



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΣΧΕΔΙΑΣΜΟΣ, ΥΛΟΠΟΙΗΣΗ ΚΑΙ
ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΔΙΑΔΙΚΤΥΑΚΟΥ
ΕΡΓΑΛΕΙΟΥ ΧΩΡΟΘΕΤΗΣΗΣ
ΟΛΟΚΛΗΡΩΜΕΝΩΝ ΚΥΚΛΩΜΑΤΩΝ

ΔΕΜΕΣΙΩΤΗΣ ΓΕΩΡΓΙΟΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

ΔΑΔΑΛΙΑΡΗΣ ΑΝΤΩΝΙΟΣ
Επικουρος Καθηγητής

Λαμία 2023



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΣΧΕΔΙΑΣΜΟΣ, ΥΛΟΠΟΙΗΣΗ ΚΑΙ
ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΔΙΑΔΙΚΤΥΑΚΟΥ
ΕΡΓΑΛΕΙΟΥ ΧΩΡΟΘΕΤΗΣΗΣ
ΟΛΟΚΛΗΡΩΜΕΝΩΝ ΚΥΚΛΩΜΑΤΩΝ

ΔΕΜΕΣΙΩΤΗΣ ΓΕΩΡΓΙΟΣ



UNIVERSITY OF
THESSALY

SCHOOL OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE & TELECOMMUNICATIONS

IMPLEMENTATION & OPTIMIZATION OF
THE BACK END
OF A WEB BASED IC GLOBAL PLACEMENT
AND LEGALIZATION ANALYSIS TOOL

DEMESIOTIS GEORGIOS

FINAL THESIS

ADVISOR

Antonios Dadaliaris
Assistant Professor.

Lamia 2023

«Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις ⁽¹⁾, που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί **χωρίς να τα περικλείω σε εισαγωγικά** και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάστηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.
2. Δέχομαι ότι η αυτολεξεί **παράθεση χωρίς εισαγωγικά**, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.
3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια
4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

Ημερομηνία: 24/07/23

Ο - Η Δηλ.

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.»

ΠΕΡΙΛΗΨΗ

Αυτή η πτυχιακή παρουσιάζει μια διαδικτυακή εφαρμογή για την οπτικοποίηση της τοποθέτησης στην φυσική σχεδίαση ολοκληρωμένων κυκλωμάτων. Αναλύει τα ενσωματωμένα κυκλώματα, τα στοιχεία τους και τις διαδικασίες παραγωγής. Παρουσιάζει τις τεχνολογίες και τις γλώσσες προγραμματισμού που χρησιμοποιούνται στο WEVIAN και την προσθήκη του, το WeVisAP, για την κατανόηση των δυνατοτήτων τους.

Μια αξιολόγηση του WEVIAN και του WeVisAP αναδεικνύει τις μοναδικές τεχνολογικές τους πτυχές, τα πλεονεκτήματα και τα περιορισμένα σημεία τους στην οπτικοποίηση της τοποθέτησης κατά τα στάδια της φυσικής σχεδίασης των ολοκληρωμένων κυκλωμάτων.

Η πτυχιακή εμβαθύνει την κατανόηση της οπτικοποίησης της τοποθέτησης. Επιπροσθέτως, παρουσιάζεται το ιστορικό υπόβαθρο των ολοκληρωμένων κυκλωμάτων, δείχνοντας την εξέλιξη τους και τη σημασία τους στα σύγχρονα ηλεκτρονικά συστήματα.

ABSTRACT

This thesis presents a web-based application for viewing the Placement in the physical design of integrated circuits. It examines integrated circuits, their components, and the manufacturing methods. It demonstrates the technologies and programming languages used in WEVIAN and its extension, WeVisAP, to help users grasp their capabilities.

An evaluation of WEVIAN and WeVisAP illustrates their distinct technological elements, advantages, and limits in visualizing the placement of integrated circuits during the physical design stages.

The thesis delves into a deeper understanding of the visualization of placement. It also covers the history of integrated circuits, demonstrating their progress and significance in current electronic systems.

Table of Contents

ΠΕΡΙΛΗΨΗ.....	I
ABSTRACT.....	III
ΚΕΦΑΛΑΙΟ 1 ΕΙΣΑΓΩΓΗ.....	6
(1.1 ΙΣΤΟΡΙΚΟ ΥΠΟΒΑΘΡΟ ΤΩΝ ΟΛΟΚΛΗΡΩΜΕΝΩΝ ΚΥΚΛΩΜΑΤΩΝ).....	6
(1.2 ΣΗΜΑΣΙΑ ΤΩΝ ΟΛΟΚΛΗΡΩΜΕΝΩΝ ΚΥΚΛΩΜΑΤΩΝ ΣΤΗ ΣΥΓΧΡΟΝΗ ΤΕΧΝΟΛΟΓΙΑ).....	7
ΚΕΦΑΛΑΙΟ 2 ΦΥΣΙΚΗ ΣΧΕΔΙΑΣΗ : ΜΙΑ ΕΠΙΣΚΟΠΗΣΗ.....	8
(2.1 ΡΟΛΟΣ ΤΗΣ ΦΥΣΙΚΗΣ ΣΧΕΔΙΑΣΗΣ ΣΤΗΝ ΣΥΝΟΛΙΚΗ ΔΙΑΔΙΚΑΣΙΑ ΣΧΕΔΙΑΣΜΟΥ IC).....	8
(2.2 ΤΑ ΚΥΡΙΑ ΒΗΜΑΤΑ ΣΤΟ ΣΤΑΔΙΟ ΤΗΣ ΦΥΣΙΚΗΣ ΣΧΕΔΙΑΣΗΣ).....	9
(2.2.1 DESIGN NETLIST (AFTER SYNTHESIS)).....	12
(2.2.2 PARTITIONING).....	13
(2.2.3 FLOORPLANNING).....	15
(2.2.4 PLACEMENT).....	17
(2.2.5 CLOCK-TREE SYNTHESIS – CTS)).....	18
(2.2.6 ROUTING).....	19
(2.2.7 PHYSICAL VERIFICATION).....	20
(2.2.8 LAYOUT POST PROCESSING ME MASK DATA GENERATION).....	21
ΚΕΦΑΛΑΙΟ 3.....	23
ΣΤΟΧΟΙ ΚΑΙ ΠΡΟΚΛΗΣΕΙΣ ΣΤΗΝ ΦΥΣΙΚΗ ΣΧΕΔΙΑΣΗ.....	23
(3.1 ΣΤΟΧΟΙ ΤΗΣ ΦΥΣΙΚΗΣ ΣΧΕΔΙΑΣΗΣ).....	23
(3.2 ΠΡΟΚΛΗΣΕΙΣ ΠΟΥ ΑΝΤΙΜΕΤΩΠΙΖΟΝΤΑΙ ΣΤΗΝ ΦΥΣΙΚΗ ΣΧΕΔΙΑΣΗ).....	25
ΚΕΦΑΛΑΙΟ 4 ISPD.....	27
(4.1 ΕΠΙΣΚΟΠΗΣΗ ΚΑΙ ΣΗΜΑΣΙΑ).....	27
(4.2 ΚΥΡΙΑ ΘΕΜΑΤΑ ΚΑΙ ΠΕΔΙΑ ΈΡΕΥΝΑΣ).....	27
(4.3 ΤΑ ΟΦΕΛΗ ΤΗΣ ΣΥΝΕΙΣΦΟΡΑΣ).....	27
ΚΕΦΑΛΑΙΟ 5 ΔΟΜΗ ΤΟΥ BOOKSHELF FORMAT.....	29
(5.1 ΕΙΣΑΓΩΓΗ).....	29
(5.2 ΟΡΓΑΝΩΣΗ ΑΡΧΕΙΩΝ ΣΤΟ BOOKSHELF FORMAT).....	29
(5.2.1 AUX).....	29
(5.2.2 NODES).....	30
(5.2.3 NETS).....	30
(5.2.4 WTS).....	30
(5.2.5 PL).....	30

(5.2.6 SCL).....	31
------------------	----

ΚΕΦΑΛΑΙΟ 6.....32

ΟΠΤΙΚΟΠΟΙΗΣΗ ΑΛΓΟΡΙΘΜΩΝ ΤΟΠΟΘΕΤΗΣΗΣ ΣΤΗΝ ΦΥΣΙΚΗ ΣΧΕΔΙΑΣΗ VLSI.....32

ΕΙΣΑΓΩΓΗ.....	32
(6.1 GNPLOT).....	32
(6.1.1 ΟΠΤΙΚΟΠΟΙΗΣΗ ΓΙΑ ΤΗΝ ΑΞΙΟΛΟΓΗΣΗ ΚΑΙ ΑΝΑΛΥΣΗ ΤΗΣ ΤΟΠΟΘΕΤΗΣΗΣ).....	32
(6.1.2 ΓΡΑΦΙΚΑ ΈΤΟΙΜΑ ΓΙΑ ΔΗΜΟΣΙΕΥΣΗ).....	33

ΚΕΦΑΛΑΙΟ 7 WEVIAN [WEB-BASED PLACEMENT VISUALIZER / ANALYZER].....34

(7.1 ΕΠΙΣΚΟΠΗΣΗ).....	34
(7.1.1 ΚΙΝΗΤΡΟ ΓΙΑ ΤΗΝ ΑΝΑΠΤΥΞΗ ΤΟΥ WEVIAN).....	34
(7.1.2 ΣΤΟΧΟΙ ΤΟΥ WEVIAN).....	34
(7.2 ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ).....	36
(7.2.1 ΤΕΧΝΟΛΟΓΙΕΣ ΚΑΙ ΓΛΩΣΣΕΣ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ).....	36
(7.2.2 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΣΥΣΤΗΜΑΤΟΣ ΚΑΙ ΤΑ ΣΤΟΙΧΕΙΑ ΑΥΤΟΥ).....	37
(7.2.3 BACKEND FUNCTIONALITY [PHP , MYSQL]).....	38
(7.2.4 FRONTEND DESIGN [HTML , CSS , JAVASCRIPT]).....	38
(7.2.5 ΕΝΣΩΜΑΤΩΣΗ ΤΕΧΝΙΚΩΝ ΟΠΤΙΚΟΠΟΙΗΣΗΣ).....	38
(7.2.6 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ ΚΑΙ ΛΕΙΤΟΥΡΓΙΕΣ ΤΟΥ WEVIAN).....	39
(7.3 ΧΡΗΣΗ ΚΑΙ ΡΟΗ ΕΡΓΑΣΙΑΣ).....	42
(7.3.1 ΔΙΑΔΙΚΑΣΙΑ ΕΓΓΡΑΦΗΣ ΚΑΙ ΣΥΝΔΕΣΗΣ ΧΡΗΣΤΗ).....	42
(7.3.2 ΜΕΤΑΦΟΡΤΩΣΗ ΚΑΙ ΑΝΑΛΥΣΗ ΑΡΧΕΙΩΝ ΤΟΠΟΘΕΤΗΣΗΣ).....	43
(7.3.3 ΟΠΤΙΚΟΠΟΙΗΣΗ ΤΟΥ ΣΧΕΔΙΑΣΜΟΥ).....	43
(7.3.4 ΔΙΑΔΡΑΣΤΙΚΕΣ ΕΠΙΛΟΓΕΣ ΚΑΙ ΣΤΟΙΧΕΙΑ ΕΛΕΓΧΟΥ).....	44
(7.3.5 ΔΗΜΙΟΥΡΓΙΑ ΑΝΑΦΟΡΩΝ ΚΑΙ ΣΤΑΤΙΣΤΙΚΩΝ).....	45
(7.3.5.1 CIRCUIT STATISTICS).....	45
(7.3.5.2 PERCENTAGE OF DIFFERENT NODES PIE CHART).....	46
(7.3.5.3 TEMPERATURE GRAPH).....	47
(7.3.5.4 CONGESTION MAP GRAPH).....	47
(7.3.5.5 RUNTIME OF DIFFERENT MODULES BAR CHART).....	48
(7.3.6 ΑΠΟΘΗΚΕΥΣΗ ΚΑΙ ΑΝΑΚΤΗΣΗ ΔΕΔΟΜΕΝΩΝ ΧΡΗΣΤΗ).....	49
(7.3.7 ΣΥΜΠΕΡΑΣΜΑ).....	49

ΚΕΦΑΛΑΙΟ 8 WEVISAP [WEB-BASED VISUALIZED ANIMATED PLACEMENT].....50

(8.1 ΕΠΙΣΚΟΠΗΣΗ).....	50
(8.2 ΚΙΝΗΤΡΟ ΓΙΑ ΤΗΝ ΑΝΑΠΤΥΞΗ ΤΟΥ WEVISAP).....	50
(8.2.1 ΑΝΑΠΤΥΞΗ ΤΟΥ WEVISAP).....	51
(8.2.2 ΑΝΑΠΤΥΞΗ ΤΟΥ BACKEND).....	51
(8.2.2 ΑΝΑΠΤΥΞΗ ΤΟΥ FRONTEND).....	52
(8.3 ΧΡΗΣΗ ΚΑΙ ΡΟΗ ΕΡΓΑΣΙΑΣ).....	54
(8.3.1 ΑΡΧΙΚΗ ΣΕΛΙΔΑ).....	54
(8.3.1.1 VISAP).....	54
(8.3.1.2 ΔΗΜΙΟΥΡΓΙΑ ΟΠΤΙΚΟΠΟΙΗΣΗΣ).....	55
(8.3.1.3.....	56

ΠΡΟΒΟΛΗ ΚΑΙ ΑΛΛΗΛΕΠΙΔΡΑΣΗ ΜΕ ΤΙΣ ΟΠΤΙΚΟΠΟΙΗΣΕΙΣ).....	56
(8.3.1.2 ΕΠΙΛΟΓΗ LIBRARY ΓΙΑ ΠΡΟΗΓΟΥΜΕΝΕΣ ΜΕΤΑΦΟΡΤΩΣΕΙΣ).....	57
(8.4 ΣΤΙΓΜΙΟΤΥΠΑ ΤΕΛΙΚΟΥ ΠΑΡΑΓΟΜΕΝΟΥ).....	58
(8.5 ΜΟΡΦΗ ΣΥΜΠΙΕΣΜΕΝΟΥ ΑΡΧΕΙΟΥ ΓΙΑ ΣΩΣΤΗ ΟΠΤΙΚΟΠΟΙΗΣΗ).....	62
(8.6 ΠΑΡΑΛΛΑΓΕΣ ΤΟΥ ΚΩΔΙΚΑ VisAP ΓΙΑ ΟΠΤΙΚΟΠΟΙΗΣΗ .).....	63
(8.6.1 ΕΠΙΣΚΟΠΗΣΗ).....	63
(8.6.2 ΚΟΙΝΗ ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑ).....	64
(8.6.3 ΒΑΣΙΚΕΣ ΔΙΑΦΟΡΕΣ ΜΕΤΑΞΥ ΤΩΝ ΠΑΡΑΛΛΑΓΩΝ).....	65
<u>ΚΕΦΑΛΑΙΟ 9 ΠΕΙΡΑΜΑΤΙΚΗ ΑΞΙΟΛΟΓΗΣΗ ΚΑΙ ΣΥΜΠΕΡΑΣΜΑΤΑ.....</u>	<u>66</u>
(9.1 ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ).....	66
(9.2 WENIAN).....	67
(9.2.1 ΣΥΓΚΡΙΣΗ ΦΥΛΛΟΜΕΤΡΗΤΩΝ).....	69
(9.3 VisAP).....	70
(9.3.1 ΣΥΓΚΡΙΤΙΚΗ ΑΝΑΛΥΣΗ ΤΩΝ ΤΡΙΩΝ ΠΑΡΑΛΛΑΓΩΝ ΤΟΥ ΚΩΔΙΚΑ).....	72
<u>ΒΙΒΛΙΟΓΡΑΦΙΑ.....</u>	<u>74</u>

Table of Figures

Εικόνα 1: Cell Highlighting / Information Display.....	39
Εικόνα 2: Net Group Highlighting.....	39
Εικόνα 3: Overlap Detection.....	40
Εικόνα 4: Overflow Detection.....	40
Εικόνα 5: Congestion Map 20x20.....	41
Εικόνα 6: Thermal Map.....	41
Εικόνα 7: Congestion Map 30x30.....	41
Εικόνα 8: Sign In/Up.....	42
Εικόνα 9: Wevian Main Screen.....	43
Εικόνα 10: Visualization.....	43
Εικόνα 11: User Options.....	44
Εικόνα 12: Circuit statistics.....	45
Εικόνα 13: Percentage of different nodes pie chart.....	46
Εικόνα 14: Temperature Graph.....	47
Εικόνα 15: Congestion Map Graph.....	47
Εικόνα 16: Runtime of different modules bar chart (ms).....	48
Εικόνα 17: Wevian Main Screen 2.....	49
Εικόνα 18: WeVisAp Main Screen.....	54
Εικόνα 19: VisAp Upload Screen.....	55
Εικόνα 20: VisAp Uploading Screen.....	55
Εικόνα 21: VisAp Proccesing Screen.....	55
Εικόνα 22: VisAp Final Screen Player.....	56
Εικόνα 23: WeVisAP Library -> Gordian.....	57
Εικόνα 24: WeVisAP Library.....	57
Εικόνα 25: WeVisAP Library -> Gordian - > Tetris - IBM01.....	57
Εικόνα 26: WeVisAP Library.....	57
Εικόνα 27: ibm13_NTUplace3_parAbacus_1.....	58
Εικόνα 28: ibm13_NTUplace3_parAbacus_2.....	58
Εικόνα 29: ibm13_NTUplace3_parAbacus_3.....	59
Εικόνα 30: ibm13_NTUplace3_parAbacus_4.....	59
Εικόνα 31: ibm08_mPL6_tetrisPlaceNets_1.....	60
Εικόνα 32: ibm08_mPL6_tetrisPlaceNets_2.....	60
Εικόνα 33: ibm08_mPL6_tetrisPlaceNets_3.....	60
Εικόνα 34: ibm08_mPL6_tetrisPlaceNets_4.....	61
Εικόνα 35: ibm08_mPL6_tetrisPlaceNets_5.....	61
Εικόνα 36: ibm08_mPL6_tetrisPlaceNets_6.....	61
Εικόνα 37: Show Overlap.....	67
Εικόνα 38: Show Overflow.....	67
Εικόνα 39: Thermal Map.....	67
Εικόνα 40: Congestion Map 10*10.....	67
Εικόνα 41: Congestion Map 50*50.....	68
Εικόνα 42: Congestion Map 100*100.....	68
Εικόνα 43: Congestion Map 100*100.....	68
Εικόνα 44: Half-Perimeter.....	68
Εικόνα 45: Visualisation.....	68
Εικόνα 46: Execution Time [Gordian].....	70
Εικόνα 47: Memory Usage [Gordian].....	70

Εικόνα 48: Memory Usage [NTUplace3].....	70
Εικόνα 49: Execution Time [NTUplace3].....	70
Εικόνα 50: Memory Usage [Eplace].....	71
Εικόνα 51: Execution Time [Eplace].....	71
Εικόνα 52: Memory Usage [mPL6].....	71
Εικόνα 53: Execution Time [mPL6].....	71
Εικόνα 54: Execution Time Average of each Benchmark.....	71
Εικόνα 55: Memory Usage Average of each Benchmark.....	72

Index of Tables

Πίνακας 1: Περιεχόμενα csv αρχείου.....	62
Πίνακας 2: Περιεχόμενα title αρχείου.....	62
Πίνακας 3: Περιεχόμενα συμπιεσμένου αρχείου.....	62
Πίνακας 4: ISPD 04 benchmarks.....	66
Πίνακας 5: Χαρακτηριστικά Συστήματος.....	66
Πίνακας 6: Αλγόριθμοι.....	70
Πίνακας 7: Μέσος Όρος Ανάμεσα σε όλα τα Benchmarks και Αλγόριθμους.....	72

ΚΕΦΑΛΑΙΟ 1 Εισαγωγή

Το πεδίο του σχεδιασμού ολοκληρωμένων κυκλωμάτων [Integrated Circuit [IC]] έχει παίξει καθοριστικό ρόλο στην προώθηση της σύγχρονης τεχνολογίας, με τα ολοκληρωμένα κυκλώματα, γνωστά και ως μικροτσιπ, να κατακτούν διάφορους κλάδους και να γίνονται αναπόσπαστο μέρος της καθημερινής μας ζωής. Αυτή η ενότητα έχει ως στόχο να παρέχει ιστορικό υπόβαθρο σχετικά με την ανάπτυξη των ολοκληρωμένων κυκλωμάτων και να αναδείξει τη σημαντική τους επίδραση στη σύγχρονη τεχνολογία

(1.1 Ιστορικό υπόβαθρο των Ολοκληρωμένων Κυκλωμάτων)

Η ιδέα της ενσωμάτωσης πολλαπλών ηλεκτρονικών εξαρτημάτων σε ένα μόνο υπόστρωμα εμφανίστηκε στα τέλη της δεκαετίας του 1940. Η εφεύρεση του τρανζίστορ το 1947 από τους μηχανικούς της Bell Labs William Shockley, John Bardeen και Walter Brattain έφερε επανάσταση στις ηλεκτρονικές συσκευές. Το πρώτο λειτουργικό τρανζίστορ παρουσιάστηκε με επιτυχία στις 23 Δεκεμβρίου 1947, προσφέροντας μια πιο συμπαγή και αξιόπιστη εναλλακτική λύση στους σωλήνες κενού [1].

Το μικρότερο μέγεθος του τρανζίστορ και η χαμηλότερη κατανάλωση ενέργειας άνοιξαν το δρόμο για την ανάπτυξη ολοκληρωμένων κυκλωμάτων. Στα τέλη της δεκαετίας του 1950 και στις αρχές της δεκαετίας του 1960, ο Jean Hoerni εισήγαγε την planar process, η οποία επέτρεψε την κατασκευή τρανζίστορ και άλλων ηλεκτρονικών εξαρτημάτων σε μια επίπεδη επιφάνεια [2]. Αυτή η σημαντική ανακάλυψη, σε συνδυασμό με τις εξελίξεις στα υλικά ημιαγωγών, οδήγησε στην πρακτική υλοποίηση των ολοκληρωμένων κυκλωμάτων.

Ο Jack Kilby της Texas Instruments και ο Robert Noyce της Fairchild Semiconductor εφηύραν ανεξάρτητα τα πρώτα πρακτικά ολοκληρωμένα κυκλώματα, το 1958. Η προσέγγιση του Kilby περιλάμβανε την κατασκευή όλων των εξαρτημάτων και διασυνδέσεων σε ένα μόνο κομμάτι Γερμανίου [Germanium [G 32]], ενώ το σχέδιο του Noyce χρησιμοποιούσε Πυρίτιο [Silicon [Si 14]] ως υλικό υποστρώματος [3]. Αυτά τα πρώτα ολοκληρωμένα κυκλώματα αποτελούνταν από λίγα μόνο τρανζίστορ, αλλά έθεσαν τα θεμέλια για τις μετέπειτα εξελίξεις στην τεχνολογία IC.

(1.2 Σημασία των Ολοκληρωμένων Κυκλωμάτων στη Σύγχρονη Τεχνολογία)

Τα ολοκληρωμένα κυκλώματα έχουν φέρει επανάσταση σε διάφορους κλάδους, διαδραματίζοντας κρίσιμο ρόλο στην πρόοδο της σύγχρονης τεχνολογίας. Η σημασία των ολοκληρωμένων κυκλωμάτων πηγάζει από αρκετούς παράγοντες:

1. Ελαχιστοποίηση μεγέθους [Miniaturization]: Με την ενσωμάτωση πολλαπλών ηλεκτρονικών εξαρτημάτων σε ένα μόνο τσιπ, τα ολοκληρωμένα κυκλώματα επιτρέπουν την σμίκρυνση των ηλεκτρονικών συστημάτων. Αυτό οδήγησε στην ανάπτυξη συσκευών μικρού μεγέθους, όπως smartphone, φορητούς υπολογιστές και διάφορες φορητές συσκευές, καθιστώντας την τεχνολογία πιο φορητή και προσβάσιμη [4].
2. Βελτιωμένη απόδοση [Performance Enhancement]: Τα ολοκληρωμένα κυκλώματα προσφέρουν υψηλές ταχύτητες επεξεργασίας σήματος και αποθήκευσης δεδομένων, οδηγώντας σε βελτιωμένη απόδοση του συστήματος. Αυτές οι εξελίξεις έχουν διευκολύνει καινοτομίες σε τομείς όπως η τεχνητή νοημοσύνη, μηχανική μάθηση [Machine Learning] [5].
3. Ενεργειακή απόδοση [Power Optimization]: Η συμβολή στην δημιουργία ενεργειακά αποδοτικών κυκλωμάτων, επιτρέποντας την ανάπτυξη συσκευών που τροφοδοτούνται από μπαταρίες με εκτεταμένη διάρκεια ζωής της μπαταρίας. Τα ενεργειακά αποδοτικά σχέδια IC έχουν διαδραματίσει κρίσιμο ρόλο στην ανάπτυξη των συστημάτων ανανεώσιμων πηγών ενέργειας και του οικοσυστήματος του Διαδικτύου των Πραγμάτων [Internet Of Things - IoT] [6].
4. Μείωση κόστους [Cost Reduction]: Η μαζική παραγωγή ολοκληρωμένων κυκλωμάτων έχει μειώσει σημαντικά το κόστος ηλεκτρονικών εξαρτημάτων. Αυτή η μείωση του κόστους έχει καταστήσει την τεχνολογία πιο προσιτή και προσβάσιμη, ενθαρρύνοντας την καινοτομία και την οικονομική ανάπτυξη σε διάφορους κλάδους [7].

ΚΕΦΑΛΑΙΟ 2 ΦΥΣΙΚΗ ΣΧΕΔΙΑΣΗ : ΜΙΑ ΕΠΙΣΚΟΠΗΣΗ

Το στάδιο της φυσικής σχεδίασης παίζει κρίσιμο ρόλο στη συνολική διαδικασία σχεδιασμού ολοκληρωμένων κυκλωμάτων. Περιλαμβάνει τον μετασχηματισμό μιας λογικής αναπαράστασης κυκλώματος σε μια φυσική διάταξη που μπορεί να κατασκευαστεί και να παρασκευαστεί. Αυτή η ενότητα παρέχει μια επισκόπηση στα στάδια της φυσικής σχεδίασης, τονίζοντας τη σημασία του και συζητώντας τα κύρια βήματα που εμπλέκονται.

(2.1 Ρόλος της Φυσικής Σχεδίασης στην Συνολική Διαδικασία Σχεδιασμού IC)

Το στάδιο της φυσικής σχεδίασης χρησιμεύει ως κρίσιμη γέφυρα μεταξύ του λογικού σχεδιασμού και της κατασκευής ενός ολοκληρωμένου κυκλώματος. Επικεντρώνεται στη μετάφραση της αφηρημένης αναπαράστασης ενός κυκλώματος σε μια συγκεκριμένη διάταξη που μπορεί να υλοποιηθεί σε μια επιφάνεια πυριτίου ή άλλα υλικά ημιαγωγών. Ο πρωταρχικός στόχος της φυσικής σχεδίασης είναι να βελτιστοποιήσει διάφορες πτυχές της διάταξης του κυκλώματος, συμπεριλαμβανομένης της περιοχής [Area Optimization], της ισχύος [Power Optimization], της απόδοσης [Performance Enhancement] και της δυνατότητας κατασκευής [Design for Manufacturability].

Κατά το στάδιο της φυσικής σχεδίασης, οι σχεδιαστές χρησιμοποιούν ηλεκτρονικά εργαλεία αυτοματοποιημένης σχεδίασης [Electronic Design Automation [EDA]] για να πραγματοποιήσουν μια σειρά από μετασχηματισμούς στο netlist, η οποία δημιουργείται μετά από λογική σύνθεση [Logic Synthesis]. Αυτοί οι μετασχηματισμοί εξασφαλίζουν ότι η τελική διάταξη πληροί τις απαιτούμενες προδιαγραφές, λαμβάνοντας υπόψη τους περιορισμούς της διαδικασίας κατασκευής.

(2.2 Τα Κύρια Βήματα στο Στάδιο της Φυσικής Σχεδίασης)

Το στάδιο της φυσικής σχεδίασης αποτελείται από αρκετά βασικά βήματα που καθορίζουν την τελική διάταξη του ολοκληρωμένου κυκλώματος. Αυτά τα βήματα εκτελούνται συνήθως διαδοχικά, με κάθε ένα από αυτά να βασίζεται στα αποτελέσματα του προηγούμενου. Τα κύρια βήματα που περιλαμβάνονται στο στάδιο της φυσικής σχεδίασης είναι τα εξής:

1. Design Netlist [After Synthesis] : Η φυσική σχεδίαση ξεκινά με την Design Netlist, η οποία αναπαριστά το λογικό κύκλωμα μετά την synthesis. Αυτή περιλαμβάνει gates, flip-flops και interconnections.

Η Design Netlist παράγεται από το στάδιο της Synthesis, όπου η υψηλού επιπέδου περιγραφή του κυκλώματος μετατρέπεται σε ένα logical κύκλωμα που αποτελείται από λογικές πύλες και flip-flops, αναπαριστώντας τη δομή και τη λειτουργία του κυκλώματος σε ένα επίπεδο υψηλής αφαίρεσης [Abstract]. Αποτελεί τη βάση για την φυσική σχεδίαση, καθώς παρέχει την αναπαράσταση των λογικών στοιχείων που θα πρέπει να υλοποιηθούν φυσικά, χρησιμοποιώντας τρανζίστορ, αντιστάσεις και άλλα εξαρτήματα.

2. Floorplanning : Περιλαμβάνει τον καθορισμό της συνολικής περιοχής του ενσωματωμένου κυκλώματος και τη διάταξη των κύριων λειτουργικών μπλοκ ή ενοτήτων μέσα σε αυτόν τον χώρο. Ο σχεδιασμός χώρου καθορίζει τα όρια των μπλοκ και τις θέσεις τους, λαμβάνοντας υπόψη παράγοντες όπως η απόδοση, η ισχύς και οι περιορισμοί στην περιοχή [8].

Κατά το Floorplanning, οι σχεδιαστές προσδιορίζουν τα μπλοκ, όπως τους επεξεργαστές, τις μνήμες και τις άλλες λειτουργικές μονάδες, στον διαθέσιμο χώρο στην επιφάνεια του ενσωματωμένου κυκλώματος. Οι αποφάσεις που λαμβάνονται κατά τον σχεδιασμό χώρου επηρεάζουν την απόδοση του κυκλώματος, τη διάταξη των συνδέσεων, την θερμική διαχείριση και άλλες σημαντικές παραμέτρους του συστήματος.

3. Partitioning: Ο διαμερισμός χωρίζει τον σχεδιασμό σε μικρότερα, διαχειρίσιμα μπλοκ ή διαμερίσματα. Αυτό το βήμα διευκολύνει την αρθρωτότητα [modularity], παρέχοντας επίσης ευκολία στον σχεδιασμό και την αποδοτική χρήση πόρων. Παράλληλα, ο διαμερισμός μπορεί να βασίζεται στη λειτουργικότητα, την απόδοση, τις απαιτήσεις ισχύος ή άλλα κριτήρια [9].

Επιπρόσθετα, ο διαμερισμός συμβάλλει στην δημιουργία μονάδων ή μπλοκ καθιστώντας τη διαχείριση τους ευκολότερη καθώς

παρέχει ευελιξία στον σχεδιασμό τους. Ο σκοπός του διαμερισμού είναι να διευκολύνει την ανεξάρτητη ανάπτυξη, τον έλεγχο και την επαναχρησιμοποίηση των μπλοκ, επιτρέποντας την παράλληλη εργασία πάνω σε διάφορες μονάδες. Επίσης, ο διαμερισμός μπορεί να βελτιώσει την απόδοση και την αξιοποίηση των πόρων, καθώς επιτρέπει την βέλτιστη τοποθέτηση και δρομολόγηση των μπλοκ στο ενσωματωμένο κύκλωμα.

4. Placement: Η τοποθέτηση καθορίζει τις συγκεκριμένες θέσεις των μεμονωμένων κυψελών ή στοιχείων εντός κάθε διαμερίσματος στο chip. Ο στόχος της είναι η βελτιστοποίηση της εγγύτητας, μειώνοντας τις καθυστερήσεις των διασυνδέσεων και βελτιστοποιώντας τα χαρακτηριστικά ισχύος και χρονισμού [10].

5. Clock-Tree Synthesis (CTS): Το CTS περιλαμβάνει το σχεδιασμό και τη βελτιστοποίηση του δικτύου διανομής ρολογιού. Ο στόχος του είναι να διασφαλίσει ότι το σήμα ρολογιού φθάνει σε όλα τα στοιχεία με ελάχιστη κλίση και σωστό χρονισμό [11].

Κατά τη σύνθεση του δέντρου ρολογιού, σχεδιάζεται ένα δίκτυο διανομής ρολογιού που εξασφαλίζει την ομοιόμορφη και ακριβή κατανομή του σήματος του σε όλα τα στοιχεία του κυκλώματος. Αυτό είναι σημαντικό για τη διασφάλιση της συγχρονισμένης λειτουργίας του κυκλώματος. Η βελτιστοποίηση του δικτύου διανομής του ρολογιού αποσκοπεί στη μείωση της απόκλισης (skew) του και στη διασφάλιση της σωστής χρονικής συμμόρφωσης των στοιχείων.

6. Routing: Η δρομολόγηση περιλαμβάνει τη δημιουργία διασυνδέσεων ή καλωδίωσης μεταξύ κυψελών ή στοιχείων. Δημιουργεί ηλεκτρικές συνδέσεις λαμβάνοντας υπόψη τους περιορισμούς σχεδιασμού όπως η ακεραιότητα του σήματος, ο χρονισμός και οι κανόνες κατασκευής [12].

Κατά τη δρομολόγηση, δημιουργούνται οι απαραίτητες διασυνδέσεις που επιτρέπουν την αποτελεσματική μεταφορά σημάτων μεταξύ των κυψελών ή στοιχείων του κυκλώματος. Βέβαια, η δρομολόγηση πρέπει να πληροί πολλαπλούς σχεδιαστικούς περιορισμούς και να λαμβάνει υπόψη την ακεραιότητα του σήματος, τη σωστή χρονική συμμόρφωση στους κανόνες κατασκευής που έχουν οριστεί για την κατασκευή του κυκλώματος.

7. Physical Verification: Η φυσική επαλήθευση ελέγχει εάν η διάταξη συμμορφώνεται προς τους κανόνες σχεδίασης και τους περιορισμούς που ορίζονται από τη διαδικασία κατασκευής. Περιλαμβάνει τον έλεγχο των παραβιάσεων των κανόνων σχεδίασης,

των απαιτήσεων απόστασης και των συγκεκριμένων παραμέτρων που αφορούν την κατασκευή [13].

Κατά τη φυσική επαλήθευση, γίνεται έλεγχος για τυχόν παραβιάσεις των κανόνων σχεδίασης που έχουν τεθεί κατά τη διαδικασία κατασκευής. Αυτό περιλαμβάνει τον έλεγχο των αποστάσεων μεταξύ των στοιχείων, τον έλεγχο για τις τηρούμενες αποστάσεις από άλλες δομές καθώς και τον έλεγχο για τις τηρούμενες περιοχές και παραμέτρους που είναι σημαντικές για την κατασκευή του κυκλώματος. Ο στόχος είναι να διασφαλιστεί ότι ο σχεδιασμός είναι συμβατός με τις δυνατότητες και τους περιορισμούς της διαδικασίας κατασκευής, εξασφαλίζοντας την αξιοπιστία και την ποιότητα του κυκλώματος.

8. Layout Post Processing με Mask Data Generation: Μετά τη φυσική επαλήθευση, εκτελείται μια σειρά επιπρόσθετων επεξεργασιών για να βελτιστοποιηθεί περαιτέρω η διάταξη. Αυτές οι επεξεργασίες μπορεί να περιλαμβάνουν την εισαγωγή ειδικών κελιών (filler cells), την προσθήκη μετάλλου [metal fill] για να διατηρηθεί η ισορροπία μεταξύ των μεταλλικών στρωμάτων και την προσθήκη ανενεργών κελιών [dummy cells]. Τέλος, δημιουργείται η Mask Data που αναπαριστά τη διάταξη με μια μορφή που είναι κατάλληλη για την κατασκευή του κυκλώματος [14].

Αυτά τα βασικά βήματα στο στάδιο της φυσικής σχεδίασης συνεισφέρουν συλλογικά στη δημιουργία μιας διάταξης που είναι κατάλληλη για την κατασκευή και βελτιστοποίηση ενός ολοκληρωμένου κυκλώματος. Παρακάτω, θα ακολουθήσει μια περαιτέρω ανάλυση αυτών.

(2.2.1 Design Netlist (After Synthesis))

Στο στάδιο της Design Netlist (After Synthesis), η λογική περιγραφή του ολοκληρωμένου κυκλώματος μετατρέπεται σε ένα Netlist, το οποίο λειτουργεί ως αφετηρία για τα επόμενα βήματα της φυσικής σχεδίασης [1]. Το Netlist αναπαριστά το κύκλωμα ως μια συλλογή από συνδεδεμένα κελιά ή συστατικά, περιλαμβάνοντας τους τύπους τους, πληροφορίες συνδεσιμότητας και άλλα σημαντικά χαρακτηριστικά. Παρέχει μια ιεραρχική αναπαράσταση του σχεδίου και αποτυπώνει τις δομικές και λειτουργικές πτυχές του κυκλώματος.

Το Netlist διαδραματίζει έναν κρίσιμο ρόλο στη φυσική σχεδίαση, λειτουργώντας ως είσοδος για διάφορους αλγορίθμους βελτιστοποίησης και τοποθέτησης. Παρέχει τις απαραίτητες πληροφορίες για τα επόμενα στάδια, όπως η floorplanning, partitioning, placement, και routing. Αδιαμφισβήτητα, η ακρίβεια του netlist είναι ζωτικής σημασίας για την επίτευξη βέλτιστης απόδοσης, κατανάλωσης ισχύος και αξιοποίησης περιοχής στην τελική διάταξη.

Παράλληλα, τα εργαλεία σύνθεσης χρησιμοποιούνται για τη δημιουργία του Netlist από την RTL [Register-Transfer Level] περιγραφή του κυκλώματος που αποτελεί μια αφαιρετική σχεδίαση της λογικής του κυκλώματος και συνήθως περιγράφεται με γλώσσες περιγραφής υλικού όπως την VHDL και την Verilog [2]. Αυτά τα εργαλεία αναλύουν την behavioral περιγραφή του σχεδίου, την αντιστοίχιση των στοιχείων RTL σε συγκεκριμένα κελιά ή οντότητες [modules] και δημιουργούν μια βελτιστοποιημένη Netlist βάση των καθορισμένων κανόνων. Συνάμα, συμβάλλουν στη βελτιστοποίηση του κυκλώματος σχετικά με την περιοχή [area], την ισχύ [power] ή τον χρονισμό [timing] και διευκολύνουν την επίλυση οποιουδήποτε περιορισμού σχεδίασης [Design Constraints].

Είναι σημαντικό να εξασφαλίζεται η ορθότητα των αποτελεσμάτων της σύνθεσης προκειμένου να αποφεύγονται οποιαδήποτε σφάλματα ή ακατάλληλες σχεδιαστικές επιλογές στα επόμενα στάδια της φυσικά σχεδίασης. Για αυτόν τον λόγο, οι τεχνικές επαλήθευσης, όπως ο έλεγχος ισοδυναμίας, χρησιμοποιούνται για τη σύγκριση της συμπεριφοράς της αρχικής RTL περιγραφής με το design netlist μετά την σύνθεση προκειμένου να εξασφαλιστεί η σωστή λειτουργία.

Επιπλέον, μπορούν να χρησιμοποιηθούν μαθηματικές μέθοδοι επαλήθευσης για τον έλεγχο ιδιοτήτων του σχεδίου, όπως η clock-domain crossing ή η βελτιστοποίηση της ισχύος. Αυτές οι τεχνικές επαλήθευσης περιλαμβάνουν τη χρήση μαθηματικών μοντέλων και αλγορίθμων για την ακριβή ανάλυση του σχεδιασμού και την επαλήθευση συγκεκριμένων ιδιοτήτων ή συμπεριφορών. Στην περίπτωση της clock-domain crossing, η μαθηματική επαλήθευση μπορεί να εξασφαλίσει τον κατάλληλο συγχρονισμό και να εντοπίσει πιθανά προβλήματα, όπως η metastability. Όσον αφορά τη βελτιστοποίηση της ισχύος, οι αλγοριθμικές τεχνικές μπορούν να επαληθεύσουν ιδιότητες που σχετίζονται με την ισχύ και να βοηθήσουν στη βελτιστοποίηση της κατανάλωσης ισχύος κατά το στάδιο της φυσικής σχεδίασης.[3]

Η ακρίβεια και η ποιότητα του Netlist επηρεάζουν σημαντικά την αποτελεσματικότητα και αποδοτικότητα των επόμενων σταδίων. Οποιαδήποτε σφάλματα ή ασυνέπειες στη netlist μπορούν να επεκταθούν και να οδηγήσουν σε μη βέλτιστα κακής ποιότητας διατάξεις [Layout], σε παραβιάσεις χρονισμού [Timing Violations] ή σε προβλήματα κατασκευής. Επομένως, η λεπτομερής επαλήθευση και επικύρωση είναι ζωτικής σημασίας για τη διατήρηση της ακεραιότητας για την δημιουργία ενός IC.

(2.2.2 Partitioning)

Το πρώτο βήμα στη φυσική σχεδίαση ολοκληρωμένων κυκλωμάτων [ICs] είναι το partitioning που περιλαμβάνει την διαδικασία της διαίρεσης του chip σε μικρότερα λειτουργικά blocks ή modules. Αδιαμφισβήτητα, παίζει ζωτικό ρόλο στην επίτευξη της αποδοτικότητας του σχεδίου, τη διευκόλυνση των parallel εργασιών και τη βέλτιστη χρήση των διαθέσιμων πόρων.

Ο στόχος του partitioning είναι να καθοριστεί η βέλτιστη κατανομή των λειτουργικών μονάδων , όπως των logic modules, των memory blocks ή των στοιχείων επεξεργασίας, σε διάφορα partitions. Κάθε partition αντιπροσωπεύει μια αυτόνομη μονάδα που μπορεί να σχεδιαστεί, να βελτιστοποιηθεί και να επαληθευτεί ανεξάρτητα. Η διαδικασία partitioning αποσκοπεί στην ισορροπία παραγόντων όπως είναι η χρήση της περιοχής , η πολυπλοκότητα των συνδέσεων και οι απαντήσεις απόδοσης.

Βέβαια, το partitioning μπορεί να πραγματοποιηθεί σε διάφορα επίπεδα, ανάλογα με την πολυπλοκότητα του σχεδίου. Πιο συγκεκριμένα, μπορεί να περιλαμβάνει τη διαίρεση σε partitions μεγάλης κλίμακας [macro-level] που αντιπροσωπεύουν κύριες λειτουργικές μονάδες ή την κατανομή του σε partitions μικρότερης κλίμακας που αντιπροσωπεύουν μικρότερα logic blocks. Προφανώς, η επιλογή του επιπέδου λεπτομέρειας του partitioning εξαρτάται από πολλούς παράγοντες όπως στην περίπτωση που το σχέδιο απαιτείται να είναι modular καθώς και από τους ήδη προαναφερθέντες.

Σημειωτέον ότι διάφορες μεθοδολογίες και αλγόριθμοι partitioning χρησιμοποιούνται στην φυσική σχεδίαση. Αυτές περιλαμβάνουν αλγόριθμους βασισμένους είτε σε γράφους [Graph-based], είτε σε αναδρομική διχοτόμηση [Recursive Bisection], είτε σε φασματική διχοτόμηση [Spectral Partitioning] ή σε επαναληπτικές μεθόδους [Iterative].

1. **Graph-based** : Οι αλγόριθμοι βασισμένοι σε γράφους μοντελοποιούν ένα γράφο, όπου οι κόμβοι αντιπροσωπεύουν τις λειτουργικές μονάδες και οι ακμές αντιπροσωπεύουν τις συνδέσεις μεταξύ τους. Αυτοί οι αλγόριθμοι αποσκοπούν στην εύρεση partitions που ελαχιστοποιούν τον αριθμό των ακμών μεταξύ των partitions ενώ ισορροπούν το μέγεθος αυτών
2. **Recursive Bisection** : Η αναδρομική διχοτόμηση είναι μια δημοφιλής τεχνική partitioning που διαιρεί αναδρομικά το chip σε δύο partitions μέχρι να επιτευχθεί ένα επιθυμητό μέγεθος ή μία ισορροπία. Αυτή η προσέγγιση είναι αποδοτική και μπορεί να χειριστεί μεγάλης κλίμακας ICs, αλλά μπορεί να οδηγήσει σε μη ικανοποιητικές λύσεις εάν οι συνδέσεις μεταξύ των στοιχείων έχουν μεγάλη πολυπλοκότητα.
3. **Spectral Partitioning** : Οι τεχνικές φασματικής διχοτόμησης αξιοποιούν τις [spectral] ιδιότητες του γράφου για να διαιρέσουν το σχεδιασμό σε partitions με καλή συνδεσιμότητα και ισορροπία.
4. **Iterative** : Οι επαναληπτικές μέθοδοι συνδυάζουν το partitioning με τα επόμενα βήματα βελτιστοποίησης προκειμένου να βελτιώσουν τη λύση του partitioning επαναληπτικά.

Φυσικά, αυτές οι μέθοδοι λαμβάνουν υπόψη ορισμένους παράγοντες όπως τους περιορισμούς χρονισμού, την κατανάλωση ισχύος και τη χρήση της περιοχής ούτως ώστε να καθοδηγήσουν τη διαδικασία του partitioning. Αναντίρρητα, με την επαναλαμβανόμενη βελτιστοποίηση των partitions, αυτές οι μέθοδοι μπορούν να αναβαθμίσουν τη συνολική ποιότητα του σχεδιασμού και να επιτύχουν τους στόχους της φυσικής σχεδίασης.

Εργαλεία και μεθοδολογίες partitioning ενσωματώνονται στα σύγχρονα ηλεκτρονικά εργαλεία αυτοματοποιημένης σχεδίασης [EDA] που χρησιμοποιούνται στη φυσική σχεδίαση. Αυτά τα εργαλεία παρέχουν δυνατότητες όπως το constraint-driven partitioning, την απεικόνιση των αποτελεσμάτων του partitioning και την ανάλυση της πολυπλοκότητας των συνδέσεων.

Για την επιτυχή υλοποίηση του partitioning απαιτείται προσεκτική μελέτη διάφορων συντελεστών, συμπεριλαμβανομένων των interconnectivity, timing requirements, power distribution, και area utilization. Απαιτείται επίσης κατανόηση των λειτουργικών μονάδων, των εξαρτήσεων τους και των συμβιβασμών (trade-offs) που σχετίζονται με τις επιλογές των διάφορων μεθόδων.

(2.2.3 Floorplanning)

Το Floorplanning είναι το δεύτερο στάδιο στη φυσική σχεδίαση ολοκληρωμένων κυκλωμάτων [ICs] που περιλαμβάνει τον καθορισμό της βέλτιστης τοποθέτησης διάφορων λειτουργικών blocks ή modules στην επιφάνεια του chip. Καθορίζει τη συνολική διάταξη του chip, συμπεριλαμβανομένης της τοποθέτησης των κύριων components, την διάταξη της διαθέσιμης περιοχής και την τακτοποίηση των διασυνδέσεων.

Ο κύριος στόχος του Floorplanning είναι να επιτευχθεί αποδοτική χρήση του chip, ενώ τηρούνται οι απαιτήσεις και περιορισμοί του σχεδιασμού. Αυτό περιλαμβάνει την ελαχιστοποίηση της συνολικής περιοχής του chip, την βελτιστοποίηση του μήκους των καλωδίων, τη διαχείριση της κατανομής ισχύος και τη διασφάλιση κατάλληλων θερμικών χαρακτηριστικών. Αναμφίβολα, ένα αποτελεσματικό Floorplanning μπορεί να επηρεάσει σημαντικά την απόδοση, την κατανάλωση ισχύος και τη συνολική κατασκευαστική δυνατότητα του IC.

Η διαδικασία του Floorplanning ξεκινά με μια αρχική εκτίμηση των απαιτήσεων περιοχής του chip, με βάση το μέγεθος και τον αριθμό των λειτουργικών blocks, των στοιχείων μνήμης και των διεπαφών [Pads] εισόδου/εξόδου [Input/Output, I/O] . Ο σχεδιαστής λαμβάνει επίσης υπόψη συγκεκριμένους περιορισμούς σχεδιασμού, όπως τις απαιτήσεις για power delivery network requirements, clock distribution και την ακεραιότητα του σήματος. Επιπλέον, λαμβάνονται υπόψη παράγοντες όπως η συμφόρηση της δρομολόγησης [Routing], η πυκνότητα τοποθέτησης και η απόσταση από κρίσιμα components.

Εκτός τούτου, για τη δημιουργία ενός βελτιστοποιημένου Floorplanning, χρησιμοποιούνται διάφορες τεχνικές και μεθοδολογίες. Αυτές περιλαμβάνουν αναλυτικές μεθόδους [Analytical Methods], αλγόριθμους βελτιστοποίησης [Optimization Algorithms] και ευρετικές [Heuristic-Based] προσεγγίσεις. Οι αναλυτικές μέθοδοι χρησιμοποιούν μαθηματικές διατυπώσεις και objective συναρτήσεις για να προσδιορίσουν το Floorplanning βάσει συγκεκριμένων μετρήσεων, όπως το μήκος των καλωδίων, η ισχύς ή ο χρονισμός. Οι αλγόριθμοι βελτιστοποίησης, όπως ο simulated annealing ή οι γενετικοί αλγόριθμοι, βελτιστοποιούν το σχέδιο μέσω της εξερεύνησης διαφορετικών επιλογών τοποθέτησης. Αξίζει, βέβαια, να σημειωθεί ότι οι heuristic-based προσεγγίσεις εκμεταλλεύονται την τεχνογνωσία και τις γνώσεις αυτού του τομέα για να καθοδηγήσουν αποτελεσματικά τη διαδικασία .

Σύγχρονα εργαλεία Floorplanning ενσωματώνουν προηγμένους αλγόριθμους και παρέχουν γραφικό περιβάλλον ώστε να υπάρχει δυνατότητα οπτικοποίησης και να αλληλεπιδράσης. Αυτά τα εργαλεία προσφέρουν χαρακτηριστικά όπως την τοποθέτηση με κατευθυνόμενους περιορισμούς, την αυτόματη δημιουργία δικτύου ισχύος και ρολογιού, την ανάλυση σε πραγματικό χρόνο καθώς και την οπτικοποίηση των μετρήσεων. Παράλληλα, επιτρέπουν στους σχεδιαστές να εξερευνήσουν διάφορες εναλλακτικές λύσεις , να αξιολογήσουν τις

συμβιβαστικές λύσεις και να βελτιστοποιήσουν το chip με επαναλαμβανόμενες διορθώσεις.

Για την επιτυχημένη εκτέλεση του Floorplanning, απαιτείται μια ισορροπία μεταξύ πολλαπλών αντικρουόμενων στόχων, όπως η ελαχιστοποίηση του wirelength έναντι της ελαχιστοποίησης της κατανάλωσης ισχύος ή η βελτιστοποίηση της πυκνότητας placement έναντι της διασφάλισης της ακεραιότητας του σήματος. Οπωσδήποτε, οι σχεδιαστές πρέπει να λάβουν υπόψη την επίδραση του στα επόμενα στάδια, όπως η διαίρεση και η τοποθέτηση, καθώς ένας καλά σχεδιασμός μπορεί να απλοποιήσει αυτές τις εργασίες και να βελτιώσει τη συνολική ποιότητα του ολοκληρωμένου κυκλώματος, παρέχοντας μια αρχική δομή που διευκολύνει την αποτελεσματική συνδεσιμότητα και τη βέλτιστη διάθεση των πόρων.

(2.2.4 Placement)

Το Placement αφορά τον προσδιορισμό της βέλτιστης θέσης για κάθε μεμονωμένο στοιχείο [cell] στην επιφάνεια chip. Ο πρωταρχικός στόχος είναι η τοποθέτησή των cells με τέτοιο τρόπο ώστε να μειωθεί το wire-length , η καθυστέρηση του σήματος και να βελτιστοποιηθούν διάφορες μετρήσεις , όπως η κατανάλωση ενέργειας και τα timing constraints.

Η διαδικασία της τοποθέτησης ξεκινά με μια netlist, η οποία προσδιορίζει τις λογικές συνδέσεις μεταξύ διαφορετικών στοιχείων. Η αρχική τοποθέτηση συνήθως περιλαμβάνει τυχαίες ή τις αρχικές θέσεις. Ωστόσο, εφαρμόζονται αλγόριθμοι και τεχνικές βελτιστοποίησης για τη βελτίωση της τοποθέτησης με βάση τους περιορισμούς και τους στόχους.

Επιπροσθέτως, οι αλγόριθμοι τοποθέτησης χρησιμοποιούν διάφορες προσεγγίσεις για να επιτύχουν βέλτιστα αποτελέσματα. Ειδικότερα, λαμβάνουν υπόψη παράγοντες όπως η πυκνότητα των cell , οι χρονικοί περιορισμοί, η κατανομή ισχύος, η ακεραιότητα του σήματος και οι θερμικές προϋποθέσεις. Οι συνηθισμένες τεχνικές περιλαμβάνουν την τοποθέτηση με Force-Directed, αναλυτικές [Analytical] μεθόδους, Simulated Annealing, γενετικούς [Genetic] αλγόριθμους και μεθόδους επαναλαμβανόμενης βελτίωσης [Iterative Improvement].

1. Το Force-directed placement είναι μια ευρέως χρησιμοποιηθέντα τεχνική που μοντελοποιεί τη διαδικασία τοποθέτησης ως ένα φυσικό σύστημα, όπου τα cells θεωρούνται σαν charged particles και intercellular δυνάμεις καθορίζουν την θέση τους. Ο στόχος είναι να επιτευχθεί ένα ισορροπημένο και σταθερό σύστημα που ελαχιστοποιεί τη συνολική potential ενέργεια.
2. Οι αναλυτικές μέθοδοι εκμεταλλεύονται μαθηματικές τυποποιήσεις και objective συναρτήσεις για να βελτιστοποιήσουν την τοποθέτηση βάσει συγκεκριμένων μετρικών [Wirelength, Power, Timing].
3. Οι Simulated annealing and genetic algorithms είναι heuristic-based τεχνικές. Η τεχνική Simulated annealing μιμείται τη διαδικασία της ψύξης στη μεταλλουργία, όπου ένα σύστημα ψύχεται σταδιακά για να ελαχιστοποιηθούν τα ελαττώματα.
4. Οι γενετικοί αλγόριθμοι μιμούνται τη βιολογική εξέλιξη, χρησιμοποιώντας μηχανισμούς όπως η selection, crossover, και mutation για να βρεθεί η βέλτιστη θέση.
5. Οι μέθοδοι επαναληπτικής βελτίωσης, γνωστές και ως iterative legalization, ξεκινούν με μια αρχική τοποθέτηση και τη βελτιώνουν επαναληπτικά εφαρμόζοντας μετασχηματισμούς όπως εναλλαγή

κελιών, αλλαγή μεγέθους ή μετακίνηση. Αυτές οι μέθοδοι λαμβάνουν υπόψη συγκεκριμένους περιορισμούς και μετρήσεις κατά τη διάρκεια των επαναλήψεων για τη σταδιακή βελτιστοποίηση του placement[3].

(2.2.5 Clock-Tree Synthesis – CTS))

Επικεντρώνεται στην αποτελεσματική διανομή των σημάτων ρολογιού στο κύκλωμα. Ταυτόχρονα, εξασφαλίζει το σωστό συγχρονισμό των σημάτων και στην ελαχιστοποίηση της απόκλισης του ρολογιού [clock skew], που αναφέρεται στη διακύμανση των χρόνων άφιξης των σημάτων σε διάφορα στοιχεία του chip.

Ο στόχος του CTS είναι να κατασκευάσει ένα καλά ισορροπημένο και χαμηλής κλίσης δέντρο ρολογιού που παρέχει αξιόπιστα σήματα σε όλα τα στοιχεία του σχεδίου. Ένα καλά δέντρο ρολογιού είναι απαραίτητο για την επίτευξη αξιόπιστης και σταθερής λειτουργίας κυκλωμάτων και την ικανοποίηση των απαιτήσεων χρονισμού.

Η διαδικασία CTS ξεκινά με τη Netlist σαν είσοδο, η οποία περιλαμβάνει τα διαδοχικά στοιχεία και τις πηγές σήματος ρολογιού. Ο πρωταρχικός στόχος είναι να δημιουργηθεί ένα δίκτυο διανομής που ελαχιστοποιεί την απόκλιση του ρολογιού και βελτιστοποιεί τις μετρήσεις όπως είναι αυτές της ισχύος, της περιοχής και της ακεραιότητας του σήματος.

Το CTS περιλαμβάνει τα εξής κύρια βήματα:

1. Την Κατασκευή Δέντρου Ρολογιού [Clock Tree Construction]
2. Την ισορροπία Δέντρου Ρολογιού [Clock Tree Balancing]
3. Την βελτιστοποίηση Δέντρου Ρολογιού
[Clock Tree Optimization]

Το δέντρο ρολογιού κατασκευάζεται επιλέγοντας κατάλληλα στοιχεία, συμπεριλαμβανομένων ενισχυτών [Buffers] και επαναληπτών [Repeaters], και συνδέοντάς τα για να δημιουργηθεί μια ιεραρχική δομή δέντρου.

Παράλληλα, η επίτευξη ισορροπημένης διανομής των σημάτων για την ελαχιστοποίηση της απόκλισης, γίνεται με τη στρατηγική τοποθέτηση των στοιχείων.

Αδιαμφισβήτητα, η αποτελεσματική σύνθεση δέντρου ρολογιού είναι κρίσιμη για την εκπλήρωση των αυστηρών απαιτήσεων χρονισμού των IC. Πιο συγκεκριμένα, βοηθά στον επαρκή συγχρονισμό των σημάτων, μειώνει την απόκλιση, ελαχιστοποιεί τις παραβιάσεις του χρόνου και κατορθώνει την αξιόπιστη λειτουργία κυκλωμάτων.

(2.2.6 Routing)

Η δρομολόγηση στοχεύει στην βέλτιστη καλωδίωση του σχεδιασμού, ελαχιστοποιώντας το μήκος των καλωδίων, τη συμφόρηση και την καθυστέρηση του σήματος, ενώ τηρεί τους κανόνες σχεδίασης και περιορισμούς.

Οι αλγόριθμοι δρομολόγησης λειτουργούν με βάση μια detailed placement λύση, λαμβάνοντας υπόψη τη Netlist, τους περιορισμούς και τους σχεδιαστικούς στόχους. Ο κύριος στόχος είναι η δημιουργία αποτελεσματικών και αξιόπιστων διασυνδέσεων που να ικανοποιούν τις απαιτήσεις χρονισμού, να ελαχιστοποιούν την καθυστέρηση του σήματος και να εξασφαλίζουν την ακεραιότητα του σήματος.

Η διαδικασία δρομολόγησης αποτελείται συνήθως από τα εξής βήματα:

1. Netlist Analysis : Ο αλγόριθμος δρομολόγησης αναλύει τις πληροφορίες συνδεσιμότητας που παρέχονται από τη Netlist, συμπεριλαμβανομένης της πηγής και του προορισμού κάθε net, μαζί με τυχόν απαιτούμενους περιορισμούς δρομολόγησης.
2. Καθολική δρομολόγηση [Global Routing] : Στη φάση αυτή, καθορίζεται μια διαδρομή υψηλού επιπέδου για κάθε net, λαμβάνοντας υπόψη τη συνολική τοπολογία. Ειδικότερα, οι αλγόριθμοι Καθολικής δρομολόγησης βελτιστοποιούν το μήκος των καλωδίων και τη συμφόρηση, στοχεύοντας στην εύρεση της πιο αποδοτικής διαδρομής, αποφεύγοντας τις επικαλύψεις [Overlaps] και τις παραβιάσεις των κανόνων σχεδίασης.
3. Λεπτομερής δρομολόγηση [Detailed Routing] : Αφού ολοκληρωθεί το Global Routing, πραγματοποιείται η λεπτομερής δρομολόγηση για την εδραίωση των πραγματικών φυσικών συνδέσεων μεταξύ της πηγής και του προορισμού κάθε net. Οι αλγόριθμοι λεπτομερούς δρομολόγησης λαμβάνουν υπόψη συγκεκριμένους περιορισμούς, όπως οι αναθέσεις επιπέδου [Circuit Layer], τοποθέτησης [Placement] και διάστασης [Spacing], προκειμένου να επιτευχθούν ακριβείς λύσεις δρομολόγησης.
4. Μετα-επεξεργασία [Post-Processing] : Μετά την ολοκλήρωση της δρομολόγησης, μπορούν να εκτελεστούν βήματα Post-Processing για περαιτέρω βελτιστοποίηση. Αυτό μπορεί να περιλαμβάνει τη διασπορά των καλωδίων [wire spreading], την απομόνωση του σήματος [signal shielding] και άλλες τεχνικές για τη βελτίωση της ακεραιότητας του σήματος και την ελαχιστοποίηση της συμπλοκής θορύβου [noise coupling].

Αξίζει να σημειωθεί ότι το Routing έχει σημαντικό ρόλο στην αντιμετώπιση των περιορισμών χρονισμού, στη μείωση της καθυστέρησης του σήματος και στη διασφάλιση της σωστής ακεραιότητας του σήματος σε όλο το chip.

(2.2.7 Physical Verification)

Η φυσική επαλήθευση αποτελεί ένα κρίσιμο βήμα στη διαδικασία σχεδιασμού ολοκληρωμένων κυκλωμάτων [ICs], το οποίο διασφαλίζει ότι ο σχεδιασμός συμμορφώνεται με τους κανόνες και τους περιορισμούς που καθορίζονται.[13],[23] Περιλαμβάνει διάφορους ελέγχους και αναλύσεις για την επαλήθευση της ορθότητας, της νομιμότητας [Legality] και της αξιοπιστίας της διάταξης [Layout] πριν από την κατασκευή.

Η διαδικασία φυσικής επαλήθευσης περιλαμβάνει τα παρακάτω κύριο βήματα.

1. Έλεγχος κανόνων σχεδίασης [Design Rule Checking [DRC]] : Επαληθευθεί αν ο σχεδιασμός συμμορφώνεται με τους καθορισμένους κανόνες που αφορούν το μέγεθος [Minimum Feature Size], τις αποστάσεις [Spacing], τις επικαλύψεις [Overlap] και την ευθυγράμμιση [Alignment]. Αυτό θα εξασφαλίζει ότι ο σχεδιασμός πληροί τις απαιτήσεις κατασκευής.[25],[20]
2. Έλεγχος Σχεδίου έναντι Σχηματολογίας [Layout versus Schematic [LVS] Checking] : Συγκρίνει τον σχεδιασμό με τον αρχικό κύκλωμα για να διασφαλίσει ότι η υλοποίηση αντιπροσωπεύει ακριβώς τον επιθυμητό σχεδιασμό . Αυτός ο έλεγχος είναι κρίσιμος για την επαλήθευση της ορθότητας των διασυνδέσεων [Interconnections], των μεγεθών των τρανζίστορ και της τοποθέτησης των στοιχείων. [13],[23],[26]
3. Έλεγχος Σχεδιασμού Κατασκευής [Design for Manufacturing [DFM] Checks] : Διεξάγονται επίσης ως μέρος της φυσικής επαλήθευσης. Αυτοί οι έλεγχοι αξιολογούν την κατασκευάσιμότητα του σχεδιασμού, λαμβάνοντας υπόψη ποικίλες διακυμάνσεις της διαδικασίας, τους περιορισμούς λιθογραφίας [Lithography Constraints] και τις άλλες απαιτήσεις που σχετίζονται με την κατασκευή. Η ανάλυση DFM αποσκοπεί στον εντοπισμό πιθανών προβλημάτων που μπορεί να επηρεάσουν την απόδοση [Yield], την αξιοπιστία και την απόδοση, όπως υπερβολική αλληλεπίδραση [Coupling], ηλεκτρική μετατόπιση [Electromigration] ή σχηματισμός hotspot [Hotspot Formation]. [14],[27]
4. Έλεγχος Κανόνων Κεραίας [Antenna Rule Checking] : Πραγματοποιείται για την αντιμετώπιση πιθανών προβλημάτων συσσώρευση φορτίου και αξιοπιστίας σε σχεδιασμούς sub-micron. Αυτό το βήμα επαλήθευσης διασφαλίζει τη συμμόρφωση με κανόνες που αποσκοπούν στην αποτροπή υπερβολικής συσσώρευσης φορτίου

[preventing excessive charge accumulation] και πιθανών αστοχιών [device failures]. [28],[25],[20]

5. Έλεγχος Ηλεκτρικών Κανόνων [Electrical Rule Checking [ERC]] : Πραγματοποιείται για την επαλήθευση της ηλεκτρικής συνδεσιμότητας [Electrical connectivity] και των πτυχών ακεραιότητας του σήματος [Signal Integrity] του σχεδιασμού. Η ανάλυση ERC διασφαλίζει την κατάλληλη σύνδεση της ισχύος [Power] και της γείωσης [Ground], απομόνωση [Isolation] και τη συνδεσιμότητα [Connectivity] των στοιχείων, εξαλείφοντας τυχόν βραχυκυκλώματα [Short Circuits], ανοιχτά κυκλώματα [Open Circuits] ή πλωτούς [Floating] κόμβους [19],[14]

Η φυσική επαλήθευση είναι μια επαναληπτική διαδικασία που περιλαμβάνει πολλαπλές τροποποιήσεων σχεδιασμού και εκ νέου επαλήθευση για την επίτευξη μιας διάταξης που ικανοποιεί όλους τους απαραίτητους περιορισμούς.

(2.2.8 Layout Post Processing με Mask Data Generation)

Μόλις ολοκληρωθούν τα στάδια φυσικής σχεδίασης και επαλήθευσης, η διάταξη υποβάλλεται σε μετα-επεξεργασία για να προετοιμαστεί για τη διαδικασία κατασκευής. Αυτό το κρίσιμο στάδιο περιλαμβάνει τη δημιουργία των δεδομένων μάσκας, τα οποία χρησιμεύουν ως σχέδιο για την κατασκευή του ολοκληρωμένου κυκλώματος [IC]. [23]

Η μετα-επεξεργασία του σχεδιασμού διάταξης περιλαμβάνει αρκετές σημαντικές εργασίες που στοχεύουν στη διασφάλιση της κατασκευασιμότητας και της ακρίβειας του IC. Ένα από αυτά τα βήματα είναι η διόρθωση του Έλεγχου Κανόνων Σχεδιασμού [Design Rule Checking [DRC]]. Κατά τη διάρκεια αυτού του βήματος, αντιμετωπίζονται οι παραβιάσεις που παραμένουν αναγνωρισμένες κατά τον φυσικό έλεγχο επαλήθευσης. Οι παραβιάσεις των κανόνων σχεδιασμού, όπως προβλήματα διάταξης και επικαλύψεις και διορθώνονται για να διασφαλιστεί ότι ο σχεδιασμός τηρεί τους καθορισμένους κανόνες . [20],[25]

Ένα άλλο κρίσιμο βήμα είναι η Οπτική Διόρθωση Εγγύτητας [Optical Proximity Correction [OPC]]. Το OPC αντισταθμίζει τις οπτικές παραμορφώσεις που συμβαίνουν κατά τη διαδικασία της λιθογραφίας. Τροποποιώντας τη γεωμετρία της διάταξης, οι αλγόριθμοι OPC βελτιστοποιούν την εκτυπωσιμότητα και την πιστότητα των μοτίβων στη γκοφρέτα πυριτίου [Silicon Wafer]. Αυτή η προσαρμογή των διαστάσεων και των σχημάτων των χαρακτηριστικών ενισχύει την ακρίβεια και την ανάλυση του τελικού IC. [14]

Μετά τις απαραίτητες διορθώσεις και βελτιστοποιήσεις, τα δεδομένα μάσκας δημιουργούνται χρησιμοποιώντας ηλεκτρονικά εργαλεία αυτοματοποιημένης σχεδίασης [EDA]. Τα δεδομένα μάσκας, που αναφέρονται επίσης ως φωτομάσκα [photomask] ή πλέγμα [Reticle], περιέχουν τις ακριβείς πληροφορίες μοτίβου

που θα μεταφερθούν στη γκοφρέτα πυριτίου κατά τη διαδικασία της λιθογραφίας. Αυτό το βήμα περιλαμβάνει τη μετάφραση του σχεδίου διάταξης σε μια μορφή κατάλληλη για την κατασκευή φωτομάσκας, όπως η μορφή [Graphic Data System version II [GDSII]]. [23] , [24]

Επιπλέον, τα δημιουργηθέντα δεδομένα μάσκας μπορεί να υποβληθούν σε επιπλέον επαλήθευση για να διασφαλιστεί η ορθότητά τους και η συμβατότητά τους με τη διαδικασία κατασκευής. Ο Έλεγχος Δεδομένων Μάσκας [Mask Data Checking [MDC]] πραγματοποιείται για να επαληθευτεί η ακρίβεια των δεδομένων μάσκας στο επίπεδο της μάσκας. Όπως και ο DRC, οι έλεγχοι MDC εξασφαλίζουν ότι τα χαρακτηριστικά της μάσκας τηρούν τους επιθυμητούς κανόνες σχεδίασης και προδιαγραφών, επαληθεύοντας την ακεραιότητα αυτών [24].

Η μετα-επεξεργασία του σχεδιασμού διάταξης και η δημιουργία δεδομένων μάσκας είναι κρίσιμες για την επίτευξη ενός IC υψηλής ποιότητας κατά τη διάρκεια της διαδικασίας κατασκευής. Τα δημιουργηθέντα δεδομένα μάσκας λειτουργούν ως βάση για τα επόμενα βήματα στη ροή κατασκευής ημιαγωγών [Semiconductors], εξασφαλίζοντας την επιτυχή υλοποίηση του IC.

ΚΕΦΑΛΑΙΟ 3

Στόχοι και Προκλήσεις στην Φυσική Σχεδίαση

Σε αυτήν την ενότητα, εξετάζουμε τους στόχους και τις προκλήσεις που αντιμετωπίζονται κατά το στάδιο της φυσικής σχεδίασης του ολοκληρωμένου κυκλώματος [IC]. Αρχικά, παρουσιάζουμε τους κύριους στόχους της φυσικής σχεδίασης [Ενότητα 3.1]. Αυτοί οι στόχοι περιλαμβάνουν την βελτιστοποίηση περιοχής [Area Optimization], την ενίσχυση της απόδοσης [Performance Enhancement], την βελτιστοποίηση της κατανάλωσης ισχύος [Power Optimization] και τον σχεδιασμό για κατασκευαστή [Design for Manufacturability [DFM]]. Στη συνέχεια συζητάμε τις προκλήσεις που αντιμετωπίζει ο φυσική σχεδίαση (Ενότητα 3.2), όπως την επίτευξη χρονισμού [Timing Closure], την πολυπλοκότητα του σχεδιασμού [Design Complexity], τη συμμόρφωση με τους κανόνες σχεδίασης [Design Rule Compliance], την ακεραιότητα του σήματος [Signal Integrity] και τους περιορισμούς σχεδίασης [Design Constraints]. Κατανοώντας αυτούς τους στόχους και τις προκλήσεις, οι σχεδιαστές μπορούν αποτελεσματικά να επιτύχουν τη βέλτιστη απόδοση και κατασκευή ενός IC.

(3.1 Στόχοι της φυσικής σχεδίασης)

Οι στόχοι της φυσικής σχεδίασης περιλαμβάνουν τη βελτιστοποίηση διαφόρων πτυχών της διάταξης του ολοκληρωμένου κυκλώματος. Αυτοί οι στόχοι περιλαμβάνουν την ελαχιστοποίηση της περιοχής που καταλαμβάνει το IC, τη βελτίωση της απόδοσης, τη μείωση της κατανάλωσης ενέργειας και τη διασφάλιση της δυνατότητας κατασκευής του σχεδίου. Προφανώς, η επίτευξη αυτών των στόχων απαιτεί αποτελεσματικές τεχνικές τοποθέτησης και δρομολόγησης, εξέταση των χρονικών περιορισμών και τήρηση κανόνων και περιορισμών σχεδιασμού.

1. Βελτιστοποίηση Περιοχής [Area Optimization] : Η φυσική σχεδίαση στοχεύει στην ελαχιστοποίηση της περιοχής που καταλαμβάνει το ολοκληρωμένο κύκλωμα ενώ πληροί τις καθορισμένες απαιτήσεις λειτουργικότητας. Ασφαλώς, χρησιμοποιούνται αποδοτικές τεχνικές τοποθέτησης και δρομολόγησης για την επίτευξη βέλτιστης της περιοχής [15].

2. Βελτίωση απόδοσης [Performance Enhancement] : Η φυσική σχεδίαση προσπαθεί να βελτιώσει τη συνολική απόδοση του κυκλώματος. Αυτό περιλαμβάνει τη βελτιστοποίηση της καθυστέρησης διάδοσης του σήματος, τη μείωση των μηκών διασύνδεσης και την ελαχιστοποίηση parasitic effects στην χωρητικότητα και την αντίσταση [5].

3. Βελτιστοποίηση Κατανάλωσης Ισχύος [Power Optimization] : Αδιαμφισβήτητα, η κατανάλωση ισχύος αποτελεί κρίσιμο παράγοντα στον σχεδιασμό ολοκληρωμένων κυκλωμάτων [IC]. Οι τεχνικές σχεδιασμού επικεντρώνονται στη μείωση της κατανάλωσης ισχύος μέσω διαφόρων μεθόδων, όπως η προσεκτική τοποθέτησή της πηγής ενέργειας [power-aware placement], η απενεργοποίηση ρολογιού [Clock Gating] και η κλίμακα της τάσης [Voltage Scaling] [16].

4. Σχεδιασμός για κατασκευαστή [Design for Manufacturability [DFM]]: Η φυσική σχεδίαση στοχεύει στο να διασφαλίσει ότι το κατασκευασμένο ολοκληρωμένο κύκλωμα πληροί τις απαιτήσεις της διαδικασίας κατασκευής. Ειδικότερα, οι παράγοντες του DFM περιλαμβάνουν την εφαρμογή των κανόνων σχεδιασμού, την αποφυγή δυνητικών προβλημάτων κατασκευής και την ενίσχυση της απόδοσης [yield] [15].

(3.2 Προκλήσεις που Αντιμετωπίζονται στην Φυσική Σχεδίαση)

Το στάδιο της φυσικής σχεδίασης παρουσιάζει μια σειρά προκλήσεων που οι σχεδιαστές πρέπει να ξεπεράσουν για να επιτύχουν ένα ολοκληρωμένο κύκλωμα (IC). Αυτές οι προκλήσεις περιλαμβάνουν την επίτευξη χρονισμού, τη διαχείριση της αυξανόμενης πολυπλοκότητας των σχεδίων IC, τη διασφάλιση της συμμόρφωσης με τους κανόνες σχεδιασμού και τους περιορισμούς. Αυτοί οι περιορισμοί τίθενται από την υλοποίηση του σχεδίου, τη διατήρηση της ακεραιότητας του σήματος σε σχεδιασμούς υψηλής ταχύτητας και την εξισορρόπηση πολλαπλών περιορισμών σχεδιασμού, όπως οι απαιτήσεις ισχύος, περιοχής και απόδοσης. Οποσδήποτε, η αντιμετώπιση αυτών των προκλήσεων απαιτεί τη χρήση προηγμένων αλγορίθμων, μεθοδολογιών και εργαλείων για τη βελτιστοποίηση της φυσικής διαδικασίας σχεδιασμού.

1. Επίτευξη χρονισμού [Timing Closure] : Η επίτευξη της χρονικής σύγκλισης αποτελεί μια σημαντική πρόκληση στην φυσική σχεδίαση. Περιλαμβάνει τη διασφάλιση ότι όλες οι διαδρομές σήματος στο κύκλωμα πληρούν τους απαιτούμενους χρονικούς περιορισμούς (timing constraints), λαμβάνοντας υπόψη παράγοντες όπως η απόκλιση του ρολογιού [Clock Skew], χρόνοι ρύθμισης [Clock Setup] και διατήρησης [Hold Times] και οι καθυστερήσεις διασύνδεσης [Interconnect Delays]. [18]

2. Πολυπλοκότητα Σχεδιασμού [Design Complexity] : Η πολυπλοκότητα του σχεδιασμού αποτελεί μια αυξανόμενη πρόκληση, καθώς τα ολοκληρωμένα κυκλώματα γίνονται όλο και πιο περίπλοκα, με εκατομμύρια ή ακόμη και δισεκατομμύρια τρανζίστορ να αποτελούν ένα μόνο τσιπ. Αυτό σημαίνει ότι η διαχείριση μιας τέτοιας πολυπλοκότητας απαιτεί προηγμένους αλγόριθμους, μεθοδολογίες και υπολογιστικούς πόρους για την αντιμετώπιση προκλήσεων στον τομέα της τοποθέτησης [Placement], της δρομολόγησης [Routing] και της βελτιστοποίησης . [19]

3. Συμμόρφωση με τους κανόνες σχεδίασης [Design Rule Compliance] : Η φυσική σχεδίαση πρέπει να τηρεί τους κανόνες σχεδίασης που καθορίζονται από τη διαδικασία κατασκευής. Η εξασφάλιση συμμόρφωσης με αυτούς τους κανόνες κατά την επίτευξη των σχεδιαστικών στόχων, μπορεί να αποτελέσει πρόκληση, ειδικά όταν οι τεχνολογίες κατασκευής συνεχώς εξελίσσονται και οι κανόνες σχεδίασης γίνονται πιο αυστηροί . [20]

4. Ακεραιότητα Σήματος [Signal Integrity] : Η διατήρηση της ακεραιότητας του σήματος είναι κρίσιμη για την αξιόπιστη λειτουργία του κυκλώματος. Ειδικότερα, οι προκλήσεις στον φυσικό σχεδιασμό που σχετίζονται με την ακεραιότητα του σήματος περιλαμβάνουν την ελαχιστοποίηση του θορύβου σήματος και των ηλεκτρομαγνητικών παρεμβολών, ιδιαίτερα σε υψηλής ταχύτητας σχεδιασμούς. [21]

5. Περιορισμούς Σχεδίασης [Design Constraints] : Η φυσική σχεδίαση πρέπει να λαμβάνει υπόψη διάφορους περιορισμούς, συμπεριλαμβανομένων των απαιτήσεων ισχύος, περιοχής και απόδοσης. Βέβαια, η εξισορρόπηση αυτών των περιορισμών και η βελτιστοποίηση του σχεδιασμού εντός των περιορισμών μπορεί να είναι περίπλοκη και γεμάτη προκλήσεις.[22].

Η αντιμετώπιση αυτών των στόχων και προκλήσεων απαιτεί έναν συνδυασμό καινοτόμων τεχνικών σχεδίασης, αλγορίθμων βελτιστοποίησης και προηγμένων εργαλείων EDA για την επίτευξη ενός επιτυχημένου IC.

ΚΕΦΑΛΑΙΟ 4 ISPD

(4.1 Επισκόπηση και Σημασία)

Το Διεθνές Συμπόσιο για την Φυσική Σχεδίαση [International Symposium on Physical Design [ISPD]] είναι ένα προβεβλημένο συνέδριο που αφορά την φυσική σχεδίαση ολοκληρωμένων κυκλωμάτων [ICs]. Λειτουργεί ως πλατφόρμα για ερευνητές και επαγγελματίες της βιομηχανίας για την ανταλλαγή προόδων στη μεθοδολογία, τους αλγορίθμους και τα εργαλεία του φυσικού σχεδιασμού . Το ISPD διαδραματίζει έναν κρίσιμο ρόλο στον προσδιορισμό του πεδίου της φυσικής σχεδίασης και την προώθηση της καινοτομίας.

(4.2 Κύρια Θέματα και Πεδία Έρευνας)

Το ISPD καλύπτει μια ευρεία γκάμα θεμάτων που σχετίζονται με την φυσική σχεδίαση, συμπεριλαμβανομένων του placement, routing, clock-tree synthesis, floorplanning και τεχνικών βελτιστοποίησης [2]. Το συμπόσιο αντιμετωπίζει τόσο την θεμελιώδη έρευνα όσο και τις πρακτικές εφαρμογές, προωθώντας συζητήσεις για αναδυόμενες τάσεις και προβλήματα που αντιμετωπίζονται.

(4.3 Τα οφέλη της συνεισφοράς)

Το ISPD προσφέρει σημαντικές συνεισφορές στον τομέα της φυσικής σχεδίασης. Δίνει τη δυνατότητα σε ερευνητές να παρουσιάσουν τα τελευταία ευρήματά τους, προάγοντας τη συνεργασία και την κοινοποίηση γνώσεων. Η διαδικασία αξιολόγησης από ομότιμους εγγυάται την ποιότητα και τη σχετικότητα της παρουσιαζόμενης έρευνας, προωθώντας την πρόοδο στη μεθοδολογία της φυσικής σχεδίασης.

Η συμμετοχή στο ISPD προσφέρει αρκετά οφέλη στους συμμετέχοντες. Επιτρέπει σε ερευνητές και επαγγελματίες της βιομηχανίας να παραμείνουν ενημερωμένοι για τις τελευταίες εξελίξεις , να αποκτήσουν εισαγωγή σε έρευνες υψηλής ποιότητας και να ανιχνεύσουν αναδυόμενες τάσεις που μπορούν να διαμορφώσουν

το μέλλον του τομέα. Επιπλέον, το ISPD διευκολύνει το δικτύωση και τη συνεργασία, δημιουργώντας μια κοινότητα experts του τομέα.

Συνοπτικά, το ISPD είναι ένα διάσημο συνέδριο στην κοινότητα της φυσική σχεδίασης, που καλύπτει βασικά θέματα και ερευνητικούς τομείς. Η συμβολή του στις προόδους και τα οφέλη που προσφέρει στους συμμετέχοντες το καθιστούν μια πολύτιμη πλατφόρμα για ερευνητές και επαγγελματίες του κλάδου.

[29]

ΚΕΦΑΛΑΙΟ 5 Δομή του **Bookshelf Format**

(5.1 Εισαγωγή)

Το Bookshelf format είναι μια τυποποιημένη μέθοδος οργάνωσης δεδομένων σε ερευνητικά έργα VLSI. Έχει ως σκοπό τη διευκόλυνση του σταδίου Placement του σχεδιασμού κυκλωμάτων, επιτρέποντας την αποθήκευση και ανάκτηση πληροφοριών με αποδοτικό τρόπο.

(5.2 Οργάνωση Αρχείων στο Bookshelf Format)

Για την εφαρμογή του Bookshelf format, τα δεδομένα διαίρονται σε έξι διακριτά αρχεία κειμένου, καθένα με ένα συγκεκριμένο ρόλο και περιεχόμενα. Αυτά τα αρχεία είναι:

- | | |
|-----------|---|
| 1. .aux | Καταγραφή των input αρχείων |
| 2. .nodes | Διαστάσεις και Αναφορά Pads για τους Κόμβους του Κυκλώματος |
| 3. .nets | Περιγραφή Συνδέσεων του Κυκλώματος |
| 4. .wts | Ορισμός Βαρών Κόμβων |
| 5. .pl | Πληροφορίες Τοποθέτησης Κόμβων |
| 6. .scl | Ορισμός Οριζόντιων Σειρών |

(5.2.1 AUX)

Το αρχείο ".aux" έχει έναν κρίσιμο ρόλο στο Bookshelf format καθώς λειτουργεί ως καταγραφή των input αρχείων. Καθορίζει το όνομα του format και καταγραφεί τα σχετικά αρχεία που αφορούν το κύκλωμα, συμπεριλαμβανομένων των αρχείων ".nodes," ".nets," ".wts," ".pl" και ".scl."

(5.2.2 NODES)

Το αρχείο ".nodes" παρέχει σημαντικές πληροφορίες για τους κόμβους (components) του κυκλώματος. Αποτελείται από μια ενότητα header και μια ενότητα body. Το header περιλαμβάνει τον συνολικό αριθμό κινητών κελιών (NumNodes) και τον αριθμό των Pad (NumTerminals). Το body περιέχει λεπτομέρειες για κάθε κόμβο, όπως το όνομα του κόμβου, το πλάτος, το ύψος και μια προαιρετική αναφορά στο Pad.

(5.2.3 NETS)

Το αρχείο ".wts" είναι υπεύθυνο για τον καθορισμό των βαρών των κόμβων μέσα στο κύκλωμα. Συσχετίζει το όνομα κάθε κόμβου με το αντίστοιχο βάρος, που αποτελεί ένα κρίσιμο παράμετρο για διάφορους αλγορίθμους που χρησιμοποιούνται στις διαδικασίες τοποθέτησης και βελτιστοποίησης.

(5.2.4 WTS)

Το αρχείο ".wts" είναι υπεύθυνο για τον καθορισμό των βαρών των κόμβων μέσα στο κύκλωμα. Συσχετίζει το όνομα κάθε κόμβου με το αντίστοιχο βάρος, που αποτελεί ένα κρίσιμο παράμετρο για διάφορους αλγορίθμους που χρησιμοποιούνται στις διαδικασίες τοποθέτησης και βελτιστοποίησης.

(5.2.5 PL)

Το αρχείο ".pl" περιέχει πληροφορίες σχετικά με την τοποθέτηση των κόμβων στο chip, συμπεριλαμβανομένης της θέσης τους, τον προσανατολισμό και την αναφοράς Pad. Κάθε καταχώρηση σε αυτό το αρχείο περιλαμβάνει το όνομα του κόμβου, τις συντεταγμένες x και y στο chip, τον προσανατολισμό του κόμβου και μια προαιρετική λέξη-κλειδί "FIXED" εάν ο κόμβος είναι Pad.

(5.2.6 SCL)

Το αρχείο ".scl" καθορίζει τις οριζόντιες σειρές του κυκλώματος. Περιέχει λεπτομέρειες όπως οι συντεταγμένες της σειράς, το ύψος της, το πλάτος του site, η απόσταση μεταξύ των site, ο προσανατολισμός του site, η συμμετρία του site, η αρχή της υποσειράς και ο αριθμός των site. Αυτές οι πληροφορίες βοηθούν στην καθιέρωση της συνολικής δομής και διάταξης του κυκλώματος.

Το Bookshelf format παρέχει μια δομημένη και τυποποιημένη προσέγγιση για τον οργανισμό δεδομένων σε ερευνητικά έργα VLSI, ειδικά για το στάδιο του Placement. Με τον διαχωρισμό των πληροφοριών σε διακριτά αρχεία, όπως τα ".aux," ".nodes," ".nets," ".wts," ".pl" και ".scl," διευκολύνεται η αποτελεσματική επεξεργασία και διαχείριση των δεδομένων, διευκολύνοντας το σχεδιασμό και τη βελτιστοποίηση ολοκληρωμένων κυκλωμάτων.

[30][31][32]

ΚΕΦΑΛΑΙΟ 6

Οπτικοποίηση Αλγορίθμων Τοποθέτησης στην Φυσική Σχεδίαση VLSI

Εισαγωγή

Η φυσική σχεδίαση ολοκληρωμένων κυκλωμάτων [Ics] διαδραματίζει ουσιαστικό ρόλο στην επίτευξη βέλτιστης απόδοσης και λειτουργικότητας του chip. Η οπτικοποίηση των αποτελεσμάτων τοποθέτησης είναι ουσιώδης για την αξιολόγηση της απόδοσης των αλγορίθμων και την ανάλυση για βελτιστοποίηση. την απόκτηση γνώσεων για περαιτέρω βελτιστοποίηση. Αυτή η ενότητα εξετάζει τη σημασία των εργαλείων οπτικοποίησης, Δίνοντας μια αφορά στο Gnuplot , στο Wevian [Web-based Placement Visualizer / Analyzer] και την καινούργια προσθήκη σε αυτό WeVisAP [Web-based Visualized Animated Placement] .

(6.1 GNUPLOT)

(6.1.1 Οπτικοποίηση για την Αξιολόγηση και Ανάλυση της Τοποθέτησης)

Το Gnuplot προσφέρει ευελιξία στη δημιουργία διάφορων τύπων γραφημάτων [Plot] παραστάσεων, συμπεριλαμβανομένων διαγραμμάτων διασποράς [Scatter Plots], γραμμών [Line Plots], ιστογραμμάτων [histograms] και τρισδιάστατων γραφημάτων [3D Plots]. Επιτρέπει στους ερευνητές να αξιολογήσουν τους αλγόριθμους τοποθέτησης οπτικοποιώντας μετρήσεις απόδοσης, συγκρίνοντας παραλλαγές αλγορίθμων και αναλύοντας τα χαρακτηριστικά των αποτελεσμάτων τοποθέτησης. Μέσω των οπτικοποιήσεων, οι ερευνητές μπορούν να αποκτήσουν γνώσεις σχετικά με την κατανομή των στοιχείων, να εντοπίσουν πιθανά ζητήματα και να αξιολογήσουν την αποτελεσματικότητα των αλγορίθμων τους.

(6.1.2 Γραφικά Έτοιμα για Δημοσίευση)

Η ικανότητα του Gnuplot να παράγει plots έτοιμα για δημοσίευση είναι εξαιρετικά επωφελής για τους ακαδημαϊκούς ερευνητές. Επιτρέπει την προσαρμογή των χαρακτηριστικών του γραφήματος, όπως γραμματοσειρές, χρώματα και στυλ γραμμής, διασφαλίζοντας ότι οι απεικονίσεις είναι οπτικά ελκυστικές και μεταφέρουν αποτελεσματικά τις απαραίτητες πληροφορίες. Μέσω των οπτικοποιήσεων, οι ερευνητές μπορούν να αποκτήσουν γνώσεις σχετικά με την κατανομή των στοιχείων, να εντοπίσουν πιθανά ζητήματα και να αξιολογήσουν την αποτελεσματικότητα των αλγορίθμων τους.

ΚΕΦΑΛΑΙΟ 7 Wevian [Web-based Placement Visualizer / Analyzer]

(7.1 Επισκόπηση)

Το Wevian είναι μια Web-based λύση οπτικοποίησης και έχει σχεδιαστεί ειδικά για την τοποθέτηση standard cell στο σχεδιασμό VLSI chip. Προσφέρει μια accurate και modern προσέγγιση για την οπτικοποίηση των VLSI chips, παρέχοντας μια φιλική προς τον χρήστη γραφικό περιβάλλον [User Interface [UI]] με διάφορες δυνατότητες αλληλεπίδρασης. Με το Wevian, οι χρήστες μπορούν να αναλύουν και να εξερευνήσουν αποτελεσματικά τα δεδομένα τοποθέτησης, αποκτώντας πολύτιμες γνώσεις για τη διάταξη του σχεδιασμού. Αυτό το εργαλείο υλοποιήθηκε από τους Κωνσταντίνος Μαντός, Νικόλαος Κούγκαλης, Γεώργιος Δεμεσιώτης υπό την επίβλεψη των καθηγητών Αντώνιος Λαδαλιάρης και Οικονόμου Παναγιώτης.[33],[34]

(7.1.1 Κίνητρο για την ανάπτυξη του Wevian)

Η ανάπτυξη του Wevian είχε ως κίνητρο την ανάγκη για ένα αποτελεσματικό εργαλείο οπτικοποίησης στον σχεδιασμό VLSI chip. Οι παραδοσιακές αναφορές με κείμενο και οι στατικές οπτικές αναπαραστάσεις συχνά δυσκολεύουν στη μετάδοση των λεπτομερειών και των σχέσεων εντός μιας τοποθέτησης. Το Wevian στοχεύει να γεφυρώσει αυτό το χάσμα προσφέροντας μια δυναμική εμπειρία οπτικοποίησης, επιτρέποντας στους χρήστες να ανιχνεύουν σφάλματα, να αναγνωρίζουν μοτίβα και να λαμβάνουν τεκμηριωμένες αποφάσεις σχεδιασμού.

(7.1.2 Στόχοι του Wevian)

Οι κύριοι στόχοι του Wevian είναι να βελτιώσει την απεικόνιση της τοποθέτησης standard cell και να βελτιώσει την εμπειρία του χρήστη. Αξιοποιώντας Web-based τεχνολογίες του, το Wevian στοχεύει να παρέχει μια ευέλικτη επιλογή για την οπτικοποίηση των VLSI chips, διευκολύνοντας την

εξερεύνηση εκατομμυρίων αντικειμένων με αποτελεσματικό και διαδραστικό τρόπο. Οι κύριοι στόχοι του Wevian περιλαμβάνουν:

1. **Accurate Visualization** : Το Wevian προσπαθεί να αναπαραστήσει με ακρίβεια την τοποθέτηση standard cells, rows, και fixed nodes. Μέσω μιας οπτικά ελκυστικής και κομψής παρουσίασης, οι χρήστες μπορούν εύκολα να κατανοήσουν τη διάταξη του σχεδιασμού.

2. **Interactivity / User Engagement** : Το Wevian προσφέρει μια σειρά δυνατοτήτων αλληλεπίδρασης, όπως κλικ, sliders και επιλογές επισήμανσης. Αυτά τα διαδραστικά στοιχεία επιτρέπουν στους χρήστες να εξερευνούν και να αναλύουν συγκεκριμένες πτυχές της τοποθέτησης, προσφέροντας μια ευέλικτη εμπειρία.

3. **Ανίχνευση και Ανάλυση Σφαλμάτων** : Το Wevian επικεντρώνεται στη διευκόλυνση της ανίχνευσης σφαλμάτων παρέχοντας χαρακτηριστικά όπως ανίχνευση Overlaps/Overflow και highlighting. Μέσω της οπτικοποίησης των Overlaps/Overflow και της παροχής ολοκληρωμένων αναφορών, το Wevian βοηθά τους χρήστες να αναγνωρίζουν και να επιλύουν θέματα τοποθέτησης αποτελεσματικά.

4. **Αναφορά και Διαχείριση Δεδομένων**: Το Wevian επιτρέπει τη δημιουργία λεπτομερών αναφορών και στατιστικών, επιτρέποντας στους χρήστες να αναλύουν και να παρακολουθούν διάφορες πτυχές της τοποθέτησης. Το εργαλείο υποστηρίζει επίσης την αποθήκευση και ανάκτηση δεδομένων του χρήστη, διασφαλίζοντας εύκολη πρόσβαση σε προηγούμενες εργασίες, αναφορές και αποτελέσματα για μελλοντική αναφορά.

Με την επίτευξη αυτών των στόχων, το Wevian στοχεύει να συνεισφέρει στον τομέα του σχεδιασμού VLSI chip παρέχοντας μια μοντέρνα και αποτελεσματική λύση οπτικοποίησης για την τοποθέτηση standard cell.

(7.2 Σχεδιασμός και Υλοποίηση)

(7.2.1 Τεχνολογίες και γλώσσες που χρησιμοποιούνται)

Η ανάπτυξη του Wevian ενσωματώνει μια σειρά τεχνολογιών και γλωσσών προγραμματισμού για να διασφαλίσει τη λειτουργικότητα και την απόδοσή του. Οι κύριες τεχνολογίες και γλώσσες που χρησιμοποιούνται περιλαμβάνουν:

1. PHP [Hypertext Preprocessor] : Το PHP χρησιμοποιείται για την backend λειτουργικότητα του Wevian. Χειρίζεται εργασίες όπως η ανάλυση αρχείων τοποθέτησης σε μορφή bookshelf, η διαχείριση των πληροφοριών των χρηστών και η δημιουργία αναφορών.
2. MySQL [Structured Query Language] : Η MySQL χρησιμοποιείται ως το σύστημα διαχείρισης βάσης δεδομένων για την αποθήκευση πληροφοριών των χρηστών και τη διευκόλυνση της ανάκτησης και αποθήκευσης δεδομένων
3. HTML [HyperText Markup Language] και CSS [Cascading Style Sheets] : Τα HTML και CSS χρησιμοποιούνται για τη σχεδίαση και τη διάταξη του frontend, καθορίζοντας τη δομή και την οπτική εμφάνιση του Wevian. Το HTML καθορίζει τη δομή και το γραφικού περιβάλλοντος, ενώ το CSS καθορίζει το στυλ και τη διάταξή του.
4. JavaScript: Η JavaScript είναι η κύρια γλώσσα προγραμματισμού που χρησιμοποιείται στο Wevian, υπεύθυνη για την υλοποίηση της βασικής λειτουργικότητας και διαδραστικότητας. Επιτρέπει δυναμικές ενημερώσεις και ενημερώσεις σε πραγματικό χρόνο, βελτιώνει την εμπειρία του χρήστη και ενσωματώνει διάφορες τεχνικές οπτικοποίησης.

(7.2.2 Αρχιτεκτονική Συστήματος και τα Στοιχεία αυτού)

Το Wevian βασίζεται σε μια καλά καθορισμένη αρχιτεκτονική συστήματος, η οποία περιλαμβάνει διάφορα στοιχεία που λειτουργούν αρμονικά. Η αρχιτεκτονική του συστήματος περιλαμβάνει τα στοιχεία backend και frontend, εξασφαλίζοντας αποτελεσματική ροή δεδομένων και αλληλεπίδρασης με τον χρήστη. Τα βασικά στοιχεία της αρχιτεκτονικής συστήματος του Wevian περιλαμβάνουν:

1. **Backend Server :** Ο Server φιλοξενεί PHP scripts που είναι υπεύθυνα για την ανάλυση αρχείων τοποθέτησης, τη διαχείριση δεδομένων χρήστη και τη δημιουργία αναφορών. Αλληλεπιδρά με το frontend και τη βάση δεδομένων MySQL για την ανάκτηση και αποθήκευση πληροφοριών.
2. **Frontend Interface:** Το Frontend Interface που έχει δημιουργηθεί με HTML, CSS και JavaScript, παρέχει το τμήμα του Wevian που απευθύνεται στον χρήστη. Δίνει τη δυνατότητα στους χρήστες να αλληλεπιδρούν με την οπτικοποίηση τοποθέτησης, να έχουν πρόσβαση σε διάφορες λειτουργίες και να προβάλλουν αναφορές και στατιστικά στοιχεία που δημιουργούνται.
3. **Βάση δεδομένων:** Η βάση δεδομένων MySQL αποθηκεύει πληροφορίες χρήστη, συμπεριλαμβανομένων των στοιχείων του λογαριασμού και των δεδομένων τοποθέτησης. Επιτρέπει την αποτελεσματική ανάκτηση και αποθήκευση δεδομένων, διασφαλίζοντας τη διαθεσιμότητα των πληροφοριών που αφορούν τον χρήστη και των προηγούμενων εργασιών.

(7.2.3 Backend Functionality [PHP , MySQL])

Το Wevian χρησιμοποιεί το PHP για την Backend λειτουργικότητα, διευκολύνοντας την αποτελεσματική επεξεργασία και διαχείριση των δεδομένων. Το PHP είναι υπεύθυνο για την ανάλυση των αρχείων τοποθέτησης σε μορφή bookshelf, την εξαγωγή σχετικών πληροφοριών και την αποθήκευσή τους στη βάση δεδομένων MySQL. Επίσης, χειρίζεται την εγγραφή και την είσοδο των χρηστών και τη δημιουργία αναφορών και στατιστικών. Το PHP αλληλεπιδρά με τη βάση δεδομένων για την ανάκτηση και ενημέρωση των πληροφοριών των χρηστών κατά την ανάγκη.

(7.2.4 Frontend Design [HTML , CSS , JavaScript])

Ο σχεδιασμός του Frontend του Wevian εστιάζει στη δημιουργία μιας διαισθητικής και οπτικά ελκυστικής διεπαφής. Η HTML ορίζει τη δομή και το περιεχόμενο της διεπαφής, ενώ η CSS καθορίζει το στυλ και τη διάταξή της. Η JavaScript διαδραματίζει κρίσιμο ρόλο στην υλοποίηση της βασικής λειτουργικότητας, επιτρέποντας ενημερώσεις σε πραγματικό χρόνο, διαδραστικές λειτουργίες και απρόσκοπτη εμπειρία χρήστη. Ο σχεδιασμός του Frontend εξασφαλίζει ένα περιβάλλον με απόκριση και φιλικό προς τον χρήστη για την οπτικοποίηση της τοποθέτησης του chip VLSI.

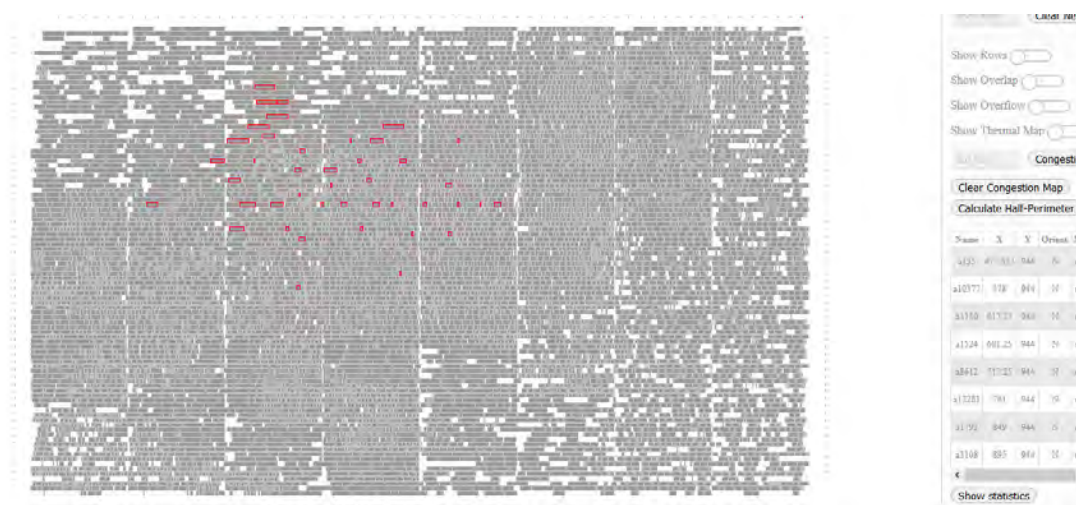
(7.2.5 Ενσωμάτωση Τεχνικών Οπτικοποίησης)

Το Wevian ενσωματώνει διάφορες τεχνικές οπτικοποίησης για τη βελτίωση της αναπαράστασης της τοποθέτησης των standard cells. Αυτές οι τεχνικές περιλαμβάνουν την οπτικοποίηση των κελιών, των γραμμών και των σταθερών κόμβων, καθώς και την παροχή διαδραστικών κουμπιών και δρομέων [Sliders] για την εμφάνιση ή απόκρυψη συγκεκριμένων στοιχείων. Οι τεχνικές οπτικοποίησης υλοποιούνται χρησιμοποιώντας HTML, CSS και JavaScript, επιτρέποντας στους χρήστες να αποκτήσουν μια ολοκληρωμένη κατανόηση της τοποθέτησης και να ανιχνεύσουν μοτίβα ή σφάλματα αποτελεσματικά.

(7.2.6 Χαρακτηριστικά και Λειτουργίες του Wevian)

Το Wevian προσφέρει μια ευρεία γκάμα δυνατοτήτων για τη βελτίωση της οπτικοποίησης και ανάλυσης της τοποθέτησης των standard cells. Ορισμένες αξιοσημείωτες δυνατότητες περιλαμβάνουν:

1. **Επισήμανση κελιού και προβολή πληροφοριών:** Οι χρήστες μπορούν να επισημάνουν συγκεκριμένα κελιά στην τοποθέτηση και να προβάλουν λεπτομερείς πληροφορίες όπως συντεταγμένες, προσανατολισμό και τύπο κίνησης.



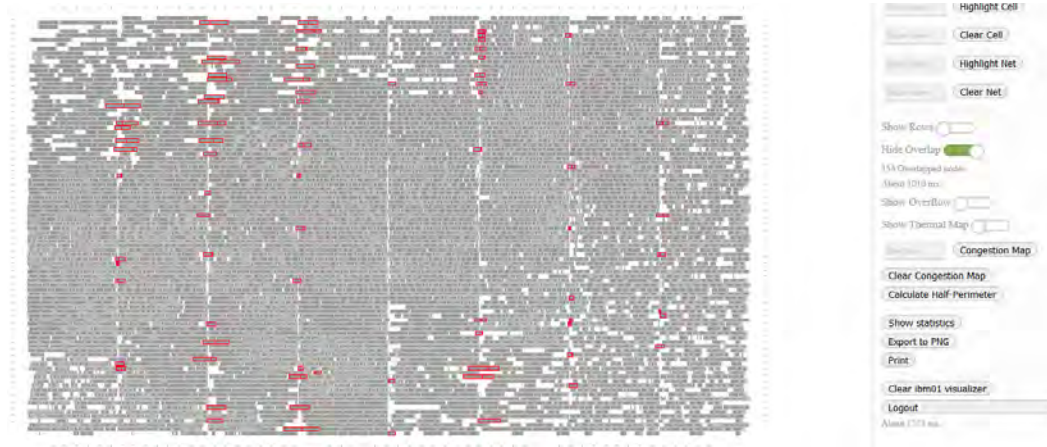
Εικόνα 1: Cell Highlighting / Information Display

2. **Επισήμανση Ομάδων Κελιών Δικτύου [Net Group]:** Το Wevian επιτρέπει στους χρήστες να επισημαίνουν ομάδες κελιών δικτύου προσδιορίζοντας ένα όνομα κελιού, παρέχοντας πληροφορίες για τη συνδεσιμότητα και τις σχέσεις μέσα στον σχεδιασμό.



Εικόνα 2: Net Group Highlighting

3. Ανίχνευση και Επισήμανση Επικαλύψεων [Overlaps]: Το Wevian ανιχνεύει και επισημαίνει τις επικαλύψεις στην τοποθέτηση, επιτρέποντας στους χρήστες να ανιχνεύουν και να επιλύουν γρήγορα προβλήματα τοποθέτησης.



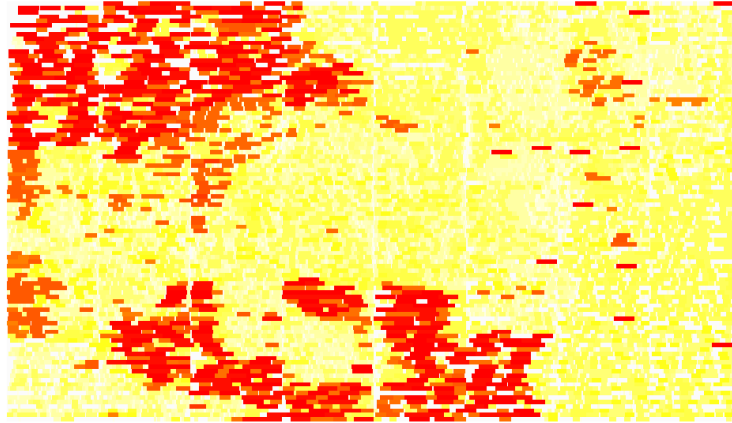
Εικόνα 3: Overlap Detection

4. Ανίχνευση και Επισήμανση Υπερχείλισεων [Overflow] : Το Wevian περιλαμβάνει μια λειτουργία ανίχνευσης υπερχείλισης που προσδιορίζει κελιά που βρίσκονται πέρα από τα όρια του πλαισίου. Αυτά τα κελιά είναι χρωματικά κωδικοποιημένα για να υποδεικνύουν πιθανή υπερχείλιση, βοηθώντας τους χρήστες να διασφαλίσουν τη σωστή τοποθέτηση εντός των ορίων του chip.



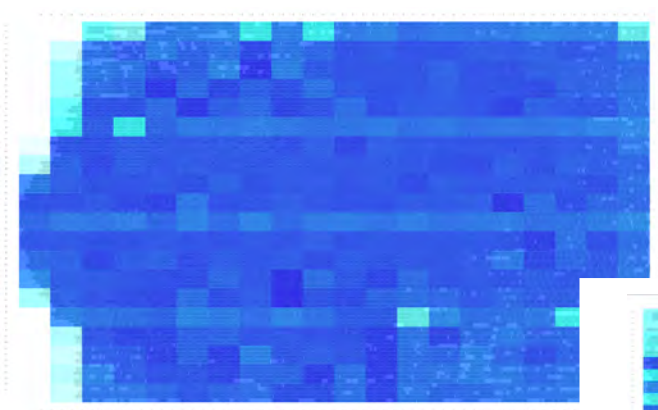
Εικόνα 4: Overflow Detection

5. Θερμικός Χάρτης [Thermal Map] : Το Wenvian δημιουργεί ένα θερμικό χάρτη για την οπτικοποίηση της κατανομής θερμοκρασίας στο benchmark. Αυτή η λειτουργία βοηθά τους χρήστες να εντοπίζουν hotspots και περιοχές με υψηλές θερμοκρασίες, επιτρέποντας την αποτελεσματική διαχείριση της θερμικής κατάστασης.

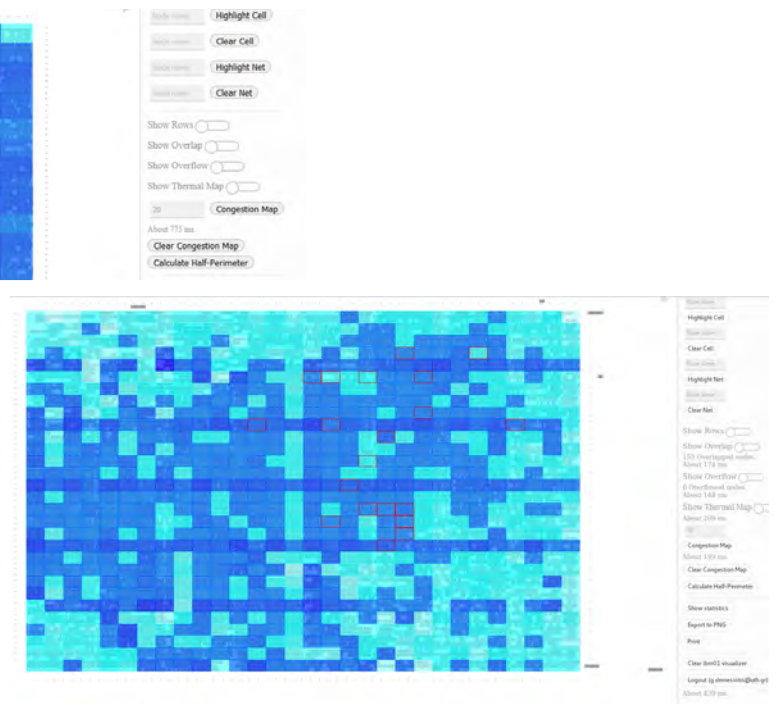


Εικόνα 6: Thermal Map

6. Χάρτης Συμφόρησης [Congestion Map]: Το Wenvian παρέχει έναν χάρτη συμφόρησης που διαιρεί την περιοχή του benchmark σε κουτιά και υπολογίζει την πυκνότητα συμφόρησης κάθε κουτιού. Η πυκνότητα αντιπροσωπεύεται από διαφορετικά χρώματα, επιτρέποντας στους χρήστες να αναγνωρίζουν περιοχές με υψηλή συμφόρηση και να βελτιστοποιούν την τοποθέτηση για βελτιωμένη δρομολόγηση και απόδοση.



Εικόνα 5: Congestion Map 20x20



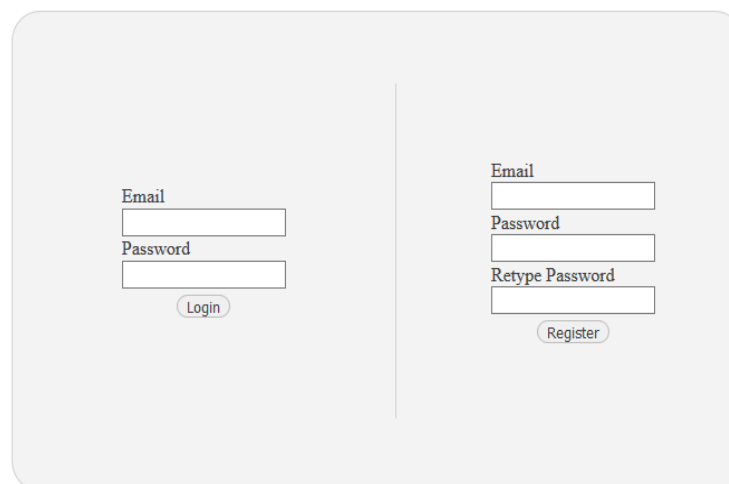
Εικόνα 7: Congestion Map 30x30

Αυτά τα χαρακτηριστικά και οι λειτουργίες συντελούν συλλογικά σε μια περιεκτική και διαδραστική εμπειρία οπτικοποίησης, επιτρέποντας στους χρήστες να αναλύουν και να κατανοήσουν αποτελεσματικά την τοποθέτηση των standard cells. Με την προσθήκη της ανίχνευσης υπερχείλισης, του θερμικού χάρτη και του χάρτη συμφόρησης, το Wevian προσφέρει επεκτεινόμενες δυνατότητες για την ανίχνευση σφαλμάτων, τη θερμική ανάλυση και τη βελτιστοποίηση της συμφόρησης στον σχεδιασμό chip.

(7.3 Χρήση και ροή εργασίας)

(7.3.1 Διαδικασία εγγραφής και σύνδεσης χρήστη)

Το Wevian περιλαμβάνει ένα σύστημα εγγραφής χρήστη και σύνδεσης για να παρέχει εξατομικευμένη πρόσβαση και διαχείριση δεδομένων. Οι χρήστες μπορούν να δημιουργήσουν έναν λογαριασμό καταχωρώντας τα διαπιστευτήριά τους, όπως όνομα χρήστη και κωδικό πρόσβασης. Η διαδικασία εγγραφής εξασφαλίζει ασφαλή πιστοποίηση και προστασία δεδομένων. Μετά την επιτυχή εγγραφή, οι χρήστες μπορούν να συνδεθούν στο Wevian χρησιμοποιώντας τα διαπιστευτήριά τους, επιτρέποντας πρόσβαση στα αποθηκευμένα έργα ή να προχωρήσουν στην οπτικοποίηση ενός καινούργιου benchmark.

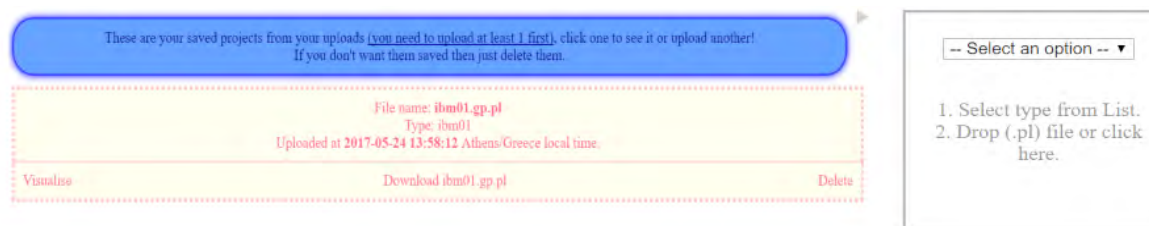


Sign In	Sign Up
Email 	Email
Password 	Password
Login	Retype Password
	Register

Εικόνα 8: Sign In/Up

(7.3.2 Μεταφόρτωση και Ανάλυση Αρχείων Τοποθέτησης)

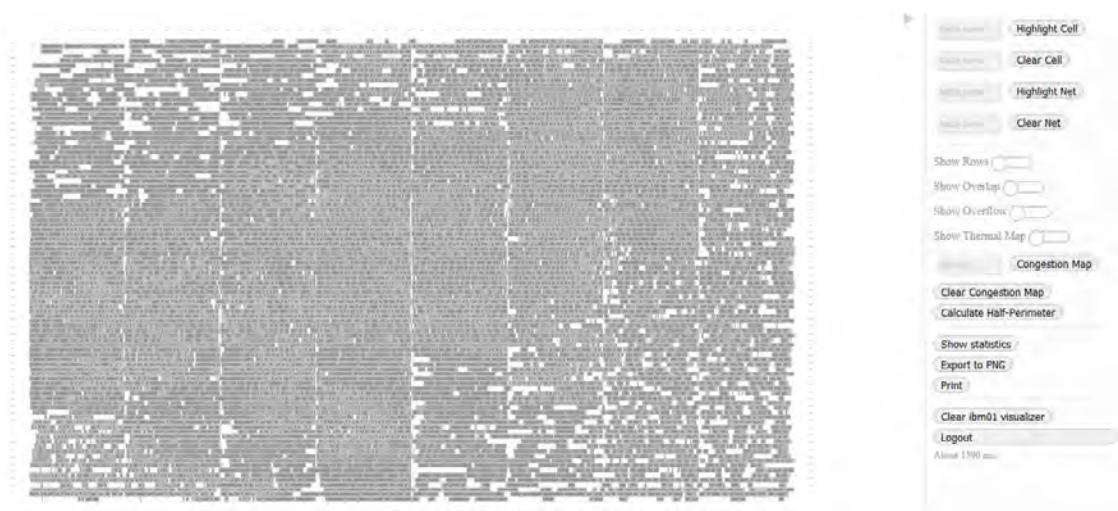
Το Wenvian επιτρέπει στους χρήστες να ανεβάζουν αρχεία τοποθέτησης σε μορφή bookshelf, που περιέχουν τις συντεταγμένες κινητών και σταθερών κόμβων μέσα στον σχεδιασμό VLSI chip. Τα ανεβασμένα αρχεία αναλύονται από το σύστημα, εξάγοντας τα σχετικά δεδομένα τοποθέτησης. Το Wenvian επεξεργάζεται και αποθηκεύει αποτελεσματικά αυτά τα δεδομένα για περαιτέρω οπτικοποίηση και ανάλυση.



Εικόνα 9: Wenvian Main Screen

(7.3.3 Οπτικοποίηση του Σχεδιασμού)

Μόλις μεταφορτωθεί και αναλυθεί το αρχείο τοποθέτησης, το Wenvian παρουσιάζει μια σαφή και κομψή απεικόνιση του σχεδιασμού του benchmark. Η οπτικοποίηση περιλαμβάνει κελιά, σειρές και σταθερούς κόμβους, που αντιπροσωπεύουν με ακρίβεια τη διάταξη. Οι χρήστες μπορούν να εξερευνήσουν και να πλοηγηθούν στην τοποθέτηση για να αποκτήσουν πληροφορίες σχετικά με τη δομή του σχεδιασμού.



Εικόνα 10: Visualization

(7.3.4 Διαδραστικές επιλογές και στοιχεία ελέγχου)

Το Wenvian παρέχει μια σειρά από διαδραστικές επιλογές και στοιχεία ελέγχου για τη βελτίωση της εμπειρίας του χρήστη και τη διευκόλυνση της ανάλυσης. . Οι χρήστες μπορούν να αλληλεπιδρούν με την οπτικοποίηση χρησιμοποιώντας διάφορους ελέγχους, όπως sliders και κουμπιά. Για παράδειγμα, το κουμπί "Show Rows" επιτρέπει στους χρήστες να εναλλάσσουν την ορατότητα των γραμμών που προέρχονται από ένα προκαθορισμένο αρχείο .scl. Αυτά τα διαδραστικά χαρακτηριστικά επιτρέπουν στους χρήστες να προσαρμόσουν την οπτικοποίηση σύμφωνα με τις συγκεκριμένες απαιτήσεις τους και να επικεντρωθούν στα σχετικά στοιχεία της τοποθέτησης.

The screenshot displays a user interface with several sections of controls:

- Cell Controls:** Includes input fields for 'a0' and 'Node name', and buttons for 'Highlight Cell', 'Clear Cell', 'Highlight Net', and 'Clear Net'.
- Node Information:** A text area showing 'n1 is connected with 0 other nodes. About 1232 ms.'
- Display Options:** Four toggle switches for 'Show Rows', 'Show Overlap', 'Show Overflow', and 'Show Thermal Map'.
- Map Controls:** Includes a 'Box Num.' input field, a 'Congestion Map' button, and buttons for 'Clear Congestion Map' and 'Calculate Half-Perimeter'.
- Data Table:** A table with 6 columns: Name, X, Y, Orient., Movetype, and x. It contains one row with data for 'a0'.
- Utility Buttons:** A group of buttons including 'Show statistics', 'Export to PNG', 'Print', and 'Clear ibm01 visualizer'.

Name	X	Y	Orient.	Movetype	x
a0	623	228	N	undefined	Delete r

Εικόνα 11: User Options

(7.3.5 Δημιουργία Αναφορών και Στατιστικών)

Το Wevian προσφέρει τη δυνατότητα δημιουργίας λεπτομερών αναφορών και στατιστικών που σχετίζονται με τη σχεδίαση. Οι χρήστες μπορούν να ζητήσουν αναφορές που παρέχουν λεπτομερείς πληροφορίες για το benchmark, συμπεριλαμβανομένων στατιστικών κυκλωμάτων και αναλύσεων. Αυτές οι πληροφορίες μπορούν να αποθηκευτούν σε μορφή PDF/PNG, επιτρέποντας στους χρήστες να ανατρέχουν σε αυτές για μελλοντική αναφορά ή collaboration. Παρακάτω θα παρουσιαστούν οι διάφορες αναφορές/στατιστικά και λίγα λόγια το καθένα.

(7.3.5.1 Circuit statistics)

NumNodes	12752	2030	Overlaps
NumTerminals	246	0	Overflows
NumRows	96	0	Min. Temperature
NumNets	14111	98	Max. Temperature
NumPins	50566	4%	Min Congestion
Half-Perimeter	105355735	98%	Max Congestion

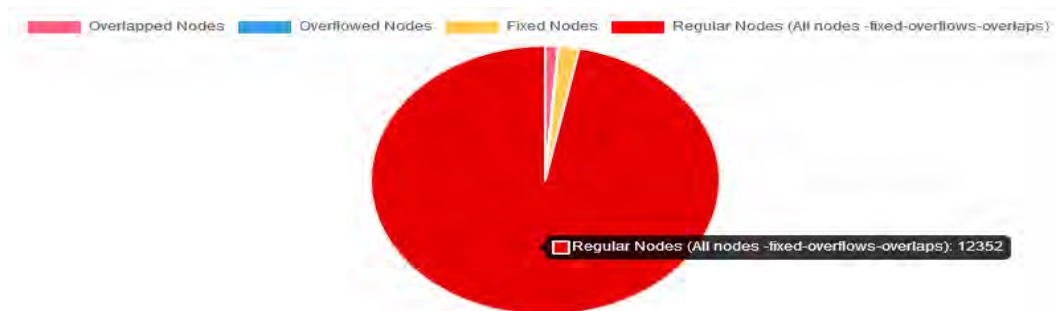
Εικόνα 12: Circuit statistics

Ο πίνακας αποτελείται από δύο στήλες, η κάθε μία περιέχει διάφορες μετρικές που σχετίζονται με ένα τοιπ.

Στην πρώτη στήλη βρίσκουμε μετρικές που σχετίζονται με τα κελιά στο τοιπ. Αυτές περιλαμβάνουν το NumNodes, που αναπαριστά τον συνολικό αριθμό των κελιών, συμπεριλαμβανομένων κινητών και ακίνητων, και το NumTerminals, που υποδηλώνει τον αριθμό των ακινήτων κελιών, όπως τα pads που περιβάλλουν τα κινητά κελιά. Το NumRows αναπαριστά τον αριθμό των σειρών που χρησιμοποιούνται για την οργάνωση των κελιών, ενώ το NumNets υποδηλώνει τον συνολικό αριθμό των ομάδων δικτύων, με κάθε ομάδα να αναπαριστά μια σύνδεση μεταξύ κελιού και άλλων. Το NumPins αναφέρεται στο συνολικό αριθμό των συνδέσεων εντός κάθε δικτύου, και το Half-Perimeter αναπαριστά το συνολικό μήκος της καλωδίωσης που χρησιμοποιείται για αυτές τις συνδέσεις.

Προχωρώντας στη δεύτερη στήλη, περιέχει μετρικές που σχετίζονται με τη διάταξη και τη θερμοκρασία του τσιπ. Τα Overlaps αναπαριστούν τον αριθμό των κελιών που επικαλύπτονται ή καταλαμβάνουν την ίδια θέση. Τα Overflows καταμετρούν τα κελιά που υπερβαίνουν τα όρια του τσιπ. Το Min. Temperature υποδηλώνει τη χαμηλότερη καταγεγραμμένη θερμοκρασία στο θερμικό χάρτη, ενώ το Max. Temperature αναπαριστά την υψηλότερη καταγεγραμμένη θερμοκρασία. Τέλος, το Min. Congestion υποδηλώνει την ελάχιστη κατειλημμένη περιοχή στον χάρτη συμφόρησης, και το Max. Congestion αναπαριστά τη μέγιστη κατειλημμένη περιοχή.

(7.3.5.2 Percentage of different nodes pie chart)

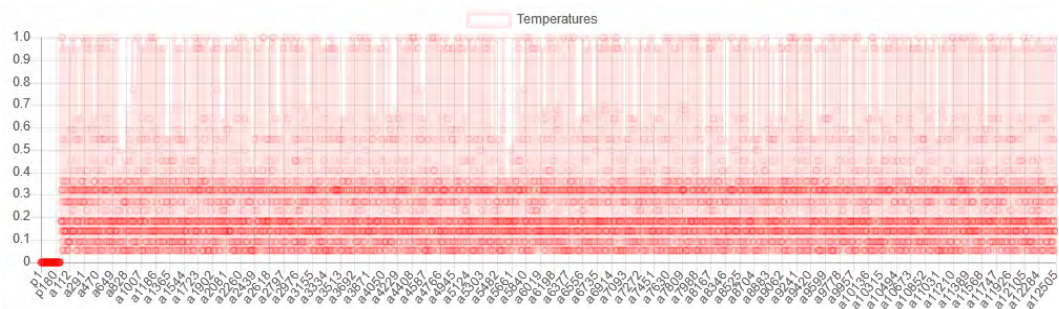


Εικόνα 13: Percentage of different nodes pie chart

Το γράφημα πίτας παρουσιάζει τα σύνολα overlap, overflow, pad και κινητών cell με αναλογικές αναλογίες, δίνοντας μια εποπτική αναπαράσταση του ποσοστού κάθε συνόλου σε σχέση με το συνολικό αριθμό των κυττάρων. Κάθε τμήμα του γραφήματος πίτας έχει ένα μοναδικό χρώμα που το αντιστοιχεί σε ένα από τα σύνολα που αναφέρθηκαν. Αυτό επιτρέπει στον χρήστη να έχει μια οπτική αντίληψη της συμβολής κάθε συνόλου στο συνολικό κύτταρο του τσιπ.

Επιπλέον, το γράφημα πίτας είναι διαδραστικό, παρέχοντας τη δυνατότητα στον χρήστη να κάνει κλικ σε οποιοδήποτε τμήμα του γραφήματος για να λάβει πληροφορίες σχετικά με αυτό. Αυτό μπορεί να γίνει μέσω ενός υπομνήματος που εμφανίζεται κατά την αλληλεπίδραση με το γράφημα πίτας. Με αυτόν τον τρόπο, ο χρήστης μπορεί να ανακτήσει λεπτομερείς πληροφορίες για το επιλεγμένο κομμάτι του γραφήματος, όπως το αντίστοιχο σύνολο κυττάρων και το ποσοστό που αντιπροσωπεύει σε σχέση με το συνολικό αριθμό κυττάρων.

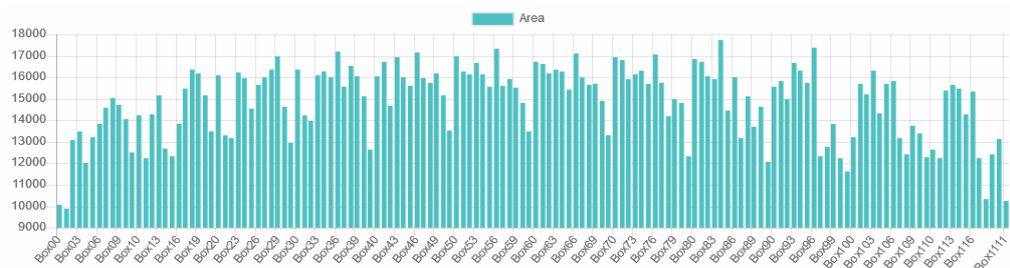
(7.3.5.3 Temperature Graph)



Εικόνα 14: Temperature Graph

Το γράφημα point-style απεικονίζει όλα τα κελιά και τους αναθέτει μια τιμή βάσει θερμοκρασίας από 0 έως 1, με αποτέλεσμα έναν τύπο θερμικού χάρτη που εμφανίζει οπτικά το εύρος θερμοκρασίας. Η διαφορά μεταξύ αυτής της εικόνας και του προηγούμενου αναφερόμενου θερμικού χάρτη [Εικόνα 5] είναι ότι τα κελιά σε αυτόν τον χάρτη ταξινομούνται κατά θερμοκρασία, ενώ τα κελιά στον άλλο χάρτη εμφανίζονται με βάση την πραγματική τους θέση στο chip.

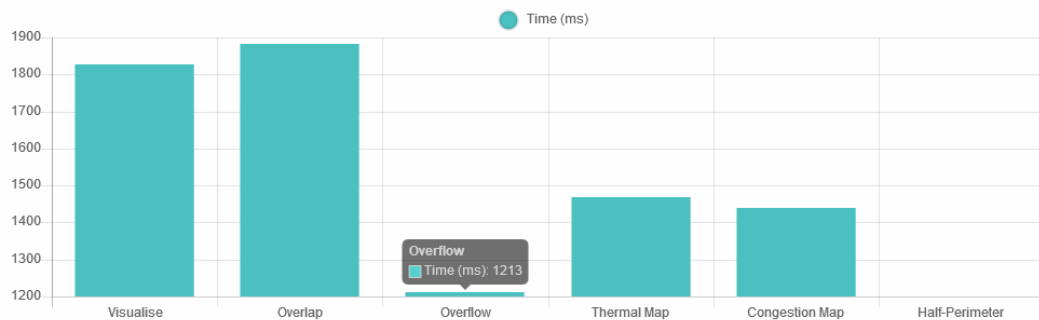
(7.3.5.4 Congestion Map Graph)



Εικόνα 15: Congestion Map Graph

Το γράφημα point-style απεικονίζει τα κουτιά που προέκυψαν από τον χάρτη συμφόρησης με βάση τις περιοχές τους. Η διαφορά μεταξύ αυτού και του προηγούμενου χάρτη συμφόρησης είναι ότι σε αυτήν την περίπτωση τα κουτιά εμφανίζονται ως γραμμές, κάνοντας τον μέσο όρο της περιοχή πιο ξεκάθαρο.

(7.3.5.5 Runtime of different modules bar chart)

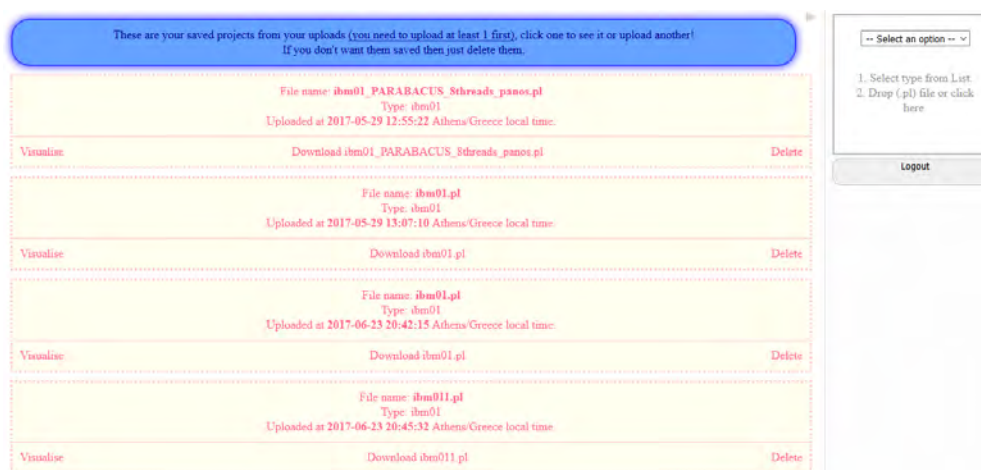


Εικόνα 16: Runtime of different modules bar chart (ms)

Ο γράφος με τις ράβδους εμφανίζει τους χρόνους εκτέλεσης διάφορων λειτουργιών, συμπεριλαμβανομένων της visualize, overlap, overflow, thermal map, congestion map και half perimeter, που αναπαρίστανται σε milliseconds [ms]. Κάθε ράβδος αντιπροσωπεύει τον χρόνο που απαιτείται από κάθε λειτουργία για να ολοκληρώσει την εκτέλεσή της. Αυτός ο γράφος παρέχει μια καθαρή οπτική αναπαράσταση των συγκριτικών χρόνων εκτέλεσης, επιτρέποντας την εύκολη αναγνώριση των λειτουργιών που χρειάστηκαν περισσότερο ή λιγότερο χρόνο για να ολοκληρώσουν τις αρμοδιότητές τους. Η ανάλυση των ράβδων μπορεί να παρέχει στοιχεία για την απόδοση και την αποτελεσματικότητα αυτών των λειτουργιών στο σύστημα.

(7.3.6 Αποθήκευση και Ανάκτηση Δεδομένων Χρήστη)

Το Wevian περιλαμβάνει λειτουργία για την αποθήκευση και ανάκτηση των δεδομένων χρήστη για να διευκολύνει τη συνέχεια και την ευκολία. Οι χρήστες μπορούν να αποθηκεύουν τα έργα τους, συμπεριλαμβανομένων των ανεβασμένων αρχείων τοποθέτησης, των ρυθμίσεων οπτικοποίησης και των δημιουργηθέντων αναφορών, μέσα στο σύστημα. Αυτό επιτρέπει στους χρήστες να ανακτούν και να συνεχίσουν την εργασία τους σε μεταγενέστερο χρόνο, εξασφαλίζοντας αδιάλειπτη ροή εργασίας και μειώνοντας τον κίνδυνο απώλειας δεδομένων.



Εικόνα 17: Wevian Main Screen 2

(7.3.7 Συμπέρασμα)

Το Wevian παρέχει μια ολοκληρωμένη ροή εργασίας και ένα φιλικό προς το χρήστη interface , επιτρέπει στους χρήστες να χρησιμοποιούν αποτελεσματικά το εργαλείο οπτικοποίησης, να μεταφορτώνουν και να αναλύουν αρχεία τοποθέτησης, να αλληλεπιδρούν με την οπτικοποίηση, να δημιουργούν αναφορές και να αποθηκεύουν την εργασία τους για μελλοντική αναφορά και συνεργασία.

ΚΕΦΑΛΑΙΟ 8 WeVisAP [Web-based Visualized Animated Placement]

(8.1 Επισκόπηση)

Ο σκοπός αυτής της ενότητας είναι να εμβαθύνει στο θέμα της οπτικοποίησης της διαδικασίας σταδιακής τοποθέτησης χρησιμοποιώντας το εργαλείο WeVisAP. Στόχος του είναι να τονίσει τη σημασία της κατανόησης της διαδικασίας τοποθέτησης πέρα από το τελικό αποτέλεσμα. Εστιάζοντας στη σταδιακή πρόοδο, αυτή η ενότητα στοχεύει να παρέχει μια ολοκληρωμένη κατανόηση των τεχνικών βελτιστοποίησης που χρησιμοποιούνται και της λήψης αποφάσεων που εμπλέκονται στη διαδικασία τοποθέτησης. Η ενότητα θα διερευνήσει τα κίνητρα πίσω από την ανάπτυξη του WeVisAP και τις δυνατότητές του στην οπτικοποίηση της διαδικασίας σταδιακής τοποθέτησης.

(8.2 Κίνητρο για την ανάπτυξη του WeVisAP)

Η διαδικασία σχεδιασμού περιλαμβάνει συχνά διάφορα στάδια, συμπεριλαμβανομένης της τοποθέτησης, όπου τα εξαρτήματα είναι διατεταγμένα σε ένα τσιπ. Η οπτικοποίηση παίζει κρίσιμο ρόλο στην κατανόηση και τη βελτιστοποίηση της διαδικασίας τοποθέτησης. Ωστόσο, τα παραδοσιακά εργαλεία σχεδίασης συχνά επικεντρώνονται αποκλειστικά στο τελικό αποτέλεσμα τοποθέτησης, αφήνοντας τους σχεδιαστές με περιορισμένες γνώσεις σχετικά με τη σταδιακή εξέλιξη και τη λήψη αποφάσεων. Αυτός ο περιορισμός μπορεί να εμποδίσει τον εντοπισμό πιθανών βελτιστοποιήσεων και βελτιώσεων σχεδιασμού. Ως εκ τούτου, υπάρχει μια αυξανόμενη ανάγκη για ένα εργαλείο που όχι μόνο παρουσιάζει την τελική τοποθέτηση, αλλά παρέχει επίσης μια οπτική αναπαράσταση της διαδικασίας σταδιακής τοποθέτησης. Ένα τέτοιο εργαλείο θα επέτρεπε στους σχεδιαστές να αποκτήσουν μια βαθύτερη κατανόηση των τεχνικών βελτιστοποίησης που χρησιμοποιούνται και θα διευκολύνει τις πιο ενημερωμένες αποφάσεις σχεδιασμού. Το κίνητρο αυτής της εφαρμογής βρίσκονται στην αντιμετώπιση αυτού του κενού και στην εξερεύνηση της οπτικοποίησης της διαδικασίας σταδιακής τοποθέτησης χρησιμοποιώντας το εργαλείο WeVisAP.

(8.2.1 Ανάπτυξη του WeVisAP)

Η διαδικασία ανάπτυξης του WeVisAP, ενός εργαλείου οπτικοποίησης σχεδίασης για την τοποθέτηση VLSI chip . Η ανάπτυξη του Backend χρησιμοποιήθηκε το Flask ως πλαίσιο ενώ η ανάπτυξη του FrontEnd χρησιμοποιήθηκε το Flutter. Αυτή η ενότητα παρέχει μια επισκόπηση των βασικών πτυχών που συμπεριλήφθηκαν στην ανάπτυξη του WeVisAP, τονίζοντας τον σκοπό πίσω από την επιλογή του Flask και του Flutter και συζητώντας τη συμβολή τους στη λειτουργικότητα και την εμπειρία των χρηστών του εργαλείου.

(8.2.2 Ανάπτυξη του Backend)

Το WeVisAP χρησιμοποιεί το Flask ως Framework του Backend, εκμεταλλευόμενο την ευελιξία και τη δυνατότητα της Python ως γλώσσα προγραμματισμού. Το Flask είναι ένα ελαφρύ και ευέλικτο Web Framework που βασίζεται στην Python, κάνοντάς το ιδανική επιλογή για την ανάπτυξη του WeVisAP.

Ένα από τα βασικά πλεονεκτήματα της χρήσης του Flask είναι η απλότητα, η αναγνωσιμότητα και η έκταση της υποστήριξης βιβλιοθηκών της Python. Η καθαρή σύνταξη της Python και η ισχυρή υποστήριξη της κοινότητας την καθιστούν μια αποτελεσματική και δημοφιλή επιλογή για γρήγορη ανάπτυξη και ευκολία συντήρησης. Η διαθεσιμότητα μιας ευρείας γκάμας βιβλιοθηκών και frameworks απλοποιεί τις διάφορες εργασίες που απαιτούνται για την επεξεργασία και δημιουργία οπτικοποιήσεων στο WeVisAP.

Η ελαφρότητα του Flask εξασφαλίζει ελάχιστη επιπρόσθετη πολυπλοκότητα και επιτρέπει μια απλή προσέγγιση για τη δημιουργία διαδικτυακών διεπαφών εφαρμογών [Application Programming Interfaces [APIs]] Παρέχει στους προγραμματιστές τα απαραίτητα εργαλεία και την ευελιξία για τη δημιουργία APIs με ελάχιστο κώδικα. Η απλότητα του Flask διευκολύνει την κατανόηση και την εργασία, προωθώντας την αποτελεσματική ανάπτυξη και μειώνοντας την πολυπλοκότητα.

Επιπλέον, το οικοσύστημα επεκτάσεων του Flask ενισχύει περαιτέρω τη λειτουργικότητά του. Αυτές οι επεκτάσεις προσφέρουν μια ευρεία γκάμα πρόσθετων χαρακτηριστικών και ενσωματώσεων, επιτρέποντας στους προγραμματιστές να επεκτείνουν τις δυνατότητες του WeVisAP όπως απαιτείται. Η διαθεσιμότητα αυτών των επεκτάσεων διευκολύνει τη διαδικασία ανάπτυξης,

επιτρέποντας στους προγραμματιστές να επικεντρωθούν στις βασικές λειτουργίες του WeVisAP χωρίς να ανακαλύπτουν τον τροχό.

Χρησιμοποιώντας το Flask στην ανάπτυξη του Backend , το εργαλείο επωφελείται από την ευελιξία της Python και την απλότητα του πλαισίου του Flask. Αυτός ο συνδυασμός παρέχει μια στερεή βάση για τη δημιουργία των απαραίτητων APIs και τη χειρισμό της λογικής που απαιτείται για τις λειτουργίες του WeVisAP. Εξασφαλίζει την αποτελεσματική επεξεργασία των αρχείων που ανεβάζουν οι χρήστες, τη δημιουργία των οπτικοποιήσεων και την ομαλή ενσωμάτωση με το Framework Flutter στο $\mathbb{S}$, με αποτέλεσμα μια ομαλή και responsive εμπειρία χρήσης.

(8.2.2 Ανάπτυξη του Frontend)

Η ανάπτυξη του Frontend γίνεται με τη χρήση του Framework ανάπτυξης Flutter και της γλώσσας προγραμματισμού Dart. Το Flutter επιλέχθηκε λόγω της δυνατότητάς του να δημιουργεί γρήγορα User Interfaces [UI] με συνοπτικό κώδικα και τη συμβατότητά του με πολλαπλές πλατφόρμες, επιτρέποντας στο εργαλείο να λειτουργεί σε πολλές πλατφόρμες, συμπεριλαμβανομένου του web, του κινητού και του desktop.

Το Flutter προσφέρει μια απλουστευμένη διαδικασία ανάπτυξης χρησιμοποιώντας έναν μόνο κώδικα για όλες τις πλατφόρμες, εξαλείφοντας την ανάγκη για ξεχωριστούς κώδικες για διάφορες πλατφόρμες. Αυτό όχι μόνο εξοικονομεί χρόνο και προσπάθεια στην ανάπτυξη, αλλά εξασφαλίζει ενιαίες εμπειρίες χρήστη σε διάφορες συσκευές.

Ένα από τα κύρια πλεονεκτήματα του Flutter είναι η πλούσια συλλογή έτοιμων επιλογών UI και γραφικών στοιχείων (UI components και widgets). Αυτά τα στοιχεία επιτρέπουν στους προγραμματιστές να δημιουργήσουν UI που είναι ευχάριστα αισθητικά και αλληλεπιδραστικά χωρίς να ξεκινούν από το μηδέν. Η διαθεσιμότητα αυτών των έτοιμων στοιχείων επιταχύνει τη διαδικασία ανάπτυξης και εξασφαλίζει μια ενιαία και περιποιημένη εμφάνιση για το UI του WeVisAP.

Ένα άλλο σημαντικό χαρακτηριστικό του Flutter είναι η δυνατότητα ανανέωσης κατά την ανάπτυξη [hot-reload functionality]. Αυτή η δυνατότητα επιτρέπει στους προγραμματιστές να βλέπουν σε πραγματικό χρόνο τις αλλαγές που έγιναν στον κώδικα χωρίς την ανάγκη για μακροχρόνιες καθυστερήσεις για τη μεταγλώττιση και την ανάπτυξη. Αυτό προωθεί μια επαναληπτική διαδικασία

ανάπτυξης, επιτρέποντας στους προγραμματιστές να πειραματίζονται, να δοκιμάζουν εκ νέου και να βελτιστοποιούν το UI με γρήγορο τρόπο, με αποτέλεσμα ταχύτερα να τα προάγονται τα Prototypes.

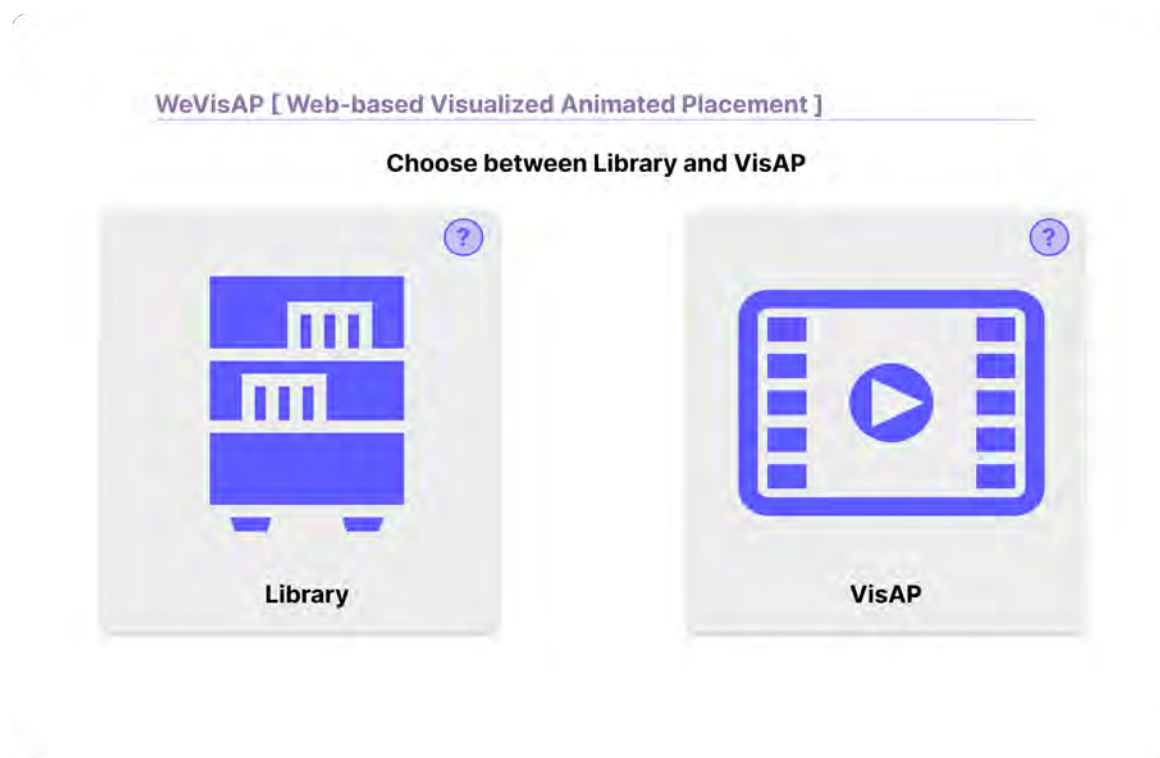
Η απόδοση του Flutter είναι επίσης εντυπωσιακή, καθώς χρησιμοποιεί μια μεταγλωττισμένη γλώσσα προγραμματισμού (Dart) και απεικονίζει τη διεπαφή χρήστη απευθείας στη συσκευή, παρέχοντας ομαλή και responsive εμπειρία χρήσης. Επιπλέον, η κοινότητα του Flutter είναι δυναμική και υποστηρικτική, προσφέροντας μια ευρεία γκάμα πακέτων, εργαλείων και πόρων για τη βελτίωση της αποτελεσματικότητας της ανάπτυξης και την επίλυση προκλήσεων κατά τη διάρκεια της ανάπτυξης.

Εκμεταλλευόμενο τις δυνατότητες του Flutter, το WeVisAP παρέχει στους σχεδιαστές ένα ευχάριστο αισθητικό και διαδραστικό UI για την οπτικοποίηση της σταδιακής διαδικασίας τοποθέτησης. Βεβαιώνει τη συμβατότητα μεταξύ πλατφόρμας, την αποτελεσματική ροή εργασίας ανάπτυξης και μια ποιοτική εμπειρία χρήστη, ενισχύοντας τη χρηστικότητα και την αποτελεσματικότητα του WeVisAP ως εργαλείου οπτικοποίησης σχεδίου.

(8.3 Χρήση και ροή εργασίας)

(8.3.1 Αρχική Σελίδα)

Την πρώτη σελίδα ο χρήστης έχει επιλογή να επιλέξει ανάμεσα στην βιβλιοθήκη και στο VisAP. Πρώτα θα ακολουθήσουμε την πορεία του χρήστη όταν επιλέγετε VisAP.



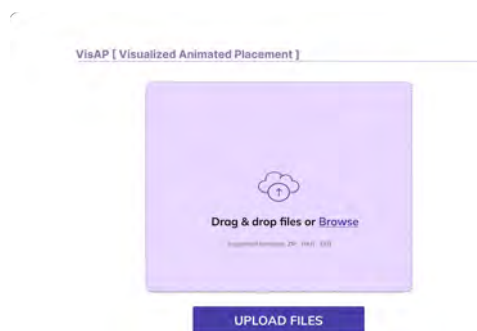
Εικόνα 18: WeVisAp Main Screen

(8.3.1.1 VisAP)

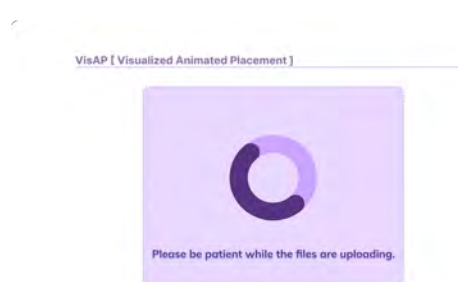
Ο χρήστης ξεκινά τη διαδικασία ανεβάζοντας αρχεία με την επιλογή της επιλογής "VisAP" στο WeVisAP. Το WeVisAP παρέχει στον χρήστη τη δυνατότητα να ανεβάσει ένα συμπιεσμένο αρχείο είτε με τον μεταφορά-και-απόθεση (drag and drop) είτε χρησιμοποιώντας τον προβλεπόμενο διαχειριστή αρχείων.

Ο χρήστης επιλέγει το επιθυμητό συμπιεσμένο αρχείο που περιέχει τα δεδομένα της διαδικασίας τοποθέτησης.

Μόλις το αρχείο ανέβει, το WeVisAP το αποστέλλει στον Backend για επεξεργασία.



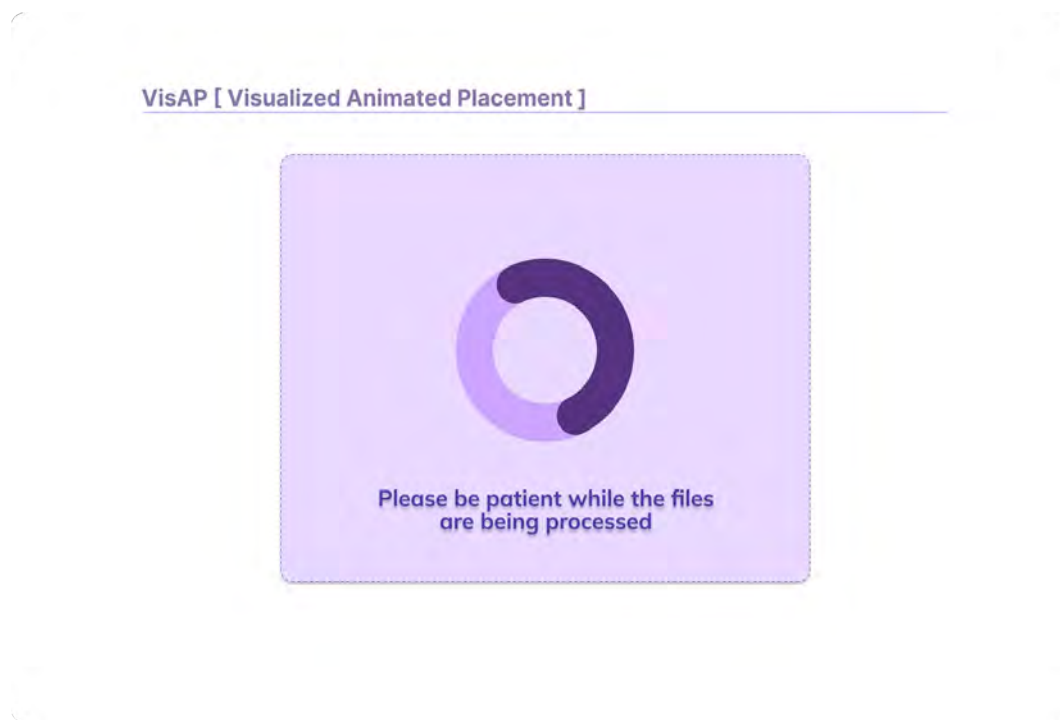
Εικόνα 19: VisAp Upload Screen



Εικόνα 20: VisAp Uploading Screen

(8.3.1.2 Δημιουργία Οπτικοποίησης)

Το Backend του WeVisAP επεξεργάζεται το ανεβασμένο αρχείο, εξάγοντας τις απαραίτητες πληροφορίες. Το back-end δημιουργεί ένα βίντεο που παρουσιάζει τη διαδικασία σταδιακής τοποθέτησης με βάση τα εξαγόμενα δεδομένα. Το βίντεο δημιουργείται σε μορφή που επιτρέπει την ομαλή αναπαραγωγή και διαδραστική εξερεύνηση.

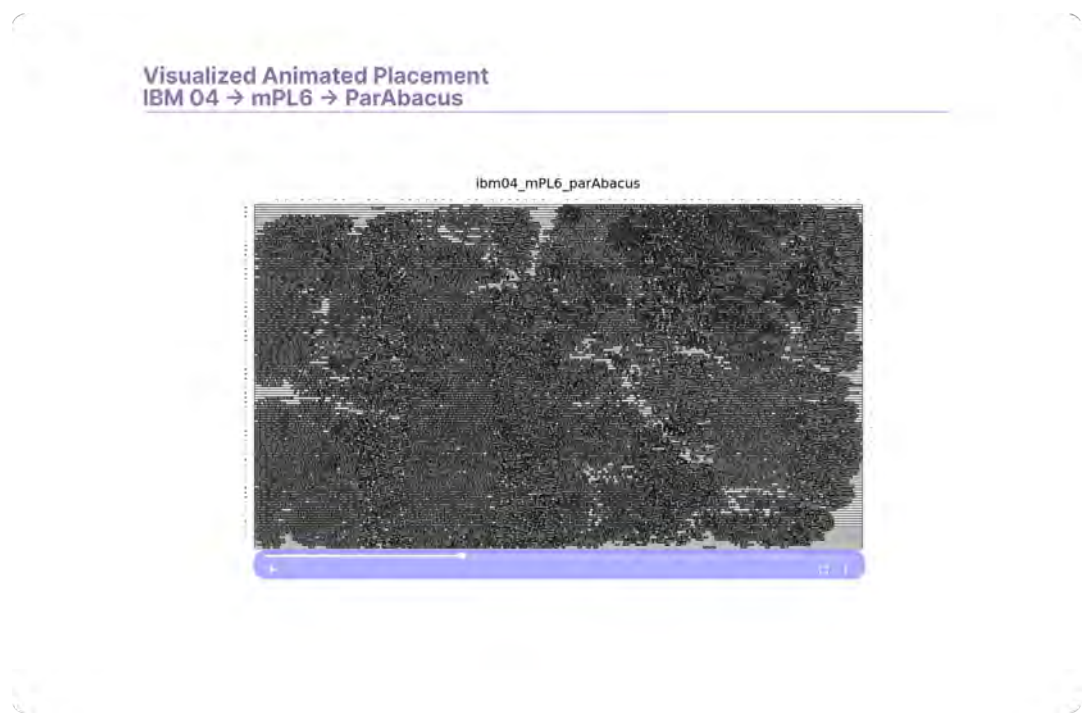


Εικόνα 21: VisAp Proccesing Screen

(8.3.1.3

Προβολή και αλληλεπίδραση με τις οπτικοποιήσεις)

Το WeVisAP παρουσιάζει στον χρήστη το βίντεο στη διεπαφή του. Ο χρήστης μπορεί να αναπαράγει, να παύει, να γυρίσει πίσω και να περιηγηθεί στο χρονοδιάγραμμα του βίντεο για να παρατηρήσει τη σταδιακή διαδικασία τοποθέτησης. Παρέχονται διαδραστικοί έλεγχοι και χαρακτηριστικά για τη βελτίωση της εξερεύνησης και της κατανόησης της διαδικασίας τοποθέτησης από τον χρήστη. Ο χρήστης μπορεί επίσης να προσαρμόσει την ταχύτητα αναπαραγωγής.



Εικόνα 22: VisAp Final Screen Player

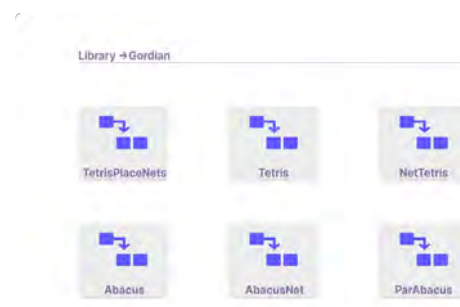
(8.3.1.2 Επιλογή Library για προηγούμενες μεταφορτώσεις)

Πέραν της επιλογής "VisAP", το WeVisAP προσφέρει μια επιλογή Library για να προβάλλει προηγούμενα μεταφορτωμένα αρχεία.

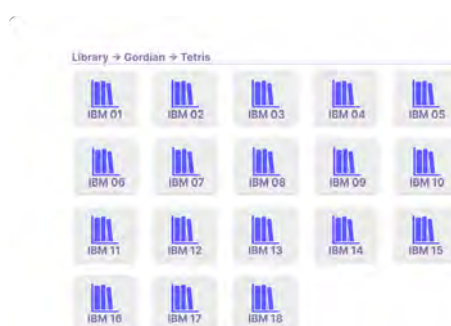
Ο χρήστης επιλέγει την επιλογή Library, η οποία εμφανίζει μια λίστα αρχείων. Ο χρήστης μπορεί να επιλέξει ένα συγκεκριμένο αρχείο από τη βιβλιοθήκη για να προβάλλει το συνοδευτικό βίντεο οπτικοποίησης και να εξερευνήσει τη σταδιακή διαδικασία τοποθέτησης για αυτό το συγκεκριμένο αρχείο. Ας πάμε να δούμε ροή που θα ακολουθήσει ο χρήστης για να επιλέξει να δει για το αρχείο IBM01 Gordian over Tetris.



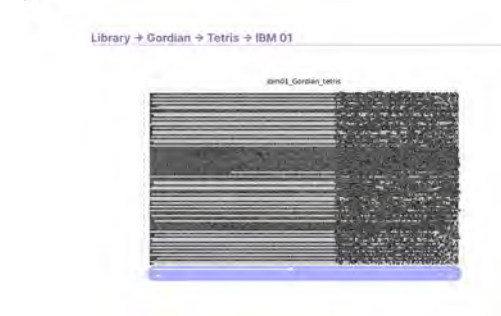
Εικόνα 24: WeVisAP Library



Εικόνα 23: WeVisAP Library -> Gordian



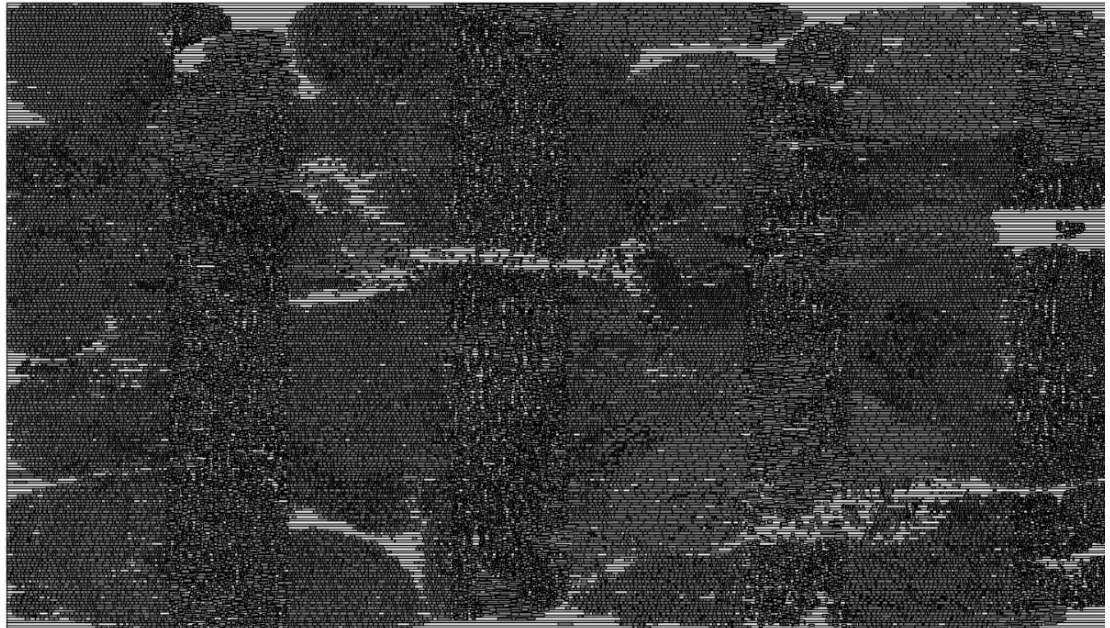
Εικόνα 26: WeVisAP Library
-> Gordian - > Tetris



Εικόνα 25: WeVisAP Library -> Gordian -
> Tetris - IBM01

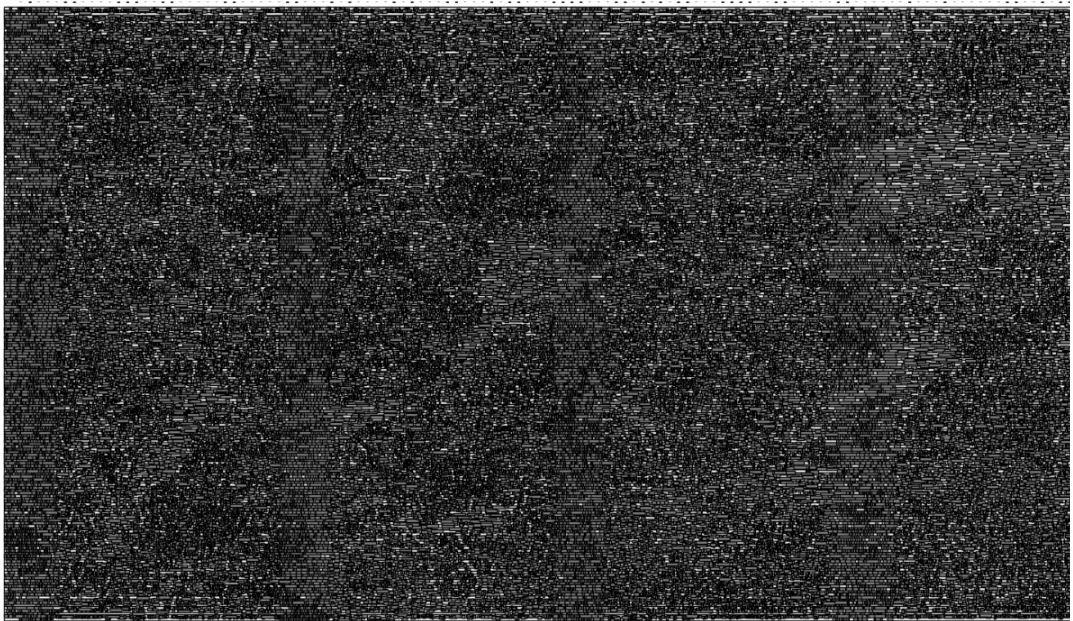
(8.4 Στιγμιότυπα τελικού παραγόμενου)

ibm10_mPL6_parAbacus



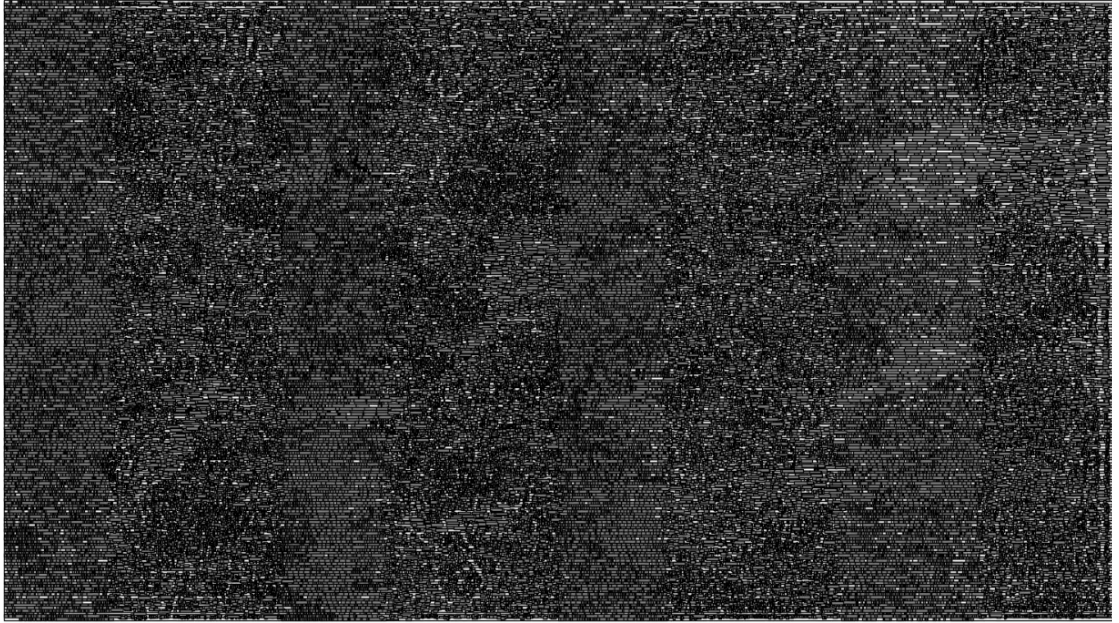
Εικόνα 27: ibm13_NTUplace3_parAbacus_1

ibm13_NTUplace3_parAbacus



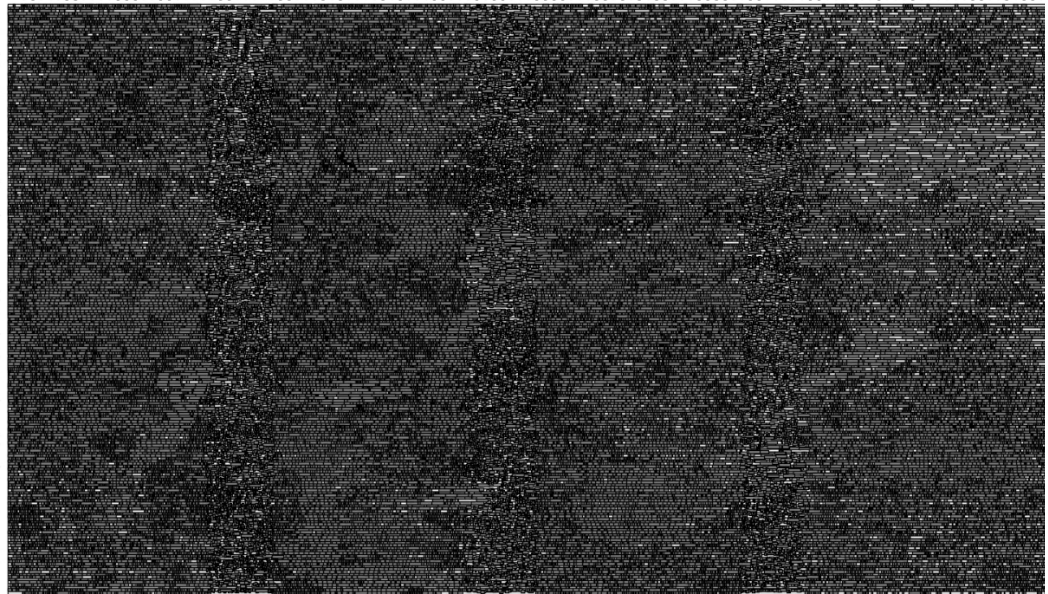
Εικόνα 28: ibm13_NTUplace3_parAbacus_2

ibm13_NTUplace3_parAbacus

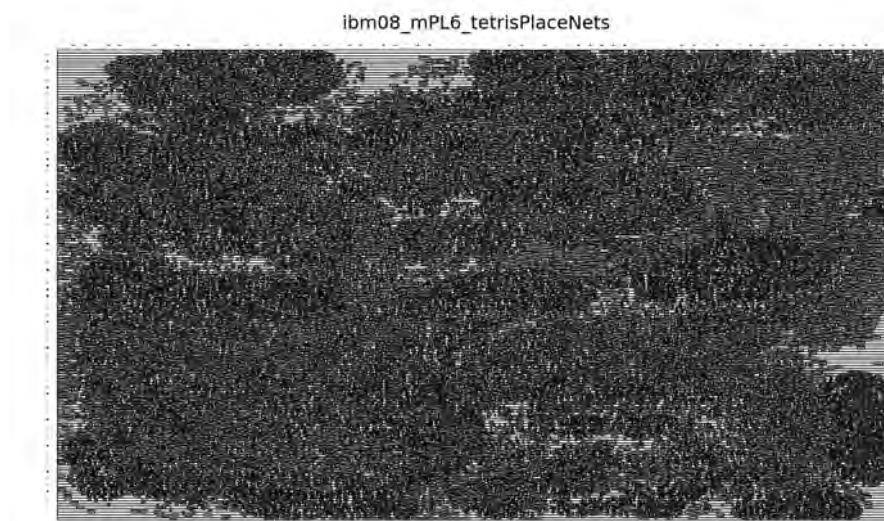


Εικόνα 29: ibm13 NTUplace3 parAbacus 3

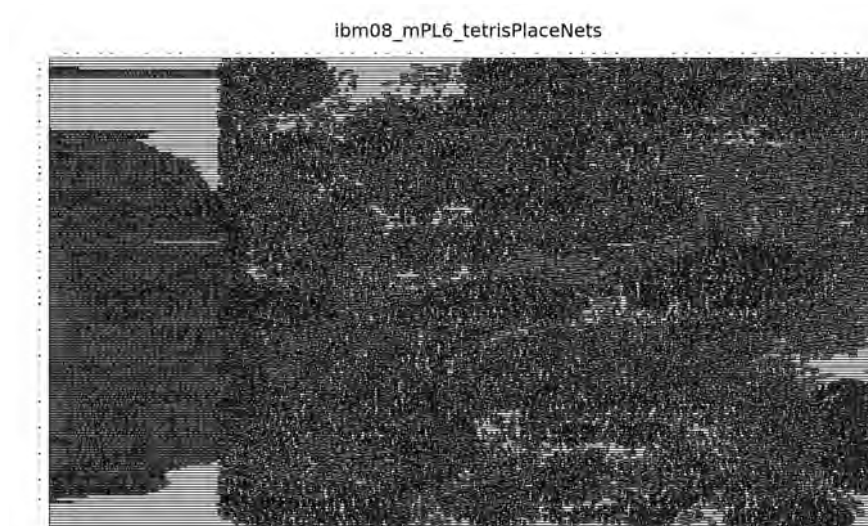
ibm13_NTUplace3_parAbacus



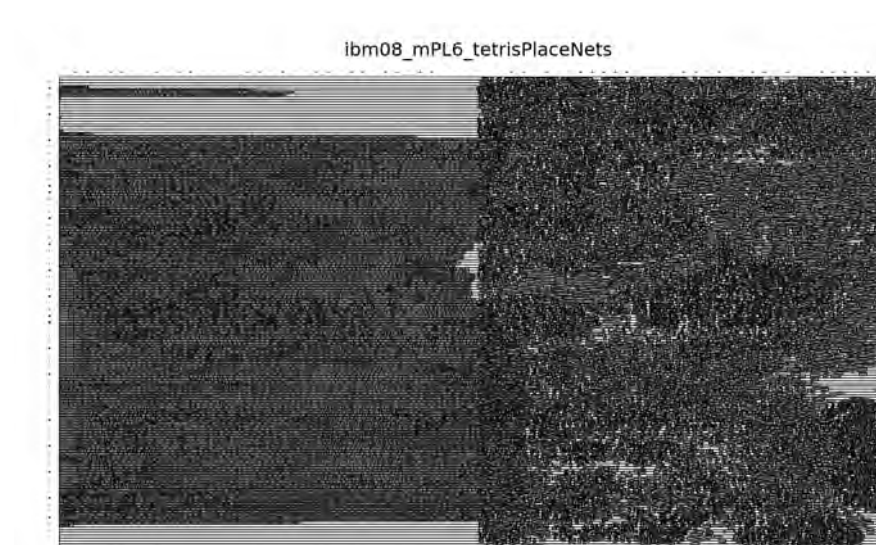
Εικόνα 30: ibm13_NTUplace3_parAbacus_4



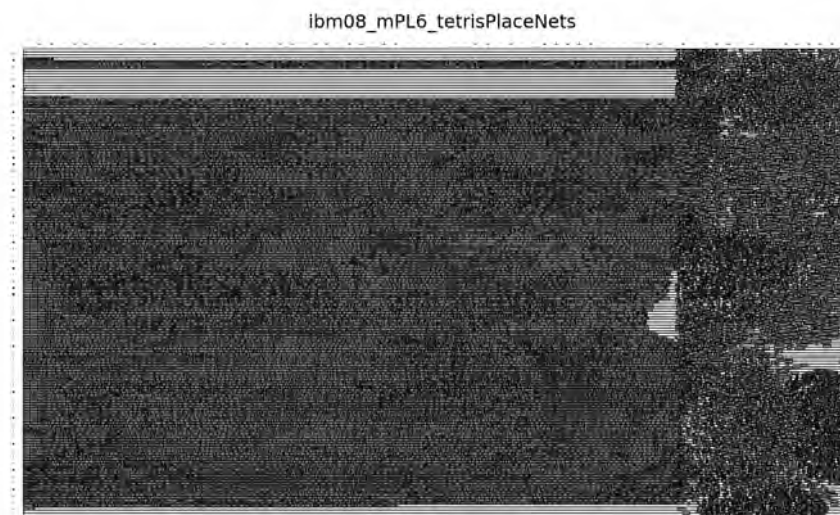
Εικόνα 31: ibm08_mPL6_tetrisPlaceNets_1



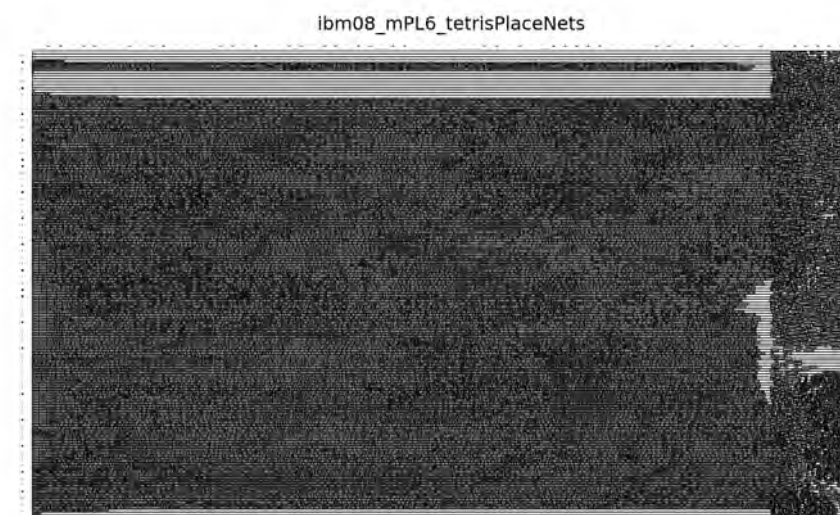
Εικόνα 32: ibm08_mPL6_tetrisPlaceNets_2



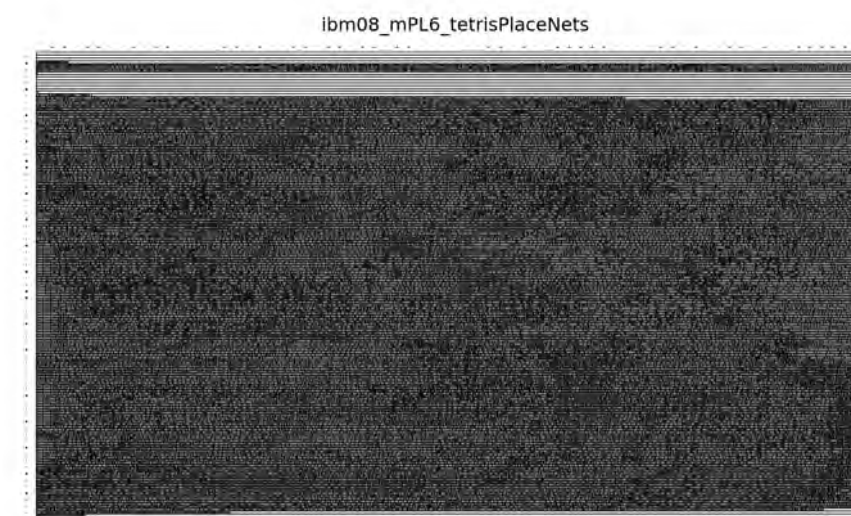
Εικόνα 33: ibm08_mPL6_tetrisPlaceNets_3



Εικόνα 34: ibm08_mPL6_tetrisPlaceNets_4



Εικόνα 35: ibm08_mPL6_tetrisPlaceNets_5



Εικόνα 36: ibm08_mPL6_tetrisPlaceNets_6

(8.5 Μορφή Συμπεσμένου αρχείου για σωστή οπτικοποίηση)

Για την κατάλληλη οπτικοποίηση στο WeVisAP, το συμπεσμένο αρχείο που ανεβαίνει πρέπει να πληροί συγκεκριμένες απαιτήσεις .

Πίνακας 1: Περιεχόμενα csv αρχείου

Συμπεσμένο Αρχείο	
title	
0000.csv	
0001.csv	
0002.csv	
.	
.	
.	
0xyz.csv	

Πίνακας 2: Περιεχόμενα title αρχείου

Αρχείο title
Όνομα αρχείου_ Αλγόριθμος0_Αλγόριθμος1_....

Πίνακας 3: Περιεχόμενα συμπεσμένου αρχείου

Αρχείο CSV				
x	y	width	height	type
1142	1200	16	16	C
16	256	14	16	C
...	...	...	...	...
1582	1542	1	1	P
0	0	1550	16	R

Το αρχείο title πρέπει να είναι μορφή text και να ξεκινά με τον τύπο του κυκλώματος που αναφέρεται ρητά ως πρώτο όριομα, ακολουθούμενο από τη λίστα των χρησιμοποιούμενων αλγορίθμων. Αυτή η δομημένη μορφή διασφαλίζει όχι μόνο την ακριβή οπτικοποίηση της σταδιακής διαδικασίας τοποθέτησης, αλλά επιτρέπει επίσης την εύκολη αρχειοθέτηση των παραγόμενων αρχείων εξόδου στη βιβλιοθήκη, διευκολύνοντας τη μελλοντική αναφορά και ανάλυση.

Επιπλέον, εντός του συμπεσμένου αρχείου, η παρουσία αρχείων CSV είναι κρίσιμη για την αποτελεσματική οπτικοποίηση της σταδιακής διαδικασίας

τοποθέτησης. Αυτά τα αρχεία CSV πρέπει να καταγράφουν στιγμιότυπα της διαδικασίας τοποθέτησης, οργανωμένα με αύξουσα σειρά για να αντιπροσωπεύουν τα στάδια αύξησης της τοποθέτησης. Το περιεχόμενο κάθε αρχείου CSV πρέπει να συμμορφώνεται με μια διεθνώς αποδεκτή μορφή, συμβατή με δημοφιλή εργαλεία οπτικοποίησης, όπως το Gnuplot. Αυτή η καθιερωμένη μορφή επιτρέπει στους χρήστες να οπτικοποιούν τη σταδιακή διαδικασία τοποθέτησης με το εργαλείο WeVisAP αποτελεσματικά.

Ακολουθώντας αυτές τις οδηγίες για την μορφή συμπιεσμένου αρχείου και ενσωματώνοντας σωστά τα αρχεία CSV, οι σχεδιαστές μπορούν να αξιοποιήσουν τις δυνατότητες του WeVisAP για να αποκτήσουν πολύτιμες πληροφορίες σχετικά με τη διαδικασία σταδιακής τοποθέτησης. Η τήρηση της καθορισμένης μορφής εξασφαλίζει ακριβή οπτικοποίηση, δίνοντας τη δυνατότητα στους σχεδιαστές να λαμβάνουν τεκμηριωμένες αποφάσεις και βελτιστοποιήσεις, οδηγώντας τελικά σε βελτιωμένα αποτελέσματα σχεδίασης.

(8.6 Παραλλαγές του Κώδικα VisAP για Οπτικοποίηση .)

Σε αυτήν την υποενότητα, συζητούμε την ανάπτυξη τριών παραλλαγών του κώδικα VisAP, ενός εργαλείου σταδιακής οπτικοποίησης της τοποθέτησης. Αυτές οι παραλλαγές σχεδιάστηκαν για να επιτύχουν την ίδια απεικόνιση, ενσωματώνοντας διαφορετικές προσεγγίσεις υλοποίησης. Θα αναδείξουμε την κοινή λειτουργικότητα του κώδικα και θα τονίσουμε τις κύριες διαφορές μεταξύ των παραλλαγών.

(8.6.1 Επισκόπηση)

Το VisAP είναι ένα εργαλείο σταδιακής οπτικοποίησης της τοποθέτησης που βασίζεται σε Python. Χρησιμοποιεί τις βιβλιοθήκες Pandas, NumPy και Matplotlib για να διαβάζει, επεξεργάζεται και διαγράφει δεδομένα από αρχεία CSV. Το εργαλείο δημιουργεί μια κινούμενη οπτικοποίηση, αναπαριστώντας τα δεδομένα ως ορθογώνια στο 2D γράφημα. Κάθε δεδομένο ανήκει σε μία από τις τρεις τύπους: "P", "R" ή "C". Ο κώδικας ενημερώνει δυναμικά τα καρέ του γραφήματος για να αντικατοπτρίσει τις αλλαγές στα δεδομένα με την πάροδο του χρόνου, δημιουργώντας μια κινούμενη οπτικοποίηση. Ουσιαστικά είναι ο κώδικας που χρησιμοποιείτε στο backend του WeVisAP για την οπτικοποίηση

(8.6.2 Κοινή Λειτουργικότητα)

Οι τρεις παραλλαγές του κώδικα έχουν αρκετές κοινές λειτουργίες:

1. Ανάγνωση και Επεξεργασία Δεδομένων:
 - Διαβάζει δεδομένα από πολλά αρχεία CSV στον τρέχοντα κατάλογο.
 - Χρησιμοποιεί τη βιβλιοθήκη Pandas για να επεξεργαστεί και να οργανώσει τα δεδομένα σε κατάλληλη μορφή για οπτικοποίηση.
 - Κάθε παραλλαγή χρησιμοποιεί διαφορετική μέθοδο επεξεργασίας δεδομένων, όπως παράλληλη επεξεργασία ή επεξεργασία με πολλαπλά νήματα.
2. Διάγραμμα και Κινούμενη Οπτικοποίηση:
 - Χρησιμοποιεί τη βιβλιοθήκη Matplotlib για να δημιουργήσει το 2D γράφημα.
 - Διακρίνει τους διαφορετικούς τύπους δεδομένων ("P", "R" και "C") και τους αναπαριστά χρησιμοποιώντας διαφορετικές ιδιότητες ορθογώνιων [πχ. χρώμα, edge style].
 - Η κινούμενη οπτικοποίηση επιτυγχάνεται με την ενημέρωση των καρτέ του γραφήματος επαναλαμβανόμενα, αντικατοπτρίζοντας τις αλλαγές στα δεδομένα με την πάροδο του χρόνου.
3. Εξαγωγή Αποτελεσμάτων:
 - Καταγράφει διάφορες μετρήσεις, συμπεριλαμβανομένου του χρόνου εκτέλεσης και της χρήσης μνήμης.
 - Αυτές οι μετρήσεις εξάγονται σε ένα αρχείο CSV για περαιτέρω ανάλυση και σύγκριση.

(8.6.3 Βασικές Διαφορές μεταξύ των Παραλλαγών)

Αν και οι τρεις παραλλαγές του VisAP επιτυγχάνουν την ίδια απεικόνιση , διαφέρουν στην προσέγγιση υλοποίησης και ορισμένες άλλες πτυχές:

1. Παραλλαγή 1 [Matplotlib-Anim]:
 - Χρησιμοποιεί το "matplotlib.animation" για τη δημιουργία της κινούμενης οπτικοποίησης.
 - Εξαρτάται από τις λειτουργίες "os" και "multiprocessing" για τη διαχείριση αρχείων και παράλληλη επεξεργασία, αντίστοιχα.
2. Παραλλαγή 2 [MoviePy-Anim]:
 - Ενσωματώνει τα "moviepy.editor" και "moviepy.video.io.bindings" για τη δημιουργία της κινούμενης οπτικοποίησης.
 - Χρησιμοποιεί τις λειτουργίες "os" και "multiprocessing" για τη διαχείριση αρχείων και παράλληλη επεξεργασία, όπως και η Παραλλαγή 1.
3. Παραλλαγή 3 [MP-MoviePy-Anim]:
 - Χρησιμοποιεί την "matplotlib.use('Qt5Agg')" για να καθορίσει το backend για την απεικόνιση.
 - Εφαρμόζει παράλληλη επεξεργασία χρησιμοποιώντας τη βιβλιοθήκη "multiprocessing.pool".
 - Αντί για τις βιβλιοθήκες "matplotlib.animation" ή "moviepy," χρησιμοποιεί τη λειτουργία "mplfig_to_npimage" για να μετατρέψει τα καρέ του γραφήματος σε ακολουθία εικόνων για τη δημιουργία της κινούμενης οπτικοποίησης.
 - Αυτή η παραλλαγή εισάγει μια διαφορετική μέθοδο για τη μέτρηση του χρόνου εκτέλεσης και της χρήσης μνήμης, επικεντρώνοντας στη διαδικασία παράλληλης δημιουργίας καρέ.

Συνοψίζοντας, οι τρεις παραλλαγές του κώδικα VisAP δείχνουν διαφορετικές προσεγγίσεις για την επίτευξη της ίδιας απεικόνισης . Αν και κοινοποιούν κοινές λειτουργίες, κάθε παραλλαγή ενσωματώνει διακριτές βιβλιοθήκες, μεθόδους και τεχνικές επεξεργασίας. Αυτές οι παραλλαγές προσφέρουν ευελιξία στην υλοποίηση, επιτρέποντας στους ερευνητές να επιλέξουν την πλέον κατάλληλη προσέγγιση βάσει των συγκεκριμένων απαιτήσεών τους.

ΚΕΦΑΛΑΙΟ 9 Πειραματική Αξιολόγηση και Συμπεράσματα

(9.1 Χαρακτηριστικά)

Παρουσιάζονται τα αποτελέσματα από διάφορα πειράματα που διεξήχθησαν στον τομέα του Placement. Τα πειράματα βασίστηκαν σε δεδομένα από τα κυκλώματα των ISPD04 benchmark, με τα χαρακτηριστικά αυτών να παρουσιάζονται στον Πίνακα 1. Τα πειράματα διεξήχθησαν σε έναν υπολογιστή με συγκεκριμένα χαρακτηριστικά, τα οποία παρουσιάζονται στον Πίνακα 2.

Πίνακας 4: ISPD 04 benchmarks

Όνομα	Cells	Pads	Nets	Pins	Rows
ibm01	12506	246	14111	50566	96
ibm02	19342	259	19584	81199	109
ibm03	22853	283	27401	93573	121
ibm04	27220	287	31970	105589	136
ibm05	28146	1201	28446	126308	139
ibm06	32332	166	34826	128182	126
ibm07	45639	287	48117	175639	166
ibm08	51023	286	50513	204890	170
ibm09	53110	285	60902	222088	183
ibm10	68685	744	75196	297567	234
ibm11	70152	406	81454	280786	208
ibm12	70439	637	77240	317760	242
ibm13	83709	490	99666	357075	224
ibm14	147088	517	152772	546816	305
ibm15	161187	383	186608	715823	303
ibm16	182980	504	190048	778823	347
ibm17	184752	743	189581	860036	379
ibm18	210341	272	201920	819697	361

Πίνακας 5: Χαρακτηριστικά Συστήματος

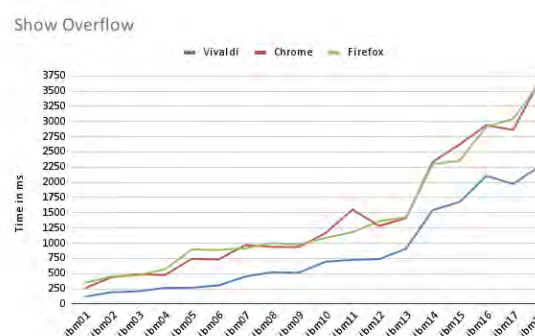
OS	Arch Linux
CPU	Intel Core i7-4702MQ
RAM	8GB DDR3 1600MHz
GPU	NVIDIA GeForce 820M

(9.2 Wevian)

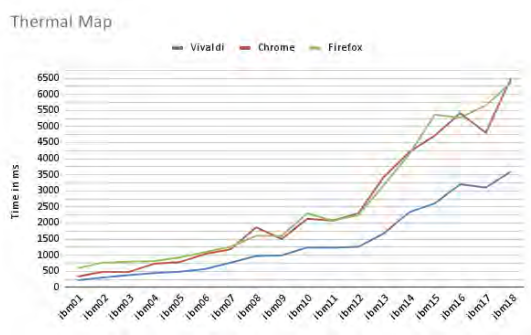
Τα κυκλώματα τέθηκαν σε θέση [Placement] και υποβλήθηκαν σε διαδικασία νομιμοποίησης [Legalized] από τον αλγόριθμο Gordian over Parabacus [35],[36]. Όλα τα πειράματα πραγματοποιήθηκαν με βάση τον μέσο όρο 10 μετρήσεων για να διασφαλιστεί η καλύτερη αξιοπιστία των αποτελεσμάτων. Λόγω του γεγονότος ότι το εργαλείο WEVIAN βασίζεται κυρίως στην JavaScript, μια γλώσσα προγραμματισμού που λειτουργεί client-side, η απόδοση του εργαλείου επηρεάζεται από τον υπολογιστή και, κατ' επέκταση, τον browser που χρησιμοποιείται. Για τον λόγο αυτό, οι μετρήσεις πραγματοποιήθηκαν σε τρεις διαφορετικούς browsers: τον Chrome, τον Firefox και τον Vivaldi. Η εκτέλεση των πειραμάτων σε αυτούς τους browsers επιτρέπει την αξιολόγηση της απόδοσης του εργαλείου σε διαφορετικά περιβάλλοντα, εξασφαλίζοντας έτσι την ολοκληρωμένη και αντιπροσωπευτική αξιολόγηση του WEVIAN, μετρήθηκαν οι χρόνοι εκτέλεσης των ενεργειών όπως Show Overlap, Show Overflow, Thermal Map, Congestion Map και Half-Perimeter, Visualisation.



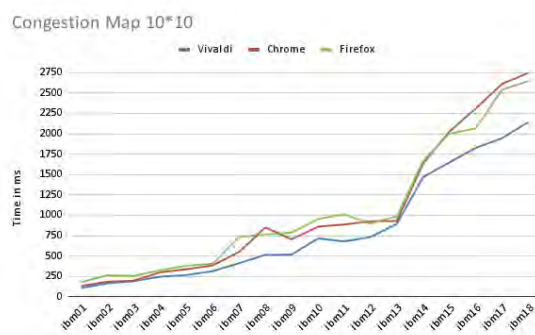
Εικόνα 37: Show Overlap



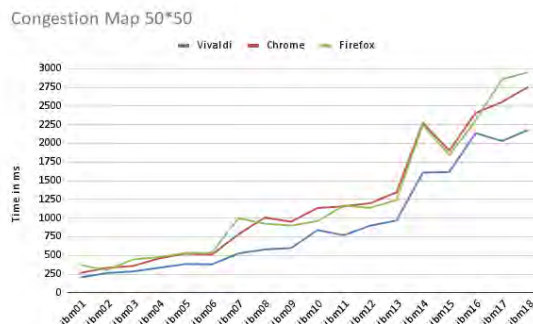
Εικόνα 38: Show Overflow



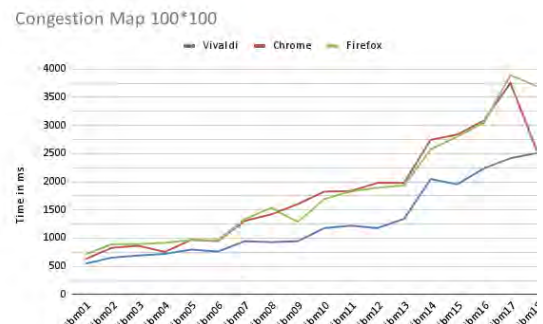
Εικόνα 39: Thermal Map



Εικόνα 40: Congestion Map 10*10



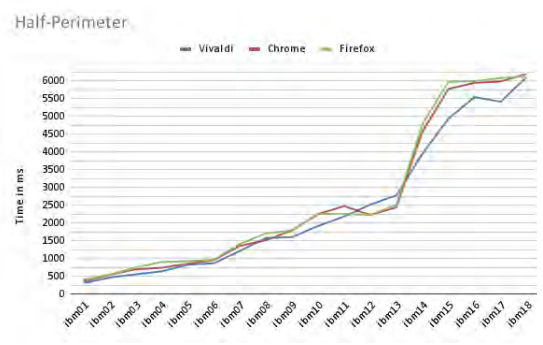
Εικόνα 41: Congestion Map 50*50



Εικόνα 42: Congestion Map 100*100

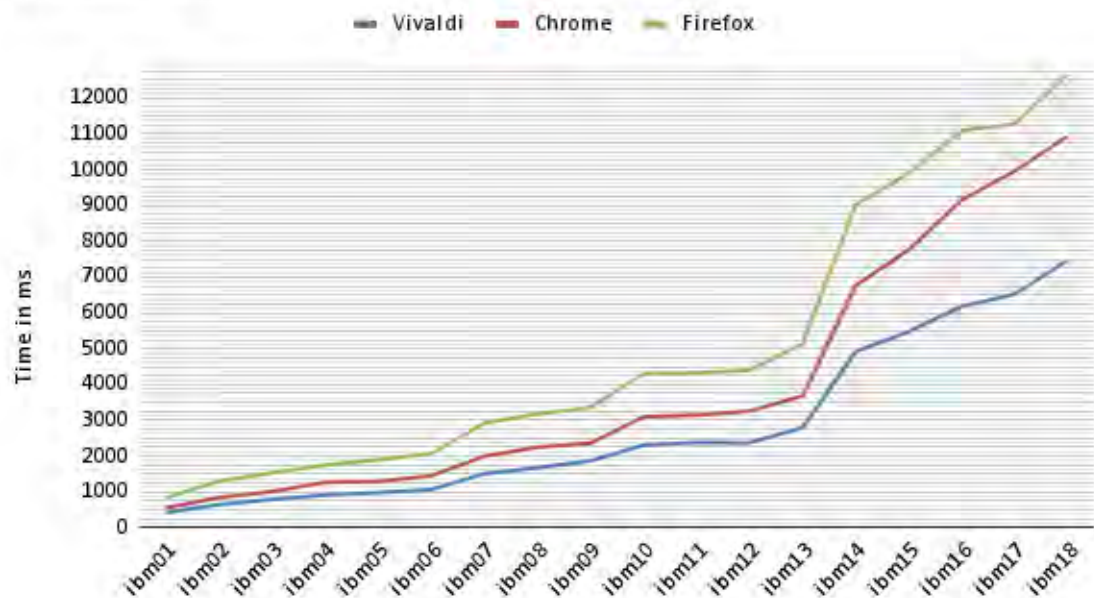


Εικόνα 43: Congestion Map 100*100



Εικόνα 44: Half-Perimeter

Visualisation



Εικόνα 45: Visualisation

(9.2.1 Σύγκριση Φυλλομετρητών)

1. Vivaldi: Σε όλη την πορεία των πειραμάτων, ο Vivaldi έδειξε συνεπώς την ταχύτερη επίδοση για διάφορες ενέργειες του Wevian. Οι χρήστες θα αντιμετωπίζουν μικρότερους χρόνους εκτέλεσης για ενέργειες σε αυτό το πρόγραμμα περιήγησης, συμβάλλοντας σε μια αποτελεσματική και ομαλή εμπειρία.
2. Chrome: Ο Chrome επίσης παρουσίασε καλή επίδοση, με χαμηλότερους χρόνους εκτέλεσης σε σύγκριση με τον Firefox για τις περισσότερες ενέργειες. Οι χρήστες μπορούν να αναμένουν απόδοση απόκρισης και ομαλή εκτέλεση εργασιών όταν χρησιμοποιούν το εργαλείο Wevian στο Chrome.
3. Firefox: Αν και ο Firefox είχε λογική επίδοση, είχε γενικά υψηλότερους χρόνους εκτέλεσης σε σύγκριση με τόσο τον Vivaldi όσο και τον Chrome για τις περισσότερες ενέργειες. Οι χρήστες ενδέχεται να αντιμετωπίσουν ελαφρώς μεγαλύτερους χρόνους εκτέλεσης όταν χρησιμοποιούν το εργαλείο στον Firefox.

Συνοψίζοντας, η ανάλυση απόδοσης του Wevian στα benchmarks ISPD 04 παρείχε πολύτιμες πληροφορίες για την αποτελεσματικότητά του και την απόκρισή του σε διαφορετικά προγράμματα περιήγησης. Η αξιολόγηση διάφορων ενεργειών , συμπεριλαμβανομένων των Show Overlap, Show Overflow, Thermal Map, Congestion Map, Visualisation και Half-Perimeter, αποκάλυψε διαφορετικά χαρακτηριστικά απόδοσης μεταξύ τριών δημοφιλών προγραμμάτων περιήγησης: Vivaldi, Chrome και Firefox.

Το Vivaldi συνεχίζει να εμφανίζει τα πιο ευνοϊκά αποτελέσματα, με ταχύτερους χρόνους εκτέλεσης και αποτελεσματική εκτέλεση των ενεργειών. Η υπεροχή της απόδοσής του το καθιστά τη συνιστώμενη επιλογή για χρήστες που αναζητούν βέλτιστη απόδοση. Ο Chrome επίσης παρουσίασε εξαιρετική απόδοση με γενικά μικρότερους χρόνους εκτέλεσης σε σύγκριση με τον Firefox, καθιστώντας τον αξιόπιστη επιλογή για χρήστες που προτιμούν αυτόν τον περιηγητή. Ωστόσο, είναι σημαντικό να αναγνωρίσουμε ότι ο Firefox, παρόλο που είχε γενικά μεγαλύτερους χρόνους εκτέλεσης, παρέδωσε ικανοποιητική απόδοση.

Με βάση την ανάλυση των πειραμάτων, προτείνονται οι περιηγητές Vivaldi και Chrome για χρήστες που αναζητούν βέλτιστη απόδοση. Και οι δύο περιηγητές επέδειξαν ταχύτερους χρόνους εκτέλεσης για διάφορες ενέργειες. Επιπλέον, η διασφάλιση της διαθεσιμότητας επαρκών πόρων υλικού θα βελτιώσει περαιτέρω την απόδοση του Wevian.

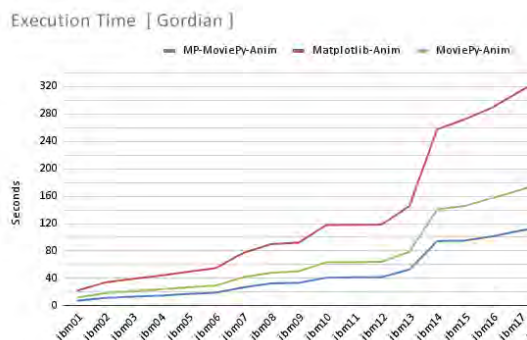
(9.3 VisAP)

Τα κυκλώματα τέθηκαν σε θέση [Placement] και υποβλήθηκαν σε διαδικασία νομιμοποίησης [Legalized] από διάφορους αλγόριθμους οι οποίοι παρουσιάζονται στον Πίνακα 6. 2. Όλα τα πειράματα πραγματοποιήθηκαν με βάση τον μέσο όρο 10 μετρήσεων για να διασφαλιστεί η καλύτερη αξιοπιστία των αποτελεσμάτων. Τα πειράματα έγιναν ξεχωριστά σε κάθε από τις μια εκδοχές/παραλλαγές του κώδικα.

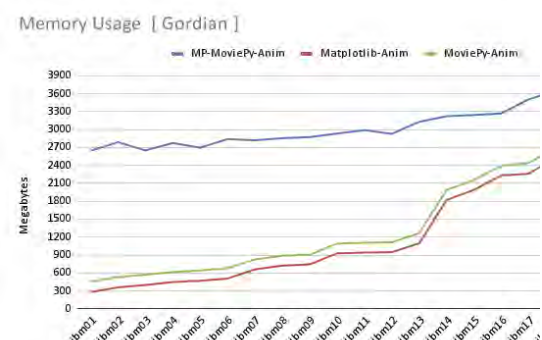
Πίνακας 6: Αλγόριθμοι

Αλγόριθμοι	
NTUPlace3	tetrisPlaceNets
mPL6	tetris
Gordian	parAbacus
Eplace	netTetris
	abacusNet
	abacusNet

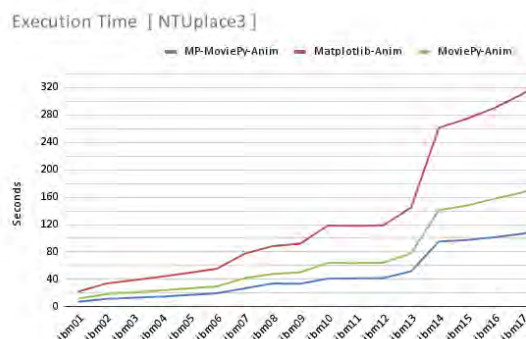
[35],[36],[37],[38],[39]



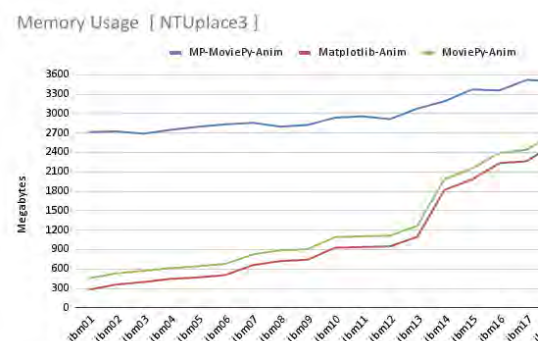
Εικόνα 46: Execution Time [Gordian]



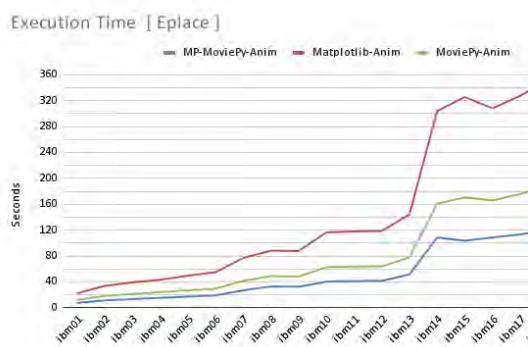
Εικόνα 47: Memory Usage [Gordian]



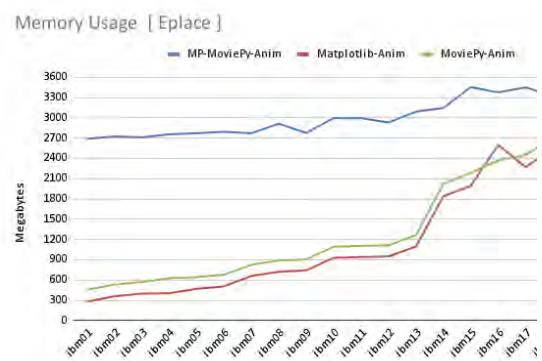
Εικόνα 49: Execution Time [NTUPlace3]



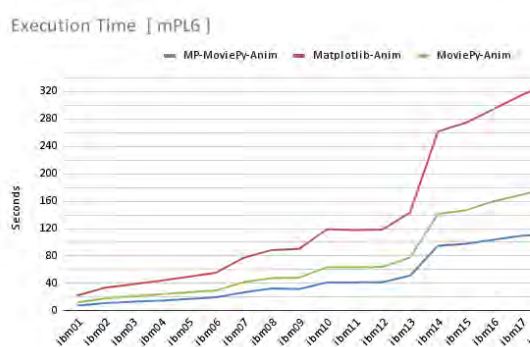
Εικόνα 48: Memory Usage [NTUPlace3]



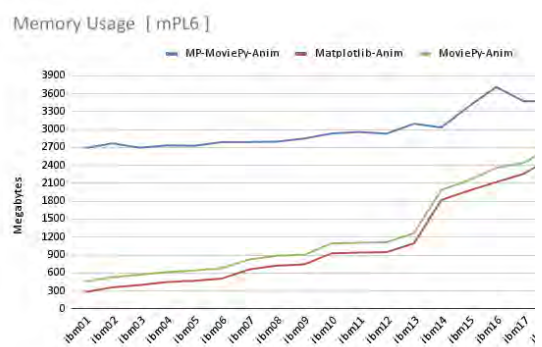
Εικόνα 51: Execution Time [Eplace]



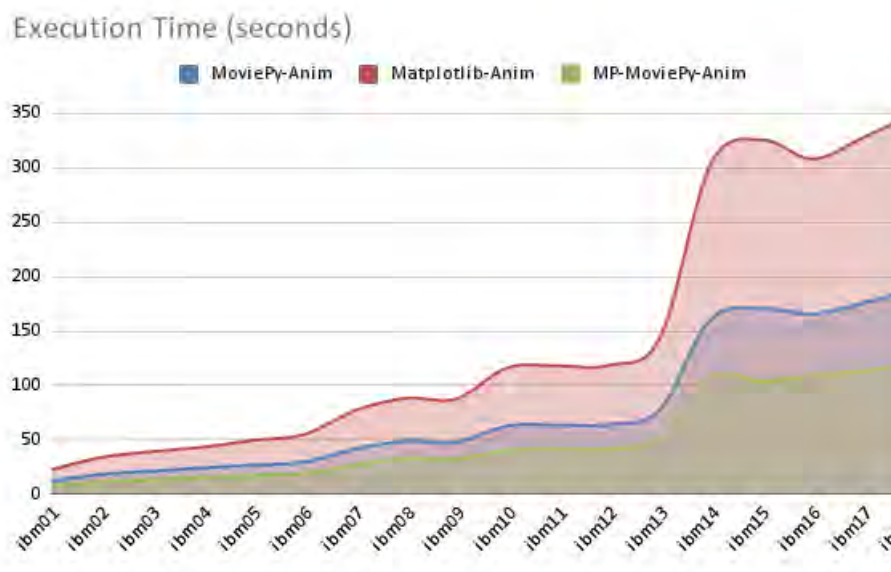
Εικόνα 50: Memory Usage [Eplace]



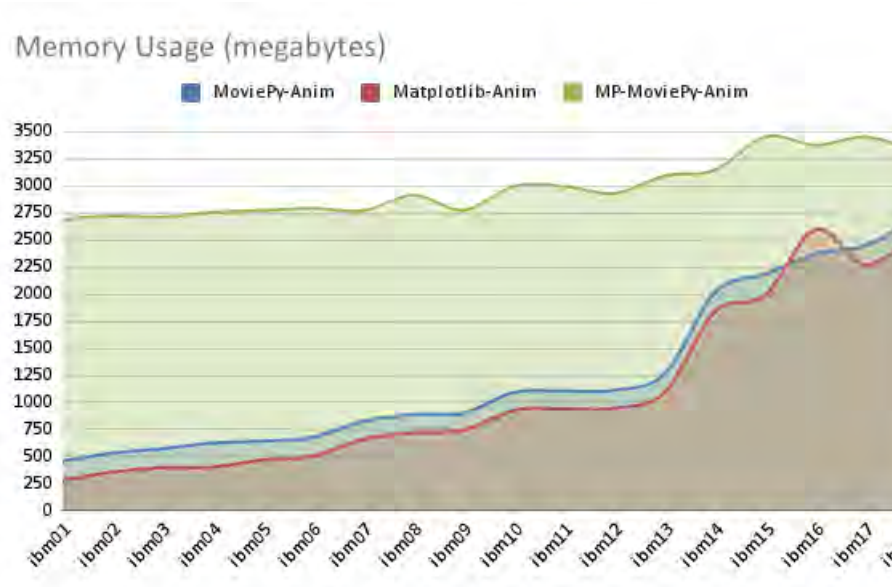
Εικόνα 53: Execution Time [mPL6]



Εικόνα 52: Memory Usage [mPL6]



Εικόνα 54: Execution Time Average of each Benchmark



Εικόνα 55: Memory Usage Average of each Benchmark

(9.3.1 Συγκριτική Ανάλυση των Τριών Παραλλαγών του Κώδικα)

Για να συγκρίνουμε τις επιδόσεις των τριών παραλλαγών του κώδικα VisAP [MoviePy-Anim, Matplotlib-Anim και MP-MoviePy-Anim], αναλύσαμε τους χρόνους εκτέλεσής τους σε δευτερόλεπτα και τη χρήση μνήμης σε megabytes ανάμεσα στα προαναφερθέντα benchmarks και αλγόριθμους.

Πίνακας 7: Μέσος Όρος Ανάμεσα σε όλα τα Benchmarks και Αλγόριθμους

Παραλλαγή	Μέσος Χρόνος Εκτέλεσης (seconds)	Μέση Χρήση Μνήμης (megabytes)
MoviePy-Anim	81.1	708.9
Matplotlib-Anim	126.5	1126.8
MP-MoviePy-Anim	32.1	2785.8

Με βάση τους μέσους χρόνους εκτέλεσης και τη χρήση μνήμης, μπορούμε να συμπεράνουμε τα εξής:

Το MP-MoviePy-Anim είναι η ταχύτερη παραλλαγή μεταξύ των τριών, με μέσο χρόνο εκτέλεσης 32,1 δευτερόλεπτα. Ξεπερνά το MoviePy-Anim κατά περίπου 49 δευτερόλεπτα [περίπου 60.5% ταχύτερη] και το Matplotlib-Anim κατά περίπου 94,4 δευτερόλεπτα [περίπου 74,6% ταχύτερη].

Το Matplotlib-Anim έχει τον υψηλότερο μέσο χρόνο εκτέλεσης 126.5 δευτερόλεπτα, καθιστώντας την πιο αργή παραλλαγή μεταξύ των τριών. Σε σύγκριση με το MP-MoviePy-Anim, είναι περίπου 94.4 δευτερόλεπτα πιο αργή [περίπου 74.6% πιο αργή] και περίπου 45.4 δευτερόλεπτα πιο αργή από το MoviePy-Anim [περίπου 56.1% πιο αργή].

Όσον αφορά τη χρήση μνήμης, το MoviePy-Anim καταναλώνει τη λιγότερη μνήμη κατά μέσο όρο 708,9 MB σε σύγκριση με τα Matplotlib-Anim και MP-MoviePy-Anim. Το Matplotlib-Anim έχει μέση χρήση μνήμης 1126,8 MB, ενώ το MP-MoviePy-Anim έχει την υψηλότερη μέση χρήση μνήμης 2785,8 MB. συμπίερασμα

Η ανάλυση δείχνει ότι το MP-MoviePy-Anim είναι η ταχύτερη παραλλαγή του κώδικα VisAp, παρέχοντας σημαντική επιτάχυνση σε σύγκριση με τα MoviePy-Anim και Matplotlib-Anim. Ωστόσο, έρχεται με κόστος υψηλότερης χρήσης μνήμης. Από την άλλη πλευρά, το MoviePy-Anim παρουσιάζει τη χαμηλότερη κατανάλωση μνήμης, αλλά είναι πιο αργό από το MP-MoviePy-Anim. Οι ερευνητές μπορούν να επιλέξουν την κατάλληλη παραλλαγή με βάση την προτεραιότητά τους μεταξύ ταχύτητας εκτέλεσης και απόδοσης μνήμης, λαμβάνοντας υπόψη τις ανάγκες και περιορισμούς τους.

BIBΛΙΟΓΡΑΦΙΑ

- [1] Bardeen, J., & Brattain, W. H. (1948, July). The Transistor, A Semi-Conductor Triode. *Phys. Rev.*, 74(2), 230-231. doi: 10.1103/PhysRev.74.230
- [2] Hoerni, J. A. (1961). Method of Making Transistor Structure. US Patent 2,981,877.
- [3] Kilby, J. S. (1959). Miniaturized Electronic Circuits. US Patent 3,138,743.
- [4] Colinge, J.-P. (2004) *Silicon-on-insulator technology: Materials to VLSI Materials To Vlsi*. New York, NY: Springer US.
- [5] Rabaey, J.M. (1996) *Digital Integrated Circuits: A design perspective*. London: Prentice-Hall.
- [6] Mead, C., & Conway, L. (1980). *Introduction to VLSI Systems*. Addison-Wesley Reading, MA.
- [7] *Semiconductor devices: Physics and technology* (2012). New York: Wiley.
- [8] Shin, J. et al. (2011) 'Floorplanning challenges in early chip planning', 2011 IEEE International SOC Conference [Preprint]. doi:10.1109/socc.2011.6085096.
- [9] Hyunchul Shin and Chunghee Kim (1993) 'A simple yet effective technique for partitioning', *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 1(3), pp. 380–386. doi:10.1109/92.238449.
- [10] Gao, T., Vaidya, P.M. and Liu, C.L. (1991) 'A new performance driven placement algorithm', *1991 IEEE International Conference on Computer-Aided Design Digest of Technical Papers* [Preprint]. doi:10.1109/iccad.1991.185187.
- [11] Clock tree synthesis Digital VLSI Design Lecture 1: Introduction - biu. Available at: <https://www.eng.biu.ac.il/temanad/files/2017/02/Lecture-8-CTS.pdf>
- [12] N. Togawa, M. Sato and T. Ohtsuki, "A simultaneous placement and global routing algorithm with path length constraints for transport-processing FPGAs," *Proceedings of ASP-DAC '97: Asia and South Pacific Design Automation Conference*, Chiba, Japan, 1997, pp. 569-578, doi: 10.1109/ASPDAC.1997.600338.
- [13] Rashinkar, P., Paterson, P. and Singh, L. (2002) *System-on-a-chip verification methodology and Techniques*. New York, NY: Springer US.

- [14] Chiang, C. and Kawa, J. (2007) *Design for manufacturability and yield for nano-scale CMOS*. Springer.
- [15] Alpert, C.J., Mehta, D.P., & Sapatnekar, S.S. (2008). *Handbook of Algorithms for Physical Design Automation*. CRC Press. ISBN: 9781000654196.
- [16] Zhao, P., Wang, Z. and Hang, G. (2010) 'Power Optimization for VLSI circuits and systems', *2010 10th IEEE International Conference on Solid-State and Integrated Circuit Technology* [Preprint]. doi:10.1109/icsict.2010.5667299.
- [17] Janert, P.K. (2016) *Gnuplot in action: Understanding data with graphs*. Shelter Island, NY: Manning Publications Co.
- [18] Ivan S. Kourtev and Eby G. Friedman. 2000. *Timing optimization through clock skew scheduling*. Springer
- [19] Andrew B. Kahng, Jens Lienig, Igor L. Markov, and Jin Hu. 2011. *VLSI Physical Design: From Graph Partitioning to Timing Closure* (1st. ed.). Springer Publishing Company, Incorporated.
- [20] Baker, R. J. (2010). *CMOS Circuit Design, Layout, and Simulation* (3rd ed.). Wiley.
- [21] Weste, N. H. E., & Harris, D. M. *CMOS VLSI Design: A Circuits and Systems Perspective*. Pearson Education.
- [22] Laung-Terng Wang, Yao-Wen Chang, and Kwang-Ting (Tim) Cheng. 2009. *Electronic Design Automation: Synthesis, Verification, and Test*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- [23] Chandrasetty, V.A. (2011). *VLSI Design: A Practical Guide for FPGA and ASIC Implementations*. Springer New York. ISBN: 9781461411208.
- [24] Wolf, S., & Tauber, R. N. (1986). *Silicon Processing for the VLSI Era: Process Integration*. Lattice Press. ISBN: 9780961672133.
- [25] Baker, R.J. (2019) *CMOS: Circuit design, layout, and simulation*. Chichester: Wiley Blackwell.
- [26] H. S. Baird and Y. E. Cho. 1975. An artwork design verification system. In *Proceedings of the 12th Design Automation Conference (DAC '75)*. IEEE Press, 414–420.

- [27] Ding, Y. (2016). Design for Manufacturability in Advanced Lithography Technologies. Iowa State University. Available at: <https://core.ac.uk/download/pdf/141671385.pdf>
- [28] What is the antenna effect in VLSI?, ChipEdge VLSI Training Company. Available at: <https://chippedge.com/what-is-the-antenna-effect-in-vlsi/>
- [29] General information International Symposium on Physical Design (ISPD). Available at: <https://ispd.cc/ispd2024/index.php>
- [30] *GSRC Bookshelf Format for placement instances VLSICAD Page*. Available at: <http://vlsicad.eecs.umich.edu/BK/ISPD06bench/BookshelfFormat.txt>
- [31] *DAC 2012 Routability-Driven Placement Contest and Benchmark Suite*. Available at: http://archive.sigda.org/dac2012/contest/dac2012_contest.html.
- [32] *Placement formats, rev. 1.2 GSRC Calibrating Achievable Design Bookshelf: Placement Formats*. Available at: <https://vlsicad.ucsd.edu/GSRC/bookshelf/Slots/Placement/plFormats.html>
- [33] Konstantinos Mantos, Nikolaos Koungalis, Georgios Demesiotis, Panagiotis Oikonomou, Antonios N. Dadaliaris, and Georgios I. Stamoulis. 2017. WEVIAN: A Web-based Placement Visualizer / Analyzer. In Proceedings of the 21st Pan-Hellenic Conference on Informatics (PCI '17). Association for Computing Machinery, New York, NY, USA, Article 16, 1–6.
- [34] Wevian-Visualizer/wevian: A web-based placement visualizer / analyzer, GitHub. Available at: <https://github.com/wevian-visualizer/wevian>
- [35] Oikonomou P, Dadaliaris AN, Kolomvatsos K, Loukopoulos T, Kakarountas A, Stamoulis GI. Improved Parallel Legalization Schemes for Standard Cell Placement with Obstacles. *Technologies*. 2019; 7(1):3. <https://doi.org/10.3390/technologies7010003>
- [36] J. M. Kleinhans, G. Sigl, F. M. Johannes and K. J. Antreich, "GORDIAN: VLSI placement by quadratic programming and slicing optimization," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 10, no. 3, pp. 356-365, March 1991, doi: 10.1109/43.67789.
- [37] Chen, T., Jiang, Z., Hsu, T., Chen, H., & Chang, Y. (2008). NTUplace3: An Analytical Placer for Large-Scale Mixed-Size Designs With Preplaced Blocks and Density Constraints *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 27, 1228-1240.

[38] Tony F. Chan, Jason Cong, Joseph R Shinnerl, Kenton Sze, and Min Xie. 2006. MPL6: enhanced multilevel mixed-size placement. In Proceedings of the 2006 international symposium on Physical design (ISPD '06). Association for Computing Machinery, New York, NY, USA, 212–214. <https://doi.org/10.1145/1123008.1123055>

[38] J. M. Kleinhans, G. Sigl, F. M. Johannes and K. J. Antreich, "GORDIAN: VLSI placement by quadratic programming and slicing optimization," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 10, no. 3, pp. 356-365, March 1991, doi: 10.1109/43.67789.

[39] J. Lu et al., "ePlace: Electrostatics based placement using Nesterov's method," 2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC), San Francisco, CA, USA, 2014, pp. 1-6, doi: 10.1145/2593069.2593133.