

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ – ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΔΙΚΤΥΩΝ
ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΑΤΡΙΒΗ

Πύρκα Αντωνία

**Ανάλυση χρονισμού ολοκληρωμένων
κυκλωμάτων VLSI υπο πραγματικές συνθήκες
λειτουργίας του δικτύου παροχής ισχύος**

Εκπονήθηκε υπό την επίβλεψη των:

Ευμορφόπουλου Νέστορα
Σταμούλη Γεώργιου
Μούντανου Ιωάννη

Βόλος, Νοέμβριος 2009



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
ΒΙΒΛΙΟΘΗΚΗ & ΚΕΝΤΡΟ ΠΛΗΡΟΦΟΡΗΣΗΣ
ΕΙΔΙΚΗ ΣΥΛΛΟΓΗ «ΓΚΡΙΖΑ ΒΙΒΛΙΟΓΡΑΦΙΑ»**

Αριθ. Εισ.: 7682/1

Ημερ. Εισ.: 21-12-2009

Δωρεά: Συγγραφέα

Ταξιθετικός Κωδικός: Δ

621.381 5

ΠΥΡ

Περιεχόμενα

1. Στατική Ανάλυση Χρονισμού

1.1 Εισαγωγή.....	5
1.2 Οι χρονισμοί των μονοπατιών.....	5
1.3 Έλεγχος των περιορισμών.....	8
1.4 Προσδιορισμός των καθυστερήσεων λογικής πύλης, μονοπατιού και ολόκληρου του κυκλώματος.....	11

2. Ανάλυση του Δικτύου Διανομής Ισχύος

2.1 Μέθοδος των κόμβων (Nodal Analysis - NA).....	17
2.1.1 Νόμος των ρευμάτων του Kirchhoff.....	19
2.1.2 Σχέση ανάμεσα στις τάσεις των κόμβων και των κλάδων (έκφραση νόμου τάσεων του Kirchhoff).....	20
2.1.3 Σχέση ρεύματος - τάσης (i-v) μεταξύ των κλάδων του κυκλώματος.....	21
2.2 Τροποποιημένη μέθοδος των κόμβων (Modified Nodal Analysis - MNA).....	23
2.3 Διαμόρφωση του συστήματος της MNA ως προς τις πτώσεις της τάσης.....	27
2.4 Αριθμητική επίλυση γραμμικών κυκλωμάτων στο διακριτό χρόνο.....	28
2.5 Μέθοδος των κόμβων στο διακριτό χρόνο.....	29

3. Προσδιορισμός των τάσεων μεγιστοποίησης της καθυστέρησης του κυκλώματος.....

32

4. Πειραματικοί έλεγχοι και αποτελέσματα.....

37

Παράρτημα Α- Κώδικας MATLAB

A.1 Δημιουργία των κυματομορφών των ρευμάτων του αρχείου εισόδου, του διανύσματος των κυματομορφών των εισόδων που ανήκουν στο δείγμα, και υπολογισμός του ρεύματος στα σημεία αυτά.....	44
A.2 Αρχικοποίηση μεταβλητών και καθορισμός των κόμβων που τοποθετούνται οι πηγές ρεύματος και οι πηγές τροφοδοσίας..	45
A.3 Υπολογισμών του διανύσματος μετακίνησης και των μεγιστικών σημείων (maximal points) - Σχηματισμός του δικτύου ισχύος και ανάλυσή του.....	46

Παράρτημα Β- NanoTime tutorial

B.1 Τα χαρακτηριστικά του NanoTime.....	51
B.2 Η ανάλυση ροής του NanoTime	53
B.2.1 Φάση Netlist.....	54
B.2.2 Διάδοση του Ρολογιού και Αναγνώριση της Τοπολογίας (Clock Propagation and Topology Recognition)	55
B.2.3 Καθορισμός των Χρονικών Περιορισμών (Specifying Timing Constraints)	55
B.2.4 Αναγνώριση και Προσομοίωση των Μονοπατιών (Path Tracing and Simulation).....	55
B.2.5 Αναφορές σχετικά με την Ανάλυση (Analysis Reporting)	55
B.3 Ξεκινώντας το NanoTime	56
B.3.1 Ξεκινώντας μία NanoTime session	56
B.3.2 Τερματίζοντας μία NanoTime session.....	56
B.4 Χρησιμοποιώντας το Shell Interface.....	56
B.4.1 Scripts Εντολών.....	57
B.4.2 Αποτελέσματα Εντολών	57
B.4.3 Μεταβλητές	58
B.4.4 Διακοπή Εκτέλεσης Εντολών	58
B.4.5 Tcl Interface.....	58
B.5 Γραπτή και Ηλεκτρονική Βοήθεια για το NanoTime.....	59
B.6 Script.....	59

Βιβλιογραφία	65
--------------------	----

Περίληψη

Στην παρούσα διπλωματική εργασία, θα ασχοληθούμε με ένα από τα σημαντικότερα προβλήματα που παρουσιάζονται στη σύγχρονη σχεδίαση ψηφιακών κυκλωμάτων. Το πρόβλημα αυτό είναι η ανάλυση χρονισμού των κυκλωμάτων κάτω από πραγματικές συνθήκες, και όχι πλασματικές (προσεγγιστικές), όπως γίνεται στην πλειοψηφία των σύγχρονων εργαλείων σχεδίασης με τη βοήθεια ηλεκτρονικών υπολογιστών (CAD εργαλείων).

Τα ολοκληρωμένα κυκλώματα υψηλής απόδοσης παραδοσιακά χαρακτηρίζονται από τη συχνότητα του ρολογιού, στην οποία δουλεύουν. Εκτιμώντας την ικανότητα ενός κυκλώματος να δουλεύει σε μία καθορισμένη ταχύτητα, απαιτείται η ικανότητα να υπολογίζεται, κατά τη διαδικασία της σχεδίασης, η καθυστέρηση του κυκλώματος. Καθώς μία τέτοια μέτρηση μπορεί να γίνει προσομοιώνοντας το κύκλωμα, μία τέτοια προσέγγιση είναι πιθανώς πολύ αργή για να γίνει. Η στατική ανάλυση χρονισμού παίζει ένα ζωτικό ρόλο στη βελτίωση του χρόνου και τον ακριβή υπολογισμό της καθυστέρησης. Στο 1^ο κεφάλαιο παρουσιάζονται τα βήματα τα οποία ακολουθεί το NanoTime προκειμένου να κάνει στατική ανάλυση χρονισμού. Επίσης, προσδιορίζονται οι καθυστερήσεις λογικής πύλης, μονοπατιού και ολόκληρου του κυκλώματος.

Στο επόμενο κεφάλαιο (2^ο), αρχικά γίνεται η περιγραφή της μεθόδου των κόμβων (NA). Η συγκεκριμένη μέθοδος, ανάγει ένα σύστημα διαφορικών εξισώσεων δεύτερης τάξης, σε ένα σύστημα διαφορικών εξισώσεων πρώτης τάξης. Εφαρμόζεται όμως σε κυκλώματα που περιέχουν μόνο στοιχεία R και C και όχι στοιχεία L. Γι' αυτό καταφεύγουμε σε μία τροποποίηση της μεθόδου των κόμβων, η οποία είναι πιο εύκολη στην επίλυση και την επεξεργασία της. Αυτή είναι η τροποποιημένη μέθοδος των κόμβων (MNA), η οποία αναλύεται στο ίδιο κεφάλαιο και με την εφαρμογή της, προκύπτει ένα σύστημα δύο αναδρομικών σχέσεων (στο συνεχή χρόνο), που δίνει το διάλυμα των πτώσεων τάσης καθώς και των ρευμάτων κλάδων.

Στο 3^ο κεφάλαιο, δίνεται ο ορισμός της αυστηρά αύξουσας μονότονης συνάρτησης (που είναι και η καθυστέρηση) και παρουσιάζονται τα μεγιστικά σημεία (maximal

points). Αυτά αποτελούν τα σημεία χειρότερης πτώσης τάσης και κατ' επέκτασιν τα σημεία που μεγιστοποιείται η καθυστέρηση του κυκλώματος.

Τέλος, στο 4^ο κεφάλαιο, προσομοιώσαμε το κύκλωμα c6288, που είναι ένας πολλαπλασιαστής. Αρχικά, χρησιμοποιήσαμε μία μέθοδο για τον έλεγχο της ακεραιότητας τροφοδοσίας και της ευστάθειας του δικτύου διανομής ισχύος. Αυτό σημαίνει πως η μέση πτώση τάσης του, πρέπει να είναι το πολύ 10% της τάσης τροφοδοσίας. Έπειτα τροφοδοτήσαμε το κύκλωμα με τις τάσεις που προέκυψαν από τη συγκεκριμένη μέθοδο και υπολογίσαμε τη μέγιστη καθυστέρησή του σε δύο περιπτώσεις. Στην πρώτη, τροφοδοτούμε το κύκλωμα με 32 διαφορετικές τάσεις και στην δεύτερη με μία, την μικρότερη από τις 32.

Η εκπόνηση της εργασίας αυτής δε θα ήταν δυνατή χωρίς τη συμβολή και βοήθεια του επιβλέποντα καθηγητή κ. Ευμορφόπουλου Νέστορα και των καθηγητών κ. Σταμούλη Γεώργιου και κ. Μούντανου Ιωάννη, τους οποίους και θα ήθελα να ευχαριστήσω.

1. Στατική ανάλυση χρονισμού

1.1 Εισαγωγή

Σε ένα σύγχρονο ψηφιακό κύκλωμα (είναι ένα ψηφιακό κύκλωμα του οποίου τα μέρη συγχρονίζονται από το σήμα ενός ρολογιού), τα δεδομένα υποτίθεται πως μετακινούνται σε ένα lockstep, παράγοντας μία φάση-κατάσταση σε κάθε στιγμή του ρολογιού. Αυτό γίνεται από τα στοιχεία συγχρονισμού, όπως είναι τα flip-flops ή τα latches, τα οποία αντιγράφουν την είσοδό τους στην έξοδό τους όταν τους το 'λέει' το ρολόι.

Ο κύριος στόχος της ανάλυσης χρονισμού είναι να επιβεβαιώσει πως, παρά όλους τους πιθανούς χρονικούς περιορισμούς, τα σήματα δεν θα φτάσουν ούτε πολύ αργά, ούτε πολύ γρήγορα και άρα θα εξασφαλιστεί σωστή λειτουργία του κυκλώματος. Με λίγα λόγια, η στατική ανάλυση χρονισμού είναι μία μέθοδος αξιολόγησης της αξιοπιστίας της απόδοσης μίας σχεδίασης, ελέγχοντας όλα τα μονοπάτια για παραβιάσεις του χρονισμού.

Το NanoTime ελέγχει για παραβιάσεις με τον ίδιο τρόπο που θα γινόταν με τα χέρια, αλλά πολύ πιο γρήγορα και με πολύ περισσότερη ακρίβεια. Για να γίνει αυτός ο έλεγχος, 'σπάει' τη σχεδίαση σε ένα σύνολο από μονοπάτια χρονισμού, υπολογίζει την καθυστέρηση διάδοσης των σημάτων κατά μήκος κάθε μονοπατιού και ελέγχει για παραβιάσεις των χρονικών περιορισμών στο εσωτερικό και στις εισόδους και εξόδους της σχεδίασης.

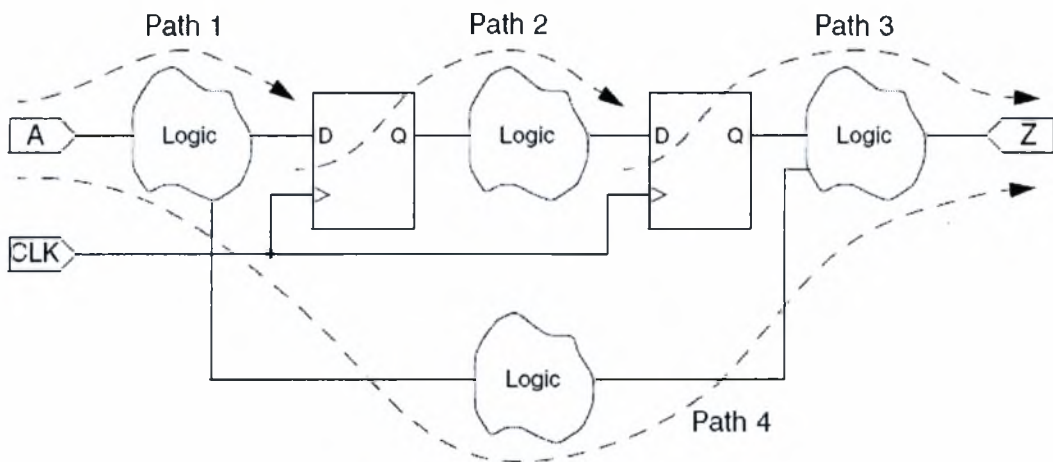
1.2 Οι χρονισμοί των μονοπατιών

Στην στατική ανάλυση χρονισμού, η λέξη *στατική* αναφέρεται στο γεγονός πως η χρονική ανάλυση γίνεται με έναν εσωτερικό και ανεξάρτητο τρόπο (αγνοεί τα false μονοπάτια, δηλαδή τα μονοπάτια που αντιδρούν σε λογικές αλλαγές που δεν έχουν συμβεί) και έχει σαν σκοπό να εντοπίσει τα χειρότερα, από άποψη καθυστέρησης, μονοπάτια του κυκλώματος ανάμεσα σε όλα τα πιθανά. Το μονοπάτι μεταξύ μίας εισόδου και μίας εξόδου με τη μέγιστη καθυστέρηση ονομάζεται κρίσιμο μονοπάτι

(critical path). Η υπολογιστική ικανότητα μιας τέτοιας προσέγγισης (ο αριθμός των ακμών στον γράφο είναι γραμμικός) έχει συμβάλει στην διάδοση της χρήσης της.

Το NanoTime αναλύει το χρονισμό κάθε μονοπατιού της σχεδίασης από το σημείο έναρξης μέχρι το σημείο τερματισμού. Το σημείο έναρξης είναι ένα σημείο στη σχεδίαση όπου τα δεδομένα ξεκινούν με τον παλμό του ρολογιού. Τα δεδομένα διασχίζουν το μονοπάτι μέσω συνδυαστικών λογικών latches και latches που χρονίζονται στην ακμή του ρολογιού και ‘αιχμαλωτίζονται’ στο σημείο τερματισμού από έναν άλλο παλμό. Η θέση των σημείων έναρξης και τερματισμού καθορίζονται κατά την αναγνώριση της τοπολογίας.

Το σημείο έναρξης ενός μονοπατιού είναι ένα clock pin ενός ακολουθιακού στοιχείου, ή πιθανώς μία από τις εισόδους της σχεδίασης (καθώς τα δεδομένα εισόδου μπορούν να προέρχονται από κάποια εξωτερική πηγή). Το σημείο τερματισμού είναι ένα data pin ενός ακολουθιακού στοιχείου, ή πιθανώς κάποια από τις εξόδους της σχεδίασης (καθώς τα δεδομένα εξόδου μπορούν να καταλήξουν σε κάποια εξωτερική πηγή).

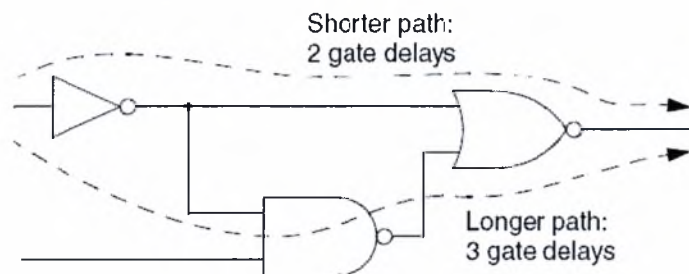


Εικόνα 1.1 Μία απλή σχεδίαση και τα μονοπάτια που περιέχονται σ' αυτήν

Σε αυτή την εικόνα, κάθε σύννεφο με το όνομα ‘logic’ αναπαριστά ένα συνδυαστικό κύκλωμα. Κάθε μονοπάτι ξεκινάει από ένα σημείο εκκίνησης των δεδομένων, διέρχεται από κάποιο συνδυαστικό κύκλωμα (πιθανώς latches που χρονίζονται στην ακμή του ρολογιού) και τελειώνει σε ένα σημείο τερματισμού.

- Το Path 1 ξεκινάει από μία είσοδο του κυκλώματος και τελειώνει στην είσοδο δεδομένων ενός ακολουθιακού στοιχείου.
- Το Path 2 ξεκινάει από το ρολόι ενός ακολουθιακού κυκλώματος και τελειώνει στην είσοδο δεδομένων ενός ακολουθιακού στοιχείου.
- Το Path 3 ξεκινάει από το ρολόι ενός ακολουθιακού κυκλώματος και τελειώνει σε μία έξοδο του κυκλώματος.
- Το Path 4 ξεκινάει από μία είσοδο του κυκλώματος και τελειώνει σε μία έξοδο του κυκλώματος.

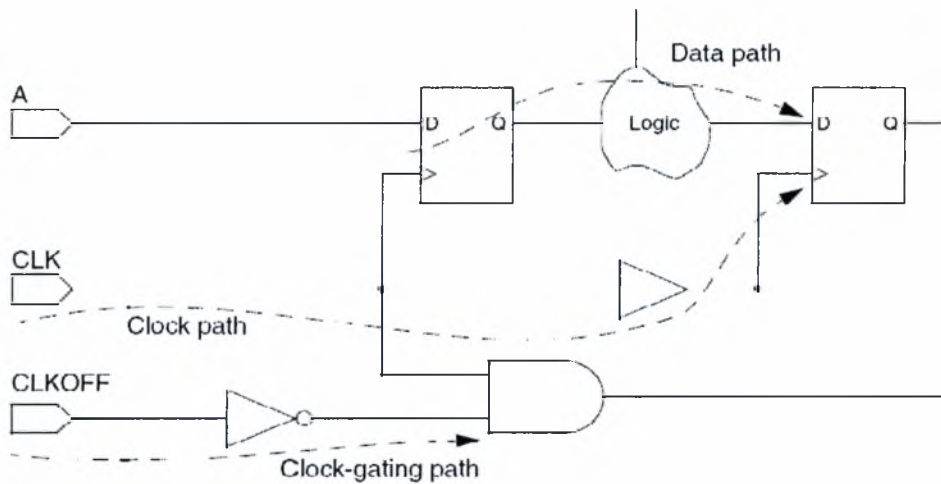
Ένα 'logic' σύννεφο μπορεί να περιέχει πολλά μονοπάτια όπως φαίνεται στην εικόνα 1.2. Το NanoTime χρησιμοποιεί το μεγαλύτερο μονοπάτι για να υπολογίσει τη μέγιστη καθυστέρηση ή το μικρότερο για να υπολογίσει την ελάχιστη καθυστέρηση.



Εικόνα 1.2 Πολλά μονοπάτια μέσα σε ένα 'logic' σύννεφο

Επιπλέον από τα μονοπάτια που μόλις περιγράφηκαν, το NanoTime μελετάει και άλλους τύπους μονοπατιών για να κάνει χρονική ανάλυση και να ελέγξει για setup και hold περιορισμούς, όπως οι παρακάτω:

- Clock path: ένα μονοπάτι που ξεκινάει από ένα ρολόι, διαπερνά buffers ή αντιστροφείς και τελειώνει στο ρολόι ενός ακολουθιακού στοιχείου.
- Clock-gating path: ένα μονοπάτι που ξεκινάει από μία είσοδο του κυκλώματος και τελειώνει σε μία πύλη.



Εικόνα 1.3 Τύποι μονοπατιών

1.3 Έλεγχος των περιορισμών

Στο σύστημα που αναφέρθηκε πιο πάνω, μόνο δύο είδη χρονικών περιορισμών είναι πιθανοί:

- Ο setup χρονικός περιορισμός, ο οποίος καθορίζει πόσος χρόνος χρειάζεται έτσι ώστε τα δεδομένα να είναι διαθέσιμα στην είσοδο μιας συνδυαστικής συσκευής, πριν από τον παλμό που 'αιχμαλωτίζει' τα δεδομένα στη συσκευή. Αυτός ο περιορισμός επιβάλλει μία μέγιστη καθυστέρηση στο μονοπάτι που σχετίζεται με το μονοπάτι του ρολογιού. Αυτού του είδους ο περιορισμός θα μας απασχολήσει στην συγκεκριμένη εργασία.
- Ο hold χρονικός περιορισμός, ο οποίος καθορίζει πόσος χρόνος χρειάζεται έτσι ώστε τα δεδομένα να βρίσκονται στην είσοδο μιας συνδυαστικής συσκευής, μετά από τον παλμό που 'αιχμαλωτίζει' τα δεδομένα στη συσκευή. Αυτός ο περιορισμός επιβάλλει μία ελάχιστη καθυστέρηση στο μονοπάτι που σχετίζεται με το μονοπάτι του ρολογιού.

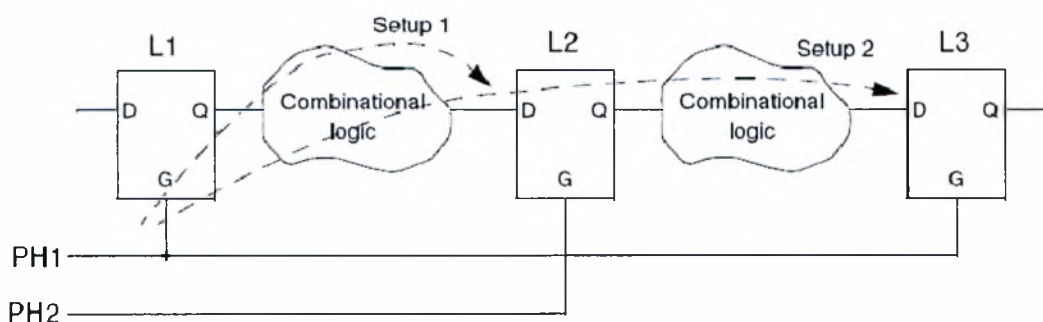
Ο χρόνος που φτάνει ένα σήμα ποικίλει εξαιτίας πολλών λόγων. Εξαρτάται από την τεχνολογία κατασκευής του ψηφιακού κυκλώματος, αλλά επηρεάζεται επίσης από την τάση τροφοδοσίας, τις οδηγούμενες χωρητικότητες και τη θερμοκρασία περιβάλλοντος. Σε γενικές γραμμές, ο χρόνος αυξάνεται με την αύξηση της θερμοκρασίας περιβάλλοντος και των χωρητικότητων, ενώ μειώνεται με την αύξηση της τάσης τροφοδοσίας.

Αφού το NanoTime καθορίσει τα μονοπάτια και υπολογίσει τις καθυστερήσεις τους, μπορεί να ελέγξει για παραβιάσεις στους χρονικούς περιορισμούς που αναφέρθηκαν.

Ο χρόνος μέχρι τον οποίο μία παραβίαση αποφεύγεται ονομάζεται slack. Ένα θετικό slack s σε έναν κόμβο δείχνει πως ο χρόνος που φτάνει το σήμα στον κόμβο μπορεί να αυξηθεί κατά s χωρίς να επηρεαστεί η συνολική καθυστέρηση του κυκλώματος. Ένα αρνητικό slack δείχνει πως ένα μονοπάτι είναι τόσο αργό, και πρέπει να γίνει πιο γρήγορο αν θέλουμε το κύκλωμα να δουλέψει στην επιθυμητή ταχύτητα. Ένα μηδενικό slack δείχνει πως η παραβίαση έχει ικανοποιηθεί.

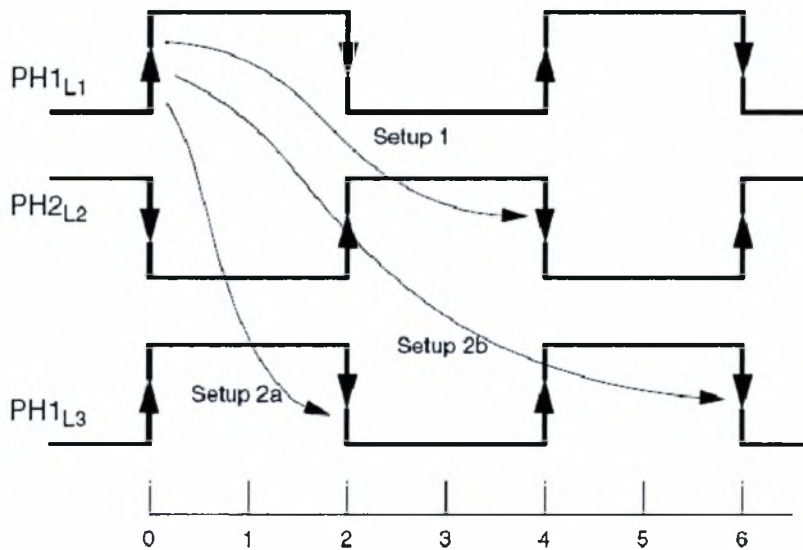
Το NanoTime συνήθως χρησιμοποιείται για την ανάλυση ολοκληρωμένων κυκλωμάτων υψηλής απόδοσης, όπως είναι οι μικροεπεξεργαστές και τα DSP τσιπς, τα οποία χρησιμοποιούν latches που χρονίζονται στην ακμή του ρολογιού. Σχεδιάσεις που στηρίζονται σε latches, μπορούν να χρησιμοποιούν πολλά ρολόγια με διαφορετικές φάσεις, έτσι ώστε να επιτυγχάνεται σωστός έλεγχος των καταχωρητών στο μονοπάτι δεδομένων.

Για παράδειγμα, θεωρήστε το δύο-φάσεων μονοπάτι που στηρίζεται σε latches και φαίνεται στην εικόνα 1.4. Και τα τρία latches είναι level-sensitive και η πύλη ενεργοποιείται όταν η είσοδος G είναι '1'. Τα L1 και L3 ελέγχονται από το PH1 και το L2 ελέγχεται από το PH2. Μία ακμή που ανεβαίνει ελευθερώνει δεδομένα από την έξοδο του latch και μία ακμή που κατεβαίνει εισάγει δεδομένα στην είσοδο του latch. Σε αυτό το παράδειγμα, θεωρούμε πως οι χρόνοι setup και delay είναι μηδέν.



Εικόνα 1.4 Μονοπάτι που βασίζεται σε latches

Η εικόνα 1.5 δείχνει πως το NanoTime κάνει έλεγχο για περιορισμούς setup ανάμεσα στα latches. Για το μονοπάτι από το L1 στο L2, η ακμή που ανεβαίνει από το PH1 ελευθερώνει τα δεδομένα. Αυτά πρέπει να φτάσουν στο L2 πριν την ακμή που πέφτει (και ολοκληρώνεται ο παλμός) από το PH2 τη χρονική στιγμή 4. Ο χρόνος που απαιτείται ονομάζεται Setup1.



Εικόνα 1.5 Έλεγχος για setup περιορισμό σε Μονοπάτι που βασίζεται σε latches

Το NanoTime υποθέτει πως το μονοπάτι από το L1 στο L3, μέσω του L2 συμβαίνει σε έναν κύκλο του ρολογιού. Γι' αυτό, ελέγχει για την άφιξη των δεδομένων στον προηγούμενο παλμό (εκεί που ολοκληρώνεται) του L3, τη χρονική στιγμή 2. Στην εικόνα αυτό ονομάζεται Setup2a. Είναι ένας ακόμη πιο περιοριστικός έλεγχος από το μονοπάτι από το L1 στο L2. Αν το L3 έχει σχεδιαστεί για να λαμβάνει δεδομένα στον επόμενο κύκλο του ρολογιού και όχι στον ίδιο, πρέπει να ελεγχθεί η συμπεριφορά στον επόμενο κύκλο. Σε αυτή την περίπτωση, το NanoTime ελέγχει για την άφιξη των δεδομένων στον επόμενο παλμό (εκεί που ολοκληρώνεται), τη χρονική στιγμή 6. Στην εικόνα αυτό ονομάζεται Setup2b.

1.4 Προσδιορισμός των καθυστερήσεων λογικής πύλης, μονοπατιού και ολόκληρου του κυκλώματος

Η ενεργός αντίσταση αγωγής ενός τρανζίστορ pmos δίνεται από τον τύπο

$$\bar{R} = \left[\frac{\mu_n \epsilon_{ox}}{t_{ox}} \frac{W}{L} \frac{(V_s - V_T)^a}{V_s} \right]^{-1} = \left[\frac{\mu_n \epsilon_{ox}}{t_{ox}} \frac{W}{L} \frac{(V_{DD} - V_d - V_T)^a}{V_{DD} - V_d} \right]^{-1} \quad (1.1)$$

και ενός τρανζίστορ nmos από τον τύπο

$$\bar{R} = \left[\frac{\mu_p \epsilon_{ox}}{t_{ox}} \frac{W}{L} \frac{(V_s - V_T)^a}{V_s} \right]^{-1} = \left[\frac{\mu_p \epsilon_{ox}}{t_{ox}} \frac{W}{L} \frac{(V_{DD} - V_d - V_T)^a}{V_{DD} - V_d} \right]^{-1} \quad (1.2)$$

όπου

- V_s : η τάση με την οποία τροφοδοτούμε την πύλη
- V_d : η πτώση τάσης που εμφανίζεται στην πύλη, σε σχέση με την τάση τροφοδοσίας ($V_d = V_{DD} - V_s$)
- a : μία σταθερά με τιμή μεταξύ του 1.3 και του 2.

Όταν η χρονική διάρκεια που το τρανζίστορ βρίσκεται στην γραμμική περιοχή είναι μεγαλύτερη, η τιμή της σταθεράς πλησιάζει το 1.3, ενώ αν είναι μεγαλύτερη όταν βρίσκεται στην περιοχή του κόρου πλησιάζει το 2.

Αυτό το μοντέλο ονομάζεται alpha-power [9] και σύμφωνα μ' αυτό το μοντέλο, η καθυστέρηση προκύπτει ως συνάρτηση της πτώσης τάσης που εμφανίζεται σε μία πύλη. Από αυτό το σημείο και στο εξής θα θεωρούμε πως $V_i \equiv V_{di}$, δηλαδή με V_i θα αναφερόμαστε στην πτώση τάσης.

Η καθυστέρηση μιας λογικής πύλης i από είσοδο k είναι ίση με την καθυστέρηση Elmore ενός RC δέντρου και δίνεται από την παρακάτω σχέση

$$D_{ik}(V_i) = \sum_{j \in \text{gate_path}(k)} R_j(V_i) C_j^{\text{tot}} \quad (1.3)$$

όπου

- V_i : η πτώση τάσης που εμφανίζεται στην πύλη i , όπως προκύπτει από την ανάλυση του δικτύου τροφοδοσίας

- $gate_path(k)$: το μονοπάτι μεγαλύτερης αντίστασης από την τροφοδοσία (αν πρόκειται για καθυστέρηση ανόδου) ή από την γείωση (αν πρόκειται για καθυστέρηση καθόδου) μέχρι την έξοδο της πύλης, το οποίο διέρχεται από την είσοδο k
- $R_j(V_i)$: η συνολική ενεργός αντίσταση από την τροφοδοσία ή την γείωση μέχρι τον κόμβο j που ανήκει στο μονοπάτι $gate_path(k)$
- C_j^{tot} : η συνολική χωρητικότητα που βρίσκεται στην κατεύθυνση ρεύματος (downstream) από τον κόμβο j που ανήκει στο μονοπάτι $gate_path(k)$

Οι αντιστάσεις $R_j(V_i)$ εφόσον πρόκειται για ενεργούς αντιστάσεις τρανζίστορ, είναι συναρτήσεις της πτώσης τάσης που εμφανίζεται στην πύλη.

Σε ένα συγκεκριμένο μονοπάτι πυλών, η καθυστέρησή του (ανόδου ή καθόδου) είναι το άθροισμα των καθυστερήσεων των πυλών που το απαρτίζουν. Έχουμε εναλλάξ την καθυστέρηση ανόδου και καθόδου καθώς όλες οι πύλες είναι αναστρέφουσες, δηλαδή μία μετάβαση ανόδου στην είσοδο, μπορεί μόνο να δώσει μία μετάβαση καθόδου στην έξοδο και αντιστρόφως. Τελικά, είναι ίση με

$$D_{circuit_path}(\mathbf{V}) = \sum_{i \in circuit_path} D_{ik}(V_i) = (D_1(V_1) + D_2(V_2) + \dots + D_{N_path}(V_{N_path})) \quad (1.4)$$

όπου

- $D_{ik}(V_i)$: η καθυστέρηση της πύλης i από την είσοδο k_onPath που συμμετέχει στο $circuit_path$ μονοπάτι.
- N_path : ο συνολικός αριθμός πυλών μονοπατιού ($circuit_path$)
- \mathbf{V} : το διάνυσμα των πτώσεων τάσης τροφοδοσίας

Τελικά, η συνολική καθυστέρηση του κυκλώματος είναι η μέγιστη από τις καθυστερήσεις όλων των κυκλωματικών μονοπατιών και είναι ίση με

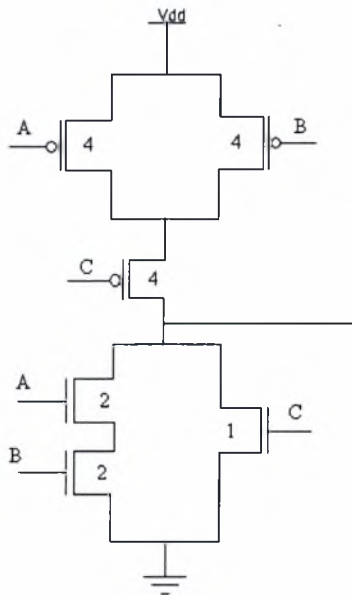
$$D_{circuit}(\mathbf{V}) = \max\{D_{circuit_paths}(\mathbf{V})\} \quad (1.5)$$

Στη συνέχεια, θα παρουσιάσουμε ένα παράδειγμα για να γίνουν πιο κατανοητές όλες οι παραπάνω εξισώσεις. Σ' αυτό, θα υπολογίζεται η καθυστέρηση μίας πύλης, στη

συνέχεια η καθυστέρηση ενός μονοπατιού που θα ανήκει η συγκεκριμένη πύλη και τέλος, η συνολική καθυστέρηση του κυκλώματος.

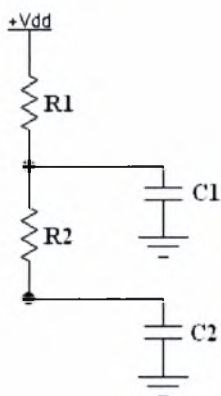
Παράδειγμα

Έχουμε την πύλη $(AB+C)$, της οποίας η υλοποίηση με τρανζίστορ είναι η παρακάτω



Για τον υπολογισμό των καθυστερήσεων χρησιμοποιούμε τυποποιημένα μεγέθη.

Για την είσοδο A το μονοπάτι μέγιστης αντίστασης από την τροφοδοσία μέχρι την έξοδο, που διέρχεται από την A, αποτελείται από τα pmos τρανζίστορ A και C. Αν αυτά αντικατασταθούν από τις ενεργούς αντιστάσεις τους προκύπτει το παρακάτω δέντρο RC:



Ορίζουμε σαν C_{fanout} την χωρητικότητα που οδηγεί στην έξοδό της η πύλη. Οι τιμές των υπόλοιπων μεγεθών φαίνονται στον πίνακα 1.1.

Οι τιμές των αντιστάσεων R_1 και R_2 και των χωρητικοτήτων C_1 και C_2 υπολογίζονται ως εξής :

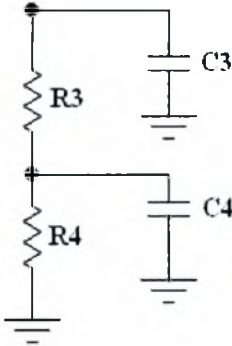
$$R_1 = R_2 = \left[\frac{\mu_p \epsilon_{ox} W (V_s - V_T)^a}{t_{ox} L V_s} \right]^{-1} = \left[K'_p \frac{4W_{min} (V_{DD} - V_d - V_T)^a}{L V_{DD} - V_d} \right]^{-1} =$$

$$= \left[30 * 10^{-6} \frac{2}{0.25} \frac{(1 - V_d - 0.4)^a}{1 - V_d} \right]^{-1} = \left[24 * 10^{-5} \frac{(0.6 - V_d)^a}{1 - V_d} \right]^{-1} \Omega$$

$$C_1 = 12W_{min} \bar{C} = 12 * 0.5 * 2 = 12 fF$$

$$C_2 = 7W_{min} \bar{C} + 4W_{min} \bar{C} + C_{fanout} = 7 * 0.5 * 2 + 4 * 0.5 * 2 + C_{fanout} = 11 fF + C_{fanout}$$

Παρόμοια, για την είσοδο A το μονοπάτι μέγιστης αντίστασης από τη γείωση μέχρι την έξοδο, που διέρχεται από την A, αποτελείται από τα προς τρανζίστορ A και B. Αν αυτά αντικατασταθούν από τις ενεργούς αντιστάσεις τους προκύπτει το παρακάτω δέντρο RC:



Οι τιμές των αντιστάσεων R_3 και R_4 και των χωρητικοτήτων C_3 και C_4 υπολογίζονται ως εξής :

$$R_3 = R_4 = \left[\frac{\mu_n \epsilon_{ox} W (V_s - V_T)^a}{t_{ox} L V_s} \right]^{-1} = \left[K'_n \frac{2W_{min} (V_{DD} - V_d - V_T)^a}{L V_{DD} - V_d} \right]^{-1} =$$

$$= \left[115 * 10^{-6} \frac{1}{0.25} \frac{(1 - V_d - 0.4)^a}{1 - V_d} \right]^{-1} = \left[46 * 10^{-5} \frac{(0.6 - V_d)^a}{1 - V_d} \right]^{-1} \Omega$$

$$C_3 = 7W_{min} \bar{C} + 12W_{min} \bar{C} + C_{fanout} = 7 * 0.5 * 2 + 12 * 0.5 * 2 + C_{fanout} = 19 fF + C_{fanout}$$

$$C_4 = 4W_{min} \bar{C} = 4 * 0.5 * 2 = 4 fF$$

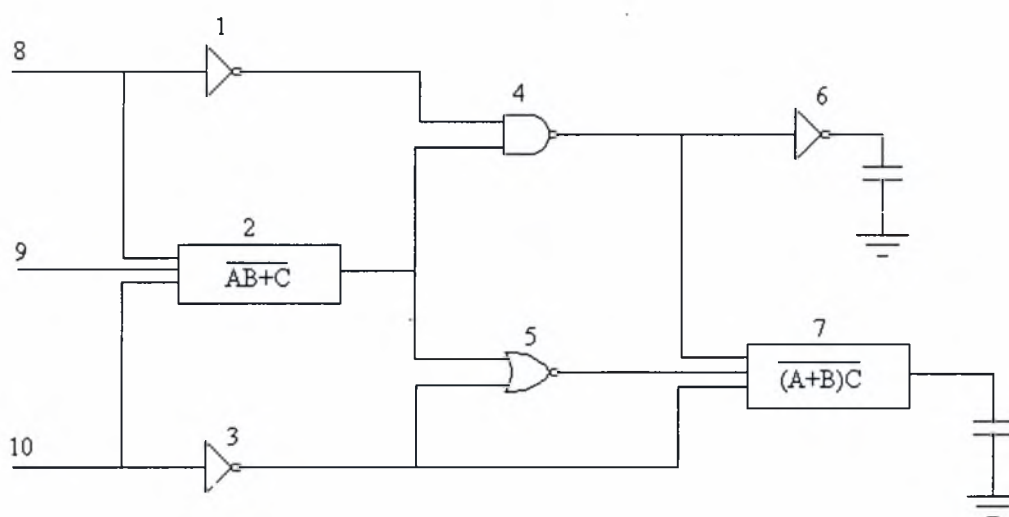
Τελικά, από την εξίσωση (1.3) προκύπτει πως η καθυστέρηση ανόδου της πύλης από την είσοδο A είναι:

$$D_{(AB+C),A} = R_1 C_1 + (R_1 + R_2) C_2$$

και η καθυστέρηση καθόδου από την ίδια είσοδο είναι:

$$D_{(AB+C),A} = (R_3 + R_4) C_3 + R_4 C_4$$

Έστω πως έχουμε το παρακάτω κύκλωμα, στο οποίο όπως φαίνεται υπάρχει και η πύλη της οποίας την καθυστέρηση έχουμε ήδη υπολογίσει (είναι η πύλη 2)



Η καθυστέρηση του μονοπατιού, που αποτελείται από τις πύλες 2,4 και 6, με βάση και την εξίσωση (1.2) είναι η εξής:

$$D_{circuit_path}(V) = D_2(V_2) + D_4(V_4) + D_6(V_6)$$

Όπου $D_2(V_2)$ είναι η καθυστέρηση της πύλης 2 και έστω πως με V_2 συμβολίζουμε την τάση με την οποία τροφοδοτούμε την πύλη 2. Σαν $D_4(V_4)$ ορίζεται η καθυστέρηση της πύλης 4 και σαν $D_6(V_6)$ η καθυστέρηση της πύλης 6. Με V_4 και V_6 ορίζουμε την τάση με την οποία τροφοδοτούμε τις πύλες 4 και 6, αντίστοιχα.

Όπως αναφέραμε και πιο πάνω, η καθυστέρηση D του κυκλώματος είναι η μέγιστη όλων των μονοπατιών. Δηλαδή για το συγκεκριμένο κύκλωμα και με βάση την εξίσωση (1.3) θα είναι ίση με:

$$D_{circuit}(V) = \max\{D_1(V_1) + D_4(V_4) + D_6(V_6), D_1(V_1) + D_4(V_4) + D_7(V_7), D_2(V_2) + D_4(V_4) + D_6(V_6) \\ D_2(V_2) + D_4(V_4) + D_7(V_7), D_2(V_2) + D_5(V_5) + D_7(V_7) \\ D_3(V_3) + D_5(V_5) + D_6(V_6), D_3(V_3) + D_7(V_7)\}$$

Παρατηρούμε πως το κύκλωμά μας, αν και με μόνο 7 πύλες, έχει 7 πιθανά μονοπάτια. Με την προσθήκη μίας μόνο πύλης, ο αριθμός των μονοπατιών αυξάνεται εκθετικά. Έτσι, δεν είναι πρακτικό να υπολογίζουμε την καθυστέρηση D δημιουργώντας μία έκφραση στην οποία θα απαριθμούνται όλα τα μονοπάτια. Είναι πολύ δύσκολο, αν όχι αδύνατο, κάτι τέτοιο να γίνει με το χέρι. Γι' αυτό, χρησιμοποιούμε τον αλγόριθμο PERT (Program or Project Evaluation and Review Technique), ο οποίος είναι πολύ διαδεδομένος για στατική ανάλυση χρονισμού. Βέβαια, στην πραγματικότητα, το όνομα PERT δεν είναι σωστό και ο αλγόριθμος που αναφέρεται σε πολλά έγγραφα σχετικά με την ανάλυση χρονισμού σαν PERT, είναι ο αλγόριθμος κρίσιμου μονοπατιού (CPM – Critical Path Method), που χρησιμοποιείται ευρέως στο project management.

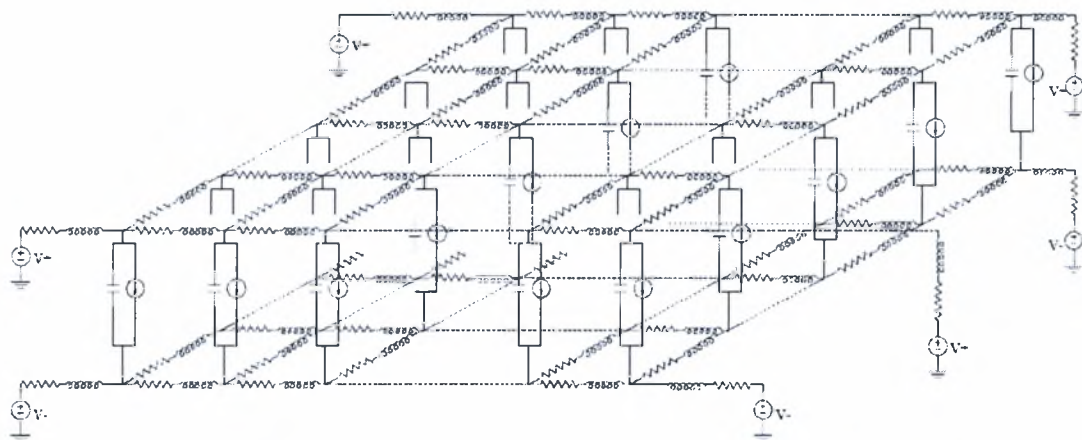
Πίνακας 1.1 Τιμές για τεχνολογία 0.25nm

Τεχνολογία 0.25nm						
	$K_n'/K_p' (A/V^2)$	L (um)	W_{min} (um)	\bar{C} (fF/um)	V_T (V)	V_{DD} (V)
nmos	$K_n' \equiv \frac{\mu_n \epsilon_{ox}}{t_{ox}} = 115 * 10^{-6}$	0.25	0.5	2	0.4	1
pmos	$K_p' \equiv \frac{\mu_p \epsilon_{ox}}{t_{ox}} = 30 * 10^{-6}$	0.25	0.5	2	0.4	1

2. Ανάλυση του δικτύου διανομής ισχύος

2.1 Μέθοδος των κόμβων (Nodal Analysis - NA)

Η μεταφορά ισχύος προς τις πύλες ή τα blocks πυλών γίνεται με μεταλλικούς αγωγούς, οι οποίοι είναι μη ιδανικοί καθώς παρουσιάζουν αντίσταση (στη διέλευση του ρεύματος), χωρητικότητα (μεταξύ τους και με το υπόστρωμα) και σε μεγάλες συχνότητες φαινόμενα αυτεπαγωγής και αμοιβαίας επαγωγής (λόγω των βρόχων ρεύματος που σχηματίζονται).



Εικόνα 2.1 Μοντέλο πλήρους δικτύου διανομής ισχύος 2 επιπέδων

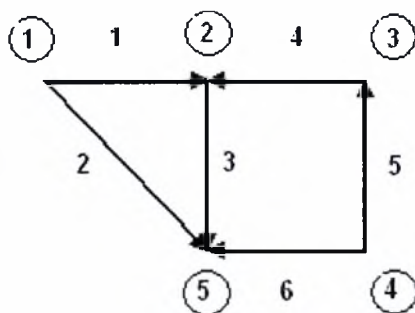
Το δίκτυο τροφοδοσίας αναπαρίσταται από ένα γραμμικό RLC μοντέλο (εικόνα 2.1) με β κλάδους R και L και χωρητικότητες προς τη γη σε κάθε κόμβο. Οι τύποι υπολογισμού των στοιχείων R, L και C είναι:

- αντίσταση κλάδου $R_j = R_{sh} \frac{l_j}{w_j}$, όπου R_{sh} η αντίσταση φύλλου (sheet resistance), l_j μήκος και w_j πλάτος του κλάδου j
- αυτεπαγωγή κλάδου $L_j = l_j \frac{\mu_0}{2\pi} (\ln 8h - \ln w_j)$, όπου μ_0 μαγνητική σταθερά του κενού και h η απόσταση από το υπόστρωμα

- χωρητικότητα προς τη γη $C_i = \sum_j (\frac{1}{2} \bar{C}_{pp(ar)} l_{ji} w_{ji} + \frac{1}{2} \bar{C}_{ff+cp(per)} l_{ji})$, όπου j οι κλάδοι που προσπίπτουν στον κόμβο i , $\bar{C}_{pp(ar)}$ χωρητικότητα παραλλήλων πλακών, $\bar{C}_{ff+cp(per)}$ περιμετρικές πλευρικές χωρητικότητες. Στους κόμβους στους οποίους βάζουμε τις πηγές τροφοδοσίας πρέπει να προσθέσουμε χωρητικότητα αποσύζευξης \bar{C}_{decoup} καθώς και χωρητικότητα ακροδεκτών \bar{C}_{pin} .

Οι κόμβοι του δικτύου, έστω n , χωρίζονται σε p κόμβους οι οποίοι δέχονται τάση τροφοδοσίας V_{dd} προς τη γη (οι κόμβοι αυτοί δεν είναι μόνο στην περιφέρεια λόγω της σύγχρονης C4 τεχνολογίας) και σε q εσωτερικούς κόμβους, από τους οποίους οι sn (με $sn \ll q$ συνήθως) συνδέονται σε πηγές ρεύματος (προς τη γη), που αντιστοιχούν στα blocks πυλών που τροφοδοτούνται και αντλούν ρεύμα από το δίκτυο διανομής ισχύος (οι μεμονωμένες πύλες-cells εντός των blocks, οι οποίες διατάσσονται σε οριζόντιες σειρές και τροφοδοτούνται από οριζόντιες γραμμές τροφοδοσίας/γείωσης σε κάποιο χαμηλό επίπεδο μετάλλου, βλέπουν χαμηλές πτώσεις τάσης σε σχέση με τις τάσεις που δέχονται τα blocks, λόγω μικρών ρευμάτων, οπότε συνήθως δεν εξετάζονται).

Θα γίνει μια σύντομη περιγραφή της μεθόδου των κόμβων, η οποία θα χρησιμοποιηθεί για την ανάλυση του δικτύου τροφοδοσίας. Η τοπολογία του κυκλώματος αναπαρίσταται με τη βοήθεια γράφων.



Εικόνα 2.2 Συνεκτικός προσανατολισμένος γράφος

2.1.1 Νόμος των ρευμάτων του Kirchhoff

Θεωρούμε το προσανατολισμένο συνεκτικό γράφημα της εικόνας 2.2, που αποτελείται από $n=5$ κόμβους και $\beta=6$ κλάδους. Σύμφωνα με τον νόμο των ρευμάτων το άθροισμα όλων των στιγμιαίων ρευμάτων, που φθάνουν ή φεύγουν από κάθε κόμβο του κυκλώματος, είναι κάθε χρονική στιγμή μηδέν. Θεωρώντας θετική τη φορά των ρευμάτων που φεύγουν από κάθε κόμβο, οι εξισώσεις κόμβων του γραφήματος της εικόνας 2.2 σε γενική μορφή είναι

$$\mathbf{A} * \mathbf{i}_{branches}(t) = 0 \quad (2.1)$$

Ο πίνακας \mathbf{A} λέγεται πίνακας συνδέσεων κόμβων (node-to-branch incidence matrix) και δείχνει τον τρόπο με τον οποίο συνδέονται οι κόμβοι του κυκλώματος μεταξύ τους. Ο πίνακας \mathbf{A} έχει n γραμμές και β στήλες. Τα στοιχεία \mathbf{A}_{ij} του πίνακα ορίζονται με τον παρακάτω τρόπο:

$$\mathbf{A}_{ij} = \begin{cases} +1, & \text{όταν ο κλάδος } j \text{ φεύγει από τον κόμβο } i \\ -1, & \text{όταν ο κλάδος } j \text{ φθάνει στον κόμβο } i \\ 0, & \text{αλλιώς} \end{cases}$$

Επειδή κάθε κλάδος του γραφήματος συνδέεται μόνο με δύο κόμβους, κάθε στήλη του πίνακα συνδέσεων κόμβων περιέχει ένα +1, ένα -1 και τα υπόλοιπα στοιχεία είναι μηδέν. Ο πίνακας συνδέσεων κόμβων είναι μια τυπική μέθοδος για την περιγραφή ενός κυκλώματος στον υπολογιστή.

Το σύστημα των Εξ.(2.1) αποτελείται από n εξισώσεις, που είναι γραμμικά εξαρτημένες μεταξύ τους. Από το σύνολο των n εξισώσεων μόνο κάθε συνδυασμός $n-1$ εξισώσεων είναι γραμμικά ανεξάρτητες μεταξύ τους. Αν ορίσουμε τον κόμβο αναφοράς, τότε η γραμμή του πίνακα που αντιστοιχεί σ' αυτόν τον κόμβο δεν είναι απαραίτητη για την επίλυση του συστήματος των εξισώσεων. Η γραμμή αυτή απαλείφεται και παραμένουν μόνο οι $n-1$ γραμμικά ανεξάρτητες εξισώσεις. Αν στο γράφημα της εικόνας 2.2 θεωρήσουμε κόμβο αναφοράς τον κόμβο 5, τότε η Εξ.(2.1) γίνεται:

$$\mathbf{A}_l * \mathbf{i}_{branches}(t) = 0 \quad (2.2)$$

Όπου ο πίνακας \mathbf{A}_l λέγεται ελαττωμένος πίνακας συνδέσεων κόμβων (reduced node-to-branch incidence matrix) και είναι τάξης $(n \times \beta)$.

2.1.2 Σχέση ανάμεσα στις τάσεις των κόμβων και των κλάδων (έκφραση νόμου τάσεων του Kirchhoff)

Με τη μέθοδο των κόμβων προσδιορίζονται οι τάσεις των κόμβων ως προς τον κόμβο αναφοράς. Τις τάσεις των κόμβων τις συμβολίζουμε με $v_{nodes_i}(t)$ ($i = 1, 2, \dots, n-1$). Επειδή μας ενδιαφέρουν οι τάσεις των κλάδων $v_{branches_k}(t)$ ($k = 1, 2, \dots, \beta$), χρειαζόμαστε μια σχέση ανάμεσα σ' αυτές και στις τάσεις των κόμβων.

Θεωρούμε πάλι την οικογένεια κυκλωμάτων, που έχει ως γράφημα αυτό της εικόνας 2.2. Είναι φανερό ότι κάθε κλάδος που συνδέεται με τον κόμβο αναφοράς θα έχει τάση κλάδου ίση με την τάση του κόμβου ή την αντίθετή της. Μ' άλλα λόγια, σ' αυτήν την περίπτωση είναι

$$v_{bk}(t) = \begin{cases} v_{ni}(t), & \text{όταν ο κλάδος } k \text{ φεύγει από τον κόμβο } i \\ -v_{ni}(t), & \text{όταν ο κλάδος } k \text{ φθάνει στον κόμβο } i \end{cases}$$

Στην περίπτωση που ο κλάδος βρίσκεται μεταξύ των κόμβων i και j έχουμε

$$v_{bk}(t) = \begin{cases} v_{ni}(t) - v_{nj}(t), & \text{όταν ο κλάδος } k \text{ φεύγει από τον κόμβο } i \\ v_{nj}(t) - v_{ni}(t), & \text{όταν ο κλάδος } k \text{ φεύγει από τον κόμβο } j \end{cases}$$

Από τα παραπάνω γίνεται φανερό ότι οι τάσεις των κλάδων $v_{bk}(t)$ είναι γραμμικός συνδυασμός των τάσεων κόμβων $v_{ni}(t)$. Αναφερόμενοι στο γράφημα της εικόνας 2.2 έχουμε

$$\mathbf{v}_{branches}(t) = \mathbf{A}_l^T \mathbf{v}_{nodes}(t) \quad (2.3)$$

Η Εξ.(2.3) συνδέει τις τάσεις των κόμβων με τις τάσεις των κλάδων και αποτελεί μια ισοδύναμη έκφραση του νόμου των κόμβων, ενώ μαζί με την Εξ.(2.2) αποτελούν τις βασικές εξισώσεις της μεθόδου των κόμβων.

2.1.3 Σχέση ρεύματος – τάσης (i-v) μεταξύ των κλάδων του κυκλώματος

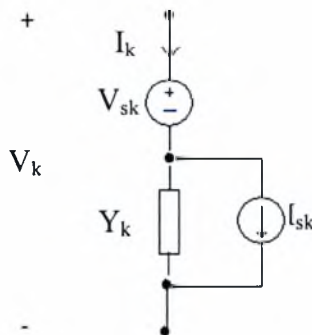
Η εξίσωση κάθε κλάδου είναι:

$$i_{bk}(t) = Y_k(D)v_{bk}(t) \quad (2.4)$$

όπου ($k = 1, 2, \dots, \beta$) και β το πλήθος των κλάδων του κυκλώματος.

$$Y_k(D) = \begin{cases} \frac{1}{R_k}, & \text{αν ο κλάδος έχει αντίσταση} \\ \frac{1}{L_k} \int_0^t \bullet dt \equiv \frac{1}{L_k} D^{-1}, & \text{αν ο κλάδος έχει αυτεπαγωγή} \\ C_k \frac{d}{dt} \bullet \equiv C_k D, & \text{αν ο κλάδος έχει χωρητικότητα} \end{cases} \quad (2.5)$$

Η γενικότητα της μεθόδου ανάλυσης, που επιχειρούμε ν' αναπτύξουμε, επιβάλλει τη θεώρηση κλάδων με όσο γίνεται γενικότερη δομή, όπως είναι η δομή που παρουσιάζεται στην εικόνα 2.3.



Εικόνα 2.3 Γενικός κλάδος κυκλώματος

Η εξίσωση του γενικού κλάδου της εικόνας 2.3 είναι

$$i_{bk}(t) = I_{sk}(t) - Y_k(D)V_{sk}(t) + Y_k(D)v_{bk}(t) \quad (2.6)$$

Όπου ($k = 1, 2, \dots, \beta$) και β το πλήθος των κλάδων του κυκλώματος.

Οι εξισώσεις των γενικών κλάδων του κυκλώματος σχηματίζουν ένα γραμμικό σύστημα β εξισώσεων, που συνδέουν τα ρεύματα των κλάδων με τις τάσεις των κλάδων. Με μορφή πινάκων η Εξ.(2.6) γράφεται

$$\mathbf{i}_{branches}(t) = \mathbf{I}_s(t) - \mathbf{Y}_{branches}(D)\mathbf{V}_s(t) + \mathbf{Y}_{branches}(D)\mathbf{v}_{branches}(t) \quad (2.7)$$

όπου $\mathbf{V}_s(t)$ και $\mathbf{I}_s(t)$ είναι τα διανύσματα στήλης των πηγών διαστάσεων ($\beta \times 1$)

$$\mathbf{V}_s(t) = [V_{s1}(t) V_{s2}(t) \dots V_{s\beta}(t)]^T$$

και (2.8)

$$\mathbf{I}_s(t) = [I_{s1}(t) I_{s2}(t) \dots I_{s\beta}(t)]^T$$

$\mathbf{v}_{branches}(t)$ και $\mathbf{i}_{branches}(t)$ είναι τα διανύσματα στήλης των τάσεων και των ρευμάτων κλάδων διαστάσεων ($\beta \times 1$)

$$\mathbf{v}_{branches}(t) = [v_{branches1}(t) v_{branches2}(t) \dots v_{branches\beta}(t)]^T$$

και (2.9)

$$\mathbf{i}_{branches}(t) = [i_{branches1}(t) i_{branches2}(t) \dots i_{branches\beta}(t)]^T$$

και $\mathbf{Y}_{branches}(D)$ τετραγωνικός διαγώνιος πίνακας τάξης ($\beta \times \beta$)

$$\mathbf{Y}_{branches}(D) = \text{diag}[Y_1(D) Y_2(D) \dots Y_\beta(D)]$$

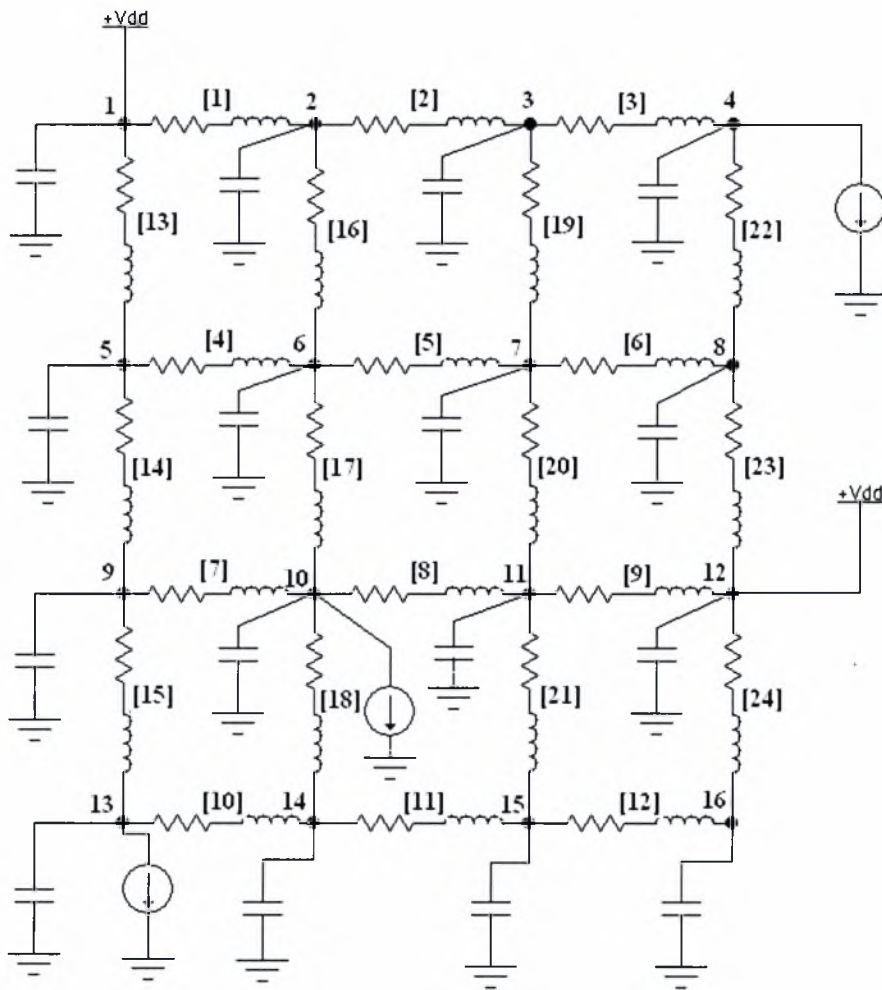
Ο πίνακας $\mathbf{Y}_{branches}(D)$, λέγεται πίνακας σύνθετων αγωγιμοτήτων κλάδων (branch admittance matrix).

Η μέθοδος που αναπτύχθηκε παραπάνω είναι γενική και, όπως αναφέραμε, προσφέρεται για την ανάλυση κυκλωμάτων με τη βοήθεια του υπολογιστή. Όμως, η μέθοδος αυτή είναι αρκετά κοπιαστική, αν πρέπει να εφαρμοστεί σε απλά κυκλώματα με το χέρι. Στις περιπτώσεις απλών κυκλωμάτων γράφουμε απευθείας τις εξισώσεις κόμβων, αφού προηγουμένως μετατρέψουμε τις πηγές τάσεις σε πηγές ρεύματος.

2.2 Τροποποιημένη μέθοδος των κόμβων (Modified Nodal Analysis - MNA)

Η μέθοδος των κόμβων δίνει στη γενική περίπτωση ένα σύστημα ολοκληρωτικό – διαφορικών εξισώσεων (ή ισοδύναμα ένα σύστημα διαφορικών εξισώσεων δεύτερης τάξης). Σημειώνεται ότι εάν οι κυματομορφές των πηγών ρεύματος αντικατασταθούν από στατικές τιμές (οπότε έχουμε ένα DC σύστημα όπου όλοι οι πυκνωτές ανοιχτοκυκλώνονται, τα πηνία βραχυκυκλώνονται και μένουν μόνο οι αντιστάσεις), τότε το σύστημα των ολοκληρωτικο-διαφορικών εξισώσεων ανάγεται σε ένα συνηθισμένο γραμμικό σύστημα και αναγόμαστε στην περίπτωση της στατικής ανάλυσης του δικτύου τροφοδοσίας.

Σε κυκλώματα που περιέχουν μόνο στοιχεία R και C δεν υπάρχουν ολοκληρωτικοί όροι και το σύστημα διαφορικών εξισώσεων που προκύπτει από την μέθοδο των κόμβων, είναι ένα σύστημα διαφορικών εξισώσεων πρώτης τάξης. Αυτό όμως, δεν ισχύει για κυκλώματα που περιέχουν και στοιχεία L, όπου το αντίστοιχο σύστημα ολοκληρωτικο-διαφορικών εξισώσεων ανάγεται σε ένα σύστημα διαφορικών εξισώσεων πρώτης τάξης. Σε αυτή την περίπτωση καταφεύγουμε σε μία τροποποίηση της μεθόδου των κόμβων, η οποία είναι πιο εύκολη στην επίλυση και την επεξεργασία της. Αυτή είναι η τροποποιημένη μέθοδος των κόμβων (στην οποία επιπλέον μεταβλητές είναι τα ρεύματα των κλάδων, εκτός από τις τάσεις των κόμβων), η οποία επιτρέπει τη διαμόρφωση ενός συστήματος διαφορικών εξισώσεων πρώτης τάξης, το οποίο μπορεί πολύ πιο εύκολα να μεταφερθεί στο διακριτό χρόνο για αριθμητική επίλυση και περαιτέρω ανάλυση συμπεριφοράς.



Εικόνα 2.4: Μοντέλο RLC δικτύου ισχύος 4x4

Θα γίνει μια σύντομη αναφορά στις εξισώσεις των RLC κυκλωμάτων που προκύπτουν χρησιμοποιώντας την τροποποιημένη μέθοδο των κόμβων. Το κύκλωμά μας αποτελείται από n κόμβους και β κλάδους (όπου $n=16$ και $\beta=24$ για το κύκλωμα της εικόνας 2.4). Οι κόμβοι παριστάνονται με απλή αρίθμηση, ενώ οι κλάδοι με αρίθμηση μέσα σε αγκύλες [].

Ο πίνακας συνδέσεων κόμβων \mathbf{A} ή πίνακας πρόσπτωσης (incidence matrix) για ένα γραμμικό κύκλωμα ορίζεται με τον παρακάτω τρόπο:

$$\mathbf{A}_{ij} = \begin{cases} +1, & \text{όταν ο κλάδος } j \text{ φεύγει από τον κόμβο } i \\ -1, & \text{όταν ο κλάδος } j \text{ φθάνει στον κόμβο } i \\ 0, & \text{Αλλιώς} \end{cases}$$

Κάθε σειρά του πίνακα συνδέσεων κόμβων \mathbf{A}_y περιέχει μόνο δύο μη μηδενικά στοιχεία (μόνο ένα αν ο κλάδος συνδέεται με τη γη). Ο πίνακας αυτός δείχνει τον τρόπο με τον οποίο συνδέονται οι κόμβοι του κυκλώματος μεταξύ τους και οι νόμοι Kirchhoff που το περιγράφουν είναι οι εξής:

$$\text{KCL :} \quad \mathbf{A} \mathbf{i}_{branches}(t) = \mathbf{I}_s(t), \quad \mathbf{i}_{branches}(t) = \begin{bmatrix} i_1(t) \\ i_2(t) \\ \dots \\ i_\beta(t) \end{bmatrix}$$

όπου $\mathbf{i}_{branches}(t)$ είναι τα ρεύματα των κλάδων και $\mathbf{I}_s(t)$ είναι τα ρεύματα στους κόμβους που συνδέονται με τις ομάδες των πυλών (διεγέρσεις). Το σύστημα των παραπάνω εξισώσεων αποτελείται από n εξισώσεις που είναι γραμμικά εξαρτημένες μεταξύ τους. Από το σύνολο των n εξισώσεων μόνο κάθε συνδυασμός $n-1$ εξισώσεων είναι γραμμικά ανεξάρτητες μεταξύ τους.

$$\text{KVL :} \quad \mathbf{A}^T \mathbf{v}_{nodes}(t) = \mathbf{v}_{branches}(t), \quad \mathbf{v}_{nodes}(t) = \begin{bmatrix} v_1(t) \\ v_2(t) \\ \dots \\ v_n(t) \end{bmatrix}$$

όπου $\mathbf{v}_{branches}(t)$ είναι οι τάσεις των κλάδων και $\mathbf{v}_{nodes}(t)$ είναι οι τάσεις στους κόμβους που βάζουμε τις πηγές τάσης. Το σύστημα των παραπάνω εξισώσεων αποτελείται από $\beta - (n-1) + 1$ εξισώσεις που είναι γραμμικά εξαρτημένες μεταξύ τους. Από το σύνολο των $\beta - (n-1) + 1$ εξισώσεων μόνο κάθε συνδυασμός $\beta - (n-1)$ εξισώσεων είναι γραμμικά ανεξάρτητες μεταξύ τους.

Το δίκτυο διανομής ισχύος (power grid) περιγράφεται από το ακόλουθο σύστημα γραμμικών διαφορικών εξισώσεων:

$$\tilde{\mathbf{G}}\mathbf{x}(t) + \tilde{\mathbf{C}}\mathbf{x}'(t) = \mathbf{I}_s(t) \quad (2.10)$$

όπου $\tilde{\mathbf{G}}$: πίνακας αγωγιμοτήτων του δικτύου,

$\tilde{\mathbf{C}}$: πίνακας χωρητικοτήτων και αυτεπαγωγών,

$\mathbf{x}(t)$: διάνυσμα τάσεων στους κόμβους και ρευμάτων διαμέσου αυτεπαγωγών,
 $\mathbf{I}_s(t)$: διάνυσμα διεγέρσεων.

Πιο συγκεκριμένα

$$\tilde{\mathbf{G}} = \begin{bmatrix} \mathbf{G} & \mathbf{A}_l^T \\ -\mathbf{A}_l & \mathbf{0} \end{bmatrix},$$

$$\tilde{\mathbf{C}} = \begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{L} \end{bmatrix},$$

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{v}_{nodes}(t) \\ \mathbf{i}_{branches}(t) \end{bmatrix}, \quad (2.11)$$

και

$$\mathbf{G} = \mathbf{A}_l^T \mathbf{G} \mathbf{A}_l,$$

$$\mathbf{C} = \mathbf{A}_l^T \mathbf{C} \mathbf{A}_l, \quad (2.12)$$

$$\mathbf{L} = \mathbf{A}_l^T \mathbf{L}^{-1} \mathbf{A}_l$$

Στις Εξ.(2.12), οι πίνακες \mathbf{G} και \mathbf{C} είναι διαγώνιοι και έχουν μέγεθος $n \times n$. Οι πίνακες \mathbf{L} και \mathbf{A}_l έχουν μέγεθος $(\beta+p) \times (\beta+p)$.

Το δίκτυο διανομής ισχύος με το οποίο θα ασχοληθούμε από δω και κάτω αποτελεί μία ειδική περίπτωση ενός πλέγματος από οριζόντιους και κάθετους κλάδους με στοιχεία R και L και πυκνωτές σε όλους τους κόμβους, διεγέρσεις ρεύματος σε μερικούς κόμβους και πηγές ρεύματος σε μερικούς άλλους.

Σημείωση:

1. Για να μπορέσουμε να υπολογίσουμε τον πίνακα \mathbf{A}_l , πρέπει να ορίσουμε μία φορά για το ρεύμα που διαρρέει τους κλάδους. Έστω πως αυτή είναι από αριστερά προς τα δεξιά (\rightarrow) και από πάνω προς τα κάτω (\downarrow).

2.3 Διαμόρφωση του συστήματος της MNA ως προς τις πτώσεις τάσης

Η MNA μέθοδος υπολογίζει τις απόλυτες τιμές των τάσεων σε κάθε κόμβο του δικτύου. Εάν όμως θέλουμε τις πτώσεις τάσης (σε σχέση με την τάση τροφοδοσίας) σε κάθε κόμβο του δικτύου ενεργούμε ως εξής:

Το διάνυσμα διεγέρσεων $\mathbf{I}_s(t)$ γράφεται $\mathbf{I}_s(t) = -\mathbf{i}_s(t) + \mathbf{G}_0 \mathbf{V}_{dd}$, όπου $\mathbf{i}_s(t)$ το διάνυσμα των πηγών ρεύματος και μόνο, με θετικό πρόσημο σε κάθε κόμβο, \mathbf{G}_0 ένας αρρ πίνακας των αγωγιμοτήτων από τους q κόμβους του δικτύου προς τις p τάσεις τροφοδοσίας και \mathbf{V}_{dd} ένα $p \times 1$ διάνυσμα με όλες τις συνιστώσες του ίσες με την τάση τροφοδοσίας.

Τώρα στο σύστημα $\mathbf{G}\mathbf{v}_{nodes}(t) + \mathbf{C}\mathbf{v}'_{nodes}(t) = -\mathbf{i}_s(t) + \mathbf{G}_0 \mathbf{V}_{dd}$ εάν θέσουμε όλες τις διεγέρσεις των πηγών ρεύματος $\mathbf{i}_s(t) = 0$ θα έχουμε σε όλους τους κόμβους $\mathbf{v}_{nodes} = \mathbf{V}_{dd}$ (καθώς σε DC όλοι οι πυκνωτές προς τη γη ανοιχτοκυκλώνονται, ενώ δεν υπάρχει κάποιο άλλο DC μονοπάτι προς τη γη), οπότε προκύπτει $\mathbf{G}\mathbf{V}_{dd} = \mathbf{G}_0 \mathbf{V}_{dd}$.

Εάν αντικαταστήσουμε την τελευταία πίσω στο αρχικό σύστημα έχουμε:

$$\mathbf{G}\mathbf{v}_{nodes}(t) + \mathbf{G}\mathbf{V}'_{nodes}(t) = -\mathbf{i}_s(t) + \mathbf{V}_{dd} \Leftrightarrow$$

$$\mathbf{G}(\mathbf{V}_{dd} - \mathbf{v}_{nodes}(t)) - \mathbf{C}\mathbf{v}'_{nodes}(t) = \mathbf{i}_s(t) \Leftrightarrow$$

$$\mathbf{G}(\mathbf{V}_{dd} - \mathbf{v}_{nodes}(t)) + \mathbf{C}(\mathbf{V}_{dd} - \mathbf{v}_{nodes}(t))' = \mathbf{i}_s(t) \Leftrightarrow$$

$$\mathbf{G}\mathbf{v}_{nodesd}(t) + \mathbf{C}\mathbf{v}_{nodesd}(t) = \mathbf{i}_s(t) ,$$

όπου $\mathbf{v}_{nodesd}(t) = \mathbf{V}_{dd} - \mathbf{v}_{nodes}(t)$ το διάνυσμα των πτώσεων τάσης στους κόμβους του δικτύου.

Εφόσον από δω και πέρα θα ασχοληθούμε μόνο με το τελευταίο σύστημα το οποίο έχει διαμορφωθεί ως προς τις πτώσεις τάσης θα αναφέρουμε το διάνυσμα $\mathbf{i}_s(t)$ των πηγών ρεύματος και μόνο, με θετικό πρόσημο σε κάθε κόμβο ως διάνυσμα διεγέρσεων $\mathbf{I}_s(t)$

2.4. Αριθμητική επίλυση γραμμικών κυκλωμάτων στο διακριτό χρόνο

Επειδή οι κυματομορφές διεγέρσεων (εικόνα 2.5) δεν έχουν γνωστή αναλυτική έκφραση ως συναρτήσεις του χρόνου, αλλά υπολογίζονται μέσω προσομοίωσης του ψηφιακού κυκλώματος που τροφοδοτείται από το δίκτυο (για έναν ή περισσότερους κύκλους ρολογιού), γίνεται αριθμητική επίλυση του συστήματος των διαφορικών εξισώσεων (ΔΕ) με διακριτοποίηση (discretization) της χρονικής μεταβλητής. Συγκεκριμένα, μπορούμε να χρησιμοποιήσουμε για την προσέγγιση της παραγώγου

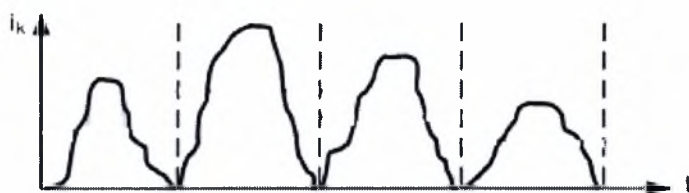
$$\mathbf{x}'(t) \approx \frac{\mathbf{x}(t+h) - \mathbf{x}(t)}{h} \quad \text{Forward Euler (FE),}$$

$$\mathbf{x}'(t) \approx \frac{\mathbf{x}(t) - \mathbf{x}(t-h)}{h} \quad \text{Backward Euler (BE) ή}$$

$$\mathbf{x}'(t) \approx \mathbf{x}'(t-h) + \frac{2(\mathbf{x}(t) - \mathbf{x}(t-h))}{h} \quad \text{Trapezoidal (TR),}$$

όπου στην περίπτωση μας, $\mathbf{x}(t) \equiv \mathbf{v}_{nodes}(t)$.

Σημειώνεται ότι το τελικό μας αποτέλεσμα είναι ανεξάρτητο της επιλογής μεθόδου προσέγγισης.



Εικόνα 2.5 Κυματομορφές διεγέρσεων

2.5 Μέθοδος των κόμβων στο διακριτό χρόνο

Θα παρουσιάσουμε μόνο την περίπτωση που γίνεται χρήση της Backward Euler προσέγγισης.

Αντικαθιστούμε το διάνυσμα παραγώγων $\mathbf{x}'(t)$ με τη διακριτοποιημένη του μορφή

$\mathbf{x}'(t) \approx \frac{\mathbf{x}(t) - \mathbf{x}(t-h)}{h}$ στην Εξ.(2.10), οπότε το σύστημα παίρνει τη μορφή:

$$\left(\tilde{\mathbf{G}} + \frac{\tilde{\mathbf{C}}}{h} \right) \mathbf{x}(t) = \frac{\tilde{\mathbf{C}}}{h} \mathbf{x}(t-h) + \mathbf{I}_s(t) \quad (2.13)$$

Αντικαθιστώντας στην παραπάνω εξίσωση τις Εξ. (2.11) προκύπτουν οι παρακάτω εξισώσεις:

$$\left(\mathbf{G} + \frac{\mathbf{C}}{h} \right) \mathbf{v}_{nodes}(t) + \mathbf{A}_l^T \mathbf{i}_{branches}(t) = \frac{\mathbf{C}}{h} \mathbf{v}_{nodes}(t-h) + \mathbf{I}_s(t) \quad (2.14)$$

$$\frac{\mathbf{L}}{h} \mathbf{i}_{branches}(t) = \frac{\mathbf{L}}{h} \mathbf{i}_{branches}(t-h) + \mathbf{A}_l \mathbf{v}_{nodes}(t) \quad (2.15)$$

Επιλύοντας την Εξ.(3.3) ως προς $\mathbf{i}_{branches}(t)$ έχουμε:

$$\mathbf{i}_{branches}(t) = \mathbf{i}_{branches}(t-h) + h\mathbf{L}^{-1} \mathbf{A}_l \mathbf{v}_{nodes}(t) \quad (2.16)$$

και αντικαθιστώντας την Εξ.(2.16) στην Εξ.(2.14) προκύπτει:

$$\left(\mathbf{G} + \frac{\mathbf{C}}{h} + h\mathbf{A}_l^T \mathbf{L}^{-1} \mathbf{A}_l \right) \mathbf{v}_{nodes}(t) = \frac{\mathbf{C}}{h} \mathbf{v}_{nodes}(t-h) - \mathbf{A}_l^T \mathbf{i}_{branches}(t-h) + \mathbf{I}_s(t) \quad (2.17)$$

που είναι της μορφής $\tilde{\mathbf{A}} \mathbf{v}_{nodes}(t) = \mathbf{b}$,

όπου $\tilde{\mathbf{A}} = \mathbf{G} + \frac{\mathbf{C}}{h} + h\mathbf{L}$,

όπου $\mathbf{L} = \mathbf{A}_l^T \mathbf{L}^{-1} \mathbf{A}_l$

Αν επιλύσουμε την Εξ.(2.17) ως προς $\mathbf{v}_{nodes}(t)$ έχουμε:

$$\begin{aligned} \mathbf{v}_{nodes}(t) = & \left(\mathbf{G} + \frac{\mathbf{C}}{h} + h\mathbf{A}_l^T \mathbf{L}^{-1} \mathbf{A}_l \right)^{-1} \frac{\mathbf{C}}{h} \mathbf{v}_{nodes}(t-h) - \left(\mathbf{G} + \frac{\mathbf{C}}{h} + h\mathbf{A}_l^T \mathbf{L}^{-1} \mathbf{A}_l \right)^{-1} \mathbf{A}_l^T \mathbf{i}_{branches}(t-h) + \\ & + \left(\mathbf{G} + \frac{\mathbf{C}}{h} + h\mathbf{A}_l^T \mathbf{L}^{-1} \mathbf{A}_l \right)^{-1} \mathbf{I}_s(t) \end{aligned} \quad (2.18)$$

Αν θέσουμε ως $\mathbf{A} \equiv h\mathbf{L}^{-1} \mathbf{A}_l$,

$$\mathbf{B} \equiv \left(\mathbf{G} + \frac{\mathbf{C}}{h} + h\mathbf{A}_l^T \mathbf{L}^{-1} \mathbf{A}_l \right)^{-1},$$

$$\mathbf{D} \equiv \mathbf{B} \frac{\mathbf{C}}{h} \text{ και}$$

$$\mathbf{E} \equiv \mathbf{B} \mathbf{A}_l^T$$

οι σχέσεις που μας δίνουν τα $\mathbf{i}_{branches}(t)$ και $\mathbf{v}_{nodes}(t)$ δίνονται από τους παρακάτω τύπους:

$$\mathbf{i}_{branches}(t) = \mathbf{A} \mathbf{v}_{nodes}(t) + \mathbf{i}_{branches}(t-h) \text{ και} \quad (2.19)$$

$$\mathbf{v}_{nodes}(t) = \mathbf{D} \mathbf{v}_{nodes}(t-h) - \mathbf{E} \mathbf{i}_{branches}(t-h) + \mathbf{B} \mathbf{I}_s(t) \quad (2.20)$$

Προκύπτει ένα σύστημα δύο αναδρομικών σχέσεων που δίνει το διάνυσμα των πτώσεων τάσης καθώς και των ρευμάτων κλάδων, δεδομένου διανύσματος διεγέρσεων $\mathbf{I}_s(t)$ στους κόμβους, σε κάθε χρονική στιγμή $t = kh$, $k = 1, \dots, N$ εντός μιας περιόδου ρολογιού T . Σε κάθε βήμα, $t = kh$, ο υπολογισμός των πτώσεων τάσης $\mathbf{v}_{nodes}(t)$ και των ρευμάτων κλάδων $\mathbf{i}_{branches}(t)$ εξαρτάται από τα αντίστοιχα διανύσματα που υπολογίστηκαν στο προηγούμενο βήμα $t = (k-1)h$, καθώς και από το διάνυσμα των διεγέρσεων $\mathbf{I}_s(t)$ στο τρέχον βήμα.

Η μέση πτώση τάσης σε έναν κύκλο ρολογιού περιόδου T , θα είναι προφανώς

$$\bar{v}_{nodes}(t) = \frac{1}{Nh} \sum_{j=0}^N v_{nodes}(j) \quad (2.21)$$

Η μέση πτώση τάσης σε έναν κύκλο ρολογιού είναι εκείνη που έχει σημασία για την αποτίμηση των επιπτώσεων και εκείνη που θα πρέπει να διατηρηθεί εντός ορίων ασφαλείας (συνήθως 10%) προκειμένου να χαρακτηριστεί το δίκτυο ως ευσταθές (robust).

3. Προσδιορισμός των τάσεων μεγιστοποίησης της καθυστέρησης του κυκλώματος

Ορισμός 1:

Μία συνάρτηση πολλών μεταβλητών (ή μία συνάρτηση με μεταβλητή διανύσματος) $f(\mathbf{y})$, ονομάζεται *αυστηρά αύξουσα μονότονη* (*strictly monotone increasing*) στο πεδίο ορισμού της $D \subset \mathbb{R}^n$, αν για κάθε $\mathbf{y}, \mathbf{y}' \in D$, που ικανοποιεί τις συνθήκες $\mathbf{y}' \geq \mathbf{y}$ και $y'_i > y_i$ για τουλάχιστον μία συνιστώσα $i = 1, \dots, n$ ισχύει $f(\mathbf{y}') > f(\mathbf{y})$.

Διαισθητικά, από τον παραπάνω ορισμό καταλαβαίνουμε πως η αύξηση μίας ή περισσότερων μεταβλητών $y_i, i = 1, \dots, n$ του διανύσματος \mathbf{y} μπορεί να έχει μόνο ένα ως αποτέλεσμα, την αύξηση της συνάρτησης $f(\mathbf{y})$. Άρα, αφού η καθυστέρηση είναι αύξουσα συνάρτηση, η αύξηση της πτώσης τάσης V_d σε μία ή περισσότερες πύλες θα έχει σαν αποτέλεσμα την αύξηση της καθυστέρησης του μονοπατιού και κατ' επέκταση την αύξηση της συνολικής καθυστέρησης του κυκλώματος. Για αυτή τη συγκεκριμένη κατηγορία συναρτήσεων, το παραπάνω ισχύει στα maximal points:

Ορισμός 2:

Ένα σημείο $\mathbf{y} \in D$ ονομάζεται *μεγιστικό* (maximal ή noninferior point) σημείο του συνόλου $D \subset \mathbb{R}^n$ αν δεν υπάρχει σημείο $\mathbf{y}' \in D$, τέτοιο ώστε $\mathbf{y}' \geq \mathbf{y}$ με τουλάχιστον μία συνιστώσα $i = 1, \dots, n$ για την οποία να ισχύει $y'_i > y_i$.

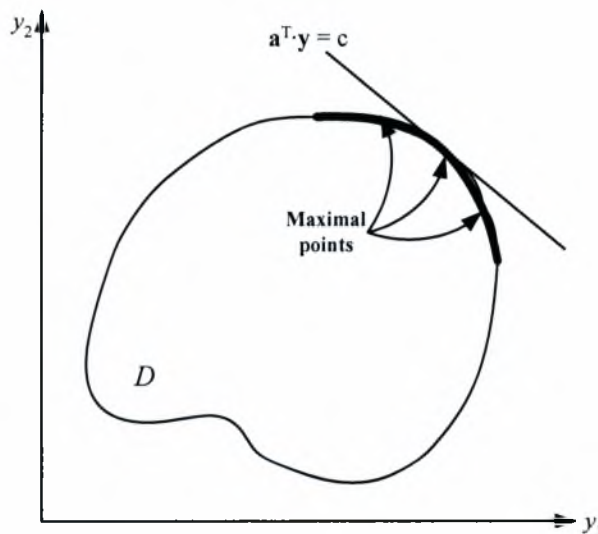
Θεώρημα 1:

Έστω πως η $f(\mathbf{y})$ είναι μία συνεχής και αυστηρά αύξουσα μονότονη συνάρτηση με μία μεταβλητή διανύσματος, η οποία ορίζεται σε ένα φραγμένο σύνολο $D \subset \mathbb{R}^n$. Αν το $\mathbf{y}^* \in D$ είναι ένα σημείο στο οποίο μεγιστοποιείται η τιμή της συνάρτησης $f(\mathbf{y})$ [δηλαδή $f(\mathbf{y}^*) = \max_{\mathbf{y} \in D} f(\mathbf{y})$], τότε το \mathbf{y}^* είναι ένα maximal point του συνόλου D .

Απόδειξη:

Υποθέτουμε πως το σημείο \mathbf{y}^* στο οποίο μεγιστοποιείται η τιμή της συνάρτησης $f(\mathbf{y})$, δεν είναι maximal point του συνόλου D . Τότε θα υπήρχε ένα σημείο $\mathbf{y}' \in D$, τέτοιο ώστε $\mathbf{y}' \geq \mathbf{y}^*$ και $y'_i > y_i^*$ για τουλάχιστον μία συνιστώσα $i = 1, \dots, n$. Ωστόσο, επειδή η $f(\mathbf{y})$ είναι αυστηρά αύξουσα μονότονη στο σύνολο D αυτό θα σήμαινε πως $f(\mathbf{y}') > f(\mathbf{y}^*)$ και άρα το \mathbf{y}^* δεν μπορεί να είναι σημείο στο οποίο μεγιστοποιείται η τιμή της συνάρτησης $f(\mathbf{y})$, το οποίο μας οδηγεί σε αντίφαση-άτοπο.

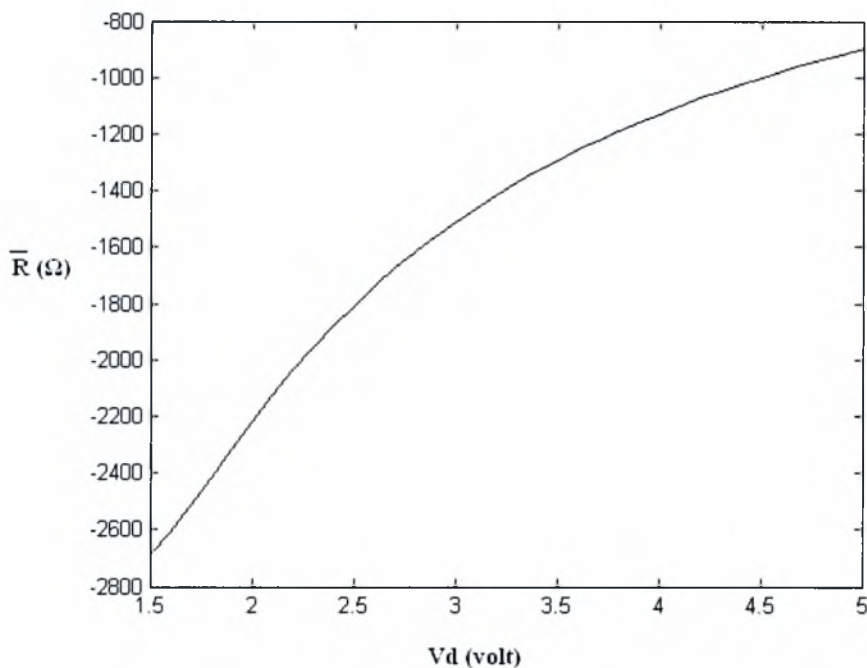
Το παραπάνω θεώρημα σημαίνει, πως προκειμένου να βρούμε τα σημεία που μεγιστοποιείται μία συνεχής και αυστηρά αύξουσα μονότονη συνάρτηση $f(\mathbf{y})$, η οποία ορίζεται σε ένα φραγμένο σύνολο $D \subset \mathbb{R}^n$ πρέπει να περιορίσουμε την έρευνά μας στο υποσύνολο που περιέχει τα maximal points του συνόλου D . (Ο χώρος του οποίου είναι υποσύνολο το σύνολο D είναι \mathbb{R}^n γιατί έχουμε συμβολίσει με n τον αριθμό των πυλών)



Εικόνα 3.1. Maximal points ενός φραγμένου συνόλου $D \subset \mathbb{R}^n$ και μεγιστοποίηση μιας αύξουσας συνάρτησης σε αυτό

Τα σημεία, λοιπόν, αυτά του συνόλου D αποτελούν σημεία – διανύσματα κυματομορφών χειρότερων διεγέρσεων για οποιοδήποτε δίκτυο τροφοδοτεί το συγκεκριμένο ψηφιακό κύκλωμα.

Όπως φαίνεται από την εξ.(1.1) η αντίσταση είναι μία αυστηρά αύξουσα μονότονη συνάρτηση της πτώσης τάσης που εμφανίζεται σε μία πύλη (εικόνα 3.2), άρα το ίδιο είναι και η καθυστέρηση μίας λογικής πύλης, κάτι το οποίο φαίνεται και από την εξ.(1.2). Επίσης, όπως φαίνεται από τις εξισώσεις (1.3) και (1.4), αυστηρά αύξουσα μονότονη συνάρτηση του διανύσματος της πτώσης τάσης είναι η καθυστέρηση ενός μονοπατιού και κατ' επέκτασιν και η συνολική καθυστέρηση του κυκλώματος.



Εικόνα 3.2. Διάγραμμα $R-V_d$, στο οποίο φαίνεται πως η ενεργός αντίσταση είναι μία αυστηρά αύξουσα μονότονη συνάρτηση ως προς την πτώση τάσης.

Βέβαια, το σύνολο D των χειρότερων κυματομορφών διεγέρσεων στην περίπτωση μας, αποτελείται από άπειρα ουσιαστικά σημεία (συγκεκριμένα 4^l , όσα και τα δυνατά δυαδικά διανύσματα εισόδων και οι αντίστοιχοι διαφορετικοί κύκλοι ρολογιού), το οποίο σημαίνει ότι δεν μπορούμε να τα ελέγξουμε όλα για να εντοπίσουμε τα maximal points ανάμεσα τους. Έτσι, καταφεύγουμε σε δειγματοληψία (τυχαία) έστω για έναν αριθμό m κύκλων ρολογιού.

Από το δείγμα αυτό των κυματομορφών υπολογίζονται οι χειρότερες ανάμεσα τους διεγέρσεις ενός δικτύου τροφοδοσίας ως maximal points του συνόλου του δείγματος στο χώρο \mathcal{R}^n . Ο τρόπος υπολογισμού των σημείων αυτών γίνεται βάσει του ορισμού 2. Συγκεκριμένα, για να βρούμε τα maximal points του χώρου \mathcal{S} , που αποτελείται από

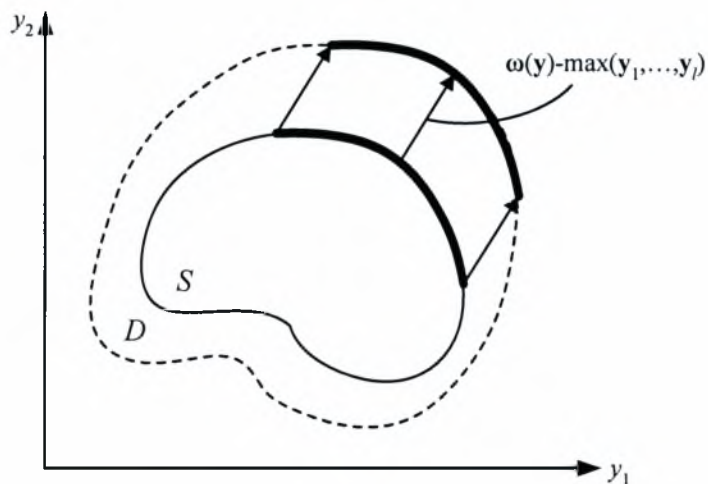
m σημεία πρέπει να συγκρίνουμε κάθε σημείο με όλα τα υπόλοιπα (για να καθορίσουμε αν ένα συγκεκριμένο σημείο δεν κυριαρχείται από κάποιο άλλο σε όλες του τις συνιστώσες), κάτι το οποίο οδηγεί σε m^2 συγκρίσεις συνολικά. Ωστόσο, μπορεί να αποδειχθεί [5] πως οι συγκρίσεις που απαιτούνται μπορούν να μειωθούν το πολύ σε $O(m(\log_2 m)^{n-2})$, όπου n είναι η διάσταση του χώρου και των σημείων που τον απαρτίζουν. Κατόπιν, με βάση στατιστικές τεχνικές θεωρίας ακραίων τιμών (Extreme Value Theory - EVT) [1,6] το σύνολο των χειρότερων αυτών σημείων στο δείγμα μετακινείται μέσα στο χώρο \mathbb{R}^n κατά την αναμενόμενη θέση του ως χειρότερο σημείο στο πλήρες σύνολο D των 4^F σημείων (εικόνα 3.3).

Δοθέντος ενός δείγματος πολλών μεταβλητών S μπορούμε να υπολογίσουμε το άνω όριο ω_k για κάθε άξονα συντεταγμένων $1 \leq i \leq n$ εφαρμόζοντας τα αποτελέσματα που έχουμε βρει, σε κάθε ένα δείγμα S μιας μεταβλητής, του S . Πλέον το σύνολο στοιχείων S είναι μια εικόνα του τρέχοντος χώρου D και σχηματίζει ένα σύνορο για τα εξωτερικά στοιχεία του \mathbb{R}^n . Επίσης έχει από μόνο του ένα σύνολο από maximal points. Το σύνολο του συνόλου S , που αποτελεί και το δειγματοχώρο μας πια, θα συμπεσεί στο σύνολο του χώρου D και συγκεκριμένα, το σύνολο των maximal points του θα προβληθεί σε κάποιο άλλο χώρο για κάθε ξεχωριστή συντεταγμένη $1 \leq i \leq n$ (εικόνα 3.2). Μια πολύ καλή προσέγγιση αυτή της προβολής για τα maximal points σαν σύνολο για κάθε $1 \leq i \leq n$ θα ήταν το $\omega_i - \max_{1 \leq k \leq m} S_{i,k}$ όπου $\max_{1 \leq k \leq m} S_{i,k}$ είναι η μέγιστη τιμή για κάθε σύνολο μιας μεταβλητής S_i (δηλαδή το μέγιστο του χώρου S για κάθε άξονα συντεταγμένων). Γράφοντας αυτό σε συντομία σε μορφή διανύσματος για όλες τις $1 \leq i \leq n$ όπως παρακάτω

$$\omega - \max_{1 \leq k \leq m} S_{i,k}$$

προκύπτει ένα διάνυσμα διαφορών με το οποίο μπορούμε να μετακινήσουμε τα maximal points του D στο \mathbb{R}^n . Τα νέα, μετακινημένα πλέον, maximal points θα επιλεγούν ως στατικά διανύσματα για τη DC ανάλυση με σκοπό να υπολογιστούν τα σημεία όπου έχουμε την μεγαλύτερη πτώση τάσης. Όπως είναι αναμενόμενο το σύνολο των maximal points του D θα έχει πολύ διαφορετική δομή και θα περιλαμβάνει πολλά περισσότερα σημεία σε σχέση με το σύνολο των maximal points του συνόλου S , όμως το μέγιστο σημείο μιας αυστηρά αύξουσας μονότονης

συνάρτησης δεν επηρεάζεται από τη δομή του συνόλου των maximal points αλλά εξαρτάται κυρίως από τη θέση του στο \mathcal{R}^n .



Εικόνα 3.3. Δειγματοχώρος ρευμάτων S και μετακίνηση των maximal points του έτσι ώστε να «συναντήσουν» τα maximal points ολόκληρου του χώρου D .

4. Πειραματικοί έλεγχοι και αποτελέσματα

Για να επιβεβαιώσουμε πως η καθυστέρηση των χειρότερων μονοπατιών της σχεδίασης είναι εντός των επιτρεπτών ορίων (το πολύ μέχρι 10ns που είναι η περίοδος), προσομοιώσαμε το κύκλωμα c6288_3004. Για την προσομοίωση χρησιμοποιήσαμε τα εργαλεία MATLAB (για να υπολογίσουμε την πτώση τάσης) και NanoTime (για να υπολογίσουμε την καθυστέρηση).

Ο αριθμός '3004', μας δείχνει το μέγεθος των blocks στα οποία χωρίζουμε τα ρεύματα που παίρνουμε απ' το αρχείο εισόδου. Το κύκλωμα έχει δίκτυο (grid) διαστάσεων $n1 \times n2$ -vn, όπου $n1$ ο αριθμός των οριζόντιων κόμβων, $n2$ ο αριθμός των κάθετων κόμβων και vn ο αριθμός των πηγών τροφοδοσίας (v_{dd}). Σε κάθε εκτέλεση τα $n1$, $n2$ και vn αλλάζουν έτσι ώστε να μελετήσουμε όσο το δυνατόν περισσότερες περιπτώσεις. Είναι τυχαία η επιλογή των κόμβων στους οποίους τοποθετούνται οι πηγές ρευμάτων και οι πηγές τροφοδοσίας.

Τα κύρια βήματα της διαδικασίας, που έγινε με το εργαλείο MATLAB, συνοψίζονται παρακάτω:

- Αρχικά δημιουργούμε το `iwan` που είναι το διάνυσμα που έχει τις κυματομορφές του ρεύματος. Αυτές προκύπτουν από τις τιμές των ρευμάτων ενός συγκεκριμένου αρχείου εισόδου λαμβάνοντας υπόψιν την περίοδο T και τον αριθμό των διαστημάτων στα οποία θέλουμε να την χωρίσουμε.
- Το μέγεθος του δείγματος το οποίο επιλέγουμε να χρησιμοποιήσουμε είναι $nm=3000$. Προκειμένου να γίνει το μέγεθος του δείγματος ίσο με 3000, παραλείπουμε από το κάθε block τα πρώτα `off` στοιχεία (`off` = μέγεθος του block - nm και στο συγκεκριμένο κύκλωμα είναι 4). Από το `iwan` προκύπτει το διάνυσμα `mvarsam` που έχει τις κυματομορφές μόνο των εισόδων που ανήκουν στο δείγμα.
- Δημιουργούμε το διάνυσμα `ialt` που δείχνει τις τιμές των ρευμάτων σε όλα τα σημεία του δείγματος, για κάθε χρονική στιγμή j , όπου $j=1, \dots, N$ σε έναν κύκλο ρολογιού.

- Κάνουμε επίλυση του δικτύου (transient ανάλυση) για καθένα από τα σημεία του δείγματος. Από αυτή δημιουργείται ένα σύνολο κυματομορφών και μεταξύ αυτών υπολογίζεται η πτώση τάσης σε όλους τους κόμβους.
- Χωρίζουμε το κάθε δείγμα σε $nm/n=100$ υποδείγματα μεγέθους $n=30$. Για κάθε $1 \leq k \leq sn$ (sn ο αριθμός των πηγών ρεύματος και είναι ίσος με το πηλίκο του μεγέθους του αρχείου εισόδου προς το μέγεθος του block του κυκλώματος που μελετάμε κάθε φορά) κατασκευάζουμε το δείγμα των μέγιστων σημείων από τα nm/n υποδείγματα.
- Από το `mvarsam` που όπως αναφέραμε αποτελεί το δειγματοχώρο, υπολογίζουμε τα μεγιστικά σημεία `nmal` (maximal points).
- Στη συνέχεια, υπολογίζουμε την προβολή των maximal points στο δειγματοχώρο, έτσι ώστε να τα μετακινήσουμε και να τα προβάλλουμε σε κάποιο άλλο χώρο χρησιμοποιώντας το διάνυσμα μετακίνησης `mon`.
- Από το σύνολο πτώσεων τάσης που έχουμε, παίρνουμε τις μέγιστες πτώσεις τάσεις μόνο από αυτά τα σημεία (maximal points).
- Τελικά, αυτές τις μέγιστες πτώσεις τάσης που πήραμε, τις αφαιρούμε από την μέγιστη τιμή (1.2 Volt). Το αποτέλεσμα είναι το διάνυσμα `Vwc` που αποτελείται από τις 32 τάσεις που θα εισάγουμε στο εργαλείο NanoTime. Επιπλέον, από αυτό το διάνυσμα ξεχωρίζουμε την μικρότερη τάση (`Vpres`) και τη χρησιμοποιούμε στο NanoTime, στην περίπτωση που έχουμε μία μόνο τάση σε όλη τη σχεδίαση.

Παρακάτω φαίνονται οι τιμές των μεγεθών που χρησιμοποιήσαμε καθώς και οι τάξεις μεγέθους τους έτσι ώστε να είναι όσο το δυνατόν πιο σωστά τα αποτελέσματα της προσομοίωσης των κυκλωμάτων.

- `rsh=1e-4`, αντίσταση οριζόντιων κλάδων (σε kOhms/sq)
- `rsv=1e-4`, αντίσταση κάθετων κλάδων (σε kOhms/sq)
- `lsh=2e-7`, αυτεπαγωγή οριζόντιων κλάδων ανά μονάδα μήκους (σε uH/um)
- `lsv=2e-7`, αυτεπαγωγή κάθετων κλάδων ανά μονάδα μήκους (σε uH/um)
- `sizv=10000`, πλάτος του δικτύου ισχύος (σε um)
- `sizh=10000`, μήκος του δικτύου ισχύος (σε um)
- `n1=20`, αριθμός οριζόντιων κόμβων του δικτύου ισχύος
- `n2=20`, αριθμός κάθετων κόμβων του δικτύου ισχύος

- με $(n_1 \times n_2)$ κόμβους και $(n_1 \times (n_2-1) + n_2 \times (n_1-1))$ κλάδους σε όλο το δίκτυο
- $n_p=15$, αριθμός πηγών τροφοδοσίας του δικτύου ισχύος
- $r_{pin}=1$, αντίσταση πηγής τροφοδοσίας σε kOhms
- $l_{pin}=1e-4$, αυτεπαγωγή πηγής τροφοδοσίας (σε uH)
- $c_{rph}=1e-4$, χωρητικότητα παραλλήλων πλακών στους οριζόντιους κλάδους σε pF/um²
- $c_{rpn}=1e-4$, χωρητικότητα παραλλήλων πλακών στους κάθετους κλάδους σε pF/um²
- $c_{ffh}=1e-4$, πλευρική χωρητικότητα στους οριζόντιους κλάδους σε pF/um
- $c_{ffv}=1e-4$, πλευρική χωρητικότητα στους κάθετους κλάδους σε pF/um
- $c_{pin}=0.01$, χωρητικότητα πηγής τροφοδοσίας pF
- $c_{decou}=100$, χωρητικότητα αποσύνδεσης σε pF

Τα κύρια βήματα της διαδικασίας, που έγινε με το εργαλείο NanoTime, συνοψίζονται παρακάτω:

- Αρχικά ενημερώνουμε το εργαλείο πως το κύκλωμα το οποίο θα επεξεργαστούμε είναι το c6288. Μόλις γίνει αυτό, δημιουργείται μία εσωτερική αναπαράσταση της σχεδίασης.
- Έπειτα, δηλώνουμε τα χαρακτηριστικά της σχεδίασης, όπως είναι η κατεύθυνση των σημάτων και η τάση τροφοδοσίας. Η τιμή της τάσης τροφοδοσίας καθορίζεται από τα αποτελέσματα που έχουμε από το εργαλείο MATLAB. Έχουμε δύο περιπτώσεις:
 - Στη σχεδίαση υπάρχουν 32 διαφορετικές τάσεις
 - Στη σχεδίαση υπάρχει μόνο μία τάση, η οποία είναι η μικρότερη (άρα μεγαλύτερη πτώση τάσης) από τις 32 που χρησιμοποιήθηκαν στην πρώτη περίπτωση
- Στη συνέχεια, δηλώνουμε τους χρονικούς περιορισμούς της σχεδίασης, δηλαδή την καθυστέρηση των σημάτων εισόδου και εξόδου.
- Αφού έχουν δηλωθεί όλοι οι περιορισμοί και τα χαρακτηριστικά, και έχει γίνει έλεγχος ότι είναι σωστά, κάνουμε στατική ανάλυση χρονισμού. Σ' αυτό το σημείο, αποθηκεύονται σε μία βάση δεδομένων πληροφορίες σχετικά με τα χειρότερα μονοπάτια της σχεδίασης.

- Τέλος, μπορούμε να πάρουμε μία ποικιλία αναφορών με διάφορες πληροφορίες σχετικά με αυτά τα μονοπάτια. Στην προσομοίωση που κάναμε οι αναφορές μας δείχνουν την καθυστέρηση των πέντε χειρότερων μονοπατιών και το μονοπάτι δεδομένων και στις δύο περιπτώσεις (με μία μόνο τάση και 32 διαφορετικές).

Στον παρακάτω πίνακα (4.1) φαίνονται τα αποτελέσματα της προσομοίωσης. Τα μονοπάτια εμφανίζονται με φθίνουσα σειρά. Αυτό σημαίνει πως για κάθε grid το μονοπάτι A έχει τη μεγαλύτερη καθυστέρηση, δηλαδή είναι το critical path.

Πίνακας 4.1 Η καθυστέρηση ενός κυκλώματος (για 1 ή 32 διαφορετικές τάσεις) στα πέντε χειρότερα μονοπάτια

Case study	Grid	Worst-Path	Delay (ns)		% difference
			32 voltages	1 voltage	
#1	G100-10	A	8.908	11.550	22.87
		B	8.903	11.544	22.88
		C	8.879	11.506	22.83
		D	8.876	11.500	22.82
		E	8.871	11.488	22.78
#2	G150-10	A	10.485	14.714	28.74
		B	10.479	14.707	28.75
		C	10.468	14.693	28.76
		D	10.452	14.686	28.83
		E	10.446	14.672	28.80
#3	G150-15	A	5.274	5.508	4.25
		B	5.272	5.505	4.23
		C	5.250	5.484	4.27
		D	5.248	5.481	4.25
		E	5.233	5.461	4.18
#4	G225-10	A	8.910	10.398	14.31
		B	8.905	10.393	14.32
		C	8.875	10.359	14.33
		D	8.869	10.354	14.34
		E	8.856	10.329	14.26
#5	G225-15	A	5.310	5.572	4.70
		B	5.307	5.569	4.70
		C	5.286	5.548	4.72
		D	5.283	5.545	4.72
		E	5.276	5.527	4.54

#6	G300-15	A	5.197	5.404	3.83
		B	5.195	5.401	3.81
		C	5.174	5.380	3.83
		D	5.173	5.378	3.81
		E	5.171	5.358	3.49
#7	G300-20	A	4.362	4.495	2.96
		B	4.359	4.493	2.98
		C	4.340	4.473	2.97
		D	4.340	4.471	2.93
		E	4.338	4.459	2.71
#8	G400-10	A	8.826	10.009	11.82
		B	8.820	10.004	11.84
		C	8.793	9.971	11.81
		D	8.787	9.966	11.83
		E	8.765	9.927	11.71
#9	G400-15	A	5.197	5.379	3.38
		B	5.194	5.377	3.40
		C	5.174	5.355	3.38
		D	5.171	5.353	3.40
		E	5.169	5.334	3.09
#10	G400-20	A	4.405	4.482	1.72
		B	4.402	4.480	1.74
		C	4.383	4.461	1.75
		D	4.381	4.458	1.73
		E	4.380	4.446	1.48
#11	G400-30	A	3.781	3.848	1.74
		B	3.779	3.846	1.74
		C	3.771	3.834	1.64
		D	3.768	3.832	1.67
		E	3.761	3.828	1.75
#12	G600-10	A	9.250	9.898	6.55
		B	9.244	9.892	6.55
		C	9.213	9.860	6.56
		D	9.208	9.855	6.57
		E	9.176	9.814	6.50
#13	G600-15	A	5.174	5.361	3.49
		B	5.172	5.358	3.47
		C	5.151	5.337	3.49
		D	5.149	5.334	3.47
		E	5.146	5.316	3.20
#14	G600-25	A	4.021	4.153	3.18
		B	4.018	4.150	3.18
		C	4.011	4.132	2.93
		D	4.009	4.129	2.91
		E	4.000	4.129	3.12

#15	G600-40	A	3.589	3.651	1.70
		B	3.587	3.648	1.67
		C	3.577	3.632	1.51
		D	3.575	3.631	1.54
		E	3.569	3.629	1.65
#16	G800-15	A	5.269	5.462	3.53
		B	5.266	5.459	3.54
		C	5.245	5.438	3.55
		D	5.242	5.435	3.55
		E	5.234	5.420	3.43
#17	G800-20	A	4.339	4.403	1.45
		B	4.336	4.401	1.48
		C	4.318	4.382	1.46
		D	4.317	4.379	1.42
		E	4.315	4.370	1.26
#18	G800-25	A	3.994	4.097	2.51
		B	3.991	4.095	2.54
		C	3.984	4.076	2.26
		D	3.981	4.075	2.31
		E	3.973	4.074	2.48
#19	G800-40	A	3.593	3.653	1.64
		B	3.591	3.651	1.64
		C	3.581	3.634	1.46
		D	3.578	3.633	1.51
		E	3.573	3.631	1.60
#20	G1200-15	A	5.133	5.269	2.58
		B	5.131	5.267	2.58
		C	5.110	5.245	2.57
		D	5.109	5.243	2.56
		E	5.108	5.225	2.24
#21	G1200-20	A	4.334	4.415	1.83
		B	4.332	4.412	1.81
		C	4.315	4.393	1.78
		D	4.313	4.391	1.78
		E	4.312	4.381	1.57
#22	G1200-25	A	3.992	4.018	0.65
		B	3.989	4.015	0.65
		C	3.982	3.998	0.40
		D	3.980	3.997	0.43
		E	3.971	3.996	0.63
#23	G1200-40	A	3.585	3.627	1.16
		B	3.582	3.625	1.19
		C	3.576	3.609	0.91
		D	3.574	3.607	0.91
		E	3.565	3.606	1.14

#24	G1225-15	A	5.170	5.269	1.88
		B	5.167	5.267	1.90
		C	5.146	5.245	1.89
		D	5.143	5.243	1.91
		E	5.139	5.225	1.65

Όπως αναφέρθηκε και σε προηγούμενο κεφάλαιο, η καθυστέρηση είναι αύξουσα συνάρτηση. Έτσι, η αύξηση της πτώσης τάσης σε μία ή περισσότερες πύλες θα έχει σαν αποτέλεσμα την αύξηση της καθυστέρησης του μονοπατιού και κατ' επέκταση την αύξηση της συνολικής καθυστέρησης του κυκλώματος. Με βάση αυτό, είναι απόλυτα κατανοητές οι παρακάτω παρατηρήσεις σε σχέση με τα αποτελέσματα που παρουσιάζονται στον πίνακα:

- Όσο μικραίνει το μέγεθος του grid, αυξάνεται η πτώση τάσης (κι άρα μειώνεται η τάση τροφοδοσίας) με αποτέλεσμα να αυξάνεται και η καθυστέρηση του κυκλώματος.
- Όποιο κι αν είναι το μέγεθος του grid, στην περίπτωση που στο κύκλωμα υπάρχουν 32 διαφορετικές τάσεις τροφοδοσίας, η πτώση τάσης είναι μικρότερη από την περίπτωση που υπάρχει μία μόνο τάση τροφοδοσίας, η μικρότερη από τις 32. Άρα και η καθυστέρηση του κυκλώματος στην πρώτη περίπτωση με τις 32 τάσεις, είναι μικρότερη.
- Όσο αυξάνεται ο αριθμός των πηγών τροφοδοσίας σε ένα grid, έχουμε καλύτερη κατανομή ισχύος στο κύκλωμα. Έτσι, μειώνεται η πτώση τάσης και άρα μικραίνει και η καθυστέρηση του κυκλώματος.
- Τέλος, όσο αυξάνεται ο αριθμός των πηγών τροφοδοσίας σε ένα grid, παρατηρούμε πως μειώνεται η διαφορά (% difference) στην τάση στις δύο περιπτώσεις, αυτής με 32 τάσεις τροφοδοσίας και αυτής με μία.

Παράρτημα Α - Κώδικας MATLAB

A.1. Δημιουργία των κυματομορφών των ρευμάτων του αρχείου εισόδου, του διανύσματος των κυματομορφών των εισόδων που ανήκουν στο δείγμα, και υπολογισμός του ρεύματος στα σημεία αυτά

```

load c6288_3004.txt           %input data file
idc=-c6288_3004(:,5)*1e-3;   %currents in mA
nms=size(idc,1);

T=10;
N=10;
h=T/N;

t1=(1:N/2)*h;               %first half of period
itri1=2*idc*t1/T;
iwav1=itri1;
t2=(N/2+1:N)*h;           %second half of period
itri2=2*idc*(N*T-T+2*t2-N*t2)/(N*T);
iwav2=itri2;
t=[t1 t2];
iwav=[iwav1 iwav2];

nm=3000;
off=4;
sn=32;

%create multivariate sample of excitations
mvarsam=zeros(nm,N*sn);
for i=1:sn
    mvarsam(:,N*(i-1)+1:N*i)=iwav(off*i+nm*(i-1)+1:off*i+nm*i,:);
end

%alternative arrangement of sample of excitations
ialt=zeros(nm,N*sn);

```

```

for j=1:N
    for i=1:sn
        ialt(:,(j-1)*sn+i)=mvarsam(:,N*(i-1)+j);
    end
end
end

```

A.2. Αρχικοποίηση μεταβλητών και καθορισμός των κόμβων που τοποθετούνται οι πηγές ρεύματος και οι πηγές τροφοδοσίας

```

%number of current sinks (sn) must already be defined

rsh=1e-4;
rsv=1e-4;
lsh=2e-7;
lsv=2e-7;
sizv=350;
sizh=350;
n1=20;
n2=20;
% -> (n1 x n2) nodes in the grid
pv=sizh/(n1-1);
ph=sizv/(n2-1);
vn=30;
rpin=0.05;%1;
lpin=1e-4;
cpph=1e-4;
cppv=1e-4;
cffh=1e-4;
cffv=1e-4;
cpin=0.01;%10;
cdecap=100;%0;

w=ones(n2+n1,1);           %vector of widths of horizontal and vertical
                            %lines

nper=randperm(n1*n2);

```

```

ivn=nper(:,1:vn);           %node identifiers where voltage sources are
                             %placed
nper=randperm(n1*n2);
isn=nper(:,1:sn);           %node identifiers where sinks are placed

```

A.3. Υπολογισμών του διανύσματος μετακίνησης και των μεγιστικών σημείων (maximal points) - Σχηματισμός του δικτύου ισχύος και ανάλυσή του

```

%capacitance and incidence matrix

ch= repmat(cpph*pv*w(1:n2, :)+cffh*pv,1,n1-1);
%horizontal branch capacitances

cv= repmat(cppv*ph*w(n2+1:n2+n1, :)+cffv*ph,1,n2-1);
%vertical branch capacitances

C=zeros(n1*n2,n1*n2);
A1=zeros(n1*n2,n1*(n2-1)+n2*(n1-1)+vn);

%enumerate horizontal branches
for i=1:n2
    for j=1:n1-1
        ni1=(i-1)*n1+j;           %node indices for current branch
        ni2=ni1+1;
        ni3=(i-1)*(n1-1)+j;      %branch index conversion from 2D to 1D
        C(ni1,ni1)=C(ni1,ni1)+(1/2)*ch(i,j);
        C(ni2,ni2)=C(ni2,ni2)+(1/2)*ch(i,j);
        A1(ni1,ni3)=1;
        A1(ni2,ni3)=-1;
    end
end
end

```



```

%enumerate vertical branches
for i=1:n1
    for j=1:n2-1
        ni1=(j-1)*n1+i;          %node indices for current branch
        ni2=ni1+n1;
        ni3=(i-1)*(n2-1)+j+n2*(n1-1);    %branch index conversion from
                                           %2D to 1D

        C(ni1,ni1)=C(ni1,ni1)+(1/2)*cv(i,j);
        C(ni2,ni2)=C(ni2,ni2)+(1/2)*cv(i,j);
        A1(ni1,ni3)=1;
        A1(ni2,ni3)=-1;
    end
end

%enumerate supply branches
for i=1:vn
    C(ivn(i),ivn(i))=C(ivn(i),ivn(i))+cpin+cdecap;
    A1(ivn(i),n1*(n2-1)+n2*(n1-1)+i)=-1;
end

%-----%

%inductance matrix

lh= repmat(lsh*pv*log(150./w(1:n2,:)),1,n1-1);
%horizontal branch inductances

lv= repmat(lsv*ph*log(150./w(n2+1:n2+n1,:)),1,n2-1);
%vertical branch inductances

lbranch=[reshape(lh',n2*(n1-1),1);reshape(lv',n1*(n2-
1),1);lpin*ones(vn,1)];

L=diag(lbranch);

%-----%

```

```

%resistance matrix

rh= repmat (rsh*pv./w(1:n2,:),1,n1-1);
%horizontal branch resistances

rv= repmat (rsv*ph./w(n2+1:n2+n1,:),1,n2-1);
%vertical branch resistances

rbranch= [reshape (rh',n2*(n1-1),1); reshape (rv',n1*(n2-
1),1); rpin*ones(vn,1)];

R=diag (rbranch);

%-----%

%transient analysis for maximal waveform excitations

Gb= [zeros (n1*n2,n1*n2) A1;A1' -R];
Cb= [C zeros (n1*n2,n1*(n2-1)+n2*(n1-1)+vn); zeros (n1*(n2-1)+n2*(n1-
1)+vn,n1*n2) -L];

D=inv (Gb+Cb/h);
B=D*Cb/h;

Is=zeros (n1*n2+n1*(n2-1)+n2*(n1-1)+vn,nm); %current excitations
%from gates
Vb=zeros (n1*n2+n1*(n2-1)+n2*(n1-1)+vn,nm); %node voltages and
%branch currents

Vmean=zeros (n1*n2,nm);

for j=1:N
    Is (isn,:)=ialt (:,sn*(j-1)+1:sn*j)';
    Vb=D*Is+B*Vb;
    Vmean=Vmean+Vb(1:n1*n2,:);
End

Vmean=Vmean/N;
Vms=Vmean (isn,:); %mean voltage drops at sinks

```

```

%+-----+%
%| Statistical maxima estimation |%
%+-----+%

n=30;          %size of sub-samples for estimation (must be at least 30)
m=nm/n;       %number of sub-samples - or size of sample of maxima - for
              %estimation (must be at least 100)

xm=zeros(m,1);

eulg=0.5772;   %Euler gamma constant

den=1/(1+n*sqrt(pi*log(n))*(erf(sqrt(log(n)))-1));
%denominator of estimate

stmat=zeros(2,sn);

for i=1:sn
    sam=Vms(:,i);
    smax=max(sam);

    for j=1:m
        xm(j)=max(sam(n*(j-1)+1:n*j));           %sample of maxima
    end

    bg=std(xm)*sqrt(6)/pi;
    ag=mean(xm)-bg*eulg;
    wg=ag+bg*den;                                %upper endpoint estimate

    stmat(:,i)=[smax;wg];
end

mov=stmat(2,:)-stmat(1,:);                       %movement vector for the sample
                                                %maximal points

```

```

%locate maximal points of the sample space
mmal=[];

for i=1:nm
    mmali=Vms(i,:);
    comp=( repmat(mmali,nm,1)<Vms);
    comp2=(sum(comp,2)==sn);
    if sum(comp2)==0
        mmal=[mmal;mmali];
    end
end

nmal=size(mmal,1);

Vwc=1.2-(mmal+repmat(mov,nmal,1))    %statistically project sample
                                     %maximal points into the whole
                                     %population

Vpes=min(min(Vwc))

fid = fopen('20x20_30.txt','w');
for i=1:32
    fprintf(fid,'v%d vdd%d 0 dc %2.4fv\n',i,i,Vwc(i));
end
fclose(fid);

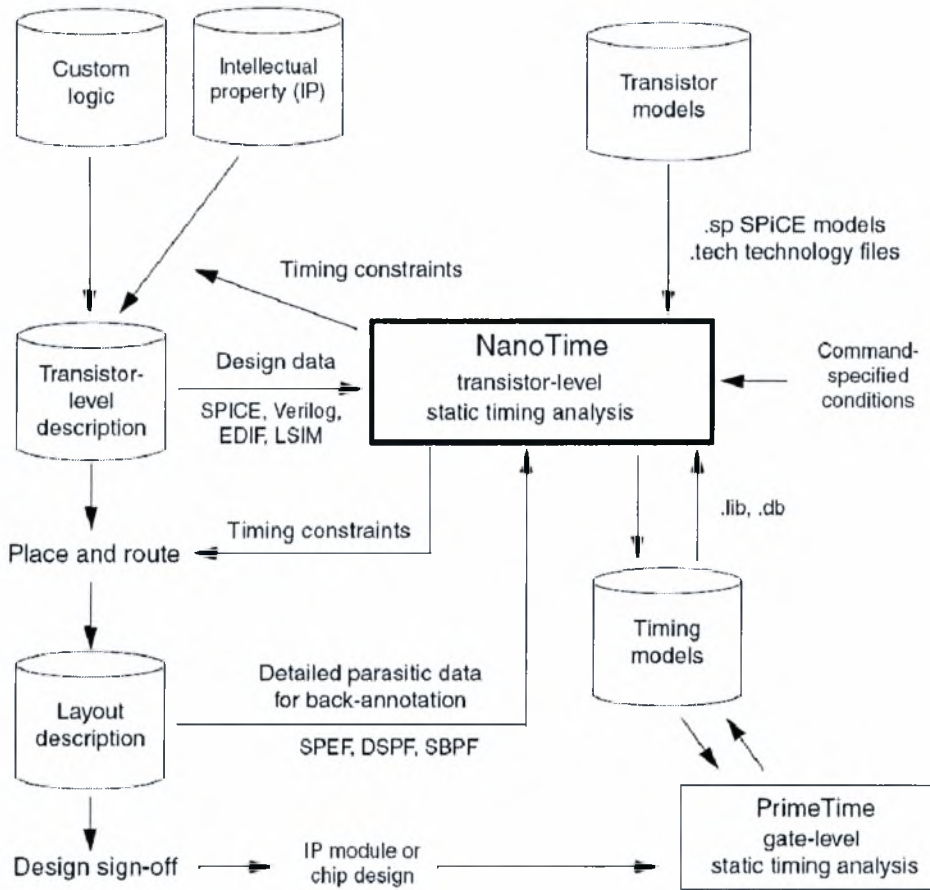
```

Παράρτημα Β - NanoTime tutorial

B.1 Τα χαρακτηριστικά του NanoTime

Το NanoTime είναι ένα εργαλείο στατικής ανάλυσης χρονισμού σε επίπεδο τρανζίστορ για επαλήθευση χρονισμού σε επίπεδο block και πλήρους ολοκληρωμένου κυκλώματος. Ένα εργαλείο εύκολο στη χρήση, για σχεδιαστές σύνθετων μικροεπεξεργαστών, ψηφιακού σήματος επεξεργαστών και custom λογικών πυλών. Το NanoTime κάνει ανάλυση σε επίπεδο τρανζίστορ έχοντας περίπου την ίδια ακρίβεια με το SPICE για όλα τα κρίσιμα μονοπάτια χρονισμού. Επιτρέπει στον system-on-chip (SoC) σχεδιαστή να εντοπίσει γρήγορα και σωστά όλες τις χρονικές παραβιάσεις, με αποτέλεσμα να μειώνεται ο χρόνος επαλήθευσης και ο χρόνος για την παραγωγή.

Το NanoTime είναι το εργαλείο επόμενης γενιάς που θα αντικαταστήσει το PathMill για χρονισμούς σε επίπεδο τρανζίστορ. Έχει την αποδεδειγμένη ακρίβεια και αξιοπιστία του PathMill, αλλά επιπλέον παρέχει χωρητικότητα, απόδοση και καινοτόμα χαρακτηριστικά απαραίτητα για να υποστηρίξουν τις σημερινές σύνθετες σχεδιάσεις και τις τεχνολογίες των 90nm και κάτω. Επιπλέον, προσφέρει ένα περιβάλλον το οποίο είναι οικείο στους χρήστες των PrimeTime και Design Compiler.



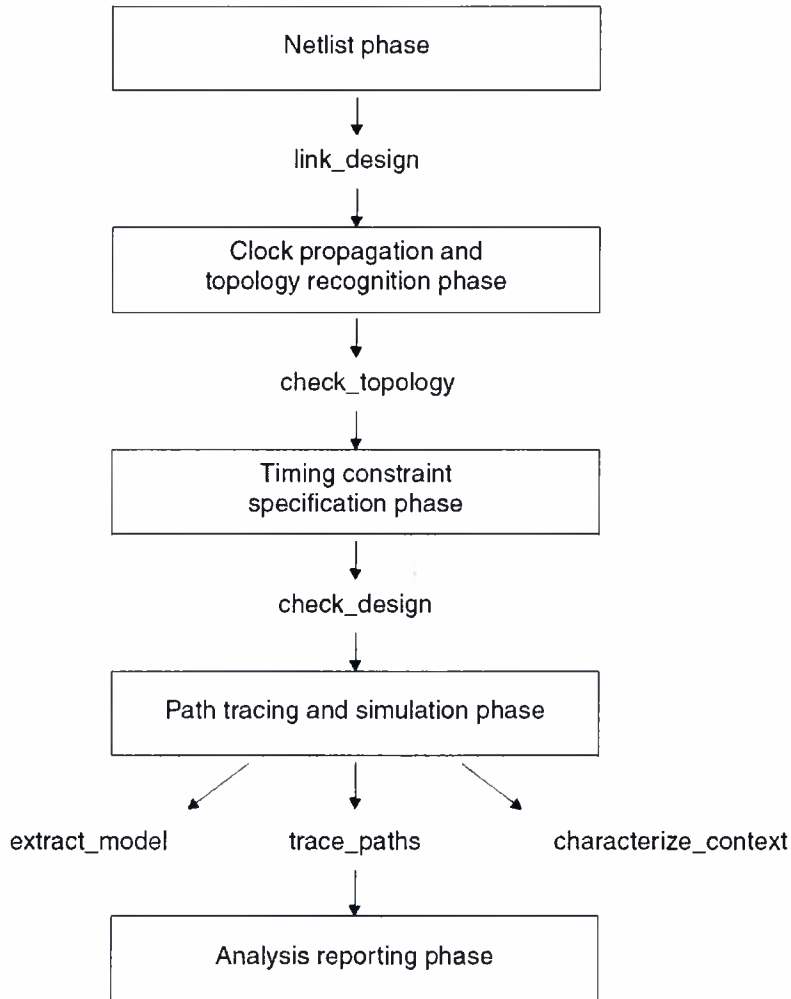
Εικόνα Β.1 Η χρήση του NanoTime σε μία τυπική ροή σχεδίασης.

Το NanoTime εξετάζει την ιεραρχία της σχεδίασης και δημιουργεί μία εσωτερική αναπαράστασή της. Στη συνέχεια, εξετάζει την τοπολογία της σχεδίασης για να αναγνωρίσει δομές κυκλωμάτων όπως latches, πολυπλέκτες, λογικές πύλες domino, πύλες μετάδοσης και κελιά RAM. Αυτόματα αναγνωρίζει τις περισσότερες δομές βασισμένο στην τοπολογία των κυκλωμάτων. Επίσης, αναγνωρίζει τις δομές που δεν αναγνώρισε ήδη, με την καθοδήγηση του χρήστη είτε αντιστοιχίζοντας τις τοπολογίες, είτε από την ονομασία των κελιών. Μόλις έχουν οριστεί όλοι οι χρονικοί περιορισμοί και οι συνθήκες λειτουργίας, το NanoTime αναγνωρίζει όλα τα μονοπάτια και δημιουργεί μία βάση δεδομένων. Από αυτή τη βάση δεδομένων, μπορείς να δημιουργήσεις μία μεγάλη ποικιλία από αναφορές χρονισμού.

B.2 Η ανάλυση ροής του NanoTime

Το μέρος της ανάλυσης του NanoTime αποτελείται από μία αλληλουχία φάσεων. Σε κάθε φάση, ο χρήστης παρέχει δεδομένα και οδηγίες που θα χρησιμοποιήσει το NanoTime στη συγκεκριμένη φάση. Το NanoTime ελέγχει κάθε οδηγία αν είναι σωστή ως προς την σύνταξή της αλλά και αν μπορεί να χρησιμοποιηθεί στο συγκεκριμένο σημείο της ροής. Κάθε φάση τελειώνει με μία εντολή (φαίνονται στην παρακάτω εικόνα), η οποία αναφέρει αν ήταν επιτυχής η ολοκλήρωση της συγκεκριμένης φάσης.

Οι φάσεις πρέπει να ολοκληρώνονται σε μία συγκεκριμένη σειρά και κάθε φάση πρέπει να ολοκληρώνεται πετυχημένα πριν ξεκινήσει η επόμενη. Κατά τη διάρκεια μίας φάσης, τα βήματα μπορούν να γίνονται με οποιαδήποτε σειρά. Αν υπάρξει κάποιο πρόβλημα, μπορείς να επιστρέψεις στην αρχή της φάσης για να αλλάξεις τα δεδομένα ή τις οδηγίες που χρειάζεται για να ολοκληρωθεί πετυχημένα η φάση. Αν είναι απαραίτητο, μπορείς να επιστρέψεις και σε προηγούμενη φάση και να επαναλάβεις όλα τα βήματα από αυτό το σημείο και μετά.



Εικόνα Β.2 Οι βασικές φάσεις στην ροή ανάλυσης.

B.2.1 Φάση Netlist

Είναι η πρώτη φάση της ανάλυσης. Το NanoTime διαβάζει τα δεδομένα της σχεδίασης και της τεχνολογίας και δημιουργεί μία εσωτερική αναπαράσταση της σχεδίασης, προκειμένου να αναλυθεί.

B.2.2 Διάδοση του Ρολογιού και Αναγνώριση της Τοπολογίας (Clock Propagation and Topology Recognition)

Ορίζονται τα χαρακτηριστικά της σχεδίασης που θα χρησιμοποιηθούν στην αναγνώριση της τοπολογίας. Αυτά είναι: τα χαρακτηριστικά του ρολογιού, οι σταθερές τιμές των εισόδων, οι λογικοί περιορισμοί και η κατεύθυνση των σημάτων.

B.2.3 Καθορισμός των Χρονικών Περιορισμών (Specifying Timing Constraints)

Για να γίνει ανάλυση χρονισμού, το NanoTime χρειάζεται πληροφορίες σχετικά με τους χρονικούς περιορισμούς. Οι πιο συνηθισμένοι τύποι αυτών είναι: τα χαρακτηριστικά του ρολογιού, οι καθυστερήσεις εισόδου-εξόδου, τα μοντέλα των τρανζίστορ, τα παρασιτικά, οι έλεγχοι χρονισμού, οι εξαιρέσεις χρονισμού, οι λογικοί περιορισμοί στα δίκτυα, οι συνθήκες οδήγησης των εισόδων ή οι χρόνοι μετάδοσης των εισόδων, οι συνθήκες φόρτωσης των εξόδων και οι σταθερές λογικές συνθήκες στις πύλες.

B.2.4 Αναγνώριση και Προσομοίωση των Μονοπατιών (Path Tracing and Simulation)

Αφού οριστούν όλοι οι περιορισμοί και ολοκληρωθεί επιτυχημένα η προηγούμενη φάση, μπορείς να προχωρήσεις στην ανάλυση χρονισμού. Το NanoTime μπορεί να πραγματοποιήσει τους εξής τύπους χρονικής ανάλυσης: Timing verification, Timing model extraction, Context characterization. Για να πραγματοποιηθούν περισσότεροι από έναν τύποι ανάλυσης χρονισμού, θα πρέπει να επαναληφθούν κάποιες από τις προηγούμενες φάσεις καθώς η ανάλυση χρονισμού επηρεάζει την βάση δεδομένων της σχεδίασης.

B.2.5 Αναφορές σχετικά με την Ανάλυση (Analysis Reporting)

Μπορείς να πάρεις μία ποικιλία αναφορών σχετικά με τη σχεδίαση και τους χρονισμούς. Αυτές προκύπτουν από τη βάση δεδομένων χωρίς να πραγματοποιηθεί κάποια επιπλέον αναγνώριση μονοπατιού ή προσομοίωση. Μπορείς να επιλέξεις το πλήθος των

πληροφοριών και τον αριθμό των μονοπατιών που θα υπάρχουν στην αναφορά, όπως επίσης και το σκοπό της σχεδίασης. Μία τυπική αναφορά καταγράφει σε λίστα τα χειρότερα μονοπάτια με σειρά αυξανόμενου slack, αρχίζοντας με το χειρότερο μονοπάτι. Για περαιτέρω ανάλυση, μπορείς να γράψεις ένα netlist αρχείο σε SPICE που θα αναπαριστά ένα επιλεγμένο χειρότερο μονοπάτι.

B.3 Ξεκινώντας το NanoTime

Το NanoTime είναι ένα εργαλείο command-line-driven που εκτελείται σε UNIX ή λειτουργικό σύστημα Linux. Πριν τη χρήση του NanoTime, το λογισμικό πρέπει να έχει εγκατασταθεί στον χώρο και να υπάρχει η άδεια χρήσης του.

B.3.1 Ξεκινώντας μία NanoTime session

Η γραμμή εντολών του NanoTime ονομάζεται `nt_shell`. Για να ξεκινήσει το `nt_shell`, εισάγεται την παρακάτω εντολή στο prompt του λειτουργικού συστήματος:

```
% nt_shell
```

Το NanoTime αυτόματα ελέγχει την άδεια και εμφανίζει το αρχικό μήνυμα και το `nt_shell` prompt.

B.3.2 Τερματίζοντας μία NanoTime session

Για να τερματιστεί μία NanoTime session, εισάγεται την εντολή `quit` ή `exit` στο prompt του NanoTime:

```
nt_shell> exit
```

B.4 Χρησιμοποιώντας το Shell Interface

Το shell του NanoTime, `nt_shell`, είναι ένα command-driven interface το οποίο βασίζεται στην Tcl γλώσσα δημιουργίας script. Το NanoTime εκτελεί όποια πράξη χρειάζεται για κάθε εντολή και αναφέρει τα αποτελέσματα σε μορφή text. Η σύνταξη των εντολών είναι

ευαίσθητη στη χρήση κεφαλαίων και πεζών χαρακτήρων. Οι εντολές, οι επιλογές τους, τα ορίσματα και οι μεταβλητές, γενικά αποτελούνται από πεζούς χαρακτήρες και πρέπει να εισάγονται έτσι στο `nt_shell` prompt. Τα ονόματα των αντικειμένων είναι επίσης ευαίσθητα στη χρήση κεφαλαίων και πεζών χαρακτήρων.

B.4.1 Scripts Εντολών

Ένα script εντολών είναι ένα text αρχείο, που περιέχει μία ακολουθία εντολών `nt_shell`. Επίσης, το αρχείο που δημιουργείται στο τέλος ενός NanoTime session μπορεί να χρησιμοποιηθεί σαν script. Μπορεί ο κάθε χρήστης να δημιουργήσει δικά του scripts, ή να χρησιμοποιήσει scripts που δημιουργήθηκαν από άλλους.

Το NanoTime μπορεί να αναγνωρίσει αρχεία script σε απλή μορφή ASCII, ή συμπιεσμένη σε μορφή gzip. Επίσης, μπορεί να αναγνωρίσει ASCII κωδικοποιημένη σε κώδικα byte από τον TclPro μεταγλωττιστή.

Για να εκτελεστεί ένα script αυτών των μορφών, χρησιμοποιείται η εντολή `source`:

```
nt_shell> source file_name
```

Για να εκτελεστεί ένα script καθορισμένων εντολών από την αρχή, χρησιμοποιείται η επιλογή `-f` όταν ξεκινά το NanoTime.

```
% nt_shell -f file_name
```

B.4.2 Αποτελέσματα Εντολών

Κάθε εντολή στο NanoTime έχει ένα αποτέλεσμα. Για κάποιες εντολές αυτό μπορεί να είναι '1' που δηλώνει επιτυχία ή '0' που δηλώνει αποτυχία. Για κάποιες άλλες, το αποτέλεσμα είναι ένα σύνολο.

Πολλές φορές οι εντολές είναι εμφωλευμένες, η μία μέσα στην άλλη, έτσι ώστε το αποτέλεσμα της μίας εντολής να χρησιμοποιείται σαν όρισμα της άλλης. Κάθε εμφωλευμένη εντολή εσωκλείεται σε τετράγωνες αγκύλες (`[]`). Όταν εισάγεται μία μεγάλη εντολή, μπορεί να χωριστεί σε πολλές σειρές χρησιμοποιώντας τον χαρακτήρα

(\). Σε κάθε επιπλέον γραμμή, το NanoTime χρησιμοποιεί ένα δευτερεύοντα prompt, ένα ερωτηματικό (?).

Η έξοδος ορισμένων εντολών, όπως η `man` ή η `report_paths` είναι μία αναφορά. Από ορισμού του το NanoTime εμφανίζει όλη την αναφορά απευθείας. Για να σταματάει ανάμεσα σε κάθε αρχείο, κάνουμε την τιμή της μεταβλητής `sh_enable_page_mode` αληθή. Για να δούμε μία μεγάλη αναφορά με αυτή τη μορφή, πατάμε το κενό. Για να ακυρώσουμε την εμφάνιση μίας μεγάλης αναφοράς και να επιστρέψουμε στο `nt_shell` prompt, πληκτρολογούμε το γράμμα *q*.

B.4.3 Μεταβλητές

Το NanoTime προσφέρει πολλές επιλογές που μπορούν να ελεγχθούν με τη δήλωση μεταβλητών. Για τη δήλωση μίας μεταβλητής, χρησιμοποιείται η εντολή `set`. Όταν δηλωθεί μία μεταβλητή, το αποτέλεσμα που φαίνεται είναι απλά η νέα τιμή της μεταβλητής. Αν η δήλωση είναι μία μη επιτρεπτή τιμή, το NanoTime εμφανίζει ένα μήνυμα λάθους. Για να ενημερωθείς για την τιμή μίας μεταβλητής, χρησιμοποιείται η εντολή `printvar`. Επίσης, μπορεί να χρησιμοποιηθεί ο χαρακτήρας (*) για να εμφανιστούν όλες οι μεταβλητές, καθώς ορισμένες ξεκινούν με το ίδιο πρόθεμα.

B.4.4 Διακοπή Εκτέλεσης Εντολών

Η εκτέλεση μίας εντολής μπορεί να διακοπεί πληκτρολογώντας `Ctrl + c`. Η διακοπή ορισμένων μπορεί να πάρει λίγο χρόνο. Πληκτρολογώντας `Ctrl + c` πολλές φορές τερματίζει το `nt_shell` και επιστρέφεις στο prompt του λειτουργικού συστήματος.

B.4.5 Tcl Interface

Το `nt_shell` interface βασίζεται στην Tcl γλώσσα δημιουργίας script, όπως το Tcl shell του PrimeTime (`pt_shell`) και του Design Compiler (`dc_shell-t`). Μπορούν να χρησιμοποιηθούν χαρακτηριστικά της Tcl όπως μεταβλητές που ορίζονται από τον χρήστη, διαδικασίες, εκτέλεση υπό προϋποθέσεις, λίστες και εξαιρέσεις.

B.5 Γραπτή και Ηλεκτρονική Βοήθεια για το NanoTime

Αν χρειαστεί βοήθεια κατά τη χρήση του NanoTime, πληροφορίες είναι διαθέσιμες από τις παρακάτω πηγές:

- Εντολή `help`

Παρέχει περιεκτικές πληροφορίες σχετικά με τις εντολές του NanoTime. Μπορείς να έχεις ένα μέρος ή μία ολοκληρωμένη λίστα από εντολές ή να δεις τη σύνταξη μίας συγκεκριμένης εντολής.

- Εντολή `man`

Παρέχει λεπτομερείς πληροφορίες από κάθε εντολή, μεταβλητή, ή μήνυμα του συστήματος. Στο `nt_shell` prompt, πληκτρολογήστε `man` που να ακολουθείται από την εντολή, τη μεταβλητή ή τον κωδικό του μηνύματος και το NanoTime ανταποκρίνεται με μία λεπτομερή περιγραφή.

- Εγχειρίδια του NanoTime

Πιο συγκεκριμένα, πληροφορίες είναι διαθέσιμες στα

- *NanoTime and NanoTime Ultra User Guide*
- *NanoTime and NanoTime Ultra Quick Reference*

τα οποία υπάρχουν στη σελίδα του Internet <http://solvnet.synopsys.com>.

- Άρθρα SolvNet
- Online έγγραφα της Synopsys (SOLD)

B.6 Script

Για να πάρουμε τις καθυστερήσεις των χειρότερων μονοπατιών του κυκλώματος χρησιμοποιήσαμε το πρώτο script για την περίπτωση που υπάρχει η ίδια-χειρότερη τάση σε όλο το κύκλωμα και το δεύτερο για την περίπτωση που υπάρχουν 32 διαφορετικές τάσεις.

1.

```
set search_path "./"
register_netlist c6288.sp
list_libraries
```

```

link_design c6288

set_port_direction -input { A15 A14 A13 A12 A11 A10 A9 A8 A7 A6 A5 A4
A3 A2 A1 A0 B15 B14 B13 B12 B11 B10 B9 B8 B7 B6 B5 B4 B3 B2 B1 B0 }

set_port_direction -output { P31 P30 P29 P28 P27 P26 P25 P24 P23 P22
P21 P20 P19 P18 P17 P16 P15 P14 P13 P12 P11 P10 P9 P8 P7 P6 P5 P4 P3 P2
P1 P0 }

get_nets vdd!
set_supply_net vdd!
set_voltage 1.2 vdd!

check_topology

report_topology

set_input_delay 0.1 { A15 A14 A13 A12 A11 A10 A9 A8 A7 A6 A5 A4 A3 A2
A1 A0 B15 B14 B13 B12 B11 B10 B9 B8 B7 B6 B5 B4 B3 B2 B1 B0 }

set_output_delay 0.1 { P31 P30 P29 P28 P27 P26 P25 P24 P23 P22 P21 P20
P19 P18 P17 P16 P15 P14 P13 P12 P11 P10 P9 P8 P7 P6 P5 P4 P3 P2 P1 P0 }

check_design

trace_paths

report_constraint -all_violators -verbose

redirect -append nanotime.out {report_paths -max -max_paths 5 -by delay
-path_type full -show_path_index}

```

2.

```

set_search_path "./"
register_netlist c6288_32.sp
list_libraries
link_design c6288

set_port_direction -input { A15 A14 A13 A12 A11 A10 A9 A8 A7 A6 A5 A4
A3 A2 A1 A0 B15 B14 B13 B12 B11 B10 B9 B8 B7 B6 B5 B4 B3 B2 B1 B0 }

set_port_direction -output { P31 P30 P29 P28 P27 P26 P25 P24 P23 P22
P21 P20 P19 P18 P17 P16 P15 P14 P13 P12 P11 P10 P9 P8 P7 P6 P5 P4 P3 P2
P1 P0 }

set_sim_cfg_use_multi_voltage true

set_oc_global_voltage 1.2

check_topology

report_topology

```

```

set_input_delay 0.1 { A15 A14 A13 A12 A11 A10 A9 A8 A7 A6 A5 A4 A3 A2
A1 A0 B15 B14 B13 B12 B11 B10 B9 B8 B7 B6 B5 B4 B3 B2 B1 B0 }

set_output_delay 0.1 { P31 P30 P29 P28 P27 P26 P25 P24 P23 P22 P21 P20
P19 P18 P17 P16 P15 P14 P13 P12 P11 P10 P9 P8 P7 P6 P5 P4 P3 P2 P1 P0 }

check_design

trace_paths

report_constraint -all_violators -verbose

redirect -append nanotime_32.out {report_paths -max -max_paths 5 -by
delay -path_type full -show_path_index}

```

όπου οι 32 διαφορετικές τάσεις που αναφέραμε στη δεύτερη περίπτωση υπάρχουν στο αρχείο c6288.sp που κάνουμε register_netlist.

- Η μεταβλητή search_path ορίζει μία λίστα από διευθύνσεις. Σε αυτές, το NanoTime θα ψάξει για να βρει τα αρχεία με τις σχεδιάσεις, με τη σειρά που έχουν ταξινομηθεί στη λίστα.
- Αφού έχουν οριστεί οι μεταβλητές, πρέπει να καταγραφεί η netlist των αρχείων με την εντολή register_netlist. Αυτή η εντολή καθορίζει τα αρχεία που περιέχουν πληροφορίες της σχεδίασης για την ανάλυση. Όταν εκτελείται αυτή η εντολή, το NanoTime ελέγχει για την ύπαρξη των αρχείων και καταγράφει τα ονόματά τους. Ωστόσο, στην πραγματικότητα δεν διαβάζει τα αρχεία, ούτε ελέγχει τα περιεχόμενά τους. Η netlist των αρχείων χρησιμοποιείται όταν εκτελείται η εντολή link_design. Κάθε αρχείο μπορεί να περιέχει μία βιβλιοθήκη. Αφού φορτωθούν τα αρχεία, για να εμφανιστούν όλες οι βιβλιοθήκες σε μία λίστα χρειάζεται η εντολή list_libraries.
- Αφού η netlist των αρχείων έχει καταγραφεί με την εντολή register_netlist, μπορεί να χρησιμοποιηθεί η εντολή link_design. Αυτή δημιουργεί μία εσωτερική αναπαράσταση της σχεδίασης για να γίνει ανάλυση χρονισμού. Για να εκτελεστεί επιτυχώς, πρέπει να υπάρχει μία ολοκληρωμένη και σωστά υλοποιημένη σχεδίαση, η οποία θα σχετίζεται με όλες τις αναφερόμενες βιβλιοθήκες και σχεδιάσεις. Το NanoTime βρίσκει και συνδέει τις αναφορές στην σχεδίαση.
- Για να διασφαλιστεί μία πετυχημένη αναγνώριση της τοπολογίας, πρέπει να καθοριστεί η κατεύθυνση της ροής των σημάτων εισόδου και εξόδου, αν αυτή η

πληροφορία δεν είναι ήδη διαθέσιμη από τη netlist. Αυτό γίνεται με την εντολή `set_port_direction` και με τις επιλογές `-input` και `-output` αντίστοιχα.

➤ Η εντολή `get_nets` δημιουργεί ένα σύνολο από κόμβους από τη netlist. Επίσης μπορεί να χρησιμοποιηθεί η εντολή `set_supply_net` για να δηλωθούν ποιοι κόμβοι είναι πηγές τροφοδοσίας και ποιοι γειώσεις. Για την περίπτωση των γειώσεων πρέπει την εντολή να ακολουθεί η επιλογή `-gnd`. Για να ορίσουμε την τιμή μιας πηγής τροφοδοσίας, χρησιμοποιούμε την εντολή `set_voltage`. Κάθε φορά που χρησιμοποιείται αυτή η εντολή για έναν κόμβο, υπερισχύει της τελευταίας εντολής `set_voltage` που υπήρχε -αν υπήρχε- για αυτόν τον κόμβο.

➤ Το NanoTime υποστηρίζει ανάλυση της σχεδίασης με πολλαπλές πηγές τροφοδοσίας. Για να γίνει μία τέτοια ανάλυση πρέπει να γίνουν τα εξής:

1. Η τιμή της μεταβλητής `sim_cfg_use_multi_voltage` πρέπει να γίνει αληθής (`true`).
2. Πρέπει να δηλωθούν οι κόμβοι που αποτελούν πηγές τροφοδοσίας με την εντολή `set_supply_net`.
3. Πρέπει να οριστεί η τιμή της κάθε πηγής τροφοδοσίας με την εντολή `set_voltage`.

Διαφορετικά, κάνουμε την τιμή της μεταβλητής `sim_cfg_use_multi_voltage` αληθή στο script και ορίζουμε την τιμή της κάθε πηγής τροφοδοσίας με την εντολή `set_voltage` στο αρχείο τύπου SPICE που έχουμε κάνει `register_netlist`.

➤ Όταν η αναγνώριση της τοπολογίας ολοκληρωθεί, η εντολή `check_topology` κάνει έναν τελικό έλεγχο σχετικά με την ακεραιότητα των δομών. Αν ο έλεγχος είναι επιτυχής, μπορούμε να προχωρήσουμε στην επόμενη φάση της ανάλυσης που είναι ο καθορισμός των χρονικών περιορισμών.

➤ Η εντολή `report_topology`, όπως και άλλες εντολές αναφοράς, χρησιμοποιείται για να ενημερωθούμε σχετικά με τις δομές που έχουν αναγνωριστεί. Μπορούμε να έχουμε λεπτομερείς αναφορές για τα ρολόγια και την τοπολογία της σχεδίασης, όπως το λόγο που κάποιος κόμβος δεν χαρακτηρίστηκε ως ρολόι ή τη συνάρτηση κάποιου τρανζίστορ. Η συγκεκριμένη εντολή καταγράφει σε μία λίστα τους τύπους των δομών και αναφέρει τον αριθμό που ο καθένας από αυτούς εμφανίζεται στη σχεδίαση.

- Για να γίνει η ανάλυση χρονισμού, το NanoTime χρειάζεται πληροφορίες σχετικά με τις καθυστερήσεις εισόδου και εξόδου. Αυτές πρέπει να καθοριστούν από το χρήστη γιατί δεν μπορούν να προκύψουν από τη netlist. Οι εντολές με τις οποίες γίνεται αυτό, είναι `set_input_delay` και `set_output_delay`, αντίστοιχα.
- Όταν όλοι οι περιορισμοί έχουν οριστεί, η εντολή `check_design` ελέγχει τη σχεδίαση αν είναι σωστή σε σχέση με τους χρονικούς περιορισμούς που υπάρχουν. Αν δεν είναι, αναφέρει κάθε ασυνέπεια ή άλλο δομικό πρόβλημα, τα οποία πρέπει να διορθωθούν πριν προχωρήσουμε στην επόμενη φάση. Για να γίνει αυτός ο έλεγχος, η σχεδίαση χωρίζεται σε τμήματα που ονομάζονται `channel-connected blocks (CCBs)`. Κάθε CCB είναι ένα σύνολο από τρανζίστορ των οποίων οι πηγές και οι υποδοχές συνδέονται σε ένα δίκτυο, σε σειρά ή παράλληλα, ή και τα δύο. Το NanoTime ελέγχει τα CCBs και αναφέρει τυχόν προβλήματα που υπάρχουν στη σύνθεση τους. (NanoTime checks the CCBs and reports any problems it has in interpreting their configuration.)
- Το NanoTime κάνει ανάλυση χρονισμού όταν χρησιμοποιείται η εντολή `trace_paths`. Μελετάει με τη σειρά όλα τα μονοπάτια της σχεδίασης και σώζει λεπτομερείς πληροφορίες για τα χειρότερα από αυτά σε μία βάση δεδομένων. Αυτή η εντολή, προσφέρει διάφορες επιλογές όσον αφορά τους τύπους των μονοπατιών που έχουν αναλυθεί και τον αριθμό αυτών που έχουν αποθηκευτεί στη βάση δεδομένων.
- Με την εντολή `report_constraint` αναφέρουμε όλους τους περιορισμούς που σχετίζονται με την σχεδίαση. Με τις επιλογές `-all_violators` και `-verbose` παίρνουμε αναλυτικές πληροφορίες για όλα τα μονοπάτια και τους περιορισμούς.
- Η εντολή `report_paths` αναφέρει όλα τα μονοπάτια της σχεδίασης. Με τη χρήση της επιλογής `-by_delay` τα μονοπάτια εμφανίζονται με σειρά αυξανόμενης καθυστέρησης (σε συνδυασμό με την επιλογή `-min`) και με σειρά μειούμενης καθυστέρησης (σε συνδυασμό με την επιλογή `-max`). Για να μειωθούν τα αποτελέσματα της αναφοράς χρησιμοποιείται η επιλογή `-max_paths` (στο script που τρέχουμε ζητάμε να εμφανιστούν μόνο τα 5 χειρότερα μονοπάτια). Με την επιλογή `-show_path_index` θα αναφέρεται ο αριθμός αναγνώρισης του μονοπατιού (ID number) και με την επιλογή `-path_type full`, θα εμφανίζεται το μονοπάτι δεδομένων και το slack.
- Στην περίπτωση των 32 διαφορετικών τάσεων, το NanoTime ορίζει μία τάση σαν τη μέγιστη. Καλό είναι, για να προκύψουν σωστά αποτελέσματα, να ορίζεται αυτή η

τάση από τον χρήστη. Αυτό γίνεται με την εντολή `set` και τη μεταβλητή `oc_global_variable`, η οποία θα αποτελέσει αυτή τη μέγιστη τάση.

➤ Για να αποθηκευτούν τα αποτελέσματα της αναφοράς σε κάποιο αρχείο χρησιμοποιούμε την εντολή `redirect -append temp.out` (όπου `temp.out` το όνομα του αρχείου).

Βιβλιογραφία

- [1] N. Evmorfopoulos, D. Karampatzakis, G. Stamoulis, “*Precise Identification of the Worst-Case Voltage Drop Conditions in Power Grid Verification*”.
- [2] Νίκος Ι. Μάργαρης, Τόμος Α, “*Ανάλυση Ηλεκτρικών Κυκλωμάτων*”.
- [3] Tsung-Hao Chen and Charlie Chung-Ping Chen, “*Efficient Large-Scale Power Grid Analysis Based on Preconditioned Krylov-Subspace Iterative Methods*”.
- [4] “*A Monte Carlo approach for maximum power estimation based on extreme value theory*”, *IEEE Trans. Computer-Aided Design*, vol.21, pp. 415-432, 2002.
- [5] H. Kung, F. Luccio, and F. Preparata, “*On finding the maxima of a set of vectors*”, *J. ACM*, vol. 22, pp. 469-476, 1975.
- [6] N.E. Evmorfopoulos, D.P. Karampatzakis, and G.I. Stamoulis, “*Voltage-Drop-Constrained Optimization of Power Distribution Network Based on Reliable Maximum Current Estimates*”.
- [7] Stephen P. Boyd, Seung-Jean Kim, Dinesh D. Patil, Mark A. Horowitz, “*Digital Circuit Optimization via Geometric Programming*”.
- [8] “*NanoTime and NanoTime Ultra User Guide*”, version B-2008.06.
- [9] T. Sakurai, A.R. Newton, “*Solid-State Circuits*”, *IEEE Journal Of*, vol. 25, no 2, pp. 584-594, April 1990.
- [10] Δ. Λιούπης, Μ. Στεφανιδάκης, “*Βασικές Έννοιες Ψηφιακών Κυκλωμάτων*”.
- [11] Jan M. Rabaey, “*Digital Integrated Circuits A Design Perspective (2nd Edition)*”, 2002.



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΙΑΣ



004000073666