ARO2

Complément : Modes d'adressage

Basé sur le cours du prof. E. Sanchez
et le cours ASP du prof. M.Starkier

Types d'instructions

- **Instructions à 3 opérandes**
  Exemple : ADD r1, r2, r3 => r1 = r2 + r3

- **Instructions à 2 opérandes**
  Exemple : ADD r1, r2 => r1 = r1 + r2

- **Instructions à 1 opérande**
  Exemple : ADD r1 Acc => Acc = Acc + r1

Modes d'adressage

Modes d'adressage = méthode d'accès au données

En fonction de l'instruction à exécuter, la donnée à utiliser peut-être différente. Nous pouvons imaginer quelques cas:

- **Valeur immédiat: opérande = donnée**
- **Valeur dans un registre: opérande = no registre**
- **Valeur en mémoire: opérande = adresse**
  - **nombreuses variantes pour ce dernier cas**

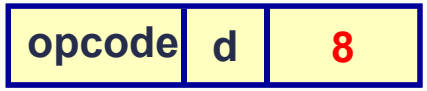
Note: Les exemples qui suivent présentent des instructions simplifiées qui ne sont pas nécessairement des instructions ARM

Modes d'adressage

Adressage immédiat : valeur (constante) dans un champ de l'instruction

mov rd, #valeur

add rd,#cte



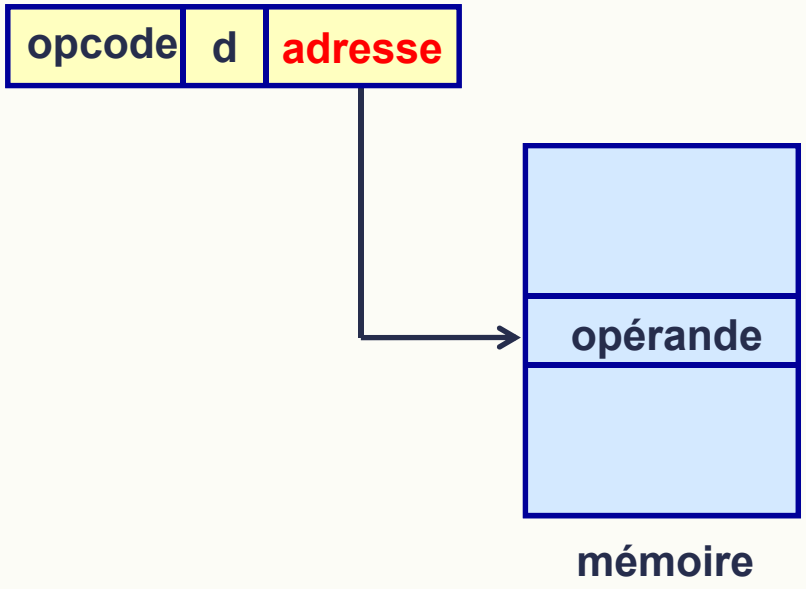
Modes d'adressage

Direct : adresse mémoire ou numéro de registre

absolu: adresse mémoire

mov rd, adresse

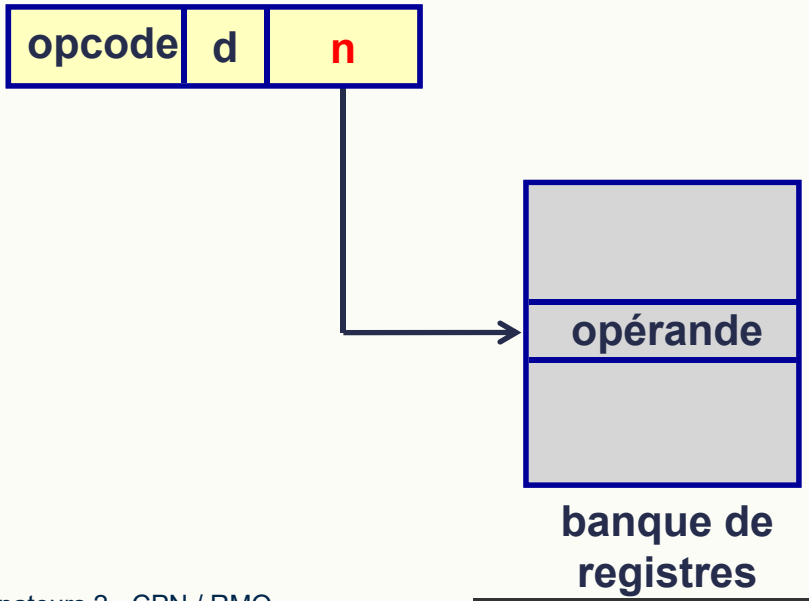
add rd,adresse



par registre: n° du registre

mov rd, rn

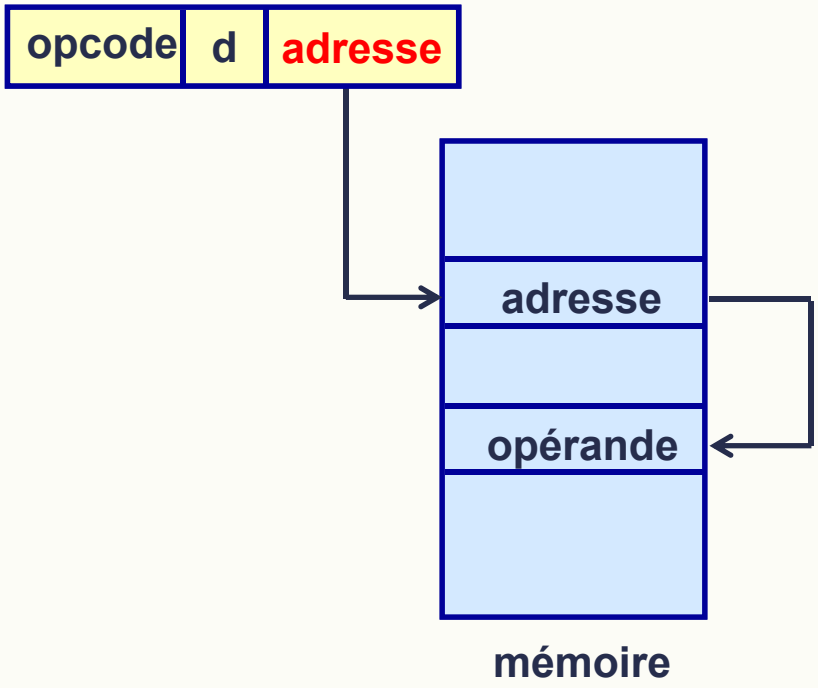
add rd,rn



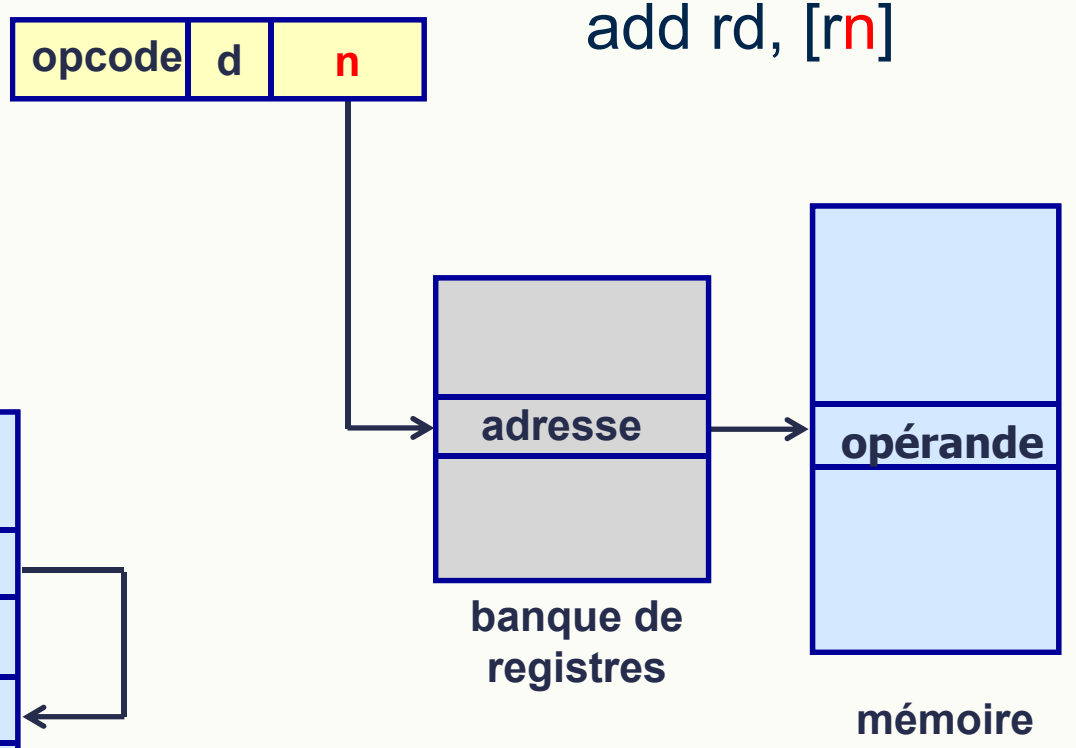
Modes d'adressage

Indirect

mov rd, [adresse]
add rd, [adresse]



mov rd, [rn]
add rd, [rn]

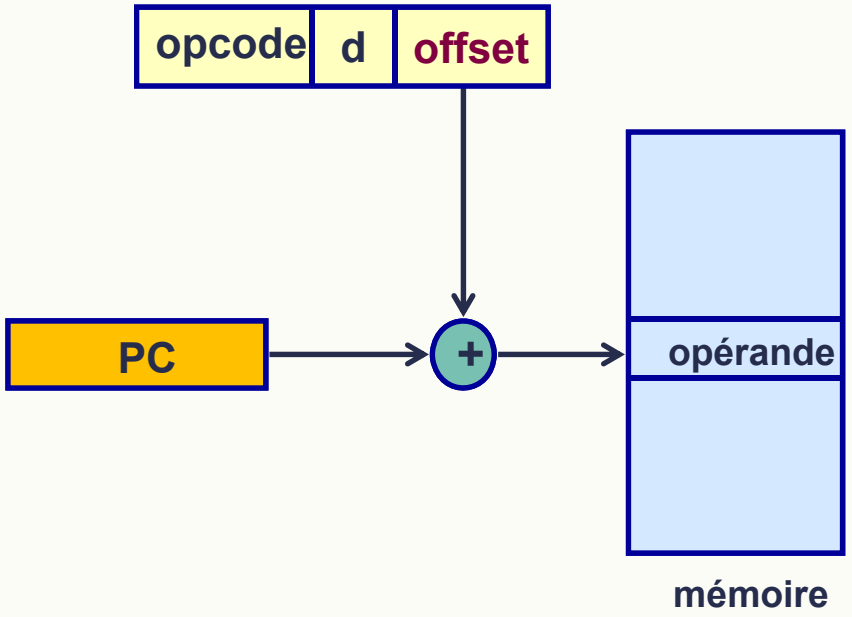


Modes d'adressage

- **Indirect relatif**

  Fréquemment on utilise le PC comme pointeur de base

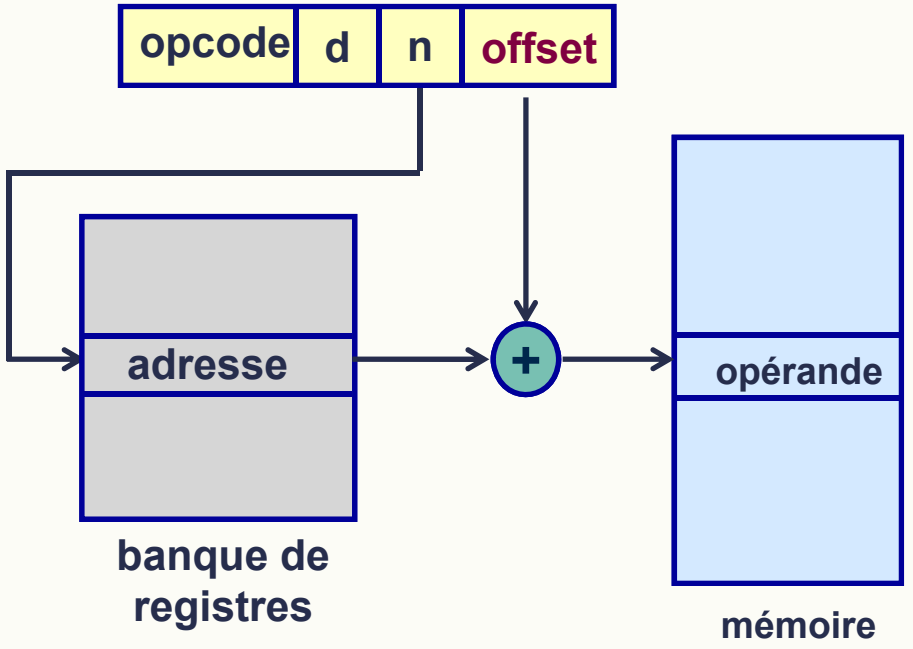
  movpc rd, [#+/- offset]



Modes d'adressage

- **Indirect relatif**
  - **Cas général avec choix d'un registre**

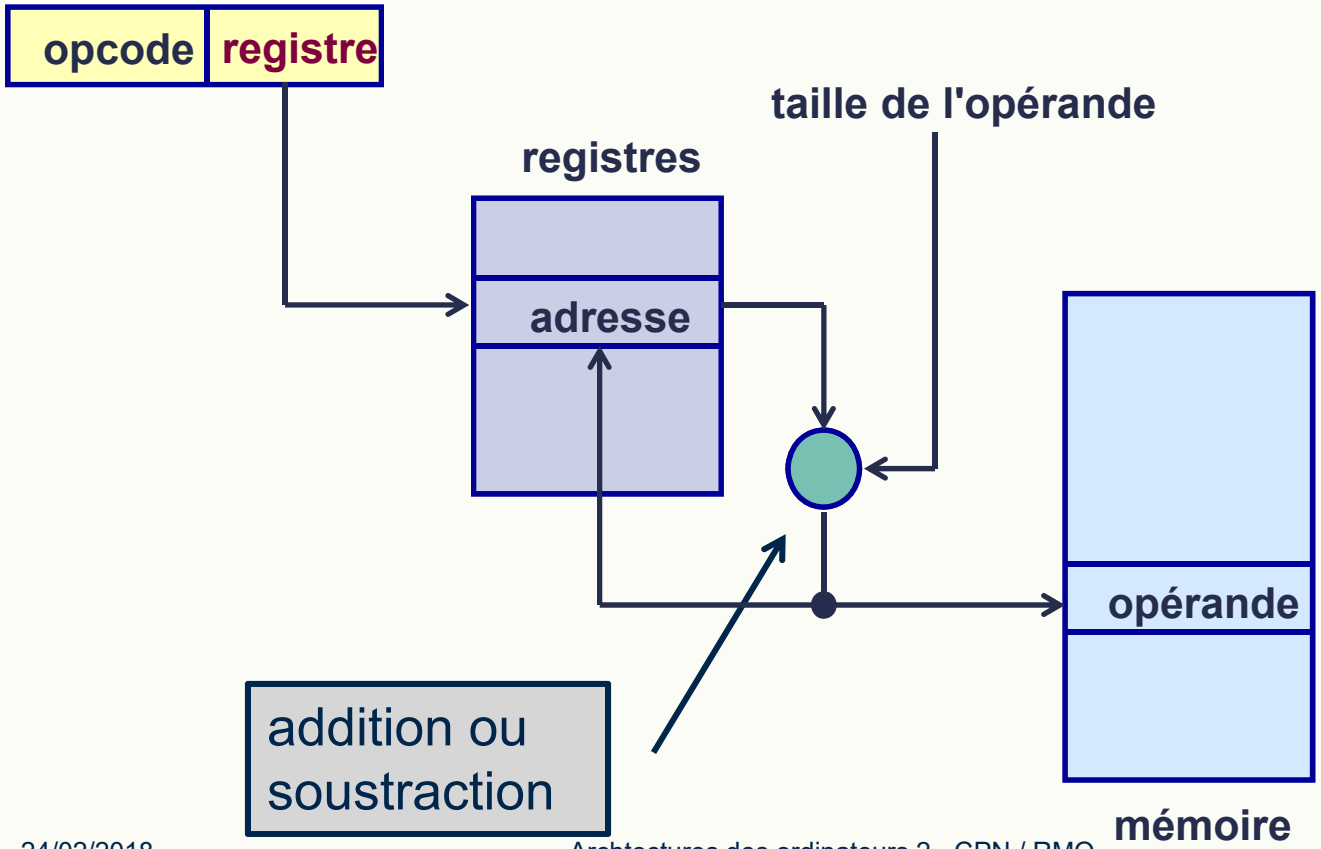
    mov rd, [rn, #+/- offset]



Modes d'adressage

Auto-incrémenté

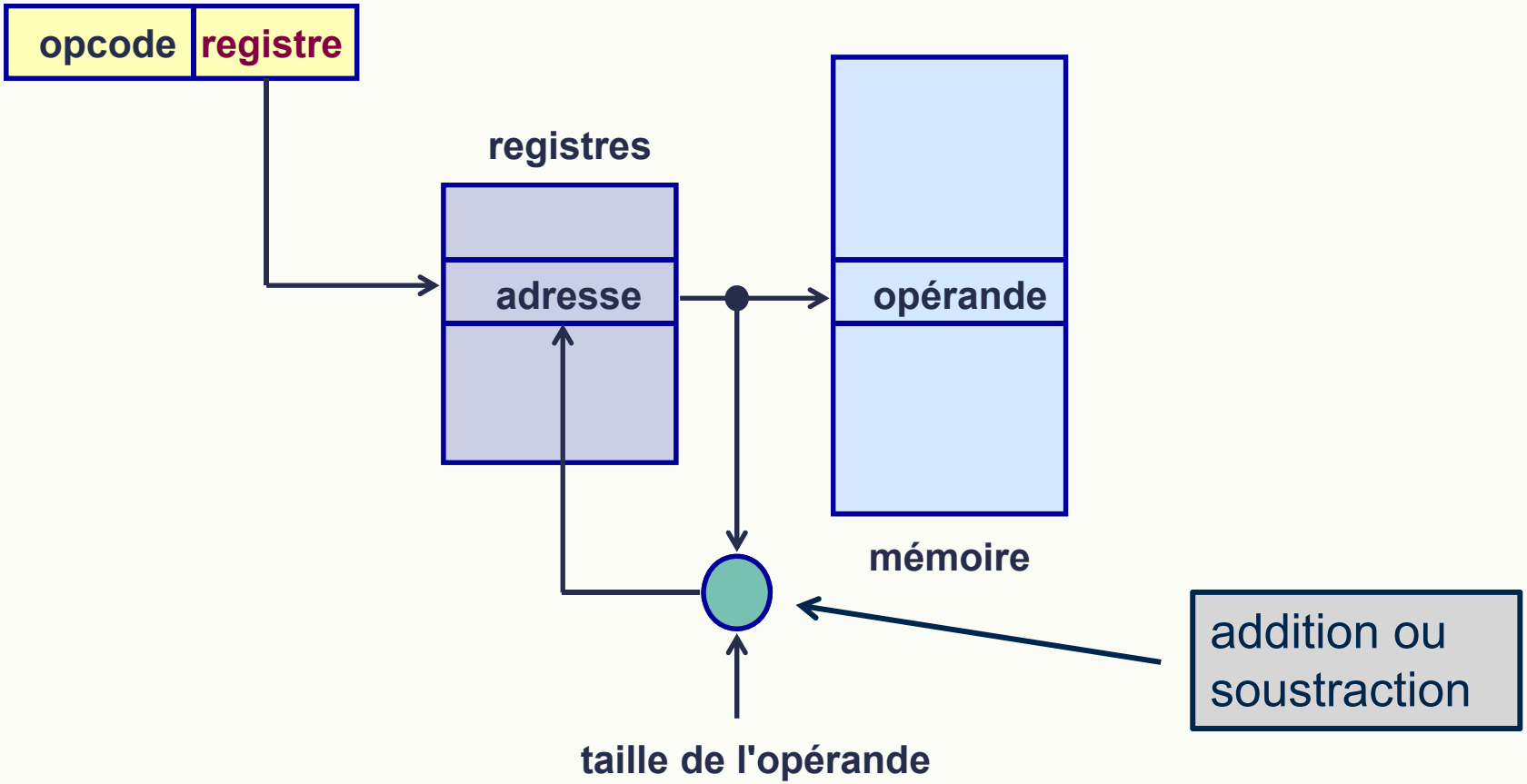
pré-incrémentation / pré-décrémentation



Modes d'adressage

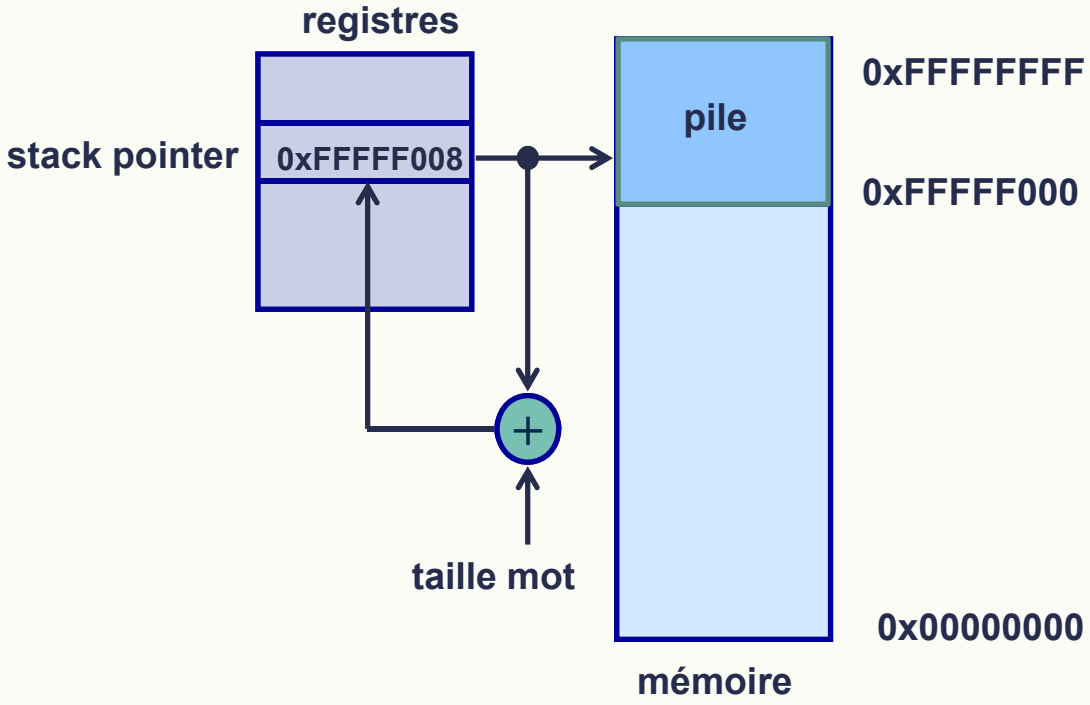
Auto-incrémenté

post-incrémentation / post-décrémentation



Pile (stack)

- Pile => zone mémoire réservée
- LIFO: Last Input , First Output
- Registre dédié => stack pointer
  - Pointe sur le bas de la pile à l'initialisation



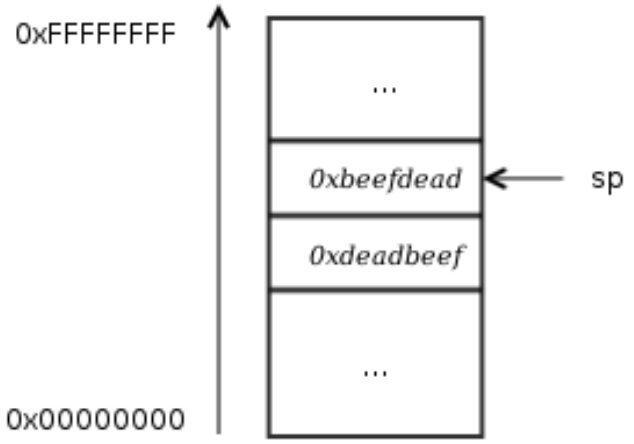
Pile (stack)

- Ecriture / lecture en mode auto-incrémenté

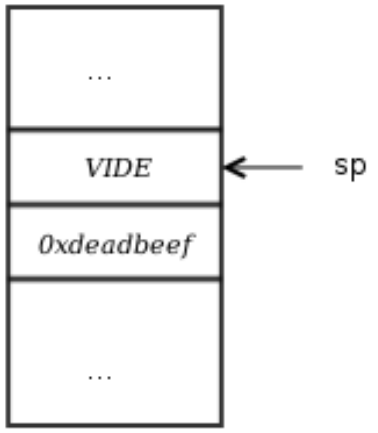
pile ascendante

- 1er mode (pile ascendante vide)
  - Push : écriture avec post-incrémentation
  - Pop : lecture avec pré-décrémentation
- 2ème mode (pile ascendante pleine)
  - Push : écriture avec pré-incrémentation
  - Pop : lecture avec post-décrémentation

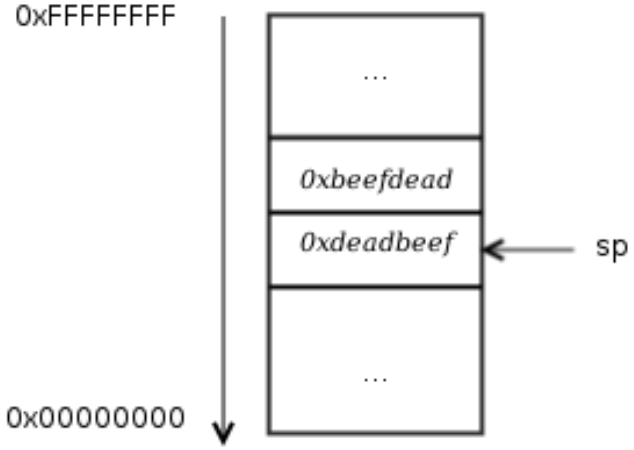
même principe pour pile descendante



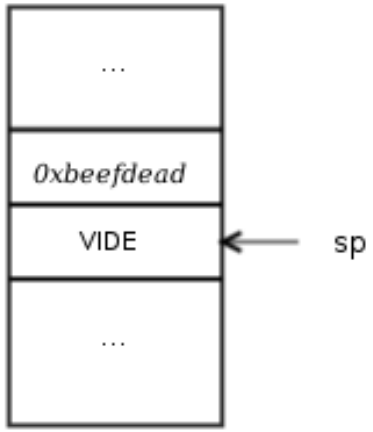
Pile ascendante



Pile ascendante vide



Pile descendante



Pile descendante vide

Pile (stack)

- ++i pré-incrémentation

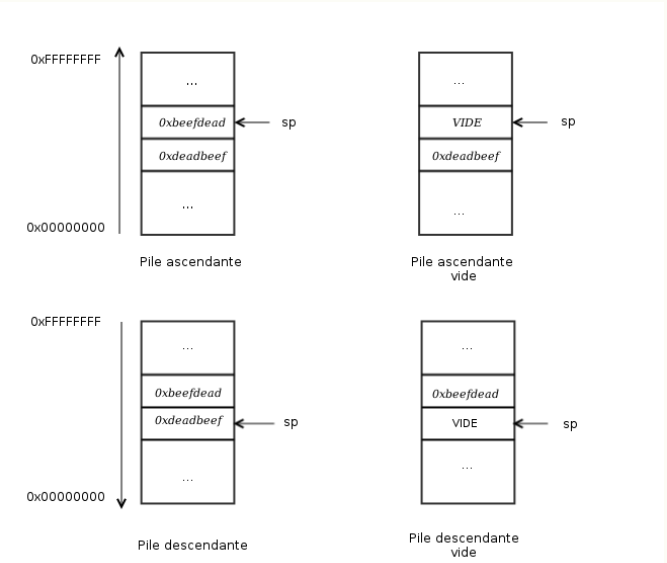
Incrémentation de SP

Puis écriture dans le registre à l'adresse SP

- i++ post incrémentation

Ecriture dans le registre à l'adresse SP

Puis incrémentation de SP



Pile (stack)

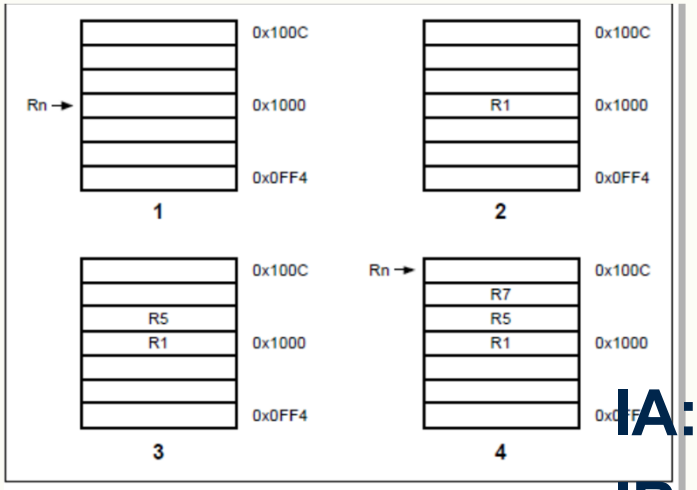


Figure 4-19: Post-increment addressing

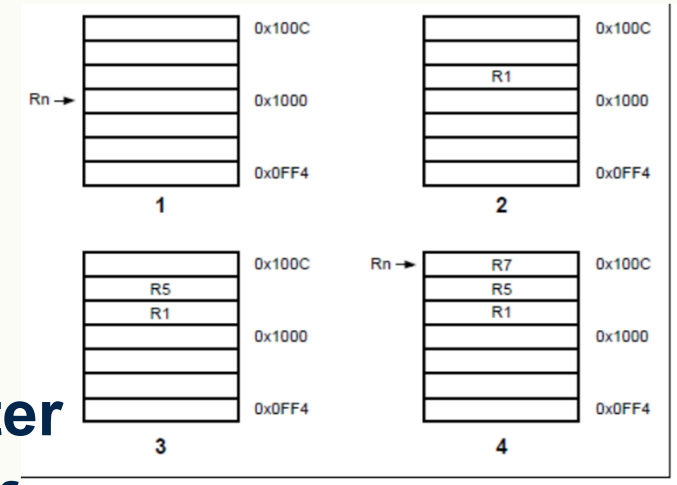


Figure 4-20: Pre-increment addressing

IA: Increment After

IB: Increment Before

DA: Decrement After

DB: Decrement Before

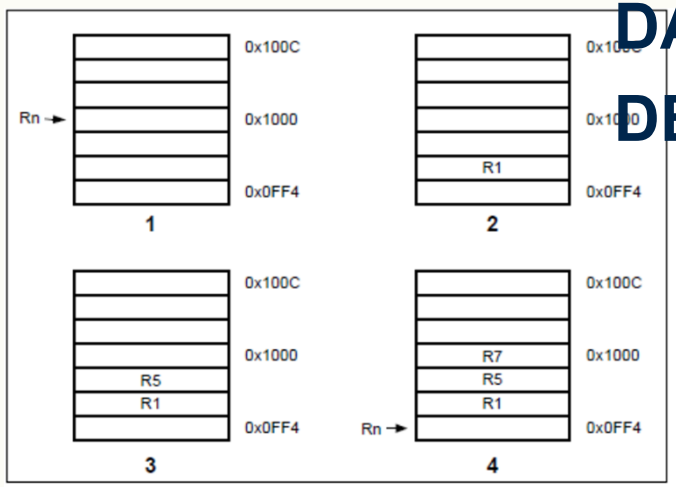


Figure 4-21: Post-decrement addressing

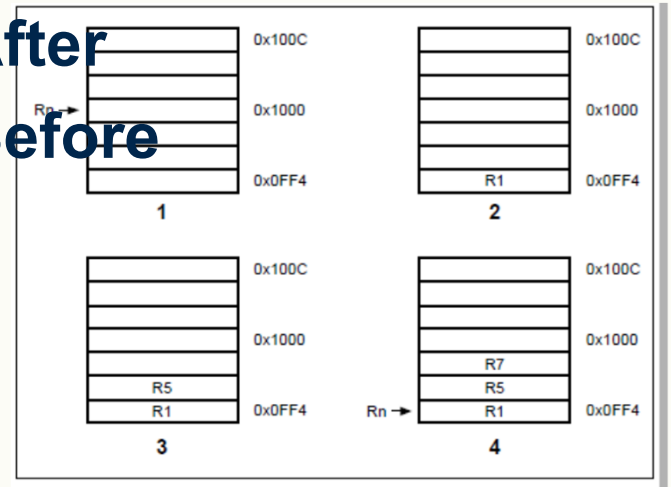


Figure 4-22: Pre-decrement addressing