

CIRC2 : Assembleur, Numération, et Circuits

Licence informatique – semestre 4

Section 1

Opérateurs bit à bit

Opérateurs logiques

- Les instructions **and**, **or**, **xor**, et **not** sont les opérations logiques bit à bit
- Sauf **not**, elles ont deux opérandes sur lesquelles elles effectuent l'opération logique et stockent le résultat dans la deuxième opérande.
- L'instruction **not** fonctionne de la même façon avec une seule opérande.

Exemple

Donnez la valeur des registres après chaque opération :

```
1 movb    $0b11001010, %al
2 movb    $0b01101110, %bl
3 orb     %al, %bl
4 andb    %bl, %al
5 notb    %al
6 xorb    %al, %bl
```

Opérateurs de décalage

- **shr** **n** Shift Right : décale les bits de **n** positions vers la droite. **n** zéros arrivent par la gauche (effectue une division par 2^n)
- **shl** **n** Shift Left : décale les bits de **n** positions vers la gauche. **n** zéros arrivent par la droite (effectue une multiplication par 2^n)

Important : le **Carry Flag** contient la valeur du dernier bit sorti par l'opération de décalage.

Donnez la valeur des registres après chaque opération :

```
1 movb    $0b11001010, %al
2 movb    $0b01101110, %bl
3 shrb    $3, %al
4 shlb    $2, %bl
```

Opérateurs de rotation

- **ror** *n* Rotate Right : effectue une rotation de *n* positions vers la droite.
- **rol** *n* Rotate Left : effectue une rotation de *n* positions vers la gauche.

Donnez la valeur des registres après chaque opération :

```
1 movb    $0b11001010, %al
2 movb    $0b01101110, %bl
3 rorb     $3, %al
4 rolb     $2, %bl
```