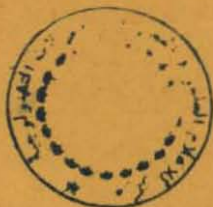


THESE



présentée à

Université Scientifique et Médicale de Grenoble  
Institut National Polytechnique de Grenoble

pour obtenir le grade de

DOCTEUR ES SCIENCES

par

Guy MAZARÉ



STRUCTURES MULTI - MICROPROCESSEURS  
PROBLEMES DE PARALLELISME,  
DEFINITION ET EVALUATION D'UN SYSTEME PARTICULIER.



Thèse soutenue le 19 juin 1978 devant la Commission d'Examen :

Président : L. ROLLIET

Examineurs : F. ANCEAU  
H. GALLAIRE  
S. KRAKOWIAK  
A. RECOQUE  
J.P. VERJUS

BIBLIOTHEQUE DU CERIST

118 1/16 22

## S O M M A I R E

### AVANT-PROPOS

p. 1

### INTRODUCTION - LES STRUCTURES MULTI-MICROPROCESSEURS

|                                                                         |       |
|-------------------------------------------------------------------------|-------|
| 1. Apparition des multi-microprocesseurs                                | p. 5  |
| 1.1. Les microprocesseurs                                               | p. 5  |
| 1.2. Les structures multi-processeurs                                   | p. 9  |
| 1.3. Les multi-microprocesseurs                                         | p. 11 |
| 2. Présentation et classification des structures multi-microprocesseurs | p. 14 |
| 2.1. Intérêts et limites                                                | p. 14 |
| 2.2. Les critères de classement                                         | p. 16 |
| 2.2.1. Critère n° 1 : le contrôle                                       | p. 18 |
| 2.2.2. Critère n° 2 : les communications entre processeurs              | p. 19 |
| 2.2.3. Critère n° 3 : spécialisation ou banalisation                    | p. 22 |
| 2.3. Le classement                                                      | p. 24 |
| 2.4. Revue des différents projets multi-microprocesseurs                | p. 26 |
| 2.4.1. Les systèmes à mémoire centrale, avec ou sans mémoire locale     | p. 26 |
| 2.4.2. Les systèmes à mémoire commune dispersée                         | p. 32 |
| 2.4.3. Les systèmes sans mémoire commune, mais à contrôle hiérarchique  | p. 37 |
| 2.4.4. Les systèmes sans mémoire commune et à contrôle coopératif       | p. 43 |
| 2.5. Principales approches : avantages et inconvénients                 | p. 48 |

|                                                             |       |
|-------------------------------------------------------------|-------|
| 3. Caractéristiques du système multi-microprocesseur étudié | p. 54 |
| 3.1. Motivations et objectifs                               | p. 55 |
| 3.2. Conséquences                                           | p. 57 |

## CHAPITRE 1 - LE PARALLÉLISME

|                                                                |        |
|----------------------------------------------------------------|--------|
| 1. Les différentes organisations parallèles                    | p. 62  |
| 1.1. Décomposition d'un travail en actions parallèles          | p. 62  |
| 1.1.1. Cas d'un ensemble d'actions distinctes                  | p. 63  |
| 1.1.2. Cas d'actions répétitives sur une collection d'objets   | 65     |
| 1.1.3. Combinaison de ces trois organisations parallèles       | p. 70  |
| 1.2. Parallélisme ET et parallélisme OU                        | p. 71  |
| 2. Principaux niveaux de parallélisme                          | p. 73  |
| 3. Existence du parallélisme interne aux algorithmes           | p. 76  |
| 4. Mise en évidence du parallélisme                            | p. 87  |
| 4.1. Langages d'expression du parallélisme                     | p. 88  |
| 4.1.1. Généralités et bibliographie                            | p. 88  |
| 4.1.2. Définition d'un langage adapté au multi-microprocesseur | 92     |
| 4.2. Détection du parallélisme                                 | p. 99  |
| 4.2.1. Rappel des techniques existantes                        | p. 100 |
| 4.2.2. Limites et intérêt de ces méthodes                      | p. 103 |
| 5. Mise en oeuvre du parallélisme                              | p. 107 |
| 6. Evaluation du parallélisme                                  | p. 112 |
| 6.1. Degré de parallélisme                                     | p. 112 |
| 6.2. Efficacité du parallélisme                                | p. 115 |
| 6.3. Mémoire utilisée et localité                              | p. 118 |
| 6.4. Méthodologie d'évaluation et outils                       | p. 122 |

## CHAPITRE 2 - DÉFINITION D'UN SYSTÈME MULTI-MICROPROCESSEUR

|                                                                  |        |
|------------------------------------------------------------------|--------|
| 1. Organisation des processeurs et des mémoires                  | p. 132 |
| 1.1. Mémoire centrale                                            | p. 132 |
| 1.2. Processeurs                                                 | p. 134 |
| 1.3. Mémoires locales                                            | p. 137 |
| 1.4. Dialogues entre processeurs                                 | p. 138 |
| 1.4.1. Démarrage d'un travail sur un processeur de même type     | 138    |
| 1.4.2. Echanges entre processeurs de types différents            | p. 141 |
| 1.4.3. Arrêt d'un processus                                      | p. 144 |
| 1.5. Avantages de ce type de dialogue                            | p. 145 |
| 2. Réalisation de la mémoire et des connexions                   |        |
| processeurs - mémoire                                            | p. 146 |
| 2.1. Mémoire centrale                                            | p. 146 |
| 2.2. Connexions processeurs - mémoire                            | p. 147 |
| 2.3. Mémoires locales et caches                                  | p. 149 |
| 2.4. Problème de cohérence de l'information                      | p. 152 |
| 2.4.1. Première politique                                        | p. 154 |
| 2.4.2. Deuxième politique                                        | p. 155 |
| 2.4.3. Troisième politique                                       | p. 157 |
| 3. Répartition du travail : le mini-O.S.                         | p. 161 |
| 3.1. Motivations et objectifs                                    | p. 161 |
| 3.2. Liste des primitives du mini-O.S., fonctions et utilisation | p. 163 |
| 3.2.1. FORK et JOIN                                              | p. 164 |
| 3.2.2. NFORK et NJOIN                                            | p. 166 |
| 3.2.3. SRF, RFORK, RJOIN                                         | p. 167 |
| 3.2.4. SOLLO et TUTTI                                            | p. 169 |
| 3.2.5. FSFAIS                                                    | p. 170 |
| 3.2.6. MABORT                                                    | p. 172 |
| 3.2.7. Autres primitives                                         | p. 172 |

|                                                         |        |
|---------------------------------------------------------|--------|
| 3.3. Mécanismes du mini-O.S.                            | p. 173 |
| 3.3.1. Principes généraux                               | p. 173 |
| 3.3.2. Contrôle de l'exécution                          | p. 175 |
| 3.3.3. Allocation des compteurs ROR'O                   | p. 175 |
| 3.3.4. Problème de saturation                           | p. 177 |
| 3.3.5. SRF et RFORK                                     | p. 178 |
| 3.3.6. Arrêt de branche en activité                     | p. 178 |
| 3.4. Implémentation de ces mécanismes                   | p. 179 |
| 3.5. Conclusion : intérêt et limites du mini-O.S.       | p. 183 |
| 4. Exécution de langages de haut niveau parallèles      | p. 186 |
| 4.1. Caractéristiques essentielles de ces langages      | p. 186 |
| 4.2. Accès aux variables des différents niveaux         | p. 188 |
| 4.3. Allocation dynamique                               | p. 192 |
| 4.3.1. ALLOUER - RENDRE                                 | p. 193 |
| 4.3.2. NFORK - NJOIN                                    | p. 193 |
| 4.3.3. SRF - RFORK - RJOIN                              | p. 193 |
| 4.3.4. FSFAIS                                           | p. 194 |
| 4.3.5. MABORT                                           | p. 197 |
| 4.3.6. Gestion de la partie libre de la zone de travail | p. 197 |
| 5. Mécanismes de protection                             | p. 199 |
| 5.1. Faisabilité de quelques solutions classiques       | p. 200 |
| 5.1.1. Protection mémoire réelle par clefs et verrous   | p. 200 |
| 5.1.2. Protection des segments par anneaux              | p. 201 |
| 5.2. Systèmes de protection bien adaptés                | p. 203 |
| 6. Système d'exploitation                               | p. 205 |
| 6.1. Organisation générale                              | p. 206 |

|                                                                       |        |
|-----------------------------------------------------------------------|--------|
| 6.2. Quelques points délicats                                         | p. 209 |
| 6.2.1. Décentralisation                                               | p. 209 |
| 6.2.2. Affectation des processeurs aux processus                      | p. 203 |
| 6.2.3. Créations et suppressions de processus ; conséquences          | 209    |
| 6.2.4. Les verrous                                                    | p. 211 |
| 6.3. Surveillance, détection et reprise                               | p. 212 |
| 6.3.1. Horloge                                                        | p. 212 |
| 6.3.2. Comptabilité                                                   | p. 213 |
| 6.3.3. Points d'arrêt et reprises                                     | p. 214 |
| 6.3.4. Processus de garde ; fonctions                                 | p. 214 |
| 6.3.5. Processus de garde ; réalisation                               | p. 215 |
| 6.3.6. Tranches de temps                                              | p. 216 |
| 7. Dégradation progressive : détection des erreurs et reconfiguration | p. 217 |
| 7.1. Détection par microprogrammes de test                            | p. 219 |
| 7.2. Structure à "haute sécurité"                                     | p. 221 |
| 7.2.1. Application au multi-microprocesseur                           |        |
| Cas d'un processeur par molécule                                      | p. 222 |
| 7.2.2. Existence de plusieurs processeurs par molécule                | p. 225 |
| 7.2.3. Intérêt de ces solutions                                       | p. 227 |
| 8. Résumé                                                             | p. 228 |
| CHAPITRE 3 - <u>ÉVALUATION DU SYSTÈME PROPOSÉ</u>                     |        |
| 1. Introduction : les problèmes d'efficacité                          | p. 231 |
| 2. Évaluations ponctuelles                                            | p. 234 |
| 2.1. Évaluations "matérielles" : mémoire, cache, processeurs          | p. 234 |
| 2.1.1. Le langage LASCAR                                              | p. 235 |
| 2.1.2. Le modèle de la molécule                                       | p. 236 |
| 2.1.3. Résultats                                                      | p. 237 |
| 2.2. Fonctionnement du mini-0.S.                                      | p. 240 |

|                                                                      |        |
|----------------------------------------------------------------------|--------|
| 2.3. Efficacité des caches                                           | p. 243 |
| 2.3.1. Caches invalidés lors des TEST & SET                          | p. 243 |
| 2.3.2. Caches contrôlés par PURGE et UPDATE                          | p. 244 |
| 2.4. Etude de quelques programmes parallèles                         | p. 249 |
| 2.4.1. Motivations et méthodologie                                   | p. 249 |
| 2.4.2. Etude du TRI de CP/CMS                                        | p. 252 |
| 2.4.3. Programme MAJPAR de SOCRATE                                   | p. 257 |
| 2.5. Evaluation du parallélisme existant dans certaines applications | 2      |
| 3. Définition d'une expérimentation complète                         | p. 268 |
| 3.1. Motivations et composition de l'expérimentation                 | p. 268 |
| 3.2. Définition d'une maquette                                       | p. 270 |
| 3.2.1. Le processeur                                                 | p. 270 |
| 3.2.2. Mémoire centrale                                              | p. 271 |
| 3.2.3. Caches et processeur molécule                                 | p. 272 |
| 3.2.4. Dispositifs d'entrée-sortie                                   | p. 272 |
| 3.2.5. Organisation générale                                         | p. 273 |
| 3.3. Réalisation de la maquette : émulation                          | p. 275 |
| 3.3.1. Organisation générale                                         | p. 276 |
| 3.3.2. Contraintes de l'émulation                                    | p. 278 |
| 3.3.3. Emulation par tranches de temps fixes                         | p. 278 |
| 3.3.4. Emulation par événements                                      | p. 279 |
| 3.3.5. Remarques générales                                           | p. 281 |
| 3.4. Système d'exploitation                                          | p. 282 |
| 3.4.1. Organisation générale                                         | p. 282 |
| 3.4.2. Le système de la maquette : SPROTO                            | p. 283 |
| 3.5. Langage, applications et mesures                                | p. 284 |
| CONCLUSION                                                           | p. 288 |
| BIBLIOGRAPHIE                                                        | p. 293 |