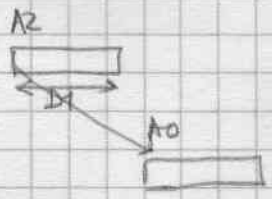
```
LB32: CMP.B   #$10,D0
      BCC     LB33

LB320: BSR    WNUMVL      met D2 = n° de la variable  ↓cas nom de variable (ou chaine [WCHAS1])

LB321 MOVE.L  TMVAR,A1    ⊗   crée V(TVARN) = variable chaine (D2)
      MOVE    D2,D0
      EXT.L   D2
      ASL.L   #2,D2
      SUB.L   D2,A1
      MOVE.L  (A1),A2

      CMP     (A2)+,D0
      BNE     ERRFAT          → numéro ne convient
      MOVE.L  (A2)+,D1        longueur
      ADDQ    #2,A6
      MOVE.L  A6,A1
      MOVE.L  D1,(A6)+

      MOVE.L  A6,A0
      ADDQ.L  #1,D1
      BCLR    #0,D1
      MOVE.L  A6,A0
      ADD.L   D1,A6
      BSR     VERA6
      BSR     LB76
      MOVE.L  A1,A2
      BRA     LB131
```

BSR VERA6 / LB76H



pushn met {A0} de la pile à sous forme prioritaire



```
LB33: CMP.B   #$70,D0
      BNE     LB340   ← BCS LB35  ⊗
      MOVE.L  A3,A0

PUSHN: ADDQ   #6,A6
      MOVE.L  A6,-(SP)
      MOVE    PRIOR,(A6)+   ← repet MB22
      BNE     LB34
      CLR.L   (A6)+
LB34: BSR     XPOSE
      BRA     LB130
```

(3)

```
        SUB
LB340:  CMP.B   #$74,D0      ⊗
        BNE     LB341
                             cas variable interne
        MOVE.L  (A3),D0
        BRA     LB305

        SUBQ.B  #2,d0
LB341:  CMP.B   #$76,D0      ⊗ ($76)  variable interne indirecte
        BNE     LB343
        MOVE.L  (A3),A3
LB342:  MOVE    (A3),D0
        BRA     LB303

        SUBQ.B  #1,D0
LB343:  CMP.B   #$77,D0      ($77) variable temps/date → voir p 51b
        BNE     LB344   ⊗
        BSR     XTIMB1       ) lit temps et date
        BRA     LB342

LB344:  SUBQ.B  #2,D0        ($79) [ff-at]  V-fonctia et procédure
        BNE     ERRMT
        BRA     LB190
```

```
LB343 : SUBQ.B  #1,D0
        BNE     LB344
        CMP.B   #"(", (A5)+
X       BEQ     .KN21
        SUBQ    #1,A5
        BSR     XTIMB1
        BRA     LB342
```

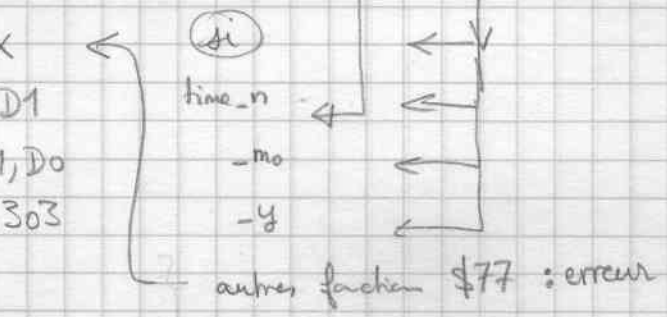
⊗ ($77) variable temps/date

```
KN21 : BSR   WJUL1
```

décode j) et met d0.W y, d1.W mo, d2.W n, d4.L d, d6 ≡ j+1 mod 7 ⊗

```
       LEA   KN22,A0
       ADD   4(A3),A0
       JMP   (A0)
```

```
KN22 : BRA   ERR1X
YFTIMEN : MOVE  D2,D1
YFTIMEMO : MOVE  D1,D0
YFTIMEY : BRA   LB303
```



(3)

```
LB35:CMP.B  #$52,D0
     BNE    ERRMT      → types mélangés
     BSR    PUSHNZ     crée valeur nulle en toppile (= valeur par défaut)
     MOVE   #2,-(SP)

LB36:BSR    APROC1
     MOVE.L TPILE,(A0)+
     MOVE.L TPILE,(A0)+
     MOVE.L TVARNB,D1
     MOVE.L D1,(A0)+
     MOVE.L D5,(A0)+
     LEA    GVALU,A2
     MOVE.L (A2),(A0)+
     CLR.L  (A0)+
     MOVE   (SP),D0
     ADDQ   #1,D0
     ASL    #4,D0
     MOVE   D0,(A2)+
     MOVE   D1,(A2)
```

(de aprocl) vient

→ D1 = tvarn

tmfor

MOVE D2,(A0)+ ⊗

⊗ ancien value place pour remember

2
3

tvarn } nouvelle variable valeur

③

```
LB35 : CMP.B   #$52,D0      L[illegible]
       BNE     ERRMT        → type mélangés
       BSR     PUSHNZ       ↓ fonctions externes → crée valeur nulle sur la pile (= valeur par défaut)
       MOVE    #2,-(SP)     ⊗ fonction variable
LB36 : BSR     APROC1       met Tpile, Tvarnd et Tvarn de la pile procédure
                            D6 = nb d'arguments max - 1
       CLR     D2           nb d'arguments
       BSR     DECCRPG      "("
       BNE     LB43         pas d'arguments
       CLR     D3           ← nb de "(" - nb de ")" - 1
LB38 : MOVE.L  A5,(A0)+
       ADDQ    #1,D2

LB39 : MOVE.B  (A5),D0
       BEQ     ERRAR
       CMP.B   #";",D0
       BEQ     ERRAR
       ADDQ    #1,A5
       CMP.B   #$22,D0
       BNE     LB41

LB40 : MOVE.B  (A5),D0      avance en fin de "
       BEQ     ERRAR
       ADD.Q   #1,A5
       CMP.B   #$22,D0
       BNE     LB40
       BRA     LB39

LB41 : ADDQ    #1,D3
       CMP.B   #"(",D0
       BEQ     LB39
       SUBQ    #1,D3
       CMP.B   #")",D0
       BEQ     LB42
       TST     D3
       BNE     LB39
       CMP.B   #",",D0
       BNE     LB39
       DBRA    D6,LB38
       BRA     ERRGR
LB42 : SUBQ    #1,D3
       BPL     LB39
```

( ↑ 1 -- ) ↑ après

```
LB43:MOVE.L  A5,(A0)+      après la fonction
     MOVE    D2,(A0)+      nb d'arguments     ~~1 proc~~
     MOVE    (SP)+,(A0)+   appel de fonction  { 2 variable
                                              { 3 chaine
⊗    MOVE.L  A0,-4(A1)     nouveau tmproca  ←  { CLR.L (A0)+  arrêt des var locales ⊗
     MOVE.L  SP,TPILE      nouveau tpile       { MOVE.L A0,(A1)  tmproc
     MOVE.L  (A3),A5       nouveau A5
     BSR     APROC2 (3)55  sauvegarde V(5) [boucle for]
     BRA     IRUN          continue en A5
```