

4) ^{decode} loop index = adr, adr, adr
a b c

met A3 = adresse de l'index
D2 = a
D4 = b
D3 = c
et initialise l'index
(A3) ← a
(après lecture de a, b, c)
pas fait : syntaxe n = a
(= n = a, a, 1)

```
DECLoop: BSR WNO MI
          BNE ERR FN
          AND # $E0, D0
          CMP # $20, D0
          BNE ERR FN
          CMP # 3, D2
          BNE ERR FN
          BSR DECCRE
          BNE ERR IS
```

→ n'est pas index
→ n'est pas #32

index = adr, (adr)

adresse de l'index

```
BSR WADR
MOVE.L D3, D2
BSR DECCRV
BNE ERR FN
GEK62: BSR WADR
        MOVE.L D3, D4
        MOVEQ #1, D3
        CMP.L D2, D4
        BGE MA70
        MOVEQ #-1, D3
MA70: BSR DECCRV
        BNE MA71
GEK64: BSR WADR
MA71: TST.L D3
        BEQ ERR FN
MA70: MOVE.L D2, (A3)
        RTS
```

a

b

[c] défaut = ±1

{ MOVE SR, -(SP)
BEQ GEK62
BSR DECCRV }

{ BEQ GEK62
ADDA #2, SP
MOVE.L D2, D4
MOVEQ #1, D3
BRA MA70 }

pas fait

{ BEQ GEK64
BSR DECCRV }

MOVE.L A4, A5

```
MA70: MOVE (SP)+, SR
        BNE GEK64
        BSR DECCRV
        BNE MA71
        BRA GEK66
GEK64: BSR DECCRV
        BNE MA71
GEK66: BSR WADR
```

pas fait