

Le Microprocesseur 8086

I. Brochage du 8086

Le microprocesseur Intel 8086 est un microprocesseur CISC 16 bits, fabriqué par la firme Intel. C'est le premier microprocesseur de la famille Intel 80x86 et le plus répandu dans le domaine informatique. Il se présente sous la forme d'un boîtier DIP (Dual In-line Package) à 40 broches, figure 1.

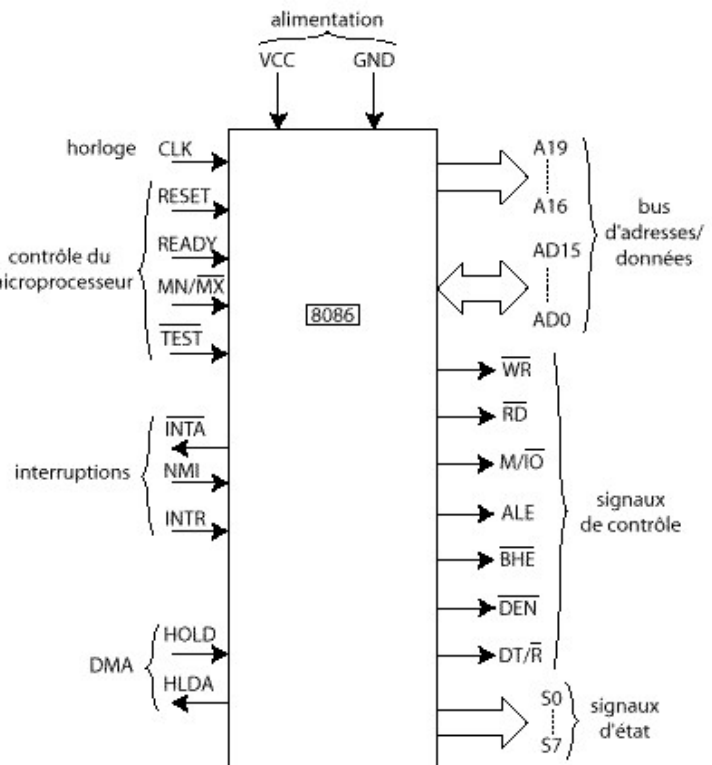
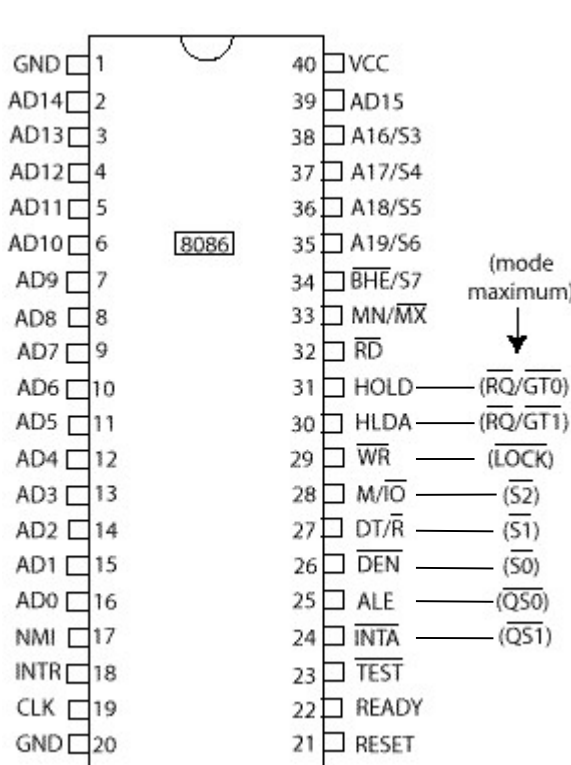


Figure 1: brochage du CPU 8086

Figure 2: Schéma fonctionnel du CPU 8086 .

II. Description des PINs

La figure 2 présente le schéma fonctionnel du CPU 8086 ou les **Vcc et GND** assure l'alimentation électrique du microprocesseur.

CLK : entrée destinée à recevoir le signal de l'horloge système, qui cadence le fonctionnement du microprocesseur. Ce signal provient d'un générateur d'horloge : le 8284 (voire sa description sur la figure 3 et sa configuration avec le CPU 8086 sur la figure 4).

READY : permet la synchronisation des mémoires et périphériques lents avec le CPU 8086. Il est élaboré à partir du RDY qui sont validé par AEN.

RESET: un signal de remise à l'état initiale est généré à partir d'un signal externe RES qui se synchronise avec l'horloge CLK. Ce signal doit rester à l'état haut pendant au moins

quatre cycles d'horloge, pour permettre au CPU 8086 de s'initialiser correctement, en commençant l'exécution à partir de l'adresse FFFF:0000.

MN/MX: Sélectionne entre l'un des deux modes de fonctionnement du 8086.

- Mode minimum: le CPU 8086 fonctionne d'une manière autonome et en monoprocesseur et il génère par lui même les signaux de bus de commande.
- Mode maximum : Ces signaux sont produit par un contrôleur de bus 8288, ce qui lui permet d'opérer dans un environnement multiprocesseur.

TEST : entrée de synchronisation entre le CPU 8086 et le coprocesseur

NMI, INTR: entrées de demande d'interruption. INTR: interruption normale, NMI (Non Masquable Interrupt) : interruption prioritaire.

INTA (Interrupt Acknowledge) : indique que le microprocesseur a pris en compte l'interruption.

HOLD: entrée de demande d'accès au bus.

HLDA: indique que le microprocesseur a pris en compte la demande d'accès au bus.

S0 à S7: Signaux d'état indiquant le type d'opération sur le bus.

A16/S3 à A19/S6: 4 bits de poids fort du bus d'adresses, multiplexés avec 4 bits d'état.

AD0 à AD15: 16 bits de poids faible du bus d'adresses, multiplexés avec 16 bits de données, d'où la nécessité d'un multiplexage pour obtenir séparément les bits d'adresse et de données.

Compléter se qui suit :

Le microprocesseur 8086 possède :

.....bits de données => c'est un microprocesseurbits;

.....bits d'adresses => l'espace adressable par le 8086 est :Octets

RD (Read): signal de lecture d'une donnée.

WR (Write): Signal d'écriture d'une donnée.

M/IO (Memory /Input-Output): indique que le CPU 8086 adresse la mémoire ou les unités d'entrées/sortie.

DEN(Data ENable): indique que la donnée est disponible sur le bus de données.

ALE (Adress Latch Enable) : indique que l'adresse est disponible sur le bus d'adresses.

DT/R (Data Transmit/ Receive): indique le sens de transfert des données.

BHE (Bus High Enable): Signal de validation de l'octet du poids fort du bus de données.

III. Démultiplexage du bus A/D

Le démultiplexage des signaux AD0 à AD15 (ou A16/S3 à A19/S6) se fait en mémorisant l'adresse lorsque celle-ci est présente sur le bus A/D, à l'aide de circuits **verrous** (latch circuit) 74373, formé de bascules D. La commande de mémorisation de l'adresse est générée par le microprocesseur : c'est le signal **ALE**.

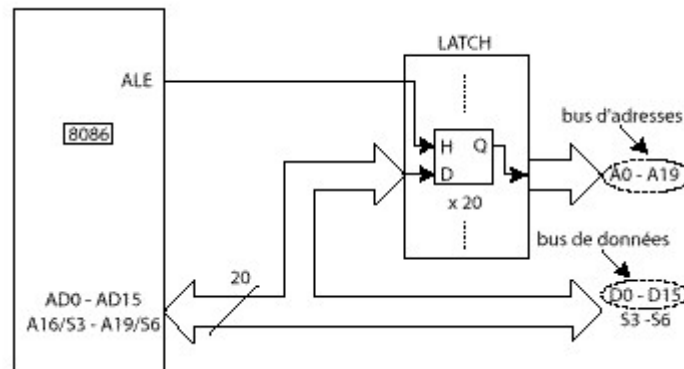


Figure 5: Principe de Circuit de démultiplexage A/D

si $ALE = 1$, le verrou est transparent ($Q = D$) ;

si $ALE = 0$, mémorisation de la dernière valeur de D sur les sorties Q;

IV. Interfaçage de la mémoire

Le CPU 8086 possède un bus d'adresses de 20 lignes, permettant d'adresser un espace de 1 Moctets, organisé en deux banques de 512 Ko :

- la banque **inférieure** (ou **paire**);
 - la banque **supérieure** (ou **impaire**).
- Ces deux banques, figure 6, sont sélectionnées par :
- A0 pour la banque paire qui contient les octets de poids faible;
 - BHE pour la banque impaire qui contient les octets de poids fort.

Seuls les bits A1 à A19 servent à désigner une case mémoire dans chaque banque de 512 Ko. Le microprocesseur peut ainsi lire et écrire des données sur 8 bits ou sur 16 bits selon A0 et BHE, tableau 1.

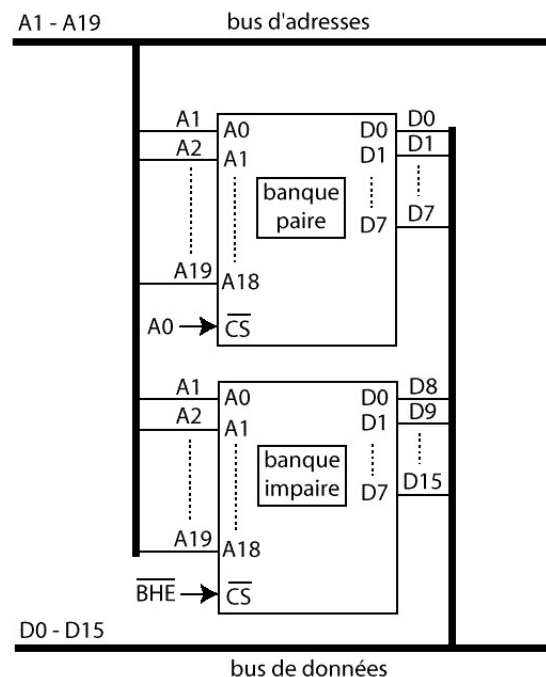


Figure 6: Organisation de la mémoire

$\overline{\text{BHE}}$	A0	Octets transférés
0	0	Les deux octets (mot complet)
0	1	Octet de poids fort (adresse impaire)
1	0	Octet de poids faible (adresse paire)
1	1	Aucun octet

Tableau 1.

Remarque :

le 8086 ne peut lire une donnée sur 16 bits en une seule fois, uniquement si l'octet de poids fort de cette donnée est rangé à une adresse impaire et l'octet de poids faible à une adresse paire (alignement sur les adresses paires), sinon la lecture de cette donnée doit se faire en deux opérations successives, d'où une augmentation du temps d'exécution du transfert dû à un mauvais alignement des données.

V. Organisation interne du CPU 8086

Le CPU 8086 est constitué de deux unités, figure 7, fonctionnant en parallèle :

- l'**unité d'exécution** (EU : Execution Unit) ;
- l'**unité d'interface de bus** (BIU : Bus Interface Unit).

L'unité d'exécution (EU) exécute les instructions arithmétique ou logique contenues dans la file d'attente.

L'unité d'interface de bus (BIU) recherche les instructions en mémoire et stocke par anticipation 6 octets instructions dans une **file d'attente**.

Les deux unités fonctionnent simultanément, ainsi, pendant que l'unité d'exécution traite une instruction, la BIU recherche dans la mémoire les octets composant la prochaine instruction à exécuter. Ce mode de fonctionnement, dit pipeline, accélère le processus d'exécution d'un programme, en réduisant le temps mort de l'exploitation du système de bus.

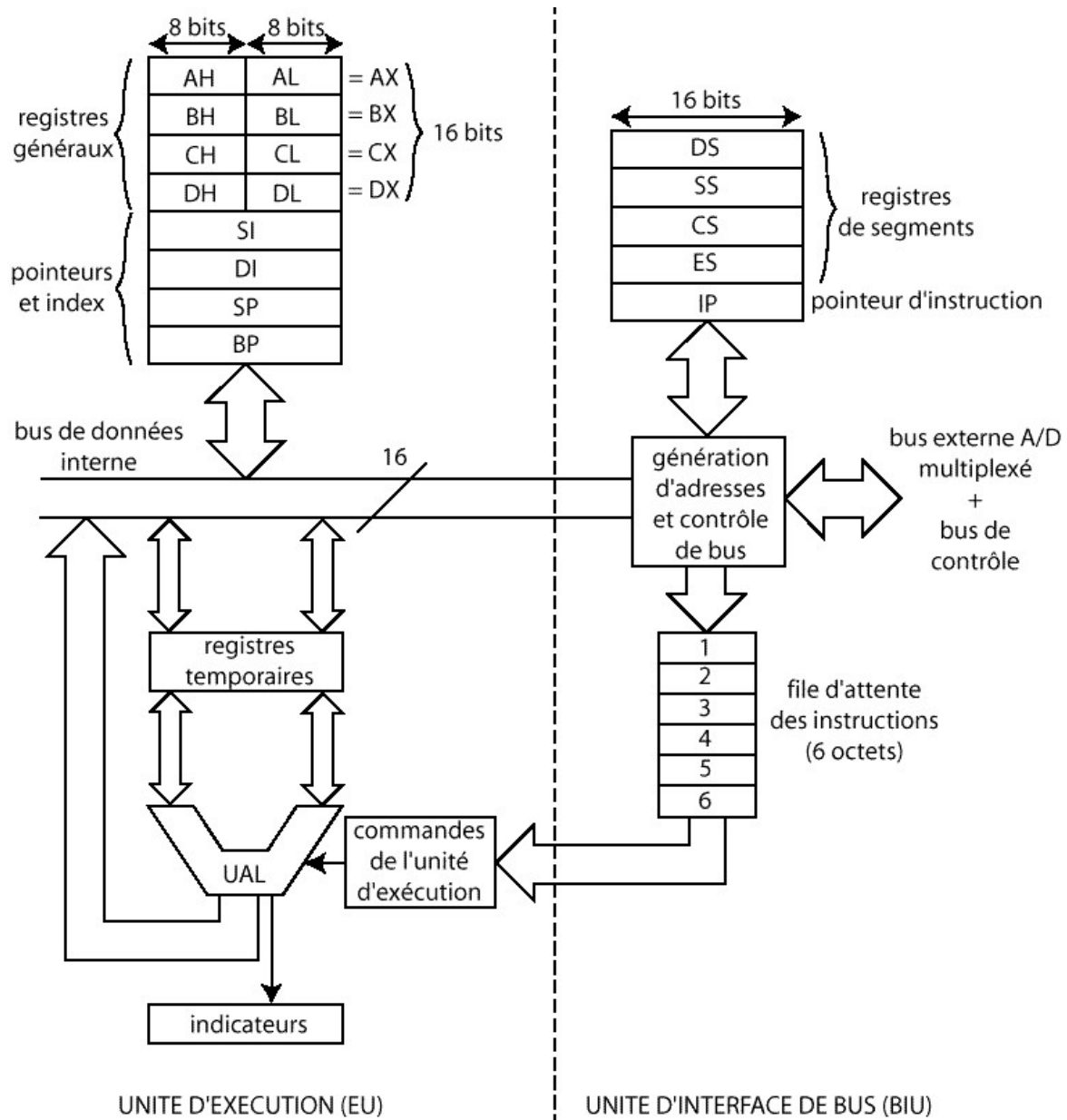


Figure 7: L'organisation interne du CPU 8086

V.1. Les registre du CPU 8086

Le CPU 8086 contient 14 registres répartis en 4 groupes :

V.1.1. Registres généraux : 4 registres sur 16 bits.

AX=(AH,AL);

BX=(BH,BL);

CX=(CH,CL);

DX=(DH,DL).

Ils peuvent être également subdivisés en deux registres de 8 bits supérieurs et inférieurs, et être référencés sous **xH** et **xL**: par exemple AH et AL, pour AX. En d'autre termes $AX=256*AH+AL$.

Ils servent à contenir temporairement des données. Ce sont des registres généraux mais ils peuvent être utilisés pour des opérations particulières.

AX comme accumulateur ;

BX comme registre de base pour l'adressage basé ;

CX comme compteur ;

DX sert aussi pour quelques opérations mathématiques.

V.1.2. Registres de pointeurs et d'index (déplacement): 4 registres sur 16 bits.

a) Pointeurs :

- **SP** : (Stack Pointer), pointeur de pile, pointe sur le sommet de la pile.

- **BP** : (Base Pointer), pointeur de base, pointe sur l'adresse de base de la pile

b) Index :

- **SI** : Source Index ;

- **DI** : Destination Index.

Ils sont utilisés pour les transferts de chaînes d'octets entre deux zones mémoire.

Les registres pointeurs et d'index contiennent des adresses de cases mémoire et sont appelés aussi registre de déplacement .

V.1.3. Registre pointeur d'instruction et registre d'état : 2 Registre sur 16 bits.

a) **Registre pointeur d'instruction** (IP : Instruction Pointer): contient l'adresse de la prochaine instruction à exécuter.

b) **Registre d'état** (PSW : Program Status Word)

La valeur représentée par ce nombre de 16 bits n'a aucune signification en tant qu'ensemble: ce registre est manipulé bit par bit. Il offre 16 bits dont seulement 9 sont utilisés:

R	R	R	R	O	D	I	T	S	Z	U	A	U	P	U	C
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Figure 8: Registre d'état (PSW)

a) Indicateurs d'état:

CF : (carry flag), indicateur de retenue. Mis à 1 si un calcul produit une retenue ;

PF : (parity flag), indicateur de parité. Mis à 1 si les 4 bits de poids faible du résultat contiennent un ;

AF : (auxiliary flag), indicateur de retenue auxiliaire. Mis à 1 si un calcul en BCD non compacté produit une retenue. ;

ZF : (zero flag), indicateur de zéro. Mis à 1 si le résultat d'un calcul (i.e. le contenu de l'accumulateur) vaut 0. ;

SF : (signe flag), indicateur de signe. Prend la valeur du bit de poids fort de l'accumulateur après un calcul. ;

OF : (overflow flag), indicateur de dépassement. Mis à 1 si une opération provoque un dépassement de capacité.

b) Indicateurs de contrôle:

TF : (trace(trap) flag), indicateur d'exécution pas à pas. Mis à 1, il force le processeur à fonctionner pas à pas ;

IF : (interruption flag), indicateur d'autorisation d'interruption. Mis à 1, il autorise les interruptions. S'il vaut 0, il les empêche. ;

DF : (direction flag), indicateur de direction. Fixe le sens (incrémentement et décrémentation) dans lequel seront effectuées les opérations de traitement de chaînes de caractères. STD le met à 1, CLD à 0.

R : bit réservé ;

U : bit indéfini ;

V.1.4. Registres de segments : 4 registres sur 16 bits.

CS : (Code Segment), registre de segment de code ;

DS : (Data Segment), registre de segment de données ;

SS : (Stack Segment), registre de segment de pile ;

ES : (Extra Segment), registre de segment supplémentaire pour les données ;

Les registres de segments, associés aux pointeurs et aux index, permettent au CPU 8086 d'adresser l'ensemble de la mémoire.

VI. Gestion de la mémoire par le CPU 8086

L'espace mémoire adressable par le 8086 est de 1 Mo. Le pointeur d'instruction fait 16 bits donc il y a possibilité d'adresser $2^{16} = 64 \text{ Ko}$ (ce qui ne couvre pas la mémoire). La stratégie de gestion de la mémoire avec ce CPU est un peu particulière. Elle consiste à diviser l'espace mémoire adressable par le CPU 8086 en **segments**. Un segment est une zone mémoire de 64 Ko définie par son adresse de départ qui doit être un multiple de 16.

Pour désigner une case mémoire parmi les 2^{16} contenues dans un segment, il suffit d'une valeur sur 16 bits.

Ainsi, une case mémoire est repérée par le CPU 8086 au moyen de deux quantités sur 16 bits :

- 1) l'adresse d'un segment ;
- 2) un déplacement ou **offset** (appelé aussi **adresse effective**) dans ce segment.

Cette méthode de gestion de la mémoire est appelée **segmentation de la mémoire**.

La donnée d'un couple (segment, offset) définit une **adresse logique**, notée sous la forme **segment : offset**.

L'adresse d'une case mémoire donnée sous la forme d'une quantité sur 20 bits (5 digits hexa) est appelée **adresse physique**, figure 9, car elle correspond à la valeur envoyée réellement sur le bus d'adresses A0 - A19.

L'adresse physique se calcule par l'expression :

$$\text{adresse physique} = 16 \times \text{segment} + \text{offset}$$

On a la multiplication par $16 = 2^4$ revient à effectuer un décalage de 4 positions vers la gauche donc le calcul s'effectue comme suit :

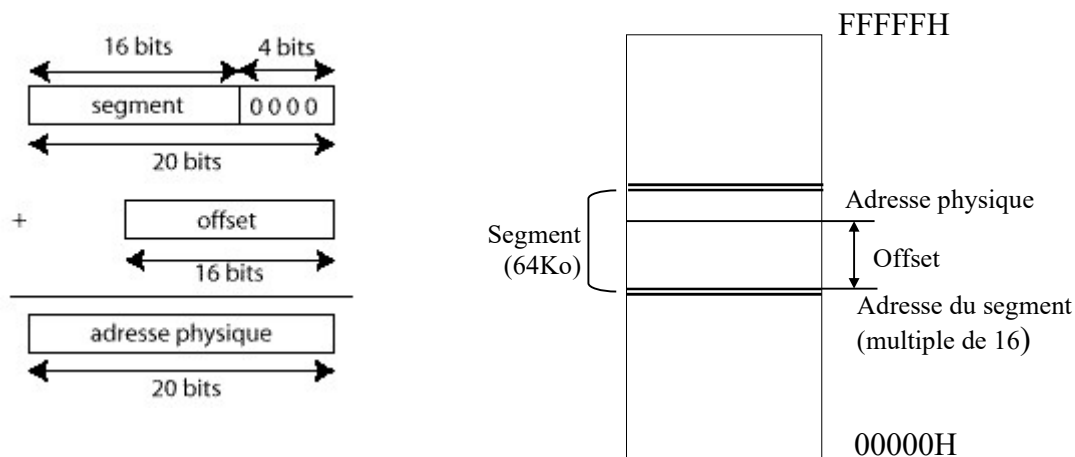


Figure 9

Le CPU 8086 peut accéder à 4 type de segments, figure 10, dont leur adresses se trouve dans les registres de segment:

- 1) Segment de code: contient les instruction du programme. Son adresse est dans le registre CS qui est associé au pointeur d'instruction IP, ainsi la prochaine instruction à exécuter se trouve à l'adresse logique CS:IP.
- 2) Segment de données : contient les donnée manipulées par le programme. Son adresse se trouve dans le registre de DS
- 3) Segment supplémentaire: peut contenir des données. Les registres de segments DS et ES peuvent être associés à un registre d'index. Exemple : DS : SI, ES : DI.
- 4) Segment de pile : contient la pile de sauvegarde. Le registre SS peut être associé au registres de pointeur de pile (SS:SP; SS:BP)

Remarque : les segments ne sont pas nécessairement distincts les uns des autres, ils peuvent se chevaucher ou se recouvrir complètement, figure 11.

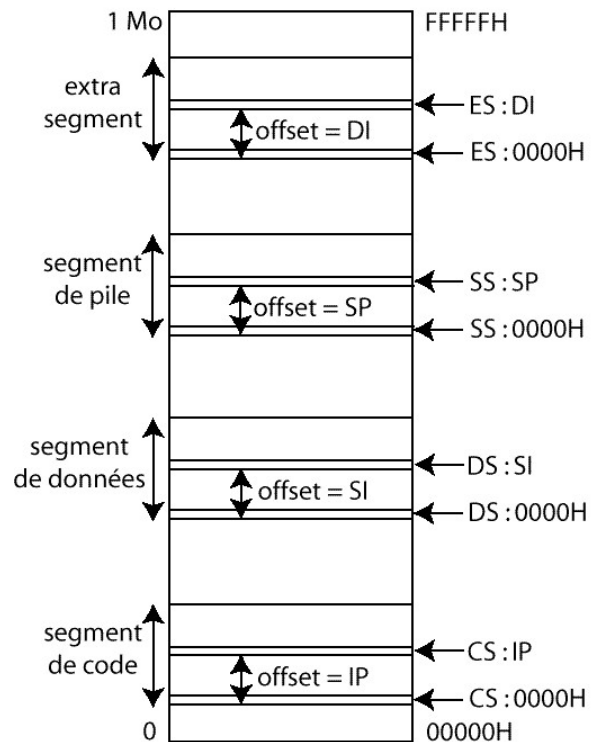


Figure 10

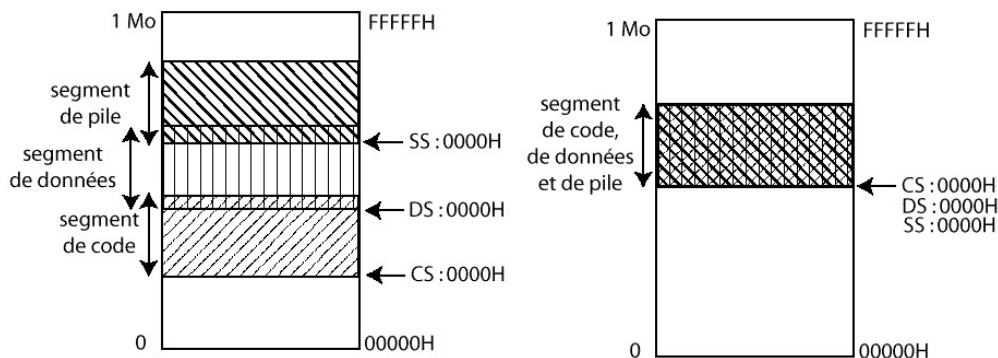


Figure 11