

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ



**ΤΜΗΜΑ ΜΟΥΣΙΚΩΝ ΣΠΟΥΔΩΝ**

**ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ «ΜΟΥΣΙΚΟΛΟΓΙΑ»**

**ΚΑΤΕΥΘΥΝΣΗ: ΜΟΥΣΙΚΗΣ ΤΕΧΝΟΛΟΓΙΑΣ**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**«Δημιουργία συστήματος για επεξεργασία και ασύρματη μετάδοση  
μουσικής ηλεκτρικής κιθάρας»**

**Γεώργιος Κόκκινος**

**Επιβλέπουσα Καθηγήτρια: Αρετή Ανδρεοπούλου,**

**Επίκουρη Καθηγήτρια Ε.Κ.Π.Α.**

**ΑΘΗΝΑ**

***Οκτώβριος 2019***



## **ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

«Δημιουργία συστήματος για επεξεργασία και ασύρματη μετάδοση μουσικής ηλεκτρικής  
κιθάρας»

**Γεώργιος Κόκκινος**

**A.M.: 170303**

**Επιβλέπουσα: Αρετή Ανδρεοπούλου, Επίκουρη Καθηγήτρια**

**Επιτροπή:**

**Αρετή Ανδρεοπούλου, Επίκουρη Καθηγήτρια**

**Αναστασία Γεωργάκη, Αναπληρώτρια Καθηγήτρια**

**Χριστίνα Αναγνωστοπούλου, Αναπληρώτρια Καθηγήτρια**

## Ευχαριστίες

Θα ήθελα ιδιαιτέρως να ευχαριστήσω την επιβλέπουσα καθηγήτρια μου κα Αρετή Ανδρεοπούλου για την πολύτιμη καθοδήγηση και τη γενικότερη συμβολή της, στην ολοκλήρωση της παρούσας διπλωματικής εργασίας. Ευχαριστώ θερμά, επίσης, την κα Αναστασία Γεωργάκη και τον κ. Γεώργιο Κωστελέτο για την εξαιρετική συνεργασία και τις γνώσεις που μου προσέφεραν καθ' όλη τη διάρκεια της φοίτησης μου στο ΠΜΣ. Τέλος, ένα πολύ μεγάλο ευχαριστώ στον/στις Απόστολο Καρνάβα, Σταυρούλα Κοπελιά, Νεφέλη Οικονόμου, για την πολύτιμη βοήθειά τους.

## ΠΕΡΙΛΗΨΗ

Η διαρκής εξέλιξη της τεχνολογίας επηρεάζει τους ανθρώπους και την καθημερινότητα τους, προσφέροντας νέα εργαλεία και μεθόδους για την εξυπηρέτηση των αναγκών τους. Οι μουσικοί, φυσικά, δεν μένουν ανεπηρέαστοι/ες από αυτήν την αέναη εξέλιξη. Η ανάγκη καλλιτεχνικής έκφρασης οδηγεί τους/τις μουσικούς στην εξερεύνηση νέων, πρωτοπόρων εργαλείων και, σε συνεργασία με τους/τις μηχανικούς του ήχου, στην εξέλιξη των ήδη υπαρχόντων. Η παρούσα εργασία έχει σαν στόχο τη μελέτη, κατασκευή και προγραμματισμό ενός τέτοιου πρωτοπόρου συστήματος χρησιμοποιώντας υλισμικό/λογισμικό ανοικτής προέλευσης. Μέσω της βιβλιογραφικής επισκόπησης, γίνεται μια ιστορική αναδρομή στην εξέλιξη της τεχνολογίας ασύρματης επικοινωνίας Bluetooth, και ειδικότερα στις εφαρμογές της σχετικά με τον ήχο (audio). Περιγράφεται, επίσης, η λειτουργία των συστημάτων πολυφέ ηλεκτρικής κιθάρας και η εξέλιξη τους μέχρι τις μέρες μας. Παράλληλα, γίνεται αναφορά στις τεχνολογίες Open-Source και στην κοινότητα DIY (do it yourself).

Το δεύτερο μέρος της εργασίας αφορά την κατασκευή και τον προγραμματισμό του συστήματος. Για το υλισμικό μέρος, αρχικά, γίνεται η επιλογή των επιμέρους εξαρτημάτων βάσει των χαρακτηριστικών που προσφέρουν. Μετά την προμήθεια των εξαρτημάτων και την σχεδίαση του ηλεκτρονικού κυκλώματος, πραγματοποιείται η κατασκευή του κυκλώματος και η σύνδεσή του με τον μικροϋπολογιστή. Στη συνέχεια, συναρμολογούνται όλα τα επιμέρους κομμάτια στο προστατευτικό κουτί με το σύστημα να αποκτά την τελική του μορφή. Το λογισμικό μέρος περιλαμβάνει το σύνολο του προγραμματιστικού κώδικα που χρειάζεται για τη λειτουργία του συστήματος. Αρχικά, παρουσιάζεται το λειτουργικό σύστημα του μικροϋπολογιστή Raspberry Pi και περιγράφονται οι βασικές εντολές του. Κατόπιν, αναλύεται η γλώσσα C, που αποτελεί την γλώσσα σύνταξης του κώδικα του συστήματος, ενώ μελετώνται οι εφαρμογές της στην ψηφιακή επεξεργασία σήματος και στην ασύρματη μετάδοση ήχου μέσω Bluetooth. Στη συνέχεια, παρουσιάζεται η δομή του κώδικα του συστήματος και χρησιμοποιούνται επεξηγηματικά σχόλια για την ανάλυση της λειτουργίας του. Τέλος, εκθέτονται αποτελέσματα και συμπεράσματα σχετικά με τη χρήση του συστήματος και γίνονται προτάσεις για μελλοντική έρευνα.

Λέξεις – κλειδιά: Ψηφιακή Επεξεργασία Σήματος,, Ασύρματη Μετάδοση Ήχου μέσω Bluetooth, Ηλεκτρονική και Εφαρμογές Ήχου, Γλώσσα Προγραμματισμού C, Τεχνολογία Ανοικτής Προέλευσης

# ABSTRACT

The ongoing evolution of technology affects people and their daily lives, offering new tools and methods to serve their needs. Musicians, also, are affected by the constant evolution of music technology. The need for artistic expression leads musicians to exploring new, pioneering tools and, in collaboration with the sound engineers, to evolve existing ones. This project aims to design, construct and program such a pioneer system using open source hardware/software. Through literature review, we go through a historical flashback to the evolution of Bluetooth wireless communication technology and in particular to its audio applications. Additionally, we examine the operation of the electric guitar multi-effect systems and their evolution until nowadays. Furthermore, we look into Open-Source technologies and the DIY (do it yourself) community.

The second part of the project involves the construction and programming of the system. For the hardware part, initially, the choice of individual components is made on the basis of the characteristics they offer. After the supply of the components and the design of the electronic circuit, the construction of the circuit and the connection with the microcomputer is carried out. Afterwards, the assembly of individual parts takes place in the protective enclosure, and therefore the system comes to its final form. The software part includes all the programmatic code needed to run the system. Initially, the operating system of the Raspberry Pi microcomputer is presented and its basic commands are described. Moreover, C language, which is the programming language of the system code, is being analyzed and especially its applications for digital signal processing and wireless audio transmission over Bluetooth. The structure of the system code is presented below and explanatory comments are used to analyze its operation. Finally, results and conclusions regarding the system are presented and suggestions are made for future research.

**Keywords:** Digital Signal Processing, Wireless Bluetooth Audio Transmission, Analog Electronics for Audio Applications, C Programming Language, Open-Source Technology

# ΠΕΡΙΕΧΟΜΕΝΑ

Εισαγωγή.....	1
1. Βιβλιογραφική Ανασκόπηση.....	4
1.1 Συστήματα Πολυεφέ.....	4
1.2 Πρωτόκολλο Ασύρματης Επικοινωνίας Bluetooth.....	6
1.2.1 Ιστορική Αναδρομή.....	7
1.2.2 Σύνδεση μεταξύ συσκευών Bluetooth.....	8
1.2.3 Επέκταση των δυνατοτήτων μέσω των προφίλ.....	9
1.2.4 Advanced Audio Distribution Profile (A2DP).....	10
1.2.5 Συμπίεση των ηχητικών δεδομένων.....	11
1.3 Τεχνολογία Open source και κοινότητα DIY (do it yourself).....	13
1.3.1 Ιστορική Αναδρομή: Ορισμοί Open Source και Closed Source.....	13
1.3.2 Πλεονεκτήματα ανοικτού κώδικα.....	14
1.3.3 Η κοινότητα DIY.....	14
2. Κατασκευή και προγραμματισμός του συστήματος.....	16
2.1 Κατασκευή υλισμικού συστήματος.....	16
2.1.1 Επιλογή μικροϋπολογιστή για την επεξεργασία του σήματος.....	16
2.1.2 Σύγκριση του Zero W με άλλα διαθέσιμα μοντέλα.....	18
2.1.3 Τα βασικά μέρη του Raspberry Pi Zero W.....	19
2.1.4 Το συνοδευτικό κύκλωμα.....	21
2.1.5 Κατασκευή Υλισμικού Συστήματος.....	25
2.2 Προγραμματισμός του Λογισμικού Συστήματος.....	31
2.2.1 Το λειτουργικό σύστημα του Raspberry Pi.....	31
2.2.2 Επιλογή γλώσσας γραφής προγραμματιστικού κώδικα.....	34
2.2.3 Εγκατάσταση βιβλιοθηκών BCM2835 και Pigpio. Δημιουργία, Σύνταξη και Εκτέλεση αρχείου .....	35
2.2.4 Ασύρματη Bluetooth μετάδοση ήχου μέσω PulseAudio.....	36
2.2.5 Προγραμματισμός Λογισμικού Συστήματος.....	38
3. Συμπεράσματα, περιορισμοί εργασίας και προτάσεις για μελλοντική έρευνα.....	42
4. Επίλογος.....	45
5. Βιβλιογραφία.....	46
Παραρτήματα.....	50





## Εισαγωγή

Στόχος της παρούσας διπλωματικής εργασίας είναι η μελέτη, ο προγραμματισμός και η κατασκευή ενός υλισμικού/λογισμικού (hardware/software) συστήματος, που θα παρέχει στον/στην χρήστη/μουσικό τη δυνατότητα να τροποποιεί τον ήχο ενός μουσικού οργάνου με τα εφέ που επιθυμεί, και να αναπαράγει τον επεξεργασμένο ήχο μέσω ασύρματης επικοινωνίας (Bluetooth) κατευθείαν σε ένα κατάλληλο Bluetooth ηχείο. Το σύνολο των εργαλείων και των συστημάτων υλισμικού και λογισμικού θα είναι ανοιχτής προέλευσης (open source/hardware), κατοχυρώνοντας με αυτόν τον τρόπο την αναπαραγωγιμότητα (reproducibility) της εν λόγω προσπάθειας από τρίτους. Ο λόγος που επιχειρείται η ασύρματη αναπαραγωγή του ήχου μέσω Bluetooth ηχείου και όχι ενσύρματη μέσω ενισχυτή, είναι γιατί προσφέρει επιπλέον ευελιξία στον/στην χρήστη, όσον αφορά την κίνησή του/της στο χώρο κατά τη διάρκεια του παιξίματος. Ταυτόχρονα, με αυτόν τρόπο δεν είναι απαραίτητη η χρήση μεμονωμένου ενισχυτή για την ενίσχυση του ήχου, διότι το Bluetooth ηχείο διαθέτει ενσωματωμένο ενισχυτή. Ωστόσο, θα δίνεται η δυνατότητα στον/στην χρήστη για ενσύρματη αναπαραγωγή του ήχου μέσω ενισχυτή.

Η σύγχρονη κοινωνία δομείται με βάση το κεφάλαιο και οργανώνεται με στόχο τη συσσώρευση και μεγιστοποίησή του. Κατά συνέπεια, το κέρδος και η παραγωγικότητα καθορίζουν οποιαδήποτε πρόοδο συντελείται. Η τεχνολογία χρησιμοποιείται ως μέσο αύξησης του κέρδους και μείωσης του χρόνου παραγωγής, αλλά ταυτόχρονα αποτελεί έναν ανεξάρτητο κλάδο που παράγει προϊόντα προς πώληση. Ειδικότερα, σε ό,τι αφορά την μουσική τεχνολογία το κέρδος των εταιριών τίθεται ως γνώμονας για τη δημιουργία νέων ιδεών και προϊόντων. Τόσο η θεωρητική γνώση που παράγεται, όσο και ο τεχνολογικός εξοπλισμός εμπορευματοποιούνται και δεν είναι προσβάσιμα σε όλα τα μέλη της κοινωνίας. Ο οικονομικός παράγοντας κυριαρχεί στη δημιουργία μουσικής και την καλλιτεχνική έκφραση γενικότερα. Η παρούσα εργασία επιχειρεί να αναδείξει ότι, αν και οι οικονομικοί όροι που θέτει η σύγχρονη κοινωνία είναι περιοριστικοί, ωστόσο δεν αποτρέπουν την ανάπτυξη πρωτοβουλιών, που προωθούν την ελεύθερη πρόσβαση στην τεχνογνωσία. Χαρακτηριστικά παραδείγματα ανάλογων πρωτοβουλιών είναι η κοινότητα DIY (Do It Yourself) και η διάδοση της τεχνολογίας ανοικτής προέλευσης (Open Source-Hardware technology). Η παρούσα εργασία αφορμάται από τις πρωτοβουλίες αυτές και έχει ως στόχο τη δημιουργία ενός ελεύθερα προσβάσιμου, χαμηλού κόστους εργαλείου, που θα δίνει τη δυνατότητα στον/στην χρήστη να το εξελίξει και να το διαμορφώσει σύμφωνα με τις ανάγκες του/της. Επιπλέον, το σύστημα θα διαθέτει ευελιξία όσον αφορά τη χρήση του, τη μεταφορά του και την αυτονομία σε τροφοδοσία ρεύματος. Η ευελιξία

αυτή σε συνδυασμό με την ικανότητα του να μπορεί να αναπαράγει τον ήχο μέσα από ένα ηχείο Bluetooth, θα το καθιστά ιδανικό για μουσικούς που παίζουν στο δρόμο και αναζητούν νέους τρόπους για να εξελίξουν και να πειραματιστούν με τη μουσική τους.

Στο πρώτο κεφάλαιο της εργασίας θα γίνει μια ιστορική αναδρομή στην εξέλιξη της τεχνολογίας ασύρματης επικοινωνίας Bluetooth, και ειδικότερα στις εφαρμογές της σχετικά με τον ήχο (audio), καθώς και στις χρήσεις της στο σύστημα της εργασίας. Στη συνέχεια θα γίνει αναφορά στα συστήματα πολυμέ και ειδικότερα στα πολυμέ ηλεκτρικής κιθάρας, θα μελετηθεί η λειτουργία τους και η εξέλιξή τους στο πέρασμα των χρόνων. Τέλος, θα παρατεθούν πληροφορίες για τον ρόλο της τεχνολογίας ανοικτής προέλευσης (Open Source/Hardware) και της κοινότητας DIY (Do It Yourself) γενικότερα.

Το δεύτερο κεφάλαιο, χωρίζεται στα δύο κύρια μέρη που αφορούν την κατασκευή και τον προγραμματισμό συστήματος. Πιο συγκεκριμένα, το υλισμικό μέρος της εφαρμογής θα περιλαμβάνει τα εξαρτήματα που απαιτούνται για την προετοιμασία του σήματος για την είσοδο στον A/D (Αναλογικό σε Ψηφιακό) μετατροπέα, τη σύνδεση του A/D μετατροπέα με τον μικροεπεξεργαστή όπου θα γίνεται η επεξεργασία του ψηφιακού σήματος, και τη σύνδεση του συστήματος με το ηχείο Bluetooth για την ασύρματη αναπαραγωγή του ήχου. Επίσης, το υλισμικό μέρος περιλαμβάνει το σύνολο των εξαρτημάτων που απαιτούνται για τη δημιουργία του ηλεκτρονικού κυκλώματος, για την ασφαλή στέγαση των μερών του συστήματος, καθώς και για τον εξωτερικό έλεγχο του συστήματος από τον χρήστη. Για το προγραμματισμό του συστήματος, η διαδικασία δημιουργίας χωρίζεται σε τρία σκέλη: στην επικοινωνία του A/D μετατροπέα με τον μικροϋπολογιστή, στη μονάδα επεξεργασίας ήχου, και στην ασύρματη αναπαραγωγή του μέσω ειδικού ηχείου Bluetooth. Για την εισαγωγή των δεδομένων στο σύστημα θα γίνει μελέτη των σχετικών πρωτοκόλλων μετάδοσης ψηφιοποιημένου σήματος από τον A/D μετατροπέα στον μικροϋπολογιστή, ώστε να επιτευχθεί η καλύτερη δυνατή ποιότητα σήματος βάσει των δυνατοτήτων των εξαρτημάτων που θα επιλεγθούν. Για τη μονάδα επεξεργασίας ήχου θα γίνει ανάπτυξη διαφόρων αλγορίθμων που θα διαμορφώνουν το σήμα δίνοντας τη δυνατότητα στον/στην μουσικό/χρήστη να δημιουργήσει, τα δικά του/της επιθυμητά ηχητικά εφέ (παραμόρφωση, καθυστέρηση, βάθος, κλπ.) ανάλογα με τις ανάγκες του/της. Για την ασύρματη αναπαραγωγή θα γίνει διερεύνηση των κατάλληλων πρωτοκόλλων και προφίλ ώστε να διασφαλιστεί η ποιότητα του ηχητικού σήματος, περιορίζοντας, κατά το μέτρο του δυνατού, την ύπαρξη θορύβου από εξωγενείς παράγοντες, και καθυστερήσεων (latency) στο τελικό σήμα σε σχέση με το αρχικό.

Στο τρίτο κεφάλαιο παρουσιάζονται τα συμπεράσματα, οι περιορισμοί της εργασίας, καθώς και προτάσεις και βελτιώσεις για μελλοντική έρευνα. Τέλος, δίνεται η βιβλιογραφία που χρησιμοποιήθηκε και τα παραρτήματα που συμπεριλαμβάνουν τα τεχνικά φυλλάδια, τα σχεδιαγράμματα και τον κώδικα.

# 1. Βιβλιογραφική Ανασκόπηση

## 1.1 Συστήματα Πολυεφέ

Ένα σύστημα εφέ είναι μια ηλεκτρονική συσκευή, αναλογική ή ψηφιακή, που χρησιμοποιείται για να αλλάξει τον ήχο ενός μουσικού οργάνου ή μιας ηχητικής πηγής. Οι μεταβολές ενός εφέ πάνω σε ένα ηχητικό σήμα μπορούν να επηρεάσουν οποιοδήποτε από τα εγγενή χαρακτηριστικά του. Έτσι, υπάρχουν εφέ που μεταβάλλουν την ένταση του ήχου (tremolo, compressor), κάποια που μεταβάλλουν το αρμονικό περιεχόμενο (overdrive, distortion), προσθέτουν ή αφαιρούν αρμονικό περιεχόμενο (envelope filter, graphic equalizer), άλλα που μεταβάλλουν την κεντρική συχνότητα και τις αρμονικές (pitch shifter), και άλλα που επηρεάζουν τον χρόνο (delay, reverb).



*Εικόνα 1: Εφέ κιθάρας σε μορφή πετάλ.  
(<https://www.proaudioeurope.com/outboard/delays-reverbs/eff-ect-pedals/bae-hot-fuzz-guitar-pedal.html>)*

Στην ορολογία της μηχανικής ήχου, το σήμα που δεν έχει υποστεί μεταβολές από εφέ αποκαλείται “dry”, ενώ εκείνο που έχει υποστεί μεταβολές από ένα ή περισσότερα εφέ θεωρείται “wet”. Στον κόσμο της ηλεκτρικής κιθάρας, συνηθίζεται τα εφέ να είναι υπό την μορφή πεταλιών, δηλαδή μεταλλικών, συνήθως, κουτιών όπου μέσα τους υπάρχει ηλεκτρονικό κύκλωμα που παράγει το ανάλογο εφέ. Ο έλεγχος των παραμέτρων των εφέ γίνεται με διάφορους τρόπους, μερικοί εκ των οποίων είναι τα ποτενσιόμετρα (δηλαδή

μεταβλητές αντιστάσεις) και τα κουμπιά επαφών χειρός ή ποδιού. Τα εφέ μπορούν να συνδυαστούν μεταξύ τους και να δημιουργήσουν συστήματα πολυεφέ. Ένα σύστημα πολυεφέ περιλαμβάνει πλήθος διαφορετικών εφέ που δίνουν τη δυνατότητα επίτευξης διαφόρων ηχητικών αποτελεσμάτων.

Τα συστήματα πολυεφέ διακρίνονται σε διάφορες μορφές. Με τη χρήση πολλών πεταλιών, κατάλληλης βάσης στήριξης και τροφοδοσίας, δημιουργείται μια “πεταλιέρα” που μπορεί να διαφοροποιείται κάθε φορά ανάλογα με τα μέρη που την απαρτίζουν. Τα επιμέρους εφέ-πετάλια δύναται να είναι είτε αναλογικά, είτε ψηφιακά, είτε μια μίξη των δύο τύπων. Ταυτόχρονα υπάρχουν έτοιμες ψηφιακές πεταλιέρες που διαθέτουν τα δικά τους εφέ σε προκαθορισμένη σειρά, δίνοντας όμως παράλληλα τη δυνατότητα για επιπλέον μορφοποίηση. Μια άλλη πιο συμπαγής και υψηλού κόστους μορφή συναρμολογούμενων πεταλιέρων, είναι τα πολυεφέ τύπου Rackmount. Πρόκειται για μονάδες εφέ που τοποθετούνται συρταρωτά σε μεταλλικό πλαίσιο, και είναι ιδιαίτερα διαδεδομένα για συναυλίες και στούντιο ηχογραφήσεων λόγω της στιβαρής κατασκευής και της υψηλής ποιότητας ηχητικού αποτελέσματος. Τα τελευταία χρόνια με την εξάπλωση των προσωπικών ηλεκτρονικών υπολογιστών υψηλών δυνατοτήτων και των συστημάτων daw (digital audio workstation), είναι πολύ διαδεδομένα τα ψηφιακά Plug-ins, δηλαδή ψηφιακές προσομοιώσεις εφέ. Τα Plug-ins εφέ διαθέτουν ισχυρούς αλγορίθμους που διαρκώς εξελίσσονται και λόγω της ευκολίας χρήσης τους δίνουν τη δυνατότητα για πειραματισμό και δοκιμές. Ένα από τα μειονεκτήματά τους αποτελεί η απαραίτητη ύπαρξη ηλεκτρονικού υπολογιστή, για αυτό και είναι πιο συνηθισμένη η χρήση τους σε περιβάλλον στούντιο (Effects Unit, n.d.).



Εικόνα 2: Πολυεφέ κιθάρας σε μορφή πεταλιέρας. (<https://www.long-mcquade.com/23265/Guitars/Guitar-Effects/Line-6/Guitar-Multi-Effects-Processor.htm>)

Τα εφέ κιθάρας αριθμούν περίπου εβδομήντα χρόνια ιστορίας. Το πρώτο εφέ που παράχθηκε ως αυτόνομο σύστημα ήταν το τρέμολο “Trem Trol 800” της εταιρίας DeArmond το 1948. Η παραγωγή όμως αυτόνομων εφέ ήταν ακόμα σε πολύ πρώιμο στάδιο λόγω των περιορισμών της τεχνολογίας. Η ενίσχυση του σήματος γινόταν ακόμα μόνο με τη χρήση λυχνιών κενού, οι οποίες καταλάμβαναν μεγάλο όγκο, είχαν μικρή σχετικά διάρκεια ζωής και υψηλό κόστος. Μέχρι την δεκαετία του 1960, τα εφέ ήταν συνήθως ενσωματωμένα σε ενισχυτές κιθάρας. Δύο πολύ σημαντικές εφευρέσεις της ηλεκτρονικής, το τρανζίστορ και λίγο αργότερα το ολοκληρωμένο κύκλωμα, κατέστησαν δυνατή την παραγωγή αυτόνομων εφέ τα οποία δεν καταλάμβαναν πολύ χώρο και άντεχαν σε μετακινήσεις. Έτσι, το πρώτο εμπορικό εφέ με τρανζίστορ κατασκευάστηκε το 1962 από την εταιρία Maestro, είχε όνομα μοντέλου “Fuzz Tone” και ήταν εφέ παραμόρφωσης. Το πρώτο εμπορικό εφέ που χρησιμοποίησε ολοκληρωμένο κύκλωμα, και ειδικότερα τελεστικό ενισχυτή (operational amplifier) για την ενίσχυση του σήματος, ήταν το Distortion+ της εταιρίας MXR το οποίο πρωτοκυκλοφόρησε το στις αρχές της δεκαετίας του 1970 (A Brief History of Effects Pedals, 2017).

Η εξάπλωση των ψηφιακών ηλεκτρονικών δεν άφησε ανεπηρέαστο τον χώρο των εφέ κιθάρας. Ήδη από την δεκαετία του 1980, παρουσιάστηκαν στην αγορά τα πρώτα συστήματα πολυεφέ τα οποία έκαναν χρήση ψηφιακών κυκλωμάτων. Πρωτοπόρες εταιρίες σε αυτόν τον τομέα ήταν η Boss, με την πεταλίερα-πολυεφέ SCC700 που βγήκε στην αγορά το 1982, δίνοντας τη δυνατότητα χειρισμού 7 διαφορετικών προγραμματισμένων εφέ μέσω ενός ελεγκτή. Από την άλλη, η Roland το 1987 με το μοντέλο GP-8 προσέφερε στον/ην χρήστη/ρια ένα πλήρες σετ ελεγχόμενων εφέ σε μια συσκευή. Τις επόμενες δεκαετίες τα ψηφιακά πολυεφέ συνέχισαν να εξελίσσονται και να βελτιώνονται, διαθέτοντας περισσότερες ηχητικές επιλογές και καλύτερη ποιότητα ήχου χάρη στους δυνατότερους επεξεργαστές και τους εξελιγμένους αλγόριθμους (A Brief History of Effects Pedals, 2017).

## **1.2 Πρωτόκολλο Ασύρματης Επικοινωνίας Bluetooth**

Το Bluetooth αποτελεί ένα χαμηλού κόστους και ενεργειακής κατανάλωσης πρωτόκολλο ασύρματης επικοινωνίας για προσωπικά δίκτυα υπολογιστών (Wireless Personal Area Networks, WPAN) μικρών αποστάσεων. Σχεδιάστηκε με σκοπό τη δημιουργία ασύρματης επικοινωνίας μεταξύ διαφόρων φορητών και μη ηλεκτρονικών συσκευών όπως κινητών τηλεφώνων, φορητών ηλεκτρονικών υπολογιστών, ακουστικών, κτλ. (Floros, Andreas & Tatlas, Nicolas-Alexander & Mourjopoulos, John., 2006). Το πρωτόκολλο Bluetooth, όπως και άλλα

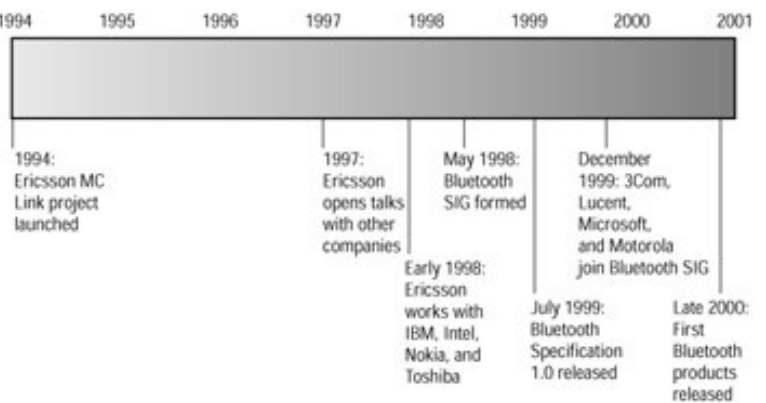
πρωτόκολλα ασύρματης επικοινωνίας (Wi-Fi, ZigBee), λειτουργεί στο αδέσμευτο φάσμα συχνοτήτων των 2,4 GHz (γιγαχέρτζ). (Bluetooth Basics. n.d. Retrieved from:

<https://learn.sparkfun.com/tutorials/bluetooth-basics>).

### 1.2.1 Ιστορική Αναδρομή

Η σύλληψη της ιδέας για την ύπαρξη ενός πρωτοκόλλου ασύρματης επικοινωνίας συνδέεται άμεσα με μια άλλη μεγάλη τεχνολογική καινοτομία των τελών του 20ου αιώνα, αυτή του κινητού τηλεφώνου. Η τεράστια εξάπλωση του συγκεκριμένου μέσου επικοινωνίας κατά τη δεκαετία του 1990 συνδυάστηκε με

τη χρήση ακουστικών τύπου hands-free για τη διευκόλυνση του χρήστη. Το επόμενο βήμα ήταν η ανάπτυξη ενός τρόπου ασύρματης δικτύωσης μεταξύ ετερογενών συσκευών, για παράδειγμα η σύνδεση ενός κινητού τηλεφώνου και ακουστικών.



Εικόνα 3: Η εξέλιξη του Bluetooth μέχρι τα τέλη του 2000. (Miller, 2001)

Το 1994 η Σουηδική εταιρεία κατασκευής κινητών τηλεφώνων Ericsson ξεκίνησε την έρευνα για τη δημιουργία ενός πρωτοκόλλου που θα επέτρεπε την ασύρματη επικοινωνία μεταξύ των κινητών της και των ακουστικών hands free. Η έρευνα απέκτησε δυναμική όταν οι μηχανικοί της εταιρίας συνειδητοποίησαν ότι μπορούσαν να χρησιμοποιήσουν μια μπάνα συχνοτήτων για την οποία δεν χρειάζονταν ειδική άδεια (Αδέσμευτο φάσμα συχνοτήτων). Κατόπιν, αποφασίστηκε η συνεργασία με άλλες εταιρίες (Nokia, IMB, κτλ) ώστε να συμβάλουν στη δημιουργία ενός ευρύτερου πρωτοκόλλου επικοινωνίας με πολύ περισσότερες εφαρμογές. Έτσι, το 1998 δημιουργήθηκε ο οργανισμός Bluetooth SIG, παίρνοντας το όνομά του από το αντικείμενο για το οποίο δημιουργήθηκε, το πρωτόκολλο Bluetooth. Μετά από περίπου ένα χρόνο, τον Ιούλιο του 1999, η πρώτη έκδοση του πρωτοκόλλου Bluetooth (1.0) δόθηκε στη δημοσιότητα. Το Bluetooth απέκτησε τεράστια απήχηση σε πολύ μικρό χρονικό διάστημα, με τα μέλη του οργανισμού Bluetooth SIG να ανέρχονται σε 2000 μετά από μόλις ένα χρόνο από την κυκλοφορία της πρώτης έκδοσης Bluetooth. Η ύπαρξη ενός πλήρως λειτουργικού πρωτοκόλλου ασύρματης επικοινωνίας, χαμηλής κατανάλωσης και μικρής εμβέλειας ήταν

πλέον γεγονός, με τα πρώτα προϊόντα που το υποστήριζαν να κάνουν την εμφάνισή τους στα τέλη του 2000 (Miller, 2001).

<i>Bluetooth version</i>	<i>Features</i>	<i>Speed/Data Rate (in theory and without obstacles)</i>	<i>Range (in theory and without obstacles)</i>
V1.0	Basic rate	<b>1Mbps</b>	<b>33ft (10m)</b>
V2.1	Basic rate SSP – Secure Simple Pairing EDR – Enhanced Data Rate	<b>3Mbps</b>	<b>100ft (30m)</b>
V3.0	Basic rate EDR SSP HS – High speed Protocol	<b>24Mbps</b>	<b>100ft (30m)</b>
V4.0, V4.1, V4.2	Basic Rate EDR SSP HS LE – Low Energy Protocol	<b>24Mbps</b>	<b>200ft (60m)</b>
V5.0	Basic Rate EDR SSP HS LE – Low Energy Protocol IoT – Internet of Things Protocol	<b>48Mbps</b>	<b>800ft (200m)</b>

Εικόνα 4: Οι εκδόσεις του Bluetooth μέχρι σήμερα και τα χαρακτηριστικά τους. (Longman, 2019).

### 1.2.2 Σύνδεση μεταξύ συσκευών Bluetooth

Για τη δημιουργία ενός δικτύου μεταξύ Bluetooth συσκευών (ή “piconet” στην ορολογία του Bluetooth), χρειάζεται ο ορισμός της μιας συσκευής ως κεντρικού κόμβου (master node). Ο κεντρικός κόμβος είναι υπεύθυνος για τις μεταβολές της δικτυακής τοπολογίας. Οι υπόλοιπες συσκευές συνδέονται ως υποκόμβοι (slave node) με τον κεντρικό κόμβο (Floros et al., 2006). Στη συνέχεια για να ολοκληρωθεί με επιτυχία μια σύνδεση θα πρέπει να γίνουν τρία βήματα με την εξής σειρά (Bluetooth Basics. n.d. Retrieved from:

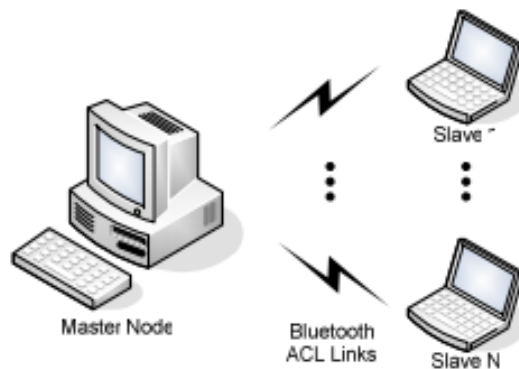
<https://learn.sparkfun.com/tutorials/bluetooth-basics>).

Βήμα 1ο: Ερώτηση (Inquiry). Οι συσκευές δεν διαθέτουν κανένα στοιχείο η μία για την άλλη. Για αυτό το λόγο η μία συσκευή πρέπει να στείλει ερωτήματα στην άλλη για να μπορέσει να την “ανακαλύψει”. Έτσι, μια συσκευή στέλνει ερωτήματα και όποια άλλη συσκευή λαμβάνει τα μηνύματα, απαντάει στέλνοντάς της πληροφορίες όπως διεύθυνση, όνομα συσκευής κτλ.



Βήμα 2ο: Δημιουργία Σύνδεσης (Paging). Κατά τη δημιουργία σύνδεσης οι δύο συσκευές πρέπει οπωσδήποτε να γνωρίζουν η μία τη διεύθυνση της άλλης (την οποία αποκτούν κατά το προηγούμενο βήμα).

Βήμα 3ο: Σύνδεση (Connection). Η κατάσταση της σύνδεσης επέρχεται όταν πραγματοποιηθεί με επιτυχία η δημιουργία σύνδεσης. Όταν οι συσκευές βρίσκονται στην κατάσταση σύνδεσης, μπορούν να συμμετέχουν ενεργά στην επικοινωνία ή να βρίσκονται σε κατάσταση χαμηλής ενεργειακής κατανάλωσης (sleep mode).



Εικόνα 5: Ένα δίκτυο (piconet) μεταξύ Bluetooth συσκευών. (Floros et al., 2006)

### 1.2.3 Επέκταση των δυνατοτήτων μέσω των προφίλ

Μεγάλο πλεονέκτημα του Bluetooth είναι η δυνατότητα να εκτείνει τις πιθανές του χρήσεις μέσω των προφίλ (Floros et al., 2006). Το προφίλ αποτελεί ένα σύνολο προδιαγραφών που αφορούν μια συγκεκριμένη πτυχή της ασύρματης επικοινωνίας μεταξύ συσκευών. Για να γίνει χρήση μιας πτυχής του Bluetooth από μια συσκευή, θα πρέπει η συσκευή αυτή να διαθέτει το αντίστοιχο προφίλ. Έτσι, μια συσκευή χρησιμοποιεί την ασύρματη επικοινωνία Bluetooth ανάλογα με τα προφίλ που διαθέτει. Υπάρχουν διάφορα προφίλ, με το καθένα από αυτά να είναι απαραίτητο για να πραγματοποιηθεί η λειτουργία που αφορά. Τα πιο διαδεδομένα είναι τα εξής (Kunde, 2014):

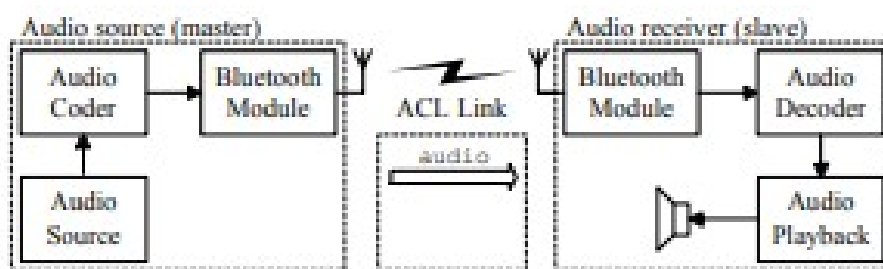
Advanced Audio Distribution Profile (A2DP)	Hands-Free Profile (HFP)
Audio/Video Remote Control Profile (AVRCP)	Human Interface Device Profile (HID)
Basic Imaging Profile (BIP)	Headset Profile (HSP)
Basic Printing Profile (BPP)	Intercom Profile (ICP)
Common ISDN Access Profile (CIP)	LAN Access Profile (LAP)
Cordless Telephony Profile (CTP)	Object Push Profile (OPP)
Device ID Profile (DIP)	Personal Area Networking Profile (PAN)
Dial-up Networking Profile (DUN)	Phone Book Access Profile (PBAP, PBA)
Fax Profile (FAX)	Serial Port Profile (SPP)
File Transfer Profile (FTP)	Service Discovery Application Profile (SDAP)
Generic Audio/Video Distribution Profile (GAVDP)	SIM Access Profile (SAP, SIM, rSAP)
Generic Object Exchange Profile (GOEP)	Synchronisation Profile (SYNCH)
Hard Copy Cable Replacement Profile (HCRP)	Video Distribution Profile (VDP)
Hands-Free Profile (HFP)	Wireless Application Protocol Bearer (WAPB)

#### **1.2.4 Advanced Audio Distribution Profile (A2DP)**

Το προφίλ A2DP καθορίζει τις διαδικασίες και τα πρωτόκολλα που χρειάζονται για την μετάδοση ηχητικού περιεχόμενου , σε μονοφωνικά ή στερεοφωνικά κανάλια. (Bluetooth SIG, 2012). Λειτουργεί πάνω στη βάση του προφίλ Generic Audio/Video Distribution Profile (GAVDP). Μια τυπική χρήση του είναι η μετάδοση μουσικής από ένα σύστημα αναπαραγωγής σε ένα ασύρματο στέρεο ηχείο. Το προφίλ αυτό καθορίζει δύο ρόλους στις ηχητικές συσκευές, έναν της πηγής (Source) και έναν της υποδοχής (Sink). Μία συσκευή αποτελεί A2DC πηγή όταν λειτουργεί ως πηγή ψηφιακού ηχητικού σήματος. Αντίθετα, μια συσκευή αποτελεί A2DP υποδοχή όταν λειτουργεί ως υποδοχή αυτού του ψηφιακού ηχητικού σήματος. Επειδή το προφίλ A2DP δεν μπορεί να διαχειριστεί εντολές απομακρυσμένου ελέγχου, συνήθως οι συσκευές χρησιμοποιούν ταυτόχρονα και το προφίλ Audio/Video Remote Control Profile (AVRCP) που υποστηρίζει την χρήση τέτοιων εντολών (Bluegiga Technologies, 2014).

Είναι ευρέως γνωστό πως μια υψηλής ποιότητας ψηφιακή αναπαραγωγή ήχου, πραγματοποιείται σε bitrates της τάξης των Mbps (περίπου 1.4Mbps για ποιότητα στέρεο CD) (Floros, Koutroubas, Tatlas, & Mourjopoulos, 2002). Όμως, το πρωτόκολλο Bluetooth έχει περιορισμένο εύρος ζώνης (bandwidth) το οποίο δεν μπορεί να εξυπηρετήσει σταθερή μετάδοση για τέτοια ποσότητα ψηφιακών δεδομένων το δευτερόλεπτο. Αναφορικά, το μέγιστο εύρος ζώνης που μπορεί να διατεθεί μέσω του προφίλ A2DP ανέρχεται στα 728kbps, το οποίο είναι περίπου το μισό της ποιότητας του στέρεο CD (1.4Mbps). Για αυτό τον λόγο είναι απαραίτητο να υπάρξει συμπίεση των δεδομένων ήχου πριν την ασύρματη μετάδοση. Όπως και

σε άλλες εφαρμογές ήχου, η αναλογία της συμπίεσης των δεδομένων είναι ένας συμβιβασμός μεταξύ της επιθυμητής ποιότητας ήχου και του διαθέσιμου εύρους ζώνης. Στην παρακάτω εικόνα παρουσιάζονται τα στάδια από τα οποία διέρχονται τα δεδομένα ήχου, από την πηγή μέχρι την αναπαραγωγή τους (ACL Link – Asynchronous Connection-Less Link, είναι ένα συμπληρωματικό πρωτόκολλο που χρησιμοποιείται κατά την ασύρματη μετάδοση δεδομένων, όταν είναι σημαντική η ακεραιότητα των δεδομένων που μεταδίδουμε, όπως είναι δηλαδή η μετάδοση δεδομένων ήχου προς αναπαραγωγή).



Εικόνα 6: Η διαδρομή που ακολουθούν τα δεδομένα ήχου. (Floros et al., 2002)

### 1.2.5 Συμπίεση των ηχητικών δεδομένων

Το προφίλ A2DP υποστηρίζει ένα πλήθος συμπίεσεων για μονοφωνικές, στερεοφωνικές και πολυκάναλες εφαρμογές ήχου. Απαραίτητη προϋπόθεση για την υποστήριξη μιας κωδικοποίησης είναι η ποιότητα της κωδικοποίησης να ξεπερνάει τα 96kbps (Floros et al., 2002). Μερικές από τις κωδικοποιήσεις που υποστηρίζονται είναι οι ISO/MPEG (MP3, AAC, κτλ.) καθώς και οι (ATRAC, WMA, Dolby Digital, κτλ.). Η μόνη κωδικοποίηση που είναι υποχρεωτική σε κάθε συσκευή με Bluetooth προφίλ A2DP, είναι η Subband codec (SBC) ή Low complexity codec (Hoene & Hyder, 2010). Έτσι, κάθε συσκευή Bluetooth πρέπει να υποστηρίζει τουλάχιστον το προφίλ A2DP και την κωδικοποίηση SBC. Αυτό γίνεται ώστε οι συσκευές ανεξάρτητα από το πόσο σύγχρονες είναι και τι εκδόσεις Bluetooth διαθέτουν, να μπορούν να συνδεθούν η μία με την άλλη, και να μεταφέρουν ηχητικά δεδομένα.

Η κωδικοποίηση SBC χαρακτηρίζεται ως “lossy” συμπίεση, δηλαδή για να μειωθεί το μέγεθος του αρχείου αφαιρούνται δεδομένα που συνήθως θεωρούνται περιττά. Σε ότι αφορά τον ήχο, αυτά τα δεδομένα είναι υψηλές συχνότητες άνω των 10kHz τις οποίες το ανθρώπινο αυτί συνήθως δεν μπορεί να αντιληφθεί. Οι συσκευές με SBC κωδικοποίηση μπορούν να δεχθούν στην είσοδο τους 16-bit PCM (Pulse Code Modulation) ηχητικό σήμα, με συχνότητα δειγματοληψίας ίση με 16, 32, 44.1 ή 48kHz (Hoene & Hyder, 2009). Με την SBC μπορεί να επιτευχθεί σχετικά υψηλός ρυθμός μετάδοσης δεδομένων ως και 345kbps για στέρεο (το MP3 έχει ως μέγιστο ρυθμό τα 320kbps), όπου λόγω του ανώτερου αλγορίθμου χάνονται ελάχιστα, αντιληπτά από το ανθρώπινο αυτί, δεδομένα ήχου. Γενικά, η κωδικοποίηση SBC διατηρεί μια ισορροπία μεταξύ της ποιότητας του ηχητικού σήματος και της κατανάλωσης ενέργειας και ισχύος επεξεργαστή της συσκευής, που απαιτούνται για να λειτουργεί.

Εκτός των κωδικοποιήσεων που αναφέρθηκαν πιο πάνω και εντάσσονται στο προφίλ A2DP, υπάρχουν και άλλες που μπορούν να χρησιμοποιηθούν από συσκευές Bluetooth. Οι κωδικοποιήσεις αυτές ανήκουν σε συγκεκριμένες εταιρίες και απαιτούν ειδικές άδειες και τέλη για τα δικαιώματά τους. Μερικές από αυτές είναι:

- LDAC, της εταιρίας Sony
- aptX, aptX HD, aptX Low Latency και aptX Adaptive, της εταιρίας Qualcomm
- Samsung HD, Samsung Scalable και Samsung UHQ-BT, της εταιρίας Samsung
- HWA LHDC, της εταιρίας Huawei

Για τις ανάγκες της παρούσας εργασίας χρησιμοποιήθηκε η κωδικοποίηση SBC για διάφορους λόγους. Αρχικά, οι συσκευές Bluetooth που χρησιμοποιούνται είναι συμβατές με την SBC (αφού υπάρχει υποχρεωτικά με το προφίλ A2DP). Προσφέρει ικανοποιητική ποιότητα ήχου (~320kbps), χωρίς να έχει μεγάλη κατανάλωση ενέργειας και υπολογιστικής ισχύος με συνέπεια λιγότερη καθυστέρηση (latency) του ηχητικού σήματος. Τέλος, αποτελεί τεχνολογία ανοικτής προέλευσης (open-source) για τη χρήση της οποίας δεν απαιτούνται άδειες, και ταυτόχρονα ο κώδικας της είναι εύκολα προσβάσιμος και μπορεί να αλλαχτεί ανάλογα με τις ανάγκες της κάθε εφαρμογής.

### 1.3 Τεχνολογία Open source και κοινότητα DIY (do it yourself)

#### 1.3.1 Ιστορική Αναδρομή: Ορισμοί Open Source και Closed Source

Με τον όρο *open source software*, ή αλλιώς *λογισμικό ανοιχτού κώδικα*, αναφερόμαστε στο λογισμικό που, μέσω άδειας, παρέχεται και ο προγραμματιστικός κώδικάς του που επιτρέπει να χρησιμοποιηθεί, να μελετηθεί, να επεξεργαστεί, να βελτιωθεί και να επανεκδοθεί από τον χρήστη. Η πρώτη ονομασία, ωστόσο, ήταν “*ελεύθερο λογισμικό*” (*free software*) και δόθηκε από τον ακαδημαϊκό και δημιουργό της πλατφόρμας GNU, Richard Stallman, στις αρχές της δεκαετίας του ‘80. Μέσω της ανάλυσης του όρου είναι προφανές ότι ο χρήστης μπορεί να διαχειριστεί τον κώδικα ελεύθερα, με μοναδικούς περιορισμούς να επαναπροωθήσει την τροποποιημένη έκδοσή του στην κοινότητα και να δημοσιευτεί ο πηγαίος κώδικας. Κάτι τέτοιο επικυρώνεται με συμφωνία των δύο μερών μέσω της άδειας (license). Ο δημιουργός μπορεί επίσης, εάν επιθυμεί να υποβάλλει κάποιο αντίτιμο για την παροχή του κώδικα, παρόλο που η πλειοψηφία των ελεύθερων λογισμικών είναι όντως δωρεάν. Το “ελεύθερο λογισμικό” ήταν για τον Stallman ένα ζήτημα φιλοσοφίας και αξιών, μία μορφή κοινωνικοπολιτικού κινήματος προς όφελος όλων των χρηστών (Stallman, 2015).

Ωστόσο, η διττή έννοια του “free” στα αγγλικά, που μπορεί να μεταφραστεί ως ελεύθερο αλλά και ως δωρεάν, σύγχυσε τους χρήστες αλλά και την αγορά, με αποτέλεσμα το πρότυπο αυτό να χαρακτηριστεί ως μη ανταγωνιστικό και μη εμπορικό. Στα τέλη του ‘90 προτάθηκε και εφαρμόστηκε -εν μέρει- από τις εταιρείες ανάπτυξης λογισμικού η τροποποίηση του όρου σε “*λογισμικό ελεύθερου κώδικα*”, έτσι ώστε να διατηρείται μεν η ελευθερία πρόσβασης στον κώδικα αλλά με επιπρόσθετο κίνητρο το κέρδος του δημιουργού. Μερικά τέτοια παραδείγματα είναι το LibreOffice, το GNU Image Manipulation Program, ο Mozilla Firefox, ο Apache και τα Linux (Open-source Software, n.d.).

Αντίθετα με τους ανοιχτούς, οι κλειστοί (closed source) μη προσβάσιμοι κώδικες, δεν μπορούν να αντιγραφούν και να υποστούν επεξεργασία από κανέναν πέραν του ατόμου ή του οργανισμού/εταιρείας που τον δημιούργησε, τον διαχειρίζεται και κατέχει τα πλήρη δικαιώματα. Ο κώδικας είναι δηλαδή νόμιμη ιδιοκτησία τους, κάτι που εξασφαλίζεται και εδώ μέσω της άδειας (license) όπου απαιτούν να υπογράψει κάθε χρήστης που επιθυμεί να χρησιμοποιήσει το λογισμικό αυτό. Ο χρήστης, είναι υποχρεωμένος να μην εκμεταλλευτεί τον κώδικα, χωρίς την έγκριση του δημιουργού του. Κάποια παραδείγματα λογισμικών κλειστού

κώδικα είναι: Microsoft Windows, macOS, Avast, iTunes, Adobe Flash, Google Earth, Opera, Skype, κ.ά. (Proprietary Software, n.d.).

### **1.3.2 Πλεονεκτήματα ανοικτού κώδικα**

Η χρήση του μοντέλου ανοιχτού κώδικα ωφελεί και τους κοινούς χρήστες, που δεν είναι προγραμματιστές. Διαδικασίες καθημερινές όπως η συνομιλία, η αναζήτηση ή το streaming, είναι δεδομένα που μεταφέρονται σε ένα απομακρυσμένο δίκτυο υπολογιστών που χρησιμοποιούν λογισμικό ανοιχτού κώδικα ώστε να μετατραπούν και να κατευθυνθούν στις συσκευή του χρήστη.

Σε επίπεδο προγραμματιστικής χρήσης, τα οφέλη της χρήσης του ανοικτού κώδικα είναι πολλά. Πέρα από τον οικονομικό παράγοντα, που συμφέρει ατομικά τους προγραμματιστές και συνολικά τους οργανισμούς, προτιμάται λόγω της ασφάλειας που παρέχει. Επειδή ο ανοιχτός κώδικας είναι προσβάσιμος και ορατός από όλους, είναι ευκολότερο να διαπιστωθούν δυσλειτουργίες, λάθη και παραλείψεις του αρχικού δημιουργού απ' ότι σε κρυφούς κώδικες. Μοιράζονται ακόμη απόψεις, προβληματισμοί και λύσεις μεταξύ της κοινότητας, και καταλήγει ο κώδικας στη βέλτιστη μορφή του. Ένας άλλος σημαντικός λόγος είναι η εξασφάλιση σταθερότητας. Οι προγραμματιστές που επιλέγουν τέτοιου είδους ανοιχτά πρότυπα για να δουλέψουν, δεν ανησυχούν για το εάν κάποια συγκεκριμένα χαρακτηριστικά ενός κώδικα χαθούν επειδή ο δημιουργός τροποποίησε την πηγή του. Τέλος, σημαντική είναι η δυνατότητα εξάσκησης των χρηστών πάνω στον κώδικα με σκοπό την βελτίωση των προγραμματιστικών τους ικανοτήτων (What is Open-Source?, n.d.).

### **1.3.3 Η κοινότητα DIY**

Αν μελετήσουμε την φιλοσοφία του DIY στη βάση του, διαπιστώνουμε ότι έχει περισσότερο τη μορφή ιδεολογίας και κουλτούρας. Μίας κουλτούρας που προωθεί την συνεργασία, την ελεύθερη διακίνηση ιδεών μεταξύ των μελών της και τη δημιουργία ιδιόκτητων προϊόντων, χωρίς την εξάρτηση από οργανισμούς και εταιρείες. Το πλαίσιο της ηθικής του DIY, ο κύριος σκοπός είναι η δημιουργικότητα και όχι το κέρδος. Οι χρήστες μοιράζονται τις γνώσεις τους ελεύθερα, με σκοπό να εμπνεύσουν τους υπόλοιπους ή να τους βοηθήσουν να επιλύσουν διάφορα προβλήματα. Δημοσιεύοντας τη δουλειά τους, παράλληλα, δέχονται κριτική και βελτιώνονται (Kuznetsov & Paulos, 2010).

Συγκρίνοντας την φιλοσοφία της κοινότητας DIY και αυτή των Open Source λογισμικών, θα περίμενε κανείς ότι τα μέλη της πρώτης θα δείχνουν προτίμηση σε προγράμματα αυτής της λογικής. Πράγματι, τα λογισμικά ανοιχτού κώδικα συναντούν μεγάλη ανταπόκριση από καλλιτέχνες, επιστήμονες και γενικότερα από οποιοδήποτε άτομο χρειάζεται έναν προγραμματιστικό κώδικα για την ανάπτυξη ενός έργου/συστήματος. Μέσω της δυνατότητας τροποποίησης του πηγαίου κώδικα, δίνεται η ελευθερία αλλά και το κίνητρο να δημιουργηθούν πρωτοποριακά έργα. Υπάρχουν μάλιστα και εταιρείες που παρέχουν υλισμικό (hardware) που αξιοποιεί αυτήν ακριβώς τη δυνατότητα (π.χ Raspberry Pi). Πάνω τους οι χρήστες μπορούν να χτίσουν κώδικες και να δημιουργήσουν εφαρμογές με αμέτρητες χρήσεις.

Συνοψίζοντας, η εφαρμογή του μοντέλου του ανοιχτού κώδικα συμβαίνει, σε όλες τις μορφές του, υπό το πλαίσιο των βασικών αρχών του. Αυτές οι αρχές είναι η διαφάνεια των πληροφοριών ώστε όλοι να έχουν πρόσβαση σε αυτές, η συνεργασία, η άμεση και συχνή έκδοση του λογισμικού για να βρίσκονται λύσεις ευκολότερα, η αξιοκρατία και η σημαντικότητα της κοινότητας. Αξίες που βέβαια μπορούν να διαπεράσουν τα όρια του προγραμματιστικού κόσμου και να προωθηθούν για να βελτιώσουν τις ανθρώπινες σχέσεις και συμπεριφορές (The open source way, n.d.).

## **2. Κατασκευή και προγραμματισμός του συστήματος**

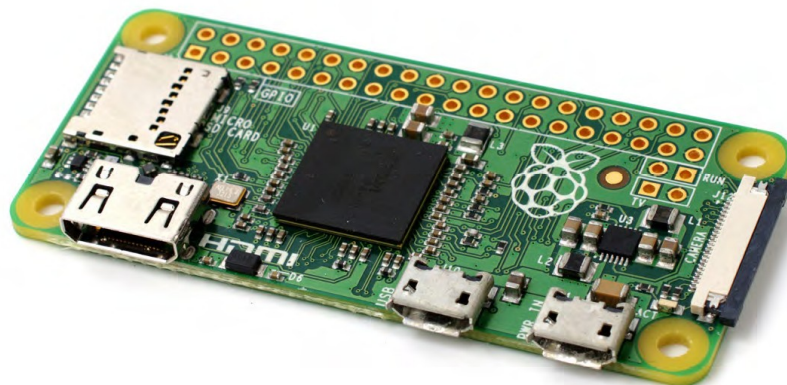
### **2.1 Κατασκευή υλισμικού συστήματος**

Το υλισμικό σύστημα αποτελείται από δύο κυρίως μέρη, τον μικροϋπολογιστή Raspberry Pi και το συνοδευτικό κύκλωμα που περιέχει όλα τα απαραίτητα στοιχεία που χρειάζονται για την προετοιμασία του αναλογικού σήματος, την μετατροπή του σε ψηφιακό, καθώς και διάφορες άλλες λειτουργίες όπως φιλτράρισμα θορύβου, κτλ. Επίσης, περιέχει όλα τα περιφερειακά στοιχεία που απαιτούνται για την σωστή σύνδεση μεταξύ των μερών του συστήματος, την πηγή σήματος (κιθάρα), την τροφοδοσία τάσης ρεύματος του μικροϋπολογιστή και του συνοδευτικού κυκλώματος. Στο κομμάτι αυτό εντάσσεται και η στέγαση όλων των επιμέρους κομματιών σε στεγανό και ασφαλές περιβάλλον ώστε να διασφαλίζεται η απρόσκοπτη λειτουργία του συστήματος. Στη συνέχεια του κεφαλαίου θα παρουσιαστούν λεπτομερώς τα μέρη από τα οποία αποτελείται το σύστημα, παραθέτοντας φωτογραφίες της διαδικασίας κατασκευής και εξηγώντας τα βήματα που ακολουθήθηκαν και τους λόγους επιλογής των συγκεκριμένων μερών.

#### **2.1.1 Επιλογή μικροϋπολογιστή για την επεξεργασία του σήματος**

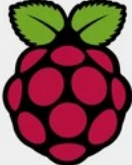
Τα εφέ που προσφέρει το σύστημα είναι υπό τη μορφή αλγόριθμων προγραμματιστικού κώδικα. Αυτό σημαίνει πως για την επεξεργασία του ηχητικού σήματος και την προσθήκη εφέ σε αυτό, απαιτείται η ύπαρξη ενός υπολογιστή με αρκετή επεξεργαστική δύναμη, ώστε να “τρέχουν” οι αλγόριθμοι των εφέ χωρίς καθυστέρηση. Απαραίτητη προϋπόθεση είναι η ύπαρξη υποδοχών GPIO (General Purpose Input/Output) μέσω των οποίων καθίσταται δυνατή η εισροή και η εκροή δεδομένων μεταξύ του υπολογιστή και των υπόλοιπων στοιχείων του συστήματος. Επιπλέον, ο υπολογιστής αυτός πρέπει να έχει μικρό μέγεθος ώστε να μπορεί να προσαρμοστεί σε ένα μικρό κουτί μαζί με τα υπόλοιπα ηλεκτρονικά μέρη, ώστε το σύστημα να έχει τη μορφή πεταλιού κιθάρας. Λαμβάνοντας υπόψη τις παραπάνω προϋποθέσεις και έχοντας ως βασικό στόχο το χαμηλό συνολικό κόστος κατασκευής, ο υπολογιστής που επιλέχθηκε για το σύστημα ήταν ο Raspberry Pi Zero W.





Εικόνα 7: Ο μικροϋπολογιστής Raspberry Pi Zero W. (Upton & Halfacree, 2016)

Το Raspberry Pi είναι ένας χαμηλού κόστους υπολογιστής, που σχεδιάστηκε ως εργαλείο εκμάθησης προγραμματισμού. Προσφέρεται, επίσης, για πειραματισμό με ηλεκτρονικά κυκλώματα χάρη στις δύο σειρές (2x20) GPIO pins, που δίνουν τη δυνατότητα ανταλλαγής δεδομένων από και προς τον υπολογιστή (Watkiss, 2016). Η εξέλιξη του Raspberry Pi από την εμφάνιση του πρώτου μοντέλου (Raspberry Pi 1 Model B) το 2011 μέχρι σήμερα είναι πολύ μεγάλη, όπως ραγδαία είναι και η εξάπλωση του. Υπάρχουν διαθέσιμα στην αγορά πολλά μοντέλα Raspberry Pi με διαφορετικά χαρακτηριστικά, ώστε να εξυπηρετούν τις ανάγκες διαφόρων εφαρμογών (McManus & Cook, 2017). Στην παρακάτω εικόνα παρουσιάζονται τα διαθέσιμα μοντέλα Raspberry Pi με τα χαρακτηριστικά τους:

	Raspberry Pi 3 Model B	Raspberry Pi Zero	Raspberry Pi 2 Model B	Raspberry Pi Model B+
Introduction Date	2/29/2016	11/25/2015	2/2/2015	7/14/2014
SoC	BCM2837	BCM2835	BCM2836	BCM2835
CPU	Quad Cortex A53 @ 1.2GHz	ARM11 @ 1GHz	Quad Cortex A7 @ 900MHz	ARM11 @ 700MHz
Instruction set	ARMv8-A	ARMv6	ARMv7-A	ARMv6
GPU	400MHz VideoCore IV	250MHz VideoCore IV	250MHz VideoCore IV	250MHz VideoCore IV
RAM	1GB SDRAM	512 MB SDRAM	1GB SDRAM	512MB SDRAM
Storage	micro-SD	micro-SD	micro-SD	micro-SD
Ethernet	10/100	none	10/100	10/100
Wireless	802.11n / Bluetooth 4.0	none	none	none
Video Output	HDMI / Composite	HDMI / Composite	HDMI / Composite	HDMI / Composite
Audio Output	HDMI / Headphone	HDMI	HDMI / Headphone	HDMI / Headphone
GPIO	40	40	40	40
Price	\$35	\$5	\$35	\$35

Εικόνα 8: Τα μοντέλα Raspberry Pi με τα χαρακτηριστικά τους. (Benchhoff, 2016)

Το Raspberry Pi Zero W που χρησιμοποιούμε στο σύστημά μας, έχει σχεδόν τις ίδιες δυνατότητες με το Zero όπως αυτές φαίνονται στην παραπάνω εικόνα. Μοναδική τους διαφορά (η οποία όμως είναι καταλυτικής σημασίας για την επιλογή υπολογιστή για το σύστημα), η ύπαρξη ενσωματωμένου Wi-Fi και Bluetooth. Αξίζει να αναφερθεί πως στην παραπάνω εικόνα δεν συμπεριλαμβάνεται το νεότερο μοντέλο Raspberry Pi Model 4 B που βγήκε σε παραγωγή τον Ιούνιο του 2019 και παρέχει αυξημένες δυνατότητες (επεξεργαστική ισχύς, περιφερειακά) σε σχέση με τα παλαιότερα μοντέλα.

### **2.1.2 Σύγκριση του Zero W με άλλα διαθέσιμα μοντέλα**

Η επιλογή του Zero W έναντι κάποιου άλλου μοντέλου της οικογένειας Raspberry Pi, έγινε έχοντας κάποιες προϋποθέσεις που προκύπτουν από τις ανάγκες της εφαρμογής, αλλά και περιορισμούς που είτε θέσαμε, είτε τέθηκαν από εξωγενείς παράγοντες. Οι σημαντικότεροι λόγοι, είναι οι εξής:

- Μικρό μέγεθος και βάρος. Το Zero W έχει μήκος 65mm, πλάτος 30mm και ύψος 13mm (συμπεριλαμβανομένων των pins υποδοχής καλωδίων). Το βάρος του ανέρχεται στα 12 γραμμάρια. Συγκριτικά, το μοντέλο 3 B+ είναι αρκετά μεγαλύτερο σε διαστάσεις (85.6mm μήκος, 56.5mm πλάτος, 17mm ύψος) και σχεδόν 4 φορές βαρύτερο (45 γραμμάρια). Οι πολύ μικρές διαστάσεις του και το σχεδόν αμελητέο βάρος, το καθιστούν ιδανικό για το σύστημά μας, λαμβάνοντας υπόψη τον στόχο για ευελιξία και ευκολία στη μεταφορά του συστήματος.
- Χαμηλή κατανάλωση ρεύματος. Το Zero W έχει κατανάλωση 180mA ανά ώρα. Το 3 B+ αντίστοιχα, μπορεί να φτάσει τα 1.13A κατανάλωση ρεύματος ανά ώρα λειτουργίας (Raspberry Pi models comparison, 2019). Η χαμηλή κατανάλωση του συστήματος είναι σημαντική για την ευελιξία του, προσδίδοντάς του μεγαλύτερη διάρκεια συνεχούς λειτουργίας με χρήση μπαταριών.
- Έχει ενσωματωμένο Bluetooth και WiFi. Αποφεύγεται, έτσι, η χρήση επιπλέον περιφερειακών αντάπτορων για αυτές τις λειτουργίες, κερδίζοντας χώρο και λιγότερη πολυπλοκότητα κατασκευής και προγραμματισμού.
- Διαθέτει αρκετή επεξεργαστική ισχύ για την εκτέλεση των αλγόριθμων εφέ και για την αποστολή των ηχητικών δεδομένων μέσω του πρωτοκόλλου Bluetooth. Το Zero W είναι ένα μοντέλο μεσαίου βεληνικού, σε ότι αφορά την ισχύ του επεξεργαστή, σε σχέση με τα άλλα μοντέλα Raspberry Pi. Παρόλα αυτά διαθέτει επεξεργαστή 1GHz και

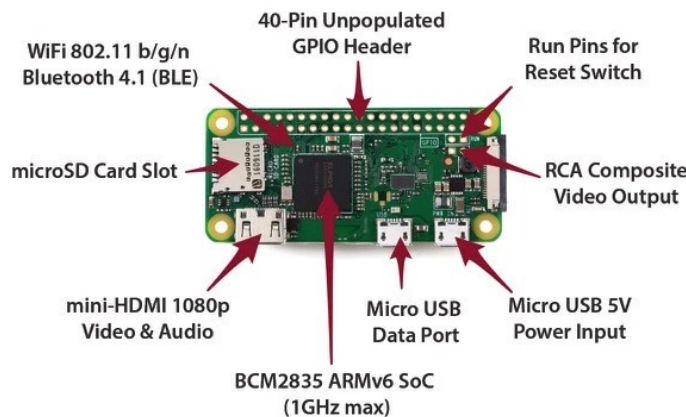
RAM 512MB που, σε συνδυασμό με τον αποκλεισμό άλλων κοστοβόρων εφαρμογών, αποδίδει ικανοποιητικά βάση των αναγκών της εφαρμογής.

- Χαμηλό κόστος. Το Zero W είναι το φθηνότερο μοντέλο της σειράς (μετά την απλή έκδοση Zero) με τιμή ως και 3 φορές χαμηλότερη από το Model 3 B+ και έως 4 φορές από το Model 4 B.

Αξίζει να σημειωθεί πως το σύστημα γίνεται να λειτουργήσει με κάποιο από τα άλλα μοντέλα Raspberry Pi, προσαρμόζοντας τον κατάλληλο αντάπτορα Bluetooth. Το γεγονός αυτό κάνει το σύστημα πιο προσιτό για αναπαραγωγή από τρίτους, καθώς το κάθε άτομο που διαθέτει έναν μικροϋπολογιστή Raspberry Pi μπορεί να κατασκευάσει και να εξελίξει το σύστημα περαιτέρω.

### 2.1.3 Τα βασικά μέρη του Raspberry Pi Zero W

Πριν περάσουμε στο συμπληρωματικό κύκλωμα και στα βήματα κατασκευής του υλισμικού συστήματος, ας δούμε τα βασικά μέρη από από τα οποία αποτελείται το Raspberry Pi Zero W και το ρόλο του καθενός στη λειτουργία του. Η παρουσίαση των μερών κρίνεται ωφέλιμη για την κατανόηση της διαδικασίας σύνδεσης των στοιχείων του συστήματος και της κατασκευής του.



*Εικόνα 9: Τα βασικά στοιχεία του Raspberry Pi Zero W.  
(<https://raspberrypi.org.au/raspberry-pi-zero-w>)*

- Ο επεξεργαστής. Το μαύρο τσιπ στο κέντρο του Raspberry Pi Zero W είναι το μοντέλο Broadcom BCM 2835 με επεξεργαστή ARM11 με ταχύτητα 1GHz και μνήμη RAM

512MB. Το ίδιο τσιπ υπήρχε και στα πρώτα μοντέλα Raspberry Pi σε ταχύτητα 512MHz (Donat, 2018).

- Υποδοχή κάρτας microSD. Για τη λειτουργία του υπολογιστή χρειάζεται microSD κάρτα με χωρητικότητα τουλάχιστον 4GB. Χωρίς την ύπαρξη κάρτας στην υποδοχή το Zero δεν μπορεί να ξεκινήσει τη λειτουργία του με το LED ένδειξης να παραμένει σβηστό (Richardson & Wallace, 2013).
- Δεξιά θύρα micro USB. Λέγεται αλλιώς και Power USB, αφού είναι η θύρα στην οποία συνδέεται η παροχή ρεύματος τάσης 5V.
- Αριστερή θύρα micro USB. Σε αυτή τη θύρα συνδέονται τυχόν περιφερειακά μέρη (Wi-Fi και Bluetooth αντάπτορες, οθόνη, κτλ).
- Θύρα Mini-HDMI. Μέσω αυτής της θύρας το Zero μπορεί να εξάγει 1080p αρχείο βίντεο.
- LED ένδειξης λειτουργίας. Ανάβει σε χρώμα πράσινο όταν το Raspberry Pi είναι ενεργοποιημένο.
- Υποδοχές GPIO (General Purpose Input and Output). Πρόκειται για υποδοχές που μπορούν να χρησιμοποιηθούν ως είσοδοι ή έξοδοι δεδομένων. Συνδέονται κατευθείαν με τις θύρες GPIO του επεξεργαστή. Επειδή ο επεξεργαστής λειτουργεί στα 3.3V, αυτή είναι και η μέγιστη τάση που μπορούν να δεχτούν οι GPIO υποδοχές. Διαφορετικά, για υψηλότερες τιμές τάσης, υπάρχει κίνδυνος πρόκλησης βλάβης στο Raspberry Pi. Οι υποδοχές μπορούν να χρησιμοποιηθούν από πρωτόκολλα σειριακής επικοινωνίας όπως τα UART (Universal Asynchronous Receiver/Transmitter), I2C (Inter-Integrated Circuit) και SPI (Serial Peripheral Interface Bus)(Watkiss, 2016). Το κάθε πρωτόκολλο χρησιμοποιεί συγκεκριμένες υποδοχές για τη μεταφορά των δεδομένων όπως φαίνεται στην εικόνα 10:

GPIO#	NAME			NAME	GPIO#
	3.3 VDC Power	1		5.0 VDC Power	2
8	GPIO 8 SDA1 (I2C)	3		5.0 VDC Power	4
9	GPIO 9 SCL1 (I2C)	5		Ground	6
7	GPIO 7 GPCLK0	7		GPIO 15 TxD (UART)	15
	Ground	9		GPIO 16 RxD (UART)	16
0	GPIO 0	11		GPIO 1 PCM_CLK/PWM0	1
2	GPIO 2	13		Ground	14
3	GPIO 3	15		GPIO 4	4
	3.3 VDC Power	17		GPIO 5	5
12	GPIO 12 MOSI (SPI)	19		Ground	20
13	GPIO 13 MISO (SPI)	21		GPIO 6	6
14	GPIO 14 SCLK (SPI)	23		GPIO 10 CE0 (SPI)	10
	Ground	25		GPIO 11 CE1 (SPI)	11
30	SDA0 (I2C ID EEPROM)	27		SCL0 (I2C ID EEPROM)	31
21	GPIO 21 GPCLK1	29		Ground	30
22	GPIO 22 GPCLK2	31		GPIO 26 PWM0	26
23	GPIO 23 PWM1	33		Ground	34
24	GPIO 24 PCM_FS/PWM1	35		GPIO 27	27
25	GPIO 25	37		GPIO 28 PCM_DIN	28
	Ground	39		GPIO 29 PCM_DOUT	29

Εικόνα 10: Οι υποδοχές GPIO στο Raspberry Pi Zero W.  
(<https://pi4j.com/1.2/pins/model-zero-w.html>)

#### 2.1.4 Το συνοδευτικό κύκλωμα

Η λειτουργία του συστήματος προϋποθέτει την ύπαρξη συνοδευτικού κυκλώματος (βλ. εικόνα 1, Παράρτημα 1) με τα κατάλληλα ηλεκτρονικά εξαρτήματα για την προετοιμασία του ηχητικού σήματος από την πηγή (κιθάρα) μέχρι την ψηφιοποίηση του από τον A/D μετατροπέα. Για την κατασκευή του κυκλώματος, χρησιμοποιήθηκε Open Source σχηματικό του διαδικτύου. (βλέπε εικόνα 2, Παράρτημα 1), με κάποιες αλλαγές για τις ανάγκες της εφαρμογής. Πριν μελετήσουμε τη λειτουργία του κυκλώματος, κρίνεται σημαντικό να αναφερθούμε στον A/D μετατροπέα, καθώς αποτελεί σημείο αναφοράς για την λειτουργία του συστήματος, και έτσι είναι λογικό το κύκλωμα να έχει σχεδιαστεί με βάση το συγκεκριμένο εξάρτημα.

### **A/D Μετατροπέας MCP3202:**

Ο μικροϋπολογιστής Raspberry Pi Zero W δε διαθέτει ενσωματωμένο A/D μετατροπέα. Έτσι, για την επεξεργασία αναλογικού ηχητικού σήματος χρειάζεται ένας εξωτερικός μετατροπέας. Ο μετατροπέας που χρησιμοποιείται στο σύστημα είναι ο MCP3202 της εταιρείας Microchip (βλ. εικόνα 1, Παράρτημα 2). Πρόκειται για έναν μετατροπέα 12-Bit, 2 εισόδων, με δυνατότητα επικοινωνίας με υπολογιστή μέσω του SPI πρωτοκόλλου. Διαθέτει εύρος τάσης λειτουργίας από 2.7V μέχρι 5V. Η συχνότητα δειγματοληψίας καθορίζεται ανάλογα με την τάση τροφοδοσίας του. Έτσι, για τάση τροφοδοσίας  $V=5V$ , έχει συχνότητα δειγματοληψίας 100kHz, ενώ για τάση τροφοδοσίας  $V=2.7V$ , η συχνότητα δειγματοληψίας ανέρχεται σε 50kHz.

Όπως φαίνεται από τα παραπάνω στοιχεία, η τάση τροφοδοσίας 5V θα ήταν ιδανική για τον MCP3202. Επειδή, όμως, οι GPIO υποδοχές του Raspberry Pi λειτουργούν στα 3.3V αυτή θα είναι και η τάση τροφοδοσίας του μετατροπέα, έχοντας έτσι συχνότητα δειγματοληψίας περίπου 50kHz. Αυτή η συχνότητα δειγματοληψίας αξιολογείται ως σχετικά καλή για ψηφιακή επεξεργασία ήχου. Γενικά, η επιλογή του MCP3202 έγινε διότι αποτελεί μια οικονομική λύση, είναι απλός στη χρήση και διαθέτει τεχνικά χαρακτηριστικά που ανταποκρίνονται στις ανάγκες του συστήματος.

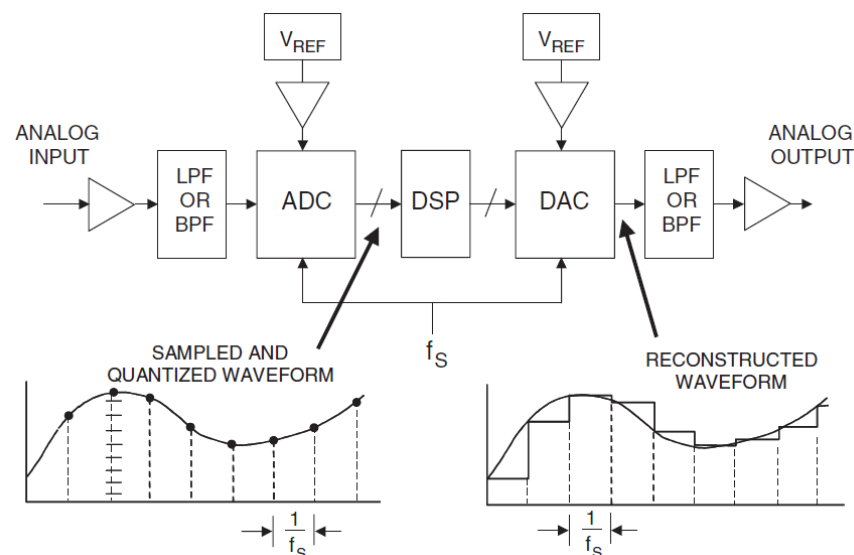
Για να γίνει πιο εύκολα κατανοητή η λειτουργία του κυκλώματος, θα το χωρίσουμε σε μέρη και θα μελετήσουμε τη λειτουργία του κάθε μέρους ξεχωριστά.

### **Κύκλωμα Εισόδου**

Το σήμα της πηγής (κιθάρα) πρέπει να ενισχυθεί πριν φτάσει στον A/D μετατροπέα. Η ενίσχυση γίνεται διότι το σήμα της κιθάρας έχει διαφορά δυναμικού περίπου 200mV, ενώ ο A/D μετατροπέας χρειάζεται ιδανικά 3.3V. Επίσης, το σήμα που παράγεται από τους μαγνήτες της κιθάρας έχει χαμηλή εσωτερική αντίσταση (impedance), οπότε πρέπει να “συναντήσει” υψηλή εσωτερική αντίσταση στο κύκλωμα, ώστε να αποφευχθεί η απώλεια σήματος (Jung, 2005). Η ενίσχυση του σήματος επιτυγχάνεται με τη χρήση ενός τελεστικού ενισχυτή (Op Amp) σε λειτουργία μη-αναστροφής (non-inverting). Τα οφέλη μιας τοπολογίας μη-αναστρεφόμενου τελεστικού ενισχυτή είναι το κέρδος τάσης, που στην περίπτωση μας είναι από 1 έως 21 και η πολύ υψηλή αντίσταση εισόδου (500KΩ, αρκετή ώστε να μην “φορτώνονται” οι μαγνήτες της κιθάρας, με αποτέλεσμα την απώλεια σήματος).

Ο τελεστικός ενισχυτής του συστήματος είναι ο MCP6002 της Microchip (βλ. εικόνα 2, Παράρτημα 2). Πρόκειται για έναν διπλό ενισχυτή, με εύρος ζώνης 1MHz, και εύρος τάσης λειτουργίας από 1.8 έως 6.0V. Η δυνατότητα λειτουργίας σε χαμηλή τάση τον καθιστά ιδανικό

για τη συγκεκριμένη εφαρμογή. Το κέρδος τάσης του ενισχυτή ρυθμίζεται από μια μεταβλητή αντίσταση 500 KΩ σε μορφή ποτενσιόμετρου, που στο κύκλωμα φαίνεται ως RV1 (βλ. εικόνα 1, Παράρτημα 1). Ένα ακόμα σημαντικό σκέλος του κυκλώματος εισόδου είναι η ύπαρξη φίλτρων για δύο πολύ σημαντικούς λόγους. Ο πρώτος λόγος είναι η αποφυγή του θορύβου που προκύπτει από συνεχές ρεύμα (DC) και του θορύβου χαμηλών συχνοτήτων (hum). Για τον σκοπό αυτό χρησιμοποιούνται υπερπαραφίλτρα (συνδυασμοί αντιστάσεων-πυκνωτών R1-C1, R3-C3). Ο δεύτερος λόγος είναι η απαλοιφή των υψηλών αρμονικών που υπάρχουν στο σήμα και που μπορούν να προκαλέσουν φαινόμενα αλλοίωσης (aliasing) κατά την διαδικασία ψηφιοποίησης του σήματος από τον A/D μετατροπέα (Jung, 2005). Χρησιμοποιούνται δύο χαμηλοπερατά φίλτρα (συνδυασμοί αντιστάσεων-πυκνωτών R2-C2, R4-C5) ή αλλιώς φίλτρα αντιαλλοίωσης (anti-aliasing).



Εικόνα 11: Ένα τυπικό σύστημα ψηφιοποίησης-αποψηφιοποίησης σήματος. (Jung, 2005)

### Κύκλωμα τροφοδοσίας

Στο κύκλωμα τροφοδοσίας εντάσσονται τα εξαρτήματα που χρειάζονται για τη τροφοδοσία του κυκλώματος με 5V και 3.3V από τις GPIO υποδοχές του Raspberry Pi. Επίσης, εντάσσονται τα φίλτρα που απαιτούνται για την απομόνωση του θορύβου που υπάρχει σε αυτές τις πηγές τροφοδοσίας. Χρησιμοποιούνται πυκνωτές αποσύνδεσης (decoupling) στον τελεστικό ενισχυτή και δύο φίλτρα τύπου CRC (πυκνωτή-αντίστασης-πυκνωτή) για την εξάλειψη

υψίσυχνου θορύβου από τις γραμμές τροφοδοσίας. Η συχνότητα αποκοπής για αυτά τα δύο φίλτρα έχει οριστεί στα 2kHz.

### **Κύκλωμα σύνδεσης περιφερειακών**

Σε αυτό το μικροκύκλωμα περιλαμβάνονται όλα τα περιφερειακά ηλεκτρονικά εξαρτήματα που χρησιμοποιούνται στο σύστημα. Πρόκειται για τις δύο επαφές τύπου (Normally Open) κομβίου, τον διακόπτη τύπου Spdt ( Single Pole, Double Throw), την επαφή τύπου διακόπτη ποδιού 3pdt (3 Pole, Double Throw), και το LED κατάστασης λειτουργίας του συστήματος.

### **Κύκλωμα Εξόδου<sup>1</sup>**

Στο συγκεκριμένο μέρος του κυκλώματος πραγματοποιείται η αποψηφιοποίηση του σήματος χρησιμοποιώντας την μέθοδο Pulse Width Modulation (PWM). Η μέθοδος αυτή είναι διαδεδομένη σε εφαρμογές επεξεργασίας σήματος χάρη στην απλή λογική της που θυμίζει μεταφορά ραδιοκυμάτων (φέρων σήμα και διαμορφωτής). Εν ολίγοις, μια ρυθμιζόμενη τιμή τάσης ρεύματος μπορεί να παραχθεί όταν ένα φέρον ψηφιακό σήμα υπόκειται διαμόρφωση τύπου Pulse Width, και στη συνέχεια το διαμορφωμένο σήμα διέρχεται ενός χαμηλοπερατού φίλτρου (Keim, 2016). Χρησιμοποιώντας δύο σήματα PWM παράλληλα επιτυγχάνεται χαμηλότερη στάθμη θορύβου και υψηλότερη συχνότητα λειτουργίας. Η μέθοδος PWM χρησιμοποιείται καθώς το Raspberry Pi δεν διαθέτει ενσωματωμένο D/A μετατροπέα (Digital to Analog). Είναι δυνατή η χρήση ενός εξωτερικού D/A μετατροπέα, αλλά θα πρόσθετε επιπλέον κόστος και πολυπλοκότητα κατασκευής στο σύστημα. Για ανάλυση 12Bit (ο περιορισμός λόγω επιλογής του συγκεκριμένου A/D μετατροπέα), η μέθοδος PWM μπορεί να έχει πολύ καλά αποτελέσματα, με αρκετά χαμηλότερο κόστος εξαρτημάτων και μικρότερη πολυπλοκότητα κυκλώματος.

Το κύκλωμα εξόδου, λοιπόν, περιλαμβάνει τα εξαρτήματα που υπάρχουν ενδιάμεσα των GPIO υποδοχών που είναι αφιερωμένες στο PWM, και στην έξοδο του κυκλώματος όπου και βγαίνει το τελικό σήμα. Μέσα στα εξαρτήματα αυτά είναι το δεύτερο κανάλι του τελεστικού

---

<sup>1</sup> Το σύστημα στην κανονική του μορφή δεν χρησιμοποιεί το κύκλωμα εξόδου, διότι το ψηφιακό σήμα αποστέλλεται μέσω Bluetooth στο ηχείο όπου γίνεται στη συνέχεια η αποψηφιοποίηση του. Επειδή, όμως, το κύκλωμα εξόδου χρειάζεται για την λειτουργία του συστήματος χωρίς Bluetooth, με σύνδεση απευθείας σε ενισχυτή κιθάρας, συμπεριλαμβάνεται στην κατασκευή του κυκλώματος του συστήματος.



ενισχυτή (ο MCP6002 διαθέτει δύο κανάλια λειτουργίας) και το 3ης τάξης χαμηλοπερατό φίλτρο τύπου Sallen & Key, με συχνότητα αποκοπής τα 5kHz (για να απαλείφει το θόρυβο που προκαλείται από το φέρον σήμα της μεθόδου PWM). Το Sallen & Key είναι ένα ενεργό φίλτρο (χρησιμοποιείται τελεστικός ενισχυτής), που χρησιμοποιείται σε εφαρμογές που απαιτείται μεγάλος ρυθμός αποκοπής (Horowitz & Hill, 2015).

### 2.1.5 Κατασκευή Υλισμικού Συστήματος

Η κατασκευή του συστήματος μπορεί να χωριστεί σε έξι κύρια βήματα. Η σειρά παρουσίασης τους είναι αντίστοιχη με τη σειρά κατασκευής.

1. Αποτύπωση του σχηματικού σε κύκλωμα μορφής πλακέτας γραμμών χαλκού (stripboard).

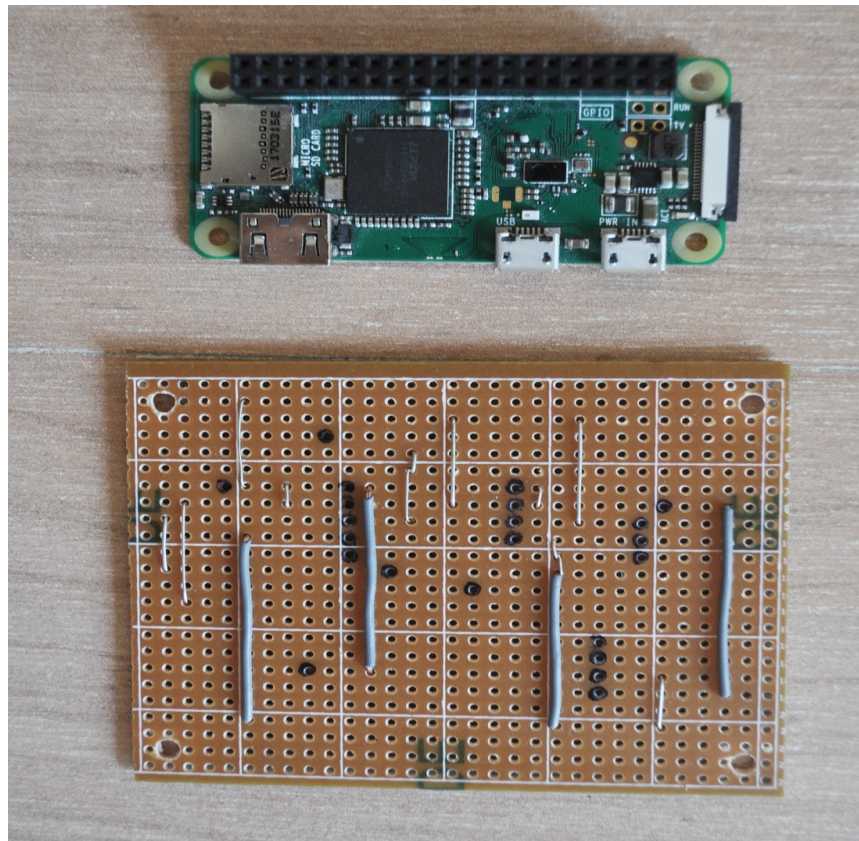
Για την κατασκευή του κυκλώματος σε μορφή πλακέτας απαιτείται η ύπαρξη κατάλληλου σχηματικού όπου φαίνονται όλα τα ηλεκτρονικά εξαρτήματα και η συνδεσμολογία τους. Χρησιμοποιήθηκε το σχηματικό της εικόνας 2, Παράρτημα 1, με κάποιες μικρές διαφοροποιήσεις που αφορούν κυρίως τιμές των εξαρτημάτων (πχ. την τιμή σε  $\Omega$  μιας αντίστασης).

Για την σχεδίαση του κυκλώματος έγινε χρήση ενός πολύ διαδεδομένου στην DIY κοινότητα προγράμματος, του DIY Layout Creator (<http://bancika.github.io/diy-layout-creator/>). Το freeware πρόγραμμα σχεδιασμού κυκλωμάτων DIY Layout Creator προσφέρεται δωρεάν στους χρήστες, χρησιμοποιεί ελεύκτο Open Source λογισμικό και γενικώς αποτελεί ένα πολύτιμο εργαλείο για εφαρμογές που απαιτούν την σχεδίαση ηλεκτρονικών κυκλωμάτων.

Η σχεδίαση του κυκλώματος αποτελεί πολύ σημαντικό μέρος της συνολικής κατασκευής. Η σωστή αποτύπωση διασφαλίζει την αποφυγή θορύβου από εξωγενείς παράγοντες (παρασιτικές χωρητικότητες), τη σταθερότητα του συστήματος σε συνθήκες συνεχούς λειτουργίας και γενικώς εξασφαλίζει την ανεμπόδιστη πορεία του ηχητικού σήματος από την πηγή μέχρι το ηχείο. Σχεδιάστηκαν συνολικά τρεις εκδόσεις του κυκλώματος. Η τρίτη κατά σειρά έκδοση του κυκλώματος χρησιμοποιήθηκε στο σύστημα μας (βλ. εικόνα 1, Παράρτημα 1).

## 2. Προετοιμασία της πλακέτας για την τοποθέτηση των ηλεκτρονικών εξαρτημάτων.

Σε αυτό το βήμα συμπεριλαμβάνονται η κοπή της διάτρητης πλακέτας με γραμμές (stripboard), ο καθαρισμός της πλακέτας από ακαθαρσίες πριν την κόλληση των εξαρτημάτων και η δημιουργία δρόμων μέσω της αποχάλκωσης συγκεκριμένων σημείων και της σύνδεσης άλλων με αγωγίμο σύρμα.

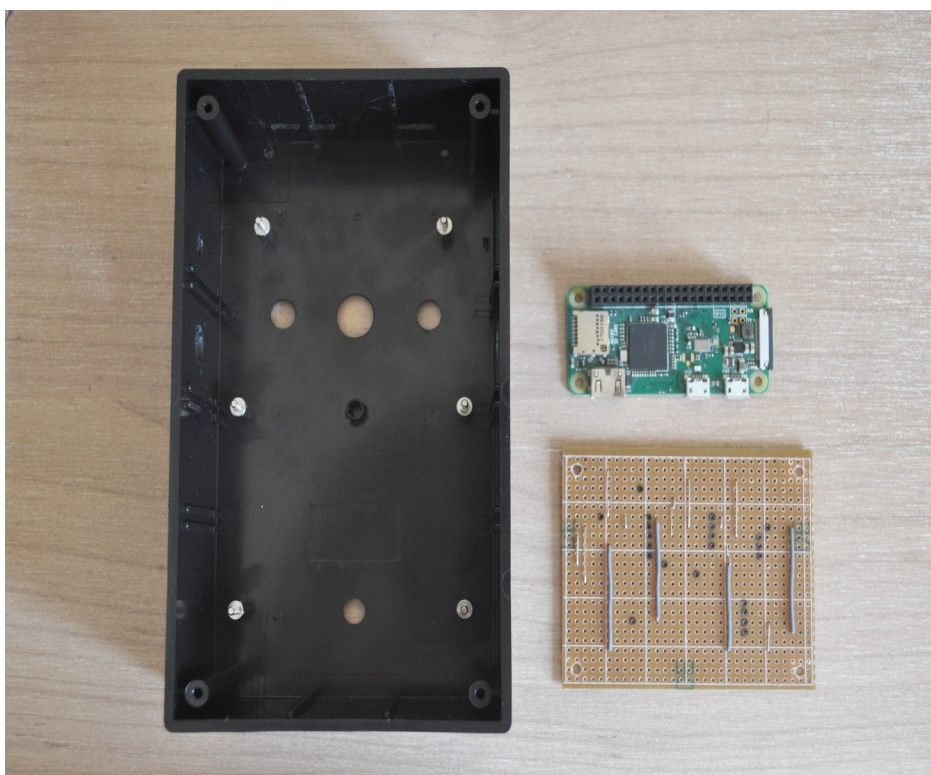


## 3. Συναρμολόγηση της τελικής μορφής της πλακέτας.

Το βήμα αυτό αφορά την κόλληση με ηλεκτρικό κολλητήρι και καλάνι όλων των ηλεκτρονικών εξαρτημάτων (ολοκληρωμένων, αντιστάσεων, πυκνωτών κτλ.) στις προκαθορισμένες θέσεις τους, όπως αυτά φαίνονται στο κύκλωμα. Επίσης, γίνεται ο τελικός έλεγχος σωστής τοποθέτησης και επιτυχημένης κόλλησης, καθώς και η πολύ σημαντική διαδικασία ελέγχου για τυχόν βραχυκυκλώματα.

#### 4. Προετοιμασία του κουτιού προστασίας του συστήματος.

Το κουτί προστασίας είναι κατασκευασμένο από σκληρό πλαστικό, ορθογώνιου σχήματος, διαστάσεων 158mm x 96mm x 53mm. Η προετοιμασία του περιλαμβάνει τη διάτρηση οπών διαφόρων διαμετρημάτων, στις οποίες τοποθετούνται τα εξαρτήματα ελέγχου του συστήματος (ποτενσιόμετρο, κομβία, LED ένδειξης λειτουργίας κτλ.) και τα βύσματα εισόδου και εξόδου. Επίσης, στο κουτί στερεώνονται με βίδες οι αποστάτες που συγκρατούν το κύκλωμα και το Raspberry Pi.

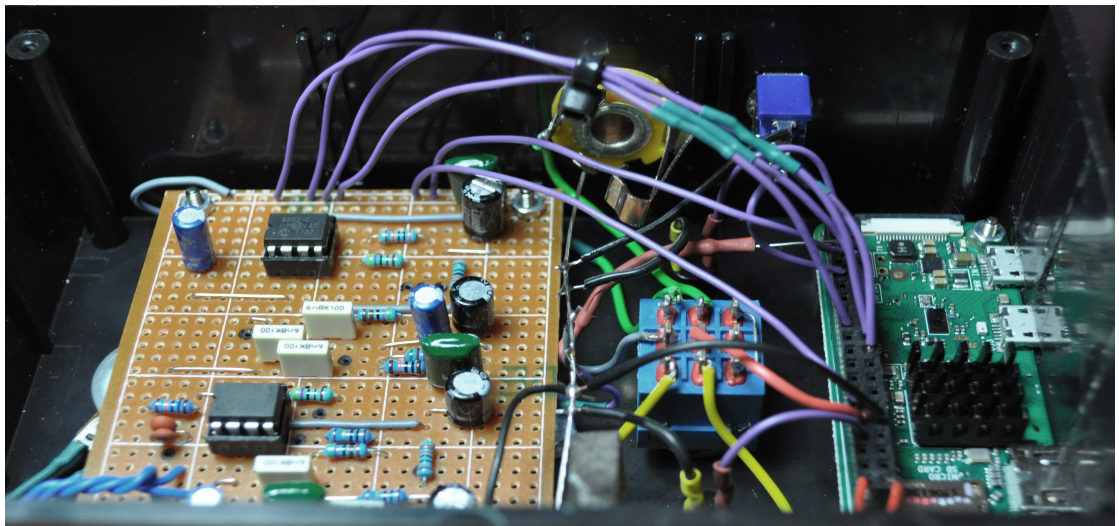
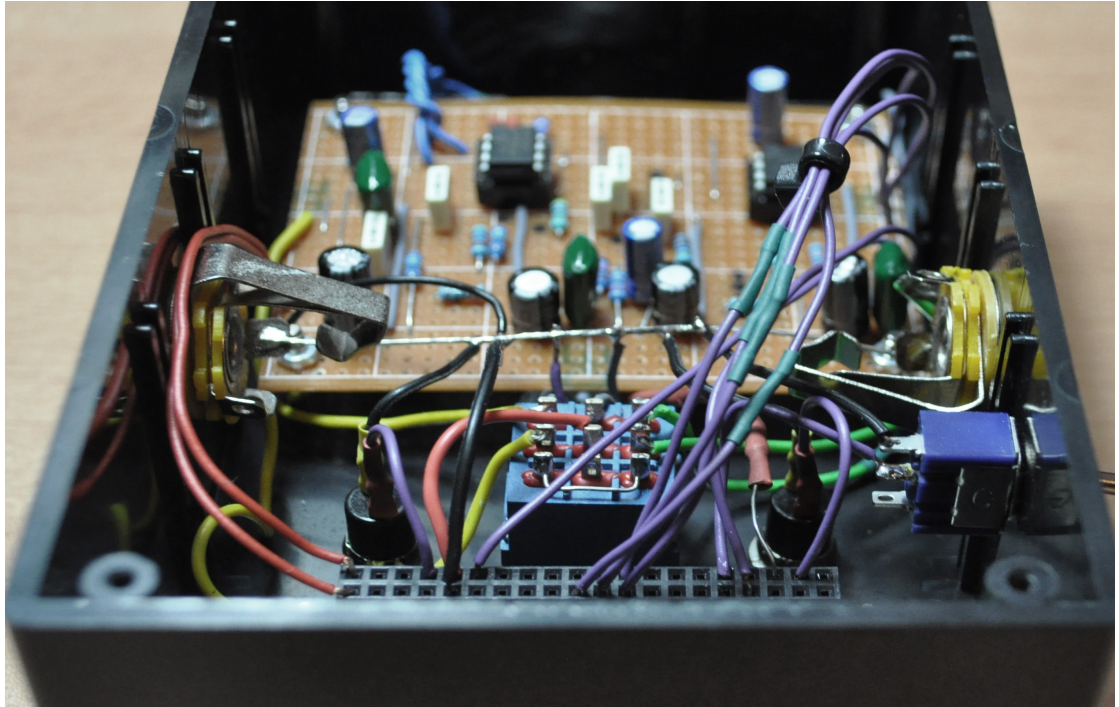




5. Τοποθέτηση των μερών του συστήματος στο κουτί προστασίας και σύνδεση μεταξύ τους.

Πρώτα τοποθετούνται τα βύσματα εισόδου εξόδου, τα εξαρτήματα ελέγχου και οι αποστάτες συγκράτησης. Στη συνέχεια στερεώνονται πάνω στους αποστάτες το κύκλωμα και το Raspberry Pi. Κατόπιν ακολουθεί το στάδιο της σύνδεσης των μερών με αγωγίμα καλώδια ώστε να εξασφαλιστεί η μεταφορά των δεδομένων, η τροφοδοσία με κατάλληλη τάση, και η σωστή γείωση όλων των ηλεκτρονικών εξαρτημάτων.







6. Τελική μορφή του συστήματος με το ηχείο Bluetooth.

Εδώ παρουσιάζεται η τελική μορφή του συστήματος έτοιμο για χρήση σε συνδυασμό με το ηχείο Bluetooth. Το ηχείο Bluetooth στη συγκεκριμένη περίπτωση είναι το JBL Flip 3. Η επιλογή του δεν έγινε βάση κάποιου ειδικού κριτηρίου αλλά για λόγους διαθεσιμότητας, αφού το συγκεκριμένο ηχείο προϋπήρχε στην κατοχή μας. Το σύστημα, εξάλλου, δεν διαθέτει κάποιο περιορισμό ως προς τα μοντέλα ηχείων Bluetooth με τα οποία μπορεί να συνδεθεί και να λειτουργήσει.



## 2.2 Προγραμματισμός του Λογισμικού Συστήματος

Το σύστημα, όπως είδαμε και στο προηγούμενο κεφάλαιο, αποτελείται από τον μικροϋπολογιστή Raspberry Pi Zero W, το συμπληρωματικό κύκλωμα όπου βρίσκονται τα ηλεκτρονικά εξαρτήματα, και τα περιφερειακά μέρη όπως το κουτί προστασίας κτλ. Ως λογισμικό σύστημα ορίζεται το σύνολο των άυλων υπολογιστικών προγραμμάτων και εφαρμογών που λαμβάνουν χώρα στο εσωτερικό ενός υπολογιστή. Τα προγράμματα αυτά είναι υπεύθυνα για τη σωστή λειτουργία του υλισμικού συστήματος και την επεξεργασία των ηλεκτρονικών δεδομένων. Στην παρούσα εργασία, το λογισμικό σύστημα ξεκινάει από το σημείο που το ψηφιοποιημένο, πλέον, σήμα εξέρχεται του A/D μετατροπέα και εισέρχεται στον μικροϋπολογιστή μέσω των GPIO υποδοχών. Τα τρία σημαντικότερα μέρη του λογισμικού συστήματος είναι η επικοινωνία, μέσω του πρωτοκόλλου SPI, του A/D μετατροπέα με τον μικροϋπολογιστή Raspberry Pi, η μονάδα επεξεργασίας ήχου στην οποία υπάρχουν οι αλγόριθμοι εφέ, και η ασύρματη αναπαραγωγή του ηχητικού σήματος μέσω Bluetooth. Στο λογισμικό σύστημα συμπεριλαμβάνεται, επίσης, το λειτουργικό σύστημα του Raspberry Pi και το σύνολο των αλγόριθμων που απαιτούνται για την υποστήριξη των υπολοίπων διαδικασιών. Στη συνέχεια του κεφαλαίου θα μελετήσουμε πιο αναλυτικά τα μέρη που αποτελούν το λογισμικό σύστημα, παραθέτοντας τους κώδικες που χρησιμοποιήθηκαν για την επίτευξη διαφόρων λειτουργιών και εξηγώντας την διαδικασία προγραμματισμού τους.

### 2.2.1 Το λειτουργικό σύστημα του Raspberry Pi

Το προκαθορισμένο λειτουργικό σύστημα του Raspberry Pi είναι το Linux. Το λειτουργικό σύστημα Linux αποτελείται από ελεύθερο λογισμικό, καθώς ο κώδικας του έχει γραφτεί από την αρχή ως ελεύθερο λογισμικό, υπό την ελεύθερη άδεια χρήσης GNU General Public License (Linux, n.d.). Υπάρχουν διάφορες διανομές του Linux που καλύπτουν διαφορετικές ανάγκες. Μερικές από αυτές είναι η Ubuntu, η Debian, η Fedora, κτλ. Το Raspberry Pi χρησιμοποιεί την διανομή Raspbian που είναι βασισμένη στη Debian (Donat, 2018). Το λογισμικό είναι ελεύθερα προσβάσιμο στην επίσημη ιστοσελίδα του Raspberry Pi (<https://www.raspberrypi.org/downloads/raspbian/>). Ο/Η χρήστης έχει τη δυνατότητα να

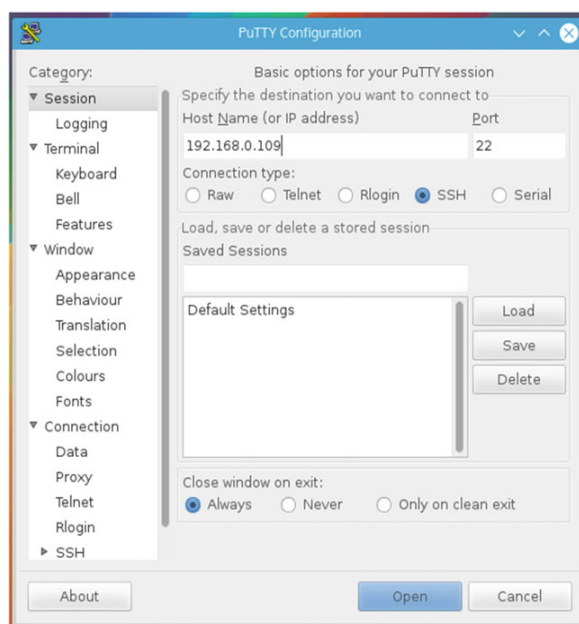
επιλέξει μεταξύ τριών εκδόσεων του Rasbian. Η βασική έκδοση διαθέτει γραφικό περιβάλλον χρήστη (GUI) και προεγκατεστημένα προγράμματα όπως τα LibreOffice, Scratch, Mathematica, Sonic Pi, κτλ. Υπάρχει, επίσης, έκδοση που περιλαμβάνει μόνο το γραφικό περιβάλλον χρήστη.

Η έκδοση του Rasbian που χρησιμοποιεί το Raspberry Pi του συστήματός μας, ονομάζεται “Lite”. Αποτελεί την πιο “ελαφριά” έκδοση, αφού δεν διαθέτει γραφικό περιβάλλον χρήστη ούτε εγκατεστημένα προγράμματα. Ο χειρισμός του υπολογιστή γίνεται μέσω της γραμμής εντολών (Command Line Interface) και είναι υπό τη μορφή εντολών κειμένου. Χρειάζονται βασικές γνώσεις προγραμματισμού σε Linux για τον χειρισμό του Raspberry Pi μέσω της έκδοσης Lite, γεγονός που καθιστά τη συγκεκριμένη έκδοση τη λιγότερο φιλική προς τον/την χρήστη. Η έκδοση Lite καταλαμβάνει, όπως γίνεται αντιληπτό, τον λιγότερο χώρο στην κάρτα microSD από τις υπόλοιπες εκδόσεις, μόλις 426Mb (τελευταία έκδοση, 10-07-2019), δηλαδή σχεδόν το ¼ του χώρου που καταλαμβάνει η βασική έκδοση.

Η διαχείριση του Raspberry Pi μπορεί να επιτευχθεί με δύο τρόπους. Ο πρώτος τρόπος είναι με τη σύνδεση εξωτερικής οθόνης και πληκτρολογίου στις θύρες mini-HDMI και micro-USB αντίστοιχα. Εναλλακτικά, υπάρχει η δυνατότητα απομακρυσμένου ελέγχου μέσω του ασφαλούς δικτυακού πρωτοκόλλου SSH (Secure Shell). Το πρωτόκολλο SSH επιτρέπει την μεταφορά δεδομένων μεταξύ δύο υπολογιστών που βρίσκονται στο ίδιο δίκτυο (Watkiss, 2016). Το Raspberry Pi διαθέτει εγκατεστημένο τον SSH διακομιστή. Στα λειτουργικά συστήματα Linux και Mac OSX είναι δυνατή η χρήση του SSH διακομιστή μέσω της γραμμής εντολών (Command line). Στο λειτουργικό σύστημα των Windows, όμως, είναι απαραίτητη η λήψη ενός εξωτερικού SSH διακομιστή όπως το λογισμικό ανοικτού κώδικα, PuTTY (Watkiss, 2016). Η σύνδεση μεταξύ των δύο υπολογιστών πραγματοποιείται μετά την εισαγωγή της διεύθυνσης διαδικτυακού πρωτοκόλλου (IP) του Raspberry Pi στο PuTTY. Το μοντέλο διαχείρισης που περιγράψαμε πιο πάνω ονομάζεται “Headless” και χρησιμοποιείται για τον έλεγχο του Raspberry Pi του συστήματος.

Για να πραγματοποιηθεί η εκκίνηση του Raspberry Pi Zero W σε headless mode για πρώτη φορά, απαιτείται μια συγκεκριμένη διαδικασία, τα βήματα της οποίας περιγράφονται αναλυτικά από τον Nelson (2018) και παρουσιάζονται στον Πίνακα 1 του Παραρτήματος 3. Μετά την αρχική εκκίνηση του Raspberry Pi, η διαδικασία σύνδεσης απλοποιείται σε μεγάλο βαθμό. Για τις επόμενες συνδέσεις χρειάζεται μόνο η εισαγωγή της IP του Raspberry Pi στο πρόγραμμα PuTTY και η εισαγωγή του ονόματος χρήστη και του κωδικού πρόσβασης στη γραμμή εντολών του Raspberry Pi.





*Εικόνα 12: Σύνδεση SSH διακομιστή μέσω του προγράμματος PuTTY. (Watkiss, 2016)*

Η πρώτη πληροφορία που δέχεται ο/η χρήστης με την είσοδό του στην γραμμή εντολών (Command Line) του λειτουργικού συστήματος Rasbian, είναι “`pi@raspberrypi ~ $`”. Το πρώτο μέρος της πρότασης “`pi`” είναι το όνομα χρήστη. Το “`raspberrypi`” είναι το όνομα του υπολογιστή (με προκαθορισμένο όνομα το “`raspberrypi`”). Το τρίτο μέρος δείχνει τον φάκελο στον οποίο βρίσκεται ο χρήστης τη δεδομένη στιγμή. Η εκκίνηση γίνεται πάντα στον home φάκελο “`~`”. Οποιαδήποτε εντολή πληκτρολογηθεί από τον/την χρήστη εμφανίζεται στα δεξιά του συμβόλου “`$`” (Membrey & Hows, 2015). Ο χειρισμός της γραμμής εντολών γίνεται μέσω κάποιων πολύ βασικών εντολών. Χωρίς αυτές τις εντολές, ο/η χρήστης είναι αδύνατον να περιηγηθεί στα αρχεία του συστήματος, ούτε έχει τη δυνατότητα να εκτελέσει κάποιο πρόγραμμα. Στον πίνακα 2 του Παραρτήματος 3 παρουσιάζονται οι βασικότερες, κατά τους Donat (2018), McManus & Cook (2017), εντολές του λειτουργικού συστήματος Linux.

### 2.2.2 Επιλογή γλώσσας γραφής προγραμματιστικού κώδικα

Η γραφή και σύνταξη του κώδικα του συστήματος έγινε στην γλώσσα προγραμματισμού C. Αμέσως, γεννάται το ερώτημα γιατί επιλέχθηκε η συγκεκριμένη γλώσσα προγραμματισμού έναντι κάποιας άλλης. Η απάντηση σε αυτό το ερώτημα είναι σύνθετη καθώς συντρέχουν δύο ειδών λόγοι, βάσει των οποίων έγινε η συγκεκριμένη επιλογή. Η πρώτη κατηγορία λόγων έχει να κάνει με τα εγγενή χαρακτηριστικά της γλώσσας C σε επίπεδο δομής κώδικα, και με τα πλεονεκτήματά της σε σύγκριση με άλλες, υψηλού επιπέδου, γλώσσες προγραμματισμού. Η δεύτερη κατηγορία, αποτελείται από εξωτερικούς παράγοντες που επηρέασαν, άλλοι λιγότερο και άλλοι περισσότερο, τη λήψη της απόφασης. Ας μελετήσουμε, λοιπόν, αυτούς τους λόγους πιο αναλυτικά.

Η C, είναι μια πολύ ευέλικτη γλώσσα προγραμματισμού που χρησιμοποιείται σε πληθώρα εφαρμογών διαφορετικής πολυπλοκότητας, από χαμηλού επιπέδου διαδικασίες ελέγχου υλισμικού σε μικροελεγκτές, μέχρι τη λειτουργία λειτουργικών συστημάτων όπως το Linux με γραφικό περιβάλλον χρήστη (GUI). Ταυτόχρονα, η γλώσσα C είναι σχετικά απλή στη χρήση της, αφού διαθέτει μόνο 20 περίπου λέξεις-κλειδιά, αλλά και πολλές βιβλιοθήκες συναρτήσεων που καλούνται ανάλογα με την κάθε εφαρμογή (Long, 2019).

Η μεγάλη διαφορά της γλώσσας C με άλλες γλώσσες υψηλού επιπέδου όπως η Python, είναι ότι ανήκει στην κατηγορία των συντάξιμων (compiled) γλωσσών, των οποίων για να εκτελεστεί ο κώδικας πρέπει πρώτα να συνταχθεί (compile) σε γλώσσα μηχανής από ένα πρόγραμμα compiler. Αντίθετα, η Python ανήκει στις λεγόμενες διερμηνευμένες (interpreted) γλώσσες οι οποίες δεν χρειάζονται σύνταξη (compilation) από εξωτερικό πρόγραμμα, αφού ο κώδικας τους μεταφράζεται τη στιγμή που γράφεται σε γλώσσα μηχανής (Boulanger & Lazzarini, 2011). Από τη μία πλευρά, οι γλώσσες αυτές διαθέτουν το πλεονέκτημα ότι το αποτέλεσμα του κώδικα τους εμφανίζεται σχεδόν άμεσα. Από την άλλη πλευρά, όμως, τα πλεονεκτήματα των συντάξιμων γλωσσών, όπως η C, είναι πολλά και σημαντικά. Πρώτον, από τη στιγμή που συντάσσεται ο κώδικας, μπορεί να “τρέξει” σε οποιονδήποτε υπολογιστή μέσω του εκτελέσιμου αρχείου. Επιπλέον, επισημαίνονται τυχόν λάθη στον κώδικα κατά τη διαδικασία σύνταξης του. Το σημαντικότερο πλεονέκτημα, όμως, των συντάξιμων γλωσσών έναντι των διερμηνευόμενων, είναι η πολύ μεγαλύτερη ταχύτητα με την οποία “τρέχουν” τους κώδικες. Αυτό συμβαίνει διότι η μετάφραση του κώδικα σε γλώσσα μηχανής έχει γίνει ήδη κατά τη διαδικασία σύνταξης (Long, 2019). Έτσι, για εφαρμογές ήχου, όπου υπάρχει μεγάλος

όγκος δεδομένων και απαιτούνται πολλοί υπολογισμοί, οι συντάξιμες γλώσσες όπως η C αποτελούν ιδανική επιλογή με την ταχύτητα που προσφέρουν.

Αναμφισβήτητα, τα πλεονεκτήματα που οφείλονται στα εγγενή χαρακτηριστικά της γλώσσας C, και ειδικά η ταχύτητα εκτέλεσης κώδικα, είχαν μεγάλη σημασία στην επιλογή της ως γλώσσα του προγραμματιστικού κώδικα του συστήματος. Όμως, άλλοι λόγοι συνέτρεξαν επίσης με κύριο παράδειγμα την ύπαρξη Open-Source κώδικα, σε γλώσσα προγραμματισμού C, της ιστοσελίδας (<https://www.electrosmash.com>), πάνω στον οποίο έχει βασιστεί μέρος του κώδικα του συστήματος. Η ύπαρξη του συγκεκριμένου κώδικα ήταν καθοριστικής σημασίας όχι μόνο στην επιλογή της C ως γλώσσα προγραμματισμού, αλλά και για τη λειτουργία του αρχικού συστήματος, θέτοντας τις βάσεις για την εξέλιξή του με την προσθήκη επιπλέον στοιχείων. Ένας άλλος λόγος που επηρέασε την απόφαση για χρήση της γλώσσας C, ήταν η ύπαρξη ανοικτών και ευκόλως προσβάσιμων βιβλιοθηκών που σχετίζονται αφενός με την εγγραφή και αναπαραγωγή των ψηφιακών δεδομένων ήχου και αφετέρου με την ασύρματη μετάδοση των δεδομένων μέσω Bluetooth. Γενικά, η γλώσσα C διαθέτει πλήθος διαθέσιμων βιβλιοθηκών στο διαδίκτυο, με επαρκή τεκμηρίωση και οδηγίες, καθώς και αρκετή υποστήριξη μέσω σχολίων και συμβουλών από χρήστες.

### **2.2.3 Εγκατάσταση βιβλιοθηκών BCM2835 και Pigpio. Δημιουργία, Σύνταξη και Εκτέλεση αρχείου**

Για τον έλεγχο των υποδοχών GPIO, καθώς και για την λειτουργία του SPI πρωτοκόλλου επικοινωνίας και της μεθόδου εξαγωγής δεδομένων PWM, απαιτείται η εγκατάσταση των κατάλληλων βιβλιοθηκών, μέσω των οποίων μπορούν να πραγματοποιηθούν οι παραπάνω λειτουργίες. Αυτές οι βιβλιοθήκες είναι η BCM2835 και η Pigpio. Παρόλο που οι συγκεκριμένες βιβλιοθήκες διεκπεραιώνουν κοινές λειτουργίες, στην παρούσα εργασία χρησιμοποιούνται και οι δύο σε συνεργασία για λόγους ευκολίας σύνταξης του κώδικα. Η επικοινωνία του A/D μετατροπέα MCP3202 με τον υπολογιστή Raspberry Pi, μέσω του SPI πρωτοκόλλου, γίνεται με χρήση της βιβλιοθήκης Pigpio. Ο λόγος που χρησιμοποιείται η εν λόγω βιβλιοθήκη είναι διότι διαθέτει ειδικές συναρτήσεις για την SPI επικοινωνία του συγκεκριμένου A/D μετατροπέα, με τον υπολογιστή Raspberry Pi. Από την άλλη πλευρά, η βιβλιοθήκη BCM2835 υποστηρίζει τη χρήση του PWM ως τρόπου εξαγωγής δεδομένων, προσφέροντας συναρτήσεις διαμόρφωσης και ελέγχου των δεδομένων PWM σε συγκεκριμένες

υποδοχές GPIO. Η διαχείριση των εξωτερικών εξαρτημάτων ένδειξης και ελέγχου των εφέ γίνεται, επίσης, μέσω της βιβλιοθήκης BCM2835.

Η λήψη και εγκατάσταση της βιβλιοθήκης BCM2835 γίνεται με την εισαγωγή του κώδικα του πίνακα 1, Παράρτημα 4, στη γραμμή εντολών του Raspberry Pi (McCauley, 2019), ενώ για την βιβλιοθήκη Pigpio χρησιμοποιείται ο κώδικας του πίνακα 2, Παράρτημα 4. (The Pigpio Library, 2019). Με εγκατεστημένες τις βιβλιοθήκες BCM2835 και Pigpio μπορούμε πλέον να δημιουργήσουμε αρχεία και να γράψουμε σε αυτά τις γραμμές κώδικα που επιθυμούμε. Για την δημιουργία ενός αρχείου μπορεί να χρησιμοποιηθεί ένα οποιοδήποτε πρόγραμμα επεξεργασίας κειμένου (text editor). Η δημιουργία και η επεξεργασία των κωδίκων, στην παρούσα εργασία, έγινε μέσω του προγράμματος Nano. Το Nano είναι ένα εύκολο στην χρήση πρόγραμμα επεξεργασίας κειμένου το οποίο διατίθεται ελεύθερα μαζί με τις εκδόσεις του Rasbian. Στην εικόνα 1 του Παραρτήματος 4, φαίνεται το περιβάλλον του Nano κατά τη δημιουργία ενός αρχείου.

Η διαδικασία και οι εντολές που χρειάζονται για τη δημιουργία, τη σύνταξη (compile) και την εκτέλεση ενός αρχείου, με χρήση των βιβλιοθηκών BCM2835 και Pigpio, παρουσιάζονται αναλυτικά στον πίνακα 3 του Παραρτήματος 4. Συνοπτικά, κατά τη δημιουργία γράφεται και αποθηκεύεται ο κώδικας του αρχείου. Για να είναι δυνατή η εκτέλεσή του στην γλώσσα C, πρέπει να προηγηθεί το στάδιο της σύνταξης. Σύνταξη (compilation), είναι η διαδικασία κατά την οποία ο κώδικας που έχει γραφτεί σε γλώσσα προγραμματισμού υψηλού επιπέδου, μετατρέπεται σε εκτελέσιμο αρχείο (.exe) γλώσσας μηχανής (Boulanger & Lazzarini, 2011). Τέλος, εφόσον έχει συνταχθεί το εκτελέσιμο αρχείο σωστά, χωρίς, δηλαδή, την εμφάνιση κάποιας ένδειξης σφάλματος (error) στη γραμμή εντολών, θα ακολουθήσει η εκτέλεση του αρχείου.

#### **2.2.4 Ασύρματη Bluetooth μετάδοση ήχου μέσω PulseAudio**

Για την επίτευξη ασύρματης μετάδοσης ηχητικού σήματος από τον μικροϋπολογιστή Raspberry Pi Zero W σε ένα κατάλληλο ηχείο Bluetooth, χρειάζεται πρώτα να γίνει σύνδεση μεταξύ των δύο συσκευών σε λογισμικό επίπεδο. Όπως είδαμε και στο κεφάλαιο 1.1.2, σε μια σύνδεση μεταξύ δύο συσκευών Bluetooth (ή αλλιώς piconet), η μια συσκευή έχει τον ρόλο υπεύθυνου κόμβου (master) ενώ η άλλη αναλαμβάνει ρόλο υποκόμβου (slave). Όπως γίνεται εύκολα αντιληπτό, ο υπολογιστής-μεταδότης λειτουργεί ως master ενώ το ηχείο-παραλήπτης

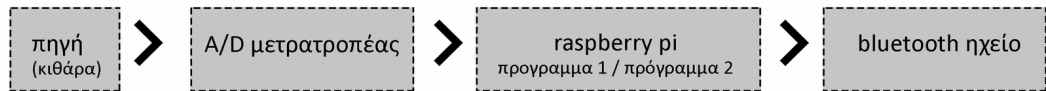
είναι slave. Για να επιτευχθεί η σύνδεση, λοιπόν, χρειάζεται να πραγματοποιηθούν με επιτυχία τα τρία στάδια: ερώτησης, δημιουργίας σύνδεσης και σύνδεσης.

Για το ηχείο Bluetooth, το μόνο που χρειάζεται ως slave συσκευή για να συνδεθεί με μια άλλη συσκευή, είναι να ενεργοποιηθεί η λειτουργία του Bluetooth μέσω ενός κουμπιού. Πιέζοντας το κουμπί, το ηχείο είναι έτοιμο για επικοινωνία με άλλες συσκευές έχοντας μια μοναδική IP την οποία μπορεί να χρησιμοποιήσει η master συσκευή. Για το Raspberry Pi, η διαδικασία είναι ελαφρώς πιο περίπλοκη σε σχέση με το ηχείο. Αρχικά, για να υπάρξει λειτουργία Bluetooth στο Raspberry Pi, θα πρέπει να είναι εγκατεστημένο το πρόγραμμα Bluez. Το Bluez, είναι το επίσημο πρόγραμμα Bluetooth για όλα τα λειτουργικά συστήματα Linux (άρα και για το Rasbian). Παρέχει πρόσβαση στα βασικά επίπεδα και προφίλ του Bluetooth. Το πρόγραμμα Bluez παρέχεται προεγκατεστημένο με το λειτουργικό σύστημα του Raspberry Pi, Rasbian. Για να γίνει η ενεργοποίηση του Bluetooth και η σύνδεση με μια συσκευή slave για πρώτη φορά, απαιτούνται μια σειρά από εντολές οι οποίες περιγράφονται στον πίνακα 4 του Παραρτήματος 4.

Μετά τη σύνδεση του Raspberry Pi με το ηχείο Bluetooth καθίσταται πλέον δυνατή η αποστολή δεδομένων από το πρώτο στο δεύτερο. Για την αναπαραγωγή ηχητικών δεδομένων στο Raspberry Pi χρειάζεται η εγκατάσταση του προγράμματος PulseAudio. Το PulseAudio είναι ένα πρόγραμμα ανοικτής προέλευσης (open-source) που λειτουργεί ως sound server, δηλαδή διαχειρίζεται τις ροές διαφορετικών ηχητικών δεδομένων και παρέχει τη δυνατότητα για μίξη αυτών σε ένα ενοποιημένο ηχητικό σήμα. Το PulseAudio παρέχεται από το πρότζεκτ freedesktop.org και έχει σχεδιαστεί για να τρέχει κυρίως σε λειτουργικά προγράμματα Linux (PulseAudio Documentation, n.d.). Η διαδικασία λήψης και εγκατάστασης του PulseAudio, καθώς και του PulseAudio Volume Control που χρησιμοποιείται για τον έλεγχο της έντασης του ήχου, περιγράφεται στον πίνακα 5 του Παραρτήματος 4.

Ένα μεγάλο πλεονέκτημα του PulseAudio είναι η ύπαρξη παραδειγμάτων για λειτουργίες ηχογράφησης και αναπαραγωγής ήχου με πλήθος διευκρινιστικών σχολίων. Σημαντικό είναι, επίσης, το γεγονός ότι μπορεί να υποστηρίξει διαφορετικές ρυθμίσεις για συχνότητα δειγματοληψίας (από 1Hz έως 192000Hz), βάθος δείγματος (8, 16, 24, 32 bits), μορφή δείγματος, και πλήθος καναλιών (έως 32 ξεχωριστά κανάλια). Παράλληλα προσφέρονται χρήσιμες συναρτήσεις για τον υπολογισμό διαφόρων μεγεθών και τιμών.

### 2.2.5 Προγραμματισμός Λογισμικού Συστήματος

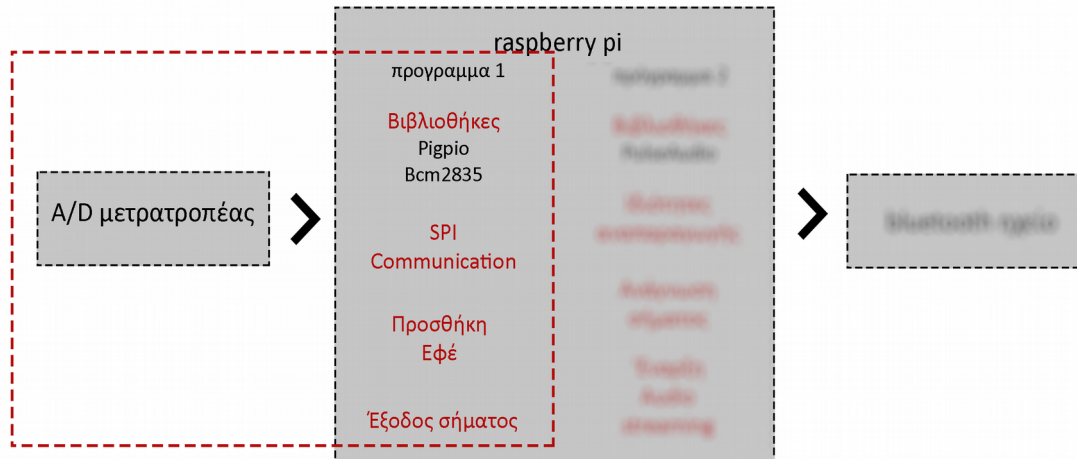


Σχήμα 1: Η διαδρομή που ακολουθεί το σήμα στο σύστημα

Έχοντας πλέον εγκαταστήσει τις απαραίτητες βιβλιοθήκες BCM2835 και Pigpio, καθώς και το πρόγραμμα PulseAudio, ακολουθεί ο προγραμματισμός του κυρίως κώδικα του λογισμικού συστήματος. Ο προγραμματισμός του κώδικα χωρίζεται σε τρία μέρη: στα δύο μέρη συμβαίνουν όλες οι σημαντικές λειτουργίες που αναφέρθηκαν πιο πάνω και αποτελούν ουσιαστικά το λογισμικό μέρος του συστήματος. Τα δύο αυτά μέρη θα τα ονομάσουμε “Πρόγραμμα 1 ή αλλιώς rawMCP3202” και “Πρόγραμμα 2 ή αλλιώς pulseaudio\_test”. Το τρίτο μέρος είναι ένα μικρό αρχείο κώδικα με ονομασία “rc.local” που λειτουργεί ως βοηθητικό στο σύστημα.

Η λειτουργία των τριών μερών του λογισμικού συστήματος συνοψίζεται ως εξής: Πρόγραμμα 1. Το ψηφιοποιημένο σήμα εξέρχεται του A/D μετατροπέα και με τη χρήση του SPI πρωτοκόλλου επικοινωνίας εισέρχεται στον μικροϋπολογιστή και αποθηκεύεται σε έναν “buffer”. Κατόπιν γίνεται η επεξεργασία του σήματος με το επιλεγμένο εφέ και το σήμα στέλνεται στο “standard output (stdout)”. Ο έλεγχος των παραμέτρων του εφέ γίνεται σε αυτό το πρόγραμμα. Με χρήση της τεχνικής “pipe” το σήμα αποστέλλεται από το “standard output” του προγράμματος 1, στο “standard input (stdin)” του προγράμματος 2. Πρόγραμμα 2. Ορίζονται τα δεδομένα της αναπαραγωγής. Στη συνέχεια ξεκινάει ένα “stream” μέσω του προγράμματος PulseAudio, όπου διαβάζονται τα δεδομένα από το “stdin” και αναπαράγονται από το Bluetooth ηχείο. Για την καλύτερη κατανόηση της λειτουργίας του λογισμικού συστήματος, θα μελετήσουμε το κάθε μέρος ξεχωριστά παραθέτοντας όπου χρειαστεί αποσπάσματα του κώδικα που χρησιμοποιήθηκε.

## Μελέτη του Προγράμματος 1 (rawMCP3202)



Σχήμα 2: Γραφική αναπαράσταση της λειτουργίας του προγράμματος 1

Το πρόγραμμα 1 είναι βασισμένο σε ένα παράδειγμα που υπάρχει στην ιστοσελίδα της βιβλιοθήκης Pigpio (<http://abyz.me.uk/rpi/pigpio/examples.html>). Το συγκεκριμένο παράδειγμα ονομάζεται “SPI bit bang MCP3202” και χρησιμοποιείται για την ταυτόχρονη ανάγνωση πολλών A/D μετατροπέων τύπου MCP3202, μέσω του πρωτοκόλλου SPI, με ακρίβεια της τάξης του ενός εκατομμυριοστού του δευτερολέπτου. Για να επιτύχει τόσο μεγάλη ακρίβεια στη μετάδοση δεδομένων με το πρωτόκολλο SPI, χρησιμοποιείται η τεχνική bit banging. Το Bit banging αποτελεί μια μέθοδο μετάδοσης δεδομένων, κατά την οποία η διαδικασία μετάδοσης πραγματοποιείται εξολοκλήρου μέσω του λογισμικού συστήματος, αντί του καθορισμένου υλισμικού συστήματος (Odunlade, 2018). Αυτό, όμως, έχει σαν αποτέλεσμα την κατανάλωση περισσότερης επεξεργαστικής ισχύος σε σχέση με τη χρήση του καθορισμένου υλισμικού, ενώ μειώνεται και η συχνότητα δειγματοληψίας του συστήματος.

Η χρήση της τεχνικής bit banging είναι πολύ σημαντική για τη λειτουργία του συστήματος της παρούσας εργασίας. Αυτό συμβαίνει διότι ο μικροϋπολογιστής Raspberry pi, με το βασισμένο σε Linux λειτουργικό του σύστημα, δεν μπορεί να πραγματοποιήσει ροή δεδομένων μέσω SPI πρωτοκόλλου, με μεγάλη ακρίβεια στην εκτέλεση του βρόγχου. Ο υπολογιστής πρέπει να αναλάβει ρόλο master στην SPI επικοινωνία, εκτός ειδικών περιπτώσεων A/D μετατροπέων που έχουν αυτή τη δυνατότητα, ορίζοντας τη συχνότητα

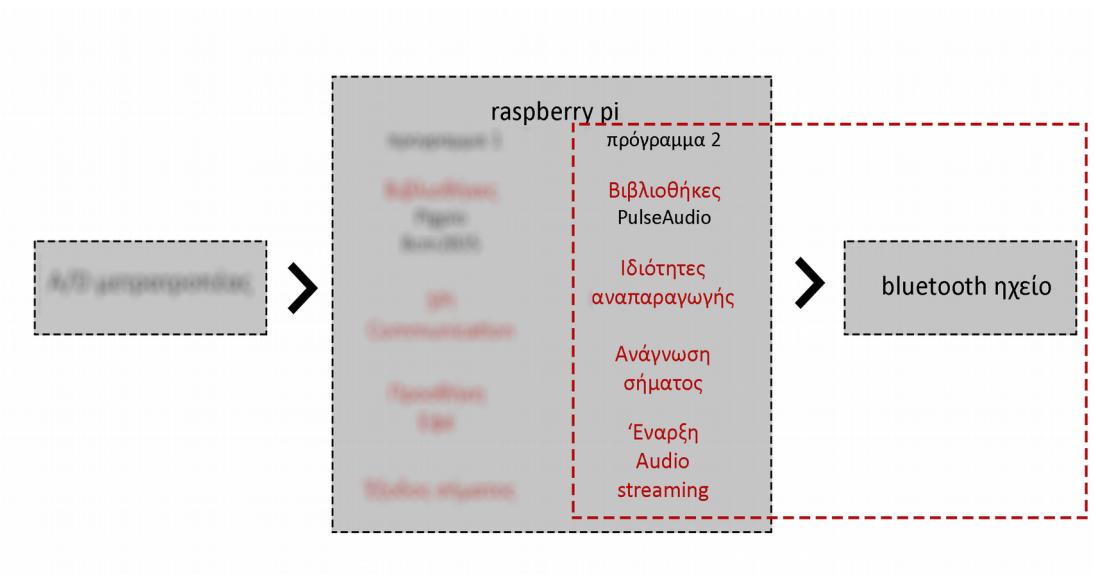
δειγματοληψίας με την οποία μεταφέρονται τα δεδομένα. Στην περίπτωση του Raspberry pi, η διενέργεια άλλων διεργασιών του λειτουργικού συστήματος με μη σταθερή κατανάλωση ισχύος για κάθε χρονική στιγμή, δεν επιτρέπει τη ροή δεδομένων με σταθερή συχνότητα δειγματοληψίας. Καθίσταται, λοιπόν, επιτακτική η ανάγκη για χρήση της τεχνικής bit banging, παρόλο που συνεπάγεται μεγάλη μείωση της συχνότητας δειγματοληψίας του συστήματος.

Στην αρχή του προγράμματός δηλώνονται, όπως γίνεται πάντα σε κώδικες της γλώσσας C, οι βιβλιοθήκες που θα χρησιμοποιηθούν από το πρόγραμμα με χρήση της εντολής “#include”, ενώ καθορίζονται οι υποδοχές GPIO που αφορούν την είσοδο και την έξοδο δεδομένων μέσω της εντολής “#define”. Στη συνέχεια, καθορίζονται οι GPIO υποδοχές που χρησιμοποιούνται για την επικοινωνία με τον A/D μετατροπέα, μέσω του SPI πρωτοκόλλου, και οι υποδοχές μέσω των οποίων πραγματοποιείται η μετάδοση δεδομένων μέσω της μεθόδου PWM. Το PWM είναι μια μέθοδος αποστολής δεδομένων ήχου και χρησιμοποιείται στο σύστημά μας για να δώσει στους/στις χρήστες τη δυνατότητα για ενσύρματη αναπαραγωγή του ήχου από μεμονωμένο ενισχυτή. Μέρη του κώδικα που αφορούν αυτές τις λειτουργίες παραθέτονται στον πίνακα 1, Παράρτημα 5.

Κατά τη διαδικασία του κυρίως βρόγχου, τα raw ψηφιακά δεδομένα ήχου που παράγει ο A/D μετατροπέας, εισέρχονται σε έναν buffer υπό τη μορφή bit blocks, ως αποτέλεσμα της SPI επικοινωνίας μέσω bit banging. Στη συνέχεια, γίνεται επεξεργασία των δεδομένων μέσω χρήσης αλγόριθμου εφέ. Σε αυτή τη μορφή του κώδικα χρησιμοποιείται ένα εφέ παραμόρφωσης (distortion) όπως αυτό φαίνεται στον πίνακα 2, Παράρτημα 5. Η δομή του αλγόριθμου του εφέ είναι πολύ απλή, αλλά αποτελεσματική. Το σήμα εισάγεται στον αλγόριθμό και όταν η τιμή του ξεπεράσει μια προκαθορισμένη τιμή κατωφλίου, κόβεται στο επίπεδο του κατωφλίου. Το ηχητικό αποτέλεσμα αυτής της διαδικασίας είναι το λεγόμενο εφέ παραμόρφωσης. Η ενεργοποίηση του εφέ και ο έλεγχος των παραμέτρων γίνεται μέσω του κώδικα του πίνακα 3, Παράρτημα 5. Το σήμα στη συνέχεια εγγράφεται στο stdout του προγράμματος. Μέσω μιας εντολής “pipe”, τα δεδομένα από την έξοδο του προγράμματος 1 καταλήγουν στην είσοδο, stdin, του προγράμματος 2.

## **Μελέτη του Προγράμματος 2 (pulseaudio\_test)**





Σχήμα 3. Γραφική αναπαράσταση της λειτουργίας του προγράμματος 2

Το πρόγραμμα 2 είναι βασισμένο σε ένα παράδειγμα που υπάρχει στην ιστοσελίδα του προγράμματος PulseAudio ([https://freedesktop.org/software/pulseaudio/doxygen/pacat-simple\\_8c-example.html](https://freedesktop.org/software/pulseaudio/doxygen/pacat-simple_8c-example.html)). Το παράδειγμα ονομάζεται “pacat-simple.c” και χρησιμοποιείται για την ανάγνωση και αναπαραγωγή ενός raw αρχείου ήχου. Η δομή του κώδικα του προγράμματος 2 είναι πολύ απλή. Ειδικότερα, αφού δηλωθούν οι κατάλληλες βιβλιοθήκες για τις ανάγκες του προγράμματος, ορίζονται παράμετροι όπως συχνότητα αναπαραγωγής, βάθος δείγματος και πλήθος καναλιών. Στη συνέχεια, ξεκινάει ένας βρόγχος και δημιουργείται ένα playback stream (αναπαραγωγή ροής δεδομένων), όπου εισέρχεται το σήμα από το stdin. Κατά τη διάρκεια λειτουργίας του βρόγχου, διαβάζεται συνεχόμενα το ηχητικό σήμα και αναπαράγεται στην καθορισμένη έξοδο. Ο κώδικας των λειτουργιών που περιγράφηκαν πιο πάνω, παραθέτεται στον πίνακα 4, Παράρτημα 5.

### Βοηθητικό πρόγραμμα “rc.local”

Το πρόγραμμα με όνομα “rc.local” περιέχει τις εντολές (βλέπε πίνακας 5, Παράρτημα 5) που χρειάζονται για να επιτευχθεί η αυτοματοποιημένη έναρξη του συστήματος κατά την ενεργοποίηση του Raspberry Pi, χωρίς να απαιτείται σύνδεση με υπολογιστή. Ειδικότερα, το πρόγραμμα είναι υπεύθυνο για: την έναρξη του προγράμματος Pulseaudio, τη σύνδεση του Raspberry Pi με το ηχείο Bluetooth και την εκτέλεση των προγραμμάτων “rawMCP3202” και “pulseaudio\_test”.

### **3. Συμπεράσματα, περιορισμοί εργασίας και προτάσεις για μελλοντική έρευνα**

Η παρούσα εργασία επιχείρησε τη μελέτη, τη κατασκευή και τον προγραμματισμό ενός υλισμικού/λογισμικού συστήματος, που θα δίνει τη δυνατότητα στον/στην χρήστη/μουσικό να αλλάζει τον ήχο του μουσικού οργάνου του/της, χρησιμοποιώντας διάφορα εφέ, και στη συνέχεια να αναπαράγει το τροποποιημένο σήμα μέσω ενός κατάλληλου ηχείου Bluetooth. Το εγχείρημα αυτό μπορεί να κριθεί επιτυχημένο, καθώς πέτυχε τους βασικούς του σκοπούς. Ειδικότερα, το υλισμικό/λογισμικό σύστημα τέθηκε σε λειτουργία, μπορούσε να λάβει το ηχητικό σήμα του μουσικού οργάνου, να το τροποποιήσει προσθέτοντας εφέ και να το στείλει μέσω Bluetooth στο κατάλληλο ηχείο Bluetooth. Παράλληλα, οι χρήστες είχαν και τη δυνατότητα ενσύρματης αναπαραγωγής του ήχου από μεμονωμένο ενισχυτή. Το εγχείρημα πραγματοποιήθηκε χρησιμοποιώντας τεχνολογία ανοιχτής προέλευσης, γεγονός που επιβεβαιώνει την αναπαραγωγιμότητα της προσπάθειας αυτής και σε συνδυασμό με το χαμηλό κόστος κατασκευής επιτρέπει σε τρίτους/τρίτες να το επιχειρήσουν. Ακόμα, τα εφέ μπορούν να δεχθούν τροποποιήσεις, ώστε να προσαρμοστούν στις εκάστοτε ανάγκες του/της κάθε χρήστη. Το σύστημα είναι αυτόνομο όσον αφορά την τροφοδοσία του, αφού μπορεί να λειτουργήσει με πακέτο μπαταριών τάσης 5V. Παρόλα αυτά, είναι δυνατή η τροφοδοσία του με ενσύρματο τροφοδοτικό αντίστοιχης τάσης. Ο σχεδιασμός του εξωτερικού μέρους του συστήματος, που περιλαμβάνει τα εξαρτήματα ελέγχου των παραμέτρων των εφέ, έγινε με τέτοιο τρόπο ώστε το σύστημα να είναι φιλικό προς τον/την χρήστη, ενώ οι σχετικά μικρές του διαστάσεις επιτρέπουν την εύκολη μεταφορά του.

Ωστόσο, υπήρξαν κάποιοι παράγοντες, που περιόρισαν και διαφοροποίησαν σε κάποια σημεία τους αρχικούς στόχους της εργασίας. Ένας από τους βασικούς περιορισμούς του συστήματος είναι η χαμηλή ποιότητα του τελικού ηχητικού αποτελέσματος, σε συνδυασμό με την ύπαρξη καθυστέρησης (latency) και θορύβου στο σήμα. Η χαμηλή ποιότητα του ήχου οφείλεται στις επιλογές των υλισμικών εξαρτημάτων και των αλγορίθμων του λογισμικού συστήματος. Εξαρχής οι προσδοκίες του συστήματος αναφορικά με την ποιότητα και την πιστότητα του ήχου ήταν συγκριμένες και αναλογούσαν στην έκταση και τις δυνατότητες μιας διπλωματικής εργασίας. Ωστόσο κρίνεται σημαντικό να αναφερθούν οι λόγοι που οδήγησαν στο συγκεκριμένο αποτέλεσμα. Καταρχάς, η επιλογή του A/D μετατροπέα (MCP3202) και της μονάδας επεξεργασίας του συστήματος (Raspberry Pi Zero W), ήταν καθοριστικής σημασίας για την ποιότητα του τελικού ηχητικού αποτελέσματος. Τα συγκεκριμένα δύο μέρη επιλέχτηκαν

μεν για το χαμηλό κόστος αγοράς τους και για τις μικρές τους διαστάσεις, αλλά κυρίως γιατί κρίθηκε σε θεωρητικό επίπεδο ότι θα συνδυάζουν την ευκολία χρήσης τεχνικά χαρακτηριστικά. Στην πράξη, όμως, αποδείχθηκε πως ήταν πολύ δύσκολο να επιτευχθεί συνεχής ροή δεδομένων (data streaming) σε υψηλές ταχύτητες μέσω του πρωτοκόλλου επικοινωνίας SPI, από τον A/D μετατροπέα στη μονάδα επεξεργασίας. Το πρόβλημα εστιάζεται στο γεγονός ότι ο υπολογιστής Raspberry pi διαθέτει λειτουργικό σύστημα, το οποίο εκτελεί άλλες διεργασίες που δεσμεύουν επεξεργαστική ισχύ, χωρίς όμως να είναι πάντα σταθερή η ποσότητα ισχύος που δεσμεύεται κάθε χρονική στιγμή. Συνεπώς, καθίσταται αδύνατη η ύπαρξη μεγάλης ακρίβειας στην εκτέλεση του βρόγχου του SPI πρωτοκόλλου όταν έχει ρόλο master το Raspberry pi. Μια πιθανή λύση σε αυτό το πρόβλημα θα ήταν να αναλάβει ο A/D μετατροπέας ρόλο master κόμβου, ορίζοντας μια σταθερή συχνότητα δειγματοληψίας για την SPI επικοινωνία. Τέτοιοι A/D μετατροπείς, όμως, είναι δυσεύρετοι και έχουν αρκετά μεγαλύτερο κόστος αγοράς σε σχέση με τον MCP3202. Η λύση που τελικά προτιμήθηκε δεν περιλάμβανε την αντικατάσταση κάποιου εξαρτήματος, αλλά ήταν λογισμικού χαρακτήρα, με χρήση της τεχνικής Bit banging. Το Bit banging προσέφερε την απαραίτητη ακρίβεια που απαιτούσε η ροή δεδομένων μεταξύ του A/D μετατροπέα και του Raspberry pi, μειώνοντας, όμως, ταυτόχρονα τη συχνότητα δειγματοληψίας του συστήματος, με αποτέλεσμα τον περιορισμό της ποιότητας του τελικού ηχητικού αποτελέσματος.

Η ύπαρξη θορύβου στον παραγόμενο ήχο του συστήματος οφείλεται σε περιορισμούς τόσο του υλισμικού, όσο και του λογισμικού συστήματος. Ειδικότερα, η τροφοδοσία του συστήματος περιέχει θόρυβο, όπου παρά το φιλτράρισμα που γίνεται μέσω των αναλογικών φίλτρων, ένα ποσοστό του περνάει στο τελικό σήμα. Επίσης, υπάρχει ψηφιακός θόρυβος λόγω διεργασιών του λογισμικού συστήματος όπως το Wi-Fi. Συνολικά, το επίπεδο θορύβου μπορεί να περιοριστεί μελλοντικά με τη χρήση περισσότερων και καλύτερων αναλογικών και ψηφιακών φίλτρων. Σε αυτό το σημείο πρέπει να αναφερθεί πως το αναλογικό κύκλωμα κατασκευάστηκε σε πρωτότυπη πλακέτα, για λόγους ευκολίας και αμεσότητας κατασκευής, αλλά και λόγω περιορισμού του συνολικού κόστους. Η κατασκευή του κυκλώματος σε μορφή πλακέτας τυπωμένου κυκλώματος (PCB), θα βελτιώσει την συνολική ποιότητα κατασκευής του συστήματος, μειώνοντας το θόρυβο και εξασφαλίζοντας την ακεραιότητα του σήματος.

Αναφορικά με την ύπαρξη σημαντικής καθυστέρησης (latency) στο σήμα, οι λόγοι που προκαλούν αυτό το φαινόμενο προκύπτουν κατά μεγάλο ποσοστό από τη λειτουργία του λογισμικού συστήματος. Είναι γεγονός πως οι αλγόριθμοι του συστήματος επιδέχονται αλλαγές που μπορούν να βελτιστοποιήσουν τη λειτουργία τους. Ωστόσο, λόγω του συγκεκριμένου

χρονικού πλαισίου για την εκπόνηση της εργασίας και σε συνδυασμό με την έλλειψη εμπειρίας μας στον προγραμματισμό, κατέστη αδύνατη η εξάλειψη αυτής της καθυστέρησης.

Η εφαρμογή του συστήματος περιορίστηκε στην ηλεκτρική κιθάρα, καθώς η δοκιμή περισσότερων οργάνων θα απαιτούσε χρόνο και πόρους σε επίπεδο υλικών, αφού θα χρειαζόταν προσαρμογή του κυκλώματος εισόδου για να δέχεται διαφορετικές πηγές εισόδου. Ταυτόχρονα, οι απαιτήσεις σε προγραμματιστικό κώδικα θα αυξάνονταν, επιβαρύνοντας με αυτό τον τρόπο τη λειτουργία του συστήματος. Συνεπώς, δεν υπάρχει η δυνατότητα γενίκευσης των αποτελεσμάτων του συστήματος. Δίνεται όμως το έναυσμα για περαιτέρω ενασχόληση και βελτίωση του συστήματος, ώστε μελλοντικά να μπορέσει να απευθυνθεί σε μεγαλύτερο και πιο ετερογενές κοινό. Ένα ακόμα στοιχείο του συστήματος που χρήζει βελτίωσης είναι ο αριθμός των εφέ που μπορούν να είναι ταυτόχρονα σε λειτουργία, καθώς το εν λόγω σύστημα δίνει τη δυνατότητα επιλογής μόνο ενός εφέ τη φορά. Το σύστημα προγραμματίστηκε με ένα εφέ ενεργό τη φορά και μάλιστα με απλή δομή αλγόριθμου, διότι σε αυτή τη φάση κρίθηκε προτιμότερο να μην υπάρξει περαιτέρω επιβάρυνση του επεξεργαστή από πιο πολύπλοκους αλγόριθμους, οι οποίοι θα μπορούσαν, πιθανώς, να προκαλέσουν αστάθεια στη λειτουργία του. Ωστόσο, δίνεται η δυνατότητα υιοθέτησης αυτής της ιδέας, η οποία, σε συνδυασμό με την τροποποίηση του εξωτερικού ελέγχου των παραμέτρων των εφέ, θα μπορούσε να βελτιώσει τη χρηστικότητα του συστήματος και παράλληλα να επεκτείνει τις δυνατότητές του.

Συνοπτικά, δίνονται παρακάτω κάποιες προτάσεις για μελλοντική έρευνα που μπορούν να συμβάλουν στη βελτίωση και περαιτέρω εξέλιξη του συστήματος. Αρχικά, θα μπορούσε θεωρητικά να επιτευχθεί καλύτερη ποιότητα ήχου με το Raspberry Pi ως επεξεργαστή του συστήματος, με τη χρήση ενός Λειτουργικού Συστήματος Πραγματικού Χρόνου (RTOS) αντί του Rasbian. Με αυτόν τον τρόπο, το σύστημα θα αποκτούσε μεγαλύτερη ακρίβεια στη διεκπεραίωση των υπολογισμών, μειώνοντας έτσι τη διακύμανση (jitter) και το επίπεδο θορύβου. Εναλλακτικά, η χρήση ενός μικροελεγκτή με ενσωματωμένο A/D μετατροπέα, όπως το Arduino στη θέση του Raspberry Pi, θα εξάλειφε οποιοδήποτε πρόβλημα συνδεσιμότητας μεταξύ των δύο. Ωστόσο, η επεξεργαστική ισχύς ενός μικροελεγκτή σαν το Arduino δεν μπορεί να συγκριθεί με αυτή ενός υπολογιστή όπως το Raspberry Pi. Τέλος, όσον αφορά την καταλληλότητα των πρωτοκόλλων ασύρματης επικοινωνίας, μια σύγκριση μεταξύ του Bluetooth και άλλων πρωτοκόλλων, όπως το Wi-Fi, με τα δεδομένα του παρόντος συστήματος, θα μπορούσε να αποτελέσει αντικείμενο μελέτης για μελλοντικές έρευνες.

## 4. Επίλογος

Η παρούσα εργασία παρουσιάζει ένα εγχείρημα, το οποίο αφορμάται από τις τεχνολογικές εξελίξεις στο χώρο της μουσικής και ταυτόχρονα επιχειρεί να ανταποκριθεί στις ανάγκες που θέτει η σύγχρονη κοινωνία. Παρουσιάζει τεχνικές και μεθόδους για την επίτευξη ασύρματης ροής ηχητικών δεδομένων σε ζωντανό χρόνο, ενώ ταυτόχρονα γίνεται προσπάθεια να παραμείνει το σύστημα ευέλικτο και εύκολο στη χρήση. Η υλοποίηση του συστήματος έγινε στο πλαίσιο εκπόνησης της Διπλωματικής εργασίας, με δεδομένους τους περιορισμούς σε υλικά και χρόνο. Ωστόσο, τα αποτελέσματα που αναλύθηκαν και τα συμπεράσματα που εξήχθησαν, αποτελούν παρακαταθήκη για μελλοντικές έρευνες. Στόχος είναι η παρούσα εργασία να αποτελέσει μια μορφή καθοδήγησης, σε άτομα που πιθανόν μελλοντικά να επιχειρήσουν να ερευνήσουν περαιτέρω τα ζητήματα που τέθηκαν. Καθώς η εργασία προέκυψε ύστερα από προσωπικό προβληματισμό και ενδιαφέρον πάνω στο θέμα, αποτελεί το πρώτο βήμα που θα τροφοδοτήσει την περαιτέρω ενασχόληση για βελτιώσεις τόσο από άποψη ποιότητας του ήχου, όσο και από άποψη δυνατοτήτων και παροχών στον/στην χρήστη.

## 5. Βιβλιογραφία

- Boulanger, R. & Lazzarini, V. (2011). *The Audio Programming Book*. Cambridge, MA: The MIT Press
- Donat, W. (2018). *Learn Raspberry Pi Programming with Python: Learn to Program on the World's Most Popular Tiny Computer*. Berkeley, CA: Apress
- Floros, A., Koutroubas, M., Tatlas, N.A., & Mourjopoulos, J. (2002). *A Study of Wireless Compressed Digital-Audio Transmission*. presented at the Audio Engineering Society 112<sup>th</sup> Convention, Munich, May 2002 (preprint 5516)
- Floros, A., Tatlas, N.A., & Mourjopoulos, J. (2006). *A High-Quality Digital Audio Delivery Bluetooth Platform*. Consumer Electronics, IEEE Transactions on. 52. 909 – 916. 10.1109/TCE.2006.1706488.
- Hoene, C. and Hyder, M. (2009). *Considering bluetooth's subband codec (SBC) for wideband speech and audio on the internet*: Technical report, Universitätsbibliothek Tübingen, Germany.
- Hoene, C. and Hyder, M. (2010). *Optimally Using the Bluetooth Subband Codec*. Proceedings of the IEEE 35th Conference on Local Computer Networks, pp. 356-359
- Horowitz, P. & Hill, W. (2015). *The Art of Electronics (3<sup>rd</sup> ed.)*. New York, NY: Cambridge University Press
- Jung, W. (2005). *Op Amp Applications Handbook*. Burlington, MA: Newnes
- Kuznetsov, S. & Paulos, E. (2010). *Rise of the expert amateur: DIY projects, communities, and cultures*. In Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries (NordiCHI '10). ACM, New York, NY, 295-304. DOI=(<http://dx.doi.org/10.1145/1868914.1868950>)
- Long, S. (2019). *An Introduction to C & GUI Programming*. Cambridge, UK: Raspberry Pi Press
- Long, S. (n.d.). *Learn To Code With C: The MagPi Essentials*. Cambridge, UK: Raspberry Pi Press

- McManus, S. & Cook, M. (2017). *Raspberry Pi for Dummies (3<sup>rd</sup> ed.)*. Hoboken, NJ: John Wiley & Sons
- Membrey, P. & Hows, D. (2015). *Learn Raspberry Pi 2 with Linux and Windows 10*. New York, NY: Apress
- Miller, Michael (2001). *Discovering Bluetooth*. Alameda, CA: SYBEX, Inc.
- Richardson, M. & Wallace, S. (2013). *Getting Started with Raspberry Pi*. Sebastopol, CA: O'Reilly Media
- Stallman, R. (2015). *Free Software, Free Society (3<sup>rd</sup> Ed.)*. Boston, MA: Free Software Foundation
- Upton, E. & Halfacree, G. (2016). *Ο μικροπολογιστής Raspberry Pi Zero*. UK: Wiley & Sons Ltd.
- Watkiss, Stewart. (2016). *Learn Electronics with Raspberry Pi: Physical Computing with Circuits, Sensors, Outputs, and Projects*. Redditch, UK: Apress Media

#### Διαδίκτυογραφία – Ηλεκτρονικές πηγές

- *A Brief History of Effects Pedals*. (2017). Retrieved September 18, 2019, from: <https://www.sunpower-uk.com/news/a-brief-history-of-guitar-effects-pedals-infographic/>
- Benchoff, B. (2016). Τα μοντέλα Raspberry Pi με τα χαρακτηριστικά τους. Retrieved September 22, 2019, from: <https://hackaday.com/2016/02/28/introducing-the-raspberry-pi-3/>
- Bluegiga Technologies. (2014). *Bluetooth® A2DP and AVRCP Profiles – iWRAP Application Note: Manual*. Retrieved from: [http://mt-system.ru/sites/default/files/documents/iwr\\_ap6\\_a2dp\\_application\\_note.pdf](http://mt-system.ru/sites/default/files/documents/iwr_ap6_a2dp_application_note.pdf)
- *Bluetooth Basics*. (n.d.). Retrieved September 12, 2019, from: <https://learn.sparkfun.com/tutorials/bluetooth-basics>
- Bluetooth SIG. (2012). *A2DC\_SPEC: Manual*. Retrieved from [https://www.dslreports.com/r0/download/2285126~a70eb148e16b921dc323dbb977d4b4b1/A2DP\\_SPEC.pdf](https://www.dslreports.com/r0/download/2285126~a70eb148e16b921dc323dbb977d4b4b1/A2DP_SPEC.pdf)

- Εφέ κιθάρας σε μορφή πεταλ. n.d. Retrieved September 18, 2019, from: <https://www.proaudioeurope.com/outboard/delays-reverbs/effect-pedals/bae-hot-fuzz-guitar-pedal.html>
- *Effects Unit*. (n.d.). In Wikipedia. Retrieved September 18, 2019, from: [https://en.wikipedia.org/wiki/Effects\\_unit#History](https://en.wikipedia.org/wiki/Effects_unit#History)
- Keim, R. (2016). *Turn Your PWM into a DAC*. Retrieved September 24, 2019, from: <https://www.allaboutcircuits.com/technical-articles/turn-your-pwm-into-a-dac/>
- Kunde, R. (2014). *Bluetooth*. Retrieved September 12, 2019, from: <https://www.slideshare.net/RajuYadav1/bluetooth-38627994>
- *Linux*. (n.d.). In Wikipedia. Retrieved September 27, 2019, from: <https://el.wikipedia.org/wiki/Linux>
- Longman, J. (2019). Οι εκδόσεις του Bluetooth μέχρι σήμερα και τα χαρακτηριστικά τους. Retrieved from: <https://www.audioreputation.com/outdoor-bluetooth-speakers/>
- McCauley, M. (n.d.). *C library for Broadcom BCM 2835 as used in Raspberry Pi: Manual*. Retrieved September 28, 2019, from: <http://www.airspayce.com/mikem/bcm2835/>
- Nelson, C. (2018). *Raspberry Pi Zero Headless Quick Start: Manual*. Retrieved September 27, 2019, from: <https://learn.adafruit.com/raspberry-pi-zero-creation>
- Odunlade, E. (2018). *Introduction to Bit Banging: SPI communication via Bit Banging*. Retrieved October 16, 2019, from: <https://circuitdigest.com/article/introduction-to-bit-banging-spi-communication-in-arduino-via-bit-banging>
- Οι υποδοχές GPIO στο Raspberry Pi Zero W. (2019). Retrieved September 22, 2019, from: <https://pi4j.com/1.2/pins/model-zerow-rev1.html>
- *Open-source Software*. (n.d.). In Wikipedia. Retrieved September 24, 2019, from: [https://en.wikipedia.org/wiki/Open-source\\_software](https://en.wikipedia.org/wiki/Open-source_software)
- Πολυεφέ κιθάρας σε μορφή πεταλιέρας. n.d. Retrieved September 18, 2019, from: <https://www.long-mcquade.com/23265/Guitars/Guitar-Effects/Line-6/Guitar-Multi-Effects-Processor.htm>
- *The Pigiopio Library*. (n.d.). Retrieved October 10, 2019, from: <http://abyz.me.uk/rpi/pigpio/index.html>

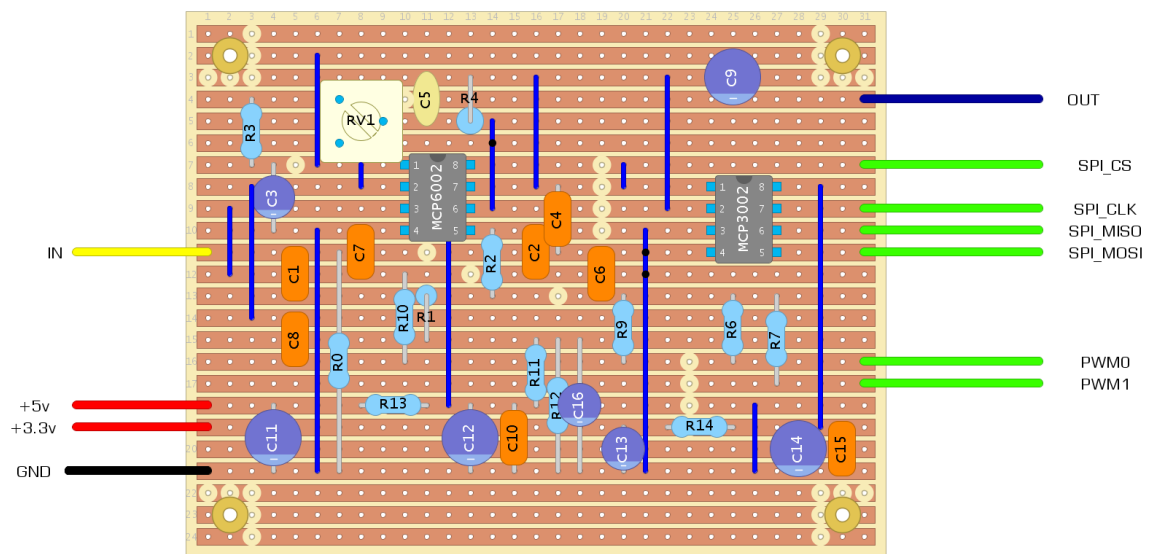


- *Proprietary Software*. (n.d.). Retrieved September 24, 2019, from: <https://www.techopedia.com/definition/4333/proprietary-software>
- *Pull-up Resistors*. (n.d.). Retrieved October 2, 2019, from: <https://www.electronics-tutorials.ws/logic/pull-up-resistor.html>
- *PulseAudio Documentation*. (n.d.). Retrieved October 7, 2019, from: <https://freedesktop.org/software/pulseaudio/doxygen/index.html>
- *Raspberry Pi models comparison*. (2019) Retrieved September 18, 2019, from: <http://socialcompare.com/en/comparison/raspberrypi-models-comparison>
- Τα βασικά στοιχεία του Raspberry Pi Zero W. n.d. Retrieved September 22, 2019, from: <https://raspberrypi.australia.com.au/raspberry-pi-zero-w>
- *The open source way*. (n.d.) Retrieved September 24, 2019, from: <https://opensource.com/open-source-way?fbclid=IwAR3PRwtb0AxIn4Vjtc1a1ZoFf3oZeLUHrYbDI9WfACnBAFodQyRwdA64-VQ>
- *What is open-source?* (n.d.). Retrieved September 24, 2019, from: [https://opensource.com/resources/what-open-source?fbclid=IwAR04FVgyDZirpJKy\\_LOWsn\\_Zpi64Z6FoTVIFWcMTwSqKGz9mc9-Ff4A-pyo](https://opensource.com/resources/what-open-source?fbclid=IwAR04FVgyDZirpJKy_LOWsn_Zpi64Z6FoTVIFWcMTwSqKGz9mc9-Ff4A-pyo)

# Παραρτήματα

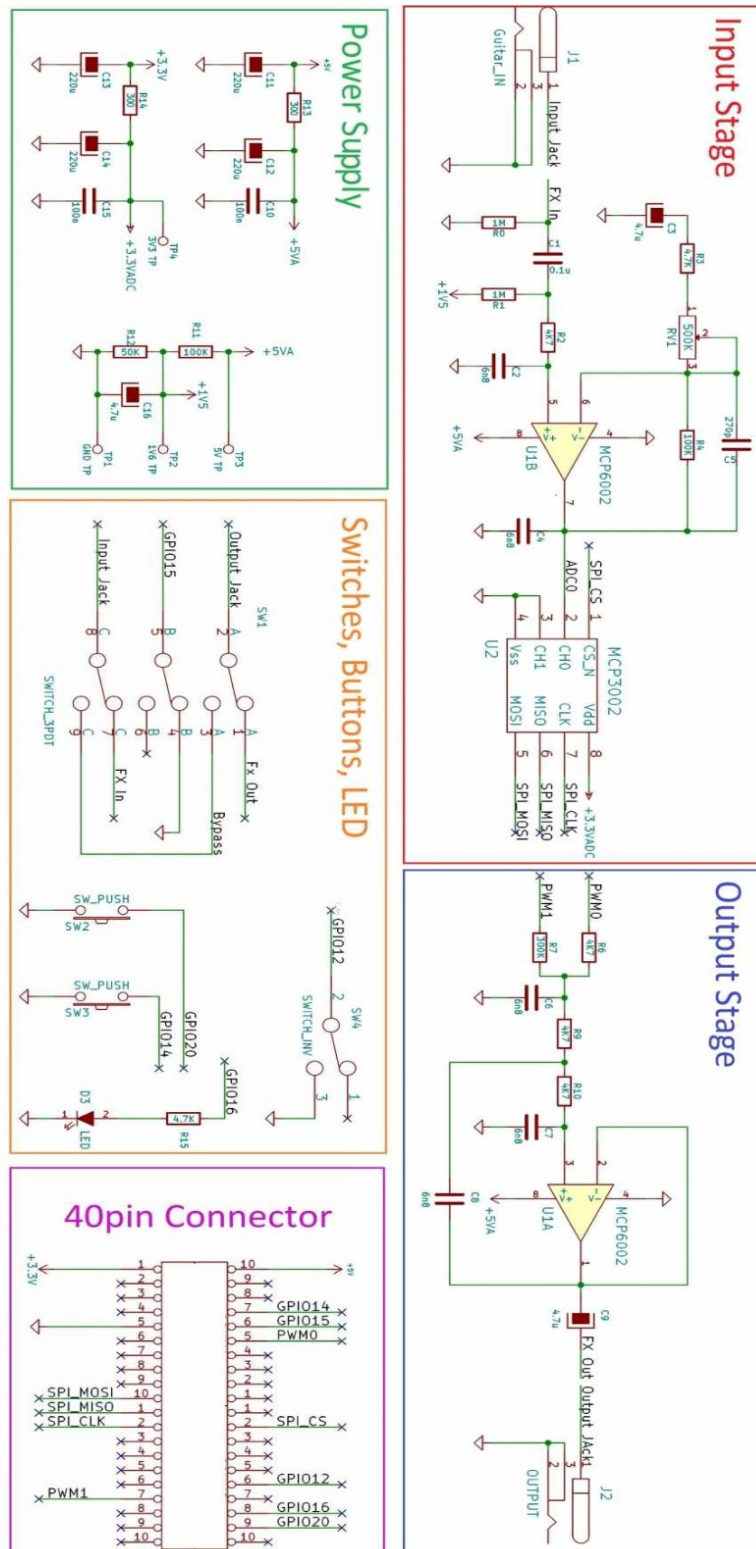
## Παράρτημα 1

Εικόνα 1. Το κύκλωμα του συστήματος αποτυπωμένο σε προσομοίωση πλακέτας με γραμμές χαλκού (Πρόγραμμα Σχεδιασμού: DIY Layout Creator)



Layout ver. 1.3  
22-09-2019

Εικόνα 2. Το σχηματικό του κυκλώματος. (πηγή: <https://www.electrosmash.com>)



## Παράρτημα 2

Εικόνα 1. Τεχνικό φυλλάδιο του A/D μετατροπέα, MCP3202



# MCP3202

## 2.7V Dual Channel 12-Bit A/D Converter with SPI Serial Interface

### Features

- 12-bit resolution
- $\pm 1$  LSB max DNL
- $\pm 1$  LSB max INL (MCP3202-B)
- $\pm 2$  LSB max INL (MCP3202-C)
- Analog inputs programmable as single-ended or pseudo-differential pairs
- On-chip sample and hold
- SPI serial interface (modes 0,0 and 1,1)
- Single supply operation: 2.7V-5.5V
- 100 ksp/s max. sampling rate at  $V_{DD} = 5V$
- 50 ksp/s max. sampling rate at  $V_{DD} = 2.7V$
- Low power CMOS technology
  - 500 nA typical standby current, 5  $\mu A$  max.
  - 550  $\mu A$  max. active current at 5V
- Industrial temp range:  $-40^{\circ}C$  to  $+85^{\circ}C$
- 8-pin MSOP, PDIP, SOIC and TSSOP packages

### Applications

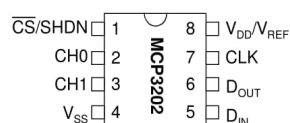
- Sensor Interface
- Process Control
- Data Acquisition
- Battery Operated Systems

### Description

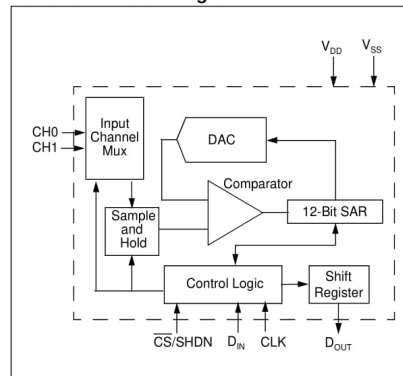
The Microchip Technology Inc. MCP3202 is a successive approximation 12-bit Analog-to-Digital (A/D) Converter with on-board sample and hold circuitry. The MCP3202 is programmable to provide a single pseudo-differential input pair or dual single-ended inputs. Differential Nonlinearity (DNL) is specified at  $\pm 1$  LSB, and Integral Nonlinearity (INL) is offered in  $\pm 1$  LSB (MCP3202-B) and  $\pm 2$  LSB (MCP3202-C) versions. Communication with the device is done using a simple serial interface compatible with the SPI protocol. The device is capable of conversion rates of up to 100 ksp/s at 5V and 50 ksp/s at 2.7V. The MCP3202 device operates over a broad voltage range (2.7V-5.5V). Low-current design permits operation with typical standby and active currents of only 500 nA and 375  $\mu A$ , respectively. The MCP3202 is offered in 8-pin MSOP, PDIP, TSSOP and 150 mil SOIC packages.

### Package Types

PDIP, MSOP, SOIC, TSSOP



### Functional Block Diagram



## Εικόνα 2. Τεχνικό φυλλάδιο του τελεστικού ενισχυτή MCP6002

# MICROCHIP MCP6001/1R/1U/2/4

## 1 MHz, Low-Power Op Amp

### Features

- Available in SC-70-5 and SOT-23-5 packages
- Gain Bandwidth Product: 1 MHz (typical)
- Rail-to-Rail Input/Output
- Supply Voltage: 1.8V to 6.0V
- Supply Current:  $I_Q = 100 \mu A$  (typical)
- Phase Margin: 90° (typical)
- Temperature Range:
  - Industrial: -40°C to +85°C
  - Extended: -40°C to +125°C
- Available in Single, Dual and Quad Packages

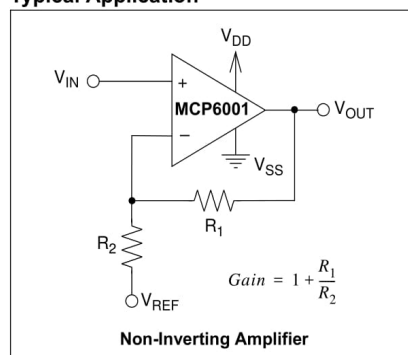
### Applications

- Automotive
- Portable Equipment
- Photodiode Amplifier
- Analog Filters
- Notebooks and PDAs
- Battery-Powered Systems

### Design Aids

- SPICE Macro Models
- FilterLab® Software
- Mindi™ Circuit Designer & Simulator
- Microchip Advanced Part Selector (MAPS)
- Analog Demonstration and Evaluation Boards
- Application Notes

### Typical Application

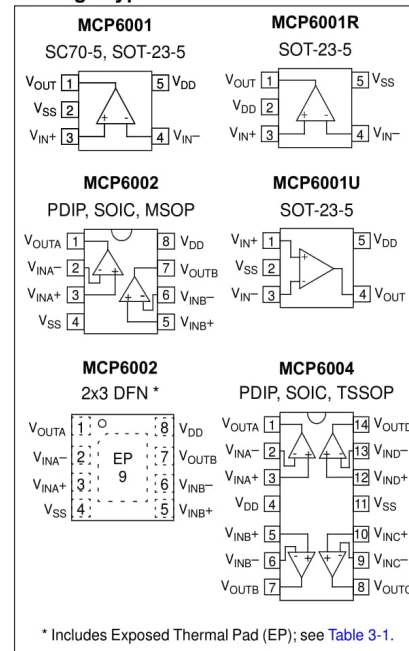


### Description

The Microchip Technology Inc. MCP6001/2/4 family of operational amplifiers (op amps) is specifically designed for general-purpose applications. This family has a 1 MHz Gain Bandwidth Product (GBWP) and 90° phase margin (typical). It also maintains 45° phase margin (typical) with a 500 pF capacitive load. This family operates from a single supply voltage as low as 1.8V, while drawing 100  $\mu A$  (typical) quiescent current. Additionally, the MCP6001/2/4 supports rail-to-rail input and output swing, with a common mode input voltage range of  $V_{DD} + 300 \text{ mV}$  to  $V_{SS} - 300 \text{ mV}$ . This family of op amps is designed with Microchip's advanced CMOS process.

The MCP6001/2/4 family is available in the industrial and extended temperature ranges, with a power supply range of 1.8V to 6.0V.

### Package Types



## Παράρτημα 3

**Πίνακας 1. Τα βήματα που απαιτούνται για την πρώτη εκκίνηση του Raspberry Pi Zero W**

- Εγκατάσταση του λειτουργικού συστήματος (Rasbian Buster Lite) στην κάρτα microSD, μέσω της χρήσης του ελεύθερου προγράμματος ανοικτού κώδικα, balenaEtcher.
- Δημιουργία αρχείου με ονομασία “wpa\_supplicant.conf” και κενού αρχείου με ονομασία “ssh”. Η δημιουργία των αρχείων είναι απαραίτητη για τη σύνδεση του Raspberry Pi Zero W στο Wi-Fi δίκτυο κατά το μοντέλο headless. Το αρχείο “wpa\_supplicant.conf” θα πρέπει να περιέχει το παρακάτω κείμενο:

```
country=GR

ctrl_interface=DIR=/var/run/wpa_supplicant GROUP=netdev
update_config=1

network={
    ssid="MyWiFiNetwork"
    psk="MyPassword"
    key_mgmt=WPA-PSK
}
```

- Τοποθέτηση της κάρτας microSD στην κατάλληλη υποδοχή και τροφοδότηση του Raspberry Pi με τάση 5V.
- Εκκίνηση του προγράμματος PuTTY και σύνδεση με το Raspberry Pi μέσω του πρωτοκόλλου SSH.
- Εισαγωγή του ονόματος χρήστη (username) και του κωδικού πρόσβασης (password), που είναι προκαθορισμένα στα Raspberry Pi σαν “pi” και “raspberrypi” αντίστοιχα.
- Ενημέρωση του λειτουργικού συστήματος μέσω των εντολών “sudo apt -get update” και “sudo apt -get upgrade”.
- Αλλαγή του κωδικού πρόσβασης μέσω του μενού διαχείρισης συστήματος “sudo raspi-config”

## Πίνακας 2. Περιγραφή της λειτουργίας βασικών εντολών του λειτουργικού συστήματος Linux

- **pwd** – (print working directory). Με αυτή την εντολή ο υπολογιστής εμφανίζει στην γραμμή εντολών το φάκελο στον οποίο βρίσκεται ο/η χρήστης, τη δεδομένη χρονική στιγμή (πχ. *pi/home/bcm2835/examples/*)
- **man** – (manual). Μια από τις σημαντικότερες εντολές του Linux. Είναι ουσιαστικά το εγχειρίδιο χρήσης κάθε εντολής, όπου εμπεριέχονται όλες τις χρήσιμες πληροφορίες και παράμετροι της εντολής που χρειάζεται να γνωρίζει ο/η χρήστης.
- **ls** – (list files). Η εντολή αυτή εμφανίζει όλα τα αρχεία που υπάρχουν στο φάκελο που βρίσκεται ο/η χρήστης τη δεδομένη χρονική στιγμή. Μπορεί να δεχθεί επιπλέον επιλογές, δίπλα από τη βασική εντολή, όπως η επιλογή **-l** που εμφανίζει τα μεγέθη των αρχείων, την ημερομηνία και ώρα τροποποίησης τους και τις άδειες που σχετίζονται με τη χρήση τους.
- **cd** – (change directory). Η συγκεκριμένη εντολή επιτρέπει την περιήγηση του/της χρήστη στους φακέλους αρχείων του υπολογιστή. Η μετάβαση σε έναν συγκεκριμένο φάκελο του συστήματος γίνεται με τη χρήση του ονόματος του φακέλου ακριβώς μετά την εντολή (πχ. *cd examples*). Με τη χρήση της καθέτου, ο/η χρήστης μπορεί να μεταβεί κατευθείαν σε έναν υποφάκελο εντός του κυρίου φακέλου (πχ. *cd example/audio*).
- **rm** – (remove). Η εντολή “rm” χρησιμοποιείται για την οριστική διαγραφή ενός αρχείου.
- **mkdir & rmdir** – (make & remove directory). Οι εντολές αυτές επιτρέπουν την δημιουργία και τη διαγραφή ενός φακέλου, αντίστοιχα.
- **mv & cp** – (move & copy). Η εντολή “mv” δεν μετακινεί ακριβώς ένα αρχείο, αλλά το μετονομάζει στη θέση που ο/η χρήστης επιθυμεί, διαγράφοντας ταυτόχρονα το παλιό αρχείο. Από την άλλη, η εντολή “cp” απλά αντιγράφει ένα αρχείο στην επιθυμητή θέση.
- **Exit**. Με τη χρήση της συγκεκριμένης εντολής διακόπτονται οι εν ενεργεία εργασίες και κλείνει η γραμμή εντολών.

- **Sudo** – (Super User **do**). Η εντολή “Sudo” δίνει την δυνατότητα στον/στην χρήστη να εκτελέσει μια εντολή με δικαιώματα υπερχρήστη. Πρόκειται για μια πολύ ισχυρή εντολή

που πρέπει να χρησιμοποιείται με προσοχή διότι οι συνέπειες της χρήσης της δεν γίνεται να ανακληθούν.

- **Kill**. Πρόκειται για εντολή άμεσης παύσης λειτουργίας προγραμμάτων που έχουν “κολλήσει” λόγω περιορισμένων πόρων.
- **Wget**. Εντολή που χρησιμοποιείται για την λήψη αρχείων από το Internet. Απαραίτητη προϋπόθεση είναι να υπάρχει ενεργοποιημένο δίκτυο Wi-fi στο Raspberry Pi.
- **Sudo apt-get** (update, upgrade, install, remove, purge). Εντολές που χρησιμοποιούνται για την ενημέρωση, αναβάθμιση, εγκατάσταση και διαγραφή προγραμμάτων.
- **Shutdown -h now & reboot**. Χρησιμοποιούνται για την άμεση απενεργοποίηση και επανεκκίνηση του συστήματος, αντίστοιχα.



## Παράρτημα 4

**Πίνακας 1. Ο κώδικας που χρειάζεται για την λήψη και εγκατάσταση της βιβλιοθήκης BCM2835**

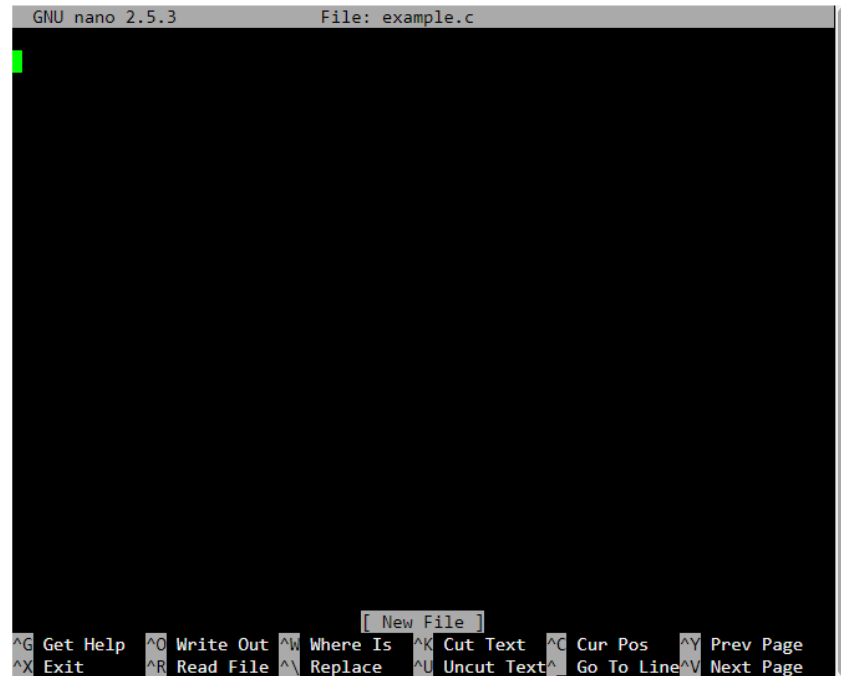
```
wget http://www.airspayce.com/mikem/bcm2835/bcm2835-1.xx.tar.gz;  
tar zxvf bcm2835-1.xx.tar.gz  
cd bcm2835-1.xx  
./configure  
make  
sudo make check  
sudo make install
```

Για την λήψη της τελευταίας έκδοσης των βιβλιοθηκών BCM2835 (1.60 - 22/07/2019, η έκδοση που χρησιμοποιείται από το Raspberry Pi του συστήματος), χρειάζεται η εισαγωγή του “1.60” στον παραπάνω κώδικα στην θέση του “1.xx”.

**Πίνακας 2. Ο κώδικας που χρειάζεται για τη λήψη και εγκατάσταση της βιβλιοθήκης Pigpio**

```
wget abyz.me.uk/rpi/pigpio/pigpio.tar  
  
rm pigpio.tar  
sudo rm -rf PIGPIO  
tar xf pigpio.tar  
cd PIGPIO  
make  
sudo make install
```

**Εικόνα 1. Το γραφικό περιβάλλον του Nano κατά τη δημιουργία ενός αρχείου με όνομα “example.c”**



*Εικόνα 13: Δημιουργία ενός αρχείου με όνομα "example" μέσω του προγράμματος Nano.*

**Πίνακας 3. Οι εντολές που χρειάζονται για τη δημιουργία, σύνταξη και εκτέλεση ενός αρχείου με όνομα “example.c”**

Λειτουργία	Εντολή
Δημιουργία αρχείου με όνομα “example.c”	sudo nano example.c
Σύνταξη του αρχείου “example.c” σε γλώσσα μηχανής	gcc -Wall -pthread -o example example.c - lpigpio -thread -lbcm2835 -lrt
Εκτέλεση του αρχείου “example.c”	sudo ./example

#### Πίνακας 4. Περιγραφή των εντολών σύνδεσης του Raspberry Pi με το Bluetooth ηχείο

- Bluetoothctl - Ξεκινάει την διαδικασία της σύνδεσης Bluetooth
- Power on – Παρέχεται ρεύμα τροφοδοσίας στο τσιπ που είναι υπεύθυνο για το Bluetooth
- Agent on - Ελέγχεται αν ο agent που είναι υπεύθυνος για τη σύνδεση είναι σε λειτουργία
- Scan on – Η συσκευή αρχίζει την ανίχνευση για την εύρεση συσκευών Bluetooth
- Scan off – Εφόσον βρεθεί διαθέσιμη συσκευή και σημειωθεί η IP της, σταματάει η ανίχνευση
- Pair “device IP” - Επιχειρείται δημιουργία σύνδεσης με την διαθέσιμη συσκευή
- Paired-devices – Εφόσον έγινε η δημιουργία σύνδεσης, η συσκευή με τη συγκεκριμένη IP θα φαίνεται στη λίστα με τις συνδεδεμένες συσκευές
- Connect “device IP” - Οριστική σύνδεση με την συσκευή slave
- Trust “device IP” - Με την εντολή “trust” αποθηκεύεται η συσκευή slave με την συγκεκριμένη IP, ώστε οι μελλοντικές συνδέσεις μαζί της να είναι αυτοματοποιημένες.
- Quit – Τερματισμός της διαδικασίας σύνδεσης

#### Πίνακας 5. Η διαδικασία λήψης και εγκατάστασης του PulseAudio και του PulseAudio Volume Control

Η λήψη και εγκατάσταση του Pulse Audio στο Raspberry Pi, γίνεται με την εντολή:

- `sudo apt-get install pulseaudio`

Επίσης, η λήψη και εγκατάσταση του PulseAudio Volume Control, που επιτρέπει τον έλεγχο της έντασης του ήχου γίνεται με την εντολή:

- `sudo apt-get install pavucontrol paprefs`

Μετά την εγκατάσταση χρειάζεται επανεκκίνηση του Raspberry Pi.

## Παράρτημα 5

Πίνακας 1. Εισαγωγή βιβλιοθηκών και καθορισμός των υποδοχών GPIO

```
// Include Libraries
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <pigpio.h>
#include <inttypes.h>
#include <fcntl.h>
#include <sys/stat.h> /* For O_* constants */
#include <semaphore.h> /* For mode constants */
#include <signal.h>
#include <bcm2835.h>
// Define Input Pins
#define PUSH1 RPI_GPIO_P1_08 //GPIO14
#define PUSH2 RPI_V2_GPIO_P1_38 //GPIO20
#define TOGGLE_SWITCH RPI_V2_GPIO_P1_32 //GPIO12
#define FOOT_SWITCH RPI_GPIO_P1_10 //GPIO15
#define LED RPI_V2_GPIO_P1_36 //GPIO16
#define SPI_SS 8 // GPIO for slave select.
#define ADCS 1 // Number of connected MCP3202.
#define BITS 12 // Bits per reading.
#define BX 6 // Bit position of data bit B11.
#define B0 (BX + BITS - 1) // Bit position of data bit B0.
#define MISO1 9 // ADC 1 MISO.
//Define GPIO pins configuration
bcm2835_gpio_fsel(PUSH1, BCM2835_GPIO_FSEL_INPT); //PUSH1 button as input
bcm2835_gpio_fsel(PUSH2, BCM2835_GPIO_FSEL_INPT); //PUSH2 button as input
bcm2835_gpio_fsel(TOGGLE_SWITCH, BCM2835_GPIO_FSEL_INPT);
//TOGGLE_SWITCH as input
bcm2835_gpio_fsel(FOOT_SWITCH, BCM2835_GPIO_FSEL_INPT); //FOOT_SWITCH as
input
bcm2835_gpio_fsel(LED, BCM2835_GPIO_FSEL_OUTP); //LED as output
bcm2835_gpio_set_pud(PUSH1, BCM2835_GPIO_PUD_UP); //PUSH1 pull-up enabled
bcm2835_gpio_set_pud(PUSH2, BCM2835_GPIO_PUD_UP); //PUSH2 pull-up enabled
bcm2835_gpio_set_pud(TOGGLE_SWITCH, BCM2835_GPIO_PUD_UP);
//TOGGLE_SWITCH pull-up enabled
bcm2835_gpio_set_pud(FOOT_SWITCH, BCM2835_GPIO_PUD_UP);
//define PWM mode
bcm2835_gpio_fsel(18, BCM2835_GPIO_FSEL_ALT5); //PWM0 signal on GPIO18
bcm2835_gpio_fsel(13, BCM2835_GPIO_FSEL_ALT0); //PWM1 signal on GPIO13
bcm2835_pwm_set_clock(2); // Max clk frequency (19.2MHz/2 = 9.6MHz)
bcm2835_pwm_set_mode(0, 1, 1); //channel 0, markspace mode, PWM enabled.
bcm2835_pwm_set_range(0, 64); //channel 0, 64 is max range (6bits):
9.6MHz/64=150KHz PWM freq.
bcm2835_pwm_set_mode(1, 1, 1); //channel 1, markspace mode, PWM enabled.
bcm2835_pwm_set_range(1, 64);
```

**Πίνακας 2. Ο αλγόριθμος του εφέ παραμόρφωσης**

```
//add distortion effect if TOGGLE_SWITCH is pressed
if (!TOGGLE_SWITCH_val) {
    if (output_signal > 863 + distortion_value)
        output_signal = 863 + distortion_value;
    if (output_signal < 863 - distortion_value)
        output_signal = 863 - distortion_value;
    output_signal = output_signal << 3;
}
```

**Πίνακας 3. Ενεργοποίηση του εφέ και έλεγχος των παραμέτρων του**

```
if (sample % 50000 == 0) {
    uint8_t PUSH1_val = bcm2835_gpio_lev(PUSH1);
    uint8_t PUSH2_val = bcm2835_gpio_lev(PUSH2);
    TOGGLE_SWITCH_val = bcm2835_gpio_lev(TOGGLE_SWITCH);
    uint8_t FOOT_SWITCH_val = bcm2835_gpio_lev(FOOT_SWITCH);
    //light the effect when the footswitch is activated.
    bcm2835_gpio_write(LED, !FOOT_SWITCH_val);

    //update booster_value when the PUSH1 or 2 buttons are pushed.
    if (PUSH1_val == 0) //less distortion
    {
        bcm2835_delay(100); //100ms delay for buttons debouncing
        if (distortion_value < 2047)
            distortion_value = distortion_value + 10;
    } else if (PUSH2_val == 0) //more distortion
    {
        bcm2835_delay(100); //100ms delay for buttons debouncing.
        if (distortion_value > 0)
            distortion_value = distortion_value - 10;
    }
}
```

#### Πίνακας 4. Βασικές λειτουργίες του προγράμματος 2

```
#ifndef HAVE_CONFIG_H
#include <config.h>
#endif
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include <errno.h>
#include <fcntl.h>
#include <pulse/simple.h>
#include <pulse/error.h>

// The Sample format to use 12480
const pa_sample_spec ss = { .format = PA_SAMPLE_S16LE, .rate = 12480,
    .channels = 1 };
pa_simple *s = NULL;
int ret = 1;
int error;

// Create a new playback stream
if (!(s = pa_simple_new(NULL, argv[0], PA_STREAM_PLAYBACK, NULL, "playback",
    &ss, NULL, NULL, &error))) {
    fprintf(stderr, __FILE__: pa_simple_new() failed: %s\n",
        pa_strerror(error));
    goto finish;
}

// Read some data ...
if ((r = read(STDIN_FILENO, buf, sizeof(buf))) <= 0) {
    if (r == 0) /* EOF */
        break;
    fprintf(stderr, __FILE__: read() failed: %s\n", strerror(errno));
    goto finish;
}

// ... and play it
if (pa_simple_write(s, buf, ((size_t) r), &error) < 0) {
    fprintf(stderr, __FILE__: pa_simple_write() failed: %s\n",
        pa_strerror(error));
    goto finish;
}
```

### Πίνακας 5. Ο κώδικας του αρχείου “rc.local”

```
pulseaudio -D -vvvv
```

```
sudo bluetoothctl connect F8:DF:15:DA:36:01
```

```
sudo ./home/pi/rawMCP3202 | .home/pi/pulseaudio_test
```