

Principes des microprocesseurs



www.jbaumann.ch/jb

Jacques BAUMANN – juin 2003

Table des matières

Introduction.....	3
Eléments fondamentaux de l'ordinateur	4
Microprocesseurs	4
Organisation interne du microprocesseur	5
Unité Arithmétique et Logique (UAL)	5
Le jeu de registres	5
L'unité de commande	6
Bus du microprocesseur	6
Bus d'adresse	6
Bus de données.....	6
Bus de commande.....	6
Programmation du microprocesseur.....	6
Système informatique classique	7
UCT	8
Unité de mémoire	8
Ports d'entrée	8
Ports de sortie	9
Historique des microprocesseurs	9
4004	9
8086	9
80186	9
80286	10
80386	10
80486	10
Pentium	10
Pentium Pro.....	11
Pentium II	11
Pentium III	11
Pentium 4	11
Traitement des instructions.....	12
File d'attente des instructions	13
Registre de segment	13
Pointeur d'instructions IP	14
Registres généraux	15
Registres de pointeurs.....	15
Drapeaux (flags)	15
L'UCT	16
Générateur d'horloge.....	17
Gestion de la mémoire	18
Ports d'entrée / sorties (E/S).....	18
Interruptions.....	19

Interrogations des E/S	19
E/S pilotées par interruptions.....	20
Interruptions logicielles	21
Accès direct à la mémoire (DMA)	21
Type CISC	22
Type RISC.....	23
Programmation en assembleur.....	23

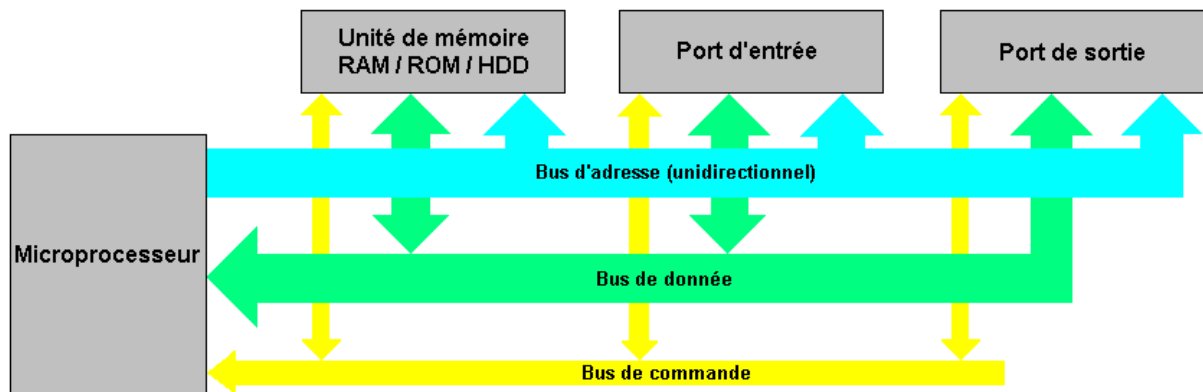
Introduction

Le microprocesseur est un élément indispensable dans un ordinateur. Il s'agit d'un élément semi-conducteur au silicium dont la fabrication nécessite une précision extrême. En effet, le microprocesseur intègre plusieurs millions de transistors utilisés en commutateurs.

Le microprocesseur est non seulement présent dans les ordinateurs, mais aussi dans d'autres types d'appareils électroniques. Ce manuel apporte un approche simple de ce type de circuit complexe.

Éléments fondamentaux de l'ordinateur

Quel que soit son rôle et sa nature, un ordinateur est toujours composé d'éléments fondamentaux. Ceux-ci sont l'Unité Centrale de Traitement (UCT), la mémoire (par exemple RAM, ROM ou disque dur), et les ports d'entrée / sorties, indispensables pour communiquer avec des éléments extérieurs. Ces éléments de base sont reliés entre eux par trois types de bus: le bus d'adresse, le bus de données et le bus de commande. Comme chaque emplacement correspond à une adresse, le bus d'adresse est utilisé pour localiser l'élément souhaité en mémoire. Le bus de donnée est utilisé pour transmettre des données que l'UCT doit traiter. Le bus de commande est utilisé pour coordonner et synchroniser tous ces échanges. Observez le schéma ci-dessous. Il représente les éléments essentiels de l'ordinateur. Les éléments sont reliés entre eux et au microprocesseur grâce aux trois bus que nous venons de mentionner.



Éléments fondamentaux de l'ordinateur

Microprocesseurs

Les microprocesseurs sont des circuits intégrés à très haute intégration présents dans des applications diverses. A l'intérieur de l'ordinateur, il est inséré sur la carte mère, mais nous en trouvons également sur les diverses cartes et périphériques, comme la carte graphique, la carte son, l'imprimante, etc. Il subsiste pourtant une différence importante entre ces microprocesseurs périphérique et l'UCT dessiné sur l'image précédente. Les microprocesseurs périphériques sont spécialisés dans le rôle qui leur est affecté, et sont incapables de communiquer directement entre eux. Dans les applications les plus diverses, nous utilisons le microcontrôleur, un élément très proche du microprocesseur. L'un de ces éléments est illustré ci-dessous.



Exemple de microcontrôleur

Il existe de très nombreux types de processeurs. Celui qui est illustré ci-dessous est un processeur récent de type x86. Ce processeur est à insérer sur la carte mère, à un emplacement prévu à cet effet. Il nécessite un dispositif de refroidissement, car sa fréquence est élevée.



Exemple de microprocesseur

Organisation interne du microprocesseur

Maintenant que nous savons ce qu'est un microprocesseur, penchons-nous sur son organisation intérieure. En effet, un nombre très important de divers éléments est compris dans la puce de silicium que vous avez pu observer ci-dessus.

Le microprocesseur est divisé en trois parties principales. Il s'agit de l'UAL (Unité Arithmétique et Logique), du jeu de registres, et de l'Unité de commande.

L'unité arithmétique et logique est un élément particulièrement important au cœur du microprocesseur. L'unité arithmétique et logique est commandée par l'unité de commande. Son rôle est d'effectuer des opérations mathématiques de base, comme des additions, des soustractions, des multiplications et des divisions. L'unité arithmétique et logique est également capable d'effectuer des opérations logiques, comme les fonctions NON, ET, OU, OU-Exclusif, etc. Les éléments que l'UAL doit traiter proviennent du jeu de registres.

Le jeu de registres

Le jeu de registre contient l'ensemble des registres du microprocesseur. Un registre est une petite partie de mémoire intégrée au microprocesseur, dans le but de recevoir des informations spécifiques, notamment des adresses et des données stockées durant l'exécution d'un programme. Il existe plusieurs types de registres. Certains d'entre eux sont affectés à des opérations d'ordre général et sont accessibles au programmeur à tout moment. Nous disons alors qu'il s'agit de registres généraux. D'autres registres ont des rôles bien plus spécifiques et ne peuvent pas servir à un usage non spécialisé. Enfin, d'autres registres sont invisibles et par conséquent inaccessible au programmeur. Ces registres ne sont accessibles qu'au microprocesseur. Lorsque nous exécutons un programme, l'UAL à toujours accès à ces registres. Nous verrons plus loin qu'il est possible d'affecter des valeurs à notre guise aux registres généraux.

L'unité de commande

Le rôle de l'unité de commande est de coordonner et de synchroniser toutes les opérations que nous venons de décrire. Elle est capable d'envoyer les signaux de synchronisation nécessaires.

Bus du microprocesseur

Le microprocesseur est relié à trois types de bus: le bus d'adresse, le bus de données et le bus de commande. Il s'agit de plusieurs pistes électroniques qui sont reliées au microprocesseur. Ces bus assurent la communication interne et externe du microprocesseur.

Bus d'adresse

Le bus d'adresse est une voie unidirectionnelle qui sert à envoyer un code d'adresse à la mémoire et à communiquer avec des dispositifs extérieurs. La largeur de ce bus est de 32 bits (32 lignes regroupées) sur les microprocesseurs Intel Pentium conventionnels. Cela signifie que ce microprocesseur est capable d'adresser 4295000000 (2^{32}) emplacements en mémoire.

Bus de donnée

Le bus de donnée peut présenter une largeur de 8 à 64 bits, selon l'année et le type du microprocesseur. C'est par le bus de donnée que sont transmises les données qui doivent être traitées par le microprocesseur. A l'inverse, c'est également par le bus de données que transitent les résultats en sortie du microprocesseur. Autrement dit, toutes les données entrantes et sortantes du microprocesseur sont véhiculées par le bus de données.

Bus de commande

Le bus de commande sert à coordonner tous les échanges d'informations décrits précédemment. Le bus de commande véhicule des données qui valident la mémoire et les ports d'entrées / sorties. Le bus de commande introduit des délais d'attente lorsque des informations sont envoyées à un périphérique qui présente une vitesse de traitement réduite. Le bus de commande évite les conflits de bus lorsque deux éléments cherchent à communiquer en même temps.

Programmation du microprocesseur

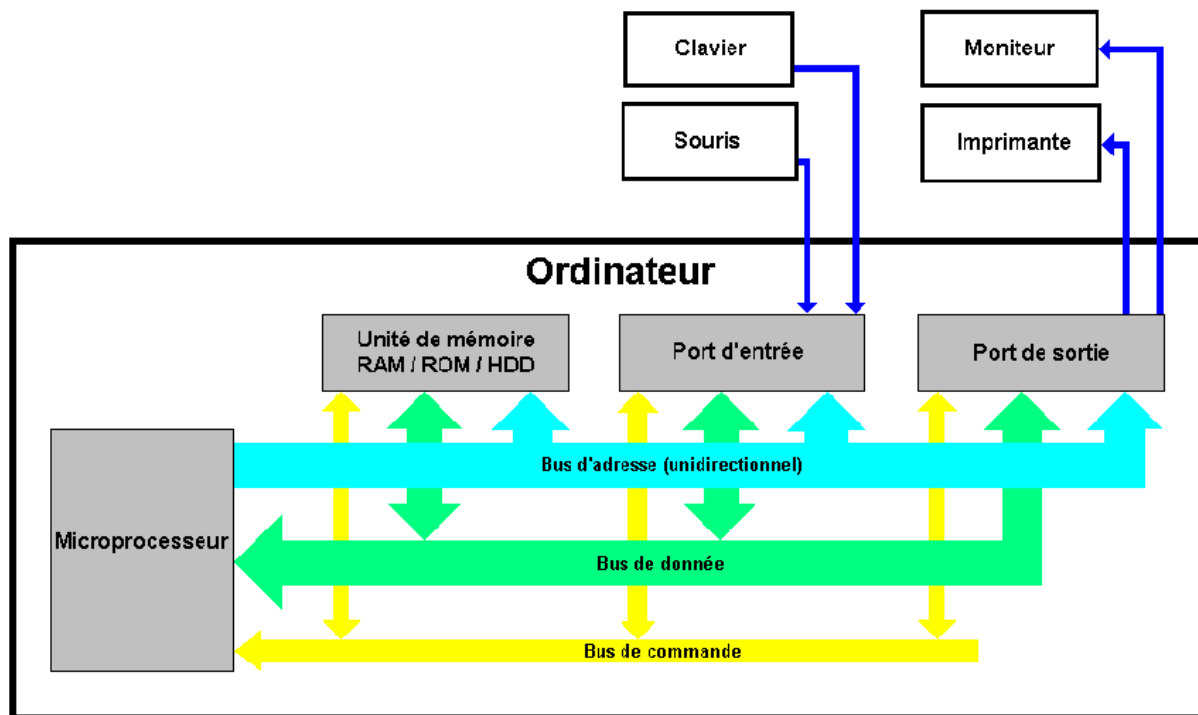
Le microprocesseur travaille à partir d'un jeu d'instruction fixe, élaboré par le fabricant. Dans les processeurs de la famille x86 conventionnels (AMD, Intel...), il existe 7 types d'instructions de base décrits ci-dessous.

- Transfert de donnée
- Arithmétique
- Traitement binaire
- Boucles et branchements
- Chaînes
- Sous-routines et interruptions
- Commande du microprocesseur

Notez qu'il existe des centaines de variantes de ces instructions de base. Chaque instruction est contenue sous la forme d'un code binaire, c'est à dire d'une série de 1 et de 0 que le microprocesseur doit décoder et exécuter. Autrefois, il fallait taper manuellement des séries de 1 et de 0 pour écrire ces instructions. Il s'agissait d'un travail particulièrement contraignant. A cet effet, le langage assembleur a été élaboré. Il s'agit d'un langage de bas niveau, dans lequel les instructions entrées sont directement traduit en langage machine (langage binaire), qui est en réalité le seul langage que le microprocesseur est capable de comprendre. Le langage assembleur et le langage machine qui en résulte sont spécifiques au type de microprocesseur utilisé. Par exemple, si nous écrivons un programme en assembleur sur un ordinateur équipé d'un microprocesseur x86 d'AMD ou Intel, ce programme ne pourra en aucun cas être exécuté sur un ordinateur équipé d'un microprocesseur Motorola. Le langage assembleur est utilisé uniquement dans la programmation des puces et microprocesseurs, mais en aucun cas dans la conception de programmes applicatifs. Pour écrire de tels programmes, nous utilisons des langages de programmation de haut niveau comme le C/C++, le Pascal ou le Basic. Ces langages de programmation nécessitent un compilateur, c'est à dire un interpréteur chargé de traduire le code du langage évolué en langage assembleur. Il est donc plus simple et plus rapide d'utiliser de tels interpréteurs pour élaborer des programmes applicatifs. Pour écrire un programme en assembleur, nous avons au minimum besoin d'un éditeur de texte (par exemple "Bloc-notes" dans Windows). Le langage assembleur sera étudié de façon détaillée plus loin.

Systeme informatique classique

Nous avons précédemment observé les éléments fondamentaux de l'ordinateur. A présent, observons un schéma plus détaillé et plus complet que ce que nous avons vu jusqu'ici.



Système informatique classique

Pour être efficace, l'ordinateur doit pouvoir recevoir et envoyer des informations vers des éléments extérieurs. A présent, examinons chaque élément contenu dans un système informatique typique comme celui que nous connaissons.

UCT: L'UCT (Unité Centrale de Traitement) est le microprocesseur que nous connaissons déjà. Il contrôle l'exécution de chaque programme, adresse des emplacements en mémoire, y envoie des données et en récupère. Il exécute les instructions qui lui sont adressées par l'intermédiaire de la mémoire.

Unité de mémoire: L'unité de mémoire est constituée de mémoire morte (ROM), de mémoire vive (RAM), et de mémoire auxiliaire de masse (disque dur). La mémoire morte (ROM) contient des données sous forme binaire qu'il n'est pas possible de modifier. L'exemple le plus évident est le BIOS (Basic Input / Output System), qui est un programme contenant des informations de configuration générales relatives à l'ordinateur utilisé. Ce programme est contenu dans la mémoire ROM, et, par conséquent, il n'est pas possible de modifier ce programme.

La mémoire RAM est volatile, ce qui signifie qu'elle est incapable de conserver des informations lorsqu'elle n'est pas alimentée. Les informations qui s'y trouvent sont contenues sous forme binaire, et font l'objet d'échanges fréquents avec le microprocesseur.

Le disque dur contient des informations qui peuvent être modifiées à tout moment. Il est capable de conserver les informations qu'il contient durant un temps indéterminé.

Ports d'entrée: L'ordinateur reçoit des informations provenant de l'extérieur par les ports d'entrée. Les périphériques les plus communs utilisant les ports d'entrée sont la souris et le clavier, par exemple. L'UCT peut consulter ces périphériques en empruntant les ports d'entrée.

Ports de sortie: Les ports de sortie sont utilisés par l'ordinateur pour envoyer des informations vers l'extérieur. Les périphériques de sortie les plus courants sont l'écran et l'imprimante, qui reçoivent des données en provenance de l'ordinateur.

Certains périphériques, comme les modems ou les disques durs, sont capables à la fois d'envoyer et de recevoir des données de l'ordinateur. Ces périphériques nécessitent deux ports bien distincts, l'un pour recevoir des données (port d'entrée), l'autre pour envoyer des données (port de sortie).

Historique des microprocesseurs

Avant de nous intéresser plus particulièrement à l'architecture interne des microprocesseurs, il convient de connaître l'évolution impressionnante de ces composants.

4004 (1971): Le premier microprocesseur conçu par Intel était le 4004, conçu à l'origine pour la calculatrice Busicom. Ce microprocesseur possédait un bus de données de 4 bits. Il fonctionnait à la fréquence de 108 KHz, et était équipé de 2300 transistors. Précisons au passage que sa puissance était égale à celle de l'ENIAC, fabriqué en 1946, qui occupait 25 mètres carrés. Ci-dessous, vous pouvez observer une photo du processeur 4004.



Microprocesseur 4004

8086: Le 8086 (développé en 1978) est le premier microprocesseur de type x86. Les microprocesseurs actuels sont en réalité des améliorations du 8086. Le 8086 était équipé d'un bus de données de 16 bits et fonctionnait à des fréquences diverses selon plusieurs variantes: 5, 8 ou 10 MHz.



Microprocesseur 8086

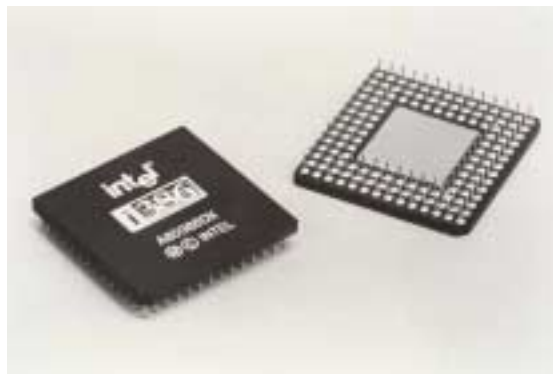
80186: Le 80186 est une version du 8086 légèrement améliorée, qui intègre à la puce plusieurs fonctionnalités supplémentaires, comme le générateur de signal d'horloge, le contrôleur système, le contrôleur d'interruption, et le contrôleur d'accès direct à la mémoire (DMA). La fréquence d'horloge était de 8, 10, et 12,5 MHz.

80286: Le 286 est développé en 1982. Il est capable d'adresser 16777216 emplacements mémoire. L'innovation la plus importante est la capacité de gérer plusieurs programmes à la fois (multitâche), si le système d'exploitation le permet.



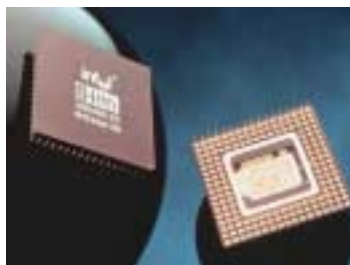
Microprocesseur 286

80386: Le 386 est développé en 1985. Il est doté d'un bus de donnée et d'un bus d'adresse de 32 bits. Le 386SX avait un bus de donnée externe de 16 bits, alors que l'autre variante, le 386DX possédait un bus de donnée externe de 32 bits. Le 386 intègre l'assemblage en pipeline, permettant d'exécuter une instruction avant que l'instruction précédente soit terminée. Le 386 présentait une fréquence de 16, 20, 25, et 33 MHz.



Microprocesseur 386

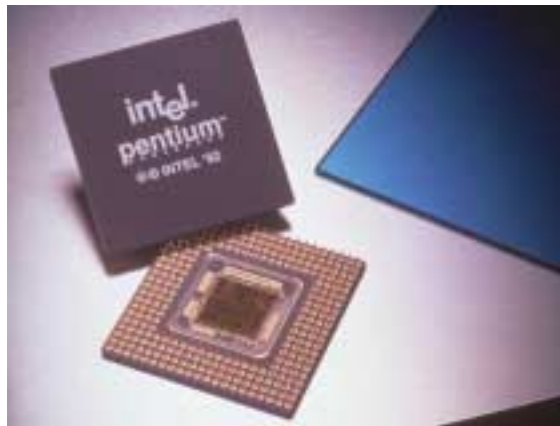
80486: Le 486 est développé en 1989. Il intègre une mémoire cache de 8 Ko, permettant d'accroître sa vitesse de traitement. Le 486 est le premier microprocesseur à être équipé de plus d'un million de transistors. Le 486 présentait des fréquences 33, 40 et 66 MHz.



Microprocesseurs 486

Pentium: Le pentium a été conçu en 1993. Ce microprocesseur est doté de deux mémoires cache de 8 Ko, la première pour les instructions, la seconde pour les données. L'architecture superscalaire (méthode à deux pipeline) permet à ce

microprocesseur d'exécuter deux instructions simultanément. Le pentium original fonctionnait à la fréquence de 66 MHz.



Microprocesseurs pentium

Pentium Pro: Le pentium pro, apparaît en 1995. Il reprend les caractéristiques du pentium original, à l'exception de trois unités de calcul supplémentaire, et d'une seconde puce de mémoire prévue pour accroître la vitesse. La fréquence d'horloge peut atteindre 200 MHz.

Pentium II: Le pentium II est apparu en 1997. Il est doté de la technologie MMX, qui permet d'exploiter plus efficacement le son, la vidéo, et le traitement d'image. A cet effet, pentium II comprend 57 instructions de base supplémentaires. Le pentium II atteint la fréquence d'horloge de 450 MHz.



Microprocesseur Pentium II

Pentium III: Le pentium III est apparu en 1999. 70 nouvelles commandes ont été ajoutées pour améliorer son utilisation dans la vidéo et le traitement d'image 3D. Le pentium III atteint une fréquence de 1 GHz.

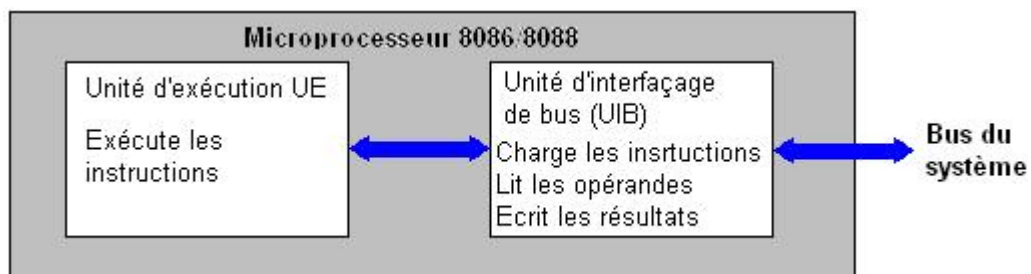
Pentium 4: Le pentium 4 est le microprocesseur utilisé aujourd'hui. Il est capable d'atteindre des fréquences particulièrement élevées, jusqu'à 3 GHz.



Microprocesseur Pentium 4

Traitement des instructions

A présent, penchons-nous sur la façon utilisée par le microprocesseur pour traiter les informations. Il existe deux unités internes distinctes: l'UE (Unité d'Exécution) et l'UIB (Unité d'Interfaçage de Bus). Le rôle de l'UIB est de récupérer et stocker les informations à traiter, et d'établir les transmissions avec les bus du système. L'UE exécute les instructions qui lui sont transmises par l'UIB. L'image ci-dessous résume les notions présentées ici. Le microprocesseur pris comme exemple est le 8086/8088. Les processeurs actuels de la famille x86 traitent les informations de la même façon.



Unités internes de traitement des données

Nous pouvons à présent examiner plus en détail le traitement des instructions par l'UE et l'UIB. Avec le microprocesseur 8085, le traitement des instructions se passait comme suit:

- Extraction des instructions par l'UIB
- Exécution des instructions
- Extraction des nouvelles instructions

Lorsque l'exécution d'une instruction est terminée, l'UE reste inactif un court instant, pendant que l'UIB extrait l'instruction suivante. Pour remédier à ce temps d'attente, le *prétraitement* ou *traitement pipeline* a été introduit dans le 8086/8088. Durant que l'UE exécute les informations qui lui sont transmises, l'instruction suivante est

chargée dans l'UIB. Les instructions qui suivront sont placées dans une file d'attente. Lorsque l'UE à fini de traiter une instruction l'UIB lui transmet instantanément l'instruction suivante, et charge la troisième instruction en vue de la transmettre à l'UE. De cette façon, l'UE est continuellement en activité. Ci-dessous, vous pouvez observer un schéma plus détaillé de l'UE et l'UIB. Nous y retrouvons les éléments dont il à été question précédemment, comme les registres généraux et l'Unité Arithmétique et Logique (UAL).

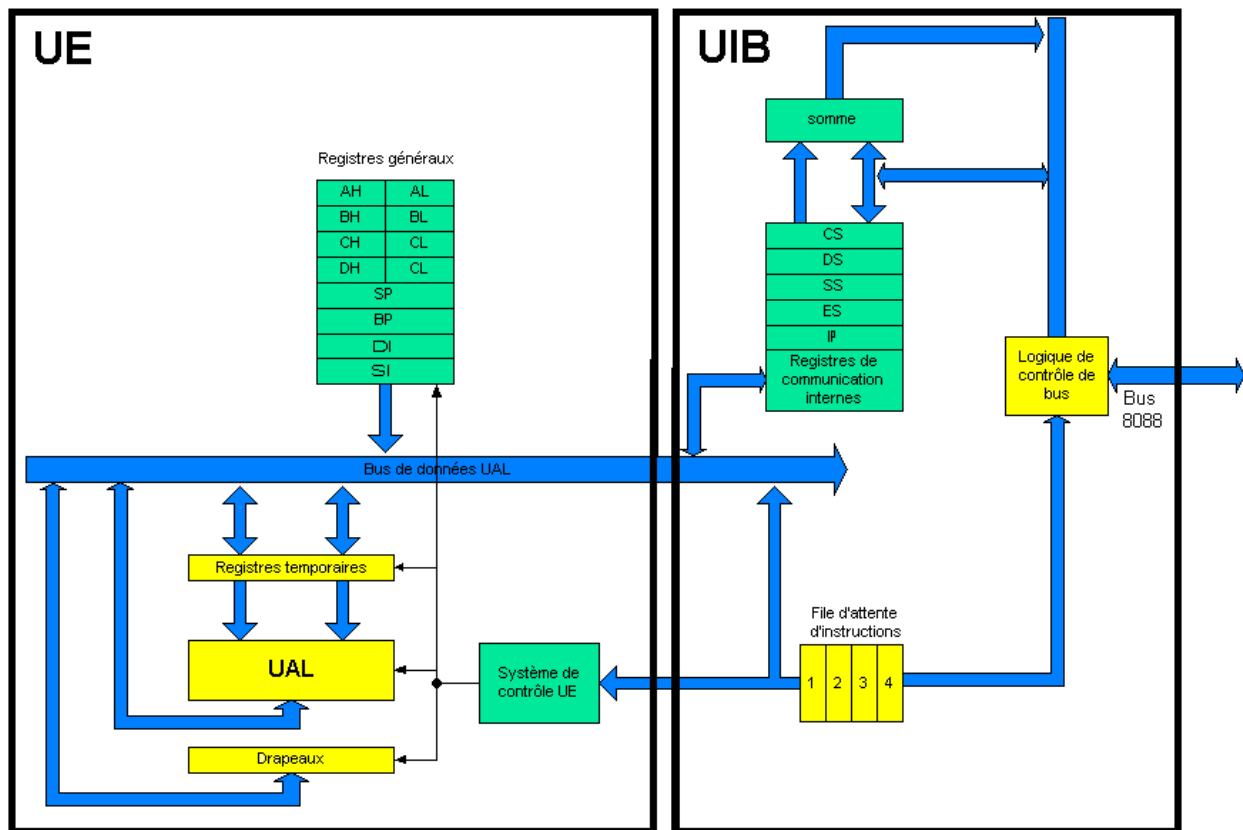


Schéma détaillé du traitement des instructions

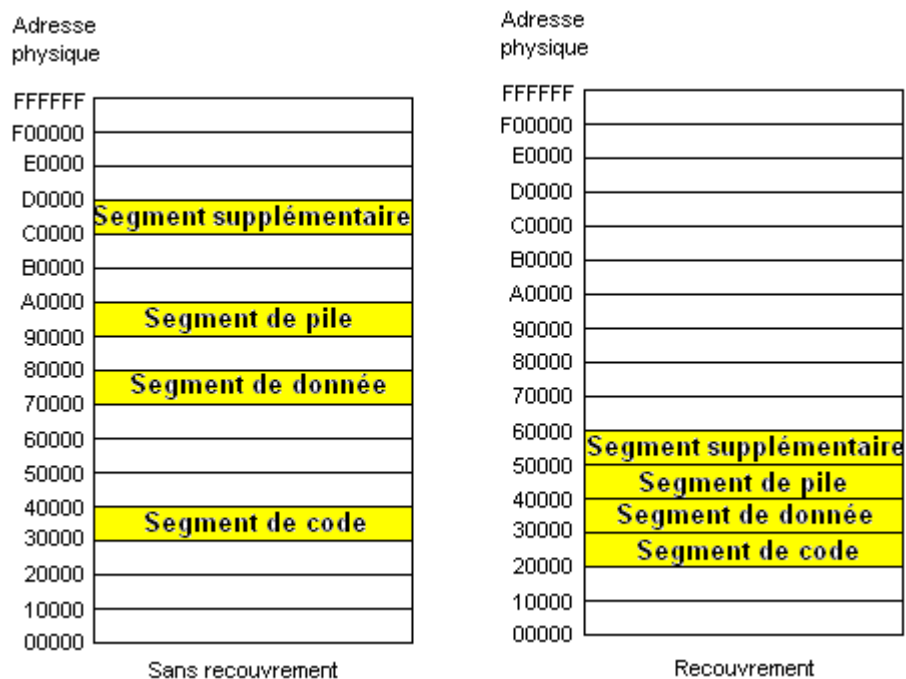
File d'attente d'instructions

Nous avons déjà mentionné la file d'attente d'instructions. Celle-ci contient des informations qui attendent d'être traités par l'UE. La file d'attente est parfois appelée *capacité de traitement*. Le microprocesseur 8088 est capable de mémoriser jusqu'à six octets. Les microprocesseurs actuels sont bien entendu équipé d'une file d'attente plus rapide et plus large, c'est à dire capable d'emmagasiner plus d'informations.

Registres de segment

Le microprocesseur 8088 possède quatre registres de segment comprenant 16 bits chacun. Il s'agit des registres CS, DS, SS et ES.

La mémoire vive est divisée en plusieurs zones de 64 Ko appelées segments, comme illustré. Ces registres peuvent être modifiés, mais cela n'est généralement pas nécessaire.



Segments en mémoire

Les registres de segments contiennent l'adresse de segments spécifiques, comme le montre le tableau ci-dessous. Notez que le recouvrement est une méthode visant à regrouper les 4 segments en mémoire dans des emplacements regroupés.

Registres de segment	Contenu
CS (Code Segment)	Segment où se trouve le programme source
DS (Data Segment)	Segment où se trouve les données
SS (Stack Segment)	Segment où se trouve la pile
ES (Extra Segment)	Segment où se trouve les données supplémentaires

Le pointeur d'instructions IP

Le pointeur d'instruction IP contient le décalage de l'instruction suivante en mémoire qui doit être exécutée. Autrement dit, il doit indiquer au processeur la prochaine instruction à exécuter. Le registre IP est constamment modifié après l'exécution de chaque instruction afin qu'il pointe sur l'instruction suivante. Les microprocesseurs de la famille x86 dépendent entièrement du registre IP pour connaître l'instruction suivante.

Les registres généraux

Les registres généraux peuvent être utilisés dans toutes les opérations arithmétiques que le programmeur insère dans le code assembleur. Un registre complet présente une grandeur de 16 bits. Comme le montre le schéma ci-dessous, chaque registre est en réalité divisé en deux registres distincts de 8 bits. De cette façon, nous pouvons utiliser une partie du registre si nous désirons y stocker une valeur n'excédant pas 8 bits. Si, au contraire, la valeur que nous désirons y ranger excède 8 bits, nous utiliserons le registre complet, c'est à dire 16 bits. Nous verrons plus loin qu'il est possible de manipuler facilement les registres généraux.

AH	AL	Accumulateur
BH	BL	Index de base
CH	CL	Index destination
DH	DL	Index source

Les registres généraux

Les registres de pointeurs

Les registres de pointeurs et les registres d'index sont présents au nombre de quatre et font généralement référence à un emplacement en mémoire. Ces quatre registres sont illustrés ci-dessous. Nous comprendrons plus loin l'utilisation exacte de ces pointeurs, lorsque nous aurons abordé la gestion de la pile.

Pointeur de pile	SP	Ensemble de pointeurs et index
Pointeur de base	BP	
Index destination	DI	
Index source	SI	

Registres de pointeurs

Drapeaux (flags)

Les drapeaux sont des indicateurs qui annoncent une condition particulière suite à une opération arithmétique ou logique, comme le montre le schéma ci-dessous.

Drapeaux de contrôle			Drapeaux d'états					
TF	DF	IF	OF	SF	ZF	AF	PF	CF

Drapeaux de contrôle:

- TF: Trace
- DF: Sens
- IF: Autorise interruption

Drapeaux d'états:

- OF: Débordement
- SF: Signe
- ZF: Zéro
- AF: Retenue auxiliaire
- PF: Parité
- CF: Retenue

Drapeaux (indicateurs d'état)

Les drapeaux peuvent modifier la suite des instructions exécutées par le microprocesseur, si cela est nécessaire.

L'UCT

Ci-dessous, vous pouvez observer le schéma simplifié de l'UCT du microprocesseur 8088. Vous n'avez pas besoin de connaître en détail l'UCT, car cet élément est quelque peu complexe.

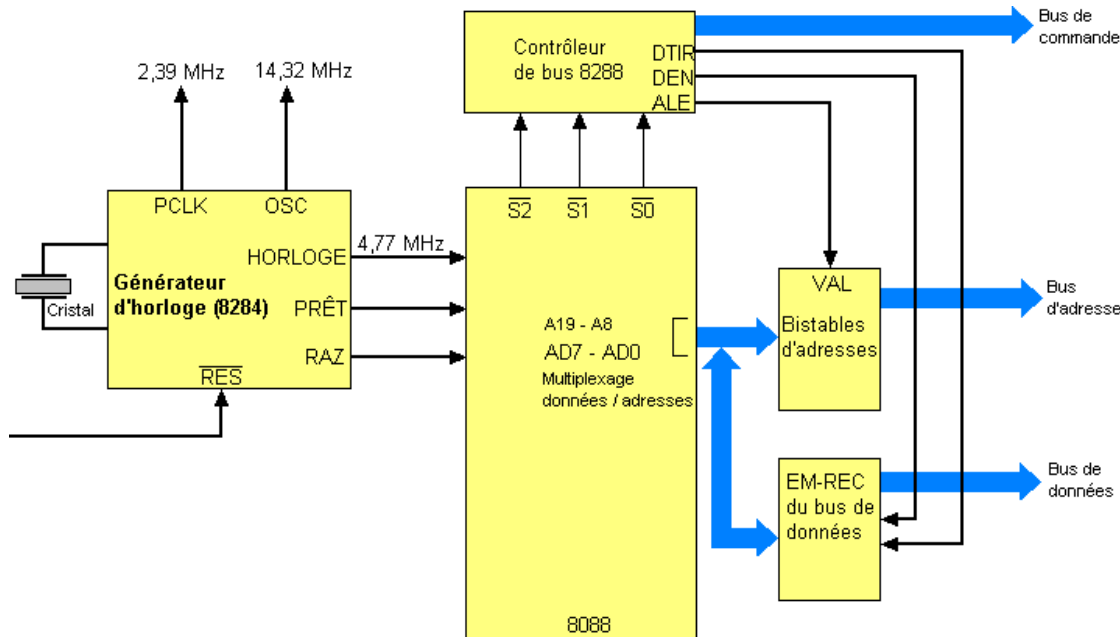


Schéma simplifié de l'UCT

Les lignes que nous pouvons observer ci-dessus étant désignées "AD7-AD0" servent à la fois de lignes d'adresses et de ligne de données grâce à un mécanisme de multiplexage. Les lignes A8-A19 servent uniquement à l'adressage. Nous retrouvons ici les trois bus déjà mentionnés.

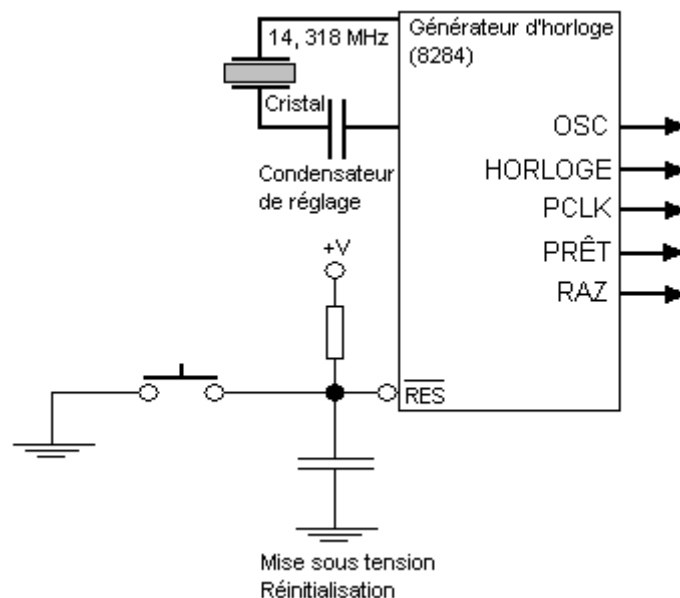
Le contrôleur de bus gère la plupart des fonctions de contrôle du bus à la place du microprocesseur, dans le but d'accroître l'efficacité de traitement de celui-ci.

Lorsqu'un signal ALE est émis, les bistables d'adresses sont verrouillés, et l'envoi d'informations sur le bus d'adresse devient impossible. Le contrôleur de bus peut donc contrôler les informations émises sur le bus d'adresse.

Le générateur d'horloge

Les microprocesseurs et les périphériques ont besoin de signaux de synchronisation précis. A cet effet, nous utilisons le générateur d'horloge, équipé d'un cristal destiné à fournir des signaux précis. Ce cristal génère une oscillation de 14,31818 MHz qui est appliquée à la sortie OSC. Un compteur à division par trois compris à l'intérieur du générateur d'horloge abaisse la fréquence à 4,772727 MHz, qui est appliquée à la sortie HORLOGE. Cette fréquence est ensuite dérivée et ajustée à 2,386363 MHz, puis appliquée à la sortie PCLK. La sortie RAZ sert à remettre le 8088 à 0, et la sortie PRÊT sert à le synchroniser avec le reste du système.

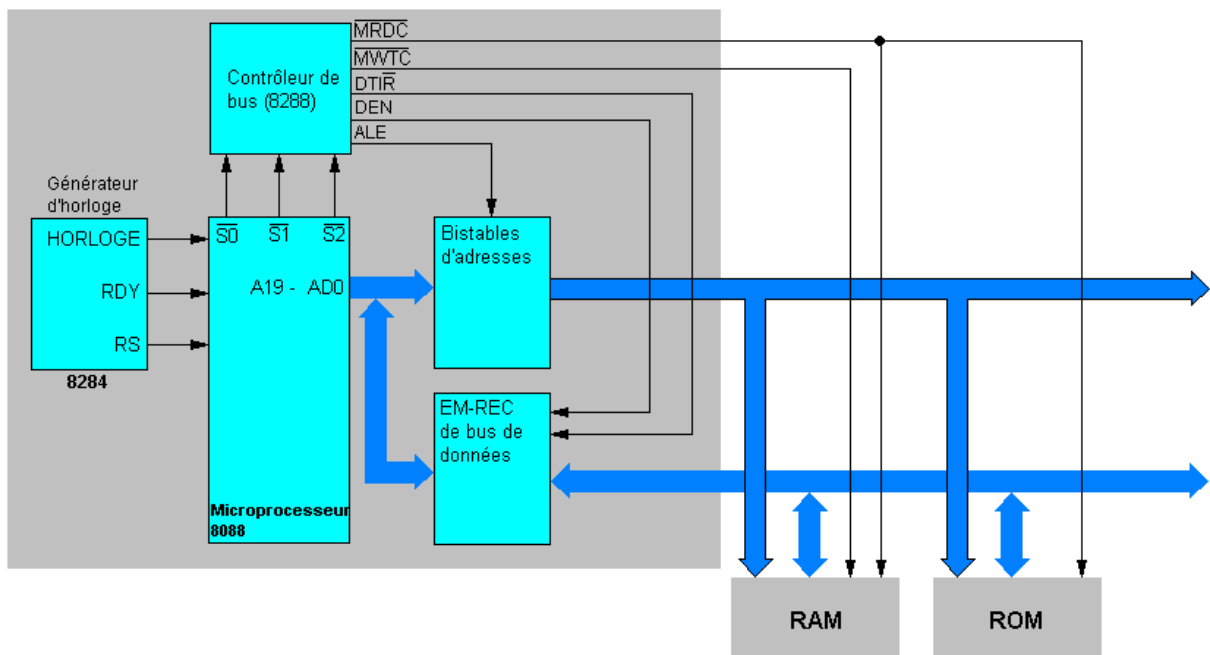
Le Pentium doit être équipé d'un oscillateur externe réglé à la même fréquence que le système, afin de contrôler sa vitesse de base. Ci-dessous, nous pouvons observer le générateur d'horloge 8284, utilisé avec le microprocesseur 8088.



Générateur d'horloge 8284

Gestion de la mémoire

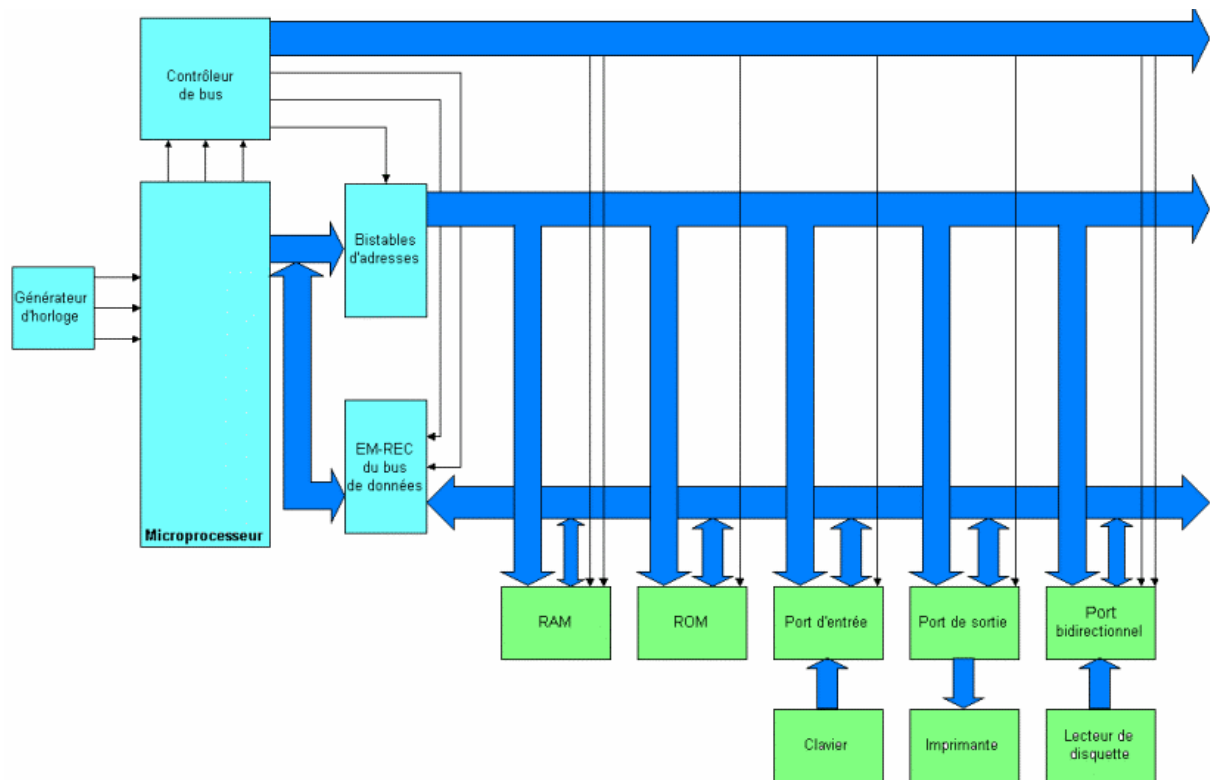
L'UCT est capable de lire et d'écrire dans la mémoire RAM. L'unique opération possible dans la mémoire ROM est bien entendu la lecture. Les opérations de lecture (MRDC) se déroulent en extrayant des données en mémoire. Les opérations d'écriture (MWTC) se déroulent en envoyant une valeur (par exemple le résultat d'une opération arithmétique) vers la mémoire ROM.



Opérations en mémoire

Les ports d'entrée / sortie (E/S)

Pour communiquer avec le monde extérieur, l'ordinateur nécessite des ports d'entrée et de sortie. Nous trouvons des ports utilisés exclusivement pour l'entrée, et d'autres ports exclusivement pour la sortie. Il existe bien entendu des ports bidirectionnels. En d'autres termes, les ports d'E/S sont les interfaces entre l'ordinateur et les périphériques de communication. Chaque périphérique nécessite donc un port adapté. Sur l'illustration suivante nous apercevons des ports d'E/S et différents périphériques. Bien entendu, les ports d'E/S sont nombreux, et ils ne sont pas tous illustrés ci-dessous. En effet, le Pentium comprend 65000 ports.



Système informatique typique

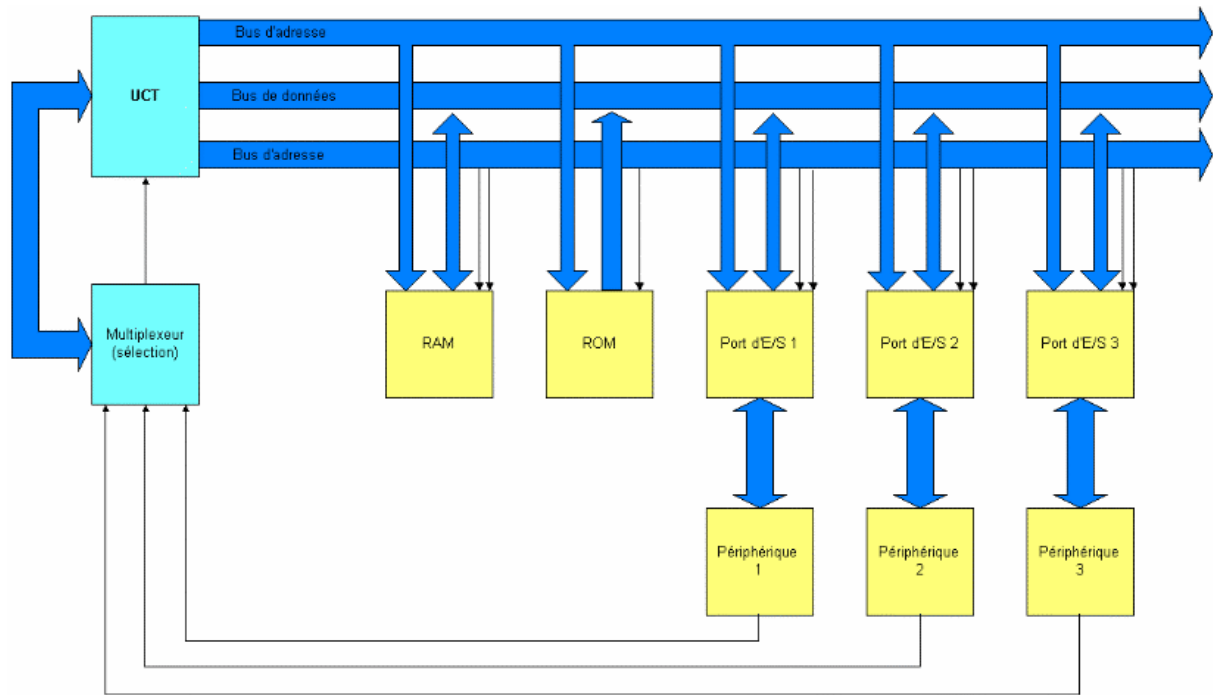
Interruptions

Il arrive fréquemment qu'un périphérique ou un programme nécessite une instruction que seule le microprocesseur peut effectuer. A cet effet, il est possible d'interrompre le microprocesseur quelques instants. Celui-ci va effectuer l'instruction demandée, puis, une fois qu'elle est exécutée, reprend le travail initial. Il existe deux types d'interruptions: les interruptions matérielles provoquées par les périphériques physiques, et les interruptions logicielles provoquées par les programmes. Nous verrons ici les deux types d'interruptions.

Il existe principalement deux façons de générer les interruptions matérielles. L'une par interrogation des E/S, l'autre par E/S pilotées par interruption.

Interrogation des E/S

Dans la méthode d'interruption par interrogation des E/S, le microprocesseur interroge chaque périphérique à tour de rôle, via le multiplexeur. Ainsi, lorsqu'un périphérique a besoin d'une interruption, il peut émettre sa requête quand le microprocesseur l'interroge. Si deux périphériques nécessitent une interruption en même temps, le premier servi sera celui qui sera interrogé en premier par le microprocesseur. Le schéma ci-dessous montre cette façon d'émettre les interruptions.

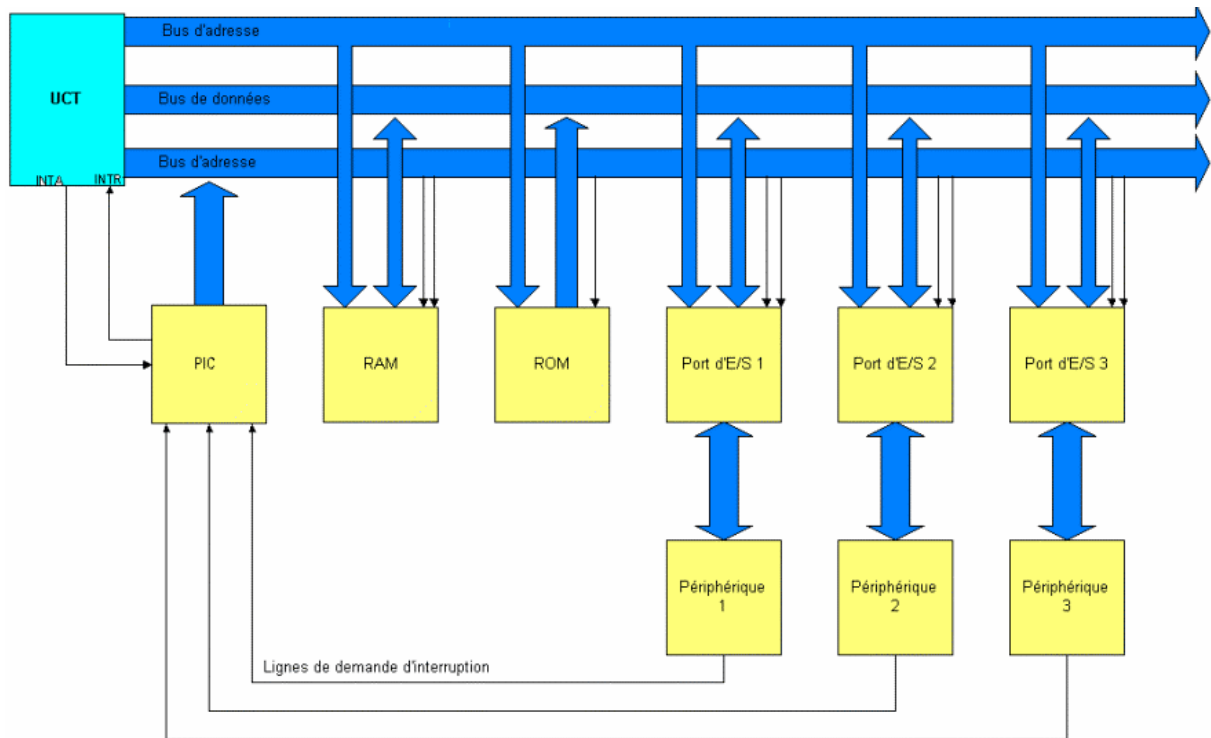


Interruptions par interrogation des périphériques

E/S pilotées par interruption

La méthode des E/S pilotées par interruption présente des avantages certains. Le microprocesseur n'interroge pas les périphériques. Si un périphérique nécessite une interruption, il génère lui-même une demande d'interruption. Le microprocesseur arrête donc le programme en cours et traite l'instruction demandée par le périphérique concerné.

Un numéro de priorité est affecté à chaque périphérique. Un contrôleur d'interruption programmable (PIC) est chargé de gérer les interruptions selon le numéro des périphériques. Si deux périphériques nécessitent une interruption en même temps, le périphérique qui présente le numéro de priorité le plus élevé sera traité en premier. L'image ci-dessous représente ce mode d'interruption.



E/S pilotées par interruption

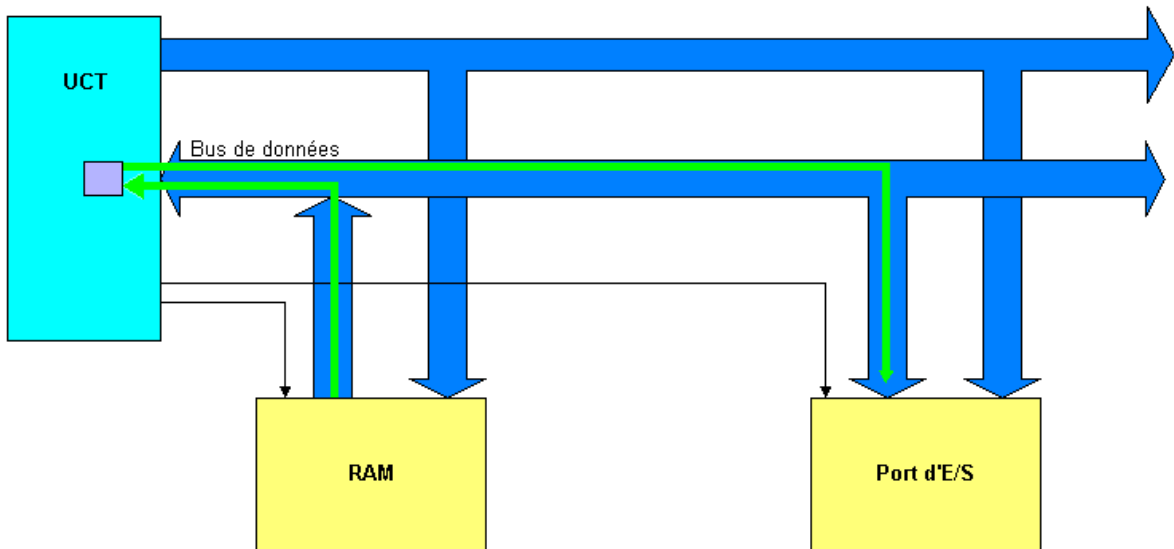
Interruptions logicielles

Les interruptions logicielles sont semblables aux interruptions matérielles. L'unique différence réside dans le fait que les interruptions logicielles sont émises par des programmes. Les cinq premières interruptions sont définies par Intel. Les autres interruptions sont définies par le DOS et le BIOS. Ces interruptions ont une fonction définie, par exemple la lecture et l'écriture sur le disque, l'écriture des données à l'écran, etc.

Accès direct à la mémoire (DMA)

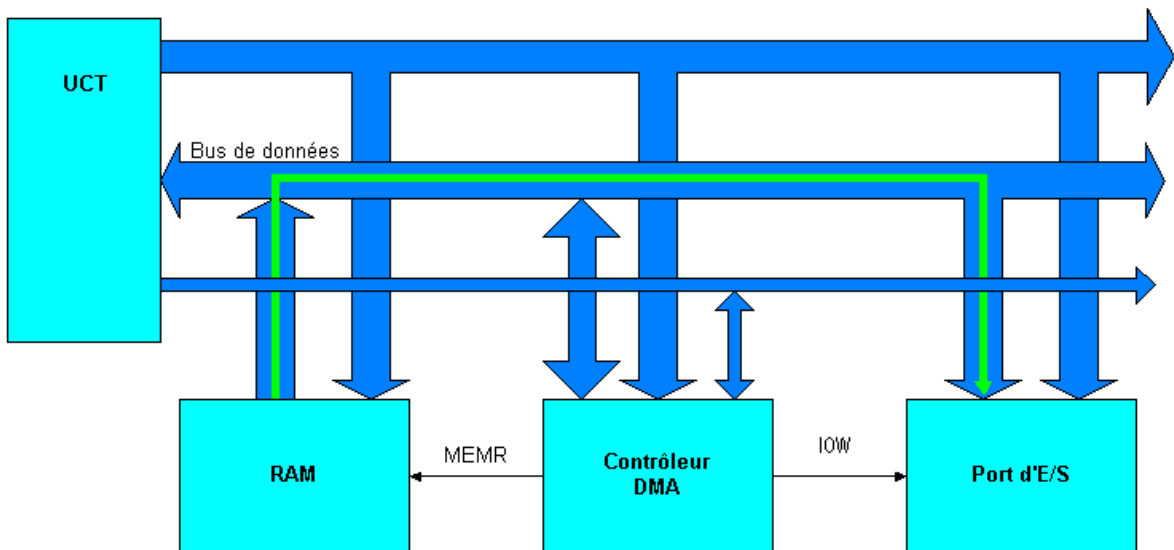
Lorsqu'un transfert en mémoire est nécessaire de la mémoire RAM à un port d'E/S, l'UCT lit le premier octet en mémoire et le charge dans l'un des registres du microprocesseur. L'UCT écrit ensuite l'octet rangé précédemment sur le port d'E/S approprié.

Il en résulte que le microprocesseur effectue des opérations de lecture et d'écriture répétées. Ainsi un certain temps est perdu entre le traitement de chaque octet. Pour remédier à ce problème, nous avons mis au point l'accès direct à la mémoire (Direct Memory Access), qui permet de transférer des données de la mémoire RAM au port d'E/S sans passer par le microprocesseur. Pour cela, nous ajoutons un contrôleur DMA, qui reprend le rôle de l'UCT, c'est à dire qu'il gère les transferts de la RAM aux ports d'E/S. Le processus de transfert passant par l'UCT est illustré ci-dessous. Le parcours des données est illustré en vert.



Transfert utilisant l'UCT

Le concept du transfert direct par la mémoire est illustré ci-dessous. Observez que l'UCT n'est pas active durant le transfert, car elle est remplacée par le contrôleur DMA.



Transfert DMA n'utilisant pas l'UCT

Type CISC

Tous les microprocesseurs que nous avons mentionnés jusqu'à maintenant sont de type CISC (Complex Instruction Set Computer). Toute la famille de microprocesseurs x86 est de type CISC. Ce type de microprocesseur peut traiter des instructions complexes, directement gravées dans le silicium de la puce, pour augmenter la rapidité. Ces instructions sont gravées dans le silicium car elles sont trop complexes

pour être réalisées à l'aide d'un programme interne. L'ajout de ces instructions augmente le prix du microprocesseur CISC. Le microprocesseur CISC ne peut traiter qu'une instruction à la fois. De plus, les instructions complexes ne peuvent pas toutes être traitées en un seul cycle d'horloge. Les microprocesseurs courants fabriqués par Intel, AMD ou Cyrix sont de type CISC.

Type RISC

Le microprocesseur de type RISC (Reduced Instruction Set Computer) n'a pas de fonctions supplémentaires gravées dans la puce de silicium. Il en résulte que les programmes utilisant ce type de microprocesseur doivent utiliser des instructions simples interprétables par le microprocesseur. Cela implique une programmation plus complexe et des compilateurs plus puissants. Les microprocesseurs RISC nécessitent également de plus grandes quantités de mémoire. En revanche, les instructions étant simplifiées, elles peuvent toutes être exécutées en un seul cycle d'horloge. De plus, ce type de microprocesseur est capable de traiter plusieurs instructions en même temps. Les microprocesseurs fabriqués par Motorola sont de type RISC.

Programmation en assembleur

Tout programme doit être transformé en langage machine (suite de code binaire) pour être compréhensible par les circuits internes du microprocesseur. Le langage machine peut être entré manuellement, mais cette tâche est complexe et sujette à de nombreuses erreurs. Le langage assembleur est le niveau supérieur au langage machine, et est utilisé uniquement dans la programmation des microprocesseurs et autres composants semi-conducteurs complexes.