

5 get #n [j,r]

put #n [j,r]

```

YGET:BSR DECCDF          (n)
      BNE ERR18          (pour le moment)
      CLR  -(SP)
      BRA  GAG30

```

```

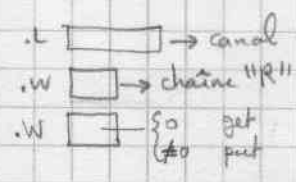
YPUT:BSR DECCDF          (n)
      BNE ERR18
      ST   -(SP)

```

```

GAG30:BSR XCANAL6        (n) ouvert "R"    d5/a1
      MOVE (A2)+, d4      } chaîne "R"
      MOVE d4, -(SP)     } canal
      MOVE.L A0, -(SP)
      MOVE.L (A2), D3
      ADDQ.L #1, D3
      BSR   WVGAD
      MOVE.L D3, (A2)
      SUBQ.L #1, D3

```

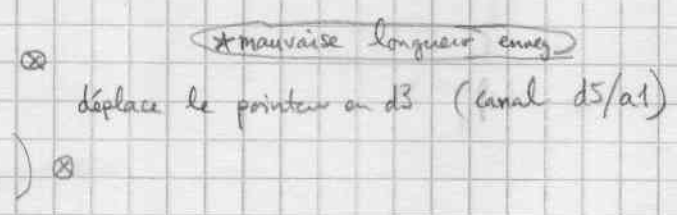
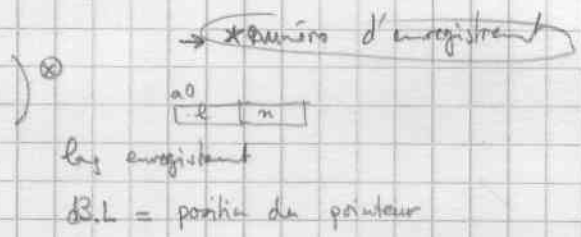


[j,r] défaut = d3

```

      BMI ERR109
      MOVE.L D3, -(SP)
      BSR   LR95C
      MOVE.L (SP)+, d3
      MOVE.L (A0)+, d1
      BSR   LMUL13
      TST.L (A0)+
      BNE  ERR108
MOVE.L A0, -(SP)
      BSR   XSEEK
      MOVEQ #8, d0
      MOVE.L d0, -(SP)

```



```

GAG32: MOVE.L (SP)+, A1
      MOVE.L (A1)+, D3
      BNE GAG33
      fin
      ADDQ #8, SP
      RTS
  
```

```

GAG33: MOVE.L (A1)+, A3
      MOVE.L (SP), A0 canal
      MOVE.L A1, -(SP)
  
```

```

TST 10(SP)
BEQ GAG36
MOVE.L A3, D4
BMI GAG35
  
```

```

GAG34: BSR PTCAN
      BRA GAG32
  
```

```

GAG35: MOVEM.L D3/A0, -(SP)
      BSR LBS95C
      CMP.L (SP)+, D3
      BNE ERR107
      MOVE.L A0, A3
      MOVE.L (SP)+, A0
      BRA GAG34
  
```

```

GAG32: MOVEM.L (SP)+, D0/A1 canal
      MOVE (SP), D4
      BSR LBS95C
      ADD.L D0, A0
      EXG A0, A1
      ADDQ.L #8, D0
      CMP.L D3, D0
      BLE GAG33
      ADDQ #4, SP
      RTS
GAG33: MOVEM.L (A1)+, D3/A3
      MOVEM.L D0/A0, -(SP)
  
```

longueur



→ get  
↓ put

↓ cas indice

cas chaîne

→ \*mauvaise longueur d'un déplacement\*

x

⊗

GAG36: MOVE.L A3, D4  
 BMI GAG38

cas got

↓ cas index

MOVE.L A6, -(SP)  
 MOVE.L A3, A6  
 BSR RDCAN  
 MOVE.L (SP)+, A6  
 BRA GAG32

GAG38: MOVE D4, -(SP)  
 ADDQ #6, A6  
 MOVE.L A6, -(SP)  
 BSR RDCAN1  
 MOVE.L (SP)+, A2  
 BSR LC12  
 MOVE (SP)+, D0  
 BSR WASGN2  
 BRA GAG32

cas chaîne d4

lit d3 octets

créé chaîne

assigne