χρήσης της εντολής "delay" στην Arduino C++:

```
void setup() {
    // Αρχικές ρυθμίσεις
}

void loop() {
    // Κώδικας επανάληψης

    // Καθυστέρηση για 1 δευτερόλεπτο (1000 ms)
    delay(1000);
}
```

Σε αυτό το παράδειγμα, ο κώδικας μέσα στη συνάρτηση loop() θα επαναλαμβάνεται συνεχώς. Αμέσως μετά την εκτέλεση του κώδικα, η εντολή delay(1000) προκαλεί μια παύση για 1 δευτερόλεπτο, πριν συνεχίσει την επόμενη επανάληψη. Αυτό δημιουργεί ένα διάστημα ενός δευτερολέπτου μεταξύ των εκτελέσεων του κώδικα μέσα στον βρόχο.

8. Η εντολή "Serial print":

χρησιμοποιείται συνήθως σε γλώσσες προγραμματισμού, για την αποστολή δεδομένων από έναν μικροελεγκτή σε έναν υπολογιστή ή άλλη συσκευή, μέσω της σειριακής θύρας.

Η συνάρτηση "Serial.print()" παίρνει ως **αναφορά** μια τιμή ή μια μεταβλητή και την αποστέλλει στη σειριακή θύρα. Ανάλογα με τον τύπο της τιμής ή της μεταβλητής, η "Serial.print()" μπορεί να εκτυπώσει ακέραιους αριθμούς, δεκαδικούς αριθμούς, χαρακτήρες, συμβολοσειρές και άλλους τύπους δεδομένων.

Ας δούμε ένα παράδειγμα χρήσης της εντολής "Serial.print()" στην Arduino C++:

```
int sensorValue = 500;
float temperature = 25.5;

void setup() {
  Serial.begin(9600); // Αρχικοποίηση της σειριακής θύρας
}

void loop() {
  // Αποστολή δεδομένων στη σειριακή θύρα
  Serial.print("Sensor value: ");
  Serial.print(sensorValue);
  Serial.print(", Temperature: ");
  Serial.print(temperature);
  Serial.println(" °C");

  delay(1000);
}
```

Σε αυτό το παράδειγμα, οι τιμές του αισθητήρα (sensorValue) και η θερμοκρασία (temperature) εκτυπώνονται μέσω της σειριακής θύρας με τη χρήση της "Serial.print()". Ο κώδικας του παραδείγματος εκτυπώνει ένα μήνυμα, που περιλαμβάνει τις τιμές των μεταβλητών, όπως "Sensor value: 500, Temperature: 25.5 °C". Η εντολή "Serial.println()" προσθέτει έναν χαρακτήρα αλλαγής γραμμής ('\n'), μετά την εκτύπωση του μηνύματος.

Ανοίγοντας τον παράλληλο παρατηρητή (serial monitor) στο περιβάλλον ανάπτυξης της Arduino, μπορείτε να δείτε τα μηνύματα που αποστέλλονται μέσω της σειριακής θύρας.

9. Η εντολή "**digital Read**":

χρησιμοποιείται συνήθως σε γλώσσες προγραμματισμού, όπως η Arduino C++, για να διαβάσει την κατάσταση μιας ψηφιακής εισόδου. Μια ψηφιακή είσοδος μπορεί να είναι σε δύο καταστάσεις: "HIGH" (υψηλό) ή "LOW" (χαμηλό), αντιπροσωπευόμενες από τις τιμές 1 και 0, αντίστοιχα.

Η εντολή "digitalRead" παίρνει ως όρισμα τον αριθμό του ακροδέκτη (pin), στον οποίο είναι συνδεδεμένη η ψηφιακή είσοδος και επιστρέφει την κατάσταση της εισόδου, ως αποτέλεσμα. Η επιστρεφόμενη τιμή είναι είτε "HIGH" (1), εάν η είσοδος είναι σε υψηλή κατάσταση, είτε "LOW" (0), εάν η είσοδος είναι σε χαμηλή κατάσταση.

Ας δούμε ένα παράδειγμα χρήσης της εντολής "digitalRead" στην Arduino C++:

```
int buttonPin = 2;
int ledPin = 13;
```

```

void setup() {
  pinMode(buttonPin, INPUT);
  pinMode(ledPin, OUTPUT);
}

void loop() {
  int buttonState = digitalRead(buttonPin);

  if (buttonState == HIGH) {
    digitalWrite(ledPin, HIGH);
  } else {
    digitalWrite(ledPin, LOW);
  }
}

```

Σε αυτό το παράδειγμα, υπάρχει ένας διακόπτης, που είναι συνδεδεμένος στον ακροδέκτη 2 (buttonPin), και ένα LED, που είναι συνδεδεμένο στον ακροδέκτη 13 (ledPin). Η εντολή "pinMode" ρυθμίζει τον ακροδέκτη του διακόπτη, ως είσοδο, και τον ακροδέκτη του LED, ως έξοδο.

Στη συνέχεια, η εντολή "digitalRead(buttonPin)" διαβάζει την κατάσταση του διακόπτη και αποθηκεύει την τιμή στη μεταβλητή "buttonState". Αν η κατάσταση του διακόπτη είναι "HIGH", τότε η εντολή "digitalWrite(ledPin, HIGH)" ενεργοποιεί το LED, ενώ αν η κατάσταση του διακόπτη είναι "LOW", τότε η εντολή "digitalWrite(ledPin, LOW)" απενεργοποιεί το LED.

Έτσι, το LED θα ανάβει, όταν ο διακόπτης είναι σε κατάσταση HIGH/ON και θα σβήσει, όταν ο διακόπτης είναι σε κατάσταση LOW/OFF.

10. Η εντολή "*analog Read*":

χρησιμοποιείται συνήθως σε γλώσσες προγραμματισμού όπως η Arduino C++ για την ανάγνωση τιμών από αναλογικές εισόδους. Μια αναλογική είσοδος παρέχει συνεχείς τιμές σήματος, σε αντίθεση με τις διακριτές τιμές "HIGH" και "LOW", που παρέχονται από τις ψηφιακές εισόδους.

Για να χρησιμοποιήσετε την εντολή "analogRead", θα πρέπει να συνδέσετε την αναλογική είσοδο σας σε έναν αναλογικό ακροδέκτη (pin) στον μικροελεγκτή (όπως η Arduino). Η εντολή "analogRead" διαβάζει την τάση που παρέχεται από την αναλογική είσοδο και επιστρέφει μια τιμή από το 0 έως το 1023, αντιπροσωπεύοντας την τάση ανάμεσα στη γείωση (0V) και την αναφορά τάσης (συνήθως 5V).

Ας δούμε ένα παράδειγμα χρήσης της εντολής "analogRead" στην Arduino C++:

```

int analogPin = A0;

void setup() {
  Serial.begin(9600);
}

void loop() {
  int sensorValue = analogRead(analogPin);
  float voltage = sensorValue * (5.0 / 1023.0);

  Serial.print("Sensor value: ");

```

```

Serial.print(sensorValue);
Serial.print(", Voltage: ");
Serial.print(voltage);
Serial.println(" V");
delay(1000);
}

```

Σε αυτό το παράδειγμα, η αναλογική είσοδος είναι συνδεδεμένη στον αναλογικό ακροδέκτη A0. Η εντολή "Serial.begin(9600)" αρχικοποιεί τη σειριακή επικοινωνία με μια ταχύτητα 9600 baud.

Στη συνέχεια, η εντολή "analogRead(analogPin)" διαβάζει την τιμή από την αναλογική είσοδο και αποθηκεύει την τιμή στη μεταβλητή "sensorValue". Η μεταβλητή "sensorValue" αναπαριστά την αναλογική τιμή, σε μορφή ακέραιου αριθμού από 0 έως 1023.

Στη συνέχεια, η τάση υπολογίζεται με βάση την τιμή του "sensorValue" και αποθηκεύεται στη μεταβλητή "voltage". Τέλος, τα μηνύματα εκτυπώνονται μέσω της σειριακής θύρας, με τη χρήση της εντολής "Serial.print" και "Serial.println".

Συνήθως, η τιμή που διαβάζεται από μια αναλογική είσοδο μετατρέπεται σε κάποια φυσική μεγέθυνση (όπως θερμοκρασία ή φωτεινότητα), χρησιμοποιώντας κατάλληλους μετασχηματισμούς και μετρήσεις.

11. Η εντολή "break":

χρησιμοποιείται για να διακόπτει την εκτέλεση ενός βρόχου (loop) ή ενός switch statement (εναλλαγή). Η εντολή break είναι ένας τύπος ελέγχου ροής (flow control), που επιτρέπει στο πρόγραμμά σας να βγει από τον τρέχοντα βρόχο ή να τερματίσει την εκτέλεση ενός switch statement.

Συνοπτικά, χρησιμοποιείται με την εξής σύνταξη:

```

c
break;

```

Ένα παράδειγμα χρήσης της εντολής break περιλαμβάνει:

Βρόχοι (Loops):

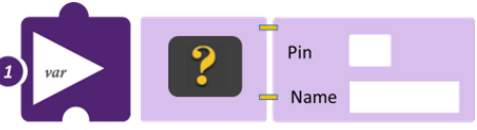
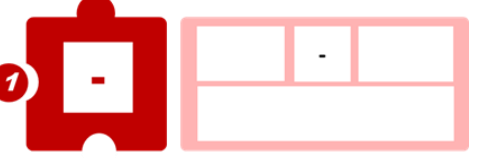
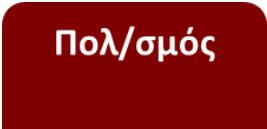
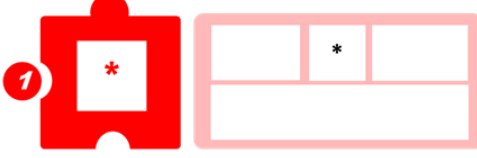
```

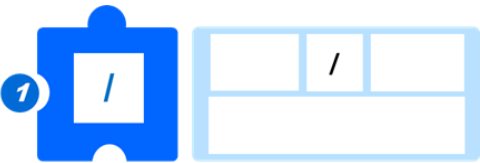
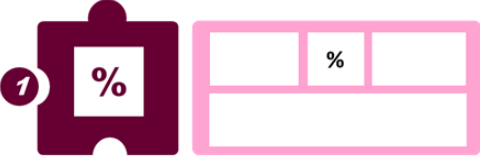
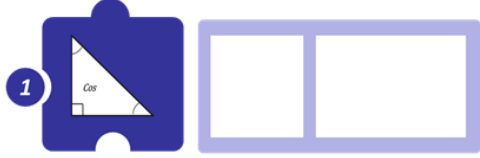
c
for (int i = 1; i <= 10; i++) {
    if (i == 5) {
        break; // Τερματίζει τον βρόχο όταν i γίνει 5
    }
    printf("%d\n", i);
}

```

Σε αυτό το παράδειγμα, ο βρόχος for θα τρέξει μέχρι το i να γίνει 5, και τότε η εντολή break θα τερματίσει τον βρόχο.

1.3.3. Μενού Επιλογής Βασικών Μαθηματικών Εντολών

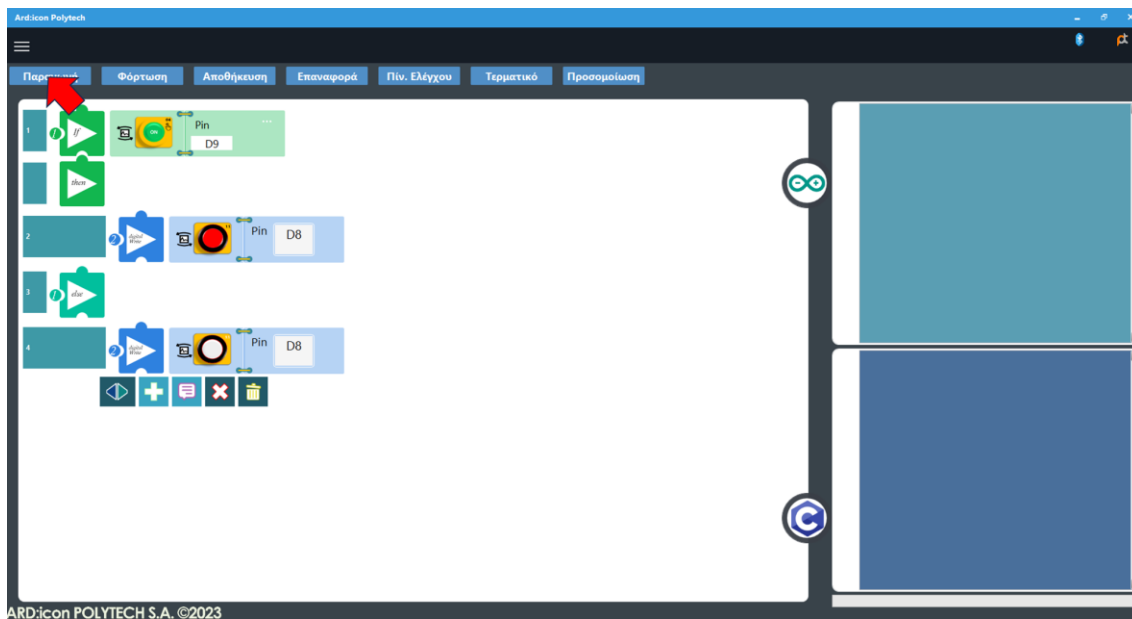
Εικονίδιο Επιλογής Εντολής	Περιγραφή	Πλακίδιο Εντολής
	<u>const (Σταθερά/constant):</u> Αυτή η εντολή προσδιορίζει μια μεταβλητή με σταθερή τιμή, η οποία δεν μπορεί να αλλάξει κατά τη διάρκεια εκτέλεσης του προγράμματος.	 <i>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</i>
	<u>Var (Μεταβλητή):</u> Αυτή η εντολή προσδιορίζει μια μεταβλητή που μπορεί να πάρει διαφορετικές τιμές. Οι τιμές της μεταβλητής Var μπορούν να αλλάζουν και κατά τη ροή του προγράμματος.	 <i>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</i>
	<u>Πρόσθεση:</u> Αυτή η εντολή + (συν), καθορίζει δύο τελεστές για την παραγωγή ενός αθροίσματος.	 <i>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</i>
	<u>Αφαίρεση:</u> Αυτή η εντολή - (μείον), καθορίζει δύο τελεστές για τη παραγωγή της διαφοράς του δεύτερου από τον πρώτο.	 <i>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</i>
	<u>Πολλαπλασιασμός:</u> Αυτή η εντολή * (επί), καθορίζει δύο τελεστές για την παραγωγή του μεταξύ τους γινομένου.	 <i>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</i>

Διαίρεση	Διαίρεση: Αυτή η εντολή / (δια), καθορίζει δύο τελεστές για τη παραγωγή του μεταξύ τους αποτελέσματος διαίρεσης.	 <i>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</i>
Υπόλοιπο Διάρθρωσης	Υπόλοιπο Διάρθρωσης: Αυτή η εντολή καθορίζει δύο τελεστές για τη παραγωγή του υπολοίπου της μεταξύ τους διαίρεσης.	 <i>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</i>
Δύναμη	Δύναμη (pow): Αυτή η εντολή υπολογίζει την τιμή ενός αριθμού υψωμένου σε κάποια δύναμη. Μπορεί να χρησιμοποιηθεί για την αύξηση ενός αριθμού σε κλασματική ισχύ. Αυτό είναι χρήσιμο για τη δημιουργία εκθετικής χαρτογράφησης τιμών ή καμπυλών.	 <i>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</i>
Τετραγωνική Ρίζα	Τετραγωνική ρίζα (sqrt): Αυτή η εντολή υπολογίζει την τετραγωνική ρίζα ενός αριθμού.	
Τυχαίος Αριθμός	Τυχαίος Αριθμός (random): Αυτή η εντολή δημιουργεί ψευδοτυχαίους αριθμούς, από ένα εύρος τιμών που έχουμε ήδη προκαθορίσει.	
Συνημίτονο	Συνημίτονο(con): Η εντολή αυτή υπολογίζει το συνημίτονο μια γωνίας, που ορίζουμε στο πρόγραμμα.	 <i>Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής</i>

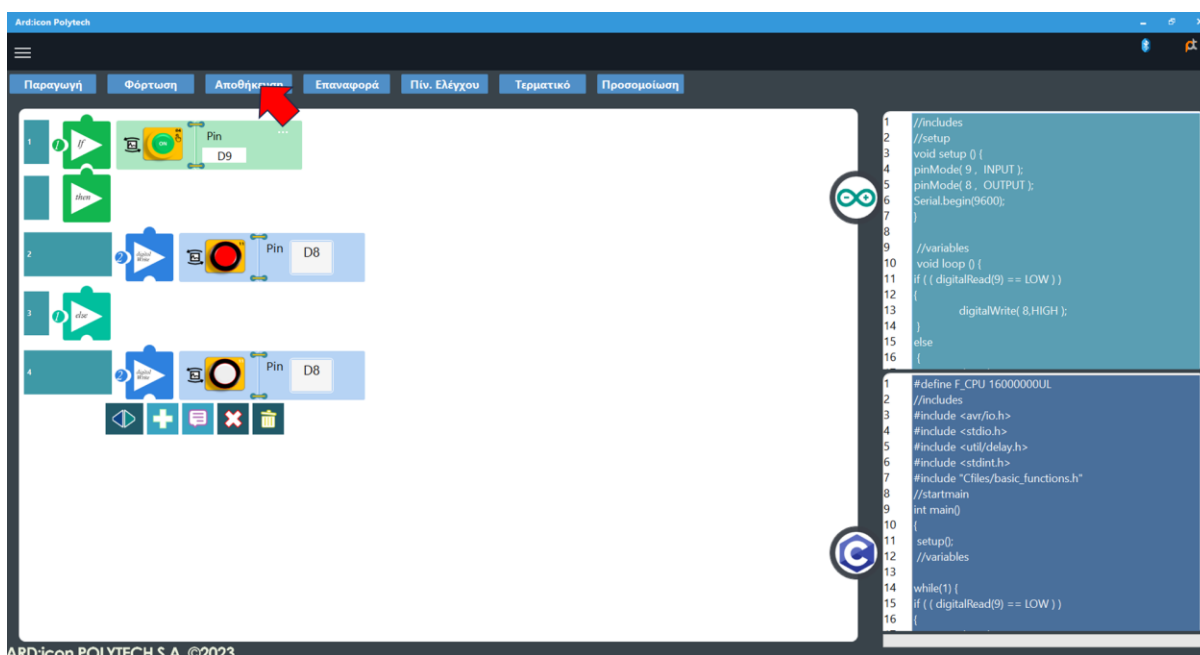
Ημίτονο	<u>Ημίτονο (sin):</u> Η εντολή αυτή υπολογίζει το ημίτονο μια γωνίας, που ορίζουμε στο πρόγραμμα.	 Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής
Εφαπτομένη	<u>Εφαπτομένη (tan):</u> Η εντολή αυτή υπολογίζει την εφαπτομένη μια γωνίας που ορίζουμε στο πρόγραμμα..	 Δείτε το σχετικό βίντεο με τις οδηγίες προγραμματισμού της εντολής

1.3.4 Αποθήκευση και Επαναφορά Ρουτίνας

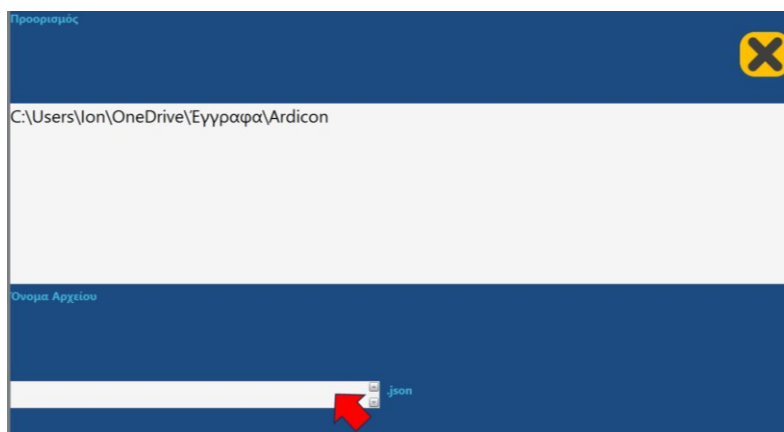
1. Δημιουργούμε τον κώδικα ανάλογα με το κύκλωμα που θέλουμε να προγραμματίσουμε και έπειτα επιλέγουμε «Παραγωγή».



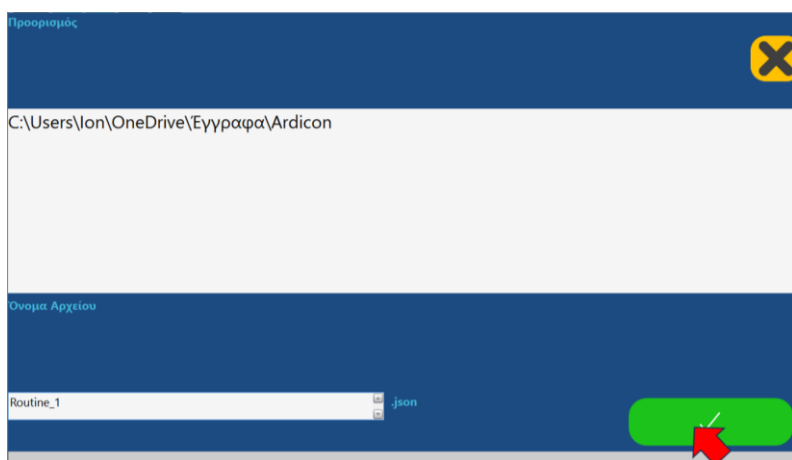
2. Αφού έχει δημιουργηθεί ο κώδικας επιλέγουμε «Αποθήκευση».



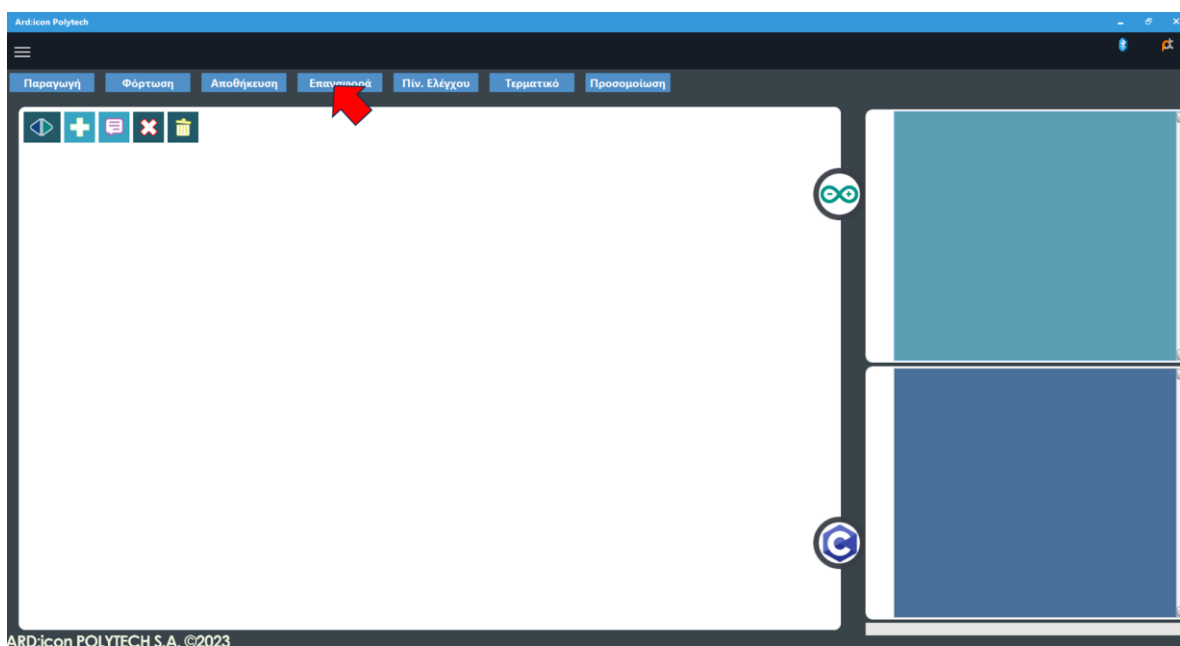
3. Στο αναδυόμενο παράθυρο, στο κελί «Όνομα αρχείου» γράψτε την ονομασία της ρουτίνας που προγραμματίσατε, πχ ROUTINA_1 (χρησιμοποιήστε μόνο λατινικούς χαρακτήρες και το ειδικό σύμβολο _)



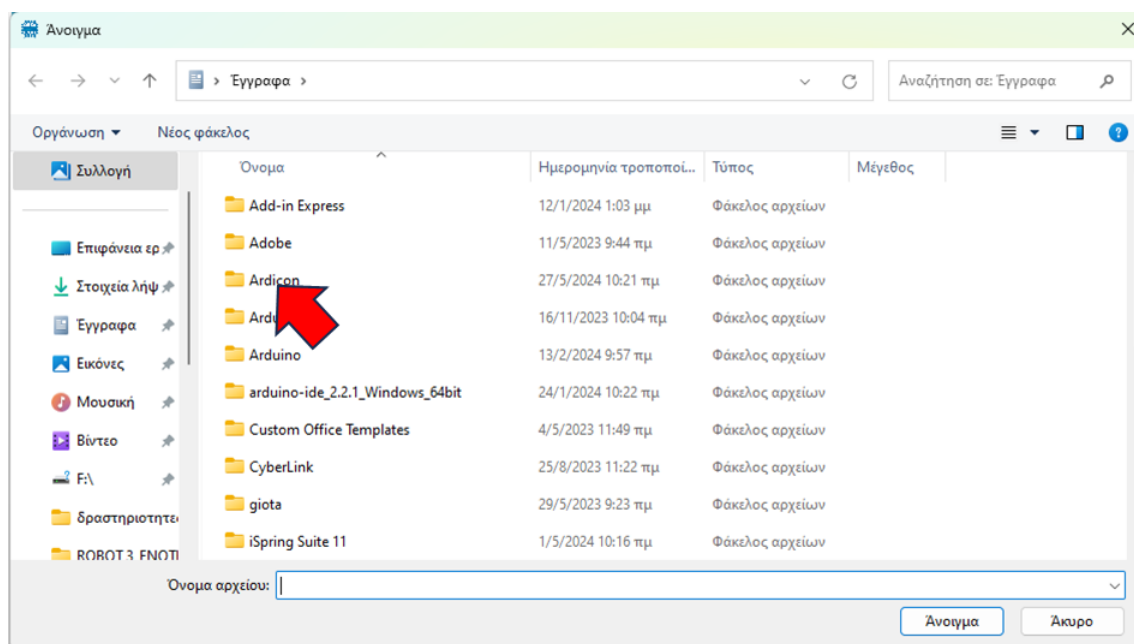
4. Επιλέξτε οκ .



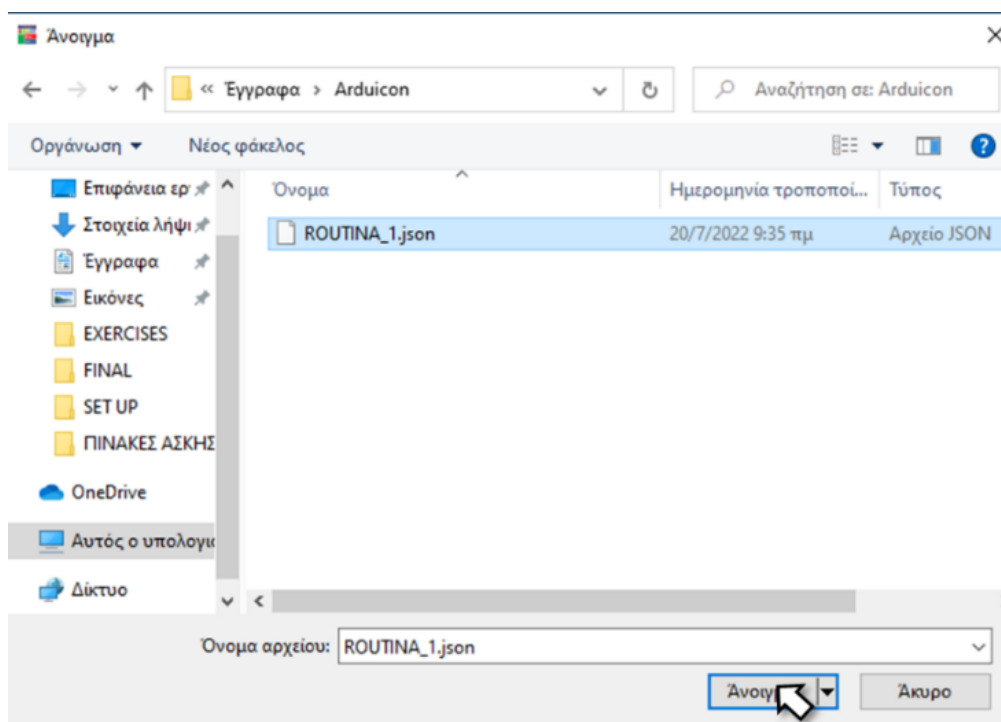
5. Αν θέλετε να επαναφέρετε τη Ρουτίνα που προγραμματίσατε, τότε στην οθόνη εντολών της εφαρμογής πρέπει να επιλέξετε «Επαναφορά».



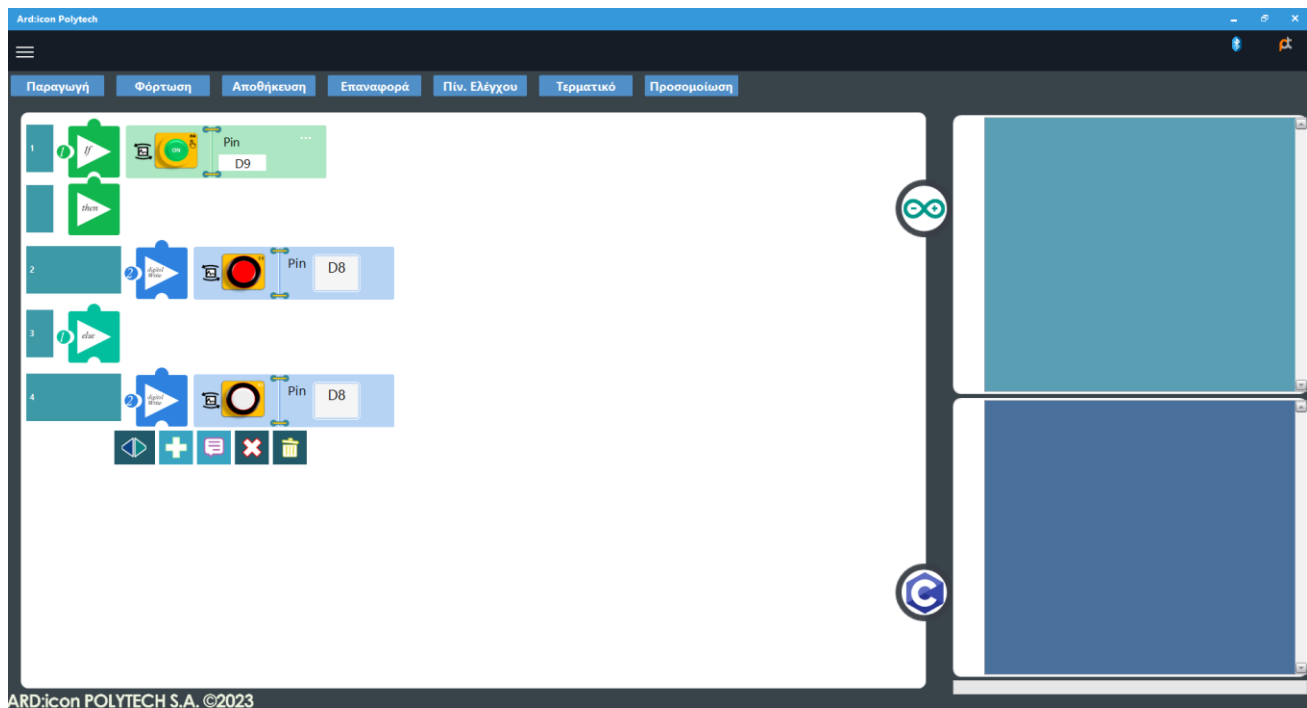
6. Στο αναδυόμενο παράθυρο, ανοίξτε τον φάκελο “**Ardicon**”.



7. Στο αναδυόμενο παράθυρο, επιλέξτε τη Ρουτίνα που θέλετε π.χ. ROUTINA_1 και επιλέξτε **Ανοιγμα**.



Τα πλακίδια προγραμματισμού της ρουτίνας θα εμφανιστεί στην οθόνη προγραμματισμού της εφαρμογής ARD;icon.



1.4 Εντολές Προγραμματισμού Ρομπότ R4

1.4.1 Φόρτωση προγράμματος λειτουργίας Bluetooth για προγραμματισμό με ARD:icon

Πριν ξεκινήσετε να προγραμματίζετε το ρομπότ με την εφαρμογή ARD:icon πρέπει πρώτα να φορτώσετε στο ρομπότ το πρόγραμμα λειτουργίας του bluetooth. Συνδέστε το ρομπότ στον υπολογιστή. Τρέξτε την εφαρμογή Ardicon IDE, αντιγράψτε και επικολλήστε τον παρακάτω κώδικα.

Κώδικας δοκιμής:

```
/*
  Λειτουργία Έξυπνου αυτοκίνητου με την εφαρμογή ARDicon.
  Έξυπνο αυτοκίνητο ελέγχου ρομποτικού βραχίονα.
*/
#include <Wire.h>
#include <Adafruit_PWMServoDriver.h>
Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver(0x47);
#include <Servo.h>
//#include <IRremote.h>
//int RECV_PIN = A0;
int echoPin = 13; // Ορισμός μονάδα υπερήχων ECHO στο D13
int trigPin = 12; // Ορισμός μονάδα υπερήχων TRIG στο D12

#define IR_Go 70
#define IR_Back 21
#define IR_Left 68
#define IR_Right 67
#define IR_Stop 64

#define SensorLeft 6 // καρφίτσα εισαγωγής του αριστερού αισθητήρα
#define SensorMiddle 7 // καρφίτσα εισαγωγής του μεσαίου αισθητήρα
#define SensorRight 8 // καρφίτσα εισαγωγής του δεξιού αισθητήρα
unsigned char SL; // κατάσταση αριστερού αισθητήρα
unsigned char SM; // κατάσταση μεσαίου αισθητήρα
unsigned char SR; // κατάσταση δεξιού αισθητήρα

Servo myservo1;
Servo myservo2;
Servo myservo3;

int k1 = 80, k2 = 120, k3 = 90; // αρχικοποίηση της τιμής γωνίας των σερβομηχανισμών
int M1[20], M2[20], M3[20];
int i = 0, j = 0, t = 0, speed = 1500, turnSpeed = 2000;
char blue_val;

void setup() {
  Serial.begin(9600); //set baud rate to 9600
  pwm.begin();
```

```

    pwm.setPWMFreq(60);
    myservo1.attach(11);
    myservo2.attach(10);
    myservo3.attach(9);
    myservo1.write(k1);
    delay(1000);
    myservo2.write(k2);
    delay(1000);
    myservo3.write(k3);
    delay(1000);
    pinMode(SensorLeft, INPUT);
    pinMode(SensorMiddle, INPUT);
    pinMode(SensorRight, INPUT);
    stop();
    //IrReceiver.begin(RECV_PIN); // ενεργοποίηση του δέκτη IR
}

void loop() {
    if (Serial.available() > 0) { // Λήψη σημάτων Bluetooth
        blue_val = Serial.read();
        Serial.println(blue_val);
        switch (blue_val) {
            case 'F': advance(); break; // Αν λάβει εντολή 'F', κάνει κίνηση μπροστά

            case 'B': back(); break; // Αν λάβει εντολή 'B', κάνει κίνηση πίσω

            case 'L': turnL(); break; // Αν λάβει εντολή 'L', στροφή αριστερά

            case 'R': turnR(); break; // Αν λάβει εντολή 'R', στροφή δεξιά

            case 'S': stop(); break; // Αν λάβει εντολή 'S', σταματάει

            case 'a': speeds_a(); break; // Αν λάβει εντολή 'a', επιταχύνει

            case 'd': speeds_d(); break; // Αν λάβει εντολή 'd', επιβραδύνει

            case 'f': servo2up(); break; // // O βραχίονας ανυψώνεται

            case 'b': servo2down(); break; // // O βραχίονας χαμηλώνει

            case 'l': servo3left(); break; // O βραχίονας στρέφεται προς τα αριστερά

            case 'c': servo3center(); break; // O βραχίονας παίρνει κεντρική θέση

            case 'r': servo3right(); break; // O βραχίονας στρέφεται προς τα δεξιά

            case 'Q': servo1on(); break; // Η δαγκάνα ανοίγει

            case 'E': servo1off(); break; // Η δαγκάνα κλείνει
        }
    }
}

```

```

    case 't': read_servo(); break; // Αν λάβει εντολή 't', καταγράφει κίνηση

    case 'i': do_servo(); break; // Αν λάβει εντολή 'i', εκτελεί κίνηση

    case 'Y': avoid(); break; // Αν λάβει εντολή 'Y' είσοδο στη λειτουργία αποφυγής
εμποδίων

    case 'X': tracking(); break; // Αν λάβει εντολή 'X' είσοδος σε λειτουργία
παρακολούθησης γραμμής

    case 'U': follow(); break; // Αν λάβει εντολή 'U' είσοδος στη λειτουργία
παρακολούθησης υπερήχων
    case 'G': Fall(); break; // Αν λάβει εντολή 'G' εισέλθει σε κατάσταση αντι-πτώσης
    default: break;
}
}
}

// Μέθοδοι κίνησης//

void advance() // Κίνηση μπροστά
{
    pwm.setPWM(0, 0, speed);
    pwm.setPWM(1, 0, 0);
    pwm.setPWM(2, 0, speed);
    pwm.setPWM(3, 0, 0);
    pwm.setPWM(4, 0, speed);
    pwm.setPWM(5, 0, 0);
    pwm.setPWM(6, 0, speed);
    pwm.setPWM(7, 0, 0);
}

void turnR() // Στροφή δεξιά
{
    pwm.setPWM(0, 0, turnSpeed);
    pwm.setPWM(1, 0, 0);
    pwm.setPWM(2, 0, turnSpeed);
    pwm.setPWM(3, 0, 0);
    pwm.setPWM(4, 0, 0);
    pwm.setPWM(5, 0, turnSpeed);
    pwm.setPWM(6, 0, 0);
    pwm.setPWM(7, 0, turnSpeed);
}

void turnL() // Στροφή αριστερά
{
    pwm.setPWM(0, 0, 0);
    pwm.setPWM(1, 0, turnSpeed);
    pwm.setPWM(2, 0, 0);
    pwm.setPWM(3, 0, turnSpeed);

```

```

    pwm.setPWM(4, 0, turnSpeed);
    pwm.setPWM(5, 0, 0);
    pwm.setPWM(6, 0, turnSpeed);
    pwm.setPWM(7, 0, 0);
}

void stop() // Σταματάει
{
    pwm.setPWM(0, 0, 0);
    pwm.setPWM(1, 0, 0);
    pwm.setPWM(2, 0, 0);
    pwm.setPWM(3, 0, 0);
    pwm.setPWM(4, 0, 0);
    pwm.setPWM(5, 0, 0);
    pwm.setPWM(6, 0, 0);
    pwm.setPWM(7, 0, 0);
}

void back() // Κίνηση προς τα πίσω
{
    pwm.setPWM(0, 0, 0);
    pwm.setPWM(1, 0, speed);
    pwm.setPWM(2, 0, 0);
    pwm.setPWM(3, 0, speed);
    pwm.setPWM(4, 0, 0);
    pwm.setPWM(5, 0, speed);
    pwm.setPWM(6, 0, 0);
    pwm.setPWM(7, 0, speed);
}

//Μεθοδοι ελέγχου ταχύτητας//

//Επιτάχυνση
void speeds_a() {
    speed=1500;
}

//Επιβράδυνση
void speeds_d() {
    speed=500;
}

//Μέθοδοι ελέγχου βραχίωνα//
void servo1on() { // Η δαγκάνα ανοίγει
    while (k1 != 20) {
        k1 -= 1;
        myservo1.write(k1);
        delay(5);
    }
}
}

```

```

void servo1off() { // Η δαγκάνα κλείνει
    bool servo1off_flag = 1;
    while (k1 != 80) {
        k1 += 1; // έλεγχος της ακρίβειας περιστροφής
        myservo1.write(k1);
        delay(5); // έλεγχος της ταχύτητας περιστροφής του σερβο
    }
}

void servo2up() { // ο ρομποτικός βραχίονας ανυψώνεται
    while (k2 != 120) {
        k2 += 1; // έλεγχος της ακρίβειας περιστροφής
        myservo2.write(k2);
        delay(10); // έλεγχος της ταχύτητας περιστροφής του σερβο
    }
}

void servo2down() { // ο ρομποτικός βραχίονας χαμηλώνει
    while (k2 != 40) {
        k2 -= 1; // έλεγχος της ακρίβειας περιστροφής
        myservo2.write(k2);
        delay(10); // έλεγχος της ταχύτητας περιστροφής του σερβο
    }
}

void servo3left() { // ο ρομποτικός βραχίονας στρέφεται αριστερόστροφα
    while (k3!=180) {
        k3 += 1; // έλεγχος της ακρίβειας περιστροφής
        myservo3.write(k3);
        delay(10); // έλεγχος της ταχύτητας περιστροφής του σερβο
    }
}

void servo3center() { // ο ρομποτικός βραχίονας στέκεται στην κεντρική θέση
    while (k3!=90) {
        if (k3 > 90) {
            k3 -= 1;
        }
        if (k3 < 90)
            {k3 += 1;}
        // έλεγχος της ακρίβειας περιστροφής
        myservo3.write(k3);
        delay(10); // έλεγχος της ταχύτητας περιστροφής του σερβο
    }
}

```

```

void servo3right() { // ο ρομποτικός βραχίονας στρέφεται δεξιόστροφα

    while (k3!=10) {
        k3 -= 1; // έλεγχος της ακρίβειας περιστροφής
        myservo3.write(k3);
        delay(10); // έλεγχος της ταχύτητας περιστροφής του σερβο
    }
}

//Μεθόδοι Μνήμης//
void read_servo() {
    M1[i] = myservo1.read(); // αποθήκευση της τιμής της γωνίας των σερβομηχανισμών σε
    πίνακα
    delay(100); // χρόνος καθυστέρησης για να αποθηκευτεί η τιμή της γωνίας
    M2[i] = myservo2.read();
    delay(100);
    M3[i] = myservo3.read();
    delay(100);
    i++; // i προσθέτει 1 κάθε φορά που αποθηκεύεται i
    j = i; // ορίστε την τιμή του i στο j
    delay(200);
}

void do_servo() {
    i = 0;
    t = 1;
    k1 = myservo1.read(); // διαβάζει τις τιμές των γωνιών και τις θέτει στην ανάλογη
    μεταβλητή
    k2 = myservo2.read();
    k3 = myservo3.read();

    while (t) {
        for (int k = 0; k < j; k++) { // επανάληψη j φορές
            if (k1 < M1[k]) { // εάν η τρέχουσα τιμή γωνίας του σερβομηχανισμού 1
                είναι μικρότερη από την τιμή του πίνακα 1
                while (k1 < M1[k]) { // ο σερβομηχανισμός περιστρέφεται στη θέση όπου είναι
                    αποθηκευμένος στον πίνακα
                    myservo1.write(k1); // servo 1 εκτελεί την κίνηση
                    k1++; // K1 προσθέτει 1
                    delay(10); // καθυστέρηση για 10ms / έλεγχος της ταχύτητας
                    περιστροφής του σερβομηχανισμού
                }
            } else { // όταν η τιμή της γωνίας του σερβομηχανισμού 1 είναι
                μεγαλύτερη από την τιμή που έχει αποθηκευτεί στον πίνακα 1
                while (k1 > M1[k]) { // ο σερβομηχανισμός περιστρέφεται στη θέση όπου είναι
                    αποθηκευμένος στον πίνακα
                    myservo1.write(k1); // servo 1 εκτελεί την κίνηση
                    k1--; // k1 αφαιρεί 1

```

```

        delay(10);          // καθυστέρηση για 10ms / έλεγχος της ταχύτητας περιστροφής
του σερβομηχανισμού
    }
}
// το ίδιο παρακάτω
if (k2 < M2[k]) {
    while (k2 < M2[k]) {
        myservo2.write(k2);
        k2++;
        delay(10);
    }
} else {
    while (k2 > M2[k]) {
        myservo2.write(k2);
        k2--;
        delay(10);
    }
}

if (k3 < M3[k]) {
    while (k3 < M3[k]) {
        myservo3.write(k3);
        k3++;
        delay(10);
    }
} else {
    while (k3 > M3[k]) {
        myservo3.write(k3);
        k3--;
        delay(10);
    }
}

if (Serial.available() > 0) { // για την έξοδο από τον βρόχο
    if (Serial.read() == 't') { // λαμβάνει 'i' τότε 't'
        t = 0;                // θέτει το 't' σε 0, έξοδος από τον βρόχο
        break;
    }
}
}
}
}

//Μεθοδος αποφυγής εμποδίων//

int Ultrasonic_Ranging() {
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);

```

```

    digitalWrite(trigPin, LOW);
    int distance = pulseIn(echoPin, HIGH); // Ανάγνωση της διάρκειας των υψηλών επιπέδων
    distance = distance / 58;               // Μετατροπή του χρόνου παλμού σε απόσταση
    delay(50);
    return distance;
}

void avoid() { //αυτόνομη κίνηση, αποφυγή εμποδίων με υπερήχους
    bool avoid_flag = 1;
    while (avoid_flag) {
        int distance = Ultrasonic_Ranging();
        Serial.print("distance=");
        Serial.println(distance);
        if (distance < 30) { // υποθέτοντας ότι η μπροστινή απόσταση είναι μικρότερη από 30
cm
            if (distance < 15) { // υποθέτοντας ότι η μπροστινή απόσταση είναι μικρότερη από 15
cm
                stop();
                delay(100);
                back();
                delay(300);
            } else { // όταν 10 < απόσταση < 30, στροφή δεξιά
                stop();
                delay(100);
                turnR();
                delay(500);
            }
        } else { // όταν η απόσταση > 30, κίνηση μπροστά
            advance();
        }
        blue_val = Serial.read();
        if (blue_val == 'S') {
            stop();
            avoid_flag = 0;
        }
    }
}

//Μέθοδος ακολούθησης γραμμής
void tracking(void) {
    bool tracking_flag = 1;
    while (tracking_flag) {
        SL = digitalRead(SensorLeft);
        SM = digitalRead(SensorMiddle);
        SR = digitalRead(SensorRight);
        if (SM == HIGH) {
            if (SL == LOW && SR == HIGH) { // Αν ανιχνευτεί μαύρο στα δεξιά και λευκό στα
αριστερά, στρίβει δεξιά
                turnR();
            }
        }
    }
}

```

```

    } else if (SR == LOW && SL == HIGH) { // αν ανιχνευτεί μαύρο στα αριστερά και λευκό
στα δεξιά, στρίβει αριστερά
        turnL();
    } else { // Αν ανιχνευτεί λευκό και στις δύο άκρες, κινείται ευθεία
        advance();
    }
} else {
    if (SL == LOW && SR == HIGH) { // Αν ανιχνευτεί μαύρο στα δεξιά και λευκό στα
αριστερά, στρίβει δεξιά
        turnR();
    } else if (SR == LOW && SL == HIGH) { // Αν ανιχνευτεί μαύρο στα αριστερά και λευκό
στα δεξιά, στρίβει αριστερά
        turnL();
    } else { // Αν όλοι οι σένσορες ανιχνεύουν λευκό, το αυτοκίνητο σταματάει
        stop();
    }
}
blue_val = Serial.read();
if (blue_val == 'S') {
    stop();
    tracking_flag = 0;
}
}
}

//Μέθοδος αισθητήρα υπέρηχων
void follow() {
    bool follow_flag = 1;
    while (follow_flag) {
        int distance = Ultrasonic_Ranging();
        if (distance < 15) { // Αν η απόσταση είναι μικρότερη απο 15 εκατοστά, κίνηση προς τα
πίσω
            back();
        } else if (distance >= 15 && distance < 20) { // Αν η απόσταση είναι μεταξύ 15 και 20
εκατοστά, σταμάτα
            stop();
        } else if (distance >= 20 && distance < 40) { // Αν η απόσταση είναι μεταξύ 20 και 40
εκατοστά, κίνηση μπροστά
            advance();
        } else {
            stop(); // αλλιώς σταμάτα
        }
        blue_val = Serial.read();
        if (blue_val == 'S') {
            stop();
            follow_flag = 0;
        }
    }
}
}

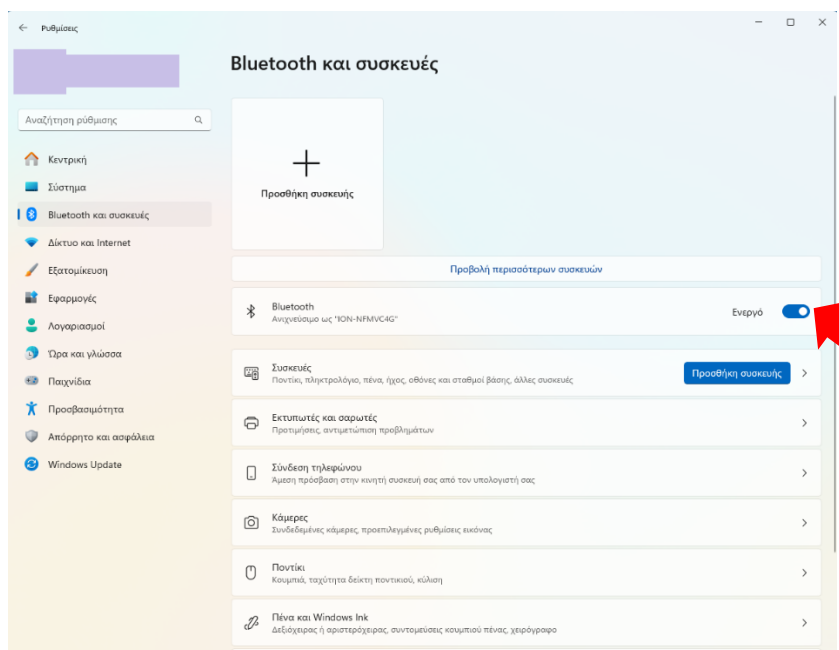
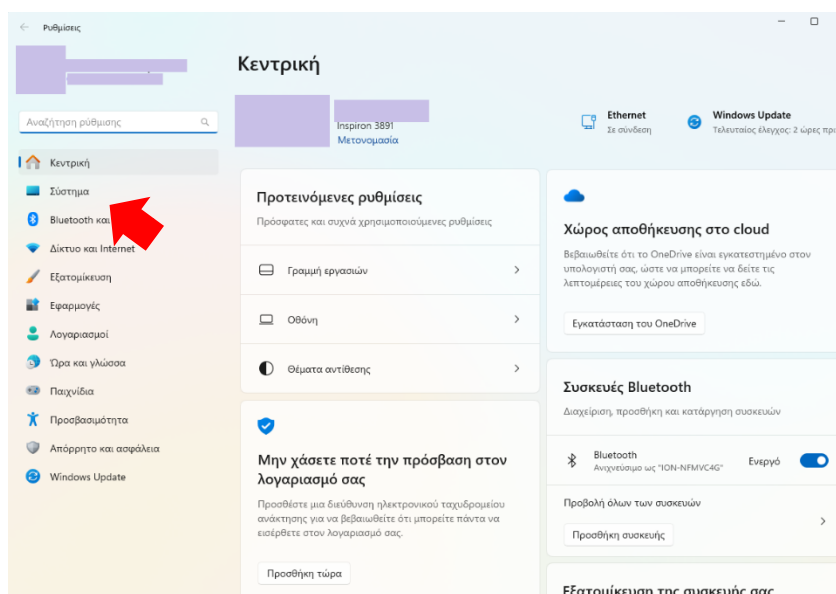
```

```
//Μέθοδος αντι-πτώσης
void Fall(void) {
    bool Fall_flag = 1;
    while (Fall_flag) {
        SL = digitalRead(SensorLeft);
        SM = digitalRead(SensorMiddle);
        SR = digitalRead(SensorRight);
        if (SM == HIGH || SL == HIGH || SR == HIGH) {
            back();
            delay(500);
            turnL();
            delay(500);
        } else {
            advance();
        }
        blue_val = Serial.read();
        if (blue_val == 'S') {
            stop();
            Fall_flag = 0;
        }
    }
}
```

Εναλλακτικά: Μέσα στον φάκελο «R4_Final_Setup» μπορείτε να ανοίξετε το αρχείο κάνοντας διπλό κλικ πάνω στο αρχείο  R4_Final_ardicon .

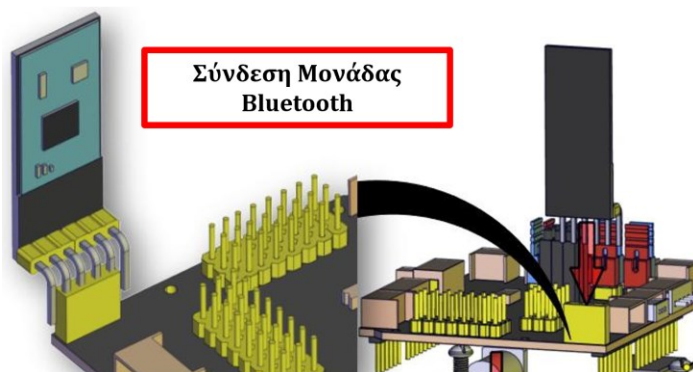
1.4.2 Ασύρματη Σύνδεση Ρομπότ R4 με τον Υπολογιστή (Bluetooth connection)

Πριν ξεκινήσετε να προγραμματίζετε το ρομπότ πρέπει πρώτα να πραγματοποιήσετε την ασύρματη σύζευξη του με τον υπολογιστή. Ενεργοποιήστε το Bluetooth του υπολογιστή. Ανοίξτε τις ρυθμίσεις του υπολογιστή σας και επιλέξτε **“Bluetooth και συσκευές”**.



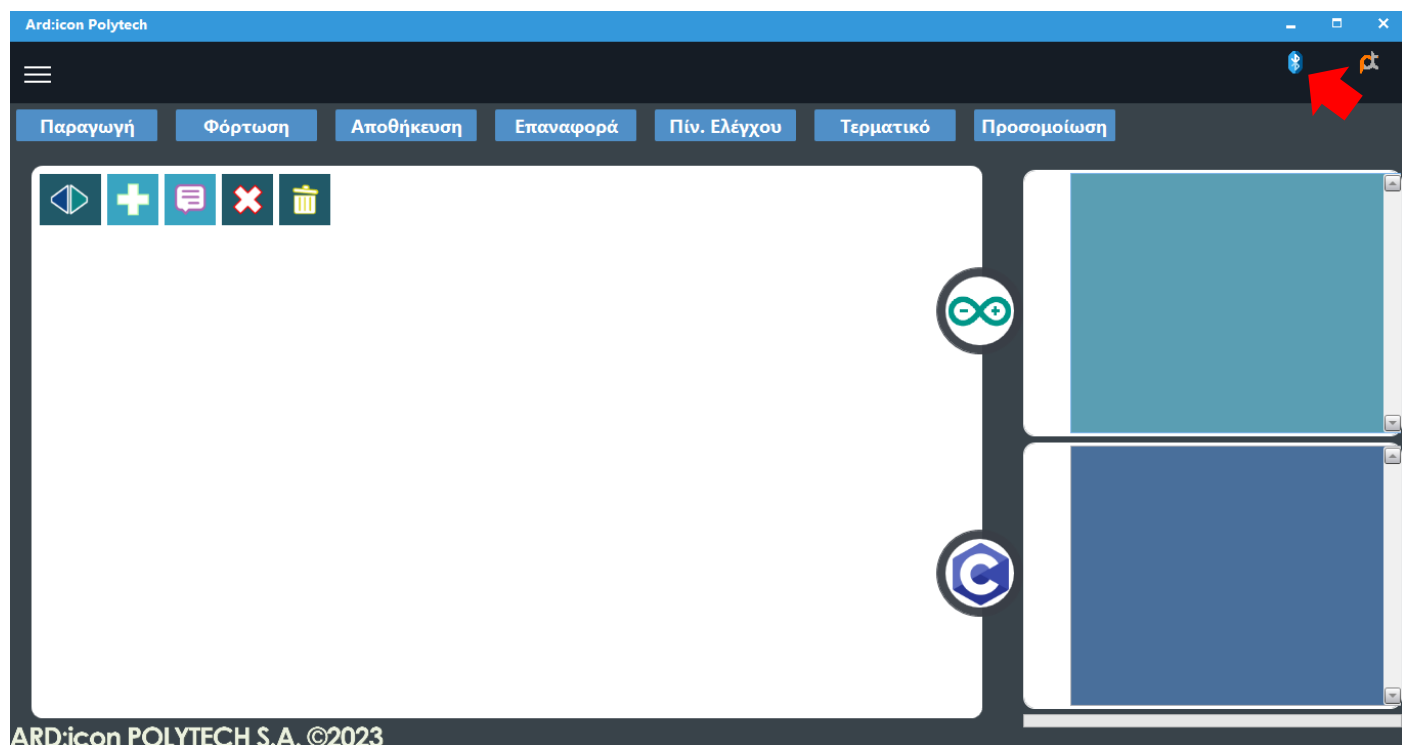
Επιλέξτε **«Ενεργό»**.

Τοποθετήστε τη συσκευή Bluetooth στην ειδική θέση του ρομπότ πάνω στη πλακέτα οδήγησης.

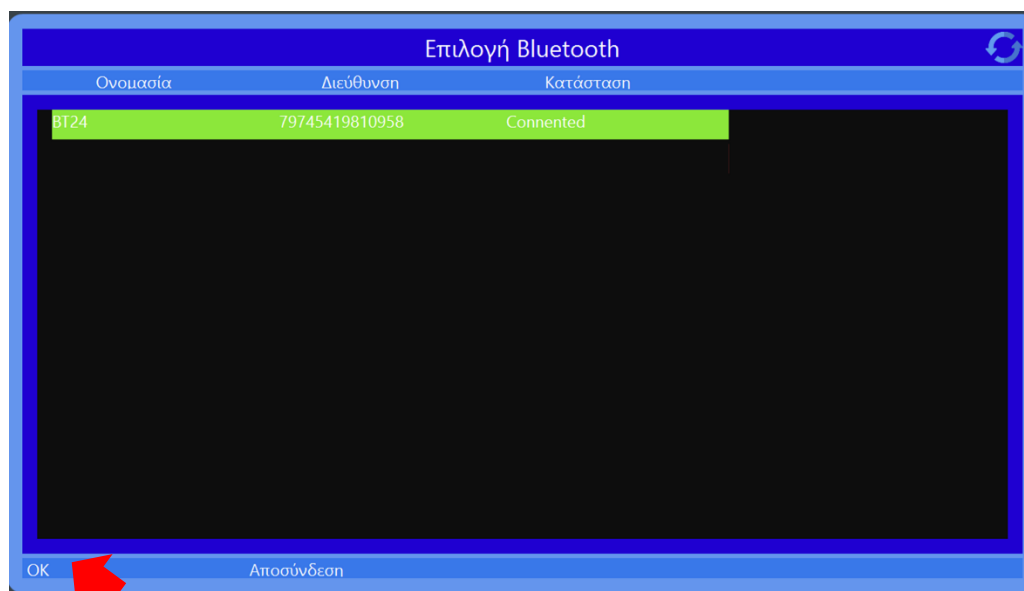


Ενεργοποιήστε το διακόπτη λειτουργίας Bluetooth στη πλακέτα οδήγησης.

Τρέξτε την εφαρμογή ARD:icon και πατήστε πάνω στο εικονίδιο Bluetooth που βρίσκεται πάνω δεξιά στην οθόνη προγραμματισμού.



Στην επόμενη οθόνη επιλέξτε τη συσκευή "BT....." και έπειτα OK.

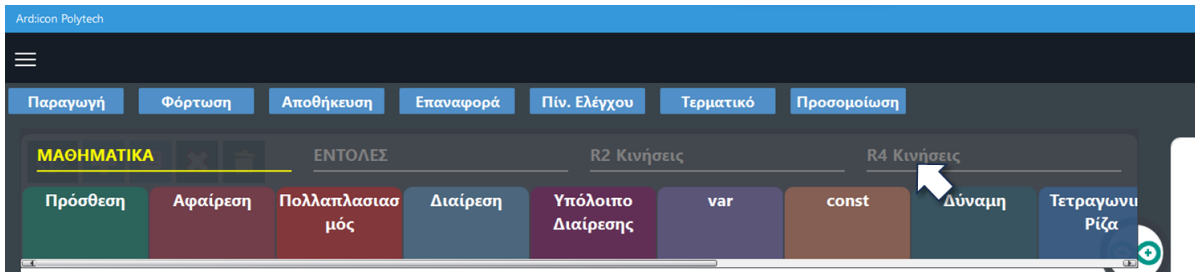


Επιλέξτε «ανανέωση» αν δεν εμφανίζεται η συσκευή στην

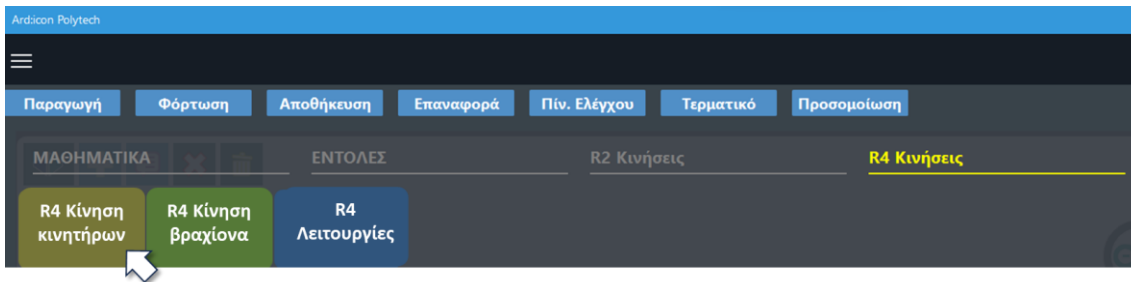
1.4.3 Προγραμματισμός εντολής “R4 Κίνηση κινητήρων”

1. Αφού έχετε ολοκληρώσει την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας επιλέξτε νέα εντολή πιέζοντας το εικονίδιο  στην αρχική οθόνη του προγράμματος ARD:icon.

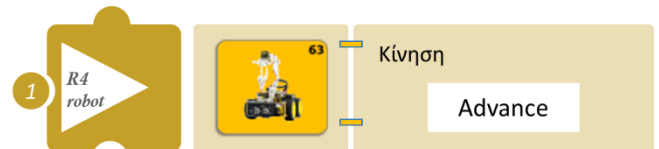
2. Επιλέξτε το μενού R4 Κινήσεις



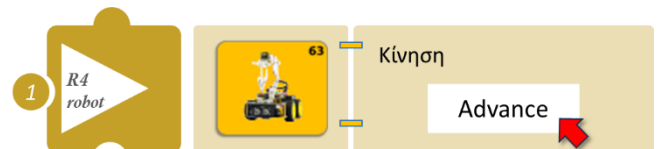
3. Επιλέξτε την εντολή “R4 Κίνηση κινητήρων”



4. Το πλακίδιο της εντολής εμφανίζεται στην αρχική οθόνη προγραμματισμού



5. Πατήστε πάνω στο κελί της κίνησης για να επιλέξετε τη κίνηση που θέλετε να εκτελέσει το ρομπότ.



6. Έχετε τις εξής επιλογές

- A. **Advance**: το ρομπότ κινείται προς τα μπροστά
- B. **TurnRight**: το ρομπότ στρίβει δεξιά
- Γ. **TurnLeft**: Το ρομπότ στρίβει αριστερά
- Δ. **Stop**: το ρομπότ σταματάει να κινείται.
- Ε. **Back**: Το ρομπότ κινείται προς τα πίσω.
- ΣΤ. **SpeedFast**: Το ρομπότ κινείται με μεγάλη ταχύτητα.
- Ζ. **SpeedLow**: Το ρομπότ κινείται με χαμηλή ταχύτητα.

7. Επιλέξτε τη κίνηση που θέλετε π.χ. Advance και επιλέξτε **Παραγωγή** για να δημιουργήσετε τον κώδικα στο λογισμικό.

8. Μετά τις εντολές “**TurnRight**” και “**TurnLeft**” και αφού προγραμματίσετε το χρονικό διάστημα που θέλετε να στρίβει το ρομπότ πρέπει να προσθέσετε την εντολή “**Stop**” πριν προγραμματίσετε κάποια άλλη κίνηση του ρομπότ.

9. Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.

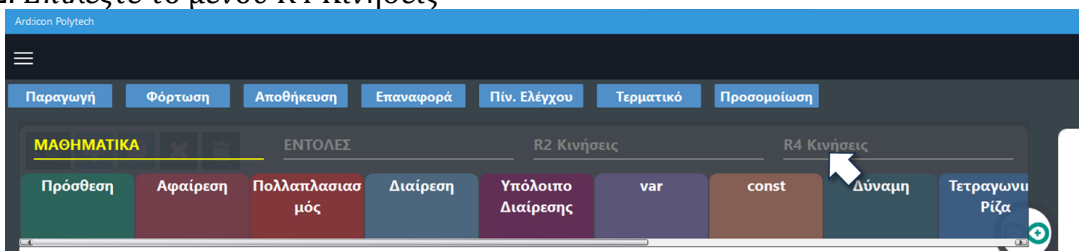
10. Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**.

11. Αν θέλετε να σταματήσετε το ρομπότ, απενεργοποιήστε τον διακόπτη λειτουργίας του ρομπότ και έπειτα επιλέξτε **OFF** στον πίνακα ελέγχου της εφαρμογής.

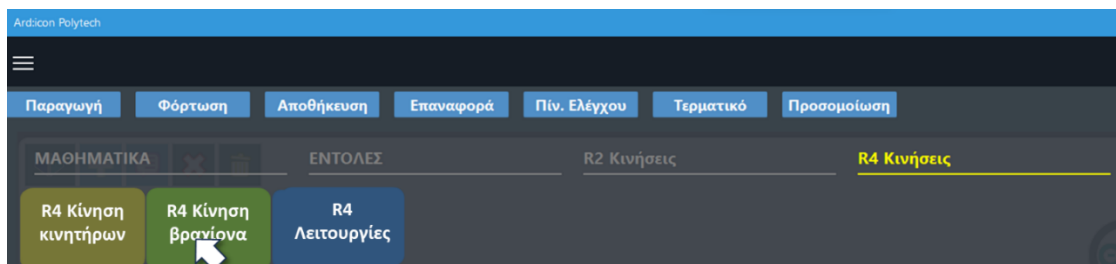
1.4.4 Προγραμματισμός εντολής “R4 Κίνηση βραχίονα”

1. Αφού έχετε ολοκληρώσει την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας επιλέξτε νέα εντολή πιέζοντας το εικονίδιο  στην αρχική οθόνη του προγράμματος ARD:icon.

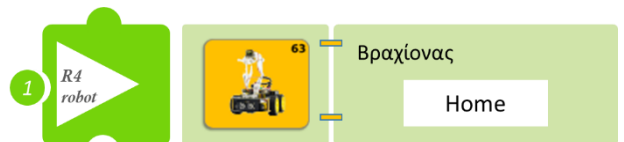
2. Επιλέξτε το μενού R4 Κινήσεις



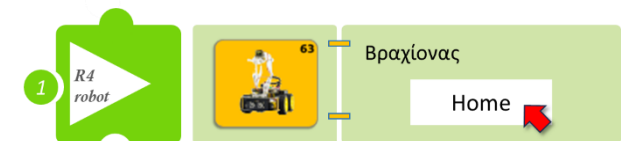
3. Επιλέξτε την εντολή “R4 Κίνηση βραχίονα”



4. Το πλακίδιο της εντολής εμφανίζεται στην αρχική οθόνη προγραμματισμού



5. Πατήστε πάνω στο κελί της κίνησης για να επιλέξετε τη κίνηση που θέλετε να εκτελέσει ο βραχίονας του ρομπότ.



6. Έχετε τις εξής επιλογές

- A. **Home**: Αρχική θέση βραχίονα
- B. **Front**: Ο βραχίονας μετακινείται μπροστά
- Γ. **Right**: Ο βραχίονας μετακινείται δεξιά
- Δ. **Left**: Ο βραχίονας μετακινείται αριστερά
- Ε. **Open**: Ανοίγει η δαγκάνα του βραχίονα
- ΣΤ. **Grab**: Κλείνει η δαγκάνα του βραχίονα

7. Επιλέξτε τη κίνηση που θέλετε π.χ. “Front” και επιλέξτε **Παραγωγή** για να δημιουργήσετε τον κώδικα στο λογισμικό.

8. Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.

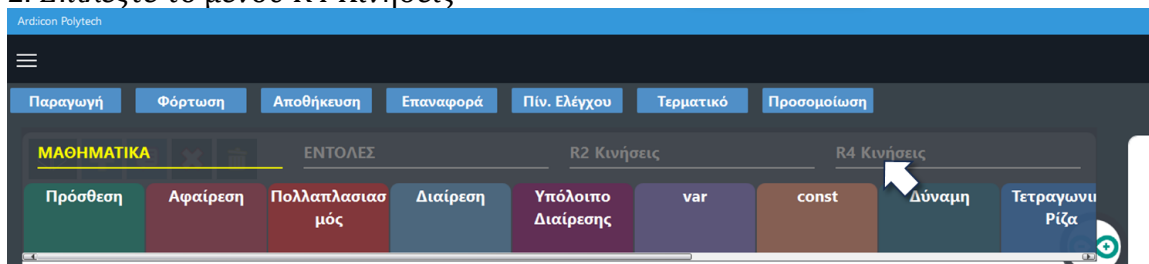
9. Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**.

10. Αν θέλετε να σταματήσετε το ρομπότ, απενεργοποιήστε τον διακόπτη λειτουργίας του ρομπότ και έπειτα επιλέξτε **OFF** στον πίνακα ελέγχου της εφαρμογής.

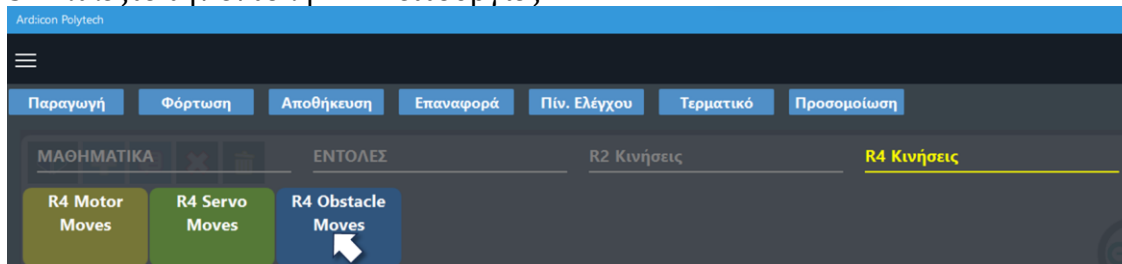
1.4.3 Προγραμματισμός εντολής “R4 Λειτουργίες”

1. Αφού έχετε ολοκληρώσει την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας επιλέξτε νέα εντολή πιέζοντας το εικονίδιο  στην αρχική οθόνη του προγράμματος ARD:icon.

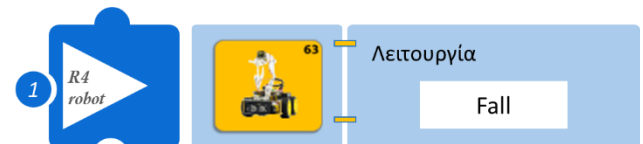
2. Επιλέξτε το μενού R4 Κινήσεις



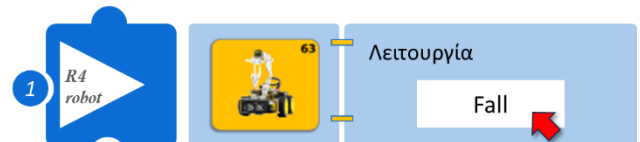
3. Επιλέξτε την εντολή “R4 Λειτουργίες”



4. Το πλακίδιο της εντολής εμφανίζεται στην αρχική οθόνη προγραμματισμού.



5. Πατήστε πάνω στο κελί της κίνησης για να επιλέξετε τη κίνηση που θέλετε να εκτελέσει ο βραχίονας του ρομπότ.



6. Έχετε τις εξής επιλογές

A. **Fall**: Λειτουργία μη πτώσης του ρομπότ

B. **Follow**: Με τη λειτουργία αυτή το ρομπότ ακολουθεί ένα εμπόδιο που τοποθετείται μπροστά του

Γ. **Tracking**: Λειτουργία παρακολούθησης γραμμής

Δ. **Avoid**: Λειτουργία αποφυγής εμποδίου

7. Επιλέξτε τη κίνηση που θέλετε π.χ. “Follow” και επιλέξτε **Παραγωγή** για να δημιουργήσετε τον κώδικα στο λογισμικό.

8. Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
9. Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί . Επιλέξτε **ON**.
10. Αν θέλετε να σταματήσετε το ρομπότ, απενεργοποιήστε τον διακόπτη λειτουργίας του ρομπότ και έπειτα επιλέξτε **OFF** στον πίνακα ελέγχου της εφαρμογής.

1.5 Δραστηριότητες με ARD:icon

Δραστηριότητα 1: Προγραμματισμός συνδυασμού κινήσεων Ρομπότ

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα μάθετε πως να δημιουργείτε μια ρουτίνα κινήσεων του ρομπότ. Θα προγραμματίσετε το ρομπότ να κινείται προς τα μπροστά για 10 δευτερόλεπτα έπειτα να στρίβει αριστερά και να συνεχίσει για άλλα 10 δευτερόλεπτα ευθεία. Αυτή η ρουτίνα κινήσεων θα συνεχίσει να εκτελείται μέχρι να απενεργοποιήσετε το ρομπότ.

Θα χρησιμοποιήσετε τις εντολές “ [R4 Κίνηση κινητήρων](#) ” και “[delay](#)”.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Καλώδιο USB

Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Ενεργοποιήστε το ρομπότ.

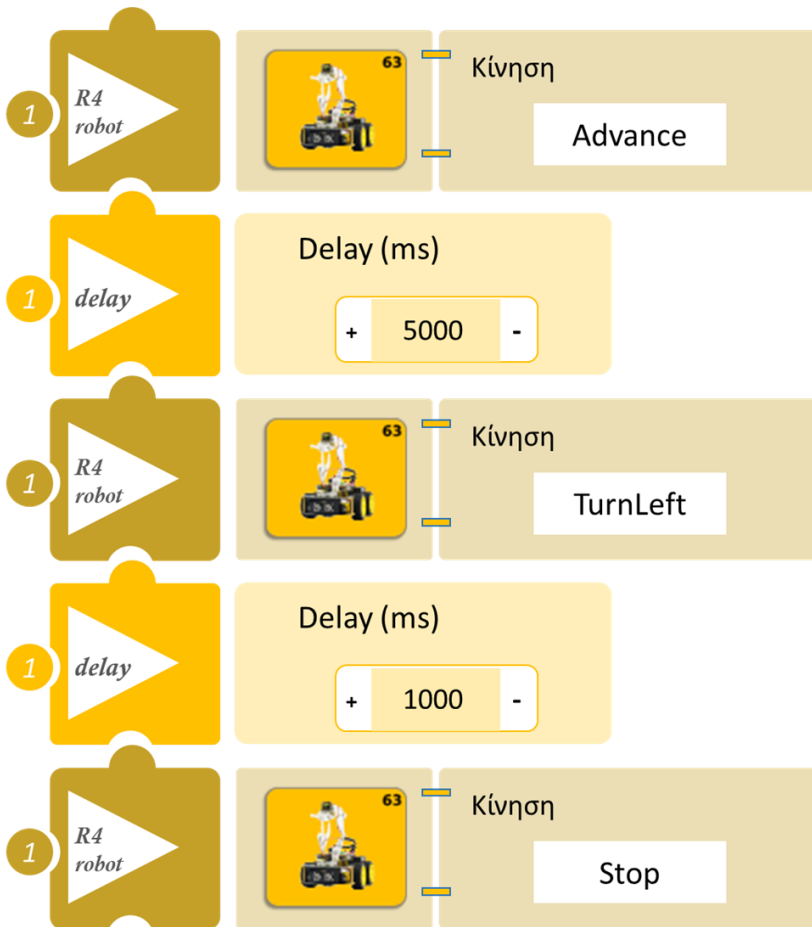
Προγραμματισμός Ρομπότ

Για το προγραμματισμό των εντολών που θα εκτελεί το ρομπότ θα χρησιμοποιούμε το παρακάτω πίνακα.

1. Στη στήλη «**Γραμμή Εντολών**» εμφανίζεται η Γραμμή Εντολών της συνάρτησης. Ξεκινάτε πάντα με τη 1^η Γραμμή Εντολών.
2. Στη στήλη «**Εντολή**» εμφανίζεται η εντολή που πρέπει να επιλέξετε, π.χ “ [R4 Κίνηση κινητήρων](#) ”.
3. Στη στήλη «**Συσκευή**» εμφανίζεται η συσκευή που ελέγχει η εντολή που επιλέξατε, π.χ. LED, αισθητήρας, Robo κτλ.
4. Στη στήλη «**Θύρα/Κατάσταση**» εμφανίζονται οι επιμέρους ρυθμίσεις τις εντολής π.χ. “Front” ή “On” κτλ.

<u>Γραμμή Εντολών</u>	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	“ R4 Κίνηση κινητήρων ”	Robot	Advance
1 ^η	“ delay ”		5000ms
1 ^η	“ R4 Κίνηση κινητήρων ”	Robot	TurnLeft
1 ^η	“ delay ”		1000ms
1 ^η	“ R4 Κίνηση κινητήρων ”	Robot	Stop

Πλακίδια Εντολών Κυκλώματος



- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Παρατηρήστε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει.
- Αν θέλετε να σταματήσετε το ρομπότ είτε απενεργοποιήστε τον διακόπτη λειτουργίας του ρομπότ και έπειτα επιλέξτε **OFF** στον πίνακα ελέγχου της εφαρμογής.
- Προσαρμόστε το χρόνο που κινείται το ρομπότ προς κάθε κατεύθυνση ("delay") ανάλογα με τις προτιμήσεις σας.

Δραστηριότητα 2: Προγραμματισμός συνδυασμού κινήσεων Βραχίονα

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα μάθετε πως να δημιουργείτε μια ρουτίνα κινήσεων του βραχίονα του ρομπότ. Θα προγραμματίσετε το ρομπότ να στρέφεται αριστερά να συλλέγει ένα αντικείμενο, έπειτα να στρέφεται δεξιά για να αφήσει το αντικείμενο και στη συνέχεια να επανέρχεται στην αρχική του θέση για 5 δευτερόλεπτα. Αυτή η ρουτίνα κινήσεων θα συνεχίσει να εκτελείται μέχρι να απενεργοποιήσετε το ρομπότ.

Θα χρησιμοποιήσετε τις εντολές “ [R4 Κίνηση βραχίονα](#) ” και “ [delay](#) ”.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Καλώδιο USB

Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

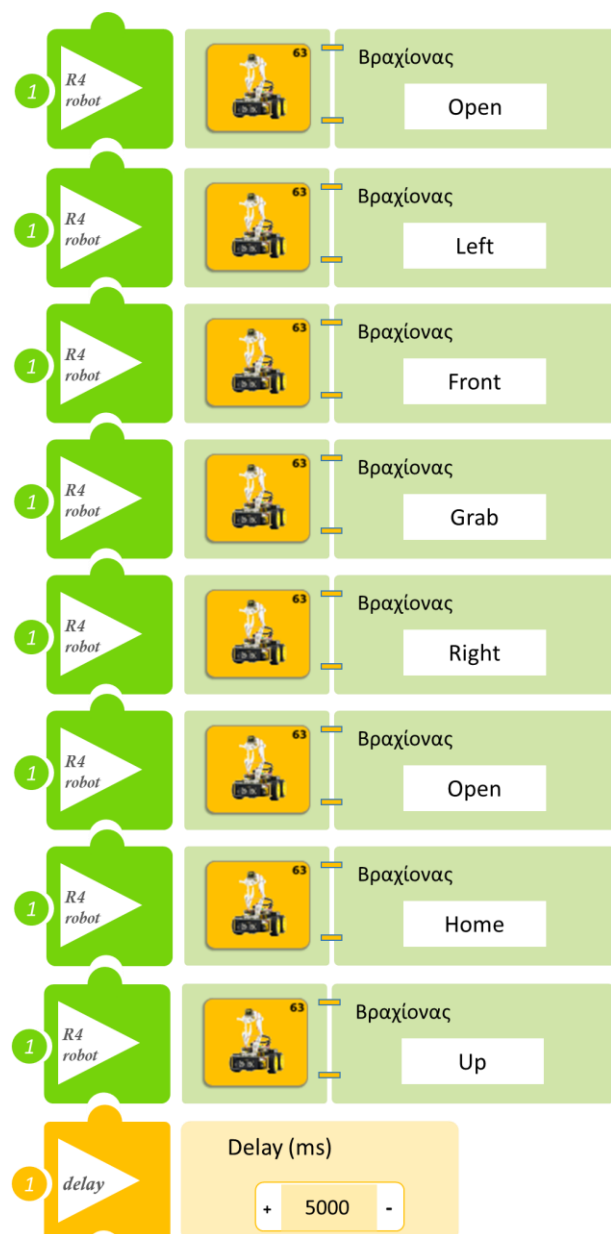
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Ενεργοποιήστε το ρομπότ.

Προγραμματισμός Ρομπότ

<u>Γραμμή Εντολών</u>	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	“ R4 Κίνηση βραχίονα ”	Robot	Open
1 ^η	“ R4 Κίνηση βραχίονα ”	Robot	Left
1 ^η	“ R4 Κίνηση βραχίονα ”	Robot	Front
1 ^η	“ R4 Κίνηση βραχίονα ”	Robot	Grab
1 ^η	“ R4 Κίνηση βραχίονα ”	Robot	Right
1 ^η	“ R4 Κίνηση βραχίονα ”	Robot	Open
1 ^η	“ R4 Κίνηση βραχίονα ”	Robot	Home
1 ^η	“ R4 Κίνηση βραχίονα ”	Robot	Up
1 ^η	“ delay ”		5000ms

Πλακίδια Εντολών Κυκλώματος



- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Παρατηρήστε το βραχίονα του ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει

Δραστηριότητα 3: Προγραμματισμός Ταχύτητας Ρομπότ

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα μάθετε πως να προγραμματίζετε την ταχύτητα με την οποία κινείται το ρομπότ.

Θα χρησιμοποιήσετε τις εντολές “ [R4 Κίνηση κινητήρων](#)” και “[delay](#)”.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Καλώδιο USB

Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

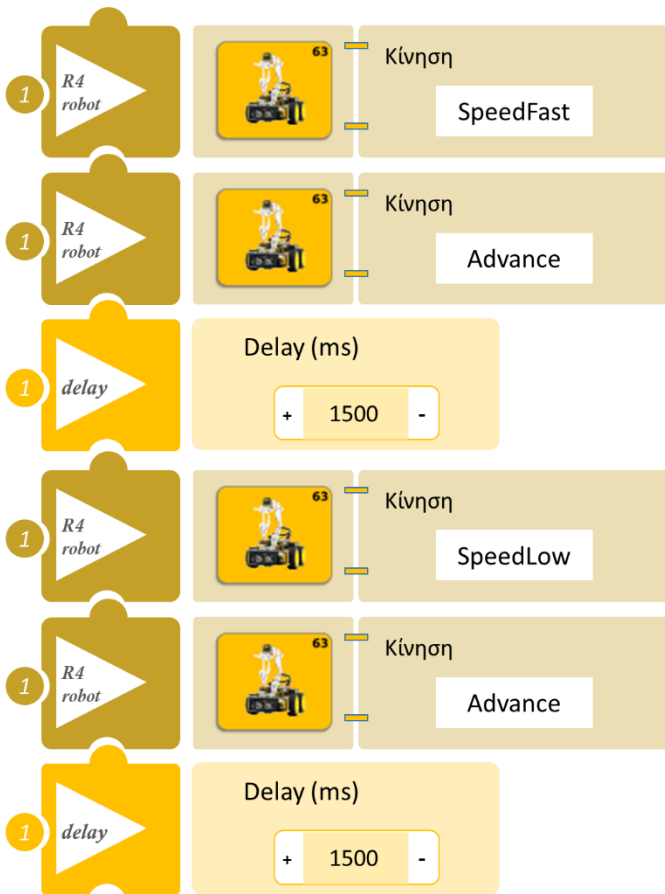
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Ενεργοποιήστε το ρομπότ.

Προγραμματισμός Ρομπότ

<u>Γραμμή Εντολών</u>	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	“ R4 Κίνηση κινητήρων ”	Robot	SpeedFast
1 ^η	“ R4 Κίνηση κινητήρων ”	Robot	Advance
1 ^η	“ delay ”		1500ms
1 ^η	“ R4 Κίνηση κινητήρων ”	Robot	SpeedLow
1 ^η	“ R4 Κίνηση κινητήρων ”	Robot	Advance
1 ^η	“ delay ”		1500ms

Πλακίδια Εντολών Κυκλώματος



- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.
-

ΑΠΟΤΕΛΕΣΜΑ

- Παρατηρήστε το ρομπότ να κινείται με δύο διαφορετικές ταχύτητες γρήγορη ταχύτητα και αργή ταχύτητα.
- Αν σε μία δραστηριότητα δεν ορίσετε την ταχύτητα τότε το ρομπότ θα εκτελεί τις κινήσεις που προγραμματίσατε με γρήγορη ταχύτητα (είναι η προκαθορισμένη ταχύτητα του ρομπότ).

Δραστηριότητα 4: Συνδυασμός κινήσεων κινητήρων και βραχίονα ρομπότ

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα προγραμματίσετε το ρομπότ να εκτελεί μία ρουτίνα κινήσεων προκειμένου, να συλλέξει ένα αντικείμενο που βρίσκεται σε κάποιο σημείο μπροστά στο χώρο και να το αφήνει σε ένα άλλο σημείο δεξιά.

Πιο συγκεκριμένα θα προγραμματίσετε το ρομπότ έτσι ώστε, να κινείται ευθεία για 3 δευτερόλεπτα και να συλλέγει ένα αντικείμενο που βρίσκεται μπροστά του. Έπειτα να στρίβει δεξιά να αφήνει το αντικείμενο και τέλος να στρίβει αριστερά, ώστε να κινηθεί πάλι στην ευθεία για 3 δευτερόλεπτα. Αυτή η ρουτίνα κινήσεων θα συνεχίσει να εκτελείται μέχρι να απενεργοποιήσετε το ρομπότ R4.

Θα χρησιμοποιήσετε τις εντολές: “R4 Κίνηση κινητήρων”, “ R4 Κίνηση βραχίονα ” και “delay”.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Καλώδιο USB

Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

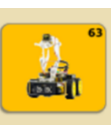
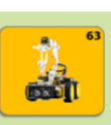
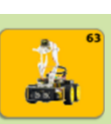
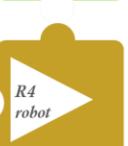
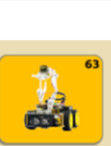
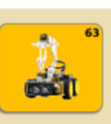
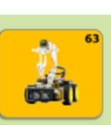
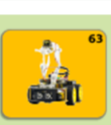
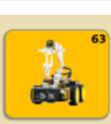
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Ενεργοποιήστε το ρομπότ.

Προγραμματισμός Ρομπότ

<u>Γραμμή Εντολών</u>	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	“R4 Κίνηση κινητήρων”	Robot	Advance
1 ^η	“delay”		3000ms
1 ^η	“R4 Κίνηση κινητήρων”	Robot	Stop
1 ^η	“R4 Κίνηση βραχίονα”	Robot	Front
1 ^η	“R4 Κίνηση βραχίονα”	Robot	Grab
1 ^η	“R4 Κίνηση κινητήρων”	Robot	TurnRight
1 ^η	“delay”		750ms
1 ^η	“R4 Κίνηση κινητήρων”	Robot	Stop
1 ^η	“R4 Κίνηση βραχίονα”	Robot	Open
1 ^η	“R4 Κίνηση βραχίονα”	Robot	Up
1 ^η	“R4 Κίνηση κινητήρων”	Robot	TurnLeft
1 ^η	“delay”		750ms
1 ^η	“R4 Κίνηση κινητήρων”	Robot	Stop

Πλακίδια Εντολών Κυκλώματος

1		 63	Κίνηση Advance
1		Delay (ms) + 3000 -	
1		 63	Κίνηση Stop
1		 63	Βραχίονας Front
1		 63	Βραχίονας Grab
1		 63	Κίνηση TurnRight
1		Delay (ms) + 750 -	
1		 63	Κίνηση Stop
1		 63	Βραχίονας Open
1		 63	Βραχίονας Up
1		 63	Κίνηση TurnLeft
1		Delay (ms) + 750 -	
1		 63	Κίνηση Stop

- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί . Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Παρατηρήστε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει.

Δραστηριότητα 5: Συνδυασμός κινήσεων κινητήρων και βραχίονα ρομπότ II

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα προγραμματίσετε το ρομπότ να εκτελεί μια ρουτίνα κινήσεων προκειμένου να συλλέξει ένα αντικείμενο που βρίσκεται σε κάποιο σημείο μπροστά στο χώρο και να το αφήνει σε ένα άλλο σημείο προς τα δεξιά **ΚΑΙ** να επιστρέφει πίσω στην αρχική του θέση. Αυτή η ρουτίνα κινήσεων θα συνεχίσει να εκτελείται μέχρι να απενεργοποιήσετε το ρομπότ.

Θα χρησιμοποιήσετε τις εντολές: "[R4 Κίνηση κινητήρων](#)", "[R4 Κίνηση βραχίονα](#)" και "[delay](#)".

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Καλώδιο USB

Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

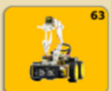
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

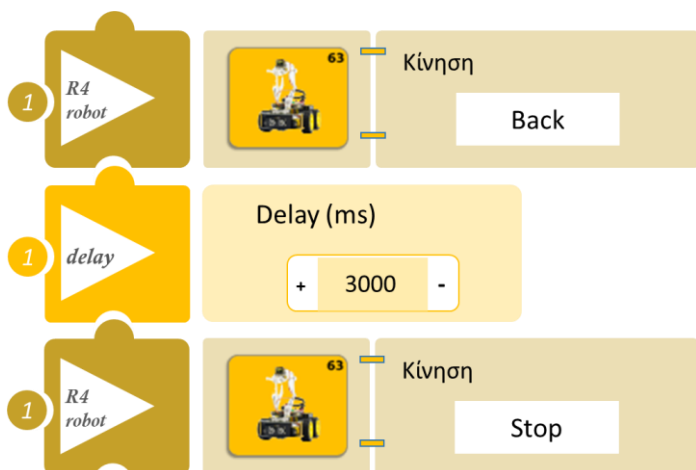
Ενεργοποιήστε το ρομπότ.

Προγραμματισμός Ρομπότ

<u>Γραμμή Εντολών</u>	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	" R4 Κίνηση κινητήρων "	Robot	Advance
1 ^η	" delay "		3000ms
1 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
1 ^η	" R4 Κίνηση βραχίονα "	Robot	Front
1 ^η	" R4 Κίνηση βραχίονα "	Robot	Grab
1 ^η	" R4 Κίνηση κινητήρων "	Robot	TurnRight
1 ^η	" delay "		750ms
1 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
1 ^η	" R4 Κίνηση βραχίονα "	Robot	Open
1 ^η	" R4 Κίνηση βραχίονα "	Robot	Up
1 ^η	" R4 Κίνηση κινητήρων "	Robot	TurnLeft
1 ^η	" delay "		750ms
1 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
1 ^η	" R4 Κίνηση κινητήρων "	Robot	Back
1 ^η	" delay "		3000ms
1 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop

Πλακίδια Εντολών Κυκλώματος

1		 63	Κίνηση Advance
1		Delay (ms) + 3000 -	
1		 63	Κίνηση Stop
1		 63	Βραχίονας Front
1		 63	Βραχίονας Grab
1		 63	Κίνηση TurnRight
1		Delay (ms) + 750 -	
1		 63	Κίνηση Stop
1		 63	Βραχίονας Open
1		 63	Βραχίονας Up
1		 63	Κίνηση TurnLeft
1		Delay (ms) + 750 -	
1		 63	Κίνηση Stop

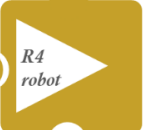
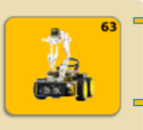
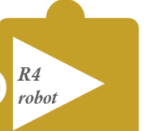
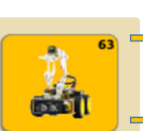
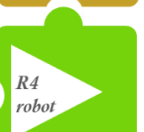
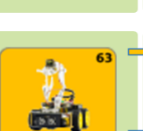
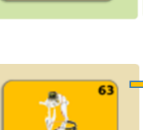
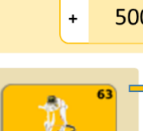
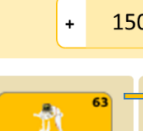
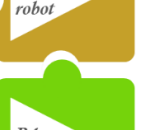
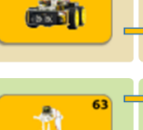


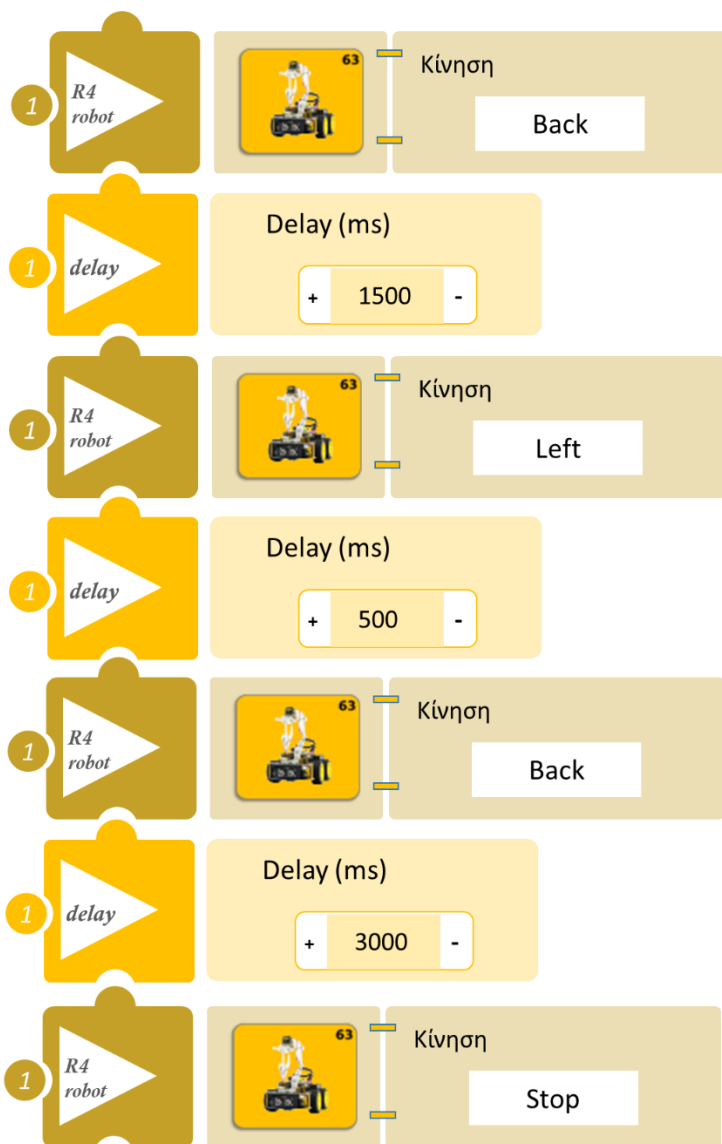
- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Παρατηρήσετε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει

Πλακίδια Εντολών Κυκλώματος

1			Κίνηση Advance
1		Delay (ms) + 3000 -	
1			Κίνηση Stop
1			Βραχίονας Front
1			Βραχίονας Grab
1			Κίνηση TurnRight
1		Delay (ms) + 500 -	
1			Κίνηση Advance
1		Delay (ms) + 1500 -	
1			Κίνηση Stop
1			Βραχίονας Open



- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**.

ΑΠΟΤΕΛΕΣΜΑ

- Παρατηρήστε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει

Δραστηριότητα 6: Έναρξη Λειτουργίας Ρομπότ με Ήχο

Περιγραφή

Στη συγκεκριμένη δραστηριότητα σε συνέχεια της προηγούμενης θα προγραμματίσετε το ρομπότ έτσι ώστε **ΑΝ** ο ήχος είναι πάνω από ένα επίπεδο **ΤΟΤΕ** να εκτελεί μια ρουτίνα κινήσεων προσπαθώντας να σχηματίσει ένα τετράγωνο.

ΔΙΑΦΟΡΕΤΙΚΑ θα παραμένει ακίνητο.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας Ήχου **AJS02**
- Καλώδια UTP (x1)
- Καλώδιο USB

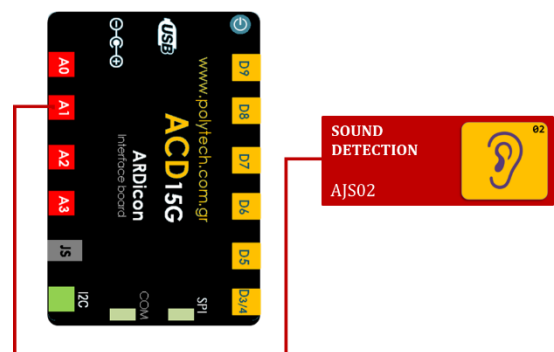
Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **AJS02** στην αναλογική θύρα 1 (A1) του ελεγκτή ARD:icon.

[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Ενεργοποιήστε το ρομπότ.



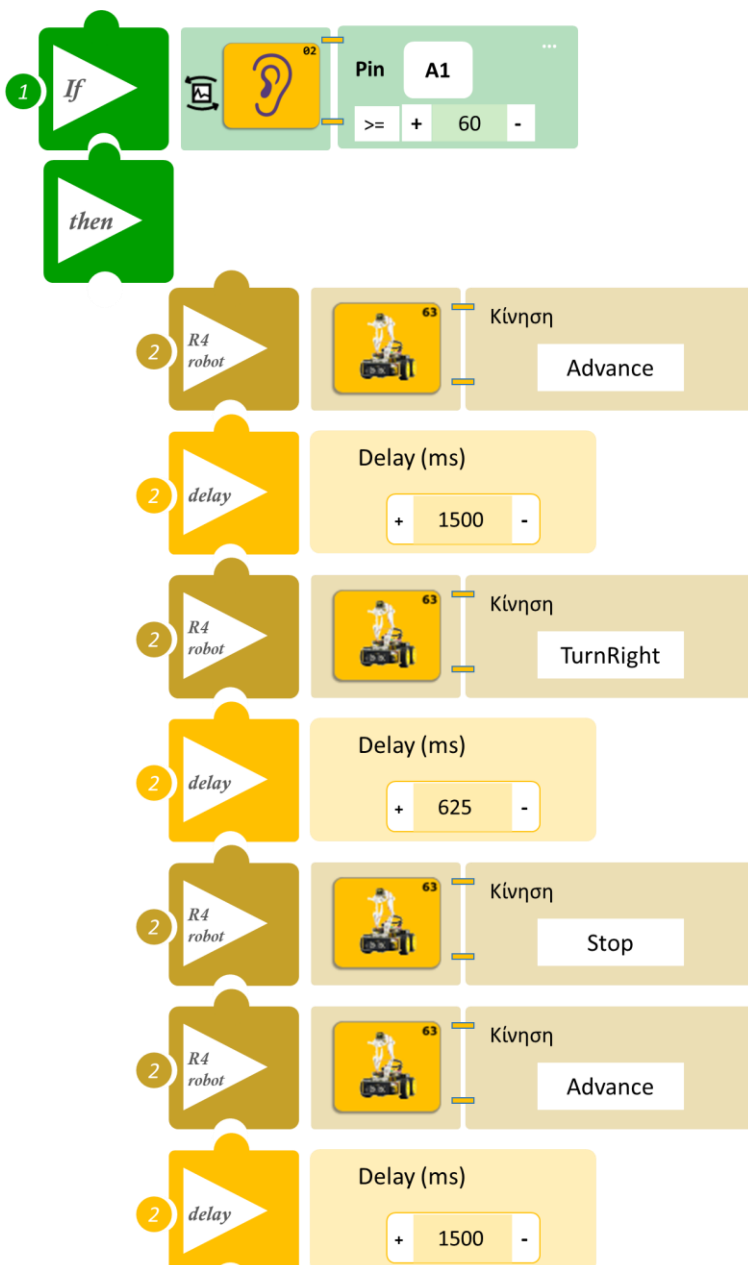
Προγραμματισμός Ρομπότ

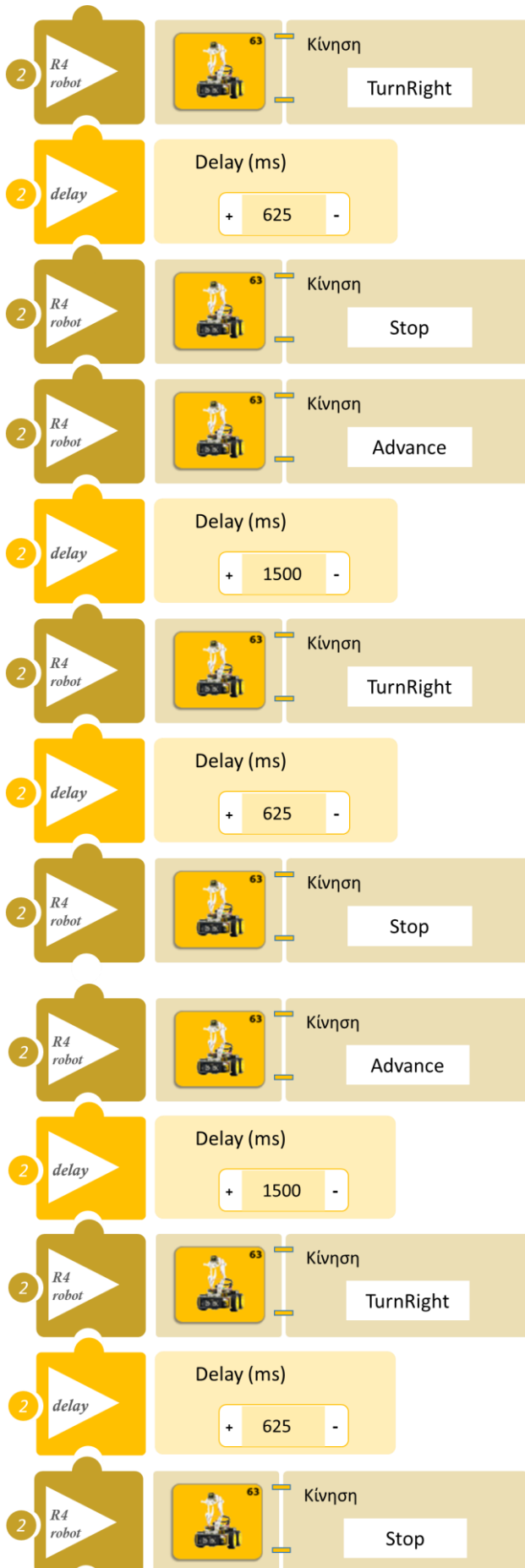
Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	"if/then"	AJS02	≥ 60 / A1
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		1500ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnRight
2 ^η	"delay"		625ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		1500ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnRight
2 ^η	"delay"		625ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		1500ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnRight
2 ^η	"delay"		625ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance

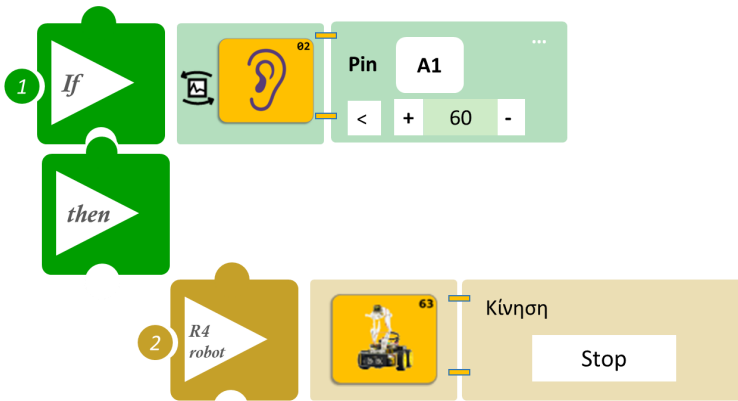
2 ^η	" <u>delay</u> "		1500ms
2 ^η	" <u>R4 Κίνηση κινητήρων</u> "	Robot	TurnRight
2 ^η	" <u>delay</u> "		625ms
2 ^η	" <u>R4 Κίνηση κινητήρων</u> "	Robot	Stop
1 ^η	" <u>if/then</u> "	AJS02	< 60 / A1
2 ^η	" <u>R4 Κίνηση κινητήρων</u> "	Robot	Stop

- **Σημείωση:** Προσαρμόστε το χρόνο που στρίβει το ρομπότ ("delay") , αν χρειαστεί προκειμένου να δημιουργήσετε το νοητό τετράγωνο. Επίσης μπορεί να χρειαστεί να προσαρμόσετε το χρόνο ανάλογα με την απόδοση των μπαταριών. Όσο μειώνεται η απόδοση των μπαταριών μειώνεται και η ταχύτητα του ρομπότ.

Πλακίδια Εντολών Κυκλώματος







- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Κάντε ένα δυνατό θόρυβο κοντά στον αισθητήρα ή φυσήξτε απαλά τον αισθητήρα και παρατηρήστε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει.

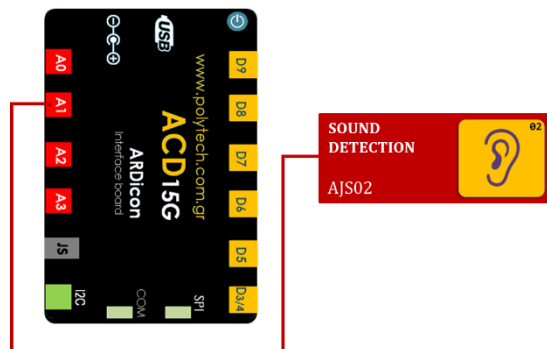
Δραστηριότητα 7: Έναρξη Λειτουργίας Ρομπότ με Ήχο II

Περιγραφή

Στη συγκεκριμένη δραστηριότητα σε συνέχεια της προηγούμενης αφού έχετε ήδη προγραμματίσει το ρομπότ να εκτελεί την κίνηση τετράγωνο και μπορέσατε να οριοθετήσετε τις γωνίες του ρομπότ, θα προσθέσετε ένα αντικείμενο στη πρώτη γωνία που κάνει το ρομπότ. Το ρομπότ θα συλλέγει το αντικείμενο για να το τοποθετήσει στη δεύτερη γωνία που σχηματίζει.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας Ήχου **AJS02**
- Καλώδια UTP (x1)
- Καλώδιο USB



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **AJS02** στην αναλογική θύρα 1 (**A1**) του ελεγκτή ARD:icon.

[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Ενεργοποιήστε το ρομπότ.

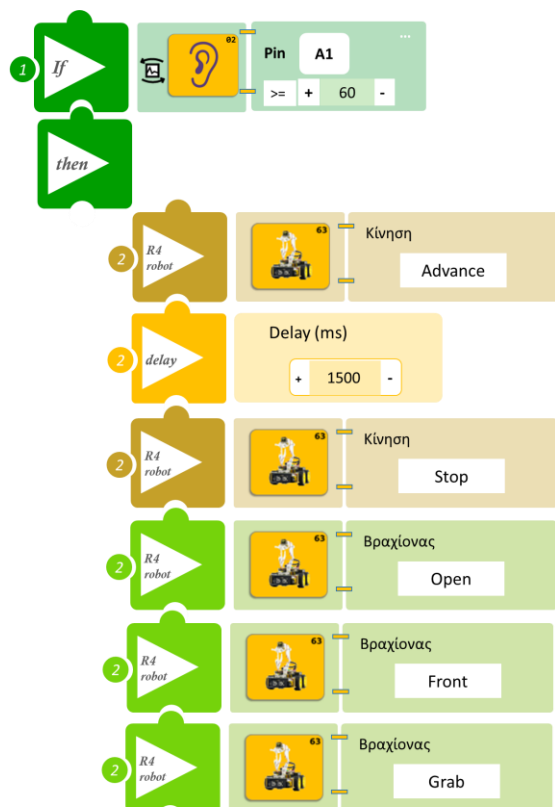
Προγραμματισμός Ρομπότ

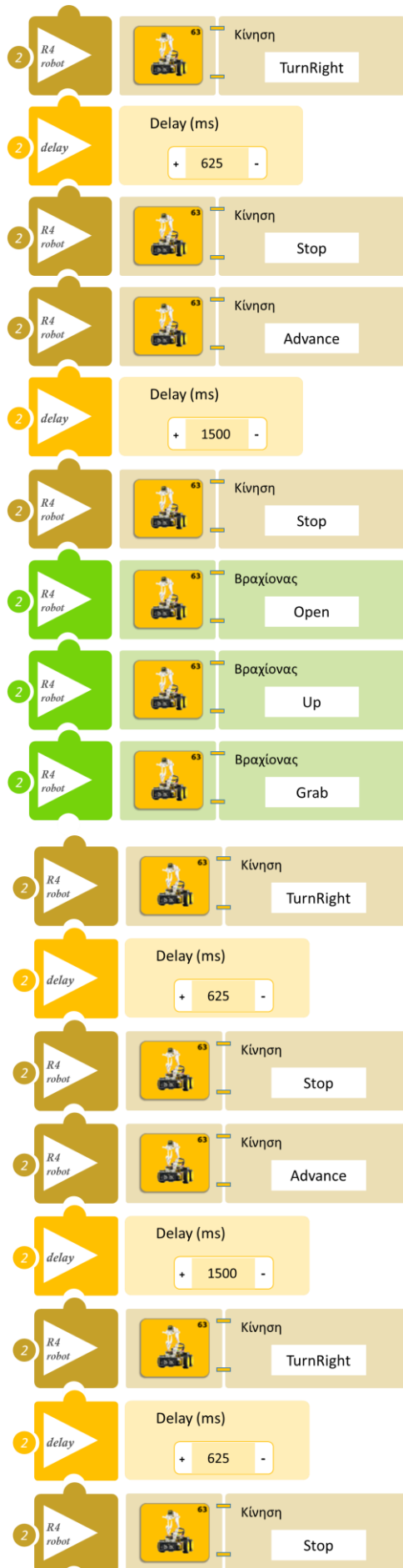
Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	" if/then "	AJS02	≥ 60 / A1
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Advance
2 ^η	" delay "		1500ms
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Open
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Front
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Grab
2 ^η	" R4 Κίνηση κινητήρων "	Robot	TurnRight
2 ^η	" delay "		625ms
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Advance
2 ^η	" delay "		1500ms
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Open
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Front
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Home
2 ^η	" R4 Κίνηση κινητήρων "	Robot	TurnRight

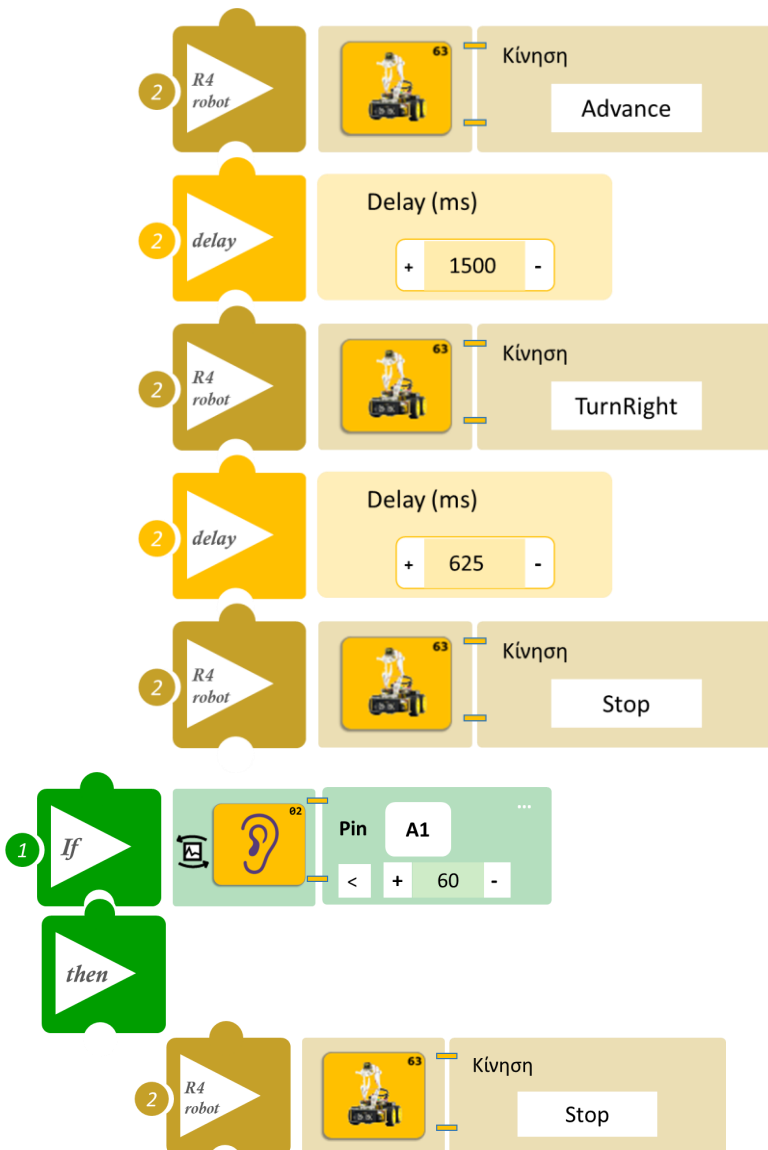
2 ^η	" delay "		625ms
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Advance
2 ^η	" delay "		1500ms
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
2 ^η	" R4 Κίνηση κινητήρων "	Robot	TurnRight
2 ^η	" delay "		625ms
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Advance
2 ^η	" delay "		1500ms
2 ^η	" R4 Κίνηση κινητήρων "	Robot	TurnRight
2 ^η	" delay "		625ms
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
1 ^η	" if/then "	AJS02	< 60 / A1
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop

- **Σημείωση:** Προσαρμόστε το χρόνο που στρίβει το ρομπότ ("delay") , αν χρειαστεί προκειμένου να δημιουργήσετε το νοητό τετράγωνο. Επίσης μπορεί να χρειαστεί να προσαρμόσετε το χρόνο ανάλογα με την απόδοση των μπαταριών. Όσο μειώνεται η απόδοση των μπαταριών μειώνεται και η ταχύτητα του ρομπότ.

Πλακίδια Εντολών Κυκλώματος







- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Κάντε ένα δυνατό θόρυβο κοντά στον αισθητήρα ή φυσήξτε απαλά τον αισθητήρα και παρατηρήστε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει.

Δραστηριότητα 8: Έναρξη Λειτουργίας Ρομπότ με βάση το επίπεδο Φωτισμού

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα προγραμματίσετε το ρομπότ έτσι ώστε **ΑΝ** το επίπεδο φωτισμού είναι πάνω από ένα επίπεδο (≥ 101) **ΤΟΤΕ** να εκτελεί μια ρουτίνα κινήσεων προσπαθώντας να σχηματίσει ένα τετράγωνο.

ΔΙΑΦΟΡΕΤΙΚΑ θα παραμένει ακίνητο.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας Φωτός LDR **AJS03**
- Καλώδια UTP (x1)
- Καλώδιο USB

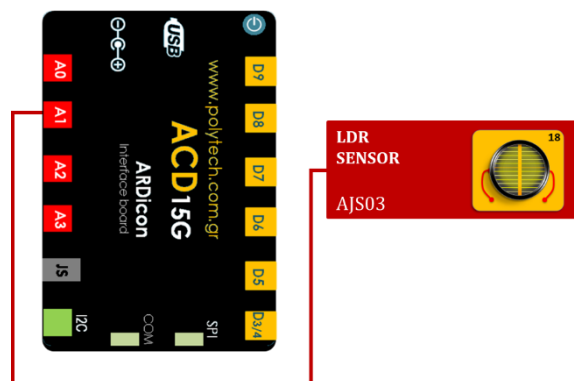
Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **AJS03** στην αναλογική θύρα 1 (A1) του ελεγκτή ARD:icon.

[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Ενεργοποιήστε το ρομπότ.

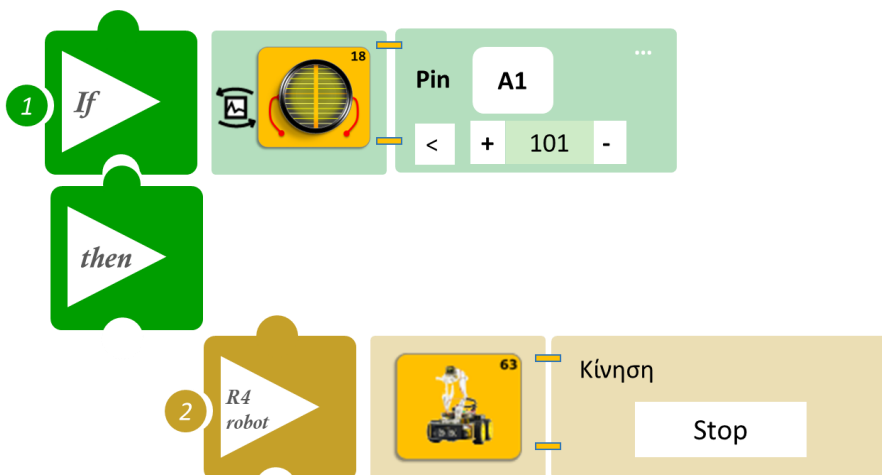
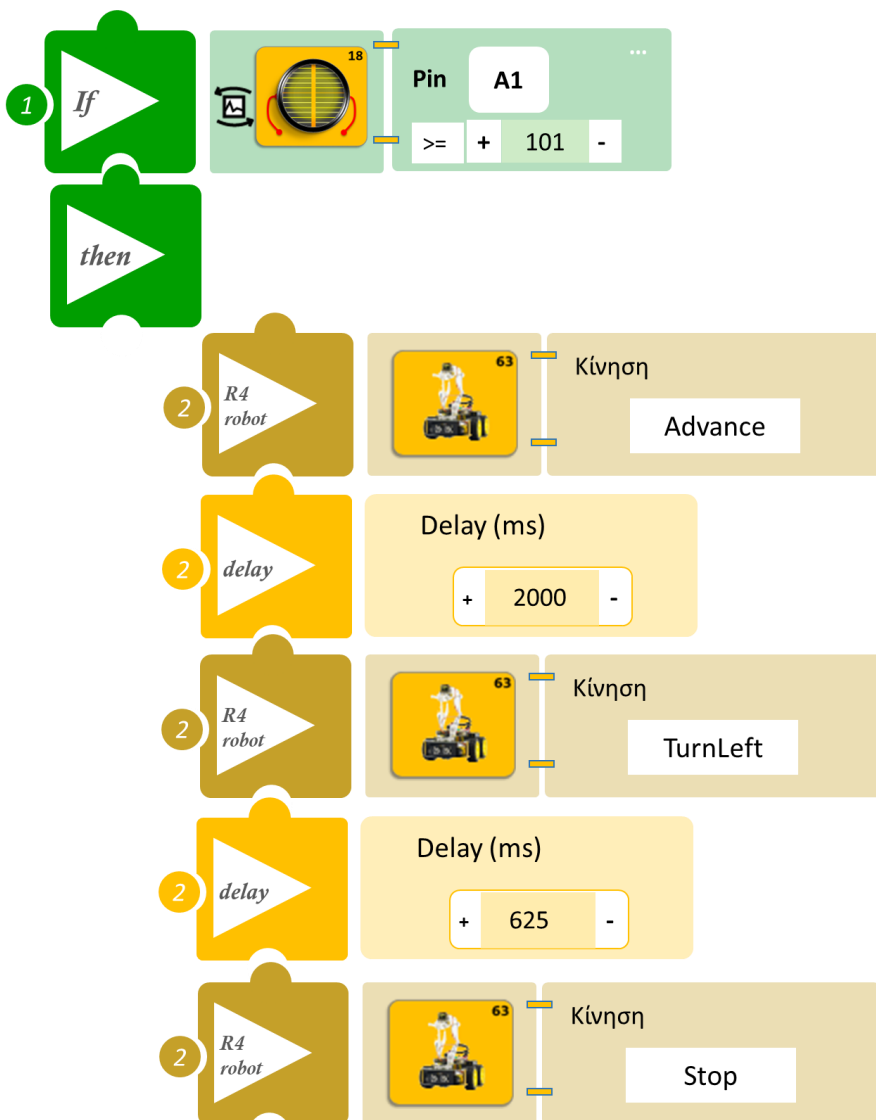


Προγραμματισμός Ρομπότ

<u>Γραμμή Εντολών</u>	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	“if/then”	AJS03	≥ 100 / A1
2 ^η	“R4 Κίνηση κινητήρων”	Robot	Advance
2 ^η	“delay”		1500ms
2 ^η	“R4 Κίνηση κινητήρων”	Robot	TurnLeft
2 ^η	“delay”		625ms
2 ^η	“R4 Κίνηση κινητήρων”	Robot	Stop
1 ^η	“if/then”	AJS03	< 100 / A1
2 ^η	“R4 Κίνηση κινητήρων”	Robot	Stop

- **Σημείωση:** Προσαρμόστε το χρόνο που στρίβει το ρομπότ (“delay”), αν χρειαστεί προκειμένου να δημιουργήσετε το νοητό τετράγωνο. Επίσης μπορεί να χρειαστεί να προσαρμόσετε το χρόνο ανάλογα με την απόδοση των μπαταριών. Όσο μειώνεται η απόδοση των μπαταριών μειώνεται και η ταχύτητα του ρομπότ.

Πλακίδια Εντολών Κυκλώματος



- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
Σημείωση: Κλείστε τα φώτα της αίθουσας
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί . Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Ανοίξτε τα φώτα της αίθουσας και παρατηρήστε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει.
- Κλείστε τα φώτα και παρατηρείστε το ρομπότ. Μπορεί να χρειαστεί να προσαρμόσετε τα όρια του αισθητήρα LDR προκειμένου να ανταποκρίνονται στα επίπεδα φωτός του δικού σας εργαστηρίου.
- Σε αυτή τη δραστηριότητα σε σύγκριση με την **δραστηριότητα 6: Έναρξη Λειτουργίας Ρομπότ με Ήχο** οι εντολές προγραμματισμού δεν είναι ίδιες παρόλο που και στις δύο το ρομπότ σχηματίζει τετράγωνο. Πιο συγκεκριμένα σε αυτή τη δραστηριότητα προγραμματίσατε μόνο την πρώτη γωνία του τετραγώνου. Μπορείτε να σκεφτείτε γιατί?

Δραστηριότητα 9: Έναρξη Λειτουργίας Ρομπότ με βάση το επίπεδο Φωτισμού II

Περιγραφή

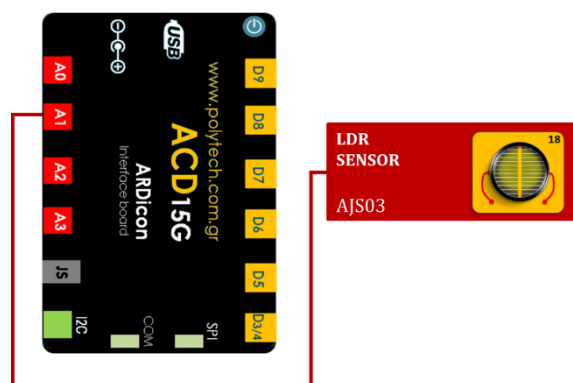
Στη συγκεκριμένη δραστηριότητα θα προγραμματίσετε το ρομπότ να κάνει ένα νοητό τετράγωνο, να συλλέγει και να τοποθετεί ένα αντικείμενο. Πιο συγκεκριμένα θα προγραμματίσετε το ρομπότ έτσι ώστε:

ΑΝ το επίπεδο φωτεινότητας του περιβάλλοντος είναι μεγαλύτερο ή ίσο με 100 **TOTE** το ρομπότ θα ξεκινάει να κινείται σε μία διαδρομή που σχηματίζει ένα νοητό τετράγωνο. Ταυτόχρονα θα συλλέγει ένα αντικείμενο που βρίσκεται στην πρώτη γωνία που σχηματίζει και να το αφήνει στη δεύτερη γωνία που σχηματίζει.

ΑΝ το επίπεδο φωτεινότητας του περιβάλλοντος είναι μικρότερο από 100 **TOTE** το ρομπότ θα παραμένει ακίνητο.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας Φωτός LDR **AJS03**
- Καλώδια UTP (x1)
- Καλώδιο USB



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **AJS03** στην αναλογική θύρα 1 (**A1**) του ελεγκτή ARD:icon.

[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Ενεργοποιήστε το ρομπότ.

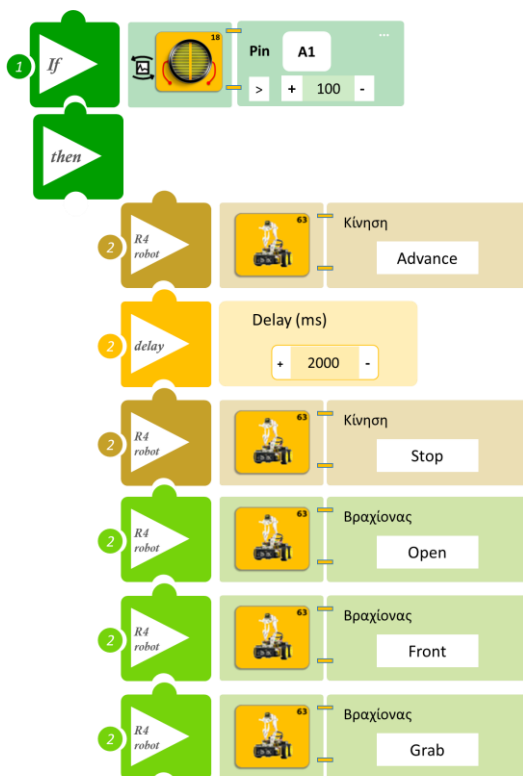
Προγραμματισμός Ρομπότ

<u>Γραμμή Εντολών</u>	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	"if/then"	AJS03	≥ 101 / A1
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		2000ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Open
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Front
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Grab
2 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnLeft
2 ^η	"delay"		625ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		2000ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnLeft
2 ^η	"delay"		625ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop

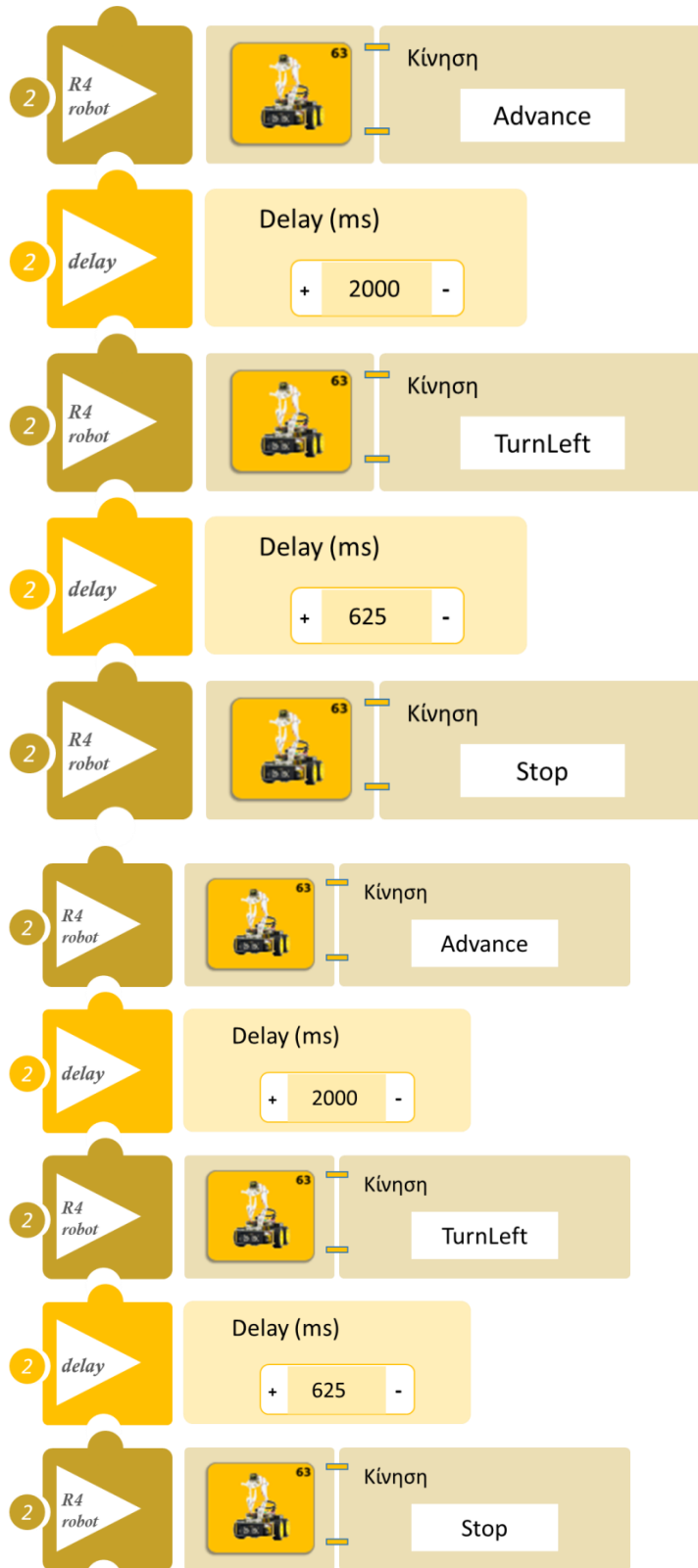
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		2000ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Open
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Front
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Home
2 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnLeft
2 ^η	"delay"		625ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		2000ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnLeft
2 ^η	"delay"		625ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
1 ^η	"if/then"	AJS03	< 101 / A1
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop

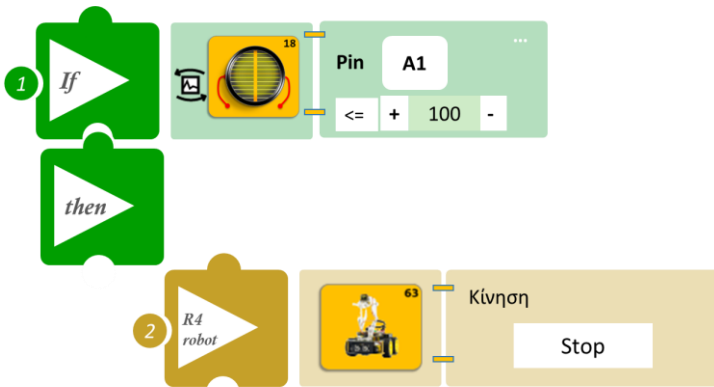
- **Σημείωση:** Προσαρμόστε το χρόνο που στρίβει το ρομπότ ("delay") , αν χρειαστεί προκειμένου να δημιουργήσετε το νοητό τετράγωνο. Επίσης μπορεί να χρειαστεί να προσαρμόσετε το χρόνο ανάλογα με την απόδοση των μπαταριών. Όσο μειώνεται η απόδοση των μπαταριών μειώνεται και η ταχύτητα του ρομπότ.

Πλακίδια Εντολών Κυκλώματος









- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
Σημείωση: Κλείστε τα φώτα της αίθουσας
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Ανοίξτε τα φώτα της αίθουσας και παρατηρήστε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει.
- Κλείστε τα φώτα και παρατηρείστε το ρομπότ. Μπορεί να χρειαστεί να προσαρμόσετε τα όρια του αισθητήρα LDR προκειμένου να ανταποκρίνονται στα επίπεδα φωτός του δικού σας εργαστηρίου.

Δραστηριότητα 10: Προγραμματισμός κινήσεων ρομπότ με τον αισθητήρα γωνίας

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα προγραμματίσετε το ρομπότ να αλλάζει την κατεύθυνση της κίνησης του σύμφωνα με τη γωνία που δίνεται στον αισθητήρα γωνίας. Πιο συγκεκριμένα θα προγραμματίσετε το ρομπότ έτσι ώστε

ΑΝ η γωνία περιστροφής του αισθητήρα είναι μικρότερη από 50° **TOTE** το ρομπότ να παραμένει ακίνητο.

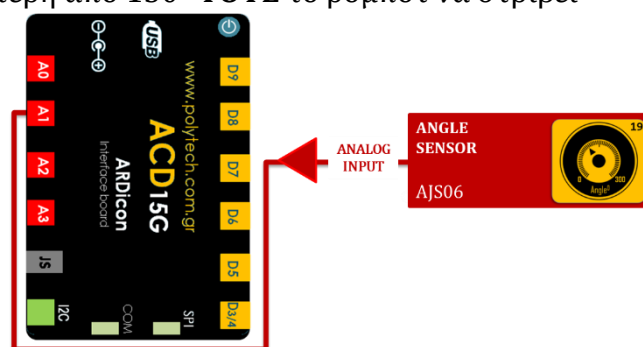
ΑΝ η γωνία περιστροφής του αισθητήρα είναι μεγαλύτερη από 50° και μικρότερη από 100° **TOTE** το ρομπότ να στρίβει αριστερά.

ΑΝ η γωνία περιστροφής του αισθητήρα είναι μεγαλύτερη από 100° και μικρότερη από 150° **TOTE** το ρομπότ να πηγαίνει ευθεία.

ΑΝ η γωνία περιστροφής του αισθητήρα είναι μεγαλύτερη από 150° **TOTE** το ρομπότ να στρίβει δεξιά.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας γωνίας **AJS06**
- Καλώδια UTP (x1)
- Καλώδιο USB



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **AJS06** στην αναλογική θύρα 1 (**A1**) του ελεγκτή ARD:icon. (Μπορείτε να προμηθευτείτε τον αισθητήρα γωνίας από το σετ S2.)

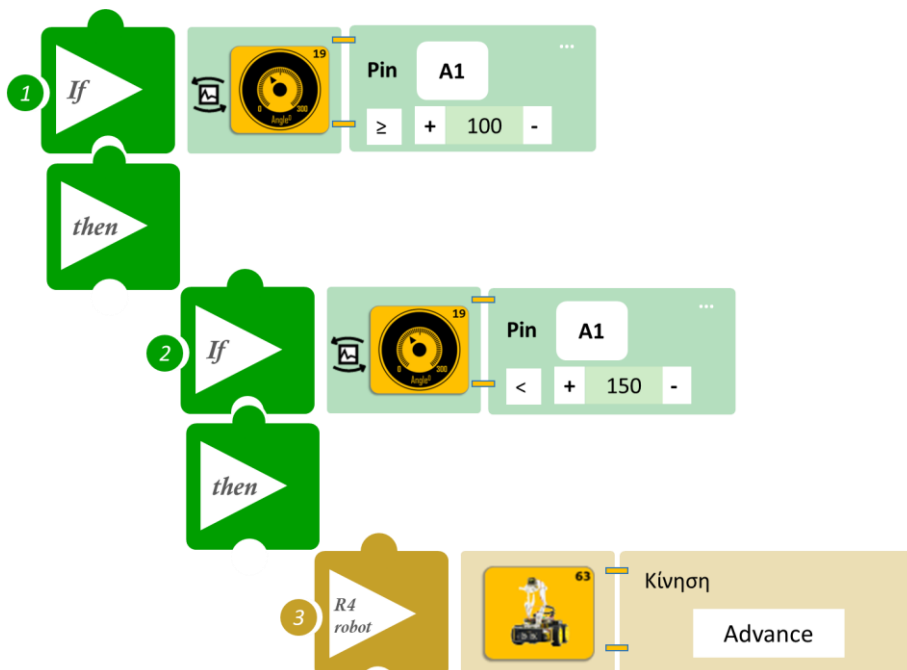
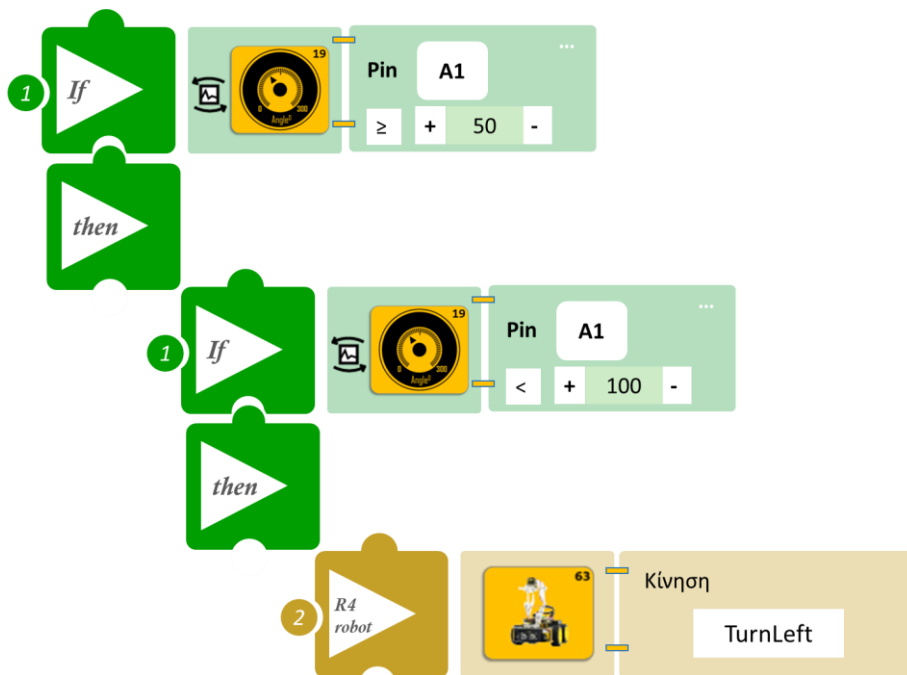
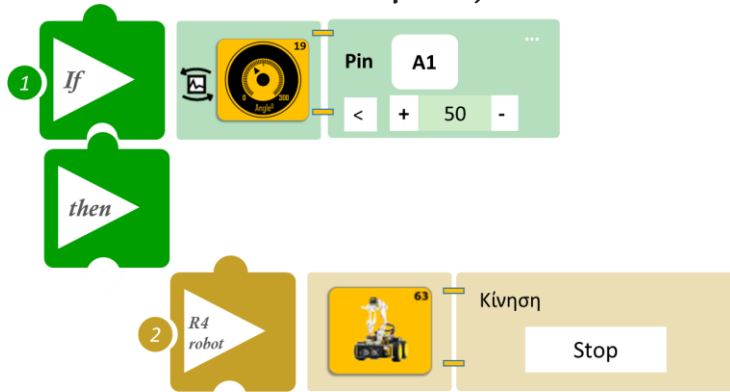
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

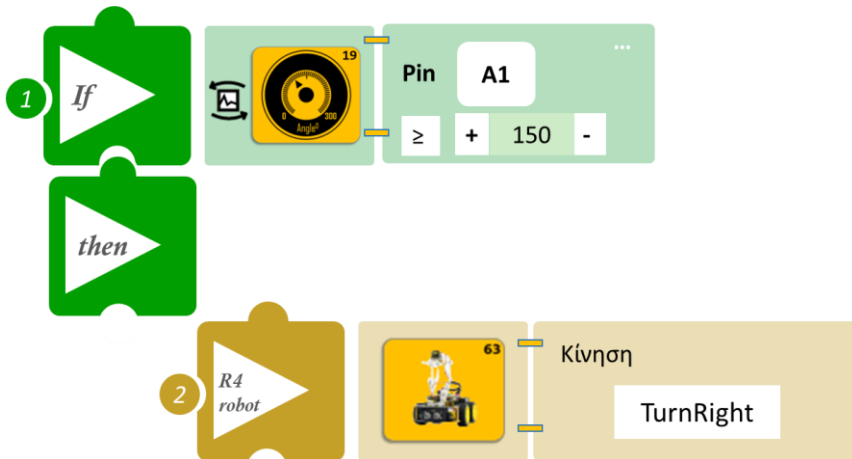
Ενεργοποιήστε το ρομπότ.

Προγραμματισμός Ρομπότ

Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	“if/then”	AJS06	< 50 / A1
2 ^η	“R4 Κίνηση κινητήρων”	Robot	Stop
1 ^η	“if/then”	AJS06	≥ 50 / A1
2 ^η	“if/then”	AJS06	< 100 / A1
3 ^η	“R4 Κίνηση κινητήρων”	Robot	TurnLeft
1 ^η	“if/then”	AJS06	≥ 100 / A1
2 ^η	“if/then”	AJS06	< 150 / A1
3 ^η	“R4 Κίνηση κινητήρων”	Robot	Advance
1 ^η	“if/then”	AJS06	≥ 150 / A1
2 ^η	“R4 Κίνηση κινητήρων”	Robot	TurnRight

Πλακίδια Εντολών Κυκλώματος





- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Περιστρέψτε τον αισθητήρα γωνίας και οδηγήστε το ρομπότ στην κατεύθυνση που επιθυμείτε.

Δραστηριότητα 11: Χειροκίνητος έλεγχος ταχύτητας ρομπότ

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα χρησιμοποιήσετε τον αισθητήρα αφής προκειμένου να ελγέξετε την ταχύτητα που κινείται το ρομπότ. Πιο συγκεκριμένα θα προγραμματίσετε το κύκλωμα σας έτσι ώστε:

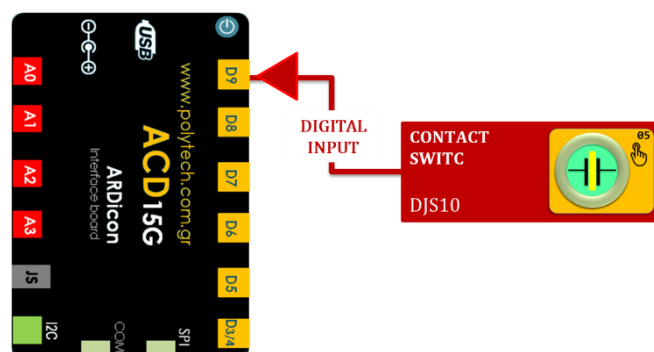
ΑΝ ο αισθητήρας αφής ανιχνεύει αφή **TOTE** το ρομπότ θα εκτελεί μία ρουτίνα κινήσεων με χαμηλή ταχύτητα.

ΑΝ ο αισθητήρας αφής δεν ανιχνεύει κίνηση **TOTE** το ρομπότ θα εκτελεί την ίδια ρουτίνα κινήσεων με υψηλή ταχύτητα.

Αυτή η ρουτίνα κινήσεων θα συνεχίσει να εκτελείται, σε χαμηλή ή υψηλή ταχύτητα ανάλογα με το σήμα που λαμβάνει ο αισθητήρας, μέχρι να απενεργοποιήσετε το ρομπότ.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας αφής **DJS10**
- Καλώδιο UTP (x1)
- Καλώδιο USB



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **DJS10** στην ψηφιακή θύρα 9 (**D9**) του ελεγκτή ARD:icon. (Μπορείτε να προμηθευτείτε τον αισθητήρα αφής από το σετ S2.)

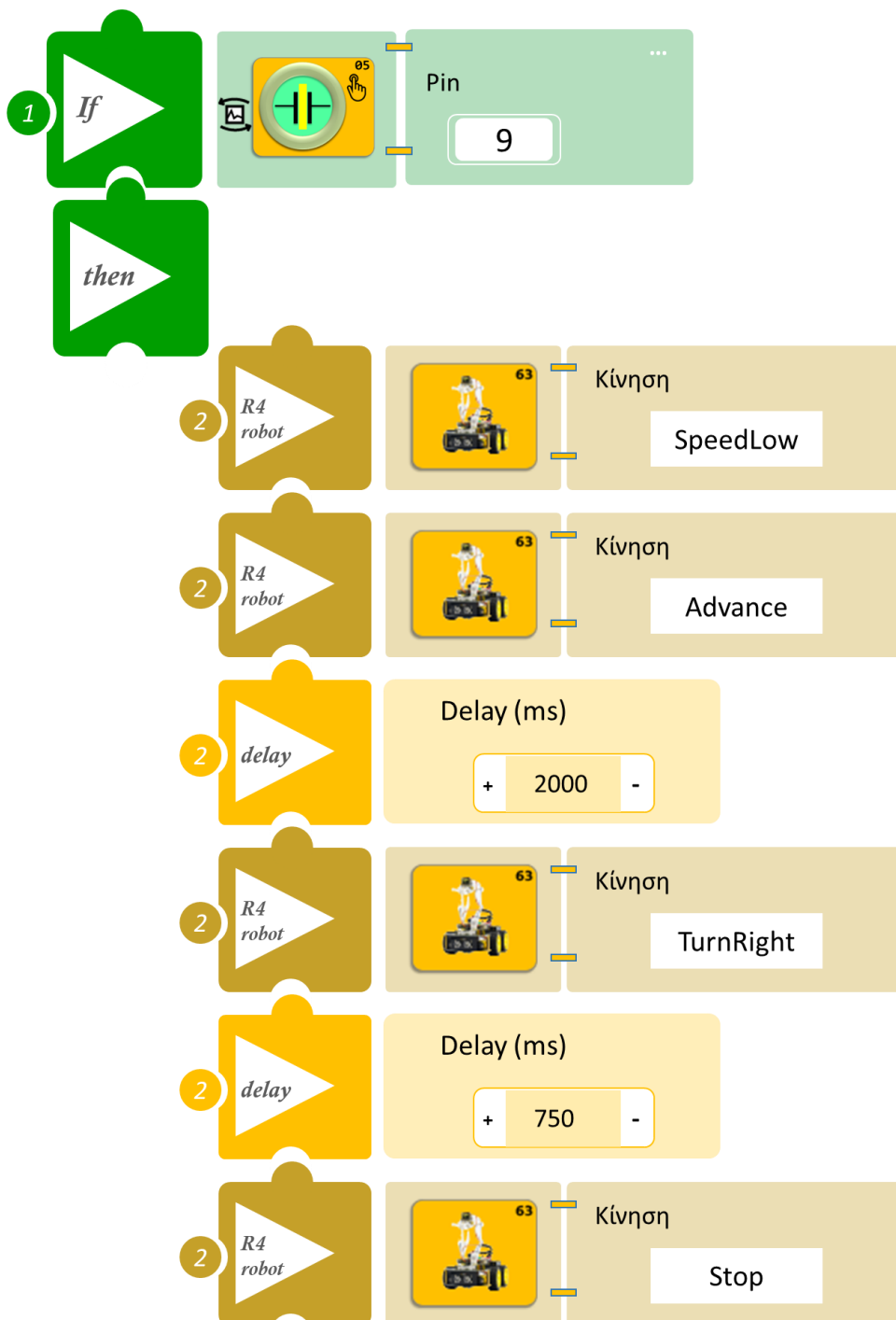
Ενεργοποιήστε το ρομπότ.

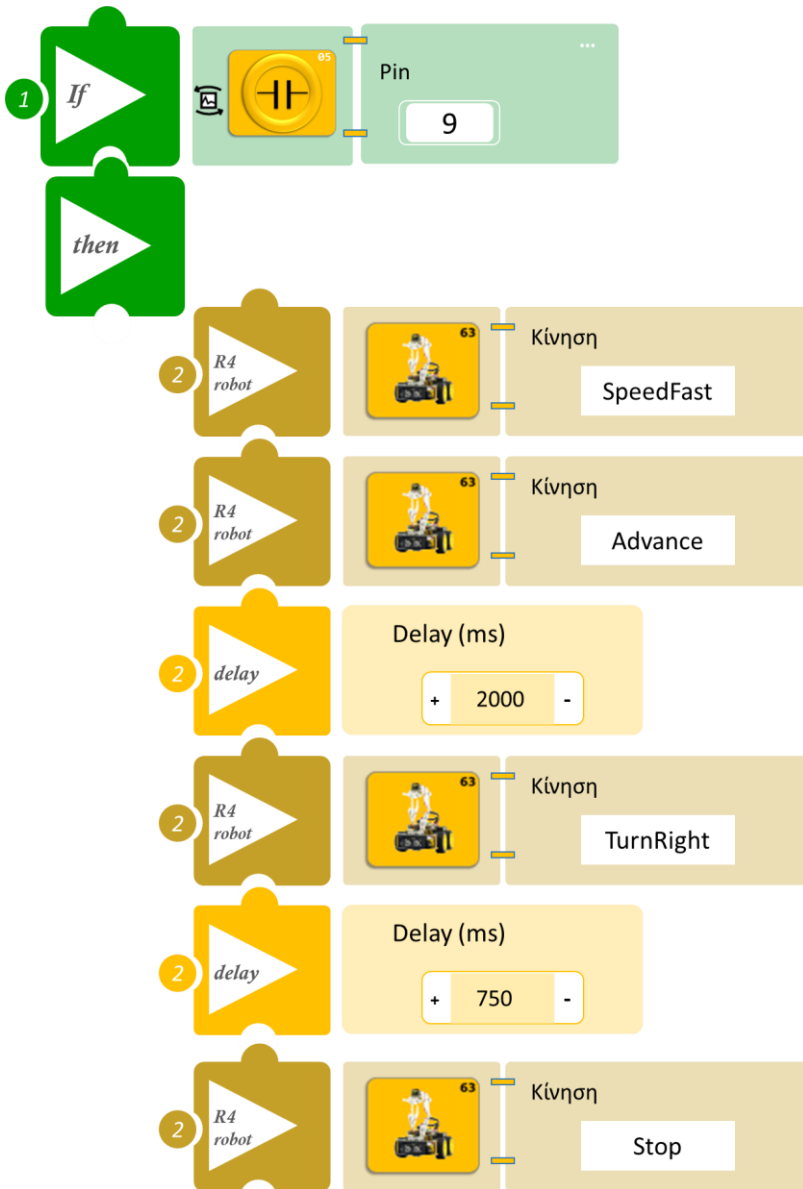
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Προγραμματισμός Ρομπότ

Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	"if/then"	DJS10	ON / D9
2 ^η	"R4 Κίνηση κινητήρων"	Robot	SpeedLow
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		2000ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnRight
2 ^η	"delay"		750ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
1 ^η	"if/then"	DJS10	OFF / D9
2 ^η	"R4 Κίνηση κινητήρων"	Robot	SpeedFast
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		2000ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnRight
2 ^η	"delay"		750ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop

Πλακίδια Εντολών Κυκλώματος





- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Παρατηρήσετε το ρομπότ να εκτελεί τις κινήσεις που προγραμματίσατε.
- Ακουμπήστε τον αισθητήρα αφής και παρατηρείστε αν η ταχύτητα του ρομπότ μειώνεται.

Δραστηριότητα 12: Χειροκίνητος έλεγχος ταχύτητας και συλλογής αντικειμένου

Περιγραφή

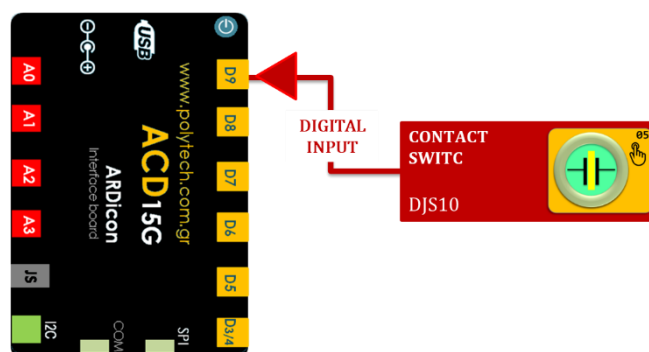
Στη συγκεκριμένη δραστηριότητα θα προγραμματίσετε τον αισθητήρα αφής να ελέγχει την ταχύτητα κίνησης του ρομπότ αλλά και τη διαδικασία συλλογής αντικειμένου. Πιο συγκεκριμένα θα προγραμματίσετε το ρομπότ έτσι ώστε:

ΑΝ ο αισθητήρας αφής δεν ανιχνεύει αφή **TOTE** το ρομπότ θα κινείται με μεγάλη ταχύτητα, θα πηγαίνει μπροστά θα στρίβει δεξιά, θα συλλέγει ένα αντικείμενο που βρίσκεται μπροστά του και θα το τοποθετεί αριστερά του και στη συνέχεια θα επιστρέφει στην αρχική του θέση.

ΑΝ ο αισθητήρας αφής ανιχνεύει αφή **TOTE** το ρομπότ θα κινείται με μικρή ταχύτητα, θα πηγαίνει μπροστά θα στρίβει αριστερά, θα συλλέγει ένα αντικείμενο που βρίσκεται μπροστά του και θα το τοποθετεί δεξιά του και στη συνέχεια θα επιστρέφει στην αρχική του θέση..

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας αφής **DJS10**
- Καλώδιο UTP (x1)
- Καλώδιο USB



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **DJS10** στην ψηφιακή θύρα 9 (**D9**) του ελεγκτή ARD:icon. (Μπορείτε να προμηθευτείτε τον αισθητήρα αφής από το σετ S2.)

Ενεργοποιήστε το ρομπότ.

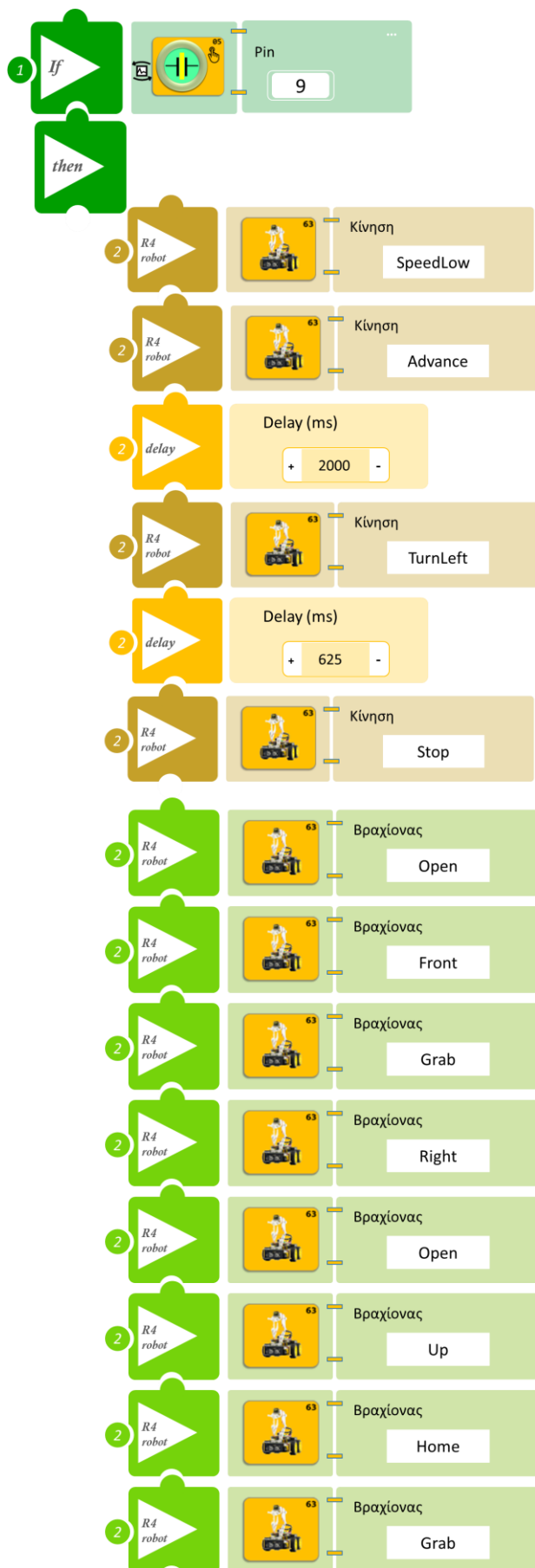
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

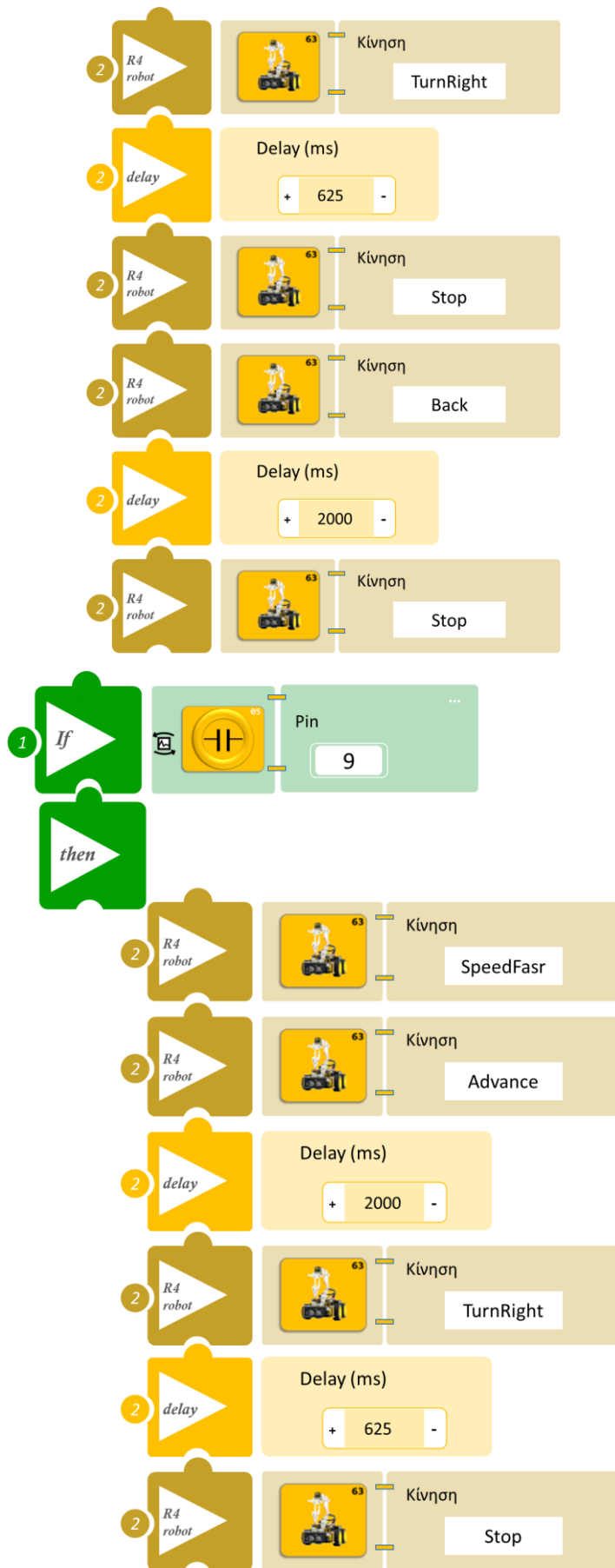
Προγραμματισμός Ρομπότ

Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	"if/then"	DJS10	ON / D9
2 ^η	"R4 Κίνηση κινητήρων"	Robot	SpeedLow
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		2000ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnRight
2 ^η	"delay"		625ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Open
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Front
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Grab
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Right

2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Open
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Up
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Home
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Grab
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	TurnLeft
2 ^η	“ <u>delay</u> ”		625ms
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	Stop
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	Back
2 ^η	“ <u>delay</u> ”		2000ms
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	Stop
1 ^η	“ <u>if/then</u> ”	DJS10	OFF / D9
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	SpeedFast
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	Advance
2 ^η	“ <u>delay</u> ”		2000ms
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	TurnRight
2 ^η	“ <u>delay</u> ”		625ms
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	Stop
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Open
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Front
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Grab
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Left
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Open
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Up
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Home
2 ^η	“ <u>R4 Κίνηση βραχίονα</u> ”	Robot	Grab
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	TurnLeft
2 ^η	“ <u>delay</u> ”		625ms
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	Stop
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	Back
2 ^η	“ <u>delay</u> ”		2000ms
2 ^η	“ <u>R4 Κίνηση κινητήρων</u> ”	Robot	Stop

Πλακίδια Εντολών Κυκλώματος







- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε τον κώδικα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί . Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Παρατηρήστε το ρομπότ να εκτελεί τις κινήσεις που προγραμματίσατε.
- Ακουμπήστε τον αισθητήρα αφής και παρατηρείστε τη ταχύτητα του ρομπότ να μειώνεται αλλά και συλλέγει το αντικείμενο από την πρώτη γωνία και να το τοποθετεί στην 3^η γωνία.

Δραστηριότητα 13: Προγραμματισμός διακόπτη έκτακτης ανάγκης

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα προσθέσετε έναν διακόπτη έκτακτης ανάγκης του κυκλώματος. Στη δραστηριότητα 8 είχατε προγραμματίσει το κύκλωμα σας έτσι ώστε:

ΑΝ το επίπεδο φωτισμού είναι πάνω από ένα επίπεδο (≥ 101) **TOTE** το ρομπότ να ξεκινάει να εκτελεί μια ρουτίνα κινήσεων προσπαθώντας να σχηματίσει ένα τετράγωνο.

ΑΝ το επίπεδο φωτισμού είναι κάτω από το επίπεδο που πριν ορίσατε (< 101) **TOTE** το ρομπότ να παραμένει ακίνητο.

Τώρα θα προγραμματίσετε το κύκλωμα σας έτσι ώστε:

ΓΙΑ ΟΣΟ το επίπεδο φωτισμού είναι πάνω από ένα επίπεδο (≥ 101) και

α) ΑΝ ο μαγνητικός αισθητήρας HALL δεν ανιχνεύει μαγνήτη **TOTE** το ρομπότ θα εκτελεί μια ρουτίνα κινήσεων προσπαθώντας να σχηματίσει ένα τετράγωνο.

β) ΑΝ ο μαγνητικός αισθητήρας HALL ανιχνεύει μαγνήτη **TOTE** το ρομπότ θα σταματάει να κινείται.

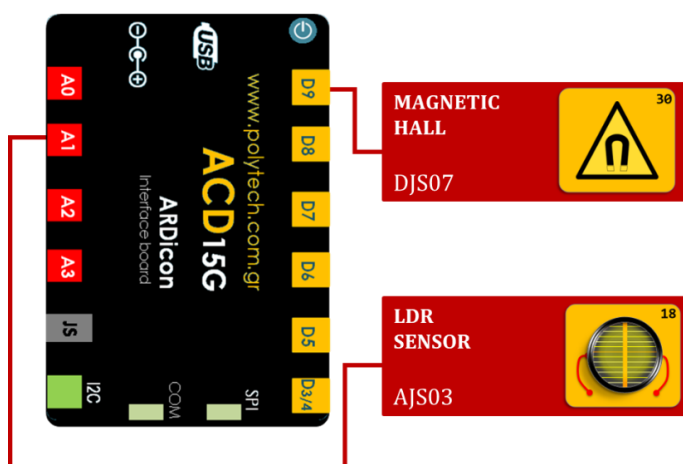
ΓΙΑ ΟΣΟ το επίπεδο φωτισμού είναι κάτω από το επίπεδο που πριν ορίσατε (< 101) **TOTE** το ρομπότ θα παραμένει ακίνητο.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon ACD15G
- Αισθητήρας LDR AJS03
- Αισθητήρας HALL DJS07
- Καλώδιο UTP (x1)
- Καλώδιο USB

Επιπλέον Υλικά

- Μαγνήτης



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **DJS07** στην ψηφιακή θύρα 9 (**D9**) του ελεγκτή ARD:icon. (Μπορείτε να προμηθευτείτε τον αισθητήρα Hall από το σετ S2.)

Συνδέστε τη συσκευή **AJS03** στην αναλογική θύρα 1 (**A1**) του ελεγκτή ARD:icon.

Ενεργοποιήστε το ρομπότ.

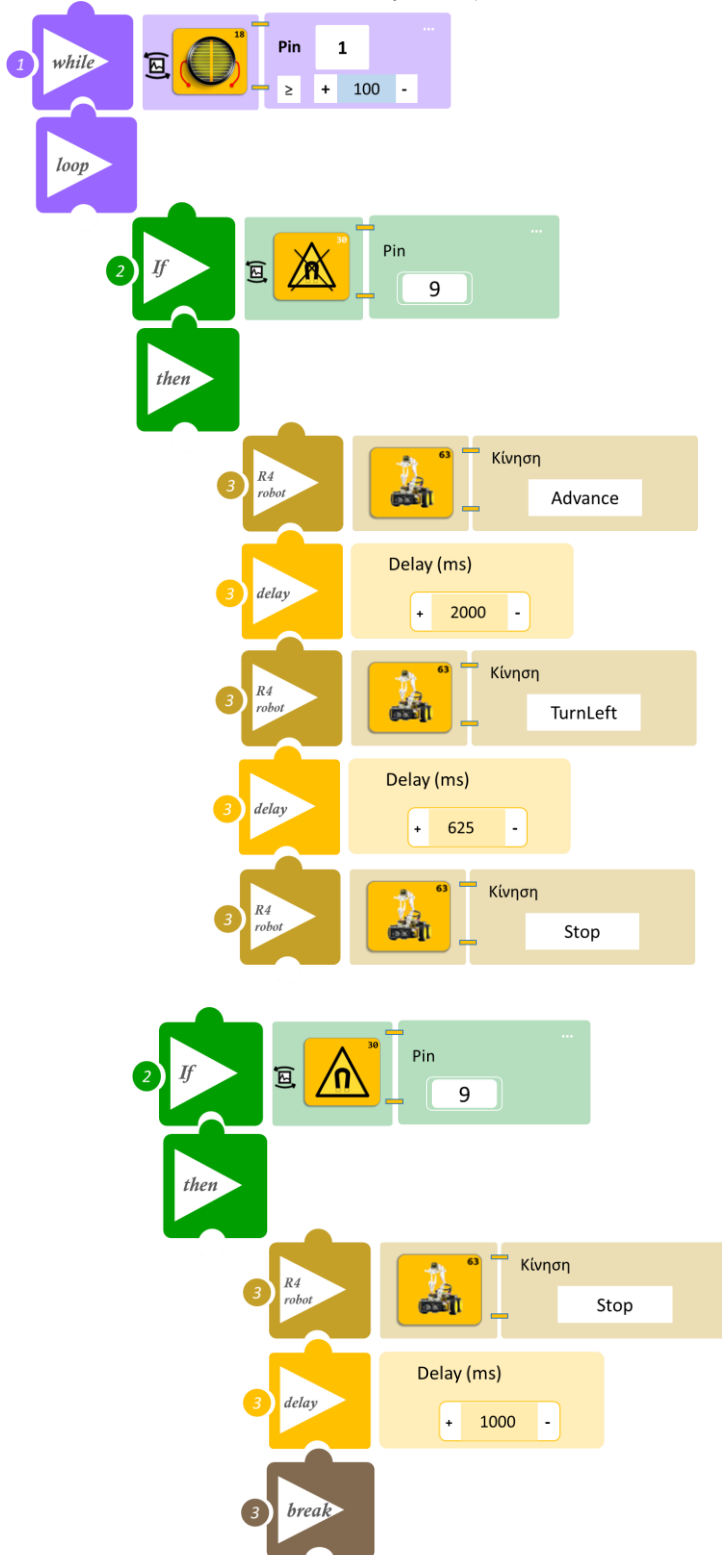
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

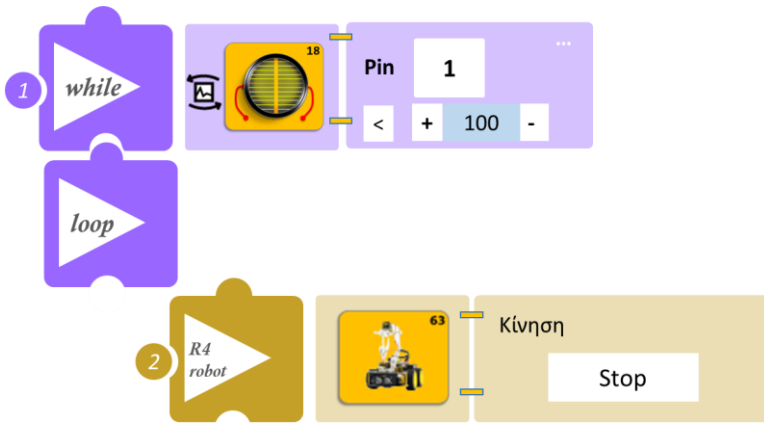
Προγραμματισμός Ρομπότ

Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	"while"	AJS03	≥ 100 / A1
2 ^η	"if"	DJS07	OFF/9
3 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
3 ^η	"delay"		2000ms
3 ^η	"R4 Κίνηση κινητήρων"	Robot	TurnLeft
3 ^η	"delay"		625ms
3 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
2 ^η	"if"	DJS07	ON/9
3 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop

3 ^η	" <u>delay</u> "		1000ms
3 ^η	" <u>break</u> "		
1 ^η	" <u>while</u> "	AJS03	< 101 / A1
2 ^η	" <u>R4 Κίνηση κινητήρων</u> "	Robot	Stop

Πλακίδια Εντολών Κυκλώματος





- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
Σημείωση: Κλείστε τα φώτα της αίθουσας
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Ανοίξτε τα φώτα της αίθουσας και παρατηρήστε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει. Εναλλακτικά, χρησιμοποιήστε τον φακό του κινητού σας και φωτίστε πάνω στον αισθητήρα.
- Πλησιάστε το μαγνήτη κοντά στον αισθητήρα Hall και παρατηρείστε το ρομπότ.
- Κλείστε τα φώτα ή τον φακό του κινητού σας και παρατηρείστε το ρομπότ.
- Προσαρμόστε το χρόνο που στρίβει το ρομπότ ("delay"), αν χρειαστεί προκειμένου να δημιουργήσετε το νοητό τετράγωνο. Επίσης μπορεί να χρειαστεί να προσαρμόσετε το χρόνο ανάλογα με την απόδοση των μπαταριών. Όσο μειώνεται η απόδοση των μπαταριών μειώνεται και η ταχύτητα του ρομπότ.

Δραστηριότητα 14: Προγραμματισμός διακόπτη έκτακτης ανάγκης II

Περιγραφή

Σε αυτή τη δραστηριότητα θα προγραμματίσετε τη Δραστηριότητα 13: Προγραμματισμός διακόπτη έκτακτης ανάγκης χρησιμοποιώντας διαφορετική ρουτίνα εντολών.

Τώρα θα προγραμματίσετε το κύκλωμα σας έτσι ώστε:

ΑΝ το επίπεδο φωτισμού είναι κάτω από το επίπεδο που πριν ορίσατε (<100) **TOTE** το ρομπότ να παραμένει ακίνητο.

ΔΙΑΦΟΡΕΤΙΚΑ:

Α) **ΑΝ** ο μαγνητικός αισθητήρας HALL δεν ανιχνεύει μαγνητικό πεδίο **TOTE** το ρομπότ να εκτελεί μια ρουτίνα κινήσεων προσπαθώντας να σχηματίσει ένα τετράγωνο.

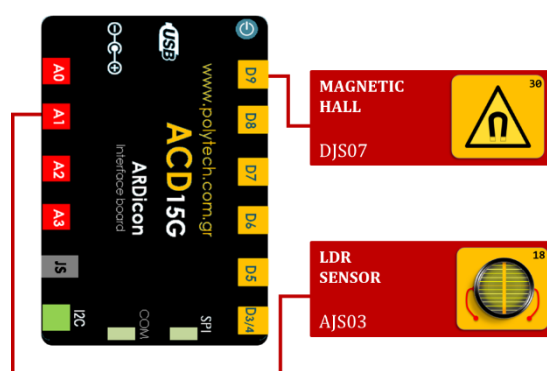
β) **ΑΝ** ο μαγνητικός αισθητήρας HALL ανιχνεύει μαγνητικό πεδίο **TOTE** το ρομπότ να σταματάει να κινείται.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας LDR **AJS03**
- Αισθητήρας HALL **DJS07**
- Καλώδιο UTP (x1)
- Καλώδιο USB

Επιπλέον Υλικά

- Μαγνήτης



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **DJS07** στην ψηφιακή θύρα 9 (**D9**) του ελεγκτή ARD:icon.

Συνδέστε τη συσκευή **AJS03** στην αναλογική θύρα 1 (**A1**) του ελεγκτή ARD:icon.

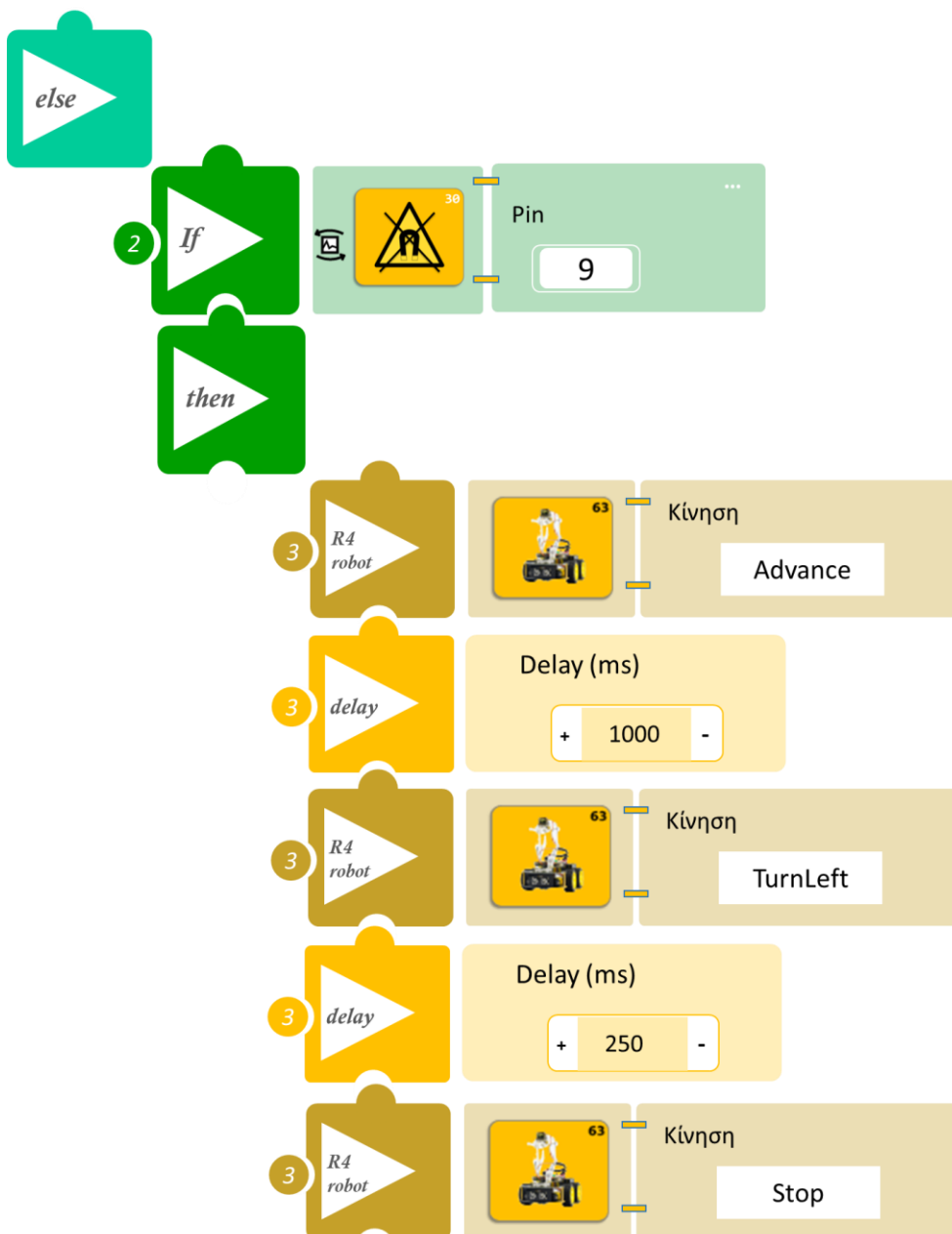
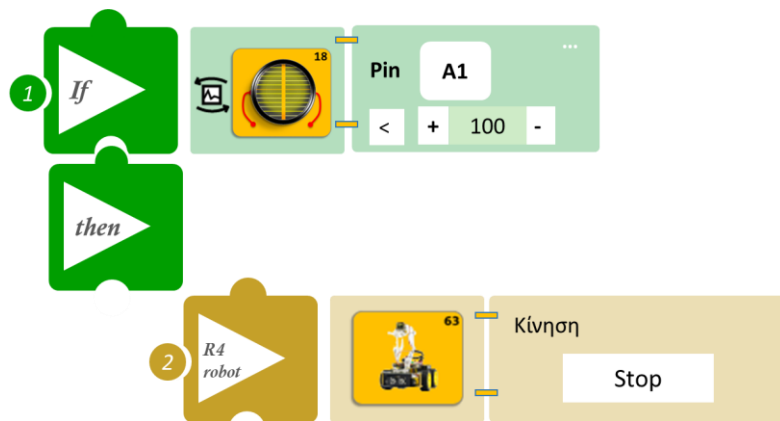
Ενεργοποιήστε το ρομπότ.

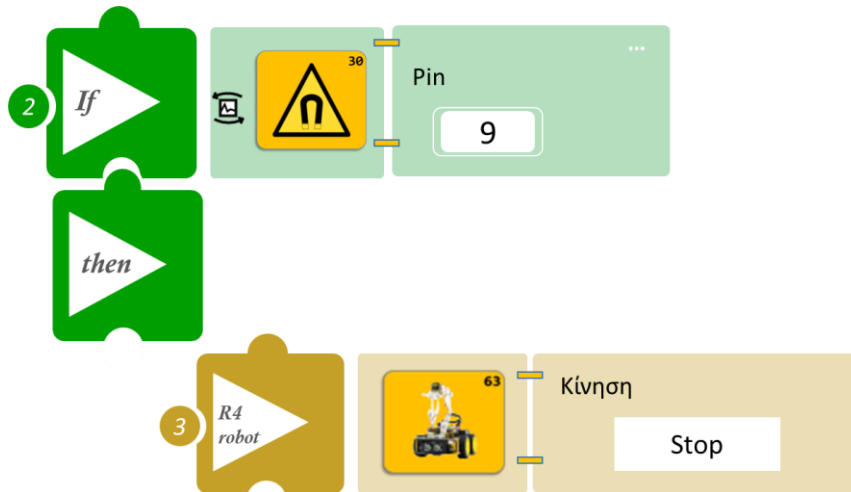
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Προγραμματισμός Ρομπότ

Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	" if "	AJS03	<100 / A1
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
1 ^η	" else "		
2 ^η	" if "	DJS07	OFF / D9
3 ^η	" R4 Κίνηση κινητήρων "	Robot	Advance
3 ^η	" delay "		2000ms
3 ^η	" R4 Κίνηση κινητήρων "	Robot	TurnLeft
3 ^η	" delay "		625ms
3 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
2 ^η	" if "	DJS07	ON / D9
3 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop

Πλακίδια Εντολών Κυκλώματος





- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
Σημείωση: Κλείστε τα φώτα της αίθουσας
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Ανοίξτε τα φώτα της αίθουσας και παρατηρήσετε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει. Εναλλακτικά, χρησιμοποιήστε τον φακό του κινητού σας και φωτίστε πάνω στον αισθητήρα.
- Πλησιάστε το μαγνήτη κοντά στον αισθητήρα Hall και παρατηρείστε το ρομπότ.
- Κλείστε τα φώτα ή τον φακό του κινητού σας και παρατηρείστε το ρομπότ.
- Προσαρμόστε το χρόνο που στρίβει το ρομπότ ("delay"), αν χρειαστεί προκειμένου να δημιουργήσετε το νοητό τετράγωνο. Επίσης μπορεί να χρειαστεί να προσαρμόσετε το χρόνο ανάλογα με την απόδοση των μπαταριών. Όσο μειώνεται η απόδοση των μπαταριών μειώνεται και η ταχύτητα του ρομπότ.

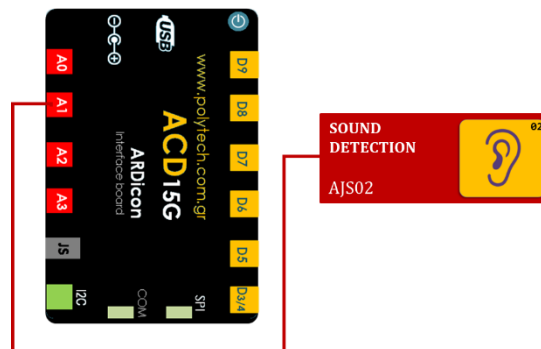
Δραστηριότητα 15: Έναρξη Λειτουργίας Ρομπότ με Ήχο

Περιγραφή

Στη συγκεκριμένη δραστηριότητα σε συνέχεια της δραστηριότητας 6: Έναρξη Λειτουργίας Ρομπότ με Ήχο, θα προγραμματίσετε το ρομπότ έτσι ώστε **ΑΝ** ο ήχος είναι πάνω από ένα επίπεδο **TOTE** να εκτελεί μια ρουτίνα κινήσεων προσπαθώντας να σχηματίσει ένα τετράγωνο χρησιμοποιώντας διαφορετικό προγραμματισμό. Συγκεκριμένα θα χρησιμοποιήσετε την εντολή “for”, προγραμματίζοντας το ρομπότ να επαναλαμβάνει 4 φορές τη ρουτίνα κίνηση μπροστά και δεξιά. **ΔΙΑΦΟΡΕΤΙΚΑ** θα παραμένει ακίνητο.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας Ήχου **AJS02**
- Καλώδια UTP (x1)
- Καλώδιο USB



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **AJS02** στην αναλογική θύρα 1 (**A1**) του ελεγκτή ARD:icon.

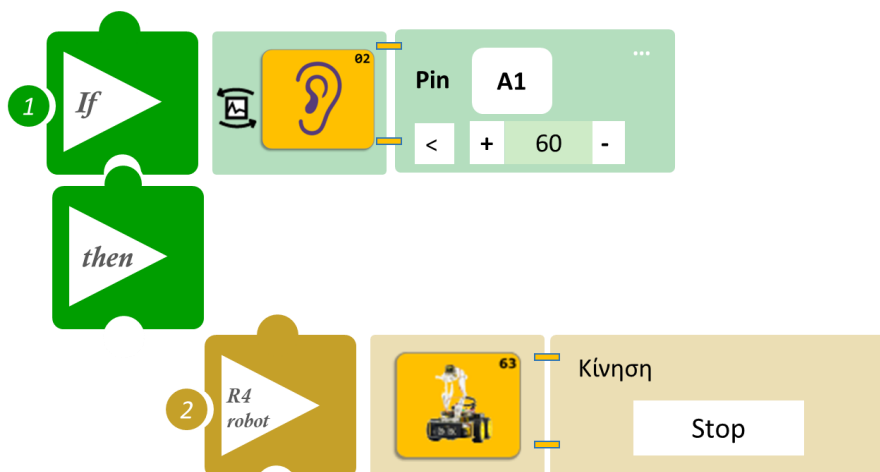
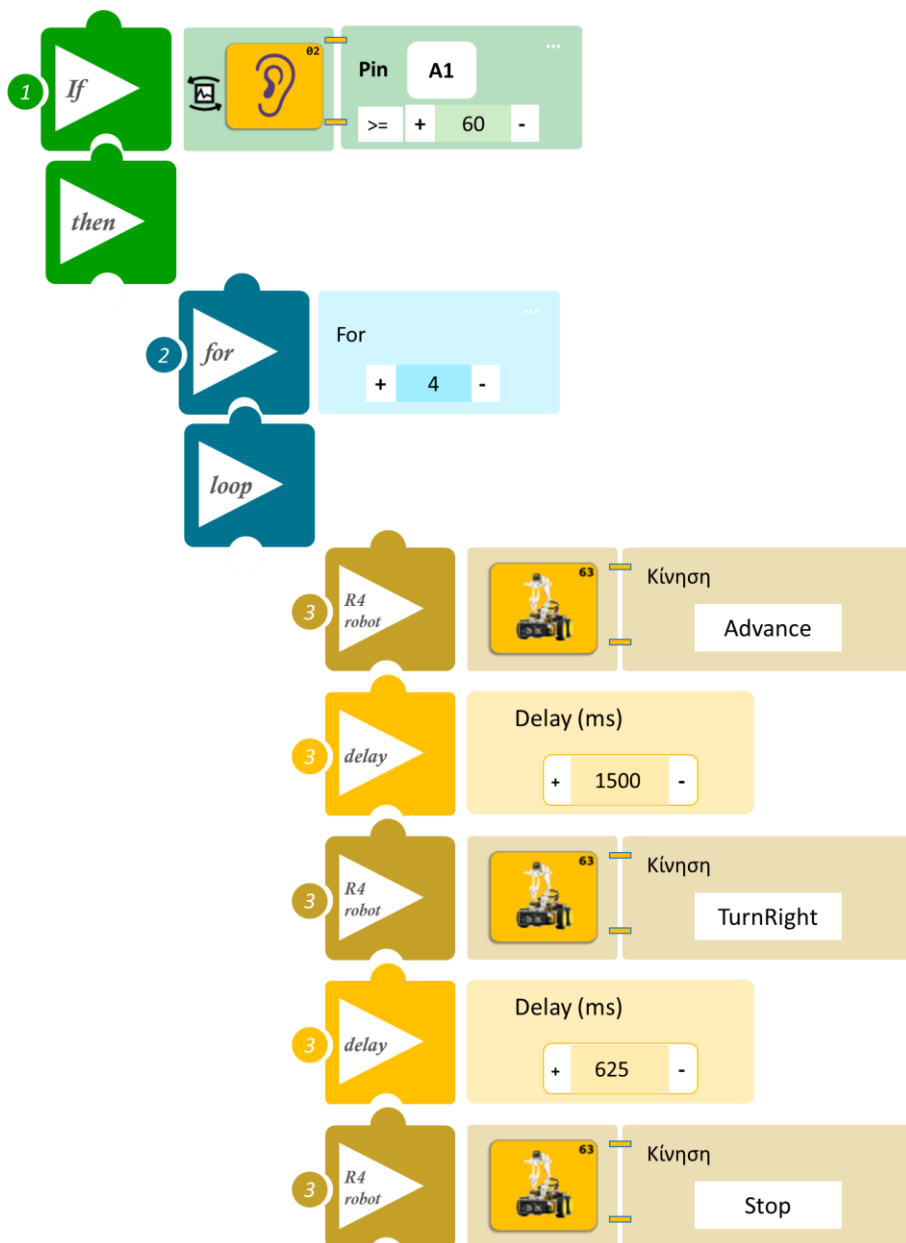
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Ενεργοποιήστε το ρομπότ.

Προγραμματισμός Ρομπότ

Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	“if/then”	AJS02	≥ 60 / A1
2 ^η	“for”		4 times
3 ^η	“R4 Κίνηση κινητήρων”	Robot	Advance
3 ^η	“delay”		1500ms
3 ^η	“R4 Κίνηση κινητήρων”	Robot	TurnRight
3 ^η	“delay”		625ms
3 ^η	“R4 Κίνηση κινητήρων”	Robot	Stop
1 ^η	“if/then”	AJS02	< 60 / A1
2 ^η	“R4 Κίνηση κινητήρων”	Robot	Stop

Πλακίδια Εντολών Κυκλώματος



- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί . Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Κάντε ένα δυνατό θόρυβο κοντά στον αισθητήρα ή φυσήξτε απαλά τον αισθητήρα και παρατηρήσετε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει.
- Προσαρμόστε το χρόνο που στρίβει το ρομπότ ("delay"), αν χρειαστεί προκειμένου να δημιουργήσετε το νοητό τετράγωνο. Επίσης μπορεί να χρειαστεί να προσαρμόσετε το χρόνο ανάλογα με την απόδοση των μπαταριών. Όσο μειώνεται η απόδοση των μπαταριών μειώνεται και η ταχύτητα του ρομπότ.

Δραστηριότητα 16: Έναρξη Λειτουργίας Ρομπότ με Φως και διακοπή με ήχο

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα προγραμματίσετε το ρομπότ έτσι ώστε:

ΓΙΑ ΟΣΟ το επίπεδο φωτισμού είναι πάνω από ένα επίπεδο (≥ 101) και

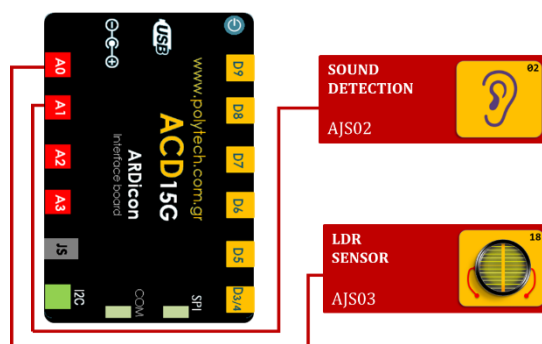
α) ΑΝ το επίπεδο ήχου είναι μικρότερο ή ίσο από ένα επίπεδο (≤ 60) **TOTE** το ρομπότ θα εκτελεί μια ρουτίνα κινήσεων προσπαθώντας να σχηματίσει ένα τετράγωνο.

β) ΑΝ το επίπεδο ήχου είναι πάνω από το επίπεδο που πριν ορίσατε (>60) **TOTE** το ρομπότ θα παραμένει ακίνητο για 5 δευτερόλεπτα.

ΓΙΑ ΟΣΟ το επίπεδο φωτισμού είναι κάτω από το επίπεδο που πριν ορίσατε (<101) **TOTE** το ρομπότ θα παραμένει ακίνητο.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας Φωτός LDR **AJS03**
- Αισθητήρας Ήχου **AJS02**
- Καλώδια UTP (x2)
- Καλώδιο USB



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **AJS03** στην αναλογική θύρα 0 (**A0**) του ελεγκτή ARD:icon.

Συνδέστε τη συσκευή **AJS02** στην αναλογική θύρα 1 (**A1**) του ελεγκτή ARD:icon.

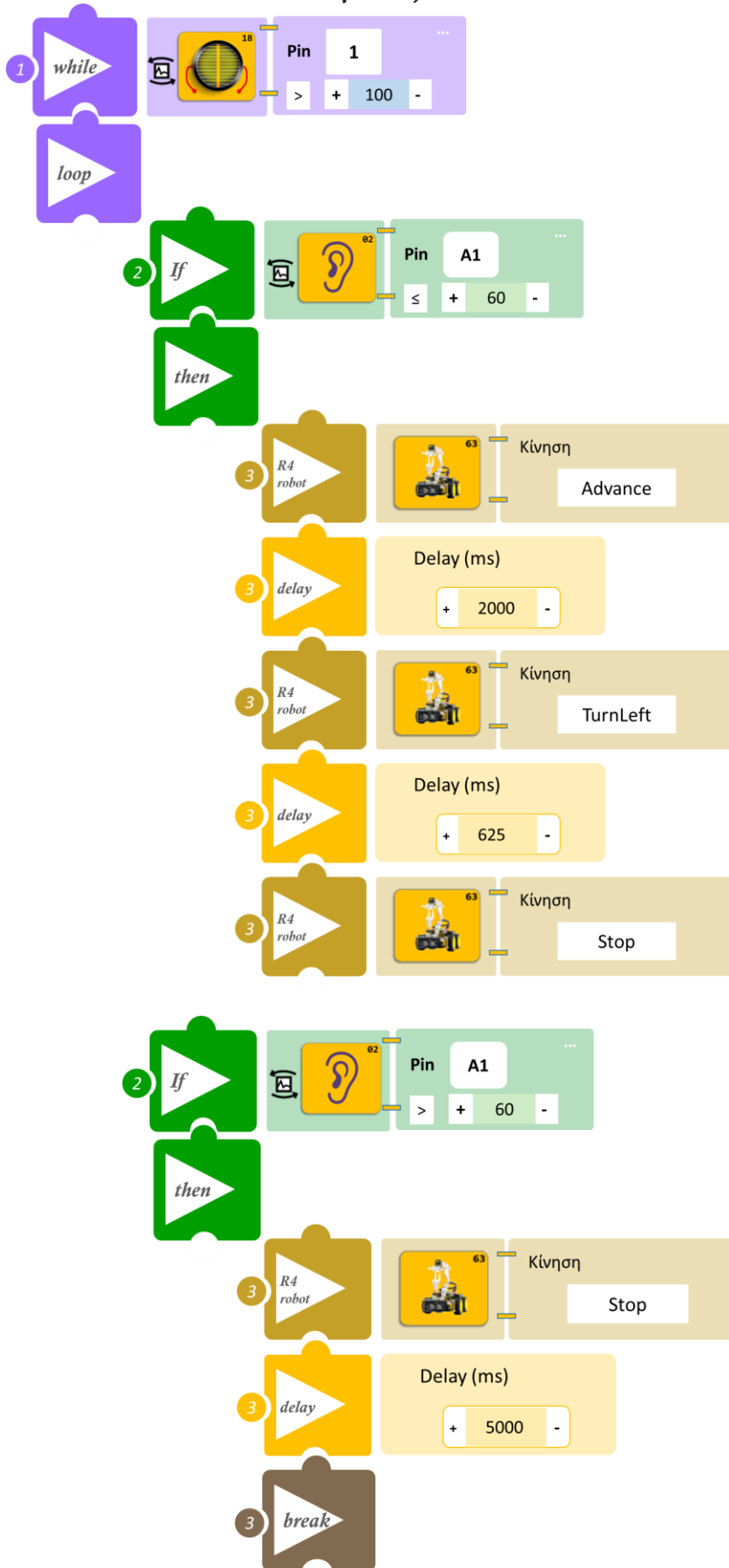
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

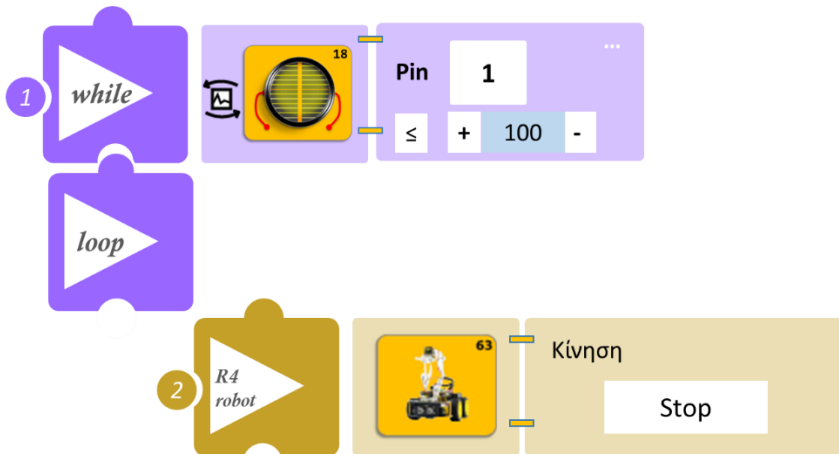
Ενεργοποιήστε το ρομπότ.

Προγραμματισμός Ρομπότ

Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	" while "	AJS03	>100 / A0
2 ^η	" if "	AJS02	≤ 60 / A1
3 ^η	" R4 Κίνηση κινητήρων "	Robot	Advance
3 ^η	" delay "		2000ms
3 ^η	" R4 Κίνηση κινητήρων "	Robot	TurnLeft
3 ^η	" delay "		625ms
3 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
2 ^η	" if "	AJS02	60 / A1
3 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
3 ^η	" delay "		5000ms
3 ^η	" break "		
1 ^η	" while "	AJS03	≤ 101 / A0
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop

Πλακίδια Εντολών Κυκλώματος





- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
Σημείωση: Κλείστε τα φώτα της αίθουσας
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Ανοίξτε τα φώτα της αίθουσας και παρατηρήστε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει. Κάντε ένα δυνατό θόρυβο κοντά στον αισθητήρα ή φυσήξτε απαλά τον αισθητήρα και παρατηρείστε το ρομπότ
- Κλείστε τα φώτα και παρατηρείστε το ρομπότ. Μπορεί να χρειαστεί να προσαρμόσετε τα όρια του αισθητήρα LDR προκειμένου να ανταποκρίνονται στα επίπεδα φωτός του δικού σας εργαστηρίου.
- Προσαρμόστε το χρόνο που στρίβει το ρομπότ ("delay"), αν χρειαστεί, προκειμένου να δημιουργήσετε το νοητό τετράγωνο. Επίσης μπορεί να χρειαστεί να προσαρμόσετε το χρόνο ανάλογα με την απόδοση των μπαταριών. Όσο μειώνεται η απόδοση των μπαταριών μειώνεται και η ταχύτητα του ρομπότ.

Δραστηριότητα 17: Προγραμματισμός συλλογής αντικειμένου ανάλογα με το χρώμα του αντικειμένου

Περιγραφή

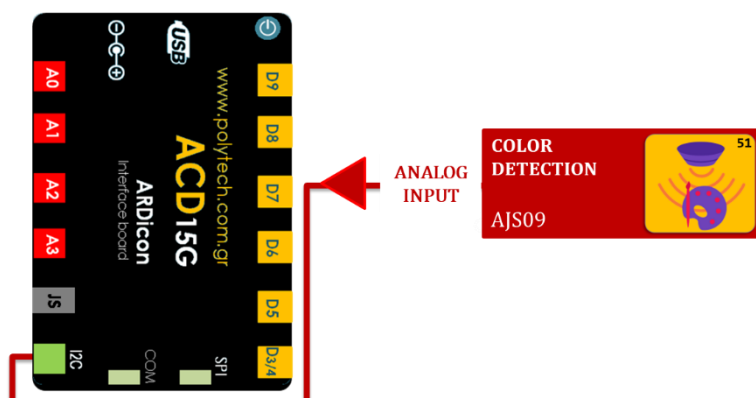
Σε αυτή τη δραστηριότητα θα προγραμματίσετε το κύκλωμα σας έτσι ώστε:

ΑΝ ο αισθητήρας χρώματος ανιχνεύσει κάποιο αντικείμενο με χρώμα μπλε **ΤΟΤΕ** το ρομπότ να κινείται προς τα μπροστά στη θέση συλλογής αντικειμένου, να συλλέγει το αντικείμενο (θέση μπροστά), να το τοποθετεί δεξιά και να επιστρέφει πίσω στην αρχική του θέση.

ΑΝ ο αισθητήρας χρώματος ανιχνεύσει κάποιο αντικείμενο με χρώμα κόκκινο **ΤΟΤΕ** το ρομπότ να κινείται προς τα μπροστά στη θέση συλλογής αντικειμένου, να συλλέγει το αντικείμενο (θέση μπροστά), να το τοποθετεί αριστερά και να επιστρέφει πίσω στην αρχική του θέση.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Αισθητήρας ανίχνευσης χρώματος **AJS09**
- Καλώδιο UTP (x1)
- Καλώδιο USB



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **AJS09** στη θύρα **I2C** του ελεγκτή ARD:icon.

Ενεργοποιήστε το ρομπότ.

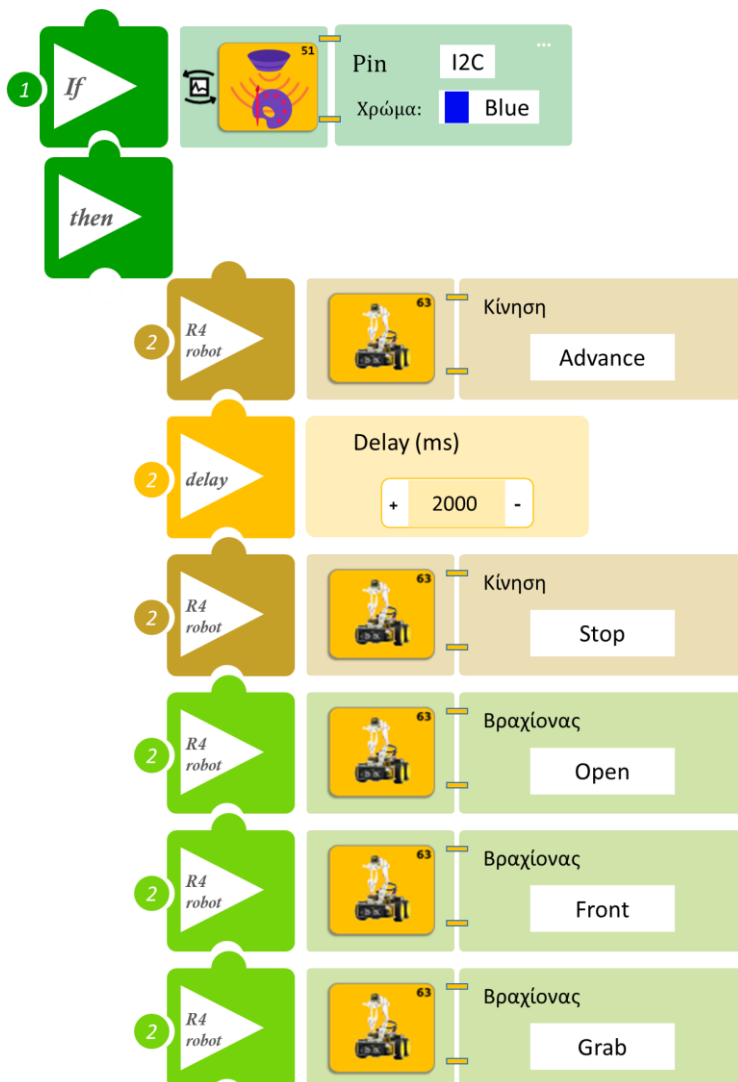
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

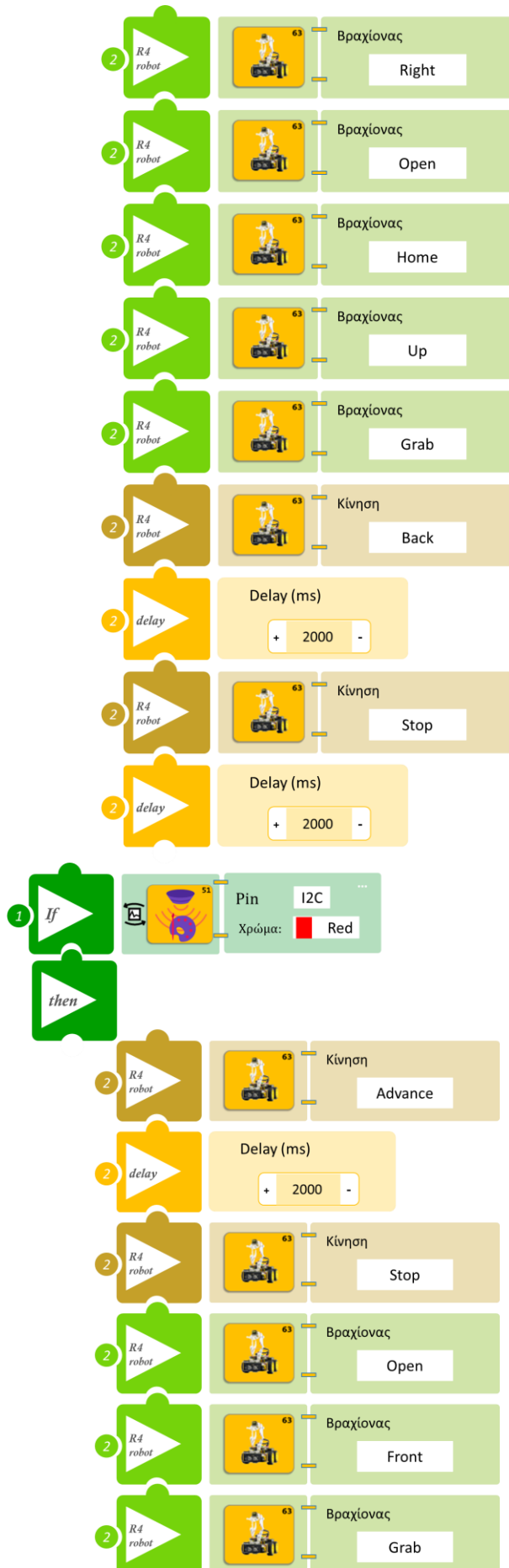
Προγραμματισμός Ρομπότ

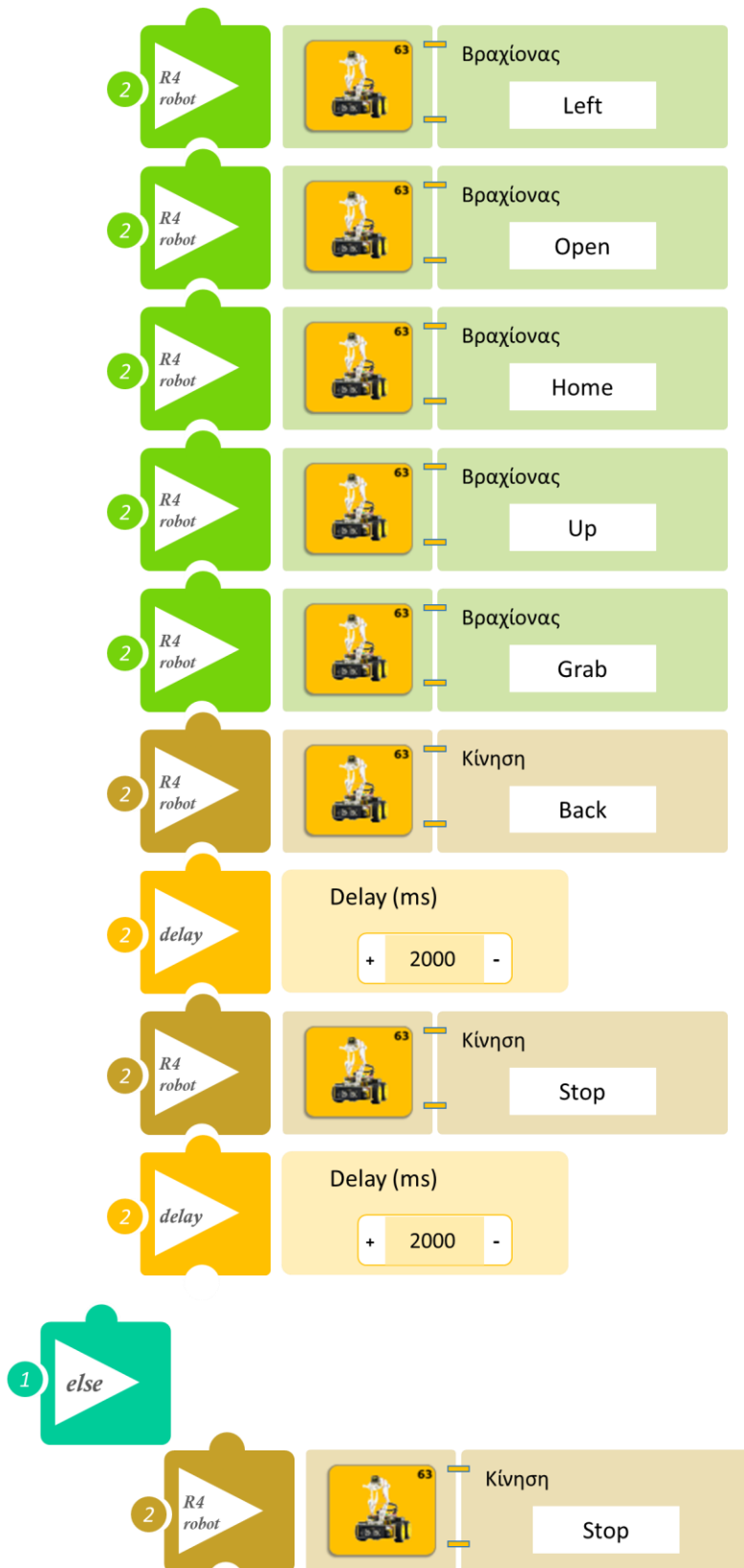
Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	" if/then "	AJS09	Blue / I2C
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Advance
2 ^η	" delay "		2000ms
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Open
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Front
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Grab
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Right
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Open
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Home
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Up
2 ^η	" R4 Κίνηση βραχίονα "	Robot	Grab
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Back
2 ^η	" delay "		2000ms
2 ^η	" R4 Κίνηση κινητήρων "	Robot	Stop
2 ^η	" delay "		2000ms
1 ^η	" if/then "	AJS09	Red / I2C

2 ^η	"R4 Κίνηση κινητήρων"	Robot	Advance
2 ^η	"delay"		2000ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Open
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Front
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Grab
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Left
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Open
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Home
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Up
2 ^η	"R4 Κίνηση βραχίονα"	Robot	Grab
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Back
2 ^η	"delay"		2000ms
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop
2 ^η	"delay"		2000ms
1 ^η	"else"		
2 ^η	"R4 Κίνηση κινητήρων"	Robot	Stop

Πλακίδια Εντολών Κυκλώματος







- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα..

ΑΠΟΤΕΛΕΣΜΑ

- Τοποθετήστε ένα μπλε αντικείμενο κοντά στον αισθητήρα για να δείτε τις κινήσεις του ρομπότ και να οριοθετήσετε το σημείο συλλογής αντικειμένων. Επαναλάβετε για ένα κόκκινο αντικείμενο.
- Τοποθετήστε το ένα μετά το άλλο, μπλε και κόκκινα αντικείμενα στο σημείο συλλογής, (δίνοντας χρονικό περιθώριο τόσο στον αισθητήρα να προλάβει να διαβάσει το χρώμα όσο και το ρομπότ για να κινηθεί) και παρακολουθήστε το ρομπότ να τα ταξινομεί ανάλογα με το χρώμα τους.

Δραστηριότητα 18: Προγραμματισμός διακόπτη έκτακτης ανάγκης III

Περιγραφή

Στη συγκεκριμένη δραστηριότητα θα προγραμματίσετε ένα διπλό διακόπτη του κυκλώματος. Ο ένας διακόπτης (μαγνητικός διακόπτης θα ενεργοποιεί και απενεργοποιεί τη λειτουργία του ρομπότ ενώ ο διακόπτης κλίσης θα απενεργοποιεί το ρομπότ σε περίπτωση σεισμού (αλλαγή κλίσης του αισθητήρα).

Θα προγραμματίσετε το κύκλωμα σας έτσι ώστε:

AN ο μαγνητικός διακόπτης ανιχνεύει μαγνητικό πεδίο και

α) AN ο διακόπτης κλίσης είναι σε επίπεδη κλίση, **TOTE** το ρομπότ να εκτελεί μια ρουτίνα κινήσεων προσπαθώντας να σχηματίσει ένα τετράγωνο.

β) AN ο ο διακόπτης κλίσης δεν είναι σε επίπεδη κλίση, **TOTE** το ρομπότ να σταματάει να κινείται.

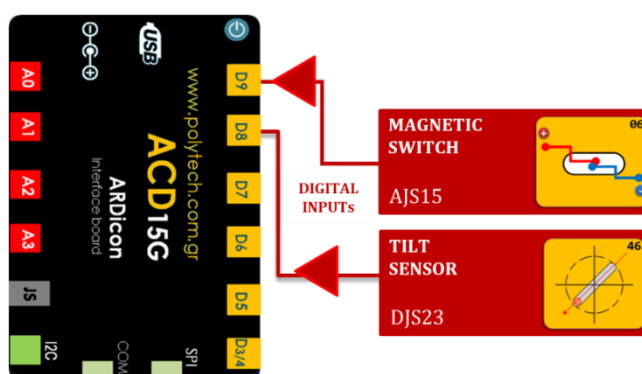
AN ο μαγνητικός διακόπτης ανιχνεύει μαγνητικό πεδίο, **TOTE** το ρομπότ να παραμένει ακίνητο.

Εξοπλισμός

- Ρομπότ R4
- Ελεγκτής ARD:icon **ACD15G**
- Μαγνητικός διακόπτης **AJS15**
- Διακόπτης κλίσης **DJS23**
- Καλώδιο UTP (x2)
- Καλώδιο USB

Επιπλέον Υλικά

- Μαγνήτης



Σύνδεση κυκλώματος

Συνδέστε τον ελεγκτή ARD:icon στον υπολογιστή σας με το καλώδιο USB.

Συνδέστε τη συσκευή **AJS15** στην ψηφιακή θύρα 9 (**D9**) του ελεγκτή ARD:icon.

Συνδέστε τη συσκευή **DJS23** στην ψηφιακή θύρα 8 (**D8**) του ελεγκτή ARD:icon.

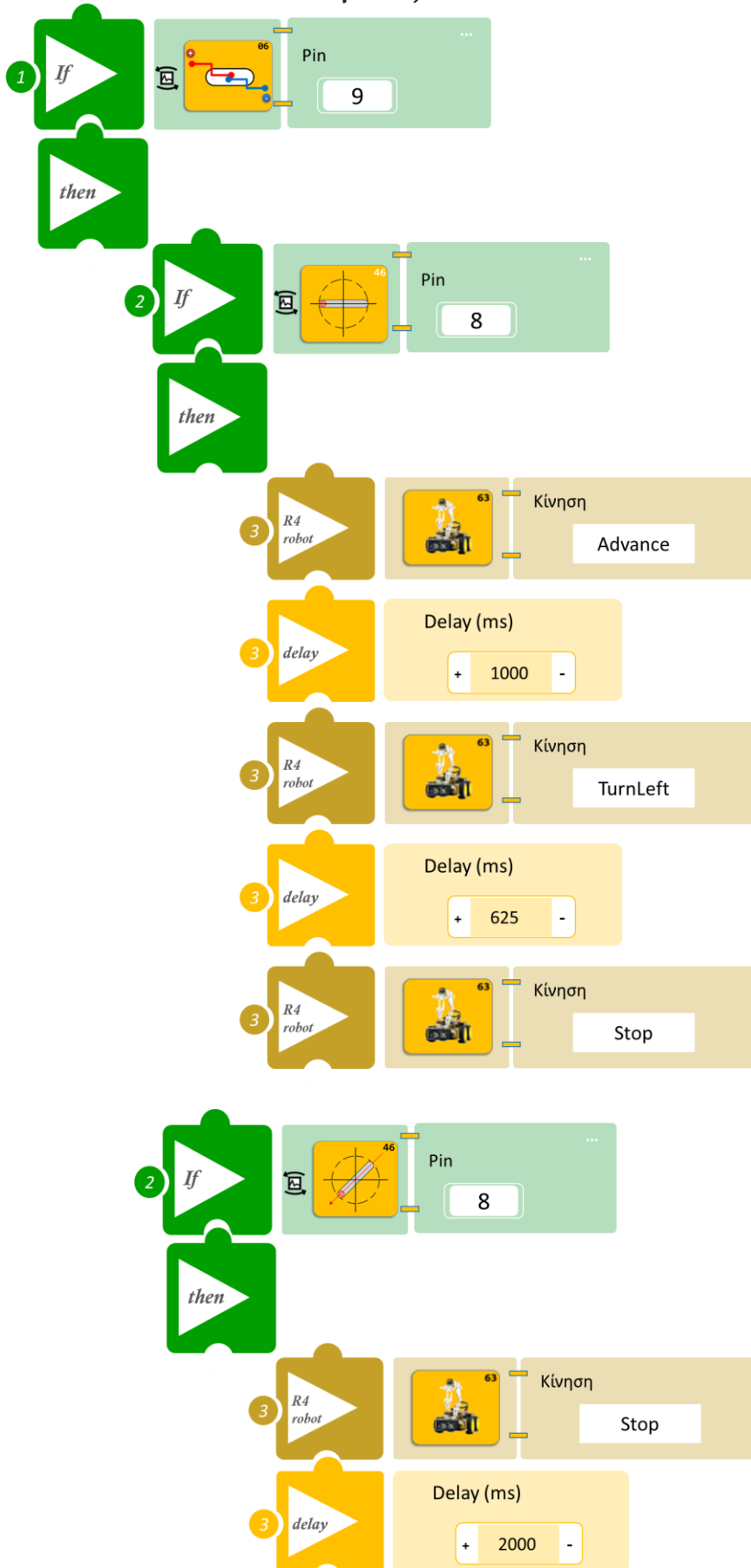
Ενεργοποιήστε το ρομπότ.

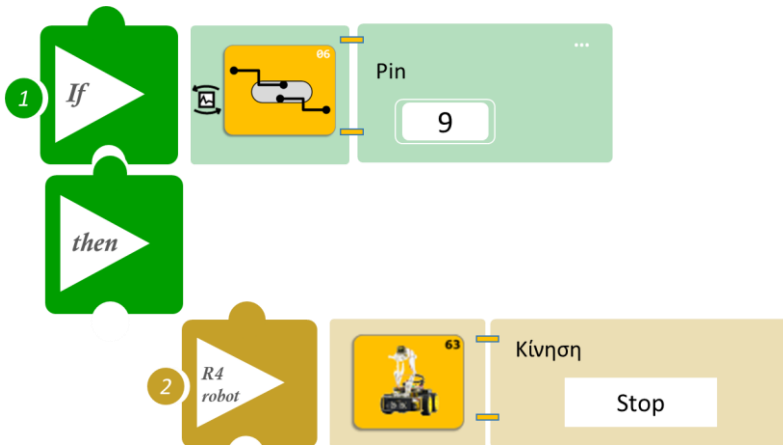
[Ενεργοποιήστε την ασύρματη σύνδεση του ρομπότ με τον υπολογιστή σας.](#)

Προγραμματισμός Ρομπότ

Γραμμή Εντολών	Εντολή	Συσκευή	Κατάσταση/ Θύρα
1 ^η	" <u>if</u> "	AJS15	ON / 9
2 ^η	" <u>if</u> "	DJS23	OFF / 8
3 ^η	" <u>R4 Κίνηση κινητήρων</u> "	Robot	Advance
3 ^η	" <u>delay</u> "		2000ms
3 ^η	" <u>R4 Κίνηση κινητήρων</u> "	Robot	TurnLeft
3 ^η	" <u>delay</u> "		625ms
3 ^η	" <u>R4 Κίνηση κινητήρων</u> "	Robot	Stop
2 ^η	" <u>if</u> "	DJS07	ON / 8
3 ^η	" <u>R4 Κίνηση κινητήρων</u> "	Robot	Stop
3 ^η	" <u>delay</u> "		2000ms
1 ^η	" <u>if</u> "	AJS15	≤ 101 / A1
2 ^η	" <u>R4 Κίνηση κινητήρων</u> "	Robot	Stop

Πλακίδια Εντολών Κυκλώματος





- Επιλέξτε **Παραγωγή (Generate)** για να δημιουργήσετε τον κώδικα στο λογισμικό.
- Περιμένετε μερικά δευτερόλεπτα και επιλέξτε **Φόρτωση (Upload)** για να φορτώσετε το πρόγραμμα στον ελεγκτή σας.
- Ανοίξτε τον Πίνακα ελέγχου της εφαρμογής επιλέγοντας το κουμπί **Πίν. Ελέγχου**. Επιλέξτε **ON**. Επιλέξτε **OFF** όταν τελειώσετε την εκτέλεση της δραστηριότητας.
- Απενεργοποιήστε το ρομπότ εκτός αν θέλετε να προχωρήσετε σε επόμενη δραστηριότητα.

ΑΠΟΤΕΛΕΣΜΑ

- Πλησιάστε τον μαγνήτη κοντά στον μαγνητικό διακόπτη και παρακολουθήστε το ρομπότ να εκτελεί τις κινήσεις που έχετε προγραμματίσει.
- Αλλάξτε την κλίση του διακόπτη κλίσης και παρατηρείστε το ρομπότ.
- Προσαρμόστε το χρόνο που στρίβει το ρομπότ ("delay"), αν χρειαστεί προκειμένου να δημιουργήσετε το νοητό τετράγωνο. Επίσης μπορεί να χρειαστεί να προσαρμόσετε το χρόνο ανάλογα με την απόδοση των μπαταριών. Όσο μειώνεται η απόδοση των μπαταριών μειώνεται και η ταχύτητα του ρομπότ.