

**ΕΛΕΓΧΟΣ ΟΡΘΗΣ ΛΕΙΤΟΥΡΓΙΑΣ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
ΚΑΙ
ΣΧΕΔΙΑΣΗ ΓΙΑ ΕΥΚΟΛΟ ΕΛΕΓΧΟ**

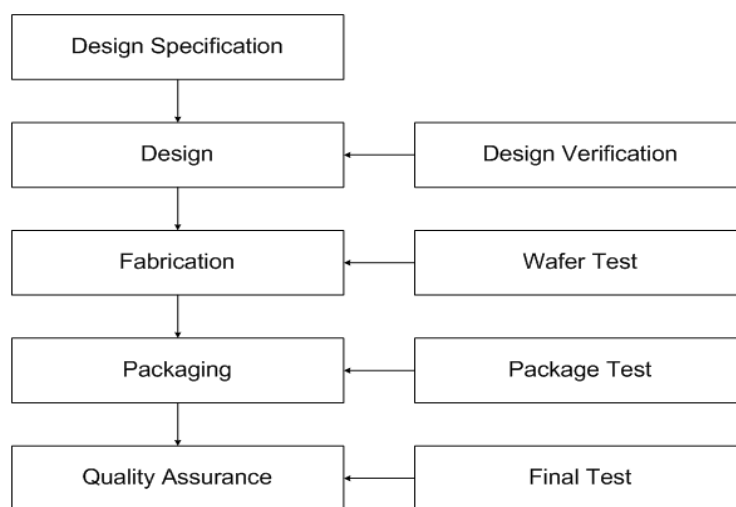
Δ. Νικολός

Κεφάλαιο 1. Εισαγωγή στον έλεγχο ορθής λειτουργίας ψηφιακών συστημάτων

Σκοπός του ελέγχου ορθής λειτουργίας είναι να ελέγξει εάν ένα προϊόν λειτουργεί σύμφωνα με τις προδιαγραφές του.

Υπάρχουν πολλά είδη ελέγχου της ορθής λειτουργίας. Τρεις γενικές κατηγορίες είναι: η επιβεβαίωση σχεδιασμού, ο έλεγχος ορθής κατασκευής και ο περιοδικός έλεγχος.

Σκοπός της επιβεβαίωσης σχεδιασμού είναι να ανιχνεύσει λάθη που έγιναν κατά τη διαδικασία σχεδίασης ενός προϊόντος, γεγονός που θα είχε ως συνέπεια, αν αυτό κατασκευαστεί να μην λειτουργεί σύμφωνα με τις προδιαγραφές του. Σκοπός του ελέγχου μετά την κατασκευή είναι να ανιχνεύσει ελαττώματα που προέκυψαν κατά τη διαδικασία της κατασκευής, με συνέπεια αυτό να μη λειτουργεί σύμφωνα με τις προδιαγραφές του. Σκοπός του περιοδικού ελέγχου ορθής λειτουργίας είναι να ανιχνεύσει ελαττώματα που προέκυψαν κατά τη διάρκεια της χρήσης του προϊόντος στο πεδίο της εφαρμογής.



Σχήμα 1.1 Διαδικασία ανάπτυξης Ολοκληρωμένου Κυκλώματος, OK

Στο παραπάνω σχήμα μπορούμε να δούμε τις σχετικές θέσεις στις οποίες λαμβάνει χώρα η επιβεβαίωση σχεδιασμού και ο έλεγχος ορθής λειτουργίας κατά την κατασκευή ενός ολοκληρωμένου κυκλώματος. Παρατηρούμε ότι η επιβεβαίωση σχεδιασμού γίνεται πριν την κατασκευή του ολοκληρωμένου κυκλώματος. Στην πραγματικότητα δεν περιμένουμε να ολοκληρωθεί η σχεδίαση ενός ολοκληρωμένου κυκλώματος και στη συνέχεια να κάνουμε την επιβεβαίωση σχεδιασμού. Για κάθε υπομονάδα του συστήματος η οποία σχεδιάζεται θα πρέπει να επιβεβαιώνεται και η ορθή της λειτουργία, δηλαδή ότι λειτουργεί σύμφωνα με τις προδιαγραφές. Όσο πιο γρήγορα διαπιστώνεται το σχεδιαστικό λάθος τόσο μικρότερο είναι το κόστος διόρθωσής του. Όταν σχεδιαστεί μια μονάδα χρησιμοποιώντας υπομονάδες, των οποίων η ορθή σχεδίαση έχει επιβεβαιωθεί, θα πρέπει επίσης να επιβεβαιωθεί η ορθή

σχεδίαση της μονάδας αυτής. Μ' αυτό τον τρόπο θα πρέπει να γίνεται η επιβεβαίωση της ορθής σχεδίασης κάθε τμήματος του συστήματος έως ότου φτάσουμε στη σχεδίαση ολόκληρου του ολοκληρωμένου κυκλώματος.

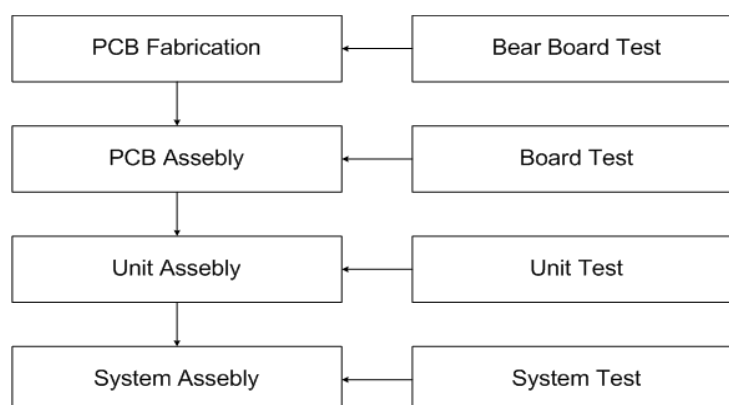
Στο παραπάνω σχήμα μπορούμε να δούμε επίσης ότι ο έλεγχος της ορθής λειτουργίας μετά την κατασκευή αποτελείται από διάφορους ελέγχους, οι οποίοι λαμβάνουν χώρα σε διάφορα βήματα της κατασκευαστικής διαδικασίας. Ο πρώτος έλεγχος λαμβάνει χώρα αμέσως μετά την κατασκευή των ολοκληρωμένων κυκλωμάτων πάνω στο δίσκο πυριτίου (wafer test). Τα ελαττωματικά ολοκληρωμένα κυκλώματα (dies) σημειώνονται και μετά το κόψιμο του δίσκου πυριτίου (wafer) σε τμήματα, εκ των οποίων κάθε ένα περιέχει και ένα ολοκληρωμένο κύκλωμα, τα σημειωμένα ως ελαττωματικά ολοκληρωμένα κυκλώματα απομακρύνονται. Κάποια από τα ελαττωματικά ολοκληρωμένα κυκλώματα θα οδηγηθούν στο τμήμα διαγνωστικού ελέγχου της εταιρίας, για να βρεθούν οι αιτίες που προκάλεσαν τα ελαττώματα κατά τη διαδικασία κατασκευής και να γίνουν οι απαιτούμενες διορθώσεις, προσαρμογές, στη διαδικασία κατασκευής ώστε να μειωθεί ο αριθμός των ελαττωματικών ολοκληρωμένων κυκλωμάτων που κατασκευάζονται.

Καθένα από τα ολοκληρωμένα κυκλώματα που πέρασαν τον παραπάνω έλεγχο τοποθετείται μέσα σε ένα πλαστικό ή κεραμικό πακέτο και συνδέεται με τον απαιτούμενο αριθμό ακροδεκτών. Κατά τη διαδικασία αυτή είναι δυνατόν να καταστραφεί το ολοκληρωμένο κύκλωμα ή να μη γίνει σωστή σύνδεση των ακροδεκτών του με τους ακροδέκτες του πακέτου. Γι' αυτό ακολουθεί έλεγχος της ορθής λειτουργίας του πακεταρισμένου ολοκληρωμένου κυκλώματος (chip).

Ο έλεγχος της ορθής λειτουργίας των ολοκληρωμένων κυκλωμάτων που γίνεται όταν αυτά βρίσκονται πάνω στο δίσκο πυριτίου (wafer test) είναι ο πλέον απαιτητικός και απαιτεί το ολοκληρωμένο κύκλωμα να έχει σχεδιαστεί κατάλληλα ώστε να μπορεί να ελεγχθεί η ορθή του λειτουργία μετά την κατασκευή του. Εμείς θα ασχοληθούμε με τις διαδικασίες και τις τεχνικές σχεδίασης των ολοκληρωμένων κυκλωμάτων ώστε να μπορεί να γίνει ο έλεγχος της ορθής τους λειτουργίας μετά την κατασκευή τους εύκολα και οικονομικά.

Όπως και στην επιβεβαίωση σχεδιασμού, έτσι και στον έλεγχο ορθής λειτουργίας μετά την κατασκευή, όσο πιο γρήγορα στην κατασκευαστική διαδικασία διαπιστωθεί ότι ένα ολοκληρωμένο κύκλωμα είναι ελαττωματικό τόσο μικρότερο είναι το κόστος. Για παράδειγμα ένα ελαττωματικό ολοκληρωμένο κύκλωμα δεν υπάρχει λόγος να τοποθετηθεί σε πακέτο.

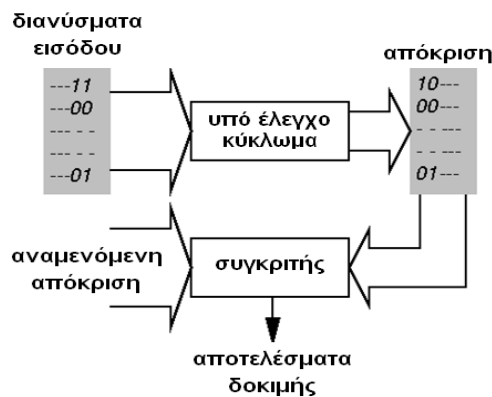
Ένα προϊόν του οποίου δεν μπορεί να ελεγχθεί η ορθή του λειτουργία είναι άχρηστο, δεν υπάρχει λόγος παραγωγής του.



Σχήμα 1.2 Διαδικασία ανάπτυξης συστήματος

Στο παραπάνω σχήμα μπορούμε να δούμε τα βήματα κατασκευής ενός συστήματος και τον έλεγχο της ορθής λειτουργίας που λαμβάνει χώρα σε κάθε βήμα. Μετά την κατασκευή της πλακέτας γίνεται έλεγχος της ορθής της λειτουργίας. Μετά την συναρμολόγηση της πλακέτας γίνεται έλεγχος της ορθής λειτουργίας και ούτω καθεξής. Είναι προφανές ότι το κόστος θα ήταν πάρα πολύ μεγαλύτερο εάν δεν γίνονταν έλεγχος της πλακέτας μετά την κατασκευή της και αναβάλλονταν να γίνει μετά τη συναρμολόγηση. Τότε θα είχαμε το πρόσθετο κόστος όλων των ηλεκτρονικών στοιχείων που θα είχαμε συνδέσει πάνω σε μια ελαττωματική πλακέτα.

Μπορούμε να διακρίνουμε τον έλεγχο ορθής λειτουργίας μετά την κατασκευή σε παραμετρικό έλεγχο και λογικό έλεγχο. Στον παραμετρικό έλεγχο μετράμε τις τιμές του ρεύματος και του δυναμικού. Στο λογικό έλεγχο μας ενδιαφέρουν οι λογικές τιμές 0 και 1. Προφανώς και πάλι μετράμε το δυναμικό αλλά δεν μας ενδιαφέρουν οι ακριβείς τιμές δυναμικού, αλλά μόνον εάν αυτές αντιστοιχούν στη λογική τιμή 1 ή τη λογική τιμή 0.



Σχήμα 1.3 Η αρχή του λογικού ελέγχου

Παρατηρούμε ότι στο σχήμα 1.3 το υπό έλεγχο κύκλωμα δέχεται ως εισόδους συνδυασμούς λογικών τιμών 0 και 1. Κάθε τέτοιος συνδυασμός λογικών τιμών αποτελεί ένα διάνυσμα δοκιμής. Για κάθε συνδυασμό τιμών εισόδου, δηλαδή για κάθε διάνυσμα δοκιμής, παίρνουμε μια απόκριση του υπό έλεγχο κυκλώματος. Πριν την κατασκευή του υπό έλεγχο κυκλώματος, με λογική εξομοίωση, για κάθε διάνυσμα δοκιμής πήραμε την αναμενόμενη απόκριση του κυκλώματος. Επομένως εάν το κύκλωμα έχει κατασκευασθεί σωστά αυτές πρέπει να είναι οι αποκρίσεις του όταν το οδηγήσουμε με τα διανύσματα δοκιμής. Επομένως κατά τον έλεγχο της ορθής του λειτουργίας συγκρίνουμε τις αποκρίσεις του υπό έλεγχο κυκλώματος με τις αναμενόμενες αποκρίσεις. Εάν οι αποκρίσεις του υπό έλεγχο κυκλώματος είναι ταυτοτικές με τις αναμενόμενες, τότε θεωρούμε ότι το κύκλωμα δεν έχει ελαττώματα. Διαφορετικά θεωρούμε ότι είναι ελαττωματικό.

Μπορούμε να διακρίνουμε το λογικό έλεγχο σε λειτουργικό έλεγχο και σε δομικό έλεγχο. Στο λειτουργικό έλεγχο, ο έλεγχος της ορθής λειτουργίας του κυκλώματος βασίζεται στη λειτουργική συμπεριφορά του. Για παράδειγμα ο έλεγχος της ορθής

λειτουργίας ενός αθροιστή βασίζεται στο γεγονός ότι στις εξόδους του θα πρέπει να πάρουμε το άθροισμα των τιμών με τις οποίες οδηγούμε τις εισόδους του. Στο λειτουργικό έλεγχο δεν χρειάζεται να γνωρίζουμε την εσωτερική δομή του υπό έλεγχο κυκλώματος, παρά μόνο τη λειτουργία του. Στο δομικό έλεγχο ελέγχουμε εάν κάθε συνιστώσα (τρανζίστορ, πύλη κλπ.), του υπό έλεγχο κυκλώματος λειτουργεί όπως αναμένεται.

Οι φυσικές βλάβες (*Physical failures*) ή ελαττώματα (*defects*) εκδηλώνονται ως ηλεκτρικές βλάβες και ερμηνεύονται σε λογικό επίπεδο ως σφάλματα (*faults*).

Για παράδειγμα

- ελάττωμα: βραχυκύκλωμα μιας γραμμής του κυκλώματος με τη γείωση
- ηλεκτρική βλάβη: ρεύμα τρέχει από τη γραμμή του κυκλώματος προς τη γείωση
- σφάλμα: η γραμμή του κυκλώματος έχει μόνιμα τη λογική τιμή 0

Τα είδη των ελαττωμάτων που μπορούν να εμφανιστούν σε ένα ολοκληρωμένο κύκλωμα είναι πάρα πολλά. Είναι λοιπόν πάρα πολύ δύσκολο να ασχοληθούμε με κάθε ένα από αυτά τα ελαττώματα. Το μεγάλο πλεονέκτημα των σφαλμάτων είναι ότι πολλά ελαττώματα αντιστοιχούν σε πολύ μικρότερο αριθμό από είδη σφαλμάτων ή ακόμη και σε ένα είδος σφάλματος. Επομένως βολεύει να ασχολούμαστε με την ανίχνευση σφαλμάτων.

Το μοντέλο απλών σφαλμάτων μόνιμης τιμής θεωρεί ότι κάθε φορά είναι δυνατόν μόνο μια γραμμή του κυκλώματος να έχει κολλήσει στη λογική τιμή 0 ή 1. Ένα σφάλμα ανιχνεύεται από ένα διάνυσμα δοκιμής (*test pattern* ή *test vector*). Αυτό είναι ένας συνδυασμός λογικών τιμών που επιβεβαιώνει την παρουσία σφάλματος. Σύνολο δοκιμής (*Test set*) είναι ένα σύνολο από διανύσματα δοκιμής

είσοδοι	απόκριση χωρίς σφάλμα	Απόκριση με σφάλμα					
		A/0	B/0	Z/0	A/1	B/1	Z/1
00	0	0	0	0	0	0	1
01	0	0	0	0	1	0	1
10	0	0	0	0	0	1	1
11	1	0	0	0	1	1	1

Σχήμα 1.4 Ανίχνευση απλών σφαλμάτων μόνιμης τιμής σε πύλη AND δύο εισόδων.

Θεωρήστε μια λογική πύλη AND με εισόδους A και B και έξοδο Z. Στον πίνακα δίνονται όλοι οι δυνατοί συνδυασμοί εισόδων της πύλης καθώς και η τιμή της εξόδου της όταν η πύλη δεν έχει κάποιο σφάλμα. Στη συνέχεια θα εξετάσουμε ποια θα είναι η τιμή της εξόδου της πύλης όταν μια από τις εισόδους της ή η έξοδος της έχει σφάλμα μόνιμης τιμής.

Κατ' αρχήν ας μελετήσουμε το σφάλμα η είσοδος A να έχει κολλήσει στη λογική τιμή 0, A/0. Από τον πίνακα παρατηρούμε ότι όταν η είσοδος A έχει κολλήσει στη λογική τιμή 0, στήλη A/0, τότε η έξοδος της πύλης έχει πάντα τη λογική τιμή 0. Ωστόσο μόνον όταν οδηγούμε και τις δύο εισόδους της πύλης με την λογική τιμή 1, η έξοδος της πύλης έχει τιμή διαφορετική της αναμενόμενης. Επομένως το σφάλμα αυτό ανιχνεύεται μόνον από το συνδυασμό εισόδων 11, δηλαδή το διάνυσμα δοκιμής του είναι το διάνυσμα δοκιμής 11.

Από τον πίνακα μπορούμε να δούμε ότι το σφάλμα η είσοδος B να έχει κολλήσει στη λογική τιμή 0, B/0 ή η έξοδος Z να έχει κολλήσει στη λογική τιμή 0, Z/0, ανιχνεύεται επίσης μόνο από το διάνυσμα δοκιμής 11.

Από τον πίνακα μπορούμε επίσης εύκολα να δούμε ότι το διάνυσμα δοκιμής που ανιχνεύει το σφάλμα η είσοδος A να έχει κολλήσει στη λογική τιμή 1, A/1, είναι το διάνυσμα 01 ενώ το διάνυσμα ανίχνευσης του σφάλματος η είσοδος B να έχει κολλήσει στη λογική τιμή 1, B/1, είναι το διάνυσμα 10.

Παρατηρούμε επίσης ότι το σφάλμα η έξοδος της πύλης Z να έχει κολλήσει στη λογική τιμή 1, Z/1, ανιχνεύεται από τρία διαφορετικά διανύσματα, τα διανύσματα δοκιμής 00, 01 και 10.

είσοδοι	απόκριση χωρίς σφάλμα	Απόκριση με σφάλμα					
		A/0	B/0	Z/0	A/1	B/1	Z/1
00	0	0	0	0	1	1	1
01	1	1	0	0	1	1	1
10	1	0	1	0	1	1	1
11	1	1	1	0	1	1	1

Σχήμα 1.5 Ανίχνευση απλών σφαλμάτων μόνιμης τιμής σε πύλη OR δύο εισόδων.

Θεωρήστε μια λογική πύλη OR με εισόδους A και B και έξοδο Z. Στον πίνακα δίνονται όλοι οι δυνατοί συνδυασμοί εισόδων της πύλης, η τιμή της εξόδου της όταν η πύλη δεν έχει κανένα σφάλμα καθώς και η τιμή της εξόδου της για κάθε απλό σφάλμα μόνιμης τιμής.

Ένα μέτρο της αποτελεσματικότητας ενός συνόλου δοκιμής είναι η κάλυψη σφαλμάτων που επιτυγχάνεται από το σύνολο δοκιμής. Η κάλυψη σφαλμάτων (ΚΣ) ορίζεται ως ο λόγος του πλήθους των σφαλμάτων που ανιχνεύονται από το σύνολο δοκιμής προς το συνολικό πλήθος σφαλμάτων. Όσο μεγαλύτερη είναι η τιμή της κάλυψης σφαλμάτων τόσο πιο αποτελεσματικό είναι το σύνολο δοκιμής. Ωστόσο ακόμη και εάν ένα σύνολο δοκιμής έχει κάλυψη σφαλμάτων 100%, ανιχνεύει μόνο ελαττώματα που αντιστοιχούν στο συγκεκριμένο μοντέλο σφαλμάτων που στοχεύει το σύνολο δοκιμής. Επομένως, εάν ελέγξουμε την ορθή λειτουργία ενός κυκλώματος με ένα σύνολο δοκιμής που έχει 100% κάλυψη σφαλμάτων, αυτό δεν συνεπάγεται ότι το κύκλωμα δεν μπορεί να έχει κάποιο ελάττωμα.

Σε κάποια κυκλώματα υπάρχουν σφάλματα τα οποία δεν είναι δυνατόν να ανιχνευθούν με κανένα συνδυασμό τιμών εισόδου (τα σφάλματα αυτά οφείλονται σε πλεονασμούς, redundant faults). Επομένως για να έχουμε ένα καλύτερο μέτρο της εκτίμησης της απόδοσης χρησιμοποιούμε ένα άλλο μέτρο, την απόδοση ανίχνευσης σφαλμάτων, η οποία ορίζεται ως ο λόγος του πλήθους των σφαλμάτων που ανιχνεύονται από το σύνολο δοκιμής, προς το πλήθος όλων των σφαλμάτων μείον το πλήθος των σφαλμάτων που δεν μπορούν να ανιχνευθούν.

Όπως έχουμε ήδη αναφέρει στο λογικό έλεγχο το υπό έλεγχο κύκλωμα δέχεται ένα σύνολο διανυσμάτων δοκιμής και παράγει ένα σύνολο αποκρίσεων οι οποίες συγκρίνονται με τις αναμενόμενες αποκρίσεις. Το ερώτημα που τίθεται τώρα είναι ποιος παράγει τα διανύσματα δοκιμής και ποιος συγκρίνει τις αποκρίσεις του υπό έλεγχο κυκλώματος με τις αναμενόμενες αποκρίσεις. Τα διανύσματα δοκιμής μπορεί να έχουν αποθηκευτεί στη μνήμη μιας εξωτερικής συσκευής (Automatic Test Equipment, ATE), η οποία τα στέλνει στο υπό έλεγχο κύκλωμα. Στην εξωτερική συσκευή έχουν αποθηκευτεί επίσης οι αναμενόμενες αποκρίσεις του υπό έλεγχο

κυκλώματος. Η εξωτερική συσκευή τροφοδοτείται με τις αποκρίσεις του υπό έλεγχο κυκλώματος, τις συγκρίνει με τις αναμενόμενες αποκρίσεις και βγάζει το συμπέρασμα σχετικά με τον αν υπάρχει σφάλμα ή όχι στο υπό έλεγχο κύκλωμα.

Τα διανύσματα δοκιμής μπορεί να παράγονται από κύκλωμα που βρίσκεται στο ίδιο ολοκληρωμένο κύκλωμα με τη λειτουργική υπό έλεγχο μονάδα. Επίσης μπορεί να υπάρχει κύκλωμα το οποίο διαβάξει τις αποκρίσεις της υπό έλεγχο μονάδας και αποφασίζει εάν το κύκλωμα έχει σφάλμα ή όχι. Είναι δυνατόν να υπάρχουν στο ίδιο ολοκληρωμένο κύκλωμα μαζί με τη λειτουργική υπό έλεγχο μονάδα κάποια κυκλώματα τα οποία χρησιμοποιούνται για την παραγωγή διανυσμάτων δοκιμής και για την αξιολόγηση της απόκρισης του υπό έλεγχο κυκλώματος, αλλά υπάρχει και εξωτερική συσκευή η οποία τροφοδοτεί τα κυκλώματα, που βρίσκονται πάνω στο ίδιο ολοκληρωμένο με την υπό έλεγχο μονάδα, με πληροφορίες απαραίτητες για την παραγωγή των διανυσμάτων δοκιμής και τη διεξαγωγή του ελέγχου της ορθής λειτουργίας. Σ' αυτή την περίπτωση ο εξωτερικός ελεγκτής είναι μια σχετικά απλή συσκευή.

Οι εξωτερικές συσκευές ελέγχου αποτελούνται από διατάξεις που χρησιμοποιούνται για την προσπέλαση των ακροδεκτών του υπό έλεγχο ολοκληρωμένου κυκλώματος, κυκλώματα για την αποθήκευση των δεδομένων δοκιμής και την οδήγηση των ακροδεκτών του υπό έλεγχο ολοκληρωμένου κυκλώματος καθώς και λογισμικό. Επίσης θα πρέπει να αποτελούνται από επιμέρους ελεγκτές για τον έλεγχο της ορθής λειτουργίας ψηφιακής λογικής, μνημών και αναλογικών κυκλωμάτων. Το κόστος των συσκευών ελέγχου είναι πάρα πολύ μεγάλο. Ως παράδειγμα αναφέρουμε ότι το κόστος μιας τέτοιας ανταγωνιστικής συσκευής το έτος 2000, που λειτουργούσε με συχνότητα 1GHz ήταν 3000 \$ ανά ακροδέκτη που μπορούσε να οδηγεί ανεξάρτητα και για κάθε γεννήτρια συναρτήσεων, η οποία απαιτείται για τον έλεγχο της ορθής λειτουργίας mixed-mode κυκλωμάτων θα απαιτείτο από 0,5 έως 1,5 εκατομμύρια δολάρια.

Στον έλεγχο της ορθής λειτουργίας που λαμβάνει χώρα μετά την κατασκευή ενός ολοκληρωμένου κυκλώματος, το λειτουργικό κύκλωμα προφανώς δεν βρίσκεται σε κατάσταση κανονικής λειτουργίας. Στην περίπτωση όμως του περιοδικού ελέγχου μπορεί να έχουμε μία από τις ακόλουθες περιπτώσεις:

- Το λειτουργικό κύκλωμα είναι εκτός κανονικής λειτουργίας (Off-line testing)
- Η υπό έλεγχο μονάδα βρίσκεται εκτός κανονικής λειτουργίας. Κάποιες άλλες μονάδες όμως του συστήματος μπορεί να βρίσκονται σε κατάσταση κανονικής λειτουργίας (On-line testing)
- Κατά την κανονική λειτουργία μιας μονάδας γίνεται ταυτόχρονα και έλεγχος της ορθής της λειτουργίας (Concurrent testing)
- Αυτοελεγχόμενα κυκλώματα.

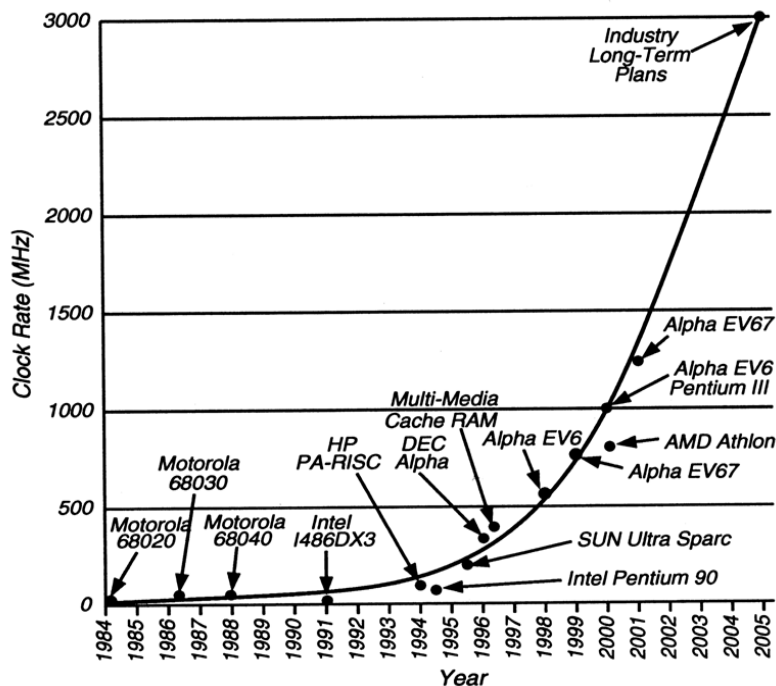
Στα αυτοελεγχόμενα κυκλώματα μια περίπτωση είναι τα διανύσματα εξόδου της μονάδας να είναι κωδικοποιημένα σε κάποιο κώδικα και ένα κύκλωμα να ελέγχει εάν η απόκριση του κυκλώματος είναι κωδική λέξη του κώδικα. Μια άλλη περίπτωση είναι να χρησιμοποιούμε δύο ταυτοτικές μονάδες και να συγκρίνουμε τις αποκρίσεις τους. Το πλεονέκτημα αυτών των τεχνικών είναι ότι ανιχνεύουν επιπλέον των μόνιμων σφαλμάτων, παροδικά και διαλείποντα σφάλματα.

Την εποχή των ολοκληρωμένων κυκλωμάτων μικρής κλίμακας ολοκλήρωσης κάθε ολοκληρωμένο κύκλωμα περιείχε ένα πολύ μικρό αριθμό από λογικές πύλες, οι εισόδοι και έξοδοι των οποίων ήταν άμεσα προσπελάσιμες από τους ακροδέκτες του ολοκληρωμένου κυκλώματος. Επομένως ήταν πάρα πολύ εύκολο να οδηγήσουμε τις εισόδους κάθε πύλης με όλους τους δυνατούς συνδυασμούς εισόδων, να πάρουμε τις

αποκρίσεις και να ελέγξουμε αν είναι οι αναμενόμενες. Σήμερα υπάρχουν ολοκληρωμένα κυκλώματα με πολύ περισσότερες από εκατό κύριες εισόδους. Επομένως, αν και η συχνότητα λειτουργίας έχει αυξηθεί πάρα πολύ, είναι χρονικά αδύνατον να τα τροφοδοτήσουμε με όλους τους δυνατούς συνδυασμούς διανυσμάτων εισόδου. Άρα μόνο σε κυκλώματα με σχετικά μικρό αριθμό ακροδεκτών εισόδου είναι δυνατόν να εφαρμόσουμε εξαντλητικό έλεγχο ορθής λειτουργίας.

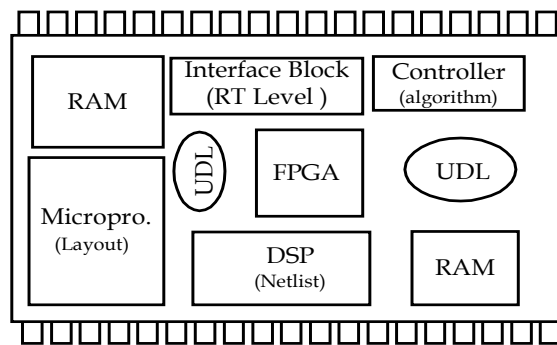
Όπως γνωρίζουμε στα ακολουθιακά κυκλώματα η τιμή των εξόδων εξαρτάται από την τιμή των εισόδων και την εσωτερική κατάσταση στην οποία βρίσκεται το κύκλωμα. Η εσωτερική κατάσταση αποθηκεύεται σε έναν καταχωρητή, ο οποίος καλείται καταχωρητής εσωτερικής κατάστασης. Το περιεχόμενο του καταχωρητή εσωτερικής κατάστασης δεν είναι άμεσα ελέγξιμο από τις κύριες εισόδους του κυκλώματος αλλά ούτε και άμεσα παρατηρήσιμο από τις κύριες εξόδους του. Επομένως η κλασική εφαρμογή του λογικού ελέγχου που βασίζεται στην εφαρμογή ενός διανύσματος δοκιμής, την παρατήρηση της απόκρισης και τη σύγκρισή της με την αναμενόμενη απόκριση, δεν είναι εφικτή. Θα δούμε αργότερα ότι το πρόβλημα αυτό λύνεται σχεδιαστικά.

Όπως ήδη έχουμε αναφέρει στον δομικό έλεγχο ελέγχουμε την ορθή λειτουργία κάθε συνιστώσας του κυκλώματος. Αυτό απαιτεί κάθε συνιστώσα να είναι ελέγξιμη και παρατηρήσιμη από τις κύριες εισόδους και τις κύριες εξόδους του ολοκληρωμένου κυκλώματος αντίστοιχα. Όταν ο λόγος του πλήθους των πυλών προς το πλήθος των ακροδεκτών εισόδου/εξόδου ενός κυκλώματος είναι πολύ μεγάλος, τότε υπάρχουν συνιστώσες που είναι εμφωλευμένες βαθιά μέσα στο ολοκληρωμένο κύκλωμα με αποτέλεσμα να είναι δύσκολο να ελέγξουμε τις τιμές των εισόδων τους από τις κύριες εισόδους του ολοκληρωμένου κυκλώματος. Για τον ίδιο λόγο είναι δύσκολο να παρατηρήσουμε τις τιμές των εξόδων κάποιων συνιστωσών του ολοκληρωμένου κυκλώματος παρατηρώντας τις κύριες εξόδους του.



Σχήμα 1.6 Η εξέλιξη της συχνότητας λειτουργίας των μικροεπεξεργαστών μέχρι το 2000

Στο παραπάνω σχήμα φαίνεται η εξέλιξη της συχνότητας λειτουργίας των μικροεπεξεργαστών μέχρι το 2000 καθώς και οι προβλέψεις που γίνονταν τότε για το μέλλον. Παρατηρούμε ότι η συχνότητα 3 GHz αναφέρεται ως μακροπρόθεσμο σχέδιο της βιομηχανίας. Η πραγματικότητα ξεπέρασε τις προσδοκίες. Από το σχήμα αυτό μπορούμε να δούμε την πολύ μεγάλη αύξηση της συχνότητας λειτουργίας των επεξεργαστών τα τελευταία χρόνια, που συνεπάγεται μεγάλες απαιτήσεις για τον έλεγχο της ορθής λειτουργίας των κυκλωμάτων στην συχνότητα λειτουργία τους (at speed testing).



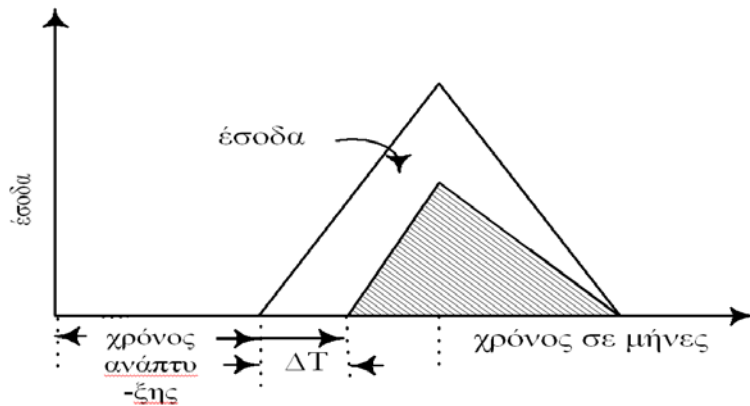
Σχήμα 1.7 Παράδειγμα συστήματος το οποίο είναι υλοποιημένο σε ένα OK, SOC.

Στο σχήμα βλέπουμε ένα σύστημα που έχει κατασκευαστεί σε ένα ολοκληρωμένο κύκλωμα. Κάποιες από τις μονάδες του συστήματος μπορεί να έχουν σχεδιαστεί από άλλη εταιρεία και ο σχεδιαστής του συστήματος να μην γνωρίζει την πλήρη εσωτερική τους δομή, παρά μόνον τη λειτουργία τους και τα σήματα εισόδου/εξόδου για τη διασύνδεσή τους με άλλες μονάδες του συστήματος. Σ' αυτή την περίπτωση οι σχεδιαστές των μονάδων τροφοδοτούν το σχεδιαστή του συστήματος και με τα σύνολα δοκιμής των μονάδων καθώς και με τη διαδικασία που πρέπει να ακολουθηθεί για να ελεγχθεί η ορθή τους λειτουργία μετά την κατασκευή.

Κάποιες από αυτές τις μονάδες όμως μπορεί να είναι εμφωλευμένες μέσα στο ολοκληρωμένο κύκλωμα, δηλαδή οι κύριες εισοδοί τους και οι κύριες εξοδοί τους να μην είναι άμεσα προσπελάσιμες από τις κύριες εισόδους και εξόδους του ολοκληρωμένου κυκλώματος. Το γεγονός ότι οι εισοδοί και εξοδοί κάποιων μονάδων του συστήματος δεν είναι άμεσα προσπελάσιμες από τις κύριες εισόδους και εξόδους του ολοκληρωμένου κυκλώματος και το γεγονός ότι για τον έλεγχο της ορθής λειτουργίας κάποιων μονάδων του συστήματος πρέπει να χρησιμοποιηθούν σύνολα δοκιμής και διαδικασίες ελέγχου ορθής λειτουργίας, οι οποίες δεν επιλέχτηκαν από τον σχεδιαστή του συστήματος, αποτελούν νέες προκλήσεις στον έλεγχο της ορθής λειτουργίας συστημάτων που υλοποιούνται σε ένα ολοκληρωμένο κύκλωμα.

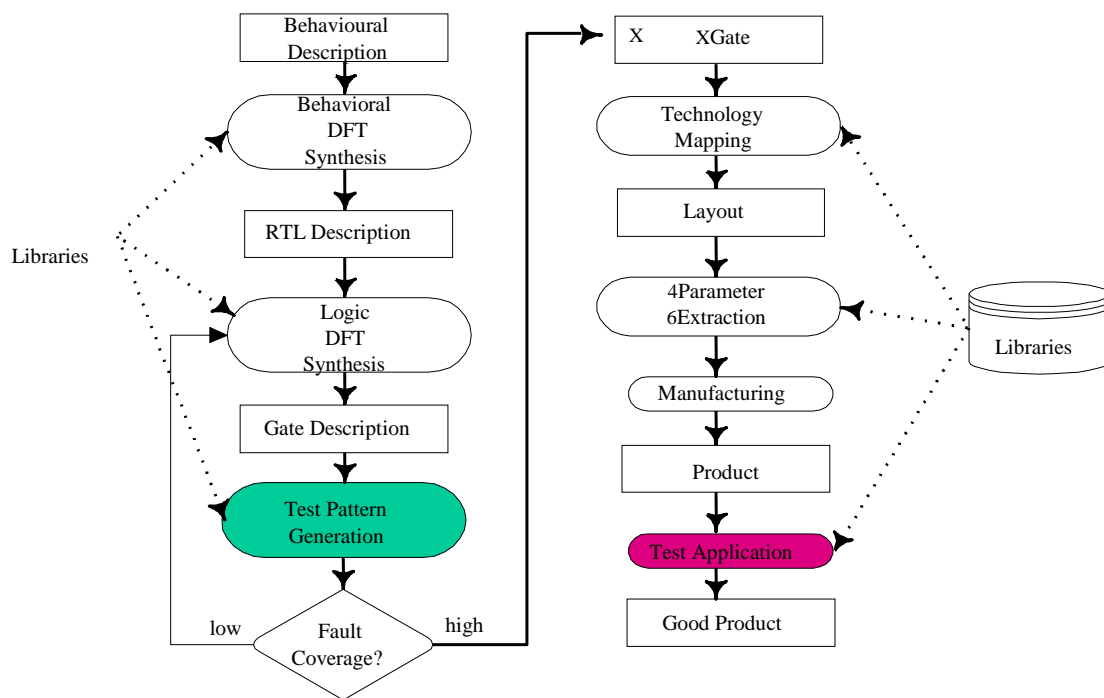
Στο σχήμα 1.8 βλέπουμε τα έσοδα (επιφάνεια του τριγώνου) που θα έχει μια εταιρεία όταν ένα προϊόν βγει στην αγορά κάποια χρονική στιγμή T και τα έσοδα που θα έχει εάν βγει ΔT χρόνο αργότερα. Είναι προφανές ότι υπάρχει μείωση των κερδών. Είναι δυνατόν ο σχεδιαστής να δέχεται πίεση από τον υπεύθυνο της παραγωγής ώστε το προϊόν να βγει στην ώρα του στην παραγωγή. Κάτω από την πίεση αυτή ο σχεδιαστής είναι δυνατόν να αφήσει, την ανάπτυξη της διαδικασίας ελέγχου ορθής λειτουργίας της μονάδας που σχεδιάζει για μετά το τέλος της

σχεδιαστικής διαδικασίας. Τότε όμως αν διαπιστώσει ότι είναι δύσκολος ο έλεγχος της ορθής λειτουργίας της μονάδας που σχεδίασε, θα πρέπει να κάνει αλλαγές οι οποίες θα οδηγήσουν σε μεγαλύτερες καθυστερήσεις. Επομένως ο σχεδιαστής θα πρέπει να λάβει υπόψη του από τα πρώτα βήματα της σχεδιαστικής διαδικασίας ότι θα πρέπει να ελέγχεται εύκολα η ορθή λειτουργία της μονάδας που σχεδιάζει.



Σχήμα 1.8 Χρόνος παραγωγής ενός συστήματος.

Η πολυπλοκότητα της διαδικασίας εξαγωγής διανυσμάτων δοκιμής και η μειωμένη προσπέλαση των μονάδων ενός συστήματος που σχεδιάζεται σε ένα ολοκληρωμένο κύκλωμα, ώθησαν στην ανάπτυξη σχεδιαστικών προσεγγίσεων που οδηγούν σε εύκολο έλεγχο της ορθής λειτουργίας του συστήματος.



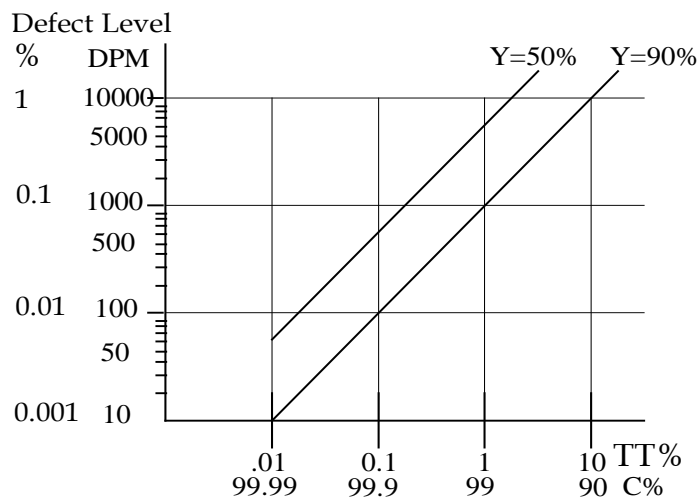
Σχήμα 1.9 Κύκλος σχεδίασης.

Στο σχήμα 1.9 βλέπουμε καθαρά ότι διεξαγωγή διανυσμάτων δοκιμής γίνεται κατά τη φάση σχεδίασης του ΟΚ/συστήματος, ενώ η χρησιμοποίηση του συνόλου δοκιμής, δηλαδή ο έλεγχος της ορθής λειτουργίας της, γίνεται μετά την κατασκευή κάθε ΟΚ/συστήματος.

Η απόδοση της γραμμής παραγωγής (yield) ορίζεται ως ο λόγος του πλήθους των ολοκληρωμένων κυκλωμάτων που πέρασαν όλους τους ελέγχους προς το πλήθος όλων των ολοκληρωμένων κυκλωμάτων που ελέγχθηκε η ορθή τους λειτουργία. Η εταιρεία κατασκευής ολοκληρωμένων κυκλωμάτων επιθυμεί η απόδοση της γραμμής παραγωγής να είναι όσο το δυνατόν πιο κοντά στο 100%.

Κάποια ελαττωματικά ολοκληρωμένα κυκλώματα περνούν τους ελέγχους ορθής λειτουργίας διότι το ελάττωμα που έχουν δεν αντιστοιχεί σε κάποιο σφάλμα του μοντέλου σφαλμάτων για το οποίο εξήχθη το σύνολο δοκιμής. Επομένως η πραγματική τιμή της απόδοσης της γραμμής παραγωγής δίνεται από τον λόγο του πλήθους των μη ελαττωματικών ολοκληρωμένων κυκλωμάτων προς το πλήθος όλων των ολοκληρωμένων. Ωστόσο ο υπολογισμός αυτής της τιμής είναι πολύ δύσκολος, λόγω της έλλειψης πλήρους πληροφορίας από τη στιγμή που τα ολοκληρωμένα κυκλώματα πωλούνται.

Το ποσοστό διαφυγής ελαττωματικών ολοκληρωμένων κυκλωμάτων (defect level) δίνει το ποσοστό των ελαττωματικών ολοκληρωμένων κυκλωμάτων που πέρασαν τον έλεγχο ορθής λειτουργίας και μετριέται σε πλήθος ελαττωματικών ολοκληρωμένων κυκλωμάτων ανά εκατομμύριο.

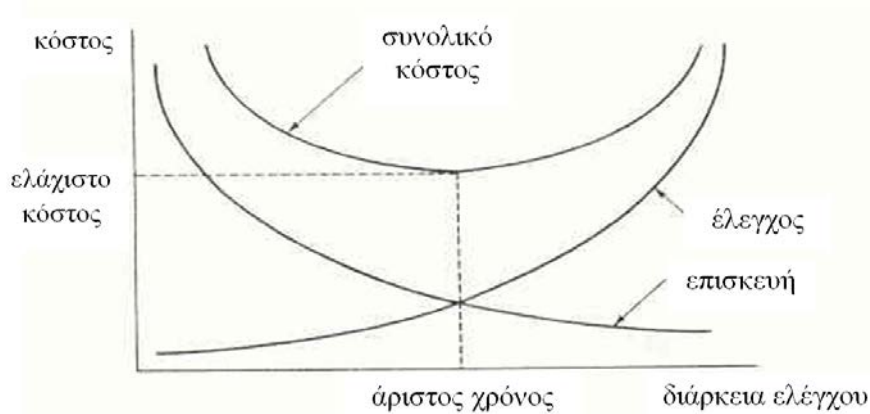


Σχήμα 1.10 Απόδοση γραμμής παραγωγής & ποσοστό ελαττωματικών ολοκληρωμένων κυκλωμάτων.

Στο σχήμα 1.10 βλέπουμε εμπειρικές καμπύλες που δείχνουν τη σχέση μεταξύ της απόδοσης της γραμμής παραγωγής, του ποσοστού των ελαττωματικών ολοκληρωμένων κυκλωμάτων και της κάλυψης σφαλμάτων. Παρατηρούμε ότι όσο μικρότερη είναι η απόδοση της γραμμής παραγωγής, τόσο μεγαλύτερη κάλυψη σφαλμάτων πρέπει να προσφέρει το σύνολο δοκιμής για να πετύχουμε το ίδιο ποσοστό διαφυγής ελαττωματικών ολοκληρωμένων κυκλωμάτων.

Το σχήμα 1.11 δείχνει τη σχέση μεταξύ του κόστους ελέγχου της ορθής λειτουργίας και του κόστους επισκευής. Ένα ολοκληρωμένο κύκλωμα, του οποίου

δεν ελέγχεται η ορθή λειτουργία μετά την κατασκευή του, θα έχει στο πεδίο της εφαρμογής ένα ρυθμό βλαβών ίσο με αυτόν της διαδικασίας κατασκευής. Το κόστος του ελέγχου ορθής κατασκευής είναι μηδενικό, αλλά το κόστος αντικατάστασης στο πεδίο της εφαρμογής λαμβάνει τη μέγιστη τιμή.



Σχήμα 1.11 Σχέση κόστους ελέγχου ορθής λειτουργίας και επισκευής.

Καθώς ο έλεγχος της ορθής λειτουργίας γίνεται πιο λεπτομερής, το πλήθος των βλαβών στο πεδίο της εφαρμογής ελαττώνεται. Ένας πιο προσεκτικός έλεγχος ορθής λειτουργίας ελαττώνει το ποσοστό των ελαττωματικών ολοκληρωμένων κυκλωμάτων που διαφεύγουν κατά τη διαδικασία ελέγχου ορθής λειτουργίας, μειώνοντας έτσι το κόστος επισκευής, αλλά αυξάνοντας το κόστος ελέγχου ορθής λειτουργίας. Η άριστη διάρκεια του ελέγχου ορθής λειτουργίας είναι αυτή που ελαχιστοποιεί το συνολικό κόστος.

Μέρος Α

Μοντέλα σφαλμάτων

Εξομοίωση σφαλμάτων

Εξαγωγή διανυσμάτων δοκιμής

Κεφάλαιο 2. Βλάβες, ελαττώματα και μοντέλα σφαλμάτων

Οι όροι βλάβες, ελαττώματα και σφάλματα μερικές φορές χρησιμοποιούνται κατά ένα συγκεκριμένο τρόπο στη βιβλιογραφία του ελέγχου ορθής λειτουργίας. Θα χρησιμοποιούμε τον όρο βλάβη για να δηλώνουμε ότι το σύστημα δεν λειτουργεί σύμφωνα με τις προδιαγραφές του. Αιτία της βλάβης είναι κάποιο ελάττωμα το οποίο μοντελοποιείται με κάποιο σφάλμα.

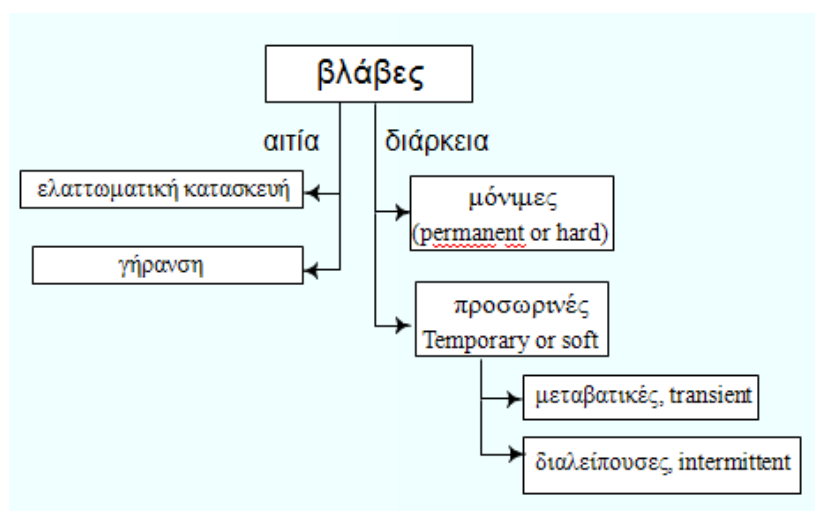
Μπορούμε να ταξινομήσουμε τις βλάβες ανάλογα της αιτίας και ανάλογα της διάρκειας τους.

Ταξινόμηση βλαβών ανάλογα της αιτίας που τις προκάλεσε:

- Βλάβες που προκλήθηκαν κατά τη διαδικασία κατασκευής του ολοκληρωμένου κυκλώματος
 - » κομμένες γραμμές
 - » βραχυκυκλώματα
 - » κλπ
- Βλάβες λόγω γήρανσης
 - » *hot carriers degradation*
 - » *oxide breakdown*
 - » *Electromigration*
 - » κλπ

Ταξινόμηση βλαβών ανάλογα της διάρκειας:

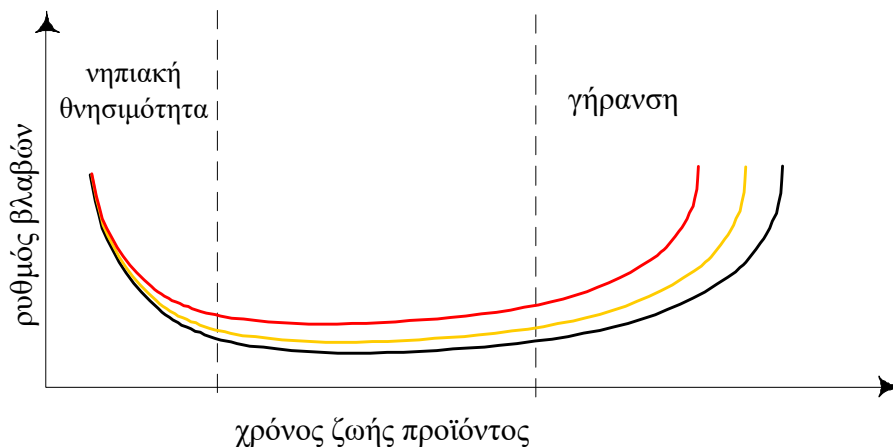
- Μόνιμες βλάβες
 - » κομμένες γραμμές
 - » βραχυκυκλώματα
 - » κλπ
- Προσωρινές βλάβες (*temporary*)
 - » μεταβατικές βλάβες (σφάλματα λόγω ακτινοβολίας, θορύβου ή διακυμάνσεων της στάσης τροφοδοσίας)
 - » διαλείπουσες (*Intermittent*) βλάβες (*caused by marginal design parameters such as timing problems that result from races, hazards, skew, or signal integrity problems caused by crosstalk, ground bounce, and other factors*)



Σχήμα 2.1 Ταξινόμηση βλαβών.

Στο σχήμα βλέπουμε την ταξινόμηση των βλαβών τόσο βάσει της αιτίας τους όσο και βάσει της διάρκειάς τους. Μεταβατικές βλάβες προκαλούνται τόσο από εσωτερικές όσο και εξωτερικές αιτίες. Διαλείπουσες βλάβες προκαλούνται από την υποβάθμιση της λειτουργίας των συνιστωσών του κυκλώματος. Για παράδειγμα η είσοδος ηλεκτρονίων στο διηλεκτρικό της πύλης ενός FET (channel hot electron, CHE, injection) προκαλεί υποβάθμιση των χαρακτηριστικών του τρανζίστορ που τελικά μπορεί να οδηγήσουν σε βλάβη.

Στο σχήμα 2.2 βλέπουμε το ρυθμό βλαβών ως συνάρτηση του χρόνου ζωής ενός ολοκληρωμένου κυκλώματος. Παρατηρούμε ότι αρχικά υπάρχει ένας αυξημένος ρυθμός βλαβών. Ο αυξημένος ρυθμός βλαβών την πρώτη περίοδο χρησιμοποίησης των ολοκληρωμένων κυκλωμάτων οφείλεται στα ελαττώματα τα οποία διέφυγαν την ανίχνευση κατά τον έλεγχο της ορθής λειτουργίας μετά την κατασκευή των ολοκληρωμένων κυκλωμάτων. Με την πάροδο του χρόνου τα ελαττωματικά ολοκληρωμένα κυκλώματα απομακρύνονται (αντικαθίστανται) και έχουμε μια περίοδο όπου ο ρυθμός βλαβών είναι μικρός. Μετά την πάροδο κάποιου χρονικού διαστήματος αρχίζει να εμφανίζεται και πάλι μια αύξηση του ρυθμού βλαβών η οποία οφείλεται στη γήρανση των στοιχείων του κυκλώματος.



Σχήμα 2.2 Ταξινόμηση βλαβών.

Στο σχήμα 2.2 υπάρχουν τρεις καμπύλες εκ των οποίων η καμπύλη με κόκκινο χρώμα αντιστοιχεί σε μικρότερες διαστάσεις (σε μικρότερη τιμή του “minimum feature size”). Παρατηρούμε ότι, καθώς χρησιμοποιούμε πιο νέες τεχνολογίες με μικρότερες διαστάσεις για την υλοποίηση των ολοκληρωμένων κυκλωμάτων, έχουμε μια αύξηση του ρυθμού βλαβών των ολοκληρωμένων κυκλωμάτων στο πεδίο της εφαρμογής. Αυτό οφείλεται στο γεγονός ότι τα ολοκληρωμένα κυκλώματα που υλοποιούνται με νέες τεχνολογίες μικρότερων διαστάσεων είναι πιο ευαίσθητα σε παροδικά σφάλματα.

Τα είδη ελαττωμάτων που μπορούν να προκύψουν κατά την κατασκευή ενός ολοκληρωμένου κυκλώματος είναι πάρα πολλά. Ενδεικτικά αναφέρουμε μερικά από αυτά. Λίγο επιπλέον μέταλλο είναι δυνατόν να προκαλέσει βραχυκύκλωμα με μια γειτονική γραμμή ενώ η απουσία μετάλλου είναι δυνατόν να οδηγήσει σε κόψιμο μιας γραμμής διασύνδεσης. Αν υπάρξει μεγάλο ηλεκτρικό πεδίο είναι δυνατόν να

έχουμε διάτρηση του οξειδίου της πύλης (oxide breakdown). Το πλάτος των γραμμών διασύνδεσης σ' ένα ολοκληρωμένο κύκλωμα έχει διακυμάνσεις. Στα στενώματα η πυκνότητα ρεύματος είναι μεγαλύτερη. Όταν η πυκνότητα ρεύματος είναι αρκετά μεγάλη και υπάρχουν και κάποιες άλλες συνθήκες, όπως υψηλή θερμοκρασία, τότε υλικό της γραμμής διασύνδεσης απομακρύνεται από τα σημεία όπου η πυκνότητα ρεύματος είναι πολύ μεγάλη. Αυτό έχει συνέπεια σε κάποια σημεία η γραμμή διασύνδεσης να γίνεται συνεχώς και πιο λεπτή, έως ότου τελικά κοπεί.



Σχήμα 2.3 Ταξινόμηση βλαβών.

Στο σχήμα 2.3 βλέπουμε τρία παραδείγματα ελαττωμάτων: (a) κόψιμο γραμμής, (b) βραχυκύκλωμα μεταξύ δύο γραμμών που βρίσκονται στο ίδιο επίπεδο και (c) βραχυκύκλωμα μεταξύ δύο γραμμών δύο γειτονικών επιπέδων

Στη συνέχεια θα μιλήσουμε για διάφορα μοντέλα σφαλμάτων.

Απλά σφάλματα μόνιμης τιμής (Single Stuck-at Faults, SSA):

Μία γραμμή παίρνει μόνιμα τη λογική τιμή 0 ή 1. Κάθε φορά στο κύκλωμα θεωρούμε ότι υπάρχει μόνο ένα σφάλμα απλής μόνιμης τιμής. Σύμφωνα με αυτό το μοντέλο σε ένα σημείο διακλάδωσης πρέπει να θεωρήσουμε διαφορετικό το απλό σφάλμα μόνιμης τιμής στο μίσχο και το απλό σφάλμα μόνιμης τιμής σε κάποια γραμμή διακλάδωσης. Επίσης πρέπει να θεωρήσουμε ως διαφορετικά τα απλά σφάλματα μόνιμης τιμής στις γραμμές διακλάδωσης του σημείου διακλάδωσης.

Πολλαπλά σφάλματα μόνιμης τιμής (Multiple Stuck-at Faults, SSA):

Δύο ή περισσότερες γραμμές του κυκλώματος έχουν σταθερές τιμές, όχι αναγκαία τις ίδιες.

Σφάλματα βραχυκυκλώματος (Bridging Faults):

Δύο ή περισσότερες γραμμές, οι οποίες κανονικά είναι ανεξάρτητες, συνδέονται ηλεκτρικά.

Τρανζίστορ που μόνιμα δεν άγει (Stuck-Open Faults, SOP)

Τρανζίστορ που άγει μόνιμα (Stuck-On Faults, SON)

Σφάλματα καθυστέρησης:

Τα σφάλματα αυτά έχουν ως συνέπεια αύξηση της καθυστέρησης του κυκλώματος

Σύμφωνα με το μοντέλο των απλών σφαλμάτων μόνιμης τιμής μια γραμμή του κυκλώματος μπορεί να έχει μόνιμα τη λογική τιμή 0 ή 1. Επομένως σε ένα κύκλωμα με k γραμμές είναι δυνατόν να έχουμε 2^k περιπτώσεις απλών σφαλμάτων μόνιμης τιμής. Στην περίπτωση πύλης AND δύο εισόδων και μιας εξόδου είναι δυνατόν να έχουμε έξι διαφορετικά απλά σφάλματα μόνιμης τιμής. Ο συμβολισμός $A/0$ χρησιμοποιείται για να δηλώσει ότι η γραμμή A έχει μόνιμα τη λογική τιμή 0.

Στο σχήμα 1.4 δίνονται οι τιμές των εισόδων της πύλης AND, η απόκριση της πύλης όταν δεν υπάρχει κάποιο σφάλμα και η απόκριση της πύλης στην περίπτωση κάθε ενός από τα έξι απλά σφάλματα μόνιμης τιμής. Όπως έχουμε ήδη αναφέρει, για να ανιχνευθεί ένα σφάλμα από ένα διάνυσμα εισόδου, θα πρέπει η τιμή εξόδου του κυκλώματος όταν υπάρχει το σφάλμα να είναι διαφορετική από την τιμή εξόδου του κυκλώματος όταν δεν υπάρχει το σφάλμα. Παρατηρούμε λοιπόν ότι τα σφάλματα $A/0$, $B/0$ και $Z/0$ ανιχνεύονται από το διάνυσμα δοκιμής $AB=11$ διότι η απόκριση της πύλης με κάποιο από αυτά τα σφάλματα είναι διαφορετική της απόκριση χωρίς σφάλμα. Παρατηρούμε επίσης ότι το σφάλμα $A/1$ ανιχνεύεται από το διάνυσμα δοκιμής $AB=01$ και το σφάλμα $B/1$ ανιχνεύεται από το διάνυσμα δοκιμής $AB=10$. Το σφάλμα $Z/1$ ανιχνεύεται από οποιοδήποτε από τα διανύσματα δοκιμής $AB=00$ ή 01 ή 10 . Επομένως οποιοδήποτε από τα έξι απλά σφάλματα μόνιμης τιμής ανιχνεύεται από το σύνολο δοκιμής που αποτελείται από τρία διανύσματα δοκιμής, τα διανύσματα 01 , 10 και 11 .

Στη γενική περίπτωση το σύνολο δοκιμής μιας πύλης AND με k εισόδους και μια έξοδο αποτελείται $k+1$ διανύσματα, τα διανύσματα:

11...110
 11...101
 11...011
 .
 .
 .
 10...111
 01...111
 11...111

Τώρα ως παράδειγμα έχουμε μια πύλη OR, η οποία είναι δύο εισόδων και μιας εξόδου, επομένως είναι δυνατόν να έχουμε έξι διαφορετικά απλά σφάλματα μόνιμης τιμής.

Στο σχήμα 1.5 δίνονται οι τιμές των εισόδων της πύλης OR, η απόκριση της πύλης όταν δεν υπάρχει κάποιο σφάλμα και η απόκριση της πύλης στην περίπτωση κάθε ενός από τα έξι απλά σφάλματα μόνιμης τιμής. Παρατηρούμε λοιπόν ότι τα σφάλματα $A/1$, $B/1$ και $Z/1$ ανιχνεύονται από το διάνυσμα δοκιμής $AB=00$, διότι η απόκριση της πύλης με κάποιο από αυτά τα σφάλματα είναι διαφορετική της απόκρισης χωρίς σφάλμα. Παρατηρούμε επίσης ότι το σφάλμα $A/0$ ανιχνεύεται από το διάνυσμα δοκιμής $AB=10$ και το σφάλμα $B/0$ ανιχνεύεται από το διάνυσμα δοκιμής $AB=01$. Το σφάλμα $Z/0$ ανιχνεύεται από οποιοδήποτε από τα διανύσματα δοκιμής $AB=01$ ή 10 ή 11 . Επομένως οποιοδήποτε από τα έξι απλά σφάλματα μόνιμης τιμής ανιχνεύεται από το σύνολο δοκιμής που αποτελείται από τρία διανύσματα δοκιμής, τα διανύσματα 00 , 01 και 10 .

Στη γενική περίπτωση το σύνολο δοκιμής μιας πύλης OR με k εισόδους και μια έξοδο αποτελείται $k+1$ διανύσματα, τα διανύσματα:

00...001
 00...010

00...100
 .
 .
 .
 01...000
 10...000
 00...000

Τα απλά σφάλματα μόνιμης τιμής έχουν κάποιες ιδιότητες οι οποίες, όπως θα δούμε, είναι χρήσιμες διότι μας οδηγούν στη μείωση των σφαλμάτων για τα οποία πρέπει να εξάγουμε διανύσματα δοκιμής. Θεωρήστε τα σφάλματα φ1 και φ2 με σύνολα δοκιμής T1 και T2 αντίστοιχα.

Ισοδυναμία (Equivalence)

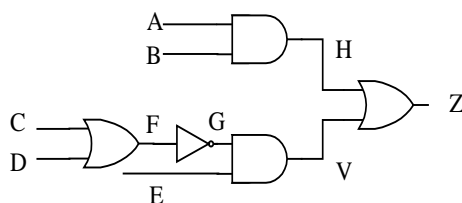
Το σφάλμα φ1 είναι ισοδύναμο του σφάλματος φ2 εάν $T1 = T2$, δηλαδή, κάθε διάνυσμα δοκιμής το οποίο ανιχνεύει το φ1 ανιχνεύει επίσης το φ2 και αντίστροφα.

Η σχέση ισοδυναμίας συνεπάγεται ότι από όλα τα σφάλματα τα οποία είναι ισοδύναμα αρκεί να επιλέξουμε ένα και να εξάγουμε διάνυσμα δοκιμής γι' αυτό το σφάλμα. Αυτό το διάνυσμα δοκιμής θα ανιχνεύει και όλα τα υπόλοιπα σφάλματα που ανήκουν στην ίδια κλάση ισοδυναμίας.

Κυριαρχία (Dominance)

Το σφάλμα φ1 κυριαρχεί του σφάλματος φ2 εάν το σύνολο δοκιμής T2 είναι υποσύνολο του συνόλου δοκιμής T1. Σ' αυτή την περίπτωση ένα διάνυσμα δοκιμής το οποίο ανιχνεύει το σφάλμα φ2 θα ανιχνεύει επίσης το σφάλμα φ1. Η σχέση δεν είναι συμμετρική.

Η σχέση κυριαρχίας συνεπάγεται ότι εάν το σφάλμα φ1 κυριαρχεί του σφάλματος φ2 τότε μπορούμε να επιλέξουμε το φ2 για να εξάγουμε διάνυσμα δοκιμής. Το διάνυσμα δοκιμής το οποίο ανιχνεύει το σφάλμα φ2 θα ανιχνεύει και το σφάλμα φ1. Θα δούμε αργότερα ότι πρέπει να είμαστε προσεκτικοί στην χρήση της σχέσης κυριαρχίας. Μπορούμε να τη χρησιμοποιούμε χωρίς κανένα πρόβλημα όταν το κύκλωμα μας δεν έχει πλεονασμούς. Ωστόσο, εάν δεν γνωρίζουμε εάν το κύκλωμά μας έχει πλεονασμούς, δεν πρέπει να τη χρησιμοποιούμε, διότι μπορεί να οδηγήσει σε μείωση του ποσοστού κάλυψης σφαλμάτων.



Σχήμα 2.4 Ένα κύκλωμα χωρίς πλεονασμούς.

Το λογικό κύκλωμα του σχήματος 2.4 είναι προφανές ότι δεν έχει πλεονασμούς. Στο κύκλωμα αυτό θα εφαρμόσουμε τις σχέσεις ισοδυναμίας και κυριαρχίας μεταξύ των σφαλμάτων απλής μόνιμης τιμής ώστε να καταλήξουμε στον ελάχιστο αριθμό σφαλμάτων απλής μόνιμης τιμής για τα οποία θα πρέπει να εξάγουμε διανύσματα δοκιμής.

Για κάθε μια πύλη του κυκλώματος καταγράφουμε τα απλά σφάλματα μόνιμης τιμής τα οποία είναι ισοδύναμα. Έτσι προκύπτουν και έξι σχέσεις ισοδυναμίας:

$$\begin{aligned} & \{A/0, B/0, H/0\} \\ & \{C/1, D/1, F/1\} \\ & \{F/1, G/0\} \\ & \{E/0, G/0, V/0\} \\ & \{H/1, V/1, Z/1\} \\ & \{F/0, G/1\} \end{aligned}$$

Από τις σχέσεις ισοδυναμίας που έχουμε καταγράψει παρατηρούμε ότι η υπ' αριθμόν 2, 3, και 4 μπορούν να ενωθούν σε μία κλάση ισοδυναμίας, την οποία στη συνέχεια δηλώνουμε ως 2,3,4. Επομένως τώρα έχουμε τις κλάσεις ισοδυναμίας:

$$\begin{aligned} 1 & \quad \{A/0, B/0, H/0\} \\ 2,3,4 & \quad \{C/1, D/1, F/1, G/0, E/0, V/0\} \\ 5 & \quad \{H/1, V/1, Z/1\} \\ 6 & \quad \{F/0, G/1\} \end{aligned}$$

Στη συνέχεια ο συμβολισμός $\varphi1 \rightarrow \varphi2$ θα δηλώνει ότι το σφάλμα $\varphi1$ κυριαρχείται από το σφάλμα $\varphi2$.

$$\begin{aligned} 7 & \quad A/1 \rightarrow H/1, \text{ έτσι το } A/1 \text{ μπορεί να αντιπροσωπεύσει το } H/1 \text{ και όλα τα ισοδύναμά} \\ & \quad \text{του σφάλματα στην κλάση ισοδυναμίας 5} \\ 8 & \quad B/1 \rightarrow H/1 \\ 9 & \quad C/0 \rightarrow F/0, \text{ έτσι το } C/0 \text{ μπορεί να αντιπροσωπεύσει το } F/0 \text{ και όλα τα ισοδύναμά} \\ & \quad \text{του σφάλματα στην κλάση ισοδυναμίας 6} \\ 10 & \quad D/0 \rightarrow F/0 \\ 11 & \quad E/1 \rightarrow V/1 \\ 12 & \quad G/1 \rightarrow V/1 \\ 13 & \quad V/0 \rightarrow Z/0, \text{ αλλά το } V/0 \text{ ανήκει στην κλάση ισοδυναμίας 2,3,4. Οποιοδήποτε} \\ & \quad \text{σφάλμα από αυτή την κλάση ισοδυναμίας κυριαρχείται από το σφάλμα } Z/0. \end{aligned}$$

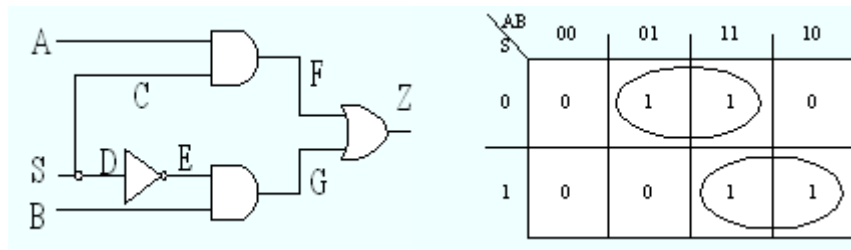
Επομένως αν εξάγουμε ένα σύνολο δοκιμής για το σύνολο απλών σφαλμάτων μόνιμης τιμής $\{A/0, A/1, B/1, C/0, C/1, D/0, E/1\}$ αυτό θα ανιχνεύει και όλα τα υπόλοιπα απλά σφάλματα μόνιμης τιμής του κυκλώματος.

Σ' ένα κύκλωμα, το οποίο δεν έχει κόμβους διακλάδωσης, οποιοδήποτε σύνολο δοκιμής ανιχνεύει τα απλά σφάλματα μόνιμης τιμής στις εισόδους του κυκλώματος, ανιχνεύει επίσης οποιοδήποτε σφάλμα απλής μόνιμης τιμής σε οποιοδήποτε κόμβο του κυκλώματος. Αυτό μπορεί να αποδειχθεί εύκολα, χρησιμοποιώντας τις σχέσεις ισοδυναμίας και κυριαρχίας. Ως παράδειγμα αναφέρουμε ότι στο σύνολο των σφαλμάτων απλής μόνιμης τιμής που καταλήξαμε στο προηγούμενο παράδειγμα ανήκουν μόνο σφάλματα στις εισόδους του κυκλώματος.

Σ' ένα συνδυαστικό κύκλωμα, το οποίο δεν έχει πλεονασμούς, ένα σύνολο δοκιμής το οποίο ανιχνεύει οποιοδήποτε απλό σφάλμα μόνιμης τιμής σε κάποια από τις κύριες εισόδους του κυκλώματος και σε οποιαδήποτε γραμμή διακλάδωσης, ανιχνεύει επίσης κάθε απλό σφάλμα μόνιμης τιμής σε οποιοδήποτε άλλο κόμβο του κυκλώματος.

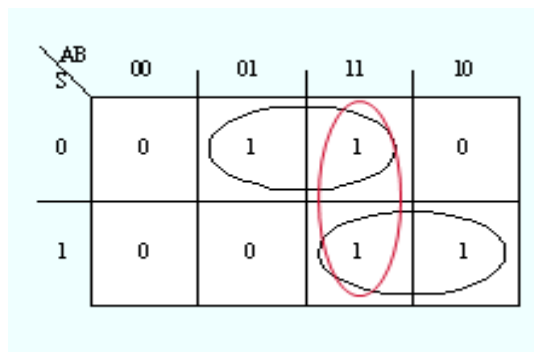
Ένα κύκλωμα χωρίς πλεονασμό είναι η υλοποίηση μιας λογικής συνάντησης που δεν περιέχει πλεονασματικές μεταβλητές ή όρους. Σε επίπεδο λογικού κυκλώματος πλεονασματικές μεταβλητές συνεπάγονται την ύπαρξη πλεονασματικών τρανζίστορ ενώ πλεονασματικοί όροι συνεπάγονται την ύπαρξη πλεονασματικών πυλών.

Γενικά ο πλεονασμός πρέπει να αποφεύγεται διότι έχει ως συνέπεια την αύξηση του υλικού και της κατανάλωσης ενέργειας του κυκλώματος. Επίσης οδηγεί σε αύξηση του χρόνου που απαιτείται για την εξαγωγή του συνόλου δοκιμής του κυκλώματος. Θα δούμε όμως ότι υπάρχουν περιπτώσεις που βάζουμε πλεονασμό για να λύσουμε κάποια προβλήματα.



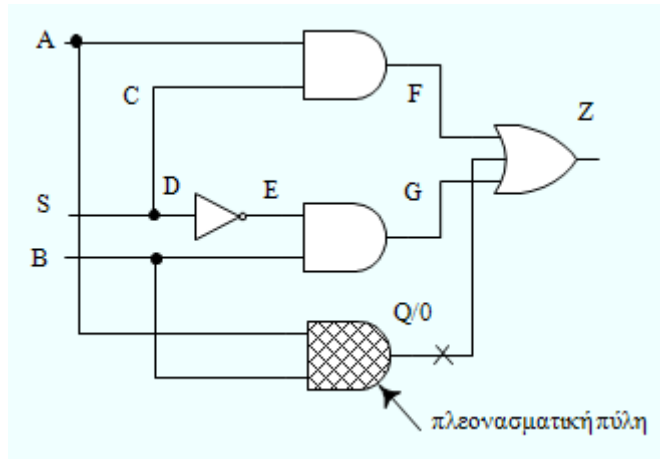
Σχήμα 2.5 Το λογικό κύκλωμα ενός πολυπλέκτη και ο πίνακας Karnaugh.

Στο σχήμα 2.5 όταν τόσο η είσοδος A όσο και η είσοδος B οδηγούνται με τη λογική τιμή 1, τότε κατά τη μετάβαση του σήματος ελέγχου S από 1 σε 0 στην έξοδο του πολυπλέκτη εμφανίζεται μια αρνητική αιχμή, static hazard (η έξοδος ήταν στην λογική τιμή 1 μεταβαίνει προσωρινά στη λογική 0 για να ξαναπάρει την τελική της τιμή που είναι η λογική τιμή 1). Αυτό οφείλεται στην καθυστέρηση της πύλης NOT. Η ανίχνευση πιθανής αιχμής κατά τη λειτουργία του κυκλώματος μπορεί να γίνει σε επίπεδο πίνακα Karnaugh, όταν υπάρχουν ανεξάρτητες ομάδες μονάδων. Η επίλυση του προβλήματος γίνεται με την επιλογή μιας ακόμη ομάδας, η οποία θα καλύπτει τις δύο ανεξάρτητες ομάδες, όπως φαίνεται στο σχήμα 2.6.



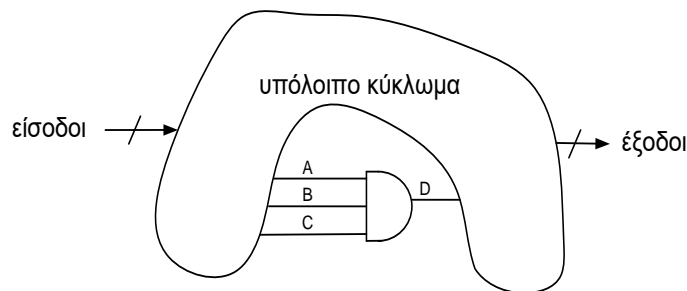
Σχήμα 2.6.

Η επιλογή μιας ακόμη ομάδας μονάδων στον πίνακα Karnaugh οδηγεί στην υλοποίηση του σχήματος 2.7, η οποία περιέχει μία πλεονασματική πύλη. Τώρα το απλό σφάλμα μόνιμης τιμής Q/0 είναι μη ανιχνεύσιμο.



Σχήμα 2.7.

Θεωρούμε μια πύλη AND με εισόδους A, B, C και έξοδο D (βλέπε σχήμα 2.8). Από τις σχέσεις κυριαρχίας της πύλης αυτής λαμβάνουμε: $A/1 \rightarrow D/1$, $B/1 \rightarrow D/1$ και $C/1 \rightarrow D/1$. Υποθέστε ότι λόγω πλεονασμού στις εισόδους της πύλης AND δεν εμφανίζονται οι συνδυασμοί: 011, 101 και 110. Τότε αν έχω εφαρμόσει τη σχέση της κυριαρχίας για να βγάλω από την λίστα σφαλμάτων το σφάλμα D/1, το σύνολο δοκιμής που θα εξάγω δεν θα ανιχνεύει το σφάλμα D/1 αν και το σφάλμα αυτό θα μπορούσε να ανιχνευτεί με το διάνυσμα 000.



Σχήμα 2.8.

Επομένως, στην περίπτωση που δεν γνωρίζουμε εάν το κύκλωμα για το οποίο θέλουμε να εξάγουμε σύνολο δοκιμής έχει πλεονασμούς, θα πρέπει για να μειώσουμε το μέγεθος της λίστας σφαλμάτων να χρησιμοποιήσουμε μόνο τις σχέσεις ισοδυναμίας.

Πολλαπλά σφάλματα μόνιμης τιμής

Θεωρούμε λογικό κύκλωμα με N γραμμές. Τότε το πλήθος των σφαλμάτων μόνιμης τιμής με πολλαπλότητα μ δίνεται από τη σχέση $2^\mu \times C(N, \mu)$ όπου $C(N, \mu) = N! / [\mu! \times (N - \mu)!]$.

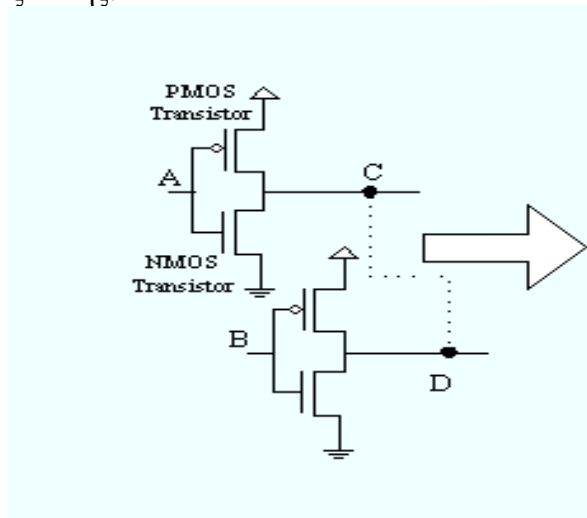
πλήθος κόμβων του κυκλώματος	πλήθος σφαλμάτων		
	απλά σφάλματα μόνιμης τιμής	διπλά σφάλματα μόνιμης τιμής	Τριπλά σφάλματα μόνιμης τιμής
N	$2 \times N$	$4 \times C(N, 2)$	$8 \times C(N, 3)$
10	20	180	960
100	200	19.800	1.3×10^6
1.000	2.000	1.998.000	$> 10^9$
10.000	20.000	199.980.000	$> 10^{12}$

Σχήμα 2.9.

Από το σχήμα 2.9 παρατηρούμε ότι το πλήθος των σφαλμάτων αυξάνεται πολύ γρήγορα με την αύξηση της πολλαπλότητας μ. Ακόμη και για μικρά κυκλώματα με 10.000 γραμμές, το πλήθος των διπλών σφαλμάτων μόνιμης τιμής και ακόμη περισσότερο των τριπλών σφαλμάτων είναι απαγορευτικά μεγάλο για να προσπαθήσουμε να εξάγουμε σύνολα δοκιμής για αυτά τα σφάλματα. Έχει διαπιστωθεί πάντως ότι τα σύνολα δοκιμής που εξάγουμε για την ανίχνευση των απλών σφαλμάτων μόνιμης τιμής ανιχνεύουν και αρκετά σφάλματα μόνιμης τιμής μεγαλύτερης πολλαπλότητας.

Σφάλματα βραχυκυκλώματος (Bridging Faults)

Δύο ή περισσότερες γραμμές, οι οποίες κανονικά είναι ανεξάρτητες, συνδέονται ηλεκτρικά. Εάν η μία είναι γραμμή σήματος και η άλλη είναι γραμμή τροφοδοσίας τότε το σφάλμα που θα προκύψει θα είναι ένα απλό σφάλμα μόνιμης τιμής. Στην περίπτωση όμως στην οποία και οι δύο γραμμές είναι γραμμές σήματος τα πράγματα είναι πολύ πιο δύσκολα. Στο σχήμα 2.10 φαίνονται δύο πύλες NOT σε CMOS τεχνολογία, των οποίων οι έξοδοι έχουν βραχυκυκλωθεί (διακεκομμένη γραμμή). Όταν η μία πύλη προσπαθεί να θέσει την έξοδό της στη λογική τιμή 1 και η άλλη στη λογική τιμή 0, δεν είναι εύκολο να προβλέψουμε την τιμή εξόδου των βραχυκυκλωμένων εξόδων, χωρίς γνωρίζουμε της αντίσταση των PMOS και NMOS τρανζίστορ κάθε μιας πύλης.



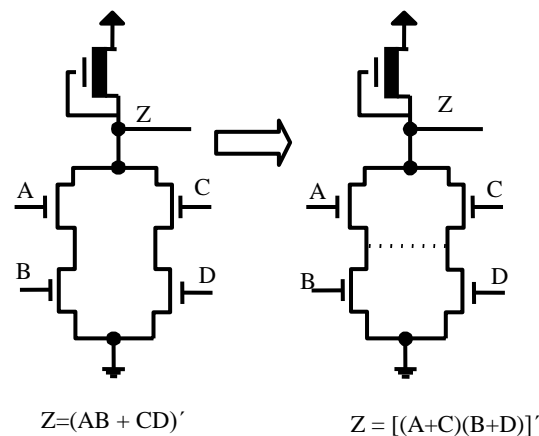
Σχήμα 2.10.

Λαμβάνοντας υπόψη ότι σε CMOS τεχνολογία έχουμε ροή ρεύματος από την τροφοδοσία μόνο κατά την αλλαγή της κατάστασης της εξόδου κάποιας πύλης, σφάλματα βραχυκυκλώματος μπορούν να ανιχνευθούν με Iddq έλεγχο. Στο Iddq έλεγχο μετράμε το ρεύμα τροφοδοσίας όταν το κύκλωμα βρίσκεται σε σταθερή κατάσταση και αν η τιμή του ρεύματος είναι μεγαλύτερη κάποιου κατωφλίου, θεωρούμε ότι το κύκλωμα είναι ελαττωματικό.

Για παράδειγμα στο κύκλωμα του σχήματος οδηγούμε τις εισόδους A και B των δύο πυλών NOT με συμπληρωματικές τιμές (πχ. A=1 και B=0) και αφού αφήσουμε να περάσει ένα χρονικό διάστημα ώστε οι εξόδοι των πυλών να πάνε σε μια σταθερή κατάσταση μετράμε το ρεύμα τροφοδοσίας. Εάν δεν υπάρχει βραχυκύκλωμα των εξόδων των δύο πυλών τότε η έξοδος C θα πάει στη λογική τιμή 0 και η έξοδος D στη λογική τιμή 1. Σ' αυτή την περίπτωση δεν υπάρχει αγώγιμο μονοπάτι από την τροφοδοσία προς τη γη. Εάν όμως υπάρχει βραχυκύκλωμα τότε υπάρχει αγώγιμο μονοπάτι από την τροφοδοσία προς τη γη μέσω του PMOS τρανζίστορ της πύλης με είσοδο B και του NMOS τρανζίστορ της πύλης με είσοδο A, οπότε θα υπάρχει σημαντική ροή ρεύματος.

Αυτή η μέθοδος έχει χρησιμοποιηθεί εκτεταμένα από τη βιομηχανία ολοκληρωμένων κυκλωμάτων για την ανίχνευση σφαλμάτων. Ωστόσο τα τελευταία χρόνια υπάρχουν κάποια προβλήματα εφαρμογής της λόγω των πολύ μεγάλων κυκλωμάτων και των μεγάλων ρευμάτων διαρροής.

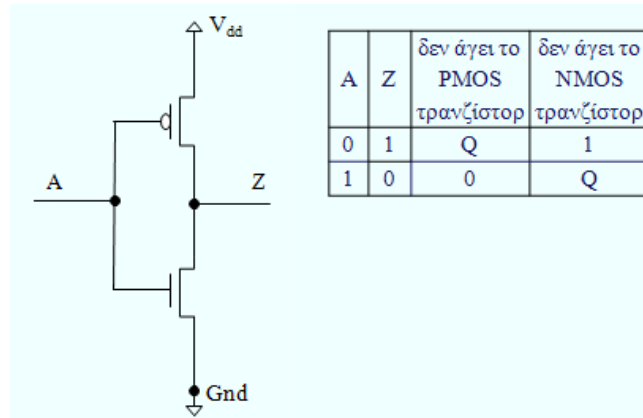
Στο προηγούμενο παράδειγμα είδαμε κάποιο σφάλμα βραχυκυκλώματος μεταξύ των εξόδων δύο πυλών. Σφάλμα βραχυκυκλώματος μπορεί να προκύψει και λόγω βραχυκυκλώματος μεταξύ δύο τρανζίστορ της ίδιας πύλης. Στο σχήμα 2.11 βλέπουμε ένα τέτοιο σφάλμα σε μια πύλη NMOS, το οποίο έχει ως συνέπεια την αλλαγή της λογικής συνάρτησης που υλοποιεί η πύλη.



Σχήμα 2.11.

Στη συνέχεια θα δούμε τις συνέπειες σε λογικό επίπεδο των επόμενων ελαττωμάτων σε κυκλώματα που είναι υλοποιημένα με τεχνολογία CMOS:

- τρανζίστορ που δεν άγει ποτέ
- τρανζίστορ που άγει μόνιμα
- ανοικτή γραμμή διασύνδεσης με πύλη (Open-on gate), πηγή (source) ή καταβόθρα (drain)
- βραχυκύκλωμα μεταξύ δύο εκ των ακροδεκτών πύλης, πηγής και καταβόθρας

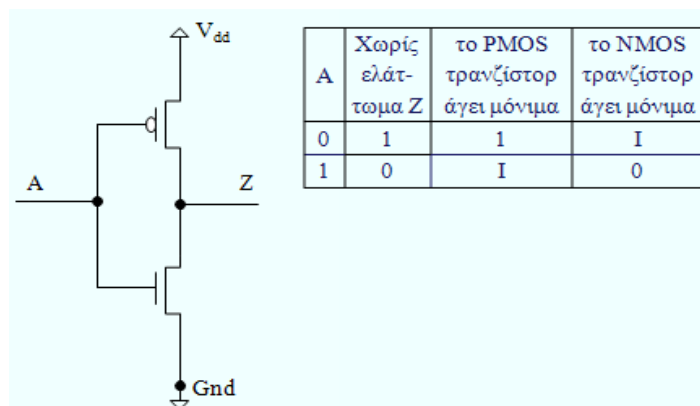


Σχήμα 2.12.

Στο σχήμα 2.12 θα εξετάσουμε δύο περιπτώσεις. Πρώτα την περίπτωση που το PMOS τρανζίστορ δεν άγει.

Όταν η είσοδος της πύλης έχει την λογική τιμή 1 τότε το NMOS τρανζίστορ άγει και η έξοδος της πύλης έχει τη λογική τιμή 0. Όταν η είσοδος της πύλης έχει την λογική τιμή 0, τότε το NMOS τρανζίστορ δεν άγει και επειδή και το PMOS τρανζίστορ λόγω του ελαττώματος δεν άγει, η έξοδος της πύλης είναι στον αέρα. Αυτή η κατάσταση στον πίνακα δηλώνεται με το σύμβολο Q. Αυτό το ελάττωμα δεν ανιχνεύεται με ένα διάνυσμα δοκιμής. Απαιτείται η χρησιμοποίηση δύο διανυσμάτων δοκιμής. Αρχικά θα οδηγήσουμε την είσοδο της πύλης με τη λογική τιμή 1 ώστε η έξοδος της να πάει στη λογική τιμή 0. Στη συνέχεια θα οδηγήσουμε την είσοδο της πύλης με τη λογική τιμή 0. Εάν υπάρχει το ελάττωμα η έξοδος της πύλης θα παραμείνει στη λογική τιμή 0, ενώ η αναμενόμενη τιμή είναι το 1. Επομένως με το ζεύγος των διανυσμάτων δοκιμής το σφάλμα ανιχνεύεται.

Η περίπτωση που το NMOS τρανζίστορ δεν άγει είναι ανάλογη.



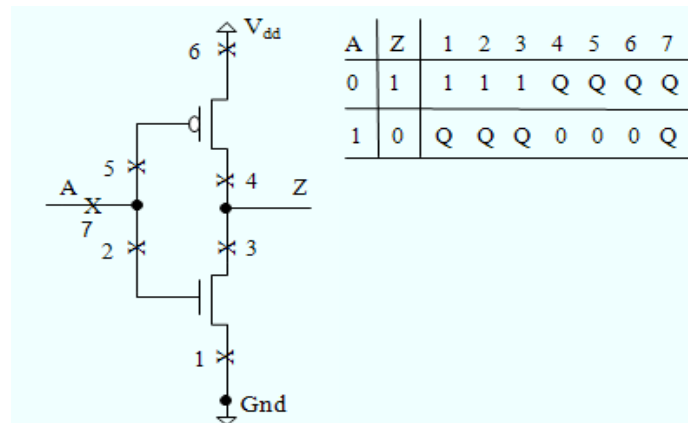
Σχήμα 2.13.

Στο σχήμα 2.13 θα εξετάσουμε δύο περιπτώσεις. Πρώτα την περίπτωση που το PMOS τρανζίστορ άγει μόνιμα.

Όταν η είσοδος της πύλης έχει τη λογική τιμή 0 τότε το NMOS τρανζίστορ δεν άγει, το PMOS τρανζίστορ άγει και η έξοδος της πύλης έχει τη λογική τιμή 1. Όταν η είσοδος της πύλης έχει τη λογική τιμή 1 τότε το NMOS τρανζίστορ άγει αλλά λόγω

του ελαττώματος άγει και το τρανζίστορ PMOS, επομένως υπάρχει αγωγίμο μονοπάτι από την τροφοδοσία προς τη γη. Χωρίς να γνωρίζουμε το λόγο των αντιστάσεων των PMOS και NMOS τρανζίστορ δεν μπορούμε να εκτιμήσουμε τη λογική τιμή της εξόδου, γι' αυτό τη θεωρούμε απροσδιόριστη. Στον πίνακα η απροσδιόριστη κατάσταση δηλώνεται με το σύμβολο I. Το ελάττωμα αυτό μπορούμε να το ανιχνεύσουμε με I_{ddq} έλεγχο.

Η περίπτωση που το NMOS τρανζίστορ άγει μόνιμα είναι ανάλογη.

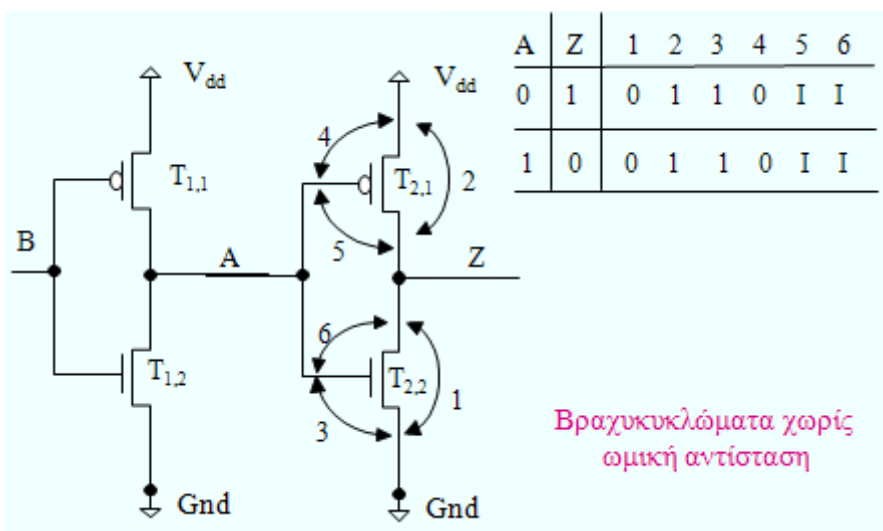


Σχήμα 2.14.

Στο σχήμα 2.14 τα ελαττώματα 1, 2 και 3 μπορούν να ανιχνευθούν εάν αρχικά οδηγήσουμε την είσοδο της πύλης με τη λογική τιμή 0 ώστε η έξοδος της να πάει στη λογική τιμή 1. Στη συνέχεια θα οδηγήσουμε την είσοδο της πύλης με τη λογική τιμή 1. Εάν υπάρχει κάποιο από τα ελαττώματα 1, 2 και 3 η έξοδος της πύλης θα παραμείνει στη λογική τιμή 1, ενώ η αναμενόμενη τιμή είναι το 0. Επομένως με το ζεύγος των διανυσμάτων δοκιμής το σφάλμα ανιχνεύεται.

Τα ελαττώματα 4, 5 και 6 ανιχνεύονται ανάλογα.

Το ελάττωμα 7 έχει ως συνέπεια με οποιαδήποτε τιμή και αν οδηγήσουμε την είσοδο της πύλης, η έξοδος της να παραμένει στον αέρα. Το σφάλμα αυτό ανήκει στην κατηγορία των πιθανώς ανιχνεύσιμων σφαλμάτων για τα οποία θα μιλήσουμε στο κεφάλαιο της εξομοίωσης σφαλμάτων.



Σχήμα 2.15 Βραχυκυκλώματα.

Στο σχήμα 2.15 τα ελαττώματα 1 και 2 από πλευράς λειτουργικότητας είναι ισοδύναμα με τα ελαττώματα το NMOS και το PMOS τρανζίστορ αντίστοιχα να άγει μόνιμα, με τα οποία έχουμε ήδη ασχοληθεί. Επομένως στη συνέχεια θα ασχοληθούμε με τα υπόλοιπα ελαττώματα. Στην ανάλυση που ακολουθεί θεωρούμε ότι τα βραχυκυκλώματα δεν έχουν ωμική αντίσταση.

Στην περίπτωση του ελαττώματος 3 ο κόμβος A θα έχει το δυναμικό 0, επομένως το PMOS τρανζίστορ άγει πάντα, το NMOS τρανζίστορ δεν άγει ποτέ και η έξοδος Z θα έχει τη λογική τιμή 1.

Στην περίπτωση του ελαττώματος 4 ο κόμβος A θα έχει το δυναμικό της τροφοδοσίας, επομένως το PMOS τρανζίστορ δεν άγει ποτέ, το NMOS τρανζίστορ άγει πάντα και η έξοδος Z θα έχει τη λογική τιμή 0.

Στη συνέχεια θα μελετήσουμε το ελάττωμα 5. Έστω ότι $B=0$. Τότε η πύλη με είσοδο το B προσπαθεί να θέσει την έξοδο A στη λογική τιμή 1, όμως επειδή τότε άγει το NMOS τρανζίστορ $T_{2,2}$, δημιουργείται ένα αγώγιμο μονοπάτι από την τροφοδοσία μέχρι τη γείωση μέσω του PMOS τρανζίστορ $T_{1,1}$, του βραχυκυκλώματος 5 και του NMOS τρανζίστορ $T_{2,2}$. Επομένως η λογική τιμή της εξόδου Z εξαρτάται από το λόγο των αντιστάσεων των δύο αυτών τρανζίστορ και είναι απροσδιόριστη αν δεν γνωρίζουμε το λόγο αυτό.

Έστω ότι $B=1$. Τότε η πύλη με είσοδο το B προσπαθεί να θέσει την έξοδο A στη λογική τιμή 0, όμως επειδή τότε άγει το PMOS τρανζίστορ $T_{2,1}$, δημιουργείται ένα αγώγιμο μονοπάτι από την τροφοδοσία μέχρι τη γείωση μέσω του PMOS τρανζίστορ $T_{2,1}$, του βραχυκυκλώματος 5 και του NMOS τρανζίστορ $T_{1,2}$. Επομένως η λογική τιμή της εξόδου Z εξαρτάται από το λόγο των αντιστάσεων των δύο αυτών τρανζίστορ και είναι απροσδιόριστη αν δεν γνωρίζουμε το λόγο αυτό.

Με τον ίδιο τρόπο μπορούμε να δούμε ότι και το ελάττωμα 6 οδηγεί σε απροσδιόριστη τιμή εξόδου.

Σφάλματα καθυστέρησης (Delay Faults)

Θα σχολιάσουμε τρία μοντέλα σφαλμάτων καθυστέρησης (Delay Faults): το μοντέλο σφαλμάτων μετάβασης (Transition faults), το μοντέλο σφαλμάτων καθυστέρησης πύλης (Gate delay fault model) και το μοντέλο σφάλματος καθυστέρησης μονοπατιών (Path delay fault model)

Το μοντέλο σφαλμάτων μετάβασης υποθέτει ότι το σφάλμα καθυστέρησης επηρεάζει μόνο μια πύλη του κυκλώματος. Το μοντέλο αυτό λαμβάνει υπόψη του την καθυστέρηση τόσο στη μετάβαση από 0 σε 1 όσο και στη μετάβαση από 1 σε 0. Το μοντέλο των σφαλμάτων μετάβασης θεωρεί ότι η καθυστέρηση που προκαλείται από το σφάλμα είναι αρκετά μεγάλη για να μην έχει φτάσει η σωστή τιμή σε κάποια έξοδο του κυκλώματος τη στιγμή της ανάγνωσής της. Δηλαδή οι συνέπειες του σφάλματος καθυστέρησης θα παρατηρηθούν ανεξάρτητα εάν η μετάβαση διαδίδεται μέσω ενός μονοπατιού μεγάλου ή μικρού μήκους.

Για να ανιχνεύσουμε ένα σφάλμα μετάβασης σ' ένα συνδυαστικό κύκλωμα απαιτούνται δύο διανύσματα δοκιμής. Το πρώτο διάνυσμα, αρχικοποιεί το κύκλωμα, το δεύτερο διάνυσμα ενεργοποιεί το σφάλμα και διαδίδει τις συνέπειές του μέχρι κάποια κύρια έξοδο του κυκλώματος. Για παράδειγμα για την ανίχνευση μιας μετάβασης από 1 σε 0 το πρώτο διάνυσμα αρχικοποιεί τη θέση του σφάλματος στην τιμή 1. Το δεύτερο διάνυσμα είναι ένα διάνυσμα δοκιμής για το απλό σφάλμα μόνιμης τιμής στη λογική τιμή 1. Το δεύτερο διάνυσμα θα πρέπει να διεγείρει το σφάλμα και να διαδώσει τις συνέπειές του σε κάποια από τις κύριες εξόδους του κυκλώματος.

Τα κύρια πλεονεκτήματα του μοντέλου αυτού είναι ότι το πλήθος των σφαλμάτων ισούται με το διπλάσιο του πλήθους των πυλών και ότι εργαλεία που έχουν αναπτυχθεί για την εξαγωγή διανυσμάτων δοκιμής και εξομοίωσης απλών σφαλμάτων μόνιμης τιμής μπορούν να τροποποιηθούν εύκολα για να δουλεύουν για σφάλματα μετάβασης. Το μειονέκτημα του μοντέλου είναι η υπόθεση ότι η καθυστέρηση που βάζει το σφάλμα είναι αρκετά μεγάλη ώστε οι συνέπειές της να εκδηλώνονται ακόμη και όταν διαδίδονται μέσω ενός μονοπατιού πολύ μικρής καθυστέρησης. Επίσης η υπόθεση ότι το σφάλμα επηρεάζει την καθυστέρηση μιας μόνο πύλης δεν είναι ρεαλιστική.

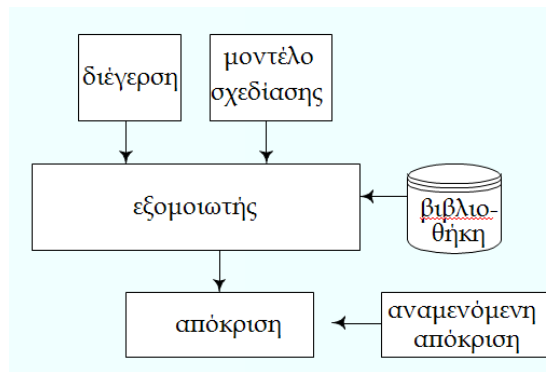
Το μοντέλο σφάλματος καθυστέρησης πύλης (gate delay fault model) και πάλι θεωρεί ότι η καθυστέρηση αθροίζεται σε μια πύλη του κυκλώματος, ωστόσο δεν θεωρεί ότι οι συνέπειες του σφάλματος θα επηρεάσουν την απόδοση του κυκλώματος άσχετα του μονοπατιού μέσω του οποίου θα διαδοθούν προς κάποια κύρια έξοδο του κυκλώματος. Θεωρεί ότι μόνο εάν οι συνέπειες του σφάλματος διαδοθούν μέσω μονοπατιού μεγάλης καθυστέρησης θα επηρεάσουν την απόδοση του κυκλώματος. Για να προσδιοριστεί εάν ένα σφάλμα αυτού του μοντέλου ανιχνεύεται, πρέπει να καθορίσουμε το μέγεθος της καθυστέρησης που βάζει. Και αυτό το μοντέλο σφαλμάτων έχει το μειονέκτημα ότι δεν λαμβάνει υπόψη του τις αθροιστικές συνέπειες σφαλμάτων καθυστέρησης σε διαφορετικές πύλες και γραμμές διασύνδεσης.

Σύμφωνα με το μοντέλο σφαλμάτων καθυστέρησης μονοπατιών (path delay fault model) ένα κύκλωμα θεωρείται ελαττωματικό εάν η καθυστέρηση οποιουδήποτε από τα μονοπάτια του ξεπερνάει κάποια συγκεκριμένη τιμή. Είναι το πλέον ακριβές μοντέλο σφαλμάτων καθυστέρησης αλλά έχει το πρόβλημα ότι το πλήθος των μονοπατιών από μια είσοδο μέχρι μια έξοδο αυξάνεται εκθετικά με το πλήθος των πυλών.

Τα προσωρινά σφάλματα εμφανίζονται πιο συχνά στο πεδίο της εφαρμογής και είναι δύσκολο να βρούμε την αιτία τους.

Κεφάλαιο 3. Λογική εξομοίωση και εξομοίωση σφαλμάτων

Η επιβεβαίωση σχεδιασμού μπορεί να γίνει με λογική εξομοίωση. Ένα από τα μειονεκτήματα της λογικής εξομοίωσης είναι ότι είναι αργή. Η εφαρμογή όλων των δυνατών συνδυασμών εισόδου δεν είναι δυνατή. Εναπόκειται στην εμπειρία και τις ικανότητες του σχεδιαστή να επιλέξει το κατάλληλο σύνολο διανυσμάτων εισόδου προς επιβεβαίωση του σχεδιασμού του. Η εμπειρία έχει δείξει ότι σχεδιαστικά λάθη εμφανίζονται πιο συχνά στις μονάδες ελέγχου απ' ό,τι στις μονάδες επεξεργασίας δεδομένων. Επομένως ο σχεδιαστής πρέπει να αφιερώσει περισσότερο χρόνο για την επιβεβαίωση σχεδιασμού της μονάδας ελέγχου. Ένας άλλος τρόπος επιβεβαίωσης σχεδιασμού είναι πριν κατασκευαστεί ένα VLSI κύκλωμα να υλοποιηθεί σε ένα ή περισσότερα FPGAs. Μια ακόμη μέθοδος, η οποία κερδίζει συνεχώς έδαφος, είναι η τυπική εξομοίωση, στην οποία η επιβεβαίωση σχεδιασμού γίνεται με μαθηματικά μοντέλα.



Σχήμα 3.1 Σύστημα λογικής εξομοίωσης

Ο εξομοιωτής δέχεται ως είσοδο μια περιγραφή της σχεδίασης και τη βιβλιοθήκη που χρησιμοποιείται για την υλοποίηση της σχεδίασης. Επίσης δέχεται ως είσοδο τη διέγερση. Ο εξομοιωτής παράγει την απόκριση του κυκλώματος για τις εισόδους που δέχεται. Η απόκριση συγκρίνεται με την αναμενόμενη απόκριση. Η διέγερση μπορεί να είναι ένα αρχείο με λογικές τιμές, με κυματομορφές, ψευδοτυχαία διανύσματα ή test benches.

Μπορούμε να διακρίνουμε τους εξομοιωτές σε κατηγορίες ανάλογα του επιπέδου στο οποίο γίνεται η περιγραφή του συστήματος, εάν γίνεται λειτουργική ή χρονική εξομοίωση, ανάλογα με τον τρόπο υλοποίησής τους και αν χρησιμοποιούνται για λογική εξομοίωση ή εξομοίωση σφαλμάτων.

Ανάλογα του επιπέδου στο οποίο γίνεται η περιγραφή του υπό εξομοίωση συστήματος μπορούμε να διακρίνουμε: εξομοίωση σε επίπεδο συμπεριφοράς (Behavioral level), εξομοίωση σε επίπεδο καταχωρητή (RTL), εξομοίωση σε λογικό επίπεδο (Logic level), εξομοίωση σε επίπεδο διακόπτη (Switch level), εξομοίωση σε επίπεδο κυκλώματος (Circuit level) και μεικτού τύπου εξομοίωση (Mixed mode simulation).

Όσο κατεβαίνουμε από εξομοίωση σε επίπεδο συμπεριφοράς προς εξομοίωση σε επίπεδο κυκλώματος, τόσο αυξάνεται η ακρίβεια της εξομοίωσης, αλλά τόσο αυξάνεται και ο απαιτούμενος χρόνος για την εξομοίωση. Σε επίπεδο διακοπών η περιγραφή του κυκλώματος γίνεται σε επίπεδο τρανζίστορ. Ωστόσο το τρανζίστορ αντιμετωπίζεται ως ένας διακόπτης που είναι είτε ανοικτός είτε κλειστός. Σε επίπεδο κυκλώματος η περιγραφή γίνεται και πάλι σε επίπεδο τρανζίστορ, αλλά το τρανζίστορ αντιμετωπίζεται ως αναλογικό κύκλωμα. Οι μεικτοί εξομοιωτές μας δίνουν τη δυνατότητα να περιγράψουμε ένα σύστημα σ' ένα υψηλότερο επίπεδο, πχ. σε επίπεδο καταχωρητή και ένα μικρό μέρος του συστήματος σε χαμηλότερο επίπεδο, πχ. σε επίπεδο κυκλώματος. Μ' αυτό τον τρόπο επιτυγχάνουμε να έχουμε την απαιτούμενη λεπτομέρεια για το τμήμα του συστήματος που μας ενδιαφέρει, χωρίς να χάνουμε σημαντικά σε χρόνο εξομοίωσης.

Η λογική εξομοίωση μας επιβεβαιώνει ότι το σχεδιασμένο κύκλωμα λειτουργεί σύμφωνα με τις προδιαγραφές του. Για παράδειγμα, ότι ένας άθροιστής παράγει πράγματι στην έξοδο του το άθροισμα των δεδομένων με τα οποία τον οδηγούμε. Δεν μας αρκεί όμως μόνο αυτό. Πρέπει να επιβεβαιώσουμε ότι το κύκλωμά μας λειτουργεί με την απαιτούμενη ταχύτητα. Αυτό γίνεται με τη χρονική εξομοίωση.

Ο χρόνος που απαιτείται για την εξομοίωση, επιπλέον του επιπέδου στο οποίο γίνεται η εξομοίωση, εξαρτάται επίσης από το χρονικό μοντέλο που υιοθετείται. Το χρονικό μοντέλο επηρεάζει και την ακρίβεια της χρονικής εξομοίωσης.

Καθυστέρηση διάδοσης (transport delay) καλείται ο χρόνος που απαιτείται από τη στιγμή που θα αλλάξει η είσοδος μιας πύλης μέχρι να αλλάξει η έξοδός της. Στην περίπτωση χρονικής εξομοίωσης δεν έχει νόημα το χρονικό μοντέλο μηδενικής καθυστέρησης. Σύμφωνα με το χρονικό μοντέλο μοναδιαίας καθυστέρησης όλες οι πύλες του κυκλώματος θεωρούμε ότι έχουν την ίδια καθυστέρηση. Σύμφωνα με το μοντέλο ονομαστικής καθυστέρησης (nominal delay) κάθε στοιχείο του κυκλώματος έχει τη δική του καθυστέρηση διάδοσης, αλλά την ίδια για μετάβαση από 0 σε 1 και μετάβαση από 1 σε 0. Συνήθως ο χρόνος διάδοσης εξαρτάται από τη μετάβαση της εξόδου από 0 σε 1 ή 1 σε 0, οπότε για κάθε πύλη δίνουμε διαφορετικούς χρόνους για κάθε είδος μετάβασης. Για να είμαστε πιο ακριβείς η καθυστέρηση διάδοσης δεν εξαρτάται μόνο από το είδος της μετάβασης της εξόδου από 0 σε 1 ή 1 σε 0 αλλά και από το ποια είσοδος της πύλης προκάλεσε τη μετάβαση. Επομένως για μεγαλύτερη ακρίβεια, για κάθε είσοδο του κυκλώματος και για κάθε είδους μετάβαση δίνουμε μια διαφορετική τιμή της καθυστέρησης διάδοσης.

Για πολλούς λόγους, όπως για παράδειγμα λόγω των διακυμάνσεων διαφόρων παραμέτρων κατά τη διαδικασία κατασκευής του ολοκληρωμένου κυκλώματος, η καθυστέρηση διάδοσης μιας πύλης μπορεί να βρίσκεται μεταξύ μιας ελάχιστης και μιας μέγιστης τιμής. Για να περιγράψουμε αυτό το φαινόμενο χρησιμοποιούμε το μοντέλο της ελάχιστης - μέγιστης καθυστέρησης. Το μοντέλο της ελάχιστης - μέγιστης καθυστέρησης μπορεί να συνδυασθεί με το γεγονός ότι ο χρόνος καθυστέρησης - διάδοσης από 0 σε 1 και από 1 σε 0 είναι διαφορετικός.

Η καθυστέρηση αδράνειας (inertial delay) ορίζεται ως η ελάχιστη χρονική διάρκεια που πρέπει να έχει ένας παλμός στην είσοδο μιας πύλης για να αλλάξει η κατάσταση της εξόδου της.

Λόγω της συνεχούς προόδου στην τεχνολογία κατασκευής ολοκληρωμένων κυκλωμάτων, οι διαστάσεις των πυλών συνεχώς μικραίνουν (scaling) με αποτέλεσμα να μικραίνει και η καθυστέρησή τους. Ωστόσο οι καθυστερήσεις των γραμμών διασύνδεσης δεν ωφελούνται τόσο πολύ από αυτή τη μείωση των διαστάσεων, με αποτέλεσμα οι καθυστερήσεις των γραμμών διασύνδεσης να παίζουν κύριο ρόλο στην καθυστέρηση ενός κυκλώματος.

Τα μονοπάτια μέγιστης καθυστέρησης σε ένα κύκλωμα μπορούν να βρεθούν χρησιμοποιώντας στατική χρονική ανάλυση (Static Timing Analysis, STA). Σύμφωνα με τη στατική χρονική ανάλυση προστίθεται ο μέγιστος χρόνος καθυστέρησης διάδοσης κάθε συνιστώσας ενός μονοπατιού του κυκλώματος για να προσδιοριστεί η μέγιστη καθυστέρηση διάδοσης μέσω αυτού του μονοπατιού. Το πλεονέκτημα της μεθόδου είναι η χαμηλή πολυπλοκότητά της. Είναι δυνατόν όμως να προσδιορισθεί ως μονοπάτι μέγιστης καθυστέρησης ένα μονοπάτι το οποίο δεν ενεργοποιείται ποτέ κατά την κανονική λειτουργία του κυκλώματος (false paths).

Όσον αφορά την υλοποίηση των εξομοιωτών θα ασχοληθούμε με τους εξομοιωτές μεταφρασμένου κώδικα (compiled) και τους οδηγούμενους από γεγονότα (event driven). Αρχικά θα ασχοληθούμε με τους εξομοιωτές μεταφρασμένου κώδικα.

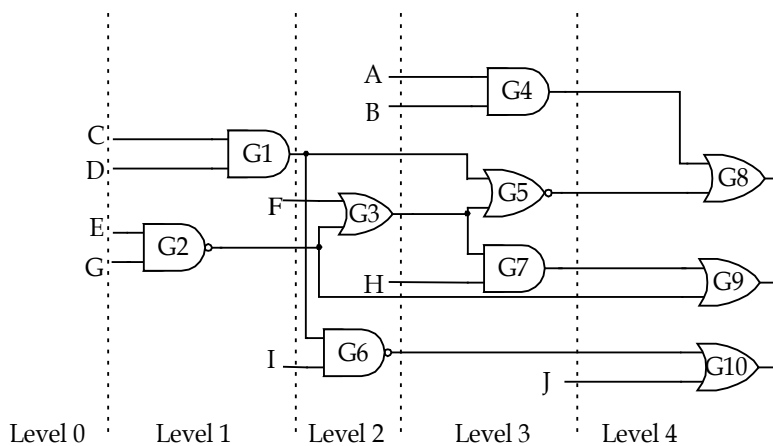
Εξομοιωτές μεταφρασμένου κώδικα

Διαδικασία

- Διάβασε την περιγραφή του κυκλώματος
- Σπάσε τα μονοπάτια ανάδρασης, αν υπάρχουν
- Διάταξε τις πύλες σε επίπεδα
- Παρήγαγε το μεταφρασμένο κώδικα
- Διάβασε την αρχική τιμή κάθε γραμμής εισόδου
- Διάβασε το επόμενο διάνυσμα εισόδου και ενημέρωσε τις τιμές

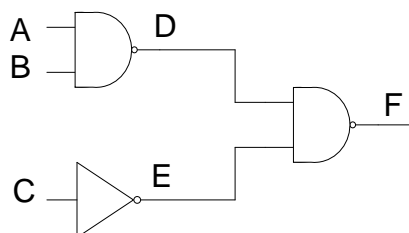
Για κάθε νέα δεδομένα εισόδου επανέλαβε
 Για κάθε επίπεδο λογικής επανέλαβε
 Για κάθε πύλη του επιπέδου επανέλαβε
 Εκτέλεσε τις εντολές που αντιστοιχούν στην πύλη
 Τέλος επανάληψης
 Τέλος επανάληψης
 Εάν οι νέες τιμές των εξόδων των γραμμών ανάδρασης είναι ίδιες με τις παλιές τότε βγάλε αποτελέσματα
 Διαφορετικά έθεσε στις εισόδους των γραμμών ανάδρασης τις τιμές των εξόδων
 Τέλος επανάληψης
Τέλος διαδικασίας

Στο σχήμα 3.2 φαίνεται ο τρόπος με τον οποίο οι πύλες του κυκλώματος κατανέμονται σε επίπεδα. Αρχικά δίνονται τιμές στις εισόδους του κυκλώματος και υπολογίζονται οι τιμές εξόδου των πυλών του πρώτου επιπέδου. Στη συνέχεια υπολογίζονται οι τιμές εξόδου των πυλών του δεύτερου επιπέδου και ούτω καθεξής. Με τον τρόπο αυτό, όταν θέλουμε να υπολογίσουμε την τιμή εξόδου μιας πύλης γνωρίζουμε ήδη τις τιμές των εισόδων της.



Σχήμα 3.2 Διάταξη των πυλών σε επίπεδα (levelization)

Στο σχήμα 3.3 βλέπουμε το λογικό κύκλωμα το οποίο θέλουμε να εξομοιώσουμε, στο σχήμα 3.4 τις εντολές του προγράμματος που περιγράφουν το κύκλωμα και στο σχήμα 3.5 τις κυματομορφές εισόδου και τις κυματομορφές εξόδου. Θεωρούμε ότι η καθυστέρηση όλων των πυλών είναι μηδενική. Σε κάθε χρονική στιγμή 0, 1, 2 ... 9 εκτελούμε το πρόγραμμα. Αυτό έχει ως αποτέλεσμα τον υπολογισμό της τιμής όλων των κόμβων του κυκλώματος.



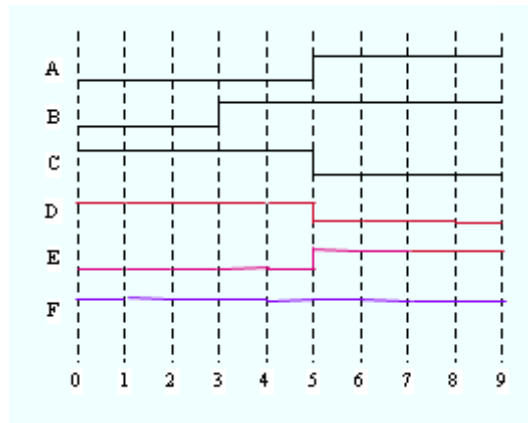
Σχήμα 3.3 Κύκλωμα για λογική εξομοίωση

```

LOAD r1, A
LOAD r2, B
AND r1, r2
COMPLEMENT r1
LOAD r3, C
COMPLEMENT r3
AND r1, r3
COMPLEMENT r1
STORE F, r1

```

Σχήμα 3.4 Εντολές του προγράμματος που περιγράφουν το κύκλωμα



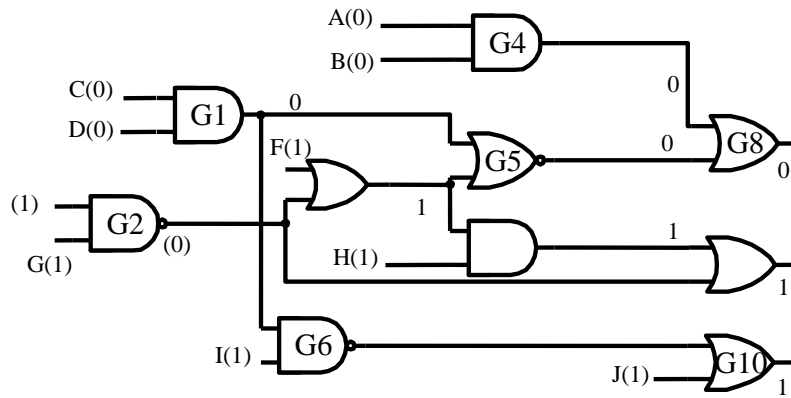
Σχήμα 3.5 Κυματομορφές εισόδου και εξόδου

Παρατηρούμε ότι αν και οι τιμές των εισόδων του κυκλώματος δεν έχουν αλλάξει κατά τη χρονική στιγμή 1 και 2, παρόλα αυτά το πρόγραμμα ξανά εκτελείται για καθεμία από αυτές τις χρονικές στιγμές. Επίσης πρώτα υπολογίζονται οι τιμές εξόδου των πυλών του πρώτου επιπέδου και μετά του δεύτερου επιπέδου.

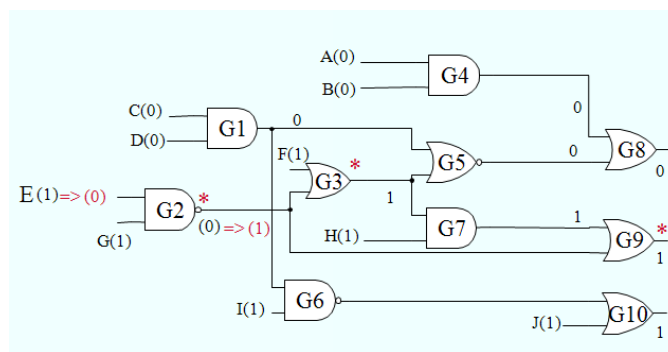
Οι εξομοιωτές μεταφρασμένου κώδικα έχουν δύο μεγάλα μειονεκτήματα. Πρώτον, για κάθε συνδυασμό εισόδου υπολογίζουν τις τιμές όλων των κόμβων του κυκλώματος, ακόμη και αν σε κανένα κόμβο δεν προκύπτει νέα τιμή. Αυτό τους καθιστά αργούς. Επίσης λόγω του τρόπου υλοποίησής τους (πρώτα υπολογίζονται οι τιμές εξόδου των πυλών του πρώτου επιπέδου μετά του δεύτερου και ούτω καθεξής) δεν μπορούν να λάβουν υπόψη τους τις καθυστερήσεις των πυλών. Επομένως δεν μπορούν να ανιχνεύσουν αιχμές που πιθανόν να εμφανίζονται στις εξόδους του υπό εξομοίωση κυκλώματος.

Εξομοιωτές οδηγούμενοι από γεγονότα

Θεωρήστε το κύκλωμα του σχήματος. Θεωρήστε επίσης ότι οι εισοδοί του κυκλώματος έχουν τροφοδοτηθεί με τις τιμές που φαίνεται στο σχήμα και έχουν υπολογιστεί οι τιμές σε όλους τους κόμβους του κυκλώματος.



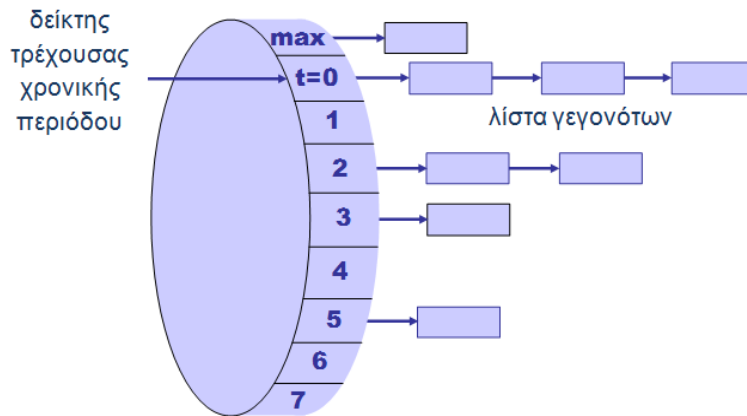
Σχήμα 3.5 Κύκλωμα για εξομοίωση – αρχικό διάνυσμα εισόδου



Σχήμα 3.6 Κύκλωμα για εξομοίωση – νέο διάνυσμα εισόδου

Θεωρήστε στη συνέχεια ότι η είσοδος E του κυκλώματος άλλαξε τιμή από 1 σε 0 (βλέπε σχήμα 3.6). Θα δούμε στη συνέχεια τι θα συμβεί κατά την εξομοίωση του κυκλώματος με έναν εξομοιωτή που οδηγείται από γεγονότα. Εφόσον άλλαξε τιμή η είσοδος μιας πύλης, θα πρέπει να υπολογιστεί η τιμή της εξόδου της. Υπολογίζεται λοιπόν η τιμή εξόδου της πύλης G2 και παρατηρούμε ότι η τιμή της άλλαξε από 0 σε 1. Επομένως θα πρέπει να υπολογιστεί η τιμή εξόδου των πυλών G3 και G9 οι οποίες οδηγούνται από την έξοδο της πύλης G2. Υπολογίζονται οι τιμές εξόδου των πυλών G3 και G9 και παρατηρούμε ότι παραμένουν οι ίδιες. Επομένως σταματά η εξομοίωση. Ο εξομοιωτής μεταφρασμένου κώδικα θα υπολόγιζε τις τιμές όλων των κόμβων του κυκλώματος.

Ο χρονοπρογραμματιστής (scheduler) είναι μια σημαντική συνιστώσα στους εξομοιωτές που οδηγούνται από γεγονότα. Παρακολουθεί το συμβάν των γεγονότων και προγραμματίζει τους αναγκαίους υπολογισμούς των τιμών των εξόδων των πυλών. Στην περίπτωση που θεωρούμε ότι οι πύλες έχουν μηδενική καθυστέρηση, η δομή της ουράς (event queue) είναι μια καλή υλοποίηση του χρονοπρογραμματιστή. Στην περίπτωση όμως στην οποία θέλουμε να λάβουμε υπόψη μας ονομαστικές καθυστερήσεις, χρειάζεται ένας πιο πολύπλοκος χρονοπρογραμματιστής διότι πρέπει να προσδιορίσει όχι μόνο την πύλη της οποίας η τιμή εξόδου πρέπει να υπολογιστεί αλλά επίσης πότε να γίνει ο υπολογισμός. Μια καλή υλοποίηση είναι ο τροχός του χρόνου (βλέπε σχήμα 3.7). Όταν αλλάζει η τιμή εισόδου μιας πύλης η οποία έχει



Σχήμα 3.7 Τροχός του χρόνου (timing wheel)

καθυστέρηση κ χρονικών μονάδων, τότε αυτή συνδέεται στη λίστα γεγονότων που βρίσκεται κ χρονικές μονάδες μετά την τρέχουσα χρονική περίοδο. Κάθε χρονική περίοδο υπολογίζονται οι τιμές εξόδου των πυλών που βρίσκονται στη λίστα που είναι συνδεδεμένη σ' αυτή τη χρονική περίοδο.

Εξομοιωτής Οδηγούμενος από γεγονότα (Event-Driven):

Διαδικασία:

Διάβασε την περιγραφή του κυκλώματος

Διάβασε διανύσματα εισόδου

Για κάθε διάνυσμα εισόδου που πρέπει να εξομοιωθεί επανέλαβε

 ενημέρωσε τις εισόδους

 χρονοπρογραμματίσε τις πύλες που οδηγούνται από εισόδους που

 άλλαξαν τιμή

 εφόσον υπάρχουν πύλες προς επεξεργασία

 υπολόγισε την τιμή εξόδου μιας πύλης

 εάν άλλαξε η τιμή εξόδου μιας πύλης

 τότε χρονοπρογραμματίσε τις πύλες που οδηγούνται από αυτή

 τέλος επανάληψης

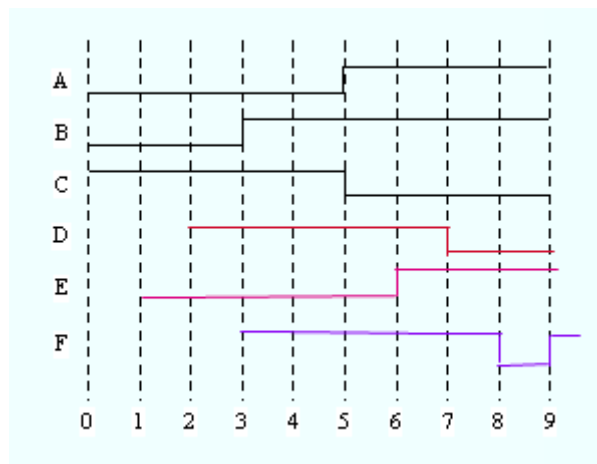
 τέλος επανάληψης

τέλος διαδικασίας

Θεωρούμε και πάλι το κύκλωμα του σχήματος 3.3. Στα σχήματα 3.8 και 3.9 δίνεται ο πίνακας χρονοπρογραμματισμού του υπολογισμού των τιμών των κόμβων του κυκλώματος και οι κυματομορφές στις εισόδους και στους διάφορους κόμβους του κυκλώματος.

χρόνος	σήμα προς υπολογισμό
1	E
2	D
3	F
4	F
5	D
6	E
7	D, F
8	F
9	F

Σχήμα 3.8



Σχήμα 3.9

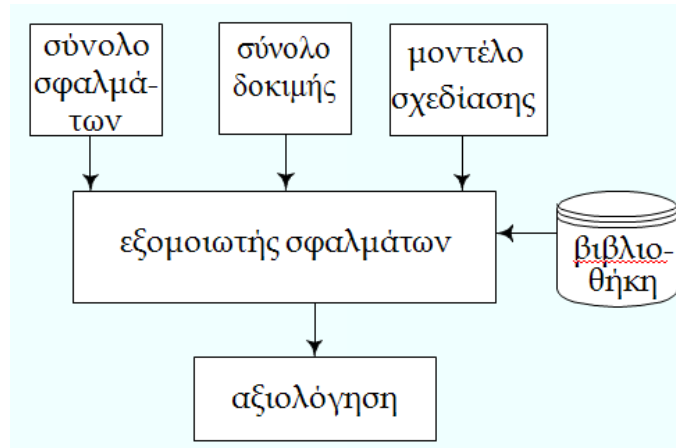
Παρατηρούμε ότι εμφανίζεται μια αιχμή (hazard) στην έξοδο του κυκλώματος η οποία δεν προέκυπτε στην περίπτωση της εξομοίωσης με τον εξομοιωτή μεταφρασμένου κώδικα.

Οι εξομοιωτές που οδηγούνται από γεγονότα έχουν δύο μεγάλα πλεονεκτήματα. Έχουν σημαντικά μικρότερο χρόνο εξομοίωσης από τους εξομοιωτές μεταφρασμένου κώδικα και μπορούν να λάβουν υπόψη τους τις χρονικές καθυστερήσεις των πυλών, γεγονός που οδηγεί στην ανίχνευση αιχμών.

Στη συνέχεια θα μιλήσουμε για την εξομοίωση σφαλμάτων. Σκοπός της εξομοίωσης σφαλμάτων είναι η ανίχνευση συγκεκριμένων εσφαλμένων συνθηκών, η καθοδήγηση της διαδικασίας εξαγωγής διανυσμάτων δοκιμής, η μέτρηση της απόδοσης του συνόλου δοκιμής και η εξαγωγή λεξικών σφαλμάτων.

Συγκρίνοντας τη λογική εξομοίωση με την εξομοίωση σφαλμάτων (βλέπε σχήμα 3.10) μπορούμε να πούμε ότι και η λογική εξομοίωση έχει ως στόχο την ανίχνευση

συγκεκριμένων εσφαλμένων συνθηκών, οι οποίες όμως προκύπτουν από σχεδιαστικά λάθη, ενώ στην περίπτωση της εξομοίωσης σφαλμάτων οι εσφαλμένες καταστάσεις προκύπτουν από κατασκευαστικά ελαττώματα. Στη λογική εξομοίωση είπαμε επίσης ότι η επιλογή των κατάλληλων συνδυασμών εισόδου είναι ένα μεγάλο πρόβλημα και εξαρτάται από την εμπειρία και την ικανότητα του μηχανικού.



Σχήμα 3.10 Εξομοίωση σφαλμάτων

Συγκρίνοντας την πληροφορία που λαμβάνει ως είσοδο ο εξομοιωτής σφαλμάτων με αυτή που λαμβάνει ο λογικός εξομοιωτής, παρατηρούμε ότι ο εξομοιωτής σφαλμάτων λαμβάνει ως είσοδο και το σύνολο των σφαλμάτων για τα οποία θέλουμε να γίνει εξομοίωση.

Όπως έχουμε ήδη αναφέρει σε ένα κύκλωμα υπάρχουν ομάδες σφαλμάτων τα οποία είναι ισοδύναμα. Αυτό σημαίνει ότι εάν κάποιο διάνυσμα δοκιμής ανιχνεύει κάποιο από τα ισοδύναμα σφάλματα, θα ανιχνεύει και όλα τα υπόλοιπα. Επομένως δεν είναι απαραίτητο και δεν προσφέρει τίποτα να κάνουμε εξομοίωση για κάθε σφάλμα μιας ομάδας ισοδύναμων σφαλμάτων. Αρκεί να κάνουμε εξομοίωση για ένα μόνο σφάλμα από κάθε ομάδα ισοδύναμων σφαλμάτων (fault collapsing). Η διαδικασία αυτή οδηγεί σε σημαντική μείωση του χρόνου εξομοίωσης σφαλμάτων.

Υπάρχουν περιπτώσεις στις οποίες μας ενδιαφέρει να δούμε ποια σφάλματα ανιχνεύονται από ένα σύνολο διανυσμάτων δοκιμής, χωρίς να μας ενδιαφέρει να βρούμε τα σφάλματα τα οποία ανιχνεύει κάθε διάνυσμα δοκιμής. Σε αυτή την περίπτωση όταν γίνεται εξομοίωση των σφαλμάτων της λίστας σφαλμάτων για ένα διάνυσμα δοκιμής, όλα τα σφάλματα που ανιχνεύονται από το διάνυσμα αυτό αφαιρούνται από τη λίστα σφαλμάτων. Επομένως όταν γίνεται εξομοίωση σφαλμάτων για ένα άλλο διάνυσμα δοκιμής δεν θα ληφθούν υπόψη τα σφάλματα αυτά (fault dropping). Η διαδικασία αυτή οδηγεί σε σημαντική μείωση του χρόνου εξομοίωσης σφαλμάτων.

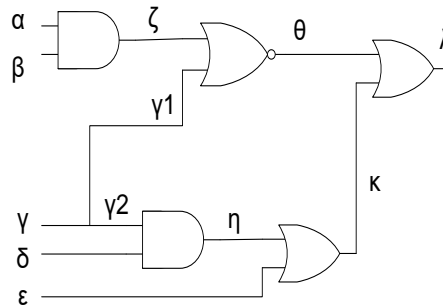
Οι εξομοιωτές σφαλμάτων σε λογικό επίπεδο ανάλογα του τρόπου υλοποίησής τους διακρίνονται σε σειριακούς, παράλληλους, επαγωγικούς και σύνδρομους.

Ο σειριακός εξομοιωτής απλών σφαλμάτων προσφέρει την απλούστερη περιγραφή του πως δουλεύει ένας εξομοιωτής σφαλμάτων. Ο σειριακός εξομοιωτής παίρνει τις τιμές εξόδου του κυκλώματος χωρίς σφάλμα κάνοντας λογική εξομοίωση και στην συνέχεια επαναλαμβάνει την λογική εξομοίωση του κυκλώματος θεωρώντας ότι ένα σφάλμα υπάρχει σ' ένα κόμβο του κυκλώματος, για παράδειγμα ότι μια γραμμή του κυκλώματος έχει μια σταθερή τιμή X , όπου $X=0$ ή 1 . Εάν οι δυο εξομοιώσεις δίνουν διαφορετικά αποτελέσματα το σφάλμα ανιχνεύθηκε. Εάν ο συνολικός αριθμός

σφαλμάτων σε ένα κύκλωμα είναι M τότε απαιτούνται $M+1$ περάσματα, εκτελέσεις του προγράμματος εξομοίωσης, για κάθε διάνυσμα δοκιμής για να γίνει η εξομοίωση αυτών των σφαλμάτων του κυκλώματος. Είναι φανερό ότι η σειριακή εξομοίωση σφαλμάτων είναι μια αργή διαδικασία.

Στη συνέχεια θα δούμε ένα παράδειγμα σειριακής εξομοίωσης.

Θεωρήστε το κύκλωμα του σχήματος 3.11, το σύνολο των διανυσμάτων δοκιμής του σχήματος 3.12 και τη λίστα σφαλμάτων του σχήματος 3.13.



Σχήμα 3.11 Κύκλωμα προς εξομοίωση

Διανύσματα δοκιμής				
α	β	γ	δ	ε
0	1	0	1	1
1	1	0	0	0
0	0	1	1	1
0	0	1	1	0

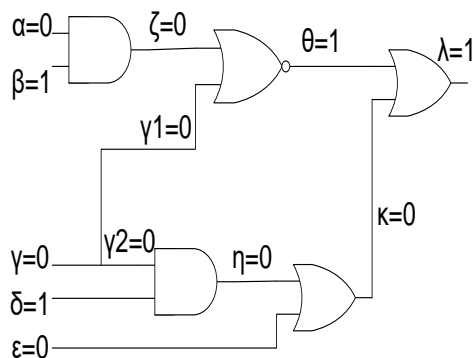
Σχήμα 3.12 Το σύνολο των διανυσμάτων δοκιμής

λίστα σφαλμάτων
θ/0
θ/1
ζ/0
ζ/1
η/0
κ/1
λ/0

Σχήμα 3.13 Λίστα σφαλμάτων

Στη σειριακή εξομοίωση σφαλμάτων θα πρέπει να επαναλάβουμε τα επόμενα βήματα τόσες φορές όσα είναι τα διανύσματα δοκιμής:

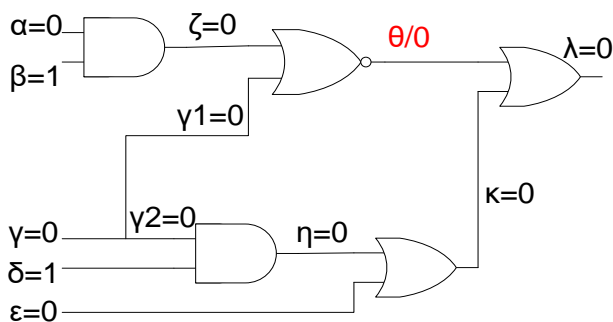
Λογική εξομοίωση για ένα διάνυσμα δοκιμής και στη συνέχεια εξομοίωση σφαλμάτων για κάθε σφάλμα που υπάρχει στη λίστα σφαλμάτων.



Σχήμα 3.14 Λογική εξομοίωση για το διάνυσμα εισόδου 01010.

Κάνουμε λογική εξομοίωση για το πρώτο διάνυσμα δοκιμής του συνόλου δοκιμής και παρατηρούμε ότι για το πρώτο διάνυσμα δοκιμής η έξοδος του κυκλώματος όταν δεν έχει σφάλμα λαμβάνει τη λογική τιμή 1 (βλέπε σχήμα 3.14).

Θέτουμε τη γραμμή θ του κυκλώματος στη λογική τιμή 0 και κάνουμε λογική εξομοίωση για το ίδιο διάνυσμα εισόδου (βλέπε σχήμα 3.15). Παρατηρούμε ότι η έξοδος λαμβάνει τη λογική τιμή 0 η οποία είναι διαφορετική της τιμής την οποία λάβαμε όταν θεωρήσαμε ότι το κύκλωμα δεν έχει κανένα σφάλμα. Επομένως το σφάλμα απλής μόνιμης τιμής της γραμμής θ στη λογική τιμή 0 ανιχνεύεται από το συγκεκριμένο διάνυσμα δοκιμής. Η διαδικασία θα επαναληφθεί για το επόμενο σφάλμα της λίστας σφαλμάτων.



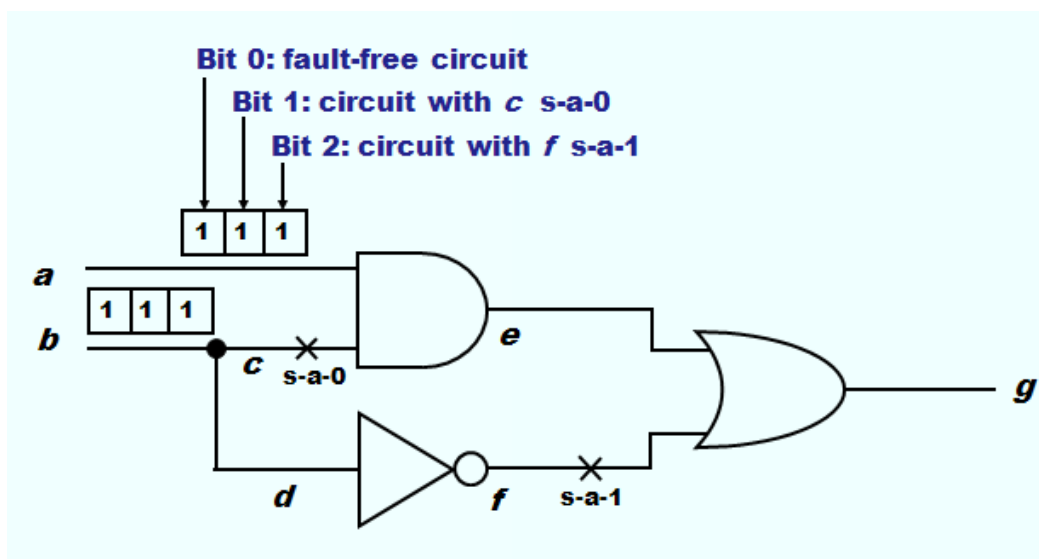
Σχήμα 3.15 Εξομοίωση για το σφάλμα $\theta/0$ και το διάνυσμα εισόδου 01010.

Η παράλληλη εξομοίωση σφαλμάτων είναι μια επέκταση της σειριακής εξομοίωσης. Στη σειριακή εξομοίωση σφαλμάτων μόνο ένα δυαδικό ψηφίο καθενός καταχωρητή χρησιμοποιείται, ενώ τα υπόλοιπα μένουν ανεκμετάλλευτα. Εάν τα υπόλοιπα δυαδικά ψηφία των καταχωρητών χρησιμοποιηθούν για την εξομοίωση άλλων απλών σφαλμάτων μόνιμης τιμής, μια απλή λογική πράξη μπορεί ταυτόχρονα να επεξεργαστεί όλα τα εξομοιωθέντα σφάλματα και να εξομοιώσει ταυτόχρονα και τη σωστή λειτουργία του κυκλώματος. Αυτή η χρήση όλων των δυαδικών ψηφίων των καταχωρητών ενός γενικού σκοπού υπολογιστή είναι το διαφοροποιό χαρακτηριστικό της παράλληλης εξομοίωσης σφαλμάτων. Η παράλληλη εξομοίωση σφαλμάτων είναι μια παλιά μέθοδος και έχει γίνει ευρέως αποδεκτή σε συστήματα αυτόματης εξαγωγής διανυσμάτων δοκιμής (Automated Test pattern Generation, ATG, systems). Αντιπροσωπευτικά συστήματα που χρησιμοποίησαν αυτή την

τεχνική είναι ο Sequential Analyzer, ο εξομοιωτής της IBM που χρησιμοποιήθηκε στο σχεδιασμό του υπολογιστή Saturn και ο εξομοιωτής TEGAS.

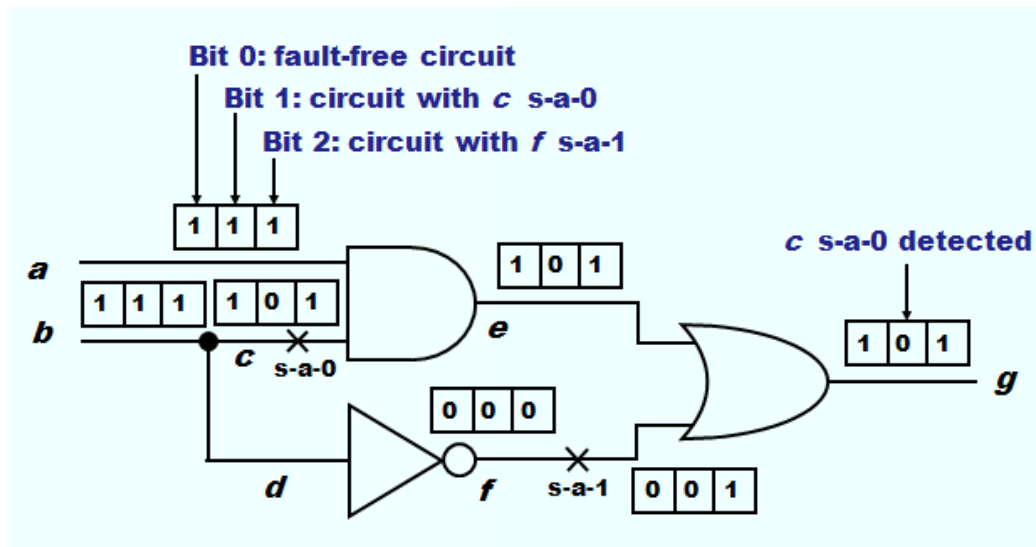
Στην παράλληλη εξομοίωση σφαλμάτων εάν μια λέξη έχει K δυαδικά ψηφία, τότε K διαφορετικά προβλήματα μπορούν να επεξεργαστούν παράλληλα. Εάν θέλουμε να εξομοιώσουμε ένα κύκλωμα για M διαφορετικά σφάλματα τότε πρέπει να εκτελεστούν $\lceil M/K \rceil$ περάσματα, εκτελέσεις του προγράμματος, όπου σε κάθε πέρασμα εξομοιώνονται K σφάλματα.

Στο σχήμα 3.16 θεωρούμε ότι οι καταχωρητές του υπολογιστή είναι των τριών δυαδικών ψηφίων. Χρησιμοποιούμε το πιο σημαντικό δυαδικό ψηφίο του καταχωρητή για να δηλώσουμε την τιμή του κυκλώματος χωρίς σφάλμα, το επόμενο λιγότερο σημαντικό δυαδικό ψηφίο για να δηλώσουμε την τιμή του κυκλώματος στην περίπτωση σφάλματος μόνιμης τιμής 0 στη γραμμή C , και το λιγότερο σημαντικό δυαδικό ψηφίο για να δηλώσουμε την τιμή του κυκλώματος στην περίπτωση απλού σφάλματος μόνιμης τιμής 1 στη γραμμή f .



Σχήμα 3.16 Παράδειγμα παράλληλης εξομοίωσης

Επομένως με κάθε κόμβο του κυκλώματος συνδέεται μια λέξη των τριών δυαδικών ψηφίων. Όλα τα δυαδικά ψηφία μιας λέξης που συνδέεται με μια κύρια είσοδο του κυκλώματος λαμβάνουν την τιμή με την οποία θέλουμε να οδηγήσουμε την είσοδο. Στο κύκλωμα του σχήματος θέλουμε να οδηγήσουμε τις εισόδους a και b με τη λογική τιμή 1, επομένως και τα τρία δυαδικά ψηφία των λέξεων που συνδέονται με τις εισόδους λαμβάνουν την τιμή 1.

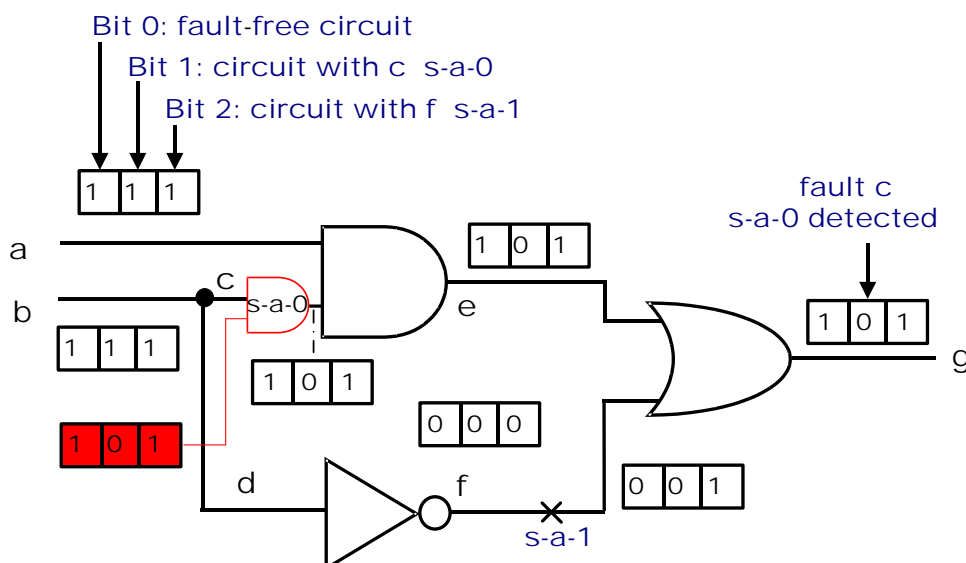


Σχήμα 3.17 Παράδειγμα παράλληλης εξομοίωσης

Στο σχήμα 3.17 βλέπουμε τις τιμές των δυαδικών ψηφίων που πρέπει να έχει κάθε μια λέξη που συνδέεται με ένα κόμβο του κυκλώματος.

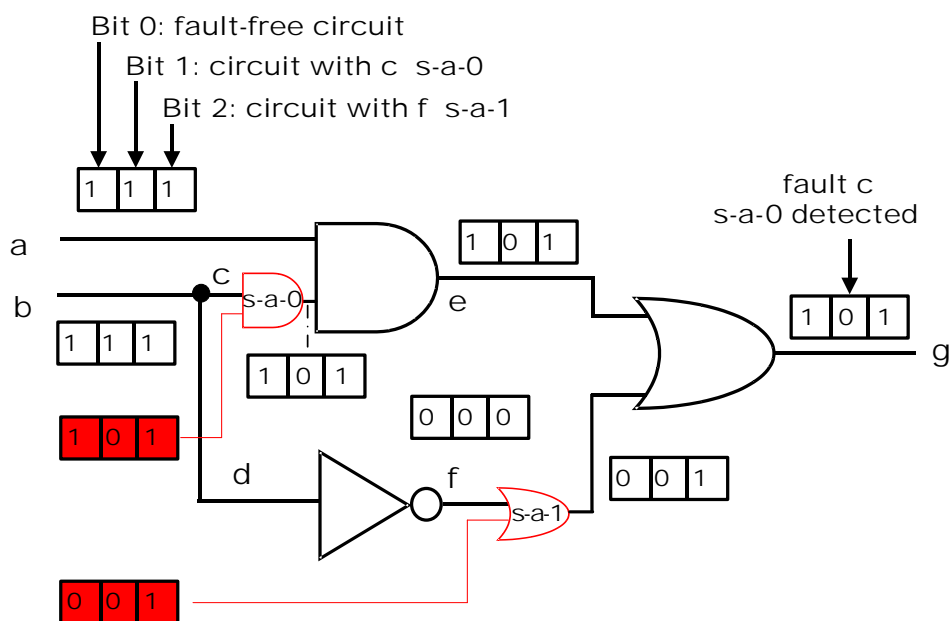
Για να πάρει η λέξη που συνδέεται με τον κόμβο C την τιμή 101 δεν θα πρέπει με κάποιο τρόπο να επιβάλουμε στο τρόπο λειτουργίας του κυκλώματος το συμβάν του σφάλματος. Αυτή η διαδικασία καλείται εισαγωγή σφάλματος προς εξομοίωση (fault injection).

Όταν θέλουμε να εξομοιώσουμε το συμβάν ενός απλού σφάλματος μόνιμης τιμής στη λογική τιμή 0, τότε στον κόμβο του σφάλματος παρεμβάλλουμε μια πύλη AND δύο εισόδων. Η μία είσοδος οδηγείται κανονικά από τον κόμβο του κυκλώματος και η άλλη από μια πρόσθετη είσοδο στην οποία το δυαδικό ψηφίο που αντιστοιχεί στο σφάλμα λαμβάνει την τιμή του σφάλματος και τα υπόλοιπα δυαδικά ψηφία τη λογική τιμή 1, δηλαδή την τιμή που δεν είναι τιμή ελέγχου της πύλης AND (βλέπε σχήμα 3.18.α).



Σχήμα 3.18.α Διαδικασία εισαγωγής σφάλματος S/0 προς εξομοίωση

Όταν θέλουμε να εξομοιώσουμε το συμβάν ενός απλού σφάλματος μόνιμης τιμής στη λογική τιμή 1, τότε στον κόμβο του σφάλματος παρεμβάλλουμε μια πύλη OR δύο εισόδων. Η μία είσοδος οδηγείται κανονικά από τον κόμβο του κυκλώματος και η άλλη από μια πρόσθετη είσοδο στην οποία το δυαδικό ψηφίο που αντιστοιχεί στο σφάλμα λαμβάνει την τιμή του σφάλματος και τα υπόλοιπα δυαδικά ψηφία τη λογική τιμή 0, δηλαδή την τιμή που δεν είναι τιμή ελέγχου της πύλης OR (βλέπε σχήμα 3.18.β).



Σχήμα 3.18.β Διαδικασία εισαγωγής σφάλματος S/1 προς εξομοίωση

Θα πρέπει να σημειώσουμε ότι στην ανάλυση της παράλληλης εξομοίωσης ακολουθήσαμε την αντίστροφη πορεία για να εξηγήσουμε την αναγκαιότητα της διαδικασίας εισαγωγής σφαλμάτων προς εξομοίωση (fault injection). Στην πραγματικότητα όταν πρόκειται να γίνει εξομοίωση σφαλμάτων για να διαπιστωθεί αν ανιχνεύονται κάποια σφάλματα πρώτα γίνεται η εισαγωγή των προς εξομοίωση σφαλμάτων, στη συνέχεια τίθενται στις εισόδους του κυκλώματος οι λέξεις με τις απαιτούμενες τιμές και εκτελείται λογική εξομοίωση.

Σύμφωνα με την επαγωγική εξομοίωση σφαλμάτων για κάθε συνδυασμό εισόδων εξομοιώνεται ρητά μόνο η συμπεριφορά του λογικού κυκλώματος χωρίς σφάλματα και ταυτόχρονα επάγονται από την τρέχουσα σωστή κατάσταση του κυκλώματος όλα τα σφάλματα, σε οποιαδήποτε εσωτερική ή εξωτερική γραμμή του κυκλώματος, που είναι ανιχνεύσιμα από τον συγκεκριμένο συνδυασμό εισόδων. Χρησιμοποιώντας την ιδέα της επαγωγικής εξομοίωσης, μπορούμε να βρούμε όλα τα ανιχνεύσιμα σφάλματα σε ένα βήμα, δηλαδή χρειάζεται να εκτελέσουμε μόνο μια εξομοίωση, για κάθε διάνυσμα δοκιμής.

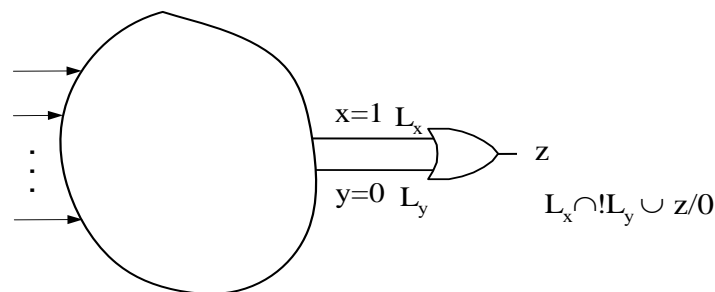
Στην παράλληλη εξομοίωση σφαλμάτων εάν M είναι ο συνολικός αριθμός των σφαλμάτων που πρέπει να εξομοιωθούν και σε κάθε πέρασμα εξομοιώνονται N σφάλματα, τότε για κάθε διάνυσμα δοκιμής απαιτούνται $\lceil M/N \rceil$ περάσματα. Ο χρόνος εξομοίωσης σφαλμάτων σε κάθε πέρασμα στην περίπτωση της επαγωγικής εξομοίωσης σφαλμάτων είναι πιθανό να είναι αρκετά μεγαλύτερος από τον χρόνο που απαιτείται για ένα πέρασμα στην παράλληλη εξομοίωση σφαλμάτων. Εν τούτοις,

αναμένεται να είναι μικρότερος από τον χρόνο για $\lceil M/N \rceil$ περάσματα που απαιτούνται στην παράλληλη εξομοίωση για να παραχθούν τα ίδια αποτελέσματα. Πράγματι έχει δειχθεί πειραματικά ότι η επαγωγική εξομοίωση είναι γρηγορότερη από την παράλληλη εξομοίωση όταν πρέπει να εξομοιωθεί ένα μεγάλο κύκλωμα με μεγάλο αριθμό σφαλμάτων. Η παράλληλη εξομοίωση είναι γρηγορότερη της επαγωγικής μόνον για μικρά (π.χ. < 1000 πυλών) κυκλώματα. Όμως οι συγκρίσεις δείχνουν ότι ένας παράλληλος εξομοιωτής μπορεί να σχεδιαστεί να απαιτεί λιγότερη μνήμη από ένα επαγωγικό εξομοιωτή σφαλμάτων.

Ένας επαγωγικός εξομοιωτής σφαλμάτων εξομοιώνει το χωρίς σφάλματα κύκλωμα και υπολογίζει λίστες σφαλμάτων. Μια λίστα σφαλμάτων L_A που συνδέεται με την γραμμή A περιέχει το όνομα ή το δείκτη κάθε σφάλματος που παράγει ένα λάθος στη γραμμή A όταν το κύκλωμα είναι στην τρέχουσα λογική κατάσταση. Δηλαδή, κάθε σφάλμα στη λίστα θα έχει ως αποτέλεσμα την αλλαγή της σωστής τιμής του σήματος της γραμμής. Αυτές οι λίστες σφαλμάτων διαδίδονται από τις κύριες εισόδους προς τις κύριες εξόδους, επίπεδο προς επίπεδο, διαμέσου του κυκλώματος. Με αυτό τον τρόπο, μια λίστα σφαλμάτων παράγεται για κάθε γραμμή του κυκλώματος και ενημερώνεται όταν είναι αναγκαίο σε κάθε αλλαγή της λογικής κατάστασης του κυκλώματος.

Στη συνέχεια θα δούμε τη διαδικασία υπολογισμού λίστες σφαλμάτων στην περίπτωση λογικών πυλών.

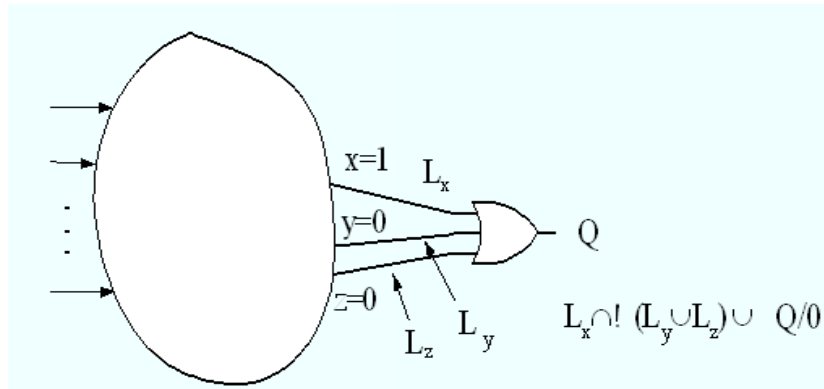
Θεωρούμε ένα κύκλωμα, σχήμα 3.19, και ότι στις εισόδους x και y μιας πύλης OR με $x=1$ και $y=0$ έχουν διαδοθεί οι λίστες σφαλμάτων L_x και L_y αντίστοιχα. Είναι προφανές ότι στην έξοδο της πύλης OR μπορεί να διαδοθεί κάποιο σφάλμα μόνο της λίστας L_x και υπό την προϋπόθεση ότι δεν ανήκει στην λίστα L_y . Επομένως στην λίστα σφαλμάτων L_z ανήκουν τα σφάλματα που ανήκουν στη λίστα σφαλμάτων L_x και δεν ανήκουν στη λίστα L_y και το απλό σφάλμα μόνιμης τιμής της γραμμής z στο 0, $z/0$.



Σχήμα 3.19 Διάδοση λίστες σφαλμάτων μέσω πύλης OR δύο εισόδων.

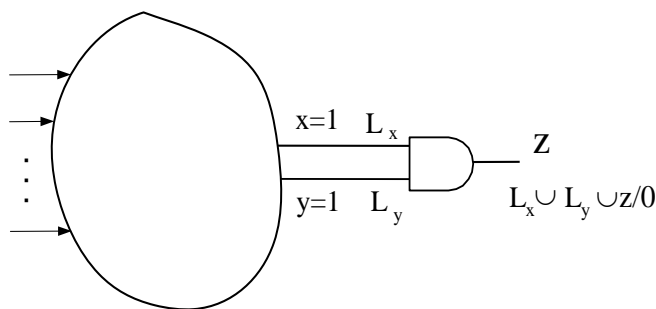
Θεωρούμε ένα κύκλωμα, σχήμα 3.20, και ότι στις εισόδους x , y και z μιας πύλης OR με $x=1$, $y=0$ και $z=0$ έχουν διαδοθεί οι λίστες σφαλμάτων L_x , L_y και L_z αντίστοιχα. Είναι προφανές ότι στην έξοδο της πύλης OR μπορεί να διαδοθεί κάποιο σφάλμα μόνο της λίστας L_x και υπό την προϋπόθεση ότι δεν ανήκει στη λίστα L_y ή στη λίστα L_z .

Επομένως στην λίστα σφαλμάτων L_Q ανήκουν τα σφάλματα που ανήκουν στη λίστα σφαλμάτων L_x και δεν ανήκουν στη λίστα L_y και L_z , και το απλό σφάλμα μόνιμης τιμής της γραμμής Q στο 0, $Q/0$.



Σχήμα 3.20 Διάδοση λίστας σφαλμάτων μέσω πύλης OR δύο εισόδων.

Θεωρούμε ένα κύκλωμα, σχήμα 3.21 και ότι στις εισόδους x και y μιας πύλης AND με $x=1$ και $y=1$ έχουν διαδοθεί οι λίστες σφαλμάτων L_x και L_y αντίστοιχα. Είναι προφανές ότι στην έξοδο της πύλης AND μπορεί να διαδοθεί κάποιο σφάλμα το οποίο ανήκει είτε στη λίστα L_x είτε στη λίστα L_y είτε και στις δύο. Στο σχήμα 3.21 βλέπουμε τα σφάλματα τα οποία ανήκουν στη λίστα σφαλμάτων L_z .



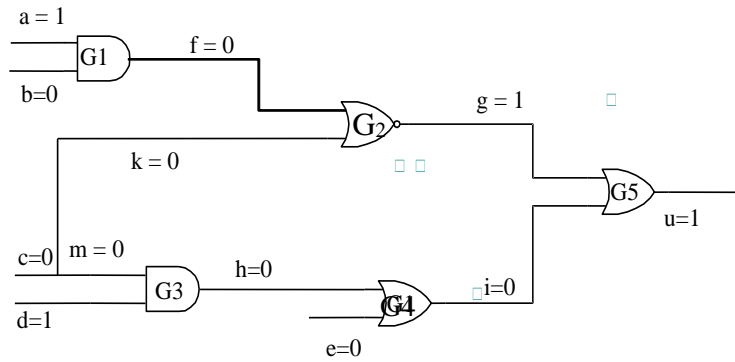
Σχήμα 3.21 Διάδοση λίστας σφαλμάτων μέσω πύλης AND δύο εισόδων.

Ο επαγωγικός εξομοιωτής χρησιμοποιεί τη διαδικασία της επιλεκτικής διέγερσης (event-driven), δηλαδή η κατάσταση εξόδου ενός λογικού στοιχείου υπολογίζεται μόνον όταν μια αλλαγή (event) έχει συμβεί σε μια από τις εισόδους του. Στην επαγωγική εξομοίωση σφαλμάτων υπάρχουν δυο είδη αλλαγών που οδηγούν στην επανάληψη υπολογισμών:

- λογικές αλλαγές (logic events) που συμβαίνουν όταν μια γραμμή του κυκλώματος αλλάζει τιμή, και
- αλλαγές στη λίστα (list events) που συμβαίνουν όταν αλλάζει το περιεχόμενο μιας λίστας σφαλμάτων.

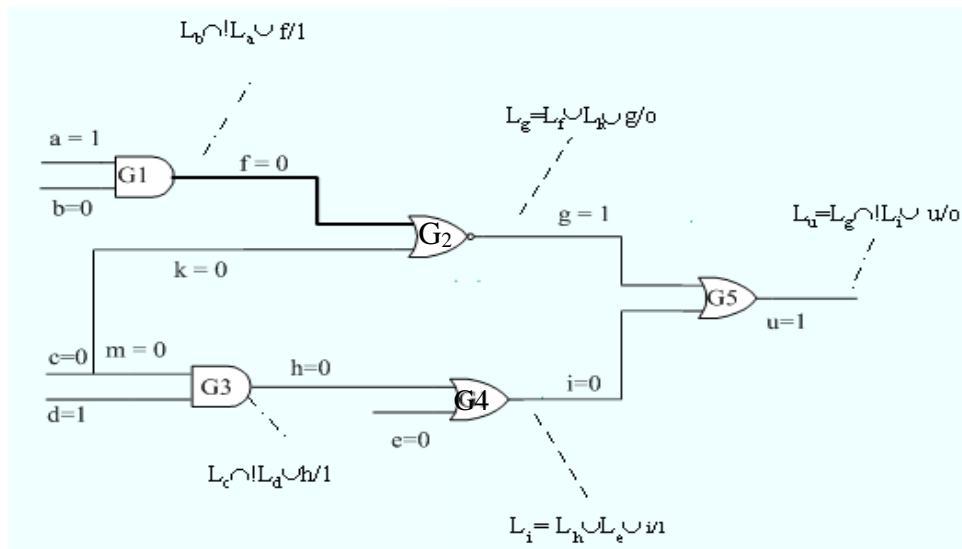
Ο εξομοιωτής πρέπει να ξαναυπολογίσει τη λίστα σφαλμάτων που συνδέεται με την έξοδο μιας πύλης όταν μια αλλαγή λίστας συμβαίνει σε κάποια από τις εισόδους της πύλης, ακόμη και εάν δεν έχει συμβεί καμία λογική αλλαγή σε κάποια από τις εισόδους της πύλης.

Θεωρούμε το λογικό κύκλωμα του σχήματος 3.22 και τις τιμές 1001 στις εισόδους του. Με λογική εξομοίωση προκύπτουν οι τιμές σε όλους τους κόμβους του κυκλώματος.



Σχήμα 3.22 Κύκλωμα για εξομοίωση σφαλμάτων

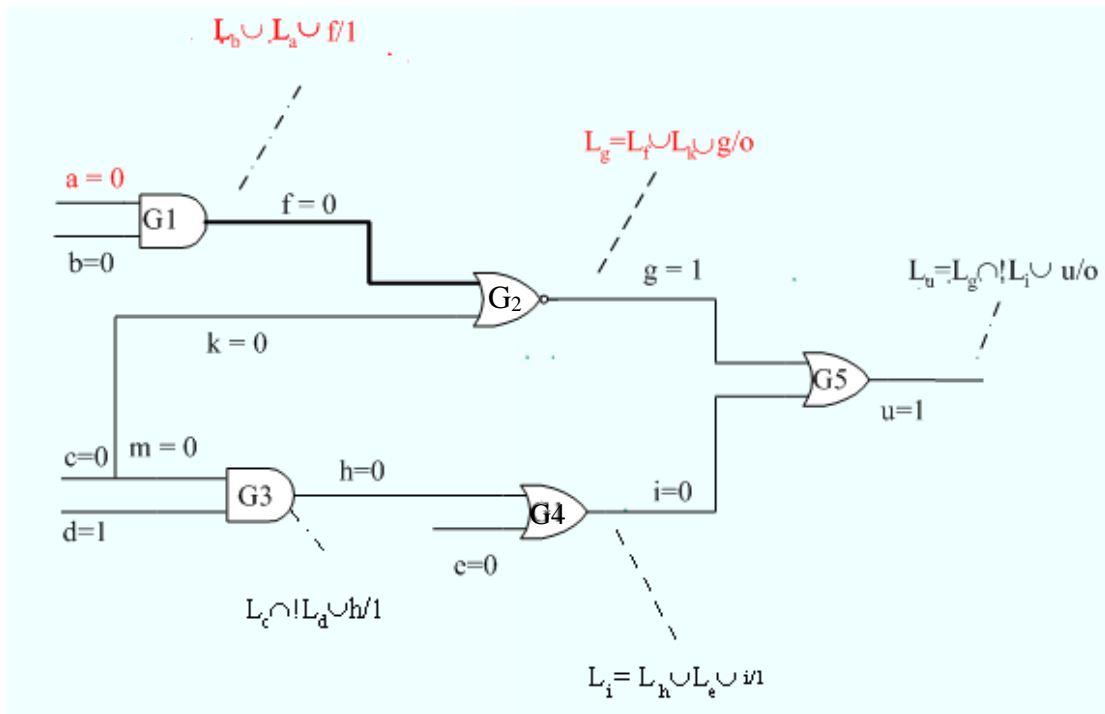
Ξεκινώντας από τις κύριες εισόδους του κυκλώματος υπολογίζουμε τη λίστα σφαλμάτων στην έξοδο κάθε μιας πύλης του κυκλώματος. Οι λίστες δίνονται στο σχήμα 3.23.



Σχήμα 3.23 Επαγωγική εξομοίωση σφαλμάτων.

Μια είσοδος του κυκλώματος, η a , αλλάζει τιμή από 1 σε 0. Αφού έχουμε αλλαγή της λογικής τιμής μιας εισόδου της πύλης G1, θα πρέπει να ξανά υπολογίσουμε τη λίστα σφαλμάτων στην έξοδο αυτής της πύλης (βλέπε σχήμα 3.24).

Παρατηρούμε ότι η λίστα που συνδέεται στην έξοδο της πύλης G1 αλλάζει. Αφού έχουμε αλλαγή της λίστας σφαλμάτων μιας εισόδου της πύλης G2, θα πρέπει να ξανά υπολογίσουμε τη λίστα σφαλμάτων στην έξοδο αυτής της πύλης. Εάν η λίστα στην έξοδο της πύλης G2 αλλάξει τότε θα πρέπει να ξανά υπολογίσουμε τη λίστα εξόδου της πύλης G5, διαφορετικά η εξομοίωση σφαλμάτων ολοκληρώθηκε.



Σχήμα 3.24 Διάδοση λίστας σφαλμάτων μέσω πύλης AND δύο εισόδων.

Στην σύνδρομη εξομοίωση σφαλμάτων όπως και στην επαγωγική, αρκεί εξομοίωση ενός βήματος για να εξαχθούν όλα τα σφάλματα που ανιχνεύονται από ένα συγκεκριμένο διάνυσμα δοκιμής. Συνήθως η συμπεριφορά ενός κυκλώματος με κάποιο σφάλμα διαφέρει σε μικρό μόνο βαθμό από την συμπεριφορά του κυκλώματος χωρίς σφάλμα. Οι λογικές τιμές στις γραμμές ενός κυκλώματος χωρίς σφάλμα και στο ίδιο κύκλωμα με σφάλμα είναι μερικές φορές ταυτοτικές, τον περισσότερο χρόνο σχεδόν ταυτοτικές και μόνον σπάνια σημαντικά διαφορετικές.

Η σύνδρομη εξομοίωση σφαλμάτων, η οποία βασίζεται ακριβώς σ' αυτή την ομοιότητα της συμπεριφοράς του κυκλώματος χωρίς σφάλμα και του κυκλώματος με σφάλμα, συνίσταται στην εξομοίωση του κυκλώματος χωρίς σφάλμα και ταυτόχρονα στην εξομοίωση του κυκλώματος με σφάλμα μόνον εάν η δραστηριότητα του κυκλώματος με σφάλμα διαφέρει πραγματικά από εκείνη χωρίς σφάλμα. Υπάρχουν κάποιες ομοιότητες μεταξύ επαγωγικής και σύνδρομης εξομοίωσης.

Ο εξομοιωτής σφαλμάτων δίνει ως αποτέλεσμα τα σφάλματα τα οποία είναι ανιχνεύσιμα με το σύνολο δοκιμής που τον τροφοδοτήσαμε, τα σφάλματα που δεν είναι ανιχνεύσιμα και τα σφάλματα που είναι πιθανώς ανιχνεύσιμα (potentially detected). Θα πρέπει να σχολιάσουμε τι εννοούμε όταν λέμε πιθανώς ανιχνεύσιμα σφάλματα. Μερικές φορές η απόφαση σχετικά με το αν ένα σφάλμα είναι ανιχνεύσιμο ή όχι είναι δύσκολη.

Για παράδειγμα, θεωρήστε ένα απλό σφάλμα μόνιμης τιμής στη λογική τιμή 0 το οποίο συμβαίνει στην είσοδο ενεργοποίησης ενός στοιχείου τριών καταστάσεων (tristate buffer). Εφόσον η είσοδος ενεργοποίησης είναι στη λογική τιμή 0, η έξοδος του στοιχείου τριών καταστάσεων είναι στον αέρα. Δεν είναι σαφές εάν το σφάλμα είναι ανιχνεύσιμο, διότι η λογική τιμή της γραμμής που βρίσκεται στον αέρα μπορεί τυχαία να είναι ίδια με την αναμενόμενη τιμή. Ωστόσο, εάν το σφάλμα εξομοιώνεται για πολλά διανύσματα δοκιμής, είναι πιθανόν τελικά να ανιχνευθεί. Γι' αυτό το λόγο πολλοί εξομοιωτές θεωρούν αυτό το είδος σφάλματος ως πιθανώς ανιχνεύσιμο.

Η κάλυψη σφαλμάτων ορίζεται ως ο λόγος των σφαλμάτων που ανιχνεύονται προς το συνολικό αριθμό σφαλμάτων ή ως ο λόγος των σφαλμάτων που ανιχνεύονται προς

τα ανιχνεύσιμα σφάλματα. Θα πρέπει να είμαστε προσεκτικοί γιατί ο εξομοιωτής σφαλμάτων μπορεί αντί για σφάλματα να αναφέρεται σε κλάσεις σφαλμάτων, δηλαδή, σε σύνολα από ισοδύναμα σφάλματα. Επίσης πρέπει να έχουμε πάντα κατά νου ότι 100% κάλυψη σφαλμάτων για απλά σφάλματα μόνιμης τιμής δεν συνεπάγεται ότι το κύκλωμα δεν έχει κανένα ελάττωμα.

Λεξικό σφαλμάτων								
Διάνυσμα δοκιμής	σφάλμα							
	1	2	3	4	5	6	7	8
1	1		1			1		
2	1	1	1					
3		1		1		1		
4		1				1		1

Σχήμα 3.25 Λεξικό σφαλμάτων.

Με την εξομοίωση σφαλμάτων, χωρίς απόρριψη των σφαλμάτων που ανιχνεύονται από κάποιο διάνυσμα δοκιμής (fault dropping), μπορούμε να κατασκευάσουμε λεξικά σφαλμάτων τα οποία μας βοηθούν στη διάγνωση των σφαλμάτων του κυκλώματος. Στον πίνακα βλέπουμε ως παράδειγμα ένα λεξικό σφαλμάτων για τέσσερα διανύσματα δοκιμής και οκτώ σφάλματα. Εάν κατά τον έλεγχο της ορθής λειτουργίας ενός ολοκληρωμένου κυκλώματος μόνο τα διανύσματα δοκιμής 2, 3 και 4 ανιχνεύουν σφάλμα, τότε βάσει του λεξικού σφαλμάτων του πίνακα στο υπό έλεγχο κύκλωμα υπάρχει το σφάλμα νούμερο 2.

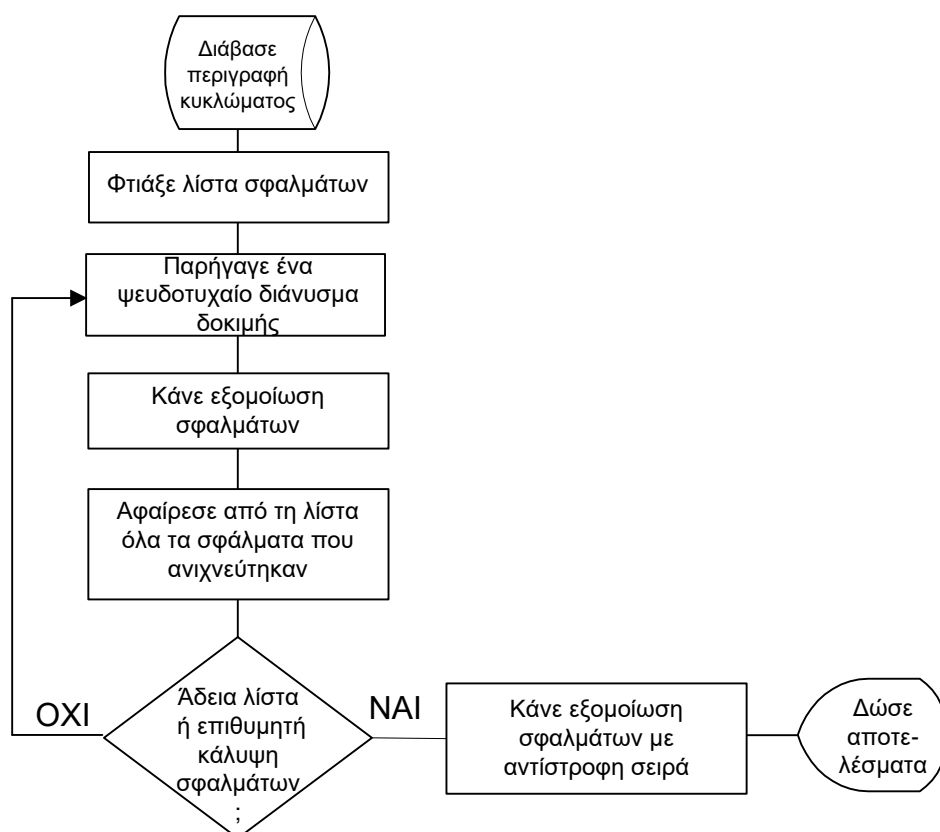
Κεφάλαιο 4. Εξαγωγή Διανυσμάτων Δοκιμής

Τα σύνολα δοκιμής, ανάλογα της πληροφορίας που λαμβάνουμε υπόψη για την εξαγωγή τους, μπορούμε να τα διακρίνουμε σε δύο μεγάλες κατηγορίες. Τα σύνολα δοκιμής για την εξαγωγή των οποίων δεν λαμβάνουμε υπόψη κάποιο συγκεκριμένο μοντέλο σφαλμάτων και αυτά για την εξαγωγή των οποίων πρέπει να λάβουμε υπόψη μας ένα συγκεκριμένο μοντέλο σφαλμάτων.

Στην πρώτη κατηγορία ανήκουν τα σύνολα δοκιμής που χρησιμοποιούμε στον εξαντλητικό και στον ψευδοεξαντλητικό έλεγχο. Στον εξαντλητικό έλεγχο το σύνολο δοκιμής αποτελείται από όλα τα δυνατά διανύσματα εισόδου. Το μεγάλο πλεονέκτημα αυτού του ελέγχου είναι ότι οποιοδήποτε ελάττωμα μπορεί να ανιχνευτεί θα ανιχνευτεί, δηλαδή οποιοδήποτε ελάττωμα επηρεάζει τη σωστή λειτουργία κυκλώματος θα ανιχνευτεί. Το πρόβλημα είναι ότι αυτό το είδος ελέγχου δεν μπορεί να εφαρμοστεί σε κυκλώματα με μεγάλο αριθμό εισόδων.

Στον ψευδοεξαντλητικό έλεγχο χωρίζουμε το κύκλωμα σε υποκυκλώματα και ελέγχουμε εξαντλητικά την ορθή λειτουργία κάθε υποκυκλώματος.

Στην κατηγορία των συνόλων δοκιμής για την εξαγωγή των οποίων λαμβάνουμε υπόψη κάποιο συγκεκριμένο μοντέλο σφαλμάτων ανήκουν τα τυχαία σύνολα δοκιμής, τα ψευδοτυχαία σύνολα δοκιμής και τα ντετερμινιστικά σύνολα δοκιμής.



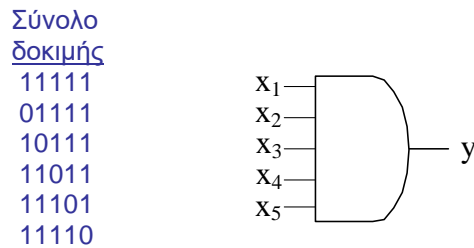
Σχήμα 4.1 Σύνολο δοκιμής ψευδοτυχαίων διανυσμάτων.

Στο παραπάνω σχήμα, μπορούμε να δούμε τα βήματα που ακολουθούνται για την εξαγωγή ενός τυχαίου ή ψευδοτυχαίου συνόλου δοκιμής. Καταρχήν έχουμε μια περιγραφή του κυκλώματος σε επίπεδο λογικών πυλών. Από την περιγραφή αυτή του κυκλώματος εξάγουμε μια λίστα με όλα τα σφάλματα για τα οποία πρέπει να βρούμε διανύσματα δοκιμής.

Στη συνέχεια παράγουμε τυχαία ή ψευδοτυχαία ένα διάνυσμα δοκιμής και κάνουμε εξομοίωση σφαλμάτων. Αφαιρούμε από τη λίστα σφαλμάτων, τα σφάλματα που ανιχνεύονται από το διάνυσμα δοκιμής. Εάν δεν έχουμε πετύχει το ποσοστό κάλυψης σφαλμάτων, τότε παράγουμε τυχαία ή ψευδοτυχαία ένα διάνυσμα δοκιμής και επαναλαμβάνουμε τα ίδια βήματα. Όταν η λίστα σφαλμάτων θα είναι κενή ή θα έχουμε πετύχει το ζητούμενο ποσοστό κάλυψης σφαλμάτων τότε θα κάνουμε εξομοίωση σφαλμάτων αρχίζοντας από το τελευταίο διάνυσμα δοκιμής του συνόλου δοκιμής, δηλαδή το διάνυσμα δοκιμής που έχουμε πάρει τελευταίο. Εάν πριν φτάσουμε στο πρώτο διάνυσμα δοκιμής του συνόλου δοκιμής έχουμε πετύχει το επιθυμητό ποσοστό κάλυψης σφαλμάτων ή η λίστα σφαλμάτων είναι κενή, τότε απορρίπτουμε όλα τα διανύσματα δοκιμής που βρίσκονται μεταξύ του τελευταίου διανύσματος που κάναμε εξομοίωση και του πρώτου διανύσματος του συνόλου δοκιμής.

Πρέπει να κάνουμε δύο παρατηρήσεις όσον αφορά τη διαδικασία εξαγωγής διανυσμάτων δοκιμής που περιγράψαμε. Ένα διάνυσμα δοκιμής το οποίο εξάγεται τυχαία ή ψευδοτυχαία και με την εξομοίωση σφαλμάτων διαπιστώνεται ότι δεν

ανιχνεύει κανένα σφάλμα, δεν το προσθέτουμε στο σύνολο δοκιμής. Όταν τα διανύσματα δοκιμής παράγονται με τυχαίο τρόπο τότε είναι δυνατόν το ίδιο διάνυσμα να εμφανιστεί και δεύτερη και τρίτη η ακόμη και περισσότερες φορές. Για το διάνυσμα αυτό πρέπει να γίνεται εξομοίωση σφαλμάτων κάθε φορά που εμφανίζεται. Βέβαια μετά την πρώτη εμφάνιση του διανύσματος θα διαπιστώνεται ότι δεν ανιχνεύει κανένα πρόσθετο σφάλμα και το διάνυσμα θα απορρίπτεται. Ωστόσο, αυτές οι εξομοιώσεις σφαλμάτων οδηγούν σε αύξηση του χρόνου εξαγωγής του συνόλου δοκιμής ενός κυκλώματος. Αυτός είναι ο λόγος για τον οποίο δεν χρησιμοποιείται ο τυχαίος τρόπος εξαγωγής διανυσμάτων δοκιμής.

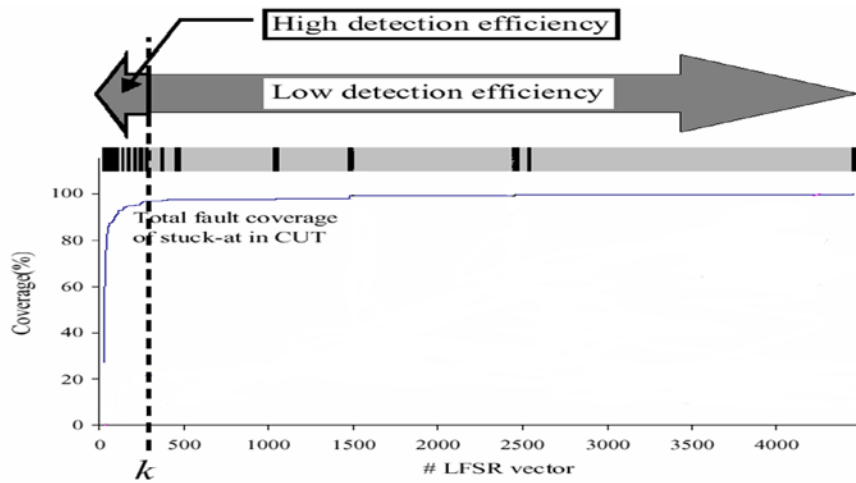


Σχήμα 4.2 Σφάλματα που δεν ανιχνεύονται εύκολα με ψευδοτυχαία διανύσματα.

Στο σχήμα 4.2 βλέπουμε μια πύλη AND με πέντε εισόδους. Το σύνολο δοκιμής αυτής της πύλης αποτελείται από τα έξι διανύσματα του σχήματος. Ωστόσο όλοι οι δυνατοί συνδυασμοί εισόδων της πύλης αυτής είναι 32 και επομένως η πιθανότητα να πάρουμε τυχαία ή ψευδοτυχαία ένα συγκεκριμένο διάνυσμα εισόδου είναι ένα προς 32. Μπορούμε επομένως να συμπεράνουμε ότι η πιθανότητα να εξάγουμε το σύνολο δοκιμής αυτής της πύλης παράγοντας έξι ψευδοτυχαία διανύσματα είναι ελάχιστη. Η πιθανότητα γίνεται πολύ μικρότερη αν λάβουμε υπόψη μας ότι η πύλη αυτή είναι εμφωλευμένη μέσα σε ένα κύκλωμα και εμείς δίνουμε τυχαίες ή ψευδοτυχαίες τιμές στις εισόδους του κυκλώματος.

Επομένως συμπεραίνουμε ότι στα κυκλώματα υπάρχουν σφάλματα τα οποία ανιχνεύονται δύσκολα με διανύσματα δοκιμής που εξάγονται με τυχαίο ή ψευδοτυχαίο τρόπο.

Στο σχήμα 4.3 δίνεται το ποσοστό κάλυψης σφαλμάτων ως συνάρτηση του πλήθους των διανυσμάτων δοκιμής που έχουν παραχθεί με ένα ψευδοτυχαίο τρόπο (στη συγκεκριμένη περίπτωση έχει χρησιμοποιηθεί ένα LFSR). Μία κάθετη μαύρη γραμμή ή μία μαύρη περιοχή δηλώνει ένα ή περισσότερα διανύσματα τα οποία ανιχνεύουν πρόσθετα σφάλματα, δηλαδή σφάλματα τα οποία δεν ανιχνεύονται από τα προηγούμενα διανύσματα δοκιμής. Μια γκρι περιοχή δηλώνει διανύσματα τα οποία δεν ανιχνεύουν πρόσθετα σφάλματα.



Σχήμα 4.3 Σχέση κάλυψης σφαλμάτων και μήκους της ακολουθίας ψευδοτυχαίων διανυσμάτων.

Παρατηρούμε ότι τα πρώτα διανύσματα που παράγονται ανιχνεύουν ένα μεγάλο αριθμό σφαλμάτων. Παρατηρούμε επίσης ότι για το συγκεκριμένο κύκλωμα ένας μικρός αριθμός από τα πρώτα διανύσματα που παράγονται με ψευδοτυχαίο τρόπο επιτυγχάνει ένα ποσοστό κάλυψης μεγαλύτερο του 85%. Στη συνέχεια όμως παρατηρούμε ότι χρειάζεται να παράγουμε μεγάλους αριθμούς από ψευδοτυχαία διανύσματα για να βρούμε κάποιο ή κάποια διανύσματα που ανιχνεύουν επιπλέον σφάλματα.

Τις ντετερμινιστικές μεθόδους εξαγωγής διανυσμάτων δοκιμής μπορούμε να τις διακρίνουμε σε δύο μεγάλες κατηγορίες: τις αλγεβρικές και τις αλγοριθμικές. Η πιο γνωστή αλγεβρική μέθοδος είναι αυτή που βασίζεται στη Boolean difference. Μεταξύ των αλγοριθμικών μεθόδων οι πιο γνωστές είναι ο αλγόριθμος, D, ο PODEM και ο FAN.

Υποθέστε ότι έχουμε ένα κύκλωμα με μια έξοδο και ότι R είναι η τιμή της εξόδου του κυκλώματος χωρίς σφάλμα και R_f είναι η τιμή της εξόδου του με σφάλμα. Προφανώς το σφάλμα ανιχνεύεται όταν $R \neq R_f$ ή ισοδύναμα όταν $R \oplus R_f = 1$. Εάν το σφάλμα είναι σε μια κύρια είσοδο του κυκλώματος τότε

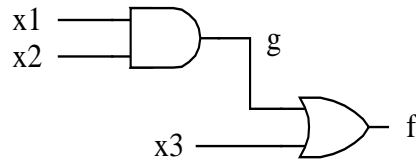
$$f(x_1, \dots, x_i = 0, \dots, x_n) \oplus f(x_1, \dots, x_i = 1, \dots, x_n) = 1$$

$$\text{ή } df(x)/dx_i$$

Η επίλυση της εξίσωσης δίνει τις τιμές των κυρίων εισόδων οι οποίες επιτρέπουν την παρατηρησιμότητα ενός απλού σφάλματος μόνιμης τιμής στην κύρια είσοδο x_i του κυκλώματος. Τα διανύσματα δοκιμής τότε δίνονται από την επίλυση των σχέσεων:

$$x_i df(x)/dx_i = 1 \text{ for s-a-0 και } x_i' df(x)/dx_i = 1 \text{ for s-a-1}$$

Θεωρούμε το κύκλωμα του σχήματος 4.4. Η λογική συνάρτηση που υλοποιεί είναι $f(x) = x_1x_2 + x_3$. Θα εξάγουμε διανύσματα δοκιμής για τα απλά σφάλματα μόνιμης τιμής στην είσοδο x_2 .



Σχήμα 4.4

Για να βρούμε τις τιμές των εισόδων του κυκλώματος οι οποίες επιτρέπουν τη διάδοση των συνεπειών του σφάλματος στη κύρια έξοδο του κυκλώματος, θα πρέπει να βρούμε τις τιμές που ικανοποιούν την εξίσωση $df(x)/dx_2 = x_3 \oplus (x_1 + x_3) = x_3'x_1 = 1$, από την εξίσωση αυτή λαμβάνουμε $x_1 = 1$ and $x_3 = 0$.

Τότε τα διανύσματα δοκιμής είναι:

για απλό σφάλμα μόνιμης τιμής στην τιμή 1, $x_2/1, x_1x_2x_3 = (100)$

και για απλό σφάλμα μόνιμης τιμής στην τιμή 0, $x_2/0, x_1x_2x_3 = (110)$

Στη συνέχεια θα εξάγουμε διανύσματα δοκιμής για τα απλά σφάλματα μόνιμης τιμής στην είσοδο x_3 . Για να βρούμε τις τιμές των εισόδων του κυκλώματος οι οποίες επιτρέπουν τη διάδοση των συνεπειών του σφάλματος στη κύρια έξοδο του κυκλώματος, θα πρέπει να βρούμε τις τιμές που ικανοποιούν την εξίσωση $df(x)/dx_3 = x_1x_2 \oplus 1 = (x_1x_2)' = 1$. Από την εξίσωση αυτή λαμβάνουμε $x_1x_2 = (00)$ ή (01) ή (10)

Τότε τα διανύσματα δοκιμής είναι:

για απλό σφάλμα μόνιμης τιμής στην τιμή 1, $x_3/1, x_1x_2x_3 = (000)$ ή (010) ή (100)

και για απλό σφάλμα μόνιμης τιμής στην τιμή 0, $x_3/0, x_1x_2x_3 = (001)$ ή (011) ή (101)

Τα προηγούμενα παραδείγματα αφορούσαν την εξαγωγή διανυσμάτων δοκιμής για την ανίχνευση σφαλμάτων απλής μόνιμης τιμής σε κάποια από τις κύριες εισόδους του κυκλώματος. Τώρα θα δούμε ένα παράδειγμα εξαγωγής διανυσμάτων δοκιμής για την ανίχνευση σφαλμάτων απλής μόνιμης τιμής σε κάποιο εσωτερικό κόμβο του κυκλώματος. Δουλεύουμε στο παραπάνω κύκλωμα και ας υποθέσουμε ότι θέλουμε να εξάγουμε διανύσματα δοκιμής για την ανίχνευση των απλών σφαλμάτων μόνιμης τιμής στον κόμβο g .

Για να βρούμε τις τιμές των εισόδων του κυκλώματος οι οποίες επιτρέπουν τη διάδοση των συνεπειών του σφάλματος στη κύρια έξοδο του κυκλώματος, θα πρέπει να βρούμε τις τιμές που ικανοποιούν την εξίσωση $df(x, g(x))/dg(x) = x_3 \oplus 1 = x_3' = 1$. Από την εξίσωση αυτή λαμβάνουμε $x_3 = 0$. Για να διεγείρουμε το σφάλμα $g/0$ θα πρέπει να θέσουμε τον κόμβο g στην τιμή 1. Αυτό επιτυγχάνεται θέτοντας $x_1x_2 = 1$ οπότε $x_1 = x_2 = 1$. Για να διεγείρουμε το σφάλμα $g/1$ θα πρέπει να θέσουμε τον κόμβο g στην τιμή 0. Αυτό επιτυγχάνεται θέτοντας $x_1x_2 = 0$, οπότε $(x_1x_2) = (00)$ or (01) or (10) . Τότε θα διανύσματα δοκιμής είναι:

για $g/0, x_1x_2x_3 = (110)$ και για $g/1, x_1x_2x_3 = (000)$ ή (010) ή (100) .

Στη συνέχεια θα μελετήσουμε αλγοριθμικούς τρόπους εξαγωγής διανυσμάτων δοκιμής. Συγκεκριμένα θα μελετήσουμε τον αλγόριθμο D, τον PODEM και τον FAN.

Σε όλους τους αλγοριθμικούς τρόπους εξαγωγής διανυσμάτων δοκιμής χρησιμοποιούνται οι ακόλουθες πέντε βασικές λειτουργίες:

- Ενεργοποίηση σφάλματος (controllability)
- Διάδοση σφάλματος (observability)
- Υποστήριξη τιμής μιας γραμμής (line justification)
- Συνεπαγωγή (implication)

- Αναδρομή (Backtracking)

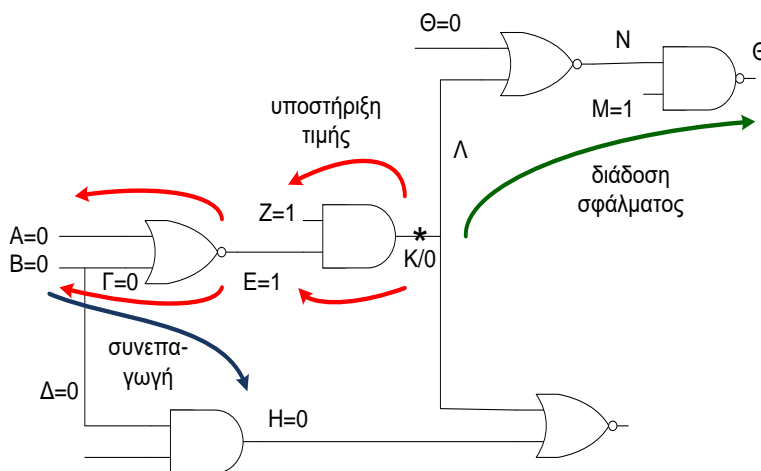
Για να ενεργοποιήσουμε ένα σφάλμα στον κόμβο X θα πρέπει να θέσουμε το κόμβο στην αντίθετη τιμή από την τιμή του σφάλματος. Ως παράδειγμα αναφέρουμε ότι για να ενεργοποιήσουμε το σφάλμα X/1 θα πρέπει να θέσουμε τον κόμβο X στην τιμή 0. Η ενεργοποίηση του σφάλματος γίνεται θέτοντας στις κατάλληλες τιμές κύριες εισόδους του κυκλώματος.

Θέτοντας κατάλληλες τιμές σε κύριες εισόδους του κυκλώματος είναι δυνατόν να διαδώσουμε τις συνέπειες του σφάλματος από τον κόμβο που συνέβη σε κάποια από τις κύριες εξόδους του κυκλώματος. Αυτό λέγεται διάδοση σφάλματος.

Κατά τη διαδικασία εξαγωγής ενός διανύσματος δοκιμής είναι δυνατόν να θέλουμε η έξοδος μιας πύλης να λάβει μια συγκεκριμένη τιμή. Για να το πετύχουμε αυτό θα πρέπει να θέσουμε μια ή περισσότερες εισόδους της πύλης σε συγκεκριμένες τιμές. Η διαδικασία αυτή καλείται υποστήριξη τιμής.

Όταν θέτουμε την είσοδο μιας πύλης στην τιμή ελέγχου της (πχ. η λογική τιμή 0 για μία πύλη AND και η λογική τιμή 1 για μια πύλη OR) τότε η έξοδος της λαμβάνει μια συγκεκριμένη τιμή. Όταν δώσουμε μια λογική τιμή σε ένα κόμβο ενός κυκλώματος τότε είναι δυνατόν οι έξοδοι μιας ή περισσότερων πυλών να πάρουν κάποια τιμή. Η διαδικασία αυτή καλείται συνεπαγωγή.

Είναι δυνατόν δίνοντας μια λογική τιμή σε ένα κόμβο του κυκλώματος ένας άλλος κόμβος του κυκλώματος να πάρει την αντίθετη από την επιθυμητή τιμή. Τότε λέμε ότι έχουμε σύγκρουση και πρέπει να γυρίσουμε πίσω, δηλαδή να κάνουμε αναδρομή, και να αλλάξουμε κάποια απόφαση που είχαμε πάρει.



Σχήμα 4.5 Βασικές λειτουργίες στην αλγοριθμική εξαγωγή διανυσμάτων δοκιμής

Στη συνέχεια θα χρησιμοποιούμε την ακόλουθη ορολογία.

- D → ο κόμβος έχει τη λογική τιμή 1 όταν το κύκλωμα δεν έχει σφάλμα και τη λογική τιμή 0 όταν έχει σφάλμα.
- D' → ο κόμβος έχει τη λογική τιμή 0 όταν το κύκλωμα δεν έχει σφάλμα και τη λογική τιμή 1 όταν έχει σφάλμα.

Έως τώρα οι κόμβοι του κυκλώματος αρχικά είχαν την άγνωστη τιμή X και στη συνέχεια ανάλογα των τιμών με τις οποίες οδηγούσαμε τις εισόδους του κυκλώματος, λάμβαναν άλλοι τη λογική τιμή 1 και άλλοι τη λογική τιμή 0. Τώρα η είσοδος μιας πύλης μπορεί να έχει τη λογική τιμή 0, τη λογική τιμή 1, την άγνωστη τιμή X , την τιμή D και την τιμή D' . Στο σχήμα 4.6 βλέπουμε τους πίνακες αληθείας των βασικών λογικών πυλών AND, OR και NOT όταν οι εισοδοί μπορούν να πάρουν οποιαδήποτε από τις ανωτέρω αναφερθείσες τιμές.

AND	0	1	x	D	D'	OR	0	1	x	D	D'	A	A'
0	0	0	0	0	0	0	0	1	x	D	D'	0	1
1	0	1	x	D	D'	1	1	1	1	1	1	1	0
x	0	x	x	x	x	x	x	1	x	x	x	x	x
D	0	D	x	D	0	D	D	1	x	D	1	D	D'
D'	0	D'	x	0	D'	D'	D'	1	x	1	D'	D'	D

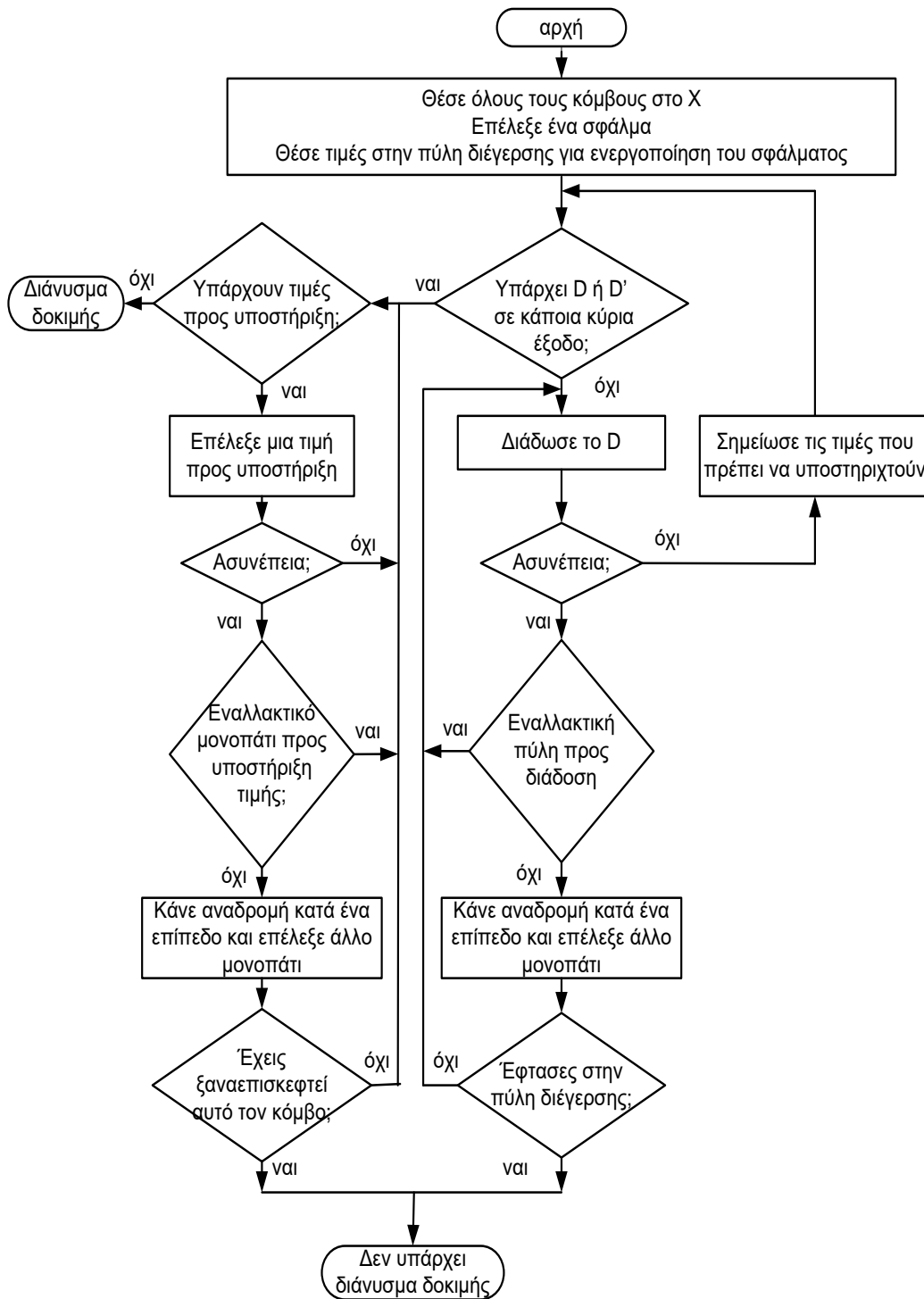
Σχήμα 4.6

Για να ενεργοποιήσουμε ένα σφάλμα πρέπει να θέσουμε τον κόμβο που έχει το σφάλμα σε μια τιμή που είναι συμπληρωματική της τιμής του σφάλματος. Για να διαδώσουμε το σφάλμα πρέπει να κάνουμε τις συνέπειες του σφάλματος να εμφανιστούν σε κάποια έξοδο του κυκλώματος ως D ή D' .

Αλγόριθμος D

Ο αλγόριθμος D δεν καθορίζει εάν πρώτα πρέπει να κάνουμε ενεργοποίηση του σφάλματος από τις κύριες εισόδους του κυκλώματος ή αν θα πρέπει πρώτα να προσπαθήσουμε να διαδώσουμε τις συνέπειες του σφάλματος στις κύριες εξόδους του κυκλώματος. Αυτός που κάνει την υλοποίηση του αλγορίθμου θα πρέπει να αποφασίσει τι να κάνει πρώτα.

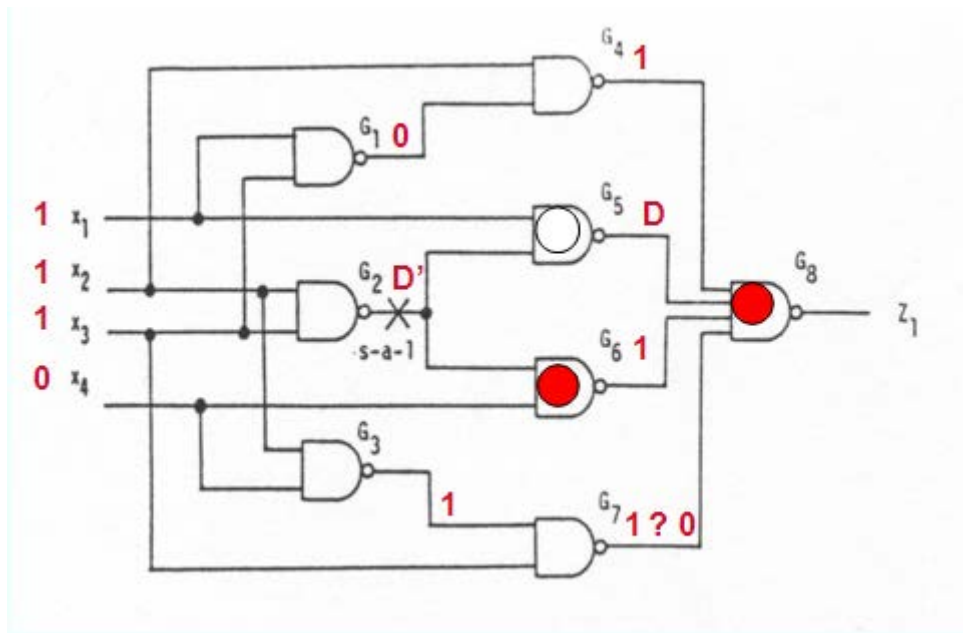
Στο σχήμα 4.7 δίνεται η περιγραφή ενός αλγορίθμου D. Στη συγκεκριμένη υλοποίηση πρώτα γίνεται η διάδοση των συνεπειών του σφάλματος. Πύλη διέγερσης καλείται η πύλη στην έξοδο της οποίας υπάρχει το σφάλμα για το οποίο θέλουμε να εξάγουμε διάγνωση δοκιμής.



Σχήμα 4.7 Αλγόριθμος D

Στο μέτωπο των D ανήκουν οι πύλες των οποίων η έξοδος έχει απροσδιόριστη τιμή ενώ κάποια είσοδος έχει την τιμή D ή D'.

Θεωρούμε το κύκλωμα του σχήματος 4.8. Υποθέστε ότι θέλουμε να εξάγουμε ένα διάλυμα δοκιμής για την ανίχνευση του σφάλματος η έξοδος της πύλης G_2 έχει κολλήσει στη λογική τιμή 1, $G_2/1$.



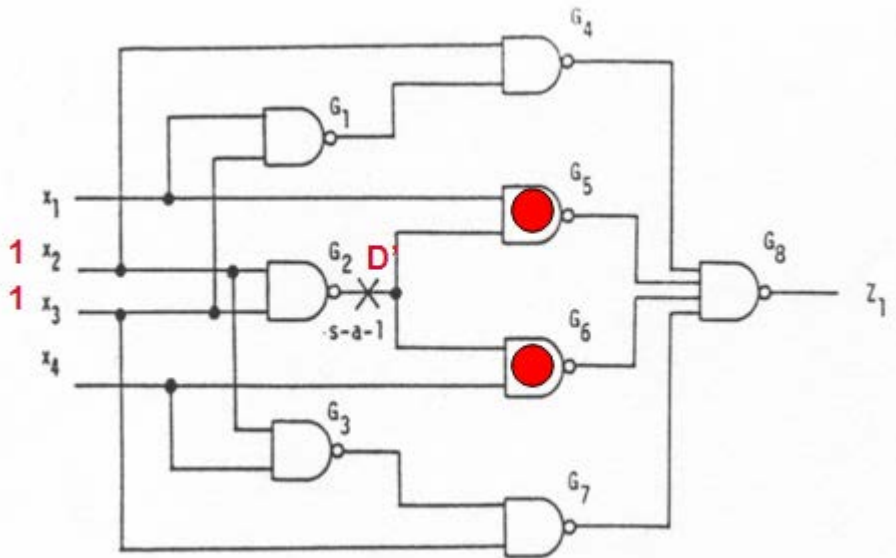
Σχήμα 4.8

Κάθε κόμβος του κυκλώματος ο οποίος δεν έχει τιμή, θεωρούμε ότι έχει την απροσδιόριστη τιμή X. Για την ενεργοποίηση του σφάλματος θέτουμε τις εισόδους X_2 και X_3 στη λογική τιμή 1. Τότε η έξοδος της πύλης G_2 λαμβάνει την τιμή D' .

Τώρα στο μέτωπο των D ανήκουν οι πύλες G_5 και G_6 . Επιλέγουμε να διαδώσουμε το D μέσω της πύλης G_5 . Για να γίνει αυτό θέτουμε την είσοδο X_1 στη λογική τιμή 1. Τότε η έξοδος της πύλης G_5 λαμβάνει την τιμή D. Τώρα στο μέτωπο των D ανήκουν οι πύλες G_6 και G_8 . Επιλέγουμε να διαδώσουμε το D μέσω της πύλης G_8 . Για να γίνει αυτό πρέπει οι εξοδοί των πυλών G_4 , G_6 και G_7 να λάβουν τη λογική τιμή 1. Η μία είσοδος της πύλης έχει ήδη τη λογική τιμή 1, επομένως η άλλη είσοδος, δηλαδή η έξοδος της πύλης G_1 , πρέπει να λάβει τη λογική τιμή 0 (υποστήριξη τιμής). Για να λάβει η έξοδος της πύλης G_1 τη λογική τιμή 0, θέτουμε την είσοδο X_1 στη λογική τιμή 1 (υποστήριξη τιμής). Για να λάβει η έξοδος της πύλης G_6 τη λογική τιμή 1, θέτουμε την είσοδο X_4 στη λογική τιμή 0 (υποστήριξη τιμής). Τότε όμως η έξοδος της πύλης G_3 λαμβάνει τη λογική τιμή 1 (συνεπαγωγή), οπότε η έξοδος της πύλης G_7 λαμβάνει τη λογική τιμή 0 (συνεπαγωγή), η οποία είναι σε αντίθεση με την επιθυμητή τιμή που είναι η λογική τιμή 1 (αντίφαση).

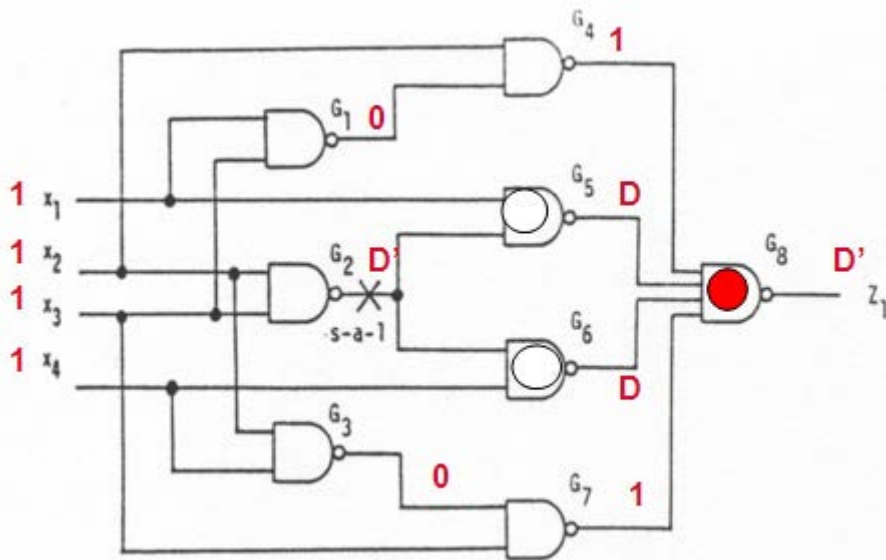
Επιστρέφουμε στο σημείο που λάβαμε την απόφαση να διαδώσουμε το D μέσω της πύλης G_5 και λαμβάνουμε την εναλλακτική απόφαση να διαδώσουμε το D μέσω της πύλης G_6 (αναδρομή). Επειδή το κύκλωμα είναι συμμετρικό, ακολουθώντας τα ίδια βήματα θα καταλήξουμε και πάλι σε αντίφαση.

Επιστρέφουμε στο σημείο (βλέπε σχήμα 4.9) που λάβαμε την απόφαση να διαδώσουμε το D μέσω της πύλης G_6 και λαμβάνουμε την εναλλακτική απόφαση να διαδώσουμε το D μέσω και των δύο πυλών G_5 και G_6 (αναδρομή).



Σχήμα 4.9

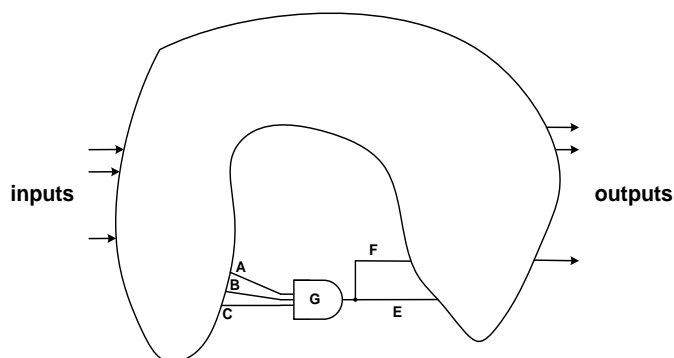
Για να περάσει το D μέσω της πύλης G₅ θέτουμε την είσοδο X₁ στη λογική τιμή 1. Τότε η έξοδος της πύλης G₅ λαμβάνει την τιμή D, η έξοδος της πύλης G₁ λαμβάνει τη λογική τιμή 0 και η έξοδος της πύλης G₄ λαμβάνει τη λογική τιμή 1. Για να περάσει το D μέσω της πύλης G₆ θέτουμε την είσοδο X₄ στη λογική τιμή 1. Τότε η έξοδος της πύλης G₆ λαμβάνει την τιμή D, η έξοδος της πύλης G₃ λαμβάνει τη λογική τιμή 0 και η έξοδος της πύλης G₇ λαμβάνει τη λογική τιμή 1. Τότε η έξοδος της πύλης G₈ λαμβάνει την τιμή D' (βλέπε σχήμα 4.10). Επομένως το διάνυσμα X₁X₂X₃X₄=1111 αποτελεί το διάνυσμα δοκιμής του σφάλματος η έξοδος της πύλης G₂ έχει κολλήσει στη λογική τιμή 1, G₂/1.



Σχήμα 4.10

Στο σχήμα 4.11 θεωρούμε ότι η πύλη G είναι εμφωλευμένη μέσα σε ένα κύκλωμα και ότι θέλουμε να εξάγουμε ένα διάνυσμα δοκιμής για το σφάλμα η έξοδος της πύλης G έχει κολλήσει στη λογική τιμή 1, G₂/1. Για να ενεργοποιήσουμε το σφάλμα, αρκεί να θέσουμε στη λογική τιμή 0 μια από τις εισόδους A,B και Γ της

πύλης G. Ο αλγόριθμος D δεν μας δίνει πληροφορία για το ποια από τις εισόδους θα πρέπει να επιλέξουμε πρώτη.



Σχήμα 4.11

Για να διορθώσουμε τις συνέπειες του σφάλματος από την έξοδο της πύλης G σε κάποια από τις κύριες εξόδους του κυκλώματος, θα πρέπει να επιλέξουμε καταρχήν ένα από τα μονοπάτια F και E. Ο αλγόριθμος D δεν μας δίνει πληροφορία για το ποιο από τα δύο μονοπάτια θα πρέπει να επιλέξουμε αρχικά.

Ο αλγόριθμος PODEM δίνει οδηγίες για το ποια είσοδος θα πρέπει να δοκιμαστεί πρώτη για να δοθεί η επιθυμητή τιμή στην έξοδο μιας πύλης, καθώς και για το ποιο μονοπάτι θα πρέπει να χρησιμοποιήσουμε πρώτο ως μονοπάτι διάδοσης των συνεπειών του σφάλματος. Πριν περιγράψουμε τον αλγόριθμο PODEM, θα πρέπει να μιλήσουμε για τα μέτρα δοκιμασιμότητας (Testability Measures).

Τα μέτρα δοκιμασιμότητας αρχικά προτάθηκαν για να εκτιμούμε εάν είναι εύκολο ή όχι να ελέγξουμε την ορθή λειτουργία ενός τμήματος του κυκλώματος, ώστε εάν δεν είναι εύκολο να τροποποιούμε το σχεδιασμό του. Σ' αυτό τον ρόλο τους τα μέτρα δοκιμασιμότητας απέτυχαν. Βρήκαν όμως εφαρμογή στους αλγόριθμους εξαγωγής διανυσμάτων δοκιμής.

Η εκτίμηση της ελεγχιμότητας και παρατηρησιμότητας ενός κόμβου του κυκλώματος μπορεί να είναι χρήσιμη για τον προσδιορισμό της ευκολίας ή δυσκολίας εξαγωγής διανυσμάτων δοκιμής.

Έχουν προταθεί διάφορα μέτρα δοκιμασιμότητας. Θα παρουσιάσουμε τα μέτρα SCOAP. Σύμφωνα με αυτά τα μέτρα, υπάρχουν τρία μέτρα για τα συνδυαστικά και τρία μέτρα για τα ακολουθιακά κυκλώματα. Θα παρουσιάσουμε μόνο τα μέτρα για συνδυαστικά κυκλώματα.

Η συνδυαστική ελεγχιμότητα ενός κόμβου του κυκλώματος για τη λογική τιμή 0 εκφράζει τη δυσκολία με την οποία μπορούμε να θέσουμε τον κόμβο αυτό στη λογική τιμή 0 θέτοντας τιμές στις κύριες εισόδους του κυκλώματος.

Η συνδυαστική ελεγχιμότητα ενός κόμβου του κυκλώματος για τη λογική τιμή 1 εκφράζει τη δυσκολία με την οποία μπορούμε να θέσουμε τον κόμβο αυτό στη λογική τιμή 1 θέτοντας τιμές στις κύριες εισόδους του κυκλώματος.

Η συνδυαστική παρατηρησιμότητα ενός κόμβου του κυκλώματος εκφράζει τη δυσκολία με την οποία μπορούμε να παρατηρήσουμε ότι ο κόμβος αυτός άλλαξε λογική τιμή παρατηρώντας τις κύριες εξόδους του κυκλώματος.

Αρχικά θέτουμε:

- τη συνδυαστική ελεγχσιμότητα για 0 και τη συνδυαστική ελεγχσιμότητα για 1 των κυρίων εισόδων του κυκλώματος ίση με 1.
- τη συνδυαστική παρατηρησιμότητα των κυρίων εξόδων του κυκλώματος ίση με 0.
- τη συνδυαστική ελεγχσιμότητα για 0, τη συνδυαστική ελεγχσιμότητα για 1 και τη συνδυαστική παρατηρησιμότητα όλων των άλλων κόμβων του κυκλώματος στην τιμή άπειρο.

Εάν η τιμή εξόδου μιας λογικής πύλης καθορίζεται θέτοντας μόνο μια είσοδό της σε μια τιμή ελέγχου τότε:

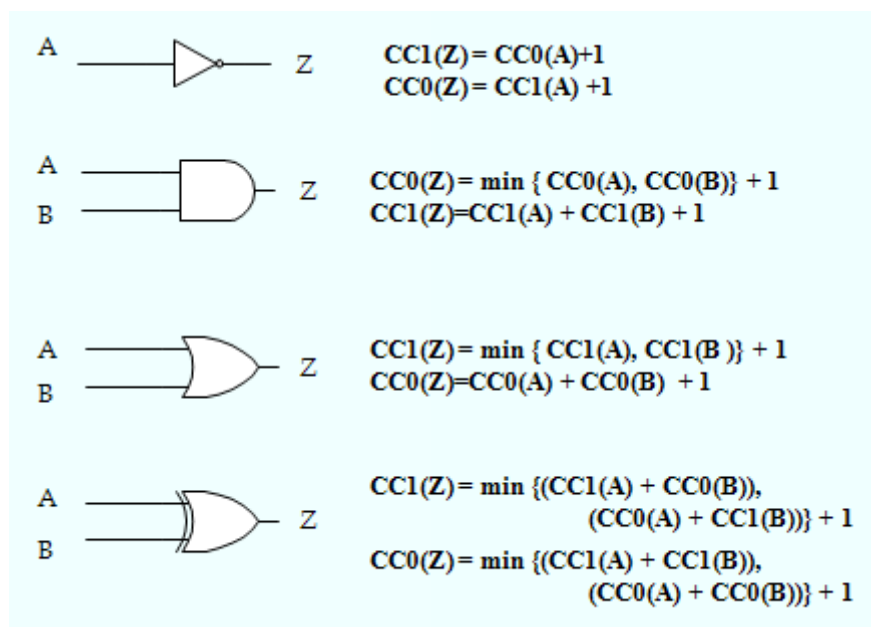
ελεγχσιμότητα εξόδου = ελεγχσιμότητα της εισόδου που έχει την ελάχιστη τιμή +1

Εάν για να θέσουμε την έξοδο μιας λογικής πύλης σε μια τιμή πρέπει να θέσουμε όλες τις εισόδους της σε συγκεκριμένες λογικές τιμές, τότε:

ελεγχσιμότητα εξόδου = το άθροισμα των ελεγχσιμοτήτων όλων των εισόδων της πύλης +1

Εάν μπορούμε να θέσουμε την έξοδο μιας λογικής πύλης σε μια τιμή, θέτοντας τις εισόδους της εναλλακτικά σε περισσότερους από έναν συνδυασμούς διαφορετικών τιμών τότε:

ελεγχσιμότητα εξόδου = το αθροίσημα των ελεγχσιμοτήτων του συνδυασμού τιμών που οδηγεί στην ελάχιστη τιμή ελεγχσιμότητας +1



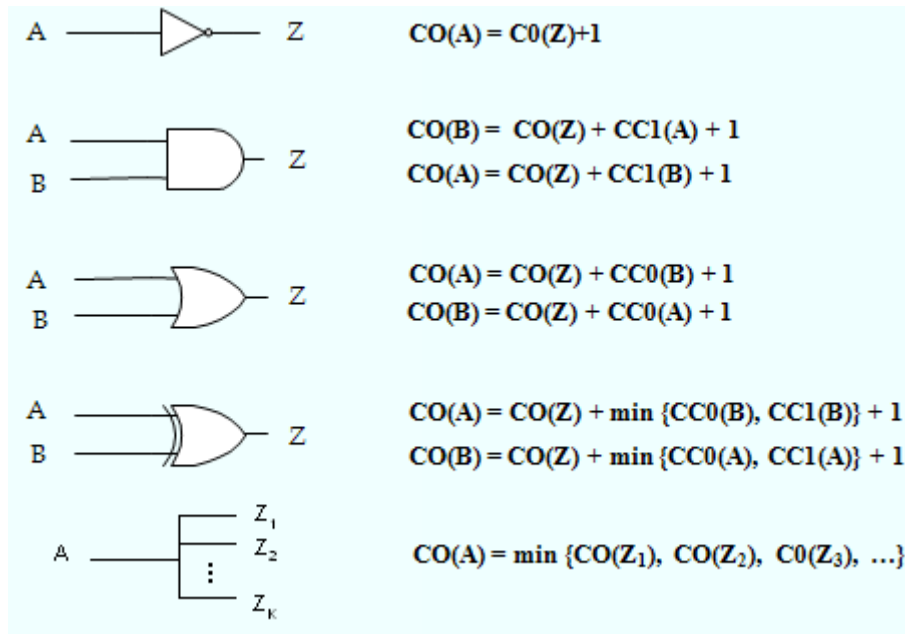
Σχήμα 4.12 Παραδείγματα.

Πρέπει να σχολιάσουμε ότι δεν είναι περισσότερο δύσκολο να θέσουμε την έξοδο μιας πύλης NOT σε μια λογική τιμή, απ' όσο είναι να θέσουμε την είσοδό της στη συμπληρωματική της τιμή. Παρόλα αυτά παρατηρούμε ότι για τον υπολογισμό της

ελεγχιμότητας της εξόδου μιας πύλης NOT προσθέτουμε τη μονάδα στην ελεγχιμότητα της εισόδου της (εξ ορισμού).

Η παρατηρησιμότητα της εισόδου μιας λογικής πύλης ισούται με το άθροισμα της παρατηρησιμότητας της εξόδου της λογικής πύλης και των ελεγχιμοτήτων των άλλων εισόδων της πύλης για τιμές μη ελέγχου.

Σε ένα σημείο διακλάδωσης η παρατηρησιμότητα του μίσχου (stem) ισούται με την παρατηρησιμότητα του κλάδου που έχει τη μικρότερη τιμή παρατηρησιμότητας.



Σχήμα 4.13 Παραδείγματα.

Αλγόριθμος PODEM

- Υπολογισμός των μέτρων δοκιμαστικότητας για κάθε κόμβο του κυκλώματος.
- Επιλογή σφάλματος.
- Βασιζόμενοι στα μέτρα ελεγχιμότητας επιλέγουμε στόχο για την ενεργοποίηση του σφάλματος.
- Βασιζόμενοι στα μέτρα ελεγχιμότητας μετακινούμε το στόχο (ιχνηλάτηση) σε μια κύρια είσοδο.
- Αποδίδουμε την τιμή στην κύρια είσοδο και κάνουμε συνεπαγωγή.
- Εάν δεν ενεργοποιήθηκε το σφάλμα επαναλαμβάνουμε τη διαδικασία καθορισμού στόχου, μετακίνησης στόχου σε κάποια από τις κύριες εισόδους και κάνουμε συνεπαγωγή.
- Εάν σε κάποιο βήμα προκύψει αντίφαση, στην κύρια είσοδο που έλαβε τελευταία τιμή δίνουμε τη συμπληρωματική της τιμή (αναδρομή) και κάνουμε συνεπαγωγή.
- Εάν το σφάλμα δεν ενεργοποιήθηκε και δεν υπάρχει άλλος συνδυασμός τιμών εισόδου, τότε το σφάλμα είναι μη ανιχνεύσιμο.

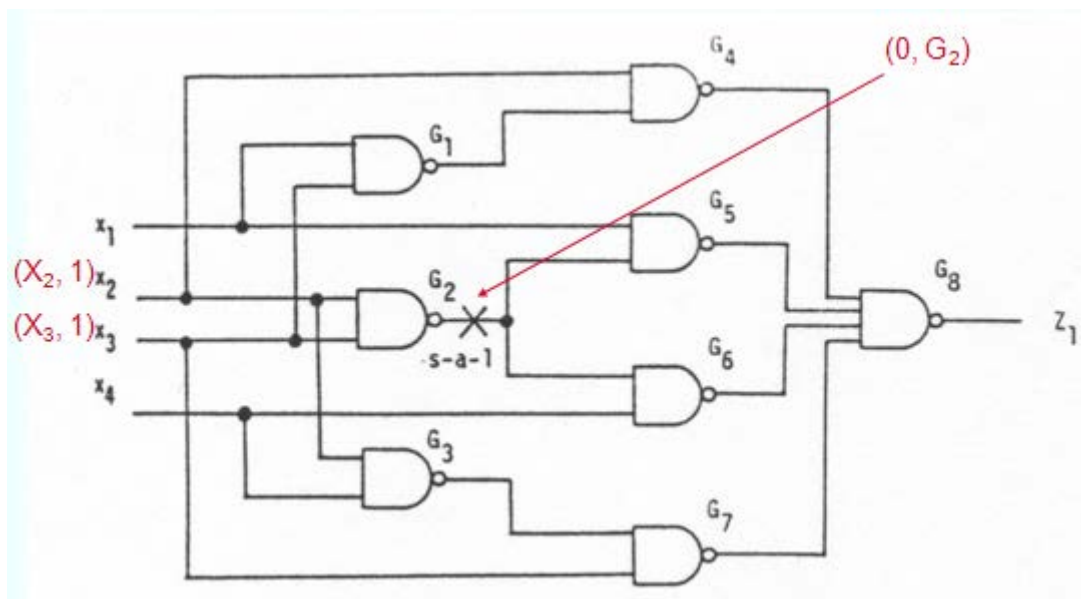
- Εάν το σφάλμα ενεργοποιήθηκε, τότε βασιζόμενοι στα μέτρα παρατηρησιμότητας επιλέγουμε πύλη και θέτουμε στόχο για τη διάδοση των συνεπειών του σφάλματος μέσω αυτής της πύλης.
- Βασιζόμενοι στα μέτρα ελεγχιμότητας μετακινούμε το στόχο (ιχνηλάτηση) σε μια κύρια είσοδο.
- Αποδίδουμε την τιμή στην κύρια είσοδο και κάνουμε συνεπαγωγή.
- Εάν οι συνέπειες του σφάλματος, δηλαδή το D ή το D' , δεν έφτασαν σε κάποια από τις κύριες εξόδους του κυκλώματος, επαναλαμβάνουμε τη διαδικασία καθορισμού στόχου, μετακίνησης στόχου σε κάποια από τις κύριες εισόδους και κάνουμε συνεπαγωγή.
- Εάν σε κάποιο βήμα προκύψει αντίφαση, στην κύρια είσοδο που έλαβε τελευταία τιμή δίνουμε τη συμπληρωματική της τιμή (αναδρομή) και κάνουμε συνεπαγωγή.
- Εάν το D ή το D' δεν διαδόθηκε σε κάποια από τις κύριες εξόδους του κυκλώματος και δεν υπάρχει άλλος συνδυασμός τιμών εισόδου, τότε το σφάλμα είναι μη ανιχνεύσιμο.

Όταν έχουμε να μετακινήσουμε ένα στόχο από την έξοδο μιας λογικής πύλης στην είσοδό της, τότε έχουμε δύο περιπτώσεις:

α. Για να θέσουμε την έξοδο της λογικής πύλης στην επιθυμητή τιμή, αρκεί να θέσουμε μια από τις εισόδους της λογικής πύλης στην τιμή ελέγχου της. Σ' αυτή την περίπτωση θα μεταφέρουμε το στόχο στην είσοδο που έχει τη μικρότερη τιμή ελεγχιμότητας.

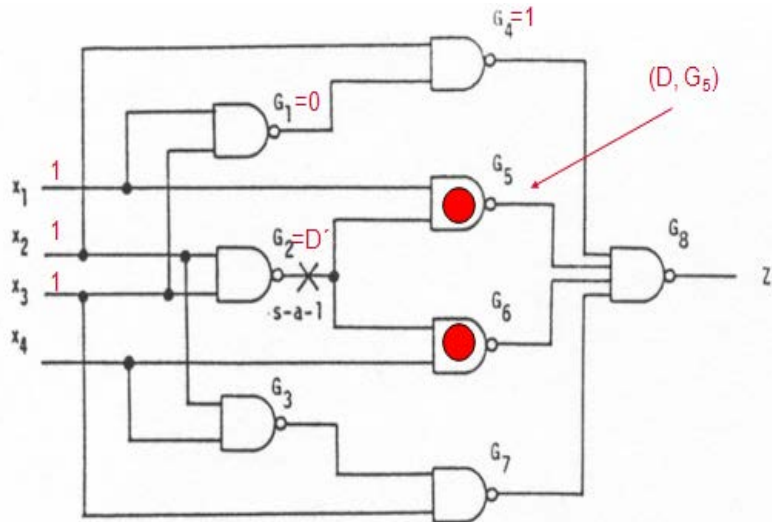
β. Για να θέσουμε την έξοδο της λογικής πύλης στην επιθυμητή τιμή, πρέπει να θέσουμε όλες τις εισόδους της λογικής πύλης στην τιμή μη-ελέγχου της. Σ' αυτή την περίπτωση θα μεταφέρουμε το στόχο στην είσοδο που έχει τη μεγαλύτερη τιμή ελεγχιμότητας.

Όταν έχουμε να διαδώσουμε το D προς τις κύριες εξόδους του κυκλώματος επιλέγουμε τη γραμμή που έχει τη μικρότερη παρατηρησιμότητα.



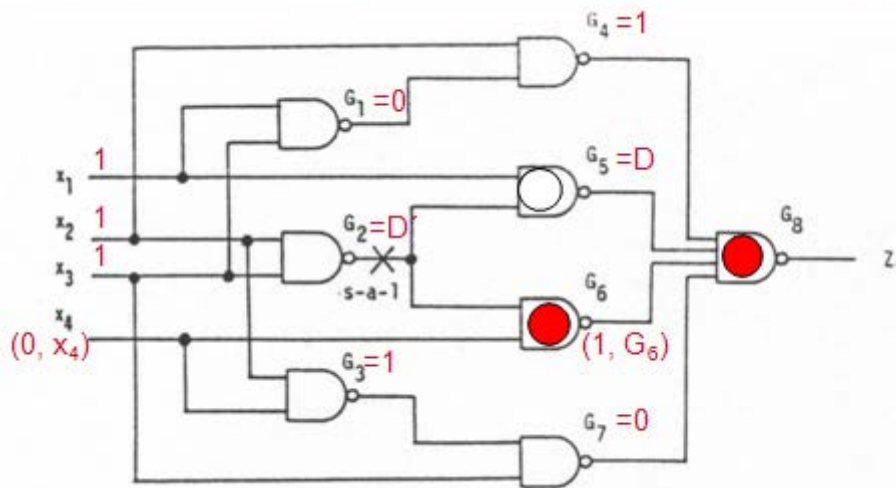
Σχήμα 4.14

Θεωρούμε το κύκλωμα του σχήματος 4.14 και το σφάλμα η έξοδος της πύλης G_2 έχει κολλήσει στη λογική τιμή 1. Για να ενεργοποιήσουμε το σφάλμα αρχικός στόχος είναι να θέσουμε την έξοδο της πύλης G_2 στη λογική τιμή 0 (στόχος $(0, G_2)$). Με ιχνηλάτηση ο στόχος μεταφέρεται στην κύρια είσοδο X_2 και θέτουμε $X_2 = 1$. Πάλι ο στόχος είναι $(0, G_2)$ και με ιχνηλάτηση μεταφέρεται στην κύρια είσοδο X_3 , οπότε θέτουμε $X_3 = 1$. Με συνεπαγωγή λαμβάνουμε $G_2 = D'$.



Σχήμα 4.15

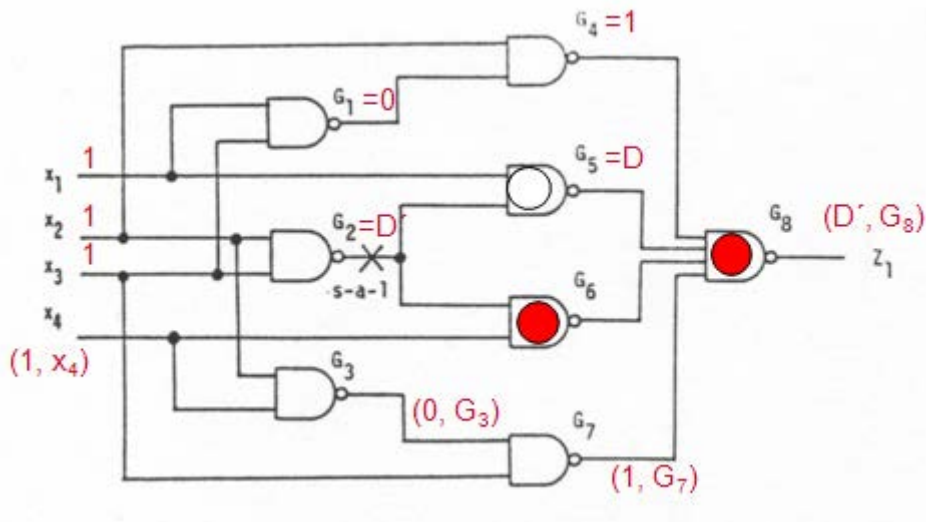
Το μέτωπο των D αποτελείται από τις πύλες G_5 και G_6 (βλέπε σχήμα 4.15). Για να διαδώσουμε τις συνέπειες του σφάλματος μέσω της πύλης G_5 θέτουμε το στόχο $(1, X_1)$, οπότε $X_1 = 1$. Τότε με συνεπαγωγή λαμβάνουμε $G_1 = 0$, $G_4 = 1$ και $G_5 = D$.



Σχήμα 4.16

Τώρα το μέτωπο των D αποτελείται από τις πύλες G_6 και G_8 (βλέπε σχήμα 4.16). Για να διαδώσουμε τις συνέπειες του σφάλματος μέσω της πύλης G_8 θέτουμε αρχικά το στόχο $(1, G_6)$. Ο στόχος μεταφέρεται στην είσοδο X_4 ως $(0, X_4)$, οπότε θέτουμε $X_4 = 0$. Με συνεπαγωγή λαμβάνουμε $G_6 = 1$, $G_3 = 1$, $G_7 = 0$ and $G_8 = 1$, οπότε έχουμε

αντίφαση. Τότε κάνουμε αναδρομή και αλλάζουμε τη λογική τιμή της εισόδου που προσδιορίσαμε τελευταία. Τελευταία δώσαμε την τιμή 0 στην είσοδο X_4 , οπότε τώρα θέτουμε $X_4 = 1$ (βλέπε σχήμα 4.17).

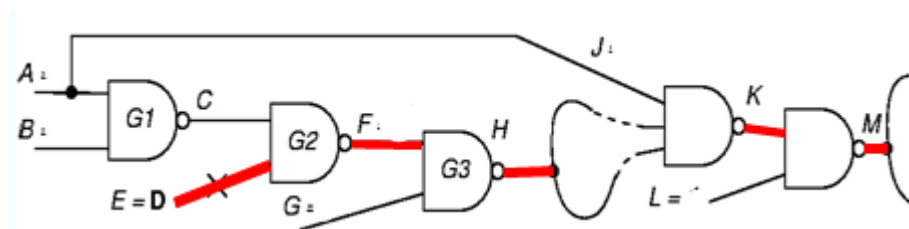


Σχήμα 4.17

Με συνεπαγωγή λαμβάνουμε $G_6=D$, $G_3=0$, $G_7=1$ and $G_8=D'$. Επομένως το διάνυσμα εισόδου $X_1X_2X_3X_4=1111$ ανιχνεύει το σφάλμα η έξοδος της πύλης G_2 έχει κολλήσει στη λογική τιμή 1.

Αλγόριθμος FAN

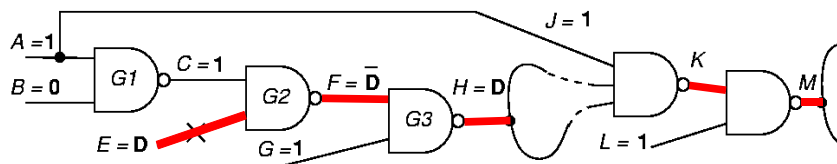
Ο αλγόριθμος D λαμβάνει αποφάσεις σε όλους τους κόμβους του κυκλώματος, ενώ ο αλγόριθμος PODEM λαμβάνει αποφάσεις μόνο στις κύριες εισόδους του κυκλώματος. Αν και το πλήθος των αποφάσεων μειώνεται σημαντικά εξακολουθεί να είναι υπερβολικά μεγάλο. Επιπλέον, επειδή ο PODEM προσπαθεί να επιτύχει έναν στόχο κάθε φορά, η διαδικασία της απόφασης είναι μερικές φορές τοπική και χάνει τη συνολική εικόνα. Ο FAN χρησιμοποιεί πρόσθετες στρατηγικές για να θεραπεύσει τα μειονεκτήματα που αναφέραμε για τον αλγόριθμο PODEM.



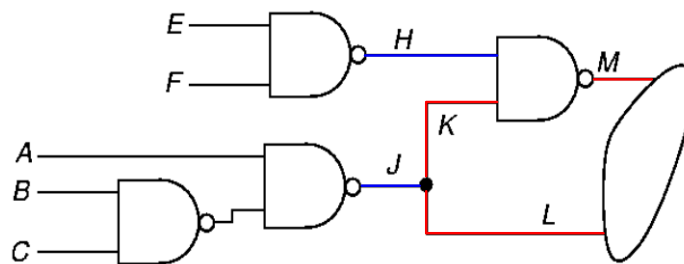
Σχήμα 4.18 Μονοπάτια μοναδικής διάδοσης.

Θεωρούμε ότι οι συνέπειες του σφάλματος έχουν φτάσει στην είσοδο E της πύλης G_2 του σχήματος 4.18. Παρατηρούμε ότι για να φτάσουν οι συνέπειες του σφάλματος σε κάποια από τις κύριες εισόδους του κυκλώματος, θα πρέπει να περάσουν

οπωσδήποτε από τα μονοπάτια του κόκκινου χρώματος. Για να ευαισθητοποιηθούν τα μονοπάτια αυτά πρέπει να λάβουν τις τιμές που φαίνονται στο σχήμα 4.19. Σύμφωνα με τον αλγόριθμο FAN τα μονοπάτια αυτά θα πρέπει να ευαισθητοποιηθούν.



Σχήμα 4.19 Ευαισθητοποίηση των μονοπατιών μοναδικής διάδοσης.



Σχήμα 4.20

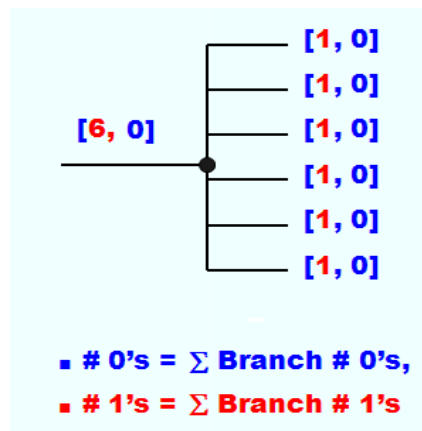
Οι γραμμές του κυκλώματος διακρίνονται σε τρεις κατηγορίες (βλέπε σχήμα 4.20).

- Οι γραμμές στις οποίες μπορούμε να φτάσουμε όταν κινηθούμε από ένα σημείο διακλάδωσης προς τις κύριες εξόδους του κυκλώματος, καλούνται δεσμευμένες γραμμές (bound lines).
- Όλες οι άλλες γραμμές καλούνται ελεύθερες γραμμές (free lines).
- Οι ελεύθερες γραμμές οι οποίες είναι γειτονικές με δεσμευμένες γραμμές καλούνται επικεφαλείς γραμμές (head lines).

Το κύκλωμα που αρχίζει από κάποιες κύριες εισόδους του κυκλώματος και φτάνει μέχρι μια επικεφαλής γραμμή θα έχει τη δομή δένδρου. Σε ένα κύκλωμα με δομή δένδρου, οποιαδήποτε τιμή στην έξοδο του κυκλώματος μπορούμε να την υποστηρίξουμε θέτοντας τιμές στις κύριες εισόδους του. Επομένως δεν υπάρχει λόγος να χάνουμε χρόνο για να υποστηρίξουμε τις τιμές στις επικεφαλείς γραμμές, ενώ μπορεί αργότερα να προκύψει μια αντίφαση και να χρειαστεί οι τιμές αυτές να αλλάξουν. Επομένως σύμφωνα με τον FAN πρέπει να σταματήσουμε την ιχνηλάτηση σε μια επικεφαλής γραμμή και να αναβάλλουμε την υποστήριξη της τιμής της για αργότερα.

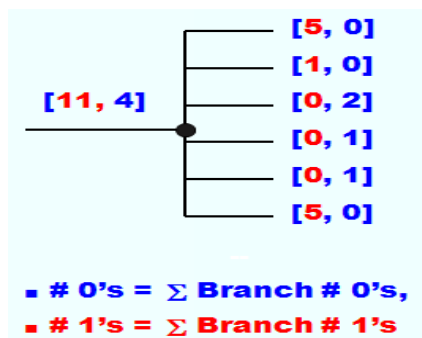
Λόγω της ύπαρξης μονοπατιών που επανασυγκλίνουν, είναι δυνατόν ξεκινώντας από ένα κόμβο του κυκλώματος και προχωρώντας προς τις κύριες εισόδους του να φτάσουμε στο ίδιο σημείο διακλάδωσης ακολουθώντας διαφορετικά μονοπάτια. Σύμφωνα με τον FAN πρέπει να κάνουμε πολλαπλή ιχνηλάτηση και εάν με την πολλαπλή ιχνηλάτηση σ' ένα σημείο διακλάδωσης προκύψουν περισσότεροι από ένας διαφορετικοί στόχοι, τότε πρέπει να σταματήσουμε την ιχνηλάτηση και να δώσουμε σ' αυτό το σημείο μια λογική τιμή. Αιτία εφαρμογής αυτής της στρατηγικής είναι ότι αφού υπάρχουν περισσότεροι από ένας στόχοι που δεν μπορούν να ικανοποιηθούν

ταυτόχρονα, υπάρχει πολύ μεγάλη πιθανότητα να προκύψει αντίφαση. Επομένως είναι καλύτερα να εντοπίσουμε την αντίφαση όσο το δυνατόν γρηγορότερα για να μη χάνουμε χρόνο.



Σχήμα 4.21 Παράδειγμα ιχνηλάτησης που συνεχίζεται.

Στο σχήμα 4.21 επειδή όλοι οι στόχοι είναι ταυτοτικοί δεν δίνεται τιμή στο μίσχο της διακλάδωση. Η ιχνηλάτηση προς τις κύριες εισόδους του κυκλώματος θα συνεχιστεί.



Σχήμα 4.22 Παράδειγμα ιχνηλάτησης που σταματά σε σημείο διακλάδωσης.

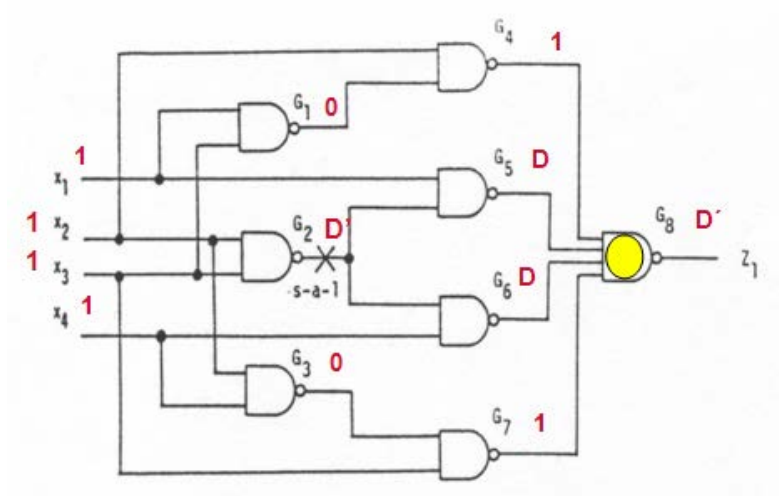
Στο σχήμα 4.22 επειδή όλοι οι στόχοι δεν είναι ταυτοτικοί, στο μίσχο δίνεται η τιμή που απαιτείται από τους περισσότερους στόχους. Η ιχνηλάτηση θα σταματήσει σ' αυτό το σημείο και θα γίνει συνεπαγωγή τιμών.

Θεωρούμε το κύκλωμα του σχήματος 4.23 και το σφάλμα η έξοδος της πύλης G_2 έχει κολλήσει στη λογική τιμή 1.

Για να ενεργοποιήσουμε το σφάλμα αρχικός στόχος είναι να θέσουμε την έξοδο της πύλης G_2 στη λογική τιμή 0 (στόχος $(0, G_2)$). Με ιχνηλάτηση ο στόχος μεταφέρεται στην κύρια είσοδο X_2 και θέτουμε $X_2 = 1$. Πάλι ο στόχος είναι $(0, G_2)$ και με ιχνηλάτηση μεταφέρεται στην κύρια είσοδο X_3 , οπότε θέτουμε $X_3 = 1$. Με συνεπαγωγή λαμβάνουμε $G_2 = D'$.

Παρατηρούμε ότι για να διαδοθούν οι συνέπειες του σφάλματος στην κύρια έξοδο του κυκλώματος, άσχετα αν διαδοθούν μέσω της λογικής πύλης G_5 ή της λογικής πύλης G_6 , θα πρέπει να διαδοθούν μέσω της λογικής πύλης G_8 . Για να γίνει αυτό θα

πρέπει οι έξοδοι των πυλών G_4 και G_7 να λάβουν τη λογική τιμή 1. Αυτό συνεπάγεται ότι οι λογικές πύλες G_1 και G_3 πρέπει να λάβουν τη λογική τιμή 0.



Σχήμα 4.23

Επομένως οι κύριες εισόδους του κυκλώματος X_1 και X_4 πρέπει να λάβουν τη λογική τιμή 1. Θέτοντας τις εισόδους X_1 και X_4 στη λογική τιμή 1, οι έξοδοι των πυλών G_1 και G_3 λαμβάνουν τη λογική τιμή 0, οι έξοδοι των πυλών G_4 και G_7 λαμβάνουν τη λογική τιμή 1 και οι έξοδοι των πυλών G_5 και G_6 λαμβάνουν την τιμή D. Τότε η έξοδος της πύλης G_8 λαμβάνει την τιμή D', το οποίο συνεπάγεται ότι το διάνυσμα εισόδου $X_1X_2X_3X_4=1111$ ανιχνεύει το σφάλμα η έξοδος της πύλης G_2 έχει κολλήσει στη λογική τιμή 1. Παρατηρούμε ότι εφαρμόζοντας τη στρατηγική 1 φτάσαμε σε ένα διάνυσμα δοκιμής του σφάλματος η έξοδος της πύλης G_2 έχει κολλήσει στη λογική τιμή 1, χωρίς καμία αναδρομή.

Circuit	Χρόνος υπολογισμού		Μέσος αριθμός αναδρομών ανά σφάλμα	
	PODEM	FAN	PODEM	FAN
1	1.3	1	4.9	1.2
2	3.6	1	42.3	15.2
3	5.6	1	61.9	0.6
4	1.9	1	5.0	0.2
5	4.8	1	53.0	23.2

Σχήμα 4.24

Στο σχήμα 4.24 γίνεται μια σύγκριση των αλγορίθμων FAN και PODEM. Συγκρίνονται οι χρόνοι εξαγωγής των συνόλων δοκιμής και ο μέσος αριθμός αναδρομών ανά σφάλμα χρησιμοποιώντας τον PODEM και FAN. Οι χρόνοι έχουν διαιρεθεί με τους χρόνους που απαιτήθηκαν από τον αλγόριθμο FAN.

Παρατηρούμε ότι για το τρίτο κύκλωμα ο αλγόριθμος PODEM απαιτεί 5,6 φορές περισσότερο χρόνο από τον FAN. Αιτία είναι το μεγάλο πλήθος αναδρομών που απαιτεί ο αλγόριθμος PODEM για το συγκεκριμένο κύκλωμα. Από τον πίνακα

βλέπουμε ότι ο αλγόριθμος PODEM απαιτεί κατά μέσο όρο 61,9 αναδρομές ανά σφάλμα, ενώ ο αλγόριθμος FAN απαιτεί μόνο 0,6 αναδρομές ένα σφάλμα.

αλγόριθμος	D	PODEM	FAN
χώρος απόδοσης τιμών	όλοι οι κόμβοι του κυκλώματος	όλες οι κύριες εισοδοι του κυκλώματος	οι επικεφαλείς γραμμές και τα σημεία διακλάδωσης
σειρά απόδοσης τιμών	εξαρτάται από την υλοποίηση	σε μια κύρια εισοδο κάθε φορά	πρώτα σε σημεία διακλάδωσης με αντιφατικούς στόχους και μετά σε επικεφαλείς γραμμές
ευριστικές μέθοδοι	καμία	μέτρα δοκιμαστικότητας	μέτρα δοκιμαστικότητας και πρόσθετες στρατηγικές

Σχήμα 4.25 Κύρια χαρακτηριστικά των D, PODEM και FAN

Στο σχήμα 4.25 δίνονται τα κύρια χαρακτηριστικά των D, PODEM και FAN.

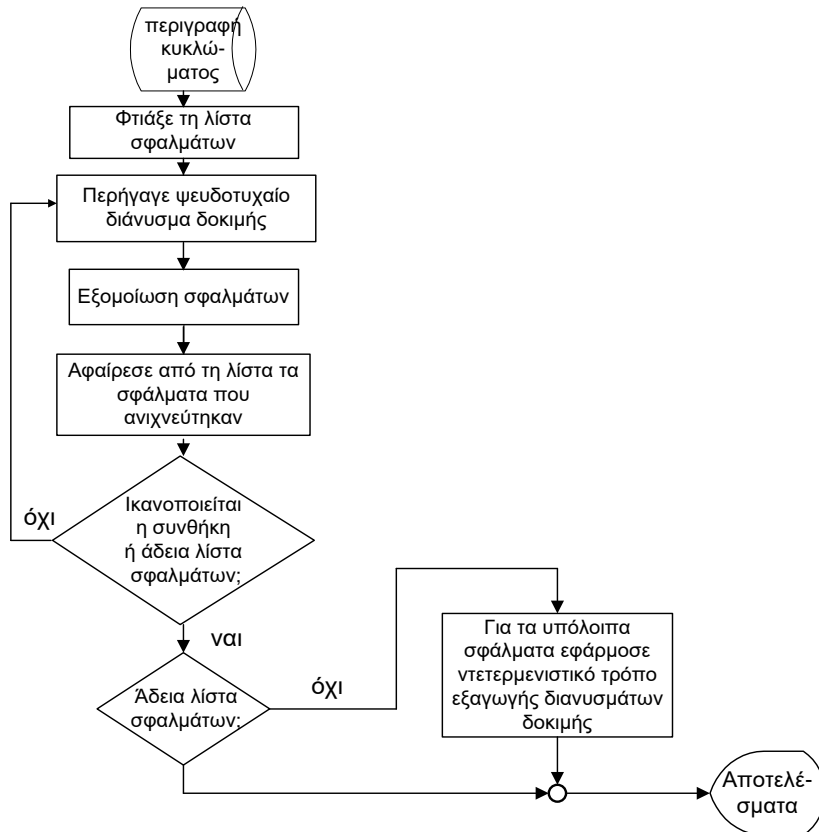
Το σύνολο δοκιμής που εξάγεται από έναν αλγόριθμο εξαγωγής διανυσμάτων δοκιμής μπορεί να είναι πολύ μεγάλο. Ίσως είναι δυνατόν να μειώσουμε το πλήθος των διανυσμάτων δοκιμής χωρίς να μειώσουμε την κάλυψη σφαλμάτων. Αυτό μπορεί να γίνει είτε δυναμικά είτε στατικά. Στη δυναμική περίπτωση η συμπίεση είναι ενσωματωμένη μέσα στον αλγόριθμο εξαγωγής διανυσμάτων δοκιμής. Ο αλγόριθμος προσπαθεί να εξάγει διανύσματα δοκιμής που κάθε ένα ανιχνεύει όσο το δυνατόν περισσότερα σφάλματα. Στη στατική συμπίεση εξετάζουμε δύο περιπτώσεις, την περίπτωση που τα διανύσματα δοκιμής είναι πλήρως ορισμένα και την περίπτωση που τα διανύσματα δοκιμής έχουν αδιάφορους όρους.

Στην πρώτη περίπτωση η συμπίεση βασίζεται στην εξομοίωση σφαλμάτων, επομένως πρέπει να είναι γνωστή η δομή του κυκλώματος. Για κάθε διάνυσμα δοκιμής κάνουμε εξομοίωση σφαλμάτων και σημειώνουμε όλα τα σφάλματα που ανιχνεύονται από αυτό το διάνυσμα. Στη συνέχεια θα πρέπει να επιλέξουμε τον ελάχιστο αριθμό διανυσμάτων δοκιμής που ανιχνεύουν όλα τα σφάλματα.

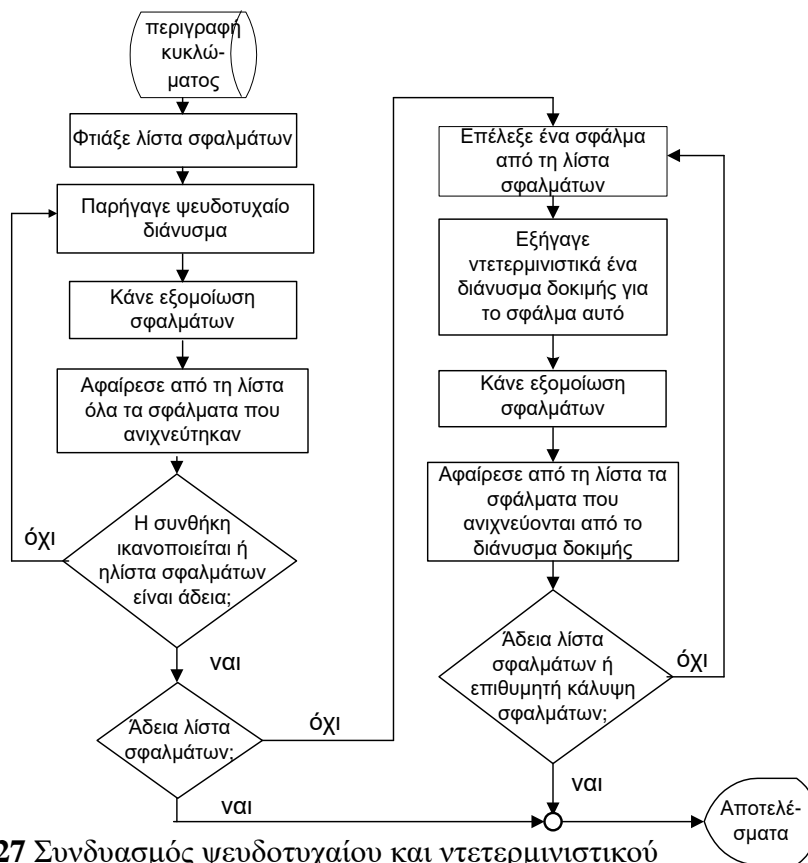
Στην περίπτωση που τα διανύσματα δοκιμής έχουν αδιάφορους όρους είναι δυνατόν να κάνουμε συμπίεση χωρίς να γνωρίζουμε τη δομή του κυκλώματος. Δύο διανύσματα δοκιμής που είναι συμβατά μεταξύ τους μπορούν να αντικατασταθούν από ένα διάνυσμα δοκιμής.

Στα σχήματα 4.26 και 4.27 δίνεται η περιγραφή συνδυασμού ψευδοτυχαίου, ντετερμινιστικού τρόπου εξαγωγής διανυσμάτων δοκιμής και εξομοίωσης σφαλμάτων.

Πρέπει να τονίσουμε ότι η εξομοίωση σφαλμάτων ως διαδικασία είναι πιο γρήγορη από τον ντετερμινιστικό τρόπο εξαγωγής διανυσμάτων δοκιμής. Η αιτία είναι ότι στην ντετερμινιστική εξαγωγή διανυσμάτων δοκιμής προκύπτουν αντιφάσεις που οδηγούν σε αναδρομές.



Σχήμα 4.26 Συνδυασμός ψευδοτυχαίου και ντετερμινιστικού τρόπου εξαγωγής διανυσμάτων δοκιμής. Η συνθήκη μπορεί είναι ένας συγκεκριμένος αριθμός από διανύσματα δοκιμής, μια συγκεκριμένη τιμή κάλυψης σφαλμάτων ή ότι κανένα σφάλμα δεν ανιχνεύτηκε από τα τελευταία κ ψευδοτυχαία διανύσματα.



Σχήμα 4.27 Συνδυασμός ψευδοτυχαίου και ντετερμινιστικού τρόπου εξαγωγής διανυσμάτων δοκιμής με εξομοίωση σφαλμάτων.

Μέρος Β

Σχεδίαση για Δοκιμασιμότητα (Design for Testability – DFT)

Κεφάλαιο 5. Ταξινόμηση των μεθόδων και γενικές οδηγίες

Κατά τη σχεδίαση ενός ολοκληρωμένου κυκλώματος θα πρέπει να έχουμε κατά νου τις ακόλουθες ερωτήσεις.

- Μπορούμε να εξάγουμε διανύσματα δοκιμής για την ανίχνευση όλων των σφαλμάτων που μας ενδιαφέρουν;
- Ο χρόνος εξαγωγής του συνόλου δοκιμής είναι αρκετά μικρός ώστε να μην έχουμε προβλήματα κόστους;
- Ο χρόνος ελέγχου της ορθής λειτουργίας του ολοκληρωμένου κυκλώματος μετά την κατασκευή του θα είναι μικρός ώστε να μην έχουμε οικονομικά προβλήματα;

Η σχεδίαση θα πρέπει να γίνει κατά τέτοιο τρόπο ώστε να μπορούμε να απαντήσουμε θετικά στα παραπάνω ερωτήματα.

Ο Σχεδιασμός για Δοκιμαστικότητα (Design for Testability – DFT) αναφέρεται σ' εκείνες τις σχεδιαστικές τεχνικές οι οποίες καθιστούν την εξαγωγή των διανυσμάτων δοκιμής και τον έλεγχο της ορθής λειτουργίας του κυκλώματος μετά την κατασκευή του (test application) αποτελεσματική και αποδοτική.

Οι μέθοδοι σχεδίασης ψηφιακών κυκλωμάτων για δοκιμαστικότητα μπορούν να ταξινομηθούν σε τρεις μεγάλες κατηγορίες.

Στην πρώτη κατηγορία ανήκουν οι γενικές οδηγίες. Στη δεύτερη κατηγορία ανήκουν μέθοδοι που εφαρμόζονται κατά περίπτωση. Αυτό συνεπάγεται ότι σε διαφορετικά κυκλώματα πρέπει να εφαρμοστούν με διαφορετικό τρόπο και με απρόβλεπτα αποτελέσματα. Επίσης είναι δύσκολο να προβλέψουμε πόσος χρόνος θα απαιτηθεί για την υλοποίηση αυτών των τεχνικών. Στην τρίτη κατηγορία ανήκουν μέθοδοι που μπορούν να εφαρμοστούν σε κάθε περίπτωση με ένα συστηματικό τρόπο και επιτυγχάνουν εγγυημένα αποτελέσματα. Επειδή αυτές οι μέθοδοι είναι συστηματικές, έχουν αναπτυχθεί εργαλεία λογισμικού τα οποία απλοποιούν και επιταχύνουν την υλοποίησή τους.

Γενικές οδηγίες

- Να αποφεύγετε κυκλώματα που έχουν λογικό πλεονασμό
- Να αποφεύγετε την ασύγχρονη λογική
- Να απομονώνετε τα σήματα χρονισμού από τα δεδομένα
- Να αποφεύγονται πύλες με μεγάλο πλήθος εισόδων
- Να προσθέτετε εισόδους ελέγχου για σήματα που είναι δύσκολο να ελεγχθούν
- Να σχεδιάζετε με τρόπο που διευκολύνεται η διάγνωση

Υπάρχουν περιπτώσεις στις οποίες ο λογικός πλεονασμός προστίθεται για να λύσει κάποια προβλήματα. Σ' αυτές τις περιπτώσεις δεν μπορούμε να τον αποφύγουμε. Θα πρέπει όμως να αποφεύγουμε το λογικό πλεονασμό που προστίθεται στο κύκλωμα λόγω κακής σχεδίασης. Όταν στο κύκλωμά μας υπάρχει λογικός πλεονασμός, υπάρχουν σφάλματα τα οποία δεν είναι δυνατόν να ανιχνευτούν. Ένας ντετερμινιστικός αλγόριθμος εξαγωγής διανυσμάτων δοκιμής για να αποφασίσει ότι ένα σφάλμα δεν ανιχνεύεται θα πρέπει να ψάξει όλο τον χώρο των λύσεων. Αυτό έχει ως συνέπεια ο χρόνος εξαγωγής διανυσμάτων δοκιμής να είναι πολύ μεγάλος.

Είναι πιο δύσκολο να ελέγξουμε την ορθή λειτουργία της ασύγχρονης λογικής. Για να είναι δυνατή η εισαγωγή μονοπατιών ολίσθησης δεν πρέπει να εμπλέκουν τα

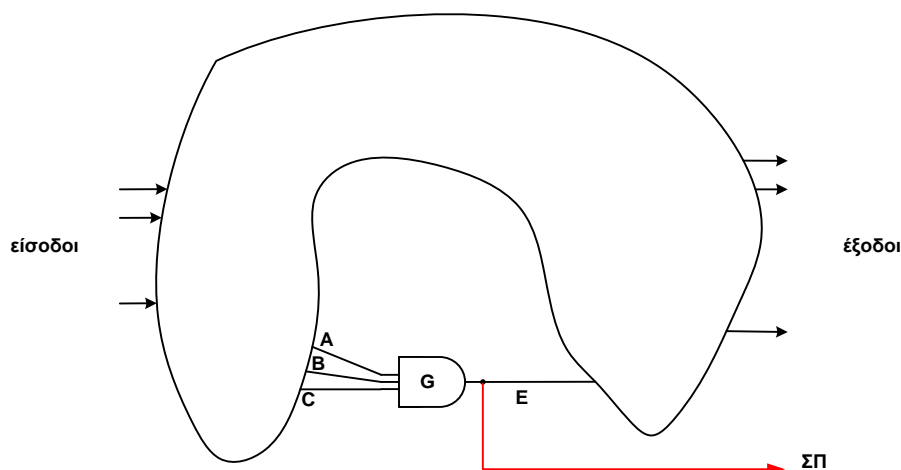
σήματα χρονισμού με τα σήματα δεδομένου. Όπως έχουμε ήδη διαπιστώσει, πύλες με μεγάλο αριθμό εισόδων έχουν ως συνέπεια την ύπαρξη σφαλμάτων που δεν ανιχνεύονται εύκολα ψευδοτυχαία διανύσματα δοκιμής. Εισάγοντας εισόδους ελέγχου αυξάνεται η ελεγχσιμότητα του κυκλώματος.

Η διάγνωση βοηθάει στον εντοπισμό των σφαλμάτων. Σε επίπεδο συστήματος αυτό προσφέρει τη δυνατότητα γρήγορου εντοπισμού της μονάδας με το ελάττωμα και την αντικατάστασή της. Σε επίπεδο ολοκληρωμένου κυκλώματος βοηθάει στο να βρούμε την αιτία του σφάλματος ώστε να βελτιώσουμε τη διαδικασία κατασκευής.

Κεφάλαιο 6. Εισαγωγή σημείων δοκιμής και ψευδο-εξαντλητικός έλεγχος

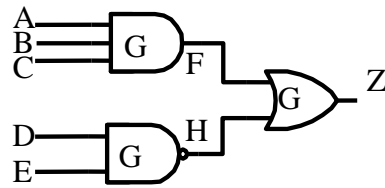
Στις τεχνικές που εφαρμόζονται ανάλογα με την περίπτωση ανήκουν οι τεχνικές της εισαγωγής σημείων δοκιμής και οι τεχνικές ψευδοεξαντλητικού ελέγχου.

Τα σημεία δοκιμής διακρίνονται σε σημεία παρατήρησης, τα οποία εισάγονται για να αυξήσουμε την παρατηρησιμότητα των εσωτερικών κόμβων ενός κυκλώματος και τα σημεία ελέγχου, τα οποία εισάγονται για να αυξήσουμε την ελεγχσιμότητα του κυκλώματος.

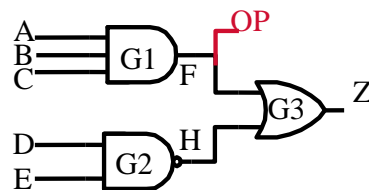


Σχήμα 6.1

Στο σχήμα 6.1 θεωρούμε ότι ο εσωτερικός κόμβος E του κυκλώματος έχει μικρή παρατηρησιμότητα και εισάγουμε ένα σημείο παρατήρησης ΣΠ για να αυξήσουμε την παρατηρησιμότητά του. Εισάγοντας κατάλληλα σημεία παρατήρησης μπορούμε να ελαττώσουμε το μέγεθος του συνόλου δοκιμής, επίσης μπορούμε να ανιχνεύσουμε σφάλματα τα οποία πριν ήταν μη ανιχνεύσιμα. Για παράδειγμα για την ανίχνευση όλων των απλών σφαλμάτων μόνιμης τιμής του κυκλώματος του σχήματος 6.2 απαιτούνται έξι διανύσματα δοκιμής 10111, 11011, 00110, 11111, 01101 και 01111. Εισάγοντας ένα σημείο παρατήρησης στην έξοδο της πύλης G1 (βλέπε σχήμα 6.3) το πλήθος των απαιτούμενων διανυσμάτων δοκιμής μειώνεται σε πέντε 11111, 01111, 10101, 11001 και 00010. Προφανώς, η μείωση κατά ένα των διανυσμάτων δοκιμής, δεν δικαιολογεί την εισαγωγή ενός σημείου παρατήρησης. Το παράδειγμα είναι πολύ μικρό για να έχουμε σημαντικά οφέλη.

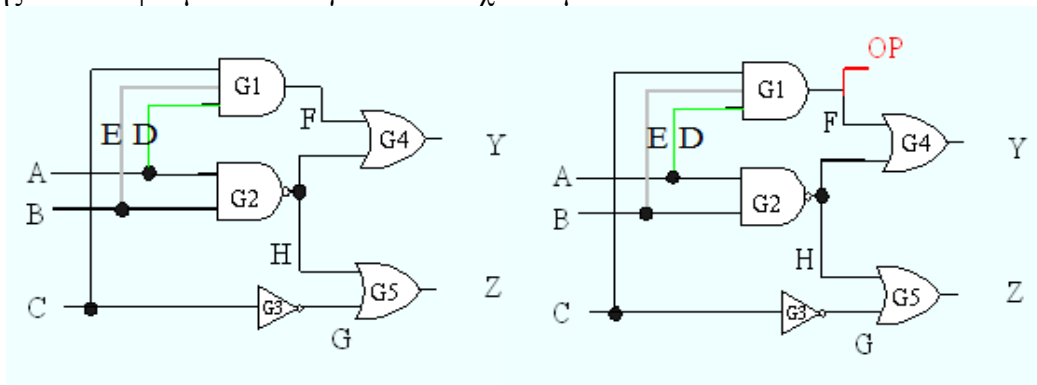


Σχήμα 6.2



Σχήμα 6.3

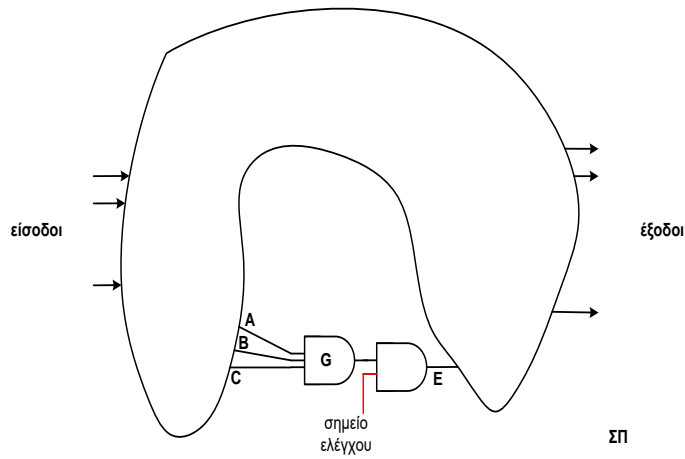
Στο κύκλωμα του σχήματος 6.4 τα απλά σφάλματα μόνιμης τιμής «η γραμμή E έχει κολλήσει στη λογική τιμή 1» ή «η γραμμή D έχει κολλήσει στη λογική τιμή 1» δεν ανιχνεύονται. Εισάγοντας ένα σημείο παρατήρησης στην έξοδο της λογικής πύλης G1 τα σφάλματα αυτά γίνονται ανιχνεύσιμα.



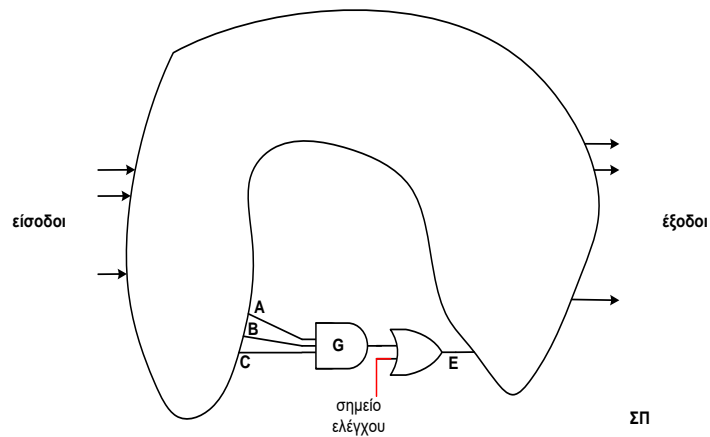
Σχήμα 6.4

Η εισαγωγή σημείων ελέγχου οδηγεί σε αύξηση της ελεγχιμότητας των κόμβων του κυκλώματος. Υποθέστε για παράδειγμα ότι είναι δύσκολο να θέσουμε τον κόμβο E του κυκλώματος του σχήματος 6.1 στη λογική τιμή 0. Μπορούμε να το κάνουμε εύκολα εισάγοντας ένα σημείο ελεγχιμότητας. Αυτό γίνεται εισάγοντας στο κύκλωμα μια λογική πύλη AND, όπως φαίνεται στο σχήμα 6.5. Θέτοντας το σημείο ελέγχου στη λογική τιμή 0 θέτουμε τον κόμβο E επίσης στη λογική τιμή 0.

Εάν στο κύκλωμα του σχήματος 6.1, αντί να θέλουμε να αυξήσουμε την ελεγχιμότητα του κόμβου E για την τιμή 0, επιθυμούσαμε να αυξήσουμε την ελεγχιμότητά του για τη λογική τιμή 1, τότε θα προσθέταμε στο κύκλωμα μια πύλη OR όπως φαίνεται στο σχήμα 6.6. Θέτοντας το σημείο ελέγχου στη λογική τιμή 1, θέτουμε τον κόμβο E επίσης στη λογική τιμή 1.

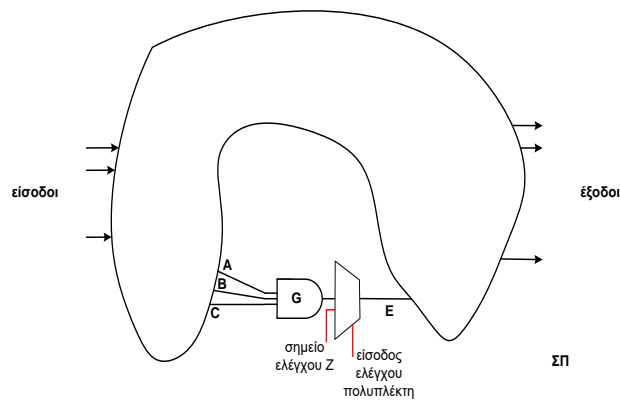


Σχήμα 6.5



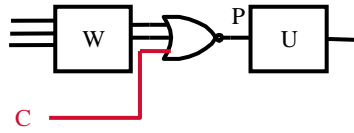
Σχήμα 6.6

Εάν θέλουμε να αυξήσουμε την ελεγχσιμότητα του κόμβου E τόσο για τη λογική τιμή 0 όσο και για τη λογική τιμή ένα, τότε θα προσθέταμε στο κύκλωμα ένα πολυπλέκτη, όπως φαίνεται στο σχήμα. Θέτοντας την είσοδο ελέγχου του πολυπλέκτη στη λογική τιμή 1, θέτουμε τον κόμβο E στη λογική τιμή της εισόδου ελέγχου Z.



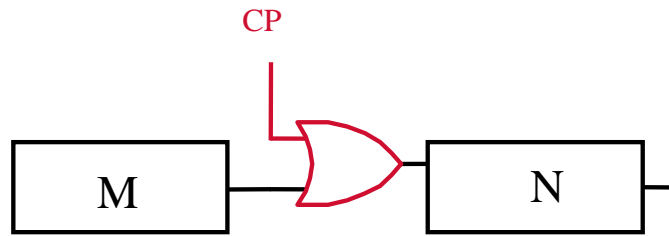
Σχήμα 6.7

Σε κάποιες περιπτώσεις για να εισάγουμε ένα σημείο ελέγχου δεν χρειάζεται να εισάγουμε πρόσθετη πύλη, αρκεί να αυξήσουμε τις εισόδους μιας πύλης (βλέπε σχήμα 6.8).



Σχήμα 6.8

Θεωρήστε ένα μετρητή των k δυαδικών ψηφίων. Τότε για να ελέγξουμε την ορθή λειτουργία του μετρητή, πρέπει να τον αφήσουμε να τρέξει από το 0 έως και το 2^k-1 . Εάν η τιμή του k είναι μεγάλη, τότε απαιτείται απαγορευτικά μεγάλος χρόνος. Το πρόβλημα λύνεται εάν σχεδιάσουμε τον μετρητή μας σαν να αποτελείται από δύο τμήματα, κάθε ένα με μήκος $k/2$, και συνδέσουμε τα δύο τμήματα μέσω μιας λογικής πύλης OR, όπως φαίνεται στο σχήμα 6.9.



Σχήμα 6.9

Όταν, ο μετρητής είναι σε κανονική λειτουργία οδηγούμε την είσοδο ελέγχου EE με τη λογική τιμή 0. Κατά τον έλεγχο της ορθής λειτουργίας θέτουμε την είσοδο ελέγχου EE στη λογική τιμή 1. Τότε τα δύο τμήματα του μετρητή αυξάνουν την τιμή τους σε κάθε μετάβαση του σήματος χρονισμού ανεξάρτητα. Επομένως για τον έλεγχο της ορθής λειτουργίας του μετρητή απαιτούνται μόνο $2^{k/2}$ κύκλοι ρολογιού.

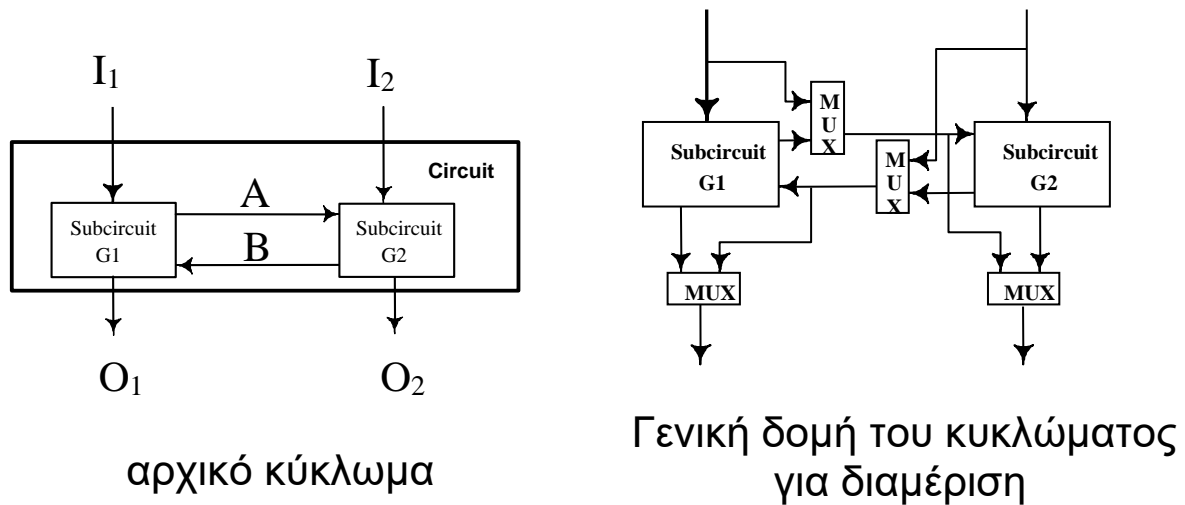
Ψευδοεξαντλητικός έλεγχος

Θα εξετάσουμε τις ακόλουθες μεθόδους ψευδοεξαντλητικού ελέγχου:

- Διαμέριση του κυκλώματος με υλικό (Hardware partitioning)
- Διαμέριση του κυκλώματος μέσω της ευαισθητοποίησης μονοπατιών (Sensitized path segmentation)
- Διαμέριση του κυκλώματος μέσω της ευαισθητοποίησης μονοπατιών αλλά και χρήσης υλικού (Partial Hardware Partitioning)
- Διαμέριση του κυκλώματος που βασίζεται στη σχέση εισόδων και εξόδων (Verification testing ή cone segmentation)

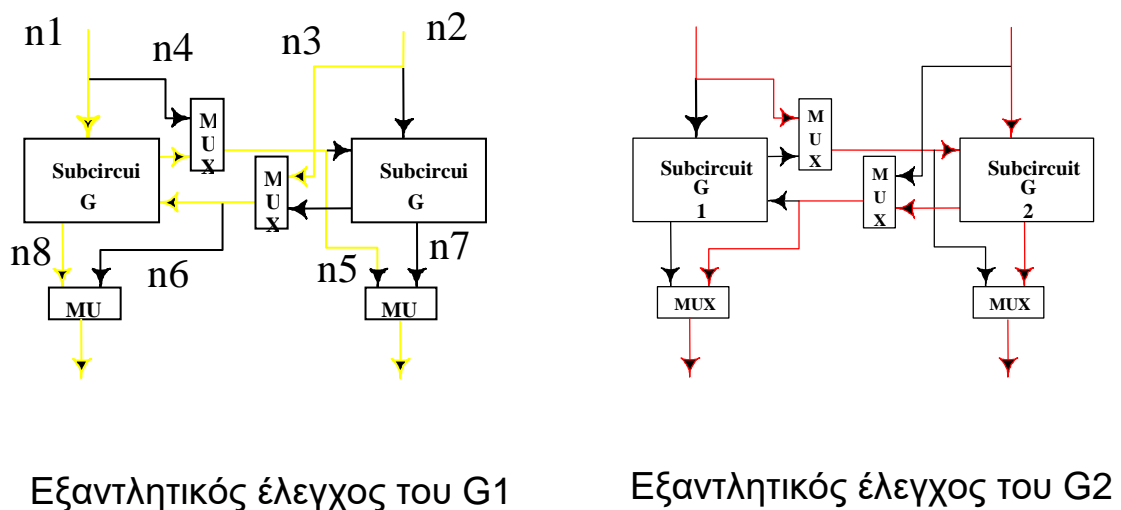
Όπως έχουμε ήδη αναφέρει το μεγάλο πρόβλημα του εξαντλητικού ελέγχου είναι ότι δεν μπορεί να εφαρμοστεί σε κυκλώματα με μεγάλο αριθμό κυρίων εισόδων. Το πρόβλημα είναι δυνατόν να λυθεί εάν διαμερίσουμε το κύκλωμα σε υποκυκλώματα έτσι ώστε το πλήθος των εισόδων κάθε υποκυκλώματος να είναι σημαντικά μικρότερο του πλήθους των εισόδων του αρχικού κυκλώματος. Στην περίπτωση που

τα υποκυκλώματα επικοινωνούν μεταξύ τους είναι απαραίτητη η χρησιμοποίηση πολυπλεκτών (βλέπε σχήμα 6.10).



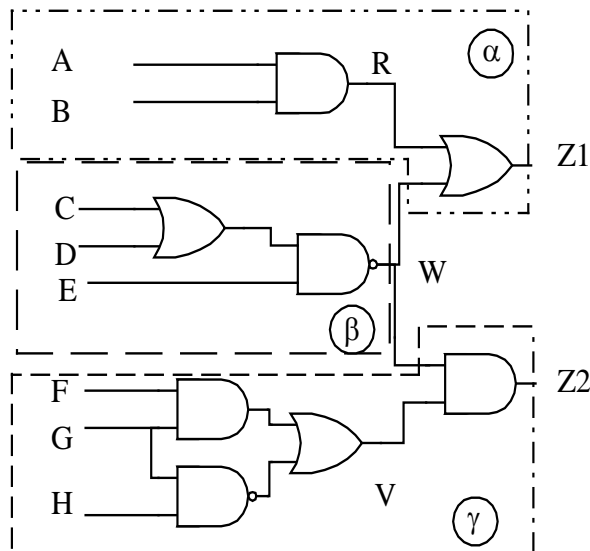
Σχήμα 6.10

Στο σχήμα 6.11 βλέπουμε ότι το αρχικό κύκλωμα έχει διαμεριστεί σε δύο υποκυκλώματα και τον τρόπο χρησιμοποίησης των πολυπλεκτών.



Σχήμα 6.11

Στο αριστερό τμήμα του σχήματος 6.11 βλέπουμε με κίτρινο χρώμα τη ροή των δεδομένων δοκιμής και της απόκρισης όταν ελέγχεται εξαντλητικά η ορθή λειτουργία του υποκυκλώματος G1. Στο τμήμα του σχήματος βλέπουμε με κόκκινο χρώμα τη ροή των δεδομένων δοκιμής και της απόκρισης, όταν ελέγχεται εξαντλητικά η ορθή λειτουργία του υποκυκλώματος G2.



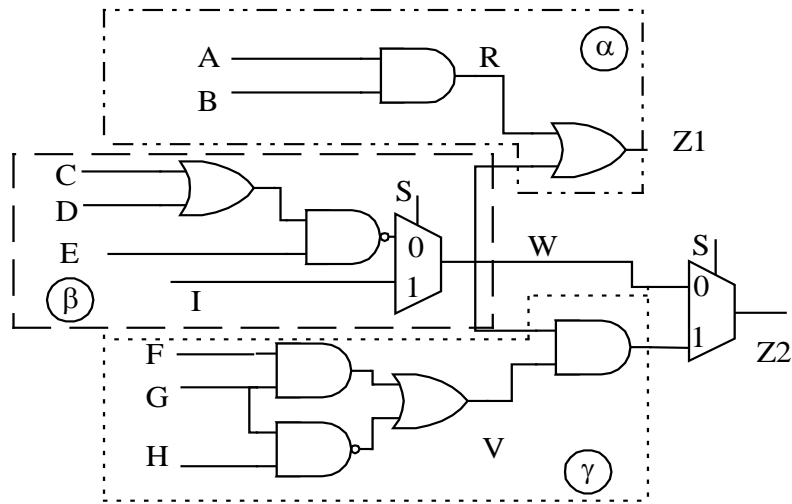
Σχήμα 6.12

Το κύκλωμα του σχήματος 6.12 έχει 8 κύριες εισόδους, επομένως για τον εξαντλητικό έλεγχο της ορθής του λειτουργίας απαιτούνται 256 διανύσματα δοκιμής. Παρατηρούμε ότι μπορούμε να διαμερίσουμε το κύκλωμα σε τρία υποκύκλωμα το α, το β και το γ. Θέτοντας τις εισόδους C, D και E στη λογική τιμή 1, ο κόμβος W θα λάβει τη λογική τιμή 0. Επομένως με τέσσερα διανύσματα εισόδου μπορούμε να ελέγξουμε εξαντλητικά το υποκύκλωμα α. Στη συνέχεια, θέτοντας τις εισόδους A και B στη λογική τιμή 0, ο κόμβος R θα λάβει τη λογική τιμή 0. Επομένως μέσω της εξόδου Z1 μπορούμε με οκτώ διανύσματα εισόδου να ελέγξουμε εξαντλητικά το υποκύκλωμα β.

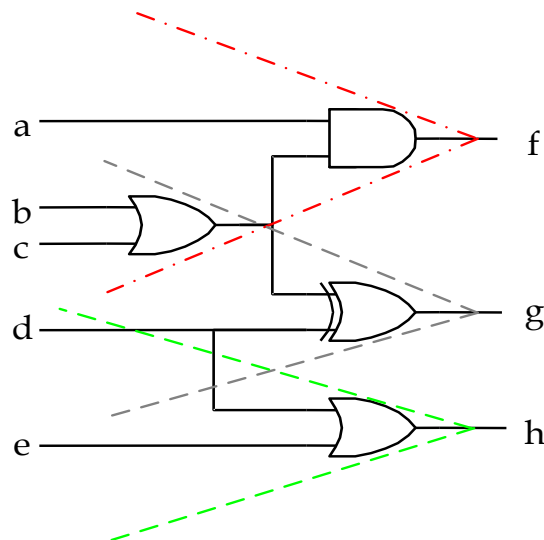
Στη συνέχεια, θέτοντας τις εισόδους C και D στη λογική τιμή 0 ή την είσοδο E στη λογική τιμή 0, ο κόμβος W θα λάβει τη λογική τιμή 1. Επομένως μέσω της εξόδου Z2 μπορούμε με οκτώ διανύσματα εισόδου να ελέγξουμε εξαντλητικά το υποκύκλωμα γ. Επομένως για τον ψευδοεξαντλητικό έλεγχο απαιτούνται συνολικά 20 διανύσματα δοκιμής, δηλαδή σε σχέση με τον εξαντλητικό έλεγχο ορθής λειτουργίας, το πλήθος των διανυσμάτων δοκιμής έχει μειωθεί περισσότερο από δέκα φορές.

Στην περίπτωση μας το υπό έλεγχο κύκλωμα ήταν σχετικά απλό. Υπάρχουν περιπτώσεις στις οποίες δεν είναι εύκολο να οδηγήσουμε την απόκριση ενός υποκύκλωματος, για παράδειγμα του υποκύκλωματος β στη συγκεκριμένη περίπτωση, μέχρι κάποια κύρια έξοδο του κυκλώματος μέσω κάποιων πυλών άλλου υποκύκλωματος. Σ' αυτή την περίπτωση για τη διευκόλυνση της ενεργοποίησης κάποιων μονοπατιών προσθέτουμε και κάποιους πολυπλέκτες (βλέπε σχήμα 6.13).

Στο κύκλωμα του σχήματος 6.13 παρατηρούμε ότι θέτοντας το σήμα ελέγχου S και την είσοδο δεδομένων I του πολυπλέκτη με έξοδο W στη λογική τιμή 1 και 0 αντίστοιχα, μπορούμε να ελέγξουμε εξαντλητικά το υποκύκλωμα α. Θέτοντας το σήμα ελέγχου S του πολυπλέκτη με έξοδο W στη λογική τιμή 0 και το σήμα ελέγχου S του πολυπλέκτη με έξοδο Z2 στη λογική 0, μπορούμε να ελέγξουμε εξαντλητικά το υποκύκλωμα β. Επίσης, θέτοντας το σήμα ελέγχου S και την είσοδο δεδομένων I του πολυπλέκτη με έξοδο W στη λογική τιμή ένα και το σήμα ελέγχου S του πολυπλέκτη με έξοδο Z2 στη λογική μηδέν, μπορούμε να ελέγξουμε εξαντλητικά το υποκύκλωμα γ.



Σχήμα 6.13



Σχήμα 6.14

Το κύκλωμα του σχήματος 6.14 έχει πέντε εισόδους, επομένως για τον εξαντλητικό του έλεγχο ορθής λειτουργίας απαιτούνται 32 διανύσματα δοκιμής. Παρατηρούμε όμως ότι η έξοδος του f εξαρτάται μόνο από τις εισόδους του a, b, και c. Επομένως για τον εξαντλητικό έλεγχο όλων των πυλών του κυκλώματος που οδηγούν στην έξοδο f απαιτούνται 8 διανύσματα δοκιμής. Η έξοδος του g εξαρτάται μόνο από τις εισόδους b, c και d. Επομένως για τον εξαντλητικό έλεγχο όλων των πυλών του κυκλώματος που οδηγούν στην έξοδο g απαιτούνται 8 διανύσματα δοκιμής. Η έξοδος του h εξαρτάται μόνο από τις εισόδους d και e. Επομένως για τον εξαντλητικό έλεγχο όλων των πυλών του κυκλώματος που οδηγούν στην έξοδο h απαιτούνται 4 διανύσματα δοκιμής. Άρα συνολικά απαιτούνται 20 διανύσματα δοκιμής.

Είναι εύκολο να δούμε ότι εάν δύο κύριες εισοδοί του κυκλώματος δεν επηρεάζουν την ίδια κύρια έξοδό του, τότε κατά τον έλεγχο της ορθής λειτουργίας

του κυκλώματος μπορούν να οδηγούνται με το ίδιο σήμα (μέγιστη ταύτιση σημάτων δοκιμής).

Όπως είδαμε, για το κύκλωμα του σχήματος 6.14 στην περίπτωση εξαντλητικού ελέγχου απαιτούνται 32 διανύσματα δοκιμής ενώ στην περίπτωση ψευδοεξαντλητικού ελέγχου απαιτούνται 20 διανύσματα δοκιμής. Το πλήθος των απαιτούμενων διανυσμάτων δοκιμής μπορεί να μειωθεί περαιτέρω αν λάβουμε υπόψη μας την παρατήρηση της μέγιστης ταύτισης σημάτων δοκιμής. Για το σκοπό αυτό φτιάχνουμε τον πίνακα εξάρτησης (dependency matrix). Ο πίνακας αυτός έχει μια γραμμή για κάθε κύρια έξοδο του κυκλώματος και μια στήλη για κάθε κύρια είσοδο του κυκλώματος. Μια θέση i, j του πίνακα έχει την τιμή ένα εάν η γραμμή i αντιστοιχεί σε μια κύρια έξοδο του κυκλώματος η οποία εξαρτάται από την κυρία είσοδο του κυκλώματος στην οποία αντιστοιχεί η στήλη j .

Για να αναγνωρίσουμε τις κύριες εισόδους του κυκλώματος, οι οποίες κατά τον έλεγχο της ορθής του λειτουργίας μπορούν να οδηγούνται με το ίδιο σήμα, θα πρέπει να διαμερίσουμε τον πίνακα σε υποπίνακες, έτσι ώστε σε κάθε υποπίνακα να μην υπάρχει κάποια γραμμή που να περιέχει περισσότερες από μία μονάδες. Τότε σε κάθε υποπίνακα υπάρχουν μόνο γραμμές εισόδου (στήλες) οι οποίες δεν επηρεάζουν την ίδια έξοδο και επομένως μπορούν κατά τον έλεγχο της ορθής λειτουργίας του κυκλώματος να οδηγούνται από το ίδιο σήμα. Επομένως σε όσο μικρότερο αριθμό υποπινάκων μπορούμε να διαμερίσουμε τον πίνακα εξάρτησης, σε τόσο μικρότερο σύνολο δοκιμής θα οδηγηθούμε.

	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>a</i>	<i>d</i>	<i>b</i>	<i>c</i>	<i>e</i>	<i>a</i>	<i>e</i>	<i>b</i>	<i>c</i>	<i>d</i>
<i>f</i>	1	1	1	0	0	1	0	1	1	0	1	0	1	1	0
<i>g</i>	0	1	1	1	0	0	1	1	1	0	0	0	1	1	1
<i>h</i>	0	0	0	1	1	0	1	0	0	1	0	1	0	0	1

(a) (b)

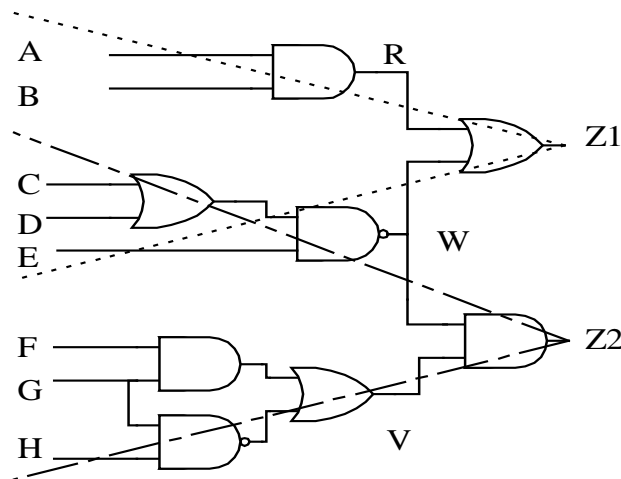
Σχήμα 6.15

Από τον πίνακα εξάρτησης, σχήμα 6.15.α, προκύπτουν δύο διαμερίσεις, σχήμα 6.15 b, εκ των οποίων η πρώτη οδηγεί σε μικρότερο αριθμό ανεξάρτητων σημάτων, συγκεκριμένα σε τρία ανεξάρτητα σήματα διότι έχουμε τρεις υποπίνακες, άρα απαιτούνται 8 διανύσματα δοκιμής. Στο σύνολο δοκιμής (βλέπε σχήμα 6.16) μπορούμε να δούμε ότι οι εισοδοί *a* και *d* λαμβάνουν τις ίδιες τιμές. Το ίδιο ισχύει για τις εισόδους *c* και *e*.

<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	0	1
1	0	0	1	0
1	0	1	1	1
1	1	0	1	0
1	1	1	1	1

Σχήμα 6.16

Το κύκλωμα του σχήματος 6.17 έχει οκτώ εισόδους επομένως ο εξαντλητικός έλεγχος απαιτεί 256 διανύσματα δοκιμής. Ο ψευδοεξαντλητικός έλεγχος απαιτεί 96 διανύσματα δοκιμής. Από το κύκλωμα σχεδιάσαμε τον πίνακα εξάρτησης, σχήμα 6.18.α, και προσπαθήσαμε να τον διαμερίσουμε σε όσο το δυνατόν λιγότερους υποπίνακες, σχήμα 6.18.β, έτσι ώστε κάθε υποπίνακας να μην έχει περισσότερες από μια μονάδες σε κάθε γραμμή του. Από τη διαμέριση προέκυψαν έξι υποπίνακες. Επομένως το πλήθος των απαιτούμενων διανυσμάτων δοκιμής είναι 64.



Σχήμα 6.17

	A	B	C	D	E	F	G	H								
Z ₁	1	1	1	1	1	0	0	0	A	F	B	G	C	D	E	H
Z ₂	0	0	1	1	1	1	1	1	1	0	0	1	1	1	1	1

(a) (b)

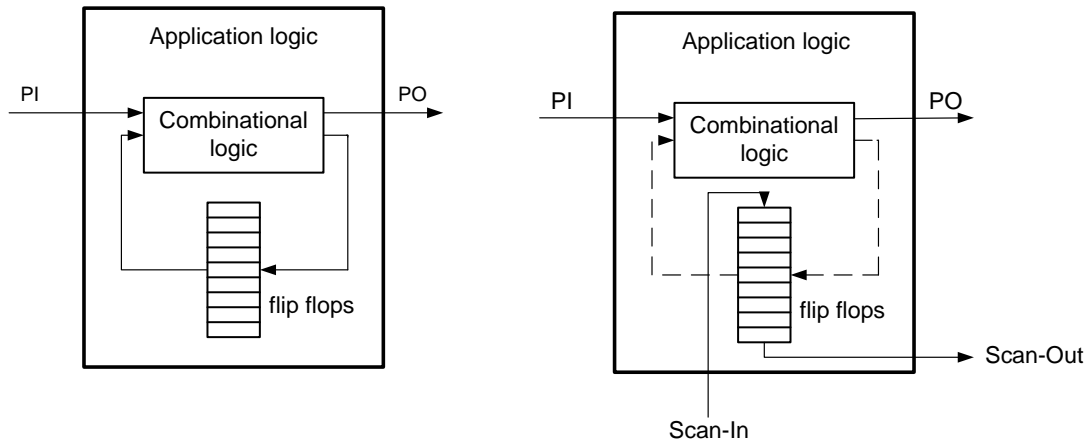
Σχήμα 6.18

Κεφάλαιο 7. Τεχνικές σχεδιασμού μονοπατιών ολίσθησης (Scan Path Design Techniques)

Όπως γνωρίζετε η απόκριση ενός ακολουθιακού κυκλώματος εξαρτάται τόσο από τις τιμές των εισόδων του όσο και από την εσωτερική κατάσταση στην οποία βρίσκεται το ακολουθιακό κύκλωμα. Επομένως για να εφαρμόσουμε την κλασική διαδικασία του λογικού ελέγχου ορθής λειτουργίας, θα πρέπει να θέσουμε τόσο τις εισόδους του κυκλώματος όσο και την εσωτερική του κατάσταση σε γνωστές τιμές και να διαβάσουμε την απόκριση του κυκλώματος, η οποία στην περίπτωση αυτή συνίσταται από δύο συνιστώσες, τις τιμές των κύριων εξόδων του κυκλώματος και τη νέα εσωτερική κατάσταση.

Ωστόσο είναι δύσκολο να θέσουμε το ακολουθιακό κύκλωμα σε μια επιθυμητή κατάσταση (για να το κάνουμε αυτό απαιτείται να εφαρμόσουμε στις εισόδους του κυκλώματος μεγάλου μήκους ακολουθίες) και επίσης είναι δύσκολο να

διαπιστώσουμε την εσωτερική κατάσταση στην οποία βρίσκεται ένα ακολουθιακό κύκλωμα.

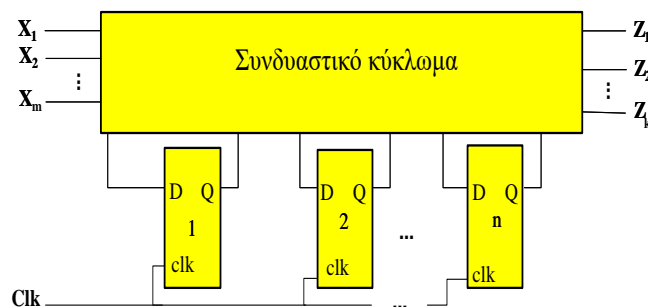


Σχήμα 7.1 Μονοπάτι ολίσθησης (Scan path).

Σ' ένα ακολουθιακό κύκλωμα χρησιμοποιούνται στοιχεία μνήμης για την αποθήκευση της εσωτερικής κατάστασης (αριστερά στο σχήμα 7.1). Κατά την κανονική λειτουργία του κυκλώματος τα στοιχεία μνήμης αποτελούν έναν καταχωρητή ο οποίος φορτώνεται παράλληλα. Είναι δυνατόν να αντικαταστήσουμε τα στοιχεία μνήμης του ακολουθιακού κυκλώματος με άλλα κατάλληλα στοιχεία μνήμης, ώστε ο καταχωρητής αποθήκευσης της εσωτερικής κατάστασης να έχει δύο τρόπους λειτουργίας (δεξιά στο σχήμα 7.1). Ο ένας τρόπος είναι αυτός της παράλληλης φόρτωσης ο οποίος χρησιμοποιείται κατά την κανονική λειτουργία του συστήματος (διακεκομμένες γραμμές στο δεξιό σχήμα). Ο καταχωρητής όμως έχει τη δυνατότητα να λειτουργεί και ως ένας ολισθητής (συνεχής γραμμή στο δεξιό σχήμα). Προφανώς θα υπάρχει κάποιο σήμα ελέγχου, η τιμή του οποίου θα καθορίζει τον τρόπο λειτουργίας του καταχωρητή (το ανάλογο σχήμα θα το δούμε αργότερα).

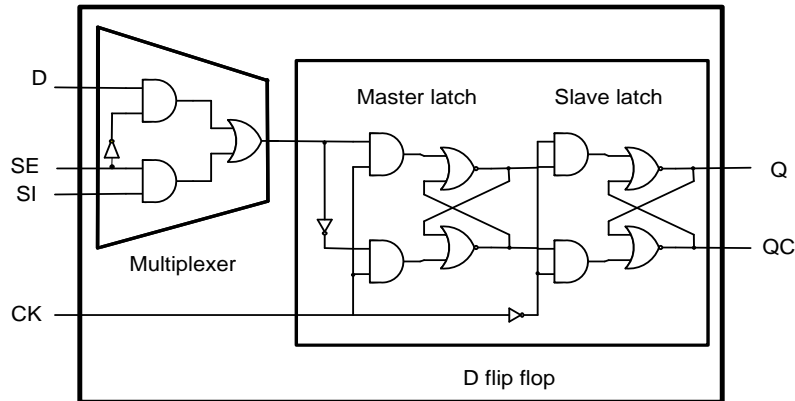
Επομένως, όταν ο καταχωρητής λειτουργεί ως ολισθητής, δημιουργεί ένα μονοπάτι ολίσθησης μέσω του οποίου μπορούμε να θέτουμε σειριακά το ακολουθιακό κύκλωμα σε μια επιθυμητή εσωτερική κατάσταση, καθώς επίσης και να διαβάζουμε σειριακά την εσωτερική κατάσταση στην οποία βρίσκεται το ακολουθιακό κύκλωμα. Επομένως να ελεγχθεί η ορθή λειτουργία ενός ακολουθιακού κυκλώματος σαν να είναι συνδυαστικό κύκλωμα.

Μπορούμε να ταξινομήσουμε τις μηχανές περασμένων καταστάσεων σε αυτές που βασίζονται στη χρήση φλιπ-φλοπ και σε αυτές που βασίζονται στη χρήση μανδαλωτών.



Σχήμα 7.2 Δομή μιας μηχανής που βασίζεται στη χρήση φλιπ-φλοπ.

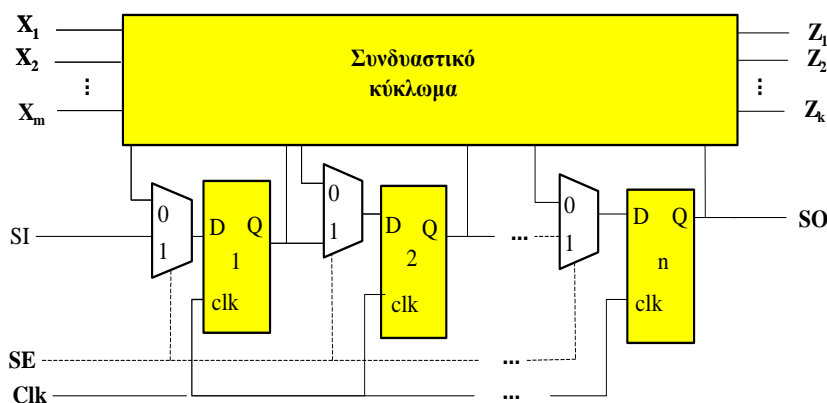
Στο σχήμα 7.2 βλέπουμε τη δομή μιας μηχανής που βασίζεται στη χρήση φλιπ-φλοπ. Για να πετύχουμε ο εσωτερικός καταχωρητής του ακολουθιακού κυκλώματος να έχει δύο τρόπους λειτουργίας, θα πρέπει να αντικαταστήσουμε τα φλιπ-φλοπ του συστήματος με άλλα κατάλληλα φλιπ-φλοπ, τα οποία θα καλούμε φλιπ-φλοπ ολίσθησης (scan flip-flops). Υπάρχουν γενικά δύο φλιπ-φλοπ που μπορούμε να χρησιμοποιήσουμε. Τα φλιπ-φλοπ που διαθέτουν πολυπλέκτη στην είσοδό τους και τα φλιπ-φλοπ δύο εισόδων.



Σχήμα 7.3 Φλιπ-φλοπ ολίσθησης με πολυπλέκτη

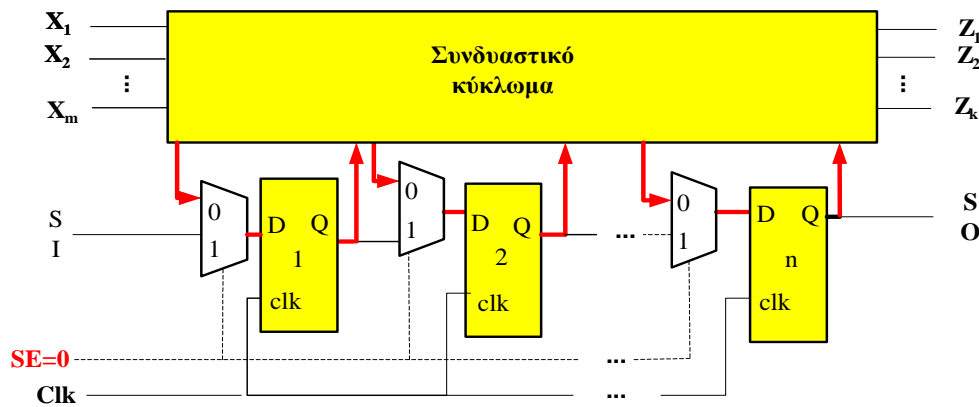
Στο σχήμα 7.3 δίνεται σε επίπεδο λογικών πυλών η δομή ενός φλιπ-φλοπ ολίσθησης με πολυπλέκτη. Παρατηρούμε ότι αποτελείται από έναν πολυπλέκτη και ένα D φλιπ-φλοπ. Όταν η είσοδος ελέγχου SE έχει τη λογική τιμή 0, τότε κατά τη μετάβαση του σήματος χρονισμού CK από 0 σε 1 στην έξοδο του φλιπ-φλοπ αποθηκεύεται η τιμή που οδηγεί την είσοδο D. Όταν η είσοδος ελέγχου SE έχει τη λογική τιμή 1, τότε κατά τη μετάβαση του σήματος χρονισμού CK από 0 σε 1 στην έξοδο του φλιπ-φλοπ αποθηκεύεται η τιμή που οδηγεί την είσοδο SI.

Συγκρίνοντας το φλιπ-φλοπ ολίσθησης με πολυπλέκτη με το φλιπ-φλοπ του συστήματος παρατηρούμε ότι έχει μεγαλύτερο κόστος υλοποίησης λόγω των πυλών που απαιτούνται για την υλοποίηση του πολυπλέκτη. Επίσης βάζει μια πρόσθετη καθυστέρηση ίση με την καθυστέρηση του πολυπλέκτη.



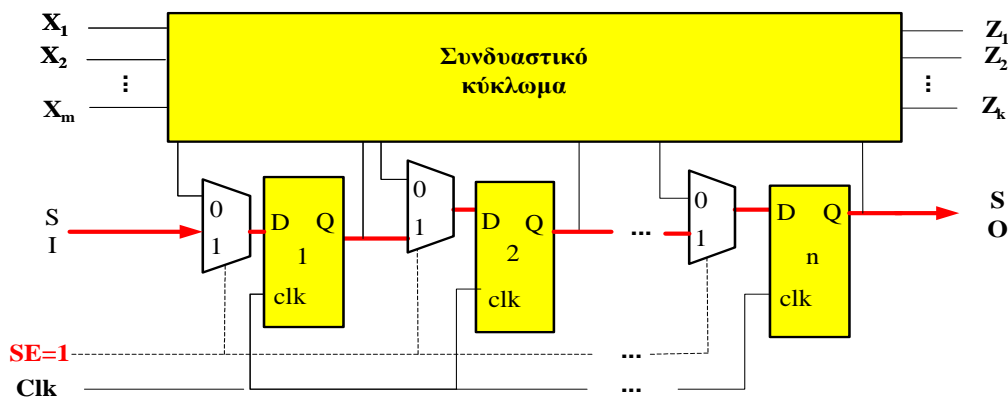
Σχήμα 7.4 Υλοποίηση μονοπατιού ολίσθησης χρησιμοποιώντας φλιπ-φλοπ ολίσθησης με πολυπλέκτη

Στο σχήμα 7.4 τα φλιπ-φλοπ του συστήματος έχουν αντικατασταθεί από φλιπ-φλοπ ολίσθησης με πολυπλέκτη. Παρατηρούμε ότι τα φλιπ-φλοπ ολίσθησης έχουν συνδεθεί μεταξύ τους έτσι ώστε η έξοδος καθενός φλιπ-φλοπ να οδηγεί μια είσοδο του πολυπλέκτη ενός άλλου φλιπ-φλοπ ολίσθησης, εκτός βέβαια από ένα φλιπ-φλοπ ολίσθησης του οποίου η έξοδος αποτελεί κύρια έξοδο του κυκλώματος, SO. Επίσης ενός φλιπ-φλοπ ολίσθησης η μία είσοδος του πολυπλέκτη αποτελεί κύρια είσοδο του κυκλώματος, SI. Παρατηρούμε επίσης ότι, επιπλέον της κύριας εισόδου SI και της κύριας εξόδου SO που έχουν προστεθεί, έχει προστεθεί και η κύρια είσοδος SE. Η τιμή της εισόδου SE καθορίζει τον τρόπο λειτουργίας του μονοπατιού ολίσθησης.



Σχήμα 7.5 Κανονικός τρόπος λειτουργίας.

Όταν η είσοδος SE οδηγείται με τη λογική τιμή 0, τότε το κύκλωμα βρίσκεται σε κανονική λειτουργία και στα φλιπ-φλοπ ολίσθησης κατά τη μετάβαση του σήματος χρονισμού CK από 0 σε 1 αποθηκεύονται τα σήματα που παράγονται από το συνδυαστικό τμήμα του ακολουθιακού κυκλώματος (σχήμα 7.5).



Σχήμα 7.6 Λειτουργία ολίσθησης.

Όταν η είσοδος SE οδηγείται με τη λογική τιμή 1, τότε τα φλιπ-φλοπ ολίσθησης σχηματίζουν ένα μονοπάτι ολίσθησης με κύρια είσοδο τον ακροδέκτη SI και κύρια έξοδο τον ακροδέκτη SO, όπως φαίνεται στο σχήμα. Κατά τη μετάβαση του σήματος χρονισμού CK από 0 σε 1 γίνεται ολίσθηση, κατά ένα δυαδικό ψηφίο, από την κύρια είσοδο SI προς την κύρια έξοδο SO (σχήμα 7.6).

Τώρα το υπό έλεγχο κύκλωμα θεωρούμε ότι αποτελείται από δύο τμήματα, το συνδυαστικό τμήμα και τα φλιπ-φλοπ.

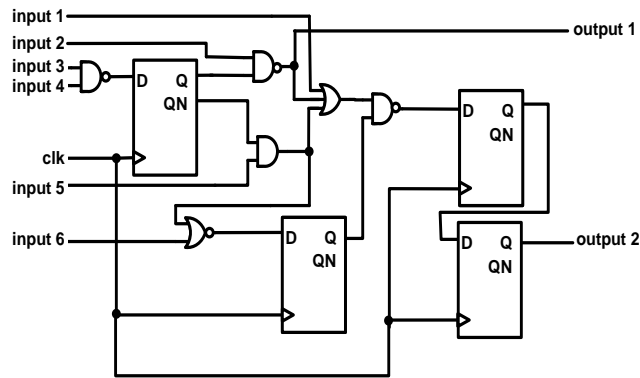
Ο έλεγχος ορθής λειτουργίας κάθε τμήματος γίνεται ξεχωριστά. Η είσοδος κάθε φλιπ-φλοπ θεωρείται ως μια έξοδος του συνδυαστικού τμήματος του υπό έλεγχο κυκλώματος. Αυτές τις εξόδους τις καλούμε ψευδοεξόδους του ακολουθιακού κυκλώματος. Η έξοδος κάθε φλιπ-φλοπ θεωρείται ως μια είσοδος του συνδυαστικού τμήματος του υπό έλεγχο κυκλώματος. Αυτές τις εισόδους τις καλούμε ψευδοεισόδους του ακολουθιακού κυκλώματος. Επομένως κάθε διάνυσμα δοκιμής του συνδυαστικού τμήματος αποτελείται από δύο τμήματα, το τμήμα που οδηγεί τις κύριες εισόδους του ακολουθιακού κυκλώματος και το κομμάτι που οδηγεί τις ψευδοεισόδους. Επίσης κάθε απόκριση του συνδυαστικού τμήματος αποτελείται από δύο τμήματα, το τμήμα που εμφανίζεται στις κύριες εξόδους του ακολουθιακού κυκλώματος και αυτό που εμφανίζεται στις ψευδοεξόδους και αποθηκεύεται στο καταχωρητή εσωτερικής κατάστασης.

Για τον έλεγχο της ορθής λειτουργίας των φλιπ-φλοπ πρέπει σε κάθε ένα φλιπ-φλοπ να εφαρμοστούν όλες οι δυνατές μεταβάσεις $0 \rightarrow 0$, $0 \rightarrow 1$, $1 \rightarrow 1$ και $1 \rightarrow 0$. Για το σκοπό αυτό η είσοδος ελέγχου SE τίθεται στη λογική τιμή 1 ώστε τα φλιπ-φλοπ ολίσθησης να σχηματίσουν ένα μονοπάτι ολίσθησης. Μέσω της σειριακής εισόδου SI ολισθαίνουμε στο μονοπάτι ολίσθησης την ακολουθία 00110011... μήκους $n + 5$, όπου n είναι το πλήθος των φλιπ-φλοπ ολίσθησης.

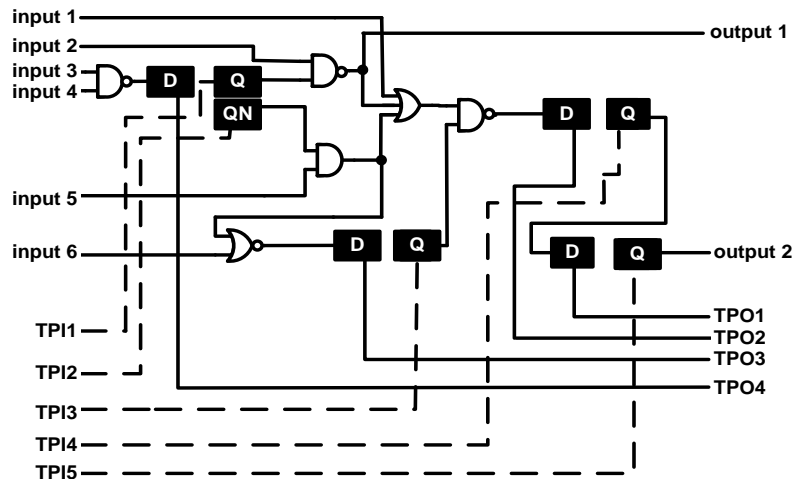
Για τον έλεγχο της ορθής λειτουργίας του συνδυαστικού τμήματος τα επόμενα βήματα θα επαναληφθούν τόσες φορές όσα είναι όλα τα διανύσματα δοκιμής:

- α. Θέσε $SE=1$ και φόρτωσε σειριακά στον καταχωρητή εσωτερικής κατάστασης το τμήμα του διανύσματος δοκιμής που αντιστοιχεί στις ψευδοεισόδους του ακολουθιακού κυκλώματος. Αυτό το βήμα απαιτεί n κύκλους ρολογιού, όπου n είναι το πλήθος των φλιπ-φλοπ ολίσθησης.
- β. Θέτουμε την είσοδο SE στη λογική τιμή 0, ώστε το κύκλωμά μας να γυρίσει σε κανονική λειτουργία και τροφοδοτούμε τις κύριες εισόδους του κυκλώματος με το τμήμα του διανύσματος δοκιμής που αντιστοιχεί σ' αυτές.
- γ. Κατά την επόμενη μετάβαση του σήματος χρονισμού από 0 σε 1, στον καταχωρητή κατάσταση αποθηκεύεται η επόμενη κατάσταση του ακολουθιακού κυκλώματος, η οποία είναι το τμήμα της απόκρισης που αντιστοιχεί στις ψευδοεξόδους του ακολουθιακού κυκλώματος. Κατ' αυτή τη μετάβαση διαβάζουμε τις τιμές των κύριων εξόδων του κυκλώματος.
- δ. Κατά τις επόμενες n χρονικές περιόδους διαβάζουμε σειριακά στην έξοδο SO της το τμήμα της απόκρισης που αντιστοιχεί στις ψευδοεξόδους του ακολουθιακού κυκλώματος.

Υποθέστε ότι θέλουμε να ελέγξουμε την ορθή λειτουργία του ακολουθιακού κυκλώματος του σχήματος 7.6 χρησιμοποιώντας τη μέθοδο του μονοπατιού ολίσθησης. Θα πρέπει κατ' αρχήν να εξάγουμε το σύνολο δοκιμής του συνδυαστικού τμήματος του κυκλώματος. Για να μπορέσουμε να χρησιμοποιήσουμε ένα εργαλείο λογισμικού για την εξαγωγή του συνόλου δοκιμής του συνδυαστικού τμήματος του κυκλώματος, θα πρέπει να έχουμε μια περιγραφή του κυκλώματος χωρίς τα φλιπ-φλοπ (βλέπε σχήμα 7.8). Θα θεωρήσουμε όμως τις εξόδους τους ως ψευδοεισόδους του συνδυαστικού κυκλώματος και τις εισόδους τους ως ψευδοεξόδους του συνδυαστικού κυκλώματος.



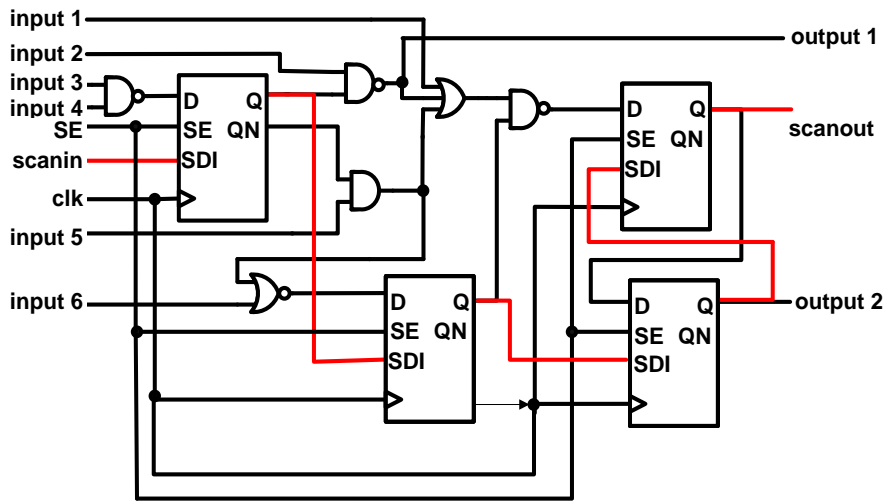
Σχήμα 7.7 Ακολουθιακό κύκλωμα.



Σχήμα 7.8 Συνδυαστικό τμήμα του ακολουθιακού κυκλώματος του σχήματος 7.7 έτοιμο να δοθεί σε ένα εργαλείο εξαγωγής διανυσμάτων δοκιμής για συνδυαστικά κυκλώματα (Scan Effective Circuit).

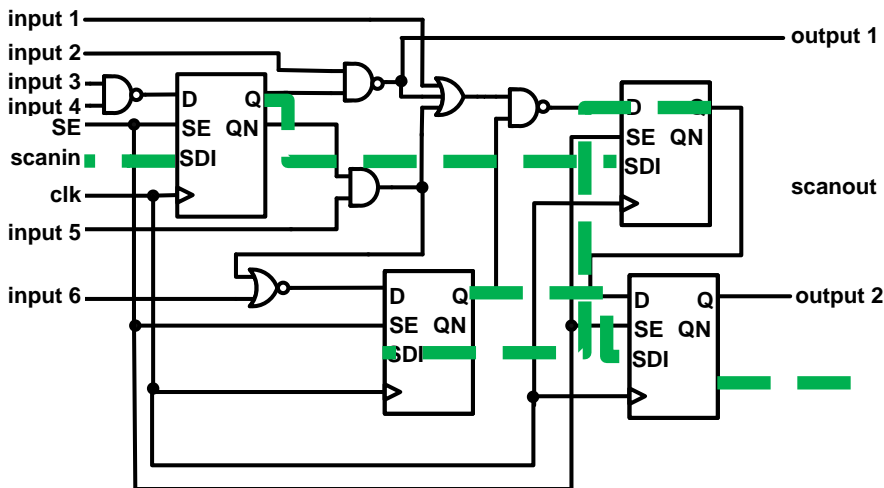
Το κύκλωμα του σχήματος 7.8 είναι έτοιμο για να δοθεί σε ένα εργαλείο εξαγωγής διανυσμάτων δοκιμής για συνδυαστικά κυκλώματα (Scan Effective Circuit). Ωστόσο πρέπει να παρατηρήσουμε ότι οι ψευδοείσοδοι TP11 και TP12 αντιστοιχούν στην κανονική και την συμπληρωματική έξοδο ενός φλιπ-φλοπ. Επομένως θα πρέπει να θέσουμε στο εργαλείο εξαγωγής των διανυσμάτων δοκιμής τον περιορισμό ότι σ' αυτές τις εισόδους μπορεί να δώσει μόνο συμπληρωματικές τιμές.

Για να μπορούμε, μετά την κατασκευή του ολοκληρωμένου κυκλώματος, να εφαρμόσουμε τα διανύσματα δοκιμής που παρήχθησαν από το εργαλείο εξαγωγής διανυσμάτων δοκιμής, θα πρέπει στο κύκλωμα να αντικαταστήσουμε τα φλιπ-φλοπ του συστήματος με φλιπ-φλοπ ολίσθησης. Θα πρέπει επίσης να συνδέσουμε αυτά τα φλιπ-φλοπ ολίσθησης έτσι ώστε να σχηματίσουν ένα μονοπάτι ολίσθησης.

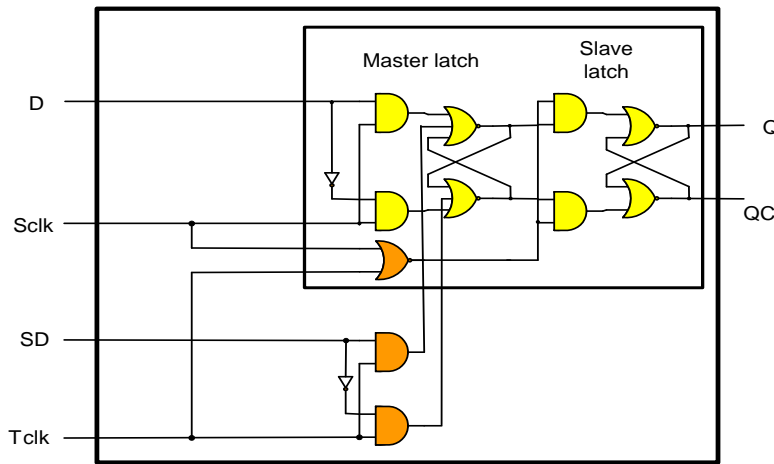


Σχήμα 7.9 Κύκλωμα με μονοπάτι ολίσθησης

Στο κύκλωμα του σχήματος 7.9 τα φλιπ-φλοπ του συστήματος έχουν αντικατασταθεί από φλιπ-φλοπ ολίσθησης τα οποία έχουν συνδεθεί μεταξύ τους ώστε να σχηματίσουν ένα μονοπάτι ολίσθησης. Η σειρά με την οποία συνδέονται τα φλιπ-φλοπ ολίσθησης για να σχηματίσουν το μονοπάτι ολίσθησης δεν έχει σημασία για τη λειτουργία του μονοπατιού ολίσθησης. Ωστόσο, επηρεάζει το μήκος των γραμμών διασύνδεσης και επομένως το συνολικό εμβαδόν του κυκλώματος. Στο κύκλωμα του σχήματος 7.10 τα φλιπ-φλοπ ολίσθησης έχουν συνδεθεί με διαφορετική σειρά. Παρατηρούμε ότι το μήκος των γραμμών διασύνδεσης τώρα είναι μεγαλύτερο.

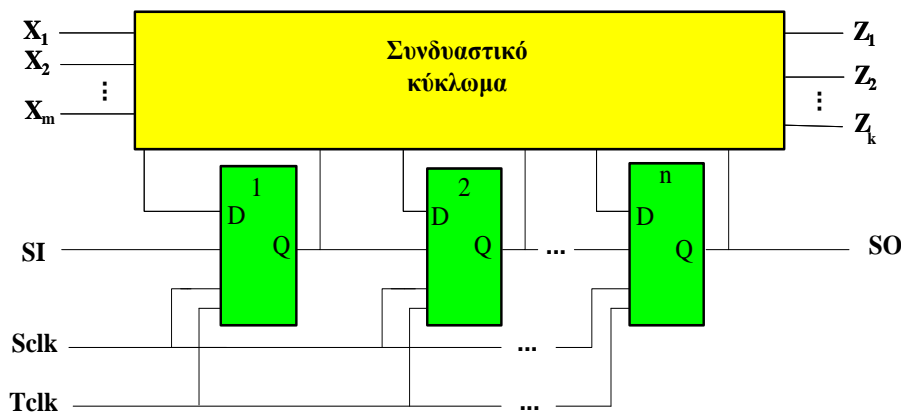


Σχήμα 7.10 Κύκλωμα με μεγάλου μήκους μονοπάτι ολίσθησης



Σχήμα 7.11 Φλιπ-φλοπ ολίσθησης με δύο πόρτες εισόδου (Two Port Scan flip-flop)

Στο σχήμα 7.11 δίνεται σε επίπεδο λογικών πυλών η δομή ενός φλιπ-φλοπ ολίσθησης με δύο πόρτες. Κατά τη μετάβαση του σήματος χρονισμού Sclk από 0 σε 1 στο φλιπ-φλοπ αποθηκεύεται η τιμή που οδηγεί την είσοδο D. Κατά τη μετάβαση του σήματος χρονισμού Tclk από 0 σε 1 στο φλιπ-φλοπ αποθηκεύεται η τιμή που οδηγεί την είσοδο SD. Σε σύγκριση με το φλιπ-φλοπ ολίσθησης με πολυπλέκτη, το φλιπ-φλοπ ολίσθησης με δύο πόρτες έχει περίπου το ίδιο κόστος υλοποίησης. Ωστόσο το φλιπ-φλοπ ολίσθησης με δύο πόρτες βάζει πολύ μικρότερη καθυστέρηση. Η καθυστέρηση αυτή οφείλεται κυρίως στο γεγονός ότι δύο πύλες NOR στο flip flop του συστήματος με δύο εισόδους έχουν αντικατασταθεί από αντίστοιχες πύλες των τριών εισόδων.

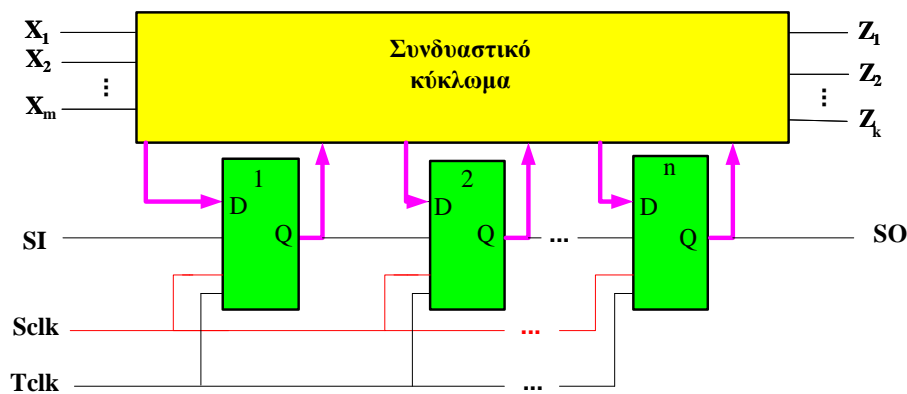


Σχήμα 7.12 Υλοποίηση μονοπατιού ολίσθησης χρησιμοποιώντας φλιπ-φλοπ ολίσθησης με δύο πόρτες.

Στο σχήμα 7.12 τα φλιπ-φλοπ του συστήματος έχουν αντικατασταθεί από φλιπ-φλοπ ολίσθησης με δύο πόρτες. Παρατηρούμε ότι τα φλιπ-φλοπ ολίσθησης έχουν συνδεθεί μεταξύ τους ώστε η έξοδος καθενός φλιπ-φλοπ να οδηγεί μια είσοδο του πολυπλέκτη ενός άλλου φλιπ-φλοπ ολίσθησης, εκτός βέβαια από ένα φλιπ-φλοπ ολίσθησης του οποίου η έξοδος αποτελεί κύρια έξοδο του κυκλώματος, SO. Επίσης ενός φλιπ-φλοπ ολίσθησης η μία είσοδος του πολυπλέκτη αποτελεί κύρια είσοδο του

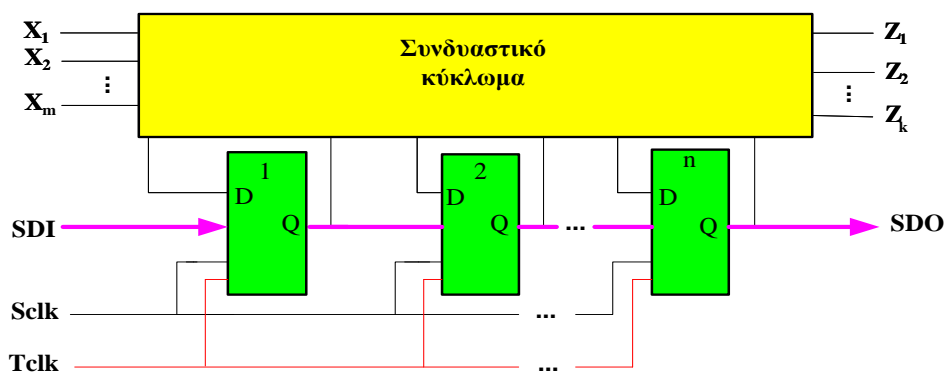
κυκλώματος, SI. Παρατηρούμε επίσης ότι, επιπλέον της κύριας εισόδου SI και της κύριας εξόδου SO που έχουν προστεθεί, έχει προστεθεί και η κύρια είσοδος Tckl.

Κατά την κανονική λειτουργία του κυκλώματος χρησιμοποιείται η είσοδος Selk, η οποία τροφοδοτείται με το σήμα χρονισμού του συστήματος ενώ η είσοδος Tckl οδηγείται με τη σταθερή λογική τιμή 0. Όταν ο καταχωρητής εσωτερικής κατάστασης χρησιμοποιείται ως ολισθητής τότε χρησιμοποιείται η είσοδος Tckl, η οποία τροφοδοτείται με το σήμα χρονισμού ελέγχου ορθής λειτουργίας, ενώ η είσοδος Selk οδηγείται με τη λογική τιμή 0.



Σχήμα 7.13 Κανονικός τρόπος λειτουργίας.

Όταν η είσοδος Tckl οδηγείται με τη σταθερή λογική τιμή 0 και η είσοδος Selk τροφοδοτείται με το σήμα χρονισμού του συστήματος, τότε έχουμε κανονική λειτουργία του κυκλώματος. Στο σχήμα 7.13 με ροζ χρώμα δηλώνεται η διακίνηση των δεδομένων και με κόκκινο χρώμα το σήμα χρονισμού του συστήματος.



Σχήμα 7.14 Λειτουργία ολίσθησης.

Όταν η είσοδος Selk οδηγείται με τη σταθερή λογική τιμή 0 και η είσοδος Tckl τροφοδοτείται με το σήμα χρονισμού του ελέγχου ορθής λειτουργίας, τότε τα φλιπ-φλοπ ολίσθησης σχηματίζουν ένα μονοπάτι ολίσθησης με κύρια είσοδο τον ακροδέκτη SI και κύρια έξοδο τον ακροδέκτη SO, όπως φαίνεται στο σχήμα 7.14. Κατά τη μετάβαση του σήματος χρονισμού Tckl από 0 σε 1 γίνεται ολίσθηση, κατά ένα δυαδικό ψηφίο, από την κύρια είσοδο SI προς την κύρια έξοδο SO. Στο σχήμα με

ροζ χρώμα δηλώνεται η διακίνηση των δεδομένων και με κόκκινο χρώμα το σήμα χρονισμού του ελέγχου ορθής λειτουργίας.

Όσον αφορά τον έλεγχο της ορθής λειτουργίας των φλιπ-φλοπ του ακολουθιακού κυκλώματος, αυτός γίνεται με τον ίδιο τρόπο που περιγράψαμε όταν για την υλοποίηση του μονοπατιών ολίσθησης χρησιμοποιούνταν φλιπ-φλοπ με πολυπλέκτη.

Όσον αφορά τον έλεγχο της ορθής λειτουργίας του συνδυαστικού τμήματος οι διαφορές είναι πολύ μικρές.

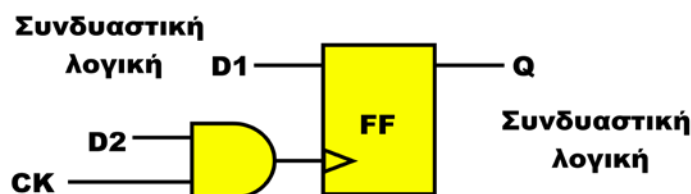
Τα επόμενα βήματα θα επαναληφθούν τόσες φορές όσα είναι όλα τα διανύσματα δοκιμής:

- α. Θέσε $Sckl=0$ και χρησιμοποιώντας το σήμα χρονισμού $Tckl$ φόρτωσε σειριακά στον καταχωρητή εσωτερικής κατάστασης το τμήμα του διανύσματος δοκιμής που αντιστοιχεί στις ψευδοεισόδους του ακολουθιακού κυκλώματος. Αυτό το βήμα απαιτεί n κύκλους ρολογιού, όπου n είναι το πλήθος των φλιπ-φλοπ ολίσθησης.
- β. Θέσε $Tckl=0$ και οδήγησε τις κύριες εισόδους του ακολουθιακού κυκλώματος με το τμήμα του διανύσματος δοκιμής που αντιστοιχεί σε αυτές.
- γ. Οδήγησε την είσοδο $Sckl$ με μια περίοδο του σήματος χρονισμού του συστήματος. Κατά τη μετάβαση του σήματος $Sckl$ από 0 σε 1 στον καταχωρητή κατάστασης αποθηκεύεται η επόμενη κατάσταση του ακολουθιακού κυκλώματος, η οποία είναι το τμήμα της απόκρισης που αντιστοιχεί στις ψευδοεξόδους του ακολουθιακού κυκλώματος. Σ' αυτή τη μετάβαση διαβάζουμε τις τιμές των κύριων εξόδων του κυκλώματος.
- δ. Θέτουμε $Sckl=0$ και χρησιμοποιώντας το σήμα χρονισμού $Tckl$ κατά τις επόμενες n χρονικές περιόδους διαβάζουμε σειριακά στην έξοδο SO το τμήμα της απόκρισης που αντιστοιχεί στις ψευδοεξόδους του ακολουθιακού κυκλώματος.

Για να μπορούμε, χρησιμοποιώντας ένα εργαλείο λογισμικού, να αντικαταστήσουμε τα φλιπ-φλοπ του συστήματος με φλιπ-φλοπ ολίσθησης, θα πρέπει να ακολουθούμε κάποιους σχεδιαστικούς κανόνες:

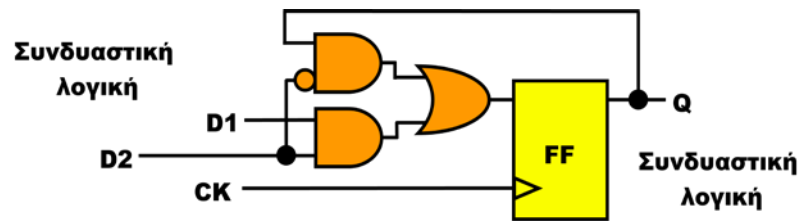
- Θα πρέπει να χρησιμοποιούνται μόνον φλιπ-φλοπ αφέντη-δούλου (master-slave flip-flops) τύπου D.
- Τουλάχιστον μία κύρια είσοδος του ολοκληρωμένου κυκλώματος πρέπει να είναι διαθέσιμη για τον έλεγχο της ορθής λειτουργίας του.
- Όλες οι εισόδους χρονισμού των φλιπ-φλοπ πρέπει να είναι ελέγξιμες από κύριες εισόδους του ολοκληρωμένου κυκλώματος.
- Σήματα χρονισμού δεν πρέπει να οδηγούν εισόδους δεδομένων των φλιπ-φλοπ.

Θεωρήσετε ότι το φλιπ-φλοπ του σχήματος 7.15 είναι ένα φλιπ-φλοπ του συστήματος. Γίνεται παραβίαση του σχεδιαστικού κανόνα: «Όλες οι εισόδους χρονισμού των φλιπ-φλοπ πρέπει να είναι ελέγξιμες από κύριες εισόδους του ολοκληρωμένου κυκλώματος.» Η επίλυση του προβλήματος δίνεται στο σχήμα 7.16



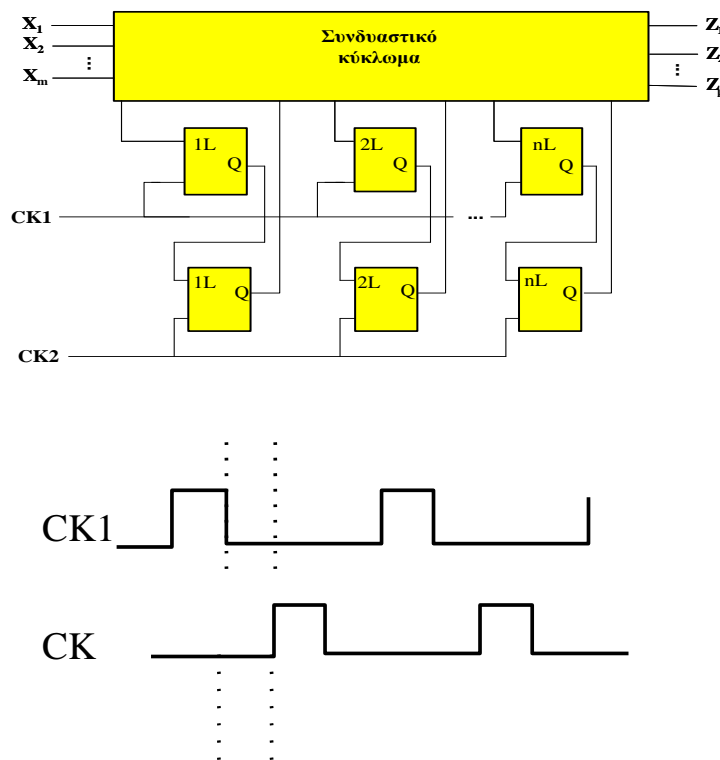
Σχήμα 7.15 Παραβίαση κανόνα.

όπου η λογική πύλη AND αντικαταστάθηκε από έναν πολυπλέκτη. Η λειτουργία είναι ταυτοτική. Όταν η είσοδος D2 έχει τη λογική τιμή 0, τότε το φλιπ-φλοπ δεν αλλάζει κατάσταση. Όταν η είσοδος D2 έχει τη λογική τιμή 1, τότε στο φλιπ-φλοπ αποθηκεύεται η τιμή της εισόδου D1.



Σχήμα 7.16 Επίλυση του προβλήματος.

Στη συνέχεια θα ασχοληθούμε με ακολουθιακές μηχανές που υλοποιούνται με τη χρήση μανδαλωτών. Οι μηχανές αυτές διακρίνονται σε δύο μεγάλες κατηγορίες. Αυτές που χρησιμοποιούν δύο σήματα χρονισμού με διαφορά φάσης και αυτές που χρησιμοποιούν ένα σήμα χρονισμού. Οι μηχανές που χρησιμοποιούν δύο σήματα χρονισμού με διαφορά φάσης διακρίνονται περαιτέρω σε αυτές που χρησιμοποιούν διπλό μανδαλωτή και σε αυτές που χρησιμοποιούν απλό μανδαλωτή.

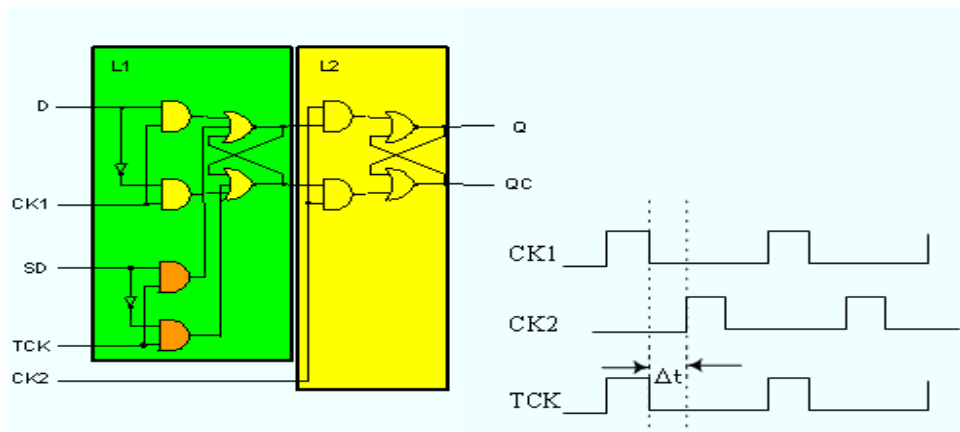


Σχήμα 7.17 Μηχανές που χρησιμοποιούν δύο σήματα χρονισμού με διαφορά φάσης και διπλό μανδαλωτή.

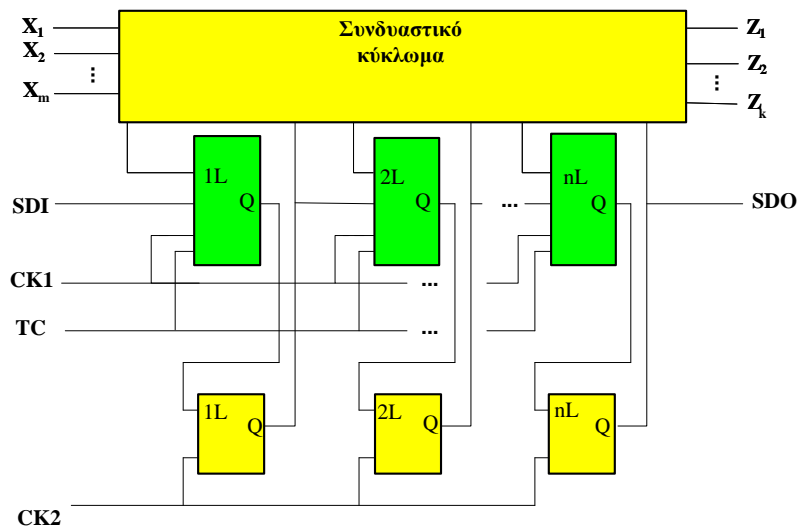
Στο σχήμα 7.17 βλέπουμε μια μηχανή που χρησιμοποιεί δύο σήματα χρονισμού με διαφορά φάσης. Επίσης βλέπουμε τα σήματα χρονισμού CKL1 και CKL2. Λόγω της διαφάνειας των μανδαλωτών δεν πρέπει να υπάρχει χρονική περίοδος για την οποία και τα δύο σήματα χρονισμού να έχουν τη λογική τιμή 1.

Επίσης λόγω της διαφάνειάς τους, οι μανδαλωτές δεν μπορούν να χρησιμοποιηθούν για τη σχεδίαση ολισθητή, παρά μόνον εάν λειτουργούν ανά ζεύγος υπό μορφή master-slave. Επομένως οι δύο μανδαλωτές 1L που οδηγούνται με τα σήματα χρονισμού CK1 και CK2 αντίστοιχα θα αποτελέσουν τη μια βαθμίδα του ολισθητή, οι δύο μανδαλωτές 2L που οδηγούνται με τα σήματα χρονισμού CK1 και CK2 αντίστοιχα θα αποτελέσουν την επόμενη βαθμίδα του ολισθητή και ούτω καθεξής. Κάθε βαθμίδα πρέπει να έχει δύο εισόδους, επομένως θα αντικαταστήσουμε τον μανδαλωτή κάθε βαθμίδας που οδηγείται με το σήμα χρονισμού CK1 με ένα μανδαλωτή δύο εισόδων όπως φαίνεται στο σχήμα 7.18.

Στο σχήμα 7.18 δίνεται η δομή μιας βαθμίδας του ολισθητή. Ο μανδαλωτής L1 έχει δύο πόρτες εισόδου. Η είσοδος D χρησιμοποιείται σε συνδυασμό με το σήμα χρονισμού CK1 του συστήματος, όταν το σύστημα είναι σε κανονική λειτουργία. Η είσοδος SD χρησιμοποιείται σε συνδυασμό με το σήμα χρονισμού TCK όταν το σύστημα είναι σε κατάσταση ολίσθησης. Την είσοδο SD θα την καλούμε σειριακή είσοδο του μανδαλωτή με δύο πόρτες εισόδου. Στο σχήμα επίσης μπορούμε να δούμε τη σχέση των φάσεων των τριών σημάτων χρονισμού CK1, CK2 και TCK



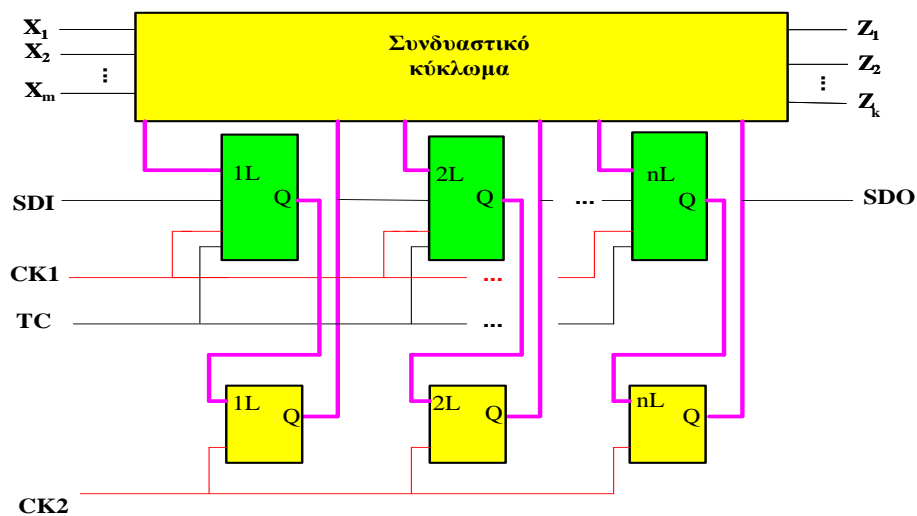
Σχήμα 7.18 Μανδαλωτής με δύο πόρτες εισόδου.



Σχήμα 7.19 Υλοποίηση μονοπατιού ολίσθησης χρησιμοποιώντας μανδαλωτές με δύο πόρτες εισόδου.

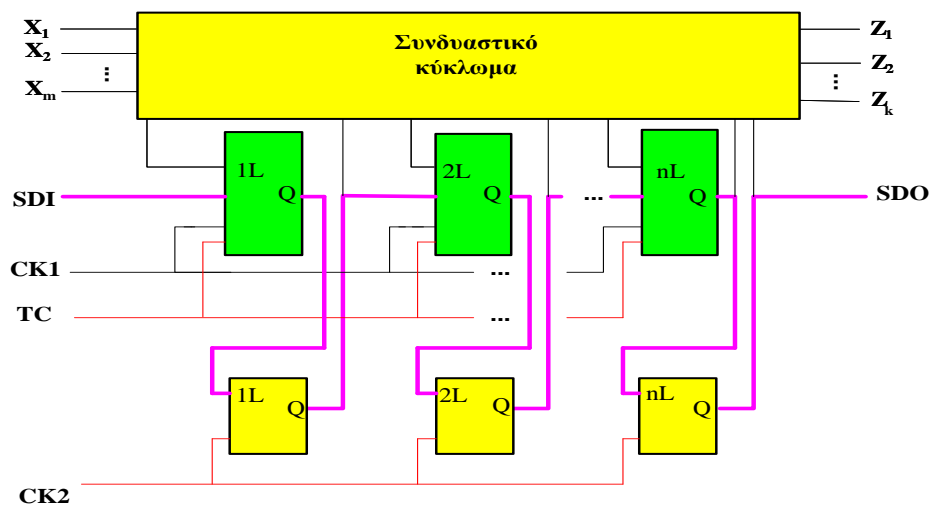
Στο σχήμα 7.19 οι μανδαλωτές του συστήματος οι οποίοι οδηγούνται από το σήμα χρονισμού CK1 έχουν αντικατασταθεί από μανδαλωτές με δύο πόρτες εισόδου. Παρατηρούμε ότι καθένας από τους μανδαλωτές που οδηγούνται με το σήμα χρονισμού CK2 οδηγεί τόσο μια είσοδο του συνδυαστικού κυκλώματος όσο και τη σειριακή είσοδο ενός μανδαλωτή με δύο πόρτες εισόδου, ώστε να σχηματιστεί ένα μονοπάτι ολίσθησης.

Επίσης η σειριακή είσοδος ενός μανδαλωτή με δύο πόρτες εισόδου αποτελεί κύρια είσοδο, SDI, του ολοκληρωμένου κυκλώματος, ενώ η έξοδος ενός μανδαλωτή που οδηγείται από το σήμα χρονισμού CK2 αποτελεί κύρια έξοδο, SDO, του ολοκληρωμένου κυκλώματος ολίσθησης. Η μία είσοδος του πολυπλέκτη αποτελεί κύρια είσοδο του κυκλώματος, SI.



Σχήμα 7.20 Κανονική λειτουργία.

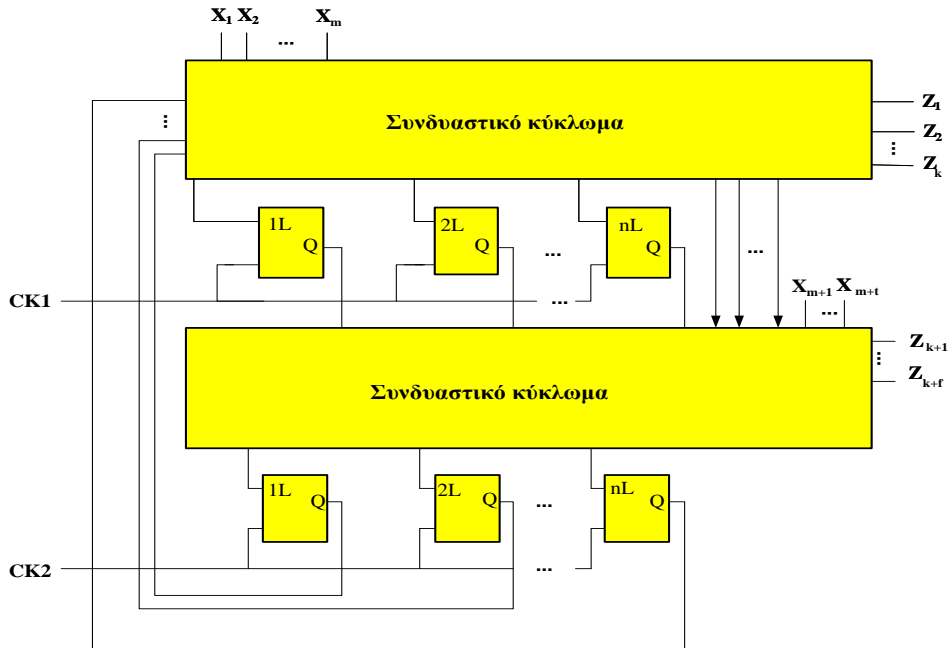
Κατά την κανονική λειτουργία του συστήματος, σχήμα 7.20, χρησιμοποιούνται τα σήματα χρονισμού CK1 και CK2. Στο σχήμα οι γραμμές ροζ χρώματος δηλώνουν τη ροή της πληροφορίας μέσω των μανδαλωτών κατά την κανονική λειτουργία του συστήματος.



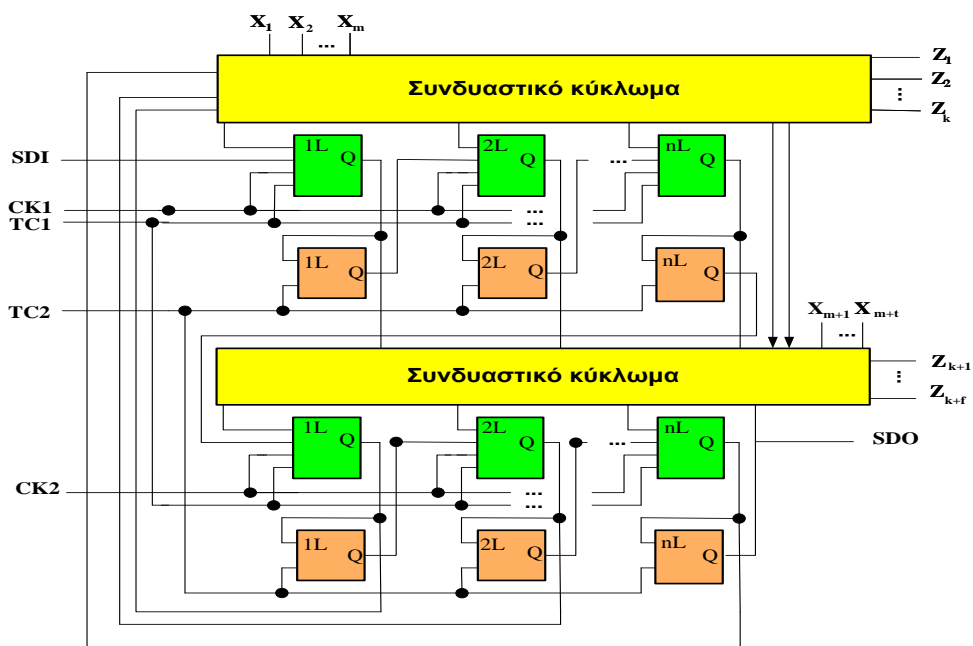
Σχήμα 7.21 Λειτουργία ολίσθησης.

Σε λειτουργία ολίσθησης, σχήμα 7.21, χρησιμοποιούνται τα σήματα χρονισμού TK και CK2. Οι γραμμές ροζ χρώματος δηλώνουν το μονοπάτι ολίσθησης.

Στο σχήμα 7.22 βλέπουμε μια μηχανή που χρησιμοποιεί δύο σήματα χρονισμού με διαφορά φάσης και απλό μανδαλωτή. Λόγω της διαφάνειάς τους, κ μανδαλωτές δεν μπορούν να χρησιμοποιηθούν για τη σχεδίαση ολισθητή των κ βαθμίδων. Υπάρχουν δύο μέθοδοι για την υλοποίηση της τεχνικής των μονοπατιών ολίσθησης. Στη συνέχεια θα δούμε καθεμία από αυτές.

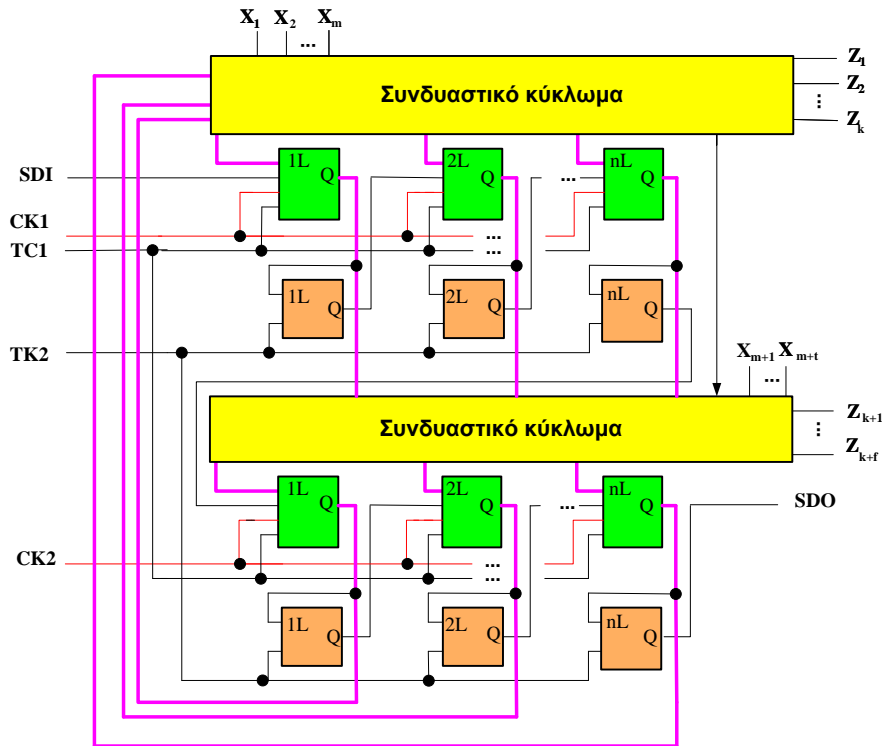


Σχήμα 7.22 Μηχανές που χρησιμοποιούν δύο σήματα χρονισμού με διαφορά φάσης και απλό μανδαλωτή.



Σχήμα 7.23 Υλοποίηση μονοπατιού ολίσθησης σε μηχανές που χρησιμοποιούν δύο σήματα χρονισμού με διαφορά φάσης και απλό μανδαλωτή- Μέθοδος 1.

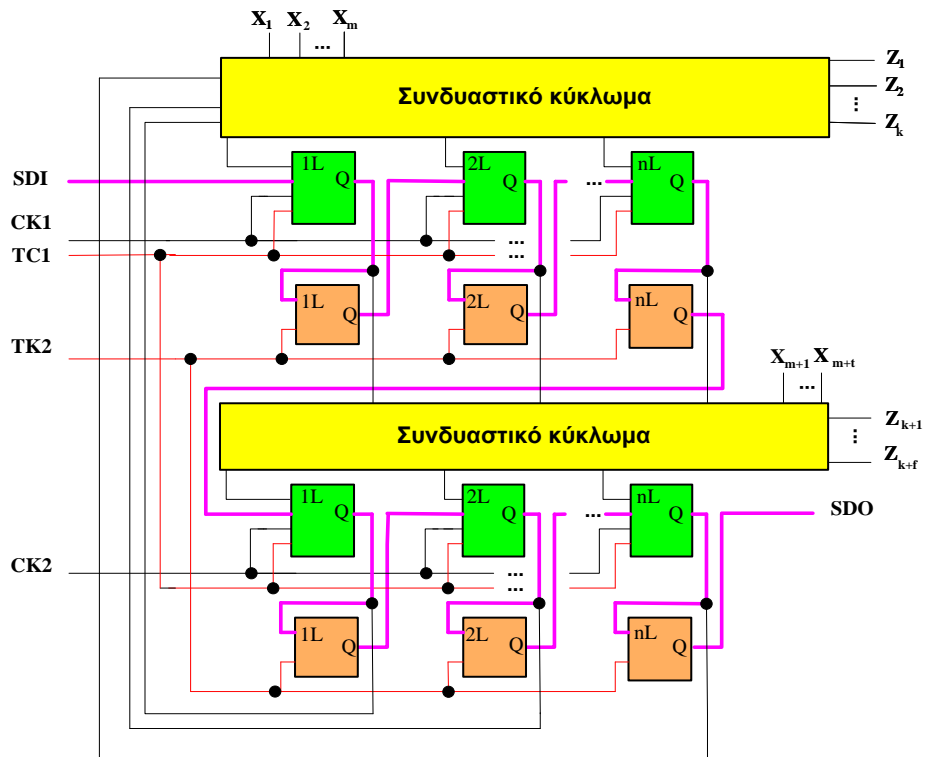
Σύμφωνα με την πρώτη μέθοδο κάθε ένας από τους μανδαλωτές αντικαθίσταται από ένα ζεύγος μανδαλωτών εκ των οποίων ο πρώτος μανδαλωτής έχει δύο πόρτες εισόδου.



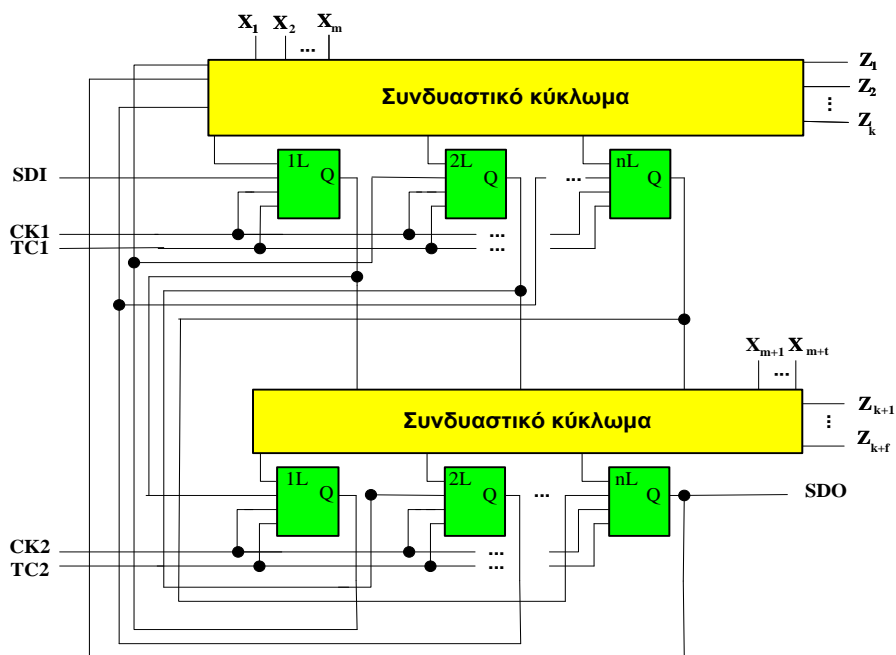
Σχήμα 7.24 Κανονική λειτουργία.

Το σχήμα 7.24 περιγράφει την κανονική λειτουργία στην περίπτωση υλοποίησης της τεχνικής του μονοπατιού ολίσθησης σε μηχανές που χρησιμοποιούν δύο σήματα χρονισμού με διαφορά φάσης και απλό μανδαλωτή με την πρώτη μέθοδο. Κατά την κανονική λειτουργία του συστήματος χρησιμοποιούνται τα σήματα χρονισμού CK1 και CK2. Στο σχήμα οι γραμμές ροζ χρώματος δηλώνουν τη ροή της πληροφορίας μέσω των μανδαλωτών κατά την κανονική λειτουργία του συστήματος.

Το σχήμα 7.25 περιγράφει τη λειτουργία ολίσθησης στην περίπτωση υλοποίησης της τεχνικής του μονοπατιού ολίσθησης σε μηχανές που χρησιμοποιούν δύο σήματα χρονισμού με διαφορά φάσης και απλό μανδαλωτή με την πρώτη μέθοδο. Σε λειτουργία ολίσθησης χρησιμοποιούνται τα σήματα χρονισμού TK και CK2. Στο σχήμα η γραμμή ροζ χρώματος δηλώνει το μονοπάτι ολίσθησης.



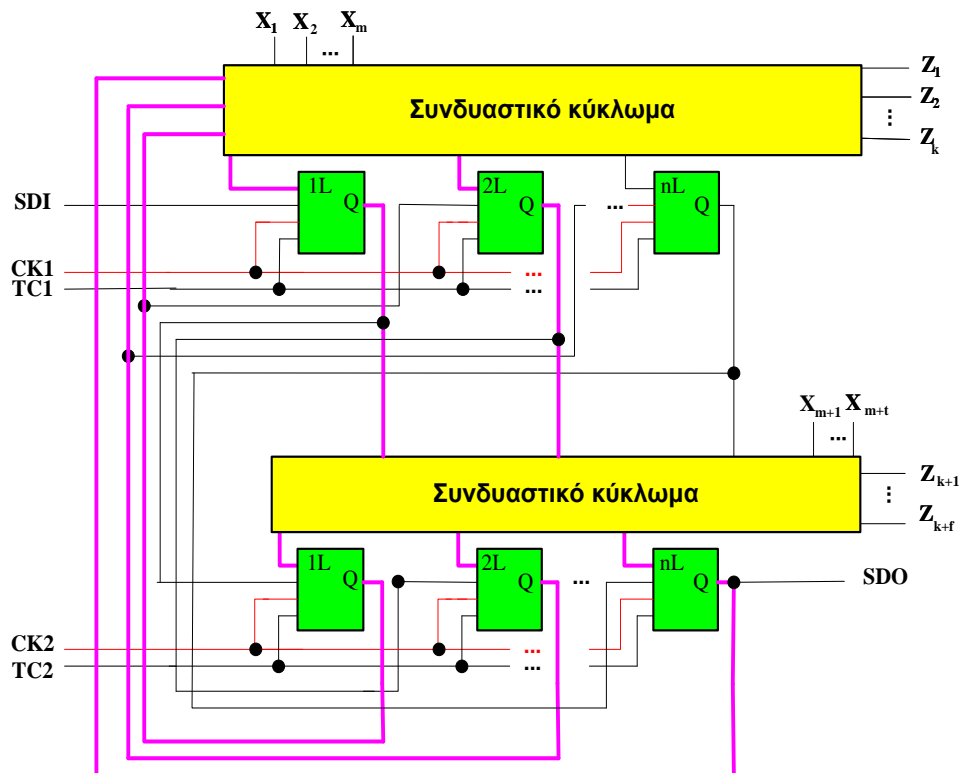
Σχήμα 7.25 Λειτουργία ολίσθησης.



Σχήμα 7.26 Υλοποίηση μονοπατιού ολίσθησης σε μηχανές που χρησιμοποιούν δύο σήματα χρονισμού με διαφορά φάσης και απλό μανδαλωτή- Μέθοδος 2.

Σύμφωνα με τη δεύτερη μέθοδο, σχήμα 7.26, κάθε ένας από τους μανδαλωτές αντικαθίσταται από ένα μανδαλωτή που έχει δύο πόρτες εισόδου. Μια βαθμίδα του

ολίσθητη υλοποιείται από ένα μανδαλωτή της πρώτης σειράς και ένα μανδαλωτή της δεύτερης σειράς.

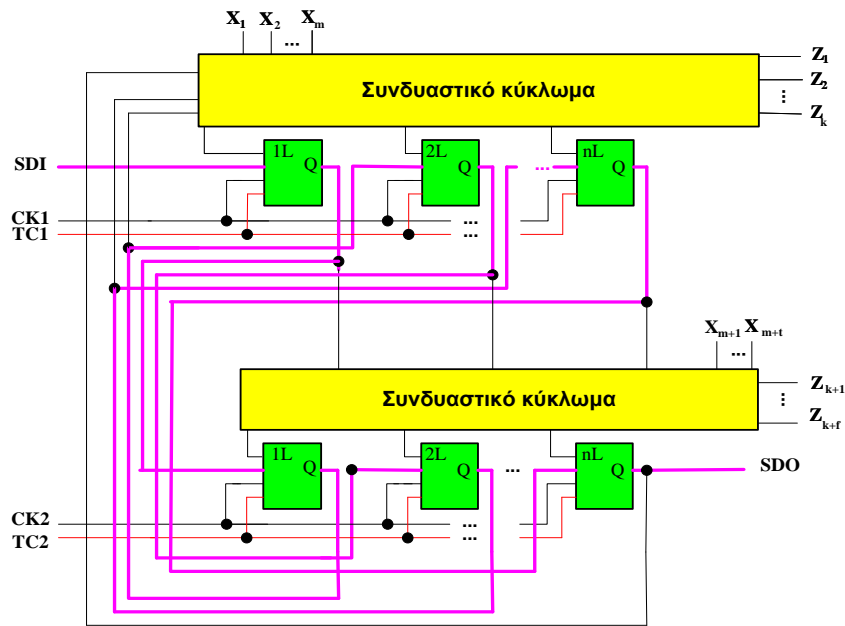


Σχήμα 7.27 Κανονική λειτουργία.

Το σχήμα 7.27 περιγράφει την κανονική λειτουργία στην περίπτωση υλοποίησης της τεχνικής του μονοπατιού ολίσθησης σε μηχανές που χρησιμοποιούν δύο σήματα χρονισμού με διαφορά φάσης και απλό μανδαλωτή με τη δεύτερη μέθοδο. Κατά την κανονική λειτουργία του συστήματος χρησιμοποιούνται τα σήματα χρονισμού CK1 και CK2. Στο σχήμα οι γραμμές ροζ χρώματος δηλώνουν τη ροή της πληροφορίας μέσω των μανδαλωτών κατά την κανονική λειτουργία του συστήματος.

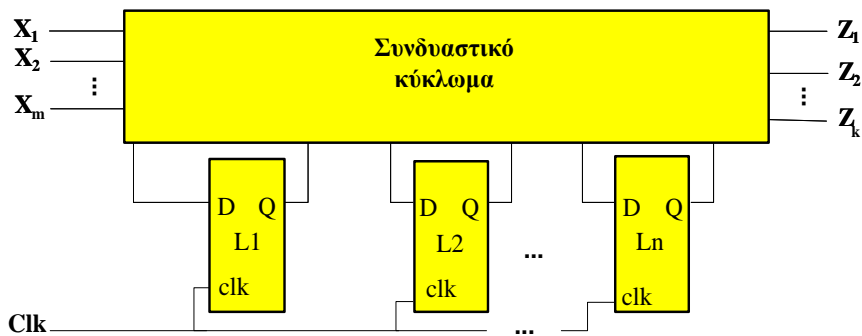
Το σχήμα 7.28 περιγράφει τη λειτουργία ολίσθησης στην περίπτωση υλοποίησης της τεχνικής του μονοπατιού ολίσθησης σε μηχανές που χρησιμοποιούν δύο σήματα χρονισμού με διαφορά φάσης και απλό μανδαλωτή με τη δεύτερη μέθοδο. Σε λειτουργία ολίσθησης χρησιμοποιούνται τα σήματα χρονισμού TK1 και TK2. Στο σχήμα η γραμμή ροζ χρώματος δηλώνει το μονοπάτι ολίσθησης.

Συγκρίνοντας την πρώτη μέθοδο με τη δεύτερη, συμπεραίνουμε ότι στην πρώτη μέθοδο αυξάνεται σημαντικά το πλήθος των πυλών που απαιτούνται για την υλοποίησή της. Στη δεύτερη μέθοδο η αύξηση του πλήθους των πυλών που απαιτούνται για την υλοποίησή της είναι μικρότερη, αλλά έχουμε σημαντική αύξηση του συνολικού μήκους των γραμμών διασύνδεσης.



Σχήμα 7.28 Λειτουργία ολίσθησης.

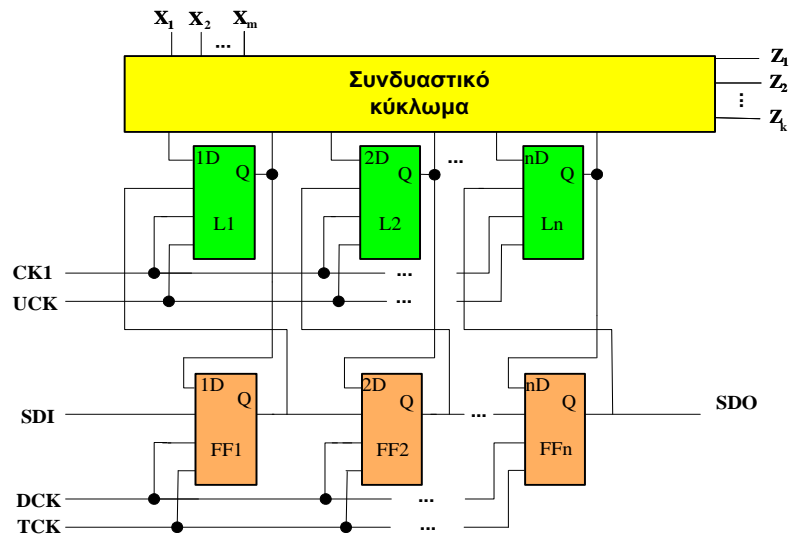
Στο σχήμα 7.29 δίνεται η δομή μιας μηχανής που χρησιμοποιεί μανδαλωτές και ένα σήμα χρονισμού. Σ' αυτές τις μηχανές για την υλοποίηση του μονοπατιού ολίσθησης έχει υιοθετηθεί μια διαφορετική τεχνική η οποία καλείται τεχνική ολίσθησης – ενημέρωσης (scan-set).



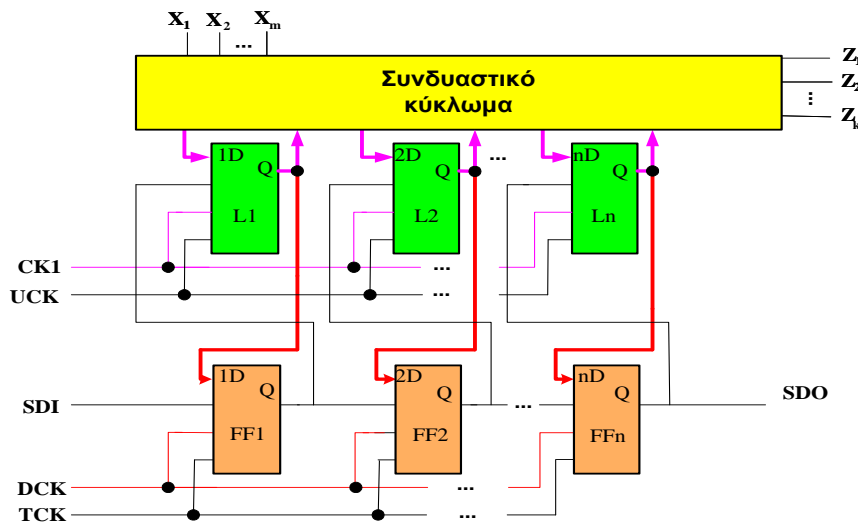
Σχήμα 7.29 Μηχανές που χρησιμοποιούν μανδαλωτές και ένα σήμα χρονισμού (One-Phase Latch Machines).

Οι μανδαλωτές του συστήματος αντικαθίστανται από μανδαλωτές με δύο πόρτες εισόδου. Επίσης για κάθε μανδαλωτή του συστήματος προστίθεται ένα φλιπ-φλοπ με δύο πόρτες εισόδου. Τα φλιπ-φλοπ συνδέονται με τέτοιο τρόπο ώστε να σχηματίσουν ένα καταχωρητή με δύο τρόπους λειτουργίας, παράλληλης φόρτωσης και ολίσθησης.

Στο σχήμα 7.30 δίνεται η δομή της μηχανής μετά την αντικατάσταση των μανδαλωτών του συστήματος και την πρόσθεση των φλιπ-φλοπ. Παρατηρούμε ότι η έξοδος καθενός μανδαλωτή οδηγεί αφενός μια είσοδο του συνδυαστικού κυκλώματος, αφετέρου μια από τις εισόδους του αντίστοιχου φλιπ-φλοπ. Επίσης η έξοδος ενός φλιπ-φλοπ οδηγεί μια είσοδο του αντίστοιχου μανδαλωτή.



Σχήμα 7.30 Υλοποίηση μονοπατιού ολίσθησης σε μηχανές που χρησιμοποιούν μανδαλωτές και ένα σήμα χρονισμού (Scan-Set Technique).



Σχήμα 7.31 Κανονική λειτουργία και λήψη στιγμιότυπου (Normal Operation & Snapshot)

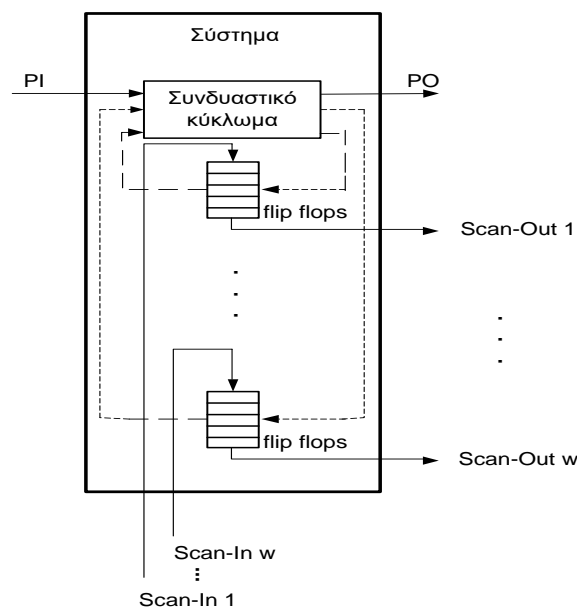
Κατά την κανονική λειτουργία του συστήματος χρησιμοποιείται το σήμα χρονισμού CK1. Μπορούμε να αποθηκεύσουμε στα φλιπ-φλοπ ένα στιγμιότυπο της κανονικής λειτουργίας του συστήματος (βλέπε σχήμα 7.31). Στέλνοντας ένα παλμό μέσω της γραμμής DCK ότι είναι αποθηκευμένο στους μανδαλωτές αποθηκεύεται παράλληλα στα αντίστοιχα φλιπ-φλοπ. Στη συνέχεια, ενώ η μηχανή λειτουργεί κανονικά, μπορούμε σειριακά να διαβάσουμε το περιεχόμενο των φλιπ-φλοπ χρησιμοποιώντας το σήμα χρονισμού TCK. Η λειτουργία αυτή φαίνεται στο σχήμα 7.32.

Πλεονεκτήματα και μειονεκτήματα της χρησιμοποίησης του μονοπατιού ολίσθησης

Το μεγάλο πλεονέκτημα της τεχνικής αυτής είναι η απλοποίηση της διαδικασίας εξαγωγής των διανυσμάτων δοκιμής ενός ακολουθιακού κυκλώματος. Η εξαγωγή των διανυσμάτων δοκιμής γίνεται χρησιμοποιώντας εργαλεία εξαγωγής διανυσμάτων δοκιμής για συνδυαστικά κυκλώματα.

Το άλλο μεγάλο πλεονέκτημα είναι η απλοποίηση της εφαρμογής των διανυσμάτων δοκιμής στο υπό έλεγχο κύκλωμα. Ωστόσο, επειδή τα διανύσματα δοκιμής εισάγονται στο μονοπάτι ολίσθησης σειριακά και το μονοπάτι ολίσθησης μπορεί να έχει πολύ μεγάλο μήκος, ο χρόνος που απαιτείται για τον έλεγχο της ορθής λειτουργίας είναι πολύ μεγάλος.

Το πρόβλημα του πολύ μεγάλου χρόνου ελέγχου της ορθής λειτουργίας του κυκλώματος μπορεί να μετριαστεί εάν αντί ενός μονοπατιού ολίσθησης χρησιμοποιήσουμε πολλά παράλληλα μονοπάτια ολίσθησης. Τότε κάθε μονοπάτι ολίσθησης συνδέεται με μια κύρια είσοδο του κυκλώματος και με μια κύρια έξοδο του κυκλώματος, όπως φαίνεται στο σχήμα 7.34.



Σχήμα 7.34 Παράλληλα μονοπάτια ολίσθησης (Parallel Scan paths).

Στο σχήμα 7.34 φαίνεται η τεχνική των παράλληλων μονοπατιών ολίσθησης. Αντί να έχουμε ένα μονοπάτι ολίσθησης μήκους k , έχουμε w μονοπάτια ολίσθησης κάθε ένα των οποίων έχει μήκος k/w . Επομένως ενώ στην περίπτωση του ενός μονοπατιού ολίσθησης απαιτούνταν k χρονικές περιόδους για να φορτώσουμε το μονοπάτι σειριακά, στην περίπτωση που έχουμε w παράλληλα μονοπάτια απαιτούνται μόνο k/w χρονικές περιόδους.

Επιπλέον της μείωσης του χρόνου που απαιτείται για τον έλεγχο της ορθής λειτουργίας του κυκλώματος, η τεχνική των παράλληλων μονοπατιών μπορεί να χρησιμοποιηθεί σε κυκλώματα που χρησιμοποιούνται περισσότερο από ένα σήματα χρονισμού κατά την κανονική λειτουργία του κυκλώματος.

Το κόστος υλοποίησης των μονοπατιών ολίσθησης οφείλεται αφενός στην αύξηση του πλήθους των ακροδεκτών του ολοκληρωμένου κυκλώματος και αφετέρου στην αύξηση του απαιτούμενου εμβαδού.

Για την υλοποίηση της τεχνικής των μονοπατιών ολίσθησης απαιτείται το ολοκληρωμένο κύκλωμα να έχει τουλάχιστον ένα επιπλέον ακροδέκτη εισόδου, ο οποίος χρησιμοποιείται για να τίθεται το κύκλωμα σε κανονική λειτουργία ή σε κατάσταση ολίσθησης. Η σειριακή είσοδος και η σειριακή έξοδος κάθε μονοπατιού ολίσθησης μπορεί να πολυπλεχτεί με κάποια από τις κύριες εισόδους και εξόδους αντίστοιχα του ολοκληρωμένου κυκλώματος. Ωστόσο αυτό δημιουργεί κάποια προβλήματα. Συνήθως υπάρχουν πρόσθετοι ακροδέκτες εισόδου και εξόδου που χρησιμοποιούνται ως εισόδοι και εξόδοι αντίστοιχα των μονοπατιών ολίσθησης.

Η αύξηση του απαιτούμενου εμβαδού οφείλεται σε δύο αιτίες. Η μία είναι η αύξηση των πυλών που απαιτούνται για την τροποποίηση των φλιπ-φλοπ του συστήματος σε φλιπ-φλοπ ολίσθησης. Η άλλη είναι η αύξηση του πλήθους των γραμμών διασύνδεσης. Για τη μείωση του συνολικού μήκους των γραμμών διασύνδεσης ο παράγοντας αυτός λαμβάνεται υπ' όψιν αφενός κατά την τοποθέτηση των φλιπ-φλοπ (layout), αφετέρου επιλέγεται κατάλληλα η σειρά με την οποία τα φλιπ-φλοπ συμμετέχουν στο μονοπάτι ολίσθησης.

Λόγω της αντικατάστασης των φλιπ-φλοπ του συστήματος με φλιπ-φλοπ ολίσθησης και της αύξησης του εμβαδού που απαιτείται για την υλοποίηση του κυκλώματος, έχουμε μείωση της απόδοσής του. Η συχνότητα χρονισμού μπορεί να μειωθεί από 5% μέχρι 10%.

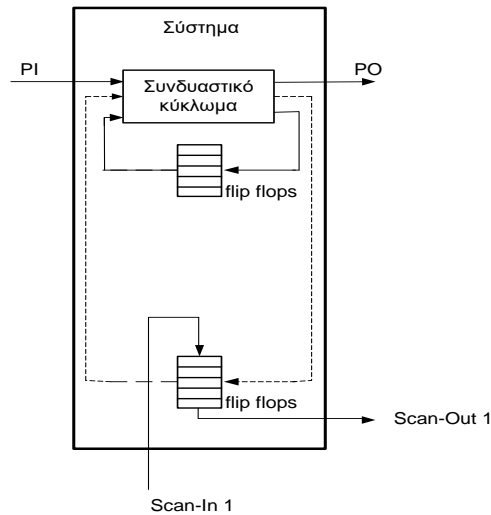
Κατά τη σειριακή λειτουργία των μονοπατιών ολίσθησης στις εισόδους του συνδυαστικού τμήματος του κυκλώματος λαμβάνουν χώρα πολλές αλλαγές τιμών. Αυτό έχει ως συνέπεια πολλοί κόμβοι του συνδυαστικού κυκλώματος να αλλάζουν λογική τιμή. Αυτό συνεπάγεται μεγάλη κατανάλωση ισχύος. Επειδή κατά την ολίσθηση πολλοί κόμβοι του κυκλώματος αλλάζουν τιμή ταυτόχρονα, έχουμε αύξηση όχι μόνο της μέσης τιμής της κατανάλωσης ισχύος αλλά και της μέγιστης τιμής της.

Η αύξηση της μέσης τιμής κατανάλωσης ισχύος κατά τον έλεγχο της ορθής λειτουργίας ενός κυκλώματος μπορεί να οδηγήσει σε υπερθέρμανσή του και μόνιμες βλάβες. Η αύξηση της μέγιστης τιμής κατανάλωσης ισχύος μπορεί να οδηγήσει σε στιγμιαία πτώση της τάσης τροφοδοσίας και προβλήματα θορύβου τα οποία έχουν ως συνέπεια ένα ολοκληρωμένο κύκλωμα που δεν έχει κανένα ελάττωμα να αξιολογηθεί ως ελαττωματικό και να απορριφθεί. Αυτό συνεπάγεται μείωση της απόδοσης (yield) της γραμμής παραγωγής.

Για την μείωση της αύξησης του απαιτούμενου υλικού και του χρόνου καθυστέρησης έχει προταθεί η τεχνική μερικής χρήσης μονοπατιών ολίσθησης (Partial Scan Paths).

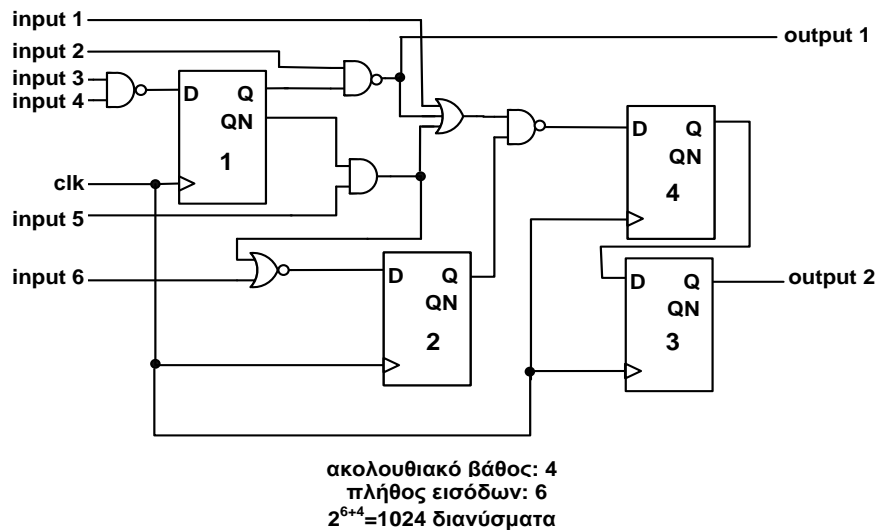
Στο σχήμα 7.35 παρατηρούμε ότι μόνο ένα υποσύνολο των φλιπ-φλοπ του συστήματος αντικαταστάθηκε από φλιπ-φλοπ ολίσθησης. Επομένως από τις κύριες εισόδους του ολοκληρωμένου κυκλώματος μπορούμε να καθορίσουμε το περιεχόμενο μόνο αυτού του υποσυνόλου των φλιπ-φλοπ. Επίσης, από τις κύριες εξόδους του ολοκληρωμένου κυκλώματος μπορούμε να διαβάσουμε το περιεχόμενο μόνο αυτού του υποσυνόλου των φλιπ-φλοπ. Το ερώτημα που προκύπτει είναι τι κερδίζουμε με τη μερική χρήση μονοπατιών ολίσθησης;

Εφόσον αντικαθιστούμε μόνο ένα υποσύνολο των φλιπ-φλοπ του συστήματος με φλιπ-φλοπ ολίσθησης, επιτυγχάνουμε μικρότερη αύξηση του απαιτούμενου εμβαδού για την υλοποίηση του ολοκληρωμένου κυκλώματος και του πλήθους των γραμμών διασύνδεσης. Επίσης, εάν γίνει κατάλληλα η επιλογή των φλιπ-φλοπ του συστήματος που αντικαθίστανται από φλιπ-φλοπ ολίσθησης έχουμε μείωση της καθυστέρησης.



Σχήμα 7.35 Τεχνική μερικής χρήσης μονοπατιών ολίσθησης (Partial Scan Paths)

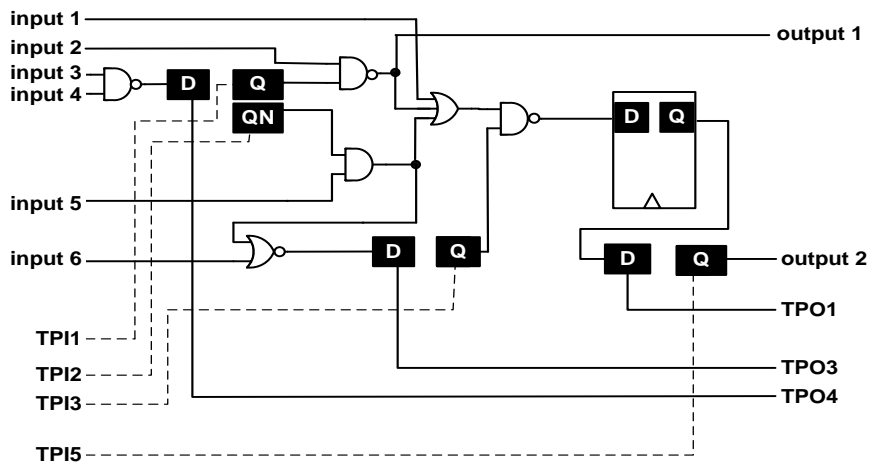
Βέβαια για να έχει νόημα χρησιμοποίησης η τεχνική της μερικής χρήσης μονοπατιών ολίσθησης, θα πρέπει να οδηγεί σε ακολουθιακά κυκλώματα για τα οποία είναι πιο εύκολο να εξάγουμε διανύσματα δοκιμής απ' ότι για το αρχικό ακολουθιακό κύκλωμα. Αυτό και πάλι εξαρτάται από τα συγκεκριμένα φλιπ-φλοπ του συστήματος, τα οποία θα επιλεγούν, για να αντικατασταθούν από φλιπ-φλοπ ολίσθησης.



Σχήμα 7.36

Το ακολουθιακό βάθος ενός κυκλώματος ισούται με το μέγιστο αριθμό των κύκλων ρολογιού που πρέπει να εφαρμοστούν για να θέσουμε τιμές σε (ή να παρατηρήσουμε τιμές από) όλα τα στοιχεία μνήμης που δεν έχουν αντικατασταθεί με φλιπ-φλοπ ολίσθησης.

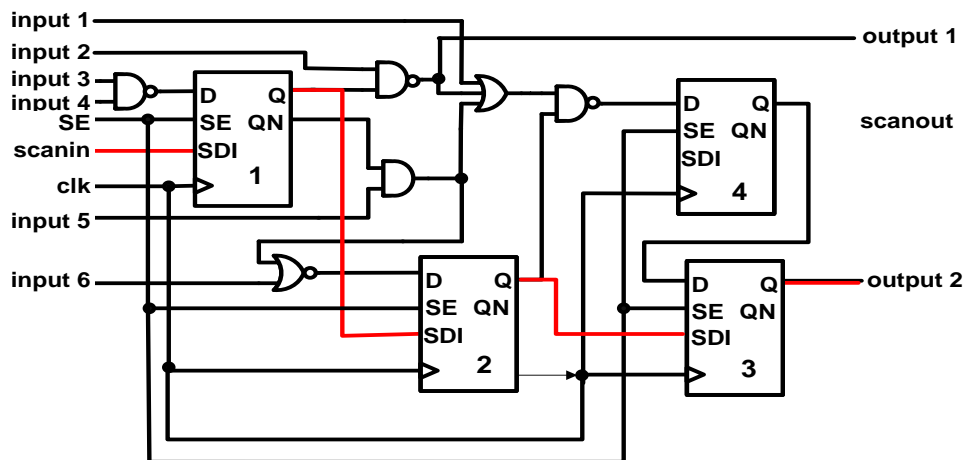
Το ακολουθιακό κύκλωμα του σχήματος 7.36 έχει ακολουθιακό βάθος 4. Θα αντικαταστήσουμε μόνο τα υπ' αριθμόν 1, 2 και 3 φλιπ-φλοπ του συστήματος με φλιπ-φλοπ ολίσθησης.



Ακολουθιακό βάθος=1:
6 είσοδοι και 4 ψευδοείσοδοι και
2 έξοδοι και 3 ψευδοέξοδοι

Σχήμα 7.37

Στο σχήμα 7.37 φαίνεται το κύκλωμα για το οποίο πρέπει να εξάγουμε σύνολο διανυσμάτων δοκιμής. Παρατηρούμε ότι το κύκλωμα δεν είναι πλέον ένα συνδυαστικό κύκλωμα, παραμένει ακολουθιακό κύκλωμα. Όμως αυτό το ακολουθιακό κύκλωμα έχει ακολουθιακό βάθος 1, ενώ το αρχικό είχε ακολουθιακό βάθος 4. Επομένως είναι πιο εύκολο να εξάγουμε γι' αυτό το κύκλωμα ένα σύνολο διανυσμάτων δοκιμής.



Σχήμα 7.38

Στο σχήμα 7.38 φαίνεται με κόκκινη γραμμή το μονοπάτι ολίσθησης που προέκυψε από την αντικατάσταση των υπ' αριθμόν 1, 2 και 3 φλιπ-φλοπ του συστήματος με φλιπ-φλοπ ολίσθησης. Όπως ήδη αναφέραμε στην τεχνική της μερικής χρήσης μονοπατιών ολίσθησης η επιλογή των φλιπ-φλοπ του συστήματος τα οποία θα αντικατασταθούν από φλιπ-φλοπ ολίσθησης επηρεάζει αφενός το εμβαδόν που απαιτείται για την υλοποίηση του κυκλώματος, αφετέρου την ευκολία με την οποία θα εξαχθεί το σύνολο δοκιμής του ακολουθιακού κυκλώματος που θα προκύψει. Επομένως η επιλογή τους είναι σημαντική.

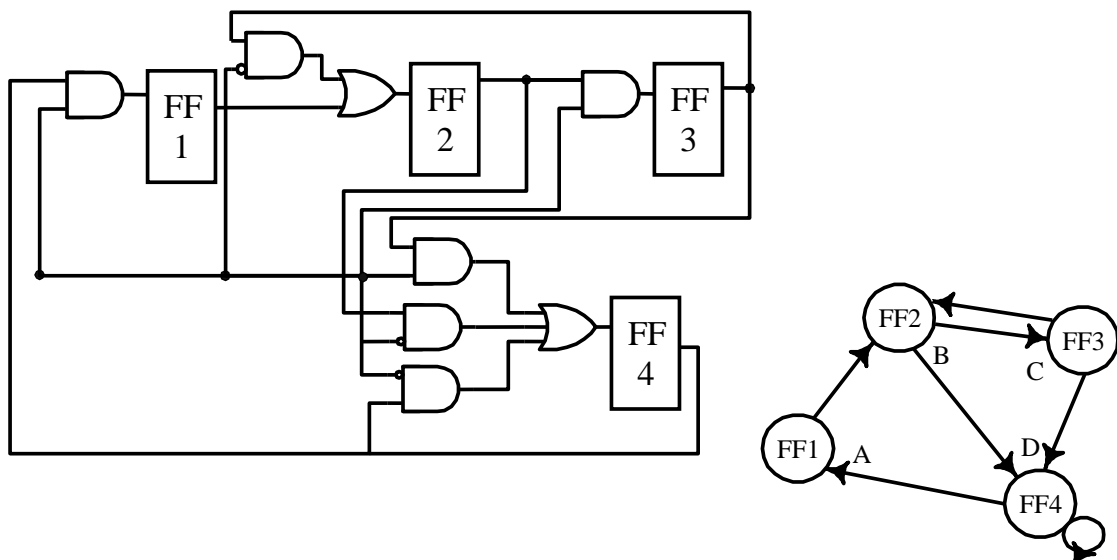
Η επιλογή των φλιπ-φλοπ βασίζεται σε κάποια από τις ακόλουθες γενικές μεθόδους.

- Διαμέριση που βασίζεται στη λειτουργικότητα πχ. μονάδα επεξεργασίας δεδομένων και μονάδα ελέγχου (Functional partitioning : data path & control)
- Κόψιμο των ακολουθιακών μονοπατιών ανάδρασης (Pipelined or feed-forward partial-scan design)
- Στόχευση συγκεκριμένου ακολουθιακού βάθους (Balanced partial scan design : target sequential depth)

Στη διαμέριση που βασίζεται στη λειτουργικότητα το κύκλωμα θεωρείται ως να αποτελείται από τη μονάδα επεξεργασίας δεδομένων και το τμήμα ελέγχου. Συνήθως δεν είναι αποδεκτή η αύξηση της καθυστέρησης στη μονάδα επεξεργασίας δεδομένων. Για το λόγο αυτό αποφεύγεται η αντικατάσταση των στοιχείων μνήμης του συστήματος με στοιχεία μνήμης ολίσθησης. Αντίθετα τα στοιχεία μνήμης του τμήματος ελέγχου αντικαθίστανται με στοιχεία μνήμης ολίσθησης. Αυτή η μέθοδος επιτυγχάνει να βελτιώσει την κάλυψη σφαλμάτων με περιορισμένη υποβάθμιση της απόδοσης του κυκλώματος.

Έχει διαπιστωθεί ότι η δυσκολία εξαγωγής διανυσμάτων δοκιμής για ένα ακολουθιακό κύκλωμα οφείλεται στην ύπαρξη ακολουθιακών δομών ανάδρασης. Το σπάσιμο των ακολουθιακών δομών ανάδρασης καθιστά την εξαγωγή διανυσμάτων δοκιμής υπολογιστικά αποδοτική. Επομένως το θέμα είναι να επιλέξουμε τον ελάχιστο αριθμό στοιχείων μνήμης του συστήματος που πρέπει να αντικατασταθούν με στοιχεία ολίσθησης, ώστε να σπάσουν όλες οι ακολουθιακές δομές ανάδρασης.

Για το σκοπό αυτό κατασκευάζουμε καταρχήν το δομικό γράφο που αντιστοιχεί στο ακολουθιακό κύκλωμα. Στη συνέχεια χρησιμοποιούμε ευρηκτικές μεθόδους για την επιλογή του ελάχιστου πλήθους των φλιπ-φλοπ που θα αντικατασταθούν από φλιπ-φλοπ ολίσθησης ώστε να σπάσουν όλοι οι κύκλοι του γράφου.

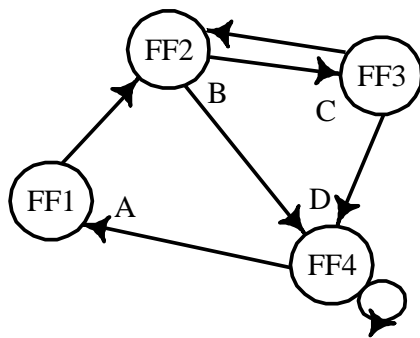


Σχήμα 7.39 Ακολουθιακό κύκλωμα και ο αντίστοιχος δομικός γράφος.

Στο σχήμα 7.39 βλέπετε ένα ακολουθιακό κύκλωμα και τον αντίστοιχο δομικό γράφο. Ο δομικός γράφος είναι ένας κατευθυνόμενος γράφος (directed graph). Σε κάθε φλιπ-φλοπ του ακολουθιακού κυκλώματος αντιστοιχεί μια κορυφή του γράφου. Στο γράφο υπάρχει ένα βέλος από την κορυφή A προς την κορυφή B, μόνον εάν στο ακολουθιακό κύκλωμα υπάρχει κάποιο μονοπάτι μέσω του οποίου το φλιπ-φλοπ που αντιστοιχεί στην κορυφή A του γράφου οδηγεί το φλιπ-φλοπ που αντιστοιχεί στην κορυφή B.

Παρατηρούμε ότι στο δομικό γράφο υπάρχουν τέσσερις κύκλοι. Ο κύκλος $FF2 \rightarrow FF3 \rightarrow FF4 \rightarrow FF2$, ο κύκλος $FF2 \rightarrow FF4 \rightarrow FF1 \rightarrow FF2$, ο κύκλος $FF2 \rightarrow FF3 \rightarrow FF4 \rightarrow FF1 \rightarrow FF2$, ο κύκλος $FF2 \rightarrow FF3 \rightarrow FF2$ και ο κύκλος $FF4 \rightarrow FF4$. Ακόμη και σ' αυτό το πολύ απλό ακολουθιακό κύκλωμα η επιλογή του ελάχιστου πλήθους φλιπ-φλοπ δεν είναι προφανής.

Στη συνέχεια θα παρουσιάσουμε μια ευρητική μέθοδο.



	FF1	FF2	FF3	FF4	total
FF1	0	1	0	0	1
FF2	0	0	1	1	2
FF3	0	1	0	1	2
FF4	1	0	0	1	2
total	1	2	1	3	

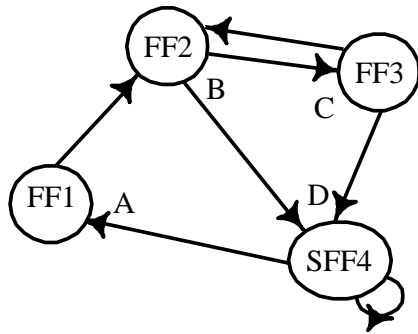
Σχήμα 7.40 Δομικός γράφος και ο αντίστοιχος πίνακας γειτονικότητας.

Από τον δομικό γράφο κατασκευάζουμε τον πίνακα γειτονικότητας (βλέπε σχήμα 7.40). Ο πίνακας γειτονικότητας έχει τόσες γραμμές και τόσες στήλες όσες είναι οι κορυφές του δομικού γράφου. Εάν υπάρχει βέλος από την κορυφή i προς την κορυφή j του γράφου τότε στη θέση $a_{i,j}$ του πίνακα γειτονικότητας θέτουμε την τιμή 1, διαφορετικά θέτουμε την τιμή 0.

Υπάρχουν διάφοροι τρόποι χρησιμοποίησης του πίνακα γειτονικότητας προς εύρεση των φλιπ-φλοπ του συστήματος που πρέπει να αντικατασταθούν με φλιπ-φλοπ ολίσθησης.

Στον πίνακα γειτονικότητας η πρώτη στήλη από δεξιά δίνει σε κάθε γραμμή το πλήθος των βελών που φεύγουν από την αντίστοιχη κορυφή του γράφου. Η τελευταία γραμμή του πίνακα σε κάθε στήλη δίνει το πλήθος των βελών που δείχνουν προς την αντίστοιχη κορυφή του γράφου. Μια ευρητική μέθοδος βασίζεται στην παρατήρηση ότι μια κορυφή θα ανήκει σε πολλούς κύκλους εάν το άθροισμα των βελών που φεύγουν από αυτή και έρχονται σε αυτή είναι μεγάλο. Βασιζόμενοι στη μέθοδο αυτή και χρησιμοποιώντας τον πίνακα γειτονικότητας συμπεραίνουμε ότι το φλιπ-φλοπ FF4, το οποίο αντιστοιχεί σε μια κορυφή του γράφου από την οποία φεύγουν δύο βέλη και έρχονται τρία βέλη (σύνολο 5), πρέπει να αντικατασταθεί από ένα φλιπ-φλοπ ολίσθησης. Παρατηρούμε ότι μετά την αντικατάσταση του φλιπ-φλοπ FF4 του

συστήματος με φλιπ-φλοπ ολίσθησης SFF4, ο μόνος κύκλος που μένει στο δομικό γράφο είναι ο κύκλος $FF2 \rightarrow FF3 \rightarrow FF2$.



	FF1	FF2	FF3	SFF4	total
FF1	0	1	0	0	1
FF2	0	0	1	0	1
FF3	0	1	0	0	1
SFF4	0	0	0	0	0
total	1	2	1	0	

Σχήμα 7.41 Νέος δομικός γράφος και ο αντίστοιχος πίνακας γειτονικότητας.

Αφού επιλέξαμε να αντικαταστήσουμε το φλιπ-φλοπ FF4 του συστήματος με φλιπ-φλοπ ολίσθησης, θα πρέπει να ενημερώσουμε τον πίνακα γειτονικότητας. Μηδενίζουμε λοιπόν όλα τα στοιχεία της γραμμής και της στήλης που αντιστοιχούν στο φλιπ-φλοπ FF4 (βλέπε σχήμα 7.41). Από το νέο πίνακα γειτονικότητας παρατηρούμε ότι το άθροισμα της στήλης και της γραμμής που αντιστοιχεί στο φλιπ-φλοπ FF2 δίνει τη μεγαλύτερη τιμή. Επομένως θα αντικαταστήσουμε το φλιπ-φλοπ FF2 του συστήματος με φλιπ-φλοπ ολίσθησης.

Όσο μικρότερο είναι το ακολουθιακό βάθος τόσο ευκολότερη είναι η εξαγωγή του συνόλου δοκιμής. Για το λόγο αυτό στην προηγούμενη μέθοδο συνεχίζουμε την αντικατάσταση φλιπ-φλοπ του συστήματος με φλιπ-φλοπ ολίσθησης έως ότου πετύχουμε την επιθυμητή τιμή ακολουθιακού βάθους.

Κεφάλαιο 8. Τεχνικές ενσωματωμένου αυτό-ελέγχου (Built-In Self-Test - BIST)

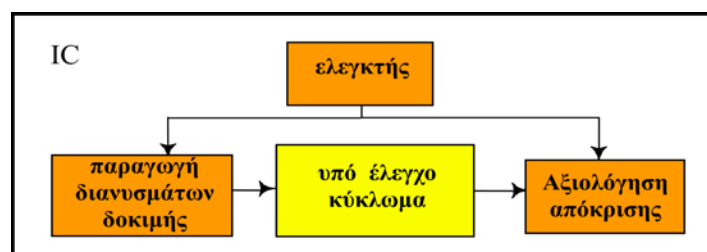
Στην ενότητα αυτή θα ασχοληθούμε με τις τεχνικές ενσωματωμένου αυτό-ελέγχου. Στο σχήμα 1.3 φαίνεται η αρχή του λογικού ελέγχου ορθής λειτουργίας ενός κυκλώματος. Το ερώτημα που τίθεται είναι ποιος τροφοδοτεί το υπό έλεγχο κύκλωμα με τα διανύσματα δοκιμής και ποιος αξιολογεί τις αποκρίσεις του υπό έλεγχο κυκλώματος. Είδαμε ότι στην περίπτωση του εξωτερικού ελέγχου υπάρχει κάποια συσκευή, ATE, η οποία κάνει αυτή τη δουλειά.

Στην περίπτωση του ενσωματωμένου αυτό-ελέγχου στο ίδιο ολοκληρωμένο κύκλωμα με την υπό έλεγχο μονάδα βρίσκονται πρόσθετα κυκλώματα για την παραγωγή διανυσμάτων δοκιμής και τον έλεγχο της απόκρισης του υπό έλεγχο κυκλώματος.

Κίνητρα για την εφαρμογή της τεχνικής του ενσωματωμένου αυτό-ελέγχου είναι ότι η τεχνική αυτή είναι κατάλληλη όχι μόνο για τον έλεγχο της ορθής λειτουργίας ενός ολοκληρωμένου κυκλώματος μετά την κατασκευή του, αλλά και για τον έλεγχο της ορθής του λειτουργίας στο πεδίο της εφαρμογής. Επίσης η τεχνική αυτή μπορεί να βοηθήσει στη διάγνωση των ελαττωμάτων και επομένως στην γρηγορότερη επιδιόρθωση του συστήματος. Ένα ακόμα πλεονέκτημα της τεχνικής αυτής είναι ότι μας απαλλάσσει από την ανάγκη αγοράς πολύπλοκων εξωτερικών συσκευών ελέγχου των οποίων το κόστος είναι πολύ υψηλό.

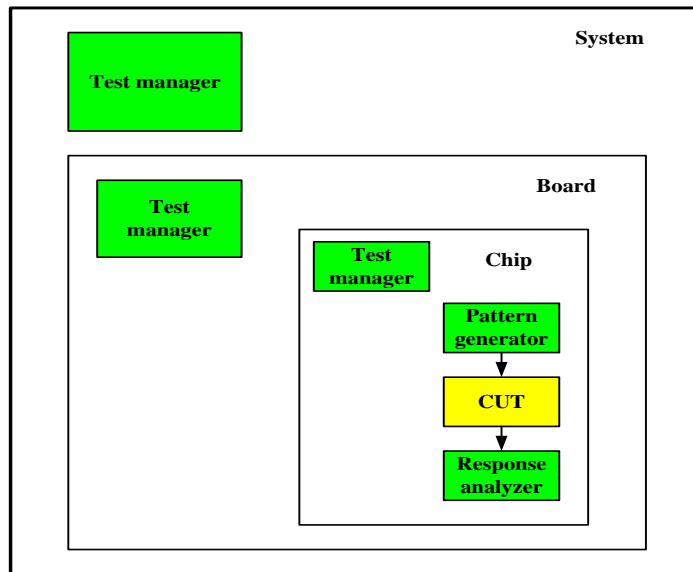
Αν και γρηγορότερα αναφέραμε ότι για την υλοποίηση της τεχνητής του ενσωματωμένου αυτό-ελέγχου υπάρχουν πρόσθετα κυκλώματα στο ίδιο ολοκληρωμένο με την υπό έλεγχο μονάδα, ο αυτο-έλεγχος μπορεί να βασιστεί και στην εκτέλεση λογισμικού. Ωστόσο, σε αυτή την περίπτωση επιτυγχάνεται μικρότερη κάλυψη σφαλμάτων, η διαγνωστική ικανότητα είναι μικρότερη και απαιτείται μεγαλύτερος χρόνος ελέγχου της σωστής λειτουργίας της υπό έλεγχο μονάδας.

Στη συνέχεια θα ασχοληθούμε μόνο με την τεχνική του ενσωματωμένου αυτο-ελέγχου που υλοποιείται με υλικό.



Σχήμα 8.1. Αρχιτεκτονική ενσωματωμένου αυτο-ελέγχου.

Από το σχήμα 8.1 βλέπουμε ότι στην περίπτωση του ενσωματωμένου αυτο-ελέγχου στο ίδιο ολοκληρωμένο κύκλωμα με την υπό έλεγχο μονάδα πρέπει να βρίσκεται ένα κύκλωμα παραγωγής διανυσμάτων δοκιμής (Test Pattern Generator, TPG), ένα κύκλωμα αξιολόγησης της απόκρισης του υπό έλεγχο κυκλώματος (response verification) και ο ελεγκτής της διαδικασίας ενσωματωμένου ελέγχου (test controller).



Σχήμα 8.2. Ιεραρχική εφαρμογή της τεχνικής του ενσωματωμένου αυτο-ελέγχου.

Ένα ακόμη μεγάλο πλεονέκτημα της τεχνικής του ενσωματωμένου αυτο-ελέγχου είναι ότι μπορεί να εφαρμοστεί ιεραρχικά σε όλα τα επίπεδα ενός συστήματος. Για να γίνει ο έλεγχος της ορθής λειτουργίας του συστήματος του σχήματος 8.2, ο διαχειριστής της διαδικασίας ελέγχου ορθής λειτουργίας σε επίπεδο συστήματος (test manager) στέλνει εντολή ελέγχου ορθής λειτουργίας στον διαχειριστή ελέγχου (test manager) κάθε πλακέτας. Ο διαχειριστής ελέγχου της ορθής λειτουργίας κάθε πλακέτας με τη σειρά του στέλνει εντολή στον διαχειριστή ελέγχου της ορθής λειτουργίας κάθε ολοκληρωμένου κυκλώματος της πλακέτας. Σε κάποιο ολοκληρωμένο κύκλωμα είναι δυνατόν να υπάρχουν περισσότερες από μία δομές αυτο-ελέγχου για τον έλεγχο της ορθής λειτουργίας περισσότερων της μιας μονάδων. Σ' αυτή την περίπτωση ο διαχειριστής ελέγχου της ορθής λειτουργίας του ολοκληρωμένου κυκλώματος στέλνει εντολή στον ελεγκτή ελέγχου κάθε μιας δομής αυτο-ελέγχου. Όταν ολοκληρώνεται ο έλεγχος της ορθής λειτουργίας μιας μονάδας ενός ολοκληρωμένου κυκλώματος, ο ελεγκτής ελέγχου αυτής της δομής στέλνει τα αποτελέσματα στον διαχειριστή ελέγχου του ολοκληρωμένου κυκλώματος. Αυτός με τη σειρά του όταν μαζεύει την πληροφορία που αφορά τον έλεγχο της ορθής λειτουργίας όλων των μονάδων του ολοκληρωμένου κυκλώματος στέλνει την πληροφορία αυτή στο διαχειριστή ελέγχου της πλακέτας στην οποία βρίσκεται το ολοκληρωμένο κύκλωμα. Όταν ο διαχειριστής ελέγχου μιας πλακέτας μαζεύει την πληροφορία που αφορά τον έλεγχο της ορθής λειτουργίας όλων των ολοκληρωμένων κυκλωμάτων της πλακέτας, στέλνει την πληροφορία αυτή στο διαχειριστή ελέγχου του συστήματος.

Σε επίπεδο ολοκληρωμένου κυκλώματος έχουμε αύξηση του κόστους σχεδίασης του ολοκληρωμένου κυκλώματος. Αυτό οφείλεται στο γεγονός ότι θα πρέπει να αναπτύξουμε τη διαδικασία και να σχεδιάσουμε το μηχανισμό ενσωματωμένου αυτο-ελέγχου. Έχουμε όμως μείωση του κόστους εξαγωγής του συνόλου δοκιμής. Το κόστος κατασκευής του ολοκληρωμένου κυκλώματος αυξάνεται διότι το εμβαδόν που καταλαμβάνει είναι μεγαλύτερο. Έχουμε όμως μείωση του κόστους ελέγχου της λειτουργίας του ολοκληρωμένου κυκλώματος μετά την κατασκευή του.

Σε επίπεδο πλακέτας έχουμε αύξηση του κόστους σχεδίασης, λόγω της σχεδίασης του διαχειριστή ελέγχου, αλλά μείωση του κόστους ανάπτυξης της διαδικασίας

ελέγχου ορθής λειτουργίας σε σχέση με την περίπτωση που δεν θα εφαρμόζαμε την τεχνική του ενσωματωμένου αυτο-ελέγχου. Έχουμε αύξηση του κόστους κατασκευής της πλακέτας αλλά μείωση του κόστους ελέγχου της ορθής της λειτουργίας. Επίσης έχουμε μείωση του κόστους διάγνωσης και επισκευής.

Level	Design and test generat.	Fabri- cation	Manuf. Test	Maintenance test	Diagnosis and repair	Service interruption
Chips	+ / -	+	-			
Boards	+ / -	+	-		-	
System	+ / -	+	-	-	-	-

+ αύξηση κόστους
 - μείωση κόστους
 +/- η αύξηση του κόστους μπορεί να αντισταθμίσει την μείωση του κόστους

Σχήμα 8.3. Κόστος εφαρμογής της τεχνικής του ενσωματωμένου αυτο-ελέγχου και οφέλη.

Σε επίπεδο συστήματος έχουμε και πάλι αύξηση του κόστους σχεδίασης αλλά μείωση του κόστους ανάπτυξης της διαδικασίας ελέγχου ορθής λειτουργίας του συστήματος. Έχουμε αύξηση του κόστους κατασκευής του συστήματος, αλλά μείωση του κόστους ελέγχου της ορθής του λειτουργίας. Επίσης έχουμε μείωση του κόστους συντήρησης, του κόστους διάγνωσης και επισκευής καθώς και μείωση του χρόνου που το σύστημα μένει εκτός λειτουργίας.

Κύκλωμα παραγωγής διανυσμάτων δοκιμής

Υπάρχουν διάφορα κυκλώματα που μπορούν να χρησιμοποιηθούν για την παραγωγή διανυσμάτων δοκιμής:

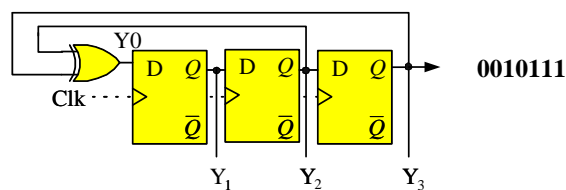
- ROM
- Μετρητής
- Γεννήτρια ψευδοτυχαίων ακολουθιών
 - » Ολισθητές γραμμικής ανάδρασης (*Linear Feedback Shift Register - LFSR*)
 - » Ολισθητές μη γραμμικής ανάδρασης (*Non Linear Feedback Shift Register - LFSR*)
 - » Cellular automata

Στην πρώτη περίπτωση το σύνολο δοκιμής της υπό έλεγχο μονάδας εξάγεται με κάποιον από τους τρόπους που έχουμε μάθει και αποθηκεύεται σε μια μνήμη ROM στο ίδιο ολοκληρωμένο κύκλωμα με τη μονάδα. Κατά τον έλεγχο της ορθής λειτουργίας της μονάδας, η μνήμη προσπελαύνεται κατάλληλα και κάθε διάνυσμα δοκιμής οδηγεί την υπό έλεγχο μονάδα. Το μεγάλο πλεονέκτημα αυτού του τρόπου παραγωγής των διανυσμάτων δοκιμής είναι ο μικρός σχετικά χρόνος που απαιτείται για τον έλεγχο της ορθής λειτουργίας της υπό έλεγχο μονάδας. Δυστυχώς αυτή η μέθοδος δεν είναι δυνατόν να εφαρμοστεί στην πλειονότητα των περιπτώσεων, διότι το σύνολο δοκιμής μιας πολύπλοκης και μεγάλης μονάδας είναι απαγορευτικά

μεγάλο για να ενσωματώσουμε αντίστοιχου μεγέθους μνήμη στο ολοκληρωμένο κύκλωμα.

Ο μετρητής είναι ένα απλό κύκλωμα που θα μπορούσε να χρησιμοποιηθεί για την παραγωγή διανυσμάτων δοκιμής. Το μειονέκτημα της ακολουθίας διανυσμάτων που παράγονται από ένα μετρητή είναι ότι για να λάβει μια έξοδος του μετρητή τη λογική τιμή ένα, θα πρέπει οι λιγότερο σημαντικές εξοδοί του να έχουν λάβει όλους τους δυνατούς συνδυασμούς τιμών. Επομένως ο μετρητής είναι κατάλληλος για την παραγωγή διανυσμάτων δοκιμής μόνο στην περίπτωση του εξαντλητικού ελέγχου της ορθής λειτουργίας μιας μονάδας.

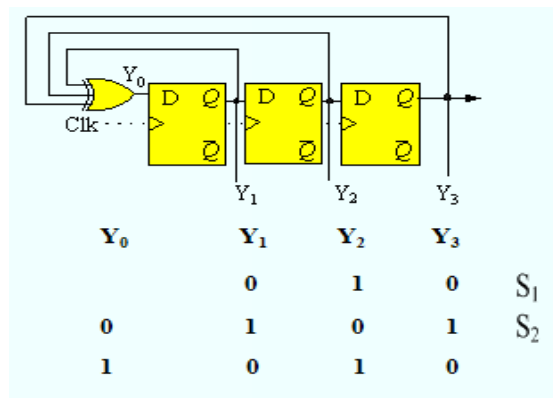
Στη συνέχεια θα ασχοληθούμε με γεννήτριες ψευδοτυχαίων ακολουθιών. Ένα κύκλωμα καλείται γεννήτρια ψευδοτυχαίων ακολουθιών εάν η επόμενη τιμή της εξόδου του είναι εξίσου πιθανόν να έχει τη λογική τιμή 1 και τη λογική τιμή 0, ανεξάρτητα από την προηγούμενη τιμή, αλλά η ακολουθία επαναλαμβάνεται μετά από ένα αριθμό δυαδικών ψηφίων.



S_1	0	1	0
S_2	1	0	1
S_3	1	1	0
S_4	1	1	1
S_5	0	1	1
S_6	0	0	1
S_7	1	0	0

Σχήμα 8.4 Ολισθητής γραμμικής ανάδρασης - παράδειγμα 1.

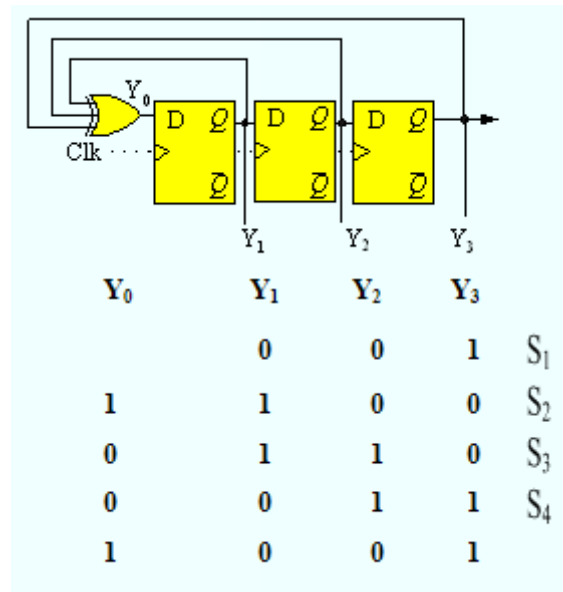
Στο σχήμα 8.4 της διαφάνειας έχουμε ένα παράδειγμα ολισθητή γραμμικής ανάδρασης που αποτελείται από τρεις βαθμίδες. Κάθε βαθμίδα υλοποιείται με ένα D φλιπ-φλοπ. Ανάδραση έχουμε από την έξοδο της δεύτερης και της τρίτης βαθμίδας. Ξεκινώντας από την αρχική κατάσταση S_1 παίρνουμε μια ακολουθία επτά διανυσμάτων. Μετά το 7ο διάνυσμα επαναλαμβάνεται η ίδια ακολουθία.



Σχήμα 8.5 Ολισθητής γραμμικής ανάδρασης - παράδειγμα 2.

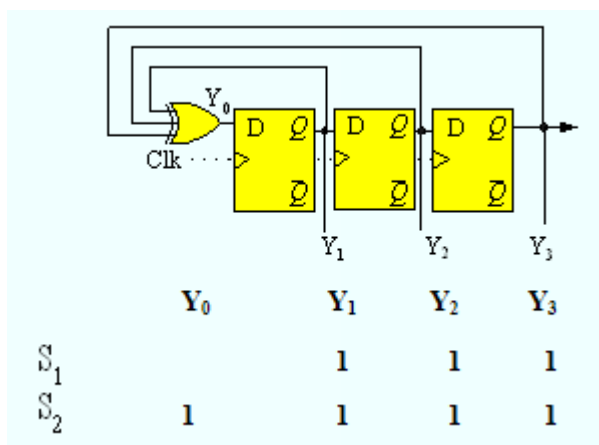
Στο σχήμα 8.5 έχουμε και πάλι ένα παράδειγμα ολισθητή γραμμικής ανάδρασης που αποτελείται από τρεις βαθμίδες. Όμως σ' αυτή την περίπτωση έχουμε ανάδραση από τις εξόδους όλων των βαθμίδων. Ξεκινώντας από την ίδια αρχική κατάσταση όπως και στο προηγούμενο παράδειγμα, παρατηρούμε ότι σ' αυτή την περίπτωση η ακολουθία εξόδου αποτελείται από δύο διανύσματα. Μετά επαναλαμβάνεται η ίδια ακολουθία.

Λαμβάνοντας υπ' όψιν ότι στα προηγούμενα δύο παραδείγματα η μόνη διαφορά ήταν στις αναδράσεις, συμπεραίνουμε ότι το μήκος της ακολουθίας διαφορετικών διανυσμάτων που παράγεται από έναν ολισθητή γραμμικής ανάδρασης εξαρτάται από τις αναδράσεις.



Σχήμα 8.6 Ολισθητής γραμμικής ανάδρασης - παράδειγμα 3.

Στο σχήμα 8.6 έχουμε τον ίδιο ολισθητή γραμμικής ανάδρασης που είχαμε και στο σχήμα 8.5. Τώρα ξεκινώντας από διαφορετική αρχική κατάσταση παρατηρούμε ότι η ακολουθία διαφορετικών διανυσμάτων που παράγεται έχει μήκος 4. Μετά μπορούμε εύκολα να δούμε ότι επαναλαμβάνεται η ίδια ακολουθία. Επομένως το μήκος της ακολουθίας διαφορετικών διανυσμάτων που παράγεται στη γενική περίπτωση από έναν ολισθητή γραμμικής ανάδρασης εξαρτάται και από την αρχική του κατάσταση.

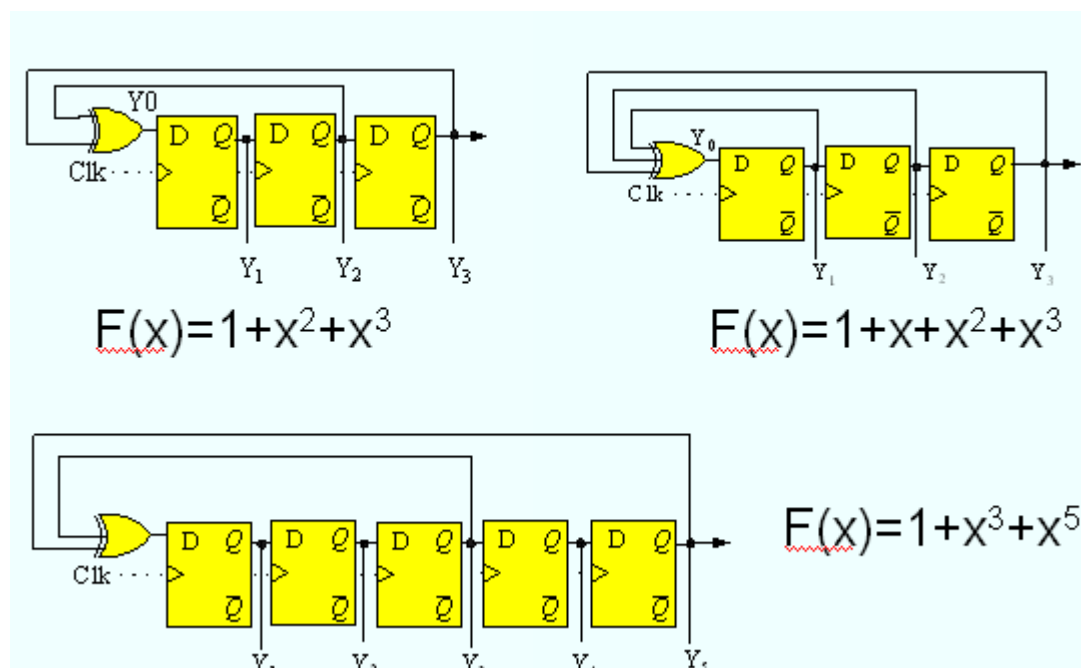


Σχήμα 8.7 Ολισθητής γραμμικής ανάδρασης - παράδειγμα 4.

Στο σχήμα 8.7 έχουμε τον ίδιο ολισθητή γραμμικής ανάδρασης που είχαμε και στο σχήμα 8.6. Τώρα ξεκινώντας από διαφορετική αρχική κατάσταση παρατηρούμε ότι η ακολουθία διαφορετικών διανυσμάτων που παράγεται έχει μήκος 1. Μετά μπορούμε εύκολα να δούμε ότι επαναλαμβάνεται η ίδια ακολουθία.

Είναι προφανές ότι εάν πάρουμε ως αρχική κατάσταση ενός ολισθητή γραμμικής ανάδρασης την όλο μηδενικά κατάσταση, ο ολισθητής δεν μπορεί να αλλάξει κατάσταση.

Σε κάθε ολισθητή γραμμικής ανάδρασης των k βαθμίδων αντιστοιχεί ένα πολυώνυμο βαθμού k , το οποίο καλείται χαρακτηριστικό πολυώνυμο του ολισθητή γραμμικής ανάδρασης. Για να βρούμε το χαρακτηριστικό πολυώνυμο που αντιστοιχεί σε ένα ολισθητή γραμμικής ανάδρασης που ολισθαίνει τα περιεχόμενά του από τα αριστερά προς τα δεξιά, αριθμούμε τις βαθμίδες του από ένα έως k αρχίζοντας από τα αριστερά προς τα δεξιά. Το χαρακτηριστικό πολυώνυμο θα είναι $F(x) = 1+x^i+x^j+ \dots + x^k$ όπου i, j, \dots, k είναι οι βαθμίδες που οδηγούν τις αναδράσεις.



Σχήμα 8.8 Ολισθητές γραμμικής ανάδρασης και τα αντίστοιχα χαρακτηριστικά τους πολυώνυμα- παραδείγματα.

Στο σχήμα 8.8 βλέπετε τρεις διαφορετικούς ολισθητές γραμμικής ανάδρασης και τα χαρακτηριστικά τους πολυώνυμα

Είδαμε τον τρόπο με τον οποίο για κάθε ολισθητή γραμμικής ανάδρασης μπορούμε να πάρουμε το χαρακτηριστικό του πολυώνυμο. Τώρα θα δούμε τον τρόπο με τον οποίο μπορούμε να σχεδιάσουμε ένα ολισθητή γραμμικής ανάδρασης που αντιστοιχεί σε ένα οποιοδήποτε χαρακτηριστικό πολυώνυμο.

Ο ολισθητής γραμμικής ανάδρασης που αντιστοιχεί σε ένα χαρακτηριστικό πολυώνυμο βαθμού k έχει k βαθμίδες. Επομένως, αν το χαρακτηριστικό πολυώνυμο έχει βαθμό k , πρέπει να σχεδιάσουμε k βαθμίδες και να τις αριθμήσουμε από αριστερά προς τα δεξιά. Από τις εξόδους των βαθμίδων που αντιστοιχούν σε εκθέτες των X στο χαρακτηριστικό πολυώνυμο, οδηγούμε τις εισόδους μιας πύλης XOR με αντίστοιχο αριθμό εισόδων και συνδέουμε την έξοδο της πύλης XOR με την είσοδο της πρώτης βαθμίδας του ολισθητή γραμμικής ανάδρασης.

Σε κάθε χαρακτηριστικό πολυώνυμο αντιστοιχεί το αντίστροφό του που παράγεται από τη σχέση $F^*(x)=x^nF(1/x)$. Η ακολουθία που παράγεται από ένα ολισθητή γραμμικής ανάδρασης που αντιστοιχεί στο $F^*(x)$ είναι η αντίστροφη της ακολουθίας που παράγεται από τον ολισθητή γραμμικής ανάδρασης που αντιστοιχεί στο χαρακτηριστικό πολυώνυμο $F(x)$.

Ένα μη ανάγωγο πολυώνυμο βαθμού k έχει τα ακόλουθα δύο χαρακτηριστικά:

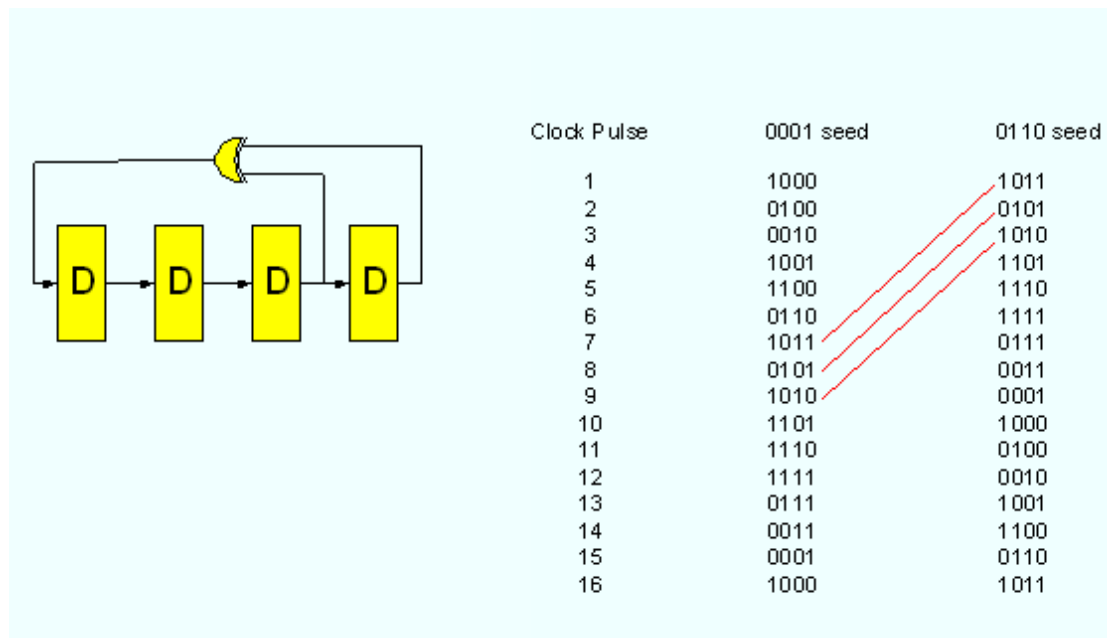
- έχει περιττό αριθμό όρων (περιλαμβάνοντας τη 1)
- διαιρεί ακριβώς το $X^{\mu}+1$, όπου $\mu=2^k-1$

Τα πολυώνυμα x^3+1 και x^3+x^2+x+1 είναι ανάγωγα διότι μπορούν να γραφούν ως γινόμενο άλλων πολυωνύμων: $x^3+1 = (x+1)(x^2+x+1)$ και $x^3+x^2+x+1 = (x+1)(x^2+1)$.

Ένα μη ανάγωγο πολυώνυμο είναι πρωταρχικό εάν ο μικρότερος θετικός ακέραιος k που επιτρέπει το πολυώνυμο να διαιρεί ακριβώς το x^k+1 είναι ο $k=2^{\mu}-1$, όπου μ είναι ο βαθμός του πολυωνύμου.

Ο ολισθητής γραμμικής ανάδρασης είναι μια μηχανή πεπερασμένων καταστάσεων χωρίς κύριες εισόδους. Επομένως, κάθε κατάσταση προσδιορίζεται μονοσήμαντα από την προηγούμενη κατάσταση βάσει των αναδράσεων. Επομένως, εάν μια κατάσταση επαναληφθεί, όλες οι επόμενες καταστάσεις θα επαναληφθούν, άρα η ακολουθία των καταστάσεων είναι περιοδική.

Η ακολουθία που παράγεται από ένα ολισθητή γραμμικής ανάδρασης των k βαθμίδων είναι περιοδική με περίοδο μικρότερη ή ίση του 2^k-1 . Στη γενική περίπτωση η περίοδος της ακολουθίας εξαρτάται από την αρχική κατάσταση. Όταν το χαρακτηριστικό πολυώνυμο είναι μη ανάγωγο η περίοδος είναι ανεξάρτητη της αρχικής κατάστασης (εκτός από την όλο μηδενικά περίπτωση) και είναι ένας παράγων του 2^k-1 , όπου k είναι ο βαθμός του πολυωνύμου. Όταν το χαρακτηριστικό πολυώνυμο είναι πρωταρχικό με βαθμό k , τότε η περίοδος ισούται με 2^k-1 .

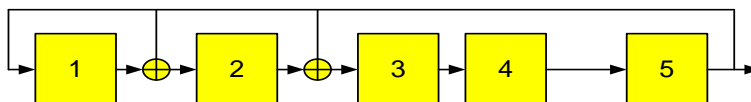


Σχήμα 8.9 Επίδραση αρχικής κατάστασης.

Όπως έχουμε αναφέρει στη γενική περίπτωση η περίοδος της ακολουθίας που παράγεται από ένα ολισθητή γραμμικής ανάδρασης εξαρτάται από την αρχική κατάσταση. Στην περίπτωση όμως που το χαρακτηριστικό πολυώνυμο του ολισθητή γραμμικής ανάδρασης είναι μη ανάγωγο ή πρωταρχικό το μήκος της περιόδου δεν εξαρτάται από την αρχική κατάσταση. Στο σχήμα βλέπουμε ολισθητή γραμμικής

ανάδρασης με πρωταρχικό χαρακτηριστικό πολυώνυμο. Στη δεύτερη στήλη και στην τρίτη δίνεται η ακολουθία που παράγεται εάν ξεκινήσουμε από δύο διαφορετικές αρχικές καταστάσεις. Παρατηρούμε ότι είναι η ίδια ακολουθία μόνο που είναι μετατοπισμένη η μια ως προς την άλλη.

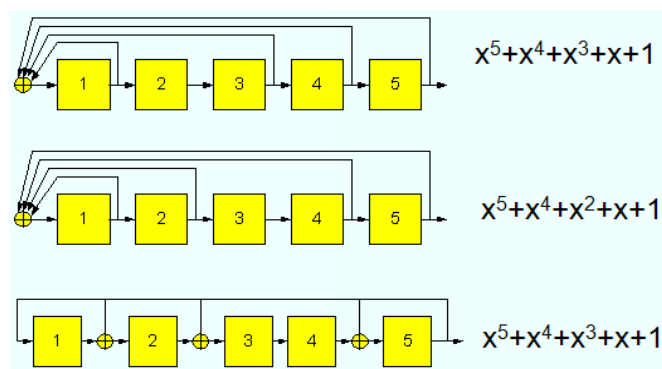
Οι ολισθητές γραμμικής ανάδρασης τους οποίους είδαμε μέχρι τώρα στα παραδείγματά μας είναι τύπου ένα ή με εξωτερικές πύλες XOR. Το χαρακτηριστικό τους είναι ότι οι πύλες XOR εμφανίζονται μόνο στα μονοπάτια ανάδρασης. Υπάρχουν και ολισθητές γραμμικής ανάδρασης στους οποίους οι πύλες XOR εμφανίζονται μόνο μεταξύ των φλιπ-φλοπ. Αυτούς τους ολισθητές γραμμικής ανάδρασης τους καλούμε τύπου δύο ή modular ή με εσωτερικές πύλες XOR.



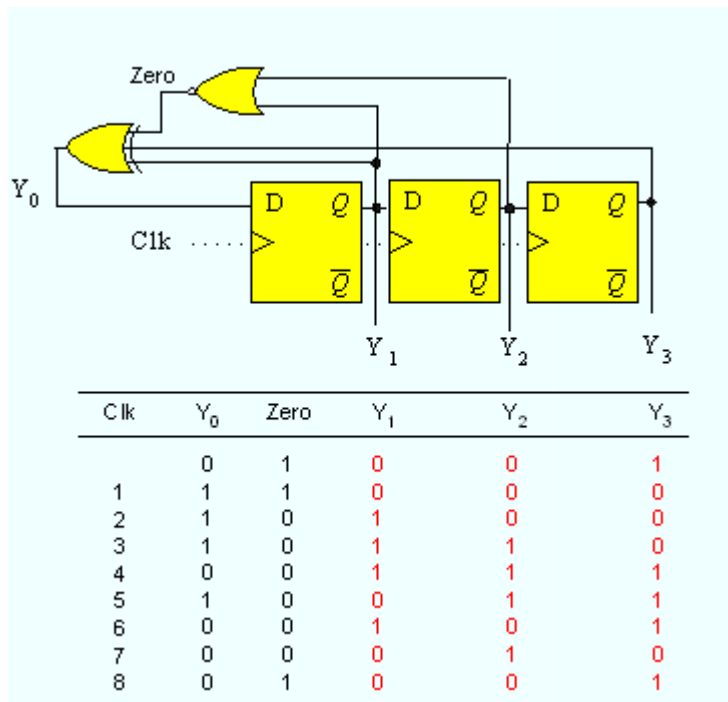
Σχήμα 8.10 Ολισθητής γραμμικής ανάδρασης τύπου ένα και τύπου δύο (Type I and Type II LFSRs)

Στο σχήμα 8.10 βλέπουμε ένα ολισθητή γραμμικής ανάδρασης τύπου δύο. Εάν μας δοθεί το χαρακτηριστικό πολυώνυμο ενός ολισθητή γραμμικής ανάδρασης τύπου ένα, είναι εύκολο να σχεδιάσουμε ένα ολισθητή γραμμικής ανάδρασης τύπου δύο, αρκεί να ακολουθήσουμε τα ακόλουθα βήματα:

- Παίρνουμε το αντίστροφο πολυώνυμο.
- Σχεδιάζουμε τον ολισθητή γραμμικής ανάδρασης τύπου ένα που αντιστοιχεί στο αντίστροφο πολυώνυμο.
- Σχεδιάζουμε τον ολισθητή γραμμικής ανάδρασης τύπου δύο θέτοντας πύλες XOR στα σημεία από τα οποία ξεκινούν οι αναδράσεις στον ολισθητή γραμμικής ανάδρασης τύπου ένα.

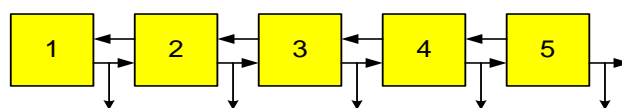


Σχήμα 8.11 Παράδειγμα: ολισθητής γραμμικής ανάδρασης τύπου ένα → τύπου δύο. Στο πρώτο σχήμα μας δίνεται ένας ολισθητής γραμμικής ανάδρασης τύπου ένα και το χαρακτηριστικό του πολυώνυμο. Από το χαρακτηριστικό πολυώνυμο $x^5+x^4+x^3+x+1$ λαμβάνουμε το αντίστροφο πολυώνυμο $x^5+x^4+x^2+x+1$. Σχεδιάζουμε τον ολισθητή γραμμικής ανάδρασης τύπου ένα που αντιστοιχεί στο χαρακτηριστικό πολυώνυμο $x^5+x^4+x^2+x+1$. Στη συνέχεια σχεδιάζουμε τον ολισθητή γραμμικής ανάδρασης τύπου δύο θέτοντας πύλες XOR στα σημεία από τα οποία ξεκινούν οι αναδράσεις στον ολισθητή γραμμικής ανάδρασης τύπου ένα.



Σχήμα 8.12 Παραγωγή όλων των δυνατών συνδυασμών τιμών με ολισθητή γραμμικής ανάδρασης.

Όπως έχουμε ήδη αναφέρει μεταξύ των συνδυασμών που παράγει ένας ολισθητής γραμμικής ανάδρασης δεν μπορεί να βρίσκεται ο όλο μηδενικά συνδυασμός. Αιτία είναι ότι ο όλο μηδενικά συνδυασμός δεν μπορεί να προκύψει εάν ξεκινήσουμε με μια αρχική κατάσταση η οποία δεν είναι η όλο μηδενικά κατάσταση. Επίσης εάν ξεκινήσουμε με την αρχική κατάσταση όλο μηδενικά δεν παράγεται κανένα άλλο διάνυσμα. Για να παράγουμε όλους τους δυνατούς συνδυασμούς τιμών θα πρέπει να προσθέσουμε στον ολισθητή γραμμικής ανάδρασης ένα μη γραμμικό στοιχείο, όπως είναι η πύλη NOR στο παράδειγμα του σχήματός μας. Από τον πίνακα που παρατίθεται μπορούμε εύκολα να δούμε ότι παράγονται όλοι και οι οκτώ δυνατοί συνδυασμοί τιμών.



Σχήμα 8.13 Κυψελωτά αυτόματα (Cellular Automata).

Στα κυψελωτά αυτόματα κάθε κυψελίδα συνίσταται από ένα φλιπ-φλοπ και ένα συνδυαστικό κύκλωμα, το οποίο υλοποιεί τη συνάρτηση επόμενης κατάστασης. Η κατάσταση q της κυψελίδας i^{th} τη χρονική περίοδο $t+1$ δηλώνεται ως $q_i^{t+1} = f(q_i^t, q_{i-1}^t, q_{i+1}^t)$, όπου f είναι συνάρτηση επόμενης κατάστασης. Παρατηρούμε επομένως ότι υπάρχουν 256 συναρτήσεις επόμενης κατάστασης, οι οποίες καλούνται κανόνες του κυψελωτού αυτόματου (CA rules). Παρατηρούμε επίσης ότι κάθε κυψελίδα επικοινωνεί το πολύ με την προηγούμενή της και την επόμενη της.

Παράδειγμα:

Κατάσταση τριών γει- τονικών κυψελίδων:	111	110	101	100	011	010	001	000
Next state (CA rule-150):	1	0	0	1	0	1	1	0
Next state (CA rule-90):	0	1	0	1	1	0	1	0

Κάθε κανόνας είναι το δεκαδικό ισοδύναμο των οκτώ τιμών επόμενης κατάστασης. Στο παράδειγμα δίνονται όλοι οι συνδυασμοί καταστάσεων στις οποίες είναι δυνατόν να βρεθούν τρεις γειτονικές κυψελίδες του κυψελωτού αυτόματου. Η συνάρτηση επόμενης κατάστασης της μεσαίας, από τις τρεις, κυψελίδας καθορίζει την τιμή που θα έχει αυτή η κυψελίδα την επόμενη χρονική περίοδο. Άρα για κάθε συνδυασμό τιμών των τριών κυψελίδων καθορίζουμε τη λογική τιμή 0 ή 1 που θα έχει η κυψελίδα την επόμενη χρονική περίοδο. Όπως φαίνεται στο παράδειγμα η σειρά των 0 και 1 είναι η δυαδική αναπαράσταση ενός δεκαδικού αριθμού. Η πρώτη γραμμή είναι η δυαδική αναπαράσταση του δεκαδικού αριθμού 150, ενώ η δεύτερη του δεκαδικού αριθμού 90, άρα έχουμε τους κανόνες 150 και 90 αντίστοιχα.

CA rule	Boolean algebra
Rule 90 :	$q_i^{t+1} = q_{i-1}^t \oplus q_{i+1}^t$
Rule 150 :	$q_i^{t+1} = q_{i-1}^t \oplus q_i^t \oplus q_{i+1}^t$
Rule 60 :	$q_i^{t+1} = q_{i-1}^t \oplus q_i^t$
Rule 102 :	$q_i^{t+1} = q_i^t \oplus q_{i+1}^t$
Rule 240 :	$q_i^{t+1} = q_{i-1}^t$

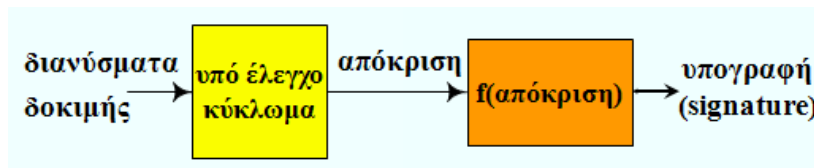
Σχήμα 8.14 Κανόνες του κυψελωτού αυτόματου – παραδείγματα.

Στον πίνακα 8.14 βλέπουμε κάποιους κανόνες και τις αντίστοιχες λογικές συναρτήσεις, της συνάρτησης επόμενης κατάστασης που υλοποιούν αυτούς τους κανόνες.

Βάσει της αρχής του λογικού ελέγχου ορθής λειτουργίας ενός κυκλώματος, θα πρέπει η απόκριση του υπό έλεγχο κυκλώματος να συγκριθεί με την αναμενόμενη απόκριση. Εάν η απόκριση του υπό έλεγχο κυκλώματος είναι διαφορετική της αναμενόμενης, τότε εξάγουμε το συμπέρασμα ότι το υπό έλεγχο κύκλωμα έχει σφάλμα, διαφορετικά θεωρούμε ότι δεν έχει. Όμως ο όγκος των δεδομένων της απόκρισης του υπό ελέγχου κυκλώματος μπορεί να είναι απαγορευτικά μεγάλος, ώστε να αποθηκεύσουμε την αναμενόμενη απόκριση σε μια μνήμη που θα προσθέσουμε για το σκοπό αυτό στο ολοκληρωμένο κύκλωμα. Επομένως είμαστε αναγκασμένοι να προσπαθήσουμε να μειώσουμε τον όγκο των δεδομένων της απόκρισης του υπό έλεγχο κυκλώματος. Στο σχήμα 8.15 βλέπουμε ότι η απόκριση του υπό έλεγχο κυκλώματος οδηγεί ένα κύκλωμα το οποίο παράγει ένα μικρό σχετικά αριθμό από δυαδικά ψηφία που καλούμε υπογραφή (signature).

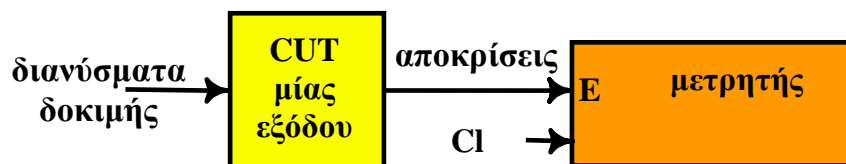
Υπάρχει η πιθανότητα κάποιο ή κάποια σφάλματα που θα ανιχνεύονταν εάν λαμβάναμε υπόψη μας την απόκριση του υπό έλεγχο κυκλώματος και την αναμενόμενη απόκριση, τώρα που γνωρίζουμε μόνο την αναμενόμενη υπογραφή και την υπογραφή του υπό έλεγχο κυκλώματος να μην ανιχνεύονται. Αυτό οφείλεται στο

γεγονός ότι δεν είναι δυνατόν να έχουμε μια προς μια απεικόνιση αποκρίσεων σε υπογραφές. Επομένως είναι δυνατόν η υπογραφή που αντιστοιχεί σε μια λάθος απόκριση να είναι ίδια με την υπογραφή που αντιστοιχεί στην αναμενόμενη απόκριση (aliasing).



Σχήμα 8.15 Μείωση του όγκου δεδομένων της απόκρισης (compaction techniques).

Έχουν προταθεί διάφορες τεχνικές μείωσης του όγκου των δεδομένων της απόκρισης (compaction techniques), όπως η μέτρηση μονάδων, η μέτρηση των μεταβάσεων στην απόκριση του υπό έλεγχο κυκλώματος, ο έλεγχος ισοτιμίας και η χρησιμοποίηση ολισθητών γραμμικής ανάδρασης.



Σχήμα 8.16 Μείωση του όγκου των δεδομένων απόκρισης μετρώντας το πλήθος των μονάδων.

Όταν χρησιμοποιούμε αυτή την τεχνική γνωρίζουμε μόνο το πλήθος των μονάδων στην αναμενόμενη απόκριση και μετράμε το πλήθος των μονάδων στην απόκριση του υπό έλεγχο κυκλώματος. Για να ληφθεί η απόφαση σχετικά με το εάν το υπό έλεγχο κύκλωμα έχει κάποιο ελάττωμα ή όχι συγκρίνουμε τον αριθμό των μονάδων στην απόκριση με τον αναμενόμενο αριθμό μονάδων.

Θεωρούμε ότι το υπό έλεγχο κύκλωμα έχει μόνο μια έξοδο. Η έξοδος αυτή οδηγεί την είσοδο ενεργοποίησης ενός μετρητή. Η είσοδος χρονισμού του μετρητή οδηγείται με ένα σήμα χρονισμού συχνότητας ίσης με τη συχνότητα με την οποία εφαρμόζουμε τα διανύσματα δοκιμής στο υπό έλεγχο κύκλωμα. Ο μετρητής είναι των $\lceil \log_2 N \rceil$ δυαδικών ψηφίων, όπου N είναι το πλήθος των διανυσμάτων που αποτελούν το σύνολο δοκιμής.

Υποθέτουμε ότι στην αναμενόμενη απόκριση του υπό έλεγχο κυκλώματος υπάρχουν k μονάδες σε συγκεκριμένες θέσεις. Το πλήθος όλων των δυνατών αποκρίσεων του υπό έλεγχο κυκλώματος είναι 2^N , από αυτές μόνο μια είναι η σωστή και όλες οι άλλες είναι λάθος. Ωστόσο, υπάρχουν $(N!)/[k! \times (N-k)!]$ αποκρίσεις που έχουν μόνο k μονάδες. Οποιαδήποτε απόκριση με k μονάδες θεωρείται σωστή. Επομένως υπάρχει η πιθανότητα κάποιο ή κάποια σφάλματα που θα ανιχνεύονταν εάν λαμβάναμε υπόψη μας την απόκριση του υπό έλεγχο κυκλώματος και την αναμενόμενη απόκριση, τώρα που γνωρίζουμε μόνο το αναμενόμενο πλήθος μονάδων και το πλήθος των μονάδων στην απόκριση του υπό έλεγχο κυκλώματος δεν ανιχνεύονται (aliasing). Εάν θεωρήσουμε ότι όλες οι $2^N - 1$ λάθος αποκρίσεις έχουν την ίδια πιθανότητα εμφάνισης, γεγονός βέβαιο το οποίο δεν είναι αληθές, τότε η πιθανότητα απόκρυψης κάποιου σφάλματος δίνεται από τη σχέση:

$$P(N/k) = [(N!)/[k! \times (N-k)!] - 1] / (2^N - 1).$$

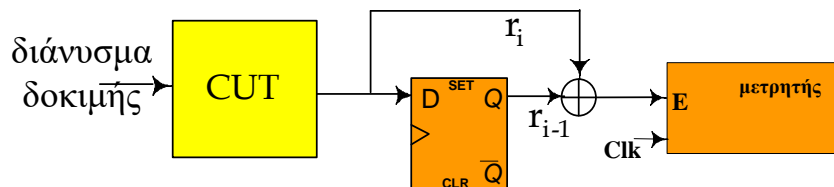
κ	$P(N/\kappa)$
0	0
1	0.047
2	0.157
3	0.268
4	0.268
5	0.157
6	0.047
7	0

Σχήμα 8.17

Στο σχήμα 8.17 δίνουμε την πιθανότητα απόκρυψης σφάλματος όταν η απόκριση είναι των 7 δυαδικών ψηφίων και το πλήθος των μονάδων στην αναμενόμενη απόκριση είναι από 0 έως και επτά. Παρατηρούμε ότι η πιθανότητα απόκρυψης σφάλματος λαμβάνει την ελάχιστη τιμή όταν το πλήθος των μονάδων στην αναμενόμενη απόκριση είναι κοντά στην ελάχιστη ή στη μέγιστη τιμή. Ωστόσο είναι πολύ δύσκολο να βάλουμε έναν ακόμα περιορισμό στη διαδικασία εξαγωγής του συνόλου δοκιμής ενός κυκλώματος και να προσπαθήσουμε να εξάγουμε ένα σύνολο δοκιμής το οποίο έχει ως συνέπεια το πλήθος των μονάδων του υπό έλεγχο κυκλώματος να είναι κοντά στην ελάχιστη ή τη μέγιστη τιμή.

Στην περίπτωση που η υπό έλεγχο μονάδα έχει πολλές εξόδους μπορούμε να χρησιμοποιήσουμε μια από τις ακόλουθες τεχνικές:

- Ένα μετρητή για κάθε έξοδο του κυκλώματος. Σ' αυτή την περίπτωση δεν επηρεάζεται ο χρόνος ελέγχου της ορθής λειτουργίας, αλλά έχουμε σημαντική αύξηση του απαιτούμενου υλικού και υπογραφή με μεγαλύτερο πλήθος δυαδικών ψηφίων.
- Ολισθητής παράλληλης φόρτωσης, στον οποίο αποθηκεύονται παράλληλα οι τιμές των εξόδων του υπό έλεγχο κυκλώματος. Στη συνέχεια το περιεχόμενο του ολισθητή μεταφέρεται σειριακά στην έξοδό του και οδηγεί την είσοδο ενεργοποίησης ενός μετρητή. Αντί του ολισθητή παράλληλης φόρτωσης μπορούμε να χρησιμοποιήσουμε έναν πολυπλέκτη ώστε κάθε έξοδος του υπό έλεγχο κυκλώματος να οδηγεί περιοδικά την είσοδο ενεργοποίησης ενός μετρητή. Σ' αυτές τις περιπτώσεις έχουμε μικρή αύξηση του απαιτούμενου υλικού αλλά μεγαλύτερο χρόνο ελέγχου της σωστής λειτουργίας.



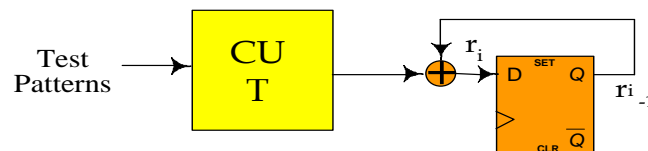
Σχήμα 8.18 Μείωση του όγκου των δεδομένων απόκρισης μετρώντας το πλήθος των μεταβάσεων.

Ένας άλλος τρόπος μείωσης του όγκου των δεδομένων απόκρισης είναι η μέτρηση του πλήθους των μεταβάσεων (βλέπε σχήμα 8.18). Όταν χρησιμοποιούμε αυτή την τεχνική γνωρίζουμε μόνο το πλήθος των μεταβάσεων στην αναμενόμενη απόκριση και μετράμε το πλήθος των μεταβάσεων στην απόκριση του υπό έλεγχο κυκλώματος.

Για να ληφθεί η απόφαση σχετικά με το εάν το υπό έλεγχο κύκλωμα έχει κάποιο ελάττωμα ή όχι συγκρίνουμε τον αριθμό των μεταβάσεων στην απόκριση με τον αναμενόμενο αριθμό μεταβάσεων.

Θεωρούμε ότι το υπό έλεγχο κύκλωμα έχει μόνο μια έξοδο. Για να ανιχνεύσουμε μια μετάβαση αρκεί να συγκρίνουμε την τρέχουσα τιμή της εξόδου με την προηγούμενη τιμή της. Αυτό επιτυγχάνεται, όπως φαίνεται στο σχήμα 8.18, χρησιμοποιώντας ένα D φλιπ-φλοπ, το οποίο αρχικοποιείται στη λογική τιμή 0, και μια πύλη XOR. Η έξοδος της XOR οδηγεί την είσοδο ενεργοποίησης ενός μετρητή. Η είσοδος χρονισμού του μετρητή οδηγείται με ένα σήμα χρονισμού συχνότητας ίσης με τη συχνότητα με την οποία εφαρμόζουμε τα διανύσματα δοκιμής στο υπό έλεγχο κύκλωμα. Ο μετρητής είναι των $\lceil \log_2(N-1) \rceil$ δυαδικών ψηφίων, όπου N είναι το πλήθος των διανυσμάτων που αποτελούν το σύνολο δοκιμής.

Υποθέτουμε ότι στην αναμενόμενη απόκριση του υπό έλεγχο κυκλώματος υπάρχουν κ μεταβάσεις σε συγκεκριμένες θέσεις. Το πλήθος όλων των δυνατών αποκρίσεων του υπό έλεγχο κυκλώματος είναι 2^N , από αυτές μόνο μια είναι η σωστή και όλες οι άλλες είναι λάθος. Ωστόσο, υπάρχουν $2 \times [(N-1)! / (\kappa! \times (N-1-\kappa)!)]$ αποκρίσεις που έχουν μόνο κ μεταβάσεις (πρέπει να σημειώσουμε ότι ο παράγον 2 εμφανίζεται διότι εμείς μετράμε μεταβάσεις και όχι το είδος των μεταβάσεων, δηλαδή δεν εξετάζουμε εάν η μετάβαση είναι από 1 σε 0 ή από 0 σε 1). Οποιαδήποτε απόκριση με κ μεταβάσεις θεωρείται σωστή. Επομένως υπάρχει η πιθανότητα κάποιο ή κάποια σφάλματα που θα ανιχνεύονταν εάν λαμβάναμε υπόψη μας την απόκριση του υπό έλεγχο κυκλώματος και την αναμενόμενη απόκριση, τώρα που γνωρίζουμε μόνο το αναμενόμενο πλήθος μεταβάσεων και το πλήθος των μεταβάσεων στην απόκριση του υπό έλεγχο κυκλώματος δεν ανιχνεύονται (aliasing). Εάν θεωρήσουμε ότι όλες οι $2^N - 1$ λάθος αποκρίσεις έχουν την ίδια πιθανότητα εμφάνισης, γεγονός βέβαιο το οποίο δεν είναι αληθές, τότε η πιθανότητα απόκρυψης κάποιου σφάλματος δίνεται από τη σχέση: $P(N/\kappa) = [2 \times [(N-1)! / (\kappa! \times (N-1-\kappa)!)] - 1] / (2^N - 1)$.

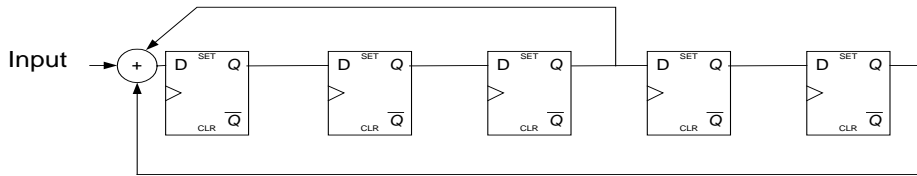


Σχήμα 8.19 Μείωση του όγκου των δεδομένων απόκρισης, κρατώντας μόνο την ισοτιμία της απόκρισης.

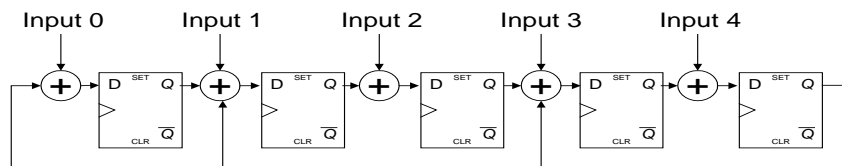
Μία άλλη μέθοδος είναι η μείωση του όγκου των δεδομένων απόκρισης, κρατώντας μόνο την ισοτιμία της απόκρισης. Όταν χρησιμοποιούμε αυτή την τεχνική γνωρίζουμε μόνο την ισοτιμία της αναμενόμενης απόκρισης και υπολογίζουμε την ισοτιμία της απόκρισης του υπό έλεγχο κυκλώματος. Για να ληφθεί η απόφαση σχετικά με το εάν το υπό έλεγχο κύκλωμα έχει κάποιο ελάττωμα ή όχι συγκρίνουμε την ισοτιμία της απόκρισης με την αναμενόμενη ισοτιμία.

Θεωρούμε ότι το υπό έλεγχο κύκλωμα έχει μόνο μια έξοδο. Για να υπολογίσουμε την ισοτιμία της απόκρισης, όπως φαίνεται στο σχήμα 8.19, χρησιμοποιούμε ένα D φλιπ-φλοπ, το οποίο αρχικοποιείται στη λογική τιμή 0, και μια πύλη XOR.

Είναι πολύ εύκολο να δούμε ότι εάν θεωρήσουμε ότι όλες οι $2^N - 1$ λάθος αποκρίσεις έχουν την ίδια πιθανότητα εμφάνισης, γεγονός βέβαιο το οποίο δεν είναι αληθές, η πιθανότητα απόκρυψης κάποιου σφάλματος δίνεται από τη σχέση: $P(N) = [(2^N/2) - 1] / (2^N - 1)$.



Σχήμα 8.20 Μείωση του όγκου των δεδομένων απόκρισης με τη χρήση ολισθητών γραμμικής ανάδρασης μιας εισόδου (Single Input Signature Register, SISR).



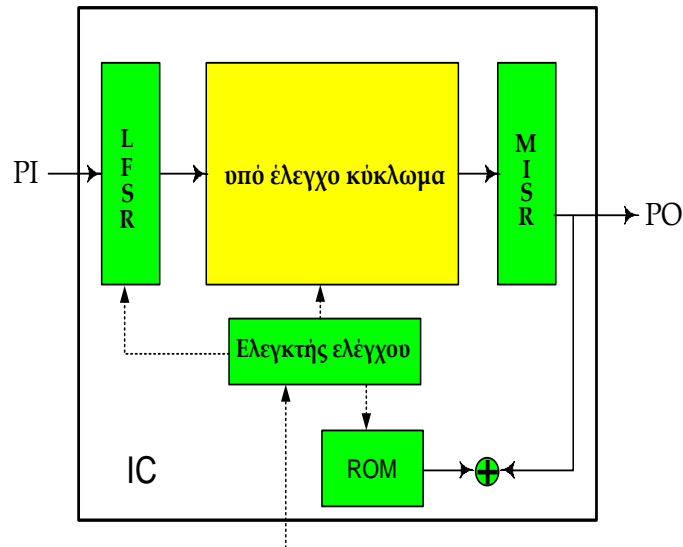
Σχήμα 8.21 Μείωση του όγκου των δεδομένων απόκρισης με τη χρήση ολισθητών γραμμικής ανάδρασης πολλών εισόδων (Multiple Input Signature Register, MISR)

Στα σχήματα 8.20 και 8.21 βλέπουμε ένα ολισθητή γραμμικής ανάδρασης μιας εισόδου και ένα ολισθητή γραμμικής ανάδρασης πολλών εισόδων αντίστοιχα. Έχει αποδειχτεί ότι εάν θεωρήσουμε ότι όλες οι λάθος αποκρίσεις έχουν την ίδια πιθανότητα εμφάνισης, τότε η πιθανότητα απόκρυψης σφάλματος (aliasing) ισούται με $1/2^k$, όπου k και είναι το πλήθος των βαθμίδων του SISR ή του MISR.

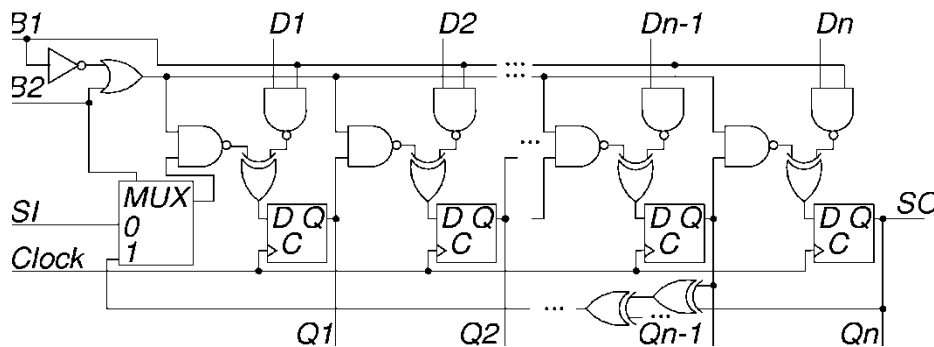
Ανάλογα με το πλήθος των περιόδων του σήματος χρονισμού που απαιτούνται για την εφαρμογή ενός διανύσματος δοκιμής στο υπό έλεγχο κύκλωμα μπορούμε να ταξινομήσουμε τα σχήματα ενσωματωμένου αυτο-ελέγχου σε δύο μεγάλες κατηγορίες.

- Εφαρμογή ενός διανύσματος δοκιμής σε κάθε κύκλο του σήματος χρονισμού (Test Per Clock BIST).
- Εφαρμογή ενός διανύσματος δοκιμής σε k κύκλους του σήματος χρονισμού, όπου η τιμή του k εξαρτάται από τον τρόπο υλοποίησης των μονοπατιών ολίσθησης (Test Per Scan BIST).

Στο σχήμα 8.22 βλέπουμε τη γενική αρχιτεκτονική του ενσωματωμένου αυτο-ελέγχου στην περίπτωση που σε κάθε κύκλο του σήματος χρονισμού στο υπό έλεγχο κύκλωμα εφαρμόζεται ένα διάνυσμα δοκιμής. Η απόκριση του υπό έλεγχο κυκλώματος, λόγω της εφαρμογής ενός διανύσματος δοκιμής, οδηγεί ένα ολισθητή γραμμικής ανάδρασης με πολλές εισόδους. Μετά την εφαρμογή και του τελευταίου διανύσματος δοκιμής του συνόλου δοκιμής το περιεχόμενο του ολισθητή γραμμικής ανάδρασης, δηλαδή η υπογραφή της απόκρισης του υπό έλεγχο κυκλώματος, συγκρίνεται με την αναμενόμενη υπογραφή η οποία είναι αποθηκευμένη σε μια μνήμη ROM. Η όλη διαδικασία του ελέγχου της ορθής λειτουργίας γίνεται κάτω από τον έλεγχο του ελεγκτή ελέγχου.



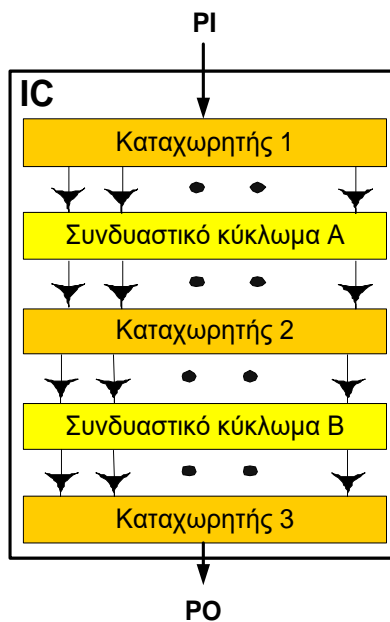
Σχήμα 8.22 Εφαρμογή ενός διανύσματος δοκιμής σε κάθε κύκλο του σήματος χρονισμού.



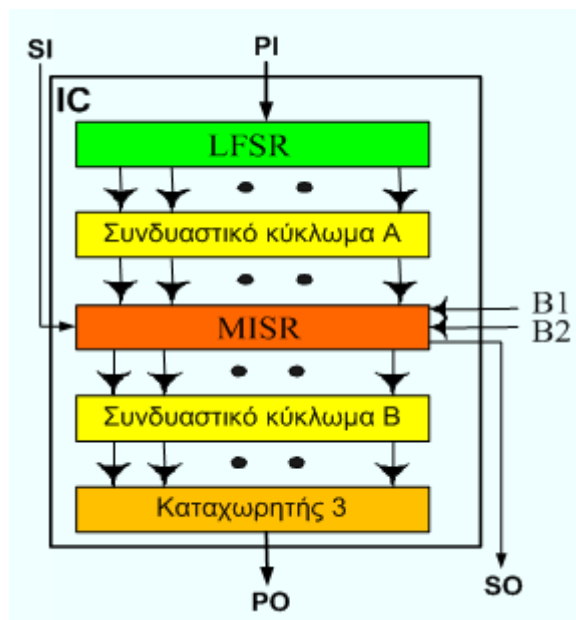
Σχήμα 8.23 Καταχωρητής BILBO (Built-in Logic Block Observer).

Στο σχήμα 8.23 δίνεται η λογική δομή ενός καταχωρητή BILBO (Built-in Logic Block Observer). Ο καταχωρητής αυτός έχει δύο εισόδους ελέγχου B1 και B2 και ανάλογα των τιμών με τις οποίες τις οδηγούμε λειτουργεί ως ένας κανονικός καταχωρητής παράλληλης φόρτωσης ή ως ένας ολισθητής με είσοδο SI και έξοδο SO ή ως μια γεννήτρια ψευδοτυχαίων διανυσμάτων ή ως MISR.

Στη συνέχεια θα δούμε ένα παράδειγμα χρησιμοποίησης καταχωρητών BILBO για τον έλεγχο της ορθής λειτουργίας ενός ολοκληρωμένου κυκλώματος. Στο ολοκληρωμένο κύκλωμα του σχήματος 8.24 παρατηρούμε ότι ο καταχωρητής 1 οδηγείται από κύριες εισόδους του ολοκληρωμένου κυκλώματος και οδηγεί το συνδυαστικό κύκλωμα A. Οι έξοδοι του συνδυαστικού κυκλώματος A οδηγούν τις εισόδους του καταχωρητή 2. Οι έξοδοι του καταχωρητή 2 οδηγούν τις εισόδους του συνδυαστικού κυκλώματος B που με τη σειρά του οδηγεί τις εισόδους του καταχωρητή 3. Οι έξοδοι του καταχωρητή 3 είναι οι κύριες έξοδοι του ολοκληρωμένου κυκλώματος.



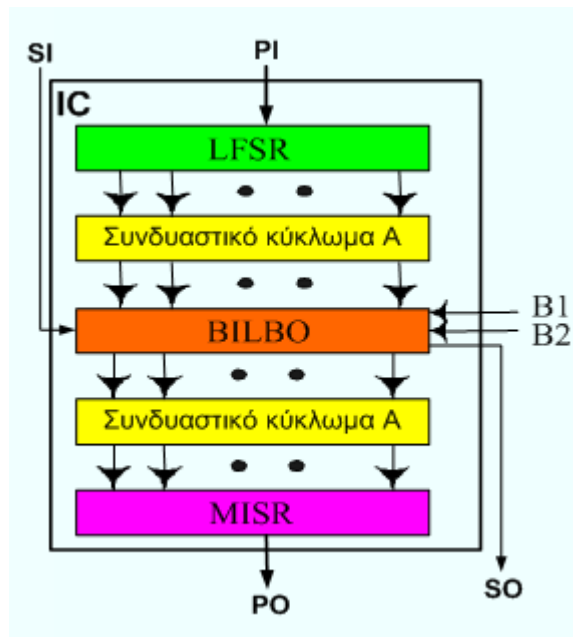
Σχήμα 8.24 Παράδειγμα εφαρμογής καταχωρητών BILBO.



Σχήμα 8.25 Έλεγχος της ορθής λειτουργίας του συνδυαστικού κυκλώματος Α.

Για τον έλεγχο της ορθής λειτουργίας του συνδυαστικού κυκλώματος Α ο καταχωρητής 1 θα χρησιμοποιηθεί για την παραγωγή διανυσμάτων δοκιμής και ο καταχωρητής 2 ως MISR. Επομένως, πρέπει καταρχήν να θέσουμε τους καταχωρητές 1 και 2 στις επιθυμητές αρχικές καταστάσεις. Οι εισοδοί του καταχωρητή 1 είναι κύριες εισοδοί του ολοκληρωμένου κυκλώματος, επομένως ο καταχωρητής 1 δεν είναι αναγκαίο να λειτουργεί και ως ολισθητής. Για να θέσουμε όμως αρχική τιμή στον καταχωρητή 2 θα πρέπει αυτός να μπορεί να τεθεί και σε λειτουργία ολισθητή.

Στη συνέχεια θα πρέπει να θέσουμε τον καταχωρητή 1 σε κατάσταση παραγωγής ψευδοτυχαίων διανυσμάτων δοκιμής και τον καταχωρητή 2 σε λειτουργία MISR. Μετά το τέλος της εφαρμογής των διανυσμάτων δοκιμής στο σχεδιαστικό κύκλωμα Α, ο καταχωρητής 2 πρέπει να τεθεί σε κατάσταση ολίσθησης ώστε να ολισθήσουμε εκτός του κυκλώματος την υπογραφή του συνδυαστικού κυκλώματος Α και να την συγκρίνουμε με την αναμενόμενη υπογραφή (βλέπε σχήμα 8.25).



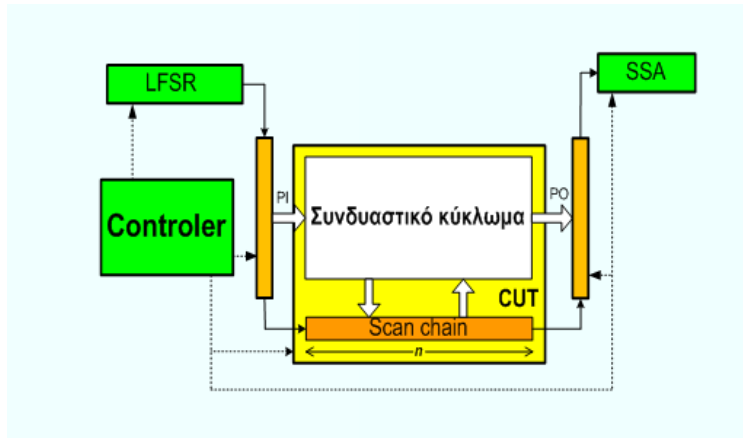
Σχήμα 8.26 Έλεγχος της ορθής λειτουργίας του συνδυαστικού κυκλώματος B.

Για τον έλεγχο της ορθής λειτουργίας του συνδυαστικού κυκλώματος B ο καταχωρητής 2 θα χρησιμοποιηθεί για την παραγωγή διανυσμάτων δοκιμής και ο καταχωρητής 3 ως MISR. Επομένως, πρέπει καταρχήν να θέσουμε τους καταχωρητές 1 και 2 στις επιθυμητές αρχικές καταστάσεις. Για να θέσουμε αρχικές τιμές στους καταχωρητές 1 και 2 θα πρέπει αυτοί να μπορούν να τεθούν και σε λειτουργία ολισθητή.

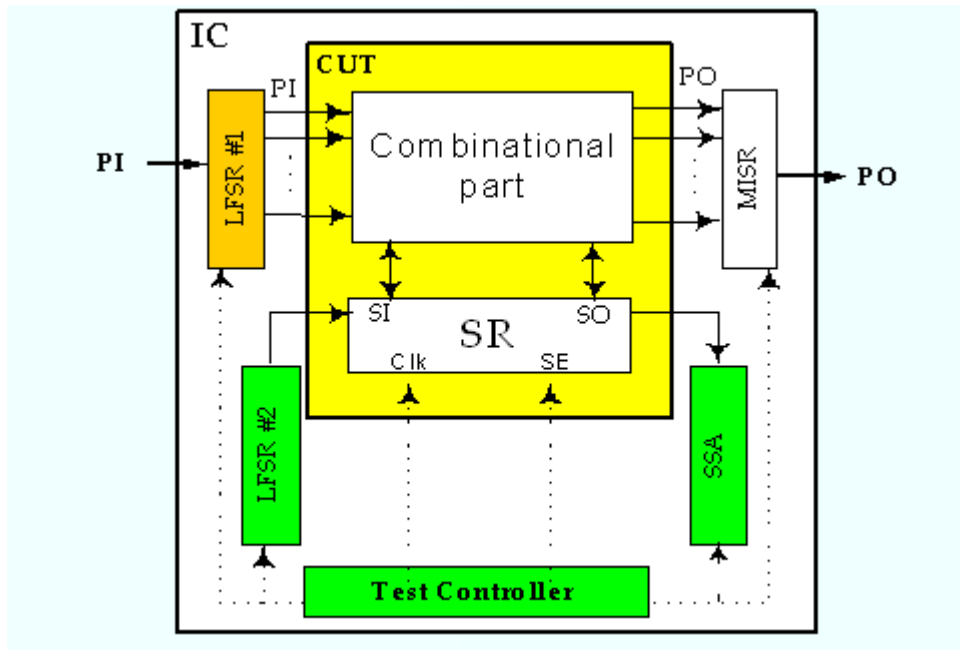
Στη συνέχεια θα πρέπει να θέσουμε τον καταχωρητή 2 σε κατάσταση παραγωγής ψευδοτυχαίων διανυσμάτων δοκιμής και τον καταχωρητή 3 σε λειτουργία MISR. Μετά το τέλος της εφαρμογής των διανυσμάτων δοκιμής στο σχεδιαστικό κύκλωμα B, στις κύριες εξόδους του ολοκληρωμένου κυκλώματος είναι διαθέσιμη η υπογραφή του συνδυαστικού κυκλώματος B, οπότε μπορούμε να την συγκρίνουμε με την αναμενόμενη υπογραφή.

Από την παραπάνω ανάλυση προκύπτει ότι ο καταχωρητής 1 θα πρέπει να έχει δύο τρόπους λειτουργίας, ως κανονικός καταχωρητής παράλληλης φόρτωσης και ως γεννήτρια παραγωγής ψευδοτυχαίων διανυσμάτων. Ο καταχωρητής 2 πρέπει να είναι ένας καταχωρητής BILBO. Ο καταχωρητής 3 πρέπει να έχει επίσης δύο τρόπους λειτουργίας, ως κανονικός καταχωρητής παράλληλης φόρτωσης και ως MISR.

Στην περίπτωση του σχήματος 8.27 στις κύριες εισόδους και τις κύριες εξόδους του κυκλώματος έχουν προστεθεί στοιχεία ολίσθησης, τα οποία σε συνδυασμό με το μονοπάτι ολίσθησης του ακολουθιακού κυκλώματος σχηματίζουν ένα μεγαλύτερου μήκους μονοπάτι ολίσθησης. Ένας ολισθητής γραμμικής ανάδρασης (LFSR) χρησιμοποιείται για την παραγωγή ψευδοτυχαίων δεδομένων τα οποία σειριακά φορτώνονται στο μονοπάτι ολίσθησης και εφαρμόζονται στις κύριες εισόδους και στις ψευδοεισόδους του κυκλώματος. Οι τιμές των ψευδοεξόδων και των εξόδων του ακολουθιακού κυκλώματος φορτώνονται στο μονοπάτι ολίσθησης και σειριακά εισάγονται στον ολισθητή γραμμικής ανάδρασης SSA, ο οποίος χρησιμοποιείται για τη μείωση του όγκου των δεδομένων της απόκρισης.



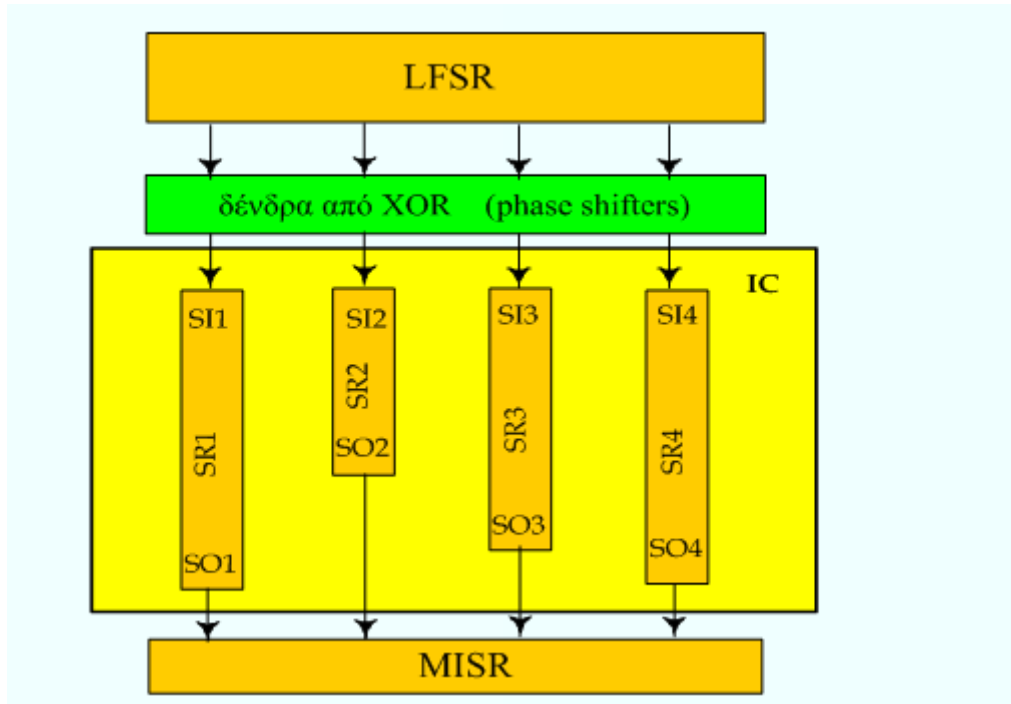
Σχήμα 8.27 Σειριακή παραγωγή των δυαδικών ψηφίων ενός διανύσματος δοκιμής & εφαρμογή ενός διανύσματος δοκιμής σε κ κύκλους του σήματος χρονισμού (*Test Per Scan BIST Scheme*).



Σχήμα 8.28 Συνδυασμός παράλληλης και σειριακής παραγωγής των δυαδικών ψηφίων ενός διανύσματος δοκιμής (*Random Test Socket*).

Όπως έχουμε ήδη αναφέρει στην περίπτωση ενός ακολουθιακού κυκλώματος που ο έλεγχος της ορθής του λειτουργίας βασίζεται στην ύπαρξη μονοπατιού ολίσθησης, κάθε διάνυσμα δοκιμής αποτελείται από δύο τμήματα. Το τμήμα που εφαρμόζεται στις κύριες εισόδους του ακολουθιακού κυκλώματος και το τμήμα που εφαρμόζεται στις ψευδοεισόδους του. Επίσης σ' αυτή την περίπτωση η απόκριση του ακολουθιακού κυκλώματος για κάθε διάνυσμα δοκιμής αποτελείται από δύο τμήματα, το τμήμα που αφορά την απόκριση που εμφανίζεται στις κύριες εξόδους και το τμήμα που εμφανίζεται στις ψευδοεξόδους του ακολουθιακού κυκλώματος.

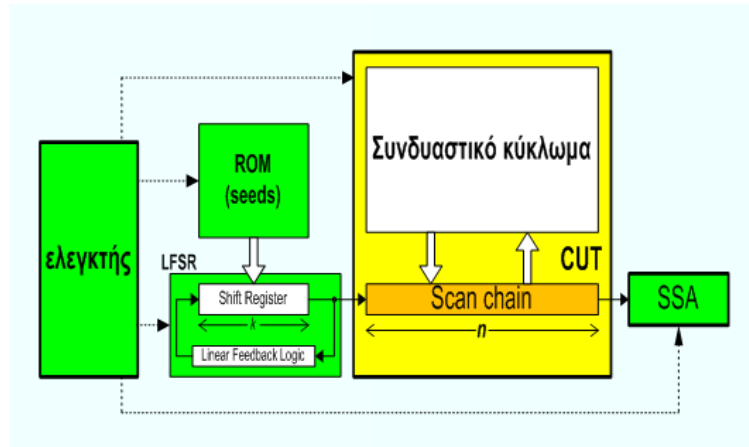
Στο σχήμα 8.28 παρατηρούμε ότι χρησιμοποιούνται διαφορετικοί ολισθητές γραμμικής ανάδρασης για την παραγωγή του τμήματος του διανύσματος δοκιμής που τροφοδοτεί τις κύριες εισόδους και του τμήματος που τροφοδοτεί τις ψευδοεισόδους. Επίσης παρατηρούμε ότι χρησιμοποιούνται διαφορετικοί ολισθητές γραμμικής ανάδρασης για τη μείωση του όγκου των δεδομένων της απόκρισης του τμήματος που αφορά τις κύριες εξόδους και του τμήματος που αφορά τις ψευδοεξόδους.



Σχήμα 8.29 Κατά τμήματα παράλληλη παραγωγή των δυαδικών ψηφίων ενός διανύσματος δοκιμής και εφαρμογή ενός διανύσματος δοκιμής σε κ κύκλους του σήματος χρονισμού (*Multiple Scan Chains*).

Όπως βλέπουμε στο σχήμα 8.29 ένας ολισθητής γραμμικής ανάδρασης χρησιμοποιείται για την παραγωγή ψευδοτυχαίων δυαδικών τιμών οι οποίες φορτώνονται παράλληλα στα μονοπάτια ολίσθησης του ολοκληρωμένου κυκλώματος. Για να αποφευχθούν οι εξαρτήσεις μεταξύ των δεδομένων που φορτώνονται στα μονοπάτια ολίσθησης του ολοκληρωμένου κυκλώματος χρησιμοποιούνται δέντρα από πύλες XOR (phase shifters) που πρέπει να έχουν σχεδιαστεί κατάλληλα.

Όπως είδαμε κατά τον ψευδοτυχαίο τρόπο παραγωγής διανυσμάτων δοκιμής στην αρχή όλα τα διανύσματα που παράγονται ανιχνεύουν κάποια σφάλματα, τα σφάλματα που ανιχνεύονται εύκολα. Μετά όμως από την παραγωγή κάποιων διανυσμάτων δοκιμής παρεμβάλλονται πάρα πολλά διανύσματα τα οποία δεν ανιχνεύουν κανένα σφάλμα. Όταν το σύνολο δοκιμής αποθηκεύεται σε έναν εξωτερικό ελεγκτή τότε είναι απλό να απορρίψουμε εκείνα τα διανύσματα τα οποία δεν ανιχνεύουν σφάλματα. Στην περίπτωση όμως που τα ψευδοτυχαία διανύσματα παράγονται από έναν ενσωματωμένο ολισθητή γραμμικής ανάδρασης, τότε ο μοναδικός τρόπος για να μην εφαρμόσουμε στο υπό έλεγχο κύκλωμα τα διανύσματα που δεν ανιχνεύουν σφάλματα είναι η επαναρχικοποίηση του ολισθητή γραμμικής ανάδρασης. Στο σχήμα 8.30 τα διανύσματα για την επαναρχικοποίηση αποθηκεύονται σε μια μνήμη ROM.

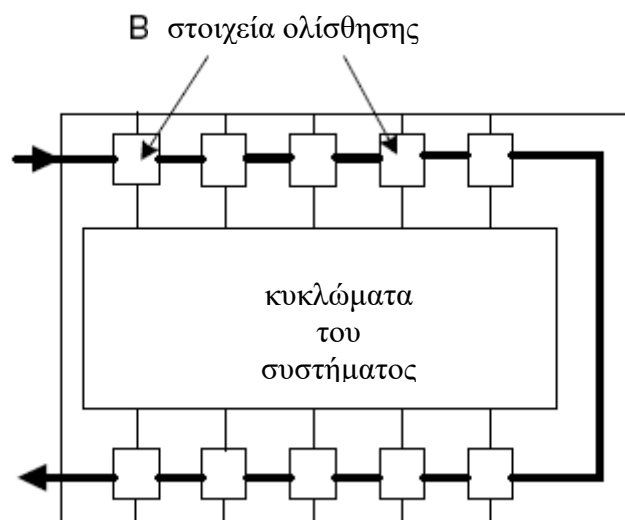


Σχήμα 8.30 Επαναρχικοποίηση του ολισθητή γραμμικής ανάδρασης (Test per scan BIST with reseeding).

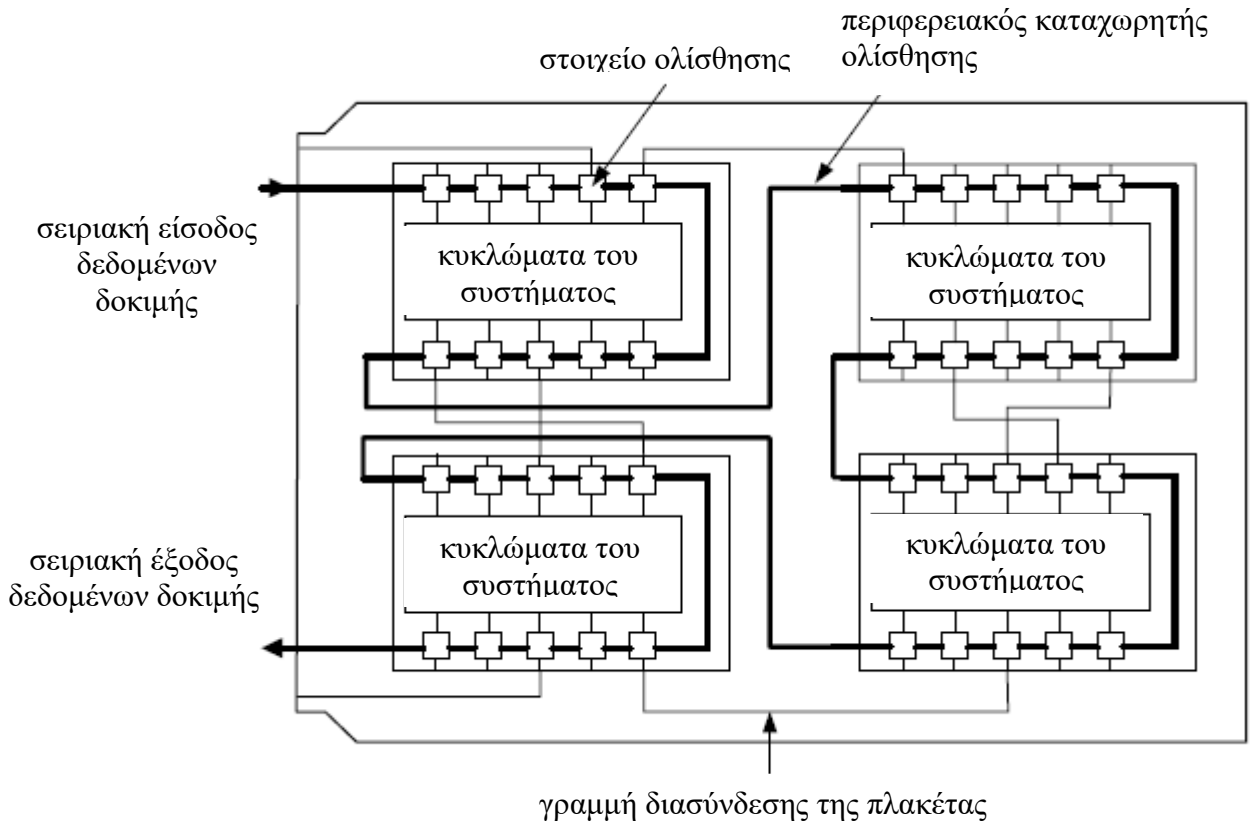
Κεφάλαιο 9. Πρότυπο περιφερειακής ολίσθησης για ψηφιακά κυκλώματα (Digital boundary scan, IEEE Std. 1149.1)

9.1 Βασική ιδέα

Το πρότυπο 1149.1 ορίζει ένα πρωτόκολλο πρόσβασης για έλεγχο της ορθής λειτουργίας και μια αρχιτεκτονική που βασίζεται σε ένα περιφερειακό καταχωρητή ολίσθησης. Ο περιφερειακός καταχωρητής ολίσθησης σχηματίζεται προσθέτοντας ένα στοιχείο ολίσθησης σε κάθε ακροδέκτη εισόδου και εξόδου ενός ολοκληρωμένου κυκλώματος και συνδέοντάς τα με τέτοιο τρόπο ώστε να μπορούν να λειτουργήσουν ως ένας ολισθητής (βλέπε σχήμα 9.1). Όταν χρησιμοποιηθούν σε μια πλακέτα (PCB) ολοκληρωμένα κυκλώματα που υλοποιούν το πρότυπο, τότε οι περιφερειακοί καταχωρητές ολίσθησης των ολοκληρωμένων κυκλωμάτων μπορούν να συνδεθούν με τέτοιο τρόπο ώστε οι ακροδέκτες εισόδου και εξόδου των ολοκληρωμένων κυκλωμάτων να είναι προσπελάσιμες από τις κύριες εισόδους και εξόδους της πλακέτας. Στο σχήμα 9.2 παρατηρούμε ότι οι περιφερειακοί καταχωρητές ολίσθησης τεσσάρων ολοκληρωμένων κυκλωμάτων μιας πλακέτας έχουν συνδεθεί έτσι ώστε να σχηματίσουν ένα ολισθητή.



Σχήμα 9.1 Περιφερειακός καταχωρητής ολίσθησης



Σχήμα 9.2 Πλακέτα με τέσσερα ολοκληρωμένα κυκλώματα που υλοποιούν το πρότυπο περιφερειακής ολίσθησης.

9.2 Αρχιτεκτονική του προτύπου και λειτουργία

Για την υλοποίηση του προτύπου επιπλέον του περιφερειακού καταχωρητή ολίσθησης απαιτούνται πρόσθετα κυκλώματα ελέγχου και αποθήκευσης πληροφορίας. Στο σχήμα 9.3 δίνεται η αρχιτεκτονική του προτύπου περιφερειακής ολίσθησης. Όλα τα κυκλώματα, εκτός από τα κυκλώματα του συστήματος, έχουν προστεθεί για την υλοποίηση του προτύπου. Τα πρόσθετα κυκλώματα μπορούμε να τα ομαδοποιήσουμε σε τέσσερις κύριες συνιστώσες.

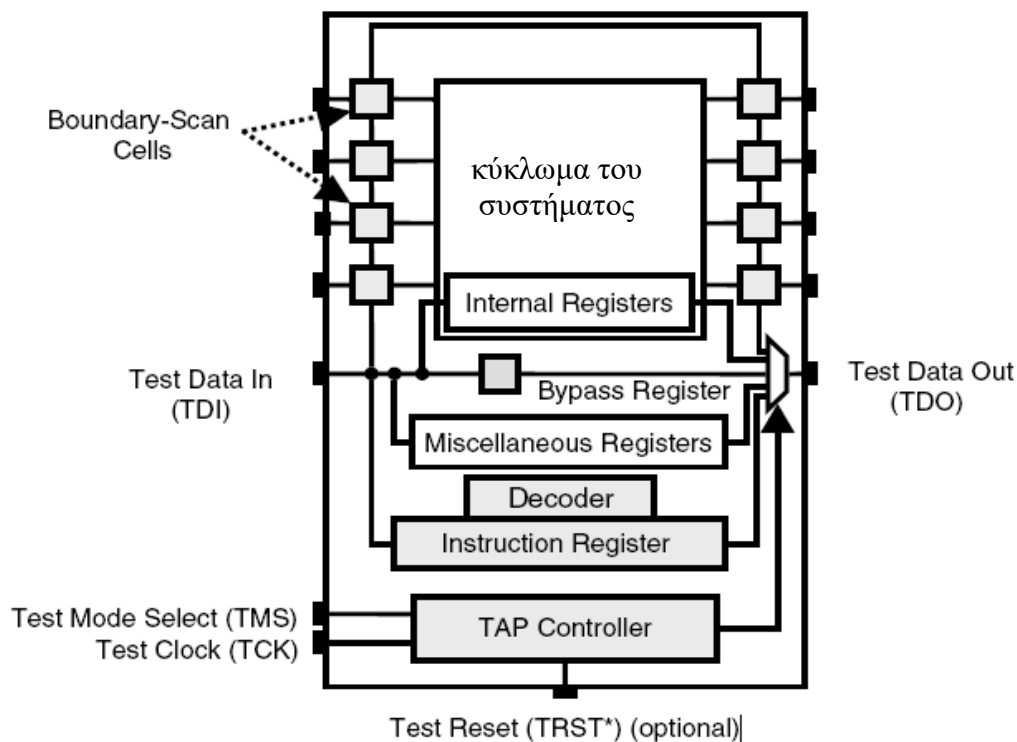
1. Πόρτα εισόδου/εξόδου δεδομένων για έλεγχο ορθής λειτουργίας (Test Access Port, TAP).

Περιλαμβάνει υποχρεωτικά τους επόμενους τέσσερις ακροδέκτες εισόδου/εξόδου :

- test data input, TDI
- test data output, TDO
- test mode select, TMS
- test clock input, TCK

και ένα προαιρετικό,

- test reset, TRST*
2. **Ελεγκτής της πόρτας εισόδου/εξόδου δεδομένων (TAP controller, TAPC)**
Είναι μια μηχανή πεπερασμένων καταστάσεων με 16 καταστάσεις, η οποία ελέγχει κάθε βήμα της διαδικασίας εισόδου δεδομένων δοκιμής, εντολών ελέγχου και εξόδου αποκρίσεων.
 3. **Καταχωρητής εντολής, (Instruction Register, IR) και ο αποκωδικοποιητής της.**
Κάθε εντολή που πρέπει να εκτελεστεί φορτώνεται σειριακά στον καταχωρητή εντολών μέσω του ακροδέκτη εισόδου δεδομένων δοκιμής (TDI).
 4. **Διάφοροι καταχωρητές δεδομένων δοκιμής (miscellaneous registers)**
Υποχρεωτικός είναι ο περιφερειακός καταχωρητής ολίσθησης (boundary scan register) και ο καταχωρητής παράκαμψης (bypass register). Προαιρετικά μπορεί να υπάρχουν πρόσθετοι καταχωρητές, όπως ένας καταχωρητής που να περιέχει τον αριθμό αναγνώρισης του ολοκληρωμένου κυκλώματος (device-ID register) και καταχωρητές που εξαρτώνται από τη συγκεκριμένη σχεδίαση και έχουν σχέση με τον έλεγχο της ορθής λειτουργίας του κυκλώματος.



Σχήμα 9.3 Αρχιτεκτονική του προτύπου. Προς αποφυγή συγχύσεως δεν έχει γίνει μετάφραση στα Ελληνικά των ονομάτων που χρησιμοποιεί το πρότυπο

Κάθε εντολή που πραγματοποιείται από την αρχιτεκτονική boundary scan πρέπει να φορτωθεί σειριακά στον καταχωρητή εντολών - Instruction Register (IR), μέσω ακροδέκτη εισαγωγής δεδομένων δοκιμής (TDI).

Εκτός από τις συνιστώσες υλικού, το πρότυπο ορίζει επίσης ένα σύνολο εντολών, που περιλαμβάνει υποχρεωτικά τις εντολές **BYPASS**, **SAMPLE**, **PRELOAD** και **EXTEST**, και μερικές προαιρετικές όπως οι **INTEST**, **RUNBIST**, **CLAMP**, **IDCODE**, **USERCODE** και **HIGHZ**. Επιτρέπει επίσης στον χρήστη να ορίσει τις δικές του εντολές.

Τα βασικά βήματα της διαδικασίας που ακολουθείται για τον έλεγχο της ορθής λειτουργίας χρησιμοποιώντας το πρότυπο είναι τα ακόλουθα:

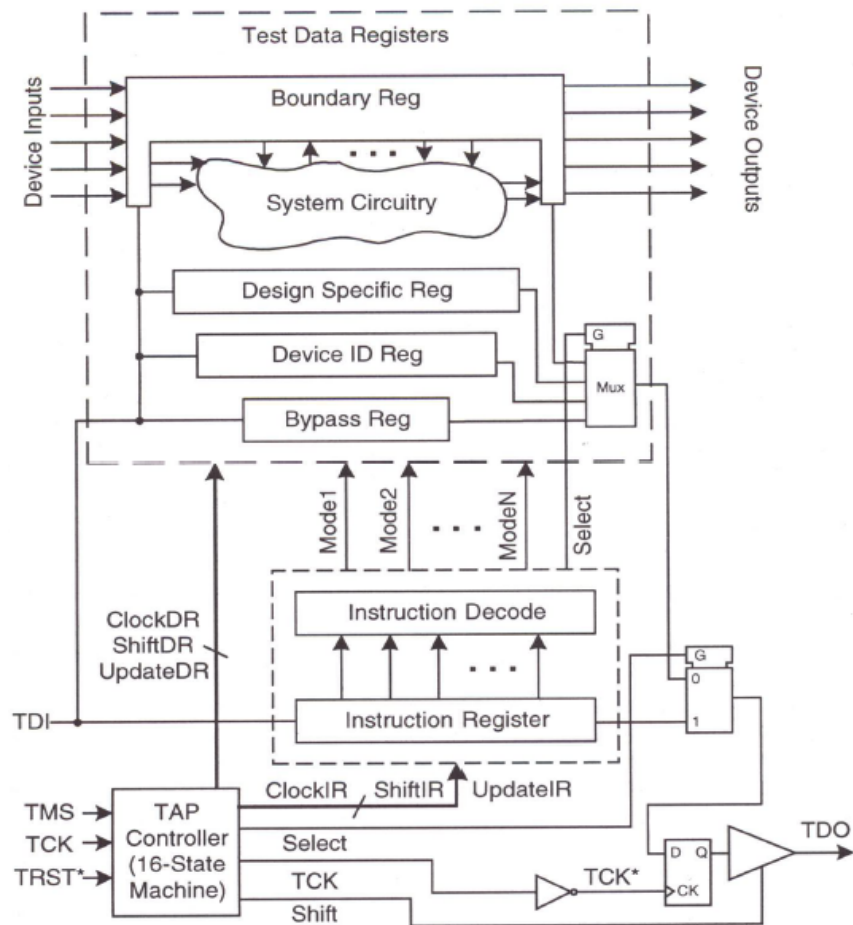
1. Μια εντολή φορτώνεται σειριακά στον καταχωρητή εντολών IR μέσω της εισόδου δεδομένων δοκιμής TDI.
2. Η εντολή αποκωδικοποιείται από τον αποκωδικοποιητή που συνδέεται με τον καταχωρητή εντολών και παράγονται τα απαραίτητα σήματα ελέγχου ώστε να διαμορφωθούν κατάλληλα τα κυκλώματα που υλοποιούν το πρότυπο.
3. Ένα διάνυσμα δοκιμής ολισθαίνει στον επιλεγμένο καταχωρητή δεδομένων δοκιμής μέσω της εισόδου δεδομένων δοκιμής TDI και τροφοδοτεί το προς έλεγχο κύκλωμα.
4. Η απόκριση του υπό έλεγχο κυκλώματος καταγράφεται σε κάποιο καταχωρητή δεδομένων.
5. Η απόκριση μεταφέρεται σειριακά μέσω του ακροδέκτη εξόδου δεδομένων TDO εκτός του ολοκληρωμένου προς παρατήρηση, ενώ ταυτόχρονα είναι δυνατόν ένα νέο διάνυσμα δοκιμής να μεταφέρεται σειριακά στο ολοκληρωμένο μέσω της πόρτας εισόδου δεδομένων TDI.
6. Τα βήματα 3 έως 5 επαναλαμβάνονται έως ότου όλα τα διανύσματα δοκιμής να ολισθήσουν μέσα στο ολοκληρωμένο και να εφαρμοστούν στο υπό έλεγχο κύκλωμα και όλες οι αποκρίσεις του υπό έλεγχο κυκλώματος, οι οποίες αντιστοιχούν στα διανύσματα δοκιμής, να μεταφερθούν εκτός του ολοκληρωμένου κυκλώματος.

Στο σχήμα 9.4 δίνεται το μπλοκ διάγραμμα ενός ολοκληρωμένου κυκλώματος, OK, το οποίο υλοποιεί το πρότυπο.

9.3 Πόρτα εισόδου/εξόδου δεδομένων για έλεγχο ορθής λειτουργίας (Test access port (TAP))

Test clock input (TCK) είναι μια είσοδος σήματος χρονισμού που χρησιμοποιείται για τον συγχρονισμό των διαδικασιών ελέγχου ορθής λειτουργίας μεταξύ των διάφορων μερών ενός ολοκληρωμένου κυκλώματος ή μεταξύ των διαφορετικών

ολοκληρωμένων κυκλωμάτων σε μια πλακέτα (PCB). Αυτή η είσοδος πρέπει να είναι ανεξάρτητη από τα σήματα χρονισμού των μονάδων του συστήματος, των οποίων η συχνότητα μπορεί να διαφέρει σημαντικά από μονάδα σε μονάδα.



Σχήμα 9.4 Ένα μπλοκ διάγραμμα ενός ολοκληρωμένου κυκλώματος, OK, το οποίο υλοποιεί το πρότυπο.

Test data input (TDI). Μέσω αυτής της εισόδου γίνεται σειριακά η μεταφορά εντολών και δεδομένων δοκιμής στον καταχωρητή εντολών και τους καταχωρητές δεδομένων δοκιμής αντίστοιχα. Η τιμή της εισόδου TDI κατά τη μετάβαση του σήματος χρονισμού TCK από 0 σε 1 αποθηκεύεται στον επιλεγμένο καταχωρητή. Η τιμή στην είσοδο TDI μπορεί να αλλάξει κατά τη μετάβαση του σήματος χρονισμού από 1 σε 0. Τα κυκλώματα που οδηγούνται από την είσοδο TDI έχουν σχεδιαστεί με τέτοιο τρόπο ώστε όταν δεν οδηγούνται να είναι σαν να οδηγούνται με τη λογική τιμή 1.

Test data output (TDO). Μέσω αυτής της εξόδου τα περιεχόμενα ενός καταχωρητή δεδομένων δοκιμής του προτύπου μπορούν να μεταφερθούν σειριακά εκτός του ολοκληρωμένου ή και της πλακέτας. Η τιμή του ακροδέκτη TDO μπορεί να αλλάξει κατά τη μετάβαση του σήματος χρονισμού TCK από 1 σε 0. Τα δεδομένα πρέπει να

μεταφέρονται από τον ακροδέκτη εισόδου δεδομένων δοκιμής TDI προς τον ακροδέκτη δεδομένων εξόδου TDO χωρίς να γίνεται αντιστροφή της τιμής τους.

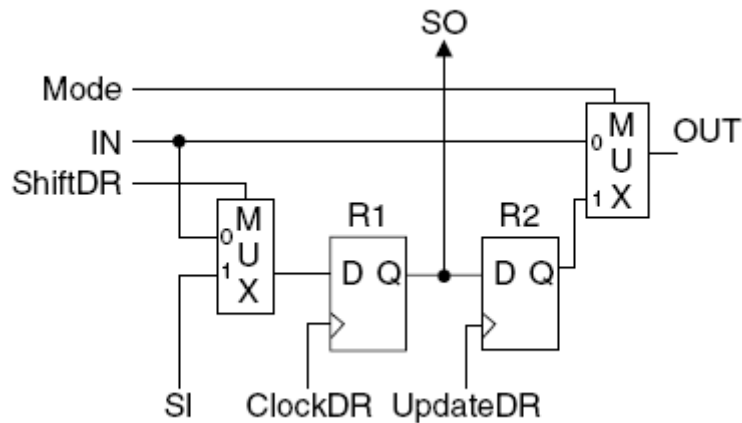
Test mode select (TMS). Είναι η μόνη είσοδος ελέγχου που δέχεται ο ελεγκτής της πόρτας εισόδου/εξόδου δεδομένων δοκιμής (TAP controller). Η ακολουθία τιμών που οδηγεί την είσοδο TMS καθορίζει αν θα γίνει ολίσθηση (shifting) του περιεχομένου κάποιου καταχωρητή δεδομένων δοκιμής, ενημέρωσή του (updating) ή αποθήκευση της απόκρισης του υπό έλεγχο κυκλώματος (capturing) σε κάποιον από τους καταχωρητές δεδομένων δοκιμής. Οι τιμές που εμφανίζονται στην είσοδο TMS διαβάζονται από τον ελεγκτή κατά τη μετάβαση του σήματος χρονισμού από 0 σε 1. Η τιμή του σήματος στην είσοδο TMS μπορεί να αλλάξει κατά τη μετάβαση του σήματος χρονισμού από 1 σε 0. Όταν η είσοδος TMS είναι ανενεργός πρέπει να οδηγείται με τη λογική τιμή 1.

Test reset (TRST*). Είναι μια προαιρετική είσοδος που χρησιμοποιείται για την αρχικοποίηση (reset) του ελεγκτή της πόρτας εισόδου/εξόδου δεδομένων για έλεγχο ορθής λειτουργίας (TAP controller). Όταν οδηγήσουμε την είσοδο TRST* με τη λογική τιμή 0, ο ελεγκτής πηγαίνει στην κατάσταση “Test Logic Reset”. Αυτή η είσοδος δεν πρέπει να χρησιμοποιείται για την αρχικοποίηση της λογικής του συστήματος.

9.4 Καταχωρητές δεδομένων δοκιμής (data registers, DR)

Όπως έχουμε ήδη αναφέρει, σύμφωνα με το πρότυπο είναι υποχρεωτική η ύπαρξη του περιφερειακού καταχωρητή και του καταχωρητή παράκαμψης. Προαιρετικά μπορεί να υπάρχουν και άλλοι καταχωρητές.

Ο περιφερειακός καταχωρητής σχηματίζεται προσθέτοντας ένα στοιχείο ολίσθησης σε κάθε ακροδέκτη εισόδου και εξόδου ενός ολοκληρωμένου κυκλώματος και συνδέοντάς τα με τέτοιο τρόπο ώστε να μπορούν να λειτουργήσουν ως ένας ολισθητής. Έχουν προταθεί διάφοροι σχεδιασμοί για το στοιχείο ολίσθησης που μπορεί να χρησιμοποιηθεί. Στο σχήμα 9.5 δίνεται η δομή ενός τυπικού στοιχείου ολίσθησης. Το ίδιο στοιχείο ολίσθησης μπορεί να τοποθετηθεί σε ακροδέκτη εισόδου ή σε ακροδέκτη εξόδου. Στην περίπτωση που το στοιχείο ολίσθησης τοποθετείται σε μια είσοδο του προς έλεγχο κυκλώματος, η είσοδος IN του στοιχείου ολίσθησης οδηγείται από μια κυρία είσοδο του ολοκληρωμένου κυκλώματος και έξοδος OUT του στοιχείου ολίσθησης οδηγεί μια είσοδο του προς έλεγχο κυκλώματος. Αντίθετα, όταν στο στοιχείο ολίσθησης τοποθετείται σε μια έξοδο του προς έλεγχο κυκλώματος, τότε η είσοδος IN οδηγείται από μια έξοδο του προς έλεγχο κυκλώματος, ενώ η έξοδος OUT του στοιχείου ολίσθησης οδηγεί μια κύρια έξοδο του ολοκληρωμένου κυκλώματος.



Σχήμα 9.5 Δομή ενός τυπικού στοιχείου ολίσθησης

Κατά τη διάρκεια της κανονικής λειτουργίας ($Mode = 0$), τα δεδομένα περνάνε κατευθείαν από την είσοδο IN στην έξοδο OUT του στοιχείου ολίσθησης. Κατά τη διάρκεια ελέγχου της ορθής λειτουργίας ($Mode = 1$), ο τρόπος λειτουργίας του στοιχείου ολίσθησης εξαρτάται από τις τιμές των τριών σημάτων ελέγχου ClockDR, ShiftDR, και UpdateDR τα οποία παράγονται από τον ελεγκτή της πόρτας εισόδου/εξόδου δεδομένων ελέγχου ορθής λειτουργίας (TAP controller).

Αποθήκευση της εξόδου του υπό έλεγχο κυκλώματος στη βαθμίδα ολίσθησης την οποία οδηγεί (Capture). Την διαδικασία αυτή στη συνέχεια θα την καλούμε αποθήκευση κατάστασης.

Η είσοδος ShiftDR τίθεται στη λογική τιμή 0, τροφοδοτούμε με ένα παλμό ρολογιού στην είσοδο ClockDR, οπότε η τιμή της εισόδου IN αποθηκεύεται στο φλιπ-φλοπ R1.

Ολίσθηση

Θα πρέπει κατ' αρχήν να επισημάνουμε ότι η είσοδος SI ενός στοιχείου ολίσθησης οδηγείται από την έξοδο SO του προηγούμενου στοιχείου ολίσθησης και ούτω καθεξής. Η είσοδος ShiftDR τίθεται στη λογική τιμή 1 και τροφοδοτούμε με παλμούς ρολογιού την είσοδο ClockDR. Τότε τα δυαδικά ψηφία του διανύσματος δοκιμής ολισθαίνουν απ' έξω προς τα μέσα διαμέσου της εισόδου SI και τα δυαδικά ψηφία της απόκρισης ολισθαίνουν προς τα έξω μέσω της εξόδου SO.

Ενημέρωση

Εφαρμόζοντας έναν παλμό ρολογιού στην είσοδο UpdateDR, η τιμή που είναι αποθηκευμένη στο φλιπ-φλοπ R1 αποθηκεύεται στο φλιπ-φλοπ R2. Εάν είμαστε σε κατάσταση ελέγχου ορθής λειτουργίας ($Mode = 1$) η έξοδος του R2 οδηγεί την έξοδο OUT.

Είναι πολύ εύκολο να δούμε ότι η αποθήκευση κατάστασης και η ολίσθηση μπορεί να γίνει και κατά την κανονική λειτουργία του κυκλώματος, δηλαδή όταν $Mode = 0$.

Πρέπει να σημειώσουμε ότι τα φλιπ-φλοπ R1 (σχήμα 9.5) των στοιχείων ολίσθησης του περιφερειακού καταχωρητή σχηματίζουν ένα ολισθητή, την οποία θα καλούμε βαθμίδα ολίσθησης του περιφερειακού καταχωρητή. Τα φλιπ-φλοπ R2 (σχήμα 9.5) των στοιχείων ολίσθησης του περιφερειακού καταχωρητή σχηματίζουν ένα καταχωρητή παράλληλης εισόδου - παράλληλης εξόδου την οποία θα καλούμε βαθμίδα ενημέρωσης του περιφερειακού καταχωρητή. Δηλαδή ο περιφερειακός καταχωρητής ολίσθησης είναι δύο βαθμίδων.

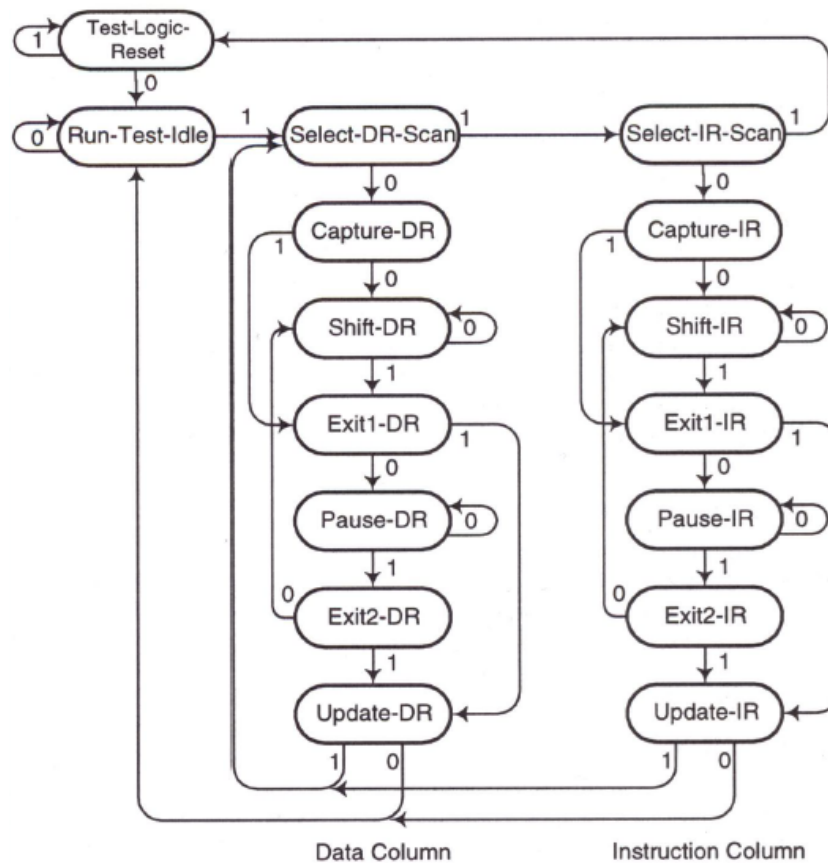
Το πρότυπο IEEE 1149.1-2001 ορίζει δέκα διαφορετικά στοιχεία ολίσθησης που μπορούν να χρησιμοποιηθούν για την υλοποίηση του περιφερειακού καταχωρητή ολίσθησης.

Ο καταχωρητής παράκαμψης (bypass register), είναι ένας καταχωρητής του ενός δυαδικού ψηφίου ο οποίος χρησιμοποιείται για να παρακάμψει ένα ολοκληρωμένο κύκλωμα όταν δεν περιλαμβάνεται στην τρέχουσα διαδικασία ελέγχου ορθής λειτουργίας. Η χρησιμοποίησή του μπορεί να μειώσει σημαντικά το χρόνο που απαιτείται για την ολίσθηση μέσα και έξω των δεδομένων δοκιμής μέσω της μεγάλης μήκους διαδρομής TDI-TDO.

Το πρότυπο 1149.1 καθορίζει επίσης έναν προαιρετικό καταχωρητή, τον αποκαλούμενο καταχωρητή ταυτότητας συσκευής (ID Register), ο οποίος μπορεί να χρησιμοποιηθεί για να υποθηκεύσουμε πληροφορίες για το προϊόν (κατασκευαστή, αριθμό προϊόντος, και αριθμό έκδοσης) ή για τη διαμόρφωση του ολοκληρωμένου κυκλώματος. Οι αποθηκευμένες πληροφορίες μπορούν να διαβαστούν ενώ το ολοκληρωμένο κύκλωμα έχει τοποθετηθεί σε κάποια πλακέτα.

9.5 Ελεγκτής της πόρτας εισόδου/εξόδου δεδομένων για έλεγχο ορθής λειτουργίας (*Test Access Port Controller, TAPC*)

Είναι μια μηχανή πεπερασμένων καταστάσεων (16 καταστάσεων) που λειτουργεί σύμφωνα με το διάγραμμα καταστάσεων που φαίνεται στο σχήμα 9.6. Ο ελεγκτής μπορεί να αλλάξει κατάσταση μόνο κατά τη μετάβαση του σήματος χρονισμού TCK από 0 σε 1. Η επόμενη κατάσταση καθορίζεται από τη λογική τιμή της εισόδου TMS (Test Mode Select).



Σχήμα 9.6 Διάγραμμα καταστάσεων του ελεγκτή της πόρτας εισόδου/εξόδου δεδομένων για έλεγχο ορθής λειτουργίας (Test Access Port Controller, TAPC)

Τα σήματα εξόδου του ελεγκτή καθορίζουν το βήμα της διαδικασίας ελέγχου της ορθής λειτουργίας που θα ακολουθήσει. Ο ελεγκτής παράγει τα ακόλουθα εννέα σήματα ελέγχου: ClockDR, ShiftDR, UpdateDR, ClockIR, ShiftIR, UpdateIR, Select, TCK, Shift και Reset (προαιρετικό). Οι κύριες λειτουργίες του ελεγκτή περιλαμβάνουν:

- Αρχικοποίηση (reset) των κυκλωμάτων που υλοποιούν το πρότυπο.
- Παραγωγή των σημάτων ελέγχου που απαιτούνται για να φορτωθεί μια εντολή στον καταχωρητή εντολών.
- Παραγωγή των σημάτων που απαιτούνται για να εκτελεστούν οι λειτουργίες αποθήκευσης κατάστασης (Capture) και ενημέρωσης (Update)
- Παραγωγή των σημάτων ελέγχου για την ολίσθηση δεδομένων δοκιμής από την είσοδο TDI στην έξοδο TDO.

Οι 16 καταστάσεις μπορούν να διαιρεθούν σε τρία μέρη. Το πρώτο μέρος (οι 2 καταστάσεις αριστερά) περιέχει την αρχικοποίηση (Test-Logic-Reset) και την “Run-Test/Idle”, το δεύτερο μέρος (οι 7 καταστάσεις στη μέση) και το τρίτο (οι 7 καταστάσεις δεξιά) ελέγχουν τη λειτουργία των καταχωρητών δεδομένων δοκιμής και εντολών, αντίστοιχα. Επειδή η μόνη διαφορά μεταξύ του δεύτερου και του τρίτου

μέρους είναι οι καταχωρητές που εξετάζουν, στη συνέχεια θα περιγράψουμε μόνο τις καταστάσεις των πρώτων 2 μερών.

- **Κατάσταση Test-Logic-Reset.** Σ' αυτή την κατάσταση το σύστημα βρίσκεται στην κανονική του λειτουργία. Υπάρχουν δύο τρόποι για να πάει ο ελεγκτής σ' αυτή την κατάσταση. Ο ένας είναι ασύγχρονος και επιτυγχάνεται οδηγώντας την είσοδο TRST* με τη λογική τιμή 0. Ο άλλος τρόπος είναι σύγχρονος και επιτυγχάνεται, από οποιαδήποτε κατάσταση και αν βρίσκεται ο ελεγκτής, εάν η είσοδο TMS έχει τη λογική τιμή ένα κατά τη μετάβαση του σήματος χρονισμού TCK από 0 σε 1 πέντε συνεχόμενες φορές.
Εάν για κάποιο λόγο η είσοδος TMS λάβει την τιμή 0, ο ελεγκτής θα πάει στην κατάσταση "Run-Test/Idle". Όμως θα επιστρέψει στην κατάσταση "Test-Logic-Reset" εάν η είσοδος TMS μείνει σταθερά στην λογική τιμή 1 για τις επόμενες τρεις χρονικές περιόδους του σήματος χρονισμού TCK.
- **Κατάσταση Run-Test/Idle.** Σ' αυτή την κατάσταση, τα κυκλώματα που υλοποιούν το πρότυπο περιμένουν να ολοκληρωθεί κάποια διαδικασία ελέγχου ορθής λειτουργίας (όπως BIST). Η κατάσταση αυτή είναι διαφορετική από την κατάσταση "Test-Logic-Reset", καθότι κατά τη διάρκειά της μπορεί να είναι σε εξέλιξη κάποιες ενέργειες που αφορούν τον έλεγχο ορθής λειτουργίας.
- **Κατάσταση Select-DR-Scan.** Αυτή είναι μια προσωρινή κατάσταση προετοιμασίας για την είσοδο στην στήλη χειρισμού καταχωρητών δεδομένων δοκιμής (του 2ου μέρους του σχήματος 9.6).
- **Κατάσταση Capture-DR.** Στην κατάσταση αυτή, στους καταχωρητές που επιλέγονται από την τρέχουσα εντολή αποθηκεύεται παράλληλα η τρέχουσα απόκριση του υπό έλεγχο κυκλώματος, ή ένα στιγμιότυπο της κανονικής του λειτουργίας.
- **Κατάσταση Shift-DR.** Στην κατάσταση αυτή, στους καταχωρητές που επιλέγονται από την τρέχουσα εντολή αποθηκεύονται σειριακά δεδομένα δοκιμής. Για όσο χρονικό διάστημα η είσοδος TMS έχει τη λογική τιμή 0, ο ελεγκτής θα μείνει σ' αυτή την κατάσταση. Για κάθε κύκλο ρολογιού, ένα δυαδικό ψηφίο των δεδομένων δοκιμής θα ολισθαίνει μέσα στον (και έξω από τον) επιλεγμένο καταχωρητή δεδομένων δοκιμής μέσω της εισόδου TDI (εξόδου TDO).
- **Κατάσταση Exit1-DR.** Και αυτή είναι μια προσωρινή κατάσταση. Όλα τα δεδομένα που φορτώθηκαν παράλληλα (από την Capture-DR κατάσταση) ή σειριακά (από την Shift-DR κατάσταση) κρατιούνται στον επιλεγμένο καταχωρητή δεδομένων δοκιμής αναμένοντας την εισαγωγή στην κατάσταση update ή pause.

- **Κατάσταση Pause-DR.** Τα κυκλώματα που υλοποιούν το πρότυπο μπαίνουν σε μια κατάσταση αναμονής περιμένοντας μερικές εξωτερικές λειτουργίες. Ως παράδειγμα, μπορούμε να αναφέρουμε την περίπτωση στην οποία ένας μεγάλος όγκος από δεδομένα δοκιμής πρέπει να φορτωθεί στο υπό έλεγχο ολοκληρωμένο κύκλωμα. Τότε ο εξωτερικός ελεγκτής (ATE) μπορεί να χρειαστεί κατά διαστήματα να φορτώνει δεδομένα από το δίσκο στη μνήμη του. Η κατάσταση Pause-DR επιτρέπει στα κυκλώματα που υλοποιούν το πρότυπο να περιμένουν έως ότου και άλλα δεδομένα δοκιμής να είναι διαθέσιμα για να φορτωθούν σειριακά στο υπό έλεγχο κύκλωμα.
- **Κατάσταση Exit2-DR.** Η κατάσταση αυτή είτε δηλώνει την ολοκλήρωση της τρέχουσας λειτουργίας αποθήκευσης κατάστασης (capturing) ή της λειτουργίας ολίσθησης (shifting) και επιτρέπει στον ελεγκτή να μπει στην κατάσταση update, είτε αντιπροσωπεύει το τέλος της λειτουργίας Pause-DR, επιτρέποντας στον ελεγκτή να πάει πίσω στην κατάσταση Shift-DR για να μεταφερθούν σειριακά προς τα μέσα ή προς τα έξω περισσότερα δεδομένα δοκιμής.
- **Κατάσταση Update-DR.** Σ' αυτή την κατάσταση δεδομένα δοκιμής που μεταφέρθηκαν σειριακά στους επιλεγμένους καταχωρητές δεδομένων δοκιμής φορτώνονται στις παράλληλες εξόδους τους. Για παράδειγμα τα δεδομένα δοκιμής που είναι αποθηκευμένα στην πρώτη βαθμίδα ολίσθησης (φλιπ-φλοπ R1 στο σχήμα 9.5) του περιφερειακού καταχωρητή αποθηκεύονται στη βαθμίδα ενημέρωσης (φλιπ-φλοπ R2 στο σχήμα 9.5). Σημειώστε ότι, με ένα καταχωρητή δύο βαθμίδων τα δεδομένα δοκιμής μπορούν να κρατηθούν στην παράλληλη έξοδο του επιλεγμένου καταχωρητή ενώ άλλα δεδομένα φορτώνονται σειριακά.

9.6 Καταχωρητής Εντολών και Σύνολο Εντολών

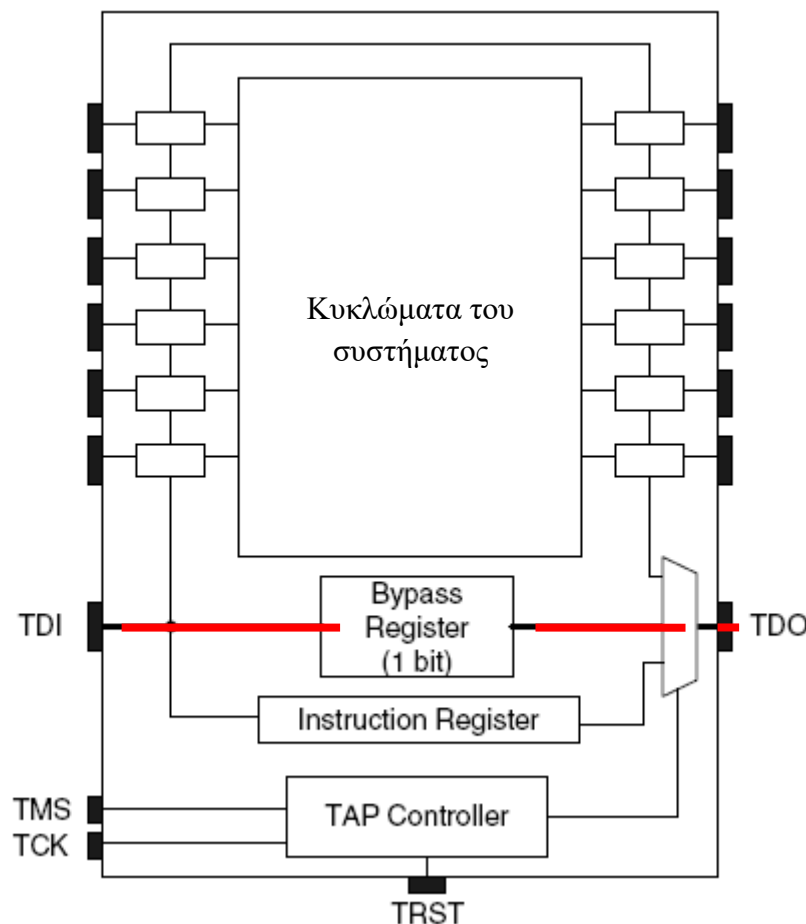
Ο καταχωρητής εντολών χρησιμοποιείται για την αποθήκευση της εντολής που θα εκτελεστεί. Σύμφωνα με το πρότυπο, ο καταχωρητής αυτός πρέπει να είναι δύο βαθμίδων, ώστε όταν φορτώνεται σειριακά μια νέα εντολή, να μην επηρεάζονται οι έξοδοι του καταχωρητή εντολής και επομένως και οι έξοδοι του αποκωδικοποιητή εντολής.

Σύμφωνα με το πρότυπο τέσσερις εντολές είναι υποχρεωτικές (SAMPLE, PRELOAD, BYPASS και EXTEST). Μια ακόμη εντολή συστήνεται, αν και δεν είναι υποχρεωτική, η εντολή INTEST. Υπάρχουν όμως και άλλες χρήσιμες εντολές, όπως οι εντολές RUNBIST, CLAMP, IDCODE, USRCODE και HIGHZ. Μπορούμε να διακρίνουμε τις παραπάνω εντολές σε δύο κατηγορίες. Στη μία κατηγορία ανήκουν οι εντολές που δεν επηρεάζουν την κανονική λειτουργία των κυκλωμάτων του συστήματος και επομένως μπορούν να εκτελεστούν παράλληλα με την κανονική λειτουργία των κυκλωμάτων του συστήματος. Στην άλλη κατηγορία ανήκουν οι εντολές που εκτελούνται όταν το σύστημα δεν είναι σε κατάσταση κανονικής

λειτουργίας. Στη συνέχεια θα περιγράψουμε τις εντολές καθεμιάς από τις δύο κατηγορίες.

Εντολές που δεν επηρεάζουν την κανονική λειτουργία του συστήματος.

- **BYPASS** Η εντολή αυτή τοποθετεί μεταξύ της σειριακής εισόδου δεδομένων δοκιμής TDI και της σειριακής εξόδου δεδομένων δοκιμής TDO τον καταχωρητή παράκαμψης (bypass register), ο οποίος είναι του ενός δυαδικού ψηφίου. Κατά τον έλεγχο της ορθής λειτουργίας μιας πλακέτας συχνά απαιτείται να στείλουμε δεδομένα δοκιμής ή να διαβάσουμε τις αποκρίσεις μόνον από ένα μικρό αριθμό συγκεκριμένων ολοκληρωμένων κυκλωμάτων της πλακέτας. Επομένως η εντολή BYPASS μας δίνει τη δυνατότητα να παρακάμψουμε τους περιφερειακούς καταχωρητές ολίσθησης των ολοκληρωμένων κυκλωμάτων που δεν θέλουμε να οδηγήσουμε με δεδομένα δοκιμής ή να διαβάσουμε τις αποκρίσεις τους. Όταν εκτελείται αυτή η εντολή τότε όταν ο ελεγκτής βρίσκεται στην κατάσταση Capture-DR στον καταχωρητή παράκαμψης αποθηκεύεται η λογική τιμή 0. Στην περίπτωση που δεν υλοποιείται η εντολή IDCODE, κατά την αρχικοποίηση του ελεγκτή στον καταχωρητή εντολής φορτώνεται ο κωδικός της εντολής BYPASS.

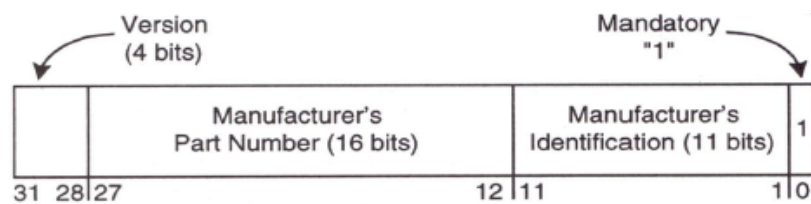


Σχήμα 9.7 Εκτέλεση της εντολής BYPASS

- **IDCODE** Η εντολή αυτή τοποθετεί μεταξύ της σειριακής εισόδου δεδομένων δοκιμής TDI και της σειριακής εξόδου δεδομένων δοκιμής TDO τον καταχωρητή αναγνώρισης συσκευής (device-ID Reg.). Αυτή η εντολή υλοποιείται μόνον όταν έχει υλοποιηθεί ο προαιρετικός καταχωρητής αναγνώρισης συσκευής. Όταν υλοποιείται αυτή η εντολή τότε κατά την αρχικοποίηση του ελεγκτή (κατάσταση Test-Logic-Reset) στον καταχωρητή εντολής φορτώνεται ο κωδικός της εντολής IDCODE.

Κατά το πέρασμα του ελεγκτή από την κατάσταση Capture-DR, στον καταχωρητή αναγνώρισης συσκευής αποθηκεύεται το περιεχόμενο που φαίνεται στο σχήμα 9.8.

- **USERCODE** Και αυτή η εντολή τοποθετεί μεταξύ της σειριακής εισόδου δεδομένων δοκιμής TDI και της σειριακής εξόδου δεδομένων δοκιμής TDO τον καταχωρητή αναγνώρισης συσκευής (device-ID Reg.). Όμως κατά το πέρασμα του ελεγκτή από την κατάσταση Capture-DR στον καταχωρητή αναγνώρισης συσκευής

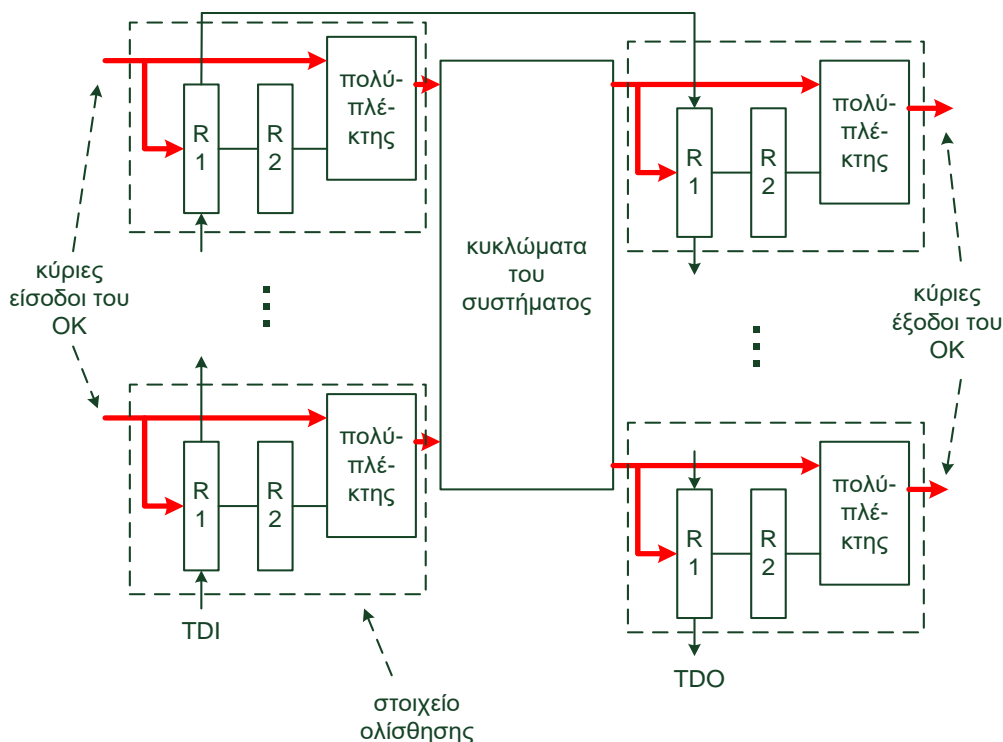


Σχήμα 9.8 Πληροφορία που αποθηκεύεται στον καταχωρητή αναγνώρισης συσκευής (device-ID register)

αποθηκεύεται μια τιμή που ορίζεται από το χρήστη. Η εντολή αυτή είναι προαιρετική, εάν όμως υλοποιείται τότε θα πρέπει να υλοποιείται και η εντολή IDCODE. Στην περίπτωση προγραμματιζόμενων ολοκληρωμένων κυκλωμάτων η εντολή IDCODE δεν αρκεί για την αναγνώριση του ολοκληρωμένου κυκλώματος και του προγραμματισμού του. Η εντολή IDCODE δίνει την πληροφορία ότι το ολοκληρωμένο κύκλωμα ήταν προγραμματιζόμενο, αλλά επειδή ο προγραμματισμός λαμβάνει χώρα μετά την κατασκευή του ολοκληρωμένου κυκλώματος (ή της πλακέτας ή του συστήματος) η εντολή USERCODE μπορεί να χρησιμοποιηθεί για να αναγνωριστεί το είδος του προγραμματισμού που έγινε. Ο χρήστης είναι ελεύθερος να ορίσει μια τιμή των 32 δυαδικών ψηφίων, η οποία μπορεί να αποτελείται από διάφορα πεδία.

- **SAMPLE** Σύμφωνα με το πρότυπο η εντολή αυτή είναι υποχρεωτική, αλλά ο συνδυασμός δυαδικών τιμών 0 και 1 που αποτελούν τον κωδικό της δεν καθορίζεται από το πρότυπο. Αυτή η εντολή τοποθετεί μεταξύ της σειριακής εισόδου δεδομένων δοκιμής TDI και της σειριακής εξόδου δεδομένων δοκιμής TDO τον περιφερειακό καταχωρητή ολίσθησης. Ωστόσο δεν αποσυνδέει τα

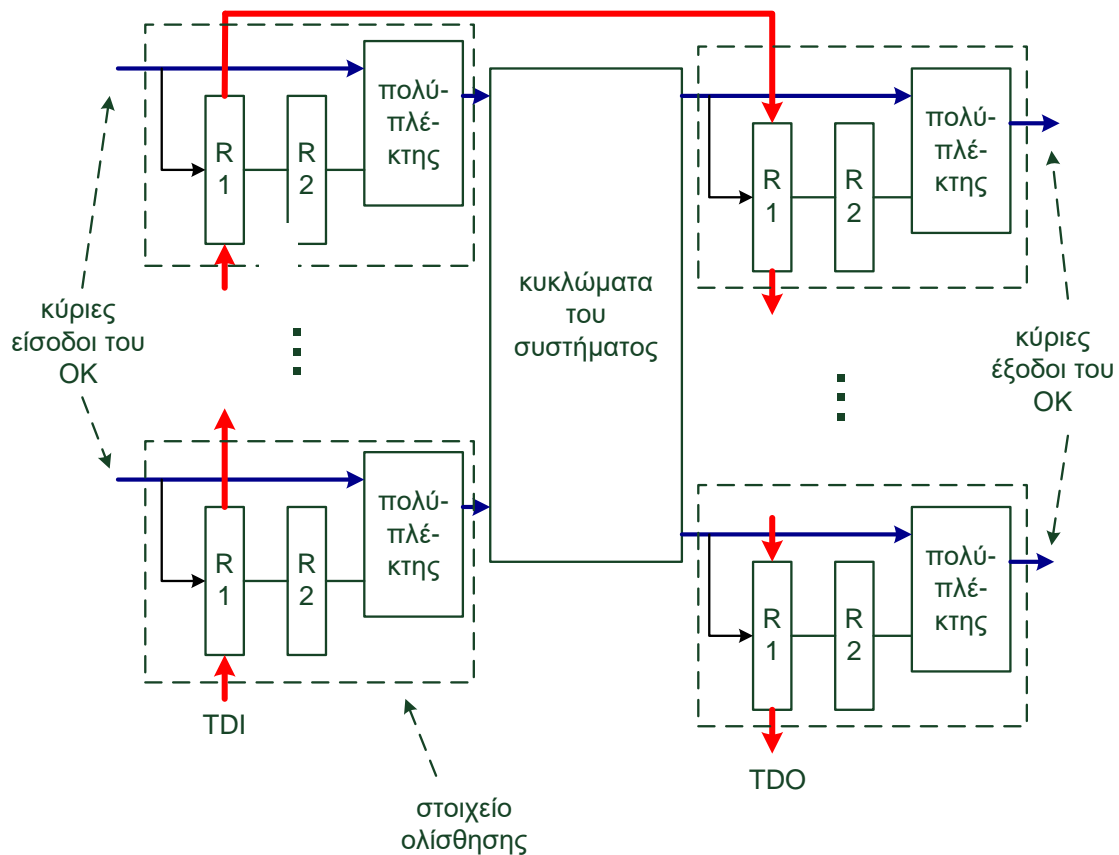
κυκλώματα του συστήματος από τους ακροδέκτες του ολοκληρωμένου κυκλώματος. Όταν ο ελεγκτής βρίσκεται στην κατάσταση Capture-DR, οι τιμές που οδηγούν τις εισόδους του κυκλώματος του συστήματος και οι τιμές των εξόδων του κυκλώματος του συστήματος αποθηκεύονται στη βαθμίδα ολίσθησης του περιφερειακού καταχωρητή ολίσθησης. Με άλλα λόγια στη βαθμίδα ολίσθησης αποθηκεύεται ένα στιγμιότυπο των τιμών των εισόδων και των εξόδων του κυκλώματος του συστήματος (βλέπε σχήμα 9.9). Το περιεχόμενο της βαθμίδας ολίσθησης μπορούμε να το μεταφέρουμε σειριακά έξω προς εξέταση.



Σχήμα 9.9 Εκτέλεση της εντολής SAMPLE

- PRELOAD** Σύμφωνα με το πρότυπο η εντολή αυτή είναι υποχρεωτική, αλλά ο συνδυασμός δυαδικών τιμών 0 και 1 που αποτελούν τον κωδικό της δεν καθορίζεται από το πρότυπο. Αυτή η εντολή τοποθετεί μεταξύ της σειριακής εισόδου δεδομένων δοκιμής TDI και της σειριακής εξόδου δεδομένων δοκιμής TDO τον περιφερειακό καταχωρητή ολίσθησης. Ωστόσο δεν αποσυνδέει τα κυκλώματα του συστήματος από τους ακροδέκτες του ολοκληρωμένου κυκλώματος. Η εντολή PRELOAD επιτρέπει δεδομένα δοκιμής να ολισθήσουν μέσα και έξω από τον περιφερειακό καταχωρητή ολίσθησης όταν ο ελεγκτής είναι στην κατάσταση Shift-DR χωρίς να επηρεάζεται η κανονική λειτουργία των κυκλωμάτων του συστήματος (βλέπε σχήμα 9.10). Όταν ο ελεγκτής είναι στην κατάσταση Update-DR, κατά τη μετάβαση του σήματος χρονισμού TCK από 0

σε 1 τα δεδομένα δοκιμής που φορτώθηκαν σειριακά στη βαθμίδα ολίσθησης (φλιπ-φλοπ R1) του περιφερειακού καταχωρητή ολίσθησης αποθηκεύονται παράλληλα στη βαθμίδα ενημέρωσης (φλιπ-φλοπ R2).



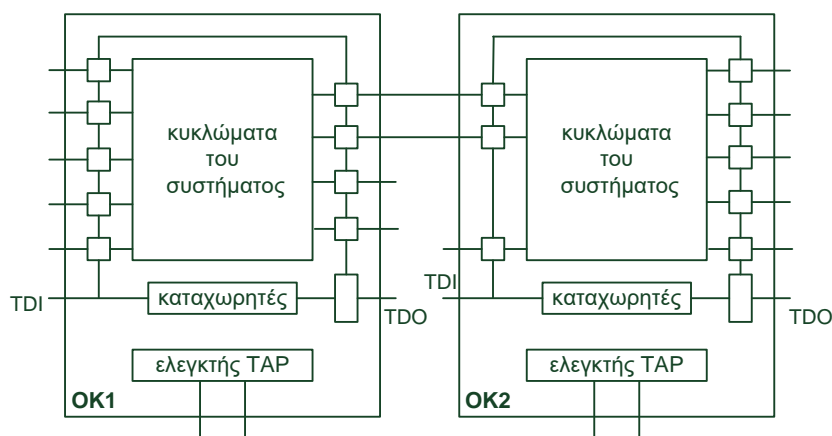
Σχήμα 9.10 Εκτέλεση της εντολής PRELOAD

Εντολές που εκτελούνται σε κατάσταση ελέγχου ορθής λειτουργίας.

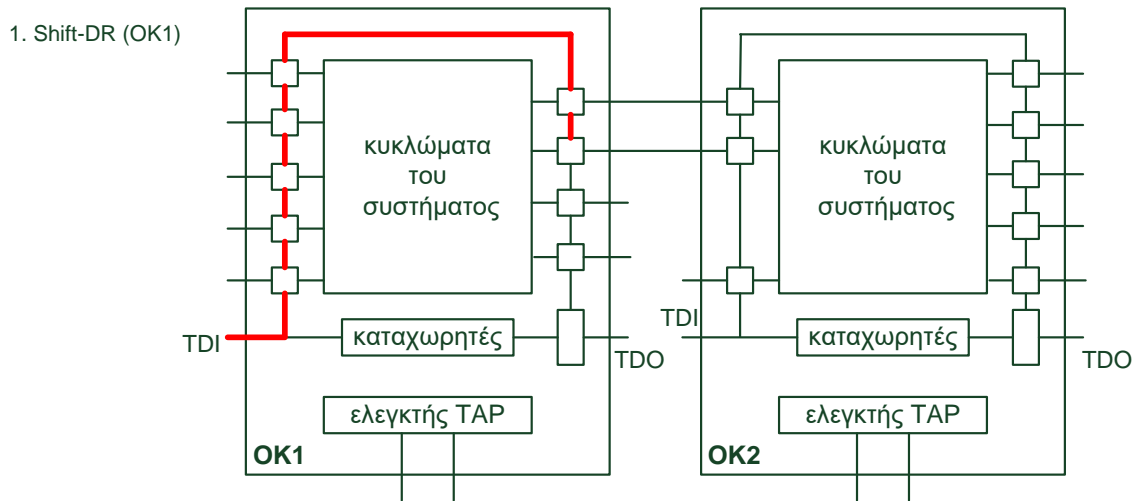
- **EXTEST** Σύμφωνα με το πρότυπο η εντολή αυτή είναι υποχρεωτική, αλλά ο συνδυασμός δυαδικών τιμών 0 και 1 που αποτελούν τον κωδικό της δεν καθορίζεται από το πρότυπο, επιλέγεται από το σχεδιαστή. Η εντολή αυτή χρησιμοποιείται για τον έλεγχο της ορθής λειτουργίας κυκλωμάτων που δεν υλοποιούν το πρότυπο και τις γραμμές διασύνδεσης μεταξύ των

ολοκληρωμένων κυκλωμάτων της πλακέτας και τις γραμμές διασύνδεσης μεταξύ των πλακετών.

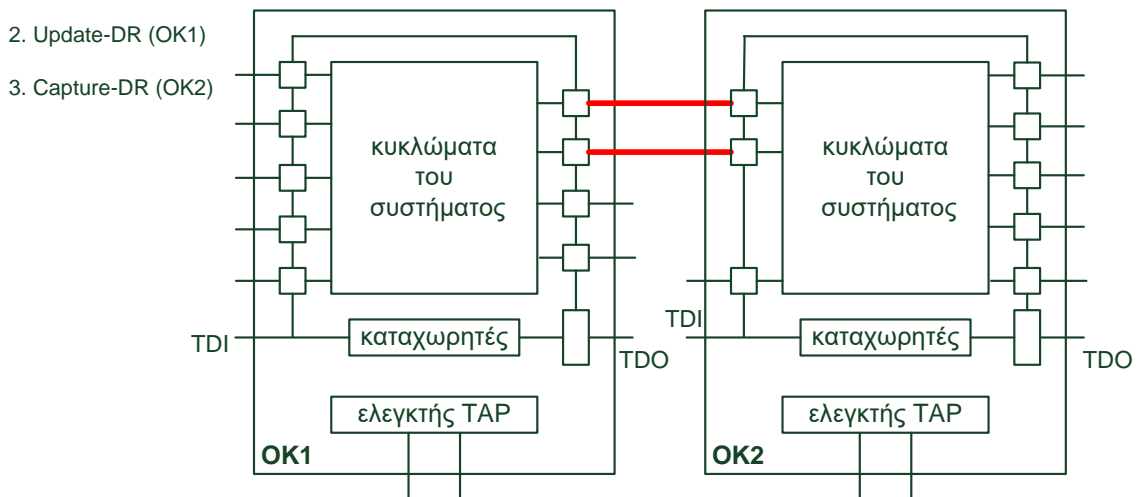
Η λειτουργία της διευκρινίζεται στο σχήμα 9.11. Υποθέτουμε ότι θέλουμε να ελέγξουμε την ορθή λειτουργία των γραμμών διασύνδεσης που οδηγούνται από δύο εξόδους ενός ολοκληρωμένου κυκλώματος, OK1, και οδηγούν δύο εισόδους ενός άλλου ολοκληρωμένου κυκλώματος, OK2 (βλέπε σχήμα 9.11.α). Κατ' αρχήν ενώ ελεγκτής του OK1 είναι στην κατάσταση Shift-DR το διανύσματα δοκιμής μεταφέρεται σειριακά, διαμέσου της σειριακής εισόδου δεδομένων δοκιμής TDI, στα φλιπ-φλοπ της βαθμίδα ολίσθησης του περιφερειακού καταχωρητή, τα οποία αντιστοιχούν στις συγκεκριμένες δύο εξόδους του OK1(βλέπε σχήμα 9.11.β). Στη συνέχεια ο ελεγκτής πηγαίνει στην κατάσταση Update-DR, οπότε τα δεδομένα δοκιμής κατά τη μετάβαση του σήματος χρονισμού TCK από 1 σε 0 αποθηκεύονται παράλληλα στη βαθμίδα ενημέρωσης του περιφερειακού καταχωρητή του OK1. Ο ελεγκτής του OK2 πρέπει να βρίσκεται στην κατάσταση Capture-DR ώστε κατά τη μετάβαση του σήματος χρονισμού TCK από 0 σε 1 οι τιμές που οδηγούν τις εισόδους του OK2 να αποθηκευτούν στη βαθμίδα ολίσθησης του περιφερειακού καταχωρητή του OK2(βλέπε σχήμα 9.11.γ). Στη συνέχεια ο ελεγκτής του OK2 τίθεται στην κατάσταση Shift-DR και η απόκριση μεταφέρεται έξω σειριακά μέσω της εξόδου TDO (βλέπε σχήμα 9.11.δ).



Σχήμα 9.11.α Τα ολοκληρωμένα κυκλώματα OK1 και OK2 υλοποιούν το πρότυπο και θέλουμε να ελέγξουμε την ορθή λειτουργία των γραμμών διασύνδεσής τους.

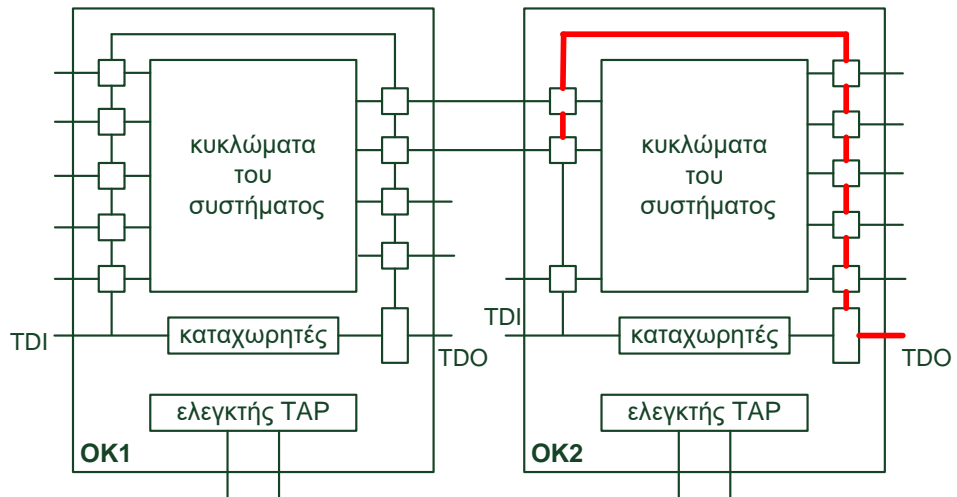


Σχήμα 9.11.β Εκτέλεση της εντολής EXTEST



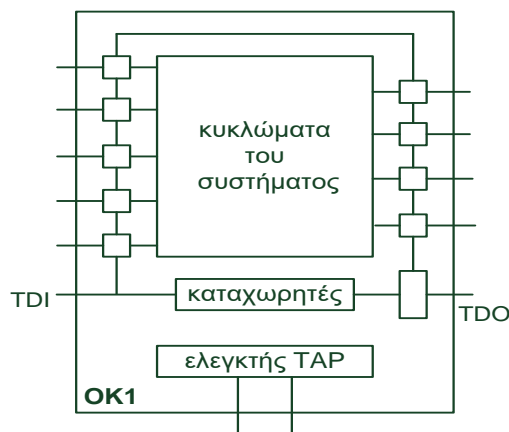
Σχήμα 9.11.γ Εκτέλεση της εντολής EXTEST

4. Shift-DR (OK2)

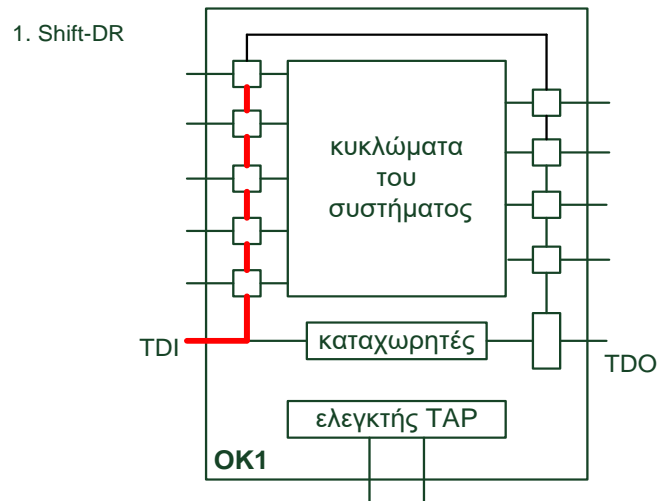


Σχήμα 9.11.δ Εκτέλεση της εντολής EXTEST

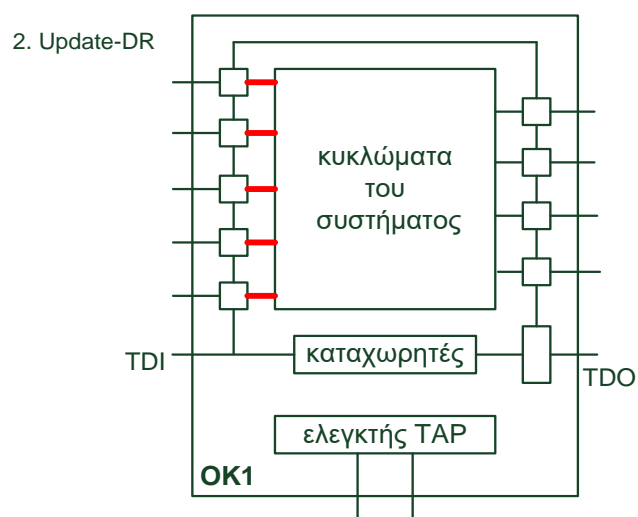
- **INTEST** Σύμφωνα με το πρότυπο η εντολή αυτή είναι προαιρετική, και ο συνδυασμός δυαδικών τιμών 0 και 1 που αποτελούν τον κωδικό της δεν καθορίζεται από το πρότυπο. Αυτή η εντολή τοποθετεί μεταξύ της σειριακής εισόδου δεδομένων δοκιμής TDI και της σειριακής εξόδου δεδομένων δοκιμής TDO τον περιφερειακό καταχωρητή ολίσθησης. Αυτή η εντολή μας δίνει τη δυνατότητα ελέγχου της ορθής λειτουργίας των κυκλωμάτων του συστήματος των OK που υλοποιούν το πρότυπο. Στο σχήμα 9.12 δίνονται τα βήματα για τον έλεγχο της ορθής λειτουργίας των κυκλωμάτων του συστήματος ενός OK που υλοποιεί το πρότυπο.



Σχήμα 9.12.α Θέλουμε να ελέγξουμε τη ορθή λειτουργία των κυκλωμάτων του συστήματος του OK1.

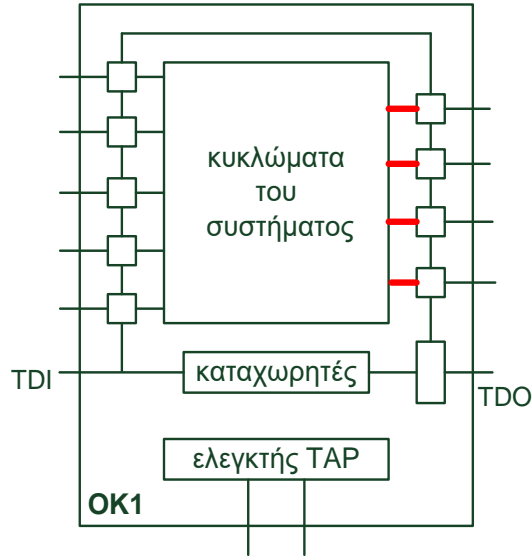


Σχήμα 9.12.β Εκτέλεση της εντολής INTEST



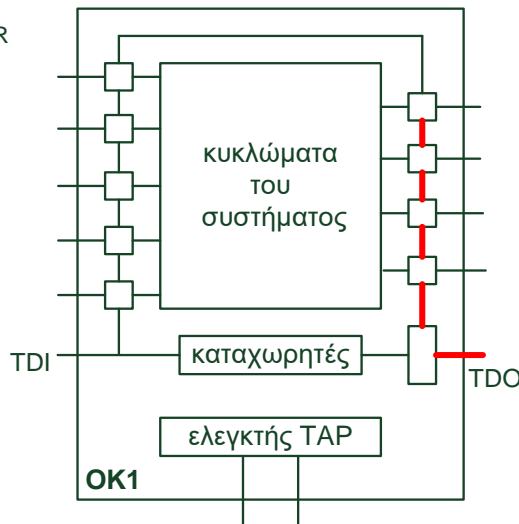
Σχήμα 9.12.γ Εκτέλεση της εντολής INTEST

3. Capture-DR



Σχήμα 9.12.δ Εκτέλεση της εντολής INTEST

4. Shift-DR



Σχήμα 9.12.ε Εκτέλεση της εντολής INTEST

- RUNBIST** Σύμφωνα με το πρότυπο η εντολή αυτή είναι προαιρετική, και ο συνδυασμός δυαδικών τιμών 0 και 1 που αποτελούν τον κωδικό της δεν καθορίζεται από το πρότυπο. Ο σχεδιαστής αποφασίζει όταν εκτελείται η εντολή αυτή ποιος καταχωρητής θα τοποθετηθεί μεταξύ της σειριακής εισόδου δεδομένων δοκιμής και της σειριακής εξόδου δεδομένων δοκιμής. Η εντολή αυτή δίνει την δυνατότητα να εκτελεστεί μια διαδικασία αυτοελέγχου μέσα στο OK χρησιμοποιώντας μόνο μια εντολή. Ο αυτοέλεγχος (self-test) γίνεται όταν ο ελεγκτής βρίσκεται στην κατάσταση RUN-TEST-IDLE. Το σήμα χρονισμού του αυτοελέγχου μπορεί να βασίζεται στο σήμα χρονισμού

TCK, στο σήμα ή σήματα χρονισμού του συστήματος ή και στα δύο. Ο αυτοέλεγχος μπορεί να διαρκεί πολλές χρονικές περιόδους. Θα πρέπει να υπάρχει δυνατότητα η απόκριση του υπό έλεγχο ολοκληρωμένου κυκλώματος να παραμένει σταθερή για οποιοδήποτε περαιτέρω αριθμό χρονικών περιόδων απαιτηθεί. Η διατήρηση της απόκρισης δίνει τη δυνατότητα περισσότερα από ένα κυκλώματα, τα οποία απαιτούν διαφορετικό αριθμό χρονικών περιόδων για αυτοέλεγχο, να εκτελούν ταυτόχρονα την εντολή RUNBIST. Οι αποκρίσεις, όταν ο ελεγκτής είναι στην κατάσταση Capture-DR, αποθηκεύονται στον καταχωρητή που όρισε ο σχεδιαστής και στη συνέχεια μεταφέρονται σειριακά εκτός πλακέτας ή συστήματος προς έλεγχο.

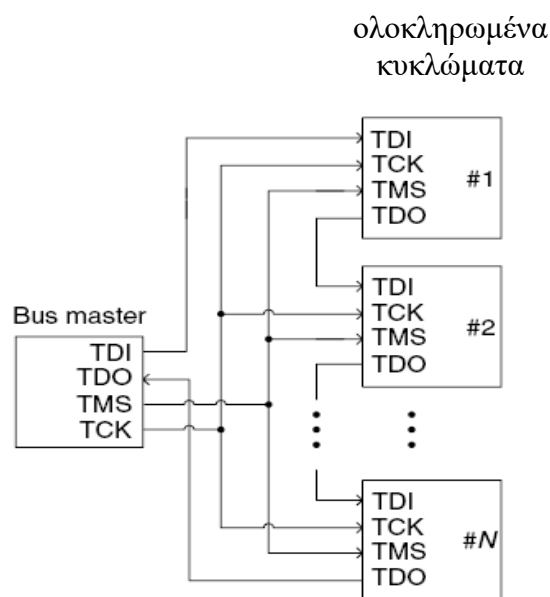
- **HIGHZ** Σύμφωνα με το πρότυπο η εντολή αυτή είναι προαιρετική, και ο συνδυασμός δυαδικών τιμών 0 και 1 που αποτελούν τον κωδικό της δεν καθορίζεται από το πρότυπο. Αυτή η εντολή τοποθετεί μεταξύ της σειριακής εισόδου δεδομένων δοκιμής TDI και της σειριακής εξόδου δεδομένων δοκιμής TDO τον καταχωρητή παράκαμψης ώστε να μειωθεί το μονοπάτι ολίσθησης. Η εντολή αυτή επίσης αναγκάζει όλους τους ακροδέκτες εξόδου και εισόδου/ εξόδου του ολοκληρωμένου να πάνε σε κατάσταση υψηλής εμπέδησης. Σκοπός της εντολής αυτής είναι να δώσει τη δυνατότητα χρησιμοποίησης “In-Circuit test” χωρίς τον κίνδυνο να καταστρέψουμε ένα ολοκληρωμένο κύκλωμα λόγω υπερδοήγησης (overdrive damage). Εκτελώντας σε ένα ολοκληρωμένο κύκλωμα OK1 την εντολή HIGHZ οι έξοδοί του τίθενται σε κατάσταση υψηλής εμπέδησης, επομένως μπορούμε να οδηγήσουμε τις εισόδους του OK2, οι οποίες συνδέονται με τις εξόδους του OK1, με ένα “In-Circuit tester”.
- **CLAMP** Σύμφωνα με το πρότυπο η εντολή αυτή είναι προαιρετική, και ο συνδυασμός δυαδικών τιμών 0 και 1 που αποτελούν τον κωδικό της δεν καθορίζεται από το πρότυπο. Αυτή η εντολή τοποθετεί μεταξύ της σειριακής εισόδου δεδομένων δοκιμής TDI και της σειριακής εξόδου δεδομένων δοκιμής TDO τον καταχωρητή παράκαμψης. Όταν εκτελείται η εντολή CLAMP, η κατάσταση όλων των σημάτων που οδηγούνται από τους ακροδέκτες εξόδου του ολοκληρωμένου κυκλώματος πρέπει να δίνεται από τις τιμές που είναι αποθηκευμένες στη βαθμίδα ενημέρωσης του περιφερειακού καταχωρητή. Το περιεχόμενο της βαθμίδας ενημέρωσης του περιφερειακού καταχωρητή έχει φορτωθεί σειριακά με μια προηγούμενη εντολή PRELOAD. Κατά τον έλεγχο της ορθής λειτουργίας μιας πλακέτας είναι συχνά απαραίτητο να θέσουμε στη λογική τιμή 0 ή 1 επιλεγμένους κόμβους για να διευκολύνουμε τον έλεγχο ή να αποφύγουμε αλληλεπίδραση. Αυτό επιτυγχάνεται εύκολα με την εντολή αυτή όταν οι συγκεκριμένοι κόμβοι οδηγούνται από περιφερειακούς καταχωρητές ολίσθησης.

9.7 Αρχιτεκτονικές Χρησιμοποίησης του Πρότυπου Περιφερειακής Ολίσθησης σε Επίπεδο Συστήματος & Πλακέτας

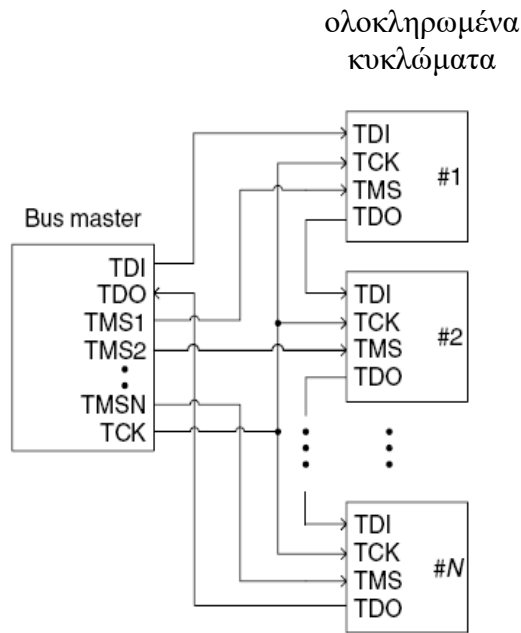
Θεωρούμε ότι το πρότυπο έχει υλοποιηθεί με επιτυχία στα ολοκληρωμένα κυκλώματα που χρησιμοποιούμε για τη σχεδίαση μιας πλακέτας ή ενός συστήματος. Ο σχεδιαστής της πλακέτας ή του συστήματος αντιμετωπίζει το πρόβλημα του πώς θα τροφοδοτήσει με δεδομένα δοκιμής και σήματα ελέγχου τα ολοκληρωμένα κυκλώματα πάνω στην πλακέτα και στο σύστημα. Για το σκοπό αυτό έχουν προταθεί διάφορες αρχιτεκτονικές.

- **Αρχιτεκτονική απλού δακτυλίου με κοινό σήμα επιλογής τρόπου λειτουργίας**

Σ' αυτή την αρχιτεκτονική (βλέπε σχήμα 9.13), οι περιφερειακοί καταχωρητές των ολοκληρωμένων κυκλωμάτων συνδέονται με τέτοιο τρόπο ώστε η σειριακή έξοδος δεδομένων του πρώτου ολοκληρωμένου να οδηγεί τη σειριακή είσοδο δεδομένων του δεύτερου ολοκληρωμένου και ούτω καθεξής. Όλα τα ολοκληρωμένα δέχονται το ίδιο σήμα TMS. Επίσης όλα τα ολοκληρωμένα θα εκτελούν πάντοτε την ίδια λειτουργία Capture, Shift ή Update υπό τον έλεγχο του ελεγκτή της πόρτας προσπέλασης δεδομένων δοκιμής (TAP controller). Ωστόσο μέσω του μονοπατιού TDI-TDO μπορούμε να οδηγήσουμε διαφορετικές εντολές στα διαφορετικά ολοκληρωμένα κυκλώματα. Το πλεονέκτημα αυτής της αρχιτεκτονικής είναι η απλότητά της. Το μειονέκτημά της είναι το πολύ μεγάλο μήκος μονοπατιού ολίσθησης που μπορεί να σχηματιστεί. Βέβαια λόγω της χρησιμοποίησης των εντολών BYPASS το πραγματικό μήκος του μονοπατιού ολίσθησης κατά τη χρησιμοποίησή του θα είναι πολύ μικρότερο από το μέγιστο μήκος του.



Σχήμα 9.13 Αρχιτεκτονική απλού δακτυλίου με κοινό σήμα επιλογής τρόπου λειτουργίας



Σχήμα 9.14 Αρχιτεκτονική απλού δακτυλίου με διακριτά σήματα επιλογής τρόπου λειτουργίας.

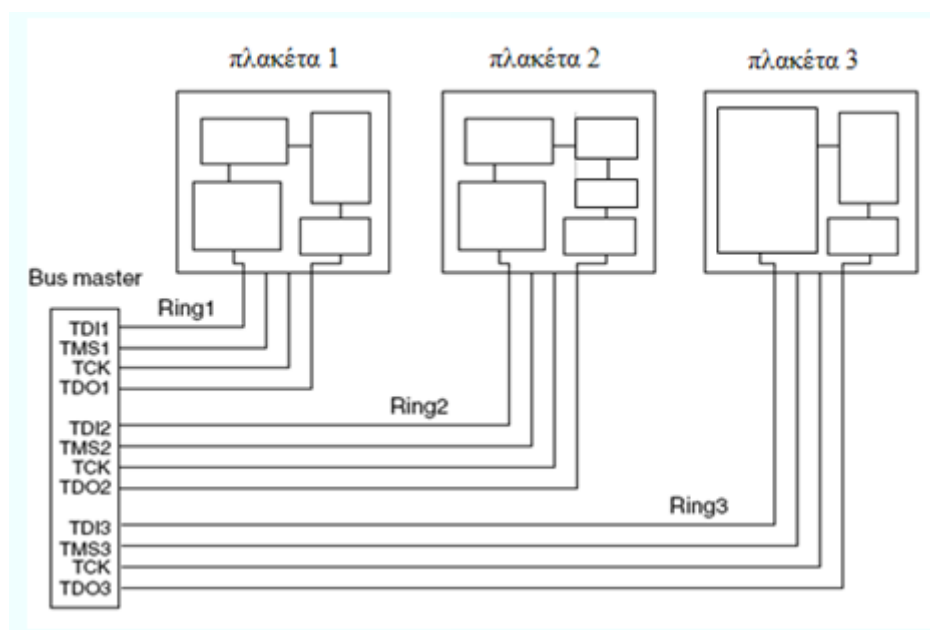
- **Αρχιτεκτονική απλού δακτυλίου με διακριτά σήματα επιλογής τρόπου λειτουργίας**

Και σ' αυτή την αρχιτεκτονική οι περιφερειακοί καταχωρητές των ολοκληρωμένων κυκλωμάτων συνδέονται με τέτοιο τρόπο ώστε να η σειριακή έξοδος δεδομένων του πρώτου ολοκληρωμένου να οδηγεί τη σειριακή είσοδο δεδομένων του δεύτερου ολοκληρωμένου και ούτω καθεξής. Όμως κάθε ένα ολοκληρωμένο κύκλωμα δέχεται το δικό του σήμα επιλογής τρόπου λειτουργίας TMS (βλέπε σχήμα 9.14). Επομένως μπορούμε να οδηγήσουμε στα ολοκληρωμένα κυκλώματα διαφορετικές εντολές αλλά και να εκτελέσουμε διαφορετικές λειτουργίες Capture, Shift ή Update. Μ' αυτή την αρχιτεκτονική όλα τα ολοκληρωμένα κυκλώματα μπορούν να ελεγχθούν ανεξάρτητα. Πλεονέκτημα και αυτής της αρχιτεκτονικής είναι η απλότητα της. Ο έλεγχος της ορθής λειτουργίας μιας πλακέτας είναι αποτελεσματικός. Ωστόσο για ένα σύστημα που περιέχει έναν αριθμό από πλακέτες που κάθε μια περιέχει πολλά ολοκληρωμένα κυκλώματα που υλοποιούν το πρότυπο, η αρχιτεκτονική του απλού δακτυλίου μπορεί να μην είναι αποδοτική λόγω του μεγάλου μήκους. Επίσης σε επίπεδο συστήματος μπορεί να προκύψουν προβλήματα που να οφείλονται στο γεγονός ότι το σύστημα δίνει τη δυνατότητα διαφορετικών διαμορφώσεων (configurations), δηλαδή το σύστημα λειτουργεί σωστά είτε σε κάποιες θέσεις (slots) υπάρχουν πλακέτες είτε όχι. Εάν σε κάποια θέση δεν

υπάρχει πλακέτα τότε θα πρέπει να τοποθετηθεί, κάποιο είδος γέφυρας (jumper), διαφορετικά σ' εκείνο το σημείο θα έχουμε διακοπή του μονοπατιού ολίσθησης.

- **Αρχιτεκτονική αστέρα ή πολλών δακτυλίων**

Στην αρχιτεκτονική αυτή κάθε πλακέτα έχει τη δική της σειριακή είσοδος δεδομένων δοκιμής TDI και σειριακή έξοδο δεδομένων δοκιμής, το δικό της σήμα χρονισμού TCK και το δικό της σήμα επιλογής τρόπου λειτουργίας TMS (βλέπε σχήμα 9.15). Πλεονέκτημα αυτής της αρχιτεκτονικής είναι ότι κάθε πλακέτα ελέγχεται σαν να είναι μόνη της. Μειονέκτημα είναι ο μεγάλος αριθμός γραμμών διασύνδεσης πάνω στην κύρια πλακέτα (backplane).

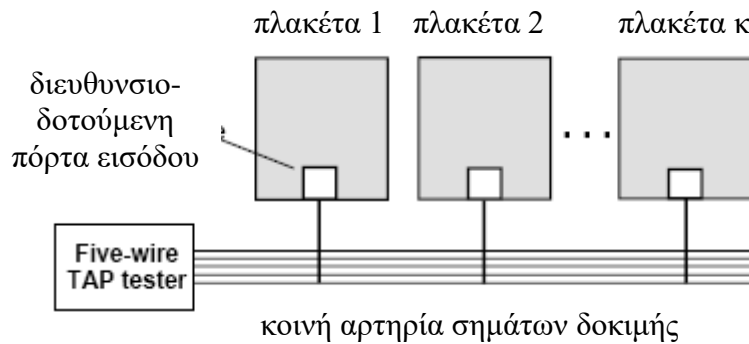


Σχήμα 9.15 Αρχιτεκτονική αστέρα ή πολλών δακτυλίων

- **Αρχιτεκτονική κοινής αρτηρίας σημάτων δοκιμής (multidrop architecture)**

Στην αρχιτεκτονική αυτή υπάρχει μια σειριακή είσοδος δεδομένων δοκιμής TDI, μια σειριακή έξοδος δεδομένων δοκιμής TDO, ένα σήμα χρονισμού TCK και ένα σήμα επιλογής τρόπου λειτουργίας TMS, τα οποία όμως συνδέονται παράλληλα προς όλες τις πλακέτες του συστήματος (βλέπε σχήμα 9.16. Για να εξασφαλίσουμε ότι κάθε φορά ελέγχεται η ορθή λειτουργία μόνο μιας πλακέτας, πάνω σε κάθε πλακέτα υλοποιείται μια διευθυνσιοδοτούμενη πόρτα εισόδου. Ο ελεγκτής ελέγχου ορθής λειτουργίας σε επίπεδο συστήματος χρησιμοποιεί ένα ειδικό πρωτόκολλο για την επιλογή της πλακέτας η οποία θα συνδεθεί στην αρτηρία σημάτων δοκιμής. Αυτή η αρχιτεκτονική χρησιμοποιείται ευρέως στη βιομηχανία. Η αρχιτεκτονική “Addressable Shadow Port” που αναπτύχθηκε από την Texas Instrument και η

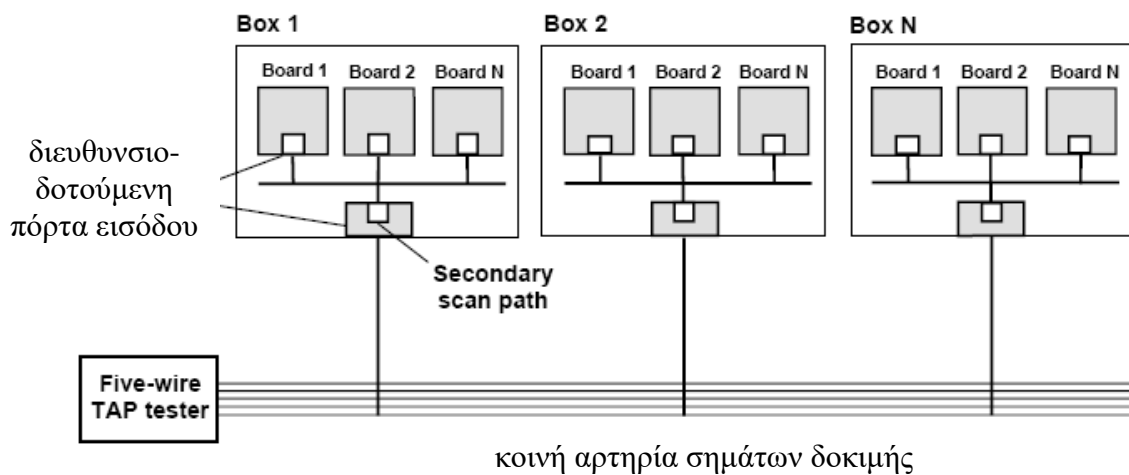
αρχιτεκτονική “Scan-Bridge” που αναπτύχθηκε από την National Semiconductor ανήκουν σ’ αυτή την κατηγορία.



Σχήμα 9.16 Αρχιτεκτονική κοινής αρτηρίας σημάτων δοκιμής (multidrop architecture)

- **Ιεραρχική αρχιτεκτονική**

Στα πολύπλοκα συστήματα το μήκος του μονοπατιού ολίσθησης μπορεί να είναι πάρα πολύ μεγάλο. Μια ιεραρχική αρχιτεκτονική έχει το πλεονέκτημα ότι σπάει το μονοπάτι ολίσθησης σε πολλά μονοπάτια υπό μορφή δέντρου, επομένως η προσπέλαση κάθε ενός ολοκληρωμένου γίνεται πολύ πιο γρήγορα. Επίσης αυτή η αρχιτεκτονική ταιριάζει με την ιεραρχική αρχιτεκτονική των συστημάτων. Η ιεραρχική αρχιτεκτονική πολλές φορές συνδυάζεται με την αρχιτεκτονική κοινής αρτηρίας σημάτων δοκιμής, όπως φαίνεται στο σχήμα 9.17.



Σχήμα 9.17 Ιεραρχική αρχιτεκτονική

Κεφάλαιο 10. Πρότυπο ελέγχου εμφωλευμένων προσχεδιασμένων μονάδων (IEEE STD 1500)

10.1 Προβλήματα ελέγχου της ορθής λειτουργίας συστημάτων που υλοποιούνται σε ένα ολοκληρωμένο κύκλωμα (System-On-Chip, SOC)

Ένα τυπικό σύστημα που υλοποιείται σε ένα ολοκληρωμένο κύκλωμα μπορεί να περιέχει ένα μεγάλο αριθμό από προσχεδιασμένες μονάδες (cores). Μια πλακέτα ενός συστήματος περιέχει πολλά ολοκληρωμένα κυκλώματα (chips). Θα μπορούσαμε επομένως να πούμε ότι υπάρχει μια αντιστοιχία μεταξύ ενός συστήματος που υλοποιείται σε επίπεδο πλακέτας και ενός συστήματος που υλοποιείται σε ένα ολοκληρωμένο κύκλωμα. Ωστόσο όσον αφορά τον έλεγχο της ορθής τους λειτουργίας υπάρχει μια σημαντική διαφορά. Σε ένα σύστημα, το οποίο αποτελείται από μία ή περισσότερες πλακέτες, τα ολοκληρωμένα κυκλώματα κατασκευάστηκαν, ελέγχθηκε η ορθή τους λειτουργία και στη συνέχεια τοποθετήθηκαν στην πλακέτα. Επομένως, όσον αφορά τον έλεγχο της ορθής λειτουργίας, το σχεδιαστή και τον κατασκευαστή των πλακετών ή του συστήματος τον απασχολεί κυρίως ο έλεγχος της ορθής λειτουργίας των γραμμών διασύνδεσης των ολοκληρωμένων κυκλωμάτων πάνω στην πλακέτα και των πλακετών στο σύστημα. Αντίθετα στην περίπτωση των συστημάτων που υλοποιούνται σε ένα ολοκληρωμένο κύκλωμα θα πρέπει μετά την κατασκευή τους να γίνει έλεγχος της ορθής λειτουργίας τόσο κάθε μιας υπομονάδας του συστήματος όσο και των γραμμών διασύνδεσης των υπομονάδων. Είναι ευθύνη του σχεδιαστή του συστήματος να αναπτύξει τη στρατηγική ελέγχου της ορθής λειτουργίας των υπομονάδων του συστήματος και των γραμμών διασύνδεσής τους. Υπάρχουν πολλά προβλήματα στην ανάπτυξη αυτής της στρατηγικής. Στη συνέχεια όταν χρησιμοποιούμε τη λέξη σύστημα θα εννοούμε ένα σύστημα που είναι υλοποιημένο σε ένα ολοκληρωμένο κύκλωμα (SOC).

- **Διαφορετικές τεχνολογίες.** Ένα σύστημα μπορεί να περιέχει προσχεδιασμένες μονάδες επεξεργαστών, μνημών, αναλογικών κυκλωμάτων κλπ. Όλες αυτές οι μονάδες πρέπει να ελεγχθούν μετά την κατασκευή του συστήματος. Ο σχεδιαστής του συστήματος είναι αδύνατον μόνος του να εξάγει τα σύνολα δοκιμής όλων αυτών των μονάδων και στη συνέχεια να σχεδιάσει τη στρατηγική ελέγχου της ορθής τους λειτουργίας. Στο έργο του αυτό θα πρέπει να βοηθηθεί από τους σχεδιαστές κάθε προσχεδιασμένης μονάδας.
- **Διαφορετικοί προμηθευτές προσχεδιασμένων μονάδων.** Οι προσχεδιασμένες μονάδες μπορεί να έχουν αγοραστεί από διαφορετικούς προμηθευτές ή να έχουν σχεδιαστεί από άλλες ομάδες της ίδιας εταιρείας. Επομένως ο προμηθευτής της προσχεδιασμένης μονάδας θα πρέπει να προμηθεύει το σχεδιαστή του συστήματος με το σύνολο δοκιμής της μονάδας και τη στρατηγική που θα πρέπει να ακολουθηθεί για τον έλεγχο της ορθής της λειτουργίας μετά την κατασκευή της. Άρα, απαιτείται ένας τυποποιημένος τρόπος επικοινωνίας μεταξύ των σχεδιαστών των προσχεδιασμένων μονάδων και των σχεδιαστών του συστήματος.

- Προστασία πνευματικής ιδιοκτησίας.**

Για λόγους προστασίας πνευματικών δικαιωμάτων δεν είναι γνωστή η εσωτερική δομή κάποιων προσχεδιασμένων μονάδων. Σ' αυτές τις περιπτώσεις ο προμηθευτής των προσχεδιασμένων μονάδων είναι υποχρεωμένος να εφοδιάσει το σχεδιαστή του συστήματος με όλη την πληροφορία που απαιτείται για τον έλεγχο της ορθής λειτουργίας της προσχεδιασμένης μονάδας μετά την κατασκευή της. Ο σχεδιαστής του συστήματος πρέπει να ακολουθήσει την τακτική την οποίαν του γνωστοποίησε ο προμηθευτής.
- Εμφωλευμένες μονάδες.**

Οι είσοδοι και οι έξοδοι μιας προσχεδιασμένης μονάδας μπορεί να μην είναι κύριες είσοδοι και κύριες έξοδοι ενός ολοκληρωμένου κυκλώματος (εμφωλευμένη μονάδα). Επομένως, απαιτείται ένας μηχανισμός προσπέλασης (test access mechanism, TAM), των εισόδων και εξόδων της προσχεδιασμένης μονάδας από τις κύριες εισόδους και εξόδους του ολοκληρωμένου κυκλώματος. Είναι επιθυμητό οι προσχεδιασμένες μονάδες να μπορούν να συνδεθούν εύκολα με το διαθέσιμο μηχανισμό προσπέλασης. Μια προσχεδιασμένη μονάδα μπορεί να έχει σχεδιασθεί χρησιμοποιώντας άλλες προσχεδιασμένες υπομονάδες, δηλαδή υπάρχει μια ιεραρχική δομή. Επομένως και ο μηχανισμός προσπέλασης πρέπει να έχει μια ιεραρχική δομή.
- Εσωτερικές συχνότητες λειτουργίας μεγαλύτερες από τις συχνότητες εισόδου/ εξόδου του ολοκληρωμένου κυκλώματος.**

Σε πολλά συστήματα υπάρχουν εμφωλευμένες μονάδες οι οποίες χρησιμοποιούν συχνότητες λειτουργίας πολύ μεγαλύτερες απ' αυτές των ακροδεκτών εισόδου/ εξόδου του ολοκληρωμένου κυκλώματος. Σ' αυτή την περίπτωση, όταν ο έλεγχος βασίζεται σε εξωτερική συσκευή ελέγχου (ATE), ακόμη και αν υπάρχει ο κατάλληλος μηχανισμός προσπέλασης, δεν είναι δυνατόν η συγκεκριμένη μονάδα να ελεγχθεί στη συχνότητα λειτουργίας της. Το πρόβλημα λύνεται υλοποιώντας πάνω στο ίδιο ολοκληρωμένο κύκλωμα ένα μέρος του ελεγκτή και του μηχανισμού παραγωγής διανυσμάτων δοκιμής. Τότε απαιτούνται εξωτερικές συσκευές ελέγχου μικρότερων δυνατοτήτων και επομένως και μικρότερου κόστους.
- Κατανάλωση ισχύος κατά τον έλεγχο της ορθής λειτουργίας του συστήματος.**

Υπάρχουν διάφοροι λόγοι για τους οποίους η κατανάλωση ισχύος κατά τον έλεγχο της ορθής λειτουργίας είναι μεγαλύτερη από ότι κατά την κανονική λειτουργία του συστήματος. Υπερβολική κατανάλωση ισχύος κατά τον έλεγχο της ορθής λειτουργίας μπορεί να οδηγήσει στο συμπέρασμα ότι κάποιο σύστημα είναι ελαττωματικό, ενώ στην πραγματικότητα δεν έχει πρόβλημα, ή ακόμη και στην καταστροφή του υπό έλεγχο συστήματος.
- Μεγάλος χρόνος ελέγχου της ορθής λειτουργίας του συστήματος.**

Ο έλεγχος της ορθής λειτουργίας μιας υπομονάδας του συστήματος είναι μεγάλος. Ο χρόνος ελέγχου της ορθής λειτουργίας του συστήματος γίνεται απαγορευτικά μεγάλος όταν ελέγχουμε την ορθή λειτουργία της μιας μονάδας μετά την άλλη σειριακά. Για να μειωθεί ο συνολικός χρόνος ελέγχου της ορθής λειτουργίας ενός συστήματος θα πρέπει ο έλεγχος της ορθής λειτουργίας κάποιων μονάδων του συστήματος να γίνεται παράλληλα. Επομένως για κάθε

σύστημα θα πρέπει να γίνεται χρονοπρογραμματισμός του ελέγχου της σωστής λειτουργίας των μονάδων του λαμβάνοντας υπ' όψιν τους διαθέσιμους μηχανισμούς προσπέλασης και την κατανάλωση ισχύος κατά τον έλεγχο της ορθής λειτουργίας.

- **Αυτοματοποίηση της διαδικασίας σχεδίασης για εύκολο έλεγχο ορθής λειτουργίας**

Από τα ανωτέρω συμπεραίνουμε ότι ο έλεγχος της ορθής λειτουργίας συστημάτων που υλοποιούνται σε ένα ολοκληρωμένο κύκλωμα και βασίζονται στη χρήση προσχεδιασμένων μονάδων έχει πολλές δυσκολίες. Είναι επιθυμητή επομένως η ανάπτυξη εργαλείων (Design for Testability, DFT, tools) τα οποία θα αυτοματοποιήσουν σε κάποιο βαθμό τη διαδικασία.

Η ύπαρξη ενός προτύπου είναι σημαντική στην ανάπτυξη εργαλείων αλλά και στη διευκόλυνση της επικοινωνίας μεταξύ των προμηθευτών των προσχεδιασμένων μονάδων και των σχεδιαστών του συστήματος, με στόχο την επίλυση των προβλημάτων που σχετίζονται με τον έλεγχο της ορθής λειτουργίας των συστημάτων που υλοποιούνται σε ένα ολοκληρωμένο κύκλωμα.

Το πρότυπο πρέπει να ορίζει μια αρχιτεκτονική ώστε να είναι δυνατός ο έλεγχος της ορθής λειτουργίας μιας προσχεδιασμένης μονάδας, η προσπέλασή της μέσω ενός μηχανισμού προσπέλασης, ο έλεγχος ορθής λειτουργίας των διασυνδέσεων μεταξύ προσχεδιασμένων και μη μονάδων, καθώς και η προστασία από βλάβη μιας μονάδας όταν γίνεται ο έλεγχος της ορθής λειτουργίας κάποιας άλλης μονάδας.

10.2 Αρχιτεκτονική

Το πιο σημαντικό χαρακτηριστικό του προτύπου 1500 είναι η ύπαρξη ενός κολάρου (wrapper) στην περιφέρεια (γύρω από τους ακροδέκτες εισόδου/εξόδου) μιας προσχεδιασμένης μονάδας. Η χρησιμοποίηση ενός κολάρου δίνει τη δυνατότητα απομόνωσης μιας προσχεδιασμένης μονάδας από τις υπόλοιπες μονάδες του ολοκληρωμένου κυκλώματος. Όπως θα δούμε η απομόνωση της προσχεδιασμένης μονάδας με τη χρησιμοποίηση ενός καταχωρητή επιτρέπει τον έλεγχο της ορθής της λειτουργίας, όταν είναι εμφωλευμένη σε ένα ολοκληρωμένο κύκλωμα, χρησιμοποιώντας το σύνολο δοκιμής που εξήχθη αντιμετωπίζοντας τη μονάδα μόνη της, δηλαδή πριν να γνωρίζουμε σε ποιο ολοκληρωμένο κύκλωμα θα εμφωλευτεί. Επίσης δίνει τη δυνατότητα να ελέγξουμε πλήρως την ορθή λειτουργία της λογικής που βρίσκεται έξω από την προσχεδιασμένη μονάδα χωρίς να πρέπει να προσπελάσουμε την προσχεδιασμένη μονάδα.

Σύμφωνα με το πρότυπο το κολάρο πρέπει υποχρεωτικά να έχει μια σειριακή πόρτα. Η σειριακή πόρτα του κολάρου (wrapper serial port, WSP) αποτελείται από ένα σύνολο ακροδεκτών εισόδου και εξόδου για σειριακές λειτουργίες. Αποτελείται από τον ακροδέκτη σειριακής εισόδου (wrapper serial input, WSI), τον ακροδέκτη σειριακής εξόδου (wrapper serial output, WSO) και εισόδους σημάτων ελέγχου της σειριακής πόρτας του κολάρου (wrapper serial control, WSC). Σε σύγκριση με το πρότυπο IEEE 1149.1, το πρότυπο IEEE 1500 έχει περισσότερες εισόδους ελέγχου. Η διαφοροποίηση οφείλεται στο γεγονός ότι το πρότυπο IEEE 1149.1 αφορά ολοκληρωμένα κυκλώματα, ενώ το πρότυπο IEEE 1500 αφορά μονάδες ενός ολοκληρωμένου κυκλώματος. Λόγω της ύπαρξης πολλών σημάτων ελέγχου η αρχιτεκτονική του 1500 δεν χρειάζεται την υλοποίηση μιας μηχανής πεπερασμένων

καταστάσεων για να προσαρμόσει ένα περιορισμένο αριθμό ακροδεκτών εισόδου σε ένα μεγάλο αριθμό από εσωτερικά σήματα ελέγχου και καταστάσεις της μηχανής πεπερασμένων καταστάσεων. Κάθε κολάρο έχει ένα καταχωρητή εντολών (wrapper instruction register, WIR), ο οποίος χρησιμοποιείται για να αποθηκευτεί η προς εκτέλεση εντολή.

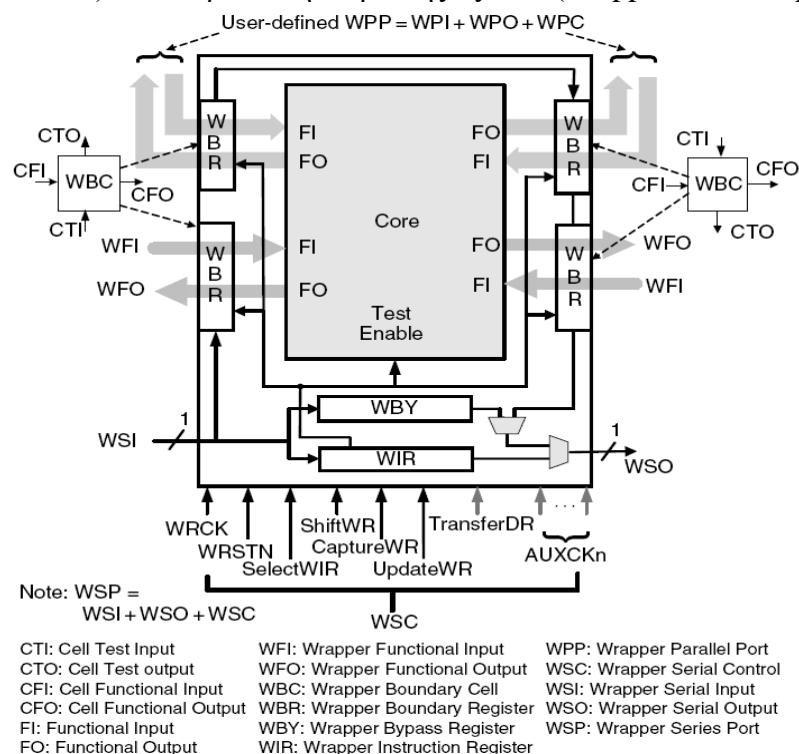
Σύμφωνα με το πρότυπο το κολάρο μπορεί να έχει προαιρετικά και μια παράλληλη πόρτα. Η παράλληλη πόρτα του κολάρου (wrapper parallel port, WPP) αποτελείται από ένα σύνολο παράλληλων ακροδεκτών εισόδου (wrapper parallel Input, WPI), ένα σύνολο παράλληλων ακροδεκτών εξόδου (wrapper parallel Output, WPO) και ένα σύνολο εισόδων ελέγχου (wrapper parallel Control, WPC) για τις παράλληλες λειτουργίες του κολάρου. Όταν το κολάρο διαθέτει παράλληλη πόρτα τότε η προσπέλαση για μεταφορά δεδομένων δοκιμής γίνεται μέσω ενός παράλληλου μηχανισμού προσπέλασης που ορίζεται από το χρήστη της προσχεδιασμένης μονάδας.

10.3 Συνιστώσες και λειτουργίες του κολάρου

Στο σχήμα 10.1 δίνεται η δομή του κολάρου του προτύπου IEEE 1500, το οποίο αποτελείται από πέντε συνιστώσες.

1. Σειριακή πόρτα (wrapper serial port, WSP)

Η σειριακή πόρτα αποτελείται από τον ακροδέκτη σειριακής εισόδου (wrapper serial input, WSI), τον ακροδέκτη σειριακής εξόδου (wrapper serial output,



Σχήμα 10.1 Οι συνιστώσες του προτύπου 1500.

WSO) και εισόδους ελέγχου της σειριακής πόρτας του κολάρου (wrapper serial control, WSC). Η σειριακή είσοδος και η σειριακή έξοδος του κολάρου

χρησιμοποιούνται για να μεταφέρουμε σειριακά εντολές δοκιμής και δεδομένα δοκιμής μέσα στο κολάρο και έξω από το κολάρο. Υπάρχουν έξι υποχρεωτικές εισοδοί ελέγχου (WRSTN, WRCK, SelectWIR, CaptureWR, ShiftWR, και UpdateWR), μια προαιρετική εισόδος ελέγχου (TransferDR) και μερικές προαιρετικές εισοδοί για σήματα χρονισμού (AUXCKn). Στη συνέχεια θα περιγράψουμε αυτές τις εισόδους ελέγχου. Κάποιες λειτουργίες που ενεργοποιούνται από αυτά τα σήματα ελέγχου θα περιγραφούν αργότερα.

- **WRCK** Αυτή η υποχρεωτική εισόδος χρησιμοποιείται μόνο για το χρονισμό των συνιστωσών του κολάρου. Η εισόδος αυτή δεν επιτρέπεται να μοιραστεί με κάποια άλλη εισόδοι χρονισμού του κολάρου. Επίσης πρέπει να είναι ανεξάρτητη από το σήμα ή τα σήματα χρονισμού της προσχεδιασμένης μονάδας.

- **AUXCKn** Από 1 έως και n προαιρετικά σήματα χρονισμού που μπορούν να χρησιμοποιηθούν για να ελέγξουν κάποιες συνιστώσες του κολάρου. Αυτά τα σήματα χρονισμού μπορούν να μοιραστούν με τα σήματα χρονισμού του συστήματος. Όταν χρησιμοποιούνται, ο χρήστης πρέπει να ορίζει με σαφήνεια την σχέση τους με το WRCK.

- **WRSTN** Αυτός ο ακροδέκτης είναι υποχρεωτικός σύμφωνα με το πρότυπο, αρχικοποιεί το κολάρο και όταν οδηγηθεί με την ενεργό τιμή το θέτει σε κανονικό τρόπο λειτουργίας. Επίσης όταν η εισόδος αυτή οδηγηθεί με την ενεργό τιμή, τότε στο καταχωρητή εντολής του κολάρου φορτώνεται αυτόματα η εντολή Bypass.

- **SelectWIR** Στο πρότυπο 1500 κάθε λειτουργία ξεκινά με την επιλογή του καταχωρητή που θα χρησιμοποιηθεί στη λειτουργία. Σύμφωνα με το πρότυπο υπάρχουν ο καταχωρητής εντολής και οι καταχωρητές δεδομένων. Όταν η εισόδος SelectWIR έχει τη λογική τιμή 1, τότε μεταξύ της σειριακής εισόδου και της σειριακής εξόδου του κολάρου τοποθετείται ο καταχωρητής εντολής. Όταν η εισόδος SelectWIR έχει τη λογική τιμή 0, τότε μεταξύ της σειριακής εισόδου του κολάρου και της σειριακής εξόδου του κολάρου τοποθετείται κάποιος από τους καταχωρητές δεδομένων. Ο καταχωρητής δεδομένων που θα επιλεγεί εξαρτάται από το είδος της εντολής που έχει φορτωθεί σειριακά στο καταχωρητή εντολής και εκτελείται.

Τα σήματα **ShiftWR**, **CaptureWR**, **UpdateWR** και **TransferDR** ενεργοποιούν τις αντίστοιχες στοιχειώσεις λειτουργίες τις οποίες θα περιγράψουμε πιο κάτω.

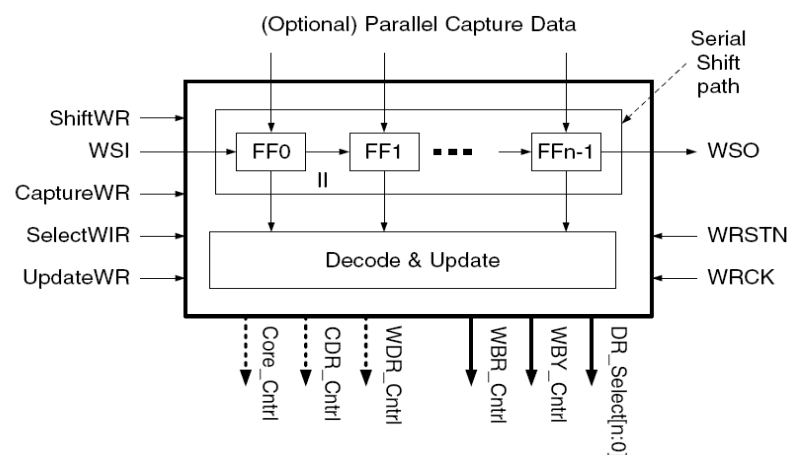
Τα SelectWIR, ShiftWR, CaptureWR και TransferDR πρέπει να δειγματοληπτούνται στη μετάβαση του σήματος χρονισμού WRCK από 0 σε 1, ενώ το σήμα UpdateWR πρέπει να δειγματοληπτείται στη μετάβαση του σήματος χρονισμού WRCK από 1 σε 0.

2. Η παράλληλη πόρτα του κολάρου (wrapper parallel port – WPP)

Η παράλληλη πόρτα είναι προαιρετική και αποτελείται από τους ακροδέκτες παράλληλης εισόδου (wrapper parallel input, WPI, terminals), τους ακροδέκτες παράλληλης εξόδου (wrapper parallel output, WPO, terminals) και τους ακροδέκτες παράλληλου ελέγχου (wrapper parallel control, WPC, terminals).

3. Ένα καταχωρητής εντολών (wrapper instruction register, WIR)

Ο καταχωρητής εντολών χρησιμοποιείται για την αποθήκευση της προς εκτέλεση εντολής. Επιλέγεται όταν η είσοδος SelectWIR έχει τη λογική τιμή 1, άσχετα από την εντολή που εκτελείται. Υλοποιείται χρησιμοποιώντας ένα σχεδιασμό δύο επιπέδων έτσι ώστε η σειριακή φόρτωση μίας νέας εντολής να μην επηρεάζει αυτή που εκτελείται. Υπάρχουν δύο σημαντικές διαφορές μεταξύ του καταχωρητή εντολής του προτύπου 1149.1 και του προτύπου 1500.



Σχήμα 10.2 Δομή του καταχωρητή εντολών WIR.

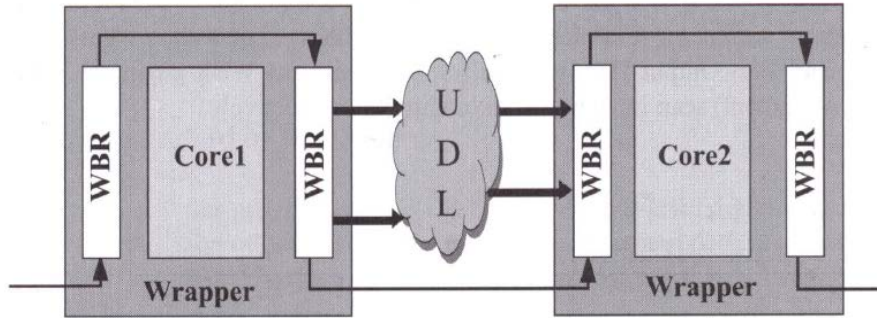
- Επειδή δεν υπάρχει μηχανή πεπερασμένων καταστάσεων στο πρότυπο 1500, οι τιμές των σημάτων ελέγχου εξαρτώνται από την εντολή του καταχωρητή εντολής που εκτελείται και την τρέχουσα τιμή των σημάτων που οδηγούν τις εισόδους ελέγχου της σειριακής πόρτας του κολάρου (wrapper serial control, WSC). Το σχήμα 10.2 δίνει τη δομή του καταχωρητή εντολών WIR. Αποτελείται από μια βαθμίδα ολίσθησης και μια βαθμίδα αποκωδικοποίησης/ενημέρωσης. Τρία είδη βασικών σημάτων (DR_Select, WBY_Cntrl και WBR_Cntrl) χρησιμοποιούνται για την επιλογή των κατάλληλων καταχωρητών, τον έλεγχο των λειτουργιών του καταχωρητή παράκαμψης και τη λειτουργία του περιφερειακού καταχωρητή του κολάρου. Μπορούν επίσης να παραχθούν και σήματα ελέγχου για καταχωρητές δεδομένων του κολάρου (wrapper data registers, WDR), καταχωρητές της προσχεδιασμένης μονάδας (core data registers, CDR) και για την ίδια.
- Εκτός της σειριακής φόρτωσης του καταχωρητή εντολών, το 1500 επιτρέπει και την παράλληλη φόρτωσή του.

4. Ο καταχωρητής παράκαμψης (wrapper bypass register, WBY)

Η λειτουργία αυτού του καταχωρητή είναι όμοια με του αντίστοιχου καταχωρητή στο πρότυπο 1149.1. Ο καταχωρητής αυτός συνδέεται μεταξύ της σειριακής εισόδου WSI και της σειριακής εξόδου WSO του κολάρου όταν η είσοδος WRSTN έχει την ενεργό τιμή ή όταν η εντολή που εκτελείται είναι η εντολή παράκαμψης. Είναι επίσης ο καταχωρητής που συνδέεται μεταξύ της σειριακής εισόδου WSI και της σειριακής εξόδου WSO του κολάρου όταν δεν έχει επιλεχθεί κάποιος καταχωρητής δεδομένων ή όταν στον καταχωρητή εντολών βρίσκεται προς εκτέλεση ένας συνδυασμός λογικών τιμών 0 και 1 που δεν αντιστοιχεί σε κάποια εντολή του κολάρου.

5. Ο περιφερειακός καταχωρητής του κολάρου (wrapper boundary register, WBR)

Ο περιφερειακός καταχωρητής επιτρέπει τη διαφοροποίηση του ελέγχου της ορθής λειτουργίας μιας προσχεδιασμένης μονάδας από τον έλεγχο της ορθής λειτουργίας της εξωτερικής λογικής και των κυκλωμάτων διασύνδεσης. Ο καταχωρητής αυτός λειτουργεί ως ένας μηχανισμός απομόνωσης ο οποίος επιτρέπει δεδομένα δοκιμής να διαδίδονται από τις εισόδους δεδομένων δοκιμής του κολάρου προς τις κύριες εισόδους της προσχεδιασμένης μονάδας και αποκρίσεις της προσχεδιασμένης μονάδας να διαδίδονται από τις κύριες εξόδους της προς τις εξόδους δεδομένων δοκιμής του κολάρου. Οι εισοδοί δεδομένων δοκιμής του περιφερειακού καταχωρητή συνήθως συνδέονται με τον μηχανισμό προσπέλασης για δοκιμή (Test Access Mechanism, TAM), του οποίου το εύρος της αρτηρίας δεδομένων δοκιμής είναι μικρότερο από το πλήθος των εισόδων της υπό έλεγχο προσχεδιασμένης μονάδας. Επομένως ο περιφερειακός καταχωρητής λειτουργεί ως ένα κύκλωμα προσαρμογής της προσχεδιασμένης μονάδας μέσα στο ολοκληρωμένο κύκλωμα. Η παρουσία του περιφερειακού καταχωρητή επιτρέπει επίσης τον έλεγχο της ορθής λειτουργίας της λογικής που βρίσκεται μεταξύ των προσχεδιασμένων μονάδων. Αυτό επιτυγχάνεται καθώς δεδομένα δοκιμής από τον περιφερειακό καταχωρητή του κολάρου μιας προσχεδιασμένης μονάδας οδηγούν την εξωτερική λογική και η απόκριση της εξωτερικής λογικής αποθηκεύεται στον περιφερειακό καταχωρητή του κολάρου μιας άλλης προσχεδιασμένης μονάδας (βλέπε σχήμα 10.3). Μια ακόμη λειτουργία του περιφερειακού καταχωρητή είναι να προστατεύσει την προσχεδιασμένη μονάδα όταν η δραστηριότητα σε επίπεδο ολοκληρωμένου κυκλώματος μπορεί να είναι καταστροφική για την προσχεδιασμένη μονάδα. Επίσης ο περιφερειακός καταχωρητής μπορεί να προσφέρει προστασία στη λογική που βρίσκεται έξω από την προσχεδιασμένη μονάδα που διαθέτει κολάρο, από δεδομένα που παράγονται από αυτή την προσχεδιασμένη μονάδα.

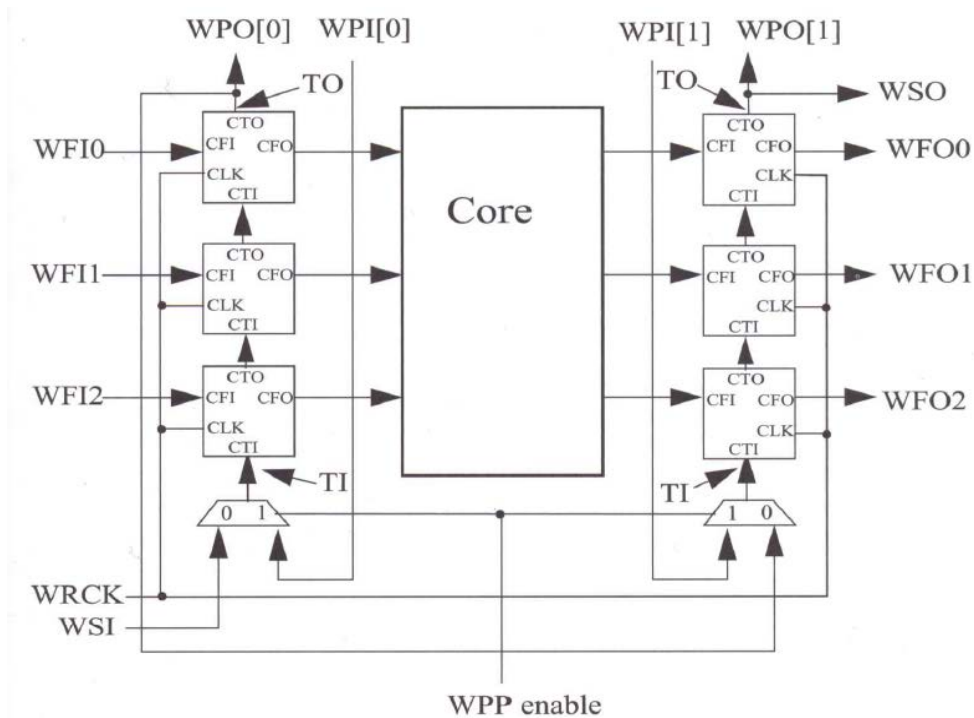


Σχήμα 10.3 Έλεγχος της ορθής λειτουργίας μονάδας που δεν υλοποιεί το πρότυπο.

Ο καταχωρητής αυτός υλοποιείται με έναν αριθμό κυψελίδων (wrapper boundary cells – WBCs). Κάθε κυψελίδα έχει τέσσερις ακροδέκτες: είσοδος λειτουργικών δεδομένων (cell functional input, CFI), έξοδος λειτουργικών δεδομένων (cell functional output, CFO), είσοδος δεδομένων δοκιμής (cell test input – CTI) και έξοδος δεδομένων δοκιμής (cell test output – CTO). Οι κυψελίδες συνδέονται μεταξύ τους για να σχηματίσουν ένα ή περισσότερους καταχωρητές ολίσθησης. Το πρότυπο επιβάλλει την ύπαρξη ενός καταχωρητή ολίσθησης, αλλά επιτρέπει και την υλοποίησή του με τέτοιο τρόπο ώστε να λειτουργεί ως ένα σύνολο καταχωρητών ολίσθησης, με σκοπό τον αυξημένο ρυθμό μεταφοράς δεδομένων δοκιμής προς και από την προσχεδιασμένη μονάδα. Όπως έχουμε ήδη αναφέρει το πρότυπο επιτρέπει την υλοποίηση μιας παράλληλης πόρτας δεδομένων δοκιμής (Wrapper Parallel Port, WPP). Υλοποίηση μιας παράλληλης πόρτας εισόδου δεδομένων δοκιμής συνεπάγεται ότι ο περιφερειακός καταχωρητής μπορεί να λειτουργήσει ως πολλοί παράλληλοι ολισθητές. Στο σχήμα 10.4 μπορούμε να δούμε ότι ο περιφερειακός καταχωρητής, ανάλογα με τη λογική τιμή της εισόδου WPP, μπορεί να λειτουργήσει είτε ως ένας ολισθητής των έξι βαθμίδων είτε ως δύο παράλληλοι ολισθητές των τριών βαθμίδων ο κάθε ένας.

Σύμφωνα με το πρότυπο 1500 η κυψελίδα θα πρέπει:

α. Να συμμετέχει στη λειτουργία ολίσθησης του περιφερειακού καταχωρητή. Αυτό συνεπάγεται ότι θα πρέπει να περιέχει τουλάχιστον ένα στοιχείο μνήμης στο μονοπάτι ολίσθησης του περιφερειακού καταχωρητή. Αυτό εξασφαλίζει ότι δεδομένα δοκιμής θα μπορούν να εφαρμοστούν στον (να παρατηρηθούν από) τον ακροδέκτη της προσχεδιασμένης μονάδας στην οποία έχει συνδεθεί μια κυψελίδα του περιφερειακού καταχωρητή.



Σχήμα 10.4 Περιφερειακός καταχωρητής με σειριακή και παράλληλη πόρτα.

β. Να περιέχει ένα στοιχείο μνήμης για τη σύλληψη (capture) δεδομένων στο περιφερειακό καταχωρητή. Αυτό εξασφαλίζει ότι η απόκριση της υπό έλεγχο μονάδας μπορεί να φορτωθεί στον περιφερειακό καταχωρητή και στη συνέχεια να μεταφερθεί σειριακά έξω προς παρατήρηση. Το στοιχείο μνήμης που χρησιμοποιείται για τη σύλληψη της πληροφορίας μπορεί να συμμετέχει στη λειτουργία ολίσθησης αλλά μπορεί και όχι.

Οι ανωτέρω απαιτήσεις του προτύπου συνεπάγονται ότι η κυψελίδα πρέπει να έχει ένα μονοπάτι ολίσθησης το οποίο να είναι διαφορετικό από το λειτουργικό μονοπάτι.

Ο ορισμός της κυψελίδα σύμφωνα με το πρότυπο 1500 είναι αρκετά ευέλικτος, αν το συγκρίνουμε με την κυψελίδα που χρησιμοποιείται στο πρότυπο 1149.1. Στόχος είναι να μπορούμε να εφαρμόσουμε μια ποικιλία από ελέγχους ορθής λειτουργίας πχ. ανίχνευση απλών σφαλμάτων μόνιμης τιμής, ανίχνευση σφαλμάτων καθυστέρησης κ.λπ. Η ευελιξία της σχεδίασης της κυψελίδας καθιστά αδύνατη την παρουσίαση όλων των κυψελίδων που ικανοποιούν το πρότυπο. Για το λόγο αυτό πιο κάτω θα δώσουμε μόνο μερικά παραδείγματα κυψελίδων που ικανοποιούν το πρότυπο.

Τρόποι λειτουργίας του περιφερειακού καταχωρητή του κολάρου :

- **Κανονική λειτουργία (Normal mode)**
Ο περιφερειακός καταχωρητής του κολάρου είναι διαφανής στο σύστημα και η προσχεδιασμένη μονάδα εκτελεί τις κανονικές της λειτουργίες.
- **Έλεγχος της ορθής λειτουργίας της προσχεδιασμένης μονάδας (Inward facing mode)**
Μεταφορά δεδομένων δοκιμής για τον έλεγχο μιας προσχεδιασμένης μονάδας. Οι κύριες εισοδοί της προσχεδιασμένης μονάδας ελέγχονται από τον περιφερειακό καταχωρητή του κολάρου και οι κύριες έξοδοί της

προσχεδιασμένης μονάδας παρατηρούνται δια του περιφερειακού καταχωρητή.

- **Έλεγχος της ορθής λειτουργίας εξωτερικής λογικής, ως προς τις προσχεδιασμένες μονάδες που υλοποιούν το πρότυπο, (Outward facing mode)**

Μεταφορά δεδομένων δοκιμής για τον έλεγχο κυκλώματος που βρίσκεται εκτός των προσχεδιασμένων μονάδων που υλοποιούν το πρότυπο.

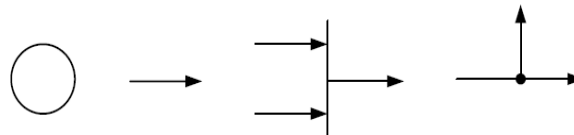
- **Εξασφάλιση ασφαλών εξόδων**

Όταν το κολάρο βρίσκεται σ' αυτό τον τρόπο λειτουργίας, τότε η προσχεδιασμένη μονάδα δεν επηρεάζει την εξωτερική λογική την οποία οδηγεί. Το κολάρο εξάγει ασφαλή δεδομένα και η προσχεδιασμένη μονάδα θα πρέπει να καταναλώνει όση λιγότερη ενέργεια γίνεται. Τα ασφαλή δεδομένα έχουν ως σκοπό να εμποδίσουν ανεπιθύμητα συμβάντα όπως η διαμάχη για την αρτηρία και η δημιουργία υπερβολικού θορύβου.

Στοιχειώδεις λειτουργίες της κυψελίδας και του περιφερειακού καταχωρητή του κολάρου

Η συμπεριφορά της κυψελίδας WBC και του περιφερειακού καταχωρητή WBR εκφράζεται σε στοιχειώδεις λειτουργίες (events), οι οποίες καθορίζονται από τις εισόδους ελέγχου της σειριακής WSC ή της παράλληλης WPC πόρτας του κολάρου και ενεργοποιούνται από το σήμα χρονισμού WRCK.

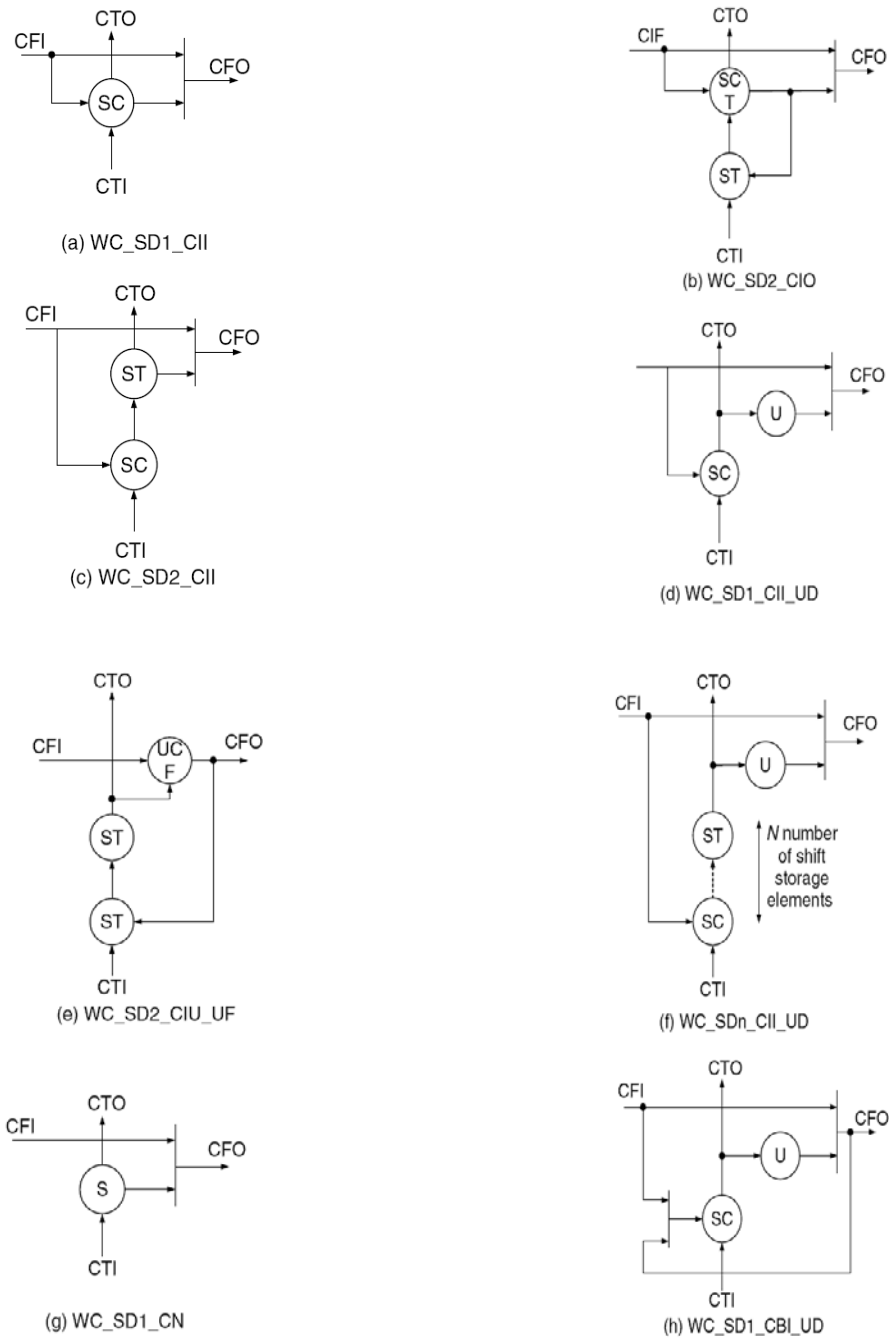
Ο περιφερειακός καταχωρητής (ή η κυψελίδα) υποστηρίζει πέντε στοιχειώδεις λειτουργίες: ολίσθηση (shift), σύλληψη (capture), ενημέρωση (update), μεταφορά (transfer) και εφαρμογή (apply). Τα σύμβολα που δίνονται στο σχήμα 10.5 χρησιμοποιούνται για να αναπαραστήσουν τη δομή μιας κυψελίδας.



Σχήμα 10.5 Σύμβολα περιγραφής στοιχειωδών λειτουργιών

Ο κύκλος συμβολίζει ένα στοιχείο μνήμης, το βέλος συμβολίζει ένα μονοπάτι δεδομένων, η κάθετη γραμμή με δύο ή περισσότερα βέλη εισόδου και ένα βέλος εξόδου συμβολίζει ένα πολυπλέκτη και δυο βέλη τα οποία ξεκινούν από το ίδιο σημείο συμβολίζουν δύο μονοπάτια δεδομένων που έχουν την ίδια πηγή. Κάθε κύκλος μπορεί να χαρακτηριστεί με ένα ή περισσότερους χαρακτήρες από το σύνολο S, C,U,T και F για να δείξει ότι το αντίστοιχο στοιχείο μνήμης είναι υπεύθυνο για τη στοιχειώδη λειτουργία ολίσθησης (shift), σύλληψης (capture), ενημέρωσης (update), μεταφοράς (transfer) και κανονικής λειτουργίας της προσχεδιασμένης μονάδας αντίστοιχα.

Στο σχήμα 10.6 δίνεται η δομή διάφορων κυψελίδων που είναι σύμφωνες με το πρότυπο 1500. Οι στοιχειώδεις λειτουργίες που υποστηρίζονται από κάθε κυψελίδα δηλώνονται με τους αντίστοιχους χαρακτήρες στα αντίστοιχα στοιχεία μνήμης. Ως παράδειγμα αναφέρουμε ότι το σχήμα 10.6.a δείχνει μια απλή κυψελίδα η οποία περιέχει μόνο ένα στοιχείο μνήμης και υποστηρίζει μόνο τις στοιχειώδεις λειτουργίες ολίσθησης και σύλληψης. Το σχήμα 10.6. d



Σχήμα 10.6 Η συμβολική δομή διάφορων κυψελίδων που μπορούν να χρησιμοποιηθούν για την υλοποίηση του περιφερειακού καταχωρητή του πρότυπου 1500.

δείχνει μια κυψελίδα η οποία περιέχει δύο στοιχεία μνήμης, εκ των οποίων το ένα υποστηρίζει τις στοιχειώδεις λειτουργίες ολίσθησης και σύλληψης και το άλλο τη στοιχειώδη λειτουργία της ενημέρωσης. Η κυψελίδα αυτή έχει τη δομή της κυψελίδας του σχήματος 9.5 που χρησιμοποιείται για την υλοποίηση του περιφερειακού καταχωρητή του προτύπου 1149.1. Η κυψελίδα του σχήματος 10.6.e αποτελείται από τρία στοιχεία μνήμης εκ των οποίων τα δύο χρησιμοποιούνται για τις στοιχειώδεις λειτουργίες της ολίσθησης και μεταφοράς και το άλλο για τη στοιχειώδη λειτουργία της ενημέρωσης και σύλληψης και την κανονική λειτουργία.

Οι στοιχειώδεις λειτουργίες περιγράφονται ως εξής:

- **Ολίσθηση.**
Είναι μια υποχρεωτική στοιχειώδης λειτουργία δια της οποίας τα δεδομένα που είναι αποθηκευμένα στο μονοπάτι ολίσθησης του περιφερειακού καταχωρητή ολισθαίνουν κατά μια θέση προς την έξοδο. Τα δεδομένα που βρίσκονται στη σειριακή είσοδο του περιφερειακού καταχωρητή ολισθαίνουν κατά μία θέση μέσα στο μονοπάτι ολίσθησης του περιφερειακού καταχωρητή.
- **Σύλληψη.**
Είναι μια στοιχειώδης λειτουργία δια της οποίας τα δεδομένα που υπάρχουν στην είσοδο CFI ή την έξοδο CFO της κυψελίδας αποθηκεύονται σε ένα στοιχείο μνήμης της κυψελίδας. Ως παράδειγμα αναφέρουμε την κυψελίδα του σχήματος 10.6.a που λαμβάνει προς αποθήκευση δεδομένα από την είσοδο CFI καθώς και την κυψελίδα του σχήματος 10.6.h που μπορεί να λάβει προς αποθήκευση δεδομένα τόσο από την είσοδο CFI όσο και από την έξοδο CFO. Αυτή η στοιχειώδης λειτουργία είναι υποχρεωτική για όλες τις κυψελίδες, εκτός από αυτές που συνδέονται σε έναν ακροδέκτη που εξαιρείται του κολάρου, όπως για παράδειγμα ένας ακροδέκτης χρονισμού ή ένας ακροδέκτης αρχικοποίησης.
- **Ενημέρωση**
Επιπλέον του μονοπατιού ολίσθησης ο περιφερειακός καταχωρητής μπορεί προαιρετικά να περιέχει και ένα καταχωρητή παράλληλης ενημέρωσης. Αυτός ο καταχωρητής υλοποιείται με στοιχεία μνήμης τα οποία δεν ανήκουν στο μονοπάτι ολίσθησης του περιφερειακού καταχωρητή. Αυτός ο καταχωρητής επιτρέπεται να κάνει σύλληψη δεδομένων από τις εισόδους CFI των κυψελίδων. Ο καταχωρητής παράλληλης ενημέρωσης μπορεί να υλοποιείται με στοιχεία μνήμης της προσχεδιασμένης μονάδας. Όταν ο περιφερειακός καταχωρητής περιέχει ένα προαιρετικό καταχωρητή παράλληλης ενημέρωσης, τότε υποστηρίζεται και η προαιρετική στοιχειώδης λειτουργία της ενημέρωσης. Ως παράδειγμα αναφέρουμε την κυψελίδα του σχήματος 10.6. d στην οποία μπορεί να γίνει η στοιχειώδης λειτουργία της ενημέρωσης.
- **Μεταφορά**
Είναι μια προαιρετική στοιχειώδης λειτουργία. Μετακινεί δεδομένα είτε προς το στοιχείο μνήμης που βρίσκεται πιο κοντά στην είσοδο CFI μιας κυψελίδας, εάν τα δεδομένα που αποθηκεύτηκαν με μια στοιχειώδη λειτουργία σύλληψης δεν βρίσκονται ήδη σ' αυτό το στοιχείο μνήμης, ή μετακινεί δεδομένα κατά μια θέση πιο κοντά προς την έξοδο CTO της κυψελίδας. Αυτή η στοιχειώδης λειτουργία μπορεί να εκτελεστεί για δύο λόγους.
 - α. Για να αποθηκευτούν σε στοιχεία μνήμης της κυψελίδας, εκτελώντας την στοιχειώδη λειτουργία της σύλληψης, τόσες τιμές όσα είναι τα στοιχεία μνήμης που ανήκουν στο μονοπάτι ολίσθησης της κυψελίδας, θα πρέπει τα δεδομένα να εισάγονται στο στοιχείο μνήμης που βρίσκεται πιο κοντά στην είσοδο CFI της κυψελίδας.
 - β. Για την ανίχνευση σφαλμάτων καθυστέρησης απαιτείται να υπάρχει η δυνατότητα να τροφοδοτήσουμε την υπό έλεγχο προσχεδιασμένη μονάδα με δύο διαδοχικά διανύσματα δοκιμής. Η στοιχειώδης λειτουργία της μεταφοράς

μετακινεί δεδομένα διαμέσου του μονοπατιού ολίσθησης έτσι ώστε δύο διαδοχικά δυαδικά ψηφία στο μονοπάτι ολίσθησης της κυψελίδας να μην μπορούν να φορτωθούν το ένα μετά το άλλο στο στοιχείο μνήμης ενημέρωσης και να οδηγήσουν την έξοδο CTO της κυψελίδας.

10.4 Σύνολο Εντολών του πρότυπου 1500

Το σύνολο εντολών που ορίζεται στο πρότυπο περιέχει υποχρεωτικές και προαιρετικές εντολές. Οι υποχρεωτικές εντολές πρέπει να υλοποιούνται πάντοτε και επιτρέπουν την κανονική λειτουργία της προσχεδιασμένης μονάδας, τον έλεγχο της ορθής λειτουργίας της προσχεδιασμένης μονάδας και το έλεγχο της ορθής λειτουργίας της εξωτερικής, ως προς τις προσχεδιασμένες μονάδες, λογικής. Οι προαιρετικές εντολές που ορίζονται προσφέρουν μηχανισμούς για πρόσθετες δυνατότητες αλλάζοντας τη διαμόρφωση και τις δυνατότητες των καταχωρητών δεδομένων του κολάρου. Το πρότυπο επιτρέπει επίσης ένα σύνολο εντολών να οριστεί από τον σχεδιαστή. Ο κωδικός λειτουργίας ακόμη και των υποχρεωτικών εντολών αφήνεται να επιλεγεί από τον σχεδιαστή.

Το σύνολο εντολών στο προτύπο 1500 αποτελείται από:

- σειριακές εντολές (τα δεδομένα χρησιμοποιούν μόνο ένα σειριακό μονοπάτι από τη σειριακή είσοδο WSI προς τη σειριακή έξοδο WSO του κολάρου και τα σήματα ελέγχου της σειριακής πόρτας)
- παράλληλες εντολές (τα δεδομένα μπορούν να χρησιμοποιήσουν ένα μηχανισμό προσπέλασης δεδομένων δοκιμής (test access mechanism, TAM) με εύρος από 0 έως n δυαδικά ψηφία μαζί με σήματα ελέγχου που ορίζονται από το σχεδιαστή)
- υβριδικές εντολές (χρησιμοποιείται ένας συνδυασμός από ακροδέκτες για σειριακές και παράλληλες εντολές)

Ο τρόπος ονομασίας των εντολών ακολουθεί κάποιες συμβάσεις ώστε από το όνομα της εντολής να συμπεραίνουμε τον τρόπο λειτουργίας της. Μια εντολή έχει την ακόλουθη μορφή:

W<S/P/H>_<εντολή> {<διαμόρφωση>}

Παρατηρούμε ότι το όνομα της εντολής αποτελείται από τρία τμήματα τα οποία χωρίζονται μεταξύ τους με το σύμβολο «_».

Το πρώτο τμήμα περιέχει δύο χαρακτήρες. Ο πρώτος χαρακτήρας είναι πάντοτε το W και δηλώνει ότι η εντολή είναι μια εντολή του προτύπου 1500. Ο δεύτερος χαρακτήρας είναι το

S, το P, ή το H και δηλώνει ότι η εντολή είναι σειριακή, παράλληλη ή υβριδική αντίστοιχα. Μια σειριακή εντολή κατά τον έλεγχο της ορθής λειτουργίας χρησιμοποιεί μόνο τη σειριακή πόρτα WSP. Μια παράλληλη εντολή κατά τον έλεγχο της ορθής λειτουργίας χρησιμοποιεί μόνο την παράλληλη πόρτα WPP (με εξαίρεση την εντολή WP_EXTEST), τον ακροδέκτη χρονισμού WRCK και τα βοηθητικά σήματα χρονισμού της σειριακής πόρτας. Η υβριδική εντολή χρησιμοποιεί ένα συνδυασμό από ακροδέκτες της σειριακής πόρτας και της παράλληλης πόρτας για να προσπελάσει καταχωρητές δεδομένων του κολάρου και να τρέξει τον έλεγχο ορθής λειτουργίας. Ο καταχωρητής εντολής WRI και ο καταχωρητής παράκαμψης WBY

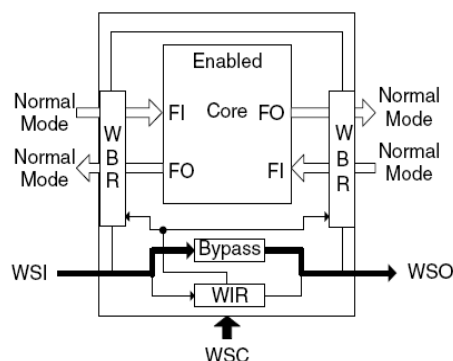
μπορούν να προσπελαστούν μόνο από τη σειριακή πόρτα του κολάρου. Εάν ο δεύτερος χαρακτήρας στο όνομα της εντολής είναι το X, αυτό δηλώνει ότι η εντολή μπορεί να είναι είτε σειριακή είτε παράλληλη είτε υβριδική.

Το δεύτερο τμήμα του ονόματος της εντολής δηλώνει την κύρια λειτουργία της εντολής. Για παράδειγμα η εντολή WS_BYPASS δηλώνει ότι ο περιφερειακός καταχωρητής και οποιοσδήποτε άλλος καταχωρητής δεδομένων παρακάμπτεται.

Το τρίτο τμήμα του ονόματος της εντολής είναι προαιρετικό και χρησιμοποιείται για να προσφέρει κάποια επιπλέον περιγραφή ή διαφοροποίηση από άλλη εντολή. Αυτό το τρίτο τμήμα χρησιμοποιείται για να διαφοροποιήσει, για παράδειγμα, τις εντολές WS_INTEST_RING και WS_INTEST_SCAN. Με τις εντολές αυτές θα ασχοληθούμε πιο κάτω.

- **WS_BYPASS**

Είναι μια υποχρεωτική εντολή. Όλα τα λειτουργικά σήματα περνούν διαμέσου του περιφερειακού καταχωρητή ως εάν να μην υπήρχε. Η προσχεδιασμένη μονάδα τίθεται σε κανονικό τρόπο λειτουργίας. Μεταξύ της σειριακής εισόδου δεδομένων δοκιμής WSI και της σειριακής εξόδου δεδομένων δοκιμής WSO τοποθετείται ο καταχωρητής παράκαμψης. Ενώ η προσχεδιασμένη μονάδα βρίσκεται σε κανονική λειτουργία ο καταχωρητής παράκαμψης μπορεί να χρησιμοποιηθεί. Η εκτέλεση της εντολής αυτής περιγράφεται στο σχήμα 10.7.

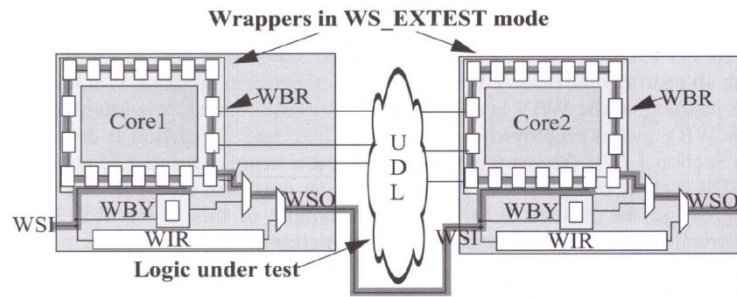


Σχήμα 10.7 Εντολή WS_BYPASS

- **WS_EXTEST**

Η εντολή αυτή είναι υποχρεωτική και επιτρέπει τον έλεγχο της ορθής λειτουργίας των διασυνδέσεων μεταξύ δύο προσχεδιασμένων μονάδων καθώς και κυκλωμάτων που βρίσκονται έξω από τις προσχεδιασμένες μονάδες που υλοποιούν το πρότυπο (user defined logic, UDL). Οι τιμές που βρίσκονται στις λειτουργικές εξόδους CFO των κυψελίδων του περιφερειακού καταχωρητή τροφοδοτούν την εξωτερική λογική ή τις γραμμές διασύνδεσης και η απόκριση της εξωτερικής λογικής ή των γραμμών διασύνδεσης με τη λειτουργία της σύλληψης αποθηκεύονται μέσω των λειτουργικών εισόδων

CFI των κυψελίδων στον περιφερειακό καταχωρητή. Η εκτέλεση της εντολής αυτής περιγράφεται στο σχήμα 10.8.



Σχήμα 10.8 Εντολή WS_EXTEST

- **WP_EXTEST**

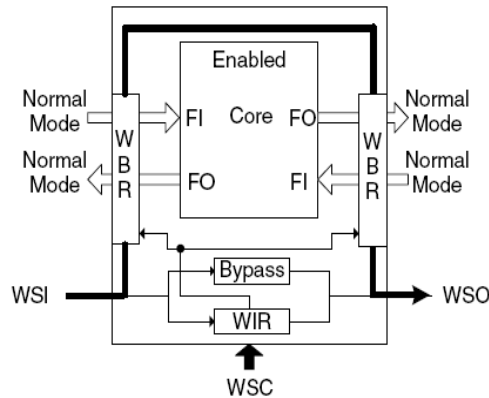
Αυτή η προαιρετική εντολή επιτρέπει τον έλεγχο της ορθής λειτουργίας των διασυνδέσεων μεταξύ προσχεδιασμένων μονάδων καθώς και κυκλωμάτων που βρίσκονται μεταξύ των προσχεδιασμένων μονάδων χρησιμοποιώντας τον περιφερειακό καταχωρητή σαν να αποτελείται από πολλά τμήματα τα οποία προσπελούνται παράλληλα. Είναι η μόνη παράλληλη εντολή που χρησιμοποιεί το σήμα χρονισμού WSC της σειριακής πόρτας WSP.

- **WH_EXTEST**

Αυτή η προαιρετική εντολή επιτρέπει τον έλεγχο της ορθής λειτουργίας των διασυνδέσεων μεταξύ προσχεδιασμένων μονάδων καθώς και κυκλωμάτων που βρίσκονται μεταξύ των προσχεδιασμένων μονάδων. Η διαφορά σε σχέση με τις εντολές WS_EXTEST και WP_EXTEST είναι ότι κάποια τμήματα του περιφερειακού καταχωρητή προσπελούνται μέσω της σειριακής πόρτας και κάποια μέσω της παράλληλης πόρτας.

- **WS_PRELOAD**

Η εντολή αυτή είναι προαιρετική και επιτρέπει το μονοπάτι ολίσθησης του περιφερειακού καταχωρητή να γεμίσει σειριακά δεδομένα δοκιμής πριν ξεκινήσει ο έλεγχος ορθής λειτουργίας. Αυτή η εντολή είναι υποχρεωτική όταν στον περιφερειακό καταχωρητή υπάρχει ένα διακριτό μονοπάτι ολίσθησης. Η κυψελίδα, για παράδειγμα, του σχήματος 10.6.d διαθέτει ένα διακριτό στοιχείο μνήμης για ολίσθηση. Η εκτέλεση της εντολής αυτής περιγράφεται στο σχήμα 10.9.



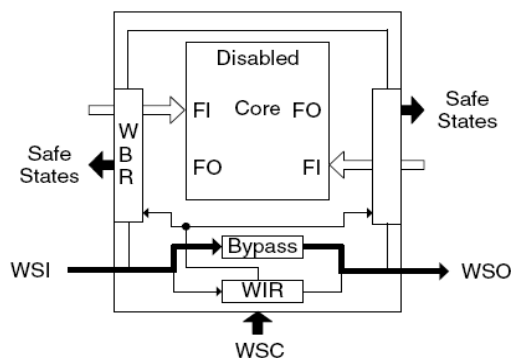
Σχήμα 10.9 Εντολή WS_PRELOAD

- **WP_PRELOAD**

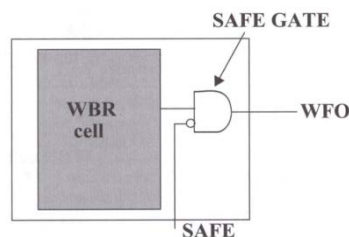
Η εντολή αυτή εξυπηρετεί τον ίδιο σκοπό με την εντολή WS_PRELOAD, μόνο που χρησιμοποιεί την παράλληλη πόρτα WPP. Σκοπός της εντολής αυτής είναι να επιτρέψει την προσπέλαση πολλών μονοπατιών ολίσθησης ταυτόχρονα για να κάνει τη διαδικασία προφόρτωσης δεδομένων δοκιμής πιο γρήγορη.

- **WS_SAFE**

Αυτή η προαιρετική εντολή προσφέρει στο σχεδιαστή του συστήματος τη δυνατότητα να θέσει τις εξόδους του κολάρου μιας προσχεδιασμένης μονάδας σε επιθυμητές σταθερές τιμές (βλέπε σχήμα 10.10). Σύμφωνα με το πρότυπο κάθε έξοδος πρέπει να έχει μια πύλη ώστε όταν ενεργοποιείται να εξασφαλίζεται η επιθυμητή ασφαλής τιμή. Το σχήμα 10.11 δείχνει ένα παράδειγμα υλοποίησης κυψελίδας.



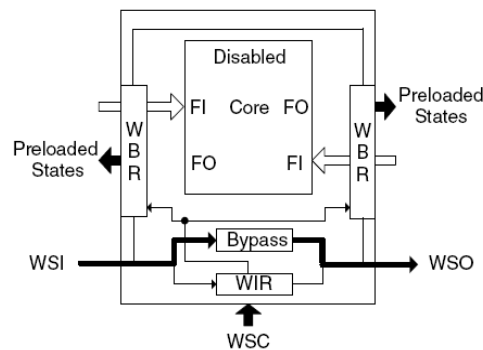
Σχήμα 10.10 Εντολή WS_SAFE.



Σχήμα 10.11 Παράδειγμα υλοποίησης κυψελίδας για την υλοποίηση της προαιρετικής εντολής WS_SAFE.

- **WS_CLAMP**

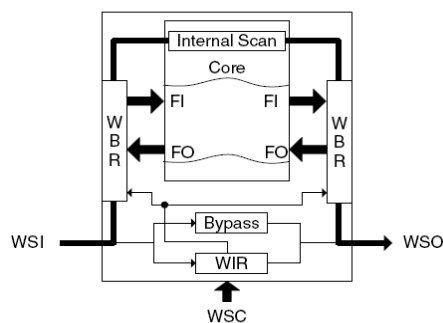
Προαιρετική εντολή που επιτρέπει οι τιμές των λειτουργικών ακροδεκτών εξόδου του κολάρου να λάβουν τις τιμές που είχαν αποθηκευτεί στον περιφερειακό καταχωρητή. Η εντολή αυτή μπορεί να χρησιμοποιηθεί μετά από την εντολή WS_PRELOAD ή WP_PRELOAD. Η εντολή WS_CLAMP μπορεί να χρησιμοποιηθεί σε συνδυασμό με την εντολή Wx_PRELOAD για να εξυπηρετήσει τον ίδιο σκοπό με την εντολή WS_SAFE, με μόνη διαφορά ότι επιτρέπει οποιαδήποτε τιμή να αποθηκευτεί σειριακά στον περιφερειακό καταχωρητή και να εμφανισθεί στις εξόδους του. Αυτό προσφέρει μεγαλύτερη ευελιξία σε σύγκριση με την εντολή WS_SAFE. Ωστόσο η εντολή WS_SAFE προσφέρει έναν αρκετά λιγότερο πολύπλοκο τρόπο για να θέσουμε τις εξόδους του περιφερειακού καταχωρητή σε ασφαλείς τιμές, αρκεί βέβαια να μας αρκεί ένας συνδυασμός τιμών. Η εκτέλεση της εντολής αυτής περιγράφεται στο σχήμα 10.12.



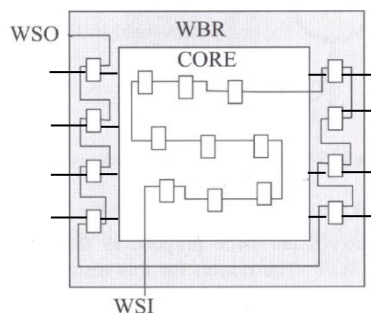
Σχήμα 10.12 Εντολή WS_CLAMP

- **WS_INTEST_SCAN**

Αυτή η εντολή συνδέει σειριακά τον περιφερειακό καταχωρητή και τα εσωτερικά μονοπάτια ολίσθησης (scan chains) της προσχεδιασμένης μονάδας. Ο περιφερειακός καταχωρητής είναι σε κατάσταση ελέγχου της ορθής λειτουργίας της προσχεδιασμένης μονάδας (inward facing mode). Χρησιμοποιείται για τον έλεγχο της ορθής λειτουργίας μιας προσχεδιασμένης μονάδας που υλοποιεί το πρότυπο. Η εκτέλεση της εντολής αυτής περιγράφεται στο σχήμα 10.13. Στο σχήμα 10.14 δίνεται ένα παράδειγμα.



Σχήμα 10.13 Εντολή WS_INTEST_SCAN

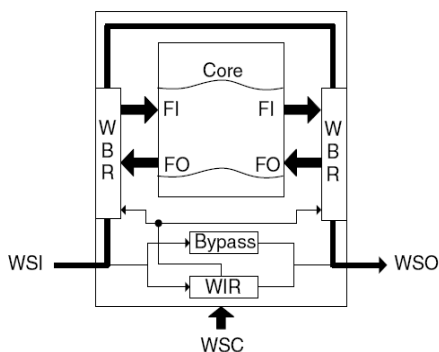


Σχήμα 10.14 Παράδειγμα υλοποίησης της εντολής WS_INTEST_SCAN

- **WS_INTEST_RING**

Η διαφορά της εντολής αυτής σε σχέση με την εντολή WS_INTEST_SCAN είναι ότι

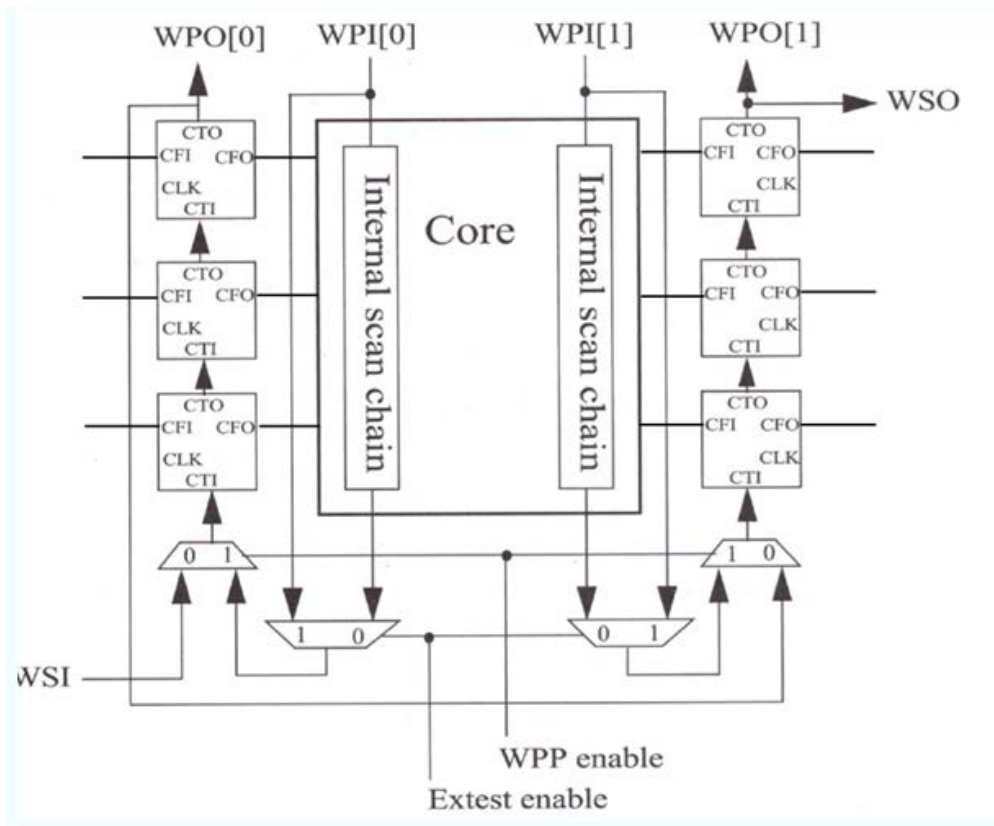
τα εσωτερικά μονοπάτια ολίσθησης (scan chains) δεν συμμετέχουν στη διαδικασία ελέγχου ορθής λειτουργίας της προσχεδιασμένης μονάδας που υλοποιεί το πρότυπο. Μια πιθανή χρήση αυτής της εντολής είναι όταν η προσχεδιασμένη μονάδα δεν διαθέτει μονοπάτια ολίσθησης και για τον έλεγχο της ορθής της λειτουργίας χρησιμοποιούνται λειτουργικά διανύσματα εισόδου. Η εκτέλεση της εντολής αυτής περιγράφεται στο σχήμα 10.15.



Σχήμα 10.15 Εντολή WS_INTEST_RING

- **WP_INTEST**

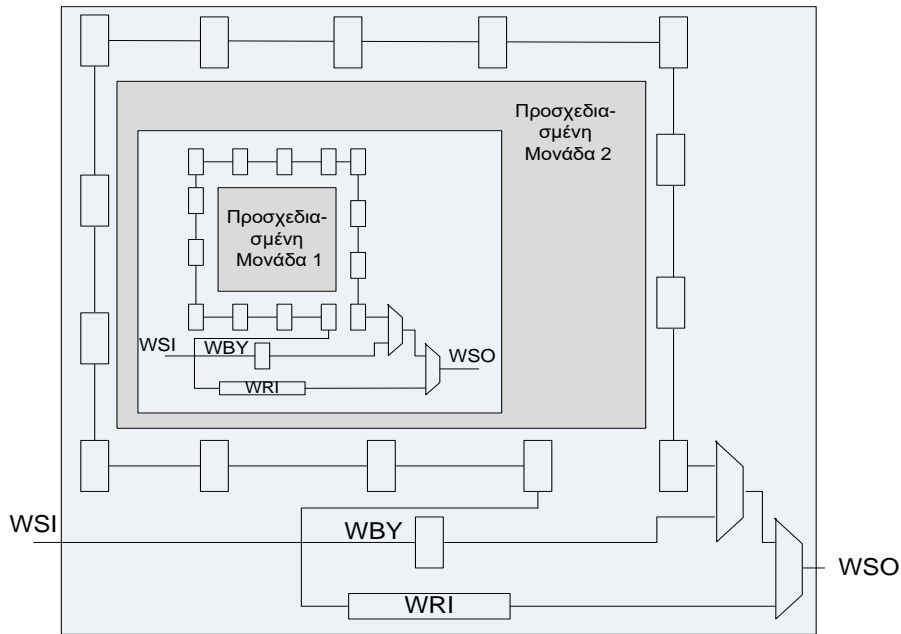
Και αυτή η εντολή χρησιμοποιείται για τον έλεγχο της ορθής λειτουργίας μιας προσχεδιασμένης μονάδας. Εάν η παράλληλη πόρτα του κολάρου έχει πολλούς ακροδέκτες εισόδου δεδομένων δοκιμής, αυτή η εντολή οδηγεί σε σημαντική μείωση του χρόνου ελέγχου της ορθής λειτουργίας της προσχεδιασμένης μονάδας. Ένα παράδειγμα υλοποίησης αυτής της εντολής δίνεται στο σχήμα 10.16.



Σχήμα 10.16. Παράδειγμα υλοποίησης της εντολής WP_INTEST

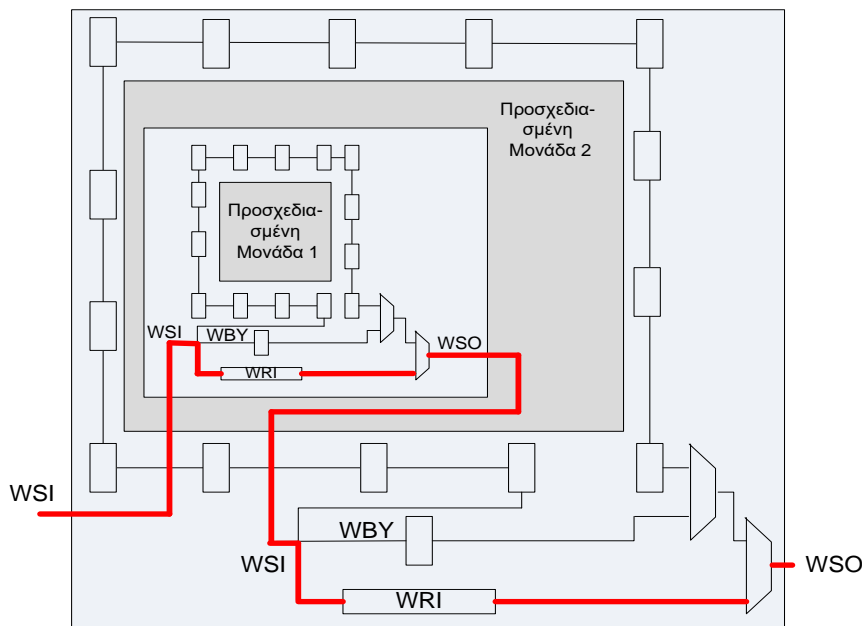
10.5 Προσχεδιασμένες μονάδες εμφωλευμένες μέσα σε προσχεδιασμένες μονάδες (Hierarchical cores)

Το πρότυπο 1500 μπορεί να χρησιμοποιηθεί και στην περίπτωση προσχεδιασμένων μονάδων που είναι εμφωλευμένες μέσα σε άλλες προσχεδιασμένες μονάδες (βλέπε σχήμα 10.17). Σ' αυτή την περίπτωση ο καταχωρητής εντολών του κολάρου καθεμιάς από τις προσχεδιασμένες μονάδες πρέπει να είναι ελέγξιμος από τη σειριακή πόρτα του κολάρου του. Ένας τρόπος για να το πετύχουμε είναι να συνδέσουμε σειριακά τους καταχωρητές εντολών των δύο προσχεδιασμένων μονάδων και να τους ελέγχουμε μέσω της σειριακής πόρτας του κολάρου της εξωτερικής μονάδας (βλέπε σχήμα 10.18).

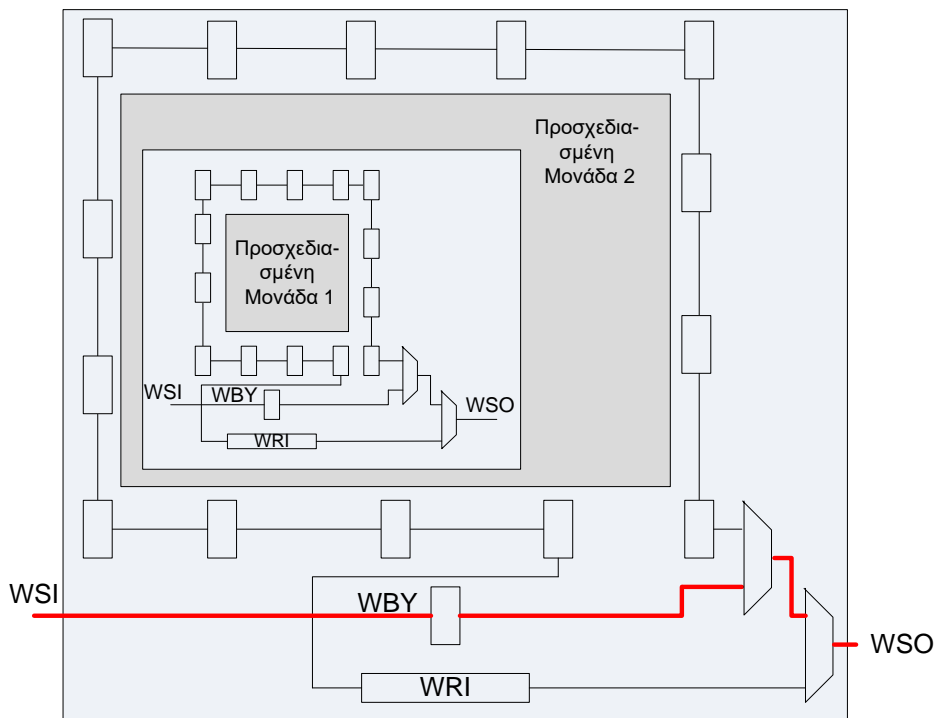


Σχήμα 10.17 Προσχεδιασμένη μονάδα εμφωλευμένη μέσα σε άλλη.

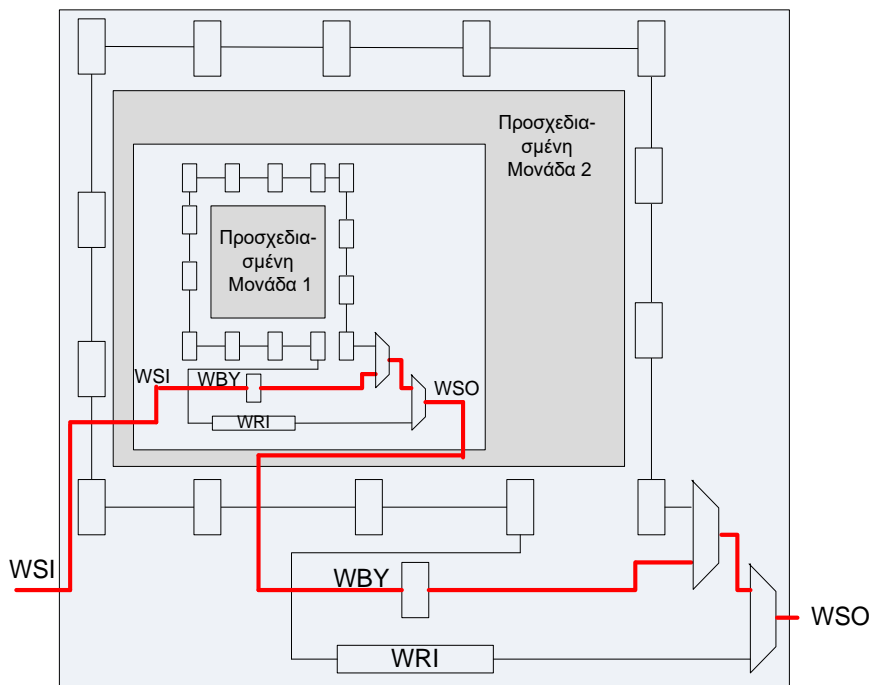
Υπάρχουν δύο δυνατοί τρόποι για να διαμορφώσουμε ένα μονοπάτι παράκαμψης στην περίπτωση μιας προσχεδιασμένης μονάδας που είναι εμφωλευμένη μέσα σε άλλη. Σύμφωνα με τον ένα τρόπο το μονοπάτι παράκαμψης σχηματίζεται από τον καταχωρητή παράκαμψης της εξωτερικής μονάδας. Αυτό επιτυγχάνεται εκτελώντας στην εξωτερική μονάδα την εντολή `WS_BYPASS` ή κάποια άλλη εντολή που συνδέει μεταξύ της σειριακής εισόδου του και της σειριακής του εξόδου τον καταχωρητή παράκαμψης (βλέπε σχήμα 10.19). Στη δεύτερη περίπτωση ο καταχωρητής παράκαμψης της εμφωλευμένης προσχεδιασμένης μονάδας συνδέεται σειριακά με το καταχωρητή παράκαμψης της εξωτερικής προσχεδιασμένης μονάδας (βλέπε σχήμα 10.20).



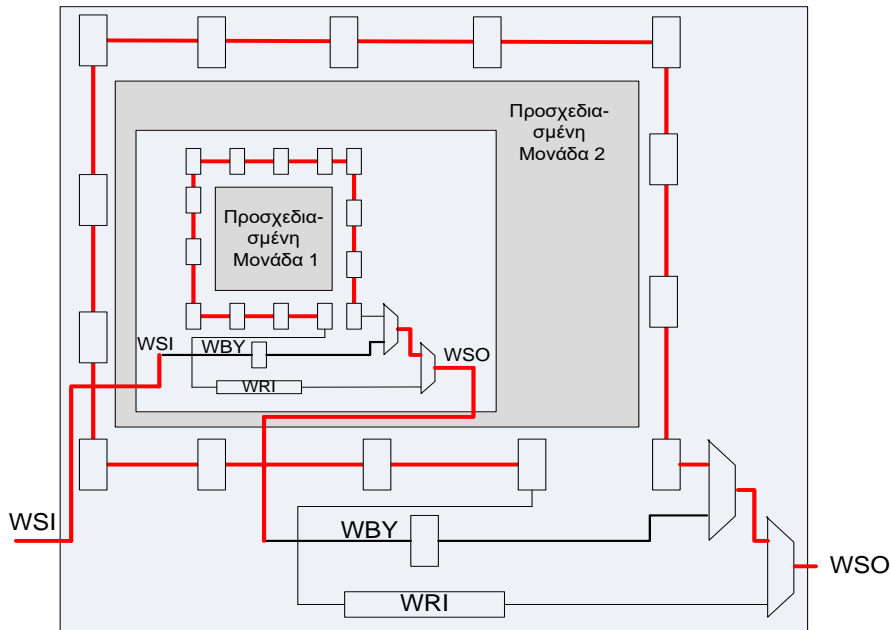
Σχήμα 10.18 Παράδειγμα προσπέλασης των καταχωρητών εντολών στην περίπτωση προσχεδιασμένης μονάδας εμφωλευμένη μέσα σε άλλη.



Σχήμα 10.19 Εκτέλεση της εντολής WS_BYPASS στην εξωτερική προσχεδιασμένη μονάδα.



Σχήμα 10.20 Εκτέλεση της εντολής WS_BYPASS και στις δύο μονάδες.



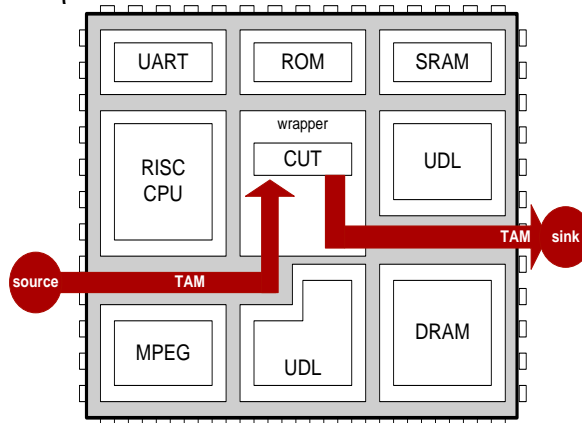
Σχήμα 10.21 Σύνδεση στη σειρά των περιφερειακών καταχωρητών.

Στο σχήμα 10.21 φαίνεται πως μπορούν να συνδεθούν οι περιφερειακοί καταχωρητές των δύο προσχεδιασμένων μονάδων ώστε το περιεχόμενο του περιφερειακού καταχωρητή της εμφωλευμένης μονάδας να είναι ελέγξιμο και παρατηρήσιμο.

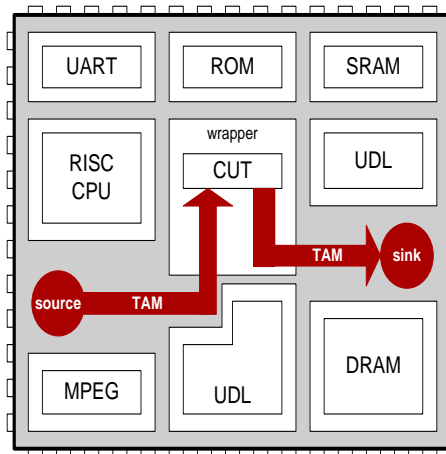
10.6 Χρησιμοποίηση του προτύπου 1500 σε επίπεδο ολοκληρωμένου κυκλώματος

Στις προηγούμενες ενότητες μιλήσαμε για τον τρόπο με τον οποίο μπορούν να χρησιμοποιηθούν δεδομένα δοκιμής που έχουν φτάσει στο κολάρο μιας προσχεδιασμένης μονάδας, η οποία υλοποιεί το πρότυπο, για τον έλεγχο της ορθής λειτουργίας αυτής της μονάδας ή τον έλεγχο της ορθής λειτουργίας γραμμών διασύνδεσης που οδηγούνται από αυτή τη μονάδα ή μονάδων που οδηγούνται από αυτή τη μονάδα και δεν υλοποιούν το πρότυπο. Είπαμε επίσης ότι μια μονάδα που υλοποιεί το πρότυπο θα διαθέτει υποχρεωτικά μια σειριακή πόρτα δεδομένων δοκιμής και προαιρετικά παράλληλη πόρτα δεδομένων δοκιμής. Το ερώτημα που τίθεται τώρα είναι πως τα δεδομένα δοκιμής θα φτάσουν στη σειριακή πόρτα ή στην παράλληλη πόρτα δεδομένων δοκιμής, εάν αυτή υπάρχει. Λόγω της ύπαρξης της σειριακής πόρτας, οι αρχιτεκτονικές που αναφέραμε στην περίπτωση του προτύπου 1149.1 μπορούν να εφαρμοστούν σε επίπεδο ολοκληρωμένου κυκλώματος και τώρα. Ωστόσο επειδή επιτρέπεται η ύπαρξη και παράλληλης πόρτας δεδομένων δοκιμής είναι χρήσιμο να την υλοποιούμε και να τη χρησιμοποιούμε, διότι κάνει τον έλεγχο της ορθής λειτουργίας πιο αποδοτικό. Στα σχήματα 10.22, 10.23 και 10.24 βλέπουμε τρεις διαφορετικές περιπτώσεις. Στο πρώτο σχήμα τα δεδομένα δοκιμής παράγονται από μια συσκευή ελέγχου που είναι εξωτερική του ολοκληρωμένου κυκλώματος του οποίου ελέγχουμε την ορθή λειτουργία. Στη δεύτερη περίπτωση η μονάδα παραγωγής διανυσμάτων δοκιμής και αξιολόγησης της απόκρισης του υπό έλεγχο κυκλώματος βρίσκεται πάνω στο ίδιο ολοκληρωμένο κύκλωμα. Στην τρίτη περίπτωση η

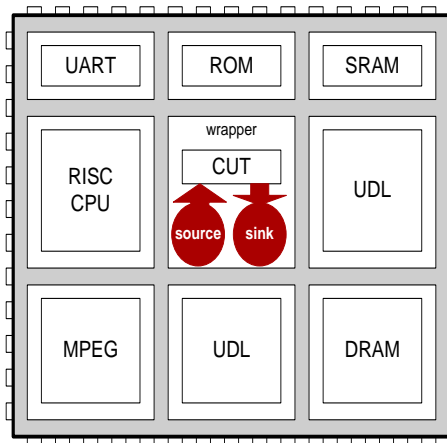
προσχεδιασμένη μονάδα διαθέτει τα δικά της κυκλώματα για την παραγωγή των διανυσμάτων δοκιμής και την αξιολόγηση της απόκρισης. Πρέπει να τονίσουμε ότι σε ένα ολοκληρωμένο κύκλωμα είναι δυνατόν να χρησιμοποιείται ένας συνδυασμός των τριών περιπτώσεων. Το σύνολο δοκιμής κάποιων προσχεδιασμένων μονάδων είναι δυνατόν να παράγεται εξωτερικά του ολοκληρωμένου κυκλώματος και επίσης εξωτερικά να γίνεται αξιολόγηση της απόκρισής του. Για κάποιες άλλες προσχεδιασμένες μονάδες είναι δυνατόν να χρησιμοποιείται μία κοινή μονάδα για την παραγωγή διανυσμάτων δοκιμής και την αξιολόγηση της απόκρισής τους. Κάποια ή κάποιες άλλες μονάδες μπορεί να έχουν τη δική τους μονάδα παραγωγής διανυσμάτων δοκιμής και αξιολόγησης της απόκρισής τους. Σε κάθε περίπτωση η σειριακή πόρτα μπορεί να χρησιμοποιηθεί για να φορτωθεί ο καταχωρητής εντολών με την επιθυμητή εντολή.



Σχήμα 10.22 Τα δεδομένα δοκιμής παράγονται από μια συσκευή ελέγχου που είναι εξωτερική του ολοκληρωμένου κυκλώματος του οποίου ελέγχουμε την ορθή λειτουργία.

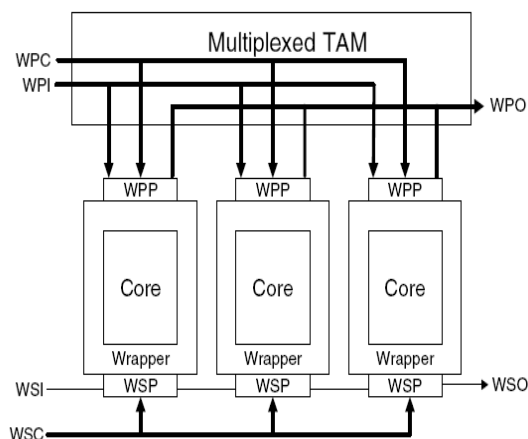


Σχήμα 10.23 Η μονάδα παραγωγής διανυσμάτων δοκιμής και αξιολόγησης της απόκρισης του υπό έλεγχο κυκλώματος βρίσκεται πάνω στο ίδιο ολοκληρωμένο κύκλωμα.

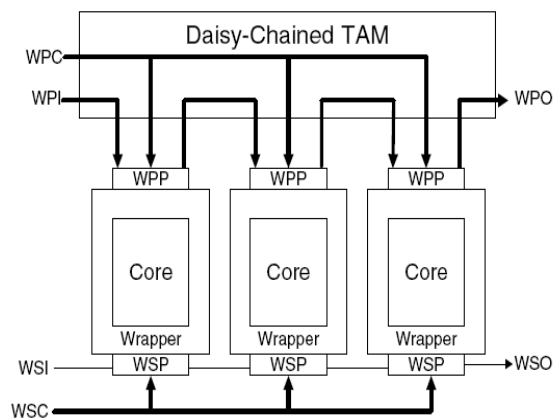


Σχήμα 10.24 Η προσχεδιασμένη μονάδα διαθέτει τα δικά της κυκλώματα για την παραγωγή των διανυσμάτων δοκιμής και την αξιολόγηση της απόκρισης.

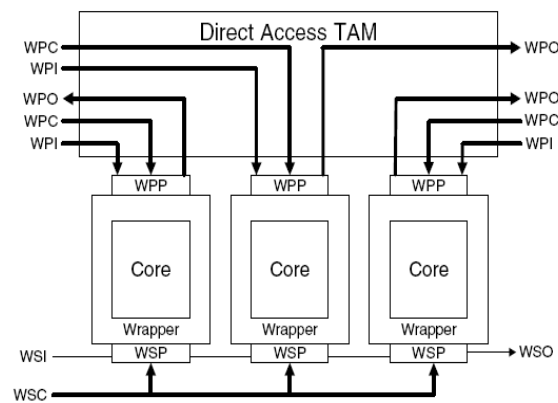
Υπάρχουν διάφοροι τρόποι διαμόρφωσης των παράλληλων μηχανισμών προσπέλασης δεδομένων δοκιμής (Test Access Mechanism, TAM) μεταξύ των οποίων μπορεί να επιλέξει ο σχεδιαστής του συστήματος, τα σχήματα 10.25, 10.26 και 10.27 δείχνουν τρεις από αυτούς. Στην πολυπλεγμένη αρχιτεκτονική, μόνο μια προσχεδιασμένη μονάδα μπορεί να χρησιμοποιεί το μηχανισμό προσπέλασης δεδομένων του ολοκληρωμένου κυκλώματος κάθε χρονική στιγμή. Ως εκ τούτου ο έλεγχος της ορθής λειτουργίας των γραμμών διασύνδεσης μεταξύ των προσχεδιασμένων μονάδων είναι δύσκολος. Στην αρχιτεκτονική αλυσίδας ο έλεγχος της ορθής λειτουργίας των προσχεδιασμένων μονάδων γίνεται ακολουθιακά. Στην αρχιτεκτονική άμεσης προσπέλασης οι γραμμές του μηχανισμού προσπέλασης δεδομένων κατανέμονται μεταξύ των προσχεδιασμένων μονάδων που υλοποιούν το πρότυπο. Σ' αυτή την περίπτωση ο καλύτερος αριθμός γραμμών του μηχανισμού προσπέλασης δεδομένων που κατανέμεται σε κάθε προσχεδιασμένη μονάδα εξαρτάται από τις απαιτήσεις της μονάδας αυτής, για παράδειγμα τη δομή των εσωτερικών μονοπατιών ολίσθησης, και τις απαιτήσεις όλων των άλλων προσχεδιασμένων μονάδων ώστε να ελαχιστοποιηθεί ο συνολικός χρόνος που διαρκεί ο έλεγχος της ορθής λειτουργίας.



Σχήμα 10.25 Μηχανισμός προσπέλασης δεδομένων πολυπλεγμένης αρχιτεκτονικής.



Σχήμα 10.26 Μηχανισμός προσπέλασης δεδομένων με αρχιτεκτονική αλυσίδας.



Σχήμα 10.27 Μηχανισμός προσπέλασης δεδομένων με αρχιτεκτονική άμεσης προσπέλασης.

Βιβλιογραφία

1. “VLSI Test Principles and Architectures, Design for Testability”, Edited by L.T. Wang, C.W. Wu and X. Wen, Morgan Kaufman, 2006.
2. “Principles of Testing Electronic Systems” S. Mourad and Y. Zorian, John Willey&Sons, 2000
3. “Logic Testing and Design for Testability”, H. Fujiwara, MIT Press, 1985
4. “Design for Test: For Digital ICs and Embedded Core Systems”, A. Crouch, Prentice Hall PTR, 1999.
5. “Built-In Test for VLSI: Pseudorandom Techniques”, P. H. Bardell, A. H. McAnney and J. Savir, John Willey&Sons, 1987.
6. “Logic Design Principles: with emphasis on testable semicustom circuits”, E. J. McCluskey, Prentice-Hall International Editions, 1986
7. “The Boundary-Scan Handbook”, 2nd edition, K. P. Parker, Kluwer Academic Publishers, 1998.
8. “The Core Test Wrapper Handbook, Rationale and Application of IEEE Std. 1500TM”, F. da Silva, T. McLaurin and T. Waayers, Springer, 2006.
9. “System On Chip Test Architectures, Nanometer Design for Testability”, Edited by L-T Wang, C. E. Straud and N. Toyba, Morgan Kaufman, 2008.
10. “Design for at-speed Test, Diagnosis and Measurement” B. Nadeau- Dostie, Kluwer Academic Publishers, 2000.
11. “System-on-a-Chip: Design and Test”, R. Rajsuman, Artech House, 2000.
12. “Testing of Digital Systems”, N. Jha and S. Gupta, Cambridge University Press 2003
13. “Test Resource Partisioning for System-on-chip”, K. Chakrabarty, V. Iyengar and A. Chandra, Kluwer Academic Publishers, 2002.
14. “Essentials of Electronic Testing”, M.L. Bushnell, V. D. Agrawal, Kluwer Academic Publishers, 2000.
15. “Arithmetic Built-In Self-Test, for Embedded Systems”, J. Rajski and J. Tyszer, Prentice Hall PTR, 1998.
16. Διεθνή επιστημονικά περιοδικά: IEEE Transactions on Computers, IEEE Transactions on CAD και IEEE Transactions on VLSI Systems.
17. Proceedings των συνεδρίων Design Automation and Test in Europe (DATE), Int. Test Conference (ITC) και Design Automation Conference (DAC).