



ΔΙΕΘΝΕΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΤΗΣ ΕΛΛΑΔΟΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ
ΣΥΣΤΗΜΑΤΩΝ



Διπλωματική Εργασία

Παλμικό οξύμετρο και μετρητής καρδιακών παλμών με διασύνδεση σε Android

Των φοιτητών:
Καρρά Νικόλαου
Φετφατζίδη Γεώργιου

Επιβλέπων Καθηγητής:

Ονοματεπώνυμο Βάσιος Βασίλειος
Βαθμίδα : Αναπληρωτής Καθηγητής

Αρ. Μητρώου: 513307
512192

Θεσσαλονίκη, 2020

Τίτλος Π.Ε. Παλμικό οξύμετρο και μετρητής καρδιακών παλμών με διασύνδεση σε Android

Κωδικός Π.Ε. 17173Σ

Όνοματεπώνυμο φοιτητών Νικόλαος Καρράς, Γεώργιος Φετφατζίδης

Όνοματεπώνυμο εισηγητή Βασίλειος Βάσσιος

Ημερομηνία ανάληψης Π.Ε. 05-04-2019

Ημερομηνία περάτωσης Π.Ε. 02-09-2020

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία των φοιτητών Καρρά Νικόλαου και Φετφατζίδα Γεώργιου που την εκπόνησαν. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιοδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

Πρόλογος

Ο πρωταρχικός στόχος αυτού του έργου είναι η ανάπτυξη ενός παλμικού οξυμέτρου το οποίο θα μετράει τον κορεσμό οξυγόνου στο αίμα και τους καρδιακούς παλμούς.

Χρησιμοποιήθηκε ένα ESP32 με ενσωματωμένη κεραία Bluetooth για τη μετάδοση των δεδομένων από τον αισθητήρα στο Smartphone και ένα πρόγραμμα Android αναπτύχθηκε ως μέρος του έργου για σύνδεση με τη μονάδα Bluetooth για λήψη και αποθήκευση των δεδομένων. Μόλις αναπτύχθηκε ο αισθητήρας και η εφαρμογή Android, ο ρυθμός παλμού και ο κορεσμός οξυγόνου συγκρίθηκαν με μετρήσεις που έγιναν από ένα εμπορικό παλμικό οξύμετρο ώστε να προσδιοριστεί η ακρίβεια της συσκευής.

Περίληψη

Ένα παλμικό οξύμετρο είναι ικανό να μετρήσει μη επεμβατικά την ποσότητα κορεσμένου οξυγόνου αιμοσφαιρίνης (οξυαιμοσφαιρίνη) στο αρτηριακό αίμα ενός ασθενούς. Καταλάβαμε το πόσο χρήσιμο και καταπληκτικό ήταν αυτό το επίτευγμα μέχρι το 1852 όταν ο Γερμανός φυσικός Όγκουστ Μπιρ απέδειξε ότι η ποσότητα του φωτός που μεταδίδεται μέσω ενός διαλύματος ποικίλλει ανάλογα με τη συγκέντρωση της διαλυμένης ουσίας. Αλλά η πρακτική εφαρμογή αυτής της ιδέας ήταν πολύ πιο δύσκολη.

Η παροχή υψηλής ποιότητας υγειονομικής περίθαλψης σε μαζικό πληθυσμό γίνεται μια από τις μεγαλύτερες προσπάθειες της σύγχρονης εποχής. Για να γίνει αυτό, πρέπει να υπάρχει μια ώθηση να αγκαλιάσουμε τη χρήση νέων τεχνολογιών που μπορεί να παρέχει άνεση, διασφαλίζοντας παράλληλα την ασφάλεια και την αξιοπιστία των παραδοσιακών μεθόδων. Αυτό το εργαλείο μπορεί να μετρήσει τον καρδιακό ρυθμό του ασθενούς και τα επίπεδα κορεσμού οξυγόνου εξασφαλίζοντας άνεση και εύκολη χρήση. Βασικά πλεονεκτήματα είναι το χαμηλό κόστος, ο χώρος που καταλαμβάνει και η διαθεσιμότητα πολλών λειτουργιών. Αυτό το σύστημα διαθέτει δύο βασικά στοιχεία, έναν μικροεπεξεργαστή ESP32 και έναν αισθητήρα MAX30100 PhotoPletysmoGraphy (PPG). Η μονάδα ESP32 επιλέχθηκε λόγω του επεξεργαστή (32-bit διπλού πυρήνα), με ενσωματωμένη μονάδα WiFi/Bluetooth και προκαθορισμένες λειτουργίες ύπνου για τη μείωση κατανάλωσης ενέργειας. Επιλέχθηκε ο αισθητήρας MAX30100 γιατί είναι μια μικρή μονάδα με απλή χρήση. Είναι πλήρως διαμορφώσιμο μέσω καταχωρητών λογισμικού και τα δεδομένα ψηφιακής εξόδου αποθηκεύονται σε ένα FIFO 16 βαθμίδων μέσα στη συσκευή. Το FIFO του επιτρέπει να συνδεθεί σε μικροελεγκτή ή μικροεπεξεργαστή σε κοινόχρηστο δίαυλο, όπου τα δεδομένα δεν διαβάζονται συνεχώς από το μητρώο της συσκευής.

Abstract

A pulse oximeter is capable of measuring non-invasively the amount of saturated hemoglobin oxygen (oxyhemoglobin) in a patient's arterial blood. We realized how useful and amazing this achievement was until 1852 when the German physicist August Beer proved that the amount of light transmitted through a solution varies depending on the concentration of the solute. But the practical application of this idea was much more difficult.

Providing high quality health care to the mass population is becoming one of the great efforts of modern society. To do this, there is an impetus to embrace the use of new technologies that can provide comfort while ensuring the safety and reliability of traditional methods. This product can measure the patient's heart rate and oxygen saturation levels ensuring comfort and ease of use. Other advantages are the low cost, the small size and its utilities. This system has two basic components, an ESP32 microprocessor and a MAX30100 PhotoPletysmoGraphy (PPG) sensor. The ESP32 module was selected because of its processor (32-bit dual-core), with a WiFi module and 3 preset sleep modes to reduce power consumption. The MAX30100 sensor was chosen because its small and simple to use. Its fully configurable via software registers and digital output data is stored in a 16-degree FIFO inside the device. FIFO allows it to connect to a microcontroller or microprocessor on a shared bus, where data is not read continuously from the device registry.

Ευχαριστίες

Σε αυτό το σημείο, θα θέλαμε να ευχαριστήσουμε ιδιαίτερα τον επιβλέπον καθηγητή κύριο Βάσσιο Βασίλειο για την στήριξη και μεγάλη βοήθεια που κέντρισε το ενδιαφέρον μας για περαιτέρω ενασχόληση.

Επίσης θα θέλαμε να ευχαριστήσουμε τους γονείς για την υποστήριξη και τη βοήθεια που μας πρόσφεραν και δεν σταμάτησαν να είναι εκεί για εμάς, να δείχνουν πάντα ενδιαφέρον και να μας βοηθούν ανεξαρτήτως.

Ένα ιδιαίτερο ευχαριστώ στον αγαπητό φίλο και συνάδελφο Τοπαλίδη Τιμόθεο για το ενδιαφέρον που έδειξε και τις συμβουλές που πρόσφερε για την ολοκλήρωση της παρούσας πτυχιακής.

Τέλος οφείλουμε να ευχαριστήσουμε τους φίλους και συναδέλφους για τις όμορφες στιγμές που περάσαμε μαζί, την υποστήριξη και την αγάπη που έδειξαν.

Συντομογραφίες

Hb (Hemoglobin)

HbO2 (Oxyhemoglobin)

SpO2 (Saturation of Periopheral Oxygen)

BPM (Beats Per Minute)

IR (Infrared)

ID (Infrared Red)

LED (Light-Emitted diode)

ADC (Analog to Digital Converter)

DSC (Digital Signal Controller)

MCU (Microcontroller Unit)

PWM (Pulse Width Modulation)

AGB (Arterial Blood Gases)

GATT (Generic Attribute Profile)

ATT (Attribute Protocol)

FIFO (First In First Out)

SCL (Serial Clock)

SDA (Serial Data)

IoT (Internet of Things)

Περιεχόμενα

Πρόλογος.....	3
Περίληψη.....	4
Abstract.....	5
Ευχαριστίες.....	6
Συντομογραφίες.....	7
Περιεχόμενα.....	8
Κατάλογος Σχημάτων.....	10
Κατάλογος Πινάκων.....	13
Κεφάλαιο 1^ο : Ιστορική Αναδρομή.....	14
1.1.Εισαγωγή.....	14
1.2 Ιστορική εξέλιξη.....	14
1.3 Επίλογος.....	16
Κεφάλαιο 2^ο : Αρχική έρευνα.....	17
2.1 Εισαγωγή.....	17
2.2 Παλμικό οξύμετρο.....	17
2.3 Οξύμετρία και Αιμοσφαιρίνη.....	23
2.4 Καρδιά και λειτουργία.....	26
2.5 Φασματοφωτομετρία και Νόμος Beer Lambert.....	29
2.6 Επίλογος.....	33
Κεφάλαιο 3^ο Ανάλυση εξαρτημάτων κατασκευής.....	34
3.1 Εισαγωγή.....	34
3.2 Bluetooth και BLE.....	34
3.3 LoRa(WAN).....	40
3.4 Αναλογικά και ψηφιακά Φίλτρα.....	41
3.5 TTGO LoRa.....	47
3.6 MAX 30100.....	50

3.7 Επίλογος.....	54
Κεφάλαιο 4° Κατασκευή.....	55
4.1 Εισαγωγή.....	55
4.2 Πρώτα Βήματα.....	55
4.3 Υλοποίηση της εφαρμογής με Blynk.....	63
4.4 Τελικός κώδικας.....	66
4.5 Διάγραμμα Ροής.....	81
Κεφάλαιο 5° Συμπεράσματα.....	93
5.1 Αποτελέσματα-συμπεράσματα.....	93
5.2 Μελλοντικές βελτιώσεις.....	93
Βιβλιογραφία.....	94
Εικόνες.....	96
Παράρτημα Α : Κώδικας.....	99

Κατάλογος Σχημάτων

Σχήμα 1.1: Το πρώτο παλμικό οξύμετρο.....	15
Σχήμα 2.1: Μεταδοτική και Ανακλαστική οξυμετρία.....	18
Σχήμα 2.2: Τα πλάτη παλμού (V_{pp}) σε V_{rms} ,.....	19
Σχήμα 2.3: Δείγμα καμπύλης βαθμονόμησης.....	20
Σχήμα 2.4: Σχέδιο μετάδοσης παλμικού οξυμέτρου.....	21
Σχήμα 2.5: Το ρεύμα/ένταση LED.....	22
Σχήμα 2.6: Διάγραμμα πίεσης οξυγόνου διαλυμένη στο αίμα.....	24
Σχήμα 2.7: Διάγραμμα μερικής πίεσης οξυγόνου.....	25
Σχήμα 2.8: Οι βαλβίδες της καρδιάς.....	27
Σχήμα 2.9: Συστηματική κυκλοφορία του οξυγόνου στο αίμα.....	28
Σχήμα 2.10: Μεσοκοιλιακά διαφράγματα.....	29
Σχήμα 2.11: Μέτρηση δείγματος με φασματοφωτομετρία.....	30
Σχήμα 2.12: Υπεριώδης ακτινοβολία.....	31
Σχήμα 2.13: Διαπερατότητα και απορρόφηση.....	32
Σχήμα 3.1: Point-to-point σύνδεση.....	34
Σχήμα 3.2: Τοπολογία σύνδεσης.....	35
Σχήμα 3.3: Αρχιτεκτονική του Bluetooth χαμηλής ενέργειας.....	37
Σχήμα 3.4: Ανίχνευση και επικοινωνία Bluetooth χαμηλής ενέργειας.....	38
Σχήμα 3.5: Δικτύωση ευρείας περιοχής LoRa.....	41
Σχήμα 3.6: Είδη των ψηφιακών φίλτρων.....	43
Σχήμα 3.7: Αποκλεισμός διαγραμμάτων φίλτρων FIR και IIR.....	44
Σχήμα 3.8: Ιδανική προσέγγιση φίλτρου χαμηλής διέλευσης.....	44
Σχήμα 3.9: Επίδραση μη γραμμικού χαρακτηριστικού φάσης.....	45

Σχήμα 3.10: Φίλτρο FIR.....	46
Σχήμα 3.11: Μοντέλο Esp32 (TTGO LoRa V1).....	47
Σχήμα 3.12: Μοντέλο LoRa.....	49
Σχήμα 3.13: Μοντέλο LoRa32 V1.....	49
Σχήμα 3.14: Μοντέλο LoRa32 V2.....	50
Σχήμα 3.15: MAX30100.....	50
Σχήμα 3.16: Διάγραμμα λειτουργίας του MAX30100.....	51
Σχήμα 3.17: Pinouts του MAX30100.....	54
Σχήμα 4.1: Pinouts του ESP32.....	56
Σχήμα 4.2: Σύνδεση ESP32 με το MAX30100.....	57
Σχήμα 4.3: Μετρήσεις στο Serial Monitoring.....	57
Σχήμα 4.4: Κυματομορφή των μετρήσεων.....	58
Σχήμα 4.5: Κυματομορφή μετρήσεων μετά από αφαίρεση DC σήματος.....	59
Σχήμα 4.6: Τελική κυματομορφή μετρήσεων.....	59
Σχήμα 4.7: Ρυθμίσεις Butterworth φίλτρου.....	60
Σχήμα 4.8: Σύνδεση MAX30100 με ESP32 και Display.....	62
Σχήμα 4.9: Μετρήσεις στο Serial Monitoring.....	62
Σχήμα 4.10: Σύγκριση μετρήσεων με Smartwatch	63
Σχήμα 4.11: Ανάπτυξη εφαρμογής μέσω Blynk	63
Σχήμα 4.12: Παλμόμετρο (αριστερά) και Display (δεξιά) στο Blynk.....	64
Σχήμα 4.13: Ονομασία εύρεσης του ESP32.....	65
Σχήμα 4.14: Μέτρηση κορεσμού και παλμών στο Display του ESP32.....	65
Σχήμα 4.15: Οι μετρήσεις στην εφαρμογή.....	66
Σχήμα 4.16: Συνδεσμολογία των Buttons.....	69
Σχήμα 4.17: Αρχικό μενού στην οθόνη του ESP32.....	72
Σχήμα 4.18: Υπομενού παλμών.....	72

Σχήμα 4.19: Υπομενού κορεσμού οξυγόνου.....	73
Σχήμα 4.20: Υπομενού παλμών και κορεσμού οξυγόνου.....	73
Σχήμα 4.21: Υπομενού στοιχείων των δημιουργών.....	74
Σχήμα 4.22: Λειτουργίες -mode για το DeepSleep.....	74
Σχήμα 4.23: Κατανάλωση ενέργειας ανάλογα την λειτουργία.....	75
Σχήμα 4.24: Σύνδεση Button στο ESP32 με Pull Down αντίσταση.....	75
Σχήμα 4.25: Επιλογή εικόνας bmp.....	76
Σχήμα 4.26: Ορισμός διαστάσεων εικόνας bmp.....	77
Σχήμα 4.27: Κώδικας εικόνας bmp.....	77
Σχήμα 4.28: Εικόνα ενεργοποίησης DeepSleep.....	78
Σχήμα 4.29: Σχεδίαση πλακέτας και εξαρτημάτων.....	79
Σχήμα 4.30: Σχεδίαση του κάτω μέρος κουτιού	79
Σχήμα 4.31: Σχεδίαση του πάνω μέρος κουτιού	79
Σχήμα 4.32: Τελική μορφή.....	80

Κατάλογος πινάκων

Πίνακας 1: Πιθανές τιμές συχνότητας δειγματοληψίας MAX30100.....	52
Πίνακας 2: Διαμόρφωσης LED.....	52
Πίνακας 3: Pinouts MAX30100.....	53
Πίνακας 4: Κοστολόγιο υλικών.....	55

Κεφάλαιο 1^ο : Ιστορική Αναδρομή

1.1 Εισαγωγή

Φανταστείτε ότι είστε γιατρός το 1975. Ένας 65χρονος άνδρας μπαίνει στο Νοσοκομείο εμφανίζοντας ελαφρώς δύσπνοια. Τα ζωτικά σημάδια αποκαλύπτουν θερμοκρασία 37 βαθμών Κελσίου, 85 παλμούς/λεπτό, αρτηριακή πίεση 155/85 mm Hg και αναπνευστικό ρυθμό 20 αναπνοών/λεπτό. Πόσο άρρωστος είναι αυτός ο άντρας;

Αμέσως μπορεί να σκεφτείτε ότι λείπει κάτι πολύ σημαντικό, ζωτικής σημασίας ίσως. Ο κορεσμός οξυγόνου δεν αναφέρεται, το 1975, το παλμικό οξύμετρο δεν ήταν ακόμη διαθέσιμο και ο έλεγχος του κορεσμού του αρτηριακού οξυγόνου στο αίμα απαιτούσε μια αρτηριακή παρακέντηση, η οποία ήταν αρκετά επώδυνη για τον ασθενή προκαλώντας παράλληλα άγχος για τον ιατρό που θα επιχειρούσε πρώτη φορά να κάνει αρτηριακή διάπλωση.

Το ανθρώπινο σώμα απαιτεί και ρυθμίζει την ισορροπία του οξυγόνου στο αίμα μας.

Η οξυγόνωση συμβαίνει όταν ένα μόριο οξυγόνου μπαίνει στους ιστούς του σώματός μας. Το φυσιολογικό επίπεδο οξυγόνου στο σώμα μας είναι 95 έως 100%. Εάν το επίπεδο του οξυγόνου μετράτε κάτω από το 90%, δημιουργεί μια κατάσταση που ονομάζεται υποξαιμία, δηλαδή χαμηλή συγκέντρωση οξυγόνου στο αίμα. Τα επίπεδα οξυγόνου κάτω του 80% θα επηρεάσουν τις λειτουργίες των πρωτογενών οργάνων.

Η παλμική οξυμετρία είναι μια μη επεμβατική μέθοδος μέτρησης και παρακολούθησης του κορεσμού του οξυγόνου.

1.2 Ιστορική εξέλιξη

Θα εκπλήξει πιθανώς τους νεότερους γιατρούς το γεγονός ότι το σύγχρονο παλμικό οξύμετρο δεν εφευρέθηκε μέχρι τις αρχές της δεκαετίας του 1970 και δεν ήταν διαθέσιμο στο εμπόριο μέχρι τη δεκαετία του 1980. Ακόμη και στα τέλη της δεκαετίας του 1990 υπήρχε ακόμη συζήτηση σχετικά με τη χρησιμότητα της ρουτίνας παλμικής οξυμετρίας για ασθενή των εκτάτων περιστατικών. Το 1988, ο Δρ. Τόμας Νεφ πρότεινε να θεωρήσουμε τον κορεσμό οξυγόνου από την παλμική οξυμετρία ως «πέμπτο ζωτικό σημάδι». Η ιδέα έπιασε και τώρα η μικρή πολύπλοκη συσκευή είναι διαθέσιμη στα περισσότερα γραφεία πρωτοβάθμιας περίθαλψης σε όλο το κόσμο[1]. Αλλά ακόμη και με την ευρεία διαθεσιμότητά του, οι περισσότεροι γιατροί σήμερα πιθανώς δεν μπορούν να εξηγήσουν πώς λειτουργεί αυτό το ισχυρό εργαλείο. Το 1939, ο Καρλ Μάττες, Γερμανός γιατρός, κατασκεύασε μια συσκευή που απέδειξε ότι ο κορεσμός της οξυαιμοσφαιρίνης μπορούσε να μετρηθεί στο αυτί. Χρησιμοποίησε 2 διαφορετικά μήκη κύματος φωτός, κόκκινο και υπέρυθρο, το υπέρυθρο.

απορροφάται περισσότερο από οξυγονωμένη αιμοσφαιρίνη και το κόκκινο περισσότερο από αποοξυγονωμένη αιμοσφαιρίνη. Μια πρώιμη εφαρμογή αυτής της συσκευής έγινε όταν ο Αμερικανός φυσιολόγος Γκλεν Μίλικα έκανε ένα οξύμετρο του αυτιού για να ειδοποιεί τους πιλότους του Β 'Παγκοσμίου Πολέμου όταν ο κορεσμός οξυγόνου τους ήταν χαμηλός. Για την συσκευή αυτή, ο Μίλικαν επινόησε τον όρο «Οξύμετρο».

Η ιδέα μπορεί να χρησιμοποιηθεί άψογα για τη μέτρηση του οξυγόνου σε έναν δοκιμαστικό σωλήνα γεμάτο αίμα.

Είναι μόνο μια απλή σύγκριση της ποσότητας του κόκκινου που απορροφάται με την ποσότητα του απορροφώμενου υπέρυθρου.

Το οξύμετρο του αυτιού, ωστόσο, στην πραγματικότητα δεν είχε πολύ

κακή απόδοση, επειδή το φως που απορροφάται από το αυτί επηρεάζεται πολύ λίγο από το αρτηριακό αίμα. Η συντριπτική πλειονότητα απορροφάται από άλλους ιστούς, συμπεριλαμβανομένου του δέρματος, του λίπους και του συνδετικού ιστού.



Σχήμα 1.1 «Το πρώτο παλμικό οξύμετρο»[1]

Στο Λονδίνο, ο Τζ.Ρ.Σκουάηρ βοήθησε να επιλύσει αυτό το πρόβλημα χρησιμοποιώντας πνευματική πίεση για να συμπιέσει το αίμα από τον ιστό του χεριού, επιτρέποντας τον υπολογισμό της αρχικής απορρόφησης χωρίς αίμα. Αυτή η ιδέα έγινε πρακτικά πιο εφικτή όταν ο Έρλ Γούντ, ένας Αμερικανός φυσιολόγος/καρδιολόγος ο οποίος είναι γνωστός για την εφεύρεσή της G-στολής (επέτρεψε στους πιλότους να επιβιώσουν σε μεγάλα υψόμετρα), άρχισε να χρησιμοποιεί το συνδυασμό του οξύμετρου του αυτιού και της πνευμονικής πίεσης και συνέχισε να επεκτείνει την επιστήμη και τη μηχανική της παλμικής οξυμετρίας στα μέσα του 20ού αιώνα[1].

Η τελευταία σημαντική ανακάλυψη σημειώθηκε από τον Τάκουο Αογιάγκι, έναν Ιάπωνα ηλεκτρολόγο/μηχανικό, ο οποίος προσπαθούσε να χρησιμοποιήσει μια έκδοση του οξύμετρου του αυτιού του Γούντ για να μετρήσει την αραίωση της βαφής για σκοπούς μέτρησης της καρδιακής απόδοσης αλλά δυσκολεύεται λόγω του συνεχούς τεχνουργήματος που δημιουργείται από τους παλμούς[1]. Αυτή η δυσκολία, ωστόσο, τον ώθησε να συνειδητοποιήσει ότι οι αλλαγές που δημιουργήθηκαν με τον παλμό αντιπροσώπευαν μόνο την αλλαγή στο αρτηριακό αίμα. Επομένως, ο χρόνος που δεν υπήρχε παλμός θα μπορούσε να χρησιμοποιηθεί ως βασική απορρόφηση, εξαλείφοντας την ανάγκη για πνευματική πίεση. Δυστυχώς, η επιχείρηση στην οποία δούλεψε ο Τάκουο Αογιάγκι δεν αναγνώρισε τις δυνατότητες της εφεύρεσής του και τελικά η παραγωγή των πρώτων οξύμετρων έγινε από άλλες εταιρίες. Όπως τα περισσότερα ιατρικά βοηθήματα, το παλμικό οξύμετρο δεν είναι τέλειο. Έχει βρεθεί ότι παρέχει ανακριβή αποτελέσματα σε ορισμένες ρυθμίσεις (οι οποίες είναι εύκολα εμφανείς όταν καταλαβαίνετε πώς λειτουργεί η συσκευή). Ένα σημαντικό ζήτημα αφορά την παρουσία καρβοξυαιμοσφαιρίνης και μεθεμοσφαιρίνης, όπως στη δηλητηρίαση από μονοξείδιο του άνθρακα και στη μεθαιμοσφαιριναιμία. Αυτά τα μόρια έχουν παρόμοια πρότυπα απορρόφησης με την οξυαιμοσφαιρίνη και ως εκ τούτου προκαλούν ψευδώς αυξημένες μετρήσεις.

Κεφάλαιο 1^ο

Άλλα ζητήματα που σχετίζονται με το χρώμα μπορεί να προκαλέσουν προβλήματα σε ορισμένες βαφές (όπως το μπλε του μεθυλενίου), τα ακρυλικά νύχια και το χρώμα του δέρματος.

Ο τελευταίος «περιορισμός» του παλμικού οξυμέτρου είναι ότι είναι μόνο ένα εργαλείο, το οποίο προοριζόταν για να χρησιμοποιηθεί από έναν επαγγελματία ιατρό. Μια σημαντική πτυχή αυτού του κανόνα είναι ότι η οξυγόνωση δεν είναι εξαερισμός. Πολλοί ασθενείς έχουν φυσιολογικό κορεσμό οξυγόνου, αλλά εξακολουθούν να έχουν πρόβλημα στην αναπνοή και αυτό οφείλεται στο ότι δεν μπορούν να καθαρίσουν το διοξείδιο του άνθρακα.

1.3 Επίλογος

Κάναμε μια μικρή εισαγωγή για την σημαντικότητα του παλμικού οξυμέτρου στην ιατρική και τα ερεθίσματα που ώθησαν προς την εφεύρεση του. Τα στάδια που πέρασε για να φτάσει στο σημείο που βρίσκεται και τα προβλήματα που αντιμετώπισαν επιστήμονες και κατασκευαστές.

Κεφάλαιο 2ο : Αρχική έρευνα

2.1 Εισαγωγή

Στο κεφάλαιο της αρχικής έρευνας θα δούμε τι είναι το παλμικό οξύμετρο και πως λειτουργεί. Τους νόμους τους οποίους ακολουθεί και με ποιο τρόπο μπορεί το φως να διαπεράσει το αίμα και φτάσει στο σημείο να μας δίνει πληροφορίες για την υγεία του ανθρώπου την λειτουργία της καρδιάς και την κυκλοφορία του οξυγόνου στο αίμα.

2.2 Παλμικό οξύμετρο

Ένα παλμικό οξύμετρο αποτελείται από ένα ηλεκτρονικό μόνιτορ και έναν αισθητήρα που μπορεί να συνδεθεί με το δάκτυλο χεριού, ποδιού, τη μύτη ή το αυτί του ασθενή. Η μονάδα του μόνιτορ εμφανίζει στην οθόνη ένα ψηφιακό ποσοστό της υπολογιζόμενης αιμοσφαιρίνης (Hg) του ασθενούς, που είναι κορεσμένο με οξυγόνο (SpO₂). Εμφανίζεται ένας δείκτης οπτικής κυματομορφής και ένα ακουστικό σήμα εκπέμπεται με κάθε παλμό, όπου ο τόνος μειώνεται με την αντίστοιχη μείωση του κορεσμού. Επίσης εμφανίζεται ένας υπολογισμένος καρδιακός ρυθμός. Οι συναγερμοί είναι διαθέσιμοι για την ειδοποίηση του χρήστη είτε σε υψηλό ή χαμηλό επίπεδο κορεσμού, είτε σε γρήγορο ή αργό ρυθμό παλμών.

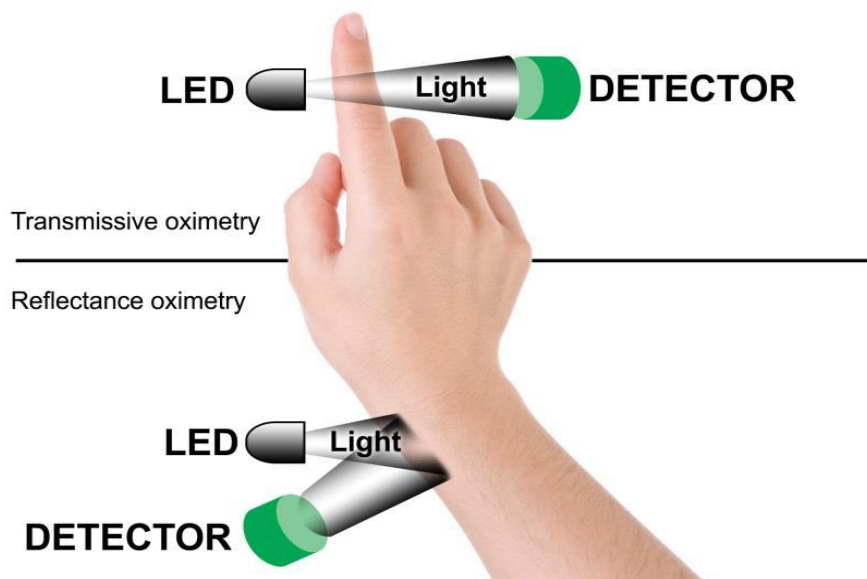
Τι κάνει

Η συσκευή μετράει δύο τύπους αιμοσφαιρίνης, οξυγονωμένης και μη οξυγονωμένης. Δεδομένου ότι μετρούνται δύο διαφορετικές ουσίες, είναι απαραίτητες δύο συχνότητες φωτός. Αυτό ονομάζεται φασματοφωτομετρία[10]. Η κόκκινη συχνότητα μετρά την ακόρεστη αιμοσφαιρίνη και η υπέρυθη την οξυγονωμένη αιμοσφαιρίνη. Εάν το οξύμετρο μετρά τη μεγαλύτερη απορρόφηση στην κόκκινη ζώνη, θα δείξει χαμηλό κορεσμό, ενώ αν βρίσκεται στη ζώνη υπέρυθρων υποδηλώνει υψηλό κορεσμό.

Πως μετράει ένα παλμικό οξύμετρο τον κορεσμό οξυγόνου στο αίμα

Ένα από τα πραγματικά ενδιαφέροντα πράγματα για την αιμοσφαιρίνη είναι πως αντανακλά και απορροφά το φως (Σχ. 2.1). Για παράδειγμα, το Hb απορροφά περισσότερο (και αντανακλά λιγότερο) ορατό κόκκινο φως[10]. Το HbO₂ απορροφά περισσότερο (και αντανακλά λιγότερο) υπέρυθρο φως. Δεδομένου ότι ο κορεσμός του οξυγόνου στο αίμα μπορεί να προσδιοριστεί συγκρίνοντας τις τιμές των Hb και HbO₂, μία μέθοδος για να γίνει αυτό είναι να λάμπει ένα κόκκινο LED και ένα υπέρυθρο LED μέσω ενός μέρους του σώματος (όπως το δάχτυλο) και στη συνέχεια, να συγκριθούν οι σχετικές τους εντάσεις. Υπάρχουν δύο κοινές μέθοδοι για να γίνει αυτό:

1. Η μέτρηση του φωτός που μεταδίδεται μέσω ιστού, ονομάζεται μεταδοτική οξυμετρία.
2. Η μέτρηση του φωτός που αντανακλάται από τον ιστό, ονομάζεται οξυμετρία ανάκλασης.



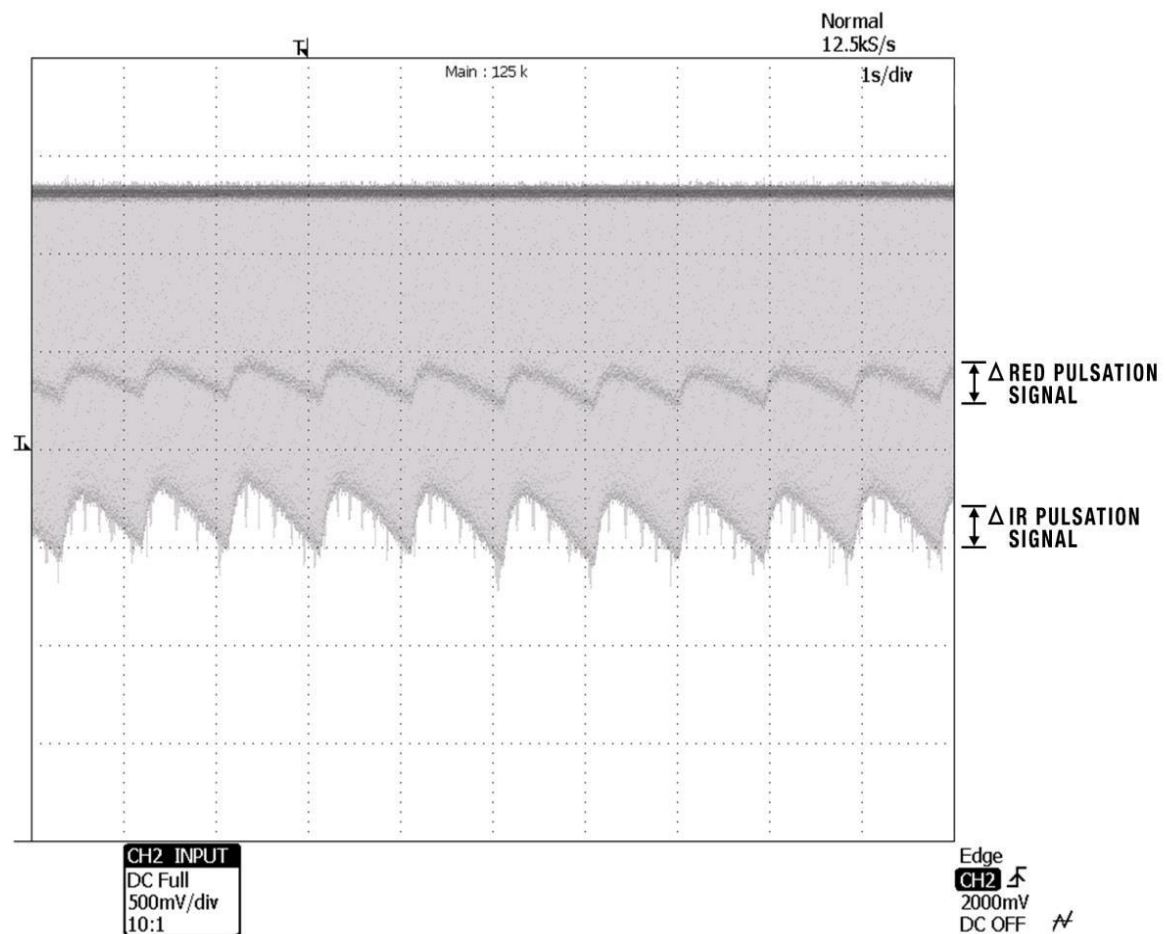
Σχήμα 2.1 «Μεταδοτική και Ανακλαστική οξυμετρία»[2]

Πως ένα παλμικό οξύμετρο μετράει τον ρυθμό παλμού

Όταν η καρδιά χτυπά, αντλεί αίμα στο σώμα. Κατά τη διάρκεια κάθε καρδιακού παλμού, το αίμα συμπιέζεται σε τριχοειδή αγγεία, του οποίου ο όγκος αυξάνεται πολύ ελαφρώς. Μεταξύ των καρδιακών παλμών, ο όγκος μειώνεται. Αυτή η αλλαγή στον όγκο επηρεάζει την ποσότητα του φωτός, όπως η ποσότητα του κόκκινου ή του υπέρυθρου φωτός, που θα μεταδοθεί μέσω του ιστού[3]. Αν και αυτή η διακύμανση είναι πολύ μικρή, μπορεί να μετρηθεί με παλμικό οξύμετρο χρησιμοποιώντας τον ίδιο τύπο ρύθμισης που χρησιμοποιείται για τη μέτρηση του κορεσμού οξυγόνου στο αίμα.

Λεπτομερής θεωρία λειτουργίας

Τα τυπικά παλμικά οξύμετρα παρακολουθούν τον κορεσμό οξυγόνου (SpO₂) του αίματος ενός ανθρώπου, με βάση το κόκκινο φως (χρησιμοποιώντας μήκος κύματος 600-750 nm) και το υπέρυθρο φως (χρησιμοποιώντας μήκος κύματος 850-1000 nm), χαρακτηριστικά απορρόφησης της οξυγονωμένης αιμοσφαιρίνης (HbO₂) και αποοξυγονωμένης αιμοσφαιρίνης (Hb). Αυτός ο τύπος παλμικού οξύμετρου αναβοσβήνει το κόκκινο και το υπέρυθρο φως εναλλάξ μέσα σε ένα μέρος του σώματος, όπως ένα δάχτυλο, σε έναν αισθητήρα φωτοδιόδου. Η φωτοδίοδος χρησιμοποιείται συνήθως για τη λήψη του μη απορροφούμενου φωτός από κάθε LED. Αυτό το σήμα στη συνέχεια αντιστρέφεται χρησιμοποιώντας έναν αντιστρεπτό λειτουργικό ενισχυτή[10]. Το σήμα που προκύπτει αντιπροσωπεύει το φως που έχει απορροφηθεί από το δάχτυλο, όπως φαίνεται στο παρακάτω σχήμα.



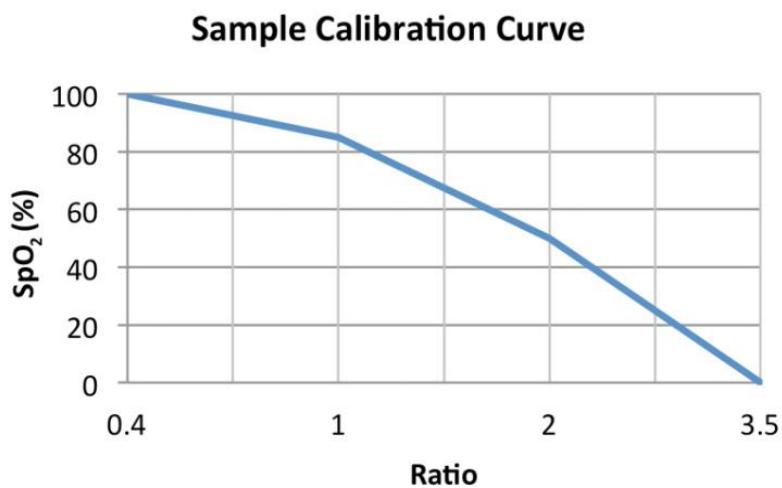
Σχήμα 2.2 «Τα πλάτη παλμού (V_{pp}) των κόκκινων και υπέρυθρων σημάτων μετρούνται και μετατρέπονται σε V_{rms} , προκειμένου να παραχθεί μια τιμή αναλογίας»[2]

Το SpO₂ μπορεί να προσδιοριστεί χρησιμοποιώντας την τιμή αναλογίας και έναν πίνακα αναζήτησης που αποτελείται από εμπειρικούς τύπους. Ο ρυθμός παλμού μπορεί να υπολογιστεί με βάση τον αριθμό δείγματος και τον ρυθμό δειγματοληψίας του αναλογικού σε ψηφιακό μετατροπέα (ADC) του παλμικού οξύμετρου[3].

Ένας πίνακας αναζήτησης είναι ένα σημαντικό μέρος ενός παλμικού οξύμετρου. Οι πίνακες

Κεφάλαιο 2°

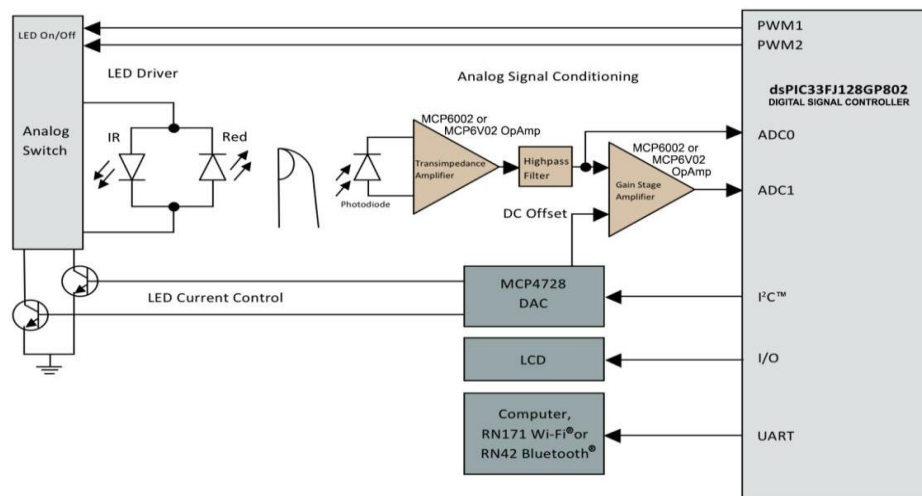
αναζήτησης είναι συγκεκριμένοι για ένα συγκεκριμένο σχέδιο οξύμετρου και συνήθως βασίζονται σε καμπύλες βαθμονόμησης (Σχ. 2.3) που προέρχονται από μεγάλο αριθμό μετρήσεων από θέματα διαφόρων επιπέδων SpO₂.



Σχήμα 2.3 «Δείγμα καμπύλης βαθμονόμησης» [2]

Περιγραφή σχεδίου κυκλώματος

Το ακόλουθο παράδειγμα περιγράφει τις διάφορες ενότητες ενός σχεδίου μετάδοσης παλμικού οξύμετρου. Αυτός ο σχεδιασμός, καταδεικνύει τη μέτρηση των καρδιακών παλμών και των επιπέδων κορεσμού του οξυγόνου στο αίμα.



Σχήμα 2.4 «Σχέδιο μετάδοσης παλμικού οξυμέτρου» [2]

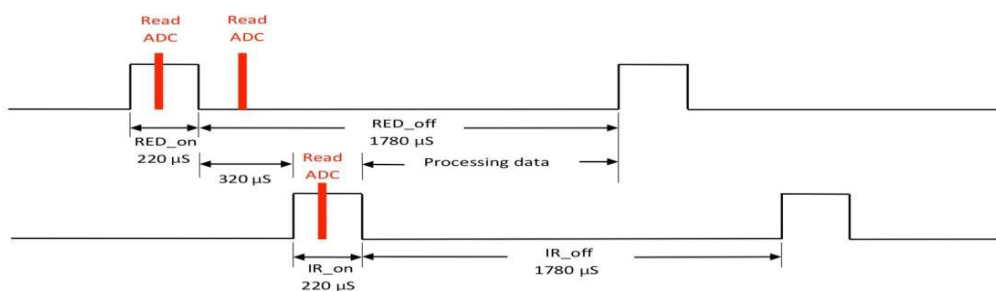
Ανιχνευτής

Ο ανιχνευτής SpO2 που χρησιμοποιείται σε αυτό το παράδειγμα είναι ένα κλιπ δαχτύλου, που ενσωματώνει ένα κόκκινο LED και ένα IR LED, καθώς και μια φωτοδίοδο. Τα LED ελέγχονται από το κύκλωμα οδήγησης LED.

Το κόκκινο φως και το φως IR που περνούν από το δάχτυλο ανιχνεύονται από το κύκλωμα ρύθμισης σήματος και στη συνέχεια τροφοδοτούνται στη μονάδα ADC 12-bit που είναι ενσωματωμένη στον ψηφιακό ελεγκτή σήματος (DSC), όπου υπολογίζεται το ποσοστό του SpO2[2].

Κύκλωμα οδηγού LED

Ένας διπλός μονοπολικός αναλογικός διακόπτης διπλής ρίψης, που οδηγείται από δύο σήματα PWM από το DSC (Σχ. 2.4), ενεργοποιεί και απενεργοποιεί εναλλάξ τις κόκκινες και υπέρυθρες λυχνίες LED[3]. Για να αποκτήσουμε το σωστό αριθμό δειγμάτων ADC και να έχουμε αρκετό χρόνο για να επεξεργαστούμε τα δεδομένα προτού ανάψει τα επόμενα LED, τα LED ανάβουν και σβήνουν σύμφωνα με το παρακάτω διάγραμμα χρονισμού.



Σχήμα 2.5 Το ρεύμα/ένταση LED ελέγχεται από έναν ψηφιακό μετατροπέα 12-bit (DAC), ο οποίος οδηγείται από το DSC. [2]

Κύκλωμα αναλογικού σήματος

Υπάρχουν δύο στάδια στο κύκλωμα σήματος. Το πρώτο στάδιο είναι ο ενισχυτής transimpedance, και το δεύτερο στάδιο είναι ο ενισχυτής κέρδους. Ένα φίλτρο υψηλής διέλευσης τοποθετείται μεταξύ των δύο σταδίων.

Ο ενισχυτής ρεύματος transimpedance μετατρέπει τους λίγους μικροενισχυτές ρεύματος, οι οποίοι δημιουργούνται από τη φωτοδίοδο, σε μερικά millivolts. Το σήμα που λαμβάνεται από αυτόν τον ενισχυτή πρώτου σταδίου περνά έπειτα από ένα φίλτρο υψηλής διέλευσης, το οποίο έχει σχεδιαστεί για να μειώνει τις παρεμβολές στο φως του περιβάλλοντος χώρου.

Στη συνέχεια, η έξοδος του φίλτρου υψηλής διέλευσης αποστέλλεται σε ενισχυτή δεύτερου σταδίου με κέρδος 22 και τάση μετατόπισης DC 220mV. Οι τιμές για το κέρδος του ενισχυτή και την μετατόπιση DC έχουν ρυθμιστεί ώστε να τοποθετούν σωστά το επίπεδο σήματος εξόδου του ενισχυτή κέρδους στο εύρος ADC του MCU.

Φυσιολογικά επίπεδα

Η φυσιολογική τιμή κορεσμού οξυγόνου (SpO₂) για τους ενήλικες χωρίς πνευμονική ασθένεια είναι πάνω από 95% και για τα παιδιά ένα επίπεδο μεγαλύτερο, πάνω από 96%. Αυτό μπορεί να είναι παραπλανητικό καθώς πρέπει να ληφθούν υπόψη και άλλοι παράγοντες. Αυτό αναφέρεται παρακάτω στην καμπύλη διάστασης οξυαιμοσφαιρίνης[4]. Όταν υπάρχει αμφιβολία ένα αρτηριακό αέριο αίματος (ABG), όπου θα πρέπει να προσαρμοστεί για να αξιολογηθεί η συσχέτιση μεταξύ του κορεσμού αρτηριακού οξυγόνου και του παλμικού οξυμέτρου.

Τοποθέτηση του αισθητήρα

Η τοποθέτηση του αισθητήρα ανίχνευσης είναι σημαντική για τη λήψη μίας ακριβούς

ανάγνωσης, όπως η χρήση του κατάλληλου τύπου και μεγέθους του ανιχνευτή. Όταν χρησιμοποιούμε έναν ανιχνευτή δακτύλου, τον εφαρμόζουμε στο χέρι που δεν χρησιμοποιείται για πιέσεις αίματος, αρτηριακές γραμμές ή με επίδεσμο πίεσης. Ο αισθητήρας θα πρέπει να εφαρμόζει με το δέρμα και να είναι ασφαλισμένος χωρίς να επηρεάζεται η κυκλοφορία στο δάκτυλο. Κατά τη διάρκεια της μακροχρόνιας συνεχούς χρήσης, η περιοχή του ανιχνευτή πρέπει να ελέγχεται συχνά για να αποφευχθεί ο τραυματισμός ιστών. Σε περίπτωση αμφιβολίας για την ακρίβεια της ανάγνωσης, επανατοποθετούμε τον αισθητήρα ή δοκιμάζουμε να χρησιμοποιήσουμε άλλο δάκτυλο. Βεβαιωνόμαστε ότι οι υψηλοί και οι χαμηλοί συναγερμοί έχουν οριστεί για να ειδοποιούν τον χρήστη αμέσως σε οποιοδήποτε αλλαγές. Οι αισθητήρες που έχουν σχεδιαστεί για πολλαπλούς ασθενείς πρέπει να καθορίζονται σύμφωνα με τις προδιαγραφές των κατασκευαστών για να μειωθεί ο κίνδυνος μόλυνσης.

2.3 Οξυμετρία και αιμοσφαιρίνη

Η αναπνοή είναι μια ζωτική διαδικασία στο ανθρώπινο σώμα. Οι ιστοί του σώματος απαιτούν οξυγόνο γιατί κερδίζουν όλη τους την ενέργεια μέσω της αναπνοής και από την ενέργεια που απελευθερώνεται μέσω αυτού.

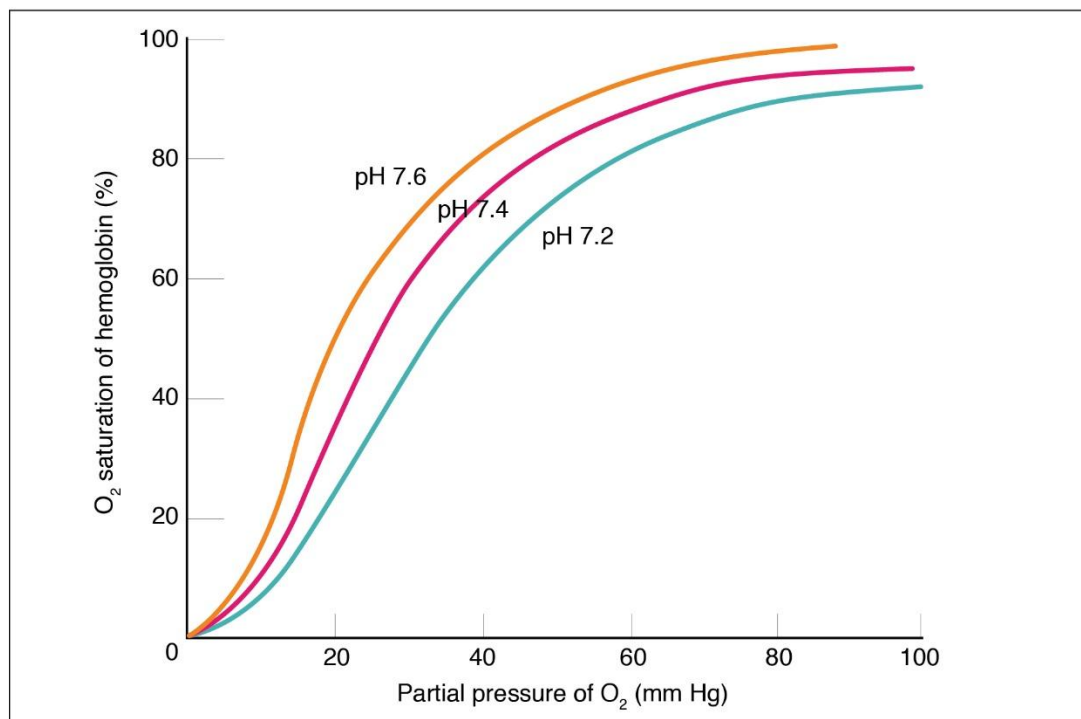
Η μέθοδος χρησιμοποιείται για την αποθήκευση της συνθετικής τριφωσφορικής αδενοσίνης. Αυτή η αποθηκευμένη ενέργεια είναι υπεύθυνη για τη διανομή οξυγόνου σε όλα τα κύρια μέρη του σώματος μέσω του αίματός μας για την σωστή ροή κυκλοφορίας. Το αίμα διέρχεται από πνευμονικές κυψελίδες όπου απελευθερώνεται διοξείδιο του άνθρακα ώστε το αίμα να παραμένει οξυγονωμένο. Αυτό ονομάζεται διαδικασία οξυγόνωσης. Το αποξυγονωμένο αίμα εισέρχεται στην καρδιά ώστε να αντληθούν οι πνεύμονες.

Παλμική οξυμετρία

Η παλμική οξυμετρία είναι μία σύντομη, μη επεμβατική και ανώδυνη εξέταση με την οποία εκτιμώνται τα επίπεδα οξυγόνου ή αλλιώς ο κορεσμός οξυγόνου στο αίμα που συμβολίζεται με SpO₂ και ορίζεται ως ο λόγος της οξυαιμοσφαιρίνης προς τη συνολική ποσότητα αιμοσφαιρίνης που είναι στο αίμα[5]. Ο γενικός δείκτης της μεταφοράς του οξυγόνου στους περιφερικούς ιστούς είναι το δάκτυλο του χεριού, ο λοβός του αυτιού και η μύτη. Το οξυγόνο είναι ένα καθαρό, άοσμο αέριο που αντιπροσωπεύει το 21% των αερίων στον αέρα γύρω μας. Είναι απαραίτητο για τη διαδικασία που χρησιμοποιεί το σώμα μας για να παράγει την ενέργεια που χρειάζεται για τον μεταβολισμό. Το ατμοσφαιρικό οξυγόνο εισέρχεται στις κυψελίδες του πνεύμονα σε κάθε εισπνοή. Από τις κυψελίδες περνά στο αίμα και συνδέεται κατά ένα μεγάλο ποσοστό με την αιμοσφαιρίνη (μία πρωτεΐνη που βρίσκεται μέσα στα ερυθρά αιμοσφαίρια) για να γίνει η κυκλοφορία. Έτσι το οξυγονωμένο αίμα φτάνει στους ιστούς. Το υπερβολικό ή το πολύ λίγο οξυγόνο (υποξία) μπορεί να προκαλέσει ασθένειες ή και θάνατο. Αυτός είναι ο λόγος για τον οποίο κατέστη αναγκαίο οι επαγγελματίες του τομέα υγείας να είναι σε θέση να υπολογίζουν την ποσότητα οξυγόνου στη ροή του αίματος[9]. Το μόριο αιμοσφαιρίνης αποτελείται από 10.000 άτομα, τέσσερα των οποίων είναι άτομα σιδήρου που προσελκύουν και συγκρατούν το οξυγόνο. Κάθε ερυθρό κύτταρο περιέχει 250 εκατομμύρια μόρια αιμοσφαιρίνης. Υπάρχουν περίπου 5.000ml αίματος στο μέσο άτομο και κάθε ml περιέχει πέντε δισεκατομμύρια ερυθρά αιμοσφαίρια[4]. Εάν βρισκόσασταν σε

Κεφάλαιο 2°

ατμόσφαιρα καθαρού οξυγόνου, η αρτηριακή πίεση του οξυγόνου θα ήταν σχεδόν ίση με 760 mm Hg στο αίμα. Επειδή η ατμόσφαιρά μας είναι μόνο 20,9% οξυγόνο, πρέπει να βρούμε το 20,9% των 760 mm για να καθορίσουμε πόσο οξυγόνο (ή μερική πίεση) εξαναγκάζεται στο αίμα μας. Υπάρχει μία σχέση μεταξύ της ποσότητας οξυγόνου που διαλύεται στο αίμα και της ποσότητας που συνδέεται με την αιμοσφαιρίνη.



(b) Effect of pH

Σχήμα 2.6 «Διάγραμμα πίεσης οξυγόνου διαλυμένη στο αίμα»[3]

Στο σχήμα 2.6 βλέπουμε ότι η μερική πίεση του οξυγόνου είναι 80 Torr, η αιμοσφαιρίνη είναι 92% κορεσμένη με οξυγόνο. Καθώς αυξάνεται η πίεση οξυγόνου, η αιμοσφαιρίνη γίνεται πιο κορεσμένη. Οποιαδήποτε πίεση πάνω από 105 καθιστά πλήρως κορεσμένη την αιμοσφαιρίνη. Η χρήση παλμικού οξυμέτρου μας επιτρέπει να αξιολογήσουμε την πίεση οξυγόνου εξετάζοντας τον κορεσμό της αιμοσφαιρίνης.

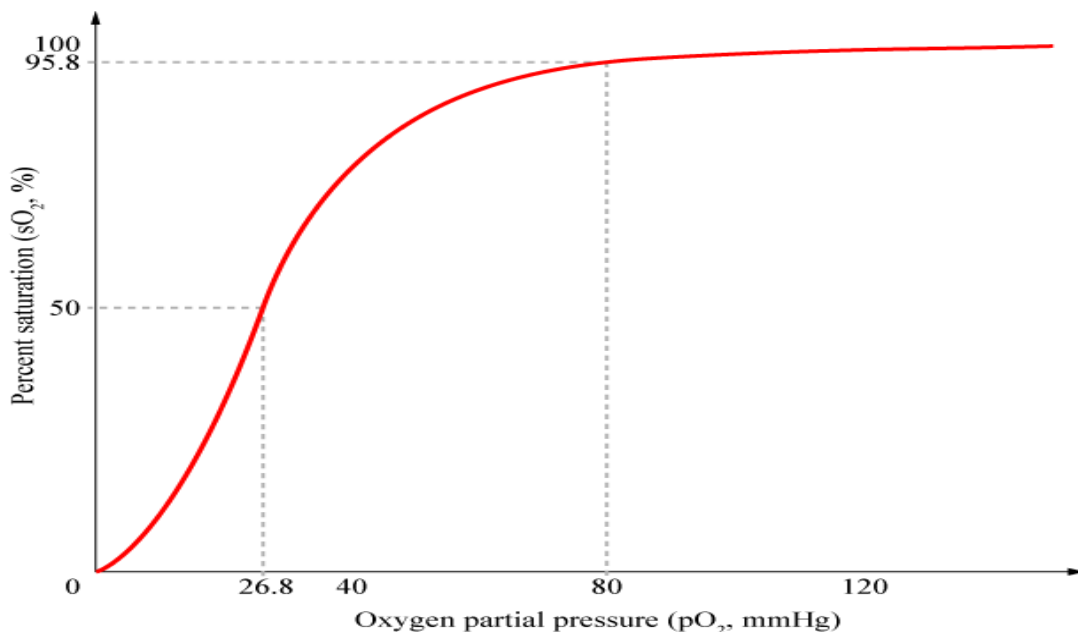
Παρόλα αυτά δεν είναι τόσο εύκολο καθώς υπάρχουν διάφοροι παράγοντες που επηρεάζουν την προσέλκυση αιμοσφαιρίνης στο οξυγόνο. Η καμπύλη μπορεί να μετακινηθεί προς τα δεξιά ή προς τα αριστερά.

Αιμοσφαιρίνη

Κάθε ενεργός ιστός στο σώμα έχει απόλυτη απαίτηση για οξυγόνο. Για τη συντριπτική πλειονότητα αυτών των ιστών, το οξυγόνο μεταφέρεται από το αίμα στους ιστούς, αν και υπάρχουν μερικές αξιοσημείωτες εξαιρέσεις (για παράδειγμα, ο κερατοειδής παίρνει το οξυγόνο του απευθείας από την ατμόσφαιρα). Μόλις το οξυγόνο εισέλθει στο αίμα από τους πνεύμονες, απορροφάται από την αιμοσφαιρίνη (Hb) στα ερυθρά αιμοσφαίρια. Η αιμοσφαιρίνη είναι μια πρωτεΐνη που αποτελείται από τέσσερις ομάδες αίματος, οι οποίες περιέχουν ιόντα σιδήρου[5]. Αυτά τα ιόντα σιδήρου (Fe^{2+}) που σχετίζονται με μόρια αιμοσφαιρίνης αντιδρούν χημικά με οξυγόνο για να σχηματίσουν την οξυαιμοσφαιρίνη. Κάθε μόριο αιμοσφαιρίνης μπορεί να περιέχει τέσσερα μόρια οξυγόνου. Είναι η αιμοσφαιρίνη που μεταφέρει το οξυγόνο καθώς μεταφέρεται γύρω από το σώμα στο αίμα. Μόνο μια μικρή ποσότητα οξυγόνου (1,5% σε αρτηριακό αίμα) διαλύεται απλά στο πλάσμα. Η αιμοσφαιρίνη είναι το κύριο συστατικό της μεταφοράς οξυγόνου σε όλο το σώμα. Τα άτομα που έχουν διαταραχές που επηρεάζουν την ποιότητα ή την ποσότητα της αιμοσφαιρίνης υποφέρουν σοβαρά προβλήματα με τη μεταφορά οξυγόνου στους ιστούς τους.

Σύνδεση οξυγόνου με αιμοσφαιρίνη

Η αιμοσφαιρίνη είναι τετραμερές - αποτελείται από τέσσερις υπομονάδες πρωτεΐνης. Ως αποτέλεσμα, αλλάζει σχήμα με βάση το πόσα μόρια οξυγόνου συνδέονται με αυτό και η αλλαγή του σχήματος προκαλεί επίσης μια αλλαγή στη συγγένεια με το οξυγόνο[5]. Όσο περισσότερο δεσμεύεται το οξυγόνο, τόσο μεγαλύτερη είναι η συγγένεια του μορίου με το οξυγόνο, αυτό είναι γνωστό ως δεσμευτική συνεργασία. Όταν δεν υπάρχει δεσμευμένο οξυγόνο, η αιμοσφαιρίνη λέγεται ότι βρίσκεται σε κατάσταση έντασης (κατάσταση T), χαμηλή συγκέντρωση οξυγόνου[6]. Στο σημείο όπου το οξυγόνο συνδέεται για πρώτη φορά, η αιμοσφαιρίνη μεταβάλλει το σχήμα της στη χαλαρή κατάσταση (κατάσταση R), η οποία έχει υψηλότερη συγκέντρωση οξυγόνου και μπορούμε να σχεδιάσουμε αυτήν την αλλαγή σε γράφημα κορεσμού οξυγόνου πάνω από μερική πίεση οξυγόνου.



Σχήμα 2.7 «Διάγραμμα μερικής πίεσης οξυγόνου» [4]

Παράδοση οξυγόνου στους ιστούς

Το ποσοστό οξυγόνου που συνδέεται με την αιμοσφαιρίνη σχετίζεται με τη μερική πίεση του οξυγόνου (pO_2) σε μια δεδομένη θέση. Εξηγεί πώς όταν η οξυαιμοσφαιρίνη φτάσει σε έναν ιστό που καταναλώνει οξυγόνο (π.χ. σκελετικός μυς), το οξυγόνο θα αποσυνδεθεί λόγω του χαμηλότερου τοπικού pO_2 [6]. Αυτό το σύστημα επιτρέπει την παροχή οξυγόνου στις περιοχές που το χρειάζονται περισσότερο. Όταν ο δεσμός είναι συνεργάσιμος γίνεται ευεργετικός γιατί λειτουργεί στα άκρα όταν υπάρχει χαμηλό οξυγόνο τοπικά, δεν θέλουμε η αιμοσφαιρίνη να διατηρεί το οξυγόνο της στενά συνδεδεμένο. Ομοίως, όταν υπάρχει υψηλό οξυγόνο (π.χ. στην πνευμονική κυκλοφορία), θέλουμε η αιμοσφαιρίνη να προσλαμβάνει όσο περισσότερο οξυγόνο μπορεί[5].

2.4 Καρδιά και λειτουργία

Η καρδιά είναι ένας μυς με μέγεθος περίπου όσο η γροθιά ενός ανθρώπου και αποτελείται από τέσσερις μυώδεις κοιλότητες, δύο πάνω με λεπτά τοιχώματα τα οποία ονομάζονται κόλποι και δύο κάτω με παχύτερα τοιχώματα που ονομάζονται κοιλίες.

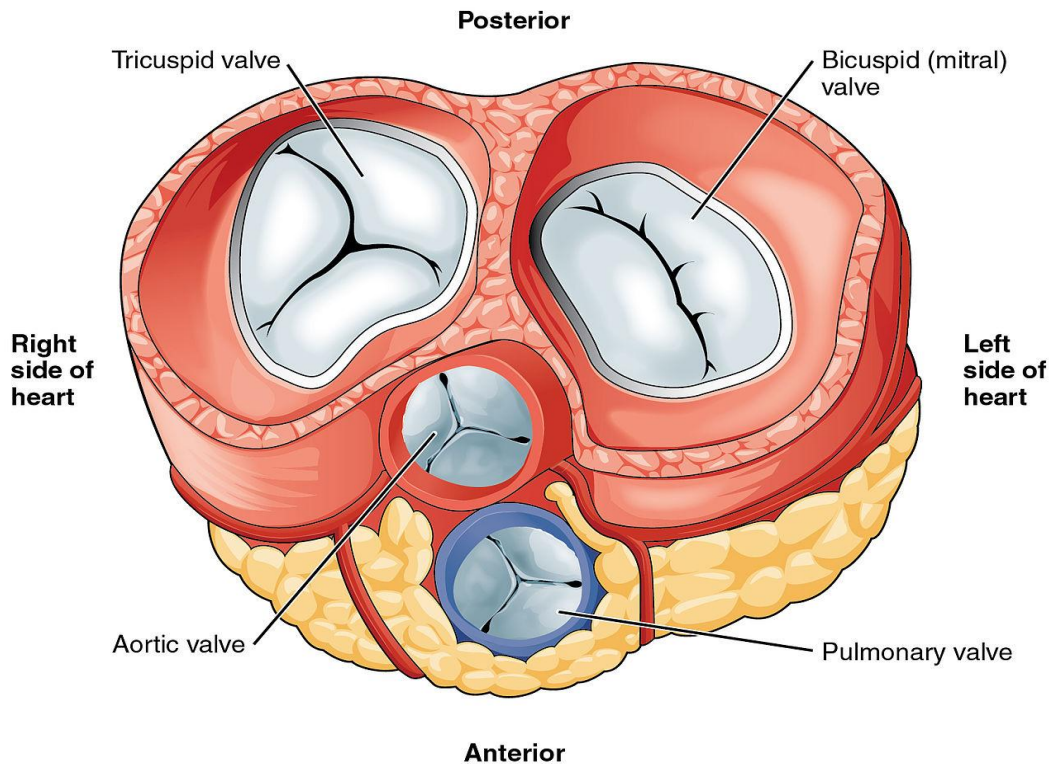
Ο δεξιός κόλπος της καρδιάς δέχεται το αίμα από όλα τα μέρη του σώματος μέσω των μεγάλων φλεβών, το προωθεί στη δεξιά κοιλία και από εκεί στην πνευμονική κυκλοφορία με στόχο την οξυγόνωσή του. Στη συνέχεια, το πλούσιο σε οξυγόνο αίμα προωθείται από τους πνεύμονες στον αριστερό κόλπο και από εκεί στην αριστερή κοιλία. Η τελευταία κοιλία αποτελεί το πιο «δυνατό» και σημαντικό τμήμα του μυοκαρδίου γιατί με τη συστολή της προωθεί το οξυγονωμένο πλέον αίμα σε όλο το σώμα, μέσω της αορτής και των μεγάλων αρτηριών. Στην αορτή το αίμα έχει πίεση πέντε ή έξι φορές υψηλότερη από ό,τι στην πνευμονική αρτηρία[7].

Βαλβίδες της καρδιάς

Η βαλβίδες που διαθέτει η καρδιά (Σχ. 2.9) χρησιμεύουν στο να επιτρέπουν την διόδο του αίματος προς μία μόνο κατεύθυνση και να εμποδίζουν την παλινδρόμησή του κατά τη διάρκεια της καρδιακής συστολής. Για παράδειγμα, η μιτροειδής βαλβίδα ανοίγει κατά τη διάρκεια της συστολής του αριστερού κόλπου και το αίμα περνάει στην αριστερή κοιλία, ενώ κλείνει κατά τη σύσπαση της κοιλίας και απαγορεύει την αντίστροφη ροή του αίματος προς τον κόλπο. Οι βαλβίδες έχουν λεπτές και σύνθετες δομές. Αποτελούνται από μικρά μέρη πολύ λεπτού αλλά ισχυρού ιστού, τις γλωχίνες που υποχρεώνουν το αίμα που βρίσκεται στην καρδιά να κυλάει προς μια μοναδική κατεύθυνση[8].

Αυτές οι βαλβίδες είναι:

- η τριγλώχινα μεταξύ δεξιού κόλπου και δεξιάς κοιλίας.
- η πνευμονική μεταξύ δεξιάς κοιλίας και πνευμονικής αρτηρίας.
- η μιτροειδής ή διγλώχινα μεταξύ αριστερού κόλπου και αριστερής κοιλίας.
- η αορτική μεταξύ αριστερής κοιλίας και αορτής.



Σχήμα 2.8 «Οι βαλβίδες της καρδιάς» [5]

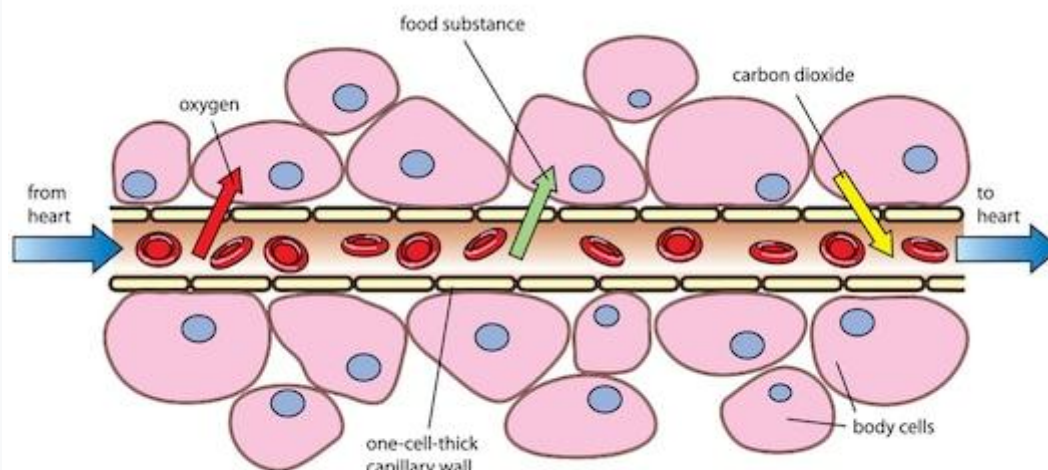
Ο μυς και οι βαλβίδες στηρίζονται πάνω σε ένα σκελετό από κολλαγόνο, ο οποίος ονομάζεται ινώδης σκελετός της καρδιάς[7]. Ο ινώδης σκελετός αποτελείται από τέσσερις δακτυλίους, οι οποίοι περιβάλλουν τις βαλβίδες και συνδέονται μεταξύ τους με δύο ινώδη τρίγωνα (δεξί και αριστερό) και δύο υμενώδεις μοίρες, το κολποκοιλιακό και το μεσοκοιλιακό διάφραγμα[8]. Ο σκελετός εμποδίζει την σύμπτωση των βαλβίδων και αποτελεί σημείο πρόσφυσης του μυοκαρδίου, των γλωχίνων των βαλβίδων ενώ ταυτόχρονα βοηθά στο διαχωρισμό της σύσπασης κόλπων και κοιλιών δρώντας σαν μονωτής του σήματος σύσπασης.

Για την αιμάτωσή της, η καρδιά έχει δύο αγγεία, την αριστερή και την δεξιά στεφανοειδή αρτηρία[7]. Η αρχή τους βρίσκεται στο αρχικό μέρος της αορτής. Η αριστερή είναι συνήθως πιο μεγάλη από τη δεξιά. Επειδή η αριστερή χωρίζεται, λίγο μετά την αρχή της, σε δύο κλάδους από λειτουργικής πλευράς, οι στεφανοειδής αρτηρίες μπορούν να θεωρηθούν και τρεις. Ο βασικός ρόλος των αυτών των αρτηριών είναι η παροχή οξυγόνου και, γενικότερα, θρεπτικών ουσιών στα κύτταρα του μυοκαρδίου[7].

Η λειτουργία του δεξιού κόλπου είναι να συλλέγει αποοξυγονωμένο αίμα από το σώμα (μέσω της άνω και κάτω κοίλης φλέβας) και να το προωθεί, μέσω της τριγλώχινας βαλβίδας, στην δεξιά κοιλία και αυτή με τη σειρά της στους πνεύμονες (πνευμονική κυκλοφορία), έτσι ώστε να μειωθεί η συγκέντρωση του διοξειδίου του άνθρακα και να αυξηθεί αυτή του οξυγόνου μέσω της ανταλλαγής αερίων. Αυτό συμβαίνει μέσα από την παθητική διαδικασία της διάχυσης[9].

Κεφάλαιο 2°

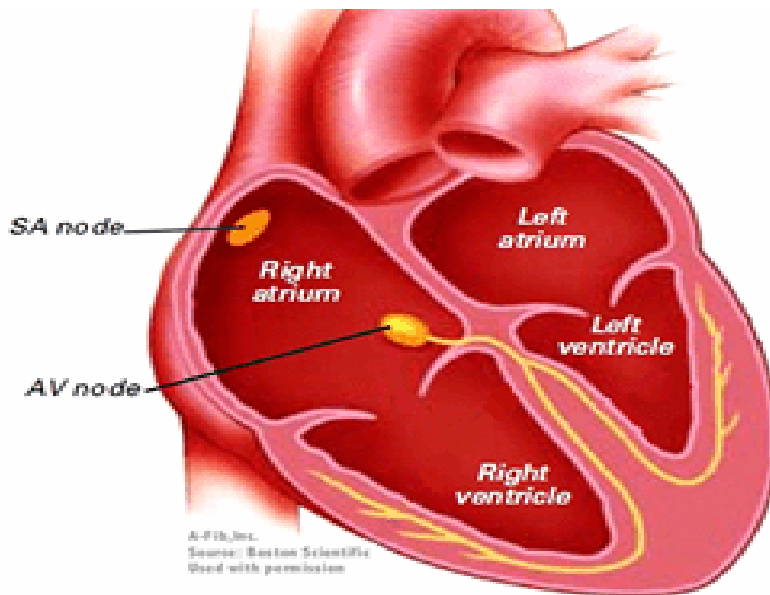
Η αριστερή πλευρά συλλέγει οξυγονωμένο αίμα από τους πνεύμονες στον αριστερό κόλπο. Από τον αριστερό κόλπο το αίμα κινείται προς την αριστερή κοιλία, μέσω της μιτροειδούς βαλβίδας, εκεί αντλείται προς το σώμα μέσω της αορτής. Στις δύο πλευρές, οι κάτω κοιλίες έχουν παχύτερα και πιο μυώδη τοιχώματα από τους κόλπους[9]. Το μυϊκό τοίχωμα της αριστερής κοιλίας είναι παχύτερο από το τοίχωμα της δεξιάς κοιλίας λόγω της υψηλότερης δύναμης που απαιτείται για την άντληση του αίματος μέσω της συστηματικής κυκλοφορίας.



Σχήμα 2.9 «Συστηματική κυκλοφορία του οξυγόνου στο αίμα» [6]

Κάποια κύτταρα στην καρδιά έχουν την ιδιότητα να αυτοδιεγύρονται χωρίς την παρουσία αντίστοιχου νευρικού σήματος. Τα κύτταρα αυτά βρίσκονται σε διάφορα σημεία στην καρδιά, στον φλεβοκόμβο, στον κολποκοιλιακό κόμβο και στις ίνες Purkinje[7]. Από αυτά, αυτό που αυτοδιεγύρεται ταχύτερα είναι ο φλεβοκόμβος και ακολουθεί τον ρυθμό που χτυπάει η καρδιά, όπως και τα υπόλοιπα κύτταρα. Ο φλεβοκόμβος βρίσκεται στην ένωση της άνω κοιλιακής φλέβας και του δεξιού κόλπου και αποτελείται από τροποποιημένα μυϊκά κύτταρα.

Τα κύτταρα παρουσιάζουν μια συνεχή διαρροή ιόντων νατρίου προς το εσωτερικό του κυττάρου με αποτέλεσμα το δυναμικό της μεμβράνης να αυξάνει μέχρι να φτάσει στο σημείο (ουδό) και παράγεται ένα δυναμικό ενέργειας. Μετά την εκπόλωση, το δυναμικό της μεμβράνης αυξάνει ξανά μέχρι να φτάσει στο σημείο να δώσει ένα νέο δυναμικό ενέργειας[9]. Το ρεύμα διαρροής ονομάστηκε παράδοξο ρεύμα επειδή οι επιστήμονες δεν περίμεναν την ύπαρξή του. Επειδή κύτταρα του μυοκαρδίου δρουν σαν ένα μεγάλο ενιαίο κύτταρο καθώς συνδέονται με κυτταροπλασματικές γέφυρες, το δυναμικό ενέργειας εξαπλώνεται από τον φλεβοκόμβο σε όλο τον κόλπο. Με την μεσοκοιλιακή δεσμίδα ή δεσμίδα Μπάχμαν, το σήμα άγεται στον αριστερό κόλπο ταχύτερα από ότι μέσω του μυοκαρδίου.



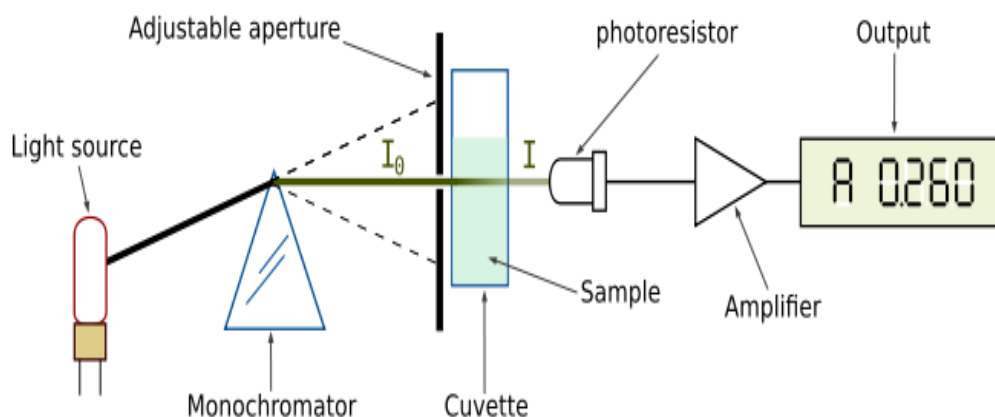
Σχήμα 2.10 «Μεσοκοιλικά διαφράγματα» [7]

Το σήμα τελικά φτάνει στο κολποκοιλιακό κόμβο (ΚΚΚ), τη μόνη οδό με την οποία ο ηλεκτρικός παλμός μπορεί φυσιολογικά να φτάσει στις κοιλίες[7]. Βρίσκεται στην οπίσθια δεξιά πλευρά του μεσοκοιλιακού διαφράγματος,(Σχ. 2.11) κοντά στο στόμιο του στεφανιαίου κόλπου και στην κοιλία. Ο κολποκοιλιακός κόμβος καθυστερεί την μετάβαση του σήματος. Αυτή η καθυστέρηση είναι αρκετή ώστε ο κόλπος να προωθήσει το αίμα στην αντίστοιχη κοιλία πριν αρχίσει αυτή να συσπάται. Μετά τον ΚΚΚ, το σήμα μεταβιβάζεται μέσω των ινών Purkinje, οι οποίες χωρίζονται σε δύο δεμάτια (του His), εκ των οποίων το αριστερό διατρυπά το μεσοκοιλιακό τοίχωμα. Οι ίνες αυτές τελικά διακλαδίζονται και φτάνουν σε όλη την έκταση των κοιλιών. Όλος αυτός ο κύκλος είναι ένας καρδιακός παλμός και διαρκεί περίπου 0,8 δευτερόλεπτα[7]. Κατά τη διάρκεια της μετάδοσης του σήματος παράγονται ηλεκτρικά ρεύματα που μεταδίδονται μέσω των υγρών του σώματος και το δέρμα και ανιχνεύονται με το ηλεκτροκαρδιογράφημα.

2.5 Η Φασματοφωτομετρία και ο νόμος Beer-Lambert

Φασματοφωτομετρία

Η φασματοφωτομετρία είναι μια μέθοδος μέτρησης της ποσότητας μιας χημικής ουσίας που απορροφά το φως μετρώντας την ένταση του φωτός καθώς μια δέσμη φωτός διέρχεται από το διάλυμα δείγματος[10]. Η βασική αρχή είναι ότι κάθε ένωση απορροφά ή μεταδίδει φως σε ένα συγκεκριμένο εύρος μήκους κύματος. Αυτή η μέτρηση μπορεί επίσης να χρησιμοποιηθεί για τη μέτρηση της ποσότητας μιας γνωστής χημικής ουσίας. Η φασματοφωτομετρία είναι μια από τις πιο χρήσιμες μεθόδους ποσοτικής ανάλυσης σε διάφορους τομείς όπως η χημεία, η φυσική, η βιοχημεία, η υλική και χημική μηχανική και οι κλινικές εφαρμογές.



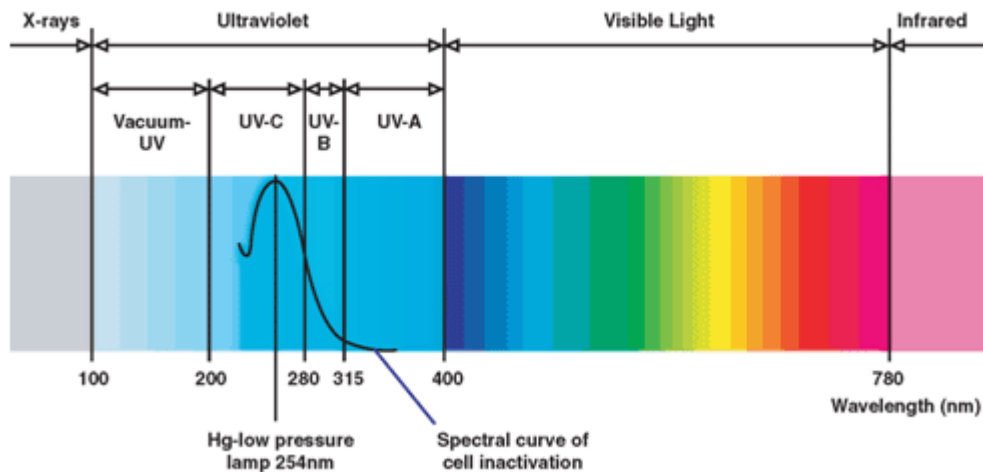
Σχήμα 2.11 «Μέτρηση δείγματος με φασματοφωτομετρία» [8]

Κάθε χημική ένωση απορροφά, μεταδίδει ή αντανακλά φως (ηλεκτρομαγνητική ακτινοβολία) σε ένα συγκεκριμένο εύρος μήκους κύματος. Η φασματοφωτομετρία είναι μια μέτρηση που απορροφά ή μεταδίδει μια χημική ουσία. Κάθε εφαρμογή που ασχολείται με χημικές ουσίες ή υλικά μπορεί να χρησιμοποιήσει αυτήν την τεχνική. Στη βιοχημεία, για παράδειγμα, χρησιμοποιείται για τον προσδιορισμό αντιδράσεων που καταλύονται από ένζυμα. Σε κλινικές εφαρμογές, χρησιμοποιείται για την εξέταση αίματος ή ιστών για κλινική διάγνωση. Υπάρχουν επίσης αρκετές παραλλαγές της φασματοφωτομετρίας όπως φασματοφωτομετρία ατομικής απορρόφησης και φασματοφωτομετρία ατομικής εκπομπής.

Ένα φασματοφωτόμετρο είναι ένα όργανο που μετρά την ποσότητα των φωτονίων (η ένταση του φωτός) που απορροφάται αφού περάσει από το διάλυμα δείγματος.

Με το φασματοφωτόμετρο, η ποσότητα μιας γνωστής χημικής ουσίας (συγκεντρώσεις) μπορεί επίσης να προσδιοριστεί μετρώντας την ένταση του φωτός που ανιχνεύεται. Ανάλογα με το εύρος του μήκους κύματος της πηγής φωτός, μπορεί να ταξινομηθεί σε δύο διαφορετικούς τύπους:

1. Φασματοφωτόμετρο ορατό με υπεριώδη ακτινοβολία: χρησιμοποιεί φως πάνω από την υπεριώδη περιοχή (185 - 400 nm) και ορατό εύρος (400 - 700 nm) φάσματος ηλεκτρομαγνητικής ακτινοβολίας. (Σχ. 2.13).
2. Φασματοφωτόμετρο IR: χρησιμοποιεί φως πάνω από την υπέρυθη περιοχή (700 - 15000 nm) φάσματος ηλεκτρομαγνητικής ακτινοβολίας. (Σχ. 2.13).



Σχήμα 2.12 «Υπεριώδης ακτινοβολία» [9]

Στην ορατή φασματοφωτομετρία, η απορρόφηση ή η μετάδοση μιας συγκεκριμένης ουσίας μπορεί να προσδιοριστεί από το παρατηρούμενο χρώμα[10]. Για παράδειγμα, ένα δείγμα διαλύματος που απορροφά φως σε όλες τις ορατές περιοχές (δηλαδή, δεν μεταδίδει κανένα από τα ορατά μήκη κύματος) εμφανίζεται μαύρο στη θεωρία. Από την άλλη πλευρά, εάν μεταδίδονται όλα τα ορατά μήκη κύματος (δηλαδή, δεν απορροφά τίποτα), το δείγμα διαλύματος εμφανίζεται λευκό. Εάν ένα δείγμα διαλύματος απορροφά κόκκινο φως (~ 700 nm), εμφανίζεται πράσινο επειδή το πράσινο είναι το συμπληρωματικό χρώμα του κόκκινου. Τα ορατά φασματοφωτόμετρα, στην πράξη, χρησιμοποιούν ένα πρίσμα για να περιορίσουν ένα ορισμένο εύρος μήκους κύματος (για να φιλτράρετε άλλα μήκη κύματος) έτσι ώστε η συγκεκριμένη δέσμη φωτός να περάσει μέσω ενός δείγματος διαλύματος.

Νόμος Beer-Lambert

Ο νόμος Beer-Lambert (επίσης γνωστός ως νόμος Beer) δηλώνει ότι υπάρχει μια γραμμική σχέση μεταξύ της απορρόφησης και της συγκέντρωσης ενός δείγματος[12]. Για το λόγο αυτό, ο νόμος του Beer μπορεί να εφαρμοστεί μόνο όταν υπάρχει γραμμική σχέση. Αναγράφεται ως εξής:

$$A = \epsilon l c$$

όπου:

- **A** είναι το μέτρο απορρόφησης (χωρίς μονάδες).
- **ϵ** είναι ο συντελεστής μοριακής απόσβεσης ($M^{-1}cm^{-1}$) ή η γραμμομοριακή απορροφητικότητα (ή ο συντελεστής απορρόφησης).
- **l** είναι το μήκος της διαδρομής (M).
- **c** είναι η συγκέντρωση (cm).

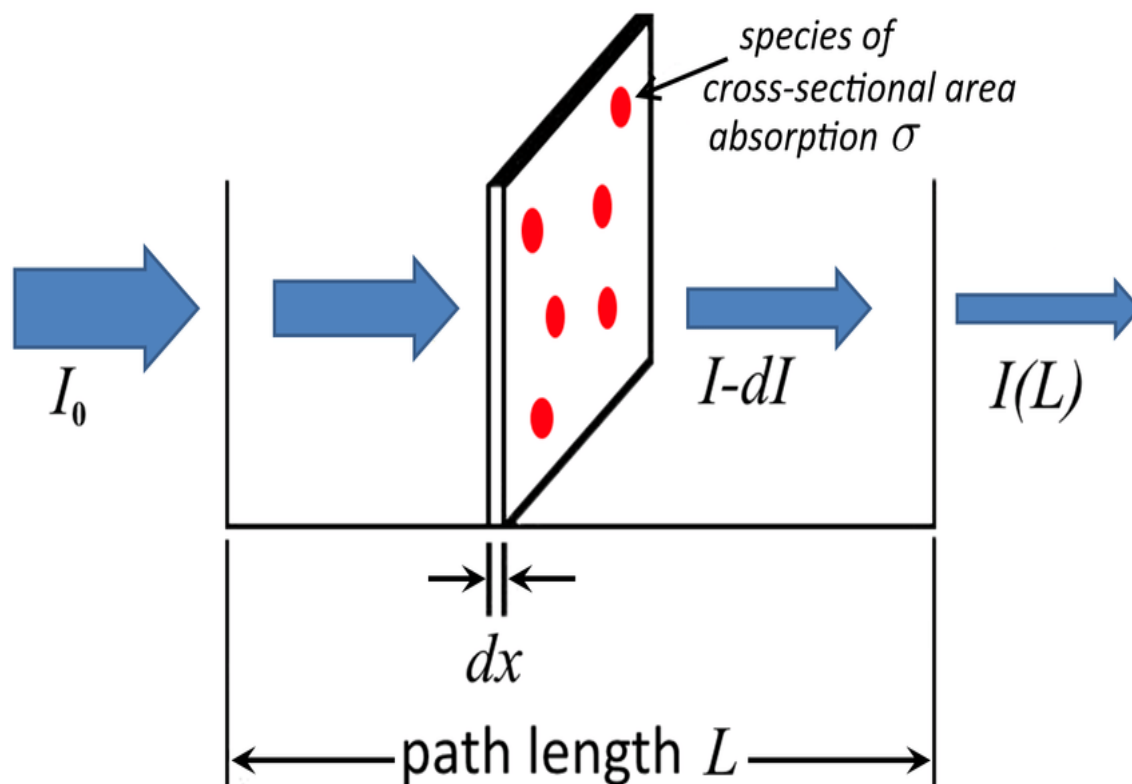
Ο συντελεστής μοριακής απόσβεσης δίνεται ως σταθερά και ποικίλλει για κάθε μόριο. Επειδή η απορρόφηση δεν φέρει καμία μονάδα, οι μονάδες για ϵ πρέπει να ακυρώσουν τις μονάδες μήκους και συγκέντρωσης.

Ως αποτέλεσμα, το ϵ έχει τις μονάδες: $L \cdot \text{mol}^{-1} \cdot \text{cm}^{-1}$. Το μήκος της διαδρομής μετράται σε εκατοστά. Επειδή ένα τυπικό φασματόμετρο χρησιμοποιεί μια κυψελίδα πλάτους 1 cm, θεωρείται πάντα ίση με 1 cm. Δεδομένου ότι είναι γνωστή η απορρόφηση, το ϵ και το μήκος διαδρομής, μπορούμε να υπολογίσουμε τη συγκέντρωση c του δείγματος[11].

Ανάλυση του νόμου

Ένα φασματόμετρο είναι μία συσκευή που μετρά την ένταση, την ενέργεια που μεταφέρεται από την ακτινοβολία ανά μονάδα περιοχής ανά μονάδα χρόνου, του φωτός που εισέρχεται σε ένα διάλυμα δείγματος και του φωτός που εξέρχεται από ένα διάλυμα δείγματος. Οι δύο εντάσεις μπορούν να εκφραστούν ως μετάδοση: η αναλογία της έντασης του εξερχόμενου φωτός προς το φως εισόδου ή το ποσοστό μετάδοσης (%T)[13]. Διαφορετικές ουσίες απορροφούν διαφορετικά μήκη κύματος φωτός.

Επομένως, το μήκος κύματος της μέγιστης απορρόφησης από μία ουσία είναι μία από τις χαρακτηριστικές ιδιότητες αυτού του υλικού. Μία εντελώς διαφανής ουσία θα έχει $I_t=I_0$ και η εκατοστιαία διαπερατότητά της θα είναι 100. Όμως, μία ουσία που δεν επιτρέπει σε καμία ακτινοβολία συγκεκριμένου μήκους κύματος να περάσει από αυτήν θα έχει $I_t=0$ και αντίστοιχη εκατοστιαία μετάδοση 0 (Σχ. 2.14).



Σχήμα 2.13 «Διαπερατότητα και απορρόφηση» [10]

Διαπερατότητα

$$T = I_t / I_0$$

$$\% \text{ Transmittance: } \%T = 100 T$$

Απορρόφηση

$$A = \log_{10} (I_0/I_i)$$

$$A = \log_{10} (1/T) = -\log_{10} (T)$$

$$A = \log_{10} (100/\%T)$$

$$A = 2 - \log_{10} (\%T)$$

- Η μετάδοση για υγρά γράφεται συνήθως ως: $T = I/I_0 = 10^{-ad} = 10^{\Sigma lc'}$
- Η μετάδοση για αέρια γράφεται ως: $T = I/I_0 = 10^{-a}l = e^{-\sigma l N}$
- I_0 και I είναι η ένταση (ή ισχύς) του προσπίπτοντος φωτός και του εκπεμπόμενου φωτός αντίστοιχα.
- Η απορρόφηση για υγρά γράφεται ως: $A = -\log_{10} (I/I_0)$
- Η απορρόφηση για αέρια γράφεται ως: $A' = -\ln(I/I_0)$

Αποκλίσεις

Ο νόμος Beer-Lambert διατηρεί τη γραμμικότητα μόνο από συγκεκριμένες συνθήκες. Ο νόμος θα κάνει ανακριβείς μετρήσεις σε υψηλές συγκεντρώσεις επειδή τα μόρια του αναλυτή εμφανίζουν ισχυρότερες διαμοριακές και ηλεκτροστατικές αλληλεπιδράσεις που οφείλονται στο μικρότερο διάστημα μεταξύ των μορίων. Αυτό μπορεί να αλλάξει τη μοριακή απορροφητικότητα του αναλυτή. Οι ψηλές συγκεντρώσεις δεν αλλάζουν μόνο τη μοριακή απορροφητικότητα, αλλά αλλάζει τον δείκτη διάθλασης της λύσης που προκαλεί αποχωρήσεις από τον νόμο Beer-Lambert[12].

Εφαρμογές

Η νομοθεσία Beer-Lambert εφαρμόζεται στην ανάλυση ενός μείγματος με φασματοφωτομετρία, χωρίς την ανάγκη εκτεταμένης προεπεξεργασίας του δείγματος. Παραδείγματα περιλαμβάνουν τον προσδιορισμό της χολερυθρίνης σε δείγματα πλάσματος αίματος. Οι μετρήσεις γίνονται σε ένα συγκεκριμένο μήκος κύματος σχεδόν μοναδικό για τη χολερυθρίνη και μία άλλη μέτρηση σε ένα δεύτερο μήκος κύματος έτσι ώστε οι παρεμβολές ή οι αποκλίσεις να μπορούν να εξαλειφθούν ή να διορθωθούν. Γενικά, μπορεί να χρησιμοποιηθεί για τον προσδιορισμό των συγκεντρώσεων μιας συγκεκριμένης ουσίας ή για τον προσδιορισμό της γραμμομοριακής απορροφητικότητας μιας ουσίας[11].

2.6 Επίλογος

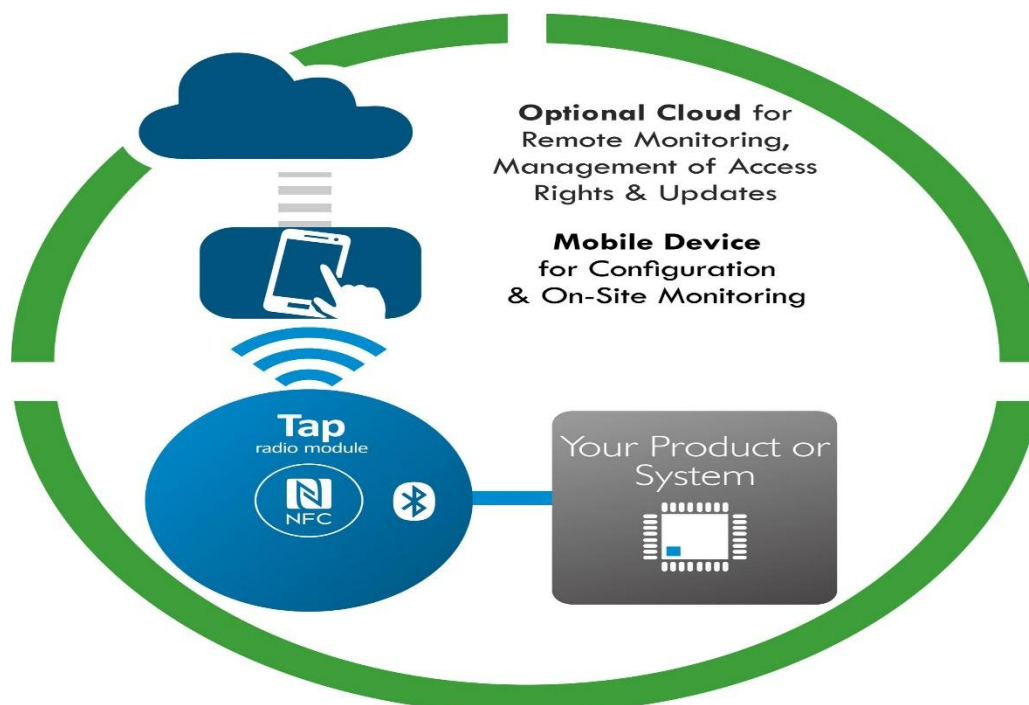
Σε αυτό το κεφάλαιο κάναμε μια περιγραφή λειτουργίας του παλμικού οξυμέτρου και αναλύσαμε την φασματοφωτομετρία και το νόμο που βασίζεται. Έγινε επίσης μια περιγραφή λειτουργίας της καρδιάς και του τρόπου που το οξυγόνο ταξιδεύει μέσω αυτής σε όλο το αίμα.

Κεφάλαιο 3^ο : Ανάλυση εξαρτημάτων κατασκευής

3.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα αναλύσουμε λεπτομερώς τα εξαρτήματα που χρησιμοποιήθηκαν για την κατασκευή του παλμικού οξυμέτρου, με ποια κριτήρια τα επιλέξαμε και πως χρησιμοποιήσαμε πρόσφατες τεχνολογίες ώστε να το αναβαθμίσουμε και να το κάνουμε πιο εύχρηστο για τον χρήστη.

3.2 Bluetooth και BLE



Σχήμα 3.1 «Point-to-point σύνδεση» [11]

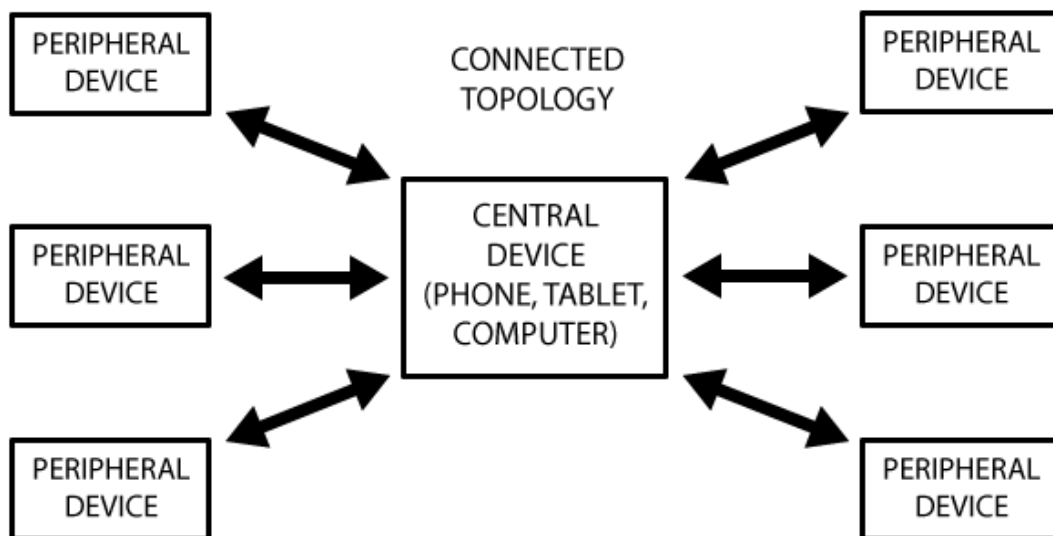
Το Bluetooth είναι μια ασύρματη τεχνολογία μεταφοράς δεδομένων μικρών αποστάσεων. Όπως το Wifi, έτσι και το Bluetooth λειτουργεί στα 2,4GHz και χρησιμοποιείται σε ασύρματες επικοινωνίες, στην επικοινωνία περιφερειακών (ακουστικά, ποντίκι, πληκτρολόγιο) με μονάδες H/Y και πολλές ακόμα εφαρμογές. Σε αντίθεση με το WiFi οι συνδέσεις Bluetooth δεν χρησιμοποιούν το πακέτο δεδομένων κινητής τηλεφωνίας επειδή δεν μεταδίδουν δεδομένα μέσω των κυμάτων αέρα με τον ίδιο τρόπο. Το Bluetooth είναι αποκλειστικά για μετάδοση συσκευής σε συσκευή (Point-to-Point (Σχ. 3.1), η απλούστερη σύνδεση μεταξύ δύο τερματικών / υπολογιστικών σημείων που επιτυγχάνεται με μια απ' ευθείας σύνδεση) όπου χρειάζεται για την μεταφορά δεδομένων με συνεχή σύνδεση[14]. Λειτουργεί συνήθως σε μικρές αποστάσεις (συνήθως απόσταση 10 μέτρων) για εξοικονόμηση ενέργειας. Πλέον είναι βασικό κομμάτι της καθημερινότητας αφού η ασύρματη επικοινωνία υπάρχει παντού και χρησιμοποιείται συνέχεια.

Πως λειτουργεί

Το Bluetooth στέλνει και λαμβάνει ραδιοκύματα σε μια ζώνη 79 διαφορετικών συχνοτήτων (κανάλια) με επίκεντρο τα 2,4GHz, διαχωρίζονται σε ραδιόφωνο, τηλεόραση, κινητά τηλέφωνα και προορίζονται για χρήση από βιομηχανικά, επιστημονικά και ιατρικά gadget.

Οι πομποί μικρής εμβέλειας των Bluetooth είναι ένα από τα μεγαλύτερα πλεονεκτήματα. Δεν χρησιμοποιούν σχεδόν καθόλου ισχύ και επειδή δεν ταξιδεύουν σε μεγάλη απόσταση τους καθιστά πιο ασφαλή από τα ασύρματα δίκτυα που λειτουργούν σε μεγαλύτερες αποστάσεις, όπως το WiFi[14]. Οι συσκευές Bluetooth εντοπίζουν και συνδέονται αυτόματα μεταξύ τους και μπορούν να επικοινωνήσουν ανά πάσα στιγμή έως και οκτώ ταυτόχρονα (Σχ. 3.2). Δεν παρεμβαίνουν μεταξύ του επειδή κάθε συσκευή χρησιμοποιεί ένα διαφορετικό από τα 79 διαθέσιμα κανάλια που έχει. Εάν δύο συσκευές θέλουν να επικοινωνήσουν επιλέγουν ένα τυχαίο κανάλι και εάν αυτό δεν είναι διαθέσιμο θα μεταβεί τυχαία σε κάποιο άλλο (η τεχνική αυτή ονομάζεται hopping). Για να ελαχιστοποιηθούν οι κίνδυνοι παρεμβολών από άλλες ηλεκτρικές συσκευές και να βελτιωθεί η ασφάλεια τα ζεύγη αλλάζουν συνεχώς συχνότητα.

Όταν μια ομάδα δύο ή περισσότερων συσκευών Bluetooth μοιράζονται πληροφορίες σχηματίζουν ένα είδος δικτύου μίνι υπολογιστών που ονομάζεται Piconet. Μπορούν να συνδεθούν και άλλες συσκευές σε ένα υπάρχον Piconet ανά πάσα στιγμή. Μία συσκευή (master) λειτουργεί ως ο γενικό ελεγκτής δικτύου ενώ οι άλλες (slaves) τηρούν τις οδηγίες του. Δύο ή περισσότερα ξεχωριστά Piconets μπορούν επίσης να ενώσουν και να μοιραστούν πληροφορίες σχηματίζοντας το Scatternet[15].



Σχήμα 3.2 «Τοπολογία σύνδεσης» [12]

Εκδόσεις των Bluetooth

1. Bluetooth 1.0 και 1.0B

Δυσκόλεψε τους κατασκευαστές να κάνουν τα προϊόντα τους απόλυτα λειτουργικά.

Είχαν υποχρεωτικά μετάδοση διεύθυνσης υλικού του Bluetooth (BD_ADDR) κατά την διαδικασία σύνδεσης καθυστετώντας την ανωνυμία αδύνατη σε επίπεδα πρωτοκόλλου, κάτι το οποίο αναστάτωνε ορισμένες υπηρεσίες που σχεδιάστηκαν για χρήση σε περιβάλλοντα Bluetooth[16].

2. Bluetooth 1.2

Έγιναν σημαντικές βελτιώσεις σε σύγκριση με την προηγούμενη τεχνολογία:

- Πιο γρήγορη σύνδεση και εύρεση.
- Καλύτερο φάσμα μετατόπισης συχνότητας (FHSS) το οποίο ενισχύει την αντίσταση στις παρεμβολές ραδιοφωνικών συχνοτήτων.
- Υψηλότερες ταχύτητες μετάδοσης (Εώς και 720 Kbit/s.
- Extended Synchronous Connections (eSCO), οι οποίες βελτίωσαν την ποιότητα ήχου των ηχητικών συνδέσεων επιτρέποντας την αναμετάδοση κατεστραμμένων πακέτων.
- Host Controller Interface (HCI) με UART τριών καλωδίων.

3. Bluetooth 2.0 + EDR

Κύρια διαφορά είναι η εισαγωγή βελτιωμένου ρυθμού δεδομένων () για ταχύτερη μεταφορά δεδομένων. Το EDR έχει χαμηλότερη κατανάλωση ενέργειας, ο ρυθμός bit είναι 3Mbit/s αλλά ο μέγιστος ρυθμός μεταφοράς δεδομένων είναι 2,1Mbits/s[16].

4. Bluetooth 3.0

Οι ταχύτητες βελτιώθηκαν και πάλι από το Bluetooth 3.0 (2009) αυξάνοντας τους ρυθμούς μεταφοράς δεδομένων στα 24Mbit/s και αξιοποίησε το πρότυπο 802.11 (χρησιμοποιείται σε WiFi).

5. Bluetooth 4.2

Η μεγαλύτερη αναβάθμιση ήρθε με την τέταρτη γενιά Bluetooth, ωστόσο ήρθε σε επόμενη έκδοση (4.2) με BLE (Bluetooth Low Energy). Σήμερα τα περισσότερα smartphone σε όλο τον κόσμο υποστηρίζουν το 4.2 το οποίο επιτρέπει παρόμοια κάλυψη και παρόμοιο εύρος ζώνης αλλά έχει μεγάλες μειώσεις στην απαίτηση ισχύος κατά την μεταφορά δεδομένων (από 50% ως 10.000% ανάλογα την περίπτωση χρήσης). Αυτό είχε σχεδιαστεί για να μειώνει όχι μόνο τις απαιτήσεις του Bluetooth σε συσκευές ελεγχόμενες από κάποιο χρήστη αλλά και για να διευκολύνει τη λειτουργία φορητών ηχείων, ακουστικών και fitness trackers.

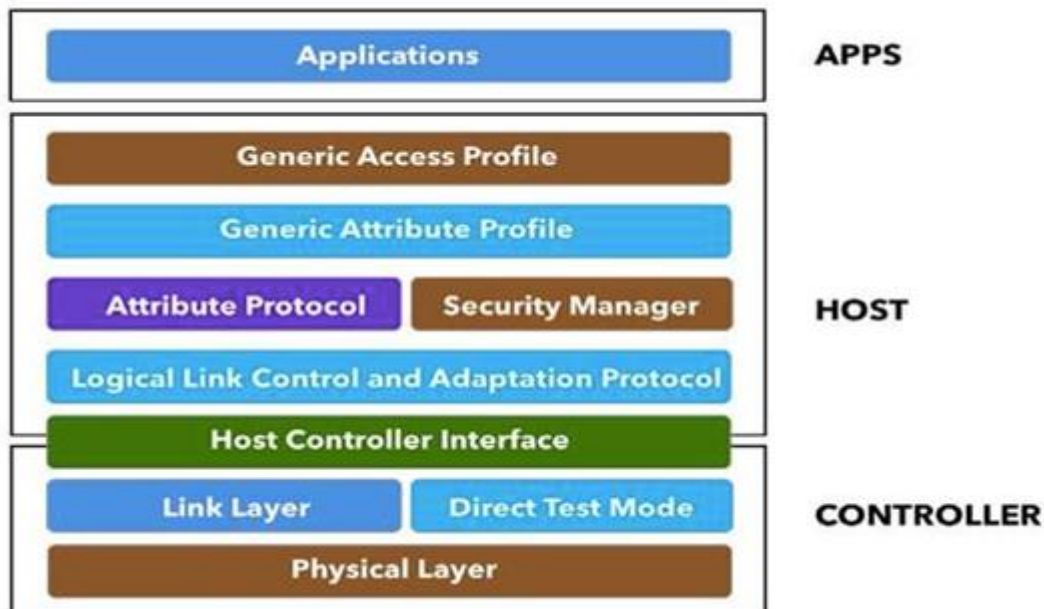
6. Bluetooth 5.0

Το Bluetooth 5 παρουσιάστηκε το 2016 με βελτιωμένη ασφάλεια και μεγαλύτερη ευελιξία για καλύτερη ταχύτητα και εμβέλεια. Οι συσκευές που υποστηρίζουν Bluetooth 5.0 μπορούν να «θυσιάσουν» η μία την άλλη, αυξάνοντας από 2 ως 4 φορές.

Το Bluetooth 5.1 κυκλοφόρησε στα τέλη του 2019 και πρόσθεσε μια λειτουργία για τον εντοπισμό της φυσικής θέσης των συσκευών μετάδοσης η οποία θα μπορούσε να είναι χρήσιμη για την εύρεση χαμένων αντικειμένων που διαθέτουν Bluetooth. Θα μπορούσε επίσης να χρησιμοποιηθεί για την ενεργοποίηση της προβολής σχετικού περιεχομένου σε οθόνες δημόσιων χώρων ή μουσείων.

Bluetooth Low Energy

Κυκλοφόρησε στην αγορά το 2011 ως το Bluetooth 4.0. Όταν μιλάμε για Bluetooth χαμηλής ενέργειας έναντι Bluetooth, η βασική διαφορά είναι η χαμηλή κατανάλωση ενέργειας του Bluetooth 4.0. Με την κατανάλωση ενέργειας του BLE, οι εφαρμογές με χαμηλή μπαταρία μπορούν να λειτουργούν έως και 5 χρόνια. Αν και δεν είναι ιδανικό για τηλεφωνική συσκευή, είναι πολύ σημαντικό για ιατρικές συσκευές και gadget, εφαρμογές ζωτικής σημασίας που πρέπει να ανταλλάσσουν περιοδικά μικρές ποσότητες δεδομένων.[17]



Σχήμα 3.3 «Αρχιτεκτονική του Bluetooth χαμηλής ενέργειας» [13]

Όπως το Bluetooth έτσι και το BLE λειτουργεί στη ζώνη ISM 2.4GHz. Σε αντίθεση με το κλασικό Bluetooth, το BLE παραμένει σε κατάσταση αναστολής συνεχώς, εκτός από την έναρξη σύνδεσης. Οι πραγματικοί χρόνοι σύνδεσης είναι μόλις μερικά ms, σε αντίθεση με το Bluetooth που θα διαρκούσε περίπου 100ms[17].

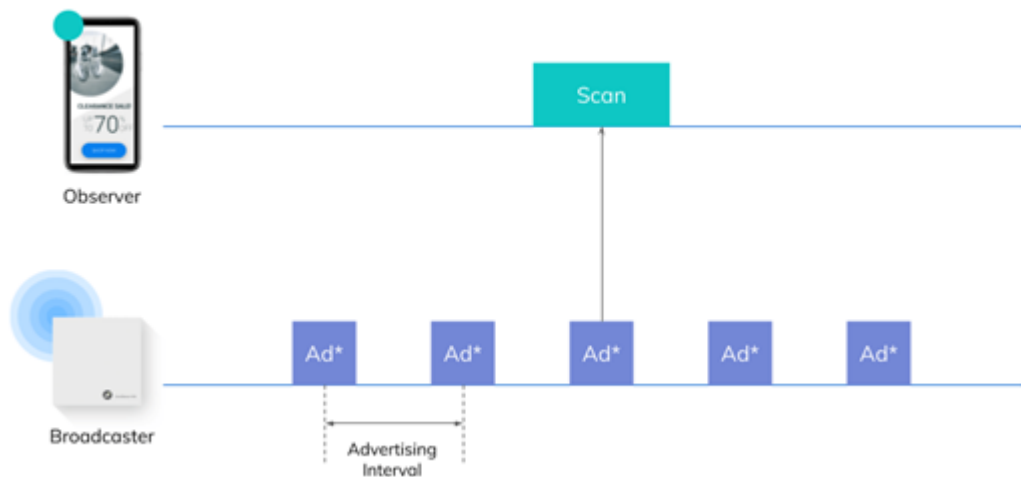
Ρόλος και ευθύνες

Για να κατανοήσετε τη διάκριση, φανταστείτε ότι διαθέτετε τηλέφωνο Android και πρόγραμμα παρακολούθησης δραστηριοτήτων που είναι η συσκευή BLE. Το τηλέφωνο υποστηρίζει τον κεντρικό ρόλο, ο ιχνηλάτης δραστηριότητας υποστηρίζει τον περιφερειακό ρόλο (για να δημιουργήσετε μια σύνδεση BLE χρειάζεστε ένα από τα δύο - δύο πράγματα που υποστηρίζουν μόνο περιφερειακά δεν μπορούν να επικοινωνήσουν μεταξύ τους)[17].

Μόλις το τηλέφωνο και ο ιχνηλάτης δραστηριότητας αποκτήσουν σύνδεση, αρχίζουν να μεταφέρουν δεδομένα GATT μεταξύ τους. Ανάλογα με το είδος των δεδομένων που μεταφέρουν, το ένα ή το άλλο μπορεί να λειτουργήσει ως διακομιστής. Για παράδειγμα, εάν ο ιχνηλάτης δραστηριότητας θέλει να στείλει δεδομένα αισθητήρα στο τηλέφωνο, μπορεί να λειτουργεί ο ιχνηλάτης δραστηριότητας ως διακομιστής. Εάν το πρόγραμμα παρακολούθησης δραστηριότητας θέλει να λαμβάνει ενημερώσεις από το τηλέφωνο, τότε μπορεί να έχει το τηλέφωνο να λειτουργεί ως διακομιστής.

Εύρεση του BLE

Οι συσκευές BLE εντοπίζονται μέσω μιας διαδικασίας που βασίζεται στη μετάδοση πακέτων. Αυτό γίνεται χρησιμοποιώντας 3 ξεχωριστά κανάλια (συχνότητες), προκειμένου να μειωθεί η παρεμβολή[18]. Η συσκευή στέλνει ένα πακέτο σε τουλάχιστον ένα από αυτά τα τρία κανάλια, με μια περίοδο επανάληψης που ονομάζεται «διάστημα διαφημίσεων». Για τη μείωση της πιθανότητας πολλαπλών διαδοχικών συγκρούσεων, προστίθεται τυχαία καθυστέρηση έως 10 χιλιοστά του δευτερολέπτου σε κάθε διάστημα. Ο σαρωτής ακούει το κανάλι για μια διάρκεια που ονομάζεται παράθυρο σάρωσης, το οποίο επαναλαμβάνεται περιοδικά σε κάθε διάστημα σάρωσης.



Σχήμα 3.4 «Ανίχνευση και επικοινωνία Bluetooth χαμηλής ενέργειας» [14]

Η καθυστέρηση εύρεσης καθορίζεται επομένως από μια διαδικασία και εξαρτάται από τις τρεις παραμέτρους (δηλαδή, το διάστημα διαφήμισης, το διάστημα σάρωσης και το παράθυρο σάρωσης). Το σχήμα ανακάλυψης του BLE υιοθετεί μια τεχνική βασισμένη σε περιοδικά διαστήματα, στην οποία μπορούν να συναχθούν ανώτατα όρια στο λανθάνοντα χρόνο ανακάλυψης για περισσότερες μετρήσεις. Ενώ οι καθυστερήσεις εύρεσης του BLE μπορούν να προσεγγιστούν από μοντέλα για καθαρά περιοδικά πρωτόκολλα που βασίζονται σε διαστήματα, η τυχαία καθυστέρηση που προστίθεται σε κάθε διάστημα διαφήμισης και η ανακάλυψη τριών καναλιών μπορεί να προκαλέσει αποκλίσεις ή ενδεχομένως να οδηγήσει σε μη περιορισμό ορισμένων παραμέτρων.

Μοντέλο λογισμικού

Όλες οι συσκευές Bluetooth χαμηλής ενέργειας χρησιμοποιούν το γενικό χαρακτηριστικό προφίλ (GATT)[19]. Η διεπαφή προγραμματισμού εφαρμογών που προσφέρει ένα λειτουργικό σύστημα με Bluetooth χαμηλής ενέργειας θα βασίζεται συνήθως σε έννοιες όπως το GATT.

Το GATT έχει την ακόλουθη ορολογία:

- Πελάτης (Client)

Μια συσκευή που δίνει εντολές και αιτήματα GATT, για παράδειγμα, ένας υπολογιστής ή ένα smartphone.

- Υπηρετής (Server)

Μια συσκευή που λαμβάνει εντολές και αιτήματα GATT και τα επιστρέφει όπως για παράδειγμα ένας αισθητήρας θερμοκρασίας.

- Υπηρεσία (Service)

Μια συλλογή σχετικών χαρακτηριστικών, τα οποία λειτουργούν μαζί για να εκτελέσουν μια συγκεκριμένη λειτουργία. Για παράδειγμα η υπηρεσία Health Thermometer περιλαμβάνει χαρακτηριστικά για μια τιμή μέτρησης θερμοκρασίας και ένα χρονικό διάστημα μεταξύ των μετρήσεων.

- Περιγραφέας (Descriptor)

Ένας περιγραφέας παρέχει πρόσθετες πληροφορίες σχετικά με ένα χαρακτηριστικό. Για παράδειγμα, ένα χαρακτηριστικό τιμής θερμοκρασίας μπορεί να έχει ένδειξη των μονάδων του (π.χ. Κελσίου) και τις μέγιστες και ελάχιστες τιμές τις οποίες μπορεί να μετρήσει ο αισθητήρας. Οι περιγραφείς είναι προαιρετικοί, κάθε χαρακτηριστικό μπορεί να έχει οποιονδήποτε αριθμό περιγραφών.

Ορισμένες τιμές και υπηρεσίες χρησιμοποιούνται για διοικητικούς σκοπούς - για παράδειγμα, το όνομα του μοντέλου και ο σειριακός αριθμός μπορούν να αναγνωστούν ως τυπικά χαρακτηριστικά στην υπηρεσία Generic Access. Οι υπηρεσίες μπορεί επίσης να περιλαμβάνουν άλλες υπηρεσίες ως δευτερεύουσες λειτουργίες. Οι κύριες λειτουργίες της συσκευής είναι οι λεγόμενες κύριες υπηρεσίες και οι βοηθητικές λειτουργίες στις οποίες αναφέρονται είναι δευτερεύουσες υπηρεσίες.

- Αναγνώστης (Identifiers)

Οι υπηρεσίες, τα χαρακτηριστικά και οι περιγραφείς αναφέρονται συλλογικά ως χαρακτηριστικά και προσδιορίζονται από το UUID. Οποιοσδήποτε μπορεί να εφαρμόσει και να επιλέξει ένα τυχαίο ή ψευδοτυχαίο UUID για ιδιόκτητες χρήσεις, αλλά το Bluetooth SIG έχει δεσμεύσει μια σειρά UUID (της μορφής xxxxxxxx-0000-1000-8000-00805F9B34FB) για τυπικά χαρακτηριστικά. Αυτοί οι αναγνώστες παρουσιάζονται ως τιμές 16-bit ή 32-bit στο πρωτόκολλο, αντί για τα απαιτούμενα 128 bit για ένα πλήρες UUID. Για παράδειγμα, η υπηρεσία πληροφοριών συσκευής έχει τον σύντομο κωδικό 0x180A αντί για 0000180A-0000-1000.

Κατανάλωση μπαταρίας

Το BLE έχει σχεδιαστεί για να επιτρέπει σε συσκευές να έχουν χαμηλή κατανάλωση ενέργειας. Αρκετοί κατασκευαστές chip έχουν εισάγει το BLE chipset από το 2014. Οι συσκευές με περιφερειακούς και κεντρικούς ρόλους έχουν διαφορετική απαίτηση ισχύος. Μια μελέτη από την εταιρία λογισμικού Aislelabs αναφέρει ότι τα περιφερειακά που τροφοδοτούνται με μπαταρία των 1000mAh λειτουργούν συνήθως για 1-2 έτη. Αυτό είναι δυνατό λόγω της απόδοσης ισχύος του BLE το οποίο μεταδίδει μόνο μικρά πακέτα σε σύγκριση με το κλασικό Bluetooth[20]. Οι συσκευές Android και iOS έχουν επίσης πολύ διαφορετικό αντίκτυπο μπαταρίας ανάλογα με τον τύπο σάρωσης και τον αριθμό των συσκευών BLE στην περιοχή. Με τα νεότερα chipset και την εξέλιξη στο λογισμικό, έως το 2014 και τα δύο τηλέφωνα είχαν αμελητέα κατανάλωση ενέργειας σε πραγματική χρήση του BLE.

3.3 LoRa(WAN)

Το LoRa είναι μια τεχνική διαμόρφωσης φάσματος διασποράς που προέρχεται από την τεχνολογία Chirp Spread Spectrum (CSS). Οι συσκευές LoRa της Semtech και η ασύρματη τεχνολογία ραδιοσυχνότητας είναι μια ασύρματη πλατφόρμα μεγάλης εμβέλειας, χαμηλής ισχύος και έχει γίνει εκ των πραγμάτων η τεχνολογία για δίκτυα Internet of Things (IoT) παγκοσμίως. Οι συσκευές LoRa και το ανοιχτό πρωτόκολλο LoRaWAN επιτρέπουν έξυπνες εφαρμογές IoT που επιλύουν μερικές από τις μεγαλύτερες προκλήσεις που αντιμετωπίζει ο πλανήτης μας: διαχείριση ενέργειας, μείωση φυσικών πόρων, έλεγχος ρύπανσης, αποδοτικότητα υποδομής, πρόληψη καταστροφών και πολλά άλλα. Οι συσκευές LoRa της Semtech και το πρωτόκολλο LoRaWAN έχουν συγκεντρώσει αρκετές εκατοντάδες γνωστές περιπτώσεις χρήσεων για έξυπνες πόλεις, σπίτια, κτίρια, γεωργία, έξυπνη μέτρηση, έξυπνη αλυσίδα εφοδιασμού, logistics και πολλά ακόμα. Με πάνω από 100 εκατομμύρια συσκευές συνδεδεμένες σε δίκτυα 100 χωρών, οι συσκευές LoRa δημιουργούν έναν πιο έξυπνο πλανήτη.

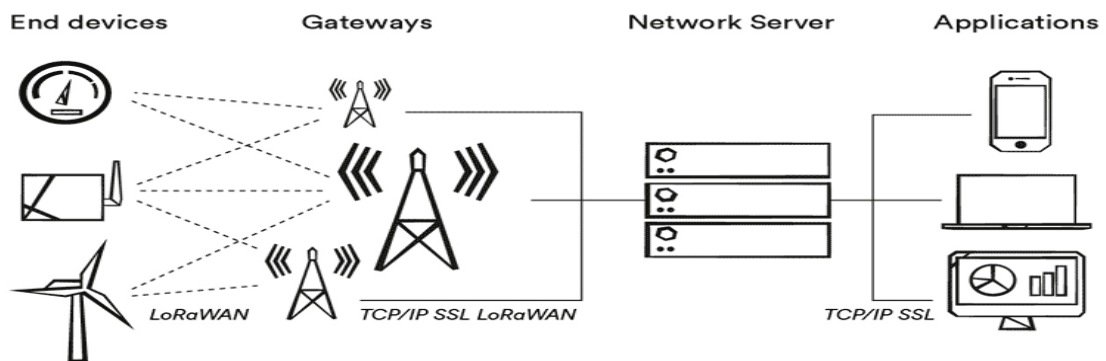
Η ασύρματη τεχνολογία RF LoRa

Οι συσκευές LoRa προσφέρουν συναρπαστικά χαρακτηριστικά για εφαρμογές IoT, συμπεριλαμβανομένων μεγάλων αποστάσεων, χαμηλή κατανάλωση ενέργειας και ασφαλή μετάδοση δεδομένων. Η τεχνολογία μπορεί να χρησιμοποιηθεί από δημόσια, ιδιωτικά ή υβριδικά δίκτυα και παρέχει μεγαλύτερη εμβέλεια από τα δίκτυα κινητής τηλεφωνίας.

Η τεχνολογία LoRa μπορεί εύκολα να συνδεθεί στην υπάρχουσα υποδομή επιτρέποντας την χαμηλή κατανάλωση μπαταρίας σε IoT εφαρμογές. Τα chipset LoRa της Semtech ενσωματώνονται σε συσκευές που κατασκευάζονται από ένα μεγάλο οικοσύστημα παρόχων IoT και συνδέονται με δίκτυα LoRaWAN σε όλο το κόσμο. Με άλλα λόγια, το LoRa συνδέει συσκευές (ή όλα) με το Cloud.

Τι είναι το LoRaWAN

Το LoRaWAN είναι ένα πρωτόκολλο δικτύωσης ευρείας περιοχής χαμηλής ισχύος (Σχ. 3.5) που βασίζεται στην τεχνολογία LoRa. Είναι σχεδιασμένο για να συνδέει ασύρματα πράγματα που λειτουργούν με μπαταρία στο διαδίκτυο, σε περιφερειακά, εθνικά ή παγκόσμια δίκτυα. Το πρωτόκολλο LoRaWAN αξιοποιεί το ραδιοφάσμα στη βιομηχανική, επιστημονική και ιατρική ζώνη (ISM). Η προδιαγραφή καθορίζει τη συσκευή-προς-υποδομή παραμέτρων φυσικού επιπέδου LoRa και το πρωτόκολλο LoRaWAN παρέχει απρόσκοπτη λειτουργικότητα μεταξύ συσκευών.



Σχήμα 3.5 «Δικτύωση ευρείας περιοχής LoRa» [15]

3.4 Αναλογικά και ψηφιακά φίλτρα

Στην επεξεργασία σήματος, η λειτουργία ενός φίλτρου είναι να απομακρύνει τα ανεπιθύμητα μέρη ενός σήματος όπως έναν τυχαίο θόρυβο ή να εξάγει χρήσιμα κομμάτια ενός σήματος, όπως οι συνιστώσες που βρίσκονται σε μια συγκεκριμένη περιοχή συχνοτήτων.

Υπάρχουν δύο κύρια είδη φίλτρων, τα αναλογικά και τα ψηφιακά. Είναι εντελώς διαφορετικά στην φυσική του δομή και στον τρόπο με τον οποίο λειτουργούν.

Σύγκριση και πλεονεκτήματα

Ένα αναλογικό φίλτρο χρησιμοποιεί ηλεκτρονικά κυκλώματα που αποτελούνται από συστατικά όπως αντιστάσεις, πυκνωτές και τελεστικοί ενισχυτές για να παραχθεί η απαιτούμενη επίδραση φίλτρου. Τέτοια κυκλώματα φίλτρων χρησιμοποιούνται ευρέως σε εφαρμογές όπως μείωση θορύβου, ενίσχυση σήματος video, γραφικοί equalizer σε hi-fi συστήματα, και σε πολλές άλλες περιοχές. Υπάρχουν καλά καθορισμένες standard τεχνικές για σχεδίαση κυκλώματος αναλογικού φίλτρου, για δοσμένες απαιτήσεις. Σε όλα τα στάδια, το σήμα που φιλτράρεται είναι μια ηλεκτρική τάση ή ένα ρεύμα το οποίο είναι ευθέως ανάλογο της φυσικής ποσότητας (π.χ. ένας ήχος ή ένα σήμα video ή η έξοδος ενός μετατροπέα) που μας ενδιαφέρει.

Το αναλογικό σήμα εισόδου πρέπει πρώτα να δειγματοληφθεί και να ψηφιοποιηθεί με τη χρήση ενός ADC (analog to digital converter) μετατροπέα από αναλογικό σε ψηφιακό. Οι δυαδικοί αριθμοί που προκύπτουν, οι οποίοι αναπαριστούν διαδοχικές τιμές από τη δειγματοληψία του σήματος εισόδου, μεταφέρονται στον επεξεργαστή, που εκτελεί αριθμητικές πράξεις σ' αυτούς. Αυτοί οι υπολογισμοί τυπικά περιέχουν πολλαπλασιασμούς των τιμών εισόδου με σταθερές και άθροιση των γινομένων. Αν είναι απαραίτητο, τα αποτελέσματα αυτών των υπολογισμών, που αναπαριστούν τιμές από δειγματοληψία του φιλτραρισμένου σήματος, γίνονται έξοδοι μέσω ενός DAC (digital to analog converter) μετατροπέα από ψηφιακό σε αναλογικό, για να μετατραπεί το σήμα και πάλι σε αναλογική μορφή.

Ένα ψηφιακό φίλτρο είναι σε θέση να προγραμματισθεί. Η λειτουργία του καθορίζεται από ένα πρόγραμμα στη μνήμη του επεξεργαστή. Αυτό σημαίνει ότι το ψηφιακό φίλτρο μπορεί να αλλάχθει εύκολα χωρίς να επηρεαστεί το κύκλωμα (hardware). Ένα αναλογικό φίλτρο, μπορεί μόνο να αλλάχθει ξανασχεδιάζοντας το κύκλωμα του φίλτρου. Τα ψηφιακά φίλτρα είναι εύκολα στη σχεδίαση, στη δοκιμή, και στην υλοποίηση σε ένα γενικού σκοπού υπολογιστή ή σε μια εγκατάσταση. Τα χαρακτηριστικά των σχεδίων των κυκλωμάτων ενός αναλογικού φίλτρου, (ιδιαίτερα αυτών που περιέχουν ενεργά συστατικά) εξαρτώνται από την τάση και βασίζονται στη θερμοκρασία. Τα ψηφιακά φίλτρα δεν μαστίζονται από τέτοια προβλήματα και γι' αυτό είναι ιδιαίτερα σταθερά ως προς τον χρόνο και τη θερμοκρασία.

Αντίθετα προς τα αναλογικά ισοδύναμα, τα ψηφιακά φίλτρα μπορούν να χειριστούν χαμηλής συχνότητας σήματα επακριβώς. Καθώς η ανάπτυξη της DSP τεχνολογίας συνεχίζει να αυξάνεται, τα ψηφιακά φίλτρα είναι σε θέση να εφαρμοστούν σε υψηλής συχνότητας σήματα στην RF (ράδιο συχνότητα) περιοχή, η οποία στο παρελθόν ήταν η αποκλειστικότητα της αναλογικής τεχνολογίας. Τα ψηφιακά φίλτρα είναι πολύ περισσότερο ευμετάβλητα στην ικανότητά τους να παράγουν σήματα με μια ποικιλία τρόπων: αυτό εμπεριέχει την ικανότητα μερικών τύπων ψηφιακών φίλτρων να προσαρμόζονται στις αλλαγές των χαρακτηριστικών του σήματος. Οι γρήγοροι DSP επεξεργαστές μπορούν να χειριστούν σύνθετους συνδυασμούς των φίλτρων παράλληλα ή σειριακά, κάνοντας τις απαιτήσεις του hardware σχετικά απλές σε σύγκριση με το ισοδύναμο αναλογικό σχέδιο του κυκλώματος.

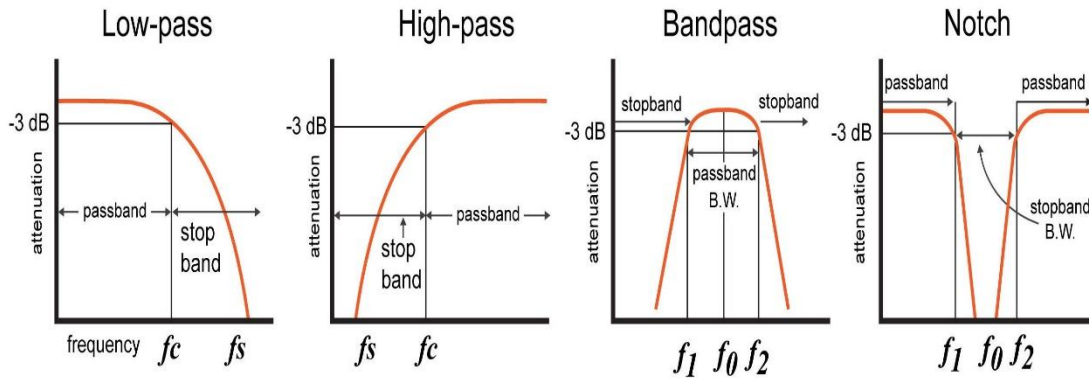
Είδη ψηφιακών φίλτρων

Τα φίλτρα μπορούν να ταξινομηθούν σε διάφορες ομάδες, ανάλογα με τα κριτήρια που χρησιμοποιούνται για την ταξινόμηση. Οι δύο κύριοι τύποι ψηφιακών φίλτρων είναι τα ψηφιακά φίλτρα πεπερασμένης απόκρισης (φίλτρα FIR) και τα ψηφιακά φίλτρα απεριόριστης ώθησης (IIR).

Και οι δύο τύποι έχουν κάποια πλεονεκτήματα και μειονεκτήματα που πρέπει να ληφθούν υπόψη προσεκτικά κατά το σχεδιασμό ενός φίλτρου. Εκτός αυτού, είναι απαραίτητο να ληφθούν υπόψη όλα τα θεμελιώδη χαρακτηριστικά ενός σήματος που πρέπει να φιλτράρεται, καθώς αυτά είναι πολύ σημαντικά όταν αποφασίζετε ποιο φίλτρο θα χρησιμοποιηθεί. Στις περισσότερες περιπτώσεις, είναι μόνο ένα χαρακτηριστικό που έχει πραγματικά σημασία και είναι εάν είναι απαραίτητο το φίλτρο να έχει γραμμική φάση ή όχι.

Το σήμα ομιλίας, για παράδειγμα, μπορεί να υποβληθεί σε επεξεργασία στα συστήματα με χαρακτηριστικό μη γραμμικής φάσης. Το χαρακτηριστικό φάσης ενός σήματος ομιλίας δεν είναι ουσιαστικό και ως εκ τούτου μπορεί να αγνοηθεί, πράγμα που έχει ως αποτέλεσμα τη δυνατότητα χρήσης πολύ ευρύτερου φάσματος συστημάτων για την επεξεργασία του.

Υπάρχουν επίσης σήματα για τα οποία το χαρακτηριστικό της φάσης είναι ουσιαστικό. Ένα τυπικό παράδειγμα είναι τα σήματα που λαμβάνονται από διάφορες αισθητήρες στη βιομηχανία. Επομένως, είναι απαραίτητο ένα φίλτρο να έχει γραμμική φάση χαρακτηριστικό για την αποφυγή απώλειας σημαντικών πληροφοριών.



Σχήμα 3.6 «Είδη των ψηφιακών φίλτρων» [16]

Όταν ένα σήμα προς φιλτράρισμα αναλύεται με αυτόν τον τρόπο, είναι εύκολο να αποφασίσετε ποιο είδος ψηφιακού φίλτρου είναι καλύτερο να χρησιμοποιήσετε. Κατά συνέπεια, εάν το χαρακτηριστικό φάσης είναι ουσιαστικό, θα πρέπει να χρησιμοποιούνται φίλτρα FIR καθώς έχουν γραμμικό χαρακτηριστικό φάσης. Αυτά τα φίλτρα είναι υψηλότερης τάξης και επομένως πιο περίπλοκα. Διαφορετικά, όταν έχει σημασία μόνο η απόκριση συχνότητας, είναι προτιμότερο να χρησιμοποιείτε ψηφιακά φίλτρα IIR τα οποία έχουν πολύ χαμηλότερη τάξη, δηλαδή είναι λιγότερο περίπλοκα, και επομένως πολύ πιο εύκολο να πραγματοποιηθούν.

Τα βασικά χαρακτηριστικά των φίλτρων πεπερασμένης απόκρισης (FIR) είναι:

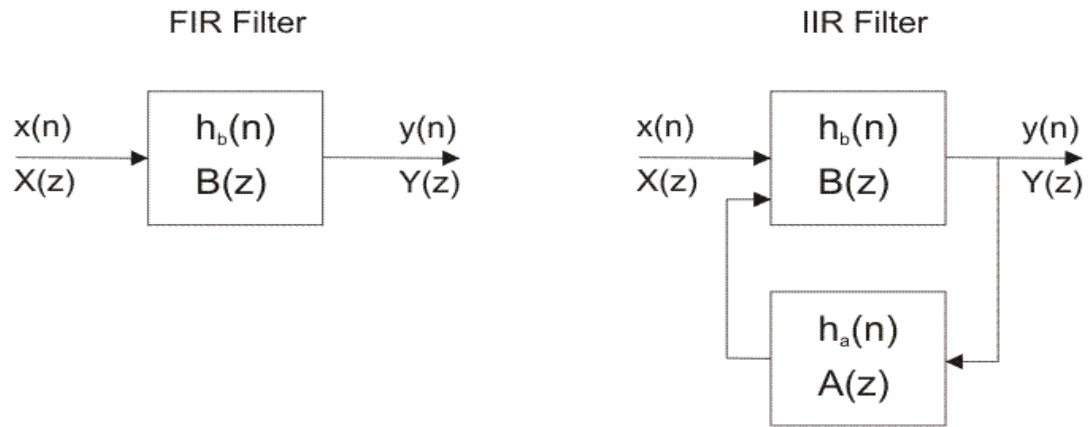
- Χαρακτηριστική γραμμική φάση.
- Υψηλή σειρά φίλτρων (πιο περίπλοκα κυκλώματα).
- Σταθερότητα.

Τα βασικά χαρακτηριστικά του Infinite Impulse Response (IIR) είναι:

- χαρακτηριστικό μη γραμμικής φάσης.
- χαμηλή σειρά φίλτρων (λιγότερο περίπλοκα κυκλώματα).
- Το προκύπτον ψηφιακό φίλτρο έχει τη δυνατότητα να γίνει ασταθές.

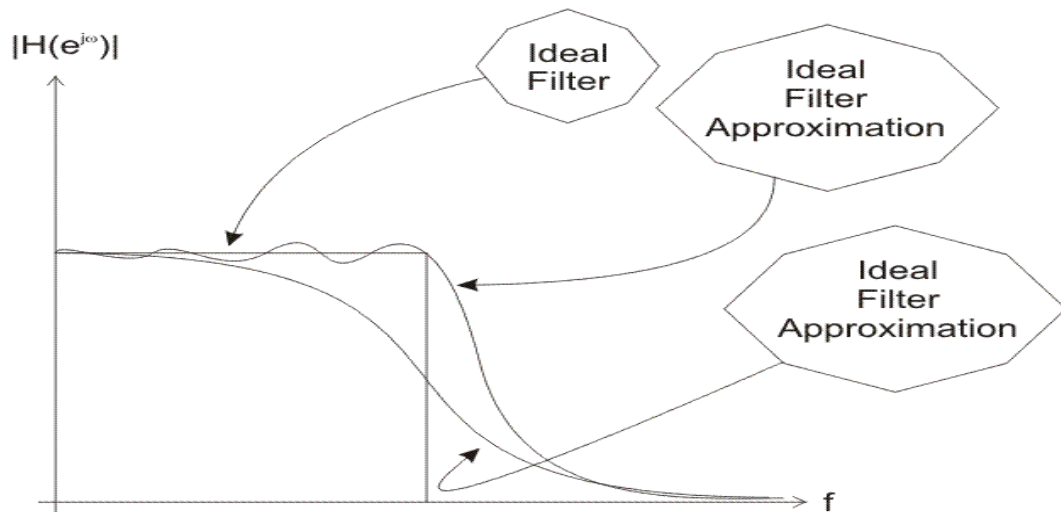
Τα FIR φίλτρα

Τα φίλτρα FIR είναι ψηφιακά φίλτρα πεπερασμένης κρουστικής απόκρισης. Είναι επίσης γνωστά ως μη αναδρομικά ψηφιακά φίλτρα καθώς δεν έχουν ανατροφοδότηση παρόλο που οι αλγόριθμοι μπορούν να χρησιμοποιηθούν για την πραγματοποίηση φίλτρων FIR[23].



Σχήμα 3.7 «Αποκλεισμός διαγραμμάτων φίλτρων FIR και IIR» [17]

Τα φίλτρα FIR μπορούν να σχεδιαστούν χρησιμοποιώντας διαφορετικές μεθόδους, αλλά τα περισσότερα βασίζονται στην ιδανική προσέγγιση φίλτρων. Ο στόχος δεν είναι να επιτευχθούν ιδανικά χαρακτηριστικά, καθώς είναι αδύνατο ούτως ή άλλως, αλλά να επιτευχθούν αρκετά καλά χαρακτηριστικά ενός φίλτρου. Η λειτουργία μεταφοράς του φίλτρου FIR πλησιάζει το ιδανικό καθώς αυξάνεται η σειρά φίλτρων, αυξάνοντας έτσι την πολυπλοκότητα και το χρόνο που απαιτείται για την επεξεργασία δειγμάτων εισόδου ενός σήματος που φιλτράρεται[23].



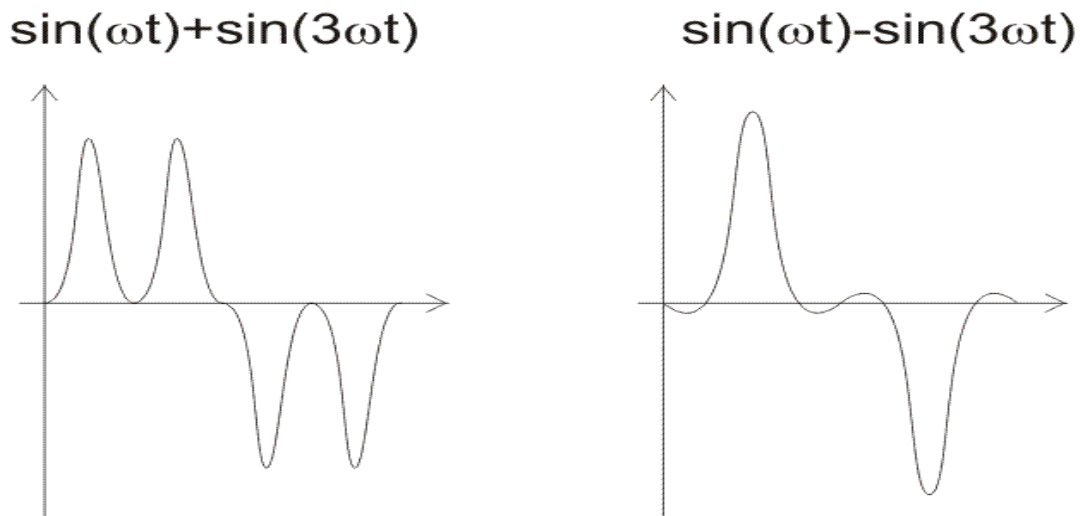
Σχήμα 3.8 «Ιδανική προσέγγιση φίλτρου χαμηλής διέλευσης» [17]

Η απόκριση συχνότητας που προκύπτει μπορεί να είναι μια μονοτονική συνάρτηση ή μια συνάρτηση ταλάντωσης εντός ενός συγκεκριμένου εύρους συχνοτήτων.

Η κυματομορφή της απόκρισης συχνότητας εξαρτάται από τη μέθοδο που χρησιμοποιείται στη διαδικασία σχεδιασμού καθώς και από τις παραμέτρους της.

Κάθε κατηγορία φίλτρου έχει πλεονεκτήματα και μειονεκτήματα. Αυτός είναι ο λόγος για τον οποίο είναι τόσο σημαντικό να επιλέξετε προσεκτικά την κατηγορία και τον τύπο ενός φίλτρου κατά τη διάρκεια της διαδικασίας σχεδιασμού.

Τα φίλτρα FIR μπορεί να έχουν χαρακτηριστικό γραμμικής φάσης, κάτι που δεν μοιάζει με φίλτρα IIR. Προφανώς, σε τέτοιες περιπτώσεις όταν είναι απαραίτητο να υπάρχει χαρακτηριστικό γραμμικής φάσης, τα φίλτρα FIR είναι η μόνη διαθέσιμη επιλογή. Εάν το χαρακτηριστικό γραμμικής φάσης δεν είναι απαραίτητο, όπως συμβαίνει με την επεξεργασία σημάτων ομιλίας, τα φίλτρα FIR δεν είναι καθόλου καλή λύση[24].



Σχήμα 3.9 «Επίδραση μη γραμμικού χαρακτηριστικού φάσης» [17]

Το σύστημα εισάγει μετατόπιση φάσης 0 ακτινίων στη συχνότητα ω και π ακτινίων τρεις φορές μεγαλύτερη από τη συχνότητα. Το σήμα εισόδου αποτελείται από φυσική συχνότητα ω και ένα αρμονικό με το ίδιο πλάτος τρεις φορές τη συχνότητα. Σχήμα 3.9 δείχνει το διάγραμμα μπλοκ του σήματος εισόδου (αριστερά) και του σήματος εξόδου (δεξιά). Είναι προφανές ότι αυτά τα δύο σήματα έχουν διαφορετικές κυματομορφές. Η ισχύς των σημάτων δεν αλλάζει, ούτε τα πλάτη των αρμονικών, αλλάζει μόνο η φάση της δεύτερης αρμονικής.

Εάν υποθέσουμε ότι η είσοδος είναι ένα σήμα ομιλίας του οποίου το χαρακτηριστικό φάσης δεν είναι ουσιαστικό, μια τέτοια παραμόρφωση στη φάση του σήματος θα ήταν ασήμαντη. Σε αυτήν την περίπτωση, το σύστημα πληροί όλες τις απαραίτητες απαιτήσεις. Ωστόσο, εάν το χαρακτηριστικό της φάσης είναι σημαντικό, δεν πρέπει να επιτρέπεται τόσο μεγάλη παραμόρφωση.

Προκειμένου το χαρακτηριστικό φάσης ενός φίλτρου FIR να είναι γραμμικό, η απόκριση ώθησης πρέπει να είναι συμμετρική ή αντισυμμετρική, η οποία εκφράζεται με τον ακόλουθο τρόπο:

$h[n] = h[N-n-1]$ συμμετρική παλμική απόκριση (για το μεσαίο στοιχείο).

$h[n] = -h[N-n-1]$ αντισυμμετρική παλμική απόκριση (για το μεσαίο του στοιχείο).

Ένα από τα μειονεκτήματα των φίλτρων FIR είναι η υψηλή σειρά σχεδιασμένων φίλτρων. Η σειρά του φίλτρου FIR είναι εξαιρετικά υψηλότερη σε σύγκριση με ένα φίλτρο IIR με την ίδια απόκριση συχνότητας. Αυτός είναι ο λόγος για τον οποίο είναι τόσο σημαντικό να χρησιμοποιείτε φίλτρα FIR μόνο όταν το χαρακτηριστικό γραμμικής φάσης είναι πολύ σημαντικό. Ένας αριθμός γραμμών καθυστέρησης που περιέχονται σε ένα φίλτρο, δηλαδή ένας αριθμός δειγμάτων εισόδου που πρέπει να αποθηκευτούν για τον υπολογισμό του δείγματος εξόδου, καθορίζει τη σειρά ενός φίλτρου. Για παράδειγμα, εάν το φίλτρο θεωρείται ότι είναι της τάξης 10, αυτό σημαίνει ότι είναι απαραίτητο να αποθηκεύσετε 10 δείγματα εισόδου πριν από το τρέχον δείγμα. Και τα έντεκα δείγματα θα επηρεάσουν το δείγμα εξόδου του φίλτρου FIR. Η συνάρτηση μετασχηματισμού ενός τυπικού φίλτρου FIR μπορεί να εκφραστεί ως πολώνυμο μιας σύνθετης μεταβλητής z^{-1} . Όλοι οι πόλοι της λειτουργίας μεταφοράς βρίσκονται στην αρχή. Για αυτόν τον λόγο, τα φίλτρα FIR είναι εγγυημένα ότι είναι σταθερά, ενώ τα φίλτρα IIR έχουν τη δυνατότητα να γίνουν ασταθή.

Σχεδιασμός ψηφιακού φίλτρου

Η έξοδος του αναλογικού κυκλώματος ρύθμισης σήματος συνδέεται με την ενσωματωμένη μονάδα ADC 12-bit του DSC. Γι αυτό το παράδειγμα, χρησιμοποιείται ένα dsPIC (DSC) από την τεχνολογία Microchip. Το dsPIC33FJ128GP802 που χρησιμοποιείται σε αυτόν τον σχεδιασμό επιτρέπει να εκμεταλλευτούμε όχι μόνο τις ενσωματωμένες δυνατότητες DSP, αλλά και το εργαλείο σχεδίασης ψηφιακών φίλτρων της Microchip[3].

Λαμβάνεται ένα δείγμα ADC κατά τη διάρκεια της χρονικής περιόδου κάθε LED και ένα δείγμα ADC λαμβάνεται κατά τη διάρκεια των περιόδων εκτός χρόνου και των δύο LED. Λόγω των προκλήσεων της λήψης μετρήσεων με βάση το φως μέσω οργανικού ιστού, το εργαλείο σχεδίασμού φίλτρου χρησιμοποιήθηκε για την εφαρμογή φίλτρου 513 τάξης, ψηφιακού-FIR, bandpass, το οποίο επέτρεψε να φιλτράρει τα δεδομένα ADC. Αυτά τα φιλτραρισμένα δεδομένα στη συνέχεια χρησιμοποιήθηκαν για τον υπολογισμό του πλάτους παλμού, όπως φαίνονται στο παρακάτω σχήμα.

Οι προδιαγραφές του φίλτρου ζώνης FIR είναι:

Συχνότητα δειγματοληψίας (Hz): 500

Συχνότητα Passband (Hz): 1 & 5

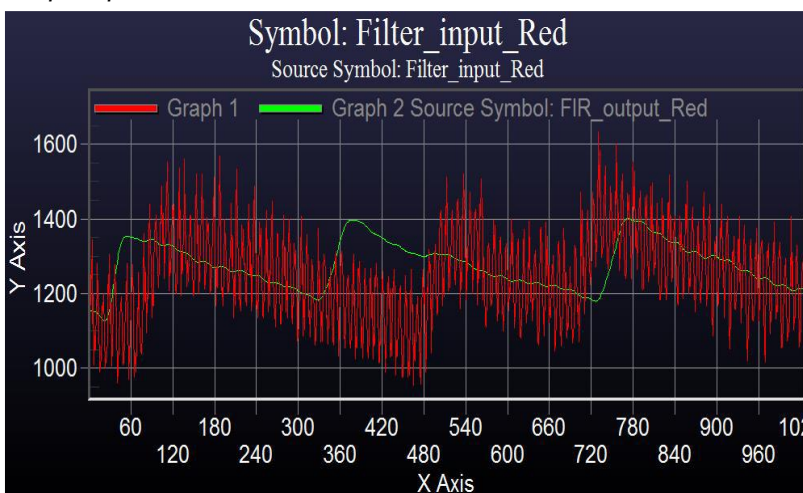
Συχνότητα Stopband (Hz): 0,05 & 25

Παράθυρο FIR: Kaiser

Ripple Passband (-dB): 0.1

Stopband Ripple (-dB): 50

Μήκος φίλτρου: 513



Σχήμα 3.10 Το γράφημα 1, που εμφανίζεται με κόκκινο χρώμα, είναι το σήμα εισόδου στο φίλτρο FIR.

Το γράφημα 2, που εμφανίζεται με πράσινο χρώμα, είναι το σήμα εξόδου από το φίλτρο FIR. Το X-Axis δείχνει τον αριθμό των δειγμάτων ADC. Το Y-Axis εμφανίζει τις τιμές ADC κώδικα. [2]

3.5 TTGO LoRa Esp32

Το βασικότερο εξάρτημα το οποίο μπορεί να χαρακτηριστεί και ως η καρδιά του παλμικού οξύμετρου είναι το ESP32 TTGO Lora. Είναι ένας προγραμματιζόμενος μικροελεγκτής χαμηλού κόστους και χαμηλής ισχύος με ενσωματωμένο Wi-Fi και Bluetooth και Bluetooth Low Energy (BLE). Το ESP32 δημιουργήθηκε και αναπτύχθηκε από την κινεζική εταιρία που εδρεύει στη Σαγκάη και κατασκευάζεται από την TSMC χρησιμοποιώντας τη διαδικασία των 40nm. Είναι διάδοχος του μικροελεγκτή ESP8266.



Σχήμα 3.11 «Μοντέλο Esp32 (TTGO LoRa V1) [18]

Η διαδικασία λιθογραφίας 40 νανομέτρων (40nm) είναι μία διαδικασία κατασκευής ημιαγωγών μισού κόμβου που χρησιμοποιείται ως διακόπτης μεταξύ των διαδικασιών 45nm και 32nm.[27] Η εμπορική ολοκληρωμένη κατασκευή κυκλωμάτων με τη διαδικασία 40nm ξεκίνησε το 2008 από κορυφαίες εταιρίες ημιαγωγών όπως η TSMC. Αυτή η τεχνολογία αντικαταστάθηκε από την εμπορική διαδικασία 32nm έως το 2010.

Οι διεργασίες 40nm General Purpose (GP) και Low Power (LP) διαθέτουν πυκνότητα ακατέργαστης πύλης που είναι 235% μεγαλύτερη από την τεχνολογία των 65nm. Το 40nm GP ξεπερνά το αντίστοιχο του 65nm έως και 40% κάτω από το ίδιο επίπεδο ρεύματος διαρροής και στο ήμισυ της κατανάλωσης ισχύος με την ίδια ταχύτητα λειτουργίας. Η διαδικασία 40nm LP μειώνει το ρεύμα διαρροής και την κατανάλωση ισχύος έως 51% σε σύγκριση με το αντίστοιχο 65nm στην ίδια ταχύτητα λειτουργίας.

Η τεχνολογία επεξεργασίας 40nm GP στοχεύει σε εφαρμογές υψηλής απόδοσης, συμπεριλαμβανομένων κεντρικών μονάδων επεξεργασίας (CPU), επεξεργαστών γραφικών, κονσόλων παιχνιδιών, δικτύων, FPGAs και σκληρών δίσκων. Οι βελτιωμένες διεργασίες LP 40nm και 40nm στοχεύουν σε smartphone, ψηφιακή τηλεόραση (DTV), αποκωδικοποιητή (Set-Top-Box), εφαρμογές παιχνιδιών και ασύρματες συνδέσεις. Η διαδικασία 40nm ULP είναι κατάλληλη για το διαδίκτυο των πραγμάτων και τις φορητές εφαρμογές[27].

Τεχνικά χαρακτηριστικά ESP32

- CPU: Xtensa 32-Bit LX6 Dual Core
- Clock: from 80 to 240 MHz
- ROM Memory: 448 KB
- SRAM Memory: 520 Kb
- Flash Memory: 32 MB
- Interface: SD, UART (3 ch), SPI (3 ch), SDIO, I2C (2 ch), I2S (2 ch), IR LED PWM (2 ch) and PWM (3 ch);
- GPIO: Digital IO (36), ADC 12 bit (16 ch), DAC 8 bit (2 ch), capacitive sensor (10 ch); LNA;
- Bluetooth:
 - 4.2 BR / EDR e BLE (Bluetooth Low Energy)
- Display OLED:
 - blue 0,96” SSD1306
- LoRa™:
 - Freq: 868 MHz
 - Modulation: FSK, GFSK, MSK, GMSK, LoRa TM, OOK
 - Sensibility: -139dBm
- Power: 3,3 V ÷ 7 V

Όπως προαναφέραμε στην κατασκευή μας χρησιμοποιήσαμε το ESP32 TTGO LORA όπου διαθέτει ενσωματωμένη υποστήριξη για επικοινωνία Wifi και Bluetooth, αλλά όχι για LoRa. Ένας πομποδέκτης SX127x ή RFM9x LoRa προσθέτει υποστήριξη για το LoRa και το πρωτόκολλο LoRaWAN που απαιτείται για το The Things Network.

Ο πομποδέκτης Semtech SX127x LoRa και οι πομποδέκτες HopeRF RFM9x LoRa είναι πανομοιότυποι. Έρχονται σε διαφορετικές παραλλαγές, ανάλογα με τη στοχευμένη ζώνη συχνοτήτων (433, 868 και 915 MHz). Ποιες συχνότητες χρησιμοποιούνται εξαρτάται από τη γεωγραφική περιοχή και τους τοπικούς κανονισμούς ζώνης ISM.

Οι πομποδέκτες ESP32 και LoRa διατίθενται σε διαφορετικές μορφές:

➤ Στις **ενότητες**.

Οι μονάδες περιέχουν πρόσθετα στοιχεία που απαιτούνται για τη λειτουργία των μακρών. Οι μονάδες χρησιμοποιούν απόσταση ακίδων 1,27 mm ή 2 mm. Για χρήση με πλακέτες και πρότυπα PCB με προσαρμογές αποστάσεων 2,54 mm.

➤ Σε **πίνακες ανάπτυξης**.

Οι πίνακες ανάπτυξης (συχνά) μετατρέπονται σε απόσταση 2,54 mm και προσθέτουν επιπλέον λειτουργίες, όπως μετατροπέα ισχύος, κουμπιά και LED. Οι πίνακες ανάπτυξης είτε τυπικές ενότητες είτε ξεχωριστά εξαρτήματα.

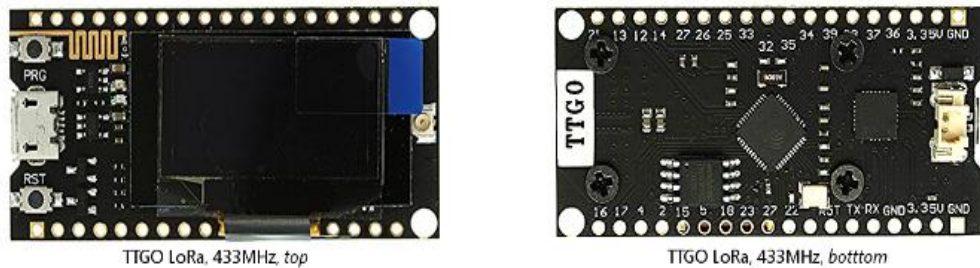
➤ Σε **προσαρμοσμένες πλακέτες** που συνδυάζουν ESP32 με πομποδέκτη LoRa.

Οι προσαρμοσμένες πλακέτες μπορούν να χρησιμοποιούν τυπικές ενότητες, ξεχωριστά εξαρτήματα ή συνδυασμό και των δύο.

Εκτός από το υλικό, απαιτείται λογισμικό για την εφαρμογή του πρωτοκόλλου LoRaWAN. Ο πομποδέκτης SX127x LoRa παρέχει διαμόρφωση ραδιοφώνου LoRa αλλά δεν εφαρμόζει το πρωτόκολλο LoRaWAN. Το πρωτόκολλο πρέπει να εφαρμοστεί σε λογισμικό που πρέπει να εκτελείται στο ESP32.

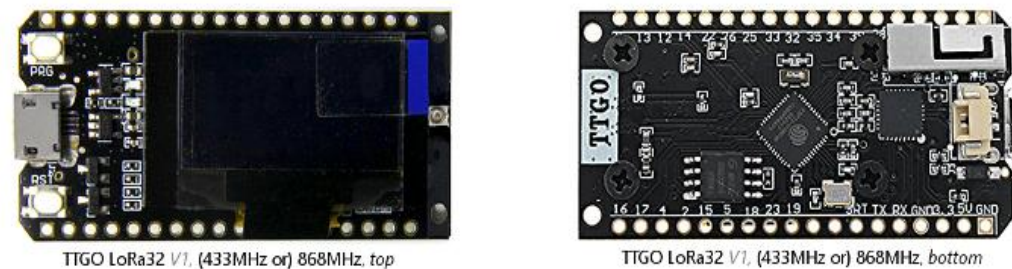
Διαφορετικά είδη TTGO LoRa32.

- LoRa (χωρίς 32): με ενσωματωμένη κεραία Wifi / Bluetooth PCB στην κορυφή. Φαίνεται να είναι διαθέσιμο μόνο για 433 MHz. Διατίθεται επίσης χωρίς οθόνη.



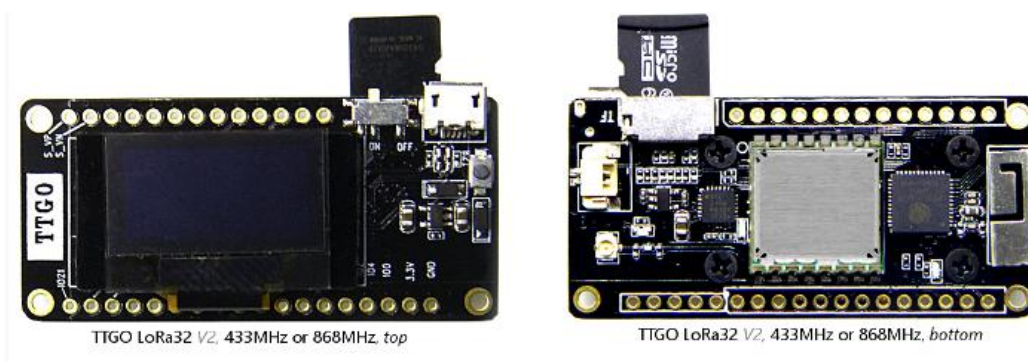
Σχήμα 3.12 «Μοντέλο LoRa» [19]

- LoRa32 V1: με ενσωματωμένη μεταλλική κεραία Wifi / Bluetooth στο κάτω μέρος. Ο σύνδεσμος I-Pex βρίσκεται στην κορυφή.



Σχήμα 3.13 «Μοντέλο LoRa32 V1» [19]

- LoRa32 V2: με ενσωματωμένη μεταλλική κεραία Wifi / Bluetooth στο κάτω μέρος (σε διαφορετική τοποθεσία). Χρησιμοποιεί ESP32-Pico-D4 (με ενσωματωμένη μνήμη flash) αντί για ESP32, χρησιμοποιεί μια (θωρακισμένη) μονάδα LoRa, υποδοχή I-Pex που βρίσκεται στο κάτω μέρος, η υποδοχή micro-USB περιστρέφεται 90 μοίρες. Επιπλέον διαθέτει υποδοχή κάρτας micro-SD στο κάτω μέρος και διακόπτη on/off για την μπαταρία δίπλα στην υποδοχή micro-USB. Σύμφωνα με το δελτίο δεδομένων ESP32-Pico-D43 μια κάρτα micro-SD είναι προσβάσιμη για κώδικα και δεδομένα, αλλά τα δεδομένα μπορούν να διαβαστούν μόνο, όχι να γραφτούν.



Σχήμα 3.14 «Μοντέλο LoRa32 V2» [19]

3.6 MAX30100

Ο αισθητήρας που θα χρησιμοποιήσουμε για να πάρουμε τις μετρήσεις των καρδιακών παλμών και του οξυγόνου στο αίμα είναι ο MAX30100 (Σχ. 3.15).



Σχήμα 3.15 «MAX30100» [20]

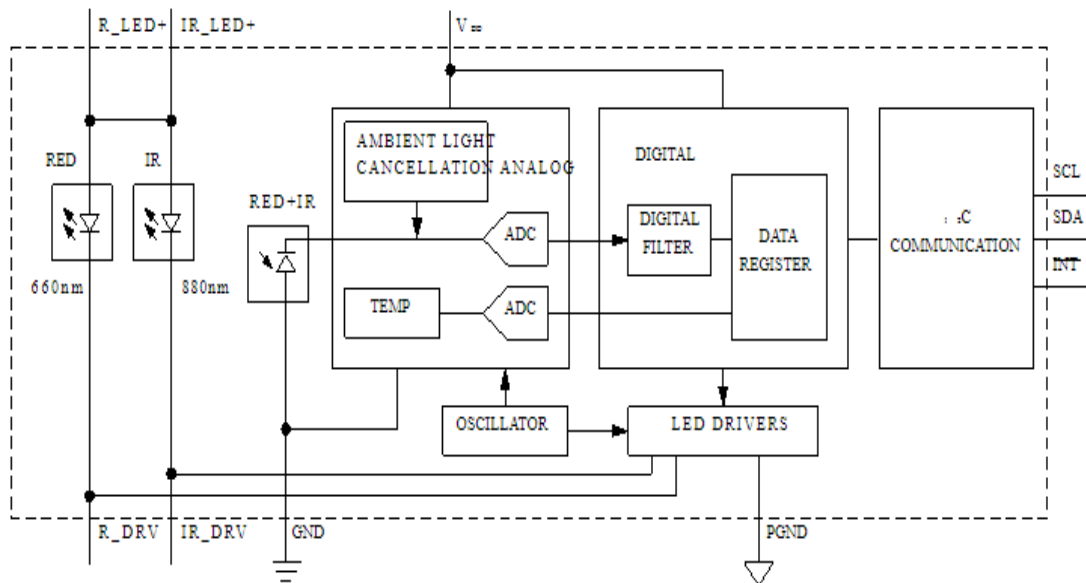
Το MAX30100 είναι ένα ολοκληρωμένο σύστημα αισθητήρα παλμού και καρδιακού ρυθμού σχεδιασμένο για απαιτήσεις φορητών συσκευών. Συνδυάζει δύο LED, έναν φωτοανιχνευτή, βελτιστοποιημένη οπτική και αναλογική επεξεργασία σήματος χαμηλού θορύβου για την ανίχνευση παλμικής οξυμετρίας και σήματα καρδιακού ρυθμού. Παρέχει πολύ μικρό μέγεθος χωρίς να θυσιάζει την οπτική ή ηλεκτρική απόδοση και απαιτούνται ελάχιστα εξωτερικά εξαρτήματα για την ενσωμάτωσή του σε μία φορητή συσκευή.

Βασικά χαρακτηριστικά

Μερικά βασικά χαρακτηριστικά του MAX30100 είναι:

1. Η I²C διεύθυνση του, είναι : 0x57
2. Τα δεδομένα αποθηκεύονται σε ένα FIFO Buffer όπου μπορούν να αποθηκευτούν έως και 16 μετρήσεις μεγέθους 4 byte η κάθε μέτρηση. Τα πρώτα 2 byte είναι για την IR μέτρηση και τα άλλα 2 για την κόκκινη.
3. Το FIFO buffer δεν μπορεί κατά συνέπεια να διαβαστεί με το I²C εφόσον το FIFO δείχνει την ίδια διεύθυνση. Πρέπει να ολοκληρωθούν η συναλλαγές της διεύθυνσης εξόδου το FIFO για να πάρει τις επόμενες μετρήσεις.
4. Έχει ενσωματωμένο φίλτρο 50/60Hz.
5. Για την ανίχνευση του παλμού απαιτείται μόνο το I.

6. Για τον κορεσμό οξυγόνου θα πρέπει να ενεργοποιηθούν και το IR LED Και το κόκκινο LED.
7. Αλλάζοντας τον ρυθμό δειγματοληψίας και το πλάτος παλμού των LED αλλάζει και η ADC ανάλυση. Είναι σημαντικό να σημειωθεί ότι ο ρυθμός δειγματοληψίας και το πλάτος παλμού είναι άμεσα συνδεδεμένα μεταξύ τους.



Σχήμα 3.16 «Διάγραμμα λειτουργίας του MAX30100» [21]

FIFO

Το FIFO είναι προσβάσιμο μέσω του I²C. Όλοι οι καταχωρητές που αφορούν το FIFO (0x02-0x05) έχουν μήκος 4 bit με εξαίρεση τα «FIFO Data» και αναφέρονται παρακάτω.

- 1) Δείκτης εγγραφής FIFO (0x02): Δείχνει την θέση όπου ο αισθητήρας θα γράψει το επόμενο δείγμα. Προχωρά αυτόματα με κάθε δείγμα να ωθείται στο FIFO.
- 2) Μετρήσεις υπερχίλησης FIFO (0x03): Όταν το FIFO είναι «γεμάτο» τα δείγματα δεν πιέζονται στο FIFO αλλά χάνονται. Το OFV-COUNTER μετρά τον αριθμό των δειγμάτων που έχουν χαθεί.
- 3) Δείκτης ανάγνωσης FIFO (0x04): Δείχνει την θέση όπου το ESP32 μπορεί να πάρει το επόμενο δείγμα από το FIFO μέσω του I²C. Ομοίως ο δείκτης εγγραφής FIFO προωθείται αυτόματα κάθε φορά που αποσύρεται ένα δείγμα από το FIFO.
- 4) Δεδομένα FIFO (0x05): Το FIFO Buffer έχει χωρητικότητα 64 bytes: 16 δείγματα των 4 byte (2 για κάθε δείγμα LED). Αυτός ο καταχωρητής δείχνει το κάθε επόμενο δείγμα που πρέπει να διαβαστεί από το FIFO. Σε αντίθεση με τα προηγούμενα, ο «FIFO Data» καταχωρητής δεν αυξάνει αυτόματα την διεύθυνση του καταχωρητή. Δεδομένου ότι κάθε δείγμα είναι 4 byte και το FIFO έχει μόνο 1 byte θα πρέπει να διαβαστεί 4 φορές για να πάρουμε ένα δείγμα (και να σιγουρευτούμε ότι δεν διαβάζει το ίδιο δείγμα δεύτερη φορά).

SrO2 υποσύστημα

Διαμόρφωση SrO2

Ο καταχωρητής (0x07) έχει τα ακόλουθα bits ελέγχου:

- 1) Ενεργοποίηση υψηλής ανάλυσης SrO2: Όταν το bit οριστεί σε κατάσταση High η ανάλυση ADC SrO2 είναι 16-bit και πλάτος παλμού 1600μs. Διαφορετικά η ανάλυση του και το πλάτος του ADC μπορεί οριστεί σε οποιαδήποτε τιμή.
- 2) Έλεγχος ρυθμού δείγματος SrO2: ορίζει τον πραγματικό ρυθμό δειγματοληψίας του IR όσο και του κόκκινου LED.
- 3) Έλεγχος πλάτους παλμού LED: αυτά τα δύο bit χρησιμοποιούνται για την επιλογή του πλάτους παλμού που σχετίζεται άμεσα με το μέγεθος δείγματος ADC.

Πίνακας 1. Πιθανές τιμές συχνότητας δειγματοληψίας MAX30100

SPO2-SR	Samples per Second
000	50
001	100
010	167
011	200
100	400
101	600
110	800
111	1000

Πίνακας 2. Διαμόρφωσης LED

Red-PA or IR-PA	Led Current
0000	0.0
0001	4.4
0010	7.6
0011	11.0
0100	14.2
0101	17.4
0110	20.8
0111	24.0
1000	27.1
1001	30.6
1010	33.8
1011	37.0
1100	40.2
1101	43.6
1110	46.8
1111	50.0

Αισθητήρας θερμοκρασίας

Το MAX30100 διαθέτει αισθητήρα θερμοκρασίας για (προαιρετική) βαθμονόμηση της θερμοκρασίας του υποσυστήματος SpO₂.

Ο αλγόριθμος SpO₂ είναι σχετικά ευαίσθητος στο μήκος κύματος των IR LED, αλλά το μήκος κύματος των κόκκινων LED είναι κρίσιμο για τη σωστή ερμηνεία των δεδομένων. Τα δεδομένα του αισθητήρα θερμοκρασίας μπορούν να χρησιμοποιηθούν για την αντιστάθμιση του σφάλματος του SpO₂ με αλλαγές θερμοκρασίας περιβάλλοντος.

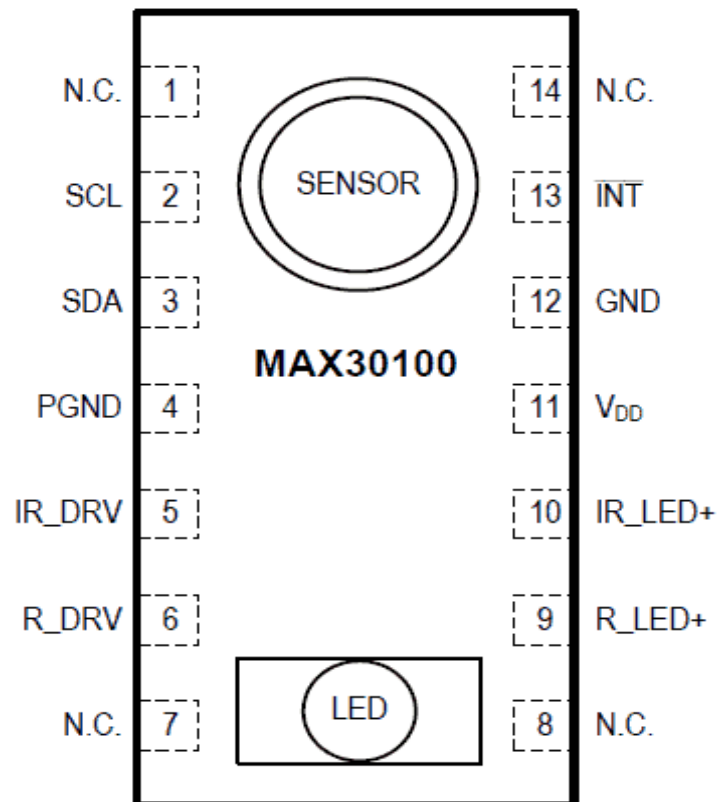
Οδηγός LED

Το MAX30100 ενσωματώνει κόκκινα και IR προγράμματα οδήγησης, για να οδήγησει παλμών LED για μετρήσεις SpO₂ και καρδιακών παλμών. Το ρεύμα LED μπορεί να προγραμματιστεί από 0mA έως 50mA (τυπικό μόνο) με κατάλληλη τάση τροφοδοσίας. Το πλάτος παλμού LED μπορεί να προγραμματιστεί από 200μs έως 1,6ms για τη βελτιστοποίηση της ακρίβειας της μέτρησης και της κατανάλωσης ισχύος με βάση τις περιπτώσεις χρήσης.

Pinouts

Πίνακας 3. Pinouts MAX30100

PIN	Ονομασία	Λειτουργία
1,7,8,14	N.C	Χωρίς σύνδεση. Συνδέεται στο PCB Pad για μηχανική σταθερότητα.
2	SDL	Είσοδος χρονοδιακόπτη I2C .
3	SDA	Δεδομένα χρονοδιακόπτη I2C, αμφίδρομη (Open-Drain).
4	PGND	Γείωση ισχύος του μπλοκ οδηγού LED.
5	IR_DRV	Κάθοδος IRLED και σημείο σύνδεσης οδηγού LED.
6	R_DRV	Κάθοδος κόκκινου LED και σημείο σύνδεσης οδηγού LED.
9	R_LED+	Τροφοδοσία (σύνδεση ανόδου) για το κόκκινο LED. Παρακάμπτει το PGND για καλύτερη απόδοση. Είναι εσωτερικά συνδεδεμένο με το IR_LED+.
10	IR_LED+	Τροφοδοσία (σύνδεση ανόδου) για IR_LED. Παρακάμπτει το PGND για καλύτερη απόδοση. Είναι εσωτερικά συνδεδεμένο με το R_LED+.
11	V DD	Αναλογική είσοδος τροφοδοσίας. Παρακάμπτει το GND για καλύτερη απόδοση.
12	GND	Αναλογική γείωση.
13	INT	Ενεργεί χαμηλή διακοπή (Open-Drain).



Σχήμα 3.17 «Pinouts του MAX30100» [22]

3.7 Επίλογος

Αυτό το κεφάλαιο παρουσίασε μια περιγραφή του αισθητήρα MAX 30100. Επιπλέον, το Κεφάλαιο αξιολόγησε πώς παρέχει ο αισθητήρας δεδομένα, πώς μπορεί να συνδεθεί με το ESP και πώς πρέπει να επεξεργαστεί το σήμα με τη σειρά για εξαγωγή τιμών HR και SpO2. Έγινε επίσης μια ανάλυση λειτουργίας των Bluetooth, πώς φτάσαμε στο BLE (Bluetooth Low Energy), τι είναι το LoRa και πώς μπορούμε να χρησιμοποιούμε. Αναλύσαμε τα τεχνικά χαρακτηριστικά του ESP32 και κάναμε μια μικρή μελέτη για τα ψηφιακά φίλτρα.

Κεφάλαιο 4^ο : Κατασκευή

4.1 Εισαγωγή

Το τελευταίο κομμάτι που θα ασχοληθούμε είναι η κατασκευή. Πως δηλαδή έγινε βήμα-βήμα η συνδεσμολογία, πως γράφτηκε ο κώδικας έτσι ώστε να δώσουμε στο Esp32 ένα δικό του μενού για να διευκολύνουμε την χρήση του και να μπορούμε εύκολα να το συνδέσουμε με συσκευές που υποστηρίζουν Bluetooth και έχουν δική τους οθόνη ώστε να βλέπουμε τα αποτελέσματα των μετρήσεων εκεί.

4.2 Πρώτα βήματα

Πίνακας 4 Κοστολόγιο υλικών

Ποσότητα	Υλικά	Κατάστημα	Τιμή
2	TTGO LORA32 868/915Mhz SX1276 ESP32 Oled-display Bluetooth WIFI Lora	TSMC	49,40 €
2	MAX30100 HEART-RATE OXIMETER PULSE SENSOR MODULE	Maxim Integrated	10,25 €
1	PCB διάτρητη πλακέτα	Roth Electronik	6,00 €
4	12x12x6mm Momentary Tactile Push Button Switch for Arduino	Wenzhou Gangyuan Electronics Co, Ltd	4,50 €
1 μέτρο	Καλώδιο	-	Αμελητέο
1	Αντίσταση 10kΩ	-	Αμελητέο
1	USB 2.0 to micro USB	HUA YIN	1,00 €
1	Arduino Software (IDE)	https://www.arduino.cc/en/main/software	Δωρεάν

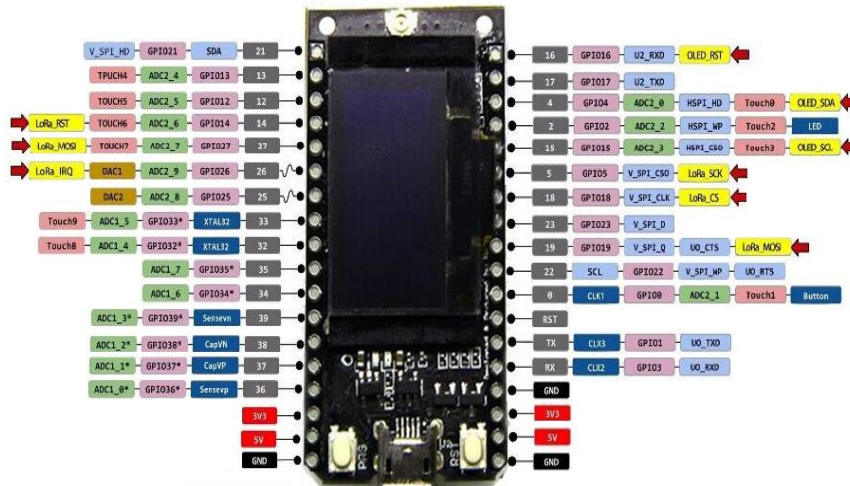
Τα υλικά τα οποία θα χρησιμοποιηθούν για την υλοποίηση είναι:

1. Ένας προγραμματιζόμενος μικροελεγκτής TTGO ESP32 LoRa V1.
2. Έναν αισθητήρα MAX30100.
3. Βραχυκλωτήρες.
4. Καλώδιο σύνδεσης ESP32 με υπολογιστή για τη φόρτωση του προγράμματος.

Κεφάλαιο 4°

5. Το πρόγραμμα Arduino Studio για την υλοποίηση του προγράμματος.

Ξεκινήσαμε από τα datasheet του TTGO ESP32 LoRa V1 και βρήκαμε που αντιστοιχούν οι εξόδοι και οι εισόδοι του.

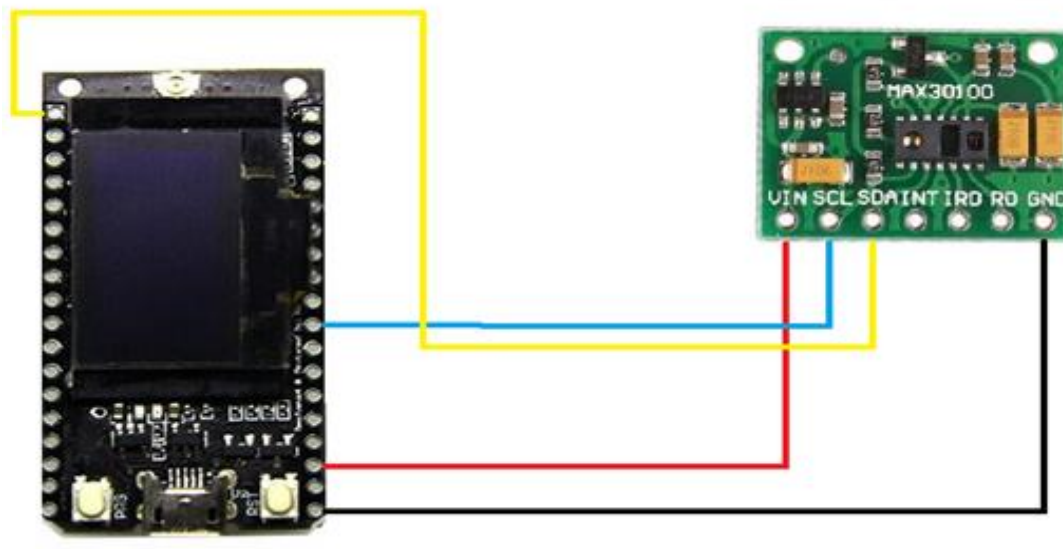


Σχήμα 4.1«Pinouts του ESP32»

Το δεύτερο βήμα ήταν να βρούμε στα datasheet του MAX30100 τις εισόδους και εξόδους που θα χρησιμοποιήσουμε.

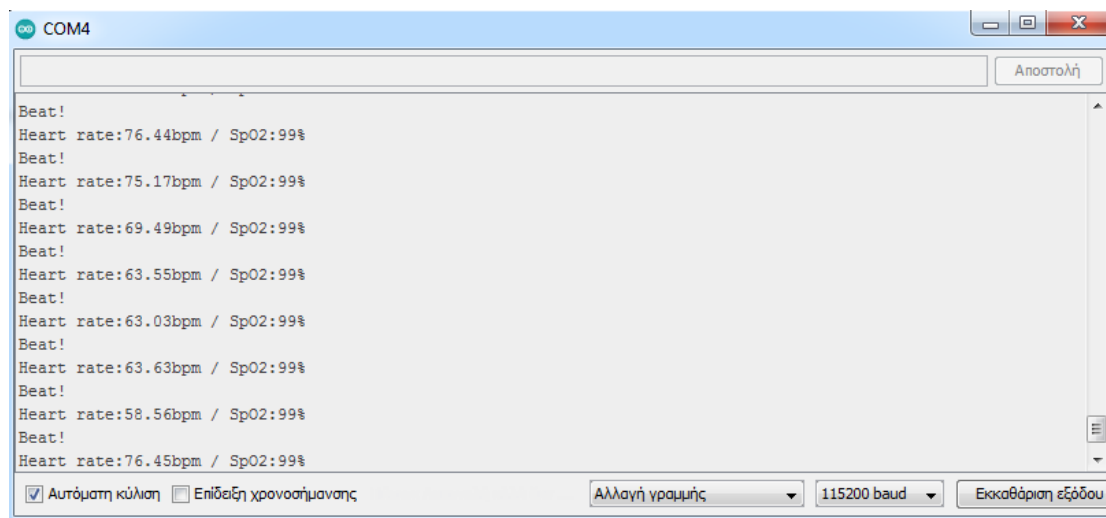
Συνδέσαμε τον αισθητήρα στο ESP32 στα ποδαράκια:

MAX30100 ESP32
SDA → SDA pin21
SCL → SCL pin22
Vin → Vin 3,3V
GND → GND



Σχήμα 4.2 «Σύνδεση ESP32 με το MAX30100»

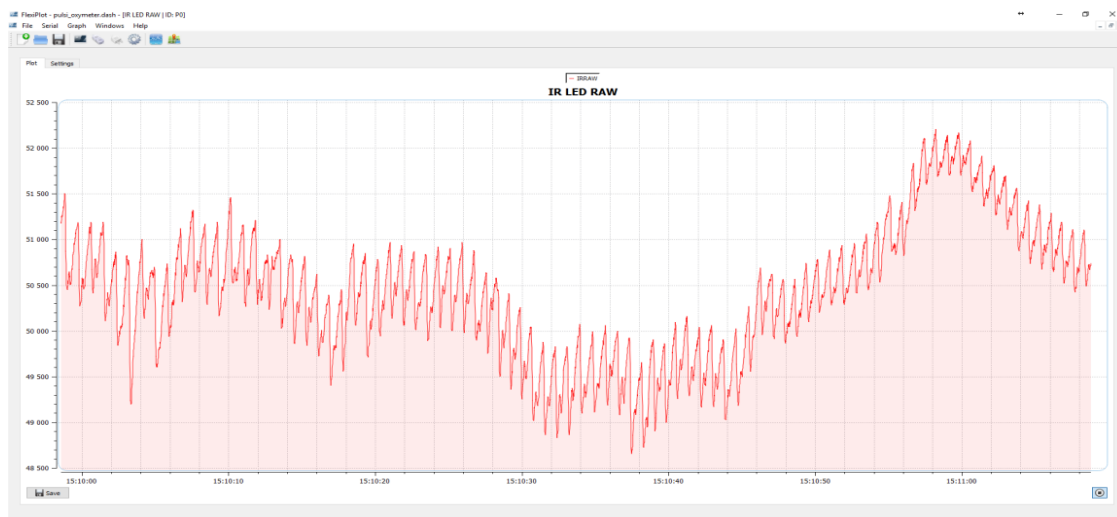
Περάσαμε τον αρχικό κώδικα με τον οποίο μετράμε τους καρδιακούς παλμούς και τον κορεσμό του οξυγόνου στο αίμα χωρίς να χρησιμοποιήσουμε την οθόνη του ESP32 και βλέπαμε τα αποτελέσματα των μετρήσεων στο Serial Monitoring.



Σχήμα 4.3 «Μετρήσεις στο Serial Monitoring»

Λόγο των αποκλίσεων στους καρδιακούς παλμούς κυματομορφή έχει την εξής μορφή:

Κεφάλαιο 4°



Σχήμα 4.4 «Κυματομορφή των μετρήσεων»

Επομένως για να πάρουμε τις σωστές τιμές των καρδιακών παλμών θα χρησιμοποιήσουμε τρία φίλτρα:

- Αφαίρεσης DC σήματος.
- Μέσου όρου.
- Butterworth.

Αφαίρεση DC σήματος

Αυτό που παρατηρούμε στην πρώτη κυματομορφή είναι:

- Η κυματομορφή ταλαντώνεται ελαφρώς.
- Έχει μετατόπιση περίπου 50000 μονάδες.

Για την επίλυση οι εξισώσεις που θα πρέπει να χρησιμοποιηθούν είναι οι εξής.

$$w(t) = x(t) + a \cdot w(t-1)$$

$$y(t) = w(t) - w(t-1)$$

$y(t)$: έξοδος φίλτρου.

$x(t)$: τιμή ρεύματος εισόδου.

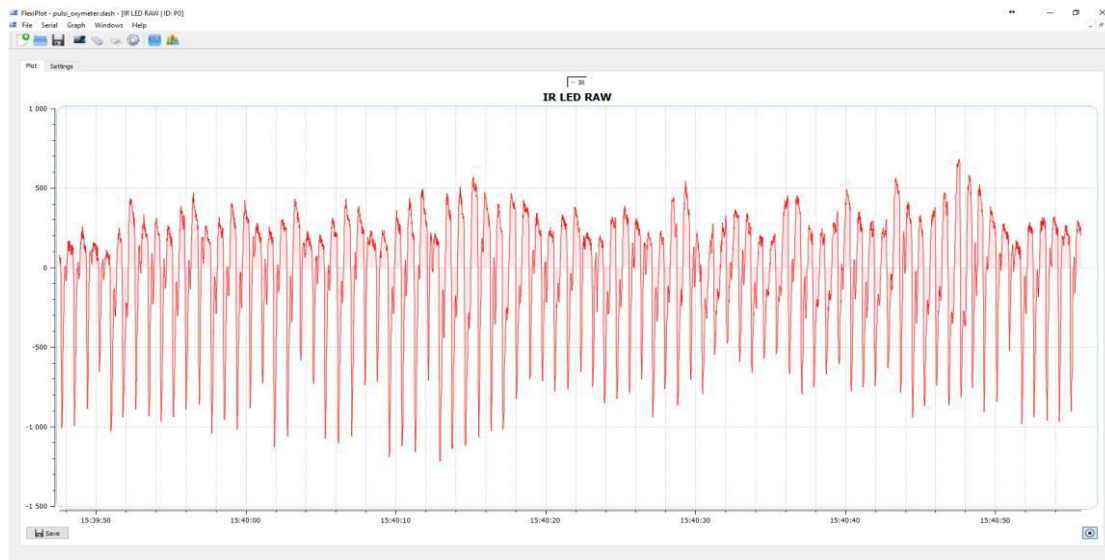
$w(t)$: τιμή DC σήματος.

a : σταθερά απόκρισης φίλτρου.

- Αν το $a=1$ τότε περνάει όλο το σήμα.
- Αν το $a=0$ δεν περνάει τίποτα.

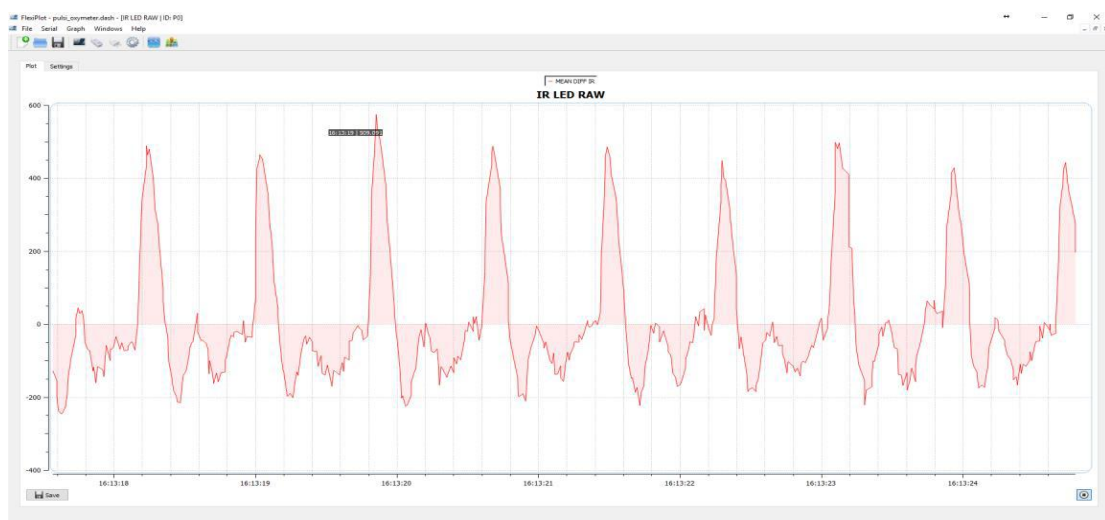
Άρα για να κόψουμε το DC σήμα χωρίς να επηρεάσουμε τα αποτελέσματα των μετρήσεων αλλά να καθαρίσει το σήμα μας θα πρέπει το a να είναι όσο το δυνατόν πιο κοντά στο 1. Επομένως $a=0,95$.

Φίλτρο μέσου όρου



Σχήμα 4.5 «Κυματομορφή μετρήσεων μετά από αφαίρεση DC σήματος»

Στις μετρήσεις παρατηρήσαμε κάποιο απότομο ανεβοκατέβασμα (πχ από 74 παλμούς στους 50 κλπ). Για να μη συμβαίνει αυτό θα πρέπει να πάρουμε τη διαφορά του σήματος διότι ο παλμός που θέλουμε είναι εκεί που γίνεται η απότομη αλλαγή τιμής, οπότε δημιουργήσαμε το παρακάτω φίλτρο και είδαμε ότι η νέα έξοδος σήματος έχει τη μορφή:



Σχήμα 4.6 «Τελική κυματομορφή μετρήσεων»

Φίλτρο Butterworth

Όπως είδαμε στο σήμα του προηγούμενου φίλτρου η κυματομορφή είναι αρκετά καθαρή και μπορούμε να βρούμε τους καρδιακούς παλμούς, αλλά για να βελτιώσουμε ακόμα περισσότερο την ακρίβεια θα πρέπει να φιλτράρουμε τις υψηλότερες κορυφές του σήματος. Για να το πετύχουμε αυτό θα χρησιμοποιήσουμε ένα low pass.

Για να το καταφέρουμε αυτό πρέπει να καθορίσουμε ρυθμό δειγματοληψίας (F_s) και συχνότητα αποκοπής (F_c).

Κεφάλαιο 4°

Ο ταχύτερος ρυθμός δειγματοληψίας για τον MAX30100 είναι 1kHz, ωστόσο εμείς επιλέξαμε τα 100Hz (συνίσταται από την βιβλιοθήκη του MAX30100). Στη συνέχεια θα πρέπει να επιλέξουμε συχνότητα αποκοπής. Η επιλογή μας προήλθε από τους 220 παλμούς, αν και είναι επικίνδυνα υψηλό επίπεδο συμβαίνει σε ορισμένες περιπτώσεις και 50 παλμούς σαν χαμηλότερη τιμή.

$$\frac{220 \text{ BPM}}{60} = 3,66\text{Hz}$$

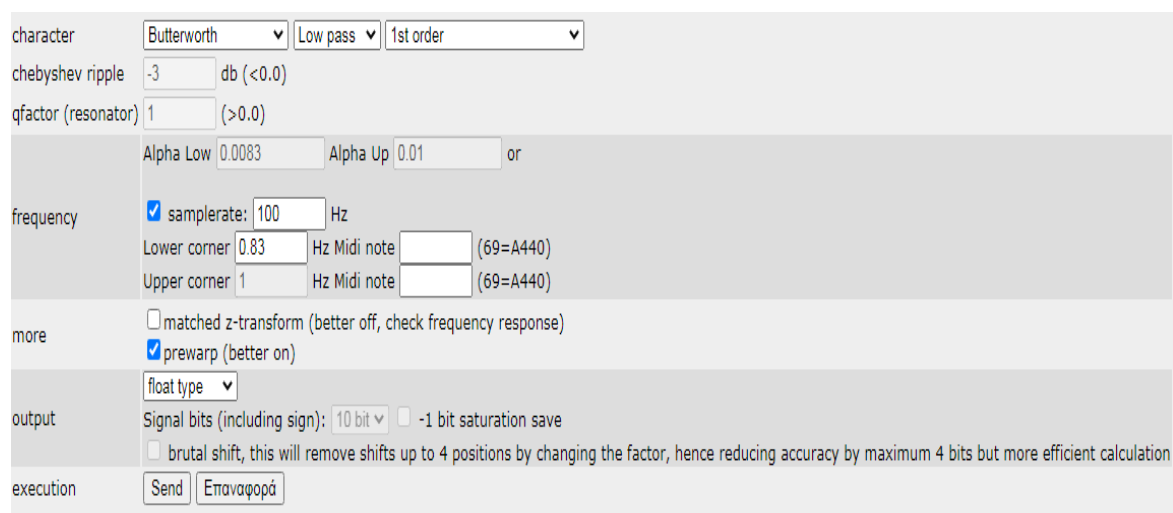
&

$$\frac{50 \text{ BPM}}{60} = 0,83\text{Hz}$$

Η συχνότητα αποκοπής θα είναι τα 4Hz.

Το φίλτρο Butterworth το δημιουργήσαμε online στην ιστοσελίδα : <http://www.schwietering.com/jayduino/filtuino/>

Επιλέξαμε Butterworth filter low pass 1st order, ορίσαμε το ρυθμό δειγματοληψίας στα 100Hz και την χαμηλότερη συχνότητα στα 0,83Hz.



Σχήμα 4.7 «Ρυθμίσεις Butterworth φίλτρου»

Ο κώδικας που βγαίνει με τα συγκεκριμένα δεδομένα μας είναι:

```
//Low pass butterworth filter order=1 alpha1=0.0083
class FilterBuLp1
{
    public:
        FilterBuLp1()
        {
            v[0]=0.0;
        }
    private:
        float v[2];
    public:
        float step(float x) //class II
```

```

    {
        v[0] = v[1];
        v[1] = (2.541819503781362136e-2 * x)
              + (0.94916360992437276423 * v[0]);
        return
            (v[0] + v[1]);
    }
};

```

Το επόμενο βήμα ήταν να τροποποιήσουμε τον κώδικα και το κύκλωμά μας έτσι ώστε να εμφανίζονται οι μετρήσεις BPM και SpO2 στην οθόνη του ESP32.

MAX30100		ESP32
SDA	→	OLED_SDA pin4
SCL	→	OLED_SCL pin15
Vin	→	Vin 3,3V
GND	→	GND

Τον αισθητήρα τον συνδέσαμε στα ποδαράκια OLED_SDA και OLE_DSCL με τα οποία επικοινωνεί το OLED display με το ESP32 μέσω I2C και SPI με την χρήση των βιβλιοθηκών:

```

#include <SPI.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

```

Διάλογος I2C

Ο διάλογος I2C είναι ένας σειριακός διάλογος που χρησιμοποιήθηκε από την Philips στις αρχές της δεκαετίας του 1980 με σκοπό να παρέχει έναν εύκολο τρόπο σύνδεσης μίας CPU με περιφερειακά τσιπ σε μία τηλεόραση. Πλέον χρησιμοποιείται και για την σύνδεση ενσωματωμένων συστημάτων, κινητά τηλέφωνα και άλλες ηλεκτρονικές συσκευές. Το BUS αποτελείται από δύο ενεργά καλώδια και μία γείωση. Τα ενεργά καλώδια, SDA (Serial Data) και SCL (Serial Clock), είναι αμφίδρομα. Το βασικό πλεονέκτημα είναι ότι απαιτούνται μόνο δύο γραμμές για την αμφίδρομη επικοινωνία μεταξύ πολλών συσκευών. Η διεπαφή συνήθως λειτουργεί με χαμηλή ταχύτητα (100kHz-400kHz). Μία συσκευή που ελέγχει και οδηγεί τη γραμμή ρολογιού SCL είναι η Κύρια (Master) και οι συσκευές υποτελείς (Slave), είναι αυτές που ανταποκρίνονται στις συσκευές Master οι οποίες δεν μπορούν να ξεκινήσουν μία μεταφορά δεδομένων πάνω στον διάλο (μόνο οι συσκευές Master μπορούν). Όταν μία συσκευή Master επικοινωνεί με μία Slave, αρχίζει να στέλνει στον διάλο μία ακολουθία έναρξης (start sequence). Η άλλη ακολουθία που ορίζεται στο διάλο I2C είναι η ακολουθία λήξης (stop sequence). Αυτές οι δύο ακολουθίες διαφέρουν στο ότι ενώ η γραμμή ρολογιού (SCL) είναι σε κατάσταση λογικού 1 μόνο αυτές μπορούν να αλλάξουν τη γραμμή δεδομένων (SDA).

SPI

Το SPI (Serial Peripheral Interface) είναι μία προδιαγραφή διασύνδεσης σειριακής επικοινωνίας για μικρές αποστάσεις, κυρίως σε ενσωματωμένα συστήματα.

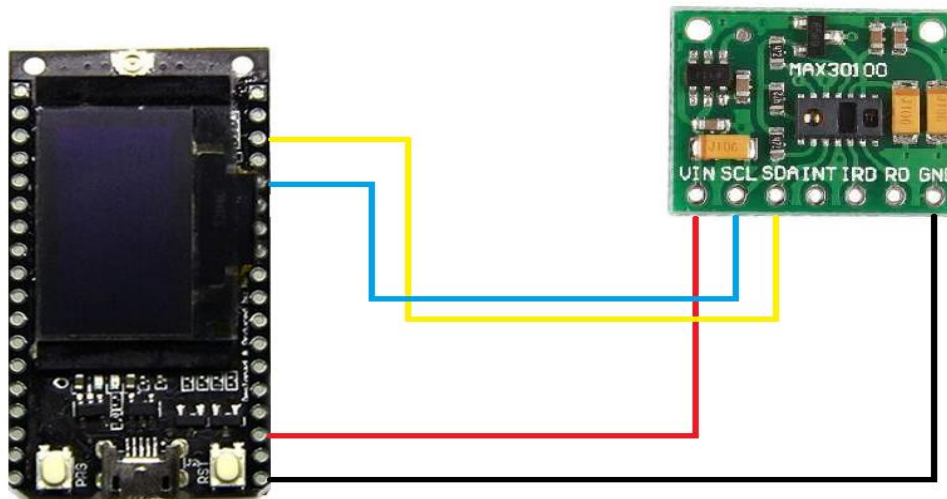
Κεφάλαιο 4°

Αυτή η διεπαφή αναπτύχθηκε στα μέσα της δεκαετίας του 1980 (λίγο μετά τον διάυλο I2C) από την Motorola. Οι εφαρμογές του περιλαμβάνουν οθόνες υγρών κρυστάλλων.

Ο τρόπος επικοινωνίας είναι αμφίδρομη με τη χρήση Master και Slave (με μόνο ένα Master).

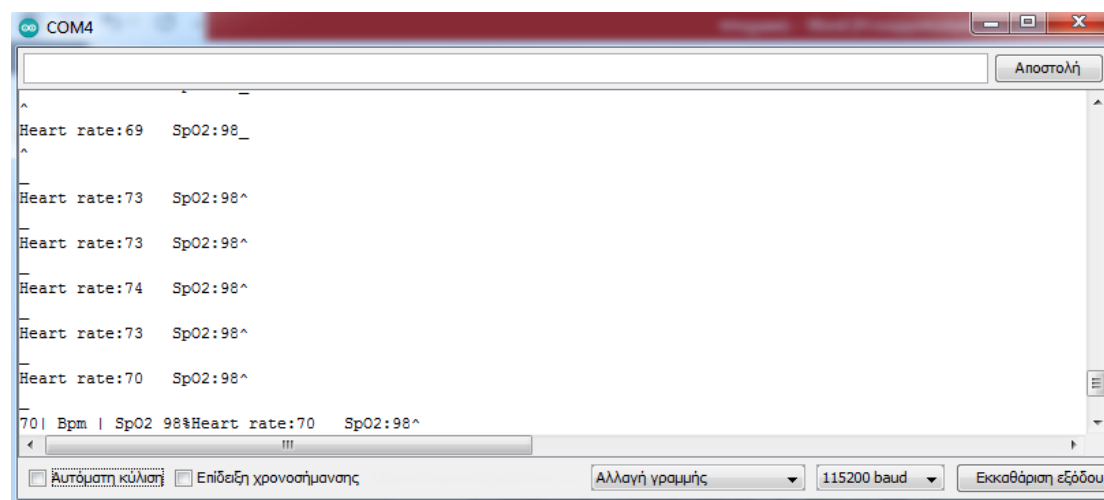
Οι εξαρτημένες συσκευές υποστηρίζονται μέσω επιλογής με μεμονωμένες εξαρτημένες συσκευές (SS), που μερικές φορές ονομάζονται chip select (SC).

Το νέο κύκλωμά μας είναι:



Σχήμα 4.8 «Σύνδεση MAX30100 με ESP32 και Display»

Τα αποτελέσματα που είχαμε με αυτές τις αλλαγές ήταν:



Σχήμα 4.9 «Μετρήσεις στο Serial Monitoring »

Οι τιμές είναι πιο σταθερές και στο τέλος εμφανίζεται η σωστή μέτρηση που οφείλεται στη χρήση του φίλτρου μέσου όρου.



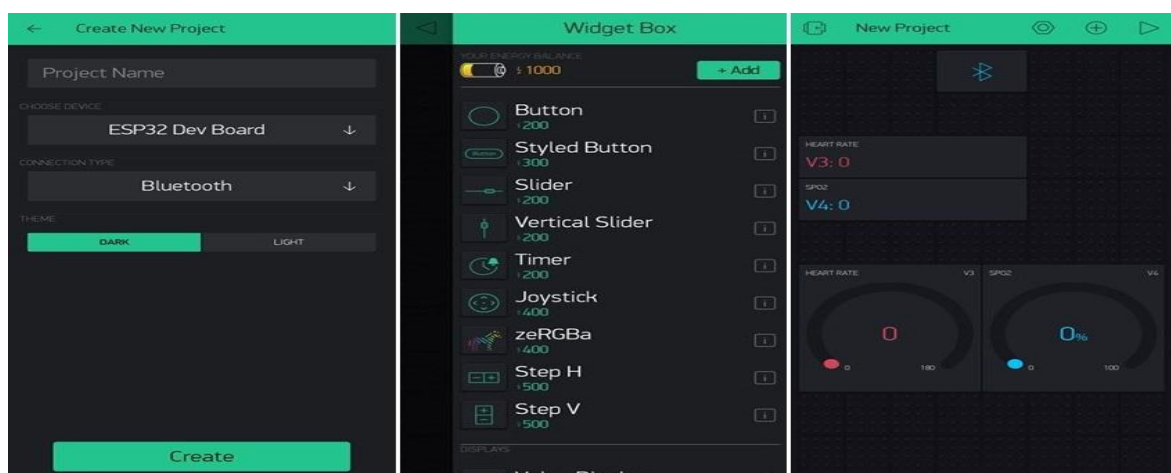
Σχήμα 4.10 «Σύγκριση μετρήσεων με Smartwatch»

Στην παραπάνω εικόνα απεικονίζονται οι μετρήσεις πάνω στο display του ESP32 και μία μέτρηση με smartwach για επιβεβαίωση ότι οι παλμοί είναι σωστοί.

Στη συνέχεια σκοπός μας είναι να συνδέσουμε το ESP32 με Bluetooth έτσι ώστε να συνδέεται με κινητό τηλέφωνο και να στέλνει τις μετρήσεις των καρδιακών παλμών και τον κορεσμό οξυγόνου στο αίμα.

4.3 Υλοποίηση της εφαρμογής με Blynk

Αρχικά βρήκαμε το Blynk, μια εφαρμογή στο Play Store (για Android λογισμικό) στην οποία μπορέσαμε να αναπτύξουμε ένα μενού στο οποίο μετά την σύνδεση του κινητού με το ESP32 θα εμφανίζονται τα αποτελέσματα των μετρήσεών μας.

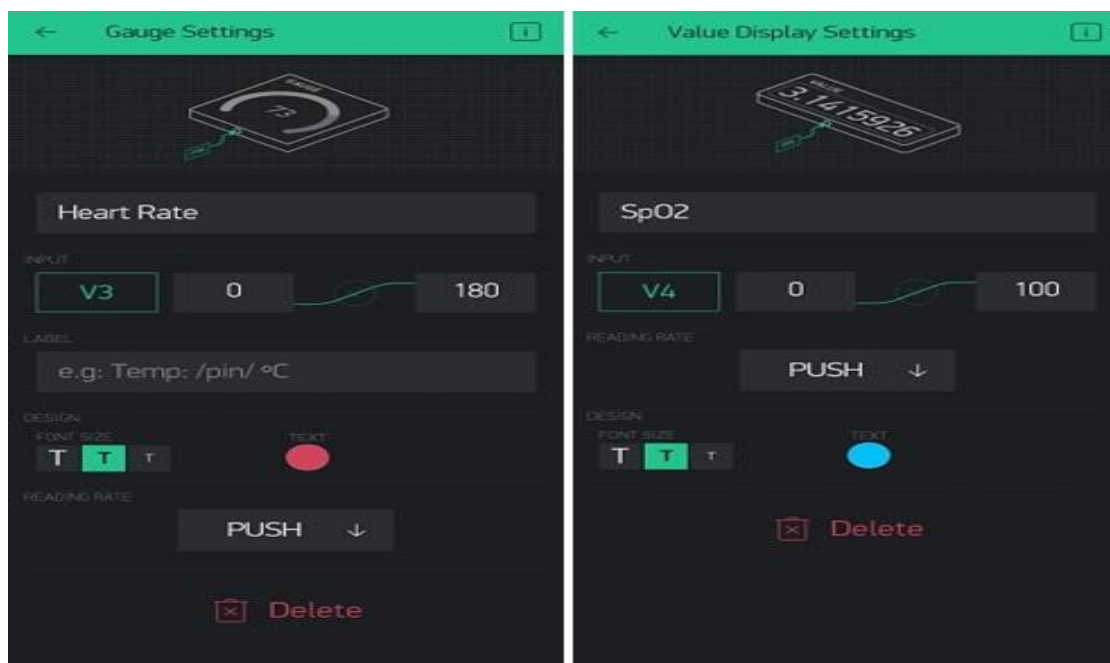


Σχήμα 4.11 «Ανάπτυξη εφαρμογής μέσω Blynk»

Αφού συνδεθήκαμε στην εφαρμογή δημιουργήσαμε ένα νέο project το οποίο έχει την δυνατότητα να επικοινωνήσει με ESP32 Dev Board δηλαδή με την πλακέτα μας (ESP32) και ως τρόπο επικοινωνίας διαλέξαμε τα Bluetooth. Στη συνέχεια από το μενού του επιλέξαμε

Κεφάλαιο 4^ο

εικονίδιο για την εύρεση της πλακέτας μέσω Bluetooth και πως θέλουμε να εμφανίζονται οι τιμές που θα στέλνουμε. Δύο Gauge (με μορφή παλμομέτρου), δύο Value Display (με μορφή οθόνης) και ορίσαμε τις μεταβλητές με τις οποίες θα στέλνονται τα δεδομένα από το ESP.



Σχήμα 4.12 «Παλμόμετρο (αριστερά) και Display (δεξιά) στο Blynk»

Το τελευταίο βήμα για την εφαρμογή είναι να την δηλώσουμε μέσα στον κώδικά μας. Αυτό το καταφέραμε με τις παρακάτω εντολές.

Αρχικά ορίσαμε τις βιβλιοθήκες και τις μεταβλητές που ανακαλεί η εφαρμογή.

```
#include <BlynkSimpleEsp32_BT.h>
#define BLYNK_PRINT Serial
#define BLYNK_USE_DIRECT_CONNECT
```

Με την δημιουργία της εφαρμογής στάλθηκε στο mail που χρησιμοποιήσαμε το οποίο περιέχει ένα σύνδεσμο ο οποίος απαιτείται από τους κατασκευαστές της εφαρμογής για τη σύνδεση της κατασκευής μας με κάποιο κινητό τηλέφωνο μέσω Bluetooth, τον οποίο καταχωρίσαμε στον κώδικα.

```
char auth[] = "_B4TN-iRn6uTwo3bNDGcaHEXx4c4lCvS";
```

Μέσα στην συνάρτηση του προγράμματος void setup() που χρησιμοποιείται για την αρχικοποίηση μεταβλητών, την έναρξη χρήσης βιβλιοθηκών κλπ, η οποία θα εκτελεστεί μία φορά, μετά από κάθε ενεργοποίηση ή απενεργοποίηση του ESP32.

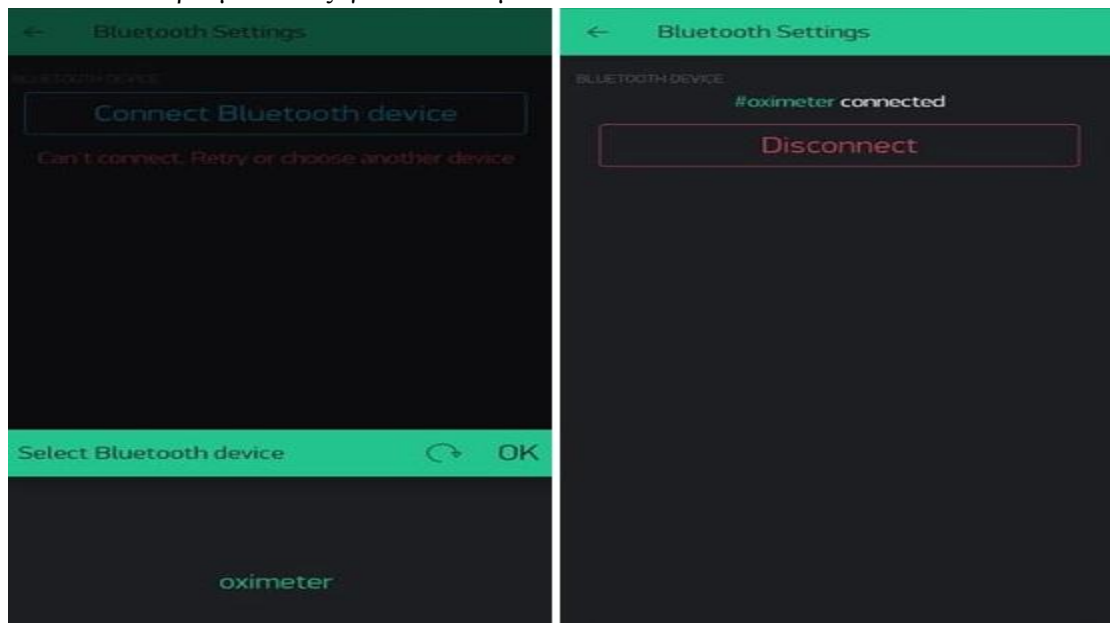
```
Blynk.setDeviceName("oximeter");
Blynk.begin(auth);
```

Με το Blynk.setDeviceName ορίζουμε το όνομα με το οποίο θα εμφανίζεται το ESP32 στην αναζήτηση του Bluetooth κινητών τηλεφώνων και το Blynk.begin(auth) δίνει την εντολή έναρξης του συνδέσμου που χρησιμοποιήσαμε παραπάνω.

Τέλος, μέσα στη void loop() η οποία είναι μία συνάρτηση επανάληψης που επιτρέπει στο πρόγραμμα να αλλάζει τα δεδομένα των μεταβλητών μας, χρησιμοποιούμε τις εντολές:

```
Blynk.run()
Blynk.virtualWrite(V3, average_beat);
Blynk.virtualWrite(V4, average_SpO2);
```

Το Blynk.run() δίνει την εντολή να ξεκινήσει η διαδικασία έναρξης της εφαρμογής και με το Blynk.virtualWrite στέλνουμε τα δεδομένα των μετρήσεων στις αντίστοιχες θέσεις που ορίσαμε στη διαδικασία δημιουργίας της εφαρμογής (στο V3 τους καρδιακούς παλμούς και στο V4 τον κορεσμό του οξυγόνου στο αίμα.



Σχήμα 4.13 «Ονομασία εύρεσης του ESP32»

Όπως βλέπουμε στην εικόνα 10 αφού επιλέξουμε το σήμα του Bluetooth από την εφαρμογή του κινητού (εικόνα 8) στην οθόνη μας, εμφανίζεται το ESP32 με το όνομα oximeter που του δώσαμε.



Σχήμα 4.14 «Μέτρηση κορεσμού και παλμών στο Display του ESP32»

Πατάμε το ok δεξιά από το Select Bluetooth device και μετά από λίγο εμφανίζεται ότι το οξύμετρο είναι συνδεδεμένο με το κινητό τηλέφωνο, έτοιμο για χρήση.

Κεφάλαιο 4^ο

Αφού τελειώσει αυτή η διαδικασία αρχίζουμε να παίρνουμε μετρήσεις.



Σχήμα 4.15 «Οι μετρήσεις στην εφαρμογή»

Όπως μπορούμε να δούμε οι τιμές που παίρνουμε με το MAX30100 στέλνονται ταυτόχρονα στο display του ESP32 και μέσω Bluetooth στην εφαρμογή την οποία δημιουργήσαμε στο κινητό τηλέφωνο.

4.4 Ο τελικός κώδικας

Με την ολοκλήρωση του προγράμματος της πτυχιακής μας θα δημιουργήσουμε ένα μενού στο display του ESP στο οποίο θα επιλέγουμε κάποια υπομενού με την χρήση κουμπιών.

Για τη δημιουργία του μενού στην αρχή του προγράμματος δηλώνουμε:

```
int menuItem = 1;  
int page = 1;
```

Το page δηλώνει τις σελίδες του μενού μας. Δηλαδή όταν page = 1 στο display εμφανίζεται η αρχική εικόνα του μενού που θα αναγράφονται τα ονόματα των υπομενού μας. Το menuItem δηλώνει το όνομα του κάθε υπομενού που έχουμε ορίσει. Όταν menuItem = 1 τότε εμφανίζεται το όνομα του πρώτου υπομενού.

Στη συνέχεια δημιουργήσαμε μία συνάρτηση void drawMenu() μέσα στην οποία γράφουμε τον κώδικα για το τι θα εμφανίζεται στο μενού μας. Χρησιμοποιήσαμε μία if, τέσσερις else if για το κύριο μενού και για τα υπομενού (page) και πέντε if για τα ονόματα των υπομενού (menuitem).

Στην πρώτη σελίδα (page=1) στην οποία θα εμφανίζεται το κύριο μενού θα αναγράφονται τα ονομασίες των υπομενού.

```
if (page==1)
{
    display.setTextSize(1);
    display.clearDisplay();
    display.setCursor(35,0);
    display.setTextColor(WHITE, BLACK);
    display.print("MAIN MENU");
    display.drawFastHLine(35,7,53,WHITE);
```

- Με το display.setTextSize(1); ορίζουμε το μέγεθος της γραμματοσειράς που θα εμφανίζεται στο display.
- Με το display.clearDisplay(); καθαρίζουμε το display έτσι ώστε να μην εμφανίζονται τα προηγούμενα αποτελέσματα.
- Με το display.setCursor(35,0); θέτουμε τον κέρσορα (να αρχίζει να εμφανίζει) στη σειρά 0 στη θέση 35.
- Με το display.setTextColor(WHITE, BLACK); ορίζουμε το background να είναι μαύρο και τα γράμματα άσπρα.
- Με το display.print("MAIN MENU"); εμφανίζουμε την φράση MAIN MENU.
- Με το display.drawFastHLine(35,7,53,WHITE); τραβήξαμε μία άσπρη γραμμή κάτω από το MAIN MENU.

Στη συνέχεια μέσα στην σελίδα 1 τοποθετήσαμε τις πέντε if για τις ονομασίες των υπομενού μας.

```
if (menuitem==1)
{
    display.setTextColor(BLACK, WHITE);
}
else
{
    display.setTextColor(WHITE, BLACK);
}
display.setCursor(0,10);
display.print("Heart Rate");
```

Κεφάλαιο 4^ο

Με αυτή την if ελέγχουμε αν είναι επιλεγμένο η ονομασία του 1^{ου} υπομενού, με την εντολή `display.setTextColor(BLACK, WHITE);` να κάνει το background άσπρο και τη γραμματοσειρά μαύρη, έτσι ώστε να μπορούμε να διακρίνουμε ποιο υπομενού είναι έτοιμο για επιλογή. Αν δεν είναι στη θέση του `menuItem==1` τότε να εμφανίζεται όπως το είχαμε ορίσει στην αρχή. Με τις εντολές `display.setCursor(0,10);` και `display.print("Heart Rate");` ορίσαμε τη θέση του κέρσορα όπως εξηγήσαμε παραπάνω και να εμφανίζει το μήνυμα Heart Rate.

Αυτό το επαναλάβαμε άλλες τέσσερις φορές για τις υπόλοιπες ονομασίες των υπομενού.

- Στο `menuItem==2` να εμφανίζει SpO2.
- Στο `menuItem==3` να εμφανίζει Bpm & SpO2.
- Στο `menuItem==4` να εμφανίζει Creators.
- Στο `menuItem==5` Turn off.

Αφού τελειώσαμε με την 1^η σελίδα που είναι και το κύριο μενού δημιουργήσαμε τις υπόλοιπες.

```
else if (page==2)
{
  display.setTextSize(1);
  display.clearDisplay();
  display.setCursor(25, 0);
  display.print("Heart Rate");
  display.drawFastHLine(25,10,60,WHITE);
  display.setCursor(5, 25);
  display.print("BPM : ");
  display.setTextSize(2);
  display.setCursor(5, 40);
  display.print(average_beat);
  display.setCursor(65,40);
  display.setTextSize(2);
  display.write(3);
  display.display();
}
```

Αν βρισκόμαστε στη 2^η εκτελούνται οι παραπάνω εντολές. Τη λειτουργία τους την εξηγήσαμε εκτός από τις:

- `display.print(average_beat);` η οποία εμφανίζει τον καρδιακό παλμό που παίρνουμε με τον MAX30100, αφού πρώτα φιλτραριστεί.
- `display.write(3);` με αυτή την εντολή εμφανίζεται μία καρδιά στη θέση που έχουμε ορίσει.

Αυτό το εφαρμόζουμε άλλες τρεις φορές για τις υπόλοιπες σελίδες του μενού μας. Να σημειωθεί ότι για το Turn off δεν χρησιμοποιούμε σελίδα γιατί είναι η επιλογή που εκτελείται η εντολή Deep Sleep του ESP32 που εξηγείται παρακάτω.

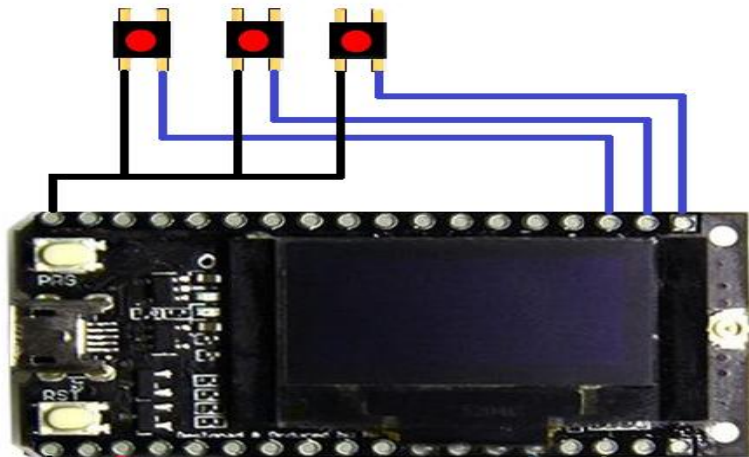
- Στην `page==3` εμφανίζεται ο κορεσμός οξυγόνου στο αίμα με την εντολή `display.print(average_SpO2);`

- Στην `page==4` εμφανίζονται οι καρδιακοί παλμοί και ο κορεσμός του οξυγόνου στο αίμα μαζί.
- Στην `page==5` εμφανίζονται τα ονοματεπώνυμα και τα ΚΑΣ των δημιουργών της κατασκευής.

Αφού τελειώσαμε την δημιουργία του μενού, μέσα σε μία `timer interrupt` η οποία βρίσκεται στην `void loop()` καλούμαι το μενού με την εντολή `drawMenu()`;

Αρχικά συνδέσαμε 3 κουμπιά με το ESP32 για τον έλεγχο του μενού. Η δήλωση των κουμπιών στο ESP32 έγιναν με τις εντολές:

Σχήμα 4.16 «Συνδεσμολογία των Buttons»



```
const int buttonPin1=13;
const int buttonPin2=12;
const int buttonPin3=21;
```

Στη συνέχεια χρησιμοποιούμε τις μεταβλητές `buttonState` οι οποίες αποθηκεύουν την τρέχουσα κατάσταση του κουμπιού. Στην αρχή τα κουμπιά είναι σε κατάσταση 0 και αν πατηθούν αλλάζουν σε κατάσταση λογικού 1. Έτσι γνωρίζουμε ποιο κουμπί πατήθηκε.

```
int downButtonState = 0;
int upButtonState = 0;
int selectButtonState = 0;
int lastDownButtonState = 0;
int lastSelectButtonState = 0;
int lastUpButtonState = 0;
```

Τις τρεις παραπάνω εντολές της χρησιμοποιούμε για τον έλεγχο που θα κάνουμε κατά τη διάρκεια που τρέχει ο κώδικας για να ελέγξουμε αν το κουμπί πατήθηκε.

Μέσα στη συνάρτηση `void setup()` δηλώνουμε το κάθε κουμπί ξεχωριστά με τη συνθήκη `pinMode(buttonPin1, INPUT_PULLUP)`;

Το `INPUT_PULLUP` το βάζουμε διότι, δεν χρησιμοποιούμε αντιστάσεις στα κουμπιά. Η λειτουργία του είναι, όταν το κουμπί δεν είναι πατημένο η εσωτερική αντίσταση είναι συνδεδεμένη στα 3.3V του ESP32 και έτσι βρίσκεται σε κατάσταση HIGH. Ωστόσο, αν το κουμπί πατηθεί η κατάσταση αλλάζει σε LOW.

Κεφάλαιο 4°

Μέσα στη συνάρτηση void loop() τοποθετούμε τη εντολή:

```
selectButtonState = digitalRead(buttonPin1); (για όλα τα κουμπιά αντίστοιχα).
```

Με την οποία διαβάζουμε την τωρινή κατάσταση των κουμπιών (HIGH). Αν κάποιο κουμπί πατηθεί η κατάσταση αλλάζει (LOW) και εκτελείται η εντολή την οποία έχουμε ορίσει με το πάτημά του. Επιπλέον βάζουμε και τις εντολές:

```
checkIfDownButtonIsPressed();  
checkIfUpButtonIsPressed();  
checkIfSelectButtonIsPressed();
```

Είναι συναρτήσεις στον κώδικά μας που κάνουν τον έλεγχο αν κάποιο κουμπί έχει πατηθεί. Αυτές οι συναρτήσεις έχουν τη μορφή:

```
Void checkIfSelectButtonIsPressed()  
{  
  if (selectButtonState != lastSelectButtonState)  
  {  
    if (selectButtonState == 0) {  
      middle=true;  
    }  
    delay(50);  
  }  
  lastSelectButtonState = selectButtonState;  
}
```

Στην αρχή με την εντολή if ελέγχουμε την τωρινή κατάσταση του κουμπιού (HIGH ή LOW) και αν είναι αντίθετες (το selectButtonState σε κατάσταση HIGH και το lastSelectButtonState σε κατάσταση LOW) τότε εκτελείται μία άλλη if, η οποία ελέγχει αν η κατάστασή του είναι 0. Αν η συνθήκη ισχύει εκτελούνται οι εντολές που θέσαμε στο κουμπί. Η συγκεκριμένη συνθήκη αφορά το κουμπί middle, το οποίο όταν πατιέται, από το κεντρικό μενού μας στέλνει στα υπόλοιπα υπομενού ή μας ξαναβγάζει στο κύριο. Τέλος ορίζουμε στην τελευταία κατάσταση του κουμπιού την τωρινή. Το ίδιο κάνουμε και για τα άλλα δύο κουμπιά (Up & Down).

Αφού τελειώσαμε με την εισαγωγή των κουμπιών και τον τρόπο ελέγχου τους, μέσα στην Void loop() βάλαμε τις εντολές με τις οποίες θα μετακινούμαστε στο μενού και θα επιλέγουμε τα υπομενού με την χρήση των κουμπιών.

```
if (up && page == 1 ) {  
  up = false;  
  menuitem--;  
  if (menuitem==0)  
  {  
    menuitem=5;  
  }  
}
```

Με την πρώτη if ελέγχουμε αν ισχύει η συνθήκη, ότι πατήθηκε το κουμπί up και ότι βρισκόμαστε στην 1^η σελίδα. Εφόσον ισχύει αλλάζουμε την κατάσταση του κουμπιού στην οποία βρισκόταν και αφαιρούμε από το υπάρχον menuItemem μία θέση έτσι ώστε να κινηθεί ο κέρσορας προς τα πάνω. Με την επόμενη if ελέγχουμε αν το menuItemem είναι 0. Με αυτό τον τρόπο στέλνουμε τον κέρσορα στο τελευταίο menuItemem.

Στη συνέχεια ο κώδικας για το κουμπί down θα έχει σχεδόν την ίδια μορφή.

- Αντί για menuItemem--; βάζουμε menuItemem++;, έτσι ώστε ο κέρσορας να αυξάνει μία θέση αντί να μειώνει και να μετακινείται προς τα κάτω.
- Αντί για menuItemem==0 και menuItemem=5; στη δεύτερη if βάζουμε menuItemem==6 και menuItemem=1;,, έτσι ώστε ο κέρσορας με το που φτάνει στο τελευταίο menuItemem να επιστρέφει πάλι στο πρώτο.

Για τις διεργασίες που θα εκτελέσει το κουμπί middle ξεκινήσαμε με μία if (middle) η οποία δίνει τις παρακάτω εντολές σε περίπτωση που το κουμπί πατηθεί.

```
if (page == 1 && menuItemem==1) {
    page=2;
}
else if (page == 2 ) {
    page=1;
}
```

Ελέγχει αν ισχύει η συνθήκη, ότι βρισκόμαστε στην πρώτη σελίδα και ότι είμαστε στο 1^ο menuItemem και στην συνέχεια μας εμφανίζει την δεύτερη σελίδα. Στην περίπτωση που είμαστε ήδη στην δεύτερη σελίδα και πατηθεί πάλι το κουμπί μας γυρνάει πίσω στην 1^η. Αυτή τη διαδικασία την επαναλαμβάνουμε άλλες τρεις φορές αλλάζοντας το menuItemem και την σελίδα η οποία θέλουμε να εμφανίζεται.

Στην τελευταία συνθήκη (5^η) ελέγχουμε αν βρισκόμαστε στην 1^η σελίδα και στο 5^ο menuItemem, εντάσσουμε τον κώδικα του Deep Sleep.

Κεφάλαιο 4°

Τα αποτελέσματα τα οποία εμφανίζονται στο display του ESP32 θα είναι:



Σχήμα 4.17 «Αρχικό μενού στην οθόνη του ESP32»



Σχήμα 4.18 «Υπομενού παλμών»



Σχήμα 4.19 «Υπομενού κορεσμού οξυγόνου»



Σχήμα 4.20 «Υπομενού παλμών και κορεσμού οξυγόνου»



Σχήμα 4.21 «Υπομενού στοιχείων των δημιουργών»

Αν το ESP32 λειτουργεί με μπαταρία, η ισχύ της θα εξαντλείται γρήγορα. Με τη χρήση του Deep Sleep θα μειωθεί η κατανάλωση ενέργειας και η μπαταρία θα μπορεί να διαρκέσει παραπάνω. Όταν το ESP32 βρίσκεται σε Deep Sleep mode σταματάνε οι δραστηριότητες που καταναλώνουν περισσότερη ενέργεια κατά τη λειτουργία του. Σε αυτό το mode δεν λειτουργούν η CPU ή το Wifi, αλλά ο co-processor Ultra Low Power (ULP) συνεχίζει να βρίσκεται σε ενεργή κατάσταση.

Για να βγάλουμε το ESP32 από το Deep Sleep mode, χρησιμοποιούμε μία πηγή wake up.

- Έναν Timer ο οποίος θα “ξυπνάει” το ESP32 χρησιμοποιώντας προκαθορισμένες χρονικές περιόδους.
- Ένα touch pin.
- Μία ή πολλές διαφορετικές εξωτερικές αφυπνίσεις.
- Τον επεξεργαστή ULP.

Στον παρακάτω πίνακα μπορούμε να συγκρίνουμε τις 5 διαφορετικές λειτουργίες.

Power mode	Active	Modem-sleep	Light-sleep	Deep-sleep	Hibernation
Sleep pattern	Association sleep pattern			ULP sensor-monitored pattern	-
CPU	ON	ON	PAUSE	OFF	OFF
Wi-Fi/BT baseband and radio	ON	OFF	OFF	OFF	OFF
RTC memory and RTC peripherals	ON	ON	ON	ON	OFF
ULP co-processor	ON	ON	ON	ON/OFF	OFF

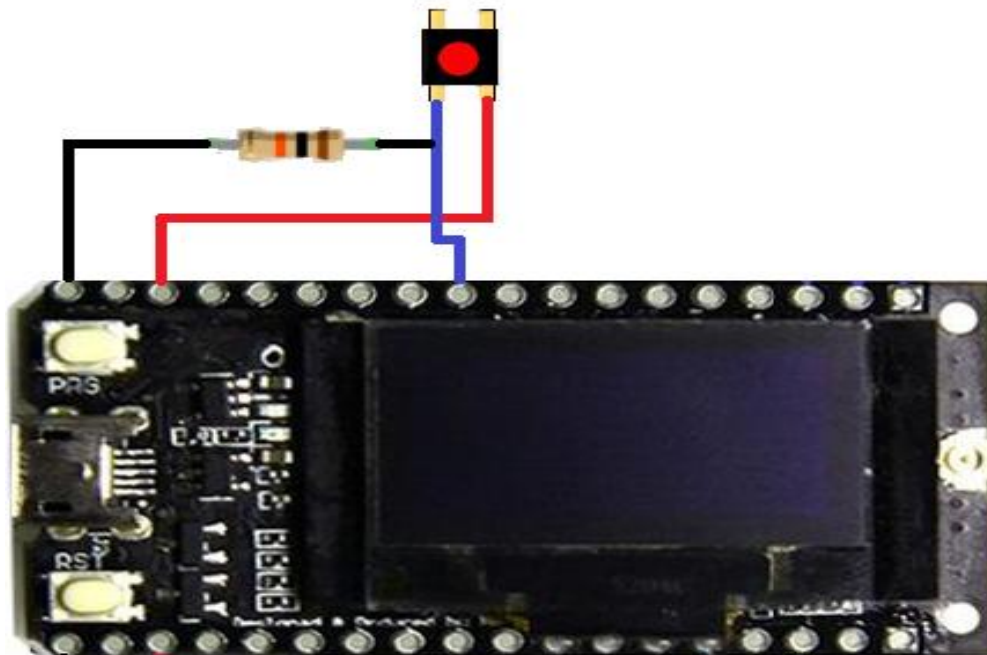
Σχήμα 4.22 «Λειτουργίες -mode για το DeepSleep»

Power mode	Description	Power consumption
Active (RF working)	Wi-Fi Tx packet 14 dBm ~ 19.5 dBm	Please refer to Table 10 for details.
	Wi-Fi / BT Tx packet 0 dBm	
	Wi-Fi / BT Rx and listening	
Modem-sleep	The CPU is powered on.	Max speed 240 MHz: 30 mA ~ 50 mA
		Normal speed 80 MHz: 20 mA ~ 25 mA
		Slow speed 2 MHz: 2 mA ~ 4 mA
Light-sleep	-	0.8 mA
Deep-sleep	The ULP co-processor is powered on.	150 μ A
	ULP sensor-monitored pattern	100 μ A @1% duty
	RTC timer + RTC memory	10 μ A
Hibernation	RTC timer only	5 μ A
Power off	CHIP_PU is set to low level, the chip is powered off	0.1 μ A

Σχήμα 4.23 «Κατανάλωση ενέργειας ανάλογα την λειτουργία»

Στον παραπάνω πίνακα συγκρίνεται η κατανάλωση ενέργειας των διαφορετικών τρόπων λειτουργίας.

Εμείς επιλέξαμε ένα κουμπί το οποίο εντάσσεται στις εξωτερικές αφυπνίσεις. Συνδέσαμε ένα ακόμα κουμπί πάνω στο ESP32 το οποίο θα παίρνει 3,3V τροφοδοσία και θα έχει μία pull down αντίσταση 10k Ω .



Σχήμα 4.24 «Σύνδεση Button στο ESP32 με Pull Down αντίσταση»

Κεφάλαιο 4°

Αφού ολοκληρώσαμε το κύκλωμά μας, ήρθε η ώρα να γράψουμε τον κώδικα. Αρχικά δηλώσαμε το pin στο οποίο συνδέσαμε το κουμπί για το deep sleep.

```
#define GPIO 35
```

Στη συνέχεια για να ενεργοποιούμε την κατάσταση deep sleep τοποθετήσαμε την εντολή `esp_deep_sleep_start()`; μέσα στη `Void loop()`. Πιο συγκεκριμένα στην τελευταία συνθήκη του κουμπιού `middle`, όπως προαναφέρθηκε παραπάνω.

Τέλος, μέσα στην `Void Setup()` με την εντολή `esp_sleep_enable_ext0_wakeup(GPIO_NUM_35,1)`;

Δίνουμε δηλώνουμε πως θα απενεργοποιείται η κατάσταση Deep Sleep. Δηλαδή όταν πατιέται το κουμπί που είναι συνδεδεμένο στο pin 35 και η κατάσταση που γίνεται λογικό 1 από λογικό 0.

Σαν έξτρα τοποθετήσαμε μία εικόνα η οποία θα εμφανίζεται για 2000ms λίγο πριν την απενεργοποίηση του ESP32. Αυτό το καταφέραμε με τον εξής τρόπο.

Αλλάξαμε το μέγεθος της εικόνας σε 128x64 που είναι οι διαστάσεις του display που χρησιμοποιούμε. Με την χρήση της ιστοσελίδας <http://javl.github.io/image2cpp/> δημιουργήσαμε τον κώδικα που θα τοποθετήσουμε στο πρόγραμμά μας με τα παρακάτω βήματα.

Βήμα 1°

1. Select image

Επιλογή αρχείων 61de41aV4bL.bmp

or

1. Paste byte array



128 x 64 px

Read as horizontal

Read as vertical

Σχήμα 4.25 «Επιλογή εικόνας bmp.»

Επιλέξαμε την εικόνα που θέλουμε να εμφανίσουμε στο display.

Βήμα 2°

2. Image Settings

Canvas size(s): 61de41av4bL.bmp (file resolution: 128 x 64)
 x glyph

Background color: White Black Transparent

Invert image colors

Brightness / alpha threshold:
0 - 255; if the brightness of a pixel is above the given level the pixel becomes white, otherwise they become black. When using alpha, opaque and transparent are used instead.

Scaling:

Center: horizontally vertically

Note: centering the image only works when using a canvas larger than the original image.

Σχήμα 4.26 «Ορισμός διαστάσεων εικόνας bmp.»

στις ρυθμίσεις επιλέξαμε το μέγεθος που θέλουμε να καλύπτει η εικόνα πάνω στο display, το χρώμα του background και ότι επιθυμούμε να εμφανίζεται οριζόντια.

Βήμα 3° και 4°

3. Preview

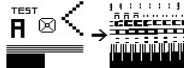


4. Output

Code output format:

Draw mode:

If your image looks all messed up on your display, like the image below, try using a different mode.



```
// '61de41av4bL', 128x64px
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
```

Σχήμα 4.27 «Κώδικας εικόνας bmp.»

Στο 3° βήμα μας δείχνει πως θα εμφανίζεται η εικόνα στο display. Από τη στιγμή που συμφωνούμε στο 4° βήμα επιλέγουμε τη μορφή που επιθυμούμε να έχει ο κώδικας. Στη συνέχεια πατάμε Generate code και μας εμφανίζεται μια σειρά θέσεων.

Κεφάλαιο 4°

Αφού τελειώσει αυτή η διαδικασία μεταφέρουμε αυτή τη σειρά θέσεων μέσα στο κύριο πρόγραμμα, στην αρχή μετά την δήλωση των μεταβλητών, με την χρήση της εντολής:
`const unsigned char heart [] PROGMEM = { };`

Για να εμφανίσουμε την εικόνα μας λίγο πριν την ενεργοποίηση του Deep sleep mode χρησιμοποιούμε τον παρακάτω κώδικα μέσα στη Void loop() στις συνθήκες του κουμπιού middle.

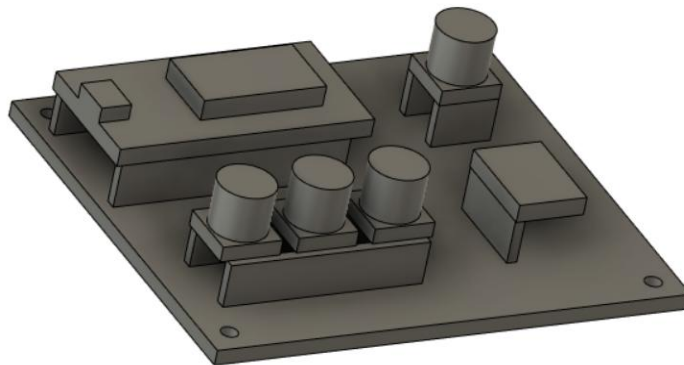
```
if (page == 1 && menuItem==5) {  
    display.clearDisplay();  
    display.drawBitmap(0, 0, heart, 128, 64, WHITE);  
    display.display();  
    delay(2000);  
    esp_deep_sleep_start();  
}
```

Αν βρισκόμαστε στην page 1 και στο menuItem 5 τότε καθαρίζουμε το display, με την εντολή `display.clearDisplay()`; στη θέση 0, 0 εμφανίζουμε την φωτογραφία η οποία είναι σε διαστάσεις 128x64 σε άσπρο χρώμα. Τέλος μετά από μία χρονοκαθυστέρηση 2000ms (`delay(2000);`) ενεργοποιείται το Deep Sleep mode.

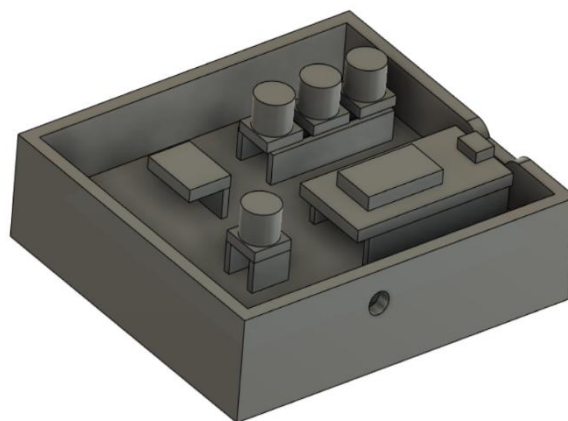


Σχήμα 4.28 «Εικόνα ενεργοποίησης DeepSleep»

Αφού ολοκληρώσαμε τη διαδικασία της δημιουργίας του παλμικού οξύμετρου, με την χρήση του προγράμματος Fusion 360 σχεδιάσαμε ένα κουτί στο οποίο θα τοποθετήσουμε την πλακέτα που κατασκευάσαμε.



Σχήμα 4.29 «Σχεδίαση πλακέτας και εξαρτημάτων»



Σχήμα 4.30«Σχεδίαση του κάτω μέρος κουτιού»



Σχήμα 4.31«Σχεδίαση του πάνω μέρους κουτιού»

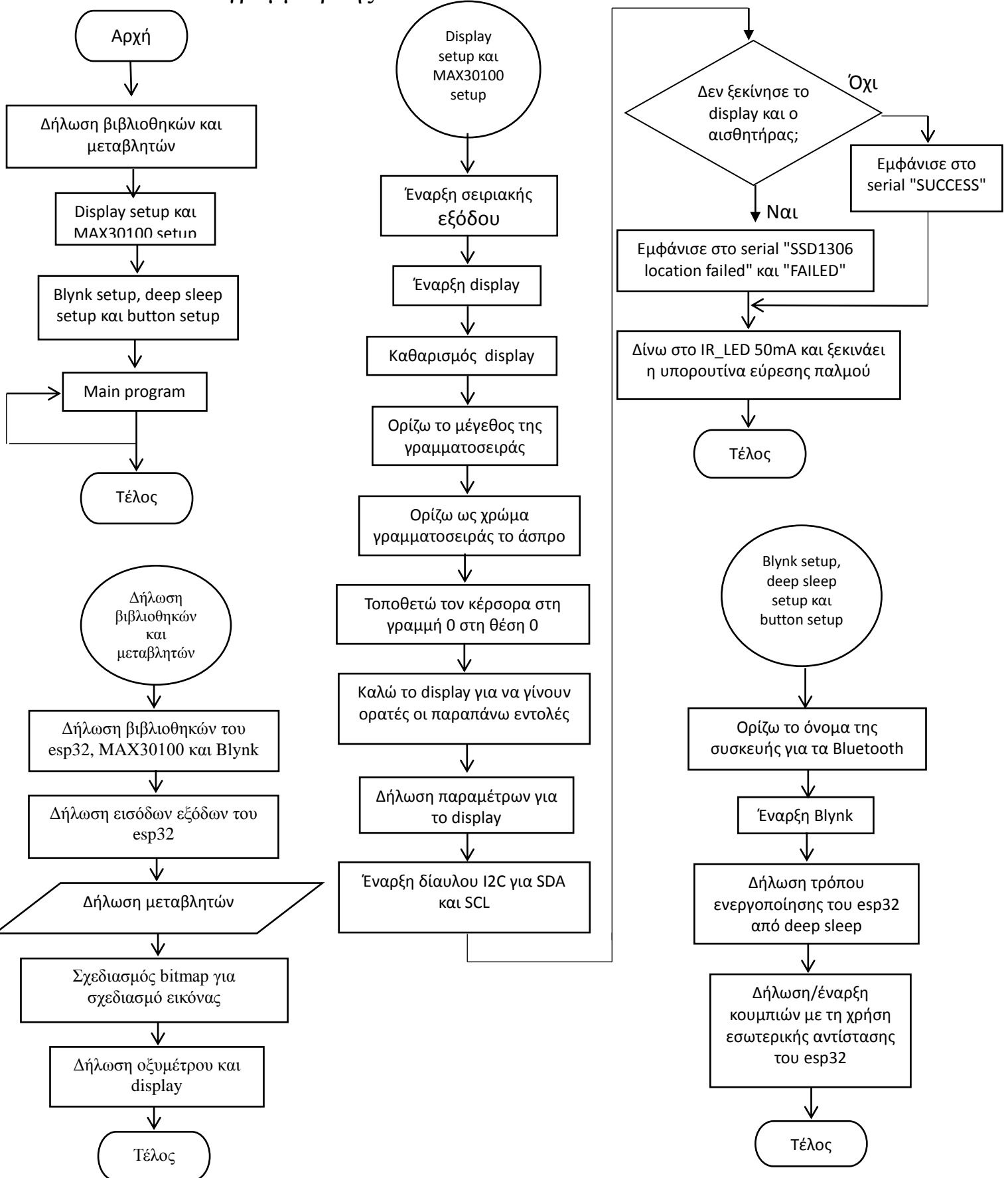
Κεφάλαιο 4°

Τέλος αφού ολοκληρώσαμε την σχεδίαση, με την χρήση του Creality3D Ender-3 pro High Precision 3D Printer τυπώσαμε το κουτί.

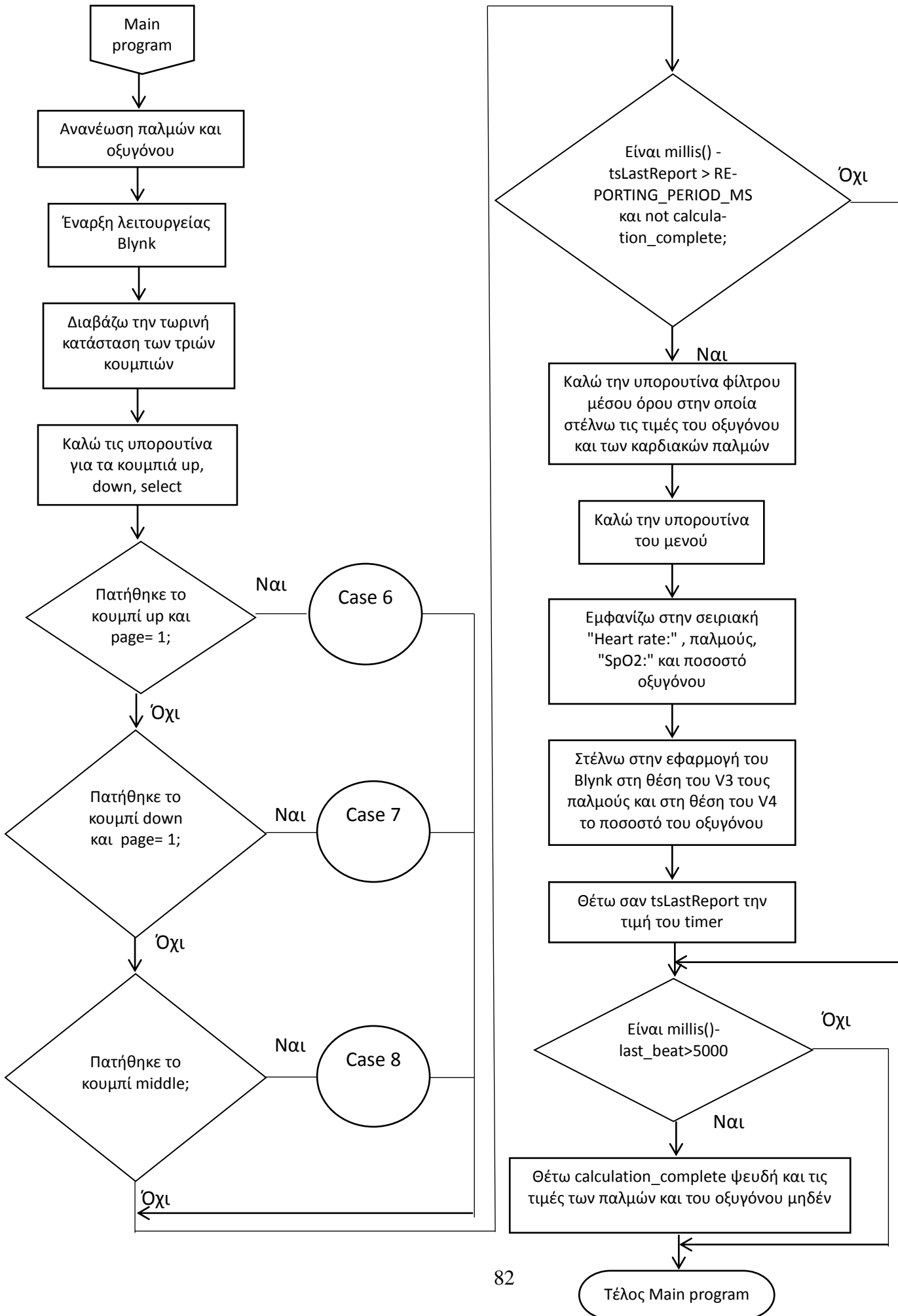


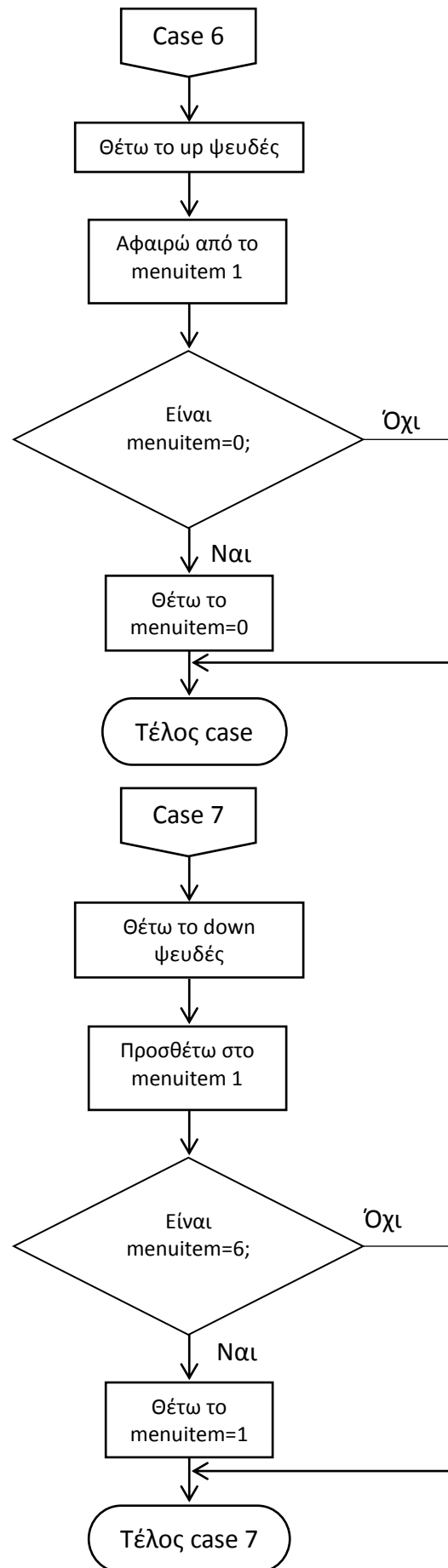
Σχήμα 4.32 «Τελική μορφή»

4.5 Διάγραμμα ροής

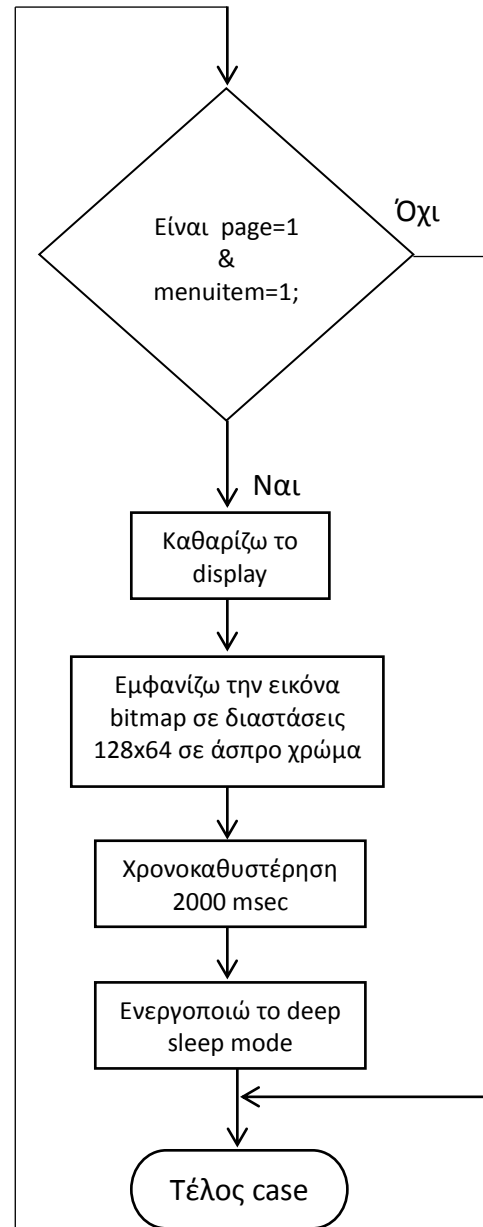
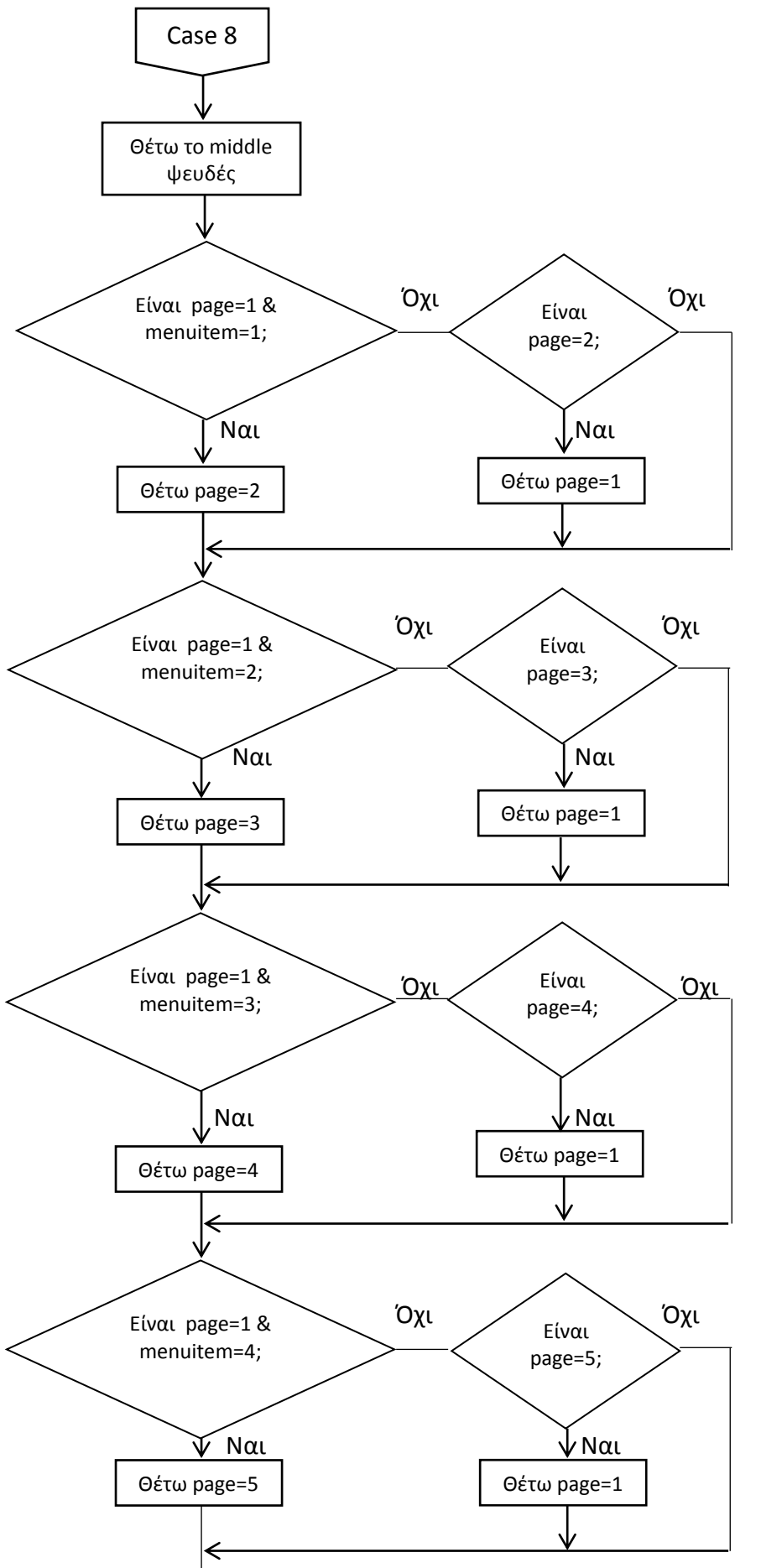


Κεφάλαιο 4°

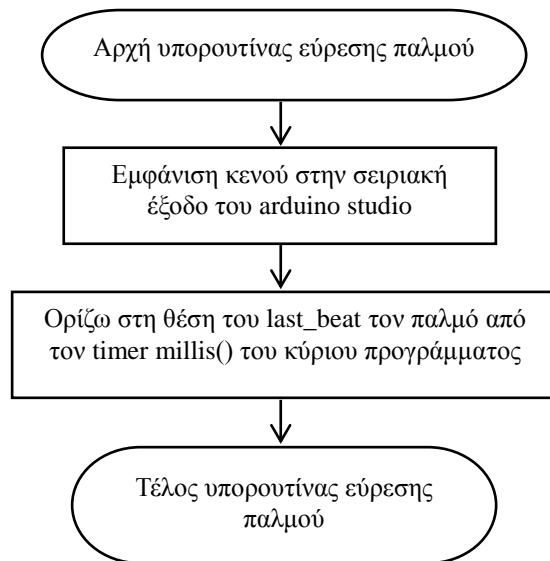




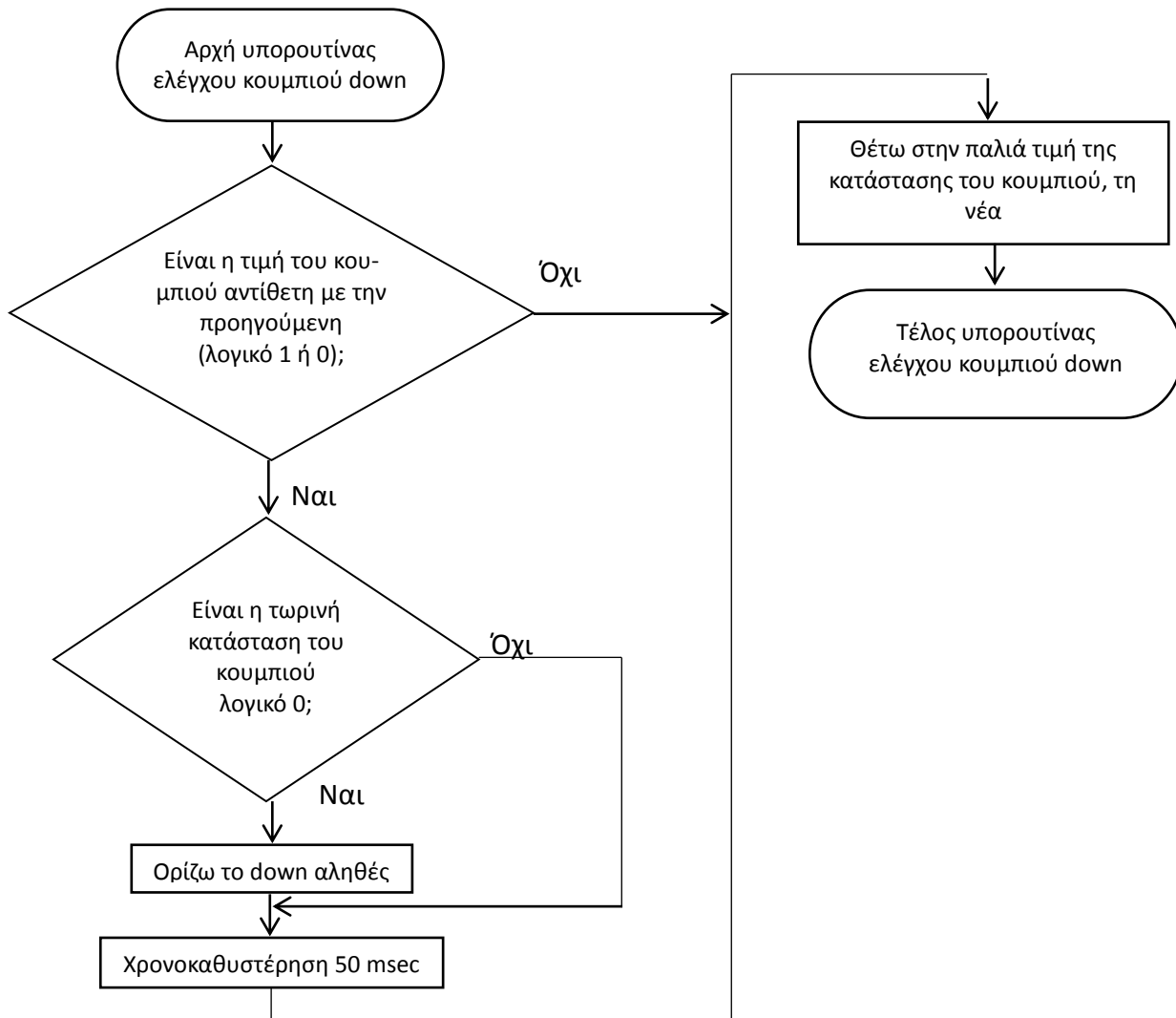
Κεφάλαιο 4°



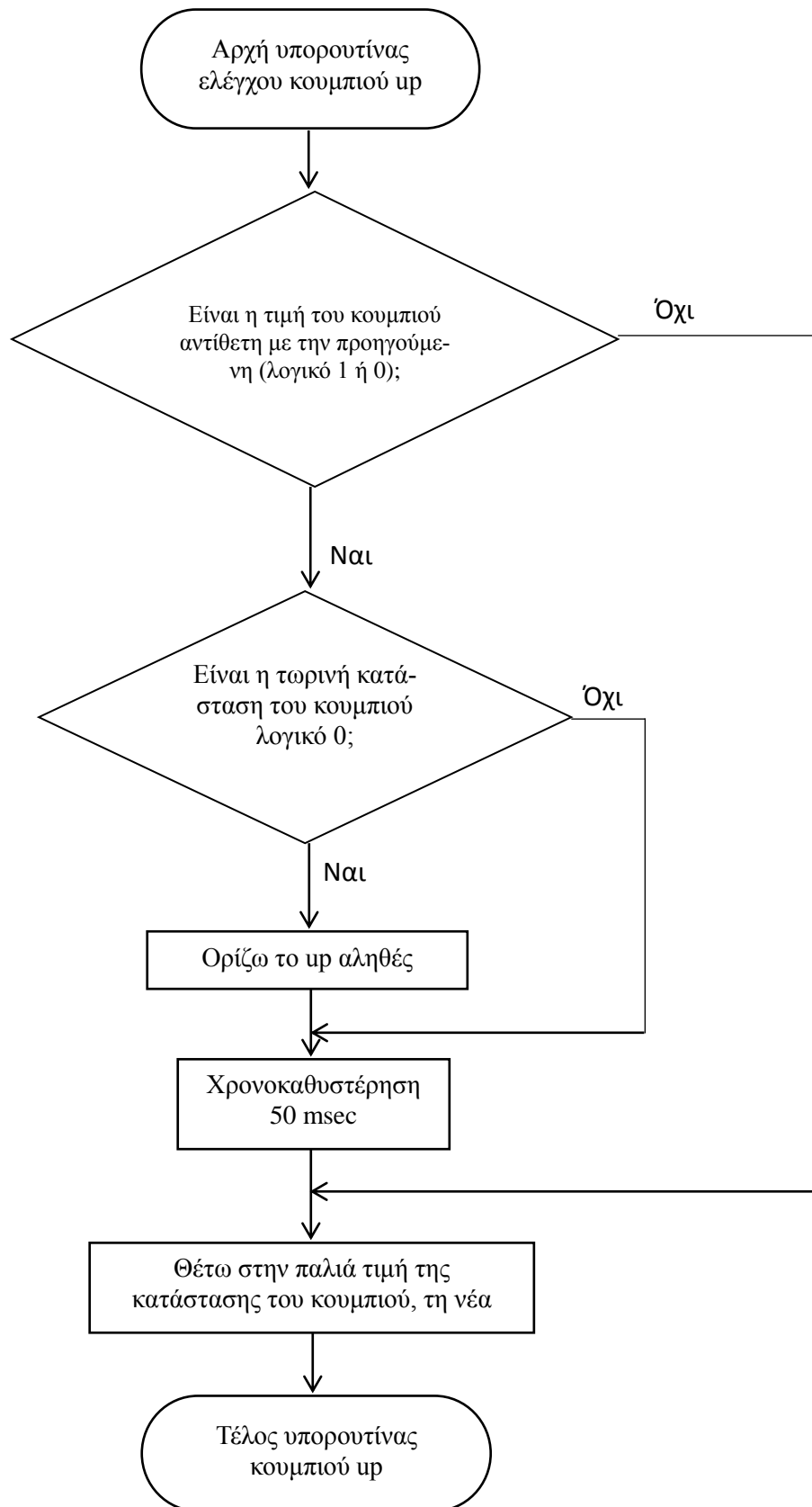
Υπορουτίνα 1 (εύρεση παλμού)



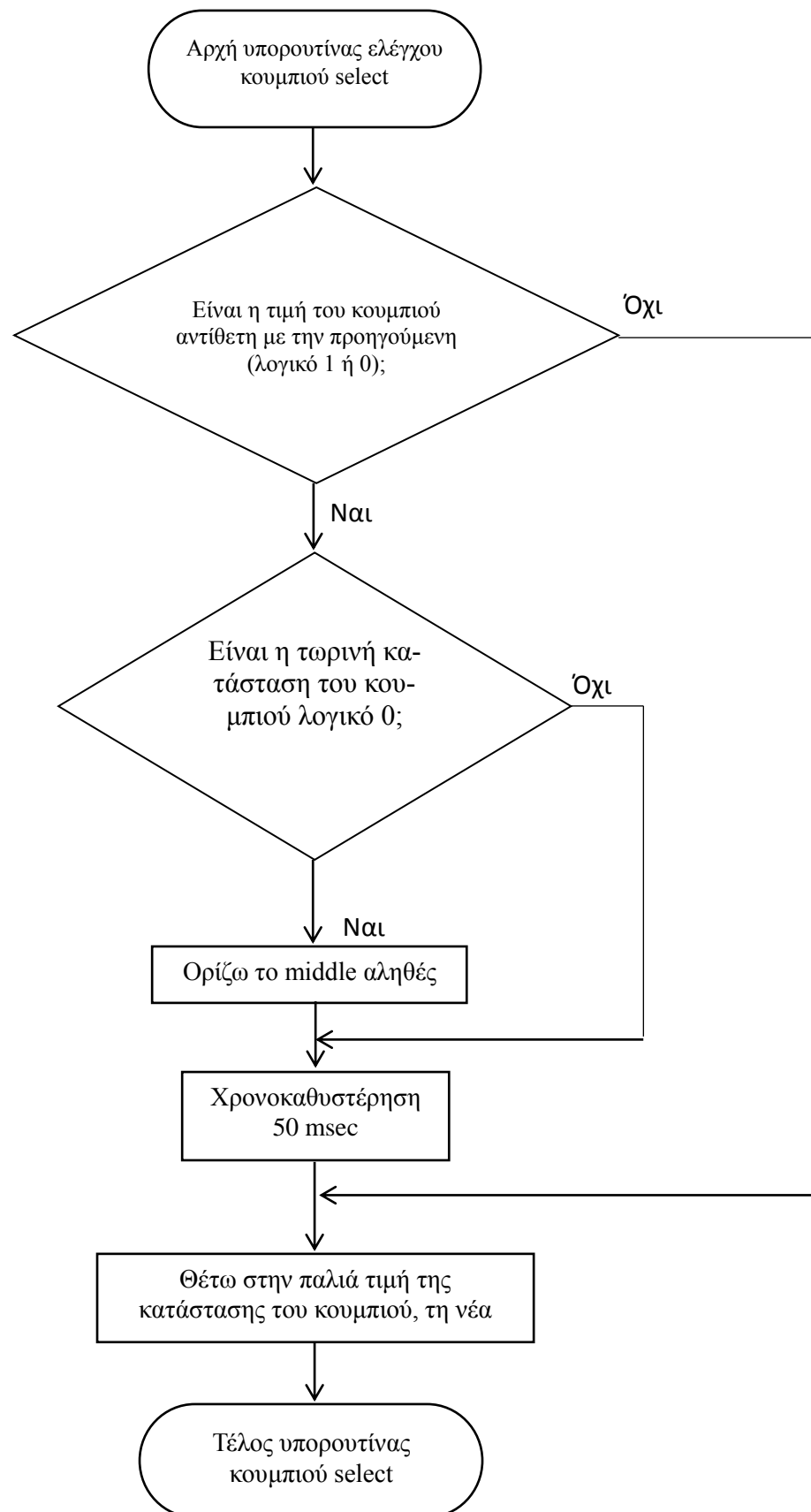
Υπορουτίνα 2 (ελέγχου κουμπιού down)



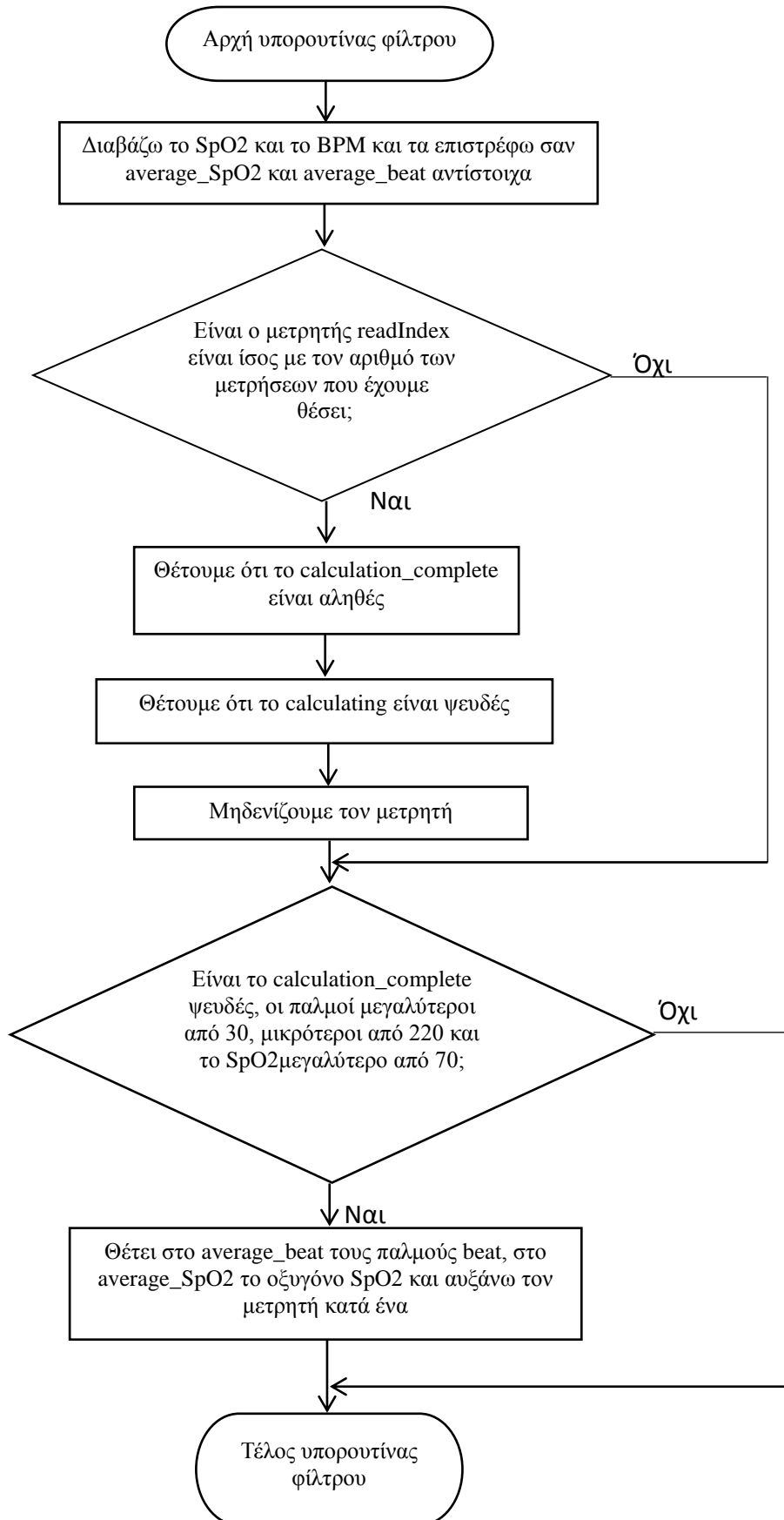
Υπορουτίνα 3 (ελέγχου κουμπιού up)



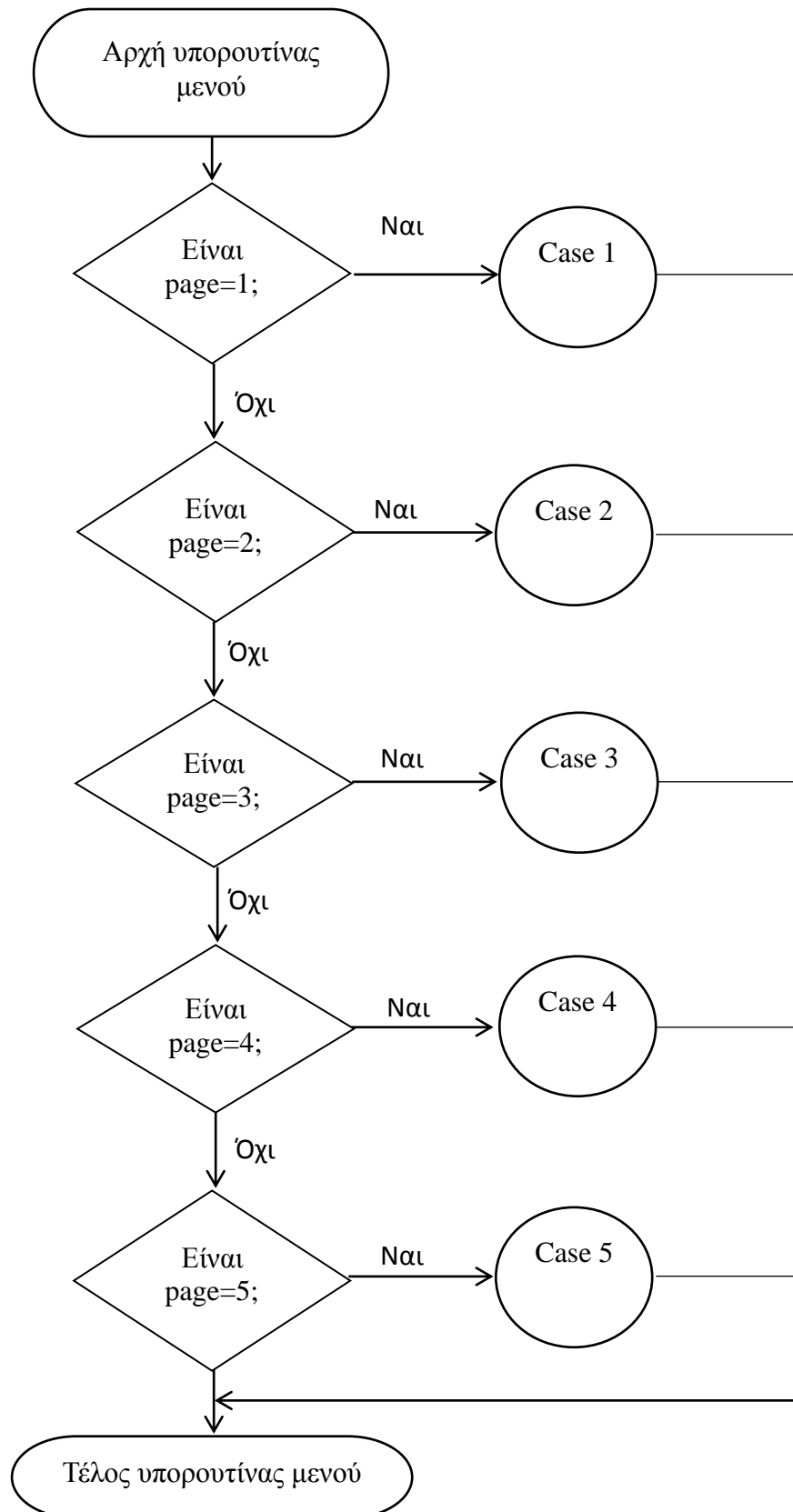
Υπορουτίνα 4 (ελέγχου κουμπιού select)

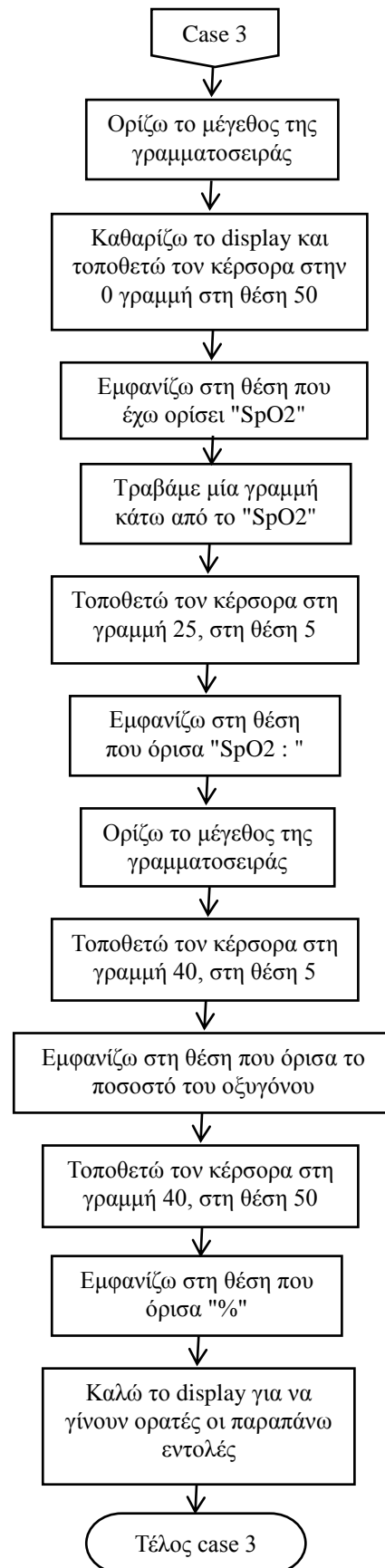
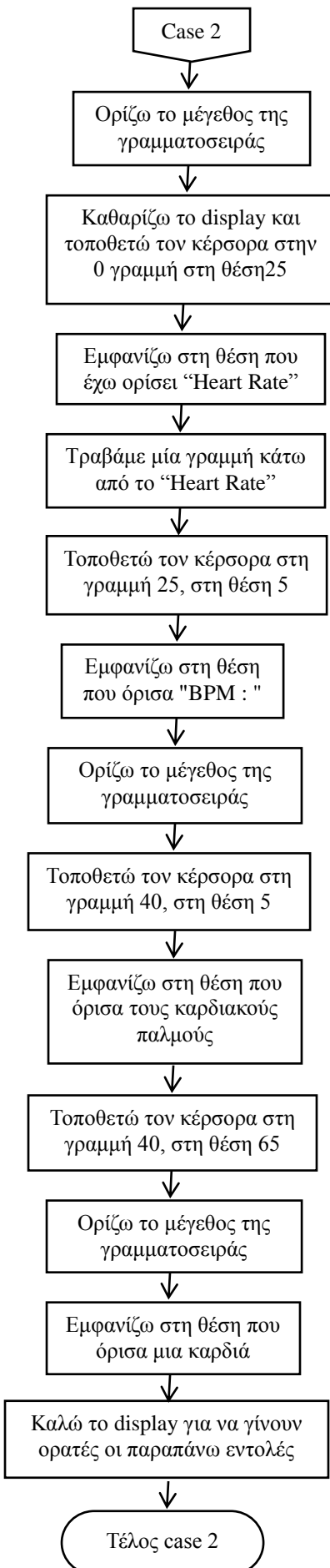


Υπορουτίνα 5 (φίλτρο μέσου όρου)

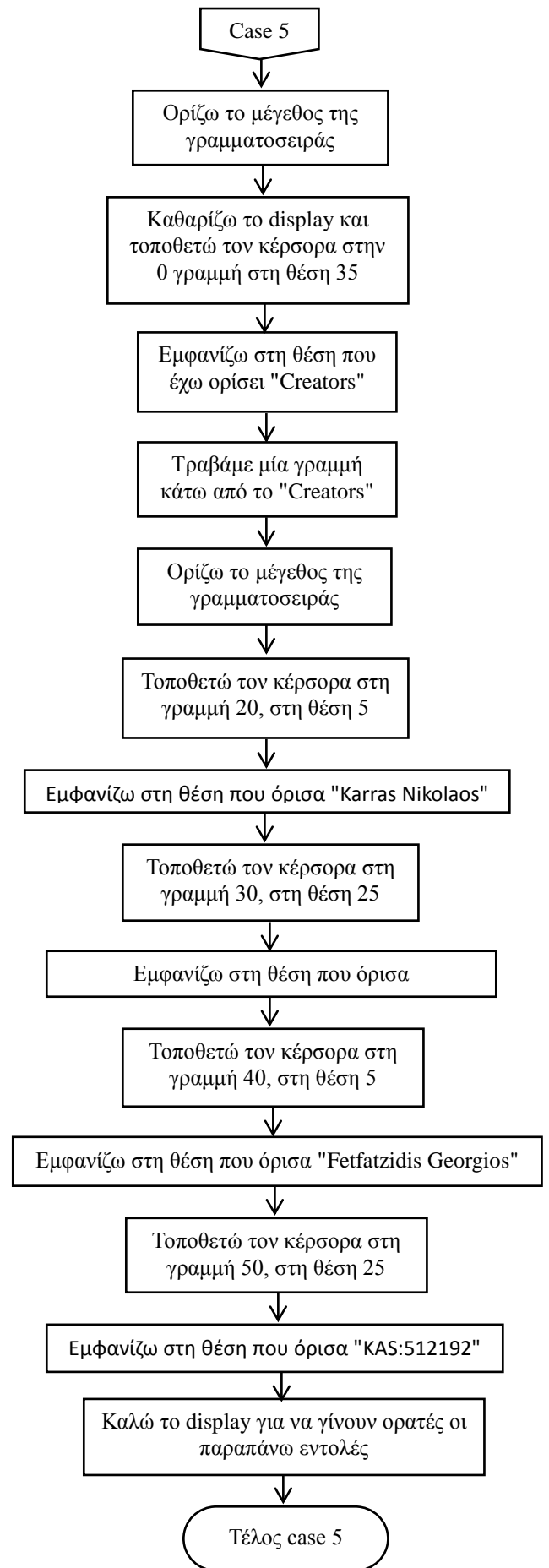
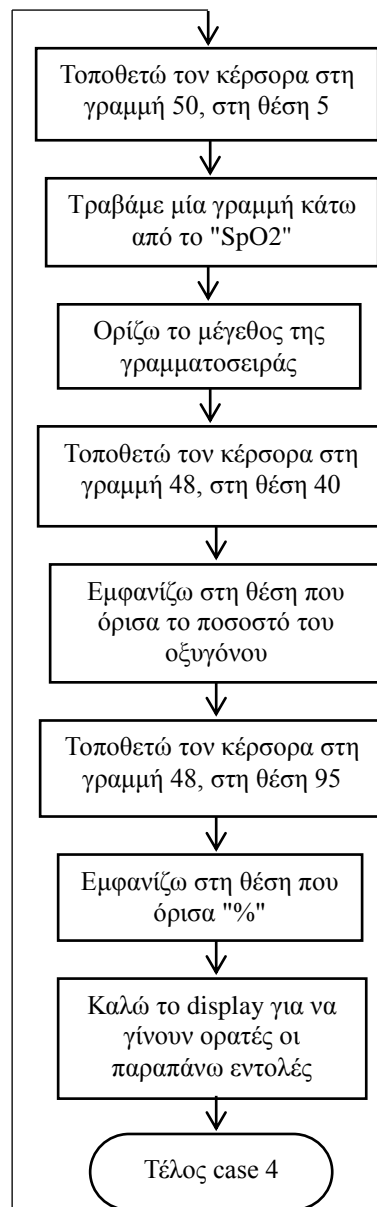
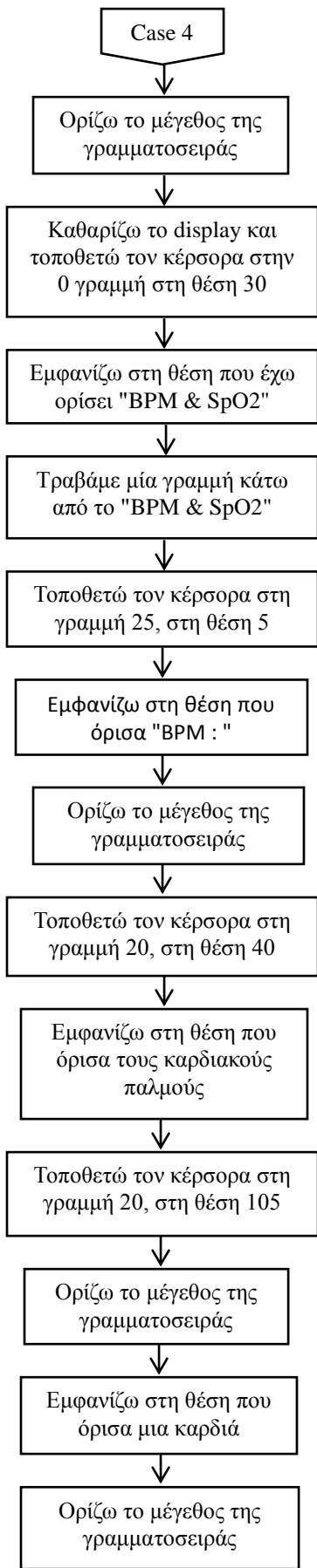


Υπορουτίνα 6 (μενού)





Κεφάλαιο 4°



Κεφάλαιο 5^ο: Συμπεράσματα

5.1 Αποτελέσματα-συμπεράσματα

Κατά τη μελέτη των πιο κοινών αισθητήρων που χρησιμοποιήθηκαν για αυτό το σενάριο, καταλήξαμε στο συμπέρασμα ότι ένας αισθητήρας PhotoPletysmoGraphy (PPG), συγκεκριμένα το MAX30100, ήταν μια καλή επιλογή λόγω της απλότητας και του μειωμένου μεγέθους. Η εφαρμογή του MAX30100 για τη μέτρηση του καρδιακού ρυθμού και του κορεσμού οξυγόνου ήταν πιο περίπλοκη από το αναμενόμενο, καθώς περιλάμβανε ουσιαστική επεξεργασία σήματος που δεν είχε ληφθεί υπόψη. Αν και το MAX30100 δεν μπορεί να επεξεργαστεί τα δεδομένα σε ένα τέλεια διαμορφωμένο σήμα, έχει την ευκολία να ενσωματώσει ένα μάλλον περίπλοκο αναλογικό κύκλωμα σε ένα εξαιρετικά μικρό πακέτο. Όταν χρησιμοποιείτε αυτό το προϊόν σημειώστε ότι απαιτούνται κατάλληλα εμπειρικά δεδομένα για τη βαθμολόγηση του αισθητήρα προκειμένου να αυξηθεί η ακρίβεια των μετρήσεων SpO₂.

Μετά από επαρκή έρευνα, χρησιμοποιήθηκε η μονάδα ESP32 για την υπολογιστική της ικανότητα (32-dual-core 32-επεξεργαστής bit), συνδεσιμότητα Bluetooth και ένα σύνολο λειτουργιών ύπνου και μειωμένης κατανάλωσης ενέργειας, την διαθεσιμότητα και το κόστος. Συγκεκριμένα, η υπολογιστική ισχύς είναι επαρκής για να εκτελεί όλη την επεξεργασία σήματος και τις λειτουργίες που απαιτούνται για τον κατάλληλο χειρισμό του σήματος PPG.

δεδομένου ότι αποτελεί σημαντική βοήθεια κατά το σχεδιασμό εφαρμογών σε πραγματικό χρόνο, παρέχοντας εργαλεία για τη δημιουργία και τη διαχείριση εργασιών και τον προγραμματισμό του. Η τελική διαμόρφωση που επιλέχθηκε για το σύστημα χρησιμοποιεί 7.6 mA ρεύματος για το IR LED, δειγματοληψία συχνότητα 50 Hz και πλάτος παλμού 200 μs. Αποδείχθηκε ότι πληρούσε τους στόχους που είχε οριστεί και έχει συνολική σταθερή και αξιόπιστη απόδοση.

5.2 Μελλοντικές βελτιώσεις

Για την βελτίωση της εργασίας θα μπορούσαμε να τοποθετήσουμε μια μπαταρία ώστε να είναι φορητή η συσκευή και να δούμε σε εφαρμογή η χαμηλή κατανάλωση ενέργειας. Δεν έγινε διότι μετά από μελέτη είδαμε ότι υπάρχει ένα μικρό ρίσκο η μπαταρία να προξενήσει δυσλειτουργία ή αν δεν είναι καλής ποιότητας μέχρι και να εκραγεί.

Μια ακόμα βελτίωση που θα μπορούσε να γίνει είναι η συνεχή σύνδεση στο διαδίκτυο μέσω ενός σημείου πρόσβασης WiFi που μπορεί να παρακολουθεί τον καρδιακό ρυθμό και τον κορεσμό οξυγόνου και να ειδοποιεί γιατρό ή πρώτες βοήθειες σε περίπτωση που οι μετρήσεις του χρήστη βρίσκονται σε κρίσιμο στάδιο.

Επίσης η χρήση καλωδίου για τον αισθητήρα είναι ακόμα ένα κομμάτι που θα μπορούσε να βελτιώσει την απόδοση των μετρήσεων κορεσμού και οξυγόνου και να κάνει την χρήση προς τον ασθενή πιο εύκολη.

Βιβλιογραφία

- [1] Witt, Chance. "Vital Signs Are Vital: The History of Pulse Oximetry." *ACP Hospitalist*. 15 May 2014. Web. June. 2020. <https://acphospitalist.org/archives/2014/05/newman.htm>
- [2] Cairns, Rebecca. "How a Small Device Called a Pulse Oximeter Measures Blood Oxygen Levels." *Insider*. Insider, 06 May 2020. Web. June 2020.
- [3] Gotter, Ana. "Pulse Oximetry: Uses, Readings, and How It Works." *Healthline*. Healthline Media, 03 Feb. 0557. Web. Dec. 2019. <https://www.healthline.com/health/pulse-oximetry>
- [4] Media, OpenSystems, Zhang Feng Is Senior Medical Applications Engineer, and Marten Smith Is Staff Medical Segment Engineer. "Measuring Heart Rate and Blood Oxygen Levels for Portable Medical and Wearable Devices." *Embedded Computing Design*. 16 Feb. 2020. Web. March. 2020. <https://www.embedded-computing.com/embedded-computing-design/measuring-heart-rate-and-blood-oxygen-levels-for-portable-medical-and-wearable-devices>
- [5] Charles Patrick Davis, MD. "Hemoglobin Ranges: Normal, Symptoms of High and Low Levels." *MedicineNet*. MedicineNet, 23 July 2019. Web. Dec. 2019. <https://www.medicinenet.com/hemoglobin/article.htm>
- [6] "Oxygen Transport - Haemoglobin - Bohr Shift." *TeachMePhysiology*. 07 July 2020. Web. 14 Jan. 2020. <https://teachmephysiology.com/respiratory-system/gas-exchange/oxygen-transport/>
- [7] "Heart." *Wikipedia*. Wikimedia Foundation, 19 Aug. 2019. Web. March. 2020. <https://en.wikipedia.org/wiki/Heart>
- [8] "4 Valves of the Heart: What Are They & How They Work." *Cleveland Clinic*. Web. March. 2020. <https://my.clevelandclinic.org/health/articles/17067-heart-valves>
- [9] Pittman, Roland N. "The Circulatory System and Oxygen Transport." *Regulation of Tissue Oxygenation*. U.S. National Library of Medicine, 01 Jan. 1970. Web. Feb. 2020. <https://www.ncbi.nlm.nih.gov/books/NBK54112/>
- [10] Libretexts. "Spectrophotometry." *Chemistry LibreTexts*. Libretexts, 15 Aug. 2020. Web. 8 Apr. 2020. [https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_\(Physical_and_Theoretical_Chemistry\)/Kinetics/Reaction_Rates/Experimental_Determination_of_Kinetics/Spectrophotometry](https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Kinetics/Reaction_Rates/Experimental_Determination_of_Kinetics/Spectrophotometry)
- [11] Granite, Site by. "Beer Lambert Law: Transmittance & Absorbance." *Edinburgh Instruments*. Web. Dec. 2019. <https://www.edinst.com/blog/the-beer-lambert-law/>
- [12] *Beer-Lambert Law*. Web. Dec. 2019. <http://life.nthu.edu.tw/~labcjw/BioPhyChem/Spectroscopy/beerslaw.htm>

- [13] Libretexts. "The Beer-Lambert Law." *Chemistry LibreTexts*. Libretexts, 15 Aug. 2019. Web. Nov. 2029. [https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_\(Physical_and_Theoretical_Chemistry\)/Spectroscopy/Electronic_Spectroscopy/Electronic_Spectroscopy_Basics/The_Beer-Lambert_Law](https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Spectroscopy/Electronic_Spectroscopy/Electronic_Spectroscopy_Basics/The_Beer-Lambert_Law)
- [14] Martindale, Jon. "What Is Bluetooth? Here's Everything You Need to Know." *Digital Trends*. Digital Trends, 04 Nov. 2019. Web. Jan. 2020. <https://www.digitaltrends.com/computing/what-is-bluetooth/>
- [15] "How Does Bluetooth Work?" *Explain That Stuff*. 11 Jan. 2020. Web. Jan. 2020. <https://www.explainthatstuff.com/howbluetoothworks.html>
- [16] "Bluetooth." *Wikipedia*. Wikimedia Foundation, Sept. 2019. Web. Jan. 2020. <https://en.wikipedia.org/wiki/Bluetooth>
- [17] Ray, Brian. "Bluetooth Vs. Bluetooth Low Energy (BLE): What's The Difference?" *Bluetooth Vs. Bluetooth Low Energy (BLE): What's The Difference?* Web. March. 2020. <https://www.citationmachine.net/bibliographies/5933c9a7-f4e8-4f23-9dec-3a34b5fb937b>
- [18] *How to Connect to BLE Devices*. Web. March. 2020. <https://evthings.com/doc/tutorials/how-to-connect-to-ble-devices.html>
- [19] "Bluetooth Overview : Android Developers." *Android Developers*. Web. March. 2020. <https://developer.android.com/guide/topics/connectivity/bluetooth>
- [20] "(PDF) Power Consumption Analysis of Bluetooth Low Energy Commercial Products and Their Implications for IoT Applications." *ResearchGate*. Web. April. 2020. https://www.researchgate.net/publication/329401909_Power_Consumption_Analysis_of_Bluetooth_Low_Energy_Commercial_Products_and_Their_Implications_for_IoT_Applications
- [23] "Basic-concepts-of-digital-filtering." *MikroElektronika*. Web. March. 2020. <https://www.mikroe.com/ebooks/digital-filter-design/basic-concepts-of-digital-filtering>
- [24] "Fir-filters." *MikroElektronika*. Web. March. 2020. <https://www.mikroe.com/ebooks/programming-dspic-microcontrollers-in-pascal/fir-filters>
- [25] Hoegger, Bruno, Sara Santos, Areza, Sheldon, Jaka, Umar Muhammad, Indrawan, Jack, Jony, Kevin Hogan, Cezabaltazar, John B, Rui Santos, Ralph Hulslander, Scott, Arthur, Naz, Luis Canossa, Hussainb, Jak, Surender Mohan, Guillermo Amorin, Soap, Nuttaphat Juprempre, Salvatore Fusto, Salvatore Fusto, Amira, Dimas, Pp, Krzysztof, Ppcast, Jajen, Alexander, Bernard Lheureux, Otto Klaasen, Tomas, Nocks, M. Lacombe, Bob, Bob Rader, Belen Hedderich, Alexandros Piperidis, David R, Zakaria, Ace, Sutejo, Carl Peter Laidlaw, Jens, Evan Garland, and Thalís. "ESP32 with LoRa Using Arduino IDE." *Random Nerd Tutorials*. 19 Oct. 2019. Web. Dec. 2019. <https://randomnerdtutorials.com/esp32-lora-rfm95-transceiver-arduino-ide>
- [26] Ali, Zahid. "Introduction to MAX30100." *The Engineering Projects*. 08 May 2019. Web. 09 Dec. 2019. <https://www.theengineeringprojects.com/2019/05/introduction-to-max30100.html>
- [27] Boral, Sayak. "What Is ESP32 and Why Is It Best for IoT Projects?" *IoT Tech Trends*. 25 May 2020. Web. Nov. 2019. <https://www.iottechrends.com/what-is-esp32/>

- [28] "ESP32 TTGO." *ESP32 TTGO*. 01 Jan. 1970. Web. 09 Sept. 2020. <http://esp32-ttgo.blogspot.com>
- [29] "90 ESP32 Projects, Tutorials and Guides with Arduino IDE." 04 Sept. 2020. Web. Dec. 2020.
- [30] Shedboy71. "ESP32 and MAX30100 Heart-rate Monitor Sensor." *ESP32 Learning*. 24 Nov. 2017. Web. Nov. 2019.
- [31] 4, Michael February, Arnoud March 3, Luis Morgado March 6, Curious August 25, PvdBos October 30, Iles January 4, Fbminis January 22, Nick January 23, Sukyong June 6, Gabriel July 3, Gabrielo July 3, Julio August 3, Patrick August 10, Dave January 14, Graciano June 27, Aman February 3, and MMS July 1. "Arduino Nokia 5110 Menu." 01 Mar. 2018. Web. Nov. 2019.
- [32] "ESP32 Projects." *Last Minute Engineers*. Last Minute Engineers. Web. 12 Sept. 2020.
- [33] "Blynk." *Blynk*. Web. 12 Sept. 2020.
- [34] "Build Software Better, Together." *GitHub*. Web. 12 Sept. 2020.

Εικόνες

- [1]"The Wood Library-Museum." *Wood Library Museum*. Web. 17 Sept. 2020.
- [2] Media, OpenSystems, Zhang Feng Is Senior Medical Applications Engineer, and Mart Smith Is Staff Medical Segment Engineer. "Measuring Heart Rate and Blood Oxygen Levels for Portable Medical and Wearable Devices." *Embedded Computing Design*. 16 Feb. 2020. Web. 17 Sept. 2020
- [3] Hinic-Frlog, Sanja. "3.1 – Gas Transport." *Introductory Animal Physiology*. Web. 17 Sept. 2020.
- [4] "Hemoglobin." *Wikipedia*. Wikimedia Foundation, 10 Sept. 2020. Web. 17 Sept. 2020
- [5] "About Heart Valves." *AATB*. 13 Oct. 2020. Web. . 2020.
- [6] Abbasi, Haroon. "The Human Gas-exchanging with a Continuous Flow of Oxygen and Clears the Blood of the Gaseous Waste Product, Carbon Dioxide." *Human Gasexchanging Continuous Flow Oxygen Clears Stock Vector (Royalty Free) 1168744138*. Web. 17 Sept. 2020.
- [7] "Ablation of the AV Node and Implanting of a Pacemaker." *Atrial Fibrillation: Resources for Patients*. 27 Aug. 2020. Web. 17 Sept. 2020.
- [8] "Spectrophotometry." *Wikipedia*. Wikimedia Foundation, 14 Sept. 2020. Web. April. 2020.
- [9] "UV SYSTEM 110W." *Hydragcom*. Web. Jan. 2020.
- [10] *Beer-Lambert Law*. Web. Dec. 2019.

- [11] "Bluetooth." *Tap Bluetooth Low Energy for Universal Point-to-Point Connection with Mobiles*. Web. April. 2020.
- [12] Townsend, Kevin. "Introduction to Bluetooth Low Energy." *Adafruit Learning System*. Web. Jan. 2020.
- [13] Afaneh, Mohammad. "The Ultimate Bluetooth Low Energy (BLE) Guide." *Novel Bits*. 19 May 2020. Web. 17 Sept. 2020.
- [14] "Bluetooth Low Energy (BLE) Beacon Technology Made Simple: A Complete Guide to Bluetooth Low Energy Beacons." *Beaconstac RSS*. Web. 17 Sept. 2020.
- [15] "LoRaWAN Network Server – Actility: Leader in IoT Network Connectivity." *Actility*. Web. Nov. 2019.
- [16] "An Introduction to Filters - Technical Articles." *All About Circuits*. Web. March. 2020.
- [17] "Introduction-iir-filter." *MikroElektronika*. Web. March. 2020.
- [18] Bluejedi, Alvarnal (Alvarnal), Verkehrsrot (Verkehrsrot), Jpy (Jpy), UdLoRa (Ud Lo Ra), Jezd (Jezd), Blem (Manuel Bl), and Tsgchile (Tsgchile). "Big ESP32 + SX127x Topic Part 3." *The Things Network*. 29 July 2020. Web. Nov. 2019.
- [19] . "ESP32 TTGO." *ESP32 TTGO*. 01 Jan. 1970. Web. Nov. 2019.
- [20] "MAX30100 Pulse Oximeter SpO2 and Heart-Rate Sensor Module." *Makerfabs*. Web. Nov. 2019.
- [21] Alldatasheet.com. "MAX30100 Datasheet(9/29 Pages) MAXIM: Integrated Ambient Light Nov. 2019.
- [22] Alldatasheet.com. "MAX30100 Datasheet(8/29 Pages) MAXIM: Integrated Ambient Light Cancellation." *MAX30100 Datasheet(8/29 Pages) MAXIM / Integrated Ambient Light Cancellation*. Web. Nov. 2019.

Παράρτημα Α : Κώδικας

```
#include <SPI.h>
#include <LoRa.h>

#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include "MAX30100_PulseOximeter.h"
#include <BlynkSimpleEsp32_BT.h>

#define BLYNK_PRINT Serial
#define BLYNK_USE_DIRECT_CONNECT

#define SCK 5
#define MISO 19
#define MOSI 27
#define SS 18
#define RST 14
#define DIO0 26
#define REPORTING_PERIOD_MS 1000

#define OLED_SDA 4
#define OLED_SCL 15
#define OLED_RST 16
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define GPIO 35

const int Led_pin = 18;
const int numReadings=10;
const int buttonPin1=13;
const int buttonPin2=12;
const int buttonPin3=21;
uint32_t tsLastReport = 0;
uint32_t last_beat=0;
int readIndex=0;
int average_beat=0;
int average_SpO2=0;
int menuitem = 1;
int page = 1;
bool calculation_complete=false;
bool calculating=false;
volatile boolean up = false;
volatile boolean down = false;
volatile boolean middle = false;
```



```

{
  Serial.println(" ");
  last_beat=millis();
}

void calculate_average(int beat, int SpO2)
{
  if (readIndex==numReadings) {
    calculation_complete=true;
    calculating=false;
    readIndex=0;
  }

  if (not calculation_complete and beat>30 and beat<220 and SpO2>70)
  {

    average_beat = beat;
    average_SpO2 = SpO2;
    readIndex++;
  }
}

void checkIfDownButtonIsPressed()
{
  if (downButtonState != lastDownButtonState)
  {
    if (downButtonState == 0)
    {
      down=true;
    }
    delay(50);
  }
  lastDownButtonState = downButtonState;
}

void checkIfUpButtonIsPressed()
{
  if (upButtonState != lastUpButtonState)
  {
    if (upButtonState == 0) {
      up=true;
    }
    delay(50);
  }
  lastUpButtonState = upButtonState;
}

void checkIfSelectButtonIsPressed()

```

```

{
  if (selectButtonState != lastSelectButtonState)
  {
    if (selectButtonState == 0) {
      middle=true;
    }
    delay(50);
  }
  lastSelectButtonState = selectButtonState;
}

void drawMenu()
{
  //page 1 saw
  if (page==1)
  {
    display.setTextSize(1);
    display.clearDisplay();
    display.setCursor(35,0);
    display.setTextColor(WHITE, BLACK);
    display.print("MAIN MENU");
    display.drawFastHLine(35,7,53,WHITE);

    if (menuitem==1)
    {
      display.setTextColor(BLACK, WHITE);
    }
    else
    {
      display.setTextColor(WHITE, BLACK);
    }
    display.setCursor(0,10);
    display.print("Heart Rate");

    if (menuitem==2)
    {
      display.setTextColor(BLACK, WHITE);
    }
    else
    {
      display.setTextColor(WHITE, BLACK);
    }
    display.setCursor(0,20);
    display.print("SpO2");

    if (menuitem==3)
    {

```

```

display.setTextColor(BLACK, WHITE);
}
else
{
    display.setTextColor(WHITE, BLACK);
}
display.setCursor(0,30);
display.print("Bpm & SpO2");

if (menuitem==4)
{
    display.setTextColor(BLACK, WHITE);
}
else
{
    display.setTextColor(WHITE, BLACK);
}
display.setCursor(0,40);
display.print("Creators");

if (menuitem==5)
{
    display.setTextColor(BLACK, WHITE);
}
else
{
    display.setTextColor(WHITE, BLACK);
}
display.setCursor(0,50);
display.print("Turn off");
display.display();
}

//page 2 saw
else if (page==2)
{
    display.setTextSize(1);
    display.clearDisplay();
    display.setCursor(25, 0);
    display.print("Heart Rate");
    display.drawFastHLine(25,10,60,WHITE);
    display.setCursor(5, 25);
    display.print("BPM : ");
    display.setTextSize(2);
    display.setCursor(5, 40);
    display.print(average_beat);
    display.setCursor(65,40)

```

```

display.setTextSize(2);
display.write(3);
display.display();
}

//page 3 saw
else if (page==3)
{
display.setTextSize(1);
display.clearDisplay();
display.setCursor(50, 0);
display.print("SpO2");
display.drawFastHLine(50,10,25,WHITE);
display.setCursor(5, 25);
display.print("SpO2 : ");
display.setTextSize(2);
display.setCursor(5, 40);
display.print(average_SpO2);
display.setCursor(50, 40);
display.print("%");
display.display();
}

//page 4 saw
else if (page==4)
{
display.setTextSize(1);
display.clearDisplay();
display.setCursor(30, 0);
display.print("BPM & SpO2");
display.drawFastHLine(30,10,60,WHITE);

display.setTextSize(1.5);
display.setCursor(5, 25);
display.print("BPM : ");
display.setTextSize(2);
display.setCursor(40, 20);
display.print(average_beat);
display.setCursor(105,20);
display.setTextSize(2);
display.write(3);

display.setTextSize(1.5);
display.setCursor(5, 50);
display.print("SpO2 : ");
display.setTextSize(2);
display.setCursor(40, 48);
display.print(average_SpO2)

```

```

display.setCursor(95,48);
  display.print("%");
  display.display();

}
else if (page==5)
{
  display.setTextSize(1);
  display.clearDisplay();
  display.setCursor(35, 0);
  display.print("Creators");
  display.drawFastHLine(35,10,50,WHITE);
  display.setTextSize(1.7);
  display.setCursor(5, 20);
  display.print("Karras Nikolaos");
  display.setCursor(25, 30);
  display.print("KAS:513307");
  display.setCursor(5, 40);
  display.print("Fetfatzidis Georgios");
  display.setCursor(25, 50);
  display.print("KAS:512192");
  display.display();
}
}

void setup()
{
  Serial.begin(115200);
  pinMode(buttonPin1, INPUT_PULLUP);
  pinMode(buttonPin2, INPUT_PULLUP);
  pinMode(buttonPin3, INPUT_PULLUP);
  display.begin();
  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(WHITE);

```

```

display.setCursor(0,0);
display.display();

Blynk.setDeviceName("oximeter");
Blynk.begin(auth);

    esp_sleep_enable_ext0_wakeup(GPIO_NUM_35,1);

pinMode(Led_pin,OUTPUT);
pinMode(OLED_RST, OUTPUT);
digitalWrite(OLED_RST, LOW);
delay(20);
digitalWrite(OLED_RST, HIGH);

Wire.begin(OLED_SDA, OLED_SCL);
if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3c, false, false) & !pox.begin() )
{
    Serial.println(F("SSD1306 allocation failed"));
    Serial.println("FAILED");
    for(;;);
}
else
{
    Serial.println("SUCCESS");
}

pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);
pox.setOnBeatDetectedCallback(onBeatDetected);
}

void loop()
{
pox.update();
Blynk.run();

selectButtonState = digitalRead(buttonPin1);
downButtonState = digitalRead(buttonPin2);
upButtonState = digitalRead(buttonPin3);

checkIfDownButtonIsPressed();
checkIfUpButtonIsPressed();
checkIfSelectButtonIsPressed();

if (up && page == 1 ) {
    up = false;
    menuitem--;
}

```

```

if (menuitem==0)
{
    menuitem=5;
}
}

if (down && page == 1) {
    down = false;
    menuitem++;
if (menuitem==6)
{
    menuitem=1;
}
}

if (middle) {
    middle = false;
if (page == 1 && menuitem==1) {
    page=2;
}
else if (page == 2 ) {
    page=1;
}
if (page == 1 && menuitem==2) {
    page=3;
}
else if (page == 3 ) {
    page=1;
}
if (page == 1 && menuitem==3) {
    page=4;
}
else if (page == 4 ) {
    page=1;
}
if (page == 1 && menuitem==4) {
    page=5;
}
else if (page == 5 ) {
    page=1;
}
if (page == 1 && menuitem==5) {

    display.clearDisplay();
    display.drawBitmap(0, 0, heart, 128, 64, WHITE);
    display.display();
}
}

```

```

    delay(2000);
    esp_deep_sleep_start();
  }
}

if ((millis() - tsLastReport > REPORTING_PERIOD_MS) and (not calculation_complete) )
{
  calculate_average(pox.getHeartRate(),pox.getSpO2());
  drawMenu();

  Serial.print("Heart rate:");
  Serial.print(average_beat);
  Serial.print(" SpO2:");
  Serial.print(average_SpO2);

  Blynk.virtualWrite(V3, average_beat);
  Blynk.virtualWrite(V4, average_SpO2);

  tsLastReport = millis();

}
if ((millis()-last_beat>5000)) {
  calculation_complete=false;
  average_beat=0;
  average_SpO2=0;
}
}

```