

## D- SPECIFICATIONS NANORESEAU

\*\*\*\*\*

## FROM BOITIER DE COMMUNICATION POUR M05 ou T07

(Informations pour développeurs)

Le NANORESEAU permet à un certain nombre (1 à 31) de postes de travail T07/M05/T0770 d'échanger et de partager les ressources d'un serveur (disques, imprimante ou autre service au fur et à mesure des développements et de l'émergence des besoins).

Ce document vise exclusivement le côté poste de travail : ce que doit maîtriser un développeur pour exploiter les ressources des couches de base du NANORESEAU.

## 1- INFORMATIONS SYSTEME AJOUTEES

Chaque poste possède quelques octets particuliers, chargés provisoirement lors de l'allumage et modifiables par le serveur.

## 1- Type d'ordinateur 1FF6/5FF6

Lors de l'allumage, chaque poste s'identifie :

T07 (0)  
M05 (1)  
T0770 (2)

Le serveur pourra recoder le poste, en particulier pour tenir compte de l'enrichissement en banques diverses.

Toujours conserver le bit 0 : 0 pour la famille T07/T0770,  
: 1 pour la famille M05  
ou alors ne pas utiliser l'ordre d'échange écran proposé dans la ROM (la caractéristique étant le codage des couleurs écran).

## 2- Application 1FF7/5FF7

-----

Lors de l'allumage, chaque poste fixe cette valeur à 0.

Lors de la phase d'initialisation par appel au serveur, celui-ci fournit le numéro de code de l'Application.

LEANORD maintiendra à jour une liste d'affectation de ces codes à la disposition des développeurs.

## 3- Les pointeurs en RAM

-----

Un certain nombre de services ne peuvent être fournis par la ROM : ceux qui dépendent de l'application et des enrichissements ajoutés à cette application, ou ceux qui sont liés aux enrichissements hard ajoutés aux postes (banques).

### CHPAGE 1FF8,9/5FF8,9 (changement de page)

Quelques pages sont définies de base, en particulier concernant la double banque de la page écran.

Tout appel à une page inconnue de la ROM fera appel à un programme pointé par CHPAGE.

### TRCT 1FFA,B/5FFA,B (Traitement de code Tâche)

Quelques codes Tâche sont exécutés par la ROM (ordres auto-exécutables, ordre d'affichage, etc...).

Tout code Tâche inconnu de la ROM renvoie à un traitement pointé par TRCT.

PIUTIL 1FFC,D/5FFC,D (Pointeur indirect utilisateur)

Il s'agit d'un service utilisable au gré du développeur.

Le transfert de plage mémoire à l'adresse X de la page 3 provoquera le calcul suivant :

- Quelle est l'adresse de (PIUTIL,+1):Résultat Y
- Quel est le contenu des 2 octets pointés par Y : Résultat Z
- L'adresse effective sera donc X + Z.

Dans ce service, Y et Z sont tous deux accessibles sans aucun autre changement de page.

Utilisation caractéristique : téléchargement de logiciel dans un poste sans connaître l'adresse de début du logiciel.

ACNTRC 1FFE,F/5FFE,F

Lorsqu'une liaison s'établit difficilement avec un poste, deux hypothèses :

- Reprendre la main par CONTROLE C,
- Attendre que le nombre maximum de répétitions autorisé est atteint.

Dans les 2 cas, le traitement est dérouté vers (ACNTRC).

Lors de l'initialisation, cette adresse renvoie vers le point d'entrée à froid de l'application après remise à zéro du Flag "présence de disque" (DKFLAG).

**LES ECHANGES ENTRE POSTES**

-----

Pour comprendre les échanges entre postes, il faut examiner d'abord le rôle de la zone consigne courante chez les deux partenaires.

Le plus simple est d'expliquer ce rôle lorsque les deux partenaires sont deux postes de travail. Côté serveur, il suffira d'émuler ce fonctionnement.

Un échange débute par l'envoi d'une trame consigne.

A partir de cet instant, l'un des deux postes (le DEMANDEUR) devient totalement passif (mise en ATTENTE) et l'autre (le RECEVEUR) prend toute l'initiative.

Il appartient donc à ce poste receveur d'exécuter le code Tâche reçu. L'exécution de ce code Tâche conduira :

- soit à charger chez le demandeur une plage mémoire,
- soit à recopier chez lui une plage mémoire du demandeur,
- soit à ordonner au demandeur d'exécuter des ordres élémentaires.

Il n'y a pas d'autre type d'activité envisageable dès l'instant où la gamme d'ordres élémentaires est enrichissable à souhait.

**ECHANGES DE PLAGES MEMOIRE (en émission ou en réception)**  
-----

Dans la trame consigne, il y a 5 octets déterminant la longueur et l'adresse d'implantation de la plage mémoire.

NBOT Haut et Bas - 2 octets : nombre d'octets de la trame

PAi Page de l'adresse d'implantation 1 octet

Ai Haut et Bas - 2 octets : adresse d'implantation dans la page

Un seul des deux postes gardant l'initiative, il suffit que celui-ci lance une invitation à émettre (GETDATA) ou une invitation à recevoir (SENDDATA) pour que le correspondant exécute cet ordre.

Une fois que l'ordre est exécuté, chaque poste modifie son Ai en ajoutant NBOT, ce qui permet de remplir séquentiellement des plages mémoires.

Le poste actif peut souhaiter reprogrammer les pointeurs du poste passif.

L'émission d'une nouvelle trame consigne viendra écraser les 5 octets et les remplacer par d'autres.

Eventuellement, ce nouveau re-formatage peut s'accompagner d'une demande d'exécuter une nouvelle consigne élémentaire.

Lorsque le poste actif a exécuté les diverses tâches correspondant à l'ordre reçu, il déconnecte le poste en attente et retourne à son activité.

**EXEMPLE D'APPLICATION****-----**  
**Ordre d'envoyer ECRAN**  
**\*\*\*\*\***

Le poste demandeur a envoyé cet ordre à un autre poste.  
 En préparant sa demande, le poste demandeur a préparé dans ses  
 pointeurs NBOT = 1F40 Page Ai = Page caractère Ecran  
 Ai = 0 (début d'écran)

Le receveur entre dans la routine d'EXECUTION d'ordre reçu.  
 Ses pointeurs sont alors chargés, identiques à ceux du demandeur.  
 Il lui suffit d'appeler la routine Imitation à recevoir pour  
 télécharger chez le demandeur 1F40 octets écrits à partir du  
 début d'écran.

**Reste la page couleur :**

Le receveur (actif) préparera sa consigne courante avec les  
 informations :

Ordre de ne rien faire  
 NBOT = 1F40 (inchangé)  
 PAi = page couleur écran  
 Ai = 0

puis appelle la routine CONSIGNE.

Ces informations se retrouvent chez le demandeur qui n'exécute  
 rien, conformément à l'ordre reçu.

Troisième étape : appel de la routine Imitation à recevoir qui  
 télécharge les 1F40 octets de la page couleur.

Le retour (RTS) de la routine d'exécution enchaînera  
 automatiquement l'ordre DECONNECTER.

Le poste demandeur sera alors satisfait et continuera son  
 travail.

Le poste sollicité sortira de l'interruption et reprendra sa  
 tâche normale.

## INITIATIVE D'UN ECHANGE

=====

Le mécanisme de l'échange ayant été expliqué, reste à indiquer comment prendre l'initiative d'un échange.

Le poste A souhaite envoyer un ordre tel que : ordre d'envoyer Ecran au poste B.

Le poste A écrit sa consigne dans une plage mémoire différente de sa consigne courante.

Appelons cette plage la consigne Original.

Cet original ne sera pas modifié par le réseau et sera réutilisé en cas d'échec de transmission (prendre garde à ce que l'exécution de l'ordre n'amène pas un écrasement de cette plage).

Une plage est ouverte aux développeurs en 1FA0/5FA0.

A l'appel de la routine RESEAU, l'index X doit pointer cette plage.

Par rapport à X, voici la signification de ces adresses :

X + 0 | Longueur sur un seul octet de la trame consigne  
 La prévoir tout de suite module 4.  
 Seule cette longueur modulo 4 sera transmise.

X + 1 | Destinataire

X + 2 | Code Tâche Réseau (avec certains gadgets)

X + 3 | Code Tâche Application (tout à fait disponible).

Les affectations de ces 2 codes Tâche sont distribuées par l'Université de Lille/ET/OU LEANORD pour éviter tout conflit ultérieur entre développeurs.

X + 4 | Nombre d'octets de la trame (haut)

X + 5 | Nombre d'octets de la trame (bas)

X + 6 | Page de l'adresse d'implantation

X + 7 | Adresse d'implantation (haut)

X + 8 | Adresse d'implantation (bas)

X + 9 | Type d'ordinateur (rempli par la ROM)

X + 10 | Type d'application (rempli par la ROM)

X + 11 | jusqu'à X + 59 au gré de l'application

Ceci étant rempli, vérifier que X est bien chargé puis faire appel à RESEAU.

Au retour, les bits NZVC du CCR seront chargés au gré des développeurs (rôle de FLRLCH côté T07 actif).

Seule configuration interdite : 1111

Tous les registres, sauf X et DP, sont modifiés par RESEAU.



## CODES TACHE RESEAU

\*\*\*\*\*

## CODES TACHE RESEAU recevables par IO7

\*\*\*\*\*

Quelques codes tâche réseau sont gérés dans la ROM.

| Code | Mnémonique | Signification   |
|------|------------|---|
| 0    | KNOOP      | consigne NON opération  |
| 1    | KOMIAT     | consigne Ordre de mise en attente<br>Le poste recevant cette consigne perd l'initiative et se bloque en attente. Le poste ayant émis cette consigne ne se bloque pas en attente. Tout ordre à l'initiative du serveur commence nécessairement par KOMIAT. |
| 2    | KOAUEX     | consigne Ordre auto-exécutable<br>L'exécution de cette consigne provoque un JMP à l'adresse DK (1F69/5F69) - achever par RTS.   |
| 3    | KAFIEC     | consigne Ordre d'afficher sur écran<br>L'exécution de cette consigne provoque l'affichage sur l'écran d'un buffer de codes ASCII pointé par Ai en page 0.   |
| 4    | KOEE       | consigne Ordre d'envoyer l'ECRAN<br>qui implique que les pointeurs à la demande soient :<br><br>NBOT = 1F40<br>PAi = 1<br>Ai = 0  |

|   |        |  |
|---|--------|--|
| 5 | KOEM   | <p>Consigne Ordre d'envoyer MEMOIRE</p> <p>La plage mémoire est définie par NBOT, PAi et Ai.</p> <p>Ces 6 codes Tâche réseau sont les seuls exécutables par la ROM. Si le T07/M05 reçoit un autre code, il appelle la routine pointée par TRCT (fait donc appel à un complément de logiciel en RAM).</p> <p>Lorsqu'une consigne est reçue par un poste, l'ordre est exécuté dès réception (avant de repasser en attente) si le bit 7 du CTR est à 0. Il est exécuté après la réception du DISCONNECT si le bit 7 de CTR est à 1.</p> |
| 6 | KRCPTR | <p>Consigne Recopie de Compte Rendu</p> <p>Lors de la réception de cet ordre, le T07/M05 va lire, à l'adresse PRCPTR (XFF4), soit SAB écrit en PRCPTR et SCB à l'adresse PRCPTR+1.</p> <p>Le T07/M05 va lire à l'adresse SABCD et trouve le nombre N d'octets à recopier. Il recopie alors N octets depuis DK (XF69) vers SABCE (SABCD+1) et suivants.</p> <p>N doit être pair.</p>  |

Tout enrichissement du CTR est géré par LEANORD pour éviter les développements incompatibles.

### CODES TACHE RESEAU EN DIRECTION DU SERVEUR

-----

Le code Tâche Réseau 0 (KNOOP) est envoyé au serveur lors de l'initialisation. C'est une demande d'initialisation. KOMIAT est non recevable par le serveur. Les applications peuvent décider d'utiliser KOAUEX, KAFIEC ou KOEE moyennant un enrichissement correspondant du serveur.

D'autres codes Tâche Réseau sont spécifiques au serveur.

|   |        |  |
|---|--------|--|
| 8 | KODOST | Ordre Emulation du disque THOMSON (DKCONT)           |
| 9 | KODOSS | Ordre DOS serveur directement exploitable par le DOS |

Tout enrichissement du CTR est géré par LEANORD pour éviter les développements incompatibles.

Deux options Bit 7 = 1 --> demande de libération immédiate  
 Bit 6 = 1 --> demande de décharger buffer si possible instantanément.

### CODES TACHES APPLICATION

-----

Cet octet peut servir de paramètre quelconque pour les échanges entre T07.

Pour un échange avec le serveur, le code Tâche application indique dans la famille CTR quelle tâche est demandée.

Par exemple : en KODOSS, GTA = code de la fonction appelée.

## LES POINTS D'ENTREE DE LA ROM

-----

|  |   |
|--|---|
| DKCONS )<br>DKBOOT )<br>DKFORM )   | )<br>) totalement émulés du DOS<br>)  |
| LECFA )<br>RECFI )<br>RECUP )<br>ECRSE )<br>ALLOD )<br>ALLOB )<br>MAJCL )<br>FINTR ) | )<br>)<br>)<br>) logiciels intégrés du MINIDOS<br>)<br>)<br>)<br>)  |
| RESEAU   | voir information ci-dessus  |
| PRLGN  | lorsqu'un T07 assure des réponses différées à une requête, rappeler cette routine pour répondre au serveur.<br>Procédure de prise de ligne.   |
| RELACH   | Lorsqu'un T07 diffère une réponse, il doit relâcher la ligne.<br>Passer par RELACH qui autorise le fonctionnement normal de PRLGN.<br>Mettre VEROU # 0 pour éviter qu'un poste ne vienne modifier la consigne courante.                                       |
| GETDAT   | L'appel de cette routine provoque la réception de la plage mémoire programmée (NBOT, PAi, Ai), le correspondant étant en attente.<br>Aucun autre paramètre. Aucune protection.<br>Si échec, neutralisation du contrôleur, bit C - 1<br>Protège DP uniquement. |
| SENDAT   | Idem que ci-dessus sauf émission de la plage mémoire programmée vers le correspondant en attente.<br>Protège DP uniquement.   |
| DESEL  | Provoque la relâche d'attente du destinataire.<br>Protège DP uniquement.  |
| EMAP   | Provoque l'injection d'une nouvelle consigne dans la consigne courante du correspondant en attente et l'exécution de cette consigne.<br>Protège DP uniquement.  |
| RSTIRQ   | Routine de réinitialisation à rappeler à chaque RESET.<br>Provoque entre autres tâches la demande d'initialisation au serveur.  |

OEE | L'appel de cette routine provoque la demande complète  
 d'émission de l'écran du poste dont le numéro est  
 dans le registre A.

RAM bas d'ECRAN  
 -----

Début écran + 1F40 = début de zone disponible

1F40 - 1F4F = réservé LOGO ?

1F50 - 1F5C = Variables provisoires Réseau régénérées à  
 chaque échange réseau

1F5D = Octet contenant 0 (octet haut de longueur  
 Consigne Courante)

1F5E = Longueur Consigne Courante

1F5F = Destinataire (recopie provisoire de  
 l'original)

-----  
 1F60 (0) = CODE Tâche RESEAU CTR  
 1F61 (1) = CODE Tâche APPLICATION CTA  
 1F62 (2) = Nombre d'octets de la trame NBOT Haut  
 1F63 (3) = Nombre d'octets de la trame NBOT Bas  
 -----

1F64 (4) = Page d'implantation PAi  
 1F65 (5) = Adresse d'implantation Ai Haut  
 1F66 (6) = Adresse d'implantation Ai Bas  
 1F67 (7) = Type d'ordinateur (T07/M05/T070) ORD  
 -----

1F68 (8) = APPLICATION APPLI (BASIC, LOGO,...)  
 recopier cette zone pour compte rendu de  
 status dans la zone de l'application

1F69 (9) = DK en principe libre selon applications

1F6A (10) = NOM1

1F6B (11) = NOM2  
 -----

1F6C (12) = NOM3

1F6D (13) = NOM4  
 -----

etc....

1F9B (59) = limite pratique 15 paquets de 4

1F9F (63) = limite max. théorique

**CONSIGNE ORIGINALE utilisable par tout développeur**  
 -----

|         |               |  |             |
|---------|---------------|--|-------------|
| X + 0   | 1FA0 (-2)     | LONGUEUR ORLGCS<br>c'est la mémoire pointée par l'adresse X<br>lors de l'appel de RESEAU |             |
| X + 1   | 1FA1 (-1)     | Destinataire   | ORDST       |
| X + 2   | 1FA2 (0)      | Code Tâche Réseau  | ORCTR       |
| X + 3   | 1FA3 (1)      | Code Tâche application   | ORCTA       |
| X + 4   | 1FA4 (2)      | Nombre d'octets de la trame  | ORNBOT Haut |
| X + 5   | 1FA5 (3)      | Nombre d'octets de la trame  | ORNBOT Bas  |
| X + 6   | 1FA6 (4)      | Page d'implantation  | ORPAi       |
| X + 7   | 1FA7 (5)      | Adresse d'implantation   | ORAI Haut   |
| X + 8   | 1FA8 (6)      | Adresse d'implantation   | ORAI Bas    |
| X + 9   | 1FA9 (7)      | Type d'ordinateur  | ORORD       |
| X + 10  | 1FAA (8)      | Application  | ORAPPL      |
| X + 11  | 1FAB (9)      | Libre à l'application  | ORDK        |
| X + 12  | 1FAC (10)     | ORNOM1   |             |
| X + 13  | 1FAD (11)     | ORNOM2   |             |
| X + 14  | 1FAE (12)     | ORNOM3   |             |
| X + 15  | 1FAF (13)     | ORNOM4   |             |
|         | etc...        |  |             |
|         | 1FDD (59)     | Limite pratique 15 paquets de 4  |             |
|         | 1FEO (63)     | Limite théorique   |             |
| jusqu'à | 1FF5          | Disponibles  |             |
|         | 1FF6 ==> 1FFF | POINTEURS de compléments de logiciels<br>(TYPOR/APPLI) CHPAGE TRCTA PLOGI<br>ACNTRC      |             |

```

0001 TYPORD SET 1 ;0=TOP, 1=MOS
0001 MOS SET TYPORD
0000 TOP SET 1-MOS

0000 IF MOS
A000 EQU $A000 ;MOS EN ROM
0000 ELSE
0000 EQU $E000 ;TOP EN ROM
0000 ENDIF

;TYPORD=TYPE DE L'ORDINATEUR
;=0 SI TOP, 1 SI MOS, 2 SI TOP0, ETC
;PAIR SI TYPE TOP : XX 3BITS FOND, 3BITS CARACTERE
;IMPAIR SI TYPE MOS : 4BITS FOND, 4 BITS CARACTERE

```

COMMENTAIRES ET EQUATES

```

0000 IF MOS
0000 ELSE
0000 BITTL
0000 ENDIF

```

TOP EQUATES ET COMMENTAIRES CONTROLEUR TOP"

VARIABLES SYSTEME

\*\*\*\*\*

APPLICATION

```

0000 IF MOS
EF00 $EF00 ;IDENTIFICATION APPLICATION
2000 MONI EQU $2000 ;PAGE 0 DU MONITEUR
EF00 TIERST EQU $E000 ;POINTEUR RESET TIEDE
EF00 FRST EQU $E000 ;POINTEUR RESET FROID
0000 ELSE
0000 IDAPPL EQU 0
0000 MONI EQU $6000
0000 TIERST EQU $1C
0000 FRST EQU $001E
0000 ENDIF

; PAGE ZERO DU SYSTEME

```

NAME RESEAU LISTL le 12 Mars 1985  
 TOO EQUATES ET COMMENTAIRES CONTROLEUR MOS

```

0000      IF      MOS
0020      DPMON EQU $20      ;DIRECT PAGE DU MONITEUR
0030      TIMER EQU $2031   ;CHRONOMETRE CURSEUR
0031      TIMERB EQU        ;
0064      IRGPT EQU $2064   ;VECTEUR D'INTERRUPTION
      ELSE
0000      DPMON EQU $60
0000      IRGPT EQU $6021
      ENDIF
2048      DISQUE EQU MONI+$48 ;DEBUT DE PLAGE DISQUE
2058      DISFIN EQU MONI+$58 ;FIN DE PLAGE DISQUE
2080      DKFL6 EQU MONI+$80  ;FLAG DE PRESENCE DE DISQUE
      PLAGE DISQUE
      DKOPC EQU DISQUE ;CODE OPERATION
2048      DKDRV EQU DISQUE+1 ;DRIVE
2049      DKTRK EQU DISQUE+2 ;TRACK=PISTE ; 2 OCTETS
204A      DKSEC EQU DISQUE+4 ;SECTEUR
204C      DKNUM EQU DISQUE+5 ;FACTEUR D'ENTRELACEMENT
204E      DKSTA EQU DISQUE+6 ;STATUS
204F      DKBLF EQU DISQUE+7 ;BUFFER DE DISQUE
      PERIPHERIQUES
      DECRAN EQU 0 ;DEBUT DE L'ECRAN
0000      PORTC EQU $A7C0 ;COMPUTATION PAGE ECRAN + COULEUR TOUR
0000      PORTCS EQU $A7C1 ;PORT SON
0001      SON EQU 1 ;BIT 0 SUR MOS
001E      MAS611 EQU X00011110
00E1      MAS800 EQU X11100001
0000      ADLC EQU $A7D0 ;CONTROLEUR DE COMMUNICATION (PUIS A7DB)
00A7      DPADLC EQU $A7 ;
0000      NUPCST EQU $A7DB ;NUMERO DE POSTE PAR SWITCHES (5 BITS)
      ELSE
0000      DECRAN EQU $4000 ;DEBUT DE L'ECRAN
0000      PORTC EQU $E7C3 ;COMPUTATION PAGE ECRAN + COULEUR TOUR
0000      PORTCS EQU $E7C1 ;PORTC SON
0000      SON EQU 8
0000      FRB EQU $E7C9 ;POUR BANQUES T07 70
0000      CRB EQU $E7CB
    
```



le 12 Mars 1985  
 TOO EQUATES ET COMMENTAIRES CONTROLLEUR MOS

```

0000 MASQ11 EQU X01110000
0000 MASQ00 EQU X10001111
0000 ADLC EQU $ETDD ;CONTROLEUR DE COMMUNICATION (PUIS A7DB)
0000 DPADLC EQU $E7
0000 TIMER EQU $E7C6 ;NBRE ALERATOIREPOUR DELAIS
0000 TIMERB EQU $E7C7 ;OCTET BAS DU TIMER
0000 NUPOST EQU $E7DB ;NUMERO DE POSTE PAR SWITCHES (5 BITS) (PUIS ATDO)
0000 ENDIF
  
```

POINTS D'ENTREE DU MONITEUR

```

0000 IF MOS
0002 : ENTREE PAR SW1, SUIVI DE CODE
000C SPUTC EQU 2 ;JSR PUTC ECRITURE SUR ECRAN
000A SBTST EQU $C ;JSR KEYTEST : LECTURE RAPIDE CLAVIER
0000 SGETC EQU $A ;JSR GET CARACTERE /CLAVIER
ELSE
0000 PUTC EQU $E803 ;ECRITURE SUR ECRAN
0000 GETC EQU $E806 ;ACQUISITION CARACTERE CLAVIER
0000 KTST EQU $E809 ;SURVEILLANCE RAPIDE CLAVIER
0000 ENDIF
  
```

0007 BEEP EQU 7 ;CODE ASCII DE BEEP

MAP ECRAN

```

1F40 RAMECR EQU DECRAN+$1F40
1F54 ORG RAMECR+$14 ;LOGO UTILISE LES 16 IERS OCTETS
1F54 ANUTRA DS 1 ;ANCIEN NUMERO DE TRAME
1F55 AAL DS 2 ;ANCIENNE ADRESSE D'IMPLANTATION
1F57 NUTR DS 1 ;NUMERO DE TRAME
1F58 DELA DS 1 ;FLAG DE 1ERE RECEPTION (0) OU NON (1)
1F59 RSPT DS 1 ;FLAG DE RESPONSABILITE (1 SI INITIATIVE)
1F5A : TESTE NON DESTRUCTION PAR MESSAGE ENTRANT
1F5A OPTREP DS 1 ;COMPTEUR DE REPETITION SI ERREUR
0006 MAXREP EQU 6 ;NBRE MAX DE REPETITIONS
1F5B RYSTAT DS 1 ;STATUT DE RECEPTION POUR DISC.
1F5C FLRUCH DS 1 ;FLAG RELACHE POUR STATUS ECHANGE RESEAU
  
```

TRAME CONSIGNE COURANTE

```

DST VOIR PAGE 0 MONITEUR
MC VOIR PAGE 0 MONITEUR
  
```

100 EQUATES ET COMMENTAIRES CONTROLER MOS

|      |        |      |               |  |
|------|--------|------|---------------|--|
| 1F5D | LGCSG  | DS   | 1             | :LONGUEUR CONSIGNE SUR 2 OCTETS                            |
| 1F5E |        |      |               | POUR CHARGEMENT DE X . METTRE 0                            |
| 1F5E | CSCOUR | DS   | 1             | :ADRESSE CONSIGNE COURANTE                                 |
|      |        |      |               | =NBRE D'OCTETS DE LA CONSIGNE                              |
| 1F5F | DSTI   | DS   | 1             | :DESTINATAIRE  |
| 1F60 | CTR    | DS   | 1             | :CODE TACHE RESEAU   |
| 1F61 | CTA    | DS   | 1             | :CODE TACHE APPLICATION                                    |
| 1F62 | NBOT   | DS   | 2             | :NBRE D'OCTETS DU MESSAGE                                  |
| 1F64 | PAI    | DS   | 1             | :PAGE ADRESSE D'IMPLANTATION                               |
| 1F65 | AI     | DS   | 2             | :ADRESSE D'IMPLANTATION DANS LA PAGE                       |
| 1F67 | ORD    | DS   | 1             | :TYPE D'ORDINATEUR   |
|      |        |      |               | 0 SI T07   |
|      |        |      |               | 1 SI MOS   |
|      |        |      |               | 2 SI T07.70  |
|      |        |      |               | ETC...   |
| 1F68 | APPLI  | DS   | 1             | :CODE DE L'APPLICATION                                     |
| 1F69 | DK     | DS   | 1             | :DISQUE  |
| 1F6A | SUICSG | DS   | 54            | :AU TOTAL, 64 OCTETS +2 MAX POUR CSG                       |
|      |        |      |               |  |
| 1FA0 | ORLGS  | DS   | 1             | :CONSIGNE ORIGINAL   |
| 1FA0 | IX     | EGU  | ORLGS         |  |
| 1FA1 | ORDST  | DS   | 1             | :DESTINATAIRE  |
| 1FA2 | ORCTR  | DS   | 1             | :CODE TACHE RESEAU   |
| 1FA3 | ORCTA  | DS   | 1             |  |
| 1FA4 | ORNBOT | DS   | 2             |  |
| 1FA6 | ORPAI  | DS   | 1             |  |
| 1FA7 | ORAI   | DS   | 2             |  |
| 1FA9 | ORORD  | DS   | 1             | :EN PRINCIPES REMPLI PAR RESEA                             |
| 1FA9 | ORU    | EGU  | ORORD         |  |
| 1FAA | ORAPPL | DS   | 1             |  |
| 1FAB | ORDK   | DS   | 1             | :EN PRINCIPES, DISQUE                                      |
|      |        |      |               |  |
|      |        |      |               | : POINTEURS SYSTEME RESEAU                                 |
|      |        |      |               | REMPLIS LORS DE L'INITIALISATION                           |
|      |        |      |               |  |
| 1FF  | MAXECR | EGU  | DECRAN+\$1FFF | :VERIFIER QUE ACNTRC = 1FFE                                |
| 1FF1 |        | ORIG | MAXECR-\$E    |  |
|      |        |      |               |  |
| 1FF1 | PGRST  | DS   | 2             | :RESET DE PAGE POUR T07 70                                 |
| 1FF3 | FLGPAG | DS   | 1             | :FLAG DE PAGE POUR T07 70                                  |
| 1FF4 | PROPTR | DS   | 2             | :POINTEUR RECOPIE COMPTE RENDU                             |
| 1FFG | TYPOR  | DS   | 1             | :TYPE D'ORDINATEUR   |
| 1FF7 | APPLIC | DS   | 1             | :TYPE D'APPLICATION (BASIC, LOGO...)                       |
| 1FF8 | CHPAGE | DS   | 2             | :TRAITEMENT DE CHGT DE PAGE                                |
| 1FFA | TRCT   | DS   | 2             | :TRAITEMENT CODE TACHE (EN PAGE 0)                         |
| 1FFC | PIUTTL | DS   | 2             | :ADRESSE DU POINTEUR DE DEBUT DE LOGICIEL DE L'APPLICATION |
| 1FFE | ACNTRC | DS   | 2             | :ADRESSE D'EXECUTION CNT C)                                |

NRNQ RESEAU USTL

le 12 Mars 1985  
 100 EQUATES ET COMMENTAIRES CONTROLLEUR MOS

```

2052      ORG  DISQUE + 10      ;XX51 RESTE DISPONIBLE
;
2052      NUPO  DS      1      ;RECOPIE DE NUJPR A CHAQUE RESET
COP2    DS      1      ;COPIE CR2 (SI INJECTE HORLOGE)
2053      MCRX  DS      1      ;MOT DE CONTROLE DE TRAME EN RECEPTION
2054      DSTTX DS      1      ;DESTINATAIRE EN EMISSION
2055      ; SEULEMENT CHARGE PAR IRQ OU DANS RESEA = CORRESPONDANT
; = EXPEDITEUR EN RECEPTION
2056      MCTX  DS      1      ;MOT DE CONTROLE EN RECEPTION
2057      OPTRES DS      1      ;COMPTEUR DE REPETITION D'APPEL RESEAU
2058      VEROU DS      1      ;AUTORISATION D'IRQ
  
```

CONTROLLEUR DE COMMUNICATION

\*\*\*\*\*

```

A7D0      CR1      EQU      ADLC      ;CONTROLE REGISTRE
A7D1      CR2      EQU      CR1+1
A7D3      CR4      EQU      CR1+3
;
A7D0      SR1      EQU      CR1      ;STATUS REGISTRE
A7D1      SR2      EQU      CR1+1
;
A7D2      TXFIFO  EQU      CR1+2
A7D2      RXFIFO  EQU      CR1+2
  
```

INFORMATIONS DIVERSES

\*\*\*\*\*

```

;
; PAGES DANS LE TO7
;
; PERMETTENT LA COMMUTATION DE BANQUE DANS LE TO7
; ET LES ADRESSAGES RELATIFS (UTIL)
;
; BANQUES TO7, MOS ET TO770
;
; PAGE 0 = EN ADRESSE ABSOLUE, SANS SELECTION DE BANQUE
; EN FAIT, PAGE CARACTERE ECRAN SELECTIONNEE
; PAGE 1 = PAGE CARACTERE ECRAN
; ATTENTION ADRESSE AI AJOUTEE A L'ADRESSE BAS D'ECRAN
; PAGE 2 = PAGE COULEUR ECRAN
; ATTENTION ADRESSE AI AJOUTEE A L'ADRESSE BAS D'ECRAN
; PAGE 3 = PAGE UTILISATEUR : AJOUTER A AI LE CONTENU DE L'ADRESSE
; POINTEE PAR PIUTIL
; AUTRES PAGES UTILISABLES : POINTEUR DU TRAITEMENT (CHPAGE)
;
;
  
```

MESSAGE EMIS PAR UN T07 A L'INITIALISATION

C'EST UNE TRAME COMMANDE DONT LES 32 OCTETS  
 A PARTIR DE DK SONT LA RECOPIE DES 32 OCTETS D'IDENTIFICATION  
 DE LA ROM APPLICATION  
 ORD=0 SI T07, 1 SI MOS, 2 SI T07/70  
 APPLI = 0, A REDEFINIR PAR LE SERVEUR

CODES TACHE RESEAU

LES CODES TACHE SONT - SOIT DES CODES TACHE MACRO  
 (EMISES PAR T07 DEMANDEUR)  
 - SOIT DES CODES TACHE ELEMENTAIRES  
 (EMISES VERS T07 EN ATTENTE)

CONSIGNES SOUS ATTENTE :

- RIEN
- EXECUTER LE CODE TACHE APPLICATION
- AFFICHER SUR ECRAN
- AUTO-EXECUTABLE
- RECOPIER LE COMPTE RENDU

OPTION

- BIT 7 = 1 : EXECUTER APRES DISCONNECT
- BIT 7 = 0 : EXECUTER DES RETOUR EN ATTENTE

REQUETES

- RIEN = DEMANDE D'INITIALISATION SI DST=SERVEUR
- EXECUTER LE CODE TACHE APPLICATION
- ORDRE D'ENVOYER L'ECRAN
- ORDRE D'ENVOYER MEMOIRE
- ORDRE DOS THOMSON
- ORDRE DOS SERVEUR

- DEUX OPTIONS POUR CES MACRO CONSIGNES
- (SOUS RESERVE D'UNE IMPLEMENTATION DANS LE SERVEUR)
- DEMANDE DE LIBERATION IMMEDIATE (BIT 7 = 1)
- T07 PRET POUR VASY-EM (BIT 6 = 1)

|      |        |     |   |                                    |
|------|--------|-----|---|------------------------------------|
| 0000 | KNOOP  | EQU | 0 | : PAS D'OPERATION                  |
| 0001 | KOMIAT | EQU | 1 | : ORDRE DE MISE EN ATTENTE         |
| 0002 | KORUEX | EQU | 2 | : ORDRE AUTO-EXECUTABLE            |
|      |        |     |   | LE LOGICIEL EN LM DEMARRANT A DK   |
| 0003 | KAFIEC | EQU | 3 | : ORDRE D'AFFICHER SUR ECRAN       |
|      |        |     |   | LE TEXTE POINTE PAR AI (EN PAGE 0) |
| 0004 | KOEE   | EQU | 4 | : ORDRE D'ENVOYER L'ECRAN          |
| 0005 | KOEM   | EQU | 5 | : ORDRE D'ENVOYER MEMOIRE          |
| 0006 | KRCPTR | EQU | 6 | : ORDRE DE RECOPIER COMPTE RENDU   |

: TOUT CODE SUPERIEUR = ALLER A [TRCT] SI RECU PAR T07

: CONSIGNES GLOBALES RECEVABLES PAR SERVEUR

```

0007      0      :DEMANDE D'INITIALISATION
0008      7      :ORDRE BASIC 1
0009      8      :ORDRE DOS THOMSON
000A      9      :ORDRE DOS SERVEUR
000B     10      :ORDRE MINIDOS
000C     11      :ORDRE SPOOL
000D     12      :ORDRE DATE
000E     13      :TRAITEMENT A DISTANCE
           14      :CONSOLE SERVEUR
    
```

: MOTS DE CONTROLE

```

0080      MOUR      EQU      $10000000      :VASY RECEIS
0090      MCPCH     EQU      $10010000      :PRISE EN CHARGE
00A0      MCAMA     EQU      $10100000      :AVIS DE MISE EN ATTENTE
00B0      MCVE      EQU      $10110000      :VASY EMETS
00C0      MCDISC    EQU      $11000000      :DECONNECTE
00D0      MCAPA     EQU      $11010000      :APPEL SOUS ATTENTE (TEST TXAP)
00E0      MCK       EQU      $11100000      :OK
00F0      MCLRA     EQU      $11100000      :UA (OK DE DISCONNECT)
           MCAPI     EQU      $11110000      :APPEL INITIAL
    
```

ORDRES AUTO-EXECUTABLES  
ET TRAITEMENT DE CODE TACHE APPLICATION

+ CE SONT DES SUBROUTINES. DONC A ACHEVER AVEC RTS  
+ AU RETOUR. CES TROIS REGISTRES U, Y, DP DOIVENT ETRE DANS LE MEME ETAT.

TRAME CONSIGNE ORIGINALE

```

: X INITIAL POINTE SUR LONGUEUR      TRAME DKCONS
: A0      X+0      LONGUEUR          16 OCTETS
: A1      X+1      DST                0 = SERVEUR
: A2      X+2      CODE TACHE RESEAU  KODOST
: A3      X+3      CODE TACHE APPLICATION  DKDFC
: A4      X+4,5    NRE D'OCTETS PAR TRAME $0080 OCTETS
: A6      X+6      PAGE ADRESSE IMPLANTION 0 = PAGE 0
: A7      X+7,8    ADRESSE IMPLANTION     DKBUF
: A9      X+9      TYPE D'ORDINATEUR (REMPLE PAR ROM)  TYPOR
: AA      X+10     APPLICATION (REMPLE PAR ROM)      APPLIC
    
```

```

:AB X+11 (DISQUE)
:AC X+12 (NOM 1)
:AD X+13 (NOM 2)
:AE X+14 (NOM 3)
:AF X+15 (NOM 4)
:BO X+16 (NOM 5)
:B1 X+17 (NOM 6)
:B2 X+18 (NOM 7)
:B3 X+19 (NOM 8)
:B4 X+20 (EXT 1)
:BS X+21 (EXT 2)
:B6 X+22 (EXT 3)

```

```

DKDRV
DKTRK HAUT
DKTRK BAS
DKSEC
X
X

```

\*\*\*\*\*  
DESCRIPTIF DES POINTS D'ENTREE RESEAU

POINT D'ENTREE RESEAU ADRESSE ROM + \$25

AVANT

```

: IL EXISTE DANS LE TOD UNE ZONE POINTEE PAR LE REGISTRE X
: ET FOURNISSANT L'ORIGINAL DE LA TRAME CONSIGNE SELON LE FORMAT
: UTILISE PLUS HAUT (A PARTIR DE ORLGCS) ; CETTE TRAME CONSIGNE
: NE DOIT PAS EXCEDER 60 OCTETS

```

ACTION DE LA ROUTINE RESEAU

```

: LE POSTE DEMANDEUR FOURNIT SA CONSIGNE, PUIS SE BLOQUE EN
: ATTENTE (SAUF CONSIGNE KOMIAT).

```

```

: LE POSTE SOLICITE A TOUTE LIBERTE POUR AGIR SUR LE POSTE
: DEMANDEUR EN UTILISANT TOUT ENCHAÎNEMENT DES 3 ORDRES :

```

- CONSIGNE = EMAP
- VASY EMETS = EMVE
- VASY REDDIS = EMVR

```

: PUIS RELACHE LE POSTE DEMANDEUR PAR L'ORDRE DECONNECTE. = EMDISC

```

```

: EN CAS D'INCIDENT LIE AUX ECHANGES RESEAU, LE POSTE SOLICITE
: ENVOIE UN ORDRE "DECONNECTE + DEMANDE DE REEMISSON". LE TOD DEMANDEUR
: REPREND ALORS TOUTE LA PROCEDURE D'ECHANGE EN REPARTANT DE LA TRAME
: CONSIGNE ORIGINALE QUI EST RESTEE INCHANGEE.

```

APRES

: A LA FIN DE CETTE CONSIGNE, UTILISATION POSSIBLE DES FLAGS DU CCR

TOO EQUATES ET COMMENTAIRES CONTROLLEUR MOS

: POUR INVITER PAR EXEMPLE A LIRE LE STATUS (CF DOS THOMSON)  
 : CHARGER FLRLCH AVANT D'ENVOYER DISCONNECT  
 : NE PAS UTILISER LES 4 BITS HAUTS DE FLRLCH, NI LA CONFIGURATION OF

: CAS DE LA CONSIGNE : ORDRE DE MISE EN ATTENTE

:  
 :  
 : SI CETTE CONSIGNE EST EMISE, LE POSTE DEMANDEUR EST  
 : LIBERE ET LE CORRESPONDANT EST BLOQUE EN ATTENTE.  
 : LE SERVEUR NE PEUT PAS ETRE MIS EN ATTENTE.

: POINT D'ENTREE EMETS

:  
 : DEUX CAS D'UTILISATION :  
 : - SUITE A L'EMISSION D'UN ORDRE DE MISE EN ATTENTE.  
 : - RECEPTION D'UN ORDRE RESEAU (SOUS IR0) FOURNISSANT UNE CONSIGNE INCONNUE  
 : TRANSMIS A UN COMPLEMENT DE LOGICIEL PAR LE POINTEUR TRCT.  
 : LES DEVELOPPEURS DEVRONT ALORS ORGANISER LES ECHANGES A L'AIDE DES 4  
 : OPTIONS D'EMETS.

: 4 OPTIONS A EMETS : = 4 POINTS D'ENTREE

: EMAP = CONSIGNE (SOUS ATTENTE)

: EMVE = VASY EMETS

: EMVR = VASY RECEITS

: EMDISC = DECONNECTE

: LES ROUTINES EMETS NE PROTEGENT AUCUN REGISTRE (SAUF DP).  
 : ELLES EXECUTENT L'ORDRE DONNE, REPETENT PLUSIEURS FOIS L'ORDRE SI INCIDENT,  
 : SORTIENT AVEC LE BIT C LEVE EN CAS D'ERREUR FATALE

: EMAP EMETS LA CONSIGNE COURANTE (= LA CONSIGNE RECUE).

: CHARGER CSOUR AVEC LE NOMBRE D'OCETS DE LA CONSIGNE

: LE NOM DU CORRESPONDANT (NECESSAIREMMENT EN ATTENTE) EST

: CHARGE AUTOMATICQUEMMENT PAR LA ROUTINE A EMETTRE.

: ATTENTION LONGUEUR MODULO 4 TRANSMISE EFFECTIVEMENT.

: DANS LE CAS OU PLUS DE 150MS SE PASSERAIENT

: ENTRE LA FIN D'UNE ROUTINE EMETS ET LE DEBUT DE LA SUIVANTE,

: PROCEDER AINSI :

: - RELACH

: - ACTIVITE EXTERNE, PREPARATION DE EMETS SUIVANT

: - PRLGN

: -EMETS \*...\*

:\*\*\*\*\*

NAND RESEAU USTL  
T00 EQUATES ET COMMENTAIRES CONTROLLEUR M05

DEBUT DE LA ROM RESEAU  
\*\*\*\*\*

A000 ORG ROM

LES 16 PREMIERS OCTETS

|       |          |         |             |                           |
|-------|----------|---------|-------------|---------------------------|
| A000  | 5245432F | DB      | 'REC', \$2F | : IDENTIFICATION RESEAU   |
| UA004 | 7E0000   | DKCON:  | JMP DKCONS  | : PE CONSIGNE DISQUE      |
| UA007 | 7E0000   | DKBOOT: | JMP DKBOOT  | : PE INITIALISATION BASIC |
| UA00A | 7E0000   | DKFM:   | JMP DKFORM  | : PE FORMATAGE DE DISQUE  |
| UA00D | 7E0000   | LECF:   | JMP LECHA   |                           |
| UA010 | 7E0000   | RECF:   | JMP RECFI   |                           |
| UA013 | 7E0000   | RECU:   | JMP RECUJ   |                           |
| UA016 | 7E0000   | ECRS:   | JMP ECORSE  |                           |
| UA019 | 7E0000   | ALLD:   | JMP ALLOD   |                           |
| UA01C | 7E0000   | ALLB:   | JMP ALLB    |                           |
| UA01F | 7E0000   | MAJC:   | JMP MAJCL   |                           |
| UA022 | 7E0000   | FINTR:  | JMP FINTR   |                           |

|       |        |         |            |   |
|-------|--------|---------|------------|---|
| UA025 | 7E0000 | RESEAU: | JMP RESO   | : EXECUTER ORDRE RESEAU (INITIATIVE, T07)       |
| UA028 | 7E0000 | PRLGN:  | JMP PRLGN  | : PROCEDURE DE PRISE DE LIGNE                   |
| UA02B | 7E0000 | RELACH: | JMP MARIRE | : RELACHE                                       |
| A02E  | C6B0   | EMVE:   | JMP #MCVE  | : PE VASY ENVOIE                                |
| A030  | 8C     | DB      | LDB \$BC   | : SAUTE 2 OCTETS                                |
| A031  | C6B0   | EMVR:   | LDB #MCVR  | : PE VASY RECEOIS                               |
| A033  | 8C     | DB      | LDB \$BC   |   |
| A034  | C6C0   | EMDISC: | LDB #MDISC | : PE DECONNECTE                                 |
| A036  | B1     | DB      | LDB \$B1   |   |
| A037  | 5F     | EMAP:   | CLRB TX    | : PE APPEL A POSTE SOUS ATTENTE                 |
| UA03B | 7E0000 | RSTIR:  | JMP RSTIR  | : RE-INITIALISATION RESET (VEROU ID)            |
| UA03E | 7E0000 | QEER:   | JMP QEER   | : ORDRE D'ENVOYER L'ECRAN (A VARIABLE D'ENTREE) |

0000 END