



ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

«Διαδικτυακή Εφαρμογή για την Ανάλυση Λειτουργίας
Βασικών Ηλεκτρονικών Διατάξεων με χρήση
Γραφικών»

Του φοιτητή
Χαραλαμπίδη Άγγελου
Αρ. Μητρώου: 144179

Επιβλέπων
Παπακώστας Δημήτριος
Καθηγητής

Ημερομηνία 27-09-2022

Τίτλος Δ.Ε: Διαδικτυακή Εφαρμογή για την Ανάλυση Λειτουργίας Βασικών Ηλεκτρονικών

Διατάξεων με χρήση Γραφικών

Κωδικός Δ.Ε: 21171

Όνοματεπώνυμο φοιτητή: Χαραλαμπίδης Άγγελος

Όνοματεπώνυμο εισηγητή: Παπακώστας Δημήτριος

Ημερομηνία ανάληψης Δ.Ε: 02-04-2021

Ημερομηνία περάτωσης Δ.Ε: 27-09-2022

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

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

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

«Σε όλους τους νέους που οραματίζονται ένα καλύτερο μέλλον»

Πρόλογος

Η παρούσα διπλωματική εργασία έχει ως αντικείμενο τη δημιουργία μιας εύχρηστης και φιλικής προς το χρήστη εφαρμογής που παρουσιάζει με διαδραστικό τρόπο τη λειτουργία των βασικών ηλεκτρονικών κυκλωμάτων και διατάξεων που διδάσκονται στα μαθήματα «Ηλεκτρονικά Κυκλώματα» και «Ηλεκτρονικές Διατάξεις» του προγράμματος σπουδών του τμήματος.

Έχοντας πρόσφατα περάσει από τη διαδικασία παρακολούθησης και εξέτασης των παραπάνω μαθημάτων, θεώρησα ότι η δημιουργία του εν λόγω εργαλείου, θα αυξήσει σημαντικά την πιθανότητα για ενασχόληση και κατά συνέπεια επιτυχία των διδασκόμενων στα προαναφερθέντα μαθήματα. Σκοπός της είναι να συμβάλλει στην καλύτερη κατανόηση και λειτουργία των επιλεγμένων κυκλωμάτων και στην εξαγωγή συμπερασμάτων που μπορεί να φανούν χρήσιμα για την καλύτερη αφομοίωση των εννοιών της υπό μελέτης ύλης.

Το όφελος είναι διττό καθώς, βοήθησε εμένα προσωπικά να καλλιεργήσω και να αναπτύξω έναν σύνθετο τρόπο σκέψης, κατά τον οποίο έπρεπε να συνδυάσω γνώριμες μαζί με καινούριες τεχνολογίες, αλλά, όπως φαίνεται κι από τα αποτελέσματα του ερωτηματολογίου που συντάχθηκε με στόχο την αξιολόγηση της εφαρμογής, τείνει να αποτελέσει σημαντικό βοήθημα στην προσπάθεια των φοιτητών για μάθηση.

Περίληψη

Σε μια συνεχώς και ραγδαία εξελισσόμενη τεχνολογικά κοινωνία, η ανάγκη για τη δημιουργία νέων εργαλείων που μπορούν να βοηθήσουν στην καλύτερη κατανόηση και εκμάθηση των γνωστικών αντικειμένων των φοιτητών γίνεται ολοένα και μεγαλύτερη.

Στα πλαίσια αυτής της εργασίας δημιουργήθηκε μια εφαρμογή η οποία έχει ως στόχο να βοηθήσει τους φοιτητές που παρακολουθούν τα μαθήματα «Ηλεκτρονικά Κυκλώματα» και «Ηλεκτρονικές Διατάξεις» του προγράμματος σπουδών του τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων, αλλά και το ευρύτερο κοινό που ενδιαφέρεται για την καλύτερη αφομοίωση εννοιών όπως: περιοχή λειτουργίας, γραμμή φορτίου, ρυθμός μεταβολής κλπ εφαρμοσμένες πάνω στα κυκλώματα που διδάσκονται στα προαναφερθέντα μαθήματα. Η υλοποίηση έγινε με τη χρήση του React JS framework της JavaScript στο προγραμματιστικό περιβάλλον του Visual Studio Code.

Για να γίνει ακόμα πιο ενδιαφέρουσα η χρήση, κυρίως για τους φοιτητές, η εφαρμογή διαχωρίζει σε θεματικές ενότητες το περιεχόμενό της, μία για τη χρήση των διπολικών τρανζίστορ, που αποτελεί αντικείμενο που πραγματεύεται το μάθημα «Ηλεκτρονικά Κυκλώματα» και μία για τους Τελεστικούς ενισχυτές, που αφορά τις «Ηλεκτρονικές Διατάξεις». Πιο συγκεκριμένα, παρουσιάζονται τα κυκλώματα της πόλωσης βάσης, πόλωσης εκπομπού και πόλωσης διαιρέτη τάσης (με το ισοδύναμό του κύκλωμα) για τα διπολικά τρανζίστορ και αυτά του αναστρέφοντος, μη αναστρέφοντος, αθροιστικού και ενισχυτή διαφορών για τους τελεστικούς ενισχυτές.

Τα αποτελέσματα που προέκυψαν από τη χρήση της κρίνονται ιδιαίτερα ενθαρρυντικά και ενδιαφέροντα καθώς, ύστερα από μερικές απαντήσεις σε ερωτηματολόγιο από άτομα που τη δοκίμασαν, διακρίνεται η ικανοποίησή τους για την υλοποίηση του εν λόγω εργαλείου.

«Web Application for the Analysis of the function of Basic Electronic Devices using Graphics»

«Charalampidis Angelos»

Abstract

In an ever-evolving technological society, the need to create new tools, that can help better understanding and learning of students' cognitive subjects, is growing.

As part of this work, an application was created which aims to help students attending the courses “Electronic Circuits” and “Electronic Devices” of the curriculum of the Department of Computer Engineering and Electronic Systems, but also the general public interested in better assimilation of concepts such as: Operating area, Load line, Slew rate, etc. applied to the circuits taught in the above courses.

To make the use even more interesting, especially for students, the application divides its content into thematic sections, one for the use of bipolar transistors, which is the subject of the course “Electronic Circuits” and one for the Operational Amplifiers, concerning the “Electronical Devices”. More specifically, the circuits of base polarization, emitter polarization and voltage divider polarization (with equivalent circuit) for bipolar transistors and those of inverting, non-inverting, summing and difference amplifier for operational amplifiers are presented.

The results obtained from its use are considered notably encouraging and interesting as, after some responses to a questionnaire from people who tested it, their satisfaction for the implementation of the specific tool can be seen.

Ευχαριστίες

Θα ήθελα να ευχαριστήσω πρώτα απ' όλους την οικογένειά μου που ήταν δίπλα μου σε κάθε βήμα της φοιτητικής μου ζωής καθώς και τον επιβλέπον καθηγητή της εργασίας κ. Παπακώστα, με του οποίου την καθοδήγηση πετύχαμε το επιθυμητό αποτέλεσμα.

Περιεχόμενα

Πρόλογος.....	v
Περίληψη	vi
Abstract	vii
Ευχαριστίες	viii
Περιεχόμενα	ix
Κατάλογος Εικόνων.....	xii
Συντομογραφίες.....	xvi
Κεφάλαιο 1ο: Εισαγωγή.....	1
1.1 Περιγραφή της ιδέας.....	1
1.2 Οργάνωση ενοτήτων.....	1
1.3 Στόχοι	2
1.4 Δομή και χρησιμότητα των virtual labs.....	2
1.5 Επίλογος.....	6
Κεφάλαιο 2ο: Κυκλώματα που χρησιμοποιήθηκαν	7
2.1 Εισαγωγή	7
2.2 Διπολικά τρανζίστορ	7
2.2.1 Ιστορικά στοιχεία και ορισμοί.....	7
2.2.2 Λειτουργία ηρη τρανζίστορ.....	8
2.2.2.1 Ηλεκτρόνια εκπομπού.....	9
2.2.2.2 Ηλεκτρόνια βάσης	10
2.2.2.3 Ηλεκτρόνια συλλέκτη	10
2.2.3 Ρεύματα και σχέσεις μεταξύ τους.....	10
2.2.4 Περιοχές λειτουργίας	12
2.3 Τελεστικοί ενισχυτές	13
2.3.1 Ιστορικά στοιχεία.....	13
2.3.2 Δομή TE.....	14
2.3.3 Κορεσμός εξόδου (MPP)	15
2.3.4 Ρυθμός μεταβολής (slew rate)	17
2.3.5 Βασικές συνδεσμολογίες TE	18
2.3.5.1 Αναστρέφων ενισχυτής.....	18
2.3.5.1.1 Ορισμός και κέρδος τάσης.....	18
2.3.5.1.2 Εύρος ζώνης.....	20

2.3.5.2	Μη - Αναστρέφων ενισχυτής.....	20
2.3.5.2.1	Ορισμός και κέρδος τάσης.....	20
2.3.5.2.2	Εύρος ζώνης.....	22
2.3.5.3	Αθροιστής.....	22
2.3.5.4	Ενισχυτής διαφορών	23
2.4	Επίλογος.....	24
Κεφάλαιο 3ο: Ανάλυση λειτουργίας επιλεγμένων κυκλωμάτων.....		25
3.1	Εισαγωγή	25
3.2	Διπολικά τρανζίστορ	25
3.2.1	Κύκλωμα πόλωσης βάσης.....	25
3.2.2	Κύκλωμα πόλωσης εκπομπού	27
3.2.3	Κύκλωμα πόλωσης διαιρέτη τάσης	29
3.2.3.1	Ισοδύναμο κύκλωμα με R_{TH}	31
3.2.3.2	Γραμμή φορτίου και σημείο λειτουργίας Q	32
3.3	Τελεστικοί ενισχυτές	35
3.3.1	Κύκλωμα αναστρέφοντος ενισχυτή	35
3.3.2	Κύκλωμα μη αναστρέφοντος ενισχυτή.....	37
3.3.3	Κυκλώματα αθροιστή και ενισχυτή διαφορών.....	39
3.4	Επίλογος.....	39
Κεφάλαιο 4ο: Περιβάλλον και τεχνολογίες που χρησιμοποιήθηκαν		41
4.1	Εισαγωγή	41
4.2	Το προγραμματιστικό περιβάλλον Visual Studio Code	41
4.2.1	Ιστορία	41
4.2.2	Χρήση και λειτουργία.....	41
4.3	Το προγραμματιστικό πλαίσιο ReactJS.....	43
4.3.1	Ιστορία	43
4.3.2	Χρήση και λειτουργία.....	44
4.4	Επίλογος.....	45
Κεφάλαιο 5ο: Ανάλυση της εφαρμογής.....		47
5.1	Εισαγωγή	47
5.2	Διαγραμματική απεικόνιση της εφαρμογής.....	47
5.2.1	Γενική δομή και επιμέρους κυκλώματα.....	47
5.2.2	Επιμέρους λειτουργίες	51
5.3	Ανάλυση της δομής του κώδικα.....	57
5.4	Ανάλυση του κώδικα και των λειτουργιών.....	58

5.4.1	Το αρχείο App.js.....	59
5.4.2	Ο φάκελος main.....	61
5.4.2.1	Το αρχείο main.jsx.....	61
5.4.2.2	Το αρχείο componentfromarray.jsx	62
5.4.3	Ο φάκελος με τα κυκλώματα (circuits).....	63
5.4.3.1	Το κύκλωμα πόλωσης διαιρέτη τάσης (VDB).....	63
5.4.3.2	Τα κυκλώματα πόλωσης βάσης και εκπομπού (Bjt και Emmiter).....	74
5.4.3.3	Το κύκλωμα του αναστρέφοντος ενισχυτή (Inverting Amplifier).....	82
5.4.3.4	Τα κυκλώματα μη-αναστρέφοντος, αθροιστικού και ενισχυτή διαφορών (Non-Inverting, Summing, Difference Amplifier)	88
5.4.4	Ο φάκελος με τα συστατικά (components)	91
5.4.4.1	Το συστατικό CircuitValueInput	91
5.4.4.2	Το συστατικό VccValueSelector	96
5.4.4.3	Το συστατικό SineWave	97
5.4.4.4	Το συστατικό Pulse.....	97
5.5	Επίλογος.....	98
Κεφάλαιο 6ο:	Αξιολόγηση λειτουργίας εφαρμογής.....	99
6.1	Εισαγωγή	99
6.2	Ανάλυση αποτελεσμάτων	99
6.3	Γενική αποτίμηση αποτελεσμάτων	104
Κεφάλαιο 7ο:	Συμπεράσματα και προτάσεις βελτίωσης.....	107
7.1	Συμπεράσματα	107
7.2	Προτάσεις βελτίωσης	107
BIBΛΙΟΓΡΑΦΙΑ.....		109
ΠΑΡΑΡΤΗΜΑ Α: ΚΩΔΙΚΑΣ ΕΦΑΡΜΟΓΗΣ.....		111
ΠΑΡΑΡΤΗΜΑ Β: ΕΡΩΤΗΜΑΤΟΛΟΓΙΟ		123

Κατάλογος Εικόνων

Εικόνα 1.1: Λογική ενός virtual lab.....	3
Εικόνα 1.2: Δομή ενός virtual lab.....	4
Εικόνα 1.3: Η λίστα με τα πειράματα.....	5
Εικόνα 1.4: Web-based πλατφόρμα του προσομοιωτή DoCircuits.....	6
Εικόνα 1.5: Οπτικοποίηση των αποτελεσμάτων με τη χρήση του εργαλείου DoCircuits.....	6
Εικόνα 2.1: Το πρώτο πειραματικό τρανζίστορ.....	7
Εικόνα 2.2: Δομή του τρανζίστορ.....	8
Εικόνα 2.3: Περιοχές απογύμνωσης.....	9
Εικόνα 2.4: Πολωμένο τρανζίστορ.....	9
Εικόνα 2.5: Ελεύθερα ηλεκτρόνια έχουν εισαχθεί μέσω της διόδου εκπομπού στη βάση.....	10
Εικόνα 2.6: Σύμβολο npn τύπου τρανζίστορ.....	10
Εικόνα 2.7: Σχηματικό διάγραμμα του 741.....	14
Εικόνα 2.8: Εσωτερικό διάγραμμα TE.....	14
Εικόνα 2.9: Σχηματικό σύμβολο ενός TE.....	15
Εικόνα 2.10: Γραφική παράσταση για την MPP.....	16
Εικόνα 2.11: Ιδανική και πραγματική έξοδος TE σε διάγραμμα $V - t$	16
Εικόνα 2.12: Πρόβλημα λόγω ρυθμού μεταβολής.....	17
Εικόνα 2.13: Ρυθμός μεταβολής τάσης εξόδου.....	17
Εικόνα 2.14: Αναστρέφων ενισχυτής.....	18
Εικόνα 2.15: Ρεύμα στις αντιστάσεις R_1 και R_f	19
Εικόνα 2.16: Όσο μεγαλώνει η συχνότητα μικραίνει το κέρδος.....	20
Εικόνα 2.17: Μη – αναστρέφων ενισχυτής.....	21
Εικόνα 2.18: Ισοδύναμο κύκλωμα μη – αναστρέφοντος ενισχυτή.....	21
Εικόνα 2.19: Αθροιστικός ενισχυτής.....	22
Εικόνα 2.20: Ενισχυτής διαφορών.....	23
Εικόνα 3.1: Κύκλωμα πόλωσης βάσης.....	25
Εικόνα 3.2: Κύκλωμα πόλωσης εκπομπού.....	27
Εικόνα 3.3: Κύκλωμα πόλωσης διαιρέτη τάσης.....	29
Εικόνα 3.4: Απλοποιημένο κύκλωμα.....	30
Εικόνα 3.5: Ισοδύναμο κύκλωμα με R_{TH}	31
Εικόνα 3.6: Γραμμή φορτίου.....	33
Εικόνα 3.7: Σημείο λειτουργίας Q.....	34
Εικόνα 3.8: Ρεύμα κορεσμού $I_{C(SAT)}$	34
Εικόνα 3.9: Τάση αποκοπής $V_{CE(CUTOFF)}$	34
Εικόνα 3.10: Μετατόπιση σημείου Q προς τον κορεσμό.....	35
Εικόνα 3.11: Μετατόπιση σημείου Q προς την αποκοπή.....	35
Εικόνα 3.12: Η είσοδος και η έξοδος σε ημιτονοειδή μορφή.....	36
Εικόνα 3.13: Η είσοδος και η έξοδος σε μορφή τετραγωνικών παλμών.....	37
Εικόνα 3.14: Τα σήματα εισόδου και εξόδου του μη αναστρέφοντος ενισχυτή σε ημιτονοειδή μορφή.....	38
Εικόνα 3.15: Τα σήματα εισόδου και εξόδου του μη αναστρέφοντος ενισχυτή σε μορφή τετραγωνικών παλμών.....	38
Εικόνα 4.1: Το VS Code Marketplace.....	42

Εικόνα 4.2: Χρήση του object props.....	44
Εικόνα 4.3: Χρήση του return	45
Εικόνα 5.1: Γενική δομή εφαρμογής	47
Εικόνα 5.2: Διάγραμμα ροής για το κύκλωμα πόλωσης διαιρέτη τάσης	48
Εικόνα 5.3: Διάγραμμα ροής για τα κυκλώματα πόλωσης βάσης και πόλωσης εκπομπού	49
Εικόνα 5.4: Διάγραμμα ροής για τα κυκλώματα αναστρέφοντος και μη αναστρέφοντος ενισχυτών ...	50
Εικόνα 5.5: Διάγραμμα ροής για τα κυκλώματα αθροιστικού και ενισχυτή διαφορών.....	51
Εικόνα 5.6: Διάγραμμα ροής ελέγχου τιμών.....	52
Εικόνα 5.7: Διάγραμμα ροής μετατροπής μονάδων μέτρησης.....	53
Εικόνα 5.8: Διάγραμμα ροής σχεδίασης γραμμής φορτίου.....	54
Εικόνα 5.9: Διάγραμμα ροής σχεδίασης ημιτονοειδών κυματομορφών και τετραγωνικών παλμών	55
Εικόνα 5.10: Διάγραμμα ροής εμφάνισης περιοχής λειτουργίας	56
Εικόνα 5.11: Ο φάκελος src και τα περιεχόμενά του.....	57
Εικόνα 5.12: Ο φάκελος components και τα περιεχόμενά του.....	57
Εικόνα 5.13: Ο φάκελος main και τα περιεχόμενά του	57
Εικόνα 5.14: Ο φάκελος circuits και τα περιεχόμενα των υποφακέλων του.....	58
Εικόνα 5.15: Η έκδοση για το node.js	59
Εικόνα 5.16: Η έκδοση για το npm.....	59
Εικόνα 5.17: Διάφορα dependencies που έχουν εγκατασταθεί	59
Εικόνα 5.18: Η εικόνα που φαίνεται με το άνοιγμα της εφαρμογής.....	60
Εικόνα 5.19: Ο ορισμός και η αρχικοποίηση των μεταβλητών.....	60
Εικόνα 5.20: Το useEffect hook	61
Εικόνα 5.21: Το σημείο που περνάει παραμετρικά το index ως key	61
Εικόνα 5.22: Εμφάνιση των διπολικών τρανζίστορ	62
Εικόνα 5.23: Εμφάνιση των τελεστικών ενισχυτών	62
Εικόνα 5.24: Το κύκλωμα πόλωσης διαιρέτη τάσης	63
Εικόνα 5.25: Αρχικές τιμές για τα πεδία που χρειάζονται εκχώρηση τιμής	64
Εικόνα 5.26: Η useEffect	65
Εικόνα 5.27: Το μήνυμα σε περίπτωση εκχώρησης μη επιτρεπτής τιμής.....	66
Εικόνα 5.28: Το μήνυμα για μετατροπή των μονάδων μέτρησης	67
Εικόνα 5.29: Το αντικείμενο plotOptions	68
Εικόνα 5.30: Η setPlotData	68
Εικόνα 5.31: Το Row μέσα στο οποίο απεικονίζονται οι εικόνες των κυκλωμάτων και το γράφημα ..	68
Εικόνα 5.32: Τα σημεία $V_{CE(CUTOFF)}$, Q και $I_{C(SAT)}$ στο γράφημα.....	69
Εικόνα 5.33: Το μήνυμα που εμφανίζεται όταν $V_C < V_B$	70
Εικόνα 5.34: Το σημείο που δημιουργείται το blinking του πεδίου.....	70
Εικόνα 5.35: Λάθος λειτουργία λόγω αρνητικής τάσης ή/και ρεύματος	71
Εικόνα 5.36: Εμφάνιση και των δύο μηνυμάτων ταυτοχρόνως	72
Εικόνα 5.37: Σταθερός διαιρέτης τάσης	73
Εικόνα 5.38: Πολύ σταθερός διαιρέτης τάσης	74
Εικόνα 5.39: Κύκλωμα πόλωσης βάσης	75
Εικόνα 5.40: Κύκλωμα πόλωσης εκπομπού.....	76
Εικόνα 5.41: Κύκλωμα πόλωσης βάσης – λειτουργία στον κορεσμό	77
Εικόνα 5.42: Κύκλωμα πόλωσης βάσης – λειτουργία στην αποκοπή	78
Εικόνα 5.43: Κύκλωμα πόλωσης εκπομπού – λειτουργία στην ενεργό περιοχή.....	79
Εικόνα 5.44: Κύκλωμα πόλωσης εκπομπού – λειτουργία στον κορεσμό.....	80

Εικόνα 5.45: Κύκλωμα πόλωσης εκπομπού – λειτουργία στην αποκοπή	81
Εικόνα 5.46: Αναστρέφων ενισχυτής	82
Εικόνα 5.47: Ο κώδικας του index.jsx.....	83
Εικόνα 5.48: Η εμφάνιση του μηνύματος όταν γίνεται mouseover στη V_{CC}	83
Εικόνα 5.49: Ημιτονοειδή σήματα για την είσοδο και την έξοδο	84
Εικόνα 5.50: Προειδοποιητικό μήνυμα και ψαλιδισμός του σήματος εξόδου	85
Εικόνα 5.51: Προβολή σημάτων ως παλμοί.....	86
Εικόνα 5.52: Προειδοποιητικό μήνυμα.....	87
Εικόνα 5.53: Η μεταβλητή bool περνάει ως props στο CircuitValueInput για το blinking της V_{out}	87
Εικόνα 5.54: Η boolean μεταβλητή bool και η μεταβλητή x που χρησιμοποιείται για τα γραφήματα .	87
Εικόνα 5.55: Ο κώδικας για την εμφάνιση του προειδοποιητικού μηνύματος	88
Εικόνα 5.56: Το πέρασμα δεδομένων μέσω props στα SineWave και Pulse ανάλογα με την τιμή του opt	88
Εικόνα 5.57: Μη αναστρέφων ενισχυτής – προβολή σημάτων ως ημίτονα	88
Εικόνα 5.58: Μη αναστρέφων ενισχυτής – προβολή σημάτων ως παλμοί.....	89
Εικόνα 5.59: Αθροιστικός ενισχυτής.....	90
Εικόνα 5.60: Ενισχυτής διαφορών.....	91
Εικόνα 5.61: Απόδοση default τιμών.....	92
Εικόνα 5.62: Η συνάρτηση restore	92
Εικόνα 5.63: Ο πίνακας validations.....	93
Εικόνα 5.64: Εμφάνιση δεκαδικών στα αποτελέσματα	93
Εικόνα 5.65: Η onClick που κάνει update τη μονάδα μέτρησης.....	94
Εικόνα 5.66: Η useEffect που κάνει τον έλεγχο currentMeasurement με measurements.length	94
Εικόνα 5.67: Εμφάνιση του E στην Re σαν κεφαλαίο small-cap.....	94
Εικόνα 5.68: Η συνάρτηση getLabel ().....	94
Εικόνα 5.69: Το circuitValueGroup.....	95
Εικόνα 5.70: Το κείμενο για προτροπή αλλαγής μονάδας μέτρησης	96
Εικόνα 5.71: Το σημείο που ενώνονται οι γραμμές των παλμών εισόδου.....	98
Εικόνα 6.1: Επιτυχία ή μη στα παραπάνω μαθήματα	100
Εικόνα 6.2: Βαθμός ικανοποίησης για τη σχεδίαση της εφαρμογής	100
Εικόνα 6.3: Ευχρηστία εφαρμογής	101
Εικόνα 6.4: Λειτουργικότητα εφαρμογής	101
Εικόνα 6.5: Απαντήσεις στην ερώτηση για τη χρήση της εφαρμογής ως συμπληρωματικό βοήθημα στα παραπάνω μαθήματα	102
Εικόνα 6.6: Χρήση ως βοήθημα για την καλύτερη κατανόηση των εννοιών των μαθημάτων	102
Εικόνα 6.7: Χρήση της εφαρμογής για τη μελέτη των παραπάνω μαθημάτων.....	103
Εικόνα 6.8: Απαντήσεις στην ερώτηση για απαίτηση προηγούμενων γνώσεων πριν τη χρήση της εφαρμογής	103
Εικόνα 6.9: Σύσταση εφαρμογής σε τρίτους.....	104
Εικόνα 6.10: Συσχέτιση επιτυχόντων και μη με το βαθμό ευχρηστίας της εφαρμογής.....	105
Εικόνα 6.11: Συσχέτιση επιτυχόντων και μη με το βαθμό λειτουργικότητας της εφαρμογής	105
Εικόνα 6.12: Συσχέτιση επιτυχόντων και μη με την πιθανή χρήση της εφαρμογής ως συμπληρωματικό βοήθημα	106
Εικόνα 6.13: Γράφημα απεικόνισης των αποτελεσμάτων συνολικά	106

Συντομογραφίες

Δ.Ε.	Διπλωματική Εργασία
ΔΙΠΙΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
BJT	Bipolar Junction Transistor
TE	Τελεστικός Ενισχυτής
SR	Slew Rate
MPP	Maximum Peak to Peak
VS Code	Visual Studio Code

Κεφάλαιο 1ο: Εισαγωγή

1.1 Περιγραφή της ιδέας

Στη σύγχρονη εποχή, η συνεχής εξέλιξη επιστημονικών πεδίων όπως η ηλεκτρονική και η πληροφορική, έχουν δημιουργήσει την ανάγκη για εκπαίδευση η οποία επιτάσσει συνδυασμό στοχευμένου θεωρητικού και πρακτικού υποβάθρου. Εξαιτίας της συνεχώς αυξανόμενης πολυπλοκότητας των προαναφερθέντων αντικειμένων, καθίσταται σχεδόν επιτακτική η δημιουργία έξυπνων εργαλείων που θα μπορούν να βοηθήσουν στην ευκολότερη και πληρέστερη κατανόηση των εννοιών τους. Με αφορμή αυτό, κρίθηκε αναγκαία η υλοποίηση μιας διαδραστικής εφαρμογής η οποία αφορά κυρίως τους φοιτητές των μαθημάτων ‘Ηλεκτρονικά Κυκλώματα’ και ‘Ηλεκτρονικές Διατάξεις’ αλλά και οποιονδήποτε θέλει να πειραματιστεί πάνω στην ανάλυση κυκλωμάτων. Η ιδέα είναι ο χρήστης να μπορεί να επιλέξει το κύκλωμα που τον ενδιαφέρει. Η επιλογή ξεκινά από την κατηγορία της αρεσκείας (διπολικά τρανζίστορ ή τελεστικοί ενισχυτές) και πειραματιζόμενος με τις τιμές των παραμέτρων, ο φοιτητής ενδέχεται να είναι σε θέση να κατανοήσει και να εξάγει χρήσιμα συμπεράσματα για έννοιες οι οποίες συμπεριλαμβάνονται στο πλαίσιο της διδασκαλίας των παραπάνω μαθημάτων, όπως:

- Ποια είναι η περιοχή λειτουργίας ενός κυκλώματος βάσει των τιμών των παραμέτρων του και γιατί
- Πώς σχεδιάζεται η γραμμή φορτίου σε ένα διάγραμμα $I_C - V_{CE}$ και πώς κινείται το σημείο λειτουργίας – ηρεμίας πάνω σε αυτήν
- Πώς απεικονίζεται η κυματομορφή εισόδου – εξόδου σε ένα διάγραμμα τάσης συναρτήσει του χρόνου για τον αναστρέφων και μη αναστρέφων τελεστικό ενισχυτή
- Πώς επηρεάζει ο ρυθμός μεταβολής (slew rate) το σχεδιασμό ενός παλμού στην είσοδο και έξοδο του ενισχυτή
- Ποια πρέπει να είναι τα όρια τροφοδοσίας του τελεστικού ενισχυτή και γιατί
- Τι είναι η διαφορά κορεσμού εξόδου από την τάση τροφοδοσίας

1.2 Οργάνωση ενοτήτων

Στο παρόν πρώτο κεφάλαιο γίνεται μια εισαγωγή στο γενικότερο πλαίσιο στο οποίο εντάσσεται η παρούσα διπλωματική εργασία, δίνονται κάποιες πληροφορίες για τη λειτουργία των virtual labs και πώς αυτά συνδέονται με την εφαρμογή που υλοποιήθηκε και τέλος αναφέρονται οι επιθυμητοί στόχοι της.

Στο δεύτερο κεφάλαιο εξηγείται λεπτομερώς η λειτουργία του διπολικού τρανζίστορ και του τελεστικού ενισχυτή, συνοδευόμενη από την ανάλυση των βασικών συνδεσμολογιών που χρησιμοποιούνται στην εφαρμογή.

Το τρίτο κεφάλαιο αποτελεί συνέχεια του δεύτερου, καθώς σε αυτό υλοποιούνται παραδείγματα βασισμένα στα κυκλώματα τα οποία απαρτίζουν το υλοποιημένο εργαλείο, μαζί με την εξήγηση πρακτικών εννοιών πάνω σε αυτά.

Ακολουθεί το τέταρτο κεφάλαιο, στο οποίο γίνεται η ανάλυση του προγραμματιστικού περιβάλλοντος πάνω στο οποίο δημιουργήθηκε η εφαρμογή, καθώς και οι τεχνολογίες που χρησιμοποιήθηκαν για το σκοπό αυτό.

Προχωρώντας, το πέμπτο κεφάλαιο αφορά την ανάλυση της εφαρμογής, οτιδήποτε δηλαδή σχετίζεται με την ανάπτυξη του παραγόμενου λογισμικού, καθώς και με τη συνολική λειτουργία της εφαρμογής. Συνοδεύεται κι από ποικίλες εικόνες που αναδεικνύουν τη λειτουργία του έργου.

Κατόπιν, στο έκτο κεφάλαιο περιλαμβάνεται η ανάλυση και τα αποτελέσματα που προέκυψαν από τις απαντήσεις σε ερωτηματολόγιο με στόχο την αξιολόγηση της ορθής λειτουργίας της εφαρμογής.

Τέλος, στο έβδομο και τελευταίο κεφάλαιο αναφέρονται τα συμπεράσματα και οι πιθανές προτάσεις βελτίωσης σε μελλοντικές επεκτάσεις της εφαρμογής.

1.3 Στόχοι

Ο στόχος της εφαρμογής είναι αφενός να γίνει ένα χρήσιμο εργαλείο εκμάθησης εννοιών της επιστήμης της ηλεκτρονικής και αφετέρου, μέσω της μαθησιοκεντρικής προσέγγισης που την χαρακτηρίζει, να δώσει τη δυνατότητα σε οποιονδήποτε φοιτητή που είτε αντιμετωπίζει δυσκολίες στην κατανόηση, είτε για οποιοδήποτε λόγο δεν έχει πρόσβαση σε εργαστηριακούς πόρους, είτε επιθυμεί να μάθει με τον δικό του ρυθμό, να πειραματιστεί σε ένα διαδραστικό web-based περιβάλλον που θα τον βοηθήσει να καταλάβει καλύτερα τα αντικείμενα του μαθήματος, χωρίς την απαίτηση για οποιαδήποτε γνώση πάνω στον προγραμματισμό. Στοχεύει επίσης να αποτελέσει σημαντικό βοήθημα στο έργο των καθηγητών για μεταλαμπάδευση της γνώσης στους φοιτητές και ίσως και έναν τρόπο παρακίνησης πληθώρας φοιτητών να ασχοληθούν με το μάθημα, ξεφεύγοντας από το συνηθισμένο τρόπο παράδοσης μέσω επίλυσης ασκήσεων σε πίνακα.

1.4 Δομή και χρησιμότητα των virtual labs

Μία από τις κρισιμότερες υποχρεώσεις ενός φοιτητή θεωρείται η πλήρης κατανόηση των θεμάτων που του επιτρέπουν να χειρίζεται με ευκολία το εκάστοτε αντικείμενο. Ωστόσο, απ' τη σκοπιά της ηλεκτρονικής, είναι αρκετά δύσκολο κάποιος να μπορέσει να αποκτήσει τη βασική γνώση και τις δεξιότητες χωρίς τη χρήση εργαστηριακού εξοπλισμού. Λύση στο πρόβλημα αυτό δίδεται από τα virtual labs. Μερικοί από τους ορισμούς που χρησιμοποιούνται για το τι είναι ένα virtual lab αναφέρονται παρακάτω:

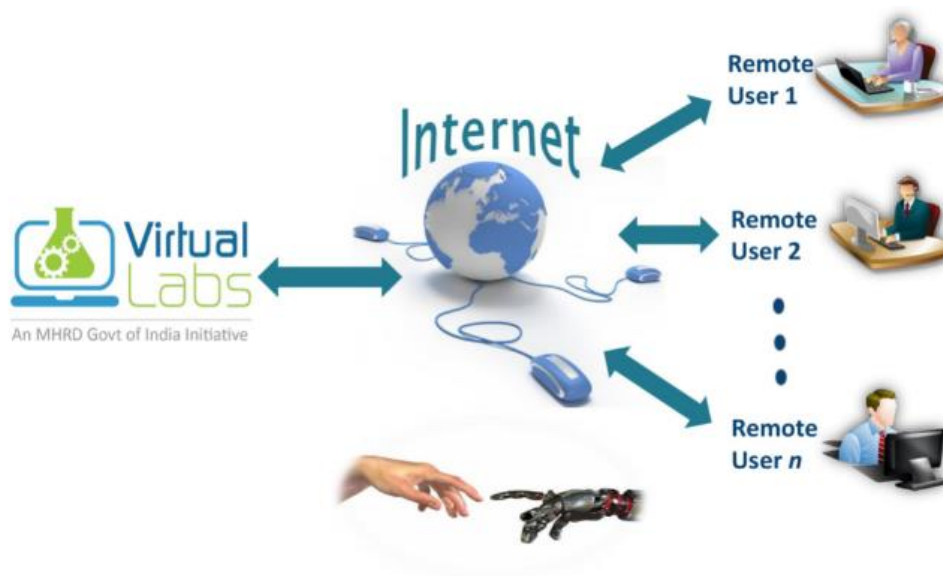
- Οποιοδήποτε λογισμικό ή ιστοσελίδα που έχει αναπτυχθεί για διαδραστική μάθηση η οποία βασίζεται σε προσομοιώσεις αληθινών φαινομένων. Είναι ένας τρόπος να μπορέσουν οι εκπαιδευόμενοι να εξερευνήσουν ένα θέμα, να συγκρίνουν διάφορα σενάρια, να μπορέσουν να σταματήσουν την προσομοίωση για να κρατήσουν σημειώσεις και να αποκτήσουν εμπειρία στην εκτέλεση πειραμάτων χρησιμοποιώντας H/Y [1].
- Η προσπάθεια για δημιουργία πραγματικού εργαστηρίου μέσω της προσομοίωσης λογισμικού. Σε σύγκριση με ένα παραδοσιακό εργαστήριο, ένα virtual lab μπορεί να αποδειχθεί πολύ χρήσιμο σε περιπτώσεις όπου κάποια πειράματα περιλαμβάνουν εξοπλισμό που μπορεί να έχει επιζήμιες για την ανθρώπινη ασφάλεια συνέπειες [1].
- Ένα προσομοιωμένο εργαστηριακό περιβάλλον υλοποιημένο σαν ένα λογισμικό προγράμματος, το οποίο επιτρέπει στους χρήστες να εκτελέσουν τα πειράματά τους [1].

Σίγουρα η χρήση των virtual labs δεν μπορεί να αντικαταστήσει εξ' ολοκλήρου τα πραγματικά εργαστήρια. Ωστόσο τα πλεονεκτήματα που προσφέρουν είναι ικανά να αντισταθμίσουν σε μεγάλο βαθμό αυτή την απόκλιση. Τα κυριότερα είναι:

- Διευκολύνουν την εξάσκηση κάποιου μέσω του διαδικτύου σε οποιοδήποτε μέρος και οποιαδήποτε στιγμή το επιθυμεί [19].
- Ενθαρρύνουν το μαθητευόμενο κοινό στην αυτόνομη μάθηση, όπερ σημαίνει ότι ο φοιτητής μπορεί να μάθει μόνος του χωρίς τη βοήθεια καθηγητή [19].

- Ο χρήστης έχει συνεχή πρόσβαση σε εργαστηριακό εξοπλισμό (έστω και εικονικά) [19].
- Μηδενίζουν τον κίνδυνο ατυχήματος σε περίπτωση χειρισμού επικίνδυνου εξοπλισμού (ραδιενέργεια κάθε είδους) [19].
- Μπορεί να φέρει σε επαφή φοιτητές από διαφορετικά πανεπιστήμια, κάτι το οποίο θα βοηθήσει στην ανταλλαγή απόψεων μεταξύ τους [19].

Συνεπώς, τα virtual labs μπορούν να αποτελέσουν ένα εξαιρετικό εκπαιδευτικό συμπλήρωμα τόσο για εκπαιδευτικούς όσο και για φοιτητές. Μέσω του φιλικού προς το χρήστη UI που διαθέτουν, είναι σε θέση να παρακινήσουν και να δώσουν κίνητρο σε οποιονδήποτε σκέφτεται να τα χρησιμοποιήσει [20]. Μάλιστα, στα περισσότερα από αυτά, περιλαμβάνεται και οδηγός, ο οποίος έχει το ρόλο του καθηγητή, κάτι που τα καθιστά ακόμα πιο ελκυστικά [2], [3].



Εικόνα 1.1: Λογική ενός virtual lab

Στην εικόνα 1.2 βλέπουμε τη δομή ενός virtual lab που αφορά την εκμάθηση των βασικών εννοιών της ηλεκτρονικής. Όπως παρατηρούμε, υπάρχουν διάφορα tabs που μπορεί να δει ο χρήστης, ανάμεσα στα οποία και τεχνικές οδηγίες για τη χρήση του εικονικού εργαστηρίου. Πιο συγκεκριμένα, γίνεται μια εισαγωγή, στην οποία εξηγείται ο σκοπός δημιουργίας του συγκεκριμένου εικονικού εργαστηρίου, ο οποίος είναι η εκτέλεση πειραμάτων εικονικά, τα οποία ανταποκρίνονται όσο το δυνατόν περισσότερο στην πραγματικότητα. Κατόπιν επισημαίνονται τα οφέλη που θα αποκομίσουν από τη χρήση του οι φοιτητές, όπου μέσω της μοντελοποίησης και προσομοίωσης των οργάνων και των υλικών θα έχουν την αίσθηση ότι λειτουργούν σε πραγματικό εργαστηριακό περιβάλλον, αλλά και καθηγητές, οι οποίοι μέσω της συγκεκριμένης πλατφόρμας, θα είναι σε θέση να συλλέξουν συμπληρωματικό υλικό για να μπορέσουν να βοηθήσουν τους μαθητευόμενους τους στην πιθανή κατανόηση δύσκολων εννοιών [3]. Ένα ακόμη πλεονέκτημα που έχει το συγκεκριμένο virtual lab είναι ότι δεδομένης της καθοδήγησης που παρέχει, καθιστά πιο δύσκολο το ενδεχόμενο ένας όχι και τόσο καλός σε θέματα ευπαθείων και ασφάλειας που μπορούν να προκύψουν από τη χρήση του διαδικτύου φοιτητής, να μη χρειάζεται γνώση που αφορά το πώς μπορεί να αντιμετωπίσει και να αποτρέψει τέτοιου είδους ζητήματα [21]. Οπότε, το μόνο που χρειάζεται είναι ο χρήστης να έχει πρόσβαση στο διαδίκτυο και έναν web browser στον υπολογιστή του [22].

Welcome to Basic Electronics Virtual Laboratory!



Introduction

The Virtual Labs Project started as an initiative from the Ministry of Human Resource and Development (MHRD) to create online interactive media which would help students learn difficult concepts in various domains. As a part of this initiative, a virtual laboratory for Basic Electronics has been developed. The objective of this lab is to perform experiments in the Basic Electronics labs virtually, and yet have close to real life experience. The platform is focused on learning aspects as much as on performing the experiments.

Virtual Labs for Teachers

Virtual Labs has the potential to bring paradigm shift in the way courses are taught today. For example, the teachers can take it to the class and demonstrate live examples while they are teaching difficult concepts to the students. Many difficult concepts are already integrated in the provided course curriculum. It can also complement the existing labs, wherein the students can be given assignments before or after the lab. The teacher can actively coordinate that effort to enable inclusive learning for all the students.

Virtual Labs for Students

Students will get a feel of the real lab, as all lab equipment, components and lab behavior would be mimicked in the system. Extensive 2D animations is used to provide real like look and feel to the students. The instruments and components are modeled accurately to provide real like response. Interactive animations are also provided at a lot of places to enable student learning.

Key features of Virtual Labs

1. Available 24X7. So students can learn difficult concepts at their own pace.
2. Real looking components, to give the user a feel of the real laboratory.
3. Step by step wizard to guide the students through the experiments.

Εικόνα 1.2: Δομή ενός virtual lab

Welcome to Basic Electronics Virtual Laboratory!


[INTRODUCTION](#)

[LIST OF EXPERIMENTS](#)

[INSTRUCTION](#)

[TECHNOLOGY USED](#)

[PEOPLE](#)

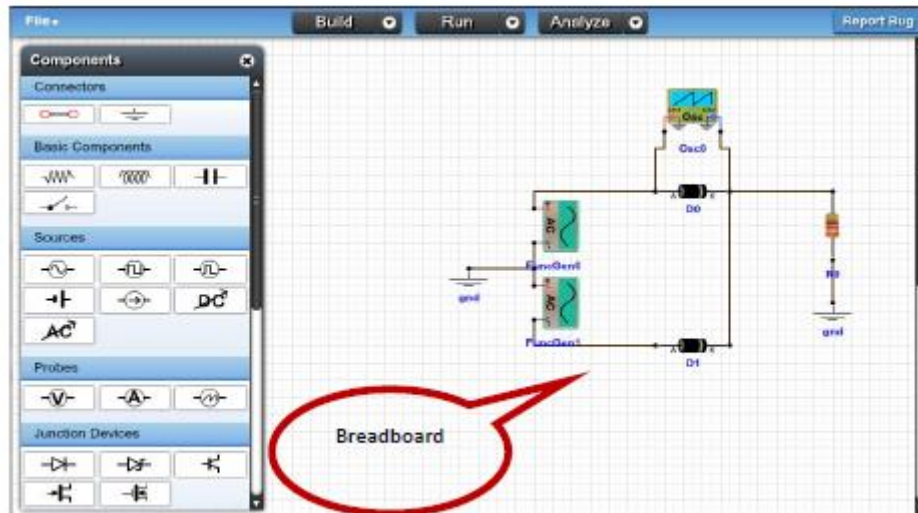
[FEEDBACK](#)

List of experiments

- Experiments Developed
 1. [Familiarisation with Resistor](#)
 2. [Familiarisation with Capacitor](#)
 3. [Familiarisation with Inductor](#)
 4. [Ohm's Law](#)
 5. [VI Characteristics of a Diode](#)
 6. [Half Wave Rectification](#)
 7. [Full Wave Rectification](#)
 8. [Capacitive Rectification](#)
 9. [Zener Diode-Voltage Regulator](#)
 10. [BJT Common Emitter Characteristics**](#)
 11. [BJT Common Base Characteristics**](#)
 12. [Studies on BJT CE Amplifier](#)
 13. [RC Frequency Response](#)
 14. [RC Differentiator and Integrator](#)
 15. [Black Box](#)
 16. [Study of basic properties of Operational Amplifier: Inverting and Non-Inverting Amplifiers**](#)
 17. [Study of Differentiator and Integrator using Operational Amplifier**](#)

Εικόνα 1.3: Η λίστα με τα πειράματα

Για να καταλάβει ακόμα καλύτερα ο αναγνώστης του παρόντος συγγράμματος πώς λειτουργεί ένα virtual lab, παρατίθεται η εικόνα 1.4 και 1.5, στις οποίες φαίνεται, με τη χρήση ενός web-based προσομοιωτή κυκλωμάτων ονόματι DoCircuits, η πλατφόρμα του εικονικού εργαστηρίου και η οπτικοποίηση των αποτελεσμάτων έπειτα από την προσομοίωση [23].



Εικόνα 1.4: Web-based πλατφόρμα του προσομοιωτή DoCircuits



Εικόνα 1.5: Οπτικοποίηση των αποτελεσμάτων με τη χρήση του εργαλείου DoCircuits

Με βάση λοιπόν τα παραπάνω, έγινε προσπάθεια η εφαρμογή να είναι όσο πιο κοντά γίνεται στα πλαίσια ενός virtual lab, προσπαθώντας να φέρει σε άμεση επαφή τον φοιτητή με τα ηλεκτρονικά κυκλώματα που πρόκειται να μελετήσει, δίχως την απαίτηση εργαστηριακού εξοπλισμού και επιτρέποντάς του να ακολουθήσει τον ρυθμό μελέτης που αυτός επιθυμεί και να το κάνει από την ασφάλεια του σπιτιού του [24].

1.5 Επίλογος

Στο παρόν εισαγωγικό κεφάλαιο έγινε μια γενική εισαγωγή στην ιδέα της εφαρμογής, παρουσιάστηκε η δομή της εργασίας οργανωμένη σε κεφάλαια, προσδιορίστηκε ο στόχος εκπόνησής της και τέλος έγινε αναφορά στα virtual labs, τον τρόπο με τον οποίο αυτά λειτουργούν και πώς συσχετίζονται με την δική μας εφαρμογή.

Κεφάλαιο 2ο: Κυκλώματα που χρησιμοποιήθηκαν

2.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα εξεταστούν οι διατάξεις και τα κυκλώματα τα οποία επιλέχθηκαν για την υλοποίηση της εφαρμογής. Οι πληροφορίες που αφορούν την ανάλυση λειτουργίας του ηρη τρανζίστορ και των ΤΕ, με όλες τις επιμέρους παραγράφους τους, προέρχονται κατά κόρον από το βιβλίο των Albert Malvino και David J. Bates [25].

2.2 Διπολικά τρανζίστορ

2.2.1 Ιστορικά στοιχεία και ορισμοί

Το τρανζίστορ (κρυσταλλοτρίδος στα ελληνικά) είναι μια ημιαγωγική διάταξη στερεάς κατάστασης αποτελούμενη από τρεις ζώνες ημιαγωγού στοιχείου οι οποίες είναι διαδοχικά τοποθετημένες. Ένα τρανζίστορ μπορεί να λειτουργήσει είτε αυτόνομα, είτε ως εξάρτημα ενός ολοκληρωμένου κυκλώματος, ανάλογα με τη λειτουργία που επιθυμούμε να έχει. Έχει διάφορες εφαρμογές όπως σταθεροποίηση τάσης, διαμόρφωση συχνότητας, ενίσχυση, λειτουργία ως διακόπτης κλπ [5], [6]. Στην παρούσα εργασία πραγματευόμαστε τον όρο *διπολικό τρανζίστορ επαφής*. Πρόκειται για συντομογραφία του ‘δύο πολικότητες’ και κάνει χρήση ελευθέρων ηλεκτρονίων και οπών. Το πρώτο τρανζίστορ επαφής παρουσιάστηκε στους προϊσταμένους των εργαστηρίων Bell Telephone στις 23/12/1947 από τους Walter O. Brattain και John Bardeen, οι οποίοι χρησιμοποίησαν ως υλικά το κρυσταλλικό γερμάνιο μαζί με επαφές χρυσού και στόχος τους ήταν να αποδείξουν τη λειτουργία του ως ενισχυτή, υλοποιώντας μια βαθμίδα ενίσχυσης.



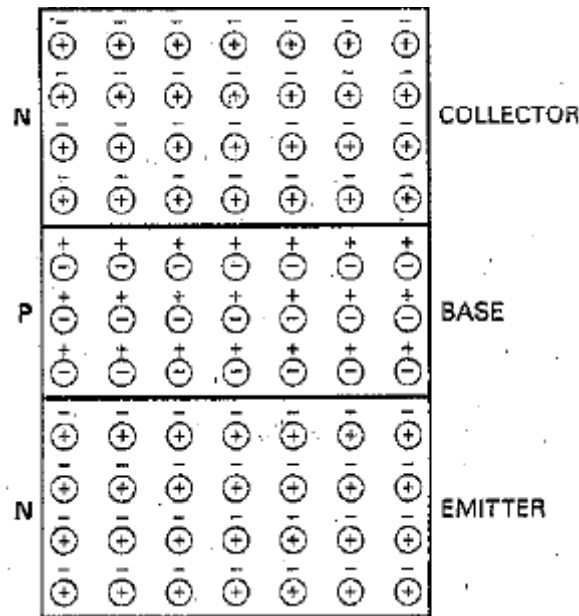
Εικόνα 2.1 Το πρώτο πειραματικό τρανζίστορ

Ο William Shockley ήθελε να λανσάρει στην αγορά ένα νέο είδος τρανζίστορ, το οποίο αντί των επαφών χρυσού θα βασίζεται σε κόμβους. Έτσι, εφηύρε το διπολικό τρανζίστορ επαφής (**Bipolar Junction Transistor**) του οποίου την κυκλοφορία ανακοίνωσε στις 4/7/1951. Ο όρος *διπολικό*, που όπως αναφέρθηκε αφορά την πολικότητα του τρανζίστορ, μπορεί να λειτουργήσει είτε ως ηρη (πιο ευρέως διαδεδομένο σήμερα) είτε ως ηρη [5], [6]. Λόγω της συνηθέστερης χρήσης του ηρη τρανζίστορ, τα κυκλώματα της εφαρμογής πραγματεύονται μόνο αυτό και όχι το ηρη. Τα διπολικά τρανζίστορ, τα οποία χρησιμοποιούν ελεύθερα ηλεκτρόνια και οπές, ήταν ο πλέον διαδεδομένος τύπος τρανζίστορ στις δεκαετίες του 1960 και 1970 καθώς, ακόμα και μετά την κυκλοφορία των MOSFET,

παρέμειναν στο προσκήνιο χάρη στην απλότητα κατασκευής και χρησιμότητάς τους σε αναλογικά κυκλώματα [5], [6].

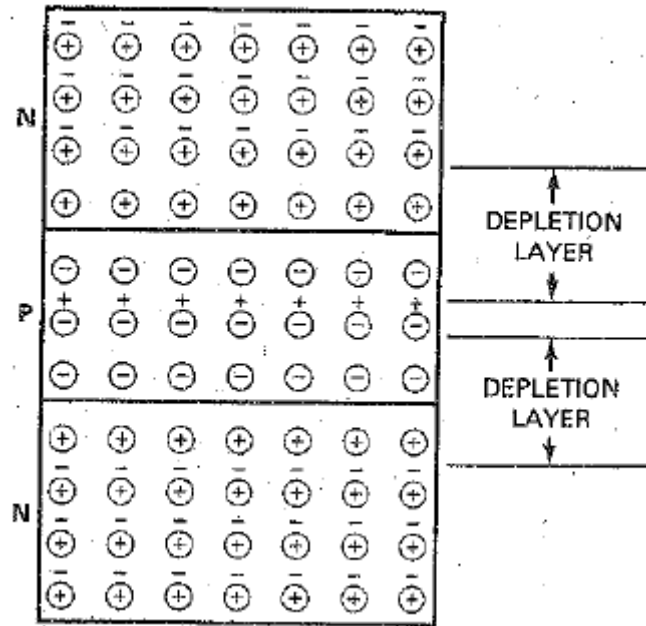
2.2.2 Λειτουργία npn τρανζίστορ

Αρχικά, είναι σκόπιμο να αναφερθεί ότι ένα τρανζίστορ αποτελείται από τρεις περιοχές. Αυτές είναι: ο εκπομπός, η βάση και ο συλλέκτης. Στην εικόνα 2.2 βλέπουμε τη δομή του τρανζίστορ. Όπως παρατηρούμε, υπάρχουν ελεύθερα ηλεκτρόνια στο υλικό τύπου n και οπές στο υλικό τύπου p.



Εικόνα 2.2 Δομή του τρανζίστορ

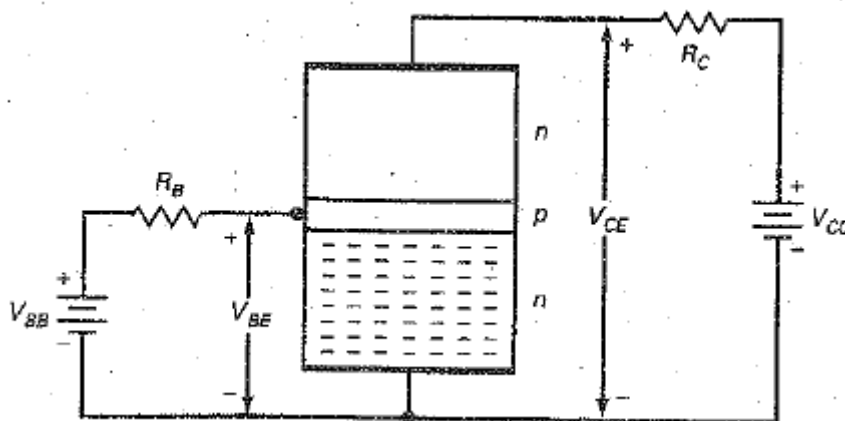
Το συγκεκριμένο τρανζίστορ είναι τύπου npn διότι ανάμεσα σε δύο περιοχές n παρεμβάλλεται μια περιοχή p. Έχει δύο επαφές, μία εκπομπού – βάσης και μία βάσης – συλλέκτη. Στην ουσία πρόκειται για δύο διόδους συνδεδεμένες μεταξύ τους. Η παραπάνω εικόνα απεικονίζει τη δομή ενός τρανζίστορ το οποίο δεν είναι πολωμένο. Στην περίπτωση αυτή, τα ηλεκτρόνια που βρίσκονται στην περιοχή του εκπομπού και του συλλέκτη, κατευθύνονται προς τη βάση ενώ κάποιες από τις οπές της βάσης διαχέονται προς τον εκπομπό και τον συλλέκτη, κάτι που έχει ως αποτέλεσμα να δημιουργείται μια περιοχή απογύμνωσης στις επαφές εκπομπού και συλλέκτη (Εικόνα 2.3). Κατά μήκος της κάθε επαφής αναπτύσσεται ένα φράγμα δυναμικού περίπου στα 0,7V στους 25°C (για ημιαγωγό πυριτίου, ο οποίος χρησιμοποιείται περισσότερο λόγω της μικρότερης εξάρτησης των χαρακτηριστικών του από την θερμοκρασία). Για να θεωρηθεί πολωμένο ένα τρανζίστορ χρειαζόμαστε την εφαρμογή ρεύματος. Όπως σε κάθε συσκευή ισχύος, έτσι και στο τρανζίστορ ρέει ρεύμα. Για να διοχετευθεί το ρεύμα, πρέπει να έχουμε κάποια εξωτερική πηγή τάσης η οποία θα το τροφοδοτεί. Επειδή τα επίπεδα εμπλουτισμού είναι διαφορετικά στις 3 προαναφερθείσες περιοχές, είναι σκόπιμο να γίνει διαχωρισμός της κίνησης του φορτίου από την μία περιοχή στην άλλη.



Εικόνα 2.3 Περιοχές απογύμνωσης

2.2.2.1 Ηλεκτρόνια εκπομπού

Όπως προαναφέρθηκε, για να έχουμε πόλωση σε ένα τρανζίστορ πρέπει σε αυτό να κυκλοφορεί σταθερό ρεύμα. Όπως βλέπουμε και στην εικόνα 2.4, τα ελεύθερα ηλεκτρόνια που βρίσκονται στην περιοχή του εκπομπού έχουν ως στόχο να καταφέρουν να περάσουν στο συλλέκτη. Για να γίνει αυτό, μεσολαβεί η βάση. Οπότε τα ηλεκτρόνια θα πρέπει πρώτα να περάσουν από εκεί. Η μεταφορά αυτή μπορεί να γίνει μόνο όταν η διόδος του εκπομπού είναι πολωμένη ορθά. Υπό αυτές τις συνθήκες, η περιοχή της βάσης, όντας πιο λεπτή από τις άλλες δύο και λιγότερο γεμάτη από φορτίο, θα είναι πλέον σε θέση να επιτρέψει τη διέλευση των ηλεκτρονίων από τον εκπομπό προς τον συλλέκτη. Για να είναι σε θέση ο συλλέκτης να πάρει αυτά τα ηλεκτρόνια θα πρέπει η διόδός του να είναι πολωμένη ανάστροφα.

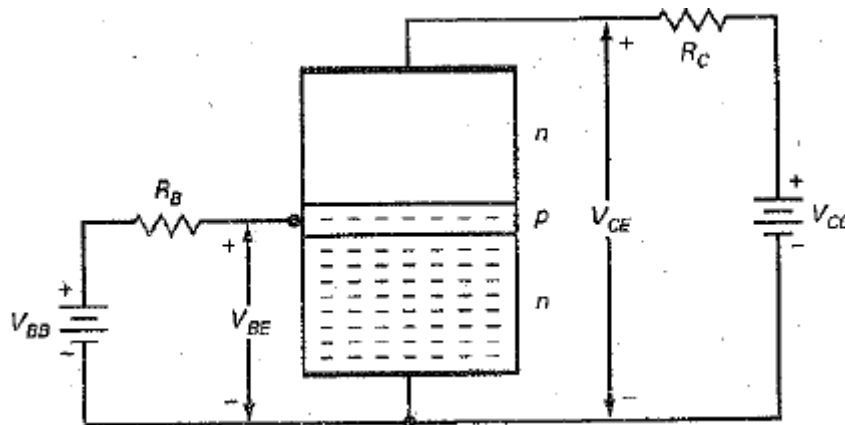


Εικόνα 2.4 Πολωμένο τρανζίστορ

Έτσι λοιπόν, η επίτευξη της ορθής πόλωσης διόδου εκπομπού και της ανάστροφης πόλωσης διόδου συλλέκτη επιτυγχάνονται, μέσω των πηγών V_{BE} και V_{CC} αντίστοιχα.

2.2.2.2 Ηλεκτρόνια βάσης

Για να μπορέσει να υπάρξει ροή ηλεκτρονίων από τον εκπομπό στην περιοχή της βάσης και ελευθέρων οπών από τη βάση στον εκπομπό θα πρέπει $V_{BE} > 0,7V$. Στην περίπτωση που $V_{BE} < 0,7V$ τότε πρακτικά η διόδος διαρρέεται από μηδενικό ρεύμα. Από τη στιγμή που τα ηλεκτρόνια έχουν εισέλθει στην περιοχή της βάσης, μπορούν να ακολουθήσουν δύο κατευθύνσεις: α) Είτε να κινηθούν προς το συλλέκτη είτε β) προς το θετικό άκρο της πηγής V_{BB} . Η ανάστροφη πόλωση της διόδου συλλέκτη δεν επιτρέπει στη βάση να διατηρήσει τα ηλεκτρόνια, τα οποία λόγω και του ηλεκτρικού πεδίου που υπάρχει στην περιοχή του συλλέκτη, κατευθύνονται προς αυτόν. Σε αυτό συμβάλλει επίσης και η πολύ λεπτή βάση, που σημαίνει ότι τα ηλεκτρόνια βρίσκονται πολύ κοντά στην επαφή του συλλέκτη. Τα ελάχιστα ηλεκτρόνια που έχουν απομείνει στην περιοχή της βάσης, μαζί με τις οπές, θα κατευθυνθούν μέσω της αντίστασης R_B στο θετικό άκρο της πηγής, ως ηλεκτρόνια σθένους.



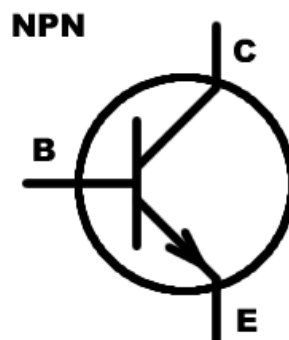
Εικόνα 2.5 Ελεύθερα ηλεκτρόνια έχουν εισαχθεί μέσω της διόδου εκπομπού στη βάση

2.2.2.3 Ηλεκτρόνια συλλέκτη

Αφού έχει ολοκληρωθεί ο διαχωρισμός των ηλεκτρονίων που κινούνται προς την θετική πλευρά της V_{BB} και στον συλλέκτη, όσα έχουν εισχωρήσει στην περιοχή του συλλέκτη, κατευθύνονται στην πηγή V_{CC} . Αυτό επιτυγχάνεται μέσω της αντίστασης συλλέκτη R_C , η οποία τα βοηθάει να οδηγηθούν στο θετικό άκρο της τάσης τροφοδοσίας του συλλέκτη.

2.2.3 Ρεύματα και σχέσεις μεταξύ τους

Μέχρι εδώ έχει γίνει περιγραφή για τη φυσική δομή του τρανζίστορ. Στην εφαρμογή μας, αλλά και σε ασκήσεις ηλεκτρονικής, συνήθως χρησιμοποιείται με το παρακάτω σύμβολο:



Εικόνα 2.6 Σύμβολο npn τύπου τρανζίστορ

Αναγνωρίζουμε ότι πρόκειται για το τρανζίστορ npn , το οποίο μελετάμε, παρατηρώντας ότι το βέλος του εκπομπού ξεκινάει από τη βάση και καταλήγει σε αυτόν. Εάν διέγραφε την αντίστροφη πορεία θα μιλούσαμε για το pnp τρανζίστορ [4]. Όπως έχει ήδη αναφερθεί παραπάνω, για να λειτουργήσει το τρανζίστορ πρέπει να είναι ορθά πολωμένο. Αυτό σημαίνει ότι πρέπει να κυκλοφορεί ρεύμα στις περιοχές του. Τρία είναι τα ρεύματα που ρέουν στο εσωτερικό του:

- Το ρεύμα εκπομπού (I_E)
- Το ρεύμα βάσης (I_B)
- Το ρεύμα συλλέκτη (I_C)

Στηριζόμενοι στον νόμο του Kirchhoff για τα ρεύματα, όπου το άθροισμα των ρευμάτων που εισέρχονται σε έναν κόμβο είναι ίσο με το άθροισμα αυτών που εξέρχονται από τον ίδιο κόμβο, δημιουργείται η γενική εξίσωση:

$$I_E = I_C + I_B \quad (2.1)$$

Λόγω του ότι η ροή των ηλεκτρονίων στο τρανζίστορ ξεκινάει από τον εκπομπό, προφανώς αυτός θα έχει το μεγαλύτερο ρεύμα. Καθώς σχεδόν όλα τα ηλεκτρόνια περνάνε μέσω της βάσης στον συλλέκτη, εξάγουμε το συμπέρασμα ότι το ρεύμα εκπομπού και το ρεύμα συλλέκτη είναι περίπου ίσα (το ρεύμα από την ανάστροφη πόλωση της διόδου συλλέκτη θεωρείται αμελητέο). Οπότε προκύπτει η σχέση:

$$I_C \approx I_E \quad (2.2)$$

Η μικρή μεταξύ τους διαφορά έγκειται στον συντελεστή α_{dc} , ο οποίος ορίζεται από τη σχέση:

$$\alpha_{dc} = \frac{I_C}{I_E} \quad (2.3)$$

και δεν μπορεί να γίνει ίσος με τη μονάδα εξαιτίας της αύξησης του ρεύματος βάσης χωρίς την αντίστοιχη του ρεύματος συλλέκτη, κάτι που οφείλεται στις ελεύθερες οπές που έχουν απομείνει στη βάση. Για να μπορεί να ελεγχθεί το ρεύμα συλλέκτη μέσω του ρεύματος βάσης, αρκεί να αυξήσουμε την τάση διόδου εκπομπού. Με αυτό τον τρόπο θα προκληθεί αύξηση του ρεύματος βάσης, που θα προκαλέσει αναλογική αύξηση του ρεύματος συλλέκτη. Η συσχέτιση μεταξύ του μικρού ρεύματος βάσης και του μεγάλου ρεύματος συλλέκτη γίνεται με τον ορισμό του συντελεστή β_{dc} που είναι γνωστός ως κέρδος ή απολαβή ρεύματος και μας δίνει τον τύπο:

$$\beta_{dc} = \frac{I_C}{I_B} \quad (2.4)$$

Για τρανζίστορ χαμηλής ισχύος, το β είναι περίπου από 100 μέχρι 300, ενώ για τα αντίστοιχα υψηλής ισχύος από 20 μέχρι 100. Με τη βοήθεια του παραπάνω τύπου μπορούμε να σχηματίσουμε τις εξισώσεις με τις οποίες μπορούμε να πάρουμε το ρεύμα συλλέκτη και το ρεύμα βάσης. Αυτές είναι οι εξής:

$$I_C = \beta_{dc} I_B \quad (2.5)$$

για το ρεύμα συλλέκτη και:

$$I_B = \frac{I_C}{\beta_{dc}} \quad (2.6)$$

για το ρεύμα βάσης

2.2.4 Περιοχές λειτουργίας

Ένα τρανζίστορ μπορεί να λειτουργήσει είτε ως ενισχυτής, είτε ως διακόπτης. Ως ενισχυτής λειτουργεί στην περιοχή που ονομάζεται *ενεργός* ενώ ως διακόπτης στις περιοχές που ονομάζονται *κορεσμός* και *αποκοπή*. Υπάρχει και μία ακόμη περιοχή, η περιοχή κατάρρευσης, στην οποία το τρανζίστορ δεν μπορεί να λειτουργήσει και συνεπώς καταστρέφεται [28].

- **Ενεργός περιοχή:** Είναι η περιοχή στην οποία το τρανζίστορ επιτελεί την κανονική του λειτουργία ως ενισχυτής. Στη συγκεκριμένη περιοχή, η δίοδος βάσης – εκπομπού είναι πολωμένη ορθά, ενώ η δίοδος βάσης – συλλέκτη πολωμένη ανάστροφα. Το ρεύμα συλλέκτη καθορίζεται μόνο από το ρεύμα βάσης, καθώς ο συλλέκτης παίρνει όλα τα ηλεκτρόνια που έχει στείλει ο εκπομπός στη βάση. Διέπεται από κάποιες εξισώσεις, των οποίων η χρήση, όταν αποδεικνύεται η λειτουργία του τρανζίστορ στην ενεργό περιοχή, μας χρησιμεύουν προκειμένου να υπολογίσουμε τα ρεύματα του τρανζίστορ. Αυτές είναι οι εξής:

$$i_B = \frac{i_C}{\beta_{dc}} \quad (1.7)$$

για το ρεύμα βάσης,

$$i_C = \beta_{dc} I_B \quad (2.8)$$

για το ρεύμα συλλέκτη και

$$i_E = i_C + i_B \quad (2.9)$$

για το ρεύμα εκπομπού

Επειδή το ρεύμα βάσης θεωρείται σχεδόν αμελητέο, προκύπτει η σχέση:

$$i_E \cong i_C \quad (2.10)$$

- **Περιοχή κορεσμού:** Σε αυτή την περιοχή το τρανζίστορ έχει διακοπτική λειτουργία. Οι δίοδοι εκπομπού – βάσης και βάσης – συλλέκτη είναι ορθά πολωμένες. Επειδή η τάση στη δίοδο βάσης – συλλέκτη δεν επαρκεί ώστε να μπορέσει να πάρει ο συλλέκτης τα ηλεκτρόνια από τη βάση, ονομάζεται και περιοχή μικρών τάσεων και το ρεύμα συλλέκτη μπορεί να αυξηθεί πολύ γρήγορα. Εντούτοις, μεταξύ συλλέκτη και εκπομπού, μπορούμε να πούμε πως επικρατεί ένα θεωρητικό βραχυκύκλωμα, το τρανζίστορ δηλαδή λειτουργεί ως κλειστός διακόπτης. Ο συντελεστής β_{dc} , για τον οποίο έγινε αναφορά και ωστόσο, είναι πάντα μεγαλύτερος από τον συντελεστή $\beta_{dc(sat)}$, τον οποίο συναντάμε στην περιοχή κορεσμού. Το $\beta_{dc(sat)}$ μας δείχνει πόσο έντονος είναι ο κορεσμός. Όσο μικρότερο είναι, τόσο εντονότερα βρισκόμαστε στον κορεσμό, ενώ όσο μεγαλώνει, τόσο απομακρυνόμαστε. Η εξίσωση που σχηματίζεται στην περιοχή κορεσμού προκύπτει από τις μικρές τάσεις που επικρατούν σε αυτήν ($V_{CE} = 0 - 1V$) και πιο συγκεκριμένα:

$$V_{CE} = V_{CE(SAT)} \cong 0.3V \quad (2.11)$$

- **Περιοχή αποκοπής:** Όπως και στην περιοχή κορεσμού, έτσι και σε αυτή της αποκοπής, το τρανζίστορ λειτουργεί ως διακόπτης. Ένα τρανζίστορ βρίσκεται σε αποκοπή όταν δεν διαρρέεται από ρεύμα (θεωρητικά υπάρχει ένα πολύ μικρό ρεύμα συλλέκτη, της τάξης των

μΑ αλλά θεωρείται αμελητέο). Προφανώς από τη στιγμή που δεν υπάρχει ρεύμα, οι δίοδοι εκπομπού – βάσης και βάσης – συλλέκτη είναι πολωμένες ανάστροφα. Αυτό οδηγεί στην εξαγωγή του συμπεράσματος ότι το τρανζίστορ λειτουργεί ως ανοιχτός διακόπτης, όπερ σημαίνει ότι επειδή δεν θα υπάρξει πτώση τάσης στην αντίσταση R_C του συλλέκτη, όλη η τάση της πηγής V_{CC} θα πάει στον συλλέκτη. Οπότε θα έχουμε:

$$V_C = V_{CC} \quad (2.12)$$

2.3 Τελεστικοί ενισχυτές

Οι τελεστικοί ενισχυτές αποτελούν το κύριο κομμάτι του μαθήματος ‘Ηλεκτρονικές διατάξεις’ του προγράμματος σπουδών και γι’ αυτό κρίθηκε σκόπιμο να ενσωματωθούν στην εφαρμογή οι 4 βασικές συνδεσμολογίες τους που είναι:

- Αναστρέφουσα συνδεσμολογία
- Μη – αναστρέφουσα συνδεσμολογία
- Αθροιστικός ΤΕ
- ΤΕ διαφορών

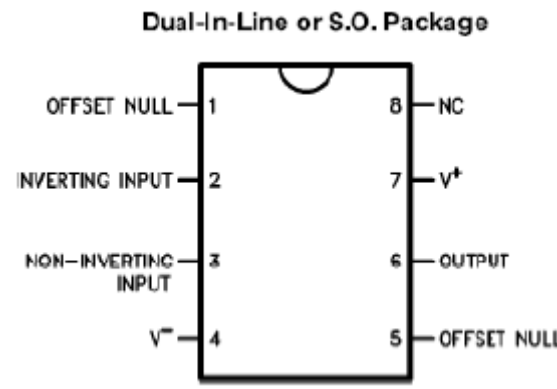
Αν θέλαμε να δώσουμε έναν ορισμό για τον τελεστικό ενισχυτή, θα μπορούσαμε να πούμε ότι πρόκειται για μια ηλεκτρονική διάταξη αποτελούμενη από επιμέρους κυκλώματα. Είναι σχεδιασμένος έτσι ώστε, όταν στα άκρα του συνδεθούν στοιχεία όπως αντιστάσεις και πυκνωτές, να επιτελεί μαθηματικές λειτουργίες όπως: πρόσθεση, αφαίρεση, πολλαπλασιασμό, διαίρεση, διαφόριση και ολοκλήρωση. Άλλωστε, από την ικανότητα τέλεσης αυτών των λειτουργιών προκύπτει και το όνομά του [27].

2.3.1 Ιστορικά στοιχεία

Ο όρος *τελεστικός ενισχυτής* έκανε για πρώτη φορά την εμφάνισή του κατά τη διάρκεια του Β’ Παγκοσμίου πολέμου. Αξιοσημείωτο είναι να αναφέρουμε ότι το πρώτο μεγάλο γεγονός, που υπήρξε ορόσημο στην ανάπτυξη της επιστήμης της ηλεκτρονικής, ήταν η εφεύρεση του διπολικού τρανζίστορ το 1947, το οποίο αποτέλεσε πολύ σημαντικό στοιχείο στη εξέλιξη της δομής του τελεστικού ενισχυτή, καθώς μέχρι τότε οι ηλεκτρονικές λυχνίες, που ήταν υπεύθυνες για τον έλεγχο του ηλεκτρικού ρεύματος, άρχισαν να αντικαθίστανται σταδιακά από τα τρανζίστορ, κάτι που συνέβαλε στην εξοικονόμηση ενέργειας και χώρου, καθώς οι τελεστικοί ενισχυτές είχαν πλέον μικρότερες φυσικές διαστάσεις. Κατόπιν, το 1965, παρουσιάστηκε ο πρώτος μονολιθικός τελεστικός ενισχυτής. Πρόκειται για τον $\mu A709$, ο οποίος σχεδιάστηκε από τον B. Widlar και κατασκευάστηκε από την Fairchild Semiconductor. Ακολούθησε ο $\mu A741$, ο οποίος ήρθε να αντισταθμίσει τα μειονεκτήματα του $\mu A709$. Λόγω των ίδιων προδιαγραφών, οι διάφορες παραλλαγές του 741 που δημιουργήθηκαν στη συνέχεια ήταν ισοδύναμες με τον $\mu A741$, οπότε το 741 καθιερώθηκε ως το ευρέως διαδεδομένο βιομηχανικό πρότυπο τελεστικού ενισχυτή. Το μεγάλο πλεονέκτημα το οποίο έχουν τα σύγχρονα κυκλώματα με ΤΕ έγκειται στην ανεξαρτησία των ιδιαίτερων χαρακτηριστικών του εκάστοτε χρησιμοποιούμενου ΤΕ και στην εξάρτηση του μόνο από εξωτερικά στοιχεία (π.χ. αντιστάσεις) [7].

2.3.2 Δομή ΤΕ

Στην εικόνα 2.7 παρουσιάζεται το διάγραμμα ακροδεκτών του ΤΕ 741

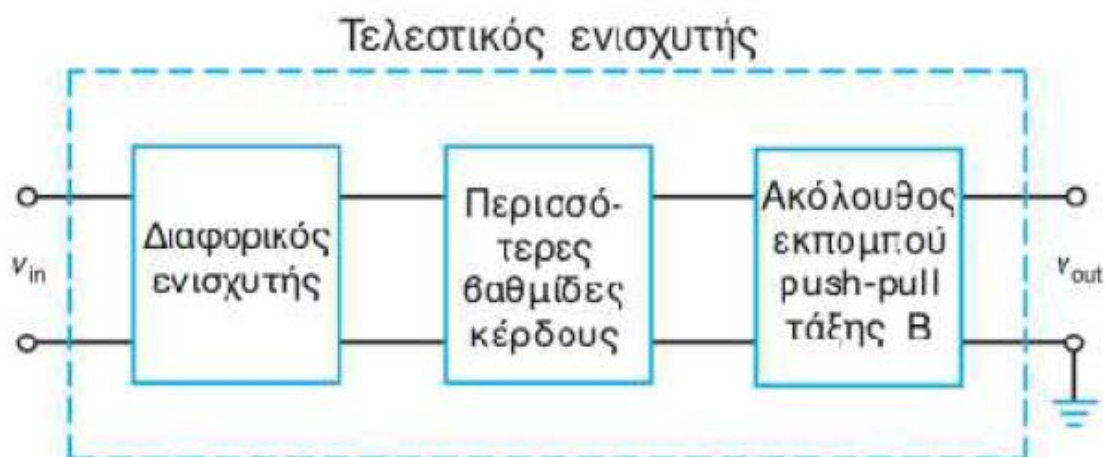


Εικόνα 2.7 Σχηματικό διάγραμμα του 741

Όπως βλέπουμε, κάθε ακροδέκτης έχει συγκεκριμένο ρόλο. Κατά σειρά λοιπόν, οι ακροδέκτες από τον 1 μέχρι τον 8 χρησιμοποιούνται για τα εξής:

- 1: Ακροδέκτης εξισορρόπησης ή μηδενισμός μετατόπισης
- 2: Ακροδέκτης αναστρέφουσας εισόδου
- 3: Ακροδέκτης μη – αναστρέφουσας εισόδου
- 4: Ακροδέκτης αρνητικής τροφοδοσίας
- 5: Ακροδέκτης εξισορρόπησης ή μηδενισμός μετατόπισης
- 6: Ακροδέκτης εξόδου
- 7: Ακροδέκτης θετικής τροφοδοσίας
- 8: Δεν χρησιμοποιείται

Στην εικόνα 2.8 φαίνεται η εσωτερική δομή του ΤΕ.

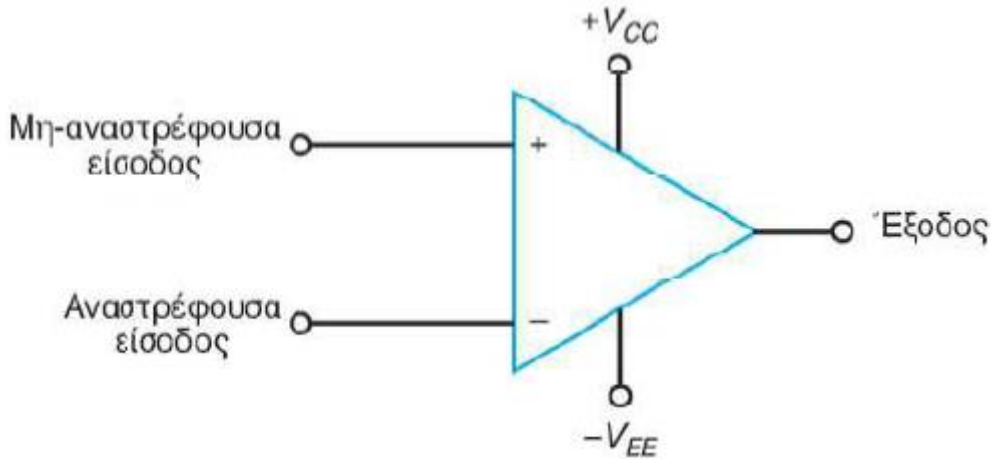


Εικόνα 2.8 Εσωτερικό διάγραμμα ΤΕ

Όπως έχει ήδη αναφερθεί, ο τελεστικός ενισχυτής αποτελείται από υποκυκλώματα. Στην είσοδο του έχει έναν διαφορικό ενισχυτή, του οποίου ο ρόλος είναι να καθορίζει τα χαρακτηριστικά εισόδου του ΤΕ. Ακολουθούν διάφορες βαθμίδες ενίσχυσης, οι οποίες περιλαμβάνουν στοιχεία όπως τρανζίστορ, πυκνωτές, καθρέφτες ρεύματος κλπ. Τέλος στην έξοδό του συναντάμε τον ενισχυτή push – pull τάξης Β. Με τη χρήση του η έξοδος του ΤΕ είναι ασύμμετρη, δηλαδή αποτελείται από ένα άκρο. Υπάρχουν

και ενισχυτές οι οποίοι δεν χρησιμοποιούν για έξοδο τον push – pull τάξης Β και έχουν συμμετρική έξοδο (δύο άκρα δηλαδή).

Στην εικόνα 2.9 απεικονίζεται το σύμβολο του ΤΕ, του οποίου γίνεται χρήση στα κυκλώματα που υλοποιήθηκαν στην εφαρμογή και θα αναλυθούν παρακάτω.

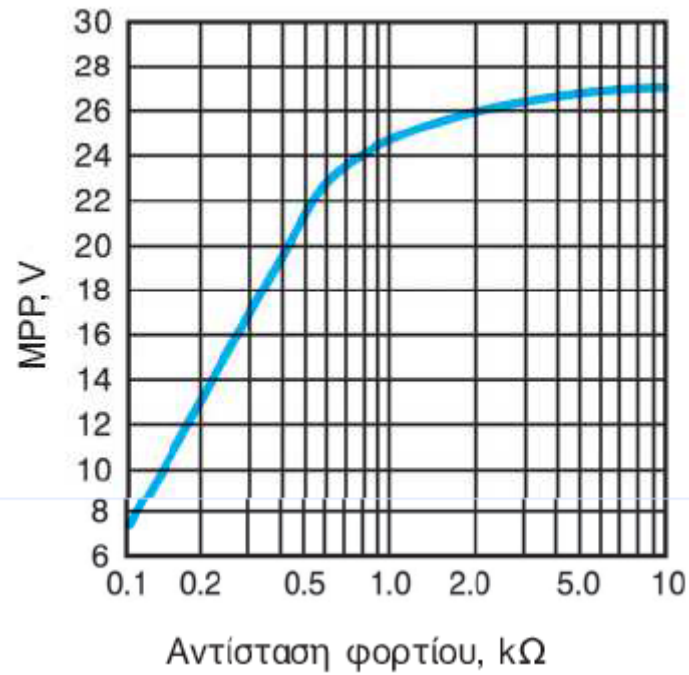


Εικόνα 2.9 Σχηματικό σύμβολο ενός ΤΕ

Αποτελείται από μία αναστρέφουσα είσοδο (ο ακροδέκτης που είναι συνδεδεμένος στο -), μία μη – αναστρέφουσα είσοδο (ο ακροδέκτης που είναι συνδεδεμένος στο +), τους ακροδέκτες τροφοδοσίας και την έξοδο. Σκοπός μας είναι να υπολογίσουμε το κέρδος του ενισχυτή, σύμφωνα με το οποίο θα ενισχύσουμε το σήμα εισόδου, είτε στην ίδια, είτε στην αντίστροφη φάση.

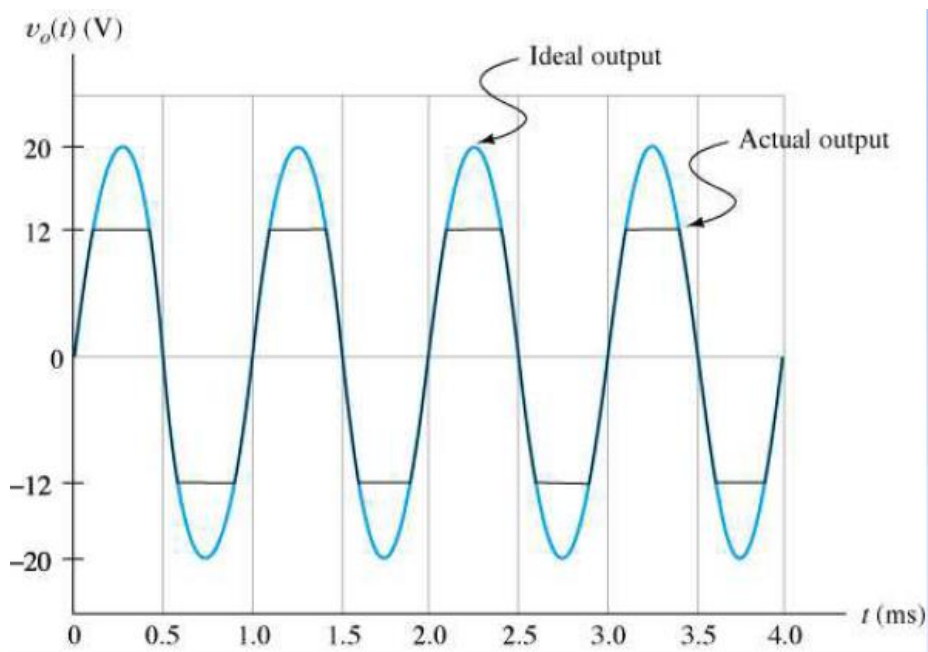
2.3.3 Κορεσμός εξόδου (MPP)

Μία βασική λειτουργία που περιλαμβάνεται στα κυκλώματα των ΤΕ της εφαρμογής που παρουσιάζουν διαγραμματική απεικόνιση (κυκλώματα αναστρέφοντος και μη – αναστρέφοντος ΤΕ) είναι η εμφάνιση της μέγιστης από κορυφή σε κορυφή εξόδου του ενισχυτή με και χωρίς ψαλιδισμό. Όπως είναι λογικό, επειδή οι ΤΕ που χρησιμοποιούνται στην εφαρμογή δεν είναι ιδανικοί, δεν έχουν δηλαδή τα χαρακτηριστικά ενός ιδανικού ΤΕ, οι τιμές των παραμέτρων τους μπορούν να αλλάζουν από τον χρήστη. Έτσι λοιπόν, η έξοδος δεν θα μπορέσει ποτέ να φτάσει την τιμή των τάσεων τροφοδοσίας σε κανονική λειτουργία. Παρακάτω βλέπουμε μια εικόνα η οποία δείχνει την μέγιστη έξοδο p-p συναρτήσει της αντίστασης φορτίου (σε kΩ).



Εικόνα 2.10 Γραφική παράσταση για την MPP

Αν έχουμε για παράδειγμα φορτίο $5k\Omega$ η MPP θα είναι περίπου $26V$, το οποίο σημαίνει ότι η έξοδος κυμαίνεται από $+13$ μέχρι $-13V$. Άρα για τάσεις τροφοδοσίας $+15V$ και $-15V$ θα έχουμε $2V$ διαφορά κορεσμού εξόδου. Η απεικόνιση σε διάγραμμα $v - t$ ήταν κάπως έτσι:



Εικόνα 2.11 Ιδανική και πραγματική έξοδος ΤΕ σε διάγραμμα $V - t$

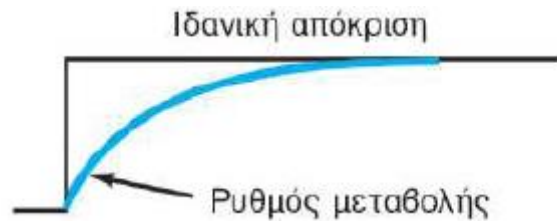
Στην περίπτωση δηλαδή που είχαμε μια έξοδο της τάξης των $+20V$ με $-20V$, λόγω του κορεσμού εξόδου και της τροφοδοσίας των ΤΕ, θα πήγαινε μέχρι τα $+12V$ και $-12V$ και το σήμα θα ψαλιδιζόταν σε εκείνο το σημείο.

2.3.4 Ρυθμός μεταβολής (slew rate)

Ο ρυθμός μεταβολής ορίζεται ως η μεταβολή της τάσης εξόδου προς τη μεταβολή του χρόνου και συμβολίζεται ως εξής:

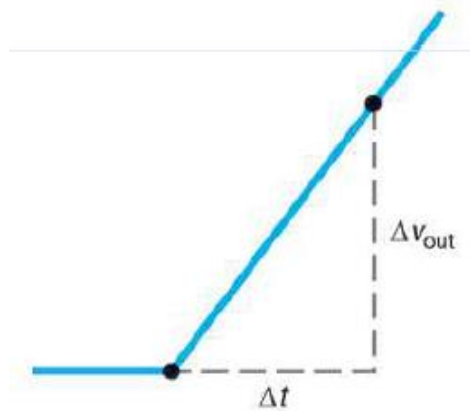
$$S_R = \frac{\Delta v_{out}}{\Delta t} \quad (2.13)$$

Για να μπορέσει να γίνει κατανοητό το πρόβλημα που προκύπτει στην έξοδο του τελεστικού λόγω του ρυθμού μεταβολής, κρίνεται σκόπιμο να παρουσιαστεί η παρακάτω εικόνα:



Εικόνα 2.12 Πρόβλημα λόγω ρυθμού μεταβολής

Μέσα στον τελεστικό ενισχυτή υπάρχει ένας πυκνωτής. Ο ρόλος αυτού του πυκνωτή είναι να αντισταθμίζει τη συχνότητα. Το πρόβλημα είναι ότι θα πρέπει συνεχώς να φορτίζεται και να αποφορτίζεται. Όσο μεγαλύτερη είναι η συχνότητα, τόσο πιο γρήγορα είναι αναγκασμένη να αποκρίνεται η έξοδος. Αυτό το πρόβλημα απεικονίζεται στην παραπάνω εικόνα όπου, ενώ θα θέλαμε να πάρουμε ως ιδανική έξοδο τον τετραγωνικό παλμό, λόγω του ρυθμού μεταβολής, παίρνουμε την καμπυλοειδή μορφή που βλέπουμε. Σχηματικά, ο ρυθμός μεταβολής ορίζεται ως εξής:



Εικόνα 2.13 Ρυθμός μεταβολής τάσης εξόδου

Έτσι λοιπόν, η κλίση ισούται με την κατακόρυφη μεταβολή της τάσης προς την οριζόντια μεταβολή του χρόνου μεταξύ δύο σημείων.

Στην εφαρμογή, έγινε προσπάθεια για όσο το δυνατόν πιο ρεαλιστική απεικόνιση του παλμού εισόδου και εξόδου με επίδραση του ρυθμού μεταβολής, παρατηρώντας τα αποτελέσματα από την εφαρμογή του SR στον παλμό τάσης εξόδου. Παρόλο που τα φυλλάδια προδιαγραφών ενός τελεστικού ενισχυτή δίνουν το ρυθμό μεταβολής ανάλογα με το μοντέλο TE που χρησιμοποιείται, κρίθηκε σκόπιμο στην εφαρμογή να μπορεί ο χρήστης να επιλέξει το SR ανάμεσα σε κάποιες συγκεκριμένες τιμές για να μπορέσει να κατανοήσει καλύτερα την επίδρασή του.

2.3.5 Βασικές συνδεσμολογίες ΤΕ

Οι παρακάτω συνδεσμολογίες ΤΕ είναι και αυτές στις οποίες δίνεται η μεγαλύτερη βαρύτητα στο μάθημα ‘Ηλεκτρονικές διατάξεις’ του προγράμματος σπουδών του τμήματος. Προφανώς επιλέχθηκαν για να αποτελέσουν κίνητρο για τη μεθοδικότερη μελέτη των φοιτητών και του πλήθους των ενδιαφερόμενων, αλλά και επειδή πρόκειται για κυκλώματα των οποίων η χρήση είναι ευρέως διαδεδομένη και αρκετά συνηθισμένη, λόγω της τέλεσης των προσδοκώμενων λειτουργιών με πιο γρήγορο και αποτελεσματικό τρόπο από άλλα κυκλώματα. Έτσι λοιπόν, τα υπό μελέτη κυκλώματα είναι τα εξής:

2.3.5.1 Αναστρέφων ενισχυτής

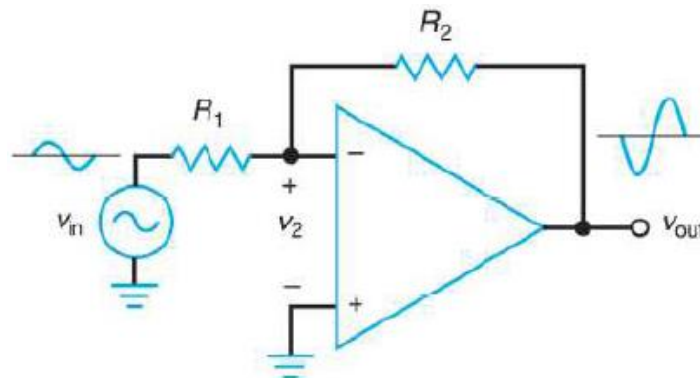
2.3.5.1.1 Ορισμός και κέρδος τάσης

Ο αναστρέφων ενισχυτής αποτελεί το πιο βασικό από τα κυκλώματα ενίσχυσης. Ο ρόλος του είναι να αντιστρέφει την πολικότητα του σήματος εισόδου και ταυτόχρονα να το ενισχύει. [27]. Για να το κάνει αυτό πρέπει να πάρει το σήμα εισόδου και μέσω κάποιου κέρδους τάσης να το παρουσιάσει ενισχυμένο και αντεστραμμένο στην έξοδο. Υπάρχουν δύο είδη κέρδους:

1. Το κέρδος ανοιχτού βρόχου ($A_{V_{OL}}$)
2. Το κέρδος κλειστού βρόχου ($A_{V_{CL}}$)

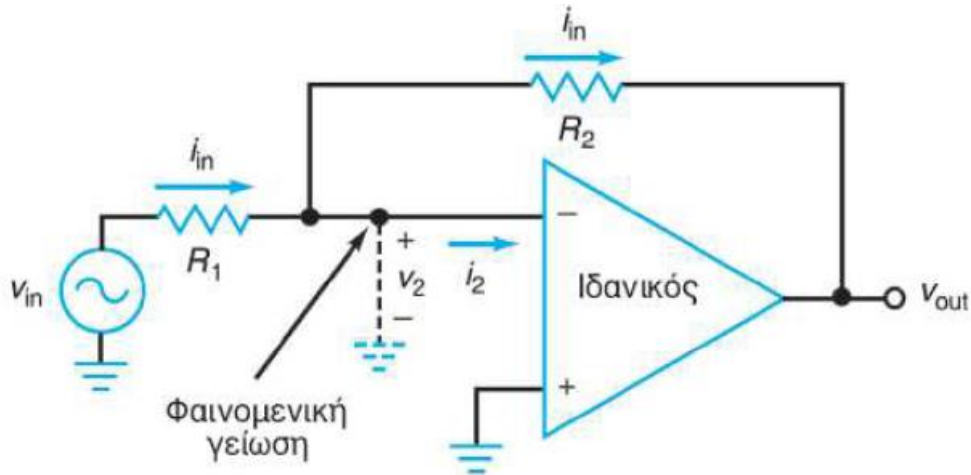
Αυτό που έχει αξία να μελετήσουμε στην εφαρμογή είναι το κέρδος κλειστού βρόχου γιατί το κέρδος ανοιχτού βρόχου παρουσιάζει αστάθεια και μπορεί να έχει πολύ μεγάλες τιμές, κάτι που δεν θα προσέδιδε νόημα στον ενδιαφερόμενο, ούτε θα τον βοηθούσε να βγάλει συμπεράσματα.

Στην παρακάτω εικόνα φαίνεται η συνδεσμολογία του αναστρέφοντος ενισχυτή:



Εικόνα 2.14 Αναστρέφων ενισχυτής

Όπως βλέπουμε, η πηγή συνδέεται στην αναστρέφουσα είσοδο του τελεστικού (-), μέσω μιας αντίστασης R_1 , ενώ η μη – αναστρέφουσα είσοδος (+) είναι συνδεδεμένη στη γείωση. Υπάρχει επίσης μια αντίσταση R_2 (ή R_F), η οποία είναι αυτή που θα ανατροφοδοτήσει το σήμα της τάσης εξόδου και πάλι στην είσοδο. Αυτό που θέλουμε είναι να μπορέσουμε να υπολογίσουμε το κέρδος κλειστού βρόχου για να πάρουμε το τελικό μας σήμα.

Εικόνα 2.15 Ρεύμα στις αντιστάσεις R_1 και R_f

Για να καταλήξουμε στο ζητούμενο, είναι θεμιτό να γίνει μια ανάλυση στα στοιχεία του κυκλώματος. Ένα ρεύμα i_{in} ξεκινάει από την πηγή και διαρρέει την αντίσταση R_1 . Από το νόμο του Ohm προκύπτει ότι:

$$i_{in1} = \frac{v_{in}}{R_1} \quad (2.14)$$

όπου v_{in} η τάση της πηγής. Το ρεύμα αυτό θα διασπαστεί και θα κατευθυνθεί προς την αντίσταση R_f και στην αναστρέφουσα είσοδο του τελεστικού. Είτε στην περίπτωση που χρησιμοποιούμε έναν ιδανικό τελεστικό ενισχυτή (τελεστικός με ιδανικά χαρακτηριστικά) είτε οποιονδήποτε άλλον, διαπιστώνεται το φαινόμενο του *εικονικού βραχυκυκλώματος* ανάμεσα στις δύο εισόδους. Αυτό σημαίνει ότι επειδή η μη – αναστρέφουσα είσοδος είναι συνδεδεμένη στη γείωση, μια *φαινομενική γείωση* δημιουργείται στο σημείο πριν την αναστρέφουσα είσοδο. Το αποτέλεσμα είναι η αναστρέφουσα είσοδος να λειτουργεί σαν γείωση για την τάση αλλά σαν ανοιχτό κύκλωμα για το ρεύμα. Οπότε, το κομμάτι του ρεύματος που θα κατευθυνθεί προς την είσοδο του τελεστικού θεωρείται σχεδόν αμελητέο. Σε αυτό συμβάλλει και η πολύ μεγάλη αντίσταση εισόδου R_{in} του τελεστικού, καθώς δεν επιτρέπει στο μεγαλύτερο ποσοστό του ρεύματος να εισέλθει στον ενισχυτή. Άρα, όλο το ρεύμα που έχει ξεκινήσει από την πηγή και διαρρέει την αντίσταση R_1 θα κατευθυνθεί προς την R_f , συνεπώς η εξίσωση που θα προκύψει είναι η εξής:

$$i_{in2} = i_{in1} = \frac{v_{in}}{R_1} \quad (2.15)$$

δηλαδή το ρεύμα που διαρρέει την R_f θα είναι ίσο με το i_{in1} λόγω της μηδενικής τάσης v_2 και του εικονικού βραχυκυκλώματος ανάμεσα στις εισόδους, καθώς και του μηδενικού ρεύματος εισόδου του τελεστικού. Για να βρούμε την έξοδο θα πρέπει να ξεκινήσουμε να μετράμε την τάση από ένα σημείο στο οποίο είναι γνωστή. Έτσι λοιπόν, στο σημείο που παρουσιάζεται η φαινομενική γείωση η τάση είναι 0 και επειδή το ρεύμα ακολουθεί ίδια φορά με αυτή που διατρέχεται το κύκλωμα, αφαιρούμε την πτώση τάσης πάνω στην αντίσταση R_2 . Με αυτά προκύπτει ο παρακάτω τύπος:

$$v_{out} = 0 - \frac{v_1}{R_1} R_2 \quad (2.16)$$

από τον οποίο συνεπάγεται:

$$v_{out} = -\frac{R_2}{R_1} v_1 \quad (2.17)$$

Οπότε το κέρδος τάσης του συγκεκριμένου κυκλώματος θα είναι:

$$A_{v(CL)} = -\frac{R_2}{R_1} \quad (2.18)$$

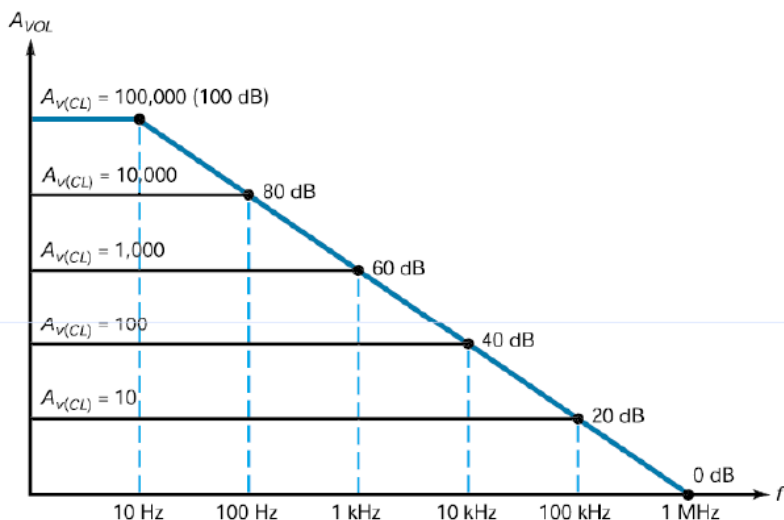
Το (-) στον τύπο αφορά την αντιστροφή φάσης 180° . Το αξιοσημείωτο συμπέρασμα που προκύπτει είναι ότι το κέρδος τάσης εξαρτάται μόνο από τις αντιστάσεις και όχι από εξωτερικές μεταβολές του κυκλώματος.

2.3.5.1.2 Εύρος ζώνης

Εκτός από τον ορισμό του κέρδους τάσης, μέσω του οποίου γίνεται ο υπολογισμός του σήματος εξόδου του ενισχυτή, αξίζει να μελετηθεί και η έννοια του εύρους ζώνης συχνοτήτων. Όπως θα αναλυθεί και στα επόμενα κεφάλαια που αφορούν τη λειτουργία της εφαρμογής, ο χρήστης, πειραματιζόμενος με τις παραμέτρους του κυκλώματος, θα μπορεί να δει στα αποτελέσματα και το f_{CL} , το εύρος ζώνης κλειστού βρόχου δηλαδή. Ορίζεται από τον παρακάτω τύπο:

$$f_{CL} = \frac{f_{unity}}{A_{v(CL)}} \quad (2.19)$$

όπου f_{unity} η συχνότητα μοναδιαίου κέρδους, η οποία στο μοντέλο τελεστικού LM741 έχει τιμή 1MHz και είναι αυτή που χρησιμοποιείται στην εφαρμογή, καθώς είναι η πιο συχνά χρησιμοποιούμενη έκδοση. Αυτό που μπορεί να διαπιστώσει κανείς βλέποντας και τα αποτελέσματα που προκύπτουν από τη χρήση της εφαρμογής είναι ότι όσο μειώνεται το κέρδος τάσης, τόσο αυξάνεται το εύρος ζώνης συχνοτήτων. Το γεγονός αυτό αποτελεί ένα από τα μειονεκτήματα των τελεστικών ενισχυτών γενικότερα, καθώς δεν τα πάνε καλά με τη συχνότητα.



Εικόνα 2.16 Όσο μεγαλώνει η συχνότητα μικραίνει το κέρδος

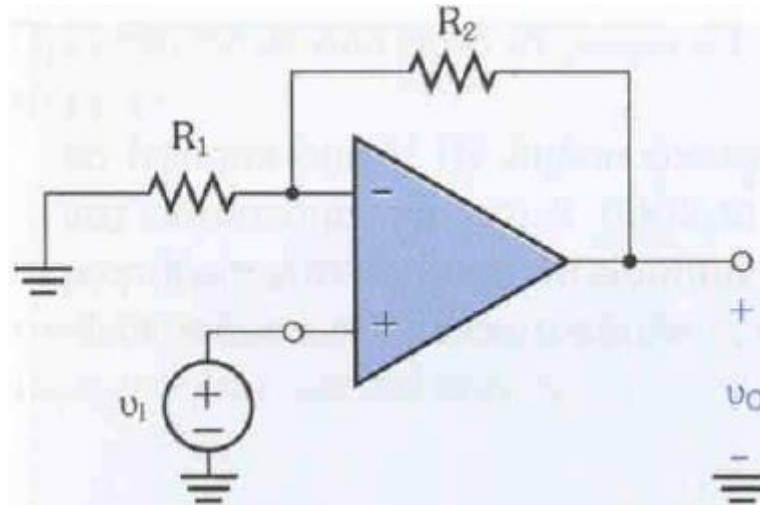
Όπως λοιπόν παρατηρούμε, το κέρδος μειώνεται κατά 20dB ανά δεκάδα συχνοτήτων.

2.3.5.2 Μη - Αναστρέφων ενισχυτής

2.3.5.2.1 Ορισμός και κέρδος τάσης

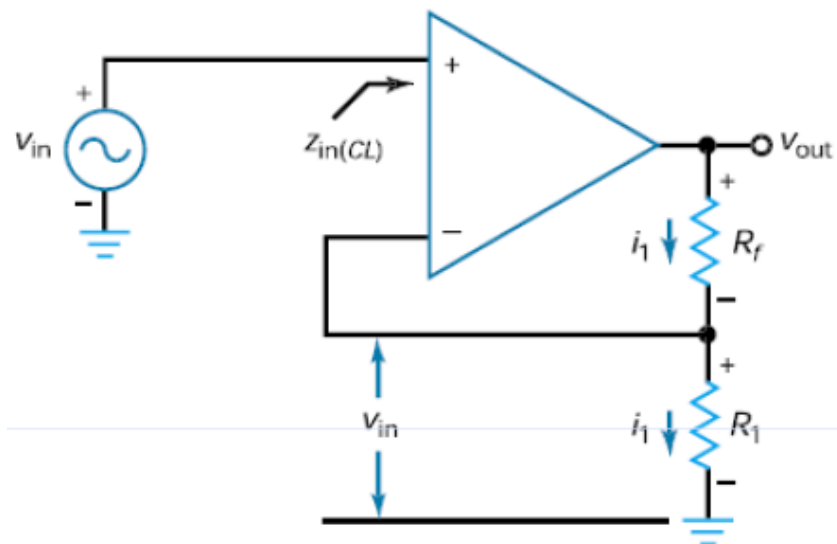
Το δεύτερο από τα βασικά κυκλώματα των τελεστικών ενισχυτών που περιλαμβάνει η εφαρμογή είναι ο μη - αναστρέφων ενισχυτής. Η συγκεκριμένη μέθοδος ενίσχυσης αφορά μια συνδεσμολογία η οποία είναι σχεδιασμένη να αποδίδει θετικό κέρδος τάσης [27]. Εδώ πέρα, το σήμα το οποίο ξεκινάει από την πηγή και πρόκειται να ενισχυθεί, εμφανίζεται στην έξοδο ενισχυμένο (όπως και στον αναστρέφων

ενισχυτή) αλλά στην ίδια φάση με το σήμα εισόδου της πηγής. Η συνδεσμολογία του μη – αναστρέφοντος ενισχυτή φαίνεται στην παρακάτω εικόνα:



Εικόνα 2.17 Μη – αναστρέφων ενισχυτής

Αντίθετα με τον αναστρέφων, η πηγή εισόδου είναι συνδεδεμένη στον θετικό ακροδέκτη του ενισχυτή. Και εδώ έχουμε αντίσταση ανάδρασης R_2 , της οποίας ο ρόλος είναι η ανατροφοδότηση ενός μέρους της τάσης του σήματος εξόδου πίσω στην είσοδο, μέσω του διαιρέτη τάσης που δημιουργείται από την συγκεκριμένη αντίσταση και την R_1 , η οποία είναι συνδεδεμένη στη γείωση. Όπως σε κάθε τέτοιου είδους κύκλωμα, έτσι κι εδώ, σκοπός είναι να σχηματιστεί μια σχέση ανάμεσα στην είσοδο και την έξοδο μέσω του κέρδους τάσης που προκύπτει. Η συνδεσμολογία του μη – αναστρέφοντος ενισχυτή μπορεί να παρουσιαστεί και ως εξής:



Εικόνα 2.18 Ισοδύναμο κύκλωμα μη – αναστρέφοντος ενισχυτή

Το μεγάλο πλεονέκτημα που έχει η συγκεκριμένη διάταξη αφορά την πολύ μεγάλη αντίσταση εισόδου που έχει, η οποία είναι περίπου ίση με την πολύ μεγάλη αντίσταση εισόδου του ΤΕ. Ξεκινώντας την ανάλυση από την πηγή, το ρεύμα που εισέρχεται στον τελεστικό είναι σχεδόν αμελητέο, άρα το μοναδικό ρεύμα που θα κυκλοφορεί θα είναι το i_1 , το οποίο θα διαρρέει και την αντίσταση ανάδρασης R_f και την R_1 . Λόγω του εικονικού βραχυκυκλώματος που συμβαίνει, η τάση της πηγής v_{in} θα πάει στα άκρα της R_1 . Συνεπώς, χάρη στον νόμο του Ohm, για την αντίσταση R_1 προκύπτει η σχέση:

$$v_{in} = i_1 R_1 \quad (2.20)$$

Αφού το εικονικό βραχυκύκλωμα λειτουργεί ως ανοιχτό κύκλωμα, το ίδιο ρεύμα i_1 θα ρέει και μέσω της R_f , όπως προαναφέρθηκε. Έτσι, η σχέση που σχηματίζεται για την τάση εξόδου είναι:

$$v_{out} = i_1 (R_f + R_1) \quad (2.21)$$

Οπότε, το κέρδος τάσης της συγκεκριμένης συνδεσμολογίας θα είναι το εξής:

$$A_{v(CL)} = \frac{R_f + R_1}{R_1} \quad (2.22)$$

από όπου προκύπτει:

$$A_v = \frac{R_f}{R_1} + 1 \quad (2.23)$$

Όπως και στον αναστρέφων, έτσι κι εδώ, ο τύπος που σχηματίζεται για το κέρδος αναδεικνύει την εξάρτηση μόνο από τις αντιστάσεις και όχι από οποιοδήποτε εξωτερικό χαρακτηριστικό ή χαρακτηριστικό του τελεστικού ενισχυτή.

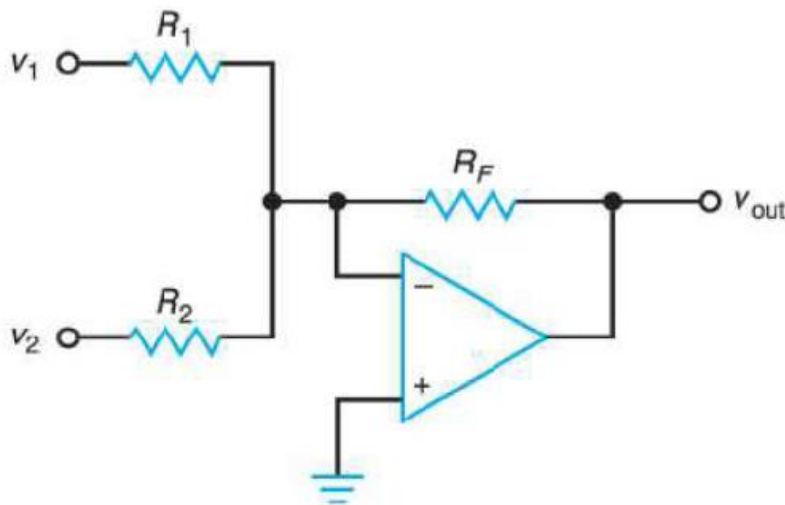
2.3.5.2.2 Εύρος ζώνης

Όσον αφορά το εύρος ζώνης συχνοτήτων του μη – αναστρέφοντος ενισχυτή, ισχύουν τα ίδια με τον αναστρέφων, όπως και για τη σχέση από την οποία προκύπτει, δηλαδή:

$$f_{CL} = \frac{f_{unity}}{A_{v(CL)}} \quad (2.24)$$

2.3.5.3 Αθροιστής

Το τρίτο από τα κυκλώματα που συνθέτουν το περιεχόμενο της σειράς κυκλωμάτων των τελεστικών ενισχυτών στην εφαρμογή είναι ο αθροιστικός ενισχυτής ή αθροιστής. Η λειτουργία του έγκειται στο να εκτελεί την πράξη της πρόσθεσης μεταξύ δύο ή περισσότερων αναλογικών σημάτων και να τα δίνει στην έξοδο [27]. Καθώς για την κατανόηση της λειτουργίας του αθροιστή δεν είναι απαραίτητη η χρήση παραπάνω από δύο εισόδων, η συνδεσμολογία, της οποίας γίνεται ανάλυση στην εφαρμογή, έχει μόνον δύο πηγές. Το σχήμα το οποίο μελετάται είναι το ακόλουθο:



Εικόνα 2.19 Αθροιστικός ενισχυτής

Όπως στον αναστρέφον, έτσι και στον αθροιστικό ενισχυτή η πηγή, ή καλύτερα το άθροισμα των πηγών για την συγκεκριμένη περίπτωση, συνδέεται στην αναστρέφουσα είσοδο του τελεστικού, ενώ η μη – αναστρέφουσα καταλήγει στη γείωση. Υπάρχει και εδώ η αντίσταση ανάδρασης R_f , η οποία συμβάλλει στον σχηματισμό της σχέσης για το κέρδος τάσης. Έχοντας ως δεδομένα τα σχεδόν μηδενικά ρεύματα εισόδων και το φαινομενικό βραχυκύκλωμα που ισχύουν σε κάθε κύκλωμα τελεστικού ενισχυτή, μπορούμε χάρη στο νόμο του Ohm να σχηματίσουμε την εξής εξίσωση για το ρεύμα:

$$i_{in} = i_1 + i_2 = \frac{v_1}{R_1} + \frac{v_2}{R_2} \quad (2.25)$$

Για να υπολογίσουμε την έξοδο ακολουθούμε τη μέθοδο κατά την οποία ξεκινάμε από ένα σημείο στο κύκλωμα στο οποίο γνωρίζουμε την τάση του μέχρι να καταλήξουμε στο σημείο που μας ενδιαφέρει. Έτσι λοιπόν αφού η τάση στο σημείο που βρίσκεται ο αριστερός ακροδέκτης της R_f είναι 0 και το ρεύμα το οποίο θα διαρρέει την R_f είναι το i_{in} θα έχουμε:

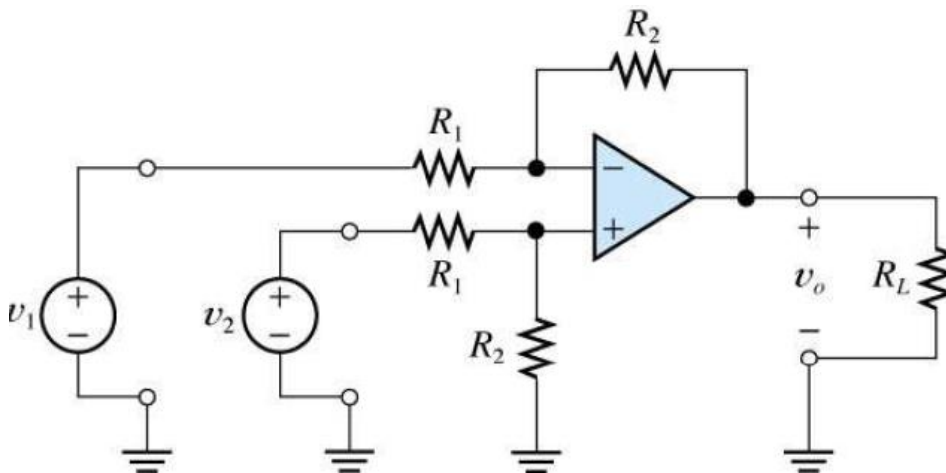
$$v_{out} = 0 - (i_1 + i_2)R_f = -\left(\frac{R_f}{R_1}v_1 + \frac{R_f}{R_2}v_2\right) \quad (2.26)$$

Άρα, οι τύποι που προκύπτουν για τα κέρδη κλειστού βρόχου είναι:

$$A_{v_1} = -\frac{R_f}{R_1} \quad \text{και} \quad A_{v_2} = -\frac{R_f}{R_2} \quad (2.27)$$

2.3.5.4 Ενισχυτής διαφορών

Το τελευταίο κύκλωμα που συνθέτει την ομάδα κυκλωμάτων των τελεστικών ενισχυτών της εφαρμογής είναι αυτό του ενισχυτή διαφορών. Η ανάλυση λειτουργίας του προέρχεται από το βιβλίο των Παναγιώτα Αθ. Παπαβραμίδου και Δημήτρη Κλ. Παπακώστα [26]. Αν και δεν ανήκει στις βασικές συνδεσμολογίες, είναι μια πολύ χρήσιμη εφαρμογή κυκλώματος που περιέχει τελεστικό ενισχυτή, γιατί μπορεί να ενισχύσει τη διαφορά των σημάτων των πηγών εισόδου, απορρίπτοντας, ωστόσο, τις κοινές συνιστώσες που υπάρχουν μεταξύ τους.



Εικόνα 2.20 Ενισχυτής διαφορών

Τα δεδομένα που έχουμε για την ανάλυση του κυκλώματος και την εύρεση του κέρδους και της τάσης εξόδου αφορούν την ύπαρξη δύο πηγών τάσης στην είσοδο, τεσσάρων αντιστάσεων, εκ των οποίων, μία αντίσταση R_1 η οποία συνδέεται στην πηγή v_1 , δύο αντιστάσεις R_1 και R_2 , οι οποίες σχετίζονται με την πηγή v_2 και μία αντίσταση ανάδρασης R_2 . Όπως φαίνεται κι από την εικόνα, οι αντιστάσεις R_1

θα έχουν ίδια τιμή. Το ίδιο ισχύει και για τις R_2 . Έχοντας λοιπόν και το δεδομένο ότι τα ρεύματα εισόδου του τελεστικού είναι σχεδόν μηδενικά και οι εισοδοί περίπου βραχυκυκλωμένες, για να καταλήξουμε στο ζητούμενο, γίνεται χρήση του θεωρήματος της υπέρθεσης, κατά το οποίο γειώνουμε εναλλάξ τις πηγές εισόδου και υπερθέτουμε το αποτέλεσμα. Έτσι, γειώνοντας αρχικά την πηγή v_2 και κρατώντας την v_1 , λόγω του παράλληλου συνδυασμού των αντιστάσεων R_1 και R_2 και του αμελητέου ρεύματος, η γείωση οδηγείται στην μη - αναστρέφουσα είσοδο. Οπότε αυτά που μένουν σχηματίζουν μια αναστρέφουσα συνδεσμολογία, από την οποία παίρνουμε:

$$V_{out1} = -\frac{R_2}{R_1}V_1 \quad (2.28)$$

Εκτελώντας την ίδια διαδικασία για την πηγή v_1 , προκύπτει ένας διαιρέτης τάσης στην μη - αναστρέφουσα είσοδο ο οποίος μας δίνει:

$$V_+ = \frac{R_2}{R_1+R_2}V_2 \quad (2.29)$$

και σε συνδυασμό με το κέρδος για την μη - αναστρέφουσα συνδεσμολογία θα πάρουμε:

$$V_{out2} = \left(1 + \frac{R_2}{R_1}\right)\frac{R_2}{R_1+R_2}V_2 \Rightarrow V_{out2} = \frac{R_2}{R_1}V_2 \quad (2.30)$$

Οπότε, με τη χρήση του θεωρήματος της υπέρθεσης έχουμε:

$$V_{out} = V_{out1} + V_{out2} = \frac{R_2}{R_1}(V_2 - V_1) \quad (2.31)$$

με κέρδος:

$$A_v = \frac{R_2}{R_1} \quad (2.32)$$

2.4 Επίλογος

Συνοψίζοντας, στο δεύτερο κεφάλαιο γίνεται μια βιβλιογραφική ανασκόπηση του απαιτούμενου θεωρητικού υποβάθρου των στοιχείων και διατάξεων που χρησιμοποιούνται στην εφαρμογή. Αρχικά, δίνονται οι απαραίτητες συστάσεις για τον ορισμό του τρανζίστορ, κάποια ιστορικά στοιχεία, η εξήγηση του εσωτερικού και της λειτουργίας του και ποια η λεπτομέρεια όταν γίνεται χρήση του όρου ‘διπολικό’. Καθότι στην εργασία επιλέχθηκε ο πιο ευρέως διαδεδομένος τύπος τρανζίστορ, δηλαδή το ηρη, αναλύεται η δομή του, η κίνηση του φορτίου στο εσωτερικό του και τις περιοχές του, καθώς και οι γενικές σχέσεις μεταξύ των ρευμάτων που ρέουν σε αυτό. Ακολουθεί η αναφορά στις περιοχές τις οποίες μπορεί να λειτουργήσει το τρανζίστορ και στις σχέσεις που ισχύουν σε κάθε μία, πράγματα που είναι σημαντικό να έχει καταλάβει ο αναγνώστης και χρήστης για να είναι σε θέση να κατανοήσει τα αποτελέσματα που προκύπτουν από τη χρήση της εφαρμογής, ιδιαίτερα στην περίπτωση που επιλέξει να πειραματιστεί με τα διπολικά τρανζίστορ. Στη συνέχεια παρουσιάζονται οι πληροφορίες που αφορούν τους τελεστικούς ενισχυτές. Όπως και στα τρανζίστορ, έτσι κι εδώ, καταγράφονται ορισμοί και ιστορικά στοιχεία, μαζί με τη βασική δομή και τη διαγραμματική απεικόνιση του ΤΕ. Το κεφάλαιο κλείνει με την ανάλυση των εννοιών που θα συναντήσει ο χρήστης της εφαρμογής (Κορεσμός εξόδου και ρυθμός μεταβολής), σε συνδυασμό με την περιγραφή των βασικών συνδεσμολογιών ΤΕ που χρησιμοποιούνται.

Κεφάλαιο 3ο: Ανάλυση λειτουργίας επιλεγμένων κυκλωμάτων

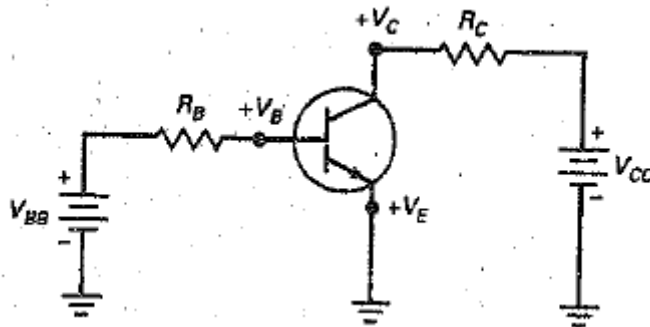
3.1 Εισαγωγή

Συνεχίζοντας την ανάλυση των κυκλωμάτων που επιλέχθηκαν για τις ανάγκες της εφαρμογής, στο τρίτο κεφάλαιο αναλύεται η λειτουργία τους με τη χρήση παραδειγμάτων. Μέσα από αυτά, γίνονται αντιληπτές οι καταστάσεις στις οποίες μπορεί να βρεθεί ένα κύκλωμα (π.χ. περιοχές λειτουργίας στα διπολικά τρανζίστορ). Όπως και στο προηγούμενο κεφάλαιο, τα παραδείγματα και οι πληροφορίες στηρίζονται στο βιβλίο των Albert Malvino και David J. Bates [25].

3.2 Διπολικά τρανζίστορ

3.2.1 Κύκλωμα πόλωσης βάσης

Το κύκλωμα πόλωσης βάσης είναι ένα κύκλωμα το οποίο αποτελείται από δύο πηγές τάσης, τη V_{BB} και τη V_{CC} , καθώς και από δύο αντιστάσεις, την αντίσταση βάσης R_B και την αντίσταση συλλέκτη R_C . Στην εικόνα που ακολουθεί φαίνεται η συνδεσμολογία του.



Εικόνα 3.1 Κύκλωμα πόλωσης βάσης

Για την επίλυσή του σκοπός είναι η εύρεση των τάσεων μεταξύ βάσης και γείωσης, μεταξύ συλλέκτη και γείωσης και μεταξύ εκπομπού και γείωσης, καθώς και των ρευμάτων που το διαρρέουν. Πιο συγκεκριμένα, χρειάζεται να βρεθούν οι τάσεις V_B , V_C και V_E , ενώ όσον αφορά τα ρεύματα υπολογίζονται τα αντίστοιχα I_B , I_C και I_E . Έστω λοιπόν ότι έχουμε ένα κύκλωμα με $V_{BB} = 4V$, $V_{CC} = 10V$, $R_B = 100k\Omega$, $R_C = 1k\Omega$ και συντελεστή κέρδους ρεύματος $\beta = 200$. Ξεκινώντας την ανάλυση, η τάση V_{BB} πολώνει ορθά τη δίοδο βάσης - εκπομπού μέσω της αντίστασης R_B . Οπότε θα έχει στα άκρα της $0.7V$. Το γεγονός αυτό σε συνδυασμό με την μηδενική τάση εκπομπού (καταλήγει στη γείωση) δίνει τη δυνατότητα για υπολογισμό της τάσης V_B , ο οποίος γίνεται ως εξής:

$$V_B = V_{BE} + V_E = 0.7 + 0 = 0.7V \quad (3.1)$$

Κατόπιν, μπορεί να υπολογιστεί το ρεύμα I_B μέσω της αντίστασης βάσης:

$$I_B = \frac{V_{BB} - V_B}{R_B} = \frac{4 - 0.7V}{100k\Omega} = 0.033mA = 33\mu A \quad (3.2)$$

Το κύκλωμα πόλωσης βάσης δύναται να λειτουργήσει είτε στην ενεργό περιοχή, είτε στον κορεσμό είτε στην αποκοπή. Για να συνεχιστεί η ανάλυσή του, γίνεται η υπόθεση ότι το κύκλωμα λειτουργεί στην ενεργό περιοχή. Οπότε μπορούν να χρησιμοποιηθούν οι εξισώσεις από τις οποίες διέπεται η λειτουργία του στη συγκεκριμένη περιοχή. Έστω λοιπόν ότι ισχύουν:

$$I_C = \beta * I_B \text{ και } I_C \cong I_E \quad (3.3)$$

Συνεπώς, το ρεύμα συλλέκτη θα είναι:

$$I_C = 200 * 0.033\text{mA} = 6.6\text{mA} \cong I_E \quad (3.4)$$

Για μεγαλύτερη ακρίβεια, το ρεύμα εκπομπού είναι:

$$I_E = I_C + I_B = 6.6 + 0.033 = 6.633\text{mA} \quad (3.5)$$

Παρόλα αυτά, λόγω του πολύ μικρού ρεύματος βάσης, συνηθίζεται να ισούται με το ρεύμα συλλέκτη. Προχωρώντας, μπορεί να υπολογιστεί πλέον η τάση συλλέκτη μέσω του νόμου του Kirchhoff:

$$V_C = V_{CC} - I_C * R_C = 10\text{V} - 6.6\text{mA} * 1\text{k}\Omega = 3.4\text{V} \quad (3.6)$$

Για να επαληθευτεί η υπόθεση ότι το κύκλωμα λειτουργεί πράγματι στην ενεργό περιοχή θα πρέπει να ισχύει η συνθήκη: $V_C > V_B > V_E$, δηλαδή η δίοδος συλλέκτη – βάσης να είναι πολωμένη ανάστροφα, ενώ η δίοδος βάσης – εκπομπού να είναι πολωμένη ορθά. Έχοντας υπολογίσει τις συγκεκριμένες τάσεις βλέπουμε ότι η υπόθεση ισχύει, άρα το κύκλωμα λειτουργεί στην ενεργό περιοχή λειτουργίας. Έστω τώρα ότι στις παραπάνω τιμές αλλάζει το κέρδος ρεύματος β και γίνεται 300. Κάνοντας και πάλι τους υπολογισμούς έχουμε:

$$V_B = V_{BE} + V_E = 0.7 + 0 = 0.7\text{V} \quad (3.7)$$

$$I_B = \frac{V_{BB} - V_B}{R_B} = \frac{4 - 0.7\text{V}}{100\text{k}\Omega} = 0.033\text{mA} = 33\mu\text{A} \quad (3.8)$$

$$I_C = 300 * 0.033\text{mA} = 9.9\text{mA} \cong I_E \quad (3.9)$$

$$V_C = V_{CC} - I_C * R_C = 10\text{V} - 9.9\text{mA} * 1\text{k}\Omega = 0.1\text{V} \quad (3.10)$$

Αυτό που παρατηρούμε είναι ότι $V_C < V_B$. Σε αυτή την περίπτωση οι εξισώσεις που χρησιμοποιήθηκαν με την υπόθεση ότι το κύκλωμα λειτουργεί στην ενεργό περιοχή δεν ισχύουν. Οπότε, από τη στιγμή που δεν βρισκόμαστε στην περιοχή αποκοπής, καθώς υπάρχει επαρκής τάση για να πολωθεί ορθά η ένωση βάσης – εκπομπού και δεν ισχύουν οι εξισώσεις της ενεργού περιοχής, σημαίνει ότι το κύκλωμα βρίσκεται στην περιοχή κορεσμού. Εδώ πέρα, καθότι ο συλλέκτης και ο εκπομπός λειτουργούν περίπου ως βραχυκυκλωμένοι, θεωρείται ότι υπάρχει μια μικρή διαφορά τάσης της τάξης των 0.3V. Αυτό οδηγεί στο σχηματισμό της εξίσωσης η οποία χρησιμοποιείται στον κορεσμό και είναι:

$$V_{CE} = V_{CE(\text{SAT})} \cong 0.3\text{V} \quad (3.11)$$

Στη συγκεκριμένη περιοχή, το κύκλωμα λειτουργεί με ένα διαφορετικό κέρδος ρεύματος β από αυτό που έχει στην ενεργό περιοχή ως ενισχυτής. Αυτό το κέρδος ονομάζεται $\beta_{dc(\text{SAT})}$ και είναι πάντα μικρότερο από το β . Όσο μικρότερο είναι τόσο πιο έντονα βρίσκεται το κύκλωμα στον κορεσμό. Γυρνώντας στους υπολογισμούς των τάσεων και των ρευμάτων, χρησιμοποιούμε τα δεδομένα που έχουμε βρισκόμενοι στον κορεσμό. Οπότε θα έχουμε:

$$V_B = V_{BE} + V_E = 0.7 + 0 = 0.7\text{V} \quad (3.12)$$

$$I_B = \frac{V_{BB} - V_B}{R_B} = \frac{4 - 0.7\text{V}}{100\text{k}\Omega} = 0.033\text{mA} = 33\mu\text{A} \quad (3.13)$$

$$V_{CE} = V_{CE(SAT)} = V_C - V_E = V_C \cong 0.3V \quad (3.14)$$

Από τα παραπάνω και με τη βοήθεια του νόμου του Ohm μπορούμε να υπολογίσουμε το ρεύμα συλλέκτη I_C ως εξής:

$$I_C = \frac{V_{CC} - V_C}{R_C} = \frac{10 - 0.3V}{1k\Omega} = 9.7mA \quad (3.15)$$

Για την εύρεση του ρεύματος εκπομπού χρησιμοποιείται η γενικευμένη εξίσωση:

$$I_E = I_C + I_B = 9.7 + 0.033 = 9.733mA \quad (3.16)$$

Τέλος, την τιμή του $\beta_{dc(SAT)}$ την παίρνουμε από τον τύπο:

$$\beta_{dc(SAT)} = \frac{I_{C(SAT)}}{I_B} = \frac{9.7mA}{0.033mA} = 293.94 < \beta \quad (3.17)$$

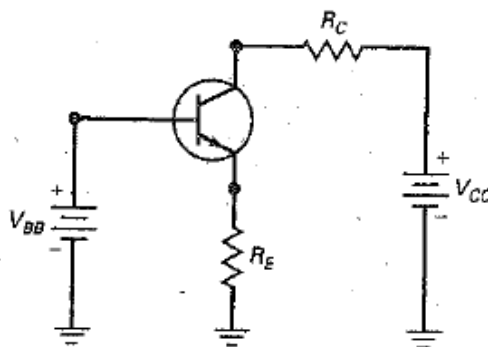
όπου $I_{C(SAT)}$ το ρεύμα συλλέκτη που υπολογίστηκε στην περιοχή κορεσμού. Τέλος, το τρανζίστορ μπορεί να λειτουργήσει στην περιοχή αποκοπής, έχοντας διακοπτική λειτουργία. Αυτό σημαίνει ότι οι ενώσεις βάσης – εκπομπού και συλλέκτη – βάσης είναι πολωμένες ανάστροφα, δηλαδή δεν υπάρχουν ρεύματα. Σε αυτή την περίπτωση μπορεί να θεωρηθεί ότι λειτουργεί σαν ανοικτό κύκλωμα. Επειδή ακριβώς δεν υπάρχει πτώση τάσης λόγω της απουσίας ρεύματος, ισχύει η εξής εξίσωση:

$$V_C = V_{CC} \quad (3.18)$$

Συνεπώς, αν στο παραπάνω παράδειγμα αντί για 4V η V_{BB} γίνει 0.6V, δεν θα υπάρχει τάση ικανή ούτως ώστε να πολωθεί ορθά η διόδος βάσης εκπομπού, οπότε το τρανζίστορ θα λειτουργήσει στην αποκοπή. Προφανώς όλα τα ρεύματα καθώς και η τάση εκπομπού V_E θα είναι 0, ενώ η τάση βάσης V_B όπως και η τάση συλλέκτη V_C θα έχουν τάση ίση με των πηγών V_{BB} και V_{CC} αντίστοιχα.

3.2.2 Κύκλωμα πόλωσης εκπομπού

Προχωρώντας την ανάλυση, το κύκλωμα πόλωσης εκπομπού έχει παρόμοια λειτουργία με αυτό της πόλωσης βάσης. Η διαφορά του είναι η μετατόπιση της αντίστασης βάσης στον εκπομπό, οπότε και η συνδεσμολογία του διαμορφώνεται ως εξής:



Εικόνα 3.2 Κύκλωμα πόλωσης εκπομπού

Η μετατόπιση αυτή δημιουργεί την ανεξαρτησία του κέρδους ρεύματος β στις μεταβολές παραμέτρων του κυκλώματος. Αυτό σημαίνει πως ότι β και να δοθεί, η περιοχή λειτουργίας θα παραμείνει ίδια και δεν θα αλλάξει. Το χαρακτηριστικό του συγκεκριμένου κυκλώματος είναι ότι παρέχει σταθερή τιμή για το ρεύμα εκπομπού, όπως αντίστοιχα και το κύκλωμα πόλωσης βάσης για το ρεύμα βάσης. Για

την επίλυσή του, έστω ότι $V_{CC} = 12V$, $V_{BB} = 4V$, $R_E = 2.2k\Omega$, $R_C = 4.7k\Omega$. Το πρώτο που μπορεί να υπολογιστεί είναι η τάσης βάσης V_B . Καθότι η τάση της πηγής δεν διαπερνάει κάποια αντίσταση, η τάση βάσης θα ισούται με την τάση της πηγής, δηλαδή:

$$V_B = V_{BB} = 4V \quad (3.19)$$

Με δεδομένο ότι η δίοδος βάσης εκπομπού είναι πολωμένη ορθά και άρα κρατάει στα άκρα της $0.7V$, υπολογίζεται η τάση εκπομπού V_E ως εξής:

$$V_E = V_B - V_{BE} = 4 - 0.7 = 3.3V \quad (3.20)$$

Έτσι, μέσω του νόμου του Ohm μπορεί να υπολογιστεί το ρεύμα εκπομπού I_E :

$$I_E = \frac{V_E}{R_E} = \frac{3.3V}{2.2k\Omega} = 1.5mA \quad (3.21)$$

Συνεχίζοντας, γίνεται η υπόθεση για λειτουργία στην ενεργό περιοχή. Αυτό σημαίνει ότι το ρεύμα συλλέκτη και το ρεύμα εκπομπού θα είναι περίπου ίσα. Η πολύ μικρή μεταξύ τους διαφορά προκύπτει από τον συντελεστή κέρδους ρεύματος κοινής βάσης α , του οποίου η τιμή προκύπτει από τη σχέση:

$$\alpha = \frac{\beta}{\beta+1} \quad (3.22)$$

Δίνοντας οπότε μια τιμή για το $\beta = 100$ το α προκύπτει 0.99 . Έτσι, το ρεύμα συλλέκτη θα προκύψει από τον τύπο:

$$I_C = \alpha I_E = 0.99 * 1.5 = 1.485mA \quad (3.23)$$

Γνωρίζοντας το ρεύμα συλλέκτη και το ρεύμα εκπομπού, μπορεί να υπολογιστεί το ρεύμα βάσης:

$$I_B = I_E - I_C = 1.5 - 1.485 = 15\mu A \quad (3.24)$$

Επειδή η τιμή που προκύπτει για το ρεύμα βάσης είναι πολύ μικρή, μπορεί να θεωρηθεί αμελητέο. Συνεπώς, αμελητέα μπορεί να θεωρηθεί και η διαφορά μεταξύ ρεύματος συλλέκτη και ρεύματος εκπομπού και γι' αυτό λογίζονται ως περίπου ίσα. Οπότε, με την εφαρμογή του κανόνα του Kirchhoff, μπορεί να υπολογιστεί η τάση συλλέκτη:

$$V_C = V_{CC} - I_C * R_C = 12V - 1.485mA * 4.7k\Omega = 5.02V \quad (3.25)$$

Τελικά, αφού ισχύει η σχέση $V_C > V_B > V_E$, η λειτουργία του τρανζίστορ πραγματοποιείται στην ενεργό περιοχή. Όπως και στο κύκλωμα πόλωσης βάσης, έτσι κι εδώ, το τρανζίστορ εκτός από την ενεργό περιοχή, μπορεί να λειτουργήσει στον κορεσμό και στην αποκοπή. Αλλάζοντας οπότε την τάση της πηγής V_{BB} από 4 σε $5V$, θα έχουμε τα εξής:

$$V_B = V_{BB} = 5V \quad (3.26)$$

$$V_E = V_B - V_{BE} = 5 - 0.7 = 4.3V \quad (3.27)$$

$$I_E = \frac{V_E}{R_E} = \frac{4.3V}{2.2k\Omega} = 1.955mA \quad (3.28)$$

$$I_C \cong I_E \quad (3.29)$$

$$V_C = V_{CC} - I_C * R_C = 12V - 1.955mA * 4.7k\Omega = 2.8115V \quad (3.30)$$

Αφού $V_C = 2.8115V < V_B = 5V$ το κύκλωμα δεν λειτουργεί στην ενεργό περιοχή και συνεπώς δεν ισχύουν οι εξισώσεις που διαμορφώνονται από την υπόθεση. Με δεδομένο ότι το κύκλωμα δεν λειτουργεί στην αποκοπή, καθώς υπάρχει τάση ικανή για την ορθή πόλωση της διόδου βάσης – εκπομπού, προκύπτει λειτουργία στον κορεσμό. Άρα τα δεδομένα που έχουμε με βάση τους παραπάνω υπολογισμούς είναι ότι $V_B = 5V$, $V_E = 4.3V$, $I_E = 1.955mA$. Με γνώμονα ότι στον κορεσμό ισχύει η εξίσωση $V_{CE} = V_{CE(SAT)} \cong 0.3V$ μπορούμε να υπολογίσουμε την τάση συλλέκτη V_C ως εξής:

$$V_C = V_E + V_{CE} = 4.3 + 0.3 = 4.6V \quad (3.31)$$

Κατόπιν, μπορεί να υπολογιστεί πλέον το ρεύμα συλλέκτη I_C :

$$I_C = \frac{V_{CC} - V_C}{R_C} = \frac{12 - 4.6V}{4.7k\Omega} = 1.574mA \quad (3.32)$$

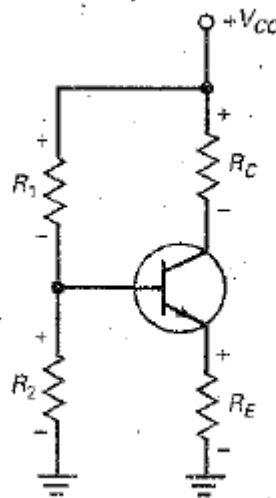
Τέλος με τη βοήθεια της εξίσωσης που ισχύει για τα ρεύματα του τρανζίστορ υπολογίζεται το ρεύμα βάσης I_B :

$$I_B = I_E - I_C = 1.955 - 1.574 = 0.381mA \quad (3.33)$$

Όσον αφορά την λειτουργία του τρανζίστορ στην αποκοπή, αλλάζοντας την τιμή της V_{BB} σε $0.5V$, η διόδος βάσης – εκπομπού δεν θα έχει τα απαιτούμενα $0.7V$ που χρειάζεται προκειμένου να πολωθεί ορθά, οπότε το κύκλωμα εισέρχεται σε λειτουργία αποκοπής, όπερ σημαίνει ότι δεν θα υπάρχουν καθόλου ρεύματα, ενώ η V_B και η V_C θα έχουν τιμή ίση με των πηγών V_{BB} και V_{CC} , δηλαδή 0.5 και $12V$ αντίστοιχα.

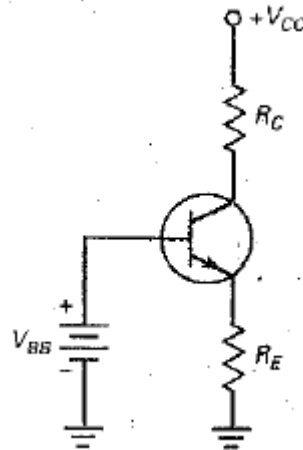
3.2.3 Κύκλωμα πόλωσης διαιρέτη τάσης

Το πιο ευρέως διαδεδομένο και χρησιμοποιούμενο κύκλωμα πόλωσης είναι αυτό του διαιρέτη τάσης. Η ονομασία προκύπτει επειδή αριστερά του τρανζίστορ, στη βάση του δηλαδή, υπάρχει ένας διαιρέτης τάσης μεταξύ δύο αντιστάσεων. Η συνδεσμολογία του φαίνεται στην παρακάτω εικόνα:



Εικόνα 3.3 Κύκλωμα πόλωσης διαιρέτη τάσης

Όπως φαίνεται, εκτός από τον διαιρέτη τάσης, αποτελείται επίσης από μια πηγή V_{CC} , μια αντίσταση συλλέκτη R_C και μια αντίσταση εκπομπού R_E . Ουσιαστικά πρόκειται για το κύκλωμα πόλωσης εκπομπού τροποποιημένο. Για την ανάλυσή του, το πρώτο πράγμα που πρέπει να βρεθεί είναι η τάση V_{BB} . Αυτή μπορεί να προκύψει με την εφαρμογή του θεωρήματος Thevenin στη βάση του τρανζίστορ, οπότε θα προκύψει η τάση V_{BB} και μια αντίσταση Thevenin R_{TH} . Επειδή η αντίσταση R_{TH} διαρρέεται από ένα πολύ μικρό ρεύμα βάσης I_B , κάτι το οποίο θα δημιουργήσει μια πολύ μικρή πτώση τάσης, δεν υπολογίζεται στην απλοποιημένη προσέγγιση του κυκλώματος. Το ισοδύναμο κύκλωμα με την εφαρμογή του θεωρήματος Thevenin, χωρίς την αντίσταση, φαίνεται παρακάτω:



Εικόνα 3.4 Απλοποιημένο κύκλωμα

Για την εύρεση των τάσεων και των ρευμάτων ακολουθείται παρόμοια διαδικασία με τα προηγούμενα κυκλώματα. Πριν από αυτό υπολογίζεται η τάση V_{BB} από το διαιρέτη τάσης μεταξύ R_1 και R_2 με τον εξής τύπο:

$$V_{BB} = \frac{R_2}{R_1 + R_2} V_{CC} \quad (3.34)$$

Δίνοντας τις τιμές $V_{CC} = 24V$, $R_1 = 10k\Omega$, $R_2 = 2.2k\Omega$, $R_E = 2k\Omega$ και $R_C = 3.6k\Omega$ θα έχουμε:

$$V_{BB} = \frac{2.2k\Omega}{10 + 2.2k\Omega} * 24V = 4.328V \quad (3.35)$$

Καθότι δεν μεσολαβεί κάποια αντίσταση μέχρι τη βάση, η τάση V_B θα ισούται με τη V_{BB} . Ακολουθώντας, υπολογίζονται η τάση και το ρεύμα εκπομπού:

$$V_E = V_B - V_{BE} = 4.328 - 0.7 = 3.628V \quad (3.36)$$

$$I_E = \frac{V_E}{R_E} = \frac{3.628V}{2k\Omega} = 1.814mA \quad (3.37)$$

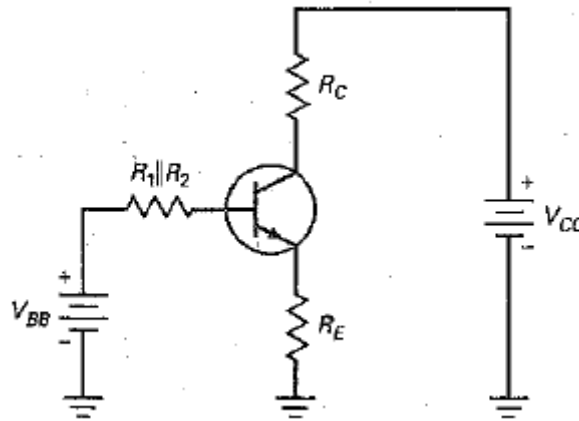
Κατόπιν, υποθέτεται η λειτουργία στην ενεργό περιοχή, η οποία μας δίνει ότι $I_C \cong I_E$ (διαφορά μόνο ως προς τον συντελεστή κέρδους ρεύματος κοινής βάσης α). Επειδή ο διαιρέτης τάσης δίνει μια σταθερή τιμή για το ρεύμα εκπομπού και το κέρδος ρεύματος β δεν επηρεάζει τη λειτουργία του κυκλώματος, επιτυγχάνεται η πόλωση, η οποία σημαίνει λειτουργία στην ενεργό περιοχή, στην περίπτωση που ικανοποιείται η γνωστή συνθήκη $V_C > V_B > V_E$. Σε διαφορετική περίπτωση, το κύκλωμα δεν λειτουργεί σωστά. Έτσι λοιπόν, αφού $I_C \cong I_E = 1.814mA$, υπολογίζεται η τάση συλλέκτη V_C :

$$V_C = V_{CC} - I_C * R_C = 24V - 1.814mA * 3.6k\Omega = 17.5V \quad (3.38)$$

και επειδή $V_C > V_B > V_E$, το κύκλωμα βρίσκεται στην ενεργό περιοχή λειτουργίας.

3.2.3.1 Ισοδύναμο κύκλωμα με R_{TH}

Στην περίπτωση που στα στοιχεία του κυκλώματος υπολογίζεται και η αντίσταση R_{TH} , η συνδεσμολογία του είναι η εξής:



Εικόνα 3.5 Ισοδύναμο κύκλωμα με R_{TH}

Όπως φαίνεται, η τιμή της αντίστασης R_{TH} προκύπτει από τον παράλληλο συνδυασμό των αντιστάσεων R_1 και R_2 , οι οποίες συμμετέχουν στο σχηματισμό του διαιρέτη τάσης στη βάση του τρανζίστορ. Η διαφορά με την προσεγγιστική επίλυση παρατηρείται στην άγνωστη πλέον τάση βάσης V_B , καθώς αυτή δεν μπορεί να πάρει όλη την τάση της πηγής λόγω της μεσολάβησης της R_{TH} . Για να βρεθεί το βασικό ρεύμα του τρανζίστορ, το ρεύμα εκπομπού, πρέπει να σχηματιστεί μια εξίσωση στον κλάδο που ξεκινάει από την τάση της πηγής V_{BB} και καταλήγει στην αντίσταση εκπομπού R_E . Οπότε προκύπτει:

$$V_{BB} - V_{BE} = I_E * R_E + I_B * R_1 || R_2 \quad (3.39)$$

Και επειδή $I_B = \frac{I_E}{\beta}$ στην ενεργό περιοχή, προκύπτει ο εξής τύπος για το ρεύμα εκπομπού:

$$I_E = \frac{V_{BB} - V_{BE}}{R_E + (R_1 || R_2) / \beta} \quad (3.40)$$

Η $R_{TH} = R_1 || R_2$ υπολογίζεται από τον τύπο:

$$R_{TH} = \frac{R_1 * R_2}{R_1 + R_2} \quad (3.41)$$

Συνεπώς, με τις τιμές που δόθηκαν στην προσεγγιστική επίλυση για τις V_{CC} , R_1 , R_2 , R_E , R_C και δίνοντας στο κέρδος ρεύματος β την τιμή 100, η R_{TH} και κατόπιν το ρεύμα εκπομπού θα είναι:

$$R_{TH} = \frac{R_1 * R_2}{R_1 + R_2} = \frac{10 * 2.2k\Omega}{10 + 2.2k\Omega} = 1.803k\Omega \quad (3.42)$$

$$I_E = \frac{V_{BB} - V_{BE}}{R_E + (R_1 || R_2) / \beta} = \frac{4.328 - 0.7V}{2 + (1.803 / 100)k\Omega} = 1.798mA \quad (3.43)$$

Ουσιαστικά, η παράμετρος του κέρδους ρεύματος β που προστέθηκε στους υπολογισμούς, δεν επηρέασε σε μεγάλο βαθμό την τιμή του ρεύματος εκπομπού, καθώς από 1.814mA έγινε 1,798mA, μεταβολή η οποία δεν έχει καμία σημασία για το κύκλωμα λόγω της ανοχής των αντιστάσεων σε μια ενδεχόμενη φυσική κατασκευή του. Η συνθήκη η οποία πρέπει να αληθεύει για να θεωρείται ότι το β δεν επηρεάζει τους υπολογισμούς είναι:

$$\frac{R_1 || R_2}{\beta} \ll 0.1 * R_E \quad (3.44)$$

δηλαδή:

$$\frac{1.803k\Omega}{100} \ll 0.1 * 2k\Omega \Rightarrow 0.01803 \ll 0.2 \text{ ισχύει} \quad (3.45)$$

Τέλος, για να θεωρείται ένας διαιρέτης τάσης ότι είναι σταθερός θα πρέπει να ισχύει:

$$R_1 || R_2 < 0.1\beta R_E \quad (3.46)$$

ενώ για να θεωρηθεί πολύ σταθερός:

$$R_1 || R_2 < 0.01\beta R_E \quad (3.47)$$

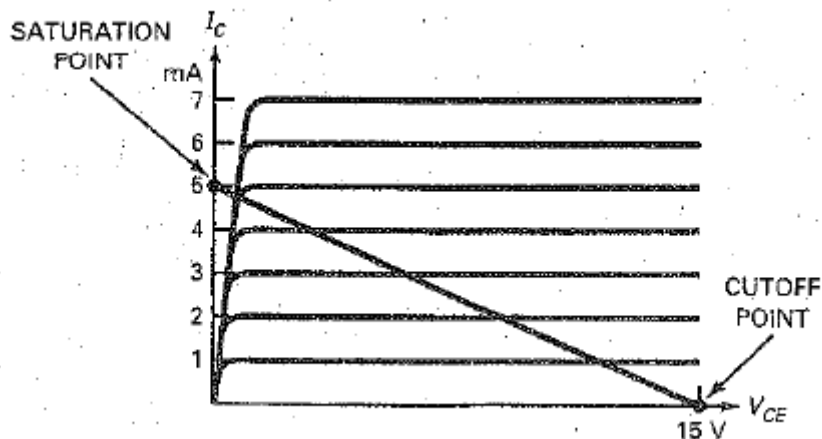
Με τις παραπάνω τιμές επαληθεύεται η δεύτερη συνθήκη καθώς:

$$1.803 < 2 \quad (3.48)$$

άρα ο διαιρέτης τάσης μπορεί να χαρακτηριστεί πολύ σταθερός

3.2.3.2 Γραμμή φορτίου και σημείο λειτουργίας Q

Σε κάθε κύκλωμα που έχει τρανζίστορ υπάρχει μια γραμμή φορτίου. Η γραμμή φορτίου είναι μια γραφική αναπαράσταση του I_C ως προς το V_{CE} . Ονομάζεται έτσι επειδή παριστάνει την επίδραση του φορτίου στα I_C και V_{CE} . Η χρησιμότητά της έγκειται στην παρουσίαση όλων των δυνατών σημείων λειτουργίας του τρανζίστορ. Το σημείο λειτουργίας μπορεί να βρεθεί σε κάθε κύκλωμα με τρανζίστορ και αποτελείται από δύο σημεία, το ρεύμα συλλέκτη I_C και την τάση συλλέκτη – εκπομπού V_{CE} . Η γραμμή φορτίου, εκτός από το σημείο λειτουργίας Q, αποτελείται από άλλα δύο σημεία, το σημείο κορεσμού και το σημείο αποκοπής. Το σημείο κορεσμού είναι εκείνο στο οποίο η γραμμή φορτίου τέμνει την περιοχή κορεσμού των καμπυλών του συλλέκτη, ενώ στο σημείο αποκοπής τέμνει την περιοχή αποκοπής των αντίστοιχων καμπυλών. Σε γραφική παράσταση $I_C - V_{CE}$ τα δύο αυτά σημεία φαίνονται όπως στην εικόνα παρακάτω:



Εικόνα 3.6 Γραμμή φορτίου

Στο σημείο κορεσμού μπορεί να βρεθεί το ρεύμα κορεσμού $I_{C(SAT)}$, ενώ στο σημείο αποκοπής βρίσκουμε την τάση αποκοπής $V_{CE(CUTOFF)}$. Ενώνοντας τα δύο αυτά σημεία παίρνουμε τη γραμμή φορτίου. Για το σημείο Q, όπως αναφέρθηκε, αρκεί να βρεθεί το ρεύμα συλλέκτη I_C και η τάση συλλέκτη – εκπομπού V_{CE} . Οπότε, με τις τιμές που δόθηκαν για τις παραμέτρους του κυκλώματος πόλωσης διαιρέτη τάσης, το $I_C \cong I_E = 1.814\text{mA}$, ενώ η V_{CE} είναι:

$$V_{CE} = V_C - V_E = 17.5 - 3.628 = 13.872\text{V} \quad (3.49)$$

Για την εύρεση του ρεύματος κορεσμού χρησιμοποιείται ο τύπος:

$$I_{C(SAT)} = \frac{V_{CC}}{R_C + R_E} \quad (3.50)$$

ενώ η τάσης αποκοπής $V_{CE(CUTOFF)}$ έχει τιμή:

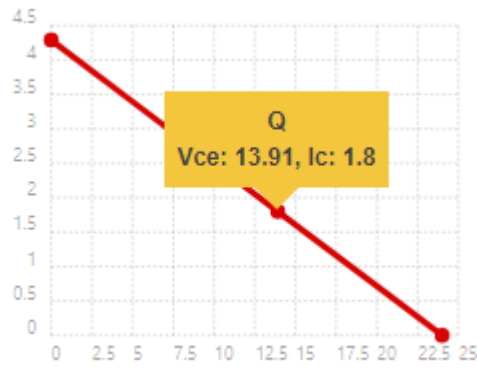
$$V_{CE(CUTOFF)} = V_{CC} \quad (3.51)$$

Έτσι λοιπόν, το ρεύμα κορεσμού και η τάση αποκοπής στο παραπάνω παράδειγμα είναι:

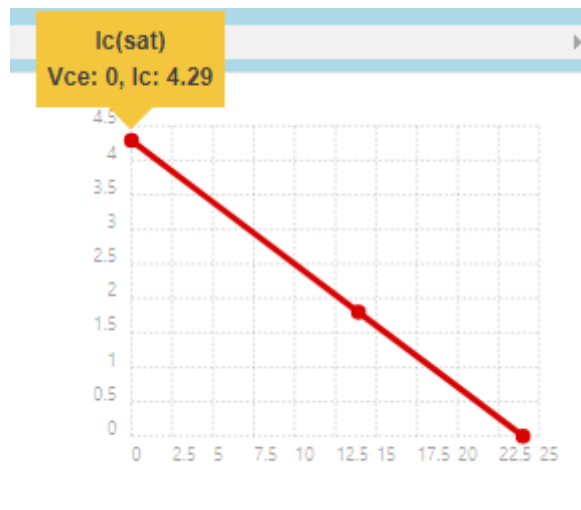
$$I_{C(SAT)} = \frac{24\text{V}}{3.6\text{k}\Omega + 2\text{k}\Omega} = 4.29\text{mA} \quad (3.52)$$

$$V_{CE(CUTOFF)} = 24\text{V} \quad (3.53)$$

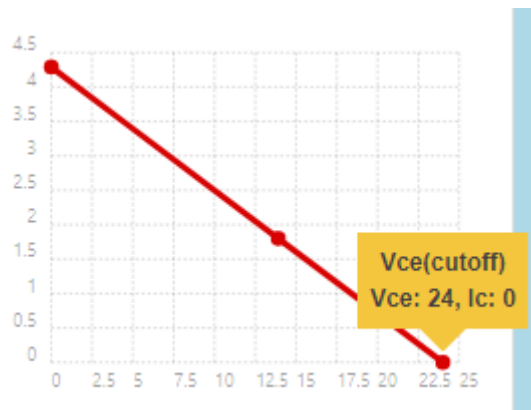
Στη γραφική παράσταση $I_C - V_{CE}$ τα παραπάνω σημεία φαίνονται όπως στα παρακάτω σχήματα, τα οποία προέρχονται από το αντίστοιχο κομμάτι υλοποίησης του γραφήματος στην εφαρμογή:



Εικόνα 3.7 Σημείο λειτουργίας Q

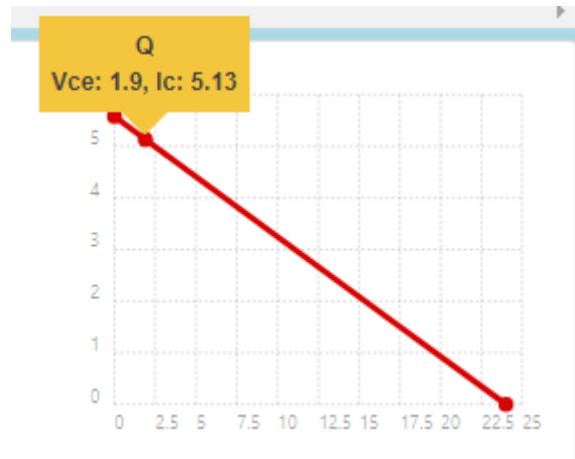


Εικόνα 3.8 Ρεύμα κορεσμού $I_{C(SAT)}$



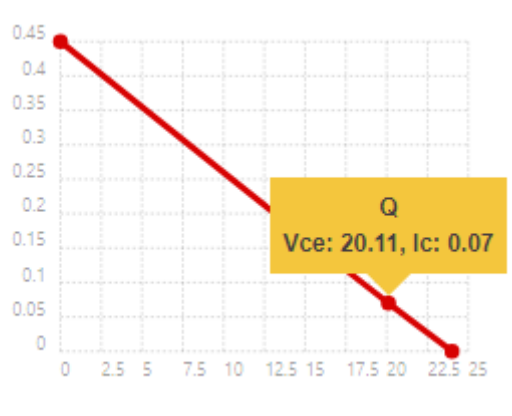
Εικόνα 3.9 Τάση αποκοπής $V_{CE(CUTOFF)}$

Οι πολύ μικρές διαφορές που παρατηρούνται στους υπολογισμούς των V_{CE} και I_C είναι λόγω της πολύ μικρής διαφοράς που υπάρχει μεταξύ I_C και I_E , τα οποία θεωρούνται περίπου ίσα στην ενεργό περιοχή. Αξίζει να αναφερθεί ότι το ρεύμα κορεσμού και η τάση αποκοπής ρυθμίζονται από τις V_{CC} , R_1 , R_2 και R_C , ενώ το σημείο Q από την R_E . Όσο η R_E μικραίνει, τόσο το σημείο Q κινείται προς τον κορεσμό, ενώ όσο μεγαλώνει, κινείται προς την αποκοπή. Για παράδειγμα αν θέσουμε την $R_E = 700\Omega$, το σημείο θα μετατοπιστεί όπως φαίνεται παρακάτω:



Εικόνα 3.10 Μετατόπιση σημείου Q προς τον κορεσμό

ενώ αν θέσουμε την $R_E = 50k\Omega$ το σημείο Q θα φαίνεται κάπως έτσι:



Εικόνα 3.11 Μετατόπιση σημείου Q προς την αποκοπή

3.3 Τελεστικοί ενισχυτές

Καθότι η λεπτομερής ανάλυση των συνδεσμολογιών των κυκλωμάτων των τελεστικών ενισχυτών και των επιμέρους εννοιών από τις οποίες απαρτίζονται έχει γίνει στο προηγούμενο κεφάλαιο, εδώ πέρα θα γίνει κατευθείαν επίλυση παραδειγμάτων, μέσα από τα οποία θα φανεί η λειτουργία τους.

3.3.1 Κύκλωμα αναστρέφοντος ενισχυτή

Έστω ένα κύκλωμα που αποτελείται από έναν 741C αναστρέφων ενισχυτή με τροφοδοσία $\pm 15V$, μια εναλλασσόμενη τάση εισόδου $V_{in} = 200mV$, $R_1 = 20k\Omega$, $R_F = 50k\Omega$ και συχνότητα $f = 1kHz$. Ο σκοπός είναι να βρεθεί το ενισχυμένο και αντεστραμμένο, λόγω της διαφοράς φάσης, σήμα εξόδου V_{out} . Το πρώτο που μπορεί να υπολογιστεί είναι το κέρδος A_{CL} της συνδεσμολογίας, το οποίο δίνεται από τον τύπο:

$$A_{CL} = -\frac{R_F}{R_1} = -\frac{50k\Omega}{20k\Omega} = -2.5 \quad (3.54)$$

Το αρνητικό πρόσημο που προκύπτει αφορά την διαφορά φάσης μεταξύ των σημάτων εισόδου και εξόδου. Αφού έχει υπολογιστεί το κέρδος, μπορεί να υπολογιστεί και η έξοδος ως εξής:

$$V_{out} = A_{CL} * V_{in} = -2.5 * 200mV = -0.5V_{pp} \quad (3.55)$$

Κεφάλαιο 3

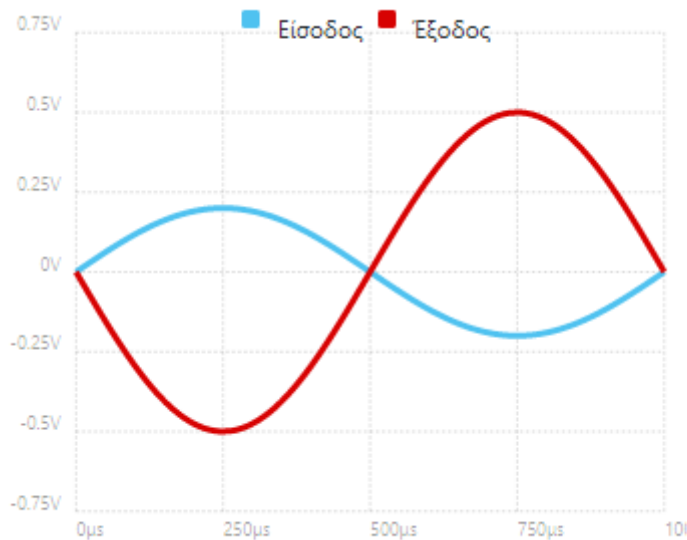
Τέλος, με δεδομένο ότι η συχνότητα μοναδιαίου κέρδους $f_{\text{unity}} = 1\text{MHz}$ για το συγκεκριμένο μοντέλο τελεστικού ενισχυτή, υπολογίζεται η συχνότητα κλειστού βρόχου f_{CL} :

$$f_{\text{CL}} = \frac{f_{\text{unity}}}{A_{\text{CL}}} = \frac{1\text{MHz}}{2.5} = 400\text{kHz} \quad (3.56)$$

Συνεπώς, η σχεδίαση των ημιτονοειδών σημάτων εισόδου και εξόδου σε χρόνο μιας περιόδου, που προκύπτει από τον τύπο:

$$T = \frac{1}{f} = \frac{1}{1\text{kHz}} = 1000\mu\text{s} \quad (3.57)$$

μοιάζει όπως στο σχήμα παρακάτω (προέρχεται από την αντίστοιχη υλοποίηση στην εφαρμογή):



Εικόνα 3.12 Η είσοδος και η έξοδος του ενισχυτή σε ημιτονοειδή μορφή

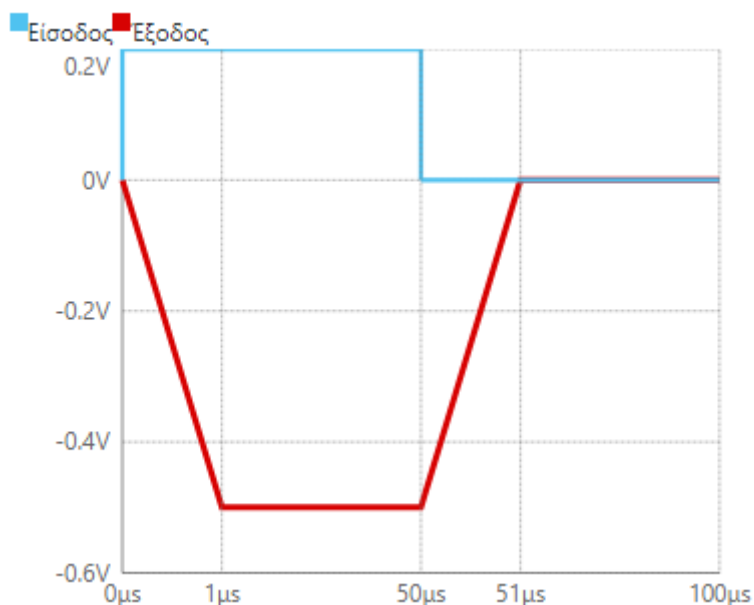
Στην περίπτωση που η είσοδος και η έξοδος θέλουμε να είναι τετραγωνικοί παλμοί, χρειάζεται να προστεθεί και ο υπολογισμός του Δ_t . Αυτό γίνεται με τη βοήθεια του S_R . Έστω λοιπόν ότι έχουμε συχνότητα $f = 10\text{kHz}$ και $S_R = 0.5\text{V}/\mu\text{s}$. Το Δ_t , δηλαδή το χρονικό διάστημα που θα χρειαστεί η τάση εξόδου για κάθε μετάβαση από τα 0 στα 0.5V υπολογίζεται από τον τύπο:

$$S_R = \frac{\Delta v}{\Delta_t} \Rightarrow \frac{0.5\text{V}}{1\mu\text{s}} = \frac{0.5\text{V}}{\Delta_t} \Rightarrow \Delta_t = 1\mu\text{s} \quad (3.58)$$

Η περίοδος για συχνότητα $f = 10\text{kHz}$ θα είναι:

$$T = \frac{1}{f} = \frac{1}{10\text{kHz}} = 100\mu\text{s} \quad (3.59)$$

Η απεικόνιση φαίνεται στο παρακάτω σχήμα (επίσης προέρχεται από την εφαρμογή):



Εικόνα 3.13 Η είσοδος και η έξοδος σε μορφή τετραγωνικών παλμών

3.3.2 Κύκλωμα μη αναστρέφοντος ενισχυτή

Για την επίλυση του κυκλώματος του μη αναστρέφοντος ενισχυτή, έστω το μοντέλο 741C και δεδομένα $V_{in} = 5V$, $R_1 = 2k\Omega$, $R_F = 3k\Omega$, $f = 5kHz$ και τροφοδοσία $\pm 15V$. Εδώ πέρα, η πηγή τάσης εισόδου είναι συνδεδεμένη στη μη αναστρέφουσα είσοδο του ενισχυτή, οπότε το σήμα της δεν θα έχει διαφορά φάσης με αυτό της εξόδου. Έτσι λοιπόν, με τρόπο αντίστοιχο όπως στην επίλυση του κυκλώματος του αναστρέφοντος ενισχυτή, υπολογίζεται πρώτα το κέρδος A_{CL} της συνδεσμολογίας ως εξής:

$$A_{CL} = 1 + \frac{R_F}{R_1} = 1 + \frac{3k\Omega}{2k\Omega} = 2.5 \quad (3.60)$$

Κατόπιν, μπορεί να υπολογιστεί το σήμα εξόδου V_{out} :

$$V_{out} = A_{CL} * V_{in} = 2.5 * 5V = 12.5V_{pp} \quad (3.61)$$

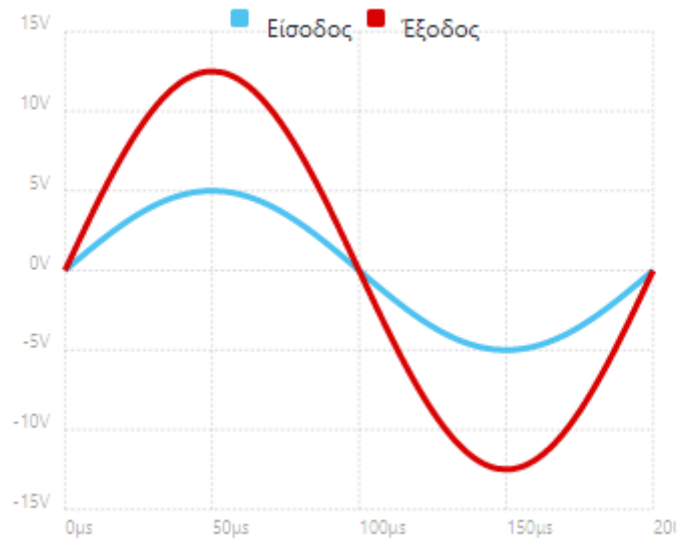
Η συχνότητα κλειστού βρόχου f_{CL} είναι:

$$f_{CL} = \frac{f_{unity}}{A_{CL}} = \frac{1MHz}{2.5} = 400kHz \quad (3.62)$$

Προκειμένου να σχεδιαστούν οι ημιτονοειδείς καμπύλες για τα σήματα εισόδου και εξόδου, υπολογίζεται, με βάση τη δοθείσα συχνότητα $f = 5kHz$, η περίοδος T :

$$T = \frac{1}{f} = \frac{1}{5kHz} = 200\mu s \quad (3.63)$$

Η απεικόνιση των σημάτων στο διάγραμμα τάσεων συναρτήσει του χρόνου είναι:

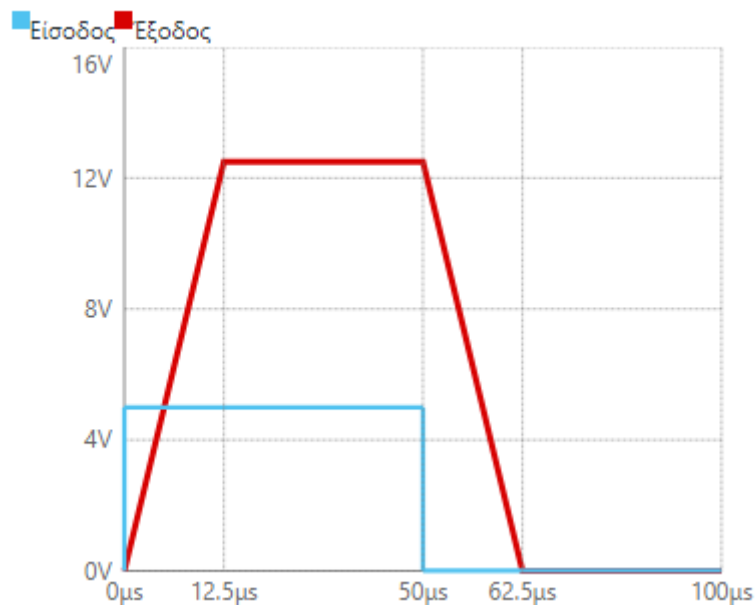


Εικόνα 3.14 Τα σήματα εισόδου και εξόδου του μη αναστρέφοντος ενισχυτή σε ημιτονοειδή μορφή Όπως και προηγουμένως, τα σήματα μπορούν να απεικονιστούν και σε μορφή τετραγωνικών παλμών, υπολογίζοντας το Δ_t μέσω του S_R . Για $S_R = 1V/\mu s$ και συχνότητα $f = 10kHz$, το Δ_t θα είναι:

$$S_R = \frac{\Delta V}{\Delta_t} \Rightarrow \frac{1V}{1\mu s} = \frac{12.5V}{\Delta_t} \Rightarrow \Delta_t = 12.5\mu s \quad (3.64)$$

σε διάστημα περιόδου:

$$T = \frac{1}{f} = \frac{1}{10kHz} = 100\mu s \quad (3.65)$$



Εικόνα 3.15 Τα σήματα εισόδου και εξόδου του μη αναστρέφοντος ενισχυτή σε μορφή τετραγωνικών παλμών

3.3.3 Κυκλώματα αθροιστή και ενισχυτή διαφορών

Προχωρώντας την επίλυση παραδειγμάτων από τα κυκλώματα των τελεστικών ενισχυτών, έστω ένας αθροιστικός ενισχυτής με δύο πηγές τάσης $V_1 = 1V$ και $V_2 = 0.5V$, ενώ οι τιμές των αντιστάσεων είναι $R_1 = 50k\Omega$, $R_2 = 20k\Omega$ και $R_F = 100k\Omega$. Η τροφοδοσία του ενισχυτή είναι $\pm 12V$. Προκειμένου να υπολογιστεί η έξοδος V_{out} , χρειάζεται να γίνει υπολογισμός των κερδών A_{C1} και A_{C2} . Αυτό γίνεται ως εξής:

$$A_{C1} = -\frac{R_F}{R_1} = \frac{100k\Omega}{50k\Omega} = -2 \quad (3.66)$$

$$A_{C2} = -\frac{R_F}{R_2} = \frac{100k\Omega}{20k\Omega} = -5 \quad (3.67)$$

Οπότε η έξοδος του ενισχυτή είναι:

$$V_{out} = (A_{C1}V_1 + A_{C2}V_2) = ((-2) * 1) + ((-5) * 0.5) = -4.5V \quad (3.68)$$

Τέλος, για την επίλυση του κυκλώματος του ενισχυτή διαφορών, έστω ότι για τις πηγές τάσης και τις αντιστάσεις της συνδεσμολογίας ισχύουν ότι: $V_1 = 5V$, $V_2 = 6V$, $R_1 = 10k\Omega$, $R_2 = 150k\Omega$ και τροφοδοσία $\pm 20V$. Το κέρδος A_{CL} της συνδεσμολογίας υπολογίζεται από τον τύπο:

$$A_{CL} = \frac{R_2}{R_1} = \frac{150k\Omega}{10k\Omega} = 15 \quad (3.69)$$

Συνεπώς, η έξοδος V_{out} θα είναι:

$$V_{out} = A_{CL} * (V_2 - V_1) = 15 * (6 - 5) = 15V \quad (3.70)$$

3.4 Επίλογος

Συμπερασματικά, το τρίτο κεφάλαιο περιλαμβάνει πρακτικής φύσεως παραδείγματα που υποδεικνύουν τους διάφορους τρόπους λειτουργίας των κυκλωμάτων της εφαρμογής. Ξεκινώντας από τα κυκλώματα πόλωσης βάσης, εκπομπού και διαιρέτη τάσης μαζί με το ισοδύναμό του και την έννοια της γραμμής φορτίου και του σημείου λειτουργίας, καταλήξαμε στην παράθεση παραδειγμάτων για τα κυκλώματα του αναστρέφοντος, μη αναστρέφοντος, αθροιστικού και τελεστικού ενισχυτή διαφορών. Με αυτόν τον τρόπο και σε συνδυασμό με το παραπάνω δεύτερο κεφάλαιο, αναλύθηκε εξ' ολοκλήρου και σε θεωρητικό και σε πρακτικό επίπεδο η λειτουργία των κυκλωμάτων που χρησιμοποιήθηκαν στην εφαρμογή.

Κεφάλαιο 4ο: Περιβάλλον και τεχνολογίες που χρησιμοποιήθηκαν

4.1 Εισαγωγή

Στο παρόν κεφάλαιο θα γίνει η ανάλυση του προγραμματιστικού περιβάλλοντος VS Code στο οποίο αναπτύχθηκε η εφαρμογή, καθώς και της ReactJS, της τεχνολογίας η οποία χρησιμοποιήθηκε για τον σκοπό αυτό. Αξίζει να αναφερθεί ότι δεν χρησιμοποιείται κάποια τεχνολογία για back-end κομμάτι. Ο λόγος είναι ότι η χρησιμότητα και η λειτουργικότητα της εφαρμογής είναι τέτοια ώστε να μην απαιτεί την καταχώρηση ή την αποθήκευση των στοιχείων του χρήστη. Ουσιαστικά αυτό που γίνεται είναι ο ενδιαφερόμενος να βλέπει τα στοιχεία και τους υπολογισμούς που θέλει, με την εκχώρηση των παραμέτρων που αυτός επιθυμεί, χωρίς να υπάρχει σκοπιμότητα για δημιουργία κάποιου προσωπικού λογαριασμού, γεγονός που θα απαιτούσε τη χρήση κάποιας βάσης δεδομένων.

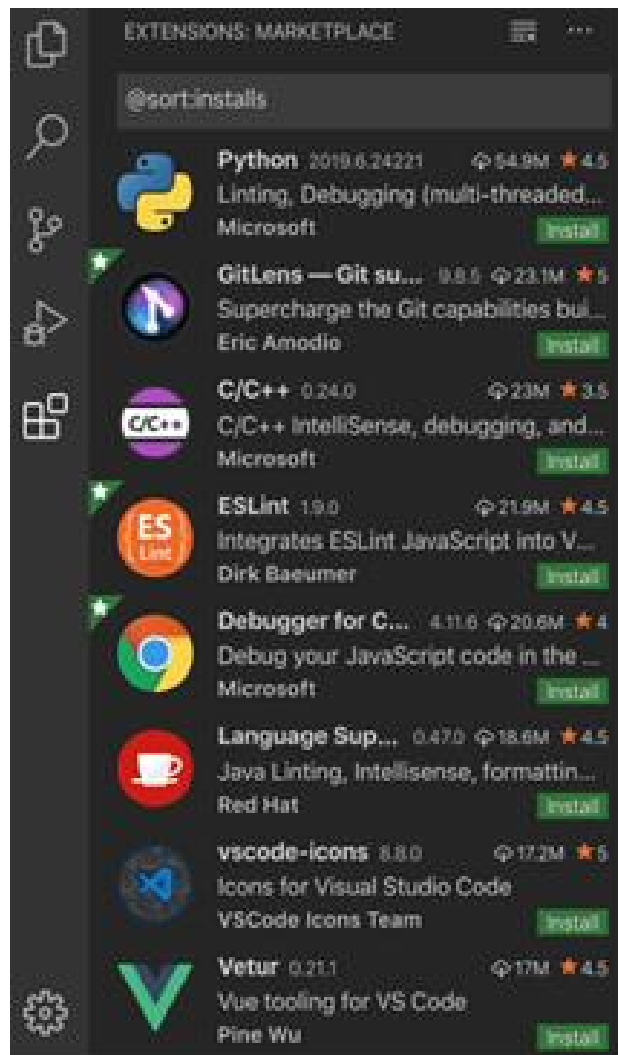
4.2 Το προγραμματιστικό περιβάλλον Visual Studio Code

4.2.1 Ιστορία

Η λειτουργία του προγραμματιστικού περιβάλλοντος Visual Studio Code, γνωστό και ως VS Code, ανακοινώθηκε για πρώτη φορά στις 29 Απριλίου 2015 από τη Microsoft στο συνέδριο Build. Λίγο αργότερα κυκλοφόρησε μια προεπισκόπηση του, ενώ στις 18 Νοεμβρίου του 2015 αναρτήθηκε ο πηγαίος κώδικάς του. Η ανάρτηση έγινε με την άδεια MIT και έγινε αμέσως διαθέσιμος και στο GitHub, στο οποίο η Microsoft έχει κυκλοφορήσει το μεγαλύτερο μέρος του κώδικα [10], [12]. Όλες οι εκδόσεις της Microsoft είναι δωρεάν και ο πηγαίος κώδικας είναι αποκλειστικός [11]. Στις 14 Απριλίου του 2016, το VS Code κυκλοφόρησε και επίσημα στο κοινό [12].

4.2.2 Χρήση και λειτουργία

Το VS Code είναι ένας επεξεργαστής πηγαίου κώδικα ο οποίος υποστηρίζει ποικίλες γλώσσες προγραμματισμού όπως C, C++, Java, Python και JavaScript [9]. Μάλιστα, σε μια έρευνα του Stack Overflow το 2021, ψηφίστηκε ως το δημοφιλέστερο εργαλείο περιβάλλοντος ανάπτυξης λογισμικού [9]. Οι λειτουργίες που υποστηρίζει έγκενται σε αποσφαλμάτωση (debugging), έξυπνη συμπλήρωση κώδικα, επισήμανση σύνταξης, αναδιαμόρφωση κώδικα καθώς και ενσωματωμένο Git [10]. Μέσα στο περιβάλλον, με τις κατάλληλες επιλογές, οι χρήστες έχουν τη δυνατότητα να προσαρμόσουν το θέμα, τις συντομεύσεις πληκτρολογίου, τις προτιμήσεις τους, καθώς και να εγκαταστήσουν μια σειρά από επεκτάσεις, οι οποίες θα τους προσφέρουν νέα χαρακτηριστικά [9]. Από μόνο του το VS Code ως προγραμματιστικό περιβάλλον προσφέρει στοιχειώδη υποστήριξη για τις περισσότερες γλώσσες προγραμματισμού [9]. Η υπογράμμιση σύνταξης, η αντιστοίχιση παρενθέσεων, η αναδίπλωση κώδικα και τα προσαρμόσιμα αποσπάσματα είναι μόνο μερικές από τις λειτουργίες που περιλαμβάνονται σε αυτό το βασικό πακέτο [12]. Περιλαμβάνει επίσης IntelliSense για JavaScript, TypeScript, JSON, CSS, HTML, καθώς και υποστήριξη για εντοπισμό σφαλμάτων που αφορά το Node.js [12]. Με την περιήγηση στο VS Code Marketplace ο χρήστης μπορεί να κατεβάσει επιπλέον πληθώρα επεκτάσεων, τις οποίες μπορεί να χρησιμοποιήσει για υποστήριξη σε επιπλέον διάφορες γλώσσες προγραμματισμού. Η διαδικασία αυτή διατίθεται δωρεάν [9]. Οι επεκτάσεις που προσφέρει είναι διαθέσιμες μέσω ενός κεντρικού αποθετηρίου και μπορούν να χρησιμοποιηθούν για την επέκταση του προγράμματος. Επιπροσθέτως, χάρη στο πρωτόκολλο Language Server Protocol είναι δυνατή η υποστήριξη νέων γλωσσών, θεμάτων και προγραμμάτων εντοπισμού σφαλμάτων. Παράλληλα, δίδεται η δυνατότητα για προσθήκη linters και για στατική ανάλυση κώδικα [12].



Εικόνα 4.1 Το VS Code Marketplace

Προχωρώντας την ανάλυση, το VS Code βασίζεται στο πλαίσιο Electron, το οποίο χρησιμοποιείται για τη δημιουργία web εφαρμογών με node.js. Οι εφαρμογές αυτές χρησιμοποιούν τη μηχανή διάταξης Blink. Επιπροσθέτως, είναι αξιοσημείωτο ότι το ίδιο συστατικό επεξεργαστή (με την κωδική ονομασία “Monaco”) που χρησιμοποιείται στο Azure DevOps, χρησιμοποιείται και στο VS Code [12]. Ένα άλλο χαρακτηριστικό του είναι ότι αντί για σύστημα έργων, επιτρέπει στους χρήστες να ανοίγουν έναν ή περισσότερους καταλόγους, οι οποίοι μπορούν στη συνέχεια να αποθηκεύονται ως χώροι εργασίας για μεταγενέστερη χρήση [9]. Ως αποτέλεσμα, μπορεί να χρησιμοποιηθεί σαν ένας επεξεργαστής κώδικα, ο οποίος είναι ανεξάρτητος από τη γλώσσα προγραμματισμού που προτιμήθηκε. Υποστηρίζει μια ποικιλία γλωσσών, όπου η καθεμία διαθέτει το δικό της σύνολο χαρακτηριστικών. Ακόμη, έχει τη δυνατότητα να εξαιρεί τα ανεπιθύμητα αρχεία και καταλόγους από την αναζήτηση settings project tree, ενώ μέσω της παλέτας εντολών παρέχεται πρόσβαση σε πολλές λειτουργίες οι οποίες δεν είναι προσβάσιμες μέσω των μενού ή του περιβάλλοντος εργασίας χρήστη [12]. Τέλος, μπορεί να διαπιστώσει κανείς ότι έχει τον έλεγχο πηγής ως προεπιλεγμένη λειτουργία. Παρόλα αυτά δύναται να αποκτήσει πρόσβαση στις ρυθμίσεις ελέγχου έκδοσης και να του προβληθούν οι πιθανές αλλαγές μέσω μιας ειδικής καρτέλας στη γραμμή μενού [12]. Για να μπορέσει

να επιτευχθεί αυτή η λειτουργία, είναι απαραίτητη η σύνδεση με ένα σύστημα ελέγχου και με έκδοση που το υποστηρίζει (Git, Apache, Subversion, Perforce κλπ.). Έτσι λοιπόν, ο χρήστης θα είναι πλέον σε θέση να δημιουργεί αποθετήρια και να κάνει αιτήσεις push και pull απευθείας από το περιβάλλον του VS Code [10], [12].

4.3 Το προγραμματιστικό πλαίσιο ReactJS

Η τεχνολογία που χρησιμοποιήθηκε για την υλοποίηση του προγραμματιστικού μέρους της εφαρμογής είναι η ReactJS.

4.3.1 Ιστορία

Η React (γνωστή είτε ως React.js είτε ReactJS) είναι μια δωρεάν και ανοικτού κώδικα front-end JavaScript βιβλιοθήκη και χρησιμοποιείται για τη δημιουργία διεπαφών χρήστη (UI) [14]. Σχεδιάστηκε από τον Jordan Walke, μηχανικό λογισμικού του Facebook, αρχικά ως “FaxJS”. Σήμερα, συντηρείται από την εταιρεία του Facebook (Meta) [13], [16]. Η ιστορική της εξέλιξη έχει ως αφετηρία το 2011, όπου χρησιμοποιήθηκε αρχικά στο News Feed του Facebook και στη συνέχεια, το 2012, στο Instagram. Τον Μάιο του 2013 μετατράπηκε σε βιβλιοθήκη ανοικτού κώδικα στο JSConf US και είχε ως άδεια χρήσης την Apache License 2.0 [13]. Ακολούθως, τον Οκτώβριο του 2014 λανσαρίστηκε η έκδοση 0.12.0 της React, η οποία έκανε update στην προηγούμενη και χρησιμοποίησε την άδεια BSD με τρεις ρήτρες, καθώς και με ένα ξεχωριστό αρχείο κειμένου PATENTS, το οποίο επιτρέπει την χρήση οποιονδήποτε πατεντών του Facebook που συνδέονται με το λογισμικό [13]. Στη συνέχεια, δημιουργήθηκε η React Native, η οποία επιτρέπει τον εγγενή προγραμματισμό σε Android, iOS με τη χρήση της React. Παρουσιάστηκε για πρώτη φορά τον Φεβρουάριο του 2015 στο React Conf του Facebook, ενώ η επίσημη κυκλοφορία της ξεκίνησε ένα μήνα μετά [13]. Δύο χρόνια μετά και συγκεκριμένα στις 18 Απριλίου του 2017, ανακοινώθηκε η React Fiber. Πρόκειται για ένα νέο σύνολο εσωτερικών αλγορίθμων απόδοσης, το οποίο αντικαθιστά τον ήδη υπάρχον της React ονόματι Stack. Η κυκλοφορία της React Fiber αποτέλεσε τροχοπέδη για τις μελλοντικές αναβαθμίσεις της βιβλιοθήκης React, καθώς και για την ανάπτυξη νέων χαρακτηριστικών, αλλάζοντας έτσι τον τρόπο εκτέλεσης [13]. Στις 23 Σεπτεμβρίου του 2017, το Facebook ανακοίνωσε ότι τα Flow, Jest, React και Immutable.js θα έπαιρναν κανονική άδεια MIT την επόμενη βδομάδα, ενώ σημείωσε ότι η React ήταν “ο ακρογωνιαίος λίθος” του έργου [13]. Η προηγούμενη τεχνολογία απόδοσης της React, το Stack, δημιουργήθηκε σε μια περίοδο κατά την οποία η εστίαση του συστήματος στη δυναμική αλλαγή δεν ήταν πλήρως κατανοητή. Για παράδειγμα, στην προσπάθεια για σχεδίαση εξελιγμένων κινουμένων σχεδίων, το Stack ήταν αργό, καθώς προσπαθούσε να κάνει τα πάντα με μία κίνηση. Λύση σε αυτό έδωσε η React Fiber που προαναφέρθηκε. Ουσιαστικά πρόκειται για συναρτήσεις JavaScript και εικονικά DOM αντικείμενα, τα οποία έχουν την ικανότητα να λειτουργούν και να ενημερώνονται ανεξάρτητα, επιτρέποντας με αυτόν τον τρόπο την ομαλότερη απόδοση στην οθόνη. Η Fiber χωρίζει το animation σε μέρη που μπορούν να απλωθούν σε πολλά καρέ [13]. Η δομή μιας σελίδας μπορεί επίσης να χωριστεί σε τμήματα που μπορούν να αλλάξουν και να διατηρηθούν ξεχωριστά. Φτάνοντας στις 16 Φεβρουαρίου του 2019, η React 16.8, έχοντας και την άδεια MIT, ήταν η έκδοση που διαδέχθηκε τις προηγούμενες και κυκλοφόρησε στο κοινό. Η πιο σημαντική προσθήκη που είχε αφορούσε τα React Hooks [13]. Η ομάδα της React ανακοίνωσε την πρώτη υποψήφια έκδοση για την v17.0 στις 10 Αυγούστου του 2020, σηματοδοτώντας έτσι την πρώτη μεγάλη έκδοση χωρίς σημαντικές αλλαγές στο API της.

4.3.2 Χρήση και λειτουργία

Η ReactJS μπορεί να χρησιμοποιηθεί σε συνεργασία και με άλλα frameworks, όπως το next.js [14]. Συνήθως, η δημιουργία εφαρμογών με τη χρήση της React απαιτεί τη χρήση πρόσθετων βιβλιοθηκών για τη δρομολόγηση καθώς και ορισμένων λειτουργιών στην πλευρά του πελάτη [14]. Το μοντέλο αρχιτεκτονικής που ακολουθεί είναι το MVC (Model View Controller) [8]. Χρησιμοποιεί το δηλωτικό παράδειγμα προγραμματισμού. Ασχολείται δηλαδή με τη διαχείριση του state, μιας μεταβλητής η οποία παρακολουθεί τα δεδομένα όταν αυτά αλλάζουν. Η αλλαγή αυτή του state απεικονίζεται στο DOM με την ενημέρωση των απαραίτητων στοιχείων. Η διαδικασία αυτή αντιτίθεται στον προστακτικό προγραμματισμό. Όσον αφορά το κομμάτι του κώδικα, αποτελείται από οντότητες γνωστές ως συστατικά (components), τα οποία είναι επαναχρησιμοποιήσιμα και πρέπει να δημιουργηθούν στον φάκελο src. Κατά τη δημιουργία τους συνηθίζεται να ακολουθείται η Pascal-Case ονοματοδοσία, ενώ στα στιγμιότυπα συνίσταται η Camel-Case [14]. Έτσι λοιπόν, χάρη στα χαρακτηριστικά και στον τρόπο λειτουργίας της γλώσσας, οι προγραμματιστές μπορούν να δημιουργήσουν περιβάλλοντα διεπαφής χρήστη, τα οποία εμφανίζουν τα επιθυμητά αποτελέσματα χωρίς να χρειαστεί να ανανεωθεί η σελίδα ή να φορτωθεί κάποια νέα [13], [14]. Προχωρώντας, το εικονικό μοντέλο αντικειμένου εγγράφου (Virtual DOM) είναι ένα ακόμη αξιοσημείωτο χαρακτηριστικό. Η λειτουργία του είναι η εξής: Η React δημιουργεί μια κρυφή μνήμη δομής δεδομένων, υπολογίζει τις διαφορές και στη συνέχεια ενημερώνει αποτελεσματικά το εμφανιζόμενο DOM του προγράμματος περιήγησης [14]. Η διαδικασία αυτή είναι γνωστή ως συμπίλιωση και επιτρέπει στον προγραμματιστή να γράφει κώδικα σαν να απεικονίζεται ολόκληρη η σελίδα σε κάθε αλλαγή, ενώ οι βιβλιοθήκες της React απεικονίζουν μόνο τα στοιχεία που αλλάζουν. Το γεγονός αυτό εξαλείφει την ανάγκη για επαναυπολογισμό του CSS, της διάταξης της σελίδας και της απόδοσης για ολόκληρη τη σελίδα [11]. Με τη χρήση της βιβλιοθήκης React DOM, τα συστατικά μπορούν να προσαρτηθούν σε ένα συγκεκριμένο στοιχείο στο DOM. Κατά την απόδοση ενός συστατικού χρησιμοποιούνται κάποια special objects για να περάσουν τιμές από το ένα στο άλλο. Τα αντικείμενα αυτά ονομάζονται props [14]. Τα props μπορούν να λειτουργήσουν για την απόδοση τιμής μόνο από το parent προς το child component κι όχι το αντίστροφο. Στην παρακάτω εικόνα φαίνεται ένα παράδειγμα χρήσης του props, όπου χάρη στη χρήση του περνάει η μεταβλητή opt στη συνάρτηση main.

```
export default function Main (props) {  
  
  const {opt} = props
```

Εικόνα 4.2 Χρήση του object props

Πριν από το κλείσιμο οποιασδήποτε συνάρτησης χρειάζεται το return. Χάρη σε αυτό καθορίζεται η προβολή των αποτελεσμάτων των εντολών του κώδικα. Επειδή όμως το return μπορεί να επιστρέφει μόνο μία τιμή, χρησιμοποιείται μία ενιαία ετικέτα (tag), η οποία περιλαμβάνει όλα τα συστατικά [14].

```

return (
  <Container className='App mb-4' style={{ backgroundColor: 'lightblue' }}>
    <div style={{ position: 'static', textAlign: 'center', fontSize: '22px' }}>Επιλέξτε κύκλωμα:</div>
    <div className='d-flex mb-2' style={{ overflowX: 'scroll' }}>
      {components.map(($, idx) => (
        <div style={{ display: 'inline-block', marginLeft: '1rem' }} key=${$.image}>
          <img
            style={{
              maxHeight: '10rem',
              padding: '1rem',
              borderStyle: 'outset', borderWidth: '7px', borderBottom: '5px solid grey',
              margin: '15px', padding: '15px',
              borderBottomColor: 'black', borderLeftColor: 'black', borderRightColor: 'black', borderTopColor: 'black',
              borderColor: ((idx === key) ? 'blue' : 'black')
            }}
            src=${$.image}
            alt=""
            onClick={() => setKey(idx)}
          />
        </div>
      ))}
    </div>
    <Row>
      <Col xs=12>
        <ComponentFromArray array={components} idx={key} />
      </Col>
    </Row>
  </Container>
)

```

Εικόνα 4.3 Χρήση του return

Ο τύπος αρχείου που χρησιμοποιείται ονομάζεται JSX (Javascript Syntax Extension) [14]. Το JSX είναι ένα πλαίσιο απόδοσης στοιχείων που μοιάζει με την HTML και χρησιμοποιεί σύνταξη με την οποία είναι εξοικειωμένοι πολλοί προγραμματιστές. Ουσιαστικά αυτό που κάνει είναι να μετατρέπει κώδικα μορφής HTML σε αντικείμενα JavaScript προς ανάλυση. Αν και δεν απαιτείται, τα συστατικά της React γράφονται συχνά σε JSX (μπορούν επίσης να γραφούν και σε plain JavaScript) [13], [15]. Το JSX σχετίζεται με τη σύνταξη επέκτασης XHP για την PHP, η οποία σχεδιάστηκε από το Facebook [15].

4.4 Επίλογος

Εν κατακλείδι, στο τέταρτο κεφάλαιο γίνεται μια ανασκόπηση της λειτουργίας του περιβάλλοντος VSCode, το οποίο χρησιμοποιείται για την υλοποίηση του προγραμματιστικού μέρους της εργασίας, καθώς και του framework της JavaScript, ReactJS, όπου είναι η γλώσσα που επιλέχθηκε. Οι λόγοι επιλογής της React για την ανάπτυξη της εφαρμογής μπορούν να συνοψιστούν ως εξής:

- Προσφέρει μεγαλύτερη ευχρηστία λόγω της component-based αρχιτεκτονικής της [8].
- Το Virtual DOM ενημερώνει μόνο τα στοιχεία τα οποία έχουν αλλάξει και δεν φορτώνει από την αρχή ολόκληρη τη σελίδα [8].
- Μονόδρομη ροή δεδομένων [8].
- Δηλωτικό UI [8].
- Δυνατότητα για επαναχρησιμοποίηση και συντήρηση κώδικα [8], κάτι που μπορεί να αποδειχθεί εξαιρετικά χρήσιμο για πιθανές μελλοντικές επεκτάσεις/βελτιώσεις της εφαρμογής.

Κεφάλαιο 5ο: Ανάλυση της εφαρμογής

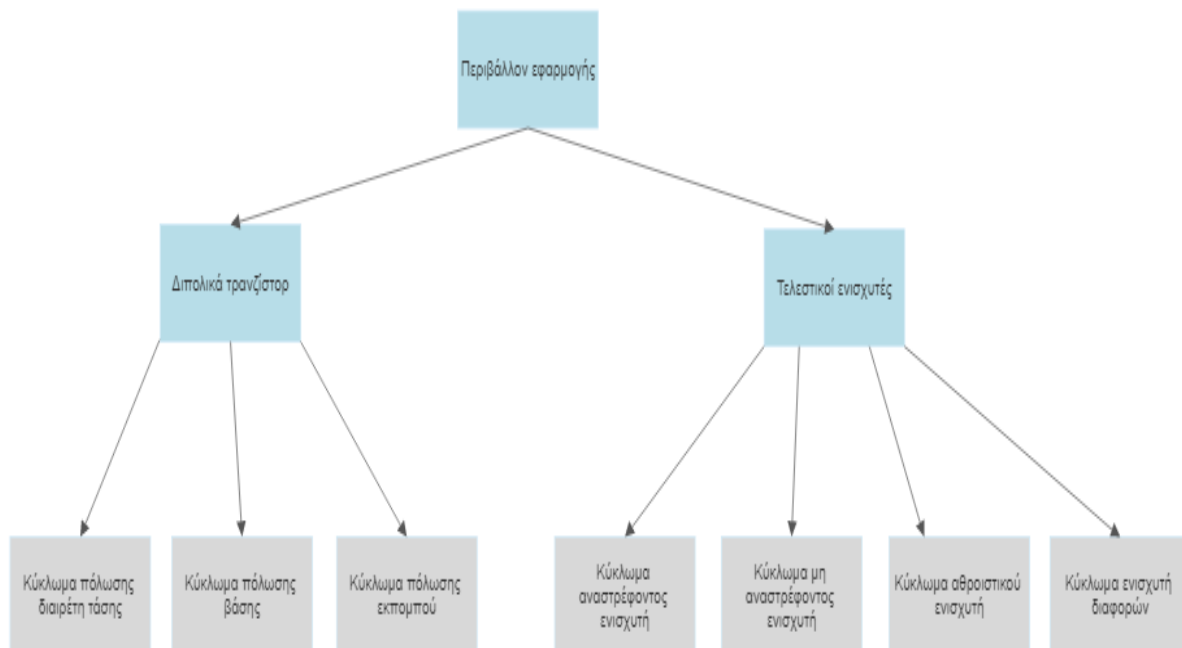
5.1 Εισαγωγή

Σε αυτό το κεφάλαιο πρόκειται να αναλυθεί εκτενώς η εφαρμογή. Ξεκινώντας από την απεικόνιση της δομής και των επιμέρους κυκλωμάτων που επιλέχθηκαν, φτάνουμε μέχρι την ανάλυση των λειτουργιών και του κώδικα που τις επιτελεί και περιλαμβάνεται μέσα στα αρχεία της εφαρμογής. Όπως αναφέρθηκε σε προγενέστερο κεφάλαιο, η εφαρμογή είναι καθαρά front-end και γι' αυτό το λόγο χρησιμοποιήθηκε η react. Οπότε δεν θα υπάρχει κάποιο κεφάλαιο για ανάλυση back-end τεχνολογιών. Ο γενικός σκοπός του προγράμματος είναι να δίνει τη δυνατότητα στο χρήστη να επιλέγει την ομάδα κυκλωμάτων και εν συνεχεία το κύκλωμα που επιθυμεί, να εκχωρεί τις επιθυμητές τιμές για τις μεταβλητές του κυκλώματος που επέλεξε και με βάση αυτές, να εμφανίζονται οι κατάλληλοι υπολογισμοί ή/και τα γραφήματα.

5.2 Διαγραμματική απεικόνιση της εφαρμογής

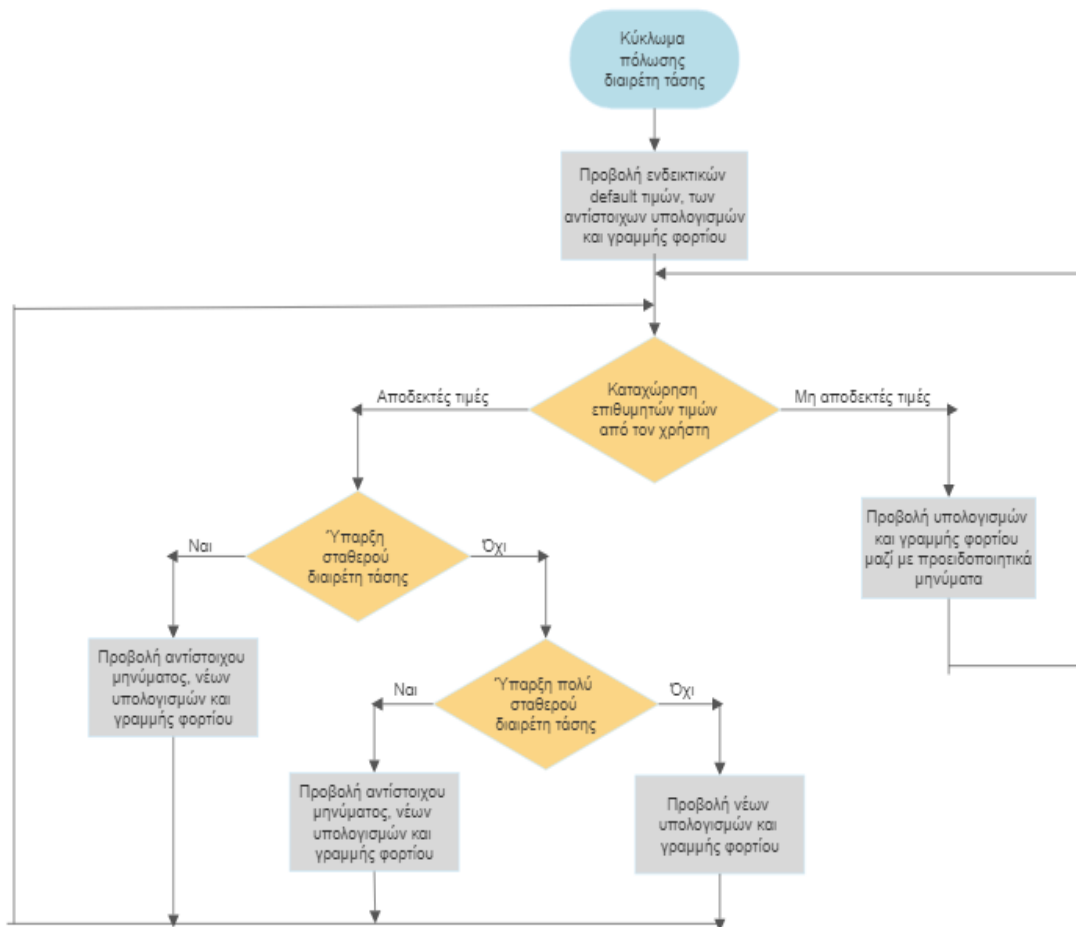
5.2.1 Γενική δομή και επιμέρους κυκλώματα

Πριν από οτιδήποτε άλλο, είναι αναγκαίο να παρουσιαστεί η ιεραρχία των επιλογών που έχει ο χρήστης της εφαρμογής. Ο τρόπος που επιλέχθηκε για να γίνει αυτό είναι με τη χρήση διαγραμμάτων ροής, τα οποία υποδεικνύουν τη χρήση των λειτουργιών που περιλαμβάνονται. Η ανάγνωση και παρατήρησή τους κρίνεται χρήσιμη για την ορθή πλοήγηση του χρήστη στα τεκταινόμενα της εφαρμογής. Η διαδικασία δημιουργίας των διαγραμμάτων έγινε με τη βοήθεια του online editor smartdraw. Πριν την παρουσίαση των επιμέρους διαγραμμάτων, είναι σκόπιμο να απεικονιστεί ένα συνολικό διάγραμμα, το οποίο δείχνει τα επιμέρους κυκλώματα των δύο κατηγοριών κυκλωμάτων.



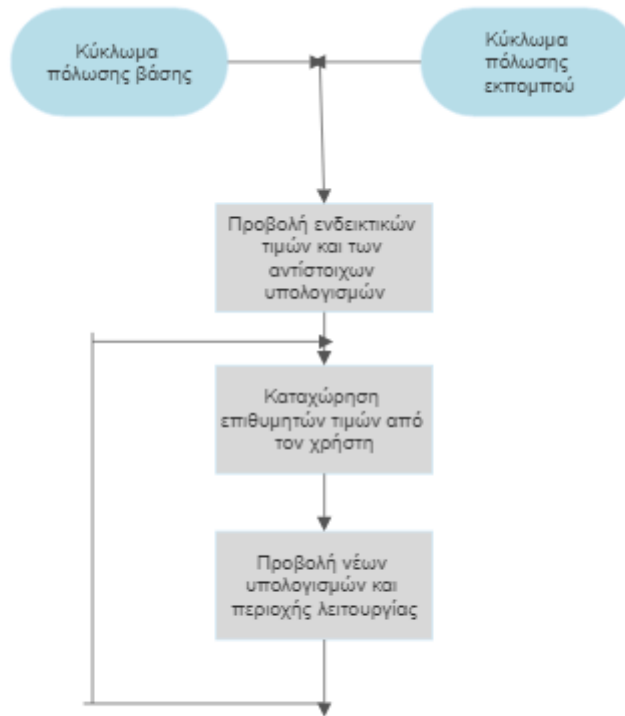
Εικόνα 5.1 Γενική δομή εφαρμογής

Κατόπιν, ακολουθούν τα επιμέρους διαγράμματα. Το πρώτο διάγραμμα ροής αφορά το κύκλωμα πόλωσης διαιρέτη τάσης και φαίνεται στο παρακάτω σχήμα:



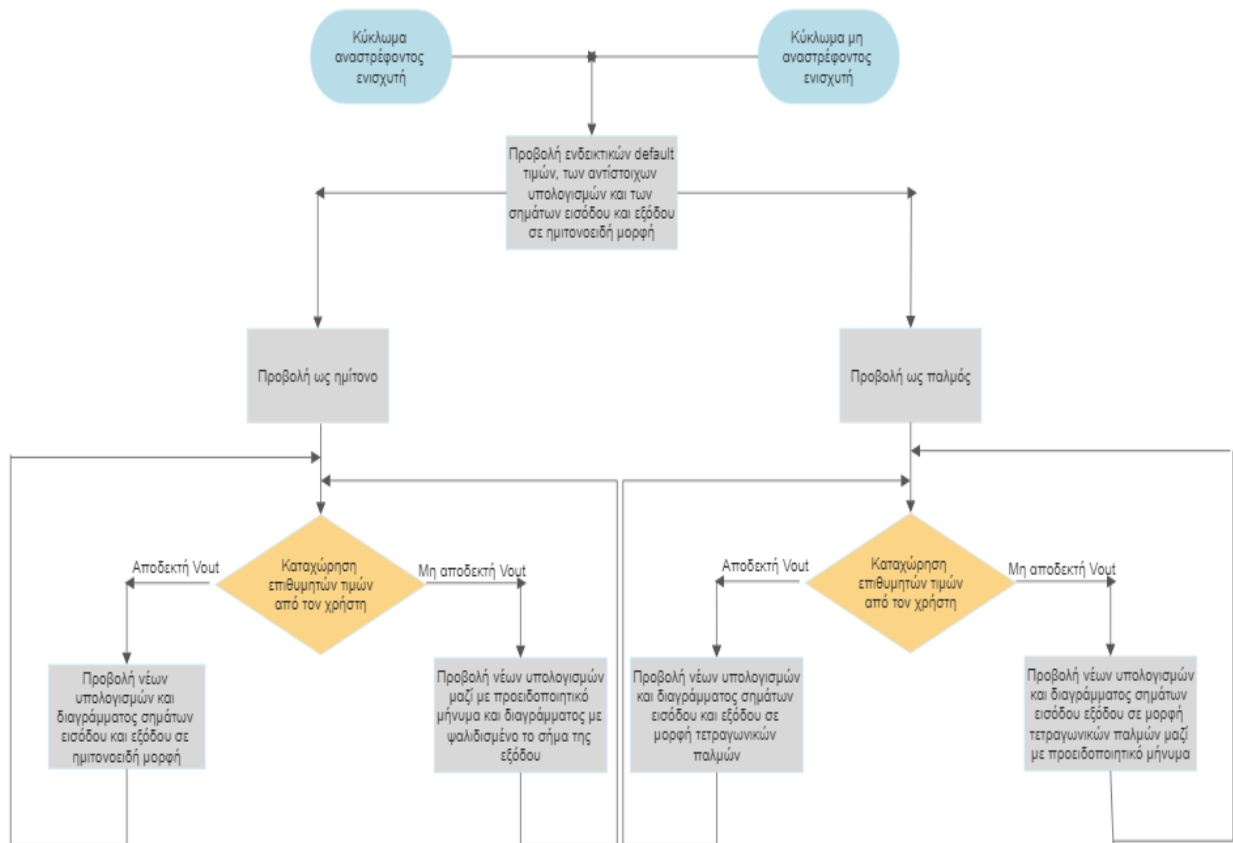
Εικόνα 5.2 Διάγραμμα ροής για το κύκλωμα πόλωσης διαιρέτη τάσης

Επιλέγοντας ο χρήστης το συγκεκριμένο κύκλωμα, βλέπει έτοιμα συμπληρωμένα τα πεδία με default τιμές, τους υπολογισμούς που προκύπτουν από αυτές και το αντίστοιχο διάγραμμα με τη γραμμή φορτίου. Αφού καταχωρήσει τις τιμές που αυτός επιθυμεί, προβάλλονται τα ανάλογα αποτελέσματα, τα οποία χωρίζονται σε τέσσερις περιπτώσεις α) εμφάνιση των νέων υπολογισμών μαζί με το μήνυμα για ύπαρξη σταθερού διαιρέτη τάσης, β) εμφάνιση των νέων υπολογισμών μαζί με το μήνυμα για ύπαρξη πολύ σταθερού διαιρέτη τάσης, γ) εμφάνιση των νέων υπολογισμών χωρίς κάποιο μήνυμα και δ) εμφάνιση των νέων υπολογισμών μαζί με προειδοποιητικά μηνύματα που υποδεικνύουν στο χρήστη ποιο είναι το λάθος.



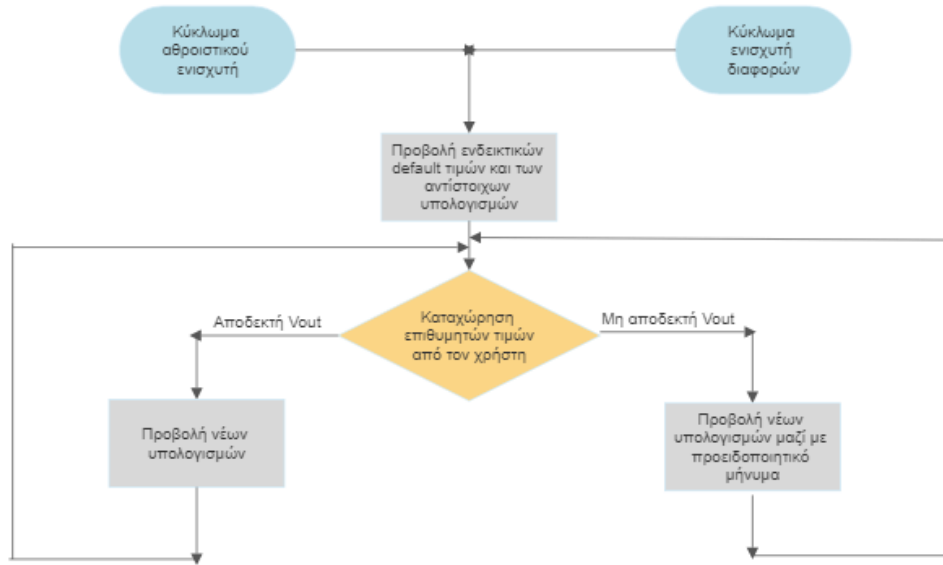
Εικόνα 5.3 Διάγραμμα ροής για τα κυκλώματα πόλωσης βάσης και πόλωσης εκπομπού

Επόμενο διάγραμμα είναι αυτό των κυκλωμάτων πόλωσης βάσης και πόλωσης εκπομπού. Η λειτουργία τους είναι η ίδια και γι' αυτό το λόγο απεικονίζονται σε ένα ενιαίο σχήμα. Όπως και προηγουμένως, προβάλλονται default τιμές στα πεδία με τις παραμέτρους των κυκλωμάτων και εμφανίζονται οι αντίστοιχοι υπολογισμοί. Ακολούθως, ο χρήστης καταχωρεί τις επιθυμητές τιμές και εμφανίζονται τα νέα αποτελέσματα. Στα συγκεκριμένα κυκλώματα δεν προϋποθέτεται η εμφάνιση μηνυμάτων.



Εικόνα 5.4 Διάγραμμα ροής για τα κυκλώματα αναστρέφοντος και μη αναστρέφοντος ενισχυτών

Όπως και στα κυκλώματα πόλωσης βάσης και εκπομπού των διπολικών τρανζίστορ, έτσι και σε αυτά του αναστρέφοντος και μη αναστρέφοντος τελεστικού ενισχυτή, το διάγραμμα ροής είναι κοινό, λόγω της παρόμοιας λειτουργίας τους. Αφού επιλεγεί το κύκλωμα και προβληθούν οι default τιμές με τους υπολογισμούς και το αντίστοιχο διάγραμμα με τις καμπύλες εισόδου εξόδου σε ημιτονοειδή μορφή, ο χρήστης έχει αρχικά τη δυνατότητα να επιλέξει μεταξύ της προβολής των σημάτων είτε ως ημίτονα είτε ως τετραγωνικοί παλμοί. Στην πρώτη περίπτωση και μετά την εκχώρηση των τιμών από τον χρήστη, θα εμφανιστούν οι νέοι υπολογισμοί μαζί με το διάγραμμα για τα σήματα εισόδου και εξόδου, όπου η έξοδος θα εμφανίζεται είτε σε κανονική ημιτονοειδή μορφή στην περίπτωση που στα αποτελέσματα προκύψει V_{out} εντός των αποδεκτών ορίων (λόγω τροφοδοσίας του τελεστικού ενισχυτή), είτε ψαλιδισμένη, όταν υπερβαίνει την δοθείσα τάση τροφοδοσίας. Η τελευταία περίπτωση συνοδεύεται από προειδοποιητικό μήνυμα για την V_{out} . Ακριβώς η ίδια διαδικασία ακολουθείται και στην προβολή των σημάτων ως παλμοί, με μόνη διαφορά αυτή του μη ψαλιδισμού του σήματος εξόδου σε περίπτωση που η V_{out} υπερβαίνει την τάση τροφοδοσίας, καθώς πρόκειται για τετραγωνικό παλμό και όχι για κάποια ημιτονοειδή καμπύλη.

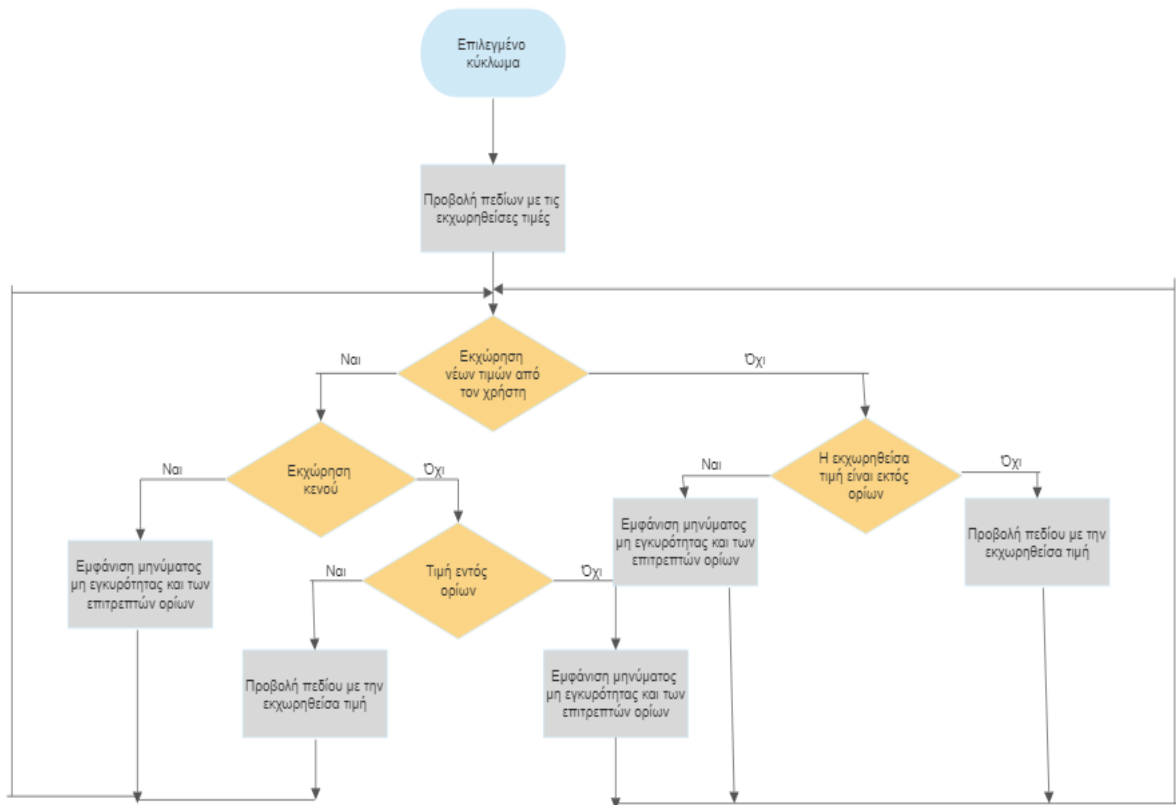


Εικόνα 5.5 Διάγραμμα ροής για τα κυκλώματα αθροιστικού και ενισχυτή διαφορών

Τελευταία περίπτωση είναι ο χρήστης να επιλέξει είτε το κύκλωμα του αθροιστικού ενισχυτή, είτε του ενισχυτή διαφορών. Λόγω της απουσίας ουσιαστικών διαφορών στις δυνατές λειτουργίες που παρέχουν, εμφανίζονται όπως και η πλειοψηφία των κυκλωμάτων παραπάνω, στο ίδιο διάγραμμα. Η λειτουργία τους δεν διαφέρει και πολύ από τα προηγούμενα κυκλώματα καθώς, μετά την προβολή των default τιμών και υπολογισμών, ο χρήστης εκχωρεί τις τιμές που θέλει και ανάλογα με την τιμή της V_{out} , εμφανίζονται οι νέοι υπολογισμοί, συνοδευόμενοι από το ανάλογο προειδοποιητικό μήνυμα που εμφανίζεται όταν αυτή έχει μη αποδεκτή τιμή (εκτός των ορίων της τάσης τροφοδοσίας).

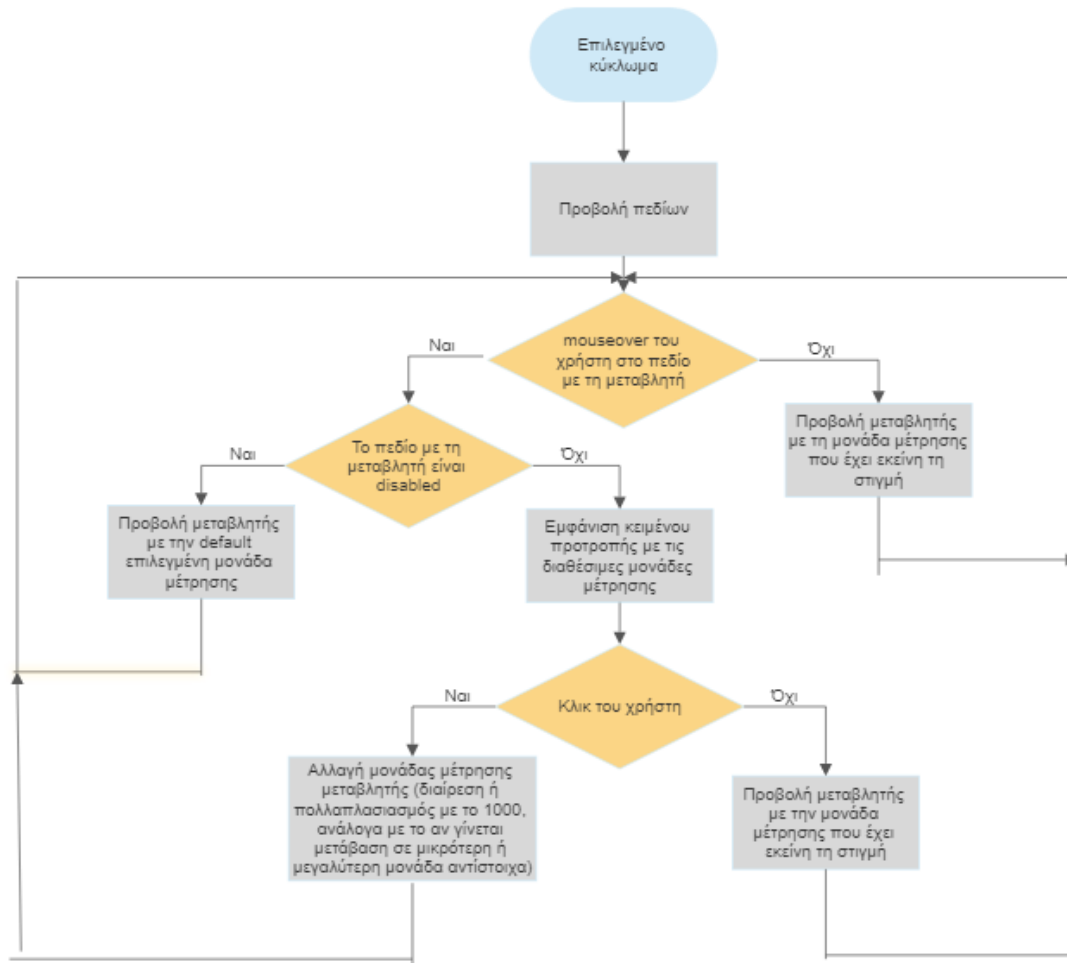
5.2.2 Επιμέρους λειτουργίες

Με την περιήγησή του στα κυκλώματα της εφαρμογής, ο χρήστης δύναται να εντοπίσει τις επιμέρους λειτουργίες που υπάρχουν. Τα διαγράμματα ροής που δίνουν μια εικόνα για αυτές είναι:



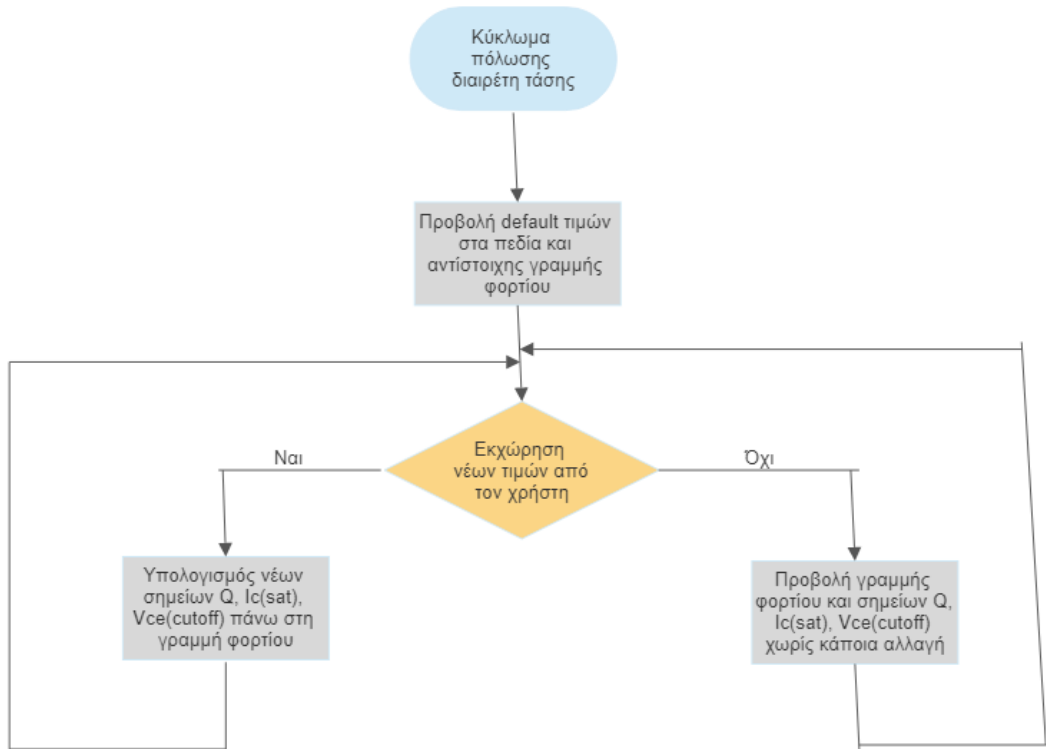
Εικόνα 5.6 Διάγραμμα ροής ελέγχου τιμών

Στο παραπάνω διάγραμμα απεικονίζεται η διαδικασία η οποία ακολουθείται με την εκχώρηση τιμών στα πεδία. Πιο συγκεκριμένα, διακρίνονται 3 περιπτώσεις: α) Ο χρήστης να εισάγει το κενό, όπου και θα εμφανιστεί μήνυμα μη εγκυρότητας, συνοδευόμενο από τα όρια μέσα στα οποία πρέπει να βρίσκεται η τιμή που θα καταχωρήσει. β) Ο χρήστης να εισάγει μία τιμή η οποία είναι εκτός των ορίων που προβλέπονται, οπότε και θα προβληθεί πάλι η παραπάνω ειδοποίηση και γ) Ο χρήστης να εισάγει μια αποδεκτή τιμή με την οποία εξασφαλίζεται η ορθή προβολή του πεδίου χωρίς κάποιο μήνυμα.



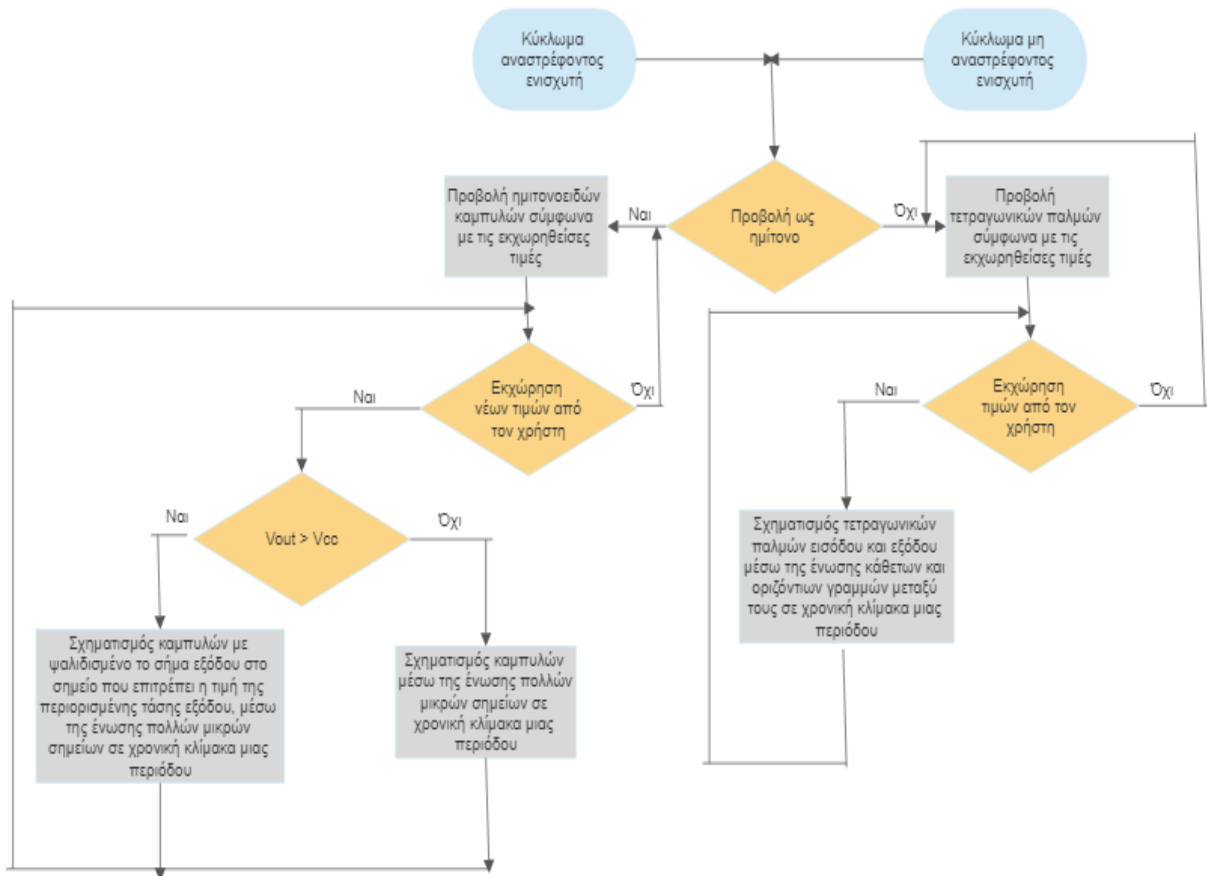
Εικόνα 5.7 Διάγραμμα ροής μετατροπής μονάδων μέτρησης

Επόμενη σημαντική λειτουργία που παρέχεται είναι αυτή της μετατροπής των μονάδων μέτρησης. Σε αυτή την περίπτωση, όταν ο χρήστης περάσει με τον κέρσορα πάνω από το πεδίο (mouseover) θα εμφανιστούν οι διαθέσιμες μονάδες μέτρησης της μεταβλητής. Στην περίπτωση που το πεδίο δεν είναι disabled, καθώς τότε δεν υπάρχει η δυνατότητα για κάποια ενέργεια, κάνοντας κλικ πάνω στο πεδίο, αλλάζει η μονάδα μέτρησης της μεταβλητής (γίνεται διαίρεση ή πολλαπλασιασμός με το 1000, ανάλογα με το αν μεταβαίνουμε σε μικρότερη ή μεγαλύτερη μονάδα).



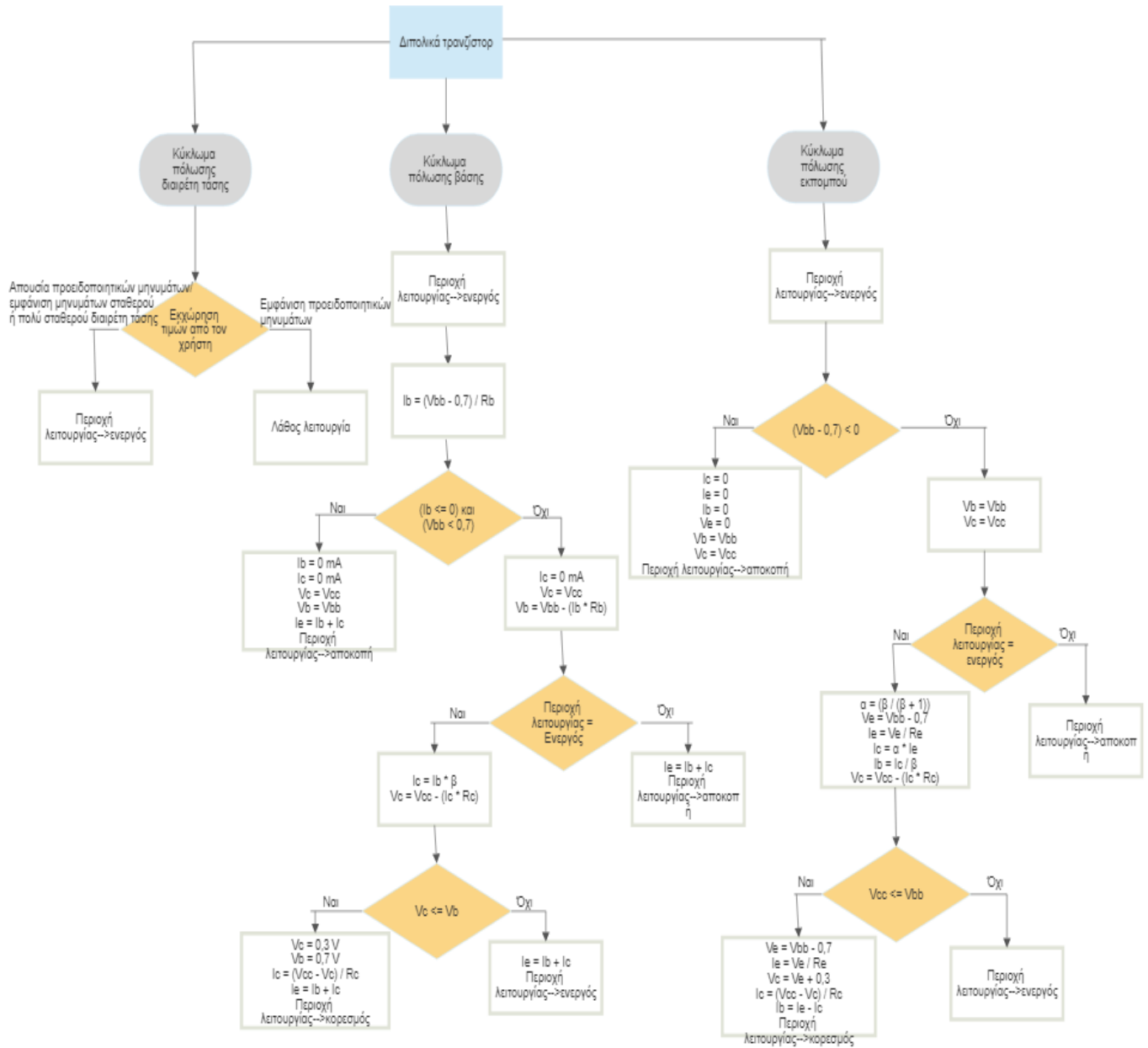
Εικόνα 5.8 Διάγραμμα ροής σχεδίασης γραμμής φορτίου

Προχωρώντας, κάτι το οποίο συναντάται στο κύκλωμα πόλωσης διαιρέτη τάσης είναι η σχεδίαση της γραμμής φορτίου. Εδώ πέρα αυτό που γίνεται, είναι μετά από κάθε καταχώρηση τιμών από τον χρήστη, να υπολογίζονται τα νέα σημεία ηρεμίας Q , ρεύματος κορεσμού $I_{C(SAT)}$ και τάσης αποκοπής $V_{CE(CUTOFF)}$ και να τοποθετούνται εκ νέου πάνω στη γραμμή φορτίου.



Εικόνα 5.9 Διάγραμμα ροής σχεδίασης ημιτονοειδών κυματομορφών και τετραγωνικών παλμών

Μια λειτουργία την οποία μπορεί να διαπιστώσει ο χρήστης όταν περιηγείται στα κυκλώματα του αναστρέφοντος και μη αναστρέφοντος τελεστικού ενισχυτή είναι αυτή της σχεδίασης των σημάτων εισόδου και εξόδου είτε σε ημιτονοειδή είτε σε μορφή τετραγωνικών παλμών. Αναλυτικότερα, στην περίπτωση που ο χρήστης επιλέξει την προβολή ως ημίτονο και εισάγει τις επιθυμητές τιμές, θα δει στην οθόνη να σχηματίζονται οι καμπύλες για τα σήματα εισόδου και εξόδου του ενισχυτή, με αυτό της εξόδου να εμφανίζεται ψαλιδισμένο στην περίπτωση που η τάση εξόδου υπερβεί την τάση τροφοδοσίας. Εναλλακτικά, στην προβολή ως παλμό, απεικονίζονται τα σήματα εισόδου και εξόδου σε μορφή τετραγωνικών παλμών, με τη βοήθεια του Δt για αυτό της εξόδου. Και στις δύο περιπτώσεις, η προβολή γίνεται σε χρονική κλίμακα μιας περιόδου, η οποία προκύπτει από την συχνότητα f την οποία έχει επιλέξει ο χρήστης.

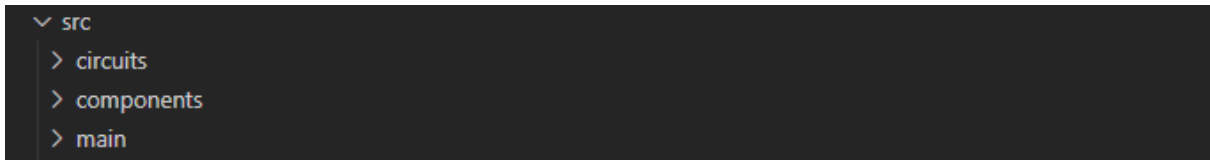


Εικόνα 5.10 Διάγραμμα ροής εμφάνισης περιοχής λειτουργίας

Τελευταίο διάγραμμα παρουσίασης των σημαντικότερων λειτουργιών της εφαρμογής είναι αυτό της εμφάνισης της περιοχής λειτουργίας, στην οποία βρίσκεται το κύκλωμα με τα δεδομένα που έχει την τρέχουσα χρονική στιγμή. Η απεικόνισή του διαφέρει λίγο από τα προηγούμενα, καθώς εμφανίζει σε κάθε περίπτωση τις τιμές για τις τάσεις και τα ρεύματα του κυκλώματος, αλλά και τους ελέγχους, μέσα από τους οποίους προκύπτει το συμπέρασμα για λειτουργία σε κάποια συγκεκριμένη περιοχή.

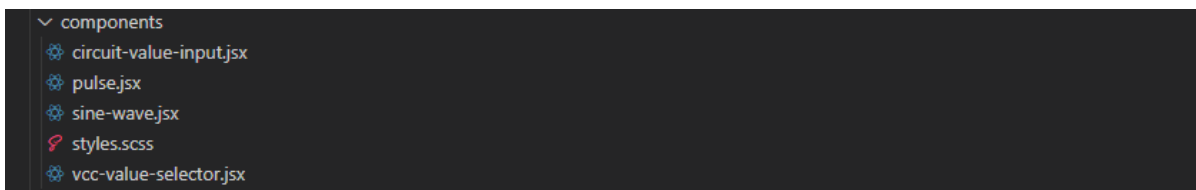
5.3 Ανάλυση της δομής του κώδικα

Προχωρώντας τη διαδικασία ανάπτυξης της εφαρμογής, είναι χρήσιμο να αναλυθεί η δομή που σχηματίζεται στο περιβάλλον εργασίας. Ανοίγοντας το VS Code, ο βασικός κώδικας βρίσκεται στον φάκελο `src`. Πιο συγκεκριμένα, περιλαμβάνονται τρεις υποφακέλοι: Ο φάκελος με τα `circuits`, ο φάκελος με τα `components` και ο φάκελος με το `main`.



Εικόνα 5.11 Ο φάκελος `src` και τα περιεχόμενά του

Στον φάκελο με τα `components` περιέχονται 5 επιμέρους αρχεία: Το `circuit-value-input.jsx`, το `pulse.jsx`, το `sine-wave.jsx`, το `styles.scss` και το `vcc-value-selector.jsx`. Η λειτουργία τους αφορά το χειρισμό των πεδίων και των μεταβλητών, καθώς και τον σχεδιασμό γραφημάτων. Η ανάλυσή τους θα πραγματοποιηθεί σε επόμενη ενότητα. Ο φάκελος `main` είναι αυτός που περιλαμβάνει τα αρχεία τα οποία χρειάζονται για την πρώτη αλληλεπίδραση του χρήστη με την εφαρμογή. Αναλυτικότερα, πρόκειται για τα αρχεία `main.jsx`, στο οποίο επιτελούνται κάποιες `css` λειτουργίες για την απεικόνιση των κυκλωμάτων, καθώς και χρήση κάποιων `props`, μέσω των οποίων γίνεται η μεταφορά πληροφοριών για την επιλογή του επιθυμητού κυκλώματος, και του `componentfromarray.jsx` που χρησιμοποιείται για να επιλεγεί το κύκλωμα που θέλει ο χρήστης, μέσω ενός `array` και ενός `index` που δέχεται από το `main.jsx`.



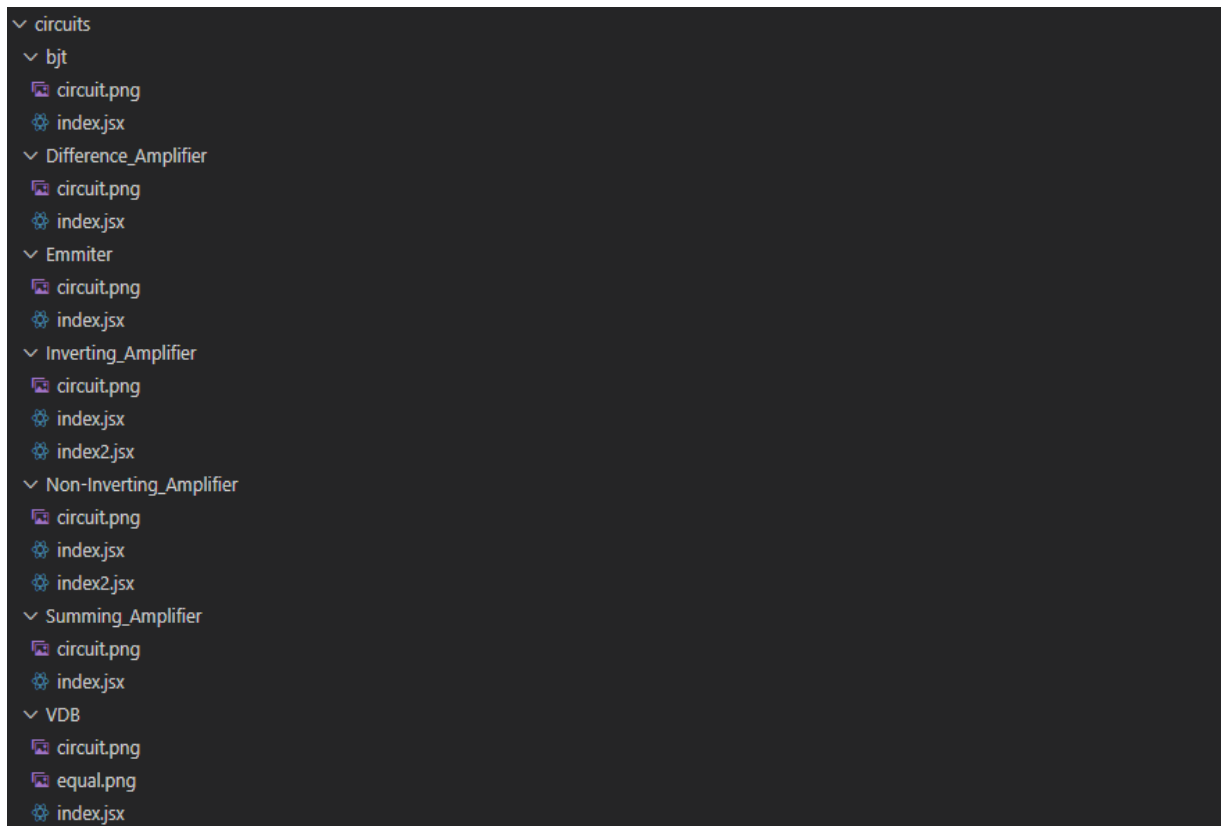
Εικόνα 5.12 Ο φάκελος `components` και τα περιεχόμενά του



Εικόνα 5.13 Ο φάκελος `main` και τα περιεχόμενά του

Ακολουθώντας, έχουμε τον φάκελο `circuits`. Μέσα σε αυτόν βρίσκονται όλα τα κυκλώματα της εφαρμογής. Κάθε επιμέρους φάκελος περιλαμβάνει ένα αρχείο `index.jsx`, μέσα στο οποίο βρίσκονται οι αρχικές `default` τιμές που φαίνονται στην οθόνη όταν ο χρήστης επιλέξει το εκάστοτε κύκλωμα, καθώς και πληροφορίες για τη μετατροπή των μονάδων μέτρησης, και το `circuit.png`, όπου είναι η εικόνα του κυκλώματος η οποία θα φαίνεται πάνω από κάθε ανάλυση κυκλώματος. Εξαιρέση αποτελούν οι φάκελοι `Inverting_Amplifier` και `Non-Inverting_Amplifier`, οι οποίοι περιλαμβάνουν εκτός από την εικόνα για το κύκλωμα, δύο αρχεία, ένα `index.jsx` και ένα `index2.jsx`. Ο λόγος είναι ότι στα συγκεκριμένα κυκλώματα θέλουμε ο χρήστης να έχει τη δυνατότητα να επιλέξει, αν εκτός από την ανάλυση του κυκλώματος και τους υπολογισμούς, επιθυμεί να εμφανίζεται γράφημα με την απεικόνιση ημιτόνων ή παλμών. Η λειτουργία αυτή επιτελείται μέσα από το `index.jsx`, όπου μέσω του `opt` στέλνεται στο `index2.jsx` η επιλογή. Στο `index.jsx` γίνεται επίσης ο χρωματισμός του `button` το

οποίο είναι επιλεγμένο ή επιλέγει ο χρήστης. Εξαίρεση στα περιεχόμενα του φακέλου `circuits` αποτελεί και ο φάκελος `VDB`, στον οποίο περιέχεται και μια επιπλέον εικόνα, αυτή του ισοδύναμου της πόλωσης διαιρέτη τάσης κυκλώματος. Η δομή του `circuits` και των υποφακέλων του φαίνεται στην παρακάτω εικόνα.



Εικόνα 5.14 Ο φάκελος `circuits` και τα περιεχόμενα των υποφακέλων του

Τα υπόλοιπα αρχεία τα οποία υπέστησαν επεξεργασία προκειμένου να επέλθει το τελικό αποτέλεσμα για την εμφάνιση και τη λειτουργία της εφαρμογής είναι τα `styles.css`, `index.css` και `App.js`, αρχείο το οποίο περιέχει τις πληροφορίες για την εκκίνηση και την πρώτη σελίδα η οποία θα εμφανιστεί.

5.4 Ανάλυση του κώδικα και των λειτουργιών

Πριν ξεκινήσει η ανάλυση, κρίνεται σκόπιμο να αναφερθεί το γεγονός της χρήσης `functional components` έναντι `class components` στην εφαρμογή. Τα `class components` είναι απλές κλάσεις οι οποίες συνθέτονται από πολλαπλές συναρτήσεις που μπορούν να προσδώσουν λειτουργικότητα σε μια εφαρμογή, την ίδια στιγμή που τα `functional components` είναι απλά JavaScript συναρτήσεις [17]. Ο λόγος προτίμησης `functional components` είναι για την ευκολότερη διαχείριση του `state` μέσα σε κάθε αρχείο, καθώς στα περισσότερα χρησιμοποιείται κάποια μεταβλητή η οποία διαχειρίζεται μια κατάσταση. Αυτό επιτυγχάνεται με τα `Hooks`, τα οποία μπορούν να χρησιμοποιηθούν μόνο μέσα σε `functional components` [17]. Γενικότερα τα `Hooks` είναι συναρτήσεις οι οποίες δίνουν τη δυνατότητα στο χρήστη να μπορεί να χειριστεί ένα `state`, καθώς και άλλα `features` όπως `lifecycle methods` κλπ [8]. Τα `Hooks` τα οποία χρησιμοποιούνται στην εφαρμογή είναι το `useState` και `useEffect` και η χρησιμότητά τους θα αναλυθεί παρακάτω, όπου αναλύεται ο κώδικας. Από εκεί και πέρα, οι πρώτες ενέργειες που έγιναν για να στηθεί η εφαρμογή στο περιβάλλον του VS Code είναι η εγκατάσταση του `node.js` και του `npm` (**n**ode.js **p**ackage **m**anager). Για να μπορέσουμε να δούμε τις εκδόσεις οι

οποίες είναι εγκατεστημένες χρησιμοποιούμε την εντολή `node --version` για το `node.js` και `npm --version` για το `npm` στο terminal του VS Code ή στο cmd των Windows. Η τρέχουσα εγκατεστημένη έκδοση για το `node.js` είναι η 16.6.0, ενώ για το `npm` η 7.19.1. Οι πληροφορίες αυτές είναι αποθηκευμένες στο αρχείο `package-lock.json`, μαζί με διάφορα άλλα `dependencies` κλπ.

```
C:\Users\admin>node --version
v16.6.0
```

Εικόνα 5.15 Η έκδοση για το `node.js`

```
C:\Users\admin>npm --version
7.19.1
```

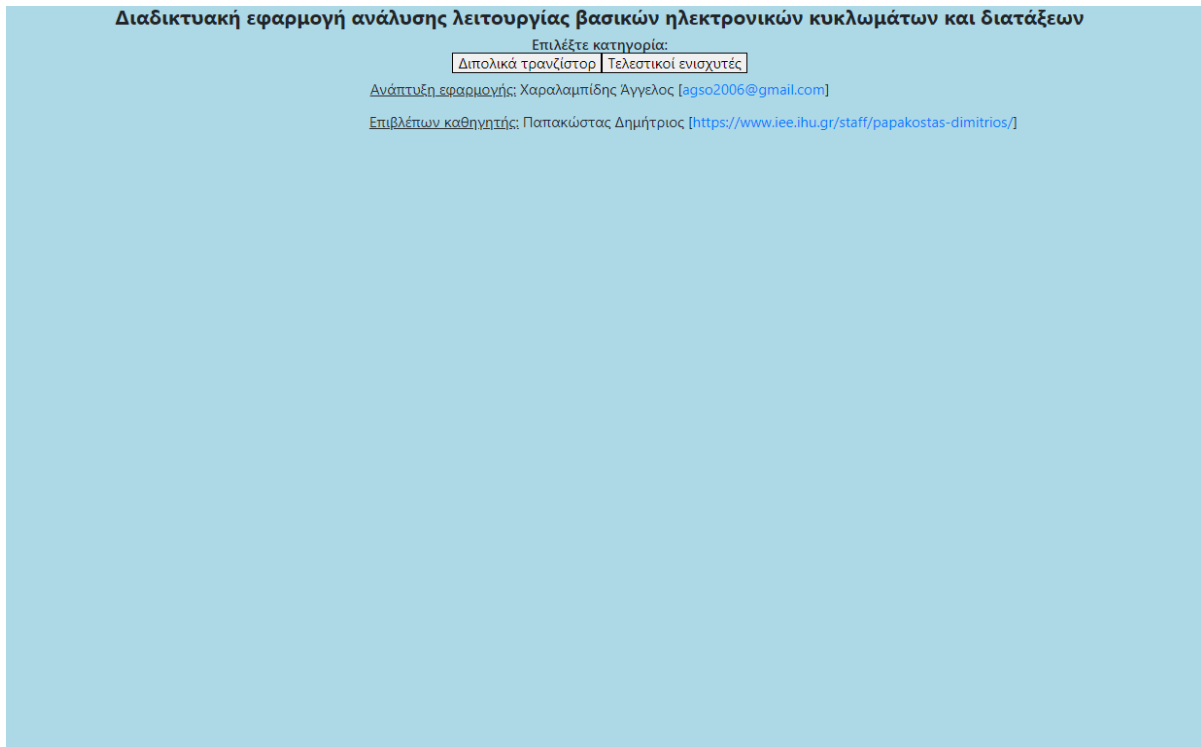
Εικόνα 5.16 Η έκδοση για το `npm`

```
"devDependencies": {
  "bootstrap": "^4.6.0",
  "chartist": "^0.10.1",
  "chartist-plugin-tooltips-updated": "^0.1.4",
  "react": "^17.0.2",
  "react-bootstrap": "^1.5.2",
  "react-chartist": "^0.14.4",
  "react-dom": "^17.0.2",
  "react-scripts": "4.0.3",
  "standard": "^16.0.3"
},
"dependencies": {
  "chartist-plugin-legend": "^0.6.2",
  "chartist-plugin-vertical-line": "github:zubenkoivan/chartist-plugin-vertical-line",
  "react-google-charts": "^4.0.0",
  "recharts": "^2.1.9",
  "sass": "^1.49.0"
}
```

Εικόνα 5.17 Διάφορα `dependencies` που έχουν εγκατασταθεί

5.4.1 Το αρχείο `App.js`

Όπως κάθε πρόγραμμα το οποίο είναι γραμμένο σε γλώσσα `React`, έτσι και αυτό, ξεκινάει από το αρχείο `App.js`. Σε αυτό υπάρχουν οι πληροφορίες για την πρώτη εικόνα που θα εμφανιστεί στον χρήστη μόλις ανοίξει την εφαρμογή.



Εικόνα 5.18 Η εικόνα που φαίνεται με το άνοιγμα της εφαρμογής

Στον κώδικα του App.js υπάρχει εξ ορισμού η συνάρτηση App (). Αρχικά ορίζονται μέσα της οι state μεταβλητές count, showMain, colorBJT και colorOP, όπου με τη χρήση του useState αρχικοποιούνται. Όπως προαναφέρθηκε, το useState είναι ένα hook που βοηθάει στη διαχείριση του state. Όταν γίνεται declare μια state μεταβλητή επιστρέφεται ένας πίνακας με 2 αντικείμενα, ένα για το current value και ένα για τη συνάρτηση που θα μας επιτρέψει να το κάνουμε update. Στην περίπτωση μας το current value είναι το count, showMain, colorBJT και colorOP ενώ οι συναρτήσεις που επιτρέπουν την επεξεργασία τους είναι οι setCount, setShowMain, setColorBJT, και setColorOP.

```
function App() {
  const [count, setCount] = useState(0);
  const [showMain, setShowMain] = React.useState(false)

  const [colorBJT, setColorBJT] = useState('black')
  const [colorOP, setColorOP] = useState('black')
```

Εικόνα 5.19 Ο ορισμός και η αρχικοποίηση των μεταβλητών

Η count είναι η μεταβλητή η οποία χρησιμοποιείται για να κάνει την εναλλαγή από διπολικά τρανζίστορ σε ΤΕ. Πιο συγκεκριμένα, όταν ο χρήστης επιλέξει για παράδειγμα τα διπολικά τρανζίστορ, μέσω του onClick attribute που μπορεί να οριστεί σε ένα button, καλείται μια anonymous function, μέσα στην οποία γίνεται update η τιμή του count σε 0. Η ίδια διαδικασία ακολουθείται και για το button των τελεστικών ενισχυτών, με τη διαφορά ότι στη συνάρτηση setCount μέσα στην anonymous function η τιμή θα γίνει 1. Κάτι παρόμοιο γίνεται και με την showMain. Η συγκεκριμένη μεταβλητή δημιουργήθηκε για να μην εμφανίζεται κανένα κύκλωμα στην οθόνη προτού επιλεγεί κάποια κατηγορία από τον χρήστη. Η αρχική της τιμή είναι false και γίνεται true μέσα στην anonymous function, το button της οποίας έχει επιλεγεί από τον χρήστη. Προχωρώντας, οι μεταβλητές

colorBJT και colorOP έχουν οριστεί για να μπορεί να χρωματίζεται το button το οποίο είναι επιλεγμένο ή επιλέγει ο χρήστης. Η αλλαγή αυτή γίνεται επίσης μέσα στην anonymous function του εκάστοτε button, μέσω της συνάρτησης setColorOP και setColorBJT, ανάλογα το button για το οποίο πρόκειται. Για να μπορέσει να γίνει η μεταφορά του count στη συνάρτηση main και να απεικονιστούν τα κατάλληλα κυκλώματα γίνεται χρήση του opt, το οποίο χρησιμοποιείται ως props object. Όπως μπορεί να διαπιστώσει κανείς και από τον κώδικα που βρίσκεται στο παράρτημα, το count θα περάσει στην main στην περίπτωση που η showMain έχει την τιμή true, έχει επιλεγθεί δηλαδή μία από τις δύο κατηγορίες.

5.4.2 Ο φάκελος main

5.4.2.1 Το αρχείο main.jsx

Αφού λοιπόν έχουμε περάσει μέσω του opt την επιλογή του χρήστη όσον αφορά την ομάδα κυκλωμάτων που επιθυμεί να εμφανιστεί, δηλώνουμε ένα array, το οποίο ανάλογα με το αν το opt είναι 0 ή 1, θα περιέχει τα κυκλώματα είτε των διπολικών τρανζίστορ είτε των τελεστικών ενισχυτών. Κατόπιν, ορίζεται η μεταβλητή key. Το key είναι το current state μας σε κάθε στιγμή το οποίο θα αλλάζει με κάθε render. Η αλλαγή με κάθε render γίνεται με τη βοήθεια του useEffect hook. Το useEffect βοηθάει στην πραγματοποίηση side effects σε function components. Τρέχει κάθε φορά, και μετά από το πρώτο render και μετά από κάθε update. Ουσιαστικά καλείται όταν γίνεται αλλαγή σε κάποιο state, κάθε φορά που γίνεται re-render ή κάνουμε save το project, χωρίς να χρειάζεται να γίνει refresh. Οπότε, ορίζουμε τη useEffect, η οποία στη συγκεκριμένη περίπτωση θα μας χρησιμεύσει για την εναλλαγή μεταξύ των κυκλωμάτων.

```
useEffect(() => setKey($ => $ >= components.length ? 0 : $), [key, setKey])
```

Εικόνα 5.20 Το useEffect hook

Για να πετύχει αυτή την εναλλαγή, ελέγχει κάθε φορά εάν το key (το οποίο περνάει παραμετρικά από τη συνάρτηση onClick που βρίσκεται παρακάτω) είναι μεγαλύτερο από τα όρια του array. Στην περίπτωση που είναι το μηδενίζει, ενώ αν δεν είναι το αφήνει όπως είναι.

```
onClick={() => setKey(idx)}
```

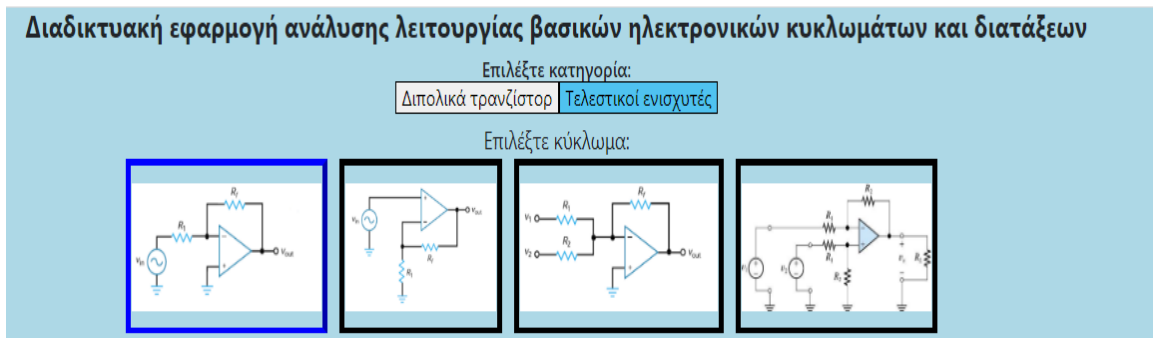
Εικόνα 5.21 Το σημείο που περνάει παραμετρικά το index ως key

Από εκεί και πέρα, η απεικόνιση γίνεται μέσα στο return και συγκεκριμένα μέσα σε ένα <Container>. Μέσα στο Container, ορίζονται αρχικά τα div με τα επιθυμητά attributes, ενώ, στη συνέχεια, γίνεται χρήση της συνάρτησης map (). Η map () ουσιαστικά δημιουργεί ένα καινούριο array καλώντας μια συνάρτηση για κάθε στοιχείο ενός array που της δίνουμε [18]. Η διαδικασία αυτή εκτελείται μία φορά για κάθε στοιχείο του αρχικού πίνακα. Καλώντας την λοιπόν για τον πίνακα components, ο οποίος θα περιέχει είτε τα κυκλώματα των διπολικών τρανζίστορ είτε αυτά των τελεστικών ενισχυτών ανάλογα με την τιμή του opt, θα εμφανίσει στην οθόνη την ανάλογη ομάδα κυκλωμάτων. Πριν την εμφάνιση του υπάρχει ένα key attribute. Ο λόγος χρήσης του είναι για να μπορεί να αναγνωρίσει η React τότε μια εικόνα έχει αλλάξει. Έτσι λοιπόν, μέσα στο και με τα κατάλληλα attributes, θα εμφανιστούν τα επιθυμητά κυκλώματα. Ιδιαίτερη αναφορά γίνεται στο borderColor, όπου ο έλεγχος που πραγματοποιείται αποσκοπεί στο να γίνεται μπλε το border του κυκλώματος που επιλέγεται. Παρακάτω, με χρήση των props array και idx, περνάει ο πίνακας με τα components και το index του

στο `component fromarray`, στο οποίο γίνεται η εμφάνιση του επιλεγμένου κυκλώματος από την τρέχουσα ομάδα κυκλωμάτων.



Εικόνα 5.22 Εμφάνιση των διπολικών τρανζίστορ



Εικόνα 5.23 Εμφάνιση των τελεστικών ενισχυτών

Τέλος, είναι σημαντικό να αναφερθεί ότι για να μπορέσει να γίνει χρήση των κυκλωμάτων μέσα στον πίνακα `components`, πρέπει αυτά να γίνουν `import` στην αρχή, μαζί με το απαραίτητο `bootstrap` και τα `hooks`. Ο κώδικας του `main.jsx` είναι διαθέσιμος στο παράρτημα του συγγράμματος.

5.4.2.2 Το αρχείο `componentfromarray.jsx`

Το δεύτερο αρχείο που βρίσκεται μέσα στο φάκελο `main` είναι το `componentfromarray.jsx`. Περνώντας λοιπόν τον πίνακα με τα `components` και το `key` μέσω των `props array` και `idx` από τη `main.jsx`, η `function ComponentFromArray` είναι σε θέση να κάνει την επιλογή του κυκλώματος. Αυτό γίνεται δημιουργώντας δύο μεταβλητές, την `Comp`, στην οποία εκχωρείται το `array` και το αντίστοιχο `index` και την `First`, η οποία έχει το πρώτο κύκλωμα των διπολικών τρανζίστορ. Ο λόγος ύπαρξης της `First` είναι επειδή όταν ο χρήστης βρίσκεται στο 4^ο κύκλωμα των τελεστικών ενισχυτών και επιλέξει τα διπολικά τρανζίστορ, θα έπρεπε να εμφανίζεται κάποιο 4^ο κύκλωμα των διπολικών τρανζίστορ. Αυτό συμβαίνει διότι με αυτόν τον τρόπο πραγματοποιείται η εναλλαγή μεταξύ των κυκλωμάτων, όταν δηλαδή επιλέγεται το πρώτο κύκλωμα των διπολικών τρανζίστορ και κατόπιν γίνει κλικ στο κουμπί για τους τελεστικούς ενισχυτές, να εμφανίζεται το πρώτο κύκλωμα από αυτήν την ομάδα, όταν επιλεγεί το δεύτερο από τα διπολικά, μετά από κλικ στους τελεστικούς να εμφανίζεται το δεύτερο κ.ο.κ. Έτσι λοιπόν, επειδή δεν υπάρχει 4^ο κύκλωμα για τα διπολικά τρανζίστορ, με τη χρήση της `First`, θα εμφανιστεί το πρώτο κύκλωμα, αυτό της πόλωσης διαίρετη τάσης με το ισοδύναμό του. Οπότε, μετά τη δημιουργία των μεταβλητών, επιστρέφεται το επιθυμητό κύκλωμα μέσα από έναν έλεγχο του `index` με το μέγεθος του `array`. Η `componentfromarray`, λόγω και του μικρού μήκους της, θα μπορούσε να ενσωματωθεί στην `main.jsx`. Ο λόγος που προτιμήθηκε η υλοποίησή της σαν ξεχωριστό

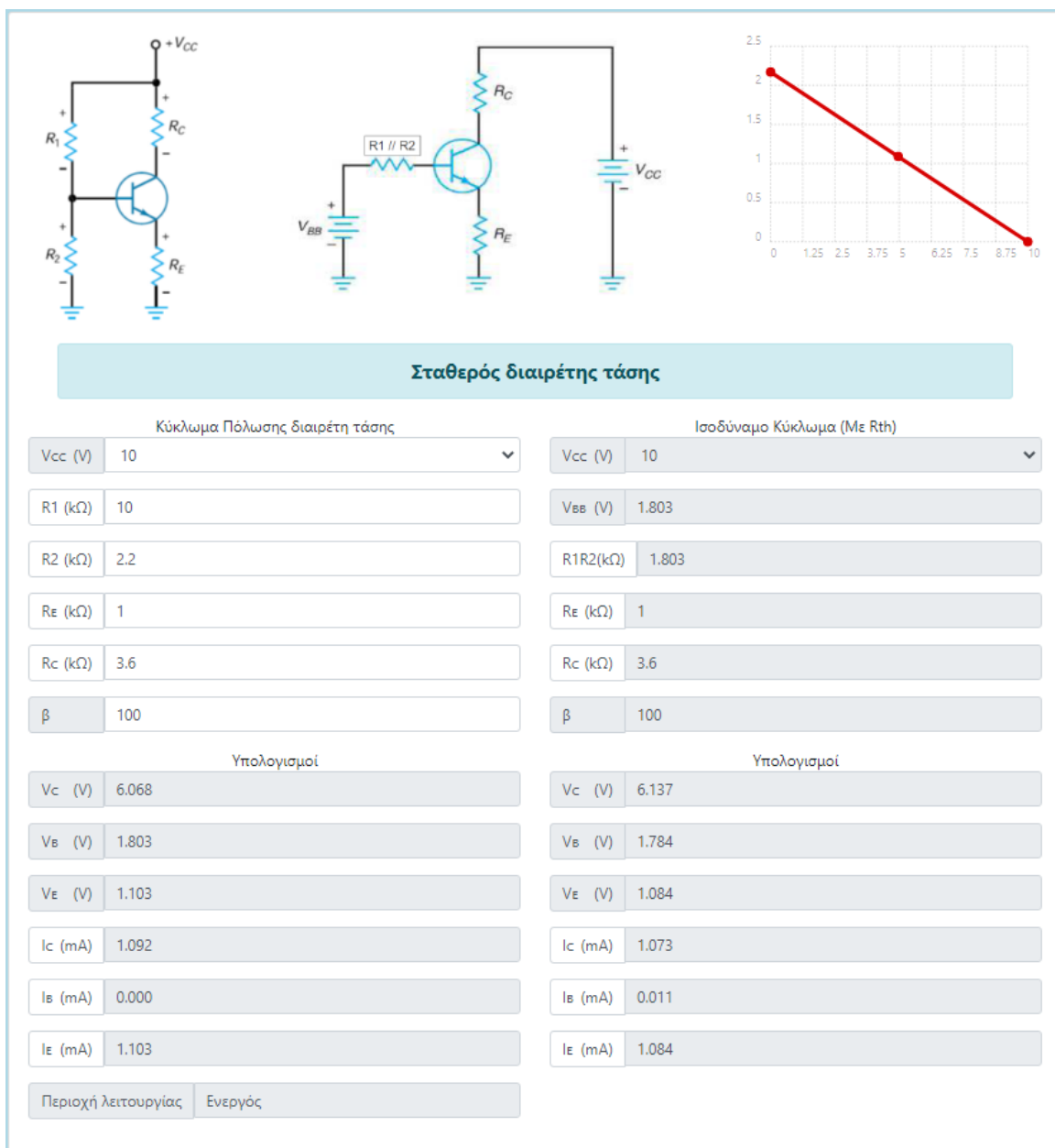
component είναι επειδή με αυτόν τον τρόπο ο κώδικας είναι πιο εύκολα επαναχρησιμοποιήσιμος και συντηρήσιμος, γεγονός που μπορεί να βοηθήσει σημαντικά σε περίπτωση που χρειαστεί κάποια μελλοντική βελτίωση ή επέκταση. Η υλοποίησή του παρατίθεται στο παράρτημα της εργασίας.

5.4.3 Ο φάκελος με τα κυκλώματα (circuits)

Όπως προαναφέρθηκε, ο φάκελος circuits περιλαμβάνει τα κυκλώματα τα οποία υλοποιήθηκαν. Κάθε υποφάκελος κυκλώματος περιέχει την αντίστοιχη εικόνα και το αρχείο με τον κώδικα που υλοποιεί τις εκάστοτε λειτουργίες του.

5.4.3.1 Το κύκλωμα πόλωσης διαιρέτη τάσης (VDB)

Το πρώτο κύκλωμα που δύναται να επιλέξει ο χρήστης από τα διπολικά τρανζίστορ είναι αυτό της πόλωσης διαιρέτη τάσης. Στην παρακάτω εικόνα φαίνεται η απεικόνισή του όταν επιλεγθεί:



Εικόνα 5.24 Το κύκλωμα πόλωσης διαιρέτη τάσης

Όπως παρατηρεί κανείς, στο πάνω μέρος βρίσκονται οι εικόνες του κυκλώματος πόλωσης διαιρέτη τάσης με το ισοδύναμό του και ακριβώς δίπλα βρίσκεται ένα γράφημα το οποίο απεικονίζει τη γραμμή φορτίου ανάλογα με τις τιμές τις οποίες θα εκχωρήσει ο χρήστης. Ακριβώς από κάτω βρίσκονται οι αναλύσεις για τα συγκεκριμένα κυκλώματα. Πριν από οτιδήποτε άλλο είναι σημαντικό να αναφερθεί ότι για να γίνει η σύνδεση του κυκλώματος, όπως και των υπολοίπων κυκλωμάτων που θα αναλυθούν παρακάτω, με το App.js και κατά συνέπεια με το main.jsx και το componentfromarray.jsx, πρέπει να υπάρχει μέσα στο border του καθενός η εικόνα του. Αυτό γίνεται στο τέλος, έξω από το component με την εντολή: `VDB.image = circuitImg`, όπου VDB το όνομα της συνάρτησης του component και circuitImg η εικόνα του. Όσον αφορά τον υπόλοιπο κώδικα, για να επιτευχθούν τα παραπάνω, η λογική ήταν να δημιουργηθούν πρώτα τα πεδία με τις default τιμές. Πιο συγκεκριμένα, για τα πεδία στο κύκλωμα πόλωσης διαιρέτη τάσης στα οποία ο χρήστης μπορεί να εκχωρήσει τιμή, έγινε χρήση μιας state μεταβλητής με όνομα values και μέσω του useState, δόθηκαν αρχικές τιμές.

```
const [values, setValues] = useState({
  Vcc: 10,
  R1: 10_000,
  R2: 2_200,
  Re: 1_000,
  Rc: 3_600,
  β: 100
})
```

Εικόνα 5.25 Αρχικές τιμές για τα πεδία που χρειάζονται εκχώρηση τιμής

Κάθε φορά που αλλάζει η μεταβλητή values τρέχει το useEffect, μέσα στο οποίο γίνονται οι υπολογισμοί.

```

useEffect(() => {

  const Vcc = parseFloat(values.Vcc)
  const R1 = parseFloat(values.R1)
  const R2 = parseFloat(values.R2)
  const Re = parseFloat(values.Re)
  const Rc = parseFloat(values.Rc)
  const h = parseFloat(values.β)

  /* eslint-disable prefer-const */
  let bjtMode = BJT_MODES.AMP
  let a = (h / (h + 1))
  let Ib = 0
  let Vb = (R2/(R1+R2)) * Vcc
  let Ve = Vb - 0.7
  let Ie = Ve / Re
  let Ic = (a * Ie)
  let Vc = Vcc - (Ic * Rc)
  let Vce = Vc - Ve

  const outcomes = { Vc, Vb, Ve, Vce, Ic: Ic * 1000, Ib: Ib * 1000, Ie: Ie * 1000, bjtMode }
  setOutcomes({ INITIAL_OUTCOMES, ...outcomes })

  /* eslint-enable */
  const equalOutcomes = calculateEqual(values)
  setEqualOutcomes({ INITIAL_EQUAL_OUTCOMES, ...equalOutcomes })

  const IcSat = (Vcc / (Rc + Re)) * 1000

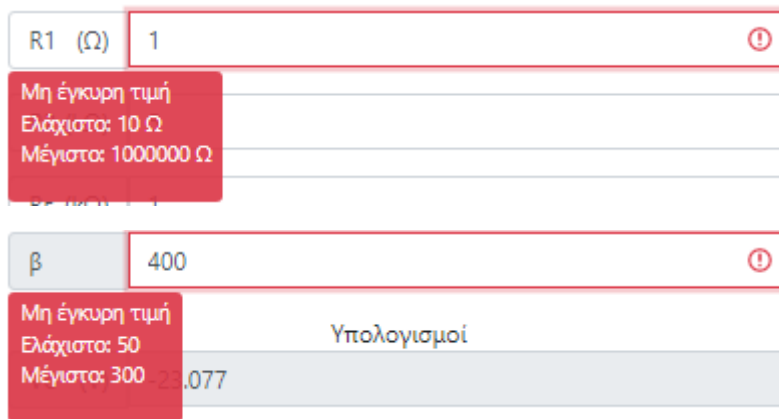
  setPlotData({
    series: [[
      { meta: 'Ic(sat)', x: 0, y: IcSat.toFixed(2) },
      { meta: 'Q', x: Vce.toFixed(2), y: (Ic * 1000).toFixed(2) },
      { meta: 'Vce(cutoff)', x: Vcc, y: 0 }
    ]],
  })
}, [values])

```

Εικόνα 5.26 Η useEffect

Η μεταβλητή `outcomes` είναι αυτή η οποία περιέχει τις τιμές για τα πεδία που υπολογίζονται. Πρόκειται επίσης για μια state μεταβλητή που έχει δημιουργηθεί πιο πάνω με τη χρήση του `useState` για αρχικοποίηση (0 οι αρχικές τιμές εξόδου) και μέσω της `update` συνάρτησης `setOutcomes` θα κάνει κάθε φορά `override` τις παλιές τιμές για να εμφανίσει τις καινούριες. Ο πολλαπλασιασμός με το 1000 που φαίνεται στα ρεύματα έχει να κάνει με τη σωστή εμφάνισή τους ανάλογα με τη μονάδα μέτρησης που επιλέγει ο χρήστης και αναλύεται παρακάτω. Με τον ίδιο τρόπο υπολογίζονται και οι τιμές για τα πεδία του ισοδύναμου κυκλώματος, με τη διαφορά ότι όλα τα πεδία είναι `disabled`, δεν έχει δηλαδή ο χρήστης τη δυνατότητα επεξεργασίας τους, καθώς αυτό δεν θα είχε ουσιαστικό νόημα. Για τον υπολογισμό των τιμών περνάει παραμετρικά η μεταβλητή `values` στη συνάρτηση `calculateEqual()`, η οποία δημιουργήθηκε για το σκοπό αυτό. Όσον αφορά τον σχηματισμό των πεδίων, αυτός έγινε με τη βοήθεια ενός άλλου `component` που δημιουργήθηκε, του `CircuitValueInput`. Καθότι το `CircuitValueInput` χρησιμοποιείται για όλα τα κυκλώματα της εφαρμογής, θα αναλυθεί παρακάτω σε ξεχωριστή ενότητα. Για ορθότερη και καλύτερη απεικόνιση των πεδίων, των εικόνων των κυκλωμάτων, καθώς και του γραφήματος γίνεται χρήση `Bootstrap`. Συγκεκριμένα, με τα κατάλληλα `imports` στην αρχή, χρησιμοποιούνται τα `Card`, `Col`, `Row` και `Container`. Για το πεδίο που περιέχει τις τιμές για την τροφοδοσία του κυκλώματος, το `Vcc`, χρησιμοποιείται ένα ξεχωριστό `component`, το `VccValueSelector`. Ο λόγος είναι διότι θέλουμε να μπορεί ο χρήστης να επιλέξει συγκεκριμένες τιμές

από ένα dropdown menu. Οι τιμές αυτές είναι 5, 10, 12, 15, 18, 20, 24. Και στην περίπτωση του V_{CC} και των υπολοίπων τιμών χρησιμοποιούνται props για να περάσουν τα αντίστοιχα στοιχεία στο επιθυμητό component. Στη συνέχεια, επειδή οι αντιστάσεις στα κυκλώματα πρέπει να έχουν κάποια όρια, έχει οριστεί ως ελάχιστη επιτρεπόμενη τιμή τα 10Ω και ως μέγιστη το $1M\Omega$. Σε περίπτωση που ο χρήστης επιχειρήσει να πληκτρολογήσει τιμή εκτός των ορίων θα του εμφανιστεί ένα μήνυμα δίπλα στο πεδίο όπου γράφει “Μη έγκυρη τιμή” και από κάτω την ελάχιστη και την μέγιστη επιτρεπόμενη. Κάτι ανάλογο συμβαίνει και με το β . Σε τρανζίστορ σαν αυτά που μελετάμε οι τιμές του β είναι από 50 μέχρι 300 περίπου, καθώς σε τιμές μικρότερες του 50 ή μεγαλύτερες του 300 το τρανζίστορ ή δεν λειτουργεί σωστά ή καταστρέφεται. Για το λόγο αυτό ο χρήστης έχει τη δυνατότητα να εκχωρήσει τιμή ανάμεσα σε αυτό το εύρος. Σε διαφορετική περίπτωση θα εμφανιστεί το ίδιο μήνυμα όπως στις αντιστάσεις.



Εικόνα 5.27 Το μήνυμα σε περίπτωση εκχώρησης μη επιτρεπτής τιμής

Παρακάτω, υπάρχει ένα πεδίο το οποίο αναφέρει την περιοχή λειτουργίας στην οποία βρίσκεται το κύκλωμα. Το κύκλωμα πόλωσης διαίρετη τάσης συγκεκριμένα λειτουργεί στην ενεργό περιοχή, αλλιώς η λειτουργία του δεν είναι σωστή. Συνεπώς, στην περιοχή λειτουργίας θα εμφανίζεται είτε ενεργός περιοχή, είτε λάθος λειτουργία αν υπάρχει κάποιο αποτέλεσμα που δεν είναι ορθό. Για να είναι σίγουρο ότι ο χρήστης θα παρατηρήσει τη λάθος λειτουργία του κυκλώματος όταν αυτή υφίσταται, το αντίστοιχο πεδίο αναβοσβήνει. Για να γίνει αυτό, στέλνεται ως prop το shouldBlink στο CircuitValueInput, στο οποίο επιτελείται η επιθυμητή λειτουργία. Ακολούθως, για να γίνει επεξεργασία των τιμών και των λειτουργιών μέσα στο VccValueSelector και το CircuitValueInput, πρέπει να περάσουν τα αντίστοιχα props. Αξίζει να σημειωθεί το γεγονός ότι όσα πεδία είναι disabled, δεν μπορεί δηλαδή η τιμή τους να αλλάξει από το χρήστη, δεν χρησιμοποιούν το setValues, καθώς το συγκεκριμένο prop χρησιμοποιείται για αρχικοποίηση και κάτι τέτοιο δεν χρειάζεται στα συγκεκριμένα πεδία. Επόμενη λειτουργία είναι αυτή της εναλλαγής μεταξύ των μονάδων μέτρησης. Αναλυτικότερα, στα πεδία των αντιστάσεων και των ρευμάτων, υπάρχει η δυνατότητα για μετατροπή από Ω σε $k\Omega$ και σε $M\Omega$ και από mA σε μA αντίστοιχα. Κάνοντας mouseover πάνω τους εμφανίζεται μέσα σε ένα μικρό πλαίσιο το μήνυμα “Κάντε κλικ για αλλαγή μονάδας μέτρησης” και ακριβώς από κάτω εμφανίζονται οι διαθέσιμες μονάδες. Στον κώδικα αυτό γίνεται με τη δημιουργία δύο πινάκων, που περιέχουν τη μονάδα μέτρησης, τη συνάρτηση για επαναφορά της τιμής και τη συνάρτηση για την εμφάνισή της. Οι συναρτήσεις αυτές καλούνται από το CircuitValueInput, στο οποίο περνάνε επίσης ως props το measurement για τη μονάδα μέτρησης και το measurements για τους παραπάνω πίνακες.

Κύκλωμα Πόλωσης διαιρέτη τάσης

Κάνε κλικ για αλλαγή μονάδας μέτρησης. Διαθέσιμες μονάδες: κΩ, Ω, ΜΩ

V_{CC} (V) 10

R1 (κΩ) 10

R2 (κΩ) 2.2

Κύκλωμα Πόλωσης διαιρέτη τάσης

Κάνε κλικ για αλλαγή μονάδας μέτρησης. Διαθέσιμες μονάδες: κΩ, Ω, ΜΩ

V_{CC} (V) 10

R1 (Ω) 10000

R2 (κΩ) 2.2

Κύκλωμα Πόλωσης διαιρέτη τάσης

Κάνε κλικ για αλλαγή μονάδας μέτρησης. Διαθέσιμες μονάδες: κΩ, Ω, ΜΩ

V_{CC} (V) 10

R1 (ΜΩ) 0.01

R2 (κΩ) 2.2

Κάνε κλικ για αλλαγή μονάδας μέτρησης. Διαθέσιμες μονάδες: mA, μA

V_E (V) 1.103

I_C (mA) 1.092

I_B (mA) 0.000

Κάνε κλικ για αλλαγή μονάδας μέτρησης. Διαθέσιμες μονάδες: mA, μA

V_E (V) 1.103

I_C (μA) 1092

I_B (mA) 0.000

Εικόνα 5.28 Το μήνυμα για μετατροπή των μονάδων μέτρησης

Κατόπιν, δημιουργήθηκε το γράφημα που απεικονίζει τη γραμμή φορτίου. Για τη δημιουργία του χρησιμοποιήθηκε η βιβλιοθήκη Chartist καθώς και κάποια plugins, το ChartistGraph και το ChartistTooltip. Αρχικά, ορίστηκε ένα αντικείμενο plotOptions για τα options του γραφήματος. Αυτό περιέχει πληροφορίες για τις υποδιαιρέσεις και τα δεκαδικά των τιμών που εμφανίζονται, καθώς και το πώς δημιουργείται ένα καινούριο σημείο. Μέσω της συνάρτησης split() και χρησιμοποιώντας για διαχωριστή το “,” το νέο σημείο θα δημιουργείται μετά από το σημείο x, το V_{CE} δηλαδή. Η διαδικασία αυτή γίνεται μία φορά καθώς τα σημεία που χρειάζονται είναι δύο, το V_{CE} και το I_C.

```
const plotOptions = {
  axisX: {
    type: Chartist.AutoScaleAxis,
    onlyInteger: false
  },

  plugins: [
    ChartistTooltip({
      transformTooltipTextFnc: function(tooltip) {
        var ttip = tooltip.split(",");
        return "Vce: " + ttip[0] + ", Ic: " + ttip[1];
      }
    })
  ]
}
```

Εικόνα 5.29 Το αντικείμενο plotOptions

Η `setPlotData`, η οποία βρίσκεται μέσα στη `useEffect` που αναφέρθηκε προηγουμένως, είναι υπεύθυνη για τις συντεταγμένες των σημείων x και y . Συγκεκριμένα, το ρεύμα κορεσμού $I_{C(SAT)}$ θέλουμε να βρίσκεται στον άξονα y οπότε το x , το V_{CE} δηλαδή έχει την τιμή 0. Με την ίδια λογική το $V_{CE(CUTOFF)}$ βρίσκεται στον άξονα x έχοντας ως τιμή αυτή της V_{CC} ενώ η τιμή στον άξονα y είναι μηδενική. Το πιο σημαντικό σημείο πάνω στη γραμμή φορτίου είναι το σημείο ηρεμίας Q το οποίο έχει τιμή και στον άξονα x (V_{CE}) και στον y (I_C).

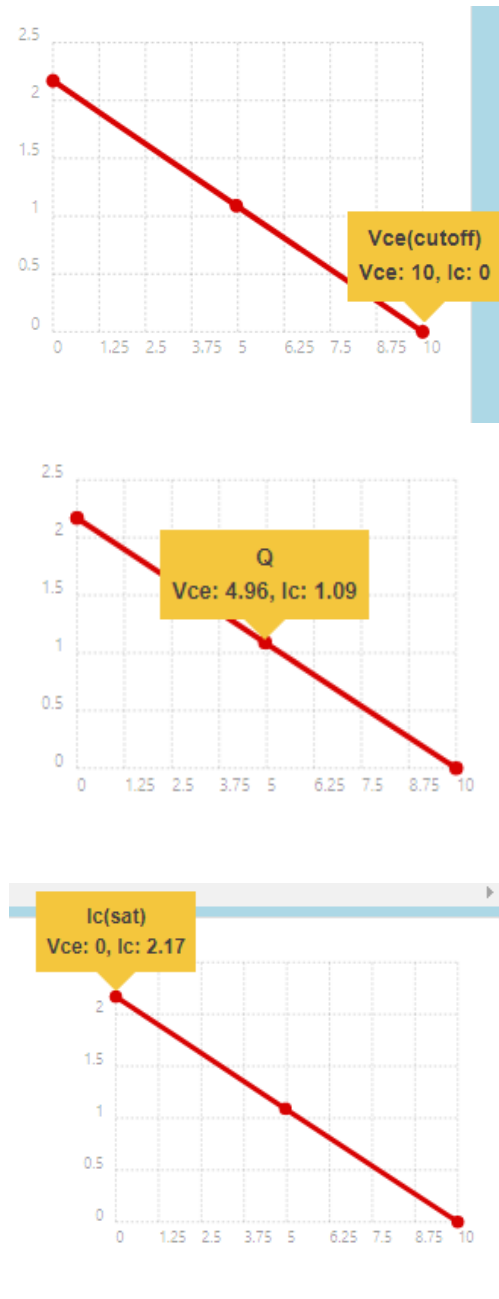
```
setPlotData({
  series: [[
    { meta: 'Ic(sat)', x: 0, y: IcSat.toFixed(2) },
    { meta: 'Q', x: Vce.toFixed(2), y: (Ic * 1000).toFixed(2) },
    { meta: 'Vce(cutoff)', x: Vcc, y: 0 }
  ]],
})
```

Εικόνα 5.30 Η `setPlotData`

Η απεικόνιση του γραφήματος γίνεται με τη βοήθεια της `state` μεταβλητής `plotData`, μέσα στο `return` της συνάρτησης, στο ίδιο `Row` με τις εικόνες των δύο κυκλωμάτων. Μέσα στο `ChartistGraph` χρησιμοποιούνται ως `attributes` τα `data` και `options` για τα `plotData` και `plotOptions` αντίστοιχα, ενώ το `className='ct-vdbchart'` παραπέμπει στο αρχείο `styles.scss` όπου υπάρχει η πληροφορία για το χρώμα της γραμμής.

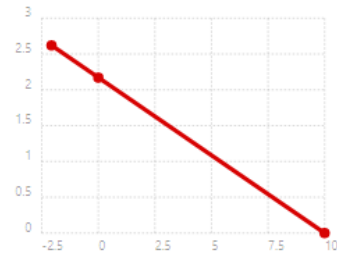
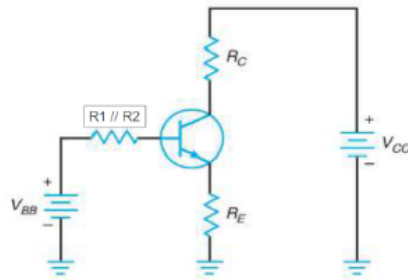
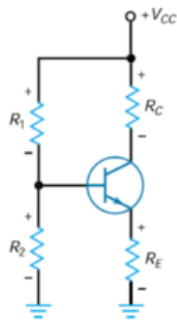
```
<Row className='mb-3'>
  <Col xs=3>
    <img style={{ float: 'initial' }} src={circuitImg} alt='' />
  </Col>
  <Col xs=5>
    <img style={{ float: 'initial', maxWidth: '100%' }} src={equalImg} alt='' />
  </Col>
  <Col xs=4>
    {plotData} && <ChartistGraph style={{ height: '16rem' }} className='ct-vdbchart' data={plotData} options={plotOptions} type='Line' />
  </Col>
</Row>
```

Εικόνα 5.31 Το `Row` μέσα στο οποίο απεικονίζονται οι εικόνες των κυκλωμάτων και το γράφημα



Εικόνα 5.32 Τα σημεία $V_{CE(CUTOFF)}$, Q και $I_{C(SAT)}$ στο γράφημα

Τελευταίο αλλά σημαντικό μέρος της ανάλυσης του κυκλώματος πόλωσης διαιρέτη τάσης είναι τα μηνύματα τα οποία εμφανίζονται στο χρήστη όταν στους υπολογισμούς προκύπτουν αποτελέσματα που θέτουν το κύκλωμα σε λάθος λειτουργία. Το πρώτο αφορά τη μη ικανοποίηση της συνθήκης $V_C > V_B > V_E$. Όταν συμβαίνει αυτό, εμφανίζεται ένα μήνυμα το οποίο προτρέπει το χρήστη να εισάγει τιμές που οδηγούν στην εύρυθμη λειτουργία του κυκλώματος.



Δεν ισχύει $V_C > V_B > V_E$... Κύκλωμα εκτός ενεργού περιοχής... Παρακαλώ ξαναεισάγετε τιμές

Κύκλωμα Πόλωσης διαιρέτη τάσης		Ισοδύναμο Κύκλωμα (Με Rth)	
Vcc (V)	10	Vcc (V)	10
R1 (Ω)	10000	VBB (V)	3.333
R2 (kΩ)	5	R1R2(kΩ)	3.333
RE (kΩ)	1	RE (kΩ)	1
RC (kΩ)	3.6	RC (kΩ)	3.6
β	200	β	200
Υπολογισμοί		Υπολογισμοί	
Vc (V)	0.567	Vc (V)	0.722
Vb (V)	3.333	Vb (V)	3.290
Ve (V)	2.633	Ve (V)	2.590
Ic (mA)	2.620	Ic (mA)	2.577
Ib (mA)	0.000	Ib (mA)	0.013
Ie (mA)	2.633	Ie (mA)	2.590
Περιοχή λειτουργίας	Λάθος λειτουργία		

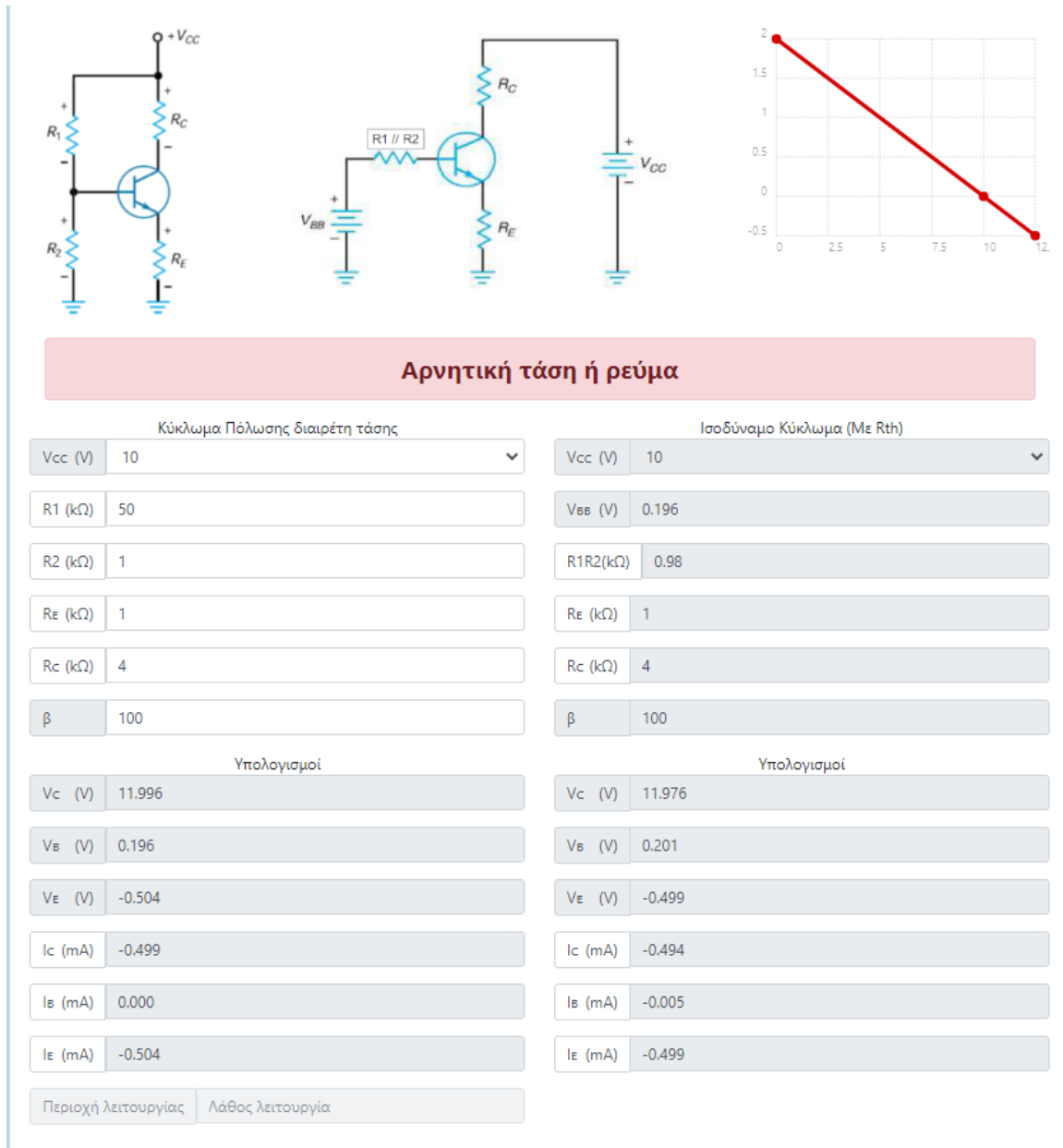
Εικόνα 5.33 Το μήνυμα που εμφανίζεται όταν $V_C < V_B$

Σε αυτή την περίπτωση το πεδίο της περιοχής λειτουργίας θα αναβοσβήνει για να προειδοποιεί τον χρήστη. Ο ρυθμός με τον οποίο θα αναβοσβήνει είναι ανά 2 second και δεν σταματάει μέχρι να εισαχθούν αποδεκτές τιμές. Η συγκεκριμένη πληροφορία βρίσκεται στο αρχείο styles.css.

```
.blink {
  animation: blinking-input 2s linear infinite;
}
```

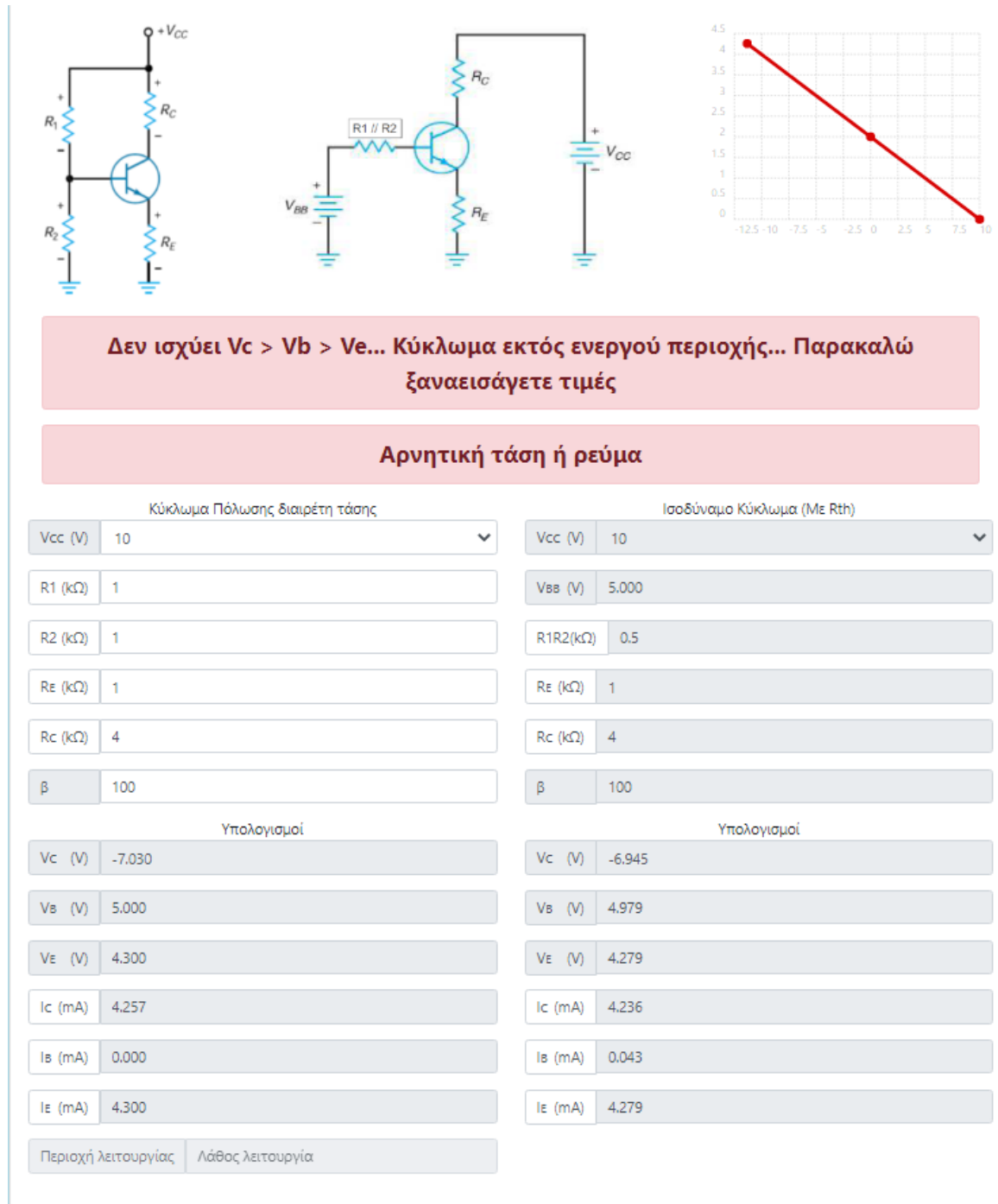
Εικόνα 5.34 Το σημείο που δημιουργείται το blinking του πεδίου

Επόμενο μήνυμα είναι αυτό το οποίο εμφανίζεται όταν προκύψει ως αποτέλεσμα αρνητική τάση ή ρεύμα. Και σε αυτή την περίπτωση η περιοχή λειτουργίας αναβοσβήνει, καθώς πρόκειται για μη αποδεκτό υπολογισμό.



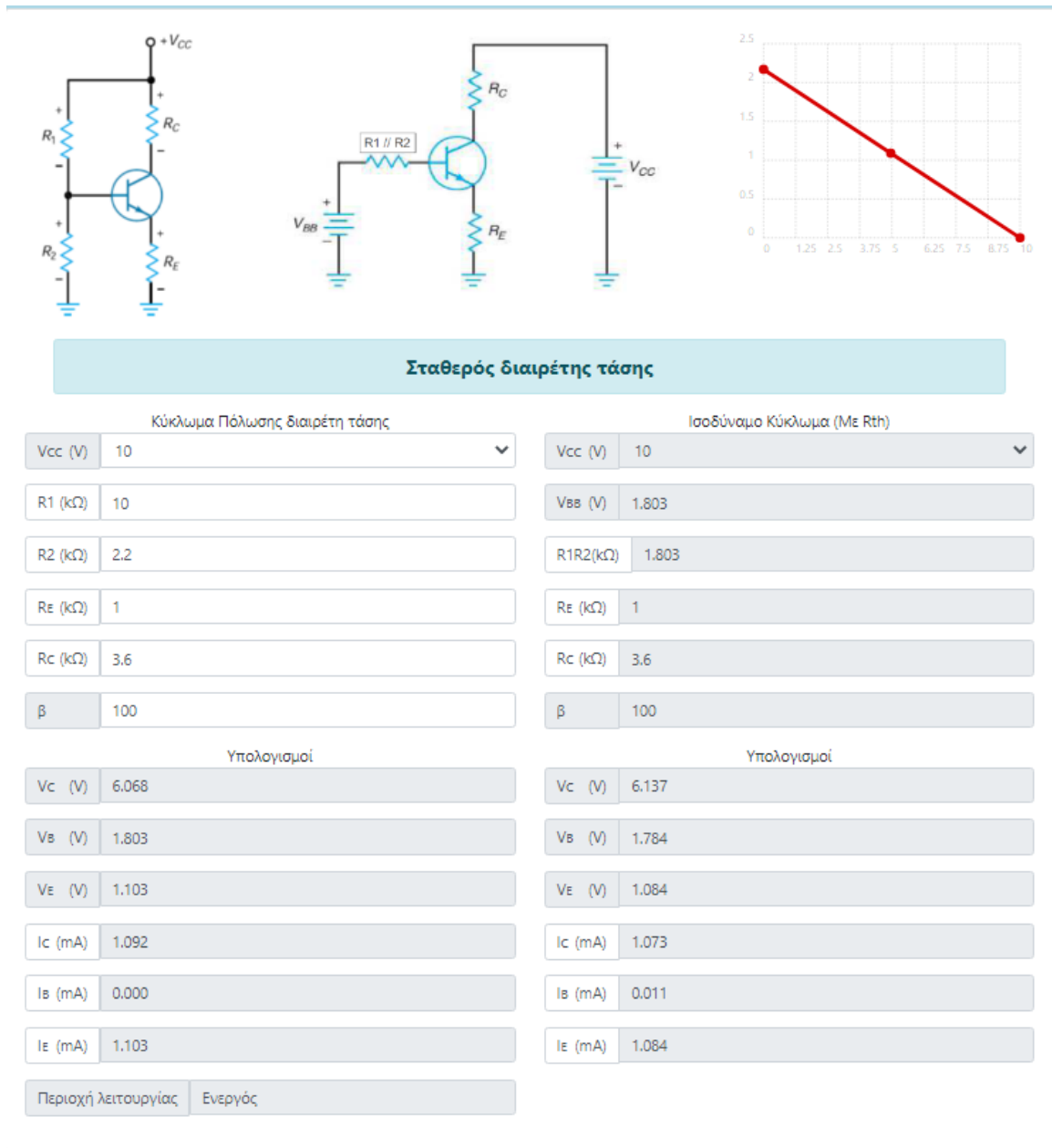
Εικόνα 5.35 Λάθος λειτουργία λόγω αρνητικής τάσης ή/και ρεύματος

Προφανώς τα δύο μηνύματα μπορούν να εμφανιστούν μαζί στην περίπτωση που ισχύουν και οι δύο συνθήκες.

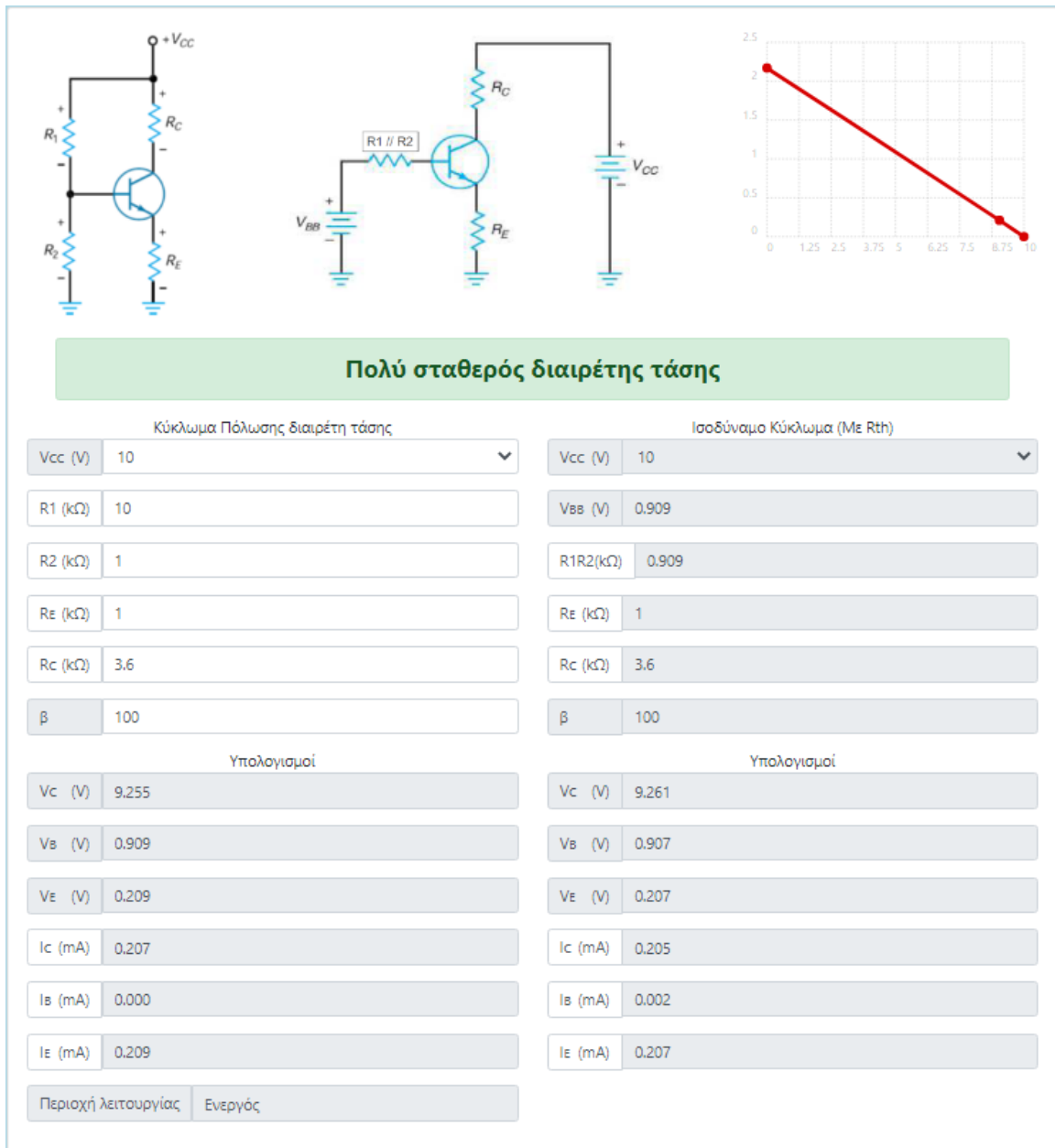


Εικόνα 5.36 Εμφάνιση και των δύο μηνυμάτων ταυτοχρόνως

Η υλοποίηση του κώδικα για την εμφάνιση των παραπάνω μηνυμάτων έγινε με το σχηματισμό κατάλληλων ελέγχων. Επειδή όταν εμφανίζεται τουλάχιστον ένα από τα δύο μηνύματα πρέπει να αλλάζει και η περιοχή λειτουργίας από ενεργός σε λάθος, υλοποιείται κατάλληλο κομμάτι στον κώδικα που το κάνει αυτό. Εκτός από τα μηνύματα που εμφανίζονται όταν στους υπολογισμούς προκύψει μη αποδεκτό αποτέλεσμα, υπάρχουν και άλλα δύο τα οποία σχετίζονται με τη σταθερότητα του διαιρέτη τάσης. Ένας διαιρέτης τάσης θεωρείται σταθερός όταν ικανοποιείται η σχέση $R_1 || R_2 < 0.1\beta_{dc}R_E$, ενώ όταν $R_1 || R_2 < 0.01\beta_{dc}R_E$ χαρακτηρίζεται πολύ σταθερός. Με βάση αυτά, ο χρήστης έχει τη δυνατότητα να διαπιστώσει πότε ισχύει κάτι από τα δύο, σύμφωνα με τις παραμέτρους που έχει εισάγει.



Εικόνα 5.37 Σταθερός διαιρέτης τάσης



Εικόνα 5.38 Πολύ σταθερός διαιρέτης τάσης

Στον κώδικα αυτό υλοποιείται με τον ίδιο τρόπο όπως τα παραπάνω μηνύματα, με χρήση Alert Bootstrap. Προφανώς τα δύο αυτά μηνύματα θα εμφανιστούν μόνο όταν το κύκλωμα έχει ορθή λειτουργία, βρίσκεται δηλαδή στην ενεργό περιοχή.

5.4.3.2 Τα κυκλώματα πόλωσης βάσης και εκπομπού (Bjt και Emmiter)

Τα επόμενα κυκλώματα προς επιλογή είναι αυτά της πόλωσης βάσης και της πόλωσης εκπομπού.

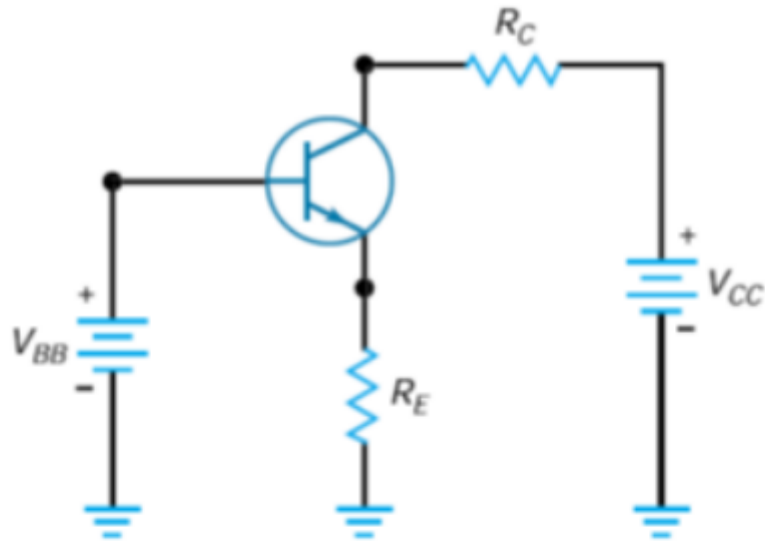
Κύκλωμα πόλωσης βάσης

V _{CC} (V)	10
V _{BB} (V)	2
R _B (kΩ)	20
R _C (kΩ)	1
β	100

Υπολογισμοί

V _C (V)	3.500
V _B (V)	0.700
V _E (V)	0.000
I _C (mA)	6.500
I _B (mA)	0.065
I _E (mA)	6.565
Περιοχή λειτουργίας	Ενεργός

Εικόνα 5.39 Κύκλωμα πόλωσης βάσης

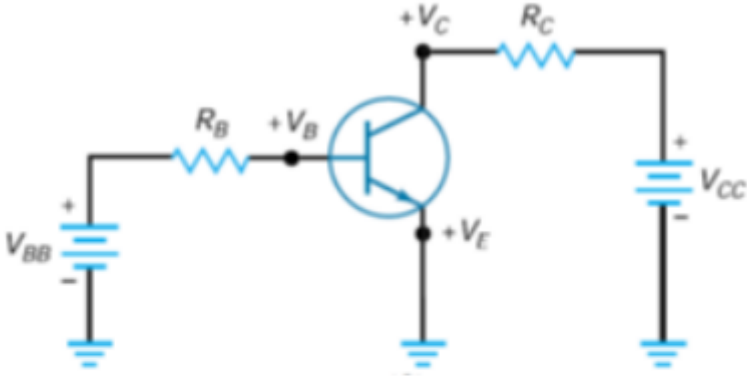


Κύκλωμα πόλωσης εκπομπού

V _{CC} (V)	10
V _{BB} (V)	6
R _E (kΩ)	3.3
R _C (kΩ)	4.7
β	100
Υπολογισμοί	
V _C (V)	5.600
V _B (V)	6.000
V _E (V)	5.300
I _C (mA)	0.936
I _B (mA)	0.670
I _E (mA)	1.606
Περιοχή λειτουργίας	Κορεσμός

Εικόνα 5.40 Κύκλωμα πόλωσης εκπομπού

Η υλοποίησή τους προγραμματιστικά έγινε ακριβώς με τον ίδιο τρόπο όπως και το κύκλωμα της πόλωσης διαιρέτη τάσης, χωρίς την ανάγκη για ύπαρξη προειδοποιητικών μηνυμάτων, καθώς τα συγκεκριμένα κυκλώματα, όσον αφορά την περιοχή λειτουργίας τους, μπορούν να βρίσκονται στην ενεργό, στον κορεσμό ή στην αποκοπή. Συνεπώς, η διαφοροποίησή τους αφορά τους υπολογισμούς για την εύρεση των τάσεων και των ρευμάτων.



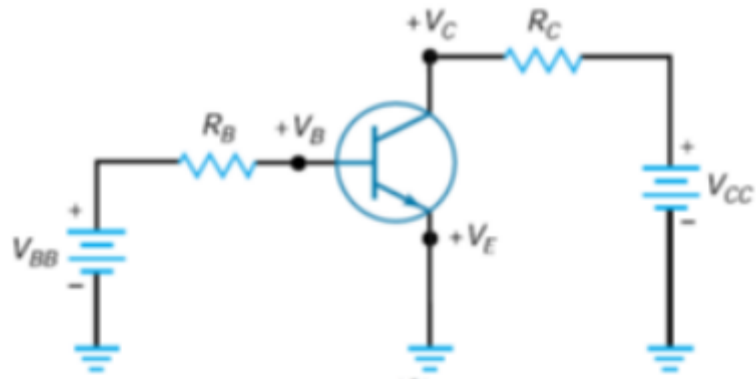
Κύκλωμα πόλωσης βάσης

V _{CC} (V)	10	▼
V _{BB} (V)	5	
R _B (kΩ)	20	
R _C (kΩ)	1	
β	100	

Υπολογισμοί

V _C (V)	0.300	
V _B (V)	0.700	
V _E (V)	0.000	
I _C (mA)	9.700	
I _B (mA)	0.215	
I _E (mA)	9.915	
Περιοχή λειτουργίας	Κορεσμός	

Εικόνα 5.41 Κύκλωμα πόλωσης βάσης – λειτουργία στον κορεσμό



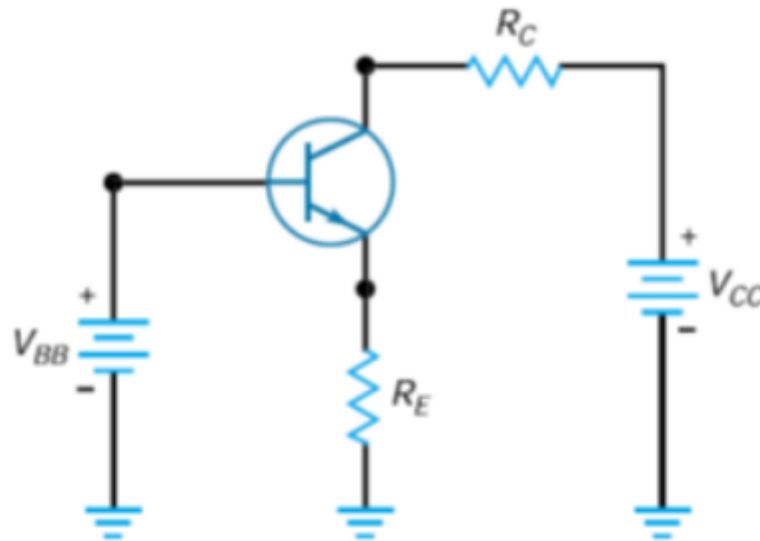
Κύκλωμα πόλωσης βάσης

V _{CC} (V)	10
V _{BB} (V)	0.6
R _B (kΩ)	20
R _C (kΩ)	1
β	100

Υπολογισμοί

V _C (V)	10.000
V _B (V)	0.600
V _E (V)	0.000
I _C (mA)	0.000
I _B (mA)	0.000
I _E (mA)	0.000
Περιοχή λειτουργίας	Αποκοπή

Εικόνα 5.42 Κύκλωμα πόλωσης βάσης – λειτουργία στην αποκοπή



Κύκλωμα πόλωσης εκπομπού

V_{CC} (V)	10
V_{BB} (V)	1
R_E (k Ω)	3.3
R_C (k Ω)	4.7
β	100

Υπολογισμοί

V_C (V)	9.577
V_B (V)	1.000
V_E (V)	0.300
I_C (mA)	0.090
I_B (mA)	0.001
I_E (mA)	0.091
Περιοχή λειτουργίας	Ενεργός

Εικόνα 5.43 Κύκλωμα πόλωσης εκπομπού – λειτουργία στην ενεργό περιοχή

Κύκλωμα πόλωσης εκπομπού

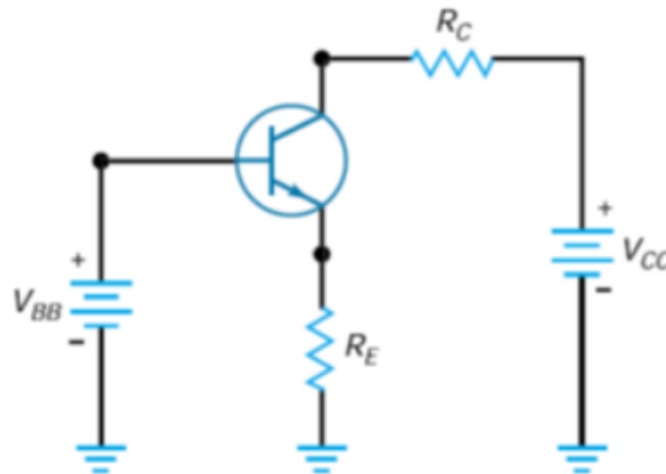
V _{CC} (V)	10
V _{BB} (V)	6
R _E (kΩ)	3.3
R _C (kΩ)	4.7
β	100

Υπολογισμοί

V _C (V)	5.600
V _B (V)	6.000
V _E (V)	5.300
I _C (mA)	0.936
I _B (mA)	0.670
I _E (mA)	1.606

Περιοχή λειτουργίας: Κορεσμός

Εικόνα 5.44 Κύκλωμα πόλωσης εκπομπού – λειτουργία στον κορεσμό



Κύκλωμα πόλωσης εκπομπού

V _{CC} (V)	10
V _{BB} (V)	0.6
R _E (kΩ)	3.3
R _C (kΩ)	4.7
β	100

Υπολογισμοί

V _C (V)	10.000
V _B (V)	0.600
V _E (V)	0.000
I _C (mA)	0.000
I _B (mA)	0.000
I _E (mA)	0.000

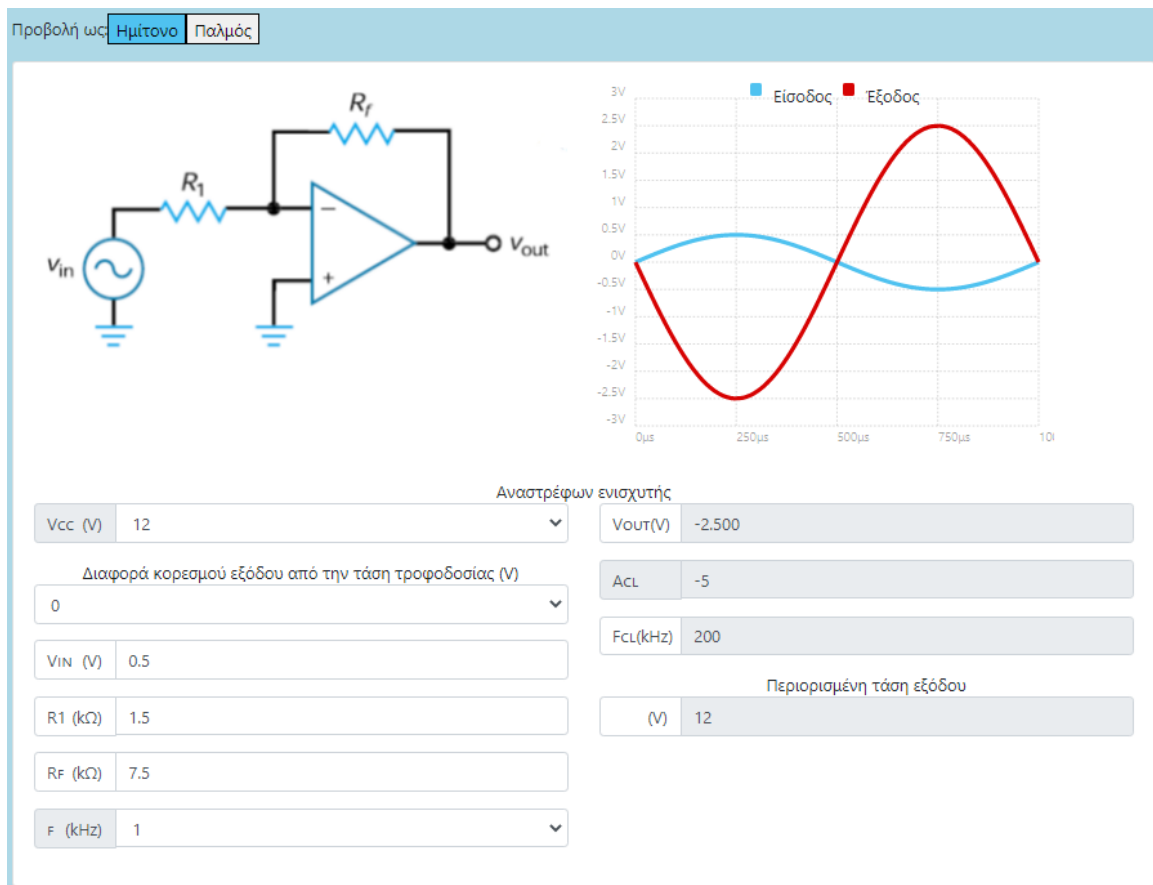
Περιοχή λειτουργίας Αποκοπή

Εικόνα 5.45 Κύκλωμα πόλωσης εκπομπού – λειτουργία στην αποκοπή

Ο περιορισμός ο οποίος ισχύει ως προς το εύρος τιμών της τάσης βάσης V_{BB} είναι ότι πρέπει να έχει ελάχιστη τιμή 0.5V και μέγιστη την τάση V_{CC} που έχει δώσει ο χρήστης. Αυτό συμβαίνει διότι χρειάζεται αφενός να μπορεί να διαπιστωθεί η λειτουργία σε αποκοπή για τάσεις $< 0.7V$ και αφετέρου να μην ξεπερνάει την V_{CC} , καθώς κάτι τέτοιο δεν συνηθίζεται. Το ελάχιστο όριο 0.5V επιλέχθηκε ούτως ώστε να έχουν κάποιο νόημα οι υπολογισμοί, καθώς σε κάθε περίπτωση για $V_{BB} < 0.7V$ το κύκλωμα βρίσκεται στην περιοχή αποκοπής. Το συγκεκριμένο εύρος τιμών ισχύει και για το κύκλωμα της πόλωσης βάσης και της πόλωσης εκπομπού.

5.4.3.3 Το κύκλωμα του αναστρέφοντος ενισχυτή (Inverting Amplifier)

Προχωρώντας στα κυκλώματα των τελεστικών ενισχυτών, το πρώτο κατά σειρά που φαίνεται είναι αυτό του αναστρέφοντος ενισχυτή. Επιλέγοντάς το ο χρήστης βλέπει στην οθόνη την παρακάτω εικόνα:



Εικόνα 5.46 Αναστρέφων ενισχυτής

Όπως φαίνεται, στο αριστερό μέρος υπάρχουν τα πεδία στα οποία ο χρήστης καταχωρεί τις επιθυμητές τιμές και στο δεξιό βρίσκονται τα αποτελέσματα που προκύπτουν. Πάνω από τους υπολογισμούς υπάρχει ένα γράφημα το οποίο απεικονίζει τα σήματα εισόδου και εξόδου είτε σε ημιτονοειδή είτε σε μορφή τετραγωνικού παλμού. Η επιλογή αυτή εξαρτάται από το κουμπί το οποίο θα πατηθεί και βρίσκεται πάνω από την εικόνα του κυκλώματος. Για τη δημιουργία του κουμπιού δημιουργήθηκε ένα ξεχωριστό index, ακολουθώντας τη λογική του App.js. Πιο αναλυτικά, χρησιμοποιείται η state μεταβλητή count για να μπορεί να αλλάζει η τιμή ανάλογα με το κουμπί που πατιέται. Συνεπώς, μέσα στην onClick συνάρτηση του <button>, εάν επιλεγθεί το button για το ημίτονο η count γίνεται 0, ενώ αν επιλεγθεί το button για τον παλμό γίνεται 1. Στο ίδιο αρχείο επιτυγχάνεται και ο χρωματισμός του button με τρόπο αντίστοιχο με αυτόν που προαναφέρθηκε στο App.js για τα buttons διπολικά τρανζίστορ και τελεστικοί ενισχυτές. Η τιμή της count περνάει στο αρχείο index2.jsx μέσω του props με όνομα opt. Στο τέλος του index γίνεται και η φόρτωση της εικόνας μέσα στο border.

```
import InvertingAmplifierr from './index2';
import CircuitImg from './circuit.png';

import React, {useState} from 'react';

export default function Inverting_Amplifier() {

  const [count, setCount] = useState(0);
  const [colorSin, setColorSin] = useState('blue')
  const [colorPulse, setColorPulse] = useState('black')

  return (
    <div>
      <p>Προβολή ως:</p>

      <button className={colorSin} onClick={() => [setCount(0), setColorPulse(() => colorPulse == 'black'), setColorSin(() => colorSin === 'black' ? 'blue' : 'blue')]}>
        Ημίτονο </button>

      <button className={colorPulse} onClick={() => [setCount(1), setColorSin(() => colorSin == 'black'), setColorPulse(() => colorPulse === 'black' ? 'blue' : 'blue')]}>
        Παλμός </button>

      <p>

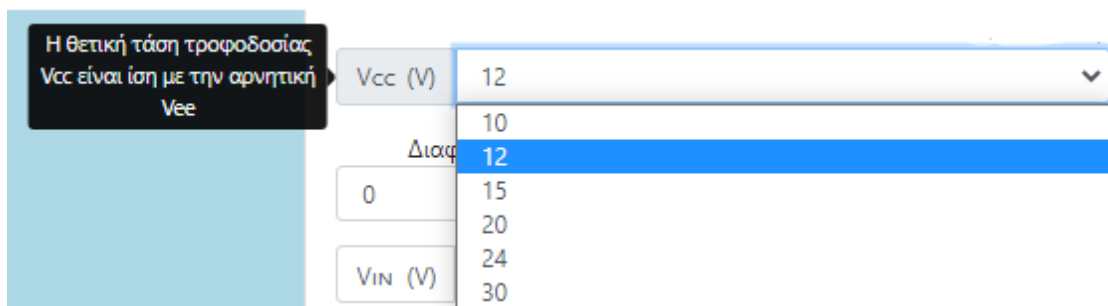
      <InvertingAmplifierr opt={count} > </InvertingAmplifierr>

    </div>
  );
}

Inverting_Amplifier.image = CircuitImg
```

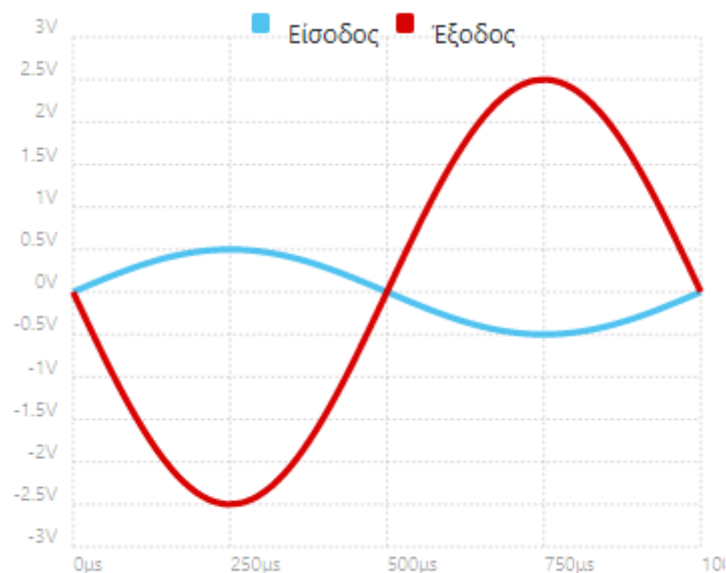
Εικόνα 5.47 Ο κώδικας του index.jsx

Τα πεδία έχουν δημιουργηθεί με ανάλογο τρόπο όπως στα προηγούμενα κυκλώματα με τη βοήθεια του component CircuitValueInput εκτός από το V_{CC} , το dropdown menu που περιέχει τις τιμές για τη διαφορά κορεσμού εξόδου από την τάση τροφοδοσίας, τη συχνότητα f και το SR (πεδίο που εμφανίζεται όταν ο χρήστης επιλέγει την προβολή των σημάτων ως τετραγωνικοί παλμοί και δίνει τη δυνατότητα επιλογής τιμής για το Slew Rate), στα οποία έχει χρησιμοποιηθεί το V_{cc} ValueSelector. Στο πεδίο της τάσης τροφοδοσίας V_{CC} , κάνοντας mouseover ο χρήστης βλέπει ένα κειμενάκι στα αριστερά, το οποίο υπάρχει ουσιαστικά για να εξηγήει ότι οι τροφοδοσίες του τελεστικού έχουν την ίδια απόλυτη τιμή. Π.χ. αν $V_{CC} = 15V$ τότε $V_{EE} = -15V$. Με αυτόν τον τρόπο δεν υπάρχει λόγος ύπαρξης δύο ξεχωριστών πεδίων για την τροφοδοσία. Προγραμματιστικά αυτό έγινε με τη χρήση μιας boolean μεταβλητής x η οποία περνάει στην V_{cc} ValueSelector και αν έχει τιμή true τότε εμφανίζεται το κειμενάκι στα αριστερά.



Εικόνα 5.48 Η εμφάνιση του μηνύματος όταν γίνεται mouseover στη V_{CC}

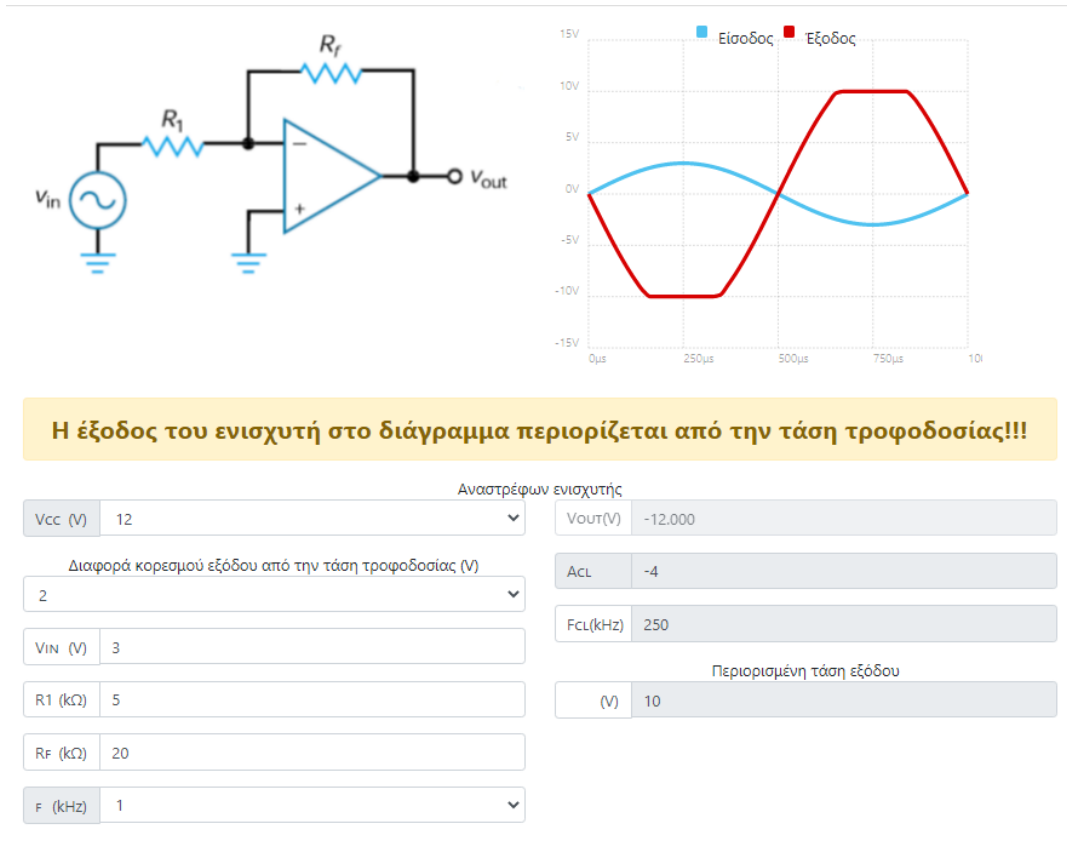
Όσον αφορά τις επιτρεπτές τιμές που μπορούν να καταχωρηθούν στα πεδία, στις αντιστάσεις ισχύει το ίδιο εύρος τιμών με τα προηγούμενα, ενώ η τάση εισόδου V_{in} μπορεί να πάρει τιμές από 0 έως 10V, καθώς σπάνια συναντάται μεγαλύτερη τιμή για εναλλασσόμενο σήμα εισόδου. Για τα πεδία που παίρνουν τιμές από dropdown menu, η τροφοδοσία του ενισχυτή μπορεί να έχει τις τιμές 10, 12, 15, 20, 24, 30V, την ίδια στιγμή που η διαφορά κορεσμού εξόδου από την τάση τροφοδοσίας έχει ως επιλογές το 0, 0.5, 1, 2V. Τέλος, η συχνότητα f , η τιμή της οποίας καθορίζει την περίοδο T , δηλαδή τις τιμές που απεικονίζονται στον άξονα x του γραφήματος, μπορεί να είναι 1, 5 ή 10 kHz, ενώ το SR, το οποίο εμφανίζεται με την επιλογή του button για προβολή ως παλμοί, δύναται να έχει τις τιμές 0.5, 1 και 2V/μs. Η διαδικασία που ακολουθήθηκε για τους υπολογισμούς των αποτελεσμάτων είναι η ίδια με τα κυκλώματα των διπολικών τρανζίστορ, με τη βοήθεια της state μεταβλητής outcomes. Μια διευκρίνιση που αφορά το πεδίο της περιορισμένης τάσης εξόδου είναι ότι πρόκειται για την τιμή μέχρι την οποία επιτρέπεται να φτάνει η έξοδος και προκύπτει από την αφαίρεση της διαφοράς κορεσμού εξόδου από την τάση τροφοδοσίας. Στη δυνατότητα εναλλαγής των μονάδων μέτρησης για τις αντιστάσεις, προστίθεται και η αντίστοιχη για την V_{in} , V_{out} και το πεδίο της περιορισμένης τάσης εξόδου από V σε mV, ενώ για την F_{CL} από Hz σε kHz και σε MHz. Η διαδικασία είναι ακριβώς η ίδια όπως στις αντιστάσεις. Προχωρώντας την ανάλυση, μία πολύ σημαντική λειτουργία της εφαρμογής είναι το γράφημα το οποίο απεικονίζει τα σήματα εισόδου και εξόδου είτε σε ημιτονοειδή είτε σε μορφή τετραγωνικών παλμών όπως προαναφέρθηκε. Η ανάλυσή τους προγραμματιστικά θα γίνει παρακάτω, όπου αναλύονται χωριστά τα αρχεία sine-wave.jsx για το ημίτονο και pulse.jsx για τον τετραγωνικό παλμό. Στην περίπτωση του ημιτόνου, στον άξονα y βρίσκονται οι τιμές για τις τάσεις ενώ στον άξονα x αυτές της περιόδου. Πιο συγκεκριμένα, σύμφωνα και με τη συχνότητα f που έχει επιλέξει ο χρήστης, η απεικόνιση των σημάτων γίνεται πάντα σε χρόνο μίας περιόδου (1000μs αν η συχνότητα είναι 1kHz, 200μs αν είναι 5kHz και 100μs αν είναι 10kHz). Το σήμα εισόδου είναι χρωματισμένο με μπλε χρώμα, ενώ της εξόδου με κόκκινο. Ο χρωματισμός και οι πληροφορίες για το υπόμνημα βρίσκονται στο αρχείο styles.scss.



Εικόνα 5.49 Ημιτονοειδή σήματα για την είσοδο και την έξοδο

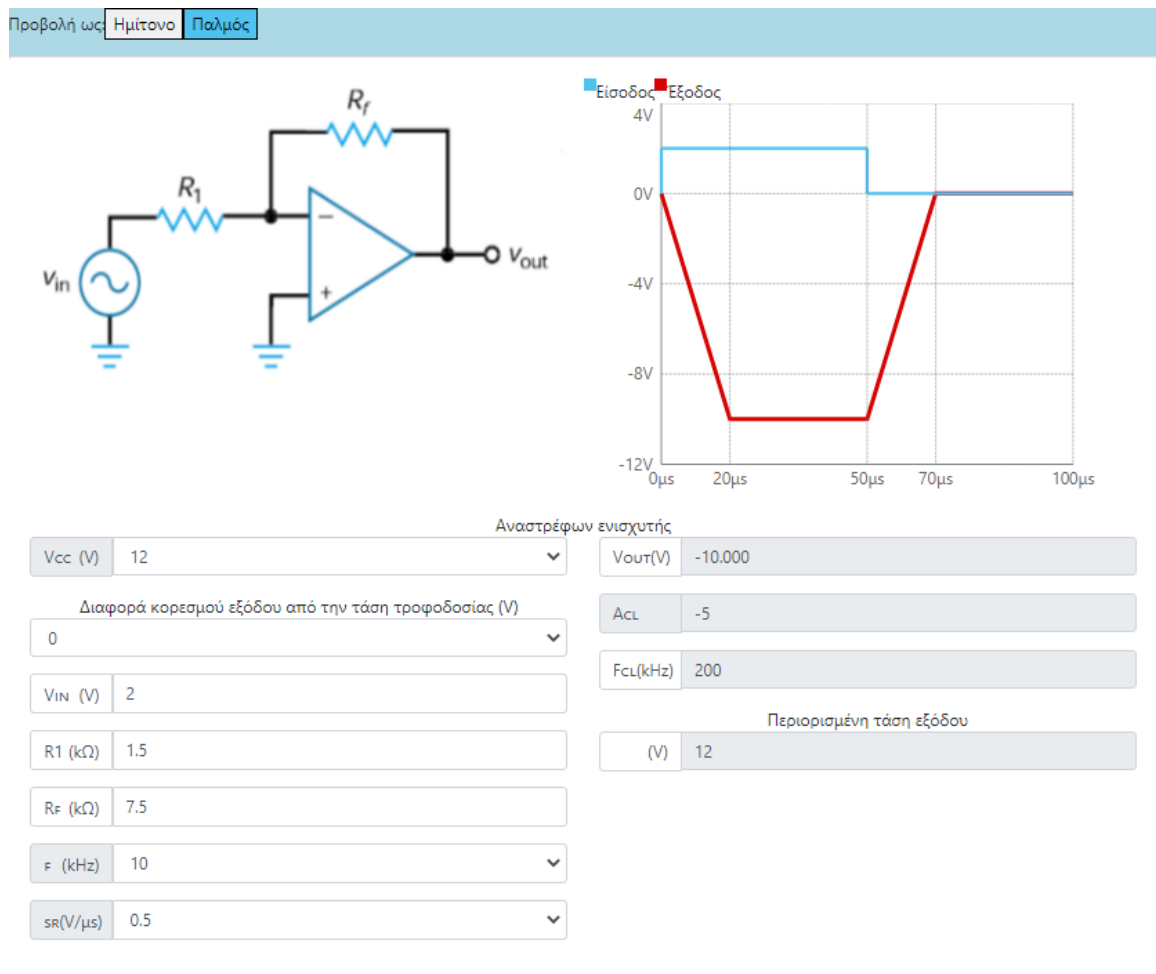
Επειδή η τιμή της εξόδου που προκύπτει είναι συνυφασμένη με την τροφοδοσία του ενισχυτή, καθώς δεν μπορεί να είναι μεγαλύτερη από αυτήν, όταν συμβαίνει κάτι τέτοιο, εμφανίζεται στην οθόνη προειδοποιητικό μήνυμα, με τρόπο αντίστοιχο όπως στο κύκλωμα της πόλωσης διαιρέτη τάσης.

Καθώς η V_{out} είναι αυτή η οποία είναι εκτός ορίων, όταν ξεπερνάει την τάση τροφοδοσίας, αναβοσβήνει όπως ακριβώς και η περιοχή λειτουργίας στο VDB. Στην περίπτωση που συμβούν τα παραπάνω, το σήμα εξόδου ψαλιδίζεται στο σημείο όπου επιτρέπει η περιορισμένη τάση εξόδου.



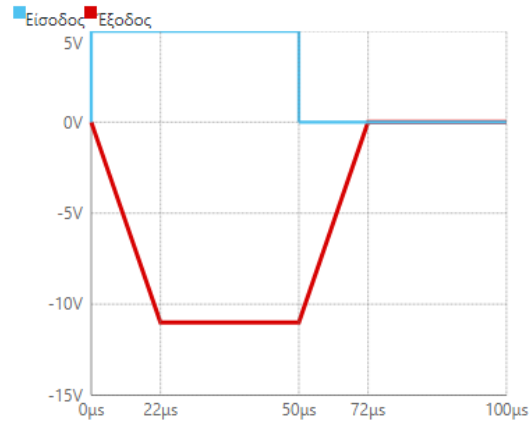
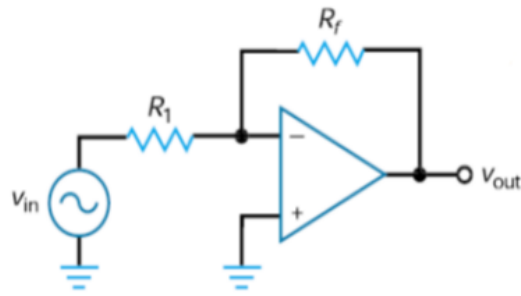
Εικόνα 5.50 Προειδοποιητικό μήνυμα και ψαλιδισμός του σήματος εξόδου

Όταν επιλεγεί το κουμπί για προβολή ως παλμός, ο χρήστης βλέπει στην οθόνη το εξής:



Εικόνα 5.51 Προβολή σημάτων ως παλμοί

Όπως και στην ημιτονοειδή μορφή, το σήμα εισόδου είναι χρωματισμένο με μπλε ενώ της εξόδου με κόκκινο. Οι τιμές στους δύο άξονες αντιπροσωπεύουν τα ίδια πράγματα και συγκεκριμένα τις τάσεις ο άξονας y και την περίοδο ο άξονας x. Το σήμα εισόδου φτάνει μέχρι την V_{in} που έχει δοθεί, πηγαίνει μέχρι τα μισά της περιόδου και από εκεί κατεβαίνει στα 0V και συνεχίζει μέχρι το τέλος αυτής. Όσο για την έξοδο, αυτή σχηματίζεται με τη βοήθεια του SR που θα δώσει ο χρήστης. Πιο συγκεκριμένα, οι τιμές στον άξονα x διαμορφώνονται από την τιμή του SR σε συνδυασμό με τη συχνότητα f. Έτσι λοιπόν, το σήμα φτάνει μέχρι την έξοδο σε χρόνο Δt και στα μισά της περιόδου ανεβαίνει με ρυθμό Δt στα 0V συνεχίζοντας ως το τέλος. Το προειδοποιητικό μήνυμα στην περίπτωση που η έξοδος υπερβεί την τάση τροφοδοσίας σύμφωνα και με την διαφορά κορεσμού εξόδου εμφανίζεται και εδώ, με την V_{out} να αναβοσβήνει επίσης.



Η έξοδος του ενισχυτή στο διάγραμμα περιορίζεται από την τάση τροφοδοσίας!!!

Αναστρέφων ενισχυτής

V _{CC} (V)	12	V _{out} (V)	-25.000
Διαφορά κορεσμού εξόδου από την τάση τροφοδοσίας (V)			
1			
V _{IN} (V)	5	A _{CL}	-5
R ₁ (kΩ)	1.5	F _{CL} (kHz)	200
R _F (kΩ)	7.5	Περιορισμένη τάση εξόδου	
F (kHz)	10	(V)	11
s _R (V/μs)	0.5		

Εικόνα 5.52 Προειδοποιητικό μήνυμα

Στον κώδικα, η εμφάνιση του προειδοποιητικού μηνύματος γίνεται με τη βοήθεια μιας boolean μεταβλητής bool. Η τιμή της αλλάζει από true σε false όταν η V_{out} υπερβεί την περιορισμένη τάση εξόδου και ακολούθως εμφανίζεται το μήνυμα. Χρησιμοποιείται επίσης και για το blinking της V_{out} περνώντας ως props στο CircuitValueInput.

```
<CircuitValueInput values={outcomes} name='Vout' measurements={measurementsV} disabled fixed shouldBlink = {!bool} />
```

Εικόνα 5.53 Η μεταβλητή bool περνάει ως props στο CircuitValueInput για το blinking της V_{out}

Υπάρχει επίσης και μια μεταβλητή x η οποία κρατάει είτε την V_{out} είτε την περιορισμένη τάση εξόδου, ανάλογα με το αν η V_{out} έχει ή όχι επιτρεπτή τιμή. Η μεταβλητή αυτή χρησιμοποιείται για να περάσει στο component SineWave την αντίστοιχη τιμή και να ζωγραφιστεί αναλόγως το ημίτονο. Αντίστοιχη διαδικασία συμβαίνει και στον παλμό όπου στέλνονται οι παράμετροι μέσω props στο Pulse για τον σχηματισμό του.

```
var bool = true
var x = outcomes.Vout

if ((outcomes.Vout < -outcomes.diff)) {bool=false; x = -outcomes.diff}
```

Εικόνα 5.54 Η boolean μεταβλητή bool και η μεταβλητή x που χρησιμοποιείται για τα γραφήματα

```
{
  (bool? null :
  <Alert className='text-center' variant='warning'><div style={{ fontSize: '160%' }}><b>Εξόδος του ενισχυτή στο διάγραμμα περιορίζεται από την τάση τροφοδοσίας!!!</b></div></Alert>
}
```

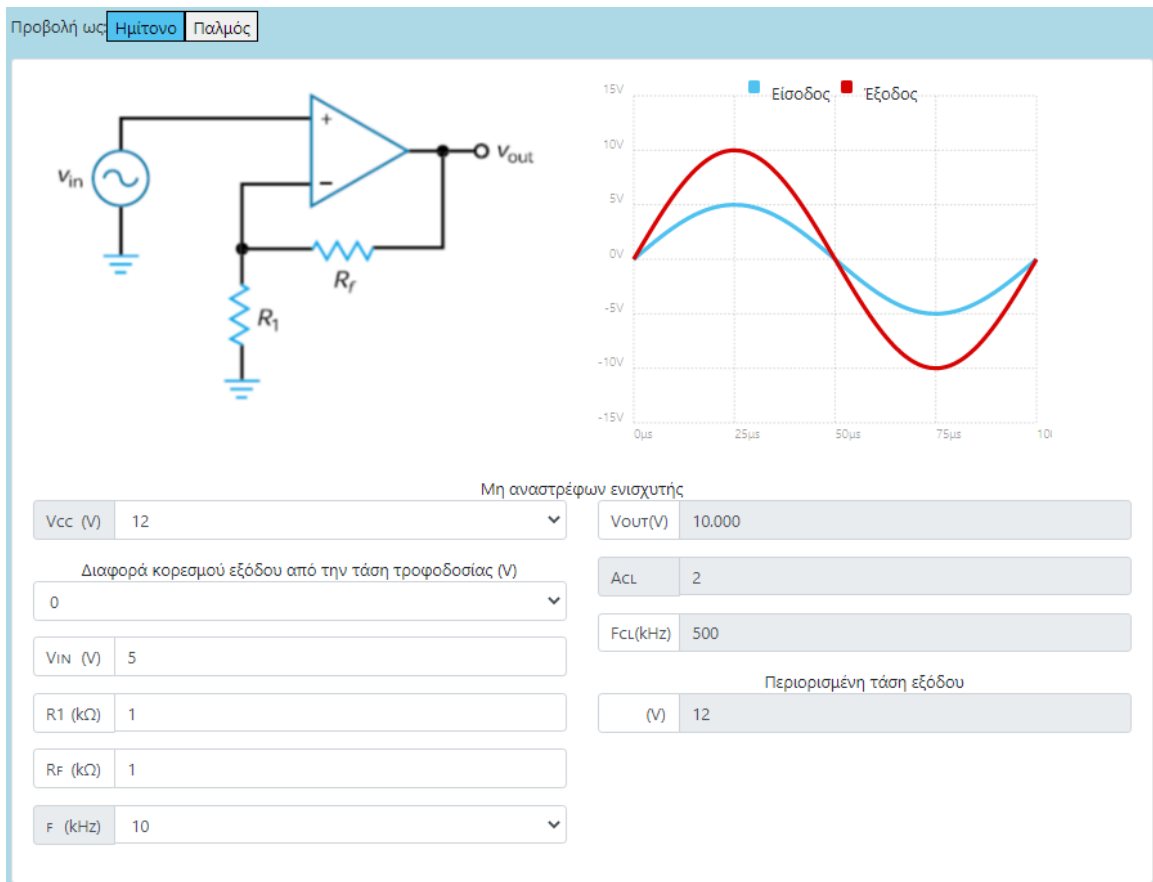
Εικόνα 5.55 Ο κώδικας για την εμφάνιση του προειδοποιητικού μηνύματος

```
{opt === 0 ?
<Col xs={5}>
  <SineWave frequency={values.f} voltages={[values.Vin, outcomes.Vout]} value={Math.abs(x)} />
</Col> :
<Col xs={3}>
  <Pulse sr={values.sr} frequency={values.f} voltages={[values.Vin, x]} />
</Col> }
```

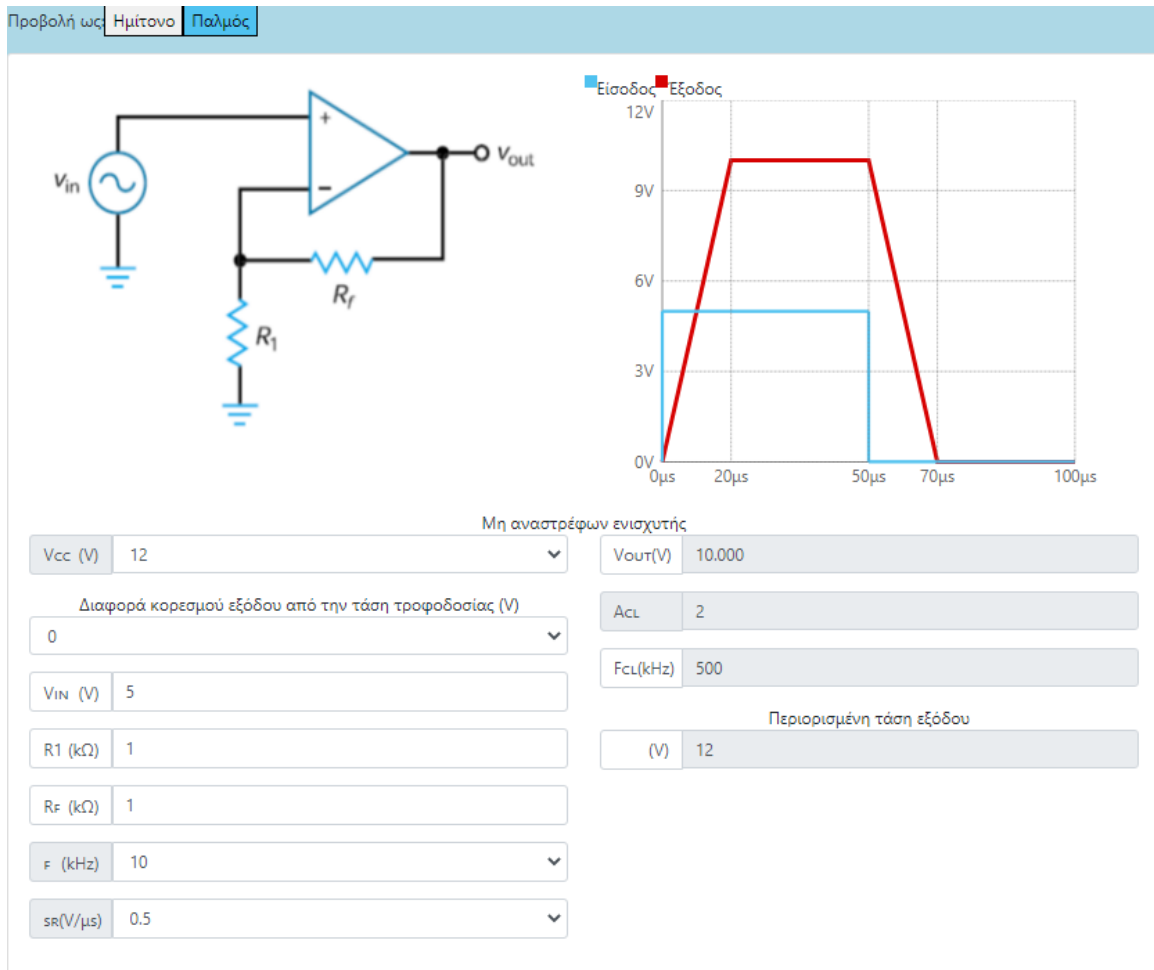
Εικόνα 5.56 Το πέρασμα δεδομένων μέσω props στα SineWave και Pulse ανάλογα με την τιμή του opt

5.4.3.4 Τα κυκλώματα μη-αναστρέφοντος, αθροιστικού και ενισχυτή διαφορών (Non-Inverting, Summing, Difference Amplifier)

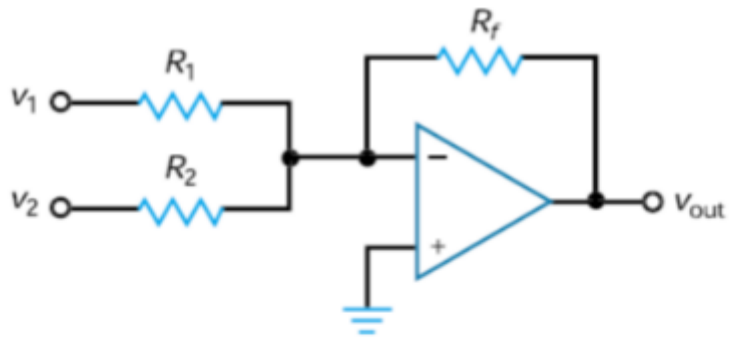
Τα υπόλοιπα κυκλώματα των τελεστικών ενισχυτών έχουν ακριβώς την ίδια λογική και τρόπο υλοποίησης με του αναστρέφοντος ενισχυτή. Η μόνη διαφορά είναι ότι στον μη αναστρέφων ενισχυτή τα σήματα εισόδου και εξόδου είναι συμφασικά μεταξύ τους και αυτό φαίνεται κι από την απεικόνισή τους στο γράφημα, την ίδια στιγμή που στον αθροιστικό ενισχυτή και στον ενισχυτή διαφορών δεν υπάρχει γράφημα.



Εικόνα 5.57 Μη αναστρέφων ενισχυτής – προβολή σημάτων ως ημίτονα



Εικόνα 5.58 Μη αναστρέφων ενισχυτής – προβολή σημάτων ως παλμοί



Αθροιστικός ενισχυτής

V_{CC} (V) 12

Διαφορά κορεσμού εξόδου από την τάση τροφοδοσίας (V)

0

V1 (V) 0.1

V2 (V) 0.2

R1 (kΩ) 20

R2 (kΩ) 10

R_F (kΩ) 100

Υπολογισμοί

V_{out}(V) -2.500

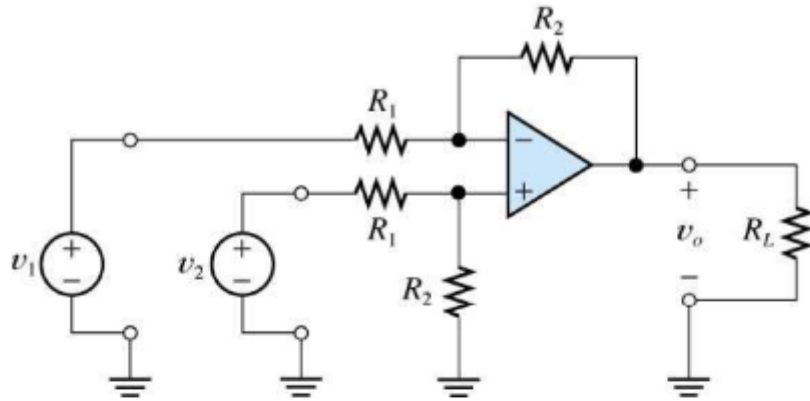
A_{c1} -5

A_{c2} -10

Περιορισμένη τάση εξόδου

(V) 12

Εικόνα 5.59 Αθροιστικός ενισχυτής



Εισχυτής διαφορών

Vcc (V)	12
Διαφορά κορεσμού εξόδου από την τάση τροφοδοσίας (V)	
	0
V1 (V)	0.1
V2 (V)	0.2
R1 (kΩ)	20
R2 (kΩ)	10
Υπολογισμοί	
Vout(V)	0.050
Acl	0.5
Περιορισμένη τάση εξόδου	
(V)	12

Εικόνα 5.60 Εισχυτής διαφορών

5.4.4 Ο φάκελος με τα συστατικά (components)

5.4.4.1 Το συστατικό CircuitValueInput

Το πιο σημαντικό για τη λειτουργία της εφαρμογής component είναι το `CircuitValueInput`. Επιτελεί σχεδόν κάθε λειτουργία των κυκλωμάτων, των πεδίων τους κλπ. Με τη χρήση props δίνονται default τιμές στις παραμέτρους που χρησιμοποιούνται για τη μεταβίβαση των πληροφοριών από τα κυκλώματα στο `CircuitValueInput`. Το `skipValidations` που φαίνεται στην παρακάτω εικόνα παίρνει

για αρχική τιμή το disabled, ούτως ώστε στα read-only πεδία που δεν μπορεί να εκχωρηθεί κάποια τιμή από το χρήστη να μην γίνει επιχείρηση για validation.

```
const { name, values, setValues, fixed, measurement, measurements, min, max, skipValidations } = {
  setValues: () => {},
  min: -Infinity,
  max: Infinity,
  measurements: [{
    symbol: props.measurement,
    restore: v => v,
    render: v => v
  }],
  skipValidations: props.disabled,
  ...props
}
```

Εικόνα 5.61 Απόδοση default τιμών

Προχωρώντας, έχουμε τη συνάρτηση restore, η οποία έχει οριστεί στους πίνακες που χρησιμεύουν στην εναλλαγή μεταξύ των μονάδων μέτρησης στα αρχεία των κυκλωμάτων. Η υλοποίησή της στο CircuitValueInput γίνεται με τη βοήθεια μιας state μεταβλητής currentMeasurement, η οποία χρησιμοποιείται ως μετρητής στο array measurements. Έτσι λοιπόν, καλείται κάθε φορά η αντίστοιχη restore από τον πίνακα με τη μονάδα μέτρησης που έχουμε εκείνη τη στιγμή και με τη χρήση regular expressions και των συναρτήσεων toString () και match () χρησιμοποιείται μόνο το νούμερο χωρίς τη μονάδα.

```
const restore = value => measurements[currentMeasurement]
  ? measurements[currentMeasurement].restore(
    parseFloat(
      value.toString().match(/(^d+(\.\d+)?)?/)[0] || 0
    )
  )
  : 0
```

Εικόνα 5.62 Η συνάρτηση restore

Ακολουθεί ο πίνακας με τα validations των τιμών. Χρησιμοποιείται για να ελέγχεται αν ο χρήστης εισήγαγε κάποια μη έγκυρη τιμή. Για την υλοποίησή του χρησιμοποιήθηκαν κι εδώ regular expressions. Οι έλεγχοι που γίνονται αφορούν την υποχρεωτική καταχώρηση αριθμού και την πιθανή χρήση υποδιαστολής ανάμεσα στους αριθμούς σε περίπτωση που ο χρήστης θέλει να εισάγει δεκαδικό αριθμό και πραγματοποιούνται με τη βοήθεια της συνάρτησης test () η οποία ελέγχει κάθε φορά το value. Στον ίδιο πίνακα μπαίνουν και τα όρια για την ελάχιστη και την μέγιστη τιμή (min και max props τα οποία περνάνε από την index κάθε κυκλώματος στο CircuitValueInput) μέσω της κλήσης της αντίστοιχης restore.

```

const validations = [
  // Check for invalid characters
  value => !/^\.(\\d)/.test(value),
  value => !/\\. {2,}/.test(value),
  value => !/\\. ([^\\d])/ .test(value),
  value => /^\\d+(\\.\\d+)?$/ .test(value),

  // Limit to min value
  value => restore(value) >= min,

  // Limit to max value
  value => restore(value) <= max
]

```

Εικόνα 5.63 Ο πίνακας validations

Η επόμενη λειτουργία αφορά την εμφάνιση δεκαδικών ψηφίων στα αποτελέσματα. Επειδή πρόκειται για μια διαδικασία η οποία εκτελείται κάθε φορά που αλλάζουν τα values, πραγματοποιείται μέσα σε `useEffect ()`. Αρχικά, γίνεται έλεγχος της τιμής που εμφανίζεται βάσει του πίνακα `measurements`, αν υπάρχει το τρέχον `measurement` και αν ισχύει, καλείται η `update` συνάρτηση `setInputValue`, η οποία έχει δημιουργηθεί μαζί με τη μεταβλητή `inputValue` και έχει ως αρχική τιμή το τρέχον `input`. Μέσα στη `setInputValue` καλείται η αντίστοιχη συνάρτηση `render` από την `index` του κυκλώματος που χρησιμοποιείται εκείνη τη στιγμή και γίνεται ο έλεγχος εάν η τιμή του πεδίου είναι `fixed` τότε να εμφανίζονται τα πρώτα τρία δεκαδικά στο αποτέλεσμα, αλλιώς να μένει όπως είναι.

```

useEffect(() => measurements[currentMeasurement] && setInputValue(
  measurements[currentMeasurement].render(
    fixed
      ? Number.parseFloat(values[name]).toFixed(3)
      : values[name]
  )
), [values, currentMeasurement, setInputValue])

```

Εικόνα 5.64 Εμφάνιση δεκαδικών στα αποτελέσματα

Κατόπιν, ακολουθεί το `circuitValueGroup`. Πρόκειται για ένα `element` το οποίο περιέχει στοιχεία για το `styling` των πεδίων. Πιο συγκεκριμένα, μέσω του `shouldBlink` που περνάει από την `index` του εκάστοτε κυκλώματος, στέλνει την πληροφορία για το αν πρέπει να γίνει `blinking` το πεδίο ή όχι. Πρακτικά αυτό αφορά το πεδίο της περιοχής λειτουργίας στα διπολικά τρανζίστορ και της εξόδου V_{out} στους τελεστικούς ενισχυτές. Μέσα στο `<InputGroup.Text>` καθορίζεται το `backgroundColor` που έχει το μήνυμα που εμφανίζεται στο χρήστη για αλλαγή μονάδων μέτρησης και γίνεται όταν το `measurements.length > 1`, σε κάθε περίπτωση δηλαδή, καθώς ο πίνακας `measurements` που κρατάει τις μονάδες μέτρησης έχει πάντα μήκος μεγαλύτερο από 1. Μέσω της `onClick ()` γίνεται `update` η μονάδα μέτρησης. Στην πράξη συμβαίνει όταν ο χρήστης κάνει κλικ και αλλάζει τη μονάδα. Επειδή θέλουμε κάθε φορά όταν φτάνει στην τελευταία μονάδα να μπορεί να ξαναπάει στην πρώτη, μέσα σε μια `useEffect ()`, γίνεται ο αντίστοιχος έλεγχος αν το `currentMeasurement` είναι μεγαλύτερο ή ίσο από το `length` του array `measurements`.

```
onClick={() => setCurrentMeasurement(currentMeasurement + 1)}
```

Εικόνα 5.65 Η onClick που κάνει update τη μονάδα μέτρησης

```
const [currentMeasurement, setCurrentMeasurement] = useState(0)
useEffect(
  () => setCurrentMeasurement(currentMeasurement >= measurements.length ? 0 : currentMeasurement),
  [currentMeasurement, setCurrentMeasurement]
)
```

Εικόνα 5.66 Η useEffect που κάνει τον έλεγχο currentMeasurement με measurements.length

Στη συνέχεια καλείται η συνάρτηση `getLabel()`. Χρησιμοποιείται για τη σωστή εμφάνιση των labels και της μονάδας μέτρησης αριστερά του πεδίου που εισάγεται τιμή. Αναλυτικότερα, αν πρόκειται για το πεδίο της περιοχής λειτουργίας εμφανίζει μόνο την ομόνυμη φράση, ενώ για τα υπόλοιπα, μέσω του props `allowLower`, εμφανίζονται πιο όμορφα τα κεφαλαία γράμματα σαν small-caps.

The image shows a single-line text input field. On the left side of the field, the text "Re (kΩ)" is displayed in a smaller font. The input field itself contains the number "1".

Εικόνα 5.67 Εμφάνιση του E στην Re σαν κεφαλαίο small-cap

Για το πεδίο της περιορισμένης τάσης εξόδου στους τελεστικούς ενισχυτές δεν εμφανίζεται τίποτα, καθώς υπάρχει label από πάνω. Ακολουθως, εμφανίζει την αντίστοιχη μονάδα μέτρησης από δίπλα. Αυτό το κάνει μέσα από έναν έλεγχο για το αν υπάρχει `currentMeasurement` και `symbol`, καθώς στην περίπτωση για παράδειγμα του β ή της περιοχής λειτουργία δεν χρειάζεται η εμφάνιση μονάδας.

```
const getLabel = () => {
  var label
  if (name === 'bjtMode') label = props.label
  else if (name === 'diff') label = ''
  else label = name
  const labelElem = <span style={{ fontVariantCaps: props.allowLower ? 'normal' : 'small-caps' }}>{label}</span>

  if (measurements[currentMeasurement] && measurements[currentMeasurement].symbol.length) {
    return (
      <>
        {labelElem}
        <span style={{ marginLeft: 'auto' }}>{measurements[currentMeasurement].symbol}</span>
      </>
    )
  } else {
    return <>{labelElem}</>
  }
}
```

Εικόνα 5.68 Η συνάρτηση `getLabel()`

Προχωρώντας στην ανάλυση του `circuitValueGroup`, είναι υπεύθυνο να κάνει disabled τα πεδία των αποτελεσμάτων και να κοκκινίζει οποιοδήποτε δεν έχει έγκυρη τιμή. Αυτό γίνεται μέσα σε ένα `<FormControl>`, ενώ μέσω της state μεταβλητής `inputIsValid` και της τιμής της γίνεται η εμφάνιση του μηνύματος σε κάθε πεδίο όταν εισάγεται μη αποδεκτή τιμή, παρουσιάζοντας και τις αντίστοιχες ελάχιστες και μέγιστες.

```

const circuitValueGroup = (
  <InputGroup className={'mb-3 ' + (props.shouldBlink ? 'blink' : '')}>
    <InputGroup.Prepend>
      <InputGroup.Text
        style={{
          width: props.label ? undefined : '5rem',
          userSelect: 'none',
          backgroundColor: measurements.length > 1 ? 'var(--bs-primary)' : undefined
        }}
        onClick={() => setCurrentMeasurement(currentMeasurement + 1)}
      >
        {getLabel()}
      </InputGroup.Text>
    </InputGroup.Prepend>
    <FormControl
      value={inputValue}
      onChange={event => setInputValue(event.target.value)}
      disabled={props.disabled}
      isValid={!inputIsValid}
    />
    {inputIsValid || (
      <FormControl.Feedback type='invalid' tooltip>
        Μη έγκυρη τιμή
        <ul style={{ padding: 0 }}>
          {!!props.min && <li>Ελάχιστο: {props.min} {measurement}</li>}
          {!!props.max && <li>Μέγιστο: {props.max} {measurement}</li>}
        </ul>
      </FormControl.Feedback>
    )}
  </InputGroup>
)

```

Εικόνα 5.69 Το circuitValueGroup

Τέλος, το circuitValueGroup καλείται στο return του CircuitValueInput, στο οποίο εμφανίζεται επίσης το κείμενο για προτροπή στον χρήστη να κάνει κλικ στην περίπτωση που θέλει να αλλάξει μονάδα μέτρησης. Για να εμφανιστούν οι σωστές μονάδες μέτρησης ανάλογα με το αν πρόκειται για τάση ή αντίσταση χρησιμοποιείται η συνάρτηση map () πάνω στον πίνακα measurements.

```

return (
  measurements.length > 1
  ? (
    <OverlayTrigger
      placement='left'
      overlay={
        <Tooltip>
          Κάνε κλικ για αλλαγή μονάδας μέτρησης.
          Διαθέσιμες μονάδες:
          <ul style={{ padding: 0 }}>
            {measurements.map(m => (<li key={m.symbol}>{m.symbol}</li>))}
          </ul>
        </Tooltip>
      }
    >
      {circuitValueGroup}
    </OverlayTrigger>
  )
  : circuitValueGroup
)
}

```

Εικόνα 5.70 Το κείμενο για προτροπή αλλαγής μονάδας μέτρησης

5.4.4.2 Το συστατικό VccValueSelector

Επόμενο component είναι το VccValueSelector. Η ονομασία του προέκυψε επειδή αρχικά χρησιμοποιούνταν μόνο για την τιμή της V_{CC} , είτε στα διπολικά τρανζίστορ, είτε στους τελεστικούς ενισχυτές, αλλά στην πορεία προέκυψαν και άλλα πεδία που έπρεπε να παίρνουν προκαθορισμένες τιμές από ένα dropdown menu. Πιο συγκεκριμένα, εκτός από την V_{CC} χρησιμοποιείται και για το πεδίο της διαφοράς κορεσμού εξόδου από την τάση τροφοδοσίας και της συχνότητας f στους τελεστικούς ενισχυτές. Η λογική της λειτουργίας του είναι σχετικά απλή, καθώς ουσιαστικά κάνει δύο πράγματα, εμφανίζει το dropdown menu με τις επιθυμητές τιμές και κατόπιν, αν το V_{CC} αντιπροσωπεύει τάση τροφοδοσίας σε τελεστικό ενισχυτή, εμφανίζεται αριστερά το κείμενο που εξηγεί ότι $V_{CC} = V_{EE}$. Αναλυτικότερα, αφού περαστούν τα props από κάθε κύκλωμα, γίνεται έλεγχος εάν το measurement έχει τιμή, κι αν αυτό συμβαίνει τότε εμφανίζεται το πεδίο με το dropdown menu με τη βοήθεια και της συνάρτησης `map()`. Ακολούθως, γίνεται η ίδια διαδικασία, με τη διαφορά της προσθήκης του μηνύματος για την ισότητα των τάσεων τροφοδοσίας στην περίπτωση που υπάρχει η μεταβλητή x , η οποία προέρχεται από την ανάλυση των κυκλωμάτων των τελεστικών ενισχυτών και το πεδίο είναι το V_{CC} . Ο λόγος είναι επειδή το V_{CC} υπάρχει και στα διπολικά τρανζίστορ και το μήνυμα πρέπει να εμφανίζεται μόνο στις V_{CC} των τελεστικών ενισχυτών. Τέλος, στην περίπτωση που ο χρήστης κάνει mouseover αλλού, δεν χρειάζεται να εμφανίζεται κάποιο μήνυμα οπότε μέσα στο else εμφανίζεται το πεδίο με την ονομασία και τη μονάδα μέτρησής του αριστερά (στην περίπτωση που πρόκειται για το V_{CC} ή το f). Ο κώδικας που υλοποιεί τις παραπάνω λειτουργίες εντοπίζεται στο παράρτημα της εργασίας.

5.4.4.3 Το συστατικό SineWave

Το component που αφορά την απεικόνιση των ημιτονοειδών σημάτων στα κυκλώματα του αναστρέφοντος και μη αναστρέφοντος ενισχυτή είναι το SineWave. Η βιβλιοθήκη που χρησιμοποιήθηκε για την υλοποίησή του είναι η Chartist μαζί με κάποια plugins όπως και στο VDB. Η λογική για τη δημιουργία των καμπυλών ήταν η δημιουργία πολλών μικρών σημείων τα οποία στη συνέχεια θα ενωθούν. Έτσι λοιπόν, μέσα σε μια useEffect (), δημιουργούνται 50 μικρά σημεία, υπολογίζεται η περίοδος με τη βοήθεια της συχνότητας που περνάει ως prop και ορίζονται τα διαστήματα [0, T/4, T/2, 3T/4, T] μέσα στα οποία δημιουργούνται τα σημεία. Κατόπιν, δημιουργούνται οι τιμές στους άξονες x και y. Όσον αφορά τη σωστή απεικόνιση των σημάτων σύμφωνα με τα δεδομένα που έχει εισάγει ο χρήστης, γίνονται κάποιοι έλεγχοι για το αν η τιμή στον άξονα y είναι μεγαλύτερη από την περιορισμένη τάση εξόδου και αν η μεταβλητή V που χρησιμοποιείται για προσπέλαση στη συνάρτηση map () είναι ίση με την τάση εξόδου, αν πρόκειται για το σήμα εξόδου δηλαδή και όχι για αυτό της εισόδου. Η ίδια διαδικασία γίνεται και για τις αρνητικές τιμές. Αυτό χρησιμεύει στο να ψαλιδίζεται το σήμα εξόδου όταν η V_{out} βγαίνει εκτός του επιτρεπτού ορίου. Η μεγαλύτερη κάθε φορά τάση καταχωρείται στη μεταβλητή refVoltage. Κατόπιν, μέσω της setPlotOptions και με τη βοήθεια της βιβλιοθήκης Chartist και των plugins της σβήνονται τα σημεία που δημιουργήθηκαν πάνω στην καμπύλη, καθώς αυτά δεν χρειάζονται πλέον. Επιπλέον προστίθενται στα νούμερα στους άξονες x και y οι αντίστοιχες μονάδες μέτρησης για την περίοδο και την τάση. Τέλος, στο return του component ζωγραφίζεται το ημίτονο, χρησιμοποιώντας και το αρχείο styles.scss για πληροφορίες για το υπόμνημα, το χρώμα κλπ μέσω του attribute `className = 'ct-octave'`. Και εδώ πέρα, ο αναγνώστης μπορεί να δει τον κώδικα για το SineWave στο παράρτημα που βρίσκεται στο τέλος του συγγράμματος.

5.4.4.4 Το συστατικό Pulse

Όσον αφορά την απεικόνιση των σημάτων εισόδου και εξόδου ως τετραγωνικοί παλμοί χρησιμοποιείται το component Pulse. Για την υλοποίησή του χρησιμοποιήθηκε η βιβλιοθήκη recharts και όχι η Chartist όπως στο ημίτονο. Ο λόγος είναι το γεγονός ότι η Chartist δεν μπορεί να δημιουργήσει γραμμές διαγώνια όπως χρειάζεται στον παλμό. Κάτι που παρατηρείται είναι ότι ο παλμός της εξόδου μεταβάλλεται μόνο ως προς τον άξονα y και όχι ως προς τον x. Αυτό συμβαίνει διότι η React δεν τρέχει on real-time (δεν είναι δηλαδή πρόγραμμα προσομοίωσης). Έτσι λοιπόν, για τη δημιουργία των τιμών χρησιμοποιείται ο πίνακας plotData, στον οποίο υπάρχουν τα δεδομένα για τα σημεία στα οποία θα ζωγραφιστούν οι γραμμές στους άξονες και οι γραμμές για τους παλμούς εισόδου και εξόδου. Ουσιαστικά αυτό που γίνεται είναι να δημιουργούνται ξεχωριστές γραμμές οι οποίες ενώνονται μεταξύ τους. Μέσα στο return ζωγραφίζονται οι γραμμές, ενώ δημιουργείται ξεχωριστό υπόμνημα γιατί το default θα περιλαμβάνει την κάθε γραμμή ξεχωριστά. Για να γίνει το γράφημα πιο ευπαρουσίαστο και κατανοητό, χρησιμοποιήθηκαν ξεχωριστά grid για τις γραμμές διότι δεν χρειάζονται σε όλα τα σημεία, παρά μόνο σε αυτά που υπάρχουν τιμές. Οι πληροφορίες για χρωματισμό των γραμμών και του υπομνήματος βρίσκονται, όπως και στο ημίτονο, στο styles.scss. Τέλος, αφού γίνει η προσθήκη των μονάδων μέτρησης στους άξονες, σβήνονται οι τελείες που υπάρχουν σε συγκεκριμένα σημεία στον παλμό εξόδου, καθώς δεν χρειάζονται πλέον. Το σημείο που δημιουργούνται εναλλάξ οι δύο κάθετες και οι δύο οριζόντιες γραμμές για τους δύο παλμούς είναι το εξής:

```
<ReferenceLine isAnimationActive={false} strokeWidth={3} stroke="#4fc2f0" dot={false} segment={{ x: 0, y: 0 }, { x: 0, y: voltages[0] }} />  
<Line isAnimationActive={false} strokeWidth={3} dataKey="in" stroke="#4fc2f0" dot={false} />  
<ReferenceLine isAnimationActive={false} strokeWidth={3} stroke="#4fc2f0" alwaysShow={true} dot={false} segment={{ x: t/2, y: voltages[0] }, { x: t/2, y: 0 }} />  
<Line isAnimationActive={false} strokeWidth={3} dataKey="in2" stroke="#4fc2f0" dot={false} />
```

Εικόνα 5.71 Το σημείο που ενώνονται οι γραμμές των παλμών εισόδου

Όπως και σε προηγούμενα components, ο πλήρης κώδικας για τη δημιουργία των τετραγωνικών παλμών συναντάται στο παράρτημα στο τέλος της εργασίας.

5.5 Επίλογος

Συνοψίζοντας, στο παρόν κεφάλαιο παρουσιάστηκε η δομή του κώδικα της εφαρμογής σε ακολουθία με την ανάλυση αυτού και των επιμέρους λειτουργιών της. Έγινε ξεχωριστή ανάλυση του κώδικα κάθε κυκλώματος και των αρχείων που συμβάλλουν στην τέλεση των λειτουργιών του καθενός. Τα παραπάνω συνοδεύονται από τις κατάλληλες εικόνες λειτουργίας και κομμάτια του κώδικα, έτσι ώστε να δίνεται η ευκαιρία στον αναγνώστη να κατανοεί πλήρως το εύρος των δυνατοτήτων που προσφέρει η εφαρμογή.

Κεφάλαιο 6ο: Αξιολόγηση λειτουργίας εφαρμογής

6.1 Εισαγωγή

Η ιδέα για τη δημιουργία της εφαρμογής περιελάμβανε δύο στάδια, αυτό της υλοποίησης και αυτό της αξιολόγησης του παραγόμενου λογισμικού. Σε αυτό το κεφάλαιο, μέσα από μια έρευνα που διεξήχθη, πραγματοποιείται η ανάλυση των αποτελεσμάτων που προέκυψαν και αφορούν την αξιολόγηση της εφαρμογής. Για τις ανάγκες της έρευνας δημιουργήθηκε ένα ηλεκτρονικό ερωτηματολόγιο αποτελούμενο από 9 ερωτήσεις, το οποίο διαμοιράστηκε ηλεκτρονικά σε φοιτητές του τμήματος, αλλά και σε άτομα τα οποία έχουν ασχοληθεί με το αντικείμενο και τις έννοιες που περιέχει η εφαρμογή. Στόχος των ερωτήσεων ήταν να αξιολογήσουν τον βαθμό ικανοποίησης των χρηστών σε ζητήματα όπως η ευχρηστία, η λειτουργικότητα και η πιθανή μελλοντική χρήση από την πλευρά των φοιτητών για τις ανάγκες των μαθημάτων “Ηλεκτρονικά Κυκλώματα” και “Ηλεκτρονικές Διατάξεις”, αλλά και οπουδήποτε αλλού κρίνεται ικανό βοήθημα.

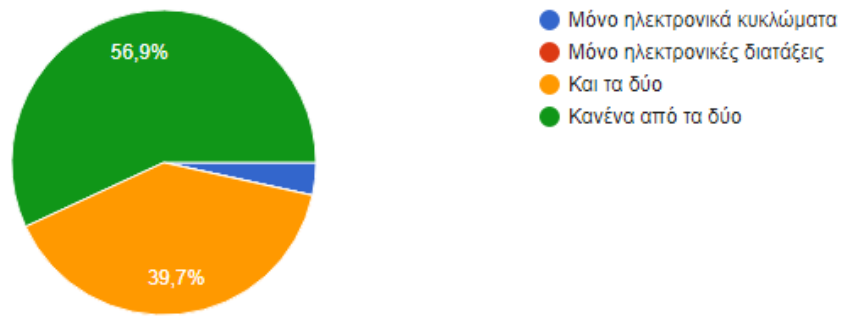
6.2 Ανάλυση αποτελεσμάτων

Οι ερωτήσεις στις οποίες κλήθηκαν να απαντήσουν οι ενδιαφερόμενοι ήταν οι εξής:

- 1) Έχετε περάσει τα μαθήματα ηλεκτρονικά κυκλώματα και ηλεκτρονικές διατάξεις;
- 2) Σε τι βαθμό σας άρεσε η σχεδίαση της εφαρμογής;
- 3) Θεωρείτε ότι η εφαρμογή είναι εύχρηστη;
- 4) Θεωρείτε ότι η εφαρμογή είναι λειτουργική;
- 5) Πιστεύετε ότι η εφαρμογή μπορεί να αποτελέσει συμπληρωματικό βοήθημα στα παραπάνω μαθήματα;
- 6) Σε τι βαθμό πιστεύετε ότι η εφαρμογή μπορεί να βοηθήσει στην καλύτερη κατανόηση των εννοιών των παραπάνω μαθημάτων;
- 7) Σε τι βαθμό σκοπεύετε να την χρησιμοποιήσετε για τη μελέτη των παραπάνω μαθημάτων;
- 8) Πιστεύετε ότι θα έπρεπε κάποιος να έχει προηγούμενες γνώσεις για να χρησιμοποιήσει την εφαρμογή;
- 9) Θα συστήνατε την εφαρμογή σε κάποιον που ασχολείται με τα παραπάνω μαθήματα ή με περιεχόμενο που περιλαμβάνεται σε αυτήν;

Για μεγαλύτερη εγκυρότητα των αποτελεσμάτων, ζητήθηκε από τον κάθε χρήστη να συμπληρώσει το mail του ούτως ώστε να αποφευχθεί κατά το δυνατόν η υποβολή του ερωτηματολογίου από το ίδιο άτομο περισσότερες από μία φορές. Το εύρος των απαντήσεων που είχαν στη διάθεσή τους οι χρήστες ξεκινούσε από το “Καθόλου” και έφτανε μέχρι το “Πάρα πολύ”. Συνολικά συγκεντρώθηκαν 58 απαντήσεις από άτομα τα οποία περιηγήθηκαν στην εφαρμογή. Τα αποτελέσματα τα οποία αναδείχθηκαν από την έρευνα φαίνονται παρακάτω και προέκυψαν με τη βοήθεια του google form.

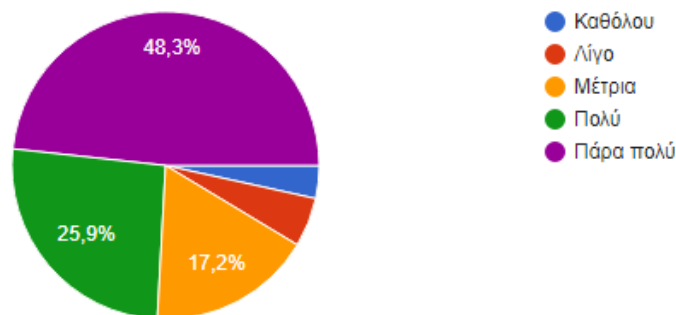
1. Έχετε περάσει τα μαθήματα ηλεκτρονικά κυκλώματα και ηλεκτρονικές διατάξεις;



Εικόνα 6.1 Επιτυχία ή μη στα παραπάνω μαθήματα

Όπως παρατηρείται από την εικόνα 6.1, το συντριπτικό μέρος των απαντήσεων που αφορούν την επιτυχία ή μη στα μαθήματα “Ηλεκτρονικά κυκλώματα” και “Ηλεκτρονικές διατάξεις” εναλλάσσονται μεταξύ ατόμων που είτε πέρασαν και τα δύο είτε κανένα. Το γεγονός αυτό οδηγεί στο συμπέρασμα ότι το εύρος των απαντήσεων που ακολουθούν στις επόμενες ερωτήσεις προέρχεται και από άτομα που παρακολούθησαν με επιτυχία τα δύο μαθήματα και από άτομα που δεν έχουν περάσει κάποιο από αυτά, χωρίς αυτό να σημαίνει απαραίτητα ότι έχουν κοπεί σε εξέταση, καθώς πιθανόν μερικοί να παρακολουθούν κάποιο από τα μαθήματα για πρώτη φορά.

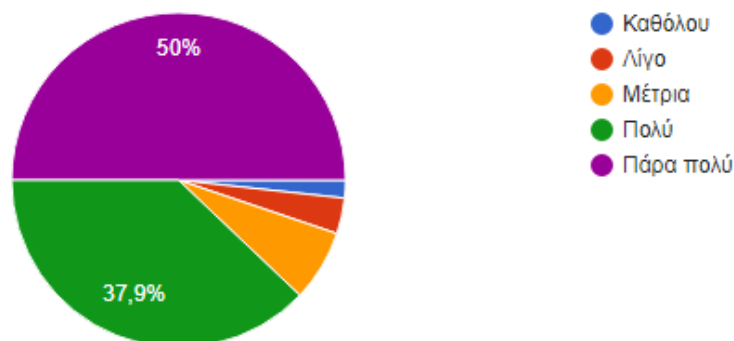
2. Σε τι βαθμό σας άρεσε η σχεδίαση της εφαρμογής;



Εικόνα 6.2 Βαθμός ικανοποίησης για τη σχεδίαση της εφαρμογής

Στην ερώτηση που αφορά τον βαθμό ικανοποίησης των χρηστών ως προς την σχεδίαση της εφαρμογής, οι απαντήσεις “Πάρα πολύ” και “Πολύ” συγκέντρωσαν αθροιστικά ένα ποσοστό της τάξης του 75% περίπου, κάτι που κρίνεται ιδιαίτερα ικανοποιητικό.

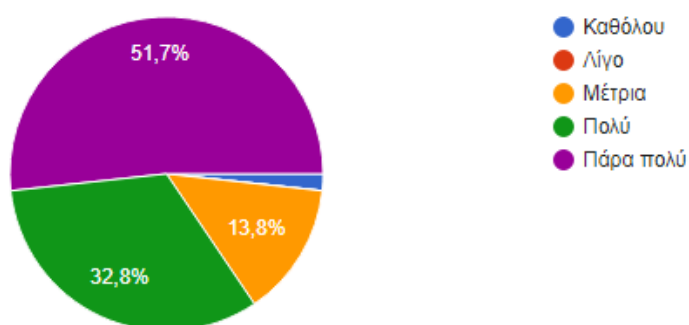
3. Θεωρείτε ότι η εφαρμογή είναι εύχρηστη;



Εικόνα 6.3 Ευχρηστία εφαρμογής

Οι απαντήσεις που λήφθηκαν στη συγκεκριμένη ερώτηση κυμαίνονται στην πλειοψηφία τους μεταξύ του “Πολύ” και του “Πάρα πολύ”. Το αποτέλεσμα είναι ιδιαίτερα ενθαρρυντικό, καθώς η ευχρηστία της εφαρμογής ήταν μια σημαντική παράμετρος η οποία λήφθηκε υπόψη κατά την υλοποίησή της, λόγω του ότι προορίζεται για εκπαιδευτικό εργαλείο μάθησης και θα πρέπει να είναι απλό και εύκολο στη χρήση του.

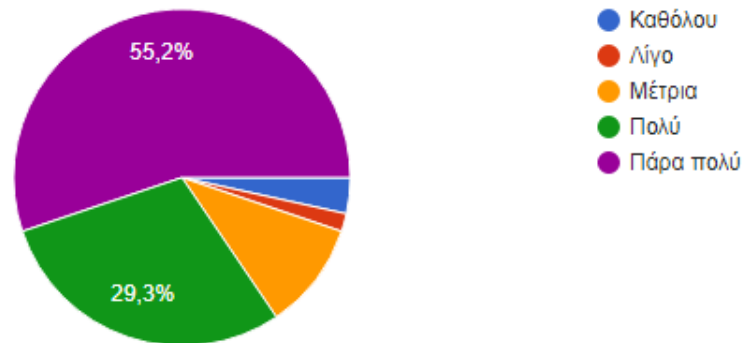
4. Θεωρείτε ότι η εφαρμογή είναι λειτουργική;



Εικόνα 6.4 Λειτουργικότητα εφαρμογής

Παρόμοια με το βαθμό ευχρηστίας της εφαρμογής ήταν και τα αποτελέσματα που αφορούν τη λειτουργικότητα. Πιο συγκεκριμένα, οι απαντήσεις “Πολύ” και “Πάρα πολύ” συγκέντρωσαν αθροιστικά περίπου το 85% των αποτελεσμάτων. Σε συνδυασμό με το μεγάλο ποσοστό του βαθμού ευχρηστίας, προκύπτει το συμπέρασμα ότι η εφαρμογή δεν είναι περίπλοκη και οι λειτουργίες που επιτελεί είναι εύκολες στην εκτέλεσή τους.

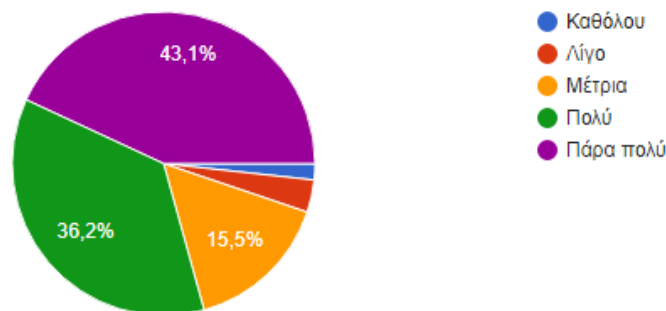
5. Πιστεύετε ότι η εφαρμογή μπορεί να αποτελέσει συμπληρωματικό βοήθημα στα παραπάνω μαθήματα;



Εικόνα 6.5 Απαντήσεις στην ερώτηση για τη χρήση της εφαρμογής ως συμπληρωματικό βοήθημα στα παραπάνω μαθήματα

Στη συγκεκριμένη ερώτηση πρέπει να ληφθεί υπόψη το γεγονός ότι στις απαντήσεις περιλαμβάνονται και κάποιοι που πιθανόν δεν γνωρίζουν την ύλη των παραπάνω μαθημάτων, αλλά αναγνωρίζοντας τις υπό μελέτη έννοιες, καθώς το δείγμα δεν αφορά τελείως άσχετα με το περιεχόμενο άτομα, οδηγήθηκαν στο να δώσουν απάντηση. Από τα ποσοστά που φαίνονται, η μεγάλη πλειοψηφία θεωρεί πως η εφαρμογή μπορεί να χρησιμοποιηθεί συμπληρωματικά για τη μελέτη των παραπάνω μαθημάτων.

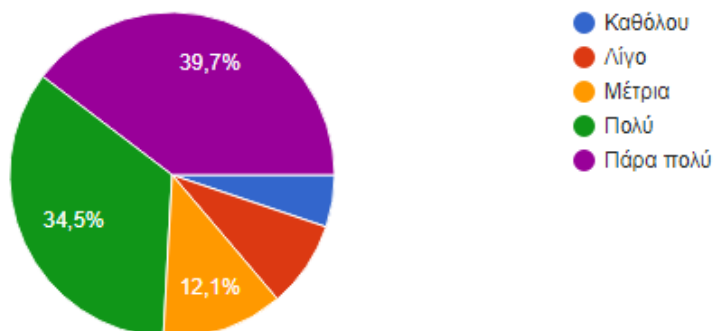
6. Σε τι βαθμό πιστεύετε ότι η εφαρμογή μπορεί να βοηθήσει στην καλύτερη κατανόηση των εννοιών των παραπάνω μαθημάτων;



Εικόνα 6.6 Χρήση ως βοήθημα για την καλύτερη κατανόηση των εννοιών των μαθημάτων

Προχωρώντας, με τις απαντήσεις που λήφθηκαν στην παραπάνω ερώτηση, ικανοποιείται ένας ακόμη στόχος που τέθηκε κατά τη δημιουργία της εφαρμογής, να βοηθήσει δηλαδή στην καλύτερη κατανόηση του περιεχομένου των μαθημάτων. Αυτό βέβαια δεν σημαίνει ότι δεν μπορεί να βοηθήσει και οποιονδήποτε άλλο την χρησιμοποιεί και ενδιαφέρεται να κατανοήσει καλύτερα το υλικό που περιλαμβάνει, κάτι που φαίνεται από το συνολικό ποσοστό που έδωσε ως απαντήσεις το “Πολύ” και το “Πάρα πολύ”, δηλαδή περίπου το 80%.

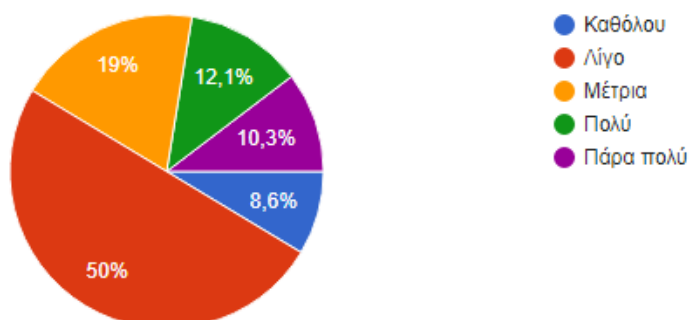
7. Σε τι βαθμό σκοπεύετε να την χρησιμοποιήσετε για τη μελέτη των παραπάνω μαθημάτων;



Εικόνα 6.7 Χρήση της εφαρμογής για τη μελέτη των παραπάνω μαθημάτων

Αναμενόμενα υψηλά είναι τα ποσοστά και σε αυτήν την ερώτηση, καθώς αποτελεί συμπλήρωμα των δύο προηγούμενων ερωτήσεων. Το ποσοστό των ατόμων που απάντησαν “Πολύ” και “Πάρα πολύ” στην επικείμενη χρήση της εφαρμογής προκειμένου να μελετήσουν τα παραπάνω μαθήματα είναι περίπου 75%.

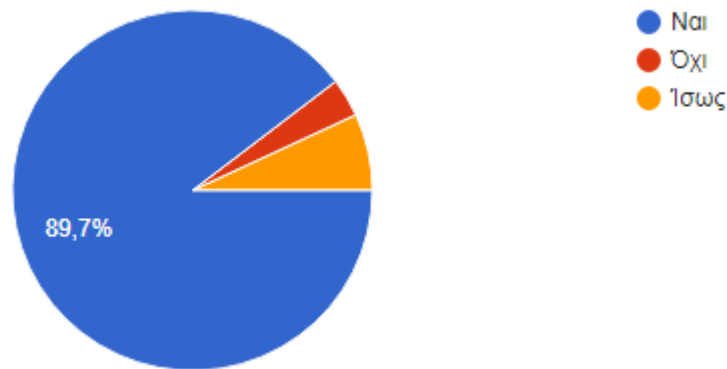
8. Πιστεύετε ότι θα έπρεπε κάποιος να έχει προηγούμενες γνώσεις για να χρησιμοποιήσει την εφαρμογή;



Εικόνα 6.8 Απαντήσεις στην ερώτηση για απαίτηση προηγούμενων γνώσεων πριν τη χρήση της εφαρμογής

Αντιστρόφως ανάλογα είναι τα ποσοστά τα οποία προκύπτουν στην ερώτηση για απαίτηση προηγούμενων γνώσεων προκειμένου να χρησιμοποιηθεί η εφαρμογή. Ακριβώς οι μισοί έχουν απαντήσει “Λίγο”, ενώ ένα 30% περίπου ακόμα έχει δώσει ως απαντήσεις το “Μέτρια” και το “Καθόλου”. Υπάρχει κι ένα 20% περίπου που θεωρεί πως απαιτούνται προηγούμενες γνώσεις. Το συμπέρασμα που προκύπτει εδώ είναι ότι η πλειοψηφία θεωρεί πως η εφαρμογή μπορεί να χρησιμοποιηθεί και από άτομα τα οποία τώρα ξεκινάνε τη διαδικασία μάθησης των παραπάνω εννοιών και δεν έχουν κάποιο ιδιαίτερο θεωρητικό υπόβαθρο. Συνεπώς, κάθε ενδιαφερόμενος μπορεί να προβεί στη χρήση της.

9. Θα συστήνατε την εφαρμογή σε κάποιον που ασχολείται με τα παραπάνω μαθήματα ή με περιεχόμενο που περιλαμβάνεται σε αυτήν;

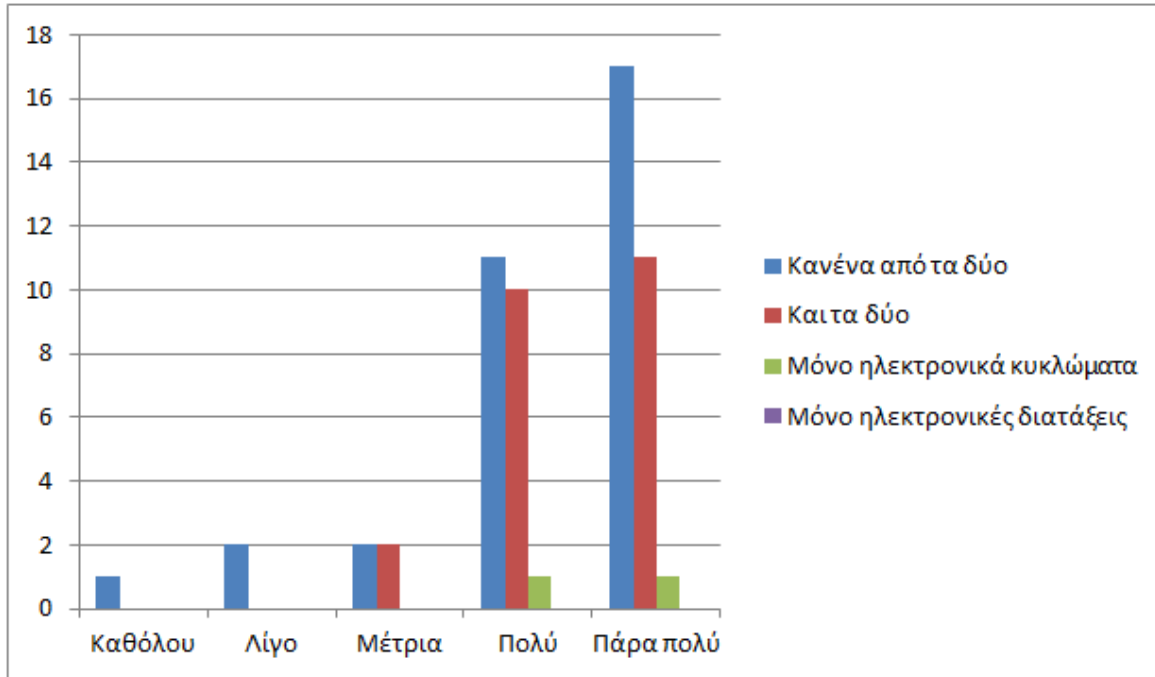


Εικόνα 6.9 Σύσταση εφαρμογής σε τρίτους

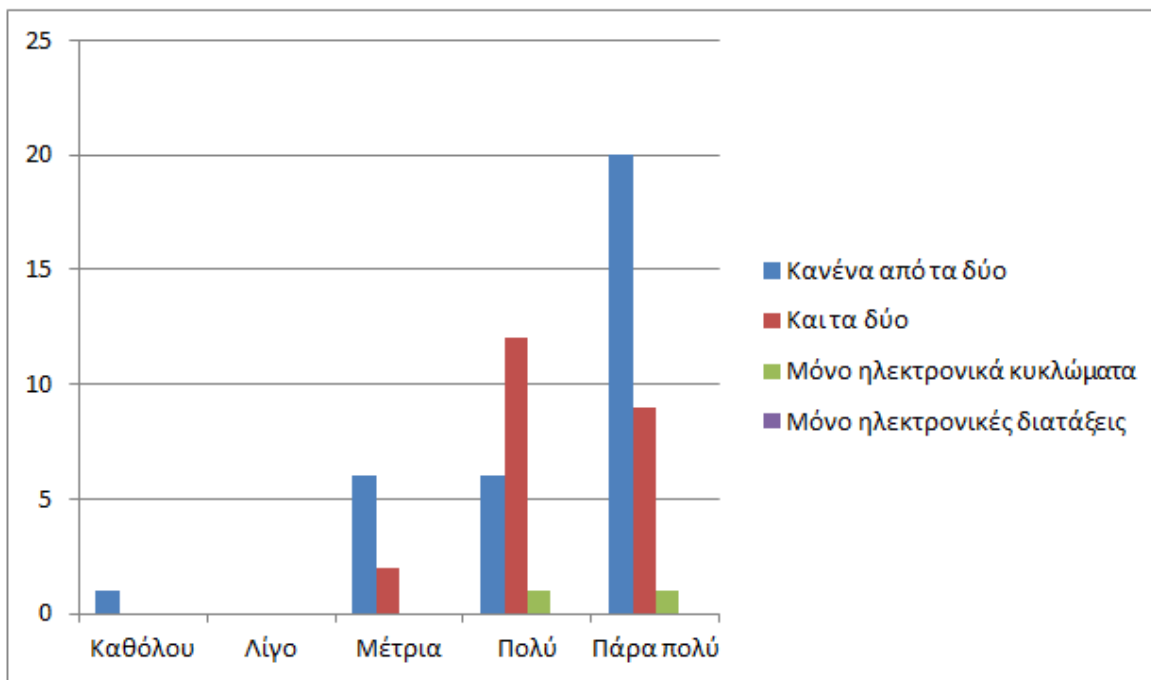
Στην τελευταία ερώτηση, που αφορά το κατά πόσο θα συστήνατε την εφαρμογή σε άτομα που ασχολούνται με τα παραπάνω μαθήματα ή με περιεχόμενο που περιλαμβάνεται στην εφαρμογή, δεν χωράει αμφιβολία ότι αξίζει να προταθεί σε κάποιον που ασχολείται με αυτά, καθώς περίπου το 90% έδωσε θετική απάντηση.

6.3 Γενική αποτίμηση αποτελεσμάτων

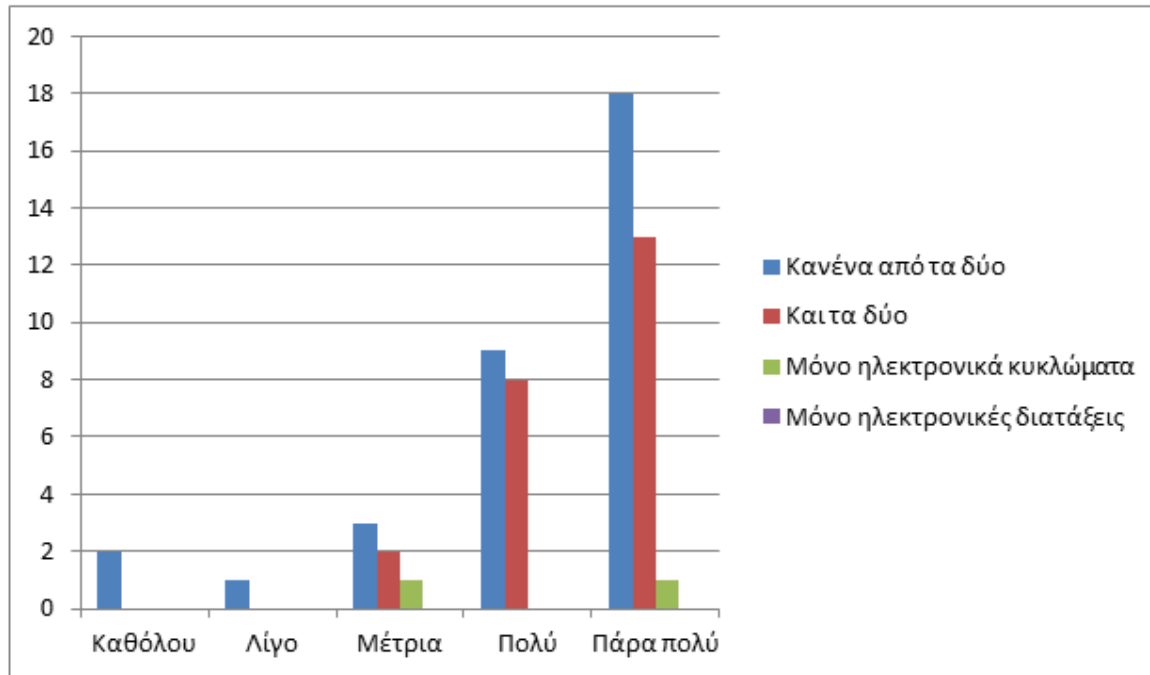
Συμπερασματικά, τα αποτελέσματα έδειξαν ότι οι χρήστες είναι κατά γενική ομολογία αρκετά ικανοποιημένοι με την υλοποίηση της εφαρμογής. Έτσι λοιπόν, φαίνεται ότι ικανοποιούνται πλήρως οι στόχοι που τέθηκαν κατά τη δημιουργία της εφαρμογής, καθώς θεωρήθηκε, από τα άτομα που απάντησαν, ένα σημαντικό εργαλείο στην προσπάθεια για κατανόηση των γνώσεων που απαιτούνται για τα παραπάνω μαθήματα και γενικά για άτομα που πιστεύουν ότι χρειάζονται βοήθεια στην εκμάθηση των εν λόγω εννοιών. Τέλος, ως συμπέρασμα από τα παραπάνω μπορεί να προκύψει και το γεγονός της πιθανής μελλοντικής χρήσης της εφαρμογής ως εκπαιδευτικό εργαλείο από καθηγητές στην προσπάθεια για σωστή μεταλαμπάδευση των γνώσεων, καλύτερη εξήγηση και πιθανώς γρηγορότερη επίλυση παραδειγμάτων, κάτι που ήταν επίσης στους επιδιωκόμενους στόχους. Συνοψίζοντας την αποτίμηση των αποτελεσμάτων, κρίθηκε σκόπιμη η δημιουργία ραβδογραμμάτων, τα οποία παρουσιάζουν με πιο εύληπτο και γραφικό τρόπο την εξαγωγή ειδικότερων συμπερασμάτων από τα αποτελέσματα της παραπάνω έρευνας. Πιο συγκεκριμένα, μελετήθηκε η συσχέτιση όσων πέρασαν ή δεν πέρασαν τα μαθήματα “Ηλεκτρονικά κυκλώματα” ή/και “Ηλεκτρονικές διατάξεις” με το βαθμό ικανοποίησης όσον αφορά την ευχρηστία, την λειτουργικότητα και την πιθανή χρήση της εφαρμογής ως συμπληρωματικό βοήθημα. Τα γραφήματα που προέκυψαν παρουσιάζονται παρακάτω:



Εικόνα 6.10 Συσχέτιση επιτυχόντων και μη με το βαθμό ευχρηστίας της εφαρμογής

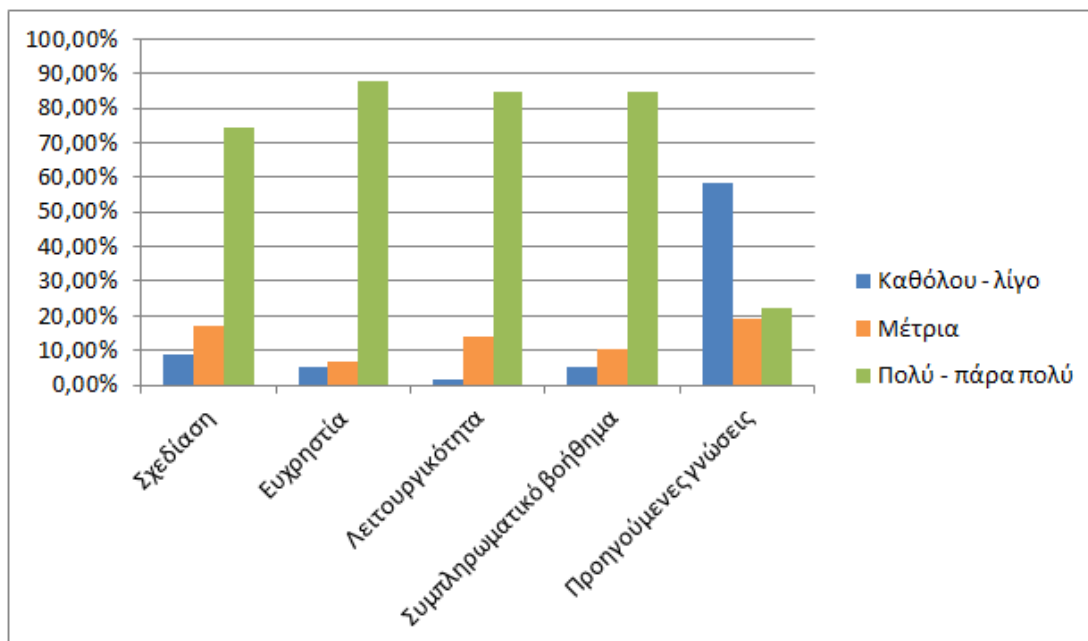


Εικόνα 6.11 Συσχέτιση επιτυχόντων και μη με το βαθμό λειτουργικότητας της εφαρμογής



Εικόνα 6.12 Συσχέτιση επιτυχόντων και μη με την πιθανή χρήση της εφαρμογής ως συμπληρωματικό βοήθημα

Τέλος, αποφάνθηκε χρήσιμη η παρουσίαση του γενικότερου πλαισίου στο οποίο κινούνται οι απαντήσεις στο παραπάνω ερωτηματολόγιο. Αυτό έγινε και εδώ με την απεικόνιση της μορφής ραβδογραμμάτων, όπου παρουσιάζονται συγκεντρωτικά τα αποτελέσματα στις διάφορες κατηγορίες ερωτήσεων που τέθηκαν. Το γράφημα που προέκυψε φαίνεται παρακάτω:



Εικόνα 6.13 Γράφημα απεικόνισης των αποτελεσμάτων συνολικά

Κεφάλαιο 7ο: Συμπεράσματα και προτάσεις βελτίωσης

7.1 Συμπεράσματα

Κατά γενική ομολογία, η συνολική εμπειρία από την υλοποίηση της εφαρμογής κρίνεται ιδιαίτερα ικανοποιητική. Αρχικά, με βοήθησε να αναπτύξω σε μεγάλο βαθμό τις γνώσεις μου πάνω σε θέματα προγραμματισμού και μου έδωσε το έναυσμα και την όρεξη να θέλω να ασχοληθώ ακόμη περισσότερο. Έπειτα, το γεγονός της δημιουργίας μιας εφαρμογής η οποία μπορεί να αποτελέσει πολύ χρήσιμο εγχειρίδιο και βοήθημα στην προσπάθεια συμφοιτητών μου για ορθή παρακολούθηση και εκμάθηση των εννοιών που περιλαμβάνει, προσφέρει και ηθική ικανοποίηση. Με την υλοποίησή της θεωρώ πως επιτεύχθηκαν σε μεγάλο βαθμό οι στόχοι οι οποίοι αποτέλεσαν το εναρκτήριο λάκτισμα για την εφαρμογή της ιδέας. Καθότι η τεχνολογία εξελίσσεται διαρκώς και οι ανάγκες για πρακτική επίλυση ζητημάτων αυξάνονται, η παρούσα εφαρμογή είναι σίγουρα ένα εργαλείο που μπορεί να χρησιμοποιηθεί από πληθώρα φοιτητών και όχι μόνο. Η ευχρηστία και η απλότητα με την οποία παρουσιάζει τα κυκλώματα είναι ικανή να δώσει κίνητρο για ενασχόληση με τα μαθήματα τα οποία πραγματεύεται. Χάρη στη χρήση του framework της React, το οποίο θεωρώ ότι είναι εξαιρετικά εύχρηστο, τα κυκλώματα παρουσιάστηκαν σαν components, κάτι που δίνει τη δυνατότητα στον προγραμματιστή να χειριστεί πιο εύκολα τον κώδικα λόγω της συνοχής και της ορθής δομής του.

7.2 Προτάσεις βελτίωσης

Όπως κάθε εφαρμογή και σύστημα επιδέχεται βελτίωση, έτσι κι εδώ, μπορούν να γίνουν κάποιες τροποποιήσεις που πιθανόν θα κάνουν καλύτερη την αλληλεπίδραση και την εμπειρία του χρήστη. Μερικές από αυτές είναι:

- Προσθήκη περισσότερων κυκλωμάτων προς εξέταση. Ο τρόπος που είναι υλοποιημένη η εφαρμογή βοηθάει σημαντικά σε αυτό, καθώς αυτό που χρειάζεται είναι στον φάκελο με τα components των κυκλωμάτων να προστεθούν τα ζητούμενα. Ενδεικτικά, θα μπορούσαν να προστεθούν τα κυκλώματα πόλωσης με διπλή τροφοδοσία, με ανάδραση συλλέκτη, με ανάδραση εκπομπού, το κύκλωμα του ενισχυτή κοινού εκπομπού, το κύκλωμα του ακόλουθου εκπομπού, και οποιοδήποτε άλλο θεωρηθεί χρήσιμο για τη μελέτη των μαθημάτων “Ηλεκτρονικά Κυκλώματα” και “Ηλεκτρονικές Διατάξεις”, αλλά και όχι μόνο.
- Τροποποίηση της εφαρμογής για ποιοτικότερη χρήση από κινητές συσκευές, καθώς δεν χρησιμοποιεί κάποιο template που να την κάνει responsive και συνεπώς η ολοκληρωμένη εμπειρία περιορίζεται στη χρήση μέσω Η/Υ.
- Προσθήκη διαγράμματος απόκρισης συχνοτήτων σε συνδυασμό με τον υπολογισμό του κέρδους A_{CL} σε dB, προκειμένου να δίνεται η δυνατότητα για υπολογισμό της τάσης εξόδου σε συγκεκριμένες συχνότητες. Προφανώς, έτσι όπως είναι διαμορφωμένο το UI της εφαρμογής, θα είναι λίγο δύσκολο να προστεθεί καινούριο διάγραμμα, οπότε θα ήταν καλό να γίνουν και μερικές αλλαγές στην σχεδίαση για το σκοπό αυτό.
- Εύρεση πληρέστερης βιβλιοθήκης για το σχεδιασμό της γραμμής φορτίου στο κύκλωμα πόλωσης διαιρέτη τάσης. Προκειμένου να μπορέσουν να απεικονιστούν σωστά όλα τα σημεία (και το σημείο κορεσμού και το σημείο ηρεμίας και το σημείο αποκοπής), η γραμμή φορτίου παίρνει μια κλίση και δεν εμφανίζεται πάντοτε σαν ευθεία γραμμή. Αυτό συμβαίνει συνήθως σε περιπτώσεις όπου το ρεύμα κορεσμού προκύπτει αρκετά χαμηλή τιμή. Οπότε θα ήταν χρήσιμη η μελέτη για εύρεση κάποιας ίσως πιο κατάλληλης βιβλιοθήκης που να διορθώνει το ζήτημα αυτό.
- Εύρεση πληρέστερης βιβλιοθήκης για το σχεδιασμό των σημάτων ως τετραγωνικοί παλμοί στους τελεστικούς ενισχυτές. Εδώ πέρα, το σήμα εξόδου δεν μετατοπίζεται ως προς τον άξονα x, με συνέπεια η κλίση του σήματος να μην μεταβάλλεται. Τα αποτελέσματα που προκύπτουν είναι σωστά, αλλά μια έρευνα πάνω σε αυτό το ζήτημα για εύρεση κάποιου τρόπου να αντιμετωπιστεί το ζήτημα θα ήταν θεμιτή, καθώς υπάρχουν και περιπτώσεις όπου

Κεφάλαιο 7

δεν εμφανίζονται όλα τα σημεία στον άξονα x όταν πρόκειται για υπερβολικά χαμηλές τιμές του Δ_t .

- Προσθήκη διαγραμμάτων για απεικόνιση των σημάτων στα κυκλώματα του αθροιστικού και του ενισχυτή διαφορών, κάτι που δεν έγινε λόγω της ύπαρξης παραπάνω από ενός σημάτων εισόδου.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Internet Site

- [1] IGI Global – What is Virtual Lab. [Online]. Available: <https://www.igi-global.com/dictionary/virtual-lab/31699>
- [2] Virtual Labs: An Initiative of Ministry of Education Under the National Mission on Education through ICT – Objectives, The Philosophy [Online]. Available: <https://www.vlab.co.in/>
- [3] Basic Electronics Virtual Laboratory. [Online]. Available: <http://vlabs.iitkgp.ac.in/be/>
- [4] Differences between NPN & PNP Transistors and their Making, 2019. [Online]. Available: <https://www.watelectronics.com/difference-between-npn-and-pnp-transistor/>
- [5] Σαν σήμερα στους υπολογιστές - Τρανζίστορ, 2012. [Online]. Available: <https://sansimeracomputers.wordpress.com/2012/12/23/dec23-1947/>
- [6] <https://el.wikipedia.org/wiki/%CE%A4%CF%81%CE%B1%CE%BD%CE%B6%CE%AF%CF%83%CF%84%CE%BF%CF%81>
- [7] Τόμπρας Γεώργιος, “Εισαγωγή στους Τελεστικούς ενισχυτές τάσης”, 2015. [Online]. Available: <https://repository.kallipos.gr/handle/11419/2042>
- [8] 10 Best ReactJS Practices For a Good React Project, 2022, [Online]. Available: <https://www.geeksforgeeks.org/10-best-reactjs-practices-for-a-good-react-project/>
- [9] <https://code.visualstudio.com>
- [10] microsoft/vscode, Github, [Online]. Available: <https://github.com/microsoft/vscode>
- [11] Visual Studio Code FAQ, Visual Studio Code, [Online]. Available: <https://code.visualstudio.com/docs/supporting/FAQ>
- [12] Visual Studio Code, [Online]. Available: https://en.m.wikipedia.org/wiki/Visual_Studio_Code
- [13] Tutorial: Intro to React, [Online]. Available: <https://reactjs.org/tutorial/tutorial.html>
- [14] What is React, w3schools, [Online]. Available: https://www.w3schools.com/whatis/whatis_react.asp
- [15] React (JavaScript library), Wikipedia, [Online]. Available: [https://en.m.wikipedia.org/wiki/React_\(JavaScript_library\)](https://en.m.wikipedia.org/wiki/React_(JavaScript_library))
- [16] facebook/react, Github, [Online]. Available: <https://github.com/facebook/react>
- [17] Differences between Functional Components and Class Components in React, [Online]. Available: <https://www.geeksforgeeks.org/differences-between-functional-components-and-class-components-in-react/>
- [18] JavaScript Array map(), w3schools, [Online]. Available: https://www.w3schools.com/jsref/jsref_map.asp

Paper in Conference Proceedings

- [19] Daniel Fernandez-Aviles, Diego Doctor, Daniel Contreras and Jose Carlos Salazar, “Virtual labs: A new tool in the education: Experience of Technical University of Madrid”, 2016
- [20] Krishnashree Achuthan, K. S. Sreelatha, Shone Surendran, Shyam Diwakar, Prema Nedungadi, Steven Humphreys, S. Sreekala C. O., Zeena Pillai, Raghu Raman, Ani Deepthi, Rathish Gangadharan, Saritha Appukuttan, Jyothi Ranganatha, Sreedha Sambhudevan and Suma Mahesh, “The VALUE @ Amrita Virtual Labs Project: Using Web Technology to Provide Virtual Laboratory Access to Students”, 2011
- [21] Lixin Tao, Li-Chiou Chen and Chienting Lin, “Improving web security education with virtual labs and shared course modules”, 2010
- [22] A. Etxebarria, I.J. Oleagordia, and M. Sanchez, “Power electronics and basic electronics real experiments through the World Wide Web”, 2001
- [23] Debabrata Bagchi, Kapil Kaushik and Bhanu Kapoor, “Virtual labs for electronics engineering using cloud computing”, 2013
- [24] Mohamed Darwish, Hassan Amreiz, Alan Janbey and Christos C. Marouchos, “Virtual Power Electronics Labs for Online Teaching”, 2020

Βιβλία

- [25] Albert Malvino, David J. Bates, *ΗΛΕΚΤΡΟΝΙΚΗ – ΑΡΧΕΣ ΚΑΙ ΕΦΑΡΜΟΓΕΣ*, 8^η Έκδοση, ΕΚΔΟΣΕΙΣ ΤΖΙΟΛΑ, 2019
- [26] Παναγιώτα Αθ. Παπαβραμίδου, Δημήτρης Κλ. Παπακώστας, *ΗΛΕΚΤΡΟΝΙΚΕΣ ΕΞΕΡΕΥΝΗΣΕΙΣ*, ΕΚΔΟΣΕΙΣ ΤΖΙΟΛΑ, 2018
- [27] Charles K. Alexander, Matthew N.O. Sadiku, *Εισαγωγή στα Ηλεκτρικά Κυκλώματα*, 4^η έκδοση, ΕΚΔΟΣΕΙΣ ΤΖΙΟΛΑ, 2019
- [28] Robert W. Maksimovic, Dragan Erickson, *Fundamentals of Power Electronics*, 3rd Edition, Springer Nature Switzerland AG, 2020

ΠΑΡΑΡΤΗΜΑ Α : ΚΩΔΙΚΑΣ ΕΦΑΡΜΟΓΗΣ

```

import React, { useState } from 'react'
import Main from "../main/main"

function App() {
  const [count, setCount] = useState(0);
  const [showMain, setShowMain] = React.useState(false)

  const [colorBJT, setColorBJT] = useState('black')
  const [colorOP, setColorOP] = useState('black')

  return (
    <div>
      <h3 style = {{ textAlign: "center", fontWeight: "bold"}}>Διαδίκτυακή εφαρμογή
ανάλυσης λειτουργίας βασικών ηλεκτρονικών κυκλωμάτων και διατάξεων</h3>
      <h5 style = {{ textAlign: "center", marginTop: "10px"}}>Επιλέξτε κατηγορία:
      <br></h5>

      <button className={colorBJT} onClick={() => [setCount(0), setShowMain(true),
setColorOP(() => colorOP == 'black'), setColorBJT(() => colorBJT === 'black' ? 'blue'
: 'blue')]}>
        Διπολικά τρανζίστορ </button>

      <button className={colorOP} onClick={() => [setCount(1), setShowMain(true),
setColorBJT(() => colorBJT == 'black'), setColorOP(() => colorOP === 'black' ? 'blue'
: 'blue')]}>
        Τελεστικοί ενισχυτές </button>

    </div>

    {showMain? <Main opt={count} /> : null }

    <footer>
      <p style = {{ textAlign: "center", fontSize: "20px" }}><u>Ανάπτυξη
εφαρμογής:</u> Χαραλαμπίδης Άγγελος [<a href = "mailto:
agso2006@gmail.com">agso2006@gmail.com</a>] </p>
      <p style = {{ textAlign: "center", fontSize: "20px", marginLeft: "257px"
}}><u>Επιβλέπων καθηγητής:</u> Παπακώστας Δημήτριος [<a href =
"https://www.iee.ihu.gr/staff/papakostas-dimitrios/">https://www.iee.ihu.gr/staff/papakostas-dimitrios/</a>]</p>
    </footer>
  </div>
)
}

export default App

```

Κώδικας 1: Ο κώδικας του App.js

```

import React, { useState, useEffect } from 'react'
import { Container, Row, Col } from 'react-bootstrap'
import Bjt from '../circuits/bjt'
import VDB from '../circuits/VDB'
import Emitter from '../circuits/Emitter'
import Inverting_Amplifier from '../circuits/Inverting_Amplifier/index'
import Non_Inverting_Amplifier from '../circuits/Non-Inverting_Amplifier/index'
import Summing_Amplifier from '../circuits/Summing_Amplifier'
import Difference_Amplifier from '../circuits/Difference_Amplifier'
import ComponentFromArray from './componentfromarray'

export default function Main (props) {

  const {opt} = props

  var components
  if (opt === 0) components = [VDB, Bjt, Emitter]
  else components = [Inverting_Amplifier, Non_Inverting_Amplifier, Summing_Amplifier,
Difference_Amplifier]

  const [key, setKey] = useState(0)

  useEffect(() => setKey($ => $ >= components.length ? 0 : $), [key, setKey])

  return (
    <Container className='App mb-4' style={{ backgroundColor: 'lightblue' }}>
      <div style={{ position: 'static', textAlign: 'center', fontSize: '22px'
}}>Επιλέξτε κύκλωμα:</div>
      <div className='d-flex mb-2' style={{ overflowX: 'scroll' }}>
        {components.map(($, idx) => (
          <div style={{ display: 'inline-block', marginLeft: '1rem' }} key=${$.image}>
            <img
              style={{
                maxHeight: '10rem',
                padding: '1rem',
                borderStyle: 'outset', borderWidth: '7px', borderBottom: '5px
solid grey',
                marginBottom: '15px', paddingBottom: '15px',
                borderBottomColor: 'black', borderLeftColor: 'black',
borderRightColor: 'black', borderTopColor: 'black',
                borderColor: ((idx === key) ? 'blue' : 'black')
              }}
              src=${$.image}
              alt=''
              onClick={() => setKey(idx)}
            />
          </div>
        ))}
      </div>
      <Row>
        <Col xs={12}>
          <ComponentFromArray array={components} idx={key} />
        </Col>
      </Row>
    </Container>
  )
}

```

Κώδικας 2: Ο κώδικας του main.jsx

```

export default function ComponentFromArray (props) {
  const Comp = props.array[props.idx]
  const First = props.array[0]
  return (
    <>
      {
        props.idx >= props.array.length
        ? <First />
        : <Comp />
      }
    </>
  )
}

```

Κώδικας 3: Ο κώδικας του componentfromarray.jsx

```

function calculateEqual (values) {
  const Vcc = parseFloat(values.Vcc)
  const R1 = parseFloat(values.R1)
  const R2 = parseFloat(values.R2)
  const Re = parseFloat(values.Re)
  const Rc = parseFloat(values.Rc)
  const h = parseFloat(values.β)

  /* eslint-disable prefer-const */

  let a = (h / (h + 1))
  let R1R2 = ((R1 * R2) / (R1 + R2)).toFixed()
  let Vbb = (R2 / (R1 + R2)) * Vcc
  let Ie = (Vbb - 0.7) / (Re + (R1R2 / h))
  let Ib = Ie / h
  let Ic = (a * Ie)
  let Vb = Vbb - (Ib * R1R2)
  let Ve = Vb - 0.7
  let Vc = Vcc - (Ic * Rc)
  let Vce = Vc - Ve

  /* eslint-enable */
  const outcomes = { Vbb, Vce, Vc, Vb, Ve, Ic: Ic * 1000, Ib: Ib * 1000, Ie: Ie *
1000, R1R2 }
  return outcomes
}

```

Κώδικας 4: Η συνάρτηση calculateEqual () του VDB

```

<Row>
  <Col xs={6}>
    <div style={{ textAlign: 'center' }}>Κύκλωμα πόλωσης διαιρέτη
τάσης</div>
    <div>
      <VccValueSelector x = {false} values={values} setValues={setValues}
name='Vcc' measurement='V' options=[ 5, 10, 12, 15, 18, 20, 24 ] />
      <CircuitValueInput values={values} setValues={setValues} name='R1'
measurement='Ω' measurements={measurementsR} min={10} max={1_000000} />
      <CircuitValueInput values={values} setValues={setValues} name='R2'
measurement='Ω' measurements={measurementsR} min={10} max={1_000000} />
      <CircuitValueInput values={values} setValues={setValues} name='Re'
measurement='Ω' measurements={measurementsR} min={10} max={1_000000} />
      <CircuitValueInput values={values} setValues={setValues} name='Rc'
measurement='Ω' measurements={measurementsR} min={10} max={1_000000} />
      <CircuitValueInput values={values} setValues={setValues} name='β'
measurement='' min={50} max={300} allowLower />
    </div>
    <div>
      <div style={{ textAlign: 'center'}}>Υπολογισμοί</div>
      <CircuitValueInput values={outcomes} name='Vc' measurement='V'
disabled fixed />
      <CircuitValueInput values={outcomes} name='Vb' measurement='V'
disabled fixed />
      <CircuitValueInput values={outcomes} name='Ve' measurement='V'
disabled fixed />
      <CircuitValueInput values={outcomes} name='Ic' measurements=
{measurementsIb} disabled fixed />
      <CircuitValueInput values={outcomes} name='Ib' measurements=
{measurementsIb} disabled fixed />
      <CircuitValueInput values={outcomes} name='Ie' measurements=
{measurementsIb} disabled fixed />
      <CircuitValueInput values={outcomes} name='bjtMode' label='Περιοχή
λειτουργίας' measurement='' allowLower disabled shouldBlink={outcomes['bjtMode'] !==
BJT_MODES.AMP} />
    </div>
  </Col>
  <Col xs={6}>
    <div style={{ textAlign: 'center' }}>Ισοδύναμο Κύκλωμα {'(Με Rth)'}
</div>
    <VccValueSelector values={values} setValues={setValues} name='Vcc'
measurement='V' options=[ 5, 10, 12, 15, 18, 20, 24 ] disabled />
    <CircuitValueInput values={equalOutcomes} name='Vbb' measurement='V'
disabled fixed />
    <CircuitValueInput values={equalOutcomes} name='R1R2' label='R1//R2'
measurements={measurementsR} disabled />
    <CircuitValueInput values={values} setValues={setValues} name='Re'
measurements={measurementsR} min={10} max={1_000000} disabled />
    <CircuitValueInput values={values} setValues={setValues} name='Rc'
measurements={measurementsR} min={10} max={1_000000} disabled />
    <CircuitValueInput values={values} setValues={setValues} name='β'
measurement='' min={50} max={300} allowLower disabled />
    <div style={{ textAlign: 'center'}}>Υπολογισμοί</div>
    <CircuitValueInput values={equalOutcomes} name='Vc' measurement='V'
disabled fixed />
    <CircuitValueInput values={equalOutcomes} name='Vb' measurement='V'
disabled fixed />
    <CircuitValueInput values={equalOutcomes} name='Ve' measurement='V'
disabled fixed />
    <CircuitValueInput values={equalOutcomes} name='Ic' measurements=
{measurementsIb} disabled fixed />
    <CircuitValueInput values={equalOutcomes} name='Ib' measurements=
{measurementsIb} disabled fixed />
    <CircuitValueInput values={equalOutcomes} name='Ie' measurements=
{measurementsIb} disabled fixed />
  </Col>
</Row>

```

Κώδικας 5: Το Row μέσα στο οποίο γίνεται ο σχηματισμός των πεδίων στο κύκλωμα πόλωσης διαιρέτη τάσης

```

const measurementsIb = [
  {
    symbol: 'mA',
    restore: v => v,
    render: v => v
  },
  {
    symbol: 'μA',
    restore: v => parseFloat(v) / 1000,
    render: v => parseFloat(v) * 1000
  }
]

const measurementsR = [
  {
    symbol: 'kΩ',
    restore: v => parseFloat(v) * 1000,
    render: v => parseFloat(v) / 1000
  },
  {
    symbol: 'Ω',
    restore: v => v,
    render: v => v
  },
  {
    symbol: 'MΩ',
    restore: v => parseFloat(v) * 1000000,
    render: v => parseFloat(v) / 1000000
  }
]

```

Κώδικας 6: Οι πίνακες με τα στοιχεία για τη μετατροπή ρευμάτων και αντιστάσεων

```

{
  (outcomes.Vc < outcomes.Vb || outcomes.Vb < outcomes.Ve || outcomes.Vc
< outcomes.Ve) &&
  (<Alert className='text-center' variant='danger'><div style={{
fontSize: '160%' }}><b>Δεν ισχύει Vc {'>'} Vb {'>'} Ve... Κύκλωμα εκτός ενεργού
περιοχής... Παρακαλώ ξαναεισαγάγετε τιμές</b></div></Alert>)
}

{
  (outcomes.Vc < 0 || outcomes.Vb < 0 || outcomes.Ve < 0 || outcomes.Ic <
0 || outcomes.Ib < 0 || outcomes.Ie < 0) &&
  (<Alert className='text-center' variant='danger'><div style={{
fontSize: '160%' }}><b>Αρνητική τάση ή ρεύμα</b></div></Alert>)
}

```

Κώδικας 7: Ο κώδικας για την εμφάνιση των προειδοποιητικών μηνυμάτων στο κύκλωμα πόλωσης διαιρέτη τάσης

```

{
  (outcomes.Vc < outcomes.Vb || outcomes.Vb < outcomes.Ve || outcomes.Vc <
outcomes.Ve) &&
  (outcomes.bjtMode = BJT_MODES.OUT)
}

{
  (outcomes.Vc < 0 || outcomes.Vb < 0 || outcomes.Ve < 0 || outcomes.Ic < 0 ||
outcomes.Ib < 0 || outcomes.Ie < 0) &&
  (outcomes.bjtMode = BJT_MODES.OUT)
}

```

Κώδικας 8: Ο κώδικας για την αλλαγή σε λάθος λειτουργία

```

<Col>
{
(outcomes.bjtMode == BJT_MODES.AMP) &&
(
(((equalOutcomes.R1R2) < (0.01 * (values.β) * (values.Re))) &&
(<Alert className='text-center' variant='success'><div style={{
fontSize: '160%' }}><b>πολύ σταθερός διαιρέτης τάσης</b></div></Alert>)) ||

(((equalOutcomes.R1R2) < (0.1 * (values.β) * (values.Re))) &&
(<Alert className='text-center' variant='info'><div style={{
fontSize: '130%' }}><b>σταθερός διαιρέτης τάσης</b></div></Alert>))
)
}
</Col>

```

Κώδικας 9: Κώδικας για την εμφάνιση μηνυμάτων σταθερότητας του διαιρέτη τάσης

```

let bjtMode = BJT_MODES.AMP

let Ib = (Vbb - 0.7) / Rb
if ((Ib <= 0) && (Vbb < 0.7)) {
  Ib = 0
  bjtMode = BJT_MODES.CUT
}

let Ic = 0
let Vc = Vcc
let Vb = Vbb - (Ib * Rb)

if (bjtMode === BJT_MODES.AMP) {
  Ic = Ib * h
  Vc = Vcc - (Ic * Rc)

  if (Vc <= Vb) {
    Vc = 0.3
    Vb = 0.7
    Ic = (Vcc - Vc) / Rc
    bjtMode = BJT_MODES.SAT
  }
}

let Ie = (Ib + Ic)

```

Κώδικας 10: Οι υπολογισμοί για το κύκλωμα πόλωσης βάσης

```

let bjtMode = BJT_MODES.AMP

let Ic = 0
let Ie = 0
let Ib = 0
let Ve = 0
let a = 0

if (((Vbb - 0.7) < 0) ){
    Ic = 0
    Ie = 0
    Ib = 0
    Ve = 0
    bjtMode = BJT_MODES.CUT
}

let Vb = Vbb
let Vc = Vcc

if (bjtMode === BJT_MODES.AMP){
    a = (h / (h + 1))
    Ve = (Vbb - 0.7)
    Ie = (Ve/Re)
    Ic = (a * Ie)
    Ib = Ic / h
    Vc = Vcc - (Ic * Rc)

    if (Vc <= Vbb){
        Ve = (Vbb - 0.7)
        Ie = (Ve/Re)
        Vc = Ve + 0.3
        Ic = (Vcc - Vc) / Rc
        Ib = (Ie - Ic)
        bjtMode = BJT_MODES.SAT
    }
}

```

Κώδικας 11: Οι υπολογισμοί για το κύκλωμα πόλωσης εκπομπού

```

<Row>
  <Col>
    <VccValueSelector x = {true} values={values} setValues={setValues}
name='Vcc' measurement='V' options=[[ 10, 12, 15, 20, 24, 30 ]] />
    <div style={{ textAlign: 'center' }}> Διαφορά κορεσμού εξόδου από την
τάση τροφοδοσίας (V) </div>
    <VccValueSelector values={values} setValues={setValues} name='field'
options=[[ 0, 0.5, 1, 2]] />
    <CircuitValueInput values={values} setValues={setValues} name='Vin'
measurement='V' measurements={measurementsV} min={0.001} max={10} />
    <CircuitValueInput values={values} setValues={setValues} name='R1'
measurement='Ω' measurements={measurementsR} min={100} max={1_000000} />
    <CircuitValueInput values={values} setValues={setValues} name='Rf'
measurement='Ω' measurements={measurementsR} min={100} max={1_000000} />
    <VccValueSelector values={values} setValues={setValues} name='f'
measurement='kHz' measurements={measurementsF} options=[[ 1, 5, 10]] />
    {opt === 1 ? <VccValueSelector values={values} setValues={setValues}
name='sr' measurement='V/μs' options=[[ 0.5, 1, 2]] /> : <></>}
  </Col>
  <Col>
    <CircuitValueInput values={outcomes} name='Vout' measurements=
{measurementsV} disabled fixed shouldBlink = {!bool} />
    <CircuitValueInput values={outcomes} name='Acl' measurement='' disabled
/>
    <CircuitValueInput values={outcomes} name='Fcl' measurements=
{measurementsF} disabled />
    <div style={{ textAlign: 'center' }}> Περιορισμένη τάση εξόδου </div>
    <CircuitValueInput values={outcomes} name="diff" measurements=
{measurementsV} measurement='V' disabled />
  </Col>
</Row>

```

Κώδικας 12: Ο κώδικας για τη δημιουργία των πεδίων του αναστρέφοντος ενισχυτή

```

const [values, setValues] = useState({
  Vin: 0.5,
  Vcc: 12,
  Vee: -12,
  R1: 1_500,
  Rf: 7_500,
  f: 1,
  sr: 0.5,
  field: 0
})

const [outcomes, setOutcomes] = useState(INITIAL_OUTCOMES)

useEffect(() => {
  const Vin = parseFloat(values.Vin)
  const R1 = parseFloat(values.R1)
  const Rf = parseFloat(values.Rf)
  const f = parseFloat(values.f)
  /* eslint-disable prefer-const */

  let Acl = - (Rf / R1)
  let Vout = Acl * Vin
  let Fcl = (- (1000000 / Acl)).toFixed()
  let Av1 = 0
  let Av2 = 0

  if (f < Fcl){
    Vout = Acl * Vin
  }

  if (f === Fcl){
    Av1 = (0.707 * Acl)
    Vout = Av1 * Vin
  }

  if (f > Fcl){
    Av2 = (Acl / Math.sqrt(1 + Math.pow((f / Fcl), 2)))
    Vout = Av2 * Vin
  }

  const diff = parseFloat(values.Vcc) - parseFloat(values.field)

  /* eslint-enable */

  const outcomes = { Vout, Acl, Fcl, diff }
  setOutcomes({...outcomes })
}, [values])

```

Κώδικας 13: Υπολογισμοί στον αναστρέφων ενισχυτή

```

import { Form, InputGroup, OverlayTrigger, Tooltip } from 'react-bootstrap'

export default function VccValueSelector(props) {
  const { x, values, setValues, name, measurement, options, disabled } = props

  if (measurement === undefined) return(
    <>
      <InputGroup className='mb-3'>
        <Form.Control
          as="select"
          value={values[name]}
          title={values[name]}
          onChange={event => setValues({ ...values, [name]: event.target.value })}
          disabled={disabled}
        >
          {options.map((voltage) => (
            <option key={voltage} value={voltage}>{voltage}</option>
          ))}
        </Form.Control>
      </InputGroup>
    </>
  )

  else
    if (x && name==='Vcc') return (
      <>
        <OverlayTrigger placement='left'
          overlay={ <Tooltip> Η θετική τάση τροφοδοσίας Vcc είναι ίση με την αρνητική
Vee </Tooltip> }>
          <InputGroup className='mb-3'>
            <InputGroup.Prepend>
              <InputGroup.Text style={{width: '5rem', userSelect: 'none'}}>
                <span style={{ fontVariantCaps: 'small-caps' }}>{name}</span>
                <span style={{ marginLeft: 'auto' }}>{'(' + measurement + ')'}</span>
              </InputGroup.Text>
            </InputGroup.Prepend>
            <Form.Control as="select" value={values[name]} title={values[name]}
              onChange={event => setValues({ ...values, [name]: event.target.value })} disabled=
{disabled}>
              {options.map((voltage) => ( <option key={voltage} value={voltage}>
{voltage}</option> ))}
            </Form.Control>
          </InputGroup>
        </OverlayTrigger>
      </>
    )
    else return (
      <>
        <InputGroup className='mb-3'>
          <InputGroup.Prepend>
            <InputGroup.Text style={{width: '5rem', userSelect: 'none'}}>
              <span style={{ fontVariantCaps: 'small-caps' }}>{name}</span>
              <span style={{ marginLeft: 'auto' }}>{'(' + measurement + ')'}</span>
            </InputGroup.Text>
          </InputGroup.Prepend>
          <Form.Control as="select" value={values[name]} title={values[name]}
            onChange={event => setValues({ ...values, [name]: event.target.value })} disabled=
{disabled}>
            {options.map((voltage) => ( <option key={voltage} value={voltage}>
{voltage}</option> ))}
          </Form.Control>
        </InputGroup>
      </>
    )
  }
}

```

Κώδικας 14: Το component VccValueSelector

```

import React, { useEffect, useState } from 'react'
import './styles.scss'
import Chartist from 'chartist'
import ChartistGraph from 'react-chartist'
import 'chartist-plugin-tooltips-updated/dist/chartist-plugin-tooltip.css'
import {Row} from 'react-bootstrap'
import Legend from "chartist-plugin-legend";

export default function SineWave(props, ) {
  const { frequency, voltages, value } = props

  const [plotData, setPlotData] = useState(null)
  const [plotOptions, setPlotOptions] = useState(null)
  const height = '23rem'

  const SAMPLES_AMOUNT = 50
  const TIME_SCALE = 1000
  var done = false

  useEffect(() => {
    const f = frequency
    const T = 1 / f

    const xAxisPoints = [0, T/4, T/2, 3*T/4, T]
      .map(p => p * TIME_SCALE)
      .map(p => p.toFixed(0))

    setPlotData({
      series: voltages.map(V => [...Array(SAMPLES_AMOUNT)].map( (_, t) => {
        const time = t * (T / (SAMPLES_AMOUNT - 1))
        var y = V * Math.sin(2 * Math.PI * f * time)

        if ((y > value) && (V == voltages[1])) {
          y = value;
          done = true
        }
        else if ((y < -value) && (V == voltages[1])){
          y=-value;
          done = true
        }
        return {
          x: time * TIME_SCALE,
          y: y
        }
      }
    )))
  })

  var refVoltage = Math.max(...voltages.map(Math.abs))
  if (refVoltage > value && done) refVoltage = value

  setPlotOptions({
    plugins: [
      Legend({
        legendNames: ['Είσοδος', 'Εξοδος'],
        clickable: false
      })
    ],
    high: 1.01 * refVoltage,
    low: 1.01 * -refVoltage,
    showPoint: false,
    axisX: {
      type: Chartist.FixedScaleAxis,
      divisor: xAxisPoints.length,
      ticks: xAxisPoints,
      stretch: true,
      labelInterpolationFnc: p => p + 'μs'
    },
    axisY: {
      labelInterpolationFnc: p => p + 'V'
    }
  })
}, [frequency, voltages])

return <>
  { plotData && <>
    <Row>
      <chartistGraph style={{ height }} className='ct-octave' data={plotData}
options={plotOptions} type='Line' />
    </Row>
  </> }
</>
}

```

Κώδικας 15: Ο κώδικας για το SineWave

```

import {LineChart, ReferenceLine, ResponsiveContainer, Line, XAxis, YAxis,
CartesianGrid} from 'recharts';
import {Row} from 'react-bootstrap'
import './styles.scss'

export default function Pulse(props) {
const {sr, frequency, voltages} = props
const TIME_SCALE = 1000
const T = 1/frequency
const t = TIME_SCALE*T

const plotData =
[
{time: 0, in: voltages[0], out: 0},
{time: Math.abs(voltages[1]/sr), in: voltages[0], out: voltages[1]},
{time: t/4, in: voltages[0], out: voltages[1]},
{time: t/2, in: voltages[0], in: voltages[0], in2: 0, out: voltages[1]},
{time: (t/2) + Math.abs(voltages[1]/sr), in2: 0, out: 0},
{time: 3*t/4, in2: 0, out: 0},
{time: t, in2: 0, out: 0}];

return (
<>
<Row> <div> <div class='box blue'></div>Είσοδος </div> <div> <div class='box
red'></div>Εξοδος </div> </Row>
<ResponsiveContainer width="310%" aspect={2}>
<LineChart data={plotData} margin={{ right: 300 }}>
<CartesianGrid vertical={false} stroke='grey' strokeDasharray="1 1" />

<XAxis dataKey="time" allowDuplicatedCategory={false} tickLine=
{false} type='category' tickFormatter={n => {if (n === t/4 || n === 3*t/4) return '';
else return n.toString() + 'μs'}} />
<YAxis tickLine={false} tickFormatter={n => {return n.toString() +
'V'}}/>

<Line isAnimationActive={false} strokeWidth={4} dataKey="out"
stroke="#d70202" dot={false} />

<ReferenceLine isAnimationActive={false} strokeWidth={3}
stroke="#4fc2f0" dot={false} segment={{ x: 0, y: 0 }, { x: 0, y: voltages[0] }} />
<Line isAnimationActive={false} strokeWidth={3} dataKey="in"
stroke="#4fc2f0" dot={false} />
<ReferenceLine isAnimationActive={false} strokeWidth={3}
stroke="#4fc2f0" alwaysShow={true} dot={false} segment={{ x: t/2, y: voltages[0] }, {
x: t/2, y: 0 }} />
<Line isAnimationActive={false} strokeWidth={3} dataKey="in2"
stroke="#4fc2f0" dot={false} />

<ReferenceLine stroke="grey" strokeDasharray="1 1" dot={false} x=
{plotData[1].time} />
<ReferenceLine stroke="grey" strokeDasharray="1 1" dot={false} x=
{plotData[3].time} />
<ReferenceLine stroke="grey" strokeDasharray="1 1" dot={false} x=
{plotData[4].time} />
<ReferenceLine stroke="grey" strokeDasharray="1 1" dot={false} x=
{plotData[6].time} />
</LineChart>
</ResponsiveContainer>
</>
);
}

```

Κώδικας 16: Ο κώδικας για τη δημιουργία των τετραγωνικών παλμών

ΠΑΡΑΡΤΗΜΑ Β : ΕΡΩΤΗΜΑΤΟΛΟΓΙΟ

Ερωτηματολόγιο για την αξιολόγηση της διαδικτυακής εφαρμογής ανάλυσης λειτουργίας βασικών ηλεκτρονικών κυκλωμάτων και διατάξεων

Το παρόν ερωτηματολόγιο δημιουργήθηκε για την αξιολόγηση της εν λόγω εφαρμογής στα πλαίσια διπλωματικής εργασίας

agso2006@gmail.com [Εναλλαγή λογαριασμού](#)



* Απαιτείται

Διεύθυνση ηλεκτρονικού ταχυδρομείου *

Η διεύθυνσή σας ηλεκτρονικού ταχυδρομεί...

Έχετε περάσει τα μαθήματα ηλεκτρονικά κυκλώματα και ηλεκτρονικές διατάξεις; *

- Μόνο ηλεκτρονικά κυκλώματα
- Μόνο ηλεκτρονικές διατάξεις
- Και τα δύο
- Κανένα από τα δύο

Σε τι βαθμό σας άρεσε η σχεδίαση της εφαρμογής; *

- Καθόλου
- Λίγο
- Μέτρια
- Πολύ
- Πάρα πολύ

Θεωρείτε ότι η εφαρμογή είναι εύχρηστη; *

- Καθόλου
- Λίγο
- Μέτρια
- Πολύ
- Πάρα πολύ

Θεωρείτε ότι η εφαρμογή είναι λειτουργική; *

- Καθόλου
- Λίγο
- Μέτρια
- Πολύ
- Πάρα πολύ

Πιστεύετε ότι η εφαρμογή μπορεί να αποτελέσει συμπληρωματικό βοήθημα στα παραπάνω μαθήματα; *

- Καθόλου
- Λίγο
- Μέτρια
- Πολύ
- Πάρα πολύ

Σε τι βαθμό πιστεύετε ότι η εφαρμογή μπορεί να βοηθήσει στην καλύτερη κατανόηση των εννοιών των παραπάνω μαθημάτων; *

- Καθόλου
- Λίγο
- Μέτρια
- Πολύ
- Πάρα πολύ

Σε τι βαθμό σκοπεύετε να την χρησιμοποιήσετε για τη μελέτη των παραπάνω μαθημάτων; *

- Καθόλου
- Λίγο
- Μέτρια
- Πολύ
- Πάρα πολύ

Πιστεύετε ότι θα έπρεπε κάποιος να έχει προηγούμενες γνώσεις για να χρησιμοποιήσει την εφαρμογή;

*

- Καθόλου
- Λίγο
- Μέτρια
- Πολύ
- Πάρα πολύ

Θα συστήνατε την εφαρμογή σε κάποιον που ασχολείται με τα παραπάνω μαθήματα ή με περιεχόμενο που περιλαμβάνεται σε αυτήν;

*

- Ναι
- Όχι
- Ίσως

