ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

ΥΛΟΠΟΙΗΣΗ ΤΟΥ ΥΠΟΣΥΣΤΗΜΑΤΟΣ ΔΙΑΧΕΙΡΙΣΗΣ ΠΟΛΛΑΠΛΩΝ ΟΥΡΩΝ ΤΟΥ ΜΕΤΑΓΩΓΕΑ ATLAS I ΣΕ FULL-CUSTOM CMOS VLSI

Γεώργιος Ε. Κορνάρος

Μεταπτυχιακή Εργασία

Ηράκλειο, Ιούνιος 1997

Υλοποίηση του Υποσυστήματος Διαχείρισης Πολλαπλών Ουρών του Μεταγωγέα ATLAS I σε Full-Custom CMOS VLSI

Μεταπτυχιακή εργασία
που υποβλήθηκε στην επιτροπή μεταπτυχιακών σπουδών
του τμήματος Επιστήμης Υπολογιστών
της σχολής Θετικών Επιστημών
του Πανεπιστημίου Κρήτης
σε μερική εκπλήρωση των απαιτήσεων για την απόκτηση του
ΔΙΠΛΩΜΑΤΟΣ ΜΕΤΑΠΤΥΧΙΑΚΗΣ ΕΙΔΙΚΕΥΣΗΣ
Ηράκλειο, Ιούνιος 1997

Συγγραφέας:

Γεώργιος Ε. Κορνάρος
Τμήμα Επιστήμης Υπολογιστών

Εξεταστική Επιτροπή:

Μανόλης Κατεβαίνης, Καθηγητής
(Επόπτης)

Απόστολος Τραγανίτης, Αναπληρωτής Καθηγητής
(Μέλος)

Δημήτριος Σερπάνος, Επίκουρος Καθηγητής
(Μέλος)

Δεκτή:

Πάνος Κωνσταντόπουλος, Καθηγητής
Πρόεδρος Επιτροπής Μεταπτυχιακών Σπουδών

Υλοποίηση του υποσυστήματος διαχείρισης πολλαπλών ουρών του μεταγωγέα ATLAS I σε full-custom CMOS VLSI

Γεώργιος Ε. Κορνάρος
Μεταπτυχιακή Εργασία

Τμήμα Επιστήμης Υπολογιστών
Πανεπιστήμιο Κρήτης

ΠΕΡΙΛΗΨΗ

Οι απαιτήσεις για ποιότητα υπηρεσιών στα σύγχρονα δίκτυα οδηγούν στη χρήση μεταγωγέων υψηλής απόδοσης. Το πρωταρχικό απαραίτητο στοιχείο σ' αυτούς είναι η διατήρηση και η διαχείριση πολλαπλών ουρών, η οποία απαιτεί υλοποίηση σε hardware ώστε να επιτευχθούν υψηλές ταχύτητες.

Αυτή η εργασία αποτελείται από τη σχεδίαση, σε full-custom VLSI, του μεγαλύτερου μέρους του υποσυστήματος διαχείρισης ουρών του ATLAS I. Το ATLAS I είναι ένας μεταγωγέας ATM ολοκληρωμένος σε ένα chip με προαιρετικό έλεγχο ροής με βάση πιστώσεις. Αυτό το chip των 4 εκατομμυρίων transistors σχεδιάζεται και θα υλοποιηθεί σε τεχνολογία 0.35 μm CMOS, με 5 επίπεδα μετάλλου και τάση λειτουργίας 3.3 V, προσφέροντας σημαντικά πλεονεκτήματα, όπως συνολική παροχή 20 Gbit/s, κοινόχρηστη μνήμη 256 κυττάρων τύπου pipeline, 3 επίπεδα προτεραιότητας, και multicasting.

Το τμήμα διαχείρισης ουρών του ATLAS I είναι μία διπλή παράλληλη pipeline, η οποία διαχειρίζεται τις πολλαπλές ουρές έτοιμων κυττάρων, τις πιστώσεις ανά ομάδα ροής, και τα κύτταρα που περιμένουν πίστωση. Αυτές οι pipelines των 3 και 4 βαθμίδων, οι οποίες χειρίζονται γεγονότα με ρυθμό μιάς άφιξης ή αναχώρησης κυττάρου και μιας άφιξης πίστωσης ανά κύκλο ρολογιού, χρησιμοποιούν πολύπορτες μνήμες, καθώς και μνήμες αναζήτησης βάσει περιεχομένου. Υλοποιήσαμε σε full-custom 3-πορτες και 4-πορτες μνήμες

CAM και SRAM που υποστηρίζουν ειδικούς τύπους προσπελάσεων. Επίσης, το τμήμα διαχείρισης ουρών περιέχει ένα νέο μηχανισμό χρονοπρογραμματισμού, ο οποίος διαιτητεύει μεταξύ ροών οργανωμένων σε ομάδες, παρέχοντας εγγυήσεις δικαιοσύνης. Το μέρος full-custom της διαχείρισης ουρών περιέχει περίπου 65 χιλιάδες τρανζίστορς σε λογική και 14 Kbits σε διάφορες ειδικές μνήμες, καταλαμβάνει χώρο 2.3 τετραγωνικά χιλιοστά, καταναλώνει 270 mWatt (στη χειρότερη περίπτωση), και λειτουργεί στα 80 MHz (στη χειρότερη περίπτωση), έναντι των 50 MHz που απαιτούνται για την υποστήριξη των συνδέσμων του μεταγωγέα στα 622 Mb/s.

Implementation of Pipelined Multi-Queue Management of ATLAS I Switch in full-custom CMOS VLSI

George E. Kornaros
Master of Science Thesis

Department of Computer Science
University of Crete

ABSTRACT

The Quality of Service (QoS) requirements of contemporary networks lead to the use of sophisticated high-performance switching schemes; the primary ingredient of these is the maintenance and management of multiple queues of cells, which needs hardware implementation to achieve high speeds.

This work consists of the design, in full-custom VLSI, of the majority of the queue management subsystem of ATLAS I. ATLAS I is a single-chip ATM switch with optional credit-based flow control. This 4-million-transistor chip is being designed and will be implemented in a 0.35 micron CMOS technology with 5 metal layers and 3.3V power supply, offering important features: 20 Gbit/s aggregate I/O throughput, a 256-cell pipelined shared memory, multicasting, and 3 levels of priorities.

The queue management block of ATLAS I is a dual parallel pipeline that manages the multiple queues of ready cells, the per-flow-group credits, and the cells that are waiting for credits. These 3- and 4-stage pipelines, which handle events at the rate of one cell arrival or departure and one credit arrival per clock cycle, are using several multi-port memories, as well as some that can be searched by content. Two, three and four-port CAMs and SRAMs which support special access operations are laid out in full-custom. In addition, a new scheduling mechanism is incorporated in queue management, which arbitrates among flows arranged in groups rather than independently, providing fairness guarantees. The full-custom part of queue management

contains approximately 65 thousand transistors in logic and 14 Kbits in various special memories, occupies 2.3 square-mm, consumes 270 mW (worst-case), and operates at 80 MHz (worst case) versus 50 MHz which is the required clock frequency to support the 622 Mb/s ATLAS I switch links.

Ευχαριστίες

Ευχαριστώ όλους όσους με βοήθησαν να ολοκληρώσω αυτήν την εργασία. Την γενική επίβλεψη και καθοδήγηση έκανε ο επόπτης καθηγητής μου Μανόλης Κατεβαίνης, τον οποίο και ευχαριστώ θερμά γι' αυτήν. Η λεπτομερής σχεδίαση και υλοποίηση έγινε σε στενή συνεννόηση και τακτική συνεργασία με την Πένη Βατσολάκη, την οποία επίσης ευχαριστώ βαθιά. Η προκαταρκτική σχεδίαση της αρχιτεκτονικής διαχείρισης πολλαπλών ουρών του ATLAS I έγινε από τους Κοζυράκη, Βατσολάκη, και Κατεβαίνη. Επίσης τα κεφάλαια 1 και 2 αυτής της εργασίας προέρχονται από τη συναφή κοινή μου δημοσίευση με τους τρεις παραπάνω συνεργάτες (Συνέδριο ARVLSI '97).

Επίσης, ευχαριστώ τους καθηγητές Απόστολο Τραγανίτη και Δημήτριο Σερπάνο για τη συμμετοχή τους στην επιτροπή αξιολόγησης αυτής της εργασίας. Ακόμη, ευχαριστώ το Γιώργο Δημητριάδη για τη βοήθεια του στο σύστημα LaTeX 2_ε.

Ευχαριστώ ακόμα το Ινστιτούτο Πληροφορικής του Ιδρύματος Τεχνολογίας- Έρευνας για την οικονομική και τεχνική υποστήριξη. Η εργασία αυτή υποστηρίχτηκε οικονομικά από το πρόγραμμα "ACTS" (Advanced Communication Technologies and Services) της Ευρωπαικής Ένωσης, μέσω του έργου "ASICCOM" (ATM Switch for Integrated Communication, Computation, and Monitoring).

Τέλος, περισσότερο από όλους θέλω να ευχαριστήσω την οικογένειά μου για την αγάπη και υποστήριξη τους κατά τη διάρκεια των μεταπτυχιακών σπουδών μου.

Περιεχόμενα

ΠΕΡΙΛΗΨΗ **vii**

ABSTRACT **ix**

Ευχαριστίες **xi**

1 Εισαγωγή 1

1.1 Ο μεταγωγέας ATM ATLAS I 4

1.1.1 Διάγραμμα του μεταγωγέα 6

1.1.2 Φάση υλοποίησης 9

1.2 Η εργασία αυτή 10

2 Οργάνωση της Διαχείρισης Πολλαπλών Ουρών 11

2.1 Αρχιτεκτονική λογικών ουρών 11

2.2 Δομές δεδομένων σε hardware 13

2.3 Οι pipelines στη διαχείριση ουρών 16

2.4 Έλεγχος των pipeline 19

2.5 Απαιτήσεις πορτών στις μνήμες του διαχειριστή ουρών 20

3 Υλοποίηση σε full-custom CMOS VLSI 23

3.1 Περιγραφή του τμήματος full-custom 24

3.1.1 Η μνήμη creditMask 24

3.1.2 Οι μνήμες flowGroup και outMask 25

3.1.3 Περιφερειακά κυκλώματα 31

3.1.4 Η μνήμη του αρχείου καταχωρητών με τους δείκτες αρχής και τέλους 33

3.1.5 Η λίστα ελεύθερων θέσεων 36

3.1.6 Ένας γραμμικός αποκωδικοποιητής προτεραιότητας 36
3.2 Εξυπηρέτηση πολλαπλών κλάσσεων με δίκαιο τρόπο 43
3.2.1 Εισαγωγικά 44
3.2.2 Ένα σχήμα ψευδοτυχαίας επιλογής 45
3.2.3 Εναλλακτικές ντετερμινιστικές υλοποιήσεις 51
3.2.4 Ο δίκαιος δυναμικός χρονοπρογραμματιστής 52
3.2.5 Η υλοποίηση 53
3.3 Αποτελέσματα: Χρονικές επιδόσεις, εμβαδόν και κατανάλωση ισχύος . . . 55

4 Επεκτάσεις και Συμπεράσματα 59
4.1 Επεκτάσεις 59
4.2 Συμπεράσματα 60

A Pin descriptions of full-custom blocks 63

Βιβλιογραφία **66**

Κατάλογος Πινάκων

1.1 Πολυπλοκότητα του πυρήνα του ATLAS I 10

2.1 Ελάχιστες απαιτήσεις πορτών στις μνήμες του διαχειριστή ουρών 20

3.1 Χαρακτηριστικά αποτελέσματα χρονισμού των κυκλωμάτων full-custom . 56
3.2 Εμβαδόν και κατανάλωση ισχύος των κυκλωμάτων full-custom 57

A.1 Pin descriptions of full-custom blocks 65

Κατάλογος Σχημάτων

1.1 *Συνοπτική περιγραφή του chip ATLAS I* 4
1.2 *Απλοποιημένο διάγραμμα του ATLAS I* 6
1.3 *Διάγραμμα του πυρήνα του μεταγωγέα ATLAS I* 8

2.1 *Η αρχιτεκτονική των λογικών ουρών* 12
2.2 *Οι δομές δεδομένων στη διαχείριση ουρών* 14
2.3 *Οι δύο pipelines στη διαχείριση ουρών* 16

3.1 *Το κυρίως τμήμα full-custom* 25
3.2 *Το γενικό διάγραμμα της μνήμης creditMask* 26
3.3 *Το δίπορτο κύτταρο της μνήμης flowGroup* 27
3.4 *Συγκριτής ενός κυττάρου μνήμης CAM βάσει λογικής τύπου NAND* 27
3.5 *Ο οδηγητής των bitlines ανίχνευσης στη μνήμη flowGroup* 28
3.6 *Το κύτταρο μνήμης τριών πορτών της μνήμης outMask* 29
3.7 *Κυματομορφές ανίχνευσης στις μνήμες flowGroup-outMask (χειρότερη περίπτωση : διαφορά σε ένα μονο bit)* 30
3.8 *Λειτουργία αλλαγής/ανάγνωσης στη μνήμη outMask* 31
3.9 *Συμβατικός αισθητήριος ενισχυτής τύπου τάσης* 32
3.10 *Προ-αποκωδικοποιητής, αποκωδικοποιητής και μανταλωτές διευθύνσεων* 33
3.11 *Διάταξη της μνήμης head-tail register file* 33
3.12 *Το κύτταρο μνήμης της HTRF* 34
3.13 *Ο αισθητήριος ενισχυτής ρεύματος (PMOS-bias type) στη μνήμη HTRF* . . . 34
3.14 *Κυματομορφές προσομοίωσης συγκρίνοντας τους δύο αισθητήριους ενισχυτές (voltage vs. current)* 35
3.15 *Το κύκλωμα της λίστας ελεύθερων θέσεων* 37
3.16 *Η διάταξη και τα κυκλώματα του αποκωδικοποιητή προτεραιότητας* . . . 39

3.17 *Κυματομορφές του αποκωδικοποιητή προτεραιότητας (σενάριο χειρότερης περίπτωσης : καμιά είσοδος δεν ήταν έγκυρο ταίριασμα)* $(NE_i == NEH_i)$ 40
3.18 *Μία πύλη OR 16-εισόδων (υπολογίζει το αποτέλεσμα στη φάση phi1)* . . . 41
3.19 *Η βελτιωμένη βαθμίδα Β του αποκωδικοποιητή προτεραιότητας* 43
3.20 *Κυματομορφές του αποκωδικοποιητή προτεραιότητας χωρίς pipeline (χειρότερη περίπτωση : καμιά είσοδος δεν ήταν έγκυρο ταίριασμα)* 44
3.21 *Η υλοποίηση της γεννήτριας* $X_{n+1} = (5X_n + 1)\ mod\, 256$ 48
3.22 *Υλοποίηση της γεννήτριας* $X_{n+1} = (5X_n + 1)\ mod\, 256$ 49
3.23 *Μία περίοδος (256) (ζεύγη συνεχόμενων σημείων* (X_n, X_{n+1})*), στο δισδυάστατο χώρο* 50
3.24 *Μία γεννήτρια SRG μήκους 8 με συνδέσεις [1,3,5,8] και μία LFG(10,7,8)* . . 51
3.25 *Γραφική παράσταση 1000 μόνο σημείων (περίοδος=130944) μίας γεννήτριας LFG(10,7,8)* 51
3.26 *Παράδειγμα αλλαγής σειράς εξόδου κυττάρων από την εισερχόμενη σειρά τους* 52
3.27 *Η υλοποίηση του δυναμικού δίκαιου χρονοπρογραμματιστή* 54

Κεφάλαιο 1

Εισαγωγή

Οι επεξεργαστές, οι μνήμες, οι μεταγωγείς, και τα interfaces εισόδου/εξόδου αποτελούν τους βασικούς δομικούς λίθους των σύγχρονων και των μελλοντικών συστημάτων δικτύων υπολογιστών. Τα δίκτυα διασύνδεσης κατέχουν κεντρικό ρόλο σε όλες τις κλίμακες : από το χώρο των συστημάτων, ως τα τοπικά αλλά και τα δίκτυα ευρείας περιοχής. (πολλά θέματα αρχιτεκτονικού χαρακτήρα είναι όμοια σε όλα αυτά τα επίπεδα). Τα δίκτυα διασύνδεσης υψηλής ταχύτητας χρησιμοποιούν συνδέσμους σημείου προς σημείο και μεταγωγείς, έτσι ώστε να αποφευχθούν οι καθυστερήσεις που οφείλονται σε διαιτησία, και για να αυξηθεί ο παραλληλισμός στην επικοινωνία.

Στα σημερινά δίκτυα διασύνδεσης υπάρχει αυξημένο ενδιαφέρον για ποιότητα υπηρεσιών (QoS). Από τη μια όμως, βελτιωμένη ποιότητα υπηρεσιών συνεπάγεται μικρή καθυστέρηση για κίνηση υψηλής προτεραιότητας, το οποίο σημαίνει μεταξύ άλλων και χρονοπρογραμματισμό βάσει προτίμησης (preemptive scheduling) στους μεταγωγείς, όπως λόγου χάρη στη μεταγωγή πακέτων ή ποσοτήτων μικρού μεγέθους (σε δύο δημοφιλείς τεχνολογίες, αυτό επιτυγχάνεται χρησιμοποιώντας *κύτταρα* στην τεχνολογία Asynchronous Transfer Mode (ATM) [LeB92], ή *flits* στο Wormhole Routing [DS87]). Από την άλλη πλευρά, για να βελτιωθεί η ποιότητα υπηρεσιών απαιτείται εξεζητημένος έλεγχος ροής, ο οποίος βασίζεται στη διαχείριση κάθε ροής ατομικά*. Κατά συνέπεια αυτό σημαίνει τη διατήρηση και τη διαχείριση πολλαπλών ουρών. Επίσης, πολλές ουρές απαιτούνται για να αποφευχθεί το φαινόμενο μπλοκαρίσματος στην αρχή της ουράς (head-of-line blocking), έτσι ώστε να είναι δυνατόν δηλαδή οι ροές υψηλής προτεραιότητας ή οι ασυμφόρητες ροές να

*σύνδεση, ομάδα από συνδέσεις, ή σετ από πακέτα

προσπεράσουν αυτές που είναι χαμηλότερης προτεραιότητας ή συμφορημένες. Η υλοποίηση πολλαπλών ουρών και η διαχείρισή τους σε ψηλές ταχύτητες κατέχει ένα σημαντικότατο ρόλο στη σχεδίαση μεταγωγέων για δίκτυα υψηλής απόδοσης.

Αυτή η εργασία παρουσιάζει την υλοποίηση της διαχείρισης πολλαπλών ουρών και εξεζητημένου ελέγχου ροής σε hardware, μέσα σε ένα μεταγωγέα υψηλής ταχύτητας. Συγκεκριμένα παρουσιάζεται ο πυρήνας του τμήματος ελέγχου ενός μεταγωγέα ATM, ολοκληρωμένος σε ένα μοναδικό chip, με 16x16 σύνδεσμους των 622Mbps και προαιρετικό έλεγχο ροής βασισμένο σε πιστώσεις (backpressure ανά ομάδα ροής). Αυτό το chip, ονομαζόμενο ATLAS I, έχει 4 εκατομμύρια τρανζίστορ, παρέχει δυνατότητα multicasting, τρία επίπεδα προτεραιότητας, διαμοιρασμένο χώρο ενταμίευσης, καθυστέρηση για cut-through λιγότερη από ένα μικροδευτερόλεπτο, δυνατότητα ομαδοποίησης συνδέσμων, πίνακα μετάφρασης εικονικών μονοπατιών/κυκλωμάτων (VP/VC) και υποστήριξη για παρακολούθηση και έλεγχο φόρτου. Ο ATLAS I βρίσκεται υπό ανάπτυξη και θα κατασκευαστεί σύντομα σε τεχνολογία CMOS 0.35 micron.

Στο παρελθόν η μεταγωγή γινόταν συχνά σε επίπεδο λογισμικού, σε υπολογιστές γενικού σκοπού ή χρησιμοποιώντας επεξεργαστές συγκεκριμένα για αυτό το σκοπό ειδικά σε περιβάλλοντα τοπικού ή ευρύ δικτύου (LAN, WAN). Μεταγωγείς εξ' ολοκλήρου σε hardware έχουν χρησιμοποιηθεί κυρίως σε πολυεπεξεργαστές. Σήμερα όμως, με την επιθυμία για βελτιωμένη ποιότητα υπηρεσιών, που συνεπάγεται μικρές καθυστερήσεις και κύτταρα μικρού μεγέθους, οι μεταγωγείς υψηλής απόδοσης πρέπει να υλοποιούνται σε hardware [Tob90], και να ενσωματώνουν δομές δεδομένων κατάλληλες για τα κύτταρα που αποθηκεύουν, οι οποίες δεν ήταν απαραίτητες πριν [CST88]. Η σύγχρονη τεχνολογία VLSI παρέχει τη δυνατότητα για να ολοκληρωθούν τέτοιοι εξελιγμένοι μεταγωγείς σε ένα μοναδικό chip, αποφεύγοντας έτσι το υψηλό κόστος και τα μειονεκτήματα απόδοσης που αφορούν λύσεις πολλών chip. Διαχείριση εξεζητημένων δομών δεδομένων σε hardware μέσα σε μεταγωγείς έχει εμφανιστεί και παλιότερα σε δρομολογητές wormhole, όπως στους [TF88], iWarp [B$^+$90], Spider [Gal97], αλλά και σε μεταγωγείς ATM που έχουν αναπτύξει οι [KES$^+$91], [SMS$^+$91], Prizma [DEI95], κ.α.

Σε σύγκριση όμως με αυτές τις εργασίες ο ATLAS διατηρεί πολλές λογικές ουρές στην έξοδο ως ένα διαμοιρασμένο χώρο από αναγνωριστές (identifiers) με έναν μοναδικό κοινόχρηστο ελεγκτή, αποφεύγοντας έτσι το ψηλό κόστος σε χώρο των ενταμιευτών και των ελεγκτών ανά ουρά εξόδου όπως στον μεταγωγέα Prizma [DEI95]. Επίσης, η λειτουργία του μοναδικού ελεγκτή στον ATLAS I είναι pipelined, μια και πρέπει να διαχειριστεί τις

δομές δεδομένων των κυττάρων σε πολύ υψηλότερους ρυθμούς από ότι παλαιότεροι μεταγωγείς [TF88], [KES+91]. Ο μεταγωγέας ATLAS υλοποιεί έλεγχο ροής πολλών λωρίδων (multilane) βασισμένο σε πιστώσεις, ενώ άλλοι μεταγωγείς είτε δεν παρέχουν έλεγχο ροής [DEI95], [KES+91], [TF88], ή μονάχα έλεγχο ροής ατομικών λωρίδων (single-lane) [SMS+91]. Ορισμένοι δρομολογητές wormhole, όπως ο iWarp [B+90] και ο Spider [Gal97] υποστηρίζουν έλεγχο ροής πολλαπλών λωρίδων (multilane) για ένα μικρό αριθμό καναλιών ή ομάδων ροών και υλοποιούν ουρές με έναν ενταμιευτή σταθερού μεγέθους ανά κανάλι. Ο ATLAS I από την άλλη πλευρά υποστηρίζει 65536 ομάδες ροών (4096 για καθεμιά από τις 16 θύρες του), διαχειρίζεται 54 ουρές εξόδου από ''έτοιμα'' κύτταρα, όπως και έναν αριθμό από κύτταρα χωρίς πιστώσεις (''creditless''). Όλα τα κύτταρα και οι ουρές μοιράζονται *δυναμικά* έναν μοναδικό χώρο αποθήκευσης. Τέλος, ο ATLAS υλοποιεί όλα τα παραπάνω χαρακτηριστικά σε ένα μοναδικό chip, παρέχοντας μια ικανοποιητική λύση όσο αφορά το κόστος για δίκτυα διασύνδεσης υψηλής ταχύτητας.

Συμπερασματικά, ο ATLAS I προσφέρει ένα συνδυασμό από έλεγχο ροής υψηλής ταχύτητας πολλαπλές ουρές μέσα σε έναν κοινόχρηστο ενταμιευτή, και δυνατότητα multicasting, συνδυασμός που είναι μοναδικός ανάμεσα στους δρομολογητές wormhole και στους μεταγωγείς ATM. Η υλοποίηση αυτών των χαρακτηριστικών απαιτεί μια εξεζητημένη μονάδα ελέγχου μέσα σ'αυτόν τον μεταγωγέα σε VLSI. Η λειτουργία της βασίζεται σε δύο παράλληλες pipelines : η μία χειρίζεται αφίξεις και αναχωρήσεις κυττάρων και η δεύτερη αφίξεις πιστώσεων. Οι εξαρτήσεις που δημιουργούνται σ' αυτές τις pipelines των τριών και τεσσάρων βαθμίδων επιλύονται με προσπεράσματα, έτσι ώστε να αποφευχθούν φαινόμενα ακινητοποίησής τους. Επίσης ο ελεγκτής αυτός χρησιμοποιεί μη τετριμένες δομές δεδομένων : οι ουρές των ''έτοιμων'' κυττάρων υλοποιούνται ως διασυνδεδεμένες λίστες, οι άδειες θέσεις του ενταμιευτή αναγνωρίζονται από μια λίστα ελεύθερων θέσεων, και τέλος οι σύνδεσμοι εξόδου ενός κυττάρου και οι διαθέσιμες πιστώσεις του αναπαριστάνονται με μάσκες.

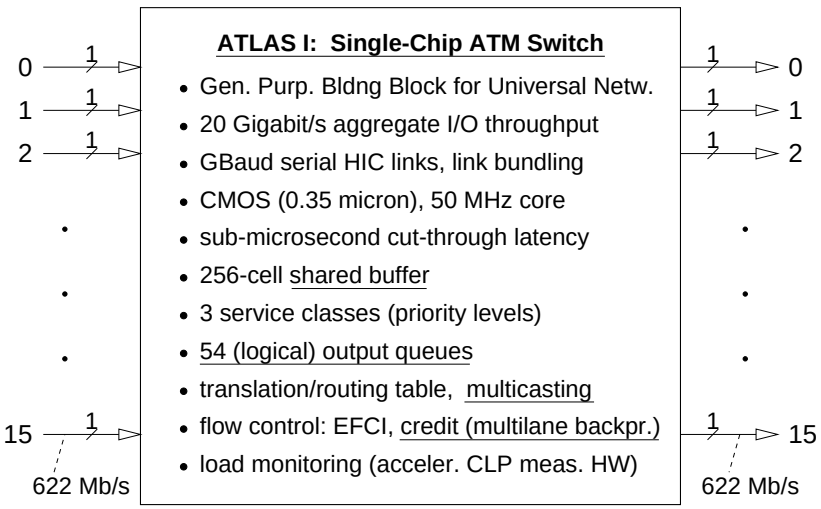
Η μονάδα ελέγχου είναι κεντρικοποιημένη και λειτουργεί πάνω στις επικεφαλίδες των κυττάρων στις διευθύνσεις και στους αναγνωριστές των πιστώσεων. Ο μέγιστος ρυθμός λειτουργίας της κατέδειξε την ανάγκη χρήσης πολύπορτων μνημών και μνημών προσπελάσιμων βάση του περιεχομένου τους, ώστε να υλοποιηθούν οι παραπάνω hardware δομές δεδομένων. Η εξεζητημένη λειτουργικότητα αυτών των μνημών, η οποία παρουσιάζεται εκτενώς στα ακόλουθα κεφάλαια, εισήγαγε την ανάγκη χρήσης τεχνικών σχεδίασης σε full-custom VLSI. Αυτή η εργασία παρουσιάζει ακριβώς την υλοποίηση σε full-custom

VLSI του υποσυστήματος διαχείρισης πολλαπλών ουρών του ATLAS I.

Ακολουθεί μία συνοπτική περιγραφή του μεταγωγέα ATM ATLAS I.

1.1 Ο μεταγωγέας ATM ATLAS I

Ο ATLAS I είναι ένας μεταγωγέας γενικού σκοπού, ολοκληρωμένος σε ένα μοναδικό chip, προοριζόμενος για χρήση σε συστήματα υψηλής παροχής και μικρών καθυστερήσεων, που μπορούν να κυμαίνονται από δίκτυα ευρείας περιοχής (WAN) σε τοπικά (LAN), ακόμα και σε συστήματα γραφείου (SAN/DAN), υποστηρίζοντας ποικιλία υπηρεσιών και εφαρμογών, από τηλεπικοινωνικές ως πολυμέσα και πολυεπεξεργαστές τύπου δικτύου σταθμών εργασίας (networks of workstations : NOWs). Ο ATLAS I αναπτύσσεται υπό το πρόγραμμα "ASICCOM"[†]



Σχήμα 1.1: *Συνοπτική περιγραφή του chip ATLAS I*

Το σχήμα 1.1 παρουσιάζει μία συνοπτική περιγραφή του ATLAS I. Είναι ένας μεταγωγέας 16x16, με σειριακούς συνδέσμους σημείου προς σημείο στα 622 Mbits/sec ο καθένας. Ο ATLAS I μπορεί ακόμη να λειτουργήσει ως μεταγωγέας 8x8 στα 1.25 Gbits/link, ή

[†]χρηματοδοτούμενο από το πρόγραμμα "ACTS" της Ευρωπαικής Ενωσης (Advanced Communication Technologies and Services). Το ASICCOM Consortium αποτελείται από συνεργάτες του βιομηχανικού χώρου (INTRACOM, Ελλάδα; SGS THOMSON, Γαλλία και Ιταλία; BULL, Γαλλία), συνεργάτες του χώρου των τηλεπικοινωνιών (TELENOR, Νορβηγία; TELEFONICA, Ισπανία), και ερευνητικά ινστιτούτα (FORTH, Ελλάδα; SINTEF, Νορβηγία; Politecnico di Milano, Ιταλία; NCSR Democritos, Ελλάδα).

ως 4x4 στα 2.5 Gbits/link, κτλ. Οι σύνδεσμοι φέρουν ATM κυκλοφορία πάνω από το IEEE Std. 1355 ''HIC/HS'' [HIC95] ως φυσικό επίπεδο, χρησιμοποιώντας τους πομποδέκτες σειριακών συνδέσμων ''STRINGS'' της BULL [MCLN93]. Το HIC προτιμήθηκε από το SONET λόγω της μικρότερης πολυπλοκότητας στα κυκλώματα, των μικρότερων καθυστερήσεων καθώς και της δυνατότητας κωδικοποίησης των πιστώσεων. Εσωτερικά ο ATLAS I λειτουργεί ως crossbar, με έναν κοινόχρηστο ενταμιευτή για 256 κύτταρα. Ο ATLAS I υλοποιεί τρία επίπεδα προτεραιότητας, όπου το καθένα έχει τις δικές του ουρές. Τα κύτταρα στον κοινόχρηστο ενταμιευτή είναι λογικά οργανωμένα σε 51 ουρές εξόδου (16 έξοδοι συν 1 θύρα διαχείρισης για καθεμιά από τις τρεις προτεραιότητες) και άλλες τρεις ουρές για multicasting[‡]. Εξαιτίας του ελέγχου εξ'ολοκλήρου σε hardware, και του εικονικού cut-through που παρέχει το crossbar, η καθυστέρηση ενός κυττάρου μέσω του μεταγωγέα ανέρχεται σε λιγότερο από ένα μικροδευτερόλεπτο όταν το δίκτυο φέρει ελαφρύ φορτίο. Ο ATLAS I παρέχει έναν πίνακα δρομολόγησης και μετάφρασης VP/VC, δυνατότητα για υποστήριξη multicasting, hardware για παρακολούθηση και έλεγχο φορτίου, και υποστήριξη του στάνταρντ ελέγχου ροής του ATM Forum, του EFCI (Error Forward Congestion Indicator).

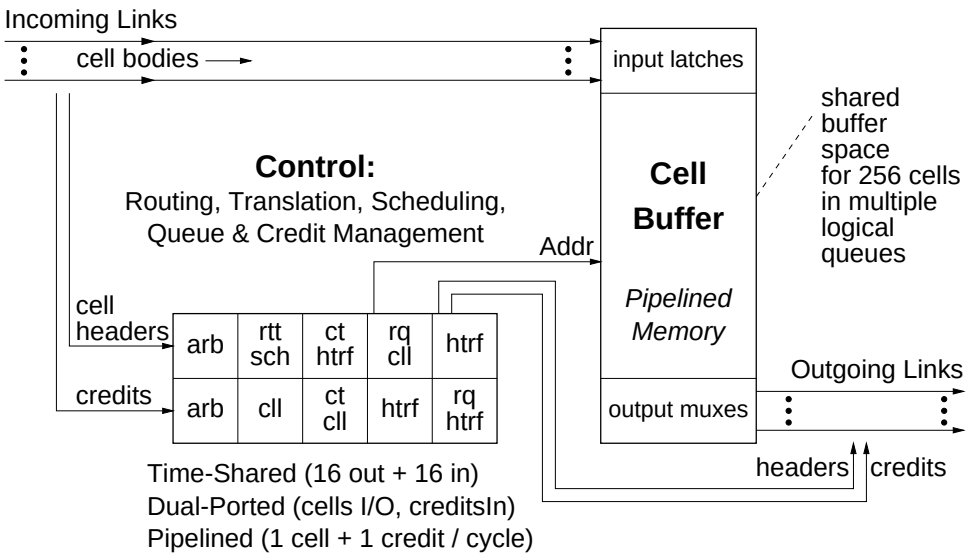
Ένα από τα χαρακτηριστικά του ATLAS I είναι η δυνατότητα (προαιρετική) υποστήριξης ελέγχου ροής βασισμένου σε πιστώσεις : τα κύτταρα αποστέλλονται στον παρακάτω γείτονα μόνο όταν έχουν αποκτήσει μια πίστωση. Οι πιστώσεις αντιπροσωπεύουν εγγυήσεις ελευθέρων θέσεων στον ενταμιευτή, έτσι ώστε οι ενταμιευτές να μην υπερχειλίζουν ποτέ και συνεπώς τα κύτταρα να μην απορρίπτονται ποτέ. Ο μηχανισμός πιστώσεων του ATLAS I είναι παρόμοιος με αυτόν του QFC [All95], αλλά είναι προσαρμοσμένος για υλοποίηση σε hardware πάνω από συνδέσμους αξιόπιστους και μικρού μήκους. Η δυνατότητα για έλεγχο ροής μέσω πιστώσεων, όπως θα δούμε στα επόμενα κεφάλαια, προκαλεί σημαντικές απαιτήσεις από την μονάδα διαχείρισης ουρών του ATLAS I. Ο μεταγωγέας ATLAS I μπορεί επίσης να συγχωνεύει πολλαπλές εισερχόμενες ομάδες ροών σε μία εξερχόμενη ομάδα ροής, εξασφαλίζοντας εγγυήσεις δικαιοσύνης ανάμεσα στις συγχωνευόμενες ροές, βάση ενός νέου μηχανισμού χρονοπρογραμματισμού.

Περισσότερες πληροφορίες για θέματα του ATLAS I που δεν αναλύονται σε αυτήν την εργασία μπορούν να βρεθούν στο [KSV96].

[‡]οι ουρές εξόδου εξαφανίζουν το πρόβλημα μπλοκαρίσματος στην αρχή της ουράς (head-of-line blocking

1.1.1 Διάγραμμα του μεταγωγέα

Στο σχήμα 1.2 φαίνεται ένα αρκετά αφηρημένο διάγραμμα του ATLAS I. Το σώμα των κυττάρων αποθηκεύεται στον κοινόχρηστο ενταμιευτή. Αυτός ο ενταμιευτής είναι υλοποιημένος σύμφωνα με την οργάνωση μνήμης κατά pipeline τρόπο, η οποία προσφέρει σημαντικά πλεονεκτήματα [KVE95]. Ο pipelined ενταμιευτής μπορεί να ξεκινήσει μία λειτουργία (διάβασμα ή γράψιμο) σε ένα κύτταρο ανά κύκλο ρολογιού. Αυτό το γεγονός καθορίζει τον ελάχιστο αριθμό κύκλων ρολογιού ανά το διάστημα χρόνου ενός κυττάρου : απαιτούνται τουλάχιστον 16 κύκλοι για την εξυπηρέτηση 16 εισόδων, συν 16 για τις 16 εξόδους, συν ένα επιπλέον κύκλο περιστασιακά για τη θύρα διαχείρισης. Το διάστημα χρόνου ενός κυττάρου είναι λίγο παραπάνω από 700 ns για ρυθμό δεδομένων 622 Mbits/sec πάνω στο σύνδεσμο. Ο ATLAS I συνεπώς χρονίζεται με ένα ρολόι συχνότητας περίπου 50 MHz, έτσι ώστε να υπάρχουν τουλάχιστον 33 κύκλοι ρολογιού ανά το χρόνο διάρκειας ενός κυττάρου. Από την άλλη πλευρά η τεχνολογία κατασκευής προσφέρεται για λειτουργία κυκλωμάτων σε μεγάλες ταχύτητες, έτσι ώστε να υπάρχει ένα σημαντικό περιθώριο ασφάλειας στην ταχύτητα του πυρήνα του ATLAS I (σχεδιάζουμε τα κυκλώματα να λειτουργούν με κύκλο ρολογιού 12 ns σε χειρότερη περίπτωση, κεφάλαιο 3).



Σχήμα 1.2: *Απλοποιημένο διάγραμμα του ATLAS I*

Το τμήμα ελέγχου του ATLAS I παίρνει αποφάσεις δρομολόγησης, μεταφράζει τους

αριθμούς σύνδεσης ATM (VP, VC), χρονοπρογραμματίζει τη χρήση της μνήμης του ενταμιευτή και της εξυπηρέτησης των συνδέσμων, και διαχειρίζεται τις ουρές των κυττάρων και τις πιστώσεις. Αυτό το τμήμα ελέγχου λειτουργεί πάνω στις επικεφαλίδες των κυττάρων, στις διευθύνσεις του ενταμιευτή και στους αναγνωριστές των πιστώσεων, δίχως να είναι ανάγκη να βλέπει καθόλου το σώμα των κυττάρων. Ο ATLAS I χρησιμοποιεί μία μοναδική μονάδα ελέγχου, η οποία είναι κοινόχρηστη στο χρόνο από όλους τους συνδέσμους. Η επιλογή αυτή έναντι μιας ''κατανεμημένης'' αρχιτεκτονικής[§] έγινε για τους ακόλουθους λόγους :

- Ένας διαφορετικός αριθμός από κύτταρα αποθηκευμένα στον ενταμιευτή δεδομένων μπορεί να προορίζεται για κάθε έξοδο. Αν κάθε έξοδος είχε τον δικό της ελεγκτή, τότε κάθε ένας από αυτούς θα έπρεπε να κρατάει ένα πολύ μεγάλο αριθμό από κύτταρα περιστασιακά (256 στη χειρότερη περίπτωση). Διατηρώντας όμως όλες τις λογικές ουρές μέσα σε μία μοναδική, κεντρική δομή δεδομένων (μεγέθους 256), εξαφανίζεται αυτομάτως η ανάγκη για τόσο επιπλέον χώρο αποθήκευσης.

- Η υλοποίηση του multicasting και της ομαδοποίησης συνδέσμων είναι απλούστερη με μία κεντρικοποιημένη αρχιτεκτονική ελέγχου.

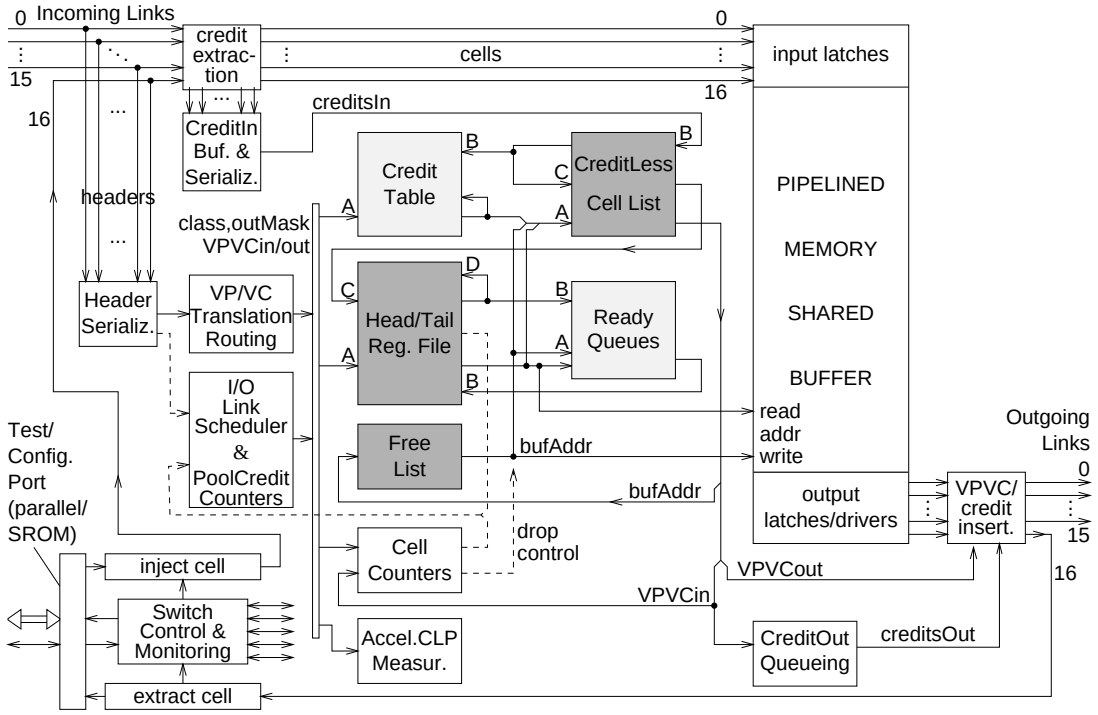
- Οταν δύο ελεγκτές στην κατανεμημένη αρχιτεκτονική είναι ανάγκη να επικοινωνήσουν, απαιτείται διαιτησία, η οποία ενέχει επιπλέον πολυπλοκότητα. Στην κεντρικοποιημένη αρχιτεκτονική η διαιτησία είναι απαραίτητη σε ένα μόνο σημείο : την πρώτη βαθμίδα της pipeline ελέγχου.

- Η λειτουργία του κεντρικού ελεγκτή συνυφαίνεται και συγχρονίζεται πολύ καλά με τη λειτουργία της μνήμης pipelined του ενταμιευτή.

Ο κεντρικός ελεγκτής βεβαίως πρέπει να έχει υψηλότερο μέσο ρυθμό λειτουργίας από κάθε ένα ξεχωριστά ελεγκτή στην περίπτωση της κατανεμημένης αρχιτεκτονικής. Ο μέγιστος όμως ρυθμός είναι ο ίδιος. Έτσι, η σχεδίαση του κεντρικοποιημένου ελεγκτή αποτελεί μία πρόκληση, είναι όμως εφικτή όπως αποδεικνύεται σε αυτήν την εργασία.

Για να ικανοποιηθεί ο μέγιστος ρυθμός λειτουργίας του μεταγωγέα, η μονάδα ελέγχου του ATLAS I πρέπει να χειρίζεται αφίξεις και αναχωρήσεις κυττάρων με ρυθμό μίας ανά κύκλο ρολογιού. Για να επιτευχθεί αυτό η μονάδα δουλεύει με pipelined τρόπο. Ο ρυθμός αφικνούμενων πιστώσεων στη χειρότερη περίπτωση σε μακρύ χρονικό διάστημα

[§]σε μία ''κατανεμημένη'' αρχιτεκτονική κάθε σύνδεσμος εισόδου και εξόδου έχει τον δικό του ελεγκτή



Σχήμα 1.3: *Διάγραμμα του πυρήνα του μεταγωγέα ATLAS I*

είναι μία ανά το χρονικό διάστημα που διαρκεί ένα κύτταρο και ανά σύνδεσμο εισόδου[¶]. Εντούτοις, ο στιγμιαίος ρυθμός πιστώσεων μπορεί να φτάσει μέχρι τα 16 ανά τη χρονική διάρκεια ενός κυττάρου. Αυτό προκαλεί τεράστια πολυπλοκότητα, οπότε ο ATLAS I περιορίζει το ρυθμό αφίξεων πιστώσεων σε οποιοδήποτε σύνδεσμο σε δύο ανά τη χρονική διάρκεια ενός κυττάρου. Έτσι το τμήμα ελέγχου πρέπει να είναι ικανό να διαχειρίζεται μέχρι 32 πιστώσεις (2x16 σύνδεσμοι) ανά τη χρονική διάρκεια ενός κυττάρου. Οι εισερχόμενες πιστώσεις διαχειρίζονται από μία δεύτερη παράλληλη pipeline, με ρυθμό μίας ανά κύκλο ρολογιού. Οι δύο pipelines της μονάδας ελέγχου ξεκινούν με μία φάση διαιτησίας, που επιλέγει έναν από τους εισερχόμενους συνδέσμους που ζητούν εξυπηρέτηση. Η pipeline των κυττάρων συνεχίζει στη βαθμίδα όπου γίνεται η δρομολόγηση/μετάφραση, ενώ παράλληλα ο χρονοπρογραμματιστής αποφασίζει αν θα εξυπηρετήσει μία έξοδο ή μία είσοδο. Οι επόμενες βαθμίδες έχουν να κάνουν με τη διαχείριση των ουρών και θα εξηγηθούν στο κεφάλαιο 2.

[¶] 1 πίστωση αντιστοιχεί σε χώρο μέσα στον ενταμιευτή ακριβώς για 1 κύτταρο

Το σχήμα 1.3 δείχνει ένα αφηρημένο διάγραμμα του πυρήνα του μεταγωγέα. Τα σκιασμένα κουτιά αποτελούν τη μονάδα ελέγχου, από τα οποία τα σκουρόχρωμα σχεδιάστηκαν σε full-custom CMOS VLSI.

1.1.2 Φάση υλοποίησης

Κατά το χρόνο συγγραφής αυτής της εργασίας τα περισσότερα τμήματα του μεταγωγέα έχουν σχεδιαστεί πλήρως σε επίπεδο πυλών ή τρανζίστορ και έλεγχος και επαλήθευση βρίσκονται σε εξέλιξη. Η πλειοψηφία του chip αποτελείται από semi-custom λογική και SRAM γεννημένη από compiler, ενώ ένα μικρό τμήμα έπρεπε να σχεδιαστεί σε full-custom. Ο ATLAS I θα κατασκευαστεί σε τεχνολογία CMOS 0.35 micron, από την SGS-Thomson, Crolles, Γαλλίας. Το chip θα αποτελείται από ένα πυρήνα που θα τοποθετηθεί σε ένα προυπάρχον δακτύλιο κατασκευασμένο από την BULL, Les Clayes sous Bois,Γαλλίας. Ο δακτύλιος έχει εξωτερικό μέγεθος 15 mm από κάθε πλευρά, και περιλαμβάνει τα pads, τους αντίστοιχους πομποδέκτες, και 16 κυκλώματα ονομαζόμενα ‘‘STRINGS’’ (μετατροπείς σειριακού σε παράλληλο) [MCLN93]. Ο πυρήνας περιλαμβάνει ολόκληρο το μεταγωγέα και 16 interfaces συνδέσμων (μηχανές πρωτοκόλλου HIC/HS και ελαστικοί ενταμιευτές). Η πολυπλοκότητα του πυρήνα φαίνεται στον ακόλουθο πίνακα, μετρημένη σε χιλιάδες τρανζίστορς σε κυκλώματα λογικής και σε Kilobits SRAM. Οι ελαστικοί ενταμιευτές είναι υλοποιημένοι από διακριτά flip-flops, γι’ αυτό και ο αριθμός των τρανζίστορ των interfaces των συνδέσμων φαίνεται υπερβολικά μεγάλος. Η μεγαλύτερη ποσότητα bits μνήμης στο chip είναι συγκεντρωμένη στον πίνακα δρομολόγησης και μετάφρασης, στον ενταμιευτή κυττάρων (μνήμη pipelined 27 βαθμίδων) και στον πίνακα πιστώσεων.

ATLAS I Core Complexity		
Block	Logic *(Kxtors)*	Memory *(Kbits)*
Semi-custom blocks:		
Link Interfaces, Elastic Buf.	260	-
Cell Buffer	30	110
Routing, VP/VC Translation	20	300
Credit Table	10	70
Queue management control	10	3
Scheduler, Control, Misc.	75	6

Πίνακας 1.1: (continued)

Block	Logic *(Kxtors)*	Memory *(Kbits)*
Full-custom blocks:		
CLL, HTRF, FL (multi-port RAM/CAM)	65	14
Total	470	503

Πίνακας 1.1: Πολυπλοκότητα του πυρήνα του μεταγωγέα ATLAS I

1.2 Η εργασία αυτή

Η παρούσα μεταπτυχιακή εργασία συνίσταται στην τελική σχεδίαση, υλοποίηση, προσομείωση και επαλήθευση των κυκλωμάτων του υποσυστήματος διαχείρισης πολλαπλών ουρών του μεταγωγέα ATLAS I. Τα κυκλώματα αυτά περιλαμβάνουν πολύπορτες μνήμες SRAM, μνήμες με ειδικές προσπελάσεις, και διευθυνσιοδοτημένες με βάση το περιεχόμενό τους, κωδικοποιητές, αποκωδικοποιητές προτεραιότητας, καθώς και τα περιφερειακά τους κυκλώματα. Η σχεδίασή τους έγινε με τεχνικές full-custom CMOS VLSI. Επίσης, η επινόηση του αλγορίθμου εξυπηρέτησης πολλαπλών συγχωνευόμενων ομάδων ροών με δίκαιο τρόπο (παραγράφος 3.2) πραγματοποιήθηκε από τον συγγράφοντα στα πλαίσια της παρούσας μεταπτυχιακής εργασίας.

Το κεφάλαιο 2 αυτής της εργασίας παρουσιάζει την αρχιτεκτονική και την εσωτερική οργάνωση του ελεγκτή διαχείρισης ουρών του ATLAS I. Περιλαμβάνει μία περιγραφή της διπλής pipeline, παρουσιάζοντας λεπτομερώς τις λειτουργίες κάθε βαθμίδας. Το κεφάλαιο 3 περιγράφει την υλοποίηση σε VLSI αυτού του διαχειριστή ουρών, και συγκεκριμένα τη σχεδίαση των κομματιών full-custom. Επίσης, παρουσιάζονται αποτελέσματα σχετικά με την ταχύτητα, το χώρο και την κατανάλωση ισχύος αυτών των κυκλωμάτων. Ακόμη, εδώ αναλύεται ο νέος μηχανισμός διαχείρισης συγχωνευόμενων ομάδων ροών. Τέλος το κεφάλαιο 4 δίνει συμπεράσματα και υποδεικνύει μελλοντικές επεκτάσεις αυτής της εργασίας.

Κεφάλαιο 2

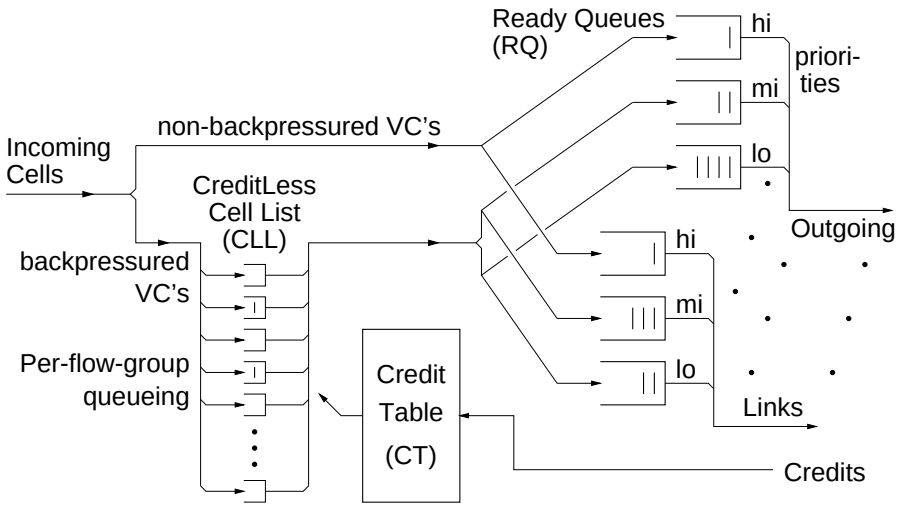
Οργάνωση της Διαχείρισης Πολλαπλών Ουρών

Η καρδιά του ελέγχου του ATLAS I είναι το τμήμα διαχείρισης ουρών, το οποίο κρατά τον έλεγχο όλων των κυττάρων που περιμένουν για πιστώσεις, τις πιστώσεις που περιμένουν για κύτταρα και τα κύτταρα που είναι έτοιμα για αναχώρηση. Αυτό το κεφάλαιο παρουσιάζει την οργάνωση αυτού του υποσυστήματος, καθώς και τις εναλλακτικές λύσεις που διερευνήθηκαν.

2.1 Αρχιτεκτονική λογικών ουρών

Τα κύτταρα μεταχειρίζονται διαφορετικά αναλόγως με το αν η σύνδεσή τους (VC) υπόκειται σε έλεγχο ροής προς τα πίσω. Η κλάση υψηλής προτεραιότητας προορίζεται για αστυνομευόμενη κίνηση πραγματικού χρόνου, όπως είναι η φωνή, όπου η απόρριψη πακέτων είναι προτιμότερη από την καθυστέρησή τους. Η κλάση αυτή επομένως δεν υπόκειται σε έλεγχο ροής προς τα πίσω. Η κλάση μεσαίας προτεραιότητας προορίζεται για αστυνομευόμενη κίνηση που υπόκειται σε έλεγχο ροής προς τα πίσω (όπως δεδομένα με εγγυήσεις εξυπηρέτησης), και τέλος η κλάση χαμηλής προτεραιότητας για μη-αστυνομευόμενη κίνηση που υπόκειται σε έλεγχο ροής προς τα πίσω (όπως δεδομένα για παράδειγμα). Αρα, μόνο οι συνδέσεις μεσαίας και χαμηλής προτεραιότητας υπόκεινται σε έλεγχο ροής βάσει πιστώσεων, όταν τέτοιος έλεγχος είναι ενεργοποιημένος. Τα κύτταρα που δεν υπόκεινται σε έλεγχο ροής προς τα πίσω αλλά και αυτά που υπόκεινται αλλά έχουν αποκτήσει την πίστωση που χρειάζονται για την αναχώρησή τους ονομάζονται ‘‘έτοιμα’’ κύτταρα, διότι είναι έτοιμα για αναχώρηση μόλις ο σύνδεσμος εξόδου τους γίνει διαθέσιμος. Τα ‘‘έτοιμα’’ τοποθετούνται στις ουρές έτοιμων κυττάρων. Υπάρχει μόνο μία ουρά έτοιμων

κυττάρων ανά σύνδεσμο εξόδου και ανά επίπεδο προτεραιότητας για τα unicast κύτταρα, όπως φαίνεται στο σχήμα 2.1. Αυτή η οργάνωση ουρών στις εξόδους εξαφανίζει το πρόβλημα μπλοκαρίσματος στην αρχή της γραμμής.



Σχήμα 2.1: *Η αρχιτεκτονική των λογικών ουρών*

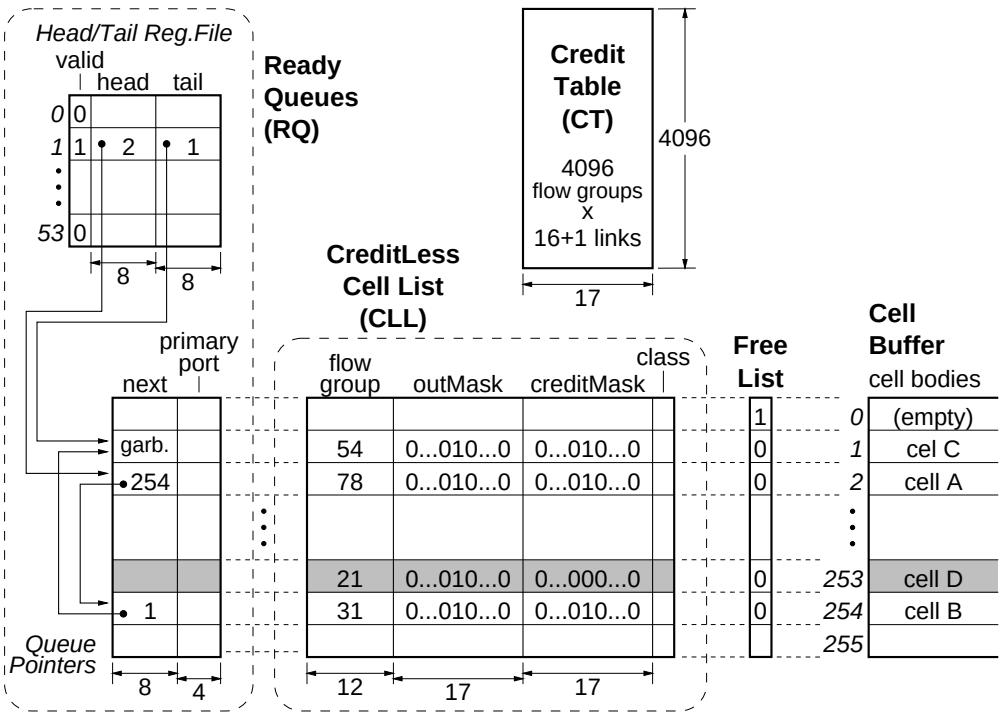
Υπάρχει επίσης μία ουρά έτοιμων κυττάρων ανά επίπεδο προτεραιότητας για τα κύτταρα που είναι multicast (δεν απεικονίζεται στο σχήμα 2.1), κοινόχρηστη από όλους τους συνδέσμους. Κάθε κύτταρο μπορεί να βρίσκεται το πολύ σε μία ουρά. Ο περιορισμός αυτός που τέθηκε για να ελεγκτεί η πολυπλοκότητα του hardware, είναι ο λόγος για τον οποίο οι ουρές multicast είναι κοινόχρηστες από όλους τους συνδέσμους. Αν το κύτταρο στην κορυφή της ουράς multicast προορίζεται για ένα σύνολο εξόδων A, τα κύτταρα που είναι multicast και βρίσκονται πίσω του είναι μπλοκαρισμένα, ακόμα και αν προορίζονται για ένα άλλο σύνολο εξόδων B. Εκτιμούμε ότι αυτό το μπλοκάρισμα θα έχει πολύ μικρή επίδραση στην απόδοση, διότι : (α) τα κύτταρα που είναι unicast και προορίζονται για τις εξόδους B δεν είναι μπλοκαρισμένα, (β) τα κύτταρα που είναι multicast αναμένεται να αποτελούν μειοψηφία στην συνολική κίνηση του μεταγωγέα, (γ) μεταξύ των κλάσεων εξυπηρέτησης, ο χρονοπρογραμματιστής εξυπηρετεί την ουρά multicast με υψηλότερη προτεραιότητα από τις ουρές unicast και τέλος (δ) τα κύτταρα στην ουρά multicast έχουν ήδη αποκτήσει όλες τις πιστώσεις που χρειάζονται, οπότε αναχωρούν για όλους τους προορισμούς τους μέσα σε χρονικό διάστημα διάρκειας ενός κυττάρου.

Τα κύτταρα που υπόκεινται σε έλεγχο ροής προς τα πίσω και δεν έχουν βρει μία κατάλληλη πίστωση περιμένουν στη ‘‘λίστα κυττάρων χωρίς πίστωση’’ (*CreditLess Cell List (CLL)*) μέχρις ότου φτάσει μία τέτοια πίστωση. Η CLL είναι λογικά οργανωμένη ως ένα σετ από ουρές, μία ανά ομάδα ροών, έτσι ώστε όταν φτάσει μία πίστωση να είναι δυνατόν να αποκτηθεί από ένα κύτταρο της ιδίας ομάδας ροής χωρίς να μπλοκάρεται από άλλες ομάδες ροών. Στην πραγματικότητα μόνο ένα κύτταρο είναι δυνατόν να υπάρξει μέσα στον ATLAS I. Αυτό επιτυγχάνεται αρχικοποιώντας τους μετρητές πιστώσεων για κάθε ομάδα ροής στο 1. Με αυτόν τον τρόπο δεν είναι ανάγκη να θυμάται κανείς την σειρά των κυττάρων μέσα σε μία ομάδα ροής, οπότε δεν είναι απαραίτητες πραγματικές ουρές μέσα στην CLL. Ο περιορισμός ενός το πολύ κυττάρου ανά ομάδα ροής μέσα στον ATLAS I δεν επηρεάζει το μέγιστο ρυθμό εξυπηρέτησης που μπορεί να επιτευχθεί σε περιβάλοντα SAN (System Area Networks), διότι ο ATLAS I είναι αρκετά γρήγορος ώστε ο χρόνος για την ολοκλήρωση μεταφοράς ενός κυττάρου και μιας πίστωσης πάνω από βραχείς συνδέσμους (round-trip-time) να είναι μικρότερος από το χρόνο διάρκειας ενός κυττάρου. Επιπλέον, αυτός ο περιορισμός έχει ευεργετικό αποτέλεσμα όσο αφορά την ανεκτικότητα σε εκρηκτική κίνηση και θερμά σημεία [KaSS96] : μία ομάδα ροής που δεν συμπεριφέρεται σωστά δεν της επιτρέπεται να χρησιμοποιήσει περισσότερες από μία θέση της μνήμης του ενταμιευτή. Δεδομένου ότι μπορεί να υπάρχουν το πολύ 256 κύτταρα μέσα στον ATLAS I σε οποιαδήποτε δεδομένη στιγμή, μπορούν να υπάρξουν το πολύ 256 ‘‘ουρές’’ μέσα στην CLL. Κατά συνέπεια η CLL έχει υλοποιηθεί ως μία μνήμη προσπελάσιμη βάσει του περιεχομένου της (CAM), όπως θα συζητηθεί παρακάτω.

2.2 Δομές δεδομένων σε hardware

Το σχήμα 2.2 απεικονίζει τις δομές δεδομένων που διατηρούνται και διαχειρίζονται από το hardware του ATLAS, έτσι ώστε να υλοποιηθεί η παραπάνω αρχιτεκτονική ουρών. Ολη η πληροφορία που αφορά ένα δεδομένο κύτταρο είναι αποθηκευμένη σε μνήμες που είναι ‘‘παράλληλες’’ προς τον ενταμιευτή κυττάρων, δηλαδή η διεύθυνση του σώματος ενός κυττάρου στον ενταμιευτή αποτελεί τον *αναγνωριστή* αυτού του κυττάρου, και χρησιμοποιείται για να προσπελάσει όλη τη σχετική με αυτό πληροφορία. Οι ουρές ‘‘έτοιμων’’ κυττάρων διατηρούνται ως διασυνδεδεμένες λίστες. Η υλοποίηση απλοποιείται αρκετά από την απόφαση να επιτρέπεται σε κάθε κύτταρο να ανήκει το πολύ σε μία ‘‘έτοιμη’’

ουρά: χρειάζεται ένας δείκτης στο επόμενο κύτταρο, που κρατείται στη μνήμη των δεικτών ουράς (Queue Pointer memory)*. Επίσης, το γεγονός ότι κάθε κύτταρο μπορεί να βρίσκεται το πολύ σε μία ‘‘έτοιμη ουρά’’ περιορίζει τις απαιτήσεις μέγιστου ρυθμού εξυπηρέτησης στον ελεγκτή. Σε διαφορετική περίπτωση, εισάγωντας κύτταρα σε πολλαπλές ουρές σε συγκεκριμένες χρονικές στιγμές θα υπαγόρευε έναν πολύ υψηλότερο στιγμιαίο ρυθμό λειτουργίας. Οι 54 ουρές που χρειάζονται στον ATLAS I αναγνωρίζονται από τα περιεχόμενα μίας μνήμης 54x17, του *Αρχείου Καταχωρητών Κεφαλής/Ουράς (HTRF)* (το bit εγκυρότητας χρησιμοποιείται για να διακρίνει τις άδειες ουρές από τις γεμάτες).



Σχήμα 2.2: *Οι δομές δεδομένων στη διαχείριση ουρών*

Μία εναλλακτική λύση που εξετάστηκε, αλλά απορρίφθηκε ήταν να υλοποιηθούν οι ‘‘έτοιμες’’ ουρές με *αριθμούς σειράς*. Για κάθε ουρά θα κρατιόταν ένας μετρητής σειράς για το *πρώτο* και το *τελευταίο* (όμοια ακριβώς με την HTRF). Για κάθε κύτταρο θα χρειαζόταν ένας αριθμός σειράς μαζί με την υπόλοιπη πληροφορία για την κατάσταση του κυττάρου (αυτό είναι όμοιο με το πεδίο δείκτη στο επόμενο κύτταρο). Εντούτοις, για να

*το πεδίο κύριας θύρας σ’ αυτή τη μνήμη αφορά μόνο τα κύτταρα multicast, και χρησιμοποιείται όταν χρονοπρογραμματίζεται η αποστολή τους

εντοπιστεί το πρώτο κύτταρο σε μία ουρά θα χρειαζόταν ένα ψάξιμο βάσει περιεχομένου στο πεδίο αριθμού σειράς, σε αντίθεση με μία απλή προσπέλαση σε μία μνήμη RAM όπως ισχύει για την οργάνωση με διασυνδεδεμένες λίστες. Βεβαίως πρέπει να σημειωθεί ότι δεν απαιτείται αυστηρή σειρά εξυπηρέτησης FIFO για τις συνδέσεις που υπόκεινται σε έλεγχο ροής προς τα πίσω, απλώς πρέπει να γίνεται ένας δίκαιος χρονοπρογραμματισμός. Αφού μέσα στο chip μπορεί να υπάρξει μόνο ένα κύτταρο ανά ομάδα ροής σε οποιαδήποτε δεδομένη στιγμή, δεν είναι ανάγκη να διατηρείται σειρά FIFO ούτε μέσα σε μία ομάδα ροής. Παρόλα αυτά, χρειάζεται σειρά FIFO για τις συνδέσεις που δεν υπόκεινται σε έλεγχο ροής προς τα πίσω, διότι η σειρά των κυττάρων μέσα σε ένα VC πρέπει να διατηρείται σύμφωνα με το ATM.

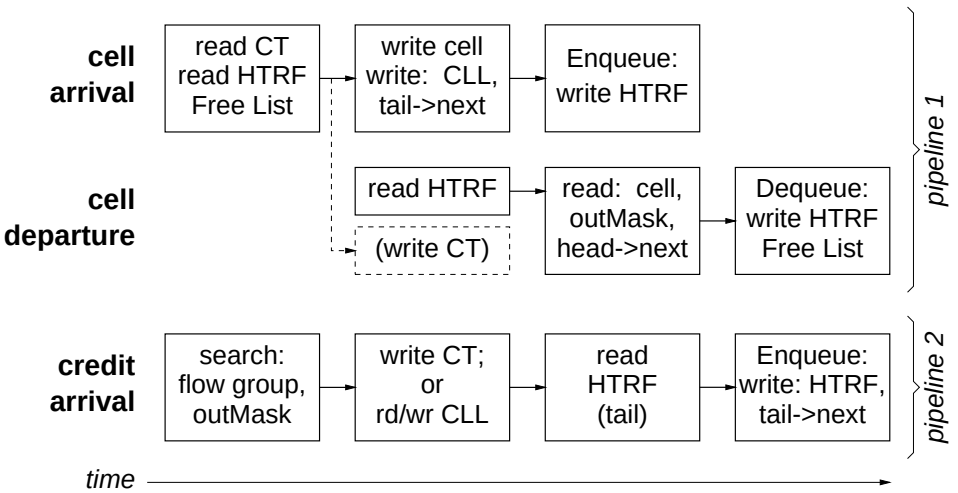
Για κάθε κύτταρο επίσης κρατιέται η εξής πληροφορία : ο αριθμός ομάδας ροής (12 bits), η μάσκα εξόδου (*outMask*), μία μάσκα των 17 bits που υποδεικνύει τους επιθυμητούς συνδέσμους εξόδου για ένα κύτταρο (ο 17ος σύνδεσμος αντιστοιχεί στη θύρα σύνδεσης του εξωτερικού μικροεπεξεργαστή). Επίσης η μάσκα πιστώσεων (*creditMask*), μία μάσκα των 17 bits που υποδηλώνει τους συνδέσμους εξόδου για τους οποίους το κύτταρο έχει αποκτήσει μία πίστωση[†]. Ολη αυτή η πληροφορία είναι σημαντική για τα κύτταρα που είναι χωρίς πιστώσεις (*creditless*) και άρα το σύνολο των μνημών που την φυλάσσει ονομάζεται λίστα των κυττάρων χωρίς πίστωση (*CreditLess Cell List (CLL)*). Όταν φτάνει μία νέα πίστωση ανιχνεύεται η CLL αν υπάρχει αποθηκευμένη καταχώρηση με την ομάδα ροής και το σύνδεσμο (outMask) της πίστωσης. Κατόπιν γίνεται ένα το αντίστοιχο bit της creditMask. Το πεδίο outMask ενδιαφέρει επίσης άμεσα τα έτοιμα κύτταρα που είναι multicast (τα άλλα έτοιμα κύτταρα μπορούν να εκμαιεύσουν όλη την πληροφορία που χρειάζονται από τον αριθμό ουράς στον οποίο ανήκουν). Τα κύτταρα που είναι χωρίς πίστωση δεν ανήκουν σε καμία έτοιμη ουρά, και άρα το πεδίο δείκτη στο επόμενο δεν χρησιμοποιείται. Στο παράδειγμα του σχήματος 2.2 κύτταρο 253 που είναι σκιασμένο είναι χωρίς πίστωση (οι μάσκες πιστώσεων και συνδέσμων εξόδου διαφέρουν κατά ένα bit).

Στον ATLAS I χρησιμοποιούνται πέντε ξεχωριστά κομάτια μνήμης για να υλοποιήσουν την παραπάνω οργάνωση. Αυτός ο διαχωρισμός που συζητείται παρακάτω, και κρίθηκε απαραίτητος λόγω των διαφορετικών ειδών προσπέλασης και χρονισμού καθενός πεδίου. Μία έκτη μνήμη μεγέθους 256x1 έχει χρησιμοποιηθεί ως λίστα των ελεύθερων θέσεων (*Free List*) του ενταμιευτή των κυττάρων.

[†]το 18ο bit αποτελεί έναν αναγνωριστή της κλάσης εξυπηρέτησης

2.3 Οι pipelines στη διαχείριση ουρών

Οι λειτουργίες που επιτελούνται στις παραπάνω δομές δεδομένων έχουν να κάνουν με την άφιξη ή αναχώρηση ενός κυττάρου και με την άφιξη μιας πίστωσης (οι αναχωρήσεις πιστώσεων συμβαίνουν σε συνδυασμό με τις αναχωρήσεις κυττάρων και περιλαμβάνονται στις λειτουργίες που έχουν να κάνουν με αυτές). Όπως ειπώθηκε στην παράγραφο 2.1 ο ελεγκτής του ATLAS I εκτελεί αυτές τις λειτουργίες χρησιμοποιώντας δύο παράλληλες pipelines, μία για τις λειτουργίες των κυττάρων και μία για τις αφίξεις των πιστώσεων. Η διαχείριση των ουρών επιτελείται στις τελευταίες τρεις βαθμίδες της pipeline των κυττάρων, καθώς και στις τελευταίες τέσσερεις της pipeline των πιστώσεων, όπως φαίνεται στο σχήμα 2.3 και εξηγείται παρακάτω. Οι λειτουργίες για τα κύτταρα αρχικοποιούνται από τον χρονοπρογραμματιστή του μεταγωγέα, στη βαθμίδα της pipeline των κυττάρων λίγο πριν αρχίσει η διαχείριση των ουρών. Ο χρονοπρογραμματιστής δίνει υψηλότερη προτεραιότητα στην εξυπηρέτηση των συνδέσμων εξόδου έναντι των συνδέσμων εισόδου, μια και η καθυστέρηση των εξόδων προκαλεί άσκοπη μείωση του ρυθμού εξυπηρέτησης, ενώ τα εισερχόμενα κύτταρα δεν θα χαθούν ποτέ ανεξάρτητα πότε θα εξυπηρετηθούν μέσα στο χρόνο διάρκειας ενός κυττάρου. Όσο αφορά τις αφίξεις των κυττάρων η δρομολόγηση και η μετάφραση συμβαίνει παράλληλα με τον χρονοπρογραμματισμό, ούτως ώστε να καθοριστεί η ομάδα ροής του κυττάρου, η μάσκα εξόδου και η κλάση εξυπηρέτησης.



Σχήμα 2.3: *Οι δύο pipelines στη διαχείριση ουρών*

Αφίξεις κυττάρων : Κατά τη διάρκεια της πρώτης βαθμίδας pipeline διαβάζεται ο πίνακας

πιστώσεων, έτσι ώστε να εξακριβωθεί αν υπάρχουν πιστώσεις για την ομάδα ροής αυτού του κυττάρου για τους επιθυμητούς σύνδεσμους εξόδου του (outMask). Αν βρεθούν όλες οι επιθυμητές πιστώσεις το κύτταρο γίνεται έτοιμο. Παράλληλα η κλάση εξυπηρέτησης και η μάσκα εξόδου για το εισερχόμενο κύτταρο χρησιμοποιούνται ως διεύθυνση για να διαβαστεί ο δείκτης ουράς της κατάλληλης έτοιμης ουράς από την HTRF. Αυτή θα χρησιμοποιηθεί για να εισαχθεί το κύτταρο στην ουρά σε περίπτωση που είναι έτοιμο. Ταυτόχρονα ψάχνεται η λίστα ελεύθερων θέσεων για την πρώτη κενή θέση.

Στην επόμενη βαθμίδα το κύτταρο καταχωρείται στην CLL. Αν το κύτταρο είναι έτοιμο ο δείκτης ουράς της κατάλληλης ουράς χρησιμοποιείται για να γραφεί η διεύθυνση του νεοαφικνούμενου κυττάρου στη θέση που δείχνει ο δείκτης στο επόμενο κύτταρο. Οι πιστώσεις που διαβάστηκαν από τον πίνακα πιστώσεων (CT) και χρησιμοποιήθηκαν από το κύτταρο καταχωρούνται στην CLL και πρέπει να σβηστούν από τον CT. Αυτό το σβήσιμο θα συνέβαινε κανονικά σε αυτή τη βαθμίδα, και θα απαιτούσε μία επιπλέον θύρα στη μνήμη CT. Για να αποφευχθεί αυτή η επιπλέον θύρα, έτσι ώστε το CT να μπορεί να είναι μία SRAM γεννημένη από compiler παρά να σχεδιαστεί σε full-custom, η λειτουργία του σβησίματος καταχωρείται σε έναν ενταμιευτή για εγγραφές και εκτελείται κατά τη διάρκεια της πρώτης βαθμίδας μιας αναχώρησης ενός κυττάρου (οι αναχωρήσεις κυττάρων δεν προσπελαύνουν τον CT). Υπάρχουν 17 καταχωρητές σε αυτόν τον ενταμιευτή για εγγραφές, καθώς και οι αντίστοιχοι συγκριτές και τα κυκλώματα για προσπεράσματα, έτσι ώστε να καλυφθεί η χειρότερη περίπτωση συνεχών αφίξεων κυττάρων από όλες τις εισόδους προτού ο χρονοπρογραμματιστής αποφασίσει να εξυπηρετήσει τις θύρες εξόδου.

Στην τελευταία βαθμίδα, αν το κύτταρο είναι έτοιμο, ο νέος δείκτης ουράς και το bit άδειας ή όχι ουράς γράφονται στη μνήμη HTRF. Αν η ουρά ήταν πριν άδεια, πρέπει να γραφτεί συγχρόνως και ο δείκτης επικεφαλίδας και ο χρονοπρογραμματιστής ειδοποιείται ότι η ουρά δεν είναι πλέον άδεια. Αυτές οι προσπελάσεις θα μπορούσαν να πραγματοποιηθούν ένα κύκλο νωρίτερα, αλλά αυτό θα ερχόταν σε σύγκρουση με προσπελάσεις στην ίδια μνήμη κατά τη διάρκεια των λειτουργιών που έχουν να κάνουν με τις αναχωρήσεις κυττάρων.

Αναχωρήσεις κυττάρων : Οι λειτουργίες που συμβαίνουν όταν ο χρονοπρογραμματιστής καθοδηγεί την pipeline των κυττάρων να χειριστεί μία αναχώρηση ενός κυττάρου από μία συγκεκριμένη έξοδο και σε ένα επίπεδο προτεραιότητας είναι οι εξείς. Στην πρώτη βαθμίδα διαβάζονται οι δείκτες επικεφαλίδας και τέλους για την αντίστοιχη έτοιμη ουρά

από την HTRF. Ο δείκτης επικεφαλίδας χρησιμοποιείται για να εντοπίσει το σώμα του κυττάρου (για μετάδοση) καθώς και τις υπόλοιπες πληροφορίες που σχετίζονται με αυτό. Ο δείκτης τέλους διαβάζεται έτσι ώστε να ανιχνευτεί αν αυτό ήταν το τελευταίο κύτταρο στην ουρά (δείκτης τέλους==δείκτη αρχής).

Στη δεύτερη βαθμίδα η μνήμη που κρατά τις μάσκες εξόδου προσπελαύνεται ως εξής: μηδενίζεται το bit που αντιστοιχεί στον τρέχον σύνδεσμο εξόδου, ενώ ταυτόχρονα διαβάζονται τα υπόλοιπα 16 bits. Αν αυτά τα 16 bits είναι όλα μηδέν, το κύτταρο πρέπει να βγει από την ουρά και η θέση του να επιστραφεί στην Free List, διότι ήταν είτε ένα κύτταρο unicast είτε ένα multicast που αναχωρεί από την τελευταία έξοδό του. Ο δείκτης στο επόμενο κύτταρο στην ουρά διαβάζεται επίσης παράλληλα, αφού θα χρειαστεί στην περίπτωση εξαγωγής του από την ουρά.

Στην τρίτη βαθμίδα το κύτταρο εξάγεται από την ουρά, αν είναι απαραίτητο. Αυτό συνεπάγεται την ενημέρωση της λίστας ελεύθερων κυττάρων καθώς και του δείκτη αρχής στη μνήμη HTRF, όπως και το bit εγκυρότητας αν αυτό ήταν το τελευταίο κύτταρο στην ουρά. Στην τελευταία περίπτωση ειδοποιείται και ο χρονοπρογραμματιστής ότι αυτή η ουρά είναι τώρα άδεια.

Αφίξεις πιστώσεων : Οι αφίξεις των πιστώσεων χειρίζονται από τη δεύτερη pipeline, με ρυθμό μίας ανά κύκλο ρολογιού, παράλληλα με τις αφίξεις και αναχωρήσεις των κυττάρων. Αυτή είναι μία pipeline τεσσάρων βαθμίδων. Στην πρώτη βαθμίδα η ομάδα ροής όπου ανήκει μία πίστωση και μία μάσκα των 16 bits που δείχνει το σύνδεσμο από όπου έφτασε χρησιμοποιούνται για να ψάξουμε την CLL (τα πεδία ομάδα ροής και μάσκα εξόδου) για κύτταρα που περιμένουν αυτήν την πίστωση. Το ψάξιμο αυτό απαιτεί ένα πλήρες ταίριασμα με το πεδίο που δηλώνει την ομάδα ροής, ενώ με το πεδίο της μάσκας εξόδου απαιτείται μερικό ταίριασμα (ένα μόνο bit). Δεν είναι ανάγκη να κοιτάξουμε καθόλου τη μνήμη creditMask : αφού υπάρχει μόνο μία πίστωση ανά ομάδα ροής για ένα δεδομένο σύνδεσμο μία δεδομένη χρονική στιγμή, δεν υπάρχει περίπτωση για ένα κύτταρο μιας δεδομένης ομάδας ροής που περιμένει ακόμα να αναχωρήσει από αυτό το σύνδεσμο (το συγκεκριμένο bit της μάσκας εξόδου είναι ένα) να έχει ήδη μία πίστωση για αυτή την ομάδα ροής και για αυτό τον σύνδεσμο.

Στη δεύτερη βαθμίδα, αν κανένα κύτταρο δεν περίμενε για αυτήν την πίστωση αυτή σημειώνεται στον πίνακα πιστώσεων (CT). Αφού μπορεί να υπάρξει μόνο μία πίστωση ανά

ομάδα ροής μία εγγραφή ενός απλού 1 είναι αρκετή (δεν είναι ανάγκη να γίνει διάβασμα-μεταβολή-γράψιμο). Ειδάλως ενημερώνεται η μάσκα των πιστώσεων του επιλεγμένου κυττάρου με το να γίνει 1 το bit που αντιστοιχεί σύνδεσμο εισόδου, ενώ ταυτόχρονα διαβάζονται τα υπόλοιπα 16 bits, αλλά και όλη η μάσκα των συνδέσμων εξόδου. Συγκρίνοντας τις δύο μάσκες αποφασίζεται αν το κύτταρο είναι έτοιμο, δηλαδή αν πήρε και την τελευταία πίστωση που περίμενε και άρα πρέπει να εισαχθεί στην ουρά.

Κατά την τρίτη και την τέταρτη βαθμίδα εκτελείται η εισαγωγή στην ουρά, αν το κύτταρο μόλις έγινε έτοιμο. Η διεύθυνση της κατάλληλης ουράς για να διαβαστεί η HTRF σχηματίζεται χρησιμοποιώντας τη μάσκα εξόδου του κυττάρου και την κλάση εξυπηρέτησής του. Στην επόμενη βαθμίδα ο δείκτης tail->next γράφεται, όπως και ο δείκτης τέλους και το bit εγκυρότητας. Αν η ουρά ήταν πριν άδεια, ο δείκτης αρχής πρέπει επίσης να γραφτεί και να ενημερωθεί και ο χρονοπρογραμματιστής.

2.4 Έλεγχος των pipeline

Οι pipelines της διαχείρισης ουρών ελέγχονται με τρόπο παρόμοιο με αυτόν στις pipelines των συνήθη επεξεργαστών RISC : τα σήματα ελέγχου γεννιούνται στην αρχική βαθμίδα και διαδίδονται στις επόμενες βαθμίδες παράλληλα με τους υπολογισμούς μέσω καταχωρητών pipeline, μέχρις ότου ‘‘καταναλωθούν’’ στην κατάλληλη βαθμίδα. Ομως ο τρόπος αυτός λειτουργίας μπορεί να οδηγήσει σε κινδύνους σε σχέση με τα δεδομένα (data hazards) : οι λειτουργίες που εκτελούνται παράλληλα μέσα σε διαφορετικές pipelines ή σε διαφορετικές βαθμίδες μίας pipeline, μπορεί να δουλεύουν ταυτόχρονα με τα ίδια δεδομένα και συνεπώς μπορεί να οδηγήσουν σε ασύμβατα αποτελέσματα, όπως να καταστρέψουν την συνεκτικότητα μίας ουράς, γεννώντας ανύπαρκτα κύτταρα, ή να απορρίψουν κύτταρα ή πιστώσεις.

Στην περίπτωση του ATLAS I, όλοι οι κίνδυνοι που σχετίζονται με δεδομένα μπορούν να αποφευχθούν χρησιμοποιώντας προσπεράσματα (*bypassing*, forwarding) [PH93], χωρίς να κωλυσιεργούν την pipeline. Υπάρχουν δύο περιπτώσεις προσπεράσματος που χειρίζονται σχεδόν ταυτόχρονες αφίξεις ενός κυττάρου και της αντίστοιχής του πίστωσης, και 12 περιπτώσεις προσπεράσματος που χειρίζονται ταυτόχρονες ή συνεχόμενες λειτουργίες στην ίδια ‘‘έτοιμη’’ ουρά. Πέντε συγκριτές και μερικές δεκάδες πυλών ανιχνεύουν και ελέγχουν τις συνθήκες προσπεράσματος.

Η ορθότητα των pipelines και ο έλεγχος τους επαληθεύτηκε με την παράλληλη προσομοίωση δύο μοντέλων : (α) ενός μοντέλου behavioral των πράξεων στη διαχείριση ουρών, και (β) ενός functional (RTL) μοντέλου των δύο pipelines και της λογικής ελέγχου. Τα διανύσματα ελέγχου περιελάμβαναν τυχαίους συνδυασμούς, αλλά και έναν εξαντλητικό κατάλογο γεννημένο από υπολογιστή, από όλες τις 432 δυνατές περιπτώσεις ταυτόχρονων ή συνεχόμενων λειτουργιών σε κύτταρα και πιστώσεις. Η ορθότητα ελέγχθηκε συγκρίνοντας την κατάσταση των ουρών και τις εξόδους των δύο προσομοιωτών. Λεπτομερής περιγραφή όλων των κανόνων προσπεράσματος μπορεί να βρεθεί στο [Koz96].

2.5 Απαιτήσεις πορτών στις μνήμες του διαχειριστή ουρών

Από τις λειτουργίες που περιγράφηκαν παραπάνω, είναι φανερό ότι όλα τα κομμάτια μνημών προσπελαύνονται σε πολλές βαθμίδες της μίας ή και των δύο pipelines, καταδείχνοντας την ανάγκη πολύπορτων μνημών. SRAMs μίας ή δύο πορτών μπορούν να γεννηθούν από compiler που χρησιμοποιούμε στο περιβάλλον σχεδιασμού. Μνήμες όμως με τρεις ή περισσότερες πόρτες, ή μνήμες CAMs έπρεπε να σχεδιαστούν σε full-custom. Ο ακόλουθος πίνακας συνοψίζει τις απαιτήσεις πορτών των μνημών του τμήματος διαχείρισης ουρών. Επίσης φαίνεται το είδος της πόρτας που απαιτείται.

Memory Block	**Number of Ports**	**Type of Ports**
HTRF	4	read (2), write (2)
Next Pointer	2	compiled 2-port SRAM
Credit Table	2	compiled 2-port SRAM
Free List	2	CAM (1), write (1)
FlowGroup	2	read/write (1), CAM (1)
OutMask	3	read (1), read/modify (1), CAM (1)
CreditMask	2	write (1), read/modify (1)

Πίνακας 2.1: Ελάχιστες απαιτήσεις πορτών στις μνήμες του διαχειριστή ουρών.

Το αρχείο καταχωρητών των δεικτών αρχής και τέλους (HTRF) είναι η πιο απαιτητική μνήμη όσο αφορά τον αριθμό των θυρών. Αλλες εναλλακτικές λύσεις εξετάστηκαν για να ελαττωθεί αυτός ο αριθμός. Η πρώτη ήταν να χρησιμοποιηθούν ξεχωριστές μνήμες για

τους δείκτες αρχής και τέλους. Όμως η λειτουργία εισαγωγής κυττάρου σε άδεια ουρά απαιτεί 4 πόρτες από το αρχείο καταχωρητών των δεικτών τέλους, διότι και οι δύο δείκτες πρέπει να ενημερώνονται. Η δεύτερη λύση ήταν να κρατάμε ξεχωριστές μνήμες για τους δείκτες αρχής και τέλους καθώς και ένα επιπλέον αντίγραφο της μνήμης με τους δείκτες τέλους. Τώρα ο αριθμός των πορτών μπορούσε να μειωθεί στις τρεις, αλλά πάλι θα έπρεπε να σχεδιαστεί σε full-custom, και θα ήταν και μεγαλύτερη σε χώρο από την αρχική τετράπορτη μνήμη, οπότε ούτε αυτή επιλέχθηκε.

Κεφάλαιο 3

Υλοποίηση της Διαχείρισης Πολλαπλών Ουρών σε full-custom CMOS VLSI

Όπως φάνηκε στο κεφάλαιο 2, η μονάδα διαχείρισης ουρών και πιστώσεων του ATLAS I απαιτεί αρκετές πολύπορτες μνήμες, καθώς και κάποιες που να μπορούν να προσπελασθούν βάσει περιεχομένου. Οι μνήμες CAM's και οι 3- και 4-πορτες SRAM's έπρεπε να σχεδιαστούν σε full-custom. Η σχεδίασή τους αναλύεται στο κεφάλαιο αυτό. Κατά τη διάρκεια της σχεδίασης των τμημάτων σε full-custom επικεντρωθήκαμε στην απλότητα και εξασφάλιση της ορθής λειτουργίας των κυκλωμάτων παρά στη χρήση περίπλοκων τεχνικών για επίτευξη υψηλότερης ταχύτητας ή πολύ χαμηλής κατανάλωσης ισχύος. Ο λόγος είναι ότι ο ATLAS I έχει σημαντική πολυπλοκότητα σε αρχιτεκτονικό επίπεδο, και συνεπώς προτιμήσαμε να ελαττώσουμε το χρόνο σχεδίασης και τους κινδύνους που ενέχει το κομμάτι full-custom. Κάτω από το ίδιο πρίσμα επιλέξαμε να μείνουμε μακρυά από οργανώσεις που θα απαιτούσαν σημαντική βετλιστοποίηση στο χρονισμό μετά το placement και το routing. Όπως αναφέρθηκε στο τμήμα 2.1, σχεδιάσαμε όλα τα κυκλώματά μας να λειτουργούν με κύκλο ρολογιού 12ns υπό τις χειρότερες συνθήκες, αν και ο ρυθμός στους συνδέσμους OC-12 των 622 Mbits/sec απαιτεί ένα κύκλο ρολογιού 20ns (ακριβέστερα : 21.44 ns). Αυτό έγινε για να έχουμε ένα καλό περιθώριο ασφαλείας από τη μιά και για να είμαστε έτοιμοι για έναν μεταγωγέα ATLAS 2 υψηλότερης ταχύτητας, από την άλλη. Σε μερικά σημεία αυτό απαιτούσε την εισαγωγή έξτρα βαθμίδων pipeline (όπως στη μετάφραση VP/VC). Θεωρούμε ότι αυτό είναι αποδεκτό για τον ATLAS I, δεδομένου ότι η καθυστέρησή του cut-through είναι πολύ μικρότερη από άλλους σύγχρονους μεταγωγείς ATM ούτως ή άλλως.

Συγκεκριμένα, όποτε κρίθηκε δυνατό χρησιμοποιήσαμε σύγχρονες (synchronous) τεχνικές σχεδίασης, οι οποίες είναι λιγότερο ευαίσθητες στις σχετικές καθυστερήσεις, είναι ευκολότερη η επαλήθευση και δεν χρειάζεται προσεκτική ρύθμιση κατά τη γέννηση των σημάτων χρονισμού. Επίσης, η εκτέλεση λογικών πράξεων μεταξύ ρολογιού και δεδομένων ή σημάτων ελέγχου περιορίστηκε στο ελάχιστο. Για τα κυκλώματα μνημών χρησιμοποιήθηκε το συμβατικό στατικό κύτταρο μνήμης CMOS, το οποίο έχει μικρότερη κατανάλωση ισχύος σε κατάσταση αναμονής και μεγαλύτερη ανοσία σε μεταβαλλόμενο θόρυβο και μεταβολές της τάσης από άλλα κύτταρα. Μεγάλα φορτία, όπως σήματα ρολογιού οδηγούνται από δέντρα με αντιστροφείς παρά από ένα μεγάλο οδηγητή ώστε να μειωθεί η κατανάλωση ισχύος. Ακόμη, οι κλάδοι των δέντρων εξισοροπήθηκαν ώστε να αποφευχθούν οι σχετικές καθυστερήσεις. Τέλος, χρησιμοποιήσαμε πολλαπλές γραμμές παροχής ισχύος ώστε να μειωθούν το πλάτος και η διάρκεια μεταβολών τάσης.

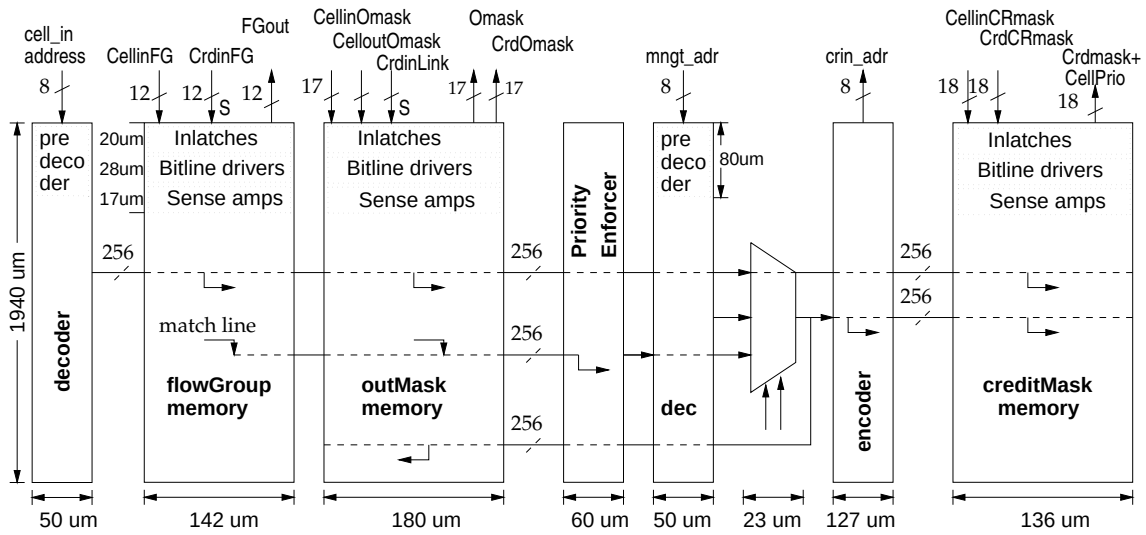
3.1 Περιγραφή του τμήματος full-custom

Χρησιμοποιήσαμε την τεχνολογία CMOS 0.35 micron της SGS Thomson, Crolles, Γαλλία. Είναι σχεδιασμένη για εφαρμογές των 3.3V (-0.6/+0.3V). Είναι τεχνολογία ενός polysilicon, twin-tub CMOS process, με πέντε επίπεδα μετάλλου. Το χαρακτηριστικό της είναι το μήκος της πύλης που είναι 0.35 μm (NMOS & PMOS).

Το κυρίως τμήμα full-custom αποτελείται από τρεις μνήμες (την flowGroup, την outMask και την creditMask), δύο αποκωδικοποιητές, έναν αποκωδικοποιητή προτεραιότητας, έναν κανονικό κωδικοποιητή καθώς και τα απαραίτητα περιφερειακά κυκλώματα για να υποστηριχτεί η σωστή λειτουργία. Ένα αφηρημένο διάγραμμα και η φυσική τους τοποθέτηση φαίνεται στο σχήμα 3.1. Ένα άλλο κομμάτι σε full-custom είναι η λίστα ελεύθερων θέσεων, αποτελούμενη από μία αλυσίδα από 256 D-flip-flops και έναν αποκωδικοποιητή προτεραιότητας. Τέλος, ένα ξεχωριστό κύκλωμα σχεδιασμένο σε full-custom αποτελεί και η τετράπορτη μνήμη του αρχείου καταχωρητών δεικτών αρχής και τέλους.

3.1.1 Η μνήμη creditMask

Το κύτταρο μνήμης της μνήμης που αποθηκεύει τις μάσκες πιστώσεων είναι ένα συμβατικό κύτταρο SRAM δύο θυρών. Το μέγεθός του είναι 7.25 μm x 7.3 μm. Αν και υπάρχουν γεννήτριες μνημών στο πακέτο σχεδίασης ακόμα και δίπορτες, η ειδική λειτουργία μεταβολής ενός κυττάρου και ανάγνωσης των υπόλοιπων δημιούργησε την ανάγκη για ένα



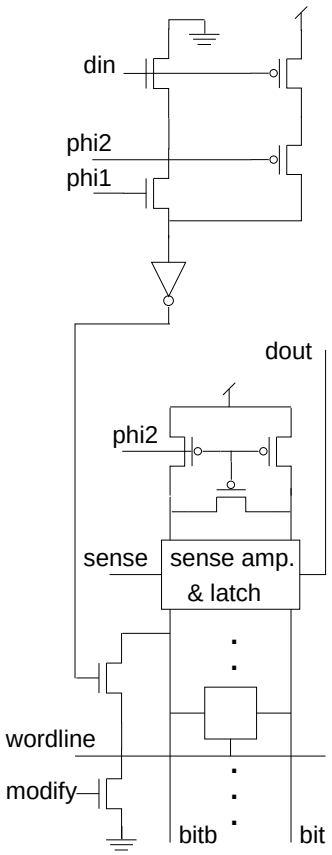
Σχήμα 3.1: *Το κυρίως τμήμα full-custom*

ελαφρώς διαφορετικό περιφερειακό κύκλωμα: μόνο οι bitlines των bit που είναι 1 τραβιούνται κάτω σε χαμηλό δυναμικό για να θέσουν τα κύτταρα που έχει ενεργοποιήσει η γραμμή λέξης, ενώ ταυτόχρονα όλα τα υπόλοιπα διαβάζονται (σχήμα 3.2). Επίσης, όσο αφορά τη δεύτερη θύρα, δεν υπάρχουν αισθητήριοι ενισχυτές και μανταλωτές εξόδου, αφού χρησιμοποιείται μόνο για εγγραφές, κάνοντας έτσι οικονομία από άποψη χώρου και ισχύος.

3.1.2 Οι μνήμες flowGroup και outMask

Η μνήμη flowGroup έχει 3072 στοιχεία προσεταιριστικής μνήμης, οργανωμένα σε ένα πίνακα 256 επί 12-bit λέξεις. Καθένα από αυτά αποτελείται από 9 τρανζίστορ όπως φαίνεται στο σχήμα 3.3. Το πάνω μέρος είναι ένα συμβατικό κύτταρο στατικής RAM των 6 τρανζίστορ, ενώ το σκιασμένο μέρος αποτελεί τη λογική αποκλειστικού-ή των 3 τρανζίστορ. Είναι πλήρως δίπορτο, με δύο ζευγάρια bitlines. Το μέγεθός του είναι 7.25 μm x 11.1 μm.

Αν και η έκδοση τύπου NAND (σχήμα 3.4) ενός κυττάρου CAM καταλαμβάνει μικρότερο χώρο, απορρίφθηκε σαν επιλογή λόγω της αργότερης εκφόρτισης της γραμμής ταιριάσματος (υπάρχουν δύο τρανζίστορ στη σειρά και μία επιπλέον επαφή ανά κύτταρο) [NC92].

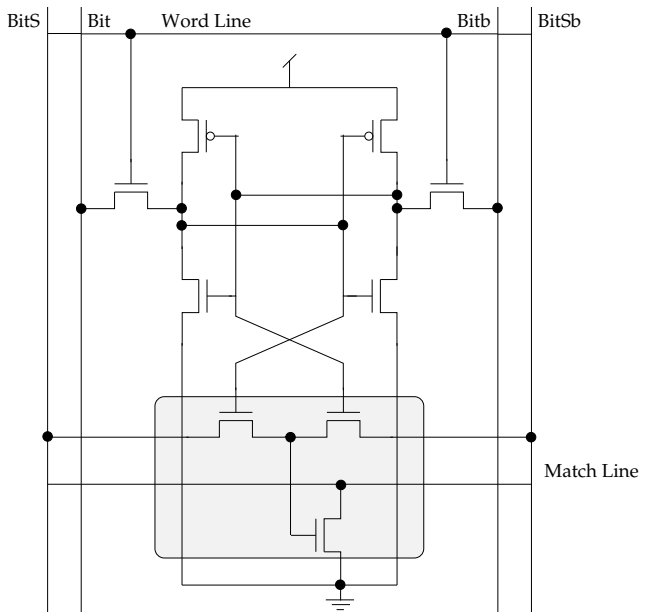


Σχήμα 3.2: *Το γενικό διάγραμμα της μνήμης creditMask*

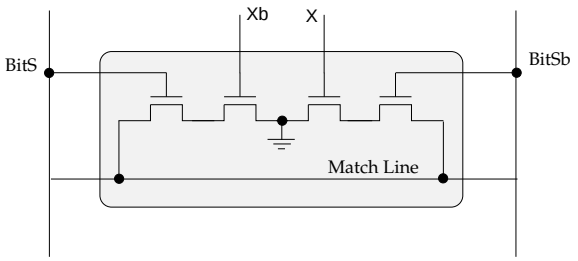
Ο οδηγητής των γραμμών ανίχνευσης και ο οδηγητής του σήματος **search** (ενεργό όταν είναι μηδέν) απεικονίζονται στο σχήμα 3.5.

Η μνήμη outMask έχει 4352 κύτταρα μνήμης, οργανωμένα σε ένα πίνακα 256 επί 17-bit λέξη. Η πρώτη πόρτα χρησιμοποιείται για κανονικές προσπελάσεις ανάγνωσης εγγραφής, η δεύτερη πόρτα για αλλαγή και ανάγνωση, ενώ η τρίτη χρησιμοποιείται για ένα ειδικό είδος διευθυνσιοδότησης μέσω περιεχομένου. Όταν ψάχνουμε τη μνήμη outMask, οποιαδήποτε αποθηκευμένη λέξη που έχει ένα ή περισσότερα bit 1 κοινά με τη λέξη που ψάχνουμε θα δώσει ένα επιτυχές αποτέλεσμα. Έτσι μία μοναδική bitline (match enable) ανά στήλη χρησιμοποιείται κατά το ψάξιμο της μνήμης (σχήμα 3.6) [S.S91]. Το κύτταρο της μνήμης καταλαμβάνει χώρο 7.25 μm x 10.25 μm.

Είναι γενικώς αποδεκτό ότι η μοντελοποίηση διαφόρων στοιχείων της μνήμης είναι λιγότερο ακριβής παρά την κατασκευή μίας πραγματικής δομής και προσπαθώντας να ενσωματώσουμε σενάρια με τις δυσκολότερες περιπτώσεις κάτω από τις χειρότερες συνθήκες. Κάτω από αυτή την προοπτική κατασκευάστηκε ένα κύκλωμα μνήμης αποτελούμενο

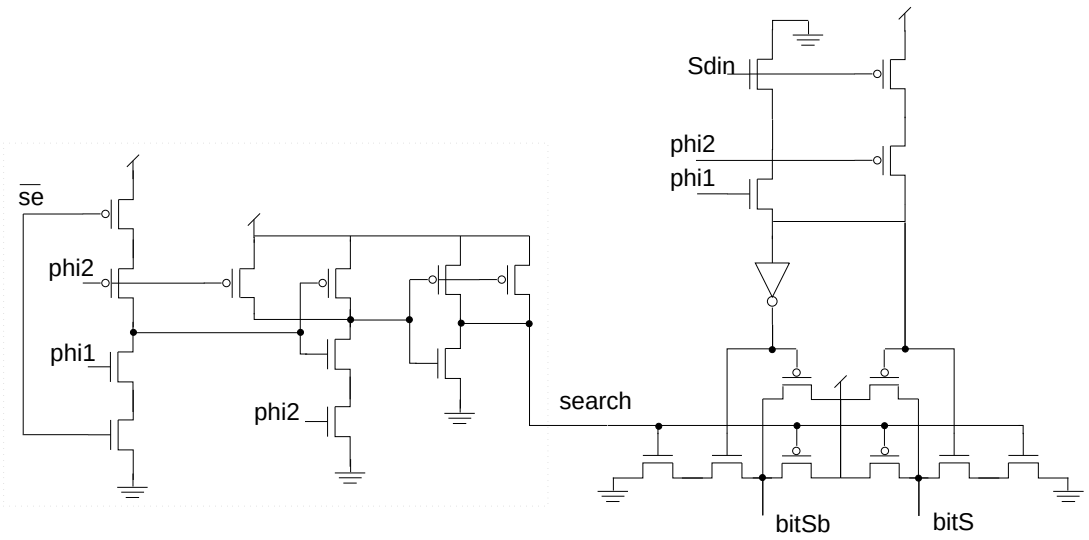


Σχήμα 3.3: *Το δίπορτο κύτταρο της μνήμης flowGroup*



Σχήμα 3.4: *Συγκριτής ενός κυττάρου μνήμης CAM βάσει λογικής τύπου NAND*

από λέξη εύρους 30 bits (έναντι των πραγματικών 29 bits), 12 κύτταρα της μνήμης flowGroup και 18 της outMask. Οι γραμμές των λέξεων οδηγούνταν από πραγματικούς οδηγητές από αποκωδικοποιητή και οι γραμμές ταιριάσματος ήταν φορτωμένες με το πραγματικό φορτίο που θα οδηγείται στο τελικό κομμάτι μνήμης. Τα κύτταρα μνήμης υπό εξαντλητικό έλεγχο (1, 2, 13, 14) βρίσκονταν στο σημείο διασταύρωσης της οριζόντιας γραμμής των κυττάρων με διπλές στήλες των 256 κυττάρων, έτσι ώστε οι bitlines να υπόκεινται στο κανονικό πραγματικό φορτίο. Στην κορυφή της λέξης των 30 bits υπάρχουν τα κυκλώματα οδήγησης των bitlines, οι αισθητήριοι ενισχυτές, οι μανταλωτές εισόδου και εξόδου, οι οδηγητές των ρολογιών των σημάτων εγγραφής, ψαξίματος και ενεργοποίησης των αισθητήριων ενισχυτών.

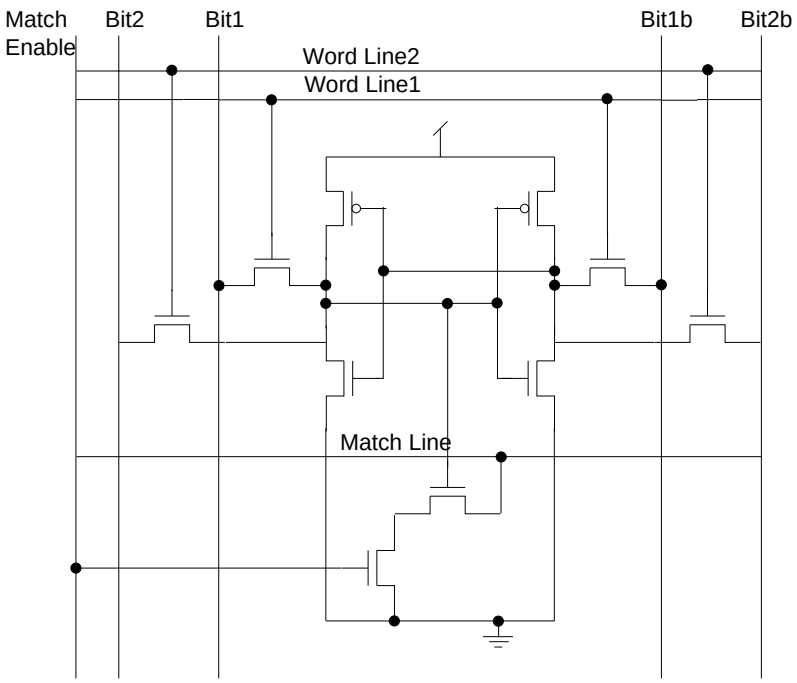


Σχήμα 3.5: *Ο οδηγητής των bitlines ανίχνευσης στη μνήμη flowGroup*

Ο ακόλουθος πίνακας (πίνακας 3.1.2) απεικονίζει ένα από τα πιο αντιπροσωπευτικά σενάρια που εφαρμόστηκαν στην παραπάνω δομή για έλεγχο. Όλες οι δυνατές και κρίσιμες λειτουργίες έχουν συμπυκνωθεί σε ένα διάστημα προσομοίωσης 6 συνεχόμενων κύκλων ρολογιού των 20 ns.

cycle	operation	Di_1/Si_1	Di_2/Si_2,…,Di_{12}/Si_{12}	Di_{13}/Si_{13}	Di_{14}/Si_{14}	Di_{15}/Si_{15},…,Di_{30}/Si_{30}	comments
1	write FG,OM	1 0	0 0, …, 0 0	1 0	1 0	0 0,…, 0 0	initialize word
2	read FG,OM search OM	1 0	0 0, …, 0 0	1 1	1 1	0 0,…, 0 0	succesful match
3	read FG,OM search FG	1 0	0 0, …, 0 0	1 0	1 0	0 0,…, 0 0	unsuccessful match
4	read FG,OM mod/rd OM	1 0	0 0, …, 0 0	0 0	1 0	0 0, …, 0 0	reset Di13
5	read FG,OM	1 0	0 0, …, 0 0	0 0	1 0	0 0, …, 0 0	read from both OM ports
6	read FG,OM search OM	1 0	0 0, …, 0 0	0 1	1 0	0 0, …, 0 0	unsuccessful match

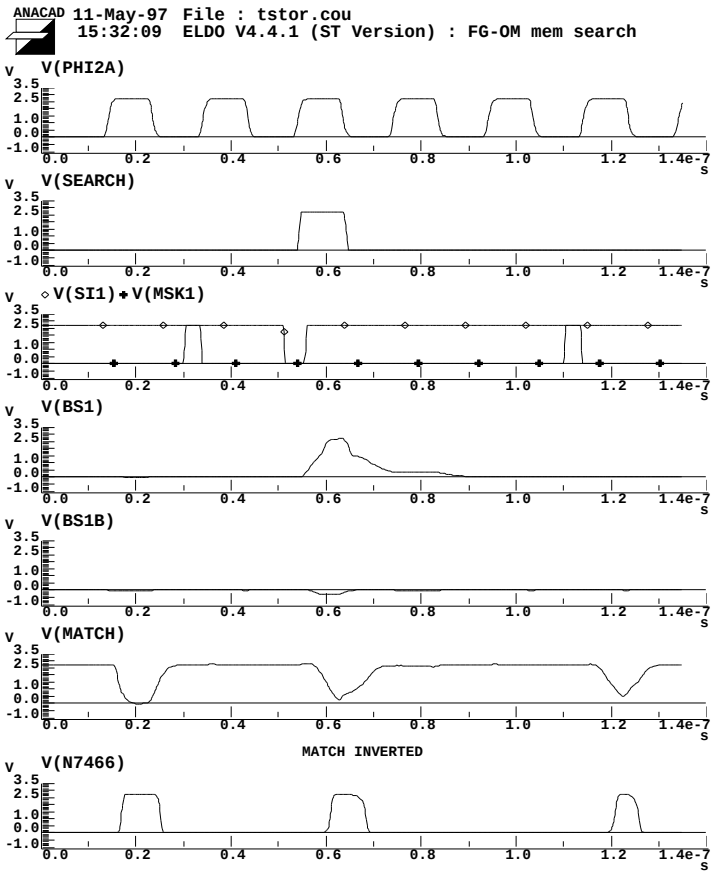
Το σχήμα 3.7 δείχνει τα αποτελέσματα προσομείωσης που επαληθεύουν την ορθή λειτουργία. Μία επικίνδυνη κατάσταση μπορεί να συμβεί αν κατά τη διάρκεια ψαξίματος



Σχήμα 3.6: *Το κύτταρο μνήμης τριών πορτών της μνήμης outMask*

στη μνήμη flowGroup όλα τα κύτταρα μίας στήλης έχουν αποθηκευμένη την ίδια τιμή με αυτήν του συγκεκριμένου bit που ψάχνουμε, εκτός από ένα μοναδικό που πρέπει να εκφορτίσει τη γραμμή ταιριάσματος. Στην περίπτωση αυτή αποτελέσματα προσωμοιώσεων δείξαν ότι ένας μέτριος οδηγητής των bitlines ψαξίματος ήταν ικανοποιητικός, συνυπολογίζοντας βεβαίως και τις παρασιτικές χωρητικότητες λόγω διασύνδεσης. Εντούτοις, αν το κύτταρο μνήμης βρίσκεται στην αντίθετη κατάσταση, ένα μικρό ρεύμα μεγέθους $4e^{-6}$A διαρρέει από την bitline ψαξίματος. Όσο και αν φαίνεται ασήμαντο, αν όλα τα 255 κύτταρα έχουν αυτή την αντίθετη πολικότητα προς το bit που ψάχνεται τότε ένα ρεύμα διαρροής, μεγέθους περίπου 1 mA είναι παραπάνω από αρκετό για να καταρρίψει την αρχική μας υπόθεση για το ρεύμα που πρέπει να παρέχει ο οδηγητής για να φορτίσει την bitline ψαξίματος. Η σχετικά αργή ανύψωση της bitline ‘‘BS1’’ οφείλεται σε αυτό το φαινόμενο.

Όπως φαίνεται μία λειτουργία ψαξίματος που αποτυγχάνει προκαλεί σχεδόν πλήρες κατέβασμα της γραμμής ψαξίματος, κάνοντας φανερό λοιπόν ότι δεν υπάρχει ανάγκη για έναν αισθητήριο ενισχυτή. Είναι επίσης φανερό ότι η γραμμή ψαξίματος μίας λέξης που γράφεται εκφορτίζεται για να αποφευχθούν καταστάσεις ανταγωνισμού όπου μία λέξη

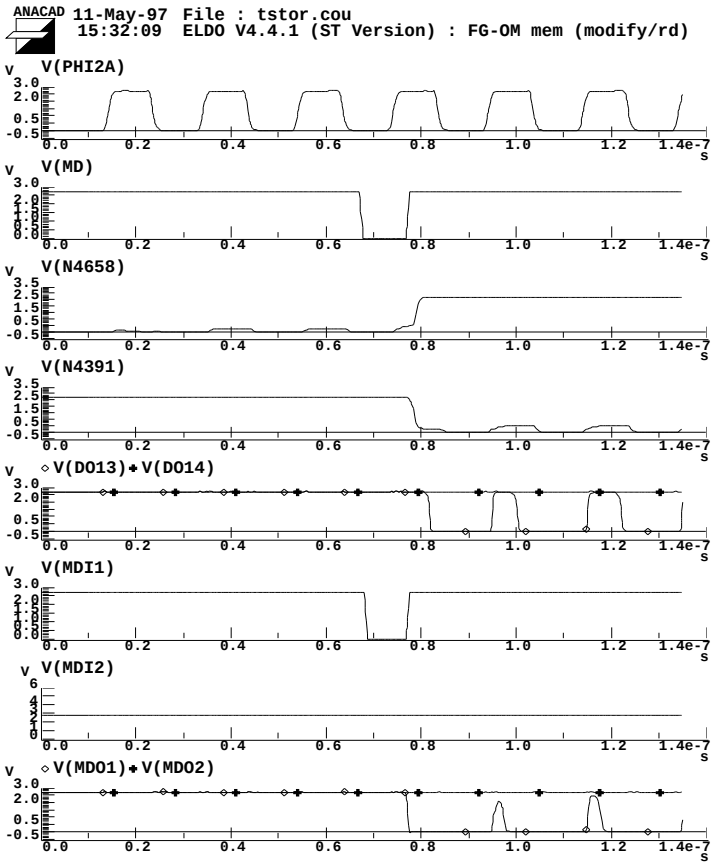


Σχήμα 3.7: *Κυματομορφές ανίχνευσης στις μνήμες flowGroup-outMask (χειρότερη περίπτωση : διαφορά σε ένα μονο bit)*

γράφεται και ψάχνεται ταυτόχρονα στον ίδιο κύκλο. Λαμβάνοντας υπόψη τον ιδιαίτερο τρόπο διευθυνσιοδότησης της μνήμης outMask, ένα ψάξιμο μπορεί να έχει ένα επιτυχές αποτέλεσμα αν ψάξουμε για μία λέξη σχηματιζόμενη από ένα υπάρχον flow group και όλα τα άλλα μηδενικά. Η σωστή λειτουργία όμως της λίστας κυττάρων χωρίς πιστώσεις (CLL) αποτρέπει εξ' ολοκλήρου τέτοια λανθανσμένη λειτουργία.

Στο σχήμα 3.8 φαίνονται οι κυματομορφές που αντιστοιχούν στη λειτουργία αλλαγής/ανάγνωσης που συμβαίνει στην μνήμη outMask. Τα σήματα ''DO'' είναι οι έξοδοι μιας κοινής θύρας ram, ενώ τα ''MDO'' της θύρας αλλαγής/ανάγνωσης. Αν ταυτόχρονα ένα κύτταρο μνήμης αλλάζει και διαβάζεται από την δεύτερη πόρτα έγκυρα δεδομένα μπορούμε να πάρουμε μετά από 7.3ns μετά την ανερχόμενη ακμή του phi2. Στον πέμπτο κύκλο συμβαίνει ανάγνωση και από τις δύο πόρτες στον ίδιο κύκλο και έχουμε ένα έγκυρο

αποτέλεσμα μετά από 5.7ns στις εξόδους ‘‘DO’’ ενώ στις εξόδους ‘‘MDO’’ μετά από 3.2ns. Η χειρότερη περίπτωση εμφανίζεται όταν το κύτταρο εξυπηρετεί και την πόρτα ψαξίματος στον ίδιο κύκλο. Ένα διάβασμα από την τυπική ram πόρτα έχει τώρα χρόνο προσπέλασης 8 ns, ενώ το διάβασμα από την πόρτα αλλαγής/ανάγνωσης έχει χρόνο 3.7 ns.

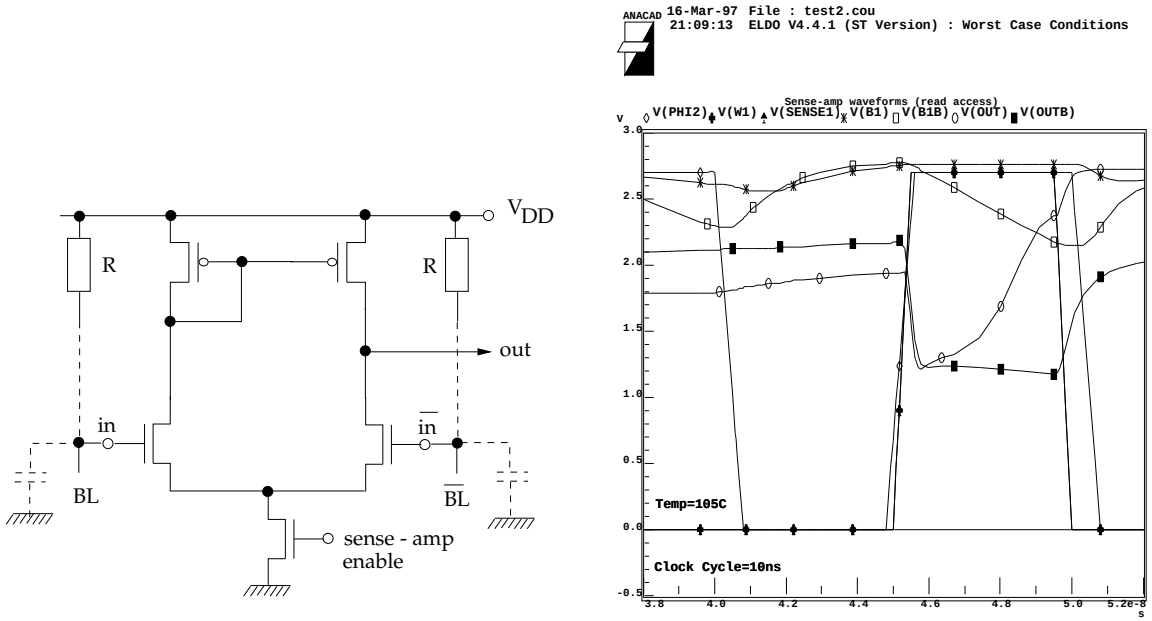


Σχήμα 3.8: *Λειτουργία αλλαγής/ανάγνωσης στη μνήμη outMask*

3.1.3 Περιφερειακά κυκλώματα

Όπως φαίνεται στο σχήμα 3.1 οι αισθητήριοι ενισχυτές των μνημών και οι μανταλωτές εξόδου βρίσκονται τοποθετημένοι μεταξύ των οδηγητών των bitlines και του πυρήνα της μνήμης. Έτσι πετυχαίνουμε σημαντική ελάττωση στην απόσταση όπου πρέπει να διανύσει το ρολόι και πιθανά προβλήματα σχετικών καθυστερήσεων. Οι προ-αποκωδικοποιητές διευθύνσεων και οι αισθητήριοι ενισχυτές ενεργοποιούνται για να λειτουργήσουν μόνο

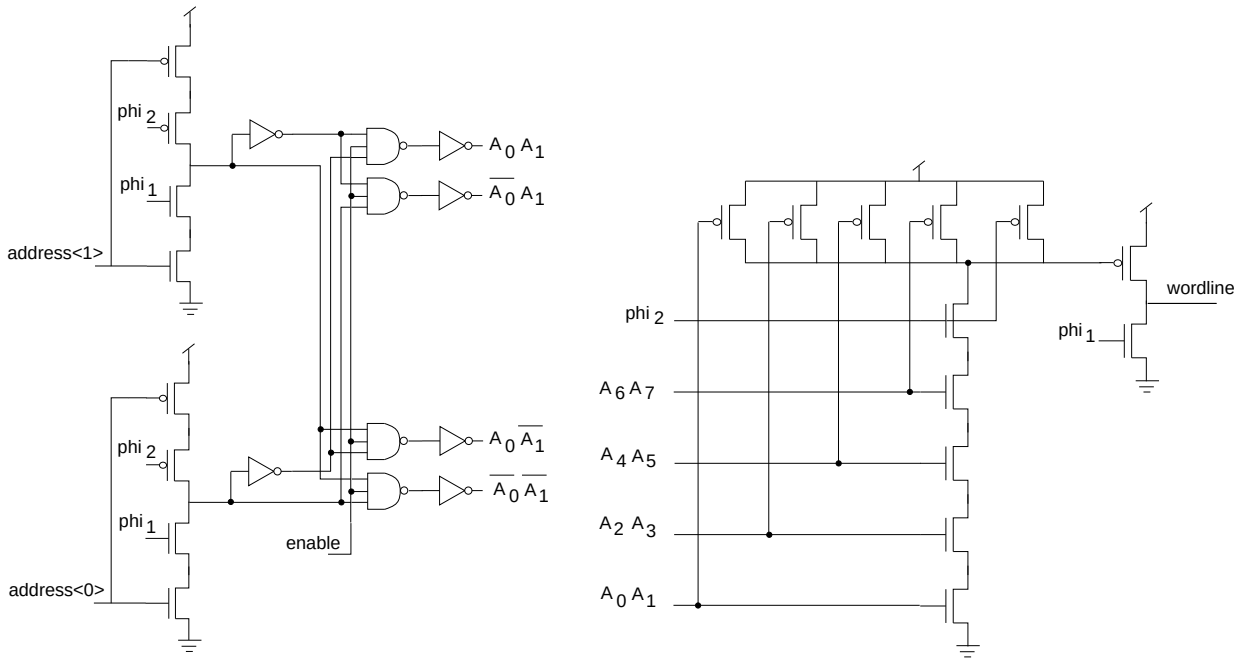
κατά τους κύκλους που χρειάζονται, κάνοντας έτσι οικονομία στην κατανάλωση ισχύος. Επιπλέον το σήμα ενεργοποίησης των αισθητήριων ενισχυτών είναι χρήσιμο για να κόβει το ρεύμα DC που καταναλώνεται από τον αισθητήριο ενισχυτή στη φάση ρολογιού που δεν χρησιμοποιείται αυτός, εξοικονομώντας έτσι ενέργεια που κυμαίνεται από 40 έως 60 % ανάλογα με τις συνθήκες λειτουργίας (τάση τροφοδοσίας, θερμοκρασία περιβάλλοντος).



Σχήμα 3.9: *Συμβατικός αισθητήριος ενισχυτής τύπου τάσης*

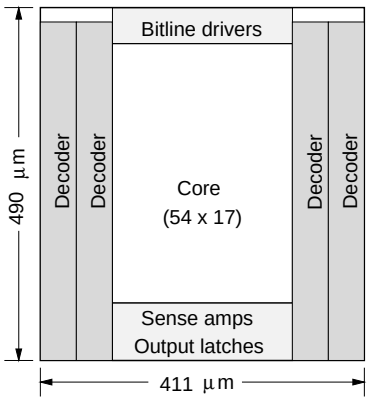
Χρησιμοποιήθηκε ένας απλός αισθητήριος ενισχυτής τάσης, ο οποίος έχει τη δυνατότητα αίσθησης με χαμηλή ισχύ, σε βάρος όμως της ταχύτητας. Βεβαίως, πιθανοί υποψήφιοι στάθηκαν αισθητήριοι ενισχυτές δύο και τριών βαθμίδων με διάφορες μεθόδους αντιμετώπισης του προβλήματος του αποτελέσματος miller και ιδανικοί για υψηλές ταχύτητες. Όμως οι επιπλέον βαθμίδες και τα σχετικά μονοπάτια ρεύματος τείνουν να έχουν ως αποτέλεσμα υψηλότερη κατανάλωση ισχύος. Αντίθετα, για μια απόδοση μέτριου κέρδους που ήταν αρκετή εδώ ένας αισθητήριος ενισχυτής τάσης που καταλαμβάνει ελάχιστο χώρο ήταν μία ικανοποιητική λύση. Επιπλέον, συνυπολογίζοντας τους περιορισμούς χρονου που είχαμε (περίοδο ρολογιού των 20ns) η επιλογή αυτή ήταν πιο ικανοποιητική σε σχέση με τους πιο γρήγορους αλλά πιο πολύπλοκους αισθητήριους ενισχυτές.

Η αποκωδικοποίηση διευθύνσεων επιτυγχάνεται μέσω ενός αποκωδικοποιητή 8 σε 256, χρησιμοποιώντας έναν προ-αποκωδικοποιητή για κάθε 2 bit. Όπως φαίνεται στο σχήμα 3.10, ένα σήμα **enable** χρησιμοποιείται για να ενεργοποιήσει τον αποκωδικοποιητή.



Σχήμα 3.10: *Προ-αποκωδικοποιητής, αποκωδικοποιητής και μανταλωτές διευθύνσεων*

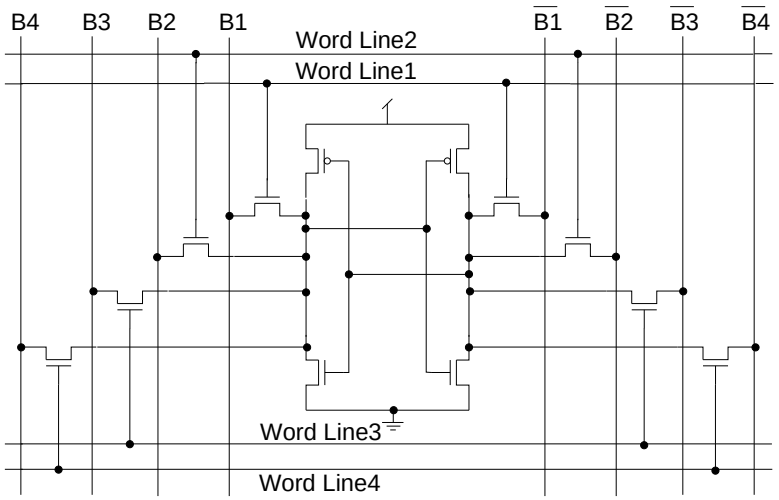
3.1.4 Η μνήμη του αρχείου καταχωρητών με τους δείκτες αρχής και τέλους



Σχήμα 3.11: *Διάταξη της μνήμης head-tail register file*

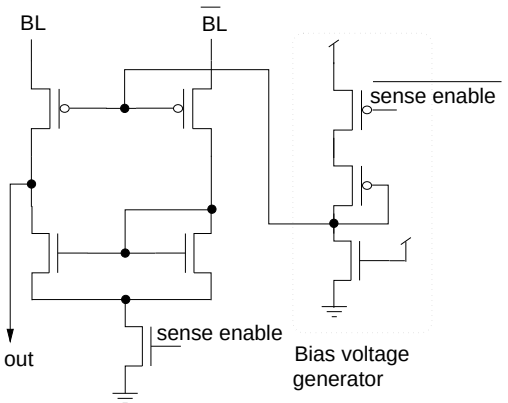
Η μνήμη HTRF είναι ένα ξεχωριστό τμήμα κυκλώματων σχεδιασμένο σε full-custom, που δεν περιλαμβάνεται στο σχήμα 3.1. Χρησιμοποιεί ένα τετράπορτο κύτταρο μνήμης SRAM που απεικονίζεται στο σχήμα 3.12. Λόγω του μικρού μεγέθους της (54 επί 17-bit λέξεις) η λειτουργία της ήταν εφικτή, επιτρέποντας ταυτόχρονες προσπελάσεις στην ίδια

λέξη από διαφορετικές πόρτες, αρκεί να μην υπάρχουν δύο εγγραφές. Ένα κύτταρο καταλαμβάνει χώρο 7.25 μm x 14.1 μm.



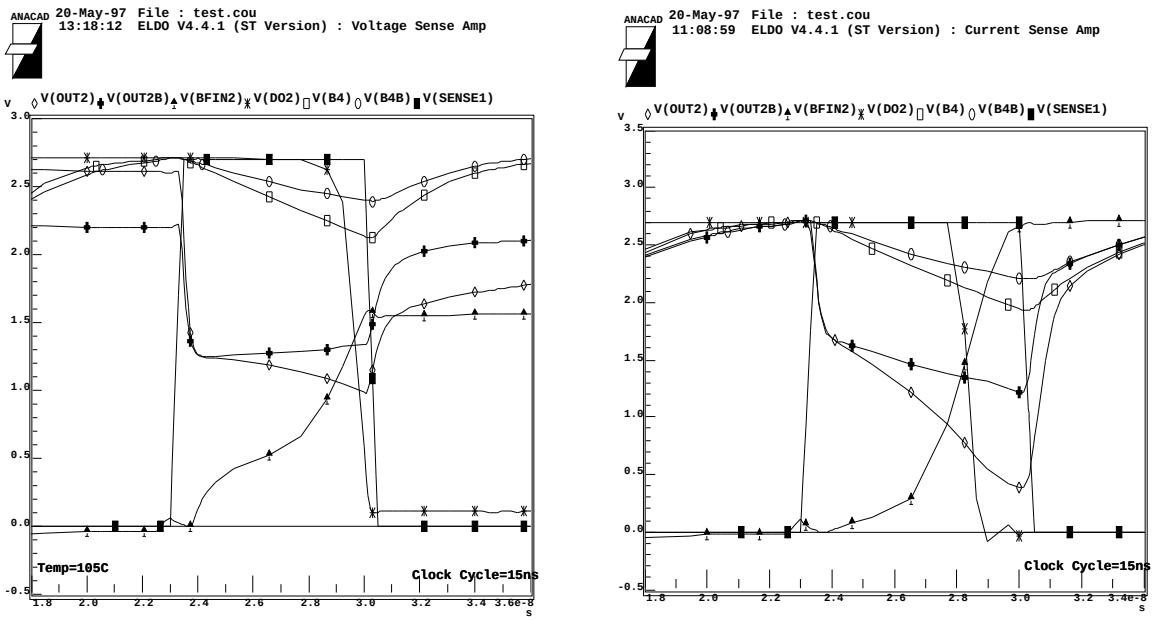
Σχήμα 3.12: *Το κύτταρο μνήμης της HTRF*

Η διάταξη του τμήματος της μνήμης φαίνεται στο σχήμα 3.11. Μία προσέγγιση εξισορρόπησης οδήγησε στην τοποθέτηση των δύο αποκωδικοποιητών αριστερά του πυρήνα της μνήμης και τους άλλους δύο δεξιά. Ακόμη, οι αισθητήριοι ενισχυτές τώρα είναι τοποθετημένοι στο κάτω μέρος, μια και δεν υπήρχε πλέον λόγος να τοποθετηθούν μαζί με τους οδηγητές των bitlines (το κατακόρυφο μήκος της μνήμης δεν είναι σχετικά μεγάλο).



Σχήμα 3.13: *Ο αισθητήριος ενισχυτής ρεύματος (PMOS-bias type) στη μνήμη HTRF*

Με σκοπό να γλυτώσουμε τα εξωτερικά κυκλώματα ελέγχου για να αποφευχθούν ταυτόχρονες προσπελάσεις στην ίδια λέξη μνήμης, υιοθετήθηκε ένας διαφορετικός αισθητήριος ενισχυτής από αυτούς που χρησιμοποιήθηκαν μέχρι στιγμής. Όπως ειπώθηκε ήδη οι αισθητήριοι ενισχυτές τάσης παρουσιάζουν φτωχά χαρακτηριστικά κέρδους τάσης ειδικά όταν είναι μονού άκρου. Γι' αυτό χρησιμοποιήθηκε ένας αισθητήριος ενισχυτής ρεύματος τύπου PMOS-bias (σχήμα 3.13), όμοιος στην αρχή λειτουργίας του με αυτόν που έχει αναφερθεί στο [SIU+92]. Αν και χρησιμοποιήθηκε ένας μονού άκρου, αποδείχτηκε ικανοποιητικός για τις συγκεκριμένες ανάγκες της μνήμης HTRF.



Σχήμα 3.14: *Κυματομορφές προσομοίωσης συγκρίνοντας τους δύο αισθητήριους ενισχυτές (voltage vs. current)*

Ερευνήθηκαν τα χαρακτηριστικά και των δύο αισθητήριων ενισχυτών με προσομειώσεις κυκλωμάτων με βάση μια δομή μνήμης αποτελούμενη από 384 κύτταρα μνήμης δύο πορτών, σχηματίζοντας μία ψηλή στήλη έτσι ώστε να επιτευχθεί μία μικρή απόκλιση των bitlines. Το ρεύμα και στους δύο αισθητήριους ενισχυτές ήταν 0.06 mA με τάση τροφοδοσίας 2.7V και θερμοκρασία 105^oC. Οι κυματομορφές της προσομοίωσης που είναι σχεδιασμένες στη φιγούρα 3.14 αποδεικνύουν την υπεροχή του αισθητήριου ενισχυτή ρεύματος. Όπως φαίνεται ο αισθητήριος ενισχυτής τάσης αντιμετωπίζει δυσκολία στην αναγνώριση της διαφοράς τάσης στις bitlines όταν η περίοδος του ρολογιού είναι κοντά στα 15 ns. Αντίθετα ο πρώτος πετυχαίνει χρόνο προσπέλασης ίσο με 5.8 ns. Τέλος έχει περίπου

το τριπλάσιο κέρδος του αισθητήριου ενισχυτή τάσης χωρίς αύξηση χώρου ή πολυπλοκότητας.

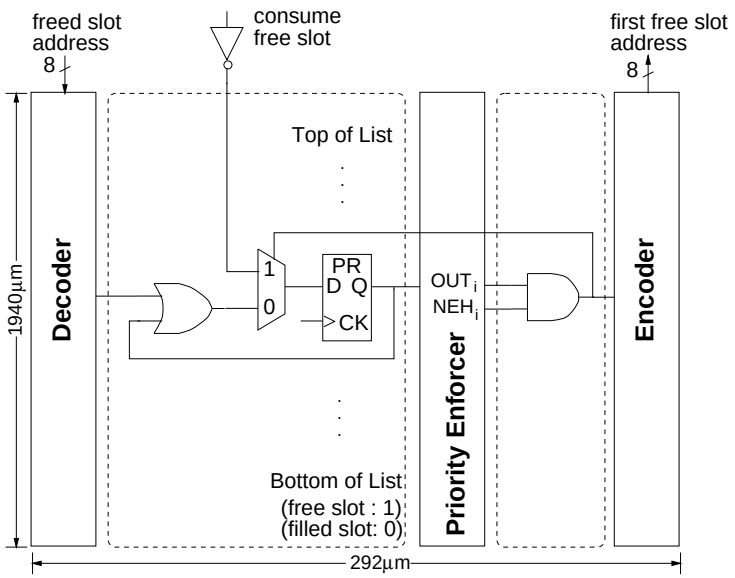
Με αυτή την τροποποίηση, τα αποτελέσματα προσομοίωσης ολόκληρης της μνήμης HTRF στις χειρότερες συνθήκες έδειξαν ότι ταυτόχρονη ανάγνωση της ίδιας λέξης και από τις 4 πόρτες έχει χρόνο προσπέλασης 4.9 ns. Μάλιστα αν η κατάσταση του κυττάρου μνήμης αλλάζει από τη μία πόρτα και οι υπόλοιπες 3 εκτελούν μία ανάγνωση στο ίδιο κύτταρο, έγκυρο αποτέλεσμα παράγεται μετά από 6.8 ns από την ανερχόμενη ακμή του ρολογιού.

3.1.5 Η λίστα ελεύθερων θέσεων

Ένα άλλο ξεχωριστό κύκλωμα σχεδιασμένο με full-custom τεχνική σχεδίασης είναι αυτό που απαιτείται για την υλοποίηση της λίστας ελεύθερων θέσεων (Free List). Αποτελείται από ένα διάνυσμα 256-bit που διατηρεί την κατάσταση των ελεύθερων θέσεων στον κοινόχρηστο ενταμιευτή δεδομένων. Αρχικά σημειώνονται όλες οι θέσεις ως ελεύθερες, ‘‘ένα’’. Ένα ψάξιμο για μία άδεια θέση γίνεται από έναν αποκωδικοποιητή προτεραιότητας, ο οποίος παράγει το πρώτο ένα που βρήκε από την κορυφή του διανύσματος. Μέχρι το τέλος του κύκλου αλλάζει την κατάσταση του αντίστοιχου flip-flop και τροφοδοτεί μία διεύθυνση κωδικοποιημένη σε 8-bits. Ένας επιπλέον αποκωδικοποιητής είναι απαραίτητος για να αποκωδικοποιήσει τη διεύθυνση της θέσης του ενταμιευτή που μόλις ελευθερώθηκε και να σημειωθεί στην αντίστοιχη θέση του διανύσματος της Free List. Το κκλωμα καταλαμβάνει χώρο 1940 μm x 292 μm.

3.1.6 Ένας γραμμικός αποκωδικοποιητής προτεραιότητας

Στις αρχιτεκτονικές με CAM υφίσταται το πρόβλημα επιλογής ενός στοιχείου από ένα σύνολο εναλλακτικών επιλογών και έχουν αναφερθεί μέθοδοι επίλυσής του ως επιλυτής πολλαπλών ταιριασμάτων, δίκτυο προτεραιοτήτων, κωδικοποιητής προτεραιότητας, κωδικοποιητής κανονικοποίησης ή διαιτητής αιτήσεων. Το πρόβλημα επίλυσης πολλαπλών αιτήσεων αναδύεται όταν το ψάξιμο σε μία προσεταιριστική μνήμη παράγει περισσότερα από ένα ισοδύναμα αποτελέσματα στη μονάδα του χρόνου. Για να επιλεγεί ένα από αυτά το σύστημα πρέπει να είναι ικανό να γεννήσει ένα **1** για την επιθυμητή επιλογή και μηδέν για όλα τα υπόλοιπα. Ή αλλιώς το πρόβλημα μπορεί να θεωρηθεί ότι είναι να διαλέξεις



Σχήμα 3.15: *Το κύκλωμα της λίστας ελεύθερων θέσεων*

από ένα διάνυσμα A μεγέθους N-bit, ένα μόνο στοιχείο που να είναι ένα λογικό *ένα*. Ανάλογα με την περίπτωση αυτή η εκλογή μπορεί να υποδεικνύεται με μία απευθείας διεύθυνση i, ένα διάνυσμα P (pointer) στο οποίο μόνο ένα P(i) είναι ένα, ή ένα διάνυσμα I (inhibit), τέτοιο ώστε μόνο το $A(i) \wedge \overline{I(i)}$ να είναι ένα.

Στο τμήμα διαχείρισης πολλαπλών ουρών του ATLAS I, ένας αποκωδικοποιητής προτεραιότητας κρίνεται απαραίτητος για τη σωστή λειτουργία των μνημών βάση περιεχομένου, αφού πολλές λέξεις μπορεί να ταιριάξουν κατά τη διάρκεια μίας λειτουργίας ψαξίματος. Έτσι ένας ελεγκτής πολλαπλών ταιριασμάτων πρέπει να έχει τον έλεγχο να επιλέγει και να προωθεί μόνο μία απόκριση. Το διάστημα χρόνου για να παράγει ένα μοναδικό αποτέλεσμα πρέπει να είναι μικρότερο ή ίσο με τον κύκλο ρολογιού της προσεταιριστικής μνήμης, έτσι ώστε να μπορεί να λειτουργεί με το ρυθμό λειτουργίας της μνήμης.

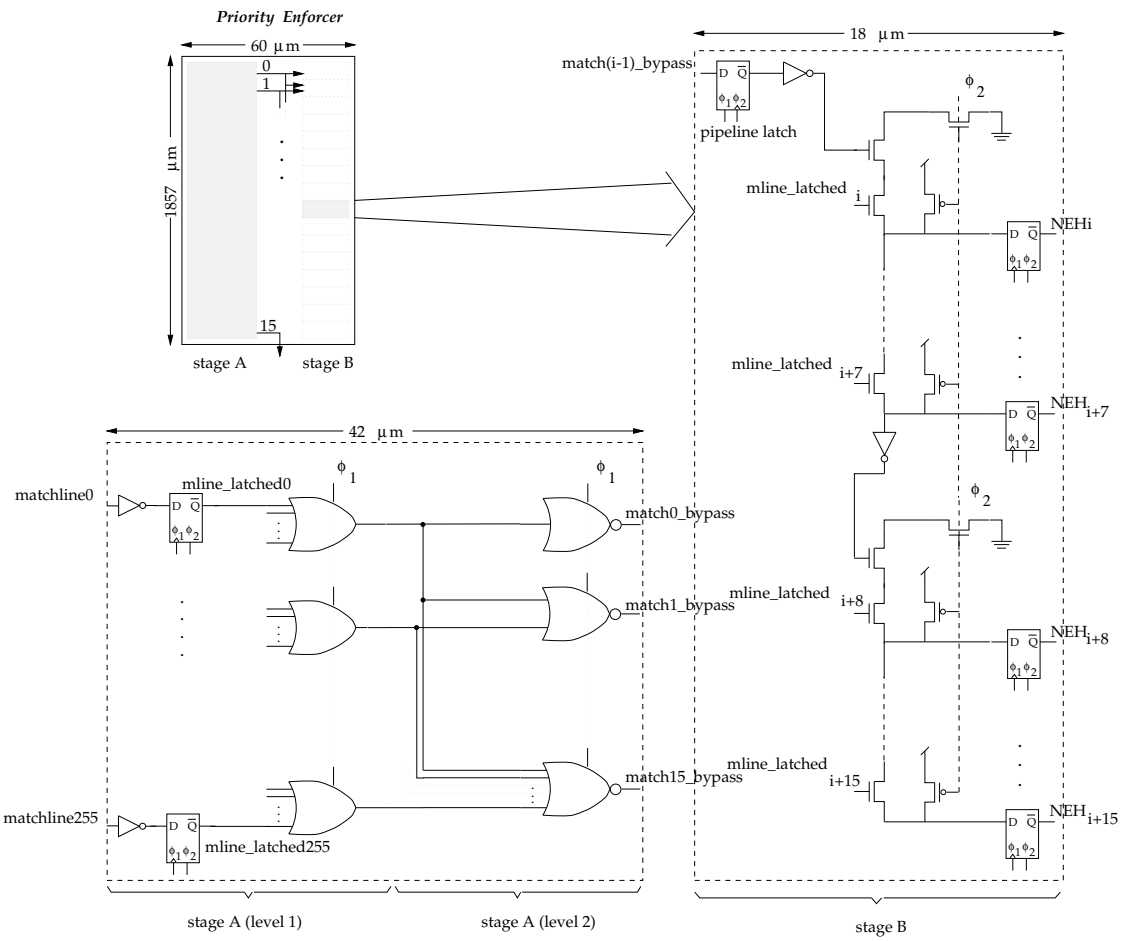
Όπως φαίνεται στο σχήμα 3.16 η είσοδος στον αποκωδικοποιητή προτεραιότητας είναι ένα διάνυσμα 256-bit αποτελούμενο από τις γραμμές ταιριάσματος από τις μνήμες flowGroup και outMask, υποδηλώνοντας αυτές τις λέξεις που ταιριάσαν με τη λέξη προς ψάξιμο. Ο αποκωδικοποιητής προτεραιότητας παράγει ένα διάνυσμα του ίδιου μεγέθους, όπου υπάρχει ένα και μόνο ‘‘1’’, αντιστοιχώντας στο πρώτο ένα του διανύσματος εισόδου. Ο υπολογισμός ενός bit της εξόδου αναπόφευκτα χρειάζεται να λάβει υπόψη του όλες τις παραπάνω του εισόδους. Αν καμιά από αυτές δεν είναι ένα ταίριασμα αλλά ούτε και η συγκεκριμένη αντίστοιχη είσοδος τίθεται ένα σήμα ονομαζόμενο Nobody-Else-Higher

(NEH) και δείχνει αυτό το αποτέλεσμα. Ουσιαστικά τότε μία έξοδος είναι το $OUT_i = NEH_{i-1} \cdot INi$.

Το κύκλωμα του αποκωδικοποιητή προτεραιότητας βασίζεται στις αναφορές των [JC89] και [THY+96]. Η δομή της αλυσίδας ριπής διαφαίνεται η πιο προκλητική επιλογή του πλεονεκτήματος που έχει αναφορικά με το χώρο που καταλαμβάνει και την απόδοσή της. Η απλότητα αυτής της τεχνικής, συνδυαζόμενη με την ευκολία κατασκευής και την δυνατότητα για περαιτέρω βελτίωση αντιστάθμισαν τα πλεονεκτήματα άλλων σχημάτων αποκωδικοποιητών προτεραιότητας ([Koz96]). Όπως διασαφηνίζουν οι Wade και Sodini η απόδοση ενός τέτοιου κυκλώματος πέφτει βαθμιαία καθώς το σήμα NEH πρέπει να διαδοθεί μέσα από μία μακρυά γραμμική δομή.

Αυτό το μειονέκτημα μπορεί να ξεπεραστεί και να επιταχύνουμε τη λειτουργία αυτή χρησιμοποιώντας πρόβλεψη. Πιο συγκεκριμένα, σχεδιάστηκε μία δομή με ένα δέντρο από πύλες OR δύο επιπέδων, που παράγει γραμμές προσπεράσματος, και υλοποιεί ένα είδος πρόβλεψης κρατουμένου, υπολογισμού του πιο νωρίς από την στιγμή που απαιτείται ([WE93]), διαδίδοντας νωρίτερα την ύπαρξη ενός ταιριάσματος. Κατά τη σχεδίαση χρησιμοποιήθηκε δυναμική προφορτιζόμενη λογική με χρονισμό domino, για να μειωθούν οι καθυστερήσεις, ο χώρος και οι ενδιάμεσες μεταβάσεις τάσης. Επιπλέον, ο αποκωδικοποιητής προτεραιότητας χωρίστηκε σε δύο βαθμίδες, η καθεμιά από τις οποίες λειτουργεί σε διαφορετική φάση ρολογιού. Έτσι η μια βαθμίδα προφορτίζεται ενώ η άλλη λειτουργεί (μεταξύ των δύο βαθμίδων υπάρχουν μανταλωτές μίας φάσης). Η πρώτη βαθμίδα περιλαμβάνει ένα δέντρο δύο επιπέδων από δυναμικές πύλες NOR, το οποίο δουλεύει ως μία γεννήτρια προσπερασμάτων των γραμμών ταιριάσματος για τη δεύτερη βαθμίδα pipeline. Οι είσοδοι συνδυάζονται σε ομάδες των 16, και για καθεμιά μία πύλη OR ανιχνεύει αν υπάρχει κάποιο ‘‘1’’. Το επόμενο επίπεδο του δέντρου αποτελείται από πύλες OR πάλι αλλά με αυξανόμενο αριθμό εισόδων κατά την κατεύθυνση διάδοσης του σήματος NEH, το οποίο τελικά βγάζει τα σήματα προσπεράσματος. Το πρώτο γεννάται από έναν απλό αντιστροφέα, ενώ το τελευταίο από μία πύλη OR 16 εισόδων. Η δεύτερη βαθμίδα pipeline αποτελείται από αλυσίδες τύπου Manchester, που λειτουργούν σε ξεχωριστές ομάδες γραμμές ταιριάσματος των 16. Η διάταξη και τα κυκλώματα του αποκωδικοποιητή προτεραιότητας φαίνονται στο σχήμα 3.16. Μία λεπτομερέστατη περιγραφή της αρχιτεκτονικής και της λειτουργίας του αποκωδικοποιητή προτεραιότητας υπάρχουν στο [Koz96].

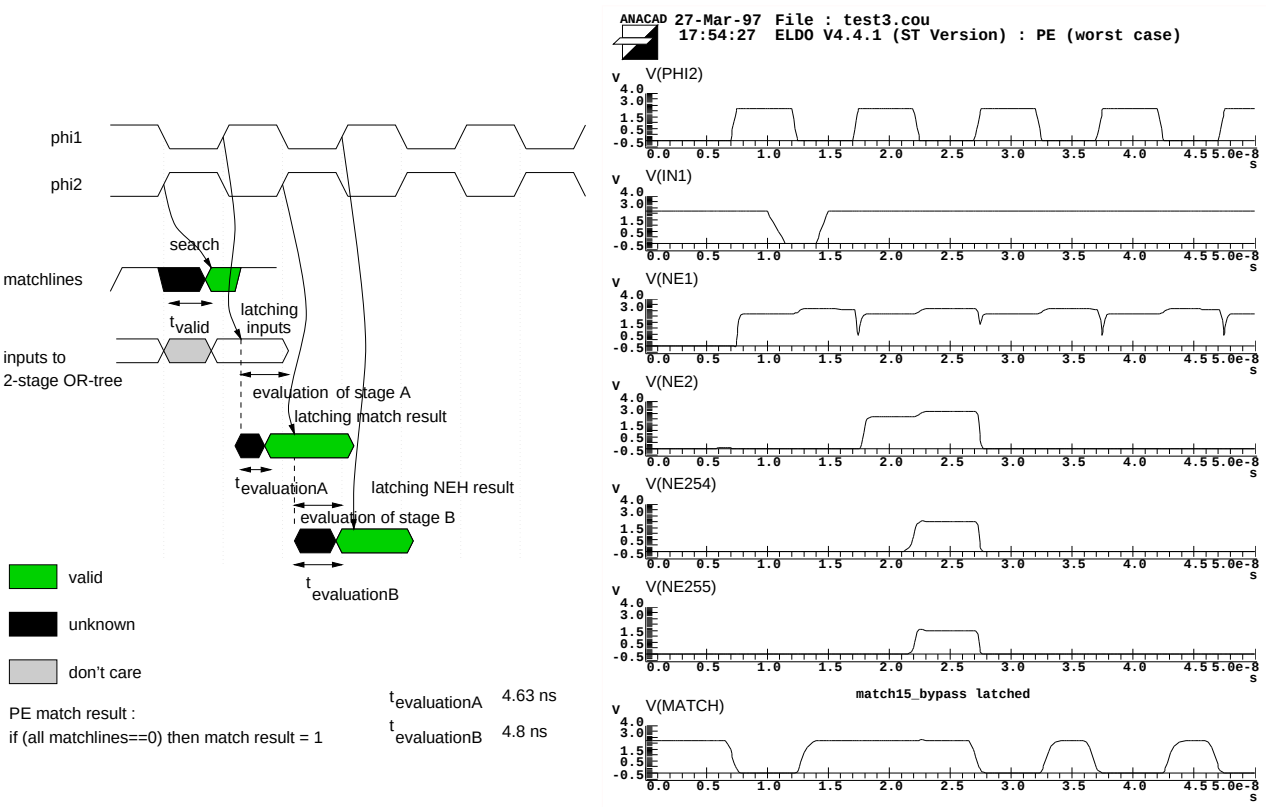
Ο αποκωδικοποιητής προτεραιότητας που φαίνεται στο σχήμα 3.16 σχεδιάστηκε σε χώρο διαστάσεων 60 μm x 1860 μm. Η κατακόρυφη διάσταση οφείλεται στο μέγεθος της



Σχήμα 3.16: *Η διάταξη και τα κυκλώματα του αποκωδικοποιητή προτεραιότητας*

μνήμης CAM η οποία τον τροφοδοτεί με τις γραμμές ταιριάσματος. Αποτελέσματα προσομοίωσης (σχήμα 3.17) έδειξαν ότι μπορεί να λειτουργήσει σε συχνότητα ρολογιού 100 MHz, κάτω από τις χειρότερες συνθήκες προσωμοίωσης.

Παρόλα αυτά, με στόχο να μην επιμηκυνθεί η δεύτερη pipeline της διαχείρισης ουρών, η οποία επεξεργάζεται τις αφίξεις πιστώσεων, εξετάστηκαν σε βάθος δύο εναλλακτικές λύσεις. Σύμφωνα με την πρώτη, η λειτουργία ψαξίματος στις μνήμες CAM και η πρώτη βαθμίδα του αποκωδικοποιητή προτεραιότητας (αποτελούμενη από το δέντρο δύο επιπέδων) έπρεπε να παράγουν ένα έγκυρο αποτέλεσμα μέσα σε μία φάση ρολογιού. Στην επόμενη φάση που φορτίζονται οι γραμμές ταιριάσματος, ταυτόχρονα γίνεται ο υπολογισμός της δεύτερης βαθμίδας του αποκωδικοποιητή προτεραιότητας, παράγοντας ένα διάνυσμα με ένα μόνο ‘‘1’’, αν στην προηγούμενη φάση υπήρχε κάποιο ταίριασμα. Η δεύτερη εναλλακτική λύση ήταν να αφιερωθεί εξ’ ολοκλήρου η μία φάση του ρολογιού στο ψάξιμο των



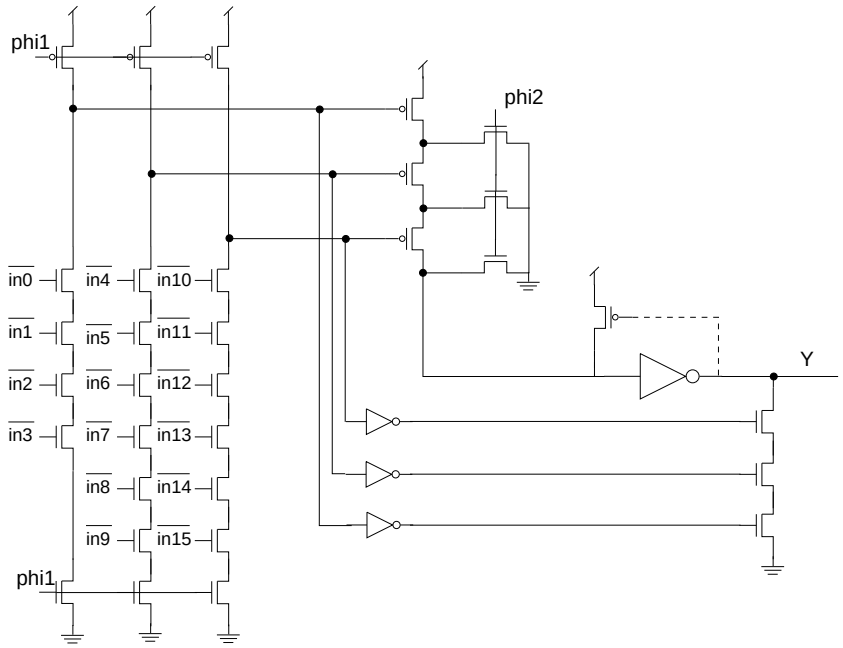
Σχήμα 3.17: *Κυματομορφές του αποκωδικοποιητή προτεραιότητας (σενάριο χειρότερης περίπτωσης : καμιά είσοδος δεν ήταν έγκυρο ταίριασμα)* $(NE_i == NEH_i)$

CAM's, και να συμπυκνωθούν και οι δύο βαθμίδες του αποκωδικοποιητή προτεραιότητας στην επόμενη φάση.

Το δυναμικό στυλ λογικής όμως, που χρησιμοποιήθηκε νωρίτερα κρίθηκε ακατάλληλο και για τις δύο εναλλακτικές λύσεις, διότι οι είσοδοι στα κυκλώματα τύπου domino δεν ήταν δυνατό να είναι σταθερές κατά τη φάση υπολογισμού. Έτσι μπορούσε να συμβεί εκφόρτιση του κόμβου εξόδου προτού οι είσοδοι να πάρουν την τελική έγκυρη τιμή τους. Γι' αυτό, παρά τα πλεονεκτήματα της δυναμικής λογικής (μικρές παρασιτικές χωρητικότητες, μεγάλη ταχύτητα, μικρή κατανάλωση ισχύος) έπρεπε να κινηθούμε προς συμπληρωματική λογική, ή λογική τύπου CVSL, χρησιμποιώντας cascode CVSL ή SSDL ([WE93]). Άλλη μία συνέπεια της πρώτης εναλλακτικής λύσης θα ήταν να χρησιμοποιηθούν αισθητήριοι ενισχυτές για τις γραμμές ταιριάσματος, όπως αυτοί που αναφέρονται στους [JC89]

και [THY+96], επιταχύνοντας έτσι την τροφοδοσία του αποκωδικοποιητή προτεραιότητας με έγκυρες τιμές. Αν και φαινόταν εφικτό, η προσπάθεια-χρόνος σχεδίασης και η κατανάλωση ισχύος των αισθητήριων ενισχυτών αντισταθμίστηκαν από τη δεύτερη λύση. Εξετάστηκαν επίσης κάποιες άλλες επιλογές, όπως άνιση διάρκεια φάσεων του κύκλου ρολογιού ή νέο σχήμα για τον αποκωδικοποιητή προτεραιότητας, αλλά απορρίφθηκαν.

Ένα υβριδικό σχήμα χρησιμοποιήθηκε για να υλοποιήσει μία πύλη OR μεγάλου fan-in. Μία πύλη OR 16-εισόδων απεικονίζεται στο σχήμα 3.18.



Σχήμα 3.18: *Μία πύλη OR 16-εισόδων (υπολογίζει το αποτέλεσμα στη φάση phi1)*

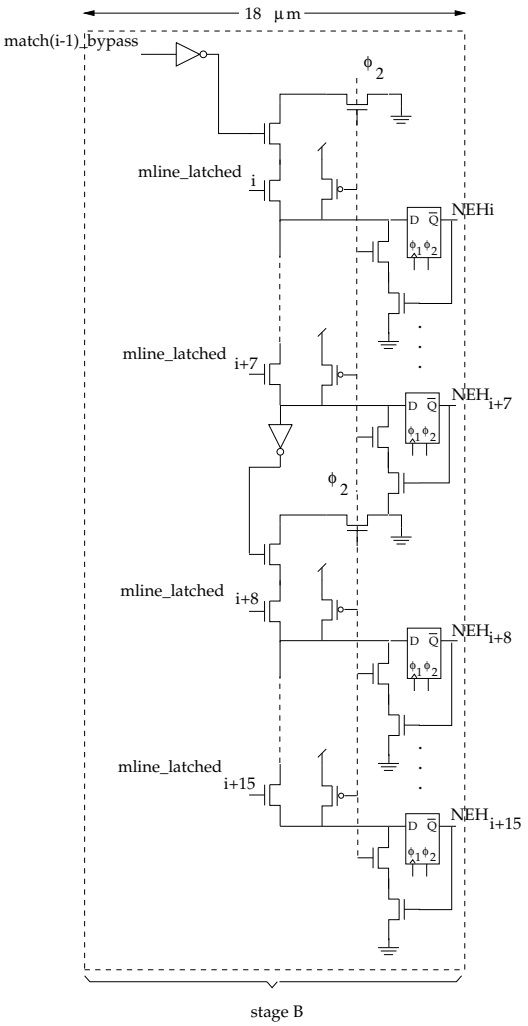
Η πύλη προφορτίζεται κατά τη φάση phi2 και υπολογίζει το αποτέλεσμα στη φάση phi1. Οι γραμμές ταιριάσματος αντεστραμμένες αποτελούν τις εισόδους της. Οι αλυσίδες pull-down από τα τρανζίστορ NMOS είναι κατανεμημένα κατά μήκος της κατακόρυφης διεύθυνσης όπου είναι οι αντίστοιχες είσοδοι. Η κοντότερη αλυσίδα βρίσκεται και πιο μακρυά από την έξοδο, για να αντισταθμίσει τη μεγαλύτερη παρασιτική χωρητικότητα λόγω συρμάτων διασύνδεσης. Η καθυστέρηση εξόδου της πιο πάνω πύλης OR που τροφοδοτεί τις 16 εισόδους του δεύτερου επιπέδου πυλών OR είναι 2.3 ns. Το πιο κάτω σήμα προσπεράσματος γεννιέται μετά από 4.3 ns από την ανερχόμενη ακμή του phi1. Αυτό συμβαίνει στην κρίσιμη περίπτωση όπου καμία είσοδος δεν είναι σε υψηλό δυναμικό (αν υπάρχει έστω και ένα ‘‘1’’ δεν συμβαίνει αλλαγή στην έξοδο), οπότε πρέπει να τραβηχτεί κάτω

η έξοδος Y όσο το δυνατόν πιο γρήγορα. Ένα μοναδικό τρανζίστορ PMOS θα μπορούσε να προστεθεί (ονομαζόμενο *αποκαταστάτη επιπέδου* στο [J.M96]) μεταξύ του V_{DD} και της εισόδου του οδηγητή αντιστροφέα, για να τραβήξει την έξοδο πιο γρήγορα στη γείωση. Όταν η έξοδος αρχίζει να χαμηλώνει ($V_{out} < V_{DD} - |V_{tp}|$) το τρανζίστορ αυτό ανοίγει και βοηθά την είσοδο να ανέβει γρηγορότερα. Το αρνητικό αποτέλεσμα είναι από τη μιά ότι αυξάνεται η χωρητικότητα του εσωτερικού κόμβου και από την άλλη ότι το τρανζίστορ μάχεται τη μείωση της τάσης στον κόμβο αυτό πριν σβήσει εντελώς, καθυστερώντας έτσι περισσότερο το χρόνο ανόδου της πύλης. Αυτό όμως δεν ενέχει καθόλου κόστος στην περίπτωσή μας, αφού η φάση προφόρτισης διαρκεί αρκετά.

Επίσης, σε περίπωση προβλημάτων χρονισμού εξετάστηκε μία επιπλέον βελτίωση. Ένα κύκλωμα με σύνδεση προς τα πίσω, όμοιο σε αρχή λειτουργίας με αυτό της προηγούμενης παραγράφου μπορεί να χρησιμοποιηθεί όπως φαίνεται στο σχήμα 3.19. Συγκεκριμένα χρησιμοποιείται ένα τρανζίστορ NMOS για να συντομεύσει το δρόμο προς τη γείωση μόλις το σήμα εξόδου αρχίζει να ανεβαίνει.

Στο σχήμα 3.20 φαίνεται η απόκριση του κυκλώματος, καταδείχνοντας ότι η έκδοση του αποκωδικοποιητή προτεραιότητας χωρίς pipeline είναι εφικτή. Πρέπει να σημειωθεί ότι οι ακίδες τάσης που φαίνονται στα σήματα NEH_i οφείλονται στη χωρητική σύζευξη μεταξύ του ρολογιού και των σημάτων εξόδου.

Αν και το πρόβλημα επίλυσης πολλαπλών ταιριασμάτων έχει ερευνηθεί αρκετά στο παρελθόν, η λειτουργία αυτή ολοκληρώνεται σε διακριτά, ανεξάρτητα διαστήματα χρόνου (είτε μετρημένα σε λειτουργίες ψαξίματος είτε σε καθυστερήσεις πυλών), χωρίς να λαμβάνονται υπόψη τα προηγούμενα αποτελέσματα. Από τη σκοπιά ενός προσεταιριστικού συστήματος η λειτουργία αυτή είναι μη προσεταιριστική, αφού είναι ανεξάρτητη από το περιεχόμενο των πολλαπλών ταιριασμάτων. Μέχρι στιγμής μία μοναδική γραμμή ταιριάσματος μπορεί να επιλεγεί βάση της φυσικής της θέσης (με μεθόδους ριπής κρατουμένου ή δομές μορφής δέντρου ή συνδυασμούς). Συνεπώς, είναι άξιο προσοχής το γεγονός ότι η επόμενη γραμμή στο χρόνο που θα επιλεγεί είναι ασυσχέτιστη με το ιστορικό της επίλυσης πολλαπλών ταιριασμάτων. Στην περίπτωση βεβαίως όπου υπάρχει μία μοναδική γραμμή που βρέθηκε να ταίριαζει και ανήκει σε μία ανεξάρτητη ροή δεν τίθεται θέμα. Όταν όμως δύο ή περισσότερα κύτταρα είναι χωρίς πιστώσεις και ένα ψάξιμο λόγω μιας εισερχόμενης πίστωσης παράγει πολλές γραμμές που ταιριάζουν, πρέπει να υπάρχει ένας μηχανισμός για δίκαιη μεταχείριση. Ένας απλός γραμμικός αποκωδικοποιητής προτεραιότητας δεν μπορεί να εκπληρώσει αυτήν την απαίτηση. Γι' αυτό ένα νέο σχήμα

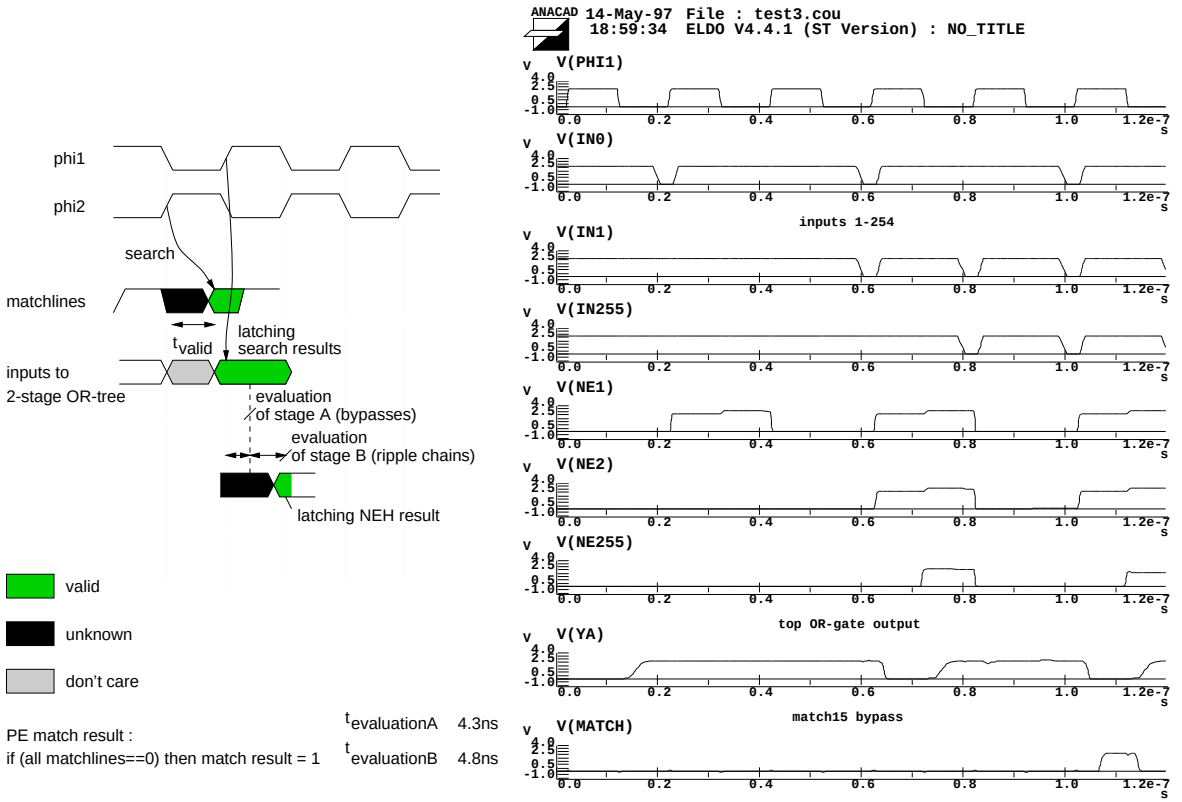


Σχήμα 3.19: *Η βελτιωμένη βαθμίδα Β του αποκωδικοποιητή προτεραιότητας*

εφευρέθηκε που παρέχει εγγυήσεις δικαιοσύνης και παρουσιάζεται στις επόμενες παραγράφους.

3.2 Εξυπηρέτηση πολλαπλών κλάσσεων ροών με δυναμικό τρόπο τύπου round-robin εξασφαλίζοντας δικαιοσύνη

Στις ακόλουθες παραγράφους θα παρουσιαστεί ένας νέος μηχανισμός χρονοπρογραμματισμού στο πλαίσιο ενός σχήματος ελέγχου ροής βάσει πιστώσεων, σε ένα δίκτυο από μεταγωγείς ATLAS I. Χρησιμοποιείται για να εφαρμόσει δίκαια διαιτησία μεταξύ ροών που είναι οργανωμένες σε ομάδες.



Σχήμα 3.20: *Κυματομορφές του αποκωδικοποιητή προτεραιότητας χωρίς pipeline (χειρότερη περίπτωση : καμιά είσοδος δεν ήταν έγκυρο ταίριασμα)*

3.2.1 Εισαγωγικά

Μία ομάδα συνδέσεων που μοιράζονται ένα κοινό μονοπάτι, είτε φυσικό είτε εικονικό, ονομάζεται *ομάδα ροών*. Ουσιαστικά είναι μία ροή πακέτων που ταξιδεύουν στη ίδια διαδρομή από την ίδια πηγή στον ίδιο προορισμό και απαιτούν τον ίδιο βαθμό εξυπηρέτησης σε κάθε σημείο μεταγωγής μέσα στη διαδρομή αυτή. Ο αντικειμενικός στόχος είναι η μεταχείριση των μελών μιας ομάδας ροών με ομοιογενή τρόπο. Θεωρούνται στενά συζευγμένες και ποτέ κάποια δεν προχωρεί ανεξάρτητα από τις άλλες. Αν κάποια υποστεί συμφόρηση θα υποφέρουν και οι υπόλοιπες.

Η έννοια των ανεξάρτητων κλάσεων αναφέρεται στη δημιουργία διακριτών σετ από ομάδες ροών. Συγκεκριμένα, ο όρος *συχωνευόμενες ροές* αναφέρεται στη συγχώνευση δύο ή περισσότερων εισερχόμενων συνδέσεων που ανήκουν σε διαφορετικές ομάδες ροών,

σε μία μοναδική σύνδεση στην έξοδο. Κατά αυτόν τον τρόπο όλα τα κύτταρα των συγχωνευόμενων ροών οδεύουν προς ένα μοναδικό συγκεκριμένο προορισμό. Άλλη μία περίπτωση συγχωνευόμενων ροών εμφανίζεται όταν κύτταρα που ανήκουν στην ίδια ομάδα ροών φτάνουν από διαφορετικές εισερχόμενες πόρτες και επιθυμούν να εξέλθουν από την ίδια πόρτα. Ένα πλεονέκτημα των συγχωνευόμενων ροών είναι η εξοικονόμηση του χώρου ενταμίευσης, αφού οι συγχωνευόμενες ροές καταναλώνουν μονάχα μία θέση στον ενταμιευτή του μεταγωγέα.

Το θέμα της δικαιοσύνης έχει να κάνει με το ρυθμό προόδου που έχει καθεμιά από τις συγχωνευόμενες ροές. Ένας ρυθμός εκτός ισορροπίας σημαίνει ότι τουλάχιστον μία από τις συγχωνευόμενες ροές απολαμβάνει υψηλότερο ρυθμό εξυπηρέτησης από ότι οι άλλες. Οι ατομικές ομάδες ροών που υπόκεινται σε έλεγχο ροής προς τα πίσω δεν αντιμετωπίζουν προβλήματα δικαιοσύνης, δεδομένης της πολιτικής εξυπηρέτησης στον ATLAS I. Σε αυτό συνηγορεί η καλή ιδιότητα των ουρών εξόδων του που δέχονται τα έτοιμα κύτταρα και είναι FIFO, καθώς και ο περιορισμός ύπαρξης το πολύ ενός κυττάρου ανά ομάδα ροών. Εντούτοις, στην περίπτωση των συγχωνευόμενων ροών αυτά τα χαρακτηριστικά του ATLAS I δεν μπορούν να εγγυηθούν δικαιοσύνη. Συγκεκριμένα, όταν δύο ή περισσότερα κύτταρα που ανήκουν σε συγχωνευόμενες ροές περιμένουν στην CLL για μία πίστωση, ένας απλός γραμμικός αποκωδικοποιητής προτεραιότητας μπορεί να επιλέξει ένα από αυτά, αλλά όχι και να εγγυηθεί δικαιοσύνη όσο αφορά το ρυθμό εξυπηρέτησης καθεμιάς ροής.

Εξετάστηκαν μερικές εναλλακτικές λύσεις, ντετερμινιστικές και μη, οι οποίες παρουσιάζονται στη συνέχεια. Στην τελευταία παράγραφο επεξηγείται λεπτομερώς ένα νέο σχήμα, το οποίο εξασφαλίζει ότι κάθε μέλος μίας συγχωνευόμενης ροής που περιμένει στον ενταμιευτή του μεταγωγέα θα έχει την ευκαιρία να εξυπηρετηθεί μία φορά προτού εξυπηρετηθεί ξανά.

3.2.2 Ένα σχήμα ψευδοτυχαίας επιλογής

Μία αποδεκτή λύση θα ήταν να κάνουμε ψευδοτυχαία επιλογή ανάμεσα στις ομάδες ροών που περιμένουν να λάβουν μία πίστωση. Ο στόχος λοιπόν στρέφεται στην επιλογή μιας γεννήτριας τυχαίων αριθμών με καλές ιδιότητες, εύκολη στην υλοποίηση και αρκετά γρήγορη για τις προδιαγραφές χρονισμού που έχουμε.

Εξετάζοντας διάφορες μεθόδους για γέννηση τυχαίων πραγματικών αριθμών U_n, ομοιόμορφα κατανεμημένων μεταξύ 0 και 1*, οι πιο δημοφιλείς γεννήτριες τυχαίων αριθμών σήμερα ακολουθούν τη μέθοδο (πολλαπλασιαστικής) γραμμικής σύγκλισης, ή κάποια παραλαγή της. Είναι μια μέθοδος από τις πιο απλές, παλιές (D.H. Lehmer,1948) αλλά και πιο ευρέως χρησιμοποιούμενη.

Για να έχουμε λοιπόν μία ακολουθία τυχαίων αριθμών (X_n) εφαρμόζουμε την αναδρομική σχέση :

$X_{n+1} = (aX_n + c)\ mod\ m,$	$n \geq 0$
m, the modulus,	m > 0,
a, the multiplier,	$0 \leq a < m,$
c, the increment,	$0 \leq c < m,$
X_0, the starting value,	$0 \leq X_0 < m.$

Αυτή ορίζεται ως μία *γραμμική, συγκλίνων ακολουθία.* Όλες οι ακολουθίες που έχουν το γενικό τύπο $X_{n+1} = f\ (X_n)$ έχουν ένα επαναλαμβανόμενο κύκλο που ονομάζεται περίοδος. Στην περίπτωση των γραμμικών συγκλίνων γεννητριών (linear congruential generators (LCG)) η περίοδος της ακολουθίας είναι m, αφού η περίοδος δεν μπορεί να έχει παραπάνω από m στοιχεία. Η κατάλληλη επιλογή των παραμέτρων m,a,c, και $_0$ εξαρτάται από την εφαρμογή και καθορίζει την περίοδο αλλά και τις ιδιότητες τυχαιότητας των LCG's.

Από μαθηματική άποψη, μία ακολουθία τυχαίων αριθμών από γεννήτρια (βεβαίως αποκαλούνται ‘‘τυχαίοι’’, αν και είναι εντελώς ντετερμινιστικοί στο χαρακτήρα) πρέπει να έχει τις ακόλουθες ιδιότητες : μεγάλη περίοδο, ασυσχέτιστη ακολουθία, ομοιομορφία, μέσο όρο=1/2 και μεταβλητότητα=1/12. Αυτές οι ιδιότητες μπορεί να είναι επιθυμητές ή αναγκαίες αλλά όχι και ικανές για τυχαιότητα. Πολλές έννοιες έχουν εισαχθεί τα τελευταία χρόνια για την αποδοχή ή απόρριψη γεννητριών τυχαίων αριθμών. Επιπλέον, αρκετά τεστ, θεωρητικά και εμπειρικά έχουν εφαρμοστεί σε ακολουθίες για να ερευνηθεί η τυχαιότητα τους. Στο πλαίσιο αυτής της εργασίας δεν θα γίνει ανάλυση των οικογενειών από τεστ (λεπτομερής έρευνα βρίσκεται στον D.Knuth). Οι γραμμικές συγκλίνοντες ακολουθίες πρέπει να περνούν το ονομαζόμενο *φασματικό* τεστ προτού γίνουν αποδεκτές ως τυχαίες. Ο στόχος του φασματικού τεστ είναι να ανακαλυφθεί η συμπεριφορά της γεννήτριας όταν οι έξοδοί της χρησιμοποιούνται σχηματίζοντας d-tuples.

*Στην πραγματικότητα γεννάμε ακεραίους X_n μεταξύ του 0 και κάπου αριθμού m, οπότε ο λόγος $U_n = X_n/m$ θα βρίσκεται μεταξύ 0 και 1

Αν και οι LCG's είναι από τις πιο ''ωραίες'', ''απλές'' και πιο αποδοτικές μεθόδους, το βασικό τους μειονέκτημα είναι ότι οι d-tuples τέτοιων ακολουθιών παρουσιάζουν μία κανονική επαναληπτική δομή όταν σχεδιαστεί σε d διαστάσεις. Αυτό το πρόβλημα είναι εγγενής στις περιοδικές ακολουθίες. Μία κυκλική ακολουθία m αριθμών, όταν την πάρουμε σε ζεύγη, εντοπίζεται πάνω στα m από τα m^2 σημεία σε ένα δισδυάστατο πλέγμα m επί m. Εν γένη, τα $m^n - m$ σημεία του πλέγματος θα είναι πάντα κενά στον n-διάστατο χώρο από τα n-tuples που λαμβάνονται από μία ακολουθία περιόδου m, δηλαδή λείπουν τα περισσότερα γινόμενα τομής. Επίσης, αυτές οι γεννήτριες έχουν αποδειχθεί ότι παρουσιάζουν συσχετίσεις μεταξύ αριθμών που βρίσκονται σε απόσταση 2^n μεταξύ τους.

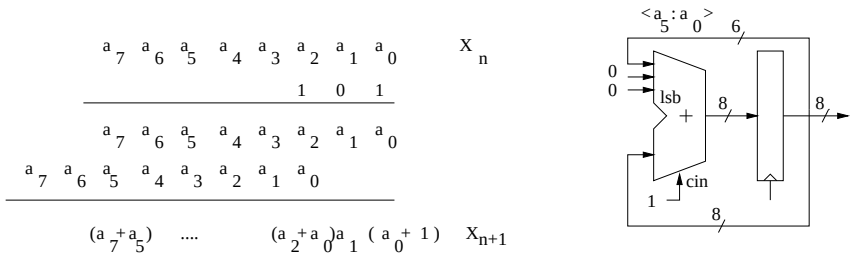
Στην περίπτωσή μας πρέπει να γεννάμε ακεραίους X_n μεταξύ 0 και 255. Βεβαίως μία γεννήτρια με περίοδο μήκους 256 σχεδόν δεν μπορεί να είναι τυχαία [D.E81]. Εντούτοις, ας προσπαθήσουμε να κατασκευάσουμε την καλύτερη δυνατή πολλαπλασιαστική γραμμική συγκλίνουσα γεννήτρια.

Προφανώς, m=256 (2^8). Ακολουθώντας τις οδηγίες του Knuth, αν το m είναι δύναμη του 2, το a πρέπει να επιλεγεί έτσι ώστε a mod 8 = 5. Πρέπει να σημειωθεί ότι η επιλογή του a μαζί με την επιλογή του c μπορεί να εξασφαλίσει ότι η γεννήτρια τυχαίων αριθμών θα παράγει όλες τις m διαφορετικές δυνατές τιμές προτού αρχίσει να επαναλαμβάνεται. Υποθέτωντας η βάση αριθμών της μηχανής που χρησιμοποιούμε είναι z=2, και $m = z^e$, τότε $a = z^k + 1,\ 2 \leq k < e$. Η τιμή του c δεν έχει σημασία όταν το a που χρησιμοποιούμε είναι καλός πολλαπλασιαστής. Έτσι μπορούμε να διαλέξουμε c=1. Η αναδρομική σχέση τώρα παίρνει τη μορφή

$$X_{n+1} = ((z^k + 1)X_n + 1)\ mod\ z^e$$

και αυτή η εξίσωση δείχνει ότι μπορούμε να αποφύγουμε τον πολλαπλασιασμό, αρκούν ολισθήσεις και προσθέσεις. Επίσης, απαιτείται μόνο μία θέση μνήμης για να κρατάμε τον τελευταίο ακέραιο της ακολουθίας.

Μία κατάλληλη γεννήτρια LCG φαίνεται να είναι η $X_{n+1} = (5X_n + 1)\ mod\ 256$. Παρά την απλότητα της $f(X_n)$, η υλοποίηση θα απαιτούσε έναν πολλαπλασιαστή και έναν αθροιστή σε πρώτη ματιά. Εντούτοις, με ένα πιο προσεκτικό μάτι φαίνεται από τη σχέση ότι αφού ο πολλαπλασιαστής είναι πάντα σταθερός, συγκεκριμένα το πέντε, ο επόμενος ακέραιος X_{n+1} μπορεί να υπολογιστεί όπως φαίνεται στο σχήμα 3.21 (υποθέτοντας ότι $X_n = a_7a_6a_5a_4a_3a_2a_1a_0$). Συνεπώς, χρειάζεται ένας απλός αθροιστής των 8 bit και μία θέση μνήμης για να κρατά την προηγούμενη τιμή X_n.



Σχήμα 3.21: *Η υλοποίηση της γεννήτριας* $X_{n+1} = (5X_n + 1)\ mod\ 256$

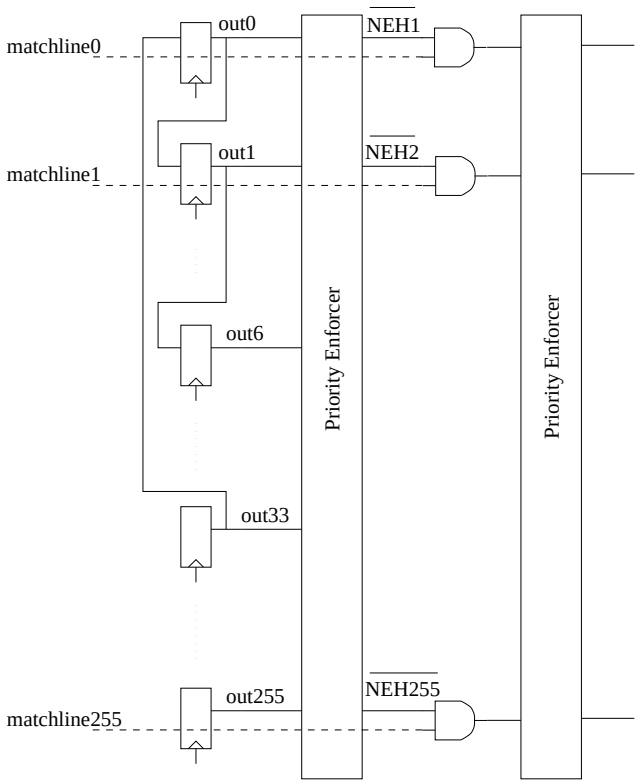
Παρόλα τούτα, για να ενσωματώσουμε αυτό το σχήμα στον αποκωδικοποιητή προτεραιότητας (PE1) που ήδη χρειαζόμαστε αφού έχει γίνει το ψαξιμο στις μνήμες CAM, πρέπει να χρησιμοποιηθεί ένας αποκωδικοποιητής 8-σε-256 και άλλος ένας αποκωδικοποιητής προτεραιότητας (PE2) για να λειτουργεί στο διάνυσμα που παράγεται από τον αποκωδικοποιητή, ώστε ο αρχικός PE1 να λειτουργεί στο αποτέλεσμα $\overline{NEH_{PE2}}ANDmatchlines$.

Εμβαθύνοντας στο παραπάνω σχήμα, αφού η ακολουθία που γεννιέται από τον τύπο είναι καθορισμένη, δηλαδή η σειρά {0,1,6,31,156,13,...} πάντα θα εμφανίζεται με την ίδια σειρά, μπορούμε να εγκαταλείψουμε το γενικό σχήμα με τον αθροιστή και τον αποκωδικοποιητή. Αυτό που χρειαζόμαστε είναι μια σταθερή αλυσίδα από 256 D-flip-flops συνδεδεμένα σύμφωνα με την προκαθορισμένη ακολουθία (σχήμα 3.22). Αυτή η σταθερή αλυσίδα ουσιαστικά γεννά ένα ‘‘ψευδοτυχαίο’’ αρχικό σημείο λειτουργίας για τον αποκωδικοποιητή προτεραιότητας.

Οι γεννήτριες LCG’s αν και είναι αρκετά απλές στην υλοποίησή τους, έχουν κάποια μειονεκτήματα όπως ειπώθηκε. Γενικά, οι d-tuples παρουσιάζουν μια κανονική επαναληπτική δομή όταν σχεδιαστούν στις d διαστάσεις. Αυτό είναι εμφανές και για την παραπάνω γεννήτρια, όπως φαίνεται στο σχήμα 3.23, μια γραφική παράσταση με μπάρες στον δισδυάστατο χώρο.

Οι γεννήτριες Φιμπονάτσι (Lagged Fibonacci Generators (LFGs)) επιχειρούν να βελτιώσουν τις LCG’s χρησιμοποιώντας περισσότερες από μία παλιά τιμή της ακολουθίας στην ακολουθιακή σχέση, ώστε να μειώσουν τη συσχέτιση και να αυξήσουν την περίοδο. Αυτές οι γεννήτριες ονομάστηκαν έτσι λόγω της ομοιότητας τους με τη γνωστή ακολουθία Φιμπονάτσι ($X_n = X_{n-1} + X_{n-2}$). Βεβαίως, μελέτες έχουν δείξει ότι αυτή δεν είναι και τόσο καλό παράδειγμα γεννήτριας τυχαίων ακολουθιών. Μία καλύτερη μέθοδος είναι οι ακολουθίες Φιμπονάτσι με βραδυπορεία (*lagged* Fibonacci sequences), όπου ο κάθε αριθμός είναι συνδυασμός δύο οποιαδήποτε προηγούμενων τιμών :

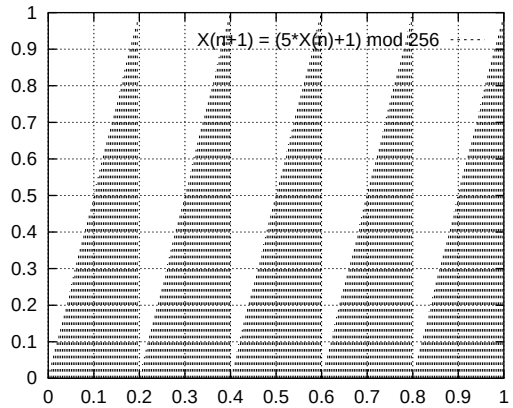
$$X_n = (X_{n-l} \odot X_{n-k}) mod m, where l > k > 0$$



Σχήμα 3.22: *Υλοποίηση της γεννήτριας* $X_{n+1} = (5X_n + 1)\ mod\ 256$

όπου τα k και l ονομάζονται *lags*, και το $\odot$ είναι οποιαδήποτε αριθμητική πράξη, όπως $+, -, *, \oplus$ (bitwise exclusive OR). Εδώ αντίθετα με πριν που είχαμε μία αρχική στις γεννήτριες LCG, χρειαζόμαστε l αρχικές τιμές, $X_0, \ldots, X_{l-1}$, για να υπολογίσουμε το επόμενο στοιχείο της ακολουθίας. Υποθέτωντας ότι χρησιμοποιούμε τον τελεστή πρόσθεσης έχουμε από τους πιο απλές και γρήγορές γεννήτριες RNGs. Επίσης, έχει ελεγχθεί εκτενώς για ιδιότητες τυχαιότητας από τον Marsaglia [G.M85] και έχει λάβει πολύ καλές κριτικές.

Αν το m είναι δύναμη του δύο, δηλαδή $m = 2^M$, και επιλεγούν σωστά οι πρώτες *l* τιμές του X, τότε η περίοδος P αυτής της γεννήτριας είναι ίση με $(2^l - 1) * 2^{M-1}$. Προφανώς η τιμή του m δεν περιορίζει την περίοδο της γεννήτριας, όπως στην περίπτωση μιας LCG. Η περίοδος μπορεί να γίνει οσοδήποτε μεγάλη αν αυξηθεί το μεγαλύτερο lag. Εμπειρικά τεστ έχουν δείξει ότι οι ιδιότητες τυχαιότητας αυτών των γεννητριών βελτιώνονται παράλληλα με την αύξηση του lag. Αυτό φυσικά σημαίνει ότι ταυτόχρονα αυξάνεται και ο αριθμός θέσεων μνήμης που χρειαζόμαστε, σε αντίθεση με την περίπτωση των LCG, όπου μία μόνο θέση μνήμης ήταν αρκετή. Είναι επίσης φανερό ότι η γεννήτρια δεν παράγει P

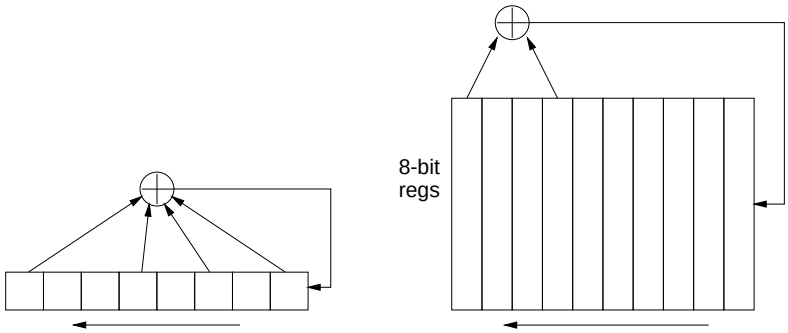


Σχήμα 3.23: *Μία περίοδος (256) (ζεύγη συνεχόμενων σημείων (X_n, X_{n+1})), στο δισδυάστατο χώρο*

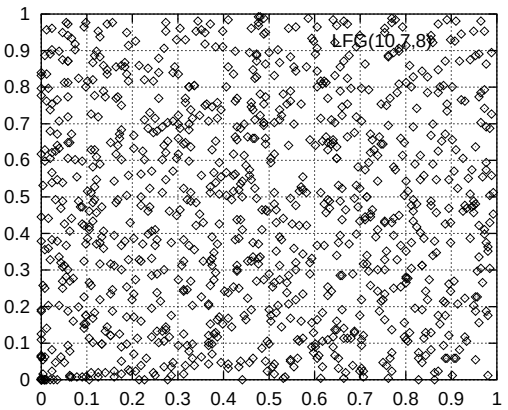
διαφορετικές τιμές, προτού αρχίσει να τις ανακυκλώνει.

Θεωρητικά μία γεννήτρια Φιμπονάτσι λειτουργεί το ίδιο όπως ένας γραμμικός καταχωρητής ολίσθησης. Όταν η αριθμητική πράξη που χρησιμοποιείται είναι το XOR, τότε οι LFGs αναφέρονται πιο συχνά ως γενικοποιημένες γεννήτριες τύπου καταχωρητή ολίσθησης [TW73]. Μία γεννήτρια γραμμικού καταχωρητή ολίσθησης (SRG) παράγει ακολουθίες που εξαρτώνται από το μήκος του καταχωρητή, τις συνδέσεις ανατροφοδότησης προς τα πίσω, και τις αρχικές συνθήκες. Αν η SRG έχει L βαθμίδες, και η ακολουθία εξόδου έχει μήκος $2^L - 1$, τότε ονομάζεται *ακολουθία μέγιστου δυνατού μήκους*. Κάθε τιμή στην ακολουθία εμφανίζεται μόνο μία φορά σε μια περίοδο. Δεδομένου του μήκους του καταχωρητή ολίσθησης, οι συνδέσεις ανατροφοδότησης προς τα πίσω καθορίζουν αν η ακολουθία θα είναι μέγιστου δυνατού μήκους ή όχι. Ο [W.W61] παρουσιάζει πλήρεις πίνακες από τους οποίους μπορούμε να φτιάξουμε κατευθείαν μία γεννήτρια ψευδοτυχαίων αριθμών.

Μία γεννήτρια SRG κατάλληλη για τις ανάγκες για τυχαία επιλογή μεταξύ 256 κυττάρων φαίνεται να είναι αυτή που απεικονίζεται στο σχήμα 3.24. Στο ίδιο σχήμα φαίνεται επίσης και μια ισορροπημένη επιλογή μεταξύ της οικογένειας των γεννητριών Φιμπονάτσι. Χρησιμοποιούνται δέκα καταχωρητές των 8 bit για να υλοποιήσουν μία LFG(10,7,8) με περίοδο P=130944. Το πρόβλημα αρχικοποίησής τους έχει προσεγγιστεί από τον [MSDM94]. Επίσης, από το σχήμα 3.25 είναι φανερό ότι το εξόφθαλμο μειονέκτημα των LCGs έχει ξεπεραστεί από το LFG(10,7,8).



Σχήμα 3.24: *Μία γεννήτρια SRG μήκους 8 με συνδέσεις [1,3,5,8] και μία LFG(10,7,8)*



Σχήμα 3.25: *Γραφική παράσταση 1000 μόνο σημείων (περίοδος=130944) μίας γεννήτριας LFG(10,7,8)*

3.2.3 Εναλλακτικές ντετερμινιστικές υλοποιήσεις

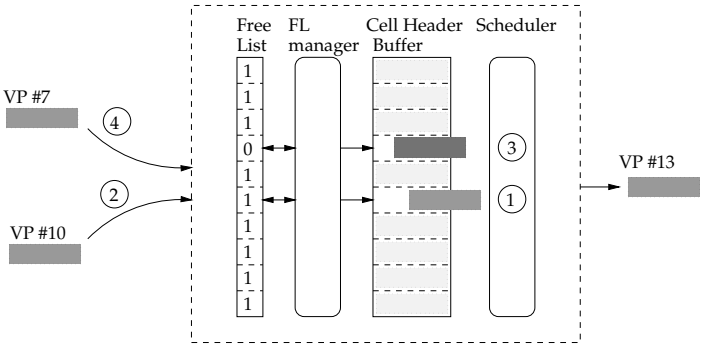
Μία πρώτη προσέγγιση του προβλήματος δείχνει ότι ένα σχήμα διαχείρισης με ντερμινιστικό τρόπο θα απαιτούσε αρκετό χώρο αποθήκευσης για τη διατήρηση δεικτών ανά κλάση συγχωνευόμενων ροών, συν αρκετή λογική ελέγχου για να χειριστεί τις απαιτούμενες λειτουργίες. Ουσιαστικά για κάθε κλάση συγχωνευόμενων ροών χρειάζεται μία διασυνδεδεμένη λίστα με τη σειρά εξυπηρέτησης των κυττάρων στη λίστα των κυττάρων χωρίς πιστώσεις, που ανήκουν σε αυτήν την κλάση ώστε να αποφευχθεί η άδικη συμπεριφορά κάποιων ροών. Αν όμως έχουμε παραπάνω από μία κλάση συγχωνευόμενων ροών το κόστος γίνεται μη αποδεκτό, τόσο σε πολυπλοκότητα, όσο και σε χώρο και τήρηση προδιαγραφών χρονισμού.

Άλλες εναλλακτικές λύσεις για μία ικανοποιητικά τυχαία και δίκαια κατανομή των πιστώσεων στην περίπτωση των κλάσεων συγχωνευόμενων ροών εξετάστηκαν με βάση

κυκλικούς αποκωδικοποιητές προτεραιότητας (διάφοροι τρόποι υλοποίησης τους παρουσιάζονται στο [Koz96]). Τέτοιοι μηχανισμοί όμως αποδείχτηκαν ανεπαρκείς για να αντιμετωπίσουν συσχετίσεις και παράνομη συμπεριφορά ροών.

3.2.4 Ο δίκαιος δυναμικός χρονοπρογραμματιστής

Ένας ιδανικός μηχανισμός υποδεικνύει ότι ένα νεοαφικνούμενο κύτταρο, που ανήκει στην ίδια κλάση με άλλα κύτταρα που είναι ενταμιευμένα λόγω συμφορήσεως στις συνδέσεις, θα λάβει άδεια να αναχωρήσει αφού όλα αυτά τα *παλαιότερα* κύτταρα αναχωρήσουν. Με άλλα λόγια : η σειρά αφίξεως των κυττάρων που ανήκουν στην ίδια κλάση θα είναι και η σειρά των αναχωρήσεων. Ο τρόπος λειτουργίας όμως της λίστας ελεύθερων θέσεων επιβάλλει στατική προτεραιότητα (επιλέγει πάντα το πρώτο ελεύθερο από την κορυφή) και αυτό μπορεί να καταλήξει σε σενάρια όπως αυτό του σχήματος 3.26, σε περιπτώσεις κίνησης με πολύ φορτίο. Οι αριθμοί στους κύκλους υποδηλώνουν την ακολουθία των γεγονότων που μπορεί να συμβούν, υποθέτοντας ότι ο κοινόχρηστος ενταμιευτής κυττάρων είναι πλήρως κατειλημένος, ή τουλάχιστον οι πιο πάνω θέσεις του. Συνεπώς ο επιθυμητός (ιδανικός) κανόνας που διατυπώθηκε στην αρχή της παραγράφου είναι δυνατόν να παραβιάζεται λόγω του τρόπου λειτουργίας απόδοσης πιστώσεων σε κύτταρα που τις περιμένουν και εύρεσης ελεύθερων θέσεων για νεοεισερχόμενα κύτταρα.



Σχήμα 3.26: *Παράδειγμα αλλαγής σειράς εξόδου κυττάρων από την εισερχόμενη σειρά τους*

Εντούτοις, αυτό που μπορεί να εγγυηθεί είναι ότι δεν μπορεί κάποιο νεοαφικνούμενο κύτταρο να προσπεράσει όλα τα κύτταρα που είναι εγγεγραμένα σε μία εικονική λίστα

από κύτταρα που ανήκουν σε συγχωνευόμενη ομάδα ροών. Συμπερασματικά, ο αντικειμενικός στόχος που επιδιώκουμε μπορεί να διατυπωθεί ως εξής :
κάθε μέλος της ομάδας θα λάβει μία ευκαιρία με ντετερμινιστικό τρόπο, προτού ξαναλάβει το δικαίωμα για δεύτερη ευκαιρία.

Ο χαρακτηρισμός του *δυναμικού* χρονοπρογραμματιστή τύπου round-robin βασίζεται στον τρόπο που δημιουργείται μία ομάδα (εικονική λίστα) από κύτταρα που ανήκουν σε συγχωνευόμενη ομάδα ροών. Μία εικονική λίστα δημιουργείται όποτε ισχύουν οι ακόλουθες συνθήκες :

1. Ένας προηγούμενος εικονικός κύκλος τύπου round-robin έχει ολοκληρωθεί αφού εξυπηρετήθηκαν όλα τα μέλη της εικονικής λίστας.

2. Υπάρχουν νέα κύτταρα της ίδιας κλάσης συγχωνευόμενης ομάδας ροών, τα οποία έφτασαν κατά το χρονικό διάστημα εξυπηρέτησης του προηγούμενου εικονικού κύκλου ή μετέπειτα.

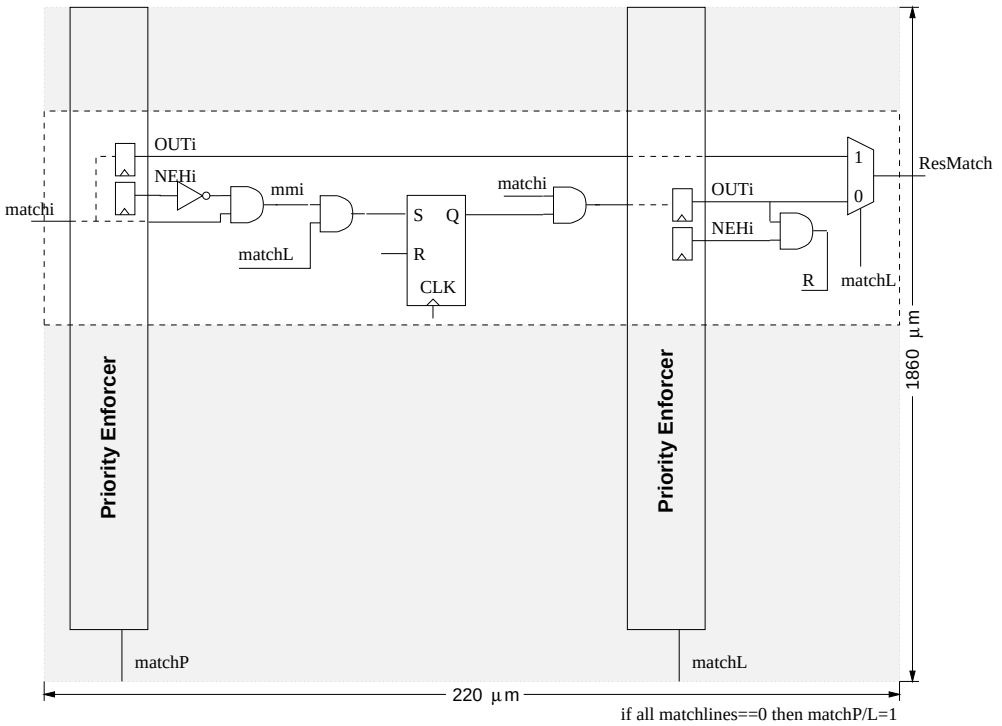
3. Μία άφιξη νέας πίστωσης σηματοδοτεί την έναρξη ενός νέου εικονικού κύκλου που περιέχει ένα μεταβλητό αριθμό συμμετεχόντων κυττάρων, που περίμεναν στη CLL για αυτήν την πίστωση.

Ένα κύτταρο που ανήκει στην ίδια συγχωνευόμενη ομάδα ροών και φτάνει στον μεταγωγέα αφού έχει δημιουργηθεί η εικονική αυτή λίστα δεν μπορεί να εγγραφεί σε αυτή. Θα υποχρεωθεί να περιμένει έως ότου η συνθήκη 1 γίνει ξανά αληθής. Βεβαίως το μειονέκτημα που περιγράφηκε νωρίτερα και αναπαριστάται στο σχήμα 3.26 (δηλαδή αδικία σε μικρές περιόδους χρόνου) δεν μπορεί να αποφευχθεί με αυτόν τον τρόπο. Μακροχρόνια όμως εξασφαλίζεται δίκαιη εξυπηρέτηση.

3.2.5 Η υλοποίηση

Ένας χρονοπρογραμματιστής που χειρίζεται πολλαπλές κλάσεις ομάδων ροών, για να είναι χρήσιμος στην πράξη, πρέπει να είναι απλός στη λειτουργία και συνεπώς αποδοτικός και ταχύς στην υλοποίηση. Συγκεκριμένα στον ATLAS I ο διαθέσιμος χρόνος για ολοκλήρωση μιας απόφασης χρονοπρογραμματισμού είναι πολύ μικρός και ο μηχανισμός πρέπει να είναι υλοποιημένος σε hardware.

Κάτω από αυτό το πρίζμα το σχήμα 3.27 απεικονίζει τη διάταξη του δίκαιου, δυναμικού χρονοπρογραμματιστή που χρησιμοποιήθηκε στο υποσύστημα διαχείρισης πολλαπλών ουρών. Ένα βασικό χαρακτηριστικό που εκμεταλλεύεται είναι η εγγενής ιδιότητα διαχωρισμού της πληροφορίας που συμβαίνει κατά το ψάξιμο της CLL. Το αποτέλεσμα του ψαξίματος υποδηλώνει σε ποια συγχωνευόμενη ομάδα ροών πρέπει να εφαρμοστεί ένας μηχανισμός για επιβολή δίκαιης εξυπηρέτησης.



Σχήμα 3.27: *Η υλοποίηση του δυναμικού δίκαιου χρονοπρογραμματιστή*

Όπως φαίνεται στο σχήμα χρησιμοποιούνται δύο αποκωδικοποιητές προτεραιότητας, όμοιοι με αυτόν που περιγράφτηκε στην παράγραφο 3.2.3. Λειτουργούν παράλληλα στον ίδιο κύκλο ρολογιού και σε κάθε κύκλο, αλλά ο καθένας με διαφορετικά διανύσματα εισόδου. Ο πρώτος δέχεται ένα διάνυσμα εισόδου που αποτελείται κατευθείαν από τα αποτελέσματα του ψαξίματος στην CLL. Ο δεύτερος εξετάζει μία πιθανή συγχωνευόμενη ομάδα ροών που ξεχωρίζει αν γίνουν ‘‘and’’ οι γραμμές ταιριάσματος με τα περιεχόμενα της μάσκας που αποθηκεύει όλες τις εικονικές λίστες. Αν όντως υπήρχε μία συγχωνευόμενη

ομάδα ροών (υποδεικνύεται από το σήμα *matchL*) και πρέπει να εξυπηρετηθεί κάποιο μέλος της από την εικονική λίστα, τα αποτελέσματα του πρώτου αποκωδικοποιητή προτεραιότητας αγνοούνται. Έτσι επιλέγεται το πρώτο από πάνω μέλος της λίστας από τον ''ενεργό'' PE ενώ ταυτόχρονα καθαρίζεται η καταχώρησή του από τη μάσκα. Με αυτόν τον τρόπο ένα επόμενο ταίριασμα της ίδιας κλάσης θα παράγει το επόμενο μέλος στη σειρά. Αν όμως δεν βρέθηκε καμιά εικονική λίστα (το *matchL* είναι ένα), το οποίο δείχνει είτε ότι μία ατομική, ανεξάρτητη ομάδα ροής ταίριασε, είτε ότι συνέβει ένα πολλαπλό ταίριασμα πρώτη φορά για αυτή τη συγχωνευόμενη ομάδα ροών, είτε δεν υπήρχε καν ταίριασμα, τότε λαμβάνονται υπόψη τα αποτελέσματα του πρώτου αποκωδικοποιητή προτεραιότητας. Εξυπηρετείται η πρώτη από πάνω ομάδα ροής, ενώ παράλληλα εάν βρέθηκε μία νέα κλάση ($\overline{NEH_i} \cdot IN_i$), καταργάφεται στη μάσκα. Στο σχήμα φαίνεται ασκίαστο το επιπλέον κύκλωμα ανά bit που απαιτείται για αυτούς τους ελέγχους.

Όπως απεικονίζεται στο σχήμα η μάσκα υλοποιείται από μια αλυσίδα 255 SR-flip-flops. Αρχικά βεβαίως πρέπει να μηδενιστούν. Επίσης, οι είσοδοι S-R σε κάθε flip-flop δεν μπορεί ποτέ να είναι 1 ταυτόχρονα τη στιγμή που ανέρχεται το σήμα του ρολογιού. Επιπλέον, αν και οι λογικές πύλες που φαίνονται στο σχήμα υπήρχαν στη βιβλιοθήκη του πακέτου σχεδίασης, υλοποιήθηκαν ξανά σε full-custom, λόγω περιορισμών στο κατακόρυφο ύψος και ανομοιομορφίας λόγω ανάμιξης κομματιών full-custom με κομμάτια semi-custom διαφορετικών προδιαγραφών.

3.3 Αποτελέσματα: Χρονικές επιδόσεις, εμβαδόν και κατανάλωση ισχύος

Τα κυκλώματα full-custom προσομειώθηκαν εκτενώς, χρησιμοποιώντας πραγματικές παρασιτικές χωρητικότητες που προέκυψαν από το layout. Ο πίνακας που ακολουθεί δίνει αποτελέσματα που προέκυψαν από προσομειώσεις υπό συνθήκες χειρότερης κατάστασης (μοντέλα τρανζίστορ nmos και pmos ελάχιστου ρεύματος, θερμοκρασία 105^{o}C, τάση τροφοδοσίας 2.7 V) (η καθυστέρηση του οδηγητή εξωτερικού φορτίου δεν περιλαμβάνεται). Το ψάξιμο είναι η πιο αργή λειτουργία δεδομένου ότι οι γραμμές ταιριάσματος διασχίζουν και τις δύο μνήμες (flowGroup, outMask) και δεν χρησιμοποιούνται αισθητήριοι ενισχυτές. Οι χρόνοι προετοιμασίας διευθύνσεων και δεδομένων μετρήθηκαν ως προς το ρολόι που μπαίνει κομμάτι του κυκλώματος full-custom.

Timing Table

parameter	min	typ	max	unit
FlowGroup match access†			6.4	ns
OutMask match access†			6.3	ns
CreditMask modify access†			4.9	ns
CreditMask cycle time	10			ns
Priority Enforcer (eval. phase stage A)			4.2	ns
Priority Enforcer (eval. phase stage B)			4.7	ns
HTRF access†			3.1	ns
Address setup time	0.5			ns
Data setup time	0.5			ns

Πίνακας 3.1: Χαρακτηριστικά αποτελέσματα χρονισμού των κυκλωμάτων full-custom

Ο επόμενος πίνακας εκθέτει μετρήσεις για το χώρο και την κατανάλωση ισχύος των διαφόρων κυκλωμάτων full-custom. Η τελευταία αποτελεί μέσο όρο από προσομοίωση διάρκειας 100 ns. Η δραστηριότητα στις εισόδους μεγιστοποιήθηκε κατά το δυνατόν. Ο χαρακτηρισμός ''typical'' αναφέρεται σε θερμοκρασία 25^o C, τάση τροφοδοσίας 3.3V και μοντέλα τρανζίστορ τυπικού ρεύματος. Το ''maximum'' αναφέρεται σε θερμοκρασία 0^o C, τάση τροφοδοσίας 3.6 Volt και μοντέλα τρανζίστορ μέγιστου ρεύματος. Ένα σχετικά μεγάλο κλάσμα της ισχύος καταναλώνεται απο τις μνήμες, που είναι σχετικά μεγάλες και πολύπορτες.

Area and Power Dissipation of Full-custom circuits					
block	Ityp (mA)	Ptyp (mWatt)	Imax (mA)	Pmax (mWatt)	Area (mm^2)
decoder A	2.6	8.6	3.6	12.9	0.097
decoder B	2.6	8.6	3.6	12.9	0.097
FlowGroup & OutMask mem	28.0	92.4	31.7	114.1	0.624
Priority Enforcer scheme	1.8	5.9	2.5	9.0	0.41
Mux3-to-1	1.9	6.3	2.1	7.6	0.045
Encoder	2.0	6.6	2.3	8.3	0.246
CreditMask mem	9.7	32.0	11.1	40.0	0.264
HTRF mem	7.0	23.1	7.9	28.5	0.2
Free List	8.4	27.7	10.6	38.0	0.566

†access times are measured from clock

Total	64.0	211.2	75.4	271.3	2.55

Πίνακας 3.2: Η εκτιμούμενη κατανάλωση ισχύος αναμένεται να παρουσιάζει μικρή απόκλιση από την πραγματική λόγω διακυμάνσεων της τεχνολογίας κατασκευής και ανακριβειών στις μετρήσεις

Κεφάλαιο 4

Επεκτάσεις και Συμπεράσματα

4.1 Επεκτάσεις

Η εξέλιξη της τεχνολογίας VLSI κάνει την ύπαρξη ενός κοινόχρηστου ενταμιευτή 512 ή 1024 κυττάρων να φαίνεται εφικτή σήμερα. Η επίδραση όμως της αύξησης του διαθέσιμου χώρου αποθήκευσης επεκτείνεται και έως τον ελεγκτή πολλαπλών ουρών, αφού οι περισσότερες από τις δομές δεδομένων πρέπει να πολλαπλασιαστούν αντίστοιχα στο μέγεθος. Εντούτοις, η ώριμη πια τεχνολογία των 0.35 μm CMOS εισάγει και άλλα επιπλέον πλεονεκτήματα, όπως δεύτερο polysilicon, στροφή των καλωδίων σε γωνία 45^o, και χρήση του πέμπτου επίπεδου μετάλλου με τη μικρότερη αντίσταση, που ήταν αχρησιμοποίητο προς το παρόν. Όλα αυτά, συνδυαζόμενα με την ήδη επωφελή τεχνολογία δεκάτων του μικρομέτρου προσφέρει αισόδοξες προοπτικές για ολοκλήρωση όλο και μεγαλύτερων μνημών σε αυτόν τον μεταγωγέα του ενός chip.

Αρκετές τεχνικές σχεδίασης έχουν αναφερθεί που θα μπορούσαν να υιοθετηθούν για να αυξηθεί η ταχύτητα του διαχειριστή πολλαπλών ουρών. Για παράδειγμα, διαίρεση των μνημών σε μικρότερες, ιεραρχική αποκωδικοποίηση, σχήματα για ταχύτατη αίσθηση μεταβολών τάσης στις SRAM και στις CAM, μείωση της μεταβολής της τάσης των bitlines ή στις γραμμές δεδομένων, ή πιο βαθιές pipeline όσο αφορά το επίπεδο αρχιτεκτονικής. Από την άποψη της τεχνολογίας, η χρήση BiCMOS μπορεί να έχει επωφελή αποτελέσματα. Οι μεγάλοι οδηγητές λόγου χάρη, τα μακρυά σύρματα με μεγάλη χωρητικότητα, οι πλήρεις μεταβολές τάσης σε φορτωμένες γραμμές θα έβλεπαν σημαντικά ωφέλη με διπολικά τρανζίστορ. Παρόλα αυτά υπάρχουν πολλοί παράγοντες που επηρεάζουν τη συναλλαγή

μεταξύ καθυστερήσεων και ισχύος. Επιπλέον, όσο αφορά την αποδοτικότητα σε ενέργεια, μια ποικιλία παραγόντων πρέπει να ληφθούν υπόψη όπως ερευνάται και από τον [CSB92]. Το στυλ της λογικής, η λογική συνάρτηση που υλοποιείται, η τεχνολογία που χρησιμοποιείται είναι από τους κύριους παράγοντες στη σχεδίαση για χαμηλή κατανάλωση. Τέλος, συνιστώσες που συνεισφέρουν στην κατανάλωση ισχύος περιλαμβάνουν ενδιάμεσες μεταβάσεις λόγω καταστάσεων κριτικού ανταγωνισμού, διαρροές και μονοπάτια απευθείας αγωγής ρεύματος, μεταβάσεις προφόρτισης και μεταβάσεις σε αχρησιμοποίητα τμήματα κυκλωμάτων.

Συμπερασματικά, διαφαίνεται εφικτή η δυνατότητα αύξησης του αριθμού συνδέσμων εισόδου/εξόδου και άρα και των γεγονότων που συμβαίνουν στη χρονική διάρκεια ενός κυττάρου, ή ισοδύναμα η δυνατότητα αύξησης του ρυθμού ροής των κυττάρων πάνω από τους ήδη υπάρχοντες συνδέσμους. Οι βελτώσεις αυτές βέβαια απαιτούν λεπτομερέστερη μελέτη και έρευνα.

4.2 Συμπεράσματα

Οι απαιτήσεις των σύγχρονων δικτύων για ποιότητα υπηρεσιών και λειτουργία σε υψηλές ταχύτητες οδηγούν στη χρήση προηγμένων σχημάτων για μεταγωγή και έλεγχο ροής, τα οποία με τη σειρά τους απαιτούν αποδοτική υλοποίηση σε hardware για τη διατήρηση και διαχείριση πολλαπλών ουρών κυττάρων. Αυτή η εργασία παρουσίασε την υλοποίηση σε VLSI του υποσυστήματος διαχείρισης πολλαπλών ουρών του ATLAS I, ενός μεταγωγέα ATM ολοκληρωμένου σε ένα chip. Το chip αυτό, το οποίο υλοποιείται σε τεχνολογία CMOS 0.35 micron, ενσωματώνει αρκετά σημαντικά χαρακτηριστικά: σειριακούς συνδέσμους υψηλής ταχύτητας, έναν κοινόχρηστο ενταμιευτή δεδομένων για 256 κύτταρα, έλεγχο ροής βάσει πιστώσεων, τρία επίπεδα προτεραιοτήτων, και δυνατότητα για multicasting.

Παρουσιάστηκε μία απλή και αποδοτική υλοποίηση της διαχείρισης ουρών, χρησιμοποιώντας τεχνικές σχεδίασης full-custom. Σχεδιάστηκαν μνήμες SRAM δύο, τριών και τεσσάρων πορτών με ειδικές λειτουργίες, μνήμες CAM δύο και τριών πορτών, αποκωδικοποιητές, αποκωδικοποιητές προτεραιότητας και τα περιφεριακά τους κυκλώματα. Επίσης, παρουσιάστηκε ένας νέος μηχανισμός εξυπηρέτησης πολλαπλών ομάδων ροών, ο οποίος εξασφαλίζει δικαιοσύνη μεταξύ των ομάδων ροών.

Το τμήμα της διαχείρισης ουρών που σχεδιάστηκε σε full-custom αποτελείται περίπου

από 65,000 τρανζίστορς σε λογική και 14 Kbits σε μνήμη, καταλαμβάνει χώρο 2.3 mm^2, και οι προσομοιώσεις απέδειξαν ότι λειτουργεί σε συχνότητα 80 MHz υπό τις χειρότερες συνθήκες, και καταναλώνει ισχύ 0.27 Watt.

Appendix A

Pin descriptions of full-custom blocks

Pin Symbol	Type	Description
CLK	Input	Reference Clock All synchronous input signals are registered on all rising edges of CLK. All synchronous signal timings are measured from the rising edges of CLK. All synchronous input signals must meet the setup and hold times.
A0-A7	Input (Synchronous)	Synchronous Address Inputs These address inputs are latched on the rising edge of CLK. All address inputs must meet the setup times referred to the rising edge of CLK.
FGDI0-FGDI11	Input (Synchronous)	Synchronous Data Inputs All data inputs to Flow-Group memory are latched on the rising edge of CLK.
FGSI0-FGSI11	Input (Synchronous)	Synchronous Search Data Inputs These search data inputs to Flow-Group memory are latched on the rising edge of CLK.
FGDOI0-FGDO11	Output (Synchronous)	Data Outputs All data outputs from the Flow-Group memory are latched on the falling edge of CLK.
$\overline{\text{FGCE}}$	Input (Synchronous)	Synchronous Flow-Group enable This input is active low and controls the decoder A and the sense amplifiers of Flow-Group memory, on read operations.

Πίνακας A.1: (continued)

Pin Symbol	Type	Description
FGOMR/$\overline{W}$	Input (Synchronous)	Synchronous Read/write This signal controls the read or write function of Flow-Group and OutMask memories. When it is low performs write operation, while when high and $\overline{FGCE}$ and $\overline{OMCE1}$ are activated a read is performed.
$\overline{SE}$	Input (Synchronous)	Synchronous Flow-Group Search enable This input is active low and is latched on the rising edge of CLK.
$\overline{OMMD}$	Input (Synchronous)	Synchronous OutMask Modify This input is active low and is latched on the rising edge of CLK. It enables a modify (reset) operation.
$\overline{OMCE1}$	Input (Synchronous)	Synchronous OutMask port1 enable This input is active low and controls the sense amplifiers of OutMask memory allowing read operations. It is latched on the rising edge of CLK.
$\overline{OMCE2}$	Input (Synchronous)	Synchronous OutMask port2 enable This input is active low and controls the sense amplifiers of OutMask memory allowing read on modify/read operations. It is latched on the rising edge of CLK.
OMDI0-OMDI16	Input (Synchronous)	Synchronous Data Inputs All data inputs to OutMask memory are latched on the rising edge of CLK.
OMMDI0-OMMDI16	Input (Synchronous)	Synchronous Modify/Data Inputs All modify data inputs to OutMask memory are latched on the rising edge of CLK.
OMSI0-OMSI16	Input (Synchronous)	Synchronous Search Data Inputs All search data inputs to OutMask memory are latched on the rising edge of CLK. These input signals are not enabled by any signal. All bits must be 0 to avoid searching.
OMDO0-OMDO16	Output (Synchronous)	Synchronous Data Outputs All data outputs from OutMask memory are latched on the falling edge of CLK.

Πίνακας A.1: (continued)

Pin Symbol	Type	Description
OMMDO0-OMMDO16	Output (Synchronous)	Synchronous Modify/Data Outputs All modify data outputs from OutMask memory are latched on the falling edge of CLK.
$\overline{\text{MGCE}}$	Input (Synchronous)	Synchronous Management Address Enable This input is active low and controls the decoder B
MGTA0-MGTA7	Input (Synchronous)	Synchronous Management Address Inputs These address inputs are latched on the rising edge of CLK. All address inputs must meet the setup times referred to the rising edge of CLK.
CRDS, MGTS	Input (Asynchronous)	Asynchronous Control Inputs These input signals control the address multiplexer.
$\overline{\text{CRCE}}$	Input (Synchronous)	Synchronous CreditMask Enable This input is active low and controls the sense amplifiers of CreditMask memory allowing read on modify/read operations. It is latched on the rising edge of CLK.
$\overline{\text{CRW}}$	Input (Synchronous)	Synchronous CreditMask Write This input is active low and it is latched on the rising edge of CLK.
$\overline{\text{CRMD}}$	Input (Synchronous)	Synchronous CreditMask Modify Input This input is active low and it is latched on the rising edge of CLK.
CRDI0-CRDI17	Input (Synchronous)	Synchronous CreditMask Data Input All data input signals to CreditMask memory are latched on the rising edge of CLK.
CRMDI0-CRMDI17	Input (Synchronous)	Synchronous Modify/Data Inputs All modify data input signals to CreditMask memory are latched on the rising edge of CLK.
CRMDO0-CRMDO17	Output (Synchronous)	Synchronous Modify/Data Outputs All modify data outputs from CreditMask memory are latched on the falling edge of CLK.

Πίνακας A.1: Pin descriptions of full-custom blocks

Βιβλιογραφία

[All95] Quantum Flow Control Alliance. Quantum Flow Control: A Cell-Relay Protocol Supporting an Available Bit Rate Service. URL: http://www.qfc.org, July 1995. Version 2.0.

[B+90] S. Borkar et al. Supporting Systolic and Memory Communication in iWarp. In *Proceedings of the 17th Int. Symp. on Computer Architecture, ACM SIGARCH*, volume 18, pages 70--81, June 1990.

[CSB92] Anantha Chandrakasan, Samuel Sheng, and Robert Brodersen. Low-power cmos digital design. *Journal of Solid-State Circuits*, 27(4):473--484, April 1992.

[CST88] J. Coudreuse, W. Sincoskie, and J.S. Turner. Guest Editorial in Broadband Packet Communications. *IEEE Journal on Selected Areas in Communications*, 6(8):1452--1454, December 1988.

[D.E81] D.E.Knuth. *The Art of Computer Programming*. Addison-Wesley, Reading PA, 1981.

[DEI95] W. Denzel, A. Engbersen, and I. Iliadis. A Flexible Shared-Buffer Switch for ATM at Gb/s Rates. In *Computer Networks & ISDN Systems*, volume 27, pages 611--624. Elsevier Science B.V., 1995.

[DS87] W. Dally and C. Seitz. Deadlock-Free Message Routing in Multiprocessor Interconnection Networks. *IEEE Transactions on Computers*, C-36(5):547--553, May 1987.

[Gal97] M. Galles. Spider: A High-Speed Network Interconnect. *IEEE Micro*, 17(1):34--39, Jan./Feb 1997.

[G.M85] G.Marsaglia. A Current View of Random Number Generators. In *Computer Science and Statistics, The Interface*. Elsevier Science B.V., 1985.

[HIC95] *IEEE Standard 1355-1995, ISO/IEC Standard 14575 DIS, Standard for Heterogeneous InterConnect (HIC):low-cost, low-latency scalable serial interconnect for parallel system construction*, 1995. URL: http://stdsbbs.ieee.org/groups/1355.

[JC89] J.P.Wade and C.G.Sodini. A Ternary Content Addressable Search Engine. *IEEE Journal on Solid-State Circuits*, 24(4):1003--1013, August 1989.

[J.M96] J.M.Rabaey. *Digital Integrated Circuits: A Design Perspective*. Prentice-Hall, 1996.

[KES+91] T. Kozaki, N. Endo, Y. Sakurai, O. Matsubara, M. Mizukami, and K. Asano. 32x32 Shared Buffer Type ATM Switch VLSI's for B-ISDN's. *IEEE Journal on Sel. Areas in Communications*, 9(8):1239--1247, October 1991.

[Koz96] Christoforos E. Kozyrakis. The Architecture, Operation and Design of the Queue Management Block in the ATLAS I ATM Switch. Technical Report TR-172, Institute of Computer Science - Foundation for Research and Technology Hellas (ICS-FORTH), July 1996. URL: file://ftp.ics.forth.gr/tech-reports/1996/1996.TR172.-QueueManagement.ps.gz.

[KSV96] M. Katevenis, D. Serpanos, and P. Vatsolaki. ATLAS I: A General-Purpose, Single-Chip ATM Switch with Credit-Based Flow Control. In *Proceedings of the Hot Interconnects IV Symposium*, pages 63--73, CA, USA, August 1996. Stanford Univ. URL: file://ftp.ics.forth.gr/tech-reports/1996/1996.HOTI.ATLAS_I_ATM-switchChip.ps.gz.

[KVE95] M. Katevenis, P. Vatsolaki, and A. Efthymiou. Pipelined Memory Shared Buffer for VLSI Switches. In *Proceedings of the ACM SIGCOMM '95 Conference*, pages 39--48, Cambridge, MA., USA, August 1995. URL: file://ftp.ics.forth.gr/tech-reports/1995/1995.SIGCOMM95.PipeMemoryShBuf.ps.gz.

[LeB92] J. LeBoudec. The Asynchronous Transfer Mode: A Tutorial. *Computer Networks and ISDN Systems*, 24(4), May 1992.

[MCLN93] R. Marbot, A. Cofler, J. C. Lebihan, and R. Nezamzadeh. Integration of Multiple Bidirectional Point-to-Point Serial Links in the Gigabits per Second Range. In *Proceedings of the Hot Interconnects I Symposium*, CA, USA, August 1993. Stanford Univ.

[MSDM94] M.Mascagni, S.Cuccaro, D.Pryor, and M.Robinson. A fast, high quality, reproducible, parallel, lagged-Fibonacci pseudorandom number generator. Technical report, Supercomputing Research center, 17100 Science Drive, Bowie, MD 20715, 1994. SRC-TR-94-115.

[NC92] N.Troullinos and C.Stormon. Design Issues in Static Content-Addressable Memory Cells. Technical Report 9208, CASE Center Syracuse University, August 1992.

[PH93] D. Patterson and J. Hennessy. *Computer Organization: the hardware/software interface*. Morgan Kaufman Publishers, 1993.

[SIU+92] Katsuro Sasaki, Koichiro Ishibashi, Kiyotsugu Ueda, Kunihiro Komiyaji, et al. A 7-ns 140mw 1-mb cmos sram with current sense amplifier. *Journal of Solid-State Circuits*, 27(11):1511--1517, November 1992.

[SMS+91] Y. Shobatake, M. Motoyama, E. Shobakate, T. Kamitake, S. Shimizu, M. Noda, and K. Sakaue. A One-Chip Scalable $8 * 8$ ATM Switch LSI Employing Shared Buffer Architecture. *IEEE Journal on Sel. Areas in Communications*, 9(8):1248--1254, October 1991.

[S.S91] S.Sidiropoulos. Fast packet switches for asynchronous transfer mode. Technical Report TR-25, Institute of Computer Science - Foundation for Research and Technology Hellas (ICS-FORTH), Heraklio,Crete,GR, August 1991. URL: file://ftp.ics.forth.gr/tech-reports/1991/1991.TR25.Fast_packet_switches.ps.Z.

[TF88] Y. Tamir and G. Frazier. High-Performance Multi-Queue Buffers for VLSI Communication Switches. In *Proceedings of the 15th Int. Symp. on Computer Architecture, ACM SIGARCH*, volume 16, pages 343--354, May 1988.

[THY+96] T.Miwa, H.Yamada, Y.Hirota, T.Satoh, and H.Hara. A 1-Mb 2-Tr/b Nonvolatile CAM Based on Flash Memory Technologies. *IEEE Journal on Solid-State Circuits*, 31(11):1601--1608, November 1996.

[Tob90] F.A. Tobagi. Fast Packet Switch Architectures for Broadband Integrated Services Digital Networks. In *Proceedings of the IEEE*, volume 78, pages 133--167, January 1990.

[TW73] T.G.Lewis and W.H.Payne. Generalized Feedback Shift Register Pseudorandom Number Algorithm. *Journal of the Association for Computing Machinery*, 20(3):456--468, July 1973.

[WE93] N. Weste and Eshraghian. *Principles of CMOS VLSI Design - a Systems Perspective.* Addison-Wesley, 2 edition, 1993.

[W.W61] W.W.Peterson. *Error Correcting Codes.* MIT Press, Cambridge, Mass, 1961. Appendix C.