

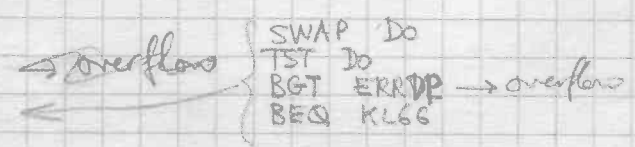
① entrée [A0] met [A2] = Int (2^{D2.W} [A0])
D2.W signé

démit D0/D1/D4/D5/A0
met A2/A6
casera le reste (D2/D3/D6/D7)
A7/A8-AS

```

XROT: MOVE.L A6, A2
      CMP    #$4000, (A0)
      BEQ    KL65
      BSR    XBNB
      MOVE.L D1, D0 ← EXT.L D2
      ADD.L  D2, D0
      BVS   ERRVL
      BGT   KL66
  
```

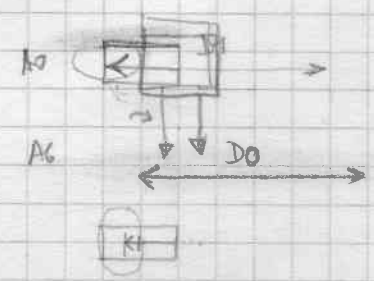
D1.L = nb de bit de [A0]



met zéro

```

KL65: MOVE #4000, (A6) ← CLR D4 ← MOVE (A0), D4
      MOVE D2, (SP) ← RTS
      NEG  D2 ← REG: SWAP
      AND  #F, D2 ← DZE [0, 15]
      ?    ← nl de ds
  
```



```

      MOVE (A0)+, D4 ← MOVE D4, (A6)
      AND  #FFFF, D4 ← AND #4000, (A6)
      BCLR #14, D4
      BNE  KL67
      MOVE (A0)+, D4 ←
  
```

```

KL67: BSET #6, (A6) ← CLR D5
      CMP  #13, D0
      BCS  KL68
      MOVE D0, D5
      LSR  #4, D5 ⊗
      ADDQ #1, D5
      ADD  D5, D5 ← BCLR #6, (A6) ⊗
      OR   D5, (A6)+ ← SUBQ #2, D5
      CLR  (A6) ⊗
  
```

```

KL68: MOVE  D4, D0
      AND  #FFEF, SR ← x=0 D0H=0
      ROXR.L D2, D0 ← TST D0
      BNE  KL70
  
```

```

X KL680: SWAP  D4 ← CLR D4
      MOVE  (A0)+, D4 ← SUB.L #16, D1 ⊗
      BMI  KL69
  
```

```

KL69: MOVE.L D4, D0
      ROXR.L D2, D0
  
```

```
KL70:OR D0,(A6)+  
CLR (A6)  
VERAC  
SUBQ #2,D5  
BPL KL680  
MOVE (SP)+,D2  
RTS
```