



Εργα (IoT) και Παιχνίδια
Ενσωματωμένα, Πρόσθετα, Mind+, Arduino, Giga, S1, S2
ARD: icon
Micro SHIELD
Nezha
Gigo
με
Mind+

Πίνακας περιεχομένων

Περίληψη.....	4
Στόχοι του βιβλίου	4
Το περιβάλλον οπτικοποιημένου προγραμματισμού Mind+	6
«Μιλώντας» στις θύρες των μικροεπεξεργαστών.....	10
Εφαρμογή για εξαγωγή σε .CSV μετρήσεων αισθητήρων	14
Δομικά υλικά τύπου Lego	20
Εξοπλισμός S1 & S2 ARDicon	30
Ο μικροελεγκτής ARD:icon	33
Αναλογικοί Αισθητήρες	35
Αναλογικός αισθητήρας ήχου (μικρόφωνο) AJS02	36
Αναλογικός αισθητήρας φωτός LDR ή Φωτοκύταρο.....	36
Αναλογικός αισθητήρας γωνίας/Ροοστάτης AJS06.....	37
Αισθητήρας υγρασίας εδάφους AJS05.....	37
Αισθητήρας υγρασίας/στάθμης νερού AJS04.....	38
Αισθητήρας πίεσης λεπτής μεμβράνης AJS07	39
Ψηφιακοί Ενεργοποιητές	40
LED (Κόκκινο) DJX06.....	42
Ενεργός βομβητής (ηχείο) AJX03.....	42
LED (RGB) DJX13	43
Ψηφιακοί Αισθητήρες	44
Αισθητήρας κίνησης PIR (Pyroelectric Infrared) DJS19.....	45
Αισθητήρας υγρασίας-θερμοκρασίας MJS22	46
Διακόπτης πίεσης DJS09	46
Αισθητήρας αποφυγής εμποδίου υπερύθρων DJS17	47
Κινητήρας 360 Degree Continuous Rotation	48
Αισθητήρας απόστασης υπερήχων DJS22	49
Οθόνη LCD 16x2 AJX04.....	50
Βηματικός κινητήρας με έλικα (ανεμιστήρας) DJX09	51
Σερβοκινητήρας EXP-AJ11, DJX11	52
Το χρήσιμο πρόσθετο στο Mind+ για ARD:icon (arduino)	54
Οδηγίες εγκατάστασης	56
Λειτουργίες (πλακίδια εντολών).....	58
Διάβασμα τιμών από αισθητήρες	59



Καταστάσεις αισθητήρων (true, false)	60
ARD:icon Παραδείγματα - Εφαρμογές χωρίς δομικά υλικά	62
1 ^ο παράδειγμα: Ανάβοντας LED.	63
2 ^ο παράδειγμα: Άναμμα LEDs υπό συνθήκη.	65
3 ^ο παράδειγμα: Αυτοματισμός παρκαρίσματος οχημάτων	69
4 ^ο παράδειγμα: Αυτοματισμός κουμπιού που παράγει ήχο και φως	74
5 ^ο παράδειγμα: Αυτοματισμός ανίχνευσης κίνησης	77
6 ^ο παράδειγμα: Ανεμιστήρας με ρυθμιζόμενη ένταση περιστροφής	80
7 ^ο παράδειγμα: Περιστροφή κινητήρων με ρυθμιζόμενη ταχύτητα	83
8 ^ο παράδειγμα: Ο μηχανισμός της πόρτας του γκαράζ (servo)	87
9 ^ο παράδειγμα: Η αξιοποίηση της μέτρησης φωτεινότητας του χώρου	94
10 ^ο παράδειγμα: Προβολή τιμών στην οθόνη του Mind+	97
11 ^ο παράδειγμα: Τιμές αισθητήρα πίεσης λεπτής μεμβράνης στην Mind+ οθόνη	98
12 ^ο παράδειγμα: Οπτικοποίηση της πίεσης που ασκούμε στη μεμβράνη.	100
ADR:icon Δραστηριότητες με Δομικά υλικά.....	106
1 ^η δραστηριότητα: Όχημα με ρυθμιστή ταχύτητας (Κινητήρες, Ποτενσιόμετρο)	107
2 ^η δραστηριότητα: Όχημα με ρυθμιστή ταχύτητας και κατεύθυνσης	113
3 ^η δραστηριότητα: Όχημα που η κίνησή του σταματά σε εμπόδιο	120
4 ^η δραστηριότητα: Όχημα με ρυθμιστή ταχύτητας που αντιδρά στα εμπόδια.....	125
5 ^η δραστηριότητα: Μετρητής υγρασίας	132
6 ^η δραστηριότητα: Ρομποτικός βραχίονας (δαγκάνα)	138
7 ^η δραστηριότητα: Μπάρα χώρου στάθμευσης	144
8 ^η δραστηριότητα: Φανός με Neopixel, διακόπτη και αισθητήρα κίνησης	149
Micro SHIELD διεπαφή Micro:bit με Mind+	154
To micro SHIELD διεπαφή micro:bit.....	155
Micro:bit, ο μικροεπεξεργαστής του micro SHIELD.....	156
Τα πρόσθετα στο Mind+	159
1 ^ο παράδειγμα: Λειτουργώντας IoTs, micro:bit στο Mind+	161
NEZHA Inventor's Kit Micro:bit	168
NEZHA Το βαλιτσάκι και το περιεχόμενό του (ενεργά και δομικά στοιχεία)	169
To board NEZHA	173
Το πρόσθετο στο Mind+ για NEZHA inventor's kit & S1,S2 αισθητήρες	178
Οδηγίες εγκατάστασης του πρόσθετου	180
NEZHA & S1,S2 Παραδείγματα με Mind+	184
1 ^ο παράδειγμα: Μπάρα χώρου στάθμευσης με NEZHA και micro:bit στο Mind+	185
2 ^ο παράδειγμα: Σταματώντας τους κινητήρες	187
3 ^ο παράδειγμα: Περιστροφή κινητήρων με το ποτενσιόμετρο	189
Περισσότερες δραστηριότητες με NEZHA	191



4 ^ο παράδειγμα: Ανάβοντας NeoPixel LEDs	192
5 ^ο παράδειγμα: Ανάβοντας elecfreaks LED, με τιμές από αισθητήρα υπερήχων/απόστασης polyteck.....	194
6 ^η δραστηριότητα: Φανός με Neopixel, αισθητήρες σύγκρουσης (αντί κουμπιού), και κίνησης	197
Εξοπλισμός Gigo.....	202
Το χρήσιμο πρόσθετο στο Mind+ για Gigo (micro:bit)	208
Οδηγίες εγκατάστασης του πρόσθετου	210
Gigo Παραδείγματα	212
1 ^ο παράδειγμα: Το άναμμα LED με κουμπί Gigo (το πεντάλ του φρένου)	213
2 ^ο παράδειγμα: Περιστροφή κινητήρων και σταμάτημα υπό συνθήκες	215
Βιογραφικό σημείωμα.....	218

Ποτέ δε ξεχνώ τα λόγια του Αϊνστάιν:

«Αν δεν μπορείς να το εξηγήσεις με απλά λόγια, ούτε εσύ ο ίδιος το έχεις καταλάβει πολύ καλά.»

Πάντα θυμάμαι αυτό που ο Λεονάρντο Ντα Βίντσι είχε πει: «...Η μελέτη χωρίς επιθυμία κλειδώνει τη μνήμη και δεν συγκρατεί τίποτα από αυτά που δέχεται...»

Εμείς ταπεινά με αυτές τις προσεγγίσεις προσπαθούμε να δημιουργήσουμε την επιθυμία για μάθηση. Το διαδραστικό υλικό και το εύχρηστο λογισμικό που έχει πλέον στη διάθεσή του ο εκπαιδευτικός, εμπλουτίζει τα μαθήματά του, δημιουργεί πολλαπλές αναπαραστάσεις στους μαθητές/τριες με στόχο την κινητοποίησή τους για δημιουργίες και τη διεύρυνση των οριζόντων τους. [BO]

Γίνεσαι χρήσιμος για όλους και καλύτερος με τον εαυτό σου, όταν προσπαθείς να βοηθήσεις, μοιράζοντας όσες γνώσεις απέκτησες με λιγότερο ή περισσότερο κόπο... [BO]



Περίληψη

Το βιβλίο αυτό αποτελεί ένα εκπαιδευτικό οδηγό **υποστήριξης για τα συστήματα**

S1, S2, ARD:icon, R2, Nezha inventor's Kit, Gigo

με ένα **ενιαίο περιβάλλον** οπτικοποιημένου προγραμματισμού Scratch like
 το **Mind+**

...το οποίο απευθύνεται σε εκπαιδευτικούς και μαθητές/τριες Δημοτικού και Γυμνασίου.

Στόχος του είναι να εισαγάγει τους αναγνώστες στον κόσμο του προγραμματισμού και των αυτοματισμών, χρησιμοποιώντας τον μικροεπεξεργαστή Arduino και τον micro:bit καθώς επίσης και ένα φιλικό περιβάλλον οπτικοποιημένου προγραμματισμού Mind+, καθώς και η ενίσχυση της δημιουργικής και αναλυτικής σκέψης μέσω πρακτικών εφαρμογών και η ανάπτυξη δεξιοτήτων STEM (Επιστήμη, Τεχνολογία, Μηχανική και Μαθηματικά).

Μέσα από απλά (βηματικής παρουσίασης) παραδείγματα και δραστηριότητες, οι μαθητές/τριες θα μάθουν να δημιουργούν αυτοματισμούς, να κατανοούν τις βασικές αρχές των ηλεκτρονικών κυκλωμάτων και να αναπτύσσουν δεξιότητες λογικής σκέψης και επίλυσης προβλημάτων.

Το βιβλίο παρέχει επίσης μικρές πρακτικές συμβουλές και υποστήριξη στους εκπαιδευτικούς για την αποτελεσματική διδασκαλία αυτών των θεμάτων στο εργαστήριο.

Στόχοι του βιβλίου

Γενικοί στόχοι για τον αναγνώστη είναι:

- Η κατανόηση των βασικών μερών του ARD:icon, του NEZHA, του Gigo και των λειτουργιών τους
- Η εξοικείωση με το περιβάλλον προγραμματισμού Mind+:
- Η εφαρμογή ενεργών και παθητικών δομικών υλικών σε κατασκευές.
- Η επίλυση προβλημάτων κατά την ανάπτυξη έργων.
- Η ανάπτυξη δεξιοτήτων STEM.



Οι αναγνώστες θα είναι σε θέση:

- Να κατανοήσουν τη λειτουργία των πρόσθετων STEM extension.
- Να σχεδιάσουν και να υλοποιήσουν απλές εφαρμογές σε Mind+ που αξιοποιούν το S1, S2, NEZHA inventor's kit και Gigo πακέτα που έχουν διανεμηθεί στα Δημόσια Σχολεία.
- Να εμπλακούν με τη συνδεσμολογία απλών περιφερειακών στο ARD:icon, το NEZHA και το Gigo.
- Να αξιοποιήσουν σημαντικές δομές προγραμματισμού.
- Να ολοκληρώσουν στο εργαστήριο, ορισμένους από τους αυτοματισμούς που συναντάμε γύρω μας.

Συνδυασμός υλικών από όλα τα πακέτα με ένα ενιαίο προγραμματιστικό περιβάλλον





Το περιβάλλον οπτικοποιημένου προγραμματισμού Mind+



Το Mind+ είναι ένα οπτικοποιημένο προγραμματιστικό περιβάλλον που έχει σχεδιαστεί για να διευκολύνει την εκμάθηση προγραμματισμού και των αυτοματισμών, χρησιμοποιώντας μπλοκ κώδικα αντί για παραδοσιακό κείμενο. Απευθύνεται κυρίως σε αρχάριους και μαθητές, καθιστώντας το ιδανικό για εκπαιδευτικά σενάρια.

Κατεβάζουμε το mind+ από τη σελίδα:

<https://mindplus.cc/download-en.html>

...και το εγκαθιστούμε στον Η/Υ μας.

Βασικά στοιχεία του Mind+:

1. **Διεπαφή βασισμένη σε μπλοκ:** Ο χρήστης δημιουργεί προγράμματα συνδέοντας οπτικά μπλοκ, τα οποία αντιπροσωπεύουν εντολές, δομές ελέγχου, μεταβλητές, λειτουργίες και άλλες ενέργειες. Κάθε μπλοκ αντιστοιχεί σε ένα τμήμα κώδικα που εκτελεί συγκεκριμένη ενέργεια.
2. **Συμβατότητα με τον Arduino και άλλους μικροεπεξεργαστές:** Το Mind+ επιτρέπει τον προγραμματισμό του μικροελεγκτή Arduino, micro:bit κα, διευκολύνοντας την ανάπτυξη εφαρμογών και έργων αυτοματισμού με αισθητήρες, κινητήρες και άλλα ηλεκτρονικά εξαρτήματα.
3. **Διαχείριση εισόδων και εξόδων:** Παρέχει εύκολη πρόσβαση στη διαχείριση εισόδων και εξόδων, όπως αισθητήρες, φώτα, κινητήρες, ήχους, κ.λπ., με στόχο τη δημιουργία διαδραστικών εφαρμογών.
4. **Απλοποίηση πολύπλοκων δομών:** Υποστηρίζει δομές ελέγχου όπως επαναλήψεις και συνθήκες, επιτρέποντας στους μαθητές να κατανοήσουν βασικές έννοιες αλγορίθμων με απλό και κατανοητό τρόπο.
5. **Υποστήριξη πολλαπλών γλωσσών:** Εκτός από την οπτική μορφή με τα μπλοκ, το Mind+ επιτρέπει στους χρήστες να δουν και τον κώδικα σε C++ (γλώσσα που χρησιμοποιείται για τον Arduino), βοηθώντας στη μετάβαση από το οπτικοποιημένο περιβάλλον στον πραγματικό κώδικα.
6. **Εκμάθηση βήμα προς βήμα:** Το περιβάλλον προσφέρει ενσωματωμένα μαθήματα και οδηγούς για τον χρήστη, κάνοντας την εκμάθηση του προγραμματισμού διαδραστική και διασκεδαστική.
7. **Ποικιλία αυτοματισμών:** Οι εκπαιδευτικοί και οι μαθητές μπορούν να δημιουργούν διάφορους αυτοματισμούς, από απλούς φωτεινούς αισθητήρες μέχρι πιο σύνθετα ρομποτικά έργα.

Το Mind+ προσφέρει μια ισορροπία μεταξύ του απλού, εισαγωγικού προγραμματισμού και των πιο προχωρημένων λειτουργιών, βοηθώντας τους μαθητές να εξελιχθούν σταδιακά.



Η παιδαγωγική αξία του Mind+

Η **σημαντικότερη παιδαγωγική αξία του Mind+** έγκειται στην ικανότητά του να κάνει τον προγραμματισμό και την εκμάθηση εννοιών αυτοματισμού προσιτή και κατανοητή σε μαθητές κάθε ηλικίας. Συγκεκριμένα:

1. Καλλιέργεια Υπολογιστικής Σκέψης:

Το Mind+ επιτρέπει στους μαθητές να αναπτύξουν δεξιότητες υπολογιστικής σκέψης, όπως η λογική, η ανάλυση προβλημάτων και η συστηματική προσέγγιση. Η χρήση μπλοκ προγραμματισμού καθιστά την κατανόηση των αλγορίθμων και των δομών δεδομένων πιο διαισθητική, χωρίς να τους δυσκολεύει η σύνταξη του κώδικα.

2. Μάθηση μέσω της Δημιουργίας (Constructionism):

Σύμφωνα με την παιδαγωγική θεωρία του Seymour Papert, η μάθηση μέσω της δημιουργίας είναι πιο αποτελεσματική. Οι μαθητές στο Mind+ δημιουργούν ρεαλιστικά έργα, συνδυάζοντας προγραμματισμό με φυσικά αντικείμενα (όπως Arduino, αισθητήρες κ.λπ.). Έτσι, ενισχύεται η ενεργητική μάθηση, καθώς οι μαθητές μαθαίνουν μέσα από την κατασκευή και πειραματίζονται.

3. Ανάπτυξη Διαθεματικών Δεξιοτήτων:

Το Mind+ ενσωματώνει αρχές STEAM (Science, Technology, Engineering, Arts, and Mathematics), προσφέροντας στους μαθητές τη δυνατότητα να αναπτύξουν δεξιότητες που συνδυάζουν τεχνολογία, μηχανική, μαθηματικά και δημιουργικότητα. Έτσι, μαθαίνουν να συνδέουν γνώσεις από διάφορα γνωστικά αντικείμενα, συμβάλλοντας στην ολιστική τους ανάπτυξη.

4. Διαφοροποιημένη Διδασκαλία:

Επιτρέπει την προσαρμογή της διδασκαλίας στο επίπεδο κάθε μαθητή. Οι αρχάριοι μπορούν να χρησιμοποιούν τα οπτικά μπλοκ, ενώ οι πιο προχωρημένοι μαθητές μπορούν να βλέπουν και να επεξεργάζονται τον κώδικα C++. Αυτό προσφέρει ευελιξία στους εκπαιδευτικούς να εξατομικεύουν τη διδασκαλία.

5. Ενεργή Συμμετοχή και Εμπλοκή:

Το Mind+ ενισχύει την ενεργή συμμετοχή των μαθητών, καθώς εμπλέκονται δημιουργικά στη διαδικασία. Οι μαθητές αισθάνονται μεγαλύτερη αυτονομία και έχουν κίνητρο να διερευνήσουν διαφορετικούς τρόπους επίλυσης προβλημάτων, δίνοντάς τους τον ρόλο του δημιουργού και όχι μόνο του χρήστη της τεχνολογίας.

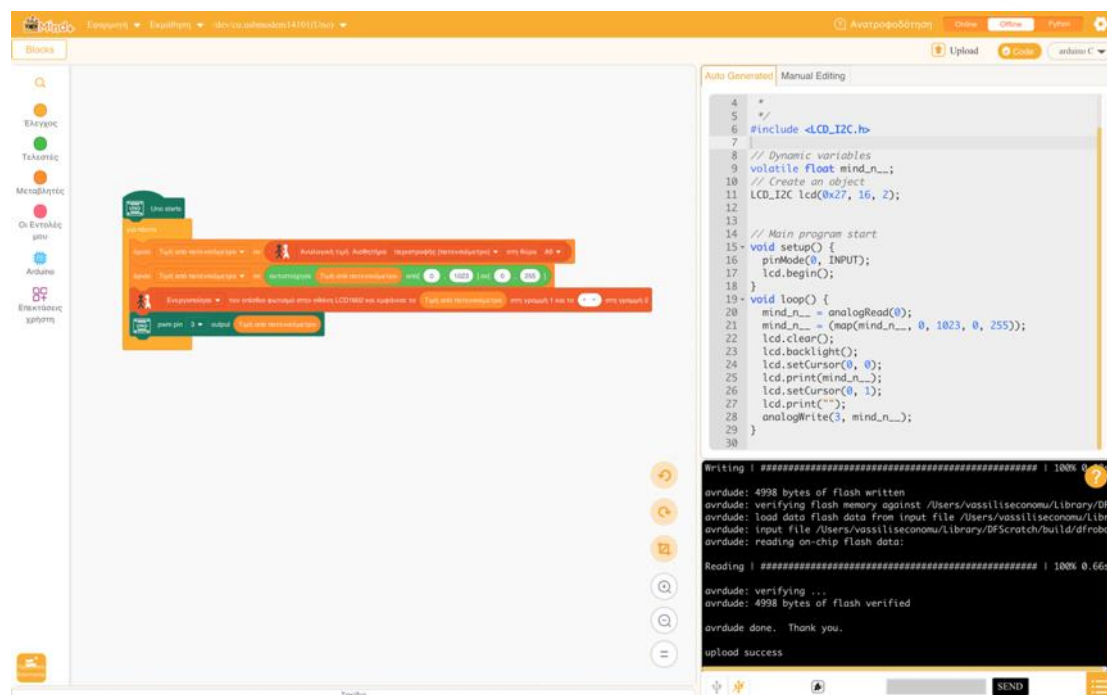
6. Εκμάθηση με βάση τη δοκιμή και το λάθος:

Η διαδικασία του προγραμματισμού στο Mind+ επιτρέπει στους μαθητές να κάνουν λάθη, να δοκιμάζουν διαφορετικές προσεγγίσεις και να



βελτιώνουν το έργο τους. Αυτή η διερευνητική μάθηση ενισχύει την αυτοπεποίθηση και την αυτοδιόρθωση.

Η παιδαγωγική αξία του Mind+ έγκειται στο ότι δημιουργεί ένα περιβάλλον όπου οι μαθητές μαθαίνουν μέσω της πράξης, της δημιουργίας και της πειραματικής διερεύνησης, ενώ αναπτύσσουν κρίσιμες δεξιότητες για το μέλλον τους.





«Μιλώντας» στις θύρες των μικροεπεξεργαστών



Μια γενική αρχή σε σχέση με τις εντολές του περιβάλλοντος προγραμματισμού είναι η επικοινωνία με τις θύρες του μικροεπεξεργαστή, και μέσω των HUBs με τα οποία αυτός συνεργάζεται.

Ακολουθούν κάποιες βασικές έννοιες (αρχικά για ειδικότερη γνώση της λειτουργίας ορισμένων από τις εντολές που θα συναντήσουμε στη συνέχεια):

Digital write

Χρησιμοποιείται σε προγραμματιστικά περιβάλλοντα, για τον έλεγχο των ψηφιακών ακροδεκτών/θυρών ενός μικροεπεξεργαστή.

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 της, όπως για παράδειγμα ένα LED.

Παράδειγμα κώδικα (τον οποίο θα συναντήσουμε ως οπτικοποιημένο πλακίδιο) που ανάβει και σβήνει ένα LED το οποίο βρίσκεται συνδεδεμένο στη θύρα που προσδιορίζει η μεταβλητή LEDPIN.

```
{
    digitalWrite (LEDPIN, HIGH);    // Άναψε το LED
    delay (200);                    // Αναμονή 200 milliseconds
    digitalWrite (LEDPIN, LOW);     // Σβήσε το LED
    delay (200);                    // Αναμονή 200 milliseconds
}
```



Digital Read

Χρησιμοποιείται συνήθως 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.

Παράδειγμα κώδικα (τον οποίο θα συναντήσουμε ως οπτικοποιημένο πλακίδιο) που διαβάζει την τιμή ενός διακόπτη που βρίσκεται συνδεδεμένος στην ψηφιακή θύρα 7 (σύμφωνα με τον αρχικό ορισμό του παραδείγματος)

```
/*  
    Ανάγνωση ψηφιακής ακίδας του Arduino  
*/  
  
#define BUTTONPIN      7  
  
void setup ()  
{  
    pinMode (BUTTONPIN, INPUT);    // Η ακίδα 7 καθορίζεται ως ψηφιακή είσοδος  
}  
  
void loop ()  
{  
    int value = digitalRead (BUTTONPIN);    // Διάβασε την κατάσταση του διακόπτη  
    // Εδώ πρέπει να κάνουμε κάτι, ανάλογα με την τιμή της ακίδας 7  
}
```



Analog write

Χρησιμοποιείται συνήθως 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 μας επιτρέπει να προσομοιώσουμε 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.

Analog Read

Χρησιμοποιείται συνήθως 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).

Serial print

Χρησιμοποιείται συνήθως 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.

Στη συνέχεια θα βρούμε ειδικά υλοποιημένη εφαρμογή που μεταφέρει τις τιμές που στέλνουμε στη σειριακή θύρα σε εγγραφές σε ένα αρχείο .CSV για περαιτέρω επεξεργασία.



Εφαρμογή
για εξαγωγή
σε .CSV
μετρήσεων
αισθητήρων



Πολλές φορές χρειάζεται να εξάγουμε σε αρχείο ,CSV όλες τις μετρήσεις από τους αισθητήρες που έχουμε ενσωματώσει στις εφαρμογές μας. Το αρχείο αυτό το ανοίγουμε με EXCEL like περιβάλλοντα ώστε να μπορέσουμε να επεξεργαστούμε περεταίρω τα δεδομένα και να τα απεικονίσουμε σε διαγράμματα κ.λπ.

A. Κατεβάζουμε την εφαρμογή (σύμφωνα με το λειτουργικό σύστημα που διαθέτουμε), από τη σελίδα:

<https://portal.stem.edu.gr/courses/data-to-csv>

B. αποσυμπίεζουμε το .zip αρχείο και το αποθηκεύουμε στον Η/Υ μας

Γ. Στο Mind+, χρησιμοποιούμε την κατάλληλη εντολή για εγγραφή στη σειριακή θύρα των δεδομένων από τους αισθητήρες (βλ. παρακάτω εντολές ανάλογα με τον μικροεπεξεργαστή)



Δ. Ξεκινάμε την εφαρμογή καταγραφής (με διπλό κλικ επάνω της). Την εφαρμογή αυτή μπορούμε να την εκκινήσουμε λίγο πριν αρχίσει να εκτελείται το πρόγραμμα ανάγνωσης των τιμών των αισθητήρων (που κατεβάζουμε στους μικροεπεξεργαστές), και τρέχει παράλληλα καταχωρίζοντας αυτόματα τις τιμές που λαμβάνει από τη σειριακή εγγραφή, σε ένα αρχείο .CSV.

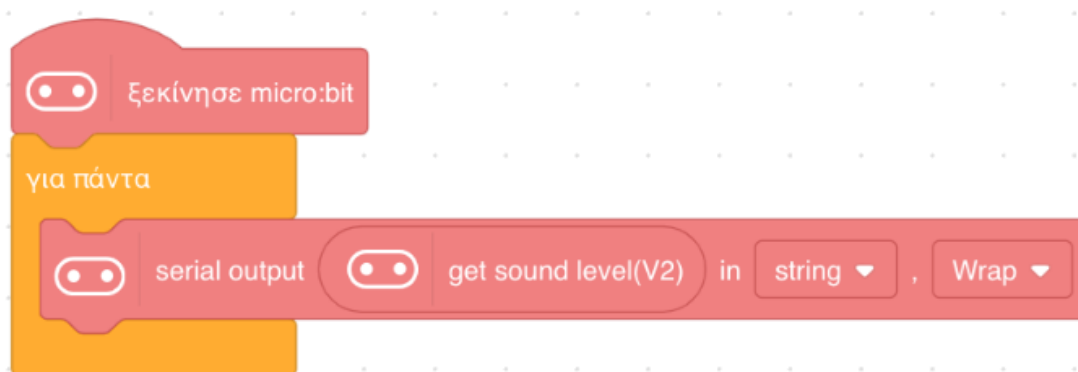
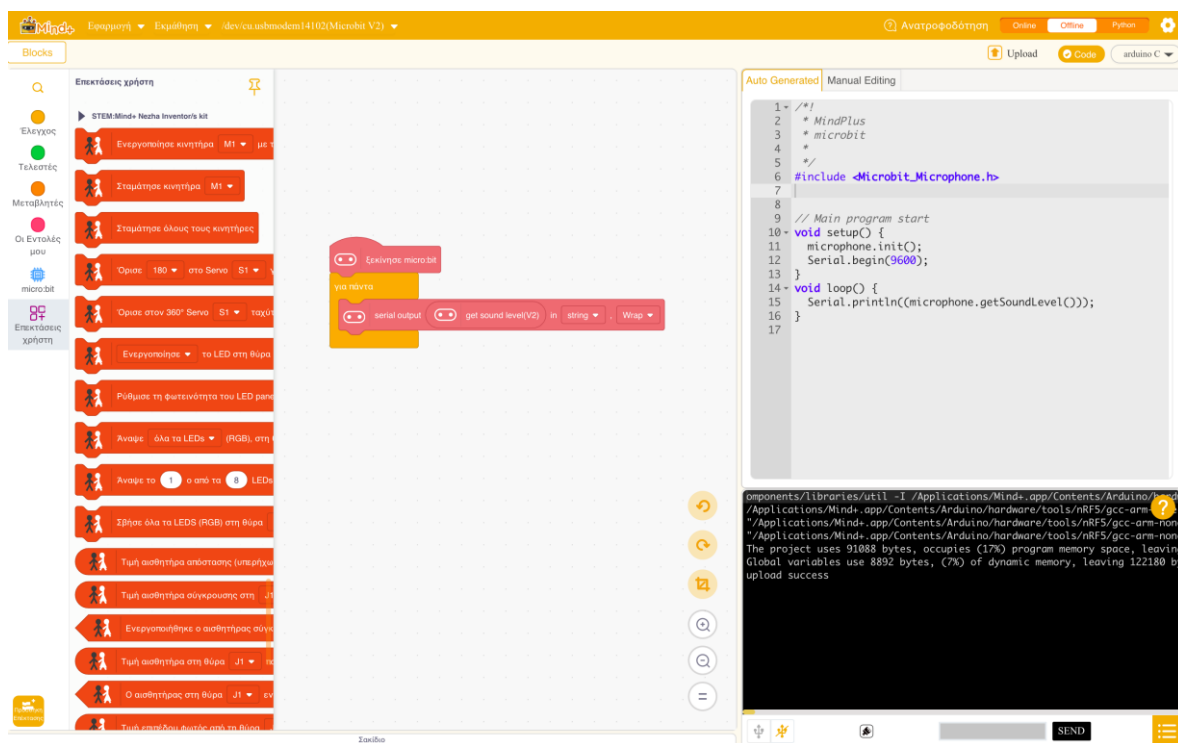
Ε. Τρέχουμε το πρόγραμμά μας στο Mind+ (το οποίο διαβάζει τιμές αισθητήρων)

Στ. Σταματάμε την εφαρμογή καταγραφής και αυτόματα δημιουργείται το αρχείο .CSV με την καταγραφή των τιμών σε στήλες, το οποίο στη συνέχεια θα ανοίξουμε με το EXCEL like πρόγραμμα, για περεταίρω επεξεργασία



Παράδειγμα

A. Δημιουργία προγράμματος στο mind+ το οποίο διαβάζει τιμές από τον αισθητήρα θορύβου στον micro:bit και στέλνει τις τιμές στη Serial:



B. Εκκίνηση του αρχείου καταγραφής

Διπλό κλικ πάνω στην εφαρμογή (στο MAC desktop)



Ανοίγει η εφαρμογή.

Βρίσκει την θύρα που είναι συνδεδεμένος ο micro:bit και προτείνει να την επιλέξουμε.

Προτείνει το ρυθμό μετάδοσης δεδομένων (Bout rate) 9600.

Μας ρωτά που θέλουμε να αποθηκευτεί το αρχείο και με πιο όνομα (προτείνοντας θέση και όνομα).

```
vassiliseconomu — serial_to_csv — 83x36
Last login: Mon Dec 30 23:07:34 on ttys000
/Users/vassiliseconomu/Desktop/serial_to_csv ; exit;

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(base) 192:~ vassiliseconomu$ /Users/vassiliseconomu/Desktop/serial_to_csv ; exit;
Εφαρμογή αυτόματης καταγραφής σε αρχείο (.csv) μετρήσεων από αισθητήρες v.1.0.0
[?] Επιλέξτε τη θύρα του microprocessor:
> BBC micro:bit CMSIS-DAP - mbed Serial Port

[?] Επιλέξτε το baud rate:
4800
> 9600
19200
38400
57600
115200
230400
460800
921600

Σύνδεση επιτυχής στη θύρα!
[?] Πού πρέπει να βρίσκονται τα αρχεία καταγραφής;: /Users/vassiliseconomu/sensor_data.csv
Πατήστε Ctrl + C (ή Command + .) για να σταματήσετε την καταγραφή.
Καταγραφή: 15
Καταγραφή: 0
Καταγραφή: 0
Καταγραφή: 0
Καταγραφή: 0
Καταγραφή: 0
Καταγραφή: 0
Καταγραφή: 0
Καταγραφή: 0
Καταγραφή: 0
```

Και αρχίζει η καταγραφή... (εφόσον έχουμε ξεκινήσει το πρόγραμμά μας στο Mind+)



Το παραπάνω στιγμιότυπο σε κείμενο:

Last login: Mon Dec 30 23:07:34 on ttys000
 /Users/vassiliseconomu/Desktop/serial_to_csv ; exit;

The default interactive shell is now zsh.

To update your account to use zsh, please run `chsh -s /bin/zsh`.

For more details, please visit <https://support.apple.com/kb/HT208050>.

(base) 192:~ vassiliseconomu\$ /Users/vassiliseconomu/Desktop/serial_to_csv ; exit;

Εφαρμογή αυτόματης καταγραφής σε αρχείο (.csv) μετρήσεων από αισθητήρες
 v.1.0.0

[?] Επιλέξτε τη θύρα του microprocessor:

> BBC micro:bit CMSIS-DAP - mbed Serial Port

[?] Επιλέξτε το baud rate:

4800

> 9600

19200

38400

57600

115200

230400

460800

921600

Σύνδεση επιτυχής στη θύρα!

[?] Πού πρέπει να βρίσκονται τα αρχεία καταγραφής::

/Users/vassiliseconomu/sensor_data.csv

Πατήστε Ctrl + C (ή Command + .) για να σταματήσετε την καταγραφή.

Καταγραφή: 0

Καταγραφή: 0

Καταγραφή: 0

...

Καταγραφή: 0

Καταγραφή: 52

Καταγραφή: 48

Καταγραφή: 41

Καταγραφή: 45

...

Καταγραφή: 11

Καταγραφή: 15

Καταγραφή: 0

ΛC

Διακοπή από τον χρήστη. Κλείσιμο σύνδεσης...

Τα δεδομένα αποθηκεύτηκαν στο '/Users/vassiliseconomu/sensor_data.csv'.

logout

Saving session...

...copying shared history...

...saving history...truncating history files...

...completed.

[Η διεργασία ολοκληρώθηκε.



Γ. Το αποτέλεσμα

Ανοίγοντας το αρχείο **sensor_data.csv** με EXCEL like εφαρμογή έχουμε το παρακάτω...

Πολύ μικρό τμήμα του αρχείο που δημιουργήθηκε κατά την καταγραφή:

[illegible]



Δομικά υλικά τύπου Lego



Μπορούμε να συνδυάσουμε τους αισθητήρες των πακέτων S1 και S2 Elecfreacks, με δομικά υλικά Lego ή τύπου Lego, προκειμένου να αξιοποιηθούν από τα παιδιά και σε κατασκευαστικό επίπεδο.

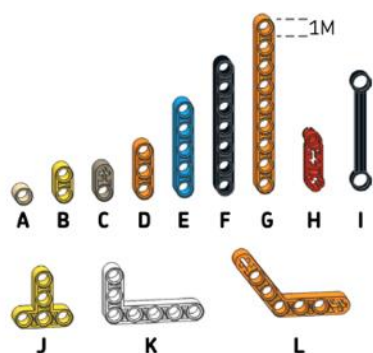
Για τη δημιουργία κατασκευών προτείνονται τα ακόλουθα [Δομικά υλικά](#)



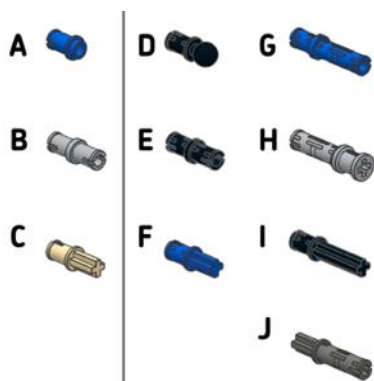
Στα δομικά υλικά υπάρχουν στο βαλιτσάκι NEZHA και μπορούν να προσαρμοστούν οι αισθητήρες των πακέτων S1 και S2 χρησιμοποιώντας τις κατάλληλες [βίδες](#)



Δομικά στοιχεία και ονοματολογία



LABEL	NAME	COLOR
A	1M beam	tan
B	2M beam	yellow
C	2M beam with cross hole	dark tan
D	3M beam	orange
E	5M beam	dark azure
F	7M beam	black
G	9M beam	orange
H	3M liftarm with pin	red
I	6M link with ball sockets	black
J	T beam	yellow
K	3x5 angular beam (L beam)	white
L	4x4 angular beam	orange



LABEL	NAME	COLOR
A	pin 1/2 with stud	blue
B	pin without friction	grey
C	axle pin without friction	tan
D	pin with towball	black
E	pin with friction	black
F	axle pin with friction	blue
G	3M pin with friction	blue
H	3M pin with stop bush	grey
I	3M axle pin with 2M axle	black
J	3M axle pin with 2M pin	dark grey

Οι **δοκοί** μετρούνται με τις τρύπες (οπές) τους.

Για παράδειγμα, μια ευθεία δοκός με τρεις οπές είναι μια δοκός 3Μ.

Ο αριθμός των οπών σε μια δοκό αντιστοιχεί στο μήκος της δοκού.

Συνδετήρες/σύνδεσμοι

Οι συνδετήρες κάνουν ένα βασικό πράγμα: συνδέουν μέρη.

Για παράδειγμα, όταν χτίζουμε με ξύλο, χρησιμοποιούμε καρφιά, κόλλα, βίδες κ.λπ. για να συνδέσουμε κομμάτια. Στη LEGO, χρησιμοποιούμε συνδετήρες, άξονες, δακτυλίους, δοκούς και διάφορα παρόμοια μπλοκ. Ι συνδετήρες συγκρατούν τις δοκούς όταν τοποθετούνται μέσα στις στρογγυλές τους τρύπες. Χωρίζονται σε δύο ομάδες: Συνδετήρες με τριβή και συνδετήρες χωρίς τριβή.

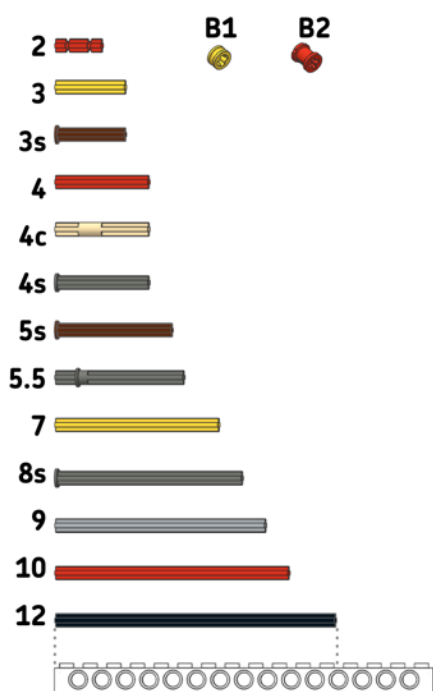
Οι συνδετήρες χωρίς τριβή (Α, Β και C) περιστρέφονται ομαλά και ελεύθερα στις οπές των δοκών. Έχουν χρωματική κωδικοποίηση: οι ακίδες των 2Μ είναι πάντα γκρι, ενώ οι ακίδες και οι ακίδες άξονα των 3Μ είναι μαύρες. Χρησιμοποιούνται κυρίως για τη σύνδεση κινούμενων δοκών (π.χ. για τη συγκράτηση ενός γραναζιού ώστε να μπορεί να περιστρέφεται ελεύθερα.

Οι συνδετήρες με τριβή (D έως J) έχουν ραβδώσεις που αυξάνουν την τριβή και κάνουν δύσκολη την περιστροφή. Οι συνδετήρες έχουν χρωματική κωδικοποίηση για να τους βοηθήσουν να αναγνωρίζουμε τη λειτουργία τους.

Οι 2Μ με τριβή είναι πάντα μαύροι και οι 3Μ και αυτοί με άξονα με τριβή είναι μπλε.

Υπάρχουν και ειδικοί συνδετήρες άξονα 3Μ (I και J).



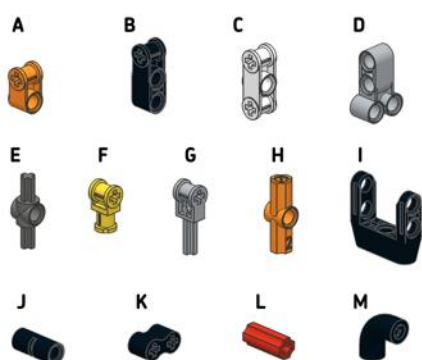


Οι **άξονες** έχουν χρωματική κωδικοποίηση.

Οι άξονες 2M, 4M και 10M είναι κόκκινοι και οι άξονες 3M και 7M είναι κίτρινοι.

Υπάρχουν άξονες 4M (4c) με κυλινδρικό στοπ στη μέση, καθώς και άξονες με στοπ στην άκρη (3s, 4s, 5s, και 8s).

Στους άξονες, το στοπ είναι ενσωματωμένο και τον εμποδίζει να περάσει μέσα από μια τρύπα ή μια εγκάρσια οπή.



Εγκάρσια μπλοκ και συνδετήρες

Τα σταυρωτά μπλοκ είναι απαραίτητα για την κατασκευή χωρίς συνδετήρες, επειδή σας επιτρέπουν να χτίζετε —και να σκέφτεστε— σε τρεις διαστάσεις.

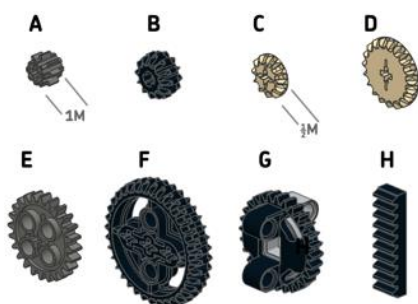
Ο γωνιακός σύνδεσμος με την ένδειξη H είναι μέρος μιας μεγάλης οικογένειας συνδετήρων.

Το εξάρτημα με την ένδειξη I μπορεί να χρησιμοποιηθεί σε κιβώτιο ταχυτήτων για να συγκρατεί κωνικά γρανάζια 90 μοιρών 12z και 20z.

Το άκρο του καμπυλωμένου συνδετήρα με την ένδειξη M μπορεί να δεχθεί άξονες αλλά μπορεί επίσης να χωρέσει σε συνδετήρες.

LABEL	NAME	COLOR
A	2M cross block	orange
B	3M cross block	black
C	3M double cross block	white
D	3M cross block 2x3	grey
E	1M beam with two axles	dark grey
F	connector with axle holes	yellow
G	connector with axle and axle hole	grey
H	angle connector #2	orange
I	gearbox cross block	black
J	pin joiner round	black
K	2M flexible axle joiner	grey
L	axle connector	red
M	curved tube with axle holes	black





LABEL	NAME	COLOR
A	8z gear	dark grey
B	12z double-bevel gear	black
C	12z bevel gear	tan
D	20z bevel gear	tan
E	24z gear	dark grey
F	36z double-bevel gear	black
G	28z turntable	black/grey
H	gear rack	black

Γρανάζια

Τα γρανάζια συνδέονται με άλλα οδοντωτά μέρη για μετάδοση κίνησης.

Η εικόνα αριστερά δείχνει τα γρανάζια ορισμένα από τα οποία περιλαμβάνονται στο σετ, με τα αντίστοιχα ονόματα τους στον πίνακα.

Τα γρανάζια LEGO προσδιορίζονται από τον αριθμό των δοντιών τους, όπως υποδεικνύεται στο όνομά τους ακολουθούμενο από z. για παράδειγμα, ένα γρανάζι 24 δοντιών ονομάζεται γρανάζι 24z.

Τα περισσότερα γρανάζια έχουν πάχος 1M, με εξαίρεση τα κωνικά γρανάζια 12z και 20z, που έχουν και τα δύο πάχος μισής μονάδας.

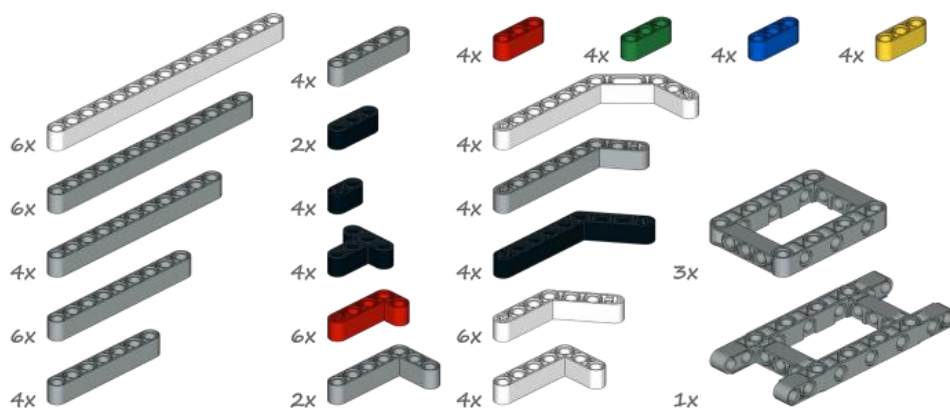
Το γρανάζι 24z (E) είναι ένα γρανάζι με στροφέιο,

Το πικάπ 28z (G) μπορεί να χρησιμοποιηθεί για τη σύνδεση περιστρεφόμενων εξαρτημάτων που δεν μπορούν να αποσπαστούν.

Μια **σχάρα ταχυτήτων** (H) είναι σαν ένα ξετυλιγμένο γρανάζι που μας επιτρέπει να μετατρέπουμε την περιστροφή σε γραμμική κίνηση.

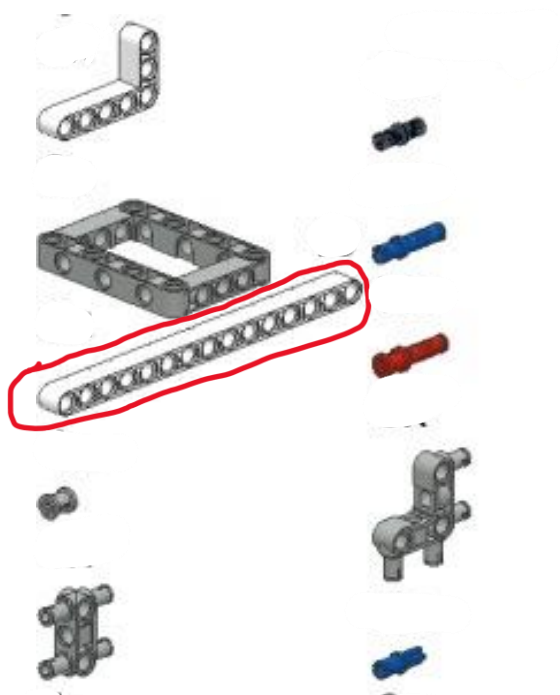


...μέσα από ασκήσεις



Υπάρχει κάποιος από τους γνωστούς τύπους συνδέσμων, στα **παθητικά στοιχεία (*)** που απεικονίζονται στη φωτογραφία αριστερά;

Απάντηση
Όχι



Επιλέξτε να κυκλώσετε αποκλειστικά έναν **σύνδεσμο (**)** με τον οποίο θα μπορούσατε να συνδέσετε παράλληλα τρία στοιχεία του τύπου που σημειώνεται με κόκκινο περίγραμμα.

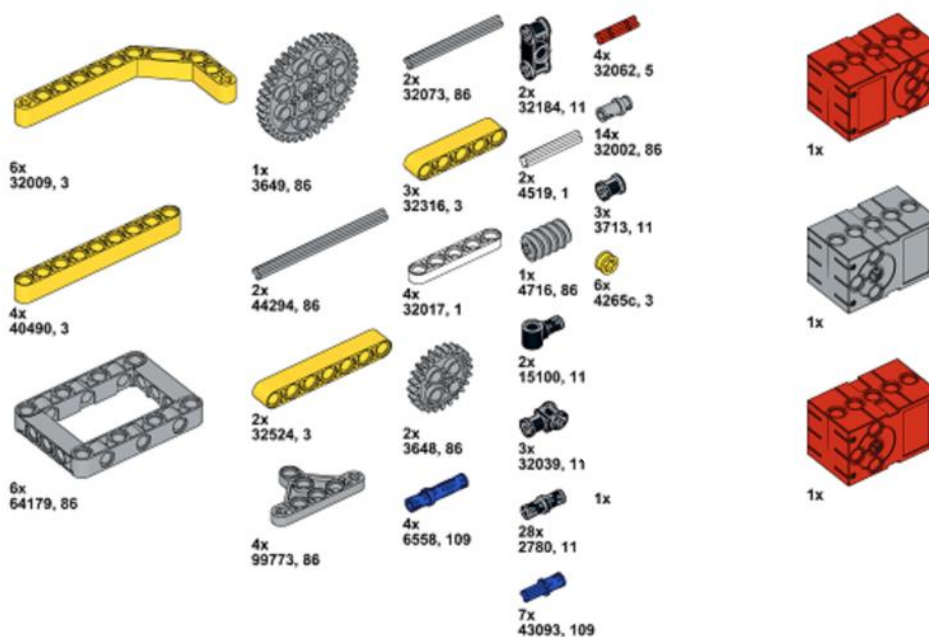
Απάντηση



(*) Παθητικά στοιχεία: Όλα τα στοιχεία που είναι απαραίτητα στην κατασκευή και δεν δέχονται ούτε εκπέμπουν κάποιο ψηφιακό σήμα ή ηλεκτρικό ερέθισμα.

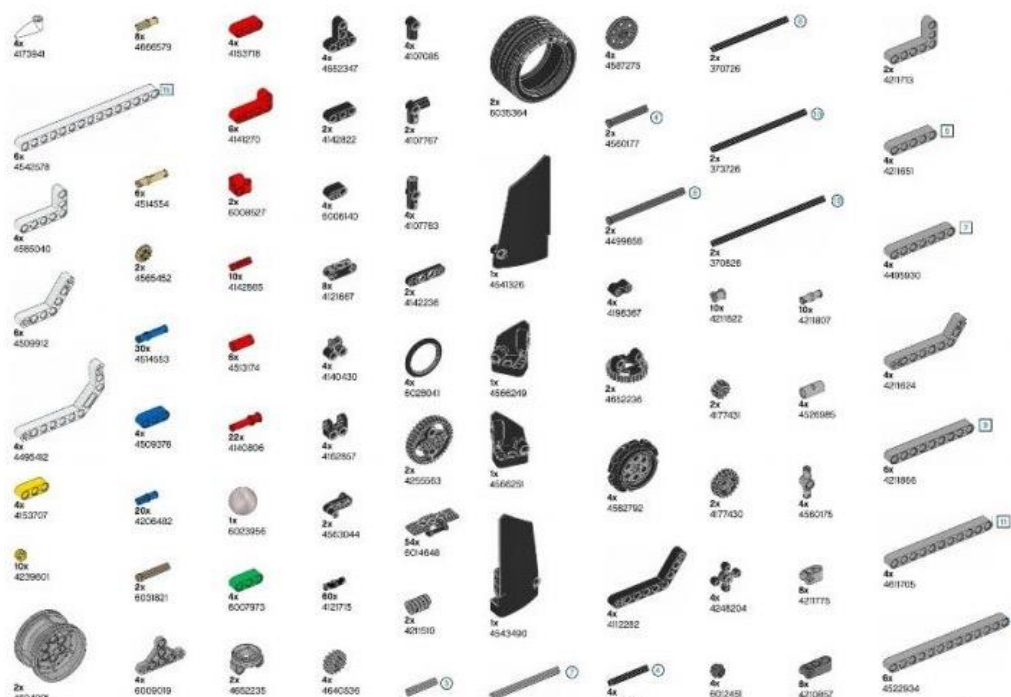
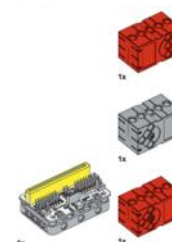
() Σύνδεσμος:** Παθητικό στοιχείο που βοηθά στη σύνδεση δύο ή περισσότερων στοιχείων (ενεργών ή/και παθητικών) σε μια κατασκευή.





**Πόσα ενεργά
στοιχεία (***)
υπάρχουν
στην διπλανή
εικόνα:**

Απάντηση

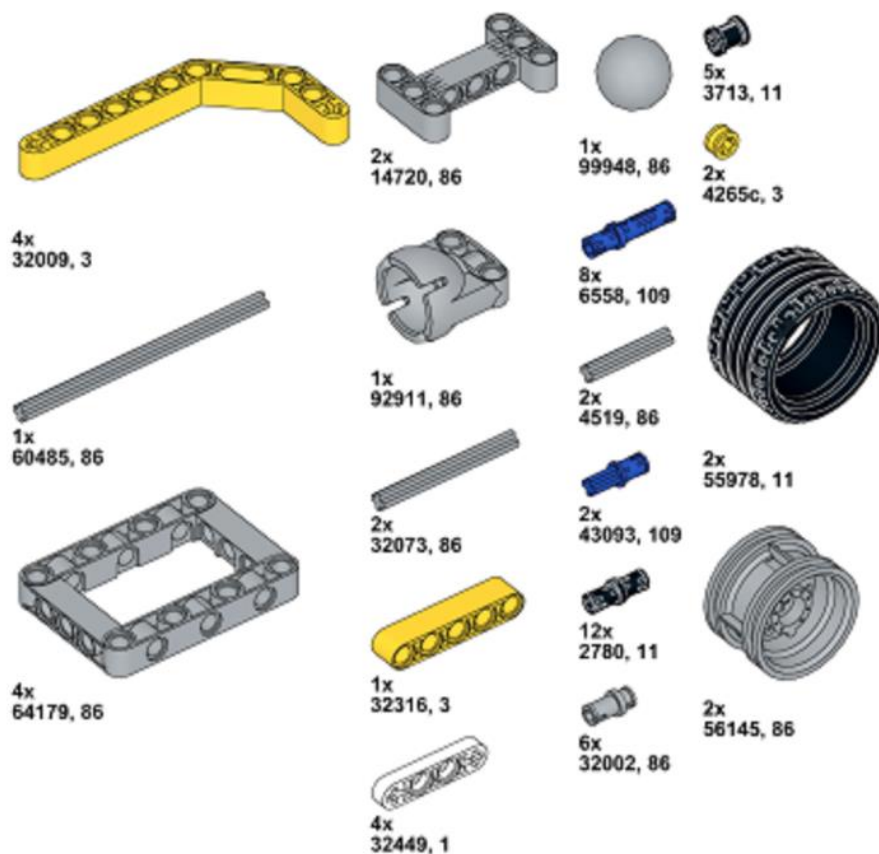


Ποια και πόσα
στοιχεία
θα χρειαστείτε
για να φτιάξετε
ένα τετράγωνο
σταθερό
πλαίσιο
με τη
μεγαλύτερη
δυνατή πλευρά
σε μήκος;

Παρακαλούμε να κυκλώσετε τα στοιχεία και να γράψετε τον αριθμό που θα χρειαστείτε για την κατασκευή αυτή.

(* Ενεργά στοιχεία:** Όλα τα στοιχεία που εκπέμπουν ή δέχονται κάποιο ψηφιακό σήμα ή ηλεκτρικό ερέθισμα, ή/και κάνουν υπολογισμούς και επιστρέφουν τιμές.





Ποια από τα στοιχεία που εικονίζονται αριστερά, συνδεόμενα μεταξύ τους μπορούν να αξιοποιηθούν ως παθητικός τροχός;

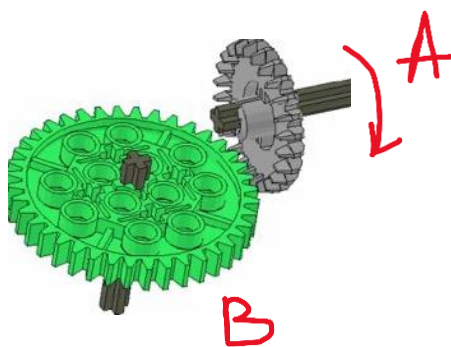
Απάντηση



Πόσα συνολικά διαφορετικά παθητικά στοιχεία υπάρχουν στην διπλανή εικόνα;

Απάντηση
8

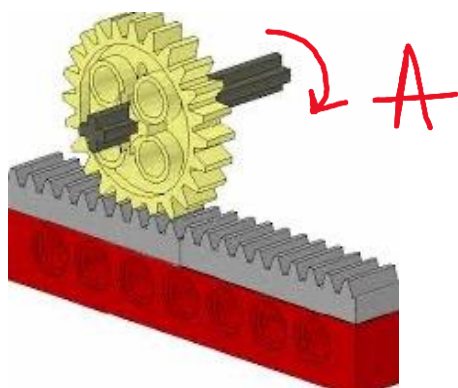




Αν στη θέση Α προσαρμόσουμε έναν κινητήρα που θα περιστραφεί με την φορά που δείχνει το βέλος στην διπλανή φωτογραφία, θα περιστραφεί... από αριστερά προς τα δεξιά ή από δεξιά προς τα αριστερά, το γρανάζι Β;

Απάντηση

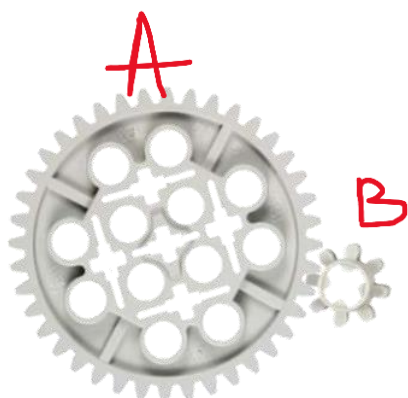
Το γρανάζι Β θα περιστραφεί... από αριστερά προς τα δεξιά.



Τι θα συμβεί αν στη θέση Α, τοποθετήσουμε έναν κινητήρα το οποίο θα περιστραφεί το γρανάζι με τη φορά του βέλους μισή στροφή; Δηλαδή, τι θα μετακινηθεί (εκτός από το γρανάζι και τον άξονα που θα περιστραφούν) και προς ποια κατεύθυνση;

Απάντηση

Θα μετακινηθεί το οριζόντιο στέλεχος προς αριστερά όπως το κοιτάμε

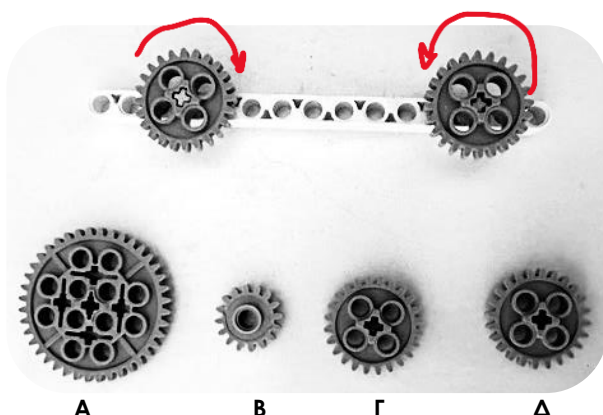


Αν στο γρανάζι Α προσαρμόσουμε έναν κινητήρα, το γρανάζι Β θα περιστραφεί περισσότερες ή λιγότερες φορές από το Α;

Απάντηση

Το γρανάζι Β θα περιστραφεί περισσότερες φορές





Ποια γρανάζια πρέπει να χρησιμοποιήσουμε ενδιάμεσα των δύο ώστε αν κινηθεί το ένα προς την κατεύθυνση του βέλους να κινηθεί και το άλλο προς την κατεύθυνση του βέλους του.

Απάντηση (Γ και Δ)



Σύνδεση με την Υπολογιστική Σκέψη

Είναι σύνηθες για τους προγραμματιστές να κάνουν προσομοιώσεις μηχανών του πραγματικού κόσμου. Αυτό συμβαίνει και σε αυτό το πρόβλημα.

Το πρόβλημα μπορεί να λυθεί χρησιμοποιώντας την υπολογιστική σκέψη, και συγκεκριμένα την διάσταση της αφαίρεσης, με έναν ή και τους δύο παρακάτω τρόπους:

1. Μπορούμε να τοποθετήσουμε δοκιμαστικά τα σετ γραναζιών που δίνονται και να εντοπίσουμε αυτό που δίνει το ζητούμενο αποτέλεσμα.
2. Μπορούμε να παρατηρήσουμε τα δεδομένα και να δημιουργήσουμε έναν κανόνα: τα εναλλασσόμενα γρανάζια στρέφονται προς αντίθετες κατευθύνσεις.

Δεν υπάρχει λόγος ανησυχίας αν δεν έχετε δώσει όλες τις απαντήσεις (πριν τις διαβάσετε). Αυτές θα προκύψουν από την ανάγνωση των επόμενων σελίδων.



Εξοπλισμός S1 & S2 ARDicon



Ας ανοίξουμε το/τα κουτί/κουτιά





Ο μικροελεγκτής ARD:icon

Το ARD:icon είναι ένας μικροελεγκτής που βασίζεται στην ανοιχτή αρχιτεκτονική της οικογένειας ελεγκτών ARDUINO, ειδικά σχεδιασμένος από την Polytech για απλή χρήση “plug and play” από μαθητές.

Χρησιμοποιεί οδηγούς ανοιχτού κώδικα και ένα απλό αλλά ισχυρό και ευέλικτο περιβάλλον προγραμματισμού προκειμένου να κάνει τον προγραμματισμό προσβάσιμο στο γενικό (μαθητικό) κοινό.

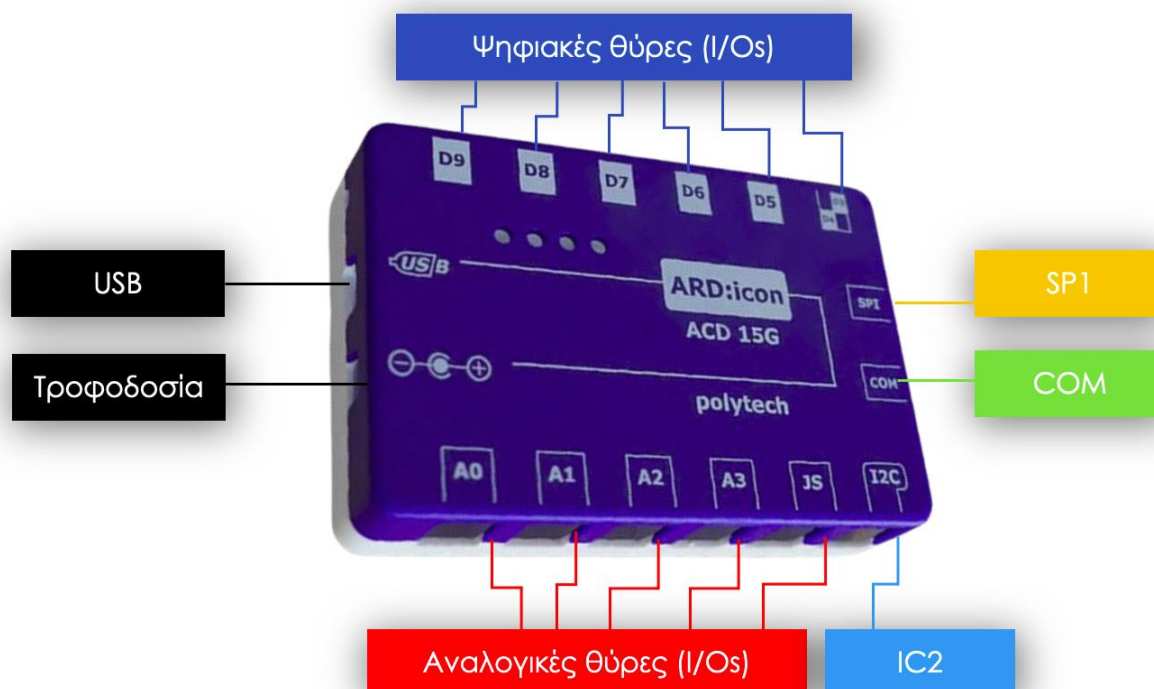
Η συσκευή μικροελεγκτή είναι ενσωματωμένη σε ανθεκτικό πλαστικό κουτί. Διαθέτει σημάδια στην επιφάνειά του, έτσι ώστε οι μαθητές να μπορούν να αναγνωρίσουν κάθε είσοδο, έξοδο, επικοινωνία, τροφοδοσία και άλλη απαραίτητη θύρα σύνδεσης. Διαθέτει ενδείξεις φώτων λειτουργίας για διάφορες καταστάσεις (Λήψη, Εκτέλεση, Μεταγλώττιση).

Διαθέτει φωτεινές ενδείξεις λειτουργίας για διάφορες καταστάσεις (Download, Run, Compile).

Η σύνδεση των αισθητήρων και των μικροσυσκευών στον ARD:icon γίνεται μέσω RJ11 καλωδίων



Τεχνικά χαρακτηριστικά:



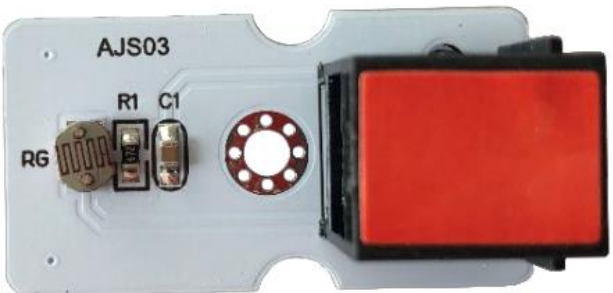
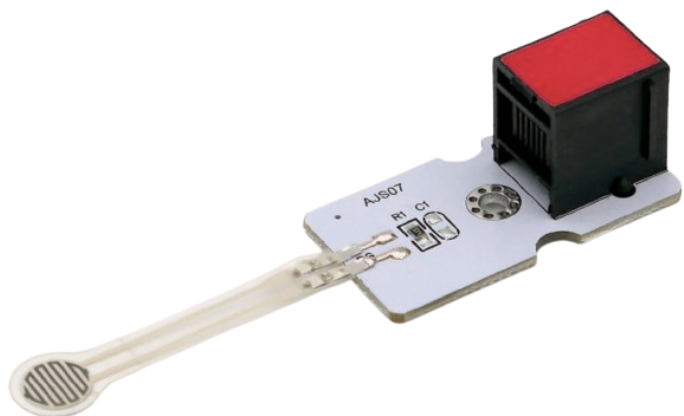
- **CPU:** ATmega328P-PU
- **Μνήμη Flash:** 32KB (ATmega328) από τα οποία 0.5 KB χρησιμοποιούνται από το bootloader
- **SRAM:** 2KB
- **Χωρητικότητα αποθήκευσης EEPROM:** 1KB
- **Συχνότητα ρολογιού:** 16 MHz
- Επιλογή επανεκκίνησης
- Δυνατότητα σύνδεσης μέσω USB, σε συσκευή με Android ή Windows OS
- **Τάση λειτουργίας:** 5V
- **Τάση Εισόδου (συνιστώμενη):** DC 7-12V
- **Ψηφιακές θύρες διπλού PIN (1):** D3/D4
- **Ψηφιακές θύρες μονού PIN (5):** D5-D9, εκ των οποίων 3 υποστηρίζουν PWM έξοδο (D5, D6, D9)
- **Αναλογικές θύρες (4):** A0-A3
- **Θύρα Joystick (1):** JS
- **Θύρα COM (1)**
- **Θύρα SPI (1)**
- **Θύρα διασύνδεσης I2C (1)**
- **Συνεχές ρεύμα ανά I/O θύρα:** 20 mA

Τροφοδοτικό

Όταν συνδέεται σε tablet ή υπολογιστή μέσω θύρας USB, **δεν απαιτείται εξωτερική τροφοδοσία. Δεν απαιτούνται μπαταρίες** για τη λειτουργία του σετ.



Αναλογικοί Αισθητήρες

Περιγραφή	Εικόνα	Μπορεί να συνδεθεί στις υποδοχές
Αισθητήρας ήχου - Μικρόφωνο	 A photograph of the AJS02 microphone module. It features a black circular microphone on the left, a blue potentiometer in the center, and a red rectangular sensor on the right. Various electronic components like resistors and capacitors are visible on the PCB.	A0, A1, A2, A3
Αισθητήρας Φωτός LDR (Φωτοκύτταρο) AJS03 Φωτοαντίσταση (ανίχνευση επιπέδου φωτός)	 A photograph of the AJS03 LDR module. It has a small LDR sensor on the left, a potentiometer in the center, and a red rectangular sensor on the right.	A0, A1, A2, A3
Επίπεδο τάσης (ποτενσιόμετρο) AJS06	 A photograph of the AJS06 potentiometer module. It features a large red potentiometer knob on the left, a potentiometer in the center, and a red rectangular sensor on the right.	A0, A1, A2, A3
Αισθητήρας πίεσης λεπτής μεμβράνης AJS07	 A photograph of the AJS07 pressure sensor module. It has a long, thin, flexible sensor strip on the left, a potentiometer in the center, and a red rectangular sensor on the right.	A0, A1, A2, A3



Αναλογικός αισθητήρας ήχου (μικρόφωνο) AJS02

Ο αναλογικός αισθητήρας ήχου είναι μια συσκευή σχεδιασμένη για την ανίχνευση ήχων στο περιβάλλον, επιτρέποντας τη δημιουργία διαδραστικών εφαρμογών και κυκλωμάτων. Με την κατάλληλη χρήση, μπορείτε να κατασκευάσετε συσκευές όπως διακόπτες που ενεργοποιούνται με φωνητικές εντολές.

Χαρακτηριστικά

- **Διεπαφή:** RJ11
- **Τάση Εισόδου:** 3.3V – 5V
- **Φορτίο Λειτουργίας:** <10mA
- **Τύπος Συσκευής:** Αναλογικός



Σημεία που Ξεχωρίζουν

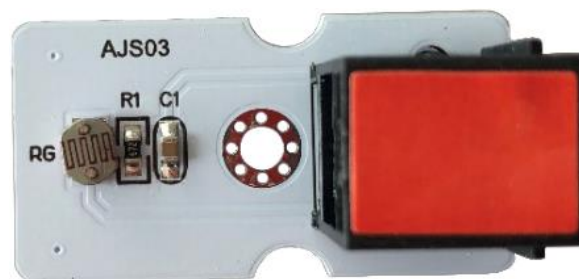
- **Ευαισθησία ήχου:** Η ευαισθησία του αισθητήρα μπορεί να ρυθμιστεί μέσω του ενσωματωμένου ποτενσιόμετρου, επιτρέποντας την προσαρμογή ανάλογα με το περιβάλλον και τις απαιτήσεις της εφαρμογής.

Αναλογικός αισθητήρας φωτός LDR ή Φωτοκύταρο

Ο **Αισθητήρας Φωτός LDR** (Light Dependent Resistor ή φωτοκύταρο) είναι μια συσκευή που αλλάζει την αντίστασή της ανάλογα με την ένταση του φωτός που προσπίπτει στην επιφάνειά της. Χρησιμοποιείται ευρέως σε εφαρμογές ανίχνευσης φωτός και συστήματα αυτοματισμού, όπως στον αυτόματο φωτισμό δρόμων ή σε συστήματα ασφαλείας.

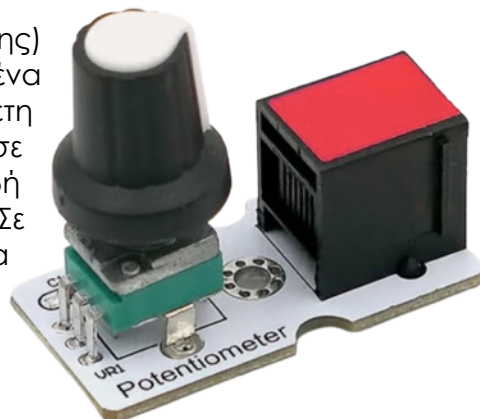
Βασικά Χαρακτηριστικά:

- **Αρχή λειτουργίας:** Η αντίσταση του LDR μειώνεται καθώς αυξάνεται η ένταση του φωτός και αυξάνεται όταν το φως μειώνεται. Αυτή η ιδιότητα επιτρέπει τη μέτρηση της φωτεινότητας και μπορεί να χρησιμοποιηθεί σε συστήματα που ανταποκρίνονται σε αλλαγές του περιβάλλοντος φωτισμού.
- **Ευαισθησία:** Ο LDR έχει υψηλή ευαισθησία στο φως, καθιστώντας τον κατάλληλο για την ανίχνευση ακόμα και μικρών μεταβολών στην ένταση φωτός.
- **Ταχύτητα απόκρισης:** Ο αισθητήρας ανταποκρίνεται γρήγορα σε μεταβολές του φωτός, καθιστώντας τον ιδανικό για εφαρμογές που απαιτούν άμεση αντίδραση.
- **Διεπαφή:** RJ11.
- **Τάση λειτουργίας:** 5V.
- **Τύπος αισθητήρα:** Αναλογικός.



Αναλογικός αισθητήρας γωνίας/Ροοστάτης AJS06

Ο **αναλογικός αισθητήρας γωνίας** (ή ροοστάτης) λειτουργεί ως μεταβλητός αντιστάτης, αποτελώντας ένα κλασικό παράδειγμα κυκλώματος μεταβλητού διαιρέτη τάσης. Η έξοδος του αισθητήρα μπορεί να χωριστεί σε 1023 διακριτά επίπεδα τάσης, επιτρέποντας ακριβή μέτρηση της γωνίας ή άλλων αναλογικών παραμέτρων. Σε συνδυασμό με άλλους αισθητήρες, μπορείτε να δημιουργήσετε και να υλοποιήσετε έργα που βασίζονται στην ανάγνωση αναλογικών τιμών μέσω των I/O θυρών.



Βασικά Χαρακτηριστικά:

- **Διεπαφή:** RJ11
- **Τάση λειτουργίας:** 3.3V – 5V
- **Τύπος αισθητήρα:** Αναλογικός (Analog)

Αυτός ο αισθητήρας είναι ιδανικός για εφαρμογές που απαιτούν ακριβή έλεγχο μέσω αναλογικών μετρήσεων, προσφέροντας μεγάλη ευελιξία στα έργα σας.

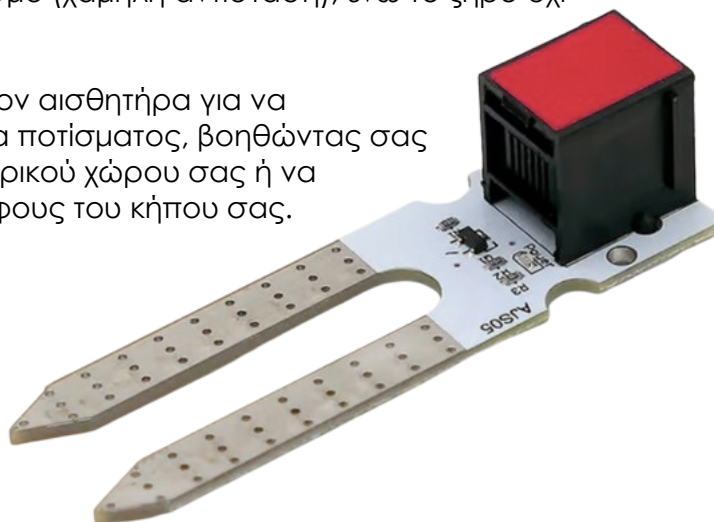
Αισθητήρας υγρασίας εδάφους AJS05

Αυτός ο αισθητήρας υγρασίας εδάφους Plug & Play μπορεί να μετρήσει πόση υγρασία υπάρχει στο έδαφος γύρω του. Είναι ιδανικό για να παρακολουθείτε τον αστικό σας κήπο ή τη στάθμη του νερού του φυτού σας.

Ο αισθητήρας λειτουργεί χρησιμοποιώντας δύο ανιχνευτές για να περάσει ένα ρεύμα μέσα από το έδαφος και στη συνέχεια μετρά την αντίσταση. Το υγρό έδαφος μεταφέρει εύκολα τον ηλεκτρισμό (χαμηλή αντίσταση), ενώ το ξηρό όχι (υψηλή αντίσταση).

Μπορείτε να χρησιμοποιήσετε αυτόν τον αισθητήρα για να δημιουργήσετε ένα αυτόματο σύστημα ποτίσματος, βοηθώντας σας να θυμάστε να ποτίζετε τα φυτά εσωτερικού χώρου σας ή να παρακολουθείτε την υγρασία του εδάφους του κήπου σας.

- **Διεπαφή:** RJ11
- **Τροφοδοτικό:** 3,3V ή 5V
- **Ρεύμα λειτουργίας:** $\leq 20\text{mA}$
- **Τάση εξόδου:** 0-2,3V
- **Τύπος αισθητήρα:** Αναλογικός
- **Φινίρισμα επιφάνειας:** κασσίτερος εμβάπτισης



Αισθητήρας υγρασίας/στάθμης νερού AJS04

Αυτός ο αισθητήρας είναι εύκολος στη χρήση, φορητός και έχει σχεδιαστεί για να ανιχνεύει και να μετράει τα επίπεδα και τις σταγόνες νερού. Μπορεί να μετρήσει την ποσότητα του νερού μέσω μιας σειράς εκτεθειμένων παράλληλων γραμμών.

Τα βασικά χαρακτηριστικά αυτού του μικρού αισθητήρα περιλαμβάνουν:

- Ομαλή μετατροπή μεταξύ στάθμης νερού και αναλογικών σημάτων
- Υψηλή ευελιξία με αναλογική έξοδο
- Χαμηλή κατανάλωση ρεύματος και υψηλή ευαισθησία
- Συμβατότητα με μικροεπεξεργαστές και άλλα λογικά κυκλώματα, κατάλληλα για διάφορες πλακέτες ανάπτυξης και ελεγκτές όπως Arduino, μικροελεγκτές STC και μικροελεγκτές AVR.
- Διεπαφή: RJ11
- Τάση λειτουργίας: DC5V
- Ρεύμα λειτουργίας: <20mA
- Τύπος αισθητήρα: Αναλογικός
- Περιοχή ανίχνευσης: 40mm x 16mm
- Θερμοκρασία λειτουργίας: 10%~90% χωρίς συμπύκνωση



Αισθητήρας πίεσης λεπτής μεμβράνης AJS07

Ο **αισθητήρας πίεσης λεπτής μεμβράνης** είναι μια ευέλικτη και ευαίσθητη συσκευή, σχεδιασμένη για την ακριβή ανίχνευση και μέτρηση της πίεσης. Με τον εξαιρετικά λεπτό και εύκαμπτο σχεδιασμό του, είναι ιδανικός για εφαρμογές που απαιτούν ακριβείς μετρήσεις πίεσης ή δύναμης σε περιορισμένο χώρο.

Αρχή Λειτουργίας:

- **Αντίσταση και πίεση:** Η αντίσταση του αισθητήρα μεταβάλλεται ανάλογα με την πίεση που ασκείται πάνω του. Όσο αυξάνεται η πίεση, τόσο μειώνεται η αντίσταση. Αυτές οι μεταβολές στην αντίσταση μετατρέπονται σε



ηλεκτρικό σήμα, το οποίο έναν μικροελεγκτή, επιτρέποντας

την ανίχνευση της πίεσης.

Χαρακτηριστικά:

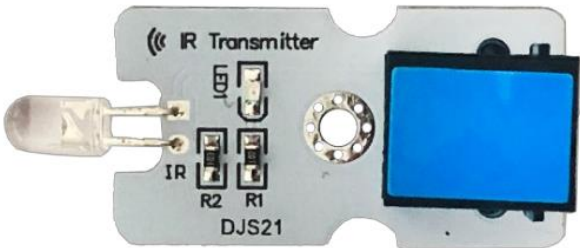
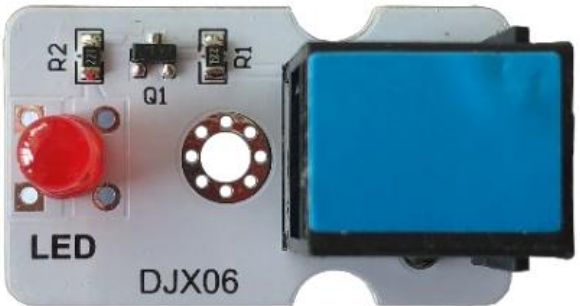
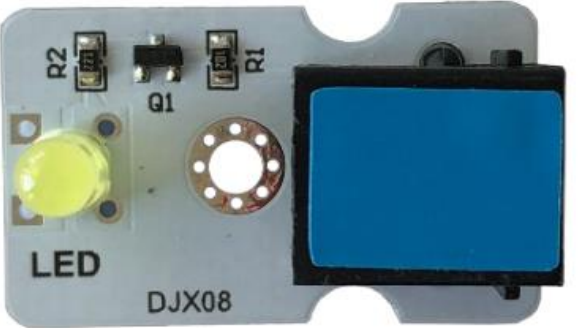
- **Εξαιρετικά λεπτός και εύκαμπτος:** Το πάχος του αισθητήρα είναι μικρότερο από 0,25mm, καθιστώντας τον κατάλληλο για εφαρμογές όπου η ευκαμψία και η λεπτότητα είναι κρίσιμα στοιχεία.
- **Αδιάβροχη κατασκευή:** Κατασκευασμένος από ανθεκτικά υλικά, ο αισθητήρας είναι αδιάβροχος και μπορεί να χρησιμοποιηθεί σε υγρά περιβάλλοντα.
- **Εύρος ανίχνευσης πίεσης:** Μπορεί να ανιχνεύσει πιέσεις από 0 έως 0,5 KG, προσφέροντας υψηλή ακρίβεια σε λεπτές μετρήσεις δύναμης.
- **Ανθεκτικότητα:** Σχεδιασμένος για πάνω από 100.000 χρήσεις, ο αισθητήρας είναι κατάλληλος για μακροχρόνιες και απαιτητικές εφαρμογές.

Τεχνικά Χαρακτηριστικά:

- **Τύπος αισθητήρα:** Αναλογικός
- **Διεπαφή:** RJ11
- **Τάση λειτουργίας:** 3,3V-5V (DC)
- **Εύρος ανίχνευσης:** 0-0,5 KG
- **Πάχος:** <0,25mm
- **Σημείο απόκρισης:** <20g
- **Επαναληψιμότητα:** <±5,8% (στο 50% του φορτίου)
- **Ακρίβεια:** ±2,5% (στο 85% του εύρους)
- **Χρόνος απόκρισης:** <1ms
- **Χρόνος αποκατάστασης:** <15ms
- **Θερμοκρασία λειτουργίας:** -20°C έως 60°C

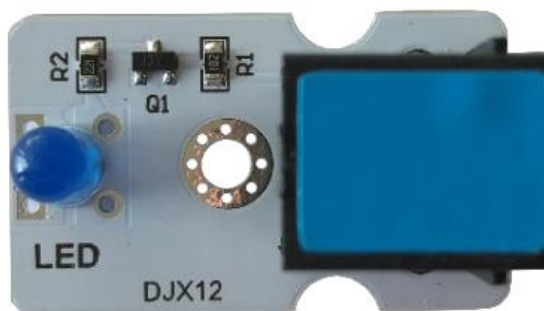


Ψηφιακοί Ενεργοποιητές

Περιγραφή	Εικόνα	Μπορεί να συνδεθεί στις υποδοχές
Πομπός υπερύθρων	 The image shows an IR Transmitter module (DJS21) on a white PCB. It features a clear IR LED on the left, a potentiometer in the center, and a blue rectangular component on the right. Labels include 'IR Transmitter', 'LED', 'IR', 'R2', 'R1', and 'DJS21'.	D5, D6, D7, D8, D9
Κόκκινο led	 The image shows a Red LED module (DJX06) on a white PCB. It features a red LED on the left, a potentiometer in the center, and a blue rectangular component on the right. Labels include 'LED', 'R2', 'Q1', 'R1', and 'DJX06'.	D5, D6, D7, D8, D9
Πράσινο led	 The image shows a Green LED module (DJX07) on a white PCB. It features a green LED on the left, a potentiometer in the center, and a blue rectangular component on the right. Labels include 'LED', 'R2', 'Q1', 'R1', and 'DJX07'.	D5, D6, D7, D8, D9
Κίτρινο led	 The image shows a Yellow LED module (DJX08) on a white PCB. It features a yellow LED on the left, a potentiometer in the center, and a blue rectangular component on the right. Labels include 'LED', 'R2', 'Q1', 'R1', and 'DJX08'.	D5, D6, D7, D8, D9

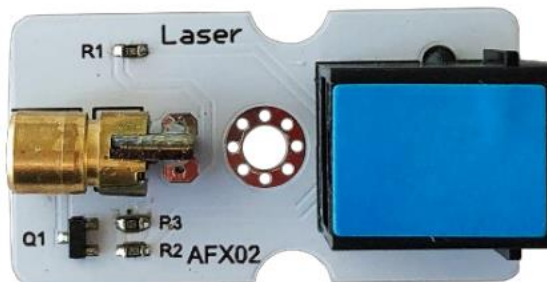


Μπλε led



D5, D6, D7, D8,
D9

Μονάδα laser



D5, D6, D7, D8,
D9

Μονάδα
επέκτασης (για
σύνδεση του
σέρβο κινητήρα
αλλά και άλλων
επεκτάσεων)



D5, D6, D7, D8,
D9



LED (Κόκκινο) DJX06

Αυτό το κόκκινο LED προσφέρει ένα εντυπωσιακό και ευδιάκριτο φωτισμό, ιδανικό για ποικιλία ηλεκτρονικών εφαρμογών. Είναι μια εξαιρετική επιλογή για έργα που απαιτούν ελκυστικό και φωτεινό κόκκινο φως.

Χαρακτηριστικά

- **Διεπαφή:** RJ11
- **Τύπος Συσκευής:** Ψηφιακός
- **Τάση Λειτουργίας:** 5V
- **Χρώμα LED:** Κόκκινο



Σημεία που Ξεχωρίζουν

- **Εύκολη σύνδεση:** Η διεπαφή RJ11 επιτρέπει γρήγορη και απλή εγκατάσταση.
- **Φωτεινότητα:** Το ζωντανό κόκκινο χρώμα προσφέρει εξαιρετική ορατότητα, κάνοντάς το ιδανικό για διακοσμητικό φωτισμό ή ενδείξεις.

Ενεργός βομβητής (ηχείο) AJX03

Αν θέλετε να δημιουργήσετε έναν δυνατό ήχο, αυτή η μονάδα βομβητή είναι η ιδανική λύση! Απλώς συνδέστε τη σε πηγή τροφοδοσίας 3,3V έως 5V και θα παράγει έναν ισχυρό ηχητικό σήμα. Είναι μια απλή αλλά αποδοτική μονάδα ήχου που μπορεί να ελεγχθεί με ψηφιακά σήματα υψηλής ή χαμηλής τάσης.



Χαρακτηριστικά:

- **Διπλή λειτουργία:** Μπορεί να ρυθμιστεί είτε να παράγει απλό ήχο είτε παράγει μελωδία.
- **Συμπαγής Σχεδίαση:** Αυτή η μονάδα είναι εύκολη στη χρήση και τοποθέτηση, ιδανική για προγράμματα που απαιτούν ήχο.
- **Καθημερινές Εφαρμογές:** Σημαντική συμβολή σε πολλές καθημερινές συσκευές, όπως υπολογιστές, ψυγεία και τηλέφωνα.

Προδιαγραφές:

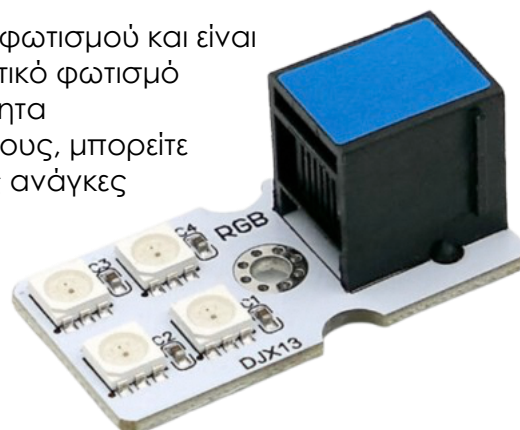
- **Διεπαφή:** RJ11
- **Τύπος:** Ψηφιακός

Τάση Λειτουργίας: 3,3V έως 5V



LED (RGB) DJX13

Αυτό το RGB LED προσφέρει ευέλικτες επιλογές φωτισμού και είναι ιδανικό για ποικιλία εφαρμογών, από διακοσμητικό φωτισμό έως ενδείξεις και διαδραστικά έργα. Με δυνατότητα προσαρμογής σε διάφορους χρωματικούς τόνους, μπορείτε να δημιουργήσετε ατμόσφαιρες ανάλογα με τις ανάγκες σας.



Χαρακτηριστικά

- **Διεπαφή:** RJ11
- **Τάση Λειτουργίας:** DC 5V
- **Ένταση:** 0.1W
- **Πηγή Φωτός:** SMD 5050 RGB
- **Μοντέλο IC:** WS2812 (4 κανάλια)
- **Τόνοι του Γκρι:** 256 επίπεδα
- **Γωνία Ακτινοβολίας:** 180°
- **Χρώματα:** Δυνατότητα ρύθμισης σε λευκό, κόκκινο, κίτρινο, μπλε, πράσινο και άλλα.

Σημεία που Ξεχωρίζουν

- **Ευελιξία Χρώματος:** Μπορείτε να επιλέξετε ανάμεσα σε ένα ευρύ φάσμα χρωμάτων, κάνοντάς το ιδανικό για κάθε εφαρμογή.
- **Απλή Σύνδεση:** Χάρη στη διεπαφή RJ11, η εγκατάσταση είναι εύκολη και γρήγορη.
- **Υψηλή Απόδοση:** Η τεχνολογία SMD 5050 παρέχει φωτεινότητα και σταθερότητα.

Αυτό το RGB LED είναι ιδανικό για έργα που απαιτούν προσαρμοστικότητα στον φωτισμό, προσφέροντας δημιουργικές δυνατότητες σε κάθε εφαρμογή σας.

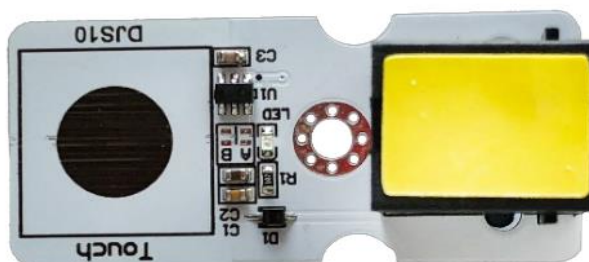


Ψηφιακοί Αισθητήρες

Περιγραφή **Εικόνα**

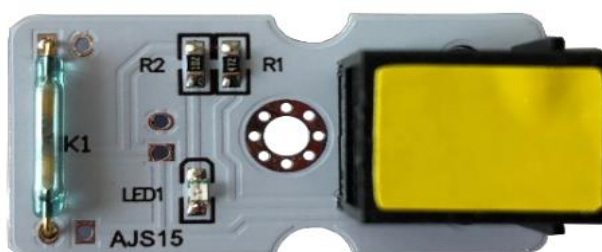
**Μπορεί να
συνδεθεί στις
υποδοχές**

Κουμπι
αφής



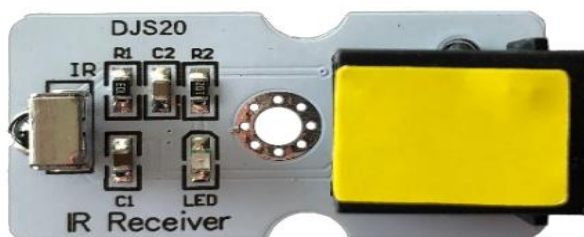
D5, D6, D7, D8, D9

Μαγνητικός
διακόπτης



D5, D6, D7, D8, D9

Δέκτης
υπερύθρων



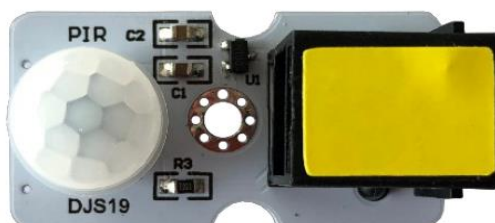
D5, D6, D7, D8, D9

Αισθητήρες
Θερμοκρασ
ίας και
υγρασίας



D5, D6, D7, D8, D9

Αισθητήρας
κίνησης
(PIR)



D5, D6, D7, D8, D9



Αισθητήρας
μαγνητικού
πεδίου

D5, D6, D7, D8, D9



Αισθητήρας κίνησης PIR (Pyroelectric Infrared) DJS19

Ο αισθητήρας κίνησης PIR (Pyroelectric Infrared) είναι μια εξαιρετικά χρήσιμη συσκευή που ανιχνεύει κίνηση ανθρώπων ή ζώων μέσω της υπέρυθρης ακτινοβολίας που εκπέμπουν τα σώματα. Ακολουθούν οι κύριες λειτουργίες και χαρακτηριστικά του αισθητήρα PIR:

Λειτουργία

- Ανίχνευση Υπέρυθρης Ακτινοβολίας:** Ο αισθητήρας ανιχνεύει την υπέρυθρη ακτινοβολία που εκπέμπουν τα κινούμενα σώματα. Όταν κίνηση, εκπέμπει ένα ηλεκτρικό σήμα, το οποίο μπορεί να χρησιμοποιηθεί για τον έλεγχο άλλων συσκευών, όπως φώτα, συναγερμοί ή συστήματα ασφαλείας.



Διάρκεια Ανίχνευσης

- Ενεργή Περίοδος:** Μετά την ανίχνευση κίνησης, ο αισθητήρας παραμένει ενεργός για 2,3 έως 3 δευτερόλεπτα, ακόμα και αν η κίνηση σταματήσει. Αυτή η δυνατότητα αποτρέπει την περιττή ενεργοποίηση ή απενεργοποίηση από μικρές, τυχαίες κινήσεις.

Τροφοδοσία

- Συμβατότητα Τάσης:** Λειτουργεί με τάση είτε 3,3V είτε 5V, καθιστώντας τον ιδανικό για χρήση με πλατφόρμες μικροελεγκτών όπως το Arduino ή το Raspberry Pi.

Εύρος και Γωνία Ανίχνευσης

- Εύρος Ανίχνευσης:** Ο αισθητήρας μπορεί να ανιχνεύσει κίνηση σε αποστάσεις έως και 7 μέτρων, επαρκές για πολλές εφαρμογές σε μικρούς και μεσαίους χώρους.
- Οπτικό Πεδίο:** Διαθέτει γωνία ανίχνευσης 100°, καλύπτοντας μια εκτενή περιοχή μπροστά του, γεγονός που τον καθιστά κατάλληλο για ανοιχτούς



Αισθητήρας υγρασίας-θερμοκρασίας MJS22

Ο **αισθητήρας υγρασίας-θερμοκρασίας** είναι ιδανικός για εφαρμογές όπου απαιτείται βασική παρακολούθηση της υγρασίας και θερμοκρασίας, όπως σε οικιακούς αυτοματισμούς, γεωργία ή συστήματα HVAC. Παρέχει αξιόπιστες μετρήσεις, αν και δεν είναι σχεδιασμένος για έργα που απαιτούν εξαιρετικά υψηλή ακρίβεια.



Βασικά Χαρακτηριστικά:

- **Τύπος αισθητήρα:** Ψηφιακός (Digital)
- **Μέτρηση υγρασίας:** Μετρά τη σχετική υγρασία σε εύρος 20-90% με ακρίβεια $\pm 5\%$.
- **Μέτρηση θερμοκρασίας:** Το εύρος μέτρησης θερμοκρασίας κυμαίνεται από 0°C έως 50°C με ακρίβεια $\pm 2^{\circ}\text{C}$.
- **Αντίσταση σε παρεμβολές:** Ο αισθητήρας DHT11 έχει καλή αντοχή σε ηλεκτρομαγνητικές παρεμβολές, καθιστώντας τον αξιόπιστο ακόμη και σε περιβάλλοντα με θόρυβο.
- **Χρόνος απόκρισης:** Ανταποκρίνεται γρήγορα στις αλλαγές θερμοκρασίας και υγρασίας.
- **Τροφοδοσία:** Λειτουργεί με τάση 5V.
- **Σταθερότητα:** Προσφέρει υψηλή αξιοπιστία και καλή μακροπρόθεσμη σταθερότητα, ιδανικός για συνεχή παρακολούθηση περιβαλλοντικών συνθηκών και κλιματικών ελέγχων.
- **Διεπαφή:** RJ11.

Διακόπτης πίεσης DJS09

Ο **διακόπτης πίεσης** είναι μια ψηφιακή μονάδα που σας επιτρέπει να ελέγχετε μια πηγή τροφοδοσίας DC με μεγάλη ευκολία.

Χρησιμοποιώντας αυτό το απλό κουμπί πίεσης, μπορείτε να ενεργοποιήσετε ή να απενεργοποιήσετε συσκευές και κυκλώματα με μία μόνο ενέργεια.

Όταν πατάτε το κουμπί, ο διακόπτης στέλνει ένα σήμα **LOW**. Όταν το απελευθερώνετε, το σήμα αλλάζει σε **HIGH**. Αυτή η απλή και αποτελεσματική λειτουργία το καθιστά ιδανικό πολλές εφαρμογές.



Χαρακτηριστικά

- **Διεπαφή:** RJ11
- **Τάση λειτουργίας:** 3.3V – 5V
- **Περιβλημα:** Υψηλής ποιότητας και ανθεκτικό
- **Τύπος αισθητήρα:** Ψηφιακός



Αισθητήρας αποφυγής εμποδίου υπερύθρων DJS17

Ο αισθητήρας αποφυγής εμποδίων υπερύθρων είναι μια χρήσιμη συσκευή που χρησιμοποιείται κυρίως σε ρομπότ για την ανίχνευση εμποδίων και την αποφυγή συγκρούσεων, αξιοποιώντας το υπέρυθρο φως.

Λειτουργία

Ο αισθητήρας διαθέτει έναν πομπό υπερύθρων που εκπέμπει μια δέσμη υπέρυθρου φωτός. Όταν αυτή η δέσμη προσκρούει σε κάποιο εμπόδιο, ανακλάται πίσω στον δέκτη του αισθητήρα. Αυτή η αντανάκλαση ενεργοποιεί το σύστημα, επιτρέποντας στο ρομπότ να σταματήσει ή να αλλάξει κατεύθυνση, αποφεύγοντας έτσι συγκρούσεις.



Χαρακτηριστικά

- **Ρύθμιση Απόστασης:** Ο αισθητήρας επιτρέπει τη ρύθμιση της απόστασης ανίχνευσης από 2 έως 40 cm μέσω ενός ρυθμιστικού κουμπιού, προσφέροντας ευελιξία για διαφορετικές εφαρμογές.
- **Ανθεκτικότητα σε Διαφορετικές Συνθήκες Φωτισμού:** Λειτουργεί αξιόπιστα σε διάφορες συνθήκες φωτισμού, είτε σε φωτεινά είτε σε σκοτεινά περιβάλλοντα.
- **Ευρεία Εφαρμογή:** Ιδανικός για αποφυγή εμποδίων σε ρομπότ, παρακολούθηση γραμμής και παρακολούθηση αντικειμένων.

Προδιαγραφές

- **Διεπαφή:** RJ11
- **Τάση Λειτουργίας:** DC 3,3V έως 5V
- **Ρεύμα Λειτουργίας:** $\geq 20\text{mA}$
- **Εύρος Θερμοκρασίας:** -10°C έως $+50^{\circ}\text{C}$
- **Απόσταση Ανίχνευσης:** 2 έως 40 cm
- **Σήμα Εξόδου:** Τάση TTL
- **Ενεργητική Γωνία:** 35°

Ο αισθητήρας αποφυγής εμποδίων υπερύθρων είναι ένα αξιόπιστο εργαλείο για τη βελτίωση της ασφάλειας και της αποτελεσματικότητας των ρομπότ. Με τις ρυθμίσεις του και την ικανότητά του να λειτουργεί σε διάφορες συνθήκες φωτισμού, αποτελεί μια εξαιρετική επιλογή για ποικιλία εφαρμογών.



Κινητήρας 360 Degree Continuous Rotation

0.7kg Geekservo 360 Degree Continuous Rotation Servo
 Forward Reverse Green 4.TB-6V PWM Control
 Compatible With legoeds Microbit

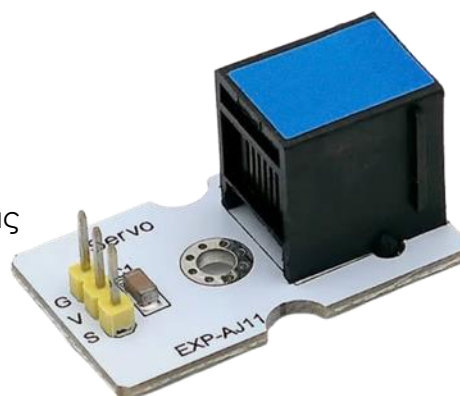
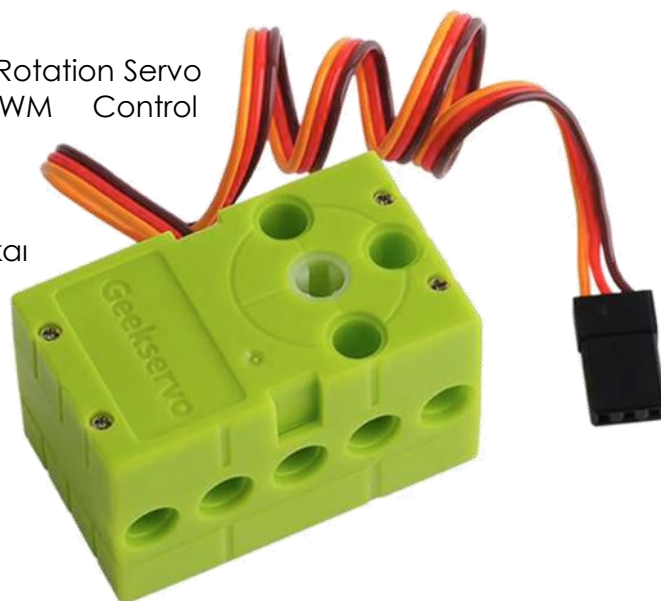
Περιγραφή:

Κινητήρας 360 μοιρών με εμπρός και αντίστροφη συνεχή περιστροφή σε ρομποτική μηχανισμούς.

Σήμα PWM για έλεγχο της ταχύτητας περιστροφής προς τα εμπρός και προς τα πίσω.

- Χρώμα: πράσινο
- Στρέψη: 0,7 kg
- Τάση λειτουργίας: 4,8-6V
- Γωνία χρήσης: 360 μοίρες
- Σήμα PWM: 500-2500
- Μήκος καλωδίου: 25cm
- Τύπος τερματικού: τερματικό DuPont
- Αναλογία μείωσης ταχύτητας: 1:140
- Ταχύτητα προϊόντος: 160/λεπτό (6V)

Ο κινητήρας αυτός είναι συμβατός με lego. Ο άξονάς του περιστρέφεται δεξιόστροφα ή αριστερόστροφα με ταχύτητα που ρυθμίζεται από το λογισμικό, έσω της εντολής servo γωνίας σύμφωνα με τον πίνακα:



Γωνία	Λειτουργία
90	Stop
91 έως 180	Δεξιόστροφη περιστροφή με ταχύτητα που αυξάνει ανάλογα με την αύξηση της τιμής της γωνίας
0 έως 89	Αριστερόστροφη περιστροφή με ταχύτητα που αυξάνει ανάλογα με την μείωση της τιμής της γωνίας



Αισθητήρας απόστασης υπερήχων DJS22

Ο **αισθητήρας απόστασης υπερήχων** χρησιμοποιεί τεχνολογία σόναρ για τη μέτρηση της απόστασης από αντικείμενα. Ο πομπός του αισθητήρα εκπέμπει υπερηχητικά κύματα με συχνότητα 40 KHz, τα οποία ταξιδεύουν στον αέρα. Όταν αυτά τα κύματα συναντήσουν ένα αντικείμενο, ανακλώνται και επιστρέφουν στον δέκτη του αισθητήρα. Ο αισθητήρας μετρά τον χρόνο που μεσολάβησε από την εκπομπή μέχρι την επιστροφή της ηχούς και, με βάση την ταχύτητα του ήχου, υπολογίζει την απόσταση του αντικειμένου.

Βασικά Χαρακτηριστικά:

- **Αρχή λειτουργίας:** Ο αισθητήρας διαθέτει έναν πομπό που εκπέμπει υπερηχητικά κύματα συχνότητας 40 KHz. Όταν αυτά τα κύματα προσκρούσουν σε ένα αντικείμενο, επιστρέφουν στον δέκτη, και η μονάδα υπολογίζει την απόσταση βάσει του χρόνου επιστροφής.
- **Διεπαφή:** RJ11.
- **Τάση λειτουργίας:** 3,3-5V (DC).
- **Ρεύμα λειτουργίας:** 65mA.
- **Συχνότητα λειτουργίας:** 40KHz.
- **Εύρος μέτρησης:** Ελάχιστη απόσταση 4 cm, μέγιστη απόσταση 300 cm.
- **Γωνία μέτρησης:** 15 μοίρες.
- **Σήμα ελέγχου:** Εισερχόμενο σήμα TTL παλμός διάρκειας 10μs.

Αυτός ο αισθητήρας είναι ιδανικός για εφαρμογές ανίχνευσης αποστάσεων, παρέχοντας αξιόπιστη απόδοση με ακρίβεια και σταθερότητα.



Αρχή λειτουργίας

- **Εκπομπή παλμού:** Ο αισθητήρας διαθέτει δύο μέρη:
 - **Εκπομπέας (Transmitter):** Εκπέμπει υπερηχητικά κύματα στη συχνότητα των 40 KHz.
 - **Δέκτης (Receiver):** Ανιχνεύει τα κύματα που ανακλώνται από ένα αντικείμενο.
- Ο μικροελεγκτής (π.χ. Arduino) στέλνει έναν παλμό διάρκειας **10 μικροδευτερολέπτων** στον ακροδέκτη **TRIG** του HC-SR04, που ενεργοποιεί τη διαδικασία μέτρησης.

Υπολογισμός απόστασης

- Ο αισθητήρας μετρά τον χρόνο που απαιτείται για να επιστρέψει το ανακλώμενο κύμα στον δέκτη.
- Ο χρόνος αυτός (Time of Flight - TOF) σχετίζεται με την απόσταση ως εξής: **Απόσταση (σε cm) = Χρόνος (σε μικροδευτερόλεπτα) × 0.034 / 2**
 - **0.034 cm/μs:** Ταχύτητα του ήχου στον αέρα (340 m/s).
 - Ο διαιρητής 2 προκύπτει επειδή το κύμα διανύει δύο φορές την απόσταση (από τον αισθητήρα στο αντικείμενο και επιστροφή).



Οθόνη LCD 16×2 AJX04

Η Οθόνη LCD 16×2 είναι μια βασική αλλά εξαιρετικά χρήσιμη οθόνη υγρών κρυστάλλων που προσφέρει μια ευανάγνωστη απεικόνιση δεδομένων. Είναι ιδανική για ποικιλία ηλεκτρονικών εφαρμογών, παρέχοντας ευκολία στη χρήση και απλή ενσωμάτωση.

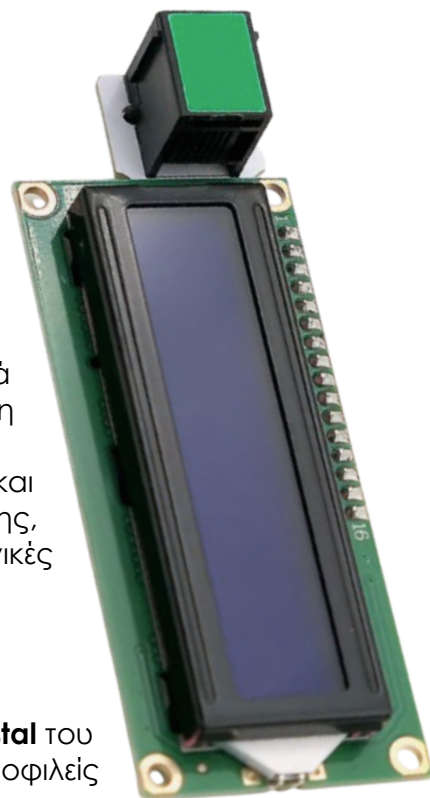
Λειτουργία:

- **Εγκατάσταση:** Η σύνδεση της οθόνης είναι εξαιρετικά απλή, καθώς απαιτεί μόλις μία γραμμή για τη ρύθμιση.
- **Χρήση:** Ιδανική για την εμφάνιση κειμένων και δεδομένων σε έργα όπως συστήματα μέτρησης, διαδραστικά εργαλεία και άλλες ηλεκτρονικές εφαρμογές.

Χαρακτηριστικά:

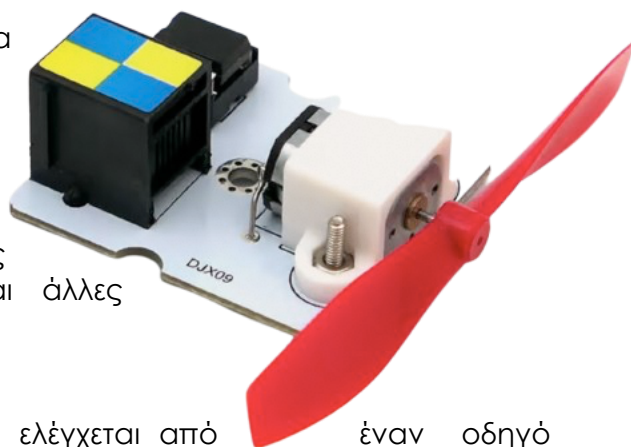
- **Διεπαφή:** RJ11
- **Συμβατότητα:** Συμβατή με τη βιβλιοθήκη **LiquidCrystal** του Arduino, διευκολύνοντας τη χρήση της με δημοφιλείς πλατφόρμες προγραμματισμού.
- **Χρώμα οθόνης:** Λευκό κείμενο σε μπλε φόντο, με μπλε οπίσθιο φωτισμό για βελτιωμένη ορατότητα.
- **Διαστάσεις:** 16 χαρακτήρες x 2 σειρές, επιτρέποντας την απεικόνιση πληροφοριών σε δύο γραμμές.
- **Διεύθυνση I2C:** 0x27, διευκολύνοντας τη σύνδεση μέσω I2C με τον μικροελεγκτή.
- **Τάση τροφοδοσίας:** 5V
- **Ρύθμιση αντίθεσης:** Περιλαμβάνει ποτενσιόμετρο για τη ρύθμιση της αντίθεσης, επιτρέποντας την προσαρμογή της ευκρίνειας του κειμένου.

Αυτή η οθόνη LCD είναι η ιδανική επιλογή για οποιοδήποτε έργο απαιτεί απλή και αποτελεσματική απεικόνιση δεδομένων.



Βηματικός κινητήρας με έλικα (ανεμιστήρας) DJX09

Ο βηματικός κινητήρας με έλικα (ανεμιστήρας) είναι μια πολυλειτουργική μονάδα, ιδανική για διάφορες εφαρμογές και έργα. Προσφέρει ευελιξία και δυνατότητες προσαρμογής, επιτρέποντας την ενσωμάτωσή του σε δημιουργίες όπως αεροσκάφη, ψυκτικά συστήματα και άλλες περιστρεφόμενες μηχανές.



Λειτουργία

- **Οδηγός Κινητήρα:** Ο κινητήρας ελέγχεται από έναν οδηγό που επιτρέπει την περιστροφή προς τα εμπρός και προς τα πίσω, παρέχοντας έλεγχο στη λειτουργία του.
- **Διαχείριση Σύνδεσης:** Συνδέεται εύκολα με μία μόνο γραμμή σε διπλή ψηφιακή θύρα, διευκολύνοντας την ενσωμάτωσή του σε συστήματα και κυκλώματα.
- **Προσαρμοστικότητα:** Χρησιμοποιείται σε ποικιλία έργων, όπως αεροσκάφη με έλικα, συστήματα ψύξης και περιστρεφόμενες μηχανές, προσφέροντας μια συναρπαστική διάσταση στις δημιουργίες σας.
- **Ικανότητα:** Ο ανεμιστήρας έχει αρκετή δύναμη για να σβήσει μια μικρή φλόγα από απόσταση 20 cm, καθιστώντας τον κατάλληλο για εκπαιδευτικά και πρακτικά έργα.
- **Εφαρμογές:** Συχνά χρησιμοποιείται σε κατηγορίες STEM για τη διδασκαλία βασικών εννοιών μηχανικής και ηλεκτρονικής.

Βασικά Χαρακτηριστικά

- **Διεπαφή:** RJ11
- **Διάμετρος Ανεμιστήρα:** 75 mm
- **Τύπος Συσκευής:** Ψηφιακός
- **Τάση Λειτουργίας:** 5V



Σερβοκινητήρας EXP-AJ11, DJX11

Ο σερβοκινητήρας είναι μια ηλεκτρομηχανική συσκευή που παρέχει ακριβή έλεγχο της θέσης και της γωνίας περιστροφής, ιδανική για διάφορες εφαρμογές ρομποτικής και αυτοματισμών. Για να χρησιμοποιήσετε έναν **Micro Servo** με τον μικροελεγκτή ARD:icon, απαιτείται η μονάδα επέκτασης Servo. Αυτή η μονάδα περιλαμβάνει τόσο τη μονάδα Servo όσο και έναν **Micro Servo** με δυνατότητα περιστροφής 180°. Μπορεί να συνδεθεί στις θύρες D5, D6, D7, D8, D9



Χαρακτηριστικά Μονάδας

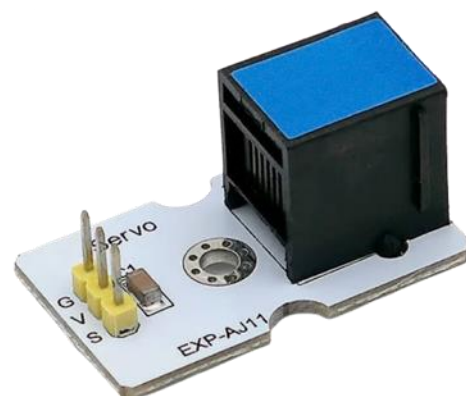
- **Plug & Play Σχεδίαση:** Η μονάδα διαθέτει υποδοχή RJ, διευκολύνοντας τη σύνδεση στον πίνακα ελέγχου με ένα μόνο καλώδιο RJ11.
- **Συμβατότητα:** Περιέχει 3 ακίδες με βήμα 2,54 mm, επιτρέποντας πλήρη συμβατότητα με τυπικές συνδέσεις σερβομηχανισμού.

Έλεγχος Σερβομηχανισμού

- **Περιστροφή:** Η γωνία περιστροφής του σερβομηχανισμού ελέγχεται μέσω του κύκλου λειτουργίας ενός σήματος PWM (Pulse-Width Modulation).
- **Τυπικός Κύκλος Σήματος PWM:** 20 ms (50 Hz).
- **Πλάτος Παλμού:** Κυμαίνεται από 0,5 ms έως 2,5 ms, το οποίο αντιστοιχεί σε γωνία περιστροφής από 0° έως 180°.

Προδιαγραφές

- **Τάση Λειτουργίας:** DC 4,8V έως 6V
- **Εύρος Γωνίας Περιστροφής:** Περίπου 180° (500 έως 2500 μsec)
- **Εύρος Πλάτους Παλμού:** 500 έως 2500 μsec
- **Ταχύτητα Χωρίς Φορτίο:**
 - $0,12 \pm 0,01 \text{ sec}/60^\circ$ (DC 4,8V)
 - $0,1 \pm 0,01 \text{ sec}/60^\circ$ (DC 6V)
- **Ρεύμα Χωρίς Φορτίο:**
 - $200 \pm 20 \text{ mA}$ (DC 4,8V)
 - $220 \pm 20 \text{ mA}$ (DC 6V)
- **Διακοπή Ροπής:**
 - $1,3 \pm 0,01 \text{ kg/cm}$ (DC 4,8V)
 - $1,5 \pm 0,1 \text{ kg/cm}$ (DC 6V)
- **Διακοπή Ρεύματος:**
 - $\leq 850 \text{ mA}$ (DC 4,8V)
 - $\leq 1000 \text{ mA}$ (DC 6V)
- **Ρεύμα Αναμονής:**
 - $3 \pm 1 \text{ mA}$ (DC 4,8V)
 - $4 \pm 1 \text{ mA}$ (DC 6V)
- **Θερμοκρασία Λειτουργίας:** -10°C έως 50°C
- **Θερμοκρασία Αποθήκευσης:** -20°C έως 60°C
- **Μήκος Καλωδίου Κινητήρα:** $250 \pm 5 \text{ mm}$



Στις σελίδες που ακολουθούν στο κεφάλαιο αυτό, θα δούμε με παραδείγματα, πως μπορούμε να αξιοποιήσουμε το **ARD:icon** και τον εξοπλισμό που υπάρχει στα **S1 & S2** πακέτα με το Mind+ και το **πρόσθετο (extension) STEM Education** που προσθέτει εντολές και λειτουργίες στην φαρέτρα μας...

Εφόσον έχουμε ήδη «κατεβάσει» και εγκαταστήσει το Mind+ στον Η/Υ μας, ας «κατεβάσουμε» και ενσωματώσουμε σ' αυτό το πρόσθετο (extension).



Το χρήσιμο
πρόσθετο
στο Mind+
για ARD:icon
(arduino)



Ένα πρόσθετο που εφοδιάζει το περιβάλλον οπτικοποιημένου προγραμματισμού Mind+ με μια **βιβλιοθήκη πλακιδίων εντολών για τη συνεργασία με τον ADR:icon** και τις περισσότερες από τις περιφερειακές συσκευές (αισθητήρες, LED κ.λπ) που βρίσκονται στο S1 και S2.

Μπορούμε να «κατεβάσουμε» το plugin
από την ηλεκτρονική διεύθυνση:

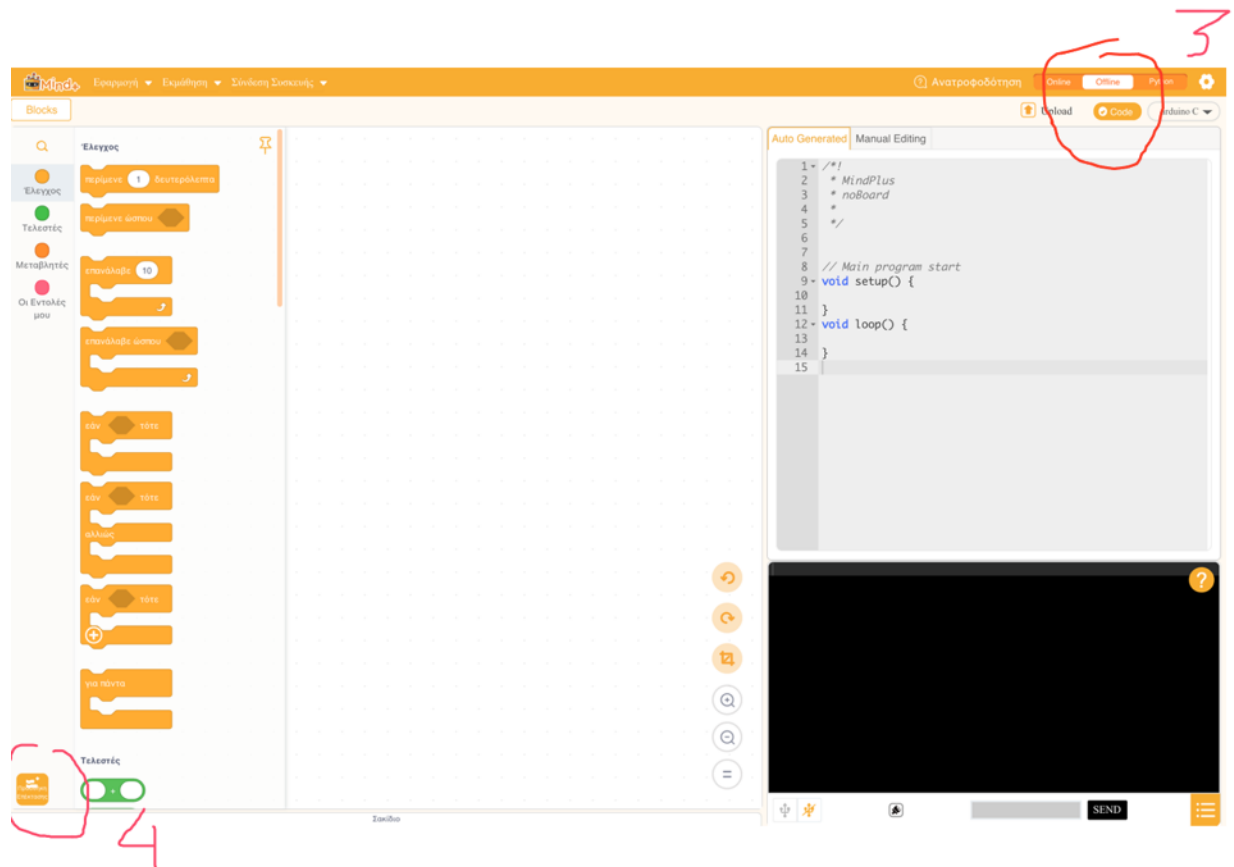
<https://portal.stem.edu.gr/courses/stem-ext-arduino/>

Αρχικά 5 απλές ενέργειες για την εγκατάσταση του πρόσθετου (extension), που εφοδιάζει το Mind+ με μια βιβλιοθήκη οπτικοποιημένων εντολών που οδηγούν τον εξοπλισμό SMART:Blox S1 (Δημοτικό) και SMART:Blox S2 (Γυμνάσιο):

1. Κατέβασμα και αποσυμπίεση του συνημμένου αρχείου .zip (που περιέχει το extension)
2. Άνοιγμα του Mind+
3. Ρύθμιση σε off line mode
4. Επιλογή μικροεπεξεργαστή: Arduino UNO
5. Εισαγωγή πρόσθετου (extension) αναζητώντας το στο σημείο που το «κατεβάσατε και αποθηκεύσατε.



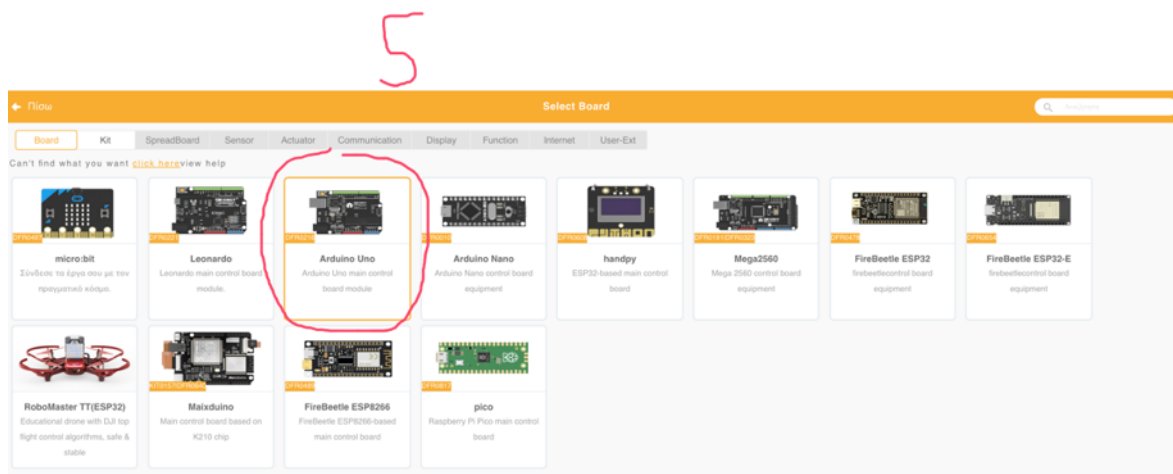
Οδηγίες εγκατάστασης



1. Κατεβάζουμε το συνημμένο πρόσθετο (Extension).
2. Ανοίγουμε το Mind+.
3. Επιλέγουμε off-line.
4. Ανοίγουμε το μενού επιλογής μικροεπεξεργαστή.

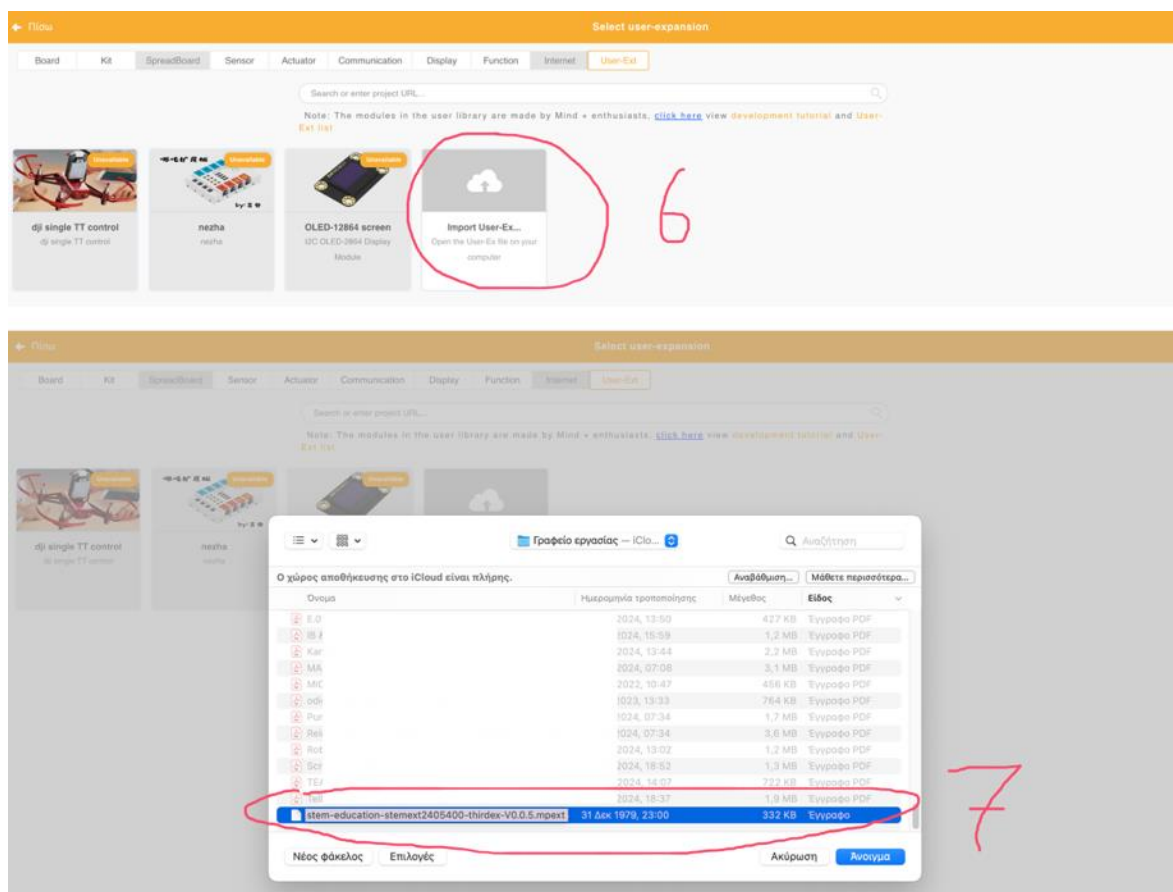


5. Επιλέγουμε Arduino UNO.



6. Επιλέγουμε "import".

7. Βρίσκουμε και επιλέγουμε το αρχείο με το extension.



Το set (βιβλιοθήκη) των οπτικοποιημένων εντολών (πλακίδια) είναι πλέον στη διάθεσή μας.



Λειτουργίες (πλακίδια εντολών)

The following table lists the commands and values from the programming blocks shown in the image:

Block Number	Command	Value(s)
1	R2 δεξιός > κινητήρας με κατεύθυνση > ρολογιού > και ταχύτητα	200
2	R2 όχημα μπροστά με > πολύ μικρή > ταχύτητα	
3	R2 όχημα μπροστά με ταχύτητα	50
4	R2 όχημα πίσω με > πολύ μικρή > ταχύτητα	
5	R2 όχημα πίσω με ταχύτητα	50
6	R2 όχημα δεξιά στροφή για	2 δευτερόλεπτα
7	R2 όχημα αριστερή στροφή για	2 δευτερόλεπτα
8	R2 όχημα σταμάτησε	
9	Κινητήρας Micro Servo στη θύρα > D5 > με γωνία > 90 > μοιρών	
10	Κινητήρας γκρι Servo στη θύρα > D5 > με γωνία > 90 > μοιρών	
11	Κινητήρας πράσινος στη θύρα > D5 > με κατεύθυνση > ρολογιού > και ταχύτητα	0
12	Κινητήρας βηματικός (έλικα) με φορά > ρολογιού > και ταχύτητα	120
13	Πρόβαλε στην οθόνη το > "hello" > (ανοίγοντας Serial > 9600 >)	
14	Άναψε > όλα τα LEDs > DJX13 (RGB), στη θύρα > D5 > , με χρώμα R: > 255 > , G: > 255 > , B: > 255 > και φωτεινότητα > 90	
15	Άναψε το > 1 > ο LED DJX13 (RGB), στη θύρα > D5 > , με χρώμα R: > 255 > , G: > 255 > , B: > 255 > και φωτεινότητα > 90	
16	Σβήσε όλα τα LEDs DJX13 (RGB), στη θύρα > D5 >	
17	Άναψε > το LED στη θύρα > D5 >	
18	Άναψε το LED στη θύρα > D5 > με ένταση > 255 > (0-255)	
19	Άναψε > τον ενεργό βομβητή στη θύρα > D5 >	
20	Παίξε στον παθητικό βομβητή (ηχείο) στη θύρα > D5 > τη νότα > Ντο > με χρόνο > 1/8 >	
21	Άναψε > το ρελέ στη θύρα > D5 >	
22	Άναψε > το φως στην οθόνη LCD1602 και εμφάνισε > "hello" > στην 1η γραμμή και > "world" > στη 2η γραμμή	



Έχει γίνει προσπάθεια τα ονόματα των εντολών, να περιγράφουν και τη λειτουργία τους...

Οδηγούν συσκευές από τα πακέτα S1, S2, R2, κινητήρες πράσινους γκρι servo κ.λπ.

Διάβασμα τιμών από αισθητήρες

Επιστρέφουν τιμές που τις αξιοποιούμε σε λειτουργίες που τις χρειάζονται.

	Ψηφιακή τιμή	Αισθητήρα εντοπισμού γραμμής ▼	στη θύρα	D5 ▼
	Τιμή κίτρινου κουμπού (πιεζόμενου διακόπτη)		στη θύρα	D5 ▼
	Τιμή θερμοκρασίας από αισθητήρα DHT11		στη θύρα	D6 ▼
	Τιμή υγρασίας από αισθητήρα DHT11		στη θύρα	D6 ▼
	Τιμή αισθητήρα απόστασης (υπερήχων SR01) σε εκατοστά			
	Τιμή αισθητήρα εντοπισμού γραμμής		στη θύρα	D5 ▼
	Τιμή αισθητήρα διακοπής φωτός		στη θύρα	D5 ▼
	Αναλογική τιμή	Αισθητήρα πίεσης ▼	στη θύρα	A0 ▼
	Τιμή ποτενσιόμετρου (αισθητήρας περιστροφής)		στη θύρα	A3 ▼
	Τιμή αισθητήρα επίπεδου φωτός		στη θύρα	A0 ▼
	Τιμή αισθητήρα θερμοκρασίας LM35 (αναλογικού)		στη θύρα	A0 ▼
	Τιμή αισθητήρα αλκοόλης MQ3		στη θύρα	A6 ▼
	Τιμή αισθητήρα στάθμης υγρού		στη θύρα	A3 ▼
	Τιμή αισθητήρα πίεσης μεμβράνης		στη θύρα	A0 ▼
	Τιμή αισθητήρα ατμού		στη θύρα	A2 ▼



Καταστάσεις αισθητήρων (true, false)

	Πατήθηκε το κίτρινο κουμπί (πεζόμενος διακόπτης) στη θύρα	D5 ▼ ;
	Ενεργοποιήθηκε ο αισθητήρας πρόσκρουσης στη θύρα	D8 ▼ ;
	Εντόπισε μαγνητικό πεδίο ο αισθητήρας στη θύρα	D5 ▼ ;
	Ανιχνεύτηκε κίνηση από τον αισθητήρα κίνησης στη θύρα	D5 ▼ ;

Σημείωση/Προσοχή:

Υπάρχει περίπτωση συναντήσετε μια μικρή αναντιστοιχία στα ονόματα των εντολών και λειτουργιών που υπάρχουν στα στιγμιότυπα του βιβλίου αυτού, σε σχέση με την πιο πρόσφατη έκδοση του STEM πρόσθετου (extension).

Αυτό συμβαίνει με τα ονόματα μόνο και όχι με τη λειτουργικότητα, καθώς προσαρμόζονται ώστε να είναι ποιο «ευανάγνωστες» από το χρήστη η λειτουργίες που εκτελούν.







ARD:icon Παραδείγματα - Εφαρμογές χωρίς δομικά υλικά



1ο

παράδειγμα: Ανάβοντας LED.

Προαπαιτούμενα (υλικά και λογισμικό):

1. ARD:icon (ACD 15G).
2. USB Καλώδιο σύνδεσης με Η/Υ.
3. Mind+.
4. STEM Extension.

Βήμα 1ο



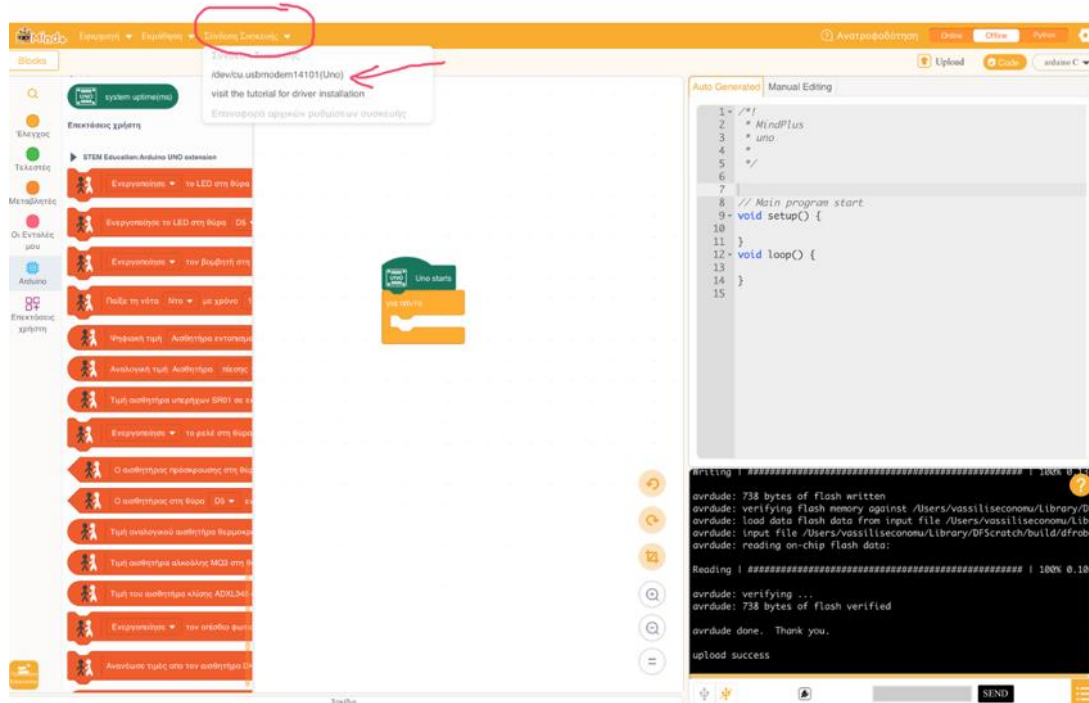
Ετοιμάζουμε τον ενεργό και παθητικό εξοπλισμό που θα χρησιμοποιήσουμε:

1. ARD:icon (ACD 15G).
2. USB Καλώδιο σύνδεσης με Η/Υ.
3. LED.
4. Καλώδιο σύνδεσης LED και ARD:icon.
5. Η/Υ.
6. Mind+
7. STEM Extension για S1&S2.



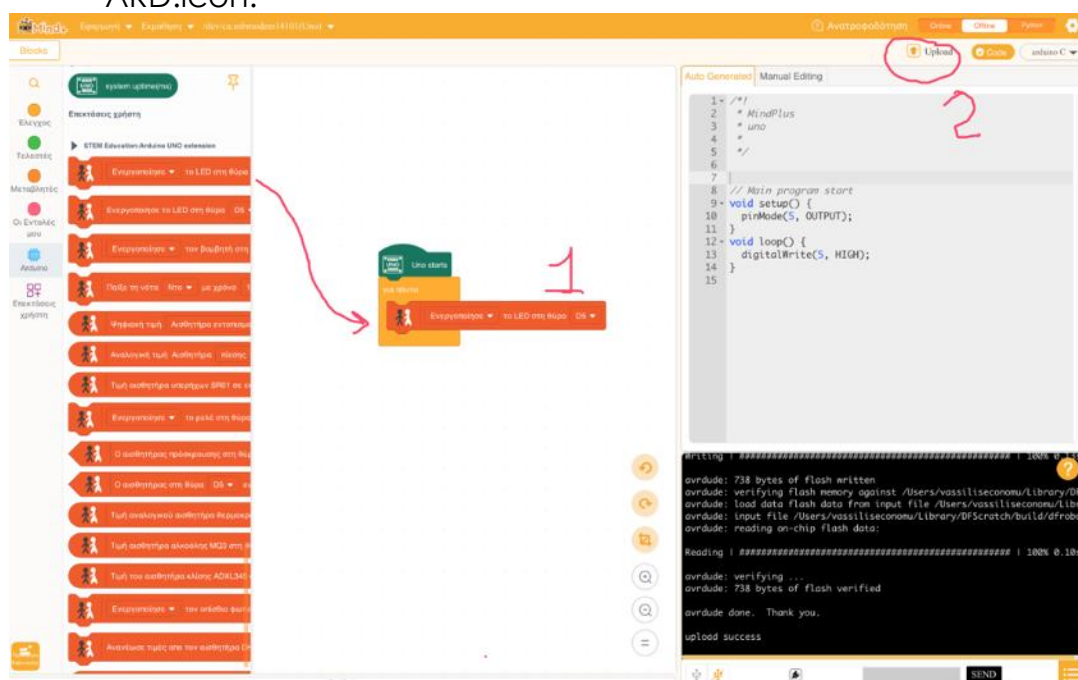
Βήμα 2ο

1. Ανοίγουμε το Mind+.
2. Έχουμε ανεβάσει ήδη την επέκταση.
3. Συνδέουμε τη συσκευή με το Mind+.



Βήμα 3ο

1. Αναπτύσσουμε το πρώτο μας πρόγραμμα χρησιμοποιώντας το κατάλληλο πλακίδιο εντολής.
2. Επιλέγουμε Upload, για να κατεβάσουμε την εφαρμογή μας στο ARD:icon.



2^ο

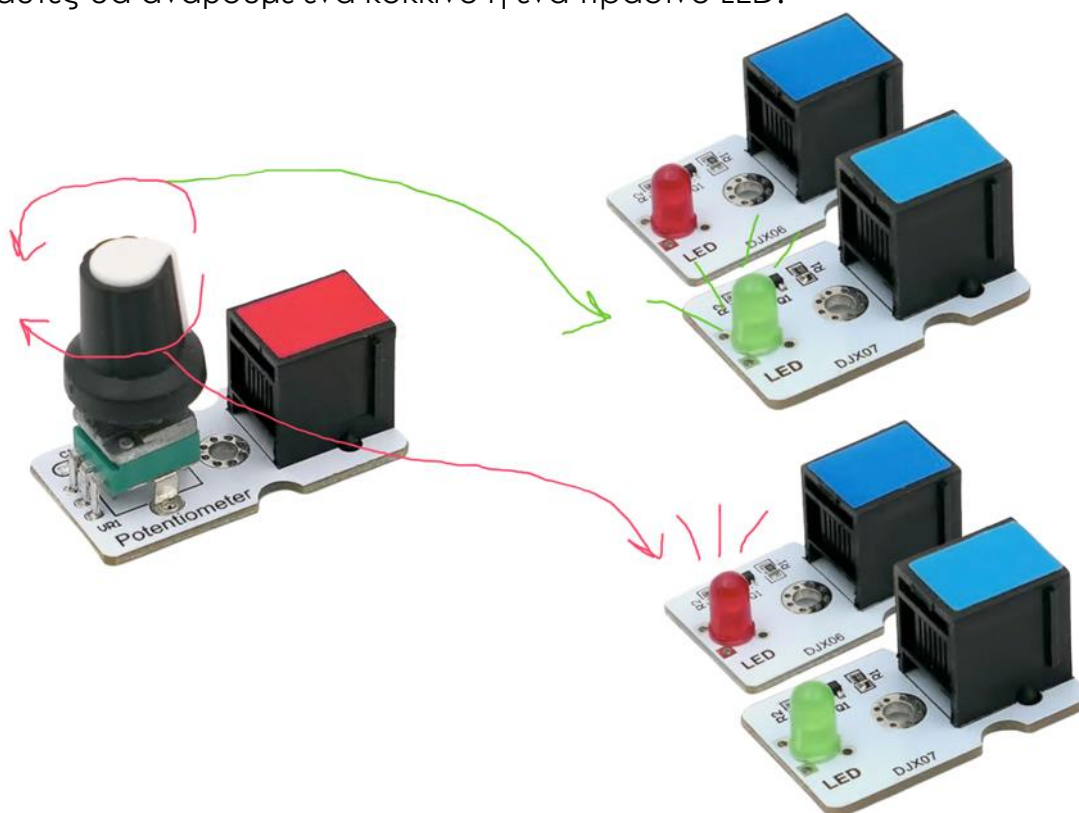
παράδειγμα: Άναμμα LEDs υπό συνθήκη.

Προαπαιτούμενα στοιχεία:

1. ARD:icon (ACD 15G).
2. USB Καλώδιο σύνδεσης με Η/Υ.
3. Mind+.
4. STEM Extension.
5. Κόκκινο LED.
6. Πράσινο LED.
7. Ποτενσιόμετρο.
8. καλώδια σύνδεσης.

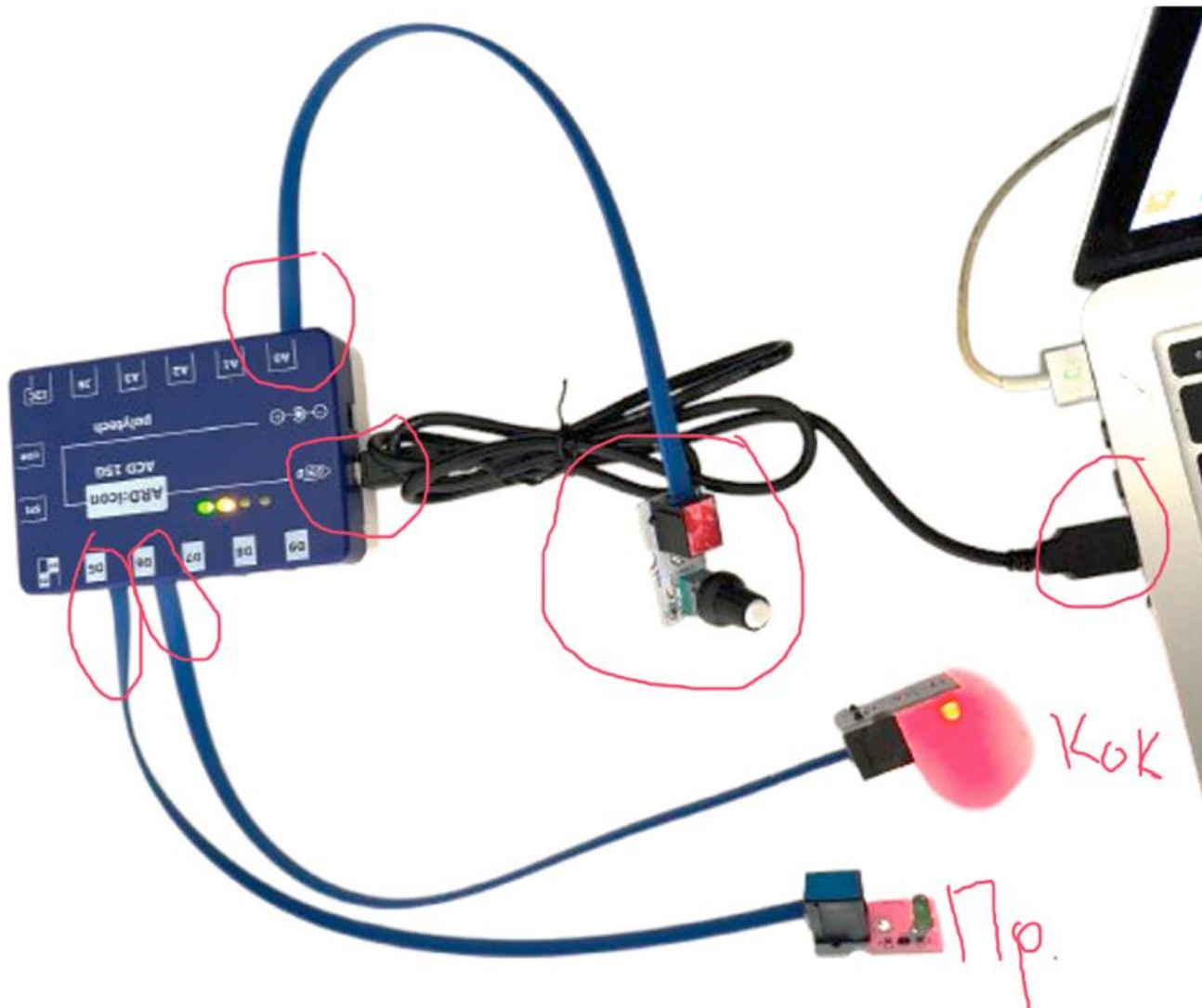
Σενάριο:

Ελέγχουμε τις τιμές που επιστρέφει ένα ποτενσιόμετρο. Ανάλογα με αυτές θα ανάβουμε ένα κόκκινο ή ένα πράσινο LED.



Βήμα 1°

Συνδέουμε τα υλικά μας, όπως στην εικόνα που ακολουθεί:



Βήμα 2ο

Γράφουμε τον κώδικα στο γραφικό περιβάλλον με πλακίδια εντολών αξιοποιώντας:

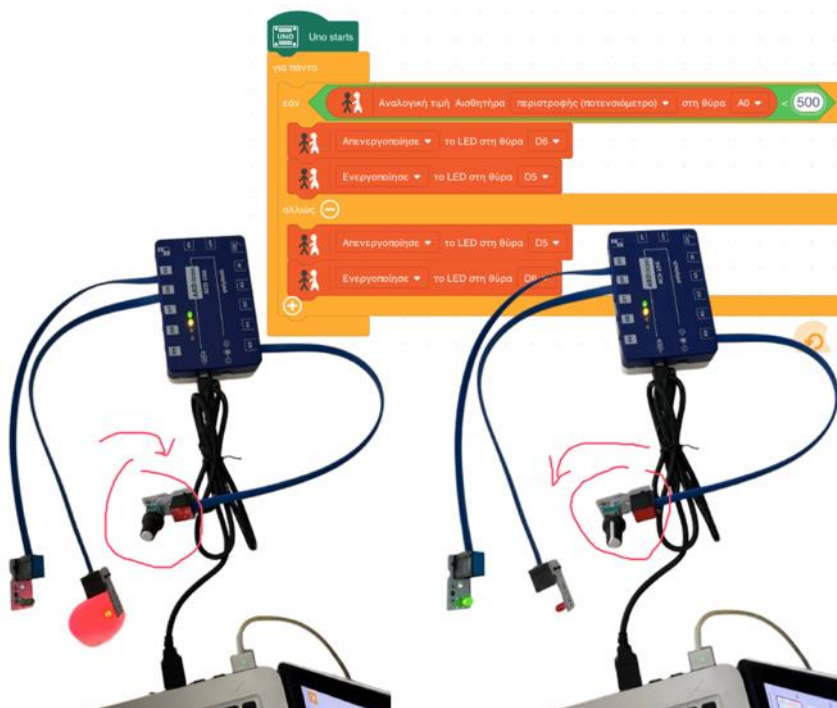
1. Την δομή Εάν – Αλλιώς
2. Την λειτουργία που επιστρέφει την τιμή του ποτενσιόμετρου που είναι συνδεδεμένο στη θύρα A0.
3. Ένα τελεστή σύγκρισης
4. Την εντολή που ανάβει ή σβήνει LEDs.



Στο οποίο περιγράφονται οι δύο καταστάσεις ανάλογα με τις τιμές του ποτενσιόμετρου:

1. Μικρότερες από 500
Σβήνει το κόκκινο και
Ανάβει το πράσινο.
2. Μεγαλύτερες από 500
Σβήνει το πράσινο και
Ανάβει το κόκκινο.

...Για πάντα.



Αντίστοιχα παράγεται και ο παρακάτω κώδικας (σε κείμενο) arduino C

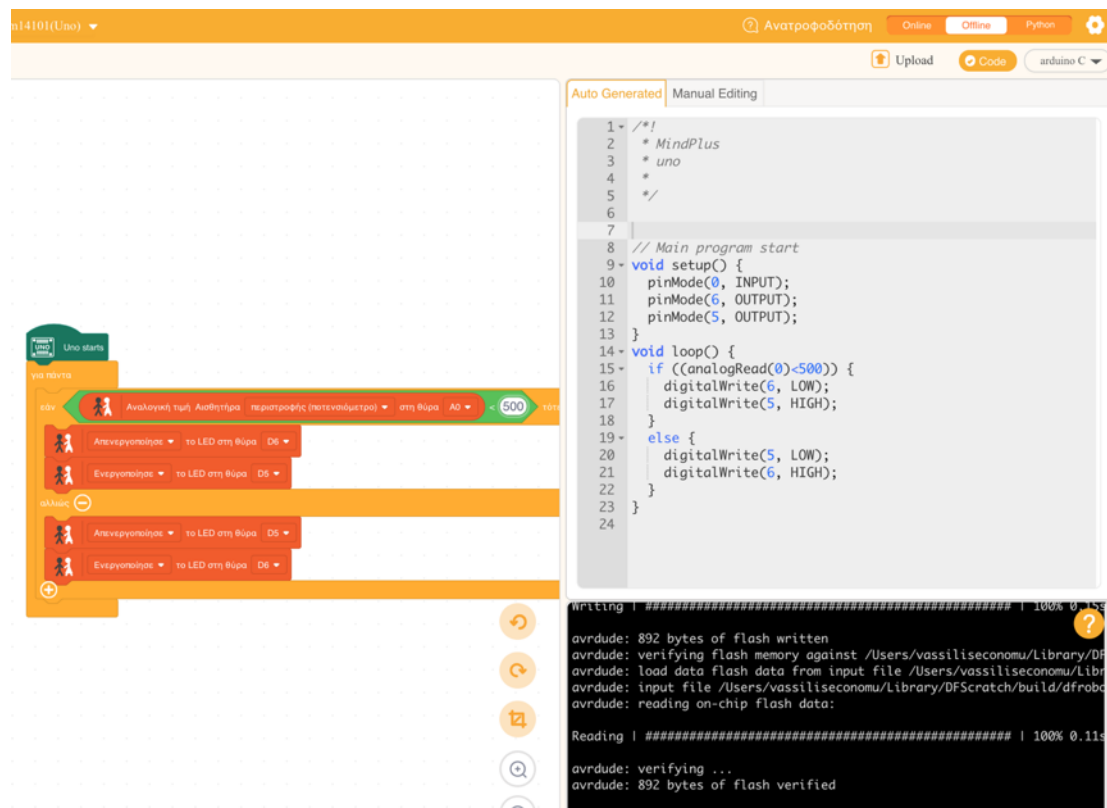
```

/*!
 * MindPlus
 * uno
 *
 */

// Main program start
void setup() {
    pinMode(0, INPUT);
    pinMode(6, OUTPUT);
    pinMode(5, OUTPUT);
}

void loop() {
    if ((analogRead(0)<500)) {
        digitalWrite(6, LOW);
        digitalWrite(5, HIGH);
    }
    else {
        digitalWrite(5, LOW);
        digitalWrite(6, HIGH);
    }
}

```



3^ο παράδειγμα: Αυτοματισμός παρκαρίσματος οχημάτων

Προαπαιτούμενα στοιχεία:

1. ARD:icon (ACD 15G).
2. USB Καλώδιο σύνδεσης με Η/Υ.
3. Mind+.
4. STEM Extension.
5. Κόκκινο LED.
6. Πράσινο LED.
7. LCD 1602 οθόνη.
8. Αισθητήρας υπερήχων.
9. Καλώδια σύνδεσης.
10. Lego βάση & στηρίγματα.

Σενάριο:

Δημιουργούμε ένα αυτοματισμό με βάση τον οποίο, ανάλογα με την απόσταση που ανιχνεύεται ένα εμπόδιο από τον αισθητήρα υπερήχων, θα ανάβει:

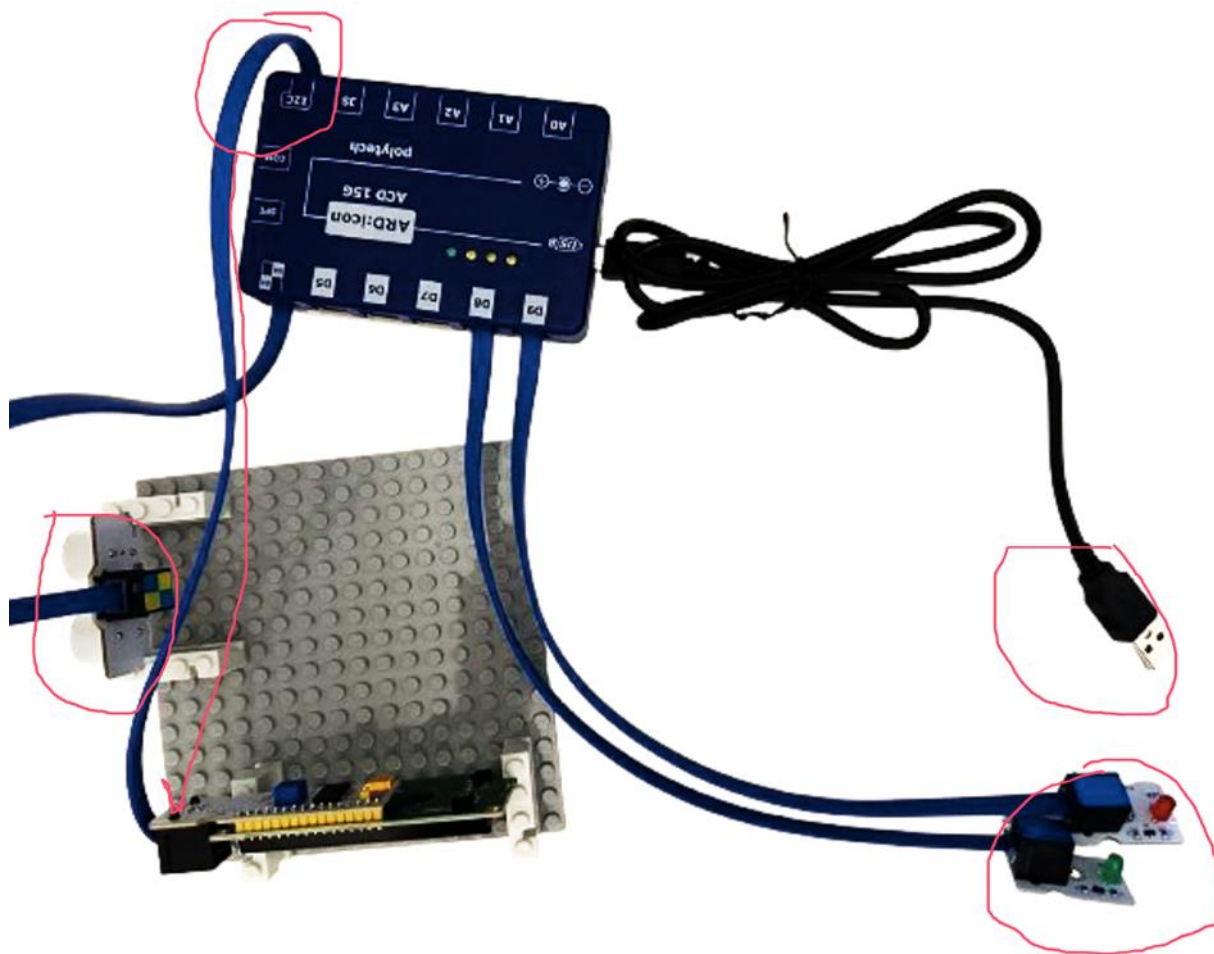
- πράσινο LED (>10 cm).
- ή κόκκινο LED (<10cm).



Συνδεσμολογία:

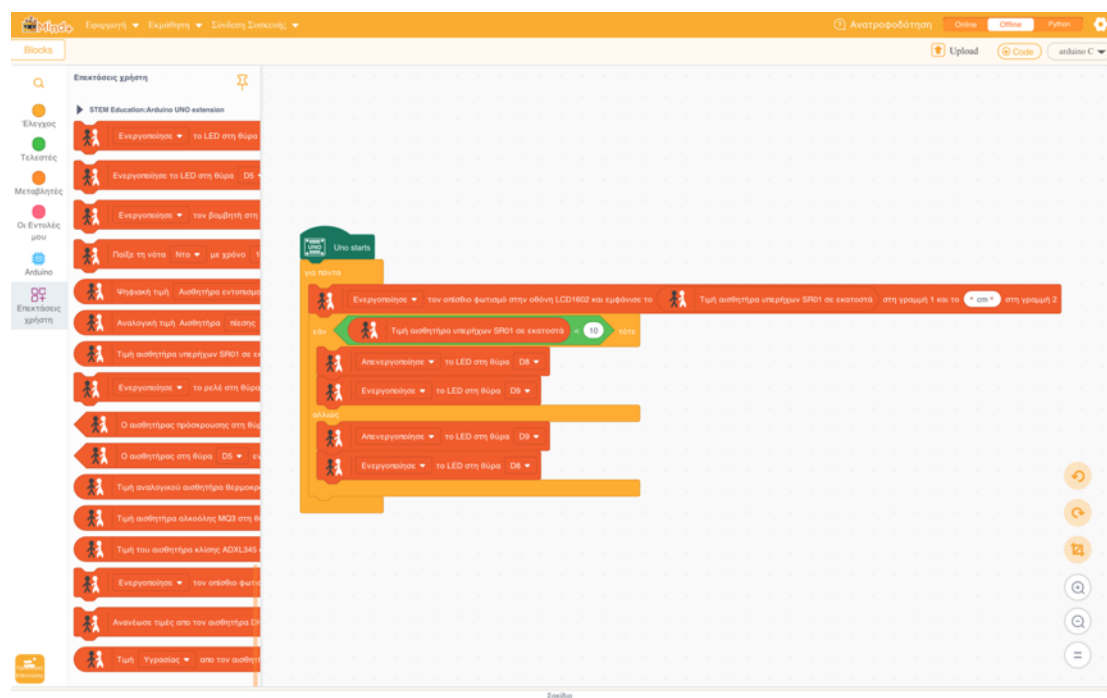
Συνδέουμε τα στοιχεία όπως στην εικόνα:

- στη θύρα I2C την LCD 1602 οθόνη.
- στη θύρα D3/D4 τον αισθητήρα υπερήχων.
- στη θύρα D8 το πράσινο LED.
- στη θύρα D9 το κόκκινο LED.

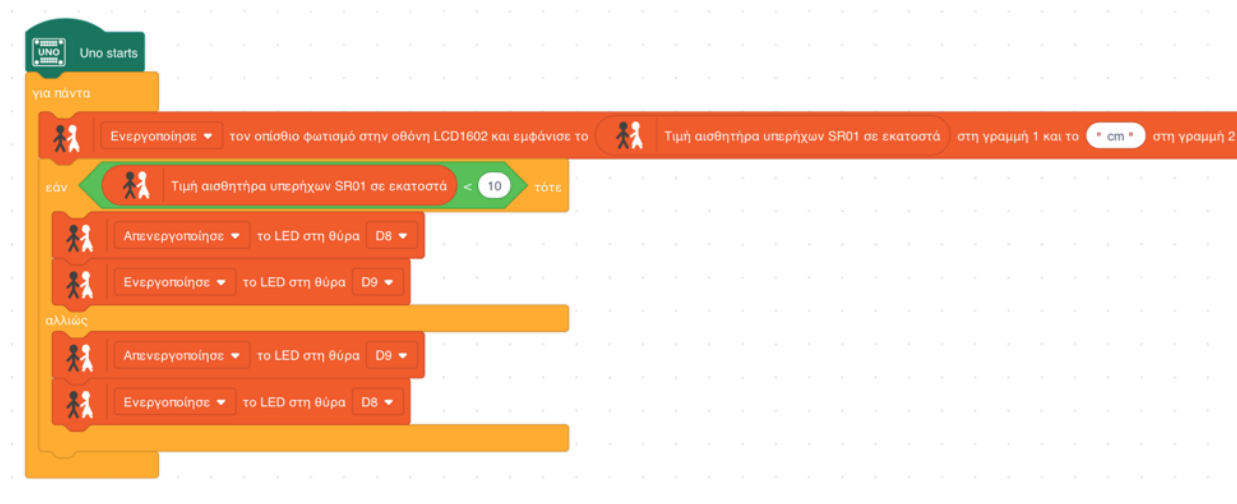


Προγραμματισμός:

Εδώ μπορούμε να δούμε τον οπτικοποιημένη προγραμματισμό:



Ο κώδικας:



...κι εδώ έχουμε τον κώδικα σε Arduino C:

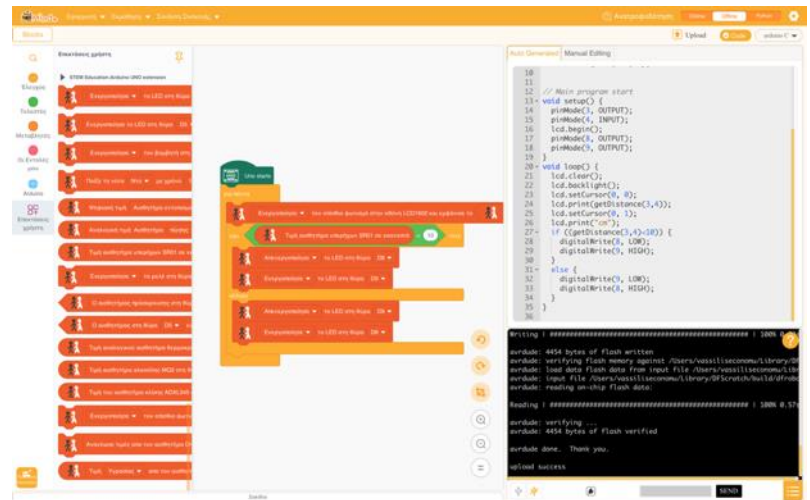
```

/*!
 * MindPlus uno
 *
 */
#include <SR01.h>
#include <LCD_I2C.h>
// Create an object
LCD_I2C lcd(0x27, 16, 2);

// Main program start
void setup() {
    pinMode(3, OUTPUT);
    pinMode(4, INPUT);
    lcd.begin();
    pinMode(8, OUTPUT);
    pinMode(9, OUTPUT);
}

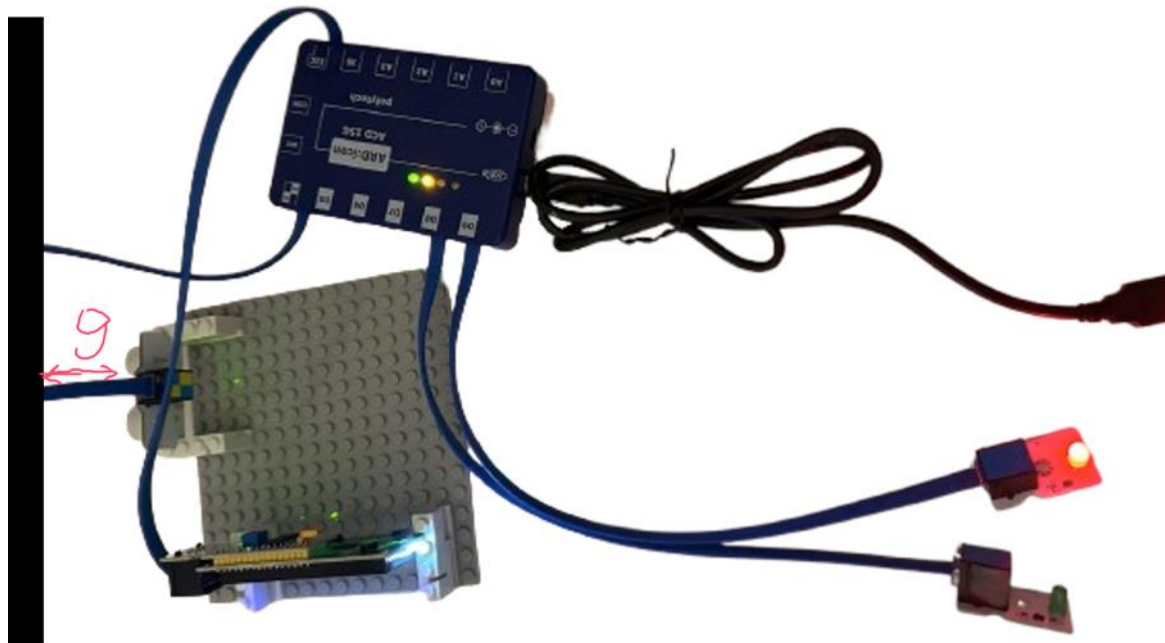
void loop() {
    lcd.clear();
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print(getDistance(3,4));
    lcd.setCursor(0, 1);
    lcd.print("cm");
    if ((getDistance(3,4)<10)) {
        digitalWrite(8, LOW);
        digitalWrite(9, HIGH);
    }
    else {
        digitalWrite(9, LOW);
        digitalWrite(8, HIGH);
    }
}

```

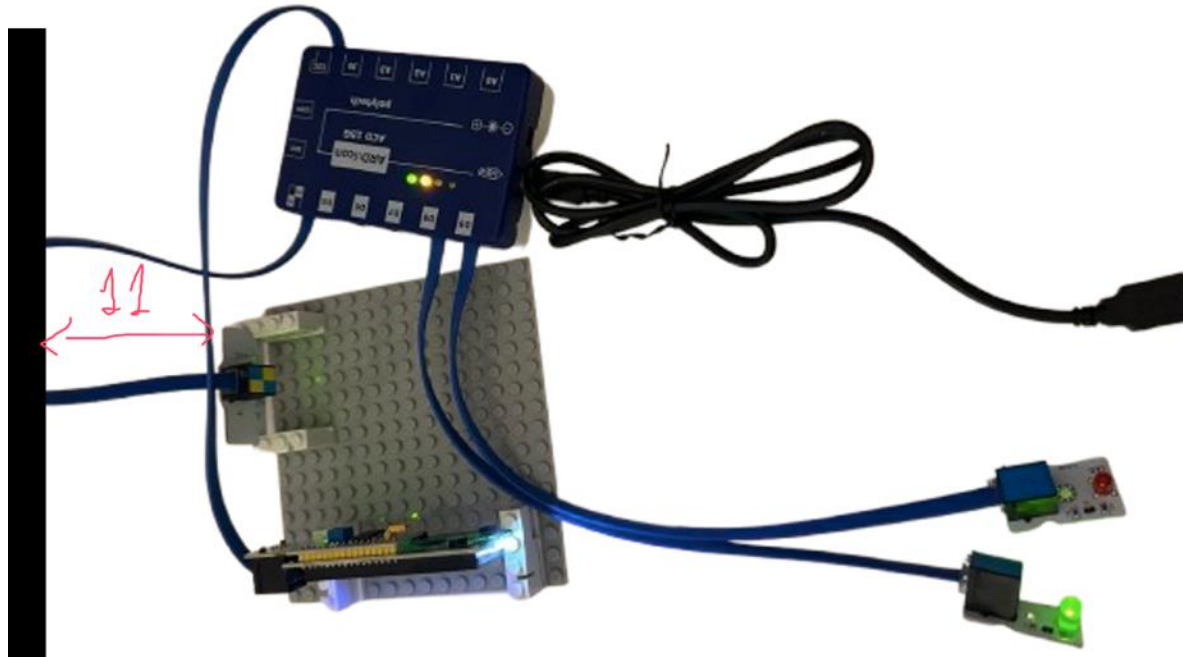


Η λειτουργία:

Α. Εμπόδιο σε μικρότερη απόσταση από 10cm, ανάβει το κόκκινο LED και η ένδειξη εμφανίζεται στην LCD οθόνη.



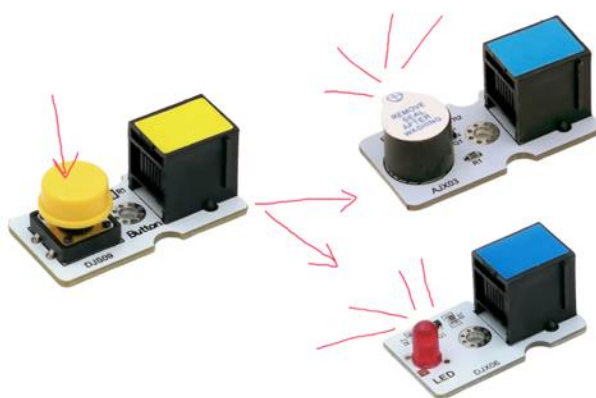
Β. Εμπόδιο σε μεγαλύτερη απόσταση από 10cm, ανάβει το πράσινο LED και η ένδειξη εμφανίζεται στην LCD οθόνη.



4^ο παράδειγμα: Αυτοματισμός κουμπιού που παράγει ήχο και φως

Σενάριο

Πατώντας ένα κουμπί συνδεδεμένο στο ARD:icon ανάβει ένα LED και ταυτόχρονα ακούγεται μια νότα στο επίσης συνδεδεμένο ηχείο (ενεργός βομβητής).



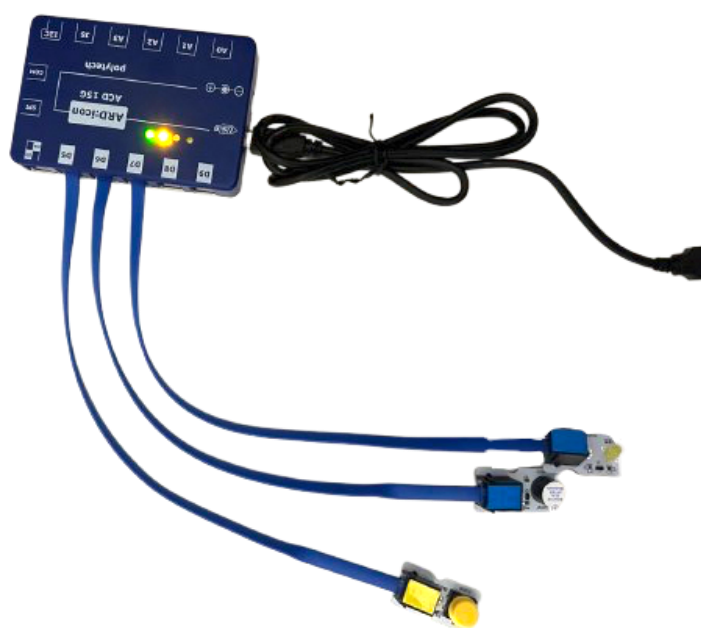
Συνδεσμολογία:

Ακολουθούμε τη συνδεσμολογία που περιγράφεται στη συνέχεια:

Θύρα: D5 κουμπί.

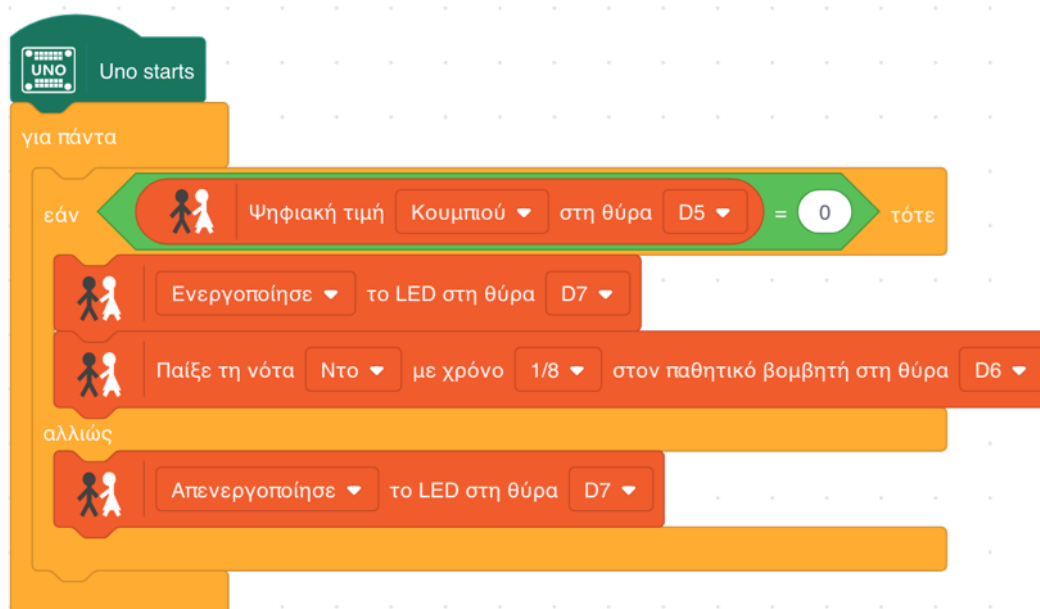
Θύρα: D6 ηχείο (ενεργός βομβητής).

Θύρα: D7 κίτρινο LED.



Κώδικας:

Εάν η τιμή που επιστρέφει το κουμπί είναι 0 (πατημένο),
Τότε άναψε το LED και παίξε τη νότα Ντο για 1/8 στο ηχείο,
Αλλιώς σβήσε το LED.



...και σε arduino C:

```

/*!
 * MindPlus
 * uno
 *
 */

// Main program start
void setup() {
  pinMode(5, INPUT);
  pinMode(7, OUTPUT);
  pinMode(6, OUTPUT);
}

void loop() {
  if ((digitalRead(5)==0)) {
    digitalWrite(7, HIGH);
    tone(6, 262, 125);
  }
  else {
    digitalWrite(7, LOW);
  }
}
  
```

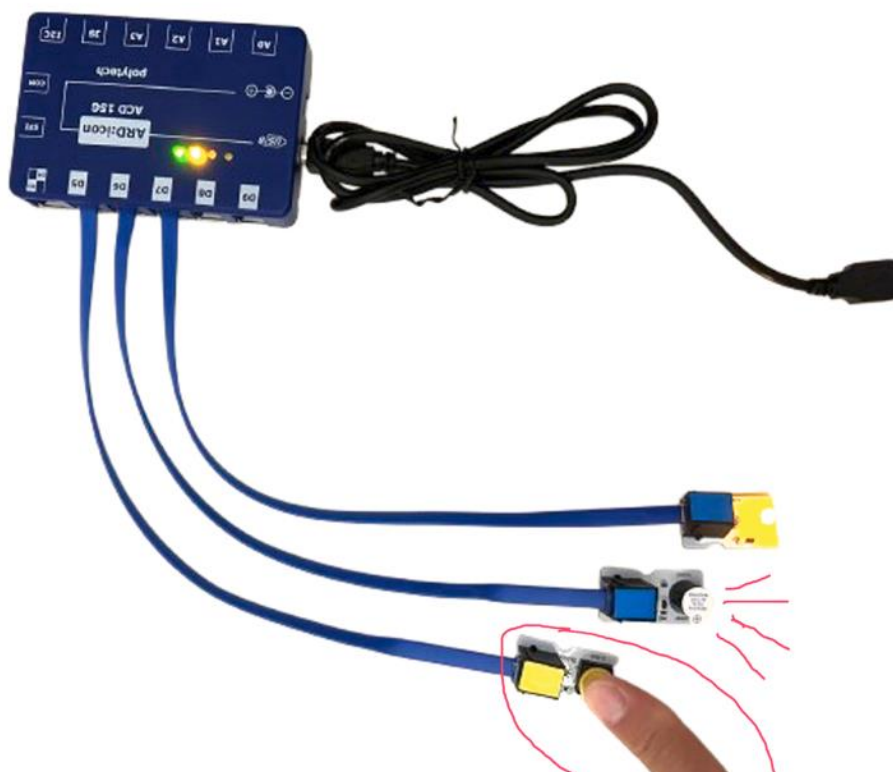


Αποτέλεσμα:

Το κουμπι **δεν είναι πατημένο**. LED σβηστό, χωρίς ήχο:



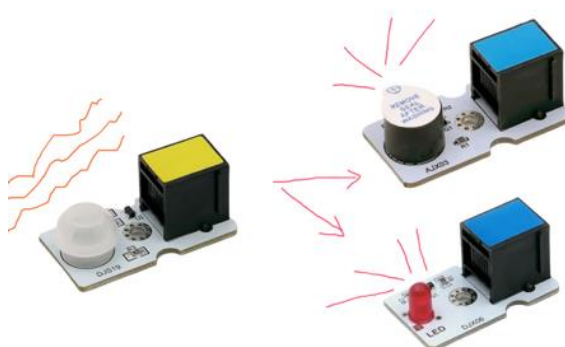
Το κουμπι **είναι πατημένο**. LED αναμμένο και ήχος από το ηχείο:



5^ο παράδειγμα: Αυτοματισμός ανίχνευσης κίνησης

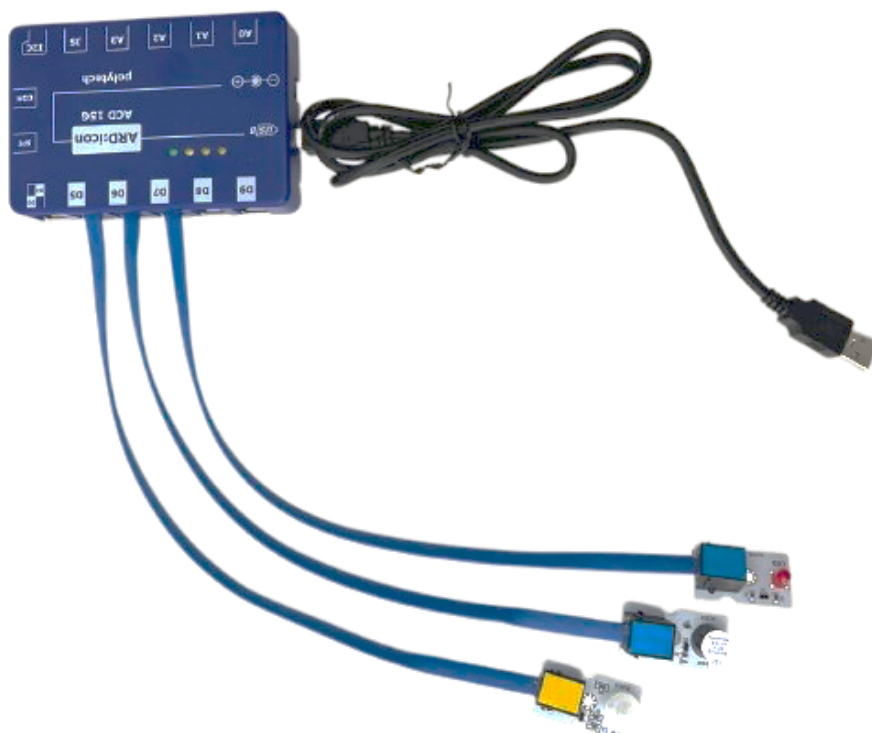
Σενάριο

Με την ανίχνευση κίνησης από τον Αισθητήρα κίνησης PIR (Pyroelectric Infrared) DJS19 συνδεδεμένο στο ARD:icon ανάβει ένα LED και ταυτόχρονα και ακούγεται μια νότα στο επίσης συνδεδεμένο ηχείο (ενεργός βομβητής).



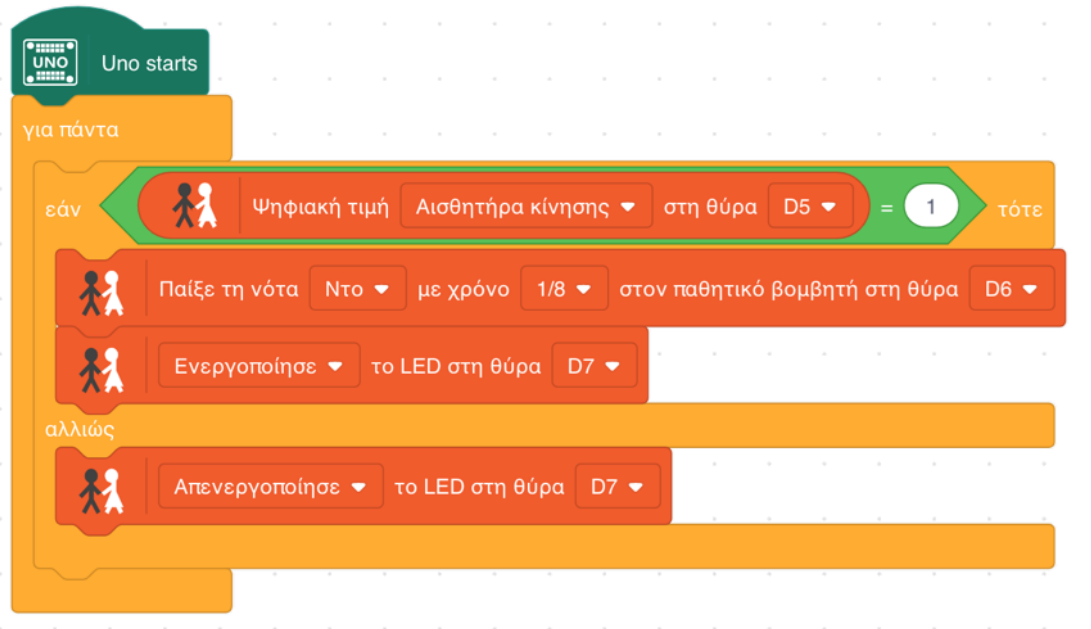
Συνδεσμολογία:

- Θύρα: D5 αισθητήρας κίνησης PIR (Pyroelectric Infrared) DJS19.
- Θύρα: D6 ηχείο (ενεργός βομβητής).
- Θύρα: D7 κόκκινο LED.



Κώδικας:

Εάν η τιμή που επιστρέφει αισθητήρας κίνησης είναι 1 (κάτι κινείται),
Τότε άναψε το LED και παίξε τη νότα Ντο για 1/8 στο ηχείο.
Αλλιώς σβήσε το LED.



...και σε arduino C:

```

/*!
 * MindPlus
 * uno
 *
 */

// Main program start
void setup() {
  pinMode(5, INPUT);
  pinMode(6, OUTPUT);
  pinMode(7, OUTPUT);
}
void loop() {
  if ((digitalRead(5)==1)) {
    tone(6, 262, 125);
    digitalWrite(7, HIGH);
  }
  else {
    digitalWrite(7, LOW);
  }
}

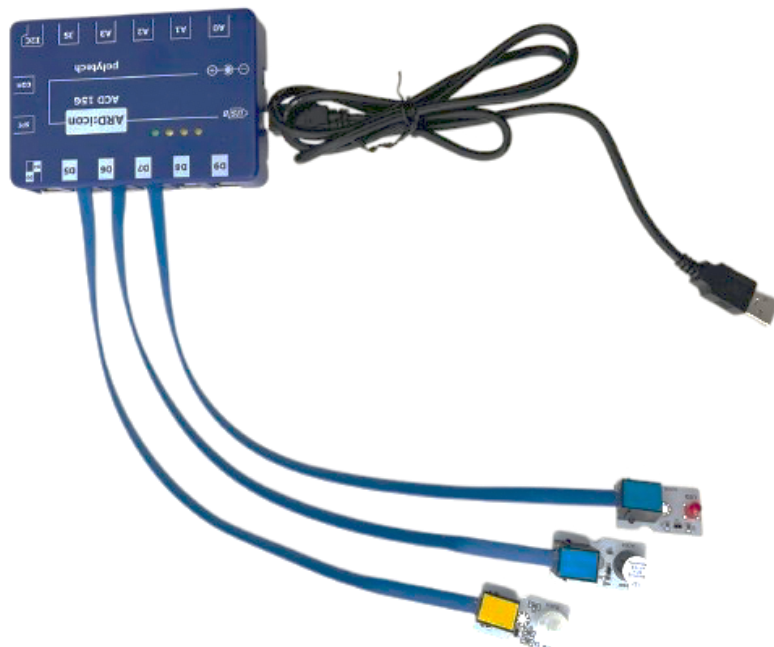
```



Αποτέλεσμα:

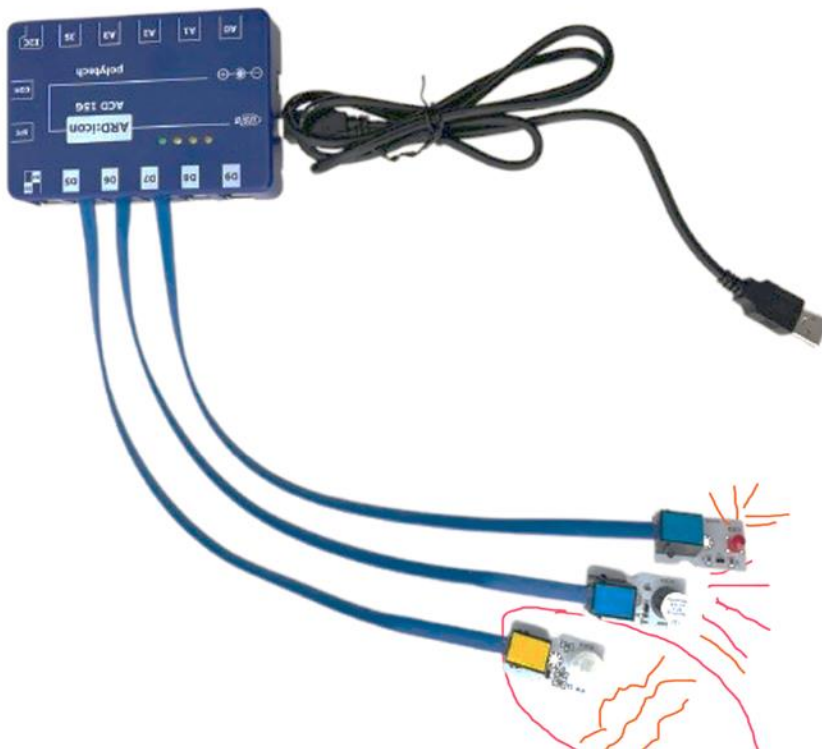
Δεν ανιχνεύεται κίνηση:

- LED σβηστό.
- Χωρίς ήχο.



Ανιχνεύεται κίνηση:

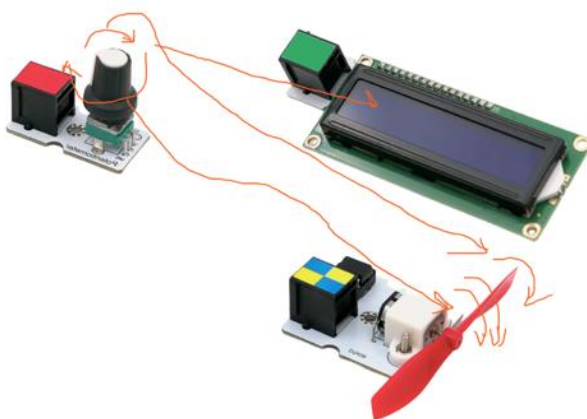
- LED αναμμένο.
- Ήχος από το ηχείο.



6^ο παράδειγμα: Ανεμιστήρας με ρυθμιζόμενη ένταση περιστροφής

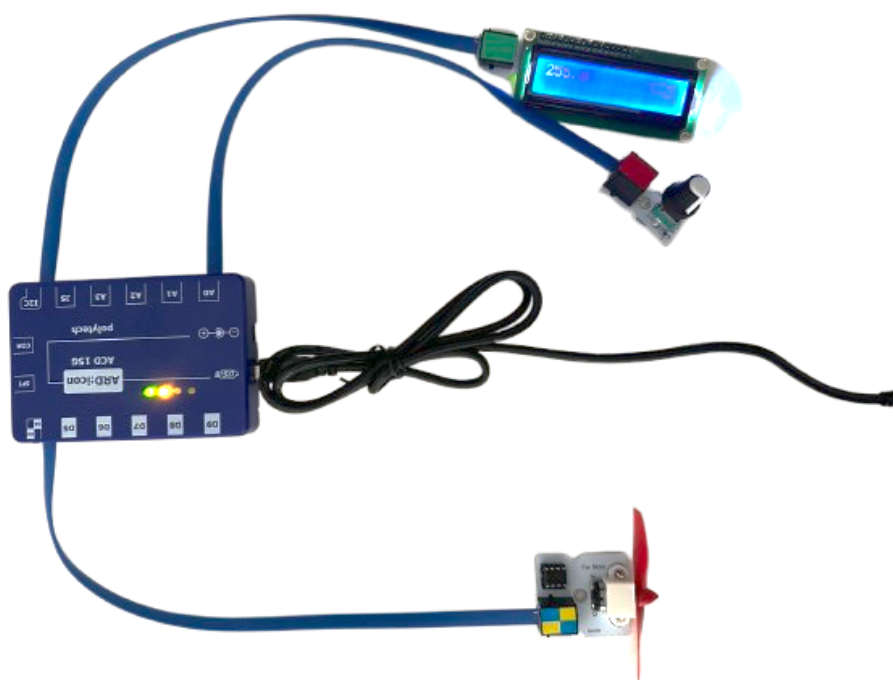
Σενάριο:

Η ένταση του ανεμιστήρα ρυθμίζεται με ένα ποτενσιόμετρο.
Η ένδειξη ταχύτητας περιστροφής εμφανίζεται στην οθόνη LCD.



Συνδεσμολογία:

- Στην θύρα A0 το ποτενσιόμετρο.
- Στην θύρα D3/D4 τον βηματικό κινητήρα με έλικα (ανεμιστήρας) DJX09.
- Στην θύρα I2C την οθόνη LCD 16x2 AJX04.



Κώδικας:



Εδώ έχουμε δημιουργία και χρήση αριθμητικής μεταβλητής. Σ' αυτήν αρχικά θα καταχωρήσουμε την τιμή του ποτενσιόμετρου και στη συνέχεια την αντιστοίχιση με το εύρος τιμών του κινητήρα.



Επιπλέον αξιοποιούμε μια λειτουργία αντιστοίχιση μεταξύ τιμών δύο πινάκων:

1. Πίνακας τιμών ποτενσιόμετρου 0-1023.
2. Πίνακας τιμών περιστροφής κινητήρα 0-255.

αντιστοίχιση		Τιμή από ποτενσιόμετρο	από[	0	,	1023	]	σε[	0	,	255	]
		ποτενσιόμετρο						κινητήρας				
		0							0			
		=							2			
		100										
		-										
		-										
		-										
		1023							255			



...και σε arduino C:

```

*
*/
#include <LCD_I2C.h>

// Dynamic variables
volatile float mind_n__;
// Create an object
LCD_I2C lcd(0x27, 16, 2);

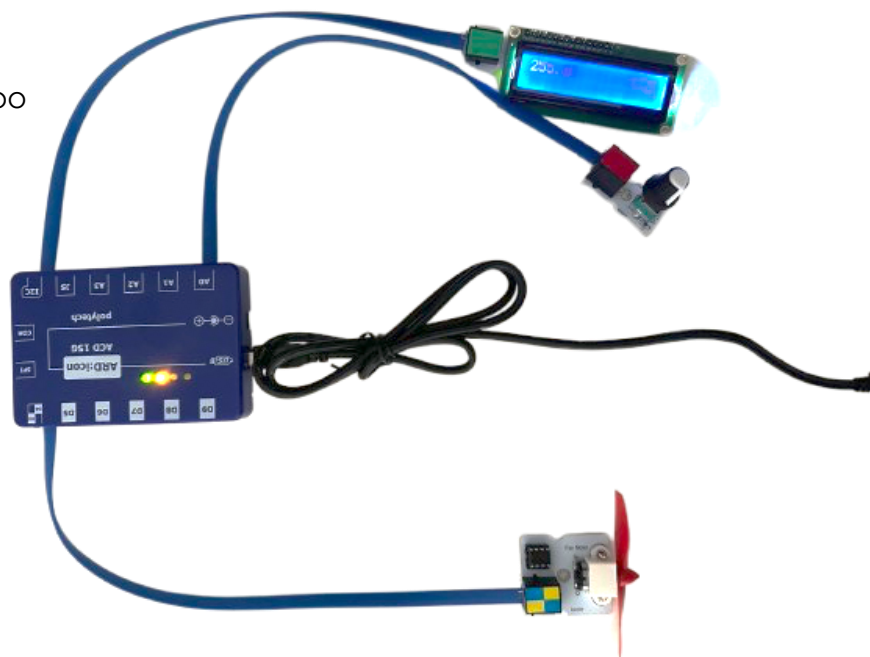
// Main program start
void setup() {
    pinMode(0, INPUT);
    lcd.begin();
}
void loop() {
    mind_n__ = analogRead(0);
    mind_n__ = (map(mind_n__, 0, 1023, 0, 255));
    lcd.clear();
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print(mind_n__);
    lcd.setCursor(0, 1);
    lcd.print("");
    analogWrite(3, mind_n__);
}

```

Αποτέλεσμα:

Όσο περιστρέφω το ποτενσιόμετρο τόσο γίνεται με μεγαλύτερη ένταση η περιστροφή του άξονα κινήτηρα και κατ' επέκταση του πτερυγίου του ανεμιστήρα.

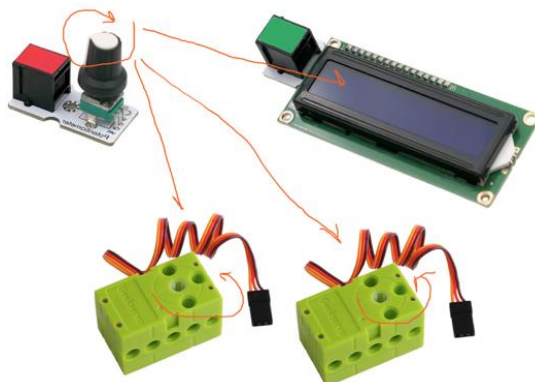
Η αντιστοιχισμένη τιμή του ποτενσιόμετρου εμφανίζεται στην LCD οθόνη.



7ο παράδειγμα: Περιστροφή κινητήρων με ρυθμιζόμενη ταχύτητα

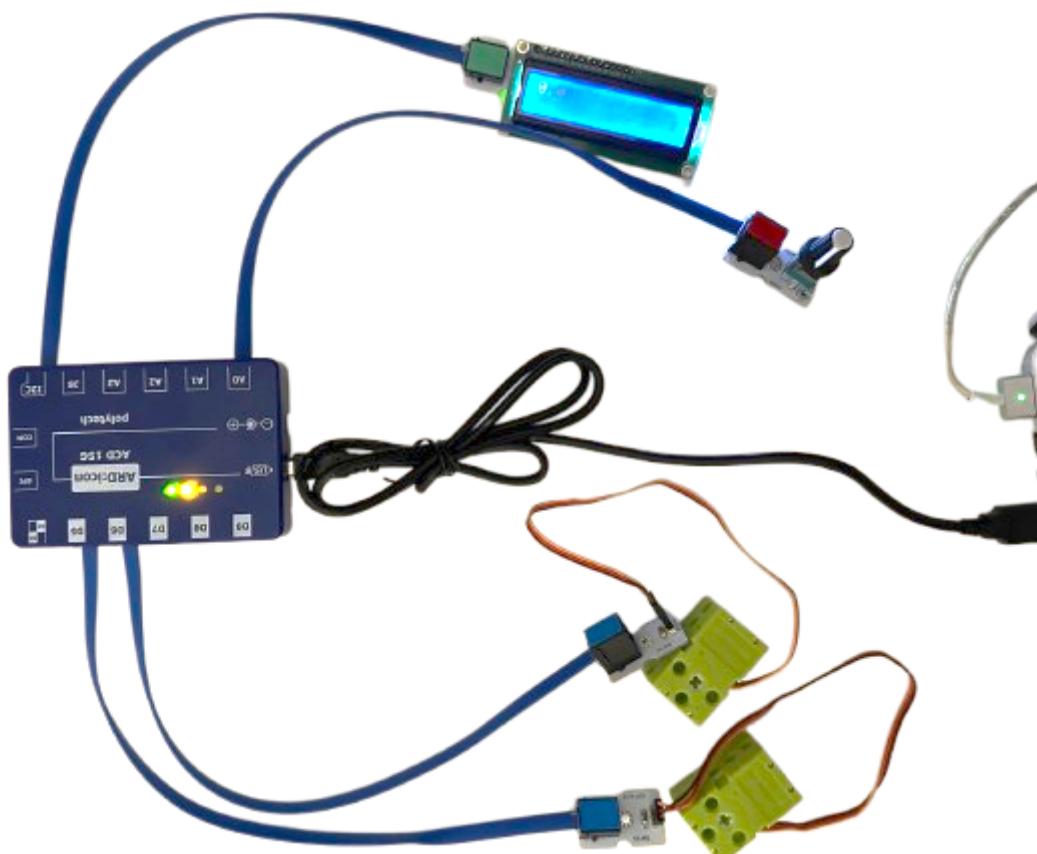
Σενάριο:

Η ένταση των δύο κινητήρων ρυθμίζεται με ένα ποτενσιόμετρο. Η ένδειξη ταχύτητας περιστροφής εμφανίζεται στην οθόνη LCD.



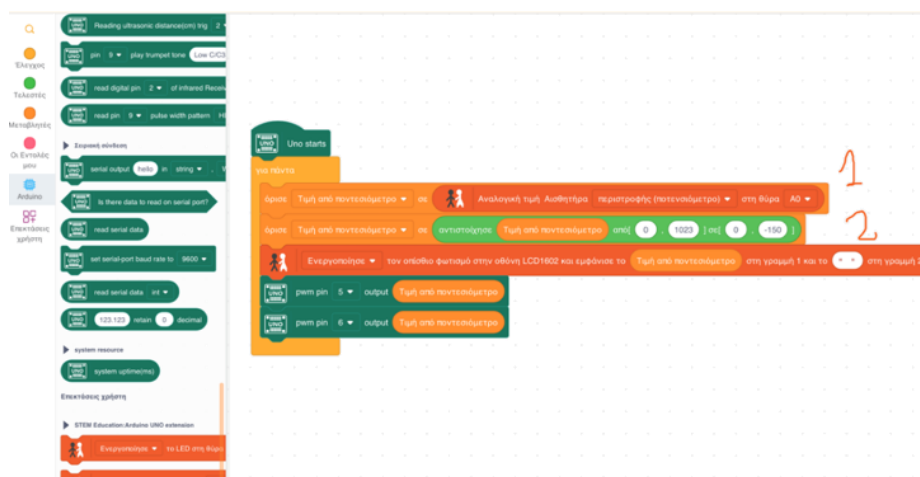
Συνδεσμολογία:

- Στην θύρα A0 το ποτενσιόμετρο.
- Στην θύρα D5 και D6 οι δύο κινητήρες συνεχούς περιστροφής.
- Στην θύρα I2C την οθόνη LCD 16x2 AJX04.
- Αξιοποιούμε και τις δύο διεπαφές servo EXP-AJ11 πάνω στις οποίες προσαρμόζουμε τους κινητήρες συνεχούς περιστροφής.



Κώδικας:

Εδώ έχουμε τη δημιουργία και χρήση αριθμητικής μεταβλητής. Σ' αυτήν αρχικά θα καταχωρήσουμε την τιμή του ποτενσιόμετρου και στη συνέχεια την αντιστοίχιση με το εύρος τιμών των κινητήρων συνεχούς περιστροφής.



Επιπλέον χρησιμοποιούμε μια σημαντική λειτουργία για αντιστοίχιση μεταξύ τιμών δύο πινάκων:

1. Πίνακας τιμών ποτενσιόμετρου 0-1023.
2. Πίνακας τιμών κινητήρα 0 μέχρι -150.

αντιστοίχισε	Τιμή από ποτενσιόμετρο	από [0 , 1023]	σε [0 , -150]
		ποτενσιόμετρο	κινητήρας
		0	0
		=	
		100	-2
		.	
		.	
		.	
		1023	-150



...και ο κώδικας σε arduino C:

```

/!*
 * MindPlus
 * uno
 *
 */
#include <LCD_I2C.h>

// Dynamic variables
volatile float mind_n__;
// Create an object
LCD_I2C lcd(0x27, 16, 2);

// Main program start
void setup() {
    pinMode(0, INPUT);
    lcd.begin();
}
void loop() {
    mind_n__ = analogRead(0);
    mind_n__ = (map(mind_n__, 0, 1023, 0, -150));
    lcd.clear();
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print(mind_n__);
    lcd.setCursor(0, 1);
    lcd.print("");
    analogWrite(5, mind_n__);
    analogWrite(6, mind_n__);
}

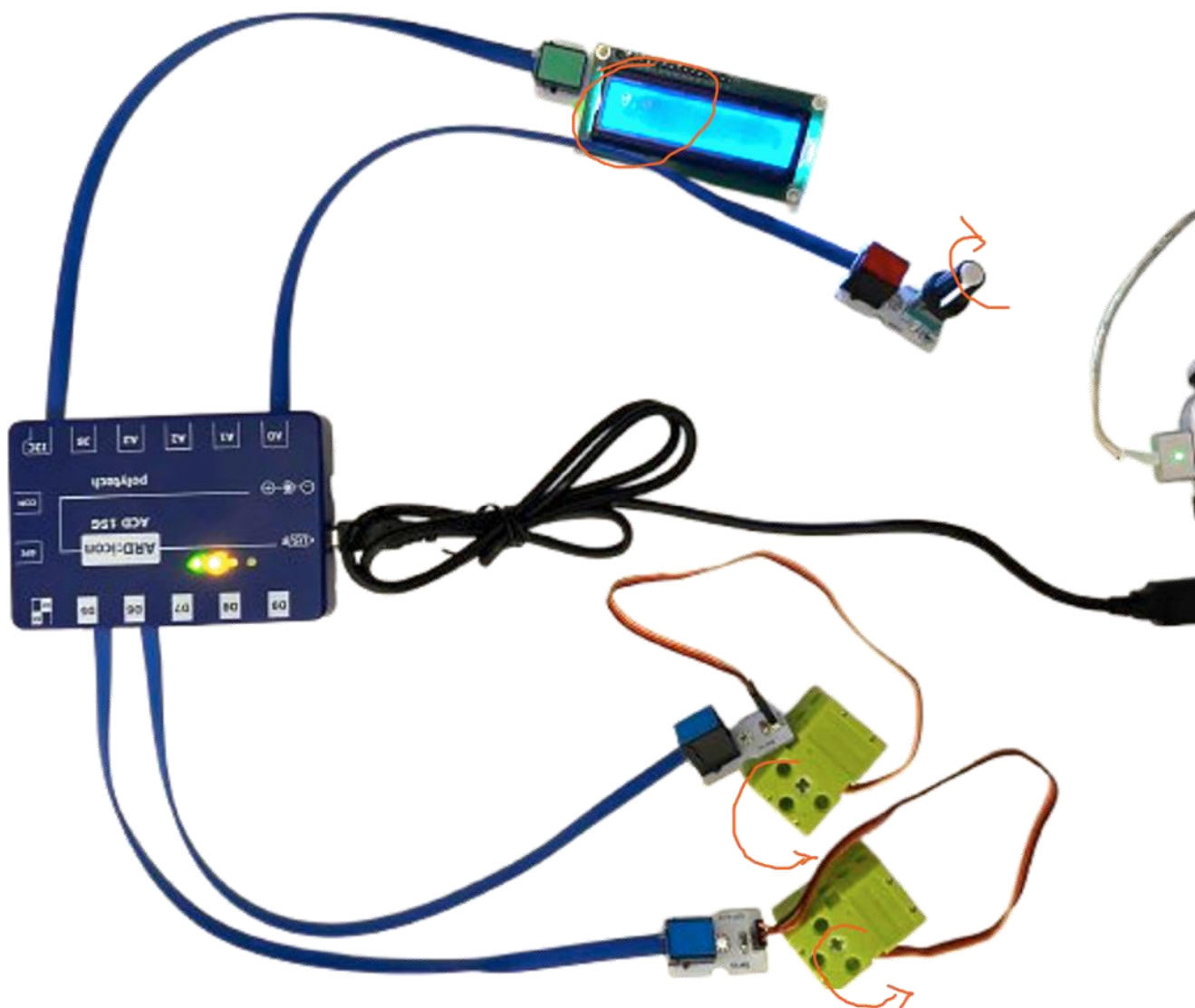
```



Αποτέλεσμα

Όσο περιστρέφω το ποτενσιόμετρο τόσο γίνεται με μεγαλύτερη ένταση η περιστροφή του άξονα των κινητήρων.

Η αντιστοιχισμένη τιμή του ποτενσιόμετρου εμφανίζεται στην LCD οθόνη.

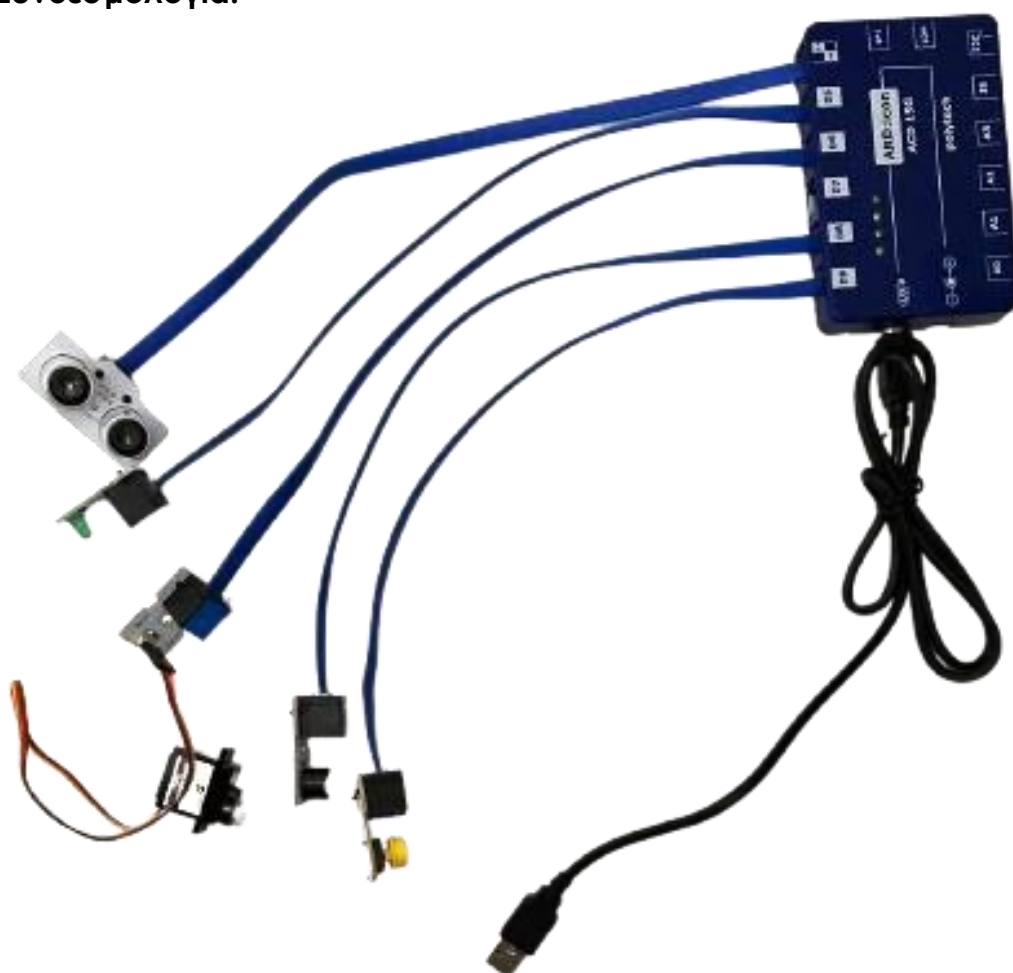


8ο παράδειγμα: Ο μηχανισμός της πόρτας του γκαράζ (servo)

Σενάριο:

Δημιουργούμε ένα αυτοματισμό με βάση τον οποίο ανοίγει η μπάρα του γκαράζ όταν πλησιάσει ένα όχημα, ανάβει το πράσινο LED. Για λόγους καλύτερου ελέγχου υπάρχει και ένα κουμπί (button) με το οποίο ανοίγει η μπάρα (από τη μέσα πλευρά)

Συνδεσμολογία:



Στις θύρες του ARD:icon:

- D3/D4:** Αισθητήρας απόστασης (Sonar)
- D5:** Πράσινο LED
- D6:** Mini Servo (μέσω του adaptor GVS)
- D8:** Buzzer (Ηχείο)
- D9:** Κίτρινος διακόπτης (button)

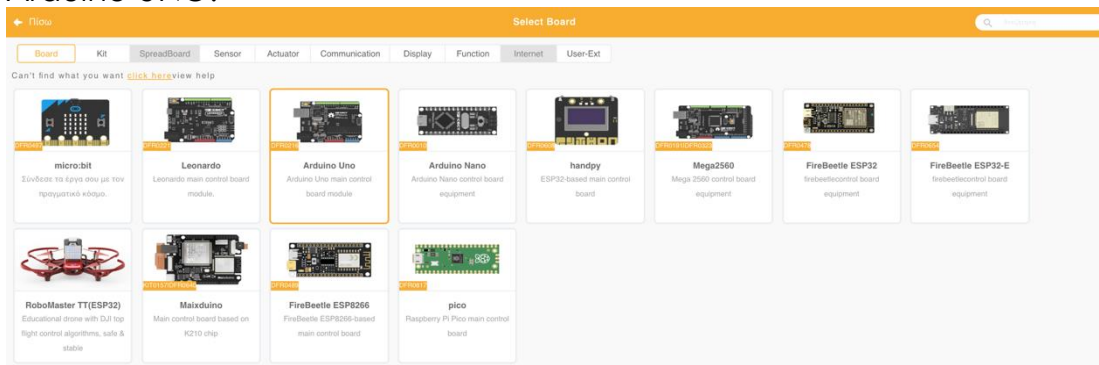


Βιβλιοθήκες

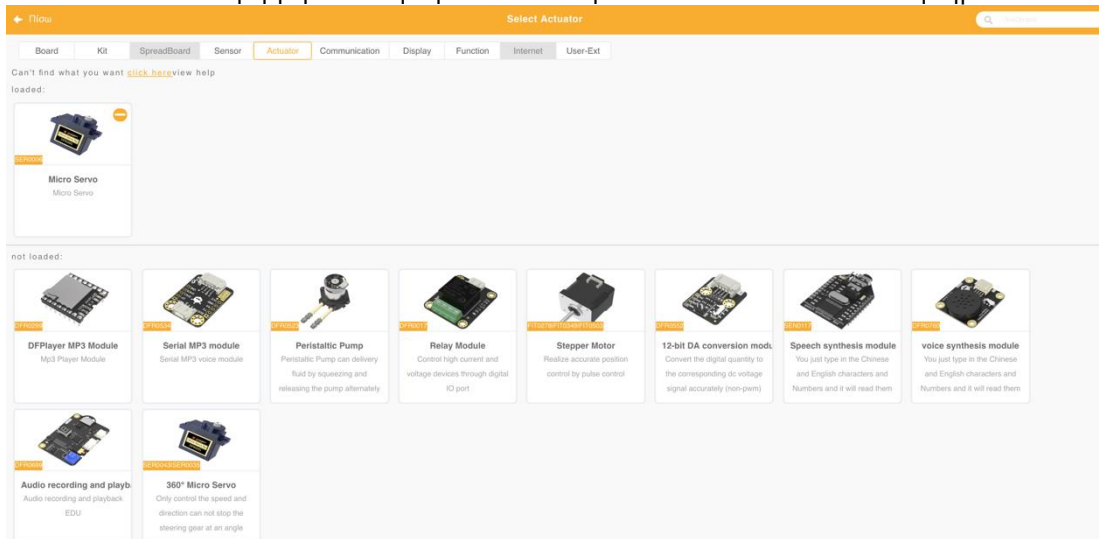
Από το εικονίδιο κάτω αριστερά του Mind+ επιλέγουμε με τη σειρά:



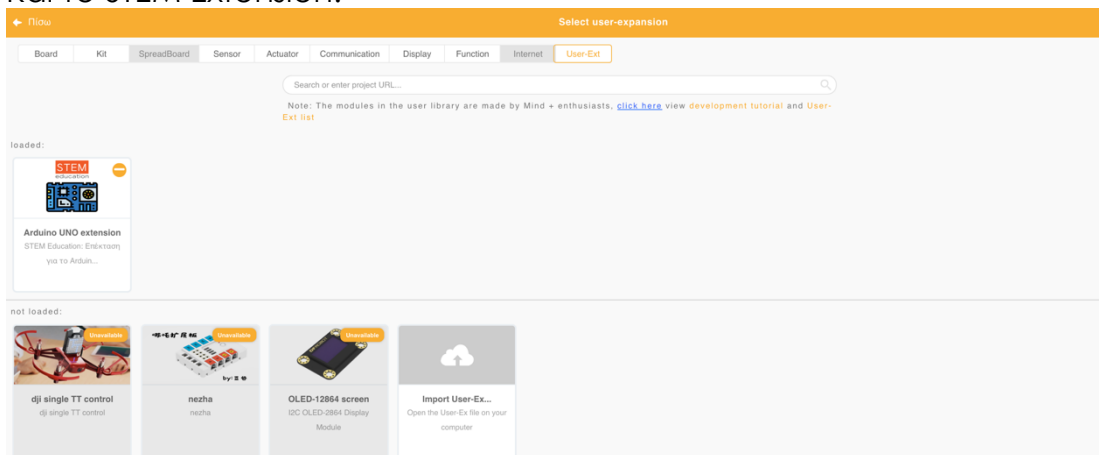
Arduino UNO:



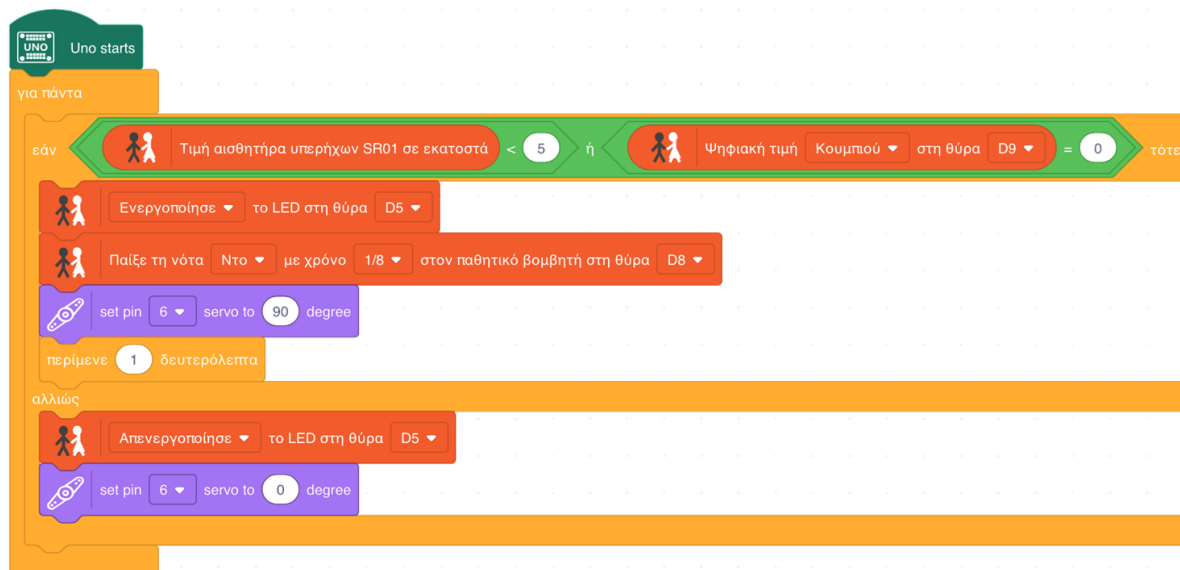
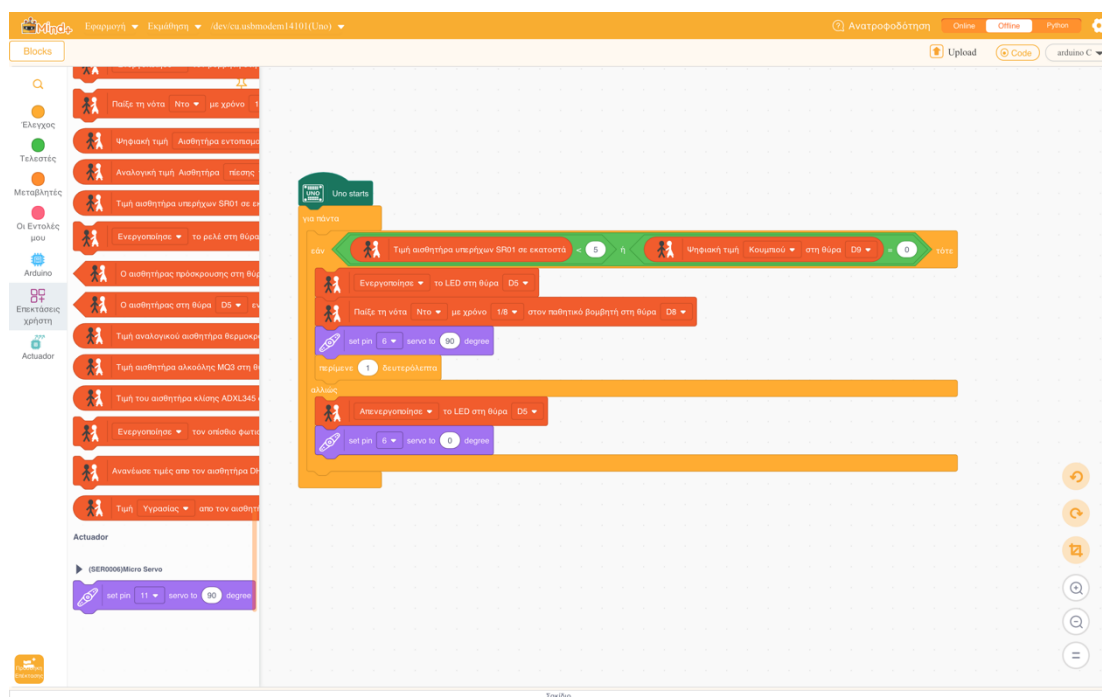
Επιπλέον επιλογή βιβλιοθήκη εντολών για τον mini Servo κινητήρα:



Και το STEM Extension:



Κώδικας:



Εάν ο αισθητήρας απόστασης επιστρέψει τιμή μικρότερη από 5 εκατοστά **ή** πατηθεί το κίτρινο button

Τότε

Άναψε το πράσινο LED

Παίξε τη νότα Ντο στον buzzer

Στρίψε τον άξονα του mini Servo στις 90 μοίρες

Αλλιώς

Σβήσε το LED

Στρίψε τον άξονα του mini Servo στις 0 μοίρες



Κώδικας σε Arduino C:

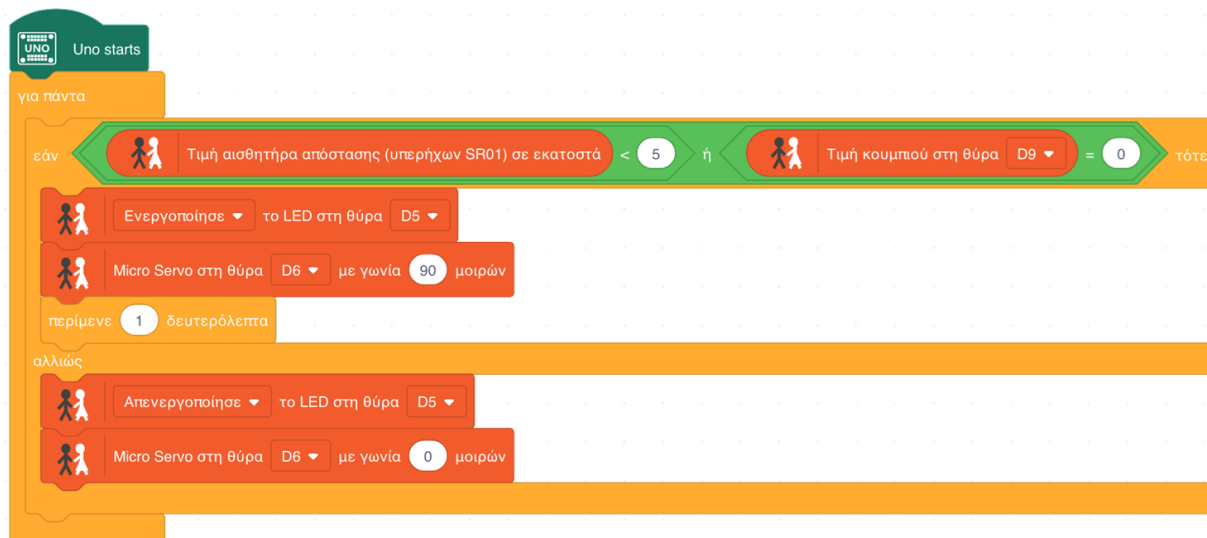
```
/*!  
 * MindPlus  
 * uno  
 *  
 */  
#include <SR01.h>  
#include <DFRobot_Servo.h>  
// Create an object  
Servo servo_6;  
  
// Main program start  
void setup() {  
    pinMode(3, OUTPUT);  
    pinMode(4, INPUT);  
    pinMode(9, INPUT);  
    pinMode(5, OUTPUT);  
    pinMode(8, OUTPUT);  
    servo_6.attach(6);  
}  
void loop() {  
    if (((getDistance(3,4)<5) || (digitalRead(9)==0))) {  
        digitalWrite(5, HIGH);  
        tone(8, 262, 125);  
        servo_6.angle(abs(90));  
        delay(1000);  
    }  
    else {  
        digitalWrite(5, LOW);  
        servo_6.angle(abs(0));  
    }  
}
```



Ο κώδικας με την νέα έκδοση του STEM extension

(Δεν χρειάζονται άλλες βιβλιοθήκες πλην αυτής, διότι περιέχονται όλες οι λειτουργίες σ' αυτό)

Με πλακίδια εντολών:



...και σε arduino C:

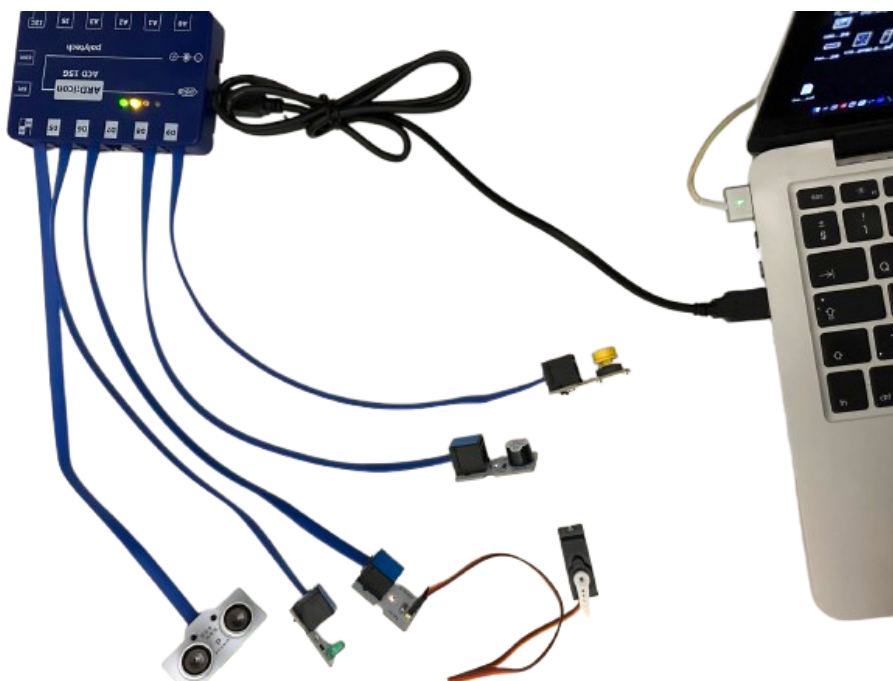
```
*!
* MindPlus
* uno
*/
#include <SR01.h>
#include <DFRobot_Servo.h>
// Create an object
Servo servo_6;

// Main program start
void setup() {
    pinMode(3, OUTPUT);
    pinMode(4, INPUT);
    pinMode(9, INPUT);
    pinMode(5, OUTPUT);
    servo_6.attach(6);
}
void loop() {
    if (((getDistance(3,4)<5) || (digitalRead(9)==0))) {
        digitalWrite(5, HIGH);
        servo_6.angle(90);
        delay(1000);
    }
    else {
        digitalWrite(5, LOW);
        servo_6.angle(0);
    }
}
```

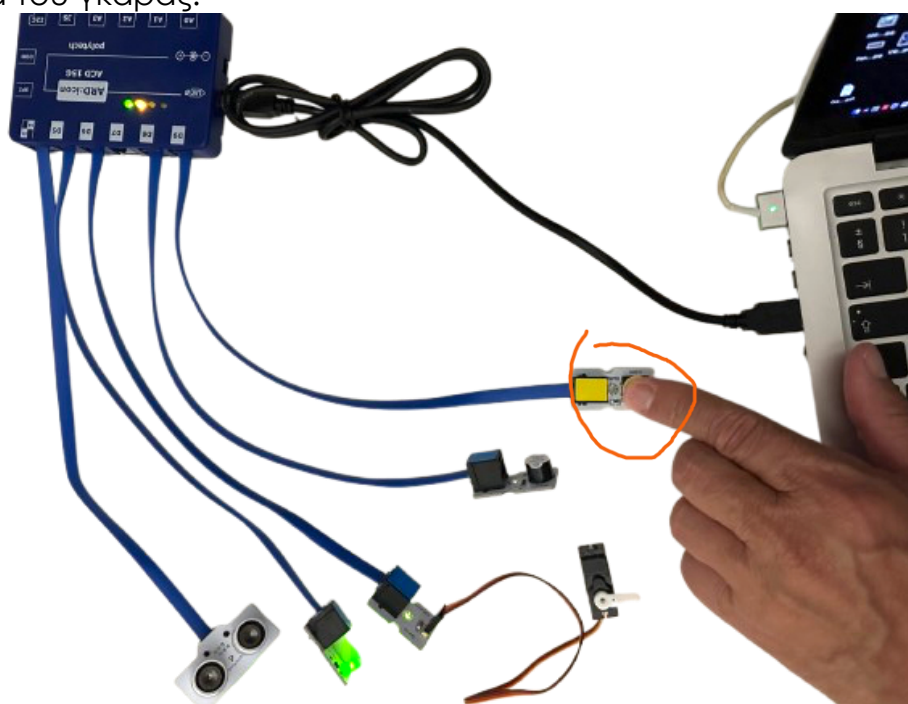


Αποτέλεσμα:

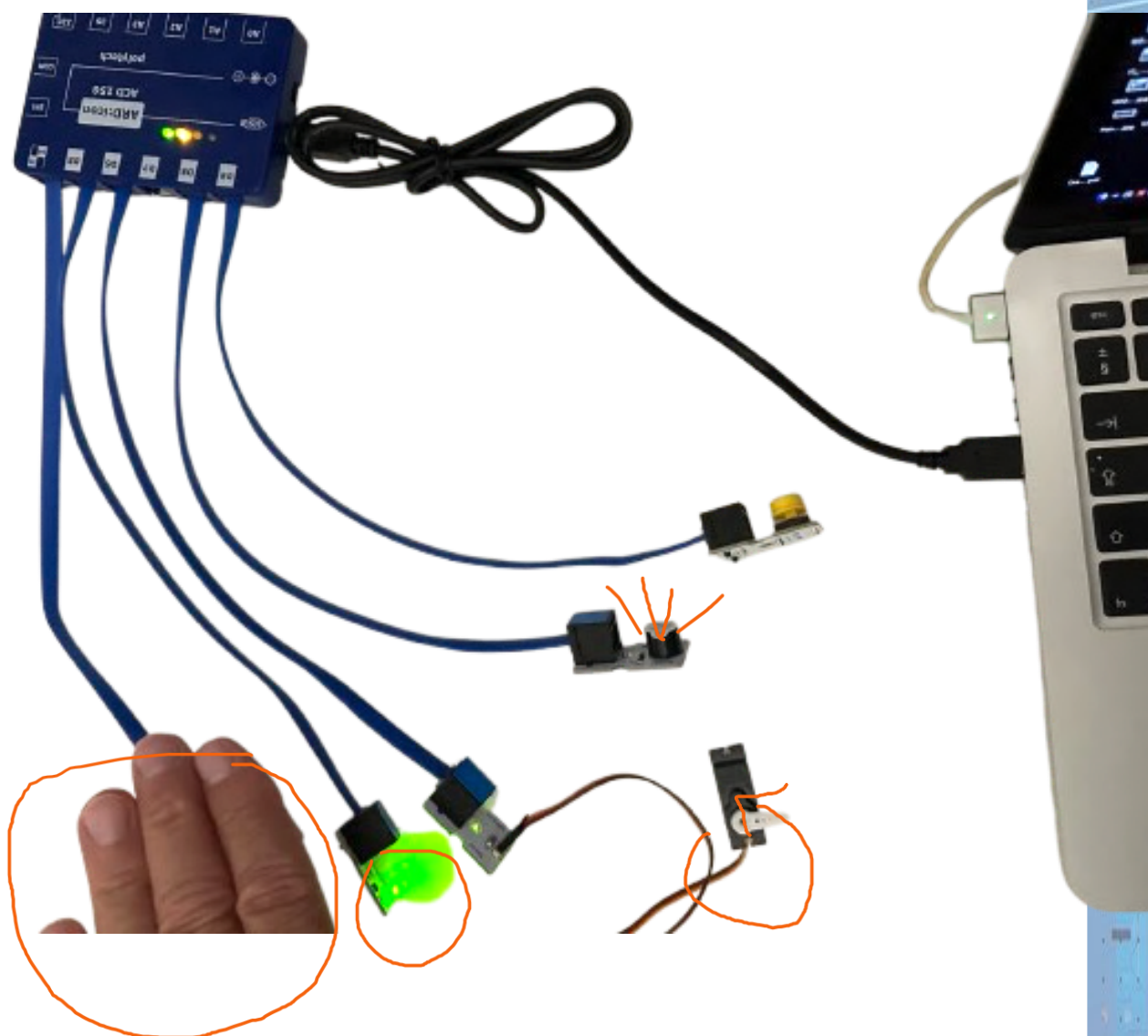
Δεν υπάρχει εμπόδιο στην εμβέλεια του αισθητήρα απόστασης, δεν πατιέται ο διακόπτης (κίτρινο button) άρα η μπάρα κλειστή, δεν ακούγεται ήχος και το πράσινο LED σβηστό.



Με το πάτημα του διακόπτη ανάβει το LED, ακούγεται ο ήχος και στρέφεται στις 90 μοίρες ο mini servo κινητήρας για να ανοίξει την μπάρα του γκαράζ.



Όταν έχουμε εμπόδιο μπροστά από τον αισθητήρα απόστασης, τότε ανάβει το πράσινο LED, ακούγεται ο ήχος και στρέφεται η μπάρα του mini servo στις 90 μοίρες (ανοίγει η μπάρα του γκαράζ)





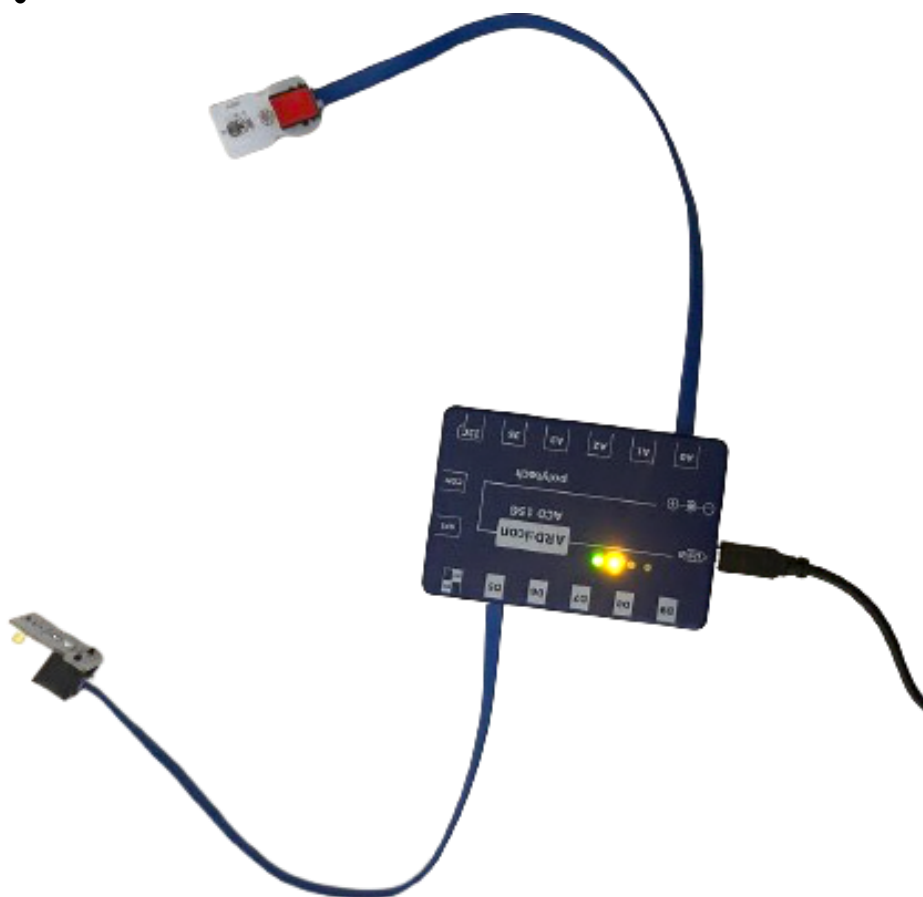
παράδειγμα: Η αξιοποίηση της μέτρησης φωτεινότητας του χώρου

Σενάριο:

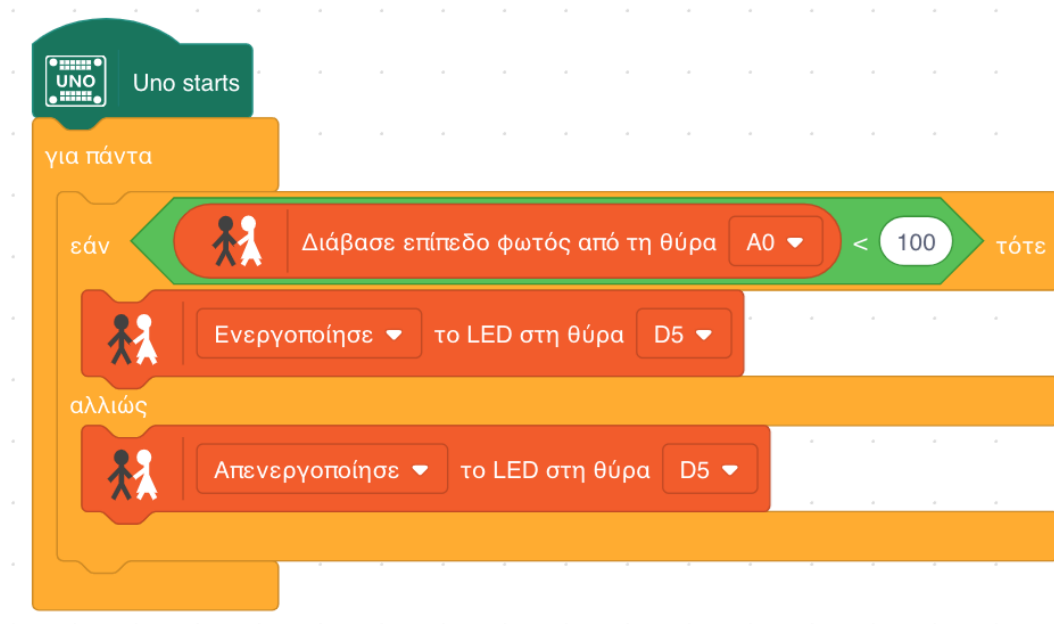
Δημιουργούμε ένα αυτοματισμό με βάση τον οποίο ανάβει ένα φως (κίτρινο LED) όταν πέσει ο φωτισμός του περιβάλλοντος (νυχτώσει) ή του χώρου που βρισκόμαστε (γίνει σκοτάδι).

Συνδεσμολογία:

- ARD:icon συνδεδεμένο στον Η/Υ μέσω USB.
- Θύρα A0: Αισθητήρας Φωτός LDR (Φωτοκύτταρο) AJS03.
- Θύρα D5: LED (Κίτρινο) DJX08.
-



Κώδικας:



Κώδικας σε Arduino C

```

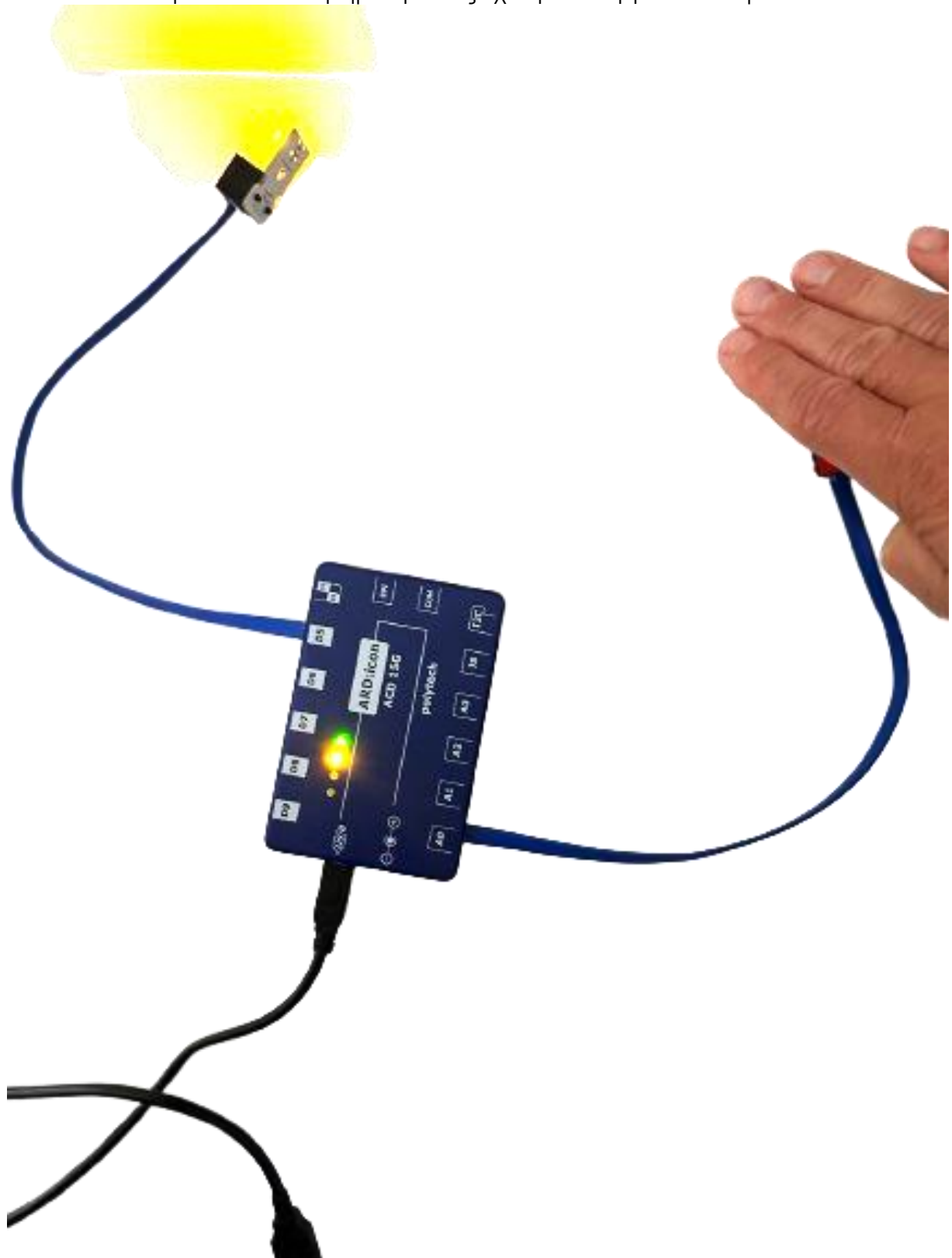
/*!
 * MindPlus
 * uno
 *
 */

// Main program start
void setup() {
    pinMode(5, OUTPUT);
}
void loop() {
    if ((analogRead(0)<100)) {
        digitalWrite(5, HIGH);
    }
    else {
        digitalWrite(5, LOW);
    }
}
  
```



Το αποτέλεσμα:

Με το σκέπασμα του αισθητήρα φωτός έχουμε άναμμα του κίτρινου LED.



Αν θέλουμε να δούμε τιμές αισθητήρων ή κάποιας μεταβλητής στη οθόνη του Η/Υ εφόσον δεν διαθέτουμε LCD οθόνη που βρίσκεται στο πακέτο S2 μπορούμε να αξιοποιήσουμε την μαύρη οθόνη κάτω δεξιά του Mind+

The screenshot displays the Arduino IDE environment. On the left, the 'Blocks' palette is visible with various sensor and control blocks. The main workspace contains a Scratch-style block diagram for an Arduino Uno. The blocks are as follows:

- Reading ultrasonic distance (cm):** Set to 2, with an echo pin of 3.
- play trumpet tone:** Set to Low G4 for 1/4 beat.
- read digital pin:** Set to 2, with an infrared receiver.
- read pin:** Set to 0, with a pulse width pattern of HIGH and a timeout of 20000.
- Serial output:** Set to 'hello' in string, wrapped.
- Is there data to read on serial port?** (Loop condition)
- read serial data:** (Loop body)
- set serial port baud rate to:** 9600.
- read serial data:** Set to int.
- serial output:** Set to 'as' in string, wrapped.
- serial output:** Set to 123.123, retained, 0 decimal.

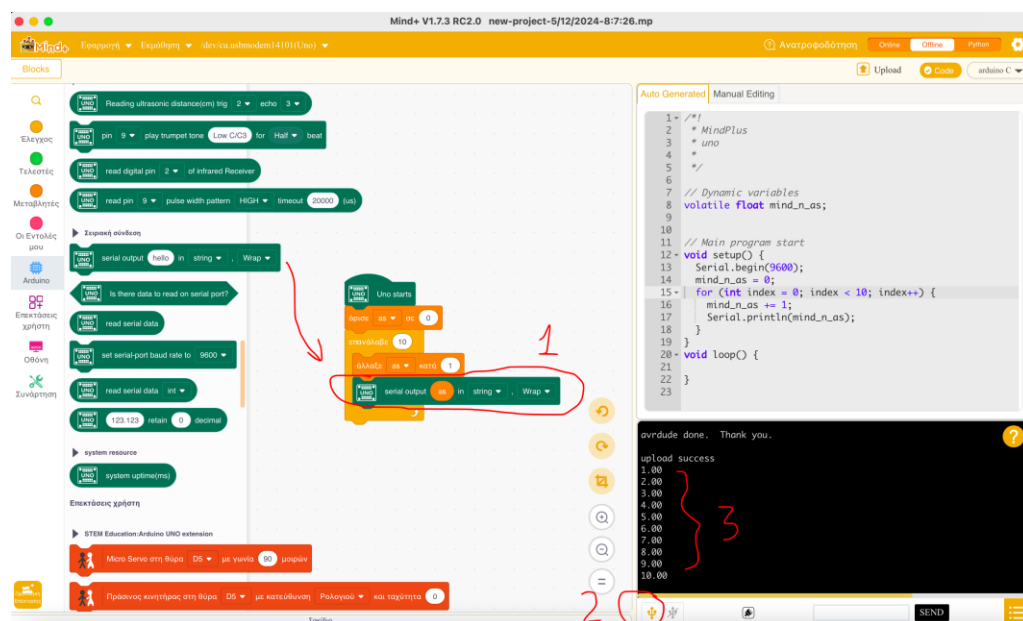
The background code in the editor is as follows:

```

1 // *
2 * MindPlus
3 * uno
4 *
5 */
6
7 // Dynamic variables
8 volatile float mind_n_as;
9
10
11 // Main program start
12 void setup() {
13   Serial.begin(9600);
14   mind_n_as = 0;
15   for (int index = 0; index < 10; index++) {
16     mind_n_as += 1;
17     Serial.println(mind_n_as);
18   }
19 }
20 void loop() {
21   //
22 }
23

```

The bottom status bar shows the Arduino IDE version (1.8.19) and the board (Arduino Uno).



11^ο

παράδειγμα: Τιμές αισθητήρα πίεσης λεπτής μεμβράνης στην Mind+ οθόνη

Προαπαιτούμενα (υλικά και λογισμικό):

1. ARD:icon (ACD 15G).
2. USB Καλώδιο σύνδεσης με H/Y.
3. Mind+.
4. STEM Extension.
5. Αισθητήρας πίεσης λεπτής μεμβράνης AJS07

Σενάριο

Θα αξιοποιήσουμε τον Αισθητήρα πίεσης λεπτής μεμβράνης AJS0, για να βρίσκουμε πίεση που ασκούμε στην μεμβράνη και να το προβάλλουμε στην οθόνη του Mind+

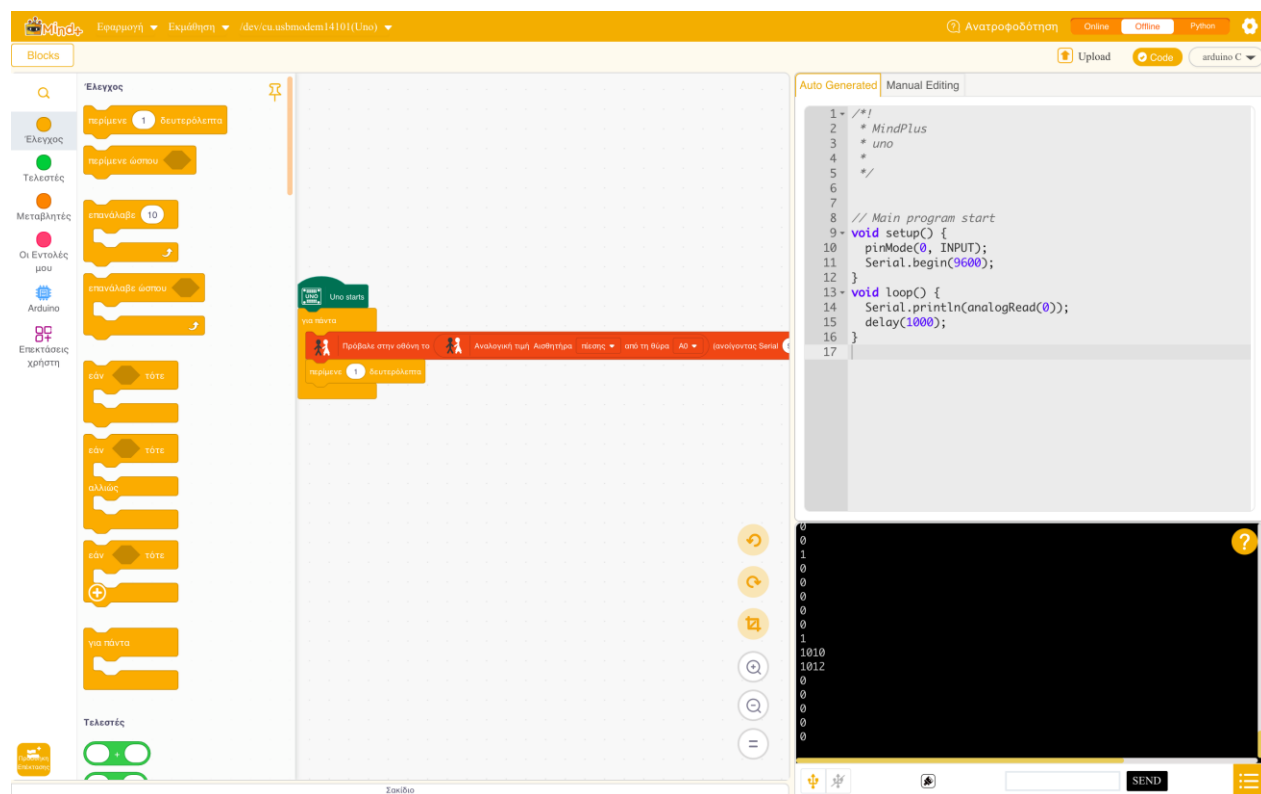
Θα χρειαστούμε



Η συνδεσμολογία



Ο κώδικας



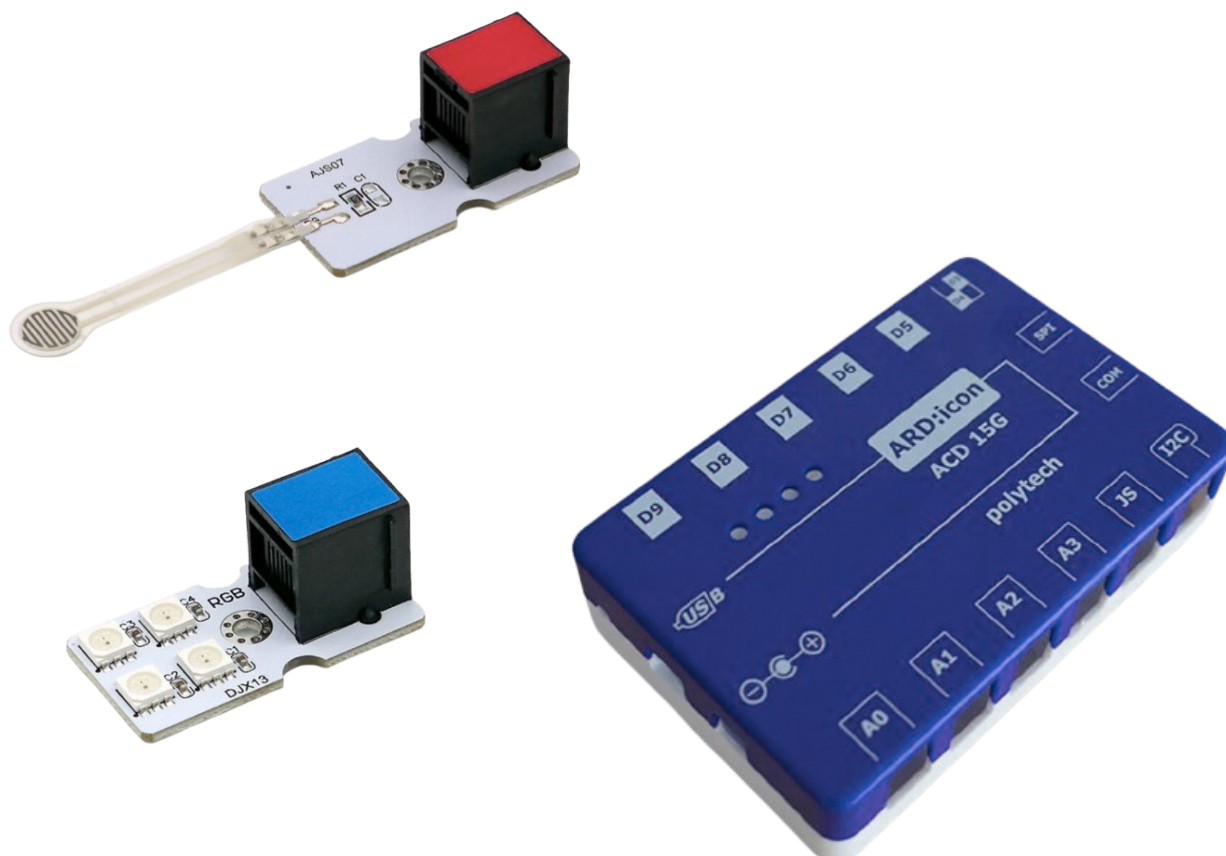
12^ο

παράδειγμα: Οπτικοποίηση της πίεσης που ασκούμε στη μεμβράνη.

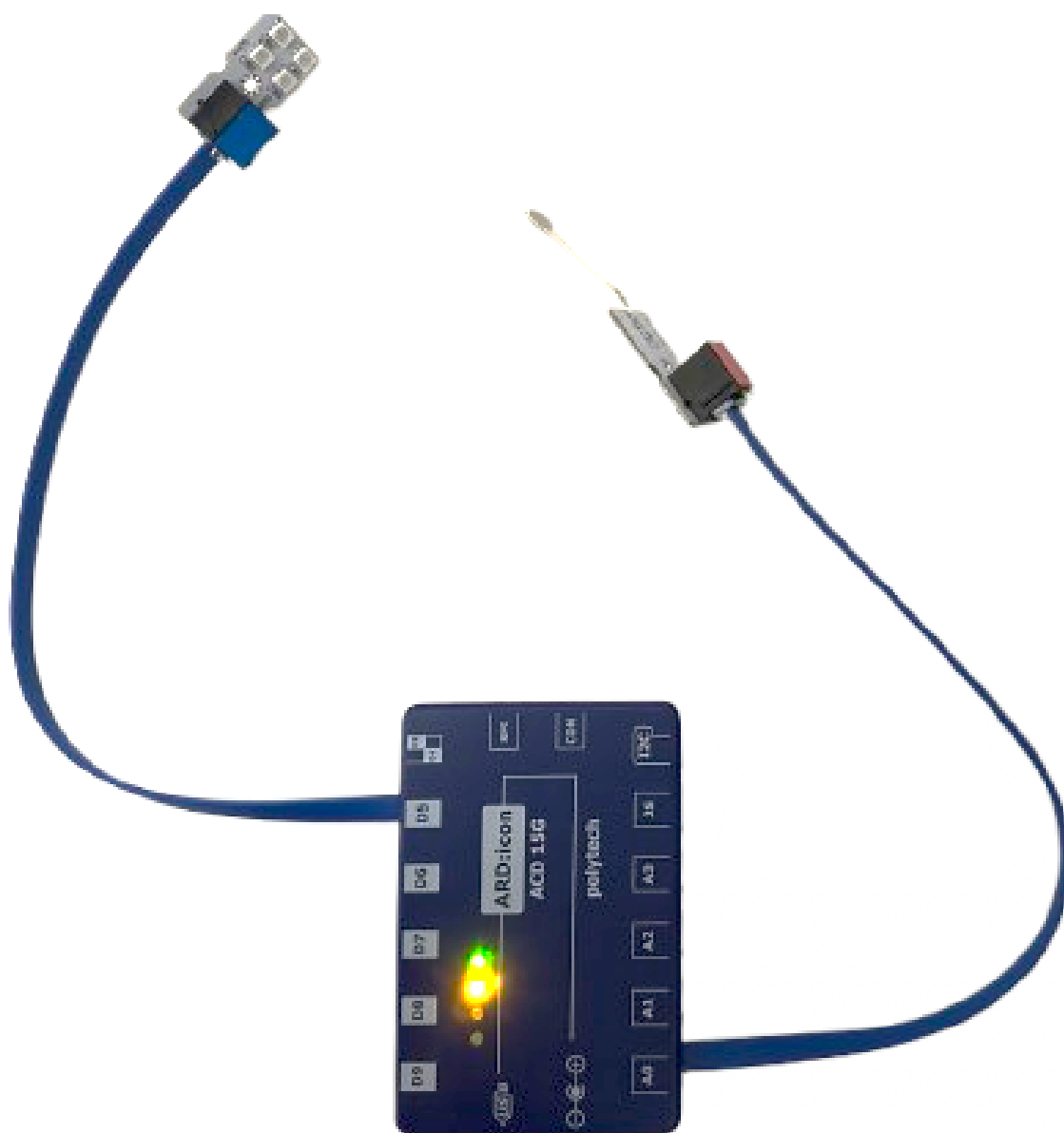
Σενάριο

Θα αξιοποιήσουμε τον αισθητήρα πίεσης λεπτής μεμβράνης AJS0, για να βρίσκουμε την πίεση που ασκούμε στην μεμβράνη και ανάλογα με αυτή θα ανάβουμε τα τέσσερα leds του LED (RGB) DJX13 από πράσινο το πρώτο (ομαλή πίεση), μέχρι κόκκινο το τέταρτο (ισχυρή πίεση).

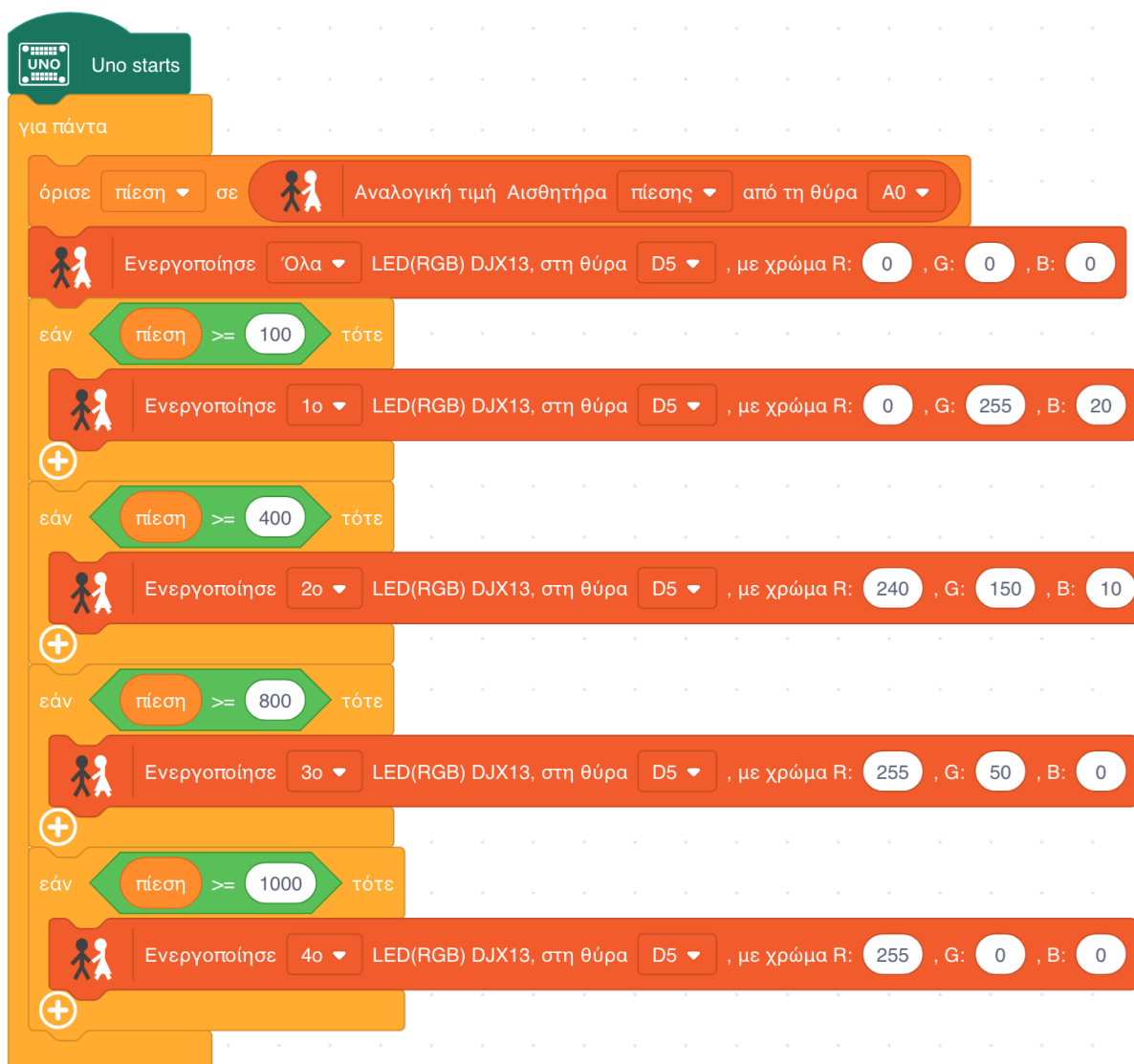
Θα χρειαστούμε



Συνδεσμολογία



Ο κώδικας



Δημιουργούμε και χρησιμοποιούμε μια **μεταβλητή με το όνομα: «πίεση»**, ώστε να έχουμε μια δειγματοληψία τιμής από τον αισθητήρα, την οποία θα αξιοποιούμε σε όλες μας τις συγκρίσεις.



Ο κώδικας σε Arduino C

```

/!*
 * MindPlus
 * uno
 *
 */
#include <Adafruit_NeoPixel.h>

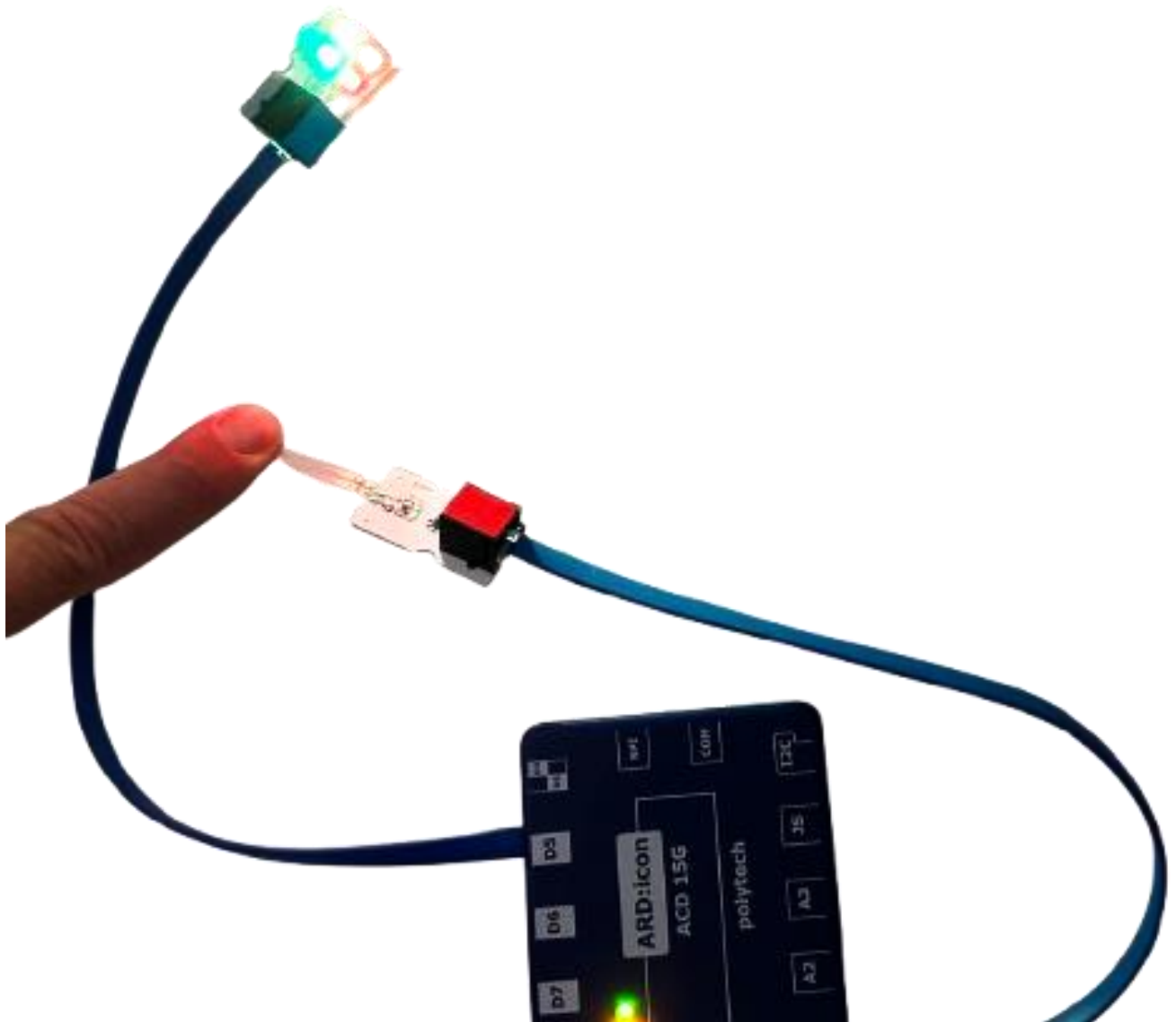
// Dynamic variables
volatile float mind_n__;
// Create an object
Adafruit_NeoPixel pixels(4, 5, NEO_GRB + NEO_KHZ800);

// Main program start
void setup() {
    pinMode(0, INPUT);
    pixels.begin();
}
void loop() {
    mind_n__ = analogRead(0);
    pixels.setPixelColor(0, pixels.Color(0, 0, 0));
    pixels.setPixelColor(1, pixels.Color(0, 0, 0));
    pixels.setPixelColor(2, pixels.Color(0, 0, 0));
    pixels.setPixelColor(3, pixels.Color(0, 0, 0));
    pixels.show();
    if ((mind_n__ >= 100)) {
        pixels.setPixelColor(0, pixels.Color(0, 255, 20));
        pixels.show();
    }
    if ((mind_n__ >= 400)) {
        pixels.setPixelColor(1, pixels.Color(240, 150, 10));
        pixels.show();
    }
    if ((mind_n__ >= 800)) {
        pixels.setPixelColor(2, pixels.Color(255, 50, 0));
        pixels.show();
    }
    if ((mind_n__ >= 1000)) {
        pixels.setPixelColor(3, pixels.Color(255, 0, 0));
        pixels.show();
    }
}

```



Το αποτέλεσμα





ADR:icon
Δραστηριότητες
με Δομικά
υλικά



1^η δραστηριότητα: Όχημα με ρυθμιστή ταχύτητας (Κινητήρες, Ποτενσιόμετρο)

A. Στοιχεία Δραστηριότητας

Σενάριο βιωματικής εμπλοκής μαθητών για τη κατασκευή και τη διερεύνηση των παραγόντων που καθιστούν μια ρομποτική κατασκευή ικανή να κινείται με ταχύτητα που ρυθμίζεται από ένα ποτενσιόμετρο. Την αξιοποίηση IoT (αισθητήρων), των αυτοματισμών που απαιτούνται, και την ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον. Αναφέρεται σε μαθητές Ε' Δημοτικού ή/και Α' Γυμνασίου στο Μάθημα Πληροφορικής (προγραμματιστικές δομές) και υλοποιείται στο εργαστήριο δεξιοτήτων.

Σκοπός:

Οι Μαθητές/τριες να διερευνήσουν και να καταλήξουν μέσα από μελέτη, έρευνα, δοκιμές και παρατήρηση στη δημιουργία ενός απλοϊκού οχήματος και την κίνησή του με ρυθμιζόμενη ταχύτητα. Η δημιουργική σύνθεση/ένταξη κινητήρων και του ποτενσιόμετρου στην παθητική κατασκευή και η κατάλληλη χρήση δομών επανάληψης και συγκρίσεων, ώστε με τον κατάλληλο προγραμματισμό να κινείται με ταχύτητα που ρυθμίζει ο οδηγός

Στόχοι

- **Επισημαίνω** Τη χρησιμότητα αξιοποίησης του ποτενσιόμετρου και των κινητήρων (IoT) για την για την κίνηση και τη ρύθμιση της ταχύτητας οχημάτων..
- **Αναγνωρίζω** Τις συνθήκες για την ομαλή κίνηση το "ηλεκτρικού" οχήματος, καθώς και τα δομικά υλικά (παθητικά και ενεργά) που απαιτούνται για την κατασκευή αυτή.
- **Σχεδιάζω** Το παραδοτέο και τα τμήματα από τα οποία αποτελείται σχολιάζοντάς τα.
- **Αναλύω** Το έργο σε μικρότερες επιμέρους αυτοτελείς ενότητες με τις οποίες θα ασχοληθώ τμηματικά.
- **Συναρμολογώ** Τα δομικά υλικά πρώτα τα παθητικά και στη συνέχεια τα ενεργά.
- **Συνδυάζω** Με τον καλύτερο (σύντομο και έξυπνο) τρόπο τα πλακίδια εντολών που απαιτούνται ώστε να θέσω σε κίνηση υπό συνθήκες το όχημά μου.
- **Σχολιάζω** τη μέθοδο που ακολούθησα ώστε να φτάσω στην ολοκλήρωση του τελικού μου παραδοτέου με βάση το έργο που μου έχει ανατεθεί
- **Προβληματίζομαι** Σε σχέση με την κίνηση του οχήματος και των αλλαγών που χρειάζεται να κάνω στην κατασκευή αλλά και το πρόγραμμά μου, ώστε να γίνει πιο ομαλή και πιο ταχεία.
- **Ανταποκρίνομαι** Στο ρόλο που μου έχει ανατεθεί με τον καλύτερο δυνατό τρόπο.
- **Κρίνω** Το αποτέλεσμα που παρουσιάζω και τον τρόπο εργασίας της ομάδας, σε σχέση με την αρχική εισήγηση και ανάθεση του έργου.

Χρόνος μάθησης

2 ώρες και 40 λεπτά.

Μέγεθος τάξης

14 Μαθητές/τριες.

B. Υλικά



ARD:icon

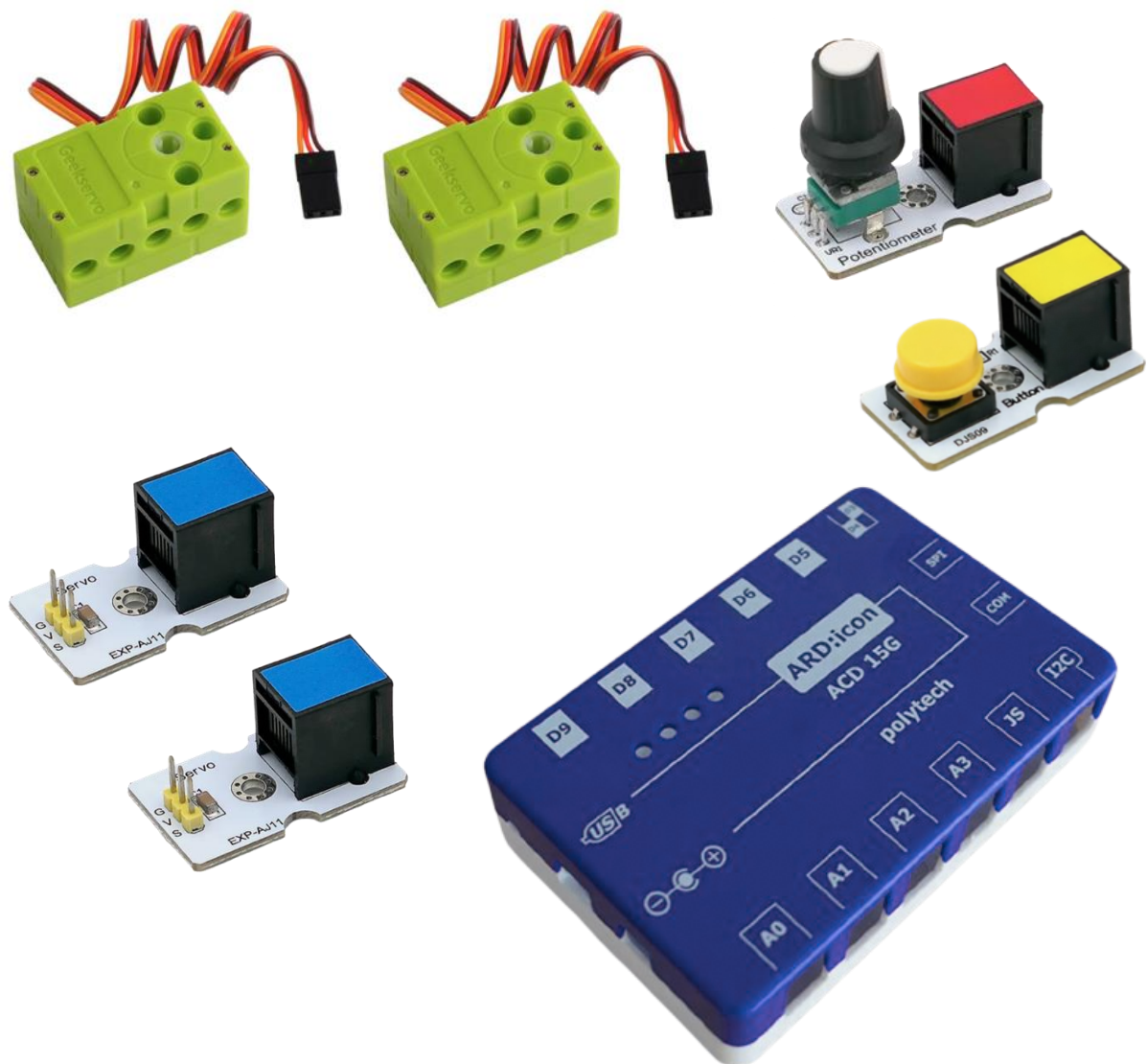
Κινητήρες × 2

Ποτενσιόμετρο

Διακόπτης

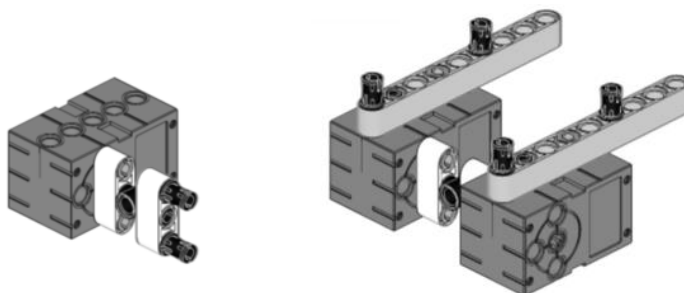
Καλώδια × 5

Παθητικά δομικά υλικά

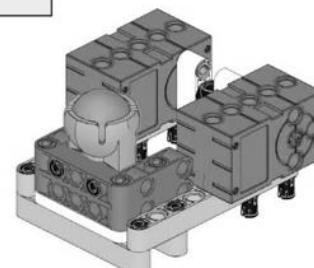
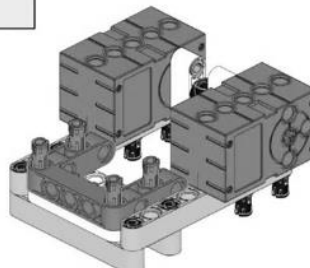
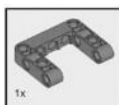
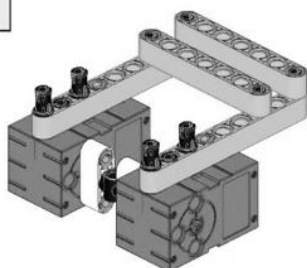


Γ. Συναρμολόγηση

1



2



3

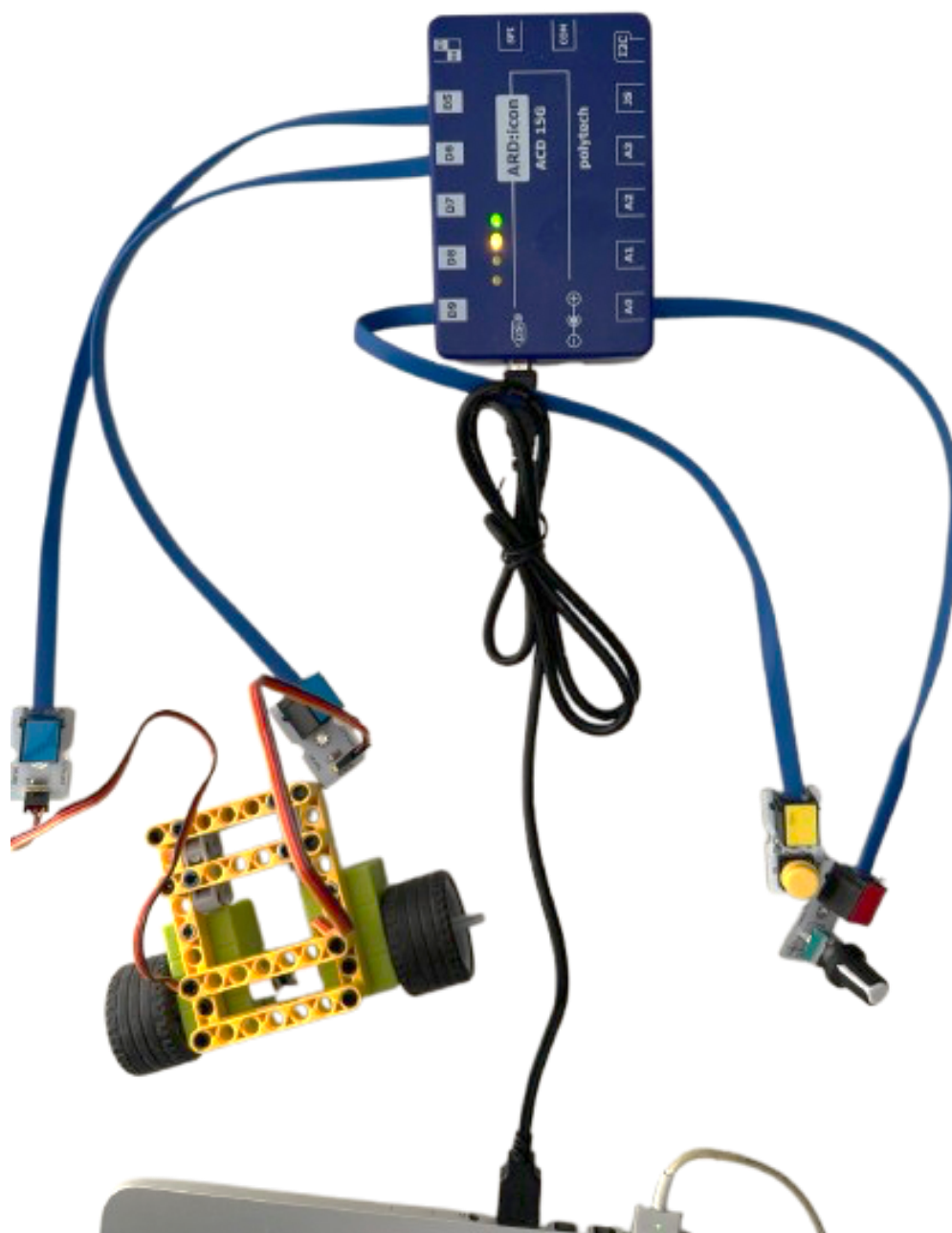
Μένει να βρούμε τρόπο στήριξης του ADR:icon και των μπαταριών του πάνω στο όχημα που ετοιμάσαμε...

Ο άξονας του κινητήρα περιστρέφεται δεξιόστροφα ή αριστερόστροφα με ταχύτητα που ρυθμίζεται από το λογισμικό, έσω της εντολής servo γωνίας σύμφωνα με τον πίνακα:

Γωνία	Λειτουργία
90	Stop
91 έως 180	Δεξιόστροφη περιστροφή με ταχύτητα που αυξάνει ανάλογα με την αύξηση της τιμής της γωνίας
0 έως 89	Αριστερόστροφη περιστροφή με ταχύτητα που αυξάνει ανάλογα με την μείωση της τιμής της γωνίας

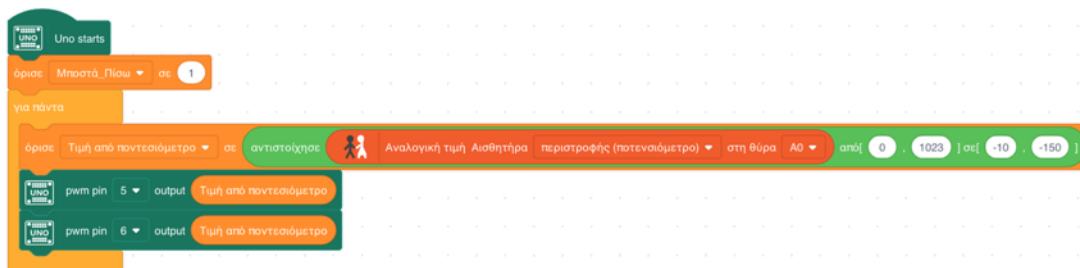


Σε πλήρη διάταξη/συνδεσμολογία

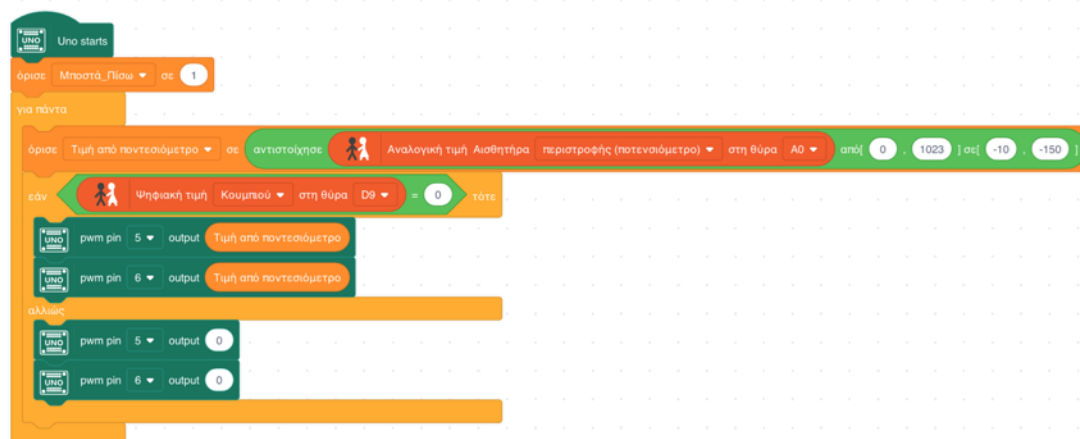


Κώδικας:

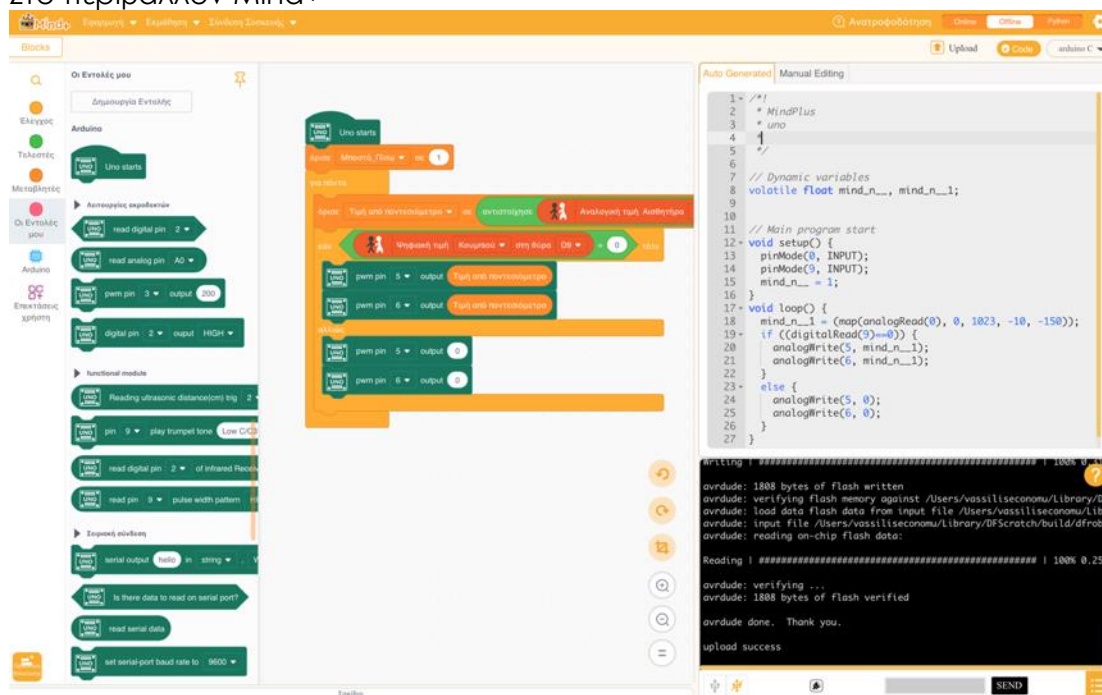
Αρχικά κίνηση με ταχύτητα που ρυθμίζεται από το ποτενσιόμετρο



Προσθέτουμε και ένα κουμπί που ξεκινά και σταματά την κίνηση (όπως το αντίστοιχο πεντάλ αυτοκινήτου)



Στο περιβάλλον Mind+



Και σε Arduino C

```
/*!  
 * MindPlus  
 * uno  
 *  
 */  
  
// Dynamic variables  
volatile float mind_n__, mind_n__1;  
  
// Main program start  
void setup() {  
    pinMode(0, INPUT);  
    pinMode(9, INPUT);  
    mind_n__ = 1;  
}  
void loop() {  
    mind_n__1 = (map(analogRead(0), 0, 1023, -10, -150));  
    if ((digitalRead(9)==0)) {  
        analogWrite(5, mind_n__1);  
        analogWrite(6, mind_n__1);  
    }  
    else {  
        analogWrite(5, 0);  
        analogWrite(6, 0);  
    }  
}
```



2η δραστηριότητα: Όχημα με ρυθμιστή ταχύτητας και κατεύθυνσης

A. Στοιχεία Δραστηριότητας

Σενάριο βιωματικής εμπλοκής μαθητών για τη κατασκευή και τη διερεύνηση των παραγόντων που καθιστούν μια ρομποτική κατασκευή ικανή να κινείται σε κατεύθυνσης και με ταχύτητα που ρυθμίζεται από ένα ποτενσιόμετρο. Την αξιοποίηση IoT (αισθητήρων), των αυτοματισμών που απαιτούνται, και την ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον. Αναφέρεται σε μαθητές Ε' Δημοτικού ή/και Α' Γυμνασίου στο Μάθημα Πληροφορικής (προγραμματιστικές δομές) και υλοποιείται στο εργαστήριο δεξιοτήτων.

Σκοπός:

Οι Μαθητές/τριες να διερευνήσουν και να καταλήξουν μέσα από μελέτη, έρευνα, δοκιμές και παρατήρηση στη δημιουργία ενός απλοϊκού οχήματος και την κίνησή του με ρυθμιζόμενη ταχύτητα και κατεύθυνση. Η δημιουργική σύνθεση/ένταξη κινητήρων και του ποτενσιόμετρου στην παθητική κατασκευή και η κατάλληλη χρήση δομών επανάληψης και συγκρίσεων, ώστε με τον κατάλληλο προγραμματισμό να κινείται με ταχύτητα και στην κατεύθυνση που ρυθμίζει ο οδηγός.

Στόχοι

- **Επισημαίνω** Τη χρησιμότητα αξιοποίησης του ποτενσιόμετρου και των κινητήρων (IoT) για την για την κίνηση και τη ρύθμιση της ταχύτητας οχημάτων..
- **Αναγνωρίζω** Τις συνθήκες για την ομαλή κίνηση το "ηλεκτρικού" οχήματος, καθώς και τα δομικά υλικά (παθητικά και ενεργά) που απαιτούνται για την κατασκευή αυτή.
- **Σχεδιάζω** Το παραδοτέο και τα τμήματα από τα οποία αποτελείται σχολιάζοντάς τα.
- **Αναλύω** Το έργο σε μικρότερες επιμέρους αυτοτελείς ενότητες με τις οποίες θα ασχοληθώ τμηματικά.
- **Συναρμολογώ** Τα δομικά υλικά πρώτα τα παθητικά και στη συνέχεια τα ενεργά.
- **Συνδυάζω** Με τον καλύτερο (σύντομο και έξυπνο) τρόπο τα πλακίδια εντολών που απαιτούνται ώστε να θέσω σε κίνηση υπό συνθήκες το όχημά μου.
- **Σχολιάζω** τη μέθοδο που ακολούθησα ώστε να φτάσω στην ολοκλήρωση του τελικού μου παραδοτέου με βάση το έργο που μου έχει ανατεθεί
- **Προβληματίζομαι** Σε σχέση με την κίνηση του οχήματος και των αλλαγών που χρειάζεται να κάνω στην κατασκευή αλλά και το πρόγραμμά μου, ώστε να γίνει πιο ομαλή και πιο ταχεία.
- **Ανταποκρίνομαι** Στο ρόλο που μου έχει ανατεθεί με τον καλύτερο δυνατό τρόπο.
- **Κρίνω** Το αποτέλεσμα που παρουσιάζω και τον τρόπο εργασίας της ομάδας, σε σχέση με την αρχική εισήγηση και ανάθεση του έργου.

Χρόνος μάθησης

2 ώρες και 40 λεπτά.

Μέγεθος τάξης

14 Μαθητές/τριες.



Β. Υλικά

ARD:icon

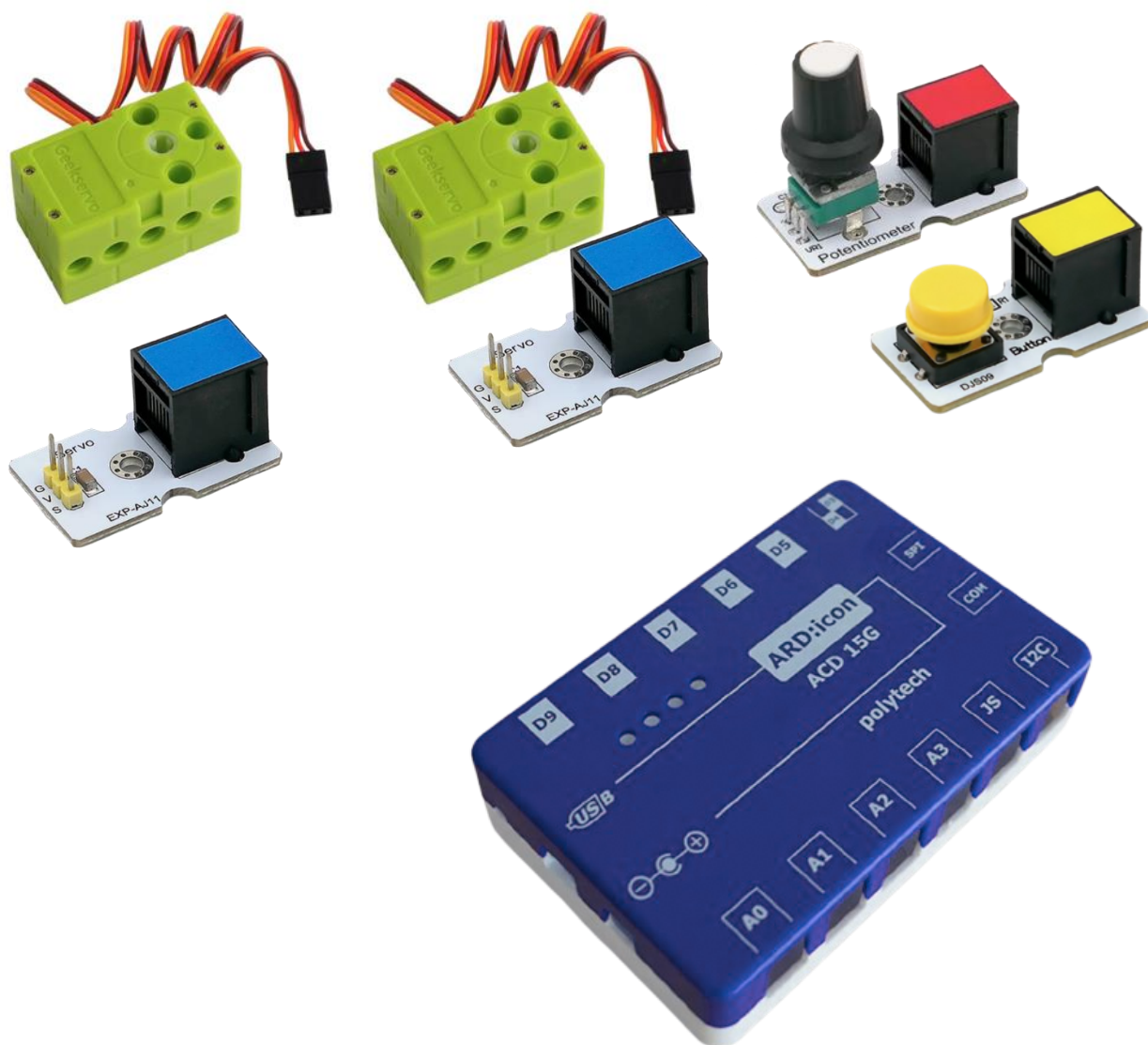
Κινητήρες × 2

Ποτενσιόμετρο

Διακόπτης

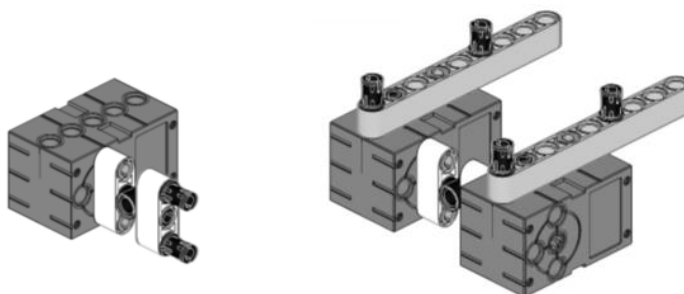
Καλώδια × 5

Παθητικά δομικά υλικά

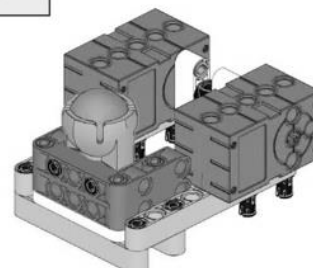
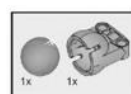
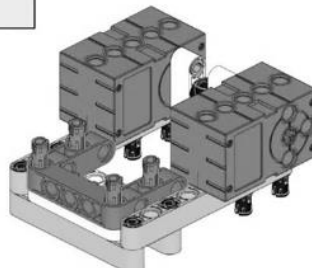
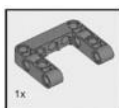
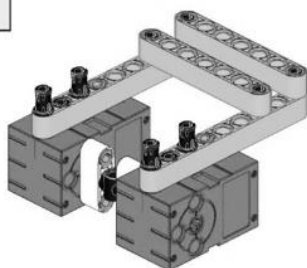


Γ. Συναρμολόγηση

1



2

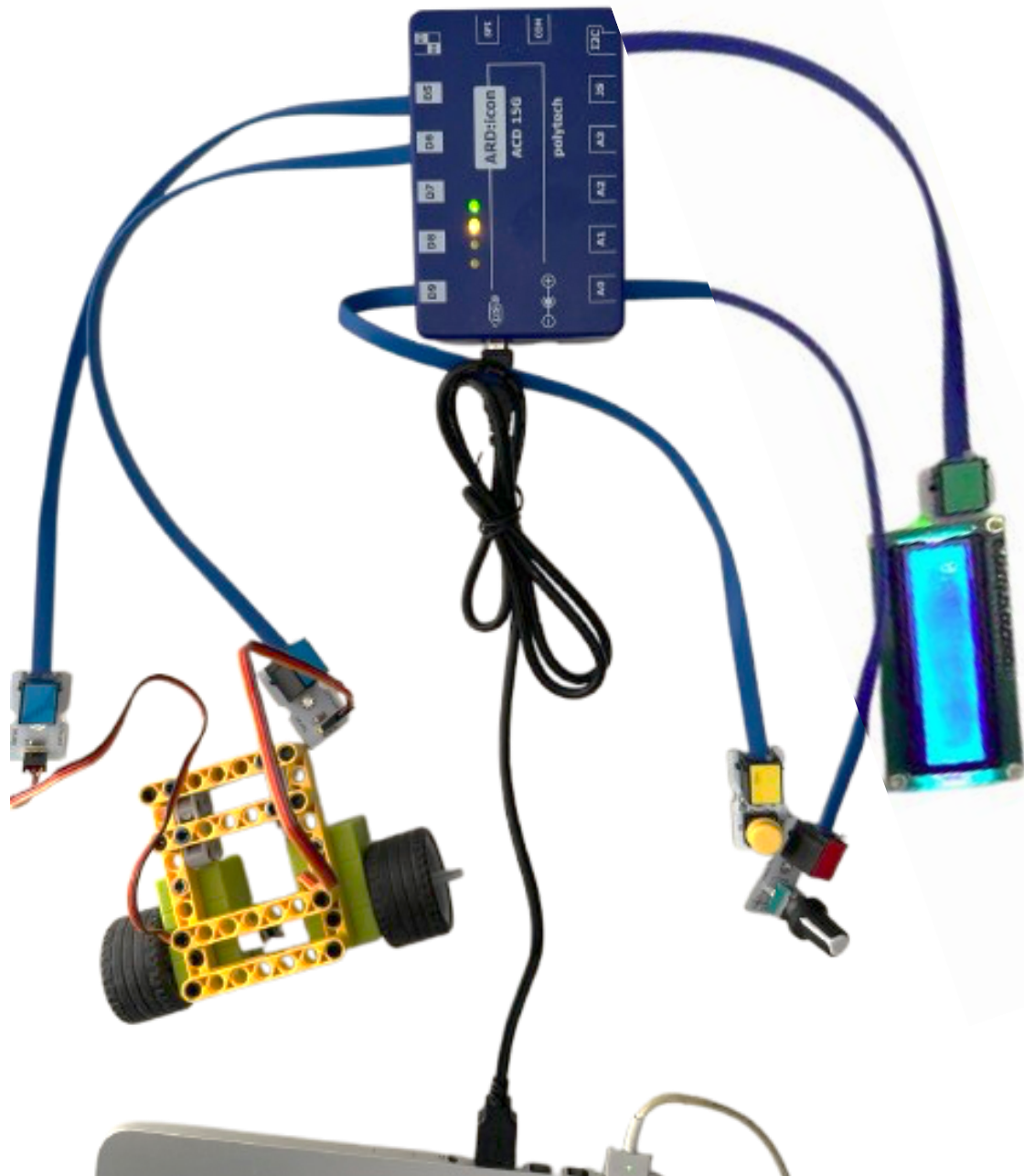


3

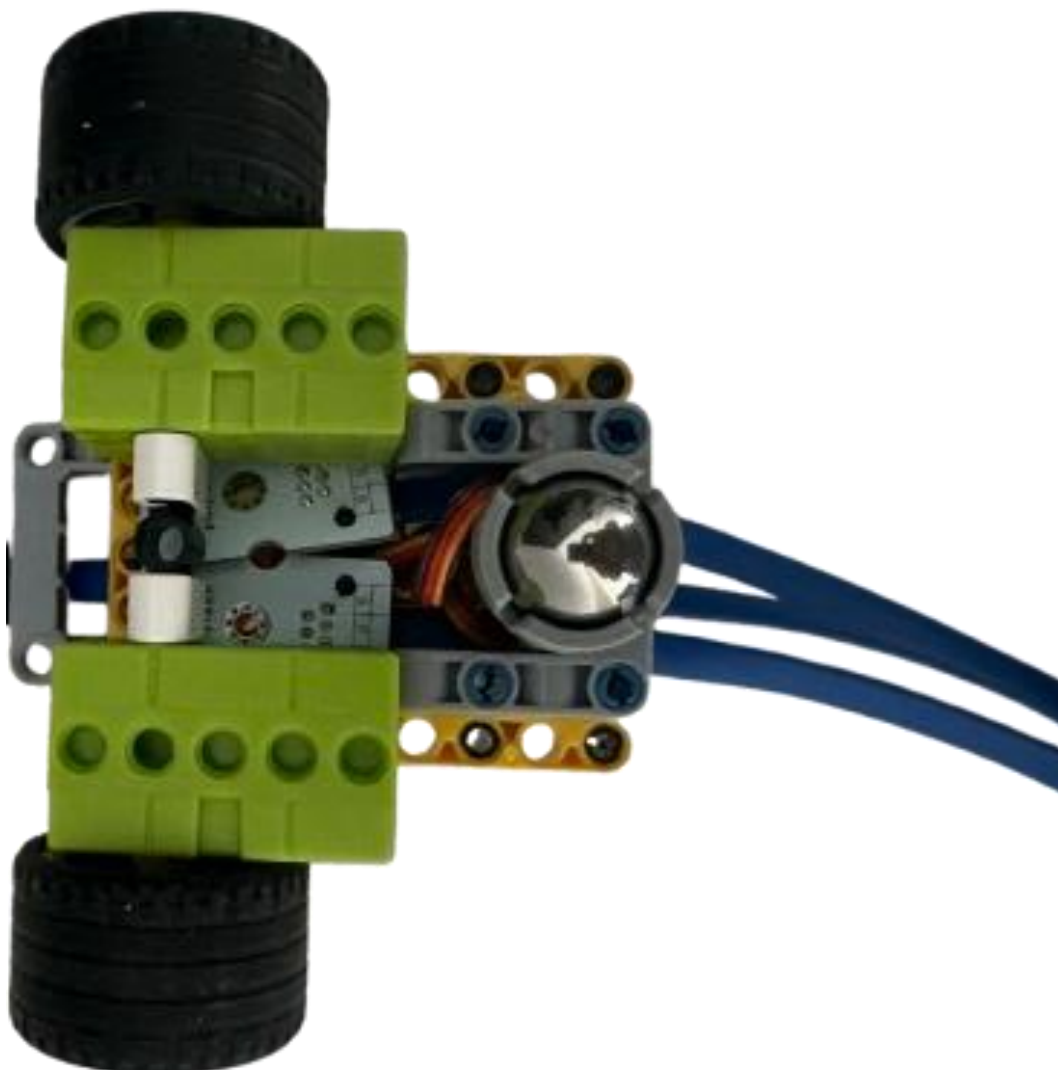
Μένει να βρούμε τρόπο στήριξης του ADR:icon και των μπαταριών του πάνω στο όχημα που ετοιμάσαμε...



Σε πλήρη διάταξη/συνδεσμολογία

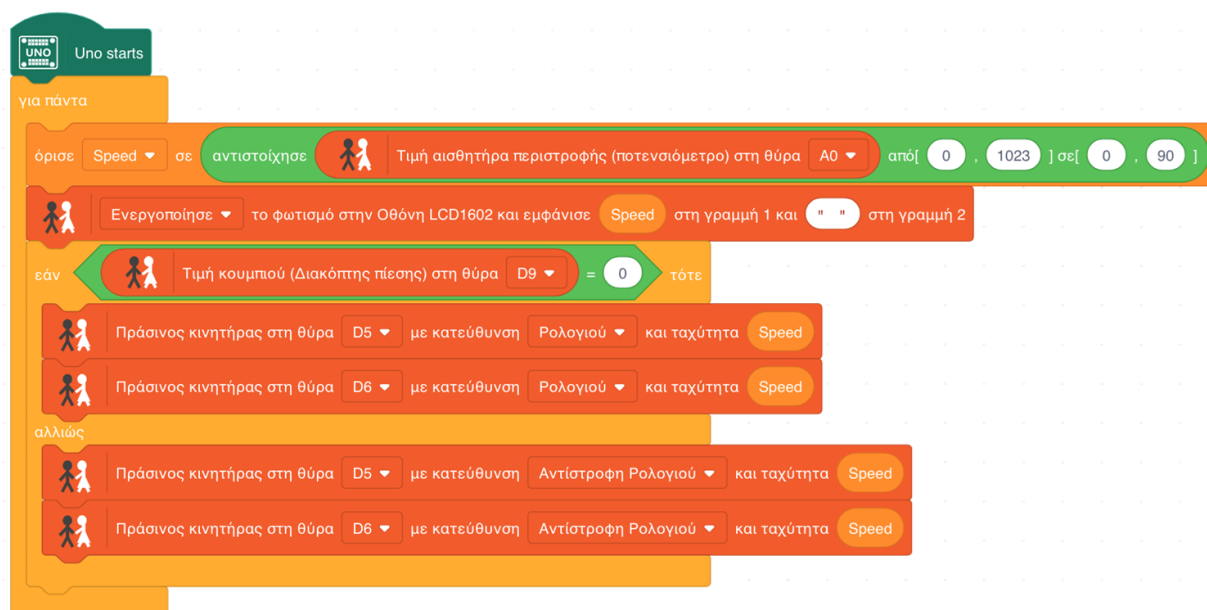


Λεπτομέρεια του οχήματος με τακτοποιημένα καλώδια.



Δ. Κώδικας

(με την νέα έκδοση του extension που περιέχει περισσότερες λειτουργίες):



Ας δούμε με μεγαλύτερη λεπτομέρεια του κώδικα αυτό.

Αρχικά γίνεται η αντιστοίχιση των τιμών που επιστρέφει το ποτενσιόμετρο (0-1023) στις τιμές της ταχύτητας (0=σταματημένος κινητήρας, 90 το μέγιστο της ταχύτητας περιστροφής του)

Δημιουργούμε και μια μεταβλητή [Speed] που θα κρατά την τιμή της ταχύτητας περιστροφής των αξόνων των κινητήρων.

Στη συνέχεια η τιμή προβάλλεται στην οθόνη LED (είναι το κοντέρ του οχήματός μας)

Ελέγχουμε το πάτημα του κουμπιού. Αν πατηθεί και μένει πατημένο τότε αλλάζουμε την κατεύθυνση περιστροφής.

Στην ουσία ορίζουμε την ταχύτητα περιστροφής των κινητήρων από το ποτενσιόμετρο και την κατεύθυνση περιστροφής από το κουμπί (διακόπτης πίεσης)



...και σε arduino C:

```

/!*
 * MindPlus
 * uno
 *
 */
#include <LCD_I2C.h>
#include <DFRobot_Servo.h>

// Dynamic variables
volatile float mind_n_Speed;
// Create an object
LCD_I2C lcd(0x27, 16, 2);
Servo servo_5;
Servo servo_6;

// Main program start
void setup() {
    pinMode(0, INPUT);
    lcd.begin();
    pinMode(9, INPUT);
    servo_5.attach(5);
    servo_6.attach(6);
}
void loop() {
    mind_n_Speed = (map(analogRead(0), 0, 1023, 0, 90));
    lcd.clear();
    lcd.backlight();
    lcd.setCursor(0, 0);
    lcd.print(mind_n_Speed);
    lcd.setCursor(0, 1);
    lcd.print("");
    if ((digitalRead(9)==0)) {
        servo_5.angle(90+mind_n_Speed);
        servo_6.angle(90+mind_n_Speed);
    }
    else {
        servo_5.angle(90-mind_n_Speed);
        servo_6.angle(90-mind_n_Speed);
    }
}
}

```



3η δραστηριότητα: Όχημα που η κίνησή του σταματά σε εμπόδιο

A. Στοιχεία Δραστηριότητας

Σενάριο βιωματικής εμπλοκής μαθητών για τη κατασκευή και τη διερεύνηση των παραγόντων που καθιστούν μια ρομποτική κατασκευή ικανή να κινείται και να σταματά σε εμπόδια. Την αξιοποίηση IoT (αισθητήρων), των αυτοματισμών που απαιτούνται, και την ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον. Αναφέρεται σε μαθητές Ε' Δημοτικού ή/και Α' Γυμνασίου στο Μάθημα Πληροφορικής (προγραμματιστικές δομές) και υλοποιείται στο εργαστήριο δεξιοτήτων.

Σκοπός:

Οι Μαθητές/τριες να διερευνήσουν και να καταλήξουν μέσα από μελέτη, έρευνα, δοκιμές και παρατήρηση στη δημιουργία ενός απλοϊκού οχήματος και την κίνησή του η οποία διακόπτεται σε οποιοδήποτε εμπόδιο. Η δημιουργική σύνθεση/ένταξη κινητήρων και του αισθητήρα στην παθητική κατασκευή και η κατάλληλη χρήση δομών επανάληψης και συγκρίσεων, ώστε με τον κατάλληλο προγραμματισμό να σταματά στα εμπόδια.

Στόχοι

- **Επισημαίνω** Τη χρησιμότητα αξιοποίησης του αισθητήρα εμποδίων και των κινητήρων (IoT) για την για την κίνηση οχημάτων..
- **Αναγνωρίζω** Τις συνθήκες για την ομαλή κίνηση το "ηλεκτρικού" οχήματος, καθώς και τα δομικά υλικά (παθητικά και ενεργά) που απαιτούνται για την κατασκευή αυτή.
- **Σχεδιάζω** Το παραδοτέο και τα τμήματα από τα οποία αποτελείται σχολιάζοντάς τα.
- **Αναλύω** Το έργο σε μικρότερες επιμέρους αυτοτελείς ενότητες με τις οποίες θα ασχοληθώ τμηματικά.
- **Συναρμολογώ** Τα δομικά υλικά πρώτα τα παθητικά και στη συνέχεια τα ενεργά.
- **Συνδυάζω** Με τον καλύτερο (σύντομο και έξυπνο) τρόπο τα πλακίδια εντολών που απαιτούνται ώστε να θέσω σε κίνηση υπό συνθήκες το όχημά μου.
- **Σχολιάζω** τη μέθοδο που ακολούθησα ώστε να φτάσω στην ολοκλήρωση του τελικού μου παραδοτέου με βάση το έργο που μου έχει ανατεθεί
- **Προβληματίζομαι** Σε σχέση με την κίνηση του οχήματος και των αλλαγών που χρειάζεται να κάνω στην κατασκευή αλλά και το πρόγραμμά μου, ώστε να γίνει πιο ομαλή και πιο ταχεία.
- **Ανταποκρίνομαι** Στο ρόλο που μου έχει ανατεθεί με τον καλύτερο δυνατό τρόπο.
- **Κρίνω** Το αποτέλεσμα που παρουσιάζω και τον τρόπο εργασίας της ομάδας, σε σχέση με την αρχική εισήγηση και ανάθεση του έργου.

Χρόνος μάθησης

2 ώρες και 40 λεπτά.

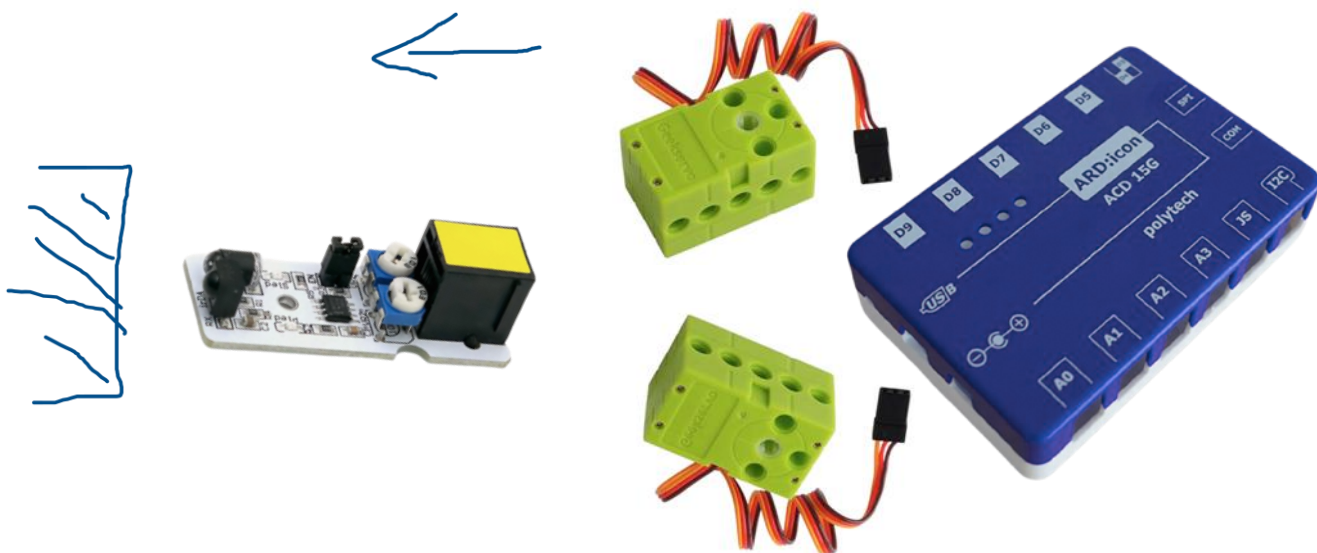
Μέγεθος τάξης

14 Μαθητές/τριες.



Β. Σενάριο:

Σχεδιάζουμε και υλοποιούμε ένα αυτοκινούμενο όχημα το οποίο προγραμματίζουμε να κινείται εμπρός μέχρι να συναντήσει ένα εμπόδιο.



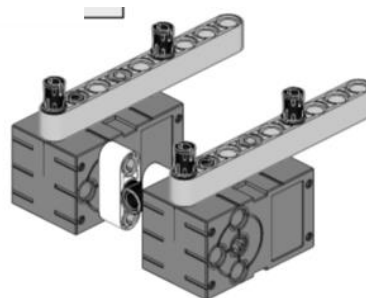
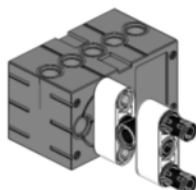
Γ. Υλικά:

- Αισθητήρας αποφυγής εμποδίου υπερύθρων DJS17
- Κινητήρας 360 Degree Continuous Rotation
- Παθητικά στοιχεία – Δομικά υλικά όπως: Τροχοί, άξονες, δοκοί 7m, μπίλια με τη βάση της, μαύρους και μπλε συνδέσμους, κ.λπ

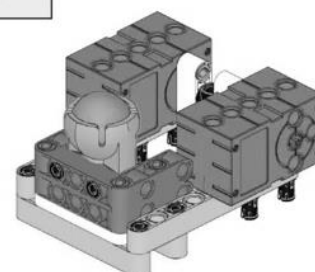
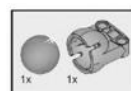
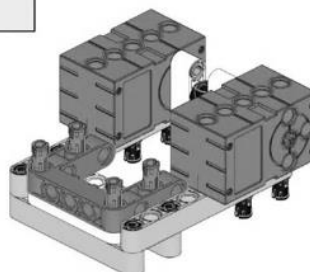
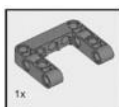
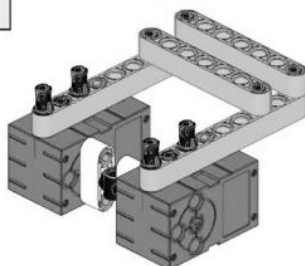


Δ. Συναρμολόγηση (παθητικών δομικών υλικών):

1



2

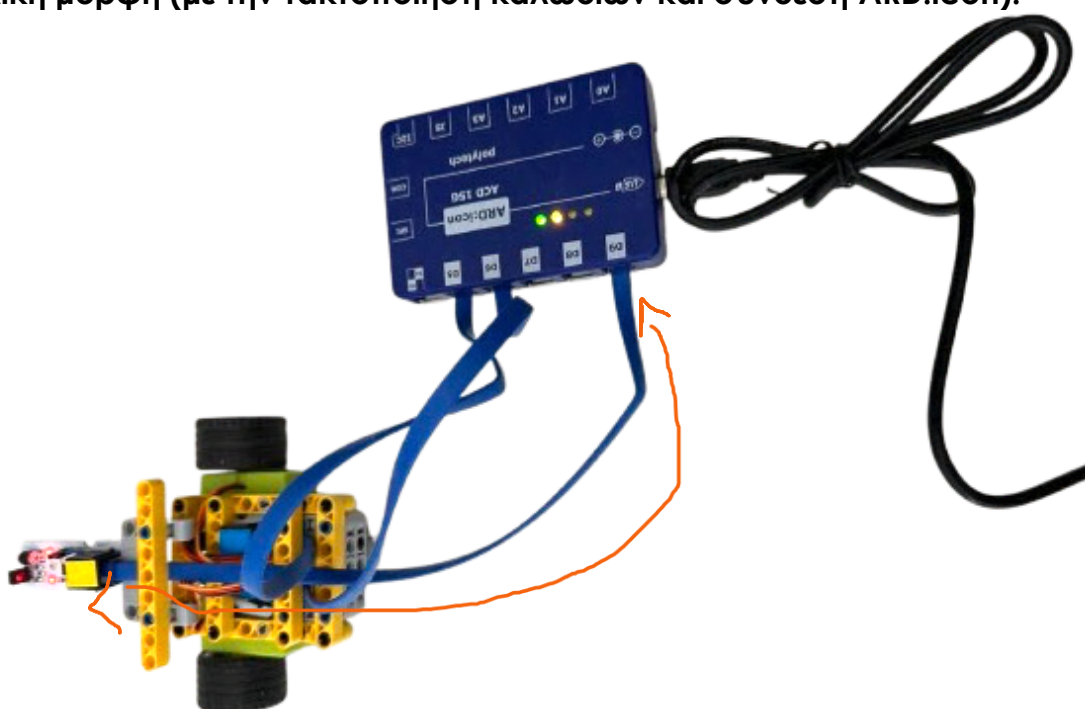


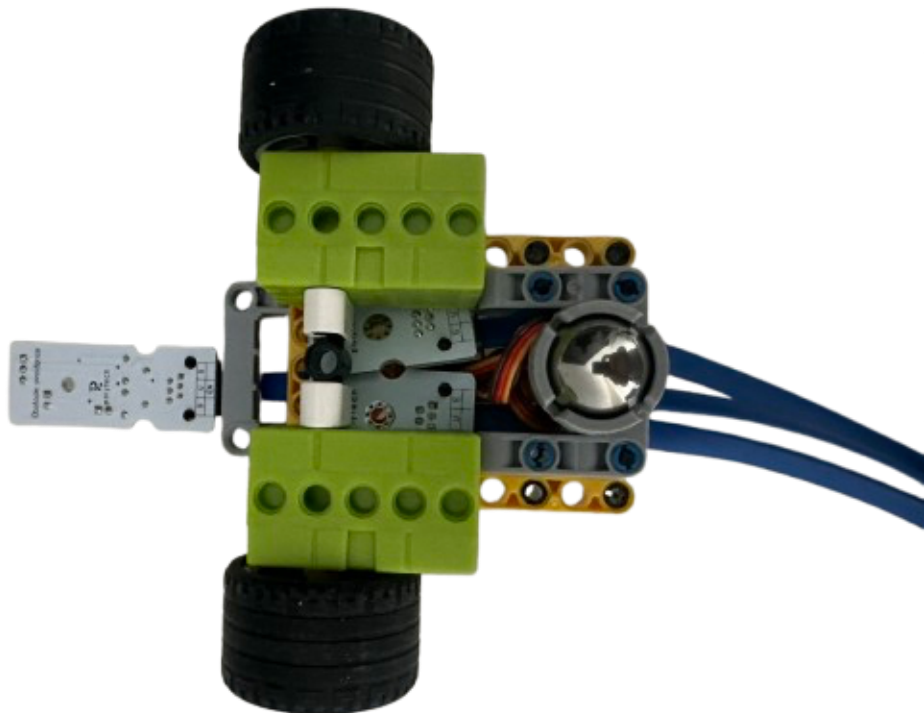
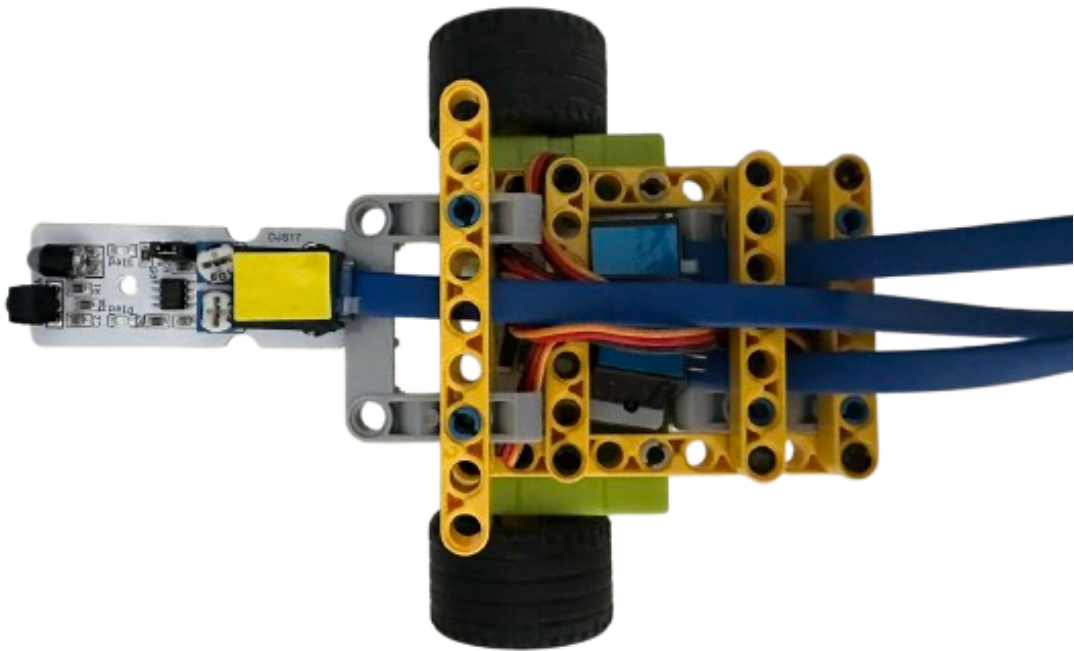
3

Μένει να βρούμε τρόπο στήριξης του ADR:icon και των μπαταριών του πάνω στο όχημα που ετοιμάσαμε...

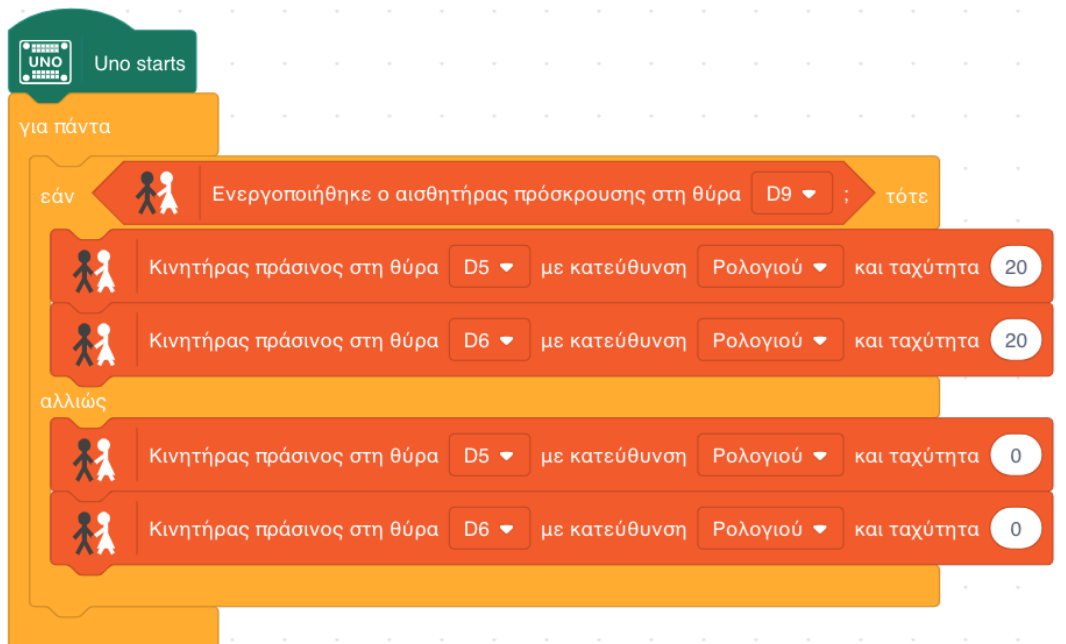
4

Τελική μορφή (με την τακτοποίηση καλωδίων και σύνδεση ADR:icon).





Ε. Προγραμματισμός



Ο κώδικας και Arduino C

```

/#!/
* MindPlus
* uno
*
*/
#include <DFRobot_Servo.h>
// Create an object
Servo servo_5;
Servo servo_6;

// Main program start
void setup() {
  pinMode(9, INPUT);
  servo_5.attach(5);
  servo_6.attach(6);
}
void loop() {
  if (digitalRead(9)==HIGH) {
    servo_5.angle(90+20);
    servo_6.angle(90+20);
  }
  else {
    servo_5.angle(90+0);
    servo_6.angle(90+0);
  }
}

```



4η δραστηριότητα: Όχημα με ρυθμιστή ταχύτητας που αντιδρά στα εμπόδια.

A. Στοιχεία Δραστηριότητας

Σενάριο βιωματικής εμπλοκής μαθητών για τη κατασκευή και τη διερεύνηση των παραγόντων που καθιστούν μια ρομποτική κατασκευή ικανή να κινείται και να σταματά σε εμπόδια. Την αξιοποίηση IoT (αισθητήρων), των αυτοματισμών που απαιτούνται, και την ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον. Αναφέρεται σε μαθητές Ε' Δημοτικού ή/και Α' Γυμνασίου στο Μάθημα Πληροφορικής (προγραμματιστικές δομές) και υλοποιείται στο εργαστήριο δεξιοτήτων.

Σκοπός:

Οι Μαθητές/τριες να διερευνήσουν και να καταλήξουν μέσα από μελέτη, έρευνα, δοκιμές και παρατήρηση στη δημιουργία ενός απλοϊκού οχήματος και την κίνησή του η οποία διακόπτεται σε οποιοδήποτε εμπόδιο. Η δημιουργική σύνθεση/ένταξη κινητήρων και του αισθητήρα στην παθητική κατασκευή και η κατάλληλη χρήση δομών επανάληψης και συγκρίσεων, ώστε με τον κατάλληλο προγραμματισμό να σταματά στα εμπόδια.

Στόχοι

- **Επισημαίνω** Τη χρησιμότητα αξιοποίησης του αισθητήρα απόστασης και των κινητήρων (IoT) για την κίνηση οχημάτων..
- **Αναγνωρίζω** Τις συνθήκες για την ομαλή κίνηση το "ηλεκτρικού" οχήματος, καθώς και τα δομικά υλικά (παθητικά και ενεργά) που απαιτούνται για την κατασκευή αυτή.
- **Σχεδιάζω** Το παραδοτέο και τα τμήματα από τα οποία αποτελείται σχολιάζοντάς τα.
- **Αναλύω** Το έργο σε μικρότερες επιμέρους αυτοτελείς ενότητες με τις οποίες θα ασχοληθώ τμηματικά.
- **Συναρμολογώ** Τα δομικά υλικά πρώτα τα παθητικά και στη συνέχεια τα ενεργά.
- **Συνδυάζω** Με τον καλύτερο (σύντομο και έξυπνο) τρόπο τα πλακίδια εντολών που απαιτούνται ώστε να θέσω σε κίνηση υπό συνθήκες το όχημά μου.
- **Σχολιάζω** τη μέθοδο που ακολούθησα ώστε να φτάσω στην ολοκλήρωση του τελικού μου παραδοτέου με βάση το έργο που μου έχει ανατεθεί
- **Προβληματίζομαι** Σε σχέση με την κίνηση του οχήματος και των αλλαγών που χρειάζεται να κάνω στην κατασκευή αλλά και το πρόγραμμά μου, ώστε να γίνει πιο ομαλή και πιο ταχεία.
- **Κρίνω** Το αποτέλεσμα που παρουσιάζω και τον τρόπο εργασίας της ομάδας, σε σχέση με την αρχική εισήγηση και ανάθεση του έργου.

Χρόνος μάθησης

2 ώρες και 40 λεπτά.

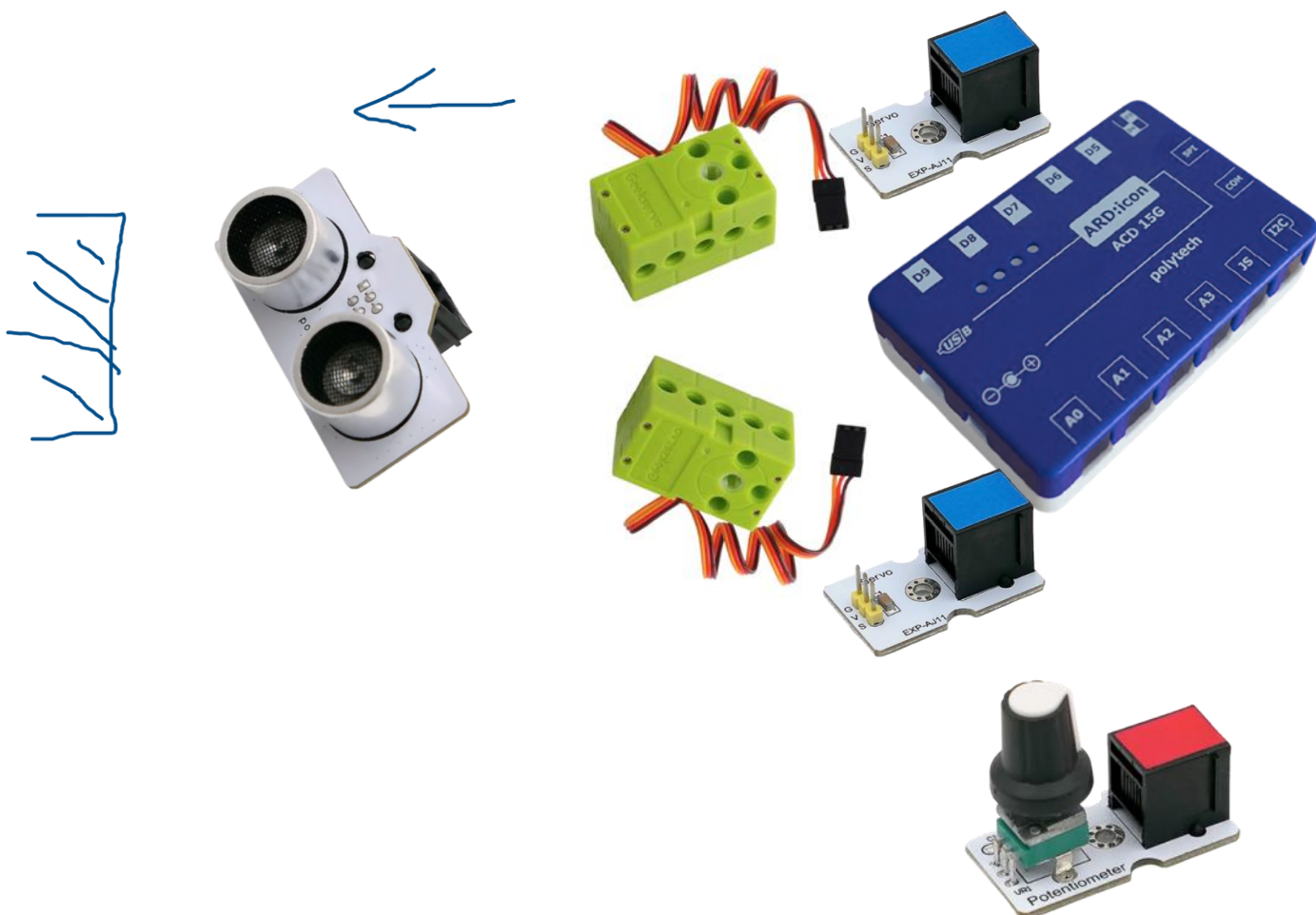
Μέγεθος τάξης

14 Μαθητές/τριες.



Β. Σενάριο:

Σχεδιάζουμε και υλοποιούμε ένα αυτοκινούμενο όχημα το οποίο προγραμματίζουμε να κινείτε εμπρός μέχρι να συναντήσει ένα εμπόδιο σε συγκεκριμένα εκατοστά.

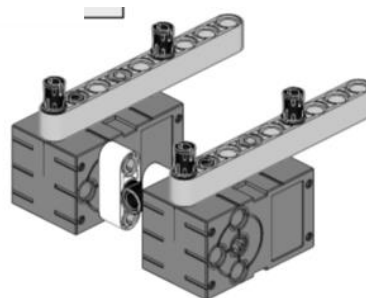
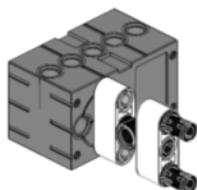
**Γ. Υλικά:**

- Αισθητήρας απόστασης (Sonar)
- Ποτενσιόμετρο
- Adaptors από SVG σε mini lack (για κινητήρες)
- Κινητήρας 360 Degree Continuous Rotation
- Παθητικά στοιχεία – Δομικά υλικά όπως: Τροχοί, άξονες, δοκοί 7m, μπίλια με τη βάση της, μαύρους και μπλε συνδέσμους, κ.λπ

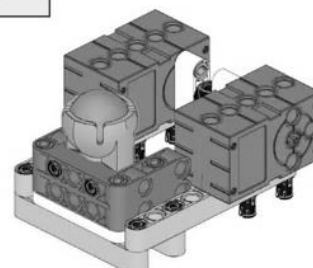
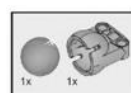
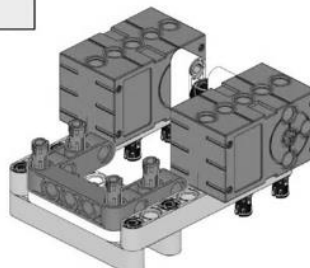
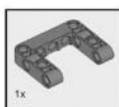
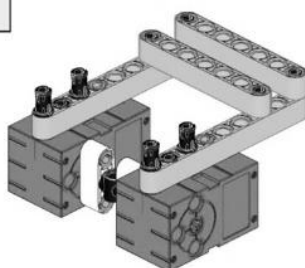
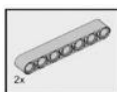


Δ. Συναρμολόγηση (παθητικών δομικών υλικών):

1

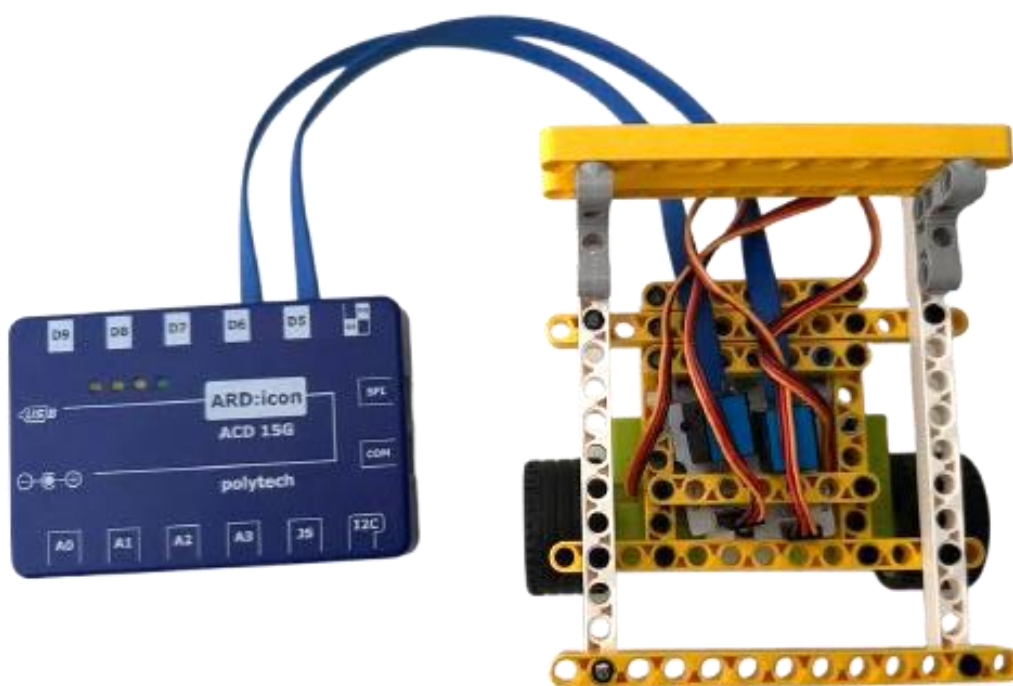


2



3

Μένει να βρούμε τρόπο στήριξης του ADR:icon και των μπαταριών του πάνω στο όχημα που ετοιμάσαμε...



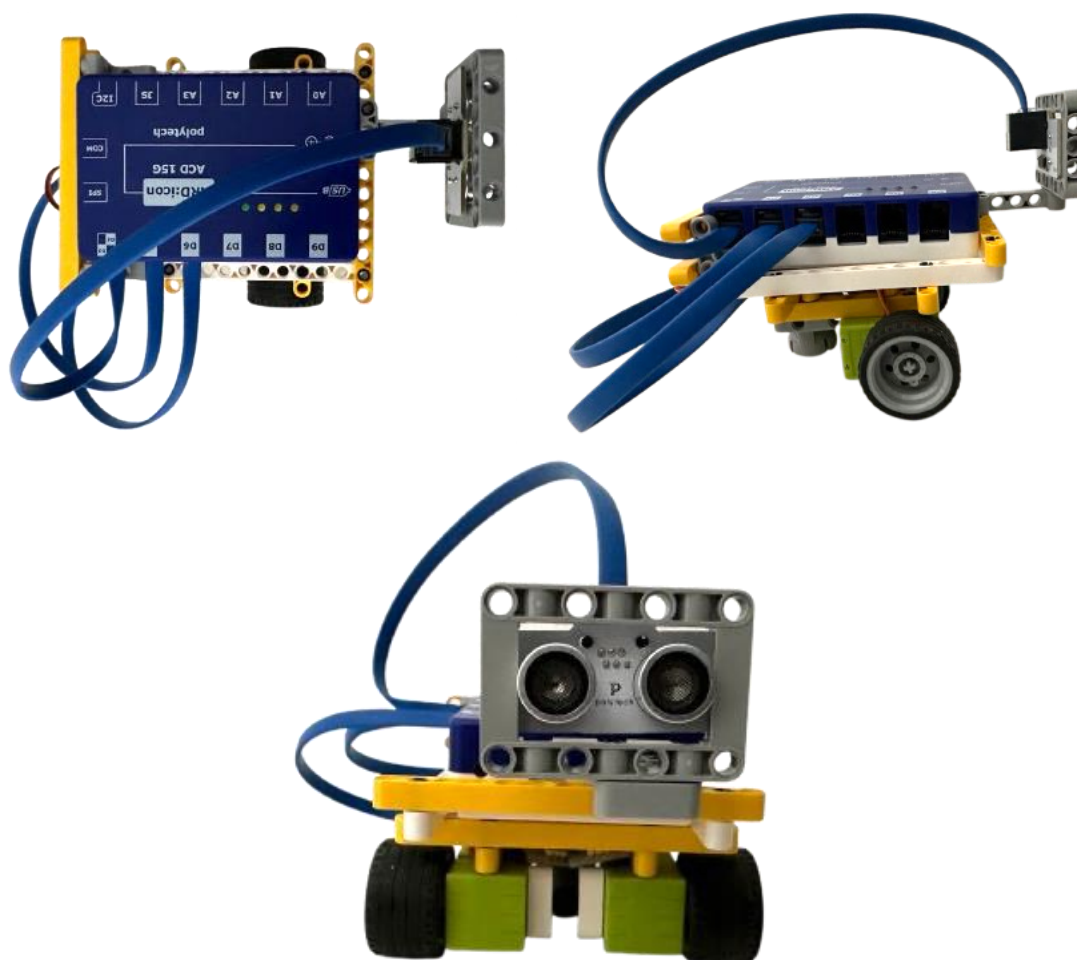
4

Τελική μορφή (με την τακτοποίηση καλωδίων και σύνδεση ARD:icon).



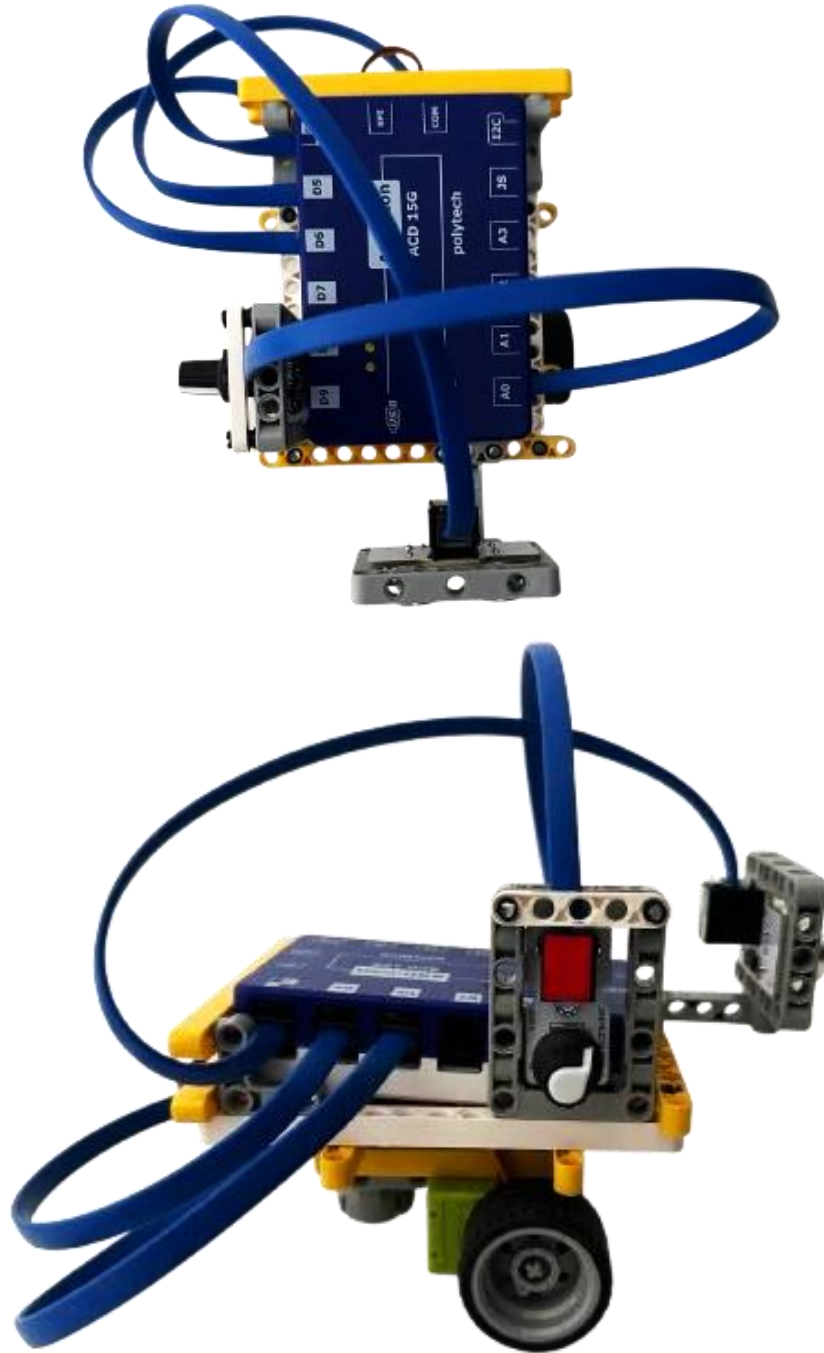
5

Με την προσθήκη του αισθητήρα απόστασης



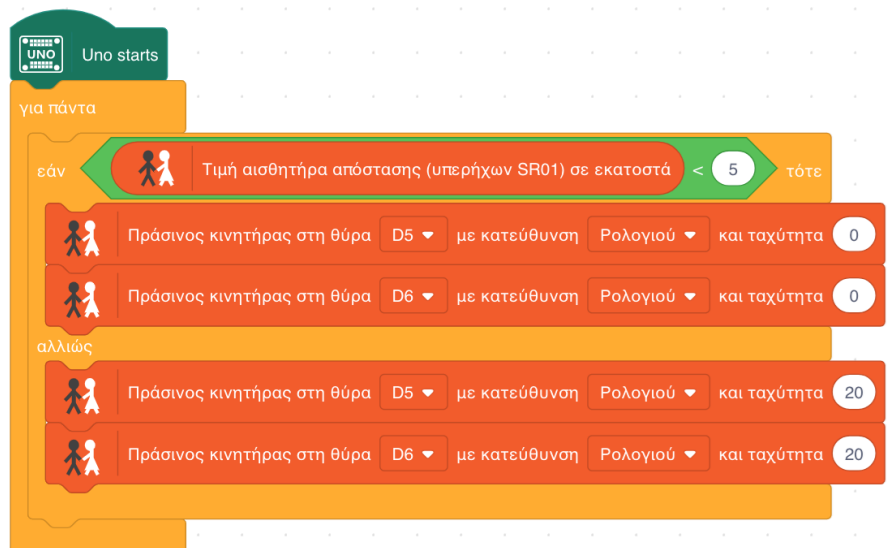
6

Με την προσθήκη του ποτενσιόμετρου (ρυθμιστής ταχύτητας)



Ε. Κώδικας

Κίνηση εμπρός μέχρι την αναγνώριση εμποδίου στα πέντε εκατοστά και σταμάτημα.



Κίνηση με ρυθμιζόμενη ταχύτητα και σταμάτημα σε εμπόδιο στα 5 εκατοστά.

Θα χρειαστούμε μια μεταβλητή για την ταχύτητα που θα επιστρέφει το ποτενσιόμετρο
 Και μια πολύ δυνατή λειτουργία η οποία αντιστοιχεί τις τιμές δύο διαφορετικών λιστών τιμών. Αυτή του ποτενσιόμετρου 0-1023 και αυτή της ταχύτητας περιστροφής του κινητήρα 0-90



Και ο κώδικας σε Arduino C

```
*!
* MindPlus
* uno
*
*/
#include <SR01.h>
#include <DFRobot_Servo.h>

// Dynamic variables
volatile float mind_n__;
// Create an object
Servo servo_5;
Servo servo_6;

// Main program start
void setup() {
    pinMode(0, INPUT);
    pinMode(3, OUTPUT);
    pinMode(4, INPUT);
    servo_5.attach(5);
    servo_6.attach(6);
}
void loop() {
    mind_n__ = (map(analogRead(0), 0, 1023, 0, 90));
    if ((getDistance(3,4)<5)) {
        servo_5.angle(90+0);
        servo_6.angle(90+0);
    }
    else {
        servo_5.angle(90+mind_n__);
        servo_6.angle(90+mind_n__);
    }
}
```



5^η δραστηριότητα: Μετρητής υγρασίας

A. Στοιχεία Δραστηριότητας

Σενάριο εμπλοκής μαθητών για την κατασκευή μιας συσκευής μέτρησης της υγρασίας σε μια καλλιέργεια, με την αξιοποίηση IoT (αισθητήρων, LCD οθόνης, LED), που απαιτούνται, και την ανάπτυξη εφαρμογών σε προγραμματιστικό περιβάλλον. Αναφέρεται σε μαθητές Ε' Δημοτικού ή/και Α' Γυμνασίου στο Μάθημα Πληροφορικής, (προγραμματιστικές δομές) Τεχνολογίας και υλοποιείται και στο εργαστήριο δεξιοτήτων.

Σκοπός:

Οι Μαθητές/τριες να διερευνήσουν και να καταλήξουν μέσα από μελέτη, έρευνα, δοκιμές και παρατήρηση στη δημιουργία ενός απλοϊκού μετρητή υγρασίας που θα μπορούσε να μας ειδοποιεί με φωτεινή ένδειξη για την ανάγκη ποτίσματος

Στόχοι

- **Επισημαίνω** Τη χρησιμότητα αξιοποίησης του αισθητήρα υγρασίας της οθόνης LCD και των LEDs
- **Αναγνωρίζω** Τις συνθήκες για την πρόκληση ανάματος του προειδοποιητικού LED.
- **Σχεδιάζω** Το παραδοτέο και τα τμήματα από τα οποία αποτελείται σχολιάζοντάς τα.
- **Αναλύω** Τις ενδείξεις του αισθητήρα υγρασίας μετά την προβολή τους στην LCD οθόνη.
- **Συναρμολογώ** Τα ενεργά στοιχεία.
- **Συνδυάζω** Με τον καλύτερο (σύντομο και έξυπνο) τρόπο τα πλακίδια εντολών που απαιτούνται ώστε να έχω το επιθυμητό αποτέλεσμα μετρήσεων και προειδοποίηση για πότισμα.
- **Σχολιάζω** τη μέθοδο που ακολούθησα ώστε να φτάσω στην ολοκλήρωση του τελικού μου παραδοτέου με βάση το έργο που μου έχει ανατεθεί
- **Ανταποκρίνομαι** Στο ρόλο που μου έχει ανατεθεί με τον καλύτερο δυνατό τρόπο.
- **Κρίνω** Το αποτέλεσμα που παρουσιάζω και τον τρόπο εργασίας της ομάδας, σε σχέση με την αρχική εισήγηση και ανάθεση του έργου.

Χρόνος μάθησης

1 ώρα και 10 λεπτά.

Μέγεθος τάξης

14 Μαθητές/τριες.



Β. Σενάριο:

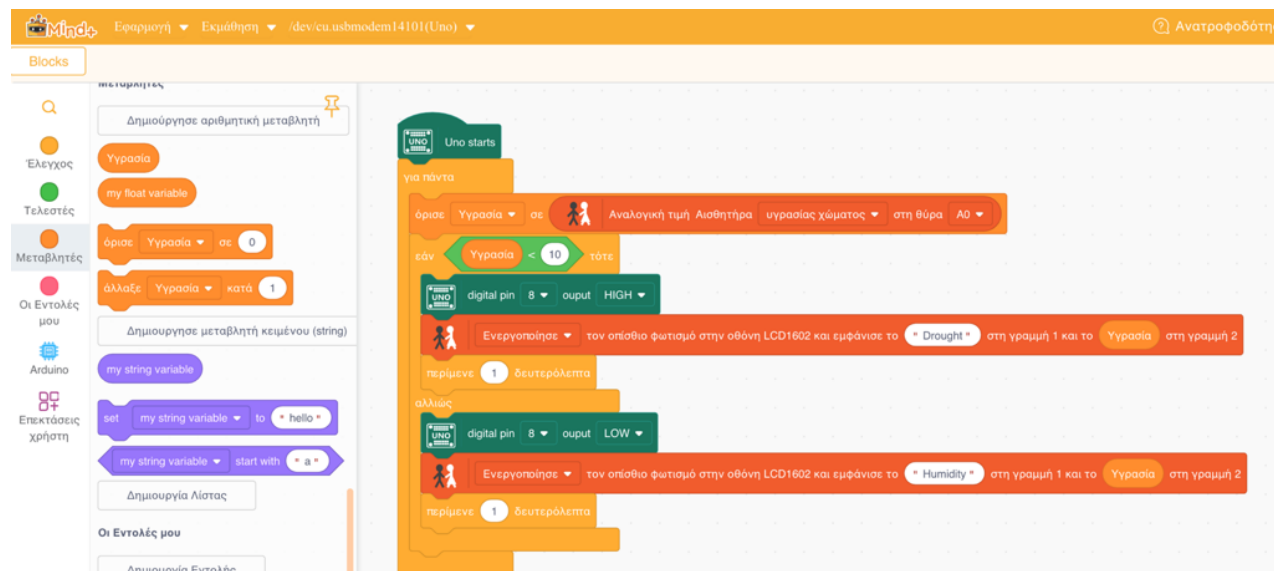
Σχεδιάζουμε και υλοποιούμε μια συσκευή που μας δείχνει την υγρασία μιας καλιέργειας και μας ειδοποιεί για πότισμα με φωτεινή ένδειξη.

Γ. Υλικά



Στ. Προγραμματισμός

Ο κώδικας με τη δημιουργία μιας μεταβλητής, και μια σύγκριση της τιμής που επιστρέφει ο αισθητήρας υγρασίας...



Ο Κώδικας σε Arduino C

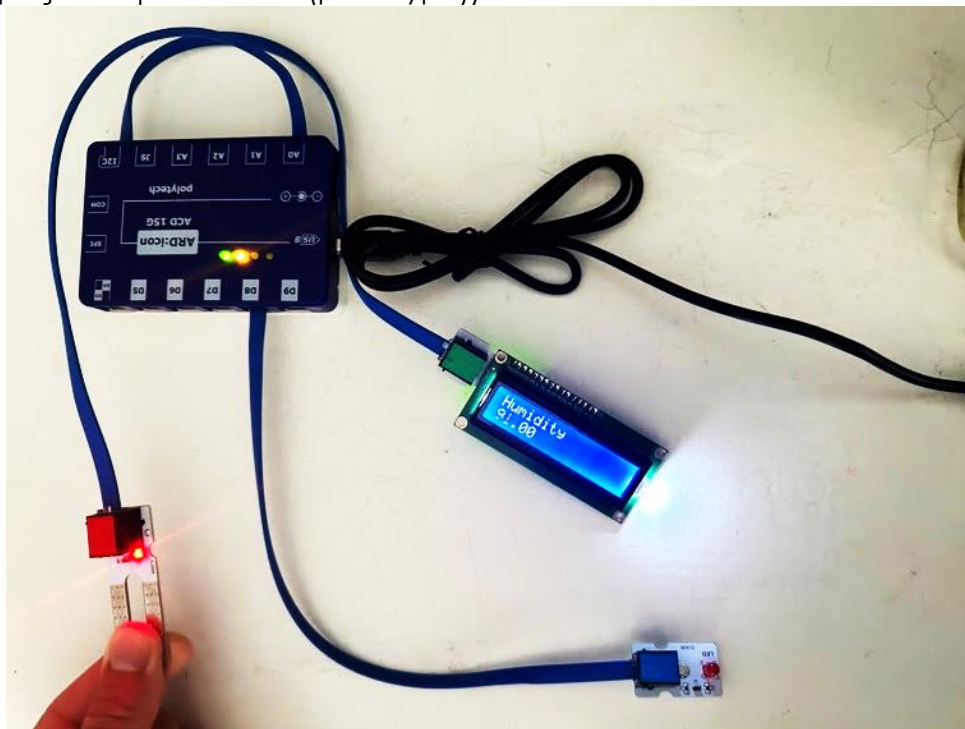
```
/*!  
 * MindPlus  
 * uno  
 *  
 */  
#include <LCD_I2C.h>  
  
// Dynamic variables  
volatile float mind_n__;  
// Create an object  
LCD_I2C lcd(0x27, 16, 2);  
  
// Main program start  
void setup() {  
    pinMode(0, INPUT);  
    lcd.begin();  
}  
void loop() {  
    mind_n__ = analogRead(0);  
    if ((mind_n__ < 10)) {  
        digitalWrite(8, HIGH);  
        lcd.clear();  
        lcd.backlight();  
        lcd.setCursor(0, 0);  
        lcd.print("Drought");  
        lcd.setCursor(0, 1);  
        lcd.print(mind_n__);  
        delay(1000);  
    }  
    else {  
        digitalWrite(8, LOW);  
        lcd.clear();  
        lcd.backlight();  
        lcd.setCursor(0, 0);  
        lcd.print("Humidity");  
        lcd.setCursor(0, 1);  
        lcd.print(mind_n__);  
        delay(1000);  
    }  
}
```



Ζ. Το αποτέλεσμα

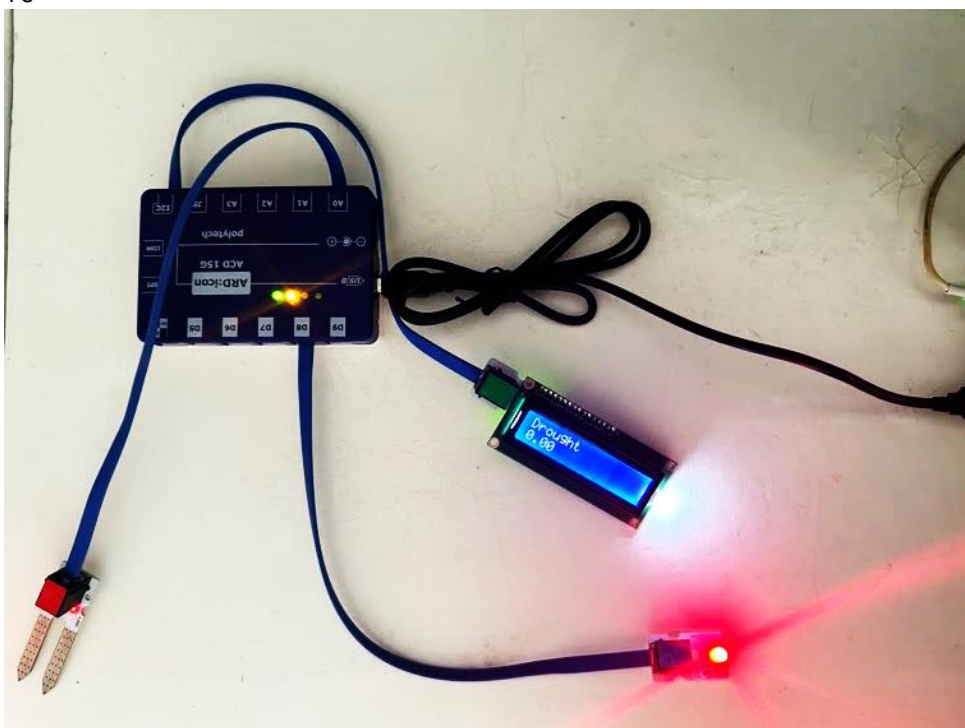
Με υγρασία.

Το κόκκινο LED είναι σβηστό γιατί η ένδειξη του αισθητήρα υγρασίας είναι μεγαλύτερη από 10 (υγρασία χεριού που ακουμπά τον αισθητήρα για τις ανάγκες του πρωτότυπου (prototype))



Χωρίς υγρασία.

Το κόκκινο LED ανάβει γιατί η ένδειξη του αισθητήρα υγρασίας είναι μικρότερη από 10



6^η δραστηριότητα: Ρομποτικός βραχίονας (δαγκάνα)

A. Στοιχεία Δραστηριότητας

Οι Μαθητές/τριες θα σχεδιάσουν και θα υλοποιήσουν ρομποτικό βραχίονα συλλογής και μετακίνησης αντικειμένων. Θα αξιοποιήσουν παθητικά και ενεργά στοιχεία και ιδιαίτερα τα γρανάζια ώστε να επιτύχουν την κίνηση των στελεχών του συλλέκτη αντικειμένων (δαγκάνα), χρησιμοποιώντας και τον δυνατό κινητήρα με τον αντίστοιχο προγραμματισμό των κινήσεων του βραχίονα ως αποτέλεσμα του πατήματος κουμπιού.

Σκοπός

Η ενίσχυση της κατανόησης των Μαθητών/τριών για τις βασικές αρχές της ρομποτικής, ενθαρρύνοντας τη δημιουργικότητα, τον προγραμματισμό, και τη συνεργατική εργασία, αποτελούν τους διδακτικούς στόχους της δραστηριότητας αυτής. Πιο συγκεκριμένα η κατανόηση βασικών αρχών της ρομποτικής, όπως η κίνηση, η αντίληψη, και ο έλεγχος των κατασκευών. Η εξοικείωση με τη χρήση των δομικών στοιχείων τύπου LEGO για τη δημιουργία του μηχανισμού και η εκμάθηση των βασικών δομών και εντολών της πλατφόρμας προγραμματισμού του ARD:icon – Mind+ για τον έλεγχο της κατασκευής. Η σχεδίαση του βραχίονα με βάση συγκεκριμένα κριτήρια, όπως, ισορροπία και αριθμός κινούμενων μερών και ο προγραμματισμός του για τον έλεγχο του βραχίονα. Η αντιμετώπιση προβλημάτων κατά τη δημιουργία, τον προγραμματισμό και τη λειτουργία του βραχίονα, μέσα από την ομαλή συνεργασία και επικοινωνία με άλλους μαθητές για την επίτευξη κοινών στόχων. Τέλος η κοινή αξιολόγηση που θα οδηγήσει στην επιλογή των βέλτιστων λύσεων για το θέμα.

Στόχοι

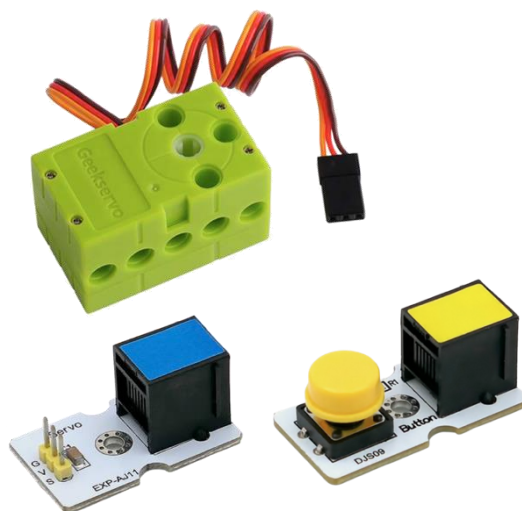
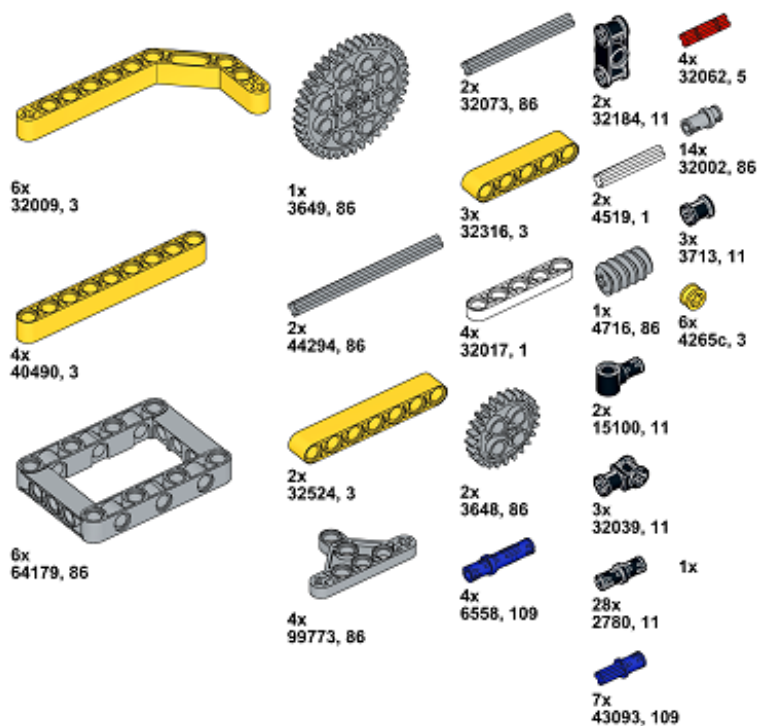
- **Ορίζω (Γνώση):** Τα θέματα της ισορροπίας των στελεχών που απαιτούνται για τον ρομποτικό βραχίονα (δαγκάνα).
- **Επιλέγω (Γνώση):** Τα κατάλληλα παθητικά και ενεργά στοιχεία που θα χρειαστώ για την κατασκευή μου.
- **Αναγνωρίζω (Γνώση):** Την αξιοποίηση του ρομποτικού βραχίονα στη βιομηχανία ή/και την ιατρική.
- **Περιγράφω λόγους για (Κατανόηση):** τους οποίους είναι σημαντική η χρήση του ρομποτικού βραχίονα στην βιομηχανία και την ιατρική.
- **Συναρμολογώ (Εφαρμογή):** Τα παθητικά και τα ενεργά στοιχεία που απαιτούνται με γνώμονα την ισορροπία και την σταθερότητα της κατασκευής μου.
- **Αναφέρω συστατικά μέρη (Ανάλυση):** του υπό κατασκευή βραχίονα.
- **Καταρτίζω (Σύνθεση):** Σχέδιο για την αξιοποίηση των πλήκτρων κίνησης των στελεχών του βραχίονα.
- **Προβλέπω (Ανάλυση):** Τον μέγεθος των στελεχών της κατασκευής, σύμφωνα πάντα με την αρχικό σχέδιο.
- **Προβληματίζομαι (Αξιολόγηση):** Για την όσο το δυνατόν καλύτερη αξιοποίηση των επιλογών έναρξης των κινήσεων του κάθε στελέχους της κατασκευής μου.
- **Σχεδιάζω (Ψυχοκινητικές δεξιότητες):** Τρόπους μεταφοράς της κίνησης από κάθετη σε οριζόντια περιστροφή αξόνων.
- **Τροποποιώ (Εφαρμογή):** Την κατασκευή μου ώστε να είναι όσο το δυνατόν πιο αποτελεσματική, σύμφωνα με τον αρχικό στόχο.
- **Αξιολογώ (Αξιολόγηση):** Τον τρόπο εφαρμογής και το αποτέλεσμα της λειτουργίας του βραχίονα.

Χρόνος μάθησης
2 ώρες.

Μέγεθος τάξης
14 Μαθητές/τριες.

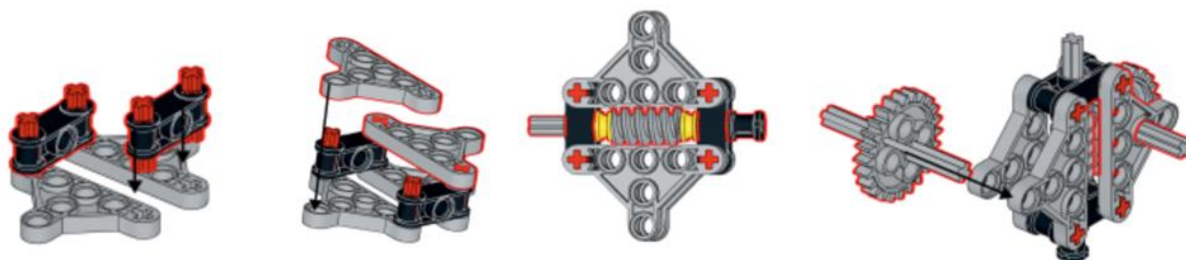


Β. Δομικά Υλικά

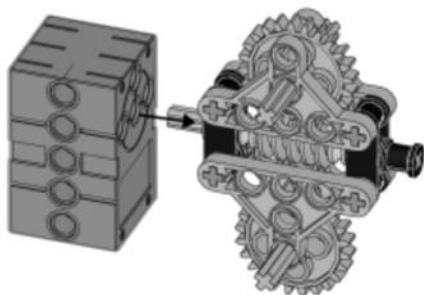


Γ. Τα στάδια της κατασκευής

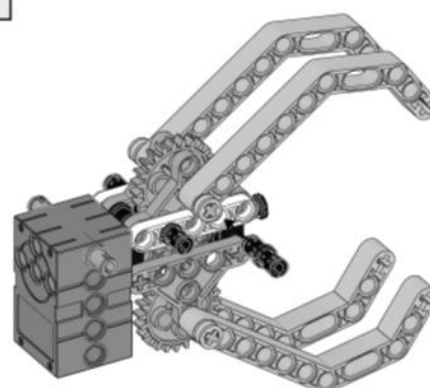
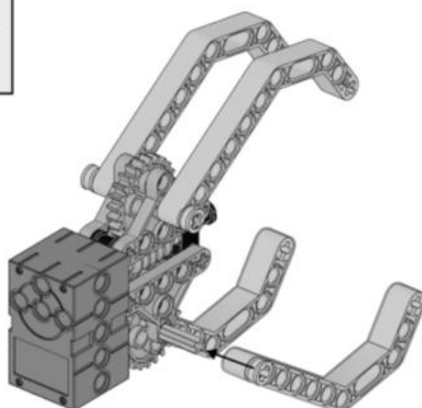
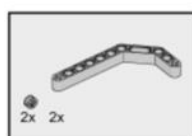
1



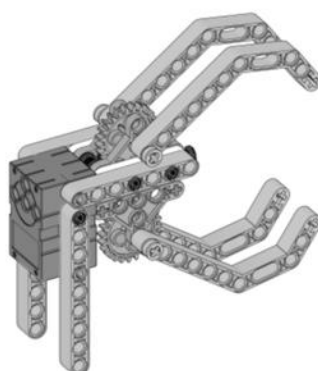
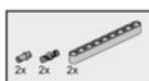
2



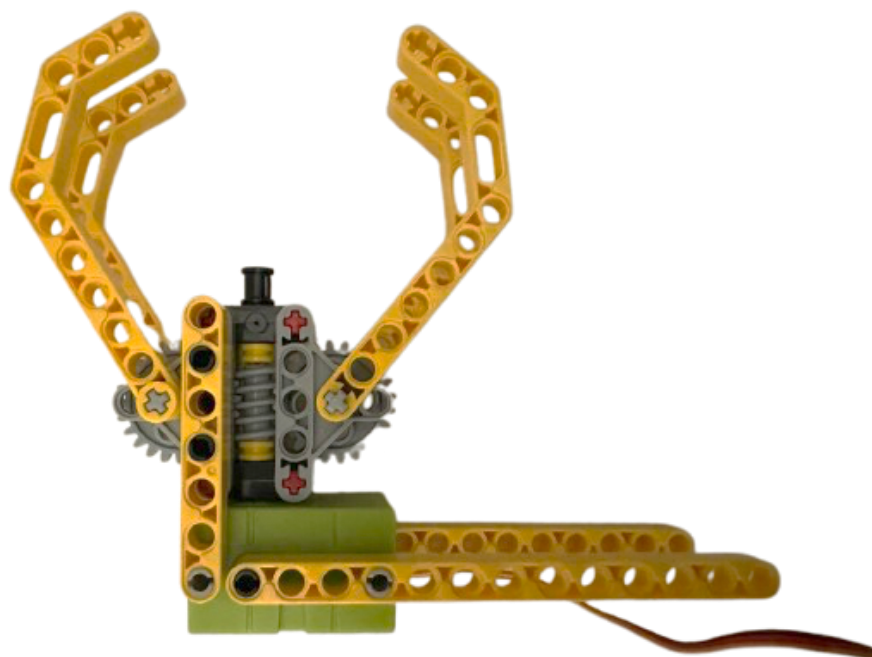
3



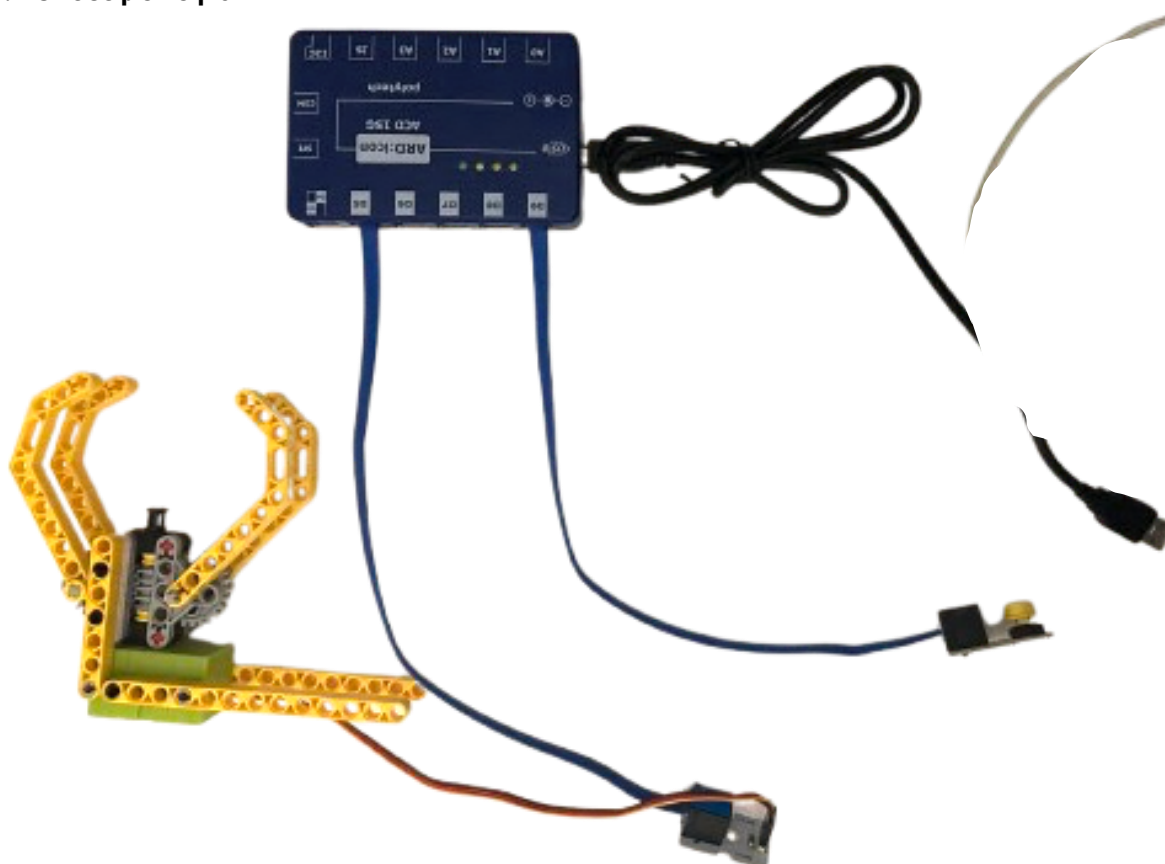
4



Η δαγκάνα με τα δομικά υλικά και τον κινητήρα



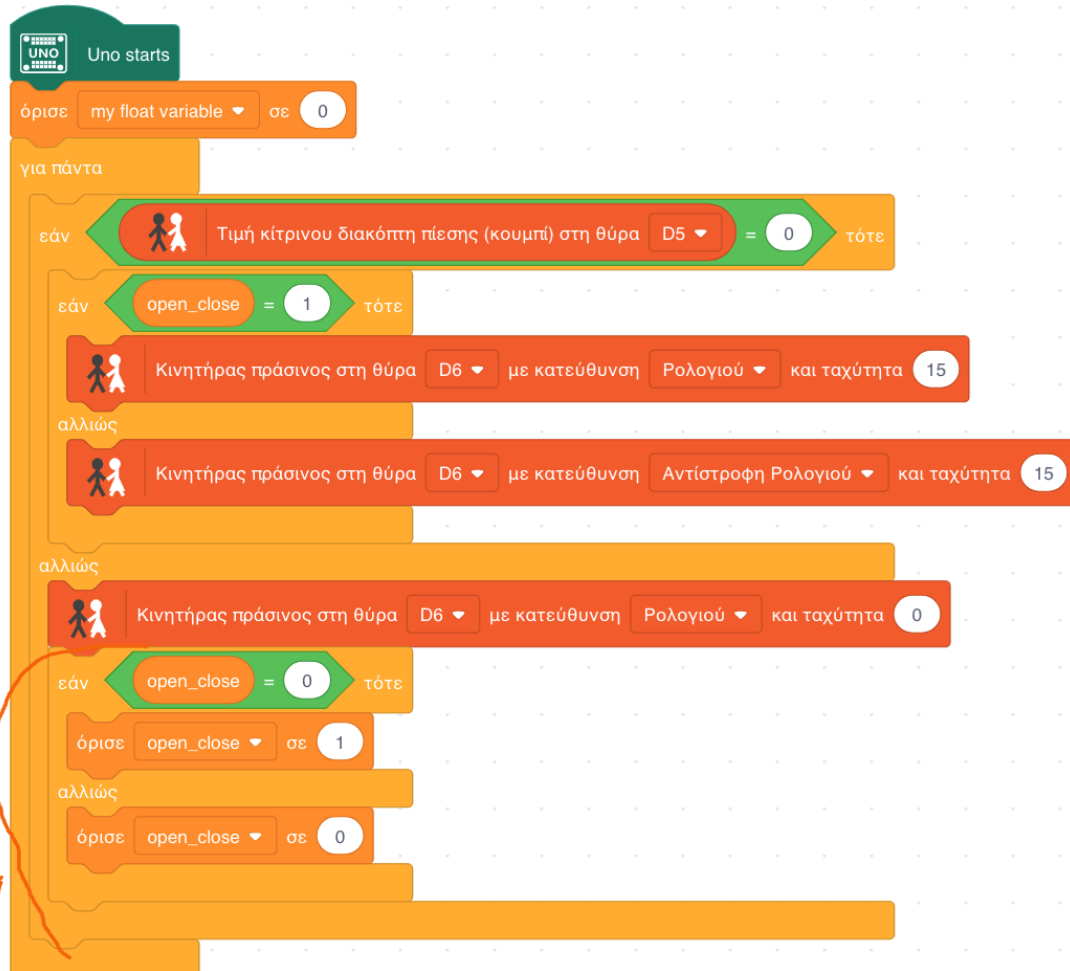
Δ. Συνδεσμολογία



Ε. Κώδικας

Θα χρειαστούμε μια μεταβλητή για να αλλάζουμε την κατεύθυνση περιστροφής του άξονα του κινητήρα με το κουμπί.

Η μεταβλητή που δημιουργούμε είναι αριθμητική και την ονομάζουμε "open_close". Θα της δίνουμε τιμή 0 για άνοιγμα και ένα για κλείσιμο (δεξιά ή αριστερή περιστροφή του άξονα)



Στο αλλιώς της αρχικής σύγκρισης:

Όταν το κουμπί πίεσης (κίτρινο button) δεν πατιέται, τότε αλλάζουμε τιμή στη μεταβλητή, ώστε όταν ξαναπατηθεί να αλλάξει η κατεύθυνση περιστροφής του άξονα και συνεπώς να ανοίξει ή να κλείσει η δαγκάνα.



ΣΤ. Κώδικας σε Arduino C

```

/!*
 * MindPlus
 * uno
 *
 */
#include <DFRobot_Servo.h>

// Dynamic variables
volatile float mind_n_my_float_variable, mind_n_open_close;
// Create an object
Servo servo_6;

// Main program start
void setup() {
    pinMode(5, INPUT);
    servo_6.attach(6);
    mind_n_my_float_variable = 0;
}
void loop() {
    if ((digitalRead(5)==0)) {
        if ((mind_n_open_close==1)) {
            servo_6.angle(90+15);
        }
        else {
            servo_6.angle(90-15);
        }
    }
    else {
        servo_6.angle(90+0);
        if ((mind_n_open_close==0)) {
            mind_n_open_close = 1;
        }
        else {
            mind_n_open_close = 0;
        }
    }
}

```



7^η δραστηριότητα: Μπάρα χώρου στάθμευσης

A. Στοιχεία Δραστηριότητας

Οι Μαθητές/τριες θα σχεδιάσουν και θα υλοποιήσουν μία μπάρα χώρου στάθμευσης. Θα αξιοποιήσουν παθητικά και ενεργά στοιχεία και ιδιαίτερα τον servo κινητήρα ώστε να επιτύχουν την κίνηση της μπάρας, όταν πλησιάζει ένα όχημα

Σκοπός

Η ενίσχυση της κατανόησης των Μαθητών/τριών για τις βασικές αρχές της ρομποτικής, ενθαρρύνοντας τη δημιουργικότητα, τον προγραμματισμό, και τη συνεργατική εργασία, αποτελούν τους διδακτικούς στόχους της δραστηριότητας αυτής. Πιο συγκεκριμένα η κατανόηση βασικών αρχών της ρομποτικής, όπως η κίνηση, η αντίληψη, και ο έλεγχος των κατασκευών. Η εξοικείωση με τη χρήση του Servo κινητήρα και η εκμάθηση των βασικών δομών και εντολών της πλατφόρμας προγραμματισμού του ARD:icon – Mind+ για τον έλεγχο της κατασκευής. Η σχεδίαση της μπάρας με βάση συγκεκριμένα κριτήρια, όπως, ισορροπία, ο αριθμός κινούμενων μερών και ο προγραμματισμός για τον έλεγχο της κατασκευής. Η αντιμετώπιση προβλημάτων κατά τη δημιουργία, τον προγραμματισμό και τη λειτουργία της μπάρας, μέσα από την ομαλή συνεργασία και επικοινωνία με άλλους μαθητές για την επίτευξη κοινών στόχων. Τέλος η κοινή αξιολόγηση που θα οδηγήσει στην επιλογή των βέλτιστων λύσεων για το θέμα.

Στόχοι

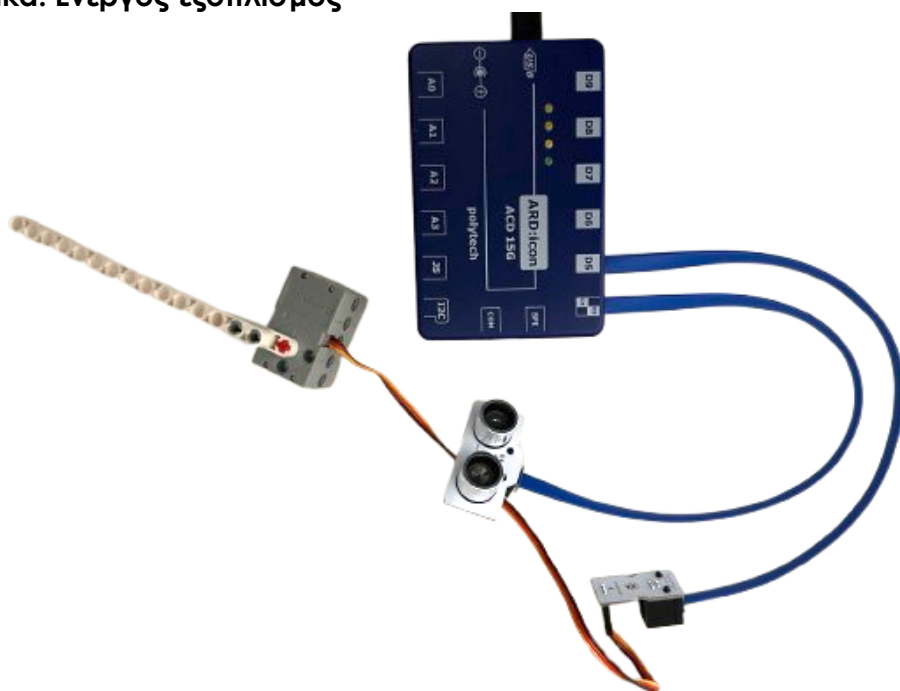
- **Ορίζω (Γνώση):** Τα θέματα της ισορροπίας των στελεχών που απαιτούνται για τον αυτοματισμό μας.
- **Επιλέγω (Γνώση):** Τα κατάλληλα παθητικά και ενεργά στοιχεία που θα χρειαστώ για την κατασκευή μου.
- **Αναγνωρίζω (Γνώση):** Την αξιοποίηση της μπάρας στους χώρους στάθμευσης για τις καθημερινές μας ανάγκες.
- **Περιγράφω λόγους για (Κατανόηση):** τους οποίους είναι σημαντική η χρήση της μπάρας στους χώρους στάθμευσης και στα διόδια των εθνικών οδών.
- **Συναρμολογώ (Εφαρμογή):** Τα παθητικά και τα ενεργά στοιχεία που απαιτούνται με γνώμονα την ισορροπία και την σταθερότητα της κατασκευής μου.
- **Αναφέρω συστατικά μέρη (Ανάλυση):** Της υπό κατασκευής μπάρας των χώρων στάθμευσης.
- **Καταρτίζω (Σύνθεση):** Σχέδιο για την λειτουργία και αξιοποίηση του αυτοματισμού μου
- **Προβλέπω (Ανάλυση):** Τον μέγεθος των στελεχών της κατασκευής, σύμφωνα πάντα με την αρχικό σχέδιο.
- **Προβληματίζομαι (Αξιολόγηση):** Για την όσο το δυνατόν καλύτερη αξιοποίηση των επιλογών έναρξης των κινήσεων του κάθε στελέχους της κατασκευής μου.
- **Τροποποιώ (Εφαρμογή):** Την κατασκευή μου ώστε να είναι όσο το δυνατόν πιο αποτελεσματική, σύμφωνα με τον αρχικό στόχο.
- **Αξιολογώ (Αξιολόγηση):** Τον τρόπο εφαρμογής και το αποτέλεσμα της λειτουργίας της μπάρας

Χρόνος μάθησης
2 ώρες.

Μέγεθος τάξης
14 Μαθητές/τριες.

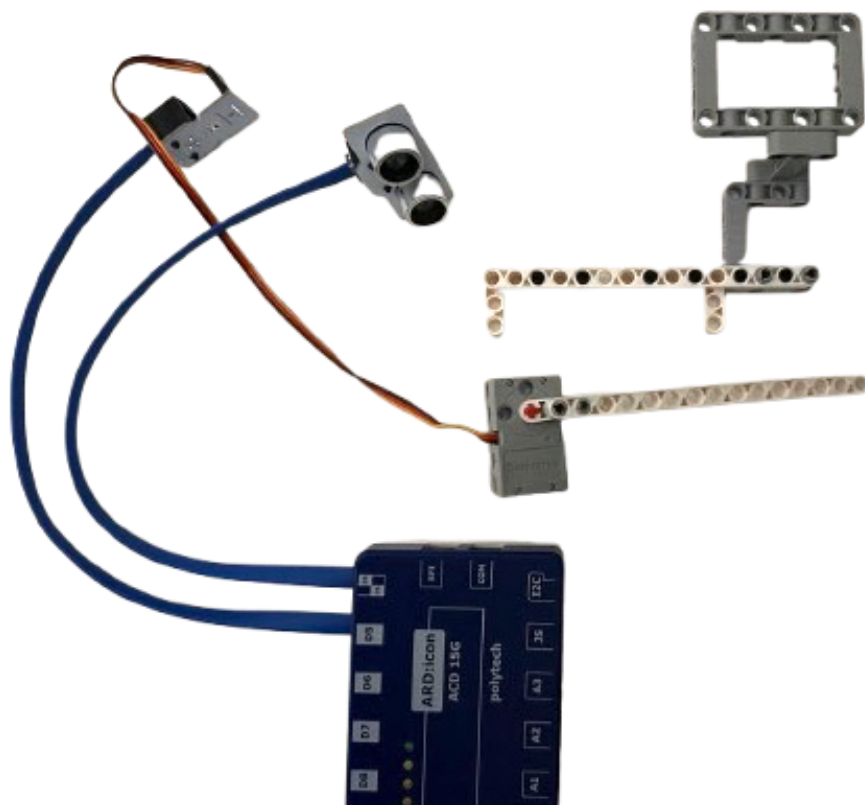


Β. Υλικά: Ενεργός εξοπλισμός



Θα χρησιμοποιήσουμε και τον γκρι Servo κινητήρα από το Nezha πακέτο

Γ. Συνδεσμολογία – Συναρμολόγηση δομικών υλικών



Δ. Κώδικας σε περιβάλλον Mind+



Ο κώδικας σε Arduino C

```

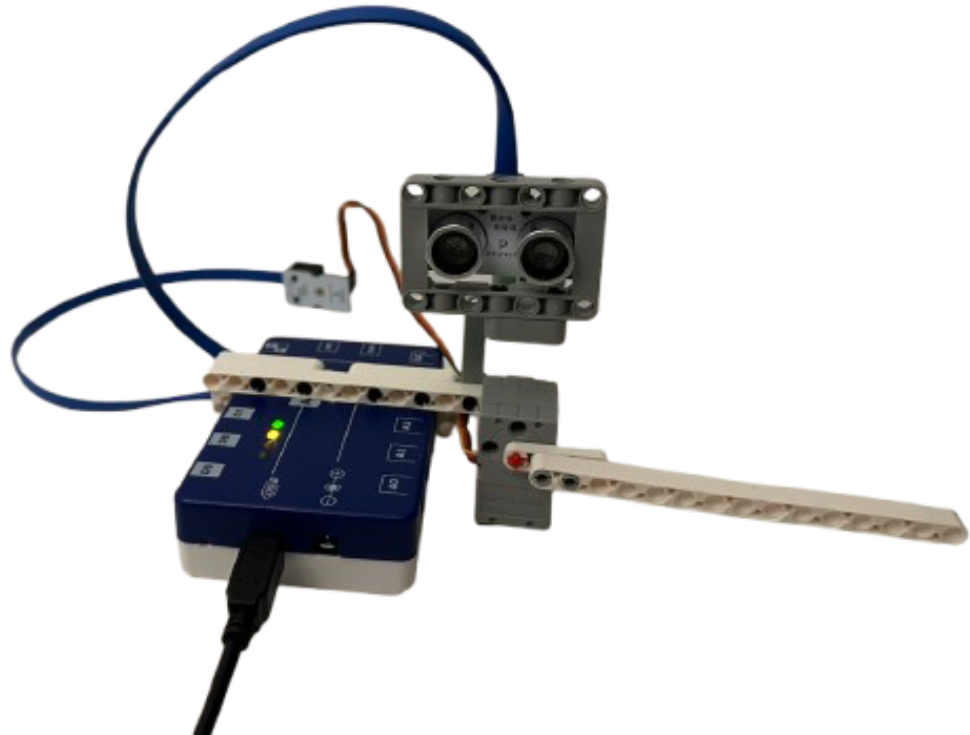
/#!
* MindPlus
* uno
*
*/
#include <SR01.h>
#include <DFRobot_Servo.h>
// Create an object
Servo servo_5;

// Main program start
void setup() {
    pinMode(3, OUTPUT);
    pinMode(4, INPUT);
    servo_5.attach(5);
}
void loop() {
    if ((getDistance(3,4)==10)) {
        servo_5.angle(0);
        delay(5000);
    }
    else {
        servo_5.angle(45);
    }
}

```

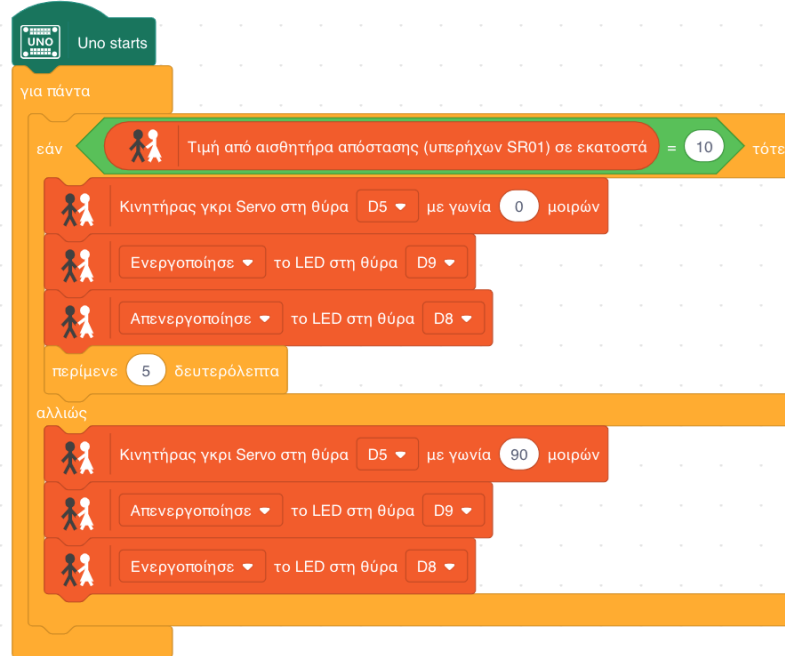


Ε. Το αποτέλεσμα



Βελτιώσεις του αυτοματισμού

Μια βελτίωση του αυτοματισμού θα ήταν η προσθήκη ενός πράσινου και ενός κόκκινου LED τα οποία θα άναβαν ανάλογα ώστε να ειδοποιούν τον οδηγό του οχήματος για το αν η μπάρα είναι ανοιχτή η κλειστή...



Ο κώδικας σε Arduino C

```

/*!
 * MindPlus
 * uno
 *
 */
#include <SR01.h>
#include <DFRobot_Servo.h>
// Create an object
Servo servo_5;

// Main program start
void setup() {
  pinMode(3, OUTPUT);
  pinMode(4, INPUT);
  servo_5.attach(5);
  pinMode(9, OUTPUT);
  pinMode(8, OUTPUT);
}
void loop() {
  if ((getDistance(3,4)==10)) {
    servo_5.angle(0);
    digitalWrite(9, HIGH);
    digitalWrite(8, LOW);
    delay(5000);
  }
  else {
    servo_5.angle(45);
    digitalWrite(9, LOW);
    digitalWrite(8, HIGH);
  }
}

```



8η δραστηριότητα: Φανός με Neopixel, διακόπτη και αισθητήρα κίνησης

A. Στοιχεία Δραστηριότητας

Οι Μαθητές/τριες θα σχεδιάσουν και θα υλοποιήσουν μία λάμπα φωτισμού όπου με αισθητήρα κίνησης ή διακόπτη θα ανάβει σε διαφορετικά χρώματα και ένταση. Θα αξιοποιήσουν παθητικά και ενεργά στοιχεία, ιδιαίτερα αισθητήρες και Neopixel LED, συνδυάζοντας δομές προγραμματισμού, ώστε να επιτύχουν το επιθυμητό αποτέλεσμα

Σκοπός

Η ενίσχυση της κατανόησης των Μαθητών/τριών για τις βασικές αρχές αυτοματισμών, ενθαρρύνοντας τη δημιουργικότητα, τον προγραμματισμό, και τη συνεργατική εργασία, αποτελούν τους διδακτικούς στόχους της δραστηριότητας αυτής. Πιο συγκεκριμένα η κατανόηση βασικών αρχών της ρομποτικής, όπως ο έλεγχος των κατασκευών. Η εξοικείωση με τη χρήση του Neopixel Led και η εκμάθηση των βασικών δομών και εντολών της πλατφόρμας προγραμματισμού του ARD:icon – Mind+ για τον έλεγχο της κατασκευής. Η σχεδίαση της βάσης της λάμπας με συγκεκριμένα κριτήρια, όπως η ισορροπία, και ο προγραμματισμός για τον έλεγχο της κατασκευής. Η αντιμετώπιση προβλημάτων κατά τη δημιουργία, τον προγραμματισμό και τη λειτουργία της λάμπας, μέσα από την ομαλή συνεργασία και επικοινωνία με άλλους μαθητές για την επίτευξη κοινών στόχων. Τέλος η κοινή αξιολόγηση που θα οδηγήσει στην επιλογή των βέλτιστων λύσεων για το θέμα.

Στόχοι

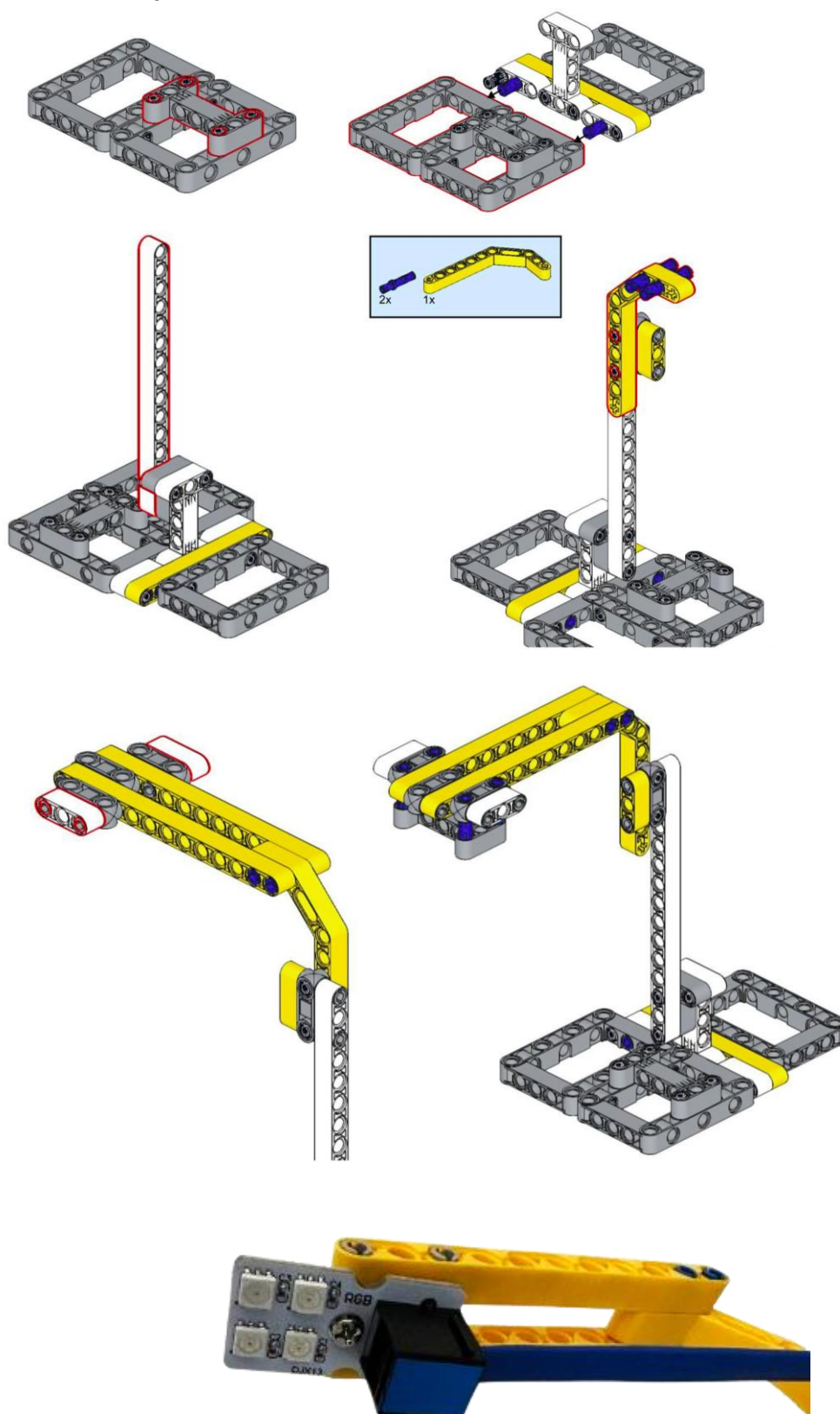
- **Ορίζω (Γνώση):** Τα θέματα της ισορροπίας των στελεχών που απαιτούνται για την κατασκευή μας.
- **Επιλέγω (Γνώση):** Τα κατάλληλα παθητικά και ενεργά στοιχεία που θα χρειαστώ για την κατασκευή μου.
- **Αναγνωρίζω (Γνώση):** Την αξιοποίηση της κατασκευής μας για τις καθημερινές μας ανάγκες και την εξοικονόμηση ενέργειας.
- **Περιγράφω λόγους για (Κατανόηση):** τους οποίους είναι σημαντική η χρήση της λάμπας αυτής στους δρόμους ή/και τις κατοικίες.
- **Συναρμολογώ (Εφαρμογή):** Τα παθητικά και τα ενεργά στοιχεία που απαιτούνται με γνώμονα την ισορροπία και την σταθερότητα της κατασκευής μου.
- **Αναφέρω συστατικά μέρη (Ανάλυση):** Της υπό κατασκευής λάμπας.
- **Καταρτίζω (Σύνθεση):** Σχέδιο για την λειτουργία και αξιοποίηση του αυτοματισμού μου
- **Προβλέπω (Ανάλυση):** Τον μέγεθος των στελεχών της κατασκευής, σύμφωνα πάντα με την αρχικό σχέδιο.
- **Προβληματίζομαι (Αξιολόγηση):** Για την όσο το δυνατόν καλύτερη αξιοποίηση των επιλογών έναρξης των κινήσεων του κάθε στελέχους της κατασκευής μου.
- **Τροποποιώ (Εφαρμογή):** Την κατασκευή μου ώστε να είναι όσο το δυνατόν πιο αποτελεσματική, σύμφωνα με τον αρχικό στόχο.
- **Αξιολογώ (Αξιολόγηση):** Τον τρόπο εφαρμογής και το αποτέλεσμα της λειτουργίας της μπάρας

Χρόνος μάθησης
2 ώρες.

Μέγεθος τάξης
14 Μαθητές/τριες.



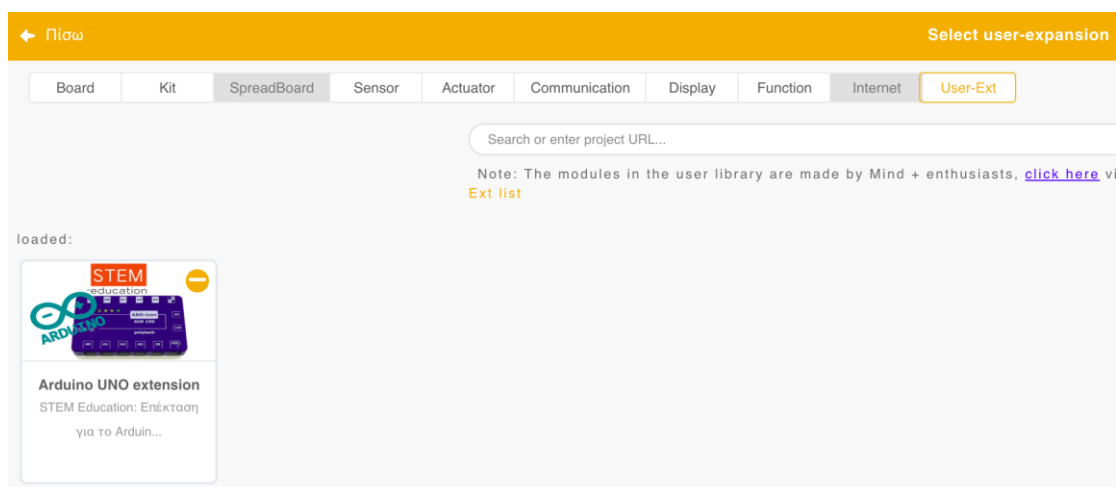
Β. Κατασκευή



Γ. Συνδεσμολογία

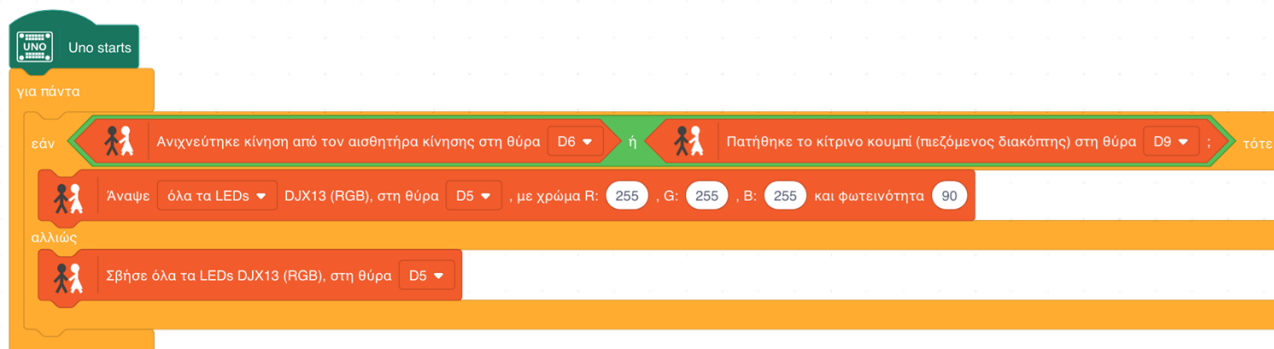


Δ. Επέκταση (extension)

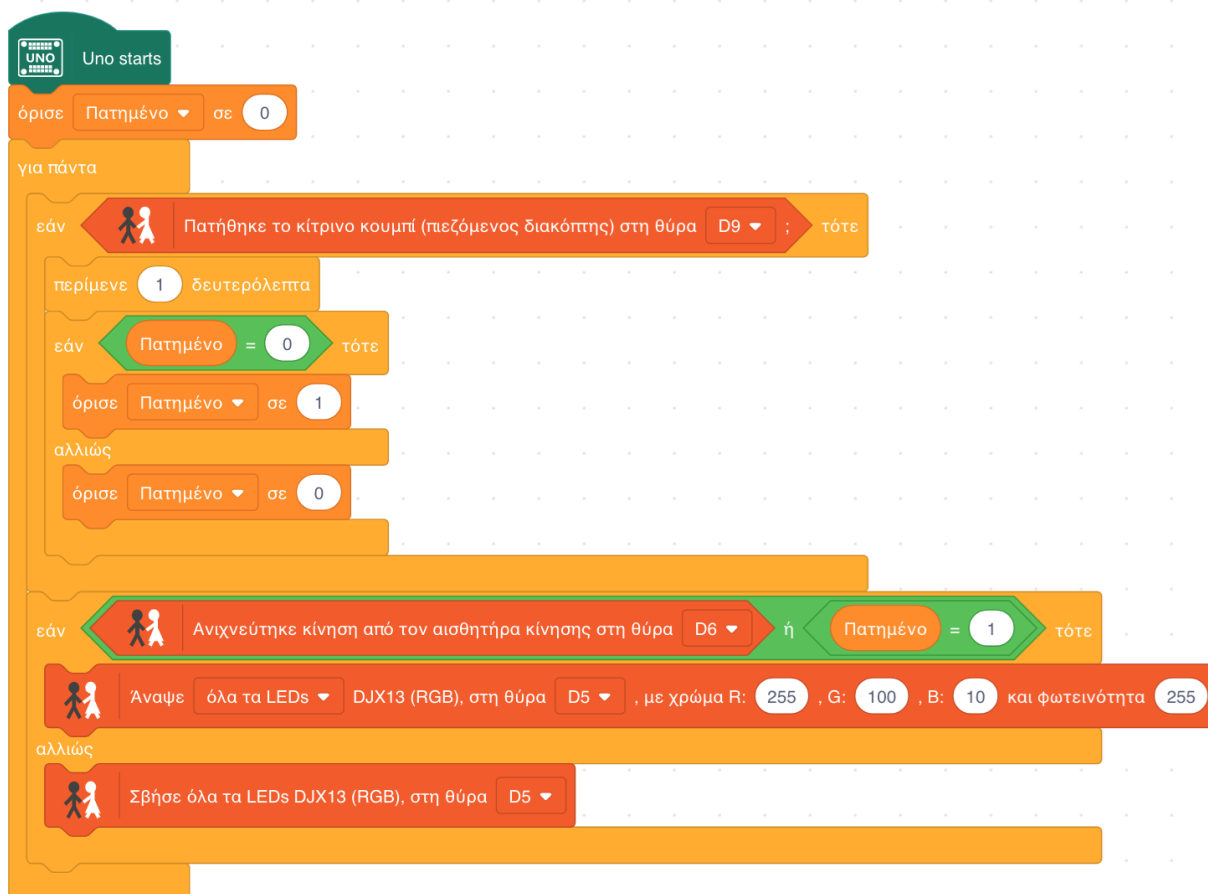


Ε. Κώδικας

Α' Προσπάθεια

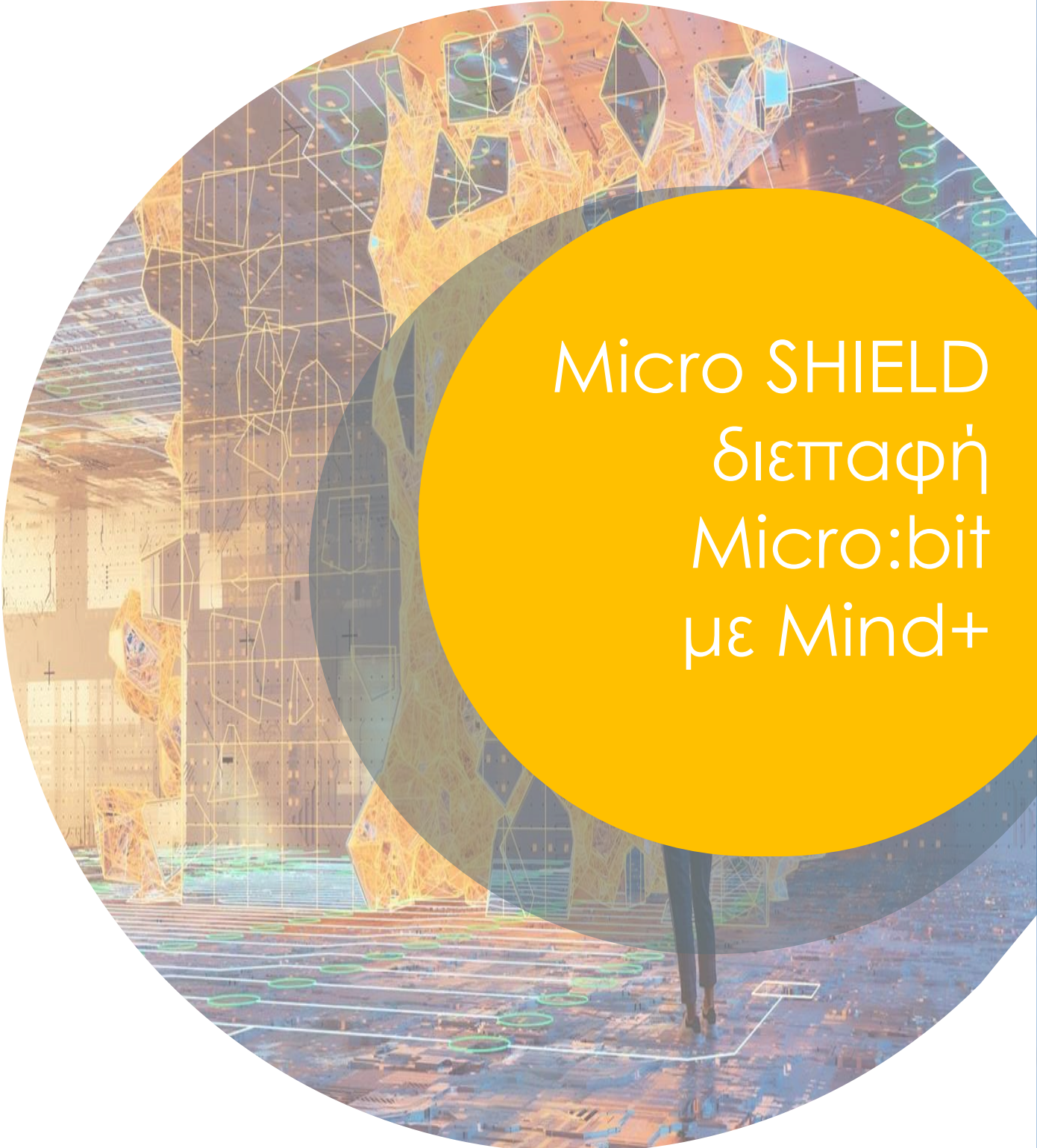


Β' Προσπάθεια



Με τον κώδικα αυτό αν ανιχνευτεί κίνηση μπροστά από το φανό ή πατηθεί το κίτρινο κουμπί, θα ανάψουν και τα τέσσερα LEDs του Neorixel με ζεστό κίτρινο χρώμα και φωτεινότητα 255 στα 255. Ο φανός θα σβήσει όταν πατηθεί το κίτρινο κουμπί ξανά.

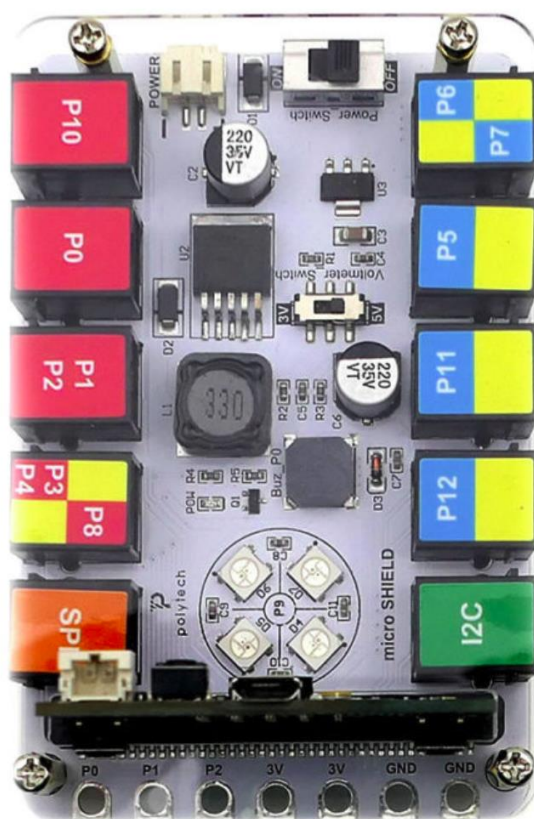




Micro SHIELD διεπαφή Micro:bit με Mind+



Το micro SHIELD διεπαφή micro:bit



Το Micro SHIELD είναι μια πρακτική πλακέτα επέκτασης που έχει σχεδιαστεί για να ενσωματώνει τον μικροελεγκτή micro:bit με ευκολία. Παρέχει μια σειρά από χαρακτηριστικά και δυνατότητες που διευκολύνουν την ανάπτυξη έργων και πειραμάτων.

Χαρακτηριστικά

- **Θύρα Σύνδεσης:** Διαθέτει υποδοχή για τον μικροελεγκτή micro, επιτρέποντας εύκολη και γρήγορη ενσωμάτωσή του.
- **LEDs:** Περιλαμβάνει 4 φωτεινές δίοδοι LED που μπορούν να χρησιμοποιηθούν για οπτικές ενδείξεις και διαδραστικότητα στα έργα σας.
- **Βομβητής:** Ένας ενσωματωμένος βομβητής παρέχει ηχητική ανατροφοδότηση, ιδανικός για παιχνίδια ή ειδοποιήσεις.
- **Διαμορφωτής Τάσης Εξόδου:** Ενσωματωμένος διαμορφωτής τάσης εξόδου που εξασφαλίζει τη σωστή λειτουργία των συνδεδεμένων συσκευών.
- **Τάση Λειτουργίας:** Λειτουργεί με τάση 9V, η οποία μπορεί να προμηθευτεί μέσω μπαταριοθήκης 6xAA 1,5V.

Η πλακέτα micro SHIELD είναι ιδανική για εκπαιδευτικά έργα και πειράματα, διευκολύνοντας τη σύνδεση και τη χρήση των διαθέσιμων συσκευών.

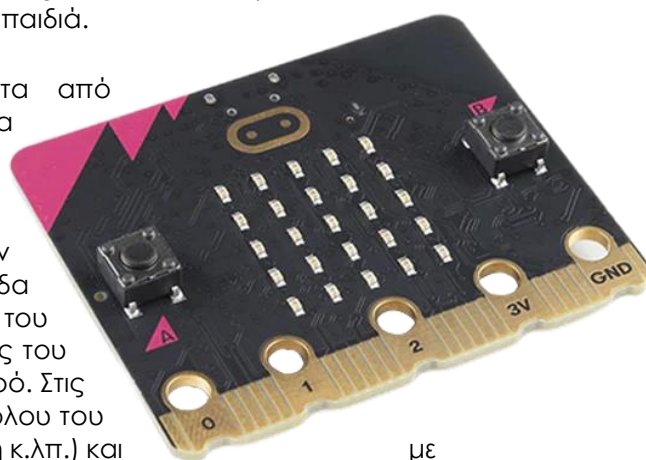


Micro:bit, ο μικροεπεξεργαστής του micro SHIELD

Το 2013 στο Ηνωμένο Βασίλειο παρουσιάστηκε ένα νέο πρόγραμμα σπουδών. Το συγκεκριμένο πρόγραμμα τροποποιήθηκε από την κλασσική διδασκαλία που έχουμε στην πληροφορική (Εξάσκηση σε δυνατότητες και σε εργαλεία του Η/Υ και παρουσίαση), σε εκμάθηση του προγραμματισμού μέσω της χρήσης. Κατά συνέπεια οι εκπαιδευόμενοι αντί να διδάσκονται πως να εργάζονται στον Η/Υ, θα διδάσκονται το πως να τον προγραμματίζουν σύμφωνα με τα θέλω τους.

Η δημιουργικότητα και η πρωτότυπη σκέψη αποκτούν πλέον μεγάλη βαρύτητα δίνοντας την ευκαιρία στους εκπαιδευόμενους να δουλεύουν όπως αυτοί θέλουν, ανακαλύπτοντας μόνοι τους τα προβλήματά τους και να λύσεις. Ο προγραμματισμός Η/Υ πλέον δεν πρέπει να θεωρείται αυτοσκοπός, αλλά ένα εργαλείο διερεύνησης που μας παρέχει ευρηματικές λύσεις. Χρησιμοποιώντας αυτή την προσέγγιση ο προγραμματισμός προσελκύει όλο και περισσότερα παιδιά.

Ο μικροελεγκτής του Micro:bit προέκυψε έπειτα από πρωτοβουλία του BBC, να δημιουργήσει μια πλακέτα όπου το κόστος της θα είναι χαμηλό και θα μπορεί να χρησιμοποιηθεί στην σχολική κοινότητα. Η χρήση του μας παρέχει μεγάλο εύρος δραστηριοτήτων από τον έλεγχο ρομπότ μέχρι και τον έλεγχο μουσικών οργάνων. Ωστόσο η ηλικιακή ομάδα που χρησιμοποιείται είναι μικρότερη από αυτή του Arduino. Ο κύριος λόγος είναι ότι οι συνδεσμολογίες του είναι πιο απλές και το μέγεθος της πλακέτας είναι μικρό. Στις ημέρες μας το Micro:bit έχει εισχωρήσει σε σχολεία όλου του κόσμου (Φινλανδία, Ισλανδία, Σρι Λάνκα, Σιγκαπούρη κ.λπ.) και γρήγορα βήματα εισέρχεται και στην Ελλάδα.



με

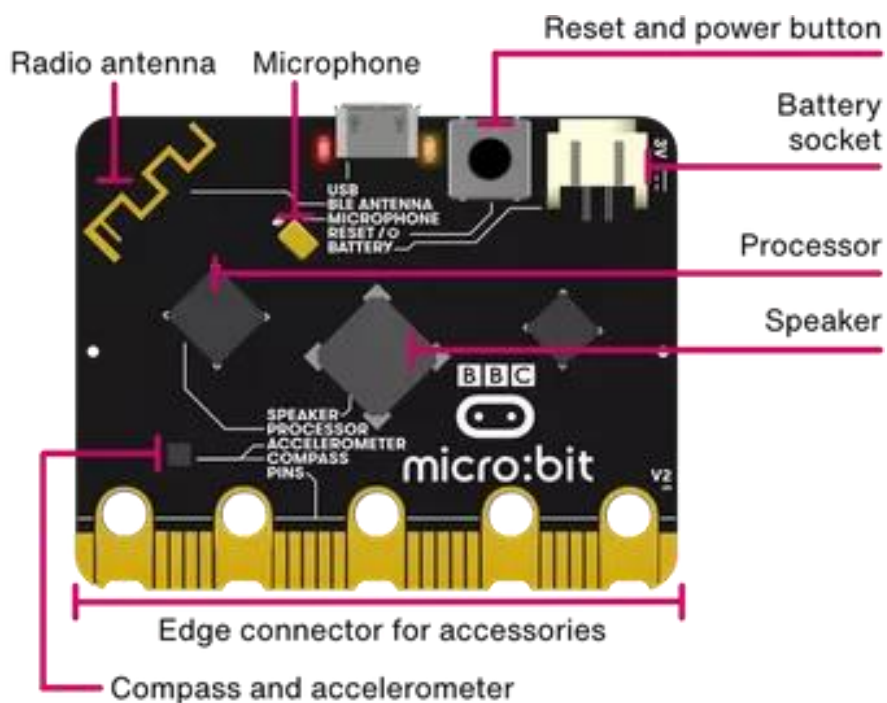
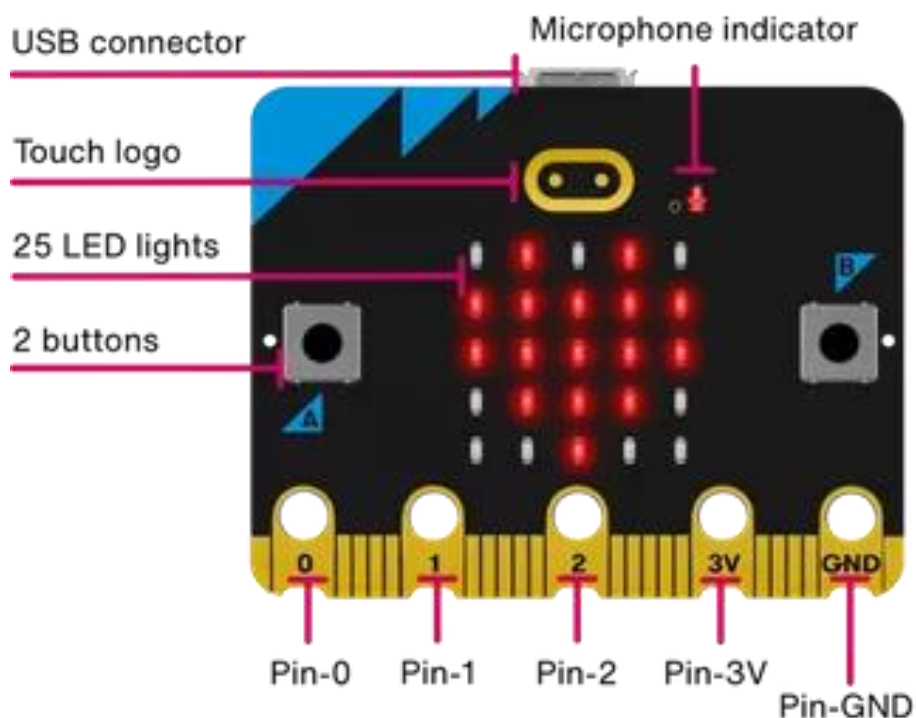
Το micro:bit είναι μια μικρή πλακέτα (όχι μεγαλύτερη από μια πιστωτική κάρτα) που περιλαμβάνει:

- έναν ισχυρό ARM M0+ μικροελεγκτή (microcontroller),
- 2 φυσικά κουμπιά (A και B),
- Ένα πλέγμα από led (led matrix - 5x5) που μπορεί να απεικονίσει κείμενο και γραφικά,
- ένα επιταχυνσιόμετρο (accelerometer) και ένα μαγνητόμετρο (magnetometer) που μπορεί να ανιχνεύσει την σχετική θέση και κατεύθυνση της πλακέτας,
- έναν αισθητήρα φωτός,
- ένα θερμόμετρο,
- μια ομάδα από ακροδέκτες στους οποίους μπορούμε να συνδέσουμε διάφορα εξωτερικά εξαρτήματα στο micro:bit.

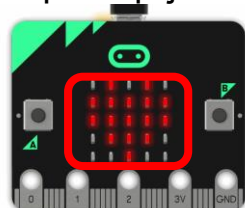
Έχει 5 στρογγυλούς μεταλλικούς ακροδέκτες. 3 από αυτούς είναι ακροδέκτες εισόδου/εξόδου στους οποίους μπορούμε να συνδέσουμε αισθητήρες, φωτοδιόδους (leds), κ.λπ.

Υπάρχει επίσης ένας ακροδέκτης 3V που μπορεί να παρέχει ρεύμα και ένας ακροδέκτης γείωσης (GND).





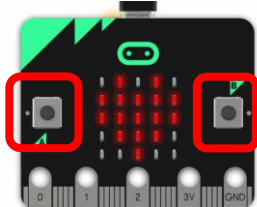
Με περισσότερες λεπτομέρειες τα χαρακτηριστικά του micro:bit



LEDs

Light Emitting Diodes δίοδοι εκπομπής φωτός, χρησιμοποιούνται ευρέως σε πολλές συσκευές ως ενδεικτικές λυχνίες ή για φωτισμό.

Το micro:bit διαθέτει 25 ξεχωριστά προγραμματιζόμενα LEDs που μας επιτρέπουν να εμφανίσουμε κείμενο, αριθμούς και εικόνες



Κουμπιά (Buttons)

Υπάρχουν δύο κουμπιά μπροστά στο micro:bit (εμφανίζονται με την επιγραφή A και B). Μπορούμε να ανιχνεύσουμε πότε αυτά τα κουμπιά πατήθηκαν, ώστε να εκτελείται κάποιος κώδικας στη συσκευή.



Ακροδέκτες σύνδεσης (PINS)

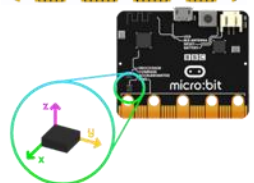
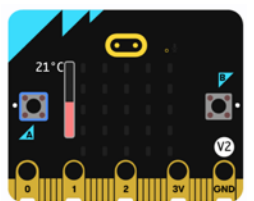
Υπάρχουν 25 εξωτερικοί ακροδέκτες στο κάτω μέρος του micro:bit που τα ονομάζουμε pins. Μέσω αυτών το micro:bit μπορεί να επικοινωνήσει ενσύρματα με τον εξωτερικό κόσμο. Μπορούμε έτσι να συνδέσουμε κινητήρες, LEDs, αισθητήρες και άλλες ηλεκτρικές και ηλεκτρονικές διατάξεις.

Αισθητήρας φωτός (Light Sensor)

Το micro:bit μπορεί αντιστρέφοντας τη λειτουργία των LEDs, να τα χρησιμοποιήσει ως είσοδο, ανιχνεύοντας την παρουσία φωτός.

Αισθητήρας θερμοκρασίας (Temperature sensor)

Αυτός ο αισθητήρας επιτρέπει στο micro:bit να ανιχνεύει την τρέχουσα θερμοκρασία σε βαθμούς κελσίου.



Επιταχυνσιόμετρο (Accelerometer)

Το επιταχυνσιόμετρο μετρά την επιτάχυνση του micro:bit, άρα μπορεί να χρησιμοποιηθεί για να ανιχνεύει πότε το micro:bit μετακινείται.



Πυξίδα (Compass)

Η πυξίδα (μαγνητόμετρο) είναι ένας αισθητήρας που μπορεί να ανιχνεύει το μαγνητικό πεδίο της γης, επιτρέποντας με τον τρόπο αυτό να προσδιορίζουμε την κατεύθυνση που το micro:bit δείχνει. Η πυξίδα θα πρέπει πριν χρησιμοποιηθεί να βαθμονομηθεί (calibration).

Ραδιοεπικοινωνία (Radio)

Το micro:bit διαθέτει δυνατότητα ασύρματης επικοινωνίας (μέσω ραδιοκυμάτων) με άλλες συσκευές ή άλλα micro:bit επιτρέποντας έτσι να κατασκευάσουμε παιχνίδια με πολλούς χρήστες κ.α.

Bluetooth

Το micro:bit διαθέτει μία κεραία BLE (Bluetooth Low Energy) που του επιτρέπει να στέλνει και να λαμβάνει σήματα Bluetooth. Έτσι μπορεί να επικοινωνήσει ασύρματα με υπολογιστές, τηλέφωνα, tablets. Μπορούμε για παράδειγμα να ελέγξουμε το τηλέφωνό μας από το micro:bit ή να στείλουμε δεδομένα και εντολές στο micro:bit από το κινητό μας τηλέφωνο.

Πριν την επικοινωνία μέσω Bluetooth θα πρέπει να γίνει κατάλληλη ζεύξη των δύο συσκευών.

Διασύνδεση USB

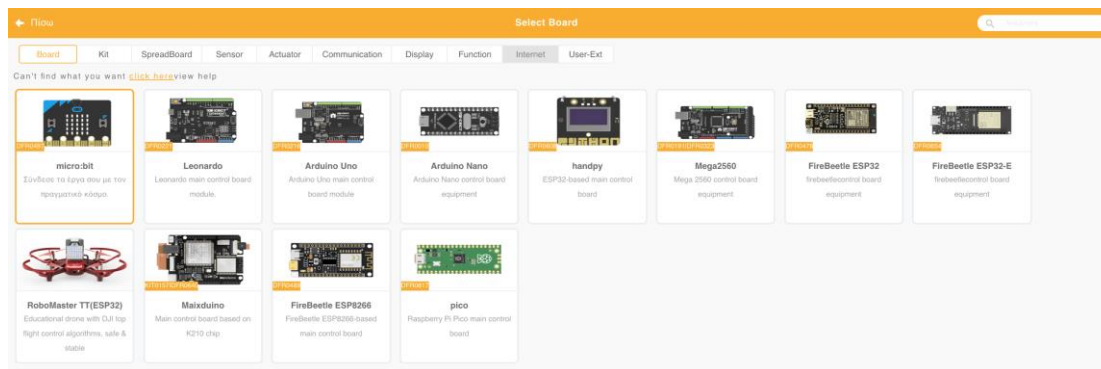
Η διασύνδεση USB επιτρέπει στο micro:bit να συνδεθεί με τον υπολογιστή μας μέσω ενός micro-USB καλωδίου, ώστε να μεταφέρουμε τα προγράμματά μας στο micro:bit αλλά ταυτόχρονα από το ίδιο καλώδιο παρέχουμε την κατάλληλη τάση τροφοδοσίας για τη λειτουργία του



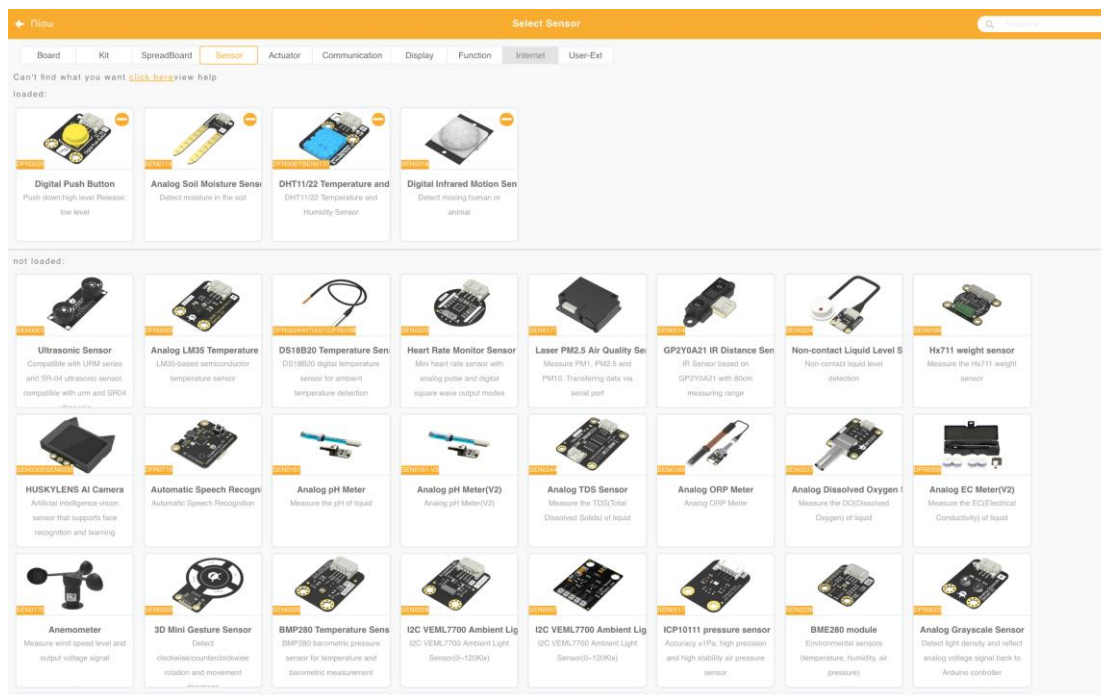
Τα πρόσθετα στο Mind+

Εισαγωγή των πρόσθετων που χρειαζόμαστε για να θέσουμε σε λειτουργία τον εξοπλισμό

Επιλογή μικροεπεξεργαστή



Επιλογή επεκτάσεων αισθητήρων



Επιλογή επέκτασης για LCD οθόνης

← Πίνακας Select Display

Board Kit SpreadBoard Sensor Actuator Communication **Display** Function Internet User-Ext

Can't find what you want [click here](#) view help

loaded:



LCD1602 Display
I2C LCD1602 Display Module

not loaded:



OLED-12864 Display
I2C OLED-12864 Display Module



WS2812 RGB LED Strip
To control WS2812-based LED strip modules



Digital LED Light Module
White, Red, Green and Blue LED Modules



TM1650 Four digit tube
Four digit tube



MAX7219 8x8 dot matrix
8x8 lattice module



8x16 RGB LED Matrix Panel
8x16 RGB LED dot matrix emoji board

Επιλογή επέκτασης για Micro Servo κινητήρα

← Πίνακας Select Display

Board Kit SpreadBoard Sensor Actuator Communication **Display** Function Internet User-Ext

Can't find what you want [click here](#) view help

loaded:



LCD1602 Display
I2C LCD1602 Display Module

not loaded:



OLED-12864 Display
I2C OLED-12864 Display Module



WS2812 RGB LED Strip
To control WS2812-based LED strip modules



Digital LED Light Module
White, Red, Green and Blue LED Modules



TM1650 Four digit tube
Four digit tube



MAX7219 8x8 dot matrix
8x8 lattice module



8x16 RGB LED Matrix Panel
8x16 RGB LED dot matrix emoji board



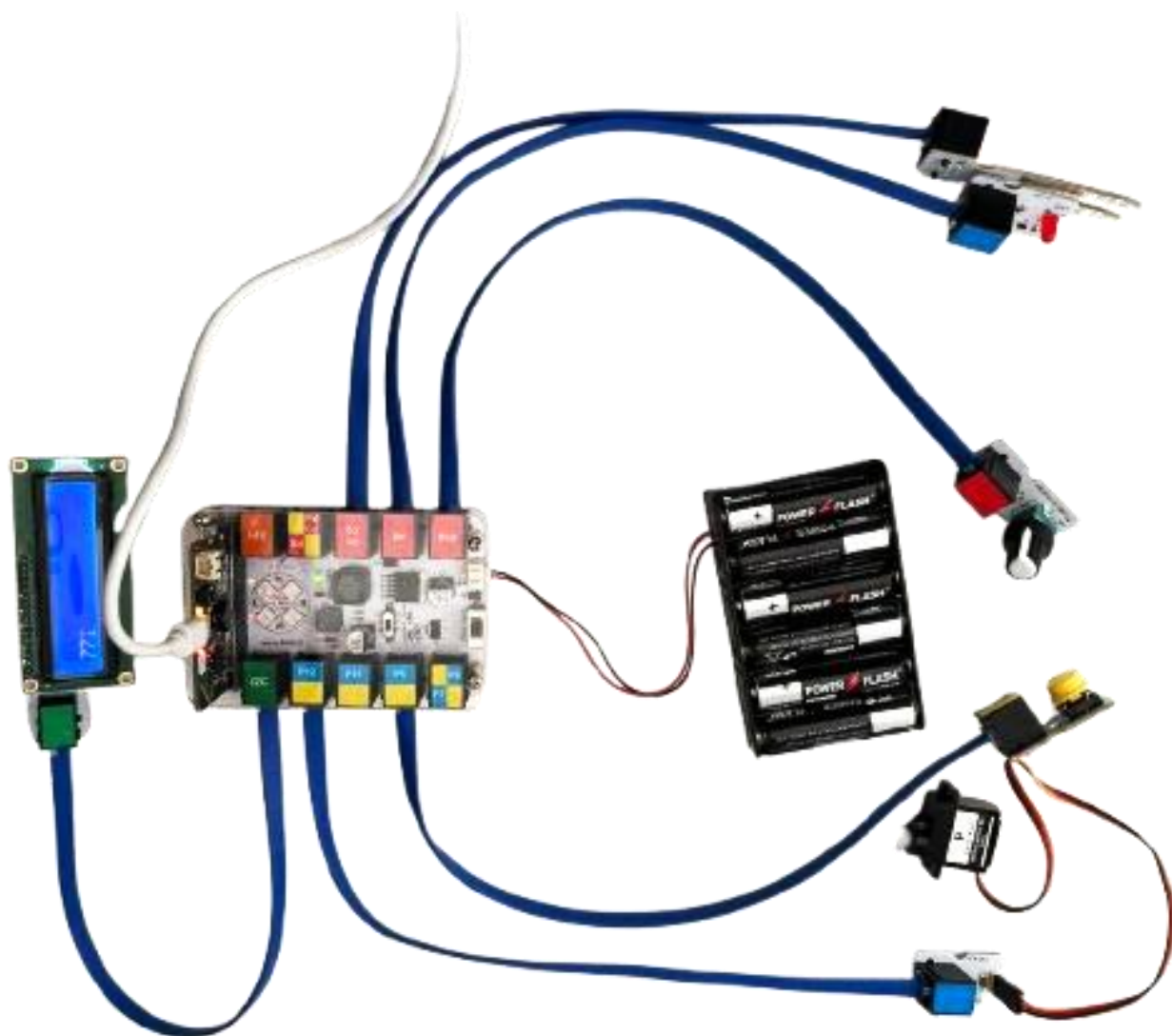
1^ο παράδειγμα: Λειτουργώντας IoTs, micro:bit στο Mind+

A. Σενάριο

Να συνδέσουμε και να λειτουργήσουμε όσο το δυνατόν περισσότερα IoTs στο **micro SHIELD** διεπαφή **micro:bit** με **Mind+**

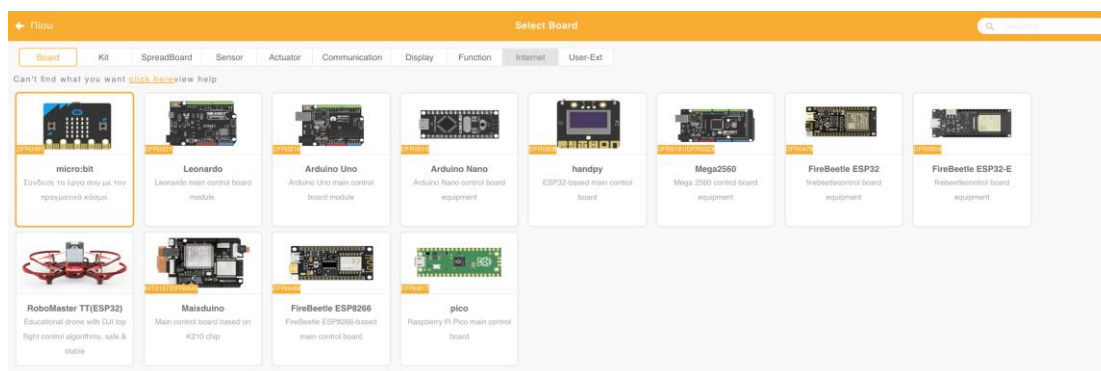
Συγκεκριμένα, με το πάτημα του κίτρινου κουμπιού που είναι συνδεδεμένο στην P5 ψηφιακή θύρα,

- εμφανίζονται στην οθόνη LCD οι ενδείξεις:
 - Τιμή ποτενσιόμετρου
 - Τιμή αισθητήρα υγρασίας εδάφους
- Ανάβει το κόκκινο Led
- Στρέφει ο Micro Servo κινητήρας στις 90°

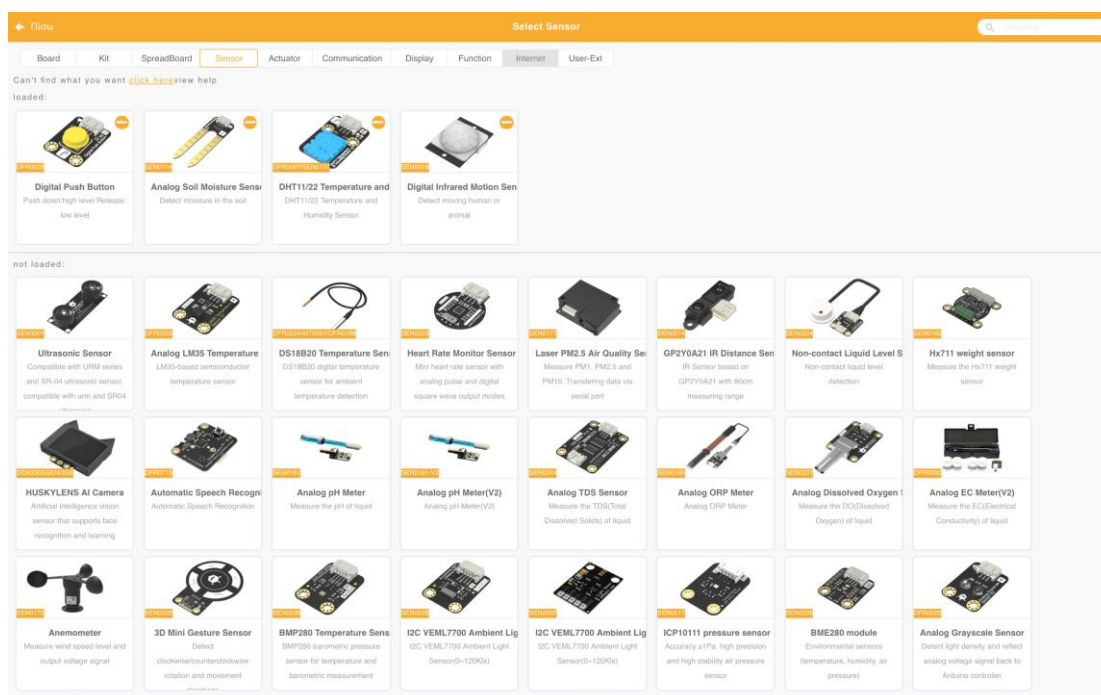


Β. Εισαγωγή των πρόσθετων που χρειαζόμαστε για να θέσουμε σε λειτουργία τον εξοπλισμό

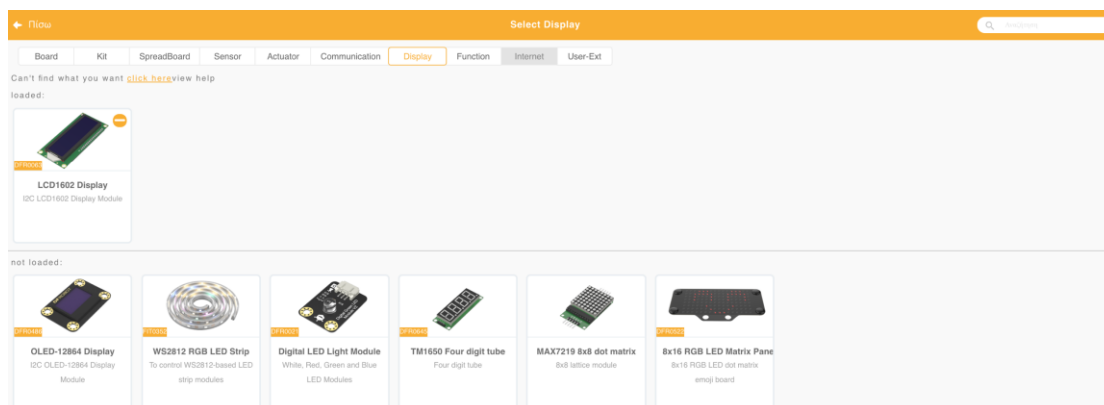
Επιλογή μικροεπεξεργαστή



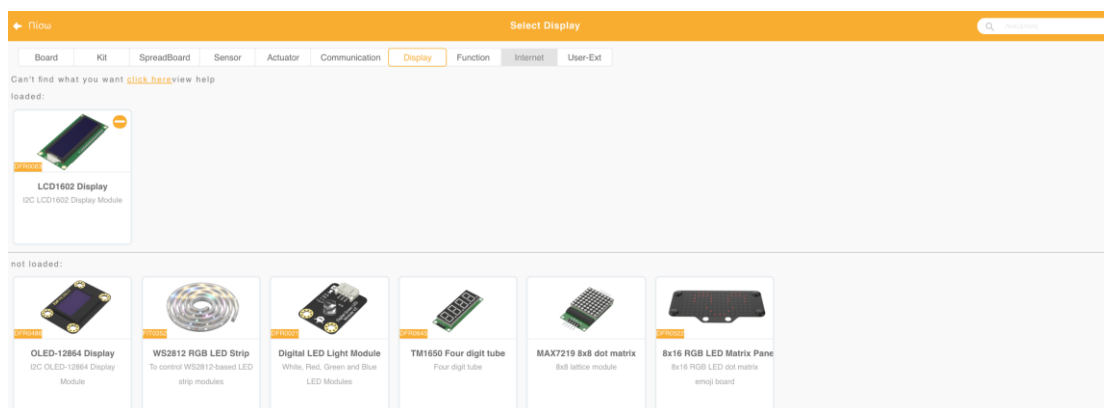
Επιλογή επεκτάσεων αισθητήρων



Επιλογή επέκτασης για LCD οθόνης



Επιλογή επέκτασης για Micro Servo κινητήρα



Γ. Προγραμματισμός (σε Mind+)

Ένα πρόγραμμα που ελέγχει τα IoTs που έχουμε συνδέσει

The screenshot shows the Mind+ V1.7.3 RC2.0 software interface. The main workspace displays a block-based program for a micro:bit. The program starts with 'start micro:bit', followed by 'display I2C device address' and 'initialize I2C LCD screen address 0x27'. A 'for all' loop contains a 'when button P5 is pressed?' block. Inside this loop, there are two branches: one for 'LOW' which clears the screen and sets a servo to 0 degrees, and another for 'HIGH' which reads a soil moisture sensor and sets a servo to 90 degrees. The program also includes a 'display' block to show the analog value of the soil moisture sensor on LCD line 2.

```

1  //
2  * MindPlus
3  * microbit
4  //
5
6  #include <DFRobot_Servo.h>
7  #include <DFRobot_IICScan.h>
8  #include <Microbit_Matrix.h>
9  #include <DFRobot_LiquidCrystal_I2C.h>
10
11 // Create an object
12 DFRobot_IICScan iicscan;
13 DFRobot_LiquidCrystal_I2C lcd1602;
14 Servo servo_12;
15
16 // Main program start
17 void setup() {
18   servo_12.attach(12);
19   Matrix.print(iicscan.scan());
20   lcd1602.begin(0x27);
21 }
22 void loop() {
23   if (digitalRead(5)) {
24     lcd1602.clear();
25     digitalWrite(0, LOW);
26     servo_12.angle(abs(0));
27   }
28   else {
29     lcd1602.println(uint32_t(1), analogRead(1));
30     digitalWrite(0, HIGH);
31     servo_12.angle(abs(90));
32   }
33   lcd1602.println(uint32_t(2), (analogRead(10)));
34 }
35
  
```

Ο κώδικας

```

start micro:bit
display I2C device address
initialize I2C LCD screen address 0x27

for all
  when button P5 is pressed?
    then
      clear the screen
      set pin P0 to LOW
      set pin P12 servo to 0 degree
    else
      display read pin P1 Soil Moisture Sensor in LCD line 1
      set pin P0 to HIGH
      set pin P12 servo to 90 degree
      display read analog I2C device P10 in LCD line 2
  
```



```

/!*
 * MindPlus
 * microbit
 *
 */
#include <DFRobot_Servo.h>
#include <DFRobot_IICScan.h>
#include <Microbit_Matrix.h>
#include <DFRobot_LiquidCrystal_I2C.h>

// Create an object
DFRobot_IICScan      iicscan;
DFRobot_LiquidCrystal_I2C lcd1602;
Servo                 servo_12;

// Main program start
void setup() {
    servo_12.attach(12);
    MMatrix.print(iicscan.scan());
    lcd1602.begin(0x27);
}
void loop() {
    if (digitalRead(5)) {
        lcd1602.clear();
        digitalWrite(0, LOW);
        servo_12.angle(abs(0));
    }
    else {
        lcd1602.println(uint32_t(1), analogRead(1));
        digitalWrite(0, HIGH);
        servo_12.angle(abs(90));
    }
    lcd1602.println(uint32_t(2), (analogRead(10)));
}

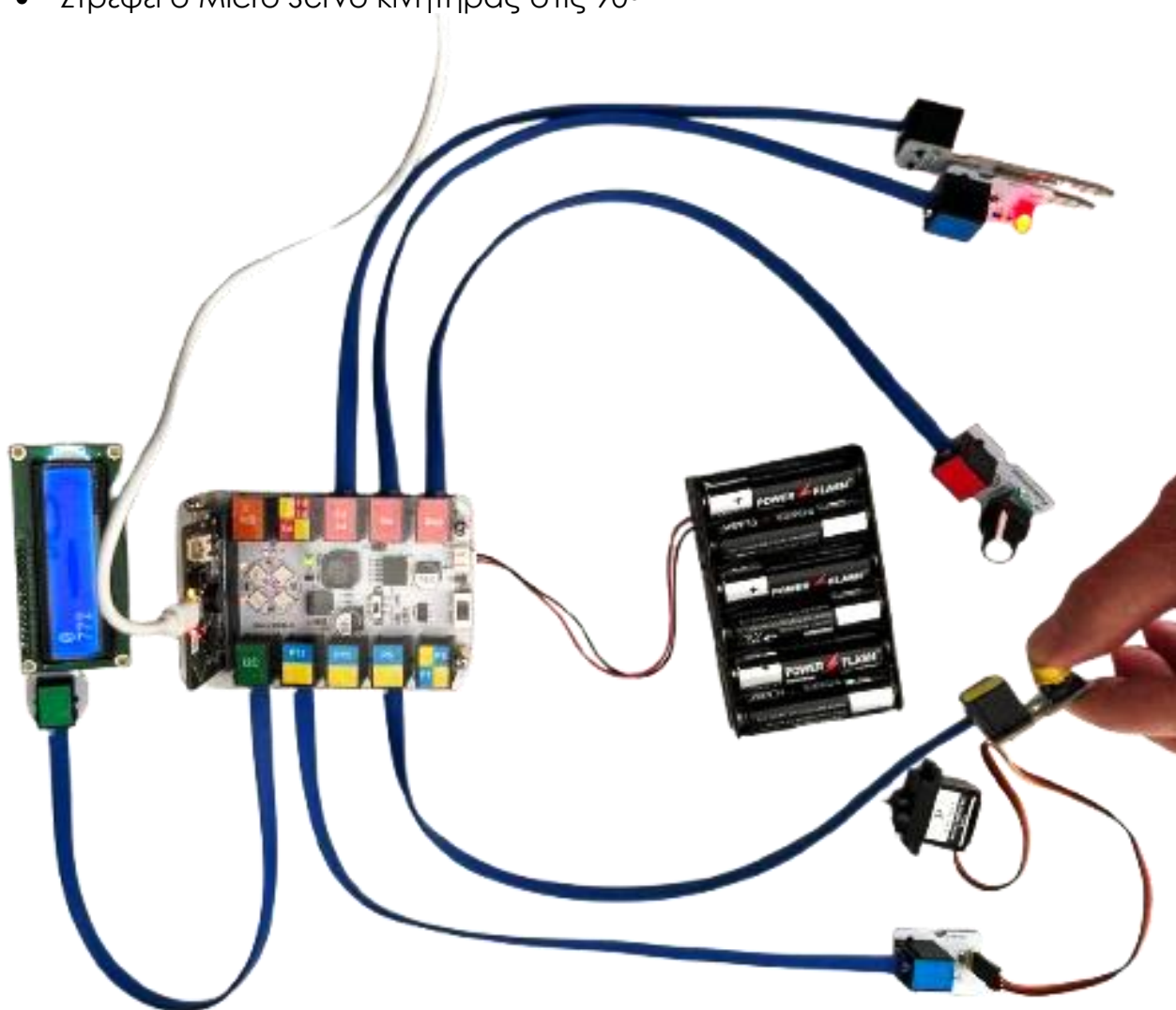
```



Δ. Η λειτουργία

Με το πάτημα του κίτρινου κουμπιού που είναι συνδεδεμένο στην P5 ψηφιακή θύρα,

- εμφανίζονται στην οθόνη LCD οι ενδείξεις:
 - ο Τιμή ποτενσιόμετρου
 - ο Τιμή αισθητήρα υγρασίας εδάφους
- Ανάβει το κόκκινο Led
- Στρέφει ο Micro Servo κινητήρας στις 90°







NEZHA

Inventor's Kit

Micro:bit



NEZHA Το βαλιτσάκι και το περιεχόμενό του (ενεργά και δομικά στοιχεία)

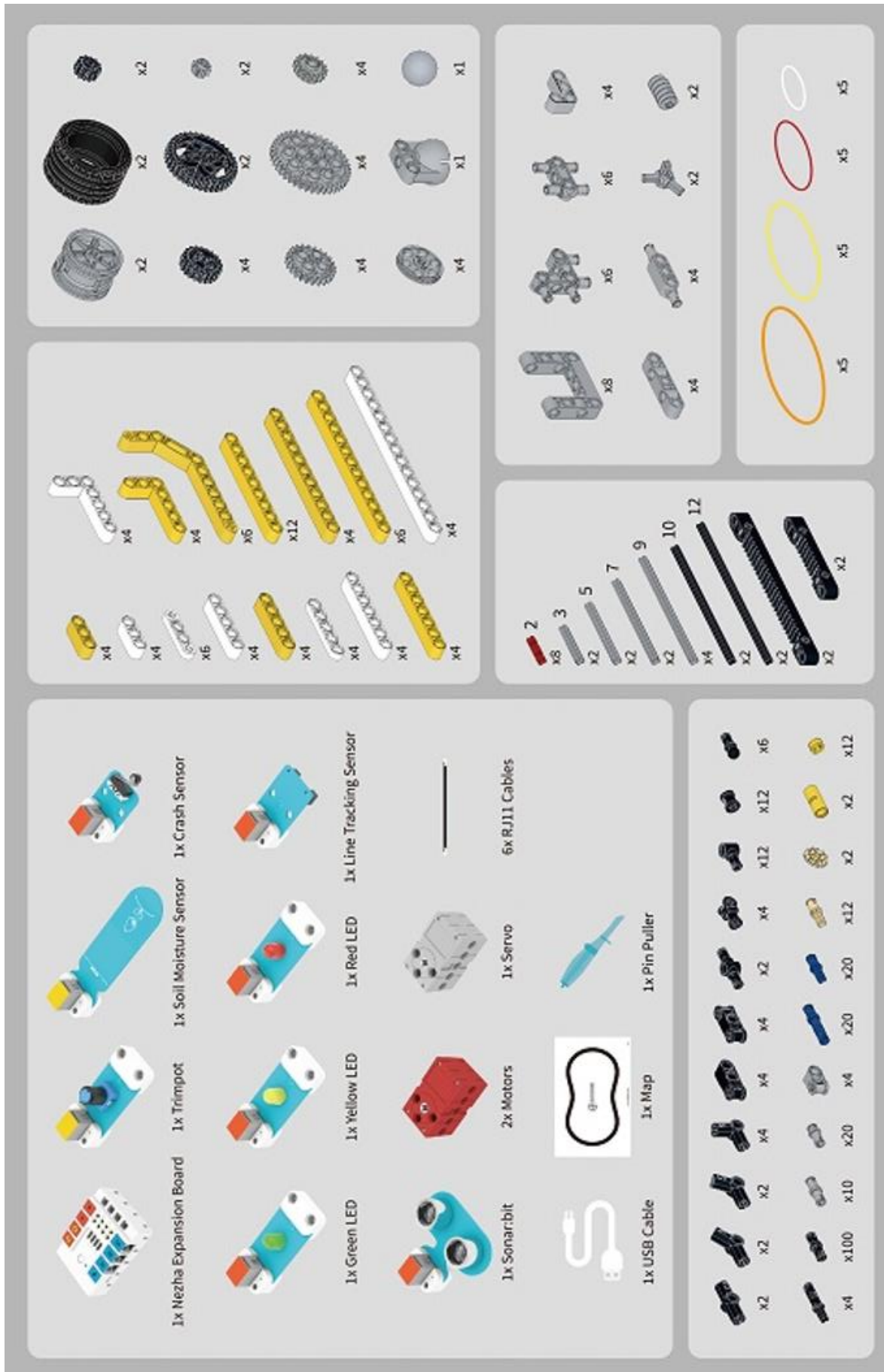


περιέχει:

A white plastic storage case is shown open, revealing its interior compartments. The case is filled with various electronic components and tools. In the top left, there is a white breadboard with several integrated circuits (ICs) and resistors. To its right, a blue and white breadboard is also populated with components. In the bottom left, there are several black and white resistors, a small red component, and a small blue component. In the bottom right, there is a small soldering iron with a green handle and a small blue component. A dashed arrow points from the text "περιέχει:" to the case.

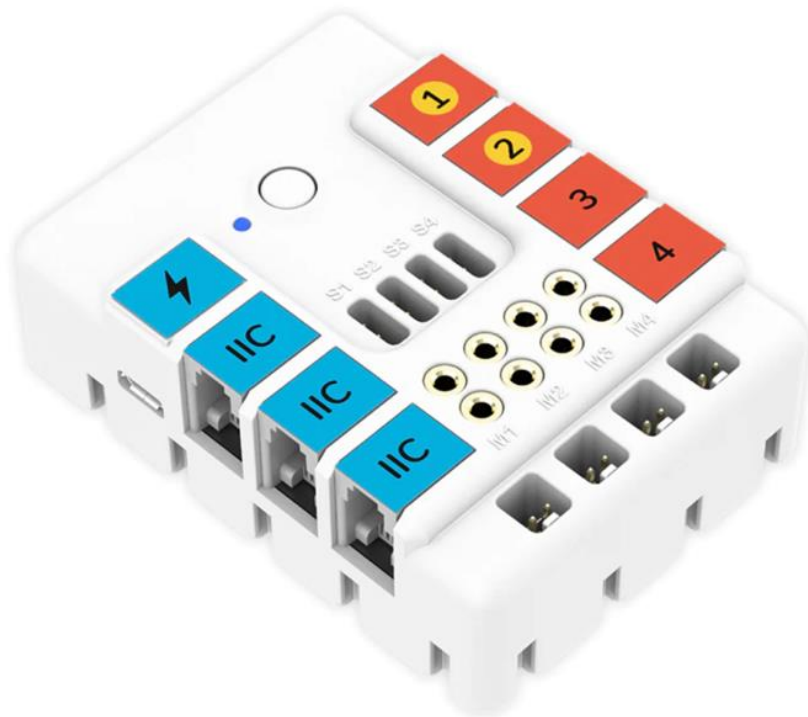
Καλό είναι να ταξινομήσουμε/ομαδοποιήσουμε το υλικό μας, για στοχευμένη αναζήτηση και εύκολη ανεύρεση.





To board NEZHA

Το Nezha breakout board είναι ένα ευέλικτο σύστημα επέκτασης για το micro: bit. Μπορεί να υποστηρίξει 4 σερβοκινητήρες, 4 κινητήρες και 7 διεπαφές αισθητήρων που χρησιμοποιούν υποδοχές RJ11. Το κέλυφος έχει επίσης σχεδιαστεί με συμβατές διεπαφές Lego και Fischertechnik.



Ειδικό σύστημα αναγνώρισης χρωμάτων, μας επιτρέπει να προσδιορίζουμε τις θύρες ψηφιακών, αναλογικών και IIC.

Για παράδειγμα μπορούμε να συνδέουμε ενεργό εξοπλισμό με βάση τις έγχρωμες ετικέτες που υπάρχουν και στο board Nezha και στους αισθητήρες (ή άλλο ενεργό εξοπλισμό) αντίστοιχα όπως παρουσιάζεται και στη συνέχεια:

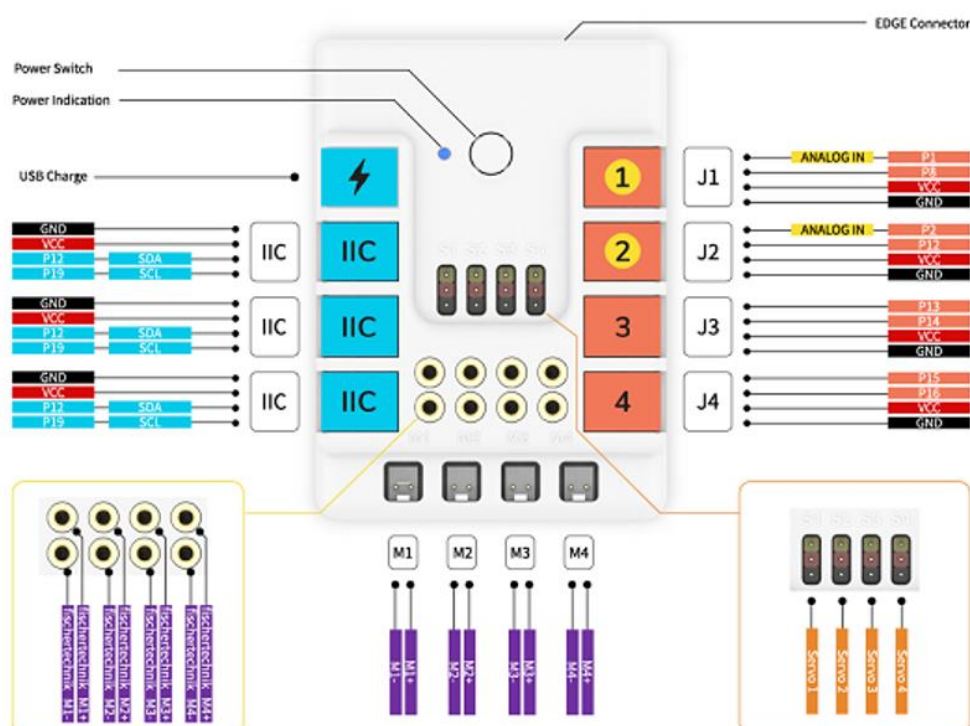
Board Nezha	Αισθητήρας ή ενεργός εξοπλισμός	Εντολή που επιστρέφει την αντίστοιχη μέτρηση ή μεταβάλλει κατάσταση



Πιο συγκεκριμένα:

- Στις θύρες στα δεξιά και αριστερά συνδέονται οι αισθητήρες,
- Στα παραλληλόγραμμα στο κέντρο συνδέεται ο σερβοκινητήρας
- Στα τετράγωνα κάτω οι κινητήρες.

Nezha breakout board V1.2

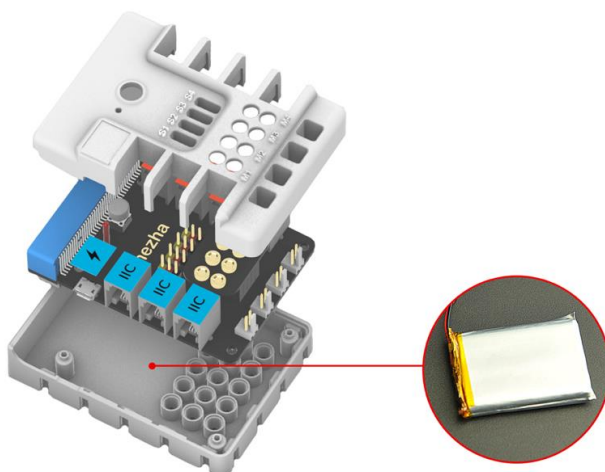


Το κόκκινο χρώμα σημαίνει ότι μια θύρα είναι ψηφιακή και το κίτρινο ότι είναι αναλογική.

Οι θύρες 1 και 2 είναι και αναλογικές και ψηφιακές.

Οι θύρες 3 και 4 είναι μόνο ψηφιακές.





Το Nezha board υποστηρίζει έως και 40+ αισθητήρες, όπως LED, buzzer, αισθητήρας υπερήχων με αισθητήρα υγρασίας εδάφους, αισθητήρας σύγκρουσης, αισθητήρας παρακολούθησης γραμμής κ.λπ.



Τα ενεργά στοιχεία (αισθητήρες, κινητήρες, servo)

Τα δομικά (παθητικά) στοιχεία




Το πρόσθετο
στο Mind+
για NEZHA
inventor's kit
& S1,S2
αισθητήρες



Ένα πρόσθετο που εφοδιάζει το περιβάλλον οπτικοποιημένου προγραμματισμού Mind+ με μια **βιβλιοθήκη πλακιδίων εντολών για τη συνεργασία με τον micro:bit** και τις περιφερειακές συσκευές (αισθητήρες, LED κ.λπ) που βρίσκονται στο NEZHA inventor's Kit, συμπεριλαμβανομένων και αισθητήρων από τα πακέτα S1, S2 με την αξιοποίηση του καλωδίου με δύο τύπους απολήξεων RJ11(4) <->RJ11(6).

Μπορούμε να «κατεβάσουμε» το plugin
από την ηλεκτρονική διεύθυνση:

<https://portal.stem.edu.gr/courses/mind-nezha-inventor-s-kit/>

Αρχικά 5 απλές ενέργειες για την εγκατάσταση του πρόσθετου (extension), που εφοδιάζει το Mind+ με μια βιβλιοθήκη οπτικοποιημένων εντολών που οδηγούν micro:bit με NEZHA Hub και τον εξοπλισμό Inventor's Kit συμπεριλαμβανομένων και αισθητήρων από τα πακέτα S1,S2 με την αξιοποίηση του καλωδίου RJ11(4) <->RJ11(6).

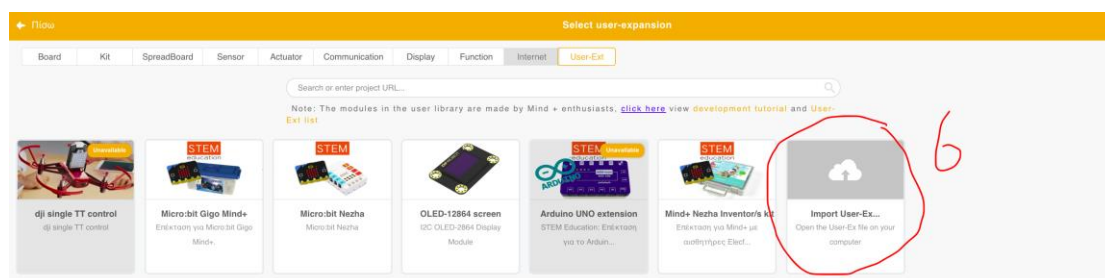
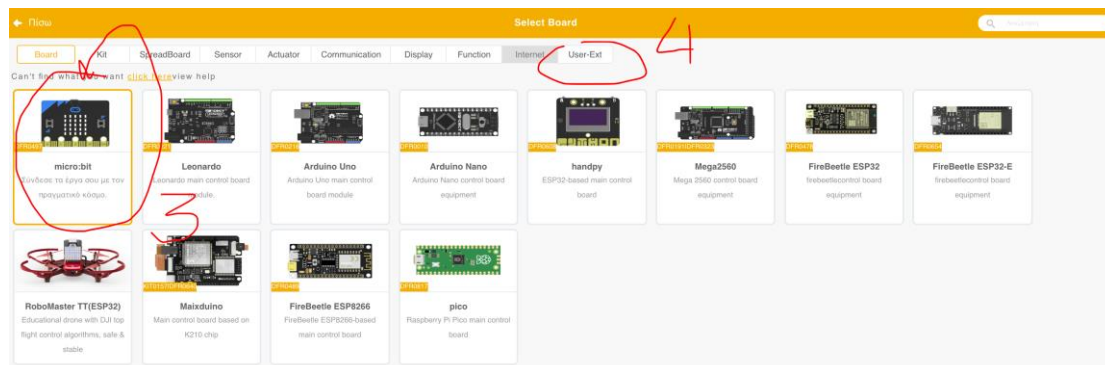
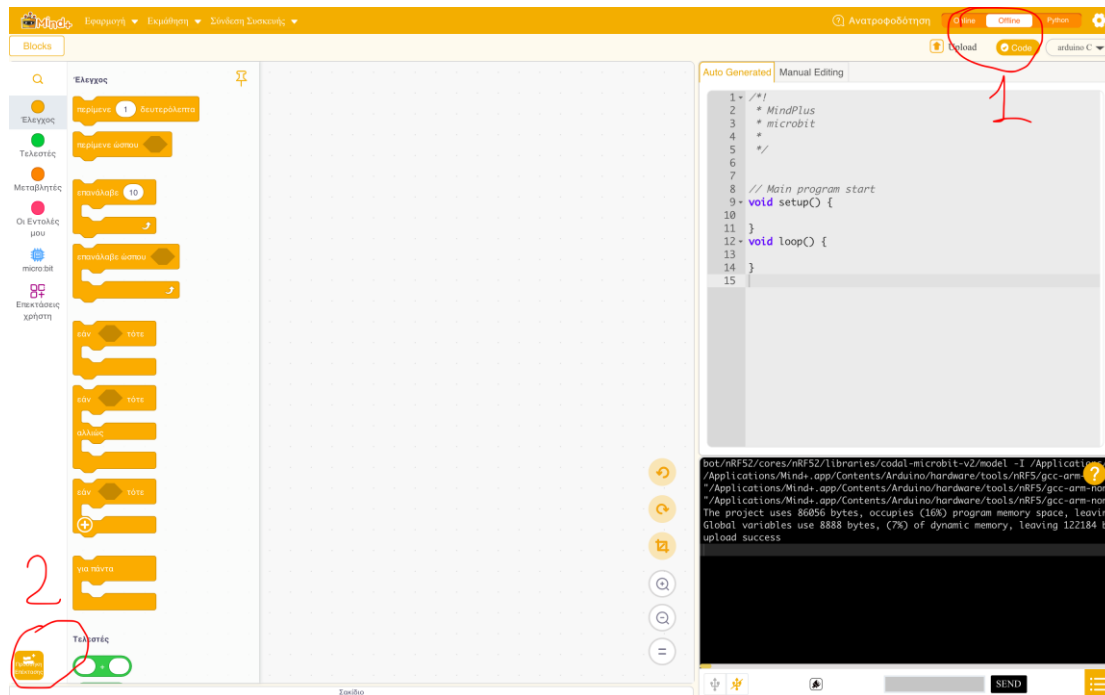
1. Κατέβασμα και αποσυμπίεση του συνημμένου αρχείου .zip (που περιέχει το extension)
2. Άνοιγμα του Mind+
3. Ρύθμιση σε off line mode
4. Επιλογή μικροεπεξεργαστή: microbit
5. Εισαγωγή πρόσθετου (extension) αναζητώντας το, στο σημείο που το «κατεβάσατε και αποθηκεύσατε.

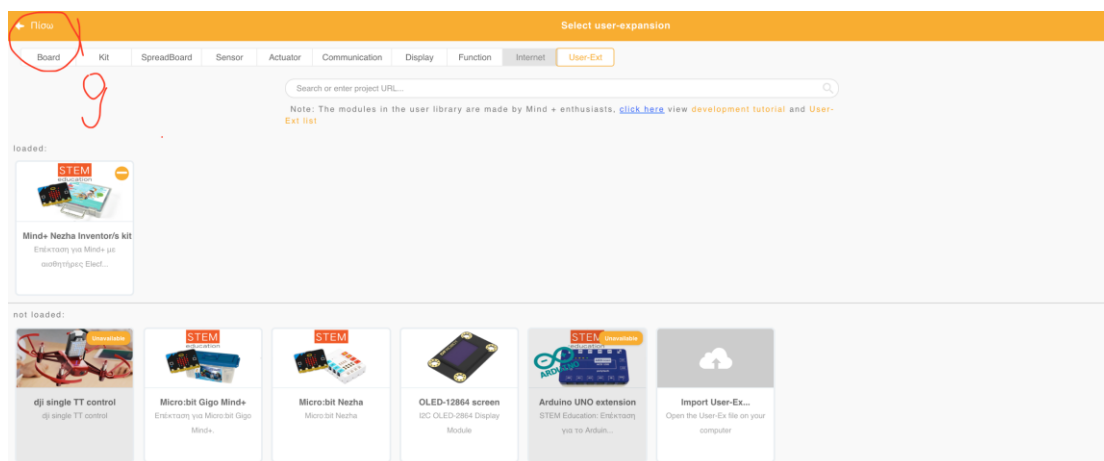
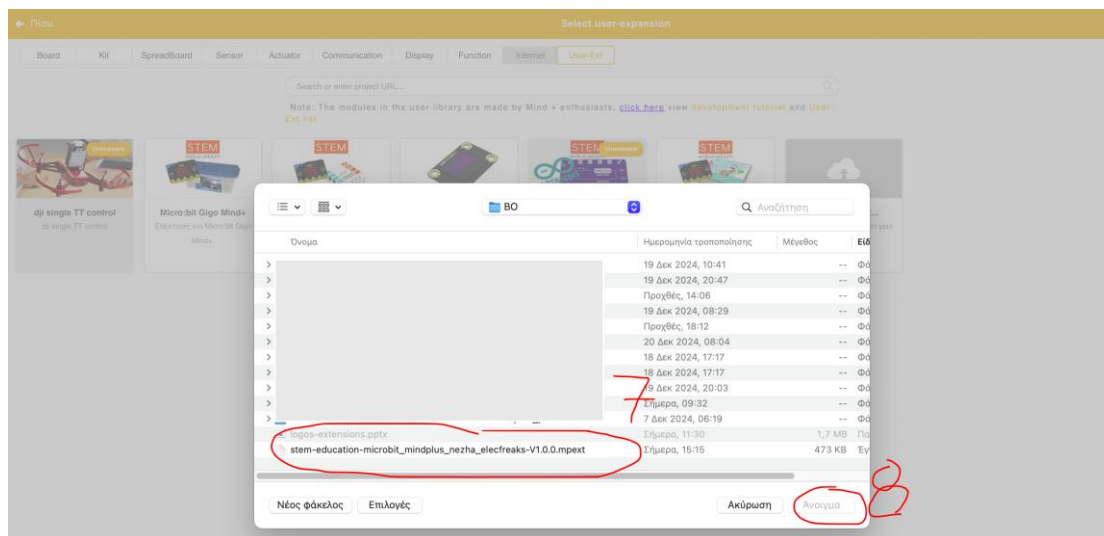


Οδηγίες εγκατάστασης του πρόσθετου

Εφόσον ...

- έχουμε κατεβάσει κι εγκαταστήσει το Mind+ στον Υπολογιστή μας
- έχουμε κατεβάσει και αποσυμπίεσει σε κάποιο σημείο του Υπολογιστή μας το πρόσθετο.
- Ανοίγουμε το Mind+ και ακολουθούμε τα παρακάτω απλά βήματα:



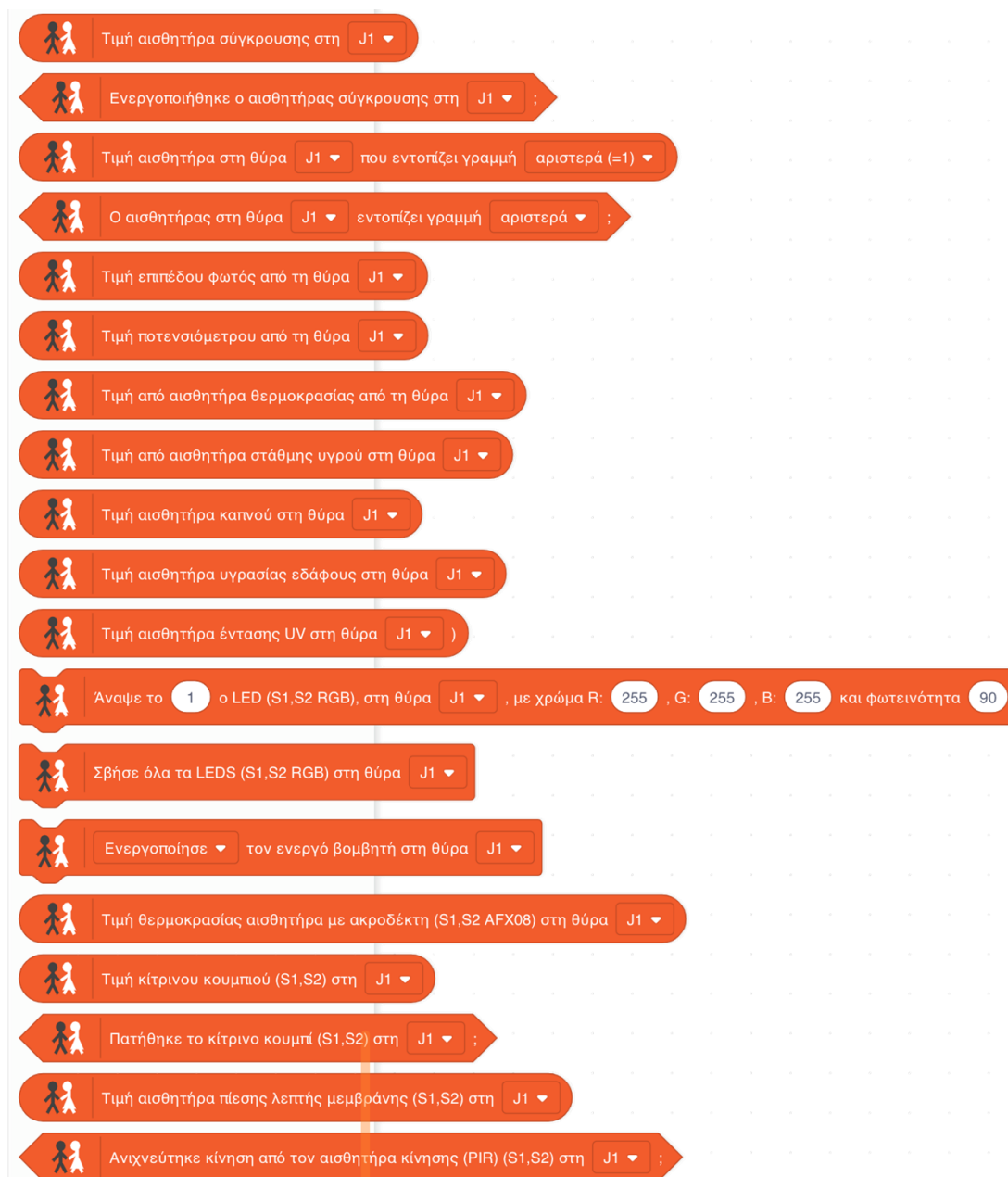


Οι εντολές της v.1.0.1

► STEM: Mind+ Nezha Inventor/s kit

- Ενεργοποίησε κινητήρα M1 με ταχύτητα 50 %
- Σταμάτησε κινητήρα M1
- Σταμάτησε όλους τους κινητήρες
- Όρισε 180 στο Servo S1 γωνία 90 °
- Όρισε στον 360° Servo S1 ταχύτητα 100 %
- Ενεργοποίησε το LED στη θύρα J1
- Ρύθμισε τη φωτεινότητα του LED panel στο micro:bit σε 0
- Άναψε όλα τα LEDs (RGB), στη θύρα J1, με χρώμα R: 255, G: 255, B: 255 και φωτεινότητα 90
- Άναψε το 1 ο από τα 8 LEDs (RGB), στη θύρα J1, με χρώμα R: 255, G: 255, B: 255 και φωτεινότητα 90
- Σβήσε όλα τα LEDs (RGB) στη θύρα J1
- Τιμή αισθητήρα απόστασης (υπερήχων) στη θύρα J1 σε cm
- Τιμή αισθητήρα σύγκρουσης στη J1
- Ενεργοποιήθηκε ο αισθητήρας σύγκρουσης στη J1 ;
- Τιμή αισθητήρα στη θύρα J1 που εντοπίζει γραμμή αριστερά (=1)
- Ο αισθητήρας στη θύρα J1 εντοπίζει γραμμή αριστερά ;
- Τιμή επιπέδου φωτός από τη θύρα J1
- Τιμή ποτενσιόμετρου από τη θύρα J1





NEZHA
&
S1,S2
Παραδείγματα
με Mind+



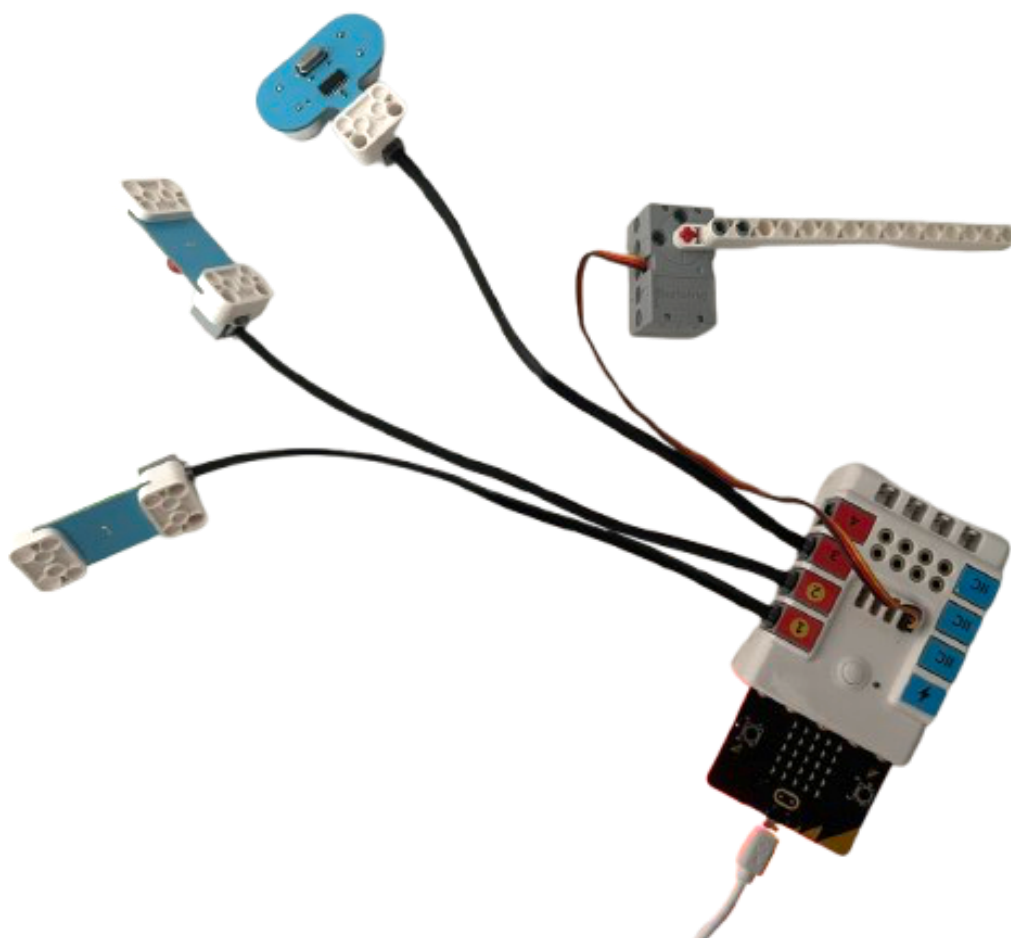
1ο παράδειγμα: Μπάρα χώρου στάθμευσης με NEZHA και micro:bit στο Mind+

A. Σενάριο

Να συνδέσουμε και να λειτουργήσουμε όσο το δυνατόν περισσότερα IoTs στο **NEZHA το micro:bit (όπως αισθητήρες elecfreaks)** και να τους προγραμματίσουμε με **Mind+**

Συγκεκριμένα, με αν ο αισθητήρας απόστασης αναγνωρίσει κινούμενο όχημα σε λιγότερο από 10 εκατοστά να ανοίξει η μπάρα ανάβοντας το πράσινο LED. Διαφορετικά να κλείσει η μπάρα ανάβοντας ένα κόκκινο LED που υποδηλώνει και με φωτεινή ένδειξη την απαγόρευση εισόδου..

B. Συνδεσμολογία



Γ. Κώδικας

The screenshot shows the Mind+ IDE interface. On the left is a 'Blocks' palette with various sensors and actuators. The main workspace contains a block-based program starting with 'Εκκίνηση micro:bit' (Start micro:bit), followed by a 'για πάντα' (Forever) loop. Inside the loop, there is an 'εάν' (If) block checking 'Τιμή αισθητήρα απόστασης (υπερήχων) στη θύρα J3 σε cm < 10' (Distance sensor value at door J3 in cm < 10). If true, it sets servo S1 to 0°, turns LED J1 on, and turns LED J2 off. If false, it sets servo S1 to 90°, turns LED J2 on, and turns LED J1 off. On the right, the 'Auto Generated' C++ code is displayed, showing the setup and loop functions for the servo and LEDs.

```

4  #
5  #
6  #include <NeZha.h>
7  #include <B0_elecfeaks.h>
8  // Create an object
9  NeZha nezha=NeZha();
10
11
12 // Main program start
13 void setup() {
14   pinMode(13, OUTPUT);
15   pinMode(14, INPUT);
16   pinMode(1, OUTPUT);
17   pinMode(2, OUTPUT);
18 }
19 void loop() {
20   if ((getDistance(13,14)<10)) {
21     nezha.set_servo_angle(360,0,0);
22     digitalWrite(1, HIGH);
23     digitalWrite(2, LOW);
24   }
25   else {
26     nezha.set_servo_angle(360,0,90);
27     digitalWrite(2, HIGH);
28     digitalWrite(1, LOW);
29   }
30 }
  
```

This block shows a detailed view of the block-based program logic. It starts with a 'Ξεκίνησε micro:bit' (Start micro:bit) block. This is followed by a 'για πάντα' (Forever) loop. Inside the loop, there is an 'εάν' (If) block with the condition 'Τιμή αισθητήρα απόστασης (υπερήχων) στη θύρα J3 σε cm < 10' (Distance sensor value at door J3 in cm < 10). If the condition is true, the following blocks are executed: 'Όρισε 360 στο Servo S1 γωνία 0°' (Set 360 to Servo S1 angle 0°), 'Ενεργοποίησε το LED στη θύρα J1' (Turn on LED at door J1), and 'Απενεργοποίησε το LED στη θύρα J2' (Turn off LED at door J2). If the condition is false (αλλιώς), the following blocks are executed: 'Όρισε 360 στο Servo S1 γωνία 90°' (Set 360 to Servo S1 angle 90°), 'Ενεργοποίησε το LED στη θύρα J2' (Turn on LED at door J2), and 'Απενεργοποίησε το LED στη θύρα J1' (Turn off LED at door J1).



2ο

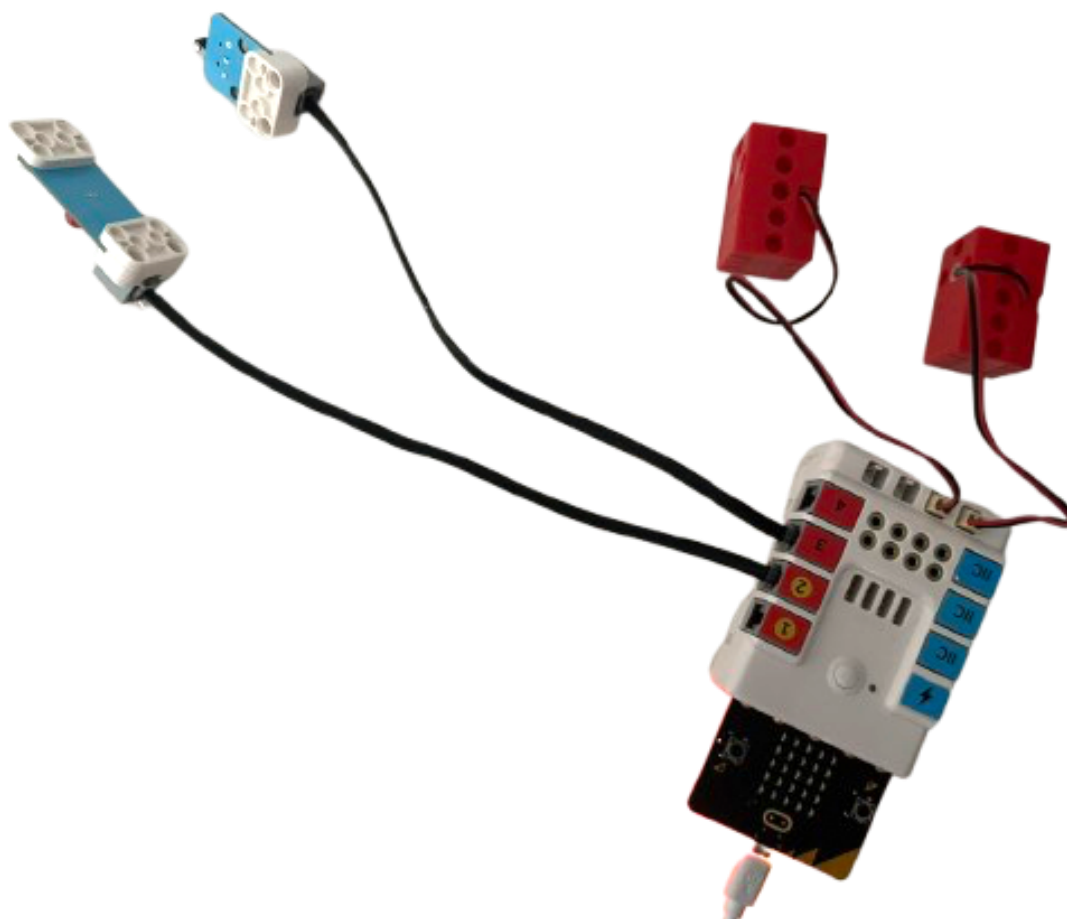
παράδειγμα: Σταματώντας τους κινητήρες

A. Σενάριο

Να συνδέσουμε και να λειτουργήσουμε όσο το δυνατόν περισσότερα IoTs στο **NEZHA το micro:bit** (όπως αισθητήρες **elecfreaks**) και να τους προγραμματίσουμε με **Mind+**

Συγκεκριμένα, με το πάτημα του ελάσματος του αισθητήρα σύγκρουσης

B. Συνδεσμολογία



Γ. Κώδικας

The screenshot shows the Mind+ IDE interface. On the left is a 'Blocks' palette with various sensors and actuators. The main workspace contains a block-based program. On the right, the 'Auto Generated' code tab shows the corresponding C++ code for the blocks.

```

1 //
2 * MindPlus
3 * microbit
4 *
5 */
6 #include <NeZha.h>
7 // Create an object
8 NeZha nezha=NeZha();
9
10
11 // Main program start
12 void setup() {
13   pinMode(14, INPUT);
14   pinMode(2, OUTPUT);
15 }
16 void loop() {
17   if ((digitalRead(14)==1)) {
18     nezha.set_motor_speed(0,50);
19     nezha.set_motor_speed(1,50);
20     digitalWrite(2, LOW);
21   }
22   else {
23     nezha.stop_all_motor();
24     digitalWrite(2, HIGH);
25   }
26 }
27
  
```

This image provides a detailed view of the block-based program logic. The program starts with a 'Ξεκίνησε micro:bit' (Start micro:bit) block, followed by a 'για πάντα' (Forever) loop. Inside the loop, an 'εάν' (If) block checks if the 'Τιμή αισθητήρα σύγκρουσης στη J3' (Collision sensor value at J3) is equal to 1. If true, it executes three blocks: 'Ενεργοποίησε κινητήρα M1 με ταχύτητα 50 %', 'Ενεργοποίησε κινητήρα M2 με ταχύτητα 50 %', and 'Απενεργοποίησε το LED στη θύρα J2'. If false ('αλλιώς'), it executes two blocks: 'Σταμάτησε όλους τους κινητήρες' (Stop all motors) and 'Ενεργοποίησε το LED στη θύρα J2'.



3^ο

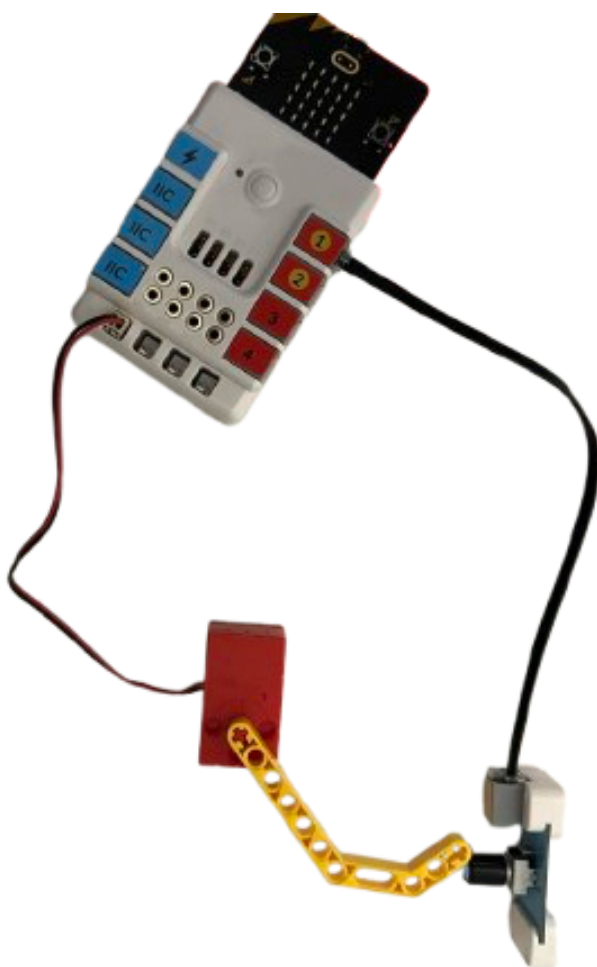
παράδειγμα: Περιστροφή κινητήρων με το ΠΟΤΕΝΣΙΟΜΕΤΡΟ

A. Σενάριο

Να συνδέσουμε και να λειτουργήσουμε όσο το δυνατόν περισσότερα IoTs στο **NEZHA το micro:bit (όπως αισθητήρες elecfreaks)** και να τους προγραμματίσουμε με Mind+

Συγκεκριμένα, με το πάτημα θα περιστρέφουμε ένα κινητήρα με ταχύτητα και φορά που καθορίζουμε από το ποτενσιόμετρο. Στο εργαστήριο θα μπορούσαμε να δημιουργήσουμε τον αυτοματισμό κίνησης ενός ηλεκτρικού οχήματος χωρίς να έχουμε ανάγκη τα πεντάλ.

B. Συνδεσμολογία



Γ. Κώδικας

The screenshot shows the Mind+ IDE interface. On the left is a 'Blocks' palette with various sensors and actuators. The main workspace contains a block-based program. On the right, the 'Auto Generated' tab shows the corresponding C++ code:

```

1 //
2 * MindPlus
3 * microbit
4 *
5 */
6 #include <NeZha.h>
7 #include <Arduino.h>
8 // Create an object
9 NeZha nezha=NeZha();
10
11
12 // Main program start
13 void setup() {
14
15 }
16 void loop() {
17   nezha.set_motor_speed(0,map(analogRead(1), 0, 1023, -100, 100));
18 }
19

```

Below the code, a terminal window shows the upload success message.

The screenshot shows the block-based program in the Mind+ IDE. The program starts with a 'when green flag clicked' block, followed by a 'when micro:bit starts' block. The main loop consists of a 'set motor speed' block with the following configuration:

- Motor: M1
- Direction: με ταχύτητα (with speed)
- Speed: αντιστοίχισε (map)
- Map range: Τμή ποτενσιόμετρου από τη θύρα J1 (Potentiometer value from port J1)
- Map range: από 0 , 1023 | σε [-100 , 100] %

Χρησιμοποιούμε την λειτουργία **[αντιστοίχισε (map)]** για να αντιστοιχίσουμε τις τιμές δύο λιστών.

1. Η πρώτη λίστα τιμών 0 - 1023 είναι οι τιμές που επιστρέφει το ποτενσιόμετρο
2. Η δεύτερη λίστα -100 – 100 είναι οι τιμές (ποσοστό) με τις οποίες αλλάζει η φορά και η ένταση περιστροφής του κινητήρα.



Περισσότερες δραστηριότητες με NEZHA

Σαράντα πέντε (45) δραστηριότητες/κατασκευές και πολλά περισσότερα στοιχεία για τον προγραμματισμό με Make Code και το NEZHA με micro:bit θα βρείτε στο βιβλίο που μπορείτε να «κατεβάσετε» από τη διεύθυνση:

https://portal.stem.edu.gr/courses/bo_book_v01/

micro:bit
Microsoft
MakeCode
Mind+
STEM
education

Προσεγγίζοντας **διαθεματικά**
αυτοματισμούς γύρω μας
και τον **προγραμματισμό** υπολογιστών
v.01

Βασίλης Οικονόμου, 2023-24



4^ο

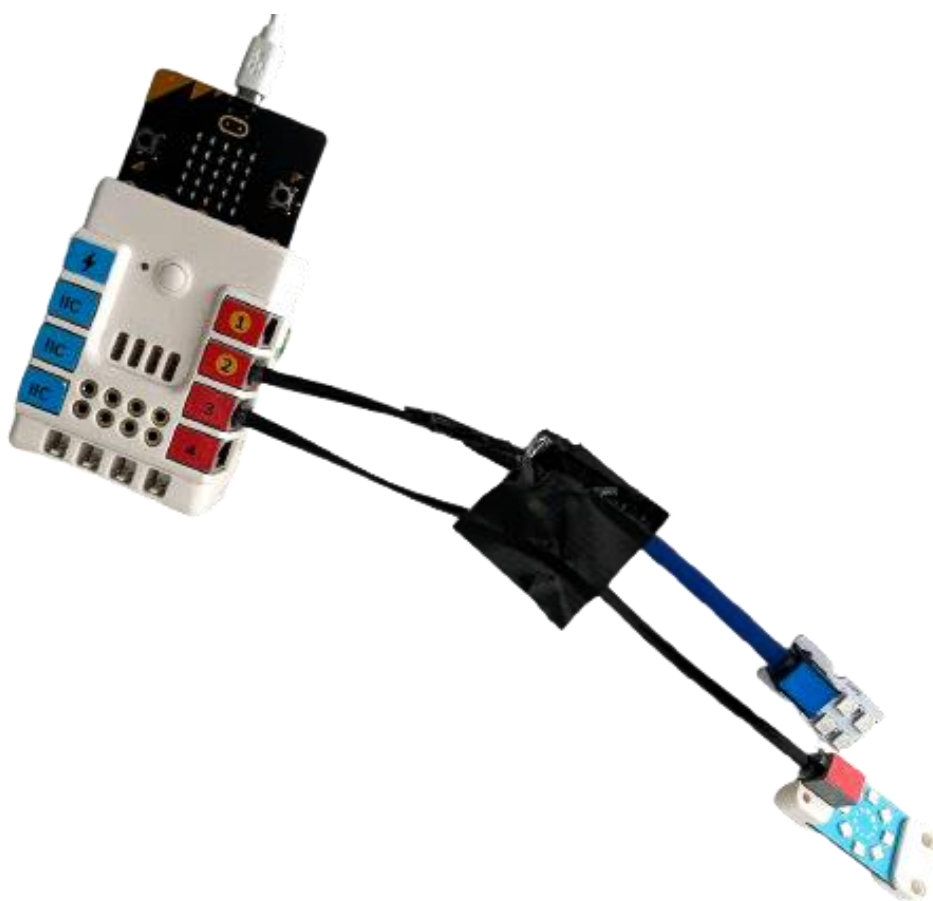
παράδειγμα: Ανάβοντας NeoPixel LEDs

A. Σενάριο

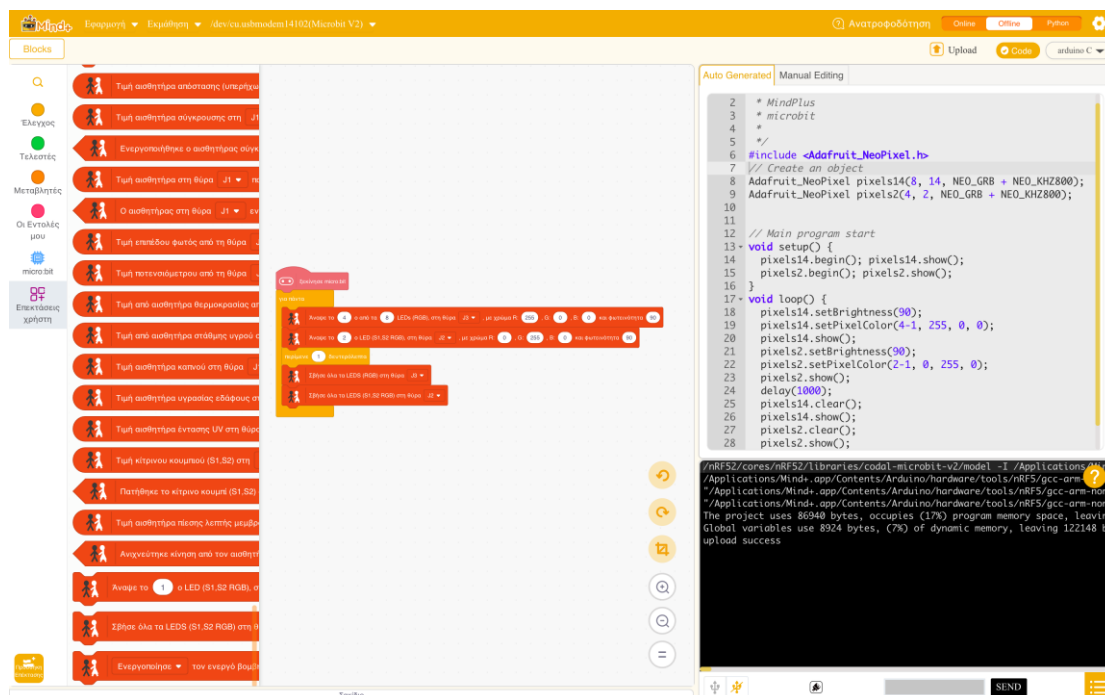
Να συνδέσουμε και να λειτουργήσουμε όσο το δυνατόν περισσότερα IoTs στο **NEZHA το micro:bit** (όπως αισθητήρες **elecfreaks** συμπεριλαμβανομένων και αυτών από τα S1 και S2 πακέτα) και να τους προγραμματίσουμε με **Mind+**

Συγκεκριμένα, θα ανάβουμε και θα σβήνουμε ένα pixel από το NeoPixel RGB με 8 leds της elecfreaks και ένα pixel από το NeoPixel RGB με 4 leds της polyteck με κόκκινο και πράσινο χρώμα αντίστοιχα και με ένταση φωτεινότητας 90 στα 255.

B. Συνδεσμολογία



Γ. Κώδικας

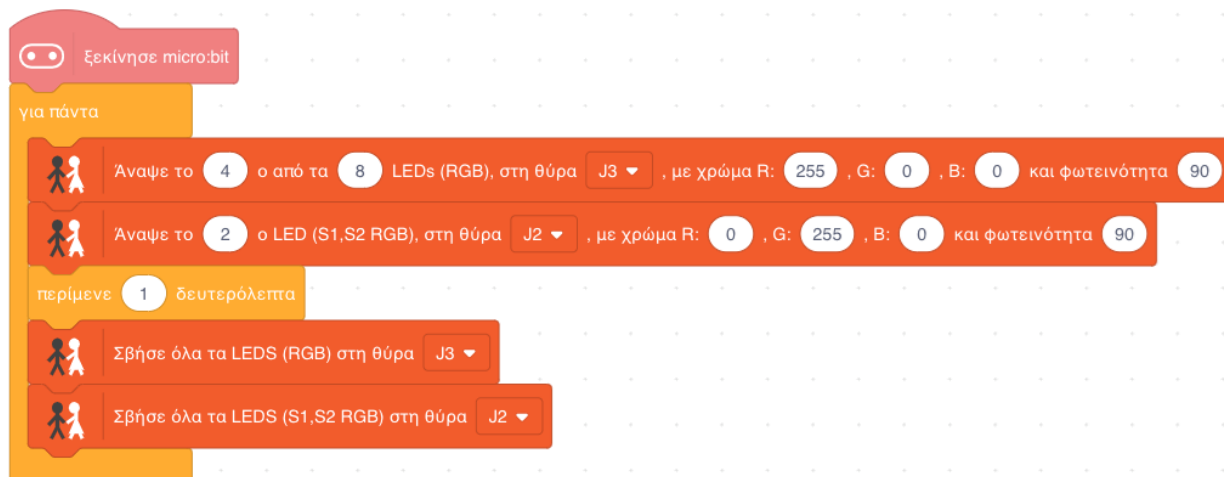


Στην αρχή χρησιμοποιούμε δύο εντολές:

- η πρώτη οδηγεί το NeoPixel RGB με 8 leds της elecfreaks
- η δεύτερη οδηγεί το NeoPixel RGB με 4 leds της polyteck

Στη συνέχεια περιμένουμε με αναμμένα LEDs

...Και με τις δύο επόμενες εντολές σβήνουμε όλα τα LEDs



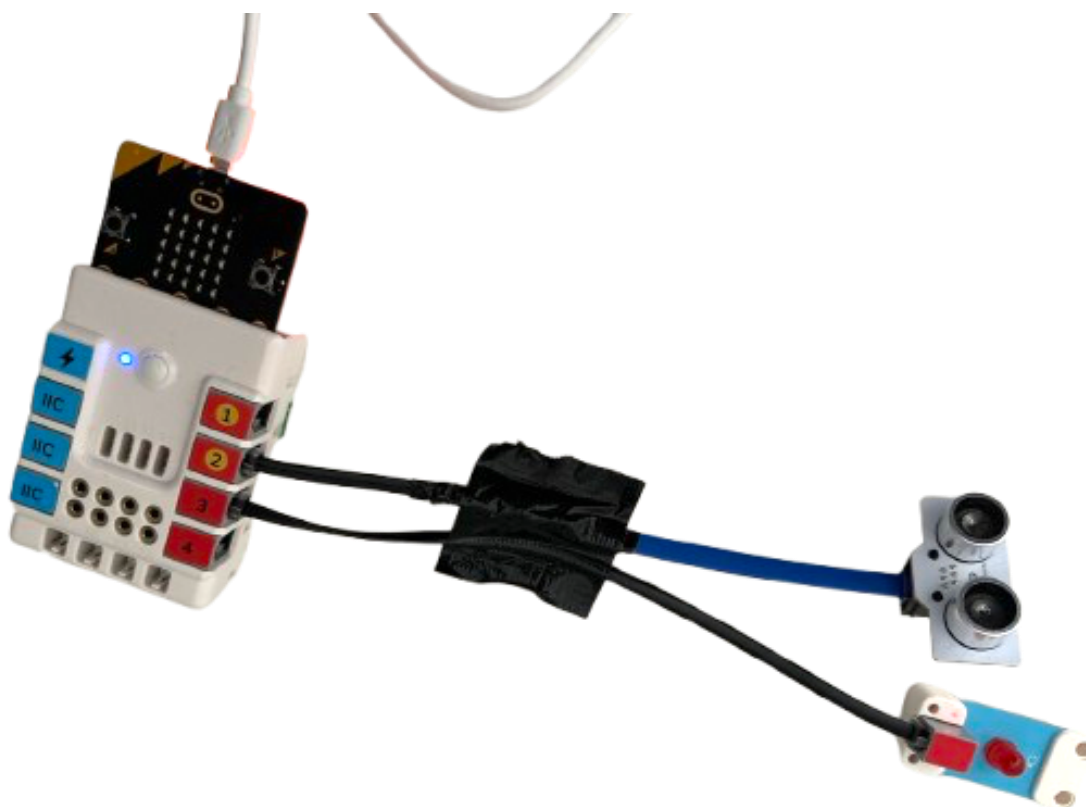
5^ο

παράδειγμα: Ανάβοντας elecfreaks LED, με
τιμές από αισθητήρα υπερήχων/απόστασης polyteck

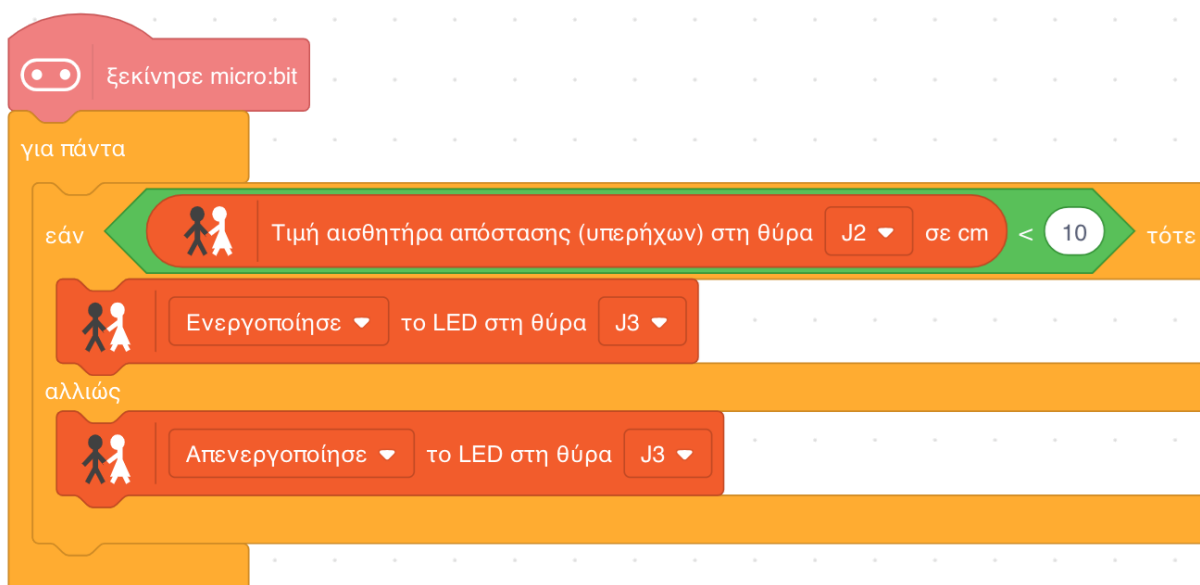
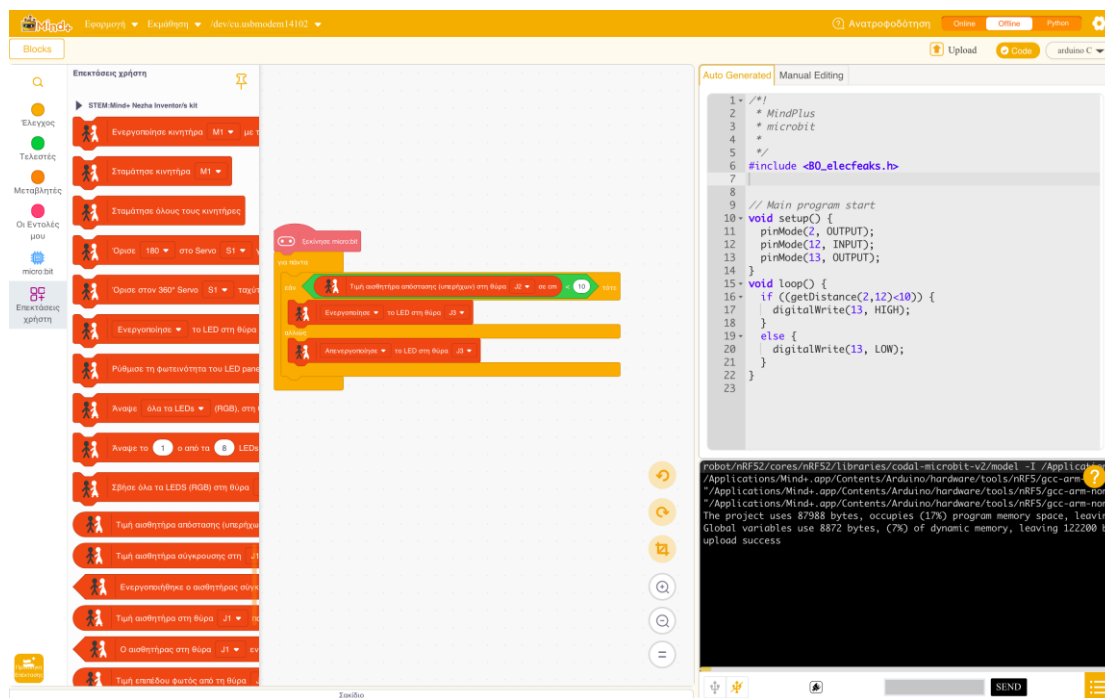
A. Σενάριο

Να συνδέσουμε και να λειτουργήσουμε όσο το δυνατόν περισσότερα IoTs στο **NEZHA το micro:bit** (όπως αισθητήρες elecfreaks συμπεριλαμβανομένων και αυτών από τα S1 και S2 πακέτα) και να τους προγραμματίσουμε με Mind+

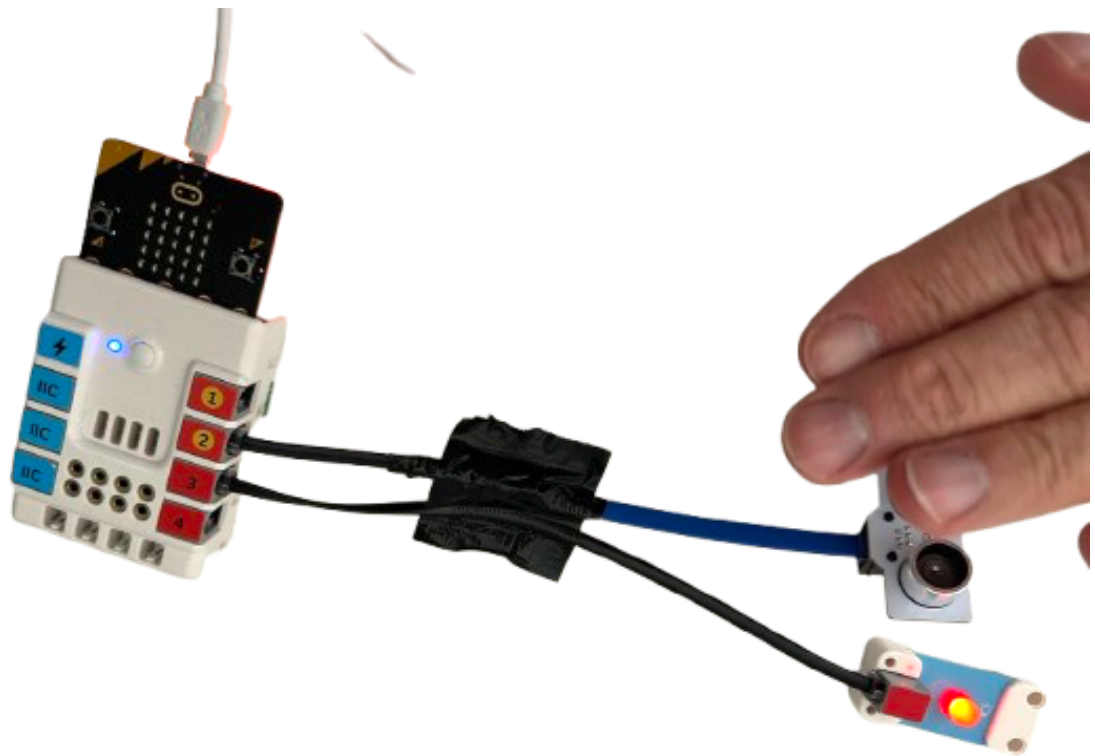
Συγκεκριμένα, αν ο αισθητήρας υπερήχων/απόστασης της Polyteck εντοπίσει ένα εμπόδιο σε κοντινότερη απόσταση από 10 εκατοστά θα ανάψει το κόκκινο LED της elecfreaks

B. Συνδεσμολογία

Γ. Κώδικας



Δ. Το αποτέλεσμα



6η δραστηριότητα: Φανός με Neorixel, αισθητήρες σύγκρουσης (αντί κουμπιού), και κίνησης

A. Στοιχεία Δραστηριότητας

Οι Μαθητές/τριες θα σχεδιάσουν και θα υλοποιήσουν μία λάμπα φωτισμού όπου με αισθητήρα κίνησης ή διακόπτη θα ανάβει σε διαφορετικά χρώματα και ένταση. Θα αξιοποιήσουν παθητικά και ενεργά στοιχεία, ιδιαίτερα αισθητήρες και Neorixel LED, συνδυάζοντας δομές προγραμματισμού, ώστε να επιτύχουν το επιθυμητό αποτέλεσμα

Σκοπός

Η ενίσχυση της κατανόησης των Μαθητών/τριών για τις βασικές αρχές αυτοματισμών, ενθαρρύνοντας τη δημιουργικότητα, τον προγραμματισμό, και τη συνεργατική εργασία, αποτελούν τους διδακτικούς στόχους της δραστηριότητας αυτής. Πιο συγκεκριμένα η κατανόηση βασικών αρχών της ρομποτικής, όπως ο έλεγχος των κατασκευών. Η εξοικείωση με τη χρήση του Neorixel Led και η εκμάθηση των βασικών δομών και εντολών της πλατφόρμας προγραμματισμού του ARD:icon – Mind+ για τον έλεγχο της κατασκευής. Η σχεδίαση της βάσης της λάμπας με συγκεκριμένα κριτήρια, όπως η ισορροπία, και ο προγραμματισμός για τον έλεγχο της κατασκευής. Η αντιμετώπιση προβλημάτων κατά τη δημιουργία, τον προγραμματισμό και τη λειτουργία της λάμπας, μέσα από την ομαλή συνεργασία και επικοινωνία με άλλους μαθητές για την επίτευξη κοινών στόχων. Τέλος η κοινή αξιολόγηση που θα οδηγήσει στην επιλογή των βέλτιστων λύσεων για το θέμα.

Στόχοι

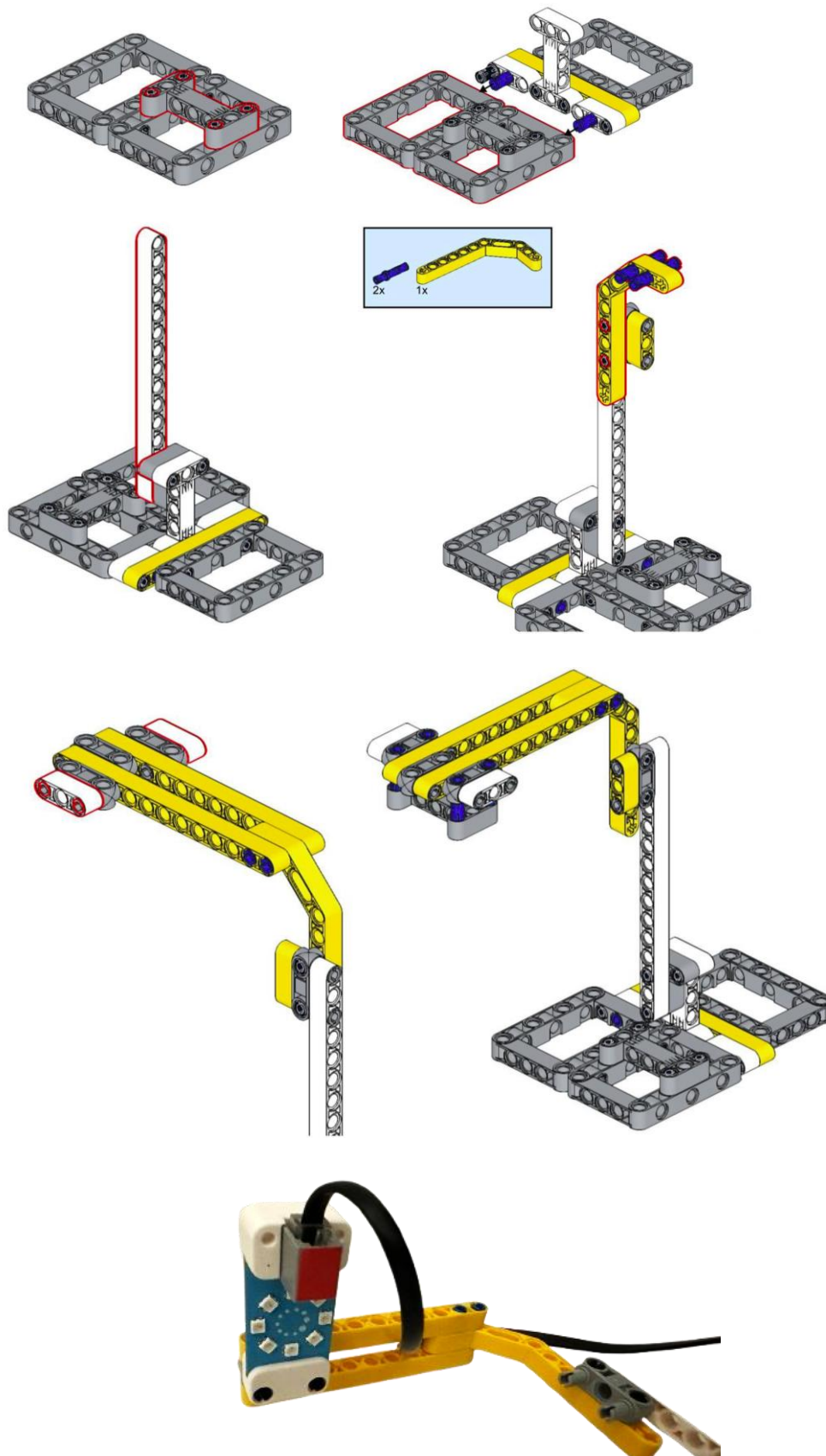
- **Ορίζω (Γνώση):** Τα θέματα της ισορροπίας των στελεχών που απαιτούνται για την κατασκευή μας.
- **Επιλέγω (Γνώση):** Τα κατάλληλα παθητικά και ενεργά στοιχεία που θα χρειαστώ για την κατασκευή μου.
- **Αναγνωρίζω (Γνώση):** Την αξιοποίηση της κατασκευής μας για τις καθημερινές μας ανάγκες και την εξοικονόμηση ενέργειας.
- **Περιγράφω λόγους για (Κατανόηση):** τους οποίους είναι σημαντική η χρήση της λάμπας αυτής στους δρόμους ή/και τις κατοικίες.
- **Συναρμολογώ (Εφαρμογή):** Τα παθητικά και τα ενεργά στοιχεία που απαιτούνται με γνώμονα την ισορροπία και την σταθερότητα της κατασκευής μου.
- **Αναφέρω συστατικά μέρη (Ανάλυση):** Της υπό κατασκευής λάμπας.
- **Καταρτίζω (Σύνθεση):** Σχέδιο για την λειτουργία και αξιοποίηση του αυτοματισμού μου
- **Προβλέπω (Ανάλυση):** Τον μέγεθος των στελεχών της κατασκευής, σύμφωνα πάντα με την αρχικό σχέδιο.
- **Προβληματίζομαι (Αξιολόγηση):** Για την όσο το δυνατόν καλύτερη αξιοποίηση των επιλογών έναρξης των κινήσεων του κάθε στελέχους της κατασκευής μου.
- **Τροποποιώ (Εφαρμογή):** Την κατασκευή μου ώστε να είναι όσο το δυνατόν πιο αποτελεσματική, σύμφωνα με τον αρχικό στόχο.
- **Αξιολογώ (Αξιολόγηση):** Τον τρόπο εφαρμογής και το αποτέλεσμα της λειτουργίας της μπάρας

Χρόνος μάθησης
2 ώρες.

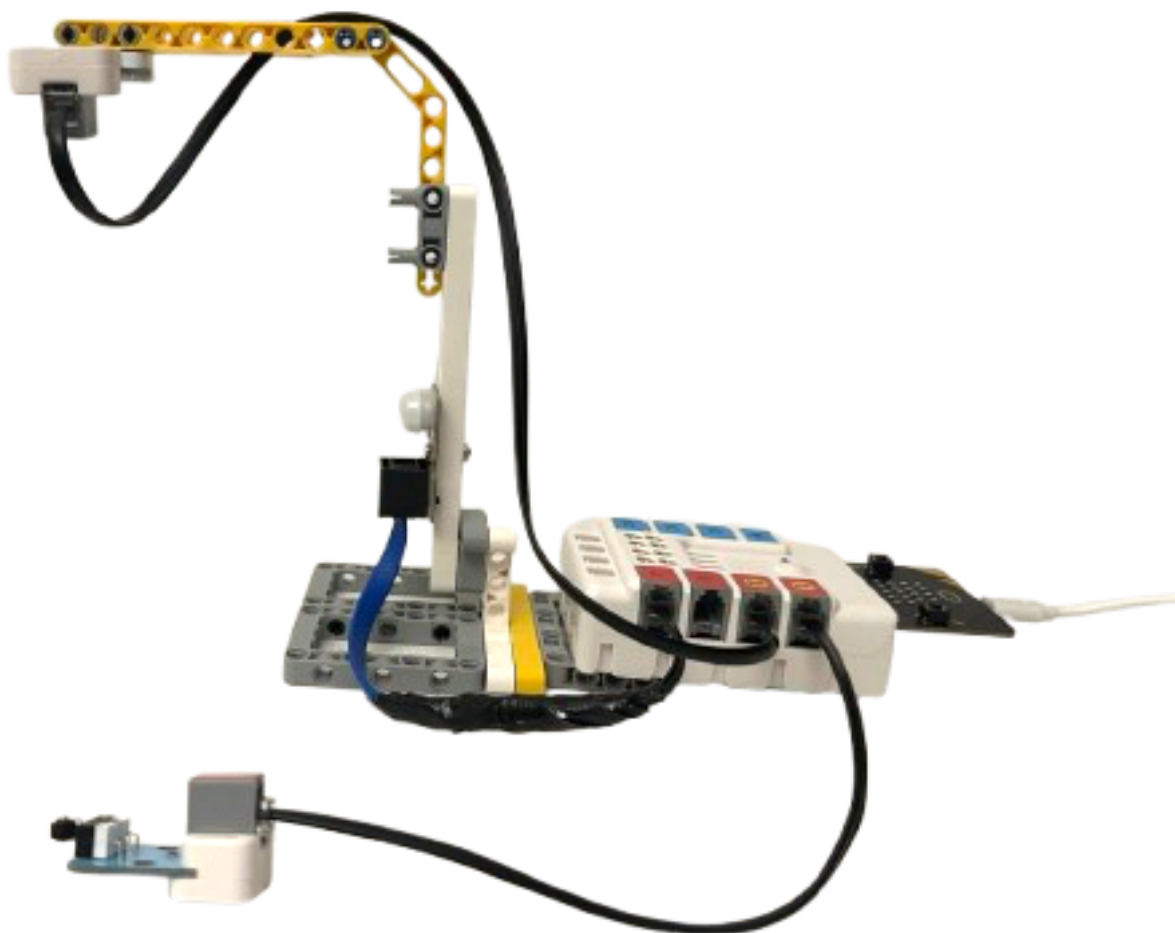
Μέγεθος τάξης
14 Μαθητές/τριες.



B. Κατασκευή

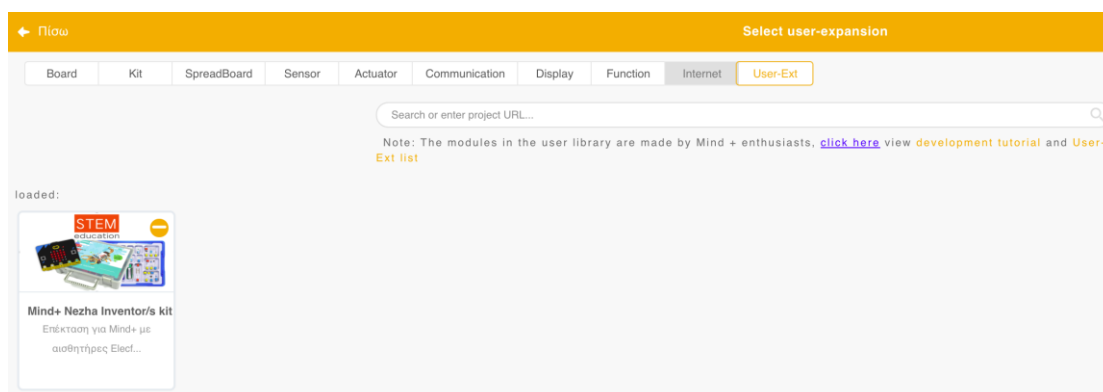


Γ. Συνδεσμολογία



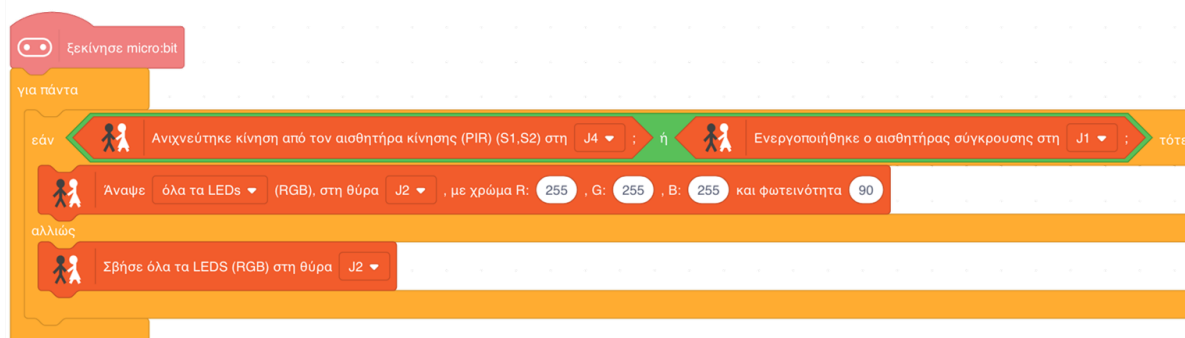
Στην κατασκευή μας αξιοποιούμε αισθητήρα ανίχνευσης κίνησης από το S1, S2 πακέτο με το ειδικό καλώδιο μετατροπής των απολήξεων των θυρών

Δ. Πρόσθετο (extension)



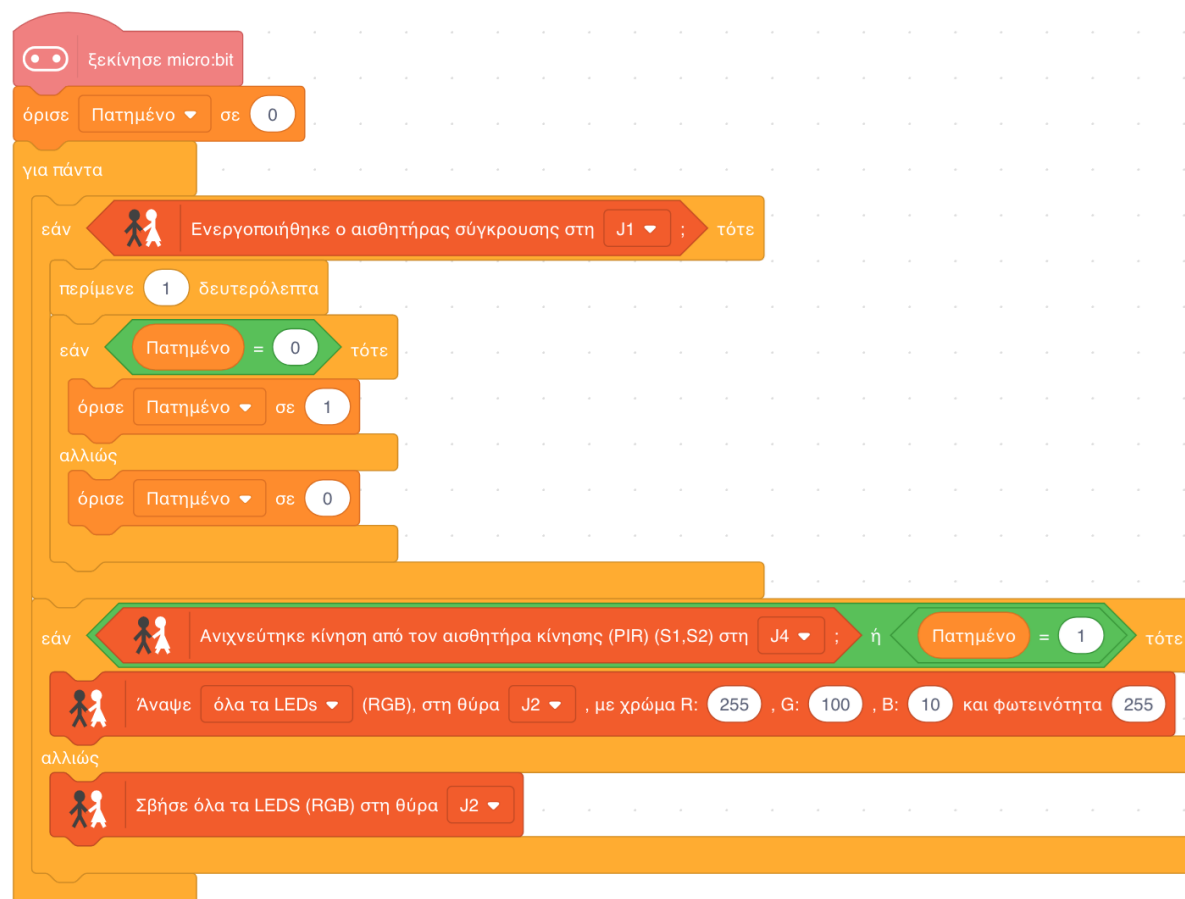
Ε. Κώδικας

Α' Προσπάθεια



Με τον κώδικα αυτό αν ανιχνευτεί κίνηση μπροστά από το φανό ή όσο μένει πατημένος ο αισθητήρας σύγκρουσης (αντί κουμπιού), θα ανάψουν και τα τέσσερα LEDs του Neorixel με λευκό χρώμα και μέτρια φωτεινότητα (90 στα 255).

Β' Προσπάθεια



Με τον κώδικα αυτό αν ανιχνευτεί κίνηση μπροστά από το φανό ή πατηθεί ο αισθητήρας σύγκρουσης (αντί κουμπιού), θα ανάψουν και τα οκτώ LEDs του Neorixel με ζεστό κίτρινο χρώμα και φωτεινότητα 255 στα 255. Ο φανός θα σβήσει όταν πατηθεί το κίτρινο κουμπί ξανά.





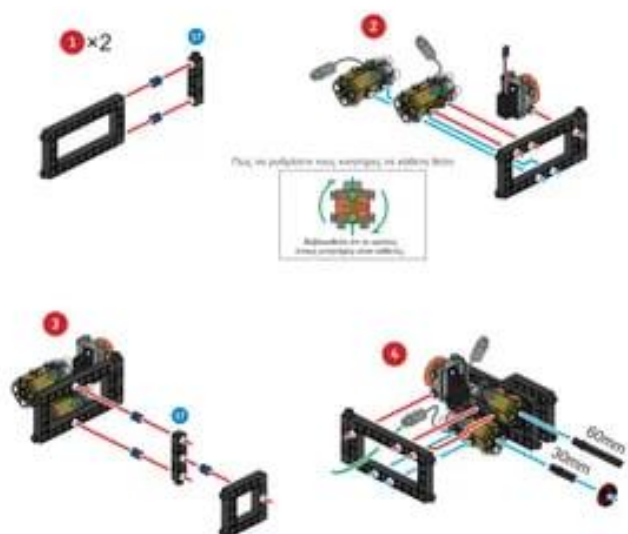
Εξοπλισμός Gigo



Ας ανοίξουμε το κουτί

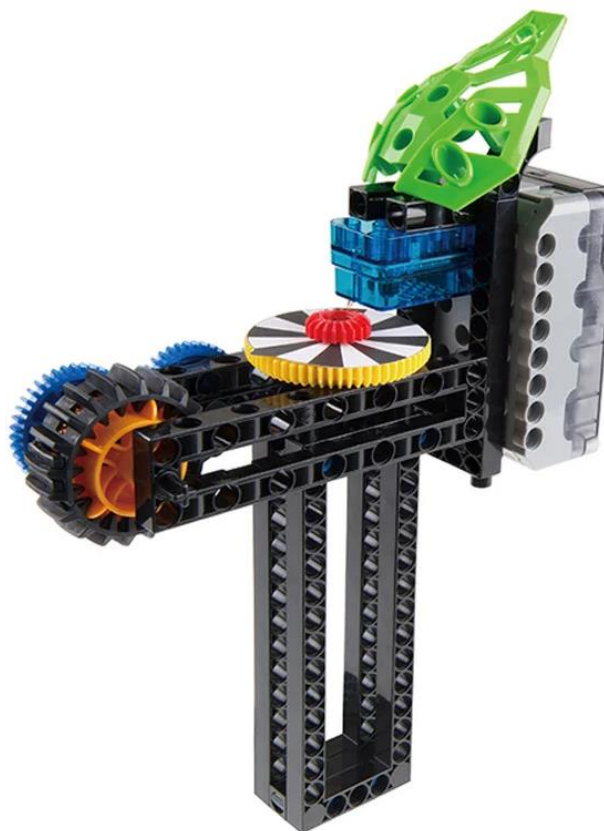
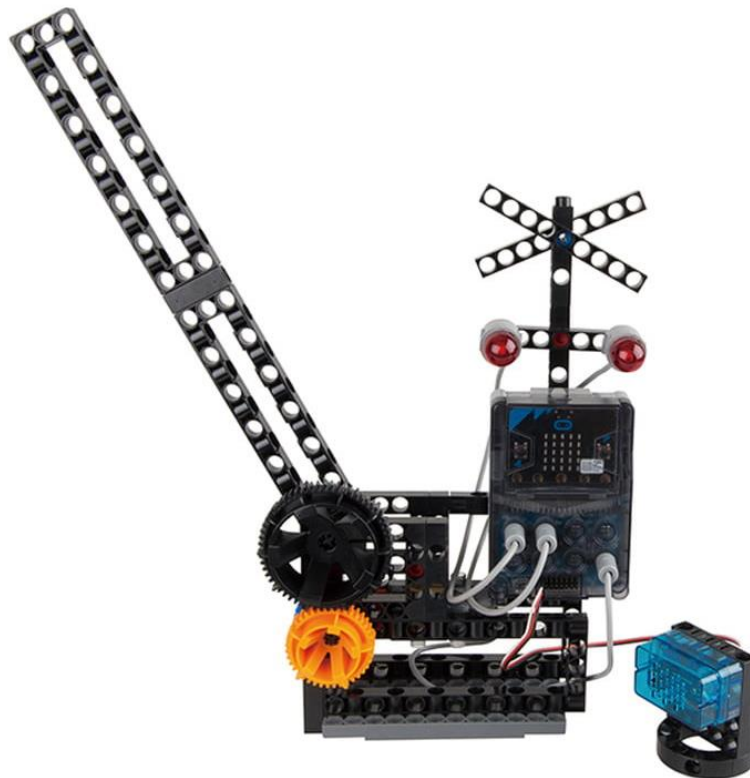


Το λεπτομερές υποστηρικτικό υλικό (βιβλίο)



Κατασκευές







Το χρήσιμο
πρόσθετο
στο Mind+
για Gigo
(micro:bit)



Ένα πρόσθετο που εφοδιάζει το περιβάλλον οπτικοποιημένου προγραμματισμού Mind+ με μια **βιβλιοθήκη πλακιδίων εντολών για τη συνεργασία με τον micro:bit** και τις περιφερειακές συσκευές (αισθητήρες, LED κ.λπ) που βρίσκονται στο Gigo Robots

Μπορούμε να «κατεβάσουμε» το plugin
από την ηλεκτρονική διεύθυνση:

<https://portal.stem.edu.gr/courses/stem-extension-mindplus-microbit-gigo/>

Αρχικά 5 απλές ενέργειες για την εγκατάσταση του πρόσθετου (extension), που εφοδιάζει το Mind+ με μια βιβλιοθήκη οπτικοποιημένων εντολών που οδηγούν τον εξοπλισμό Gigo (micro:bit)

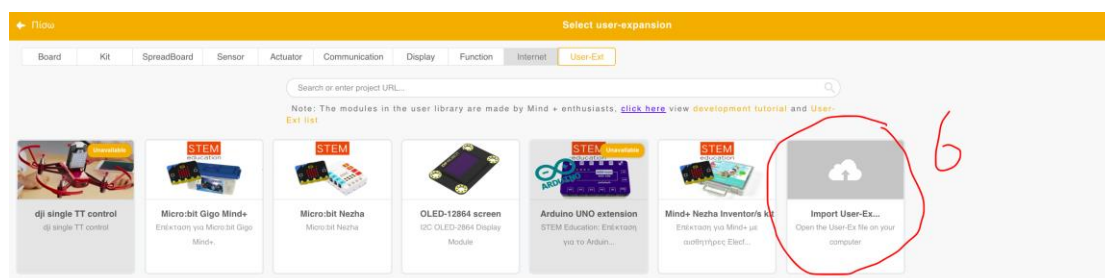
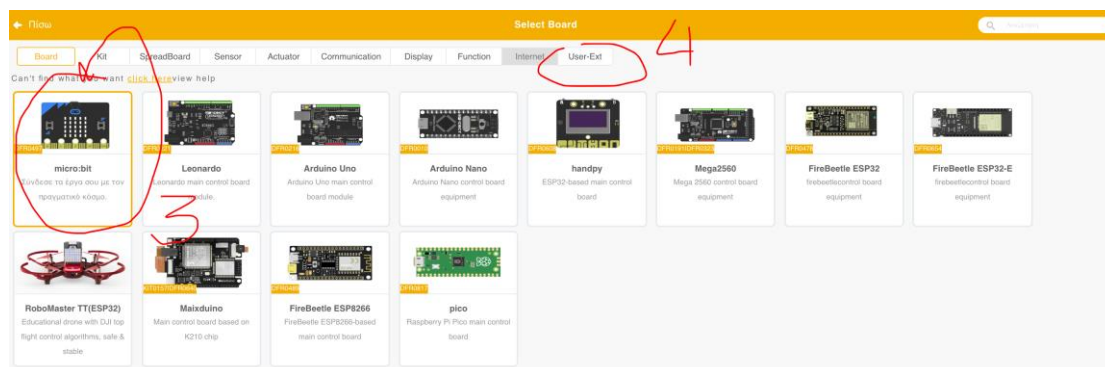
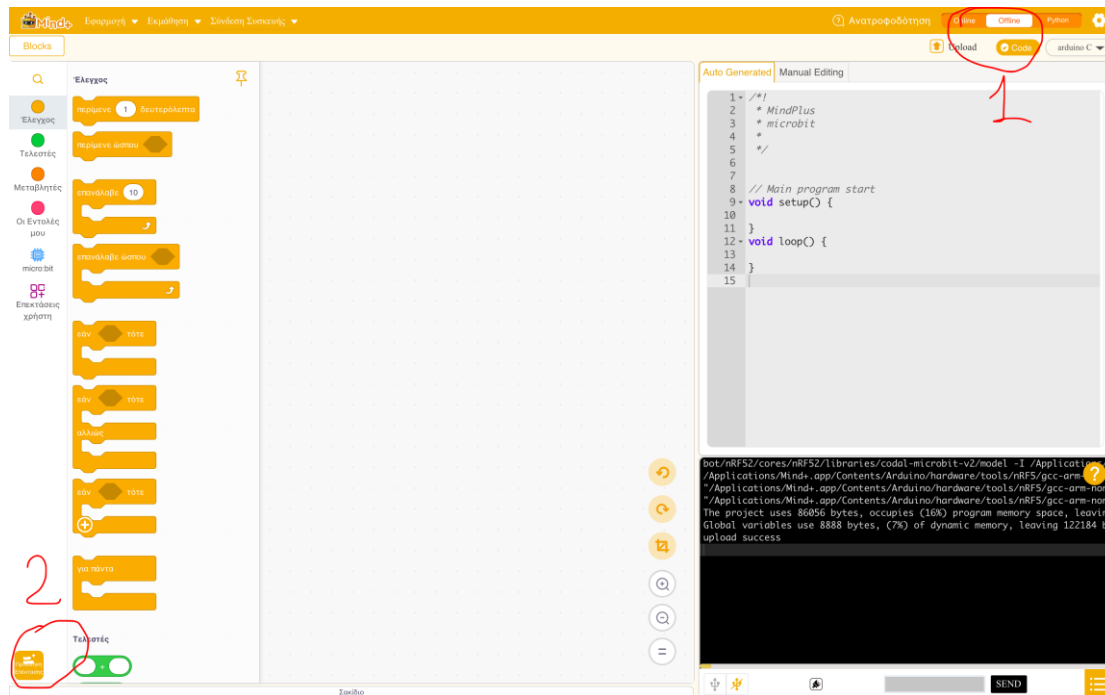
6. Κατέβασμα και αποσυμπίεση του συνημμένου αρχείου .zip (που περιέχει το extension)
7. Άνοιγμα του Mind+
8. Ρύθμιση σε off line mode
9. Επιλογή μικροεπεξεργαστή: microbit
10. Εισαγωγή πρόσθετου (extension) αναζητώντας το, στο σημείο που το «κατεβάσατε και αποθηκεύσατε.

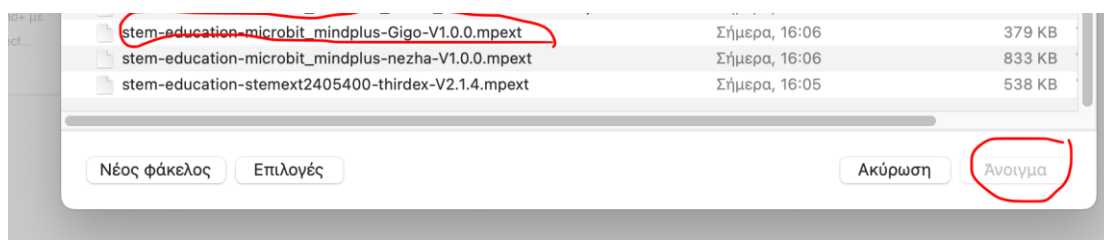


Οδηγίες εγκατάστασης του πρόσθετου

Εφόσον ...

- έχουμε κατεβάσει κι εγκαταστήσει το Mind+ στον Υπολογιστή μας
- έχουμε κατεβάσει και αποσυμπίεσει σε κάποιο σημείο του Υπολογιστή μας το πρόσθετο.
- Ανοίγουμε το Mind+ και ακολουθούμε τα παρακάτω απλά βήματα:





Οι εντολές της επέκτασης (v.1.0.0)

► STEM:Micro:bit Gigo Mind+

- Όταν πατηθεί το στο micro:bit
- Στρέψε το Servo στη θύρα σε γωνία
- Περιστροφή κινητήρα στη θύρα με κατεύθυνση και ταχύτητα
- Άναψε το LED στη θύρα
- Ρύθμισε τη φωτεινότητα του LED panel στο micro:bit σε
- Τιμή θερμοκρασίας από το micro:bit
- Τιμή πυξίδας από το micro:bit
- Τιμή επίπεδου φωτός από το micro:bit
- Το κουμπί πίεσης στη θύρα είναι πατημένο;
- Ο αισθητήρας γραμμής στη θύρα βλέπει τη γραμμή;
- Το κουμπί στο micro:bit είναι πατημένο;





Gigo Παραδείγματα



1^ο

παράδειγμα: Το άναμμα LED με κουμπί Gigo
(το πεντάλ του φρένου)

A. Σενάριο

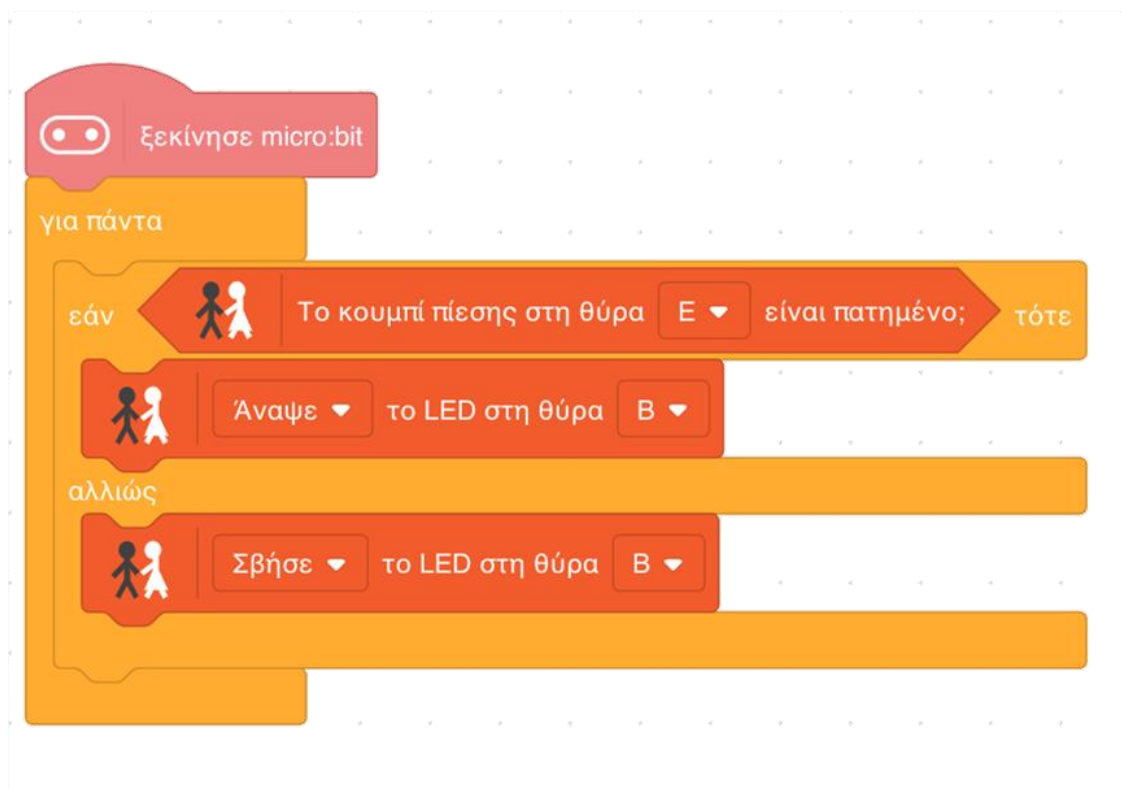
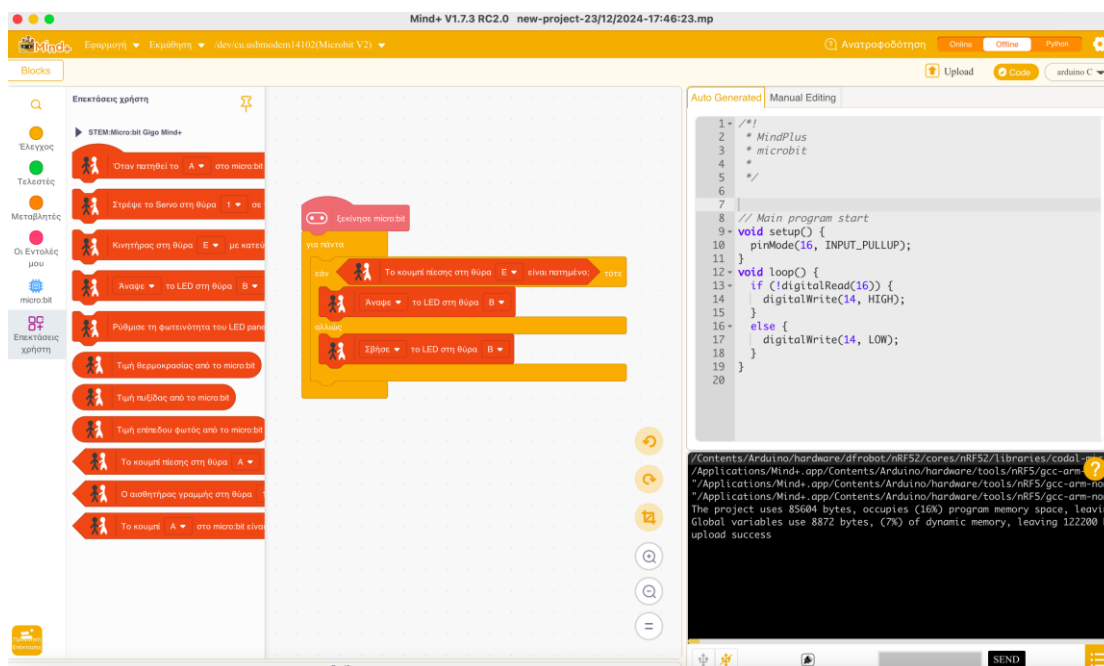
Να συνδέσουμε και να λειτουργήσουμε όσο το δυνατόν περισσότερα IoTs στο **Gigo με micro:bit**) και να τους προγραμματίσουμε με **Mind+**

Συγκεκριμένα, ανάβουμε το κόκκινο LED με το πάτημα του κουμπιού του Gigo

B. Συνδεσμολογία



Γ. Κώδικας



2^ο

παράδειγμα: Περιστροφή κινητήρων και σταμάτημα υπό συνθήκες

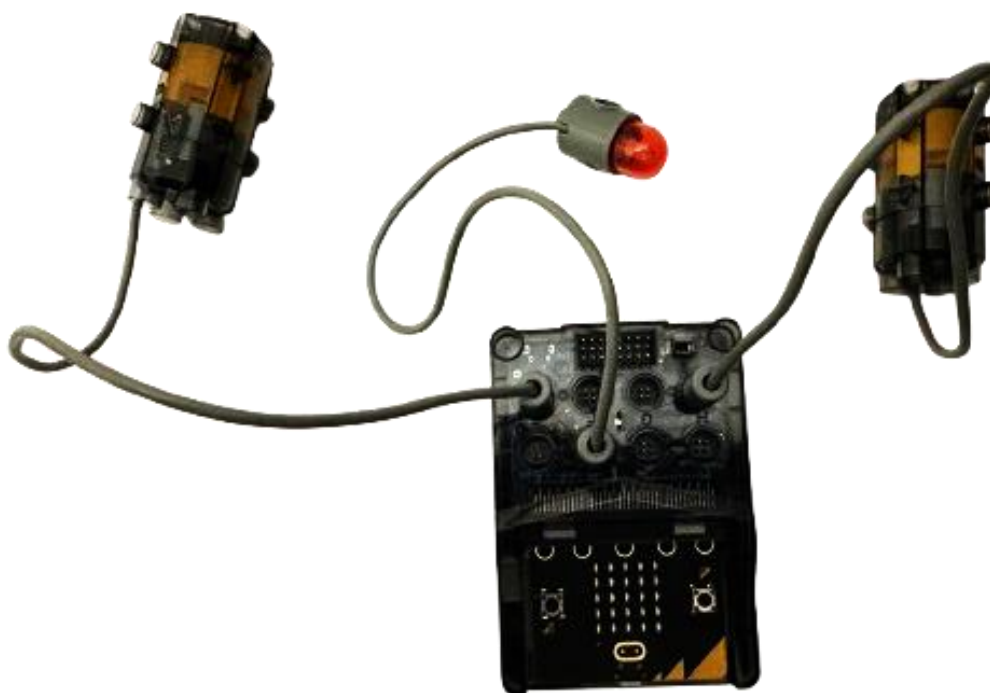
A. Σενάριο

Να συνδέσουμε και να λειτουργήσουμε όσο το δυνατόν περισσότερα IoTs στο **Gigo με micro:bit** και να τους προγραμματίσουμε με **Mind+**

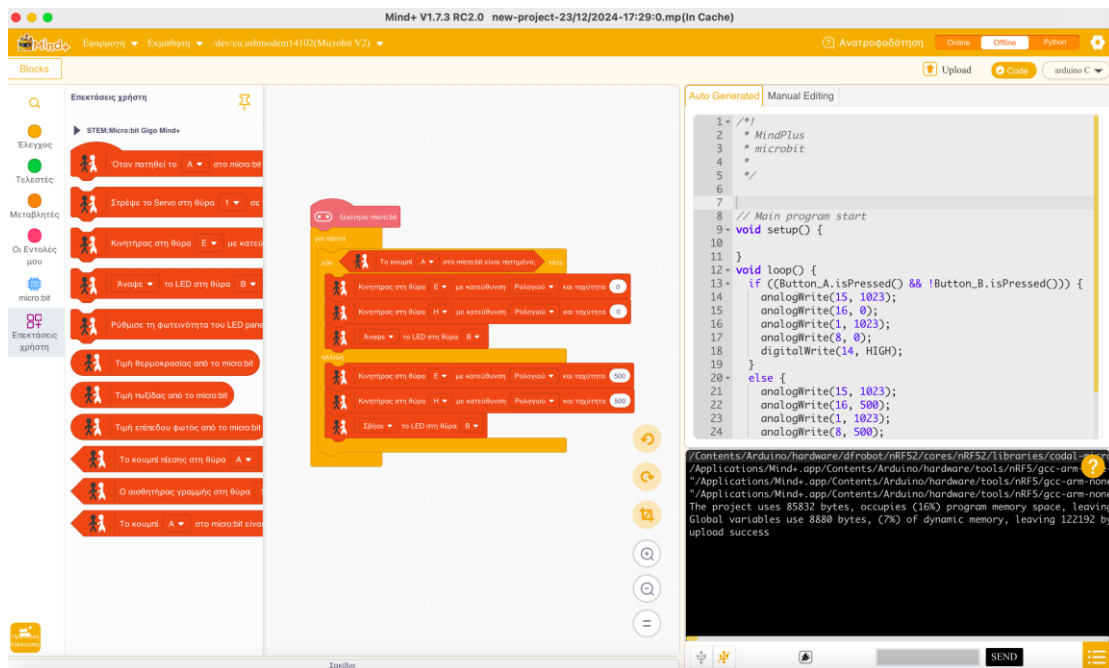
Συγκεκριμένα, να ανάβει το κόκκινο LED όταν πατάμε το κουμπί [A] στο micro:bit και να σταματούν την περιστροφή τους οι κινητήρες που σε κάθε άλλη περίπτωση περιστρέφονται διαρκώς.

Στο εργαστήριο προσομοιάζουμε απλοποιημένη τη λειτουργία του πεντάλ του φρένου ενός ηλεκτροκίνητου αυτοκινήτου.

B. Συνδεσμολογία



Γ. Κώδικας



Πηγές

<https://mindplus.cc/en.html>

<https://smartblox.gr>

<https://mindplus.dfrobot.com/introduction>

https://gitlab.com/ale3andro/mindplus_ext_s1

<https://portal.stem.edu.gr>

<https://www.why.gr>

https://www.stem-ib-support.gr/images/S2_Ardicon_Ver2.pdf?utm_source=chatgpt.com



Βιογραφικό σημείωμα



Ο Βασίλης Οικονόμου κατείχε τη θέση του Συμβούλου για θέματα Πληροφορικής στο Ελληνικό Ίδρυμα Πολιτισμού. Διετέλεσε Διευθυντής Πληροφορικής και Ψηφιακής Εκπαίδευσης στο Ελληνο – Αμερικανικό Εκπαιδευτικό Ίδρυμα – Κολλέγιο Αθηνών και Ψυχικού. Από το 1994 μέχρι και το 2020 ήταν Υπεύθυνος Πληροφορικής και Ψηφιακής Εκπαίδευσης στα Εκπαιδευτήρια Δούκα, καθώς και της Ομάδας Μαθητικού Υπολογιστή, με στόχο την ένταξη και αξιοποίηση του ατομικού Μαθητικού Υπολογιστή στην εκπαιδευτική διαδικασία. Είναι πιστοποιημένος αξιολογητής εταιριών με πρακτική εμπειρία στην αξιολόγηση εταιριών στο επίπεδο «Δέσμευση στην Επιχειρηματική Αριστεία» – European Foundation for Quality Management Validator. Από το 2014 είναι σύμβουλος της [SoFIA Education Experts Ltd.](#) Το 2013 κερδίζει τη διάκριση Expert Educator και [εκπροσωπεί τη χώρα μας](#) (2014) στον Παγκόσμιο Διαγωνισμό Πρωτοπόρων Δασκάλων που διοργανώνει η [Microsoft στη Βαρκελώνη](#). Το 2014 κερδίζει με την ομάδα μαθητών του τον Ευρωπαϊκό διαγωνισμό προγραμματισμού: [Kodu Kup](#) (e-skills, European School Net, Microsoft). Το 2015 επιλέγεται εκ νέου ως Expert Educator (MIEExpert15) και ορίζεται ως ο ένας από τους δεκατρείς Microsoft Fellows στον κόσμο και κριτής των έργων των ομάδων οι οποίες διαγωνίστηκαν στο [SEATTLE \(USA\)](#). Έχει διακριθεί και βραβευτεί για έργα σχετικά με την [εισαγωγή της μεθοδολογίας «1:1»](#) στην Εκπαίδευση, καθώς και για την [επίδραση του ψηφιακού υλικού](#) στις μεθοδολογίες εκπαίδευσης και στο ρόλο του Εκπαιδευτικού. Έχει συμμετάσχει ως ερευνητής και αναλυτής – προγραμματιστής στην σχεδίαση και υλοποίηση πάνω από 50 ερευνητικών έργων σχετικά με τις ΤΠΕ στην Εκπαίδευση αλλά και στην Ειδική Αγωγή. Έχει συμμετάσχει επίσης στην ανάπτυξη ερευνητικού και αναπτυξιακού λογισμικού στις περιοχές αυτές, με πάνω από 80 τίτλους λογισμικού, καταρτίζοντας παράλληλα εκατοντάδες εκπαιδευτικούς στην αξιοποίηση των ΤΠΕ στη διδακτική πρακτική. Έχει αναπτύξει Συστήματα Διαχείρισης Δεδομένων (M.I.S.), σε αρκετά προγραμματιστικά περιβάλλοντα. Έχει δημοσιεύσει άρθρα και [μελέτες](#) σε περιοδικά και έχει [παρουσιάσει εισηγήσεις](#) σε επιστημονικά συνέδρια με θέμα την «Εισαγωγή των ΤΠΕ στην εκπαιδευτική διαδικασία» και την «Ποιότητα στην Εκπαίδευση». Τα ερευνητικά του ενδιαφέροντα εστιάζουν στη διεύρυνση της μεθοδολογίας «1:1», το Mixed Reality στην εκπαίδευση, τις διαδραστικές επιφάνειες και το ποιοτικό εκπαιδευτικό λογισμικό. Όραμά του είναι να εμπλέξει όσο το δυνατό μεγαλύτερο αριθμό μαθητών, εκπαιδευτικών και γονέων στην αξιοποίηση της Τεχνολογίας στην Εκπαίδευση.

<https://economu.wordpress.com>



Mind+ & ARD:icon, NEZHA, Gigo



Βασίλης Οικονόμου
economu.wordpress.com