

Spécification des instructions pour une machine à pile virtuelle

Les instructions peuvent avoir plusieurs octets de long. Le premier octet de la suite (et parfois le seul) contient les informations sur l'instruction (mode d'adressage, type d'opérande, etc), les octets suivants contiennent l'opérande.

Les instructions ont quatre formats de base, selon le mode d'adressage utilisé. Les bits 6 et 7 du premier octet spécifient ce format.

00 : mode d'adressage de type « système », instructions spéciales et entrées/sorties;

01 : mode d'adressage immédiat : l'opérande est un immédiat de 16 bits;

10 : mode d'adressage direct : l'opérande est une adresse de 24 bits;

11 : mode d'adressage relatif : l'opérande est un déplacement de 13 bits.

Les instructions dans les modes 01 et 10 n'utilisent pas toutes l'opérande mentionné. Les instructions arithmétiques, par exemple, n'utilisent pas l'opérande; leurs opérandes sont déjà sur la pile d'exécution.

Format 00 :

Octet 0

7	6	5	4	3	2	1	0
0	0	Oper			Format		

Valeurs pour le champ Oper :

000 : HALT (Format=000)

001 : SAVE (Format=000)

010 : RESTORE (Format=000)

011 : READ

100 : WRITE

Valeur pour le champ Format (READ et WRITE)

000 : « %c »

001 : « %d »

010 : « %s »

011 : « %f »

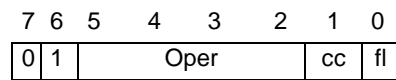
100 : « %u »

101 : « %x »

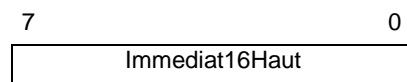
L'instruction HALT arrête l'exécution. Les instructions READ et WRITE fonctionnent comme printf et scanf, les formats étant identiques. Les valeurs à imprimer se trouvent sur le dessus de la pile.

Format 01 :

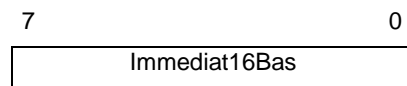
Octet 0



Octet 1

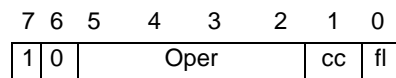


Octet 2

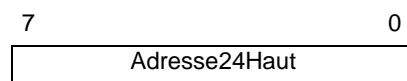


Format 10 :

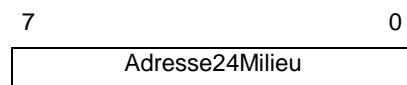
Octet 0



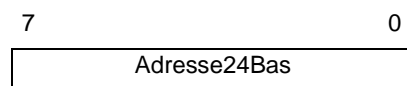
Octet 1



Octet 2



Octet 2



Ces deux formats sont utilisés pour les mêmes opérations.

Les instructions suivantes correspondent aux valeurs du champ Oper :

0000 : PUSH
0001 : POP
0010 : ADD
0011 : SUB
0100 : MUL
0101 : DIV
1111 : JMP

Seule l'instruction PUSH prend un immédiat de 16 bits dans le format 01.

Les valeurs entières placées dans cet immédiat sont en représentation complément à 2 sur un demi-mot.

Seules les instructions PUSH et POP prennent une adresse de 24 bits dans le format 10.

Les autres instructions auront donc uniquement l'octet 0 d'utilisé et **leur longueur sera donc de 1 octet.**

Les instructions ADD, SUB, MUL et DIV et JMP en format 10 appliquent un mode d'adressage indirect. Les opérandes sur la pile sont considérés comme les adresses où se trouvent les nombres à calculer.

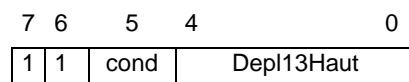
Les champs immédiat16 et adresse24 sont divisés en plaçant d'abord les bits les plus significatifs (8 à 15 ou 16 à 23 selon le cas) dans l'octet 1, jusqu'aux bits les moins significatifs (0 à 7) dans l'octet 2 ou 3, respectivement.

Le champ **cc** spécifie si l'instruction va modifier les codes condition ou non (1 = cc modifié, 0 = cc inchangé).

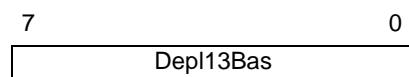
Le champ **fl** spécifie si l'on travaille avec des nombres en virgule flottante ou avec des nombres entiers (0 = entier, 1 = virgule flottante).

Format 11 :

Octet 0



Octet 1



Ce format est utilisé pour les instructions de branchement.

La valeur du bit 5 indique la condition :

- 0 : BZ (branchement si zéro)
- 1 : BNEG (branchement si négatif)

Le champ Depl13 est un déplacement en octets dans le code. Ses bits 8 à 12 sont dans l'octet 0 et ses bits 0 à 7 sont dans l'octet 1.

Seules les instructions suivantes sont implantées dans le simulateur partiel disponible pour ce devoir :

HALT
PUSH (format 01)
ADD, SUB, DIV, MUL (format 01)
WRITE (seul le format "%d" est supporté)

Seuls les nombres entiers (positifs et négatifs) sont supportés.