

Deuxième partie

**Conception des
macro-instructions du BIOS**

Chapitre 2

Architecture du PC

Nous avons vu, dans une première étape, comment programmer un microprocesseur et comment est conçu un tel microprocesseur. Le microprocesseur est le cœur d'un (micro-)ordinateur mais ce n'est pas son seul composant, y compris du point de vue de la programmation. Nous avons vu, dans le chapitre précédent, comment utiliser les macro-instructions du BIOS pour effectuer des entrées-sorties. Bien entendu, nous voudrions savoir comment sont conçues les routines de service de ces interruptions. Pour cela il faut connaître l'architecture de l'ordinateur.

Connaître l'architecture d'un ordinateur n'est pas une fin en soi pour le programmeur. Nous allons donc en dire l'essentiel permettant de concevoir les macro-instructions du BIOS, c'est-à-dire les routines de service des interruptions concernées.

2.1 Architecture de l'IBM PC

Le PC d'IBM (pour *Personal Computer*) a été introduit en 1982. Les trois principaux composants matériels de l'IBM-PC sont, comme nous l'avons déjà vu lors de la prise en main utilisateur de celui-ci, l'*unité centrale*, le *clavier* et l'*écran*. Les unités proposées en option sont : le *lecteur de disquettes*, l'*imprimante* et les diverses *cartes d'extension*.

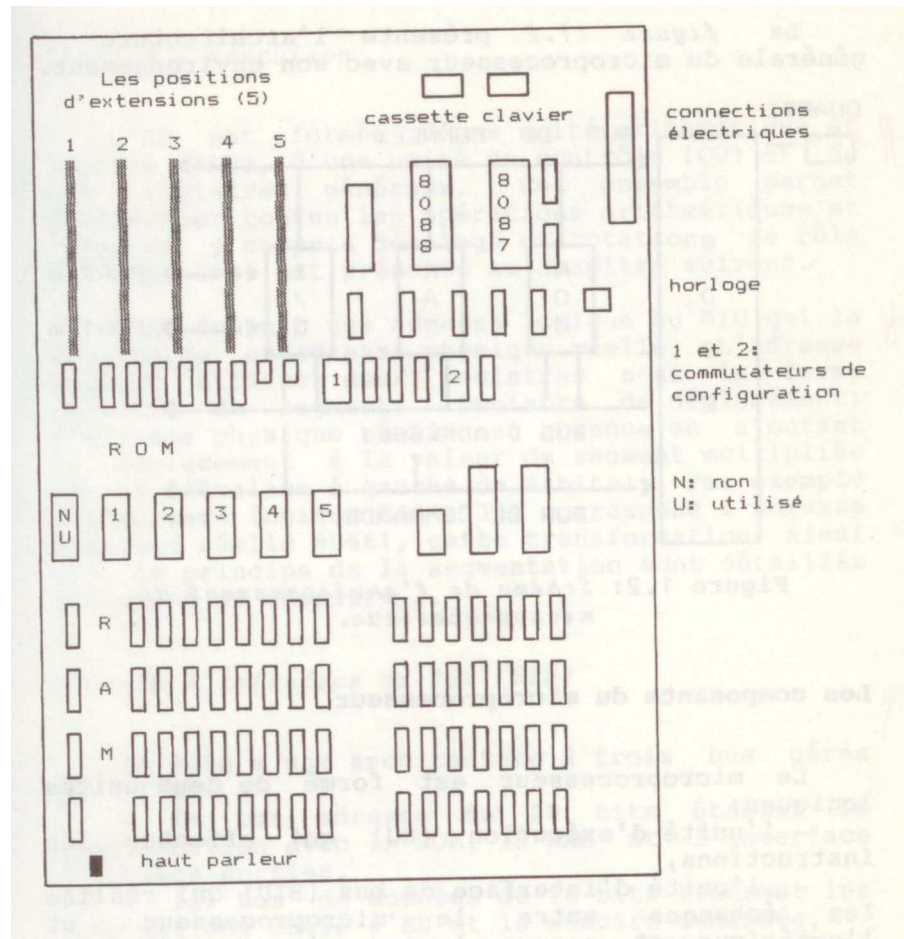


FIGURE 2.1 – Carte mère du PC

L'unité centrale contient la **carte mère** (*motherboard* ou *systemboard* en anglais) et le **bloc d'alimentation** électrique.

Les principaux éléments de la carte mère sont (voir figure 2.1) :

- Le microprocesseur Intel 8088 cadencé à 4,77 MHz.
- 40 KiO de mémoire ROM, décomposés en 32 KiO pour le BASIC et 8 KiO pour le BIOS.
- 64 KiO ou plus de mémoire RAM.
- cinq **adaptateurs d'extension** (*bus slot*) permettant d'ajouter des *cartes d'extension*. Deux de ceux-ci sont utilisés pour une carte graphique et un contrôleur de disquette. IBM met ainsi en œuvre une *architecture ouverte* suivant le principe qui avait fait le succès de l'*Apple II*.
- Des **coupleurs** (*support chip* en anglais), qui sont des circuits intégrés déchargeant le microprocesseur d'une partie de son travail.

Mémoire.- La capacité mémoire était limitée à 1 MiO par l'utilisation du microprocesseur 8088. En fait IBM a choisi de diviser cet espace mémoire en 640 Ko de RAM et 384 Ko de mémoire réservée. De plus seuls 16 KiO de DRAM et 8 KiO de ROM étaient soudés à la carte mère du tout premier PC d'IBM. Les 640 KiO ne pouvaient être obtenus que grâce à une carte d'extension.

Choix des coupleurs.- Les coupleurs du PC sont les suivants :

- Le **contrôleur de bus** (Intel 8288) est utilisé pour engendrer la plupart des signaux du bus de contrôle, en particulier les signaux de lecture et d'écriture de la mémoire ou des entrées-sorties.
- Le **contrôleur DMA** (Intel 8237) permet d'écrire des données directement dans la RAM ou bien de lire des données directement depuis la RAM, sans passer par le microprocesseur. Ceci est surtout utilisé pour les entrées-sorties sur disquette, qui sont relativement lentes. Il comprend quatre canaux. Le canal 0 est dédié au rafraîchissement de la mémoire. Les trois autres canaux sont éventuellement utilisés pour le réseau, le contrôleur de disquette et le contrôleur de disque dur.
- Le **contrôleur d'interruption** (Intel 8259) reçoit les demandes d'interruption (dont plusieurs peuvent survenir en même temps, provenant de différents composants du système). Il attribue un rang de priorité déterminé à chaque demande d'interruption, en fonction de son origine, et retransmet en premier lieu au microprocesseur la demande d'interruption dont le rang de priorité est le plus élevé.
Le 8259A fournit huit entrées d'interruption, appelées **IRQ** pour *Interrupt ReQuest*. IRQ0 est utilisé pour le temporisateur 8253 (voir ci-dessous); IRQ1 est utilisé pour l'interface de périphérie PPI 8255 (voir ci-dessous).
- L'**interface de périphérie** (Intel 8255) établit la liaison entre le microprocesseur et les périphériques tels que le clavier, le haut-parleur, les commutateurs DIP de configuration et le lecteur de cassette.
- Le **générateur d'horloge** (Intel 8284) est utilisé pour obtenir le signal d'horloge de 4,77 MHz du système à partir d'un oscillateur à cristal de 14,318 18 Megahertz. Il sert également à synchroniser d'autres composants du système.
Comme aucun des composants électroniques ne tourne à une fréquence d'horloge aussi élevée, celle-ci est modulée sur chaque composant pour obtenir la fréquence caractéristique de chacun.
- Le **temporisateur** (*timer* en anglais) (Intel 8253) peut être utilisé comme compteur ou comme chronomètre. Il peut émettre des impulsions à intervalle de temps déterminés et constants sur l'un de ses canaux de sortie, plus exactement tous les n tops d'horloge, où n est un paramètre du temporisateur.

Les trois canaux de ce temporisateur sont utilisés pour engendrer les tops d'horloge ($\sim 18,2$ tops/s), pour rafraîchir la DRAM et pour engendrer les sons du haut-parleur.

- Comme le 8088 n'est pas conçu pour manipuler les nombres réels (ou, plus exactement, les nombres à virgule flottante), un **socle** (*socket* en anglais) est prévu sur la carte mère pour ajouter un **coprocesseur mathématique** (Intel 8087). Très peu d'utilisateurs avaient choisi d'acheter, à l'époque, ce circuit intégré d'un coût relativement élevé.

Entrées-sorties isolées et entrées-sorties configurées en mémoire.- L'interfaçage du microprocesseur à un périphérique peut se faire de deux façons différentes :

- La première consiste à utiliser des **ports** avec des adresses différentes de celles de la mémoire ; on parle alors d'**entrées-sorties isolées**.
- La seconde consiste à utiliser une adresse *a priori* utilisée pour y relier un élément de mémoire ; on parle alors d'**entrées-sorties configurées en mémoire**.

L'avantage de la première technique est que toutes les adresses mémoire sont disponibles pour la mémoire. L'avantage de la seconde est qu'il y a en général plus de primitives disponibles : pour une entrée, par exemple, on peut directement ajouter celle-ci au contenu d'un registre.

Dans le cas du MacIntosh on n'utilise que des entrées-sorties configurées en mémoire alors que dans le cas de l'IBM-PC il y a principalement des entrées-sorties isolées.

Limitations du nombre de ports.- Le microprocesseur 8088 a des adresses de ports d'entrées-sorties codées sur 16 bits. Ceci permet donc d'accéder à 65 536 ports. Cependant IBM a décidé de ne décoder que les 10 bits d'adresse de niveau bas, soit A0–A9. Les bits de niveaux plus élevés, à savoir A10–A15, sont ignorés par les décodeurs d'adresse. Ceci signifie que le PC est limité à 1 024 ports différents, d'adresses 0000–03FFh.

De plus les ports doivent être localisés sur la carte mère lorsque A9 est 0 et sur les cartes d'extension lorsque A9 est 1. Ceci signifie que les ports du bus doivent avoir pour adresse 200h–3FFh.

2.2 Le microprocesseur

La première étape de la construction d'un micro-ordinateur est de connaître les caractéristiques du microprocesseur. On a besoin à la fois de savoir comment programmer le microprocesseur mais également de connaître les caractéristiques physiques de celui-ci.

2.2.1 Présentation matérielle du microprocesseur

Notion de modèle matériel.- Jusqu'ici nous avons utilisé un *modèle logiciel* du microprocesseur 8086, bien adapté à l'étude de son comportement. Pour relier ce microprocesseur aux autres composants, afin d'obtenir un ordinateur, nous avons besoin d'un **modèle matériel**, spécifiant en particulier la fonction de chacune de ses broches.

Le microprocesseur.- Le microprocesseur utilisé dans le premier PC de 1981 est un 8088, dont le modèle logiciel est le même que le 8086 mais dont le modèle matériel diffère.

Le 8086, annoncé en 1978, est le premier microprocesseur 16 bits introduit par *Intel*. Il est suivi du 8088 en 1979. Ce dernier est entièrement compatible avec son prédécesseur. La différence réside dans leur architecture matérielle. Le 8086 et le 8088 ont une architecture interne à 16 bits, cependant le 8086 a un bus des données de 16 bits alors que celui du 8088 est de 8 bits.



FIGURE 2.2 – Boîtier du 8088

Aspect extérieur.- Les deux microprocesseurs sont placés dans un boîtier duquel sortent 40 **broches** (*pin* en anglais; voir photo 2.2). Les broches du boîtier peuvent être insérées dans des trous de la carte mère et soudées. Les broches peuvent également être placées dans un *socket* qui, lui, est soudé à la carte mère.

Repérage des broches.- La seule chose qui nous intéresse dans le modèle matériel est le repérage de chacune des 40 broches. Un boîtier est toujours considéré vu de dessus (on doit voir le nom du microprocesseur apparaître). Il y a vingt broches de chaque côté. Pour différencier ces côtés d'une part, et orienter chacun des côtés d'autre part, il y a une encoche semi-circulaire indiquant le « haut » du boîtier. Les broches sont alors numérotées à gauche de 1 à 20 en partant du haut, et à droite de 21 à 40 en partant du bas.

Le schéma 2.3 résume ceci tout en donnant un nom mnémotechnique (décidé par *Intel*) à chacune des broches.

Exercice.- Procurez-vous un 8088 et repérez les broches.

Broches à plusieurs fonctions.- On peut remarquer que certaines broches ont un nom multiple. Ceci vient du fait qu'elles correspondent à plusieurs fonctions.

Par exemple la broche 16 correspond à la fois au bit de poids le plus faible d'une adresse (traditionnellement dénommé A0) et au bit de poids le plus faible d'une donnée (traditionnellement dénommé D0), d'où son nom de AD0. Bien entendu il faut un moyen pour indiquer à un moment donné s'il faut la considérer comme donnant une adresse ou une donnée, moyen que nous verrons plus tard.

On parle de **multiplexage** pour dire qu'une même broche correspond à un bit d'adresse à un certain moment et à un bit de donnée à un autre moment.

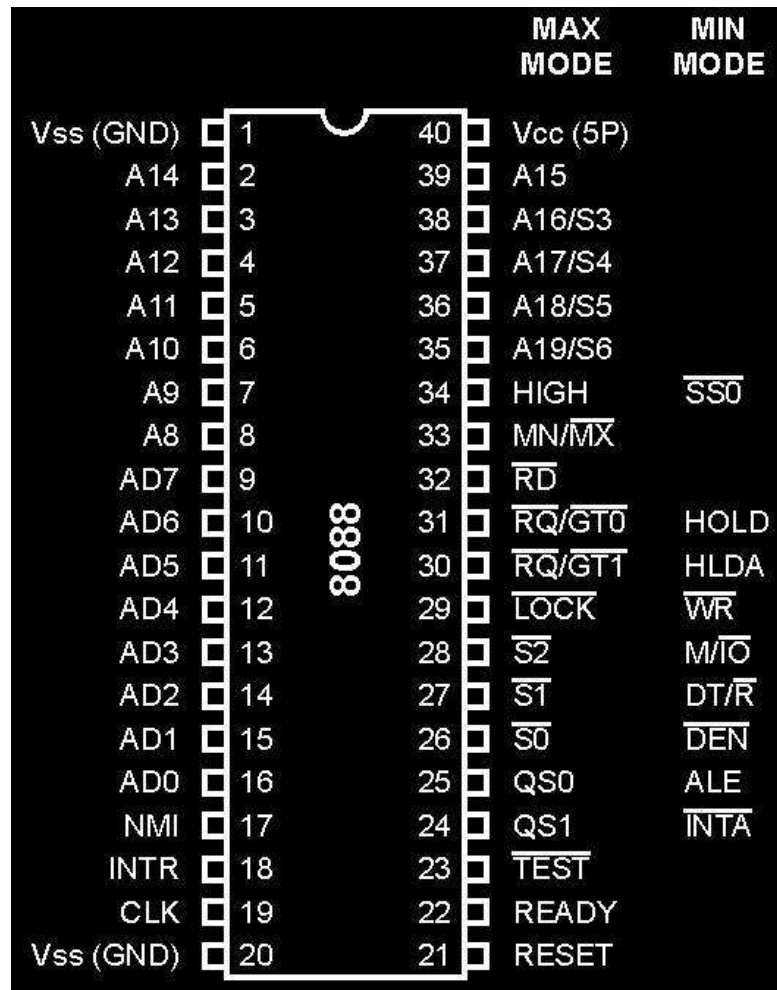


FIGURE 2.3 – Broches du 8088

Modes.- Le microprocesseur 8086/88 fonctionne dans deux modes, appelés **mode minimum** et **mode maximum**. Le mode minimum est sélectionné en appliquant une tension haute à la broche 33 (MN/ \overline{MX}) ; il s'agit du seul mode utilisé par l'IBM-PC. Le mode maximum est sélectionné en appliquant une tension basse à cette même broche 33 ; il sert lors de son utilisation dans de plus gros systèmes que les micro-ordinateurs, comprenant plusieurs microprocesseurs.

2.2.2 Détail du brochage du microprocesseur 8088

Commentons le brochage du microprocesseur 8088 correspondant au mode minimum, le seul qui nous intéresse ici.

Tension.- Une tension de +5 V est fournie à la broche 40 (V_{CC}), la terre (GND pour *ground*) se trouvant à la fois à la fois à la broche 1 et à la broche 20 (ces deux broches doivent donc nécessairement être reliées).

Pourquoi deux broches pour la terre ?

Bus des adresses.- Le **bus des adresses** est utilisé pour apporter l'information sur les adresses à la fois à la mémoire vive et aux ports.

Le bus des adresses (de 20 bits pour la mémoire et de seize bits pour les ports) correspond aux vingt broches A0 à A19 : A0 représente le bit de plus faible poids (celui des unités) et A19 le bit de plus fort poids.

Ces broches sont réparties en trois groupes pour des raisons de multiplexage. A0 à A7 correspondent aux broches (dans l'ordre décroissant) 16 à 9 : elles sont multiplexées avec le bus des données, d'où leur nom effectif de ADO à AD7. Les broches A8 à A14 correspondent aux broches (dans l'ordre décroissant) 8 à 2. Enfin les broches A15 à A19 correspondent aux broches 39 à 35 : les quatre dernières sont multiplexées avec des broches de signaux, d'où leurs noms effectifs de A16/S3 à A19/S6.

Bus des données.- Le **bus des données** permet de recevoir ou des transmettre des données à la mémoire vive et aux ports.

Les données transitent octets par octets sur le 8088, d'où les broches D0 à D7 : D0 correspond au bit de plus faible poids et D7 à celui de poids le plus élevé.

Comme nous venons de le voir ci-dessus, les broches correspondantes sont multiplexées avec des broches du bus des adresses, d'où les noms effectifs de ADO à AD7 (broches 16 à 9).

Signaux de statut.- Les bits des broches S4 et S3 constituent un code (à deux bits) identifiant le registre interne de segment (**cs**, **ds**, **ss** ou **es**) utilisé pour engendrer l'adresse physique située en sortie sur le bus des adresses lors du cycle actuel de bus.

Comme nous l'avons déjà dit, ces broches sont multiplexées avec les broches A17 et A16.

Le code utilisé est le suivant :

S4	S3	Statut de l'adresse
0	0	ES
0	1	SS
1	0	CS
1	1	DS

Le bit de la broche S5 reflète une autre caractéristique interne du microprocesseur. C'est le niveau logique de l'indicateur interne du positionnement de l'interruption.

Le bit de la broche S6 est toujours au niveau logique 0 dans le mode minimum.

Signaux de contrôle.- Les **signaux de contrôle** permettent de contrôler l'échange des données entre le microprocesseur et la mémoire ou les ports. Ils indiquent le moment où une adresse valide se trouve dans le bus, dans quelle direction les données sont transmises, le moment où des données valides en écriture sont sur le bus, à quel moment il faut mettre des données sur le bus.

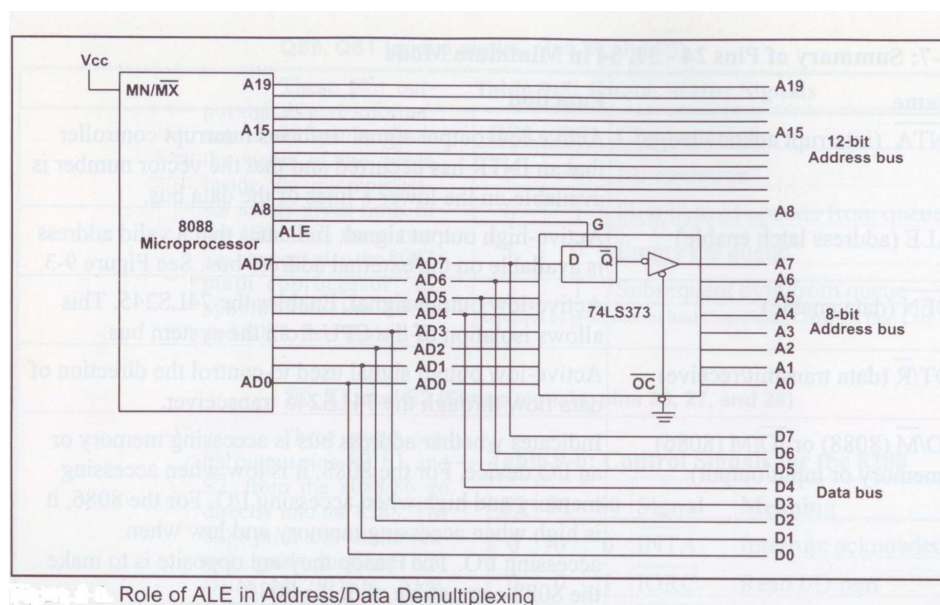


FIGURE 2.4 – Rôle de la broche ALE du 8088

- La broche 25 comporte le signal ALE (pour *Address Latch Enable*) qui par une impulsion à la logique 1, spécifie que les informations sur les broches AD0-AD15 concernent soit une adresse, soit une donnée (voir figure 2.4).
- Les broches 28, 27 et 34 correspondent respectivement aux signaux $\overline{IO/\overline{M}}$ (pour *IO/Memory*), $\overline{DT/\overline{R}}$ (pour *Data Transmit/Receive*) et \overline{SSO} (pour *System Status Output*). Ceux-ci permettent au 8088 de contrôler le type de cycle de bus et la direction d'échange des données sur le bus.
 - Le niveau logique de $\overline{IO/\overline{M}}$ dit au circuit externe que le contenu du bus des adresses correspond soit à un port (niveau 1), soit à de la mémoire vive (niveau 0).
 - Le sens du transfert des données est donné par $\overline{DT/\overline{R}}$: lorsque la broche a la valeur 1, il s'agit de transfert de données du microprocesseur vers l'extérieur ; la valeur 0 indique qu'il s'agit de la réception de données de l'extérieur vers le microprocesseur.

- Le signal de \overline{SSO} , associé aux signaux IO/\overline{M} et DT/\overline{R} , permet de déterminer le cycle du bus en cours selon le tableau suivant :

IO/M	DT/R	SSO	Fonction
0	0	0	Interrupt acknowledge
0	0	1	Memory read
0	1	0	Memory write
0	1	1	Halt
1	0	0	Opcode fetch
1	0	1	I/O read
1	1	0	I/O write
1	1	1	Passive

- Les broches 32, 29 et 26 correspondent à \overline{RD} (pour non *ReaD*), à \overline{WR} (pour non *WRite*) et à \overline{DEN} (pour non *Data ENable*) respectivement.
 - Le microprocesseur met \overline{WR} à 0 pour signaler à l'extérieur qu'une donnée valide en écriture se trouve sur le bus des données.
 - Il met \overline{RD} et \overline{DEN} à 0 pour signaler à l'extérieur qu'il est prêt à lire une donnée sur le bus des données. Plus exactement la mise de \overline{DEN} à 0 active les tampons externes du bus des données.
- La broche 22 correspond au signal **READY**. Celui-ci joue un rôle lors d'un échange avec la mémoire ou un port pour engendrer des temps d'attente (*wait states*) d'un certain nombre de périodes de l'horloge. Il permet à la mémoire ou au port de signaler au microprocesseur le moment où ils sont prêts à compléter le transfert des données.

Signaux d'interruption.- On a plusieurs types d'interruptions :

- Les broches 18 et 24 correspondent aux signaux **INTR** (pour *INTerrupt Request*) et \overline{INTA} (pour non *INTerrupt Acknowledge*) respectivement.

Lorsqu'un périphérique veut signaler qu'il veut intervenir, il envoie un signal à **INTR**. Cette broche est testée à la fin de chaque cycle. Lorsqu'elle est à 1, le microprocesseur termine son cycle et envoie deux impulsions 0 à la broche \overline{INTA} pour en accuser réception.
- La broche 23 correspond au signal **TEST**. L'exécution d'une instruction **WAIT** force le microprocesseur à vérifier le niveau logique de \overline{TEST} . S'il s'agit du niveau 1, le microprocesseur suspend l'opération en cours et se place dans un état appelé *idle state* : il n'exécute plus alors d'instruction proprement dite, il vérifie sans cesse le niveau logique de \overline{TEST} jusqu'à ce qu'il passe à 0. Dans ce dernier cas il passe à l'instruction suivante du programme. Ceci permet de synchroniser le microprocesseur avec un événement d'un périphérique.
- La broche 17 correspond au signal **NMI** (pour *Non Maskable Interrupt*). Lors d'une transition de 0 à 1 de **NMI**, le contrôle est passé à la routine de service d'une interruption non masquable, celle qui possède la priorité la plus haute; aucune interruption logicielle ne peut l'interrompre.
- La broche 21 correspond au signal **RESET**. Un signal 0 à cette broche entraîne une initialisation des registres internes.

Signaux de l'interface DMA.- Il existe deux signaux concernant l'interface DMA (*Direct Memory Access*), HOLD et HLDA, correspondants respectivement aux broches 31 et 30.

- Lorsqu'un périphérique extérieur veut prendre le contrôle du bus, il signale ce fait au microprocesseur en plaçant un signal de niveau 1 sur HOLD. Le microprocesseur termine alors le cycle en cours puis les signaux des lignes AD0 à AD7, A8 à A15, A166/S3 à A19/S6, $\overline{SS0}$, IO/ \overline{M} , DT/ \overline{R} , \overline{RD} , \overline{WR} , \overline{DEN} et INTR sont tous positionnés à l'état Z-haut.
- Le microprocesseur signale alors au périphérique qu'il se trouve dans cet état en plaçant un signal de niveau 1 à HLDA.

Signal de synchronisation.- La broche 19 correspond au signal CLK (c'est-à-dire *CLock*). Elle reçoit les impulsions de synchronisation.

2.2.3 Caractéristiques électriques

Alimentation.- La tension électrique doit être appliquée entre les broches 40 (V_{CC}) et 1 ou 20 (GND). Les deux broches GND, 1 et 20, doivent être reliées ensemble.

La tension à appliquer doit être de +5 V avec une tolérance de 10%. À température ambiante (25 °C), l'intensité est de 340 mA.

Niveaux logiques.- Les caractéristiques électriques pour que le microprocesseur se comporte correctement sont données dans le tableau ci-dessous :

Symbole	Signification	Minimum	Maximum	Condition de test
VIL	Input Low Voltage	- 0.5 V	+ 0.8 V	IOL = 2.0 mA IOH = - 0.4 mA
VIH	Input High Voltage	+ 2.0 V	VCC + 0.5 V	
VOL	Output Low Voltage		+ 0.45 V	
VOH	Output High Voltage	+ 2.4 V		

On voit qu'un microprocesseur est caractérisé par six données : V_{IL} , V_{IH} , V_{OL} , V_{OH} , I_{OL} et I_{OH} .

2.3 Les bus

2.3.1 Caractéristiques électriques des bus

2.3.1.1 Notion

Signal numérique.- Un bus est un ensemble de lignes électriques, chaque ligne ayant un certain niveau discret de différence de potentiel. Lorsque ce niveau varie en fonction du temps, on parle de l'information sur cette ligne comme d'un **signal numérique**.

Caractéristiques électriques des signaux numériques.- Il faut convenir des niveaux de différence de potentiel des signaux numériques. Ces niveaux s'atténuent avec le temps ; il faut donc les entretenir.

De plus, une sortie ne peut être reliée qu'à un certain nombre d'entrées. Il faut donc éventuellement amplifier le niveau de sortie si le nombre d'entrée est supérieur au maximum permis.

Tout ceci concerne les **caractéristiques électriques** des signaux numériques.

2.3.1.2 Niveaux de différence de potentiel des signaux

Nous venons de voir qu'il faut convenir d'une différence de potentiel représentant le niveau 1 et d'une différence de potentiel représentant le niveau 0. Il faut bien sûr une certaine tolérance autour de ces deux valeurs, puisqu'il est impossible physiquement d'obtenir, et surtout de maintenir, une tension donnée. En fait on ne choisit pas vraiment ces valeurs de tension, elles sont imposées par la famille de circuit électronique utilisée.

Exemple 1 : signaux TTL-compatibles.- Beaucoup de microprocesseurs et de circuits numériques fournissent des **signaux TTL-compatibles** (pour *Transistor-Transistor-Logic*). Pour cette famille logique, toute tension inférieure à 0,4 V est considérée comme le niveau logique 0 et toute tension supérieure à 2,4 V est considérée comme le niveau logique 1. On prend comme valeurs représentatives 0,2 V et 3,4 V. Il s'agit ici des signaux de sortie. Un receveur TTL accepte des tensions inférieures à 0,8 V comme niveau 0 et des tensions supérieures à 2,0 V comme niveau 1.

Exemple 2 : signaux CMOS-compatibles.- Une autre famille fréquemment utilisée est celle des **signaux CMOS-compatibles** (pour *Complementary Metal-Oxide Semiconductor*). Les niveaux logiques sont 0 V (pour le niveau 0) et 5 V (pour le niveau 1).

Caractéristiques électriques d'un signal.- Les caractéristiques électriques d'un signal numérique sont au nombre de huit :

- V_{OH} , pour *Voltage Output High*, représente la différence de potentiel minimum pour le niveau 1;
- V_{OL} , pour *Voltage Output Low*, représente la différence de potentiel maximum pour le niveau 0;
- V_{IL} , pour *Voltage Input Low*, représente la différence de potentiel maximum acceptable pour le niveau 0;
- V_{IH} , pour *Voltage Input High*, représente la différence de potentiel minimum acceptable pour le niveau 1;
- I_{IL} , pour *Intensity Input Low*, représente l'intensité maximum pour une entrée de niveau 0;
- I_{IH} , pour *Intensity Input High*, représente l'intensité maximum pour une entrée de niveau 1;
- I_{OH} , pour *Intensity Output High*, représente l'intensité maximum pour une sortie de niveau 1;
- I_{OL} , pour *Intensity Output Low*, représente l'intensité maximum pour une sortie de niveau 0.

L'expéditeur place un signal logique de niveau 1 V_{OH} ou de niveau 0 V_{OL} sur le bus. Celui-ci se propage et est reçu comme V_{IH} ou V_{IL} , les valeurs n'étant pas les mêmes à cause de l'atténuation du signal. Les valeurs dépendent de la famille logique utilisée. Bien entendu, pour que le système de bus fonctionne il faut que :

$$V_{OH} > V_{IH} \text{ et } V_{OL} < V_{IL}.$$

Tableau des caractéristiques.- Le tableau suivant donne ces caractéristiques pour trois familles : TTL, CMOS et LSTTL (pour *Low-power Schottky TTL*).

Nom	TTL	LSTTL	CMOS
V_{OH}	2,4 V	2,7 V	4,6 V
V_{OL}	0,4 V	0,5 V	0,4 V
V_{IL}	0,8 V	0,8 V	1,5 V
V_{IH}	2,0 V	2,0 V	3,5 V
I_{IL}	- 1,6 mA	- 0,4 mA	- 1 μ A
I_{IH}	40 μ A	20 μ A	1 μ A
I_{OH}	- 400 μ A	- 400 μ A	- 360 μ A
I_{OL}	16 mA	8 mA	0,36 mA

Immunité au bruit.- L'immunité au bruit est :

$$\min(V_{OH} - V_{IH}, V_{IL} - V_{OL}).$$

Par exemple, pour une logique TTL on a 0,4 V. Ceci signifie que le bus tolère une impulsion parasite de presque 0,4 V.

2.3.1.3 Forme du signal

Dans le modèle de la logique numérique, le signal est carré. Bien entendu cela n'est pas vrai dans la pratique (d'ailleurs comment obtiendrait-on un segment vertical). La photographie ci-dessous :

photo Uffenbeck, p33

est un instantané d'oscilloscope. On remarque les pics très importants, même s'ils ne durent que très peu de temps.

2.3.1.4 Interface entre signaux

Un signal de sortie ne peut alimenter qu'un certain nombre d'entrées, sinon ses caractéristiques électriques sont trop faibles.

Exemple.- Calculons le nombre de receveurs TTL qu'un transmetteur TTL peut piloter.

Pour le niveau logique 1, un receveur TTL demande $40 \mu\text{A}$. Un transmetteur TTL peut assurer $400 \mu\text{A}$. Ainsi peut-il y avoir jusqu'à 20 receveurs.

Pour le niveau logique 0, un receveur TTL demande $1,6 \text{ mA}$. Un transmetteur TTL peut assurer 16 mA . Ainsi peut-il y avoir jusqu'à 10 receveurs.

Le niveau logique 1 est le pire cas. Le bus est donc limité à 10 receveurs.

Remarque.- On peut voir de même qu'un transmetteur TTL peut piloter un nombre presque illimité de receveurs CMOS (à condition d'élever le niveau logique du transmetteur TTL à 5 V), mais qu'un transmetteur CMOS est limité à un seul receveur TTL.

2.3.1.5 Réflexion et terminateur

La nature physique d'un bus est en général un ensemble de traces parallèles sur une carte de circuits imprimés. Imaginons qu'il s'agisse purement d'une ligne imprimée. L'impulsion d'un transmetteur se propage alors le long de cette ligne et, lorsqu'elle atteint l'une extrémité, elle revient vers le transmetteur, phénomène appelé **réflexion**.

La figure 2.5 montre l'effet de cette réflexion sur une impulsion courte et sur une impulsion longue.

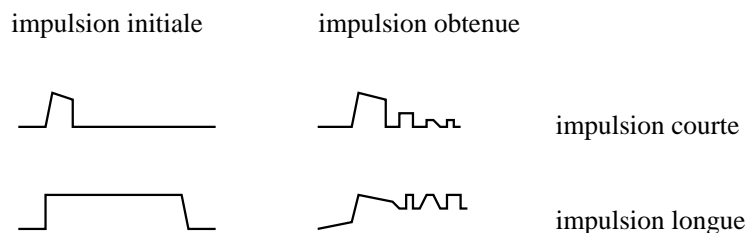


FIGURE 2.5 – Effets du phénomène de réflexion

Si l'impulsion est très courte, l'effet est une série d'impulsions reflétées. Ceci peut produire des résultats erronés à des flip-flops, par exemple.

Si l'impulsion est plus longue, on obtient une série d'ondulations (quelquefois appelée **sonnerie**).

Terminateur passif.- Une première solution consiste à placer un receveur à l'extrémité de la ligne de transmission qui doit absorber toute l'énergie résiduelle de l'impulsion. Ceci est réalisé sous la forme d'une simple résistance. On obtient ainsi un **terminateur passif**.

Il faut que la résistance du receveur soit égale à la résistance de la ligne de transmission, soit 100 à 200 Ω .

Malheureusement cette solution a pour conséquence d'engendrer un courant sur le transmetteur. Ceci abaisse V_{OH} et augmente V_{OL} .

Terminateur actif.- Une meilleure solution consiste à utiliser une division de la différence de potentiel, telle que montrée à la figure 2.6.

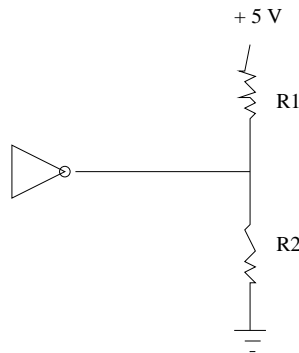


FIGURE 2.6 – Terminateur actif

Si $R_1 = 470 \Omega$ et $R_2 = 220 \Omega$ alors :

$$R_{termination} = 470\Omega || 220\Omega = 150\Omega,$$

$$V_{bus} = \frac{220\Omega}{220\Omega + 470\Omega} \times 5V = 1,6V.$$

Un tel circuit est appelé un **terminateur actif**.

La division de la différence de potentiel assure que l'amplitude de l'impulsion reflétée sera toujours moindre que l'amplitude de l'impulsion incidente. Cette solution convient pour les bus d'une longueur inférieure à 1 m.

2.3.1.6 Amplification des signaux

Nous avons vu qu'un transmetteur ne peut alimenter qu'un nombre restreint de receveurs.

DÉFINITION.- On appelle **bus buffering** toute méthode utilisée pour s'assurer que les niveaux logiques sont bien présents sur le bus.

Trois types de bus.- Du point de vue de l'amplification, on peut distinguer trois types de bus :

- les bus de **type 1** sont caractérisés par un seul transmetteur et plusieurs receveurs. C'est le cas du bus des adresses ;
- les bus de **type 2** sont caractérisés par un seul receveur et plusieurs transmetteurs ;
- les bus de **type 3** sont **bidirectionnels**, ils ont plusieurs receveurs et plusieurs transmetteurs. C'est le cas du bus des données.

Sélection des signaux : notion de tristate

Dans le cas des bus de type 1 ou 3, il peut y avoir
Comment sélectionner le signal ?

2.3.2 Bus du 8088 en mode minimum

Bus des adresses.- Nous avons vu que, sur le 8088, les adresses et les données sont multiplexées. Pour démultiplexer les signaux provenant des broches AD, on utilise une bascule, comme nous l'avons déjà vu à la figure 2.4.

Bus de contrôle.- Les trois signaux \overline{RD} , \overline{WR} et IO/\overline{M} permettent d'engendrer les signaux classiquement appelés \overline{MEMR} pour la lecture en mémoire, \overline{MEMW} pour l'écriture en mémoire, \overline{IOR} pour la lecture sur un périphérique et \overline{IOW} pour l'écriture sur un périphérique, comme l'indique le tableau ci-dessous :

\overline{RD}	\overline{WR}	IO/\overline{M}	Signal
0	1	0	\overline{MEMR}
1	0	0	\overline{MEMW}
0	1	1	\overline{IOR}
1	0	1	\overline{IOW}
0	0	x	Ne se produit pas

On produit ces signaux de la façon montrée sur la figure 2.7.

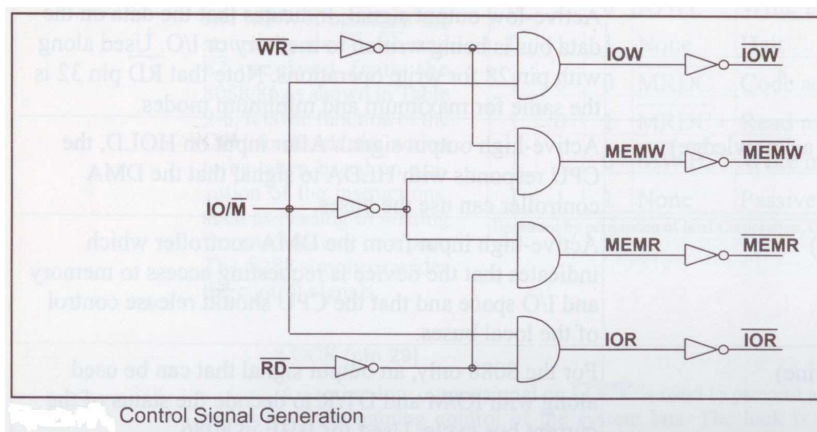


FIGURE 2.7 – Signaux de contrôle du 8088

Il existe plusieurs façons de réaliser ce circuit : la figure 2.8 montre comment le réaliser en utilisant des 74LS32 et des 74LS04.

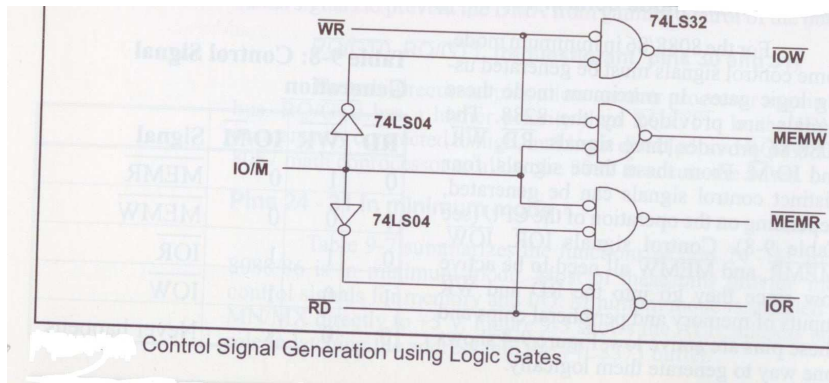


FIGURE 2.8 – Réalisation des signaux de contrôle du 8088

En conclusion, le principe de constitution du système de bus du microprocesseur 8088 en mode minimum est montré à la figure 2.9. Les lignes électriques partant du microprocesseur pour obtenir ce bus système, qui constituent elles aussi un bus, constituent le **bus local** (par opposition au bus système).

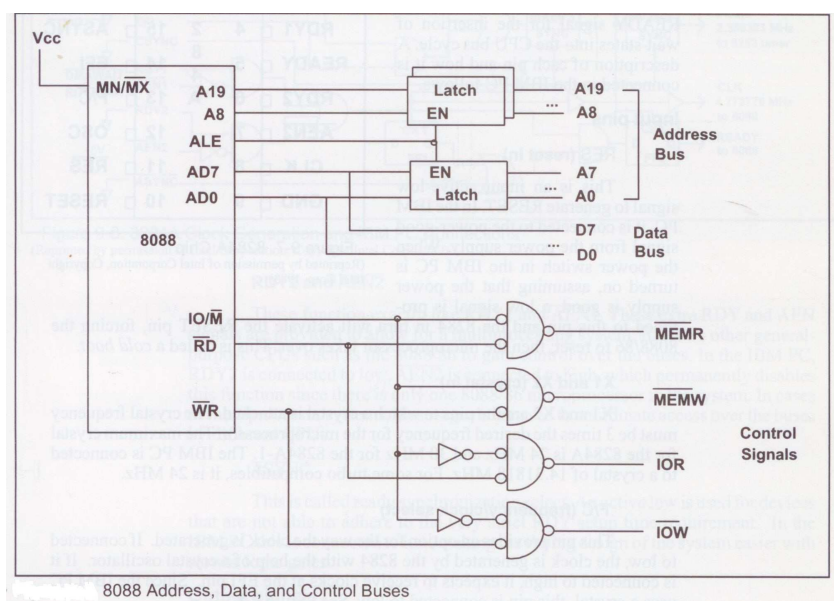


FIGURE 2.9 – Réalisation des bus avec le 8088 en mode minimum

2.3.3 Contrôleur de bus 8288

Dans le cas du 8088/86, on peut utiliser un contrôleur de bus spécialement conçu par *Intel* pour fournir les signaux voulus. Le 8288 est un circuit intégré à 20 broches comme le montre la figure 2.10.

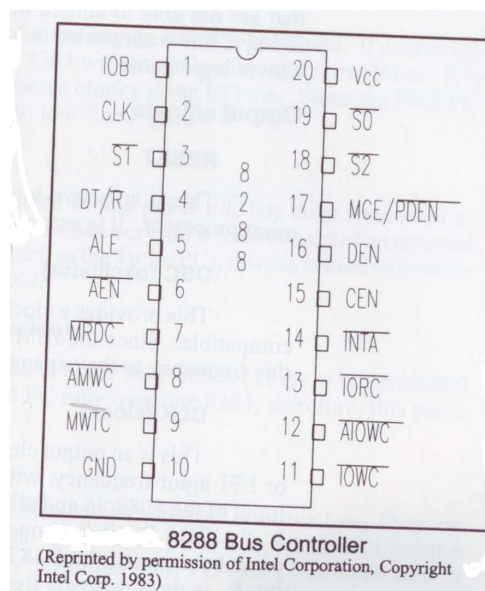


FIGURE 2.10 – Brochage du 8288

Signaux d'entrée.- Les signaux d'entrée sont les suivants :

— $\overline{S0}$, $\overline{S1}$, $\overline{S2}$ (**entrée de statut**)

Ces signaux correspondent aux broches 19, 3 et 18 respectivement. Les entrées proviennent directement du microprocesseur. Suivant les données du microprocesseur, le contrôleur fournit le signal de contrôle indiqué dans le tableau suivant :

$\overline{S2}$	$\overline{S1}$	$\overline{S0}$	État du microprocesseur	Commandes du 8288
0	0	0	Interrupt acknowledge	\overline{INTA}
0	0	1	Read input/output port	\overline{IORC}
0	1	0	Write input/output port	\overline{IOWC} , \overline{AIOWC}
0	1	1	Halt	Aucun
1	0	0	Code access	\overline{MRDC}
1	0	1	Read memory	\overline{MRDC}
1	1	0	Write memory	\overline{MWTC} , \overline{AMWC}
1	1	1	Passive	Aucun

— **CLK (clock)**

La broche 2 permet de synchroniser tous les signaux et toutes les commandes avec le microprocesseur.

- $\overline{\text{AEN}}$ (**Address ENable**)
Le signal AEN, actif à tension basse, à la broche 6, active les sorties du 8288 au moins 115 ns après son déclenchement.
Sur l'IBM-PC, cette broche est reliée à la sortie AEN du microprocesseur.
- **CEN (Command ENable)**
Un signal, actif à tension haute, à la broche 15 active les signaux de commande et DEN.
Sur l'IBM-PC, cette broche est aussi reliée à la sortie AEN du microprocesseur.
- **IOB (Input/Output Bus mode)**
Un signal, actif à tension haute, à la broche 1 permet d'opérer dans le mode du bus d'entrée-sortie plutôt que dans le mode bus système.
Puisque l'IBM-PC est conçu avec des bus système, la broche est reliée à une tension basse.

Signaux de sortie.- Les signaux de sortie sont les suivants :

- $\overline{\text{MRDC}}$ (**Memory ReaD Command**)
La broche 7 à tension basse fournit le signal $\overline{\text{MEMR}}$ (*MEMory Read*).
- $\overline{\text{MWTC}}$ (**Memory WriTe Command**), $\overline{\text{AMWC}}$ (**Advanced Memory Write Command**)
Ces deux signaux, correspondants respectivement aux broches 9 et 8, sont actifs à tension basse. Ils sont utilisés pour dire à la mémoire d'enregistrer la donnée présente sur le bus. C'est la même chose que le signal $\overline{\text{MEMW}}$ (*MEMory Write*), la seule différence étant que le signal $\overline{\text{AMWC}}$ est activé légèrement plus tôt afin de laisser du temps aux périphériques lents.
Sur l'IBM-PC, seul le signal $\overline{\text{MWTC}}$ est utilisé comme contrôle de signal $\overline{\text{MEMW}}$; le signal $\overline{\text{AMWC}}$ n'est pas utilisé.
- $\overline{\text{IORC}}$ (**I/O Read Command**)
Ce signal, actif à tension basse et correspondant à la broche 13, dit au périphérique de déposer sa donnée sur le bus.
Sur l'IBM-PC, il sert de signal de contrôle $\overline{\text{IOR}}$ (*I/O Read*).
- $\overline{\text{IOWC}}$ (**I/O Write Command**), $\overline{\text{AIOWC}}$ (**Advanced I/O Write Command**)
Ces deux signaux, correspondants respectivement aux broches 11 et 12 et actifs à tension basse, sont utilisés pour dire au périphérique de récupérer la donnée se trouvant sur le bus. Le signal $\overline{\text{AIOWC}}$ est disponible légèrement plus tôt pour prévenir suffisamment longtemps à l'avance les périphériques lents.
Sur l'IBM-PC, $\overline{\text{IOWC}}$ correspond au signal $\overline{\text{IOW}}$.
- $\overline{\text{INTA}}$ (**INTerrupt Acknowledge**)
Ce signal, correspondant à la broche 14 et actif à tension basse, informe le périphérique ayant demandé une interruption que celle-ci est prise en compte et il fournit l'adresse du vecteur d'interruption correspondant dans le bus des adresses.
Sur l'IBM-PC, cette broche est reliée à la broche INTA du contrôleur d'interruption 8259.

— **DT/ \overline{R} (Data Transmit/Receive)**

Ce signal, correspondant à la broche 4, est utilisé pour contrôler le sens des données du 8088/86.

Sur l'IBM-PC, cette broche est reliée à la broche DIR d'un 74LS245. Lorsque le 8086/88 écrit une donnée, ce signal est à tension haute et cela permet aux données d'aller du côté A au côté B du 74LS245. Lorsque le microprocesseur lit une donnée, ce signal a une tension basse, ce qui permet à la donnée de passer du côté B au côté A du 74LS245.

— **DEN (Data ENable)**

Ce signal, actif à tension haute à la broche 16, transforme le bus des données soit en bus des données local, soit en bus des données système.

Sur l'IBM-PC, il est utilisé avec un signal du contrôleur d'interruption 8259 pour activer la broche G du transmetteur 74LS245.

— **MCE/PDEN (Master Cascade Enable/Peripheral Data Enable)**

Ce signal, correspondant à la broche 17, est utilisé en lien avec le contrôleur d'interruption 8259 en configuration maître.

Sur l'IBM-PC, le 8259 est utilisé en mode esclave, aussi cette broche est-elle ignorée.

— **ALE (Address Latch Enable)**

Le signal ALE, correspondant à la broche 5 actif à tension haute, est utilisé pour activer les bascules d'adresse puisque le microprocesseur 8088 multiplexe les adresses et les données AD0 à AD7 pour minimiser le nombre de broches.

Sur l'IBM-PC, ALE est relié à l'entrée G d'un 74LS373, rendant possible le démultiplexage des adresses.

2.3.4 Cas de l'IBM PC

2.3.4.1 Implémentation des trois bus du PC

L'IBM PC est un ordinateur à trois bus : le bus des adresses, le bus des données et le bus de contrôle.

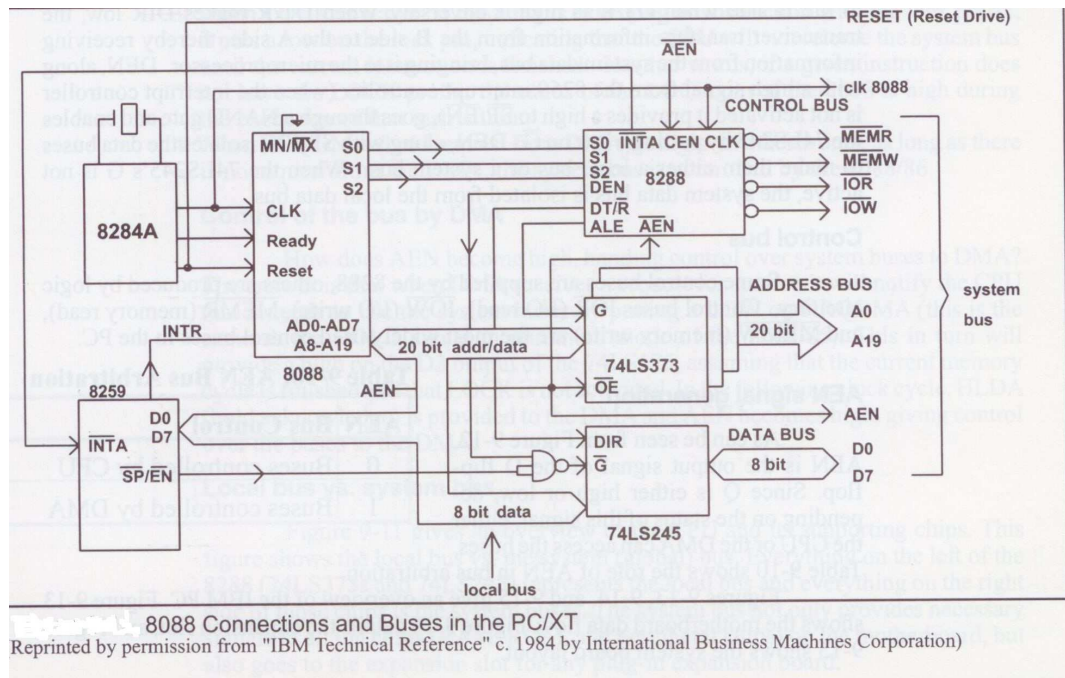


FIGURE 2.11 – Les bus sur l'IBM PC

Bus local et bus système.- La figure 2.11 montre les connexions entre le microprocesseur et ses coupleurs. Tout ce qui est à gauche du 8288 (à savoir les 74LS373 et le 74LS245) représente les **bus locaux** et tout ce qui est à droite les **bus système**. Les bus système fournissent les signaux nécessaires à tous les circuits (RAM, ROM et circuits intégrés de périphériques) de la carte mère mais aussi aux slots d'extension.

Bus des adresses.- Comme le montre la figure 2.11, trois circuits intégrés 74LS373 (voir la description à la figure 2.12) sont utilisés pour réaliser deux fonctions :

- 1°) Pour obtenir une adresse stable sur tout l'ordinateur à partir d'une adresse fournie par le microprocesseur. Le bus des adresses est unidirectionnel.

Les circuits intégrés 74LS373 sont activés par les signaux de contrôle AEN et ALE. Lorsque la broche AEN du 8088, reliée à \overline{OE} (*Output Enable*) du 74LS373, est au niveau bas, le 8088 indique par là qu'il fournit une adresse au système. Le 8288 émet alors un signal sur sa broche ALE (reliée à la broche G du 74LS373), permettant au 74LS373 de relever l'adresse fournie par le microprocesseur et de la présenter de façon stable à la mémoire, aux périphériques et aux slots d'extension.

Le démultiplexage des adresses A0 – A7 est effectué par le 74LS373 relié aux broches AD0 – AD7 du microprocesseur. Les broches A8 – A15 sont reliées à un second 74LS373 et A16 – A19 à un troisième. Une moitié du troisième 74LS373 n'est donc pas utilisée.

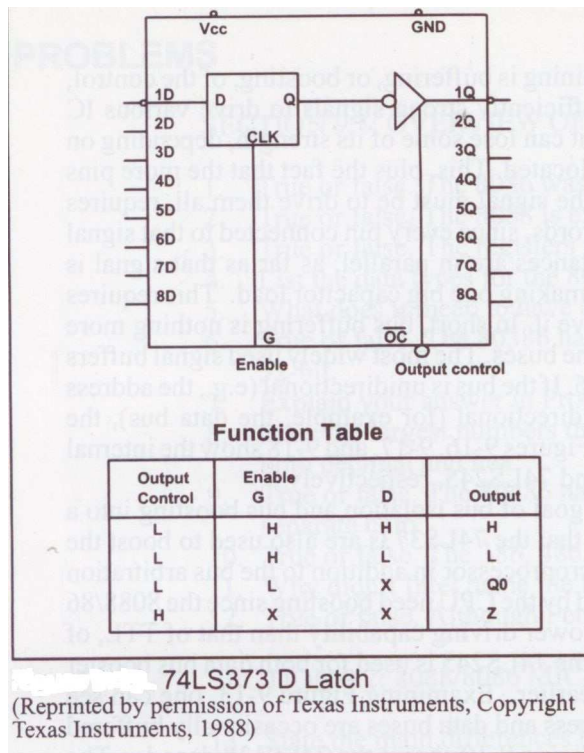


FIGURE 2.12 – La bascule 74LS373

- 2°) Pour isoler les bus des adresses du bus local des adresses. En effet le bus système peut être utilisé par le microprocesseur mais aussi par le DMAC. Ceci est effectué par le 74LS373 à travers AEN.

Bus des données.- Le bus bidirectionnel des données passe à travers le transcepteur 74LS245 (voir la description à la figure 2.13). Les broches DT/ \overline{R} et DEN du 8288 fournissent les deux signaux qui activent le 74LS245. La broche DT/ \overline{R} , reliée à la broche DIR du 74LS245, active, à niveau haut, le passage de l'information du côté A vers le côté B du transcepteur. Réciproquement, à niveau bas, le transcepteur transfère l'information du côté B vers le côté A, permettant ainsi au microprocesseur de recevoir de l'information.

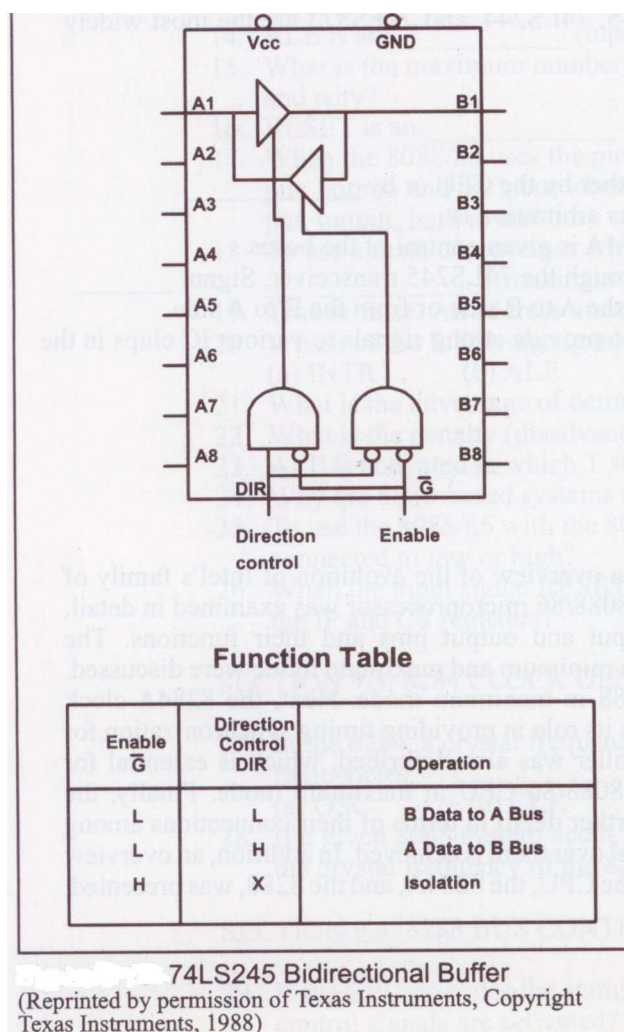


FIGURE 2.13 – Le tampon bidirectionnel 74LS245

La broche DEN du 8288 est combinée à un signal à niveau haut du contrôleur d'interruption 8259 dans une porte ET pour activer le 74LS245 en fournissant un niveau bas à la broche \overline{G} . La broche DEN isole donc les bus des données pour en faire soit un bus local, soit un bus système. Lorsque la broche G du 74LS243 n'est pas active, le bus système des données est isolé du bus local des données.

Bus de contrôle.- Certains bus de contrôle sont fournis par le 8288, d'autres par la circuiterie logique. Les bus de contrôle $\overline{\text{IOR}}$ (I/O read), $\overline{\text{IOW}}$ (I/O write), $\overline{\text{MEMR}}$ (*memory read*) et $\overline{\text{MEMW}}$ (*memory write*) sont les bus de contrôle les plus utilisés du PC.

2.3.4.2 Arbitrage entre microprocesseur et DMA

Il y a deux maîtres qui peuvent accéder à ces bus : le microprocesseur 8088 et le DMAC 8237. Puisqu'aucun bus ne peut servir deux maîtres à la fois, il faut mettre en place une façon de permettre à l'un des deux, le microprocesseur ou le DMAC, de contrôler les bus. C'est ce qu'on appelle l'**arbitrage** (*arbitration* en anglais), effectué par le circuit de génération de AEN.

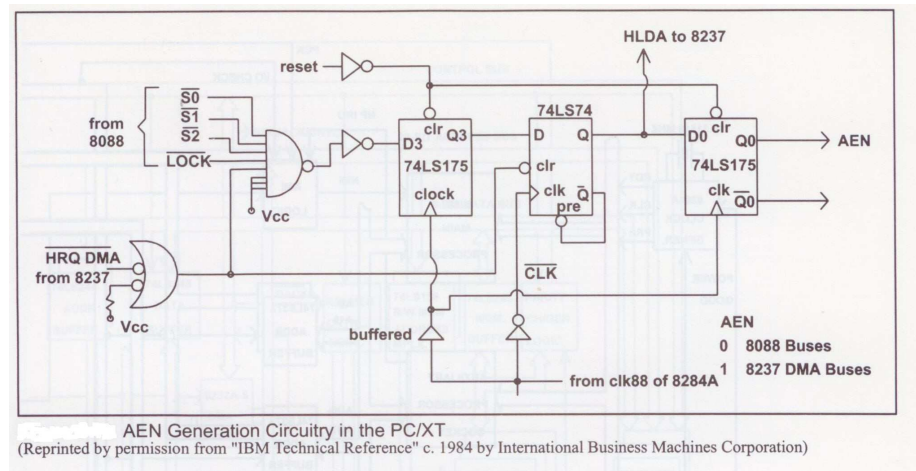


FIGURE 2.14 – Engendrement du signal AEN

Engendrement du signal AEN.- Comme on le voit à la figure 2.14, AEN est le signal de sortie de la bascule D. Puisque Q est au niveau haut ou bas selon la valeur de ce signal, le microprocesseur ou le DMAC peuvent accéder aux bus conformément à l'arbitrage suivant :

AEN	Contrôle des bus
0	Bus contrôlés par le microprocesseur
1	Bus contrôlés par le DMAC

Contrôle des bus par le microprocesseur.- Le microprocesseur contrôle les bus système lorsque AEN est à niveau bas. Ceci arrive dans les situations suivantes :

- Lorsque l'ordinateur est en phase de réinitialisation RESET (on appuie sur le bouton d'alimentation pour un démarrage à froid). RESET est un signal actif au niveau haut, celui-ci étant alors inversé pour fournir un signal de niveau bas, signal CLEAR nécessaire du 74LS175.
- Si un des signaux de statut est au niveau bas. Comme nous l'avons vu à propos des signaux du 8288, ceci arrive presque toujours sauf lorsque le microprocesseur est en mode passif. La figure 2.15 montre que le cycle mémoire du 8088 comprend quatre tics d'horloge : T_1 , T_2 , T_3 et T_4 . Les signaux de statut sont tous à niveau haut (passifs) dans l'état T_4 , donnant une chance à une requête DMA en attente d'utiliser les bus.

- Lorsque LOCK est actif. Si le 8088 exécute une instruction ayant le préfixe LOCK, elle place un niveau bas sur la broche LOCK, indiquant par là que le microprocesseur ne libèrera pas le bus système avant que l'instruction ne soit terminée.
- Lorsque $\overline{\text{HRQDMA}}$ (*Hold ReQuest from DMA*) est à niveau haut, autrement dit tant qu'il n'y a pas de requête DMA.

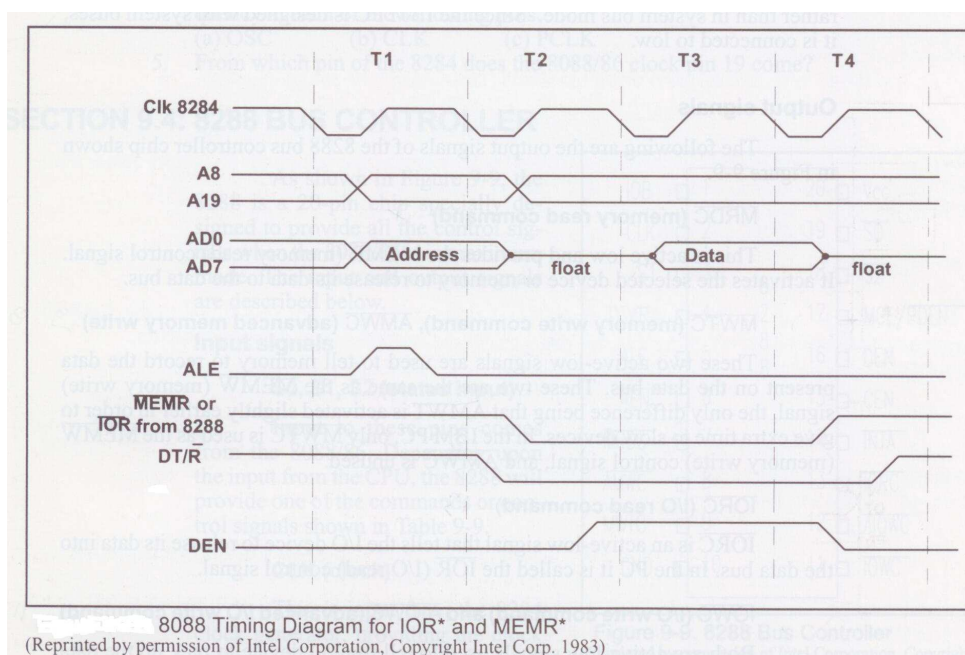


FIGURE 2.15 – Diagramme temporel du 8088

Contrôle des bus par le DMAC.- Lorsque le DMAC reçoit une requête de service, il notifie au microprocesseur qu'il a besoin d'utiliser les bus système en plaçant un niveau bas sur sa broche HRQDMA. Le microprocesseur fournit un niveau haut sur la sortie D3 du 74LS175 lorsque le cycle mémoire en cours est terminé et que LOCK n'est pas activé. Lors du cycle suivant, HLDA (*HoLD Acknowledge*) est fourni au DMAC et AEN se place à niveau haut, donnant ainsi le contrôle des bus au DMAC.

2.3.4.3 Tampons pour les bus

Les signaux des bus (contrôle, adresse et donnée) ont besoin d'être renforcé pour pouvoir piloter les divers coupleurs. On parle de **tampon** (*bus buffering* ou *boosting* en anglais).

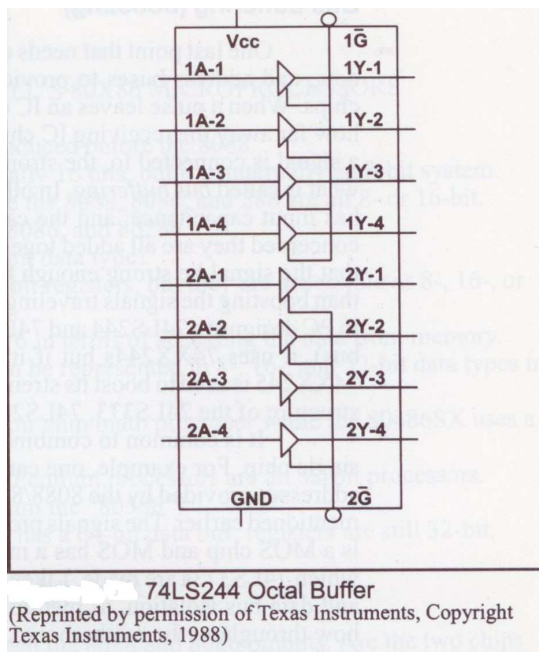


FIGURE 2.16 – Le tampon octal 74LS244

En effet lorsqu'une impulsion quitte un circuit intégré, elle peut perdre de sa force lorsqu'elle arrive loin de celui-ci. De plus une broche peut être reliée à plusieurs broches. Les tampons les plus utilisés sur le PC sont le 74LS244 (figure 2.16) et le 74LS245 (figure 2.13), suivant que le bus est unidirectionnel ou bidirectionnel.

On combine parfois les deux fonctions de tampon et d'isolation, comme dans le cas du 74LS373 (figure 2.12).

2.4 Commentaire du BIOS : vérifications au démarrage

Ce que nous avons vu dans ce chapitre est intéressant pour comprendre un peu mieux la carte mère mais rien n'est à programmer. Le BIOS ne s'en préoccupe que pour effectuer quelques vérifications au démarrage.

Voici le début de RESET :

```

301
302 ;-----
303 ;           8088 PROCESSOR TEST           :
304 ; DESCRIPTION                               :
305 ;           VERIFY 8088 FLAGS, REGISTERS :
306 ;           AND CONDITIONAL JUMPS       :
307 ;-----
308           ASSUME  CS:CODE,DS:NOTHING,ES:NOTHING,SS:NOTHING
E05B        309           ORG      OE05BH
E05B        310  RESET  LABEL  FAR
E05B FA     311  START: CLI                ; DISABLE INTERRUPTS
E05C B4D5   312           MOV      AH,OD5H          ; SET SF, CF, ZF, AND AF FLAGS ON
E05E 9E     313           SAHF
E05F 734C   314           JNC      ERRO1           ; GO TO ERR ROUTINE IF CF NOT SET
E061 754A   315           JNZ      ERRO1           ; GO TO ERR ROUTINE IF ZF NOT SET
E063 7B48   316           JNP      ERRO1           ; GO TO ERR ROUTINE IF PF NOT SET
E065 7946   317           JNS      ERRO1           ; GO TO ERR ROUTINE IF SF NOT SET
E067 9F     318           LAHF          ; LOAD FLAG IMAGE TO AH
E068 B105   319           MOV      CL,5             ; LOAD CNT REG WITH SHIFT CNT
E06A D2EC   320           SHR      AH,CL           ; SHIFT AF INTO CARRY BIT POS
E06C 733F   321           JNC      ERRO1           ; GO TO ERR ROUTINE IF AF NOT SET
E06E B040   322           MOV      AL,40H          ; SET THE OF FLAG ON
E070 D0E0   323           SHL      AL,1             ; SETUP FOR TESTING
E072 7139   324           JNO      ERRO1           ; GO TO ERR ROUTINE IF OF NOT SET
E074 32E4   325           XOR      AH,AH           ; SET AH + 0
E076 9E     326           SAHF          ; CLEAR SF, CF, ZF, ANF PF
E077 7634   327           JBE      ERRO1           ; GO TO ERR ROUTINE IF CF ON
328           ; GO TO ERR ROUTINE IF ZF ON
E079 7832   329           JS       ERRO1           ; GO TO ERR ROUTINE IF SF ON
E07B 7A30   330           JP       ERRO1           ; GO TO ERR ROUTINE IF PF ON
E07D 9F     331           LAHF          ; LOAD FLAG IMAGE TO AH
E07E B105   332           MOV      CL,5             ; LOAD CNT REG WITH SHIFT CNT
E080 02EC   333           SHR      AH,CL           ; SHIFT AF' INTO CARRY BIT POS
E082 7229   334           JC       ERRO1           ; GO TO ERR ROUTINE IF ON
E084 DOE4   335           SHL      AH,1             ; CHECK THAT OF' IS CLEAR
E086 7025   336           JO       ERRO1           ; GO TO ERR ROUTINE IF ON
337
338 ;----- READ/WRITE THE 8088 GENERAL AND SEGMENTATION REGISTERS
339 ;           WITH ALL ONE'S AND ZEROES'S.
340
E088 B8FFFF 341           MOV      AX,OFFF5H          ; SETUP ONE'S PATTERN IN AX
E08B F9     342           STC
E08C 8ED8   343  C8:    MOV      DS,AX                ; WRITE PATTERN TO ALL REGS
E08E 8CD8   344           MOV      BX,DS
E090 8EC3   345           MOV      ES,BX
E092 8CC1   346           MOV      CX,ES
E094 8ED1   347           MOV      SS,CX
E096 8CD2   348           MOV      DX,SS
E098 8BE2   349           MOV      SP,DX
E09A 8BEC   350           MOV      BP,SP
E09C 88F5   351           MOV      SI,BP
E09E 88FE   352           MOV      DI,SI
EOA0 7307   353           JNC      C9                    ; TST1A
EOA2 33C7   354           XOR      AX,DI                ; PATTERN MAKE IT THRU ALL REGS
EOA4 7507   355           JNZ      ERRO1                ; NO - GO TO ERR ROUTINE
EOA6 F8     356           CLC
EOA7 EBE3   357           JMP      C8
EOA9        358  C9:    ; TST1A
EOA9 DBC7   359           OR       AX,DI                ; ZERO PATTERN MAKE IT THRU
EOAB 7401   360           JZ       C10                 ; YES - GO TO NEXT TEST
EOAD F4     361  ERRO1: HLT                ; HALT SYSTEM

```

Commentaires.- 1^o) Le segment de code utilisé est CODE (ligne 308). Les segments de données, de pile et supplémentaires ne sont pas importants (ligne 308).

Le segment de code de la ROM résidente est défini à partir de la ligne 230 :

```

                230 ;-----
                231 ;      ROM RESIDENT CODE      :
                232 ;-----
-----        233 CODE      SEGMENT AT 0F000H
0000 (57344    234          DB      57344 DUP(?)      ; FILL LOWEST 56K
      ??
      )
E000 31353031353132    235
      20434F50522E20    236          DB      '1501512 CPR. IBM 1981'      ; COPYRIGHT NOTICE
      49424D20313938
      32
                237

```

- 2^o) Le début du code est placé à l'adresse absolue E05Bh (ligne 309) pour correspondre à l'emplacement où on a été renvoyé par le saut inconditionnel.

- 3^o) Les interruptions masquables sont désactivées (ligne 311). On essaie d'initialiser les indicateurs SF, CF, ZF et AF (lignes 312 et 313) et on les teste (lignes 314 à 317). Si l'un d'eux ne se comporte pas comme voulu, on arrête purement et simplement le système (ligne 361).

- 4^o) Si tout s'est bien passé, on place le contenu du registre des indicateurs dans AH (ligne 318).

On place la position de l'indicateur AF dans le registre CL (ligne 319), on effectue un décalage avec CF (ligne 320). Si l'indicateur AF n'est pas positionné, on arrête le système (ligne 321).

On positionne l'indicateur OF en effectuant un décalage (lignes 322 et 323). De même, on arrête le système s'il n'est pas positionné (ligne 324).

- 5^o) On essaie d'effacer tous les indicateurs (lignes 325 et 326). Si ce n'est pas le cas pour l'un d'eux, on arrête le système (lignes 327 à 330).

Si tout s'est bien passé, on recommence à l'envers ce que l'on a fait au 3^o), c'est-à-dire que l'on arrête le système si l'un d'entre eux n'a pas la valeur 0 (lignes 331 à 336).

2.5 La carte mère

Nous avons vu le composant essentiel d'un micro-ordinateur, à savoir le microprocesseur. Nous avons vu également les composants auxiliaires que sont le système de bus, la mémoire vive et les interfaces d'entrées/sorties. Nous avons déjà dit que ces divers composants se placent sur la *carte mère*. Maintenant que nous avons vu un peu plus en détail ces divers composants, voyons le principe de réalisation d'une carte mère, en prenant comme exemple la carte mère du premier IBM-PC.

2.6 Bibliographie

[Zdany] [Introduction générale.]

[Tanenbaum] [Introduction générale.]

[Uff-87] UFFENBECK, John, **The 80x86 Family : Design, Programming, and Interfacing**, 1987, 2nd ed. 1998, Prentice-Hall, ix+ 566 p. + diskette.

[Comme son nom l'indique, il s'agit d'une introduction aux microprocesseurs et aux micro-ordinateurs. Dans une première partie est décrite l'architecture des microprocesseurs 8086, 8088, 80386, 80486, Pentium et Pentium Pro (chapitre 3), ainsi que la façon de les programmer à travers DEBUG (chapitres 4 et 5) et MASM (chapitre 6), donc en utilisant les interruptions.

La seconde partie est une description des modules de mémoire et de leur interfacement (chapitre 7), des techniques d'entrées-sorties parallèles, programmées et en utilisant un PPI (chapitre 8), pilotées par interruption et avec accès direct à la mémoire (chapitre 9), séries en insistant sur le cas des modems (chapitre 10), et enfin sur les systèmes de bus du PC (ISA, Microchannel, EISA, VESA, PCI, SCSI et USB) au chapitre 11. Le chapitre 11 donne aussi l'architecture du premier IBM-PC.

L'ampleur du sujet montre que tout ne peut pas être abordé en détail, et que les concepts généraux ne sont pas toujours bien dégagés. Pour ces derniers on peut se référer à l'autre livre de John Uffenbeck cité ci-dessus. La première partie n'est pas la plus intéressante (on la trouve traitée ailleurs, de façon plus structurée). C'est bien sûr l'architecture du PC qui est ici intéressante, expliquée précédemment par quelques cas (interfaçage d'un élément de mémoire,...).]

[Bre-97] BREY, Barry B., **The Intel Microprocessors 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, and Pentium Pro Processor : Architecture, Programming, and Interfacing**, Prentice-Hall, Fourth Edition, 1997, xv + 907 p.

[Maz-95] MAZIDI, Muhammad Ali and MAZIDI, Janice Gillipsie, **The 80x86 IBM PC and Compatible Computers : Assembly Language, Design, and Interfacing**, Prentice Hall, 1995. Second edition, 1998, xxxviii + 984 p.

[IBM-83] **IBM Personal Computer XT, Hardware Reference Library, Technical Reference**, Revised Edition, April 1983. Version électronique disponible à l'adresse :

<http://www.retroarchive.org/dos/docs/ibm5160techref.pdf>