

**ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ**  
**ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ**  
**ΤΜΗΜΑ ΜΗΧΑΝΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ**



**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**«Βελτιωτικός (ανα)σχεδιασμός πολυμορφικής μικρο-ρομποτικής διάταξης (πλατφόρμας μικρο-ρομπότ)»**

**«Improved (re)design of a multi-purpose micro-robotic platform»**

Φοιτήτρια: Ειρήνη-Μαρία Δήλερη [ΑΜ: 272017051]

Επιβλέπων Καθηγητής: Χαμηλοθώρης Γεώργιος

Αθήνα 2023

Η παρούσα διπλωματική εργασία εγκρίθηκε ομόφωνα από την τριμελή εξεταστική επιτροπή, η οποία ορίστηκε από την Γ.Σ. του Τμήματος Μηχανικών Βιομηχανικής Σχεδίασης και Παραγωγής του Πανεπιστημίου Δυτικής Αττικής, σύμφωνα με το νόμο και τον εγκεκριμένο Οδηγό Σπουδών του τμήματος.

Επιτροπή Αξιολόγησης:

<b>ΟΝΟΜΑΤΕΠΩΝΥΜΟ</b>	<b>ΥΠΟΓΡΑΦΗ</b>
ΧΑΜΗΛΟΘΩΡΗΣ ΓΕΩΡΓΙΟΣ	
ΣΤΕΡΓΙΟΥ ΚΩΝΣΤΑΝΤΙΝΟΣ	
ΧΑΤΖΟΠΟΥΛΟΣ ΑΒΡΑΑΜ	

## ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Ο/η κάτωθι υπογεγραμμένος/η **ΔΗΛΕΡΗ ΕΙΡΗΝΗ ΜΑΡΙΑ** του **ΚΩΝΣΤΑΝΤΙΝΟΥ**, με αριθμό μητρώου **272017051** φοιτητής/τρια του Πανεπιστημίου Δυτικής Αττικής της Σχολής Μηχανικών του Τμήματος Μηχανολόγων Μηχανικών, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της διπλωματικής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Ο/Η Δηλών/ούσα



## Ευχαριστίες

Με την ολοκλήρωση της διπλωματικής μου εργασίας θα ήθελα να ευχαριστήσω θερμά όλους όσους με βοήθησαν. Ευχαριστώ θερμά τον επιβλέποντα καθηγητή μου, κ. Γεώργιο Χαμηλοθώρη, για την εμπιστοσύνη που μου έδειξε αναθέτοντας μου αυτό το ιδιαίτερα ενδιαφέρον θέμα και για το ενδιαφέρον και την καθοδήγηση του καθ' όλη την διάρκεια της διπλωματικής. Επίσης, ιδιαίτερες ευχαριστίες οφείλω προς τον προπονητή μου κ. Ιωάννη Τούρλο, από την συμμετοχή μου στην ομάδα αγωνιστικής ρομποτικής, για την αμέριστη συμπαράσταση του και τις πολύτιμες συμβουλές του κατά την διάρκεια της παρούσας διπλωματικής. Τέλος, ένα τεράστιο ευχαριστώ στη δική μου παρέα για την υποστήριξη τους, την Δήμητρα, τον Νίκο, την Μαρία, τον Αποστόλη και τον Παντελή.

## Περίληψη

Η εργασία αφορά την ανάπτυξη μιας πολυμορφικής αυτο-κινούμενης μικρο-ρομποτικής πλατφόρμας και σχετικών ενδεικτικών εκπαιδευτικών ασκήσεων. Περιλαμβάνει το σχεδιασμό και την κατασκευή ενδεικτικών μικρο-ρομποτικών διατάξεων με προσθετική κατεργασία (3D printing). Το βασικό χαρακτηριστικό της ηλεκτρονικής διάταξης είναι ο επεξεργαστής RP2040, ο οποίος αναπτύχθηκε από την Raspberry Pi, για τον μικροελεγκτή Pico, σε συνδυασμό με τους αισθητήρες ακολουθίας τροχιάς στο έδαφος. Για τον προγραμματισμό του επιλέγεται η προγραμματιστική γλώσσα MicroPython, που αναπτύχθηκε από την Python 3 με σκοπό να χρησιμοποιηθεί για μικροελεγκτές. Σκοπός της διάταξης είναι να εισαχθεί στο εργαστηριακό μέρος του μαθήματος της Μηχατρονικής. Για αυτό δίνονται κάποιες ενδεικτικές ασκήσεις, μία εκ των οποίων είναι η παρακολούθηση τροχιάς με οπτικούς αισθητήρες. Συνεπώς, εξετάζει το ρόλο και μελετά τα χαρακτηριστικά μιας μικρο-ρομποτικής πλατφόρμας κατάλληλης για τις σύγχρονες διδακτικές ανάγκες.

## Abstract

The object of this undergraduate thesis is the development of a polymorphic self-driving micro-robotic platform and relevant training exercises. One point of focus is the design and manufacture of a sample of the micro-robot with 3D printing. The key feature of the electronic layout is the RP2040 processor, developed by Raspberry Pi, for the Pico microcontroller, along the infrared line following sensors. For its programming necessities, the newly developed language MicroPython is chosen, derived by Python 3 in order to be used for microcontrollers. The purpose of the device is to be introduced it into the laboratory part of the Mechatronics course. For this reason, some indicative exercises are given, one of which is course tracking with optical sensors. Therefore, it examines the role and studies the characteristics of a micro-robotic platform suitable for modern teaching needs.

## Περιεχόμενα

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ .....	3
Ευχαριστίες .....	4
Περίληψη .....	5
Abstract.....	5
Κατάλογος Πινάκων .....	8
Κατάλογος Διαγραμμάτων.....	9
Κατάλογος Εικόνων .....	9
Εισαγωγή .....	12
Κεφάλαιο 1ο    Ανάλυση του προβλήματος.....	14
1.1. Μηχατρονική στην εκπαίδευση .....	15
1.2. Σκοπός και χαρακτηριστικά της αναβαθμισμένης διάταξης.....	17
1.3. Μεθοδολογία αξιολόγησης και σχεδιασμού .....	18
Κεφάλαιο 2ο    Σχεδιασμός της διάταξης .....	30
2.1. Παραδοχές.....	30
2.1.1. Πολυμορφία .....	30
2.1.2. Διαστάσεις .....	31
2.2. Τεχνολογία τρισδιάστατης εκτύπωσης .....	32
2.2.1. Τι είναι μια 3D εκτύπωση .....	32
2.2.2. Ιστορία των τρισδιάστατων εκτυπωτών.....	32
2.2.3. Γιατί επιλέχθηκε η 3D εκτύπωση .....	32
2.2.4. Τρισδιάστατη εκτύπωση στην εκπαίδευση .....	33
2.2.5. Επιλογή υλικού .....	34
2.3. Σχέδια για την κατασκευή.....	34
2.3.1. Πάνω πλάκα .....	34
2.3.2. Κάτω πλάκα .....	35
2.4. Προτεινόμενη κατασκευή .....	39
2.4.1. Τρισδιάστατος εκτυπωτής.....	40
2.4.2. Πρόγραμμα εκτύπωσης.....	40
2.4.3. Τελική κατασκευή.....	42
Κεφάλαιο 3ο    Ηλεκτρονική Διάταξη .....	44
3.1. Raspberry .....	44
3.1.1. Raspberry Pi RP2040/ Pico.....	45
3.1.2. Maker Pi Pico.....	47
3.1.3. Maker Pi RP2040.....	49

3.2. Αισθητήρες .....	51
3.2.1. Αισθητήρας Infrared TCRT5000- Line Follower .....	52
3.2.2. Οπτικός Αισθητήρας Μέτρησης Περιστροφών (Photo Interrupter) .....	55
Κεφάλαιο 4ο    Γλώσσα προγραμματισμού .....	57
4.1. MicroPython .....	58
4.1.1. Βιβλιοθήκες Python/ MicroPython .....	59
4.1.2. Ψηφιακές ακίδες .....	60
4.1.3. Αναλογικές Ακίδες.....	62
4.1.4. Κινητήρες DC .....	63
4.1.5. Κινητήρες Servo .....	64
4.1.6. Απλές Κινήσεις.....	64
4.2. Αλγόριθμος PID/ PD.....	65
4.3. Εγκατάσταση λογισμικού .....	67
4.3.1. Στην πλακέτα .....	67
4.3.2. Στον υπολογιστή .....	68
Κεφάλαιο 5ο    Ενδεικτικές ασκήσεις.....	69
5.1. Μέρος Α: Ανάλυση ασκήσεων για τον διδάσκοντα .....	70
Άσκηση 1 .....	70
Άσκηση 2 .....	72
Άσκηση 3 .....	76
Άσκηση 4 .....	77
Άσκηση 5 .....	79
5.2. Μέρος Β: Φύλλα εργασίας για τους φοιτητές.....	79
Κεφάλαιο 6ο    Συμπεράσματα .....	80
Βιβλιογραφία .....	81
Παράρτημα Α: Αναλυτικοί Πίνακες εμπορικών ρομποτικών διατάξεων.....	86
Παράρτημα Β: Σχέδια κατασκευής.....	94
Παράρτημα Γ: Ηλεκτρονικά Σχέδια .....	98
Παράρτημα Δ: Κώδικες Επίλυσης των ενδεικτικών Ασκήσεων για την πλακέτα Maker Pi RP2040	101
Παράρτημα Ε: Φύλλα έργου φοιτητών.....	103

## Κατάλογος Πινάκων

Πίνακας 1-1: Αναλυτικά στοιχεία για το μάθημα της Μηχατρονικής στο Πανεπιστήμιο Δυτικής Αττικής.....	16
Πίνακας 1-2: Προτεινόμενα σενάρια ασκήσεις για το εργαστηριακό μάθημα της Μηχατρονικής.....	18
Πίνακας 1-3: Τεχνικά χαρακτηριστικά .....	20
Πίνακας 1-4: Κλίμακα αξιολόγησης κόστους.....	22
Πίνακας 1-5: Αρχική Αξιολόγηση Εμπορικών ρομποτικών διατάξεων .....	22
Πίνακας 1-6: Αρχική Αξιολόγηση Εμπορικών ρομποτικών διατάξεων (συνέχει πίνακα 1-5).....	23
Πίνακας 1-7: Κλίμακα για την ποσοτικοποίηση.....	24
Πίνακας 1-8: Διαμόρφωση τελικής βαθμολογίας μετά την ποσοτικοποίηση.....	24
Πίνακας 1-9: Διαμόρφωση τελικής βαθμολογίας μετά την ποσοτικοποίηση (συνέχεια πίνακα 8).....	25
Πίνακας 2-1: Χρόνος εκτύπωσης ανάλογα με το κομμάτι της κατασκευής.....	41
Πίνακας 2-2: Συνολική λίστα με τα υλικά για την προτεινόμενη κατασκευή.....	42
Πίνακας 4-1: Τυπικές βιβλιοθήκες της Python.....	59
Πίνακας 4-2: Τυπικές βιβλιοθήκες της MicroPython .....	60
Πίνακας 4-3: Κώδικας για την ακίδα 5 της πλακέτας RP2040.....	60
Πίνακας 4-4: Κώδικας για τα NeoPixel LED, με χρήση εντολής “r,g,b” για τον ορισμό του χρώματος. ....	61
Πίνακας 4-5: Κώδικας που χρησιμοποιείται για την ανάγνωση των αναλογικών ακίδων, συγκεκριμένα της ακίδας 0. ....	62
Πίνακας 4-6: Κομμάτι κώδικα για τον προγραμματισμό κινητήρα DC .....	63
Πίνακας 4-7: Κομμάτι κώδικα για τον προγραμματισμό κινητήρων Servo .....	64
Πίνακας 4-8: Κομμάτι κώδικα για τον προγραμματισμό κινητήρα DC, ώστε να κινηθεί μπροστά (forward) για 2 δευτερόλεπτα και μετά να σταματήσει. (συνέχεια από τον πίνακα 4-6) .....	65
Πίνακας 4-9: Κομμάτι κώδικα από το συνολικό πρόγραμμα για τον αλγόριθμο ακολουθίας γραμμής. ....	67
Πίνακας 5-1: Προτεινόμενα σενάρια/ ασκήσεις για το εργαστηριακό μάθημα της Μηχατρονικής, μαζί με τις βιβλιοθήκες/ εντολές και την κατασκευή που χρησιμοποιούν (Ασκήσεις 1 και 2).....	69
Πίνακας 5-2: Προτεινόμενα σενάρια/ ασκήσεις για το εργαστηριακό μάθημα της Μηχατρονικής, μαζί με τις βιβλιοθήκες/ εντολές και την κατασκευή που χρησιμοποιούν (Ασκήσεις 3, 4 και 5)- συνέχεια πίνακα 5-1 .....	70
Πίνακας 5-3: Βιβλιοθήκες, εντολές και η περιγραφή τους για την άσκηση 1 .....	71
Πίνακας 5-4: Κώδικας LED- Pin 5 .....	71
Πίνακας 5-5: Βιβλιοθήκες, εντολές και η περιγραφή τους για την άσκηση 2 .....	73
Πίνακας 5-6 Κώδικας για έναν κινητήρα - Pins 8 and 9 (1ο μέρος).....	74



Πίνακας 5-7: Κώδικας με χρήση def (2ο μέρος) .....	74
Πίνακας 5-8: Βιβλιοθήκες, εντολές και η περιγραφή τους για την άσκηση 3 .....	76
Πίνακας 5-9: Κώδικας για την ανάγνωση αναλογικού αισθητήρα.....	76
Πίνακας 5-10: Βασικός κώδικας για το αλγόριθμο PD .....	77

## Κατάλογος Διαγραμμάτων

Διάγραμμα 1-1: Radar Chart για το μοντέλο Actobotics kit-ActoBitty 2.....	26
Διάγραμμα 1-2: Radar Chart για το μοντέλο Mini round robot Chassis Kit- 2WD w DC Motors.....	26
Διάγραμμα 1-3: Radar Chart για το μοντέλο Mini robot rover chassis Kit- 4WD w DC Motors .....	26
Διάγραμμα 1-4: Radar Chart για το μοντέλο Mini robot rover chassis Kit- 2WD w DC Motors .....	27
Διάγραμμα 1-5: Radar Chart για το μοντέλο Mini 3-layer round robot chssis kit- 2WD w DC motors .....	27
Διάγραμμα 1-6: Radar Chart για το μοντέλο AlphaBot.....	27
Διάγραμμα 1-7: Radar Chart για το μοντέλο PiRacer DonkeyCar, AI Racing Robot for Raspberry Pi 4 .....	28
Διάγραμμα 1-8: Radar Chart για το μοντέλο PiRacer Pro High Speed, AI Racing Robot for Raspberry Pi 4 .....	28
Διάγραμμα 1-9: Radar Chart για το μοντέλο AlphaBot2 Robot Building Kit for Raspberry Pi 3 Model B.....	28
Διάγραμμα 1-10: Συνολική τελική βαθμολογία από την αξιολόγηση .....	29
Διάγραμμα 2-1: Στο διάγραμμα 1.α. απεικονίζεται η διαδικασία κατασκευής για τις κλασσικές τεχνικές, ενώ στο 1.β απεικονίζεται η αντίστοιχη διαδικασία για την τρισδιάστατη εκτύπωση .....	33
Διάγραμμα 4-1: Παράδειγμα ΣΑΕ (α) Σύστημα Ανοιχτού Βρόγχου (β) Σύστημα Κλειστού Βρόγχου .....	57
Διάγραμμα 4-2: Απλός έλεγχος PID για ένα σύστημα. [46].....	65

## Κατάλογος Εικόνων

Εικόνα 1-1: Τα πεδία εφαρμογής και τα στοιχεία της Μηχατρονικής.....	15
Εικόνα 1-2: Πεδία εφαρμογής του Arduino.....	19
Εικόνα 1-5: Σχεδιάγραμμα πλακέτας ESP32.....	21
Εικόνα 1-3: Σχεδιάγραμμα πλακέτας Raspberry .....	21
Εικόνα 1-4: Σχεδιάγραμμα πλακέτας Arduino .....	21
Εικόνα 2-1: LEGO Education SPIKE Prime [50].....	30

Εικόνα 2-2: Τελική προτεινόμενη κατασκευή, δύο κινητήρες DC, ένα κινητήρα servo σε μηχανισμό gripper, δύο αισθητήρες TCRT5000, δύο οπτικούς αισθητήρες μέτρησης περιστροφών, την πλακέτα Maker Pi RP2040 και μία μπαταρία 3.7 V (1200mAh).....	31
Εικόνα 2-3: Φωτογραφία από το πρόγραμμα Inventor Pro 2022, όπου απεικονίζεται η κοινή πάνω πλάκα. ....	34
Εικόνα 2-4: Παραδείγματα μηχανισμού αρπάγης (gripper) με ενσωματωμένους κινητήρες Servo, συγκεκριμένα έχουν microservo motor .....	35
Εικόνα 2-5: Φωτογραφία από το πρόγραμμα Inventor Pro 2022, όπου απεικονίζεται η κάτω πλάκα για κινητήρες DC.....	36
Εικόνα 2-6: Δύο διαφορετικά μοντέλα κινητήρα DC του εμπορίου, σύμφωνα με τα οποία σχεδιάστηκε η κάτω πλάκα.....	37
Εικόνα 2-7: Στηρίγματα του εμπορίου για τους κινητήρες DC της εικόνας 4.....	38
Εικόνα 2-8: Τα δύο στηρίγματα για τους κινητήρες DC, όπου το α είναι του δεξιού και το β είναι του αριστερού κινητήρα .....	38
Εικόνα 2-9: Φωτογραφία από το πρόγραμμα Inventor Pro 2022, όπου απεικονίζεται η κάτω πλάκα για κινητήρες Servo .....	39
Εικόνα 2-10: Στην εικόνα 7.α. απεικονίζεται ένας τυπικός σερβοκινητήρας και στην 7.β. είναι το στηρίγμα σύμφωνα με το σχεδιάστηκε η κάτω πλάκα σε αυτή την περίπτωση .....	39
Εικόνα 2-11: Τρισδιάστος εκτυπώτης Creality CR-20 Pro .....	40
Εικόνα 2-12: Το περιβάλλον του προγράμματος Creality Slicer 4.8.2. με τα δύο στηρίγματα για τους κινητήρες DC.....	40
Εικόνα 2-13: Το μενού του προγράμματος εκτύπωσης για τις μεταβλητές που μπορούν να αλλάξουν ανάλογα με τις ανάγκες του υλικού και του μοντέλου. ....	41
Εικόνα 2-15: Τελική προτεινόμενη κατασκευή μπροστινή όψη.....	43
Εικόνα 2-14: Τελική προτεινόμενη κατασκευή με την πλακέτα Maker Pi RP2040.....	43
Εικόνα 3-1: Δύο από τα προϊόντα της εταιρείας Raspberry Pi. Στην (α) είναι το παραδοσιακό Raspberry Pi, ενώ στην (β) είναι ο μικροελεγκτής Raspberry Pi Pico[11] .....	45
Εικόνα 3-2: (α) Ο επεξεργαστής RP2040 της Raspberry Pi και (β) η εξήγηση για το όνομα του επεξεργαστή RP2040[11] .....	45
Εικόνα 3-3: Το σύστημα του επεξεργαστή RP2040 [32] .....	46
Εικόνα 3-4: Το μοντέλο Raspberry Pi Pico[11].....	47
Εικόνα 3-5: Ηλεκτρονικές πλακέτες με τον RP2040 που δεν είναι της Raspberry. (α) Waveshare RP2040 Mini [51], (β)Beetle RP2040 Mini Development [52], (γ) Adafruit KB2040-RP2040 Kee Board driver [53] και (δ) SparkFun Pro Micro- RP2040 [54] .....	47
Εικόνα 3-6: Το Maker Pi Pico (α) η μπροστινή όψη, μαζί με το Raspberry Pi Pico, και (β) η πίσω όψη, με αναλυτικά στοιχεία για τις ακίδες .....	48
Εικόνα 3-7: Το σχέδιο που δείχνει τις ακίδες για το μοντέλο Raspberry Pi Pico [32] .....	48

Εικόνα 3-8: Το σχέδιο με τις αναλυτικές λειτουργίες και την τοποθέτηση τους στο μοντέλο Maker Pi Pico[55].....	49
Εικόνα 3-9: Οι διαστάσεις της πλακέτας Maker Pi Pico[55].....	49
Εικόνα 3-10: Η πλακέτα Maker Pi RP2040, (α) η μπροστινή όψη και (β) η πίσω όψη [33].....	50
Εικόνα 3-11: Το σχέδιο με τις αναλυτικές λειτουργίες και την τοποθέτηση τους στο μοντέλο Maker Pi RP2040 [33].....	51
Εικόνα 3-12: Οι διαστάσεις της πλακέτας Maker Pi RP2040[33].....	51
Εικόνα 3-13: Διάγραμμα τυπικής λειτουργίας ενός αισθητήρα [56].....	52
Εικόνα 3-14: (α) Σχέδιο του αισθητήρα TCRT 5000 όπως εμφανίζεται στο φύλλο δεδομένων[35] (β) ο αισθητήρας TCRT 5000 όπως διατίθεται στο εμπόριο.....	53
Εικόνα 3-15: Δύο από τις εκδοχές που διατίθεται στο εμπόριο για τον αισθητήρα TCRT5000 [57] (α) με αναλογικές και ψηφιακές ακίδες και (β) μόνο με ψηφιακή ακίδα.....	53
Εικόνα 3-16: Τα ηλεκτρονικά κομμάτια από τα οποία αποτελείται το ολοκληρωμένο ηλεκτρονικό σύστημα και η ανάλυση των ακίδων [58].....	54
Εικόνα 3-17: Τα ηλεκτρονικά σχέδια του αισθητήρα TCRT5000 (α) μόνο με ψηφιακή ακίδα και (β) με αναλογικές και ψηφιακές ακίδες [57].....	54
Εικόνα 3-18: (α) Σχηματική αναπαράσταση του οπτικού αισθητήρα μέτρησης περιστροφών [59] και (β) ηλεκτρονικό σχέδιο του αισθητήρα [60].....	55
Εικόνα 3-19: Αναμενόμενο διάγραμμα που εξελίσσεται στο χρόνο για τον οπτικό αισθητήρα μέτρησης περιστροφών [60].....	56
Εικόνα 3-20: (α) οπτικός αισθητήρα μέτρησης περιστροφών και (β) ο διάτρητος τροχός που χρησιμοποιείται για την μέτρηση των περιστροφών [61].....	56
Εικόνα 4-1: Αποτέλεσμα του κώδικα του πίνακα 4-3 .....	61
Εικόνα 4-2: Αποτέλεσμα κώδικα του πίνακα 4-4, όπου τελικά ανάβει το NeoPixel LED κόκκινο χρώμα .....	62
Εικόνα 4-3: Προγραμματιστικό περιβάλλον του MU Editor.....	68
Εικόνα 4-4: Προγραμματιστικό περιβάλλον του Thonny Editor.....	68
Εικόνα 5-1: Ενδεικτική τροχιά μαύρου ίχνους.....	77
Εικόνα 5-2:Παράδειγμα συνολικής δοκιμασίας.....	79

## Εισαγωγή

Στην σύγχρονη εποχή η τεχνολογία έχει εξελιχθεί τόσο πολύ που πλέον κάθε οικία έχει τουλάχιστον ένα μηχανικό σύστημα, ακόμα και δεν μπορεί να το αναγνωρίσει. Η Μηχανική είναι ένα σύγχρονο διεπιστημονικό πεδίο το οποίο καλύπτει ένα ευρύ φάσμα εφαρμογών. Ένα μηχανικό σύστημα μπορεί να έχει πολλές μορφές και πολλές λειτουργίες. Κάποια παραδείγματα αυτών είναι ένα πολύπλοκο ρομποτικό σύστημα σε ένα εργοστάσιο ή μια απλή κατασκευή καφετιέρας που μπορεί να υπάρχει σε ένα οπουδήποτε σπίτι.

Αυτή η εξελικτική πορεία των μηχανικών συστημάτων οφείλεται στην ραγδαία ανάπτυξη των ηλεκτρονικών εξαρτημάτων και των υπολογιστών. Με την χρήση ηλεκτρονικών ένα απλό μηχανικό σύστημα, για παράδειγμα μια μπάρα που διακόπτει την κυκλοφορία για να περάσει ο σιδηροδρομικός σταθμός, μετατρέπεται σε ηλεκτρομηχανικό. Με την εισαγωγή προγράμματος το σύστημα καθίσταται μηχανικό και οι δυνατότητες που έχει αυξάνονται ανάλογα με την πολυπλοκότητα της εργασίας που καλείται να εκτελέσει.

Στο παράδειγμα της μπάρα του σιδηροδρόμου, από απλό μηχανικό χειροκίνητο σύστημα με την εισαγωγή ενός ηλεκτροκινητήρα γίνεται ηλεκτρομηχανολογικό. Με την τοποθέτηση ενός ρολογιού και του κατάλληλου προγράμματος, η μπάρα μπορεί σηκώνεται και να κατεβαίνει αυτόνομα χωρίς την ανθρώπινη παρέμβαση. Με την τοποθέτηση μιας κάμερας και κατάλληλων αλλαγών στο πρόγραμμα, η μπάρα μπορεί να εντοπίζει τυχόν απρόσεκτους οδηγούς ή την άφιξη του τρένου, άμα αυτό έχει καθυστερήσει.

Αυτή η ευρεία εφαρμογή των μηχανικών συστημάτων καθιστά σχεδόν απαραίτητη την εκπαίδευση των νέων μηχανικών στον σχεδιασμό, τον προγραμματισμό αλλά και τον έλεγχο τέτοιων συστημάτων. Σκοπός της παρούσας εργασίας είναι να δημιουργήσει μια πολυμορφική πλατφόρμα για το μάθημα της Μηχανικής του Πανεπιστημίου Δυτικής Αττικής. Αυτή η διάταξη θα είναι δυνατόν να χρησιμοποιηθεί κατά την διάρκεια του εργαστηριακού μέρους του μαθήματος, εισάγοντας τους φοιτητές πρακτικά στον χειρισμό ενός μηχανικού συστήματος.

Η συγκεκριμένη μηχανική διάταξη έχει την μορφή τροχοκίνητου οχήματος, το οποίο μπορεί να κινείται αυτόνομα στον χώρο. Για την υλοποίηση της χρησιμοποιήθηκε τεχνολογία προσθετικής κατασκευής (3D printing) για την δημιουργία ενός καινούργιου σώματος. Επίσης η ηλεκτρονική διάταξη αποτελείται κυρίως από τον επεξεργαστή Raspberry RP2040, που βρίσκεται στον καινούργιο μικροελεγκτή της Raspberry το Raspberry Pi Pico. Ο επεξεργαστής αυτός προσφέρει την δυνατότητα προγραμματισμού σε MicroPython, μια εναλλακτική μορφή της Python 3 προσαρμοσμένη για μικροελεγκτές.

Στο κεφάλαιο 1, αναλύεται ο λόγος για τον οποίο επιλέχθηκε να κατασκευαστεί από την αρχή η μικρορομποτική διάταξη της εργασίας, αντί να αγοραστεί από το εμπόριο. Ταυτόχρονα αξιολογούνται κάποιες από τις διαθέσιμες πλατφόρμες με βάση τα περιεχόμενα και τις ανάγκες των μαθημάτων του Πανεπιστημίου Δυτικής Αττικής στις σχολές Μηχανολόγων Μηχανικών, Ηλεκτρολόγων και Ηλεκτρονικών Μηχανικών, Μηχανικών Πληροφορικής και Υπολογιστών και Μηχανικών Βιομηχανικής Παραγωγής.

Στο κεφάλαιο 2, εξηγείται αναλυτικά η διαδικασία της σχεδίασης της κατασκευής. Αναλύονται οι παραδοχές που χρειάστηκε να γίνουν δεκτές και η διαδικασία επιλογής των χαρακτηριστικών της διάταξης. Επίσης δίνεται ένα παράδειγμα μιας εκ των πιθανών κατασκευών, η οποία υλοποιήθηκε στην πραγματικότητα για τις ανάγκες του προγραμματισμού.

Το κεφάλαιο 3 περιλαμβάνει τα ηλεκτρονικά στοιχεία της διάταξης, δηλαδή τις προτεινόμενες ηλεκτρονικές πλακέτες αλλά και του αισθητήρες που είναι συμβατοί με αυτή. Σε αυτό το σημείο εξηγείται η λειτουργία και η χρήση των αισθητήρων.

Στο κεφάλαιο 4 αναλύεται ο προγραμματισμός της πλατφόρμας. Έχει επιλεχθεί η προγραμματιστική γλώσσα MicroPython και δίνονται τα κατάλληλα παραδείγματα για την ευκολότερη κατανόηση της.

Το κεφάλαιο 5 είναι εκπαιδευτικού περιεχομένου και αφορά το μάθημα της Μηχατρονικής. Δίνονται πέντε ασκήσεις/ παραδείγματα, με αύξουσα δυσκολία, οι οποίες μπορεί να πραγματοποιηθούν κατά την διάρκεια του εξαμήνου. Στο αντίστοιχο παράρτημα δίνονται και τα ενδεικτικά φύλλα έργου για τρεις εκ των πέντε ασκήσεων.

Τέλος, στο κεφάλαιο 6 γίνεται η αξιολόγηση της μικρορομποτικής πλατφόρμας και δίνονται μερικές ιδέες για την επέκταση της σε μελλοντική εργασία.

## Κεφάλαιο 1ο      Ανάλυση του προβλήματος

Στην σύγχρονη εποχή, η τεχνολογία καταλαμβάνει μεγάλο ποσοστό της καθημερινότητας και εμπλέκεται σε πολλούς τομείς της, όπως η εργασία, η εκπαίδευση και η διασκέδαση. Οι περισσότεροι άνθρωποι χρησιμοποιούν τα κινητά τηλέφωνα, τους υπολογιστές, τα αυτοκίνητα και άλλα τεχνολογικά δημιουργήματα για να κάνουν την ζωή ευκολότερη, έχοντας σχεδόν πλήρη άγνοια για το πόσο πολύπλοκες διαδικασίες αυτά εκτελούν. Ωστόσο, η τεχνολογική ανάπτυξη δεν σταμάτα σε αυτά τα απλά παραδείγματα, αλλά επεκτείνεται και σε επιστημονικά πεδία, όπως η Ιατρική ή οι Τέχνες. Στην Ιατρική, για παράδειγμα, γίνεται χρήση διαφόρων μηχανημάτων με σκοπό την διεξαγωγή εξετάσεων για την καλύτερη διάγνωση και θεραπεία ενός ασθενή.

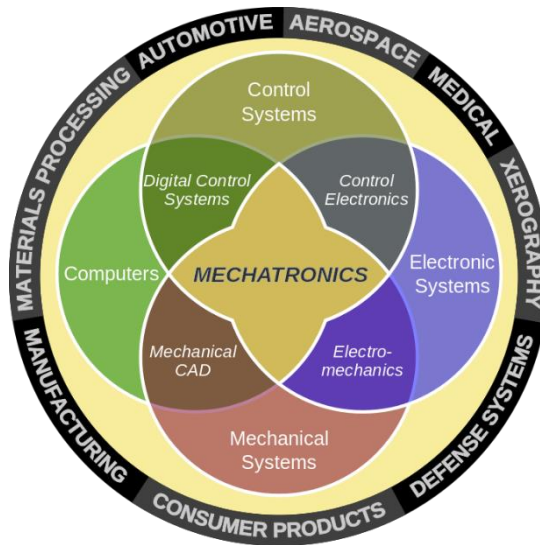
Αυτά αποτελούν μερικά παραδείγματα της τεχνολογικής ανάπτυξης. Αυτή οφείλεται τόσο στις βασικές επιστήμες, στις οποίες κατατάσσονται τα Μαθηματικά, η Φυσική, η Χημεία και η Βιολογία, αλλά και σε εξειδικευμένους κλάδους, όπως η Μηχανολογία, η Ηλεκτρολογία, η Ηλεκτρονική και η Πληροφορική. Αυτί οι κλάδοι δεν αποτελούν τους μοναδικούς παράγοντες ανάπτυξης, καθώς υπάρχουν αρκετοί ακόμα.

Πριν από κάποια χρόνια, οι κλάδοι αυτοί λειτουργούσαν ανεξάρτητα ο ένας από τον άλλο, λόγω των περιορισμένων γνώσεων που υπήρχαν. Όμως, τόσο η συνεχή αύξηση των απαιτήσεων της καθημερινότητας όσο και της βιομηχανίας, οδήγησαν στην δημιουργία διεπιστημονικών κλάδων για την κάλυψη αυτών των αναγκών. *Η «Μηχατρονική» είναι ένα διεπιστημονικό πεδίο το οποίο δανείζεται στοιχεία από την Μηχανολογία, την Ηλεκτρονική- Ηλεκτρολογία και την Πληροφορική.*[1] Ως «μηχατρονικό σύστημα» ορίζεται κάθε μηχανολογικό σύστημα εμπλουτισμένο με ηλεκτρονικά και ηλεκτρολογικά στοιχεία, και συχνά εμπεριέχουν λογισμικό.

Ο όρος «Μηχατρονική» εμφανίστηκε πρώτη φορά στην Ιαπωνία το 1969. Έχει αποδοθεί στον Tetsuro Mori, που άνηκε στην Yasakawa Electric Corporation. Εκείνη την περίοδο, η εταιρεία είχε ξεκινήσει να εισάγει ηλεκτρονικά στοιχεία σε μηχανολογικά συστήματα και ήθελε να δώσει ένα καινούργιο όνομα στην νέα αυτή σύνθεση τεχνολογιών.[2] Έτσι, επινοήθηκε ο όρος «Μηχατρονική». Στην αρχή, η νέα τεχνολογία δεν είχε δημοσιότητα και δεν χρησιμοποιούταν ο όρος, αλλά από τα 1980 και έπειτα η «Μηχατρονική» γνώρισε ραγδαία ανάπτυξη. Με τον όρο «Μηχατρονική» να γίνεται δημοφιλής, ο ορισμός του άλλαξε και περιλάμβανε κάθε είδους σύστημα το οποίο είναι ο συνδυασμός μηχανολογικών και ηλεκτρονικών στοιχείων.

Τα παραδείγματά στην καθημερινότητα είναι το ίδιο πολυπληθές με εκείνα της βιομηχανίας ή άλλων επιστημονικών πεδίων. Ως γενικός ορισμός, η «Μηχατρονική» είναι η συνεργασία μεταξύ των μηχανολογικών και των ηλεκτρονικών στοιχείων, τα οποία αποτελούν ένα σύστημα. Αυτό το σύστημα μπορεί να είναι ένα «έξυπνο» πλυντήριο ρούχων, το σύστημα ABS ενός αυτοκινήτου, ένας ρομποτικός βραχίονας σε ένα εργοστάσιο και άλλα.

Τα πεδία εφαρμογής της Μηχατρονικής είναι πολλά. Μερικά από αυτά είναι η αεροναυπηγική, η βιομηχανία αυτοκινήτων, η κατασκευή υλικών και πολλά ακόμη. Για την καλύτερη κατανόηση της ευρείας χρήσης μηχανολογικών συστημάτων παρατίθεται το παρακάτω σχέδιο. (εικόνα 1-1)



*Εικόνα 1-1: Τα πεδία εφαρμογής και τα στοιχεία της Μηχατρονικής*

Είναι εμφανές ότι τα συστήματα αυτά λειτουργούν χάρη στο συνδυασμό μηχανικών και ηλεκτρονικών στοιχείων. Από τα πιο πολύπλοκα συστήματα που είναι μία πλήρως αυτοματοποιημένη παραγωγή έως τα πιο απλά που εμφανίζονται στην καθημερινότητα, τα μηχατρονικά συστήματα είναι πλέον βασικό στοιχείο της σύγχρονης εποχής. Για αυτόν τον λόγο, η «Μηχατρονική» αποτελεί και κομμάτι της εκπαίδευσης σε σχολές μηχανικών.

### **1.1. Μηχατρονική στην εκπαίδευση**

Τα προγράμματα σπουδών των Μηχανολόγων Μηχανικών, των Ηλεκτρολόγων-Ηλεκτρολόγων Μηχανικών, των Μηχανικών Βιομηχανικής Σχεδίασης και Παραγωγής και των Μηχανικών Πληροφορικής και Υπολογιστών, του Πανεπιστημίου Δυτικής Αττικής, περιέχουν το μάθημα της «Μηχατρονικής». Αυτές οι σχολές, αν και αφορούν διαφορετικές επιστήμες, μπορούν να συνδυαστούν σε έναν διεπιστημονικό κλάδο, την «Μηχατρονική». Σκοπός του μαθήματος είναι οι φοιτητές να είναι σε θέση να αναγνωρίζουν, διατυπώνουν και αναλύουν πρακτικά συστήματα μηχατρονικής, καθώς και να επιλύουν τυχόν σφάλματα.

Παρόλο που ο γενικός σκοπός του μαθήματος είναι αυτός το περιεχόμενο μεταβάλλεται ανάλογα με την αρχική επιστημονική πλευρά. Σύμφωνα με τα προγράμματα σπουδών, στους Μηχανολόγους θα δοθεί μεγαλύτερη βαρύτητα στην υλική σύσταση του συστήματος. Οι κινητήρες, ο χρόνος αντίδρασης των συστημάτων, οι αισθητήρες και εφαρμογές στη βιομηχανία είναι μέρος του μαθήματος. Αντίθετα, στους Μηχανικούς Πληροφορικής και Συστημάτων, δίνεται έμφαση στο προγραμματιστικό μέρος, μέσω της επεξεργασίας σήματος, της θεωρίας ελέγχου και των ελεγκτών PLC. Στην συνέχεια παρατίθεται ένας αναλυτικός πίνακας (Πίνακας 1) στον οποίο εμφανίζονται οι σχετικές πληροφορίες για τα τέσσερα αυτά τμήματα.

Για την επιλογή των στοιχείων του πίνακα 1 επιλέχθηκαν βασικές αλλά σημαντικές πληροφορίες σχετικά με το μάθημα και πως διαρθρώνεται η ύλη. Προτιμήθηκε το περιεχόμενο έναντι των μαθησιακών αποτελεσμάτων λόγω της ομοιότητας που εμφανίζεται. Σκοπός του μαθήματος είναι

η κατανόηση των μηχανικών συστημάτων, αλλά η σκοπιά με την οποία επιτυγχάνεται αυτό διαφέρει ανάλογα με το τμήμα.

Πίνακας 1-1: Αναλυτικά στοιχεία για το μάθημα της Μηχατρονικής στο Πανεπιστήμιο Δυτικής Αττικής

<b>Τμήμα</b>	<b>Εξάμηνο</b>	<b>Κατεύθυνση</b>	<b>Περιεχόμενο</b>
<i>Μηχανολόγων Μηχανικών [3]</i>	9	Κατεύθυνση 2: Κατασκευαστικός τομέας	Εισαγωγή στη Μηχατρονική, Μετατροπείς Ενέργειας – Ενεργοποιητές, Ηλ. Μηχανές, Κινητήρες DC, βηματικοί κινητήρες, σερβοκινητήρες, Μετρήσεις-Αισθητήρια, Ημιαγωγοί & Ημιαγωγικές διατάξεις, Μικρο-ηλεκτρομηχανικά συστήματα (MEMS), Μικροελεγκτές, Προγραμματισμός μικροελεγκτών, Έλεγχος-Προγραμματισμός Μηχατρονικών Συστημάτων, Εφαρμογές μηχανικής στην αυτοκινητοβιομηχανία, Εφαρμογές Μηχατρονικής
<i>Ηλεκτρολόγων και Ηλεκτρονικών Μηχανικών [4]</i>	9	Γ κύκλος σπουδών "Ηλεκτρονικής & Υπολογιστικών Συστημάτων"	Εισαγωγή στη Μηχατρονική. Δυναμική συστημάτων κίνησης. Έλεγχος κίνησης. Ηλεκτρομηχανικοί ενεργοποιητές. Αναλογικά ηλεκτρονικά σε Μηχατρονικά συστήματα. Σχεδιασμός Συστημάτων. Το μοντέλο V. Μετατροπείς Ενέργειας-Ενεργοποιητές. Έλεγχος-Προγραμματισμός Μηχατρονικών Συστημάτων. Αισθητήρια. Ηλεκτρονικές διατάξεις προσαρμογής σημάτων Αισθητηρίων-ενεργοποιητών.
<i>Μηχανικών Πληροφορικής και Υπολογιστών [5]</i>	8	Ρόη Υλικού και Υπολογιστικών Συστημάτων	Εισαγωγή στη Μηχατρονική. Τεχνολογικό υπόβαθρο Μηχατρονικής : Ψηφιακά και Αναλογικά σήματα & συστήματα. Επεξεργασία σήματος. Ηλεκτρονικά & ενισχυτές ισχύος. Οπτοηλεκτρονικά. DIAC, TRIAC. Μετατροπείς A/D & D/A. Αισθητήρες & Ενεργοποιητές. Κινητήρες DC, AC, βηματικοί. Ψηφιακή λογική. Τεχνολογίες μικροελεγκτών. Συστήματα Ανάπτυξης συστημάτων με μικροελεγκτή Arduino. Σύστημα ανοικτού υλικού λογισμικού Raspberry. Θεωρία ελέγχου. Σχεδίαση ελεγκτών. PLC. Μεθοδολογία σχεδιασμού συστημάτων Μηχατρονικής.
<i>Μηχανικών Βιομηχανικής Σχεδίασης και Παραγωγής [6]</i>	7	Τομέας I: Συστημάτων Βιομηχανικής Παραγωγής	Εισαγωγή στη Μηχατρονική, ιστορικό, ορισμοί, οικονομική και κοινωνική σημασία. Το Μηχατρονικό Σύστημα: δομή και ροή πληροφορίας, ρόλος των διαστροφών (interfaces). Υποσύστημα δράσης, όργανα δράσης και διαστώματα: μηχανικά (ηλεκτρικοί κινητήρες DC, βηματικοί κινητήρες, σερβοκινητήρες), φωτοβολίας και άλλων ηλεκτρομαγνητικών εκπομπών. Υποσύστημα αίσθησης, αισθητήρια όργανα και διαστώματα: διακοπτικά, αναλογικά, διαιρέτης τάσης. Βασικά



		κυκλώματα μικροελεγκτών: Αναλογική-Ψηφιακή μετατροπή, βρόχος υστέρησης. Υποσύστημα επικοινωνιών: διατάξεις και πρωτόκολλα Asynchronous Serial, Serial Peripheral Interface, Inter-Integrated Circuit. Στοιχεία υλισμικού και κυκλώματα μικροελεγκτών: Αναλογική-Ψηφιακή μετατροπή, βρόχος υστέρησης, σύνδεση διακόπτη. Στοιχεία προγραμματισμού μικροελεγκτών και βασικά χαρακτηριστικά λειτουργικού συστήματος. Προγραμματισμός πραγματικού χρόνου: διαχείριση του χρόνου (timing), πολυπλεξία εκτέλεσης (Multitasking), χρήση διακοπών (Interrupt Service Routine). Παραδείγματα και εφαρμογές
--	--	--

Πέρα από τα τμήματα του Πανεπιστημίου Δυτικής Αττικής, υπάρχουν και άλλες σχολές στις οποίες διδάσκεται το μάθημα της Μηχατρονικής. Σε αυτό το σημείο έχει γίνει μια έρευνα σε αυτές τις σχολές και στην συνέχεια παρατίθενται αναλυτικά οι σχολές, τα τμήματα, τα εξάμηνα και το περιεχόμενο του μαθήματος.

## **1.2. Σκοπός και χαρακτηριστικά της αναβαθμισμένης διάταξης**

Όπως αναφέρθηκε ανωτέρω, η προϋπάρχουσα διάταξη έχει κλείσει δύο δεκαετίες χρήσης. Οι κατά καιρούς προσαρμογές ήταν περιορισμένες- αφορούσαν κυρίως την υποστήριξη νέων συστημάτων μικρο-ελεγκτών. Σήμερα διαπιστώνεται ότι η υφιστάμενη διάταξη αντιστοιχεί μόνο μερικώς στις απαιτήσεις του μαθήματος της Μηχατρονικής. Αυτό είναι το κίνητρο για την ανάπτυξη μιας νέας διάταξης.

Με την εισαγωγή μιας επικαιροποιημένης πλατφόρμας το μάθημα θα γίνει πιο διαδραστικό [7] και οι φοιτητές θα μπορέσουν να κατανοήσουν καλύτερα τις έννοιες που διδάσκονται. Σε αρκετά μαθήματα οι εργαστηριακές ασκήσεις γίνονται σε ψηφιακό περιβάλλον, χωρίς την επαφή του φοιτητή με την πειραματική διάταξη που καλείται να κατανοήσει. Η Μηχατρονική είναι ένα μάθημα με διεπιστημονικό υπόβαθρο, με αποτέλεσμα να εμφανίζονται δυσκολίες που οφείλονται στην αλλαγή επιστημονικής σκοπιάς. Συνεπώς, οφείλεται να δοθεί προσοχή για την καλύτερη κατανόηση των εννοιών και αυτό μπορεί να γίνει με την χρήση μιας μικρορομποτικής διάταξης [8].

Με σκοπό η αναβάθμιση της διάταξης να γίνει με βάση τα παραπάνω, δημιουργήθηκαν πέντε ασκήσεις τις οποίες θα μπορεί να εκτελέσει. Τα σενάρια αυτά φαίνονται στην συνέχεια σε μορφή πίνακα και είναι καταγεγραμμένα με αύξουσα δυσκολία. Επίσης, οι ασκήσεις αυτές δεν είναι δεσμευτικές και πιθανές αλλαγές και μετατροπές μπορούν να γίνουν για τις ανάγκες του μαθήματος. Ο τρόπος επίλυσης που προτείνεται σε επόμενο κεφάλαιο δεν είναι μοναδικός, καθώς είναι επιθυμητό να αναπτυχθούν και διαφορετικές λύσεις από τους φοιτητές και διδάσκοντες.

Πίνακας 1-2: Προτεινόμενα σενάρια ασκήσεις για το εργαστηριακό μάθημα της Μηχατρονικής

Τίτλος άσκησης	Περιγραφή	Προτεινόμενος χρόνος
Εισαγωγή με χρήση LED	Ο φοιτητής μαθαίνει να χρησιμοποιεί τις βασικές εντολές για να ανάψει και να σβήσει τα λαμπάκια LED.	1 βδομάδα
Απλές Κινήσεις	Ο φοιτητής προγραμματίζει το ρομπότ να εκτελεί απλές διαδρομές στον χώρο.(1 <sup>ο</sup> μέρος) Στη συνέχεια, ο φοιτητής εισαγάγει την πολυπλοκότητα του serno και μαθαίνει να συντάσσει λειτουργίες, οι οποίες θα ανακαλεστούν στο κύριο πρόγραμμα. (2 <sup>ο</sup> μέρος)	3 βδομάδες
Ανάγνωση Αισθητήρων	Ο φοιτητής προγραμματίζει τη διάταξη να αναγνωρίζει τους αναλογικούς αισθητήρες και να αναγνώσει τις τιμές.	1 βδομάδα
Εισαγωγή/ Ρύθμιση του PD	Οι φοιτητές καλούνται να συντονίσουν τους συντελεστές του αλγορίθμου PD, για διαφορετικές διαδρομές.	3 βδομάδες
Συνολική δοκιμασία	Στους φοιτητές δίνεται μια δοκιμασία με ακολουθία διαδρομής στο έδαφος, απλών κινήσεων και χρήση του επιπλέον μηχανισμού.	5 βδομάδες

### 1.3. Μεθοδολογία αξιολόγησης και σχεδιασμού

Ο σκοπός της ρομποτικής διάταξης είναι να διδάξει στους φοιτητές τις έννοιες της Μηχατρονικής και να βοηθήσει στην κατανόηση τους. Με βάση μια έρευνα [9], αποφασίστηκαν κάποια κριτήρια τα οποία θα οδηγήσουν στην τελική απόφαση. Τα αυτά είναι τα εξής:

1. Το Κόστος Αγοράς
2. Το Κόστος Συντήρησης
3. Η Διάρκεια ζωής/ Στιβαρότητα
4. Η Πολυμορφία
5. Η Επεκτασιμότητα
6. Η Χρησιμότητα στο μάθημα

Στο μάθημα της Μηχατρονικής έχει εισαχθεί ήδη η ρομποτική διάταξη, όμως με τις νέες απαιτήσεις αυτή δεν μπορεί να χρησιμοποιηθεί πλέον. Έτσι, μία εναλλακτική αποτελεί η αγορά μιας έτοιμης πλατφόρμας. Αυτό συνεπάγεται με αξιολόγηση της σχετικής με το αντικείμενο αγοράς, δηλαδή έτοιμες εμπορικές διατάξεις που πωλούνται.

Το κόστος αγοράς θα συμπεριλαμβάνει την τιμή της διάταξης και όλα τα πιθανά περιφερειακά που κρίνεται απαραίτητο να υπάρχουν. Αυτά κατηγοριοποιούνται σε:

1. Πλακέτα μικροελεγκτή ή μικροεπεξεργαστή

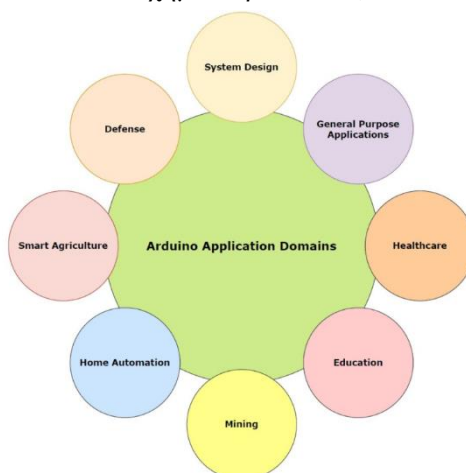
2. Αισθητήρες
3. Κινητήρες
4. Τροφοδοσία

Το κόστος συντήρησης θα αναφέρεται μόνο στα αναλώσιμα, όπως είναι οι μπαταρίες για την τροφοδοσία, και στα ευαίσθητα στοιχεία της διάταξης, όπως είναι οι αισθητήρες, και μηχανικά μέρη, όπως ρόδες και στηρίγματα που δέχονται καταπονήσεις. Η διάρκεια ζωής, ή στιβαρότητα, της διάταξης θα αξιολογηθεί με βάση τα στοιχεία που δίνουν οι κατασκευαστές για το προϊόν τους. Η πολυμορφία αφορά την δυνατότητα αλλαγής της κατασκευής για να την εκτέλεση διαφορετικών εργασιών με τα ίδια μηχανολογικά και ηλεκτρονικά στοιχεία, ενώ η επεκτασιμότητα είναι η ικανότητα της διάταξης να δεχτεί και άλλα μηχανολογικά κυρίως κομμάτια, τα οποία δεν είναι μέρος της αρχικής κατασκευής.

Στην αγορά κυκλοφορούν διάφορες έτοιμες διατάξεις, άλλες οικονομικές και άλλες λιγότερο. Για τον σκοπό της διπλωματικής έγινε επιλογή μερικών από αυτών, οι οποίες κρίθηκαν κατάλληλες για τουλάχιστον μία από τις παραπάνω προτεινόμενες ασκήσεις του Πίνακα 2. Επομένως, οι έτοιμες διατάξεις οφείλουν να είναι απλές κατασκευές με δυνατότητα επέκτασης για επιπλέον μηχανισμούς και αισθητήρες. Επίσης, πρέπει να είναι συμβατές με το μέγεθος των μικροεπεξεργαστών ή μικροελεγκτών που προτείνονται και να είναι διαθέσιμες στην Ελλάδα.

Για την συγκεκριμένη εργασία, αποφασίστηκε να επιλεγεί μια από τις τρεις πλακέτες που είναι διαθέσιμες. Αυτές είναι το Arduino Uno Rev3, από την Arduino, το Raspberry Pi 4 Model B, από την Raspberry Pi Foundation, και το ESP32, της Espressif. Τα τεχνικά χαρακτηριστικά φαίνονται στον πίνακα 3. Στον πίνακα αναγράφεται το είδος της κάθε πλακέτας, άμα είναι μικροελεγκτής ή μικροεπεξεργαστής. Ως μικροεπεξεργαστής ορίζεται ένας απλός επεξεργαστής που υπάρχει σε συσκευές όπως ένας ηλεκτρονικός υπολογιστής, και δεν μπορεί να λειτουργήσει ανεξάρτητα, χωρίς λογισμικό. Για αυτό το Raspberry χρειάζεται μία κάρτα μνήμης όπου βρίσκεται το λογισμικό. Από την άλλη, ένας μικροελεγκτής, έχει όλα τα απαραίτητα στοιχεία και μπορεί να λειτουργήσει ανεξάρτητα. [10] Είναι σχεδιασμένος για ελέγχει μία συγκεκριμένη διαδικασία.

Καθένα από τα παραπάνω έχει ένα ευρύ πεδίο εφαρμογών. Για αρχή το Arduino είναι από τις πιο διαδομένες πλακέτες, τόσο για έμπειρους επιστήμονες αλλά και για αρχάριους.[7] Έχει δικό του προγραμματιστικό περιβάλλον και χαμηλό κόστος. Το Arduino είναι μικροελεγκτής και χρησιμοποιείται για την διαχείριση συστημάτων και αισθητήρων. Μερικά πεδία εφαρμογής που εντοπίζεται είναι στην Εκπαίδευση, στον τομέα της Υγείας, στην Αναβάθμιση Σπιτιών, σε «Έξυπνες» Συσκευές και σε άλλα, όπως φαίνεται στο σχήμα παρακάτω. (εικόνα 1-2)



Εικόνα 1-2: Πεδία εφαρμογής του Arduino

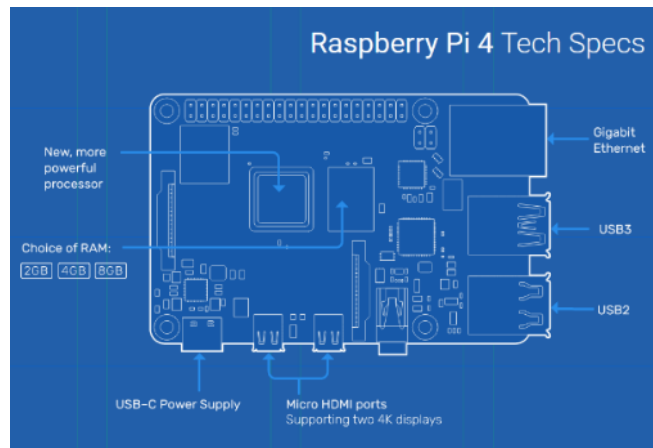
Το Raspberry είναι μικροεπεξεργαστής και σε αντίθεση με το Arduino, που απλά προγραμματίζεται, χρειάζεται λογισμικό σύστημα. Αυτό πρέπει να είναι ή Linux ή κάποιο από τα έτοιμα λογισμικά της εταιρείας.[11] Οι εφαρμογές του Raspberry είναι παρόμοιες με το Arduino με την διαφορά ότι μπορεί να ελέγχει παραπάνω από ένα συστήματα και έχει την δυνατότητα να χρησιμοποιεί κάμερα. Επίσης, το Raspberry μπορεί να χρησιμοποιηθεί και ως ηλεκτρονικός υπολογιστής με ότι αυτό συνεπάγεται.

Τέλος, το ESP32 είναι ένας μικροελέγκτης παρόμοιος με το Arduino. Όμως το ESP32 χρησιμοποιείται κυρίως σε εφαρμογές σχετικές με το Internet of Things (IoT). Αυτό είναι ένα δίκτυο συστημάτων τα οποία αποτελούνται από μηχανολογικά στοιχεία και αισθητήρες που επικοινωνούν μεταξύ τους. Το IoT βρίσκει εφαρμογές στην Βιομηχανία, στην Υγεία, στα Ενεργειακά συστήματα και σε άλλα.

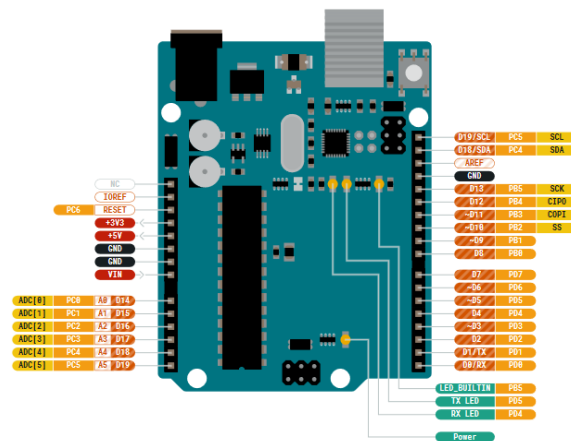
Στην συνέχεια, παρατίθενται τα τεχνικά χαρακτηριστικά των πλακετών σε μορφή πίνακα (πίνακας 1-3) καθώς και τα σχέδια (εικόνες 1-4, 1-5 και 1-6) που περιγράφουν κάποια βασικά στοιχεία.

Πίνακας 1-3: Τεχνικά χαρακτηριστικά

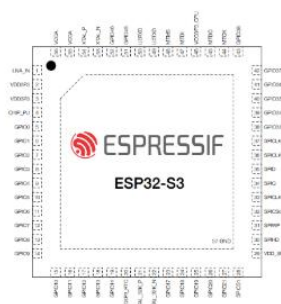
Χαρακτηριστικά/Πλακέτα	Raspberry Pi 4		
	Arduino Uno Rev 3[12]	Model B [13]	ESP32 [14]
Είδος	Μικροελεγκτής	Μικροεπεξεργαστής	Μικροελεγκτής
Μικροεπεξεργαστής	ATmega328P	Broadcom BCM2711, Quad core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz	Xtensa® dual-core 32-bit LX7 microprocessor
Τάση λειτουργίας	5 V	5V	3.3V
Τάση εισόδου	7 -12 V	5V	3.3V
Ψηφιακά I/O	14 (of which 6 provide PWM output)	13	39
Αναλογικά I/O	6	27	13
DC τάση ανά I/O	20mA	2A	40mA
DC τάση ανά 3.3V	50mA	--	50mA
Μνήμη Flash	32KB	32 KB eMMC on-board flash memory	
Στατική μνήμη	2KB	ανάλογα το μοντέλο	520KB
Clock speed	16MHz	1.4 GHz	
Λογισμικό λειτουργίας	Windows, Linux, Mac OSX	Linux	Windows, Linux, Mac OSX
Διαστάσεις πλακέτας	68.6mm x 53.4mm	86mm x 54mm x 17mm	7mm x 7mm x 0.9mm



Εικόνα 1-4: Σχεδιάγραμμα πλακέτας Raspberry



Εικόνα 1-5: Σχεδιάγραμμα πλακέτας Arduino



Εικόνα 1-3: Σχεδιάγραμμα πλακέτας ESP32

Σε αυτό σημείο, γίνεται η αξιολόγηση των επιλεγμένων εμπορικών ρομποτικών διατάξεων με βάση τα κριτήρια που οριστήκαν παραπάνω. Η αναλυτική αξιολόγηση φαίνεται στο παράρτημα. Στον Πίνακα 5 γίνεται αξιολόγηση ξεχωριστά σε κάθε διάταξη με βάση κάποιες παραδοχές. Στο κόστος αγοράς και συντήρησης, η αξιολόγηση έγινε χρησιμοποιώντας τον εξής πίνακα:

Πίνακας 1-4: Κλίμακα αξιολόγησης κόστους

Κόστος αγοράς	Κόστος συντήρησης
Χαμηλό:<100€	Χαμηλό: <50€
Μέτριο: 100€-200 €	Μέτριο: 50€-100€
Μεγάλο: >200 €	Μεγάλο: >100€

Επίσης, για την δυσκολία της επεκτασιμότητας των διαφόρων διατάξεων ισχύει ότι μπορεί να προστεθούν αισθητήρες και μηχανισμοί, αλλά ιδιαίτερη προσοχή χρειάζεται ώστε να είναι προσαρμοσμένοι στην κατασκευή. Για την χρήση στις εργαστηριακές ασκήσεις του μαθήματος Μηχατρονικής, η αξιολόγηση γίνεται με βάση τα προτεινόμενα σενάρια από τον πίνακα 2.

Πίνακας 1-5: Αρχική Αξιολόγηση Εμπορικών ρομποτικών διατάξεων

Μοντέλο	Arduino/ Raspberry/ ESP	Κόστος Αγοράς	Κόστος Συντήρησης	Διάρκεια Ζωής	Πολυμορφία	Δυσκολία Επεκτασιμότητας	Χρήση στο μάθημα
	<i>Actobotics kit- ActoBitty 2</i>	Arduino	Μέτριο	Μέτριο	Μεγάλη	Από μόνο του δεν υπάρχει δυνατότητα	Μικρή
<i>Mini round robot Chassis Kit- 2WD w DC Motors</i>	Όλες	Χαμηλό	Χαμηλό-Μέτριο	Μέτρια	ομοίως	Μικρή	1
<i>Mini robot rover chassis Kit- 4WD w DC Motors</i>	Όλες	Χαμηλό	Χαμηλό-Μέτριο	Μέτρια	ομοίως	Μικρή	1
<i>Mini robot rover chassis Kit- 2WD w DC Motors</i>	Όλες	Χαμηλό	Χαμηλό-Μέτριο	Μέτρια	ομοίως	Μικρή	1

Πίνακας 1-6: Αρχική Αξιολόγηση Εμπορικών ρομποτικών διατάξεων (συνέχει πίνακα 1-5)

		Arduino/ Raspberry/					Χρήση	
<i>Μοντέλο</i>		Κόστος	Κόστος	Διάρκεια		Δυσκολία	στο	
		Αγοράς	Συντήρησης	Ζωής	Πολυμορφία	Επεκτασιμότητας	μάθημα	
		ESP						
<i>Mini 3-layer round robot chassis kit-2WD w DC motors</i>		Όλες	Μέτριο	Χαμηλό-Μέτριο	Μέτρια	ομοίως	Μέτρια	1-2
<i>AlphaBot</i>		Όλες	Μέτριο	Μέτριο	Μικρή-Μέτρια	ομοίως	Μέτρια	1-2
<i>PiRacer DonkeyCar, AI Racing Robot for Raspberry Pi 4</i>		Raspberry	Μεγάλο	Μεγάλο	Μεγάλη	όχι πολύ μεγάλη, γιατί είναι έτοιμη λύση	Μεγάλη	1-2
<i>PiRacer Pro High Speed, AI Racing Robot for Raspberry Pi 4</i>		Raspberry	Μεγάλο	Μεγάλο	Μεγάλη	ομοίως	Μεγάλη	1-2
<i>AlphaBot2 Robot Building Kit for Raspberry Pi 3 Model B</i>		Raspberry	Μέτριο-Μεγάλο	Μέτριο-Μεγάλο	Μέτρια-Μεγάλη	ομοίως	Μικρή -Μέτρια	1-3

Ωστόσο, αυτή η αξιολόγηση μόνο δεν είναι ικανοποιητική και χρειάζεται να γίνει ποσοτικοποίηση του πίνακα 5. Με τη βοήθεια του πίνακα 6, δημιουργείται η τελική βαθμολογία της κάθε διάταξης, όπως φαίνεται στον πίνακα 7.

Πίνακας 1-7: Κλίμακα για την ποσοτικοποίηση

**Κλίμακα**

**1: Απολύτως ακατάλληλο**

**2: Ελαφρώς ακατάλληλο**

**3: Ενδιάμεση καταλληλότητα**

**4: Ελαφρώς κατάλληλο**

**5: Απολύτως κατάλληλο**

Πίνακας 1-8: Διαμόρφωση τελικής βαθμολογίας μετά την ποσοτικοποίηση

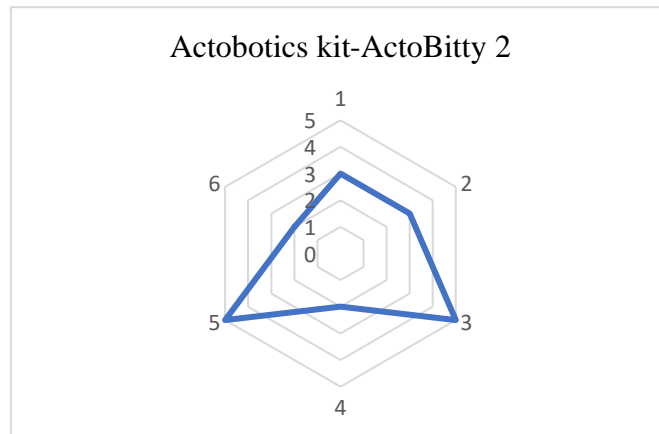
<i>Μοντέλο</i>	<b>Κόστος Αγοράς</b>	<b>Κόστος Συντήρησης</b>	<b>Διάρκεια Ζωής</b>	<b>Πολυμορφία</b>	<b>Δυσκολία Επεκτασιμότητας</b>	<b>Χρήση στο μάθημα</b>	<b>Τελική βαθμολογία</b>
<i>Actobotics kit- ActoBitty 2</i>	3	3	5	2	5	2	20
<i>Mini round robot Chassis Kit- 2WD w DC Motors</i>	5	4	3	1	5	3	21
<i>Mini robot rover chassis Kit- 4WD w DC Motors</i>	5	4	3	1	5	2	20
<i>Mini robot rover chassis Kit- 2WD w DC Motors</i>	5	4	3	1	5	2	20



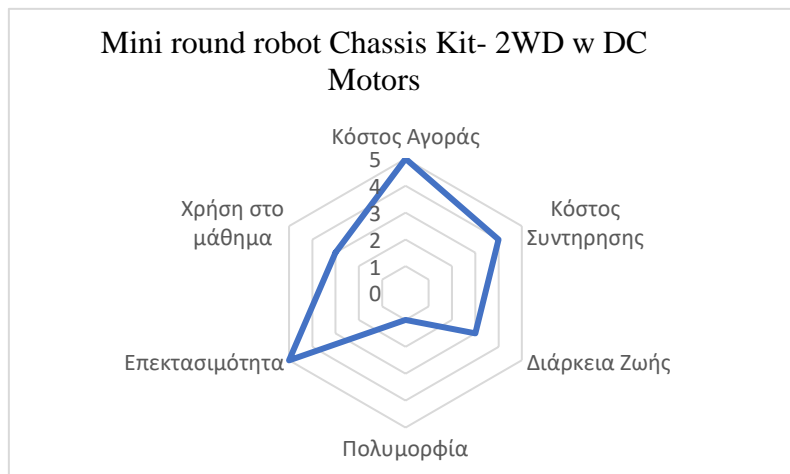
Πίνακας 1-9: Διαμόρφωση τελικής βαθμολογίας μετά την ποσοτικοποίηση (συνέχεια πίνακα 8)

<i>Μοντέλο</i>	<b>Κόστος Αγοράς</b>	<b>Κόστος Συντήρησης</b>	<b>Διάρκεια Ζωής</b>	<b>Πολυμορφία</b>	<b>Δυσκολία Επεκτασιμότητας</b>	<b>Χρήση στο μάθημα</b>	<b>Τελική βαθμολογία</b>
<i>Mini 3-layer round robot chassis kit- 2WD w DC motors</i>	3	4	3	2	3	3	18
<i>AlphaBot</i>	3	3	2	3	3	3	17
<i>PiRacer DonkeyCar, AI Racing Robot for Raspberry Pi 4</i>	1	1	5	4	1	4	16
<i>PiRacer Pro High Speed, AI Racing Robot for Raspberry Pi 4</i>	1	1	5	4	1	4	16
<i>AlphaBot2 Robot Building Kit for Raspberry Pi 3 Model B</i>	2	2	4	3	4	3	18

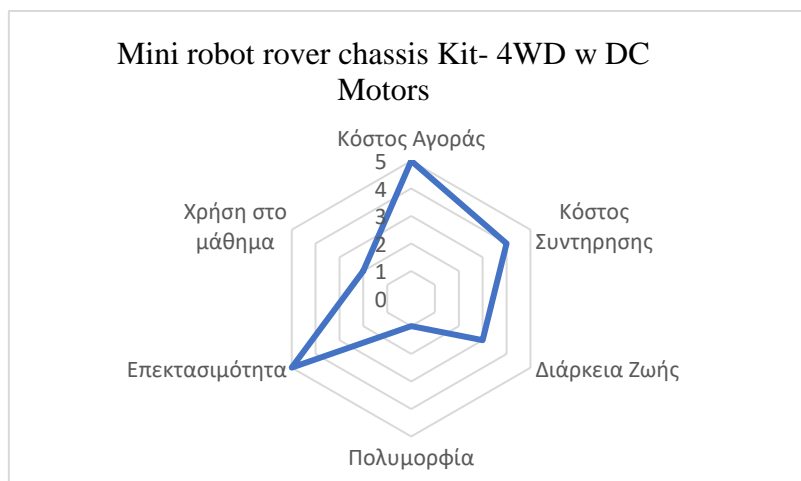
Για την καλύτερη κατανόηση των αποτελεσμάτων της αξιολόγησης παρατίθενται τα παρακάτω διαγράμματα.



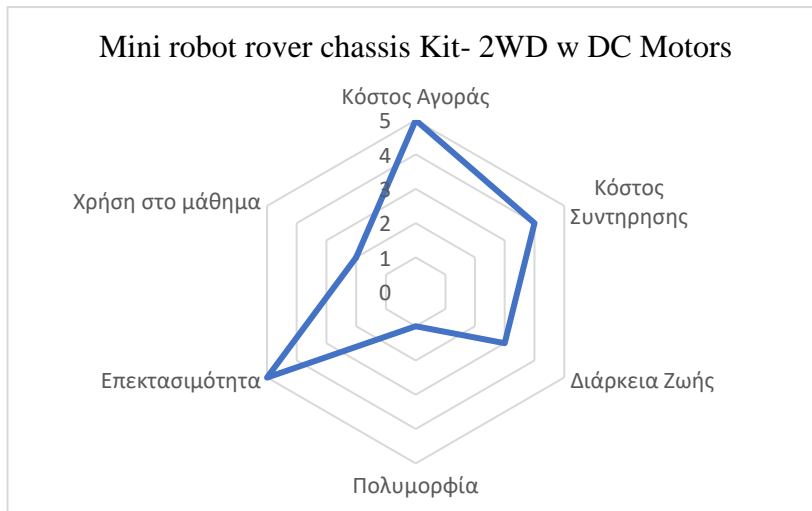
Διάγραμμα 1-1: Radar Chart για το μοντέλο Actobotics kit-ActoBitty 2



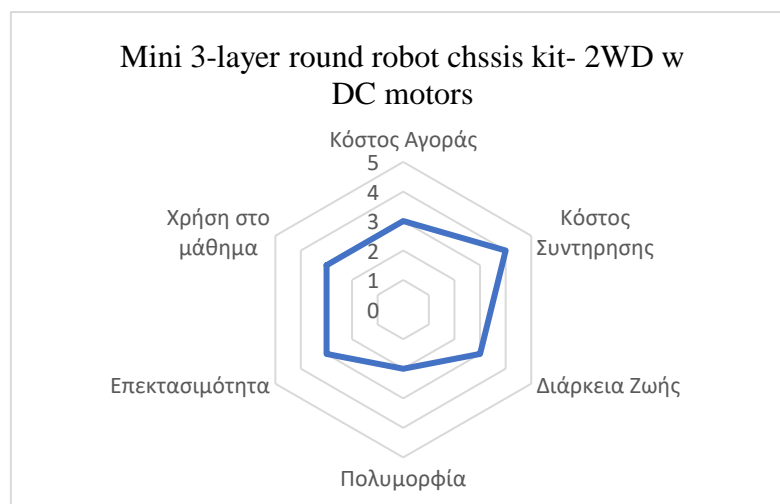
Διάγραμμα 1-2: Radar Chart για το μοντέλο Mini round robot Chassis Kit- 2WD w DC Motors



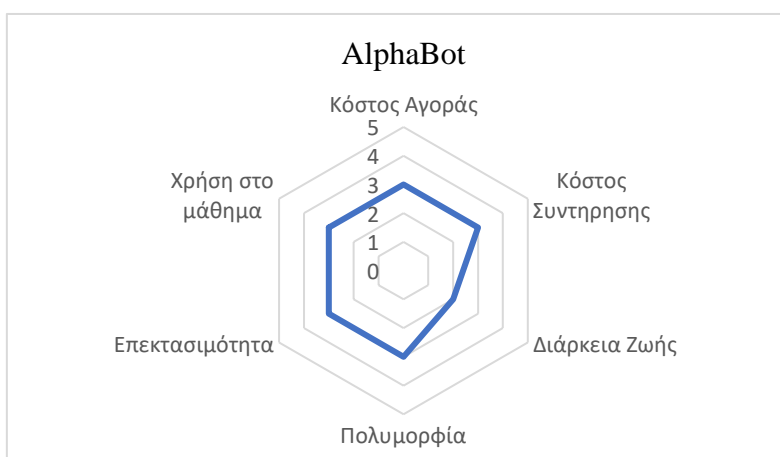
Διάγραμμα 1-3: Radar Chart για το μοντέλο Mini robot rover chassis Kit- 4WD w DC Motors



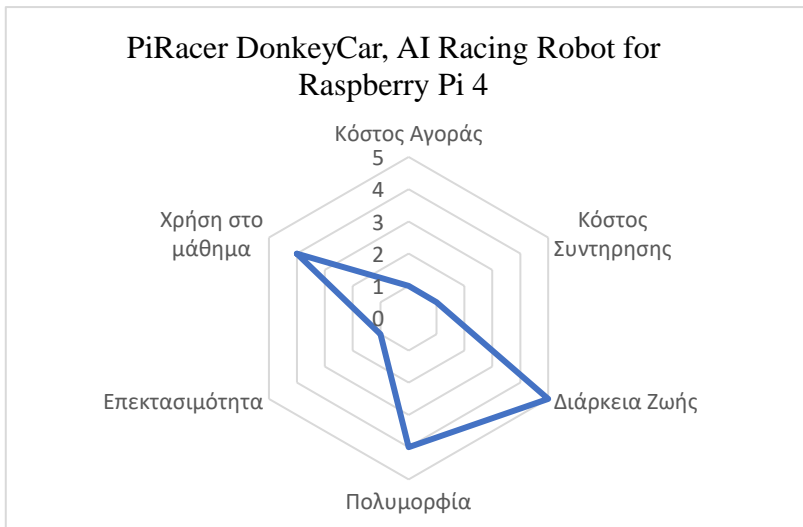
Διάγραμμα 1-4: Radar Chart για το μοντέλο Mini robot rover chassis Kit-2WD w DC Motors



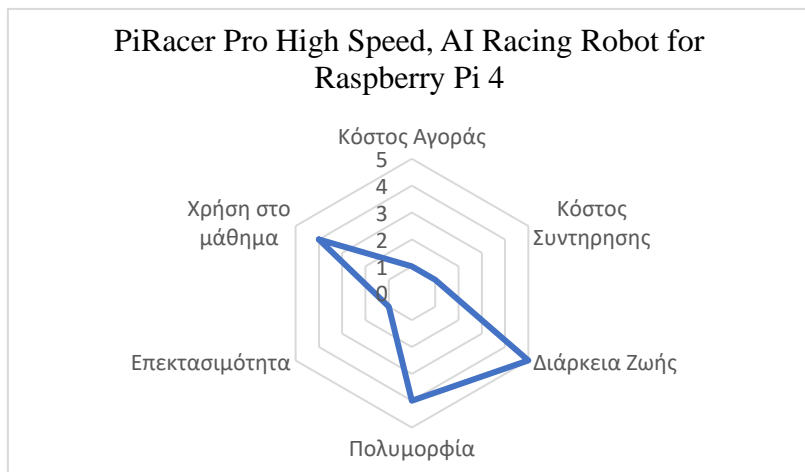
Διάγραμμα 1-5: Radar Chart για το μοντέλο Mini 3-layer round robot chassis kit- 2WD w DC motors



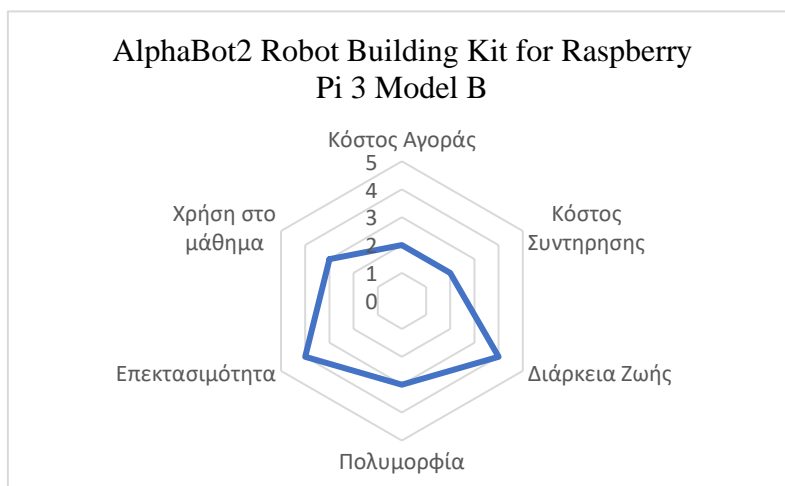
Διάγραμμα 1-6: Radar Chart για το μοντέλο AlphaBot



Διάγραμμα 1-7: Radar Chart για το μοντέλο PiRacer DonkeyCar, AI Racing Robot for Raspberry Pi 4

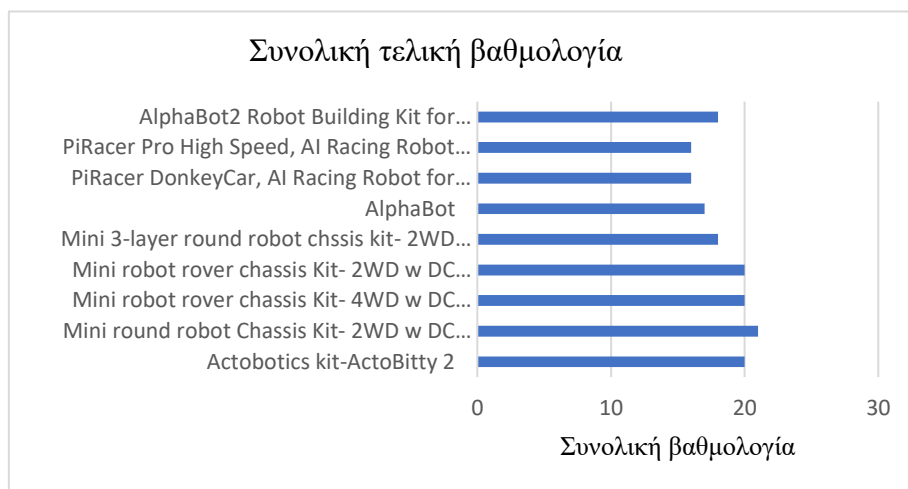


Διάγραμμα 1-8: Radar Chart για το μοντέλο PiRacer Pro High Speed, AI Racing Robot for Raspberry Pi 4



Διάγραμμα 1-9: Radar Chart για το μοντέλο AlphaBot2 Robot Building Kit for Raspberry Pi 3 Model B

Με βάση τον πίνακα 7, δημιουργείται και το τελικό διάγραμμα με την τελική βαθμολογία για την αξιολόγηση των εμπορικών ρομποτικών διατάξεων.



Διάγραμμα 1-10: Συνολική τελική βαθμολογία από την αξιολόγηση

Για τον σχεδιασμό της ρομποτικής διάταξης η μεθοδολογία προσαρμόζεται κατάλληλα. Αρχικά, ο σκοπός της είναι η πολυμορφία και η δυνατότητα επέκτασης της ρομποτικής διάταξης. Στις περισσότερες διαθέσιμες εμπορικές πλατφόρμες η πολυμορφία τους περιορίζεται στα διαθέσιμα εξαρτήματα της ίδιας σειράς προϊόντων και εταιρείας. Η νέα διάταξη πρέπει να είναι συμβατή με διαφορετικά υλικά από διαφορετικές σειρές και εταιρείες. Το σχέδιο του βασικού σώματος της κατασκευής χρειάζεται να είναι απλό, για να μπορέσει να είναι συμβατό με όλα τα απαραίτητα υλικά. Με βάση τα διαθέσιμα προϊόντα, επιλέγεται η διάταξη να έχει δύο πλάκες, μία πάνω για τα ηλεκτρονικά και μία κάτω για την βασική κίνηση. Ωστόσο είναι εξίσου σημαντικό να μειωθεί όσο το δυνατόν περισσότερο το τελικό κόστος αγοράς αλλά και συντήρησης.

Επομένως, με βάση τα παραπάνω, αλλά και με τα κριτήρια της αξιολόγησης, η μεθοδολογία του σχεδιασμού ακολουθεί την παρακάτω διαδικασία. Πρώτον, γίνονται οι κατάλληλες παραδοχές, κυρίως όσον αφορά την πολυμορφία και τις τελικές διαστάσεις της διάταξης. Στην συνέχεια, επιλέγονται τα είδη των βασικών κινητήρων, για να γίνει ο αρχικός σχεδιασμός και παράλληλα, είναι απαραίτητο να γίνει η επιλογή της μορφής της τελικής διάταξης σχετικά με τα είδη αυτοκινούμενου. Στην συγκεκριμένη περίπτωση, δίνονται δύο πιθανές επιλογές, να έχει τρεις τροχούς (δύο κινητήριους και ένα βοηθητικό) ή να έχει τέσσερις (δύο κινητήριους και δύο βοηθητικούς), καθώς αυτοί κρίνονται κατάλληλοι για τις εργασίες που προτείνονται. Το επόμενο σημείο ενδιαφέροντος είναι ο σχεδιασμός της πάνω πλάκας, η οποία θα έχει τον κατάλληλο χώρο για την ηλεκτρονική διάταξη. Σημαντικό είναι να γίνει κατάλληλη μετατροπή του αρχικού, άμα χρειάζεται, για να μπορέσουν να τοποθετηθούν οι αισθητήρες. Συνεπώς, ακολουθώντας τα παραπάνω ξεκινάει ο σχεδιασμός και η τελική υλοποίηση της πλατφόρμας, μια διαδικασία η οποία αναλύεται εκτενώς σε αυτή την εργασία.

## Κεφάλαιο 2ο Σχεδιασμός της διάταξης

Ένα ρομπότ έχει δημιουργηθεί με σκοπό να εκπληρώσει κάποιες εργασίες. Αυτές μπορεί να περιλαμβάνουν πολύπλοκες κατεργασίες σε κάποιο προϊόν σε κάποια γραμμή παραγωγής ή πιο απλές διαδρομές που μπορεί να κάνει ένα οικιακό ρομπότ σκούπα. Σε κάθε περίπτωση, είναι εμφανές ότι μία κατασκευή δεν μπορεί να ικανοποιήσει κάθε ανάγκη. Ωστόσο, κάθε διαδικασία μπορεί να κατηγοριοποιηθεί και σε κάθε κατηγορία υπάρχει μια βασική κατασκευή που ξεχωρίζει.

Στα πλαίσια του μαθήματος της Μηχανικής αλλά και της παρούσας εργασίας, αποφασίστηκε να ακολουθηθεί η στρατηγική της προηγούμενης εργασίας: να κατασκευαστεί και να προγραμματιστεί μια ρομποτική πλατφόρμα με μικροελεγκτή σε μορφή τροχοκίνητου οχήματος. Συγκεκριμένα, ορίζεται ως απαραίτητη προϋπόθεση για την αναβάθμιση της πλατφόρμας η πολυμορφία της. Σκοπός της αναβαθμισμένης πλατφόρμας είναι επίσης και η ευκολία στην χρήση και αλλά και το χαμηλό κόστος, όπως αναλύθηκε και σε προηγούμενο κεφάλαιο.



Εικόνα 2-1: LEGO Education SPIKE Prime [50]

Στο παρόν κεφάλαιο θα αναλυθεί η νέα κατασκευή της ρομποτικής πλατφόρμας. Σε αυτό το σημείο οφείλεται να υπογραμμιστεί ότι η προσπάθεια αυτή γίνεται με σκοπό όχι απλά την αναβάθμιση της πλατφόρμας αλλά και τον εκσυγχρονισμό όλων των στοιχείων. Ωστόσο, η συνθήκη της πολυμορφίας μια ιδιαίτερα πλατιά έννοια. Για αυτό πριν παρουσιαστεί η κατασκευή, όπως τελικά διαμορφώθηκε και αναλυθούν τα χαρακτηριστικά, ορίζονται κάποιοι απλοί και βασικοί περιορισμοί και παραδοχές.

### 2.1. Παραδοχές

#### 2.1.1. Πολυμορφία

Η πραγματική πολυμορφία μιας τέτοιας πλατφόρμας έγκειται συνήθως στο πλήθος των εξαρτημάτων από τα οποία αυτή αποτελείται. Όμως, όπως είναι φυσικό το να σχεδιάσει κάποιος μια

σειρά τέτοιων εξαρτημάτων είναι ιδιαίτερα χρονοβόρο. Επίσης απαιτεί εμπειρία και αρκετές δοκιμές, ώστε να είναι βέβαιη η ποιότητα των εξαρτημάτων. Ένα παράδειγμα μιας τέτοιας σχεδόν πλήρους πολυμορφικής πλατφόρμας είναι η σειρά της LEGO Mindstorms, η οποία αποτελείται από πολλά διαφορετικά εξαρτήματα, όπως φαίνεται και στις επόμενες φωτογραφίες. Ωστόσο, για την ρομποτική διάταξη επιλέχθηκε μια πιο απλή εκδοχή της πολυμορφίας.

Το τελικό ρομπότ αποτελείται από δύο (2) πλάκες, μία πάνω και μία κάτω, οι οποίες φέρουν οπές σε διάφορα σημεία για να μπορέσουν να ενωθούν διαφορετικά υλικά του εμπορίου, όπως κινητήρες, αισθητήρες αλλά και κάποιοι μηχανισμοί. Η επιλογή του σχήματος, ορθογώνια κατασκευή, έγινε ακολουθώντας την τάση των εμπορικών ρομποτικών διατάξεων, καθώς πέρα από σχεδιαστική ευκολία διαθέτει και τον απαραίτητο χώρο για την τοποθέτηση εξοπλισμού, όπως μηχανισμοί αλλά και διαφορετικές ηλεκτρονικές διατάξεις.

Τα βασικά κομμάτια του ρομπότ κατασκευάζονται με την χρήση ενός 3D printer, αλλά το υλικό μπορεί να αλλάξει χωρίς μεταβολές στο αρχικό σχέδιο. Βέβαια, η βασική λογική της πλατφόρμας δεν θα αλλάξει. Είναι ένα τροχοκίνητο όχημα, το οποίο μπορεί να έχει κάποια από τα χαρακτηριστικά που αναλύονται στην συνέχεια.



*Εικόνα 2-2: Τελική προτεινόμενη κατασκευή, δύο κινητήρες DC, ένα κινητήρα servo σε μηχανισμό gripper, δύο αισθητήρες TCRT5000, δύο οπτικούς αισθητήρες μέτρησης περιστροφών, την πλακέτα Maker Pi RP2040 και μία μπαταρία 3.7 V (1200mAh)*

### 2.1.2. Διαστάσεις

Σκόπιμο θα είναι να γίνει αναφορά και στις τελικές διαστάσεις του ρομπότ. Για να μπορεί να είναι συμβατό το τελικό αποτέλεσμα με αρκετά από τα εξαρτήματα και υλικά του εμπορίου, οι διαστάσεις του είναι σχετικά κοντά σε αυτές μίας πλατφόρμας του εμπορίου. Από τον πίνακα (του προηγούμενου κεφαλαίου) παρατηρείται ότι οι συνήθεις διαστάσεις, με μερικές εξαιρέσεις, είναι μεταξύ 140x100 mm για ορθογώνιες κατασκευές και περίπου 120 mm διάμετρος για κυκλικές κατασκευές.

Εκ των άλλων, είναι σκόπιμο να γίνει αναφορά στον χρονικό ορίζοντα κατά τον οποίο πραγματοποιήθηκε η προσπάθεια αυτή. Όπως αναφέρεται και παραπάνω, η πολυμορφία απαιτεί ένα πλήθος ξεχωριστών εξαρτημάτων. Ωστόσο για να βρεθούν οι κατάλληλες διαστάσεις και να συνδέονται σωστά τα κομμάτια, θα χρειαζόταν περισσότερος χρόνος από τον διαθέσιμο. Αυτό συμβαίνει, γιατί ενώ το πρόγραμμα έχει την δυνατότητα να δείχνει τα κομμάτι σε προσομοίωση, όταν αυτά εκτυπώνονται ή

γενικά κατασκευάζονται υπάρχουν μικρές αλλαγές στις διαστάσεις. Ο λόγος για αυτό το πρόβλημα είναι τα εργαλεία, καθώς επιλέχθηκε να γίνει η κατασκευή με το μικρότερο δυνατό κόστος, κάτι το οποίο συζητήθηκε στο προηγούμενο κεφάλαιο.

## **2.2. Τεχνολογία τρισδιάστατης εκτύπωσης**

### 2.2.1. Τι είναι μια 3D εκτύπωση

Οι τρισδιάστατοι εκτυπωτές ή, όπως είναι γνωστοί, 3D printers είναι το βασικό εργαλείο που χαρακτηρίζει τον τομέα των προσθετικών κατασκευών. Η λογική του είναι αρκετά απλή, αφού είναι η αντίθετη με τις κατεργασίες αφαίρεσης υλικού. Στην πρώτη περίπτωση, ένα τρισδιάστατο αντικείμενο δημιουργείται με την προσθήκη στρωμάτων υλικού, ενώ στην δεύτερη το αντικείμενο προκύπτει από την αφαίρεση υλικού από ένα κομμάτι της επιλεγμένης πρώτης ύλης. [15]

### 2.2.2. Ιστορία των τρισδιάστατων εκτυπωτών

Η ιδέα της Προσθετικής Κατασκευής (αγγλικά Additive Manufacturing, AM) εμφανίστηκε πριν από σχεδόν 150 χρόνια, με βασικά σημεία εφαρμογής την κατασκευή τοπογραφικών χαρτών ελεύθερης μορφής και φωτογλυπτικών από δισδιάστατα (2D) στρώματα.

Με την πρόοδο της τεχνολογίας, ακολούθησε η εξέλιξη προγραμμάτων σχεδιασμού και κατασκευής με τη βοήθεια υπολογιστή (CAD). Σε αυτά συμπεριλαμβάνονται οι εργαλειομηχανές αριθμητικού ελέγχου, που εμφανίστηκαν στις αρχές της δεκαετίας του 1950, γραφικών υπολογιστών και εργαλείων CAD, στις αρχές της δεκαετίας του 1960 και συστημάτων CAD / CAM στα τέλη της δεκαετίας του 1960. Εξίσου σημαντική εξέλιξη για την διάδοση των τρισδιάστατων εκτυπωτών αποτέλεσε και η ευρεία διαθεσιμότητα υπολογιστών λόγω πλέον χαμηλού κόστους αγοράς από τις αρχές της δεκαετία του 1970.

Στα επόμενα χρόνια, οι εκτυπωτές ήταν πλέον προσιτοί και στο ευρύ κοινό, με τις νέες γενιές εμπορικών συστημάτων να προσφέρουν νέα και βελτιωμένα χαρακτηριστικά. Η ποιότητα βελτιώθηκε σε σημείο που οι τεχνολογίες προσθετικής κατασκευής θα μπορούσαν να χρησιμοποιηθούν για την παραγωγή τελικών εξαρτημάτων. [16] Οι φτηνοί, ισχυροί υπολογιστές βοήθησαν να γίνουν οι νέες γενιές εκτυπωτών ακόμα μικρότερες και πιο προσιτές.

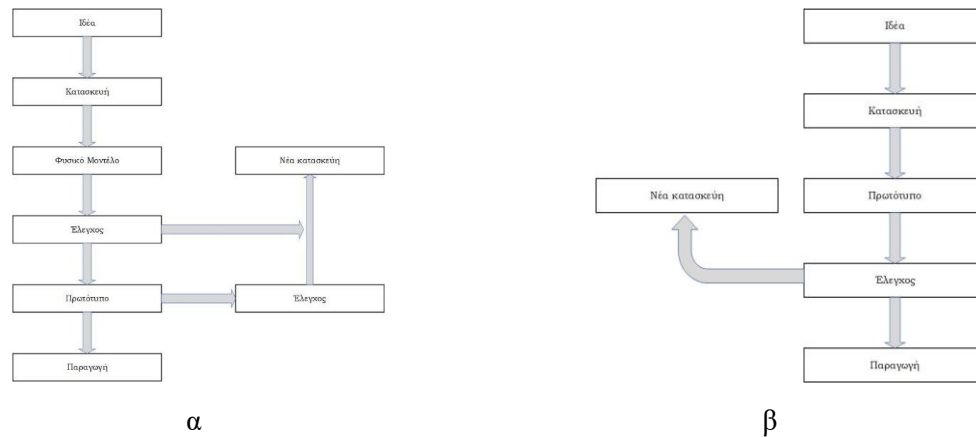
Ένα σημείο που βοήθησε στην ανάπτυξη της αγοράς ήταν η λήξη κάποιων βασικών δικαιωμάτων ευρεσιτεχνίας για ορισμένες παλαιότερες διαδικασίες. Αυτό, σε συνδυασμό με μια αναπτυσσόμενη κοινότητα με ερασιτεχνικό ενδιαφέρον από το ευρύ κοινό, ώθησε στην καινοτομία, οδηγώντας σε σημαντική επέκταση της προσφοράς και της ζήτησης της αγοράς. Σήμερα, τα προϊόντα και οι υπηρεσίες της προσθετικής κατασκευής υποστηρίζουν ένα ευρύ φάσμα δραστηριοτήτων, όπως η ενέργεια, οι μεταφορές, η τέχνη, η αρχιτεκτονική, η εκπαίδευση, τα χόμπι και άλλα.

### 2.2.3. Γιατί επιλέχθηκε η 3D εκτύπωση

Οι λόγοι για τους οποίους επιλέχθηκε αυτός ο τρόπος κατασκευής είναι αρκετοί. Αρχικά οι τρισδιάστατοι εκτυπωτές χρησιμοποιούνται για την ταχύτερη δημιουργία πρωτοτύπων κατασκευών. [17] Η κατασκευή ενός αντικειμένου μπορεί να γίνει «από το μηδέν» ή να «προστεθεί» ένα επιπλέον κομμάτι. Τα αντικείμενα που δημιουργούνται ως πρωτότυπα κατασκευάζονται και σχεδιάζονται ταχύτερα από ότι άμα είχε επιλεγεί μια παραδοσιακή μέθοδος. [18] Ένα βοηθητικό διάγραμμα δίνεται στην συνέχεια όπου περιγράφει την διαφορά μεταξύ της χρήσης τρισδιάστατου εκτυπωτή και παραδοσιακής μεθόδου για την δημιουργία ενός αντικειμένου.



Ένα ακόμα χαρακτηριστικό των εκτυπωτών είναι δεν αυξάνεται το κόστος των αντικειμένων με την πολυπλοκότητα του σχεδίου τους. Είναι αρκετά εύκολο να προσαρμοστούν τα σχέδια στις ανάγκες της κάθε εργασίας και είναι δυνατόν να εκτυπωθούν σχήματα και γεωμετρίες που δυσκολεύονται να κατασκευαστούν μέσω αφαιρετικών τεχνικών κατασκευής. Η προσθετική κατασκευή παρέχει υψηλό βαθμό ελευθερίας για καινοτόμες σχεδιαστικές λύσεις. Τόσο από τον ακαδημαϊκό χώρο όσο και από τη βιομηχανία, έχουν δημοσιευτεί πολλά προϊόντα που αξιοποιούν αυτές τις δυνατότητες.[19]



Διάγραμμα 2-1: Στο διάγραμμα 1.α. απεικονίζεται η διαδικασία κατασκευής για τις κλασικές τεχνικές, ενώ στο 1.β απεικονίζεται η αντίστοιχη διαδικασία για την τριδιάστατη εκτύπωση

Επίσης, οι εκτυπωτές 3D προσφέρουν πολλές ευκαιρίες σχεδιασμού και ελευθερίες. Σημαντικότερα, για ένα αντικείμενο που θα κατασκευαστεί είναι πιθανό να επιλεχθούν τα υλικά (PLA, ABS, μέταλλο κτλ) και το χρώμα του. [16]Ακόμα, όπως προαναφέρθηκε, υπάρχουν ελάχιστοι έως καθόλου περιορισμοί για την γεωμετρία του αντικειμένου. Είναι πιθανό να γίνει άμεση ενοποίηση των εξαρτημάτων, άμα είναι απαραίτητο, καθώς και να τυπωθούν ενσωματωμένα ηλεκτρονικά κυκλώματα.

Οι εκτυπωτές 3D μπορούν να θεωρηθούν περιβαλλοντικά βιώσιμοι, επειδή χρησιμοποιούν λιγότερες πρώτες ύλες, παράγουν λιγότερα απόβλητα υποπροϊόντα και καταναλώνουν λιγότερη ενέργεια, σε σχέση με άλλες τεχνικές κατασκευής. [20] Η ανάπτυξη φιλικών προς το περιβάλλον τεχνολογιών είναι απαραίτητη καθώς ο παγκόσμιος πληθυσμός αυξάνεται, γίνεται πλουσιότερος και απαιτεί περισσότερα αγαθά. Αυτό θα οδηγήσει σε αυξανόμενη χρήση των ήδη περιορισμένων πόρων.

#### 2.2.4. Τριδιάστατη εκτύπωση στην εκπαίδευση

Η τριδιάστατη εκτύπωση είναι κυρίως συνδεδεμένη με την βιομηχανία. Ωστόσο, τα τελευταία χρόνια έχει γίνει προσπάθεια να ενταχθεί σε διάφορα προγράμματα σπουδών και στις περισσότερες περιπτώσεις έχει γίνει με μεγάλη επιτυχία. Αυτό που οδήγησε στην ένταξη τους στην εκπαίδευση είναι η ανάγκη των σχολών να ακολουθήσουν την εξέλιξη της βιομηχανίας και να προετοιμάζουν καλύτερα τους φοιτητές για την αγορά εργασίας.

Στον τομέα της μηχανικής, η εκπαίδευση οφείλει να προωθεί την διεπιστημονικότητα, την τόνωση της πρωτοβουλίας και της δημιουργικότητας. Η διεπιστημονική εκπαίδευση διευκολύνει την επίλυση των προβλημάτων επαγγελματικής μετατροπής.[21] Έτσι είναι εύκολο κανείς να κατανοήσει την σημασία που έχει η χρήση ενός τριδιάστατου εκτυπωτή στην δημιουργία ενός μηχανικού συστήματος.

Εκτός από την βιομηχανία, ο εκτυπωτής 3D σε εκπαιδευτικό επίπεδο μπορεί να χρησιμοποιηθεί σε διάφορους τομείς. Μερικά ιδιαίτερα παραδείγματα αποτελούν η αναπαράσταση ιστορικών

αντικείμενων για να εξετασθούν χωρίς να κινδυνεύει το αυθεντικό.[22] Σε άλλες περιπτώσεις είναι δυνατόν να εκτυπωθούν τοπογραφικοί, δημογραφικοί ή πληθυσμιακοί χάρτες (τομέας γεωγραφίας), 3D μοντέλα μορίων (τομέας χημείας) ή κύτταρα, ιούς, όργανα και άλλα βιολογικά αντικείμενα (τομέας βιολογίας).

### 2.2.5. Επιλογή υλικού

Για την κατασκευή της πολυμορφικής ρομποτικής επιλέχθηκε το υλικό PLA (Polylactic Acid). Ένας λόγος είναι γιατί το υλικό αυτό είναι άμεσα διαθέσιμο στο εμπόριο, κάτι που το καθιστά αρκετά οικονομικό, και προσφέρεται σε μία ποικιλία χρωμάτων. [23] Διαθέτει αρκετά καλές μηχανικές ιδιότητες, αν και δεν είναι ένα ιδιαίτερα σημαντικό χαρακτηριστικό, καθώς δεν προβλέπεται η διάταξη να καταπονηθεί από τους μηχανισμούς της. Ωστόσο, το υλικό είναι αρκετά ανθεκτικό και θεωρείται ότι θα ανταπεξέλθει επαρκώς στις ανάγκες της κατασκευής. Επίσης, είναι μια οικολογική λύση, αφού αποτελεί πλέον ένα βιοδιασπώμενο πλαστικό. [24]

## **2.3. Σχέδια για την κατασκευή**

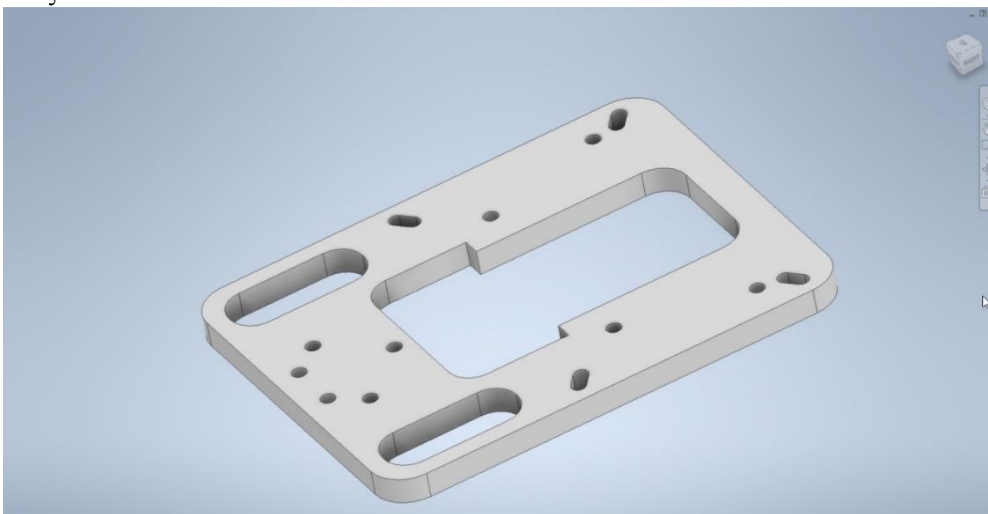
Σύμφωνα με τις παραπάνω παραδοχές δημιουργούνται τα εξής κομμάτια της ρομποτικής πλατφόρμας:

- Η κοινή πάνω πλάκα
- Η κάτω πλάκα για κινητήρες DC
- Η κάτω πλάκα για κινητήρες Servo
- Τα στηρίγματα για κινητήρες DC

Για τον σχεδιασμό της πλατφόρμας χρησιμοποιήθηκε το πρόγραμμα της Autodesk, Autodesk Inventor Professional 2022. Τα αναλυτικά σχέδια δίνονται στο παράρτημα Β.

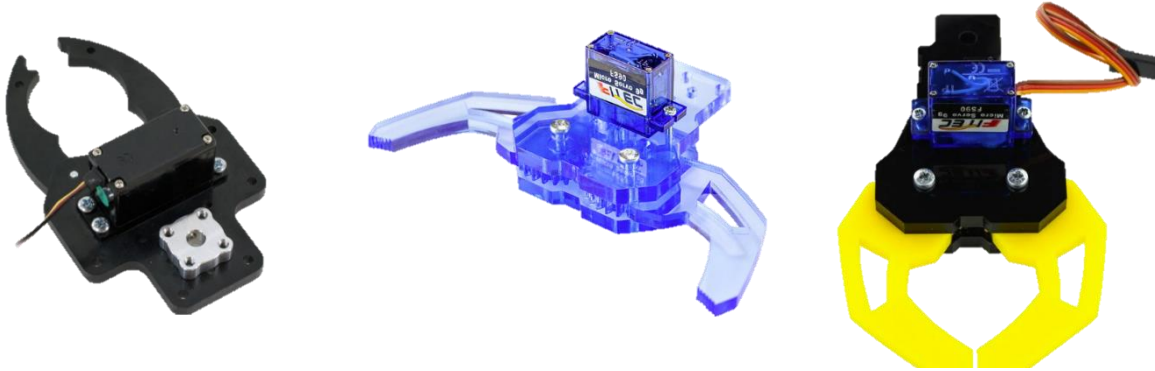
### 2.3.1. Πάνω πλάκα

Στην πάνω πλάκα της πλατφόρμας θα χρειαστεί σε κάθε περίπτωση να τοποθετηθούν τα ηλεκτρονικά συστήματα, όποια και αν επιλεγθούν. Παράλληλα πρέπει να είναι ένα απλό σχέδιο και έχει αρκετό χώρο για τυχόν μηχανισμούς. Επίσης καλό θα είναι να έχει όσο περισσότερα κενά γίνεται για την εξοικονόμηση χρόνου και υλικού. Επομένως με βάση τα προηγούμενα το τελικό σχέδιο της πάνω πλάκας είναι το ακόλουθο:



Εικόνα 2-3: Φωτογραφία από το πρόγραμμα Inventor Pro 2022, όπου απεικονίζεται η κοινή πάνω πλάκα.

Το συγκεκριμένο σχέδιο καλύπτει όλες τις ανάγκες όπως αυτές ορίστηκαν προηγουμένως. Αρχικά οι γενικές διαστάσεις είναι 130mm x 100mm x 7mm. Οι τέσσερις οπές που είναι συγκεντρωμένες στο «μπροστινό μέρος» είναι εκεί με σκοπό να μπορούν να τοποθετηθούν μηχανισμοί όπως αυτοί που ακολουθούν στην συνέχεια, οι οποίοι είναι διαθέσιμοι στο εμπόριο. Ταυτόχρονα τα κενά που υπάρχουν χρησιμεύουν στην ευκολότερη τακτοποίηση των καλωδίων και στην πρόσβαση τους στα υπόλοιπα στοιχεία της πλατφόρμας.



Εικόνα 2-4: Παραδείγματα μηχανισμού αρπάγης (gripper) με ενσωματωμένους κινητήρες Servo, συγκεκριμένα έχουν *microservo motor*

Οι μηχανισμοί αυτοί είναι μόνο παραδείγματα μηχανισμού αρπάγης (gripper), είναι εξαρτήματα του εμπορίου και μπορεί κανείς να τα προμηθευτεί από το διαδίκτυο. Η πλάκα έχει σχεδιαστεί με τέτοιο τρόπο ώστε τέτοιοι μηχανισμοί να μπορούν να τοποθετηθούν εκεί.

### 2.3.2. Κάτω πλάκα

Η κάτω πλάκα είναι το ουσιαστικό σημείο ενδιαφέροντος της πολυμορφίας της διάταξης. Είναι σημαντικό να σημειωθεί ότι ένας ακόμα λόγος για την επιλογή του εκτυπωτή είναι ότι με αυτό τον τρόπο ικανοποιείται η συνθήκη της πολυμορφίας. Με αυτό τον τρόπο αντί να σχεδιαστεί μια πλατφόρμα με όλα τα απαραίτητα χαρακτηριστικά σχεδιάζονται δυο διαφορετικές, οι οποίες είναι συμβατές με διαφορετικούς τύπους κινητήρων. Επομένως, η διάταξη αυτή είναι συμβατή με κινητήρες συνεχούς τάσης DC αλλά και με Servo.

#### Διαφορά κινητήρων DC και Servo

Ένας ηλεκτροκινητήρας είναι μια συσκευή μετατροπής της ηλεκτρικής ενέργειας σε μηχανική με αποτελέσματα την περιστροφή ενός άξονα. Με βάση τους τύπους ηλεκτρικής τροφοδοσίας, οι ηλεκτροκινητήρες ταξινομούνται σε πολλούς τύπους όπως κινητήρες συνεχούς ρεύματος (Direct Current- DC), επαγωγικοί κινητήρες (Induction Motors), σύγχρονοι κινητήρες (Synchronous Motors), βηματικοί κινητήρες (Stepper Motors), σερβοκινητήρες (Servo Motors) και πολλά άλλα. Τα περισσότερα ρομπότ μικρού έως μεσαίου μεγέθους χρησιμοποιούν σερβοκινητήρες ή κινητήρες DC.[25]

Όπως υποδηλώνει το όνομά τους, ένας κινητήρας DC μετατρέπει την ηλεκτρική ενέργεια με τη μορφή συνεχούς ρεύματος (DC) σε μηχανική ενέργεια (περιστροφή άξονα). Οι κινητήρες DC έχουν δύο τύπους κατασκευών, τον κινητήρα DC με ψήκτρες και τον κινητήρα συνεχούς ρεύματος χωρίς ψήκτρες.

Ένας κινητήρας συνεχούς ρεύματος αποτελείται από ένα τμήμα στάτορα που σχηματίζει το σύστημα μαγνητικού πεδίου και ένα τμήμα ρότορα που λειτουργεί ως οπλισμός του κινητήρα. Ένας συλλέκτης είναι επίσης τοποθετημένος στον άξονα του ρότορα. Τα τμήματα του συλλέκτη συνδέονται με τα άκρα της περιέλιξης του οπλισμού. Το ηλεκτρικό ρεύμα που ρέει μέσω της περιέλιξης του οπλισμού παράγει ένα μαγνητικό πεδίο το οποίο αλληλεπιδρά με το μαγνητικό πεδίο του στάτορα για να παράγει ροπή στον ρότορα. Λόγω αυτής της ροπής, ο ρότορας περιστρέφεται για να παράγει μηχανική ενέργεια εξόδου στον άξονα.

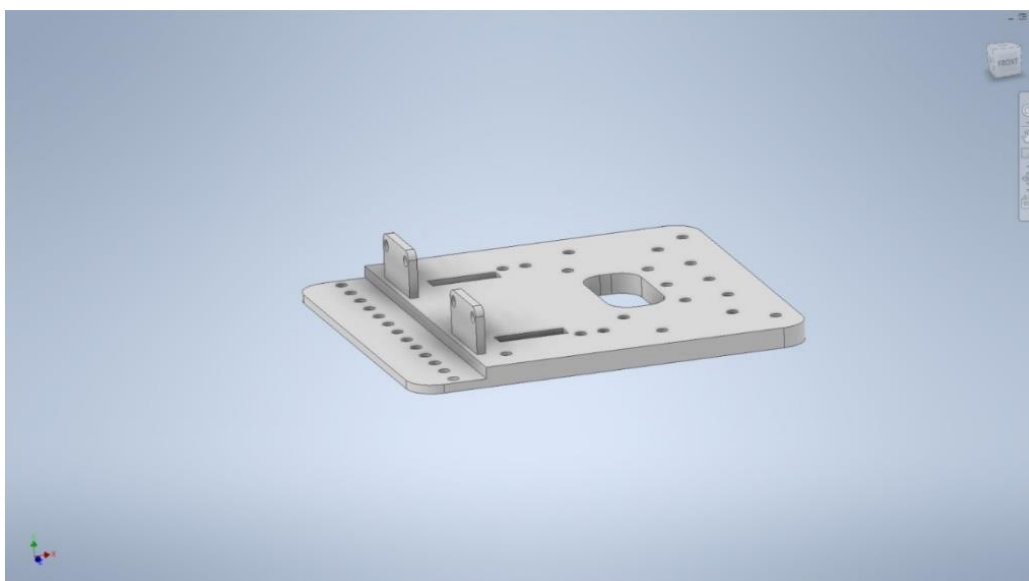
Οι κινητήρες συνεχούς ρεύματος χρησιμοποιούνται συνήθως σε ηλεκτρονικές συσκευές, γερανούς, παιχνίδια, ανελκυστήρες και ανελκυστήρες, ηλεκτρικά εργαλεία και σε πολλές άλλες συσκευές.

Ένας τύπος ηλεκτροκινητήρα που επιτρέπει τον ακριβή έλεγχο της γωνιακής ταχύτητας, της θέσης και της επιτάχυνσης είναι γνωστός ως σερβοκινητήρας. Ένας σερβοκινητήρας αποτελείται από έναν αισθητήρα για ανάδραση θέσης στον ρότορα που χρησιμεύει για τον έλεγχο της θέσης και της ταχύτητάς του ανά πάσα στιγμή. Έτσι, οι σερβοκινητήρες χαρακτηρίζονται γενικά από το σχεδιασμό του συστήματος ελέγχου κλειστού βρόχου.

Οι σερβοκινητήρες μπορούν να είναι δύο τύπων, δηλαδή σερβοκινητήρες AC και σερβοκινητήρες DC. Όπου, οι σερβοκινητήρες AC χρησιμοποιούνται σε μεγάλες εφαρμογές, ενώ οι σερβοκινητήρες DC χρησιμοποιούνται σε μικρές ηλεκτρονικές συσκευές όπως ρομποτικές αρθρώσεις, σύστημα αυτόματης εστίασης κάμερας, συστήματα εντοπισμού κεραίας, μηχανή CNC κ.λπ.

#### *Κάτω πλάκα για κινητήρες DC*

Στην περίπτωση των κινητήρων DC, η πλάκα διαμορφώνεται με τα ακόλουθα χαρακτηριστικά. Οι γενικές διαστάσεις είναι 140mm x 130mm x 22mm (στο μέγιστο ύψος) ή 3mm (στο ελάχιστο). Το βασικό σχέδιο παραμένει απλό αλλά στην συγκεκριμένη περίπτωση τα χαρακτηριστικά έχουν προσαρμοστεί για να μπορούν να τοποθετηθούν οι κινητήρες, οι ρόδες και οι αισθητήρες. Αρχικά, το βύθισμα που κάνει θεωρείται το μπροστινό μέρος, καθώς εκεί μπορεί να τοποθετηθούν αισθητήρες στραμμένοι προς το έδαφος για να ακολουθήσουν μία διαδρομή ή στραμμένοι προς τα μπροστά για να εντοπίζει και να αποφεύγει εμπόδια.



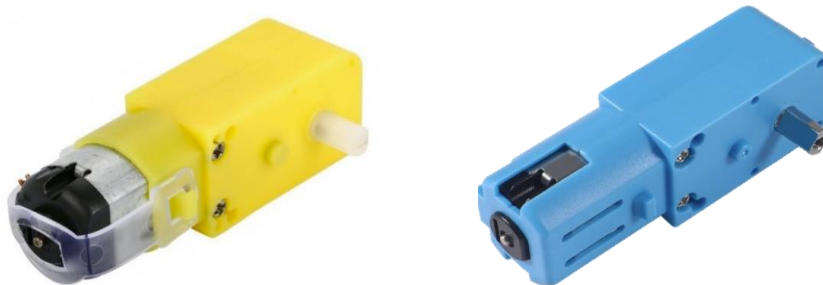
Εικόνα 2-5: Φωτογραφία από το πρόγραμμα Inventor Pro 2022, όπου απεικονίζεται η κάτω πλάκα για κινητήρες DC.

Ακριβώς από πίσω βρίσκονται δύο υψωμένα κομμάτια, τα οποία είναι σχεδιασμένα μόνο για αυτή την περίπτωση, αποτελούν την βάση των οπτικών αισθητήρων μέτρησης περιστροφών. Αντιστοίχως, οι εγκοπές που εμφανίζονται δίπλα τους είναι τοποθετημένες εκεί για να μπορεί να λειτουργήσει ελεύθερα ο αισθητήρας. Η λειτουργία του βασίζεται στην περιστροφή ενός διάτρητου τροχού παράλληλα με τις ρόδες και ένας φωτοευαίσθητος αισθητήρας μπορεί να αντιληφθεί τα κενά και να τα μετρήσει. Έτσι είναι δυνατόν να γνωρίζει και να μπορεί να ελέγξει κανείς τους κινητήρες. Για την καλύτερη κατανόηση ακολουθεί μια φωτογραφία των όσων περιγράφονται, ενώ η πλήρη ανάλυση ακολουθεί στο επόμενο κεφάλαιο που αναλύονται τα ηλεκτρονικά.

Στο πίσω μέρος εντοπίζεται μια ομάδα από οπές οι οποίες είναι σχεδιασμένες για να τοποθετούνται εκεί οι πίσω ρόδες της διάταξης. Η διάταξη μπορεί να έχει μόνο μία βοηθητική ρόδα στο κέντρο ή να έχει δύο ρόδες οι οποίες βρίσκονται στα πλάγια. Αυτές οι ρόδες μπορεί να είναι μπίλιες, γνωστά και ως *caster*, ή να είναι οι τυπικές ρόδες. Με αυτό τον τρόπο, η πλατφόρμα μπορεί να είναι τρίτροχη ή τετράτροχη, ανάλογα με τις ανάγκες του προβλήματος. Τέλος, το κενό στην μέση είναι για εξοικονόμηση βάρους, υλικού και χρόνου. Ωστόσο, είναι και χρήσιμο για την τακτοποίηση καλωδίων.

### *Στηρίγματα για τους κινητήρες DC*

Στην αγορά υπάρχουν αρκετοί τύποι κινητήρων DC. Ωστόσο αυτοί που επιλέχθηκαν για την κατασκευή είναι αυτοί που εμφανίζονται στην πλειοψηφία των ρομποτικών διατάξεων του εμπορίου. Ο βασικός λόγος είναι το κόστος τους και η διαθεσιμότητά τους. Όπως φαίνεται και στην παρακάτω εικόνα, το σχήμα τους είναι απλό και είναι ιδανικό για διάφορες κατασκευές. Τα αναλυτικά χαρακτηριστικά τους αναλύονται στο επόμενο κεφάλαιο. Επομένως επιλέχθηκαν αυτού του τύπου οι κινητήρες DC.



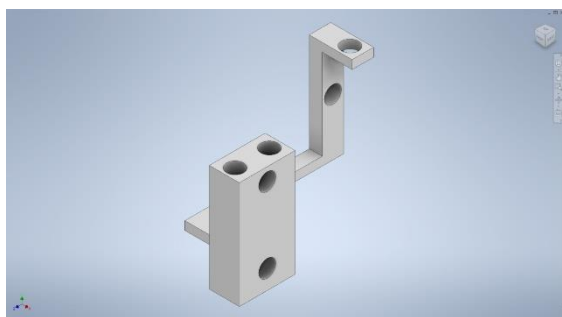
*Εικόνα 2-6: Δύο διαφορετικά μοντέλα κινητήρα DC του εμπορίου, σύμφωνα με τα οποία σχεδιάστηκε η κάτω πλάκα.*

Για να μπορέσουν να τοποθετηθούν οι κινητήρες DC στην πλάκα χρειάζονται το κατάλληλο στήριγμα. Στο εμπόριο προσφέρονται οι δύο παρακάτω τύποι, αλλά δεν είναι ικανοί να στηρίξουν επαρκώς τον κινητήρα ώστε να περιστρέφεται ελεύθερα. Για αυτό σχεδιάστηκε ένα στήριγμα προσαρμοσμένο στις ανάγκες του παρόντος προβλήματος.

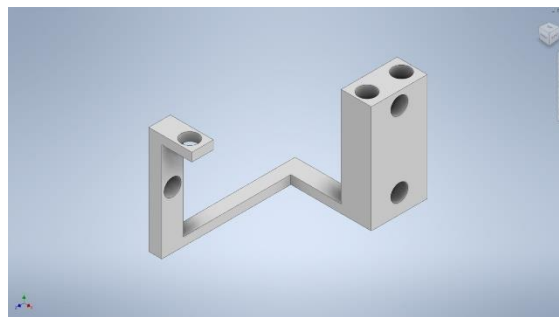


Εικόνα 2-7: Στηρίγματα του εμπορίου για τους κινητήρες DC της εικόνας 4

Το στήριγμα σχεδιάστηκε με βάση αυτά του εμπορίου, με την διαφορά ότι του προστέθηκε ένας επιπλέον τρόπος στηρίγματος στο «μπροστινό» μέρος του κινητήρα. Αυτό προσφέρει μεγαλύτερη σταθερότητα στην στήριξη του κινητήρα. Στην συνέχεια φαίνονται τα δύο στηρίγματα, για τον αριστερό και για τον δεξί κινητήρα.



α



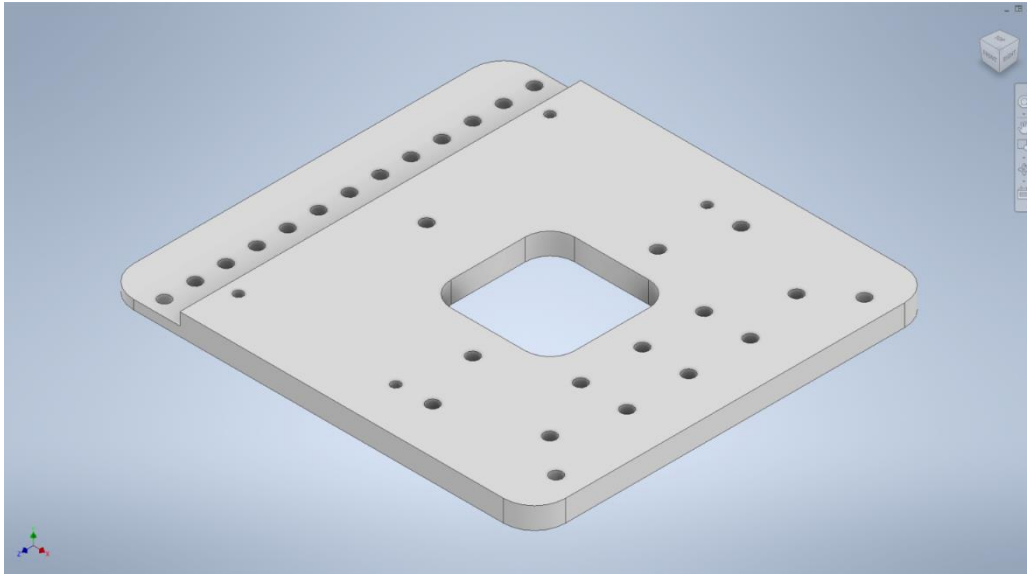
β

Εικόνα 2-8: Τα δύο στηρίγματα για τους κινητήρες DC, όπου το α είναι του δεξιού και το β είναι του αριστερού

### Κάτω πλάκα για κινητήρες Servo

Για την περίπτωση που χρησιμοποιούνται Servo κινητήρες, το σχέδιο της κάτω πλάκας αλλάζει ελαφρώς. Οι γενικές διαστάσεις παραμένουν οι ίδιες, δηλαδή 140mm x 130mm x 7mm (στο μέγιστο ύψος) ή 3mm (στο ελάχιστο). Δεν υπάρχει στηρίγμα για οπτικούς αισθητήρες μέτρησης περιστροφών, καθώς ένα από τα χαρακτηριστικά των αυτών των κινητήρων είναι η ενσωμάτωση του αισθητήρα. Η εικόνα που ακολουθεί είναι το προτεινόμενο σχέδιο για την κάτω πλάκα όταν χρησιμοποιούνται κινητήρες σερβο.

Το βύθισμα στο μπροστινό μέρος είναι για τους αισθητήρες, παρόμοια με την λογική της πλάκας για κινητήρες DC. Το ίδιο ισχύει και για τις οπές στο πίσω μέρος όπου μπορούν να μετατρέψουν την διάταξη σε τρίτροχο ή σε τετράτροχο.



Εικόνα 2-9: Φωτογραφία από το πρόγραμμα Inventor Pro 2022, όπου απεικονίζεται η κάτω πλάκα για κινητήρες Servo

### Στηρίγματα σερβοκινητήρων

Παρόμοια με του κινητήρες DC, οι σερβοκινητήρες έχουν διάφορους τύπους με τον πιο γνωστό να είναι ο εικονιζόμενος στη εικόνα που ακολουθεί. Επομένως επιλέχθηκαν αυτοί να χρησιμοποιηθούν στην διάταξη. Όμως σε αντίθεση με τους κινητήρες DC, τα στηρίγματα για αυτούς τους κινητήρες είναι ικανοποιητικά και δεν απαιτείται περαιτέρω βελτίωση.



α

β

Εικόνα 2-10: Στην εικόνα 7.α. απεικονίζεται ένας τυπικός σερβοκινητήρας και στην 7.β. είναι το στηρίγμα σύμφωνα με το σχεδιάστηκε η κάτω πλάκα σε αυτή την περίπτωση

## 2.4. Προτεινόμενη κατασκευή

Μετά την παραπάνω ανάλυση για μπορέσει να ελεγχθεί η ποιότητα της κατασκευής αποφασίστηκε να δημιουργηθεί μία πρωτότυπη μικρορομποτική διάταξη. Όπως αναφέρεται και παραπάνω το βασικό κομμάτι της κατασκευής γίνεται με την χρήση τρισδιάστατου εκτυπωτή. Επίσης επιλέχθηκε η εναλλακτική με τους κινητήρες DC για να δοκιμαστούν και τα νέα στηρίγματα που σχεδιάστηκαν. Στην συνέχεια αναλύεται η διαδικασία που ακολουθείται για την κατασκευή.

### 2.4.1. Τρισδιάστατος εκτυπωτής

Για το εγχείρημα αυτό χρησιμοποιήθηκε ο τρισδιάστατος εκτυπωτής Creality CR-20. Είναι ένα σχετικά παλιό μοντέλο της εταιρείας, καθώς έχει αντικατασταθεί με το Creality 3D CR-10 και Creality 3D CR-10 Smart. [26] Αυτός ο εκτυπωτής διαθέτει μονή άκρη εναπόθεσης υλικού, που σημαίνει ότι μπορεί να τυπώσει μόνο με ένα υλικό κάθε φορά. Είναι από τους συνηθέστερους τύπους 3D εκτυπωτή. Μπορεί να χρησιμοποιήσει PLA και ABS, με μέγιστη ταχύτητα εκτύπωσης 100mm/s.

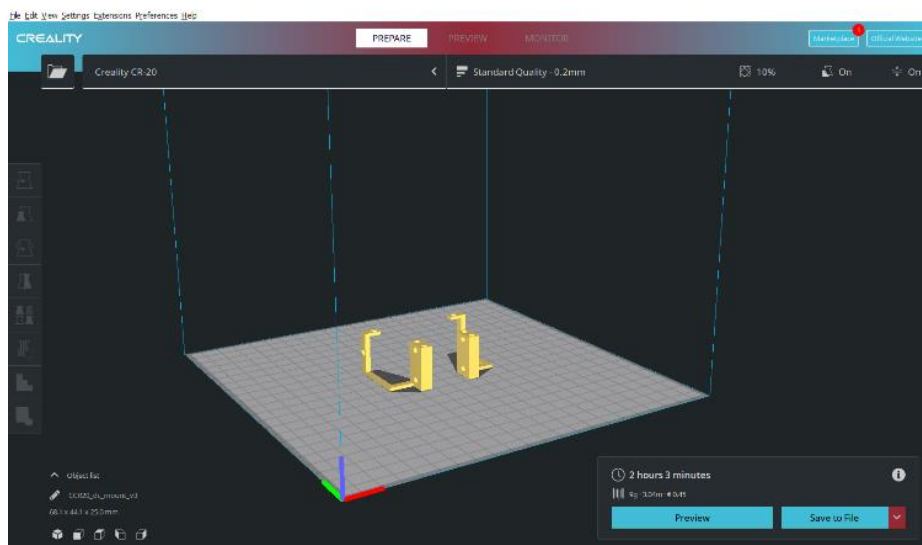
Είναι συναρμολογούμενο μοντέλο χωρίς προστασία τριγύρω και οι διαστάσεις του είναι 420×470×410 mm, ενώ οι διαθέσιμες διαστάσεις για εκτύπωση είναι 220×220×250 mm. Το μέγεθος του νήματος είναι 1.75 mm, που είναι το συνηθέστερο στην αγορά. Ο τρόπος που μεταφέρεται το πρόγραμμα είναι με μια SD Card, η οποία τοποθετείται στον εκτυπωτή με το πρόγραμμα να είναι σε μορφή GCODE.



Εικόνα 2-11: Τρισδιάστος εκτυπωτής Creality CR-20 Pro

### 2.4.2. Πρόγραμμα εκτύπωσης

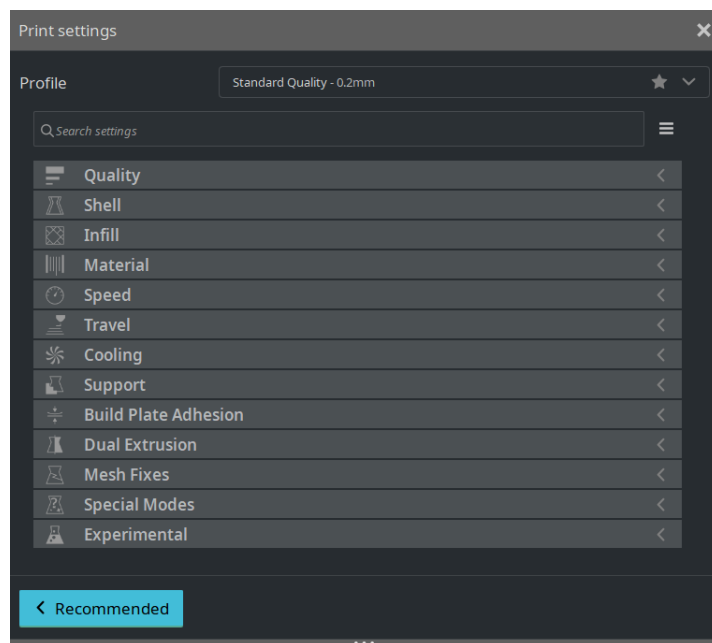
Αφού έχουν γίνει τα σχέδια, σε οποιοδήποτε σχεδιαστικό πρόγραμμα, είναι απαραίτητο να γίνει η μετατροπή τους από το σχεδιαστικό σε αρχείο stl, για να μπορεί να εισαχθεί στο πρόγραμμα του Slicer. Αυτό το πρόγραμμα μετατρέπει τα ψηφιακά μοντέλα 3D σε αρχεία προς εκτύπωση. Εκτός από το ίδιο το μοντέλο, οι οδηγίες περιέχουν παραμέτρους εκτύπωσης 3D που έχουν εισαχθεί από το χρήστη, όπως το ύψος του στρώματος, την ταχύτητα και τις ρυθμίσεις δομής στήριξης. Ο εκτυπωτής είναι συμβατός με τα προγράμματα εκτύπωσης (slicer) Creality Slicer 4.8.2, Cura και Simplify3D.



Εικόνα 2-12: Το περιβάλλον του προγράμματος Creality Slicer 4.8.2. με τα δύο στηρίγματα για τους κινητήρες DC



Για αυτή την κατασκευή χρησιμοποιήθηκε το πρόγραμμα Creality Slicer 4.8.2. και το περιβάλλον στο οποίο εισάγονται τα σχέδια είναι αυτό που ακολουθεί στις εικόνα. Μέσα από αυτά τα προγράμματα μπορούν να αλλάξουν μεταβλητές όπως το πάχος του τοιχώματος του μοντέλου, η ταχύτητα με την τυπώνεται, η ποιότητα της επιφάνειας, το ποσοστό γεμίσματος του τελικού υλικού και αρκετά ακόμα. Στην εικόνα 11 έχουν εισαχθεί στο πρόγραμμα τα δύο στηρίγματα για τους κινητήρες DC, τα οποία όπως φαίνεται θα χρειαστούν 2 ώρες για εκτυπωθούν.



Εικόνα 2-13: Το μενού του προγράμματος εκτύπωσης για τις μεταβλητές που μπορούν να αλλάξουν ανάλογα με τις ανάγκες του υλικού και του μοντέλου.

Με βάση το πρόγραμμα δίνονται στην συνέχεια ο εκτιμώμενος χρόνος που χρειάζεται το κάθε μοντέλο για να τυπωθεί. Αυτοί οι χρόνοι είναι για το παλιότερο μοντέλο του εκτυπωτή και λογικά σε ένα πιο σύγχρονο θα είναι μικρότεροι.

Πίνακας 2-1: Χρόνος εκτύπωσης ανάλογα με το κομμάτι της κατασκευής

<b>Κομμάτι κατασκευής</b>	<b>Χρόνος εκτύπωσης</b>
Στηρίγματα DC	2 ώρες
Κάτω πλάκα DC	8 ώρες
Κάτω πλάκα Servo	7 ώρες και 20 λεπτά
Πάνω πλάκα	4 ώρες και 20 λεπτά

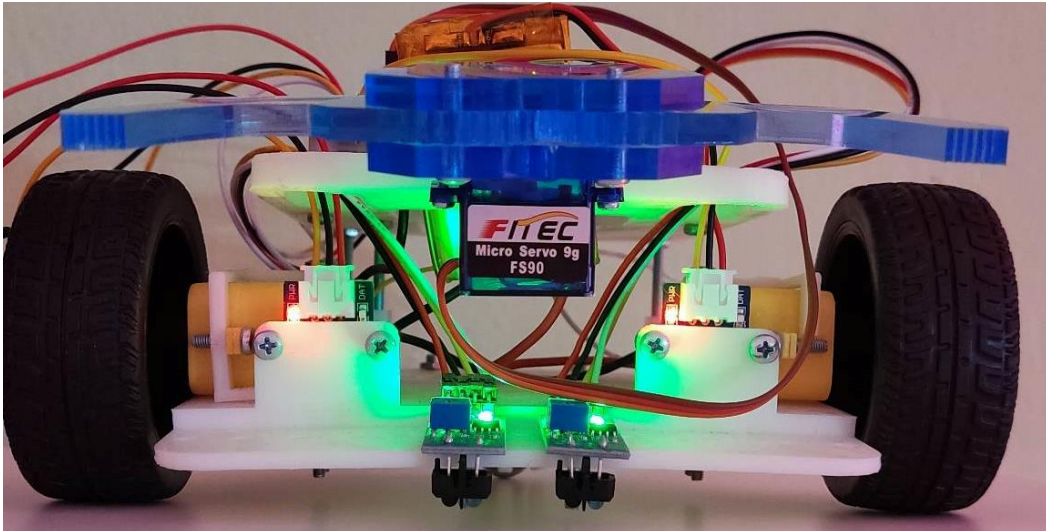
### 2.4.3. Τελική κατασκευή

Με βάση όλα τα παραπάνω δημιουργείται μια αναλυτική λίστα με όλα τα κομμάτια της προτεινόμενης κατασκευής και το κόστος τους.

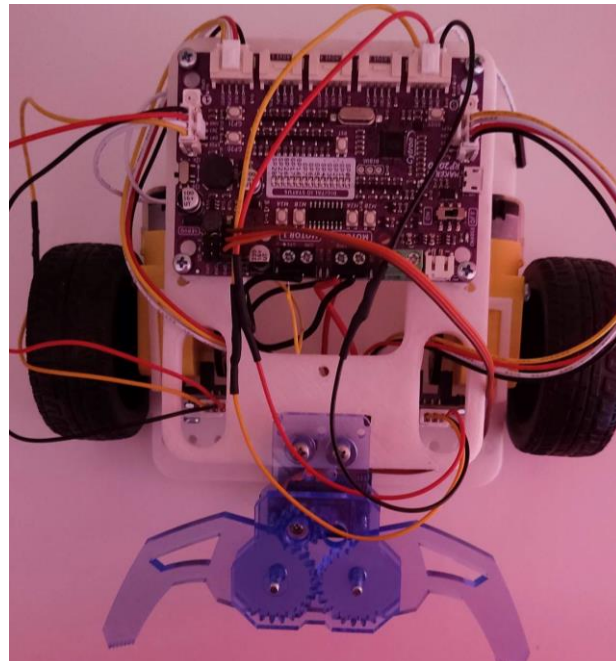
Πίνακας 2-2: Συνολική λίστα με τα υλικά για την προτεινόμενη κατασκευή

Υλικά	Ποσότητα	Τιμή (€/τεμάχιο)
Πάνω πλάκα	1	17.50*
Στηρίγματα DC	2	
Κάτω πλάκα DC	1	
Κινητήρες DC	2	1.80
Μηχανισμός Gripper	1	9.90
Ροδάκι Caster	1	1.20
Ρόδα (66x26 mm)	2	1.20
Αισθητήρας TCRT500	2	1.60
Οπτικός αισθητήρας (περιέχει 1 ροδέλα για τον αισθητήρα)	2	4.40
πλακέτα Maker Pi RP2040 (περιέχει: 4 καλώδια για τις θύρες Grove)	1	15.00
Μπαταρία Λίθου 3.7 V /1200 mAh	1	8.20
Βίδες (M3 /Μήκος: 10 mm)	14	0.03
Βίδες (M3 / Μήκος: 16 mm)	11	0.03
Βίδες (M3 / Μήκος: 30 mm)	5	0.04
Βίδες (M3 / Μήκος: 35 mm)	8	0.08
Αποστακτικά F/F (M3 / Μήκος: 10 mm)	9	0.2
Περικόχλια	38	0.04
Φορτιστής Μπαταρίας	1	7.40
Καλώδιο micro USB	1	2.20
Καλώδιο USB type C	1	2.40
<b>Συνολικό ποσό ανά κατασκευή</b>		<b>86.71 (69.21)</b>

\*Η τιμή του βασικού σώματος του ρομπότ δεν αφορά τα κομμάτια ξεχωριστά άλλα το υλικό που θα τοποθετηθεί στον εκτυπωτή. Στην συγκεκριμένη περίπτωση είναι PLA 1.75 mm και δίνεται ανά κιλό. Με ένα κιλό υλικό μπορούν να κατασκευαστούν 9 τέτοιες κατασκευές.



*Εικόνα 2-14: Τελική προτεινόμενη κατασκευή μπροστινή όψη*



*Εικόνα 2-15: Τελική προτεινόμενη κατασκευή με την πλακέτα Maker Pi RP2040*

## Κεφάλαιο 3ο Ηλεκτρονική Διάταξη

Τα μηχανικά συστήματα είναι αναπόσπαστο κομμάτι της σύγχρονης κοινωνίας. Όπως αναλύθηκε σε προηγούμενο κεφάλαιο, η Μηχανική είναι το διεπιστημονικό πεδίο που ασχολείται με τον συνδυασμό τριών μεγάλων επιστημών για ένα σύστημα. Μία από αυτές είναι η Μηχανολογία, δηλαδή το φυσικό σύστημα, το οποίο αναλύθηκε στο προηγούμενο κεφάλαιο. Η δεύτερη συνιστώσα είναι η Ηλεκτρολογία- Ηλεκτρονική, που αφορά τις ηλεκτρονικές διατάξεις του εξεταζόμενου συστήματος. [2] Σε αυτή ανήκουν οι ηλεκτρονικές πλακέτες, οι αισθητήρες και οι ηλεκτροκινητήρες που θα αναλυθούν στο παρόν κεφάλαιο. Τέλος η Μηχανική περιέχει κομμάτι της πληροφορικής, συγκεκριμένα ότι αφορά τον προγραμματισμό φυσικών, κυρίως, συστημάτων.

Το ηλεκτρονικό κομμάτι του μηχανικού συστήματος χαρακτηρίζεται από κάποιες βασικές παραμέτρους. Η σημαντικότερη είναι ο επεξεργαστής σύμφωνα με τον οποίο ορίζονται και οι πραγματικές δυνατότητες του συστήματος, πέρα από τους φυσικούς περιορισμούς της κατασκευής. Τα περισσότερα μηχανικά συστήματα χρησιμοποιούν μικρολεγκτές ή μικροεπεξεργαστές, καθώς αυτοί προσφέρουν όσα είναι απαραίτητα για το σύστημα.

Η σύγχρονη κοινωνία χαρακτηρίζεται από ένα τεράστιο πλήθος ηλεκτρονικών συσκευών, όπως το κινητό τηλέφωνο «smartphone», τον ηλεκτρονικό υπολογιστή και αρκετές ακόμα. Όλες αυτές οι συσκευές περιέχουν ηλεκτρονικές διατάξεις με μικροελεγκτές ή μικροεπεξεργαστές και αισθητήρες, τα οποία είναι τα βασικά χαρακτηριστικά όλων των ρομποτικών διατάξεων. Για να λειτουργήσει ένα ρομποτικό σύστημα, πέρα από την κίνηση που γίνεται μέσω των κινητήρων, όπως αναφέρθηκε και στο προηγούμενο κεφάλαιο, είναι απαραίτητο το ηλεκτρονικό σύστημα.

Ο μικροελεγκτής ή ο μικροεπεξεργαστής που επιλέχθηκε σε κάθε συσκευή αποτελεί το κύριο όργανο ελέγχου στο οποίο δίνονται οι εντολές για την λειτουργία του συστήματος.[10] Ωστόσο, για να μπορέσει να δοθεί η κατάλληλη εντολή στο σύστημα, αυτό πρέπει να μπορεί να επιδρά στο περιβάλλον που βρίσκεται. Η αλληλεπίδραση περιβάλλοντος και συστήματος γίνεται με τα αισθητήρια όργανα που επιλέγονται ανάλογα με τις ανάγκες και τον σκοπό του συστήματος.

Σύμφωνα με τα δεδομένα της ρομποτικής διάταξης που αναπτύσσεται σε αυτή την εργασία, αναλύονται στην συνέχεια κάποια πιθανές ηλεκτρονικές διατάξεις που μπορούν να χρησιμοποιηθούν. Επειδή ο σκοπός της διάταξης, πέρα από την πολυμορφία, είναι και η καινοτομία, αποφασίστηκε η διάταξη να είναι συμβατή με πολλούς μικροελεγκτές. Η ηλεκτρονική διάταξη αποτελείται από τον κεντρικό μικροελεγκτή τον RP2040, της εταιρείας Raspberry Pi [13] που χρησιμοποιείται στο Raspberry Pi Pico. Ωστόσο, για να απλοποιηθεί η διάταξη και να μην αποτελέσει τροχοπέδη στην εκπαίδευση των φοιτητών, επιλέχθηκαν δύο εμπορικές πλακέτες με τα βασικά κυκλώματα έτοιμα. Αυτές είναι η Maker Pi Pico και Maker Pi RP2040 από την εταιρεία Cytron[27].

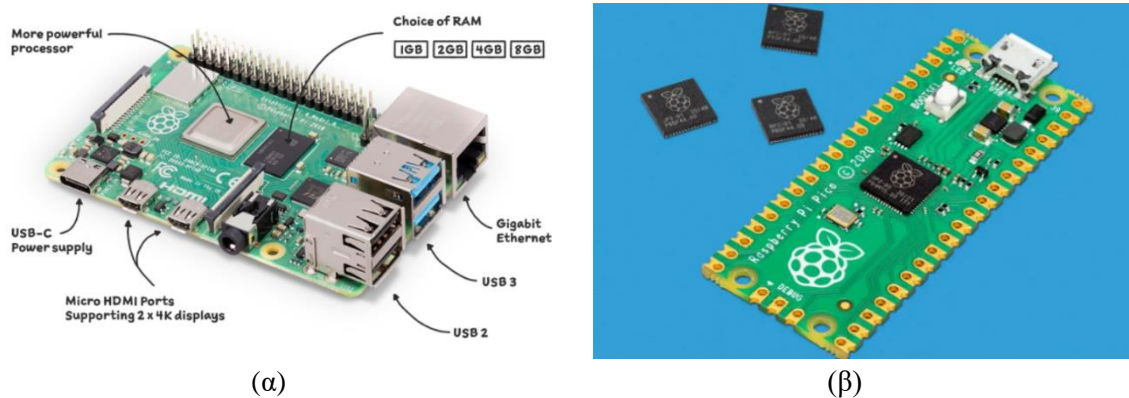
### 3.1. Raspberry

Η εταιρεία Raspberry Pi έχει αποκτήσει μία θέση στην κοινή συνείδηση ερασιτεχνών και επαγγελματιών χάρη στις ισχυρές, χαμηλού κόστους πλακέτες μικροεπεξεργαστών.[28] Οι πλακέτες Raspberry Pi είναι οι πιο δημοφιλείς από τις πολλές πλακέτες μικροεπεξεργαστών που διατίθενται. Ακόμη πιο σημαντικό είναι το γεγονός πως τα Raspberry Pi έχουν σχεδιαστεί για την εκπαίδευση. Οι εκπαιδευτικοί μπορούν να χρησιμοποιήσουν το Raspberry Pi για να διδάξουν ηλεκτρονικές διατάξεις,

αυτοματισμούς ακόμα και την σύγχρονη επιστήμη του Διαδικτύου των Πραγμάτων (IoT) χρησιμοποιώντας γλώσσες προγραμματισμού όπως Python, Java ή C/C++.

Στην εκπαίδευση συχνά επιλέγεται το Raspberry αντί του Arduino, γιατί το Raspberry έχει προϊόντα όπως το Raspberry Pi model 3 B, το οποίο είναι ένας χαμηλού κόστους ολοκληρωμένος υπολογιστής, που λειτουργεί είτε με το ανοιχτό λογισμικό Linux είτε με το προσφερόμενο από την Raspberry.[29][30] Με την εισαγωγή των ηλεκτρονικών διατάξεων, οι μαθητές διδάσκονται μέσω παραδειγμάτων και ενθαρρύνονται να συνδυάζουν διαφορετικούς κλάδους για την επιτυχή επίλυση προβλημάτων.

Η θεωρία συμπληρώνεται από φυσικά πειράματα και επιδείξεις. Το Raspberry Pi είναι μια ιδανική πλατφόρμα για τους μαθητές να αποκτήσουν νέες γνώσεις και ικανότητες σε διάφορους τομείς που μπορεί να χρησιμοποιηθεί.[31] Οι φοιτητές πρέπει να ενθαρρύνονται να χρησιμοποιούν νέες τεχνολογίες κατά τη διάρκεια της εκπαίδευσης και των σπουδών τους.

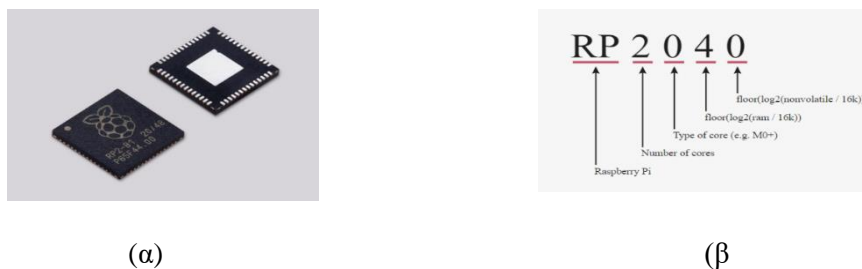


Εικόνα 3-1: Δύο από τα προϊόντα της εταιρείας Raspberry Pi. Στην (α) είναι το παραδοσιακό Raspberry Pi, ενώ στην (β) είναι ο μικροελεγκτής Raspberry Pi Pico[11]

Στις φωτογραφίες φαίνονται τα προϊόντα Raspberry Pi Model 4 (α) και Raspberry Pi Pico, με τον επεξεργαστή RP2040 (β). Το προϊόν στην εικόνα (α) είναι ο ολοκληρωμένος υπολογιστής χαμηλού κόστους (περίπου 35\$, στις 10/12/2022), ενώ στην (β) είναι ο μικροελεγκτής με εξίσου χαμηλό κόστος (περίπου 7\$, στις 10/12/2022)

### 3.1.1. Raspberry Pi RP2040/ Pico

Ωστόσο, η Raspberry δεν έμεινε μόνο στους μικροεπεξεργαστές και ολοκληρωμένους υπολογιστές, αλλά επεκτάθηκε και στους μικροελεγκτές με το RP2040 είναι το πρώτο «τσιπάκι» μικροελεγκτή διαθέσιμο στην αγορά. Υπό φυσιολογικές συνθήκες το όνομα του προϊόντος είναι συνήθως κάποιος αύξων κωδικός ή κάποιος κωδικός στο στάδιο της κατασκευής. Όμως το RP2040 είναι ένας κωδικός που προσφέρει στοιχεία σχετικά με τις τεχνικές δυνατότητες του.

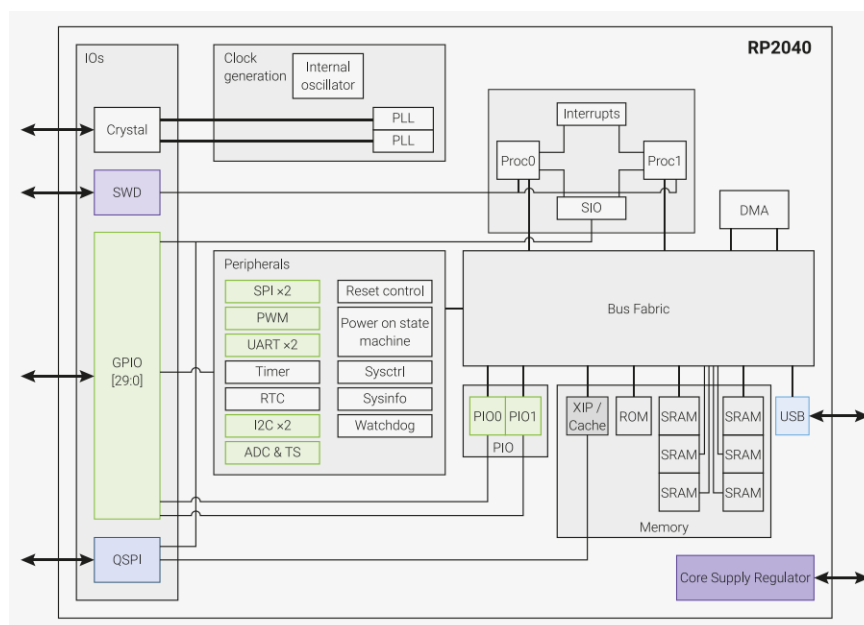


Εικόνα 3-2: (α) Ο επεξεργαστής RP2040 της Raspberry Pi και (β) η εξήγηση για το όνομα του επεξεργαστή RP2040[11]

Ο RP2040 είναι ένας χαμηλού κόστους αλλά υψηλών δυνατοτήτων μικροελεγκτής[32] και τα βασικά στοιχεία του είναι τα εξής:

- Διπλός επεξεργαστής Cortex M0+, έως 133MHz
- 264kB ενσωματωμένη SRAM
- 30 πολυχρηστικά GPIO
- 6 ακίδες IO αποκλειστικά για SPI Flash (supporting XIP)
- Υλικό αποκλειστικά προσαρμοσμένο για περιφερειακά που χρησιμοποιούνται συνήθως
- Προγραμματιζόμενα IO για υποστήριξη περαιτέρω περιφερειακών
- 4 κανάλια ADC με αισθητήρα εσωτερικής θερμοκρασίας, 500ksps, 12-bit conversion
- USB 1.1 Host/Device

Στην συνέχεια δίνεται το σχέδιο του επεξεργαστή RP2040, όπως αυτό φαίνεται στο φύλλο δεδομένων της εταιρείας.



Εικόνα 3-3: Το σύστημα του επεξεργαστή RP2040 [32]

Το μοντέλο Pico είναι η συνέχεια του επεξεργαστή RP2040, αφού εκεί έκανε για πρώτη φορά το τσιπάκι την εμφάνιση του. Είναι ένα ακόμα προϊόν της εταιρείας και τα βασικά τεχνικά του χαρακτηριστικά, σύμφωνα με τον επίσημο ισότοπο [32] είναι τα εξής:

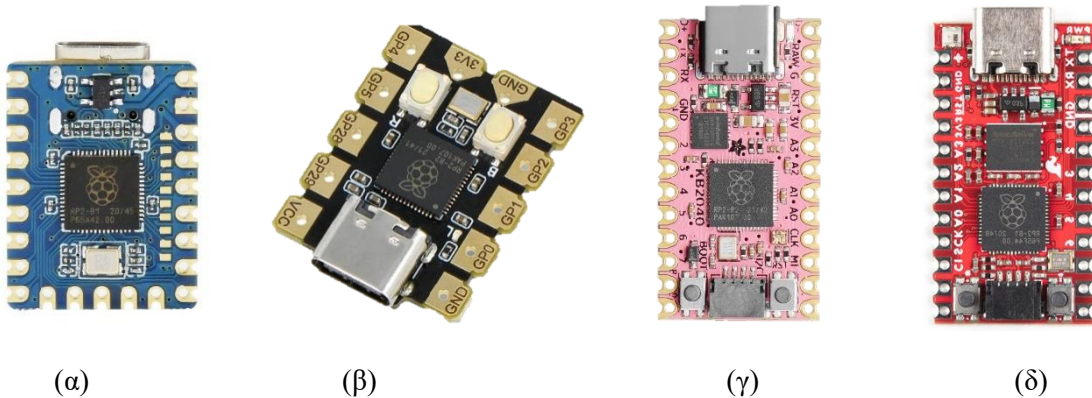
- Διπλός ARM Cortex-M0+ στα 133 MHz
- SRAM 264kB σε έξι ανεξάρτητες τράπεζες
- Υποστήριξη για έως και 16MB μνήμης Flash εκτός τσιπ μέσω αποκλειστικού διαύλου QSPI
- Ελεγκτής DMA
- Πλήρως συνδεδεμένη AHB crossbar
- Interpolator and integer divider peripherals
- Προγραμματιζόμενο LDO για την παραγωγή τάσης πυρήνα
- 2 on-chip PLL για τη δημιουργία ρολογιών πυρήνα και USB
- 30 ακίδες GPIO, 4 από τις οποίες μπορούν να χρησιμοποιηθούν ως αναλογικές εισόδους
- Περιφερειακά
  - 2 UARTs

- 2 ελεγκτές SPI
- 2 ελεγκτές I2C
- 16 ακίδες PWM
- Ελεγκτής USB 1.1 και PHY, με υποστήριξη κεντρικού υπολογιστή και συσκευής
- 8 PIO state machines



Εικόνα 3-4: Το μοντέλο Raspberry Pi Pico[11]

Όμως, το μοντέλο Raspberry Pi Pico δεν είναι το μοναδικό μοντέλο που χρησιμοποιεί αυτόν τον επεξεργαστή. Χάρη στην εκτεταμένη βιβλιογραφία, έχουν εμφανιστεί στην αγορά ηλεκτρονικές πλακέτες που πέρα από τον RP2040 έχουν και άλλες δυνατότητες. Εταιρείες όπως η Adafruit, SparkFun, Waveshare και Cytron έχουν δικά τους προϊόντα με βασικό επεξεργαστή τον RP2040. Μερικά από αυτά φαίνονται στις παρακάτω φωτογραφίες.



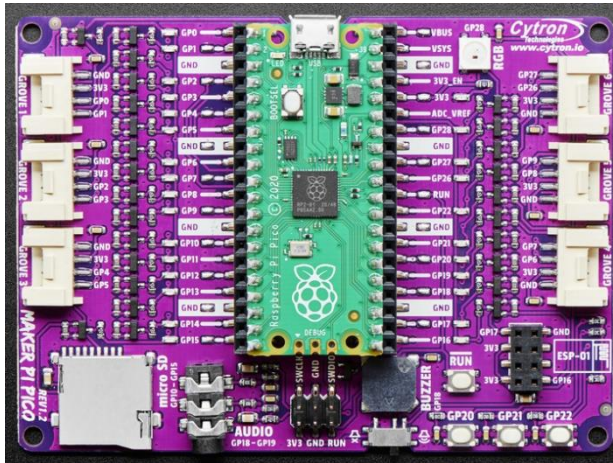
Εικόνα 3-5: Ηλεκτρονικές πλακέτες με τον RP2040 που δεν είναι της Raspberry. (α) Waveshare RP2040 Mini [51], (β) Beetle RP2040 Mini Development [52], (γ) Adafruit KB2040-RP2040 Kee Boar driver [53] και (δ) SparkFun Pro Micro- RP2040 [54]

Ωστόσο, για την ρομποτική διάταξη επιλέχθηκε να είναι συμβατή με τις πλακέτες Maker Pi Pico και Maker Pi RP2040 από την εταιρεία Cytron, οι οποίες αναλύονται στην συνέχεια.

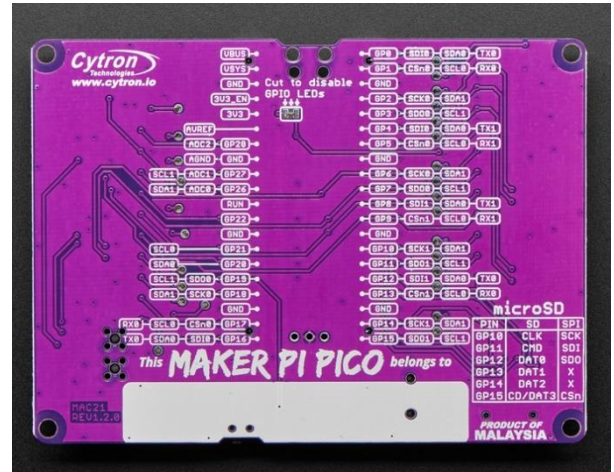
### 3.1.2. Maker Pi Pico

Το Maker Pi Pico είναι μία από τις αρκετές εμπορικές πλακέτες που είναι συμβατή με το Raspberry Pi Pico. Η πλακέτα αυτή είναι εξοπλισμένη με ένα κουμπί για το Raspberry Pi Pico αλλά και με τρία προγραμματιζόμενα κουμπιά. Έχει ηχείο και ενσωματωμένα φωτάκια LED για μεγαλύτερη ευκολία στον προγραμματισμό. Επίσης, έχει έξι θέσεις για αισθητήρες με κούμπωμα Grove, κάτι το οποίο το καθιστά ιδιαίτερα εύχρηστο καθώς δεν απαιτείται κάποιου είδους συγκόλληση και έχουν σχεδιαστεί να είναι συμβατοί με αναλογικού αλλά και ψηφιακού αισθητήρες. Αναλυτικά τα χαρακτηριστικά της είναι:

- Συμβατό με Raspberry Pi Pico
- LED για κάθε ακίδα GPIO
- 3 προγραμματιζόμενα κουμπιά (GP20-22)
- 1 NeoPixel LED (GP28)
- 1 ηχείο (GP18)
- 1 έξοδο για 3.5mm stereo audio jack (GP18-19)
- 1 θέση Micro SD (GP10-15)
- 1 ESP-01 socket (GP16-17)
- 6 θέσεις Grove για περιφερειακά



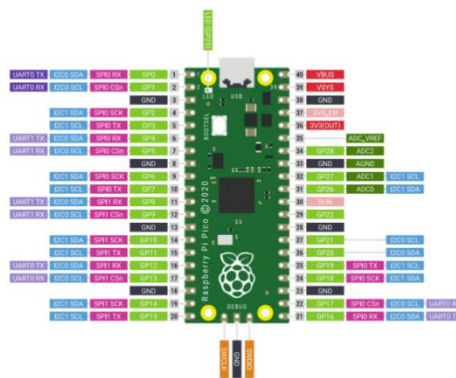
(α)



(β)

Εικόνα 3-6: Το Maker Pi Pico (α) η μπροστινή όψη, μαζί με το Raspberry Pi Pico, και (β) η πίσω όψη, με αναλυτικά στοιχεία για τις ακίδες

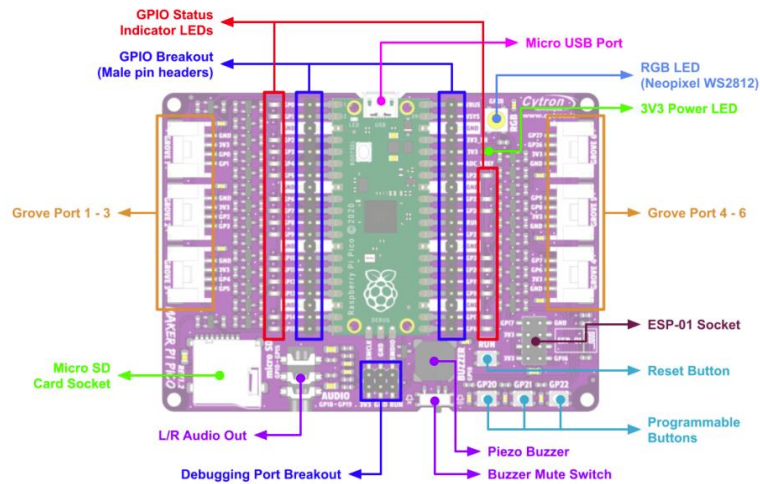
Η τροφοδοσία της πλακέτας γίνεται με τον ίδιο τρόπο που τροφοδοτείται το Raspberry Pi Pico, ενώ η επικοινωνία του με τον υπολογιστή γίνεται μέσω micro-USB. Στο επόμενο σχέδιο φαίνεται το διάγραμμα ακίδων του Raspberry Pi Pico.



Εικόνα 3-7: Το σχέδιο που δείχνει τις ακίδες για το μοντέλο Raspberry Pi Pico [32]

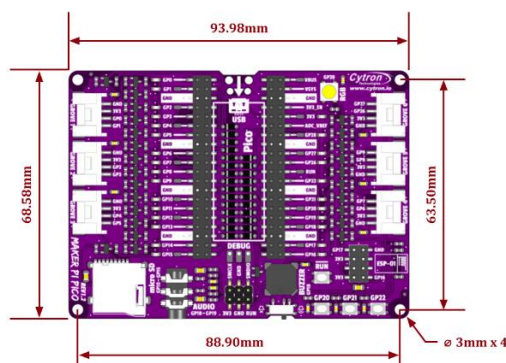
Αναλυτικά φαίνονται οι λειτουργίες της πλακέτας στο επόμενο σχήμα. Τα αναλυτικά ηλεκτρονικά σχέδια της πλακέτας δίνονται στο παράρτημα Γ.





Εικόνα 3-8: Το σχέδιο με τις αναλυτικές λειτουργίες και την τοποθέτησή τους στο μοντέλο Maker Pi Pico[55]

Ένα εξίσου σημαντικό σημείο είναι ότι η πλακέτα αυτή είναι σχετικά μικρή σε μέγεθος. Στην περίπτωση που επιλεγόταν να γίνει αυτή η διάταξη με άλλα εξαρτήματα, όπως διάτρητες πλακέτες, διακόπτες, ποτενσιόμετρα και ούτως καθεξής, θα έπρεπε να είχε προβλεφθεί και ο κατάλληλος χώρος. Ωστόσο, με αυτή τη διάταξη οι διαστάσεις είναι περιορισμένες και το κόστος μικρό, καθώς ανέρχεται στα 10 €, περίπου χωρίς το Raspberry Pi Pico.



Εικόνα 3-9: Οι διαστάσεις της πλακέτας Maker Pi Pico[55]

### 3.1.3. Maker Pi RP2040

Είναι ενδιαφέρον το ότι υπάρχουν τέτοια προϊόντα που απλοποιούν την εκπαίδευση και διευκολύνουν καθηγητές και φοιτητές να εμπλακούν ενεργά με την επίλυση ενός προβλήματος. Ακόμη μία τέτοια διάταξη είναι η πλακέτα Maker Pi RP2040[33], της ίδιας εταιρείας. Η μεγαλύτερη διαφορά μεταξύ των δύο αυτών διατάξεων είναι ότι η συγκεκριμένη δημιουργήθηκε με κύριο σκοπό να χρησιμοποιηθεί σε ρομποτικές διατάξεις.

Επομένως, η πλακέτα Maker Pi RP2040 έχει ενσωματωμένο τον επεξεργαστή RP2040 και έχει συγκεκριμένες θύρες για κινητήρες. Συγκεκριμένα δέχεται μέχρι δύο κινητήρες συνεχούς τάσης (DC motors) και τέσσερις σερβοκινητήρες (servo motors). Επίσης, έχει δύο προγραμματιζόμενα κουμπιά και επτά θέσεις για αισθητήρες Grove. Αναλυτικά τα χαρακτηριστικά της είναι:

- Χρησιμοποιεί τον επεξεργαστή Raspberry Pi RP2040, ακριβώς όπως το Raspberry Pi Pico
- Για την κίνηση

- 4 x Servo motors
- 2 x DC motors με κουμπιά για γρήγορη δοκιμή
- Επιλογή τροφοδοσίας
  - USB 5V
  - LiPo (1-cell) [Ενσωματωμένη προστασία για υπερφόρτωση]
  - Vin (3.6-6V)
- Power on/off switch
- 13 φωτάκια LED για τις ακίδες GPIO
- 1 ηχείο με διακόπτη για σίγαση
- 2 προγραμματιζόμενα κουμπιά
- 2 RGB NeoPixel LED
- 7 θύρες Grove (για αισθητήρες ή άλλα περιφερειακά όπως digital, analog, I2C, SPI, UART...)

Στην επόμενη φωτογραφία φαίνεται η μπροστινή και η πίσω όψη της πλακέτας, όπου αναγράφονται αναλυτικά και ξεκάθαρα όλα τα χαρακτηριστικά της. Είναι εμφανές το σημείο που ενώνονται οι κινητήρες και οι αισθητήρες, καθώς και οι θέσεις τροφοδοσίας. Επίσης στο πίσω μέρος αναγράφονται και οι λειτουργίες της κάθε ακίδας για μεγαλύτερη ευκολία του χειριστή.

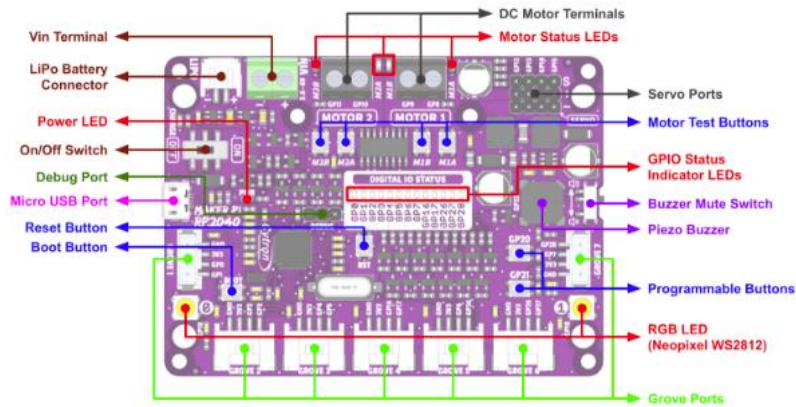


(α)

(β)

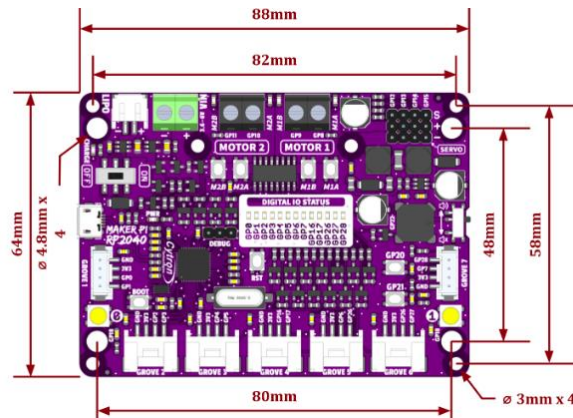
Εικόνα 3-10: Η πλακέτα Maker Pi RP2040, (α) η μπροστινή όψη και (β) η πίσω όψη [33]

Όπως αναφέρθηκε πιο πάνω η τροφοδοσία της πλακέτας μπορεί να γίνει με τρεις τρόπους. Ο πρώτος είναι μέσω micro USB, με το οποίο γίνεται και η μεταφορά δεδομένων μεταξύ διάταξης και υπολογιστή. Οι άλλοι δύο γίνονται με μπαταρίες. Στην μία περίπτωση με μία Single cell LiPo/Li-Ion (3.7V) και στην άλλη με συνδυασμό μπαταριών που βγάζουν τάση από 3.6 έως 6V. Τα σημεία της τροφοδοσίας φαίνονται στην επόμενη φωτογραφία.



Εικόνα 3-11: Το σχέδιο με τις αναλυτικές λειτουργίες και την τοποθέτηση τους στο μοντέλο Maker Pi

Τέλος, είναι σημαντικό να σχολιαστούν οι διαστάσεις της πλακέτας, η οποία είναι μικρότερη από την Maker Pi Pico, χωρίς να υστερεί σε κάτι σε σχέση με αυτή. Επίσης, το κόστος της ανέρχεται περίπου στα 15€, σχετικά ακριβότερα από την προηγούμενη αλλά δεν είναι απαραίτητη η αγορά του Raspberry Pi Pico.

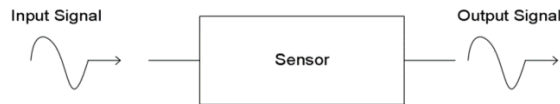


Εικόνα 3-12: Οι διαστάσεις της πλακέτας Maker Pi RP2040[33]

### 3.2. Αισθητήρες

Ένας αισθητήρας είναι μια συσκευή που λαμβάνει ένα ερέθισμα από τον φυσικό κόσμο και το μετατρέπει σε ηλεκτρικό σήμα [34]. Αυτά τα φαινόμενα μπορεί να είναι το φως, τα αέρια, οι υδρατμοί και ούτω καθεξής. Εκτός αυτών, φαινόμενα όπως θερμοκρασία, ηλεκτρικό ρεύμα, νερό, άνεμος και άλλα μπορούν επίσης να μετρηθούν από τους αισθητήρες. Αυτοί είναι συνήθως χαμηλού κόστους ηλεκτρονικές διατάξεις που έχουν σχεδιαστεί για να μπορούν να μετρήσουν, συνήθως, μόνο ένα από τα φαινόμενα.

Η αρχή λειτουργίας των αισθητήρων είναι απλή. Ένας αισθητήρας δεν επηρεάζει το φαινόμενο που μετράει, αλλά συλλέγει ένα μέρος του και το μετατρέπει σε ηλεκτρικό σήμα, ώστε να μπορέσει να γίνει η επεξεργασία. [28]



Εικόνα 3-13: Διάγραμμα τυπικής λειτουργίας ενός αισθητήρα [56]

Η βασική κατηγοριοποίηση των αισθητήρων είναι σε αναλογικούς και ψηφιακούς. Η κύρια διαφοροποίηση τους είναι στο αποτέλεσμα το οποίο δίνουν. Οι αναλογικοί αισθητήρες βγάζουν ως αποτέλεσμα ένα εύρος τάσης, συνήθως μεταξύ 0 και 5V, αλλά αυτό εξαρτάται από τον αισθητήρα. Στην συνέχεια, απαιτείται ένα ειδικό «αναλογικό σε ψηφιακό» κύκλωμα για τη μετατροπή της τάσης σε αριθμό.

Από την άλλη, οι ψηφιακοί αισθητήρες έχουν σχεδιαστεί με σκοπό να δίνουν μια σειρά bits, αλλά μόνο ένα bit κάθε φορά. Τα bits αντιπροσωπεύονται με 1 ή 0, ανάλογα την τάση. Η υψηλή τάση, που σε κάποιες περιπτώσεις χαρακτηρίζεται και ως «ON» είναι 1 και η χαμηλή τάση ή «OFF» είναι 0. Αυτές οι ακολουθίες τιμών 1 και 0 ονομάζονται διακριτές, επειδή ο αισθητήρας παράγει το ένα ή το άλλο σε παλμούς.

Υπάρχουν πολλοί τύποι αισθητήρων που χρησιμοποιούνται σε διάφορους τομείς ανάλογα το φαινόμενο που μετράνε. Για παράδειγμα ένας αισθητήρας υγρασίας είναι απαραίτητος σε μία αποθήκη τροφίμων αλλά όχι σε μια κατοικία. Ένας αισθητήρας μέτρησης στάθμης αποτελεί απαραίτητο και πολύ σημαντικό μέτρο ασφαλείας σε ένα διυλιστήριο αλλά είναι απλά βοηθητικός σε ένα οικιακό περιβάλλον. Επομένως η επιλογή των αισθητήρων γίνεται σύμφωνα με τις απαιτήσεις του προβλήματος.

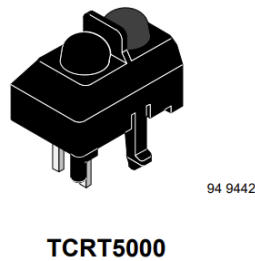
Στην περίπτωση μιας ρομποτικής διάταξης, ειδικά μίας κινούμενης, είναι απαραίτητα αισθητήρια όργανα τα οποία βοηθάνε στην αλληλεπίδραση της με το περιβάλλον. Αυτοί οι αισθητήρες είναι συνήθως αισθητήρες μέτρησης φωτός, αποστάσεων, χρώματος και ταχύτητας. Για την ρομποτική διάταξη της εργασίας προτιμήθηκαν στο αρχικό στάδιο που βρίσκεται να χρησιμοποιηθούν αισθητήρες μέτρησης φωτός και ταχύτητας. Συγκεκριμένα οι αισθητήρες TCRT5000 και οι Photo Interrupter / Encoder

### 3.2.1. Αισθητήρας Infrared TCRT5000- Line Follower

Ο βασικός λόγος που επιλέχθηκε ο αισθητήρας μέτρησης φωτός είναι εξαιτίας της πληθώρας εφαρμογών που έχει σε μία ρομποτική πλατφόρμα. Αυτός ο αισθητήρας μπορεί να χρησιμοποιηθεί για να αναγνωρίσει το ρομπότ εμπόδια, να ξεχωρίσει χρώματα (σκούρο ή ανοιχτό) και να ακολουθήσει μια προσχεδιασμένη διαδρομή. Αυτά είναι μερικές πολύ βασικές απαιτήσεις για μπορέσει το ρομποτικό σύστημα να κινηθεί.

Για αυτήν την λειτουργία επιλέχθηκε ο αισθητήρας TCRT500, ο οποίος είναι ένας αισθητήρας υπέρυθρου φωτός (Infrared). Έχει σχεδιαστεί με βάση την αρχή λειτουργίας των αντανακλαστικών αισθητήρων (reflective sensors). Το φως, που προέρχεται από ένα φωτάκι LED, εκπέμπεται από τον πομπό και φτάνει στο αντικείμενο που βρίσκεται μπροστά του. [35]Αυτό αντανακλά το φως, το οποίο εισέρχεται στον δέκτης και μετατρέπεται σε ηλεκτρικό σήμα. Με αυτό τον τρόπο έχει γίνει μία μέτρηση. Αν δεν υπάρχει αντικείμενο μπροστά στον αισθητήρα τότε η τιμή του φωτός που επιστρέφει εξαρτάται από την ευαισθησία του.

Σε αντανακλαστικούς αισθητήρες, ο πομπός και ο δέκτης είναι τοποθετημένοι ο ένας δίπλα στον άλλον. Ο πομπός του αισθητήρα TCRT5000 εκπέμπει ακτινοβολία μήκους κύματος 950 nm, δηλαδή υπέρυθρο φως. [36] Ένα φωτοτρανζίστορ είναι ο δέκτης, τα οποία βελτιστοποιούν την ανάγνωση τους στο φάσμα του ερυθρού προς υπέρυθρου φωτός. Ο αισθητήρας στην παρακάτω εικόνα είναι χωρίς ολοκληρωμένο κύκλωμα.



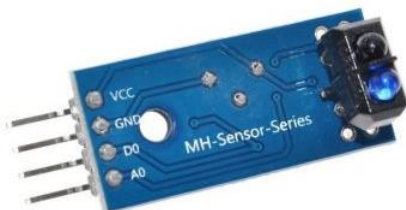
(α)



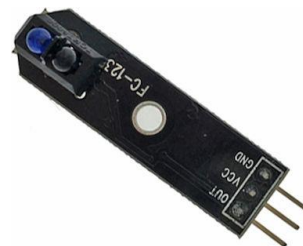
(β)

*Εικόνα 3-14: (α) Σχέδιο του αισθητήρα TCRT 5000 όπως εμφανίζεται στο φύλλο δεδομένων[35] (β) ο αισθητήρας TCRT 5000 όπως διατίθεται στο εμπόριο*

Από την παραπάνω φωτογραφία είναι ξεκάθαρο ότι ο πομπός και ο δέκτης έχουν διαφορετικά χρώματα. Συγκεκριμένα, η μπλε πλευρά είναι το LED, που εκπέμπει υπέρυθρο φως, και το μαύρο είναι ο δέκτης. Αν και είναι διαθέσιμοι στην αγορά με αυτή την μορφή, συνήθως προτιμάται να αγοραστούν μαζί με το ολοκληρωμένο κύκλωμα. Είναι πιο εύρηστοι και είναι ευκολότερο να προστεθούν σε ένα σύστημα όταν έχουν έτοιμη την ηλεκτρονική διάταξη. Οι δύο εκδοχές που υπάρχουν διαθέσιμες είναι μόνο ψηφιακοί ή ψηφιακοί και αναλογικοί.



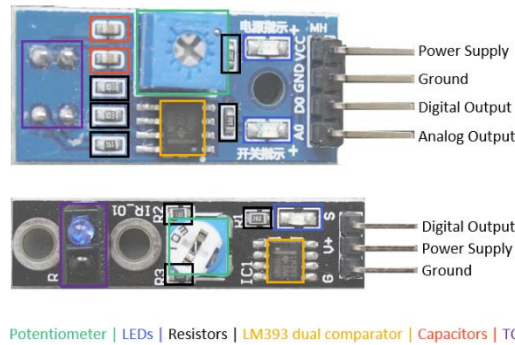
(α)



(β)

*Εικόνα 3-15: Δύο από τις εκδοχές που διατίθεται στο εμπόριο για τον αισθητήρα TCRT5000 [57] (α) με αναλογικές και ψηφιακές ακίδες και (β) μόνο με ψηφιακή ακίδα*

Οι αισθητήρες με το ολοκληρωμένο κύκλωμα έχουν μερικά ηλεκτρονικά κομμάτια. Αρχικά ένα ποτενσιόμετρο για να ορίζεται η τάση που αλλάζει το ψηφιακό αποτέλεσμα και δύο φωτάκια LED ένα που δείχνει αν ο αισθητήρας εντοπίζει αντικείμενο ή όχι και ένα για την λειτουργία του, αν δέχεται ή σε δέχεται τάση. Επίσης, είναι απαραίτητα μερικές αντιστάσεις για να μην υπερφορτωθούν τα συστήματα αλλά και για να χωρίζουν την τάση. Ακόμα, έχουν το LM393 για τη σύγκριση του σήματος που δημιουργείται από τον δέκτη υπέρυθρων με την προκαθορισμένη τιμή μέσω του ποτενσιόμετρου και για τον έλεγχο της κατάστασης του LED που υποδεικνύει την κατάσταση της ψηφιακής εξόδου.

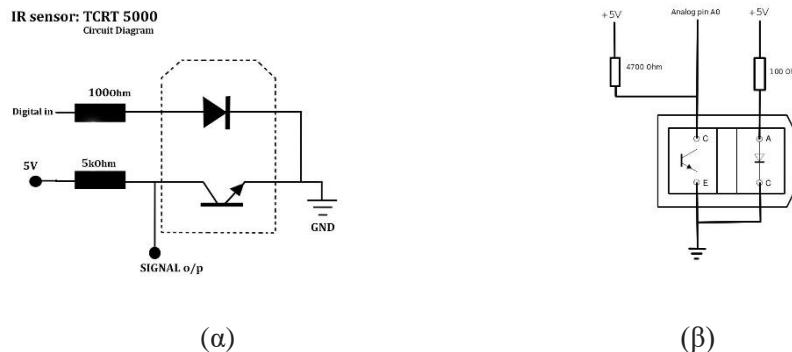


Εικόνα 3-16: Τα ηλεκτρονικά κομμάτια από τα οποία αποτελείται το ολοκληρωμένο ηλεκτρονικό σύστημα και η ανάλυση των ακίδων [58]

Ένα σημαντικό χαρακτηριστικό και των δύο είναι πως το ποτενσιόμετρο είναι ελεγχόμενο από τον χρήστη με την χρήση ενός απλού κατσαβιδιού. Η μεγαλύτερη διαφορά που έχουν μεταξύ τους είναι οι ακίδες. Στην περίπτωση του απλού ψηφιακού υπάρχουν τρεις ακίδες, ενώ στην άλλη εκδοχή υπάρχουν τέσσερις ακίδες. Αυτές είναι:

- A0: υπάρχει μόνο στον αναλογικό και ψηφιακό, μεταφέρει το αναλογικό σήμα
- GND/G: Ακίδα γείωσης
- VCC/V+: Ακίδα για την τάση εισόδου
- D0/S: Ακίδα για ψηφιακό σήμα, σύμφωνα με το όριο που έχει τεθεί από τον ποτενσιόμετρο

Τα ηλεκτρονικά σχέδια και για τις δύο εκδοχές παρουσιάζονται στην συνέχεια. Αυτά τα σχέδια είναι ανεξάρτητα από την πλακέτα που χρησιμοποιείται.



Εικόνα 3-17: Τα ηλεκτρονικά σχέδια του αισθητήρα TCRT5000 (α) μόνο με ψηφιακή ακίδα και (β) με αναλογικές και ψηφιακές ακίδες [57]

Αυτοί οι αισθητήρες χρησιμοποιούνται σε αρκετές ρομποτικές κατασκευές, με βασική λειτουργία τους την ακολουθία μιας προσχεδιασμένης διαδρομής. Είναι γνωστοί στην αγορά με το όνομα «TCRT500 Line follower». Αυτό αποδεικνύεται από το φύλλο δεδομένων της εταιρείας που τους παράγει. Σύμφωνα με τα δεδομένα, όταν ο αισθητήρας είναι τοποθετημένος κάθετα στην επιφάνεια και σε απόσταση από 2 έως 8 mm [37], βρίσκειται στην καλύτερη θέση.

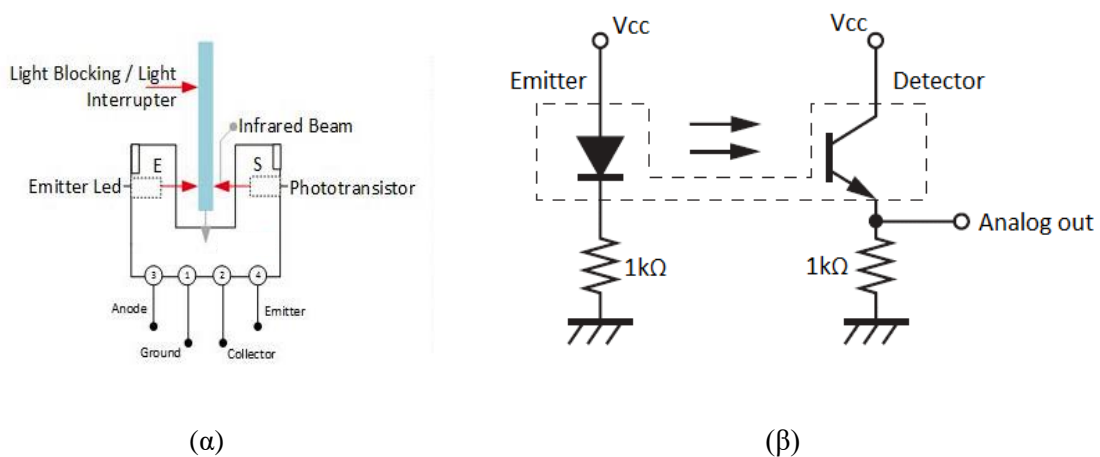
Ωστόσο, όπως όλα τα πράγματα, αυτός ο αισθητήρας έχει μερικά αρνητικά. Πρώτον, επηρεάζεται σε μεγάλο βαθμό από το περιβάλλον του. Ο δέκτης δεν είναι προστατευμένος με κάποιο τρόπο και το διάχυτο φως επηρεάζει τις μετρήσεις. Αυτό σημαίνει ότι μέσα στην μέρα ο αισθητήρας θα χρειαστεί να βαθμονομηθεί αρκετές φορές. Επίσης, δεν είναι κάθε επιφάνεια και κάθε αντικείμενο

το ίδιο. Το χρώμα, αλλά και το είδος της επιφάνειας επηρεάζουν την μέτρηση. Τέλος, δεν μπορεί να ξεχωρίσει διαφορετικά αντικείμενα. Μπορεί να μετρήσει την απόσταση, αλλά δεν μπορεί να διακρίνει ούτε το χρώμα αλλά ούτε και το είδος τους.

Από τα παραπάνω, μπορεί κανείς να συμπεραίνει ότι είναι ένας απλός αισθητήρας που βρίσκει εφαρμογή σε μια αρκετά σημαντική λειτουργία της ρομποτικής διάταξης. Από τις δύο διαθέσιμες εκδοχές, για την ρομποτική διάταξη επιλέγεται ο αναλογικός και ψηφιακός αισθητήρας, καθώς, όπως θα αναλυθεί και σε επόμενο κεφάλαιο, η προγραμματιστική ρουτίνα απαιτεί ένα εύρος τιμών για να λειτουργήσει σωστά. Επομένως ο απλός ψηφιακός αισθητήρας θα προκαλούσε μεγάλη δυσκολία στον προγραμματισμό του.

### 3.2.2. Οπτικός Αισθητήρας Μέτρησης Περιστροφών (Photo Interrupter)

Στην προηγούμενη ενότητα αναλύθηκε ένας τύπος αντανακλαστικού αισθητήρα. Βέβαια, δεν είναι ο μοναδικός τύπος. Ένας άλλος τύπος είναι ο οπτικός αισθητήρας μέτρησης περιστροφών, γνωστός με την αγγλική ονομασία Photo Interrupter. Σε αντίθεση με τον προηγούμενο τύπο, στον αισθητήρα αυτόν ο πομπός και ο δέκτης δεν βρίσκονται δίπλα ο ένας στον άλλο. Αυτά βρίσκονται απέναντι, σε μία μικρή απόσταση και χρησιμοποιείται στις περισσότερες εφαρμογές για μικρές αποστάσεις και στενά αντικείμενα.

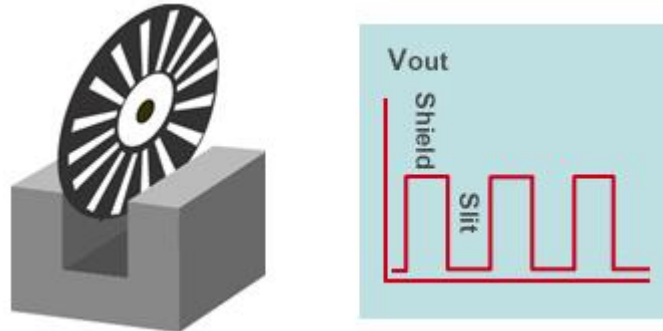


Εικόνα 3-18: (α) Σχηματική αναπαράσταση του οπτικού αισθητήρα μέτρησης περιστροφών [59] και (β) ηλεκτρονικό σχέδιο του αισθητήρα [60]

Σε μία κινούμενη ρομποτική διάταξη μπορεί να χρησιμοποιηθεί για την μέτρηση περιστροφών. Στον άξονα του κινητήρα τοποθετείται ένας διάτρητος τροχός με κενά ανά γνωστές μοίρες (αλλάζουν ανάλογα τον τροχό). Στην συνέχεια, τοποθετείται ο αισθητήρας με τέτοιο τρόπο ώστε ο τροχός να βρίσκεται ενδιάμεσα από τον πομπό και τον δέκτη. Με αυτό τον τρόπο ανάλογα με το πόσες φορές ο δέκτης λαμβάνει το φως από τον πομπό σημαίνει ότι ο τροχός έχει περιστραφεί τις ανάλογες μοίρες.

Συγκεκριμένα ο υπολογισμός της ταχύτητας του τροχού είναι αρκετά απλός. Για να μην υπάρχει παρανόηση τα νούμερα είναι προσαρμοσμένα για να γίνουν οι πράξεις με μεγαλύτερη ευκολία, αλλά η λογική είναι κοινή για κάθε περίπτωση.

Ο διάτρητος τροχός έχει 20 εγκοπές. Επομένως ο κινητήρας έχει εκτελέσει μια ολόκληρη περιστροφή μόλις μετρήσει 20. Στην συνέχεια χρειάζεται να είναι γνωστό το πλήθος των παλμών (PWM) ανά δευτερόλεπτο και έχοντας αυτό γνωστό είναι εύκολο να βρεθεί η πραγματική ταχύτητα το τροχού. Ωστόσο, η ταχύτητα του τροχού και του κινητήρα διαφέρουν λόγω των εσωτερικών μηχανισμών που τον επιβραδύνουν ελαφρά. Αν οι παλμοί ανά δευτερόλεπτο του κινητήρα είναι 20, τότε ο τροχός περιστρέφεται μία φορά το δευτερόλεπτο και επομένως, πολλαπλασιάζοντας με το 60 για να υπολογισθεί ανά λεπτό, έχουμε 60 στροφές ανά λεπτό.



Εικόνα 3-19: Αναμενόμενο διάγραμμα που εξελίσσεται στο χρόνο για τον οπτικό αισθητήρα μέτρησης περιστροφών [60]

Μετά, με μερικές ακόμα μαθηματικές πράξεις υπολογίζονται και μερικές άλλες χρήσιμοι παράμετροι. Μια εξαιρετικά σημαντική και χρήσιμη είναι ο προσδιορισμός της μετατόπισης της διάταξης. Για να υπολογισθεί αυτό, είναι απαραίτητη η περίμετρος της ρόδας, η οποία είναι ο πολλαπλασιασμός της διάμετρο με το  $\pi$  (3.14). Αυτή είναι η απόσταση που θα διανύσει το ρομπότ σε μία περιστροφή. Επομένως, ανάλογα με το πόσο μακριά και πόσο γρήγορα είναι απαραίτητο να κινηθεί, προσαρμόζονται τα νούμερα αυτά.

Ο αισθητήρας που επιλέχθηκε για την ρομποτική διάταξη είναι ο Optical photo interrupter sensor, της εταιρείας Waveshare. Ο συγκεκριμένος είναι μόνο ψηφιακός και έχει την είσοδο Grove, όπως και οι πλακέτες που αναλύθηκαν. Οι ακίδες είναι τρεις:

- DOUT: ακίδα του ψηφιακού σήματος
- GND: ακίδα της γείωσης
- VCC: ακίδα τροφοδοσίας (3.3 με 5 V)

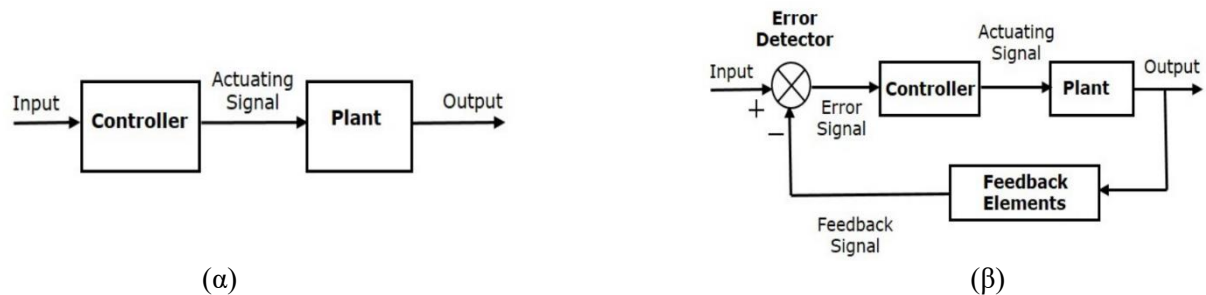


Εικόνα 3-20: (α) οπτικός αισθητήρα μέτρησης περιστροφών και (β) ο διάτρητος τροχός που χρησιμοποιείται για την μέτρηση των περιστροφών [61]



## Κεφάλαιο 4ο Γλώσσα προγραμματισμού

Για λειτουργήσει μία ρομποτική διάταξη χρειάζεται να υπάρχει το αντίστοιχο σύστημα ελέγχου. Αν και υπάρχουν δύο ειδών συστήματα αυτόματου ελέγχου (ΣΑΕ), τα ανοιχτού και τα κλειστού βρόγχου, είναι πολύ πιο διαδεδομένα τα συστήματα κλειστού βρόγχου. Συγκεκριμένα, σε ένα σύστημα κλειστού βρόγχου υπάρχει ανατροφοδότηση των τελικών δεδομένων σε έναν ελεγκτή με σκοπό την διόρθωση του σφάλματος όσο το δυνατόν γρηγορότερα, ενώ αντίθετα σε ένα σύστημα ανοιχτού βρόγχου δεν υπάρχει αυτό. Παρακάτω φαίνεται τα σχήματα με τα οποία συνηθίζεται να αναπαρίστανται τα συστήματα αυτά.



Διάγραμμα 4-1: Παράδειγμα ΣΑΕ (α) Σύστημα Ανοιχτού Βρόγχου (β) Σύστημα Κλειστού Βρόγχου

Οι ρομποτικές διατάξεις αποτελούν πλέον βασικό κομμάτι της καθημερινότητας σε όλους τους τομείς. Χρησιμοποιούνται για μια πληθώρα εργασιών και διαδικασιών, ανάλογα με το μέγεθος τους και τις απαιτήσεις που καλούνται να καλύψουν. Έτσι, είναι πιθανόν κανείς να συναντήσει ρομποτικά συστήματα μεγάλου μεγέθους, όπως σε εργοστάσια [38], αλλά και μικρότερα[39], τα οποία όλα χρησιμοποιούν κάποιου είδους αλγόριθμο για την κίνηση τους.

Με την επιλογή της ηλεκτρονικής διάταξης, το φυσικό μέρος της ρομποτικής διάταξης έχει ολοκληρωθεί. Αυτό που ακολουθεί είναι η επιλογή μιας πιο σύγχρονης γλώσσας προγραμματισμού, για να μπορέσει το σύστημα να εκτελέσει τις εργασίες για τις οποίες έχει δημιουργηθεί. Ένα μεγάλο πλεονέκτημα που έχει ο μικροελεγκτής RP2040, ο οποίος αποτελεί την βάση στις ηλεκτρονικές διατάξεις που επιλέχθηκαν, είναι ότι μπορεί να προγραμματιστεί με Python, συγκεκριμένα με MicroPython.

Η Python είναι μια πολύ δημοφιλής γλώσσα προγραμματισμού και είναι εντελώς δωρεάν και άμεσα διαθέσιμη στο ευρύ κοινό. Δεν είναι μία νέα γλώσσα, όμως είναι δοκιμασμένη και αξιόπιστη. Αναπτύχθηκε από τον Guido van Rossum και δημοσιεύτηκε το 1991. Η εισαγωγή της Python στην εκπαίδευση έχει γίνει εδώ και αρκετά χρόνια με κύριους τομείς την εκμάθηση προγραμματισμού. Σαν προγραμματιστική γλώσσα, είναι ιδιαίτερα απλή και αυτό την κάνει μια πολύ καλή επιλογή για αρχάριους αλλά και για επαγγελματίες.

Η MicroPython είναι μία σύγχρονη λιτή και αποτελεσματική προσπάθεια εφαρμογής της Python 3 για προγραμματισμό μικροελεγκτών.[40] Χαρακτηρίζεται από το περιβάλλον της Python και περιλαμβάνει ένα υποσύνολο των βιβλιοθηκών της, καθώς και μερικές καινούργιες που είναι σχεδιασμένες για μικροελεγκτές. Είναι προφανές ότι μπορεί να χρησιμοποιηθεί για τον έλεγχο όλων των ειδών ηλεκτρονικών διατάξεων. Στην συνέχεια, περιγράφεται η εφαρμογή της στην ρομποτική διάταξη και παρατίθενται μερικά σημαντικά αποκόμματα κώδικα.

Ένα τροχοκίνητο ρομποτικό όχημα, όπως αυτό της εργασίας, έχει σχεδιαστεί να εκτελεί κάποιες εργασίες. Η πιο βασική είναι να μπορεί να κάνει διάφορες διαδρομές στον χώρο. Η επιλογή των αισθητήρων έγινε με βάση και τις προγραμματιστικές ανάγκες της διάταξης. Μια αρκετά απλή και πολυχρησιμοποιημένη προγραμματιστική ρουτίνα είναι η ρουτίνα ακολουθία διαδρομής που βρίσκεται στο έδαφος. Η ρουτίνα έχει δημιουργηθεί από τις αρχές του 19ου αιώνα, αλλά δεν είχε βρει τις κατάλληλες εφαρμογές ακόμα.

Η σύγχρονη εποχή, με την τεχνολογία με τους ρυθμούς ανάπτυξης και τις απαιτήσεις, ωθεί την εξέλιξη σε φρενήρεις ρυθμούς. Ένας τομέας που επηρεάζεται άμεσα είναι τα συστήματα ελέγχου, καθώς πάνω σε αυτά καλείται να βασιστεί η «τέταρτη βιομηχανική επανάσταση» [41] Αυτό που μπορεί αποφανθεί από τα τωρινά δεδομένα είναι ότι η πίεση που ασκείται στα συστήματα αυτοματισμού και ελέγχου δεν πρόκειται να πάψει να υφίσταται και αυτό οδηγεί σε νέες ερευνητικές ερωτήσεις.

#### **4.1. MicroPython**

Ο βασικός σκοπός της MicroPython είναι να συνδυαστούν η ευκολία στη μάθηση που έχει η Python με τους μικροελεγκτές. Με αυτόν τον τρόπο γίνεται ευκολότερη η υλοποίηση ιδεών με ηλεκτρονικές διατάξεις, καθώς υπάρχουν περισσότερες επιλογές τόσο για αρχαίους αλλά και για επαγγελματίες.

Η MicroPython είναι μια λιτή και αποτελεσματική έκδοση της Python, που χρησιμοποιείται για τον προγραμματισμό μικροελεγκτών. Αυτό έρχεται σε αντίθεση με την Python, η οποία είναι προγραμματιστική γλώσσα που χρησιμοποιεί διερμηνέα και καθυστερεί περισσότερο από άλλες γλώσσες που χρησιμοποιούν μεταγλωττιστές. Έτσι η MicroPython σχεδιάστηκε για να είναι όσο το δυνατόν πιο αποτελεσματική, ώστε να μπορεί να λειτουργεί σε μικροελεγκτές που κανονικά είναι πιο αργοί και έχουν πολύ λιγότερη μνήμη από μια τυπική διάταξη.

Ωστόσο, ο διερμηνέα προσφέρει ένα σημαντικό πλεονέκτημα. Βρίσκεται απευθείας στον μικροελεγκτή και έτσι μπορεί να παρακαμφθεί η προετοιμασία του κώδικα, που είναι απαραίτητη για γλώσσες που έχουν μεταγλωττιστές. [28] Αυτό επιτρέπει στους κατασκευαστές μικροελεγκτών να κατασκευάζουν μικρές, φθηνές πλακέτες που περιλαμβάνουν MicroPython στο ίδιο τσιπ με τον επεξεργαστή. Αυτό τη δυνατότητα στον χειριστή να συνδεθεί στο υλικό, να γράψει τον κώδικα και να τον εκτελέσει άμεσα.

Το μεγαλύτερο θετικό της MicroPython είναι ότι χρησιμοποιεί την Python. Εξίσου σημαντικό είναι ότι είναι συμβατή με την πλειοψηφία των βιβλιοθηκών της Python, όπως είναι η JavaScript Object Notation (JSON). Επίσης, διαθέτει ενσωματωμένες βιβλιοθήκες στο υλικό και επιτρέπει έτσι την απευθείας ενεργοποίηση ή απενεργοποίηση συστημάτων. Σε αυτά περιλαμβάνονται οι ακίδες, αναλογικές και ψηφιακές, η ανάγνωση αναλογικών δεδομένων και ο έλεγχος υλικού μέσω της διαμόρφωσης εύρους- παλμού (Pulse- Width Modulation/ PWM). Ακόμα έχει την δυνατότητα επέκτασης με προσθήκη νέων βιβλιοθηκών, για έμπειρους χρήστες.[40]

Ωστόσο, πάρα τα πολλά πλεονεκτήματα, η MicroPython έχει και κάποιους περιορισμούς. Το μεγαλύτερο μειονέκτημα της είναι η ευκολία της χρήσης της, που σημαίνει ότι ο κώδικας εκτελείται δυναμικά. Έτσι συστήματα, τα οποία απαιτούν υψηλό βαθμό ακριβείας, όπως η δειγματοληψία δεδομένων με υψηλό ρυθμό ή επικοινωνία μέσω USB, ενδέχεται να μην εκτελούνται με την επιθυμητή ταχύτητα. Σε μερικές περιπτώσεις έχει περιορισμένη πρόσβαση στο υλικό. [42] Επίσης, η MicroPython

χρησιμοποιεί περισσότερη μνήμη σε σχέση με άλλες γλώσσες προγραμματισμού και δεν είναι συμβατή με όλες τις βιβλιοθήκες της Python 3, κάποιες από τις οποίες είναι ιδιαίτερα εύχρηστες.

Συνεπώς, η εισαγωγή της MicroPython στην εκπαίδευση πάρα τα αρνητικά έχει πολλά να προσφέρει. Είναι μια εξαιρετική επιλογή τόσο για αρχάριους όσο και για επαγγελματίες. [43] [44] Βοηθάει τους φοιτητές να αναπτύσσουν τις δεξιότητες τους στον προγραμματισμό κατά την διάρκεια των σπουδών τους και αποτελεί μια θετική εμπειρία για αυτούς.

#### 4.1.1. Βιβλιοθήκες Python/ MicroPython

Οι βιβλιοθήκες στις γλώσσες προγραμματισμού είναι κώδικες οι οποίοι βοηθούν τους προγραμματιστές να βελτιστοποιήσουν τον δικό τους κώδικα. Στην Python υπάρχουν αρκετές βιβλιοθήκες που χρησιμοποιούνται καθημερινά από προγραμματιστές και κάποιες από αυτές είναι συμβατές και με την MicroPython. Μία από τις πιο σημαντικές είναι η “io”, η οποία χρησιμεύει για την είσοδο και την έξοδο δεδομένων από τον χρήστη. Εξίσου χρήσιμη είναι η βιβλιοθήκη time με την οποία ορίζεται ο χρόνος για τον οποίο θα εκτελεστεί το πρόγραμμα. Στον πίνακα αναγράφονται μερικές από τις βιβλιοθήκες της Python που είναι συμβατές με την MicroPython.

*Πίνακας 4-1: Τυπικές βιβλιοθήκες της Python*

<b>Όνομα βιβλιοθήκης</b>	<b>Περιγραφή βιβλιοθήκης</b>
<i>array</i>	Πίνακες αριθμητικών δεδομένων
<i>binascii</i>	Μετατροπή σε δυαδικό ή ASCII
<i>cmath</i>	Μαθηματικές συναρτήσεις για μιγαδικούς αριθμούς
<i>errno</i>	Κώδικες σφάλματος του συστήματος
<i>io</i>	Είσοδοι/ έξοδοι δεδομένων
<i>json</i>	Κωδικοποίηση και αποκωδικοποίηση JSON
<i>os</i>	Βασικές λειτουργίες του συστήματος
<i>random</i>	Επιλογή τυχαίων αριθμών
<i>re</i>	Απλές κοινές εκφράσεις
<i>sys</i>	Συγκεκριμένες λειτουργίες συστήματος
<i>time</i>	Χρονικές συναρτήσεις
<i>uasyncio</i>	Ασύγχρονος συντονισμός δεδομένων εισόδου/εξόδου

Ωστόσο, έχουν δημιουργηθεί μερικές βιβλιοθήκες αποκλειστικά για MicroPython. Η πιο σημαντική είναι η βιβλιοθήκη “machine”, η οποία ουσιαστικά βοηθάει να ελέγχει το φυσικό υλικό του συστήματος. Είναι βασική και εισάγεται πρώτη σε κάθε πρόγραμμα το οποίο αφορά ρομποτική διάταξη. Κάποιες από τις βιβλιοθήκες φαίνονται στον παρακάτω πίνακα.

Πίνακας 4-2: Τυπικές βιβλιοθήκες της MicroPython

Όνομα βιβλιοθήκης	Περιγραφή βιβλιοθήκης
<i>bluetooth</i>	Για χαμηλού επιπέδου Bluetooth
<i>machine</i>	Λειτουργίες για εργασίες στο φυσικό υλικό
<i>micropython</i>	Πρόσβαση και έλεγχος των εσωτερικών λειτουργιών της MicroPython
<i>neopixel</i>	Έλεγχος WS2812/ NeoPixel LEDs
<i>network</i>	Σύνδεση με το δίκτυο
<i>uctypes</i>	Πρόσβαση σε δυαδικά δεδομένα

Τέλος υπάρχει μία βιβλιοθήκη αποκλειστικά για τις λειτουργίες του RP2040, η βιβλιοθήκη “*tr2*”. Αυτή αφορά κυρίως τις λειτουργίες εισόδου και εξόδου δεδομένων μέσω των ακίδων (Pin Input/Output- PIO). Στην συνέχεια δίνονται μερικά παραδείγματα κώδικα για τις βασικές λειτουργίες της ρομποτικής διάταξης μέσω της MicroPython.

#### 4.1.2. Ψηφιακές ακίδες

Οι ψηφιακές ακίδες απαιτούν το πιο απλό πρόγραμμα για την λήψη δεδομένων. Σε κάθε πρόγραμμα οι πρώτες γραμμές αφιερώνονται στην εισαγωγή των απαραίτητων βιβλιοθηκών και στην συνέχεια συντάσσεται το κυρίως πρόγραμμα. Το συνηθέστερο υλικό που συνδέεται σε αυτές είναι τα λαμπάκια LED.

Συνήθως το πρώτο πρόγραμμα που δοκιμάζεται σε ένα μικροελεγκτή είναι ο έλεγχος των LED. Η πλακέτα RP2040 έχει 13 προγραμματιζόμενα LED, που αντιστοιχούν στις ακίδες P0- P7, P16-P17 και P26-P28. Στο παρακάτω πρόγραμμα φαίνεται ο κώδικας που χρησιμοποιείται για να ανάψει το λαμπάκι στην ακίδα 5. Χρησιμοποιούνται οι βιβλιοθήκες “*machine*” και “*time*”. Συγκεκριμένα, ορίζεται μία μεταβλητή και μετά αυτή μέσω της βιβλιοθήκης “*machine*” μπορεί να λάβει τιμές.

Πίνακας 4-3: Κώδικας για την ακίδα 5 της πλακέτας RP2040.

---

```
# code for LED light fro Maker Pi RP2040
# digital pin 5

# importing libraries
import machine
import time

# initialize variables
p5 = machine.Pin(5, machine.Pin.OUT)
```

---

---

```
# LED in pin 5 is turned on for 3 seconds
```

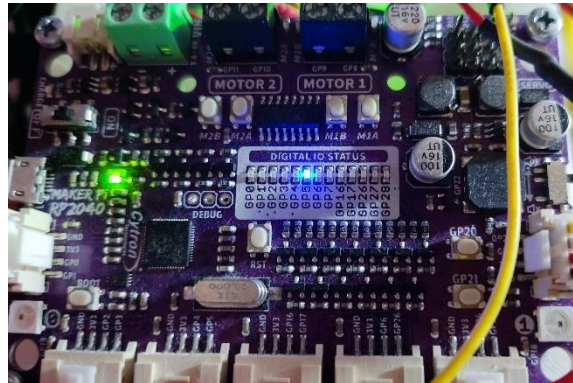
```
p5.value(1)
```

```
time.sleep(3)
```

```
# LED in pin 5 is turned off
```

```
p5.value(0)
```

---



Εικόνα 4-1: Αποτέλεσμα του κώδικα του πίνακα 4-3

Η πλακέτα RP2040 έχει και 2 NeoPixel LED φωτάκια. Ο κώδικας για αυτά είναι λίγο πιο περίπλοκος. Χρειάζεται να οριστεί μία μεταβλητή, όπως και προηγουμένως αλλά στην συνέχεια πρέπει να οριστεί και το χρώμα, με βάση την λογική RGB (Red, Green, Blue). Στην επόμενη φωτογραφία φαίνεται ο κώδικας για το NeoPixel για χρώμα κόκκινο.

Πίνακας 4-4: Κώδικας για τα NeoPixel LED, με χρήση εντολής “r,g,b” για τον ορισμό του χρώματος.

---

```
# Blinking NeoPixel
```

```
# Code for NeoPixel in Maker Pi RP2040
```

```
# Importing the necessary libraries
```

```
import machine
```

```
import neopixel
```

```
import time
```

```
pin = machine.Pin(18, machine.Pin.OUT) # set PIO0 to output to drive NeoPixels
```

```
np = neopixel.NeoPixel(pin, 8) # create NeoPixel driver on PIO0 for 8 pixels
```

```
np[0] = (150, 0, 0) # set the first pixel to color red
```

```
np.write() # write data to all pixels
```

```
r, g, b = np[0] # get first pixel colour
```

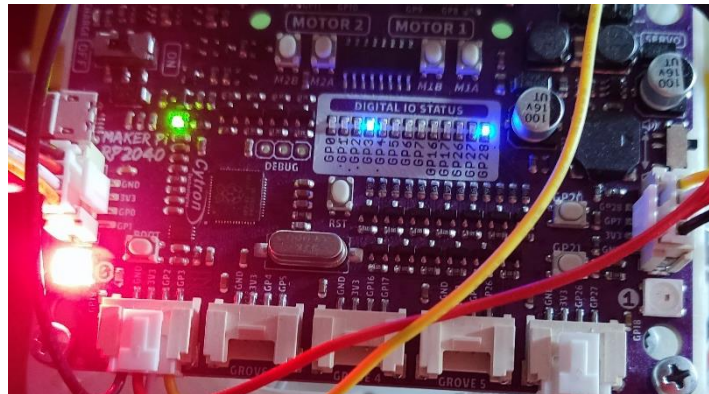
---

---

```
time.sleep(0.1)
```

```
np[0] = (0, 0, 0)           # set the first pixel to off  
np.write()                # write data to all pixels  
r, g, b = np[0]           # get first pixel colour  
time.sleep(0.05)
```

---



Εικόνα 4-2: Αποτέλεσμα κώδικα του πίνακα 4-4, όπου τελικά ανάβει το NeoPixel LED κόκκινο χρώμα

Για την ανάγνωση ψηφιακών αισθητήρων χρησιμοποιείται η ίδια σύνταξη, καθώς τόσο τα λαμπάκια LED όσο και οι ψηφιακοί αισθητήρες βασίζονται στην λογική των δύο τιμών, ON και OFF.

#### 4.1.3. Αναλογικές Ακίδες

Η εισαγωγή των αισθητήρων σε μία ρομποτική διάταξη την καθιστά ικανή να αλληλεπιδρά με το περιβάλλον της. Όμως, για να υπάρξουν δεδομένα για επεξεργασία χρειάζεται το πρόγραμμα να αναγνωρίζει τους αισθητήρες. Οι αισθητήρες μίας διάταξης μπορεί να είναι είτε ψηφιακοί είτε αναλογικοί. Όπως αναφέρεται και παραπάνω, το πρόγραμμα για τους ψηφιακούς αισθητήρες είναι ιδιαίτερα απλό.

Το πρόγραμμα για τους αναλογικούς διαφέρει κάπως, αλλά είναι το ίδιο εύκολο με τους ψηφιακούς. Αρχικά, είναι απαραίτητο να οριστούν οι μεταβλητές και να μετατραπούν από αναλογικές σε ψηφιακές τιμές. Αυτό γίνεται με την εντολή “machine.ADC”, που είναι μέρος της βιβλιοθήκης “machine”. Για να διαβάσει το πρόγραμμα την τιμή της αναλογικής ακίδας, η οποία έχει πλέον μετατραπεί σε ψηφιακή, χρειάζεται την εντολή “μεταβλητή.read\_u16()”. Αυτό επιτρέπει στον κώδικα να διαβάσει την μεταβλητή και συνεπώς την τιμή.

*Πίνακας 4-5: Κώδικας που χρησιμοποιείται για την ανάγνωση των αναλογικών ακίδων, συγκεκριμένα της ακίδας 0.*

---

```
# Code for Maker Pi RP2040
```

```
# Analog Sensor read
```

---

---

```

# Importing necessary libraries
import machine
import time

# Setting up the variables
analog1 = machine.ADC(0)

# Reading the sensor value for 3 seconds
print("1: ", analog1.read_u16())
time.sleep(3)

```

---

#### 4.1.4. Κινητήρες DC

Είναι φυσικό πως εκτός από τα LED, η διάταξη είναι εξοπλισμένη με κινητήρες. Το πρόγραμμα για να λειτουργήσουν οι κινητήρες έχει τις ίδιες βιβλιοθήκες με τα προηγούμενα. Σε αντίθεση με αυτά, εδώ χρησιμοποιείται το κομμάτι της βιβλιοθήκης “machine” που έχει να κάνει με τους κινητήρες το PWM. Επίσης, όπως φαίνεται και στον παρακάτω κώδικα, η μέγιστη ταχύτητα που μπορεί να φτάσει ο κινητήρας είναι 65535, ενώ μετά από δοκιμές η ελάχιστη ταχύτητα είναι περίπου 15000. Αυτά τα νούμερα δίνονται από τους κατασκευαστές και αναφέρονται στα 16 κανάλια που αφορούν το PWM, τα οποία υπάρχουν διαθέσιμα στην πλακέτα, δηλαδή

$$2^{16} = 65536$$

Επομένως το εύρος είναι από 0 έως 65535, με το 0 να αποτελεί την αρχική τιμή.

*Πίνακας 4-6: Κομμάτι κώδικα για τον προγραμματισμό κινητήρα DC*

---

```

# Code for Maker Pi RP2040
# Code for DC motor 1

# Importing necessary libraries
import machine
import time

# Setup DC Motor pins
M1A = machine.PWM(machine.Pin(8)) # Pin forward
M1B = machine.PWM(machine.Pin(9)) # Pin backward

M1A.freq(80) # Desired frequency

```

---

---

**M1B.freq(80)**

**# Speed range**

**max\_speed = 65535**

**min\_speed = 15000**

---

#### 4.1.5. Κινητήρες Servo

Ομοίως, για τους κινητήρες σέρβο ο κώδικας είναι παρόμοιος με τον κώδικα των DC κινητήρων, όπως είναι εμφανές από το απόκομμα που ακολουθεί.

*Πίνακας 4-7: Κομμάτι κώδικα για τον προγραμματισμό κινητήρων Servo*

---

**# Code for Maker Pi RP2040**

**# Code for servo motor in pin 12**

**# Importing necessary libraries**

**import machine**

**import time**

**# Servo motor P12**

**# fine tune the duty cycle values to suit your servo motor**

**MIN\_DUTY = 1000**

**MAX\_DUTY = 8400**

**pwm = machine.PWM(machine.Pin(12))**

**pwm.freq(50)**                      **# Desired frequency**

---

#### 4.1.6. Απλές Κινήσεις

Σε αυτό το σημείο, αφού έχουν οριστεί οι κατάλληλες ακίδες, συνεχίζεται το πρόγραμμα. Για να κινηθεί ο κινητήρας DC χρησιμοποιείται η εντολή “duty\_u16”. Αυτή η εντολή ανήκει στην βιβλιοθήκη “machine”. Σε συνέχεια της προηγούμενης εξήγησης το pulse-width modulation (PWM) είναι ένας τρόπος για να αποκτήσει μια τεχνητή αναλογική έξοδο σε μια ψηφιακή ακίδα. Αυτό επιτυγχάνεται με γρήγορη εναλλαγή του πείρου από χαμηλό σε υψηλό. Υπάρχουν δύο παράμετροι που σχετίζονται με αυτό: η συχνότητα της εναλλαγής και ο κύκλος λειτουργίας (duty cycle). Ο κύκλος λειτουργίας ορίζεται ως το χρονικό διάστημα για το οποίο η ακίδα έχει υψηλή τάση.



Σαν παράδειγμα δίνεται η κίνηση στον έναν εκ των δύο κινητήρων, καθώς στο κεφάλαιο που ακολουθεί δίνονται με μεγαλύτερη λεπτομέρεια περισσότερα κομμάτια κώδικα σχετικά με τις κινήσεις.

Πίνακας 4-8: Κομμάτι κώδικα για τον προγραμματισμό κινητήρα DC, ώστε να κινηθεί μπροστά (forward) για 2 δευτερόλεπτα και μετά να σταματήσει. (συνέχεια από τον πίνακα 4-6)

---

```

print("forward")
M1A.duty_u16(min_speed)      # Duty Cycle must be between 0 until 65535
M1B.duty_u16(0)
time.sleep(2)

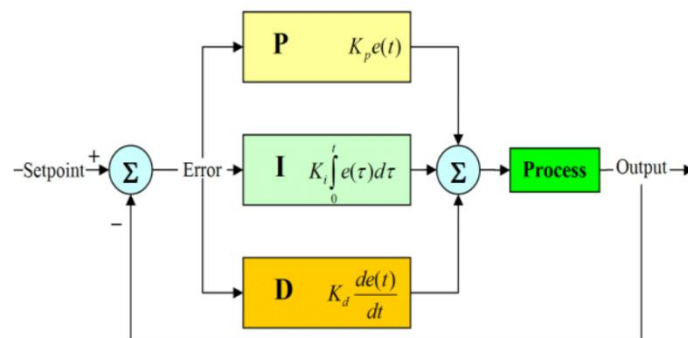
print("stop")
M1A.duty_u16(0)              # Duty Cycle must be between 0 until 65535
M1B.duty_u16(0)
time.sleep(1)

```

---

## 4.2. Αλγόριθμος PID/ PD

Ένας από τους βασικότερους και απλούστερους είναι ο αλγόριθμος PID, ο οποίος αναπτύχθηκε στις αρχές του 19<sup>ου</sup> αιώνα σαν θεωρία. [45] Ο ελεγκτής PID είναι μια ακολουθία εντελών κλειστού βρόγχου και ελέγχεται από τρεις συνιστώσες, την συνιστώσα *Proportional*, *P*, η οποία αφορά την γραμμική συμπεριφορά του συστήματος, την συνιστώσα *Integral*, *I*, που είναι η ολοκληρωματική έκφραση και την συνιστώσα *Derivative*, *D*, η οποία είναι η παραγωγική έκφραση του. Αυτός ο αλγόριθμος έχει δημιουργηθεί με σκοπό να ελέγχεται ένα κομμάτι του συστήματος, συνήθως κάποιος κινητήρας μέσω αισθητήρων, με σκοπό να βρίσκεται γύρω από μία τιμή-στόχο. Σε διαγραμματική μορφή φαίνεται στο διάγραμμα 2. [46]



Διάγραμμα 4-2: Απλός έλεγχος PID για ένα σύστημα. [46]

Σε αρκετά συστήματα της βιομηχανίας, η χρήση PID είναι ο βασικός τρόπος ελέγχου. Η απλούστερη μαθηματική έκφραση του ελεγκτή PID είναι για συνεχείς τιμές και είναι όπως φαίνεται παρακάτω:

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (1)$$

Όπου:

$t$ : μετρούμενη χρονική στιγμή

$u$ : ταχύτητα την συγκεκριμένη χρονική στιγμή

$e(t)$ : απόκλιση της μετρούμενης τιμής από την τιμή στόχο

$K_p$ : συντελεστής διόρθωσης για την αναλογική συνιστώσα

$\int e(t) dt$ : σύνολο των προηγούμενων σφαλμάτων του συστήματος

$K_i$ : συντελεστής διόρθωσης για την ολοκληρωματική έκφραση

$\frac{de(t)}{dt}$ : τιμή της παραγωγού του σφάλματος

$K_d$ : συντελεστής διόρθωσης για την παραγωγική έκφραση

Αυτή η έκφραση έχει ως εισόδους της τρεις συνιστώσες που δίνουν και το όνομα τους στον αλγόριθμο, το  $e(t)$  το  $\int e(t) dt$ , και το  $\frac{de(t)}{dt}$ . Έξοδος είναι η ταχύτητα  $u(t)$ , η οποία μεταβάλλεται σε κάθε χρονική στιγμή.[47] Αυτές οι μεταβλητές στο επίπεδο των μαθηματικών δεν έχουν κάποια φυσική σημασία, όμως, καθώς αυτή είναι η βασική μαθηματική σχέση για έναν αλγόριθμο που χρησιμοποιείται σε φυσικά συστήματα, αποκτούν μεγαλύτερη βαρύτητα.

Η συνιστώσα P, που περιγράφεται από το  $e(t)$ , σχετίζεται με το άμεσο σφάλμα που εμφανίζεται μεταξύ της μετρούμενης τιμής και της τιμής-στόχο και περιγράφεται κυρίως από μία απλή πρώτης τάξης μαθηματική εξίσωση, για αυτό ονομάζεται και «αναλογική» συνιστώσα. Αυτό το κομμάτι ελέγχει την άμεση ταχύτητα  $u(t)$ , σε περίπτωση που οι υπόλοιποι συντελεστές έχουν μηδενιστεί. Η έκφραση  $\int e(t) dt$  περιγράφει το σύνολο των τιμών των προηγούμενων σφαλμάτων μέχρι την χρονική στιγμή  $t$ . Αυτό το σημείο της μαθηματικής έκφρασης (1) σε ένα φυσικό σύστημα, αντιστοιχεί στο πλήθος των διακριτών προηγούμενων θέσεων που έχει βρεθεί. Τέλος, η παράγωγος του σφάλματος,  $\frac{de(t)}{dt}$ , προσπαθεί να προβλέψει το επόμενο πιθανό σφάλμα για να μειωθεί ο χρόνος διόρθωσης.

Με βάση τις παραπάνω εξηγήσεις και σύμφωνα με την έρευνα [48], η σχέση (1) μπορεί να αποτυπωθεί και ως εξής:

$$\Delta u(kT) = K_p u(kT) + K_i d(kT) + K_d a(kT) \quad (2)$$

Όπου τα  $u(kT)$ ,  $d(kT)$  και  $a(kT)$  δίνονται από τις εξισώσεις:

$$\text{Ταχύτητα:} \quad u(kT) = \frac{d(kT) - d[(k-1)T]}{T} \quad (3)$$

$$\text{Μετατόπιση:} \quad d(kT) = e(kT) \quad (4)$$

$$\text{Επιτάχυνση:} \quad a(kT) = \frac{u(kT) - u[(k-1)T]}{T} \quad (5)$$

Παρά την πολυετή εφαρμογή του στα συστήματα αυτοματισμού, το PID εμφανίζει κάποια αρνητικά. Το βασικότερο είναι ότι για να μπορέσει να λειτουργήσει χρειάζεται να έχει προγραμματιστεί από κάποιον με αρκετή εμπειρία και είναι μία αρκετά χρονοβόρα διαδικασία. [47] Επίσης, είναι ιδιαίτερα επιρρεπές στις εξωτερικές συνθήκες περιβάλλοντος, ιδιαίτερα αν οι αισθητήρες μετράνε αντανάκλαση του φωτός, όπως κάποιοι υπέρυθροι (infrared) αισθητήρες.

Μια από τις εκδοχές του PID είναι το PD. Είναι μια απλοποιημένη μορφή και έχει αρκετά θετικά. Αρχικά ο κώδικας είναι απλούστερος και δεν δυσκολεύει τους φοιτητές με περαιτέρω μαθηματικές και προγραμματιστικές σκέψεις. Δεν αποκλίνει σημαντικά από την αποδοτικότητα του κλασσικού αλγορίθμου, αφού έχει σχεδιαστεί να είναι το ίδιο σταθερός. Επιπλέον, ο κλασσικός αλγόριθμος απαιτεί σε κάποιες περιπτώσεις, αρκετές ώρες για τον συντονισμό του. Αφαιρώντας μία μεταβλητή το πρόγραμμα γίνεται ταχύτερο και προσφέρει στους μαθητές περισσότερη ώρα να ασχοληθούν με άλλα σημεία του προβλήματος που μπορεί να απαιτούν μεγαλύτερη προσοχή.[49]

Το βασικό μέρος του κώδικα δίνεται παρακάτω, ενώ μεγαλύτερο μέρος του προγράμματος δίνεται σε επόμενο κεφάλαιο μαζί με τις ενδεικτικές ασκήσεις και τις λύσεις τους.

*Πίνακας 4-9: Κομμάτι κώδικα από το συνολικό πρόγραμμα για τον αλγόριθμο ακολουθίας γραμμής.*

---

```
error = analog2.read_u16() - 2100  
Pout = float((Kp / normal) * error )  
Dout = float((Kd / normal) * (error - last_error))  
steering = float(Pout + Dout)
```

---

Στο παραπάνω απόκομμα, το “error” είναι το σφάλμα μεταξύ των τιμών. Άμα είναι διαθέσιμοι δύο αισθητήρες είναι η αφαίρεση του ενός από του άλλου, ενώ άμα υπάρχει μόνο ένας είναι αφαίρεση της τιμής του αισθητήρα προς την επιθυμητή τιμή. Αυτή συνήθως είναι η μέση τιμή μεταξύ άσπρου και μαύρου, ή σε γενική περίπτωση μεταξύ των δύο χρωμάτων με βάση τα οποία κινείται το σύστημα.

Το “Pout” είναι η μεταβλητή που χρησιμοποιείται για τον υπολογισμό της παραμέτρου “P”, της αναλογικής σχέσης των δεδομένων, του PD. Η μεταβλητή “normal” είναι η κανονικοποίηση στα νούμερα, καθώς οι αισθητήρες έχουν εύρος τιμών από 0 έως 65536, κάτι το οποίο δυσχεραίνει ιδιαίτερα την διόρθωση των τιμών. Αντίστοιχα, το “Dout” είναι η μεταβλητή της παραμέτρου της παραγωγού. Τα “Kp” και “Kd” ορίζονται μετά από δοκιμές.

### **4.3. Εγκατάσταση λογισμικού**

#### **4.3.1. Στην πλακέτα**

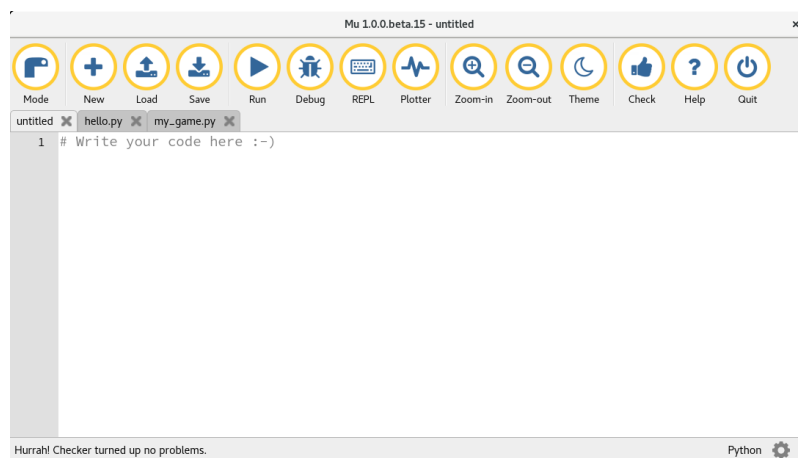
Αρχικά, ο μικροεπεξεργαστής χρειάζεται το κατάλληλο πρόγραμμα για να λειτουργήσει, ανεξάρτητα από το ποιο μοντέλο έχει επιλέγει. Ο μικροεπεξεργαστής RP2040 είναι ιδιαίτερα εύκολο να ετοιμασθεί για τον προγραμματισμό του. Μέσα στην ιστοσελίδα της εταιρείας Raspberry Pi Documentation (<https://tinypurl.com/yc7zes6r>) υπάρχει η αναλυτική διαδικασία για το πως γίνεται αυτό καθώς και το κατάλληλο πρόγραμμα. Για μεγαλύτερη ευκολία και για πληρότητα της εργασίας τα βασικά βήματα δίνονται στην συνέχεια.

Αφού έχει επιλεγθεί το σωστό πρόγραμμα, το πρώτο βήμα είναι στην πλακέτα να εντοπιστεί το κουμπί του «BOOTSEL» για την περίπτωση του Raspberry Pi Pico και συνεπώς της πλακέτας Maker

Pi Pico, ή του «BOOT» για την πλακέτα Maker Pi RP2040. Για να ξεκινήσει η διαδικασία, είναι απαραίτητο να κρατηθεί πιεσμένο, με την πλακέτα να είναι συνδεδεμένη στον υπολογιστή μέσω USB. Στην συνέχεια αφού συνδεθεί η πλακέτα στον υπολογιστή, αφήνεται το κουμπί. Η πλακέτα θα αναγνωρισθεί ως συσκευή μαζικής αποθήκευσης με όνομα RPI-RP2. Για να τελειώσει η διαδικασία, πρέπει απλά να τοποθετηθεί το αρχείο MicroPython UF2 στον φάκελο RPI-RP2. Αυτό γίνεται με το απλό «Drag and Drop». Η πλακέτα θα κάνει επανεκκίνηση και είναι πλέον έτοιμο να προγραμματιστεί.

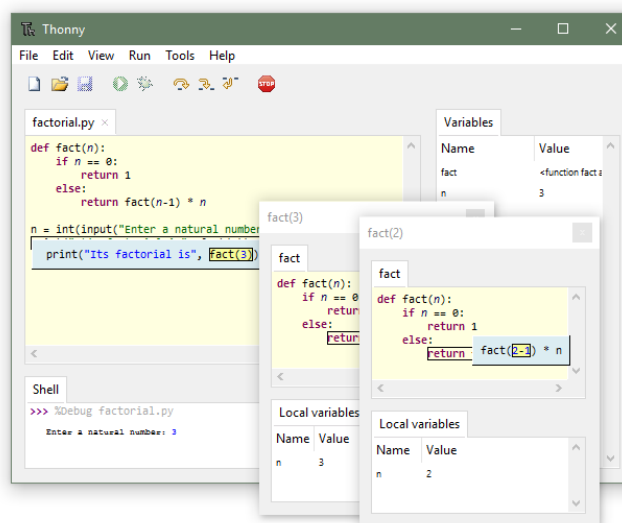
#### 4.3.2. Στον υπολογιστή

Όπως είναι λογικό για μπορέσει η διάταξη να προγραμματιστεί είναι απαραίτητο ένα κατάλληλο προγραμματιστικό περιβάλλον. Υπάρχουν πολλά προγράμματα τα οποία είναι κατάλληλα για αυτή τη δουλειά. Ωστόσο από αυτά επιλέχθηκαν το Thonny και MU Editor, τα οποία είναι προσιτά για αρχάριους. Όποιο και από τα δύο να αποφασιστεί να χρησιμοποιηθεί,



Εικόνα 4-3: Προγραμματιστικό περιβάλλον του MU Editor

Για να εγκατασταθούν, απλά ακολουθούνται οδηγίες που δίνονται στους ιστότοπους του κάθε προγραμματιστικού περιβάλλοντος. Για το Thonny Editor, οι οδηγίες βρίσκονται στον ιστότοπο: <https://thonny.org/>, ενώ για το MU Editor ο σύνδεσμος είναι: <https://codewith.mu/>. Στην συνέχεια φαίνεται σε φωτογραφίες το περιβάλλον του κάθε Editor.



Εικόνα 4-4: Προγραμματιστικό περιβάλλον του Thonny Editor

## Κεφάλαιο 5ο Ενδεικτικές ασκήσεις

Το ακαδημαϊκό ημερολόγιο στα περισσότερα πανεπιστήμια έχει διάρκεια 13 εβδομάδες. Το ίδιο ισχύει και για το Πανεπιστήμιο Δυτικής Αττικής. Σύμφωνα με αυτό τα μαθήματα οφείλουν να προσαρμοστούν ανάλογα. Στο περιεχόμενο του μαθήματος της Μηχανικής αναφέρεται ότι θα αναλυθούν κινητήρες (DC, Servo, Stepper), αισθητήρια όργανα, ημιαγωγοί και ημιαγωγικές διατάξεις και μικρο-ηλεκτρομηχανικά συστήματα. Επίσης κατά την διάρκεια των μαθημάτων αναμένεται να ασχοληθεί ο φοιτητής και με τον προγραμματισμό των μικροελεγκτών και συνεπώς των μηχανικών διατάξεων.

Σε αυτό το σημείο του προγράμματος του μαθήματος μπορεί να ενταχθεί η ρομποτική πλατφόρμα που έχει αναλυθεί σε προηγούμενα κεφάλαια. Για αρχή στα πρώτα μαθήματα προτείνονται μερικές πιο απλές ασκήσεις για να μπορέσουν οι φοιτητές να εξοικειωθούν με τις βασικές έννοιες του προγραμματισμού και με την επίδραση που έχουν οι αλλαγές στο πρόγραμμα στο φυσικό σύστημα. Οι χρόνοι που δίνονται στον πίνακα είναι σύμφωνα με τον οδηγό σπουδών όπου το εργαστηριακό μέρος του μαθήματος πραγματοποιείται για 1 ώρα την εβδομάδα.

Πίνακας 5-1: Προτεινόμενα σενάρια/ ασκήσεις για το εργαστηριακό μάθημα της Μηχανικής, μαζί με τις βιβλιοθήκες/ εντολές και την κατασκευή που χρησιμοποιούν (Ασκήσεις 1 και 2)

Τίτλος άσκησης	Περιγραφή	Προτεινόμενος χρόνος	Βιβλιοθήκες/ εντολές	Απαραίτητη κατασκευή
Εισαγωγή με χρήση LED	Ο φοιτητής μαθαίνει να χρησιμοποιεί τις βασικές εντολές για να ανάψει και να σβήσει τα λαμπάκια LED.	1 βδομάδα	machine.Pin time.sleep μεταβλητή.value() for loop	Ηλεκτρονική διάταξη για LED
Απλές Κινήσεις	Ο φοιτητής προγραμματίζει το ρομπότ να εκτελεί απλές διαδρομές στον χώρο.(1 <sup>ο</sup> μέρος) Στη συνέχεια, ο φοιτητής εισαγάγει την πολυπλοκότητα του servo και μαθαίνει να συντάσσει λειτουργίες, οι οποίες θα ανακαλεστούν στο κύριο πρόγραμμα. (2 <sup>ο</sup> μέρος)	3 βδομάδες	machine.Pin machine.PWM time.sleep for loop μεταβλητή.duty_u16(ταχ) def	2 κινητήρες κίνησης, 1 μηχανισμός με κινητήρα Servo

Πίνακας 5-2: Προτεινόμενα σενάρια/ ασκήσεις για το εργαστηριακό μάθημα της Μηχατρονικής, μαζί με τις βιβλιοθήκες/ εντολές και την κατασκευή που χρησιμοποιούν (Ασκήσεις 3, 4 και 5)- συνέχεια πίνακα 5-1

Τίτλος άσκησης	Περιγραφή	Προτεινόμενος χρόνος	Βιβλιοθήκες/ εντολές	Απαραίτητη κατασκευή
Ανάγνωση Αισθητήρων	Ο φοιτητής προγραμματίζει τη διάταξη να αναγνωρίζει τους αναλογικούς αισθητήρες και να αναγνώσει τις τιμές.	1 βδομάδα	machine.Pin machine.PWM time.sleep for loop while loop μεταβλητή.duty_u16(ταχ) def machine.ADC() analog.read_u16()	2 κινητήρες κίνησης, 1 ή 2 αισθητήρες TCRT5000 με ικανότητα ανάγνωσης του εδάφους.
Εισαγωγή/ Ρύθμιση του PD	Οι φοιτητές καλούνται να συντονίσουν τους συντελεστές του αλγορίθμου PD, για διαφορετικές διαδρομές.	3 βδομάδες		2 κινητήρες κίνησης, 1 ή 2 αισθητήρες TCRT5000
Συνολική δοκιμασία	Στους φοιτητές δίνεται μια δοκιμασία με ακολουθία διαδρομής στο έδαφος, απλών κινήσεων και χρήση του επιπλέον μηχανισμού.	5 βδομάδες		2 κινητήρες κίνησης, 1 κινητήρας servo για τον μηχανισμό, 1 ή 2 αισθητήρες TCRT5000

## 5.1. Μέρος Α: Ανάλυση ασκήσεων για τον διδάσκοντα

### Άσκηση 1

Τίτλος: Εισαγωγή με χρήση LED

Σκοπός: Ο φοιτητής εισάγεται για πρώτη φορά στο προγραμματιστικό περιβάλλον της MicroPython. Καλείται να χρησιμοποιήσει τις βασικές βιβλιοθήκες για να λειτουργήσει η πλακέτα/ ηλεκτρονική διάταξη σωστά. Αφού εισάγει τις σωστές βασικές βιβλιοθήκες, καλείται να προγραμματίσει ένα λαμπάκι LED να αναβοσβήνει για μερικά δευτερόλεπτα.

Στόχος: Ο φοιτητής να εξοικειωθεί με τις βασικές βιβλιοθήκες της MicroPython. Επίσης θα χρησιμοποιηθούν βασικές εντολές και προγραμματιστικές λογικές, όπως η δομή της επανάληψης, και θα διδαχτούν στους φοιτητές με σκοπό να τις χρησιμοποιούν, να τις αναγνωρίζουν και να τις

τροποποιούν ανάλογα με το πρόβλημα που καλούνται να λύσουν. Ο καθηγητής χρειάζεται να εξηγήσει στους φοιτητές τις βιβλιοθήκες που απαιτούνται καθώς και τις εντολές. Αναλυτικά, η χρήση της κάθε εντολής περιγράφεται στον επόμενο πίνακα

Στην περίπτωση που επιλεγεί η πλακέτα Maker Pi Pico, είναι απαραίτητο οι φοιτητές να δημιουργήσουν μία ηλεκτρονική διάταξη με λαμπάκι LED. Άμα επιλεγεί η πλακέτα Maker Pi RP2040, διαθέτει ήδη 13 λαμπάκια LED, έτοιμα προς χρήση.

Κατασκευαστική Ανάγκη: Ηλεκτρονική διάταξη για ένα ή περισσότερα LED.

Πίνακας 5-3: Βιβλιοθήκες, εντολές και η περιγραφή τους για την άσκηση 1

Βιβλιοθήκες	Εντολές	Περιγραφή Εντολής
<i>machine</i>	<code>variable = machine.Pin( pin number, machine.Pin.OUT)</code>	Ορίζεται η μεταβλητή της ακίδας ( <i>pin number</i> ) ως ψηφιακή και μπορεί να χρησιμοποιηθεί με το όνομα της μεταβλητής.
	<code>variable.value () [0: off, 1: on]</code> εναλλακτικά: <code>variable.on()</code> <code>variable.off()</code>	Χρησιμοποιείται για ανάψει ή να σβήσει το LED
<i>time</i>	<code>time.sleep(time)</code> <code>time.sleep_ms(time)</code> <code>time.sleep_us(time)</code>	Μετράει τον χρόνο σε δευτερόλεπτα, μίλι-δευτερόλεπτα (ms) και μικρο-δευτερόλεπτα (us)
<i>γενικά</i>	<code>for i in range (0, number)</code>	Επανάληψη των εντολών που βρίσκονται από κάτω τόσες φορές όσες αναγράφεται στην θέση του αριθμού <i>number</i> .

Ενδεικτική επίλυση: Στην συνέχεια δίνεται ένας κώδικας με την λύση της άσκησης, θεωρώντας ότι το λαμπάκι είναι στην ακίδα 5.

Πίνακας 5-4: Κώδικας LED- Pin 5

---

```
# Importing the necessary libraries
import machine
import time
# Initialize the variable
p5 = machine.Pin(5, machine.Pin.OUT)
for i in range(0,10):
```

---

---

```
# LED in Pin 0 is turned on for 3 seconds
p5.value(1)
time.sleep(0.1)
# LED in Pin 0 is turned off for 1 second
p5.value(0)
time.sleep(0.05)
```

---

- ❖ Αν έχει επιλεγθεί το Maker Pi RP2040, τότε ο καθηγητής μπορεί να ζητήσει από τους φοιτητές να κάνουν την ίδια άσκηση για το NeoPixel λαμπάκι, με την επιπλέον απαίτηση να αλλάξει και χρώμα μετά από κάποιες επαναλήψεις. Στον κώδικα που ακολουθεί, αλλά δίνεται στο παράρτημα, το λαμπάκι αλλάζει από κίτρινο σε μπλε μετά από 10 επαναλήψεις.

## Άσκηση 2

Τίτλος: Απλές Κινήσεις

Σκοπός: Η διάταξη είναι έτοιμη να κινηθεί. Ο φοιτητής καλείται να προγραμματίσει την διάταξη να κάνει μία εξαιρετικά απλή διαδρομή.

Στόχος: Χρησιμοποιούνται οι γνωστές βιβλιοθήκες και εισάγονται στο πρόγραμμα εντολές για την κίνηση της διάταξης. Οι εντολές αυτές αφορούν τον έλεγχο των κινητήρων DC/ Servo για την κίνηση. (1<sup>ο</sup> μέρος) Στην συνέχεια γίνεται εισαγωγή της προγραμματιστικής έννοιας «def» και προγραμματισμός του ρομπότ με την χρήση αυτών των συναρτήσεων. Ιδανικά η διάταξη θα έχει 3 κινητήρες, 2 για την κίνηση και 1 για κάποιον μηχανισμό. (2<sup>ο</sup> μέρος)

Σε κάθε περίπτωση είναι απαραίτητο να είναι γνωστή η ακίδα που είναι συνδεδεμένος ο κινητήρας καθώς και η φορά με την οποία κινείται. Η προτεινόμενη διαδρομή είναι:

*κίνηση μπροστά για 3 δευτερόλεπτα → κίνηση πίσω για 2 δευτερόλεπτα →  
δεξιά στροφή → κίνηση πίσω για 2 δευτερόλεπτα → αριστερή στροφή →  
κίνηση μπροστα για 2 δευτερόλεπτα (1<sup>ο</sup> μέρος)*

Με την πολυπλοκότητα που έχει πλέον το πρόγραμμα, σημαντικό είναι να υπάρχουν ξεχωριστά και κατανοητά ονόματα μεταβλητών, καθώς το πλήθος τους αυξάνεται. Η προτεινόμενη διαδρομή είναι για το 2<sup>ο</sup> μισό της άσκησης είναι:

*κίνηση μπροστά για 3 δευτερόλεπτα → κίνηση πίσω για 2 δευτερόλεπτα →  
χρήση του τρίτου κινητήρα προς τη μία φορά → δεξιά στροφή →  
κίνηση πίσω για 2 δευτερόλεπτα → χρήση του τρίτου κινητήρα προς την άλλη φορά →  
αριστερή στροφή → κίνηση μπροστα για 2 δευτερόλεπτα →  
χρήση του τρίτου κινητήρα με την μισή κίνηση (2<sup>ο</sup> μέρος)*

Κατασκευαστική Ανάγκη: 2 κινητήρες κίνησης, 1 μηχανισμός με κινητήρα Servo



Πίνακας 5-5: Βιβλιοθήκες, εντολές και η περιγραφή τους για την άσκηση 2

Βιβλιοθήκες	Εντολές	Περιγραφή Εντολής
<i>machine</i>	<code>variable = machine.PWM(machine.Pin(pin number))</code>	Ορίζεται η μεταβλητή της ακίδας ( <i>pin number</i> ) ως ψηφιακή PWM (pulse width modulation) και μπορεί να χρησιμοποιηθεί με το όνομα της μεταβλητής.
	<code>variable.freq(number)</code>	Χρησιμοποιείται για οριστεί η συχνότητα του PWM. Μετά από δοκιμές αποφασίστηκε ότι η καταλληλότερες συχνότητες είναι μεταξύ 50 και 100 Hz, ιδανική την 80 Hz.
	<code>variable.duty_u16(speed)</code>	Χρησιμοποιείται για να δοθεί η ταχύτητα στον κινητήρα. Έχει εύρος τιμών από 0 έως 65535.
	<code>for i in range(0, number)</code>	Επανάληψη των εντολών που βρίσκονται από κάτω τόσες φορές όσες αναγράφεται στην θέση του αριθμού <i>number</i> .
	<code>print()</code>	Βγάζει στην οθόνη του υπολογιστή αυτό που βρίσκεται στην παρένθεση.
<i>γενικά</i>	<code>def name(variable)</code>	Είναι μία συνάρτηση που ορίζεται στην αρχή του προγράμματος. Ονομάζεται ανάλογα την λειτουργία της και η μεταβλητή αλλάζει ανάλογα την τιμή της κατά την διάρκεια του προγράμματος.

Ενδεικτική επίλυση -1<sup>ο</sup> μέρος:

Για έναν κινητήρα.

*Πίνακας 5-6 Κώδικας για έναν κινητήρα - Pins 8 and 9 (1ο μέρος)*

---

```
# Importing necessary libraries
import machine
import time

# Setup DC Motor pins
M1A = machine.PWM(machine.Pin(8)) # Pin forward
M1B = machine.PWM(machine.Pin(9)) # Pin backward

M1A.freq(80) # Desired frequency
M1B.freq(80)

# Speed range
max_speed = 65535
min_speed = 15000

print("forward")
M1A.duty_u16(min_speed) # Duty Cycle must be between 0 until 65535
M1B.duty_u16(0)
time.sleep(2)

print("stop")
M1A.duty_u16(0) # Duty Cycle must be between 0 until 65535
M1B.duty_u16(0)
time.sleep(2)
```

---

Ενδεικτική επίλυση- 2<sup>ο</sup> μέρος:

*Πίνακας 5-7: Κώδικας με χρήση def (2ο μέρος)*

---

```
# Code with def

# Importing libraries
import machine
```

---

---

```
import time

# Setup DC Motor pins
M1A = machine.PWM(machine.Pin(9))
M1B = machine.PWM(machine.Pin(8))
M1A.freq(50)
M1B.freq(50)

# Speed range
max_speed = 65535
min_speed = 15000
stop = 0

# functions for def
def forward(speed, time):
    M1A.duty_u16(0)
    M1B.duty_u16(speed)
    time.sleep(time)

def backward(speed, time):
    M1A.duty_u16(0)
    M1B.duty_u16(speed)
    time.sleep(time)

def right_spin(speed, time):
    M1B.duty_u16(0)
    M1A.duty_u16(speed)
    time.sleep(time)

# Main code
# Move forward for 2 seconds
print("forward")
forward(min_speed, 2)
print(" stop")
forward(stop,1)
```

---

### Άσκηση 3

Τίτλος: Εισαγωγή στους αισθητήρες

Σκοπός: Ο φοιτητής καλείται να αναγνωρίσει τους αναλογικούς αισθητήρες και να τους προγραμματίσει ώστε να αναγνώσει τις τιμές τους.

Στόχος: Αρχικά θα χρησιμοποιήσει τις κατάλληλες εντολές για να αναγνώσει τον/ους αναλογικό/ους αισθητήρα/ες. Αφού διαβάσει τις τιμές του/ων αισθητήρα/ων καλείται να προγραμματιστεί τη διάταξη να κινηθεί μπροστά έως ότου εντοπίσει το κατάλληλο σημείο για να σταματήσει όπως υποδεικνύεται από κάποιο σημείο στο έδαφος (πχ. Μαύρο ίχνος)

Ολοκληρώνεται η κατασκευή και γίνεται η ανάγνωση των τιμών με τον/ους αισθητήρα/ες TCRT5000, σύμφωνα με το πρόγραμμα παράδειγμα. Εν συνεχεία, η ρομποτική διάταξη προγραμματίζεται να προχωρήσει μπροστά μέχρι να αναγνωρίσει το μαύρο ίχνος στο έδαφος.

*κίνηση μπροστά εως ο/οι αισθητήρας/ες να αναγνωρίσουν την σωστή τιμή*

Κατασκευαστική Ανάγκη: 2 κινητήρες κίνησης, 1 ή 2 αισθητήρες TCRT5000 με ικανότητα ανάγνωσης του εδάφους.

*Πίνακας 5-8: Βιβλιοθήκες, εντολές και η περιγραφή τους για την άσκηση 3*

<b>Βιβλιοθήκες</b>	<b>Εντολές</b>	<b>Περιγραφή Εντολής</b>
<i>machine</i>	<code>variable = machine.ADC(pin number)</code>	Ορίζεται η μεταβλητή της ακίδας ( <i>pin number</i> ) από αναλογική σε ψηφιακή και μπορεί να χρησιμοποιηθεί με το όνομα της μεταβλητής.
	<code>variable.read_u16()</code>	Χρησιμοποιείται για να διαβάσει το πρόγραμμα την τιμή του αισθητήρα
<i>γενικά</i>	<code>while condition True/ False</code>	Επανάληψη των εντολών που βρίσκονται από κάτω τόσες φορές όσες χρειάζεται μέχρι να ικανοποιηθεί η συνθήκη.

Ενδεικτική επίλυση:

*Πίνακας 5-9: Κώδικας για την ανάγνωση αναλογικού αισθητήρα*

---

```
# Analog Sensors read
```

```
# Importing necessary libraries
```

```
import machine
```

```
import time
```

---

---

```
# Setting up the variables
analog1 = machine.ADC(0)

# Reading the sensor value for 3 seconds
print("1: ", analog1.read_u16())
time.sleep(3)
```

---

Η ενδεικτική επίλυση για την αναγνώριση μαύρου ίχνους δίνεται στο παράρτημα.

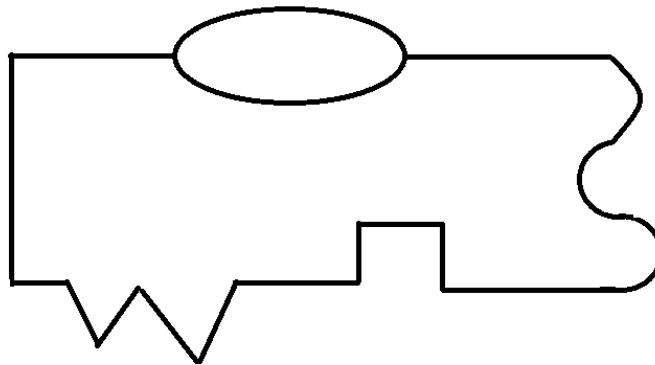
#### Άσκηση 4

Τίτλος: Εισαγωγή/ Ρύθμιση του PD

Σκοπός: . Αρκετά ρομποτικά συστήματα χρησιμοποιούν κάποιου είδους αλγόριθμο για την λειτουργία τους και ένας από τους πιο γνωστούς είναι ο αλγόριθμός του PID. Ωστόσο, αλγόριθμοί που αναπτύχθηκαν μετά τον PID έχουν αποδείξει ότι έχουν καλύτερα αποτελέσματα. Ένας από αυτούς είναι ο αλγόριθμος του PD, τον οποίο οι φοιτητές καλούνται κατανοήσουν και να ρυθμίσουν.

Στόχος: Στο έδαφος έχει διαμορφωθεί μια απλή κυκλική διαδρομή με ευθείες και τετθλασμένες γραμμές. Η ρομποτική διάταξη προγραμματίζεται να μπορεί να ακολουθήσει την διαδρομή

Η ενδεικτική διαδρομή θα μπορούσε να είναι της μορφής:



Κατασκευαστική ανάγκη: 2 κινητήρες και 1 ή 2 αισθητήρες TCRT5000

*Εικόνα 5-1: Ενδεικτική τροχιά μαύρου ίχνους*

Ενδεικτική επίλυση:

*Πίνακας 5-10: Βασικός κώδικας για το αλγόριθμο PD*

---

```
# PD variables
Kp = 2
Kd = 2
```

---

---

```
# initial variables
error = 0
last_error = 0

# MAIN PROGRAM
while True:
    error = analog2.read_u16() - 2100
    Pout = float((Kp) * error )
    Dout = float((Kd) * (error - last_error))
    steering = float(Pout + Dout)
    print("1: ", analog1.read_u16())
    print("2: ", analog2.read_u16())
    print("steering: ", steering)

    # straight
    if steering == 0:
        back1A.duty_u16(0)
        front1B.duty_u16(speed1)
        back2A.duty_u16(0)
        front2B.duty_u16(speed2)

    # left
    if steering < 0:
        back1A.duty_u16(0)
        front1B.duty_u16(speed1 + steering)
        back2A.duty_u16(speed2 + steering)
        front2B.duty_u16(0)

    # right
    if steering > 0:
        back1A.duty_u16(speed1 + steering)
        front1B.duty_u16(0)
        back2A.duty_u16(0)
        front2B.duty_u16(speed2 + steering)
    last_error = error
    time.sleep(0.001)
```

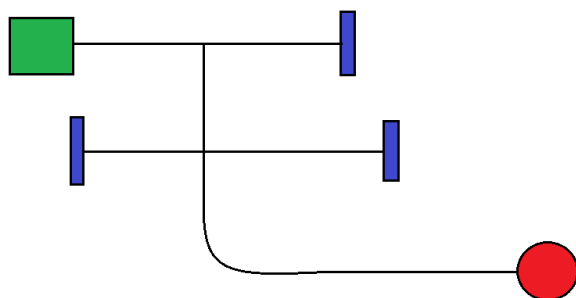
---

## Άσκηση 5

Τίτλος: Συνολική Δοκιμασία

Σκοπός/ Στόχος: Με βάση όλες τις προηγούμενες ασκήσεις, ο φοιτητής καλείται να επιλέξει την κατασκευή και να την προγραμματίσει με σκοπό να ολοκληρώσει μία πολύπλοκη διαδρομή. Αυτή περιλαμβάνει ακολουθία γραμμής

Πρόβλημα προς επίλυση: Η προτεινόμενη δοκιμασία ακολουθεί στο επόμενο σχήμα. Η διαδρομή ξεκινάει από το πράσινο τετράγωνο και καταλήγει στο κόκκινο. Στα ενδιάμεσα μπλε παραλληλόγραμμα βρίσκονται πιθανώς αντικείμενα τα οποία η διάταξη καλείται να μετακινήσει από τη θέση τους.



Εικόνα 5-2: Παράδειγμα συνολικής δοκιμασίας

Ενδεικτική επίλυση: Με βάση τους παραπάνω κώδικες, το πρόγραμμα αυτής της άσκησης είναι ο συνδυασμός των προηγούμενων.

### **5.2. Μέρος Β: Φύλλα εργασίας για τους φοιτητές**

Πριν δοθούν στους φοιτητές οι παρακάτω ασκήσεις πρέπει να είναι έτοιμη η κατασκευή της ρομποτικής διάταξης, να έχει γίνει η επιλογή πλακέτας και συνεπώς οι κατάλληλες ηλεκτρονικές διατάξεις. Ακόμα είναι απαραίτητη η εγκατάσταση κάποιου από τα δύο προγραμματιστικά περιβάλλοντα, όπως αναφέρονται στην ενότητα 4.3 Εγκατάσταση λογισμικού, στο Κεφάλαιο 4<sup>ο</sup>: Προγραμματισμός. Οι ασκήσεις αυτές μπορούν να υλοποιηθούν ατομικά ή ομαδικά, έως 2 άτομα. Κάθε ομάδα πρέπει να έχει τη δική της ρομποτική πλατφόρμα η οποία έχει κοινά χαρακτηριστικά για όλες τις ομάδες.

Οι παρακάτω ασκήσεις έχουν γίνει με βάση την προτεινόμενη κατασκευή, δηλαδή με δύο κινητήρες DC, έναν μηχανισμό gripper με κινητήρα Servo, δύο αισθητήρες TCRT5000 και για πλακέτα επιλέχθηκε η Maker Pi RP2040. Επομένως, με αυτή την εξήγηση ακολουθούν τα κατάλληλα φύλλα έργου για φοιτητές για τις ασκήσεις 1 έως 3.

Είναι σημαντικό να σημειωθεί ότι για κάθε άσκηση **δεν υπάρχει μία μοναδική λύση**. Για να μπορούν να χρησιμοποιηθούν τα φύλλα έργου δίνονται στο παράρτημα Ε της εργασίας.

## Κεφάλαιο 6ο      Συμπεράσματα

Με την ολοκλήρωση της εργασίας, είναι απαραίτητο να γίνει μια συνολική αξιολόγηση για την μικρορομποτική διάταξη. Η αξιολόγηση αυτή γίνεται για τα σημεία στα οποία έχει επικεντρωθεί η εργασία. Συνολικά η πολυμορφική μικρορομποτική πλατφόρμα ήταν ένα δύσκολο εγχείρημα να επιτευχθεί. Υπάρχουν αρκετοί περιορισμοί και παράμετροι που πρέπει να ληφθούν υπόψιν πριν τον σχεδιασμό και την υλοποίησή της.

Σε κατασκευαστικό επίπεδο έπρεπε να γίνουν κάποιες παραδοχές με βάση τις διαθέσιμες πλατφόρμες του εμπορίου την τρέχουσα περίοδο. Αυτό σημαίνει ότι σε λίγα χρόνια θα χρειαστεί ξανά αναβάθμιση. Από την άλλη, δημιουργήθηκε μια σύγχρονη πολυμορφική πλατφόρμα με ικανότητα να δεχτεί δύο ειδή κινητήρων στην βασική κίνηση, αλλά και αρκετούς επιπλέον μηχανισμούς. Τα βασικά σχέδια είναι σχετικά απλά ώστε να μην υπάρξουν αυστηροί περιορισμοί στο υλικό από το οποίο θα φτιαχτεί. Επίσης είναι συναρμολογούμενη και συμβατή με αρκετά υλικά του εμπορίου, κάτι που καθιστά την συντήρηση εύκολη και οικονομική.

Οι ηλεκτρονικές διατάξεις που επιλέχθηκαν είναι σύγχρονες και χρησιμοποιούν νέα υλικά όπως ο επεξεργαστής Raspberry RP2040. Σε καμία από τις δύο προτεινόμενες ηλεκτρονικές πλακέτες δεν υπάρχει πλήρη επίδραση από τον χειριστή, καθώς είναι έτοιμες πλακέτες. Ο βασικός λόγος που επιλέχθηκαν είναι το μικρό κόστος αγοράς και το μικρό μέγεθος τους. Σε αντίθεση με μία πλήρως ανοιχτή ηλεκτρονική διάταξη, που συνήθως περιλαμβάνει breadboard, αντιστάσεις, ποτενσιόμετρα και άλλα υλικά, οι πλακέτες αυτές είναι ικανές να εκτελέσουν όλες τις βασικές λειτουργίες σε πολύ μικρότερο μέγεθος.

Το κομμάτι του προγραμματισμού είναι μια ιδιαίτερη διαδικασία. Επιλέχθηκε να αναλυθεί μια νέα προγραμματιστική γλώσσα, η MicroPython. Αυτή προσφέρει πολλά θετικά σε αντίθεση με άλλες γλώσσες, καθώς η συγγραφή του κώδικα δεν χρειάζεται να ακολουθεί αυστηρές προϋποθέσεις. Για έναν αρχάριο, είναι ένα φιλικό περιβάλλον με άμεση εισαγωγή στην λογική του προγραμματισμού και για έναν έμπειρο προγραμματιστή είναι μια γρηγορότερη λύση. Σε συνδυασμό με τις επιλεγμένες πλακέτες ο προγραμματισμός της μικρορομποτικής διάταξης γίνεται πιο απλός και αυξάνονται οι εφαρμογές που μπορεί να εκτελέσει.

Εξίσου σημαντικό είναι το γεγονός ότι αυτή η πλατφόρμα σχεδιάστηκε με σκοπό την εκπαιδευτική διαδικασία. Το μάθημα της Μηχατρονικής έχει σκοπό να διδάξει τους φοιτητές την σημασία των διαστρωμάτων ενός συστήματος. Η πολυμορφία δίνει την δυνατότητα στην κατασκευή να εξερευνηθούν διαφορετικοί τύποι ρομπότ, ενώ οι ηλεκτρονικές πλακέτες διευκολύνουν τον φοιτητή στην μαθησιακή διαδικασία.

Συνεπώς, με την ολοκλήρωση της εργασίας η πολυμορφική μικρορομποτική διάταξη που σχεδιάστηκε είναι ικανή να καλύψει τις ανάγκες του εργαστηριακού μέρους του μαθήματος. Βέβαια, ένα ακόμα στοιχείο της πλατφόρμας είναι η δυνατότητα επέκτασης που προσφέρει η απλή κατασκευή της. Ένα χαρακτηριστικό που θα μπορούσε να βελτιωθεί είναι η ηλεκτρονική διάταξη, η οποία θα μπορούσε να αντικατασταθεί με breadboard. Ακόμα, θα μπορούσε να εξελιχθεί περισσότερο ο προγραμματισμός με την χρήση επιπλέον αισθητήρων και μηχανισμών, αυξάνοντας έτσι την πολυπλοκότητα των εργασιών που είναι ικανή να εκτελέσει η πλατφόρμα.



## Βιβλιογραφία

- [1] Elsevier, “Mechatronics Overview.” <https://www.sciencedirect.com/topics/engineering/mechatronics> (accessed Apr. 28, 2022).
- [2] U. S. Dixit, M. Hazarika, and J. P. Davim, *History of Mechatronics*. 2017.
- [3] Πανεπιστήμιο Δυτικής Αττικής, “Μηχατρονική- Μηχανολόγων Μηχανικών.” <https://mech.uniwa.gr/course/michatroniki/> (accessed Apr. 27, 2022).
- [4] Πανεπιστήμιο Δυτικής Αττικής, “Μηχατρονική- Ηλεκτρολόγων & Ηλεκτρονικών Μηχανικών.” [https://eee.uniwa.gr/documents/program/perigrammata/ex\\_7\\_9\\_g\\_kiklos/09\\_Γ/EEE.9-3.6\\_Μηχατρονική.pdf](https://eee.uniwa.gr/documents/program/perigrammata/ex_7_9_g_kiklos/09_Γ/EEE.9-3.6_Μηχατρονική.pdf) (accessed Apr. 27, 2022).
- [5] Πανεπιστήμιο Δυτικής Αττικής, “Μηχατρονική- Μηχανικών Πληροφορικής & Υπολογιστών.” <http://www.ice.uniwa.gr/education/undergraduate/courses/introduction-to-mechatronics/> (accessed Apr. 27, 2022).
- [6] Πανεπιστήμιο Δυτικής Αττικής, “Μηχατρονική- Μηχανικών Βιομηχανικής Σχεδίασης & Παραγωγής.” [https://idpe.uniwa.gr/images/2022\\_GR-Syllabus/7001\\_GR.pdf](https://idpe.uniwa.gr/images/2022_GR-Syllabus/7001_GR.pdf) (accessed Apr. 27, 2022).
- [7] H. K. Kondaveeti, N. K. Kumaravelu, S. D. Vanambathina, S. E. Mathe, and S. Vappangi, “A systematic literature review on prototyping with Arduino: Applications, challenges, advantages, and limitations,” *Comput. Sci. Rev.*, vol. 40, p. 100364, 2021, doi: 10.1016/j.cosrev.2021.100364.
- [8] P. Reguera, D. García, M. Domínguez, M. A. Prada, and S. Alonso, “A Low-cost Open Source Hardware in Control Education. Case Study: Arduino-Feedback Ms-150,” *IFAC-PapersOnLine*, vol. 48, no. 29, pp. 117–122, 2015, doi: 10.1016/j.ifacol.2015.11.223.
- [9] A. Chatzopoulos, M. Kalogiannakis, S. Papadakis, and M. Papoutsidakis, “A Novel, Modular Robot for Educational Robotics Developed Using Action Research Evaluated on Technology Acceptance Model,” *Educ. Sci.*, vol. 12, no. 4, p. 274, 2022, doi: 10.3390/educsci12040274.
- [10] R. Bannatyne and G. Viot, “Introduction to microcontrollers,” *Wescon Conf. Rec.*, pp. 564–574, 1997, doi: 10.1201/9781420077681.ch1.
- [11] Raspberry Pi Foundation, “Raspberry Pi OS.” <https://www.raspberrypi.com/software/> (accessed Apr. 29, 2022).
- [12] Arduino, “Arduino.” <https://store.arduino.cc/products/arduino-uno-rev3> (accessed Apr. 29, 2022).

- [13] Raspberry Pi Foundation, “Raspberry Pi.” <https://www.raspberrypi.org/> (accessed Apr. 29, 2022).
- [14] Espressif, “ESP32-S3.” [https://www.espressif.com/sites/default/files/documentation/esp32-s3\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32-s3_datasheet_en.pdf) (accessed Apr. 29, 2022).
- [15] A. Savini and G. G. Savini, “A short history of 3D printing, a technological revolution just started,” *Proc. 2015 ICOHTEC/IEEE Int. Hist. High-Technologies their Socio-Cultural Context. Conf. HISTELCON 2015 4th IEEE Reg. 8 Conf. Hist. Electrotechnol.*, 2015, doi: 10.1109/HISTELCON.2015.7307314.
- [16] M. K. Thompson *et al.*, “Design for Additive Manufacturing: Trends, opportunities, considerations, and constraints,” *CIRP Ann. - Manuf. Technol.*, vol. 65, no. 2, pp. 737–760, 2016, doi: 10.1016/j.cirp.2016.05.004.
- [17] B. Vayre, F. Vignat, and F. Villeneuve, “Designing for additive manufacturing,” in *Procedia CIRP*, 2012, vol. 3, no. 1, pp. 632–637, doi: 10.1016/j.procir.2012.07.108.
- [18] J. Szulżyk-Cieplak, A. Duda, and B. Sidor, “3D Printers – New Possibilities in Education,” *Adv. Sci. Technol. Res. J.*, vol. 8, no. 24, pp. 96–101, 2014, doi: 10.12913/22998624/575.
- [19] M. Kumke, H. Watschke, P. Hartogh, A. K. Bavendiek, and T. Vietor, “Methods and tools for identifying and leveraging additive manufacturing design potentials,” *Int. J. Interact. Des. Manuf.*, vol. 12, no. 2, pp. 481–493, 2018, doi: 10.1007/s12008-017-0399-7.
- [20] E. Arvanitidi, C. Drosos, E. Theocharis, and M. Papoutsidakis, “3D Printing and Education,” *Int. J. Comput. Appl.*, vol. 177, no. 24, pp. 55–59, 2019, doi: 10.5120/ijca2019919711.
- [21] D. Liliana and P. S. Florina, “Education, Knowledge and Innovation from a Mechatronics Perspective,” *Procedia - Soc. Behav. Sci.*, vol. 203, pp. 205–209, 2015, doi: 10.1016/j.sbspro.2015.08.283.
- [22] D. Assante, G. M. Cennamo, and L. Placidi, “3D printing in education: An european perspective,” *IEEE Glob. Eng. Educ. Conf. EDUCON*, vol. 2020-April, pp. 1133–1138, 2020, doi: 10.1109/EDUCON45650.2020.9125311.
- [23] M. Á. Caminero, J. M. Chacón, E. García-Plaza, P. J. Núñez, J. M. Reverte, and J. P. Becar, “Additive manufacturing of PLA-based composites using fused filament fabrication: Effect of graphene nanoplatelet reinforcement on mechanical properties, dimensional accuracy and texture,” *Polymers (Basel)*, vol. 11, no. 5, 2019, doi: 10.3390/polym11050799.
- [24] S. A. Raj, E. Muthukumaran, and K. Jayakrishna, “A Case Study of 3D Printed PLA and Its Mechanical Properties,” *Mater. Today Proc.*, vol. 5, no. 5, pp. 11219–11226, 2018, doi: 10.1016/j.matpr.2018.01.146.

- [25] Y. Koren and S. Malkin, “Torque and Speed Control of DC-Servomotors for Robots,” *CIRP Ann. - Manuf. Technol.*, vol. 33, no. 1, pp. 239–242, 1984, doi: 10.1016/S0007-8506(07)61417-6.
- [26] Creality, “Creality CR series.” [https://www.creality.com/pages/products?collection=CR+Series&spm=.index.header\\_1.1](https://www.creality.com/pages/products?collection=CR+Series&spm=.index.header_1.1) (accessed Feb. 17, 2023).
- [27] Cytron Marketplace, “Cytron,” *Cytron*, 2023. <https://www.cytron.io/> (accessed Jan. 08, 2023).
- [28] C. Bell, *Beginning MicroPython with the Raspberry Pi Pico*. 2022.
- [29] S. Yamanoor and N. S. Yamanoor, “High Quality, Low Cost Education with the Raspberry Pi,” no. 46, pp. 3–7, 2016.
- [30] D. R. Loker, “Embedded Systems using the Raspberry Pi Pico,” *ASEE Annu. Conf. Expo. Conf. Proc.*, 2022.
- [31] B. Balon and M. Simic, “Using Raspberry Pi Computers in Education,” *2019 42nd Int. Conv. Inf. Commun. Technol. Electron. Microelectron.*, no. March 2018, pp. 671–676, 2019, doi: 10.23919/mipro.2019.8756967.
- [32] Raspberry Pi Ltd, “RP2040 Datasheet A microcontroller by Raspberry Pi.” 2022, [Online]. Available: <https://datasheets.raspberrypi.com/rp2040/rp2040-datasheet.pdf>.
- [33] Cytron Marketplace, “Maker Pi RP2040 : Simplifying Robotics with Raspberry Pi® RP2040,” *Cytron*. <https://www.cytron.io/p-maker-pi-rp2040-simplifying-robotics-with-raspberry-pi-rp2040> (accessed Jan. 26, 2023).
- [34] J. Fraden, “Chapter 1: Data Acquisition,” in *Handbook of Modern Sensors: Physics, Designs and Application*, 4th ed., Springer New York Heidelberg Dordrecht London, 2010, p. 681.
- [35] Vishay Semiconductors, “Application of Optical Reflex Sensors Vishay Semiconductors Optoelectronic Sensors,” no. 80107, pp. 1–17.
- [36] NASA, “Visible Light,” *NASA science*. [https://science.nasa.gov/ems/09\\_visiblelight](https://science.nasa.gov/ems/09_visiblelight) (accessed Jan. 26, 2023).
- [37] V. Semiconductors, “Vishay Semiconductors BLOCKING FILTER PARAMETER SYMBOL Vishay Semiconductors Reflective Optical Sensor with PARAMETER PARAMETER,” *Pulse*, pp. 1–6, 2002.
- [38] I. Daniyan, K. Mpofo, B. Ramatsetse, E. Zeferino, G. Monzambe, and E. Sekano, “Design and simulation of a flexible manufacturing system for manufacturing operations of railcar subassemblies,” *Procedia Manuf.*, vol. 54, pp. 112–117, 2020, doi:

- 10.1016/j.promfg.2021.07.018.
- [39] M. Elsayed, M. Sokar, S. Abd Rabbo, and M. El-Arabi, “Wheeled Mobile Robot Control Using Advanced Fuzzy Control,” *Int. Conf. Appl. Mech. Mech. Eng.*, vol. 17, no. 17, pp. 1–15, 2016, doi: 10.21608/amme.2016.35372.
- [40] D. George and et al, “MicroPython,” *Python*. <http://micropython.org/> (accessed Nov. 25, 2022).
- [41] Y. He and L. Liang, “Application of robotics in higher education in industry 4.0 era,” *Univers. J. Educ. Res.*, vol. 7, no. 7, pp. 1612–1622, 2019, doi: 10.13189/ujer.2019.070715.
- [42] A. Tsai and F. Shiwa, “Lab Prototype Board for the Pi Pico RP2040,” no. January, 2021, [Online]. Available: [https://vanhunteradams.com/6930/Tsai\\_Shiwa.pdf](https://vanhunteradams.com/6930/Tsai_Shiwa.pdf).
- [43] R. Fraanje, T. Koreneef, A. Le Mair, and S. De Jong, “Python in robotics and mechatronics education,” *2016 11th Fr. 9th Eur. Congr. Mechatronics, MECATRONICS 2016 / 17th Int. Conf. Res. Educ. Mechatronics, REM 2016*, pp. 14–19, 2016, doi: 10.1109/MECATRONICS.2016.7547108.
- [44] M. Khamphroo, N. Kwankeo, K. Kaemarungsi, and K. Fukawa, “2017 8th International Conference of Information and Communication Technology for Embedded Systems (IC-ICTES) : proceedings : May 7-9, 2017, Bangsaen Heritage Hotel, Chonburi Beach, Thailand,” 2017.
- [45] A. Zulu, “Towards explicit PID control tuning using machine learning,” *2017 IEEE AFRICON Sci. Technol. Innov. Africa, AFRICON 2017*, pp. 430–433, 2017, doi: 10.1109/AFRCON.2017.8095520.
- [46] Nikunj Mehta, Dharmendra Chauhan, Sagar Patel, and Siddharth Mistry, “Design of HMI Based on PID Control of Temperature,” *Int. J. Eng. Res.*, vol. V6, no. 05, 2017, doi: 10.17577/ijertv6is050074.
- [47] C. T. Chao, N. Sutarna, J. S. Chiou, and C. J. Wang, “Equivalence between fuzzy PID controllers and conventional PID controllers,” *Appl. Sci.*, vol. 7, no. 6, 2017, doi: 10.3390/app7060513.
- [48] B. M. Mohan and A. Sinha, “Analytical structure and stability analysis of a fuzzy PID controller,” *Appl. Soft Comput. J.*, vol. 8, no. 1, pp. 749–758, 2008, doi: 10.1016/j.asoc.2007.06.003.
- [49] I. Brand, J. Roy, A. Ray, J. Oberlin, and S. Oberlix, “PiDrone: An Autonomous Educational Drone Using Raspberry Pi and Python,” *IEEE Int. Conf. Intell. Robot. Syst.*, pp. 5697–5703, 2018, doi: 10.1109/IROS.2018.8593943.
- [50] LEGO, “LEGO education SPIKE Prime Set.” <https://education.lego.com/en-us/products/lego-education-spike-prime-set/45678#spike™-prime> (accessed Feb. 16, 2023).

- [51] GROBOTRONICS, “Waveshare RP2040 Mini,” *GROBOTRONICS*.  
[https://grobotronics.com/waveshare-rp2040-mini.html#group\\_1641163d24899a4fd4-2](https://grobotronics.com/waveshare-rp2040-mini.html#group_1641163d24899a4fd4-2)  
 (accessed Jan. 26, 2023).
- [52] GROBOTRONICS, “Beetle RP2040 Mini Development Board,” *GROBOTRONICS*.  
<https://grobotronics.com/beetle-rp2040-mini-development-board.html> (accessed Jan. 26, 2023).
- [53] Adafruit, “Adafruit KB2040 - RP2040 Kee Boar Driver,” *Adafruit*.  
<https://www.adafruit.com/product/5302> (accessed Jan. 26, 2023).
- [54] SparkFun, “SparkFun Pro Micro - RP2040,” *SparkFun*.  
<https://www.sparkfun.com/products/18288> (accessed Jan. 26, 2023).
- [55] Adafruit, “Maker Pi Pico Base,” *Adafruit*. <https://www.adafruit.com/product/5160> (accessed Jan. 26, 2023).
- [56] J. S. Wilson, “Intro to Sensors Overview,” 2008, [Online]. Available:  
[https://books.google.de/books?hl=en&lr=&id=5UE6YCjDG-MC&oi=fnd&pg=PP1&dq=introduction+to+sensor+technology&ots=jfGSg8gGLV&sig=bVPdfgAc04QyWenTwCtQyy8588o&redir\\_esc=y#v=onepage&q=introduction+to+sensor+technology&f=false](https://books.google.de/books?hl=en&lr=&id=5UE6YCjDG-MC&oi=fnd&pg=PP1&dq=introduction+to+sensor+technology&ots=jfGSg8gGLV&sig=bVPdfgAc04QyWenTwCtQyy8588o&redir_esc=y#v=onepage&q=introduction+to+sensor+technology&f=false).
- [57] GROBOTRONICS, “Infrared Sensor TCRT5000,” *GROBOTRONICS*.  
<https://grobotronics.com/infrared-sensor-tcrt5000.html?sl=en> (accessed Jan. 26, 2023).
- [58] DiyIoT, “TCRT5000 Line Tracking Module for Arduino, ESP8266 and ESP32,” *DiyIoT*.  
<https://diyi0t.com/tcrt5000-line-tracking-module-arduino-esp8266-esp32/> (accessed Jan. 26, 2023).
- [59] 14CORE, “Wiring Photo Interrupter Light Blocking Switch with Raspberry Pi,” *14CORE*.  
<https://www.14core.com/wiring-photo-interrupter-light-blocking-switch-with-raspberry-pi/>  
 (accessed Jan. 26, 2023).
- [60] ROHM, “What is a Photointerrupter?,” *ROHM Semiconductors*.  
<https://www.rohm.com/electronics-basics/photointerrupters/what-is-a-photointerrupter>  
 (accessed Jan. 26, 2023).
- [61] GROBOTRONICS, “Waveshare Photo Interrupter Sensor,” *GROBOTRONICS*.  
<https://grobotronics.com/waveshare-photo-interrupter-sensor.html?sl=en> (accessed Jan. 26, 2023).

## Παράρτημα Α: Αναλυτικοί Πίνακες εμπορικών ρομποτικών διατάξεων

Αναλυτική Περιγραφή και Εκτιμώμενη Τιμή και Χρόνος Ζωής

Μοντέλο	Περιέχονται	Περιφερειακά	Πλακέτα	Συντήρηση	τιμή (περίπου)	Εκτιμώμενος χρόνος ζωής (χρόνια)
Actobotics kit-ActoBitty 2	1x Aluminum Channel - 4.50" (585444)	4x AA Batteries (1.25V)	Arduino	2x Micro Gearmotor - 90 RPM	5	10
	2x Micro Gearmotor - 90 RPM	1x Battery charger		4x AA Batteries (1.25V)	10	4--6
	2x Micro Gearmotor - Enclosure	1x Arduino Uno		1x Arduino Uno	25	
	2x Hub - Set Screw (3mm Bore) (545348)	1x motor driver (L239D OR L298N)		1x motor driver (L239D OR L298N)	5	
	2x Precision Disc Wheel - 2" (Clear) (595715)	2x IR line tracking sensors		2x IR line tracking sensors	7	3--5
	2x Rubber Ring - 1.65"ID x 1/8"W	1x Ultrasonic sensor		1x Ultrasonic sensor	4	2
	12x Machine Screw - Socket Head (6-32 ; 3/16")	2x Encoder Sensor		2x Encoder Sensor	6	3
	2x Actobotics Channel Mount - Arduino (585008)			2x Precision Disc Wheel - 2" (Clear) (595715)	5	
	2x Acetal Battery Mount					
	1x Acetal Castor/Sensor Mount					

	1x Acetal Castor						
	1x 4 Cell AA Battery Pack						
	2x Phillips head Screw - 2-32 x 3/16"						
	1x 1/16" Hex Key						
	1x 7/64" Hex Key						
Mini round robot Chassis Kit- 2WD w DC Motors	2x Drive Motors (drive with 3-6VDC, 200-400 mA run, 1.5A hard stall)	Batteries	Any	2x Drive Motors (drive with 3-6VDC, 200-400 mA run, 1.5A hard stall)	10	10	
	2x Wheels	1x Battery charger		Batteries	15-40	4--6	
	1x Plastic Caster Ball	1x battery holder		1x Microcontroller or microprocessor	15-50		
	Anodized aluminum frame and all mounting hardware for assembly	1x Microcontroller or microprocessor		1x motor driver	10 -- 20		
				1x motor driver	2x IR line tracking sensors	7	
				2x IR line tracking sensors	1x Ultrasonic sensor	4	
				1x Ultrasonic sensor	2x Encoder Sensor	6	
				2x Encoder Sensor	2x Wheels	5	

Mini robot rover chassis Kit- 4WD w DC Motors	4x Wheels	Batteries	Any	4x DC Motors in MicroServo shape	20	10
	4x DC Motors in MicroServo shape	1x Battery charger		Batteries	15-40	4--6
	1x Metal Chassis	1x battery holder		1x Microcontroller or microprocessor	15-50	
		1x Microcontroller or microprocessor		1x motor driver	10 -- 20	
		1x motor driver		2x IR line tracking sensors	7	
		2x IR line tracking sensors		1x Ultrasonic sensor	4	
		1x Ultrasonic sensor		2x Encoder Sensor	6	
		2x Encoder Sensor		4x Wheels	10	
Mini robot rover chassis Kit- 2WD w DC Motors	2x Wheels	Batteries	Any	2x Wheels	5	
	2x DC Motors in MicroServo shape	1x Battery charger		2x DC Motors in MicroServo shape	10	10
	1x Support Wheel	1x battery holder		Batteries	15-40	4--6
	1x Metal Chassis	1x Microcontroller or microprocessor		1x Microcontroller or microprocessor	15-50	
		1x motor driver		1x motor driver	10--20	



		2x IR line tracking sensors		2x IR line tracking sensors	7	
		1x Ultrasonic sensor		1x Ultrasonic sensor	4	
		2x Encoder Sensor		2x Encoder Sensor	6	
Mini 3-layer round robot chssis kit- 2WD w DC motors	2x Drive Motors (drive with 3-6VDC, 200-400 mA run, 1.5A hard stall)	Batteries	Any	2x Drive Motors (drive with 3-6VDC, 200-400 mA run, 1.5A hard stall)	10	10
	2x Wheels	1x Battery charger		Batteries	15-40	4--6
	1x Plastic Caster Ball	1x battery holder		1x Microcontroller or microprocessor	15-50	
	Anodized aluminum frames and all mounting hardware for assembly	1x Microcontroller or microprocessor		1x motor driver	10--20	
		1x motor driver		2x IR line tracking sensors	7	
		2x IR line tracking sensors		1x Ultrasonic sensor	4	
		1x Ultrasonic sensor		2x Encoder Sensor	6	
		2x Encoder Sensor		2x Wheels	5	

AlphaBot	1x AlphaBot mainboard	2x Batteries 18650	Any	1x Tracker Sensor		
	1x Tracker Sensor	1x Battery charger		2x Photo Interrupter Sensor	8	
	2x Photo Interrupter Sensor	1x battery holder		2x Infrared Proximity Sensor	8	
	2x Infrared Proximity Sensor	2x IR line tracking sensors		2x Motor with gearbox	20	10
	2x Motor with gearbox	2x Encoder Sensor		2x AlphaBot wheel	5	
	2x AlphaBot wheel			2x Batteries 18650	30	4--6
	1x AlphaBot acrylic chassis			2x IR line tracking sensors	7	
	4x Motor mounting plate			2x Encoder Sensor	6	
	1x omni-direction wheel					
	2x 20-slots encoder disk					
	1x IR remote controller					
	2x XH2.54 4cm 4Pin					
	2x XH2.54 4cm 3Pin					
	1x XH2.54 4cm 7Pin					
	1x AlphaBot screws					

PiRacer DonkeyCar, AI Racing Robot for Raspberry Pi 4	6x PiRacer metal kit	1x micro SD 16GB	1x RPi Camera (G)	30	
	2x EVA felt pad	1x Raspberry	2x Metal Gearmotor	25	10
	1x 3D-printed motor lid	1x motor driver	1x MG996R servo	15	10
	1x PiRacer expansion board	3x Batteries 18650	1x MG996R servo wheel	10	
	1x RPi Camera (G)	2x IR line tracking sensors	1x Raspberry	50	
	1x Acrylic piece	2x Encoder Sensor	1x motor driver	20	
	1x Servo mount		3x Batteries 18650	35	4--6
	2x Metal Gearmotor		2x IR line tracking sensors	7	
	4x Tire		2x Encoder Sensor	6	
	2x Coupler				
	1x MG996R servo				
	1x MG996R servo wheel				
	2x Steering knuckle				
	4x Ball bearing				
	4x M3 plastic ball joint				
	2x M3 pull bar				
	1x 12.6V battery charger + EU head				
	1x Wireless gamepad				
	1x 6Pin cable				
	2x Spanner				

Raspberry

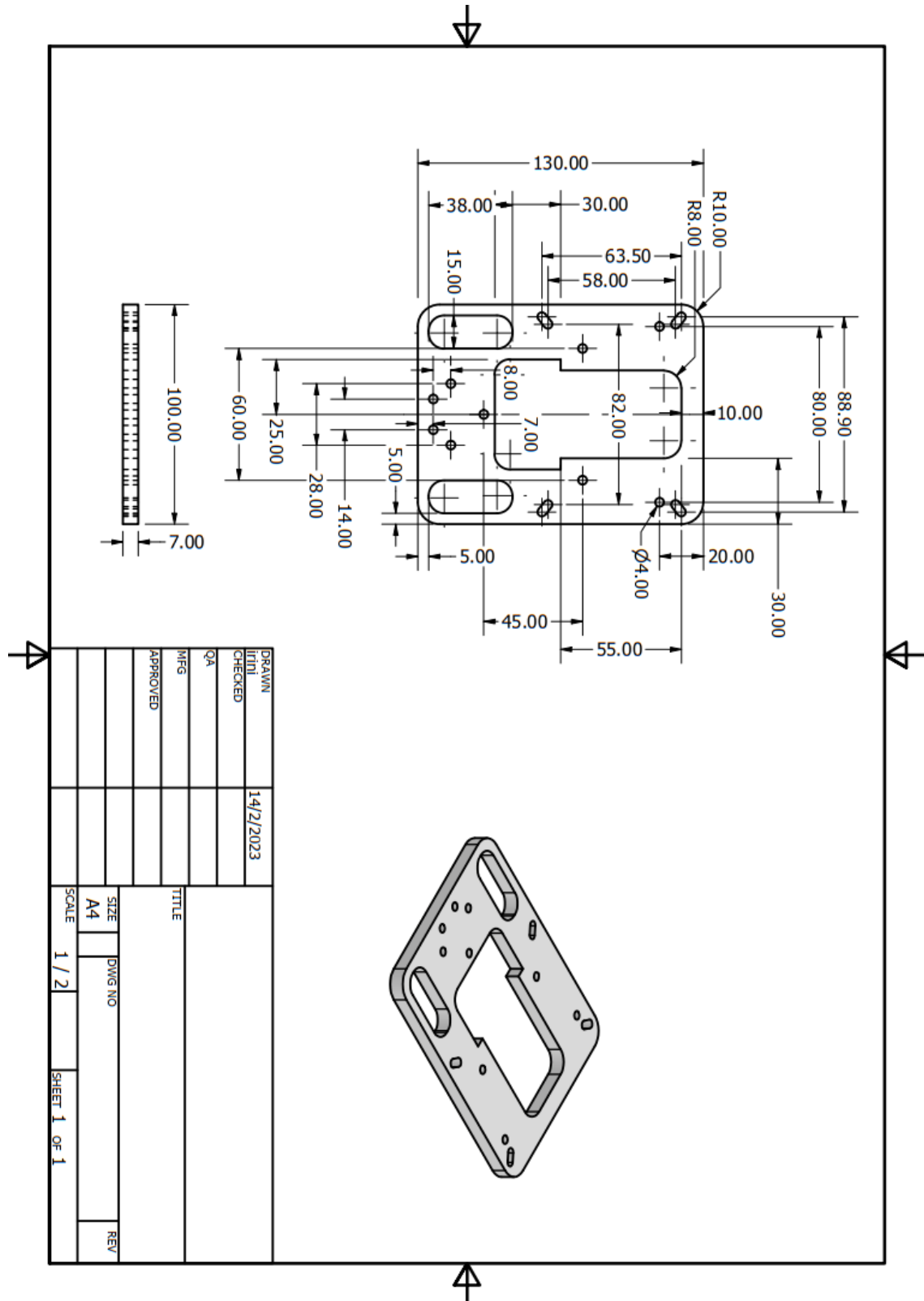
	2x Screwdriver					
	1x Mini cross wrench sleeve					
	1x Micro SD card reader					
	1x Screws pack					
	1x Track map					
PiRacer Pro High Speed, AI Racing Robot for Raspberry Pi 4	1x On-road chassis	1x micro SD 16GB	Raspberry	1x RPi Camera (G)	30	
	1x PiRacer Pro expansion board	1x Raspberry		1x Raspberry	50	
	1x RPi Camera (G)	1x motor driver		1x motor driver	20	
	1x Acrylic camera spacer	3x Batteries 18650		3x Batteries 18650	35	4--6
	1x Camera holder	2x IR line tracking sensors		2x IR line tracking sensors	7	
	1x 8.4V battery charger + EU head	2x Encoder Sensor		2x Encoder Sensor	6	
	1x Wireless gamepad					
	1x 6Pin cable					
	1x Spanner					
	2x Screwdriver					
	1x Micro SD card reader					
	1x Screws pack					
1x Track map						
AlphaBot2 Robot Building Kit for Raspberry Pi 3 Model B	1x AlphaBot2-Pi (adapter board)	1x Raspberry	Raspberry	1x RPi Camera (B)	20	
	1x AlphaBot2-Base (base chassis)	1x motor driver		2x SG90 servo	10	10
	1x RPi Camera (B)	2x Batteries 14500		1x Raspberry	50	

1x Micro SD Card 16GB	2x IR line tracking sensors		
1x Power adapter US standard 5V/3A micro	2x Encoder Sensor		
2x SG90 servo			
1x 2 DOF pan and tilt kit			
1x IR remote controller			
1x FC-20P cable 8cm			
1x USB type A plug to micro B plug cable			
1x 15PIN FFC 25cm			
1x AlphaBot2-Pi screws			
1x Micro SD Card Reader			
1x Screwdriver			
		1x motor driver	20
		2x Batteries 14500	25 4--6
		2x IR line tracking sensors	7
		2x Encoder Sensor	6

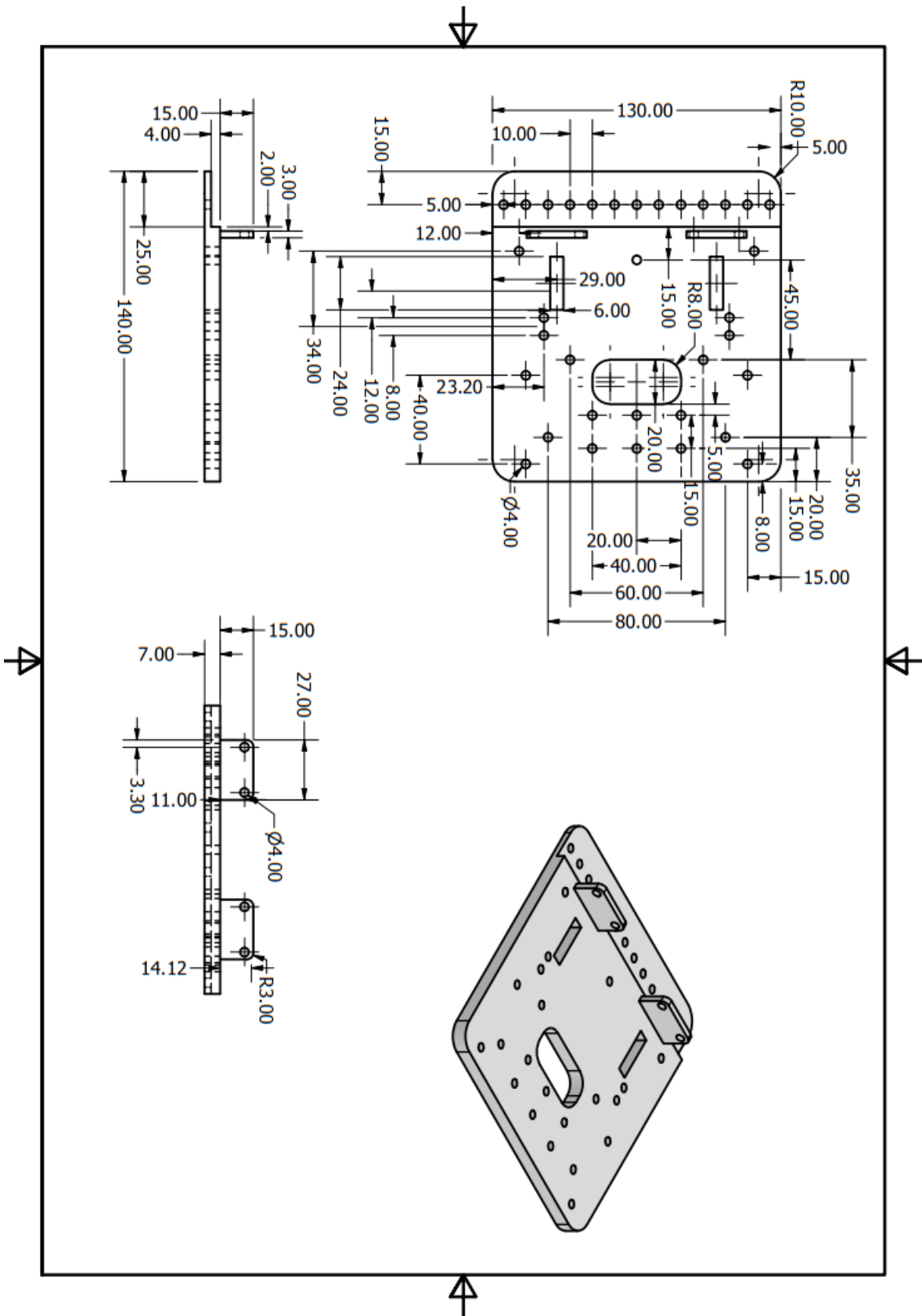
## Παράρτημα Β: Σχέδια κατασκευής

Αυτά που ακολουθούν αποτελούν φωτογραφίες/ στιγμιότυπα οθόνης και τα αρχεία σε μορφή PDF, έχουν σταλεί μαζί με την εργασία στον επιβλέπων καθηγητή.

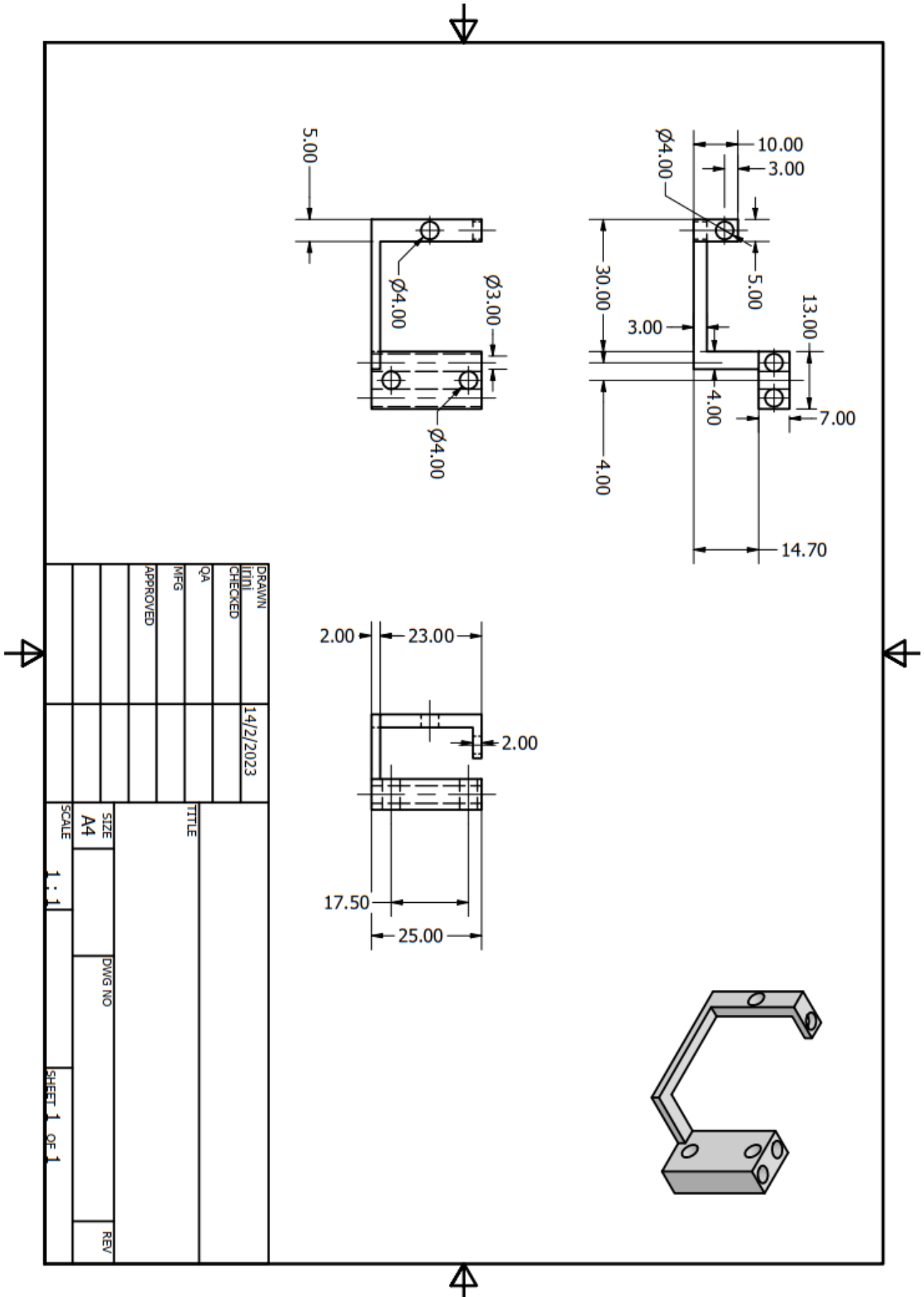
Σχέδιο της πάνω πλάκας, σε κλίμακα 1 / 2



Σχέδιο κάτω πλάκας για κινητήρες DC, σε κλίμακα 1/2



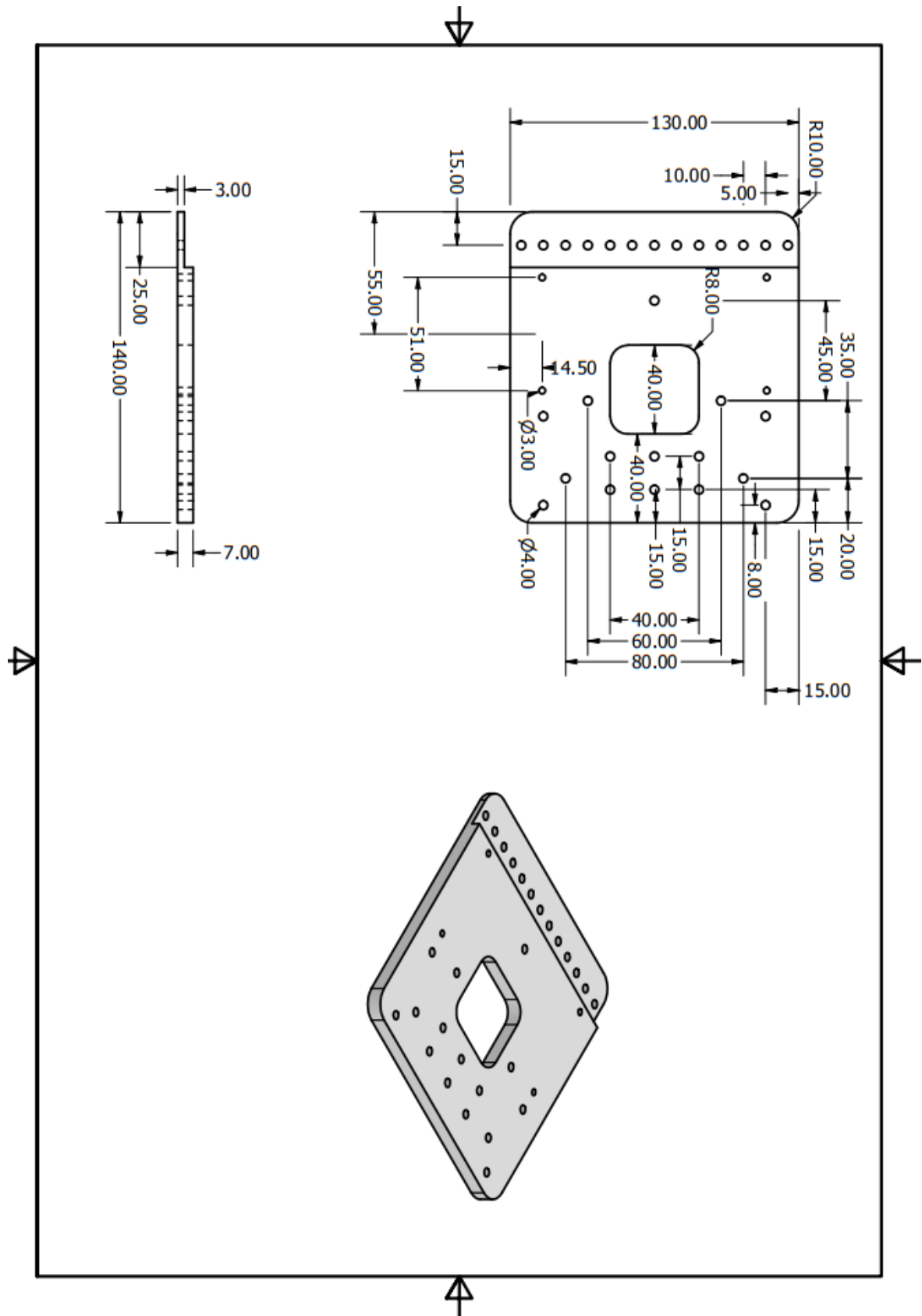
Σχέδιο για το δεξί στήριγμα του κινητήρα DC, σε κλίμακα 1:1



DRAWN		14/2/2023	TITLE	
CHECKED			SIZE	
QA			A4	DWG NO
MFG			SCALE	REV
APPROVED			1 : 1	SHEET 1 OF 1

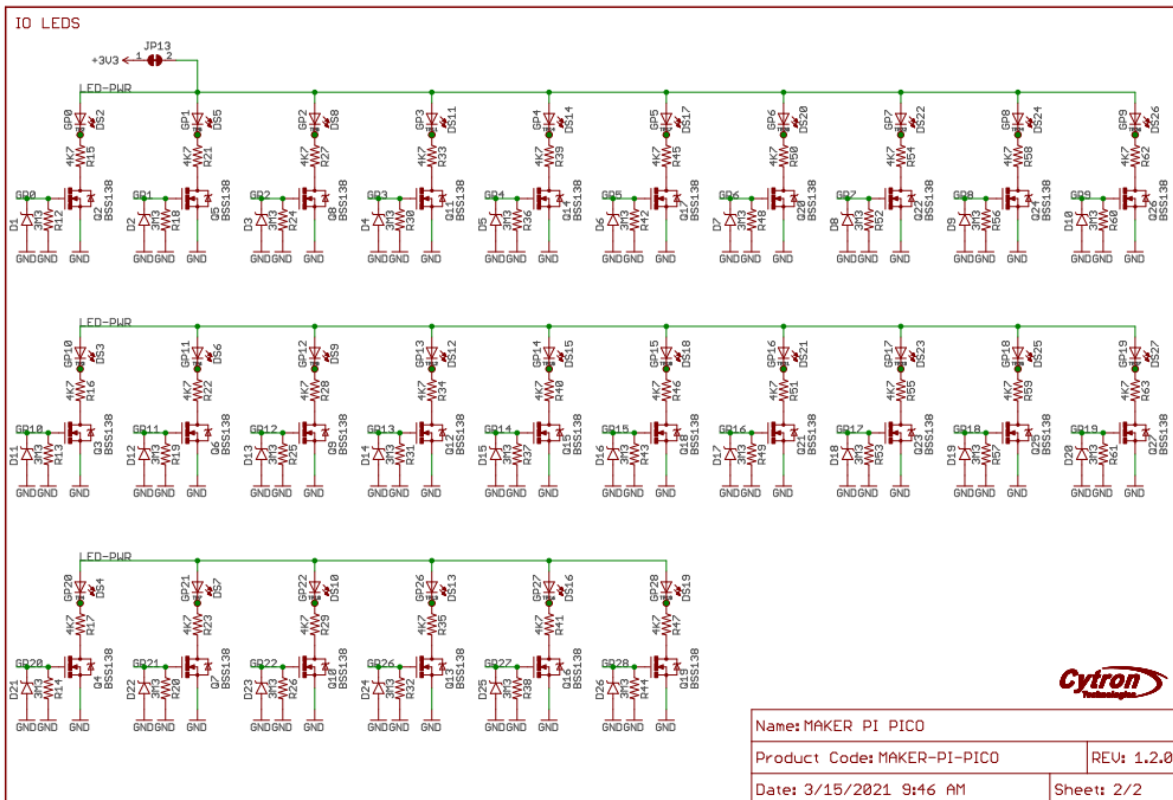
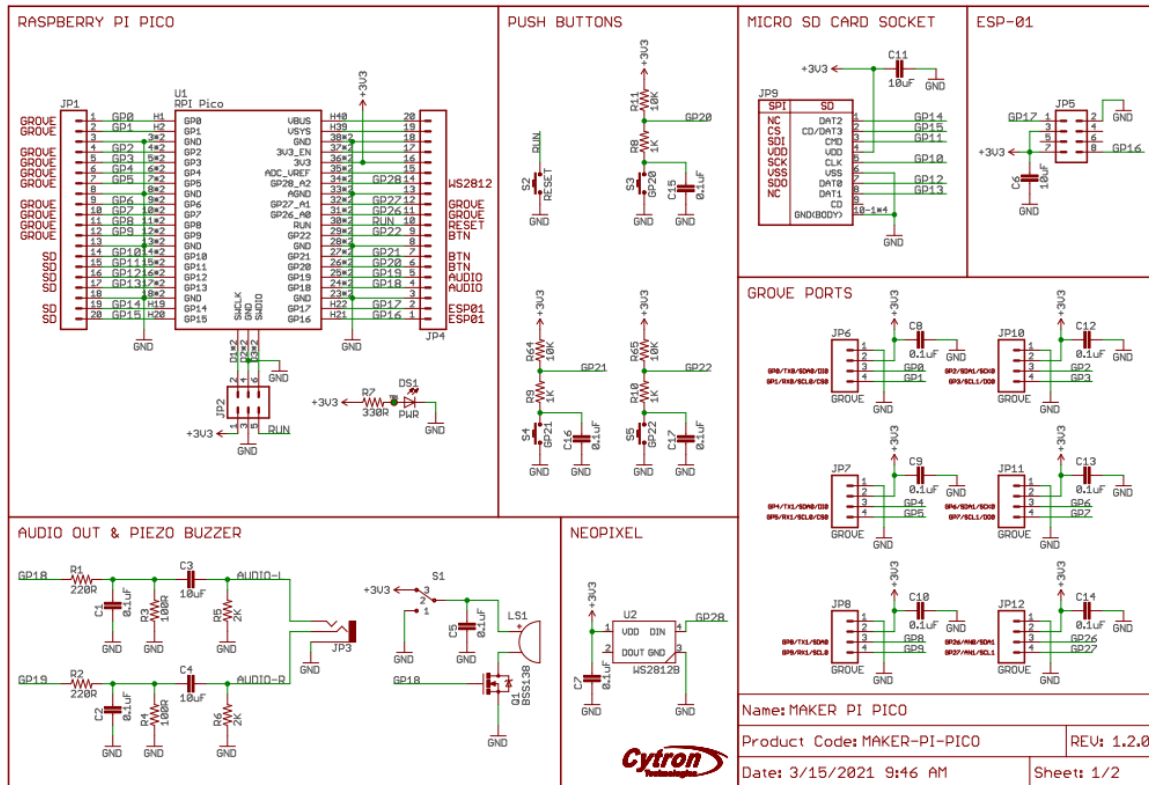


Σχέδιο κάτω πλάκας για κινητήρες Servo, σε κλίμακα 1/2

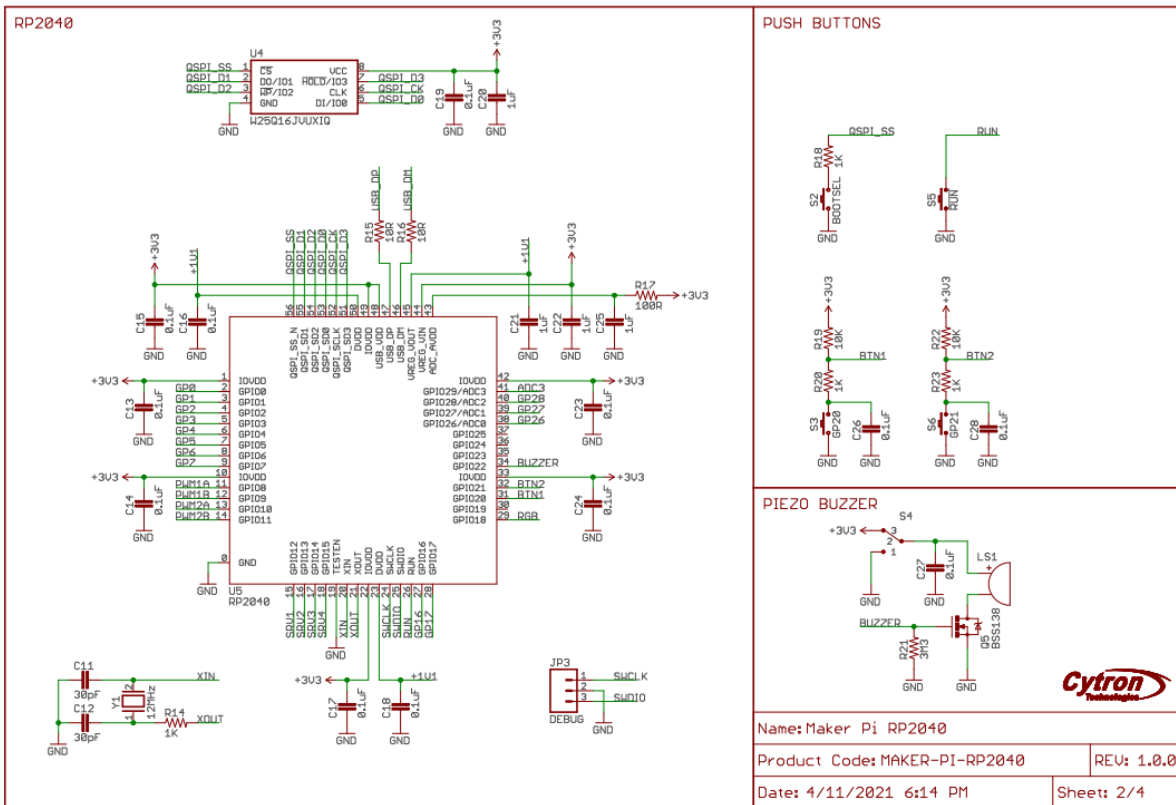
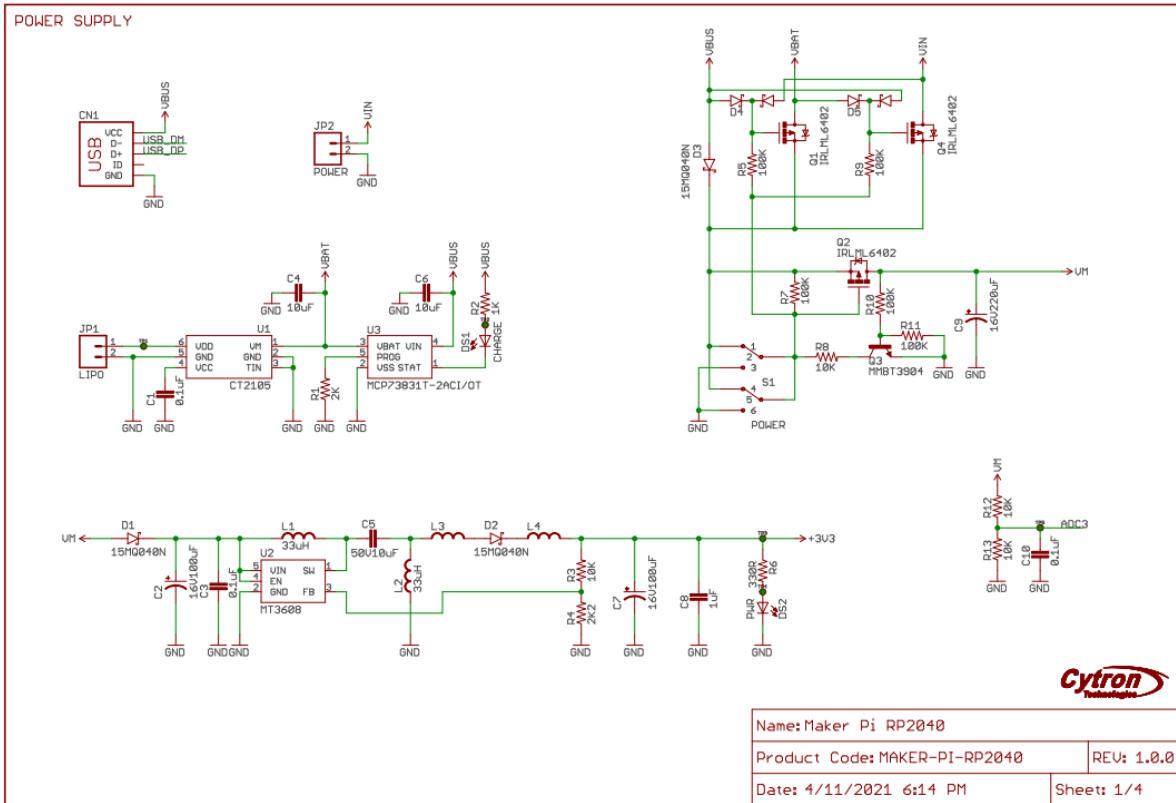


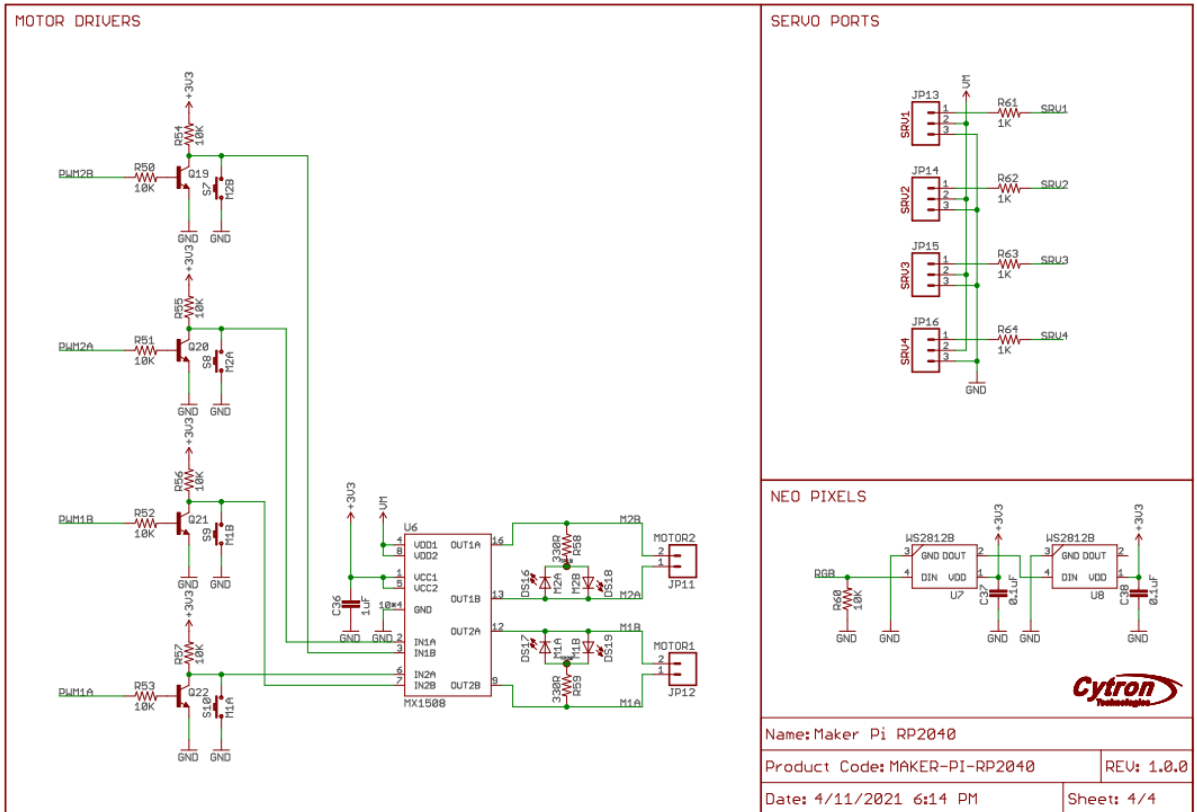
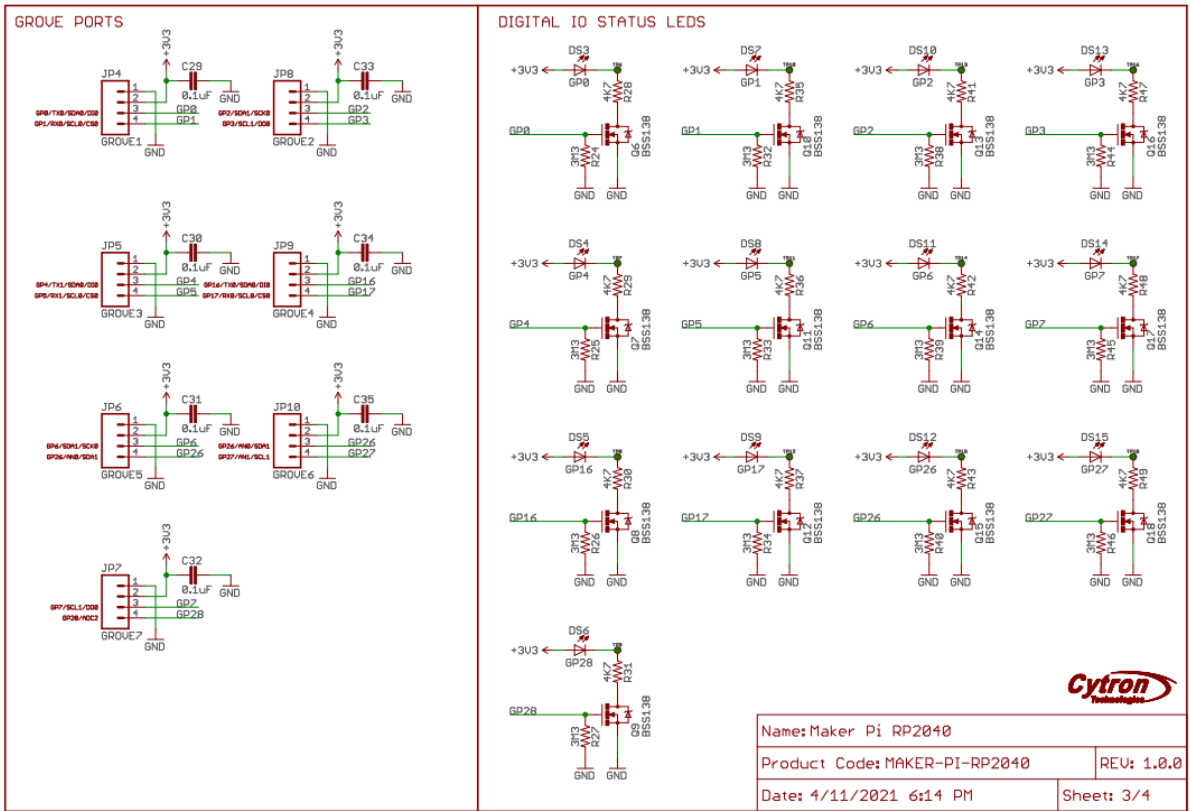
# Παράρτημα Γ: Ηλεκτρονικά Σχέδια

## Maker Pi Pico



# Maker Pi RP2040





## Παράρτημα Δ: Κώδικες Επίλυσης των ενδεικτικών Ασκήσεων για την πλακέτα Maker Pi RP2040

Στην συνέχεια δίνονται μερικοί κώδικες, τους οποίους κάποιος μπορεί να χρησιμοποιήσει για την επίλυση των ασκήσεων.

### Άσκηση 1

```
Κώδικας για NeoPixel

# Blinking NeoPixel
# Code for NeoPixelin Maker Pi RP2040

# Importing the necessary libraries
import machine
import neopixel
import time

pin = machine.Pin(18, machine.Pin.OUT) # set PIO0 to output to drive NeoPixels
np = neopixel.NeoPixel(pin, 8)        # create NeoPixel driver on PIO0 for 8 pixels

for i in range(0, 10):
    np[0] = (100, 50, 0)                # set the first pixel to color yellow
    np.write()                          # write data to all pixels
    r, g, b = np[0]                     # get first pixel colour
    time.sleep(0.1)

    np[0] = (0, 0, 0)                   # set the first pixel to off
    np.write()                          # write data to all pixels
    r, g, b = np[0]                     # get first pixel colour
    time.sleep(0.05)

for i in range(0, 10):
    np[0] = (0, 0, 190)                  # set the first pixel to blue
    np.write()                          # write data to all pixels
    r, g, b = np[0]                     # get first pixel colour
    time.sleep(0.1)

    np[0] = (0, 0, 0)                   # set the first pixel to off
    np.write()                          # write data to all pixels
```

```
r, g, b = np[0]          # get first pixel colour
time.sleep(0.05)
```

### Άσκηση 3

Κώδικας για κίνηση μπροστά με ανάγνωση μαύρου ίχνους

```
import machine
import time

# Setup analog pins for tcr5000
analog1 = machine.ADC(0)
analog2 = machine.ADC(26)

# Setup DC Motor pins
M1A = machine.PWM(machine.Pin(9))
M1B = machine.PWM(machine.Pin(8))
M2A = machine.PWM(machine.Pin(11))
M2B = machine.PWM(machine.Pin(10))
M1A.freq(50)
M1B.freq(50)
M2A.freq(50)
M2B.freq(50)

def forward(speed):
    M1A.duty_u16(0)
    M1B.duty_u16(speed)
    M2A.duty_u16(0)
    M2B.duty_u16(speed)

# MAIN PROGRAM
while analog2.read_u16() < 3000: # Example of sensor read
    forward(25000)

forward(0)
```

## Παράρτημα Ε: Φύλλα έργου φοιτητών

### Άσκηση 1: Εισαγωγή στην μικρορομποτική διάταξη: Χρήση LED

#### Στόχος

Αυτή η πρώτη εισαγωγική άσκηση είναι η αρχή του εργαστηριακού μέρους του μαθήματος της Μηχατρονικής. Βασικός στόχος είναι να εισαχθείτε στο προγραμματισμό μικροελεγκτών με τη χρήση μιας καινούργιας προγραμματιστικής γλώσσας της MicroPython. Η MicroPython είναι κομμάτι της Python και αυτό σημαίνει ότι έχει πρόσβαση σε αρκετές από τις βιβλιοθήκες της.

#### Οδηγίες

Με αυτήν την άσκηση καλείστε να δημιουργήσετε έναν κώδικα, ο οποίος είναι ικανός να ελέγξει τουλάχιστον 5 LED λαμπάκια. Οι εργασίες που χρειάζεται να ολοκληρώσετε είναι οι εξής:

Εργασία	Περιγραφή
1	Ορίζετε τις μεταβλητές για τα LED από το 1 έως το 5 με τις ακόλουθες ονομασίες: LED0, LED1, LED2, LED3, LED4.
2	Ανάβετε τα LED διαδοχικά από το 1 μέχρι το 5 κάθε 2 δευτερόλεπτα, με την εξής σειρά: LED0 > LED1 > LED2 > LED3 > LED4
3	Σβήνετε τα LED διαδοχικά από το 1 μέχρι το 5 κάθε 2 δευτερόλεπτα, με την εξής σειρά: LED0 > LED1 > LED2 > LED3 > LED4
4	Ανέβετε τα LED 1, 3 και 5 ταυτόχρονα, για 2 δευτερόλεπτα: LED0 > LED2 > LED4
5	Σβήνετε τα LED 1, 3 και 5 ταυτόχρονα, για 2 δευτερόλεπτα: LED0 > LED2 > LED4

Οι βασικές εντολές που θα χρησιμοποιήσετε για την επίτευξη της εργασίας εξηγούνται από τον διδάσκοντα και δίνονται στον παρακάτω πίνακα:

Βιβλιοθήκες	Εντολές
machine	<code>variable = machine.Pin( pin number, machine.Pin.OUT)</code>
	<code>variable.value () [0: off, 1: on]</code> εναλλακτικά: <code>variable.on()</code>

	<code>variable.off()</code>
time	<code>time.sleep(time)</code> <code>time.sleep_ms(time)</code> <code>time.sleep_us(time)</code>
γενικά	<code>for i in range (0, number)</code>

### Παραδοτέο

Με την ολοκλήρωση της άσκησης 1, παραδίδεται στον διδάσκοντα ο κώδικας που χρησιμοποιήθηκε μαζί με τον κατάλληλο σχολιασμό, σε μορφή κειμένου. Επίσης, μαζί με τον κώδικα παραδίδονται τα σχετικά συμπεράσματα από την εργασία. Μερικές ενδεικτικές ερωτήσεις που μπορείτε να απαντήσετε κατά την διάρκεια της εργασίας είναι:

- Εκτελέστηκε με επιτυχία ο αρχικός κώδικας;
- Ήταν ξεκάθαρος και εύκολος ο ορισμός των μεταβλητών;
- Ήταν κατανοητή η δομή του κώδικα;
- Υπήρχε κάποιο κομμάτι της άσκησης το οποίο ήταν δυσνόητο ή δύσκολο;
- Θα προτείνατε κάποια αλλαγή;

### Εργασίες Διερεύνησης (προαιρετικά)

Με την επιτυχή ολοκλήρωση των παραπάνω εργασιών, μπορείτε να εκτελέσετε και κάποιες από τις παρακάτω εργασίες.

Εργασία	Περιγραφή
6	Ανέβετε τα LED 1 , 3 και 5 ταυτόχρονα, για 2 δευτερόλεπτα: LED0 > LED2 > LED4 Σβήνετε τα LED 2 και 4 ταυτόχρονα, για 2 δευτερόλεπτα: LED1 > LED3
7	Εκτελείτε την εργασία 6 για 15 φορές.
8	Ελεύθερο



## Άσκηση 2: Απλές κινήσεις

### Μέρος 1<sup>ο</sup>

#### Στόχος

Με την ολοκλήρωση της άσκησης 1, είστε σε θέση να προγραμματίσετε τους κινητήρες. Μετά την εισαγωγή της βασικής βιβλιοθήκης της MicroPython για μικροελεγκτές, machine, ο προγραμματισμός των κινητήρων είναι το επόμενο μεγάλο βήμα στην κατανόηση της κίνησης μιας πλατφόρμας. Για αυτό και η άσκηση 2, μέρος 1<sup>ο</sup>, επικεντρώνεται στις κινήσεις

#### Οδηγίες

Με την χρήση της πλακέτας Maker Pi RP2040, οι κινητήρες DC ενώνονται στις ακίδες 8 και 9 για τον motor 1 και 10 και 11 για τον motor 2. Επειδή οι διατάξεις είναι κοινές, ποιες ακίδες είναι για την μπροστά κίνηση και ποιες είναι για την πίσω είναι ορισμένες από την αρχή αλλά σε κάθε περίπτωση η πλακέτα διαθέτει τέσσερα κουμπιά για να ελέγξετε την φορά των κινητήρων. Σε περίπτωση που δεν είναι όπως ορίζονται στην συνέχεια, ρωτήστε τον υπεύθυνο διδάσκοντα.

Κινητήρας	Πλευρά	Ακίδα	Κίνηση
Motor 1	δεξιά	8	μπροστά
		9	πίσω
Motor 2	αριστερά	10	μπροστά
		11	πίσω



Θέσεις των κινητήρων DC στην πλακέτα Maker Pi RP2040

Στην συνέχεια για κινηθεί η πλατφόρμα χρειάζεται να ορίζεται η ταχύτητα. Επειδή η πλακέτα είναι δυναμικότητας 16 Bits, το εύρος των τιμών που μπορεί να δεχτεί θεωρητικά είναι από 0 έως 65535, δηλαδή 65536. Μετά από δοκιμές το πραγματικό εύρος των τιμών για συχνότητα 50 Hz είναι από 15000 μέχρι 65535.

Μαζί με τις απλές κινήσεις είναι και οι στροφές. Οι στροφές που θα χρησιμοποιηθούν κατά την διάρκεια των μαθημάτων ονομάζονται “SPIN” και χρησιμοποιούν και τους δύο κινητήρες με αντίθετες ταχύτητες. Επειδή είναι δύσκολο το κομμάτι αυτό δίνεται στην συνέχεια ένα απόκομμα κώδικα με τυχαία στροφή:

Κώδικας
M1B.duty_u16(0) M1A.duty_u16(speed) M2B.duty_u16(0) M2A.duty_u16(speed) time.sleep(time)

Οι εργασίες που καλείστε να ολοκληρώσετε αποτελούν κομμάτια μιας απλής διαδρομής, η οποία δίνεται αναλυτικά στον επόμενο πίνακα.

Εργασία	Περιγραφή
1	Κίνηση μπροστά με ταχύτητα 25000 για 3 δευτερόλεπτα και τυπώστε την λέξη «forward»
2	Σταμάτημα της κίνησης
3	Κίνηση πίσω με ταχύτητα 30000 για 2 δευτερόλεπτα και τυπώστε την λέξη «back»
4	Σταμάτημα της κίνησης
5	Δεξιά στροφή με ταχύτητα 28000 για 0.8 δευτερόλεπτα και τυπώστε την λέξη «right»
6	Σταμάτημα της κίνησης
7	Κίνηση πίσω με ταχύτητα 32000 για 2.5 δευτερόλεπτα και τυπώστε την λέξη «back»
8	Σταμάτημα της κίνησης
9	Αριστερή στροφή με ταχύτητα 28000 για 0.8 δευτερόλεπτα και τυπώστε την λέξη «left»
10	Κίνηση μπροστά με ταχύτητα 35000 για 3.2 δευτερόλεπτα και τυπώστε την λέξη «forward»
11	Σταμάτημα της κίνησης

Οι βασικές εντολές που θα χρησιμοποιήσετε για την επίτευξη της εργασίας εξηγούνται από τον διδάσκοντα και δίνονται στον παρακάτω πίνακα:

Βιβλιοθήκες	Εντολές
machine	<i>variable</i> = machine.PWM(machine.Pin( <i>pin number</i> ))
	<i>variable</i> .freq( <i>number</i> )

	<code>variable.duty_u16(speed)</code>
γενικά	<code>for i in range (0, number)</code>
	<code>print()</code>

### Παραδοτέο

Με την ολοκλήρωση της άσκησης 2, παραδίδεται στον διδάσκοντα ο κώδικας που χρησιμοποιήθηκε μαζί με τον κατάλληλο σχολιασμό, σε μορφή κειμένου. Επίσης, μαζί με τον κώδικα παραδίδονται τα σχετικά συμπεράσματα από την εργασία. Μερικές ενδεικτικές ερωτήσεις που μπορείτε να απαντήσετε κατά την διάρκεια της εργασίας είναι:

- Εκτελέστηκε με επιτυχία ο αρχικός κώδικας;
- Ήταν ξεκάθαρος και εύκολος ο ορισμός των μεταβλητών και των ακίδων;
- Ήταν κατανοητή η δομή του κώδικα;
- Υπήρχε κάποιο κομμάτι της άσκησης το οποίο ήταν δυσνόητο ή δύσκολο;
- Θα προτείνατε κάποια αλλαγή;

### Εργασίες Διερεύνησης (προαιρετικά)

Με την επιτυχή ολοκλήρωση των παραπάνω εργασιών, μπορείτε να εκτελέσετε και κάποιες από τις παρακάτω εργασίες.

Εργασία	Περιγραφή
12	Βελτίωση του κώδικα με την χρήση της τάξης <code>def</code>
13	Κατάλληλη τροποποίηση του κώδικα για να εκτελέσει ένα τετράγωνο

Ερώτηση διερεύνησης : Τι πιστεύετε ότι θα γίνει άμα δεν βάζε το σταμάτημα μετά από κάθε κίνηση;

## Άσκηση 2: Απλές κινήσεις

### Μέρος 2<sup>ο</sup>

#### Στόχος

Σε συνέχεια του μέρος 1<sup>ου</sup>, καλείστε να εισάγετε την χρήση του τρίτου κινητήρα. Στην περίπτωση αυτή είναι ένας Servo, ο οποίος είναι μαζί με έναν μηχανισμό αρπάγης (gripper). Με αυτόν το τρόπο μπορείτε να εξοικειωθείτε με τις κινήσεις και τις κατάλληλες βιβλιοθήκες. Επίσης σε αυτήν την άσκηση εισάγεται η έννοια της συνάρτησης def, η οποία είναι ένα κομμάτι κώδικα που εκτελεί μια συγκεκριμένη εργασία σε αρχικό σημείο στο πρόγραμμα και δεν επηρεάζει το κύριο κομμάτι του κώδικα.

#### Οδηγίες

Με τη χρήση της πλακέτας Maker Pi RP2040, η χρήση των κινητήρων είναι αρκετά απλή. Στην πλακέτα υπάρχουν οι κατάλληλες ακίδες για τέσσερις κινητήρες servo. Αυτές αντιστοιχούν στις ακίδες 12 έως 15. Επιλέξτε μία από αυτές και χρησιμοποιήστε το κομμάτι του κώδικα που δίνεται στην συνέχεια για εντοπίσετε την φορά με την οποία ανοίγει και κλείνει ο μηχανισμός.



Για να λειτουργήσει ο κώδικας πρέπει να οριστούν πρώτα οι ακίδες και στη συνέχεια να δοθεί η σωστή εντολή για την κίνηση του μηχανισμού. Αφού γίνει αυτό, μπορείτε να λύσετε τις εργασίες όπως αναφέρονται στον πίνακα.

Κώδικας για τον κινητήρα servo στην ακίδα 12

```
# Servo motor GP12
# fine tune the duty cycle values to suit your servo motor
MIN_DUTY = 1000
MAX_DUTY = 8400

pwm = machine.PWM(machine.Pin(12))
pwm.freq(50)
pwm.duty_u16(MIN_DUTY)
time.sleep_ms(500)
pwm.duty_u16(MAX_DUTY)
time.sleep_ms(500)
```

Για να θεωρηθεί επιτυχημένη αυτή η άσκηση ο κύριος κώδικας πρέπει να έχει κλήσεις από τις κατάλληλες συναρτήσεις def με τα ονόματα των κινήσεων που έχουμε από την άσκηση 2, μέρος 1° .

Εργασία	Περιγραφή	Όνομα συνάρτησης
0	Ο μηχανισμός βρίσκεται στην θέση 0, δηλαδή βρίσκεται στην κλειστή του θέση.	
1	Κίνηση μπροστά με ταχύτητα 25000 για 3 δευτερόλεπτα και τυπώστε την λέξη «forward»	def forward (DC_speed, time)
2	Σταμάτημα της κίνησης	def stop (DC_speed, time)
3	Κίνηση πίσω με ταχύτητα 30000 για 2 δευτερόλεπτα και τυπώστε την λέξη «back»	def back (DC_speed, time)
4	Σταμάτημα της κίνησης	
5	Κινητήρας σέρβο ανοίγει με ταχύτητα 3000 για γωνία ίση με 90 μοίρες	def servo_open (servo_speed, angle)
6	Δεξιά στροφή με ταχύτητα 28000 για 0.8 δευτερόλεπτα και τυπώστε την λέξη «right»	def right_spin (DC_speed, time)
7	Σταμάτημα της κίνησης	
8	Κίνηση πίσω με ταχύτητα 32000 για 2.5 δευτερόλεπτα και τυπώστε την λέξη «back»	
9	Σταμάτημα της κίνησης	
10	Κινητήρας σέρβο κλείνει με ταχύτητα 3000 για γωνία ίση με 90 μοίρες	def servo_close (servo_speed, angle)
11	Αριστερή στροφή με ταχύτητα 28000 για 0.8 δευτερόλεπτα και τυπώστε την λέξη «left»	def left_spin (DC_speed, time)
12	Σταμάτημα της κίνησης	
13	Κίνηση μπροστά με ταχύτητα 35000 για 3.2 δευτερόλεπτα και τυπώστε την λέξη «forward»	
14	Σταμάτημα της κίνησης	
15	Κινητήρας σέρβο ανοίγει με ταχύτητα 3000 για γωνία ίση με 180 μοίρες	

### Παραδοτέο

Με την ολοκλήρωση της άσκησης 2, παραδίδεται στον διδάσκοντα ο κώδικας που χρησιμοποιήθηκε μαζί με τον κατάλληλο σχολιασμό, σε μορφή κειμένου. Επίσης, μαζί με τον κώδικα παραδίδονται τα σχετικά συμπεράσματα από την εργασία. Μερικές ενδεικτικές ερωτήσεις που μπορείτε να απαντήσετε κατά την διάρκεια της εργασίας είναι:

- Εκτελέστηκε με επιτυχία ο αρχικός κώδικας;

- Ήταν ξεκάθαρος και εύκολος ο ορισμός των μεταβλητών και των ακίδων;
- Ήταν κατανοητή η δομή του κώδικα;
- Υπήρχε κάποιο κομμάτι της άσκησης το οποίο ήταν δυσνόητο ή δύσκολο;
- Θα προτείνατε κάποια αλλαγή;

**Εργασίες Διερεύνησης (προαιρετικά)**

Εργασία	Περιγραφή
10	Ολοκλήρωση ενός τετραγώνου με τις λιγότερες δυνατές γραμμές κώδικα

### Άσκηση 3: Εισαγωγή στους αισθητήρες

#### Στόχος

Οι προηγούμενες ασκήσεις αποτελούσαν την εισαγωγή στην προγραμματιστική γλώσσα και στην κίνηση του ρομπότ. Ωστόσο, οι περισσότερες ρομποτικές διατάξεις χρησιμοποιούν και μια πληθώρα αισθητήρων. Επομένως σε αυτή την άσκηση, καλείστε να αναπτύξετε ένα κώδικα σύμφωνα με τον οποίο η διάταξη θα κινηθεί μπροστά μέχρι να εντοπίσει κάποιο ίχνος στο έδαφος.

#### Οδηγίες

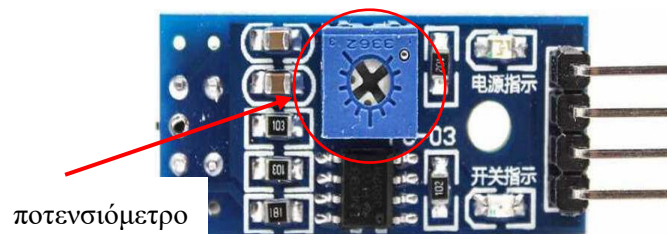
Με την εισαγωγή των αισθητήριων οργάνων σε μία διάταξη, είναι ευκολότερο να εκπληρωθεί ένα μεγαλύτερο εύρος εργασιών. Ωστόσο παράλληλα με αυτό αυξάνεται και ο βαθμός δυσκολίας και πολυπλοκότητας του προγράμματος. Σε αυτή την εργασία, χρησιμοποιώντας έναν από τους δυο διαθέσιμους αισθητήρες αναγνώρισης ίχνους, καλείστε να συντάξετε τον κώδικα που θα εκτελεί την εξής εργασία:

Εργασία	Περιγραφή
1	Κίνηση μπροστά με ταχύτητα 20000, μέχρι ο αισθητήρας να δει τη μαύρη γραμμή.
2	Σταμάτημα της κίνησης

Για αυτήν εργασία είναι απαραίτητη η τοποθέτηση μίας μαύρης γραμμής στο έδαφος, ιδανικά σε κάποια άσπρη επιφάνεια παράλληλη με το έδαφος. Η γραμμή αυτή μπορεί να γίνει με μία απλή μονωτική ταινία.

Πριν ξεκινήσετε το πρόγραμμα, πρέπει να βεβαιωθείτε ότι ο αισθητήρας είναι στην κατάλληλη ευαισθησία για τον χώρο που βρίσκεστε. Αυτό γίνεται περιστρέφοντας το ποτενσιόμετρο του αισθητήρα. Όταν ανάβουν και τα 2 LED σημαίνει ότι ο αισθητήρας είναι στην ανοιχτόχρωμη περιοχή, ενώ όταν ανάβει μόνο το ένα σημαίνει ότι εντοπίζει την μαύρη γραμμή. Με την ολοκλήρωση αυτής της διαδικασίας χρησιμοποιείτε τον παρακάτω κώδικα για να διαβάσετε την τιμή του αισθητήρα στην ανοιχτόχρωμη και στην μαύρη περιοχή και συμπληρώστε τα στον πίνακα

Ανοιχτόχρωμη περιοχή	Μαύρη γραμμή



## Παραδοτέο

Με την ολοκλήρωση της άσκησης 3, παραδίδεται στον διδάσκοντα ο κώδικας που χρησιμοποιήθηκε μαζί με τον κατάλληλο σχολιασμό, σε μορφή κειμένου. Επίσης, μαζί με τον κώδικα παραδίδονται τα σχετικά συμπεράσματα από την εργασία. Μερικές ενδεικτικές ερωτήσεις που μπορείτε να απαντήσετε κατά την διάρκεια της εργασίας είναι:

- Τι πιστεύετε ότι θα συμβεί άμα δεν βάλετε το σταμάτημα της κίνησης στο τέλος;
- Εκτελέστηκε με επιτυχία ο αρχικός κώδικας;
- Ήταν ξεκάθαρος και εύκολος ο ορισμός των μεταβλητών και των ακίδων;
- Ήταν κατανοητή η δομή του κώδικα;
- Υπήρχε κάποιο κομμάτι της άσκησης το οποίο ήταν δυσνόητο ή δύσκολο;
- Θα προτείνατε κάποια αλλαγή;

## Εργασίες Διερεύνησης (προαιρετικά)

Εργασία	Περιγραφή
3	Κίνηση μπροστά με ταχύτητα 20000, μέχρι ο αισθητήρας να αναγνωρίσει γραμμή σε άλλο χρώμα.