

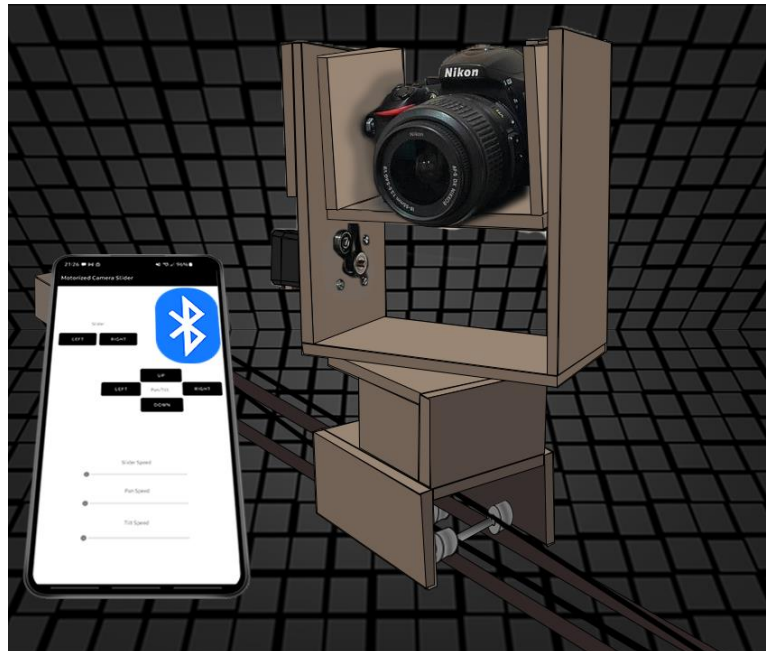


ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ & ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ

Διπλωματική Εργασία

Σχεδίαση και Ανάπτυξη Συστήματος Αυτοματοποιημένου Ελέγχου Κίνησης Ψηφιακής Κάμερας



Φοιτητής: Έρικ Τροφίμενκο
ΑΜ: 50106714

Επιβλέπων Καθηγητής

Διονύσης Κανδρής
Καθηγητής

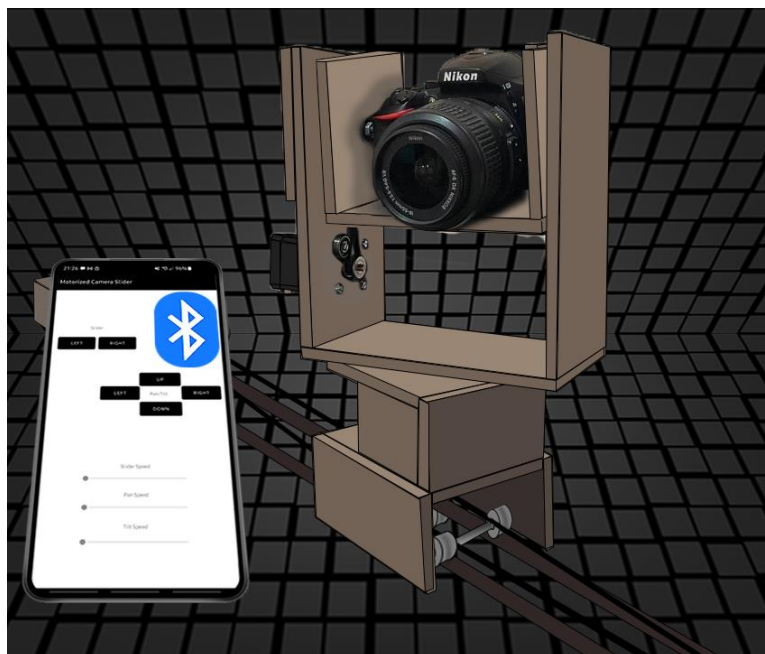
ΑΘΗΝΑ-ΑΙΓΑΛΕΩ, ΟΚΤΩΒΡΙΟΣ 2023



UNIVERSITY OF WEST ATTICA
FACULTY OF ENGINEERING
DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

Diploma Thesis

Design and Development of an Automated Camera Slider System



Student: Erik Trofimenko
Registration Number: 50106714

Supervisor

Dionisis Kandris
Professor

ATHENS-EGALEO, OCTOBER 2023

Η Διπλωματική Εργασία έγινε αποδεκτή και βαθμολογήθηκε από την εξής τριμελή επιτροπή:

Διονύσης Κανδρής, Καθηγητής	Αλέξανδρος Αλεξανδρίδης, Καθηγητής	Ηλίας Ζώης Αναπληρωτής Καθηγητής
(Υπογραφή)	(Υπογραφή)	(Υπογραφή)

Copyright © Με επιφύλαξη παντός δικαιώματος. All rights reserved.

ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ και Τροφίμενκο Έρικ, Οκτώβριος, 2023

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τους συγγραφείς.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον/την συγγραφέα του και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις θέσεις του επιβλέποντος, της επιτροπής εξέτασης ή τις επίσημες θέσεις του Τμήματος και του Ιδρύματος.

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Ο κάτωθι υπογεγραμμένος **Έρικ Τροφίμενκο** του **Βολοντίμιρ**, με αριθμό μητρώου **50106714** φοιτητής/τρια του Πανεπιστημίου Δυτικής Αττικής της Σχολής ΜΗΧΑΝΙΚΩΝ του Τμήματος ΗΛΕΚΤΡΟΛΟΓΩΝ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΜΗΧΑΝΙΚΩΝ,

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 διπλώματός μου.»

Ο Δηλών

Έρικ Τροφίμενκο



Περίληψη

Αυτή η διπλωματική εργασία εστιάζει στην ανάπτυξη και υλοποίηση μιας λειτουργικής πρωτότυπης συσκευής για τον ασύρματο έλεγχο της κίνησης για ψηφιακές κάμερες, η οποία μπορεί να χρησιμοποιηθεί σε εφαρμογές όπως η βιντεογράφιση – κινηματογράφιση και παρακολούθηση απομακρυσμένων αντικειμένων ή ο έλεγχος συστημάτων ασφαλείας. Ειδικότερα, στο πλαίσιο αυτής της διπλωματικής εργασίας πραγματοποιήθηκε η σχεδίαση και υλοποίηση ενός συστήματος κίνησης μιας ψηφιακής κάμερας, η οποία ελέγχεται ασύρματα από μια συσκευή Android μέσω Bluetooth. Το σύστημα χρησιμοποιεί τρεις βηματικούς κινητήρες Nema 17 και τρεις drivers A4988 για τον έλεγχο της κίνησης της κάμερας όπως την περιστροφή της στον οριζόντιο άξονα (pan), στον κατακόρυφο άξονα (tilt) καθώς και τη μετατόπισή της κατά μήκος μιας ράγας. Ο έλεγχος του συστήματος επιτυγχάνεται μέσω ενός μικροελεγκτή ATmega328.

Στη διατριβή αρχικά περιγράφεται η συγκρότηση του υλισμικού και του λογισμικού που απαιτούνται για την υλοποίηση της εφαρμογής, παρέχοντας λεπτομερή ανάλυση των τεχνολογιών που χρησιμοποιήθηκαν και των βημάτων που ακολουθήθηκαν. Στη συνέχεια, αναλύεται η αρχιτεκτονική του συστήματος και η υλοποίηση του σε επίπεδο υλισμικού και λογισμικού. Εξηγείται τη λειτουργία του Bluetooth module και πώς αλληλοεπιδρά με τη συσκευή Android για τον έλεγχο της κίνησης της κάμερας. Επίσης, αναλύεται η λειτουργία του ATmega328 και του κώδικα που χρησιμοποιήθηκε για τον έλεγχο των βηματικών κινητήρων. Τέλος, αναλύονται τα προβλήματα που παρουσιάστηκαν κατά την εκπόνηση του συγκεκριμένου έργου, επεξηγούνται οι διαδικασίες επίλυσής τους, καταγράφονται σχετικές παρατηρήσεις και συμπεράσματα που προέκυψαν και περιγράφονται προτάσεις μελλοντικής εξέλιξης.

Λέξεις – κλειδιά

Ολισθητήρας κάμερας, Κινηματογράφιση, Βιντεογράφιση, Κίνηση κάμερας, Κίνηση κάμερας σε πραγματικό χρόνο, Έλεγχος κάμερας, Αυτοματισμός, Εφαρμογή βηματικού κινητήρα σε παντοκατευθυντική κάμερα, Εφαρμογή Bluetooth για τον έλεγχο κίνησης κάμερας

Abstract

This thesis focuses on the development and implementation of a functional prototype device for wireless motion control for digital cameras, which can be used in applications such as remote object videography and monitoring or security system control. In particular, in the context of this diploma thesis, the design and implementation of a motion system for a digital camera, which is wirelessly controlled by an Android device via Bluetooth, is carried out. The system uses three Nema 17 stepper motors and three A4988 drivers to control the movement of the camera such as its rotation on the horizontal axis (pan), on the vertical axis (tilt) as well as its displacement along a rail. Control of the system is achieved through an ATmega328 microcontroller.

In the document of this thesis at first the construction of the hardware and software required for the implementation of the application is described, providing a detailed analysis of the technologies used and the steps followed. Then, the system architecture and its implementation at the hardware and software level are analyzed. The function of the Bluetooth module and how it interacts with the Android device to control the camera movement is explained. Also, the operation of the ATmega328 and the code used to control the stepper motors is described. Finally, the problems presented during the development of the specific project are analyzed, the procedures for solving them are explained, relevant observations made and conclusions drawn are recorded, and proposals for future development are described.

Keywords

Camera slider, Cinematography, Videography, Camera Movement, Real-time Camera Movement, Camera Control, Automation, Application of a stepper motor in a pan-tilt camera, Application of Bluetooth in camera movement

Περιεχόμενα

Κατάλογος Πινάκων	9
Κατάλογος Εικόνων	9
Αλφαβητικό Ευρετήριο	11
ΕΙΣΑΓΩΓΗ	13
Αντικείμενο της διπλωματικής εργασίας	13
Σκοπός και στόχοι	13
Μεθοδολογία	13
Καινοτομία	14
Δομή	14
1 ΚΕΦΑΛΑΙΟ 1^ο : Κινηματογράφηση και Συστήματα Κίνησης Καμερών	15
1.1 Εισαγωγή Κεφαλαίου 1	15
1.2 Μορφές κίνησης κάμερας	15
1.2.1 Περιστροφή και κλίση – Pan και Tilt.....	16
1.2.2 Μετατόπιση της κάμερας (Dolly).....	16
1.3 Συστήματα Κίνησης Κάμερας	17
1.3.1 Steadicam.....	17
1.3.2 Κινηματογραφικός Γερανός.....	18
1.3.3 Camera Dolly.....	19
1.3.4 Κινηματογραφικός Ολισθητήρας.....	20
2 ΚΕΦΑΛΑΙΟ 2^ο : Τεχνολογίες και Εξοπλισμός ενός Συστήματος Ασυρμάτου Ελέγχου Κίνησης Κάμερας	22
2.1 Εισαγωγή Κεφαλαίου 2	22
2.2 Bluetooth	22
2.2.1 Εφαρμογές.....	23
2.2.2 Αρχιτεκτονική Στοιβάς Πρωτοκόλλου και το Μοντέλο Αναφοράς OSI	24
2.2.3 Αρχιτεκτονική Στοιβάς Πρωτοκόλλου του Bluetooth.....	25
2.2.4 Τύποι συσκευών Bluetooth	28
2.2.5 Τύποι Συνδέσεων.....	29
2.2.6 Σύζευξη και Σύνδεση Συσκευών	29
2.2.7 Ασφάλεια	30
2.3 Bluetooth Module HC-05	31
2.4 Arduino Nano	32
2.4.1 ATmega328	33
2.4.2 Τεχνικά χαρακτηριστικά	33
2.5 Βηματικοί κινητήρες	34
2.5.1 Είδη βηματικών κινητήρων.....	34
2.5.2 Τρόποι οδήγησης ενός βηματικού κινητήρα.....	36
2.5.3 Τύποι βηματικών κινητήρα.....	37
2.5.4 Λειτουργία βηματικών κινητήρα	38
2.5.5 Βηματικός κινητήρας Nema 17.....	40
2.5.6 Οδηγοί βηματικών κινητήρων A4988.....	41
2.5.7 Αποδόσεις του A4988	42
2.5.8 Microstepping	42
2.6 Υλοποίηση του κυκλώματος ελέγχου	44
2.6.1 Υλικά που χρησιμοποιήθηκαν	44

2.6.2	Συνδεσμολογία του συστήματος	46
2.7	Κατασκευή του συστήματος	50
2.7.1	Υλικά που χρησιμοποιήθηκαν	50
2.7.2	Τρόπος κατασκευής	50
3	ΚΕΦΑΛΑΙΟ 3^ο : Διασύνδεση εφαρμογής smartphone με Arduino και Ανάλυση του Κώδικα	55
3.1	Εισαγωγή Κεφαλαίου 3	55
3.2	Υλοποίηση εφαρμογής σε περιβάλλον Android	55
3.3	Υλοποίηση συστήματος Arduino και ο προγραμματισμός του	68
4	ΕΠΙΛΟΓΟΣ	73
4.1	Εισαγωγή Κεφαλαίου 4	73
4.2	Σύνοψη Εργασίας	73
4.3	Προβλήματα και η Επίλυσή τους	73
4.4	Παρατηρήσεις – Συμπεράσματα	74
4.5	Προτάσεις Μελλοντικής Εξέλιξης	74
	Βιβλιογραφία – Αναφορές - Διαδικτυακές Πηγές	75
	Παράρτημα Α	77
	Παράρτημα Β	78
	Παράρτημα Γ	82

Κατάλογος Πινάκων

Πίνακας 2. 1 Πίνακας καταστάσεων μικροβηματισμού	43
--	----

Κατάλογος Εικόνων

Εικόνα 1. 1 Περιστροφή και κλίση κάμερας	16
Εικόνα 1. 2 Μετατόπισή της κάμερας.....	17
Εικόνα 1. 3 Steadicam	18
Εικόνα 1. 4 Κινηματογραφικός γερανός	18
Εικόνα 1. 5 Camera dolly	19
Εικόνα 1. 6 Camera dolly	19
Εικόνα 1. 7 Σύστημα ολισθητήρα κάμερας	21
Εικόνα 2. 1 Λογότυπο Bluetooth	23
Εικόνα 2. 2 Σχεδιάγραμμα Αρχιτεκτονικής στοίβας Μοντέλου Αναφοράς OSI.....	24
Εικόνα 2. 3 Σχεδιάγραμμα Αρχιτεκτονικής στοίβας πρωτοκόλλου Bluetooth.....	28
Εικόνα 2. 4 HC-05 Module for Arduino.....	31
Εικόνα 2. 5 Arduino Nano.....	32
Εικόνα 2. 6 ATmega328P	33
Εικόνα 2. 7 Διάγραμμα ακροδεκτών του Arduino Nano	34
Εικόνα 2. 8 Βηματικός κινητήρας με μόνιμο μαγνήτη	35
Εικόνα 2. 9 Βηματικού κινητήρα μεταβλητού μαγνητισμού.....	35
Εικόνα 2. 10 Υβριδικός βηματικός κινητήρας.....	36
Εικόνα 2. 11 Μονοπολική οδήγηση βηματικού κινητήρα.....	37
Εικόνα 2. 12 Διπολική οδήγηση βηματικού κινητήρα	37
Εικόνα 2. 13 Μονοπολικός βηματικός κινητήρας	38
Εικόνα 2. 14 Διπολικός βηματικός κινητήρας.....	38
Εικόνα 2. 15 Χαρακτηριστική ροπής – βηματισμών βηματικού κινητήρα	40
Εικόνα 2. 16 Βηματικός κινητήρας Nema 17	40

Εικόνα 2. 17 Οδηγός βηματικού κινητήρα A4988	41
Εικόνα 2. 18 Συνδεσμολογία του A4988 με το Nema 17	42
Εικόνα 2. 19 Αναπαράσταση της ομαλότητας της κίνησης σε σχέση με την ανάλυση ^[35]	44
Εικόνα 2. 20 Διάτρητη πλακέτα 60x80mm	44
Εικόνα 2. 21 Μετατροπέας DC-DC Step-Down 5V 1A	45
Εικόνα 2. 22 Μικροδιακόπτης τύπου SPDT	46
Εικόνα 2. 23 Τροφοδοτικό και power jack	46
Εικόνα 2. 24 Κύκλωμα της εφαρμογής.....	47
Εικόνα 2. 25 Συνδεσμολογία του HC-05 με το Arduino	48
Εικόνα 2. 26 Συνδεσμολογία A4988 με το Arduino και τους βηματικούς κινητήρες	48
Εικόνα 2. 27 Συνδεσμολογία A4988 με το Arduino και λοιπών εξαρτημάτων.....	49
Εικόνα 2. 28 Ηλεκτρονικά εξαρτήματα τοποθετημένα σε διάτρητη πλακέτα.....	49
Εικόνα 2. 29 Timing pulleys 60T, 20T, Idler, Ιμάντας και Ρουλεμάν	50
Εικόνα 2. 30 Κάτω μέρος τη βάσης.....	51
Εικόνα 2. 31 Επάνω μέρος τη βάσης	51
Εικόνα 2. 32 Κινητό μέρος περιστροφής (Pan)	52
Εικόνα 2. 33 Κινητό μέρος κλίσης (Tilt)	52
Εικόνα 2. 34 Ένωση κινητών μερών περιστροφής και κλίσης (Pan και Tilt)	53
Εικόνα 2. 35 Μέρος κατασκευής σε λειτουργία	53
Εικόνα 2. 36 Προσομοίωση κατασκευής συστήματος ελέγχου κίνησης κάμερας.....	54
Εικόνα 2. 37 Κατασκευή συστήματος ελέγχου κίνησης κάμερας εν κινήσει.....	54
Εικόνα 3. 1 Διεπαφή Χρήστη	55
Εικόνα 3. 2 Παράθυρο διαλόγου για την ενεργοποίηση του Bluetooth	64
Εικόνα 3. 3 Παράθυρο διαλόγου για τη σύνδεση ασύρματης συσκευής.....	65
Εικόνα 3. 4 Ένδειξη διαδικασίας σύνδεσης ασύρματης συσκευής	65
Εικόνα 3. 5 Παράθυρο συστήματος για τη χορήγηση αδειών χρήσης Bluetooth.....	67
Εικόνα Παραρτήματος Γ 1 Συνδεσμολογία κυκλώματος σε περιβάλλον λογισμικού Fritzing	82

Αλφαβητικό Ευρετήριο

AFH: Adaptive Frequency Hopping
API: Application Programming Interface
AT: Attention Commands
ATMEL: Advanced Technology for Memory and Logic
BLC: Baseband Link Controller
CMOS: Complementary Metal-Oxide-Semiconductor
CNC: Computer Numerical Control Router
CSR: Cambridge Silicon Radio
DC: Direct Current
DIR: Direction
DIY: Do It Yourself
EDR: Enhanced Data Rate
EEPROM: Erasable Programmable Read-Only Memory
GND: Ground
GPS: Global Positioning System
HCI: Host Controller Interface
HID: Human Interface Device
I/O: Input/Output
I2C: Inter-Integrated Circuit
ICSP: In-Circuit Serial Programming
IDE: Integrated Development Environment
IoT: Internet of Things
L2CAP: Logical Link Control and Adaptation Protocol
LC: Link Controller
LED: Light-Emitting Diodes
LMP: Link Manager Protocol
MAC: Media Access Control
MDF: Medium Density Fiberboard
OBEX: Object Exchange
OSI: Open Systems Interconnection Model
P2M: Point to Multipoint
P2P: Point to Point
PIN: Personal Identification Number
PIO: Programmed Input/Output
PWM: Pulse-Width Modulation
RF: Radio Frequency
RFCOMM: Radio Frequency Communication
RPM: Revolutions Per Minute
RST: Reset
RX: Transmitting pin

RXD: Receive Data
SDP: Service Discovery Protocol
SPI: Serial Peripheral Interface
SPP: Serial Port Profile
SRAM: Static Random-Access Memory
TCS: Telephony Control Protocol Specification
TDM: Time Division Multiplexing
TX: Transmitting pin
TXD: Transmit Data
UART: Universal Asynchronous Receiver-Transmitter
UI: User Interface
USART: Universal Synchronous/Asynchronous Receiver/Transmitter
USB: Universal Serial Bus
UUID: Universal Unique Identifier
WAP: Wireless Application Protocol

ΕΙΣΑΓΩΓΗ

Η διπλωματική αυτή εργασία, ερευνά τον σχεδιασμό και την υλοποίηση ενός συστήματος ασύρματης κίνησης κάμερας, καθώς και τον προγραμματισμό για τον απομακρυσμένο έλεγχο της κίνησής της. Εξετάζονται οι τεχνολογικές πτυχές, οι δυνατότητες και η σημασία αυτής της τεχνολογίας στον σύγχρονο κινηματογράφο και η δημιουργία ποιοτικού περιεχομένου.

Αντικείμενο της διπλωματικής εργασίας

Το κυρίως θέμα της εργασίας αφορά τον προσδιορισμό και την ανάπτυξη ενός συστήματος ασύρματης κίνησης κάμερας, καθώς και τον προγραμματισμό ενός μικροελεγκτή για τον ασύρματο έλεγχο της κίνησής της.

Το ενδιαφέρον για αυτό το θέμα αυξάνεται ραγδαία καθώς η ασύρματη κίνηση κάμερας προσφέρει νέες δυνατότητες στη δημιουργία κινηματογραφικών έργων, την ενίσχυση της δημιουργικότητας και την απομάκρυνση των περιορισμών που επιβάλλουν οι παραδοσιακοί μηχανισμοί κίνησης. Επιπλέον, η τεχνολογία αυτή είναι σημαντική στην εξέλιξη του χώρου του κινηματογράφου και ευρύτερα της σύλληψης εικόνας, καθώς επιτρέπει τη δημιουργία εντυπωσιακών και διακριτικών πλάνων, αλλά και τη βελτίωση της παραγωγικότητας σε επαγγελματικές εφαρμογές.

Η επιλογή αυτού του θέματος το καθιστά επίκαιρο καθώς πηγάζει από τον αυξανόμενο ενδιαφέρον και τη σημασία που έχει αποκτήσει η ασύρματη κίνηση κάμερας σε διάφορους τομείς της τεχνολογίας κινηματογράφου, παρακολούθησης εκδηλώσεων και πολλών άλλων εφαρμογών.

Σκοπός και στόχοι

Στόχος αυτής της διπλωματικής εργασίας είναι η ανάπτυξη ενός συστήματος ασύρματης κίνησης κάμερας που θα επιτρέπει στους χρήστες να ελέγχουν την κίνηση της κάμερας από απόσταση, προσφέροντας έτσι νέες δυνατότητες δημιουργίας και κινηματογραφικής έκφρασης. Αυτή η εργασία θα εξετάσει τις τεχνικές, τις τεχνολογίες και τις πρακτικές που σχετίζονται με την ανάπτυξη και την υλοποίηση αυτού του είδους των συστημάτων, καθώς και τις προκλήσεις που αντιμετωπίζονται κατά τη διάρκεια της διαδικασίας ανάπτυξης.

Επιπλέον, θα εξετάσει τις δυνατότητες της ασύρματης κίνησης κάμερας στη δημιουργία εντυπωσιακών παραγωγών και την επίδρασή της στον κόσμο της τέχνης και της ψυχαγωγίας. Τέλος, θα αναλύσει τις προοπτικές και τις μελλοντικές εξελίξεις σε αυτόν τον σημαντικό τομέα της τεχνολογίας κινηματογράφου και εικόνας.

Μεθοδολογία

Η εργασία που εκπονήθηκε διαιρέθηκε σε τρία βασικά στάδια. Στο πρώτο από αυτά πραγματοποιήθηκε η ανάπτυξη του κώδικα για την ασύρματη επικοινωνία ανάμεσα στο smartphone και το Arduino μέσω του Bluetooth. Στο δεύτερο στάδιο, πραγματοποιήθηκε η σύνδεση όλων των εξαρτημάτων του κυκλώματος και η ανάπτυξη του κώδικα που διαχειρίζεται τις λειτουργίες του συστήματος. Το τρίτο και τελευταίο στάδιο συμπεριλάμβανε την πραγματική κατασκευή του συστήματος και την συναρμολόγησή του. Κάθε ένα από αυτά τα στάδια συνέβαλε στην πλήρη υλοποίηση του έργου, καθιστώντας το ένα καινοτόμο και λειτουργικό σύστημα ασύρματου ελέγχου της κίνησης της κάμερας.

Καινοτομία

Το κύριο καινοτόμο χαρακτηριστικό της εργασίας αυτής είναι ο ασύρματος έλεγχος της κάμερας μέσω τεχνολογίας Bluetooth, προσφέροντας ευελιξία και δυνατότητες δημιουργικής λήψης βίντεο. Η εφαρμογή στον κινηματογράφο αποτελεί τον βασικό λόγο για τον οποίο αυτό το θέμα είναι επίκαιρο και σημαντικό, καθώς προσφέρει νέες δυνατότητες για τη δημιουργία πλούσιου κινηματογραφικού περιεχομένου. Η συνδυασμένη χρήση μηχανολογικών στοιχείων, ηλεκτρονικών στοιχείων και ασύρματης επικοινωνίας αποτελεί τον πυρήνα της καινοτομίας σε αυτήν την εργασία, καθιστώντας την αξιόλογη προσφορά στον τομέα της εικονοληψίας.

Δομή

Η διπλωματική αυτή εργασία αποτελείται από τέσσερα κεφάλαια. Το πρώτο κεφάλαιο επικεντρώνεται στην εξέλιξη της κινηματογραφίας και τη σημασία της κίνησης της κάμερας για τη δημιουργία εντυπωσιακών πλάνων. Επιπλέον, αναλύονται οι βασικοί τρόποι ελέγχου της κινηματογραφικής κάμερας. Το δεύτερο κεφάλαιο εστιάζει στις τεχνολογίες που χρησιμοποιήθηκαν για την ανάπτυξη ενός ασύρματου συστήματος ολίσθησης κάμερας και περιγράφει τη διαδικασία κατασκευής. Το τρίτο κεφάλαιο εξετάζει τον κώδικα που απαιτείται για τον έλεγχο του συστήματος από ένα smartphone καθώς και τον κώδικα που διοχετεύει τις εντολές στον μικροελεγκτή Arduino. Το τέταρτο και τελευταίο κεφάλαιο παρουσιάζει μια σύνοψη της εργασίας, αναφέρει τα προβλήματα που προέκυψαν κατά την υλοποίηση του συστήματος και τους τρόπους αντιμετώπισής τους, καθώς και παρέχει παρατηρήσεις, συμπεράσματα και προτάσεις για μελλοντική εξέλιξη.

1 ΚΕΦΑΛΑΙΟ 1^ο : Κινηματογράφηση και Συστήματα Κίνησης Καμερών

1.1 Εισαγωγή Κεφαλαίου 1

Στα πρώτα χρόνια της κινηματογραφικής βιομηχανίας, οι ταινίες γυρίζονταν με μια σταθερή κάμερα και οι σκηνές διαδραματιζόνταν μπροστά από αυτήν. Εκείνη την εποχή οι δυνατότητες καταγραφής της εικόνας ήταν αρκετά περιορισμένες λόγω του μεγάλου βάρους των κινηματογραφικών μηχανών καθώς και του εξοπλισμού τους. Οπότε το καλύτερο που θα μπορούσαν να κάνουν οι κινηματογραφιστές ήταν να ασφαλίζουν τις κάμερες με την τοποθέτησή τους σε τρίποδα ή σε παρόμοια στηρίγματα και να τις αφήνουν να εικονογραφούν σε σταθερό πλαίσιο. Ήταν ένα επιτυχές σύστημα στην εφαρμογή του, όμως μειονεκτούσε στο ότι προσομοίαζε μια θεατρική σκηνή με την θέση του φακού να προσεγγίζει την ιδανική θέση του θεατή στο κοινό, με κοντινά πλάνα όταν ο ηθοποιός προσέγγιζε τον φακό και μακρινά πλάνα όταν απομακρυνόταν από αυτόν.

Με τον καιρό οι κινηματογραφιστές ανακάλυπταν νέες τεχνικές, οι διαστάσεις και το βάρος της κάμερας να μειώνεται κάνοντάς την ολοένα και πιο κινητή, με τον θεατή να απεγκλωβίζεται από την ιδέα του σταθερού πλάνου και να πηγαίνει σε μια νέα πραγματικότητα. Η εξέλιξη της τεχνολογίας επέτρεπε στους κινηματογραφιστές να δημιουργούν ένα πιο δυναμικό αποτέλεσμα, καθώς τους έδινε τη δυνατότητα να ακολουθούν τους ηθοποιούς και να καταγράφουν πιο εντυπωσιακές σκηνές.

Η κίνηση της κάμερας έχει γίνει ένα απαραίτητο εργαλείο για τους κινηματογραφιστές όχι μόνο για να καταγράφουν κινούμενες εικόνες αλλά και για να δημιουργούν εντυπωσιακά και αποτελεσματικά πλάνα. Καθώς μπορεί να δώσει έμφαση σε ένα συγκεκριμένο αντικείμενο ή ηθοποιό, να δημιουργήσει ατμόσφαιρα και να μεταδώσει συναισθήματα στο κοινό.

Ως από τη φύση του, ο άνθρωπος εθίζεται πολύ εύκολα, οι θεατές άρχισαν να απαιτούν όλο και περισσότερα από τον κινηματογράφο, δημιουργώντας έτσι το ζήτημα της οπτική διέγερσης. Γρήγορα, η καινοτομία αυτή μάγεψε τους θεατές ανεβάζοντας τα επίπεδα αδρεναλίνης, μεταδίδοντας συναισθήματα και δημιουργώντας μια νέα εμπειρία εντελώς διαφορετική. Η κίνηση της κάμερας έγινε ένα από τα μεγαλύτερα τεχνάσματα το οποίο ενεργοποιεί τους θεατές και διατηρεί το ενδιαφέρον τους ^[1].

1.2 Μορφές κίνησης κάμερας

Στην κινηματογράφηση εφαρμόζονται διάφορες τεχνικές λήψης σκηνών. Η κίνηση είναι ένας σημαντικός παράγοντας ο οποίος πρέπει να ληφθεί υπόψιν από έναν κινηματογραφιστή και διακρίνεται σε τρία βασικά είδη: στην κίνηση του αντικειμένου, στην κίνηση της κάμερας και στην κίνηση του φωτισμού. Οι κινηματογραφιστές από την αρχή της τέχνης του κινηματογράφου δοκίμαζαν διάφορους τρόπους για την κίνηση της κάμερας, όπως η τοποθέτησή της σε διάφορες κινητήριες πλατφόρμες, όπως οχήματα. Επίσης, χρησιμοποιούν ειδικό εξοπλισμό για την ομαλή εικονογράφηση κατά τη κίνηση του αντικειμένου.

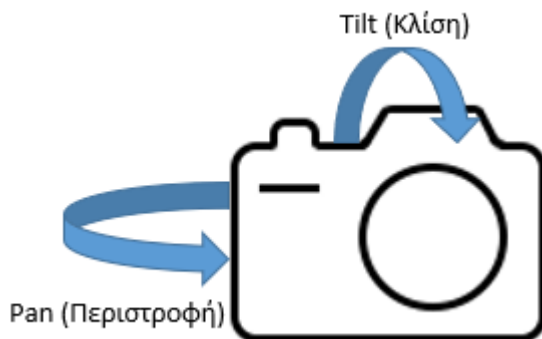
1.2.1 Περιστροφή και κλίση – Pan και Tilt

Οι πιο βασικές κινήσεις της κάμερας είναι η περιστροφή της (pan) και η κλίση (tilt).

Η κίνηση περιστροφής της κάμερας ή αλλιώς pan, ως συντομία της πανοραμικής κίνησης, αναφέρεται στην οριζόντια κίνηση της κάμερας δεξιόστροφα ή αριστερόστροφα ^[2]. Κατά τη διάρκεια αυτής της κίνησης, η κάμερα παραμένει σταθερή ως προς τον ορίζοντα, αλλά κινείται από τη μία πλευρά στην άλλη ^[3]. Η κίνηση περιστροφής επιτρέπει στον σκηνοθέτη να αποκαλύψει νέους χαρακτήρες ή αντικείμενα που βρίσκονται εκτός του πεδίου θέασης της κάμερας και να αλλάξει την κατεύθυνση της προβολής.

Από την άλλη πλευρά, η κίνηση κλίσης αναφέρεται στην κατακόρυφη κίνηση της κάμερας πάνω ή κάτω. Κατά τη διάρκεια αυτής της κίνησης, η κάμερα αλλάζει γωνία προβολής και μπορεί να δείξει πράγματα που βρίσκονται σε υψηλότερο ή χαμηλότερο επίπεδο από το επίπεδο της κάμερας. Η κίνηση αυτή συνήθως χρησιμοποιείται για να αναδείξει σημαντικά στοιχεία σε μια σκηνή ή να προκαλέσει ένταση ή αγωνία. Η κλίση της κάμερας χρησιμοποιείται συνήθως για να δείξει το ύψος ενός αντικειμένου ή τοποθεσίας του σε σχέση με τον χαρακτήρα ή το θέμα του πλάνου. Μπορεί επίσης να χρησιμοποιηθεί για να δείξει μια αλλαγή στην κατάσταση του χαρακτήρα, όπως τον προβληματισμό ή την ενθουσιώδη διάθεση του. ^[4]

Και οι δύο αυτές κινήσεις επιτρέπουν στον κινηματογραφιστή να δώσει βάθος και διάσταση στην εικόνα, καθώς μπορεί να αλλάξει τη γωνία λήψης της κάμερας και να προσαρμόζει το πλάνο της σκηνής στις ανάγκες του σκηνοθέτη. Αυτή η μορφή κίνησης της κάμερας προσομοιάζει την κίνηση του κεφαλιού και των ματιών του θεατή, επομένως του δίνει άμεσα φυσική αίσθηση.



Εικόνα 1. 1 Περιστροφή και κλίση κάμερας

1.2.2 Μετατόπιση της κάμερας (Dolly)

Η μετατόπιση της κάμερας, γνωστή και ως Dolly, είναι μια κίνηση κάμερας όπου η κάμερα κινείται προς τα εμπρός ή προς τα πίσω σε μια ευθεία υπό κλίση. Αυτή η κίνηση δημιουργεί ένα εφέ που μας δείχνει ότι ο θεατής πλησιάζει ή απομακρύνεται από το θέμα του πλάνου ^[4]. Η κάμερα τοποθετείται σε μια συσκευή Dolly, η οποία είναι μια πλατφόρμα με ρόδες που μπορούν να κινηθούν ομαλά πάνω στο έδαφος. Μια άλλη επιλογή είναι η χρήση ειδικών στεφανιών, γνωστών ως tracks, πάνω στα οποία τοποθετείται η συσκευή Dolly. Η μετατόπιση της κάμερας μπορεί να γίνει ομαλά και αργά ή γρήγορα, ανάλογα με την επιθυμητή επίδραση και το μήκος της κίνησης που θέλει ο κινηματογραφιστής να αποτυπώσει στο πλάνο. Η κίνηση της κάμερας σε συνδυασμό

με την κίνηση του θέματος του πλάνου δημιουργεί μια εντυπωσιακή εικόνα και μπορεί να δώσει μια διαφορετική αίσθηση κίνησης στο κοινό.



Εικόνα 1. 2 Μετατόπιση της κάμερας

Η μετατόπιση της κάμερας χρησιμοποιείται συχνότερα από την εστίαση (zoom) από τους κινηματογραφιστές όταν ακολουθούν το αντικείμενο της λήψης. Ο λόγος είναι ότι κατά τη μετατόπιση μεταβάλλεται η απόσταση από το αντικείμενο και την κάμερα ενώ κατά το ζουμάρισμα η εστιακή απόσταση του φακού. Κατά την εστίαση δεν αλλάζει το πλάνο λήψης, αλλά μεταβάλλεται το κατά πόσο φαίνεται το πλάνο, ενώ κατά την μετατόπιση, λόγω του ότι η κάμερα μεταβάλλει θέση φανερώνεται περισσότερο από τη σκηνή. Η λήψη με τη μέθοδο της εστίασης τείνει να είναι αρκετά μονότονη λόγω του ότι δεν αλλάζει η οπτική γωνία της λήψης καθώς η κάμερα βρίσκεται σε σταθερό σημείο. Για αυτό προτιμάται, συνήθως, η μέθοδος της μετατόπισης για να προσδώσει ζωντάνια στη λήψη. Παρ' όλα αυτά η επιλογή ανάμεσα στις δυο αυτές μεθόδους είναι αισθητικό θέμα [\[2\]](#) [\[3\]](#) [\[4\]](#).

1.3 Συστήματα Κίνησης Κάμερας

Παρακάτω περιγράφονται τα πιο δημοφιλή συστήματα κίνησης κάμερας που χρησιμοποιούνται στον κινηματογράφο. Ο κινηματογράφος είναι ένας τομέας που στηρίζεται σε συνεχείς καινοτομίες στην τεχνολογία εικόνας και ήχου, και τα συστήματα κίνησης κάμερας αποτελούν ένα καίριο εργαλείο για τη δημιουργία εντυπωσιακών πλάνων. Από τις παραδοσιακές μηχανικές κινήσεις μέχρι την προηγμένη ψηφιακή τεχνολογία, αυτά τα συστήματα προσφέρουν ευελιξία και δυνατότητα δημιουργίας μοναδικών πλάνων που εντυπωσιάζουν το κοινό.

1.3.1 Steadicam

Ένα από τα πιο γνωστά συστήματα κίνησης κάμερας που χρησιμοποιούν τις κινήσεις που αναφέρθηκαν στην προηγούμενη υποενότητα είναι το Steadicam. Το Steadicam είναι ένα σύστημα σταθεροποίησης κάμερας το οποίο επιτρέπει στους χρήστες να κινούνται ελεύθερα γύρω από τους ηθοποιούς ή την σκηνή και να καταγράφουν ομαλά και σταθερά βίντεο. Το σύστημα αυτό εφαρμόζεται επάνω στο χρήστη σε μορφή γιλέκου και αποτελείται από βάρη τα οποία τοποθετούνται αντίθετα του βάρους της κάμερας για την σταθεροποίησή της, στυλό ή αλλιώς Gimbal - μια κινητή μονάδα που κρατά τη βάση και επιτρέπει στον χρήστη να περιστρέφει την κάμερα. Αυτό το σύστημα αποτελεί ένα εξαιρετικά χρήσιμο εργαλείο για τη δημιουργία ομαλών και σταθερών πλάνων σε κινηματογραφικές και τηλεοπτικές παραγωγές, επιτρέποντας στους κινηματογραφιστές να καταγράφουν δυναμικές σκηνές με εντυπωσιακή σταθερότητα. [\[5\]](#) [\[6\]](#)



Εικόνα 1. 3 Steadicam

1.3.2 Κινηματογραφικός Γερανός

Άλλο ένα σύστημα κίνησης της κάμερας είναι ο κινηματογραφικός γερανός ή αλλιώς camera crane, ένα σύστημα το οποίο συνδυάζει τις κινήσεις περιστροφής και κλίσης για να παρέχει ομαλές λήψεις από διάφορες γωνίες και ύψη.



Εικόνα 1. 4 Κινηματογραφικός γερανός

Ο κινηματογραφικός γερανός είναι ένα σύστημα που χρησιμοποιείται στη βιομηχανία του κινηματογράφου για τη λήψη πλάνων από ψηλά ή από διαφορετικές γωνίες. Αποτελείται από μια μεγάλη βάση στήριξης, στην οποία συνδέεται μια μακριά μπάρα (επίσης ονομάζεται και boom) με τοποθετημένη στην άκρη της μια κάμερα. Ο κινηματογραφικός γερανός είναι συνήθως σταθερός στο έδαφος, αλλά μπορεί επίσης να τοποθετηθεί σε οχήματα ή σε πλατφόρμες για τη λήψη εντυπωσιακών πλάνων. ^[7]

Οι κινήσεις του γερανού ελέγχονται από έναν οπερατέρ, εκτελούνται πάνω από το έδαφος και συνήθως περιλαμβάνουν την κίνηση ανύψωσης ή κατεβάσματος ^[8] της κάμερας καθώς και κίνηση προς τα αριστερά ή τα δεξιά, κίνηση προς τα εμπρός ή προς τα πίσω και περιστροφή της κάμερας γύρω από τον άξονά της. Οι κινήσεις αυτές μπορούν να είναι απαλές και μετρημένες για τη δημιουργία ενός συγκεκριμένου αισθητικού αποτελέσματος στην εικόνα.

Οι κινηματογραφικοί γερανοί είναι διαθέσιμοι σε διάφορα μεγέθη και σχέδια, ανάλογα το φορτίο και τις ανάγκες της συγκεκριμένης παραγωγής ^[9]. Οι μεγαλύτεροι γερανοί μπορούν να φτάσουν ύψος έως και 30 μέτρα και να κινηθούν σε εκτεταμένες διαδρομές.

Σημαντικό πλεονέκτημα των κινηματογραφικών γερανών είναι η δυνατότητα να κινούνται ελεύθερα στον αέρα, χωρίς να χρειάζεται να τοποθετηθούν σε ένα συγκεκριμένο σημείο ή

επιφάνεια, όπως συμβαίνει με τους τρίποδες ή τους σταθερούς γεραμούς. Αυτό επιτρέπει στους κινηματογραφιστές να δημιουργούν εντυπωσιακές λήψεις από ψηλά, όπως για παράδειγμα τη λήψη μιας πόλης από ψηλά, μιας μεγάλης σκηνής μάχης ή μιας μεγάλης προσέγγισης σε ένα αντικείμενο. Επίσης, οι κινηματογραφικοί γεραμοί μπορούν να συνδυαστούν με άλλα συστήματα κίνησης, όπως sliders ή τροχούς, για να δημιουργηθούν ακόμη πιο εντυπωσιακές λήψεις.

1.3.3 Camera Dolly

Ένα άλλο δημοφιλές σύστημα είναι το camera dolly. Είναι ένα σύστημα το οποίο είναι τοποθετημένο είτε πάνω σε ράγες, είτε ελεύθερο με τροχούς, το οποίο επιτρέπει τη μετατόπιση της κάμερας μπροστά και πίσω καθώς και αριστερά και δεξιά. Αυτό το σύστημα επιτρέπει στους χρήστες να δημιουργήσουν ομαλές κινήσεις και πλάνα ενώ κινούνται κοντά στο θέμα. [\[10\]](#) [\[11\]](#)



Εικόνα 1. 5 Camera dolly [\[12\]](#)



Εικόνα 1. 6 Camera dolly [\[12\]](#)

Το σύστημα αυτό συνήθως χρησιμοποιείται σε κινηματογραφικές σκηνές με μεγάλο πλάνο, όπου απαιτούνται ομαλές κινήσεις για τη λήψη του πλάνου.

Απαιτεί εξειδικευμένο προσωπικό και εξοπλισμό ^[12], όπως κινητήρες, πλατφόρμα διανομής βάρους (weight distribution platform) για να διανείμει το βάρος της κάμερας και του εξοπλισμού στο Dolly, ένα ειδικό σύστημα πέδησης (braking system) για ακρίβεια στην παραμονή του Dolly στο επιθυμητό σημείο, κάθετους στύλους (vertical columns) για την ανύψωση της κάμερας σε μεγαλύτερα ύψη και άλλα συστήματα τα οποία βοηθούν στην ομαλή κίνηση της κάμερας και στην αποφυγή κραδασμών κατά τη διάρκεια της λήψης.

Σε ορισμένες περιπτώσεις, οι κινηματογραφιστές χρησιμοποιούν το Dolly σε συνδυασμό με άλλα συστήματα όπως ολισθητήρες και γερανούς, για να δημιουργήσουν πιο πολύπλοκες κινήσεις κάμερας ^[6].

1.3.4 Κινηματογραφικός Ολισθητήρας

Ο κινηματογραφικός ολισθητήρας, ή γνωστός ως slider, είναι ένα σύστημα, το οποίο επιτρέπει στην κάμερα να κινείται οριζόντια πάνω σε μια ομαλή επιφάνεια.

Η ολίσθηση της κάμερας με χρήση slider είναι μια συνηθισμένη μέθοδος κίνησης της κάμερας, η οποία επιτρέπει στον κινηματογραφιστή να παρακολουθεί ένα αντικείμενο ή να εξερευνά το χώρο καθώς κινείται σε μια σκηνή ^[13]. Αυτό επιτυγχάνεται με τη χρήση ενός ολισθητήρα, ενός τραπεζοειδούς συστήματος, τοποθετημένο σε τρίποδα ή μη ^[6], το οποίο κινείται παράλληλα με την επιφάνεια της κάμερας.

Οι ολισθητήρες μπορούν να χρησιμοποιηθούν για να παρακολουθήσουν ένα αντικείμενο που κινείται στο πλάνο της κάμερας, είτε αυτό είναι σταθερό είτε κινούμενο. Η κίνηση του ολισθητήρα επιτρέπει στον κινηματογραφιστή να αλλάζει το εστιακό μήκος της φακού, προσαρμόζοντας την εστίαση της κάμερας στην κίνηση του αντικειμένου.

Η χρήση των ολισθητήρων συνήθως απαιτεί εξειδικευμένο εξοπλισμό και ομάδα κινηματογραφιστών για να επιτευχθεί η ιδανική κίνηση της κάμερας. Ωστόσο, οι τεχνολογικές εξελίξεις έχουν διευκολύνει τη χρήση τους με μικρότερο εξοπλισμό και με μεγαλύτερη ακρίβεια στον έλεγχο της κίνησης. Τα συστήματα ολισθητήρων που χρησιμοποιούνται σήμερα στην κινηματογραφική βιομηχανία δίνουν τη δυνατότητα στον χρήστη να ελέγχει την κίνηση της κάμερας με μεγάλη ακρίβεια και σταθερότητα ^[14].

Στην πραγματικότητα, υπάρχουν πλέον διαφορετικοί τύποι ολισθητήρων, ανάλογα με τη χρήση και τον τομέα που απευθύνονται. Για παράδειγμα, υπάρχουν ολισθητήρες που χρησιμοποιούνται για καταγραφή βίντεο και άλλοι που χρησιμοποιούνται για φωτογραφίες, ενώ υπάρχουν και ολισθητήρες που προορίζονται για επαγγελματική χρήση και άλλα για ερασιτεχνική.

Τα συστήματα ολισθητήρων μπορούν να χρησιμοποιηθούν σε διάφορες συνθήκες και διάφορα περιβάλλοντα. Για παράδειγμα, μπορούν να χρησιμοποιηθούν σε εξωτερικούς χώρους και σε συνδυασμό με τις προηγμένες τεχνολογίες κατασκευής και την ανάπτυξη της ασύρματης τεχνολογίας, οι ολισθητήρες έχουν γίνει ευρέως διαθέσιμοι σε πολλές διαφορετικές μορφές και μεγέθη, από μικρούς ολισθητήρες που μπορούν να χρησιμοποιηθούν με μια μόνο κάμερα έως μεγάλα ολισθητήρες που μπορούν να χρησιμοποιηθούν με ολόκληρο το σύστημα

κινηματογραφικής παραγωγής. Οι ολισθητήρες είναι επίσης ιδανικοί για χρήση σε στενούς εσωτερικούς χώρους και σε συνθήκες που απαιτούν αθόρυβη λειτουργία, καθώς μπορούν να λειτουργήσουν με αθόρυβους ηλεκτρικούς κινητήρες και να προσφέρουν σταθερή και ομαλή κίνηση [\[15\]](#).



Εικόνα 1. 7 Σύστημα ολισθητήρα κάμερας

2 ΚΕΦΑΛΑΙΟ 2^ο : Τεχνολογίες και Εξοπλισμός ενός Συστήματος Ασύρματου Ελέγχου Κίνησης Κάμερας

2.1 Εισαγωγή Κεφαλαίου 2

Σε αυτό το κεφάλαιο παρουσιάζεται ο τρόπος κατασκευής ενός συστήματος ασύρματου ελέγχου κίνησης κάμερας, το οποίο επιτρέπει στον χρήστη να ελέγχει την κίνηση μιας κάμερας ασύρματα, χρησιμοποιώντας μια συσκευή smartphone με λογισμικό σύστημα Android. Περιγράφονται επίσης ο εξοπλισμός και τα υλικά που χρησιμοποιήθηκαν για την κατασκευή και αναλύεται ο τρόπος ανάπτυξης μιας εφαρμογής για τον έλεγχο του συστήματος από τη συσκευή smartphone και του προγραμματισμού του μικροελεγκτή ο οποίος θα λαμβάνει ασύρματα τις εντολές του χρήστη και θα τις μετατρέπει σε κίνηση. Επίσης, σε αυτό το κεφάλαιο αναλύονται οι διάφορες φάσεις της κατασκευής και του προγραμματισμού του συστήματος, καθώς και οι τεχνολογίες που χρησιμοποιήθηκαν για την υλοποίησή του.

2.2 Bluetooth

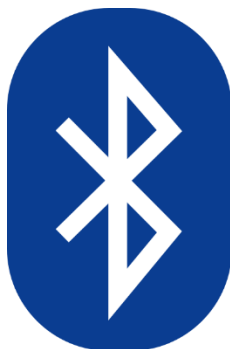
Η τεχνολογία Bluetooth είναι μια ασύρματη τεχνολογία που χρησιμοποιείται για την ανταλλαγή δεδομένων μεταξύ συσκευών. Η τεχνολογία αυτή δημιουργήθηκε από την Ericsson Mobile Communications το 1994 όταν αυτή ξεκίνησε τη μελέτη εναλλακτικών τρόπων επικοινωνίας καταργώντας τον παραδοσιακό ενσύρματο τρόπο επικοινωνίας. Αρχικά χρησιμοποιήθηκε για να συνδέσει μια συσκευή κινητής τηλεφωνίας με μια ακουστική συσκευή. Από τότε, η τεχνολογία Bluetooth έχει εξελιχθεί και χρησιμοποιείται ευρέως σε πολλούς τομείς όπως στην κινητή τηλεφωνία, στην αυτοκινητοβιομηχανία, στην υγεία, στα οικιακά ηλεκτρονικά και στη ψυχαγωγία. ^[16]

Την εποχή της ανάπτυξης της τεχνολογίας Bluetooth επικρατούσε η τεχνολογία των υπέρυθρων για την ασύρματη επικοινωνία μεταξύ συσκευών η οποία είχε ως κυρίαρχο μειονέκτημα ότι απαιτούσε οπτική επαφή μεταξύ αποστολέα και παραλήπτη. Αυτό είχε ως αποτέλεσμα την ανάγκη της ανάπτυξης μιας ασύρματης τεχνολογίας με χρήση ραδιοκυμάτων για τον λόγο ότι τα ραδιοκύματα δεν είναι κατευθυντικά και δεν απαιτούν οπτική επαφή μεταξύ αποστολέα και παραλήπτη.

Οι προδιαγραφές του Bluetooth εφόσον στόχευαν την αντικατάσταση των καλωδίων, ήταν απαραίτητο να είναι χαμηλού κόστους, υψηλής αξιοπιστίας και ευρέως διαθέσιμο. Για τον λόγο ότι αντικαθιστούσε τα καλώδια, θα έπρεπε αναμφισβήτητα, να είναι φθηνότερο της εναλλακτικής του καλωδίου και μικρό σε μέγεθος. Δεδομένου ότι είναι ασύρματη τεχνολογία, σχεδιάστηκε να λειτουργεί με μπαταρία, οπότε ήταν ανάγκη να είναι επίσης εξαιρετικά χαμηλής ισχύος και να λειτουργεί σε χαμηλές τάσεις, έτσι ώστε να μην καταναλώνει πολλή ενέργεια και να διαρκεί όσο δυνατόν γίνεται η μπαταρία. Τέλος, έπρεπε να είναι μικρό σε μέγεθος και ελαφρύ ώστε να μην δυσχεραίνει τη σχεδίαση των συσκευών στις οποίες θα είναι ενσωματωμένο. ^[17]

Η συγκεκριμένη τεχνολογία βασίζεται σε ένα ασύρματο πρωτόκολλο που επιτρέπει τη μετάδοση δεδομένων μεταξύ συσκευών σε κοντινή απόσταση, συνήθως μέχρι 10 μέτρα, φτάνοντας και το μέγιστο των 100 μέτρων σε ανοιχτό χώρο και σε ιδανικές συνθήκες ^{[16] [18]} .

Είναι μια τεχνολογία εύκολη ως προς τη χρήση και είναι ένα από τα πιο δημοφιλή πρωτόκολλα ασύρματης επικοινωνίας στον κόσμο και διαθέτει αυξημένη δυνατότητα επικοινωνίας μεταξύ συσκευών, σε σχέση με άλλες τεχνολογίες ασύρματης επικοινωνίας. Παρόλο που οι ταχύτητες του Bluetooth δεν είναι πολύ υψηλές, η τεχνολογία αυτή είναι αρκετά γρήγορη για αρκετές εφαρμογές. Τέλος, προσφέρει υψηλά επίπεδα ασφάλειας κατά την ασύρματη μεταφορά δεδομένων καθώς χρησιμοποιεί κρυπτογράφηση για την προστασία των δεδομένων [\[18\]](#).



Εικόνα 2. 1 Λογότυπο Bluetooth

2.2.1 Εφαρμογές

Οι εφαρμογές της τεχνολογίας Bluetooth καλύπτουν ένα ευρύ φάσμα πεδίων και παρέχουν ασύρματες λύσεις για ποικίλες ανάγκες επικοινωνίας και ελέγχου μεταξύ συσκευών. Αυτές οι εφαρμογές προσφέρουν ανεξαρτησία από τα καλώδια και δημιουργούν νέες ευκαιρίες για την απλοποίηση της καθημερινής ζωής. Παρακάτω περιγράφονται μερικές από τις δημοφιλείς εφαρμογές της τεχνολογίας Bluetooth.

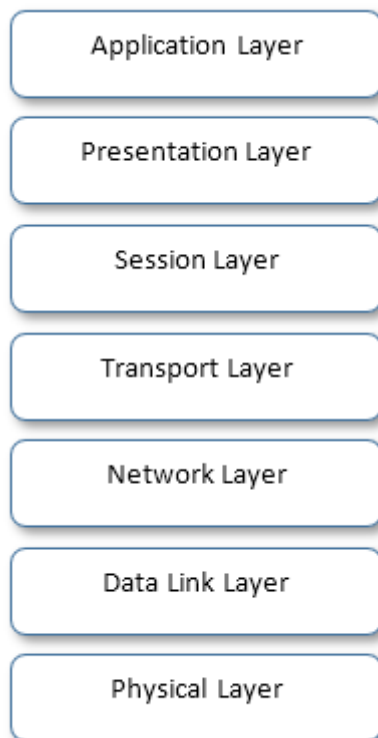
1. **Ασύρματα Ακουστικά:** Οι ασύρματες συσκευές ακουστικών χρησιμοποιούν την τεχνολογία Bluetooth για να προσφέρουν κρυστάλλινο ήχο και ελευθερία κίνησης. Οι χρήστες μπορούν να ακούν μουσική, να πραγματοποιούν κλήσεις και να απολαμβάνουν πολυμέσα χωρίς την ανάγκη καλωδίων.
2. **Ασύρματα συστήματα αυτοκινήτου:** Τα συστήματα ψυχαγωγίας στο αυτοκίνητο χρησιμοποιούν το Bluetooth για να συνδέονται με τα κινητά τηλέφωνα. Αυτό επιτρέπει στους οδηγούς να πραγματοποιούν κλήσεις hands-free και να ακούν μουσική ασφαλώς εν κινήσει.
3. **Εφαρμογές IoT (Διαδίκτυο των Πραγμάτων):** Η τεχνολογία Bluetooth είναι κρίσιμη για το IoT. Συνδέει έξυπνες συσκευές στο σπίτι, όπως θερμοστάτες, φώτα, και κλειδαριές, επιτρέποντάς τις να αλληλοεπιδρούν και να διαμορφώνουν ένα έξυπνο σύστημα.
4. **Συσκευές Υγείας και Fitness:** Αισθητήρες υγείας και συσκευές fitness χρησιμοποιούν το Bluetooth για τη μετάδοση δεδομένων, όπως καρδιακός παλμός και βήματα, σε κινητά τηλέφωνα ή άλλες πλατφόρμες υγείας.
5. **Συσκευές Εισόδου/Εξόδου:** Τα ασύρματα πληκτρολόγια, ποντίκια και game controllers συνδέονται με υπολογιστές, τηλεοράσεις και κονσόλες βιντεοπαιχνιδιών μέσω Bluetooth.

6. Έλεγχος Οικιακού Αυτοματισμού: Η τεχνολογία Bluetooth επιτρέπει στους χρήστες να ελέγχουν φώτα, θερμοστάτες, και άλλες συσκευές στο σπίτι από απόσταση μέσω smartphone ή tablet.
7. Συσκευές Εντοπισμού: Τα κλειδιά, η τσάντα, και άλλα αντικείμενα μπορούν να εξοπλιστούν με Bluetooth trackers για να τα εντοπίζουμε πιο εύκολα.

Οι εφαρμογές της τεχνολογίας Bluetooth συνεχώς εξελίσσονται και επεκτείνονται, προσφέροντας ποικίλες λύσεις για την εύκολη και ασύρματη επικοινωνία μεταξύ συσκευών. [\[19\]](#)

2.2.2 Αρχιτεκτονική Στοίβας Πρωτοκόλλου και το Μοντέλο Αναφοράς OSI

Η αρχιτεκτονική στοίβας πρωτοκόλλου αποτελείται από ένα σύνολο πρωτοκόλλων που χρησιμοποιούνται στο διαδίκτυο και σε άλλα δίκτυα επικοινωνίας. Η στοίβα αυτή περιλαμβάνει διάφορα επίπεδα πρωτοκόλλων, τα οποία επιτρέπουν την επικοινωνία μεταξύ των διαφορετικών συσκευών και εφαρμογών. Κάθε επίπεδο παρέχει συγκεκριμένες λειτουργίες και υπηρεσίες, και τα διαφορετικά επίπεδα συνεργάζονται μεταξύ τους για την αποστολή και τη λήψη των δεδομένων. Έχει σχεδιαστεί έτσι ώστε να εξασφαλίζεται η αποτελεσματική και αξιόπιστη επικοινωνία μεταξύ των συσκευών, με την χρήση διαφορετικών πρωτοκόλλων σε κάθε επίπεδο. Η στοίβα πρωτοκόλλου βοηθά στην ανάπτυξη συμβατών συστημάτων επικοινωνίας, ενώ επίσης επιτρέπει την ανεξαρτησία των συσκευών και των εφαρμογών από τις λεπτομέρειες της υλοποίησης της επικοινωνίας στο δίκτυο. [\[20\]](#)



Εικόνα 2. 2 Σχεδιάγραμμα Αρχιτεκτονικής στοίβας Μοντέλου Αναφοράς OSI

Το **Physical Layer** ή **Φυσικό Επίπεδο** είναι υπεύθυνο για την ηλεκτρική διεπαφή με το επικοινωνιακό μέσο, το οποίο συμπεριλαμβάνει τη διαμόρφωση και η κωδικοποίηση του καναλιού. Συνεπώς, αυτό καλύπτει το Radio και τμήμα του Baseband.

Το **Data Link Layer** ή **Επίπεδο Σύνδεσης Δεδομένων** είναι υπεύθυνο για την κατασκευή και την εκπομπή των πλαισίων, ορισμό της αρχής και του τέλους κάθε πλαισίου, την ανίχνευση και επιδιόρθωση σφαλμάτων μετάδοσης, έλεγχο για την αποφυγή ταυτόχρονης εκπομπής των πλαισίων και τη μεταβολή στη ροή των πλαισίων αναλόγως των ρυθμών του παραλήπτη.

Το **Network Layer** ή **Επίπεδο Δικτύου** είναι υπεύθυνο για τη δρομολόγηση πακέτων από δίκτυο σε δίκτυο. Επιλέγει και ελέγχει την αξιοπιστία των διαδρομών διάφορων δρομολογητών (routers) προκειμένου ένα πακέτο να φτάσει στον τελικό προορισμό του.

Το **Transport Layer** ή **Επίπεδο Μεταφοράς** εξασφαλίζει τη αξιόπιστη διανομή δεδομένων και απαλλαγμένη σφάλματα, απώλειες πακέτων ή διπλοτυπίες. Επίσης είναι υπεύθυνο για τη βελτιστοποίηση της αξιοπιστίας και ποιότητας των υπηρεσιών του δικτύου.

Το **Session Layer** ή **Επίπεδο Συνόδου** παρέχει τη διαχείριση των υπηρεσιών ελέγχου ροής δεδομένων τα οποία ελέγχονται στο επίπεδο L2CAP και μέρος του επιπέδου RFCOMM/SDP στο μοντέλο του Bluetooth.

Το **Presentation Layer** ή **Επίπεδο Παρουσίασης** μετασχηματίζει διαφορετικές μορφές δεδομένων διαφορετικών κόμβων, ώστε να μπορούν να αναπαρασταθούν σε τυπική μορφή στο επίπεδο εφαρμογής. Αυτή είναι, επίσης, η κύρια διεργασία του επιπέδου RFCOMM/SDP στο μοντέλο του Bluetooth.

Τέλος, το **Application Layer** ή **Επίπεδο Εφαρμογής** είναι υπεύθυνο για τη διαχείριση της επικοινωνίας μεταξύ κεντρικών εφαρμογών. ^[20]

2.2.3 Αρχιτεκτονική Στοίβας Πρωτοκόλλου του Bluetooth

Το κύριο χαρακτηριστικό της τεχνολογίας Bluetooth είναι ότι δίνει τη δυνατότητα σε συσκευές διαφορετικών κατασκευαστών να συνδέονται με ευκολία μεταξύ τους και να συνεργάζονται. Το Bluetooth επίσης ορίζει μια στοίβα λογισμικού με την οποία επιτρέπει στις εφαρμογές να αναζητήσει άλλες συσκευές, να καταλαβαίνει τι υπηρεσίες μπορούν να παρέχουν και να τις χρησιμοποιήσουν.

Η αρχιτεκτονική στοίβας πρωτοκόλλου Bluetooth ορίζεται ως μια σειρά η οποία αποτελείται από επίπεδα. Κάποια από τα χαρακτηριστικά προσπερνούν μερικά επίπεδα. Κάθε επίπεδο του παρακάτω σχεδιαγράμματος αναπαρίσταται από μπλοκ. ^[17]

Το επίπεδο **Bluetooth Radio** είναι το χαμηλότερο στρώμα της στοίβας Bluetooth και είναι υπεύθυνο για τη φυσική μετάδοση των δεδομένων μέσω του ασύρματου καναλιού. Αυτό το στρώμα αναλαμβάνει τον έλεγχο της συχνότητας του ραδιοκυματισμού, του ισχύος του σήματος και της διαμόρφωσης του σήματος που θα μεταδοθεί μέσω του ασύρματου καναλιού. Σε αυτό το επίπεδο, οι συσκευές Bluetooth ανταλλάσσουν πακέτα δεδομένων μέσω ραδιοκυμάτων σε συχνότητα

2,4 GHz. Αυτό το στρώμα περιλαμβάνει το αναλογικό και ψηφιακό κύκλωμα ασύρματου ελέγχου και το κύκλωμα συχνότητας του Bluetooth, το οποίο διαμορφώνει το σήμα που θα μεταδοθεί μέσω του ασύρματου καναλιού.

Το επίπεδο **Baseband Link Controller** (BLC) είναι το δεύτερο στρώμα της στοίβας Bluetooth και είναι υπεύθυνο για τη διαχείριση της φυσικής σύνδεσης μεταξύ δύο συσκευών. Περιλαμβάνει το σχεδιασμό και τη διαχείριση της διεπαφής ασύρματης επικοινωνίας, τη διαχείριση της σύνδεσης, τη διαχείριση της συχνότητας, τη διαχείριση του ρολογιού, τη διαχείριση των λειτουργιών αποκωδικοποίησης/κωδικοποίησης και τη διαχείριση της ασφάλειας. Το στρώμα αυτό επιτρέπει την ασύρματη επικοινωνία μεταξύ συσκευών Bluetooth που βρίσκονται σε μικρή απόσταση. Αυτό το στρώμα είναι επίσης υπεύθυνο για την επιλογή της κατάλληλης συχνότητας και τη διαχείριση των συγκρούσεων στην ασύρματη επικοινωνία.

Το επίπεδο **Link Manager Protocol** (LMP) είναι υπεύθυνο για τη διαχείριση των συνδέσεων μεταξύ συσκευών Bluetooth και τη διαχείριση της συγχρονισμένης μετάδοσης δεδομένων μεταξύ τους. Περιλαμβάνει διάφορα υποσυστήματα όπως το σύστημα αναζήτησης συσκευών, το σύστημα σύζευξης και το σύστημα διαχείρισης συνδέσεων. Η κύρια λειτουργία του είναι να επιτρέπει τη σύνδεση και τη διαχείριση συνδέσεων μεταξύ συσκευών Bluetooth. Περιλαμβάνει μηχανισμούς αναζήτησης συσκευών και σύζευξης, καθώς και τη διαχείριση των συνδέσεων και την ανανέωση τους αν χρειαστεί. Επίσης, είναι υπεύθυνο για τη διαχείριση των αιτημάτων σύνδεσης και τη δημιουργία των συνδέσεων μεταξύ συσκευών. Αυτό περιλαμβάνει τον έλεγχο της πρόσβασης στο κανάλι και την ανταλλαγή πληροφοριών για τη σύνδεση. Ο σχεδιασμός του επιπέδου αυτού επιτρέπει τη δημιουργία συνδέσεων μεταξύ δυο ή περισσότερων συσκευών Bluetooth και την επικοινωνία μεταξύ τους. Η επικοινωνία μπορεί να γίνει μεταξύ δύο συσκευών (point-to-point) ή μεταξύ περισσότερων συσκευών (multipoint).

Το επίπεδο **Host Controller Interface** είναι το υπεύθυνο στρώμα για την επικοινωνία του Bluetooth με τον υπολογιστή ή τη συσκευή ελέγχου. Συγκεκριμένα, το επίπεδο αυτό αναλαμβάνει τη διαχείριση των συσκευών Bluetooth που είναι συνδεδεμένες στο διακοσμητή, τη δημιουργία και τη διαχείριση των συνδέσεων Bluetooth και την υλοποίηση των υπηρεσιών Bluetooth. Το στρώμα Host Controller είναι υλοποιημένο σε λογισμικό και συνήθως περιλαμβάνει έναν ειδικό οδηγό που διαχειρίζεται τη σύνδεση του διακοσμητή με τη συσκευή Bluetooth.

Το επίπεδο **Logical Link Control and Adaptation Protocol** (L2CAP) είναι ένα από τα υψηλότερα στρώματα στο πρωτόκολλο Bluetooth και παρέχει υπηρεσίες προσαρμογής στα παραπάνω στρώματα. Το επίπεδο αυτό επιτρέπει την πολυπλοκότητα των δεδομένων να μεταφέρονται μεταξύ των συσκευών Bluetooth. Επίσης, υποστηρίζει διάφορες υπηρεσίες, όπως τη μεταφορά μεγάλων δεδομένων, τη μεταφορά αρχείων, τη φωνητική επικοινωνία και την επικοινωνία δεδομένων σε πραγματικό χρόνο. Επιπλέον, χρησιμοποιείται για τη διαχείριση της κρυπτογράφησης και της ασφάλειας στη σύνδεση Bluetooth. Το L2CAP αποτελεί ένα σημαντικό στρώμα στο πρωτόκολλο Bluetooth και επιτρέπει στις συσκευές να επικοινωνούν αποτελεσματικά μεταξύ τους.

Το **Radio Frequency Communication** (RFCOMM) είναι ένα από τα πρωτόκολλα που χρησιμοποιούνται στο στρώμα L2CAP του Bluetooth, για τη δημιουργία εικονικών σειριακών

πορτών στην επικοινωνία μεταξύ συσκευών. Διευκολύνει τη μεταφορά δεδομένων μεταξύ διαφορετικών εφαρμογών, όπως η επικοινωνία με ένα ασύρματο ακουστικό ή ένα GPS δέκτη. Χρησιμοποιεί μια εικονική σειριακή πόρτα για την επικοινωνία μεταξύ των δύο συσκευών, και επιτρέπει την ανταλλαγή δεδομένων με τη μορφή πακέτων μέσω αυτής της εικονικής πόρτας. Είναι επίσης υπεύθυνο για τη διαχείριση της επανασύνδεσης σε περίπτωση απώλειας σύνδεσης και για τη διαχείριση των διακοπών κλήσεων.

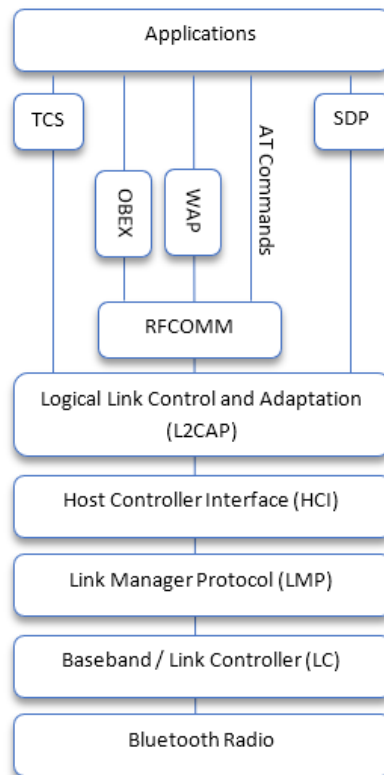
Το **Wireless Application Protocol (WAP)** είναι ένα πρωτόκολλο επικοινωνίας που χρησιμοποιείται στα κινητά τηλέφωνα και άλλες φορητές συσκευές για τη μετάδοση δεδομένων και την πρόσβαση σε διαδικτυακούς πόρους. Το WAP παρέχει μια επιφάνεια προγραμματισμού εφαρμογών (API) για τη δημιουργία εφαρμογών που μπορούν να εκτελούνται σε φορητές συσκευές.

Το **Object Exchange (OBEX)** είναι ένα πρωτόκολλο ανταλλαγής αντικειμένων που χρησιμοποιείται επίσης σε φορητές συσκευές και κινητά τηλέφωνα. Το OBEX επιτρέπει τη μεταφορά αρχείων και άλλων αντικειμένων μεταξύ φορητών συσκευών μέσω Bluetooth, υπέρυθρων και άλλων τεχνολογιών ασύρματης επικοινωνίας.

Το **Service Discovery Protocol (SDP)** είναι ένα πρωτόκολλο που επιτρέπει στις συσκευές να ανακαλύπτουν άλλες συσκευές Bluetooth στην περιοχή τους και να ανακαλύπτουν τα χαρακτηριστικά και τις υπηρεσίες που παρέχουν. Αυτό επιτρέπει στις συσκευές να επικοινωνούν και να αλληλεπιδρούν μεταξύ τους. Η αναζήτηση συσκευής μπορεί να βασίζεται σε κριτήρια όπως οι υπηρεσίες που παρέχονται από τη συσκευή, το όνομα της συσκευής, τον τύπο της συσκευής και άλλα χαρακτηριστικά.

Το **Telephony Control Protocol Specification (TCS)** είναι ένα πρωτόκολλο που χρησιμοποιείται στην τεχνολογία Bluetooth για την υποστήριξη φωνητικών επικοινωνιών σε ασύρματες συσκευές. Παρέχει λειτουργίες για τη δημιουργία, τη διατήρηση και τη λήξη μιας συνεδρίας φωνητικής επικοινωνίας μεταξύ δύο Bluetooth συσκευών. Επιπλέον, παρέχει δυνατότητες για την αποστολή και λήψη φωνητικών δεδομένων μεταξύ των συσκευών. Το TCS χρησιμοποιείται συνήθως σε συσκευές hands-free, ακουστικά Bluetooth και συσκευές κινητής τηλεφωνίας.

Το **Applications Layer** ή **Επίπεδο Εφαρμογών** στο Bluetooth είναι το υψηλότερο επίπεδο της στοίβας πρωτοκόλλων και αποτελεί τη διεπαφή μεταξύ της συσκευής Bluetooth και των εφαρμογών που τη χρησιμοποιούν. Το επίπεδο αυτό παρέχει πρωτόκολλα για τη διαχείριση διαφόρων υπηρεσιών, όπως τη μεταφορά αρχείων, τον έλεγχο πρόσβασης στο διαδίκτυο, την αυτόματη σύνδεση σε ασύρματα δίκτυα και άλλες εφαρμογές. [\[16\]](#) [\[17\]](#)



Εικόνα 2. 3 Σχεδιάγραμμα Αρχιτεκτονικής στοίβας πρωτοκόλλου Bluetooth

Η στοίβα πρωτοκόλλων του μοντέλου αναφοράς OSI και του Bluetooth δεν μπορούν αντιστοιχιστούν άμεσα καθώς είναι διαφορετικά συστήματα πρωτοκόλλων με διαφορετικούς σκοπούς. [\[17\]](#)

2.2.4 Τύποι συσκευών Bluetooth

Για να επιτραπεί στις συσκευές να μεταβαίνουν σε νέες συχνότητες μετά από κάθε πακέτο δεδομένων, πρέπει να συμφωνήσουν όλες στην ακολουθία των συχνοτήτων που θα χρησιμοποιήσουν. Οι συσκευές Bluetooth μπορούν να λειτουργούν σε δυο καταστάσεις: ως Master ή ως Slave. Ο Master καθορίζει την ακολουθία του αναπήδησης συχνότητας ή αλλιώς frequency hopping. Οι Slaves συγχρονίζονται στον Master ως προς τον χρόνο και τη συχνότητα ακολουθώντας την ακολουθία αυτή.

Κάθε συσκευή Bluetooth διαθέτει μοναδική διεύθυνση συσκευής Bluetooth και ρολόι Bluetooth. Το τμήμα της προδιαγραφής Bluetooth που αναφέρεται στη βάση περιγράφει έναν αλγόριθμο που μπορεί να υπολογίσει μια ακολουθία αλλαγής συχνότητας από τη διεύθυνση της συσκευής Bluetooth και το ρολόι της. Όταν οι Slaves συνδέονται σε έναν Master, λαμβάνουν τη διεύθυνση της συσκευής Bluetooth και το ρολόι του Master. Στη συνέχεια, τα χρησιμοποιούν αυτά για να υπολογίσουν την ακολουθία αλλαγής συχνότητας. Επειδή όλοι οι Slaves χρησιμοποιούν το ρολόι και τη διεύθυνση του Master, όλοι συγχρονίζονται με την ακολουθία αλλαγής συχνότητας του Master.

Επιπρόσθετα, ο Master ελέγχει πότε επιτρέπεται στις συσκευές να μεταδίδουν. Ο Master επιτρέπει στους Slaves να μεταδίδουν αναθέτοντας χρονικά κομμάτια για φωνητική κυκλοφορία ή κυκλοφορία δεδομένων. Στα χρονικά κομμάτια κυκλοφορίας δεδομένων, οι Slaves επιτρέπεται να μεταδίδουν μόνο όταν απαντούν σε μετάδοση προς αυτούς από τον Master. Στα χρονικά κομμάτια κυκλοφορίας φωνής, απαιτείται από τους Slaves να μεταδίδουν τακτικά σε διατηρημένα κομμάτια, ανεξαρτήτως του εάν απαντούν στον Master.

Ο Master επιβλέπει επίσης τον τρόπο κατανομής του συνολικού διαθέσιμου εύρους ζώνης μεταξύ των Slaves, καθορίζοντας πότε και πόσο συχνά πρέπει να επικοινωνεί με κάθε Slave. Ο αριθμός των χρονικών κομματιών που κάθε συσκευή λαμβάνει εξαρτάται από τις απαιτήσεις της στη μεταφορά δεδομένων. Αυτό το σύστημα διαίρεσης των χρονικών κομματιών μεταξύ πολλαπλών συσκευών ονομάζεται Πολυπλεξία Χρονικής Διαίρεσης (Time Division Multiplexing - TDM). [\[16\]](#) [\[17\]](#)

2.2.5 Τύποι Συνδέσεων

Οι τύποι συνδέσεων Bluetooth καλύπτουν διάφορες διατάξεις για την ασύρματη επικοινωνία μεταξύ συσκευών. Ας αναλύσουμε τους τέσσερις βασικούς τύπους:

1. **Point-to-Point (P2P):** Σε μια σύνδεση P2P, υπάρχει απευθείας σύνδεση ανάμεσα σε δύο συσκευές Bluetooth, μια Master και μια Slave. Ο Master είναι υπεύθυνος για τον έλεγχο της επικοινωνίας. [\[16\]](#) [\[17\]](#)
2. **Point-to-Multipoint (P2M):** Σε μια σύνδεση P2M, ένας Master συνδέεται ταυτόχρονα με πολλά Slaves. Ο Master διαχειρίζεται την επικοινωνία με όλες τις Slave συσκευές. [\[17\]](#)
3. **Mesh Networks:** Σε ένα δίκτυο Mesh, κάθε συσκευή συνδέεται με κάθε άλλη συσκευή στο δίκτυο, δημιουργώντας ένα αυτοοργανωμένο δίκτυο. Δεν υπάρχει κεντρικός Master. [\[21\]](#) [\[22\]](#)
4. **Scatternets:** Ένα Scatternet είναι ένα δίκτυο που αποτελείται από πολλά Piconets (μικρότερα δίκτυα) που συνδέονται μεταξύ τους μέσω μιας συσκευής που λειτουργεί ως Master σε ένα Piconet και ως Slave σε ένα άλλο Piconet. [\[16\]](#) [\[17\]](#) [\[18\]](#)

Κάθε τύπος σύνδεσης έχει τις δικές του εφαρμογές και χρήσεις, και η επιλογή εξαρτάται από τις απαιτήσεις της συγκεκριμένης εφαρμογής ή σεναρίου.

2.2.6 Σύζευξη και Σύνδεση Συσκευών

Οι συσκευές Bluetooth μεταδίδουν πληροφορίες κατόπιν αιτήματος, συμπεριλαμβανομένου του ονόματος της συσκευής, της κλάσης της, του καταλόγου υπηρεσιών και των τεχνικών λεπτομερειών. Έχουν τη δυνατότητα να πραγματοποιήσουν αναζητήσεις για να συνδεθούν με άλλες συσκευές, αλλά απαντούν μόνο σε αιτήματα άμεσης σύνδεσης εάν είναι γνωστή η διεύθυνση της συσκευής. Οι συσκευές μπορούν να συνδεθούν μόνο με μία συσκευή τη φορά, εμποδίζοντας την εμφάνισή τους σε αναζητήσεις μέχρι να αποσυνδεθούν. Κάθε συσκευή διαθέτει μια μοναδική διεύθυνση 48 bit, αλλά χρησιμοποιούνται φιλικά ονόματα Bluetooth αντί για αυτήν.

Τα περισσότερα τηλέφωνα και φορητοί υπολογιστές εμφανίζουν μόνο τα φιλικά ονόματα Bluetooth.

Οι υπηρεσίες Bluetooth μπορούν να αποκαλύψουν προσωπικά δεδομένα ή να επιτρέψουν τον έλεγχο των συσκευών. Για να διασφαλίσει την ασφάλεια, το Bluetooth χρησιμοποιεί τη σύνδεση (bonding) και τη σύζευξη (pairing) για να καθιερώσει συνδέσεις χωρίς παρέμβαση του χρήστη. Η σύνδεση ενεργοποιείται από αιτήματα του χρήστη ή αυτόματα όταν απαιτείται η ταυτότητα μιας συσκευής για λόγους ασφαλείας. Η σύζευξη συχνά περιλαμβάνει την αλληλεπίδραση του χρήστη για την ταυτοποίηση των συσκευών. Μόλις ολοκληρωθεί, δημιουργείται μια σύνδεση μεταξύ των συσκευών, επιτρέποντας μελλοντικές συνδέσεις χωρίς την επανάληψη της διαδικασίας. Οι χρήστες μπορούν να καταργήσουν τη σχέση σύνδεσης κατά βούληση. ^[19]

2.2.7 Ασφάλεια

Η ασφάλεια αποτελεί ένα θεμέλιο στοιχείο της τεχνολογίας Bluetooth, καθώς εξασφαλίζει την προστασία των συσκευών και των δεδομένων κατά τη διάρκεια της ασύρματης επικοινωνίας. Παρακάτω περιγράφονται μερικά από τα κύρια στοιχεία που σχετίζονται με την ασφάλεια της τεχνολογίας Bluetooth:

1. **Αυθεντικοποίηση:** Η διαδικασία της αυθεντικοποίησης απαιτεί την επαλήθευση της ταυτότητας δύο συσκευών πριν από την επιτροπή επικοινωνίας. Οι χρήστες συνήθως πρέπει να καθορίσουν έναν κωδικό πρόσβασης (PIN) προκειμένου να εξασφαλίσουν την ασφάλεια της σύνδεσης. ^{[16] [18]}
2. **Κρυπτογράφηση Δεδομένων:** Τα δεδομένα που ανταλλάσσονται μεταξύ συσκευών Bluetooth κρυπτογραφούνται για να προστατευτούν από ανεπιθύμητη πρόσβαση. Αυτό σημαίνει ότι ακόμα και αν κάποιος καταφέρει να καταγράψει την επικοινωνία, τα δεδομένα παραμένουν ασφαλή. ^[18]
3. **Ενημέρωση Λογισμικού:** Η ενημέρωση του λογισμικού των συσκευών Bluetooth είναι κρίσιμη για τη διόρθωση αδυναμιών ασφαλείας και την προστασία από επιθέσεις. ^[19]
4. **Περιορισμός Ανιχνευσιμότητας:** Οι χρήστες μπορούν να ρυθμίσουν το επίπεδο ανιχνευσιμότητας της συσκευής τους, μειώνοντας τον κίνδυνο ανεπιθύμητης πρόσβασης. ^{[16] [18]}
5. **Διαχείριση Δικαιωμάτων:** Οι συσκευές μπορούν να διαχειρίζονται τα δικαιώματά ως προς τη χρήση Bluetooth, προστατεύοντας έτσι την ασφάλειά τους. Αυτό σημαίνει ότι μόνο εξουσιοδοτημένοι χρήστες έχουν πρόσβαση σε συγκεκριμένες λειτουργίες.

Η τεχνολογία Bluetooth διαρκώς εξελίσσεται και με αυτόν τον τρόπο εμφανίζονται και νέες προκλήσεις στον τομέα της ασφάλειας. Η αντιμετώπιση αυτών των προκλήσεων είναι σημαντική για τη διασφάλιση της ασφάλειας των συσκευών της.

2.3 Bluetooth Module HC-05

Το HC-05 είναι ένα απλό στη χρήση Bluetooth module, σχεδιασμένο για ασύρματη σειριακή σύνδεση μεταξύ συσκευών. Το module αυτό χρησιμοποιείται συχνά σε DIY (Κάντο μόνος σου) project και σε εφαρμογές IoT (Διαδίκτυο των Πραγμάτων), και προσφέρει μια εύκολη και αξιόπιστη λύση για την ασύρματη επικοινωνία μεταξύ δύο συσκευών με τη χρήση του Bluetooth.

Το HC-05 είναι πιστοποιημένο για Bluetooth V2.0+EDR (ενισχυμένο ρυθμό μετάδοσης δεδομένων) και χρησιμοποιεί ένα μόνο εξωτερικό μικροελεγκτή Bluetooth CSR Bluecore 04 με τεχνολογία CMOS και χαρακτηριστικό AFH (Adaptive Frequency Hopping). Το module έχει μικρό μέγεθος, με διαστάσεις 12.7mm x 27mm και είναι συμβατό με πολλές πλατφόρμες και μικροελεγκτές, όπως το Arduino. [\[23\]](#)

Μπορεί να ρυθμιστεί εύκολα μέσω του AT σετ εντολών, προσφέροντας πολλές επιλογές στους χρήστες, όπως η ρύθμιση του ρυθμού μετάδοσης, της ισχύος και του ονόματος της συσκευής. Επίσης, υποστηρίζει το Serial Port Profile (SPP) για απευθείας σύνδεση σειριακού καναλιού και το HID Profile (Human Interface Device) για σύνδεση μεταξύ ασύρματων συσκευών εισόδου/εξόδου.

Επιπλέον, το HC-05 υποστηρίζει τη λειτουργία Master και Slave, καθιστώντας το ιδανικό για ασύρματη επικοινωνία μεταξύ δύο συσκευών. Το module λειτουργεί στη συχνότητα 2,4 GHz και χρησιμοποιεί το πρωτόκολλο SPP για τη μεταφορά δεδομένων μεταξύ συσκευών. [\[24\]](#)



Εικόνα 2. 4 HC-05 Module for Arduino

Τεχνικά Χαρακτηριστικά [\[23\]](#)

Hardware

- Τυπική ευαισθησία -80dBm
- Έως και +4dBm RF ισχύς μετάδοσης
- Χαμηλή Ισχύς Λειτουργίας 1.8V, από 1.8 έως 3.6V I/O
- Έλεγχος προγραμματιζόμενης εισόδου/εξόδου (PIO control)
- Διεπαφή UART με προγραμματιζόμενο baud rate
- Ενσωματωμένη κεραία
- Ακροδέκτες

Λογισμικό

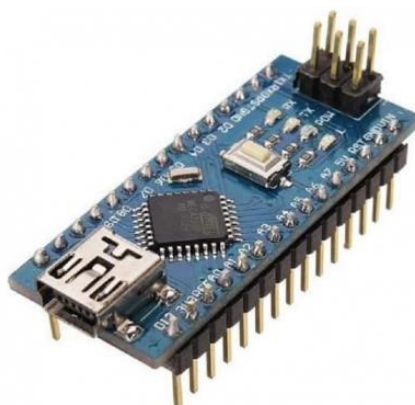
- Προεπιλεγμένο Baud rate: 38400, Data bits:8, Stop bit:1, Parity: No parity, Data control: περιλαμβάνει.

Υποστηριζόμενα baud rate: 9600, 19200, 38400, 57600, 115200, 230400, 460800.

- Εάν δοθεί παλμός στο PIO0, η συσκευή θα αποσυνδεθεί.
- Θύρα πληροφόρησης κατάστασης PIO1: Low - αποσυνδέθηκε, High - συνδέθηκε;
- PIO10 και PIO11 μπορούν να συνδεθούν στο κόκκινο και στο μπλε LED ξεχωριστά. Όταν το master και το slave είναι συζευγμένα, το κόκκινο και το μπλε LED αναβοσβήνει 1 φορά κάθε 2 δευτερόλεπτα σε διαστήματα, και όταν αποσυνδεδεμένα μόνο το μπλε LED αναβοσβήνει στον ίδιο ρυθμό.
- Αυτόματη σύνδεση στη προηγούμενη συσκευή που λειτουργούσε ως προεπιλογή.
- Επιτροπή στη σύζευξη σε συσκευή για σύνδεση, ως προεπιλογή.
- Αυτόματη σύζευξη με κωδικό PIN: "1234" ως προεπιλογή.
- Αυτόματη σύνδεση σε 30 λεπτά όταν αποσυνδέεται λόγω αποκοπής σήματος λόγω μη εφικτού εύρους.

2.4 Arduino Nano

Για τον προγραμματισμό για τον έλεγχο των κινητήρων stepper-motors επιλέχθηκε το Arduino Nano, ένας μικροελεγκτής που βασίζεται στην δημοφιλή πλατφόρμα Arduino. Επιλέχθηκε ειδικά για το συγκεκριμένο project λόγω του μικρού του μεγέθους και της ευελιξίας που προσφέρει. Παρά το μικρό του μέγεθος, διαθέτει μια πληθώρα αναλογικών και ψηφιακών ακροδεκτών, οι οποίοι είναι απαραίτητοι για την εκτέλεση του project. Προσφέρει μια ιδανική ισορροπία μεταξύ μικρού μεγέθους και υψηλής λειτουργικότητας, καθιστώντας το τέλειο για εφαρμογές που απαιτούν ελάχιστο χώρο αλλά χωρίς να θυσιάζουν τις δυνατότητες που παρέχουν οι περισσότεροι ακροδέκτες. Επιπλέον, η χρήση του διευκολύνει την ανάπτυξη του project, καθώς η προγραμματιστική διεπαφή Arduino IDE είναι εύχρηστη και προσαρμοσμένη στις ανάγκες των χρηστών.



Εικόνα 2. 5 Arduino Nano



Εικόνα 2. 6 ATmega328P

2.4.1 ATmega328

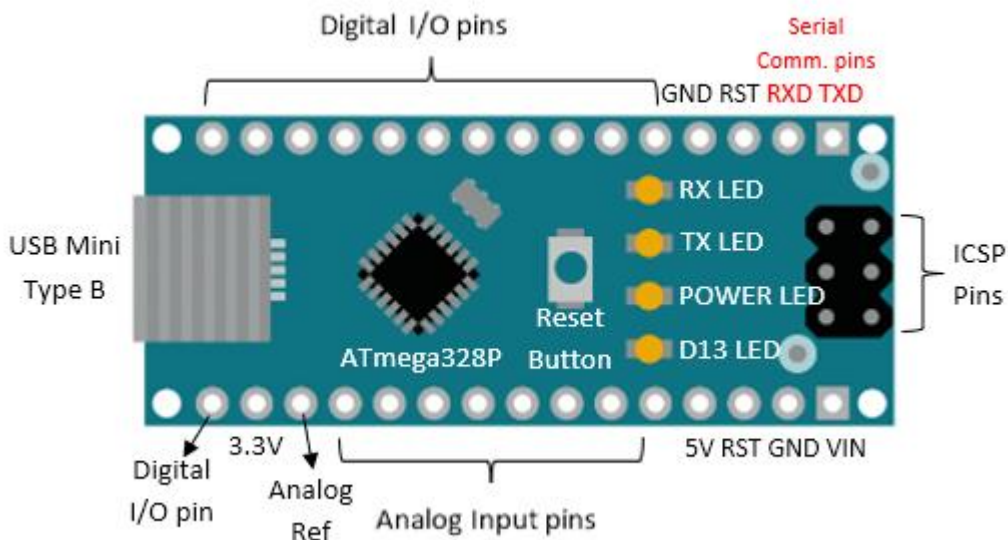
Το Arduino Nano ενσωματώνει τον ATmega328, έναν μικροελεγκτή 8-bit που παράγεται από την εταιρεία ATMEL, η οποία είναι τώρα μέρος της Microchip Technology. Αυτός ο μικροελεγκτής είναι πολύ δημοφιλής και βρίσκεται στην καρδιά πολλών πλακετών Arduino, συμπεριλαμβανομένου του Arduino Uno και του Arduino Nano. [\[25\]](#)

2.4.2 Τεχνικά χαρακτηριστικά [\[26\]](#) [\[27\]](#)

Το Arduino Nano χρησιμοποιεί τον μικροελεγκτή ATmega328, της ATMEL, που παρέχει 32KB flash μνήμη, 2KB SRAM, 1KB EEPROM, και έως 20MHz συχνότητα. Έχει 14 ψηφιακά pins εισόδου/εξόδου (από τα οποία τα 6 μπορούν να χρησιμοποιηθούν ως PWM outputs), 8 αναλογικά inputs, μια σύνδεση μεταφοράς σειριακών δεδομένων (USART), και έναν εξωτερικό κρύσταλλο 16MHz. Υποστηρίζει επίσης SPI και I2C επικοινωνία.

Τεχνικά χαρακτηριστικά:

- Μικροελεγκτής: Microchip ATmega328P
- Τάση λειτουργίας: 5 Volts
- Τάση εισόδου: 5 to 12 Volts
- Ψηφιακοί I/O ακροδέκτες: 14 (6 optional PWM outputs)
- Αναλογικοί ακροδέκτες: 8
- DC ανά I/O pin: 40 mA
- DC για ακροδέκτη τάσης 3.3V: 50 mA
- Μνήμη Flash: 32 KB, εκ των οποίων 2 KB χρησιμοποιούνται από τον bootloader
- SRAM: 2 KB
- EEPROM: 1 KB
- Συχνότητα ρολογιού (Clock speed): 16 MHz
- Μήκος: 45 mm
- Πλάτος: 18 mm
- Μάζα: 7 g
- USB: Mini-USB Type-B
- ICSP Header: Ναι
- DC Power Jack: Όχι



Εικόνα 2. 7 Διάγραμμα ακροδεκτών του Arduino Nano

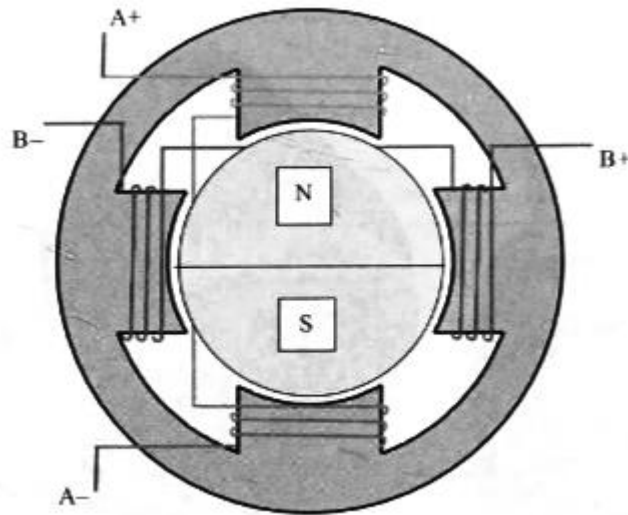
2.5 Βηματικοί κινητήρες

Ο βηματικός κινητήρας ή αλλιώς stepper-motor είναι ένας ηλεκτρικός κινητήρας επαγωγής ο οποίος μετατρέπει ψηφιακό σήμα σε μηχανική περιστροφή του άξονά του. Η περιστροφή του άξονα πραγματοποιείται σε προκαθορισμένα διακριτά βήματα. Κάθε βήμα είναι μια πολύ μικρή γωνία περιστροφής και ο κινητήρας μπορεί να περιστρέψει τον άξονά του κατά αυτή τη γωνία με μεγάλη ακρίβεια. Η δυνατότητα αυτή προκύπτει από την κατασκευή του κινητήρα. Ένας βηματικός κινητήρας αποτελείται από έναν στάτορα (το σταθερό μέρος του κινητήρα) που περιέχει πηνία και έναν ρότορα (το κινούμενο μέρος του κινητήρα). Όταν τα πηνία τροφοδοτούνται με ηλεκτρικό ρεύμα, προκαλούν ένα μαγνητικό πεδίο που περιστρέφει τον ρότορα. [\[28\]](#)

2.5.1 Είδη βηματικών κινητήρων

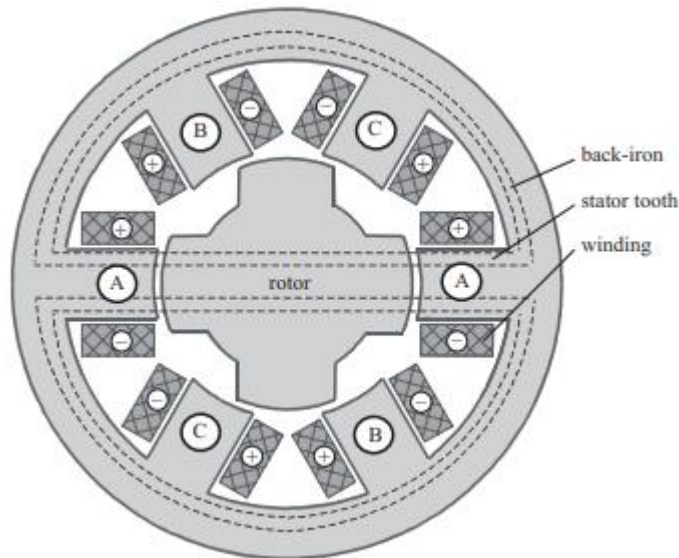
Οι βηματικοί κινητήρες διακρίνονται σε τρεις βασικούς τύπους: μόνιμου μαγνήτη, μεταβλητού μαγνητισμού και υβριδικού. [\[29\]](#)

Βηματικός κινητήρας με μόνιμο μαγνήτη: Σε αυτόν τον κινητήρα, ο ρότορας είναι κατασκευασμένος από έναν μόνιμο μαγνήτη. Αυτός ο κινητήρας μπορεί να παράγει υψηλή ροπή σε χαμηλές ταχύτητες, αλλά η ροπή μειώνεται γρήγορα με την αύξηση της ταχύτητας. [\[30\]](#)



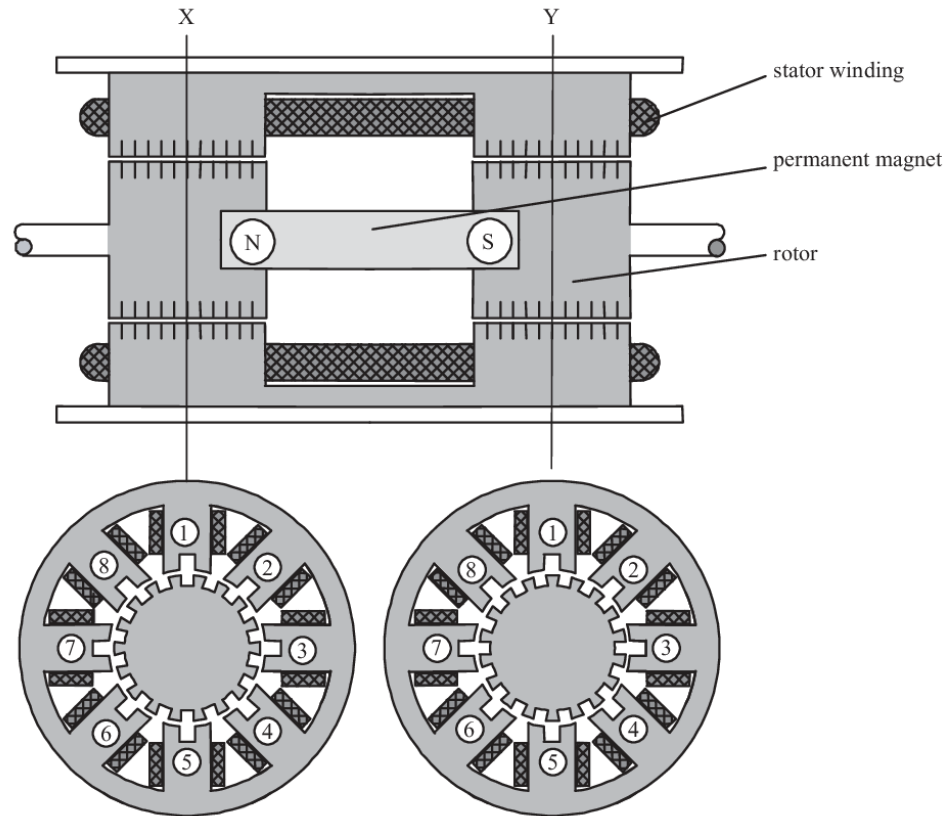
Εικόνα 2. 8 Βηματικός κινητήρας με μόνιμο μαγνήτη

Βηματικός κινητήρας μεταβλητού μαγνητισμού: Ο ρότορας αυτού του κινητήρα είναι κατασκευασμένος από μαγνητικό υλικό και τα πηνία είναι τοποθετημένα στο στάτορα. Η αλλαγή του μαγνητικού πεδίου προκαλεί την περιστροφή του ρότορα. Αυτός ο κινητήρας έχει μεγαλύτερη απόδοση σε υψηλότερες ταχύτητες σε σχέση με τον κινητήρα με μόνιμο μαγνήτη. [\[30\]](#)



Εικόνα 2. 9 Βηματικού κινητήρα μεταβλητού μαγνητισμού

Υβριδικός: Οι υβριδικός βηματικός κινητήρας συνδυάζει τα χαρακτηριστικά του κινητήρα με μόνιμο μαγνήτη και του μεταβλητού μαγνητισμού, προσφέροντας υψηλότερη ροπή σε μεγάλο εύρος ταχυτήτων. Είναι ο πιο δημοφιλής τύπος βηματικού κινητήρα λόγω της ευελιξίας και της υψηλής απόδοσης τους. [\[30\]](#)

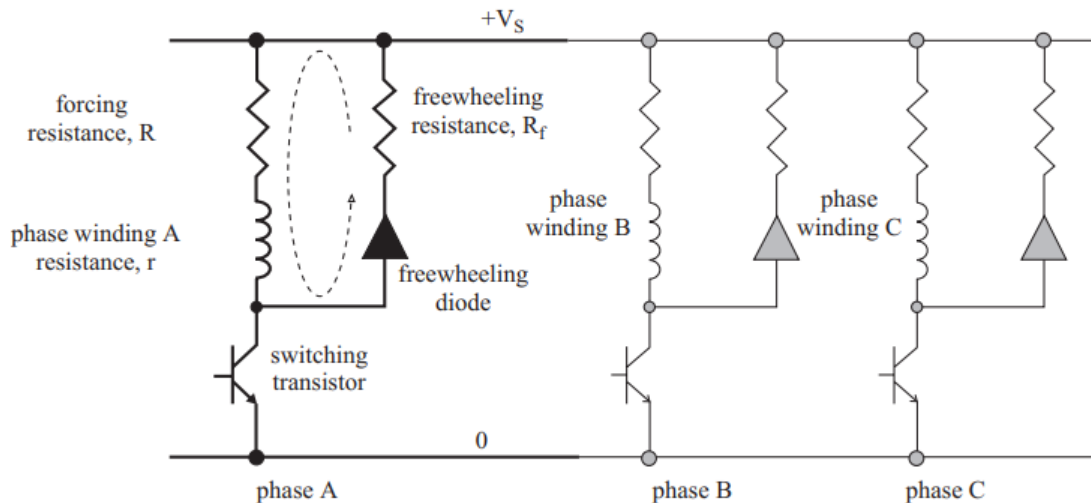


Εικόνα 2. 10 Υβριδικός βηματικός κινητήρας

2.5.2 Τρόποι οδήγησης ενός βηματικού κινητήρα

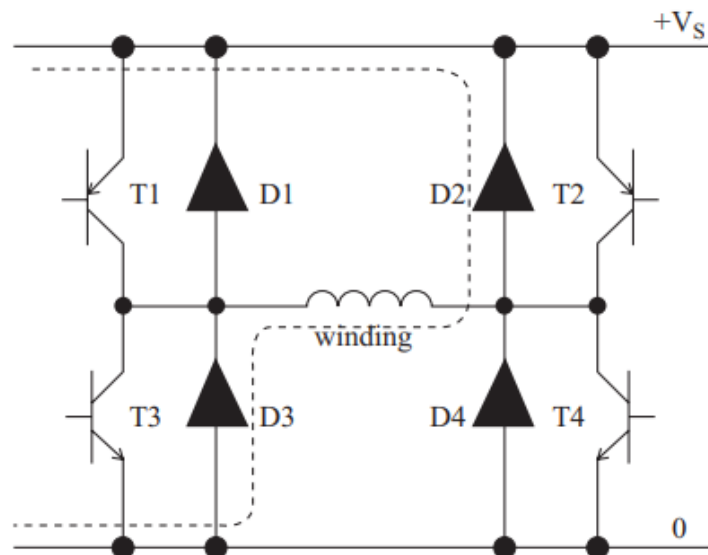
Οι βηματικοί κινητήρες μπορούν να οδηγηθούν με διάφορους τρόπους, με κάθε μέθοδο να έχει τα δικά της πλεονεκτήματα και αδυναμίες. Οι πιο συνηθισμένες μέθοδοι οδήγησης είναι οι εξής:

Μονοπολική οδήγηση: Παρακάτω παρουσιάζεται ένα απλό κύκλωμα οδήγησης μονοπολικού τύπου για χρήση σε τριφασικό βηματικό κινητήρα μεταβλητής αντίστασης. Κάθε φάση ενεργοποιείται από ένα ξεχωριστό κύκλωμα οδήγησης, το οποίο ελέγχεται από ένα χαμηλής ισχύος "σήμα ελέγχου φάσης". Ο βρόγχος κάθε φάσης ενεργοποιείται όταν το αντίστοιχο transistor βρίσκεται σε κορεσμό από επαρκές ρεύμα στη βάση του. Τότε, εφαρμόζεται η πλήρης τάση τροφοδοσίας DC στον σειριακό συνδυασμό των βρόγχων, εφόσον η πτώση τάσης στο κορεσμένο transistor είναι μικρή (συνήθως 0,2 V). Η τάση τροφοδοσίας DC (Vs) επιλέγεται έτσι ώστε να παράγει το ρεύμα της φάσης (I) όταν εφαρμόζεται στη συνολική αντίσταση του κυκλώματος φάσης, η οποία είναι ίση με το άθροισμα της αντίστασης της φάσης (r) και της αντίστασης επιβολής (forcing resistance, R). Η προσθήκη της αντίστασης επιβολής, με αναλογική αύξηση της τάσης τροφοδοσίας, μειώνει τον φυσικό χρόνο απόκρισης της φάσης, επιτρέποντας τη λειτουργία σε ευρύτερο εύρος ταχυτήτων. [\[30\]](#)



Εικόνα 2. 11 Μονοπολική οδήγηση βηματικού κινητήρα

Διπολική οδήγηση: Το παρακάτω κύκλωμα που παρουσιάζεται, αποτελεί κύκλωμα διπολικής οδήγησης με γέφυρα transistor, κατάλληλο για εφαρμογή σε υβριδικούς ή βηματικούς κινητήρες με μόνιμο μαγνήτη. Τα transistor ενεργοποιούνται σε ζεύγη ανάλογα με την κατεύθυνση του ρεύματος που απαιτείται. Ένα σύνολο διόδων παρέχει τη διαδρομή για τα ρεύματα αντιστάθμισης, επιτρέποντας την επιστροφή αποθηκευμένης ενέργειας στην παροχή. Αυτή η διαδικασία είναι πιο αποδοτική από την οδήγηση μονοπολικού κυκλώματος, επομένως χρησιμοποιείται συχνά για μεγάλους βηματικούς κινητήρες. ^[30]

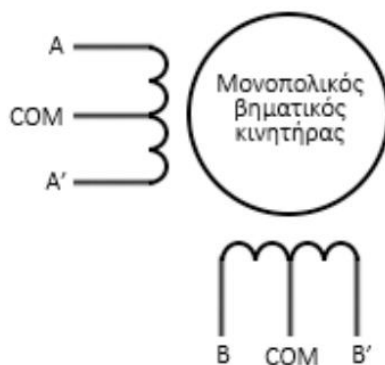


Εικόνα 2. 12 Διπολική οδήγηση βηματικού κινητήρα

2.5.3 Τύποι βηματικών κινητήρα

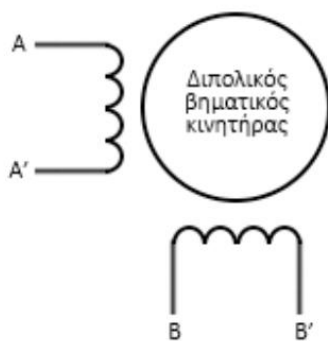
Οι δύο βασικοί τύποι βηματικών κινητήρων είναι οι μονοπολικοί και οι διπολικοί. Παρακάτω περιγράφονται αναλυτικά οι δυο τύποι και οι διαφορές τους.

Μονοπολικοί βηματικοί κινητήρες ή unipolar stepper motors: Έχουν ένα ενιαίο πηνίο για κάθε φάση με το κέντρο του πηνίου να συνδέεται σε κοινή τροφοδοσία. Αυτό δημιουργεί ένα μαγνητικό πεδίο μόνο στη μία κατεύθυνση, για αυτό και το όνομα "μονοπολικός". Οι μονοπολικοί κινητήρες είναι συνήθως πιο εύκολοι στη χρήση, αλλά λιγότερο αποδοτικοί σε σχέση με τους διπολικούς. [\[31\]](#)



Εικόνα 2. 13 Μονοπολικός βηματικός κινητήρας

Διπολικοί βηματικοί κινητήρες ή bipolar stepper motors: Έχουν δύο πηνία για κάθε φάση χωρίς κοινή σύνδεση, το οποίο επιτρέπει τον έλεγχο του μαγνητικού πεδίου σε δύο κατευθύνσεις. Αυτό σημαίνει ότι μπορούν να δημιουργήσουν περισσότερη ροπή για το ίδιο μέγεθος κινητήρα σε σύγκριση με τους μονοπολικούς κινητήρες. Ωστόσο, οι διπολικοί κινητήρες απαιτούν πιο πολύπλοκους οδηγούς για να ελέγξουν την κατεύθυνση του ρεύματος που διαρρέει μέσα από τα πηνία τους. [\[31\]](#)



Εικόνα 2. 14 Διπολικός βηματικός κινητήρας

2.5.4 Λειτουργία βηματικών κινητήρα

Οι βηματικοί κινητήρες παρουσιάζουν ένα αξιοσημείωτο χαρακτηριστικό σχετικά με την ταχύτητα λειτουργίας και τη ροπή. Συγκεκριμένα, η ροπή του βηματικού κινητήρα αλλάζει με την αλλαγή της ταχύτητας. Όταν ο βηματικός κινητήρας λειτουργεί σε χαμηλές ταχύτητες, μπορεί να

παράγει τη μέγιστη ροπή του. Αυτή είναι η ροπή που αναφέρεται συχνά στις τεχνικές προδιαγραφές του κινητήρα. Ωστόσο, καθώς η ταχύτητα αυξάνεται, η ροπή του κινητήρα μειώνεται. Αυτό οφείλεται σε διάφορους παράγοντες, συμπεριλαμβανομένων των απωλειών διέλευσης ρεύματος μέσα στα πηνία του κινητήρα και της αυξημένης αντίστασης στον αέρα. Αντίθετα, σε πολύ χαμηλές ταχύτητες, οι βηματικοί κινητήρες μπορεί να μην μπορούν να παράγουν επαρκή ροπή για να ξεκινήσουν την κίνηση, εξαιτίας της στατικής τριβής. Είναι σημαντικό να αναγνωρίζεται η σχέση της ταχύτητας με τη ροπή κατά το σχεδιασμό εφαρμογών όπου γίνεται χρήση βηματικών κινητήρων, καθώς αυτό μπορεί να επηρεάσει την απόδοση του συστήματος.

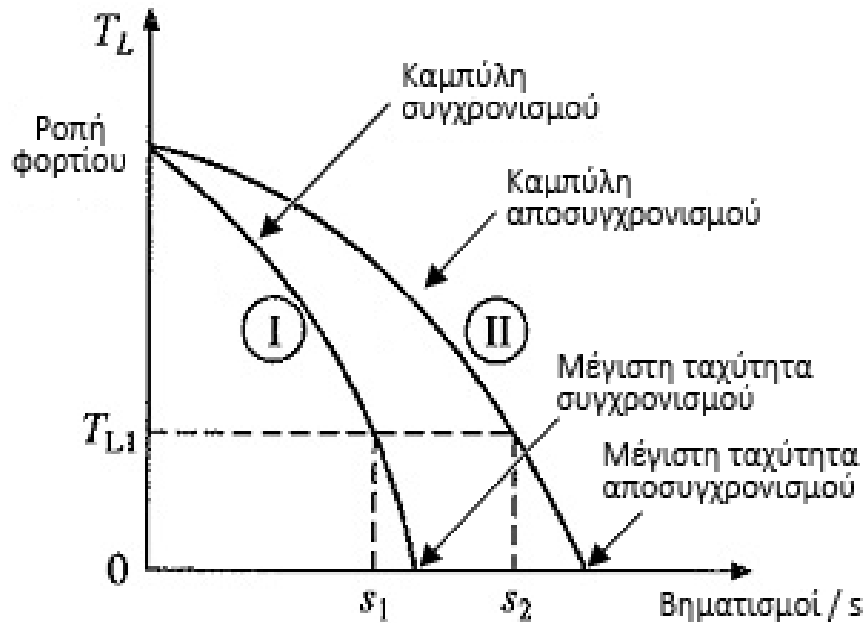
Η λειτουργία ενός βηματικού κινητήρα μπορεί να εξεταστεί σε διάφορες περιοχές, ανάλογα με την ταχύτητα λειτουργίας και την ανταπόκρισή του στον έλεγχο. Εξετάζοντας αυτές τις περιοχές, μπορεί να κατανοηθεί καλύτερα η λειτουργία ενός βηματικού κινητήρα.

Η **περιοχή ελέγχου** είναι η περιοχή στην οποία ο κινητήρας μπορεί να λειτουργήσει με πλήρη ελεγχόμενη ταχύτητα και να διατηρήσει τη θέση του σε σχέση με τα ελεγχόμενα βήματα. Σε αυτήν την περιοχή, ο κινητήρας μπορεί να επιταχύνει, να επιβραδύνει, να σταματήσει ή να αλλάξει κατεύθυνση, όλα σε συγχρονισμό με τα βήματα ελέγχου. Η περιοχή αυτή απεικονίζεται στη χαρακτηριστική ροπής – βηματισμών 2.15 όπου το άνω όριο της περιοχής ελέγχου συμπίπτει με την καμπύλη συγχρονισμού, πέραν της οποίας ο κινητήρας μπορεί να αρχίσει να χάνει τον συγχρονισμό του.

Η **περιοχή ταχύτητας** είναι η περιοχή όπου ο κινητήρας μπορεί να λειτουργήσει σε ταχύτητα που εξαρτάται από την τροφοδοσία ρεύματος και τη σχεδίαση του κινητήρα, αλλά δεν μπορεί να διατηρήσει τον ακριβή έλεγχο της θέσης του. Στη περιοχή αυτή, ο κινητήρας δεν μπορεί να ακολουθήσει πλέον την ταχύτητα των εντολών ελέγχου και αποσυγχρονίζεται, δηλαδή αποτυγχάνει να παραμείνει σε βήμα με τις εντολές ελέγχου. Στην χαρακτηριστική ροπής – βηματισμών 2.15 εκτείνεται από τη μέγιστη ταχύτητα συγχρονισμού έως τη μέγιστη ταχύτητα αποσυγχρονισμού ή από την καμπύλη I έως τη καμπύλη II.

Η **μέγιστη ταχύτητα συγχρονισμού** είναι η μέγιστη ταχύτητα στην οποία ο κινητήρας μπορεί να λειτουργήσει και να διατηρήσει τον συγχρονισμό του με τα ελεγχόμενα βήματα.

Η **μέγιστη ταχύτητα αποσυγχρονισμού** είναι η ταχύτητα στην οποία ο κινητήρας χάνει οριστικά το συγχρονισμό του με τα ελεγχόμενα βήματα. ^[32]

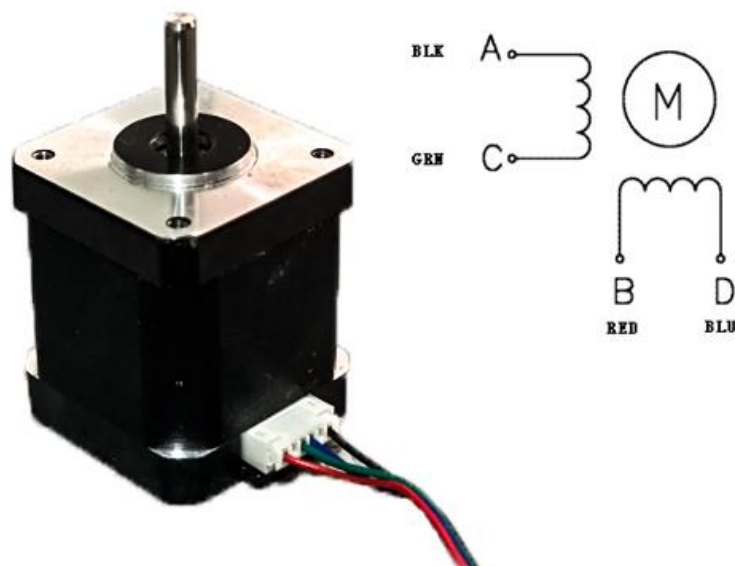


Εικόνα 2. 15 Χαρακτηριστική ροπής – βηματισμών βηματικού κινητήρα

2.5.5 Βηματικός κινητήρας Nema 17

Ο Nema17 είναι ένας υβριδικός διπολικός βηματικός κινητήρας, ο οποίος είναι πολύ δημοφιλής σε εφαρμογές όπως οι 3D εκτυπωτές, CNC routers, ρομποτικά συστήματα, και άλλες εφαρμογές που αποτελούνται από κινητά μέρη.

Συγκεκριμένα στην εφαρμογή αυτή χρησιμοποιήθηκαν τρεις stepper motors Nema 17 ροπής 59Ncm. Είναι κινητήρες μικρών διαστάσεων, όμως προσφέρουν αρκετή δύναμη για να εκτελέσουν τις κινήσεις μιας κάμερας.



Εικόνα 2. 16 Βηματικός κινητήρας Nema 17

Οι προδιαγραφές του Nema17 59Ncm περιλαμβάνουν τα εξής κύρια χαρακτηριστικά:

Διαστάσεις: Οι διαστάσεις της βάσης του Nema17 είναι 43.2 x 43.2 mm και το ύψος του κινητήρα της εφαρμογής είναι 48 mm.

Ροπή: Η ροπή του Nema17 της εφαρμογής αυτής είναι 59 Newton εκατοστόμετρα (Ncm). Αυτή η υψηλή ροπή το καθιστά κατάλληλο για πολλές εφαρμογές που απαιτούν αρκετή δύναμη.

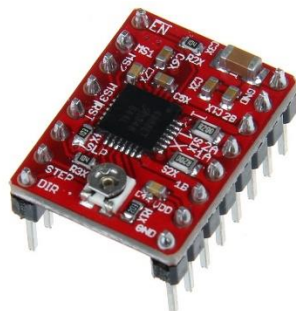
Βήματα: Ο Nema17 59Ncm είναι κινητήρας 200 βημάτων, πράγμα που σημαίνει ότι μπορεί να κάνει 200 πλήρη βήματα ανά περιστροφή, ή 1.8 μοίρες ανά βήμα.

Τροφοδοσία: Συνήθως, οι Nema17 κινητήρες χρειάζονται μια τάση τροφοδοσίας από 12V έως 24V, ανάλογα με τη συγκεκριμένη εφαρμογή και τον ελεγκτή που χρησιμοποιείται. Σε αυτήν την εφαρμογή η τάση τροφοδοσίας των κινητήρων επιλέχθηκε να είναι 12V. [\[33\]](#)

Είναι σημαντικό να σημειωθεί ότι, για τον πλήρη έλεγχο ενός βηματικού κινητήρα όπως ο Nema17, θα χρειαστεί ένας οδηγός, όπως το A4988, ο οποίος περιγράφεται παρακάτω.

2.5.6 Οδηγοί βηματικών κινητήρων A4988

Το A4988 είναι ένα ολοκληρωμένο κύκλωμα οδηγού βηματικού κινητήρα (stepper motor driver) που παράγεται από την εταιρεία Allegro MicroSystems. Είναι ικανό να ελέγχει έναν μονοφασικό κινητήρα βημάτων με έως 2A ανά φάση.

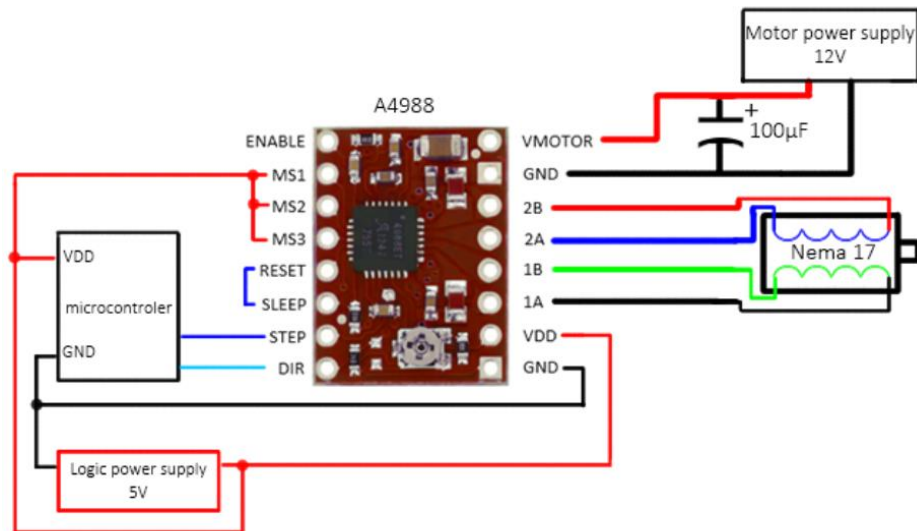


Εικόνα 2. 17 Οδηγός βηματικού κινητήρα A4988

Παρακάτω περιγράφονται μερικά από τα κύρια χαρακτηριστικά του A4988:

1. Λειτουργία Καθοδήγησης: Το A4988 μπορεί να ελέγξει τόσο την κατεύθυνση όσο και την ταχύτητα ενός βηματικού κινητήρα. Οι εντολές περνούν μέσω δύο ακροδεκτών, το DIR (κατεύθυνση) και το STEP (βήμα).
2. Ρύθμιση Ρεύματος: Το A4988 περιλαμβάνει ένα ενσωματωμένο κύκλωμα ρύθμισης ρεύματος που επιτρέπει την προσαρμογή του ρεύματος που παρέχεται στον κινητήρα, καθώς αυτό μπορεί να έχει μεγάλη επίδραση στην απόδοση και την αποδοτικότητα του κινητήρα.
3. Μικροβηματισμός: Το ολοκληρωμένο αυτό κύκλωμα υποστηρίζει πέντε διαφορετικούς τρόπους μικροβηματισμού: πλήρη, 1/2, 1/4, 1/8 και 1/16 βήμα, επιτρέποντας έτσι μεγαλύτερο έλεγχο επί της ακρίβειας και της ταχύτητας του κινητήρα.

4. Προστασία Από Υπερθέρμανση και Υπερφόρτωση: Ο οδηγός A4988 διαθέτει ενσωματωμένη προστασία από υπερθέρμανση και υπερφόρτωση. Σε περίπτωση που η θερμοκρασία του ολοκληρωμένου αυξηθεί πάνω από ένα ορισμένο όριο ή σε περίπτωση που η τάση τροφοδοσίας υπερβεί τα ασφαλή όρια, το ολοκληρωμένο θα απενεργοποιηθεί αυτόματα για να προστατεύσει τον εαυτό του και τον βηματικό κινητήρα από βλάβη. [\[34\]](#)



Εικόνα 2. 18 Συνδεσμολογία του A4988 με το Nema 17

2.5.7 Αποδόσεις του A4988

Ο A4988 είναι ένας ευέλικτος οδηγός για βηματικούς κινητήρες, που μπορεί να λειτουργήσει σε ευρύ φάσμα ταχυτήτων. Όπως και με τους περισσότερους οδηγούς βηματικών κινητήρων, η απόδοσή του μπορεί να διαφέρει ανάλογα με την ταχύτητα λειτουργίας.

Σε χαμηλές ταχύτητες ο A4988 είναι συνήθως σε θέση να παράγει πολύ ακριβείς και ομαλές κινήσεις. Αυτό είναι ιδιαίτερα χρήσιμο στη συγκεκριμένη εφαρμογή αλλά και σε άλλες όπως για παράδειγμα οι 3D εκτυπωτές, όπου η ακρίβεια και η ανάλυση είναι κρίσιμης σημασίας. Επιπλέον, ο οδηγός αυτός υποστηρίζει microstepping, που προσφέρει ακόμη μεγαλύτερη ακρίβεια και ομαλότητα στην κίνηση σε χαμηλές ταχύτητες.

Σε υψηλές ταχύτητες αντιθέτως, όσο η ταχύτητα αυξάνεται, η απόδοση του A4988 μπορεί να μειωθεί λίγο. Αυτό οφείλεται κυρίως στη φυσική ιδιότητα των βηματικών κινητήρων, όπου η ροπή μειώνεται καθώς η ταχύτητα αυξάνεται. Παρ' όλα αυτά, με την κατάλληλη τροφοδοσία τάσης και ρεύματος, ο A4988 μπορεί να ελέγχει αποτελεσματικά έναν βηματικό κινητήρα σε υψηλές ταχύτητες σε πολλές εφαρμογές.

2.5.8 Microstepping

Ο μικροβηματισμός ή αλλιώς microstepping είναι μια τεχνική που επιτρέπει σε έναν βηματικό κινητήρα να κάνει μικρότερα βήματα από τα πλήρη βήματα που κάνει συνήθως. Αυτό γίνεται πολύ χρήσιμο όταν υπάρχει ανάγκη για πιο ακριβή και ομαλή κίνηση.

Ένας βηματικός κινητήρας που είναι ένα μοντέλο 200 βημάτων κάνει 200 πλήρη βήματα ανά περιστροφή, ή 1.8 μοίρες ανά βήμα. Με τον μικροβηματισμό, ο κινητήρας μπορεί να κάνει πολλαπλάσια του βήματος αυτού, όπως 1/2, 1/4, 1/8, 1/16, ή ακόμη και 1/32 του βήματος. Στη συγκεκριμένη περίπτωση ο οδηγός A4988 υποστηρίζει μέχρι και 1/16 microstepping. Αυτό επιτυγχάνεται με τον αποδοτικό έλεγχο των μαγνητικών πεδίων στα πηνία του κινητήρα, γεγονός που επιτρέπει την περιστροφή του ρότορα σε μικρότερες αποστάσεις από ό,τι στην πλήρη λειτουργία βήματος. Ο μικροβηματισμός βελτιώνει την ομαλότητα και την ακρίβεια της κίνησης, αλλά μπορεί επίσης να μειώσει ελαφρώς την επιτρεπόμενη ροπή του κινητήρα.

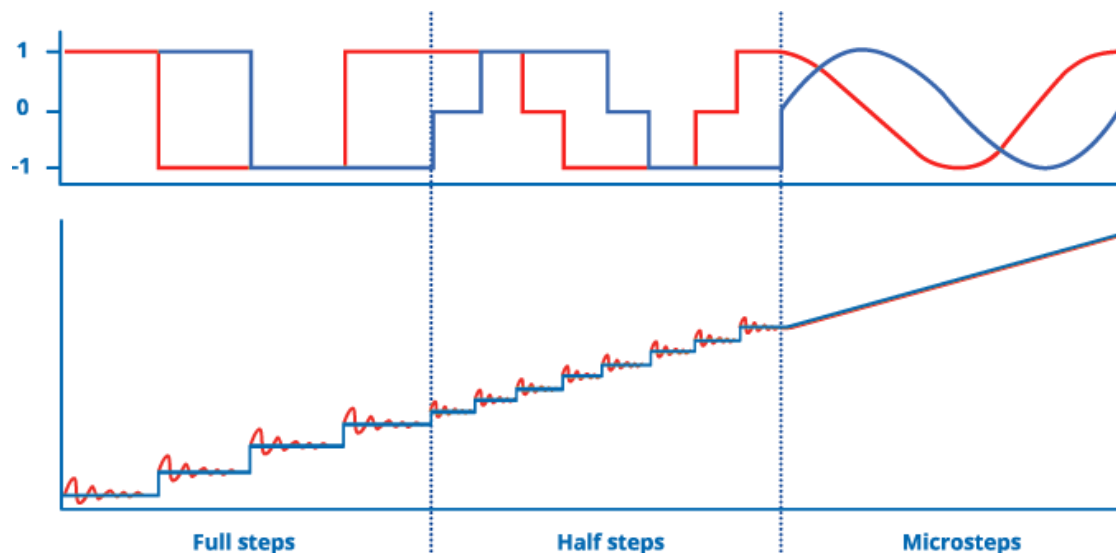
Οι ακροδέκτες MS1, MS2 και MS3 στον οδηγό βηματικού κινητήρα A4988 χρησιμεύουν για τον έλεγχο του μικροβηματισμού ρυθμίζοντας τον αριθμό των μικροβημάτων θα κάνει ο βηματικός κινητήρας ανά πλήρες βήμα. Με την αύξηση του αριθμού των μικροβημάτων, γίνεται η βελτίωση της ανάλυσης και της ομαλότητας της κίνησης του κινητήρα.

Ο παρακάτω πίνακας περιγράφει το τρόπο ελέγχου της ανάλυσης των μικροβημάτων.

MS3	MS2	MS1	ΑΝΑΛΥΣΗ
LOW	LOW	LOW	ΠΛΗΡΕΣ ΒΗΜΑ
LOW	LOW	HIGH	1/2 ΒΗΜΑ
LOW	HIGH	LOW	1/4 ΒΗΜΑΤΟΣ
LOW	HIGH	HIGH	1/8 ΒΗΜΑΤΟΣ
HIGH	HIGH	HIGH	1/16 ΒΗΜΑΤΟΣ

Πίνακας 2. 1 Πίνακας καταστάσεων μικροβηματισμού

Στην εφαρμογή αυτή με βάση τον παραπάνω λογικό πίνακα, για να πετύχουμε τη μέγιστη επιτρεπτή ανάλυση από το A4988 που είναι το 16 βήματα ανά πλήρες βήμα, τροφοδοτήθηκαν οι ακροδέκτες MS1, MS2 και MS3 5V, δηλαδή λογικό HIGH. Αυτό σημαίνει ότι ο κινητήρας θα κάνει 16 βήματα για κάθε πλήρες βήμα, αυξάνοντας την ακρίβεια και την ομαλότητα της κίνησης, αλλά μειώνοντας ελαφρώς την ροπή. [\[34\]](#)



Εικόνα 2. 19 Αναπαράσταση της ομαλότητας της κίνησης σε σχέση με την ανάλυση^[35]

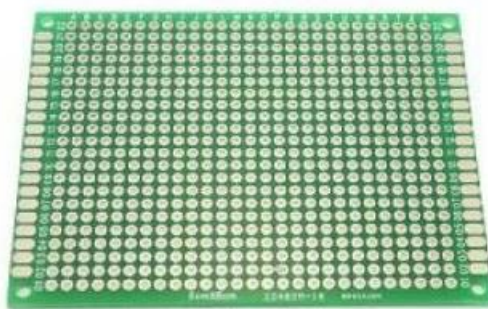
Σε αυτό το project, χρησιμοποιήθηκαν τρεις βηματικοί κινητήρες Nema 17 και ένας οδηγός A4988 για κάθε βηματικό κινητήρα, επιτυγχάνοντας προσεγμένο και έλεγχο της κίνησης μιας κάμερας. Κάθε βηματικός κινητήρας συνδυάζεται με έναν οδηγό A4988, που επιτρέπει τον ακριβή έλεγχο της κίνησής του. Ο πρώτος βηματικός κινητήρας ελέγχει τη μετατόπιση της κάμερας κατά μήκος της ράγας, προσφέροντας ευελιξία στην προβολή και την καταγραφή. Ο δεύτερος κινητήρας είναι υπεύθυνος για την περιστροφή της κάμερας, ενώ ο τρίτος ρυθμίζει την κλίση της, επιτρέποντας μια πλήρη, τρισδιάστατη βιντεογράφιση στον χώρο.

2.6 Υλοποίηση του κυκλώματος ελέγχου

2.6.1 Υλικά που χρησιμοποιήθηκαν

Παρακάτω περιγράφονται τα συνολικά τα υλικά που χρησιμοποιήθηκαν για την υλοποίηση του κυκλώματος ελέγχου της κίνησης του συστήματος.

1. Μια **διάτρητη πλακέτα** διαστάσεων 60x80 mm και 22x27 σημείων κόλλησης η οποία είναι τέλεια για τις διαστάσεις που προσφέρει για να χωρέσει όλα τα υπόλοιπα ηλεκτρονικά εξαρτήματα του κυκλώματος.



Εικόνα 2. 20 Διάτρητη πλακέτα 60x80mm

2. Μια πλακέτα μικροελεγκτή **Arduino Nano**. Όπως περιεγράφηκαν παραπάνω τα χαρακτηριστικά αυτού του μικροελεγκτή, είναι μια από τις μικρότερες πλακέτες της τεχνολογίας Arduino.
3. Ένα Bluetooth module **HC-05** για τη λήψη εντολών από μια ασύρματη συσκευή.
4. Δυο **αντιστάτες 1kΩ** και **2kΩ** αντίστοιχα, οι οποίοι θα τοποθετηθούν στο κύκλωμα σε συνδεσμολογία διαιρέτη τάσης στον ακροδέκτη RXD του HC-05 για να μετατρέψει την τάση στον ακροδέκτη από 5V στα 3.3V για τη προστασία του Bluetooth module.
5. Τρεις οδηγοί βηματικών κινητήρων **A4988**. Οι κινήσεις της κάμερας οι οποίες μας ενδιαφέρουν στο project αυτό, όπως ειπώθηκαν είναι (α) η μετατόπιση της κάμερας, (β) η περιστροφή και (γ) η κλήση της και θα εκτελούνται με βηματικούς κινητήρες. Έπομένως, για κάθε βηματικό κινητήρα αναθέτεται από έναν οδηγό.
6. Τρεις βηματικοί κινητήρες **Nema 17 59 Ncm**.
7. Τρεις **πυκνωτές 100μF**, οι οποίοι θα συνδεθούν παράλληλα στην τροφοδοσία και στις εισόδους V_{motor} κάθε οδηγού A4988 για την εξομάλυνση της τάσης τροφοδοσίας. Κατά την λειτουργία ενός βηματικού κινητήρα, τα ρεύματα στους πηνία αλλάζουν γρήγορα, δημιουργώντας παλμούς τάσης. Αυτοί οι παλμοί μπορεί να προκαλέσουν ασταθείς συνθήκες τροφοδοσίας και να προκαλέσουν προβλήματα στην λειτουργία του οδηγού κινητήρα. Εισάγοντας έναν πυκνωτή παράλληλα στην τροφοδοσία, αυτοί οι παλμοί τάσης μπορούν να εξομαλυνθούν. Ο πυκνωτής λειτουργεί εξισορροπώντας τους παλμούς τάσης, απορροφώντας το ρεύμα όταν η τάση αυξάνεται και απελευθερώνοντας το όταν η τάση πέφτει. Αυτό βοηθά στην διατήρηση μιας σταθερής τάσης τροφοδοσίας για τον οδηγό κινητήρα, βελτιώνοντας έτσι την συνολική απόδοση και αξιοπιστία του συστήματος.
8. Ένας **μετατροπέας DC-DC Step-Down 5V 1A**. Είναι μια μονάδα υψηλής απόδοσης και χαμηλής παραγωγής θερμότητας η οποία λειτουργεί ως ρυθμιστής τάσεων εισόδου DC από 5.5V έως 32V και εξάγει τάση 5V και ρεύμα 1A. Ένα χρήσιμο και αρκετά μικρό σε μέγεθος εξάρτημα το οποίο θα δώσει τη δυνατότητα του περιορισμού της τάσης και του ρεύματος σε ευαίσθητα εξαρτήματα όπως είναι ο μικροελεγκτής.



Εικόνα 2. 21 Μετατροπέας DC-DC Step-Down 5V 1A

9. Τέσσερις **Μικροδιακόπτες** τύπου SPDT. Θα χρησιμοποιηθούν για την ανίχνευση των ορίων στα οποία μπορούν να εκτελούνται οι κινήσεις της κάμερας. Ένας μικροδιακόπτης θα τοποθετηθούν στην αρχή της ράγας και ένας στο τέλος, έτσι ώστε όταν η κάμερα φτάσει στα όρια οι διακόπτες να δίνουν εντολή να σταματήσει η κίνησή της. Οι υπόλοιποι δυο μικροδιακόπτες θα τοποθετηθούν πάνω στη βάση της κάμερας για να περιορίσουν την κίνηση της περιστροφής της, για αποφυγή ζημίας του συστήματος.



Εικόνα 2. 22 Μικροδιακόπτης τύπου SPDT

10. Ένα **τροφοδοτικό DC 12V 2A** και ένας θηλυκός **DC Power jack** για τη τροφοδοσία του συστήματος.



Εικόνα 2. 23 Τροφοδοτικό και power jack

11. **Pin headers και καλώδια:**

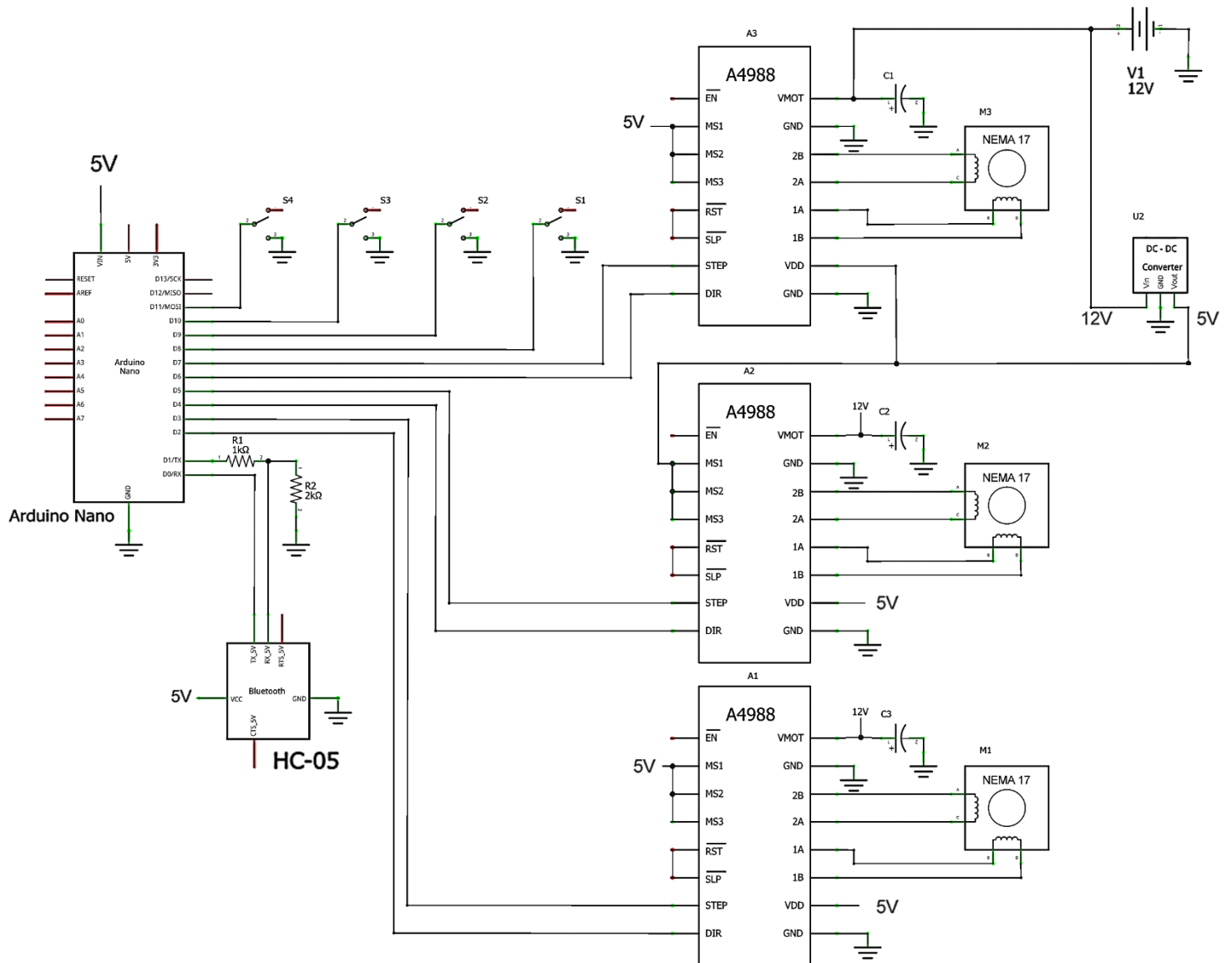
- Έξι θηλυκά pin headers 8-ακροδεκτών, δυο για κάθε οδηγό A4988.
- Δυο θηλυκά pin headers 15-ακροδεκτών για το Arduino Nano.
- Ένα θηλυκό pin header 6-ακροδεκτών για το Bluetooth module HC-05.
- Τρία αρσενικά pin headers 4-ακροδεκτών και τρία θηλυκά καλώδια 4-ακροδεκτών για τη σύνδεση κάθε βηματικού κινητήρα με το αντίστοιχο οδηγό του.
- Δυο αρσενικά pin headers 4-ακροδεκτών και τέσσερα καλώδια για τους μικροδιακόπτες.

2.6.2 Συνδεσμολογία του συστήματος

Σε αυτήν την υποενότητα παρουσιάζεται το κύκλωμα το οποίο ελέγχει την κατασκευή καθώς και η συνδεσμολογία όλων των ηλεκτρονικών εξαρτημάτων του κυκλώματος.

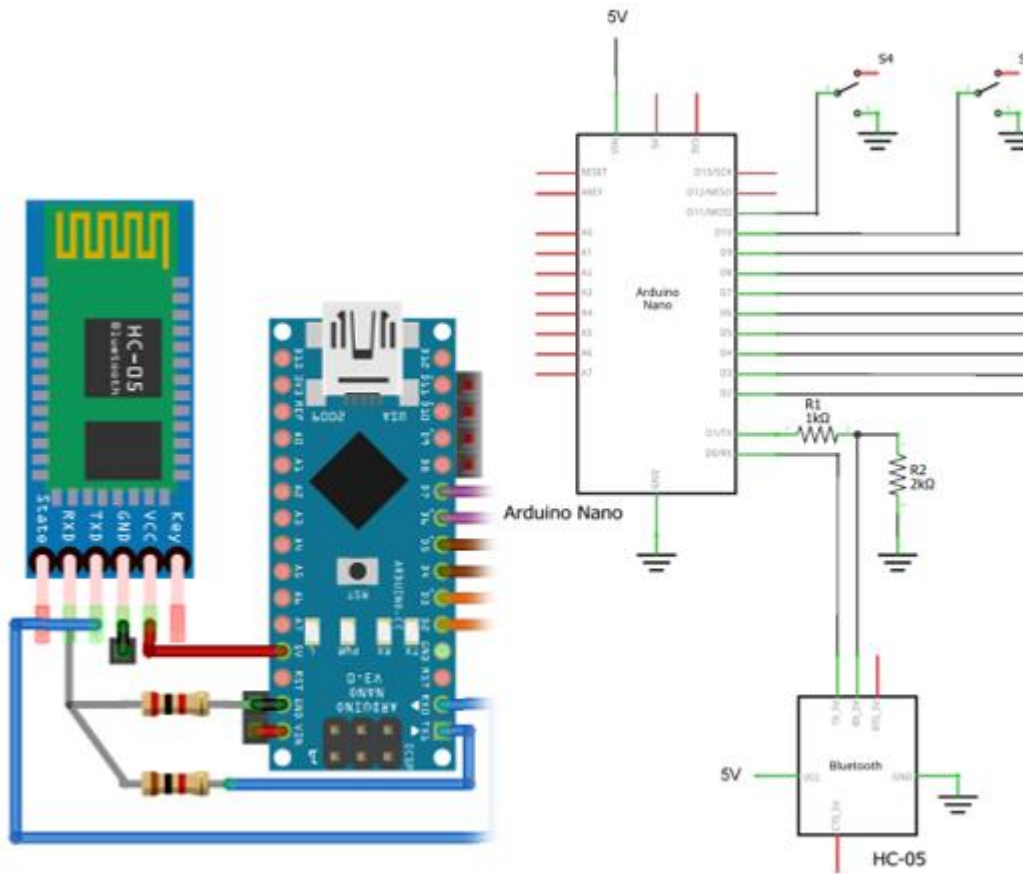
Το κύκλωμα σχεδιάστηκε μέσω του λογισμικού Fritzing, το οποίο παρέχει τη βέλτιστη αναπαράσταση του κυκλώματος.

Αριστερά του κυκλώματος βρίσκεται ο μικροελεγκτής του Arduino Nano ο οποίος συνδέεται με το Bluetooth module HC-05, τους οδηγούς των βηματικών κινητήρων A4988 και τους μικροδιακόπτες.



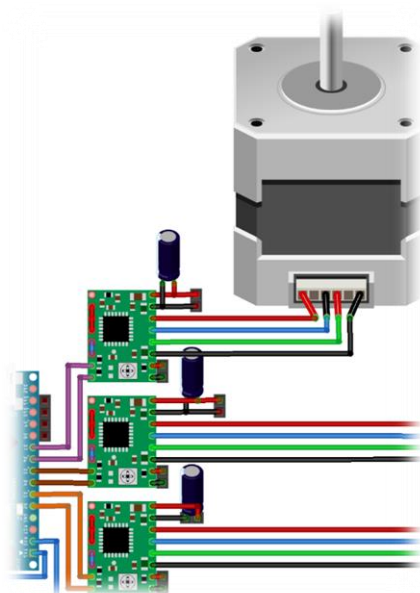
Εικόνα 2. 24 Κόκλωμα της εφαρμογής

Σημαντικό να σημειωθεί πως οι επαφές του Bluetooth module RX και TX συνδέονται αντίστροφα με τις επαφές του μικροελεγκτή Arduino. Συγκεκριμένα, η επαφή RX του Bluetooth module συνδέεται με την TX του Arduino, με ένα διεραϊτή τάσης, οποίος περιορίζει την τάση στα 3.3V όπως προϋποθέτει να έχει το Bluetooth module. Αντίστοιχα, η TX επαφή του Bluetooth module συνδέεται με την RX του Arduino.

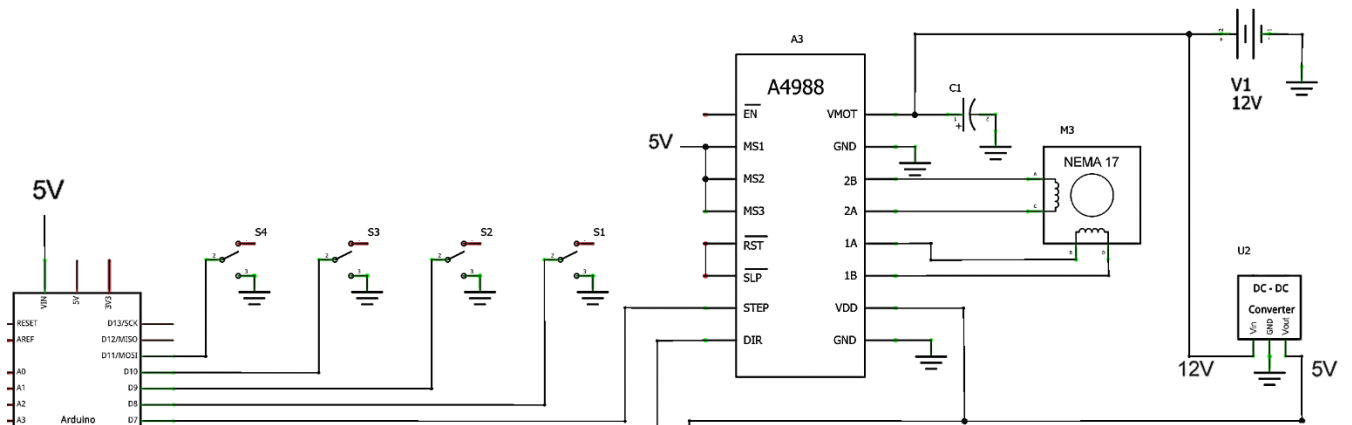


Εικόνα 2. 25 Συνδεσμολογία του HC-05 με το Arduino

Στις δυο παρακάτω εικόνες, παρουσιάζεται η συνδεσμολογία των οδηγών βηματικών κινητήρων με το μικροελεγκτή Arduino καθώς και με τους ίδιους βηματικούς κινητήρες.



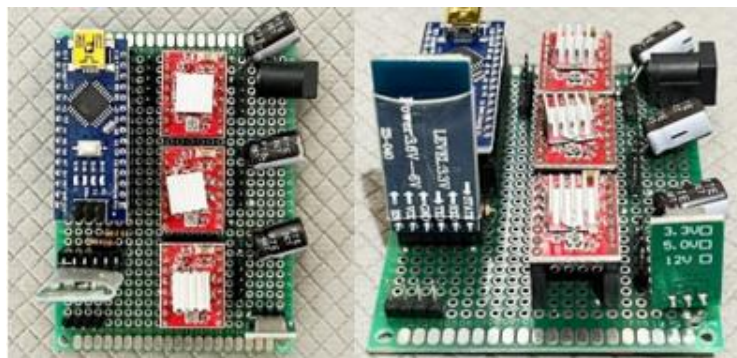
Εικόνα 2. 26 Συνδεσμολογία A4988 με το Arduino και τους βηματικούς κινητήρες



Εικόνα 2. 27 Συνδεσμολογία A4988 με το Arduino και λοιπών εξαρτημάτων

Όπως φαίνεται στο παραπάνω κύκλωμα, στις ψηφιακές επαφές εισόδου του Arduino συνδέονται τέσσερις μικροδιακόπτες, οι οποίοι θα διακόπτουν τη λειτουργία των κινητήρων όταν ενεργοποιούνται. Η ενεργοποίησή ενός μικροδιακόπτη πραγματοποιείται όταν ο διακόπτης «κλείσει», ο ψηφιακός ακροδέκτης ο οποίος του αντιστοιχεί θα λάβει τη τιμή του λογικού μηδέν ή αλλιώς LOW.

Σχετικά με τους οδηγούς βηματικών κινητήρων, οι επαφές τους: MS1, MS2 και MS3 συνδέονται με τροφοδοσία 5V, δηλαδή παρέχονται συνεχώς με λογικό HIGH, το οποίο ρυθμίζει τους οδηγούς να παρέχουν την υψηλότερη δυνατή ανάλυση μικροβημάτων, όπως σημειώθηκε στις προηγούμενες υποενότητες. Οι επαφές RESET και SLEEP βραχυκυκλώνονται έτσι ώστε το κύκλωμα του οδηγού να είναι ενεργοποιημένο, βάσει των πληροφοριών στο φύλλο δεδομένων ή αλλιώς datasheet του A4988. Οι ακροδέκτες STEP και DIR των οδηγών συνδέονται τους ακροδέκτες ψηφιακής εξόδου του μικροελεγκτή Arduino, από όπου θα λαμβάνουν σήματα για τον έλεγχο των κινητήρων. Οι βηματικοί κινητήρες συνδέονται στις επαφές 2B, 2A, 1B, 1A των οδηγών που αντιστοιχούν. Τέλος, παρέχεται τάση 12V στον ακροδέκτη VMOT για τη τροφοδοσία των βηματικών κινητήρων μέσω των οδηγών A4988 και τάση των 5V (η οποία προκύπτει από τη μετατροπή που γίνεται μέσω του κυκλώματος DC-DC Converter) στα κυκλώματα των οδηγών.



Εικόνα 2. 28 Ηλεκτρονικά εξαρτήματα τοποθετημένα σε διάτρητη πλακέτα

2.7 Κατασκευή του συστήματος

Παρακάτω παρουσιάζεται η κατασκευή του συστήματος κίνησης κάμερας με ασύρματο έλεγχο, τα υλικά που χρησιμοποιήθηκαν και ο τρόπος κατασκευής ενός τέτοιου συστήματος.

2.7.1 Υλικά που χρησιμοποιήθηκαν

Για την κατασκευή χρησιμοποιήθηκαν κομμάτια ξύλου τύπου MDF τα οποία ενώθηκαν με καρφιά και ξυλόκολλα και δυο χαλκοσωλήνες οι οποίοι θα λειτουργούν ως ράγες για τη κίνηση της μετατόπισης της κάμερας.

Για τα κινητά μέρη της κατασκευής εγκαταστάθηκαν έξι τροχαλίες ή αλλιώς timing pulley, όπου δυο αντιστοιχούν σε κάθε άξονα. Συγκεκριμένα, χρησιμοποιήθηκαν δυο τροχαλίες με 60 δόντια (με διάμετρο 40mm), μια για την κίνηση της κλίσης και η μια για την κίνηση της περιστροφής της κάμερας, οι οποίες ολοκληρώνονται η κάθε μια με μια τροχαλία με των 20 δοντιών (με διάμετρο 18mm). Τέλος, για την κίνηση της μετατόπισης χρησιμοποιήθηκε μια τροχαλία των 20 δοντιών και μια ίδιου μεγέθους τροχαλία τύπου idler.



Εικόνα 2. 29 Timing pulleys 60T, 20T, Idler, Ιμάντας και Ρουλεμάν

Επίσης, τοποθετήθηκαν ρουλεμάν στις θέσεις των οδηγούμενων τροχαλιών, οι οποίοι διευκολύνουν την περιστροφή των κινητών μερών της κατασκευής.

Τέλος, για το κινητό μέρος το οποίο εκτελεί τη μετατόπιση της κάμερας χρησιμοποιήθηκαν έξι ρόδες, εκ τις οποίες τρεις είναι για τη αριστερή και τρεις είναι για τη δεξιά ράγα. Οι ρόδες βρίσκονται πάνω σε κατάλληλους οδηγούς μορφής ράβδου, οι οποίοι είναι κολλημένοι πάνω στο ξύλινο κινητό μέρος.

2.7.2 Τρόπος κατασκευής

Το ξύλο MDF που χρησιμοποιήθηκε για την κατασκευή είχε πάχος 8mm και παρακάτω αναφέρονται οι διαστάσεις κάθε μέρους της κατασκευής.

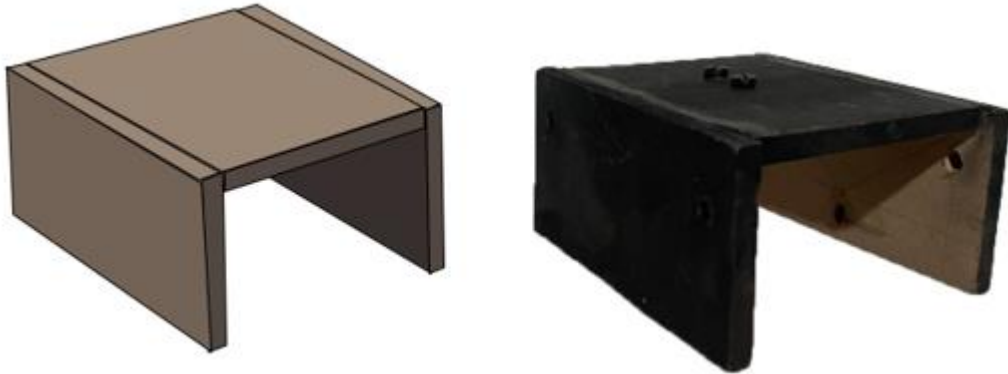
Για τη βάση χρειάζονται:

Κάτω μέρος

- 2 κομμάτια διαστάσεων 15cm x 7cm
- 1 κομμάτι διαστάσεων 15cm x 10cm

Το συγκεκριμένο μέρος της κατασκευής θα το ελέγχει ένας από τους βηματικούς κινητήρες ο οποίος θα βρίσκεται τοποθετημένος στην μία άκρη των χαλκοσωλήνων, με τη βοήθεια ενός

ιμάντα, μια τροχαλία 20 δοντιών εγκαταστημένη πάνω στον κινητήρα και μια τροχαλία τύπου idler τοποθετημένη στο άλλο άκρο των χαλκοσωλήνων.

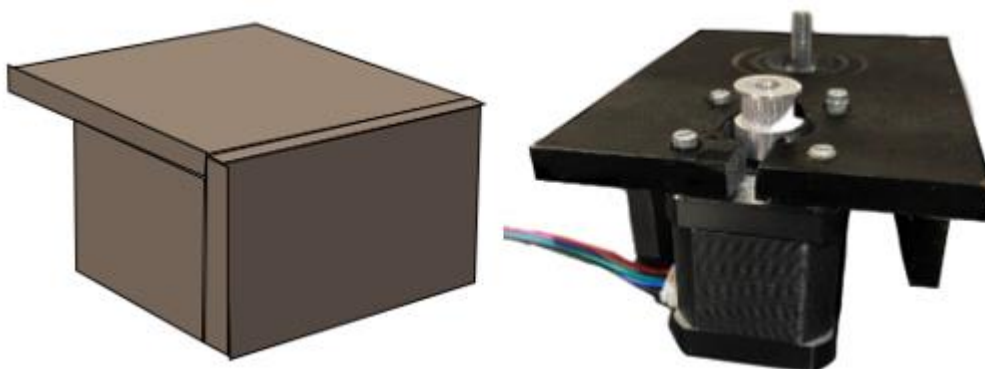


Εικόνα 2. 30 Κάτω μέρος τη βάσης

Επάνω μέρος

- 1 κομμάτι διαστάσεων 13.5cm x 10cm
- 2 κομμάτια διαστάσεων 8cm x 5cm
- 1 κομμάτι διαστάσεων 10cm x 6cm

Στην εξοχή του επάνω κομματιού του μέρους αυτού, θα τοποθετηθεί ένας από τος βηματικούς κινητήρες. Επάνω στον κινητήρα θα είναι τοποθετημένη μια τροχαλία 20 δοντιών, η οποία θα ενώνεται με μια τροχαλία 60 δοντιών μέσω ενός ιμάντα. Ο συγκεκριμένος κινητήρας θα ελέγχει την κίνηση του ακριβώς επάνω κομματιού, δηλαδή του μέρους το οποίο ελέγχει την περιστροφή της κάμερας

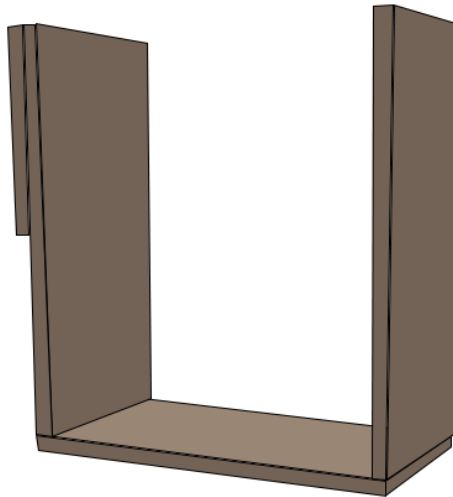


Εικόνα 2. 31 Επάνω μέρος τη βάσης

Για την κινητό μέρος περιστροφής (pan) χρειάζονται:

- 2 κομμάτια διαστάσεων 21cm x 10cm
- 1 κομμάτι διαστάσεων 19cm x 10cm
- 1 κομμάτι διαστάσεων 10cm x 10cm
- 4 γωνίες στήριξης

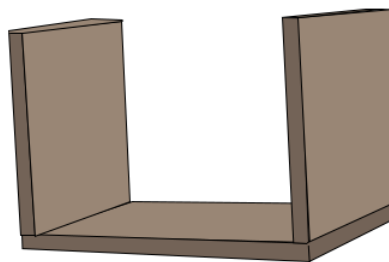
Στο ψηλό κομμάτι του αριστερού μέρους θα τοποθετηθεί ένας από τους τρεις βηματικούς κινητήρες για να ελέγχει την κίνηση του κινητού μέρους που ελέγχει την κλίση της κάμερας. Επάνω στον κινητήρα θα τοποθετηθεί μια τροχαλία 20 δοντιών η οποία θα κινεί μια τροχαλία 60 δοντιών, τοποθετημένη στο επάνω άκρο του κομματιού για επιτευχθεί ο έλεγχος αυτός.



Εικόνα 2. 32 Κινητό μέρος περιστροφής (Pan)

Για το κινητό μέρος κλίσης (tilt) χρειάζονται:

- 1 κομμάτι διαστάσεων 10cm x 15.5cm
- 2 κομμάτια διαστάσεων 10cm x 10cm



Εικόνα 2. 33 Κινητό μέρος κλίσης (Tilt)



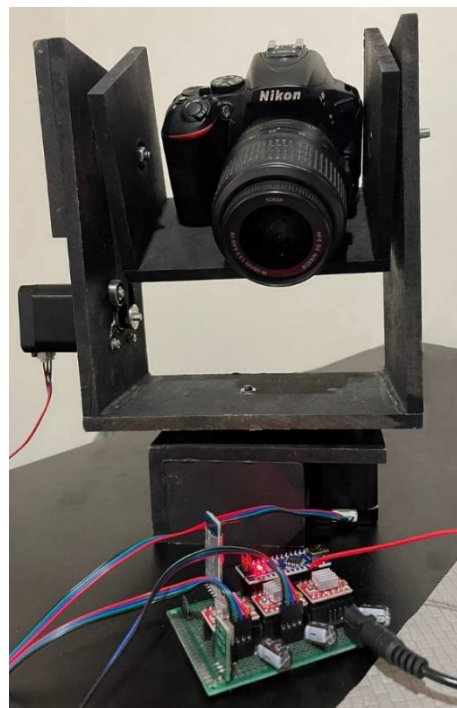
Εικόνα 2. 34 Ένωση κινητών μερών περιστροφής και κλίσης (Pan και Tilt)

Τα ζεύγη των τροχαλιών ανά άξονα που αναφέρθηκαν παραπάνω «κλείνουν» με έναν μιάντα timing belt το καθένα και ασφαλίζονται με βίδες και παξιμάδια.

Οι τροχαλίες των 20 δοντιών λειτουργούν ως τροχαλίες οδηγού, ενώ των 60 δοντιών ως τροχαλίες φορτίου. Η παρακάτω σχέση περιγράφει την ταχύτητα περιστροφής της τροχαλίας φορτίου.

$$RPM \text{ φορτίου} = \frac{RPM \text{ τροχαλίας οδηγού}}{\text{Αναλογία τροχαλιών}}$$

Με βάση την παραπάνω σχέση, προσδιορίζεται εύκολα η ταχύτητα περιστροφής των τροχαλιών φορτίου, η οποία θα είναι το 33% της ταχύτητας περιστροφής εισόδου, αφού η αναλογία των τροχαλιών είναι $\frac{60 \text{ Δόντια}}{20 \text{ Δόντια}} = 3$. [\[36\]](#)

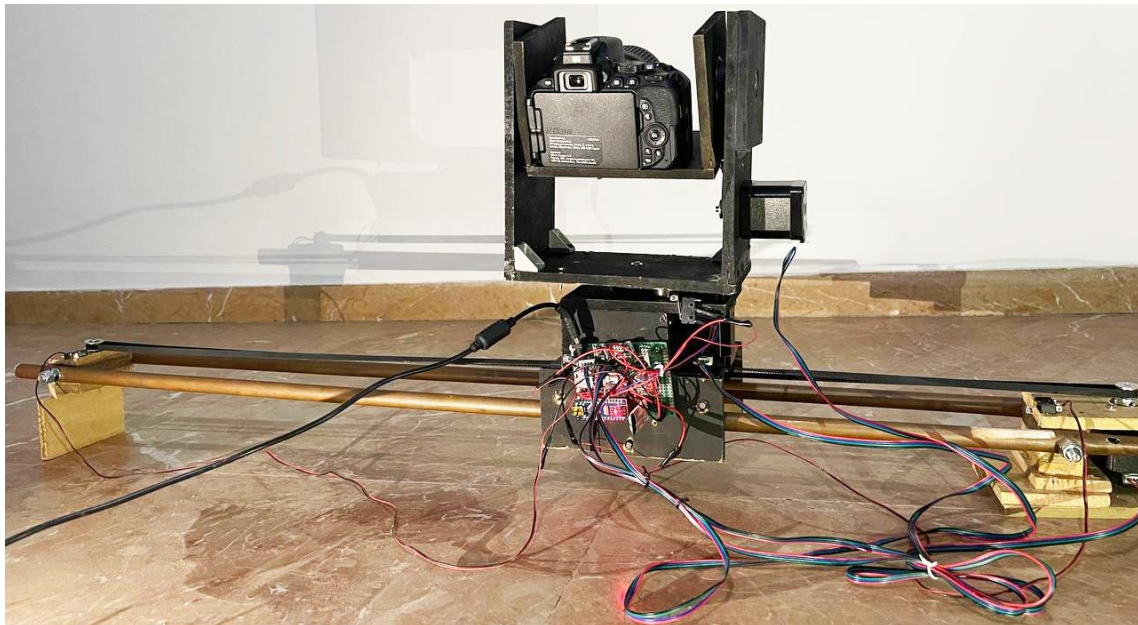


Εικόνα 2. 35 Μέρος κατασκευής σε λειτουργία

Τέλος, στην παρακάτω εικόνα αναπαρίσταται σε τρισδιάστατη μορφή το τελικό αποτέλεσμα της κατασκευής με ενωμένα όλα τα κινητά μέρη της.



Εικόνα 2. 36 Προσομοίωση κατασκευής συστήματος ελέγχου κίνησης κάμερας



Εικόνα 2. 37 Κατασκευή συστήματος ελέγχου κίνησης κάμερας εν κινήσει

3 ΚΕΦΑΛΑΙΟ 3^ο : Διασύνδεση εφαρμογής smartphone με Arduino και Ανάλυση του Κώδικα

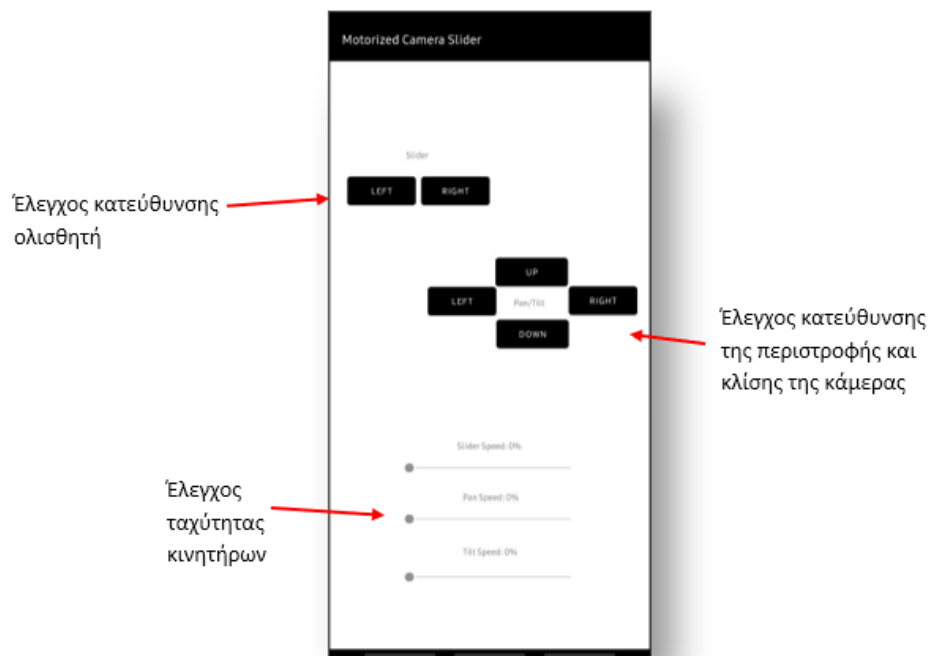
3.1 Εισαγωγή Κεφαλαίου 3

Σε αυτό το Κεφάλαιο, παρουσιάζεται η πλήρης υλοποίηση της εφαρμογής για smartphone σε περιβάλλον Android Studio, καθώς και ο αντίστοιχος κώδικας που αναπτύχθηκε για τη δόμηση της εφαρμογής, με σκοπό τον έλεγχο του συστήματος Arduino. Μέσα από αυτήν την αναλυτική παρουσίαση, δίνεται έμφαση στα βήματα που ακολουθήθηκαν για τη διασύνδεση του smartphone με το Arduino, επιτρέποντας την αμφίδρομη επικοινωνία και έλεγχο του συστήματος. Επιπλέον, αναλύεται ο κώδικας που χρησιμοποιήθηκε στο περιβάλλον Arduino για την ακριβή υλοποίηση των λειτουργιών ελέγχου που προσφέρει το σύστημα. Μέσα από αυτήν την ολοκληρωμένη παρουσίαση, αναδεικνύονται οι τεχνικές πτυχές της ανάπτυξης και ο συντονισμός μεταξύ των δύο περιβαλλόντων προγραμματισμού, προσφέροντας μια εμπειρισταωμένη κατανόηση της λειτουργίας και της δομής του ενσωματωμένου συστήματος που αναπτύχθηκε.

3.2 Υλοποίηση εφαρμογής σε περιβάλλον Android

Παρακάτω παρουσιάζεται το Περιβάλλον Χρήστη ή αλλιώς Διεπαφή Χρήστη (UI ή User Interface) της εφαρμογής που υλοποιήθηκε σε το λειτουργικό σύστημα Android καθώς αναλύεται πλήρως ο κώδικας σε γλώσσα Java που χρησιμοποιήθηκε για την υλοποίησή της εφαρμογής.

Στο παρακάτω στιγμιότυπο παρουσιάζεται το Περιβάλλον Χρήστη σε κατάσταση ελέγχου της ασύρματης συσκευής.



Εικόνα 3. 1 Διεπαφή Χρήστη

Εισαγωγή χρησιμων τάξεων (classes), όπως έτοιμες τάξεις που συμπεριλαμβάνουν κώδικα για την επικοινωνία Bluetooth, στοιχεία UI, κλπ.

```
package com.example.cammotorizedslider;

import android.Manifest;
import android.annotation.SuppressLint;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.pm.PackageManager;
import android.net.Uri;
import android.os.Build;
import android.os.Bundle;
import android.provider.Settings;
import android.util.Log;
import android.widget.Button;
import android.widget.SeekBar;
import android.widget.TextView;
import android.widget.Toast;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

import java.nio.charset.Charset;
import java.nio.charset.StandardCharsets;
import java.util.UUID;
import java.util.concurrent.atomic.AtomicReference;
```

Η τάξη Main Activity και αρχική δήλωση μεταβλητών και αντικειμένων. Βασικό να σημειωθεί ότι εδώ δηλώνεται η διεύθυνση MAC του Bluetooth Module HC-05 σε μορφή string. Επίσης, δηλώνεται το αναγνωριστικό ταυτοποίησης UUID σε μορφή string με κωδικό ο οποίος χρησιμοποιείται για την επικοινωνία συσκευών Bluetooth.

```
public class MainActivity extends AppCompatActivity {
    private static final String TAG = "MainActivity";
    Charset asciiCharset = StandardCharsets.UTF_8;
    private static final UUID mUUID =
        UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");
    private static final int REQUEST_ENABLE_BT = 0;
    private static final int REQUEST_BLUETOOTH_ADMIN_PERMISSION = 1;
    private static final int ENABLE_BT = 1;
    private static final int CONNECT = 2;
    BluetoothAdapter BTAdapter;
    BluetoothDevice BTDevice;
    BluetoothConnectionService BluetoothConnection;
    TextView sliderTxt, panTxt, sliderspeedTxt, panspeedTxt, tiltspeedTxt;
    SeekBar sliderSpeed, panSpeed, tiltSpeed;
    Button sliderLeftbtn, sliderRightbtn, panLeftbtn, panRightbtn, tiltUpbtn, tiltDownbtn;

    String MACAddress = "98:D3:11:FC:69:14";
    private AlertDialog alertDialog = null;
    AtomicReference<Boolean> isAlertDialogDisplayed = new AtomicReference<>(false);
```

Οι Broadcast Receivers λειτουργούν ως παρακολουθητές, επιτρέποντας στις εφαρμογές να ακούν και να ανταποκρίνονται σε διακοπές μεταδόσεων (broadcasts) που μπορούν να προέρχονται από

το σύστημα, άλλες εφαρμογές ή το περιβάλλον της ίδιας της εφαρμογής. Αυτή η μοναδική δυνατότητα επιτρέπει στις εφαρμογές να ανταποκρίνονται δυναμικά σε γεγονότα όπως αλλαγές στην κατάσταση του δικτύου, ειδοποιήσεις ή ακόμα και αλλαγές στην κατάσταση της μπαταρίας. Μέσω της εγγραφής σε συγκεκριμένες προθέσεις μετάδοσης, οι εφαρμογές μπορούν να προσαρμοστούν και να ανταποκριθούν σε διάφορα σενάρια, επιτρέποντας την ομαλή διακοπή των εργασιών τους με βάση τις αλλαγές που συμβαίνουν στο περιβάλλον του συστήματος.

Παρακάτω αναπτύσσεται κώδικας του πρώτου Broadcast Receiver ο οποίος παρακολουθεί την κατάσταση λειτουργίας του Bluetooth, όπως η ανενεργή κατάσταση (State Off), κατάσταση προς απενεργοποίηση (State Turning Off), ενεργή κατάσταση (State On), κατάσταση προς ενεργοποίηση (State Turning On). Σημαντικό να σημειωθεί πως σε αυτό το Broadcast Receiver δίνεται έμφαση η ανενεργή κατάσταση, όπου καλείται η μέθοδος `bluetooth_is_disabled()` και επίσης δίνεται έμφαση στην ενεργή κατάσταση, όπου καλείται η μέθοδος κατά την οποία ο χρήστης ερωτάται για τη σύνδεση με την ασύρματη συσκευή.

```
// Create a BroadcastReceiver for ACTION_FOUND
private final BroadcastReceiver BroadcastReceiver1 = new BroadcastReceiver() {
    public void onReceive(Context context, @NonNull Intent intent) {
        String action = intent.getAction();
        // When discovery finds a device
        if (action.equals(BTAdapter.ACTION_STATE_CHANGED)) {
            final int state = intent.getIntExtra(BluetoothAdapter.EXTRA_STATE, BTAdapter.ERROR);

            switch (state) {
                case BluetoothAdapter.STATE_OFF:
                    bluetooth_is_disabled();
                    break;
                case BluetoothAdapter.STATE_TURNING_OFF:
                    break;
                case BluetoothAdapter.STATE_ON:
                    show_dialog(CONNECT);
                    break;
                case BluetoothAdapter.STATE_TURNING_ON:
                    break;
            }
        }
    }
};
```

Το δεύτερο Broadcast Receiver παρακολουθεί την κατάσταση σύζευξης με την ασύρματη συσκευή. Εάν έχει γίνει η σύζευξη καλείται η μέθοδος η οποία στη συνέχεια εκτελεί τη λειτουργία σύνδεσης με την ασύρματη συσκευή. Όταν γίνεται η σύζευξη, εμφανίζεται μήνυμα σε μορφή Toast στην οθόνη το οποίο ειδοποιεί το χρήστη για την κατάσταση. Στην περίπτωση που δεν έχει γίνει η σύζευξη με την επιθυμητή ασύρματη συσκευή, εμφανίζεται παράθυρο διαλόγου όπου ερωτάται ο χρήστης για τη σύνδεση με την ασύρματη συσκευή. Εάν ο χρήστης αποδεχτεί το μήνυμα αρχικά πραγματοποιείται η σύζευξη με την ασύρματη συσκευή και στη συνέχεια η σύνδεση με αυτή.

```

/**
 * Broadcast Receiver that detects bond state changes (Pairing status changes)
 */
private final BroadcastReceiver BondBroadcastReceiver = new BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {
        final String action = intent.getAction();

        if(action.equals(BluetoothDevice.ACTION_BOND_STATE_CHANGED)){
            BluetoothDevice mDevice = intent.getParcelableExtra(BluetoothDevice.EXTRA_DEVICE);
            //3 cases:
            //case1: bonded already
            if (mDevice.getBondState() == BluetoothDevice.BOND_BONDED){
                //BTDevice = mDevice;
                connect_to_device();
            }
            //case2: creating a bone
            if (mDevice.getBondState() == BluetoothDevice.BOND_BONDING) {
                Toast.makeText(MainActivity.this, "Pairing... ", Toast.LENGTH_SHORT).show();
            }
            //case3: breaking a bond
            if (mDevice.getBondState() == BluetoothDevice.BOND_NONE) {
                show_dialog(CONNECT);
            }
        }
    }
};

```

Το παρακάτω Broadcast Receiver παρακολουθεί την κατάσταση σύνδεσης με την ασύρματη συσκευή. Όπως φαίνεται και στον κώδικα, εάν πραγματοποιηθεί επιτυχημένα η σύνδεση με την ασύρματη συσκευή, ενεργοποιούνται τα εικονικά πλήκτρα στη Διεπαφή Χρήστη και εμφανίζεται μήνυμα το οποίο ενημερώνει τον χρήστη για την επιτυχή σύνδεση. Ωστόσο, εάν αποτύχει η σύνδεση με τη συσκευή, απενεργοποιούνται τα εικονικά πλήκτρα στην οθόνη και πραγματοποιείται ο έλεγχος της κατάστασης λειτουργίας Bluetooth.

```

private final BroadcastReceiver mReceiver = new BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {
        String action = intent.getAction();

        if (action.equals(BluetoothDevice.ACTION_ACL_CONNECTED)) {
            enable_ui(true);
            Toast.makeText(MainActivity.this, "Connected to device.", Toast.LENGTH_SHORT).show();
        } else if (action.equals(BluetoothDevice.ACTION_ACL_DISCONNECTED)) {
            enable_ui(false);
            check_bluetooth();
        }
    }
};

```

Η μέθοδος onDestroy() καλείται όταν τερματίζεται η εφαρμογή και χρησιμοποιείται για τη αποδέσμευση των Broadcast Receiver με στόχο την αποδέσμευση πόρων που έχουν δεσμευτεί από την εφαρμογή για την απελευθέρωση πόρων μνήμης.

```

@Override
protected void onDestroy() {
    super.onDestroy();
    unregisterReceiver(BroadcastReceiver1);
    unregisterReceiver(BroadcastReceiver2);
    unregisterReceiver(BondBroadcastReceiver);
    unregisterReceiver(mReceiver);
}

```

Δημιουργία μεθόδου onCreate, η οποία καλείται κατά την εκκίνηση της εφαρμογής για τη φόρτωση δεδομένων και την αρχικοποίηση βασικών στοιχείων του User Interface. Βασικό να σημειωθεί ότι σε αυτή τη μέθοδο τα βασικά στοιχεία της Διεπαφής Χρήστη όπως τα εικονικά πλήκτρα (Buttons), ετικέτες (TextView) και ελεγχόμενες μπάρες (SeekBar) καταχωρούνται σε μεταβλητές για να μπορούν χρησιμοποιηθούν σε συναρτήσεις και μεθόδους. Επίσης, γίνεται η δήλωση και αρχικοποίηση των φίλτρων προθέσεων (Intent Filters) οι οποίοι θα βοηθήσουν σε μεγάλο βαθμό στην ανίχνευση της κατάστασης λειτουργίας του Bluetooth, όπως το απενεργοποιημένο, προς ενεργοποίηση, ενεργοποιημένο, προς σύνδεση με συσκευή, συνδεδεμένο, αποστολή δεδομένων κλπ.

```

@SuppressLint("ResourceType")
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    BTAdapter = BluetoothAdapter.getDefaultAdapter();

    IntentFilter filter1 = new IntentFilter(BTAdapter.ACTION_SCAN_MODE_CHANGED);
    registerReceiver(BroadcastReceiver2, filter1);

    IntentFilter filter2 = new IntentFilter(BluetoothDevice.ACTION_BOND_STATE_CHANGED);
    registerReceiver(BondBroadcastReceiver, filter2);

    sliderTxt = findViewById(R.id.sliderSpeed_txt);
    pantiltTxt = findViewById(R.id.pan_tilt_txt);
    sliderspeedTxt = findViewById(R.id.sliderSpeed_txt);
    pantiltSpeedTxt = findViewById(R.id.pantiltSpeed_txt);

    sliderSpeed = findViewById(R.id.sliderSpeed_bar);
    sliderLeftbtn = findViewById(R.id.sliderLeft_btn);
    sliderRightbtn = findViewById(R.id.sliderRight_btn);

    pantiltSpeed = findViewById(R.id.pantiltSpeed_bar);
    panLeftbtn = findViewById(R.id.panleft_btn);
    panRightbtn = findViewById(R.id.panright_btn);
    tiltUpbtn = findViewById(R.id.tiltup_btn);
    tiltDownbtn = findViewById(R.id.tiltDown_btn);

    sliderSpeed.setProgress(0);
    sliderSpeed.setMax(100);
}

```

Κλήση μεθόδου η οποία απενεργοποιεί τα πλήκτρα στην οθόνη έως ότου να γίνει σύνδεση με επιθυμητή συσκευή Bluetooth καθώς και της μεθόδου η οποία ελέγχει τη κατάσταση του Bluetooth. Παρακάτω θα παρουσιαστεί αναλυτικά ο κώδικας των δυο αυτών μεθόδων.

```
enable_ui(false);

check_bluetooth();
```

Για κάθε πλήκτρο της Διεπαφής Χρήστη συντάσσεται μια μέθοδος `setOnClickListener()`. Η μέθοδος αυτή καλείται όταν ο χρήστης αλληλοεπιδρά με τα πλήκτρα στην οθόνη, πατώντας τα. Όπως φαίνεται στον παρακάτω κώδικα, κατά τη κλήση της μεθόδου καλείται η μέθοδος `outstream()` η οποία στην ουσία μεταδίδει δεδομένα μέσω προς την ασύρματη συσκευή σε μορφή `String`. Σε κάθε πλήκτρο αντιστοιχεί ένα ξεχωριστό `string`, το οποίο λειτουργεί ως εντολή.

```
sliderLeftbtn.setOnClickListener(v -> outstream("SLIDERLEFT"));
sliderRightbtn.setOnClickListener(v -> outstream("SLIDERRIGHT"));
panLeftbtn.setOnClickListener(v -> outstream("PANLEFT"));
panRightbtn.setOnClickListener(v -> outstream("PANRIGHT"));
tiltUpbtn.setOnClickListener(v -> outstream("TILTUP"));
tiltDownbtn.setOnClickListener(v -> outstream("TILTDOWN"));
```

Αντίστοιχα για τις μπάρες ελέγχου ταχύτητας των βηματικών κινητήρων, παρακάτω συντάσσονται μέθοδοι οι οποίοι καλούνται κατά την αλληλεπίδραση του χρήστη με αυτές στην οθόνη. Κατά την επιλογή του ποσοστού της μπάρας με εύρος 0-100% γίνεται κλήση της μεθόδου `outstream()` όπου μεταδίδεται σε μορφή `string` το δεδομένου του ποσοστού της μπάρας καθώς η αλλαγή της αντιστοιχί ετικέτας στην οθόνη για να γνωρίζει ο χρήστη το ποσοστό που έχει επιλέξει.

```
sliderSpeed.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
        if (sliderTxt != null) {
            sliderTxt.setText("Slider Speed: " + String.valueOf(progress) + "%");
        }
    }

    @Override
    public void onStartTrackingTouch(SeekBar seekBar) {
    }

    @Override
    public void onStopTrackingTouch(SeekBar seekBar) {
        int progress = seekBar.getProgress();
        if (progress < 10) outstream("s00" + String.valueOf(progress));
        else if (progress >= 10 && progress < 100) outstream("s0" + String.valueOf(progress));
        else outstream("s" + String.valueOf(progress));
    }
});
```

```
panSpeed.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener(){
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
        if (panspeedTxt != null) {
            panspeedTxt.setText("Pan Speed: " + String.valueOf(progress) + "%");
        }
    }
    @Override
    public void onStartTrackingTouch(SeekBar seekBar) {
    }
    @Override
    public void onStopTrackingTouch(SeekBar seekBar) {
        int progress = seekBar.getProgress();
        if(progress < 10) ostream("p00" + String.valueOf(progress));
        else if (progress >= 10 && progress < 100) ostream("p0" + String.valueOf(progress));
        else ostream("p" + String.valueOf(progress));
    }
});
```

```
tiltSpeed.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener(){
    @Override
    public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
        if (tiltspeedTxt != null) {
            tiltspeedTxt.setText("Tilt Speed: " + String.valueOf(progress) + "%");
        }
    }
    @Override
    public void onStartTrackingTouch(SeekBar seekBar) {
    }
    @Override
    public void onStopTrackingTouch(SeekBar seekBar) {
        int progress = seekBar.getProgress();
        if(progress < 10) ostream("t00" + String.valueOf(progress));
        else if (progress >= 10 && progress < 100) ostream("t0" + String.valueOf(progress));
        else ostream("t" + String.valueOf(progress));
    }
});
}
```

Ο κώδικας της μεθόδου η οποία ενεργοποιεί ή απενεργοποιεί τα στοιχεία της Διεπαφής Χρήστη στην οθόνη όπως επισημάνθηκε παραπάνω:

```
private void enable_ui(Boolean toggle) {
    sliderLeftbtn.setClickable(toggle);
    sliderRightbtn.setClickable(toggle);

    if(toggle == false){
        sliderSpeed.setProgress(0);
        panSpeed.setProgress(0);
        tiltSpeed.setProgress(0);
    }

    panLeftbtn.setClickable(toggle);
    panRightbtn.setClickable(toggle);

    tiltUpbtn.setClickable(toggle);
    tiltDownbtn.setClickable(toggle);

    sliderSpeed.setClickable(toggle);
    panSpeed.setClickable(toggle);
    tiltSpeed.setClickable(toggle);
}
}
```

Μέθοδος η οποία καλείται για να ελέγξει την κατάσταση του Bluetooth. Σε περίπτωση που το Bluetooth είναι απενεργοποιημένο καλείται η μέθοδος `bluetooth_is_disabled()` η οποία εμφανίζει ένα παράθυρο διαλόγου όπου γίνεται ερώτηση στον χρήστη για την ενεργοποίηση του Bluetooth. Η μέθοδος ζητά επίσης άδειες χρήσης Bluetooth εάν δεν έχουν χορηγηθεί.

```
private void check_bluetooth() {
    if (BTAdapter == null) {
        Toast.makeText(MainActivity.this, "Bluetooth not supported", Toast.LENGTH_SHORT).show();
        return;
    }
    if (!BTAdapter.isEnabled()) {
        try {
            bluetooth_is_disabled();
            //checkBTPermissions();
        }
        catch (Exception e) {
            requestBluetoothAdminPermission();
        }
    }
    else{
        show_dialog(CONNECT);
    }
}
}
```

Η μέθοδος `enable_Bluetooth()` η οποία προβαίνει σε φάση ενεργοποίησης του Bluetooth.

```
private void bluetooth_is_disabled() {
    show_dialog(ENABLE_BT);
}
public void enable_BlueTooth() {
    Intent enableBtIntent = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
    startActivityForResult(enableBtIntent, REQUEST_ENABLE_BT);

    IntentFilter filter = new IntentFilter(BluetoothAdapter.ACTION_STATE_CHANGED);
    registerReceiver(BroadcastReceiver1, filter);
}
}
```

Η παρακάτω μέθοδος καλείται μέσα από παράθυρο διαλόγου μόνο όταν το Bluetooth είναι ενεργοποιημένο. Αυτή η μέθοδος προβαίνει σε σύζευξη με επιλεγμένη ασύρματη συσκευή.

```
private void pair_to_device(){
    BTAdapter.cancelDiscovery();
    BTDevice = BluetoothAdapter.getDefaultAdapter().getRemoteDevice(MACAddress);
    //Check if already paired.
    if (BTDevice.getBondState() == BluetoothDevice.BOND_BONDED) {
        connect_to_device();
    }
    else{
        //create the bond.
        //NOTE: Requires API 17+? I think this is JellyBean
        if(Build.VERSION.SDK_INT > Build.VERSION_CODES.N){
            BTDevice.createBond();

            BluetoothConnection = new BluetoothConnectionService(MainActivity.this);
        }
    }
}
```

Η μέθοδος connect_to_device() καλείται μέσω της παραπάνω μεθόδου pair_to_device(). Αφού έχει γίνει η σύζευξη των συσκευών, αυτή η μέθοδος προβαίνει στη σύνδεσή τους.

```
private void connect_to_device() {
    try{
        BluetoothConnection = new BluetoothConnectionService(MainActivity.this);
        BluetoothConnection.startClient(BTDevice,mUUID);
    }
    catch (NullPointerException e){
        Toast.makeText(MainActivity.this, "Unable to connect. Please try again.",
Toast.LENGTH_SHORT).show();
        show_dialog(CONNECT);
    }

    IntentFilter filter = new IntentFilter();
    filter.addAction(BluetoothDevice.ACTION_ACL_CONNECTED);
    filter.addAction(BluetoothDevice.ACTION_ACL_DISCONNECTED);
    registerReceiver(mReceiver, filter);
}
```

Η παρακάτω μέθοδος show_dialog() δημιουργεί ένα παράθυρο διαλόγου στη Διεπαφή Χρήστη. Παίρνει ως παράμετρο κατά την κλήση της μια εντολή ενέργειας η οποία αναπαρίσταται ένα ακέραιο αριθμό. Το παράθυρο διαλόγου όπως φαίνεται παρακάτω στον κώδικα εμφανίζεται στην

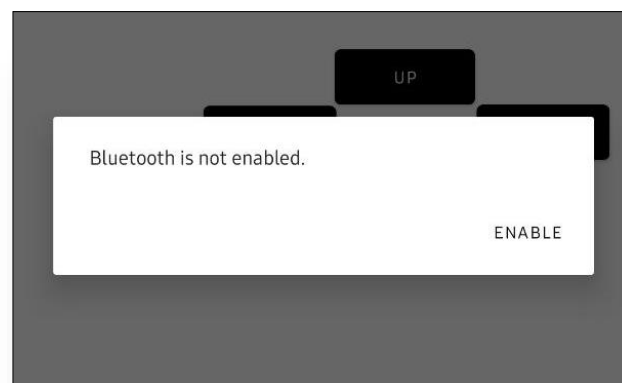
οθόνη σε δυο καταστάσεις, στην κατάσταση όπου ο ερωτάται χρήστης στην ενεργοποίηση του Bluetooth ή στη σύζευξη ή σύνδεση συσκευής.

```
private void show_dialog(Integer action) {
    //Closes other alert dialogs if there are any shown
    close_any_dialogs();

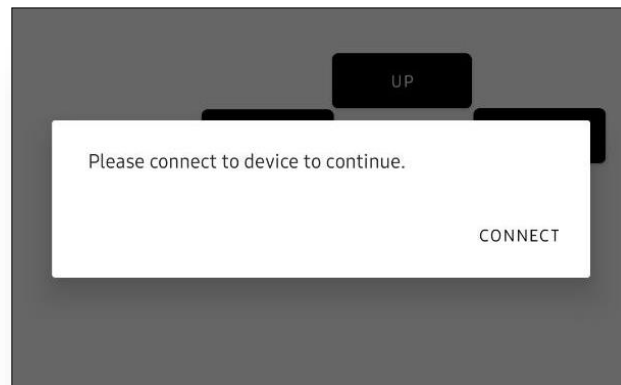
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setCancelable(false);
    if (action.equals(ENABLE_BT)) {
        builder.setMessage("Bluetooth is not enabled.");
        builder.setPositiveButton("Enable", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                enable_BlueTooth();
                dialog.dismiss();
            }
        });
    } else if (action.equals(CONNECT)) {
        builder.setMessage("Please connect to device to continue.");
        builder.setPositiveButton("Connect", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                pair_to_device();
                dialog.dismiss();
            }
        });
    }
}

AlertDialog alertDialog = builder.create();
alertDialog.setCancelable(false);
alertDialog.show();

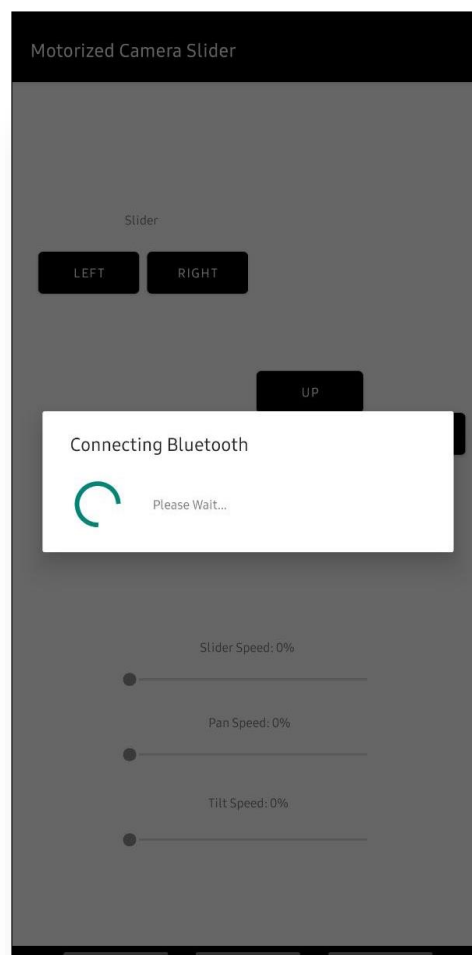
builder = null;
}
```



Εικόνα 3. 2 Παράθυρο διαλόγου για την ενεργοποίηση του Bluetooth



Εικόνα 3. 3 Παράθυρο διαλόγου για τη σύνδεση ασύρματης συσκευής



Εικόνα 3. 4 Ένδειξη διαδικασίας σύνδεσης ασύρματης συσκευής

Η παρακάτω μέθοδος καλείται μέσω της μεθόδου `show_dialog()` για να κλείσει ή να ακυρώσει οποιοδήποτε άλλο παράθυρο διαλόγου μπορεί να υπάρχει πάνω στην οθόνη. Στην ουσία είναι μια

χρήσιμη μέθοδος η οποία διασφαλίζει την ομαλή λειτουργία της εφαρμογής κατά την εμφάνιση διαλόγων.

```
private void close_any_dialogs() {
    if (alertDialog != null && alertDialog.isShowing()) {
        alertDialog.dismiss();
        alertDialog = null;
    }
}
```

Στο παρακάτω απόσπασμα του κώδικα αναλύεται μια μέθοδος η οποία καλείται για να ερωτηθεί ο χρήστης για τη χορήγηση άδειας χρήσης του Bluetooth. Αφού χορηγηθεί η άδεια η εφαρμογή προβαίνει στην ενεργοποίηση του Bluetooth

```
//Ask for permission to toggle Bluetooth on and off
private void requestBluetoothAdminPermission() {
    if (ContextCompat.checkSelfPermission(this, Manifest.permission.BLUETOOTH_ADMIN) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.BLUETOOTH_ADMIN},
REQUEST_BLUETOOTH_ADMIN_PERMISSION);
    }
}
```

Έλεγχος κατάστασης χορήγησης αδειών για τη χρήση του Bluetooth. Εάν χορηγήθηκαν κατά την ενέργεια του χρήστη στο User Interface, καλείται η μέθοδος ενεργοποίησης του Bluetooth. Σε περίπτωση που ο χρήστης απορρίψει τη διαδικασία χορήγησης άδειας, ερωτάται ξανά έως ότου να εγκρίνει τη χορήγηση της άδειας.

```

@Override
public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {
    super.onRequestPermissionsResult(requestCode, permissions, grantResults);
    if (requestCode == REQUEST_BLUETOOTH_ADMIN_PERMISSION) {
        if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
            // Permission granted
            enable_BlueTooth();
        } else {
            // Permission denied
            Toast.makeText(this, "Permission denied to enable Bluetooth",
                Toast.LENGTH_SHORT).show();
            startActivityForResult();
        }
    }
}

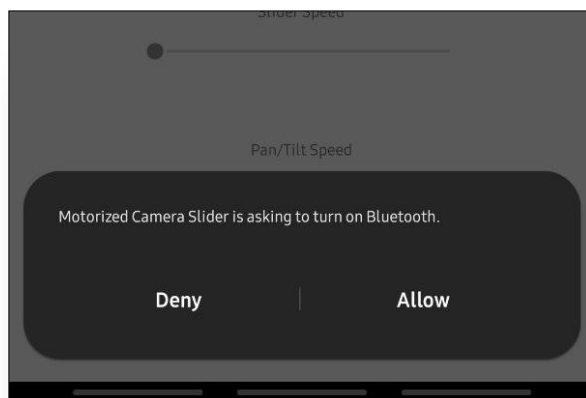
//To enable the user to grant the permission again
private void startActivityForResult() {
    Intent intent = new Intent();
    intent.setAction(Settings.ACTION_APPLICATION_DETAILS_SETTINGS);
    Uri uri = Uri.fromParts("package", getPackageName(), null);
    intent.setData(uri);
    startActivityForResult(intent, REQUEST_ENABLE_BT);
}

//In case the user denies the bluetooth to be enabled. Continue to persist.
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if (requestCode == REQUEST_ENABLE_BT) {
        BluetoothAdapter bluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
        if (bluetoothAdapter != null && bluetoothAdapter.isEnabled()) {
            // Bluetooth is enabled
        } else {
            // Bluetooth is not enabled
            show_dialog(ENABLE_BT);
        }
    }
}
}

```

Στην παρακάτω εικόνα παρουσιάζεται το συστημικό μήνυμα το οποίο ερωτά το χρήστη την έγκριση χορήγησης της άδειας χρήσης Bluetooth.



Εικόνα 3. 5 Παράθυρο συστήματος για τη χορήγηση αδειών χρήσης Bluetooth

Τέλος, η μέθοδος `outstream()` καλείται για τη μετάδοση δεδομένων. Ως παράμετρο λαμβάνει `string` το οποίο το μετατρέπει σε `Bytes`.

```
private void outstream(String str) {  
    byte[] bytes = str.getBytes(asciiCharset);  
    BluetoothConnection.write(bytes);  
}  
  
}
```

3.3 Υλοποίηση συστήματος Arduino και ο προγραμματισμός του

Παρακάτω αναλύεται και επεξηγείται ο κώδικας που συντάχθηκε για το μικροελεγκτή του Arduino. Αρχικά, χρειάστηκε να πραγματοποιηθεί η εισαγωγή τριών βασικών βιβλιοθηκών:

1. `SoftwareSerial.h`: Παρέχει συναρτήσεις για σειριακή επικοινωνία με το Bluetooth module όπως `Serial.begin()`, `Serial.read()` και `Serial.write()`.
2. `AccelStepper.h`: Παρέχει συναρτήσεις για τον έλεγχο βηματικών κινητήρων.
3. `ezButton.h`: Επιτρέπει τη διαχείριση και την ανάγνωση καταστάσεων διακοπών `microswitch`.

Παρακάτω, με τη διαδικασία `#define` δηλώνονται οι ακροδέκτες του Arduino στους οποίους αντιστοιχούν ακροδέκτες `STEP` και `DIR` του κάθε βηματικού κινητήρα. Επιπλέον, δηλώνεται ο αριθμός βημάτων ανά πλήρη περιστροφή βηματικού κινητήρα και η μέγιστη ταχύτητα που μπορεί να έχει.

```
#include <SoftwareSerial.h>  
#include <AccelStepper.h>  
#include <ezButton.h>  
  
#define STEPPER1_STEP_PIN 3  
#define STEPPER1_DIR_PIN 2  
  
#define STEPPER2_STEP_PIN 5  
#define STEPPER2_DIR_PIN 4  
  
#define STEPPER3_STEP_PIN 7  
#define STEPPER3_DIR_PIN 6  
  
#define SLIDER_LEFT_PIN 8  
#define SLIDER_RIGHT_PIN 9  
#define PAN_LEFT_PIN 10  
#define PAN_RIGHT_PIN 11  
  
#define stepsPerRev 4096  
  
#define MAX_SPEED 70
```

Εδώ, με τη βοήθεια της βιβλιοθήκης `ezButton.h` δηλώνονται οι ακροδέκτες του Arduino οι οποίοι θα διαβάζουν τις καταστάσεις των διακοπών `microswitch`.

```
ezButton slider_left_Switch(SLIDER_LEFT_PIN);
ezButton slider_right_Switch(SLIDER_RIGHT_PIN);

ezButton pan_left_Switch(PAN_LEFT_PIN);
ezButton pan_right_Switch(PAN_RIGHT_PIN);
```

Παρακάτω, με σε μια γραμμή δηλώνονται δυο ακροδέκτες του Arduino, ο ένας της αποστολής δεδομένων TX και ο άλλος της λήψης δεδομένων RX. Αυτοί είναι οι ακροδέκτες που θα χρησιμοποιηθούν για την επικοινωνία του Bluetooth module HC-05. Επιπλέον, όπως φαίνεται στο απόσπασμα κώδικα, δηλώνονται οι μεταβλητές που θα χρησιμοποιηθούν.

```
SoftwareSerial BTserial(0, 1);

// Initialize stepper motors
AccelStepper stepper1(1, STEPPER1_STEP_PIN, STEPPER1_DIR_PIN);
AccelStepper stepper2(1, STEPPER2_STEP_PIN, STEPPER2_DIR_PIN);
AccelStepper stepper3(1, STEPPER3_STEP_PIN, STEPPER3_DIR_PIN);

// Set default speed for steppers
int sliderSpeed = 0, sliderpos = 0;
int currentPosition = 0;
int panSpeed = 0;
int tiltSpeed = 0;

int slider_direction = 1;
int pan_direction = 1;

String input = "";

bool slider_left_enabled = false;
bool slider_right_enabled = false;
bool pan_left_enabled = true;
bool pan_right_enabled = true;
```

Η συνάρτηση setup() είναι μια από τις βασικές συναρτήσεις του Arduino και χρησιμοποιείται για την αρχικοποίηση των ρυθμίσεων πριν από την έναρξη του κύριου προγράμματος. Αυτή η συνάρτηση εκτελείται μόνο μία φορά, κατά την εκκίνηση του Arduino. Σε αυτή τη συνάρτηση δηλώνεται το Baud rate της σειριακής θύρας και παράμετροι αρχικοποίησης των βηματικών κινητήρων.

Το Baud rate που επιλέχθηκε είναι τα 9600 bits ανά δευτερόλεπτο και είναι κατάλληλο για την εφαρμογή αυτή εφόσον πραγματοποιείται η μετάδοση μικρών ποσοτήτων δεδομένων.

```

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

  slider_left_Switch.setDebounceTime(50);
  slider_right_Switch.setDebounceTime(50);
  pan_left_Switch.setDebounceTime(50);
  pan_right_Switch.setDebounceTime(50);

  stepper1.setMaxSpeed(4000);
  stepper2.setMaxSpeed(4000);
  stepper3.setMaxSpeed(4000);

  stepper1.setAcceleration(1000);
  stepper2.setAcceleration(1000);
  stepper3.setAcceleration(1000);

  stepper1.setCurrentPosition(0);
  stepper2.setCurrentPosition(0);
  stepper3.setCurrentPosition(0);

  stepper1.setSpeed(0);
  stepper2.setSpeed(0);
  stepper3.setSpeed(0);

  Serial.write("Setup.\n");
}

```

Η συνάρτηση loop() είναι η κύρια συνάρτηση του προγράμματος η οποία καλείται αμέσως μετά τη setup(). Σε αυτή τη συνάρτηση θα εκτελείται επανειλημμένα ο κώδικας οποίος είναι τοποθετημένος σε αυτή.

Για κάθε διακόπτη microswitch θα εκτελείται η συνάρτηση loop() όπου γίνεται η ανίχνευση της κατάστασης του εκάστοτε μικροδιακόπτη. Τέλος, θα εκτελείται η συνάρτηση run() της οποίας ο κώδικας θα αναλυθεί παρακάτω.

```

void loop() {
  slider_left_Switch.loop();
  slider_right_Switch.loop();

  pan_left_Switch.loop();
  pan_right_Switch.loop();

  run();
}

```

Η συνάρτηση run() περιέχει το κύριο κώδικα του προγράμματος της εφαρμογής αυτής και ξεκινά με την ανίχνευση δεδομένων στη σειριακή θύρα στην οποία έχει συνδεσμοποιηθεί το Bluetooth module. Κάθε χαρακτήρας ο οποίος ανιχνεύεται στην σειριακή θύρα αποθηκεύεται σε ένα string. Κατά το πέρας της ανάγνωσης το string απαλείφεται από ειδικούς χαρακτήρες (νέας γραμμής, πέρας κειμένου κλπ.) από τη συνάρτηση trim().

```

void run(){

  if (BTserial.available(>0) {
    String inputString = ""; // create an empty string to store the incoming data
    while (BTserial.available()) {
      char data = BTserial.read();

      //inputString += data;
      if (isPrintable(data)) {
        inputString += data;
      }
      delay(3);
    }

    inputString.trim();
    Serial.print(inputString); Serial.print("\n");
  }
}

```

Εδώ γίνεται έλεγχος του πρώτου χαρακτήρα του ληφθέντος string. Υπάρχουν τρεις περιπτώσεις το string να ξεκινά με μικρό γράμμα. Μπορεί να ξεκινά με 's', 'p' ή 't'. Εάν ξεκινά με ένα από αυτά τα τρία μικρά γράμματα, τότε ότι ακολουθεί θα είναι πάντοτε αριθμός από 0 έως 100 και δηλώνει τη ταχύτητα που πρέπει να τεθεί ο βηματικός κινητήρας που αντιστοιχεί. Εάν το string ξεκινά με 's' το πρόγραμμα καταλαβαίνει ότι αφορά το βηματικό κινητήρα του μετατοπιστή. Σε περίπτωση που ξεκινά με 'p' αφορά τον κινητήρα της περιστροφής και τέλος, εάν ξεκινά με 't' αφορά τον κινητήρα της κλίσης.

```

if(inputString.charAt(0) == 's')
  sliderSpeed = inputString.substring(1).toInt() * MAX_SPEED;
if(inputString.charAt(0) == 'p')
  panSpeed = inputString.substring(1).toInt() * MAX_SPEED;
if(inputString.charAt(0) == 't')
  tiltSpeed = inputString.substring(1).toInt() * MAX_SPEED;

```

Παρακάτω είναι οι υπόλοιπες περιπτώσεις δεδομένων που μπορούν να ληφθούν από τη σειριακή θύρα. Αφορούν τις εντολές της κατεύθυνσης του εκάστοτε βηματικού κινητήρα καθώς και τον τερματισμό της περιστροφής τους. Πάντοτε περιγράφονται με κεφαλαία γράμματα για να ξεχωρίζουν από τις προηγούμενες περιπτώσεις.

```

if (inputString == "SLIDERLEFT"){
    slider_direction = -1;
}
else if (inputString == "SLIDERRIGHT"){
    slider_direction = 1;
}
else if (inputString == "PANLEFT") {
    pan_direction = -1;

} else if (inputString == "PANRIGHT") {
    pan_direction = 1;

} else if (inputString == "TILTUP") {
    tiltSpeed = -1 * abs(tiltSpeed);

} else if (inputString == "TILTDOWN") {
    tiltSpeed = abs(tiltSpeed);

} else if (inputString == "STOP") {
    Serial.println("ALL MOTORS STOPPED.");
    sliderSpeed = 0;
    panSpeed = 0;
    tiltSpeed = 0;
}
}
if (slider_direction == -1)
    stepper1.setSpeed(sliderSpeed);
else
    stepper1.setSpeed(-sliderSpeed);

if (pan_direction == -1)
    stepper2.setSpeed(-panSpeed);
else
    stepper2.setSpeed(panSpeed);

stepper3.setSpeed(tiltSpeed);
stepper3.runSpeed();

```

Παρακάτω γίνεται ο έλεγχος της κατάστασης των μικροδιακοπών σε συνδυασμό με την κατεύθυνση του αντίστοιχου βηματικού κινητήρα. Εάν ο μικροδιακόπτης έχει πατηθεί γίνεται παύση λειτουργίας του βηματικού κινητήρα που του αντιστοιχεί.

```

if ( ( slider_left_Switch.getState() == LOW && slider_direction == -1) ||
(slider_right_Switch.getState() == LOW && slider_direction == 1) ){
    stepper1.stop();
}
else{
    stepper1.runSpeed();
}

if ( ( pan_left_Switch.getState() == LOW && pan_direction == -1) || (pan_right_Switch.getState() == LOW
&& pan_direction == 1) ){
    stepper2.stop();
}
else{
    stepper2.runSpeed();
}
}

```


4 ΚΕΦΑΛΑΙΟ 4^ο : ΕΠΙΛΟΓΟΣ

4.1 Εισαγωγή Κεφαλαίου 4

Αυτό το κεφάλαιο αποτελεί μια ολοκληρωμένη σύνοψη του έργου που πραγματοποιήθηκε στο πλαίσιο αυτής της διπλωματικής εργασίας. Επισημαίνονται τα προβλήματα που προέκυψαν κατά την υλοποίηση του συστήματος και παρουσιάζονται οι τρόποι με τους οποίους αντιμετωπίστηκαν αποτελεσματικά. Επιπλέον, παρέχονται πολύτιμες προτάσεις για τη μελλοντική εξέλιξη και βελτίωση του συστήματος, ανοίγοντας τον δρόμο για περαιτέρω έρευνα και ανάπτυξη.

4.2 Σύνοψη Εργασίας

Ο ολισθητήρας κάμερας, γνωστός και ως camera slider, αποτελεί ένα απαραίτητο εργαλείο για κινηματογραφιστές και φωτογράφους που αναζητούν απρόσκοπτη λήψη ομαλών πλάνων βίντεο. Αυτή η καινοτομία στον χώρο της εικόνας και του βίντεο διευκολύνει τη δημιουργία εντυπωσιακών κινηματογραφικών στιγμιότυπων με ευκρίνεια και σταθερότητα. Εκείνο που καθιστά αυτό το εργαλείο ακόμη πιο ενδιαφέρον είναι η δυνατότητα ασύρματου ελέγχου, όπως παρουσιάστηκε σε αυτήν την εφαρμογή. Ο ασύρματος έλεγχος αντιπροσωπεύει έναν τρόπο ανάπτυξης της δημιουργικότητας των κινηματογραφιστών και φωτογράφων και επιτρέπουμε στους χειριστές να επικεντρωθούν αποκλειστικά στην τέχνη της λήψης. Η τεχνολογία Bluetooth εισβάλλει με επιτυχία στον κινηματογραφικό κόσμο, προσφέροντας έναν πλήρως ασύρματο έλεγχο των συσκευών, απαλείφοντας τον διασυνδετισμό με περιττά καλώδια. Αυτή η εξέλιξη ανοίγει νέους δρόμους για την κινηματογραφική δημιουργία, με την απελευθέρωση των δημιουργών από τους περιορισμούς της σύνδεσης με καλώδια και την προσφορά αβίαστης δημιουργικότητας.

4.3 Προβλήματα και η Επίλυσή τους

Ένα από τα κυριότερα μειονεκτήματα της εφαρμογής αυτής αφορά το υλικό από το οποίο κατασκευάστηκε. Χρησιμοποιώντας ξύλο, συγκεκριμένα ξύλο τύπου MDF, η κατασκευή αποκτά επιπρόσθετο βάρος σε σύγκριση με το ενδεχόμενο να είχε κατασκευαστεί από πλαστικό, χρησιμοποιώντας τρισδιάστατο εκτύπωση. Επιπλέον, με την πάροδο του χρόνου και τη συχνή χρήση, το ξύλο μπορεί να χάσει την αρχική του ποιότητα και να προκαλεί θορύβους κατά τη χρήση της κατασκευής. Η επιλογή τρισδιάστατης εκτύπωσης με πλαστικό υλικό θα είχε αρκετά πλεονεκτήματα, καθώς θα προσέφερε μεγαλύτερη στιβαρότητα, ελαφρύτερο βάρος και αυξημένη αντοχή στον χρόνο. Αυτό θα βελτιώνε την ανθεκτικότητα και την αξιοπιστία της συσκευής σε μακροπρόθεσμη χρήση.

Επιπλέον, ένα άλλο προς εξέταση μειονέκτημα αφορά τους οδηγούς βηματικών κινητήρων τύπου A4988. Παρόλο που είναι αποδοτικοί από άποψη κόστους, δημιουργούν περισσότερο θόρυβο σε σύγκριση με άλλους οδηγούς διαφορετικών κατασκευαστών. Μια εναλλακτική λύση είναι οι οδηγοί TMC2209, οι οποίοι εκτός από την αθόρυβη λειτουργία, προσφέρουν βελτιωμένη ακρίβεια στον μικροβηματισμό και μικρότερα βήματα από αυτά που μπορούν να προσφέρουν οι A4988. Οι

οδηγοί TMC2209 αποτελούν μια βελτιωμένη επιλογή που μπορεί να αυξήσει την απόδοση και την αθόρυβη λειτουργία της κατασκευής σας.

4.4 Παρατηρήσεις – Συμπεράσματα

Κατά την ανάπτυξη του κώδικα, ο οποίος ελέγχει το Arduino, παρατηρήθηκε ότι οι εντολές οποίες εισέρχονται σε μορφή byte, φέρανε ένα επιπρόσθετο ειδικό χαρακτήρα σήμανσης τέλους string. Χρειάστηκε να αφαιρεθεί ο ειδικός χαρακτήρας επέμβαινε στην αποτελεσματική ανάγνωση των εισερχόμενων δεδομένων από το Bluetooth module, με την εντολή trim().

Επιπλέον, είναι σημαντικό να σημειωθεί πως το αρχικό πλάνο της κατασκευής δεν συμπεριλάμβανε τους μικροδιακόπτες οι οποίοι χρησιμεύουν για τη διακοπή κίνησης των βηματικών κινητήρων. Κατά τη διάρκεια της ολοκλήρωσης της κατασκευής, παρατηρήθηκε η σημαντικότητα ύπαρξης των μικροδιακοπών για τον ορισμό των ορίων περιστροφής – κίνησης των κινητών μερών της κατασκευής.

Ακόμη ένα πρόβλημα το οποίο παρατηρήθηκε κατά τη κατασκευή του συστήματος, ήταν η προσαρμογή του μιάντα στις τροχαλίες στην ένωση των κινητών μερών περιστροφής – κλίσης. Εξαιτίας του περιορισμένου χώρου σε αυτήν την περιοχή, η εγκατάσταση του μιάντα στο ιδανικό μήκος αποδείχθηκε δύσκολη. Για να αντιμετωπιστεί αυτό το ζήτημα, προστέθηκε μια επιπλέον τροχαλία σε ένα ενδιάμεσο σημείο ανάμεσα στις άλλες δύο τροχαλίες. Αυτή η επιπλέον τροχαλία ασκεί τάση στον μιάντα, επιλύοντας το πρόβλημα προσαρμογής.

4.5 Προτάσεις Μελλοντικής Εξέλιξης

Ο κώδικας που χρησιμοποιήθηκε για την ανάπτυξη της εφαρμογής αυτής αποτελεί έναν κώδικα απλού ελέγχου βηματικών κινητήρων. Με το πάτημα ενός πλήκτρου στην οθόνη μιας έξυπνης συσκευής ενεργοποιούνται οι κινητήρες και ελέγχεται η ταχύτητά τους.

Για μια προοπτική εξέλιξης της εφαρμογής, μπορεί να εξεταστεί η ανάπτυξη ενός πιο εξελιγμένου κώδικα, ο οποίος θα επιτρέπει την αυτόματη κίνηση των βηματικών κινητήρων με προγραμματισμένο τρόπο από τον χρήστη. Αυτό θα σήμαινε ότι με το πάτημα ενός απλού πλήκτρου, θα μπορούσαν να εκτελούνται πολύπλοκες προκαθορισμένες κινήσεις του ολισθητήρα κάμερας. Επιπλέον, θα μπορούσαν να δημιουργηθούν προαποθηκευμένα προγράμματα και έτοιμες κινήσεις για ευκολότερη χρήση.

Βιβλιογραφία – Αναφορές - Διαδικτυακές Πηγές

- [1] Figgis, M. (2007). *Digital Filmmaking*. <https://doi.org/10.1604/9780571226252>
- [2] Musburger, R. B., & Ogden, M. R. (2014). *Single-Camera Video Production*. Focal Press.
- [3] Brown, B. (2011). *Cinematography: Theory and Practice: Image Making for Cinematographers and Directors*.
- [4] Leeuw, B. D. (1997). *Digital Cinematography: Lighting and Photographing Computer Generated Animation*. <https://doi.org/10.1604/9780122088759>
- [5] Steadicam - Wikipedia. (2021, April 1). Retrieved from <https://en.wikipedia.org/wiki/Steadicam>
- [6] Ascher, S. (2019). *The Filmmaker's Handbook: A Comprehensive Guide for the Digital Age: Fifth Edition*. Plume.
- [7] Hart, D. (2012). *The Camera Assistant: A Complete Professional Handbook*. Routledge.
- [8] Bordwell, D., Thompson, K., & Smith, J. (2019). *ISE Film Art: an Introduction*.
- [9] Crane shot - Wikipedia. (2009, May 1). Retrieved from https://en.wikipedia.org/wiki/Crane_shot
- [10] Creative Uses of Dolly Shots: Camera Movements and Angles. (2019, February 18). Retrieved from <https://www.cinemagics.com/amp/the-dolly-shot-creative-uses-of-camera-movements-shots-angles-by-jason-hellerman>
- [11] Yeager, C. (2019, August 20). *The Dolly Shot: How It Works and Why It's Powerful*. Retrieved from <https://www.premiumbeat.com/blog/how-to-achieve-perfect-dolly-shot/>
- [12] Camera dolly - Wikipedia. (2021, December 1). Retrieved from https://en.wikipedia.org/wiki/Camera_dolly
- [13] Katz, S. (2011). *Film Directing: Shot by Shot - Visualizing from Concept to Screen*.
- [14] *Camera sliders for the perfect shot* – igus Blog. (n.d.). Retrieved from <https://blog.igus.eu/camera-sliders-for-the-perfect-shot/>
- [15] *What Is the Best Length for a Camera Slider?* - Your Photo Advisor. (2021, March 16). Retrieved from <https://yourphotoadvisor.com/what-is-the-best-length-for-a-camera-slider/>
- [16] Miller, B. A., & Bisdikian, C. (2000). *Bluetooth Revealed: The Insider's Guide to an Open Specification for Global Wireless Communications*. <https://doi.org/10.1604/9780130902948>
- [17] Bray, J., & Sturman, C. (2001). *Bluetooth: Connect Without Cables*. <https://doi.org/10.1604/9780130898401>
- [18] Bakker, D. M., & Gilster, D. M. (2002). *Bluetooth End to End*. <https://doi.org/10.1604/9780764523182>

- [19] Bluetooth - Wikipedia. (1998, May 7). Retrieved from <https://en.wikipedia.org/wiki/Bluetooth>
- [20] Tanenbaum, A., & Wetherall, D. (2010). *Computer Networks*.
- [21] Bluetooth Mesh | Bluetooth® Technology Website. (n.d.). Retrieved from <https://www.bluetooth.com/learn-about-bluetooth/feature-enhancements/mesh/>
- [22] Kranz, A. (2021, March 19). What Is Bluetooth Mesh? How Is Bluetooth Mesh Work? Retrieved from <https://www.mokoblue.com/what-is-bluetooth-mesh/>
- [23] Serial Port Bluetooth Module (Master/Slave) : HC-05 - ITEAD Wiki. (n.d.). Retrieved from [https://wiki.iteadstudio.com/Serial_Port_Bluetooth_Module_\(Master/Slave\)_:_HC-05](https://wiki.iteadstudio.com/Serial_Port_Bluetooth_Module_(Master/Slave)_:_HC-05)
- [24] Bluetooth Module HC-05 Pinout, AT Commands & Arduino Programming. (n.d.). Retrieved from <https://www.electronicwings.com/sensors-modules/bluetooth-module-hc-05->
- [25] ATMEGA328P. (n.d.). Retrieved from <https://www.microchip.com/en-us/product/atmega328p>
- [26] Nano | Arduino Documentation. (n.d.). Retrieved from <https://docs.arduino.cc/hardware/nano>
- [27] Arduino Nano - Wikipedia. (2020, November 25). Retrieved from https://en.wikipedia.org/wiki/Arduino_Nano
- [28] What is a stepper motor? - Principles, types and crontollers. (n.d.). Retrieved from https://www.omega.co.uk/prodinfo/stepper_motors.html
- [29] V.V. Athani. (1997). *Stepper Motors: Fundamentals, Applications and Design*. New Age.
- [30] Acarnely, P. (2002). *Stepping Motors: A Guide to Modern Theory and Practice* (Vol. No. 63). <https://doi.org/10.1604/9780852960295>
- [31] Unipolar vs bipolar stepper motors. (n.d.). Retrieved from <https://techexplorations.com/guides/arduino/motors/unipolar-vs-bipolar-stepper-motors>
- [32] E. (2019, January 14). *Advantage and Disadvantage of Stepper Motors | Slew Range | Slewing Mode*. Retrieved from <https://www.eeeguide.com/advantage-and-disadvantage-of-stepper-motors/>
- [33] NEMA Motor - RepRap. (n.d.). Retrieved from https://reprap.org/wiki/NEMA_Motor
- [34] Pololu - A4988 Stepper Motor Driver Carrier. (n.d.). Retrieved from <https://www.pololu.com/product/1182>
- [35] Motor Control. (n.d.). Retrieved from <https://www.trinamic.com/technology/motor-control-technology/>
- [36] Uicker, J., Pennock, G., & Shigley, J. (2010). *Theory of Machines and Mechanisms*.
- [37] Bluetooth overview | Android Developers. (n.d.). Retrieved from <https://developer.android.com/guide/topics/connectivity/bluetooth>
- [38] Belinski, E. (n.d.). Android API Levels. Retrieved from <https://apilevels.com/>

Παράρτημα Α

Παρακάτω αναλύεται ο κώδικας του Manifest της εφαρμογής Android όπου δηλώνονται βασικά στοιχεία που καθορίζουν τη λειτουργία καθώς και την εμφάνιση της εφαρμογής. Μερικά από κύρια στοιχεία είναι η ονομασία της εφαρμογής, οι άδειες τις οποίες απαιτεί για τη λειτουργία του και το ελάχιστο επίπεδο API το οποίο απαιτείται για να λειτουργήσει η εφαρμογή, όπως φαίνεται παρακάτω στον κώδικα είναι το 31 το οποίο αντιστοιχεί στην έκδοση 12 του Android [\[38\]](#).

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <uses-permission android:name="android.permission.BLUETOOTH"
        android:maxSdkVersion="30" />
    <uses-permission android:name="android.permission.BLUETOOTH_ADMIN"
        android:maxSdkVersion="30" />
    <uses-permission android:name="android.permission.BLUETOOTH_ADVERTISE" />
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.BLUETOOTH_CONNECT" />
    <uses-feature android:name="android.hardware.bluetooth" android:required="true"/>
    <uses-permission android:name="android.permission.BLUETOOTH_SCAN" />
    <uses-permission android:name="android.permission.BLUETOOTH_PRIVILEGED" />
    <uses-permission android:name="android.permission.ACCESS_SURFACE_FLINGER" />

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="Motorized Camera Slider"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme"
        tools:targetApi="31">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Παράρτημα Β

Παρακάτω αναλύεται ο κώδικας που χρησιμοποιήθηκε για τη δημιουργία της τάξης BluetoothConnectionService η οποία αποτελεί θεμέλιο της εφαρμογής. Η τάξη βασίστηκε στον κώδικα που παρέχει η ενότητα Developer της επίσημης ιστοσελίδας του Android. Παρέχει χρήσιμες μεθόδους για τη συνδεσιμότητα και επικοινωνία με άλλες συσκευές Bluetooth. [\[37\]](#)

```
package com.example.cammotorizedslider;

import android.app.ProgressDialog;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;
import android.bluetooth.BluetoothServerSocket;
import android.bluetooth.BluetoothSocket;
import android.content.Context;
import android.util.Log;

import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.nio.charset.Charset;
import java.nio.charset.StandardCharsets;
import java.util.UUID;

public class BluetoothConnectionService {
    private static final String TAG = "BluetoothConnectionServ";

    private static final String appName = "cammotorizedslider";

    private static final UUID MY_UUID_INSECURE =
        UUID.fromString("00001101-0000-1000-8000-00805F9B34FB");

    private final BluetoothAdapter mBluetoothAdapter;
    Context mContext;

    private AcceptThread mInsecureAcceptThread;

    private ConnectThread mConnectThread;
    private BluetoothDevice mmDevice;
    private UUID deviceUUID;
    ProgressDialog mProgressDialog;
    Charset asciiCharset = StandardCharsets.UTF_8;

    private ConnectedThread mConnectedThread;

    public BluetoothConnectionService(Context context) {
        mContext = context;
        mBluetoothAdapter = BluetoothAdapter.getDefaultAdapter();
        start();
    }
}
```

```

/**
 * This thread runs while listening for incoming connections. It behaves
 * like a server-side client. It runs until a connection is accepted
 * (or until cancelled).
 */
private class AcceptThread extends Thread {

    // The local server socket
    private final BluetoothServerSocket mmServerSocket;

    public AcceptThread(){
        BluetoothServerSocket tmp = null;

        // Create a new listening server socket
        try{
            tmp = mBluetoothAdapter.listenUsingInsecureRfcommWithServiceRecord(appName,
MY_UUID_INSECURE);

            Log.d(TAG, "AcceptThread: Setting up Server using: " + MY_UUID_INSECURE);
        }catch (IOException e){
            Log.e(TAG, "AcceptThread: IOException: " + e.getMessage() );
        }

        mmServerSocket = tmp;
    }

    public void run(){
        Log.d(TAG, "run: AcceptThread Running.");

        BluetoothSocket socket = null;

        try{
            // This is a blocking call and will only return on a
            // successful connection or an exception
            Log.d(TAG, "run: RFCOM server socket start.....");

            socket = mmServerSocket.accept();

            Log.d(TAG, "run: RFCOM server socket accepted connection.");

        }catch (IOException e){
            Log.e(TAG, "AcceptThread: IOException: " + e.getMessage() );
        }

        //talk about this is in the 3rd
        if(socket != null){
            connected(socket,mmDevice);
        }

        Log.i(TAG, "END mAcceptThread ");
    }

    public void cancel() {
        Log.d(TAG, "cancel: Canceling AcceptThread.");
        try {
            mmServerSocket.close();
        } catch (IOException e) {
            Log.e(TAG, "cancel: Close of AcceptThread ServerSocket failed. " + e.getMessage() );
        }
    }
}
}

```

```

/**
 * This thread runs while attempting to make an outgoing connection
 * with a device. It runs straight through; the connection either
 * succeeds or fails.
 */
private class ConnectThread extends Thread {
    private BluetoothSocket mmSocket;

    public ConnectThread(BluetoothDevice device, UUID uuid) {
        Log.d(TAG, "ConnectThread: started.");
        mmDevice = device;
        deviceUUID = uuid;
    }

    public void run(){
        BluetoothSocket tmp = null;
        Log.i(TAG, "RUN mConnectThread ");

        // Get a BluetoothSocket for a connection with the
        // given BluetoothDevice
        try {
            Log.d(TAG, "ConnectThread: Trying to create InsecureRfcommSocket using UUID: "
                +MY_UUID_INSECURE );
            tmp = mmDevice.createRfcommSocketToServiceRecord(deviceUUID);
        } catch (IOException e) {
            Log.e(TAG, "ConnectThread: Could not create InsecureRfcommSocket " + e.getMessage());
        }

        mmSocket = tmp;

        // Always cancel discovery because it will slow down a connection
        mBluetoothAdapter.cancelDiscovery();

        // Make a connection to the BluetoothSocket
        try {
            // This is a blocking call and will only return on a
            // successful connection or an exception
            mmSocket.connect();

            Log.d(TAG, "run: ConnectThread connected.");
        } catch (IOException e) {
            // Close the socket
            try {
                mmSocket.close();
                Log.d(TAG, "run: Closed Socket.");
            } catch (IOException e1) {
                Log.e(TAG, "mConnectThread: run: Unable to close connection in socket " +
                    e1.getMessage());
            }
            Log.d(TAG, "run: ConnectThread: Could not connect to UUID: " + MY_UUID_INSECURE );
        }

        //will talk about this in the 3rd video
        connected(mmSocket,mmDevice);
    }

    public void cancel() {
        try {
            Log.d(TAG, "cancel: Closing Client Socket.");
            mmSocket.close();
        } catch (IOException e) {
            Log.e(TAG, "cancel: close() of mmSocket in Connectthread failed. " + e.getMessage());
        }
    }
}

```



```

// Keep listening to the InputStream until an exception occurs
while (true) {
    // Read from the InputStream
    try {
        bytes = mmInStream.read(buffer);
        String incomingMessage = new String(buffer, 0, bytes);
        Log.d(TAG, "InputStream: " + incomingMessage);
    } catch (IOException e) {
        Log.e(TAG, "write: Error reading Input Stream. " + e.getMessage() );
        break;
    }
}

//Call this from the main activity to send data to the remote device
public void write(byte[] bytes) {
    String text = new String(bytes, asciiCharset);
    Log.d(TAG, "write: Writing to outputstream: " + text);
    try {
        mmOutputStream.write(bytes);
    } catch (IOException e) {
        Log.e(TAG, "write: Error writing to output stream. " + e.getMessage() );
    }
}

/* Call this from the main activity to shutdown the connection */
public void cancel() {
    try {
        mmSocket.close();
    } catch (IOException e) { }
}

private void connected(BluetoothSocket mmSocket, BluetoothDevice mmDevice) {
    Log.d(TAG, "connected: Starting.");

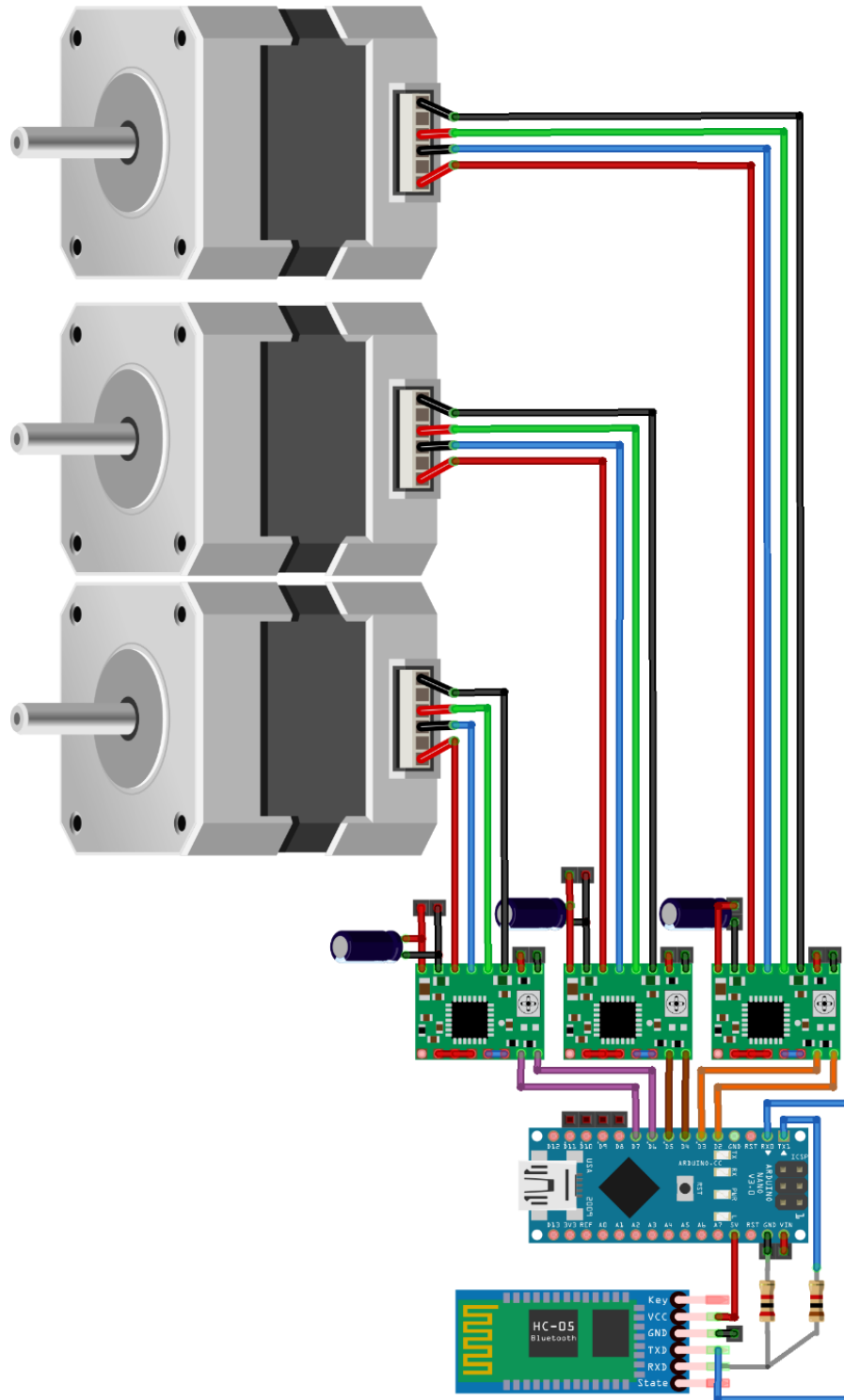
    // Start the thread to manage the connection and perform transmissions
    mConnectedThread = new ConnectedThread(mmSocket);
    mConnectedThread.start();
}

public void write(byte[] out) {
    mConnectedThread.write(out);
}
}

```

Παράρτημα Γ

Παρακάτω παρουσιάζεται η συνδεσμολογία του κυκλώματος σε λογισμικό περιβάλλον Fritzing, πλην των μικροδιακοπών, οι οποίοι έχουν τοποθετηθεί εξωτερικά της πλακέτας του κυκλώματος.



Εικόνα Παραρτήματος Γ 1 Συνδεσμολογία κυκλώματος σε περιβάλλον λογισμικού Fritzing