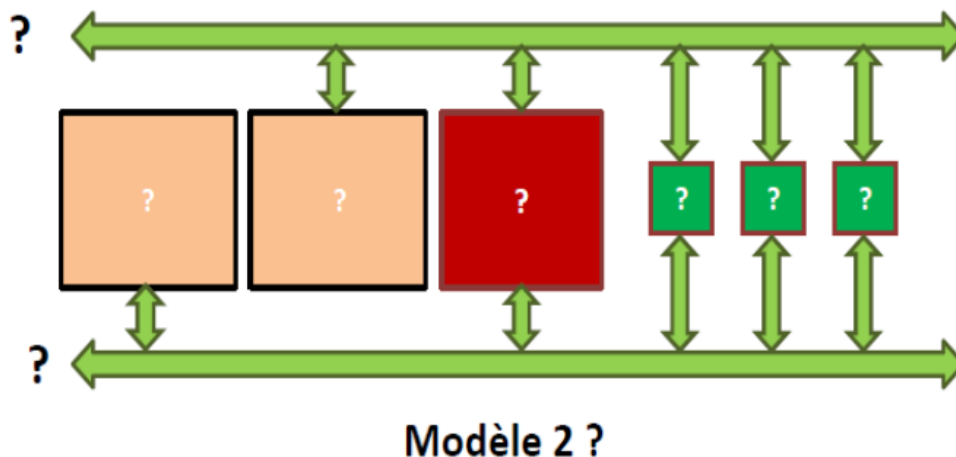
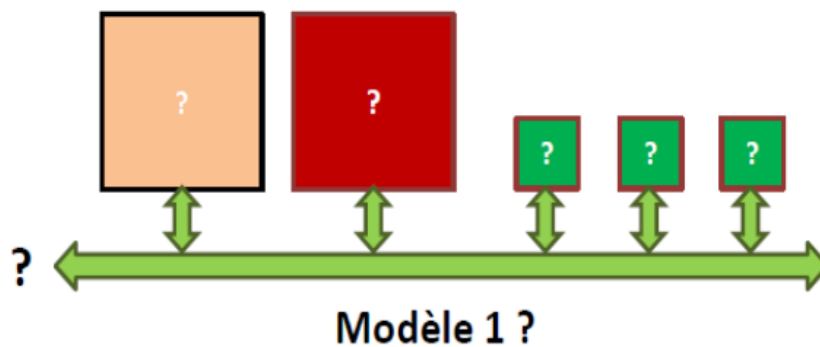


Attention : L'ensemble des exercices ci-dessous ne seront pas tous corrigés en cours !

Exercice 1 : Architecture du processeur

- 1) Compléter les schémas ci-dessous.
- 2) Quels sont les caractéristiques de chacun des deux modèles.
- 3) Définir : bus d'adresses, bus de données et bus de contrôle.
- 4) Quels sont les principaux registres utilisés ?
- 5) Où sont effectués les calculs ?



Exercice 2 : Adressage

Indiquer le mode d'adressage des instructions suivantes :

- 1) `mov esi, [esp+4]`
- 2) `mov eax, 0`
- 3) `mov [SI+146H], BL`
- 4) ~~`je egal`~~
- 5) `inc eax`

Exercice 3 : (Exemple d'instruction assembleur INTEL 8086)

L'instruction suivante (avec AX un accumulateur 16 bits) **AND AX, 06** a pour code machine : 25 06 00_H. Cette instruction est implantée à l'adresse 01 00_H.

1) Indiquer le contenu des registres IR/RI (*Instruction Register ou Registre d'Instruction*) et IP (ou PC ou CO, *Instruction Pointer ou Program Counter ou Compteur Ordinal*) juste avant l'exécution de l'instruction. IR :

2) Indiquer le contenu du registre IP juste après exécution de l'instruction.

NB : Le microprocesseur 8086 utilise un bus d'adresse de 20 bits, ce qui lui permet d'atteindre $2^{20} = 1\text{M}$ cases mémoire de 1 octet chacune.

Exercice 4 : Jeux d'instruction

Soit l'extrait de programme ASSEMBLEUR INTEL 8086 suivant, stocké à l'adresse 0100_H (via le code (ASM ORG 100h) avec les valeurs initiales :

AX = 0000_H, BX = 0000_H, et le Flag Z = 0.

L'état de pile : FFFE_H = 0000_H, FFFC_H = 0000_H, FFFA_H = 0000_H.

SP (Pointeur de pile) = FFFE_H

1	MOV AX, 0100h	B8 00 01	Ecrit 01 00h dans AX : (AX) = 01 00 – (AH) =01 ; (AL)=00
2	MOV BX, 0304h	BB 04 03	Ecrit 03 04h dans BX : (BX) = 03 04 – (BH) =03 ; (BL)=04
3	Boucle : ADD AL, 1	04 01	Ajoute 1 à l'octet de poids faible de AX noté AL : (AL) = (AL)+1
4	CMP AL, 2	3C 02	Compare AL à 2 ; place Z à 1 en cas d'égalité
5	JNE Boucle	75 FA	Saut à l'étiquette Boucle si Z=0 (si pas d'égalité)
6	PUSH AX	50	Empile le contenu de AX dans la pile (STACK) AX -> STACK
7	PUSH BX	52	Empile le contenu de BX dans la pile (STACK) BX -> STACK

Compléter le Tableau 1 correspondant aux contenus des différents registres sachant que chaque ligne représente une étape d'exécution du code précédent.

	Instruction	IP	AX	BX	Flag Z	SP	Stack : FFFF, FFFE, FFFD, FFFC, FFFB, FFFA
0	Etat initial	0100	0000	0000	0	FFFE	00 00 00 00 00 00
1	MOV AX, 0100h (B8 00 01)						
2	MOV BX, 0304h (BB 04 03)						
3							
4							
5							

6							
7							
8							
9							
10							

Tableau 1