



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ**  
**ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ**  
**ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ**

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

**Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών**

**Γεώργιος Δ. Μπελαλής**  
**Ιωάννης Δ. Μπελαλής**

**Εισηγητής: Δρ Ιωάννης Έλληνας, Καθηγητής**

**ΑΘΗΝΑ**  
**ΝΟΕΜΒΡΙΟΣ 2019**

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ**

**Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών**

**Γεώργιος Δ. Μπελαλής**

**A.M. 35672**

**Ιωάννης Δ. Μπελαλής**

**A.M. 36696**

**Εισηγητής: Δρ Ιωάννης Έλληνας, Καθηγητής**

**Εξεταστική Επιτροπή: Αμοργίνος Ιωάννης  
Βελώνη Αναστασία**

**Ημερομηνία εξέτασης: 11/2019**

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

## ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Οι κάτωθι υπογεγραμμένοι Γεώργιος Μπελαλής του Δημητρίου, με αριθμό μητρώου 35672 και Ιωάννης Μπελαλής του Δημητρίου, με αριθμό μητρώου 36696, φοιτητές του Τμήματος Μηχανικών Η/Υ Συστημάτων Τ.Ε. του Α.Ε.Ι. Πειραιά Τ.Τ. πριν αναλάβουμε την εκπόνηση της Πτυχιακής Εργασίας μας, δηλώνουμε ότι ενημερωθήκαμε για τα παρακάτω:

«Η Πτυχιακή Εργασία (Π.Ε.) αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο του συγγραφέα, όσο και του Ιδρύματος και θα πρέπει να έχει μοναδικό χαρακτήρα και πρωτότυπο περιεχόμενο.

Απαγορεύεται αυστηρά οποιοδήποτε κομμάτι κειμένου της να εμφανίζεται αυτούσιο ή μεταφρασμένο από κάποια άλλη δημοσιευμένη πηγή. Κάθε τέτοια πράξη αποτελεί προϊόν λογοκλοπής και εγείρει θέμα Ηθικής Τάξης για τα πνευματικά δικαιώματα του άλλου συγγραφέα. Αποκλειστικός υπεύθυνος είναι ο συγγραφέας της Π.Ε., ο οποίος φέρει και την ευθύνη των συνεπειών, ποινικών και άλλων, αυτής της πράξης.

Πέραν των όποιων ποινικών ευθυνών του συγγραφέα σε περίπτωση που το Ίδρυμα του έχει απονείμει Πτυχίο, αυτό ανακαλείται με απόφαση της Συνέλευσης του Τμήματος. Η Συνέλευση του Τμήματος με νέα απόφασής της, μετά από αίτηση του ενδιαφερόμενου, του αναθέτει εκ νέου την εκπόνηση της Π.Ε. με άλλο θέμα και διαφορετικό επιβλέποντα καθηγητή. Η εκπόνηση της εν λόγω Π.Ε. πρέπει να ολοκληρωθεί εντός τουλάχιστον ενός ημερολογιακού 6μήνου από την ημερομηνία ανάθεσής της. Κατά τα λοιπά εφαρμόζονται τα προβλεπόμενα στο άρθρο 18, παρ. 5 του ισχύοντος Εσωτερικού Κανονισμού.»

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

## **ΕΥΧΑΡΙΣΤΙΕΣ**

Η παρούσα πτυχιακή εργασία ολοκληρώθηκε ύστερα από έρευνα και ιδιαίτερη προσπάθεια σε ένα ενδιαφέρον γνωστικό αντικείμενο. Την προσπάθειά μας αυτή υποστήριξε ο επιβλέπων καθηγητής, κ. Ιωάννης Έλληνας τον οποίο θα θέλαμε να ευχαριστήσουμε για την εμπιστοσύνη που μας έδειξε, καθώς και την καθοδήγηση μέσα του μέσα από τις σωστές τοποθετήσεις και κατευθύνσεις καθ' όλη τη διάρκεια της εκπόνησης της.

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών



## ΠΕΡΙΛΗΨΗ

Στη παρούσα εργασία κατασκευάστηκε ένα ψηφιακό γωνιόμετρο, για τη μέτρηση του εύρους κίνησης των αρθρώσεων, ασθενών με κινητικά προβλήματα στο στάδιο της αποκατάστασης ή/και αθλητών. Για την κατασκευή του οργάνου, αξιοποιήθηκαν συστήματα ανοιχτού κώδικα και συγκεκριμένα, η πλατφόρμα Arduino Uno. Η πλατφόρμα προγραμματίστηκε, ώστε να λαμβάνει τις μετρήσεις από το όργανο, να τις αποστέλλει και αποθηκεύει σε έναν Server. Προκειμένου να γίνει η λήψη της σχετικής θέσης των αξόνων του γωνιόμετρου, χρησιμοποιήθηκε πλακέτα οπτικού ποντικιού με μικροεπεξεργαστή ADNS-2083. Μέσω του Arduino Uno, η σχετική θέση μετατρέπεται σε μοίρες και αποστέλλεται στον Server, μέσω WiFi. Για τη σύνδεση στο WiFi χρησιμοποιείται το WiFi Module ESP8266. Με τον τρόπο αυτό τα δεδομένα αποθηκεύονται για διάφορες χρονικές περιόδους ώστε να υπάρχει πρόσβαση σε αυτά από τους χρήστες.

## ABSTRACT

The present thesis concerns the manufacturing of a digital goniometer, for the precise measurement of the range of motion around a joint of an athlete or a patient in rehabilitative therapy. For the manufacturing to be achieved, open-source platforms were used and more specifically, Arduino Uno. This microcontroller board has been programmed to receive measurements from a sensor and then transfer them to a server. The sensor consists of an optical mouse's board, which contains the ADNS-2083 microprocessor. Arduino Uno transforms the relative position to degrees and then transmits the data over WiFi. The WiFi connection is being handled by the ESP-8266 module. Finally, the data are being kept in the server, for users to have access to them.

Επιστημονική περιοχή: Σχεδίαση και ανάπτυξη ιατρικού οργάνου

Λέξεις κλειδιά: Arduino Uno, πλακέτα οπτικού ποντικιού ADNS-2083, WiFi Module ESP8266, ψηφιακό γωνιόμετρο, εύρος κίνησης

Keywords: Arduino Uno, optical mouse ADNS-2083 sensor, WiFi Module ESP8266, digital goniometer, range of motion

Περιεχόμενα

<b>ΕΙΣΑΓΩΓΗ</b> .....	<b>11</b>
<b>1. ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΑΝΑΣΚΟΠΗΣΗ</b> .....	<b>13</b>
<b>1.1 Η πλατφόρμα Arduino</b> .....	<b>13</b>
<b>1.2 Arduino Uno</b> .....	<b>16</b>
1.2.1 Τροφοδοσία.....	17
1.2.2 Μνήμη.....	17
1.2.3 Ακροδέκτες .....	18
<b>1.3 Πρότυπα ασύρματης επικοινωνίας</b> .....	<b>20</b>
1.3.1 ZigBee .....	21
1.3.2 IEEE 802.11: WiFi .....	21
1.3.3 Bluetooth και BLE.....	22
<b>1.4 Optical mouse</b> .....	<b>23</b>
<b>2. ΣΥΣΤΗΜΑΤΑ ΑΝΟΙΧΤΟΥ ΚΩΔΙΚΑ ΣΕ ΙΑΤΡΙΚΕΣ ΕΦΑΡΜΟΓΕΣ</b> .....	<b>25</b>
<b>3. ΜΕΘΟΔΟΛΟΓΙΑ – ΕΦΑΡΜΟΓΗ</b> .....	<b>31</b>
3.1 Κατασκευή γωνιόμετρου.....	31
3.2 Πλακέτα οπτικού ποντικιού .....	33
3.3 Πλακέτα WiFi – ESP8266 .....	36
3.4 Σύνδεση optical mouse (ADNS-2083) με Arduino .....	38
3.5 Σύνδεση Arduino με WiFi module .....	43
3.6 Προγραμματισμός του Arduino .....	45
3.7 Δημιουργία σελίδας στον Server .....	46
<b>4. ΣΥΜΠΕΡΑΣΜΑΤΑ</b> .....	<b>49</b>
<b>ΒΙΒΛΙΟΓΡΑΦΙΑ</b> .....	<b>69</b>

## ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Εικόνα 1.1: Arduino Duemilanove .....	14
Εικόνα 1.2: Arduino Uno R3 .....	16
Εικόνα 1.3: Optical mouse.....	23
Εικόνα 2.1: Shield connected with Arduino .....	27
Εικόνα 2.2: Αισθητήρας σε σύνδεση με το shield και το Arduino.....	27
Εικόνα 2.3: Arduino και αισθητήρας παλμών .....	28
Εικόνα 2.4: Αξονικός τομογράφος Jansen .....	29
Εικόνα 3.1: Γωνιόμετρο .....	31
Εικόνα 3.2: Κίνηση κινητού βραχίονα .....	32
Εικόνα 3.3: Χρήση γωνιόμετρου.....	33
Εικόνα 3.4: Τρόπος λειτουργίας οπτικού ποντικιού.....	34
Εικόνα 3.5: Πλακέτα optical mouse προσαρμοσμένη στην κατασκευή .....	35
Εικόνα 3.6: Ο φακός του οπτικού ποντικιού .....	35
Εικόνα 3.7: Οι ακροδέκτες του ADNS-2083 .....	36
Εικόνα 3.8: ESP-8266 .....	36
Εικόνα 3.9: Το μέγεθος του ESP-8266 .....	37
Εικόνα 3.10: Οι ακροδέκτες του ESP-8266 .....	38
Εικόνα 3.11: Σχέδιο συνδέσεων μεταξύ ADNS-2083 και Arduino .....	39
Εικόνα 3.12: Διάγραμμα περιγραφής διαδικασίας εγγραφής δεδομένων στον ADNS-2083 .....	40
Εικόνα 3.13: Διάγραμμα περιγραφής διαδικασίας ανάγνωσης δεδομένων από τον ADNS-2083 .....	40
Εικόνα 3.14: Διευθύνσεις των εγγραφών του ADNS-2083 .....	41
Εικόνα 3.15: Συνδέσεις/ κολλήσεις ADNS-2083.....	42
Εικόνα 3.16: Συνδέσεις τροφοδοσίας των περιφερειακών του Arduino .....	42
Εικόνα 3.17: Σειριακές συνδέσεις ανταλλαγής δεδομένων του Arduino .....	43
Εικόνα 3.18: Μπλόκ διάγραμμα του κυκλώματος της κατασκευής .....	44
Εικόνα 3.19: Σύγκριση χειροκίνητων μετρήσεων και απεικόνιση τους στο γράφημα στη θέση 0°.....	47
Εικόνα 3.20: Σύγκριση χειροκίνητων μετρήσεων και απεικόνιση τους στο γράφημα στη θέση 90°.....	47
Εικόνα 3.21: Σύγκριση χειροκίνητων μετρήσεων και απεικόνιση τους στο γράφημα στη θέση 190°.....	48
Εικόνα 3.22: Σύγκριση χειροκίνητων μετρήσεων και απεικόνιση τους στο γράφημα στη θέση 150°.....	48
Εικόνα 3.23: Σύγκριση χειροκίνητων μετρήσεων και απεικόνιση τους στο γράφημα στη θέση 50°.....	48

## ΕΙΣΑΓΩΓΗ

Ως κινητικότητα ορίζεται το εύρος κίνησης μιας άρθρωσης ή μιας σειράς αρθρώσεων. Το εύρος κίνησης μιας άρθρωσης εξαρτάται από τη δυνατότητα επιμήκυνσης των μυϊκών και των συνδετικών ιστών που περιβάλλουν την άρθρωση, διατηρώντας την αρχική τους δομή. Ως στατική κινητικότητα ορίζεται το συνολικό εύρος της κίνησης μιας άρθρωσης το οποίο μπορεί να επιτευχθεί και ακολούθως να διατηρηθεί για κάποιο χρονικό διάστημα, χωρίς παράλληλα να υπάρχει κίνηση στην άρθρωση. Το εύρος της κίνησης μπορεί να επιτευχθεί είτε ενεργητικά, όταν ο ίδιος ο ασκούμενος κινεί την άρθρωση ή τμήμα του σώματός του, ή παθητικά, όταν το τμήμα του σώματος κινείται με εξωτερική παρέμβαση.

Για τη μέτρηση του εύρους κίνησης μιας άρθρωσης χρησιμοποιείται ποικιλία οργάνων. Τα όργανα αυτά μπορεί να είναι από απλές χαράξεις στο χαρτί ή η ταινία μέτρησης έως τα ηλεκτρογωνιόμετρα και τα συστήματα ανάλυσης της κίνησης. Ο εξεταστής μπορεί να επιλέξει να χρησιμοποιήσει ένα συγκεκριμένο όργανο με βάση τον σκοπό της μέτρησης (κλινικό ή ερευνητικό), την κίνηση που αξιολογείται, την ακρίβεια του οργάνου, το μέγεθος, τη διαθεσιμότητα, το κόστος ή την ευκολία χρήσης του.

Το κοινό γωνιόμετρο αποτελείται από ένα μοιρογνωμόνιο 180ο ή 360ο, με άξονα ο οποίος ενώνει δύο βραχίονες. Ο ένας βραχίονας είναι σταθερός και προσαρμοσμένος στη θέση «μηδέν» και ο άλλος μπορεί να κινείται γύρω από τον άξονα του μοιρογνωμόνιου. Το μέγεθος του γωνιόμετρου που πρόκειται να χρησιμοποιηθεί ορίζεται από το μέγεθος της άρθρωσης που αξιολογείται. Μεγαλύτερα γωνιόμετρα χρησιμοποιούνται συνήθως για τη μέτρηση του εύρους κίνησης σε μεγάλες αρθρώσεις. Ο βαθμός αξιοπιστίας του γωνιόμετρου ποικίλει ανάλογα με την άρθρωση και την κίνηση που αξιολογείται. Ωστόσο, έχει βρεθεί ότι ο βαθμός αυτός κυμαίνεται από «καλός» ως «εξαιρετικός». Επιπλέον, έχει βρεθεί ότι η αξιοπιστία του οργάνου είναι υψηλότερη όταν οι μετρήσεις γίνονται από τον ίδιο εξεταστή συγκριτικά με τις μετρήσεις που γίνονται από διαφορετικό εξεταστή. Συμπερασματικά, το εύρος κίνησης μιας άρθρωσης μπορεί να μετρηθεί αξιόπιστα με τη χρήση ενός τυπικού γωνιόμετρου, αρκεί

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

οι επαναλαμβανόμενες μετρήσεις να εκτελούνται από τον ίδιο εξεταστή, χρησιμοποιώντας ένα «σταθερό, τυποποιημένο πρωτόκολλο μέτρησης»

Στόχος της παρούσας πτυχιακής εργασίας είναι η κατασκευή ενός ψηφιακού γωνιόμετρου, το οποίο μπορεί να χρησιμοποιηθεί τόσο από ασθενείς σε στάδιο αποκατάστασης κινητικών προβλημάτων προκειμένου να μπορούν να ελέγχουν την πρόοδο τους, όσο και από επαγγελματίες. Η κατασκευή αποτελείται από δύο βασικά μέρη: τα μηχανικά μέρη, από τα οποία κατασκευάστηκε ουσιαστικά το γωνιόμετρο και τα ηλεκτρονικά μέρη, τα οποία συνδέθηκαν με το όργανο προκειμένου, αφενός, να μετρηθεί η παραγόμενη πληροφορία (εύρος κίνησης), αφετέρου, να μπορεί να πραγματοποιηθεί σύνδεση με το διαδίκτυο για τη μεταφορά των δεδομένων που προέρχονται από αυτό. Στο πρώτο μέρος της εργασίας παρουσιάζονται οι βασικές έννοιες και οι τεχνολογίες που αξιοποιήθηκαν για την κατασκευή του οργάνου και αποσαφηνίζονται τα πλεονεκτήματα και τα μειονεκτήματά τους, μέσα από τη διεθνή βιβλιογραφία. Στη συνέχεια, περιγράφεται η μεθοδολογία που αναπτύχθηκε έτσι ώστε οι εν λόγω τεχνολογίες να αξιοποιηθούν για την κατασκευή του αντικείμενου της εργασίας και αναλύεται η εφαρμογή τους. Τέλος, παρατίθενται τα βασικότερα θετικά και αρνητικά σημεία της εν λόγω προσέγγισης και διατυπώνονται οι προοπτικές-προτάσεις για περαιτέρω έρευνα και βελτίωση του προτεινόμενου οργάνου.

## 1. ΒΙΒΛΙΟΓΡΑΦΙΚΗ ΑΝΑΣΚΟΠΗΣΗ

Στο σημείο αυτό παρουσιάζονται και αποσαφηνίζονται οι βασικές τεχνολογίες που επιλέχθηκαν να χρησιμοποιηθούν για την κατασκευή του γωνιόμετρου. Πιο συγκεκριμένα αναλύονται τα χαρακτηριστικά της υπολογιστικής πλατφόρμας Arduino Uno, τα ασύρματα δίκτυα επικοινωνίας, αλλά και η πλακέτα του οπτικού ποντικιού και ο τρόπος λειτουργίας τους, που θα συμβάλλουν στην κατανόηση του τρόπου χρήσης για την κατασκευή του οργάνου.

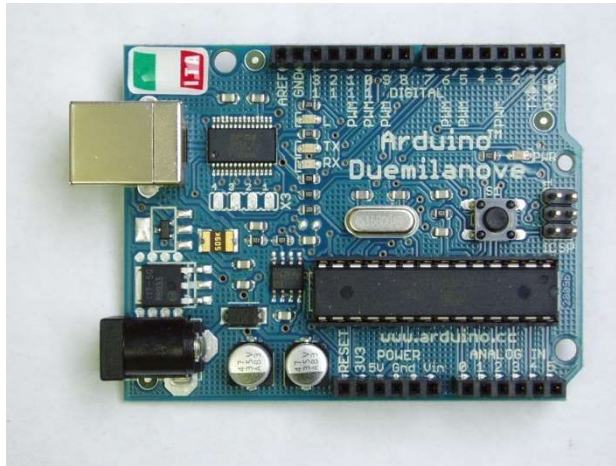
### 1.1 Η πλατφόρμα Arduino

Το Arduino είναι μία ανοιχτού κώδικα υπολογιστική πλατφόρμα προτυποποίησης ηλεκτρονικών βασισμένη σε μία απλή μητρική πλακέτα με ενσωματωμένο μικροελεγκτή προγραμματιζόμενη σε απλουστευμένη C/C++. Χρησιμοποιείται για την ανάπτυξη διαδραστικών αντικειμένων, ενώ έχει δυνατότητα σύνδεσης με υπολογιστή. Οι περισσότερες εκδόσεις του υπάρχουν προσυναρμολογημένες, ωστόσο, υπάρχουν ελεύθερα διαθέσιμα τόσο τα υλικά όσο και το διάγραμμα για τον καθένα που επιθυμεί να συναρμολογήσει ο ίδιος τη δική του πλακέτα. Τα ηλεκτρονικά σχέδια του Arduino διανέμονται υπό την Creative Commons Attribution Share- Alike 2.5 άδεια και είναι διαθέσιμα στην ιστοσελίδα του Arduino. Αφού κατασκευαστεί μπορεί να συμπεριφερθεί σαν ένας μικροσκοπικός υπολογιστής, αφού ο χρήστης μπορεί να συνδέσει επάνω του πολλαπλές μονάδες εισόδου/ εξόδου και να προγραμματίσει τον μικροελεγκτή να δέχεται δεδομένα από τις μονάδες εισόδου, να τα επεξεργάζεται και να στέλνει κατάλληλες εντολές στις μονάδες εξόδου.

Η ιστορία του Arduino ξεκινά το 2005 από τους ιδρυτές Massimo Banzi και David Cueartielles στην προσπάθεια να φτιαχτεί μία συσκευή για τον έλεγχο προγραμμάτων διαδραστικών σχεδίων από μαθητές, η οποία θα ήταν πιο φθηνή από άλλα πρωτότυπα συστήματα διαθέσιμα εκείνη την περίοδο. Η κατασκευή της συσκευής ξεκίνησε με το όνομα Arduin στην Ιταλία και η υλοποίηση της πλατφόρμας βασίστηκε στην πλατφόρμα Wiring και είχε προγραμματιστεί χρησιμοποιώντας την ομώνυμη γλώσσα προγραμματισμού. Έκτοτε ξεκίνησε μία σειρά εκδόσεων και το 2006 έγινε το λανσάρισμα του Arduino Mini, ενώ το 2008 του Arduino Duemilanove. Το 2009 ήταν η

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

χρονιά του Arduino Mega και ακολούθησαν αρκετές ακόμα φτάνοντας σήμερα τις δεκαέξι επίσημες πλατφόρμες, οι οποίες παρουσιάζονται στον πίνακα.



Εικόνα 1.1: Arduino Duemilanove

Στην ουσία πρόκειται για ένα ηλεκτρονικό κύκλωμα που βασίζεται στον μικροελεγκτή ATmega της Atmel. Το Arduino διαθέτει σειριακό interface. Ο μικροελεγκτής ATmega υποστηρίζει σειριακή επικοινωνία, την οποία το Arduino προωθεί μέσα από έναν ελεγκτή Serial-over-USB ώστε να συνδέεται με τον υπολογιστή μέσω usb. Η σύνδεση αυτή χρησιμοποιείται για την μεταφορά προγραμμάτων που σχεδιάζονται από τον υπολογιστή στο Arduino αλλά και για αμφίδρομη επικοινωνία του Arduino με τον υπολογιστή μέσα από το πρόγραμμα την ώρα που εκτελείται.

Οι επίσημες πλακέτες Arduino

Πλακέτα	Στοιχεία
Serial Arduino	Προγραμματισμένο με ATmega8 & σειριακή σύνδεση
Arduino Extreme	Προγραμματισμένο με ATmega8 & USB interface
Arduino Mini	Προγραμματισμένο με surface mounted ATmega168
Arduino Nano	Προγραμματισμένο με surface mounted ATmega168
LilyPad Arduino	Προγραμματισμένο με surface mounted ATmega328
Arduino NG	Προγραμματισμένο με ATmega8 & USB interface
Arduino NG Plus	Προγραμματισμένο με surface mounted ATmega168

Arduino Bluetooth	Προγραμματισμένο με surface mounted ATmega168
Arduino Diecimilla	Προγραμματισμένο με ATmega168 & USB interface
Arduino Duemillanuoove	Προγραμματισμένο με ATmega168 & USB interface
Arduino Mega	Προγραμματισμένο με surface mounted ATmega1280
Arduino Uno	Προγραμματισμένο με ATmega168 & USB interface
Arduino Mega 2560	Προγραμματισμένο με surface mounted ATmega2560
Arduino Leonardo	Προγραμματισμένο με ATmega32U4 chip
Arduino Esplora	Προγραμματισμένο με ATmega168 & USB interface
Arduino Due	Προγραμματισμένο με ATmega168 & USB interface

Το Arduino μπορεί να χρησιμοποιηθεί για να αναπτυχθούν διαδραστικές εφαρμογές, χρησιμοποιώντας το πλήθος των επαφών του είτε ως εισόδους είτε ως εξόδους ώστε να επιτευχθεί η σύνδεσή του με οποιαδήποτε ηλεκτρονική/ ηλεκτρική συσκευή. Τα projects (οι κατασκευές) του μπορούν να είναι αυτόνομα ή μπορούν να επικοινωνούν με κάποιο πρόγραμμα που τρέχει στον προσωπικό μας υπολογιστή. Τα περιφερειακά του μπορούν να συναρμολογηθούν στο χέρι ή να αγοραστούν έτοιμα.

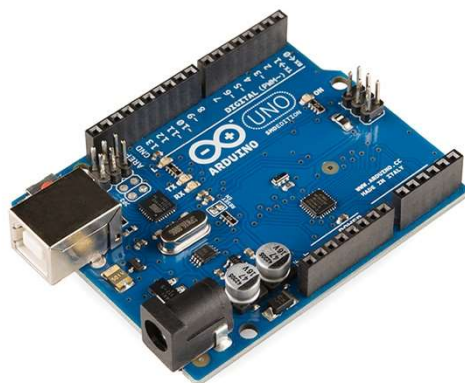
Το Arduino βέβαια, δεν είναι ο μοναδικός τρόπος για την δημιουργία μιας οποιασδήποτε διαδραστικής ηλεκτρονικής συσκευής. Το κύριο πλεονέκτημά του είναι η τεράστια κοινότητα που το υποστηρίζει και η οποία έχει δημιουργήσει, συντηρεί και επεκτείνει μια online γνωσιακή βάση, ανάλογου μεγέθους. Υπάρχουν αμέτρητα forums και ιστοσελίδες που αναφέρονται στο Arduino και μπορούν να καθοδηγήσουν, βοηθήσουν, διδάξουν και εμπνεύσουν τον κάθε χρήστη. Επιπλέον, το λογισμικό που χρησιμοποιείται είναι ανοιχτού κώδικα, γεγονός που τον καθιστά διαθέσιμο στον οποιοδήποτε για να τον μελετήσει και τροποποιήσει σύμφωνα με τις ανάγκες του. Οι χρήστες μπορούν μέσα από τις βιβλιοθήκες τις C++ αλλά και μέσα από τις βιβλιοθήκες του προγράμματος του Arduino να γράψουν τον δικό τους κώδικα και να τον μοιραστούν. Το ίδιο ισχύει και για το υλικό (hardware), καθώς τα σχέδια των περιφερειακών του είναι ανοικτού κώδικα, πράγμα που σημαίνει ότι οι χρήστες μπορούν να τα επεκτείνουν και να τα αναβαθμίσουν. Τέλος, ένα από τα βασικότερα πλεονεκτήματα του Arduino είναι το χαμηλό κόστος και το γεγονός πως το IDE του



Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

Arduino είναι συμβατό με τα τρία δημοφιλέστερα λειτουργικά συστήματα (Windows, Mac OS, Linux) εξυπηρετώντας σχεδόν όλο το εύρος των χρηστών προσωπικών υπολογιστών.

## 1.2 Arduino Uno



Εικόνα 1.2:Arduino Uno R3

Το Arduino Uno είναι βασισμένο στο μικροελεγκτή ATmega328P. Διαθέτει 14 ψηφιακούς ακροδέκτες και 6 αναλογικούς. Επίσης διαθέτει σειριακό interface. Ο μικροελεγκτής ATmega υποστηρίζει σειριακή επικοινωνία, την οποία το Arduino προωθεί μέσα από έναν ελεγκτή Serial-over-USB ώστε να συνδέεται με τον υπολογιστή μέσω USB. Η σύνδεση αυτή χρησιμοποιείται για τη μεταφορά των προγραμμάτων που σχεδιάζονται από τον υπολογιστή στο Arduino αλλά και για αμφίδρομη επικοινωνία του Arduino με τον υπολογιστή μέσα από το πρόγραμμα την ώρα που εκτελείται. Η συχνότητα του είναι στα 16mhz, η οποία δεν είναι η καλύτερη που υπάρχει αλλά είναι αρκετή για τις περισσότερες εφαρμογές.

### 1.2.1 Τροφοδοσία

Το Arduino Uno τροφοδοτείται μέσω σύνδεσης USB από κάποιον υπολογιστή ή μέσω εξωτερικής τροφοδοσίας από κάποιον μετασχηματιστή ή από μπαταρία. Για την σωστή λειτουργία του, η επιθυμητή τροφοδοσία πρέπει να είναι από 7 έως 20V διότι η μικρότερη τροφοδοσία μπορεί να προκαλέσει αστάθεια ενώ η μεγαλύτερη υπερθέρμανση και καταστροφή της πλακέτας.

Οι ακροδέκτες τροφοδοσίας είναι:

**Vin:** Σε συνδυασμό με το pin γείωσης χρησιμοποιείται ως μέθοδος τροφοδοσίας του Arduino στην περίπτωση που δεν χρησιμοποιείται εξωτερική τροφοδοσία από μετασχηματιστή ή μπαταρία. Σε αντίθετη περίπτωση το pin αυτό μπορεί να τροφοδοτήσει άλλα εξαρτήματα με τάση ίση με 7-20V δηλαδή την τάση πριν την επεξεργαστεί ο ρυθμιστής τάσης.

**5V:** Με το pin αυτό τροφοδοτούνται τα διάφορα εξαρτήματα με τάση 5V. Η τάση αυτή προέρχεται είτε από την θύρα USB που λειτουργεί στα 5V είτε από την εξωτερική τροφοδοσία με την βοήθεια του ρυθμιστή τάσεως.

**3.3V:** Η τάση αυτή παράγεται από το ολοκληρωμένο FTDI και τροφοδοτεί με 3.3V. Δεν προέρχεται από την εξωτερική τροφοδοσία αλλά παράγεται από τον ελεγκτή Serial-over-USB και έτσι το όριο άντλησης ρεύματος είναι 50mA.

**RESET:** Όταν γειωθεί, έχει ως αποτέλεσμα την επανεκκίνηση του Arduino.

**IOREF:** Παρέχει την τάση αναφοράς με την οποία λειτουργεί ο επεξεργαστής.

**GND:** Επαφή γείωσης.

### 1.2.2 Μνήμη

Ο μικροεπεξεργαστής ATmega328P έχει τρεις ομάδες μνήμης. Διαθέτει flash memory, στην οποία αποθηκεύονται τα Arduino sketch, SRAM (static random access memory), στην οποία δημιουργείται το sketch και χρησιμοποιεί τις μεταβλητές όταν τρέχει, και EEPROM, η οποία χρησιμοποιείται από τους προγραμματιστές για την αποθήκευση μακροχρόνιων πληροφοριών.

**2KB μνήμης SRAM:** Η SRAM είναι η ωφέλιμη μνήμη που χρησιμοποιείται από τα προγράμματα για την αποθήκευση μεταβλητών, πινάκων κ.α. Η μνήμη χάνει τα

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

δεδομένα της όταν σταματήσει η παροχή ρεύματος στο Arduino ή όταν πατηθεί το κουμπί επανεκκίνησης.

1KB μνήμης EEPROM: Χρησιμοποιείται για τις εγγραφές ή τις αναγνώσεις δεδομένων από τα προγράμματα. Με την απώλεια τροφοδοσίας ή επανεκκίνησης τα περιεχόμενα δεν χάνονται.

2KB μνήμης Flash: 2 KB χρησιμοποιούνται από το firmware του Arduino που έχει εγκαταστήσει ήδη ο κατασκευαστής του. Το firmware είναι αναγκαίο για την εγκατάσταση προγραμμάτων στο μικροελεγκτή μέσω της θύρας USB. Τα υπόλοιπα 30KB της μνήμης Flash χρησιμοποιούνται για την αποθήκευση αυτών ακριβώς των προγραμμάτων, αφού πρώτα μεταγλωττιστούν στον υπολογιστή. Η μνήμη Flash, δε χάνει τα περιεχόμενά της με απώλεια τροφοδοσίας ή επανεκκίνησης.

### 1.2.3 Ακροδέκτες

Το Arduino Uno διαθέτει 14 ψηφιακούς ακροδέκτες που μπορούν να χρησιμοποιηθούν είτε σαν είσοδοι είτε σαν έξοδοι. Κάθε ακροδέκτης λειτουργεί στα 5V και μπορεί να παρέχει ή να λαμβάνει ρεύμα μέγιστης τιμής 40mA. Επίσης διαθέτει μία εσωτερική pull-up αντίσταση τιμής 20-50kOhms. Σαν έξοδος, κάποιο pin μπορεί μέσω κάποιου προγράμματος να τεθεί σε κατάσταση LOW ή HIGH και το Arduino ανάλογα με την κατάσταση να διοχετεύσει ή όχι ρεύμα σε αυτό το pin. Έτσι μπορεί να γίνει ο έλεγχος ενός led που είναι συνδεδεμένο σε αυτό. Σαν είσοδος, το Arduino μπορεί να διαβάζει την κατάσταση του pin που αλλάζει με βάση την παροχή ρεύματος από κάποια εξωτερική συσκευή, για παράδειγμα έναν διακόπτη. Κάποιοι ακροδέκτες εκτός από την I/O λειτουργία διαθέτουν και κάποιες επιπρόσθετες.

Serial 0,1: Λειτουργούν ως RX και TX αντίστοιχα, όταν ενεργοποιείται η σειριακή λειτουργία. Σε αυτήν την λειτουργία τα δεδομένα στέλνονται και στη θύρα USB μέσω του ελεγκτή Serial-Over-USB και στο pin 0, για να τα διαβάσει κάποια άλλη συσκευή. Στην περίπτωση που δέχεται δεδομένα σειριακά, αυτά είναι διαθέσιμα και στο pin 1. Με αυτό τον τρόπο το Arduino χρησιμοποιεί 2 ψηφιακές εισόδους/εξόδους λιγότερες. Το Arduino διαθέτει και 2 εξωτερικά leds RX και TX τα οποία ενεργοποιούνται αναλόγως την διαδικασία.

Serial 2,3: Λειτουργούν ως εξωτερικά interrupt. Δηλαδή μέσω ενός προγράμματος μπορούν να ρυθμιστούν ώστε να λειτουργούν αποκλειστικά ως ψηφιακές εισοδοι και να σταματάει η κανονική ροή του προγράμματος όταν συμβαίνουν συγκεκριμένες αλλαγές σε αυτές και στη συνέχεια να εκτελείται μία συγκεκριμένη συνάρτηση. Τα εξωτερικά interrupt είναι ιδιαίτερα χρήσιμα σε εφαρμογές που απαιτούν συγχρονισμό μεγάλης ακρίβειας.

Serial 3,5,6,9,10,11: Μπορούν να λειτουργήσουν ως ψευδοαναλογικές έξοδοι με το σύστημα PWM (Pulse Width Modulation). Αυτό το σύστημα χρησιμοποιείται ευρέως για τον ακριβή έλεγχο κινητήρων PWM. Ένα καλό παράδειγμα αποτελούν οι μητρικές των υπολογιστών που χρησιμοποιούν το εν λόγω σύστημα για να ελέγχουν τις ταχύτητες των ανεμιστήρων. Με τον ίδιο τρόπο, μπορεί να συνδεθεί ένα LED σε αυτούς τους ακροδέκτες και να γίνει ο έλεγχος της φωτεινότητάς του με ανάλυση 8bit δηλαδή 256 καταστάσεις από 0-σβηστό ως 255-πλήρως αναμμένο. Έτσι αυτά τα pin διαφοροποιούνται από τα υπόλοιπα που έχουν απλά την δυνατότητα αναμμένο-σβηστό. Το PWM δεν είναι πραγματικά αναλογικό σύστημα, δηλαδή θέτοντας στην έξοδο την τιμή 127, δεν σημαίνει ότι η έξοδος θα δίνει 2.5V αντί της κανονικής τιμής των 5V, αλλά ότι θα δίνει ένα παλμό που θα εναλλάσσεται με μεγάλη συχνότητα και για ίσους χρόνους μεταξύ των τιμών 0 και 5V.

Serial 10,11,12,13: Οι ακροδέκτες αυτοί υποστηρίζουν SPI επικοινωνία με χρήση της βιβλιοθήκης SPI.

Serial 13: Σε αυτόν τον ακροδέκτη είναι συνδεδεμένο ένα ενσωματωμένο στο Arduino, Led. Όταν το pin είναι σε κατάσταση HIGH το led ανάβει και όταν είναι LOW σβήνει.

Το Uno διαθέτει 6 αναλογικές εισόδους A0-A5 με 10 bit ανάλυσης δηλαδή 1024 διαφορετικές τιμές. Κανονικά μετράνε από 0 έως 5 V αλλά είναι δυνατό να μεταβληθεί το άνω άκρο μέσω του ακροδέκτη AREF και της analogReference(). Ο AREF είναι ένας ακροδέκτης που δίνει την τροφοδοσία αναφοράς. Η κύρια λειτουργία των αναλογικών ακροδεκτών είναι να διαβάζουν αναλογικά σήματα αλλά επίσης μπορούν να χρησιμοποιηθούν και ως ψηφιακοί. Επιπροσθέτως οι ακροδέκτες 4 και 5 μπορούν να χρησιμοποιηθούν σαν SDA και SCL αντίστοιχα σε TWI(I2C) επικοινωνία.

### 1.3 Πρότυπα ασύρματης επικοινωνίας

Στη σύγχρονη εποχή, το ενδιαφέρον για ασύρματα συστήματα σε ιατρικές εφαρμογές αυξάνεται ραγδαία. Προσφέροντας πληθώρα πλεονεκτημάτων έναντι των ενσύρματων εναλλακτικών, όπως, ευκολία χρήσης, μειωμένο κίνδυνο μόλυνσης, μειωμένες πιθανότητες αποτυχίας, μείωση της δυσφορίας του ασθενούς και αυξημένη κινητικότητα, οι ασύρματες εφαρμογές προσφέρουν ιδιαίτερες δυνατότητες για νέες εφαρμογές στην ιατρική αγορά.

Φορητές συσκευές, όπως οθόνες απεικόνισης καρδιακού παλμού, οξύμετρα παλμού, σπιρόμετρα και μετρητές αρτηριακής πίεσης, αποτελούν βασικά όργανα εντατικής θεραπείας. Παραδοσιακά, οι αισθητήρες για το όργανα αυτά συνδέονται με τον ασθενή με καλώδια, συνεπώς και ο ασθενής παραμένει κλινήρης. Επιπλέον, όταν ο ασθενής πρέπει να μετακινηθεί, όλες οι συσκευές παρακολούθησης πρέπει να αποσυνδεθούν και να συνδεθούν ξανά. Πλέον, όλες αυτές οι χρονοβόρες εργασίες θα μπορούσαν να τερματιστούν και οι ασθενείς θα μπορούσαν να απελευθερωθούν από τα όργανα και το κρεβάτι μέσω της ασύρματης τεχνολογίας. Ενσωματώνοντας τεχνολογία ασύρματης σύνδεσης, αυτές οι συσκευές θα μπορούσαν να επικοινωνούν με μια πύλη συνδεμένη στο δίκτυο ιατρικού κέντρου και να μεταδίδουν δεδομένα από τους αισθητήρες σε αντίστοιχους αποθηκευτικούς χώρους, για παρακολούθηση, έλεγχο ή αξιολόγηση σε πραγματικό χρόνο ή και εκτός σύνδεσης, μετά την αποθήκευσή τους.

Η ταχεία ανάπτυξη των τεχνολογιών επεκτείνει τη δυνατότητα εκμετάλλευσης της αγοράς ασύρματων ιατρικών εφαρμογών. Χάρη στο μεγάλη κλίμακας ασύρματο δίκτυο, οι θεράποντες ιατροί μπορούν να έχουν πρόσβαση σε πληροφορίες ζωτικής ή μη σημασίας, αναφορικά με τους ασθενείς τους οπουδήποτε και οποτεδήποτε. Καλύπτοντας ένα ευρύ φάσμα περιπτώσεων σε ποικίλα περιβάλλοντα και εξυπηρετώντας διαφορετικές απαιτήσεις, δεν υπάρχει κάποιο ασύρματο πρότυπο επικοινωνίας που μπορεί να υπερισχύσει έναντι των υπολοίπων. Στο στάδιο αυτό θα αναφερθούν οι κυρίαρχες τεχνολογίες ασύρματης συνδεσιμότητας, παραθέτοντας τα βασικά χαρακτηριστικά τους, βάσει των οποίων επιλέγονται για τις εκάστοτε εφαρμογές υγείας.

### 1.3.1 ZigBee

Πρόκειται για μία ασύρματη τεχνολογία της ομάδας PAN (Personal Area Network), ένα ανοιχτό παγκόσμιο πρότυπο που έχει σχεδιαστεί ειδικά για χρήση σε δίκτυα M2M (Machine to Machine). Η τεχνολογία είναι οικονομική και δεν απαιτεί πολλή ισχύ, καθιστώντας την ιδανική λύση για πολλές βιομηχανικές εφαρμογές. Η τεχνολογία έχει μικρή καθυστέρηση μεταφοράς και μικρό κύκλο λειτουργίας, επιτρέποντας στα προϊόντα να μεγιστοποιούν τη διάρκεια ζωής της μπαταρίας.

Το πρωτόκολλο ZigBee προσφέρει κρυπτογράφηση AES 128bit. Η τεχνολογία χρησιμοποιείται, επίσης, σε δίκτυα mesh, τα οποία επιτρέπουν στους κόμβους να συνδέονται μεταξύ τους μέσω πολλαπλών οδών. Η ασύρματη τεχνολογία προσδοκά να επικρατήσει έναντι άλλων σε συσκευές, όπως οι έξυπνες οικιακές συσκευές. Η δυνατότητα της τεχνολογίας να συνδέει ταυτόχρονα πολλαπλές συσκευές την καθιστά ιδανική για ένα συνδεδεμένο οικιακό περιβάλλον όπου οι χρήστες επιθυμούν οι διάφορες έξυπνες οικιακές συσκευές να επικοινωνούν μεταξύ τους.

### 1.3.2 IEEE 802.11: WiFi

Το WiFi χρησιμοποιεί ραδιοκύματα για να επιτρέψει σε δύο συσκευές να επικοινωνούν μεταξύ τους. Η τεχνολογία χρησιμοποιείται συνήθως για τη σύνδεση δρομολογητών διαδικτύου σε συσκευές όπως υπολογιστές, tablet και τηλέφωνα, ωστόσο, μπορεί να χρησιμοποιηθεί για τη σύνδεση οποιονδήποτε δύο εξαρτημάτων υλικού (hardware). Το WiFi είναι ένα τοπικό ασύρματο δίκτυο που λειτουργεί σύμφωνα με τα πρότυπα 802.11 που ορίζει το Ινστιτούτο Ηλεκτρολόγων και Ηλεκτρονικών Μηχανικών (IEEE).

Το WiFi μπορεί να χρησιμοποιεί τις παγκόσμιες ζώνες ραδιοσυχνοτήτων UHF 2.4GHz και 5GHz SHF<sup>1</sup> ISM<sup>2</sup>. Ο οργανισμός «WiFi Alliance» πιστοποιεί ορισμένα προϊόντα, επιτρέποντάς τους να χαρακτηριστούν ως «Wi-Fi Certified». Για να λάβει τον ορισμό αυτό, το προϊόν πρέπει να περάσει από τη δοκιμή πιστοποίησης

---

<sup>1</sup> Ο όρος Super High Frequency (SHF) χρησιμοποιείται για ραδιο-συχνότητες (RF) μεταξύ 3 και 30 GHz.

<sup>2</sup> Industrial, Scientific and Medical (ISM): αναφέρεται σε εύρη συχνοτήτων «κατελιμμένα» για βιομηχανική, επιστημονική και ιατρική χρήση, εκτός των τηλεπικοινωνιών.

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

διαλειτουργικότητας του οργανισμού. Τα πρότυπα 802.11b, 802.11g και 802.11n λειτουργούν στη ζώνη ISM 2.4GHz. Η ζώνη είναι επιρρεπής σε παρεμβολές από ορισμένες συσκευές Bluetooth καθώς και από ορισμένους φούρνους μικροκυμάτων και ασύρματα τηλέφωνα.

### 1.3.3 Bluetooth και BLE

Το Bluetooth και το Bluetooth Low Energy (BLE) είναι ασύρματες τεχνολογίες που χρησιμοποιούνται για τη μεταφορά δεδομένων σε μικρές αποστάσεις. Η τεχνολογία χρησιμοποιείται συχνά σε μικροσυσκευές που συνδέονται με «έξυπνα» κινητά τηλέφωνα και tablets. Η τεχνολογία Bluetooth Low Energy χρησιμοποιεί λιγότερη ενέργεια από την τυπική τεχνολογία Bluetooth και χρησιμοποιείται σε συσκευές όπως συσκευές παρακολούθησης γυμναστικής, έξυπνα ρολόγια χειρός, ακουστικά και άλλες συνδεδεμένες συσκευές, προκειμένου να μεταδίδει ασύρματα δεδομένα χωρίς να επηρεάζει σε μεγάλο βαθμό την ισχύ της μπαταρίας του τηλεφώνου ενός χρήστη.

Το Bluetooth χρησιμοποιεί ραδιοκύματα UHF για μεταφορά δεδομένων. Η τεχνολογία αρχικά τυποποιήθηκε ως IEEE 802.15.1, αλλά το IEEE δεν διατηρεί πλέον αυτό το συγκεκριμένο πρότυπο. Οι εταιρείες που συνεργάζονται με το Bluetooth συνδέονται συχνά με την ομάδα ειδικών συμφερόντων Bluetooth (SIG). Η ομάδα έχει επί του παρόντος πάνω από 20.000 μέλη και πρέπει να πιστοποιήσει ένα προϊόν πριν να διατεθεί στην αγορά ως συσκευή Bluetooth. Η πιστοποίηση αυτή διασφαλίζει ότι όλες οι συσκευές Bluetooth λειτουργούν με τυποποιημένο τρόπο και παρέχουν παρόμοια εμπειρία στους καταναλωτές.

#### 1.4 Optical mouse

Το optical mouse ή αλλιώς οπτικό ποντίκι, αντικατέστησε με επιτυχία το παλιό ποντίκι με την περιστρεφόμενη μπάλα. Χρησιμοποιώντας έναν μικροσκοπικό αισθητήρα φωτός (κάμερα), καταφέρνει με την βοήθεια ενός μικροεπεξεργαστή να μεταφράσει την κάθε κίνηση σε σχετικές συντεταγμένες. Στέλνοντας στον υπολογιστή αυτές τις συντεταγμένες επιτυγχάνεται η κίνηση του κέρσορα στην οθόνη.



Εικόνα 1.3: Optical mouse

Πιο συγκεκριμένα, το οπτικό ποντίκι φωτίζει την επιφάνεια εργασίας με LED για να αποκαλύψει ένα μικροσκοπικό μοτίβο προβολών και σκιών. Αυτά τα σχέδια αντικατοπτρίζονται στον αισθητήρα πλοήγησης του ποντικιού, που παίρνει χιλιάδες εικόνες ανά δευτερόλεπτο. Οι ίδιες εικόνες επεξεργάζονται από τον επεξεργαστή ψηφιακού σήματος (DSP) για τον προσδιορισμό της κατεύθυνσης και της απόστασης κίνησης. Κατά την επεξεργασία των εικόνων, συγκρίνονται όλες οι διαδοχικές φωτογραφίες και από τις διαφορές τους προκύπτει η ταχύτητα και η κατεύθυνση τού ποντικιού, δημιουργώντας έτσι τιμές για τη σχετική αλλαγή της θέσης. Οι πληροφορίες κίνησης είναι προσβάσιμες από ένα μικροελεγκτή μέσα στο ποντίκι και στη συνέχεια αποστέλλονται στον υπολογιστή, καθοδηγώντας την ακριβή θέση του κέρσορα στην οθόνη του υπολογιστή.



Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

Αν και είναι πιο βολικό από τα παλιά ποντίκια με την μπίλια, έχει και αυτό τους περιορισμούς του: δεν μπορεί να ανταπεξέλθει σε γυαλί ή καθρέφτη. Αυτό συμβαίνει γιατί, όταν η κάμερα πάει να διαβάσει τις φωτογραφίες, δεν παίρνει αντιπροσωπευτικές εικόνες, λόγω διάθλασης και ανάκλασης. Επίσης, μπορεί να μην αποδίδει ικανοποιητικά σε κάποιες επιφάνειες που είναι λίγο τραχιές ή σε 3D mousepad. Παρόλα αυτά, δε χρειάζεται mousepad, λειτουργεί σε οποιαδήποτε σχεδόν επιφάνεια, δε λερώνονται οι αισθητήρες του και δε φθείρεται στο εσωτερικό ώστε να υπολειτουργεί και να αχρηστευτεί και, τέλος, έχει μεγαλύτερη ακρίβεια.

## 2. ΣΥΣΤΗΜΑΤΑ ΑΝΟΙΧΤΟΥ ΚΩΔΙΚΑ ΣΕ ΙΑΤΡΙΚΕΣ ΕΦΑΡΜΟΓΕΣ

Έχοντας ήδη περιγράψει και κατανοήσει πλήρως τον τρόπο λειτουργίας και κατασκευής της πλατφόρμας Arduino, μίας ανοιχτού κώδικα υπολογιστικής πλατφόρμας, κρίνεται σκόπιμο να αναφερθούν ορισμένα παραδείγματα συσκευών στην κατασκευή των οποίων χρησιμοποιήθηκε η πλατφόρμα. Αρχικά, προσδιορίζεται ο όρος των ανοιχτού κώδικα συστημάτων και τα πλεονεκτήματά τους για χρήση στον τομέα της ιατρικής και στη συνέχεια γίνεται περιγραφή ορισμένων εφαρμογών τους.

Με τον όρο συστήματα ανοιχτού κώδικα (open source hardware) νοούνται τα συστήματα τα οποία έχουν σχεδιαστεί ώστε να είναι διαθέσιμα δημόσια στον καθένα που επιθυμεί να τα μελετήσει, τροποποιήσει και διανέμει περαιτέρω. Ανάμεσα στα διάφορα ερευνητικά έργα στα οποία μπορούν να χρησιμοποιηθούν συστήματα ανοιχτού κώδικα περιλαμβάνονται δυνητικά και συσκευές ιατρικής φύσεως. Η προσέγγιση του ανοιχτού κώδικα προσφέρει πληθώρα πλεονεκτημάτων, μεταξύ των οποίων συγκαταλέγονται η μείωση του κόστους και η δυνατότητα ταχύτερης εξέλιξης και καινοτομίας.

Ένα ευρύ φάσμα συστημάτων ανοιχτού κώδικα που περιλαμβάνει αισθητήρες υγειονομικής περίθαλψης και χαμηλού κόστους υπολογιστές είναι άμεσα διαθέσιμο στην αγορά. Το Arduino, είναι αδιαμφισβήτητα μία από τις πιο διαδεδομένες και επιτυχημένες πλατφόρμες ανοιχτού κώδικα, περιέχει έναν μικροελεγκτή με ένα εύχρηστο περιβάλλον ανάπτυξης το οποίο το καθιστά ιδανικό για την κατασκευή συσκευών. Έχει μια μεγάλη κοινότητα υποστηρικτών και υπάρχουν διαθέσιμες βιβλιοθήκες ανοιχτού λογισμικού για την απλούστευση πολλών εργασιών.

Η ριζική αλλαγή στην προσέγγιση είναι ότι αυτά τα νέα σχέδια είναι ανοιχτά. Κατ' αρχήν, ο καθένας μπορεί να κατασκευάσει ένα Arduino, επειδή ο πλήρης σχεδιασμός του είναι διαθέσιμος στον οποιονδήποτε για να τον χρησιμοποιήσει. Αυτό το άνοιγμα έχει ωθήσει πολλούς ανθρώπους να κατασκευάσουν ένα τεράστιο φάσμα αισθητήρων και άλλων συσκευών που είναι συμβατές με αυτό, αυξάνοντας σημαντικά τις πωλήσεις τους και μειώνοντας τις τιμές τους στην παγκόσμια αγορά, κυρίως εξαιτίας του γεγονότος ότι υπάρχει ελεύθερη πρόσβαση στα σχέδια τους.

Η κατασκευή του υλικού που διατίθεται με άδεια ανοιχτού κώδικα επιτρέπει σε οποιονδήποτε να βελτιώσει και να συμβάλει στον σχεδιασμό της συσκευής, οδηγώντας σε πολύ ταχεία καινοτομία σε σύγκριση με τις παραδοσιακές μεθόδους. Επίσης, επιτρέπει στο σχεδιασμό να τροποποιηθεί για πολύ συγκεκριμένες χρήσεις και οι συσκευές είναι εύκολο να επισκευαστούν, παράγοντες που μειώνουν επίσης τις επιπτώσεις που αυτές οι συσκευές έχουν στο περιβάλλον. Μια ιατρική συσκευή ανοιχτού κώδικα αυξάνει την ασφάλεια και την ευρωστία επιτρέποντας σε περισσότερα άτομα να επιθεωρούν και να βελτιώνουν τα σχέδιά τους. Στην πραγματικότητα, είναι τόσο δύσκολο να αναπτυχθεί καλή ασφάλεια υπολογιστών ώστε οι καλύτεροι αλγόριθμοι κρυπτογράφησης αναπτύσσονται πάντα ανοιχτά για να επιτρέπουν σε άλλους να τα επιθεωρήσουν και να τα βελτιώσουν. Επιπλέον, εάν το λογισμικό για τη σύνδεση συσκευών είναι ανοιχτού κώδικα και οι φυσικές διεπαφές τυποποιημένες, μπορεί να αποτραπεί ο αποκλεισμός του προμηθευτή. Το λογισμικό ανοιχτού κώδικα επιτρέπει, επίσης, σε πολλούς κατασκευαστές να χρησιμοποιούν τον ίδιο σχεδιασμό, αλλά να διαφοροποιούνται με άλλους τρόπους προκειμένου να ανταγωνιστούν, για παράδειγμα όσον αφορά τη χρηστικότητα, την υποστήριξη ή την ευρύτερη διαλειτουργικότητα με άλλες συσκευές. Η ίδια προσέγγιση μπορεί να βελτιώσει τις ιατρικές συσκευές.

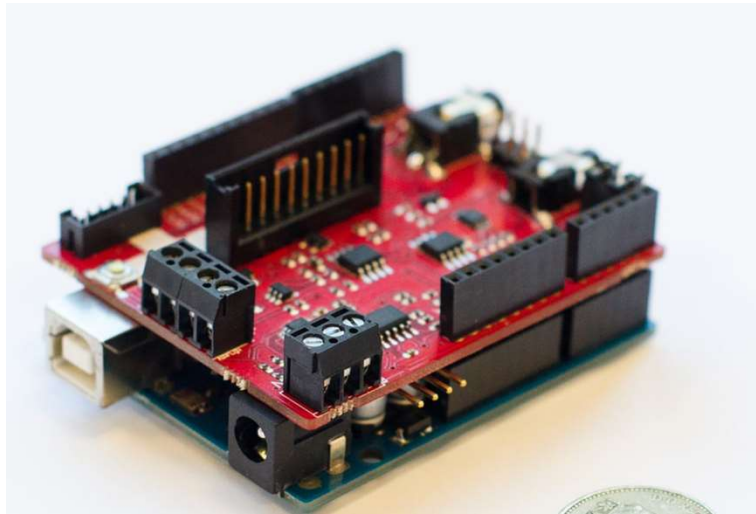
Μία πλήρης και συστημική προσέγγιση των συσκευών ανοιχτού κώδικα που έχουν κατασκευαστεί και χρησιμοποιηθεί στον τομέα της ιατρικής είναι αδύνατο να πραγματοποιηθεί σε αυτό το στάδιο, καθώς, αφενός αρκετές από αυτές είναι ακόμη σε αρχικό στάδιο και δεν έχουν δημοσιευθεί στην επιστημονική βιβλιογραφία, αφετέρου, θα αποτελούσε αυτοτελές αντικείμενο μιας άλλης πτυχιακής εργασίας. Ωστόσο, κρίνεται σκόπιμο να αναφερθούν ενδεικτικά ορισμένα παραδείγματα.

#### e-Health sensor platform

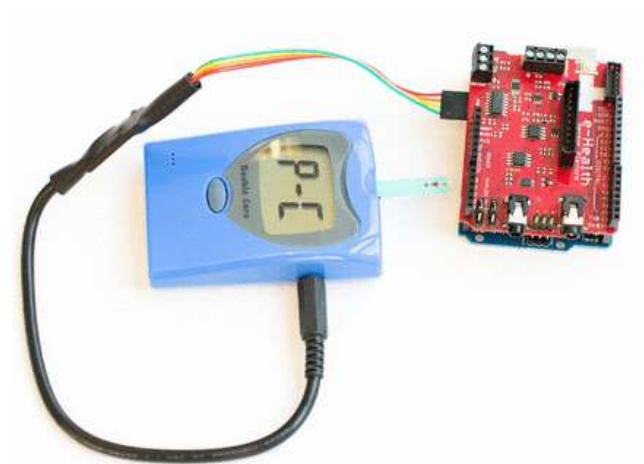
Με στόχο την παρακολούθηση της φυσιολογίας των ασθενών κατασκευάστηκε από τον Ισπανό κατασκευαστή ηλεκτρονικών προϊόντων Libelium η πλατφόρμα αισθητήρων e-Health30. Πρόκειται για μια ανοικτή πλατφόρμα φυσιολογικής παρακολούθησης που χρησιμοποιεί εννέα διαφορετικούς αισθητήρες: παλμό, οξυγόνο

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

στο αίμα (SpO<sub>2</sub>), ροή αέρα (αναπνοή), θερμοκρασία σώματος, ηλεκτροκαρδιογράφημα, γλυκόμετρο, σφυγμομανόμετρο και τη θέση του ασθενούς (επιταχυνσιόμετρο). Η πλατφόρμα, που παρουσιάζεται στις εικόνες, αποτελείται από ένα board ή shield που συνδέεται σε Arduino ή Raspberry Pi και διασυνδέεται με τους διάφορους αισθητήρες, επιτρέποντας άμεσα την επεξεργασία των σημάτων που προέρχονται από αυτούς τους αισθητήρες.



Εικόνα 2.1: Shield connected with Arduino



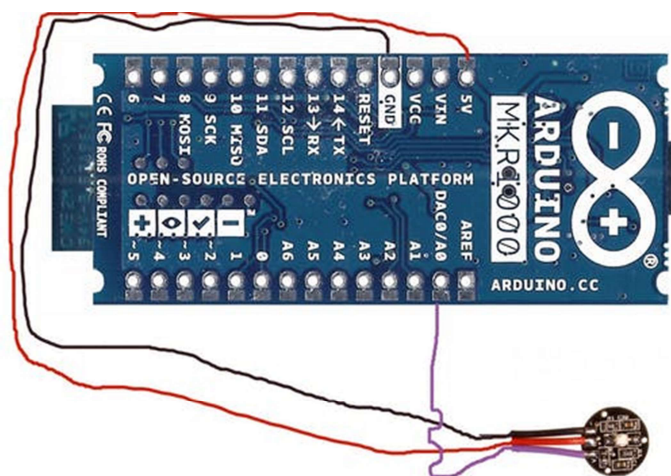
Εικόνα 2.2: Αισθητήρας σε σύνδεση με το shield και το Arduino

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

Για την κατασκευή της συσκευής απαιτείται ένα Raspberry Pi ή Arduino. Δεν απαιτούνται ειδικές δεξιότητες, καθώς πρόκειται για ένα αρθρωτό κιτ όπου τα διάφορα εξαρτήματα είναι απλά συνδεδεμένα. Τα παρεχόμενα εξαρτήματα είναι αισθητήρες off-the-shelf και προϊόντα που έχουν τροποποιηθεί για να λειτουργούν με το κιτ. Ο πηγαίος κώδικας είναι διαθέσιμος για Raspberry Pi και Arduino.

### E-Health: Low Cost Sensors for Early Disease Detection

Οι ζωτικές ενδείξεις των ανθρώπων είναι εξαιρετικά κρίσιμες στη σύγχρονη ιατρική. Σε αυτό το πρότζεκτ ο Saudin Dizdarevic σχεδίασε ένα σύστημα για τη μέτρηση του καρδιακού ρυθμού για μεγάλες χρονικές περιόδους. Για την κατασκευή της φορητής ηλεκτρονικής συσκευής χρησιμοποιήθηκαν ένας αισθητήρας παλμών και ένας μικροελεγκτής Arduino.



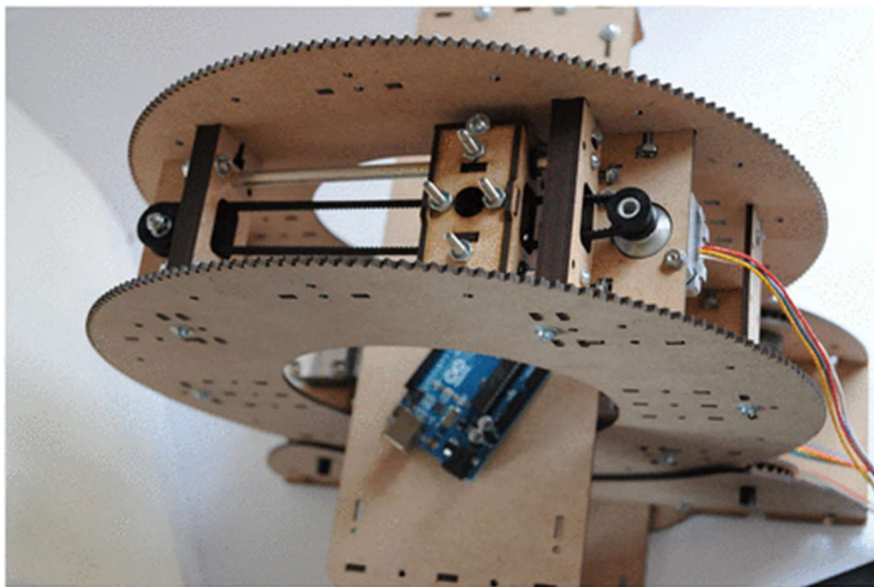
Εικόνα 2.3: Arduino και αισθητήρας παλμών

Τα βασικά πλεονεκτήματα της συγκεκριμένης εφαρμογής είναι πως έχει τη δυνατότητα να μετρά τον καρδιακό ρυθμό καθ' όλη τη διάρκεια χρήσης του αισθητήρα με μεγάλη ακρίβεια, το κόστος κατασκευής της είναι πολύ χαμηλότερο συγκριτικά με αντίστοιχης λειτουργίας συσκευές που υπάρχουν στην αγορά και υπάρχει η δυνατότητα προγραμματισμού της ώστε να διατηρείται ιστορικό των μετρήσεων για διάστημα τουλάχιστον τριών μηνών. Ο κώδικας που χρησιμοποιήθηκε για τον

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

προγραμματισμό της πλατφόρμας είναι διαθέσιμος, προκειμένου ο καθένας να μπορεί να τον χρησιμοποιήσει και εξελίξει.

CT scanner



Εικόνα 2.4: Αξονικός τομογράφος Jansen

Ο Jansen κατασκεύασε έναν επιτραπέζιο αξονικό τομογράφο για μικρά αντικείμενα με χρήση ανοιχτού κώδικα. Τα πλήρη αρχεία για το σχεδιασμό του οργάνου είναι διαθέσιμα στο διαδίκτυο, συμπεριλαμβανομένου του μηχανικού σχεδιασμού, των σχεδίων του κυκλώματος και του κώδικα. Ο αξονικός τομογράφος χρησιμοποιεί μία πολύ χαμηλής έντασης πηγή ραδιοϊσότοπου ακτίνων Χ, που σημαίνει ότι ακόμη και εικόνες χαμηλής ανάλυσης χρειάζονται ώρες για να μετρηθούν. Έχει σχεδιαστεί για ακαδημαϊκούς και εκπαιδευτικούς σκοπούς, αλλά προσδοκάται πως θα μπορούσε τελικά να χρησιμοποιηθεί ως αξονικός τομογράφος στις αναπτυσσόμενες χώρες.

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

Ο Jansen σχεδίασε τον αξονικό τομογράφο με τέσσερις άξονες κίνησης, ένας από τους οποίους είναι ένα μεγάλο περιστροφικό είδος γερανού. Κάθε γραμμικός άξονας έχει ένα μικρό φορείο που περιέχει υποστηρίγματα είτε για την πηγή είτε για τον ανιχνευτή. Για να ελέγχεται τόσο η κίνηση όσο και ο ανιχνευτής χρησιμοποιήθηκε η πλατφόρμα Arduino.

### 3. ΜΕΘΟΔΟΛΟΓΙΑ – ΕΦΑΡΜΟΓΗ

Στο κεφάλαιο αυτό θα παρουσιαστούν τα βασικά μέρη που χρησιμοποιήθηκαν για την ολοκλήρωση της εργασίας και πως συνδυάστηκαν, βάσει των τεχνολογιών που αναφέρθηκαν στο προηγούμενο μέρος, προκειμένου να κατασκευαστεί ένα γωνιόμετρο το οποίο θα έχει τη δυνατότητα να μετράει το εύρος κίνησης μιας άρθρωσης και να αποστέλλει τα δεδομένα των μετρήσεων σε έναν εξυπηρετητή (server), έτσι ώστε να μπορεί να έχει πρόσβαση σε αυτά τόσο ο ασθενής όσο και ο θεράπων ιατρός ανά πάσα στιγμή.

#### 3.1 Κατασκευή γωνιόμετρου

Ευκαμψία ή ευκινησία είναι η ικανότητα του νευρομυϊκού συστήματος για πλήρη κίνηση μιας άρθρωσης χωρίς περιορισμούς και πόνο κατά την εκτέλεσή της. Το φυσιολογικό εύρος κίνησης μιας άρθρωσης είναι αυτό που ισχύει για τον γενικό υγιή πληθυσμό. Για την επίτευξη του ενεργητικού εύρους κίνησης μιας άρθρωσης, ενεργοποιείται μια μυϊκή ομάδα η οποία και εκτελεί ένα συγκεκριμένο εύρος κίνησης. Το ενεργητικό εύρος κίνησης χρησιμοποιείται για να εκτιμηθεί τόσο το μέγεθος όσο και η ποιότητα της ενεργητικής κίνησης. Για τη μέτρηση της κινητικότητας των αρθρώσεων χρησιμοποιείται το γωνιόμετρο.

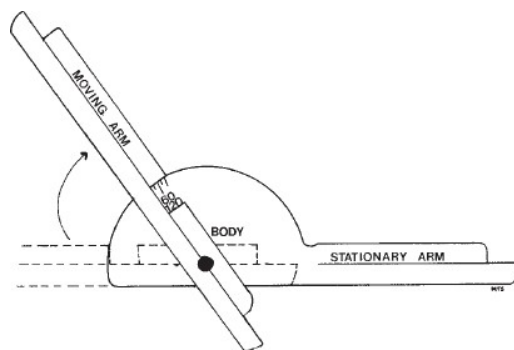


Εικόνα 3.1: Γωνιόμετρο

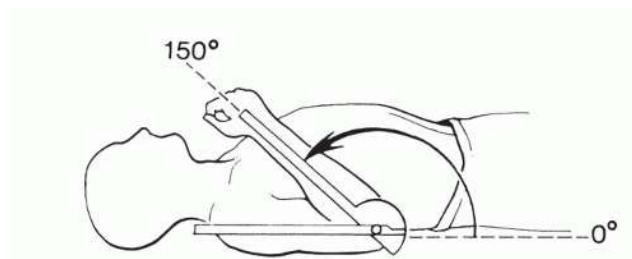


## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

Το γωνιόμετρο είναι συνήθως ηλεκτρονικό ή μηχανικό, και με το εργαλείο αυτό μπορεί να εκτιμηθεί η κινητικότητα μιας άρθρωσης ποσοτικά. Υπάρχουν πολλών ειδών γωνιόμετρα, πλαστικά, μεταλλικά κ.ά. Το κοινό γωνιόμετρο αποτελείται από το σώμα ή κυρίως στέλεχος και από δύο επιμήκεις βραχίονες, έναν σταθερό και έναν κινητό. Το σώμα του γωνιόμετρου μοιάζει με μοιρογνωμόνιο. Μπορεί να είναι κύκλος, ημικύκλιο ή τόξο κύκλου και συνήθως βρίσκεται στο άκρο του σταθερού βραχίονα. Ο σταθερός βραχίονας αποτελεί συνέχεια του σώματος και δεν παρουσιάζει ανεξάρτητη κίνηση. Ο κινητός βραχίονας συνδέεται με το κέντρο του σώματος έτσι ώστε να κινείται ελεύθερα γύρω από αυτόν. Το μήκος των βραχιόνων κυμαίνεται από 3 έως 40 εκατοστά. Ο κλινικός επιλέγει το κατάλληλο μήκος, ανάλογα με την άρθρωση που μετρά. Έτσι για τη μέτρηση της κάμψης έκτασης του ισχίου απαιτείται γωνιόμετρο με μακρούς βραχίονες, για την καλύτερη ευθυγράμμισή του με τα ανατομικά οδηγία σημεία της άρθρωσης. Αντίθετα, για τη μέτρηση των αρθρώσεων των δακτύλων επιλέγεται γωνιόμετρο με μικρό κινητό βραχίονα. Το γωνιόμετρο είναι ένα εργαλείο που χρησιμοποιείται στην αποκατάσταση και το οποίο αξιολογεί την πρόοδο της κινητικότητας των αρθρώσεων, ώστε να υπάρξει η ανάλογη παρέμβαση στο πρόγραμμα από τον κλινικό.



Εικόνα 3.2: Κίνηση κινητού βραχίονα



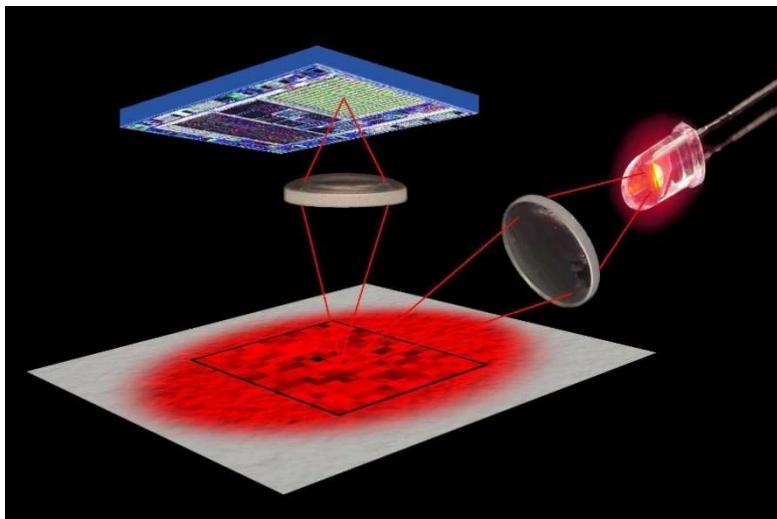
Εικόνα 3.3: Χρήση γωνιόμετρου

Για την κατασκευή του γωνιόμετρου στην παρούσα εργασία χρησιμοποιήθηκαν δύο μεταλλικές ορθογώνιες πλάκες (βραχίονες) μήκους δώδεκα εκατοστών και πλάτους ενός εκατοστού. Μεταξύ τους τοποθετήθηκε ένα ρουλεμάν, μέσω του οποίου επιτρέπεται η κυκλική κίνηση της μίας εκ των δύο, ενώ η άλλη παραμένει σταθερή. Το ρουλεμάν χρησιμοποιείται για την αποφυγή τριβών μεταξύ των επιφανειών. Προκειμένου να μπορεί να γίνει η αναγνώριση της θέσης των βραχιόνων του οργάνου, και να μετρηθεί η γωνία της κίνησης, τοποθετήθηκε μία πλακέτα οπτικού ποντικιού (optical mouse) με ADNS-2083 optical sensor chip. Για τη σταθεροποίηση της πλακέτας οπτικού ποντικιού πάνω στις πλάκες τοποθετήθηκαν μεταλλικοί κυκλικοί δίσκοι, καλυμμένοι από φύλλα φελλού για την αποφυγή βραχυκυκλωμάτων.

### 3.2 Πλακέτα οπτικού ποντικιού

Το οπτικό ποντίκι είναι ιδιαίτερα διαδεδομένο ως συσκευή εισόδου σε ηλεκτρονικούς υπολογιστές και αποτελεί μία σχετικά οικονομική συσκευή. Ωστόσο, δεν έχει δοθεί μεγάλη σημασία στο γεγονός ότι στο τσίπ του αισθητήρα ενσωματώνεται μία μικρή ψηφιακή κάμερα. Αυτό το εξάρτημα επιτρέπει την κατασκευή πολλών νέων συναρπαστικών συσκευών εισόδου, λόγω του συμπαγούς αποτυπώματος και του ενσωματωμένου επεξεργαστή ψηφιακών σημάτων. Σε αυτή την εργασία, η πλακέτα του οπτικού ποντικιού χρησιμοποιήθηκε έτσι ώστε να αναγνωρίζεται η θέση των βραχιόνων του γωνιόμετρου και να υπολογίζεται η μεταξύ τους μετατόπιση. Στη συνέχεια περιγράφεται πως ακριβώς λειτουργεί η τεχνολογία και οι δυνατότητες που παρέχει, καθιστώντας την κατάλληλη για την παρούσα κατασκευή.

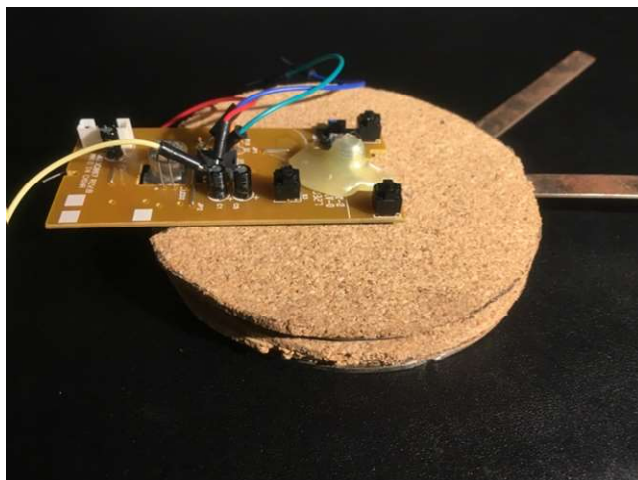
Ένα ποντίκι υπολογιστή είναι ακριβές και μπορεί να ανιχνεύσει κίνηση μικρότερη από ένα χιλιοστό. Το οπτικό ποντίκι έχει ενσωματωμένη μία πολύ μικρή κάμερα που ανιχνεύει αλλαγές στις εικόνες της επιφάνειας που βρίσκονται από κάτω του. Οι αλλαγές αυτές στη συνέχεια μετατρέπονται σε διαφορά μετατόπισης στους άξονες X και Y. Μία κόκκινη λυχνία LED φωτίζει την επιφάνεια για να αυξήσει την αντίθεση των εικόνων. Στην παρούσα κατασκευή επιλέχθηκε η χρήση του αισθητήρα οπτικού ποντικίου καθώς, δε διαθέτει μηχανικά μέρη που μπορούν να σπάσουν ή να φθαρούν ενώ, ταυτόχρονα, είναι αρκετά μικρό και οικονομικό.



Εικόνα 3.4: Τρόπος λειτουργίας οπτικού ποντικίου

Αρχικά, χρειάστηκε να ανοιχτεί ένα συμβατικό ποντίκι προκειμένου να έχουμε πρόσβαση στην πλακέτα. Στην εικόνα 12 βλέπουμε την πλακέτα του κυκλώματος με τον αισθητήρα και την λυχνία LED. Στην εικόνα 13 βλέπουμε το κάτω μέρος της πλακέτας όπου βρίσκεται ο φακός, ο οποίος χρησιμοποιείται για την ενίσχυση του φωτός καθώς και ως φακός για τον αισθητήρα που λαμβάνει τις εικόνες. Ο αισθητήρας βρίσκεται στη μέση της πλακέτας του κυκλώματος. Βάσει του δελτίου των τεχνικών χαρακτηριστικών του αισθητήρα εντοπίζονται οι ακροδέκτες +5V τάση εισόδου, GND γείωση, SCK ρολόι σειριακής θύρας και SDIO δεδομένα σειριακής θύρας.

## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών



Εικόνα 3.5: Πλακέτα optical mouse προσαρμοσμένη στην κατασκευή

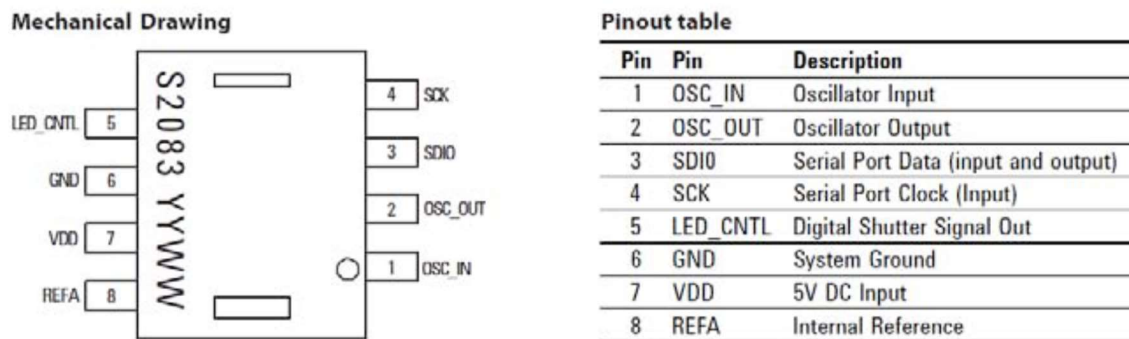


Εικόνα 3.6: Ο φακός του οπτικού ποντικιού

Ο ADNS-2083 είναι ένας αισθητήρας οπτικού ποντικιού μικρού μεγέθους που βασίζεται στην τεχνολογία οπτικής πλοήγησης, η οποία μετρά τις σχετικές μετατοπίσεις, λαμβάνοντας διαδοχικά εικόνες επιφανειών και καθορίζοντας μαθηματικά την κατεύθυνση και το μέγεθος της κίνησης. Η έξοδος πραγματοποιείται με τη χρήση μίας σειριακής θύρας δύο ακροδεκτών. Η ανάλυση είναι 400 μετρήσεις ανά ίντσα (cpi) με ταχύτητες κίνησης έως και 12 ίντσες το δευτερόλεπτο (ips).

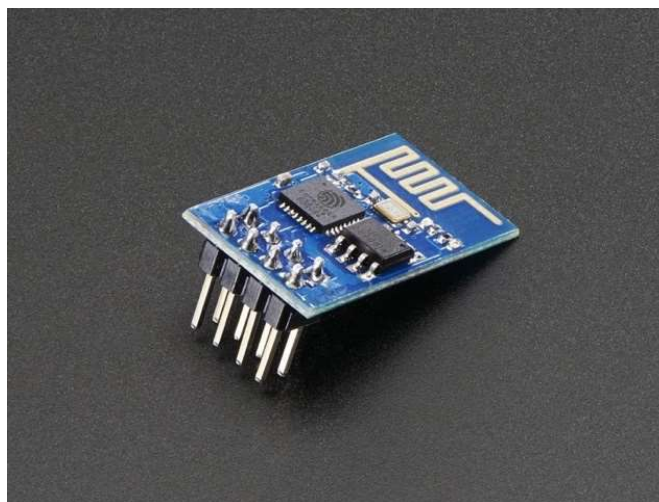
Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

Σημειώνεται ότι ο ADNS-2083 περιέχει ένα σύστημα λήψης εικόνων, έναν ψηφιακό επεξεργαστή σημάτων και μια σειριακή θύρα δύο ακροδεκτών. Στην παρούσα κατασκευή οι ακροδέκτες που χρησιμοποιήθηκαν ήταν οι 3, 4, 6 και 7, όπως εμφανίζονται στην εικόνα 14.



Εικόνα 3.7: Οι ακροδέκτες του ADNS-2083

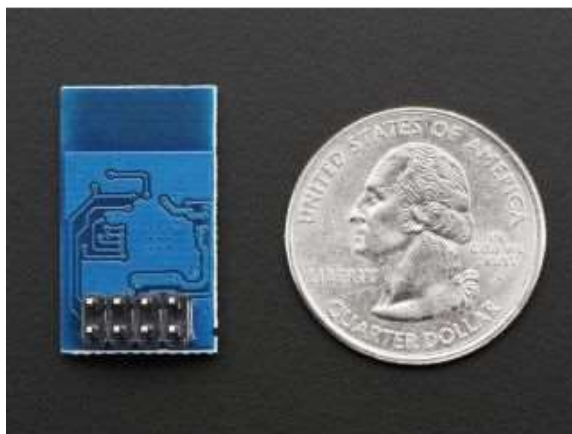
### 3.3 Πλακέτα WiFi – ESP8266



Εικόνα 3.8: ESP-8266

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

Ο ESP8266, είναι ένας οικονομικός Wi-Fi μικροεπεξεργαστής με υποστήριξη TCP/IP και δυνατότητες μικροελεγκτή, ο οποίος παράγεται από την Espressif Systems, που εδρεύει στην Σανγκάη. Οι κατασκευαστές του δυτικού κόσμου ανακάλυψαν το συγκεκριμένο επεξεργαστή τον Αύγουστο του 2014 με την πλακέτα ESP-01, που αναπτύχθηκε από την Ai-Thinker. Αυτή η μικρή πλακέτα, επιτρέπει σε μικροελεγκτές να συνδεθούν σε ασύρματο δίκτυο και μέσω αυτού να δημιουργήσουν απλές TCP/IP συνδέσεις, χρησιμοποιώντας εντολές τύπου Hayes<sup>3</sup>. Τότε όμως, δεν υπήρχε σχεδόν κανένα εγχειρίδιο χρήσης και εντολών για την πλακέτα στην αγγλική γλώσσα. Ωστόσο, η χαμηλή τιμή της και το μικρό μέγεθός της προσέλκυσε πολλούς χάκερς να εξερευνήσουν την πλακέτα, τον μικροεπεξεργαστή και το λογισμικό τους, καθώς και να μεταφράσουν τα κινεζικά εγχειρίδια.



Εικόνα 3.9: Το μέγεθος του ESP-8266

Η πλακέτα χρησιμοποιεί τον RISC<sup>4</sup> μικροεπεξεργαστή L106 των 32 bits με χρονισμό στα 80 MHz. Διαθέτει μνήμη 80 KB για χρήση δεδομένων. Είναι συμβατή με τα πρωτόκολλα WiFi IEEE 802.11 b/g/n καθώς και με τα πρωτόκολλα κρυπτογράφησης Wired Equivalent Privacy (WEP) και WiFi Protected Access (WPA/WPA2).

<sup>3</sup> Οι εντολές Hayes αναπτύχθηκαν από τον Dennis Hayes για το Hayes Smartmodem 300 το 1981.

<sup>4</sup> Reduced Instruction Set Computer: Η αρχιτεκτονική του μικροεπεξεργαστή επιτρέπει τη χρήση λιγότερων κύκλων για κάθε εντολή, σχετικά με τους Complex Instruction Set Computers (CISC).



Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

Οι ακροδέκτες σύνδεσης είναι οι εξής:

1. VCC, τροφοδοσία (+3.3V, αντοχή ως +3.6V)
2. GND, γείωση (0V)
3. RX, λήψη δεδομένων
4. TX, αποστολή δεδομένων
5. CH\_PD, (chip power-down)
6. RST, επαναφορά (reset)
7. GPIO 0, γενικού σκοπού είσοδος/έξοδος
8. GPIO 2, γενικού σκοπού είσοδος/έξοδος



Εικόνα 3.10: Οι ακροδέκτες του ESP-8266

### 3.4 Σύνδεση optical mouse (ADNS-2083) με Arduino

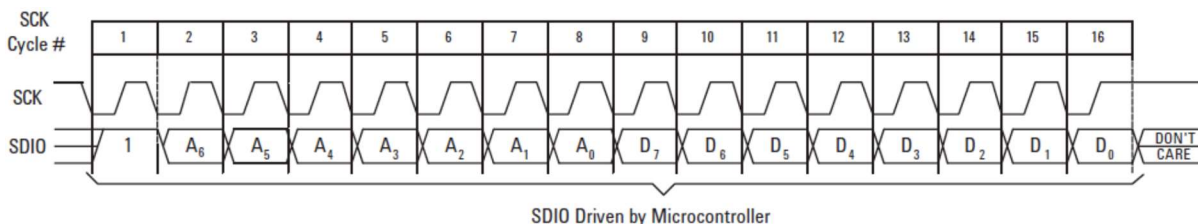
Προκειμένου να γίνει η λήψη των δεδομένων μετατόπισης που μετρώνται από την πλακέτα οπτικού ποντικιού είναι απαραίτητο να συνδεθεί με έναν μικροελεγκτή, που στην προκειμένη περίπτωση επιλέχθηκε να είναι το Arduino Uno. Τα δεδομένα μεταφέρονται από την πλακέτα του optical mouse στο Arduino συνδέοντας με καλώδια τους ακροδέκτες που αναφέρθηκαν παραπάνω στους ακροδέκτες του μικροελεγκτή. Πιο συγκεκριμένα, το +5V συνδέεται στον ακροδέκτη 5V Power στο Arduino, το GND στον ακροδέκτη GND Power, το SCK στον ψηφιακό ακροδέκτη 2 και το SDIO στον ψηφιακό ακροδέκτη 3 του Arduino.





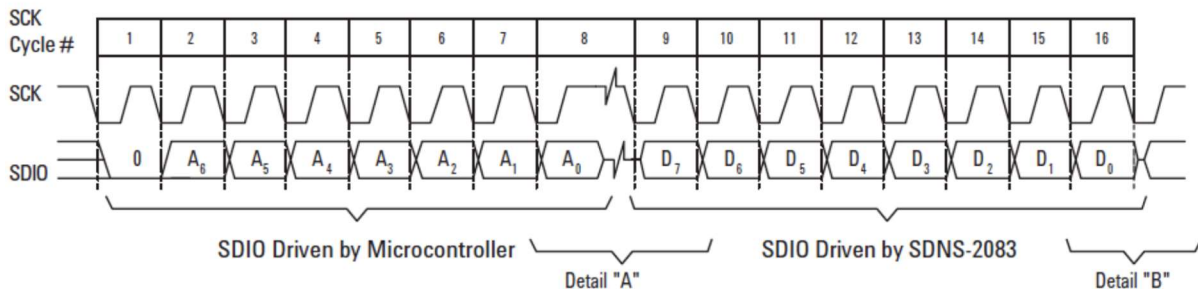
## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

της. Αυτό το «1» σημαίνει ότι ο μικροελεγκτής θέλει να γράψει δεδομένα και η διεύθυνση είναι ένα μητρώο στο οποίο θα γίνει η εγγραφή. Το δεύτερο byte περιέχει τα δεδομένα που ορίζουμε να εγγραφούν. Το ρολόι συγχρονίζει τη μεταφορά δεδομένων, επειδή ο μικροελεγκτής αλλάζει τη γραμμή δεδομένων (SDIO) στις φθίνουσες άκρες του ρολογιού και ο αισθητήρας διαβάζει το SDIO στις ανερχόμενες ακμές του ρολογιού.



Εικόνα 3.12: Διάγραμμα περιγραφής διαδικασίας εγγραφής δεδομένων στον ADNS-2083

Μια λειτουργία ανάγνωσης λειτουργεί με τον ίδιο τρόπο και αποτελείται επίσης από δύο bytes. Το πρώτο byte έχει ένα "0" ως το σημαντικότερο bit της (για να υποδείξει μια λειτουργία ανάγνωσης) και μια διεύθυνση επτά bit για να ζητήσει τα δεδομένα από μία συγκεκριμένη εγγραφή. Το δεύτερο byte είναι τα δεδομένα που αποστέλλονται από τον αισθητήρα στον μικροελεγκτή. Μετά το τελευταίο bit της διεύθυνσης υπάρχει απειροελάχιστη καθυστέρηση για να προετοιμάσει ο αισθητήρας τα δεδομένα.



Εικόνα 3.13: Διάγραμμα περιγραφής διαδικασίας ανάγνωσης δεδομένων από τον ADNS-2083

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

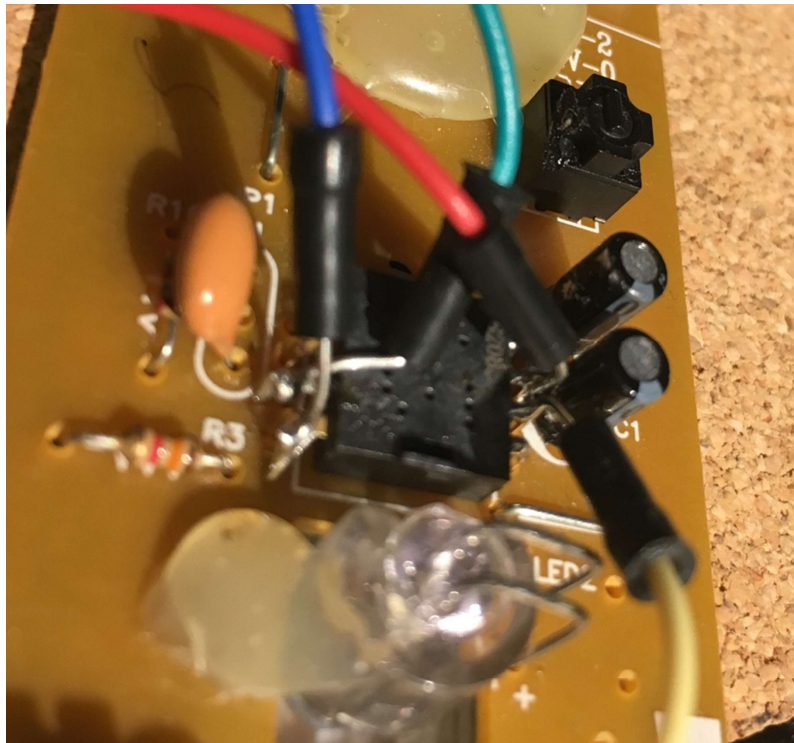
Στο δελτίο των τεχνικών χαρακτηριστικών του αισθητήρα υπάρχει μια λίστα διευθύνσεων. Από τις διευθύνσεις 0x42 και 0x43 λαμβάνεται το ΔΧ και ΔΥ (σχετική μετατόπιση στους άξονες Χ και Υ). Για παράδειγμα, προκειμένου να διαβαστεί μία τιμή του ΔΥ πρέπει να εκτελεστεί μια λειτουργία ανάγνωσης. Το πρώτο byte που στέλνει ο μικροελεγκτής είναι 0x42, αλλά με "0" ως το σημαντικότερο bit της. Σε μορφή bit, αυτό μοιάζει με "01000010". Στη συνέχεια, μετά από καθυστέρηση της τάξης των 100 μικροδευτερολέπτων ο αισθητήρας θα απαντήσει με την επιθυμητή τιμή σε ένα byte.

Address	Register	Notes
0x40	Configuration	Reset, Power Down, LED mode, Forced Awake, etc
0x41	Status	Product ID, Mouse state of Asleep or Awake
0x42	Delta_Y	Y Movement
0x43	Delta_X	X Movement
0x44	SQUAL	Measure of the number of features visible by the sensor
0x45	Maximum_Pixel	
0x46	Minimum_Pixel	
0x47	Pixel_Sum	
0x48	Pixel Data	Actual picture of sensor
0x49	Shutter High	
0x4A	Shutter Low	
0x4B	Frame Period	

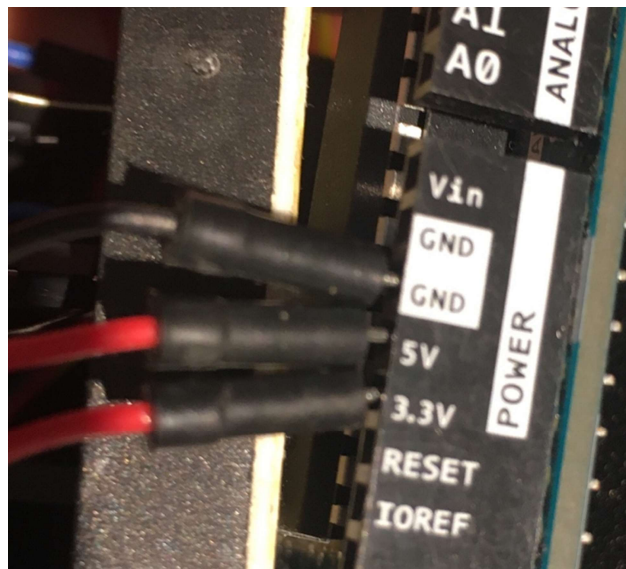
Εικόνα 3.14: Διευθύνσεις των εγγραφών του ADNS-2083

Όπως εξηγήθηκε, για να πραγματοποιηθεί η επικοινωνία ανάμεσα στο Arduino και τον μικροελεγκτή ADNS-2083 χρειάστηκε να συνδεθούν οι ακροδέκτες 3 (SDIO – μεταφορά δεδομένων) και 4 (SCK – ρυθμός μετάδοσης) του μικροεπεξεργαστή με τους ακροδέκτες 3 και 2 του Arduino αντίστοιχα. Οι συγκεκριμένοι ορίζονται στον κώδικα ώστε να αρχικοποιηθεί η σειριακή επικοινωνία μεταξύ τους χρησιμοποιώντας την [βιβλιοθήκη Optical](#). Η τροφοδοσία του ADNS-2083 γίνεται με τη χρήση των ακροδεκτών «5V» και «GND» του Arduino.

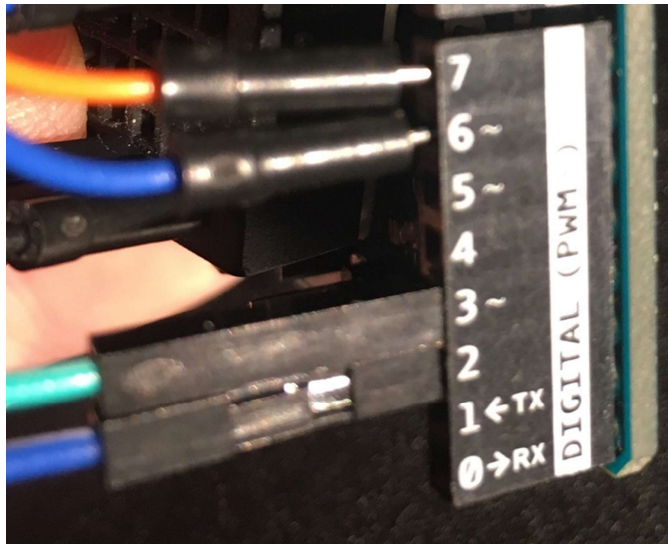
Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών



Εικόνα 3.15: Συνδέσεις/ κολλήσεις ADNS-2083



Εικόνα 3.16: Συνδέσεις τροφοδοσίας των περιφερειακών του Arduino



Εικόνα 3.17: Σειριακές συνδέσεις ανταλλαγής δεδομένων του Arduino

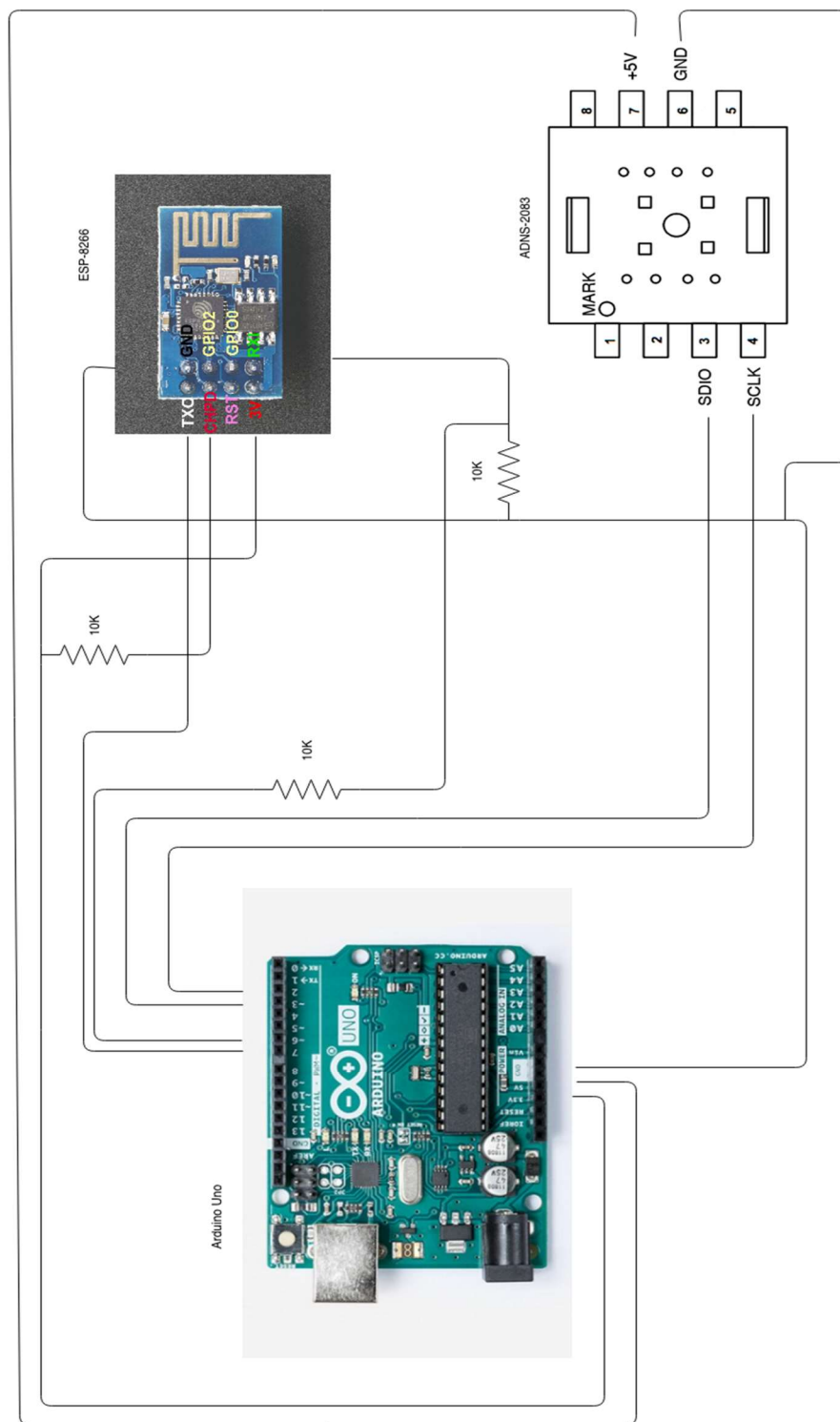
Στο παράρτημα παρατίθεται ο κώδικας που χρησιμοποιήθηκε στο προγραμματιστικό περιβάλλον του Arduino για να προγραμματιστεί η πλατφόρμα, έτσι ώστε να δέχεται τις πληροφορίες από τον αισθητήρα οπτικού ποντικιού, μαζί με το σχέδιο των συνδέσεων.

### 3.5 Σύνδεση Arduino με WiFi module

Προκειμένου να αποθηκευτούν οι μετρήσεις, πρέπει το Arduino να συνδεθεί στο διαδίκτυο, έτσι ώστε να σταλούν τα δεδομένα σε server όπου και θα κρατηθούν. Για να επιτευχθεί η σύνδεση της κατασκευής σε ασύρματο δίκτυο (WiFi) και από εκεί στο διαδίκτυο, χρησιμοποιήθηκε το ESP8266. Το ESP8266 είναι ένα οικονομικό (~5\$) WiFi Module που χρησιμοποιεί εντολές AT. Δηλαδή, χρησιμοποιώντας τις σειριακές θύρες του Arduino μπορούν να σταλούν εντολές στο ESP8266 και στη συνέχεια να διαβαστούν οι απαντήσεις. Στον κώδικα έχουν οριστεί κάποιες σταθερές μεταβλητές που αφορούν τη σύνδεση στο δίκτυο, όπως είναι το αναγνωριστικό και ο κωδικός του τοπικού ασύρματου δικτύου που θα συνδεθεί η συσκευή, καθώς και η διεύθυνση, ο

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

τύπος και η μεταβλητή που θα έχει το ερώτημα με το οποίο θα σταλούν τα δεδομένα στον server.



Εικόνα 3.18: Μπλόκ διάγραμμα του κυκλώματος της κατασκευής

### 3.6 Προγραμματισμός του Arduino

Ο προγραμματισμός του Arduino έγινε χρησιμοποιώντας το επίσημο Arduino IDE που μπορούμε να βρούμε στη σελίδα του<sup>5</sup>. Επίσης, για να επιτευχθεί η ευκολότερη σειριακή επικοινωνία με τον ελεγκτή του optical mouse, έγινε χρήση μέρους της βιβλιοθήκης OptiMouse<sup>6</sup>. Ο κώδικας που αναπτύχθηκε για τον προγραμματισμό του Arduino χωρίζεται νοητά σε τρία μέρη.

Στο πρώτο μέρος εισάγονται οι απαραίτητες βιβλιοθήκες, ενώ ταυτόχρονα ορίζονται και αρχικοποιούνται οι μεταβλητές που χρησιμοποιούνται κατά τη διάρκεια της ρύθμισης του συστήματος ή/και κατά την εκτέλεση του προγράμματος.

Στο δεύτερο μέρος (setup function) λαμβάνει χώρα η αρχικοποίηση (αρχική ρύθμιση) του συστήματος. Πιο συγκεκριμένα, αρχικοποιούνται οι συνδέσεις του Arduino με τα επιμέρους συστήματα (modules) και στη συνέχεια η σύνδεση του ESP8266 με το τοπικό ασύρματο δίκτυο (WiFi).

Στο τρίτο μέρος (loop function), το οποίο επαναλαμβάνεται καθ' όλη την υπόλοιπη διάρκεια λειτουργίας του συστήματος, χρησιμοποιείται μία μεταβλητή γραμματοσειράς (String variable) η οποία φέρει τις τιμές της σχετικής μετατόπισης του οπτικού αισθητήρα στον οριζόντιο άξονα (x). Η τοποθέτηση του αισθητήρα πάνω στον κυκλικό δίσκο έχει γίνει με τέτοιο τρόπο ώστε να κινείται κυκλικά μαζί του, με αποτέλεσμα να μην υπάρχει μετατόπιση αυτού στον κάθετο άξονα (y) και να μην χρειάζεται η μέτρηση/ αποθήκευση των τιμών αυτών. Έπειτα από μετρήσεις βρέθηκε πως, σε κάθε πλήρη περιστροφή του βραχίονα, ο αισθητήρας μετράει 3600 διαφορετικά σημεία, τιμή η οποία χρησιμοποιήθηκε για τον υπολογισμό των μοιρών. Αφού υπολογιστούν οι τιμές των μοιρών της μετατόπισης, συνενώνονται στην μεταβλητή γραμματοσειράς, η οποία αποστέλλεται με αίτημα, μόλις το μέγεθός της υπερβεί το καθορισμένο ως μέγιστο μέγεθος για αυτή. Το αίτημα χρησιμοποιεί μέθοδο τύπου GET και η μεταφορά γίνεται με χρήση HTTP/TCP πρωτοκόλλων.

---

<sup>5</sup> <https://www.arduino.cc/en/Main/Software>

<sup>6</sup> <https://github.com/zapmaker/OptiMouse>

### 3.7 Δημιουργία σελίδας στον Server

Ο προγραμματισμός του apache server έχει γίνει σε PHP χρησιμοποιώντας το IDE PhpStorm της JetBrains με academic license. Έχει αναπτυχθεί ένα απλό framework το οποίο χρησιμοποιεί SPL autoloader για να φορτωθούν οι κλάσεις που βρίσκονται στα αρχεία. Το framework αποτελείται από δύο κλάσεις οι οποίες, εφόσον φορτωθεί, βοηθούν κατά την εκτέλεση της αποθήκευσης και της ανάκτησης των δεδομένων.

Τα δεδομένα αποθηκεύονται σε ένα αρχείο txt με την μορφή που λαμβάνονται από την κατασκευή, αφού προστεθεί στην αρχή της νέας γραμμής η σήμανση χρόνου (timestamp) της στιγμής που ελήφθησαν τα δεδομένα. Τη διαδικασία αυτή χειρίζεται το αρχείο κώδικα (script) “receive\_data.php” που βρίσκεται στον αρχικό φάκελο της εφαρμογής (root path) “gmeter”. Επομένως, το URL που στέλνονται τα δεδομένα δημιουργείται από:

- Την IP του εξυπηρετητή (ή το domain name, αν έχει οριστεί)
- Τη «διαδρομή» (path) του αρχείου που χειρίζεται την αποθήκευση (/gmeter/receive\_path.php)
- Την παράμετρο data που αποτελείται από γραμματοσειρά με τα δεδομένα προς αποθήκευση (?data=millisA\_degreesA;millisB\_degreesB)

Στον ίδιο φάκελο υπάρχει και το αρχείο “index.php” που απαντά στα ερωτήματα των περιηγητών (browsers) που θέλουν να πάρουν τα δεδομένα. Αρκεί να περιηγηθούν στην αρχική διαδρομή (path) της εφαρμογής (/gmeter), ώστε να πάρουν την απάντηση που περιέχει πληροφορίες για την τοποθεσία των απαραίτητων βιβλιοθηκών, καθώς κι ενός ακόμα αρχείου που θα αναλάβει να χειριστεί την ασύγχρονη κλήση για την ανάκτηση των δεδομένων. Η βιβλιοθήκη jQuery<sup>7</sup> βοηθά στην εύκολη υλοποίηση της ασύγχρονης κλήσης (αποστολής ερωτήματος) και ανάκτησης των δεδομένων. Η βιβλιοθήκη amCharts<sup>8</sup> βοηθά στην απεικόνιση των δεδομένων σε γράφημα, ενώ ταυτόχρονα δίνει στον χρήστη αρκετά εργαλεία για τη χρήση του γραφήματος όπως η απεικόνιση συγκεκριμένων περιοχών και η εξαγωγή/ αποθήκευση του γραφήματος.

---

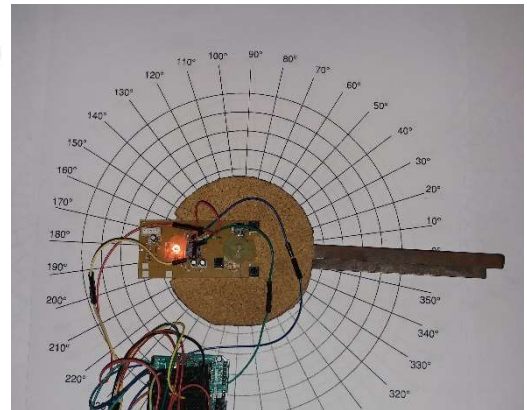
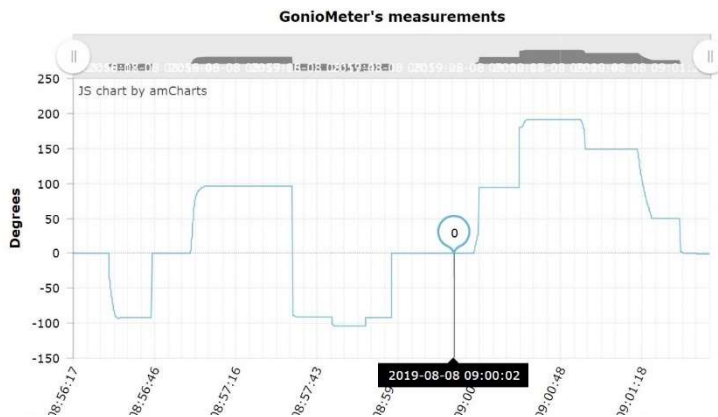
<sup>7</sup> <http://jquery.com>

<sup>8</sup> <https://www.amcharts.com>

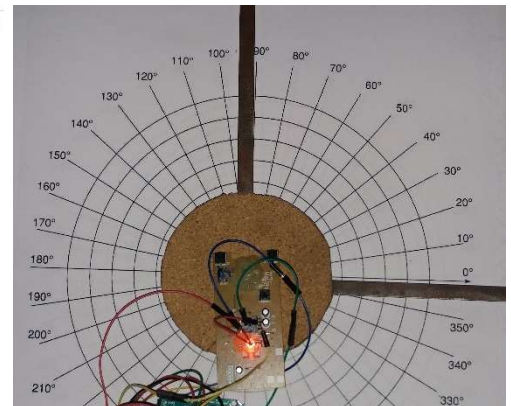
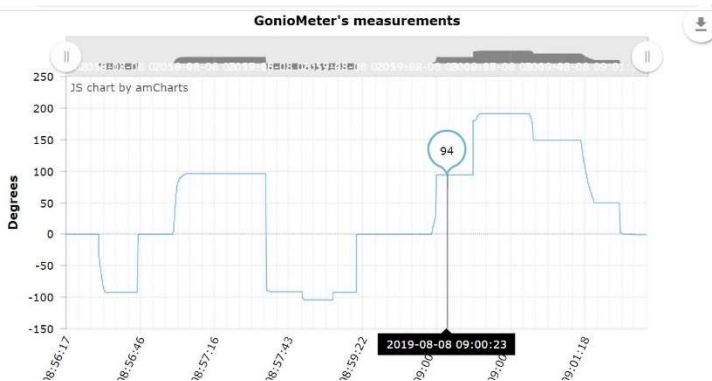


## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

Στο σημείο αυτό παρατίθενται εικόνες από τα γραφήματα που έχουν εξαχθεί από τη σελίδα στον Server και αντιπαρατίθενται με τις αντίστοιχες εικόνες από τις μετρήσεις με το γωνιόμετρο στις αντίστοιχες ενδεικτικές θέσεις.



Εικόνα 3.19: Σύγκριση χειροκίνητων μετρήσεων και απεικόνιση τους στο γράφημα στη θέση 0°



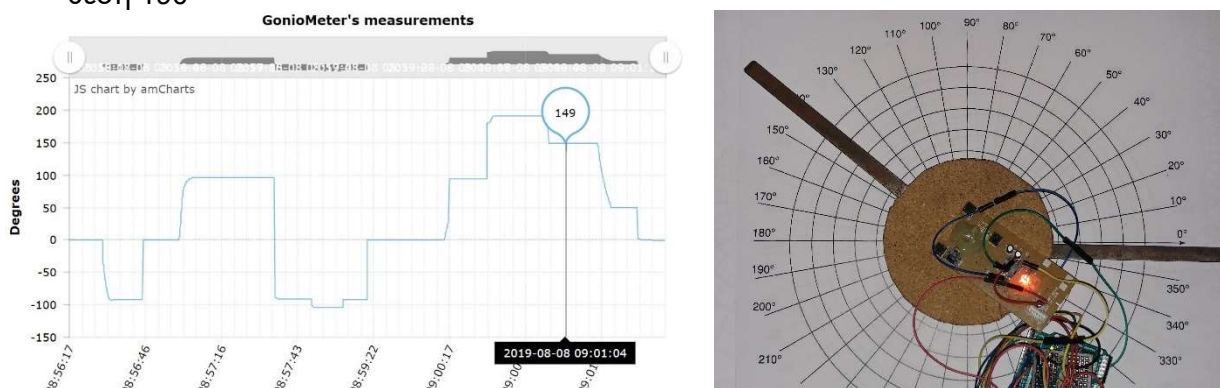
Εικόνα 3.20: Σύγκριση χειροκίνητων μετρήσεων και απεικόνιση τους στο γράφημα στη θέση 90°



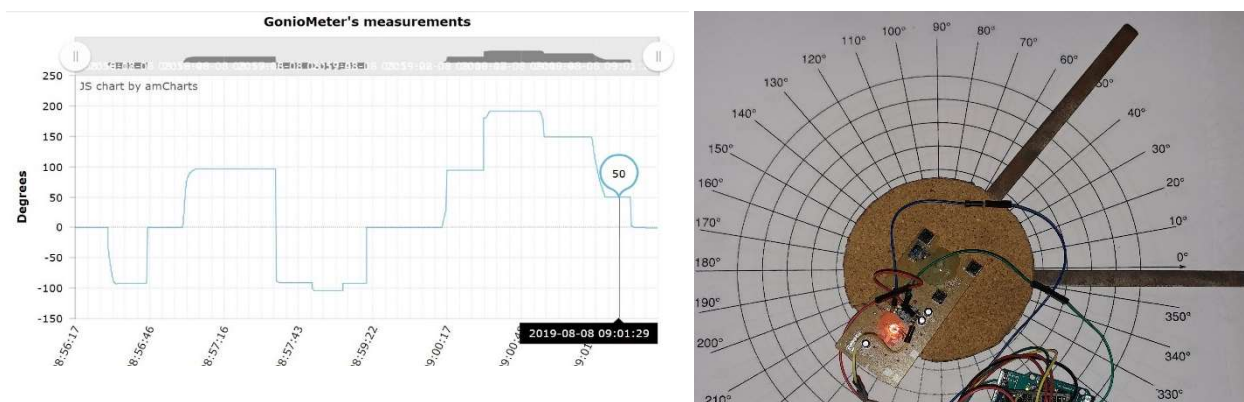
## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών



Εικόνα 3.21: Σύγκριση χειροκίνητων μετρήσεων και απεικόνιση τους στο γράφημα στη θέση 190°



Εικόνα 3.22: Σύγκριση χειροκίνητων μετρήσεων και απεικόνιση τους στο γράφημα στη θέση 150°



Εικόνα 3.23: Σύγκριση χειροκίνητων μετρήσεων και απεικόνιση τους στο γράφημα στη θέση 50°

#### 4. ΣΥΜΠΕΡΑΣΜΑΤΑ

Στόχος της παρούσας εργασίας ήταν η κατασκευή ενός ψηφιακού γωνιόμετρου, χρησιμοποιώντας τις σύγχρονες εξελίξεις και δυνατότητες της τεχνολογίας και πιο συγκεκριμένα των συστημάτων ανοιχτού κώδικα. Αρχικά, πραγματοποιήθηκε βιβλιογραφική ανασκόπηση προκειμένου να αποσαφηνιστούν οι τεχνολογίες που αξιοποιήθηκαν για την κατασκευή του οργάνου, αναδεικνύοντας τα θετικά και τα αρνητικά τους σημεία. Παράλληλα, μελετήθηκαν και παρουσιάστηκαν ορισμένες εφαρμογές που δημιουργήθηκαν, αξιοποιώντας τις εν λόγω πλατφόρμες- συστήματα ανοιχτού κώδικα, προσανατολισμένες στον τομέα της ιατρικής. Τελικά, αναλύθηκαν τα στάδια της κατασκευής τόσο των μηχανικών όσο και των ηλεκτρονικών μερών, καθώς και η μεταξύ τους σύνδεση, που οδήγησαν στην ολοκλήρωση του έργου.

Το κυριότερο θετικό σημείο της εν λόγω προσέγγισης είναι πως το όργανο που κατασκευάστηκε είναι αρκετά οικονομικό. Η προσιτή τιμή της κατασκευής του είχε τεθεί εξαρχής ως στόχος και βάσει αυτού έγιναν και οι αντίστοιχες επιλογές των υλικών και τεχνολογιών που χρησιμοποιήθηκαν. Με τον τρόπο αυτό, δίνεται η δυνατότητα να χρησιμοποιηθεί τόσο από τους επαγγελματίες του χώρου, όσο και από τους ίδιους τους ασθενείς, που βρίσκονται στο στάδιο αποκατάστασης κινητικών προβλημάτων, προκειμένου να ελέγχουν την πρόοδο τους. Επιπλέον, η επιλογή μιας πλατφόρμας ανοιχτού κώδικα, του Arduino, επιτρέπει τη δυνατότητα άμεσης εξέλιξης, σε κάποιον που επιθυμεί να συμβάλει και να βελτιώσει την υπάρχουσα κατασκευή, αλλά ακόμη και στην ταχύτερη επισκευή της σε περίπτωση που χρειαστεί.

Ένα, επίσης, από τα βασικότερα πλεονεκτήματα της κατασκευής είναι πως τα δεδομένα των μετρήσεων αποστέλλονται απευθείας στον server μέσω WiFi. Με τον τρόπο αυτό, επιτρέπεται η αναπαραγωγή και πρόσβαση σε δεδομένα παρελθοντικών μετρήσεων και μπορεί ο ενδιαφερόμενος να εστιάσει σε συγκεκριμένες χρονικές περιόδους ανά πάσα στιγμή. Επιπλέον, τα δεδομένα που συλλέγονται από τα δίκτυα αυτά μπορούν να αποθηκευτούν για ένα μεγάλο χρονικό διάστημα και να φανούν χρήσιμα εκτός από την πιο αποτελεσματική ανάλυση του ιστορικού κάθε ασθενούς και στην ιατρική έρευνα. Τέλος, η σελίδα περιήγησης είναι πολύ απλή στη χρήση της και

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

υπάρχει άμεση δυνατότητα εξαγωγής και αποθήκευσης των δεδομένων σε διάφορες μορφές αρχείων για περαιτέρω επεξεργασία.

Έχοντας παρουσιάσει τα κυριότερα θετικά σημεία της κατασκευής κρίνεται σκόπιμο να αναφερθούν ορισμένα σημεία που θα μπορούσαν να αποτελέσουν τμήματα περαιτέρω βελτίωσης. Αρχικά, τα μηχανικά μέρη του γωνιόμετρου θα μπορούσαν να κατασκευαστούν από διαφορετικό υλικό, όπως πλαστικό ή ανθρακόνημα,, έτσι ώστε να είναι πιο ελαφριά η κατασκευή στο σύνολο της και ταυτόχρονα πιο εύχρηστη. Επίσης, την ευχρηστία του γωνιόμετρου θα βελτίωνε η αντικατάσταση των ηλεκτρονικών μερών του με πιο ακριβά, μειώνοντας έτσι τον όγκο του. Στην παρούσα προσέγγιση, ωστόσο, έγιναν διαφορετικές επιλογές, καθώς θεωρήθηκε προτιμότερο να δοθεί έμφαση στη μείωση του κόστους, χωρίς, σαφώς, να επηρεάζεται σε μεγάλο βαθμό η ευχρηστία. Τέλος, ένα σημείο που θα μπορούσε να εξελιχθεί είναι ο τρόπος που το γωνιόμετρο επικοινωνεί με το WiFi. Στην παρούσα κατασκευή χρειάζεται σύνδεση με υπολογιστή που έχει εγκατεστημένο το Arduino IDE για να ρυθμιστεί η σύνδεση WiFi (SSID και password). Εναλλακτικά, θα μπορούσε να προστεθεί σχετική οθόνη στην πλατφόρμα του Arduino για να επιτευχθεί η προαναφερθείσα σύνδεση ή ακόμη να δημιουργηθεί σχετική εφαρμογή για smartphone, τόσο για την αμεσότερη επικοινωνία με το WiFi όσο και για τη διευκόλυνση του χρήστη.

## ΠΑΡΑΡΤΗΜΑ Α΄

Στο παράρτημα αυτό παρατίθεται ο κώδικας ανάπτυξης της παρούσας εφαρμογής.

### Κώδικας Arduino

```
#define BAUDRATE 57600

// Basic serial communication with ESP8266
// Uses serial monitor for communication with ESP8266
//
// Pins
// Arduino pin 7 (RX) to ESP8266 TX
// Arduino pin 6 to voltage divider then to ESP8266 RX
// Connect GND from the Arduino to GND on the ESP8266
// Pull ESP8266 CH_PD HIGH
//
// When a command is entered in to the serial monitor on the computer
// the Arduino will relay it to the ESP8266
//
#include <SoftwareSerial.h>
SoftwareSerial Serial2(7, 6); // RX | TX

#define WIFISSID "*****"
#define WIFIPASS "*****"

#define DESTINATION "83.212.105.76"
#define REQUEST_URL "GET http://83.212.105.76/gmeter/receive_data.php?"
//#define DESTINATION "192.168.2.2"
```

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
##define REQUEST_URL "GET http://192.168.2.2/gmeter/receive_data.php?"
```

```
#define PARAMETER "data"
```

```
#define TESTWIFI false
```

```
#define TESTOPTI false
```

```
// Local network: 440 seems ideal - 450 produces some errors
```

```
// Wide network: not more than 400
```

```
#define MAXDATALENGTH 400
```

```
// Local Networks: 15 seems OK for local networks even with errors - 20 seems ideal
```

```
// Wide Network: 50 seems OK
```

```
#define DELAYAFTERCONNECT 50
```

```
#define DELAYAFTERREQUEST 50
```

```
// 0 for best performance, 20 for debugging
```

```
#define DELAYPRINTRESPONSE 0
```

```
// Optical mouse's dependencies and initialization
```

```
#include <ADNS2083.h>
```

```
#define SCLK 2
```

```
#define SDIO 3
```

```
ADNS2083 Optical = ADNS2083(SCLK, SDIO);
```

```
// Initialization
```

```
int circle = 3600;
```

```
signed long x_init = 0;
```

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
signed long x = 0;
signed long ppd = 0;
signed long deg = 0;
int c = 0;
boolean start = false;

void setup() {

  // Serial object is being used for the USB connection
  Serial.begin(BAUDRATE);

  while (!Serial);
  Serial.print("Remember to set Both NL & CR and ");
  Serial.print(BAUDRATE);
  Serial.println(" baud in the serial monitor.");
  Serial.println("Initializing...");

  Serial.println("Configuring WiFi module...");

  // Serial2 is being used for the ESP8266 connection
  // Open serial communication to ESP8266 and wait for port to open
  Serial2.begin(BAUDRATE);
  String cmd_baudrate = "AT+CIOBAUD=";
  cmd_baudrate += BAUDRATE;
  Serial2.println(cmd_baudrate);
  while (!Serial2);

  while (Serial2.available() > 0)
    Serial2.read();
}
```

## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
// resetting ESP8266
Serial2.println("AT+RST");
delay(5000);
Serial2.println(cmd_baudrate);
delay(1000);

// connecting to wifi
boolean connected = false;

Serial2.println("AT+CWQAP");
delay(500);

String command = "AT+CWJAP=\"";
command += WIFISSID;
command += "\",\"";
command += WIFIPASS;
command += "\"";

for (int i = 0; i < 5; i++) {

    Serial2.println(command);
    delay(5000);
    //if (Serial2.find("OK")) {

        //connected = true;
        //break;
    //}

    connected = true;
```

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
}

if (!connected)
  while(1);

delay(5000);

Serial2.println("AT+CIFSR");
Serial.print("WLAN IP: ");
delay(1000);
while (Serial2.available())
  Serial.write(Serial2.read());

// set the single connection mode
Serial2.println("AT+CIPMUX=0");

Serial.println();
Serial.println("WiFi connection established");

// Initializing optical mouse
// Optical object is being used for the communication with the optical mouse
Optical.begin();
// String metrheis[100];
// for ( int i = 0; i<6; i++ )
// {

//int firstCharIndex = message.indexOf('-');
//int secondCharIndex = message.indexOf('-', firstCharIndex + 1);
//  Serial.write(message.charAt(secondCharIndex + 1));
```



Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
// Serial.write(message.charAt(secondCharIndex + 2));
//String metrshsh = String((message.charAt(secondCharIndex + 1) * 10) +
(message.charAt(secondCharIndex + 2)));
// metrshseis[i] = metrshsh;
// char message;
// if(Serial2.available()) message = Serial2.write('AT+CWLAP="AI-THINKER_FED217"');
//if(Serial2.available()){
// Serial2.write('AT+CWLAP="AI-THINKER_FED217"'); //}
// while(Serial2.available()){Serial.write(Serial2.read());}
// while(Serial2.available() > 0)
// {
// Serial.write(Serial2.read());
// }

// while (Serial2.available() > 0)
// Serial.write(Serial2.read());

// message = Serial2.read();
// Serial.write(message);
// Serial.write("\n");
// delay(1500);
// }
//Serial.print(metrshseis);

Serial.println("Setup finished");
Serial.println("Start exercising!");
}

void loop() {
```

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
/* TESTWIFI: Open serial to ESP8266 for testing */
if (TESTWIFI) {

  while(1) {

    while(Serial2.available()) Serial.write(Serial2.read());
    while(Serial.available()) Serial2.write(Serial.read());
  }
  return;
}

// Disconnecting all connections ESP8266
// Serial2.println("AT+CIPCLOSE=5");
// delay(500);

String degrees = "";

while(degrees.length() < MAXDATALENGTH) {

  if (x_init == 0) {
    x_init = Optical.dx();
    circle += x_init;
    ppd = circle / 360;
  }

  x += Optical.dx();
  if (c++ & 0x80) {

    deg = x / ppd;
```

## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
/* TESTOPTI: Review optical sensor's behavior */
if (TESTOPTI) {

    Serial.println();
    Serial.print(deg, DEC);
    Serial.print(" degrees - millis ");
    Serial.print(millis());
}

degrees += millis();
degrees += "_";
degrees += deg;
degrees += ",";
c = 0;
}
}

if (TESTOPTI) {
    return;
}

// Connecting ESP8266 to server
//AT+CIPSTART="TCP","83.212.105.76",80
String cmd_connect = "AT+CIPSTART=\\\"TCP\\\",\\\"";
cmd_connect += DESTINATION;
cmd_connect += "\\\",80";
Serial2.print(cmd_connect);
Serial2.println();
```

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
delay(DELAYAFTERCONNECT);

// Sending request to server
String request = REQUEST_URL;
request += PARAMETER;
request += "=";
request += degrees;

Serial2.print("AT+CIPSEND=");
Serial2.print(request.length()+4);
Serial2.println();
delay(DELAYAFTERREQUEST);
Serial2.print(request);
Serial2.println();
Serial2.println();
Serial.println("request sent!");

// Serial.println();
// Serial.print("Sending request: ");
// Serial.print(request);

while (Serial2.available()) {

    Serial.write(Serial2.read());
    delay(DELAYPRINTRESPONSE);
}

// Serial.println();
// Serial.println("=====");
```

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
//Serial2.println("AT+CIPCLOSE=5");  
// delay(500);  
}
```

## Κώδικας server (ανά αρχείο)

index.php

```
<?php
require_once __DIR__ . DIRECTORY_SEPARATOR . 'GMeter.php';
GMeter::initialize();

$html = '<!DOCTYPE html>';
$html .= '<html>';

$html .= '<head>';
$html .= '<title>GonioMeter</title>';

$html .= '<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>';
$html .= '<link rel="stylesheet"
href="https://ajax.googleapis.com/ajax/libs/jqueryui/1.12.1/themes/smoothness/jquery-ui.css">';
$html .= '<script src="https://ajax.googleapis.com/ajax/libs/jqueryui/1.12.1/jquery-ui.min.js"></script>';
$html .= '<script src="https://www.amcharts.com/lib/3/amcharts.js"></script>';
$html .= '<script src="https://www.amcharts.com/lib/3/serial.js"></script>';
$html .= '<script src="https://www.amcharts.com/lib/3/plugins/export/export.min.js"></script>';
$html .= '<link rel="stylesheet" href="https://www.amcharts.com/lib/3/plugins/export/export.css" type="text/css"
media="all" />';
$html .= '<script src="https://www.amcharts.com/lib/3/themes/light.js"></script>';

$html .= '<script type="text/javascript" src="/gmeter/js/script.js? . time() . ""></script>';

$html .= '</head>';
$html .= '<body>';

$html .= '<div id="chartdiv" style="width: 100%; height: 500px;"></div>';

$html .= '<div>';
$html .= 'Refresh every <span id="refreshValue"></span> seconds';
$html .= '</div>';
$html .= '<div class="slidecontainer">';
$html .= '<input type="range" min="3" max="10" value="5" class="slider" id="refreshTimeoutSlider">';
$html .= '</div>';
$html .= '<div><input type="button" id="reset" value="Delete data" onclick="resetData();" /></div>';

$html .= '</body>';
$html .= '</html>';
echo $html;
```

## script.js

```
var chart, chartData,
    intervalHandler,
    slider, sliderValueSpan;

$(document).ready(function () {

    getData();
    intervalHandler = setInterval(getData, 5000);
    slider = document.getElementById("refreshTimeoutSlider");
    sliderValueSpan = document.getElementById("refreshValue");
    slider.value = 5;
    sliderValueSpan.innerHTML = 5;

    slider.oninput = function (ev) {
        clearInterval(intervalHandler);
        intervalHandler = setInterval(getData, this.value*1000);
        sliderValueSpan.innerHTML = this.value;
    }
});

function getData() {

    $.ajax({
        type: 'GET',
        dataType: 'JSON',
        url: '/gmeter/data.json.php',
        data: {},
        success: function(response) {

            if (response.type !== 'success')
                return;
            chartData = response.data;
            initializeChart();
        }
    });
}

function initializeChart() {

    chart = AmCharts.makeChart("chartdiv", {
        "type": "serial",
        "theme": "light",
        "titles": [
            {
                "text": "GonioMeter\'s measurements"
            }
        ],
        "marginRight": 80,
        "autoMarginOffset": 20,
        "marginTop": 7,
        "dataProvider": chartData,
        "valueAxes": [{
            "title": "Degrees",
            "axisAlpha": 0.2,
            "dashLength": 1,
```

```
    "position": "left"
  },
  "mouseWheelZoomEnabled": true,
  "graphs": [{
    "id": "g1",
    "balloonText": "[[value]]",
    "bullet": "round",
    "bulletBorderAlpha": 1,
    "bulletColor": "#FFFFFF",
    "hideBulletsCount": 50,
    "title": "red line",
    "valueField": "value",
    "useLineColorForBulletBorder": true,
    "balloon":{
      "drop":true
    }
  }],
  "chartScrollbar": {
    "autoGridCount": true,
    "graph": "g1",
    "scrollbarHeight": 40
  },
  "chartCursor": {
    "limitToGraph": "g1"
  },
  "categoryField": "time",
  "categoryAxis": {
    "title": "Date & time",
    // "parseDates": true,
    "axisColor": "#DADADA",
    "dashLength": 0.4,
    "minorGridEnabled": true,
    "labelRotation": 65
  },
  "export": {
    "enabled": true
  }
});
}

// this method is called when chart is first initialized as we listen for "rendered" event
function zoomChart() {
  // different zoom methods can be used - zoomToIndexes, zoomToDates, zoomToCategoryValues
  chart.zoomToIndexes(chartData.length - 40, chartData.length - 1);
}

function resetData() {
  $.ajax({
    type: 'POST',
    dataType: 'JSON',
    url: '/gmeter/reset_data.php',
    data: {},
    success: function(response) {

      if (response.type !== 'success')
        return;
      chartData = response.data;
    }
  });
}
```



## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
        initializeChart();
    }
});
}
```

### data.json.php

```
<?php

require_once __DIR__ . DIRECTORY_SEPARATOR . 'GMeter.php';
GMeter::initialize();

$content = file_get_contents(GMeterConfiguration::$data_file);
$rows = explode(PHP_EOL, $content);

$response = [
    'type' => 'error',
    'message' => 'An error occurred',
    'data' => []
];

foreach ($rows as $row) {

    $row_exploded = explode(
        GMeterConfiguration::$time_data_separator,
        $row
    );

    if (sizeof($row_exploded) != 2)
        continue;

    $timestamp = $row_exploded[0];
    $data_set =
        explode(
            GMeterConfiguration::$data_separator,
            $row_exploded[1]
        );

    $data_set = array_reverse($data_set);

    for ($i = sizeof($data_set); $i >= 0; $i--) {

        if (!isset($data_set[$i]))
            continue;

        $sup_time_value_exploded =
            explode(
                GMeterConfiguration::$sup_time_ms_measure_separator,
                $data_set[$i]
            );

        if (sizeof($sup_time_value_exploded) != 2)
            continue;

        if (!is_numeric($sup_time_value_exploded[0]) ||
            !is_numeric($sup_time_value_exploded[1]))
```

## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
continue;

$time = isset($next_up_time_ms) ?
    $timestamp * 1000 :
    $time - ($next_up_time_ms - $up_time_value_exploded[0]);

$next_up_time_ms = $up_time_value_exploded[0];

$response['data'][] = [
    'time' =>
        GMeterHelper::timestampToDateAndTimeString(
            $time/1000
        ),
    'value' => $up_time_value_exploded[1]
];
}

unset($next_up_time_ms);
}

$response['type'] = 'success';
$response['message'] = 'OK';
echo json_encode($response);
return;
```

### receive\_data.php

```
<?php

require_once __DIR__ . DIRECTORY_SEPARATOR . 'GMeter.php';
GMeter::initialize();

if (!isset($_GET['data']))
    return;

$data = $_GET['data'];
$content = GMeterConfiguration::getDataHead() . $data;

file_put_contents(GMeterConfiguration::$data_file, $content, FILE_APPEND);

echo 'aaaaaaaaa';
```

### GMeter.php

```
<?php

class GMeter {

    protected static $class_files_array;

    /**
     * Initialize the framework
     */
    public static function initialize() {
```

## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

```
//find the framework class files and load it in an array
self::$class_files_array =
    self::__find_the_framework_class_files();
spl_autoload_register('GMeter::autoloader');
}

/**
 * Finds the required class files and paths for the application framework
 * @return multitype:string
 */
private static function __find_the_framework_class_files() {

    /* @var $file SplFileInfo */
    foreach (
        new RecursiveIteratorIterator(
            new RecursiveDirectoryIterator(
                __DIR__,
                RecursiveDirectoryIterator::SKIP_DOTS
            )
        ) as $file
    ){
        if ($file->getExtension() == 'php') {

            $filename_without_extension =
                explode(".php", $file->getFilename());
            $filename_without_extension =
                $filename_without_extension[0];
            $class_files_array[$filename_without_extension] =
                $file->getRealPath();
        }
    }

    return $class_files_array;
}

/**
 * Includes the php file for this class
 * @param string $class_name
 * @return boolean
 */
public static function autoloader($class_name) {

    //do this only for gmeter classes
    if (strpos(strtolower($class_name), 'gmeter') !== false) {

        if (self::$class_files_array[$class_name]) {

            include self::$class_files_array[$class_name];
            return true;
        }
    }

    return false;
}
}
```

## GMeterHelper.php

```
<?php
class GMeterHelper {

    public static function precho($anything) {

        echo "<pre>";
        print_r($anything);
        echo "</pre>";

    }

    public static function dateTimeStringToTimestamp($date_time_string) {

        $dateTime = new DateTime($date_time_string);
        return $dateTime->format('U');

    }

    public static function timestampToDateAndTimeString($timestamp) {

        return date('Y-m-d H:i:s', $timestamp);

    }

}
```

## GMeterConfiguration.php

```
<?php
class GMeterConfiguration {

    public static $base_path = __DIR__ . DIRECTORY_SEPARATOR .
        '..' . DIRECTORY_SEPARATOR;

    public static $data_file = __DIR__ . DIRECTORY_SEPARATOR .
        '..' . DIRECTORY_SEPARATOR .
        'files' . DIRECTORY_SEPARATOR .
        'data.txt';

    public static $time_data_separator = ':';
    public static $data_separator = ',';
    public static $up_time_ms_measure_separator = '_';
    public static $filename_time_separator = '_';

    public static function getDataHead() {

        return PHP_EOL . time() . self::$time_data_separator;

    }

}
```

## Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

### Sample of the data.txt file

```
1519050132:75918_141;75978_137;76040_132;76100_128;76161_123;76221_118;76282_113;76343_109;76403_105;76464_101;76525_98;76585_95;76646_92;76706_89;76767_86;76828_84;76889_81;76949_78;77009_76;77070_73;77131_71;77192_68;77252_66;77313_64;77374_63;77434_62;77495_62;77555_63;77616_65;77677_68;77737_71;77798_74;77859_78;77920_82;77980_86;78041_90;78101_94;78162_98;78223_103;78283_107;78344_112;78405_117;78466_121;78526_125;78586_129;78647_134;78708_137;78769_141;78829_145;78889_148;78951_150;79011_152;79072_154;1519050145:89401_119;89461_112;89522_104;89583_97;89644_90;89704_84;89764_79;89825_74;89886_70;89947_67;90007_64;90067_60;90129_56;90189_52;90250_49;90310_46;90371_45;90432_45;90492_47;90553_50;90613_54;90674_58;90735_63;90796_68;90856_72;90916_78;90978_82;91038_87;91099_93;91159_97;91219_103;91281_108;91341_114;91402_120;91462_123;91524_126;91584_128;91644_128;91705_128;91765_128;91827_128;91887_128;91948_128;92008_128;92069_128;92130_128;92190_128;92251_128;92312_128;92372_128;92433_128;92493_128;92555_128;
```

## ΒΙΒΛΙΟΓΡΑΦΙΑ

- Μ. Κώστα, Σ. Νάκα, “ Ασύρματα δίκτυα”, Πτυχιακή εργασία, Τμήμα Τηλεπληροφορικής και Διοίκησης, ΤΕΙ Ηπείρου, 2006
- Σ. Λασκαρίδης, “Σχεδιασμός & ανάπτυξη πλατφόρμας αποκατάστασης άνω άκρων ασθενών”, Πτυχιακή εργασία, Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστικών Συστημάτων, ΑΕΙ Πειραιά Τ.Τ, 2016
- Ι. Μελετόπουλου, “Τεχνικές σύνταξης πτυχιακής εργασίας”, Πτυχιακή εργασία, Τμήμα Ηλεκτρονικών Υπολογιστών Συστημάτων, ΤΕΙ Πειραιά, 2004
- Κ. Σιάχρας, “ Μέτρηση τάσης δικτύου με Arduino και εμφάνιση της σε android συσκευή μέσω bluetooth”, Πτυχιακή εργασία, Τμήμα Μηχανικών Ηλεκτρονικών Υπολογιστικών Συστημάτων, ΑΕΙ Πειραιά Τ.Τ, 2014
- Π. Τσιμέας, Γ. Μήτρου, Γ. Κουτεντάκης, Χ. Καρατζαφέρη, “ Μέτρηση και αξιολόγηση της ευκινησίας”, στο Χ. Καρατζαφέρη (επ.) Εγχειρίδιο για την σωματική αξιολόγηση αθλητών: δοκιμασίες εργαστηρίου και πεδίου για την επιστημονική υποστήριξη του αγωνιστικού αθλητισμού, Κάλλιπος, pp. 90-132, 2015
- Μ. Jansen, “An optical mouse sensor as an extension for lego NXT”, Report technology masterclass, 2008
- Α. Η. Kioumars, L. Tang, “Wireless network for health monitoring: heart rate and temperature sensor”, Fifth International Conference on Sensing Technology, pp. 362-369, 2011
- Ν. Lethaby, “Wireless connectivity for the Internet of Things: One size does not fit all”, Texas Instruments, 2017

- C.K. Lim, Z. Luo, IM Chen, S.H. Yeo, “A low cost wearable optical-based goniometer for human joint monitoring”, *Frontiers of Mechanical Engineering*, vol. 6, pp. 13-22, 2011
- U. Minoni, A. Signorini, “Low-cost optical motion sensors: An experimental characterization”, *Sensors and Actuators*, vol. 128, pp. 402-408, 2006
- T.W. Ng, “The optical mouse as a two-dimensional displacement sensor”, *Sensors and Actuators*, vol. 107, pp. 21-25, 2003
- G. Niezen, P. Eslambolchilar, H. Thimbleby, “Open-source hardware for medical devices”, *BMJ Innovations*, vol. 2, pp. 78-83, 2016
- A. Pantelopoulos, N.G. Bourbakis, “A Survey on Wearable Sensor-Based Systems for Health Monitoring and Prognosis”, *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, vol. 40, pp. 1-12, 2010
- Y. Ren, N. Lyu, “A Pulse Measurement and Data Management System Based on Arduino Platform and Android Device”, *Proceedings of 2016 IEEE 13th International Conference on Networking, Sensing, and Control Mexico City, Mexico, 2016*
- K. Sohrawy, D. Minoli, T. Znati, “Wireless Sensor Networks: Technology, Protocols, and Applications”, *John Wiley & Sons*, 2007
- P. Teikari, R.P. Najjar, H. Malkki, et al. “An inexpensive Arduino-based LED stimulator system for vision research”, *Journal of Neuroscience Methods*, vol. 211, pp. 227-236, 2012

Συσκευή μέτρησης γωνίας για την αποκατάσταση της κίνησης ασθενών

T. Thaker, “ESP8266 based Implementation of Wireless Sensor Network with Linux Based Web-Server”, Symposium on Colossal Data Analysis and Networking, 2016

SN. Yu, JC. Cheng, “A Wireless Physiological Signal Monitoring System with Integrated Bluetooth and WiFi Technologies”, IEEE Engineering in Medicine and Biology 27th Annual Conference, 2006

ESP8266 Technical Reference, 2017 Διαθέσιμο στο:

[https://www.espressif.com/sites/default/files/documentation/esp8266-technical\\_reference\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp8266-technical_reference_en.pdf)