

**République Algérienne Démocratique et Populaire**  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
**Université de Batna**  
**Faculté des Sciences de l'Ingénieur**  
**Département d'Informatique**

# Thèse

*En vue de l'obtention du diplôme de*

**Doctorat d'Etat en Informatique**

*Intitulée*

***Une Architecture de Commutateur de  
Paquets utilisant un Réseau  
d'Interconnexion Multi-étages***

*Présentée le 04/07/2005*

*Par*

***Azeddine BILAMI***

*Devant le jury composé de:*

|                   |                          |                                  |
|-------------------|--------------------------|----------------------------------|
| <i>Président</i>  | Mohamed Chaouki Batouche | Prof., Université de Constantine |
| <i>Rapporteur</i> | Mustapha Lalam           | Prof., Université de Tizi Ouzou  |
| <i>Examineurs</i> | Mohamed Benmohammed      | M. C., Université de Constantine |
|                   | Belhadri Messabih        | M. C., Université d'Oran         |
|                   | Brahim Belattar          | M. C., Université de Batna       |

## Résumé

Dans la conception des architectures de commutateurs à hauts débits utilisés dans les réseaux Ethernet et ATM, se pose de nos jours, le problème de l'inadéquation de la vitesse des lignes de transmission, à celle des commutateurs conçus à base de composants électroniques.

Dans un souci de rapidité, les fonctionnalités des commutateurs de paquets à hauts débits, sont généralement réalisées par matériel.

Les architectures de commutateurs avec des tampons en entrées, et celles basées sur l'emploi des tampons en entrées et en sorties combinés, représentent les solutions les plus prometteuses, pour répondre d'une part, aux débits sans cesse croissants des lignes de transmission, et d'autre part, aux besoins en commutateurs de grandes capacités reliant un nombre de plus en plus important d'utilisateurs. L'ordonnancement préalable des paquets avant de les transmettre est l'une des fonctions d'une telle architecture. Les solutions théoriquement valides, basées sur des modèles mathématiques, et visant les performances élevées ont été proposées, et plusieurs algorithmes performants ont été écrits à cet effet. Cependant, la plupart d'entre eux introduisent une grande complexité en temps et en matériel qui influe sur le coût du commutateur, et rend délicate sa réalisation matérielle. Dans la pratique, les algorithmes réellement adoptés, sont ceux qui font appel à de simples heuristiques, n'introduisant pas une grande complexité.

La mémorisation des paquets sous forme de files dans des mémoires tampons, est une deuxième fonction importante d'un commutateur. Les mémoires DRAM classiques qui présentent les meilleurs coûts, pour la réalisation des commutateurs à grande capacité, n'offrent malheureusement qu'une bande passante limitée. Cette dernière contrainte incite à la recherche de nouvelles techniques, qui permettent de maintenir un coût raisonnable au commutateur, tout en améliorant la bande passante de la mémoire.

La fonction de routage des paquets dans un commutateur peut être assurée par différentes architectures utilisant diverses stratégies. Dans la littérature, on retrouve des architectures basées sur le simple bus, dont l'extensibilité dégrade les performances, au réseau Crossbar dont l'extensibilité augmente de manière quadratique le coût, en passant par les architectures multi-étages qui offrent réellement une solution médiane en termes de coût et de performances.

En définitive, nous dirons que dans la mise en œuvre des fonctionnalités d'un commutateur à haut débit, les solutions réellement adoptées, reposent sur un compromis entre la faisabilité, le coût et les hautes performances.

L'objectif de cette thèse est de proposer une architecture d'un commutateur de paquets avec un minimum de complexité matérielle, et donc à coût relativement bas, pouvant être exploitée dans les réseaux ATM, Ethernet ainsi que dans les routeurs Internet. L'utilisation d'un réseau BENES avec un routage automatique comme réseau d'interconnexion, et le choix d'un algorithme WBSA, basé sur une recherche dans une fenêtre, des paquets à transmettre et réalisé par circuit matériel, nous a permis de définir une architecture à complexité réduite et à coût relativement bas. Des simulations ont été élaborées sous un environnement VHDL, pour vérifier le fonctionnement et mesurer les performances en termes de bande passante, latence, retard et pertes de cellules.

**Mots clés:** commutateur, réseau d'interconnexion multi-étages, Crossbar, matrice de commutation, ordonnancement, réseau Benes, routage automatique.

## Abstract

Nowadays, the major constraint to design high performance packet switches, is to answer the speed growth of current and future communication lines. The switch design major challenge, while considering the main functions of 'electronic' packet switches, is to find the best possible compromise between feasibility, ease of implementation and high performances.

Many theoretical solutions related to buffering and routing switch functionalities, have been presented in the literature. Different selection policies and scheduling algorithms, providing high throughput have been proposed. However, in practice many of them, introducing a high complexity, are not foreseeable. The actual solutions are based on the use of some heuristics, to reduce the high complexity of optimum models. The limit of dynamic RAM (DRAM) bandwidth, commonly used in order to provide large buffer storage space with low cost, involve the use of some memory management techniques, to increase the memory bandwidth.

To route packets from input ports to output ports, switches based architectures are commonly used as interconnection networks. Multistage Interconnection Networks (MINs) are preferred to crossbar architecture in case of large switches. MIN(s) are well adapted to VLSI implementations and present a relatively low cost.

Finally, we can say that the design of high speed switches always goes by a compromise (which must be defined clearly and stated) between performances, cost and feasibility.

The objective of this thesis is to define a high speed packet switch with a low cost, introducing a minimum of hardware and time complexity. We suggest for this purpose, an input-queuing architecture, with a simple cell selection policy implemented by hardware, and using a multistage interconnection network. The proposal is described and simulated using a VHDL language. Performances in terms of throughput, latency, delay, and cell loss are evaluated.

**Keywords:** Switch, Multistage Interconnection Network (MIN), Benes Network, self routing, VHDL

# Table des matières

|   |           |
|---|-----------|
| <b>REMERCIEMENTS</b> .....  | <b>2</b>  |
| <b>RESUME</b> .....   | <b>3</b>  |
| <b>ABSTRACT</b> .....   | <b>5</b>  |
| <b>TABLE DES MATIERES</b> .....   | <b>6</b>  |
| <b>LISTE DES FIGURES</b> .....  | <b>9</b>  |
| <b>LISTE DES TABLES</b> .....   | <b>11</b> |
| <b>TERMINOLOGIE ET NOTATIONS</b> .....  | <b>12</b> |
| <b>INTRODUCTION GENERALE</b> .....  | <b>13</b> |
| PROBLEMATIQUE .....   | 14        |
| ORGANISATION DE LA THESE .....  | 15        |
| <b>CHAPITRE 1: COMMUTATEURS DE PAQUETS A HAUTS DEBITS. UN ETAT DE L'ART</b> ..... | <b>17</b> |
| 1.1 INTRODUCTION .....  | 17        |
| 1.2 MODES DE COMMUTATION .....  | 18        |
| 1.2.1 <i>Commutation de circuits</i> .....  | 18        |
| 1.2.2 <i>Commutation de messages</i> .....  | 19        |
| 1.2.3 <i>Commutation de paquets</i> .....   | 19        |
| 1.2.4 <i>Wormhole</i> .....   | 20        |
| 1.3 COMMUTATEUR DE PAQUETS .....  | 21        |
| 1.3.1 <i>Définition</i> .....   | 21        |
| 1.3.2 <i>Classifications des architectures de commutateurs de paquets</i> .....   | 23        |
| 1.3.2.1 Architecture à mémoire partagée.....                                      | 24        |
| 1.3.2.2 Architecture à support partagé .....                                      | 24        |
| 1.3.2.3 Architecture à multiplexage d'espace (SDM) .....                          | 25        |
| 1.3.2.4 les réseaux avec blocage et les réseaux sans blocage .....                | 27        |
| 1.4 COMMUTATEURS ATM.....   | 28        |
| 1.4.1 <i>Routage dans les réseaux ATM</i> .....                                   | 29        |
| 1.4.2 <i>Technique d'auto routage</i> .....                                       | 29        |
| 1.4.2.1 Utilisation d'une table de routage.....                                   | 29        |
| 1.5 EXEMPLES DE COMMUTATEURS DE PAQUETS .....                                     | 30        |
| 1.5.1 <i>GIGA switch</i> .....  | 30        |
| 1.5.2 <i>Knockout</i> .....   | 30        |
| 1.5.3 <i>ForeRunner ASX-200</i> .....   | 31        |
| 1.5.4 <i>Tiny Terra</i> .....   | 31        |
| 1.5.5 <i>Atlas</i> .....  | 31        |
| 1.5.6 <i>Sunshine</i> .....   | 31        |
| 1.5.7 <i>ATOM switch</i> .....  | 31        |
| 1.6 CONCLUSION .....  | 32        |
| <b>CHAPITRE 2: RESEAUX D'INTERCONNEXION</b> .....                                 | <b>33</b> |
| 2.1 INTRODUCTION .....  | 33        |
| 2.2 RESEAUX POINT A POINT .....   | 33        |
| 2.3 SUPPORT PARTAGE.....  | 35        |
| 2.4 RESEAUX A BASE DE COMMUTATEURS .....  | 35        |
| 2.4.1 <i>Le réseau Crossbar</i> .....   | 35        |
| 2.4.2 <i>Réseaux Multi-étages</i> .....   | 36        |
| 2.4.2.1 Le réseau Banyan .....  | 37        |
| 2.4.2.2 Le réseau Butterfly .....   | 37        |
| 2.4.2.3 Réseau Delta .....  | 38        |
| 2.4.2.4 Réseau Oméga.....   | 39        |
| 2.4.2.5 Réseau Clos.....  | 40        |
| 2.4.2.6 Réseau Benes .....  | 40        |
| 2.4.3 <i>Contrôle des réseaux d'interconnexion multi-étages</i> .....             | 41        |

|   |           |
|---|-----------|
| 2.5 PRINCIPALES POLITIQUES DE ROUTAGE.....  | 42        |
| 2.5.1 Fonction de routage.....  | 42        |
| 2.5.2 Politiques déterministes.....   | 43        |
| 2.5.3 Politiques adaptatives.....   | 43        |
| 2.5.4 Routage dans les réseaux multi-étages.....  | 44        |
| 2.5.4.1 Routage centralisé.....   | 44        |
| 2.5.4.2 Routage automatique.....  | 44        |
| 2.6 CONCLUSION.....   | 44        |
| <b>CHAPITRE 3: MISE EN FILES DES PAQUETS.....</b>   | <b>45</b> |
| 3.1 INTRODUCTION.....   | 45        |
| 3.2 MISE EN FILES: UTILISATION DES MEMOIRES TAMPONS.....  | 45        |
| 3.2.1 Tampons en sorties.....   | 46        |
| 3.2.2 Tampons partagés.....   | 47        |
| 3.2.3 Tampons en entrées et en sorties Combinés.....  | 48        |
| 3.2.4 Tampons en entrées.....   | 49        |
| 3.2.5 Mémoire tampon en interne.....  | 50        |
| 3.2.6 Complexité des files.....   | 51        |
| 3.3 GESTION DES FILES DANS LES ARCHITECTURES A TAMPONS EN ENTREES.....  | 53        |
| 3.3.1 Technique FIFO.....   | 53        |
| 3.3.2 Files en sorties virtuelles (VOQ).....  | 54        |
| 3.3.3 Files non FIFO.....   | 55        |
| 3.3.3.1 File Commune partagée.....  | 55        |
| 3.3.3.2 Tampons complètement connectés alloués statiquement (SAFC: Statically allocated fully connected).....           | 55        |
| 3.3.3.3 Files multiples allouées statiquement (SAMQ: Statically Allocated Multi -Queue).....                            | 55        |
| 3.3.3.4 Tampons à Files multiples Allouées dynamiquement. DAMQ (Multiple file Dynamically allocated Multi - Queue)..... | 56        |
| 3.4 CONCLUSION.....   | 57        |
| <b>CHAPITRE 4: L'ORDONNANCEMENT DANS LES COMMUTATEURS DE PAQUETS A TAMPONS EN ENTREES.....</b>                          | <b>58</b> |
| 4.1 INTRODUCTION.....   | 58        |
| 4.2 MAXIMUM MATCHING DANS UN GRAPHE BIPARTITE.....  | 59        |
| 4.3 LES ALGORITHMES D'ORDONNANCEMENT.....   | 60        |
| 4.3.1 First In First Out (FIFO).....  | 61        |
| 4.3.2 Algorithmes à priorité absolue.....   | 61        |
| 4.3.3 Maximum matching.....   | 61        |
| 4.3.4 PIM (Parallel Iterative Matching).....  | 62        |
| 4.3.5 Algorithme LQF (Long Queue First).....  | 64        |
| 4.3.6 Algorithme OCF (Oldest Cell First).....   | 65        |
| 4.3.7 Algorithme RRM (Round Robin Matching).....  | 65        |
| 4.3.8 Algorithme iSLIP.....   | 66        |
| 4.4 CONCLUSION.....   | 67        |
| <b>CHAPITRE 5: UNE ARCHITECTURE DE COMMUTATEUR A TAMPONS EN ENTREES.....</b>  | <b>68</b> |
| 5.1 INTRODUCTION.....   | 68        |
| 5.2 RESEAU DE ROUTAGE.....  | 68        |
| 5.2.1 Réseau Benes.....   | 68        |
| 5.2.2 Construction récursive du réseau Benes.....   | 69        |
| 5.2.3 Caractéristiques du réseau Benes.....   | 70        |
| 5.2.3.1 Chemins disjoints.....  | 70        |
| 5.2.3.2 Tolérance aux pannes.....   | 71        |
| 5.2.4 Réseau de permutations.....   | 72        |
| 5.2.4.1 Définitions.....  | 72        |
| 5.2.4.2 Routage des permutations.....   | 72        |
| 5.2.4.3 Familles de permutations FUB.....   | 73        |
| 5.2.5 Routage automatique (Self Routing).....   | 74        |
| 5.2.5.1 Définition.....   | 75        |
| 5.2.5.2 Techniques de routage automatique.....  | 75        |
| 5.2.5.3 Familles compatibles de permutations.....   | 78        |
| 5.3 ALGORITHME DE ROUTAGE.....  | 79        |
| 5.4 ALGORITHME DE SELECTION.....  | 80        |
| 5.4.1 Stratégies de sélection.....  | 80        |

|  |  |            |
|--|--|------------|
| 5.4.2  | <i>Modèle proposé</i> .....  | 82         |
| 5.4.2.1  | Les mémoires Tampons .....   | 83         |
| 5.4.3  | <i>Principe du circuit de sélection</i> .....                        | 86         |
| 5.4.3.1  | L'algorithme adopté.....   | 86         |
| 5.4.3.2  | Exemple .....  | 89         |
| 5.4.3.3  | Complexité matérielle.....   | 90         |
| 5.4.3.4  | Complexité en temps.....   | 91         |
| 5.4.3.5  | Utilisation d'une CAM.....   | 91         |
| 5.4.4  | <i>Evaluation de l'opération de sélection</i> .....                  | 93         |
| 5.4.4.1  | Cycle de vie d'une sélection d'une permutation.....                  | 95         |
| <b>CHAPITRE 6: SIMULATION VHDL .....</b>                         |  | <b>98</b>  |
| 6.1  | INTRODUCTION .....   | 98         |
| 6.2  | PRESENTATION DE VHDL .....   | 98         |
| 6.2.1  | <i>L'utilisation de VHDL pour l'évaluation de performances</i> ..... | 99         |
| 6.2.2  | <i>L'environnement Active-HDL</i> .....                              | 99         |
| 6.3  | SIMULATION DE L'ARCHITECTURE .....                                   | 100        |
| 6.3.1  | <i>Le réseau d'interconnexion</i> .....                              | 100        |
| 6.3.2  | <i>Files</i> .....   | 105        |
| 6.3.3  | <i>Le circuit de Sélection</i> .....                                 | 106        |
| 6.4  | EVALUATION DES PERFORMANCES .....                                    | 107        |
| <b>BIBLIOGRAPHIE.....</b>  |  | <b>115</b> |
| <b>ANNEXE A : L'ENVIRONNEMENT DE SIMULATION ACTIVE-HDL .....</b> |  | <b>120</b> |
| 1.   | INTRODUCTION .....   | 120        |
| 2.   | COMPOSANTS DE ACTIVE -HDL .....                                      | 120        |
| 2.1.   | <i>Outils de contrôle</i> .....                                      | 121        |
| 2.2.   | <i>Outils d'entrées de conception</i> .....                          | 122        |
| 2.3.   | <i>Spectateurs de la production de la simulation</i> .....           | 123        |
| 2.4.   | <i>Outils de débogage</i> .....                                      | 124        |
| 3.   | LE LANGAGE VHDL.....   | 124        |
| <b>ANNEXE B: EXEMPLES DE RESULTATS DE SIMULATION.....</b>        |  | <b>126</b> |
| 1.   | RESULTATS DE SIMULATION.....   | 126        |
| 2.   | CODE SOURCE VHDL.....  | 131        |