

```

TFRUN: D.W 0
GEQ 47: BSR RERDK 223
GEQ 47: MOVE D0, -(SP)
BSR RERPOP
MOVE (SP)+, D0
GEQ 47: BSR RERDK 223
    
```

0 edit/source
 ≠ 0 run/debug/immediat
 examine si erreur lors de decode

```

SRERF: CLR TPRINT
        BSR XESAVH
        BSR DAMPR1
    
```

si "error label" est valide se branche en label
 (seul si mode erreur éditeur)

```

MOVE.L TDEBAS, A5
BSR YERRT 149
CLR TPRINT
BSR RERM
    
```

message d'erreur

```

REDIT: TST TFRUN
    
```

```

BEQ HD289 → éditeur
MOVE # $4E75, escape → désable escape
    
```

```

MOVE.B # ". ", TDF4
MOVE.L TDEBAS, A5
BSR X9
BSR LOCALS
    
```

localise l'instruction exécutée BSR LOCALT

```

MOVE.L TDECAS, A5 ← SUBQ #1, A5
CMP.L A0, A5
BCS PD32
CMP.L A1, A5
BCC PD32
    
```

A5E ligne trouvée
 MOVE D1, -(SP)

sortir A0 A5

```

BSR PD44 ← MOVE (SP)+, D1
BRA PD34 ← BEQ PD42
    
```

→ si hors de la pile avec le dernier point décodé

```

PD32: BSR LOCALS
      CMP.L A0, A5
      BCS PD34
      CMP.L A1, A5
      BCC PD34
      BSR PD44
    
```

sort la ligne en erreur

```

PD34: MOVE.L TDEBAS, A5
      MOVE #100, D6
      BSR LOCALS
    
```

localise l'instruction exécutée

```

TST D1
BEQ PD40
instruction dans la pile avec
    
```

```

PD36: CMP.B #1, (A1)
      BNE PD43
      MOVE.L 2(A1), A5
      SUBQ #2, A5
      DBRA D6, PD35
      BRA PD40

```

~~PD40: CMP.L #IMM12, A5~~ cas immédiat
 { BEQ PD42

```

PD43: TST.B (A1)+
      BNE PD43
      BRA PD36

```

```

PD40: BSR PLGN1M
PD42: BRA HD289

```

soit la ligne

PD44: ~~BSR X15~~

SP

repete ME12

```

MOVE.B (A5), -(SP)
MOVE.B #11?, (A5)+
MOVE.B (A5), -(SP)
CLR.B (A5)+
MOVE.L A5, -(SP)

```

SUB #75, A5

CMP.L A5, A0

BCC PD46

LEA PD48, A0

BSR X15

MOVE.L A5, A0

PD46: BSR X15N

MOVE.L (SP)+, A5

MOVE.B (SP)+, -(A5)

MOVE.B (SP)+, -(A5)

BRA PLGN1M

PD48: D.B "...", 0

x