

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ,
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΔΙΚΤΥΩΝ



Ιεραρχικές Μέθοδοι Προσομοίωσης για Δίκτυα Τροφοδοσίας
Ολοκληρωμένων Κυκλωμάτων Μεγάλης Κλίμακας

Hierarchical Methods for Large-Scale Power Delivery Network
Analysis

Διπλωματική Εργασία

Ιωαννίδης Κ. Σταύρος

Βόλος, Σεπτέμβριος 2012

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ,
ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΔΙΚΤΥΩΝ



Ιεραρχικές Μέθοδοι Προσομοίωσης για Δίκτυα Τροφοδοσίας
Ολοκληρωμένων Κυκλωμάτων Μεγάλης Κλίμακας

Hierarchical Methods for Large-Scale Power Delivery Network
Analysis

Διπλωματική Εργασία

Ιωαννίδης Κ. Σταύρος

Επιβλέποντες Καθηγητές: Ευμορφόπουλος Νέστωρ
Επίκουρος Καθηγητής

Σταμούλης Γεώργιος
Καθηγητής

Εγκρίθηκε από την διμελή εξεταστική επιτροπή την 2^η Οκτωβρίου 2012

.....
Ευμορφόπουλος Νέστωρ
Επίκουρος Καθηγητής

.....
Σταμούλης Γεώργιος
Καθηγητής

Διπλωματική Εργασία για την απόκτηση του Διπλώματος του Μηχανικού Ηλεκτρονικών Υπολογιστών, Τηλεπικοινωνιών και Δικτύων του Πανεπιστημίου Θεσσαλίας, στα πλαίσια του Προγράμματος Προπτυχιακών Σπουδών του Τμήματος Μηχανικών Η/Υ, Τηλεπικοινωνιών και Δικτύων του Πανεπιστημίου Θεσσαλίας.

(Υπογραφή)

.....

Ιωαννίδης Σταύρος

Copyright ©Ioannidis Stavros, 2012

Με επιφύλαξη παντός δικαιώματος. All rights reserved. Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Στην οικογένειά μου

Ευχαριστίες

Με την περάτωση της παρούσης εργασίας, θα ήθελα να ευχαριστήσω θερμά τους επιβλέποντες καθηγητές μου κ. Ευμορφόπουλο Νέστορα και κ. Σταμούλη Γεώργιο, για την δυνατότητα που μου έδωσαν να ασχοληθώ με το συγκεκριμένο θέμα και για την άριστη συνεργασία μας.

Επίσης θα ήθελα να ευχαριστήσω τον διδάκτορα του τμήματος Μπουντά Δημήτριο καθώς και τους υποψήφιους διδάκτορες Νταλούκα Κωνσταντή, Δρασίδη Γεώργιο και Γσιαμπά Μηχαήλ, για την άμεση μετάδοση γνώσεων, την παροχή τεχνικών συμβουλών και την βοήθεια τους σχετικά με την επίλυση διαφόρων προβλημάτων που παρουσιάστηκαν.

Ένα μεγάλο ευχαριστώ θα ήθελα να δώσω στην ομάδα της Helic S.A. για την βοήθειά της στα πλαίσια της πρακτικής μου άσκησης και για την διάθεση υπερσύγχρονων υπολογιστικών συστημάτων αλλά και λογισμικού, η χρήση των οποίων κατέστησε δυνατή την περάτωση της παρούσης εργασίας μέσα σε λογικά χρονικά πλαίσια. Ιδιαίτερες ευχαριστίες θα ήθελα να απευθύνω στον διευθύνοντα σύμβουλο της Helic S.A. κ. Μπαντά Σωτήριο, για την ουσιώδη καθοδήγηση του καθώς και για την ευκαιρία να ασχοληθώ με την ερευνητική δραστηριότητα της εταιρίας.

Τέλος, θα ήθελα να δώσω τις θερμότερες ευχαριστίες από τα βάθη της καρδιάς μου, στην οικογένεια μου, για την άοκνη ψυχική αλλά και οικονομική στήριξη, την αμέριστη αγάπη και την κατανόηση που έδειξαν κατά την διάρκεια εκπόνησης της παρούσης εργασίας αλλά και καθ' όλη την ακαδημαϊκή μου πορεία.

Περιεχόμενα

Ευχαριστίες	9
Κατάλογος εικόνων	12
Περίληψη	13
Λέξεις κλειδιά.....	13
1. Εισαγωγή.....	14
1.1 Στόχος της εργασίας	14
1.2 Διάρθρωση της εργασίας.....	14
1.3 Περιγραφή προβλήματος της IR drop ανάλυσης.....	14
1.3.1 Ορισμός της πτώσης τάσης	14
1.3.2 Λοιπά προβλήματα αξιοπιστίας	15
1.4 Η σημαντικότητα της IR drop ανάλυσης.....	15
1.5 Η IR drop ανάλυση στον σχεδιασμό του τσιπ.....	16
1.6 Υπάρχουσες μέθοδοι για IR drop ανάλυση	16
1.6.1 Στατική IR drop ανάλυση.....	16
1.6.2 Δυναμική IR drop ανάλυση	17
1.6.3 Ταχύτερες μέθοδοι για Δυναμική IR drop ανάλυση.....	17
2. Παρουσίαση Ιεραρχικής Μεθόδου.....	19
2.1 Βασική ιδέα.....	19
2.2 Τμηματοποίηση	19
2.2.1 Τμηματοποίηση του ψηφιακού κυκλώματος.....	19
2.2.2 Τμηματοποίηση του δικτύου τροφοδοσίας	20
2.3 Βήματα ιεραρχικής μεθόδου	20
2.4 Χρονικά οφέλη μεθόδου.....	20
2.5 Προβλήματα σύγκλισης και λύσεις τους	21
2.5.1 Περιοριστικός παράγοντας	21
2.5.2 Εξομάλυνση κυματομορφών εξόδου	22
3. Υλοποίηση Ιεραρχικής Μεθόδου.....	26
3.1 Δομή υλοποίησης	26
3.2 Το κύκλωμα C1355	26
3.3 Το δίκτυο τροφοδοσίας	27
3.4 Προσομοίωση με FineSim SPICE.....	28
4 Πειραματικά αποτελέσματα υλοποίησης.....	30
4.1 Παρουσίαση αποτελεσμάτων	30

4.2	Αξιολόγηση αποτελεσμάτων.....	34
	Παράρτημα	37
	Βιβλιογραφία.....	51

Κατάλογος εικόνων

Εικόνα 1 - Παράδειγμα δυναμικής IR drop ανάλυσης.....	15
Εικόνα 2 - Δομή κυκλώματος για ταχύτερη δυναμική IR drop ανάλυση.....	17
Εικόνα 3 - Δομή κυκλώματος για ιεραρχική μέθοδο.....	19
Εικόνα 4 - Επίδραση εξομάλυνσης κυματομορφής με SMA.....	23
Εικόνα 5 - Επίδραση εξομάλυνσης κυματομορφής με WMA και $k=3$	24
Εικόνα 6 - Επίδραση εξομάλυνσης κυματομορφής με WMA και $k=7$	24
Εικόνα 7 - Επίδραση εξομάλυνσης κυματομορφής με WMA και $k=20$	25
Εικόνα 8 - Δομή της υλοποίησης της ιεραρχικής μεθόδου.....	26
Εικόνα 9 - Στοιχεία του σχεδίου του κυκλώματος C1355.....	26
Εικόνα 10 - Τυχαία λογική είσοδος για το ψηφιακό κύκλωμα.....	27
Εικόνα 11 - Πηγές τάσης που λειτουργούν ως λογικές εισοδοι.....	27
Εικόνα 12 - Το καθολικό τμήμα του δικτύου τροφοδοσίας.....	28
Εικόνα 13 - Το ανώτερο επίπεδο ενός τοπικού τμήματος του δικτύου τροφοδοσίας.....	28
Εικόνα 14 - Παράδειγμα μετατροπής αποτελεσμάτων σε πηγές τάσης τύπου PWL.....	29
Εικόνα 15 - Αποτελέσματα τάσεων πριν την εφαρμογή περιοριστικού παράγοντα.....	30
Εικόνα 16 - Αποτελέσματα τάσεων μετά την εφαρμογή περιοριστικού παράγοντα.....	31
Εικόνα 17 - Μέσο σχετικό σφάλμα ιεραρχικής μεθόδου.....	35
Εικόνα 18 - Μέγιστο σχετικό σφάλμα ιεραρχικής μεθόδου.....	35

Περίληψη

Καθώς οι παρασιτικές ιδιότητες των δικτύων τροφοδοσίας ολοκληρωμένων κυκλωμάτων αυξάνονται, η ανάγκη για προσομοίωση με σκοπό την λήψη πληροφορίας σχετικά με την πτώση τάσης, γίνεται ολοένα και πιο απαραίτητη. Μέθοδοι για IR drop ανάλυση που χρησιμοποιούνται σήμερα, είναι ή αρκετά πολύπλοκες και χρονοβόρες ώστε να εφαρμοστούν σε κυκλώματα μεγάλης κλίμακας ή παρέχουν προσεγγιστικές λύσεις πολύ χαμηλής ακρίβειας.

Στην παρούσα εργασία γίνεται παρουσίαση και δοκιμαστική υλοποίηση ενός παράλληλου επαναληπτικού αλγορίθμου που εκμεταλλεύεται την ιεραρχία του δικτύου τροφοδοσίας και υπολογίζει με ακρίβεια την πτώση τάσης.

Λέξεις κλειδιά

Πτώση τάσης, Δυναμική IR drop ανάλυση, Δίκτυα τροφοδοσίας ολοκληρωμένων κυκλωμάτων, Ιεραρχικά δίκτυα τροφοδοσίας, Προσομοίωση κυκλωμάτων.

1. Εισαγωγή

1.1 Στόχος της εργασίας

Στόχος της παρούσης εργασίας είναι η παρουσίαση μιας μεθόδου προσομοίωσης του δικτύου τροφοδοσίας ενός ολοκληρωμένου κυκλώματος, με σκοπό την εξαγωγή αξιόπιστων αποτελεσμάτων σχετικά με την πτώση τάσης που αυτό προκαλεί. Ένα σύγχρονο ολοκληρωμένο κύκλωμα μεγάλης κλίμακας, αποτελείται από μερικές εκατοντάδες εκατομμύρια στοιχεία και η προσομοίωση του δικτύου τροφοδοσίας του, απαιτεί τεράστιο πλήθος υπολογισμών. Ακόμα και με χρήση τελευταίας τεχνολογίας υπολογιστικών συστημάτων και λογισμικών προσομοίωσης, η προσομοίωση του δικτύου τροφοδοσίας για ένα τέτοιας κλίμακας τσιπ είναι πρακτικά αδύνατη. Έτσι, οι διάφοροι μέθοδοι που χρησιμοποιούνται σήμερα, μπορούν να προσφέρουν απλώς μια χονδρική εκτίμηση των αποτελεσμάτων της προσομοίωσης μέσα σε λογικά χρονικά πλαίσια.

1.2 Διάρθρωση της εργασίας

Η μέθοδος που παρουσιάζεται στην παρούσα εργασία, καθιστά δυνατή την προσομοίωση μεγάλης κλίμακας κυκλωμάτων και παρέχει πολύ μεγαλύτερης ακρίβειας αποτελέσματα, σε σχέση με αυτά που παράγουν άλλες σύγχρονες μέθοδοι. Η διάρθρωση της εργασίας έχει ως εξής: Στο πρώτο κεφάλαιο γίνεται μια εισαγωγή στο πρόβλημα της IR drop ανάλυσης και παρέχεται το απαραίτητο θεωρητικό υπόβαθρο. Στο δεύτερο κεφάλαιο περιγράφεται αναλυτικά η ιεραρχική μέθοδος, ενώ στο τρίτο κεφάλαιο παρουσιάζεται μια δοκιμαστική υλοποίησή της. Στο τέταρτο κεφάλαιο γίνεται παρουσίαση και αξιολόγηση των αποτελεσμάτων της υλοποίησης. Τέλος, στο παράρτημα επισυνάπτονται ενδεικτικά τμήματα του κώδικα που αναπτύχθηκε στα πλαίσια της υλοποίησης της ιεραρχικής μεθόδου.

1.3 Περιγραφή προβλήματος της IR drop ανάλυσης

Σε ένα ψηφιακό κύκλωμα, κάθε στοιχείο (τρανζίστορ ή πύλη) είναι συνδεδεμένο με την πηγή τροφοδοσίας, από την οποία και αντλεί την ενέργειά του. Για τον διαμοιρασμό του ρεύματος σε όλα τα στοιχεία ενός κυκλώματος, χρησιμοποιείται ένα δίκτυο τροφοδοσίας. Λόγο των παρασιτικών ιδιοτήτων του μη ιδανικού δικτύου αυτού, παρουσιάζεται μια διαφορά στην τάση που «βλέπει» κάθε στοιχείο, σε σχέση με την τάση που παρέχει η πηγή τροφοδοσίας. Η μέθοδος που παρουσιάζεται στην παρούσα εργασία, αφορά την προσομοίωση του δικτύου τροφοδοσίας, με σκοπό τον υπολογισμό της πτώσης τάσης αυτής.

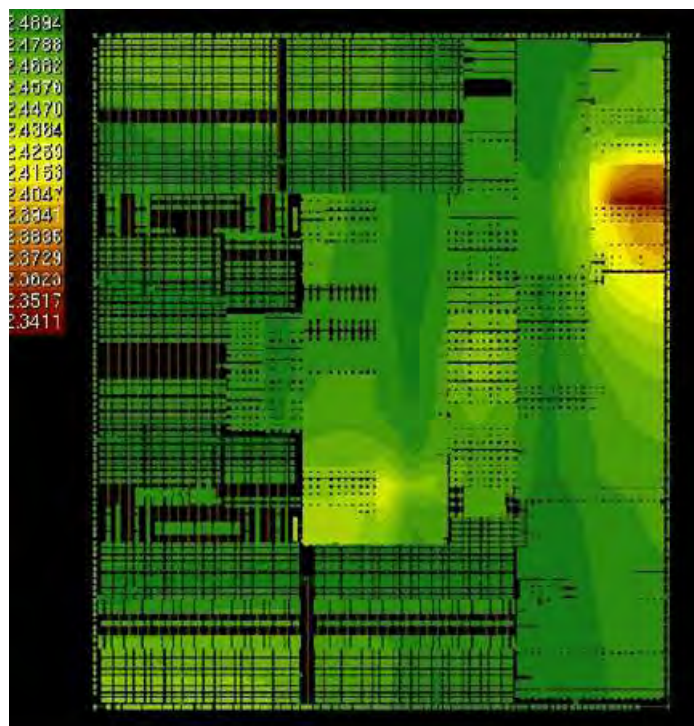
1.3.1 Ορισμός της πτώσης τάσης

Πτώση τάσης (voltage drop) ονομάζουμε τη μείωση που παρατηρείται στην τάση τροφοδοσίας ενός ηλεκτρικού κυκλώματος, καθώς το ρεύμα διαρρέει τα παθητικά του στοιχεία. Η πτώση αυτή οφείλεται στην αντίσταση που έχουν τα παθητικά στοιχεία. Για τον υπολογισμό της πτώσης τάσης χρησιμοποιούμε τον νόμο του Ohm, οποίος λέει πως η διαφορά στην τάση (V) που παρουσιάζουν δύο σημεία-κόμβοι στο κύκλωμα, ισούται με το γινόμενο της αντίστασης του κλάδου μεταξύ των δύο σημείων (R) και του ρεύματος που διαρρέει τον κλάδο (I):

$$V = IR$$

Λόγω της παραπάνω σχέσης, η πτώση τάσης ονομάζεται και IR drop και η διαδικασία υπολογισμού της ονομάζεται IR drop ανάλυση.

Συνήθως, όσο πιο μακριά από την πηγή τροφοδοσίας βρίσκεται ένας κόμβος, τόσο μικρότερη τάση θα έχει. Στην Εικόνα 1 φαίνεται ένα αποτέλεσμα δυναμικής IR drop ανάλυσης σε κάποιο μεγάλης κλίμακας κύκλωμα. Τα διαφορετικά χρώματα αναπαριστούν τις διαφορετικές τάσεις.



Εικόνα 1 - Παράδειγμα δυναμικής IR drop ανάλυσης

1.3.2 Λοιπά προβλήματα αξιοπιστίας

Πέρα από την πτώση τάσης λόγω των παρασιτικών αντιστάσεων του δικτύου τροφοδοσίας, υπάρχουν και άλλα προβλήματα που επηρεάζουν την αξιοπιστία του κυκλώματος. Τέτοια είναι το πρόβλημα της μεταβαλλόμενης γείωσης (ground bounce) που προκαλείται λόγω των παρασιτικών αυτεπαγωγών, καθώς και το πρόβλημα της ηλεκτρομετακίνησης (electromigration) που προκαλείται λόγω εκτεταμένων συγκεντρώσεων ρεύματος. Η παρούσα εργασία δεν ασχολείται με τα προβλήματα αυτά, λόγω της μικρότερης επίδρασής που έχουν στο κύκλωμα σε σχέση με το πρόβλημα της πτώσης τάσης.

1.4 Η σημαντικότητα της IR drop ανάλυσης

Με την εξέλιξη της τεχνολογίας στον τομέα των ολοκληρωμένων κυκλωμάτων, οι διαστάσεις ενός τσιπ σιλικόνης μειώνονται συνεχώς ενώ παράλληλα αυξάνεται το πλήθος των στοιχείων και η συχνότητα λειτουργίας. Αυτά έχουν ως αποτέλεσμα την αύξηση της αντίστασης των καλωδίων αλλά και την αύξηση του ρεύματος που τα διαρρέει. Τα παραπάνω, σε συνδυασμό με την συνεχή μείωση της τάσης τροφοδοσίας, συνεπάγονται μειούμενα περιθώρια για θόρυβο. Συνεπώς, σε μελλοντικές γενιές ολοκληρωμένων κυκλωμάτων, το πρόβλημα του IR drop αναμένεται να είναι εντονότερο και η IR drop ανάλυση πιο απαραίτητη.

Η IR drop ανάλυση είναι απολύτως απαραίτητη για τεχνολογίες μικρότερες των 130nm, όπου τα παρασιτικά επηρεάζουν περισσότερο την λειτουργία του κυκλώματος. Για τέτοιες τεχνολογίες, η παραμέληση εκτέλεσης IR drop ανάλυσης μπορεί να επηρεάσει τον χρονισμό του τσιπ και την λειτουργικότητα και τελικώς να οδηγήσει στην αποτυχία του τσιπ (silicon failure).

1.5 Η IR drop ανάλυση στον σχεδιασμό του τσιπ

Η προσομοίωση του δικτύου τροφοδοσίας γίνεται κατά τη διαδικασία του φυσικού σχεδιασμού (physical design). Σε περίπτωση που η ανάλυση υποδεικνύει προβλήματα στη λειτουργία του τσιπ, η ομάδα σχεδιασμού καλείται να προβεί σε διάφορες ενέργειες για την επίλυσή τους, όπως για παράδειγμα: εισαγωγή πρόσθετων πυκνωτών απόζευξης (decoupling capacitors), τροποποίηση της δομής του δικτύου τροφοδοσίας, εισαγωγή βραχυκυκλωτήρων (jumpers), τροποποίηση του πλήθους των ακροδεκτών τροφοδοσίας (power pins) κ.α.

Πέρα από την on-chip πτώση τάσης, έχουμε πτώση τάσης και στο package αλλά και στο board [4]. Η παρούσα εργασία επικεντρώνεται στην on-chip πτώση τάσης. Ο υπολογισμός την πτώσης τάσης στο package και στο board, είναι σχετικά απλούστερος και μπορεί εύκολα να ενσωματωθεί στις μεθόδους που παρουσιάζουμε, απλά επεκτείνοντας το δίκτυο τροφοδοσίας με την προσθήκη νέων στοιχείων.

1.6 Υπάρχουσες μέθοδοι για IR drop ανάλυση

Το δίκτυο τροφοδοσίας περιγράφεται ως ένα γραμμικό κύκλωμα αποτελούμενο από αντιστάσεις, χωρητικότητες και αυτεπαγωγές, το οποίο εξομοιώνει τις παρασιτικές ιδιότητες του δικτύου. Το ψηφιακό κύκλωμα από την άλλη, μπορεί να περιγραφεί σε μορφή RTL, gate level ή transistor level.

1.6.1 Στατική IR drop ανάλυση

Η στατική IR drop ανάλυση σχεδιάστηκε ώστε να παρέχει πλήρη κάλυψη, χωρίς την απαίτηση εκτεταμένων προσομοιώσεων. Μια τυπική στατική ανάλυση ακολουθεί τα παρακάτω βήματα:

1. Εξαγωγή των παρασιτικών αντιστάσεων του δικτύου τροφοδοσίας,
2. Κατασκευή αντίστοιχου πίνακα από αντιστάσεις,
3. Υπολογισμός του μέσου ρεύματος που καταναλώνει κάθε τρανζίστορ ή πύλη που συνδέεται στο δίκτυο τροφοδοσίας,
4. Διαμερισμός του μέσου ρεύματος στον πίνακα αντιστάσεων σύμφωνα με την φυσική θέση των τρανζίστορ ή πυλών,
5. Σύνδεση πηγών τροφοδοσίας σε κάθε είσοδο τροφοδοσίας,
6. Υπολογισμός της πτώσης τάση που προκαλεί ο πίνακας αντιστάσεων, προσομοιώνοντας με χρήση κάποιου στατικού επιλυτή.

Λόγο της απλοϊκής μορφής των παρασιτικών, που μοντελοποιούνται εξολοκλήρου με αντιστάσεις, η στατική ανάλυση έχει σημαντικά μειωμένο χρόνο προσομοίωσης. Η χρήση

την μέση κατανάλωσης ρεύματος από την άλλη, παρέχει μια πρόχειρη προσέγγιση της πραγματικής πτώσης τάσης που προκαλεί το δίκτυο τροφοδοσίας.

1.6.2 Δυναμική IR drop ανάλυση

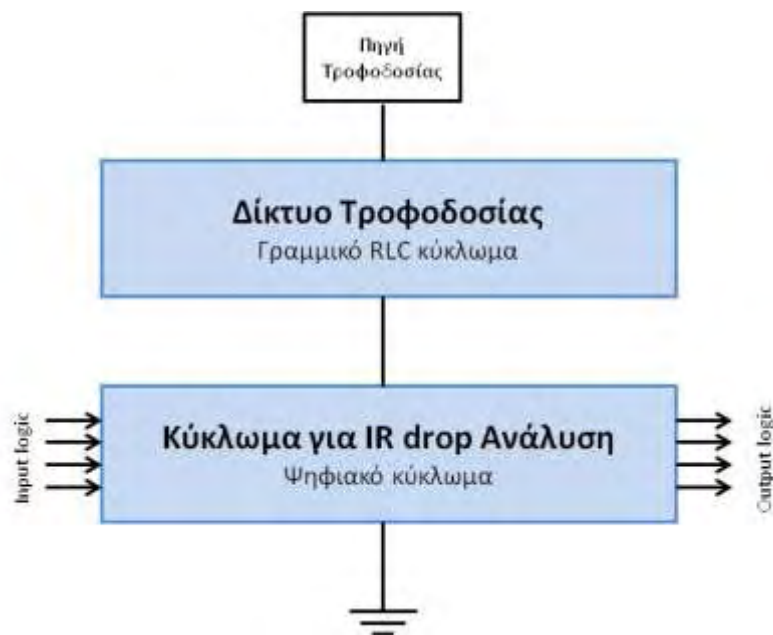
Κατά την δυναμική IR drop ανάλυση χρησιμοποιούνται οι παρασιτικές αντιστάσεις αλλά και οι παρασιτικές χωρητικότητες του δικτύου τροφοδοσίας. Μια τυπική δυναμική ανάλυση ακολουθεί τα παρακάτω βήματα:

1. Εξαγωγή της παρασιτικής αντίστασης και παρασιτικής χωρητικότητας του δικτύου τροφοδοσίας,
2. Σχεδιασμός ενός κυκλώματος σε μορφή λίστας στοιχείων (netlist), το οποίο περιέχει το ψηφιακό κύκλωμα μαζί με το RC κύκλωμα των παρασιτικών,
3. Προσομοίωση του συνολικού κυκλώματος με χρήση εργαλείων προσομοίωσης, δίνοντας συγκεκριμένες λογικές εισόδους στο ψηφιακό κύκλωμα.

Η δυναμική ανάλυση επικεντρώνεται στην ακρίβεια των αποτελεσμάτων. Όμως, η πολυπλοκότητα των υπολογισμών που απαιτείται κατά την προσομοίωση, καθιστά την δυναμική ανάλυση απαγορευτική για μεγάλης κλίμακας κυκλώματα.

1.6.3 Ταχύτερες μέθοδοι για Δυναμική IR drop ανάλυση

Ταχύτερες μέθοδοι για δυναμική IR drop ανάλυση, χωρίζουν το δίκτυο τροφοδοσίας και το ψηφιακό κύκλωμα σε δύο ανεξάρτητα τμήματα που προσομοιώνονται ξεχωριστά, όπως φαίνεται στην Εικόνα 2. Με αυτό τον τρόπο η τάξη μεγέθους του προς επίλυση προβλήματος μειώνεται σημαντικά. Επίσης κατά την προσομοίωση του τμήματος που μοντελοποιεί τα παρασιτικά του δικτύου τροφοδοσίας, αξιοποιούνται οι δυνατότητες που προσφέρουν τα σύγχρονα εργαλεία προσομοίωσης, για ταχεία προσομοίωση γραμμικών κυκλωμάτων.



Εικόνα 2 - Δομή κυκλώματος για ταχύτερη δυναμική IR drop ανάλυση

Σε πρώτο στάδιο προσομοιώνεται το ψηφιακό κύκλωμα θεωρώντας ιδανική τροφοδοσία. Σαν αποτέλεσμα την προσομοίωσης λαμβάνεται το ρεύμα που καταναλώνει κάθε στοιχείο του κυκλώματος. Σε δεύτερο βήμα, προσομοιώνεται το κύκλωμα που περιγράφει το δίκτυο τροφοδοσίας, στο οποίο έχουν προστεθεί τα ρεύματα που καταναλώνει το ψηφιακό κύκλωμα (τα οποία υπολογίστηκαν κατά το πρώτο βήμα), σε μορφή ανεξάρτητων πηγών ρεύματος. Το αποτέλεσμα του δεύτερου βήματος είναι οι ζητούμενες τάσεις, δηλαδή οι τάσεις που «βλέπουν» τα στοιχεία του ψηφιακού κυκλώματος.

Η παραπάνω μέθοδος δίνει μια προσέγγιση της πραγματικής πτώσης τάσης, όπου πραγματική πτώση τάσης θεωρούμε αυτή που θα λαμβάναμε, αν προσομοιώναμε το ψηφιακό κύκλωμα και το δίκτυο τροφοδοσίας μαζί. Η πτώση τάσης της παραπάνω μεθόδου είναι συνήθως αρκετά μεγαλύτερη από την πραγματική. Αυτό προκύπτει από το γεγονός ότι για τον υπολογισμό των ρευμάτων που καταναλώνουν τα στοιχεία του ψηφιακού κυκλώματος, χρησιμοποιείται ιδανική τροφοδοσία, αντί της πραγματικής τάσης η οποία θα είναι προφανώς μικρότερη. Κατά συνέπεια το ψηφιακό κύκλωμα θα φαίνεται να καταναλώνει περισσότερο ρεύμα από ότι στην πραγματικότητα, το οποίο όταν αργότερα χρησιμοποιηθεί στην προσομοίωση του δικτύου τροφοδοσίας θα προκαλέσει μεγαλύτερη πτώση τάσης.

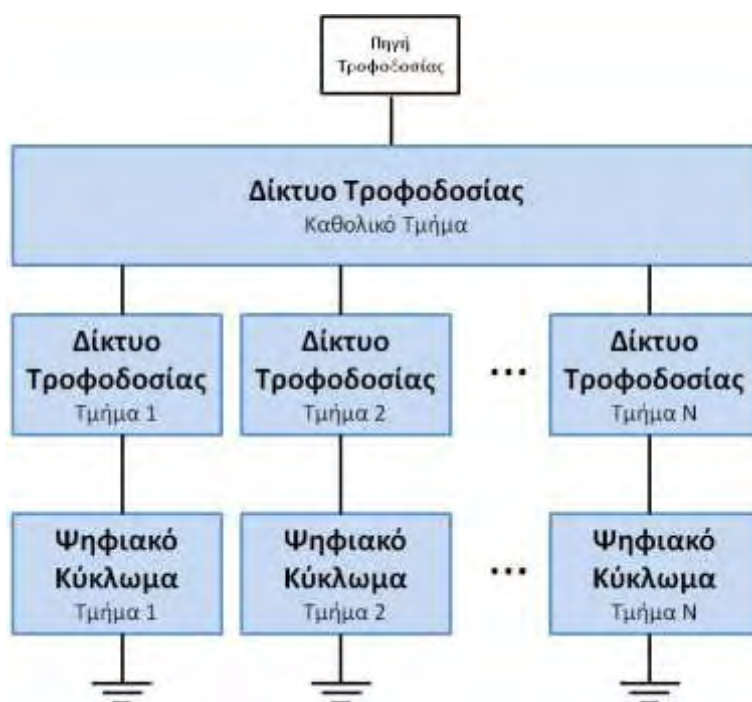
Μια πολύ χονδρική προσέγγιση μπορεί να οδηγήσει σε άλλου είδους προβλήματα. Για παράδειγμα αν η IR drop ανάλυση υποδείξει αρκετά μεγαλύτερη πτώση τάσης από την πραγματική, αυτό μπορεί να οδηγήσει στην προσθήκη νέου υλικού (π.χ. πυκνωτές απόξυξης), σπαταλώντας επιφάνεια επάνω στο τσιπ και κάνοντας το σχέδιο λιγότερο ανταγωνιστικό. Συνεπώς η βελτίωση της ακρίβεια στην εκτέλεση IR drop ανάλυσης είναι κάτι σημαντικό.

Βελτίωση της ακρίβειας των αποτελεσμάτων που παράγονται με την παραπάνω μέθοδο, μπορεί να επιτευχθεί εκτελώντας περισσότερες επαναλήψεις των δύο βημάτων. Σε κάθε επανάληψη αντί της ιδανικής τάσης, θα πρέπει να χρησιμοποιούνται οι τάσεις που υπολογίστηκαν κατά την τελευταία επανάληψη. Έτσι με κάθε επανάληψη θα βελτιώνεται η εκτίμηση της πτώσης τάσης και τελικώς, μετά από κάποιο πλήθος επαναλήψεων, τα αποτελέσματα της μεθόδου θα συγκλίνουν στα πραγματικά. Η μέθοδος αυτή χρησιμοποιείται σήμερα σε κάποια εμπορικά εργαλεία προσομοίωσης. Κανένα όμως εργαλείο δεν υποστηρίζει την προσομοίωση κυκλωμάτων με πλήθος στοιχείων της τάξης των εκατοντάδων εκατομμυρίων.

2. Παρουσίαση Ιεραρχικής Μεθόδου

2.1 Βασική ιδέα

Η βασική ιδέα της ιεραρχικής μεθόδου βασίζεται στην επαναληπτική μέθοδο των δύο βημάτων, που περιγράφηκε στην παράγραφο 1.6.3. Εκμεταλλευόμενη την ιεραρχική δομή του δικτύου τροφοδοσίας, η ιεραρχική μέθοδος χρησιμοποιεί το δίκτυο τροφοδοσίας και το ψηφιακό κύκλωμα, χωρισμένα σε μικρότερα τμήματα. Πιο συγκεκριμένα, η μέθοδος θεωρεί πως το ψηφιακό κύκλωμα παρέχεται σε N ανεξάρτητα τμήματα, κάθε ένα από τα οποία συνδέεται με ένα από τα N τμήματα του δικτύου τροφοδοσίας. Επίσης, παρέχεται ένα καθολικό τμήμα του δικτύου τροφοδοσίας το οποίο συνδέει τα N επιμέρους τμήματα του δικτύου, όπως φαίνεται στην Εικόνα 3.



Εικόνα 3 - Δομή κυκλώματος για ιεραρχική μέθοδο

2.2 Τμηματοποίηση

2.2.1 Τμηματοποίηση του ψηφιακού κυκλώματος

Η δομή ενός ψηφιακού κυκλώματος περιγράφεται, από τα πρώτα κιόλας στάδια της σχεδίασης του, σε μια γλώσσα περιγραφής υλικού όπως η Verilog. Ένα τέτοιο ψηφιακό κύκλωμα, έχει διαρθρωτική δομή (structural), δηλαδή αποτελείται από ένα σύνολο από στοιχειώδη εξαρτήματα (components), όπως για παράδειγμα λογικές πύλες. Πολλαπλά αντίγραφα των εξαρτημάτων αυτών συνδυάζονται και συνθέτουν μεγαλύτερα εξαρτήματα, όπως για παράδειγμα έναν πλήρη αθροιστή 3-bit (full adder). Αυτή η διαδικασία εφαρμόζεται αναδρομικά μέχρι το υψηλότερο επίπεδο του κυκλώματος, στο οποίο το πλήθος των εξαρτημάτων που το συνθέτουν μπορεί να φθάνει τις μερικές δεκάδες.

Η δομή του υψηλότερου επιπέδου, διατηρείται καθ' όλη τη διάρκεια του σχεδιασμού του κυκλώματος, πολλές φορές και μετά τον φυσικό σχεδιασμό του. Τα εξαρτήματα που συνθέτουν το επίπεδο αυτό, μπορούν να χρησιμοποιηθούν ως τα N τμήματα που απαιτεί η ιεραρχική μέθοδος. Συνεπώς η απαίτηση της ιεραρχικής μεθόδου για διαχωρισμό του ψηφιακού κυκλώματος σε N τμήματα, είναι κάτι εφικτό.

2.2.2 Τμηματοποίηση του δικτύου τροφοδοσίας

Ο σχεδιασμός του δικτύου τροφοδοσίας βασίζεται στη δομή του ψηφιακού κυκλώματος. Είναι εύκολο το δίκτυο τροφοδοσίας να σχεδιαστεί έτσι ώστε να αποτελείται από N τμήματα, κάθε ένα από τα οποία θα τροφοδοτεί ένα από τα τμήματα του ψηφιακού κυκλώματος, καθώς και από ένα καθολικό τμήμα το οποίο θα συνδέει τα N επιμέρους τμήματα μεταξύ τους.

2.3 Βήματα ιεραρχικής μεθόδου

Η ιεραρχική μέθοδος αποτελείται από τα εξής βήματα:

Βήμα 1^ο: Ανάλυση μεταβατικής κατάστασης λειτουργίας (transient analysis) των N τμημάτων του ψηφιακού κυκλώματος χρησιμοποιώντας τις τάσεις που υπολογίστηκαν από την τελευταία επανάληψη. Για την πρώτη επανάληψη χρησιμοποιείται ιδανική τάση. Υπολογισμός των ρευμάτων που καταναλώνουν τα στοιχεία των τμημάτων του ψηφιακού κυκλώματος.

Βήμα 2^ο: Ανάλυση μεταβατικής κατάστασης λειτουργίας των N τμημάτων του δικτύου τροφοδοσίας χρησιμοποιώντας τα ρεύματα που υπολογίστηκαν στο 1^ο βήμα. Υπολογισμός των ρευμάτων που αντλεί κάθε τμήμα του δικτύου τροφοδοσίας από το καθολικό τμήμα.

Βήμα 3^ο: Ανάλυση μεταβατικής κατάστασης λειτουργίας του καθολικού τμήματος του δικτύου τροφοδοσίας, χρησιμοποιώντας τα ρεύματα που υπολογίστηκαν στο 2^ο βήμα. Υπολογισμός των τάσεων στα σημεία μεταξύ των N τμημάτων του δικτύου τροφοδοσίας και του καθολικού.

Βήμα 4^ο: Ανάλυση μεταβατικής κατάστασης λειτουργίας των N τμημάτων του δικτύου τροφοδοσίας χρησιμοποιώντας τις τάσεις που υπολογίστηκαν στο 3^ο βήμα. Υπολογισμός των τάσεων στα σημεία μεταξύ των τμημάτων του δικτύου τροφοδοσίας και των τμημάτων του ψηφιακού κυκλώματος.

Τα παραπάνω βήματα εκτελούνται επαναληπτικά. Κάθε επανάληψη βελτιώνει τα αποτελέσματα της προηγούμενης, αφού για κάθε νέα επανάληψη χρησιμοποιούνται οι τελευταίες γνωστές - και πιο ακριβείς - τάσεις, οι οποίες υπολογίστηκαν κατά την τελευταία επανάληψη. Η ιεραρχική μέθοδος συγκλίνει τελικώς στα πραγματικά αποτελέσματα.

2.4 Χρονικά οφέλη μεθόδου

Κατά την ιεραρχική μέθοδο το δίκτυο τροφοδοσίας και το ψηφιακό κύκλωμα προσομοιώνονται ξεχωριστά. Όπως αναφέρεται και στην παράγραφο 1.6.3, αυτό οδηγεί σε σημαντικά μικρότερο χρόνο προσομοίωσης, κυρίως λόγω της ελάττωσης της τάξης μεγέθους του προς επίλυση προβλήματος.

Η ιεραρχική μέθοδος προσφέρει χώρο για περαιτέρω βελτιστοποιήσεις στον συνολικό χρόνο προσομοίωσης, που βασίζονται σε παράλληλη εκτέλεση τμημάτων της μεθόδου. Πιο συγκεκριμένα, λόγω της απουσίας άμεσης σύνδεσης μεταξύ των N τμημάτων του ψηφιακού κυκλώματος αλλά και μεταξύ των N τμημάτων του δικτύου τροφοδοσίας, τα βήματα 1, 2 και 4 της ίδιας επανάληψης της ιεραρχικής μεθόδου, μπορούν να εκτελεστούν ανεξάρτητα. Για την εκτέλεση του βήματος 3, θα πρέπει να έχουν ολοκληρωθεί οι υπολογισμοί του βήματος 2 και για τα N τμήματα. Τέλος, υπολογισμοί που αφορούν διαφορετικές επαναλήψεις, δεν δύναται να εκτελεστούν παράλληλα, λόγω της εξάρτησης δεδομένων που υπάρχει μεταξύ των τάσεων που υπολογίζει η μια επανάληψη και αυτών που χρησιμοποιεί η επόμενη.

2.5 Προβλήματα σύγκλισης και λύσεις τους

Σε περιπτώσεις που το δίκτυο τροφοδοσίας προκαλεί σημαντική πτώση τάσης (της τάξης του 10% της τάσης τροφοδοσίας), τα αποτελέσματα της πρώτης επανάληψης της ιεραρχικής μεθόδου, μπορεί αποκλίνουν πολύ από τα πραγματικά. Η απόκλιση αυτή δεν περιορίζεται μόνο στην συνιστώσα της τάσης, αλλά επεκτείνεται και στην συνιστώσα του χρόνου, διότι τα στοιχεία του ψηφιακού κυκλώματος λειτουργούν με διαφορετική ταχύτητα για διαφορετικές τάσεις στην είσοδό τους. Αυτές οι αποκλίσεις μπορεί να καθυστερήσουν την σύγκλιση της ιεραρχικής μεθόδου, να δημιουργήσουν παραμορφώσεις στις κυματομορφές εξόδου και σε ορισμένες περιπτώσεις να οδηγήσουν και σε αποτυχία της μεθόδου.

Για την αντιμετώπιση τέτοιων προβλημάτων και για την εξασφάλιση της σύγκλισης σε κάθε περίπτωση, είναι αναγκαία η χρήση των δύο ακόλουθων τεχνικών:

1. εισαγωγή του περιοριστικού παράγοντα (restraint factor)
2. εξομάλυνση των κυματομορφών εξόδου (waveform relaxation)

Στις παραγράφους που ακολουθούν γίνεται μια παρουσίαση των ανωτέρω τεχνικών.

2.5.1 Περιοριστικός παράγοντας

Ο περιοριστικός παράγοντας είναι ένας συντελεστής που περιορίζει τις τάσεις που υπολογίζει κάθε επανάληψη της ιεραρχικής μεθόδου. Πιο συγκεκριμένα, συνδυάζει ένα ποσοστό των αποτελεσμάτων της τελευταίας επανάληψης με ένα ποσοστό των αποτελεσμάτων της τρέχουσας για να υπολογίσει τα τελικά αποτελέσματα της τρέχουσας επανάληψης, αποτρέποντας έτσι ανεξέλεγκτες αλλαγές στις τάσεις εξόδου. Ο συντελεστής αυτός (έστω C) λαμβάνει τιμές στο διάστημα $[0, 1]$ και διαμορφώνει τις τελικές τάσεις που υπολογίζονται σε ένα βήμα της ιεραρχικής μεθόδου, σύμφωνα με την παρακάτω σχέση:

$$V(t) = V_{prev}(t) + C[V_{new}(t) - V_{prev}(t)]$$

Για μικρές τιμές του C , οι ιεραρχική μέθοδος συγκλίνει με αργότερο ρυθμό, αλλά παρέχει αυξημένη αξιοπιστία ενώ για μεγάλες τιμές του C η σύγκλιση επιτυγχάνεται ταχύτερα, αλλά υπάρχει κίνδυνος απόκλισης μετά από κάποιο πλήθος επαναλήψεως και κατά συνέπεια αποτυχίας της μεθόδου. Η ακραία περίπτωση όπου $C=1$, ισοδυναμεί με μη χρήση του περιοριστικού παράγοντα.

Στον παρακάτω πίνακα φαίνονται τα σημεία της ιεραρχικής μεθόδου, στα οποία εμφανίζεται ο περιοριστικός παράγοντας:

- Βήμα 1^ο Ως τάσεις στις πύλες ή τρανζίστορ του ψηφιακού κυκλώματος, δίνεται ένας συνδυασμός των τάσεων που χρησιμοποιήθηκαν στο 1^ο βήμα της προηγούμενης επανάληψης και των τάσεων που υπολογίσθηκαν κατά το 4^ο βήμα της προηγούμενης επανάληψης. Για την πρώτη επανάληψη δίνεται τάση ίση με την τάση τροφοδοσίας.
- Βήμα 2^ο Ως τάσεις στα σημεία όπου τα N τμήματα του δικτύου τροφοδοσίας συνδέονται με το καθολικό τμήμα, δίνεται ένας συνδυασμός των τάσεων που χρησιμοποιήθηκαν στο 2^ο βήμα της προηγούμενης επανάληψης και των τάσεων που υπολογίσθηκαν κατά το 3^ο βήμα της προηγούμενης επανάληψης.
- Βήμα 3^ο Δεν γίνεται χρήση του περιοριστικού παράγοντα λόγο του ότι η τάση που εφαρμόζεται στο καθολικό τμήμα του δικτύου τροφοδοσίας είναι πάντα η ιδανική.
- Βήμα 4^ο Ως τάσεις στις πύλες ή τρανζίστορ του ψηφιακού κυκλώματος δίνονται αυτούσιες οι τάσεις που χρησιμοποιήθηκαν κατά το 1^ο βήμα της τρέχουσας επανάληψης.

Ο περιοριστικός παράγοντας μπορεί να επιβραδύνει τις επιπτώσεις που προκαλούν οι τοπικές ανωμαλίες (spikes) στις κυματομορφές εξόδου, αλλά δεν μπορεί να τις εξαλείψει πλήρως. Μπορεί λοιπόν να εγγραφεί την μερική σύγκλιση της ιεραρχικής μεθόδου, αλλά για ακριβή αποτελέσματα είναι απαραίτητη η εξομάλυνση των κυματομορφών εξόδου.

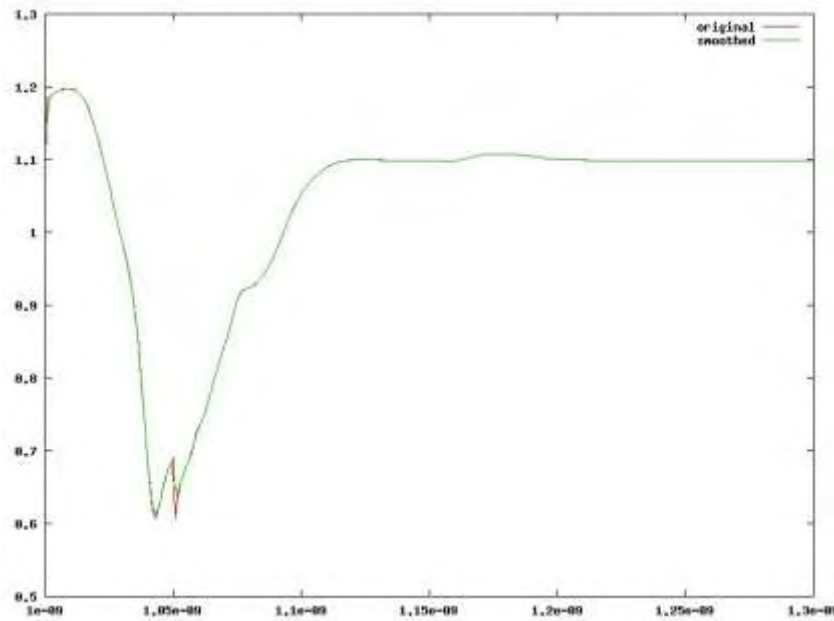
2.5.2 Εξομάλυνση κυματομορφών εξόδου

Η εξομάλυνση των κυματομορφών τάσης αποσκοπεί στην εξάλειψη των τοπικών ανωμαλιών με όσο το δυνατόν μικρότερη παραμόρφωση της «υγιής» πληροφορίας που φέρει η κυματομορφή. Διάφορα είδη εξομάλυνσης που ποικίλουν ανάλογα με την πολυπλοκότητα των υπολογισμών τους και τον βαθμό που παραμορφώνουν την κυματομορφή, μπορούν να επηρεάσουν την τελική ακρίβεια των αποτελεσμάτων της ιεραρχικής μεθόδου αλλά και τον χρόνο που απαιτείται για την σύγκλιση. Λανθασμένη επιλογή της μεθόδου εξομάλυνσης, μπορεί να προκαλέσει προβλήματα στην ιεραρχική μέθοδο καθώς μπορεί να επηρεάσει την ακρίβεια των αποτελεσμάτων της.

Η απλούστερη μέθοδος εξομάλυνσης αφορά την σάρωση των σημείων της κυματομορφής και της αντικατάστασης κάθε τιμής της με έναν μέσο όρο κάποιων σταθερού πλήθους γειτονικών τιμών. Πιο «έξυπνες» παραλλαγές της παραπάνω μεθόδου, χρησιμοποιούν βεβαρημένο μέσο όρο με γραμμικά ή εκθετικά αυξανόμενα βάρη. Άλλες πιο πολύπλοκες μέθοδοι αφορούν επαναληπτικές σαρώσεις της κυματομορφής ή και επίλυση γραμμικών και διαφορικών εξισώσεων.

Στις παρακάτω εικόνες φαίνονται οι επιπτώσεις διάφορων ειδών εξομάλυνσης κυματομορφής σε μια κυματομορφή τάσης. Στην κυματομορφή της Εικόνα 4 έχει εφαρμοστεί η μέθοδος του μετακινούμενου μέσου όρου (simple moving average) η οποία σε μία σάρωση της κυματομορφής, υπολογίζει μια νέα τιμή για κάθε σημείο σύμφωνα με τον τύπο:

$$V_{smoothed}(t) = \frac{V(t-1) + V(t) + V(t+1)}{3}$$



Εικόνα 4 - Επίδραση εξομάλυνσης κυματομορφής με SMA

Η παραπάνω μέθοδος εξομάλυνσης, αν και παραμορφώνει ελάχιστα την κυματομορφή, δεν είναι σε θέση να εξαλείψει έντονες τοπικές ανωμαλίες, όπως αυτή που εμφανίζεται στο σημείο $1.05e-9$.

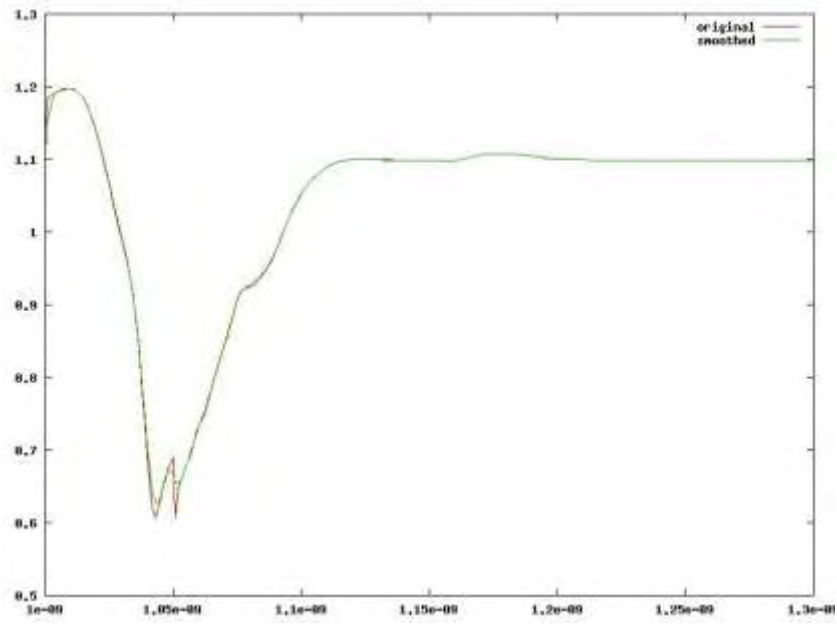
Η μέθοδος εξομάλυνσης που χρησιμοποιήθηκε στις τρεις επόμενες εικόνες, είναι η μέθοδος μετακινούμενου μέσου όρου με γραμμικά βάρη (weighted moving average), η οποία σε μία σάρωση της κυματομορφής, υπολογίζει μια νέα τιμή για κάθε σημείο, λαμβάνοντας τον βεβαρημένο μέσο όρο των k τιμών στα αριστερά και των k τιμών στα δεξιά του σημείου αυτού, σύμφωνα με τον τύπο:

$$V_{smoothed}(t) = \frac{W(k)V(t) + \sum_{i=1}^k W(k-i)[V(t-i) + V(t+i)]}{\sum W(i)}$$

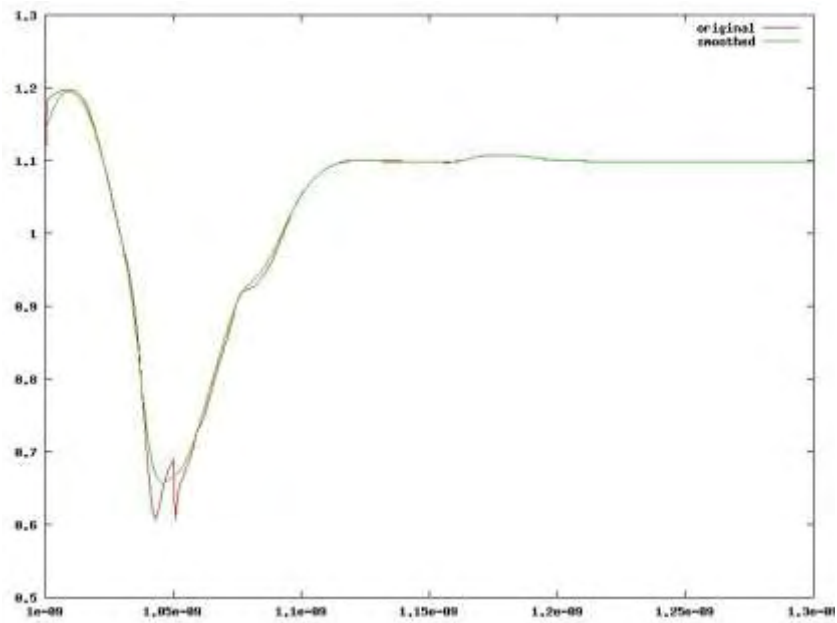
όπου ο κάθε συντελεστής βάρους υπολογίζεται σύμφωνα με τον τύπο:

$$W(i) = i + 1$$

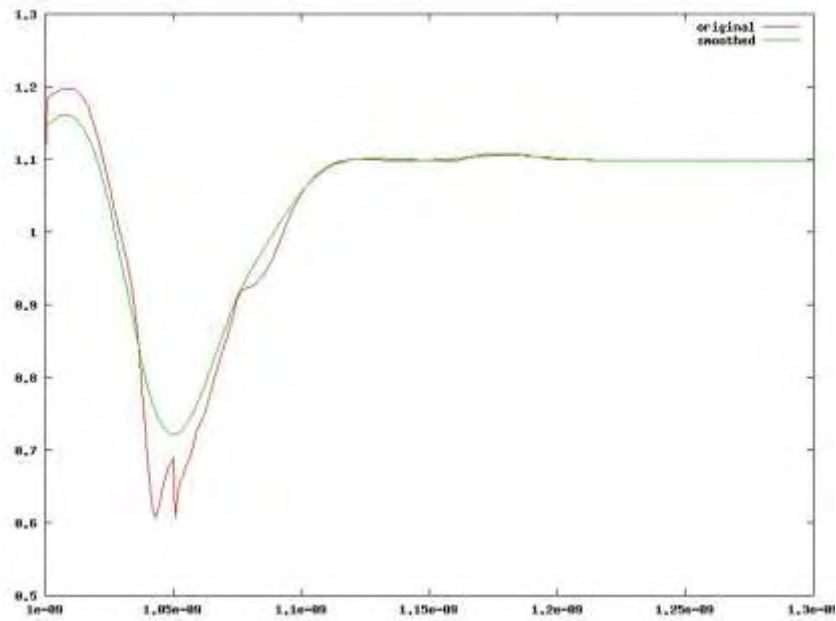
Οι παρακάτω εικόνες είναι για $k=3$, $k=7$ και $k=20$ αντίστοιχα:



Εικόνα 5 - Επίδραση εξομάλυνσης κυματομορφής με WMA και $k=3$



Εικόνα 6 - Επίδραση εξομάλυνσης κυματομορφής με WMA και $k=7$



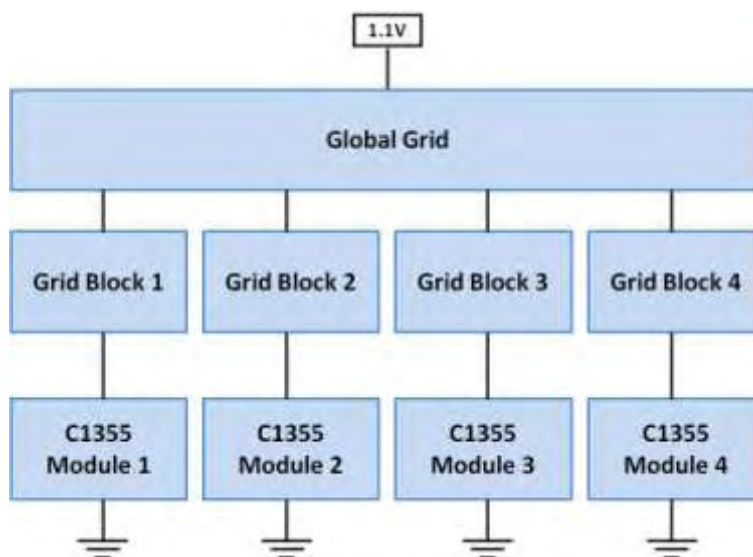
Εικόνα 7 - Επίδραση εξομάλυνσης κυματομορφής με WMA και $k=20$

Είναι εμφανές πως για μεγαλύτερες τιμές του k η μέθοδος εξομάλυνσης έχει ισχυρότερα αποτελέσματα. Αυτό σημαίνει πως μπορεί να παραμορφώσει σε μεγάλο βαθμό την κυματομορφή και να προκαλέσει την απώλεια της ακρίβειας των αποτελεσμάτων της ιεραρχικής μεθόδου.

3. Υλοποίηση Ιεραρχικής Μεθόδου

3.1 Δομή υλοποίησης

Για την υλοποίηση της ιεραρχικής μεθόδου χρησιμοποιήθηκαν 4 αντίγραφα του κυκλώματος C1355 καθώς και ένα αντίστοιχο ιεραρχικό δίκτυο τροφοδοσίας, χωρισμένο σε 4 τοπικά τμήματα (blocks) και ένα καθολικό (global), όπως φαίνεται στην Εικόνα 8.



Εικόνα 8 - Δομή της υλοποίησης της ιεραρχικής μεθόδου

3.2 Το κύκλωμα C1355

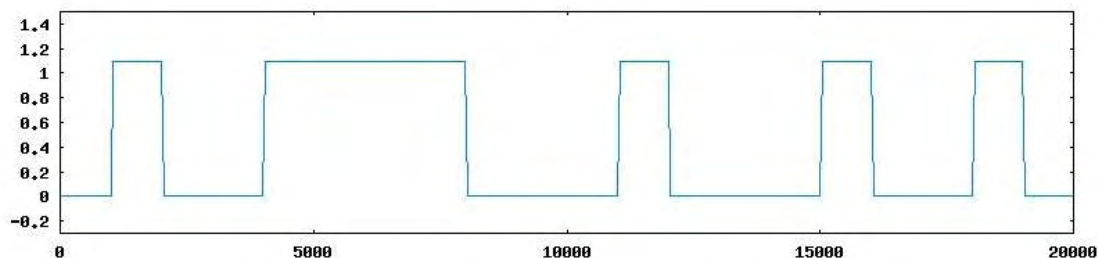
Το C1355 είναι ένα μικρής κλίμακας κύκλωμα το οποίο αποτελείται από 522 πύλες τύπου NAND, NOR και NOT. Ο λόγος που επιλέχθηκε ένα μικρής κλίμακας κύκλωμα, είναι η δυνατότητα προσομοίωσής του μαζί με το δίκτυο τροφοδοσίας (μέσα σε λογικά χρονικά πλαίσια), ώστε να υπολογιστούν τα ιδανικά αποτελέσματα με τα οποία θα συγκριθούν τα αποτελέσματα της ιεραρχικής μεθόδου. Στην Εικόνα 9 παρέχονται κάποια στοιχεία για το κύκλωμα C1355.

Στοιχεία σχεδίου	Πλήθος
Λογικές εισοδοι	41
Λογικές έξοδοι	32
Πύλες	522
Flip flop	0
Επίπεδα	6

Εικόνα 9 - Στοιχεία του σχεδίου του κυκλώματος C1355

Στα 4 στιγμιότυπα του C1355 δίνονται τυχαίες λογικές, με περίοδο 1ns και με πιθανότητα εναλλαγής από λογικό 0 σε λογικό 1 και αντίστροφα, ίση με 40%. Η περίοδος αυτή επιλέχθηκε μετά από δοκιμές, που έδειξαν ότι οι πύλες του κυκλώματος χρειάζεται περίπου

400-450ps για να σταθεροποιηθούν στην τελική λογική τους κατάσταση. Συνεπώς η περίοδος 1ns είναι μια καλή τιμή για περίοδο που εξασφαλίζει αφ' ενός την ορθή λειτουργία του κυκλώματος και αφ' εταίρου ελαχιστοποιεί τον χρόνο για τον οποίο στο κύκλωμα δεν υπάρχει δραστηριότητα. Ένα παράδειγμα μια λογικής εισόδου φαίνεται στην Εικόνα 10 για χρόνο 20 περιόδων. Στον οριζόντιο άξονα φαίνεται η τάση της λογικής εισόδου, όπου η τάση 0V αντιστοιχεί σε λογικό 0, ενώ η τάση 1.1V (τάση τροφοδοσίας) σε λογικό 1.



Εικόνα 10 - Τυχαία λογική είσοδος για το ψηφιακό κύκλωμα

Στον παρακάτω πίνακα φαίνεται ένα ενδεικτικό τμήμα των πηγών τάσης που λειτουργούν ως λογικές εισοδοί για το κύκλωμα C1355:

vMod2G37	Mod2G37	0	PWL(0	0.0	2000p	0.0	2050p	1.1	4000p	1.1	4050p	0.0	8000p	0.0	8050p	1.1)
vMod2G38	Mod2G38	0	PWL(0	0.0	1000p	0.0	1050p	1.1	5000p	1.1	5050p	0.0)				
vMod2G39	Mod2G39	0	PWL(0	0.0	4000p	0.0	4050p	1.1	6000p	1.1	6050p	0.0)				
vMod2G4	Mod2G4	0	PWL(0	0.0	1000p	0.0	1050p	1.1	6000p	1.1	6050p	0.0	8000p	0.0	8050p	1.1)
vMod2G40	Mod2G40	0	PWL(0	0.0	3000p	0.0	3050p	1.1	6000p	1.1	6050p	0.0	7000p	0.0	7050p	1.1)
vMod2G41	Mod2G41	0	PWL(0	0.0	7000p	0.0	7050p	1.1)								
vMod2G5	Mod2G5	0	PWL(0	0.0	5000p	0.0	5050p	1.1	6000p	1.1	6050p	0.0)				
vMod2G6	Mod2G6	0	PWL(0	0.0	1000p	0.0	1050p	1.1	3000p	1.1	3050p	0.0	9000p	0.0	9050p	1.1)
vMod2G7	Mod2G7	0	PWL(0	0.0	3000p	0.0	3050p	1.1	5000p	1.1	5050p	0.0)				

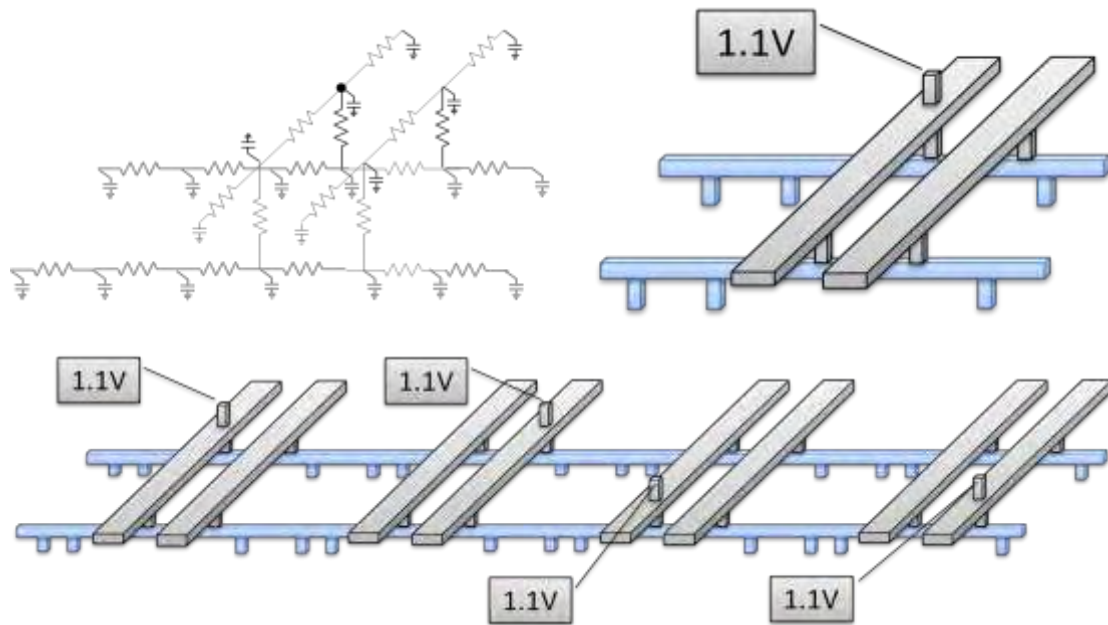
Εικόνα 11 - Πηγές τάσης που λειτουργούν ως λογικές εισοδοί

Στα 4 στιγμιότυπα του κυκλώματος C1355 δόθηκαν διαφορετικές λογικές εισοδοί, ώστε η υλοποίηση να μοιάζει όσο το δυνατόν περισσότερο με μια πραγματική εφαρμογή της ιεραρχικής μεθόδου, όπου τα N τμήματα του ψηφιακού κυκλώματος θα έχουν διαφορετική συμπεριφορά. Η αποθήκευση των τιμών των λογικών εισόδων είναι απαραίτητη, καθ' όλη την διάρκεια εκτέλεσης της ιεραρχικής μεθόδου, ώστε να χρησιμοποιούνται οι ίδιες λογικές εισοδοί στα βήματα 1 και 4 όλων των επαναλήψεων.

3.3 Το δίκτυο τροφοδοσίας

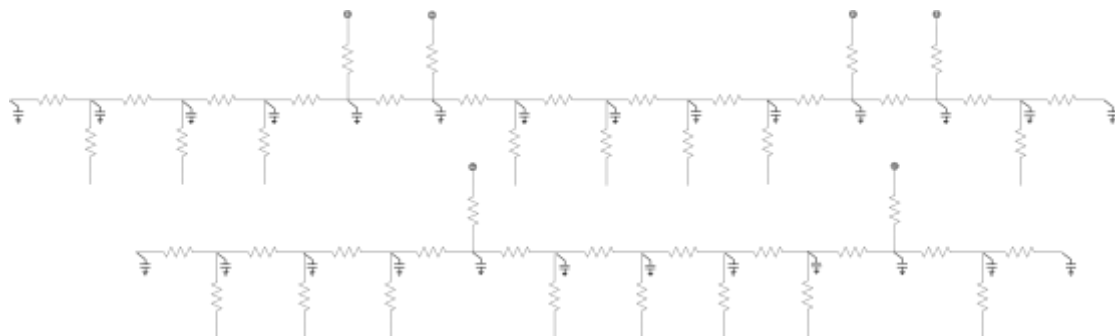
Τα 4 τμήματα του δικτύου τροφοδοσίας καθώς και το καθολικό τμήμα, αποτελούνται από αντιστάσεις και χωρητικότητες. Σε μια φυσική υλοποίηση του συστήματος της Εικόνα 8, το δίκτυο τροφοδοσίας δεν θα προκαλούσε πτώση τάσης μεγαλύτερη του 1% της τάσης τροφοδοσίας. Για τον λόγο αυτό, αλλά και για την μελέτη της συμπεριφοράς της ιεραρχικής μεθόδου σε περιπτώσεις σημαντικού IR drop, οι τιμές των αντιστάσεων τροποποιήθηκαν κατάλληλα, ώστε η πτώση τάσης να φτάνει το 10-20% της τάσης τροφοδοσίας.

Στο επάνω μέρος της Εικόνα 12 φαίνεται η δομή του τμήματος του καθολικού μέρους του δικτύου τροφοδοσίας που οποίο είναι υπεύθυνο για τη διανομή ενέργειας σε ένα αντίγραφο του C1355. Κάθε τέτοιο τμήμα αποτελείται από 2 επίπεδα και έχει 6 ακροδέκτες, μέσω των οποίων συνδέεται με το αντίστοιχο τοπικό τμήμα του δικτύου τροφοδοσίας. Στο κάτω μέρος της εικόνας φαίνεται ολόκληρο το καθολικό τμήμα, καθώς και οι 4 ακροδέκτες τροφοδοσίας (power pins).



Εικόνα 12 - Το καθολικό τμήμα του δικτύου τροφοδοσίας

Στην Εικόνα 13 φαίνεται το ανώτερο επίπεδο ενός από τα 4 τοπικά τμήματα του δικτύου τροφοδοσίας, το οποίο συνδέεται με το αντίστοιχο τμήμα του καθολικού, μέσω των 6 γκρι κόμβων.



Εικόνα 13 -Το ανώτερο επίπεδο ενός τοπικού τμήματος του δικτύου τροφοδοσίας

Τα υπόλοιπα (κατώτερα) επίπεδα κάθε τοπικού τμήματος, περιέχουν μόνο αντιστάσεις που συνδέονται σειριακά, ενώ στους κόμβους που βρίσκονται μεταξύ των αντιστάσεων αυτών, συνδέονται οι ακροδέκτες τροφοδοσίας των πυλών του ψηφιακού κυκλώματος.

3.4 Προσομοίωση με FineSim SPICE

Για την ανάλυση μεταβατικής κατάστασης λειτουργίας των τμημάτων του ψηφιακού κυκλώματος καθώς και του δικτύου τροφοδοσίας, χρησιμοποιήθηκε το εργαλείο FineSim SPICE της Magma. Η επιστροφή των αποτελεσμάτων γίνεται μέσω αρχείων κειμένου (κωδικοποίηση ASCII), ενώ η εκτύπωση των κυματομορφών γίνεται μέσω του μαθηματικού εργαλείου GNU Octave.

Για την δημιουργία των αρχείων εισόδου που οδηγούν τις προσομοιώσεις του FineSim SPICE, αναπτύχθηκε μια σειρά από προγράμματα επεξεργασίας κειμένου (parsers) σε

γλώσσα C. Τα περιεχόμενα των αρχείων-οδηγών που δημιουργούν οι parsers, είναι τα παρακάτω:

1. Περιγραφή του εκάστοτε κυκλώματος προς προσομοίωση:
 - a. Διάταξη πυλών σε μορφή λίστας στοιχείων (netlist), συμβατή με SPICE.
 - b. Μοντέλα τρανζίστορ (π.χ. ορισμός παραμέτρων τεχνολογίας).
 - c. Μοντέλα πυλών (π.χ. σύνθεση πυλών με χρήση τρανζίστορ).
2. Σε περίπτωση προσομοίωσης ψηφιακού κυκλώματος, ανάθεση τιμών στις λογικές εισόδους του κυκλώματος, με χρήση πηγών τάσεων όπως περιγράφεται στην παράγραφο 3.2.
3. Συνδέσεις με την πηγή τροφοδοσίας και τη γείωση.
4. Ορισμός και συνδέσεις πηγών τάσεων ή/και ρεύματος τύπου PWL, οι τιμές των οποίων προέρχονται από τα αποτελέσματα προηγούμενων βημάτων ή επαναλήψεων της ιεραρχικής μεθόδου.
5. Ορισμός των κόμβων (ή συσκευών) των οποίων την τάση (ή ρεύμα) θα υπολογίσει η προσομοίωση.
6. Περιγραφή του τύπου της ανάλυσης, της διάρκειας και του ελάχιστου βήματος.

Οι parsers αυτοί είναι υπεύθυνοι και για την επεξεργασία των αποτελεσμάτων που παράγει κάθε προσομοίωση και για τον συνδυασμό τους (π.χ. με χρήση του περιοριστικού παράγοντα), πριν την δημιουργία των απαραίτητων πηγών τάσης ή ρεύματος τύπου PWL. Για παράδειγμα, για τον υπολογισμό της τάσης που εφαρμόζεται στον κόμβο με όνομα mod1_i278, ο οποίος τροφοδοτεί μια πύλη τύπου NAND του κυκλώματος C1355, κατά το 1^ο βήμα της 6^{ης} επανάληψης της ιεραρχικής μεθόδου, και για περιοριστικό παράγοντα ίσο με 0.6, ο parser συνδυάζει τα αποτελέσματα του 4^{ου} βήματος της 5^{ης} επανάληψης (Εικόνα 14 - αριστερά) και τα αποτελέσματα που υπολογίστηκαν και χρησιμοποιήθηκαν κατά το 1^ο βήμα της 5^{ης} επανάληψης (Εικόνα 14 - δεξιά), και δημιουργεί μια νέα πηγή τάσης τύπου PWL (Εικόνα 14 - κάτω).

time v(mod1_i278)	time v(mod1_i278)
0.000000e+00 1.099826e+00	0.000000e+00 1.099800e+00
1.000000e-12 1.099826e+00	1.000000e-12 1.099800e+00
2.000000e-12 1.099826e+00	2.000000e-12 1.099800e+00
...	...
vMod1_I278 Mod1_I278 0 pwl(0.00e+00 1.099816e+00 1.00e-12 1.099816e+00 2.00e-12 1.099816e+00 ...	

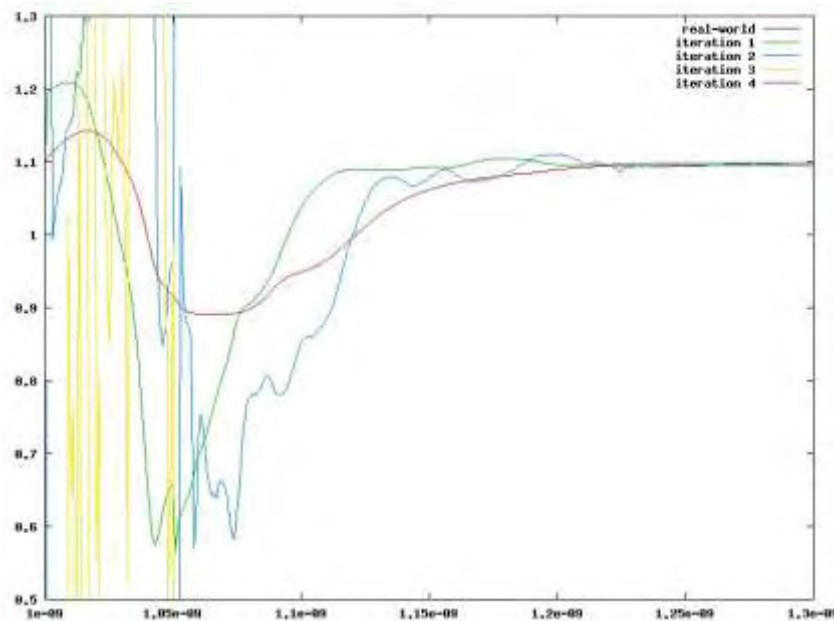
Εικόνα 14 - Παράδειγμα μετατροπής αποτελεσμάτων σε πηγές τάσης τύπου PWL

Η εκτέλεση των parsers και των προσομοιώσεων σε FineSim SPICE γίνονται εναλλάξ. Την οργάνωση των εκτελέσεων αυτών αναλαμβάνουν προγράμματα αυτοματοποίησης διαδικασιών (bash-shell scripts) που αναπτύχθηκαν στα πλαίσια της υλοποίησης της ιεραρχικής μεθόδου. Τα scripts αυτά αναλαμβάνουν και την παράλληλη εκτέλεση των προσομοιώσεων, ως παράλληλες διεργασίες, όταν αυτό επιτρέπεται από την ιεραρχική μέθοδο, δηλαδή κατά τα βήματα 1, 2 και 4 κάθε επανάληψης, καθώς και τον συγχρονισμό που απαιτείται για την ορθή εκτέλεση του βήματος 3. Όλες οι δοκιμές της παρούσης υλοποίησης έγιναν με ενεργοποιημένη την λειτουργία της παράλληλης εκτέλεσης.

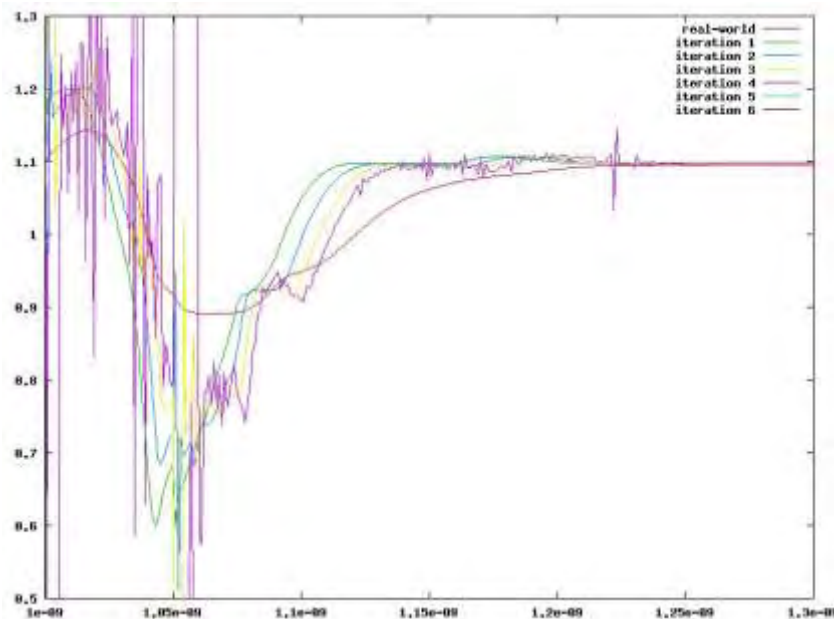
4 Πειραματικά αποτελέσματα υλοποίησης

4.1 Παρουσίαση αποτελεσμάτων

Για την επίτευξη της σύγκλισης της υλοποίησης της ιεραρχικής μεθόδου, έγινε χρήση του περιοριστικού παράγοντα στις τάσεις καθώς και εξομάλυνση των κυματομορφών τάσης, όπως περιγράφεται στις παραγράφους 2.5.1 και 2.5.2. Στις παρακάτω εικόνες φαίνονται οι κυματομορφές τάσης που επιστρέφει ως αποτέλεσμα η υλοποίηση για έναν τυχαίο κόμβο, σε σύγκριση πάντα με την κυματομορφή της πραγματικής τάσης του ίδιου κόμβου. Η Εικόνα 15 δείχνει τις τάσεις πριν την εφαρμογή του περιοριστικού παράγοντα, ενώ η Εικόνα 16 δείχνει τις τάσεις μετά την εφαρμογή του περιοριστικού παράγοντα.



Εικόνα 15 - Αποτελέσματα τάσεων πριν την εφαρμογή περιοριστικού παράγοντα



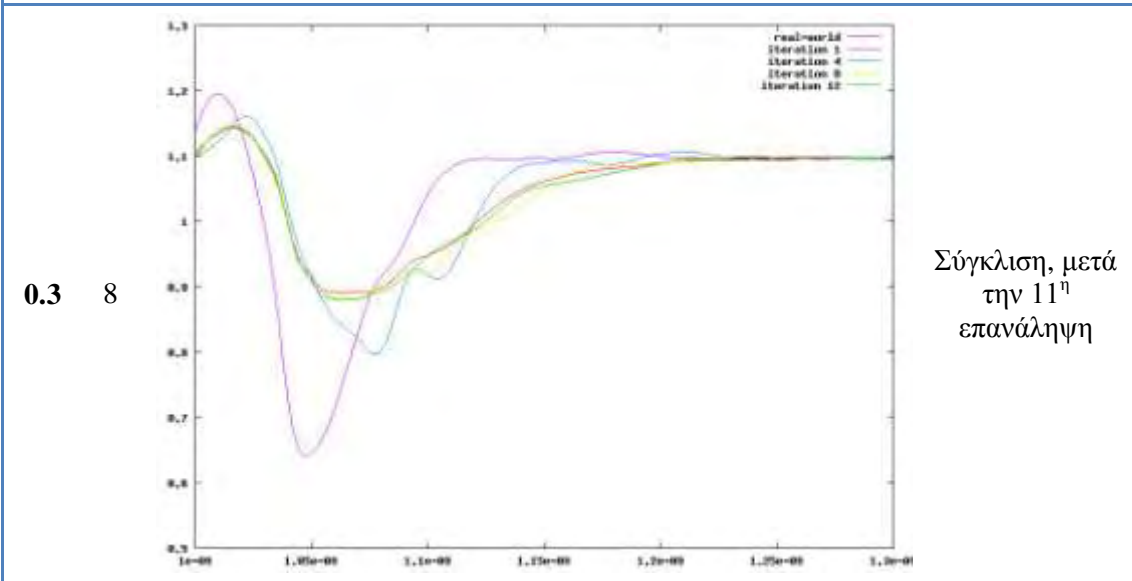
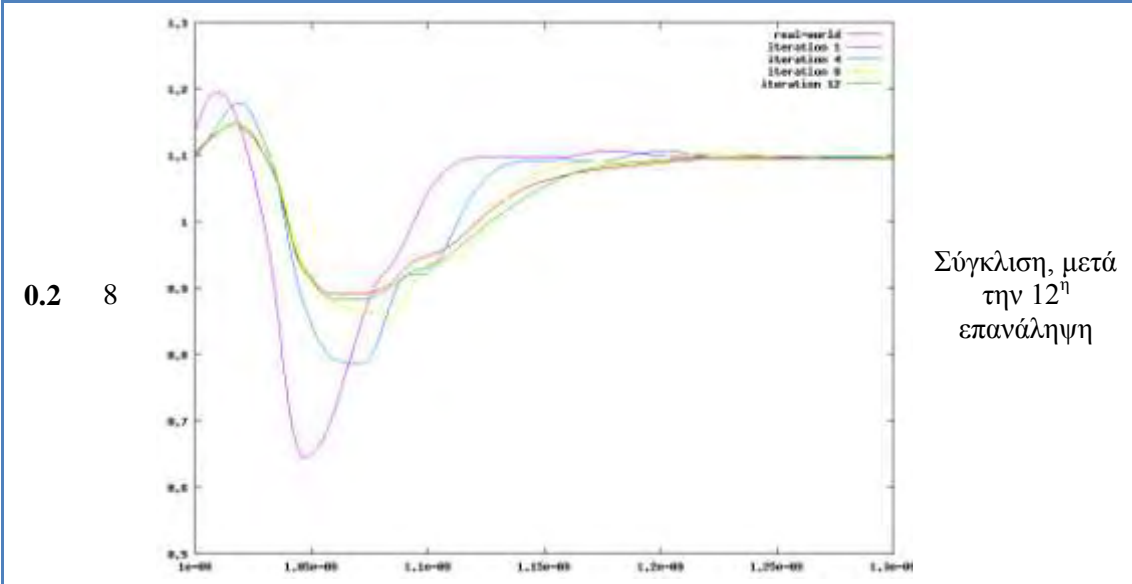
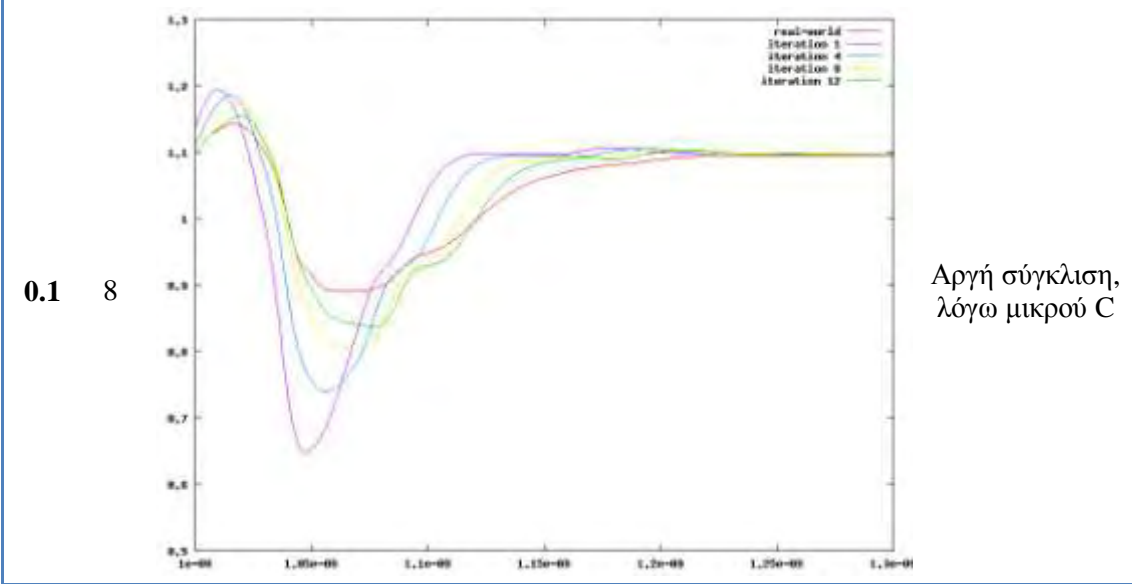
Εικόνα 16 - Αποτελέσματα τάσεων μετά την εφαρμογή περιοριστικό παράγοντα

Είναι εμφανές ότι χωρίς την χρήση του περιοριστικού παράγοντα, η σύγκλιση δεν είναι δυνατόν να επιτευχθεί, ενώ με χρήση του περιοριστικού παράγοντα η σύγκλιση επιτυγχάνεται μερικώς, λόγω των τοπικών ανωμαλιών που υπάρχουν στην κυματομορφή. Για την επίτευξη της τέλει σύγκλισης, είναι απαραίτητη και η εξομάλυνση των κυματομορφών εξόδου.

Για την εξομάλυνση κυματομορφών υλοποιήθηκε η μέθοδος του μετακινούμενου μέσου όρου με γραμμικά βάρη. Έτσι οι δύο μοναδικοί συντελεστές που θα πρέπει να οριστούν από τον χρήστη πριν την εκτέλεση της ιεραρχικής μεθόδου είναι ο περιοριστικός παράγοντας (έστω C) και το πλήθος των σημείων που συμβάλουν στον υπολογισμό του μέσου όρου κατά την εξομάλυνση κυματομορφής (έστω K).

Στον παρακάτω πίνακα φαίνονται κάποιες ενδεικτικές εκτελέσεις της υλοποίησης, για διάφορους συνδυασμούς των C και K .

C Κ Εικόνα Κυματομορφών Τάσης Σύγκλιση



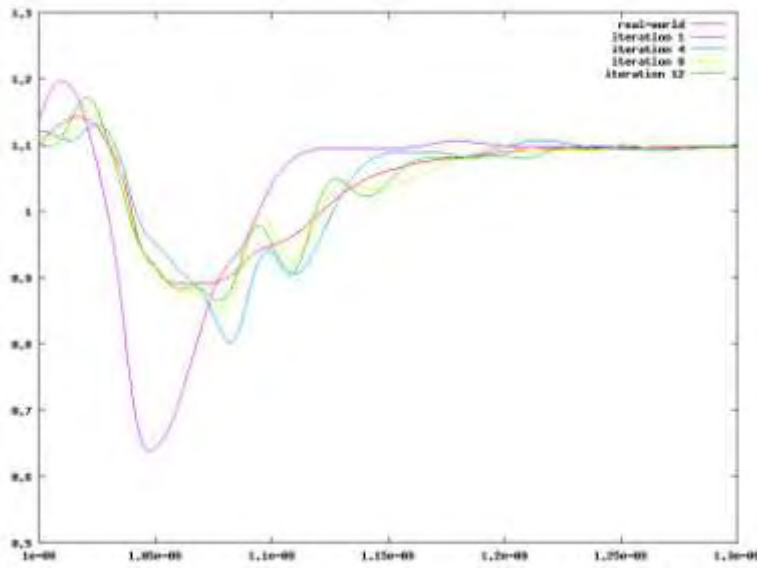
C

K

Εικόνα Κυματομορφών Τάσης

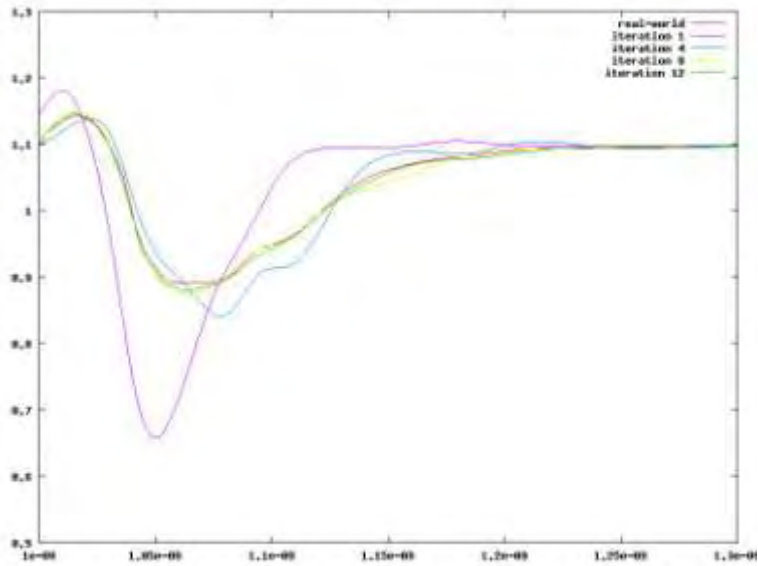
Σύγκλιση

0.4 8



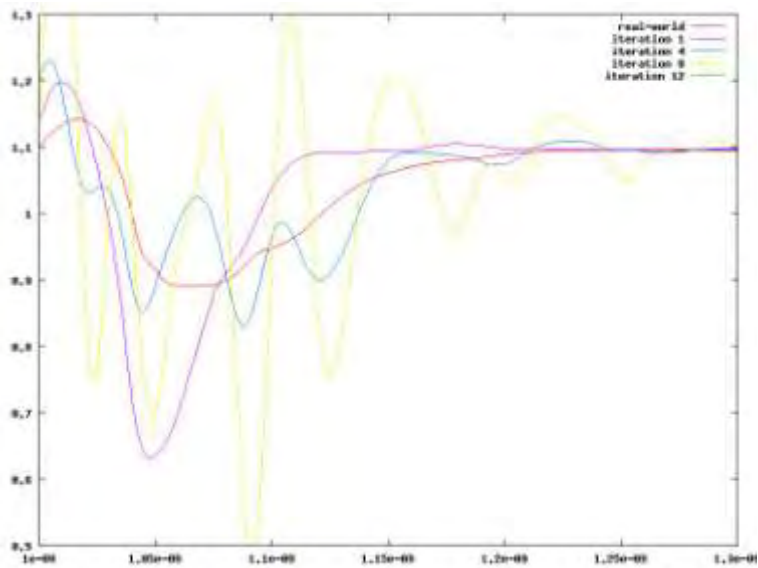
Μερική σύγκλιση, μικρή απόκλιση λόγω ταλαντώσεων

0.4 14

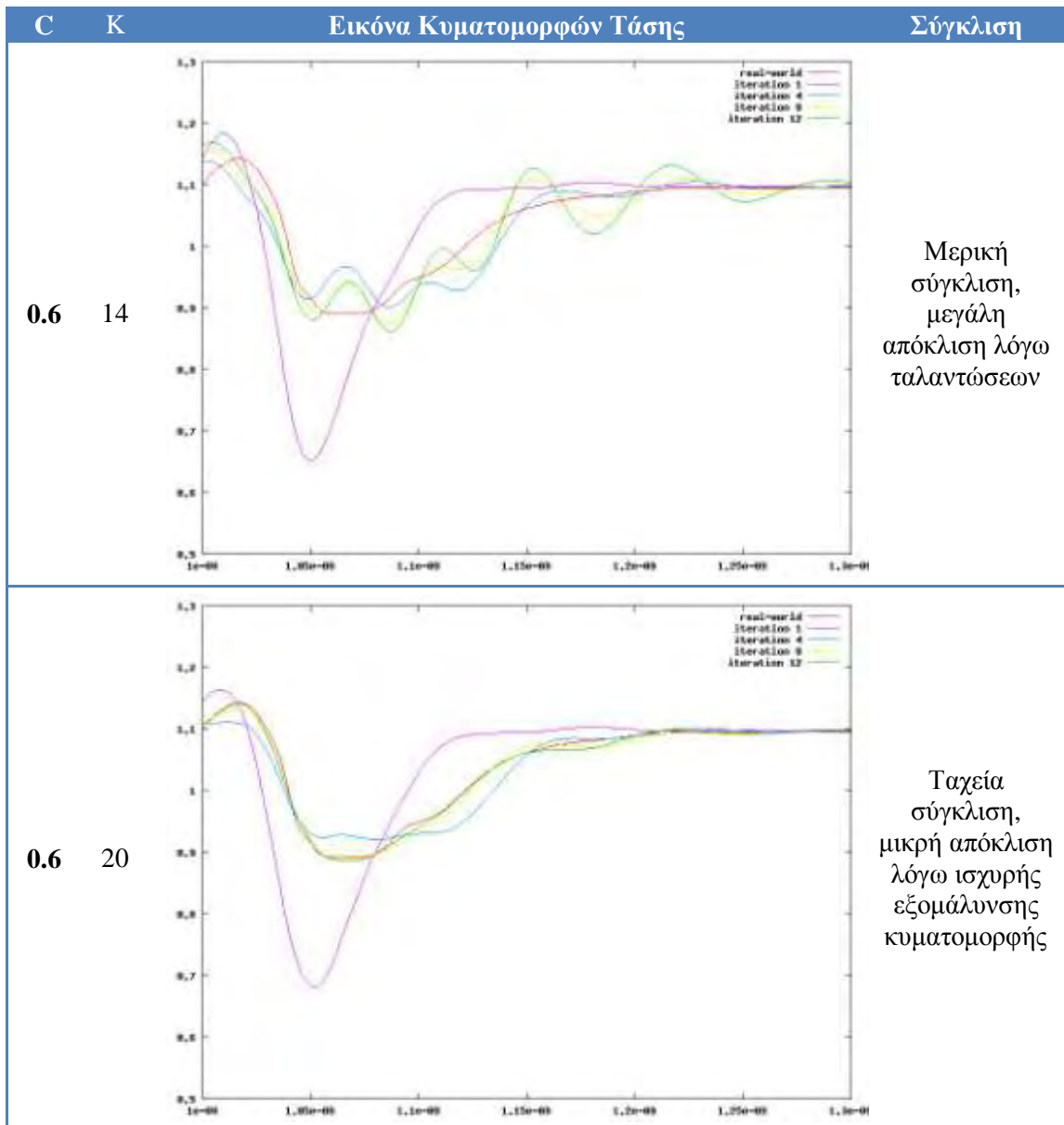


Σύγκλιση, μετά την 9^η επανάληψη

0.6 8



Όχι, λόγω υψηλών ταλαντώσεων



4.2 Αξιολόγηση αποτελεσμάτων

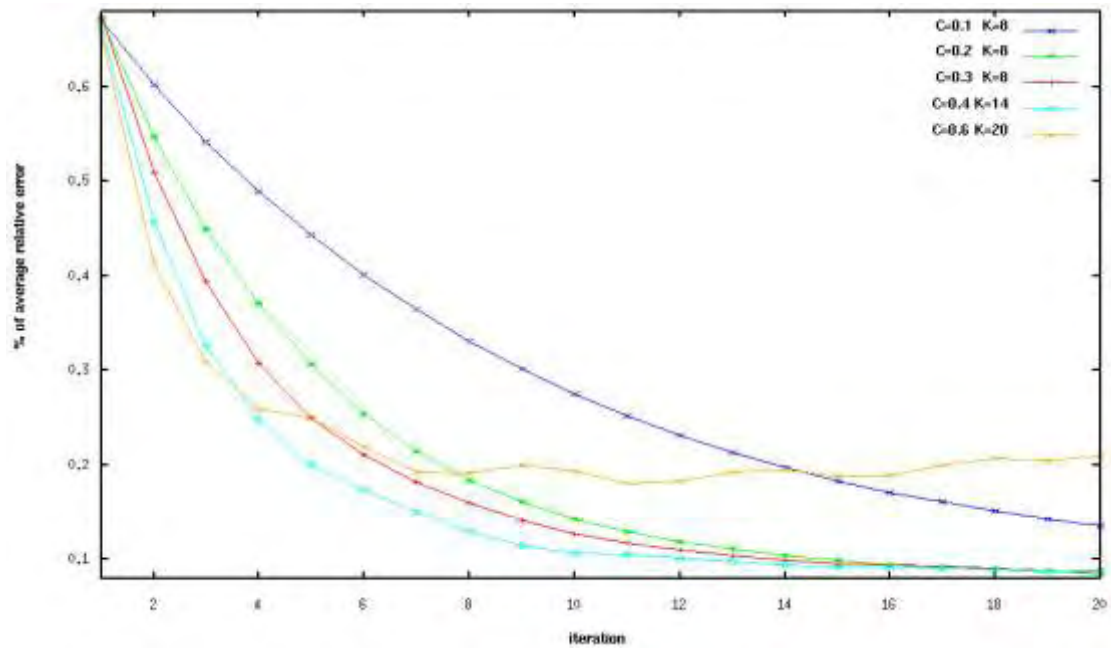
Με χρήση κατάλληλων τιμών για τα C και K, η ιεραρχική μέθοδος μπορεί να συγκλίνει στα πραγματικά αποτελέσματα μέσα σε 8-12 επαναλήψεις. Αύξηση του περιοριστικού παράγοντα C, απαιτεί και την ανάλογη αύξηση της ισχύος της μεθόδου εξομάλυνσης κυματομορφής, διαφορετικά εμφανίζονται ταλαντώσεις στις κυματομορφές εξόδου.

Για τις περιπτώσεις ικανοποιητικά καλής σύγκλισης, έχει υπολογιστεί το μέσο και το μέγιστο σχετικό σφάλμα, σε σχέση με τα πραγματικά αποτελέσματα, σύμφωνα με την σχέση:

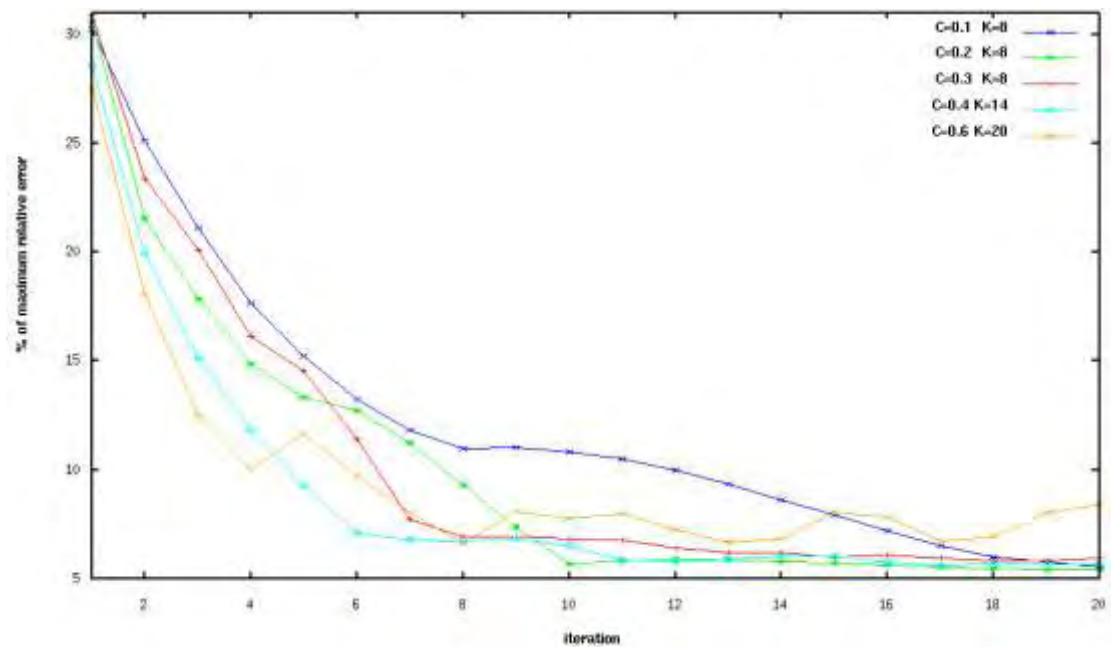
$$E_{relative} = \frac{|V_{real} - V_{iter}|}{V_{real}}$$

Στον υπολογισμό των σφαλμάτων συνέβαλαν οι τάσεις εξόδου, όλων των κόμβων που συνδέουν το ψηφιακό κύκλωμα με το δίκτυο τροφοδοσίας ($4 * 522 = 2088$ κόμβοι).

Το μέσο σχετικό σφάλμα και το μέγιστο σχετικό σφάλμα, φαίνονται στην Εικόνα 17 και στην Εικόνα 18 αντίστοιχα. Το μέσο σφάλμα είναι πάντα μικρότερο του 1% της τάσης τροφοδοσίας (λιγότερο από 5mV). Το μέγιστο σφάλμα, είναι της τάξης του 5-15% μετά από την 8^η επανάληψη και ανάλογα με τις τιμές των C και K που χρησιμοποιούνται.



Εικόνα 17 - Μέσο σχετικό σφάλμα ιεραρχικής μεθόδου



Εικόνα 18 - Μέγιστο σχετικό σφάλμα ιεραρχικής μεθόδου

Οι τιμές 0.4 και 14 για τα C και K αντίστοιχα, είναι μια καλή επιλογή, καθώς στην περίπτωση αυτή η ιεραρχική μέθοδος παρουσιάζει μια σταθερότητα στο ρυθμό που συγκλίνει όπως φαίνεται στην εικόνα του μέσου σχετικού σφάλματος. Επίσης το μέγιστο σχετικό σφάλμα είναι μικρότερο του 5% της τάσης τροφοδοσίας από την 8^η επανάληψη και μετά.

Για $C=0.6$ και $K=20$, η ιεραρχική μέθοδος παρουσιάζει αστάθειες, καθώς διαδοχικές επαναλήψεις μπορεί να δώσουν χειρότερα αποτελέσματα όπως φαίνεται στην εικόνα με το μέσο σχετικό σφάλμα, μεταξύ των επαναλήψεων 8 και 9. Συνεπώς, τιμές για το C μεγαλύτερες ή ίσες του 0.6, είναι πιθανόν να επηρεάσουν αρνητικά την σύγκλιση της ιεραρχικής μεθόδου.

Από την οπτική γωνία του συνολικού χρόνου εκτέλεσης, η υλοποίηση δεν παρέχει κάποια πληροφορία σχετικά για το αν και κατά πόσο ταχύτερη είναι η ιεραρχική μέθοδος, σε σχέση με την επίπεδη προσομοίωση (κατά την οποία το ψηφιακό κύκλωμα και το δίκτυο τροφοδοσίας προσομοιώνονται μαζί). Λόγο της πολύ μικρής κλίμακας του κυκλώματος C1355 που χρησιμοποιεί η υλοποίηση, ο συνολικός χρόνος προσομοίωσης για την ιεραρχική μέθοδο είναι πολύ μεγαλύτερος από τον χρόνο της επίπεδης προσομοίωσης. Ενδεικτικά αναφέρεται ότι ο χρόνος επίπεδης προσομοίωσης της υλοποίησης για χρόνο 10 περιόδων είναι 11 λεπτά ενώ ο χρόνος προσομοίωσης με χρήση της ιεραρχικής μεθόδου είναι 134 λεπτά.

Οι δοκιμές που έγιναν στα πλαίσια της παρούσης εργασίας, επιβεβαιώνουν πως η ιεραρχική μέθοδος μπορεί να παράγει ακριβή αποτελέσματα. Καθώς με λοιπές μη-ιεραρχικές επαναληπτικές μεθόδους είναι δυνατή η προσομοίωση κυκλωμάτων με πλήθος στοιχείων της τάξης των μερικών εκατομμυρίων, είναι λογικό να υποθέσουμε πως με την ιεραρχική μέθοδο το όριο μεγέθους του προς προσομοίωση ψηφιακού κυκλώματος θα αυξηθεί. Για παράδειγμα αν ένα υπολογιστικό σύστημα είναι ικανό να προσομοιώσει κυκλώματα μεγέθους το πολύ X στοιχείων με χρήση της μη-ιεραρχικής μεθόδου που περιγράφεται στην παράγραφο 1.6.3, τότε θα μπορεί να προσομοιώσει κύκλωμα μεγέθους το πολύ $N \cdot X$ στοιχείων με χρήση της ιεραρχικής μεθόδου, εάν το κύκλωμα παρέχεται σε N τμήματα των X στοιχείων το καθένα. Θεωρητικά λοιπόν, λαμβάνοντας υπόψη ότι το πλήθος των ανεξάρτητων τμημάτων που συντελούν ένα πραγματικό ολοκληρωμένο κύκλωμα ανέρχεται συνήθως σε μερικές δεκάδες και πως τα σημερινά εργαλεία προσομοίωσης μπορούν να υποστηρίξουν δυναμική IR drop ανάλυση σε κυκλώματα το πολύ 5-10 εκατομμυρίων στοιχείων, είναι λογικό να υποθέσουμε πως με την ιεραρχική μέθοδο είναι δυνατή η προσομοίωση κυκλωμάτων με πλήθος στοιχείων της τάξης των 100 εκατομμυρίων.

Παράρτημα

Κώδικας σε μορφή bash-shell script που υλοποιεί την ιεραρχική μέθοδο (παράλληλα):

```
#!/bin/bash

#define all the needed variables
CC=gcc

#location of this script
SCRIPT_PATH="$( cd "$( dirname "${BASH_SOURCE[0]}" )" && pwd )"
SRC=$SCRIPT_PATH/src
BIN=$SCRIPT_PATH/bin

#location of some GBs of free disk space
STORE_PATH=/data/nanotropic/iosteve
DRIVERS=$STORE_PATH/drivers
RESULTS=$STORE_PATH/results
MAX_BLOCKS=4

VDD=1.1
TRAN_TIME_STEP=1p
TRAN_TIME_MAX=1500p

RESTRAINT_FACTOR=0.3
SMOOTHING_POINTS_STEP1=1
SMOOTHING_POINTS_STEP2=8
SMOOTHING_POINTS_STEP3=8
SMOOTHING_POINTS_STEP4=8
SMOOTHING_TYPE=WMA

#location of the files containing the cells
CELLS_CONNECTED_PATH=$SCRIPT_PATH/cells/connected
CELLS_UNCONNECTED_PATH=$SCRIPT_PATH/cells/unconnected
CELLS_INPUT=$SCRIPT_PATH/cells/cell_inputs.vec

#location of the files containing the power grid
GRID_MODULES_PATH=/users/iosteve/C1355/...

#location of the transistor and gate models
MODELS_PATH=models.cir
FINESIM_OPTIONS_LIST=-spice
FINESIM_OPTIONS_FILE=$SCRIPT_PATH/finesim_options.sp
FINESIM_PATH=/cad2/magma/finesim/bin/finesim

#create the required folders
echo -ne "Initializing:"
echo -ne "Removing old files..."
rm -rf $CELLS_CONNECTED_PATH
rm -rf $BIN
rm -rf $DRIVERS
rm -rf $RESULTS
mkdir $CELLS_CONNECTED_PATH
mkdir $BIN
mkdir $DRIVERS
mkdir $RESULTS
echo -ne "Old files removed"
```

```

#connect the cell blocks to the grid blocks
echo -ne "Compiling..."
$CC $SRC/connect_cells.c -o $BIN/connect_cells
$CC $SRC/result_smoothing.c $SRC/parsers.c -o $BIN/result_smoothing

for num in 1 2 3 4 ; do
#compile the create_driver source code
$CC -Wall -O4 $SRC/create_driver_step$num.c $SRC/parsers.c -o
    $BIN/create_driver_step$num &> /dev/null
done
echo -ne "Compilation completed"

for file_path in $CELLS_UNCONNECTED_PATH/*.sp ; do
#parse the X from "file_name_X.sp"
file=$(basename $file_path)
base="${file%.*}"
num=${base#${base%?}}

#run the connect_cells executable
echo -ne "Creating driver..."
$BIN/connect_cells $file_path
    $CELLS_CONNECTED_PATH/cells_connected_$num.sp
    $GRID_MODULES_PATH/Module$num/Mod$num\_Block_VDD.info $num
echo -ne "Created file:"
    $CELLS_CONNECTED_PATH/cells_connected_$num.sp"
done

iter=1

while true ; do
prev_iter=$(expr $iter - 1)

#create the driver for step 1 (simulation of current of cell
    block using VDD supply) and simulate
echo -ne "Step 1:"
for num in `seq 1 $MAX_BLOCKS` ; do
if [ $iter == 1 ]; then
    cell_voltage=-
    cell_voltage_prev=-
    cell_voltage_next=-
else
    cell_voltage=$RESULTS/driver_iter$prev_iter\_step4_mod$num\_smoothed.txt
    cell_voltage_prev=$RESULTS/cell_voltages_iter$prev_iter\_mod$num.txt
    cell_voltage_next=$RESULTS/cell_voltages_iter$iter\_mod$num.txt
fi

#run the create_driver_step1 executable
echo -ne "Creating driver..."
$BIN/create_driver_step1 $CELLS_CONNECTED_PATH/cells_connected_$num.sp
    $DRIVERS/driver_iter$iter\_step1_mod$num.sp
    $GRID_MODULES_PATH/Module$num/Mod$num\_Block_VDD.info $num
    $cell_voltage_prev $cell_voltage $cell_voltage_next $RESTRAINT_FACTOR
    $FINESIM_OPTIONS_FILE $CELLS_INPUT $VDD $TRAN_TIME_STEP $TRAN_TIME_MAX
    $MODELS_PATH
echo -ne "Created file: driver_iter${iter}_step1_mod$num.sp"
done

```

```

echo -ne "Simulating cell blocks..."

for num in `seq 1 $MAX_BLOCKS` ; do
    #run the simulation
    cd $RESULTS
    $FINESIM_PATH $FINESIM_OPTIONS_LIST
        $DRIVERS/driver_iter$iter\_step1_mod$num.sp &> /dev/null &
done
wait
echo -ne "\b\b\b  "

for num in `seq 1 $MAX_BLOCKS` ; do
    #run the result_smoothing executable
    echo -ne "Smoothing results..."
    $BIN/result_smoothing $RESULTS/driver_iter$iter\_step1_mod$num.pt0
        $RESULTS/driver_iter$iter\_step1_mod$num\_smoothed.txt $STRAN_TIME_STEP
        $STRAN_TIME_MAX $SMOOTHING_TYPE $SMOOTHING_POINTS_STEP1 "i"
    echo -ne "Created file:
        driver_iter${iter}_step1_mod${num}_smoothed.txt"
done

#create the driver for step 2 (simulation of current of grid-blocks using VDD
supply and cell currents computed in step 1) and simulate
echo -ne "Step 2:"
for num in `seq 1 $MAX_BLOCKS` ; do
    if [ $iter == 1 ]; then
        block_voltage=-
        block_voltage_prev=-
        block_voltage_next=-
    else
        block_voltage=$RESULTS/driver_iter$prev_iter\_step3_smoothed.txt
        block_voltage_prev=$RESULTS/block_voltages_iter
            $prev_iter\_mod$num\_step2.txt
        block_voltage_next=$RESULTS/block_voltages_iter$iter\_mod$num\_step2.txt
        #this will calculate the prevIter_step4.txt again
    fi

    #run the create_driver_step2 executable
    echo -ne "Creating driver..."
    $BIN/create_driver_step2 $GRID_MODULES_PATH/Module$num/Mod$num\_std_VDD.cir
        $GRID_MODULES_PATH/Module$num/Mod$num\_glob_VDD_M1M6.cir
        $DRIVERS/driver_iter$iter\_step2_mod$num.sp $num
        $RESULTS/driver_iter$iter\_step1_mod$num\_smoothed.txt
        $block_voltage_prev $block_voltage $block_voltage_next $RESTRAINT_FACTOR
        $FINESIM_OPTIONS_FILE $VDD $STRAN_TIME_STEP $STRAN_TIME_MAX $MODELS_PATH
    echo -ne "Created file: driver_iter${iter}_step2_mod$num.sp"
done

echo -ne "Simulating grid blocks..."

for num in `seq 1 $MAX_BLOCKS` ; do
    #run the simulation
    cd $RESULTS
    $FINESIM_PATH $FINESIM_OPTIONS_LIST
        $DRIVERS/driver_iter$iter\_step2_mod$num.sp &> /dev/null &
done
wait

```

```

for num in `seq 1 $MAX_BLOCKS` ; do
    #run the result_smoothing executable
    echo -ne "Smoothing results..."
    $BIN/result_smoothing $RESULTS/driver_iter$iter\_step2_mod$num.pt0
        $RESULTS/driver_iter$iter\_step2_mod$num\_smoothed.txt $STRAN_TIME_STEP
        $STRAN_TIME_MAX $SMOOTHING_TYPE $SMOOTHING_POINTS_STEP2 "i"
    echo -ne "Created file: driver_iter${iter}_step2_mod${num}_smoothed.txt"
done

#create the driver for step 3 (simulation of voltage of hierarchical grid using
    VDD supply and block currents computed in step 2) and simulate

echo -ne "Step 3:"
#create a list of the results of step 2 for every module
MODULES_RESULTS_LIST=""
for num in `seq 1 $MAX_BLOCKS` ; do
    MODULES_RESULTS_LIST="$MODULES_RESULTS_LIST
        $RESULTS/driver_iter${iter}_step2_mod${num}_smoothed.txt"
done

#run the create_driver_step3 executable
echo -ne "Creating driver..."
$BIN/create_driver_step3
    $GRID_MODULES_PATH/Hierarchical/Hierarchicalglob_VDD_M7M8.cir
    $DRIVERS/driver_iter$iter\_step3.sp $MAX_BLOCKS $MODULES_RESULTS_LIST
    $GRID_MODULES_PATH/Hierarchical/HierarchicalVia_VDD_M6M7.info
    $FINESIM_OPTIONS_FILE $VDD $STRAN_TIME_STEP $STRAN_TIME_MAX $MODELS_PATH
echo -ne "Created file: driver_iter${iter}_step3.sp"

#run the simulation
cd $RESULTS
echo -ne "Simulating hierarchical grid..."
$FINESIM_PATH $FINESIM_OPTIONS_LIST $DRIVERS/driver_iter$iter\_step3.sp &>
    /dev/null

#run the result_smoothing executable
echo -ne "Smoothing results..."
$BIN/result_smoothing $RESULTS/driver_iter$iter\_step3.pt0
    $RESULTS/driver_iter$iter\_step3_smoothed.txt $STRAN_TIME_STEP $STRAN_TIME_MAX
    $SMOOTHING_TYPE $SMOOTHING_POINTS_STEP3 "v"
echo -ne "Created file: driver_iter${iter}_step3_smoothed.txt"

#create the driver for step 4 (simulation of voltage of grid-blocks using
    voltages computed in step 3 and cell currents computed in step 1) and
    simulate
echo -ne "Step 4:"
for num in `seq 1 $MAX_BLOCKS` ; do

    if [ $iter == 1 ]; then
        block_voltage=$RESULTS/driver_iter$iter\_step3_smoothed.txt
        block_voltage_prev=-
        block_voltage_next=$RESULTS/block_voltages_iter$iter\_mod$num\_step4.txt
    else
        block_voltage=$RESULTS/driver_iter$iter\_step3_smoothed.txt
        block_voltage_prev=$RESULTS/block_voltages_iter$iter\_mod$num\_step2.txt
        block_voltage_next=$RESULTS/block_voltages_iter$iter\_mod$num\_step4.txt
    fi

```



```

#run the create_driver_step4 executable
echo -ne "Creating driver..."
$BIN/create_driver_step4 $GRID_MODULES_PATH/Module$num/Mod$num\_std_VDD.cir
    $GRID_MODULES_PATH/Module$num/Mod$num\_glob_VDD_M1M6.cir
    $DRIVERS/driver_iter$iter\_step4_mod$num.sp $num
    $RESULTS/driver_iter$iter\_step1_mod$num\_smoothed.txt
    $block_voltage_prev $block_voltage $block_voltage_next
    $GRID_MODULES_PATH/Module$num/Mod$num\_Block_VDD.info $RESTRAINT_FACTOR
    $FINESIM_OPTIONS_FILE $VDD $STRAN_TIME_STEP $STRAN_TIME_MAX $MODELS_PATH
echo -ne "Created file: driver_iter${iter}_step4_mod$num.sp"
done

echo -ne "Simulating grid blocks..."
for num in `seq 1 $MAX_BLOCKS` ; do

    #run the simulation
    cd $RESULTS
    $FINESIM_PATH $FINESIM_OPTIONS_LIST
    $DRIVERS/driver_iter$iter\_step4_mod$num.sp &> /dev/null &

done
wait

for num in `seq 1 $MAX_BLOCKS` ; do
    #run the result_smoothing executable
    echo -ne "Smoothing results..."
    $BIN/result_smoothing $RESULTS/driver_iter$iter\_step4_mod$num.pt0
    $RESULTS/driver_iter$iter\_step4_mod$num\_smoothed.txt $STRAN_TIME_STEP
    $STRAN_TIME_MAX $SMOOTHING_TYPE $SMOOTHING_POINTS_STEP4 "v"
    echo -ne "Created file: driver_iter${iter}_step4_mod${num}_smoothed.txt"
done

iter=$(expr $iter + 1)

echo -ne "Continue to iteration $iter (Y/N)?"
answer=y
read answer
if test "$answer" != "Y" -a "$answer" != "y"; then
    break
fi

done

#remove everything created by this script, except from results
rm -rf $CELLS_CONNECTED_PATH
rm -rf $BIN
rm -rf $DRIVERS

echo -ne "-----DONE-----"

```

Κώδικας σε γλώσσα C, για την μέθοδο μετακινούμενου μέσου όρου με γραμμικά βάρη, η οποία χρησιμοποιείται για εξομάλυνση των κυματομορφών τάσης:

```
void weighted_moving_average(double **matrix, int x, int y, int points){

    int i, j, k;
    double **tmp=NULL;
    double sum, weight, sum_weight;

    tmp = (double**)malloc(points*sizeof(double*));
    NULL_CHECK(tmp);
    for(i=0; i<points; i++){
        tmp[i] = (double*)malloc(x*sizeof(double));
        NULL_CHECK(tmp[i]);
    }

    for(i=0; i<y; i++){
        for(j=0; j<x; j++){

            // add the middle point
            weight = points+1;
            sum = matrix[i][j+1] * weight;
            sum_weight = weight;

            // add the previous points
            for(k=MY_MAX(0, points-i); k<points; k++){
                weight = k+1;
                sum += tmp[k][j] * weight;
                sum_weight += weight;
            }

            // add the following points
            for(k=i+1; k<MY_MIN(y, i+points); k++){
                weight = points-(k-i-1);
                sum += matrix[k][j+1] * weight;
                sum_weight += weight;
            }

            for(k=MY_MAX(0, points-i-1); k<points-1; k++) tmp[k][j] = tmp[k+1][j];
            tmp[points-1][j] = matrix[i][j+1];

            sum = sum / sum_weight;
            matrix[i][j+1] = sum;
        }
    }

    for(i=0; i<points; i++) free(tmp[i]);
    free(tmp);
}
```

Parser σε γλώσσα C, που δημιουργεί τα αρχεία που οδηγούν τις προσομοιώσεις του FineSim SPICE, για το 1^ο βήμα της ιεραρχικής μεθόδου (με αντίστοιχο τρόπο λειτουργούν και οι parsers για τα υπόλοιπα βήματα της ιεραρχικής μεθόδου):

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include "parsers.h"

int main(int argc, char *argv[]){
    FILE *fout, *finfo;
    char line[1024], tmp_line[MAX_STRING_LENGTH];
    int i, j;
    int new_exists, prev_exists; //indicate if the new and prev files exist
    char **nodes=NULL;
    int nodes_count;
    double **voltages_prev=NULL;
    double **voltages_new=NULL;
    double **voltages_next=NULL;
    int timepoints;
    double c=0.0;
    double Vdd=0.0;
    int time_step;
    int time_max;

    if(argc < 14){
        printf("Please specify: <cells.sp> <driver.sp> <cells.info> <mod_no>
            <cell_voltage_prev> <cell_voltage_new> <store_cell_voltage>
            <restraint_factor> <finesim_options_file.sp> <cell_inputs.vec> <VDD>
            <time_step> <time_max> [<transistor_model_files>]*\n");
        return 0;
    }

    fout = fopen(argv[2], "w");
    if(fout==NULL){
        printf("Error opening file %s\n", argv[2]);
        return 0;
    }

    finfo = fopen(argv[3], "r");
    if(finfo==NULL){
        printf("Error opening file %s\n", argv[3]);
        return 0;
    }

    // set the restraint factor
    sscanf(argv[8], "%lf", &c);

    // set the Vdd
    sscanf(argv[11], "%lf", &Vdd);

    // set the time_step and time_max
    sscanf(argv[12], "%dp", &time_step);
    sscanf(argv[13], "%dp", &time_max);

    fprintf(fout, "* Step 1: Simulation of cell-blocks *\n");
    fprintf(fout, "\n");

    fprintf(fout, "* include transistor models\n");
    for(i=14; i<argc; i++) fprintf(fout, ".inc '%s'\n", argv[i]);
    fprintf(fout, "\n");
```

```

fprintf(fout, "* include the cell-block\n");
fprintf(fout, ".inc '%s'\n", argv[1]);
fprintf(fout, "\n");

fprintf(fout, "* include input signals for the cells\n");
fprintf(fout, ".inc '%s'\n", argv[10]);
fprintf(fout, "\n");

fprintf(fout, "* include options file for finesim\n");
fprintf(fout, ".inc '%s'\n", argv[9]);
fprintf(fout, "\n");

fprintf(fout, "* take an instance of every subckt\n");
fprintf(fout, "xMod%s_cells_instance ", argv[4]);
fprintf(fout, "%s\n", Parse_cir_file(argv[1]));
fprintf(fout, "\n");

fprintf(fout, "* connect the VDD\n");
fprintf(fout, "v_my_dummy_VDD VDD 0 %lf\n", Vdd);
fprintf(fout, "\n");

fprintf(fout, "* connect the VSS\n");
fprintf(fout, "v_my_dummy_VSS VSS 0 0\n");
fprintf(fout, "\n");

// parse the txt file that contains the voltage of the nodes that connect the
// cells to the grid-block (Vprev)

prev_exists = Parse_pt0_file(argv[5], &nodes, &nodes_count, &voltages_prev,
    time_step, time_max);

if(!prev_exists){
    timepoints=0;
    nodes_count=0;
}
else{
    timepoints=(time_max/time_step)+1;
}

// parse the pt0 file that contains the voltage of the nodes that connect the
// cells to the grid-block (Vnew)

new_exists = Parse_pt0_file(argv[6], &nodes, &nodes_count, &voltages_new,
    time_step, time_max);

if(!new_exists){
    timepoints=0;
    nodes_count=0;
}
else{
    timepoints=(time_max/time_step)+1;
}

// calculate the next cell voltages (Vnext)
voltages_next = calculateVnext(prev_exists, new_exists, nodes_count, time_max,
    time_step, voltages_prev, voltages_new, c, Vdd);

// store Vnext to file
if(new_exists && argv[7][0]!='-') storeNext(argv[7], nodes, nodes_count,
    voltages_next, timepoints, 'v');

```

```

fprintf(fout, "* connect the cells to power supply\n");
fseek(finfo, 0, SEEK_SET);

while(fgets(line, sizeof(line), finfo)){
    if(line[0]=='.') continue;
    sscanf(line, "%*[^]]%*[]] %s", tmp_line);
    if(!new_exists){
        fprintf(fout, "v%s VDD xMod%s_cells_instance.%s 0\n", tmp_line, argv[4],
            tmp_line);
    }
    else{
        fprintf(fout, "v%s %s_dummy xMod%s_cells_instance.%s 0\n", tmp_line,
            tmp_line, argv[4], tmp_line);
        fprintf(fout, "v%s_dummy %s_dummy 0 pwl(", tmp_line, tmp_line);

        for(i=0; i<nodes_count; i++){
            if(!strcasecmp(nodes[i], tmp_line)){
                for(j=0; j<(time_max/time_step)+1; j++) fprintf(fout, "%e %e ",
                    voltages_next[j][0], voltages_next[j][i+1]);
                break;
            }
        }
        fprintf(fout, ")\n");
    }
}

if(prev_exists){
    for(i=0; i<timepoints; i++) free(voltages_prev[i]);
    free(voltages_prev);
}
if(new_exists){
    for(i=0; i<timepoints; i++) free(voltages_new[i]);
    free(voltages_new);
    for(i=0; i<timepoints; i++) free(voltages_next[i]);
    free(voltages_next);
}
if(prev_exists || new_exists){
    for(i=0; i<nodes_count; i++) free(nodes[i]);
    free(nodes);
}
fprintf(fout, "\n");

fprintf(fout, "* include probe and print statements\n");
fseek(finfo, 0, SEEK_SET);

while(fgets(line, sizeof(line), finfo)){
    if(line[0]=='.') continue;
    sscanf(line, "%*[^]]%*[]] %s", tmp_line);
    fprintf(fout, ".probe tran i(v%s)\n", tmp_line);
    fprintf(fout, ".print tran i(v%s)\n", tmp_line);
}

fprintf(fout, "\n");
fprintf(fout, ".tran %s %s\n", argv[12], argv[13]);
fprintf(fout, ".save\n");
fprintf(fout, ".end\n");

fclose(fout);
fclose(finfo);

return 0;
}

```



```

    tmp = str2num(word);
    if(tmp<from_time || tmp>to_time)
        continue;
    endif
    time(i) = tmp;
    j=1;
    while(j<=node_pos)
        [word, line] = strtok(line, " \t\n");
        j++;
    endwhile
    x(i) = str2num(word);
    i++;
    time = resize(time, 1, i);
    x = resize(x, 1, i);
endwhile
fclose(f);
return;
endfunction

#Print 20 plots, one for every iteration

clear;

node = "mod1_i2197";
time_start = 1.0e-9;
time_end = 1.3e-9;

figure(1, "visible", "off");

[t, x] = Parse_pt0_file("tmp_signal_flat.txt", node, time_start, time_end);
pl = plot(t, x, "r");
axis ([1.0e-9 1.3e-9 0.5 1.3]);
hold on;

i=1;
while (i<=20)
    [ti, xi] = Parse_pt0_file(strcat("tmp_signal_", int2str(i), ".txt"), node,
        time_start, time_end);

    pli = plot(ti, xi, "g");
    legend("real-world", strcat("iteration ", int2str(i)));
    hold on;
    print(strcat("iter", int2str(i), ".jpg"), "-djpg", "-s800,600");
    delete(pli);
    i++;
endwhile

exit(1);

```

Κώδικας σε γλώσσα C, για δημιουργία τυχαίων τιμών για τις λογικές εισόδους του ψηφιακού κυκλώματος C1355:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define VDD_SUPPLY          (1.1)
#define SWITCH_PROBABILITY (0.4) // as 40%
#define PERIOD              (1000) // in psec
#define TIME_MAX            (10000) // in psec

int main(int argc, char *argv[]){
    FILE *fin, *fout;
    char lin[1024], tmp[64];
    int i, j, k, p;
    float val, last_val;

    srand(time(NULL));

    if(argc < 2){
        printf("Please specify at least one file containing cells.\n");
        return 0;
    }

    fout = fopen("cell_inputs.vec", "w");
    if(fout==NULL){
        printf("Error oppening file \"cell_inputs.vec\".\n");
        return 0;
    }

    fprintf(fout, "* random input signals for the cells *\n");
    fprintf(fout, "* switching probability: %.1f  \n", SWITCH_PROBABILITY);
    fprintf(fout, "*                      period: %.1f ns\n", (float)PERIOD/1000.0);
    fprintf(fout, "*                      maximum time: %.1f ns\n\n", (float)TIME_MAX/1000.0);

    for(i=1; i<argc; i++){
        fin = fopen(argv[i], "r");
        if(fin==NULL){
            printf("Error oppening file \"%s\".\n", argv[i]);
            return 0;
        }

        if(fgets( lin, sizeof(lin), fin)){
            j=0;
            while(lin[j++]!=' '); // ignore the .subckt
            while(lin[j++]!=' '); // ignore the subckt's name

            k=0;
            while(lin[j]!='\n' && lin[j]!='\\'){
                if(lin[j]!=' '){
                    tmp[k] = lin[j];
                    k++;
                    j++;
                }
                else{
                    tmp[k]='\0';
                    fprintf(fout, "v%s %s 0 PWL(0 0.0)", tmp, tmp);
                    last_val = 0.0;
                    for(p=PERIOD; p<TIME_MAX; p+=PERIOD){
                        if((double)(rand()%100) < SWITCH_PROBABILITY*100.0){
```



```
        if(last_val == 0.0) val = VDD_SUPPLY;
        else val = 0.0;
        fprintf(fout, " %dp %.1f %dp %.1f", p, last_val, p+50,
                val);
        last_val = val;
    }
    fprintf(fout, "\n");
    k=0;
    j++;
}
}
}
fclose(fin);
}
fclose(fout);
}
```

Αρχείο σε μορφή λίστας στοιχείων (netlist), που οδηγεί την προσομοίωση του FineSim SPICE κατά το 1^ο βήμα της ιεραρχικής μεθόδου. Παράχθηκε με τον parser που περιέχεται στο παράρτημα:

```
* Step 1: Simulation of cell-blocks *

* include transistor models
.inc 'models.cir'
.inc 'NangateOpenCellLibrary.spi'

* include the cell-block
.inc '/users/iosteve/tests_C1355/cells/connected/cells_connected_1.sp'

* include input signals for the cells
.inc '/users/iosteve/tests_C1355/cells/cell_inputs.vec'

* include options file for finesim
.inc '/users/iosteve/tests_C1355/finesim_options.sp'

* take an instance of every subckt
xMod1_cells_instance  Mod1G1 Mod1G10 Mod1G11 Mod1G12 Mod1G13 Mod1G14 Mod1G15 Mod1G16
                      Mod1G17 Mod1G18 Mod1G19 Mod1G2 Mod1G20 Mod1G21 Mod1G22 Mod1G23
                      Mod1G24 Mod1G25 Mod1G26 Mod1G27 Mod1G28 Mod1G29 Mod1G3 Mod1G30
                      Mod1G31 Mod1G32 Mod1G33 Mod1G34 Mod1G35 Mod1G36 Mod1G37
                      Mod1G38 Mod1G39 Mod1G4 Mod1G40 Mod1G41 Mod1G5 Mod1G6 Mod1G7
                      Mod1G8 Mod1G9 \
                      Mod1G1324 Mod1G1325 Mod1G1326 Mod1G1327 Mod1G1328 Mod1G1329
                      Mod1G1330 Mod1G1331 Mod1G1332 Mod1G1333 Mod1G1334 Mod1G1335
                      Mod1G1336 Mod1G1337 Mod1G1338 Mod1G1339 Mod1G1340 Mod1G1341
                      Mod1G1342 Mod1G1343 Mod1G1344 Mod1G1345 Mod1G1346 Mod1G1347
                      Mod1G1348 Mod1G1349 Mod1G1350 Mod1G1351 Mod1G1352 Mod1G1353
                      Mod1G1354 Mod1G1355 Hierarchicalc1355

* connect the VDD and VSS
v_my_dummy_VDD VDD 0 1.100000
v_my_dummy_VSS VSS 0 0

* connect the cells to power supply
vMod1_I1005 Mod1_I1005_dummy xMod1_cells_instance.Mod1_I1005 0
vMod1_I1005_dummy Mod1_I1005_dummy 0 pwl(0.000000e+00 1.100000e+00 1.000000e-12 ...
vMod1_I2232 Mod1_I2232_dummy xMod1_cells_instance.Mod1_I2232 0
vMod1_I2232_dummy Mod1_I2232_dummy 0 pwl(0.000000e+00 1.100000e+00 1.000000e-12 ...
vMod1_I288 Mod1_I288_dummy xMod1_cells_instance.Mod1_I288 0
vMod1_I288_dummy Mod1_I288_dummy 0 pwl(0.000000e+00 1.100000e+00 1.000000e-12 ...
... ← Παραλείπονται πηγές τάσης για λόγους συντομίας

* include probe and print statements
.probe tran i(vMod1_I1005)
.print tran i(vMod1_I1005)
.probe tran i(vMod1_I2232)
.print tran i(vMod1_I2232)
.probe tran i(vMod1_I288)
.print tran i(vMod1_I288)
... ← Παραλείπονται δηλώσεις probe και plot για λόγους συντομίας

* include analysis type
.tran lp 10000p
.save
.end
```

Βιβλιογραφία

- [1] Daniel Siu Regional Technical Marketing Manager ICD Asia-Pacific Operations Cadence Design Systems Inc. - *“Power grid analysis on IR drop and electromigration”*.
- [2] Selçuk Köse and Eby G. Friedman Department of Electrical and Computer Engineering University of Rochester Rochester, New York 14627 *“Fast Algorithms for IR Voltage Drop Analysis Exploiting Locality”*.
- [3] Yu ZHONG Dept. of Electrical and Computer Engineering Univ. of Illinois at Urbana-Champaign Urbana, IL, 61801, Martin D. F. Wong Dept. of Electrical and Computer Engineering Univ. of Illinois at Urbana-Champaign Urbana, IL, 61801 - *“Fast Algorithms for IR Drop Analysis in Large Power Grid”*.
- [4] Sam Chitwood, Ji Zheng - *“IR DROP in High-Speed IC Packages and PCBs”*.
- [5] Chris Halford Advanced Layout Solutions Ltd *“IR-Drop Analysis”*
- [6] Synopsys - *“Understand and Avoid Electromigration (EM) & IR-drop in Custom IP Blocks”*