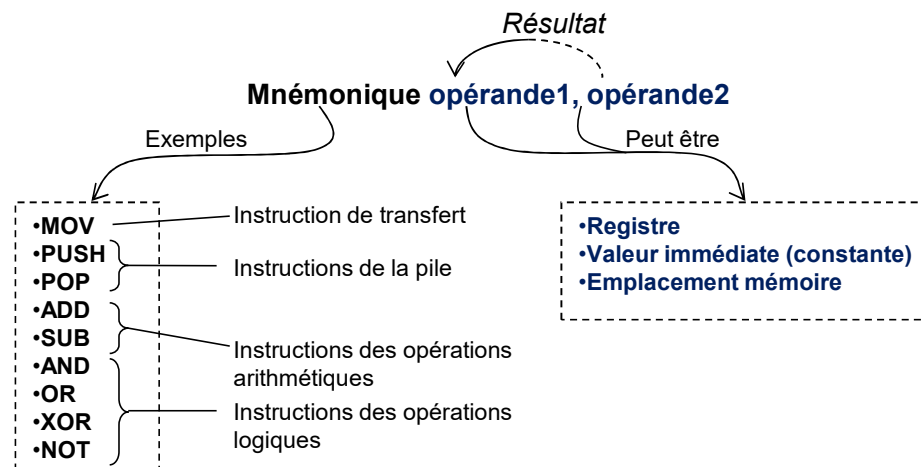




Révision des instructions du langage du CPU 8086

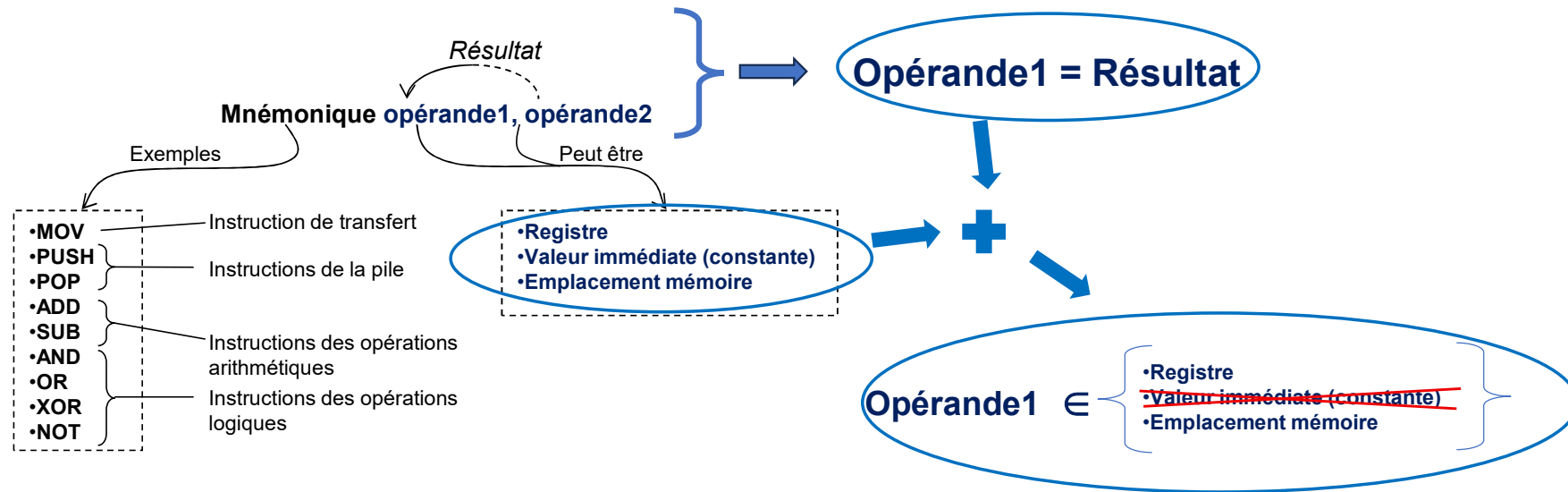


La structure d'une instruction du langage d'assemblage du CPU 8086





La structure d'une instruction du langage d'assemblage du CPU 8086





Langage d'assemblage

Algorithme

Opérande1 = Résultat

Mnémonique **opérande1**, **opérande2**

Opérande1 ← **Opérande1** opération **opérande2**

Exemples:

•ADD AX,BX

▶ **AX** ← **AX** + BX

•SUB CX,0125

▶ **CX** ← **CX** - 0125h

•AND BL, FE

▶ **BL** ← **BL** AND FEh

•OR CH, AL

▶ **CH** ← **CH** OR AL

•XOR DX, FFFF

▶ **DX** ← **DX** XOR FFFFh



Langage d'assemblage

Algorithme

Opérande1 = Résultat

Mnémonique opérande1, opérande2

Opérande1 ← Opérande1 opération opérande2

CH ← AL XOR CH

AX ← AX OR FFFFh

BL ← F1h AND BL

AX ← BX + AX

DX ← BX - DX



Langage d'assemblage

Algorithme

Opérande1 = Résultat

Mnémonique opérande1, opérande2

Opérande1 ← Opérande1 opération opérande2

XOR CH,AL

CH ← AL XOR CH

OR AX,FFFF

AX ← AX OR FFFFh

AND BL,F1

BL ← F1h AND BL

ADD AX,BX

AX ← BX + AX

NEG DX
ADD DX,BX

DX ← BX - DX



Attention aux opérations non commutatives



Langage d'assemblage

Algorithme

Opérande1 = Résultat

Mnémonique opérande1, opérande2

Opérande1 ← Opérande1 opération opérande2

DX ← AX + DX - BX - 0125h





Langage d'assemblage

Algorithme

Opérande1 = Résultat

Mnémonique opérande1, opérande2

Opérande1 ← Opérande1 opération opérande2

ADD DX,AX
SUB DX,BX
SUB DX,125

DX ← AX + DX - BX - 0125h



Langage d'assemblage

Algorithme

Opérande1 = Résultat

Mnémonique opérande1, opérande2

Opérande1 ← Opérande1 opération opérande2

DX ← AX + CX - BX - 1



Langage d'assemblage

Algorithme

Opérande1 = Résultat

Mnémonique opérande1, opérande2

Opérande1 ← Opérande1 opération opérande2

MOV DX,AX
ADD DX,CX
SUB DX,BX
DEC DX

