

République Algérienne démocratique et Populaire
Ministre de l'Enseignement Supérieure et de la Recherche Scientifique
Université Ziane Achour Djelfa

Faculté : Sciences Exactes et Informatique
Département : Mathématiques et Informatique
Filière : Informatique
Spécialité : Systèmes Informatiques (SI)

Intitulé du cours : Architecture de l'ordinateur

Chapitre 04 : Le processeur (Le microprocesseur MIPS 3000)

Unité d'enseignement fondamentale : UEF1
Crédits : 5
Coefficient : 3

Enseignant : Dr. RABEHI Ratiba
Cours, TD et TP : Dr. RABEHI Ratiba
Contact : rabehiratiba@yahoo.fr

Chapitre 04 : Le processeur (Le microprocesseur MIPS 3000)

1. Introduction

Si la carte mère peut être considérée comme le cœur d'un système, le processeur est assimilé au cerveau. Le rôle principal d'un microprocesseur consiste à interpréter et exécuter le programme machine placé dans la mémoire principale. Cette exécution s'effectue instruction après instruction de la première jusqu'à l'instruction STOP, dernière instruction du programme. Ce séquençement de l'exécution des instructions constitue ce que l'on appelle le *flux d'exécution du programme machine*.

Architecture RISC / CISC

Durant la décennie 1970-1980, les processeurs ont progressé en complexité, profitant de l'intégration poussée des transistors pour offrir **des jeux d'instructions plus complets** et **un séquenceur micro-programmé**, malheureusement au détriment des autres composants internes, restreints à la portion congrue sur la puce.

Entre autres, **les données des instructions se trouvent en registre et en mémoire**, même pour les opérations arithmétiques.

En 1981, deux universitaires ont suggéré de réduire drastiquement le jeu d'instructions des processeurs, par exemple en interdisant les opérandes situés en mémoire, sauf pour les transferts, en limitant les modes d'adressage possibles, et en proscrivant les instructions trop compliquées. Ils avaient constaté que seulement 20 % du jeu d'instructions était utilisé dans 80 % des instructions d'un programme standard.

Ils ont alors proposé de construire un processeur ayant un jeu d'instructions réduit (RISC, *Reduced Instruction Set Computer*) par opposition aux processeurs existants (CISC, *Complex Instruction Set Computer*). Cela devait permettre de simplifier fortement le séquenceur, de revenir à **un séquenceur câblé**, d'**introduire un pipeline**, d'optimiser le tout pour **accroître la**

vitesse d'horloge, et d'utiliser la place libérée sur la puce pour augmenter le nombre de registres (et par là même limiter le nombre d'accès mémoire) et introduire la mémoire cache directement dans le processeur. La réduction du jeu d'instructions entraînant un allongement des programmes, les promoteurs de ces processeurs espéraient alors que le ralentissement induit serait plus que compensé par l'accélération des performances des processeurs. En fait, le passage d'un processeur CISC à un processeur RISC transférait la complexité du séquenceur au compilateur, qui devait optimiser le code pour utiliser le pipeline et profiter des nombreux registres.

Pendant longtemps, les partisans des deux types de processeurs se sont affrontés sur le terrain des performances pour aboutir à une synthèse dans les ordinateurs actuels : les techniques des processeurs RISC (séquenceur câblé, pipeline, nombreux registres, cache...) ont été intégrées à tous les processeurs, tandis que les instructions CISC sont toujours présentes *via* un séquenceur plus compliqué, réservé à leur usage.

Cycle Per Instruction « CPI »

C'est un terme utilisé pour décrire un aspect de performance d'un processeur. C'est le nombre de cycles exécutés pendant une instruction. Il est donné par la relation :

$$CPI = \frac{\sum_i(IC_i)(CC_i)}{IC}$$

Tel que : IC_i est le nombre d'instructions d'une instruction type i .

CC_i : Clock Cycle

IC : Compteur total d'instruction.

Le micro-processeur MIPS R3000

Le processeur **MIPS R3000** est un processeur 32 bits industriel conçu dans les années 80. Son jeu d'instructions est de type RISC. Il existe plusieurs réalisations industrielles de cette architecture (SIEMENS, NEC, LSI LOGIC, SILICON GRAPHICS, etc...)

Cette architecture est suffisamment simple pour présenter les principes de base de l'architecture des processeurs, et suffisamment puissante pour supporter un système d'exploitation multi-tâches tel qu'UNIX.

L'architecture externe du MIPS R3000

L'architecture externe représente ce que doit connaître un programmeur souhaitant programmer en assembleur, ou la personne souhaitant écrire un compilateur pour ce processeur :

- Les registres visibles ;
- L'adressage de la mémoire ;
- Le jeu d'instructions ;
- Les mécanismes de traitement des interruptions et exceptions.

Registres visibles du logiciel

Tous les registres visibles du logiciel, c'est à dire ceux dont la valeur peut être lue ou modifiée par les instructions, sont des registres 32 bits.

Afin de mettre en œuvre les mécanismes de protection nécessaires pour un système d'exploitation multi-tâches, le processeur possède deux modes de fonctionnement : utilisateur/superviseur. Ces deux modes de fonctionnement imposent d'avoir deux catégories de registres.

1) Registres non protégés

Le processeur possède 35 registres manipulés par les instructions standards (c'est à dire les instructions qui peuvent s'exécuter aussi bien en mode utilisateur qu'en mode superviseur).

- **R_i** ($0 \leq i \leq 31$) 32 registres généraux

Ces registres sont directement adressés par les instructions, et permettent de stocker des résultats de calculs intermédiaires.

Le registre **R0** est un registre particulier :

- la lecture fournit la valeur constante "0x00000000"
- l'écriture ne modifie pas son contenu.

Le registre **R31** est utilisé par les instructions d'appel de procédures (instructions **BGEZAL**, **BLTZAL**, **JAL** et **JALR**) pour sauvegarder l'adresse de retour.

- **PC** Registre compteur de programme (Program Counter)

Ce registre contient l'adresse de l'instruction en cours d'exécution. Sa valeur est modifiée par toutes les instructions.

- **HI et LO** Registres pour la multiplication ou la division

Ces deux registres 32 bits sont utilisés pour stocker le résultat d'une multiplication ou d'une division, qui est un mot de 64 bits.

Contrairement à d'autres processeurs plus anciens, le processeur R3000 ne possède pas de registres particuliers pour stocker les "codes conditions". Des instructions de comparaison permettent de calculer un booléen qui est stocké dans l'un quelconque des registres généraux. La valeur de ce booléen peut ultérieurement être testée par les instructions de branchement conditionnel.

2) Registres protégés

L'architecture **MIPS** définit 32 registres (numérotés de 0 à 31), qui ne sont accessibles, en lecture comme en écriture, que par les instructions privilégiées (c'est à dire les instructions qui ne peuvent être exécutées qu'en mode superviseur). On dit qu'ils appartiennent au "coprocesseur système". En pratique, cette version du processeur **MIPS R3000** en utilise 4 pour la gestion des interruptions et des exceptions.

- **SR** Registre d'état (Status Register).

Il contient en particulier le bit qui définit le mode : superviseur ou utilisateur, ainsi que les bits de masquage des interruptions. (Ce registre possède le numéro 12)

- **CR** Registre de cause (Cause Register).

En cas d'interruption ou d'exception, son contenu définit la cause pour laquelle on fait appel au programme de traitement des interruptions et des exceptions. (Ce registre possède le numéro 13)

- **EPC** Registre d'exception (Exception Program Counter).

Il contient l'adresse de retour (PC + 4) en cas d'interruption. Il contient l'adresse de l'instruction fautive en cas d'exception (PC). (Ce registre possède le numéro 14)

- **BAR** Registre d'adresse illégale (Bad Address Register).

En cas d'exception de type "adresse illégale", il contient la valeur de l'adresse mal formée.
(Ce registre possède le numéro 8)

Adressage mémoire

1) Adresses octet

Toutes les adresses émises par le processeur sont des adresses octets, ce qui signifie que la mémoire est vue comme un tableau d'octets, qui contient aussi bien les données que les instructions.

Les adresses sont codées sur 32 bits. Les instructions sont codées sur 32 bits. Les échanges de données avec la mémoire se font par mot (4 octets consécutifs), demi-mot (2 octets consécutifs), ou par octet.

2) Calcul d'adresse

Il existe un seul mode d'adressage, consistant à effectuer la somme entre le contenu d'un registre général **R_i**, défini dans l'instruction, et d'un déplacement qui est une valeur immédiate signée, sur 16 bits, contenue également dans l'instruction :

$$\text{adresse} = \mathbf{R_i} + \text{Déplacement}$$

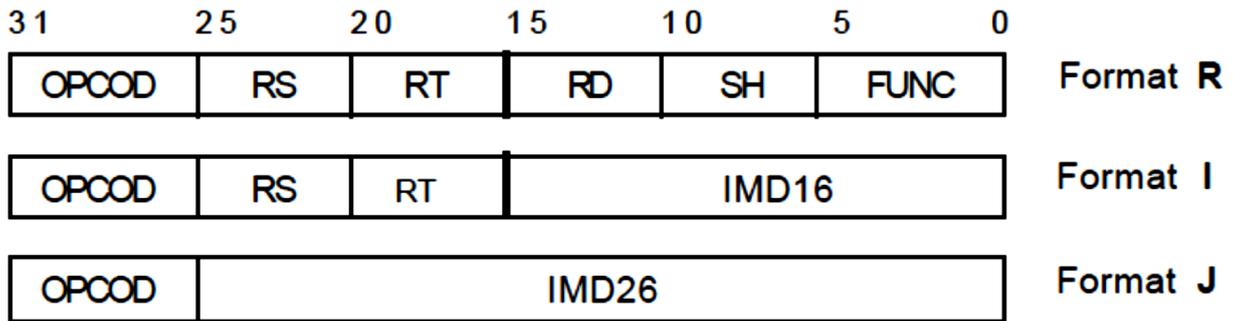
Jeu d'instructions

1) Généralités

Le processeur possède 57 instructions qui se répartissent en 4 classes :

- 33 instructions arithmétiques/logiques entre registres ;
- 12 instructions de branchement ;
- 7 instructions de lecture/écriture mémoire ;
- 5 instructions systèmes.

Toutes les instructions ont une longueur de 32 bits et possèdent un des trois formats suivants :



- Le format **J** n'est utilisé que pour les branchements à longue distance (inconditionnels).
- Le format **I** est utilisé par les instructions de lecture/écriture mémoire, par les instructions utilisant un opérande immédiat, ainsi que par les branchements courte distance (conditionnels).
- Le format **R** est utilisé par les instructions nécessitant 2 registres sources (désignés par RS et RT) et un registre résultat désigné par RD.

L'architecture interne

L'architecture interne dépend des choix de réalisation matérielle. Deux implantations matérielles de cette architecture ont été réalisées à l'Université Pierre et Marie Curie (jusqu'au silicium) dans un but d'enseignement et de recherche : la version micro-programmé R3000_M, simple mais peu performante, et la version pipe-line R3000_P, plus performante mais plus complexe.