



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ
ΥΠΟΛΟΓΙΣΤΩΝ

[HRY 419]-Ανάπτυξη Εργαλείων CAD για
Σχεδίαση Ολοκληρωμένων Κυκλωμάτων
Αναφορά 2^{ου} εργαστηρίου

Ιωάννης Περίδης
Α.Μ. 2018030069

20 Μαρτίου 2021

1 Εισαγωγή:

Σκοπός της εργαστηριακής άσκησης αυτής, είναι η κατανόηση της έννοιας του μετρημένου, αλλά και του αποδεκτού σφάλματος στην δημιουργία προσεγγιστικών εργαλείων CAD. Συγκεκριμένα, μας ζητήθηκε η δημιουργία ενός μακρομοντέλου ενός NMOS τρανζίστορ. Στόχος μας, ήταν η υλοποίηση ενός προσεγγιστικού μακρομοντέλου ταχύτερο σε χρόνο εύρεσης του ρεύματος μεταξύ drain-source, αλλά και φθηνότερο σε κόστος πράξεων, σε σχέση με τον χρόνο επίλυσης και κόστος που θα είχαμε χρησιμοποιώντας τις αναλυτικές εξισώσεις του NMOS. Φυσικά, απαραίτητη προϋπόθεση για την υλοποίησή μας, είναι τα αποτελέσματα του μακρομοντέλου να φράσσονται από ένα όριο αποδεκτού σφάλματος, με σκοπό να πετύχουμε μια ικανοποιητική σχέσης μεταξύ κόστους, ωφέλειας και ακρίβειας.

2 Υλοποίηση:

Ξεκινώντας, το πρόγραμμα δέχεται σαν είσοδο το συνολικό πλήθος των NMOS προς επίλυση, αλλά και τις διαφορετικές παραμέτρους που έχει το κάθε ένα:

V_{gs} (τάση μεταξύ gate-source), σαν έναν δεκαδικό αριθμό ακρίβειας float

V_{ds} (τάση μεταξύ drain-source), σαν έναν δεκαδικό αριθμό ακρίβειας float

W, L (πλάτος και μήκος καναλιού), σαν δεκαδικούς αριθμούς ακρίβειας float, παρόλο που παίρνουν μόνο ακεραίες τιμές, με σκοπό την ευκολία πράξεων και αποφυγή μετατροπών από integer σε float.

Οι τιμές των τάσεων κυμαίνονται από 0 έως μέγιστο 5 volt με συνεχές τιμές, ενώ οι τιμές των παραμέτρων διάστασης καναλιού παίρνουν διακριτές τιμές βήματος 1, από 2 έως 10.

Έπειτα, κάνοντας τους απαραίτητους υπολογισμούς, το πρόγραμμα δίνει σαν έξοδο την τιμή του ρεύματος I_{ds} (drain-source) υπολογισμένο με δύο τρόπους, μέσω των αναλυτικών εξισώσεων και μέσω του μακρομοντέλου. Ακόμη, δίνει σαν έξοδο και την σύγκριση των δύο ως προς την ακρίβεια και το κόστος τους. Δηλαδή υπολογίζει το ξεχωριστό σφάλμα του μακρομοντέλου ως προς τις εξισώσεις που προέκυψε από κάθε τρανζίστορ ξεχωριστά, αλλά και το συνολικό για ολόκληρες τις δοκιμές. Μαζί με αυτό, υπολογίζεται και το κόστος επίλυσης, δηλαδή το πλήθος των διαφορετικών πράξεων που χρειάστηκαν οι δύο τρόποι για τους υπολογισμούς, για τις πράξεις μεταξύ ακεραίων και μεταξύ δεκαδικών.

3 Επίλυση με χρήση αναλυτικών εξισώσεων NMOS:

Η υλοποίηση του κώδικα για επίλυση του τρανζίστορ με την χρήση των αναλυτικών εξισώσεων ήταν πολύ απλή. Χρησιμοποιήθηκαν οι παρακάτω εξισώσεις:

$$I_{DS} = k[2(V_{GS} - V_T)V_{DS} - V_{DS}^2] , \text{όταν } 0 \leq V_{DS} \leq V_{GS} - V_T$$

(Linear or Triode or Non – Saturated)

$$I_{DS} = k(V_{GS} - V_T)^2 , \text{όταν } 0 \leq V_{GS} - V_T \leq V_{DS}$$

(Saturated)

$$I_{DS} = 0 , \text{όταν } V_{GS} - V_T \leq 0$$

(Cutoff)

$$k = 0.2 \frac{W}{L} \text{ και } V_T = +1V_{olt}$$

Έτσι, αφού υπολογιζόταν το κ για κάθε τρανζίστορ, απλά γινόταν μια συνθήκη ελέγχου για τον καθορισμό της περιοχής λειτουργίας που βρισκόμαστε κάθε φορά και υπολογιζόταν αντίστοιχα το κάθε ρεύμα. Τα κ και τα ρεύματα Ids, αποθηκεύονται σε arrays για εκτύπωση αργότερα αν χρειαστεί.

Το κόστος υπολογισμού του κάθε κ, είναι ενιαίο σε όλα τα τρανζίστορ και ίσο με 1 floating point πολλαπλασιασμό και 1 floating point διαίρεση.

Το κόστος υπολογισμού του Ids , διαφέρει στις 3 περιπτώσεις. Για γραμμική περιοχή είναι το ακριβότερο ,καθώς και έχει τον πιο σύνθετο τύπο,5 floating point πολλαπλασιασμοί και 1 floating point αφαίρεση. Για περιοχή κορεσμού, δηλαδή για την περιοχή λειτουργίας και την πιο συχνή περίπτωση των NMOS τρανζίστορ, το κόστος άγεται σε 2 floating point πολλαπλασιασμούς και 1 floating point αφαίρεση. Τέλος, όπως είναι φανερό για την αποκοπή το κόστος είναι μιας ανάθεσης στο 0 ,δηλαδή 1 πράξη Integer.

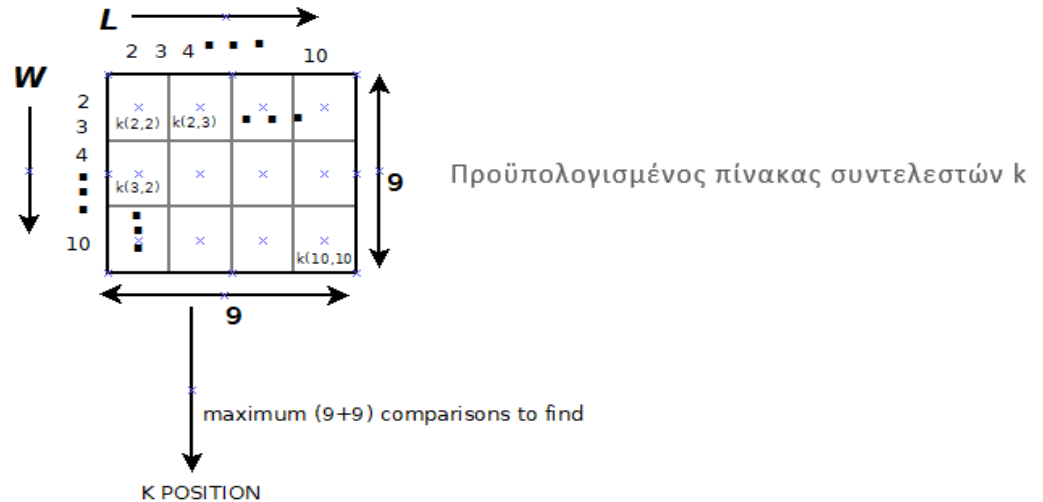
Να σημειωθεί εδώ πως τα κόστη αφαίρεσής που συμβαίνουν μέσα στις συνθήκες ελέγχου if , επίσης καταμετρούνται, συγκεκριμένα συνολικά και στα 3 ifs (ένα για κάθε περιοχή λειτουργίας), κάνω 4 αφαιρέσεις (2 στο saturated και 1 στις άλλες) , άρα θα έχω κόστος 4 πράξεις integer.

4 Επίλυση με χρήση μακρομοντέλου:

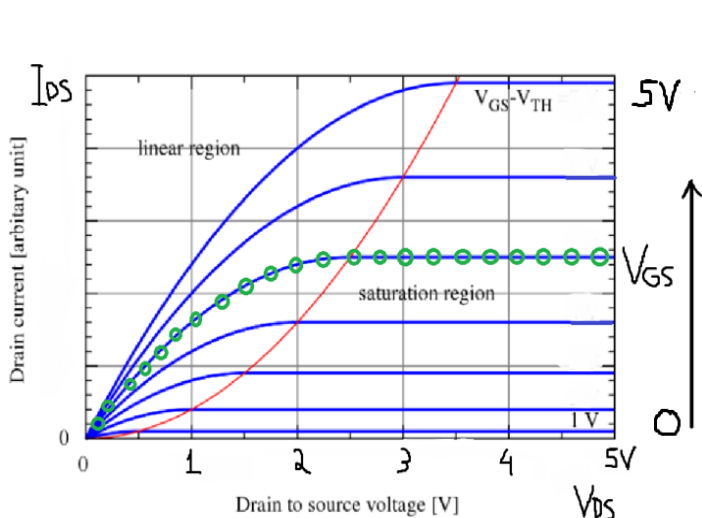
Για την δημιουργία του μακρομοντέλου, η πολυπλοκότητα της δημιουργίας του όπως και ήταν αναμενόμενο, αυξήθηκε σε έναν αρκετά μεγάλο βαθμό. Η λογική πίσω από την ιδέα της λειτουργίας του μακρομοντέλου είναι η εξής. Θα προσπαθήσουμε να εκμεταλλευτούμε το γεγονός ότι δίνεται η δυνατότητα χρήσης μικρών προϋπολογισμένων πινάκων έως 1000 στοιχείων. Η χρήση αυτών των πινάκων φυσικά δεν επιβραδύνει την ταχύτητα των υπολογισμών, γιατί ενώ χρειαζόμαστε προσβάσεις στην μνήμη για να πάρουμε τα στοιχεία από τους πίνακες, είναι αρκετά μικροί ώστε να χωράνε σε μια μνήμη cache με αποτέλεσμα το κόστος πρόσβασης να είναι μηδαμινό.

Αρχικά, θα δημιουργήσουμε έναν πίνακα με όλες τις δυνατές τιμές του σταθερού συντελεστή κ, με σκοπό να αποφύγουμε το κόστος υπολογισμού του σε κάθε ένα τρανζίστορ (αναλύθηκε παραπάνω). Για να το κάνουμε αυτό, θα χρειαστούμε έναν πίνακα με 81 στοιχεία, δηλαδή όλους τους δυνατούς συνδυασμούς της σχέσης 0.2 W/L (αυτό διότι το W,L παίρνουν 9 διαφορετικές τιμές το κάθε ένα). Ύστερα, δεδομένου των W,L που έδωσε ο χρήστης, θα γίνεται ένας αριθμός συγκρίσεων με τον προϋπολογισμένο πίνακα και θα παίρνεται από τον πίνακα έτοιμο το στοιχείο αυτό και θα καταχωρείται στο κ. Πρακτικά επομένως , εκτός από

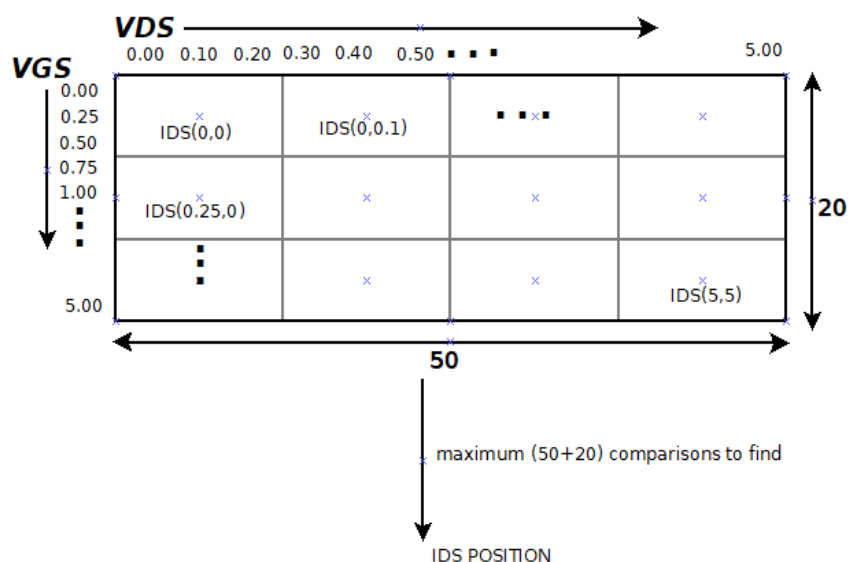
το κόστος πρόσβασης στην μνήμη (που δεν το καταμετράμε) δεν θα έχουμε κάποιο κόστος στον υπολογισμό του k και ταυτόχρονα, δεν έχουμε κάποιο σφάλμα τιμών, καθώς και τα μεγέθη W, L αντιστοιχίζονται ακριβώς στα πραγματικά (εφόσον παίρνουν τόσο συγκεκριμένες τιμές).



Αντίστοιχα, θα δημιουργήσουμε έναν πίνακα με όσες περισσότερες δυνατές τιμές του ρεύματος I_{ds} μπορούμε, (χωρίς να υπολογίζουμε τον παράγοντα k στις εξισώσεις). Στην περίπτωση αυτή, όμως το ρεύμα εξαρτάται από τις τιμές των τάσεων V_{gs} και V_{ds} , οι οποίες δεν έχουν διακριτές τιμές, αλλά συνεχείς στο διάστημα 0 έως 5 volt, οπότε δεν μπορούμε να πάρουμε όλες τις πιθανές τιμές, αλλά παίρνουμε συνδυασμούς τιμών μέχρι να φτάσουμε τις 1000, που είναι ο μέγιστος επιτρεπτός πίνακας. Επιλέγω να πάρω 20 τιμές του V_{gs} , δηλαδή τιμές από 0 έως 5 με βήμα 0.25 και 50 τιμές του V_{ds} , δηλαδή από 0 έως 5 με βήμα 0.1. Άρα γεμίζω έναν πίνακα με $20 \times 50 = 1000$ διαφορετικές τιμές του I_{ds} . Όπως πριν, προσπαθώ με συγκρίσεις των δεδομένων τάσεων του χρήστη να αντιστοιχίσω τις τιμές αυτές σε κάποιο ρεύμα του πίνακά μου. Παρόλα αυτά, σε αυτήν την περίπτωση εφόσον δεν έχω όλες τις δυνατές τιμές βρίσκω την κοντινότερη τιμή του V_{gs} και του V_{ds} στις αληθινές και παίρνω τον συνδυασμό αυτών των 2 στον πίνακα. Όπως είναι φανερό εφόσον έχω μεγαλύτερη διακριτικότητα στο V_{ds} , θα βρίσκω συνήθως καλύτερες προσεγγίσεις της αληθινής τιμής, ενώ για το V_{gs} θα υπάρχουν πιο συχνά αποκλίσεις. Τέλος πολλαπλασιάζω τα δύο στοιχεία από τους πίνακες k και I_{ds} και βρίσκω το τελικό ρεύμα με κόστος μονάχα ενός πολλαπλασιασμού floating point! Παρόλα αυτά, έχει εισαχθεί στα αποτελέσματα ένα σφάλμα το οποίο είναι ανάλογο με τις αποκλίσεις στις τιμές των τάσεων (άλλες φορές μεγαλύτερο, άλλες φορές μικρότερο ή ακόμα και σχεδόν μηδενικό). Τεστάροντας στην συνέχεια το μακρομοντέλο σε ένα μεγάλο εύρος τιμών θα αποδείξουμε ότι το συνολικό σφάλμα, είναι εντός του αποδεκτού.



Διάγραμμα (I_{ds} - V_{ds}) για πολλά διαφορετικά V_{gs}



Προϋπολογισμένος πίνακας ρευμάτων I_{ds}

Για την υλοποίηση αυτών σε κώδικα, αρχικά δημιουργήθηκαν για όλες τις τιμές που προαναφέρθηκαν μέσω 2 for loops οι πίνακες με τις τιμές των ρευμάτων και των συντελεστών κ. Οι τιμές αυτές προέρχονται επίσης από τις εξισώσεις του NMOS. Να σημειωθεί εδώ πως στην πραγματικότητα δεν είναι πίνακες αλλά διανύσματα. Για τα ρεύματα, έχουμε τα πρώτα 50 στοιχεία να είναι όλες οι τιμές του V_{ds} για $V_{gs}=0$, μετά το ίδιο για $V_{gs}=0.25/0.5/0.75\dots$ και συνεχίζει έτσι μέχρι να γεμίσουν και οι 1000 θέσεις. Αντίστοιχα για τα κ, έχουμε όλες τις του L για $W=2$ αρχικά και μετά για 3,4,5... μέχρι να γεμίσουν και οι 81 θέσεις.

Στην συνέχεια με άλλα 4 for loops ένα για κάθε παράμετρο του NMOS, V_{gs}, V_{ds}, W, L γίνεται η αντιστοίχιση της τιμές που έχουμε σαν είσοδο με την τιμή που θα χρησιμοποιηθεί στον υπολογισμό. Συγκεκριμένα για W, L οι τιμές θα ταυτίζονται άρα η διαφορά εισόδου και τιμής υπολογισμού τους θα πρέπει να είναι 0, ενώ για το V_{gs} δεχόμαστε η διαφορά τους να είναι μικρότερη ή ίση της διακριτικότητας 0.249, και για την V_{ds} δεχόμαστε διαφορά μικρότερη ή ίση της διακριτικότητας 0.09. Επομένως, εφόσον τα διανύσματα έχουν αυτή την διάταξη οι δύο θέσεις σε αυτά που μας ενδιαφέρουν θα είναι :

$$kPosition = W * 9 + L \quad \text{και} \quad I_{Ds}position = V_{GS} * 50 + V_{DS}$$

Τέλος, απλά κάνουμε τον πολλαπλασιασμό κ με I_{ds} για τα 2 positions Που βρήκαμε , σε μια κοινή περίπτωση τώρα, καθώς και ο διαχωρισμός των ρευμάτων έχει γίνει στον προϋπολογισμένο πίνακα.

Σε ένα corner case έχουμε μια ιδική μεταχείριση που θα δούμε στην συνέχεια.

5 Υπολογισμός σφάλματος:

Για να υπολογιστεί το ποσοστιαίο σφάλμα του κάθε τρανζίστορ ξεχωριστά, χρησιμοποιήθηκε ο τύπος:

$$ERROR(\%) = \left(\frac{I_{Ds}Macromodel}{I_{Ds}Equations} - 1 \right) * 100$$

Ενώ για το συνολικό σφάλμα σε ποσοστό, έχω το άθροισμα των απολύτων των σφαλμάτων σε κάθε τρανζίστορ δια το ολικό πλήθος των τρανζίστορ που επιλύθηκαν:

$$ERROR(\%)_{TOTAL} = \frac{\sum |error(\%)|}{total\ NMOS\ number}$$

6 Testbench και αποτελέσματα:

Για να ελεγχθεί το μακρομοντέλο σωστά, δημιουργήθηκε ένα testbench σε αρχείο .txt που έχει format η κάθε σειρά να αποτελείται από τους 4 συντελεστές του NMOS , V_{gs}, V_{ds}, W, L με αυτήν την συγκεκριμένη σειρά (όπου οι τάσεις είναι δεκαδικές ενώ οι διαστάσεις ακέραιες). Χρησιμοποιήθηκε 1248 διαφορετικές τιμές τρανζίστορ(γραμμές), (δηλαδή 4992 διαφορετικά στοιχεία), με σκοπό να έχουμε μια απόλυτα ολοκληρωμένη εικόνα για την αποτελεσματικότητά του μακρομοντέλου , σε κάθε δυνατή περιοχή λειτουργίας του. Επιλέχθηκαν 50 τυχαίες τιμές του v_{ds} , αυξανόμενες από 0 έως 5 με ένα μικρό ,μη σταθερό τυχαίο βήμα. Ακόμη, επιλέχθηκαν και 50 τυχαίοι συνδιασμοί των W, L . Κάθε μια από αυτές τις τιμές συνδυάζεται αντίστοιχα με 25 τυχαίες τιμές V_{gs} από 0 έως 5, αυξανόμενες με λίγο μεγαλύτερο τυχαίο βήμα.

Προσοχή για να τρέξει το testbench, πρέπει να εισαχθεί στον κώδικα η σωστή θέση του αρχείου testbench.txt

Παρακάτω, φαίνεται ένα παράδειγμα από τον τρόπο εκτύπωσης των εισόδων από το testbench:

```
*****
***** NMOS TRANSISTOR MACROMODEL: *****
*****

ALL THE NMOS TRANSISTOR's VALUES: Number of NMOS Transistors Tested: 1248
=====
| VGS | VDS | W | L |
=====
U 1| 0.000 | 0.000 | 2.000 | 10.000
U 2| 0.000 | 0.160 | 3.000 | 9.000
U 3| 0.000 | 0.240 | 4.000 | 8.000
U 4| 0.000 | 0.370 | 5.000 | 7.000
U 5| 0.000 | 0.440 | 6.000 | 6.000
U 6| 0.000 | 0.520 | 7.000 | 5.000
U 7| 0.000 | 0.680 | 8.000 | 4.000
U 8| 0.000 | 0.730 | 9.000 | 3.000
U 9| 0.000 | 0.800 | 10.000 | 2.000
U 10| 0.000 | 0.920 | 9.000 | 2.000
```

U1006	4.190	1.730	3.000	10.000
U1007	4.190	1.890	3.000	4.000
U1008	4.190	1.920	4.000	4.000
U1009	4.190	2.070	5.000	6.000
U1010	4.190	2.160	6.000	2.000
U1011	4.190	2.240	3.000	3.000
U1012	4.190	2.310	4.000	10.000
U1013	4.190	2.440	2.000	4.000
U1014	4.190	2.520	10.000	6.000
U1015	4.190	2.680	7.000	5.000
U1016	4.190	2.730	8.000	7.000
U1017	4.190	2.800	9.000	3.000

Παρακάτω, φαίνεται ένα παράδειγμα από τον τρόπο εκτύπωσης των αποτελεσμάτων-εξόδων:

TRANSISTOR NUMBER	IDS EQN (mA)	IDS MACROMODEL (mA)	ERROR (perCent)
U 944	2.325780	2.226000	-4.29
U 945	4.192000	4.160000	-0.76
U 946	4.239360	4.131001	-2.56
U 947	2.688000	2.666667	-0.79
U 948	1.981280	1.886500	-4.78
U 949	1.470600	1.382400	-6.00
U 950	1.066317	1.018333	-4.50
U 951	0.757029	0.736000	-2.78
U 952	0.521963	0.506250	-3.01
U 953	0.325547	0.312889	-3.89
U 954	0.447378	0.438600	-1.96
U 955	1.176525	1.134000	-3.61
U 956	1.582080	1.558000	-1.52
U 957	1.369650	1.333333	-2.65
U 958	5.028480	4.914001	-2.28
U 959	1.702400	1.672000	-1.79
U 960	0.689304	0.680800	-1.23
U 961	0.878400	0.864000	-1.64
U 962	2.956800	2.916667	-1.36

Παρακάτω, φαίνεται η εκτύπωση του συνολικού σφάλματος για το συγκεκριμένο testbench:

```
=====
MEAN VALUE OF ALL THE TRANSISTOR's ERRORS IS: 8.401 (Per Cent)
=====
```

Παρατηρήθηκε πως το συνολικό σφάλμα σε ένα πολύ αναλυτικό , σε όλο το εύρος περιπτώσεων testbench ήταν μικρότερο από 10% , επομένως ήταν εντός του αποδεκτού.

Αναλυτικά για τα σφάλματα:

Οι τιμές όλων των ρευμάτων του μακρομοντέλου είναι κατά ένα πολύ μικρό ποσοστό μικρότερες από ότι οι τιμές των εξισώσεων, επομένως αυτό δικαιολογεί το αρνητικό πρόσημο των σφαλμάτων.

Είναι φανερό πως όσο μεγαλύτερη απόκλιση υπάρχει μεταξύ κυρίως της τάσης V_{gs} , καθώς και αυτή παίζει τον καταλυτικό παράγοντα επηρεασμού των σφαλμάτων, εφόσον είχε την μικρότερη διακριτική ικανότητα , κατά 0.25 μονάδες, τόσο μεγαλύτερο θα είναι και το σφάλμα. Αντίστοιχα όσο πιο κοντά είναι η V_{gs} στην πραγματική τάση, τόσο πιο μικρό έως και μηδενικό (αν είναι ίδιες) είναι το σφάλμα. Το μεγαλύτερο σφάλμα λοιπόν δημιουργείται σε περιπτώσεις , όπου έχω τάση εισόδου $V_{gs}=X,49/X,99$ οι οποίες πηγαίνουν οριακά στις περιπτώσεις $V_{gs}=X,25/X,50$ αυτό διότι τότε έχουν την μέγιστη επιτρεπτή απόσταση μεταξύ τάσεων θα είναι η διαφορά τους 0,24. Φυσικά, όσο μικρότερο είναι το X , τόσο περισσότερο επηρεάζει αυτή η διαφορά στο σύνολο της τάσης, άρα και στο σφάλμα (μέγιστο σφάλμα στο $X=1,49/1,99$). Τέτοιες περιπτώσεις έχουν προστεθεί στο tesbnech, για να γίνει κατανοητό πως ακόμα και αν υπάρχουν ατυχίες και πάλι το συνολικό error θα είναι μικρότερο του 10%.

Corner cases:

Οι οριακές καταστάσεις για τις τάσεις να είναι και οι 2 στα 5 Volt λειτουργούν τέλεια, με μηδενικό σφάλμα! Αυτό διότι οι τιμές για V_{gs} και V_{ds} ίσες με το 5 είναι εντός του προϋπολογισμένου πίνακα.

Σε μια μόνο περίπτωση χρειάστηκε ιδιαίτερη αντιμετώπιση του αποτελέσματος. Συγκεκριμένα, για κάποιες πολύ μικρές τιμές της τάσης V_{gs} , η λύση των εξισώσεων, πηγαίνει σε saturation mode ,ενώ λόγω κακής διακριτικότητας της V_{gs} του μακρομοντέλου, η λύση πηγαίνει σε cutoff mode αφού το $V_{gs}-V_t < 0$. Οπότε από τις εξισώσεις το ρεύμα παίρνει μια πάρα πολύ μικρή τιμή, ενώ από το μακρομοντέλο το ρεύμα παίρνει μηδενική τιμή, οπότε για οποιαδήποτε τιμή ρεύματος που κατατάσσεται σε αυτό το σενάριο, θα έχω σφάλμα ίσο με -100%. Για την αντιμετώπιση αυτού του γεγονότος, υλοποίησα μια συνθήκη ελέγχου που λέει πως αν το ρεύμα του μακρομοντέλου είναι 0 και το ρεύμα των εξισώσεων είναι διαφορετικό του 0, τότε θέσε το νέο ρεύμα του μακρομοντέλου να είναι το $k*0,029$.

Το 0,029 σαν τιμή επιλέχθηκε μετά από πολλές δοκιμές και καταλήγει να δίνει λύσεις με ελάχιστο σφάλμα μονάχα 0,35%.(παράδειγμα παρακάτω)

U 207	0.000000	0.000000	0.00
U 208	0.000000	0.000000	0.00
U 209	0.000000	0.000000	0.00
U 210	0.001920	0.001933	0.69
U 211	0.002890	0.002900	0.35
U 212	0.004129	0.004143	0.35
U 213	0.005780	0.005800	0.35
U 214	0.008092	0.008120	0.35
U 215	0.011560	0.011600	0.35
U 216	0.017340	0.017400	0.35
U 217	0.028900	0.029000	0.35
U 218	0.026010	0.026100	0.35
U 219	0.015413	0.015467	0.35
U 220	0.010115	0.010150	0.35
U 221	0.006936	0.006960	0.35

Σε αυτό το σημείο πρέπει να γίνει μια διευκρίνιση για την οριακή αυτή κατάσταση. Το κόστος όταν βρισκόμαστε στην κατάσταση αυτή, αυξάνεται κατά μια πράξη floating point πολλαπλασιασμού, διότι ενώ υπολογίστηκε ήδη μια φορά , ύστερα διορθώνεται με δεύτερο υπολογισμό.

Ακόμη, σχετικά με το κόστος προστέθηκε ακόμα ένας έλεγχος if (που έχει κόστος 1 integer) για την περίπτωση που είμαι σε cutoff, ο οποίος απλά αναθέτει με κόστος 1 πράξη integer το 0 στο ρεύμα. Το κόστος για τον έλεγχο θα συμβαίνει σε κάθε επανάληψη (δηλαδή για όλα τα τρανζίστορ) , αλλά γλιτώνω με αυτόν τον τρόπο τον πολλαπλασιασμό floating point που

θα γινόταν για να πάρω την τιμή ρεύματος 0 από τον προϋπολογισμένο πίνακα και να την πολλαπλασιάσω με την τιμή κ από τον άλλο πίνακα. Άρα πιθανότατα θα μπορούσαν να είχαν αφαιρεθεί κάποιες τιμές του πίνακα ρευμάτων που δίνουνε cutoff και να είχαν γίνει πιο συμπυκνωμένες οι τιμές στο linear και στο saturated mode ή να είχε λιγότερα στοιχεία ο αρχικός προϋπολογισμένος πίνακας.

Τέλος, δεν αναφέρθηκε πριν πως γλιτώνω το κόστος των Integers συγκρίσεων για τις περιοχές saturated και Linear καθώς και στις δύο εκτελείται ο ίδιος τύπος , δηλαδή ο απλός πολλαπλασιασμός των στοιχείων των πινάκων.

7 Σύγκριση αποτελεσμάτων μεθόδων:

Συγκρίνοντας τις δύο μεθόδους , όπως ήταν και επιθυμητό, δημιουργήσαμε ένα μακρομοντέλο με μέσο όρο σφάλματος μικρότερο του 10% με εξαιρετικά μικρότερο συνολικό κόστος υπολογισμών. Πρακτικά το κόστος σχεδόν μειώθηκε μονάχα σε έναν πολλαπλασιασμό floating point ανά τρανζίστορ και σε μια πράξη Integer για τον έλεγχο. Η σημαντικότερη πράξη που αποφεύχθηκε είναι η διαίρεση , καθώς και είναι κοστίζει περισσότερο από όλα.

Φυσικά δεν υπολογίστηκε το κόστος των προσβάσεων της στην μνήμη ούτε το κόστος δημιουργίας των αρχικών πινάκων. Αυτό διότι σε μια προσομοίωση με χιλιάδες NMOS το κόστος αυτό θα ήταν αμελητέο.

(Δεν είμαι σίγουρος αν στο κόστος πράξεων θα έπρεπε να υπολογιστούν και οι συγκρίσεις για την αντιστοίχιση των τιμών από τους πίνακες ή όχι, καθώς και αυτό δεν είναι κάποιο καθαρά υπολογιστικό κόστος).

Παρακάτω, φαίνεται η εκτύπωση όλων των συνολικών κοστών του testbench = το πλήθος από κάθε είδους πράξεις:

OPERATIONS	EQN	MACROMODEL
INTEGER/BOOLEAN	5200	1508
FP MUL	4654	1323
FP ADD	0	0
FP SUBS	1040	0
FP DIVS	1248	0