



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΒΙΟΙΑΤΡΙΚΗΣ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ ΜΕ ΘΕΜΑ:

Κατασκευή φορητής συσκευής καταγραφής ζωτικών σημείων

Construction of a portable device for recording vital signs

Ον/υμο Σπουδαστή: Νασόπουλος Απόστολος Α.Μ.:16069

Επιβλέπων Καθηγητής : ΠΑΝΤΕΛΕΗΜΩΝ ΑΣΒΕΣΤΑΣ

ΑΘΗΝΑ 2022

Η Τριμελής Εξεταστική Επιτροπή

Ο Επιβλέπων Καθηγητής

Παντελεήμων Ασβεστάς

Αναπληρωτής Καθηγητής

[ΨΗΦΙΑΚΗ ΥΠΟΓΡΑΦΗ]

Δημήτριος Γκλώτσος

Αναπληρωτής Καθηγητής

[ΨΗΦΙΑΚΗ ΥΠΟΓΡΑΦΗ]

Σπυρίδων Κωστόπουλος

Αναπληρωτής Καθηγητής

[ΨΗΦΙΑΚΗ ΥΠΟΓΡΑΦΗ]

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Ο υπογράφων Νασόπουλος Απόστολος του Νικολάου, με αριθμό μητρώου 48016069 φοιτητής του Τμήματος Μηχανικών Βιοϊατρικής Τεχνολογίας της Σχολής Μηχανικών του Πανεπιστημίου Δυτικής Αττικής, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της διπλωματικής εργασίας και 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».

Ημερομηνία

18/07/2022

Ο/Η Δηλών/ούσα



ΠΕΡΙΛΗΨΗ

Η εκπόνηση της παρούσας διπλωματικής εργασίας έγινε υπό την επίβλεψη και τη καθοδήγηση του αναπληρωτή καθηγητή κ. Ασβεστά Παντελεήμωνα, στα πλαίσια της ολοκλήρωσης και απόκτησης του προπτυχιακού διπλώματος Μηχανικών Βιοιατρικής Τεχνολογίας του Πανεπιστημίου Δυτικής Αττικής κατά το ακαδημαϊκό έτος 2021-2022.

Σκοπός της παρούσας πτυχιακής εργασίας είναι η παρουσίαση της μελέτης, σχεδίασης και κατασκευής μίας φορητής συσκευής, με τη χρήση του μικροελεγκτή ESP8266 και την παρουσίαση των δεδομένων του σε πραγματικό χρόνο, η οποία θα καταγράφει μετρήσεις των πιο βασικών λειτουργιών του σώματος. Συγκεκριμένα, θα καταγράφονται η θερμοκρασία του σώματος και ο καρδιακός ρυθμός με χρήση ηλεκτροδίων κατά Einthoven. Η συσκευή παρέχει τη δυνατότητα απομακρυσμένης αποθήκευσης σε βάση δεδομένων. Με τη συσκευή αυτή, παρακολουθούνται διαρκώς οι μεταβολές της θερμοκρασίας του σώματός του ατόμου αλλά και οι καρδιακοί παλμοί προς αποφυγή πυρετού, καρδιακών προσβολών, εγκεφαλικών και λοιπών νοσημάτων που μπορούμε να αντιληφθούμε από αυτές τις μεταβολές, χάρη στους αισθητήρες cardiac pulse & temperature sensor.

Η εργασία αυτή παρουσιάζει την ακολουθία σκέψης μέσω των οποίων έγινε η επιλογή των ηλεκτρονικών διατάξεων, η ανάλυση του προβλήματος, η προσέγγιση της βέλτιστης λύσης και τα αποτελέσματα που εξάχθηκαν με τις μεθοδολογίες που χρησιμοποιήθηκαν.

Αυτό το συνολικό project υλοποιήθηκε κατόπιν έρευνας αντίστοιχων εφαρμογών προκειμένου να παρθεί ένα feedback και να παραχθεί κάτι καινοτόμο στον κόσμο της Ιατρικής Τεχνολογίας.

Συνεπώς η διπλωματική εργασία παρουσιάζει ένα προϊόν το οποίο ανιχνεύει, μελετά και προλαμβάνει θέματα υγείας του ασθενούς, προσφέροντάς του επομένως έναν περισσότερο ποιοτικό χρόνο ζωής.

ABSTRACT

The elaboration of the present thesis was done under the supervision and guidance of the associate professor Mr. Asvestas Panteleimonas, in the context of the completion and acquisition of the undergraduate degree in Biomedical Engineering of the University of West Attica during the academic year 2021-2022.

The purpose of this thesis is to present the study, design and construction of a portable device, using the ESP8266 microcontroller and to present its data in real time, which will record measurements of the most basic body functions (vital signs). Specifically, body temperature and heart rate will be recorded using Einthoven electrodes. The device provides the possibility of remote storage in a database. With this device, the changes of the individual's body temperature are constantly monitored, as well as the heartbeats to avoid fever, heart attacks, strokes and other diseases that we can perceive from these changes, thanks to the cardiac pulse & temperature sensor.

This work presents the thought sequence through which the electronic devices were selected, the problem analysis, the optimal solution approach and the results obtained with the methodologies used.

This overall project was implemented after researching relevant applications in order to get feedback and produce something innovative in the world of Medical Technology

The thesis therefore presents a product that detects, studies and prevents health issues of the patient, thus offering him a better quality of life.

ΕΥΧΑΡΙΣΤΙΕΣ – ΑΦΙΕΡΩΣΕΙΣ

Στο σημείο αυτό θα ήθελα να δηλώσω την ευγνωμοσύνη μου και να ευχαριστήσω τους καθηγητές μου για την αμέριστη υποστήριξη και συμπαράσταση που μου έδειξαν με κάθε τρόπο καθ' όλη τη διάρκεια των σπουδών μου.

Στην προσπάθεια αυτή συνέβαλαν πολλοί άνθρωποι με την πολύτιμη βοήθειά τους και την υποστήριξή τους.

Αρχικά, θα ήθελα να ευχαριστήσω θερμά τον καθηγητή κ. Ασβεστά Παντελεήμονα, ο οποίος μου έδωσε τη δυνατότητα να ασχοληθώ με το συγκεκριμένο θέμα και μου παρείχε υποστήριξη και καθοδήγηση όλο αυτό το διάστημα. Επίσης, θα ήθελα να ευχαριστήσω τους συμφοιτητές μου που με βοήθησαν να ξεπεράσω όποια προβλήματα παρουσιάστηκαν κατά τη διάρκεια της εργασίας.

Τέλος, ένα μεγάλο ευχαριστώ στην οικογένεια και τους φίλους μου για την υποστήριξή τους κατά τη διάρκεια των σπουδών μου.

Ευελπιστώ οι ιδέες και ο σκοπός που παρατίθενται στο παρόν έγγραφο εν συνεχεία να αποτελέσουν κίνητρο για περαιτέρω ανάλυση και διερεύνηση του χώρου των Ιατροτεχνολογικών εφαρμογών προκειμένου να βοηθηθούν άτομα που έχουν πραγματικά ανάγκη τέτοιου είδους υπηρεσιών.

Νασόπουλος Απόστολος

Αθήνα 2022

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΠΕΡΙΛΗΨΗ	5
ABSTRACT	6
ΕΥΧΑΡΙΣΤΙΕΣ – ΑΦΙΕΡΩΣΕΙΣ	7
ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ	8
ΛΙΣΤΑ ΔΙΑΓΡΑΜΜΑΤΩΝ	10
ΛΙΣΤΑ ΕΙΚΩΝΩΝ	11
ΛΙΣΤΑ ΣΧΗΜΑΤΩΝ	13
ΛΙΣΤΑ ΠΙΝΑΚΩΝ	14
ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ	15
ΚΕΦΑΛΑΙΟ 2. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ	16
2.1 Η έννοια της θερμοκρασίας	16
2.1.1 Γενικά για την θερμοκρασία	16
2.1.2 Σχετική θερμοκρασία	17
2.1.3 Απόλυτη θερμοκρασία	17
2.1.4 Εύρος θερμοκρασιών του ανθρώπινου σώματος	17
2.2 Η λειτουργία της καρδιάς	19
2.2.1 Ηλεκτρικά σήματα από το ανθρώπινο σώμα	19
2.2.2 Η λειτουργία της καρδιάς ως ηλεκτρικό σήμα	21
2.2.3 Ανάλυση της μορφής του καρδιογραφήματος	21
2.2.4 Ag/AgCl ηλεκτρόδια (υγρά)	23
ΚΕΦΑΛΑΙΟ 3. ΥΛΙΚΟ ΜΕΡΟΣ ΣΥΣΤΗΜΑΤΟΣ (HARDWARE)	24
3.1 Μικροελεγκτής NodeMCU ESP8266	24
3.2 AD8232 Monitor καρδιακού παλμού μίας απαγωγής	29
3.3 CD4051 Αναλογικός πολυπλέκτης 8 καναλιών	32
3.4 NTC Thermistor 10kΩ 3977K	34
3.5 OPA2340 Rail-to-Rail ενισχυτής	36
3.6 Διαιρέτης TLE2426	38
ΚΕΦΑΛΑΙΟ 4. ΠΛΑΤΦΟΡΜΕΣ ΑΝΑΠΤΥΞΗΣ ΛΟΓΙΣΜΙΚΟΥ (SOFTWARE)	40
4.1 Παρουσίαση πακέτων λογισμικού	40

4.1.1 Arduino IDE	40
4.1.2 Notepad ++	42
4.1.3 XAMPP	43
4.1.4 phpMyAdmin	44
4.2 Γλώσσες Προγραμματισμού & Server	45
4.2.1 HTML	45
4.2.2 CSS	47
4.2.3 PHP	48
4.2.4 MySQL.....	48
4.2.5 JavaScript	49
4.2.6 APACHE HTTP WEB SERVER	49
ΚΕΦΑΛΑΙΟ 5. ΚΑΤΑΣΚΕΥΑΣΤΙΚΟ ΜΕΡΟΣ ΣΥΣΚΕΥΗΣ.....	50
5.1 Κατασκευή Θερμομέτρου	50
5.2 Κατασκευή Καρδιογράφου	55
5.3 Ενσωμάτωση θερμομέτρου και ΗΚΓ με την χρήση του πολυπλέκτη	62
5.4 Σύνδεση συσκευής στο διαδίκτυο μέσω Wi-Fi	68
5.5 Παρουσίαση Συσκευής	77
ΑΞΙΟΛΟΓΗΣΗ ΜΕΘΟΔΟΛΟΓΙΑΣ.....	86
ΠΙΘΑΝΕΣ ΜΕΛΛΟΝΤΙΚΕΣ ΥΛΟΠΟΙΗΣΕΙΣ.....	87
ΣΥΜΠΕΡΑΣΜΑΤΑ	88
ΒΙΒΛΙΟΓΡΑΦΙΑ	89

ΛΙΣΤΑ ΔΙΑΓΡΑΜΜΑΤΩΝ

Διάγραμμα 2-1: Η θερμοκρασία του ανθρώπινου σώματος σε ένα 24-ωρο.....σελ.18	σελ.18
Διάγραμμα 2-2: Διάγραμμα ροής διαδικασίας λήψης και επεξεργασίας ενός βιοσήματος..... σελ.21	σελ.21
Διάγραμμα 3-1: Το μπλοκ διάγραμμα του ESP-12E.....σελ.25	σελ.25
Διάγραμμα 3-2: Το διάγραμμα των Pinout του ESP8266.....σελ.27	σελ.27
Διάγραμμα 3-3: Το λειτουργικό μπλοκ διάγραμμα της πλακέτας AD8232.....σελ.30	σελ.30
Διάγραμμα 3-4: Το διάγραμμα των Pin του CD4051.....σελ.32	σελ.32
Διάγραμμα 3-5: Η καμπύλη αλλαγής της τιμής αντίστασης του θερμίστορ 10kΩ σε σχέση με την θερμοκρασία..... σελ.35	σελ.35
Διάγραμμα 3-6: Το διάγραμμα των Pin του OPA2340.....σελ.36	σελ.36
Διάγραμμα 3-7: Το μπλοκ διάγραμμα του OPA2340.....σελ.37	σελ.37
Διάγραμμα 3-8: Το διάγραμμα των Pin του TLE2426.....σελ.38	σελ.38
Διάγραμμα 3-9: Το μπλοκ διάγραμμα του TLE2426.....σελ.39	σελ.39
Διάγραμμα 5-1: Η συνδεσμολογία των εξαρτημάτων του θερμομέτρου και του ESP8266.....σελ.52	σελ.52
Διάγραμμα 5-2: Η συνδεσμολογία του AD8232 με το ESP8266.....σελ.55	σελ.55
Διάγραμμα 5-3: Το καρδιακό σήμα του ΗΚΓ.....σελ.57	σελ.57
Διάγραμμα 5-4: Η συνδεσμολογία της συσκευής.....σελ.62	σελ.62
Διάγραμμα 5-5: Η γραφική απεικόνιση της συσκευής στο Fritzing.....σελ.77	σελ.77

ΛΙΣΤΑ ΕΙΚΩΝΩΝ

Εικόνα 2-1: Διάφορα είδη θερμομέτρων.....	σελ.18
Εικόνα 2-2: Θερμόμετρο χώρου.....	σελ.18
Εικόνα 3-1: Το NodeMCU ESP8266 της Espressif.....	σελ.24
Εικόνα 3-2: Το ESP-12E της Espressif.....	σελ.25
Εικόνα 3-3: Το σχηματικό του NodeMCU ESP8266.....	σελ.28
Εικόνα 3-4: Η πλακέτα του AS8232.....	σελ.29
Εικόνα 3-5: Τυπική τοποθέτηση αισθητήρων του AD8232.....	σελ.31
Εικόνα 3-6: Ο πολυπλέκτης CD4051.....	σελ.32
Εικόνα 3-7: Ο NTC θερμίστορ 10kΩ.....	σελ.34
Εικόνα 3-8: Μαθηματική εξίσωση παραμέτρων Beta.....	σελ.34
Εικόνα 3-9: Ο ενισχυτής OPA2340.....	σελ.36
Εικόνα 3-10: Ο διαιρέτης TLE2426.....	σελ.38
Εικόνα 4-1: Το περιβάλλον του Arduino IDE.....	σελ.40
Εικόνα 4-2: Το περιβάλλον του Notepad ++.....	σελ.42
Εικόνα 4-3: Το περιβάλλον του XAMPP.....	σελ.43
Εικόνα 4-4: Το λογότυπο της HTML.....	σελ.45
Εικόνα 4-5: Παράδειγμα σύνταξης HTML.....	σελ.46
Εικόνα 4-6: Παράδειγμα σύνταξης CSS.....	σελ.47
Εικόνα 4-7: Το λογότυπο του PHP.....	σελ.48
Εικόνα 4-8: Το λογότυπο της MySQL.....	σελ.48
Εικόνα 5-1: Αποτελέσματα μετρήσεων θερμομέτρου.....	σελ.54
Εικόνα 5-2: Το καλώδιο τριών ηλεκτροδίων για σύνδεση με την πλακέτα AD8232.....	σελ.56
Εικόνα 5-3: Αποτέλεσμα μέτρησης του ΗΚΓ.....	σελ.57
Εικόνα 5-4: Αποτέλεσμα μέτρησης της συσκευής.....	σελ.64
Εικόνα 5-5: Ενεργοποίηση του Apache server και της MySQL.....	σελ.69
Εικόνα 5-6: Εμφάνιση της Ιστοσελίδας.....	σελ.76
Εικόνα 5-7: Η τελική μορφή της συσκευής.....	σελ.77

Εικόνα 5-8: Το κυκλωματικό μέρος του θερμομέτρου.....σελ.78
Εικόνα 5-9: Το κυκλωματικό μέρος του ΗΚΓ & του πολυπλέκτη.....σελ.78
Εικόνα 5-10: Αναμμένο LED έπειτα από σύνδεση στο Wi-Fi.....σελ.79
Εικόνα 5-11: Μέτρηση ΗΚΓ με την χρήση προσομοιωτή.....σελ.79
Εικόνα 5-12: Αποτελέσματα μετρήσεων με τον προσομοιωτή.....σελ.80
Εικόνα 5-13: Αποτέλεσμα μέτρησης θερμοκρασίας δαχτύλου με ηλεκτρονικό θερμόμετρο.....σελ.81
Εικόνα 5-14: Αποτέλεσμα μέτρησης θερμοκρασίας δαχτύλου με υδραργυρικό θερμόμετρο.....σελ.81
Εικόνα 5-15: Τοποθέτηση των ηλεκτροδίων στο ανθρώπινο σώμα.....σελ.81
Εικόνα 5-16: Αποτελέσματα μετρήσεων με τον προσομοιωτή.....σελ.82

ΛΙΣΤΑ ΣΧΗΜΑΤΩΝ

Σχήμα 2-1: Η ανατομία της καρδιάς.....	σελ.19
Σχήμα 2-2: Γράφημα δύο παλμών μιας καρδιάς.....	σελ.20
Σχήμα 2-3: Τοποθέτηση ηλεκτροδίων στα άκρα του σώματος.....	σελ.22
Σχήμα 2-4: Τοποθέτηση ηλεκτροδίων στο στήρνο.....	σελ.22
Σχήμα 2.5: Σχηματική αναπαράσταση ενός κανονικού ηλεκτροκαρδιογραφήματος.....	σελ.22
Σχήμα 2.6: Υγρό ηλεκτρόδιο Ag/AgCl.....	σελ.23
Σχήμα 3-1: Το διάγραμμα των pin της πλακέτας AD8232.....	σελ.30
Σχήμα 3-2: Το διάγραμμα της πλακέτας TLE2426.....	σελ.39
Σχήμα 5-1: Το σχηματικό διάγραμμα του θερμίστορ με τον ενισχυτή.....	σελ.50

ΛΙΣΤΑ ΠΙΝΑΚΩΝ

Πίνακας 3-1: Οι βασικές προδιαγραφές του NodeMCU.....	σελ.26
Πίνακας 3-2: Οι βασικές προδιαγραφές του CD4051.....	σελ.33
Πίνακας 3-3: Οι βασικές προδιαγραφές του thermistor 10kΩ.....	σελ.35
Πίνακας 3-4: Οι παράμετροι λειτουργίας του TLE2426.....	σελ.39
Πίνακας 4-1: Τα βασικά εργαλεία του περιβάλλοντος ανάπτυξης.....	σελ.41

ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ

Στην σημερινή εποχή η τεχνολογία παίζει σημαντικό ρόλο στους περισσότερους τομείς κάθε βιομηχανίας όσο και στην προσωπική ζωή κάθε ανθρώπου. Ο τομέας της Υγείας είναι από τους σημαντικότερους τομείς όπου η συνεχής αναζήτηση λύσεων και η συνεχής βελτίωση είναι απαραίτητη.

Οι μηχανικοί βιοϊατρικής εφαρμόζουν ηλεκτρικές, χημικές, οπτικές, μηχανικές και άλλες φυσικές αρχές έτσι ώστε να κατανοήσουν, να τροποποιήσουν ή να ελέγξουν βιολογικά συστήματα (ανθρώπων ή ζώων). Αρχικά το εύρος της δραστηριότητας των μηχανικών βιοϊατρικής ήταν απλά η ανάπτυξη ιατρικών μηχανημάτων αλλά πλέον έχει διευρυνθεί σε πολλούς νέους τομείς. Τέτοιοι τομείς είναι:

- Η εφαρμογή συστημάτων ανάλυσης (προσομοιώσεις, φυσιολογικά μοντέλα κτλ.).
- Ο εντοπισμός, μέτρηση και παρακολούθηση φυσιολογικών σημάτων (όπως οι βιοαισθητήρες).
- Διαγνωστική ερμηνεία μέσω τεχνικών επεξεργασιών σημάτων βιοηλεκτρικών δεδομένων.
- Διαδικασίες και συσκευές για θεραπεία και αποκατάσταση.
- Συσκευές για αντικατάσταση ή αύξηση σωματικών λειτουργιών (τεχνητά όργανα).
- Ανάλυση δεδομένων ασθενών για την καλύτερη λήψη αποφάσεων αντιμετώπισης των προβλημάτων.
- Ιατρική απεικόνιση, όπως είναι η αναλυτική γραφική απεικόνιση ανατομίας ή μίας φυσιολογικής λειτουργίας.
- Η δημιουργία νέων συσκευών/προϊόντων.

Επομένως ο κλάδος των μηχανικών βιοϊατρικής είναι βασισμένος τόσο στην φυσικοχημεία όσο και στις επιστήμες υγείας. Ξεκινάει από θεωρητικά θέματα και φτάνει μέχρι σύγχρονες εφαρμογές. Εμπεριέχει έρευνα, ανάπτυξη, εφαρμογή και λειτουργία πραγμάτων.

Η ανάπτυξη βιοϊατρικών συσκευών και εξοπλισμού έχει σημειώσει μεγάλη πρόοδο και έχει συνεισφέρει στην βελτίωση της υγείας των ανθρώπων. Με την κατασκευή συσκευών που επιτρέπουν καλύτερη παρακολούθηση προσφέροντας ταυτόχρονα περισσότερη άνεση, οι ασθενείς ξοδεύουν λιγότερο χρόνο σε εξετάσεις και ανάρρωση με εξίσου καλά ή και καλύτερα αποτελέσματα από πριν.

Οι ψηφιακές συσκευές παίζουν μεγάλο ρόλο στην καθημερινή ζωή του ανθρώπου από την επικοινωνία μέχρι την ψυχαγωγία. Έτσι και στον τομέα της ιατρικής, η δύναμη φεύγει από τα νοσοκομεία και τους γιατρούς και μεταφέρεται σε προσωπικές φορητές συσκευές οι οποίες μπορούν να παρακολουθούν τα ζωτικά τους σημεία, να κάνουν μέχρι έναν ικανοποιητικό βαθμό εξετάσεις και άλλες λειτουργίες. Αυτό γίνεται πιο αργά στον τομέα της Ιατρικής καθότι είναι πιο συντηρητική η εξέλιξη αφού κάθε αλλαγή πρέπει να έχει ικανοποιήσει κάποιες απαιτήσεις. Η μετάβαση που αναφέρουμε έχει ήδη ξεκινήσει καθώς έχουμε ψηφιοποιήσει τις περισσότερες δραστηριότητες του ανθρώπου και πλέον μπορούμε να έχουμε μετρήσεις από το σπίτι του ασθενή όπως είναι το καρδιογράφημα, η πίεση, η θερμοκρασία του σώματος, η συγκέντρωση οξυγόνου και πολλών άλλων παραμέτρων, κρίσιμων για την υγεία του ανθρώπου.

Στην παρούσα εργασία ασχολούμαστε με το σήμα που παράγεται από την καρδιά και μέσω ενός κυκλώματος, προσπαθούμε να το απεικονίσουμε σε πραγματικό χρόνο σε μία βάση δεδομένων, ενώ ταυτοχρόνως μέσω άλλου κυκλώματος, πραγματοποιούμε λήψεις θερμοκρασιών σε πραγματικό χρόνο στην ίδια βάση δεδομένων. Η συσκευή αυτή είναι φορητή και εύχρηστη για να μπορεί να χρησιμοποιείται με ευκολία.

ΚΕΦΑΛΑΙΟ 2. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

Για την ευκολότερη κατανόηση της παρούσας εργασίας, παρουσιάζεται μια εισαγωγή σε βασικές γνώσεις που απαιτούνται. Σε αυτό το κεφάλαιο, εισάγονται έννοιες που θα χρησιμοποιηθούν στη συνέχεια, παρουσιάζεται η ανάλυση τους σε επιμέρους τμήματα, ενώ στα επόμενα κεφάλαια παρατίθενται το υλικό μέρος του συστήματος και οι πλατφόρμες ανάπτυξης λογισμικού που χρησιμοποιήθηκαν για την εκπόνηση της εργασίας.

2.1 Η έννοια της θερμοκρασίας

Θερμοκρασία είναι η φυσική ποσότητα που μετρά την ενέργεια κίνησης ή ταλάντωσης της ύλης σε ατομικό επίπεδο. Η ανταλλαγή της ενέργειας αυτής, όταν πιάνουμε κάτι με το χέρι για παράδειγμα, μας δίνει την αίσθηση του ζεστού ή του κρύου, με την κατάσταση μεγαλύτερης ενέργειας να αντιστοιχεί στο «ζεστό» ή «θερμό», όταν συνολικά παίρνουμε ενέργεια, και της κατάστασης μικρότερης ενέργειας, κατά την οποία αντιλαμβανόμαστε να χάνουμε συνολικά ενέργεια, να αντιστοιχεί στο «κρύο». Το αίτιο που δημιουργεί το αίσθημα του θερμού ή ψυχρού είναι η θερμότητα που όταν χορηγείται (απορροφάται) ή αφαιρείται (εκλύεται) από ένα σώμα προκαλεί "μεταβολή θερμοκρασίας". Συνεπώς θερμότητα και θερμοκρασία είναι διαφορετικές έννοιες. Η μεν θερμότητα είναι μορφή ενέργειας, η δε θερμοκρασία ιδιότητα και μέγεθος και εκπροσωπεί διαφορά δυναμικού [1].

2.1.1 Γενικά για την θερμοκρασία

Η θερμοκρασία μετριέται με ειδικά όργανα που λέγονται θερμόμετρα, η λειτουργία των οποίων βασίζεται στο φαινόμενο της διαστολής ή συστολής ως αποτέλεσμα παροχής ή αφαίρεσης της θερμότητας. Αλλά και η μεταβολή της θερμοκρασίας (ύψωση ή υποβιβασμός) είναι επίσης αποτέλεσμα της παροχής ή αφαίρεσης της θερμότητας. Έτσι με την παρατήρηση της διαστολής ή συστολής του υδραργύρου, που χρησιμοποιείται συνήθως στα θερμόμετρα, διαπιστώνεται και η μεταβολή της θερμοκρασίας η οποία αναγιγνώσκεται στη κατάλληλα βαθμολογημένη κλίμακα του θερμομέτρου. Γενικώς τα θερμόμετρα διακρίνονται σε "κοινά" ή "υδραργυρικά" και σε "θερμόμετρα οινόπνεύματος" (για χαμηλότερες θερμοκρασίες). Χρησιμοποιούνται επίσης και "ηλεκτρονικά θερμόμετρα" που βασίζονται στην αρχή του θερμοηλεκτρικού στοιχείου, επίσης τα "οπτικά" ή ηλεκτρονικά "πυρόμετρα" καθώς και άλλα ειδικών κατηγοριών.

Η βαθμολογία των θερμομέτρων γίνεται σε βαθμούς...:

- Celsius (Κελσίου) °C , στο μετρικό σύστημα,
- και σε βαθμούς Fahrenheit (Φαρενάιτ) °F, στο αγγλικό σύστημα.

Στο θερμόμετρο Κελσίου το μηδέν της κλίμακας (0°C) αντιστοιχεί στη θερμοκρασία τήξεως του πάγου, το δε 100 (100°C) στη θερμοκρασία βρασμού του ύδατος. Η ενδιάμεση αυτών απόσταση υποδιαιρείται σε 100 ίσα μέρη που καλούνται "βαθμοί Κελσίου".

Στο θερμόμετρο Φαρενάιτ η θερμοκρασία τήξεως του πάγου αντιστοιχεί στους 32°F, η δε θερμοκρασία βρασμού στους 212°F. Το ενδιάμεσο αυτών διάστημα υποδιαιρείται σε 180 ίσα μέρη που καλούνται "βαθμοί Φαρενάιτ".

Εκ των παραπάνω συμπεραίνεται ότι οι 212 βαθμοί Φαρενάιτ που περιέχονται μεταξύ 32°F και 212°F, αντιστοιχούν στους 100 βαθμούς Κελσίου, που περιέχονται μεταξύ 0°C και 100°C. Επομένως ένας βαθμός Κελσίου ισούται με 1,8 βαθμούς Φαρενάιτ +32. Όπου και ακολουθούν οι σχέσεις:

$$C = (F - 32)100/180 \text{ ή } C = (F - 32)/1,8$$
$$F = 180/100 \times C + 32 \text{ ή } F = 1,8 C + 32$$

2.1.2 Σχετική θερμοκρασία

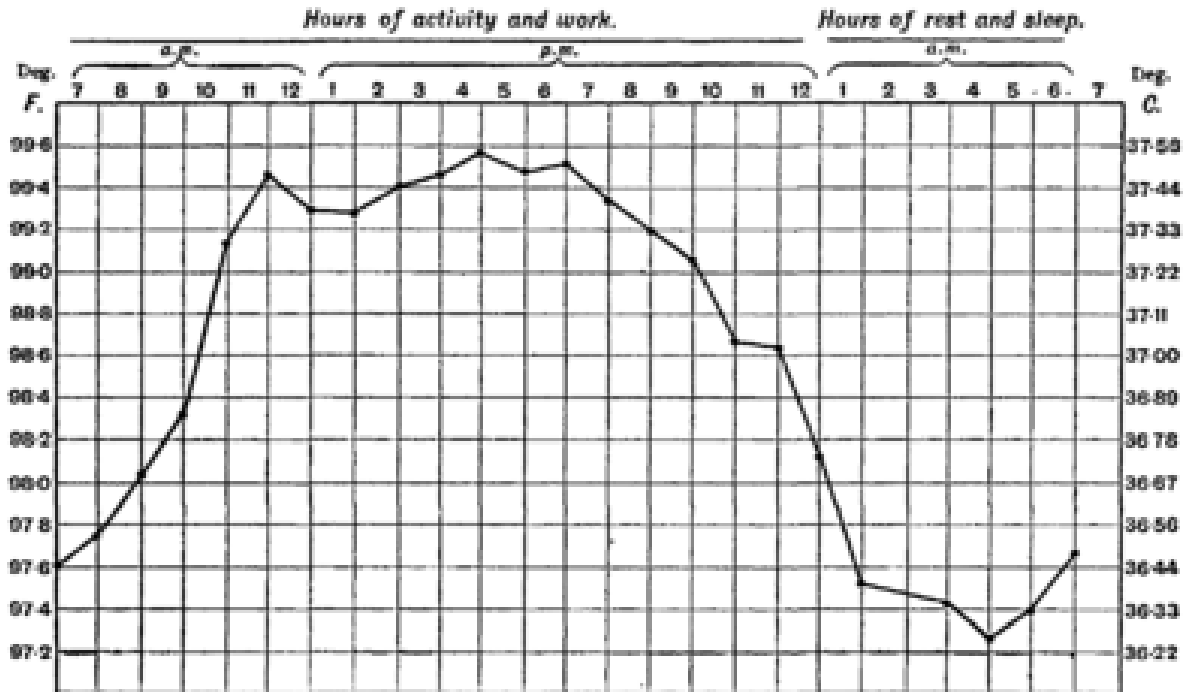
Κάθε θερμοκρασία που μετριέται αρχίζοντας από το 0° της κλίμακας Κελσίου ονομάζεται σχετική θερμοκρασία και καλείται θετική όταν είναι υψηλότερη του μηδενός και αρνητική όταν είναι χαμηλότερη. Η σχετική θερμοκρασία έχει ιδιαίτερα ευρεία χρήση τόσο στη καθημερινή ζωή του ανθρώπου όσο και στις διάφορες τεχνικές και μηχανολογικές εφαρμογές. Συμβολίζεται με το λατινικό γράμμα t.

2.1.3 Απόλυτη θερμοκρασία

Κάθε θερμοκρασία που έχει ως αρχή μέτρησης το απόλυτο μηδέν της όποιας κλίμακας (Κελσίου ή Φαρενάιτ) χαρακτηρίζεται απόλυτη θερμοκρασία. Το απόλυτο μηδέν είναι η θερμοκρασία από την οποία ξεκινά η κλίμακα Κέλβιν και που προσδιορίζεται για μεν την κλίμακα Κελσίου στους -273,15°C και για δε την κλίμακα Φαρενάιτ στους -459,67°F. Θεωρητικά είναι η κατάσταση εκείνη στην οποία ένα υλικό δεν έχει καμία άλλη ενέργεια παρά αυτή από τις κβαντομηχανικές ταλαντώσεις των ατόμων που το αποτελούν (ενέργεια μηδενικού σημείου). Η απόλυτη θερμοκρασία συμβολίζεται με το γράμμα T.

2.1.4 Εύρος θερμοκρασιών του ανθρώπινου σώματος

Η θερμοκρασία ενός ανθρώπινου σώματος, όπως και όλα τα υπόλοιπα βιοσήματα, υπόκειται σε διάφορες αλλαγές. Το σύνηθες εύρος είναι 36.1 – 37.2 °C, ανάλογα με την ώρα της ημέρας, όπως φαίνεται και στο Διάγραμμα 2-1 [2]. Η θερμοκρασία εξαρτάται από του ασθενή το φύλο, την ηλικία, την ώρα της ημέρας, την κατάσταση υγείας του (π.χ. μεταβολές της θερμοκρασίας του σώματος που παρουσιάζονται στις γυναίκες όταν βρίσκονται στις ημέρες της περιόδου), το σημείο του σώματος στο οποίο γίνεται η μέτρηση, το αν ο ασθενής κοιμάται ή όχι, όπως επίσης και από την συναισθηματική του κατάσταση [3]. Όταν η θερμοκρασία ενός ανθρώπινου σώματος βρεθεί σε διαστήματα εκτός των φυσιολογικών, είναι πολύ πιθανό να σημαίνει κάτι επικίνδυνο. Τιμές υπό των 35.5 °C δηλώνουν υποθερμία στον άνθρωπο, κάτι που μπορεί να αποβεί μοιραίο. Από την άλλη, θερμοκρασίες άνω των 37.5 °C εμφανίζονται όταν κάποιος ανεβάζει πυρετό, ένας φυσικός μηχανισμός του ανθρώπινου σώματος, ενώ τιμές άνω των 40.0 °C είναι άκρως επικίνδυνες. Επίσης, το ανθρώπινο σώμα υπόκειται σε θερμοκρασιακές αλλαγές κατά τη διάρκεια του ύπνου: πτώση της τάξης των 1-2°C είναι κάτι το φυσιολογικό κατά τη διάρκεια του βαθύ ύπνου [4]. Το ανθρώπινο σώμα μπορεί να ρυθμίσει την θερμοκρασία του ώστε να βρίσκεται στις ονομαστικές του τιμές, καθώς η θερμοκρασία επηρεάζει τον ρυθμό με τον οποίο συμβαίνουν χημικές ενώσεις. Ο έλεγχος αυτός είναι μέρος της ομοιόστασης, της ικανότητας δηλαδή ενός οργανισμού να διατηρεί το εσωτερικό του περιβάλλον σταθερό παρά τις μεταβολές του εξωτερικού περιβάλλοντος [5].

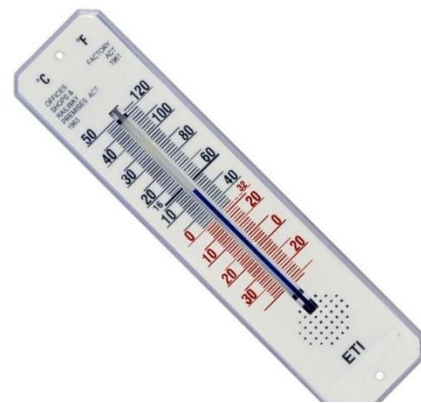


Διάγραμμα 2-1: Η θερμοκρασία του ανθρώπινου σώματος σε ένα 24-ωρο.

Η λήψη της θερμοκρασίας ενός σώματος συμβαίνει σε κάποια συγκεκριμένα σημεία του, σε σημεία τέτοια που η μέτρηση θα είναι όσο πιο κοντινή γίνεται στην πραγματική, αφού η εσωτερική θερμοκρασία ενός σώματος δεν ταυτίζεται πάντα με τη θερμοκρασία του δέρματος. Τέτοια σημεία είναι το εσωτερικό του στόματος ή του αυτιού, οι μασχάλες, ο πρωκτός και το μέτωπο. Ένα γνωστό παράδειγμα είναι όταν ακουμπάμε το μέτωπο κάποιου για να αντιληφθούμε με έναν πρόχειρο τρόπο αν έχει ανεβάσει πυρετό. Η πιο αξιόπιστη μέτρηση γίνεται με τη χρήση του απλού θερμόμετρου, είτε ηλεκτρονικού είτε με υδράργυρο. Στην παρούσα διπλωματική, αναζητούμε τη θερμοκρασία του δέρματος στο σημείο του δαχτύλου. Το σημείο εκείνο ανήκει στα σημεία του σώματος που δεν αντιπροσωπεύουν την πραγματική τιμή της θερμοκρασίας του σώματος, οπότε αναμένεται οι τιμές να μην είναι κοντά σε σχέση με την κανονική θερμομέτρηση. Παρατηρώντας τις αλλαγές της θερμοκρασίας, εξάγονται συμπεράσματα για την κατάσταση του ασθενή.



Εικόνα 2-1: Διάφορα είδη θερμόμετρων



Εικόνα 2-2: Θερμόμετρο χώρου

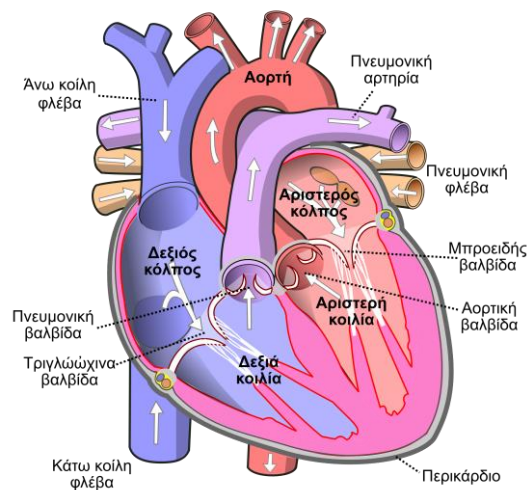
2.2 Η λειτουργία της καρδιάς

Όπως είναι γνωστό η καρδιά είναι ένας μυς όπου στην ουσία αποτελεί την αντλία για την κυκλοφορία του αίματος σε όλο το σώμα. Η κυκλοφορία του αίματος είναι απαραίτητη για να μεταφέρει σε όλα τα όργανα και τους μύες του σώματος φρέσκο οξυγόνο και να φύγει το διοξείδιο του άνθρακα που δημιουργείται κατά την καύση.

Η καρδιά αποτελείται από τέσσερις κοιλότητες. Οι δύο πάνω κοιλότητες ονομάζονται κόλποι και οι δύο κάτω ονομάζονται κοιλίες, αριστερή και δεξιά αντίστοιχα. Το αίμα εισέρχεται από τον δεξιό κόλπο της καρδιάς μέσω των φλεβών και προωθείται στην δεξιά κοιλία όπου και το αίμα οξυγονώνεται μέσω των πνευμόνων. Έπειτα το αίμα αυτό πρέπει να σταλεί σε όλα τα μέρη του σώματος. Αυτό επιτυγχάνεται πηγαίνοντας στον αριστερό κόλπο και μετά την αριστερή κοιλία. Η αριστερή κοιλία είναι το σημαντικό κομμάτι του μυοκαρδίου καθώς αυτό είναι υπεύθυνο για την αποστολή του φρέσκου σε οξυγόνο αίματος με τη συστολή του μέσω της αορτής και των υπόλοιπων αρτηριών. Για την κυκλοφορία του αίματος μέσα στην καρδιά υπεύθυνες είναι οι βαλβίδες (τριγλώχινα, πνευμονική, μιτροειδής και αορτική) οι οποίες επιτρέπουν την κυκλοφορία του αίματος προς μία μόνο κατεύθυνση εμποδίζοντας τυχόν παλινδρόμηση [6].

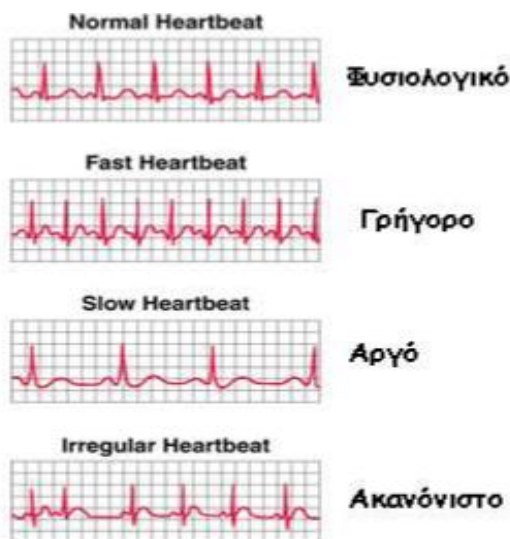
2.2.1 Ηλεκτρικά σήματα από το ανθρώπινο σώμα

Η δραστηριότητα των ανθρωπίνων μυών παράγουν ηλεκτρικό σήμα. Η πηγή του σήματος αυτού προέρχεται από τα κύτταρα και πιο συγκεκριμένα την κυτταρική μεμβράνη. Αρχικά η κυτταρική μεμβράνη είναι ηλεκτρικά πολωμένη καθώς υπάρχει διαφορά δυναμικού μεταξύ του εσωτερικού και του εξωτερικού της. Όταν λοιπόν τα κύτταρα ενός μυ διεγερθούν, τότε το δυναμικό αυτό αλλάζει και στην ουσία η κυτταρική μεμβράνη εκπολώνεται για κάποιο διάστημα, δηλαδή η διαφορά του δυναμικού μεταξύ εσωτερικού και εξωτερικού φτάνει κοντά στο μηδέν. Όλα τα κύτταρα μαζί προκαλούν ένα ηλεκτρικό σήμα όταν διεγείρονται το οποίο μεταφέρεται μέσα από τους ιστούς του οργανισμού με αποτέλεσμα να μπορεί να μετρηθεί αν τοποθετηθούν αισθητήρες (ηλεκτρόδια) στα σωστά σημεία [7].



Σχήμα 2-1: Η ανατομία της καρδιάς

Τα σήματα αυτά αποστέλλονται για αναλογική επεξεργασία και έπειτα ψηφιακή μετατροπή. Εκεί τα σήματα ενισχύονται, φιλτράρονται και λαμβάνουν ψηφιακή μορφή. Όταν το αναλογικό σήμα έχει ψηφιοποιηθεί και έχει μετατραπεί σε μορφή στην οποία μπορεί να αποθηκευτεί και να επεξεργαστεί από ηλεκτρονικούς υπολογιστές, υπάρχουν πολλές μέθοδοι και εφαρμογές που μπορούν να εφαρμοστούν.



Σχήμα 2-2: Γράφημα δύο παλμών μιας καρδιάς

Ένα σύστημα που διαβάζει βιοσήματα συνήθως εμπεριέχει και μία οθόνη στην οποία προβάλλει τα αποτελέσματα έτσι ώστε οι άνθρωποι να μπορούν να τα δουν και να τα καταλάβουν. Οι απεικονίσεις αυτές μπορεί να είναι αριθμητικές ή γραφικές, συνεχείς ή διακριτές, μόνιμες ή προσωρινές. Οι περισσότερες συσκευές απεικόνισης είναι οπτικές, αλλά υπάρχουν και άλλων ειδών όπως ακουστικές (πχ. ένα μπιπ με κάθε παλμό της καρδιάς).

Εκτός από την απεικόνιση πολλά συστήματα έχουν τη δυνατότητα αποθήκευσης των δεδομένων. Σε πολλές συσκευές τα δεδομένα αποθηκεύονται για μικρό χρονικό διάστημα έτσι ώστε να επεξεργαστούν περισσότερο ή ώστε κάποια άλλη διεργασία να χρησιμοποιήσει τα δεδομένα αυτά. Σε άλλες περιπτώσεις τα σήματα αποθηκεύονται μόνιμα ώστε να μπορούν να υποστούν διαφορετικές τεχνικές επεξεργασίας.

Σήμερα με την τεχνολογία των smartphones και του Internet οι δυνατότητες διευρύνονται κατά πολύ. Τα βιοσήματα μπορούν να διαβάζονται από μία συσκευή σε ένα μέρος όπως το σπίτι του ασθενή ή σε ένα εξεταστικό κέντρο σε ένα απομακρυσμένο νησί, και τα βιοσήματα αυτά να αποστέλλονται σε άλλη συσκευή για επεξεργασία και ανάλυση. Κάτι τέτοιο κάνει δυνατή την γρήγορη διάγνωση εξ' αποστάσεως εάν ο ασθενής έχει κάποιο πρόβλημα.

2.2.2 Η λειτουργία της καρδιάς ως ηλεκτρικό σήμα

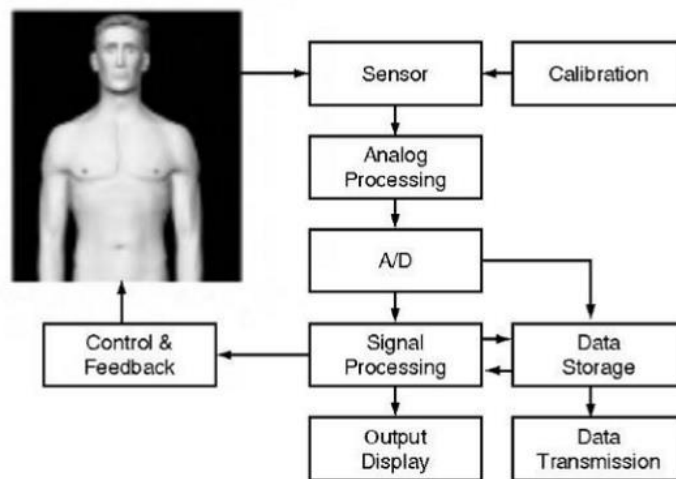
Όπως αναφέρθηκε και προηγουμένως, η καρδιά είναι ουσιαστικά ένας μυς που βοηθάει στην κυκλοφορία του αίματος σε όλο το σώμα. Η δραστηριότητα της καρδιάς μπορεί να παρατηρηθεί και να μελετηθεί ως ηλεκτρικό φαινόμενο αφού σαν λειτουργία έχει τη συστολή και διαστολή επιμέρους μυών της.

Η διέγερση των μυών της καρδιάς γίνεται από ένα σύστημα στην καρδιά που ονομάζεται βηματοδότης. Κομμάτι του βηματοδότη είναι ο φλεβόκομβος ο οποίος είναι αυτός που δίνει τον καρδιακό παλμό και δουλεύουν οι υπόλοιποι μυς στην καρδιά. Η ακριβής μετάβαση του σήματος ξεκινά από τον φλεβόκομβο που βρίσκεται ανάμεσα στην δεξιά κόλπο και την άνω κοίλη φλέβα. Από εκεί το σήμα μεταφέρεται στους δύο κόλπους με αποτέλεσμα την σύσπασή τους. Έπειτα το σήμα πηγαίνει σε αυτό που ονομάζουμε κολποκοιλιακό κόμβο. Ο κολποκοιλιακός κόμβος λειτουργεί ως ελεγκτής του ηλεκτρικού σήματος μεταξύ του προηγούμενου συστήματος και των κοιλιών ώστε να υπάρχει σωστός συγχρονισμός. Στην πράξη επιβάλλει μια μικρή καθυστέρηση του σήματος της τάξης του 0.1 s προτού περαστεί στις κοιλίες και προκαλέσει τη σύσπασή τους [8].

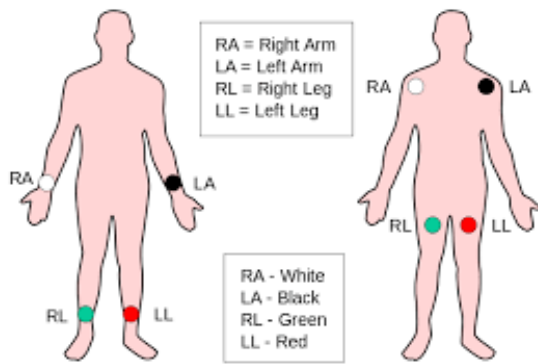
2.2.3 Ανάλυση της μορφής του καρδιογραφήματος

Το ηλεκτροκαρδιογράφημα είναι η διαδικασία κατά την οποία καταγράφεται η ηλεκτρική δραστηριότητα της καρδιάς για κάποιο χρονικό διάστημα χρησιμοποιώντας ηλεκτρόδια πάνω στο σώμα του ασθενή. Τα ηλεκτρόδια αυτά εντοπίζουν πολύ μικρές ηλεκτρικές αλλαγές στο δέρμα που προκαλούνται από την κίνηση των μυών της καρδιάς σε κάθε παλμό. Στην παρακάτω εικόνα παρουσιάζεται η κυματομορφή παλμών της καρδιάς.

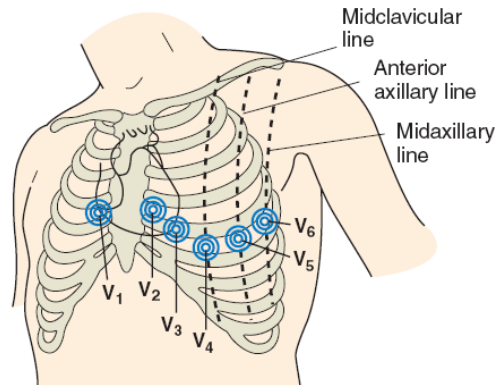
Στη συνήθη μέτρηση του ηλεκτροκαρδιογραφήματος χρησιμοποιούνται 10 ηλεκτρόδια όπου τοποθετούνται στο στήθος και πλευρά του ασθενή όπως στα σχεδιαγράμματα των Σχημάτων 2-4 και 2-5. Έτσι μετριέται το ηλεκτροκαρδιογράφημα συνήθως σε διάρκεια 10s. Στην παρούσα εργασία θα γίνει χρήση 3-πολικού καλωδίου, δηλαδή καλωδίου με 3 ηλεκτρόδια.



Διάγραμμα 2-2: Διάγραμμα ροής διαδικασίας λήψης και επεξεργασίας ενός βιοσήματος



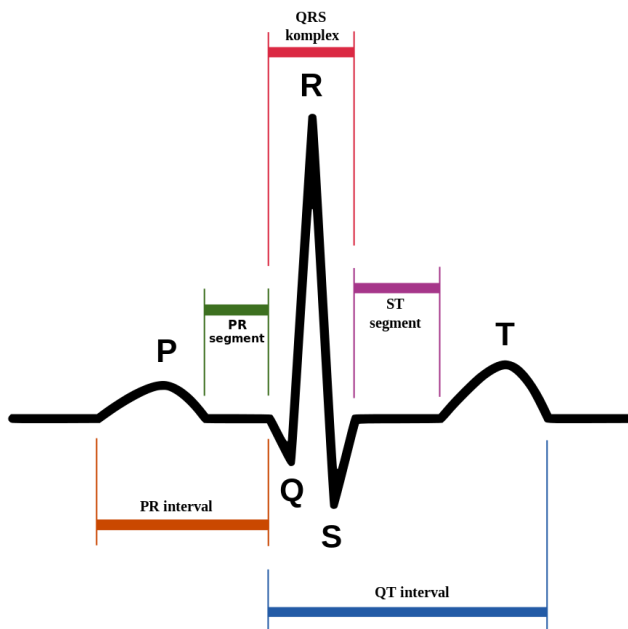
Σχήμα 2-3: Τοποθέτηση ηλεκτροδίων στα άκρα του σώματος



Σχήμα 2-4: Τοποθέτηση ηλεκτροδίων στο στήθος

Ένα πλήρες ηλεκτροκαρδιογράφημα μπορεί να παράσχει πολλές χρήσιμες πληροφορίες στον έμπειρο γιατρό. Μερικές από αυτές τις πληροφορίες είναι η μέτρηση των παλμών και του ρυθμού της καρδιάς, το μέγεθος και τη θέση των καρδιακών θαλάμων καθώς και την ύπαρξη ζημίας στα μυϊκά κύτταρα της καρδιάς, την επιρροή καρδιακών φαρμάκων και την λειτουργία βηματοδοτών.

Ένα τυπικό ηλεκτροκαρδιογράφημα είναι ένα επαναλαμβανόμενος κύκλος τριών κομματιών: το κύμα P (atrial depolarization), το QRS complex (ventricular depolarization) και το κύμα T (ventricular repolarization).



Σχήμα 2.5: Σχηματική αναπαράσταση ενός κανονικού ηλεκτροκαρδιογραφήματος

Το κύμα P αναπαριστά την εκπόλωση του κόλπου. Έχει διάρκεια μικρότερη από 80 ms. Εάν έχει μεγαλύτερη διάρκεια μπορεί να σημαίνει κολπική διεύρυνση. Το P κύμα είναι συνήθως όρθιο στις μετρήσεις.

Το διάστημα PR μετριέται από την αρχή του κύματος P μέχρι την αρχή του QRS complex. Το διάστημα αυτό πρέπει να έχει διάρκεια 120 ms–200 ms.

Το QRS complex αναπαριστά την γρήγορη εκπόλωση των κοιλιών. Λόγω της μεγαλύτερης μυϊκής μάζας που έχουν οι κοιλίες σε σχέση με τους κόλπους το σύμπλεγμα QRS έχει συνήθως πολύ μεγαλύτερο πλάτος από αυτό του κύματος P. Η διάρκειά του είναι από 80 ms έως 100 ms.

Το σημείο J είναι το σημείο στο οποίο το σύμπλεγμα QRS τελειώνει και το τμήμα ST ξεκινά. Το τμήμα ST είναι το τμήμα που ενώνει το σύμπλεγμα QRS με το κύμα T. Αναπαριστά την περίοδο με την οποία εκπολώνονται οι κοιλίες. Το κύμα T αναπαριστά την επαναπόλωση των κοιλιών. Γενικά είναι όρθιο στις μετρήσεις όπως το κύμα P. Η διάρκειά του είναι στα 160 ms. Το κύμα U θεωρείται ότι προκαλείται από την επαναπόλωση του μεσοκοιλιακού διαφράγματος. Συνήθως έχει μικρό πλάτος και πολλές φορές δεν φαίνεται καν. Εάν το κύμα U είναι πολύ επιφανές μπορεί να σημαίνει προβλήματα.

Όλοι τα στοιχεία ενός καρδιακού παλμού έχουν καθορισμένες προδιαγραφές χρόνου, διάρκειας, πλάτους και μορφολογίας. Οποιαδήποτε απόκλιση από τις προδιαγραφές αυτές μπορεί να έχει παθολογική και κλινική σημασία. Τα σήματα ενός καρδιογραφήματος είναι της τάξης του 1 mV. Όπως και στα υπόλοιπα βιοσήματα, το ηλεκτροκαρδιογράφημα επηρεάζεται από την κίνηση του ασθενή. Οι κινήσεις αυτές μπορούν να προκαλέσουν ψεύτικες ενδείξεις στην μέτρηση για δυσρυθμίες και άλλα προβλήματα [9].

2.2.4 Ag/AgCl ηλεκτρόδια (υγρά)

Τα υγρά ηλεκτρόδια είναι αυτά που χρησιμοποιούνται κατά κόρον στα νοσοκομεία και σε κλινικές. Είναι καλά χαρακτηρισμένα και μελετημένα μετά από τόσες δεκαετίες όπου χρησιμοποιούνται. Οι περισσότερες από τις ιδιότητές τους είναι γνωστές και κατανοητές και υπάρχουν πολλές τεχνικές πλέον όπου μπορούν να αντιμετωπίσουν γνωστά προβλήματα όπως είναι ο θόρυβος και η ολίσθηση χαμηλών συχνοτήτων. Με σωστή προετοιμασία του ασθενή, τα υγρά ηλεκτρόδια μπορούν και παρέχουν άριστο σήμα για μετρήσεις και ανάλυση. Ο λόγος που χρησιμοποιούνται τόσο πολύ τα υγρά ηλεκτρόδια και δεν έχουν αντικατασταθεί από διαφορετική τεχνολογία ηλεκτροδίων είναι η άριστη ποιότητα του σήματος που προσφέρουν. Λόγω της κόλλας που χρησιμοποιούν, κολλάνε καλά στο δέρμα οπότε ο θόρυβος από κίνηση είναι πολύ μικρός. Λόγω του τζελ τα ηλεκτρόδια έχουν αρκετά μικρή σύνθετη αντίσταση καθώς το τζελ είναι από αγώγιμο υλικό που βοηθάει στην μεταφορά του σήματος από το σώμα στο κύκλωμα [10].



Σχήμα 2.6: Υγρό ηλεκτρόδιο Ag/AgCl

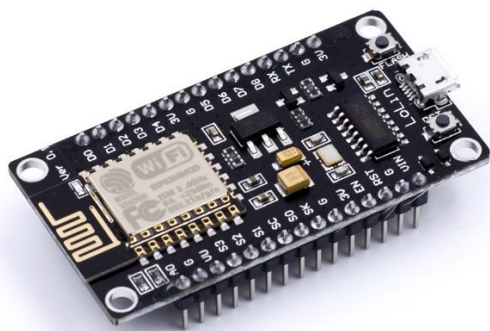
ΚΕΦΑΛΑΙΟ 3. ΥΛΙΚΟ ΜΕΡΟΣ ΣΥΣΤΗΜΑΤΟΣ (HARDWARE)

Για να καταφέρουμε να λάβουμε τα βιοσήματα που μας ενδιαφέρουν και να τα επεξεργαστούμε, θα πρέπει να επιστρατεύσουμε διάφορα εργαλεία που θα μας βοηθήσουν στην υλοποίηση. Θα χρειαστεί να χρησιμοποιήσουμε διάφορα ηλεκτρονικά εξαρτήματα από τα οποία θα λαμβάνονται κάποια φυσικά σήματα και με έναν μικροελεγκτή θα μετατρέπονται σε ψηφιακά. Ο μικροελεγκτής αποτελεί το κυρίως τμήμα της συσκευής. Η αρχιτεκτονική του μικροελεγκτή επιτρέπει τη σύνδεση διαφόρων εξαρτημάτων (αισθητήρων, συσκευών κτλ) τα οποία με τον κατάλληλο προγραμματισμό επιτελούν έναν στόχο. Ο μικροελεγκτής ελέγχει και διαχειρίζεται ένα σύνολο υλικών-εξαρτημάτων. Παρακάτω φαίνονται και αναλύονται τα εξαρτήματα και ο μικροελεγκτής που χρησιμοποιήθηκαν για την συσκευή.

3.1 Μικροελεγκτής NodeMCU ESP8266

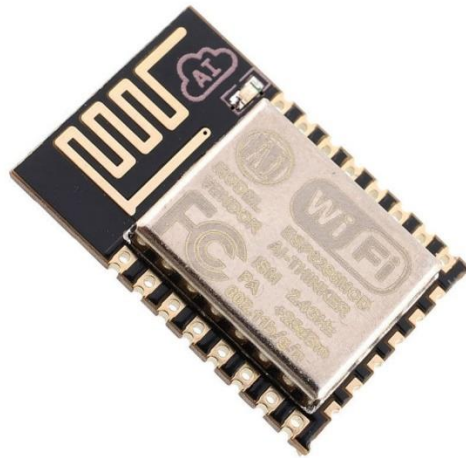
Το NodeMCU (Node MicroController Unit) είναι ένα λογισμικό ανοιχτού κώδικα με περιβάλλον ανάπτυξης hardware που βασίζεται σε ένα φθινό System-on-a-Chip (SoC) που ονομάζεται ESP8266. Στηρίζεται σε Lua ανοιχτού κώδικα και περιλαμβάνει υλικό που εξαρτάται από την μονάδα ESP-12E. Το ESP8266, που σχεδιάστηκε και κατασκευάστηκε από την Espressif Systems, περιέχει τα κρίσιμα στοιχεία ενός υπολογιστή: CPU, RAM, δικτύωση (WiFi), ακόμη και ένα σύγχρονο λειτουργικό σύστημα και SDK. Αυτό το καθιστά εξαιρετική επιλογή για εφαρμογές που βασίζονται στο IoT.

Το NodeMCU μπορεί να τροφοδοτηθεί χρησιμοποιώντας στην υποδοχή του ένα Micro USB καλώδιο και μια ακίδα VIN (Εξωτερική ακίδα τροφοδοσίας). Υποστηρίζει διασύνδεση UART, SPI και I2C ενώ περιλαμβάνει ακίδες σύνδεσης με τον μικροεπεξεργαστή καθώς και καρφιά για στερέωμα σε κάποια επιφάνεια (π.χ. breadboard). Στην Εικόνα 3-1 φαίνεται το εν λόγω board, όπου διακρίνονται τα επιμέρους στοιχεία του που προαναφέρθηκαν, όπως επίσης και το πιο βασικό του, ο μικροελεγκτής ESP8266 [11].

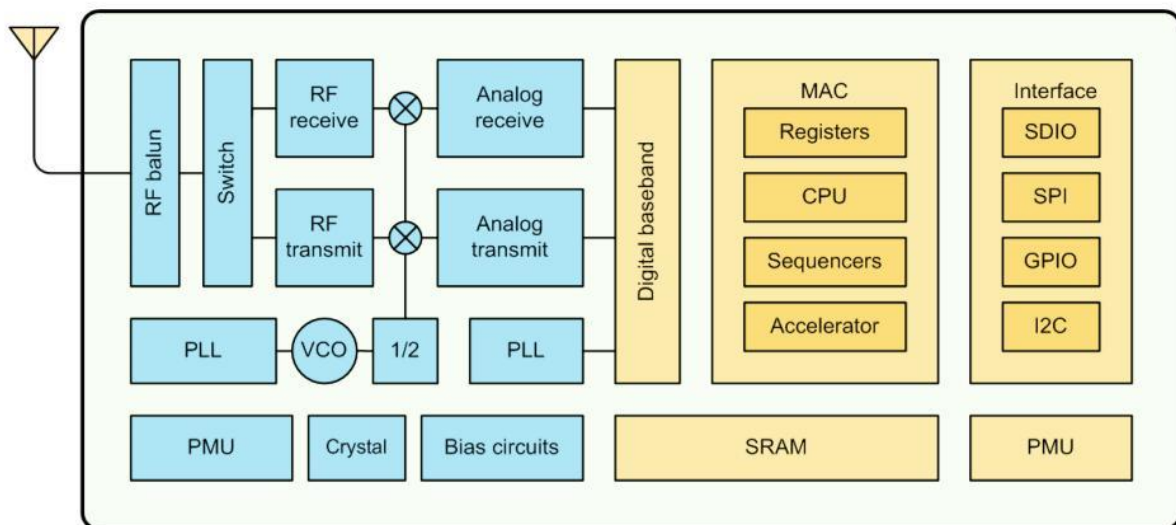


Εικόνα 3-1: Το NodeMCU ESP8266 της Espressif

Το ESP8266 αποτελεί ένα μεγάλο κομμάτι της επιφάνειας της πλακέτας, που είναι και λογικό αφού γύρω από αυτόν χτίζεται η πλακέτα καθώς αποτελεί και τον «εγκέφαλό» της. Ως μέρος του module ESP-12E, ανήκει στην πλατφόρμα Smart Connectivity Platform (ESCP), μια σειρά χαμηλού κόστους και ισχύος System on a Chip (SoC) μικροελεγκτών που διαθέτουν ενσωματωμένο Wi-Fi και Bluetooth. Στον πυρήνα του περιλαμβάνει μικροεπεξεργαστή Tensilica Xtensa 32-bit LX106 RISC [12]. Στην Εικόνα 3-2 φαίνεται ο μικροελεγκτής ESP-12E, ενώ στο Διάγραμμα 3-3 βλέπουμε το μπλοκ διάγραμμα του.



Εικόνα 3-2: Το ESP-12E της Espressif



Διάγραμμα 3-1: Το μπλοκ διάγραμμα του ESP-12E

Τα βασικά χαρακτηριστικά του ESP8266 είναι:

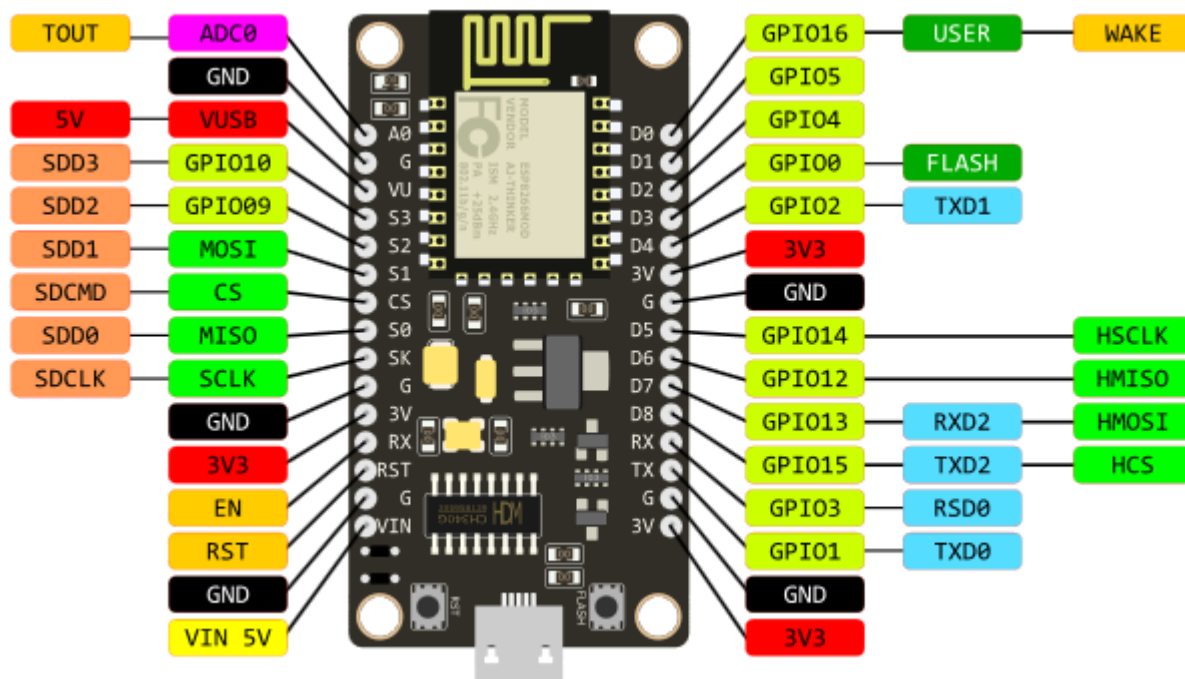
- Υποστηρίζει Wi-Fi 802.11 b/g/n
- Υπάρχει Ενσωματωμένος μικροελεγκτής 32 bit χαμηλής ισχύος
- Ενσωματωμένος μετατροπέας αναλογικού σε ψηφιακό των 10 bit
- Ενσωματωμένη πλήρης στοίβα TCP/IP
- Υποστηρίζει λειτουργίες Station, Soft Access Point, Station + Soft Access Point
- Ενσωματωμένος διακόπτης TR, balun, LNA, ενισχυτής ισχύος και αντίστοιχο δίκτυο
- Διαθέτει εσωτερικούς PLL, ρυθμιστές και μονάδες διαχείρισης ενέργειας
- Υποστηρίζει ποικιλομορφία χώρου / ποικιλομορφία κεραίας
- WiFi 2,4 GHz με υποστήριξη για WPA/WPA2
- Υποστηρίζει WiFi Direct
- Η λειτουργία Smart Link υποστηρίζεται τόσο για συσκευές Android όσο και για iOS
- Αφύπνιση και μετάδοση πακέτων σε λιγότερο από 2 mS
- Ισχύς εξόδου έως +20 dBm σε λειτουργία 802.11b
- Η κατανάλωση ρεύματος σε κατάσταση αναμονής είναι μικρότερη από 1 mW
- Η ισχύς του Deep Sleep είναι μικρότερη από 10μΑ και το ρεύμα διαρροής απενεργοποίησης είναι μικρότερο από 5μΑ
- Περιλαμβάνει UART – 2 Tx & 1 Rx
- Πιστοποιημένο από FCC, CE, TELEC, Wi-Fi Alliance και SRRC

Πίνακας 3-1: Οι βασικές προδιαγραφές του NodeMCU

n/n	Official NodeMCU	NodeMCU Carrier Board	LoLin NodeMCU
Microcontroller	ESP-8266 32-bit	ESP-8266 32-bit	ESP-8266 32-bit
NodeMCU Model	Amica	Amica	Clone LoLin
NodeMCU Size	49mm x 26mm	49mm x 26mm	58mm x 32mm
Carrier Board Size	n/a	102mm x 51mm	n/a
Pin Spacing	0.9" (22.86mm)	0.9" (22.86mm)	1.1" (27.94mm)
Clock Speed	80 MHz	80 MHz	80 MHz
USB to Serial	CP2102	CP2102	CH340G
USB Connector	Micro USB	Micro USB	Micro USB
Operating Voltage	3.3V	3.3V	3.3V
Input Voltage	4.5V-10V	4.5V-10V	4.5V-10V
Flash Memory/SRAM	4 MB / 64 KB	4 MB / 64 KB	4 MB / 64 KB
Digital I/O Pins	11	11	11
Analog In Pins	1	1	1
ADC Range	0-3.3V	0-3.3V	0-3.3V
UART/SPI/I2C	1/1/2001	1/1/2001	1/1/2001
WiFi Built-In	802.11 b/g/n	802.11 b/g/n	802.11 b/g/n
Temperature Range	-40C - 125C	-40C - 125C	-40C - 125C
Product Link		NodeMCU	NodeMCU

Οι παραπάνω προδιαγραφές αποτέλεσαν και κριτήριο για την επιλογή του ESP8266. Πιο συγκεκριμένα, τα πλεονεκτήματα που έχει η πλακέτα αυτή, όσον αφορά το θέμα της διπλωματικής, είναι:

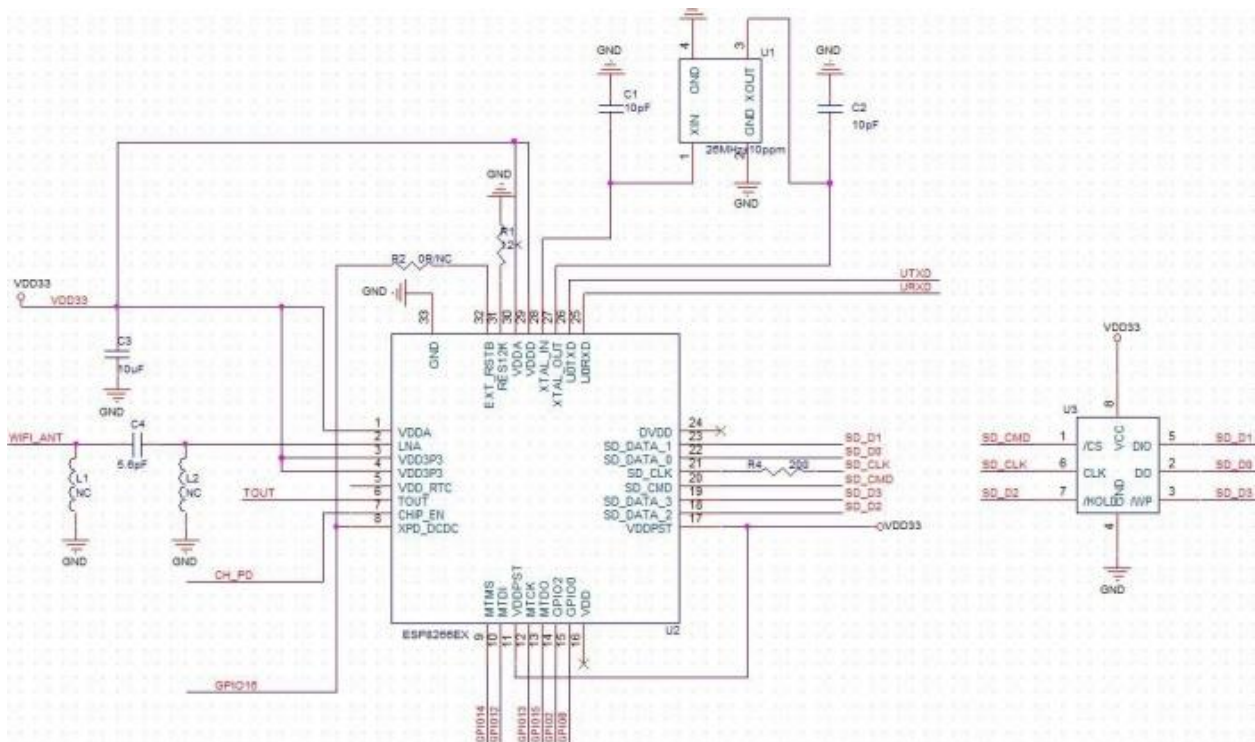
1. Η δυνατότητα του module ESP-12E για σύνδεση μέσω Bluetooth και Wi-Fi μας εξασφαλίζει ότι οι δύο συσκευές θα μπορούν να συνδεθούν μεταξύ τους και να στείλουν δεδομένα (μέσω του Bluetooth), καθώς και ότι θα μπορούμε να έχουμε σύνδεση της μίας συσκευής με server (μέσω του Wi-Fi).
2. Η διαθεσιμότητα των πρωτοκόλλων επικοινωνίας I 2C και SPI προσφέρουν επικοινωνία των μικροελεγκτών με τους υπόλοιπους αισθητήρες.
3. Το αρκετά μικρό του μέγεθος μας βεβαιώνει πως οι τελικές συσκευές που θα κατασκευαστούν δεν θα είναι ογκώδεις, άβολες ή βαριές.
4. Το USB-to-UART στοιχείο της πλακέτας επιτρέπει τη σύνδεση του μικροελεγκτή κατευθείαν με υπολογιστή για προγραμματισμό και debugging.



Διάγραμμα 3-2: Το διάγραμμα των Pinout του ESP8266

Οι ακίδες του NodeMCU board με βάση το Διάγραμμα 3-2 διαμορφώνονται ως εξής [13]:

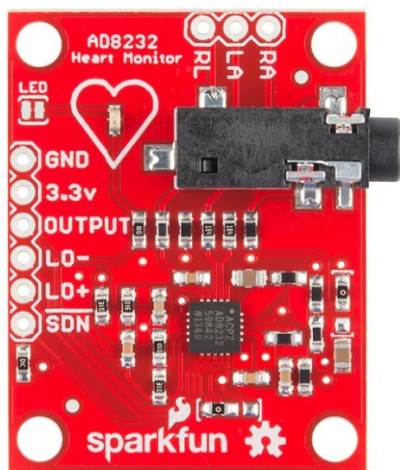
- **Micro-USB** - Το NodeMCU τροφοδοτείται μέσω της θύρας USB.
- **RST** - Ακίδα επανεκκίνησης του μικροελεγκτή.
- **3V** - Η έξοδος του 3.3V ρυθμιστή τάσης.
- **GND** - Η κοινή γείωση για τις τάσεις τροφοδοσίας και τάσεις λογικής
- **Vin** - Εξωτερική τροφοδοσία 5V.
- **A0** - Μετατροπέας αναλογικού σε ψηφιακό σήμα 10 bit. Χρησιμοποιείται για τη μέτρηση αναλογικής τάσης στην περιοχή από 0-3,3V.
- **EN** - Chip ενεργοποίησης pin.
- **D0 to D8, RX, TX** -Τα pins εισόδου/εξόδου. Χρησιμοποιούνται επίσης για UART, SPI, I2C, Flash και wake pin.
- **SD1, SCMD, SD0, SCLK** - Τέσσερις ακίδες διαθέσιμες για επικοινωνία SPI.
- **SD2** - Pin δεδομένων 0 για διεπαφή SDIO και χρησιμοποιείται επίσης ως GPIO9.
- **SD3** - Pin δεδομένων 3 για διεπαφή SDIO και χρησιμοποιείται επίσης ως GPIO10.
- **VV** - Κατοχυρωμένο pin 5V.



Εικόνα 3-3: Το σχηματικό του NodeMCU ESP8266

3.2 AD8232 Monitor καρδιακού παλμού μίας απαγωγής

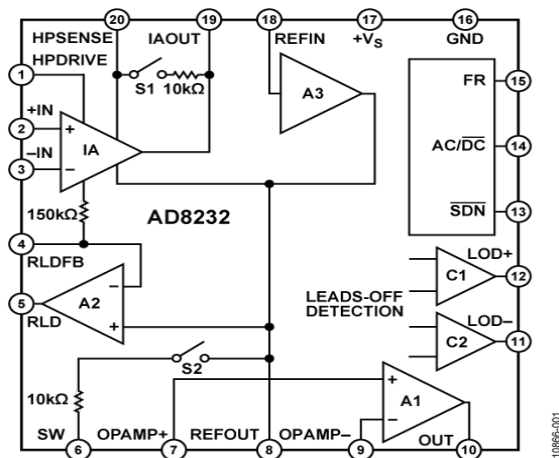
Η πλακέτα παρακολούθησης καρδιακών παλμών AD8232 της εταιρείας Sparkfun είναι ένας αποδοτικός τρόπος μέτρησης της ηλεκτρικής δραστηριότητας της καρδιάς, δηλαδή του ΗΚΓ και ως έξοδο έχει αναλογικό σήμα. Τα ΗΚΓ μπορεί να είναι εξαιρετικά θορυβώδη κι έτσι το AD8232 λειτουργεί ως ενισχυτής, για να βοηθήσει να αποκτηθεί εύκολα ένα καθαρό σήμα από τα διαστήματα PR και QT. Έχει σχεδιαστεί για να εξάγει, να ενισχύει και να φιλτράρει δυναμικά μικρά σήματα παρουσία θορυβωδών συνθηκών, όπως η κίνηση ή η τοποθέτηση απομακρυσμένου ηλεκτροδίου. Τα SDN, LO+, LO-, OUTPUT, 3.3V, GND παρέχουν τη δυνατότητα για λειτουργία με το ESP8266 και το Arduino. Παρέχονται επίσης σε αυτήν την πλακέτα οι ακίδες RA (δεξιού βραχίονα), LA (αριστερού βραχίονα) και RL (δεξιού ποδιού) για να προσαρτηθούν και να χρησιμοποιηθούν αισθητήρες. Επιπλέον, υπάρχει μία ενδεικτική λυχνία LED που αναβοσβήνει σύμφωνα με τον καρδιακό ρυθμό [14].



Εικόνα 3-4: Η πλακέτα του AS8232

Αποτελείται από έναν εξειδικευμένο ενισχυτή οργάνων (IA), έναν τελεστικό ενισχυτή (A1), ενισχυτή δεξιού ποδιού (A2) και προσωρινό buffer αναφοράς (A3). Το AD8232 μπορεί να εφαρμόσει ένα διπολικό υψηλοπερατό φίλτρο για την εξάλειψη των τεχνουργημάτων κίνησης και του δυναμικού ημιστοιχείου ηλεκτροδίου. Αυτό το φίλτρο είναι στενά συνδεδεμένο με την αρχιτεκτονική οργάνων του ενισχυτή για να επιτρέψει τόσο μεγάλο κέρδος όσο και υψηλοπερατό φιλτράρισμα σε ένα μόνο στάδιο, εξοικονομώντας έτσι χώρο και κόστος. Ένας μη δεσμευμένος λειτουργικός ενισχυτής επιτρέπει στο AD8232 να δημιουργήσει ένα τριπολικό χαμηλοπερατό φίλτρο για την αφαίρεση πρόσθετου θορύβου. Ο χρήστης μπορεί να επιλέξει την αποκοπή συχνότητας όλων των φίλτρων για να ταιριάζει σε διαφορετικούς τύπους εφαρμογών.

Περιλαμβάνει επίσης μια λειτουργία γρήγορης επαναφοράς που μειώνει τη διάρκεια των κατά τα άλλα μακρών ουρών καθίζησης των υψηλοπερατών φίλτρων. Μετά από μια απότομη αλλαγή σήματος που οδηγεί τον ενισχυτή (όπως μια κατάσταση απαγωγής), το AD8232 προσαρμόζεται αυτόματα σε υψηλότερη αποκοπή φίλτρου. Αυτή η δυνατότητα επιτρέπει στο AD8232 να ανακάμψει γρήγορα και, επομένως, να λάβει έγκυρες μετρήσεις αμέσως μετά τη σύνδεση των ηλεκτροδίων στο ανθρώπινο σώμα. Αυτός ο σχεδιασμός επιτρέπει σε έναν μετατροπέα αναλογικού σε ψηφιακό (ADC) χαμηλής ισχύος ή σε έναν μικροελεγκτή, την εύκολη απόκτηση του σήματος εξόδου [15].

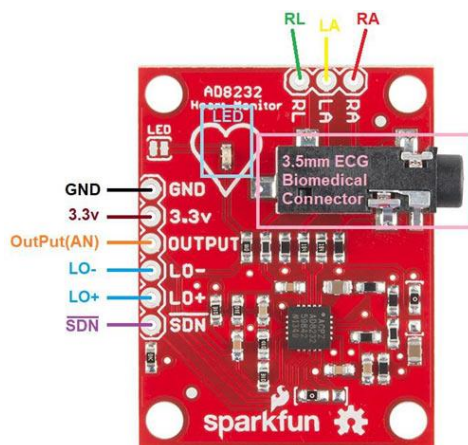


Διάγραμμα 3-3: Το λειτουργικό μπλοκ διάγραμμα της πλακέτας AD8232

Τα εξαρτήματα της μονάδας ΗΚΓ AD8232 μπορούν να χωριστούν στις ακόλουθες κατηγορίες [16].

- Pins εισόδου ισχύος (3.3, GND)
- Pins σύνδεσης pad ηλεκτροδίου (RA, LA, RL)
- Pin εξόδου δεδομένων (OUTPUT)
- Ακροδέκτες εξόδου ανίχνευσης (LO-, LO+)
- Pin ελέγχου τερματισμού λειτουργίας (~SDN)
- LED

Το παρακάτω σχήμα δείχνει το διάγραμμα pinout της μονάδας ΗΚΓ. Από αυτό το IC συνολικά εννέα ακίδες έχουν εκτεθεί στο pinout και παρέχονται με τη μορφή μονάδας ECG AD8232 για φιλική προς το χρήστη διεπαφή και κατάλληλη προστασία ασφάλειας στο IC.

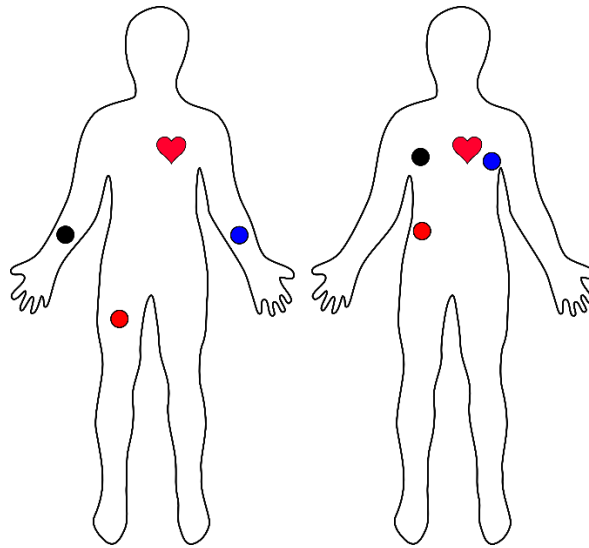


Σχήμα 3-1: Το διάγραμμα των pin της πλακέτας AD8232

Τα pins σύνδεσης των ηλεκτροδίων του ΗΚΓ είναι τα ακόλουθα [17].

- Το LA είναι η θετική είσοδος (+IN) του ενισχυτή οργάνων του IC AD8232. Εδώ λαμβάνεται ένα σήμα από το ηλεκτρόδιο που συνδέεται με τον αριστερό βραχίονα ή το αριστερό τμήμα του στήθους του ανθρώπινου σώματος.
- Το RA είναι η αρνητική είσοδος (-IN) του ενισχυτή οργάνων του IC AD8232. Εδώ λαμβάνεται ένα σήμα από το ηλεκτρόδιο που είναι συνδεδεμένο με τον δεξιό βραχίονα ή το δεξί τμήμα του στήθους του ανθρώπινου σώματος.
- Το RL είναι ένα πράσινο ή κόκκινο βιοϊατρικό ηλεκτρόδιο που λειτουργεί ως είσοδος ηλεκτροδίου και συνδέεται με το δεξί πόδι ή το κάτω δεξί μέρος της κοιλιακής χώρας του ανθρώπινου σώματος.

Όσο πιο κοντά στην καρδιά είναι τα pads, τόσο καλύτερη είναι η μέτρηση. Τα καλώδια έχουν χρωματική κωδικοποίηση για να βοηθήσουν στον εντοπισμό της σωστής τοποθέτησης όπως φαίνεται στην εικόνα με βάση το τρίγωνο του Einthoven.



Εικόνα 3-5: Τυπική τοποθέτηση αισθητήρων του AD8232

Τα βασικά χαρακτηριστικά του AD8232 είναι [18]:

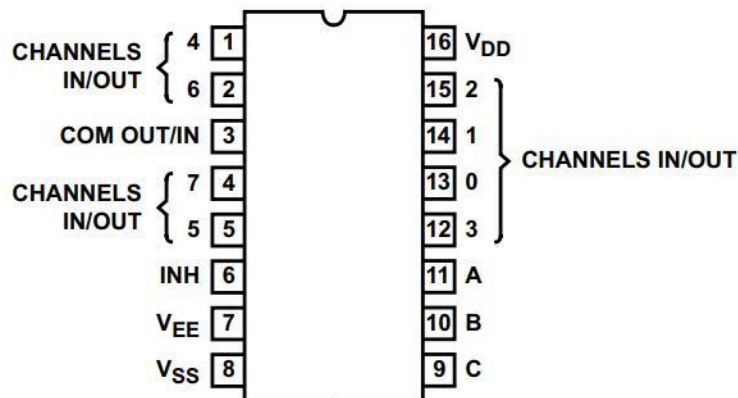
- Τάση λειτουργίας: 2,5 έως 3,3 VDC.
- Λόγος απόρριψης κοινής λειτουργίας: 80 dB (dc έως 60 Hz).
- Η λειτουργία γρήγορης επαναφοράς βελτιώνει την καθίζηση του φίλτρου.
- Ρεύμα λειτουργίας: 170 μ A.
- Ηλεκτροστατική εκκένωση για το μοντέλο του ανθρώπινου σώματος: 8 kV.
- Εκτιμώμενο εύρος θερμοκρασίας: 0 έως 70 βαθμούς Κελσίου.
- Εύρος θερμοκρασίας λειτουργίας από -40 έως 85 βαθμούς Κελσίου.
- Διαθέτει Pin τερματισμού λειτουργίας που μπορεί να χρησιμοποιηθεί για είσοδο σε λειτουργία εξοικονόμησης ενέργειας.
- Πλήρως ενσωματωμένο μπροστινό άκρο ΗΚΓ μονής απαγωγής.
- Διαμορφώσεις δύο ή τριών ηλεκτροδίων.
- Διαστάσεις: 3,5cm x 3cm.

3.3 CD4051 Αναλογικός πολυπλέκτης 8 καναλιών

Το chip CD4051 είναι ένας πολυπλέκτης οκτώ (8) καναλιών και έχει τρεις ψηφιακές εισόδους ελέγχου που ονομάζονται A, B και C. Έτσι, αναφέρονται συνήθως ως ψηφιακός ελεγχόμενος αναλογικός διακόπτης. Αυτές οι εισοδοί συνδέουν μόνο 1 από τα 8 κανάλια στην έξοδο για να λάβουν την επιθυμητή έξοδο. Οι ακροδέκτες εισόδου/εξόδου καναλιού γίνονται έξοδοι και οι κοινοί ακροδέκτες εισόδου/εξόδου γίνονται ακροδέκτες εισόδου όταν το CD 4051 χρησιμοποιείται ως αποπολυπλέκτης. Το CD-4051 έχει πολλά εκπληκτικά χαρακτηριστικά π.χ. Η αποκωδικοποίηση δυαδικής διεύθυνσης στο τσιπ, η διακοπή πριν από την εναλλαγή εξαλείφει την επικάλυψη καναλιών, το ευρύ φάσμα αναλογικών σημάτων καθώς και τα επίπεδα ψηφιακού σήματος. Το CD-4051 μπορεί να χρησιμοποιηθεί σε πύλη σήματος, μετατροπέα αναλογικού σε ψηφιακό, μετατροπέα ψηφιακού σε αναλογικό, αναλογικής και ψηφιακής πολυπλεξίας και αποπολυπλεξίας. Η πιο σημαντική όμως τιμή είναι η μέγιστη τάση εισόδου που είναι 5 V για τον αναλογικό πολυπλέκτη επειδή τα 5V ταιριάζουν καλά με όλους τους μικροελεγκτές Arduino και EPS8266 [19].



Εικόνα 3-6: Ο πολυπλέκτης CD4051



Διάγραμμα 3-4: Το διάγραμμα των Pin του CD4051

Όπως είναι φανερό στο παραπάνω διάγραμμα pinout του CD4051 παρατηρούμε ότι περιλαμβάνει:

- 3 Pins εισόδου (A,B και C).
- 8 Pins I/O.
- 3 Pins τροφοδοσίας (VDD, VEE και VSS).
- Το pin # 6 είναι Inhibit.
- Το Pin # 3 είναι το Pin O/I.

Η απαίτηση ισχύος μιας συσκευής μπορεί να εκτιμηθεί από τις αξιολογήσεις της συγκεκριμένης συσκευής. Οι παράμετροι λειτουργίας του CD4051 δίνονται στον πίνακα που φαίνεται στην παρακάτω εικόνα.

CD4051 Ratings		
Parameters	Values	Units
Voltage supply	-0.5 to 20	V
DC Input voltage	-0.5 to $V_{DD}+0.5$	V
Junction temperature	265	°C
Storage Temperature	-65 to 150	°C
DC Input current	-10 to 10	mA

Πίνακας 3-2: Οι βασικές προδιαγραφές του CD4051

Οι παραπάνω προδιαγραφές κρίθηκαν σημαντικές για την επιλογή του πολυπλέκτη CD4051 εφόσον:

- 1) Η μέγιστη τάση εισόδου είναι 5 V για τον αναλογικό πολυπλέκτη και μπορεί να τροφοδοτηθεί από το Pin Vin του μικροελεγκτή ESP8266.
- 2) Η διακοπή BBM (Break Before Make) πριν από την εναλλαγή εξαλείφει την επικάλυψη καναλιών.
- 3) Επιτρέπει την αύξηση των αναλογικών εισόδων στην είσοδο του A0 στον μικροελεγκτή (8 πιθανά I/O), δεδομένο σημαντικό για την παρούσα εργασία, επειδή ο μικροελεγκτής περιλαμβάνει μια αναλογική είσοδο ενώ εμείς χρειαζόμαστε δύο.
- 4) Είναι μία πολύ οικονομική μέθοδος.

3.4 NTC Thermistor 10kΩ 3977K

Τα θερμίστορ είναι στοιχεία ανίχνευσης θερμοκρασίας κατασκευασμένα από ημιαγωγό υλικό που έχει υποστεί πυροσυσσωμάτωση προκειμένου να εμφανίζει μεγάλες αλλαγές στην αντίσταση σε αναλογία με μικρές αλλαγές στη θερμοκρασία. Ο θερμίστορ αποτελείται από ένα τσιπ με δύο κονσερβοποιημένα στερεά επιχάλκωμένα καλώδια. Είναι γκρι λακαρισμένο και χρωματικά κωδικοποιημένο, αλλά όχι μονωμένο. Αυτή η αντίσταση μπορεί να μετρηθεί χρησιμοποιώντας ένα μικρό και μετρούμενο συνεχές ρεύμα, ή dc, που διέρχεται από το θερμίστορ για να μετρηθεί η παραγόμενη πτώση τάσης. Αυτοί οι αισθητήρες θερμοκρασίας στερεάς κατάστασης στην πραγματικότητα λειτουργούν σαν ηλεκτρικές αντιστάσεις που είναι ευαίσθητες στη θερμοκρασία.



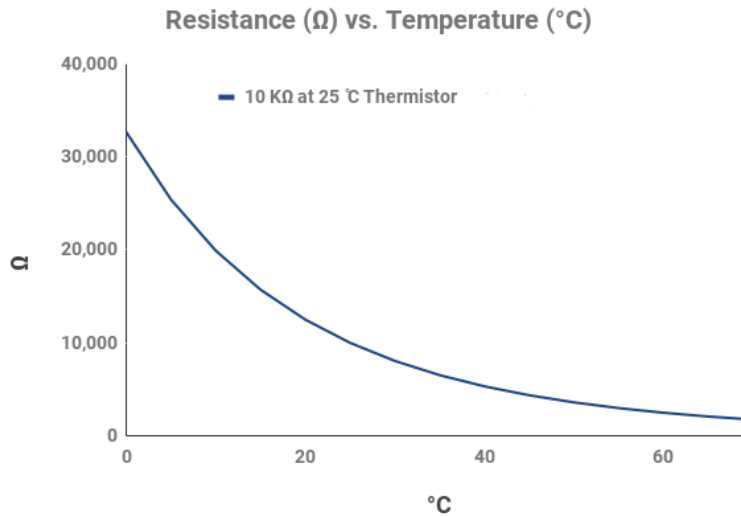
Εικόνα 3-7: Ο NTC θερμίστορ 10kΩ

Τα θερμίστορ NTC είναι μη γραμμικές αντιστάσεις, οι οποίες αλλάζουν τα χαρακτηριστικά αντιστάσεώς τους με τη θερμοκρασία. Η αντίσταση του NTC θα μειωθεί όσο αυξάνεται η θερμοκρασία. Ο τρόπος με τον οποίο μειώνεται η αντίσταση σχετίζεται με μια σταθερά γνωστή στη βιομηχανία ηλεκτρονικών ως Beta ή B. Το βήτα μετριέται σε °K. Μπορούμε εύκολα να υπολογίσουμε την αντίσταση των θερμίστορ NTC σε μια δεδομένη θερμοκρασία χρησιμοποιώντας την B, αλλά υπάρχει ένας ακόμη πιο ακριβής τρόπος για να το πετύχουμε αυτό χρησιμοποιώντας την εξίσωση Steinhart & Hart. Στην συγκεκριμένη εργασία θα χρησιμοποιήσουμε την εξίσωση B παραμέτρου. Τα θερμίστορ NTC είναι επίσης μια εξαιρετική εναλλακτική λύση στα κυκλώματα ημιαγωγών για την επίλυση προβλημάτων που σχετίζονται με τη θερμοκρασία. Μπορούν εύκολα να χρησιμοποιηθούν για τον υπολογισμό του συντελεστή θερμοκρασίας [20].

Στις παρακάτω εικόνες φαίνονται η εξίσωση παραμέτρων B και το διάγραμμα καμπύλης μεταξύ αντίστασης και θερμοκρασίας.

$$R = R_0 e^{-B \left(\frac{1}{T_0} - \frac{1}{T} \right)} \quad \frac{1}{T} = \frac{1}{T_0} + \frac{1}{\beta} \cdot \ln \frac{R_T}{R_0}$$

Εικόνα 3-8: Μαθηματική εξίσωση παραμέτρων Beta



Διάγραμμα 3-5: Η καμπύλη αλλαγής της τιμής αντίστασης του θερμίστορ 10kΩ σε σχέση με την θερμοκρασία.

Οι παράμετροι λειτουργίας του θερμίστορ με Βετα 3977 παρατίθενται στον παρακάτω πίνακα [21].

QUICK REFERENCE DATA		
PARAMETER	VALUE	UNIT
Resistance value at 25 $^{\circ}\text{C}$	3.3 to 470K	Ω
Tolerance on R_{25} -value	± 2 ; ± 3 ; ± 5	%
$B_{25/85}$ -value	2880 to 4570	K
Tolerance on $B_{25/85}$ -value	± 0.5 to ± 3	%
Operating temperature range: At zero power dissipation; continuously	- 40 to + 125	$^{\circ}\text{C}$
At zero power dissipation; for short periods	≤ 150	
Response time (in oil)	≈ 1.2	s
Thermal time constant τ (for information only)	15	s
Dissipation factor δ (for information only)	7 8.5 (for R_{25} -value $\leq 680 \Omega$)	mW/K
Maximum power dissipation at 55 $^{\circ}\text{C}$	500	mW
Climatic category (LCT/UCT/days)	40/125/56	-
Weight	≈ 0.3	g

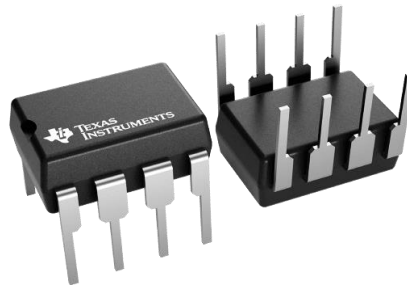
Πίνακας 3-3: Οι βασικές προδιαγραφές του thermistor 10kΩ

Τα βασικά χαρακτηριστικά του θερμίστορ 10kΩ είναι [22]:

- Ακρίβεια σε μεγάλο εύρος θερμοκρασίας
- Υψηλή σταθερότητα για μεγάλη διάρκεια ζωής
- Χαρακτηρίζεται από αντίσταση και ανοχή
- Εξαιρετική σχέση τιμής/απόδοσης

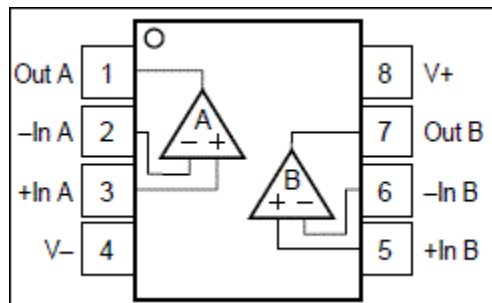
3.5 OPA2340 Rail-to-Rail ενισχυτής

Οι λειτουργικοί ενισχυτές Rail-to-Rail CMOS της σειράς OPA340 είναι βελτιστοποιημένοι για λειτουργία χαμηλής τάσης και μίας παροχής. Η είσοδος και η έξοδος Rail-to-Rail και η λειτουργία υψηλής ταχύτητας τα καθιστούν ιδανικά για την οδήγηση δειγματοληψίας μετατροπέων αναλογικού σε ψηφιακό (A/D). Είναι επίσης κατάλληλα για εφαρμογές γενικής χρήσης και ήχου, καθώς και για μετατροπή I/V στην έξοδο μετατροπέων ψηφιακού σε αναλογικό (D/A).



Εικόνα 3-9: Ο ενισχυτής OPA2340

Οι σειρές OPA340 λειτουργούν με μία μόνο παροχή από 2,5 V με εύρος τάσης εισόδου κοινής λειτουργίας που εκτείνεται 500 mV κάτω από το έδαφος και 500 mV πάνω από τη θετική παροχή. Η αιώρηση της τάσης εξόδου είναι εντός 1 mV από τις ράγες τροφοδοσίας με φορτίο 100 kΩ. Αυτές οι συσκευές προσφέρουν εξαιρετική δυναμική απόκριση ($BW = 5,5 \text{ MHz}$, $SR = 6 \text{ V}/\mu\text{s}$). Οι διπλοί και οι τετραπλοί σχεδιασμοί διαθέτουν εντελώς ανεξάρτητα κυκλώματα για χαμηλότερη αλληλεπίδραση και ελευθερία από αλληλεπίδραση. Όλα προδιαγράφονται από -40°C έως 85°C και λειτουργούν από -55°C έως 125°C . Ένα μακρομοντέλο SPICE είναι διαθέσιμο για ανάλυση σχεδιασμού [23].



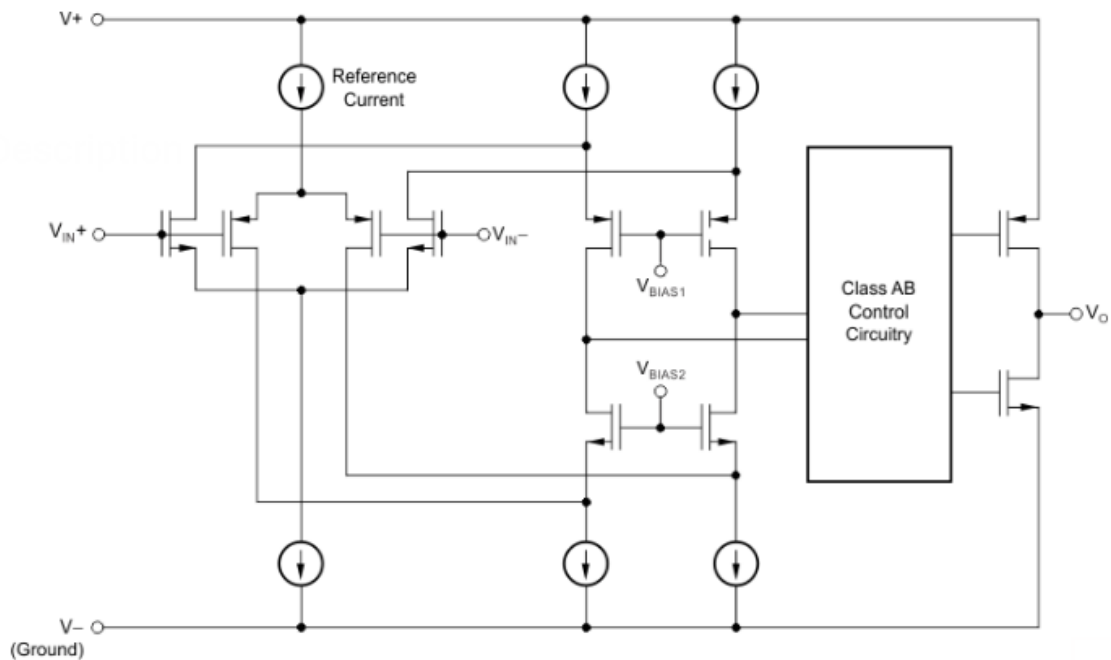
Διάγραμμα 3-6: Το διάγραμμα των Pin του OPA2340

Οι ακίδες του ενισχυτή OPA2340 με βάση το Διάγραμμα 3-6 διαμορφώνονται ως εξής [24]:

- - IN A - Αρνητικό (αντιστρέφον) κανάλι εισόδου A.
- +IN A - θετικό (αντιστρέφον) κανάλι εισόδου A.
- - IN B - Αρνητικό (αντιστρέφον) κανάλι εισόδου B.
- +IN B - θετικό (αντιστρέφον) κανάλι εισόδου B.
- OUT A - κανάλι εξόδου A.
- OUT B - κανάλι εξόδου B.
- V- - αρνητική τροφοδοσία ισχύς.
- V+ - θετική τροφοδοσία ισχύς.

Τα βασικά χαρακτηριστικά του ενισχυτή είναι [25]:

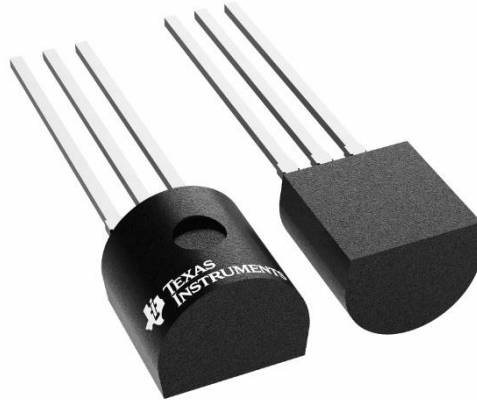
- Είσοδος Rail-to-Rail.
- Έξοδος Rail-to-Rail (εντός 1 mV).
- Πακέτα MicroSize.
- Ευρύ Bandwidth: 5,5 MHz.
- Υψηλός ρυθμός περιστροφής (Slew Rate): 6 V/μs.
- Χαμηλό THD + Θόρυβος: 0,0007% (f = 1 kHz).
- Χαμηλό ρεύμα ηρεμίας: 750 μΑ/Κανάλι.
- Εκδόσεις Single, Dual και Quad.



Διάγραμμα 3-7: Το μπλοκ διάγραμμα του OPA2340

3.6 Διαιρέτης TLE2426

Σε εφαρμογές συντήρησης σήματος που χρησιμοποιούν μία μόνο πηγή ισχύος, απαιτείται τάση αναφοράς ίση με το μισό της τάσης τροφοδοσίας για τον τερματισμό όλων των γειώσεων αναλογικού σήματος. Η Texas Instruments παρουσιάζει μια γείωση ακριβείας της οποίας η τάση εξόδου είναι πάντα ίση με το μισό της τάσης εισόδου, τον "διαιρέτη τάσεων" TLE2426.



Εικόνα 3-10: Ο διαιρέτης TLE2426

Ο μοναδικός συνδυασμός υψηλής απόδοσης, ενός λειτουργικού ενισχυτή ακριβείας και ένας διαιρέτης σε ένα chip σιλικόνης έχει ως αποτέλεσμα ακριβή λόγο V_O/V_I ίσο με 0,5 ενώ τροφοδοτείται με ρεύμα. Το TLE2426 παρέχει χαμηλή έξοδο σύνθετης αντίστασης με ικανότητα πηγής 20 mA ενώ αντλεί λιγότερο από 280 μ A ρεύματος τροφοδοσίας σε όλο το εύρος εισόδου από 4V έως 40V. Ένας σχεδιαστής δεν χρειάζεται να πληρώσει το τίμημα όσον αφορά τον χώρο της πλακέτας για μια συμβατική γείωση σήματος που αποτελείται από αντιστάσεις, πυκνωτές, λειτουργικούς ενισχυτές και αναφορές τάσης. Η απόδοση και η ακρίβεια του TLE2426 είναι διαθέσιμη σε ένα εύχρηστο πακέτο LP με 3 ακροδέκτες που εξοικονομεί χώρο [26].

**LP PACKAGE
(TOP VIEW)**



Διάγραμμα 3-8: Το διάγραμμα των Pin του TLE2426

Από το παραπάνω διάγραμμα Pinout του διαιρέτη TLE2426 έχουμε:

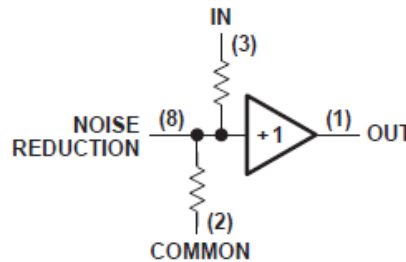
- **IN** – Pin τροφοδοσίας.
- **COMMON** - Pin γείωσης τάσης τροφοδοσίας.
- **OUT** - Pin εξόδου τάσης.

Οι παράμετροι λειτουργίας του TLE2426 δίνονται στον παρακάτω πίνακα.

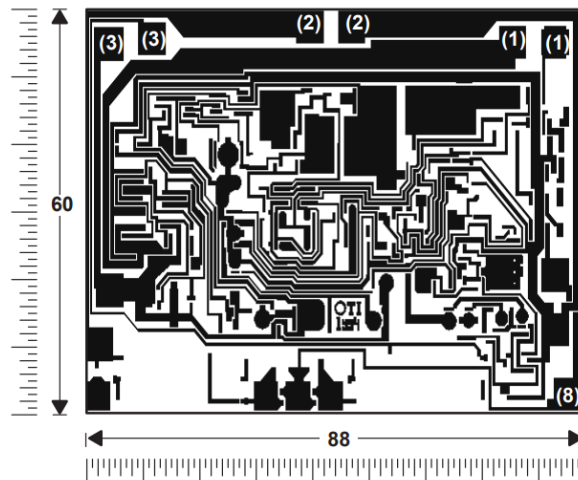
Output current (Τυπ) (mA)	47
Vs (Min) (V)	4
Vs (Max) (V)	40
Operating temperature range (C)	-55 to 125

Πίνακας 3-4: Οι παράμετροι λειτουργίας του TLE2426

Στα επόμενα διαγράμματα παρουσιάζονται το μπλοκ διάγραμμα του διαιρέτη και η αντίστοιχη πλακέτα.



Διάγραμμα 3-9: Το μπλοκ διάγραμμα του TLE2426



Σχήμα 3-2: Το διάγραμμα της πλακέτας TLE2426

ΚΕΦΑΛΑΙΟ 4. ΠΛΑΤΦΟΡΜΕΣ ΑΝΑΠΤΥΞΗΣ ΛΟΓΙΣΜΙΚΟΥ (SOFTWARE)

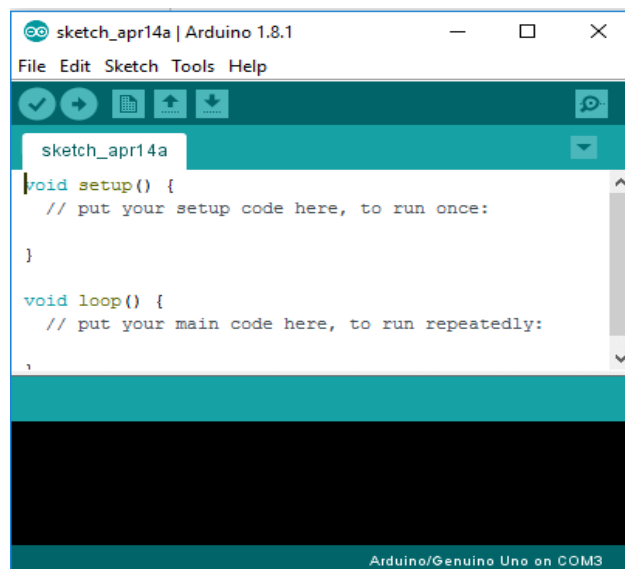
Το λογισμικό, με την πιο γενική του έννοια, είναι ένα σύνολο οδηγιών ή προγραμμάτων που καθοδηγούν έναν υπολογιστή να κάνει συγκεκριμένες εργασίες. Το λογισμικό είναι ένας γενικός όρος που χρησιμοποιείται για να περιγράψει προγράμματα υπολογιστών. Έτσι και στην παρούσα εργασία χρειαζόμαστε πλατφόρμες με τις οποίες θα είναι δυνατή η λήψη των δεδομένων μέσω του μικροελεγκτή και η εμφάνισή τους σε βάσεις δεδομένων, με την χρήση συγκεκριμένων εντολών.

4.1 Παρουσίαση πακέτων λογισμικού

Παρακάτω παρουσιάζονται οι πλατφόρμες ανάπτυξης λογισμικού οι οποίες θα χρησιμοποιηθούν στην παραγωγή των αποτελεσμάτων λειτουργίας της συσκευής καθώς και της εμφάνισής τους στο διαδίκτυο μέσω Wi-Fi.

4.1.1 Arduino IDE

Το Arduino IDE, είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (integrated development environment), που περιέχει έναν επεξεργαστή πηγαίου κώδικα, έναν μεταγλωττιστή, εργαλεία αυτόματης παραγωγής κώδικα, αποσφαλματωτή, συνδέτη, σύστημα ελέγχου εκδόσεων και εργαλεία κατασκευής γραφικών διασυνδέσεων χρήστη για τις υπό ανάπτυξη εφαρμογές. Είναι μια βιβλιοθήκη λογισμικού ανοιχτού κώδικα και συνδέεται με το hardware μέρος του Arduino/ESP8266 για να φορτώσει προγράμματα και να επικοινωνεί μαζί τους. Με λίγα λόγια είναι το πρόγραμμα που χρησιμοποιείται για τη σύνταξη του κώδικα και διατίθεται με τη μορφή αρχείου με δυνατότητα λήψης στον ιστότοπό του Arduino.. Ο κώδικας που έχει γραφεί για το Arduino ονομάζεται sketch [27].








Εικόνα 4-1: Το περιβάλλον του Arduino IDE

Το περιβάλλον IDE κατανέμεται κυρίως σε τρεις ενότητες:

- 1) Menu bar - Μπάρα Μενού
- 2) Text editor - Χώρος επεξεργασίας κειμένου
- 3) Output pane - Παράθυρο εξόδου

Στον ακόλουθο πίνακα παρουσιάζονται τα εργαλεία του περιβάλλοντος ανάπτυξης, υπό μορφή κουμπιών.

Εργαλεία ανάπτυξης Arduino IDE	
Εργαλείο	Περιγραφή
	Ελέγχει για συντακτικά λάθη στον κώδικα.
	Μεταγλωττίζει τον κώδικα και το φορτώνει στο Arduino. Αν δεν είναι συντακτικά σωστός δε μπορεί να γίνει η φόρτωση.
	Δημιουργεί ένα νέο sketch.
	Παραθέτει ένα μενού με όλα τα sketch. Ενεργοποιώντας ένα από αυτά, θα ανοίξει αυτόματα στο τρέχον παράθυρο.
	Αποθηκεύει ένα sketch.

Πίνακας 4-1: Τα βασικά εργαλεία του περιβάλλοντος ανάπτυξης

Το Arduino IDE είναι βασισμένο σε Java και συγκεκριμένα παρέχει:

- Ένα πρακτικό περιβάλλον για τη συγγραφή των προγραμμάτων, με συντακτική χρωματική σήμανση.
- Βιβλιοθήκες για προέκταση της, εύκολες στην πρόσβαση μέσω διαδικτύου.
- Compiler (μεταγλωττιστής) για τη μεταγλώττιση των sketch.
- Μία σειριακή οθόνη (serial monitor) που παρακολουθεί τις επικοινωνίες της σειριακής (USB), αναλαμβάνει να στείλει αλφαριθμητικά στο Arduino/ESP8266 μέσω αυτής και είναι ιδιαίτερα χρήσιμο για τον προσδιορισμό τυχόν λαθών των sketch (σεναρίων).
- Την επιλογή για μεταφορά των μεταγλωττισμένων sketch στο Arduino.

Η γλώσσα συγγραφής κώδικα του Arduino, βασίζεται στη γλώσσα Wiring, μια παραλλαγή C/C++ για μικροελεγκτές αρχιτεκτονικής AVR όπως ο ATmega, και υποστηρίζει όλες τις βασικές δομές της C, καθώς και μερικά χαρακτηριστικά της C++. Για compiler χρησιμοποιείται ο AVR-GCC και ως βασική βιβλιοθήκη C χρησιμοποιείται η AVR libc. Ο AVR-GCC είναι ένας μεταγλωττιστής που παίρνει κώδικα υψηλού επιπέδου στη γλώσσα C και δημιουργεί μια δυαδική πηγή που μπορεί να μεταφορτωθεί σε έναν μικροελεγκτή AVR. Έτσι, το AVR-GCC μπορεί να θεωρηθεί ως cross compiler «C» για την παραγωγή κώδικα AVR [28].

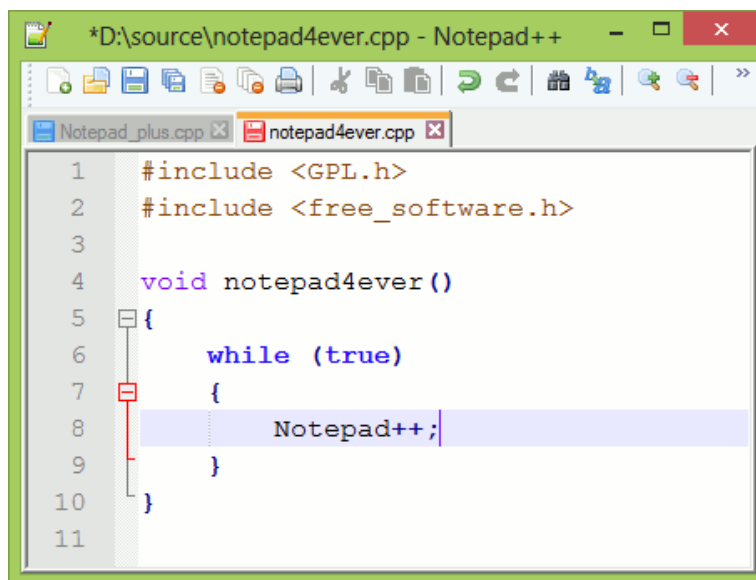
Λόγω της καταγωγής της από τη C, στη γλώσσα συγγραφής κώδικα του Arduino, μπορούν να χρησιμοποιηθούν ουσιαστικά οι ίδιες βασικές εντολές και συναρτήσεις, με την ίδια σύνταξη, τους ίδιους τύπων δεδομένων και τους ίδιους τελεστές όπως και στη C. Πέρα από αυτές όμως, υπάρχουν κάποιες ειδικές εντολές, συναρτήσεις και σταθερές που βοηθούν για τη διαχείριση του ειδικού hardware του Arduino/ESP8266.

Κάθε sketch εκτελείται μέσα από τον ορισμό δυο συναρτήσεων:

- ✓ Setup(): Η συγκεκριμένη συνάρτηση καλείται όταν ξεκινάει ένα sketch. Χρησιμοποιείται για αρχικοποίηση μεταβλητών, ορισμό ακροδεκτών και για την εισαγωγή βιβλιοθηκών. Η Setup συνάρτηση εκτελείται μόνο μια φορά, κατά την παροχή τροφοδοσίας ή μετά από reset.
- ✓ Loop(): Μετά την Setup συνάρτηση, ακολουθεί η Loop (βρόχος) συνάρτηση, η οποία κάνει ακριβώς αυτό που δηλώνει το όνομα της, δηλαδή εκτελείται διαδοχικούς «κύκλους», επιτρέποντας στο πρόγραμμα να αλλάζει και να ανταποκρίνεται. Χρησιμοποιείται για να ελέγχεται η πλακέτα Arduino.

4.1.2 Notepad ++

Η πλατφόρμα λογισμικού, Notepad++, είναι ένας επεξεργαστής πηγαίου κειμένου (source code editor). Χρησιμοποιείτε για την ανάπτυξη όλων των script σε php, css, JavaScript, html, MySQL, Ajax. Πρόκειται για ανοιχτού λογισμικού πρόγραμμα, το οποίο εκτός από τις παραπάνω μπορεί να χρησιμοποιηθεί για τη συγγραφή και άλλων γλωσσών προγραμματισμού. Το "++" στο όνομα είναι μια αναφορά στον τελεστή αύξησης σε γλώσσες προγραμματισμού όπως C, C++, Java και JavaScript [29].



```
*D:\source\notepad4ever.cpp - Notepad++
Notepad_plus.cpp x notepad4ever.cpp x
1 #include <GPL.h>
2 #include <free_software.h>
3
4 void notepad4ever ()
5 {
6     while (true)
7     {
8         Notepad++;
9     }
10 }
11
```

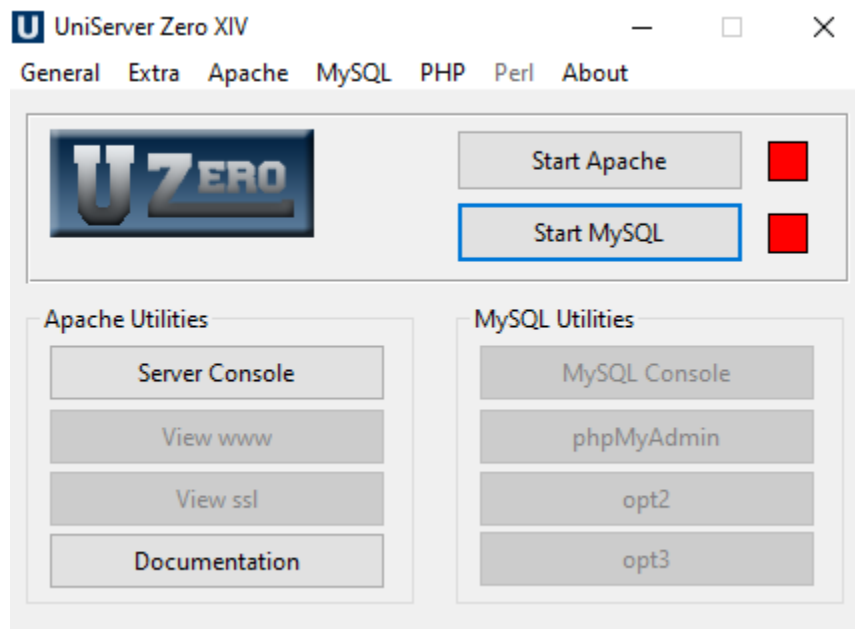
Εικόνα 4-2: Το περιβάλλον του Notepad ++

Τα βασικά χαρακτηριστικά του Notepad ++ είναι:

- Επεξεργασία αρχεία κειμένου μεγέθους έως 2 GB (το μέγιστο μέγεθος στο Σημειωματάριο των Windows είναι 58 MB).
- Επεξεργασία πολλών αρχείων και οργάνωση σε καρτέλες.
- Αρίθμηση γραμμών.
- Επισήμανση σύνταξης για περισσότερες από 70 γλώσσες προγραμματισμού, συμπεριλαμβανομένων ομαδικών αρχείων HTML και Windows.
- Προηγμένη εύρεση και αντικατάσταση, με υποστήριξη για κανονικές εκφράσεις.
- Δημιουργία και επεξεργασία αρχείων κειμένου για διαφορετικά λειτουργικά συστήματα, συμπεριλαμβανομένων των macOS και Linux.
- Διαίρεση οθόνης για επεξεργασία και προβολή πολλών αρχείων ταυτόχρονα ή πολλαπλών τμημάτων του ίδιου αρχείου.
- Μακροεντολές για την εγγραφή μιας ακολουθίας εντολών επεξεργασίας που πρέπει να εκτελούνται επανειλημμένα.

4.1.3 UniServer Zero XIV

Ο UniServer Zero XIV είναι ένα πακέτο προγραμμάτων ελεύθερου λογισμικού για Windows, περιέχει εξυπηρετητή ιστοσελίδων http Apache, βάση ιστοσελίδων MySQL και ένα διερμηνέα για σενάρια γραμμένα σε γλώσσες προγραμματισμού PHP και Perl. Ο τοπικός διακομιστής UniServer, χρησιμοποιείτε για την ανάπτυξη του ιστοχώρου με την απεικόνιση που προσφέρει. Πρόκειται για ανοιχτού λογισμικού πρόγραμμα και προσομοιώνει τη φιλοξενία ενός ιστοχώρου σε web server (διακομιστής διαδικτύου) [30].



Εικόνα 4-3: Το περιβάλλον του XAMPP

Μερικά από τα πλεονεκτήματα του UniServer παρουσιάζονται στην συνέχεια.

- Σε σύγκριση με άλλους διακομιστές ιστού (WAMP) όπως το XAMPP, δεν απαιτείται εγκατάσταση του προγράμματος στον υπολογιστή.
- Είναι εύκολος ως προς την ρύθμισή του και την λειτουργία του.
- Με μία μόνο εντολή, μπορεί να ξεκινήσει και να σταματήσει ολόκληρη η στοίβα διακομιστή web και βάσης δεδομένων.
- Η έκδοση του UniServer διατίθεται δωρεάν στο διαδίκτυο.
- Διαθέτει έναν πίνακα ελέγχου που μπορείτε να δείτε ότι περιέχει κουμπιά έναρξης και διακοπής για συγκεκριμένους μηχανισμούς, όπως το Apache, το οποίο εκτελείται μέσω του Πίνακα Ελέγχου του.
- Περιλαμβάνει επίσης OpenSSL, phpMyAdmin, WordPress και πολλές πρόσθετες ενότητες.

4.1.4 phpMyAdmin

Το phpMyAdmin είναι ένα δωρεάν εργαλείο λογισμικού γραμμένο σε PHP, που προορίζεται να χειριστεί τη διαχείριση της MySQL μέσω του Web. Το phpMyAdmin υποστηρίζει ένα ευρύ φάσμα λειτουργιών σε MySQL και MariaDB. Οι λειτουργίες που χρησιμοποιούνται συχνά (διαχείριση βάσεων δεδομένων, πίνακες, στήλες, σχέσεις, ευρετήρια, χρήστες, δικαιώματα, κ.λπ.) μπορούν να εκτελεστούν μέσω της δι' επαφής χρήστη, ενώ εξακολουθεί να υπάρχει τη δυνατότητα να εκτελεστεί απευθείας οποιαδήποτε πρόταση SQL. Για να διευκολύνεται η χρήση σε ένα ευρύ φάσμα ατόμων, το phpMyAdmin μεταφράζεται σε 72 γλώσσες και υποστηρίζει γλώσσες LTR και RTL [31].

Τα βασικά χαρακτηριστικά του phpMyAdmin είναι:

- Διασθητικό web interface.
- Υποστήριξη για τις περισσότερες δυνατότητες της MySQL:
 - περιήγηση και αποθήκευση βάσεων δεδομένων, πινάκων, προβολών, πεδίων και ευρετηρίων.
 - δημιουργία, αντιγραφή, αποθήκευση, μετονομασία και αλλαγή βάσεων δεδομένων, πινάκων, πεδίων και ευρετηρίων.
 - διακομιστή συντήρησης, βάσεις δεδομένων και πίνακες, με προτάσεις για τη διαμόρφωση διακομιστή.
 - εκτέλεση, επεξεργασία και προσθήκη σελιδοδεικτών σε οποιαδήποτε δήλωση SQL, ακόμη και ερωτήματα δέσμης.
 - διαχείριση λογαριασμών και προνομίων χρηστών MySQL.
 - διαχείριση αποθηκευμένων διαδικασιών και ενεργειών.
- Εισαγωγή δεδομένων από CSV και SQL.
- Εξαγωγή δεδομένων σε διάφορες μορφές.
- Διαχείριση πολλών διακομιστών.
- Δημιουργία γραφικών της διάταξης της βάσης δεδομένων σε διάφορες μορφές.
- Καθολική αναζήτηση σε μια βάση δεδομένων ή ένα υποσύνολο αυτής.
- Μετατροπή αποθηκευμένων δεδομένων σε οποιαδήποτε μορφή χρησιμοποιώντας ένα σύνολο προκαθορισμένων λειτουργιών, όπως η εμφάνιση δεδομένων BLOB ως εικόνα ή σύνδεσμος λήψης

4.2 Γλώσσες Προγραμματισμού & Server

Για το σχεδιασμό και τη σύσταση του ιστοχώρου, επιλέγονται οι γλώσσες προγραμματισμού ή περιεχομένου που μπορούν να φέρουν εις πέρας ένα τέτοιο έργο και έπειτα η βάση δεδομένων στην οποία θα υπάρχουν όλες οι δυναμικές πληροφορίες, που απαιτεί ο ιστοχώρος για την ορθή λειτουργία του.

4.2.1 HTML

Η λέξη HTML ορίζεται από τα αρχικά των λέξεων HyperText Markup Language και αποτελεί μια γλώσσα υπολογιστή σχεδιασμένη για να επιτρέψει τη δημιουργία μίας ιστοσελίδας. Οι ιστοσελίδες αυτές μετά την δημιουργία τους και την εισαγωγή τους στο διαδίκτυο μπορούν να είναι προσβάσιμες από οποιονδήποτε χρήστη συνδεδεμένο σε αυτό. Η γλώσσα αυτή αποτελεί ένα σύνολο κανόνων για την διαμόρφωση της εμφάνισης και του περιεχομένου μιας ιστοσελίδας. Ουσιαστικά, δεν μπορούμε να την ορίσουμε ως γλώσσα προγραμματισμού, αλλά γλώσσα περιγραφής ιδιοτήτων των στοιχείων που την αποτελούν [32].



Εικόνα 4-4: Το λογότυπο της HTML

Η HTML γράφεται υπό μορφή στοιχείων HTML τα οποία αποτελούνται από ετικέτες (tags), οι οποίες περικλείονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (για παράδειγμα <html>), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα <h1> και </h1>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης (ή σε άλλες περιπτώσεις ετικέτα ανοίγματος και ετικέτα κλεισίματος αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κλπ. Οι οδηγίες είναι case insensitive, δηλαδή, δεν επηρεάζονται από το αν έχουν γραφτεί με πεζά (μικρά) ή κεφαλαία και ένα αρχείο HTML πρέπει να έχει κατάληξη htm ή html.

Τα στοιχεία της HTML χρησιμοποιούνται για το χτίσιμο όλων των ιστότοπων. Η HTML επιτρέπει την ενσωμάτωση εικόνων και άλλων αντικειμένων μέσα στη σελίδα, και μπορεί να χρησιμοποιηθεί για να εμφανίσει διαδραστικές φόρμες. Παρέχει τις μεθόδους δημιουργίας δομημένων εγγράφων (δηλαδή εγγράφων που αποτελούνται από το περιεχόμενο που μεταφέρουν και από τον κώδικα μορφοποίησης του περιεχομένου), καθορίζοντας δομικά σημαντικά στοιχεία για το κείμενο, όπως κεφαλίδες, παραγράφους,

λίστες, συνδέσμους, παραθέσεις. Μπορούν επίσης, να ενσωματώνονται σενάρια εντολών σε γλώσσες όπως η JavaScript, τα οποία επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML.

Παράδειγμα της βασικής δομής ενός html αρχείου παρατίθεται παρακάτω.

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta charset="UTF-8">
5      <title>Title goes here</title>
6    </head>
7    <body>
8
9    </body>
10 </html>
```

Εικόνα 4-5: Παράδειγμα σύνταξης HTML

Όπως ήδη προαναφέραμε με τα tags <html> </html> αρχίζει και ολοκληρώνεται ένα αρχείο τύπου (.html) και με το DOCTYPE html δηλώνουμε ότι το αρχείο θα είναι αυτού του τύπου. Απαραίτητα δομικά στοιχεία επίσης είναι η ετικέτα <head> </head> όπου εκεί εισάγουμε τον τίτλο που θα εμφανίζεται, αλλά και λοιπά χαρακτηριστικά όπως κωδικοποίηση utf-8 , αναφορές σε άλλα αρχεία για λήψη χαρακτηριστικών από αυτά όπως bootstrap, css κλπ. Στο εσωτερικό των ετικετών <body> </body> υλοποιείται όλη η μορφοποίηση της πληροφορίας που θέλουμε να δει ο χρήστης στον περιηγητή που χρησιμοποιεί. Στο παραπάνω παράδειγμα το tag <title></title> δηλώνει ένα header, δηλαδή έναν τίτλο που θα είναι ορατός στον χρήστη και θα διακρίνεται με μεγαλύτερα και εντονότερα γράμματα.

Η ετικέτα <meta> ορίζει μεταδεδομένα για ένα έγγραφο HTML. Τα Metadata είναι δεδομένα σχετικά με δεδομένα. Οι ετικέτες <meta> βρίσκονται πάντα μέσα στο στοιχείο <head> και συνήθως χρησιμοποιούνται για τον καθορισμό του συνόλου χαρακτήρων, της περιγραφής της σελίδας, των λέξεων-κλειδιών, του συγγραφέα του εγγράφου και των ρυθμίσεων θύρας προβολής. Μια ακόμη σημαντική εντολή είναι η <table>, η οποία ορίζει έναν πίνακα στην HTML. Ο πίνακας αποτελείται από ένα ή περισσότερα <tr>, <td> ή <th>.

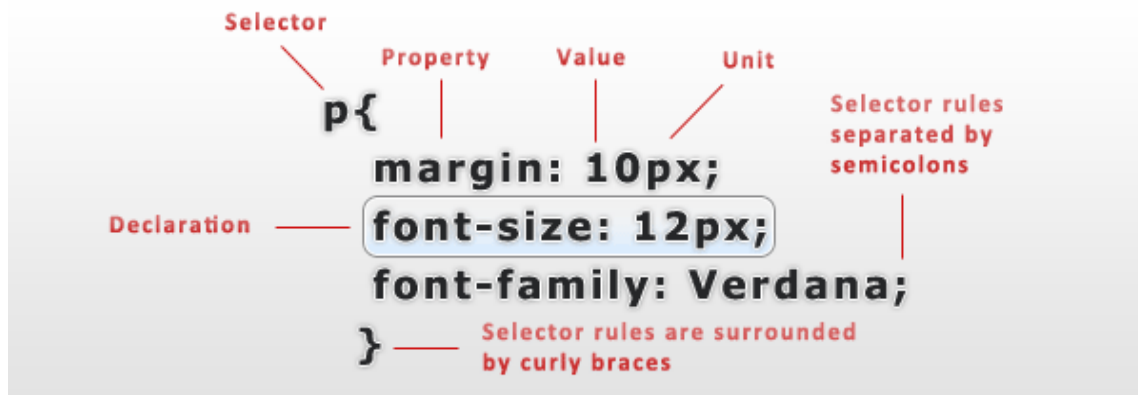
4.2.2 CSS

Η CSS (Cascading Style Sheets - Διαδοχικά Φύλλα Στυλ ή Αλληλουχία Φύλλων Στυλ), είναι μια γλώσσα υπολογιστή, που ανήκει στην κατηγορία των γλωσσών φύλλων στυλ που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης. Χρησιμοποιείται δηλαδή για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML, για τον έλεγχο της εμφάνισης μιας ιστοσελίδας και γενικότερα ενός ιστότοπου. Η CSS είναι μια γλώσσα υπολογιστή προορισμένη να αναπτύσσει το παρουσιαστικό μιας ιστοσελίδας δηλαδή να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση ενώ προσφέρει περισσότερες δυνατότητες σε σχέση με την html. Για μια όμορφη και καλοσχεδιασμένη ιστοσελίδα η χρήση της CSS κρίνεται ως απαραίτητη [33].

Ένας κανόνας CSS αποτελείται από δύο κύρια μέρη:

- Επιλογέας (selector): Στοιχείο προς μορφοποίηση. Μπορεί να είναι είτε μία ετικέτα html ή κάποιο συγκεκριμένα αντικείμενο html.
- Δηλώσεις (declaration): Αποτελούνται από τις ιδιότητες (property) που χαρακτηρίζουν τον επιλογέα και τις τιμές (value) που παίρνουν αυτές.

Παρακάτω απεικονίζεται ένα πρότυπο σύνταξης κανόνα CSS:



Εικόνα 4-6: Παράδειγμα σύνταξης CSS

4.2.3 PHP

Η Hypertext PreProcessor, με τα αρχικά PHP, είναι μία γλώσσα συγγραφής σεναρίων (scripting language) που ενσωματώνεται μέσα στον κώδικα της HTML και εκτελείται στην πλευρά του server (server-side scripting). Η PHP το μεγαλύτερο μέρος της σύνταξής της, το έχει δανειστεί από την C, την Java και την Perl, διαθέτοντας και μερικά δικά της μοναδικά χαρακτηριστικά. Είναι κατάλληλη για την ανάπτυξη διαδικτυακών εφαρμογών και προγραμμάτων συστήματος. Μια σελίδα PHP, περνά από επεξεργασία, από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε να παραχθεί σε πραγματικό χρόνο το τελικό περιεχόμενο, που θα σταλεί στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML [34].



Εικόνα 4-7: Το λογότυπο του PHP

Η PHP περιέχει κάποιες βασικές μεταβλητές οι οποίες βοηθούν στην ανάπτυξη διαδικτυακών δυναμικών εφαρμογών. Η PHP μεταβλητή, μπορεί να παρομοιαστεί με ένα δοχείο. Έτσι, μπορούμε να πούμε ότι η PHP μεταβλητή είναι ένα δοχείο που αποθηκεύονται πληροφορίες.

4.2.4 MySQL

Η MySQL, είναι μια πολύ γρήγορη σε απόδοση και ισχυρή σε δυνατότητες υλοποίηση, ενός συστήματος διαχείρισης σχεσιακών βάσεων δεδομένων. Τα δεδομένα στη MySQL αποθηκεύονται σε αντικείμενα βάσης δεδομένων, τα οποία ονομάζεται πίνακες. Ένας πίνακας, είναι μια συλλογή από σχετικές καταχωρήσεις δεδομένων και αποτελείται από στήλες και γραμμές. Οι βάσεις δεδομένων είναι χρήσιμες για την αποθήκευση πληροφοριών σε κατηγορίες.

Η διαχείριση των δεδομένων στις βάσεις, γίνονται μέσω των SQL εντολών, στη MySQL. Η πιο σημαντική ίσως κατηγορία εντολών που χρησιμοποιείται είναι τα queries. Ένα query είναι ένα ερώτημα ή ένα αίτημα. Με τη MySQL, επιτρέπεται η διερεύνηση σε μια βάση δεδομένων και η επιστροφή ζητηθέντων πληροφοριών [35].



Εικόνα 4-8: Το λογότυπο της MySQL

4.2.5 JavaScript

Η Javascript είναι η πιο δημοφιλής scripting γλώσσα στον κόσμο. Είναι η τυπική γλώσσα που χρησιμοποιείται σε ιστοσελίδες, αλλά χρησιμοποιείται ευρέως και από desktop εφαρμογές, εφαρμογές κινητής τηλεφωνίας, καθώς και διακομιστές διαδικτύου. Μια γλώσσα scripting είναι μια ελαφριά γλώσσα προγραμματισμού που υποστηρίζει τη συγγραφή σεναρίων. Σεναρία είναι γραμμές κώδικα που μπορούν να ερμηνεύονται και να εκτελούνται χωρίς μεταγλώττιση [36].

Βασικές λειτουργίες της java script είναι οι εξής:

- Χειρισμός CSS.
- Δυναμική αλλαγή ετικετών και περιεχομένου HTML.
- Αποθήκευση και ανάκτηση πληροφοριών στον υπολογιστή του χρήστη.
- Εκτέλεση μετά από συμβάν (όπως ενεργοποίηση επιλογής μενού).
- Επικοινωνία με php, xml, json αρχεία, αλλά και με άλλες ιστοσελίδες.

Η υλοποίηση της Javascript, είναι client – side (υπολογιστής πελάτη). Το χαρακτηριστικό αυτό, προσδίδει ασφάλεια σε δεδομένα που εισάγονται, καθώς πριν αυτά αποσταλούν στο διακομιστή (server) μπορούν να ελεγχθούν για την εγκυρότητα τους. Με αυτόν τον τρόπο επιτυγχάνεται η αποφυγή κακόβουλων επιθέσεων από τους χρήστες.

4.2.6 APACHE HTTP WEB SERVER

Ο Apache HTTP Web Server πρόκειται για έναν εξυπηρετητή (server) του παγκόσμιου Ιστού (Web). Με τον όρο εξυπηρετητή πολλοί ίσως να θεωρούν τους ηλεκτρονικούς υπολογιστές που φιλοξενούν ιστοσελίδες και όχι άδικα. Ο όρος αυτός μπορεί να χρησιμοποιηθεί και για το μηχάνημα εξυπηρετητή (hardware) αλλά και για το πρόγραμμα (software) [37].

Ο Apache μπορεί να εγκατασταθεί σε έναν υπολογιστή ο οποίος έχει τη δυνατότητα να χρησιμοποιεί διάφορα λειτουργικά συστήματα όπως Linux, Unix, Microsoft Windows, GNU, FreeBSD, Solaris, Novell NetWare, Mac OS X, OS/2, TPF. Ο ρόλος του Apache είναι να αναμένει αιτήσεις από διάφορα προγράμματα – χρήστες (clients) όπως είναι ο φυλλομετρητής (browser) ενός χρήστη και κατόπιν να εξυπηρετεί αυτές τις αιτήσεις «σερβίροντας» τις σελίδες που ζητούν είτε απευθείας μέσω μιας ηλεκτρονικής διεύθυνσης (URL), είτε μέσω ενός συνδέσμου (link). Ο τρόπος με τον οποίο ο Apache εξυπηρετεί αυτές τις αιτήσεις, είναι σύμφωνα με τα πρότυπα που ορίζει το πρωτόκολλο HTTP (Hypertext Transfer Protocol).

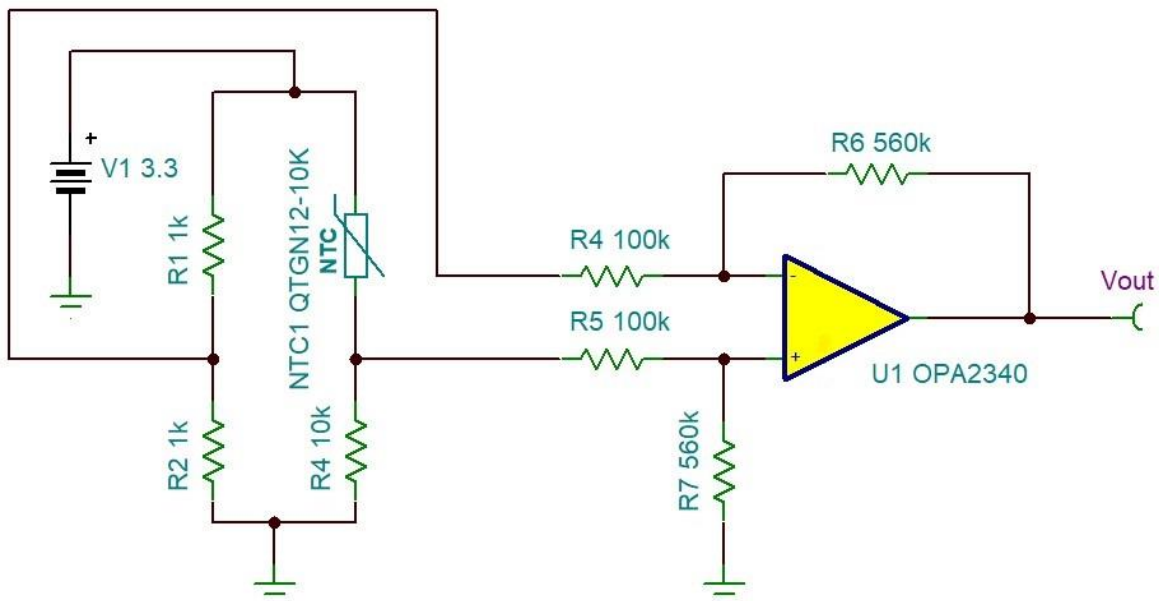
Ένα από τα πιο βασικά χαρακτηριστικά του, το οποίο και του δίνει μεγάλες δυνατότητες, είναι ότι μπορεί να προσαρμόσει επάνω του πολλές προσθήκες προγραμμάτων (modules), τα οποία με τη σειρά τους παρέχουν διαφορετικές λειτουργίες. Ένα επίσης, χαρακτηριστικό ή αλλιώς δυνατότητα του Apache HTTP είναι ότι μπορεί να εγκατασταθεί σε διάφορα λειτουργικά συστήματα. Επιπλέον, ο Apache HTTP υποστηρίζει αρκετές διάσημες εφαρμογές και γλώσσες προγραμματισμού όπως MySQL, PHP, Perl, Python κ.λπ.

ΚΕΦΑΛΑΙΟ 5. ΚΑΤΑΣΚΕΥΑΣΤΙΚΟ ΜΕΡΟΣ ΣΥΣΚΕΥΗΣ

Σε αυτό το κεφάλαιο θα περιγραφεί η υλοποίηση της συσκευής. Για την καλύτερη κατανόηση της, η περιγραφή της θα γίνει τμηματικά, ενώ στο τέλος θα γίνει η παρουσίαση της τελικής συσκευής.

5.1 Κατασκευή Θερμομέτρου

Η κατασκευή του θερμομέτρου βασίζεται στον συνδυασμό της γέφυρας Wheatstone και του ενισχυτή OPA2340, τον οποίον περιγράψαμε στο κεφάλαιο 3. Το κύριο πλεονέκτημα της γέφυρας Wheatstone είναι η ικανότητά της να παρέχει εξαιρετικά ακριβείς μετρήσεις (σε αντίθεση με κάτι σαν τον απλό διαιρέτη τάσης) [38]. Η λειτουργία της είναι παρόμοια με το αρχικό (original) ποτενσιόμετρο. Ο ενισχυτής από την άλλη πλευρά παράγει αναλογικά μεγαλύτερο σήμα πλάτους στην έξοδό του, παρέχοντας καλύτερη διακριτική ικανότητα. Η ποσότητα της ενίσχυσης που παρέχεται από τον ενισχυτή μετριέται από το κέρδος του, που είναι συνάρτηση του λόγου των αντιστάσεων που τον απαρτίζουν ($R7/R5$ ή $R6/R4$). Στην ουσία μας παρέχει την δυνατότητα να αξιοποιήσουμε όλο το εύρος τάσης του μικροελεγκτή, δηλαδή 0 - 3.3V. Αυτό σημαίνει επιπλέον πως η χαμηλότερη θερμοκρασία που θα λάβουμε θα είναι περίπου στα 0V και η υψηλότερη περίπου στα 3.3V στην έξοδο του κυκλώματος.



Σχήμα 5-1: Το σχηματικό διάγραμμα του θερμοίσοτ με τον ενισχυτή

Στο παραπάνω σχήμα, το NTC 10kΩ είναι ο θερμοίσοτ του οποίου η αντίσταση πρέπει να μετρηθεί. Τα R1 και R2 είναι αντιστάσεις γνωστής τιμής αντίστασης, οι οποίες για λόγους ευκολίας έχουν αντικατασταθεί με τον διαιρέτη τάσης TLE2426. Στα σημεία μεταξύ των δύο ακροδεκτών που καταλήγουν στον ενισχυτή, η διαφορά δυναμικού μεταξύ των δύο μεσαίων σημείων θα είναι περίπου μηδέν. Για αυτό τον λόγο, ο λόγος των δύο αντιστάσεων στο γνωστό σκέλος ($R1/R2$) είναι ίσος με τον λόγο των δύο αντιστάσεων στο άγνωστο σκέλος ($R_T/R4$). Επομένως η τιμή R4 καθορίζεται από την τιμή R_T του θερμοίσοτ. Η R4

θέλουμε να έχει τιμή αντίστασης, η οποία θα αντιστοιχεί στην χαμηλότερη θερμοκρασία που θέλουμε το θερμόμετρο να μας εμφανίζει. Για να βρούμε την τιμή της R4 όμως, θα πρέπει πρώτα να υπολογίσουμε την τιμή της αντίστασης RT στην χαμηλότερη και υψηλότερη θερμοκρασία που θέλουμε να λαμβάνουμε. Για αυτόν τον σκοπό θα χρειαστούμε να κάνουμε χρήση της εξίσωσης παραμέτρων B, την οποία και είδαμε στο υποκεφάλαιο 3.4. Το εύρος θερμοκρασιών που επιλέχθηκε για την εργασία είναι 25-42 βαθμούς Κελσίου επειδή τα ηλεκτρονικά θερμόμετρα μετρούν συνήθως θερμοκρασίες από 26,7°-42,2°C σε συνεχή λειτουργία [39].

Κάνοντας επομένως χρήση της εξίσωσης, υπολογίσαμε ότι στους 25°C ο θερμίστορ είχε αντίσταση 10kΩ και στους 42°C 4,9kΩ. Άρα η αντίσταση R4 θα ισούται με 10kΩ επειδή όπως προαναφέραμε θέλουμε να αντιστοιχεί στην χαμηλότερη θερμοκρασία. Έπειτα, κάνοντας χρήση της εξίσωσης κυκλώματος διαιρέτη, υπολογίστηκαν οι τάσεις της γέφυρας V1 και V2, των οποίων η διαφορά V2-V1 ισούται με το Vout, δηλαδή την τάση εξόδου. Η τάση V1 θα είναι σταθερή λόγω του διαιρέτη τάσης που έχουμε χρησιμοποιήσει, επομένως η V1 θα είναι το μισό των 3.3V, δηλαδή 1.65V. Στην πράξη όμως η τάση μετρήθηκε στα 1.62V και θα προτιμήσουμε αυτή την μέτρηση για μεγαλύτερη ακρίβεια. Όσον αφορά την τάση V2, ο υπολογισμός της έγινε με την εξίσωση διαιρέτη τάσης ως εξής:

$$V2 = [R4 / (R4+RT)] * 3.3 \text{ V.}$$

Η τιμή της RT αυτή την φορά θα ισούται με την αντίσταση που υπολογίσαμε στους 42 βαθμούς Κελσίου επειδή θέλουμε να υπολογίσουμε ποια θα είναι η μέγιστη τάση εξόδου (Vout) της γέφυρας Wheatstone. Επομένως έχουμε:

$$V2 = [10 / (10 + 4,9)] * 3,3 = 0,67 * 3,3 = 2,21 \text{ V.}$$

Αφαιρώντας από την V2 την V1 έχουμε:

$$V_{out} = V2 - V1 = 2,21 - 1,62 = 0,59 \text{ V.}$$

Στην συνέχεια υπολογίστηκε το κέρδος λόγου που χρειαζόμαστε από τον ενισχυτή ώστε να μπορέσουμε να αξιοποιήσουμε όλο το εύρος των 3,3V.

$$\text{Κέρδος} = A_d = 3,3 / V_{out} = 3,3 / 0,59 = 5,6.$$

Τέλος, από τον λόγο του κέρδους βρήκαμε τις κατάλληλες αντιστάσεις που ο ίδιος ο ενισχυτής χρειάζεται.

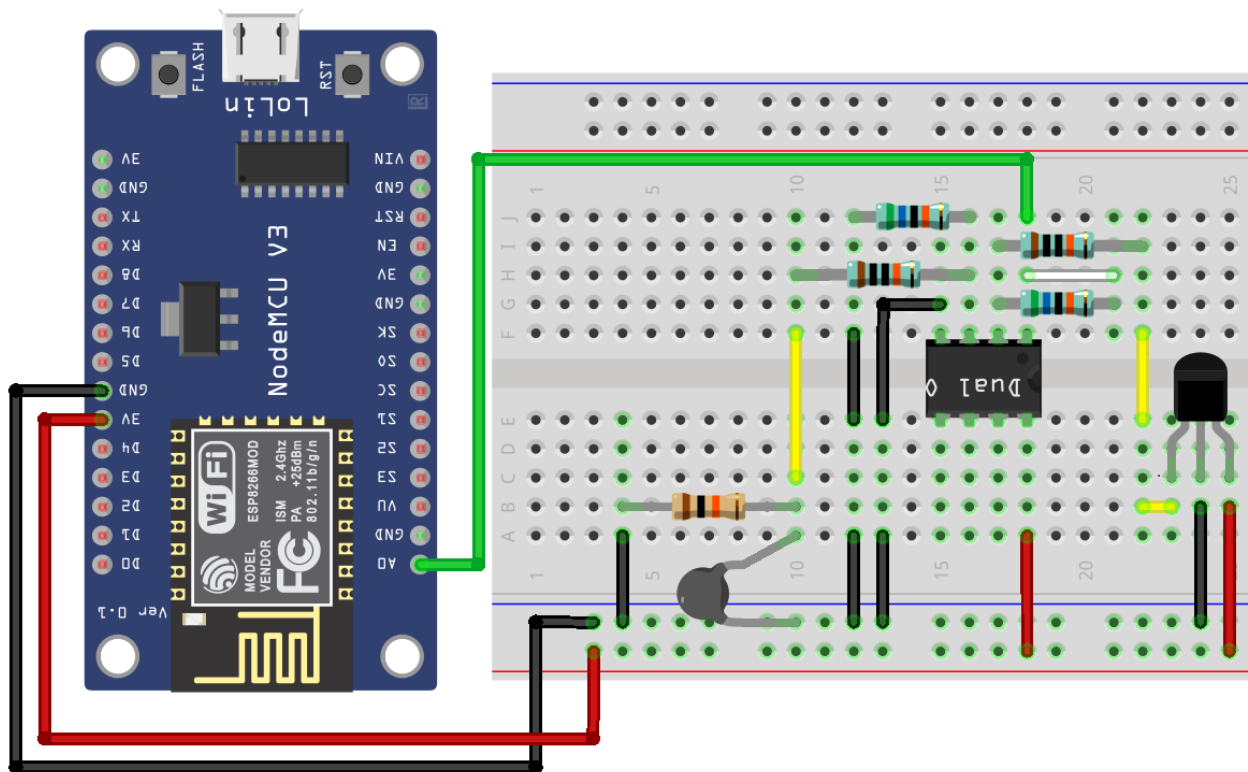
$$A_d = R7/R5 = R6/R4.$$

Επιλέχθηκαν οι αντιστάσεις με τιμές:

$$R4 = R5 = 100 \text{ k}\Omega \pm 1\% \ \& \ R6 = R7 = 560 \text{ k}\Omega \pm 1\%.$$

Αφότου ολοκληρώθηκαν οι αναγκαίοι υπολογισμοί και μετρήσεις για την επιλογή των εξαρτημάτων του κυκλώματος, ακολούθησε η κατασκευή του πάνω στο breadboard. Για την παρουσίαση του, έγινε η γραφική αναπαράσταση του κυκλώματος στο πρόγραμμα Fritzing. Το Fritzing είναι μια πρωτοβουλία ανοιχτού κώδικα για την ανάπτυξη λογισμικού CAD για τη σχεδίαση ηλεκτρονικών υλισμικών, με σκοπό να επιτρέψει σε σχεδιαστές και καλλιτέχνες να κατασκευάσουν πιο μόνιμα κυκλώματα από πρωτότυπα [40].

Παρακάτω παρουσιάζεται το διαγραμματικό κύκλωμα του θερμομέτρου με τον μικροελεγκτή ESP8266.



Διάγραμμα 5-1: Η συνδεσμολογία των εξαρτημάτων του θερμομέτρου και του ESP8266

Κατά την γραφή του κώδικα στο Arduino IDE επιλέξαμε αρχικά να δώσουμε ονομασίες για τις τιμές που ξέραμε ότι είναι σταθερές όπως η τάση V_a του διαιρέτη, τον λόγο κέρδους του ενισχυτή A_d , την αντίσταση R_4 , την σταθερά B του θερμοίστορα και την T_0 σε βαθμούς Kelvin.

```
//Parameters for Temperature (Temp)
const float Va = 1.62;
const float Ad = 5.6;
const float R4 = 9.8;
const float Beta = 3977;
const float T0 = 298.15;
```

Έπειτα ξεκινήσαμε μία σειριακή επικοινωνία με τον μικροελεγκτή όπως φαίνεται παρακάτω. Ο αριθμός 9600 σημαίνει ότι η σειριακή θύρα μπορεί να μεταφέρει έως και 9600 bits ανά δευτερόλεπτο. Μέσα στις σκουρές αγκύλες της εντολής «Void setup» γράφεται ο κώδικας που θέλουμε να εκτελεστεί μία φορά μόλις ξεκινήσει η εκτέλεση του προγράμματος.

```
void setup() {
  Serial.begin(9600); //Establishing serial communication with ESP8266
}
```

Στο «Void loop» γράφεται ο κώδικας που θέλουμε να επαναλαμβάνεται ανά τακτά χρονικά διαστήματα. Στην συγκεκριμένη περίπτωση ξεκινήσαμε με τον ορισμό μίας αναλογικής εισόδου από όπου θα μπορούμε να λάβουμε την τιμή μέτρησης του θερμίστορ.

```
int n = analogRead(A0); //Read analog input for Temperature
```

Εν συνεχεία ψηφιοποιήσαμε αυτή την τιμή ώστε να δούμε σε τι τιμή του Volt αντιστοιχεί.

```
float Vout = (n * 3.3)/1023; //Conversion of analog input to Voltage
```

Οι υπολογισμοί που ακολουθούν είναι στην ουσία η αντίστροφη διαδικασία από αυτήν που ακολουθήσαμε στην περιγραφή της κατασκευής του θερμομέτρου. Υπολογίσαμε αρχικά την τάση V21 η οποία είναι η τάση διαφοράς στην γέφυρα Wheatstone ώστε να υπολογίσουμε την τάση Vb ή V1, η οποία είναι η τάση μεταξύ του θερμίστορ και της αντίστασης R4. Με την εξίσωση του διαιρέτη τάσης υπολογίσαμε εν συνεχεία την τιμή αντίστασης του θερμίστορ RT και εντέλει με την εκθετική εξίσωση παραμέτρων B βρίσκουμε την θερμοκρασία σε βαθμούς Κέλβιν. Τέλος μετατρέπουμε την θερμοκρασία σε βαθμούς Κελσίου.

```
float V21 = Vout / Ad;  
float Vb = V21 + Va;  
float RT = R4 * ((3.3 / Vb) - 1);  
float T = (1.0/T0) + (1.0/Beta)*log(RT/10);  
T = 1/T;  
T = T - 273.15;
```

Για τον έλεγχο του προγράμματος, πραγματοποιείται παρουσίαση των αποτελεσμάτων για την θερμοκρασία και το αντίστοιχο Vout με την εντολή Serial.print. Επίσης τίθεται ο χρόνος παρουσίασης σε msec με την εντολή delay.

```
Serial.print("TEMP: "); //Print temperature  
Serial.print(T);  
  
Serial.print("      Voltage: "); //Print output voltage  
Serial.print(Vout);  
  
Serial.print('\n');  
delay(1000); //Print results every 1 second
```

Παρακάτω παρατίθενται 2 εικόνες με 2 διαφορετικά αποτελέσματα. Το πρώτο αποτέλεσμα στα αριστερά είναι η μέτρηση θερμοκρασίας του περιβάλλοντος, ενώ η δεξιά φωτογραφία δείχνει την μέτρηση θερμοκρασίας του δαχτύλου. Η θερμοκρασία είναι σε φυσιολογικές τιμές αφού όπως έχουμε προαναφέρει στο δάχτυλο η θερμοκρασία του ανθρώπου είναι χαμηλότερη, περίπου 30-33 °C.

COM6		COM6	
TEMP: 27.22	Voltage: 2.15	TEMP: 32.10	Voltage: 2.62
TEMP: 27.19	Voltage: 2.15	TEMP: 32.21	Voltage: 2.63
TEMP: 27.19	Voltage: 2.15	TEMP: 32.28	Voltage: 2.64
TEMP: 27.19	Voltage: 2.15	TEMP: 32.28	Voltage: 2.64
TEMP: 27.19	Voltage: 2.15	TEMP: 32.28	Voltage: 2.64
TEMP: 27.19	Voltage: 2.15	TEMP: 32.35	Voltage: 2.65
TEMP: 27.31	Voltage: 2.16	TEMP: 32.31	Voltage: 2.64
TEMP: 27.28	Voltage: 2.15	TEMP: 32.35	Voltage: 2.65
TEMP: 27.28	Voltage: 2.15	TEMP: 32.38	Voltage: 2.65
TEMP: 27.28	Voltage: 2.15	TEMP: 32.38	Voltage: 2.65
TEMP: 27.28	Voltage: 2.15	TEMP: 32.38	Voltage: 2.65

Εικόνα 5-1: Αποτελέσματα μετρήσεων θερμομέτρου

Παρακάτω παρατίθεται ο ολοκληρωμένος κώδικας του θερμομέτρου που γράφτηκε στο λογισμικό του Arduino IDE.

Κώδικας λειτουργίας του θερμομέτρου

```
//Parameters for Temperature (Temp)
const float Va = 1.62;
const float Ad = 5.6;
const float R4 = 9.8;
const float Beta = 3977;
const float T0 = 298.15;
//-----

void setup() {
  Serial.begin(9600); //Establishing serial communication with ESP8266
}
//-----

void loop() {

  int n = analogRead(A0); //Read analog input for Temperature

  float Vout = (n * 3.3)/1023; //Conversion of analog input to Voltage
  float V21 = Vout / Ad; //Calculate voltage difference
  float Vb = V21 + Va; //Calculate voltage between the thermistor
                        and resistance R4
  float RT = R4 * ((3.3 / Vb) - 1); //Calculate thermistor's resistor value
  float T = (1.0/T0) + (1.0/Beta)*log(RT/10); //Read temperature in Kelvin
  T = 1/T;
  T = T - 273.15; //Convert Kelvin to Celsius

  Serial.print("TEMP: "); //Print temperature
  Serial.print(T);

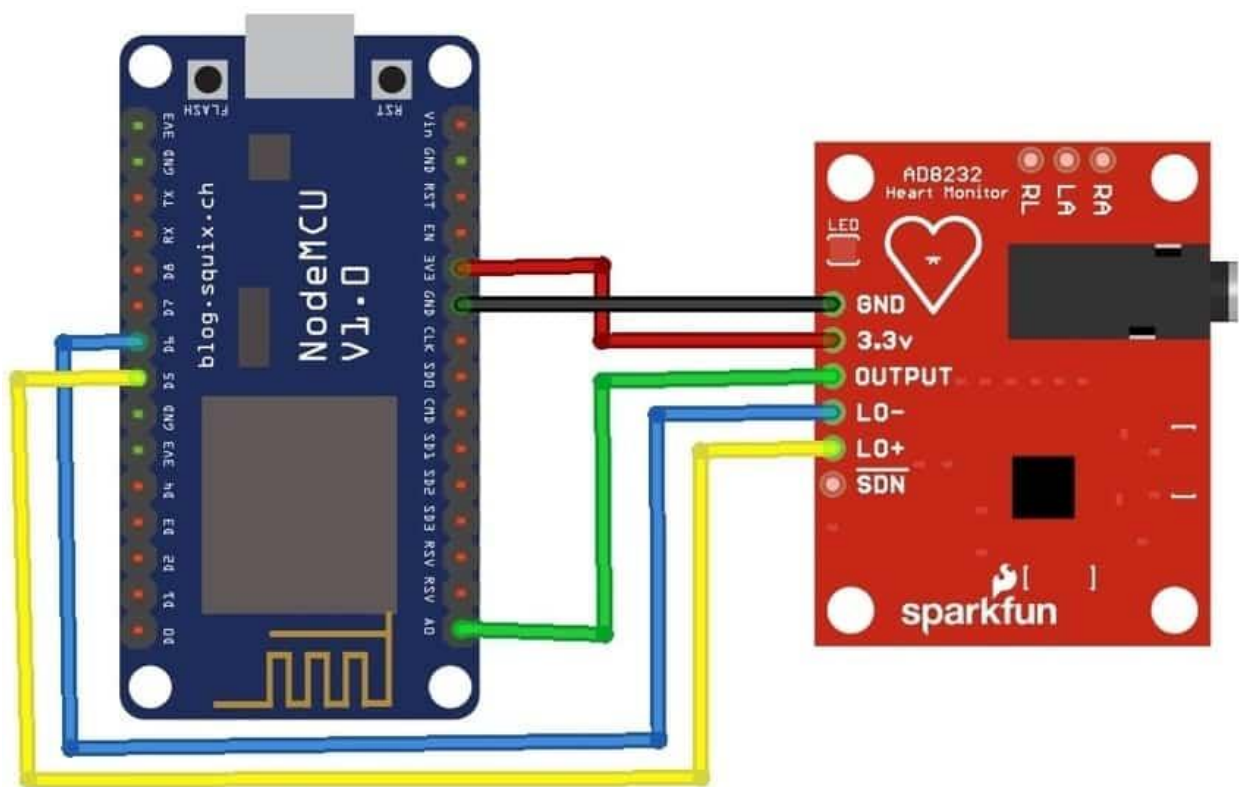
  Serial.print(" Voltage: "); //Print output voltage
  Serial.print(Vout);

  Serial.print('\n');
  delay(1000); //Print results every 1 second
}
```

5.2 Κατασκευή Καρδιογράφου

Η κατασκευή του καρδιογράφου βασίζεται στην πλακέτα AD8232. Το AD8232 μπορεί να χρησιμοποιηθεί ως εναλλακτική λύση στο τυπικό μόνιτορ ΗΚΓ για βασική παρακολούθηση και ανάλυση ΗΚΓ. Το κύριο πλεονέκτημα της παρούσας συσκευής είναι ότι μπορούμε να χρησιμοποιήσουμε μια απλή μονάδα ΗΚΓ με έναν υπολογιστή. Οι περισσότερες από τις μονάδες ΗΚΓ είναι είτε δαπανηρές, απαιτούν αποκλειστικό hardware, προγραμματισμό, υλικά όπως raspberry pi με rython ή άλλα ενσωματωμένα συστήματα. Ο γιατρός συχνά δεν έχει μια λεπτομερή ιδέα σχετικά με τις γλώσσες προγραμματισμού υπολογιστών. Όντας ένα απλό αξεσουάρ ενός υπολογιστή με φορητό εξοπλισμό, ένας γιατρός μπορεί εύκολα να λάβει δεδομένα από το ΗΚΓ χωρίς καμία δαπανηρή ρύθμιση.

Παρακάτω παρουσιάζεται το διαγραμματικό κύκλωμα του ΗΚΓ με τον μικροελεγκτή ESP8266.



Διάγραμμα 5-2: Η συνδεσμολογία του AD8232 με το ESP8266

Αρχικά πραγματοποιήθηκε έλεγχος στην ποιότητα του σήματος που μας προσφέρει η πλακέτα. Ο έλεγχος αυτός είναι σημαντικός, διότι προκειμένου να επιτευχθεί στην συνέχεια της κατασκευής η εύρεση και παρουσίαση των χτύπων ανά λεπτό, πρέπει να είμαστε βέβαιοι για το πόσο «καθαρό» ποιοτικά είναι το σήμα, δηλαδή αν υπάρχουν εξωτερικά σήματα ή θόρυβοι που επηρεάζουν το σήμα μας και το εάν ακολουθεί σωστά το διάγραμμα PQRST. Πριν γίνει όμως ο έλεγχος, πρέπει να χρησιμοποιήσουμε τα ηλεκτρόδια που θα μας βοηθήσουν στην λήψη του σήματος μας.

Το καλώδιο που χρησιμοποιήθηκε για την σύνδεση της πλακέτας AD8232 με τα τρία ηλεκτρόδια είναι της εταιρείας Sparkfun και παρουσιάζεται παρακάτω. Το μαύρο καλώδιο τοποθετείται στο άνω δεξί άκρο, το μπλε στο άνω αριστερό άκρο και το κόκκινο στο κάτω δεξί άκρο.



Εικόνα 5-2: Το καλώδιο τριών ηλεκτροδίων για σύνδεση με την πλακέτα AD8232

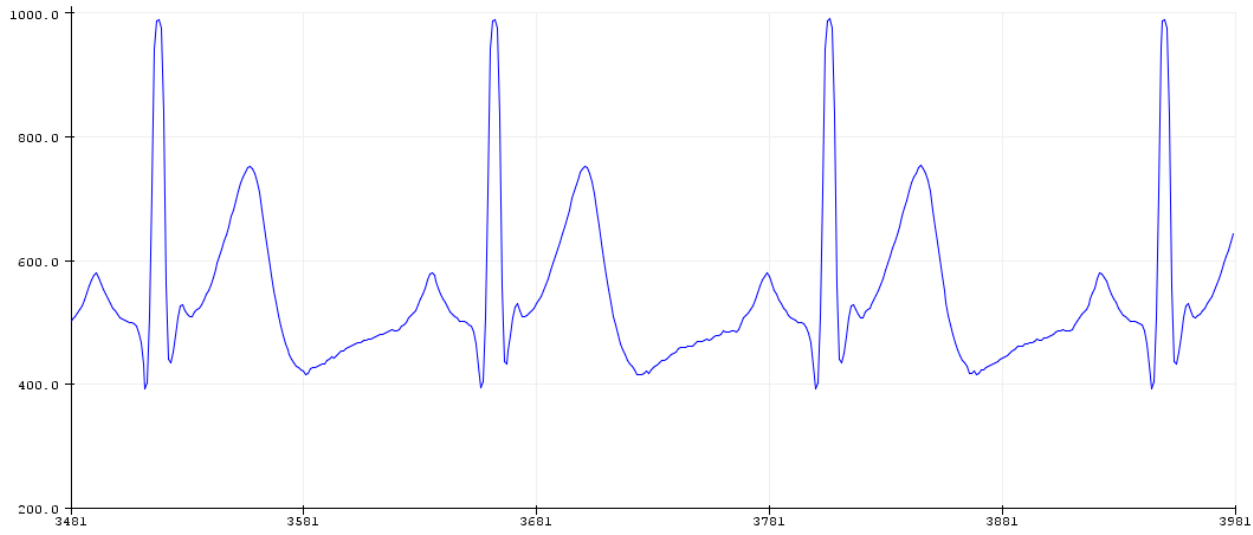
Έπειτα από την τοποθέτηση των ηλεκτροδίων σε έναν προσομοιωτή καρδιακού σήματος, γράψαμε και ανεβάσαμε τον απαραίτητο κώδικα για τον έλεγχο του σήματος, ο οποίος παρουσιάζεται παρακάτω.

Κώδικας ελέγχου σήματος ΗΚΓ

```
void setup()
{
  // initialize the serial communication:
  Serial.begin(9600);
  pinMode(4, INPUT); // Setup for leads off detection LO +
  pinMode(5, INPUT); // Setup for leads off detection LO -
}

void loop() {
  Serial.println(analogRead(A0)); //Read analog input for Heart signal
  delay(1);
}
```


Το σήμα το οποίο λάβαμε ήταν απολύτως ικανοποιητικό και πληρούσε όλα τα στάνταρ τα οποία προαναφέραμε. Επομένως η σχεδίαση και καταγραφή του BPM είναι δυνατή. Το παρακάτω διάγραμμα παρουσιάζει το σήμα που λάβαμε.



Διάγραμμα 5-3: Το καρδιακό σήμα του ΗΚΓ

Ο υπολογισμός του BPM, δηλαδή των χτύπων ανά λεπτό, είναι μία πιο περίπλοκη διαδικασία. Καταρχάς, πρέπει να υπολογιστεί η απόσταση μεταξύ δύο κορυφών R κατά την διάρκεια της καταγραφής των καρδιακών χτύπων (διάρκεια R-R). Πρώτα πρέπει να βρεθεί η χρονική στιγμή της πρώτης κορυφής, έπειτα η χρονική στιγμή της δεύτερης κορυφής που επακολουθεί και εντέλει να γίνει αφαίρεση των δύο αυτών χρονικών στιγμών. Για την ολοκλήρωση της διαδικασίας και τον υπολογισμό του BPM, πρέπει το 1 λεπτό να διααιρεθεί με την χρονική διαφορά που υπολογίσαμε προηγουμένως, δηλαδή:

$$\text{BPM} = 60 / \text{R-R διάρκεια (σε sec)} \quad \text{ή} \quad \text{BPM} = 60.000 / \text{R-R διάρκεια κύκλου (σε msec)}$$

Αυτή η ιδέα ακριβώς ακολουθήθηκε και κατά την γραφή του κώδικα.

Επιλέξαμε για άλλη μία φορά να ξεκινήσουμε την γραφή του κώδικα με το να δώσουμε ονομασίες για τις τιμές και τις μεταβλητές που χρειαζόμαστε. Πολύ σημαντική παράμετρος στον υπολογισμό του BPM είναι ο προκαθορισμός του ανώτατου κατωφλίου (UpperThreshold) επειδή εάν η τιμή του είναι χαμηλότερη από την αναγκαία, δεν θα μπορέσει να γίνει σωστά ο υπολογισμός. Η μεταβλητή reading χρησιμοποιείται για την αναγνώριση των δεδομένων στην αναλογική είσοδο, ενώ η μεταβλητή bool χρησιμοποιείται γιατί δίνει την δυνατότητα true/false. Στις υπόλοιπες μεταβλητές πραγματοποιήθηκε αρχικοποίηση, εκτός των δύο delayTime. Η delayTime είναι ο χρόνος ανά τον οποίον δίνονται εσωτερικά αποτελέσματα στο εάν έχει βρεθεί κάποια κορυφή στο καρδιακό σήμα, ενώ η delayTime2 είναι ο χρόνος ανά τον οποίον γίνεται η εκτύπωση των αποτελεσμάτων του BPM.

```

//Parameters for Heart Rate (BPM)
int UpperThreshold = 800;
int LowerThreshold = 500;
int reading = 0;
float BPM = 0.0;
bool IgnoreReading = false;
bool FirstPulseDetected = false;
unsigned long FirstPulseTime = 0;
unsigned long SecondPulseTime = 0;
unsigned long PulseInterval = 0;
const unsigned long delayTime = 10;
const unsigned long delayTime2 = 1000;
const unsigned long baudRate = 115200;
unsigned long previousMillis = 0;
unsigned long previousMillis2 = 0;

```

Αφότου επιτευχθεί ξανά μία σειριακή επικοινωνία με τον μικροελεγκτή όπως πραγματοποιήθηκε και στον κώδικα του θερμομέτρου, τότε ξεκινάει ο βασικός κώδικας υπολογισμού του BPM. Το πρώτο κομμάτι του κώδικα χρησιμοποιήθηκε για την εύρεση των κορυφών του πρώτου και δεύτερου καρδιακού χτύπου. Η ιδέα πίσω από την μέθοδο είναι πώς επιλέγονται δύο χρονικές στιγμές FirstPulseTime και SecondPulseTime στο καρδιακό σήμα, ελέγχονται οι αντίστοιχες τιμές του reading και όσο οι τιμές είναι υψηλότερες του UpperThreshold, το SecondPulseTime αντικαθιστά το FirstPulseTime και η διαδικασία αυτή επαναλαμβάνεται έως ότου παύσει να ισχύει ο βασικός περιορισμός. Τότε η διαφορά των δύο χρονικών στιγμών είναι η διάρκεια R-R ή αλλιώς PulseInterval.

```

// Get current time
unsigned long currentMillis = millis();

// First event - finding the peaks of the PQRST signal
if(myTimer1(delayTime, currentMillis) == 1){

    reading = analogRead(A0); //Read analog input for BPM

    // Heart beat leading edge detected.
    if(reading > UpperThreshold && IgnoreReading == false){
        if(FirstPulseDetected == false){
            FirstPulseTime = millis();
            FirstPulseDetected = true;
        }
        else{
            SecondPulseTime = millis();
            PulseInterval = SecondPulseTime - FirstPulseTime;
            FirstPulseTime = SecondPulseTime;
        }
        IgnoreReading = true;
    }
}

```

Ένα κομμάτι που δεν θα πρέπει να αγνοηθεί στην μελέτη του σήματος είναι το κατώτατο κατώφλι. Για αυτό τον λόγο για να διαβάξει το πρόγραμμα μας, αυτό το τμήμα του σήματος, συμπεριλαμβάνουμε επίσης μία εντολή.

```
// Heart beat trailing edge detected.
if(reading < LowerThreshold && IgnoreReading == true){
  IgnoreReading = false;
}
```

Για τον υπολογισμό του BPM, αφότου έχει βρεθεί η χρονική διάρκεια R-R (σε msec), το μόνο που απομένει να κάνουμε είναι να διαιρέσουμε το 1 λεπτό (ή 60.000 msec) με την διάρκεια R-R. Πριν από αυτό το βήμα όμως, προηγείται ένας έλεγχος ώστε το PulseInterval να είναι διάφορο του μηδενός. Αυτό συμβαίνει επειδή εάν βρεθεί να ισούται με το 0, τότε το BPM δεν θα εμφανίζει τιμή αλλά θα τείνει στο άπειρο.

```
// Calculate Beats Per Minute.
if (PulseInterval != 0) {
  BPM = 60000.0/PulseInterval;
}
}
```

Η εκτύπωση του BPM πραγματοποιείται ξανά με μία εντολή Serial.print και εντός του myTimer2.

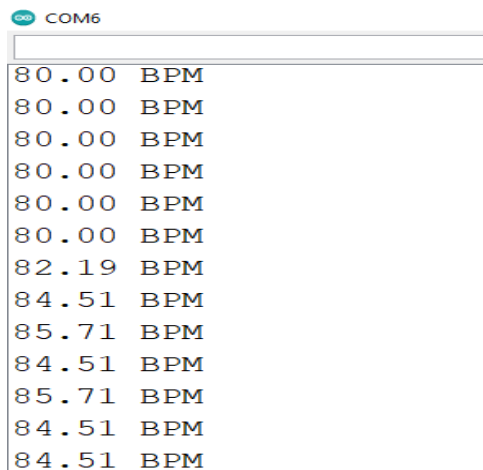
```
// Second event - printing results
if(myTimer2(delayTime2, currentMillis) == 1){
  Serial.print(BPM); //Print BPM (Beats per minute)
  Serial.println(" BPM");
}
}
```

Τέλος για να δώσουμε τις εντολές με τις οποίες θα γίνονται οι εκτυπώσεις στα χρονικά διαστήματα που έχουμε ορίσει, πρέπει να υπάρχουν και οι κατάλληλες μεταβλητές που ορίζουν αυτά τα χρονικά διαστήματα και τα εντοπίζουν. Αυτές οι μεταβλητές που έχουμε ορίσει είναι ο myTimer1 και ο myTimer2.

```
// First event timer
int myTimer1(long delayTime, long currentMillis){
  if(currentMillis - previousMillis >= delayTime){previousMillis = currentMillis;return 1;}
  else{return 0;}
}

// Second event timer
int myTimer2(long delayTime2, long currentMillis){
  if(currentMillis - previousMillis2 >= delayTime2){previousMillis2 = currentMillis;return 1;}
  else{return 0;}
}
```

Παρακάτω παρατίθεται ένα παράδειγμα μέτρησης στο Arduino IDE, στο οποίο χρησιμοποιήθηκε ένας προσομοιωτής καρδιάς και δείχνει την ακριβή μεταβολή του BPM από τους 80 στους 85 καρδιακούς παλμούς. Αξίζει να σημειωθεί πως η καταγραφή στην συγκεκριμένη περίπτωση συνέβαινε ανά 1 δευτερόλεπτο, το οποίο αποδεικνύει και την αποτελεσματικότητά του χρονικά ως προς τις μεταβολές του BPM.



```
COM6
80.00 BPM
80.00 BPM
80.00 BPM
80.00 BPM
80.00 BPM
80.00 BPM
82.19 BPM
84.51 BPM
85.71 BPM
84.51 BPM
85.71 BPM
84.51 BPM
84.51 BPM
```

Εικόνα 5-3: Αποτέλεσμα μέτρησης του ΗΚΓ

Παρακάτω παρατίθεται ο ολοκληρωμένος κώδικας του ΗΚΓ που γράφτηκε στο λογισμικό του Arduino IDE.

Κώδικας υπολογισμού BPM

```
//Parameters for Heart Rate (BPM)
int UpperThreshold = 800;
int LowerThreshold = 500;
int reading = 0;
float BPM = 0.0;
bool IgnoreReading = false;
bool FirstPulseDetected = false;
unsigned long FirstPulseTime = 0;
unsigned long SecondPulseTime = 0;
unsigned long PulseInterval = 0;
const unsigned long delayTime = 10;
const unsigned long delayTime2 = 1000;
unsigned long previousMillis = 0;
unsigned long previousMillis2 = 0;
//-----

void setup(){
  Serial.begin(115200); //Establishing serial communication with ESP8266
}
//-----

void loop(){

  // Get current time
```

```

unsigned long currentMillis = millis();

// First event - finding the peaks of the PQRST signal
if(myTimer1(delayTime, currentMillis) == 1){

    reading = analogRead(A0); //Read analog input for BPM

    // Heart beat leading edge detected.
    if(reading > UpperThreshold && IgnoreReading == false){
        if(FirstPulseDetected == false){
            FirstPulseTime = millis();
            FirstPulseDetected = true;
        }
        else{
            SecondPulseTime = millis();
            PulseInterval = SecondPulseTime - FirstPulseTime;
            FirstPulseTime = SecondPulseTime;
        }
        IgnoreReading = true;
    }

    // Heart beat trailing edge detected.
    if(reading < LowerThreshold && IgnoreReading == true){
        IgnoreReading = false;
    }

    // Calculate Beats Per Minute.
    if (PulseInterval != 0) { //Check for PulseInterval to be different than
0
        BPM = 60000.0/PulseInterval; //Calculate BPM (Beats per minute)
    }
}

// Second event - printing results
if(myTimer2(delayTime2, currentMillis) == 1){
    Serial.print(BPM); //Print BPM (Beats per minute)
    Serial.println(" BPM");
}
}

// First event timer
int myTimer1(long delayTime, long currentMillis){
    if(currentMillis - previousMillis >= delayTime){previousMillis =
currentMillis;return 1;}
    else{return 0;}
}

// Second event timer
int myTimer2(long delayTime2, long currentMillis){
    if(currentMillis - previousMillis2 >= delayTime2){previousMillis2 =
currentMillis;return 1;}
    else{return 0;}
}

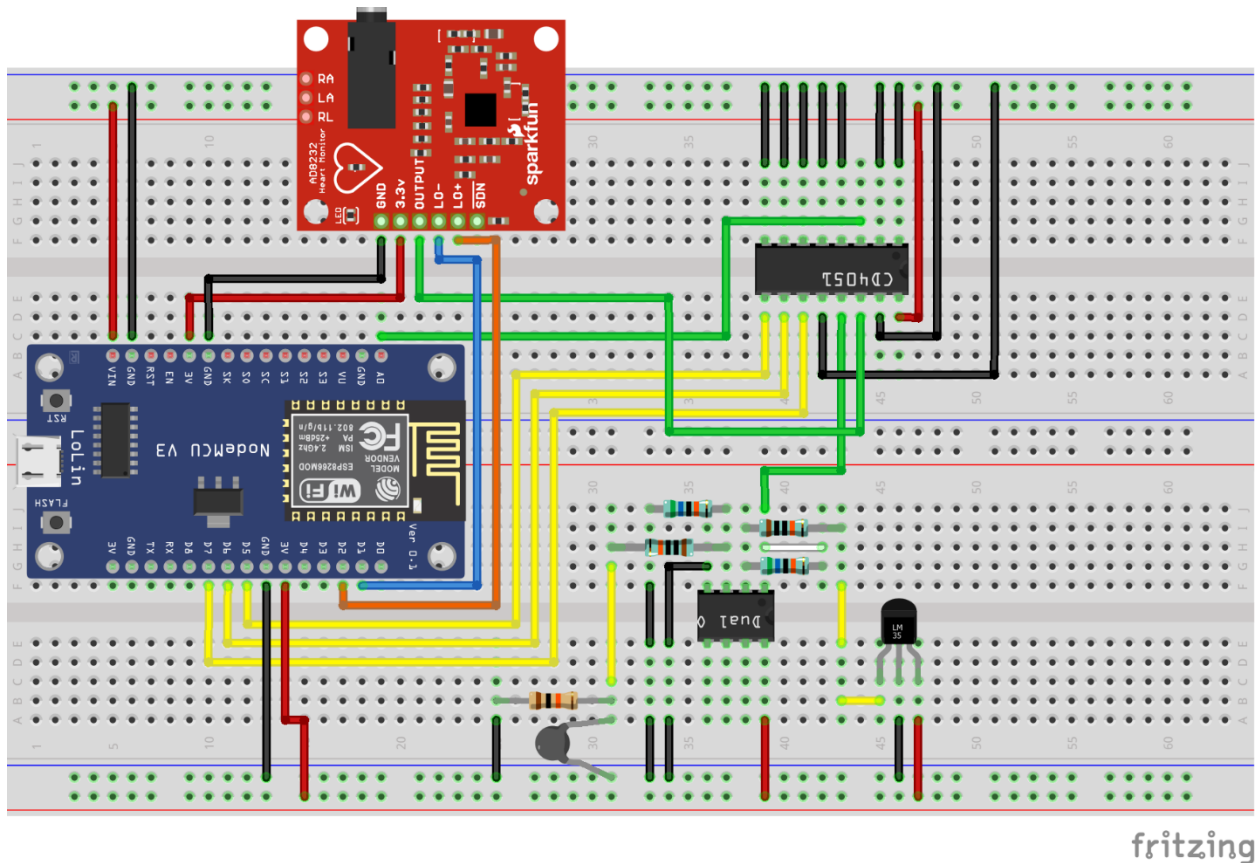
```

5.3 Ενσωμάτωση θερμομέτρου και ΗΚΓ με την χρήση του πολυπλέκτη

Η κατασκευή της φορητής συσκευής μας απαιτεί την ενσωμάτωση των 2 κατασκευών που ήδη πραγματοποιήσαμε, δηλαδή του θερμομέτρου και του καρδιογράφου, σε ένα ενιαίο σύστημα. Ο μικροελεγκτής ESP8266 όμως μας προσφέρει μόνο μία αναλογική είσοδο, το οποίο δεν επαρκεί στην περίπτωση μας, αφού χρειαζόμαστε 2 αναλογικές εισόδους, μία που θα δέχεται δεδομένα για το θερμομέτρο και μία για τον καρδιογράφο. Για αυτό τον λόγο ακριβώς αποφασίστηκε η χρήση του πολυπλέκτη CD4051 ο οποίος προσφέρει την δυνατότητα 8 αναλογικών εισόδων. Η λειτουργικότητα του CD4051 ως αναλογικού πολυπλέκτη εξηγείται σχετικά εύκολα.

Ο πολυπλέκτης λαμβάνει το σήμα γραμμής επιλογής από τις 3 ψηφιακές εξόδους του μικροελεγκτή. Η κάθε ψηφιακή είσοδος μπορεί να έχει κατάσταση HIGH ή LOW. Επειδή οι καταστάσεις που μπορεί να έχουν είναι 2 και οι ψηφιακές έξοδοι/είσοδοι είναι 3, οι πιθανές καταστάσεις ενεργοποίησης είναι 2^3 , δηλαδή 8 εισοδοί/έξοδοι.

Παρακάτω παρουσιάζεται το διαγραμματικό κύκλωμα της ενσωμάτωσης των δύο προηγούμενων κατασκευών, θερμομέτρου και καρδιογράφου, με τον πολυπλέκτη CD4051 και τον μικροελεγκτή.



Διάγραμμα 5-4: Η συνδεσμολογία της συσκευής

Ο κώδικας της ενσωμάτωσης του θερμομέτρου και του καρδιογράφου με τον πολυπλέκτη, είναι μια σχετικά εύκολη διαδικασία καθώς οι κώδικες των δύο πρώτων θα ενσωματωθούν γραμμένοι όπως είναι. Η μόνη διαφοροποίηση είναι ότι θα συμπεριληφθεί και ένα κομμάτι κώδικα για τον πολυπλέκτη, ώστε να επιτρέψει την αναγνώριση και ανάγνωση των 2 αναλογικών θυρών. Στην συγκεκριμένη περίπτωση επιλέχθηκαν ως αναλογικές εισοδοι από τον πολυπλέκτη η A0 για την καταγραφή της θερμοκρασίας και η A1 για την καταγραφή του καρδιακού παλμού, δηλαδή οι τρεις ψηφιακές εισοδοι ρυθμίστηκαν έτσι ώστε στην A0 όλες οι ψηφιακές εισοδοι να είναι σε κατάσταση LOW και στην A1 να είναι HIGH, LOW και LOW. Οι Ψηφιακές εισοδοι που επιλέχθηκαν είναι οι D5, D6 και D7.

```
const int selectPins[3] = {D7, D6, D5};
```

Αμέσως μετά γραφτήκαν οι παράμετροι θερμοκρασίας και καρδιογραφήματος, που ήδη έχουν αναλυθεί στα 2 παραπάνω υποκεφάλαια.

```
//Parameters for Temperature (Temp)
const float Va = 1.62;
const float Ad = 5.6;
const float R4 = 9.8;
const float Beta = 3977;
const float T0 = 298.15;
float Temp;
//-----
//Parameters for Heart Rate (BPM)
int UpperThreshold = 800;
int LowerThreshold = 500;
int reading = 0;
float BPM = 0.0;
float BeatRate;
bool IgnoreReading = false;
bool FirstPulseDetected = false;
unsigned long FirstPulseTime = 0;
unsigned long SecondPulseTime = 0;
unsigned long PulseInterval = 0;
const unsigned long delayTime = 10;
const unsigned long delayTime2 = 1000;
unsigned long previousMillis = 0;
unsigned long previousMillis2 = 0;
```

Έπειτα προχωρήσαμε στην γραφή του κώδικα στο Void setup. Πρώτα ξεκινήσαμε με την σειριακή διασύνδεση με τον μικροελεγκτή και στην συνέχεια θέσαμε τα 3 ψηφιακά pins σαν εξόδους. Η κατάσταση τέθηκε για ευκολία ως LOW ώστε να ξεκινάει πρώτα με την A0.

```

void setup() {
  Serial.begin(115200); //Establishing serial communication with ESP8266
  // Set up the select pins as outputs:
  for (int i = 0; i < 3; i++)
  {
    pinMode(selectPins[i], OUTPUT);
    digitalWrite(selectPins[i], LOW);
  }
}

```

Σε αυτό το σημείο συμπεριλήφθηκαν ξανά οι κώδικες υπολογισμού της θερμοκρασίας και του ΗΚΓ που ήδη έχουν αναλυθεί με την μόνη διαφορά πως στον κώδικα θέσαμε για τις καταγραφές και τις εκτυπώσεις 2 επιπλέον εντολές με τις οποίες αντιλαμβάνεται το πρόγραμμά μας πότε να καταγράψει θερμοκρασία και πότε ΗΚΓ.

```

void loop() {

  digitalWrite(selectPins[0], HIGH); //When it's HIGH record data for temperature

```

```

  digitalWrite(selectPins[0], LOW); //When it's LOW record data for BPM

```

Παρακάτω παρατίθεται ένα παράδειγμα μέτρησης στο Arduino IDE, στο οποίο χρησιμοποιήθηκε ένας προσομοιωτής καρδιάς και δείχνει το BPM στους 80 καρδιακούς παλμούς, ενώ ο όρος Temp παρουσιάζει την μεταβολή της θερμοκρασίας κατά το άγγιγμα του δαχτύλου στον θερμίστορ.

```

Temp: 29.18
      BPM: 80.00
Temp: 29.97
      BPM: 80.00
Temp: 31.68
      BPM: 80.00
Temp: 32.49
      BPM: 80.00
Temp: 32.74
      BPM: 80.00
Temp: 33.17
      BPM: 80.00
Temp: 33.42
      BPM: 80.00
Temp: 33.39

```

Εικόνα 5-4: Αποτέλεσμα μέτρησης της συσκευής

Πριν παραθέσουμε τον κώδικα αξίζει να σημειωθεί πως ο κώδικας της θερμοκρασίας, για την καλύτερη δυνατή μέτρηση και ακριβή εκτύπωση του, έχει προτιμηθεί να γραφτεί εντός του myTimer2.

Παρακάτω παρατίθεται ο κώδικας της συσκευής που γράφτηκε στο λογισμικό του Arduino IDE.

Κώδικας υπολογισμού Θερμοκρασίας & BPM

```
//Selection of digital pins which will record the two different analog
inputs

const int selectPins[3] = {D7, D6, D5};
//-----
//Parameters for Temperature (Temp)
const float Va = 1.62;
const float Ad = 5.6;
const float R4 = 9.8;
const float Beta = 3977;
const float T0 = 298.15;
float Temp;
//-----
//Parameters for Heart Rate (BPM)
int UpperThreshold = 800;
int LowerThreshold = 500;
int reading = 0;
float BPM = 0.0;
float BeatRate;
bool IgnoreReading = false;
bool FirstPulseDetected = false;
unsigned long FirstPulseTime = 0;
unsigned long SecondPulseTime = 0;
unsigned long PulseInterval = 0;
const unsigned long delayTime = 10;
const unsigned long delayTime2 = 1000;
unsigned long previousMillis = 0;
unsigned long previousMillis2 = 0;
//-----

void setup() {

  Serial.begin(115200); //Establishing serial communication with ESP8266

  // Set up the select pins as outputs:
  for (int i = 0; i < 3; i++)
  {
    pinMode(selectPins[i], OUTPUT);
    digitalWrite(selectPins[i], LOW);
  }
}
//-----

void loop() {

  digitalWrite(selectPins[0], HIGH); //When it's HIGH record data for
                                     temperature
```

```

//-----
int m = analogRead(A0); //Read analog input for BPM

// Get current time
unsigned long currentMillis = millis();

// First event - finding the peaks of the PQRST signal
if (myTimer1(delayTime, currentMillis) == 1)
{
    reading = m;

    // Heart beat leading edge detected.
    if (reading > UpperThreshold && IgnoreReading == false)
    {
        if (FirstPulseDetected == false) {
            FirstPulseTime = millis();
            FirstPulseDetected = true;
        }
        else {
            SecondPulseTime = millis();
            PulseInterval = SecondPulseTime - FirstPulseTime;
            FirstPulseTime = SecondPulseTime;
        }
        IgnoreReading = true;
    }

    // Heart beat trailing edge detected.
    if (reading < LowerThreshold && IgnoreReading == true) {
        IgnoreReading = false;
    }

    // Calculate Beats Per Minute.
    if (PulseInterval != 0) { //Check for PulseInterval to be
        //different than 0
        BPM = 60000.0 / PulseInterval; //Calculate BPM (Beats per minute)
    }
}

// Second event - printing results
if (myTimer2(delayTime2, currentMillis) == 1) {

    Serial.print("    BPM: ");
    Serial.println(BPM); //Print BPM (Beats per minute)
    digitalWrite(selectPins[0], LOW); //When it's LOW record data for BPM
    //delay(10);

    int n = analogRead(A0); //Read analog input for Temperature

    float Vout = (n * 3.3) / 1023; //Conversion of analog input to Voltage
    float V21 = Vout / Ad; //Calculate voltage difference
    float Vb = V21 + Va; //Calculate voltage between the thermistor and R4
    float RT = R4 * ((3.3 / Vb) - 1); //Calculate thermistor's RT value
    float T = (1.0 / T0) + (1.0 / Beta) * log(RT / 10); //Read temperature
    //in Kelvin
}

```

```

T = 1 / T;
T = T - 273.15; //Convert Kelvin to Celsius

Serial.print("Temp: "); Serial.println(T); //Print the temperature in
                                           Celsius

}
}

// First event timer
int myTimer1(long delayTime, long currentMillis) {
  if (currentMillis - previousMillis >= delayTime) {
    previousMillis = currentMillis;
    return 1;
  }
  else {
    return 0;
  }
}

// Second event timer
int myTimer2(long delayTime2, long currentMillis) {
  if (currentMillis - previousMillis2 >= delayTime2) {
    previousMillis2 = currentMillis;
    return 1;
  }
  else {
    return 0;
  }
}
}

```

5.4 Σύνδεση συσκευής στο διαδίκτυο μέσω Wi-Fi

Κάθε πληροφορία για τον ασθενή ακολουθείται και από μία χρονική στιγμή που αυτή λήφθηκε, έχοντας μία «ένα-προς-ένα» αντιστοιχία αυτών. Λαμβάνοντας τα δεδομένα από την Συσκευή, το μόνο που μένει είναι να τα στείλουμε σε έναν κεντρικό διακομιστή ο οποίος θα δύναται να τα ταξινομεί, να τα παρουσιάζει σε πραγματικό χρόνο και να δίνει την δυνατότητα στο ιατρικό προσωπικό να τα μελετήσει. Για αυτό τον λόγο πρέπει να κατασκευαστεί πρώτα μία βάση δεδομένων στην οποία θα καταχωρούνται οι μετρήσεις και έπειτα θα παρουσιάζονται σε μία ιστοσελίδα.

Για τη σύνδεση της Συσκευής μας (δηλαδή του μικροελεγκτή) με έναν server, κάνουμε χρήση των βιβλιοθηκών που μας προσφέρονται από την Espressif. Το πρώτο βήμα που πρέπει να εκτελέσουμε, είναι η σύνδεση της συσκευής στο τοπικό δίκτυο Wi-Fi. Αυτό γίνεται μέσω της συνάρτησης WiFi.begin(), η οποία δέχεται ως ορίσματα το SSID και τον κωδικό του δικτύου και αφού εκτελεστεί, έπειτα από λίγα δευτερόλεπτα, ο μικροελεγκτής συνδέεται στο τοπικό δίκτυο Wi-Fi.

```
//Establishing connection to Wi-Fi network
WiFi.begin(ssid, password);
```

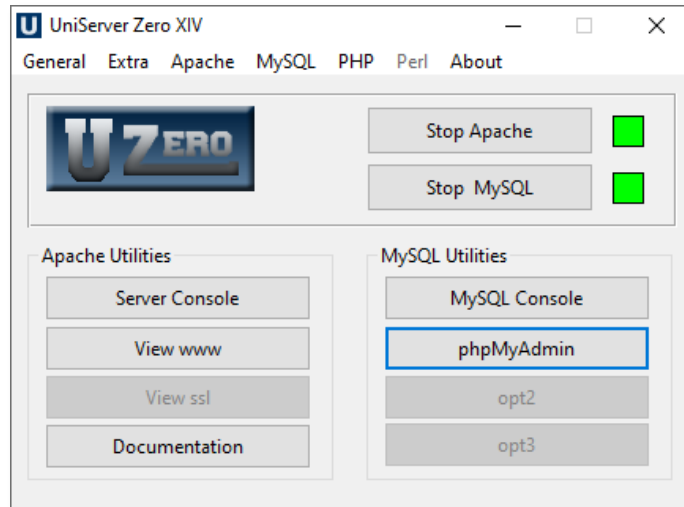
Τα απαραίτητα ορίσματα δίνονται ως global μεταβλητές.

```
// Replace with your network credentials
const char* ssid      = "OTe30d80";
const char* password = "15a14A57";
```

Έχοντας συνδεθεί στο δίκτυο, αποκτούμε πρόσβαση στο διαδίκτυο και μπορούμε να λάβουμε και να αποστείλουμε δεδομένα με οποιονδήποτε server.

Η σύνδεση Wi-Fi χρησιμοποιείται αρχικώς για να λάβουμε την τωρινή ημερομηνία. Ο μικροελεγκτής δεν διαθέτει μια τέτοια ικανότητα από μόνος του χωρίς σύνδεση στο ίντερνετ και είναι απολύτως απαραίτητο να γνωρίζουμε την ώρα που λαμβάνονται τα βιοσήματα. Για αυτό τον σκοπό χρησιμοποιείται η εντολή TIMESTAMP στο phpMyAdmin, όρισμα της οποίας είναι οι διαφορές ώρας ανάλογα με την τοποθεσία και ένας εξυπηρετητής NTP (Network Time Protocol), ο οποίος είναι συγχρονισμένος με το εθνικό πρότυπο χρόνου. «Ο τύπος δεδομένων TIMESTAMP χρησιμοποιείται για τιμές που περιέχουν μέρη ημερομηνίας και ώρας. Εφόσον η ημερομηνία έχει συγχρονιστεί μία φορά μέσω του εξυπηρετητή, ένας μετρητής στον μικροελεγκτή αναλαμβάνει την ευθύνη να συγχρονίζεται με την πάροδο του χρόνου, οπότε, ανά πάσα στιγμή, η ώρα είναι διαθέσιμη για ανάγνωση.

Όσον αφορά την μεριά του Server, αυτή υλοποιήθηκε μέσω του προγράμματος UniServer Zero XIV. Ο UniServer (OS 'X', Apache, MariaDB, MySQL, PHP, Perl) είναι ένα πακέτο ανοιχτού κώδικα που δίνει τη δυνατότητα σε προγραμματιστές να τρέχουν στους υπολογιστές τους Apache Web Servers σε συνδυασμό με MySQL, PHP και Perl. Εκτελώντας τον UniServer, ενεργοποιούμε τις υπηρεσίες Web Server και MySQL για να ακούμε εισερχόμενα αιτήματα και να έχουμε πρόσβαση σε μία βάση δεδομένων, όπως φαίνεται παρακάτω.



Εικόνα 5-5: Ενεργοποίηση του Apache server και της MySQL

Η βασική δομή λειτουργίας είναι η εξής: ο μικροελεγκτής στέλνει τα δεδομένα που έχει συλλέξει στον server, ο οποίος τα αποθηκεύει στη βάση δεδομένων. Έπειτα, όποτε θελήσουμε να προβάλουμε τα δεδομένα, πληκτρολογούμε στον server το εύρος που μας ενδιαφέρει και ο τελευταίος αναζητά τα δεδομένα στη βάση δεδομένων και τα παρουσιάζει σε μία ιστοσελίδα. Επομένως, η όλη διαδικασία μπορεί να χωριστεί σε δύο τμήματα: 1) Στη σχέση **ESP8266 -> Web Server -> Database** που ευθύνεται για την αποστολή των δεδομένων από τον μικροελεγκτή και την κατάληξή τους στη βάση δεδομένων και 2) στη σχέση **Web Server <-> Database** η οποία είναι υπεύθυνη για την επιλογή των δεδομένων που μας ενδιαφέρουν και η προβολή αυτών.

Το πρώτο βήμα για την κατάληξη των δεδομένων σε μία βάση δεδομένων ξεκινάει προφανώς από την αποστολή τους από τον μικροελεγκτή. Στα πλαίσια της διπλωματικής, η τελική μέθοδος που επιλέχτηκε για την αποστολή των βιοσημάτων που έχουν συλλεχτεί είναι η αποστολή τους μέσω HTTP Requests και HTTP POST. Πιο συγκεκριμένα, εισήχθη η βιβλιοθήκη <HTTP.Client.h>, η οποία καθιστά πιο εύκολη την χρήση του ESP8266 ως Client.

```
#include <HTTPClient.h>
```

Για την αποστολή των δεδομένων, είναι απαραίτητη η εισαγωγή του local IP του υπολογιστή στον οποίο λειτουργεί ο Server, ώστε να γνωρίζει ο μικροελεγκτής που να στείλει τα αιτήματα, καθώς και το port στο οποίο ο Server ακούει τα HTTP Requests. Επίσης αναγκαίο είναι να γνωρίζει ο μικροελεγκτής σε ποιο αρχείο του Server να στείλει τα αιτήματα τα οποία θα λαμβάνουν τα απεσταλμένα δεδομένα.

```
// REPLACE with your Domain name and URL path or IP address with path
const char* serverName = "http://192.168.1.3/tempandbpm/post-esp-data.php";
```

Κάθε φορά που το ESP8266 είναι συνδεδεμένο στο δίκτυο Wi-Fi και πρέπει να στείλει τα αρχεία για τη βάση δεδομένων, ξεκινάει με το να εκτελεί τη συνάρτηση `http.begin()`, η οποία προετοιμάζει την HTTP σύνδεση με τον διακομιστή, σε κάποιο port που αυτός «ακούει» και σε ένα συγκεκριμένο php αρχείο.

```
// Your Domain name with URL path or IP address with path
http.begin(client, serverName);
```

Ύστερα, η συνάρτηση αυτή εισάγει το header που απαιτείται πριν την αποστολή του αιτήματος το οποίο είναι “Content-Type”=“x-www-form-urlencoded”.

```
// Specify content-type header
http.addHeader("Content-Type", "application/x-www-form-urlencoded");
```

Αν η συνάρτηση http.begin() επιστρέφει true, αυτό σημαίνει πως η σύνδεση είναι ανοιχτή και ο μικροελεγκτής θα προχωρήσει στο επόμενο βήμα, αλλιώς η όλη διαδικασία σταματάει. Το επόμενο βήμα αποτελείται από την πραγματική αποστολή των δεδομένων.

Εκτελείται η συνάρτηση httpRequestData η οποία ξεκινάει τη σύνδεση του μικροελεγκτή με την εσωτερική του μνήμη και αναζητάει δεδομένα σε αυτήν.

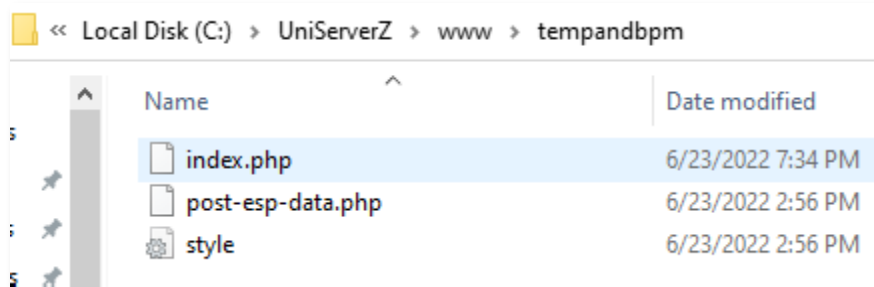
```
// Prepare your HTTP POST request data
String httpRequestData = "api_key=" + apiKeyValue + "&temp=" + String(Temp) + "&bpm=" + String(BPM) + "";
```

Με το που βρει ένα αρχείο, το ανοίγει για να προβάλλει τα δεδομένα που είναι αποθηκευμένα σε αυτό και προσπαθεί να τα στείλει μέσω http.POST(httpRequestData).

```
int httpResponseCode = http.POST(httpRequestData);
```

Με την ολοκλήρωση της σύνδεσης του μικροελεγκτή στο δίκτυο και την αποστολή των δεδομένων από αυτόν, πρέπει να εισάγουμε τα δεδομένα σε κάποια βάση δεδομένων. Γι’ αυτό τον λόγο θα χρησιμοποιήσουμε τον UniServer, ο οποίος μας δίνει την δυνατότητα να δημιουργήσουμε ιστότοπο WordPress. Όπως προαναφέραμε, αφότου ενεργοποιήσουμε τις υπηρεσίες Web Server και MySQL μπορούμε να ακούμε εισερχόμενα αιτήματα και να έχουμε πρόσβαση σε μία βάση δεδομένων.

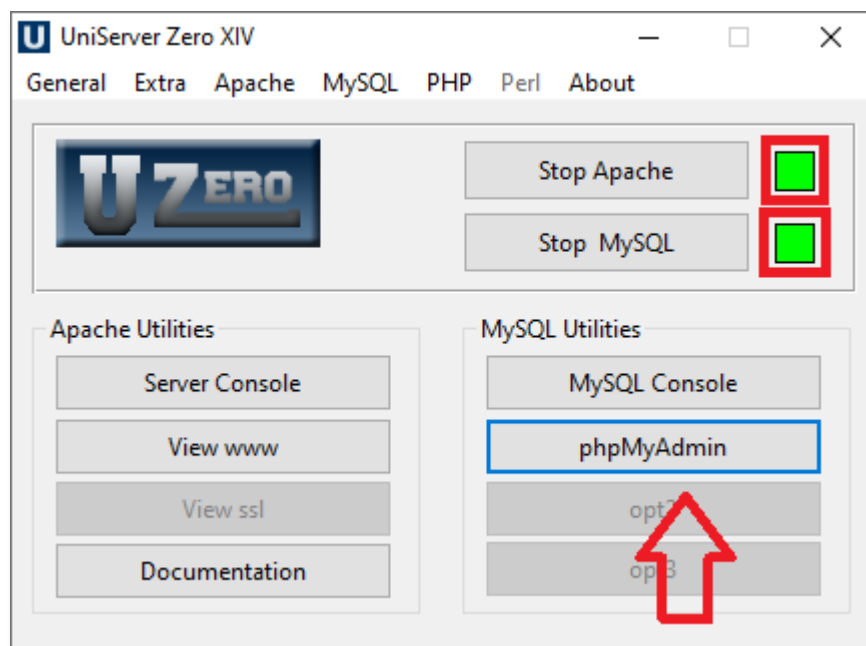
Εφόσον πρόκειται να χρησιμοποιήσουμε το UniServer, πρέπει να έχουμε έναν φάκελο για να κρατάμε όλα τα αρχεία έργου μας εκεί. Σε αυτήν την περίπτωση θα τοποθετήσουμε όλα τα αρχεία του έργου μας μέσα στο φάκελο **tempandbpm** στη διαδρομή **C:\UniServerZ\www\tempandbpm**, όπως φαίνεται παρακάτω.



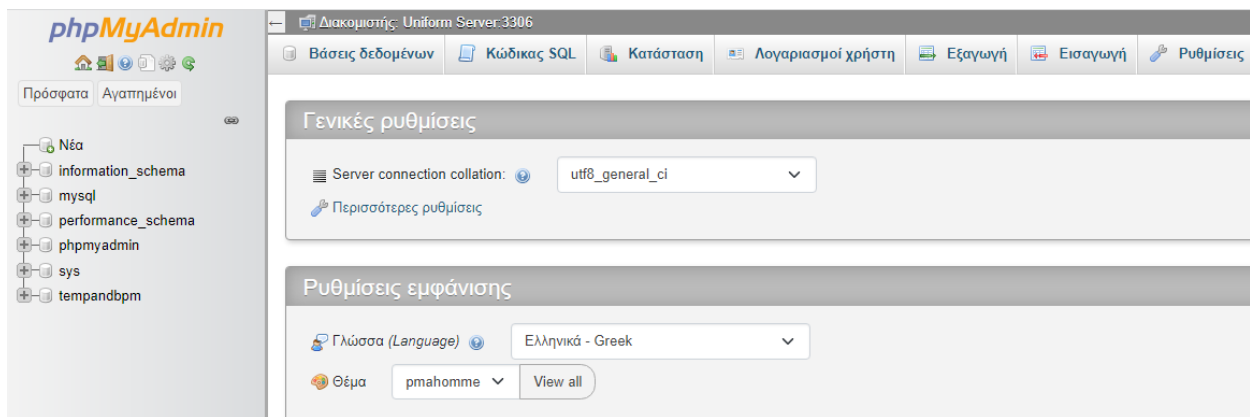
Ο φάκελος αυτός περιλαμβάνει τα εξής αρχεία:

- **index.php**: Σενάριο PHP που εκτελείται για να εμφανίσει το περιεχόμενο της βάσης δεδομένων σε μια ιστοσελίδα.
- **post-esp-data.php** : περιέχει το σενάριο PHP που είναι υπεύθυνο για τη λήψη εισερχόμενων αιτημάτων από το ESP8266 και την εισαγωγή των δεδομένων σε μια βάση δεδομένων MySQL.
- **style.css**: για την τροποποίηση της εμφάνισης της ιστοσελίδας μας.

Για την πρόσβαση στη βάση δεδομένων της MYSQL επιλέγουμε την επιλογή phpMyAdmin η οποία μας στέλνει στην διεύθυνση URL της phpMyAdmin.



Αφότου καταφέρουμε επιτυχώς να εισέλθουμε στην phpMyAdmin, ο περιβάλλον χώρος θα μοιάζει σαν και αυτόν της παρακάτω εικόνας.



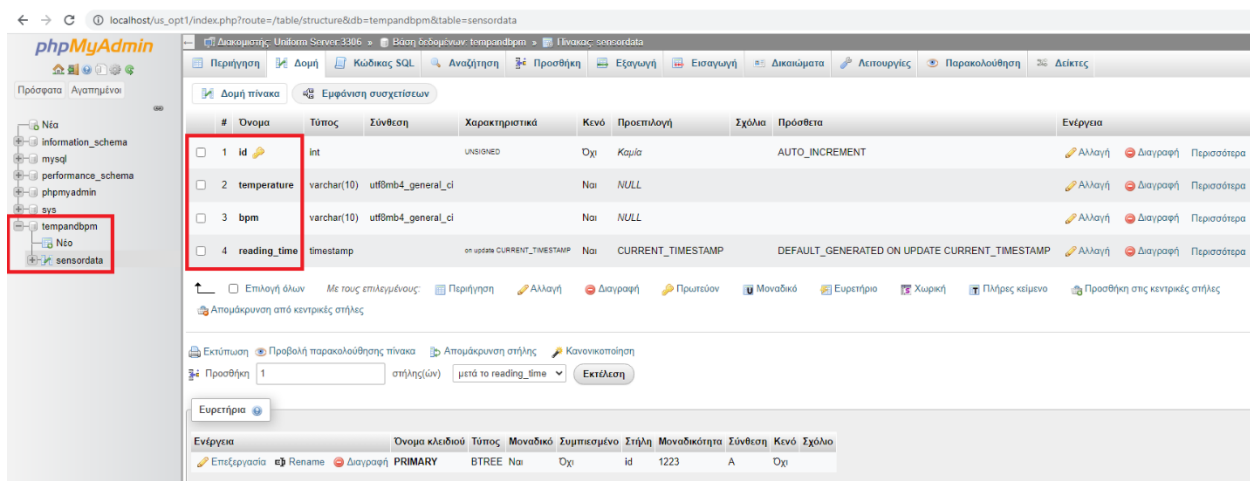
Στη συνέχεια, για να δημιουργήσουμε μια νέα βάση δεδομένων, κάνουμε κλικ στην επιλογή **Νέο > Εισαγωγή ονόματος βάσης δεδομένων > Δημιουργία** και στην συνέχεια εισάγουμε τον κώδικα που είναι γραμμένος σε γλώσσα MySQL στην επιλογή **Κώδικας SQL**. Ο κώδικας για την δημιουργία της βάσης δεδομένων φαίνεται παρακάτω.

```

1  /*creating a table in the database*/
2  CREATE TABLE tempandbpm (
3      id int UNSIGNED AUTO_INCREMENT PRIMARY KEY,
4      temp varchar(10) ,
5      bpm varchar(10) ,
6      reading_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
7  );
8

```

Αφού εκτελέσουμε την εισαγωγή του κώδικα θα εμφανιστεί ο περιβάλλον χώρος στον οποίο μπορούμε να δούμε την δομή του πίνακά μας. Στην αριστερή πλευρά μπορούμε να βρούμε την βάση δεδομένων μας με όνομα **tempandbpm** και τον πίνακα που δημιουργήσαμε με όνομα **sensordata**.



Όπως ήδη προαναφέραμε, το αρχείο **post-esp-data.php** είναι αυτό που μας επέτρεψε την εισαγωγή των δεδομένων στη βάση δεδομένων της MySQL. Ο αντίστοιχος κώδικας παρατίθεται παρακάτω.

```
<?php
$servername = "localhost";
$username = "root";
$password = "qwerty";
$dbname = "tempandbpm";

/*$conn = mysqli_connect($dbServername, $dbUsername, $dbPassword,
$dbName);*/

$api_key_value = "tPmAT5Ab3j7F9";

$api_key= $value1 = $value2 = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $api_key = test_input($_POST["api_key"]);
    if($api_key == $api_key_value) {
        $value1 = test_input($_POST["temp"]);
        $value2 = test_input($_POST["bpm"]);

        // Create connection
        $conn = new mysqli($servername, $username, $password, $dbname);
        // Check connection
        if ($conn->connect_error) {
            die("Connection failed: " . $conn->connect_error);
        }

        $sql = "INSERT INTO SensorData (temperature, bpm) VALUES ('" .
        $value1 . "', '" . $value2 . "')";

        if ($conn->query($sql) === TRUE) {
            echo "New record created successfully";
        }
        else {
            echo "Error: " . $sql . "<br>" . $conn->error;
        }

        $conn->close();
    }
    else {
        echo "Wrong API Key provided.";
    }
}
else {
    echo "No data posted with HTTP POST.";
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
```

Η βασική του εντολή είναι η `$conn -> query(sql) === TRUE`, η οποία και πραγματοποιεί την διασύνδεση του μικροελεγκτή με την βάση δεδομένων. Στην ουσία αποδέχεται τα εισερχόμενα αιτήματα του μικροελεγκτή, αφότου επιστρέψει ως αποτέλεσμα την τιμή TRUE, και τοποθετεί τα δεδομένα που ο μικροελεγκτής καταγράφει στην βάση. Επίσης σημαντικό είναι το API KEY. Τα κλειδιά API χρησιμοποιούνται για τον έλεγχο ταυτότητας ενός καλούμενου προγράμματος σε άλλο API, συνήθως για να επιβεβαιώσουν ότι ένα έργο είναι εξουσιοδοτημένο για σύνδεση. Πραγματοποιείται για λόγους ασφαλείας.

Με την ολοκλήρωση της κατασκευής της βάσης δεδομένων μας το μόνο που απομένει είναι η δημιουργία μίας ιστοσελίδας στην οποία θα απεικονίζουμε τα δεδομένα μας. Όπως ήδη προαναφέρθηκε, το αρχείο **index.php** εκτελείται για να εμφανίσει το περιεχόμενο της βάσης δεδομένων σε μια ιστοσελίδα. Ο αντίστοιχος κώδικας παρατίθεται παρακάτω.

```
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="refresh" content="5" >
  <link rel="stylesheet" type="text/css" href="style.css"
media="screen"/>

  <title> Sensor Data </title>
</head>
<body>

  <h1>SENSOR DATA</h1>
<?php
$servername = "localhost";
$username = "root";
$password = "qwerty";
$dbname = "tempandbpm";

// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "SELECT id, temperature, bpm, reading_time FROM sensordata ORDER BY
id DESC"; /*select items to display from the sensordata table in the data
base*/

echo '<table cellpadding="5" cellspacing="5">
  <tr>
    <th>ID</th>
    <th>Date & Time</th>
    <th>Temperature &deg;C</th>
    <th>BPM</th>
  </tr>';
```

```

if ($result = $conn->query($sql)) {
    while ($row = $result->fetch_assoc()) {
        $row_id = $row["id"];
        $row_reading_time = $row["reading_time"];
        $row_value1 = $row["temperature"];
        $row_value2 = $row["bpm"];

        // Uncomment to set timezone to - 1 hour (you can change 1 to any
number)
        // $row_reading_time = date("Y-m-d H:i:s",
strtotime("$row_reading_time - 1 hours"));

        // Uncomment to set timezone to + 4 hours (you can change 4 to any
number)
        // $row_reading_time = date("Y-m-d H:i:s",
strtotime("$row_reading_time + 4 hours"));

        echo '<tr>
            <td>' . $row_id . '</td>
            <td>' . $row_reading_time . '</td>
            <td>' . $row_value1 . '</td>
            <td>' . $row_value2 . '</td>

        </tr>';
    }
    $result->free();
}

$conn->close();
?>
</table>

</body>
</html>

</body>
</html>

```

Η εντολή **echo** καθοδηγεί τις στήλες που θα εμφανίζονται, πιο συγκεκριμένα με ποια σειρά θα εμφανίζονται τα δεδομένα, τί ονομασία έχει η κάθε στήλη και το ποια δεδομένα θα καταχωρούνται στην κάθε στήλη. Η εντολή **\$sql** επιλέγει τα δεδομένα του πίνακα **sensordata** και τα τοποθετεί στις αντίστοιχες στήλες που εμείς ορίζουμε. Επίσης έχουμε ήδη δει την εντολή `<body>`, μέσα στην οποία υλοποιείται όλη η μορφοποίηση της πληροφορίας που θέλουμε να δει ο χρήστης στον περιηγητή που χρησιμοποιεί. Όλα αυτά μπορούν να πραγματοποιηθούν μόνο εφόσον δημιουργηθεί η διασύνδεση μεταξύ της βάσης δεδομένων και του Server με την εντολή **\$conn**, την οποία και περιγράψαμε παραπάνω.

Όσον αφορά την τροποποίηση της εμφάνισης της ιστοσελίδας μας, όπως ήδη προαναφέραμε, χρησιμοποιήθηκε το αρχείο **style.css**. Με την γλώσσα CSS περιγράφεται η παρουσίαση της ιστοσελίδας μας, συμπεριλαμβανομένων των χρωμάτων, των διατάξεων, του μεγέθους των γραμμάτων, της μεταξύ τους απόστασης, των γραμματοσειρών που έχουμε επιλέξει κλπ. Ο αντίστοιχος κώδικας παρατίθεται παρακάτω.

```

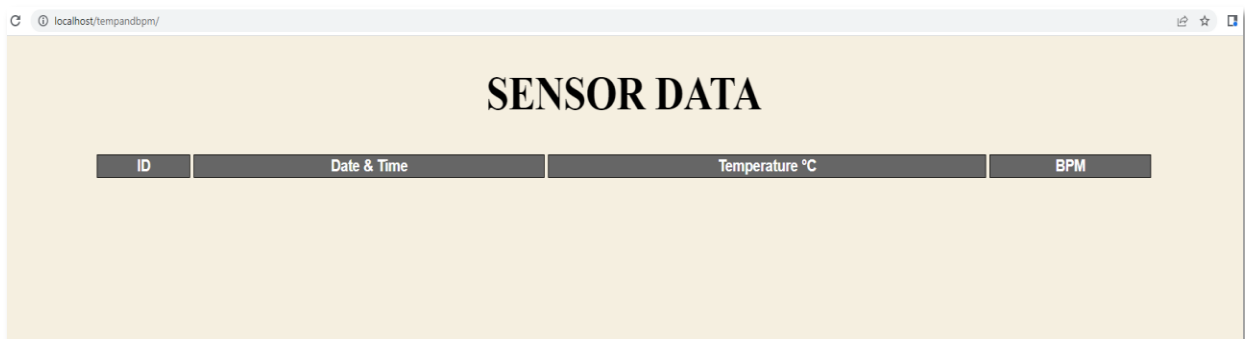
body {
  background: #f5efe0;
  box-sizing: border-box;
  color: #000;
  font-size: 1.8rem;
  letter-spacing: -0.015em;
  text-align: center;
}
table {
  margin-left: auto;
  margin-right: auto;
  width: 80%;
}

th {
  font-family: Arial, Helvetica, sans-serif;
  font-size: 20px;
  background: #666;
  color: #FFF;
  padding: 2px 6px;
  border-collapse: separate;
  border: 1px solid #000;
}

td {
  font-family: Arial, Helvetica, sans-serif;
  font-size: 15px;
  text-align: center;
  border: 1px solid #DDD;
}

```

Με την ολοκλήρωση της ιστοσελίδας το μόνο που μας απομένει είναι να δούμε πώς φαίνεται. Για αυτό τον λόγο χρησιμοποιούμε το URL: <http://localhost/tempandbpm/>. Παρακάτω παρουσιάζεται η εμφάνιση της ιστοσελίδας που κατασκευάστηκε.

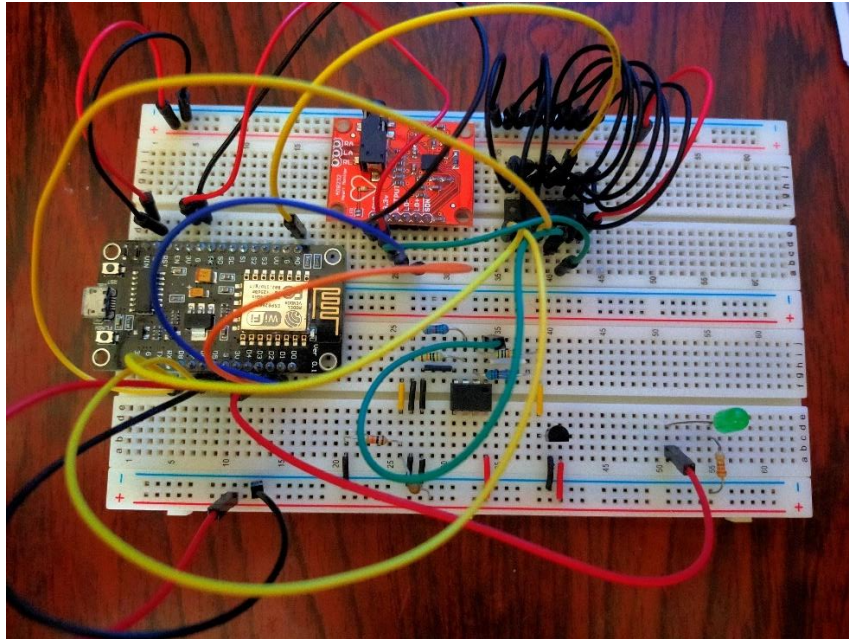


Εικόνα 5-6: Εμφάνιση της Ιστοσελίδας

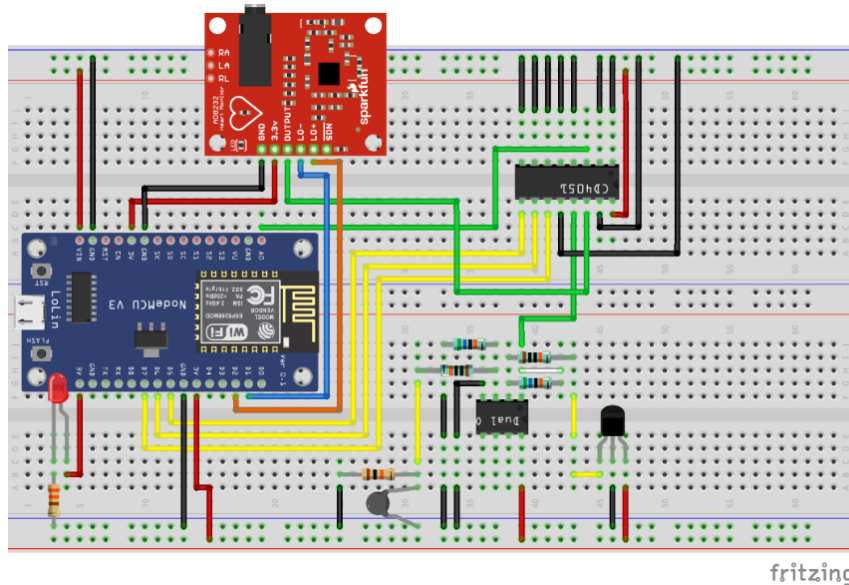
5.5 Παρουσίαση Συσκευής

Έχοντας ολοκληρώσει την περιγραφή της σύνδεσης της συσκευής μας στον Server, της δημιουργίας της βάσης δεδομένων και την κατασκευή της ιστοσελίδας όπου θα εμφανιστούν τα δεδομένα από τον μικροελεγκτή ESP8266, μπορούμε πλέον να παρουσιάσουμε την ολοκληρωμένη συσκευή μας και στο τέλος να πραγματοποιηθεί η εξαγωγή συμπερασμάτων.

Παρακάτω παρουσιάζονται η τελική μορφή της συσκευής και μία αντίστοιχη γραφική απεικόνισή της στο πρόγραμμα Fritzing.

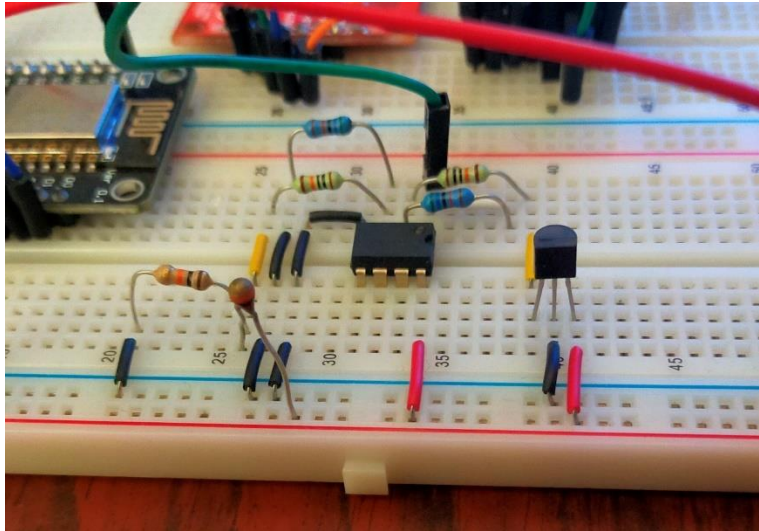


Εικόνα 5-7: Η τελική μορφή της συσκευής



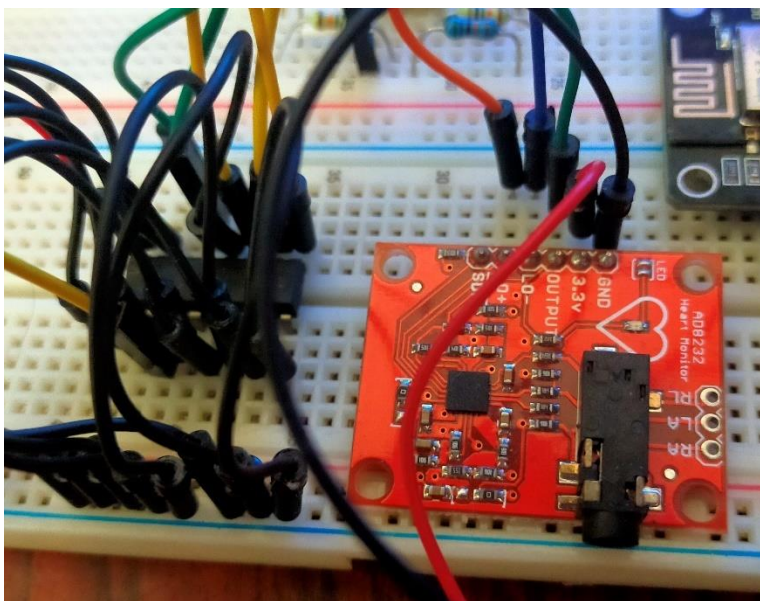
Διάγραμμα 5-5: Η γραφική απεικόνιση της συσκευής στο Fritzing

Όπως φαίνεται από την παραπάνω εικόνα το θερμόμετρο βρίσκεται στο κάτω μέρος της συσκευής. Όταν αγγίξουμε τον θερμίστορ με τα δάχτυλα μας τότε θα υπάρξει μεταβολή της θερμοκρασίας.



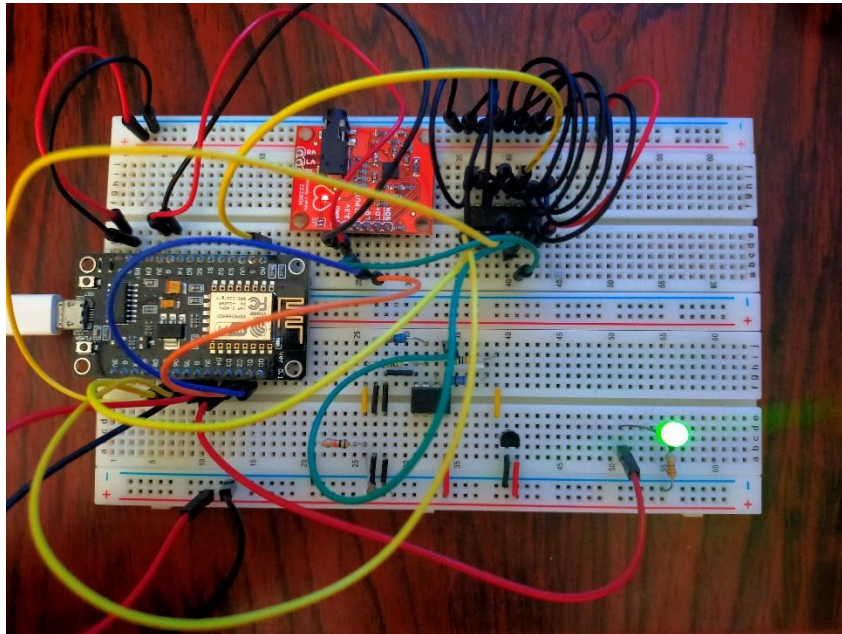
Εικόνα 5-8: Το κυκλωματικό μέρος του θερμομέτρου

Στο άνω μέρος της συσκευής βρίσκεται ο καρδιογράφος και ο πολυπλέκτης. Ο καρδιογράφος έχει μία είσοδο υποδοχής καλωδίου όπου θα συνδέσουμε το καλώδιό μας με τα τρία ηλεκτρόδια και από εκεί θα λάβουμε το ΗΚΓ.



Εικόνα 5-9: Το κυκλωματικό μέρος του ΗΚΓ & του πολυπλέκτη

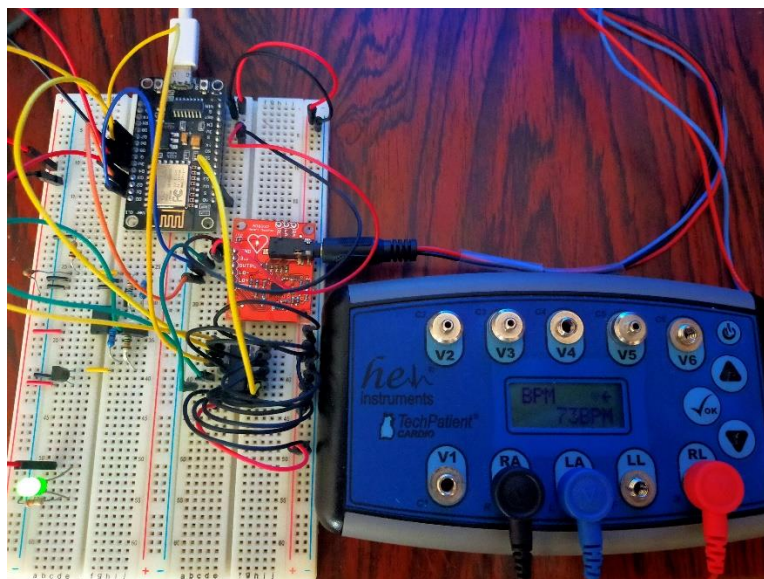
Στην τελική συσκευή προσθέσαμε ένα LED το οποίο θα ανάβει μόλις η συσκευή συνδεθεί στο διαδίκτυο. Αρχικά πρέπει να κάνουμε upload τον τελικό κώδικα που έχουμε γράψει στο Arduino IDE και στη συνέχεια να ενεργοποιήσουμε τις επιλογές MySQL και Apache στον UniServer για την διασύνδεση στον Server και την εμφάνιση των αποτελεσμάτων στην ιστοσελίδα.



Εικόνα 5-10: Αναμμένο LED έπειτα από σύνδεση στο Wi-Fi

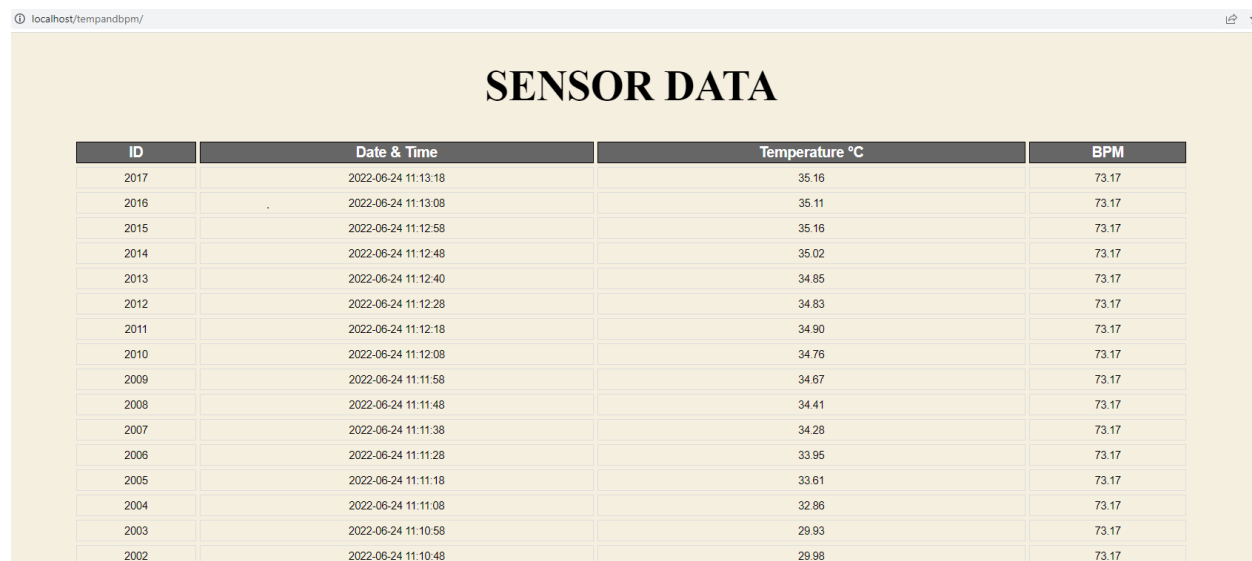
Έπειτα πληκτρολογούμε το URL <http://localhost/tempandbpm/> σε ένα web browser της επιλογής μας (όπως π.χ. Google Chrome) και θα εμφανιστεί η ιστοσελίδα μας.

Πρώτα έγινε δοκιμή του προσομοιωτή καρδιάς στην συσκευή μας. Ταυτόχρονα μετρήθηκε η θερμοκρασία με την χρήση των δακτύλων.



Εικόνα 5-11: Μέτρηση ΗΚΓ με την χρήση προσομοιωτή

Η τιμή του BPM που δώσαμε στον προσομοιωτή είναι 73 παλμοί το λεπτό, όπως φαίνεται και από την παραπάνω εικόνα και η θερμοκρασία την ημέρα και ώρα που έγιναν οι μετρήσεις ήταν περίπου 30 βαθμούς Κελσίου. Τα αποτελέσματα των μετρήσεων εμφανίζονται παρακάτω.



The screenshot shows a web browser window with the URL 'localhost/tempandbpm/'. The page title is 'SENSOR DATA'. Below the title is a table with four columns: ID, Date & Time, Temperature °C, and BPM. The table contains 15 rows of data, showing a steady increase in temperature from 29.98°C to 35.16°C and a steady decrease in BPM from 73.17 to 29.98.

ID	Date & Time	Temperature °C	BPM
2017	2022-06-24 11:13:18	35.16	73.17
2016	2022-06-24 11:13:08	35.11	73.17
2015	2022-06-24 11:12:58	35.16	73.17
2014	2022-06-24 11:12:48	35.02	73.17
2013	2022-06-24 11:12:40	34.85	73.17
2012	2022-06-24 11:12:28	34.83	73.17
2011	2022-06-24 11:12:18	34.90	73.17
2010	2022-06-24 11:12:08	34.76	73.17
2009	2022-06-24 11:11:58	34.67	73.17
2008	2022-06-24 11:11:48	34.41	73.17
2007	2022-06-24 11:11:38	34.28	73.17
2006	2022-06-24 11:11:28	33.95	73.17
2005	2022-06-24 11:11:18	33.61	73.17
2004	2022-06-24 11:11:08	32.86	73.17
2003	2022-06-24 11:10:58	29.93	73.17
2002	2022-06-24 11:10:48	29.98	73.17

Εικόνα 5-12: Αποτελέσματα μετρήσεων με τον προσομοιωτή

Όπως παρατηρούμε το BPM παραμένει αρκετά σταθερό χωρίς ιδιαίτερες μεταβολές και με αρκετά καλή μέτρηση της τιμής των καρδιακών παλμών. Το πλεονέκτημα του προσομοιωτή είναι πως οι τιμές που δίνει είναι αρκετά σταθερές επειδή αποκόπτει εξωτερικά παράσιτα που θα μπορούσαν να επηρεάσουν το σήμα, ενώ εκμηδενίζει την πιθανή αλλοίωση των μετρήσεων αφού μπορεί να παραμείνει τελείως ακίνητο σε αντίθεση με έναν ζωντανό οργανισμό. Όσον αφορά την θερμοκρασία, παρατηρούμε ομαλή μεταβολή της θερμοκρασίας κατά την διάρκεια που τα δάχτυλα αγγίζουν τον θερμίστορ (από 30 στους 35 βαθμούς Κελσίου), ενώ η θερμοκρασία βρίσκεται σε λογικά επίπεδα από την στιγμή που γνωρίζουμε πως η θερμοκρασία των δαχτύλων είναι χαμηλότερη κατά μερικούς βαθμούς Κελσίου σε σχέση με την αντίστοιχη μέτρηση μέσω του στόματος ή του πρωκτού.

Για την ακρίβεια της μέτρησης της θερμοκρασίας χρησιμοποιήθηκαν 2 θερμομέτρα ως μέτρο σύγκρισης, ένα υδραργυρικό και ένα ηλεκτρονικό. Το υδραργυρικό θερμομέτρο μέτρησε θερμοκρασία στους 35 βαθμούς Κελσίου ενώ το ηλεκτρονικό στους 34,8. Το θερμομέτρο της συσκευής μας σταθεροποιήθηκε περίπου στους 35,1 βαθμούς Κελσίου, δεδομένο που το καθιστά αρκετά αξιόπιστο από την στιγμή που η μέγιστη απόκλισή του, σε σχέση με τα άλλα δύο θερμομέτρα, ήταν της τάξης των 0,3 βαθμών Κελσίου. Σε αυτό το σημείο αξίζει να σημειωθεί πως η θερμοκρασία που μετρείται στα δάχτυλα μπορεί να επηρεαστεί από μερικούς παράγοντες όπως η εποχή στην οποία γίνεται η συγκεκριμένη μέτρηση (π.χ. καλοκαίρι, χειμώνας) και το φύλο του εξεταζόμενου ατόμου. Παρακάτω παρατίθενται οι εικόνες των μετρήσεων των δύο θερμομέτρων.



Εικόνα 5-13: Αποτέλεσμα μέτρησης θερμοκρασίας δαχτύλου με ηλεκτρονικό θερμόμετρο



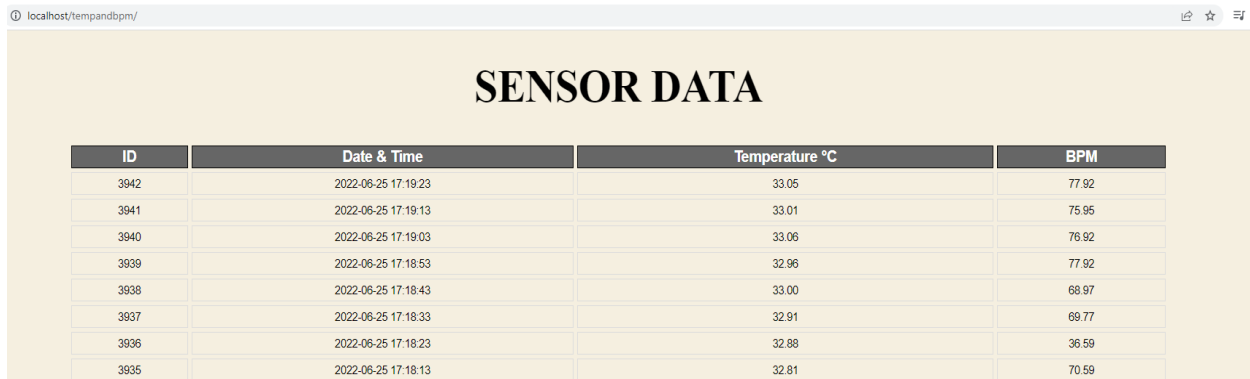
Εικόνα 5-14: Αποτέλεσμα μέτρησης θερμοκρασίας δαχτύλου με υδραργυρικό θερμόμετρο

Η δεύτερη δοκιμή που ακολουθήσε ήταν αυτή της χρήσης των ηλεκτροδίων με pads κατά Einthoven πάνω στο ανθρώπινο σώμα. Παρακάτω παρουσιάζεται η αντίστοιχη εικόνα της τοποθέτησης των ηλεκτροδίων στο σώμα.



Εικόνα 5-15: Τοποθέτηση των ηλεκτροδίων στο ανθρώπινο σώμα

Τα αποτελέσματα των μετρήσεων εμφανίζονται παρακάτω.



ID	Date & Time	Temperature °C	BPM
3942	2022-06-25 17:19:23	33.05	77.92
3941	2022-06-25 17:19:13	33.01	75.95
3940	2022-06-25 17:19:03	33.06	76.92
3939	2022-06-25 17:18:53	32.96	77.92
3938	2022-06-25 17:18:43	33.00	68.97
3937	2022-06-25 17:18:33	32.91	69.77
3936	2022-06-25 17:18:23	32.88	36.59
3935	2022-06-25 17:18:13	32.81	70.59

Εικόνα 5-16: Αποτελέσματα μετρήσεων με τον προσομοιωτή

Στην συγκεκριμένη περίπτωση αυτό που μας ενδιαφέρει είναι να δούμε πως συμπεριφέρεται ο ΗΚΓ μας στην περίπτωση όπου οι μετρήσεις παρθούν από το ανθρώπινο σώμα. Όπως παρατηρούμε το BPM που μετρήθηκε κυμάνθηκε μεταξύ των 70-77 παλμών ανά το λεπτό, το οποίο ανήκει στα φυσιολογικά όρια του BPM ενός ανθρώπου σε κατάσταση σχετικής ηρεμίας (60-100). Είναι προφανές το γεγονός πως εξωγενείς παράγοντες επηρεάζουν αυτή την σχετικά μικρή μεταβολή του καρδιακού ρυθμού, ιδιαιτέρως λόγω των μικρών κινήσεων του σώματος που δεν μπορούν να ελεγχθούν από τον ίδιο τον άνθρωπο με απόλυτη ακρίβεια. Για αυτό και είναι σημαντικό να μην κινείται ο εξεταζόμενος κατά την διάρκεια της εξέτασης, ώστε να μην υπάρξει λάθος μέτρηση.

Ο τελικός κώδικας του προγράμματός μας παρατίθεται παρακάτω.

Κώδικας υπολογισμού και εμφάνισης Θερμοκρασίας & BPM στην ιστοσελίδα

```
//Libraries for establishing ESP8266 connection to Wi-Fi network
#ifdef ESP32
#include <WiFi.h>
#include <HTTPClient.h>
#else
#include <ESP8266WiFi.h>
#include <ESP8266HTTPClient.h>
#include <WiFiClient.h>
#endif
//-----
//Selection of digital pins which will record the two different analog
inputs
const int selectPins[3] = {D7, D6, D5};
//-----
//Parameters for Temperature (Temp)
const float Va = 1.62;
const float Ad = 5.6;
const float R4 = 9.8;
const float Beta = 3977;
const float T0 = 298.15;
float Temp = 0;
//-----
```

```

//Parameters for Heart Rate (BPM)
int UpperThreshold = 800;
int LowerThreshold = 500;
int reading = 0;
float BPM = 0.0;
float BeatRate;
bool IgnoreReading = false;
bool FirstPulseDetected = false;
unsigned long FirstPulseTime = 0;
unsigned long SecondPulseTime = 0;
unsigned long PulseInterval = 0;
const unsigned long delayTime = 10;
const unsigned long delayTime2 = 10000;
unsigned long previousMillis = 0;
unsigned long previousMillis2 = 0;
//-----

// Replace with your network credentials
const char* ssid      = "OTe30d80";
const char* password = "15a14A57";

// REPLACE with your Domain name and URL path or IP address with path
const char* serverName = "http://192.168.1.3/tempandbpm/post-esp-data.php";

// Keep this API Key value to be compatible with the PHP code provided in
the project page.
// If you change the apiKeyValue value, the PHP file /post-esp-data.php
also needs to have the same key
String apiKeyValue = "tPmAT5Ab3j7F9";
//-----

const int LED = 0;

void setup() {
  pinMode(LED, OUTPUT);

  //Establishing connection to Wi-Fi network
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
  }

  // Set up the select pins as outputs:
  for (int i = 0; i < 3; i++)
  {
    pinMode(selectPins[i], OUTPUT);
    digitalWrite(selectPins[i], LOW);
  }
}
//-----

void loop() {
  digitalWrite(selectPins[0], HIGH); //When it's HIGH record data for
temperature

```

```

int m = analogRead(A0); //Read analog input for BPM

// Get current time
unsigned long currentMillis = millis();

// First event - finding the peaks of the PQRST signal
if (myTimer1(delayTime, currentMillis) == 1)
{
    reading = m;

    // Heart beat leading edge detected.
    if (reading > UpperThreshold && IgnoreReading == false)
    {
        if (FirstPulseDetected == false) {
            FirstPulseTime = millis();
            FirstPulseDetected = true;
        }
        else {
            SecondPulseTime = millis();
            PulseInterval = SecondPulseTime - FirstPulseTime;
            FirstPulseTime = SecondPulseTime;
        }
        IgnoreReading = true;
    }

    // Heart beat trailing edge detected.
    if (reading < LowerThreshold && IgnoreReading == true) {
        IgnoreReading = false;
    }

    // Calculate Beats Per Minute.
    if (PulseInterval != 0) { //Check if PulseInterval is different than 0
        BPM = 60000.0 / PulseInterval;
    }
}
if (myTimer2(delayTime2, currentMillis) == 1) {

    digitalWrite(selectPins[0], LOW); //When it's LOW record data for BPM
    delay(10);
    int n = analogRead(A0); //Read analog input for Temperature

    float Vout = (n * 3.3) / 1023; //Conversion of analog input to
    // Voltage
    float V21 = Vout / Ad; //Calculate voltage difference
    float Vb = V21 + Va; //Calculate voltage between the
    // thermistor and resistance R4
    float RT = R4 * ((3.3 / Vb) - 1); //Calculate thermistor's resistor
    // value
    float T = (1.0 / T0) + (1.0 / Beta) * log(RT / 10); //read temperature
    // in Kelvin
    T = 1 / T;
    Temp = T - 273.15; //Convert Kelvin to Celsius

    //Check WiFi connection status
    if (WiFi.status() == WL_CONNECTED) {
        digitalWrite(LED, HIGH); // turn the LED on
    }
}

```

```

WiFiClient client;
HTTPClient http;

// Your Domain name with URL path or IP address with path
http.begin(client, serverName);

// Specify content-type header
http.addHeader("Content-Type", "application/x-www-form-urlencoded");

// Prepare your HTTP POST request data
String httpRequestData = "api_key=" + apiKeyValue + "&temp=" +
String(Temp) + "&bpm=" + String(BPM) + "";
Serial.print("httpRequestData: ");
Serial.println(httpRequestData);

int httpResponseCode = http.POST(httpRequestData);

if (httpResponseCode > 0) {
}
else {
    Serial.println(httpResponseCode);
}

// Free resources
http.end();
}
else {
    digitalWrite(LED, LOW); // turn the LED off
}
}
}

// First event timer
int myTimer1(long delayTime, long currentMillis) {
    if (currentMillis - previousMillis >= delayTime) {
        previousMillis = currentMillis;
        return 1;
    }
    else {
        return 0;
    }
}

// Second event timer
int myTimer2(long delayTime2, long currentMillis) {
    if (currentMillis - previousMillis2 >= delayTime2) {
        previousMillis2 = currentMillis;
        return 1;
    }
    else {
        return 0;
    }
}
}

```

ΑΞΙΟΛΟΓΗΣΗ ΜΕΘΟΔΟΛΟΓΙΑΣ

Όπως παρουσιάστηκε στα προηγούμενα κεφάλαια, καταφέραμε να σχεδιάσουμε και να κατασκευάσουμε μία φορητή συσκευή η οποία είναι ικανή να υπολογίσει την θερμοκρασία και τον ρυθμό του καρδιακού παλμού σε πραγματικό χρόνο. Η επαλήθευση της μεθοδολογίας που υλοποιήθηκε είναι ίσως και η πιο σημαντική παρατήρηση που πρέπει να γίνει στο παρόν κεφάλαιο. Τα αποτελέσματα στην οθόνη μαρτυρούν ικανοποιητικές μετρήσεις από τα λαμβανόμενα σήματα. Η συσκευή μας δίνει τη δυνατότητα να ελέγχουμε τις συγκεκριμένες παραμέτρους και όλα αυτά με αρκετά χαμηλή κατανάλωση ενέργειας.

Η σύλληψη των αλγορίθμων που συλλέγουν τα σήματα έγινε κάνοντας χρήση των αισθητήριων οργάνων πάνω στο σώμα του συγγραφέα. Η σωστή τεχνική δημιουργίας τέτοιων τρόπων λύσης ενός προβλήματος γίνεται κάνοντας επαλήθευση για τον αλγόριθμο που σχεδιάστηκε και λειτουργεί το ίδιο αξιόπιστα σε άλλα δείγματα ατόμων. Μπορεί μεν η μεθοδολογία που σχεδίασε ένας μηχανικός να λειτουργεί στον ίδιο, αλλά για να γίνει περαιτέρω συνέχεια του έργου του, πρέπει να επαληθεύσει πως λειτουργεί το ίδιο καλά και σε ένα δείγμα ανθρώπων. Στην παρούσα εργασία, όσον αφορά την μεθοδολογία εύρεσης του καρδιακού παλμού και της μέτρησης της θερμοκρασίας, τα άτομα πάνω στα οποία δοκιμάστηκε είναι ο ίδιος ο συγγραφέας και 1-2 στενά του πρόσωπα. Το μόνο άτομο δηλαδή που ήλεγχε την αξιοπιστία της συσκευής είναι ο συγγραφέας, έχοντας στο νου του τη βιβλιογραφία και το υπόλοιπο υλικό που υπάρχει επί του αντικειμένου.

Όπως ήδη αναφέρθηκε κατά την εύρεση του καρδιακού παλμού, η χρήση του ΗΚΓ καθώς και της μεθόδου που αναλύθηκε ενέχει και τα σφάλματά της. Οι μικρές κινήσεις του σώματος είναι πολύ πιθανό να αλλοιώσουν τα αποτελέσματα, καθώς αυτές επηρεάζουν την σχετική ηρεμία της καρδιακής λειτουργίας και επομένως μπορεί να μεταφραστούν σε λανθασμένη μέτρηση. Ταυτόχρονα, έχει σημασία ο τρόπος με τον οποίο ανασαίνει ο ασθενής καθώς ο καρδιακός ρυθμός (HR) ποικίλλει ανάλογα με την αναπνοή. Η σωστή μέτρηση γίνεται μόνο εφόσον η εισπνοή και η εκπνοή του ασθενούς είναι βαθιά και αργή ώστε η καρδιακή λειτουργία να βρίσκεται σε κατάσταση χαλάρωσης. Σε οποιαδήποτε άλλη περίπτωση τα αποτελέσματα δεν θα είναι αντικειμενικά και θα αποκλίνουν αρκετά μεταξύ τους. Άλλος ένας περιορισμός είναι η στάση του σώματος κατά τη διάρκεια της μέτρησης. Ο εξεταζόμενος πρέπει να είναι ξαπλωμένος σε οριζόντια θέση και να μην είναι σφιγμένος.

Ως προς την μέτρηση της θερμοκρασίας, όσον αφορά την αξιοπιστία, όταν ο εξεταζόμενος ακουμπήσει τα δάχτυλα του στον αισθητήρα τότε η θερμοκρασία της συσκευής σταδιακά αρχίζει να ταυτίζεται με αυτήν του δέρματος και τα αποτελέσματα είναι αρκετά ικανοποιητικά. Ένας περιορισμός όμως της συσκευής είναι αυτή η χρονική διάρκεια που απαιτείται ώστε η συσκευή να φτάσει στα θερμοκρασιακά επίπεδα του σώματος. Τα πρώτα λεπτά ύστερα από την εφαρμογή της διάταξης στον ασθενή, θα αποτελούνται από μη ακριβή δείγματα, ενώ εάν ο ασθενής αφαιρεί συχνά τα δάχτυλα του από τον αισθητήρα, τα αποτελέσματα δεν θα είναι αντιπροσωπευτικά.

Άλλος ένας περιορισμός του συστήματος αφορά τη σύνδεση της συσκευής στο δίκτυο Wi-Fi και της επικοινωνίας του με τον Server. Η συσκευή θα πρέπει να γνωρίζει το όνομα του δικτύου, τον κωδικό του, καθώς και την local IP του υπολογιστή στον οποίο υπάρχει ο Server. Παρόλα αυτά, ένα μεγάλο πλεονέκτημα είναι πως οποιοσδήποτε μπορεί να δει τα αποτελέσματα των συσκευών μέσω ενός browser, είτε από κινητό είτε από υπολογιστή, αρκεί η συσκευή του να ανήκει στο ίδιο δίκτυο Wi-Fi. Για παράδειγμα ένας ιατρός θα μπορούσε από το γραφείο του να παρακολουθεί έναν ασθενή που βρίσκεται σε διαφορετικό δωμάτιο του ίδιου χώρου (ώστε να «ανήκουν» και οι δύο στο ίδιο δίκτυο).

ΠΙΘΑΝΕΣ ΜΕΛΛΟΝΤΙΚΕΣ ΥΛΟΠΟΙΗΣΕΙΣ

Το σύστημα μας έχει δυνατότητες περαιτέρω εξέλιξης. Αρχικά, θα μπορούσε να προστεθούν αισθητήρες στις συσκευές με σκοπό να γίνεται μία πιο ολοκληρωμένη καταγραφή ζωτικών σημάτων, όπως αισθητήρες ρυθμού αναπνοής, αρτηριακής πίεσης και άλλοι. Τα σήματα αυτά θα συμπλήρωναν τα υπόλοιπα ώστε να δημιουργήσουν μία σαφέστερη ανάλυση της κατάστασης του ασθενή. Έπειτα, για εξοικονόμηση χώρου, η συσκευή θα μπορούσε να περαστεί σε πλακέτα PCB για μία πιο σταθερή και χωρητικά οικονομικότερη λύση.

Όσον αφορά την μεριά του Server, για το όνομα του δικτύου και τον κωδικό, αυτά θα μπορούσαν να αποθηκεύονται σε μία SD κάρτα, την οποία θα ενσωματώναμε στην συσκευή, σε κάποιο αρχείο το οποίο θα προσπέλαζε η συσκευή ώστε να τα ανακτήσει. Έτσι, εάν άλλαζε το όνομα ή ο κωδικός του δικτύου, πολύ απλά θα σημειώνονταν η αλλαγή στην SD κάρτα και δεν θα απαιτούνταν να προγραμματιστεί η αλλαγή μέσω καλωδίου USB. Κάποιος που θα παρακολουθούσε τον ασθενή, θα μπορούσε να βρίσκεται σε διαφορετικό μέρος και θα είχε τη δυνατότητα να προβάλλει την κατάστασή του.

Επιπρόσθετα, ανοίγεται με αυτόν τον τρόπο η δυνατότητα κατασκευής πολλών τέτοιων συσκευών που θα παρακολουθούν έναν αριθμό ασθενών. Με τη δημιουργία μίας πλατφόρμας για πολλούς ασθενείς και ιατρικό προσωπικό, θα μπορούσε κάποιος να εισάγει τα στοιχεία του ως ιατρός (για authentication), να επιλέγει τον ασθενή που τον ενδιαφέρει και ύστερα να οδηγείται στη σελίδα επιλογής χρονικού παραθύρου.

Μία ακόμη σημαντική προσθήκη θα ήταν η δημιουργία ενός συστήματος ειδοποιήσεων στην περίπτωση που ο ασθενής δείξει κάποια σημάδια επικινδυνότητας. Συγκρίνοντας τα αποτελέσματα που έχουν ήδη εξαχθεί, θα μπορούσε να υπάρχει μία ένδειξη για την κατάσταση του ασθενή όποτε προβάλλονται τα καταγεγραμμένα σήματά του όπως «Υγιής» ή «Υψηλός ρυθμός αναπνοής» και σε περιπτώσεις που κάποιες τιμές ξεπεράσουν ένα κατώφλι, να στέλνεται ένα σήμα ειδοποίησης

ΣΥΜΠΕΡΑΣΜΑΤΑ

Συνοψίζοντας αντιλαμβανόμαστε ότι ζούμε και αναπτυσσόμαστε σε μία διαρκώς τεχνολογικά μεταβαλλόμενη κοινωνία. Ο τομέας του ελέγχου ζωτικών σημείων και των εφαρμογών ηλεκτρονικής υγείας διερευνάται και εξελίσσεται εκθετικά. Οι παροχές υπηρεσιών αυξάνουν και ποικίλουν χάρη στα νέα συστήματα που υλοποιούνται για τους σκοπούς της υγείας και παράγεται ένα αποτέλεσμα που προσδίδει αξία σε συνολικό αριθμό προβλημάτων που ένα άτομο μπορεί να αντιμετωπίσει. Με ορισμένες τροποποιήσεις στην συσκευή, θα μπορούσε να δημιουργηθεί ένα εμπορικό προϊόν το οποίο θα κάλυπτε την ανάγκη ολοκληρωμένης παρακολούθησης ενός ασθενή με αρκετά χαμηλό κόστος. Παράλληλα σε συνδυασμό με τον απομακρυσμένο έλεγχο του ασθενή, μας παρέχεται η ευκαιρία να αντιληφθούμε την υπέρτατη αξία της πρόληψης και των δυνατοτήτων που μας προσφέρει. Συνεπώς κατανοούμε την αξία της συσκευής αυτής που υλοποιήθηκε όχι μόνο σε επίπεδο παροχών υπηρεσιών αλλά και σε επιστημονικό επίπεδο της έρευνας και της μελέτης.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] Κ.Δ. Αλεξόπουλος, Δ. Μαρίνος, Φυσική, τ. Θερμότητα, Αθήνα 1970.
- [2] «Human Body Temperature», Wikipedia, [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Human_body_temperature.
- [3] I. Geneva, B. Cuzzo, T. Fazili και W. Javaid, «Normal Body Temperature: A Systematic Review.,» Open Forum Infect Dis., Apr 2019.
- [4] N. Ambardekar, «What Happens to Your Body When You Sleep?,» WebMD, 2006.
- [5] A. A. Romanovsky, «Skin temperature: its role in thermoregulation,» Acta Physiol (Oxf), τόμ. 2014 Mar 210(3): 498–507, 2014 Mar.
- [6] Β. Κώτσης, Διευθυντής Καρδιοχειρουργικής Κλινικής Ιατρικού Κέντρου αθηνών, άρθρο: «Η λειτουργία της Καρδιάς».
- [7] «Ηλεκτροφυσιολογία της καρδιάς», Wikipedia, Παράγραφος: Ηλεκτρική δραστηριότητα του κυττάρου. Available: https://el.wikipedia.org/wiki/Ηλεκτροφυσιολογία_της_καρδιάς.
- [8] «Το ηλεκτρικό σύστημα της καρδιάς & το ΗΚΓ», Heart Rhythm Center, Παράγραφος: Το ηλεκτρικό σύστημα της καρδιάς & το ΗΚΓ, Available: <http://www.eponline.gr/ilektiko-sistima-kardias-erethismatagogo-systima-kardias/>.
- [9] «Ηλεκτροφυσιολογία της καρδιάς», Wikipedia, Παράγραφος: Ηλεκτροκαρδιογράφημα. Available: https://el.wikipedia.org/wiki/Ηλεκτροφυσιολογία_της_καρδιάς.
- [10] Shinwari MW, Zhitomirsky D, Deen IA, Selvaganapathy PR, Deen MJ, Landheer D. Microfabricated Reference Electrodes and their Biosensing Applications. Sensors. 2010;10:1679–715.
- [11] «NodeMCU ESP8266 Detailed Review», Make-It.ca, Παράγραφος: What is NodeMCU?. Available: <https://www.make-it.ca/nodemcu-details-specifications/>.
- [12] «NodeMCU ESP8266», COMPONENTS101, Παράγραφος: Brief About NodeMCU ESP8266. Available: <https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet/>
- [13] «ESP8266 pinout reference and how to use GPIO pins», microcontrollerslab, Παράγραφος: ESP8266 12E NodeMCU Development Board pinout. Available: <https://microcontrollerslab.com/esp8266-pinout-reference-gpio-pins/>
- [14] «AD8232 Single Lead Heart Rate Monitor», GROBOTRONICS, Παράγραφος: DESCRIPTION. Available: <https://grobotronics.com/ad8232-single-lead-heart-rate-monitor.html/>
- [15] «AD8232 ECG Sensor», Utmel, Παράγραφος: AD8232 Description. Available: <https://www.utmel.com/components/ad8232-pinout-advantage-and-theory-of-operation?id=160/>
- [16] «AD8232 ECG Module with Arduino – Heart Rate Monitor», microcontrollerslab, Παράγραφος: Pin Configuration. Available: <https://microcontrollerslab.com/ad8232-ecg-module-pinout-interfacing-with-arduino-applications-features/>
- [17] «AD8232 Heart Rate Monitor Hookup Guide», sparkfun, Παράγραφος: Sensor Pad Placement. Available: <https://learn.sparkfun.com/tutorials/ad8232-heart-rate-monitor-hookup-guide/all/>
- [18] «AD8232 ECG Module with Arduino – Heart Rate Monitor», microcontrollerslab, Παράγραφος: AD8232 ECG module Features. Available: <https://microcontrollerslab.com/ad8232-ecg-module-pinout-interfacing-with-arduino-applications-features/>
- [19] S.Z.NASIR, «Introduction to CD4051. A complete step by step Introduction to CD4051», 2017.
- [20] «What Is An NTC Thermistor», AMETHERM, Παράγραφος: What Is An NTC Thermistor. Available: <https://www.ametherm.com/thermistor/what-is-an-ntc-thermistor/>
- [21] «NTC Thermistors, Radial Leaded, Standard Precision», Vishay Intertechnology, Παράγραφος: QUICK REFERENCE DATA, Datasheet.
- [22] «NTC Thermistors, Radial Leaded, Standard Precision», Vishay Intertechnology, Παράγραφος: FEATURES, Datasheet.

- [23] «Dual single-supply, rail-to-rail, low power operational amplifier», Texas Instruments, Παράγραφος: Description. Available: <https://www.ti.com/product/OPA2340#params/>
- [24] «Dual single-supply, rail-to-rail, low power operational amplifier», Texas Instruments, Παράγραφος: PINS. Available: <https://www.ti.com/product/OPA2340#params/>
- [25] «Dual single-supply, rail-to-rail, low power operational amplifier», Texas Instruments, Παράγραφος: Features. Available: <https://www.ti.com/product/OPA2340#params/>
- [26] «The 'Rail Splitter' Precision Virtual Ground», Texas Instruments, Παράγραφος: Description. Available: <https://www.ti.com/product/TLE2426#description/>
- [27] «A complete step by step tutorial on the Introduction to Arduino IDE», Technopreneur, Παράγραφος: INTRODUCTION TO ARDUINO IDE. Available: <https://www.theengineeringprojects.com/2018/10/introduction-to-arduino-ide.html/>
- [28] «AVR-GCC », Stanford. Available: https://ccrma.stanford.edu/~juanig/articles/wiriavrlib/AVR_GCC.html/
- [29] «Notepad++ ». Available: <http://notepad-plus-plus.org/>
- [30] «The Uniform Server – A WAMP WordPress Tutorial», WebTNG, Παράγραφος: Introduction. Available: <https://www.webtng.com/the-uniform-server-a-wamp-wordpress-tutorial/>
- [31] «Bringing MySQL to the web», phpMyAdmin, Παράγραφος: About. Available: <https://www.phpmyadmin.net/#:~:text=phpMyAdmin%20is%20a%20free%20software,operations%20on%20MySQL%20and%20MariaDB.>
- [32] «HTML», Wikipedia. Available: <https://el.wikipedia.org/wiki/HTML>
- [33] «CSS», Wikipedia. Available: <https://el.wikipedia.org/wiki/CSS>
- [34] «HPP», Wikipedia. Available: <https://el.wikipedia.org/wiki/PHP>
- [35] «MySQL», Wikipedia. Available: <https://el.wikipedia.org/wiki/MySQL>
- [36] «JavaScript», Wikipedia. Available: <https://en.wikipedia.org/wiki/JavaScript>
- [37] «Apache HTTP Server», Wikipedia. Available: https://en.wikipedia.org/wiki/Apache_HTTP_Server
- [38] «Wheatstone bridge», Wikipedia. Available: https://en.wikipedia.org/wiki/Wheatstone_bridge
- [39] «Electronic Thermometer», Science Direct, Παράγραφος: Vital Signs. Available: <https://www.sciencedirect.com/topics/nursing-and-health-professions/electronic-thermometer>
- [40] «Fritzing», Wikipedia. Available: <https://en.wikipedia.org/wiki/Fritzing>