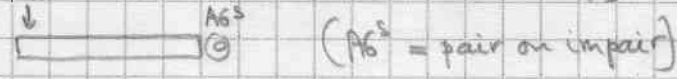




écrit en libra = A0<sup>s</sup> la chaîne "view" (directing)



de l'adresse

```

VIEW2: MOVE.L A6, -(SP) ← LEA T3FDB6, A3

```

```

LEA T3E1AA, A0
BSR GES60

```

```

MOVE d5, do
SUBQ #2, d5
BLE GAM10
BTST #4, d3
BEQ GAM11

```

```

PEA T3FDB6 ← MOVE.L A3, -(SP)

```

tampon de stockage

```

MOVE #1A, -(SP)
TRAP #1
ADDQ #6, SP

```

```

LEA T3E1AA, A0
MOVEQ #0, do
MOVE.B (A0)+, d1
BEQ GAL60
CMP.B #":", (A0)
BNE GAL60
AND ##, d1
MOVE d1, do

```

```

GAL60: BSR DFREET
MOVEQ #10, D1
BSR GES64

```

```

GAM11: BSR GES66

```

```

MOVE #3F, -(SP)
PEA T3E1AA
MOVE #4E, -(SP)
TRAP #1
ADDQ #8, SP
TST Do
BNE S3E1A2

```

```

LEA GES7E, A0
BSR GES60

```

"octets libres"

```

GAL60: CMP #2, do
BLE GAM10
BTST #4, d3
BEQ GAM11
GAM10: BSR dfreet

```

```

S3E14C: MOVEQ #3, D5

```

```

BRA S3E14F GES90
S3E14E: MOVE.B #",", (A6)+

```

```

S3E14F: MOVE.L T3FDB6+26, Do
MOVEQ #6, D1
BSR GES64
MOVEQ #26, Do
MOVE.B T3FDB6+21, D1

```

```

GES90: MOVEQ #32, D4
TST D3
BNE S3E14F
MOVE.L 26(A3), Do
MOVEQ #6, D1
BSR GES64
MOVE.B D4, (A6)+
BTST #4, 21(A3)
BEQ S3E14F
MOVE.B #"+", -1(A6)

```

```

I3E15C: BEQ I3E15D
ADDQ #1, Do
ASR.B #1, D1
BRA I3E15C

```

```

I3E15D: MOVE.B Do, (A6)+

```

```

S3E14F: MOVEQ #11, D3 D1 ← MOVEQ #32, D4
LEA T3FDB6+30, A0 ← LEA 30(A3), A0

```

```

S3E168: MOVE.B (A0)+, Do
BEQ S3E176

```

```

MOVE.B Do, (A6)+
DBF D3, S3E168
BRA S3E176 GES83

```

```

S3E176: MOVE.B #1, (A6)+
DBRA D1, S3E176

```

```

GES83: BTST #0, D3
      BEQ  GES84
      MOVE.B 21(A3), D0
      MOVE.B D4, (A6)+
      BSR  GES78
  
```

type du fichier

```

GES84: TST D3
      BEQ  GES86
  
```

```

GES84: BTST #1, D3
      BEQ  GES88
  
```

```

GES86: MOVE.L 26(A3), D0
      MOVEQ #7, D1
      BSR  GES64
  
```

layeur

```

GES88: BTST #2, D3
      BEQ  S3E17E GES89
  
```

date et heure

```

      MOVE.B D4, (A6)+
      MOVE 24(A3), D0
      BSR  XDATEQ
      MOVE.B D4, (A6)+
      MOVE 22(A3), D0
      BSR  XTIMEQ
  
```

date

```

GES89: BTST #3, D3
      BEQ  S3E17E
  
```

heure

x

~~S3E17E: MOVE #4F, -(SP)~~  
~~TRAP #1~~  
~~ADDQ #2, SP~~  
~~TST D0~~  
~~BNE S3E1A3~~  
~~DBRA D5, S3E14E~~  
~~BSR GESG6~~  
~~BRA S3E14C~~

```

S3E17E: MOVE #4F, -(SP)
TRAP #1
ADDQ #2, SP
TST D0
BNE S3E1A3
DBRA D5, S3E14E
BSR GESG6
BRA S3E14C

```

TST D3  
DBNE D5, S3E14E

```

S3E1A2: LEA U3FA35, A0
BSR GESG6

```

```

S3E1A3: BSR GESG6
MOVE.L (SP)+, A0
RTS

```

← CLR.B (A6)



6

SP

```

GES60: MOVE.B (A0)+, D0
      BEQ GES62
      BSR VERAG
      MOVE.B D0, (A6)+
      BRA GES60

GES62: RTS

```

recopie A0 → A6 avec vérif

SP

```

GES64: MOVE.L A6, A0
      BSR PINT
      MOVE.L (A0), A6
      RTS
      MOVE.L A0, A6
      BRA VERAG

```

écrit do.L sur dt espaces

SP

```

GES66: MOVE.B #13, (A6)+
      MOVE.B #10, (A6)+
      RTS
      BRA VERAG

```

écrit 13 et 10

SP

après 33FC4

```

→ GES78: MOVE.B D0, D1
      LSR.B #4, D0
      BSR GES80
      MOVE D1, D0

GES80: AND #0F, D0
      ADD #10H, D0
      CMP #39, D0
      BLE GES82
      ADDQ #7, D0

GES82: MOVE.B D0, (A6)+
      RTS

```

écrit do.b et hexa