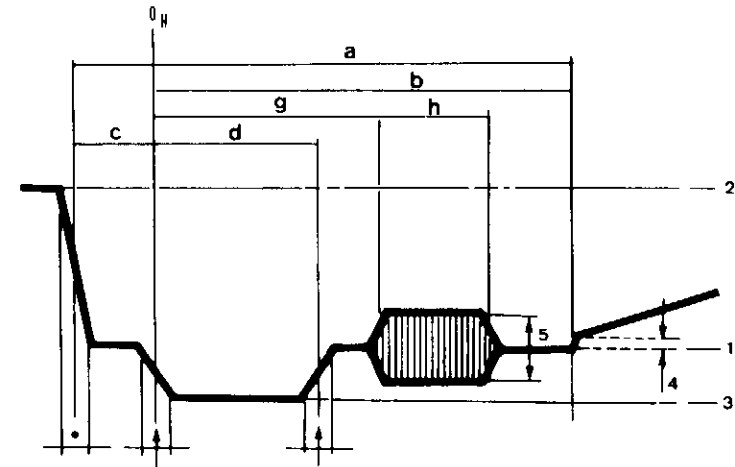


Designation du signal	Valeur d'adaptation	Numéro du contact	Mis au potentiel de référence.
Suppression (Commutation rapide)	0 à +0,4V état logique "zéro" (6) -1V à +3V état logique "un" (6) Impédance de charge 75Ω (1) (7)	16	
Masse suppression		18	
Bus de données pour l'intercommunication N° 1	Contact non utilisé (utilisation future à l'étude)	12	
Bus de données pour l'intercommunication N° 2	Contact non utilisé (utilisation future à l'étude)	10	
Masse commune des bus de données pour l'intercommunication		14	
Blindage de la fiche		21	Conditions de mesure et observation

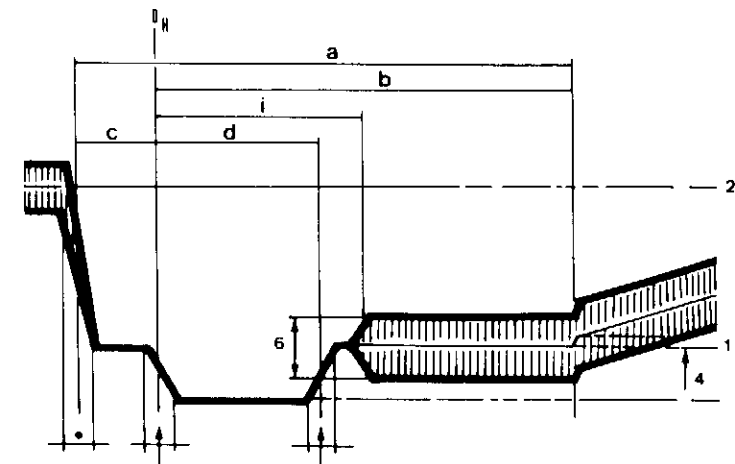
NOTES

- (1) L'existence de divers modes de fonctionnement des circuits audio (mono, stereo, voies séparées) peut exiger des commutations appropriées dans certains appareils.
- (2) Pour des fréquences supérieures ou égales à 20 Hz.
- (3) Voir Fig. 1a et 1b. Pour les systèmes de télévision à modulation vidéo positive, la tolérance peut être portée à -3, -6 dB.
- (4) Les tensions spécifiées pour l'entrée et la sortie vidéo ou une adaptation d'impédance est requise doivent s'entendre comme suit : tension appliquée à une charge de mesure fictive.
- (5) Pour des signaux analogiques en mode différentiel l'écart entre deux composantes quelconques ne doit pas dépasser $\pm \theta$ blanc.
- (6) L'état logique "un" correspond à la suppression.
- (7) La bande passante et les temps de transit doivent être adaptés aux signaux R, Y, B.

Niveaux dans le signal composite et détails des signaux de synchronisations de ligne



a) Systèmes NTSC en PAL



b) Système SECAM

3.5 Le clavier

Le clavier du TO7 est du type clavier à membrane souple, matriciel à 8 lignes et 8 colonnes.

On se reportera pour l'étude de son fonctionnement au chapitre 2.6.

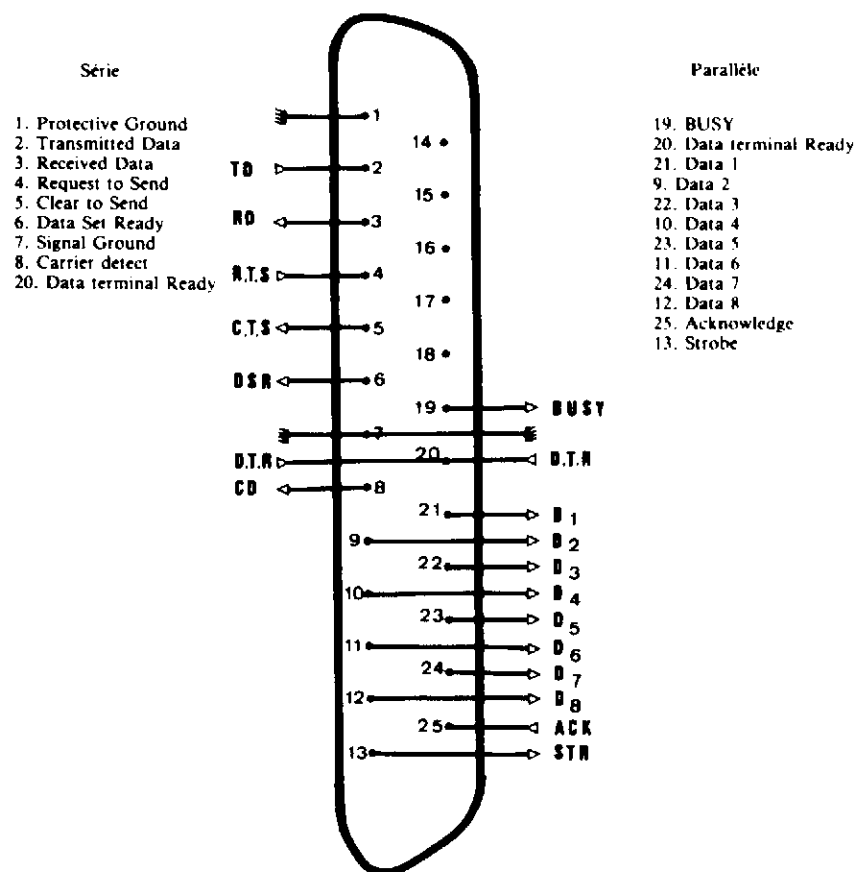
4. Les interfaces

4.1 L'interface RS232/CENTRONICS

L'extension communication est encartable sur le bus standard du TO7. Les entrées/sorties vers les périphériques se font par un connecteur miniature type D 25 broches.

Cette extension permet la communication avec n'importe quel type de terminal, sous forme de liaison série ou parallèle. Le standard de communication série est conforme à la norme RS 232, et l'interface parallèle est de type CENTRONICS.

CONTROLEUR DE COMMUNICATIONS



• L'interface série RS 232 se compose de 7 lignes :

- Deux sorties (DSR et CD) figées à +12V sous 330Ω.
- Deux sorties (RD et CTS) pouvant prendre les valeurs logiques 0 si $9V \leq V_s \leq 12V$ sous 300Ω
- 1 si $-12V \leq V_s \leq -8V$ sous 300Ω
- le courant est limité à 10mA.
- Trois entrées (TD, RTS et DTR) qui transforment les tensions en valeurs logiques
- 0 si $1,25V \leq V_e \leq 30V$
- 1 si $-30V \leq V_e \leq 1V$

• L'interface parallèle se compose de 11 lignes :

- Neuf sorties TTL : STROBE et Data 1 à Data 8
- Deux entrées DTR au niveau logique (BUSY) et Acknowledge
- 0 si $V_e < 1V$
- 1 si $V_e > 1,25V$

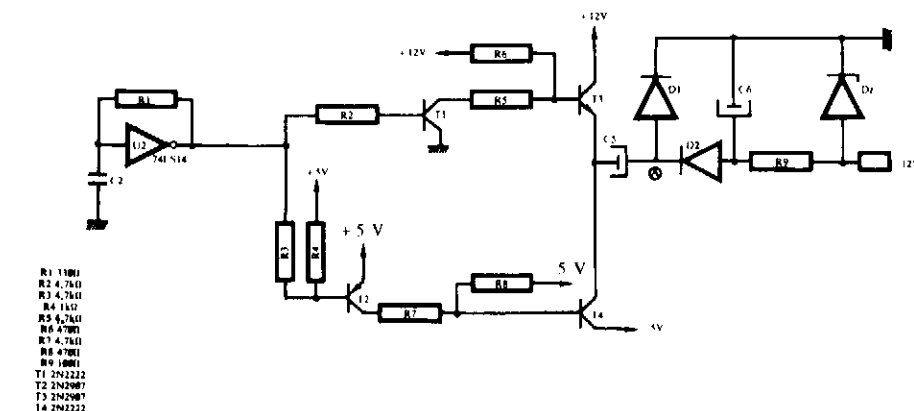
1. Étude des circuits

— Pour la comptabilité avec le standard RS 232 il a été nécessaire de fabriquer du -12V à partir des tensions existantes du TO7, c'est-à-dire le -5V et le +12V.

Un oscillateur a été réalisé à l'aide d'une porte inverseuse Trigger de Schmitt 74LS14, d'une résistance R1 de 330Ω et d'un condensateur C2 de 100nF.

La porte bascule pour une tension typique sur front montant de 1,6V et possède un hystérésis de 0,8V. La fréquence des oscillations rectangulaires en sortie voisine les 33 kHz typiques.

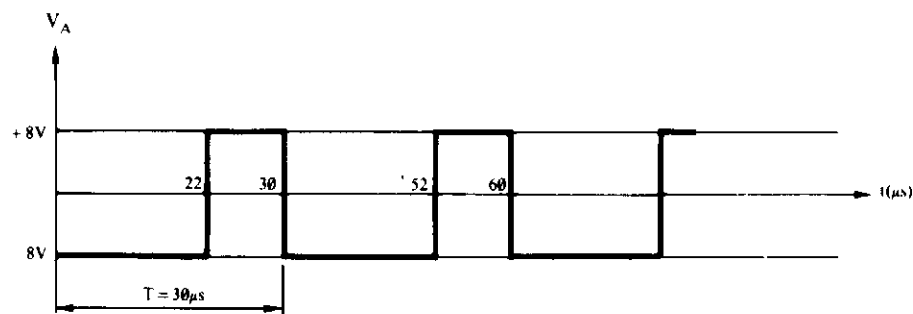
Alimentation — 12 V Interface RS 232



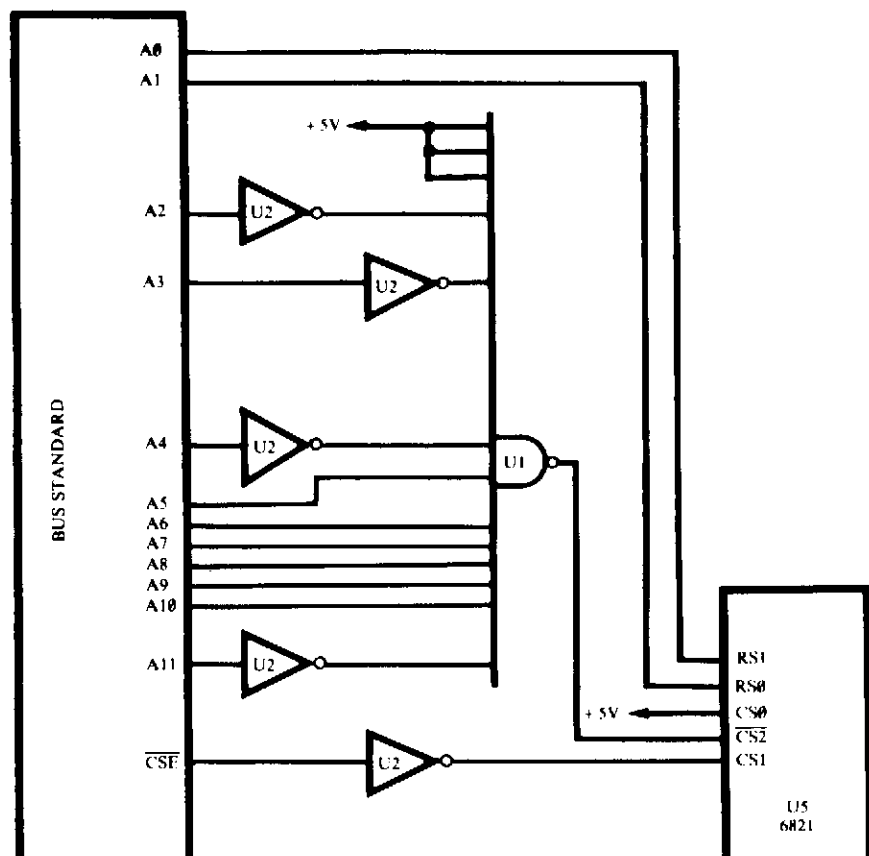
Ce signal rectangulaire attaque deux amplificateurs en opposition de phase, fonctionnant en mode saturé/bloqué et dont les sorties en court-circuit basculent entre +12V et -5V.

Le signal obtenu derrière C5 = 4,7 μ F/25V aura donc l'allure ci-dessous.

Allure de la tension au point A



Décodage d'adresse du 6821 RS 232



Le doubleur de tension réalisé par D1, D2 (1N4001) et C5, C6 (22 μ F/16V) permet d'obtenir une tension négative d'environ 16V qui sera stabilisée à -12V par R9 (100 Ω) et Dz (BZX12V).

Le décodage d'adresse est réalisé conformément au schéma ci-après et permet grâce au signal CSE (\$E000-\$EFFF) de sélectionner le PIA de \$E7E0 à \$E7E3.

On remarquera que les liaisons A1 \leftrightarrow RS0 et A0 \leftrightarrow RS1 permettent d'adresser ce PIA par mots de 16 bits puisque l'on trouve d'abord les 2 registres de direction DDRA, DDRB puis les deux registres de contrôle CRA et CRB.

2. Étude du logiciel de gestion

Le contrôleur de communication est un sous-programme du moniteur TO7 accessible à l'adresse \$E812.

Ce logiciel, appelé RSCOS, travaille selon le même principe que K7CO\$. Il utilise en effet un registre d'état RS.STA (\$602C), et un mot de code RS.OPC d'adresse \$602B, dont la valeur permet de choisir l'une des opérations ci-dessous :

RS.OPC	RS.OPR	%	00000001	Ouverture en lecture/écriture (RS232)
	RS.RDC	%	00000010	Lecture d'un octet
	RS.OPW	%	00000100	Ouverture en écriture seulement (RS232)
	RS.WRC	%	00001000	Écriture un octet
	RS.CLS	%	00010000	Fermeture
	RS.CPY	%	00100000	Copie graphique d'écran
	RS.OPP	%	01000000	Ouverture en écriture parallèle
RS.STA	RS.NRD	%	10000000	Dispositif pas prêt

• Mode série

a) En ouverture, RD et CTS (PA0 et PA1 du 6821) sont positionnés à 1. Puis le TO7 teste DTR :

— Si DTR = 1 \rightarrow le terminal n'est pas prêt

— Si DTR = 0 \rightarrow le terminal est prêt

L'entrée DTR correspond au bit 6 du PORTA. Si ce bit est à 0 alors C (bit de carry) est mis à 0, sinon c'est mis à 1.

Le programme d'initialisation ci-dessous réalise les opérations ci-dessus après avoir initialisé le PIA.

```

RSINI  LDD    BLOCZ   Bloc de 00
      STD    CRA2     Acces aux DDR

      LDD    #Z0000001111111111
              Bit7:TXDATA-INPUT
              Bit6:DTTRMN-INPUT
              Bit5:REQTS-INPUT
              Bit1:CLRTS-OUTPUT
              Bit0:RXDATA-OUTPUT
              PORTB: OUTPUT
      STD    PRA2     Ecrit dans PRA2
                      et PRB2

      LDD    #Z00000010000111100
              CA2:inactif
              CA1:actif sur
              front descendant
              avec IRQ inhibee
              (RTS)
              CB2: OUTPUT a 1
              (STR)
              CB1:actif sur
              front descendant
              avec IRQ inhibee
              (ACKNOWLEDGE)

      STD    CRA2     dans CRA2 et CRB2
      LDA    PRB2     Test de l'existen
                      du controleur: on
                      lit PRB

      COMA    STA    PRB2   On reecrit dedans
                      son complement
      CMPA    PRB2   Et on verifie la

                      reecriture
      BNE    BUSY     Si non conforme,
                      controleur non
                      present
      COM    PRB2     Si present, on le
                      remet dans son
                      etat initial.

```

```

LDA    #3
STA    PRA2      CLEAR TO SEND :
                  masque et
                  RXDATA=mark
DTTEST LDA    PRA2   Bit6 de A = DATA
                  TERMINAL READY
      EORA    RS.STA  Si RS>STA=RSOPP
                  A(6) est inverse
      ANDA    #Z00100000
                  On ne conserve
                  que le bit 6.
      ADDA    #Z11111111
                  C=1 si A<>0 (Flag
                  de BUSY) et C=0
                  si A=0

      RTS

```

b) Pour émettre un octet plusieurs vitesses sont possibles. La vitesse sélectionnée devra être placée dans le registre BAUDS (\$6044) selon le code suivant :

\$ 046A → vitesse de 110 Bauds : Durée 9090 (μs)
 \$ 019B → vitesse de 300 Bauds : Durée 3333 (μs)
 \$ 00CA → vitesse de 600 Bauds : Durée 1666 (μs)
 \$ 0062 → vitesse de 1200 Bauds : Durée 833 (μs)
 \$ 002E → vitesse de 2400 Bauds : Durée 416 (μs)
 \$ 0015 → vitesse de 4800 Bauds : Durée 208 (μs)
 \$ 0007 → vitesse de 9600 Bauds : Durée 104 (μs)

On peut également envoyer 7 ou 8 bits (sans parité) suivant le contenu du registre NOMBRE (\$6046) :

Si NOMBRE :

% 10000000 → 8 bits

% 01000000 → 7 bits

L'octet à transmettre est placé dans l'accumulateur B avant l'appel de la routine :

Le moniteur commence par tester DTR :

— Si DTR = 1 → ERREUR

— Si DTR = 0 → il envoie un START, les 7 ou 8 bits de l'octet puis deux STOPS sur la ligne RD

Puis il attend que DTR passe à 0 pour sortir.

c) A la réception d'un octet, les registres BAUDS et NOMBRE doivent être correctement positionnés.

Le moniteur teste alors DTR :

— Si DTR = 1 → ERREUR

— Si DTR = 0 → il teste RTS :

— Si RTS = 0, il positionne CTS à 0 et attend que TD passe à 0, après quoi il attend un 1/2 délai puis teste à nouveau TD :

— Si TD = 1 on recommence à attendre que TD passe à 0.

— Si TD = 0, on a alors bien un bit de START.

Il saisit alors les 7 ou 8 bits à chaque délai. Après avoir saisi le dernier bit, l'octet est placé dans B, CTS est remis à 1, on attend 1/2 délai et on attend que DTR passe à 0 pour sortir.

— Si RTS = 1, le périphérique n'a rien à envoyer et on ressort quand DTR passe à 0.

• Mode parallèle

a) *En ouverture*, le STROBE est mis à 1, puis le moniteur teste DTR :

— Si DTR = 1 → le terminal n'est pas prêt

— Si DTR = 0 → la ligne est ouverte

b) *Pour émettre un octet*, on teste DTR :

— Si DTR = 1 → ERREUR

— Si DTR = 0 → on envoie l'octet sur le PORTB, puis le STROBE est mis à 0 pendant 8μs avant de revenir à 1. (Action sur CRB2)

Le moniteur attend alors que DTR passe à 1 puis à nouveau à 0 pour sortir.

Dans ce mode, l'octet à transmettre est également passé par l'accumulateur B.

• Copie graphique d'écran

Dans ce mode, le moniteur envoie en mode parallèle un premier octet contenu dans le registre GRCODE (\$6047) et qui est le code "BEL" — \$07.

Puis sans tester DTR, il envoie les 8000 octets de l'écran à raison d'un caractère toutes les 75μs environ, ce qui permet de recopier l'écran sur imprimante thermique en 6s. environ.

• Copie d'écran avec l'imprimante PR 90-080

L'imprimante à impact tape des points par paquets de 8 points verticaux.

Pour recopier l'écran, il faut donc le lire verticalement 8 lignes par lignes sur une largeur d'écran (40 colonnes) puis recommencer avec les 8 lignes suivantes, etc.

C'est ce que permet de faire le programme ci-joint qui illustre parfaitement l'utilisation en assembleur du programme moniteur RSCO\$.

```

#####
!
! Copie d'un ecran complet sur
! l'imprimante PR 90-080
!
! CARRY = 0 Tout est OK.
! CARRY = 1 Erreur, voir RS.STA
!
! Cinc sauts de ligne sont effectues
! a la fin de la copie de facon a
! tenir trois copies dans une page
! 12 ".
!
! Taille = 145 octets
!
#####

```

```

4000 STAD EQU $4000
5F40 ENDAD EQU $5F40
E812 RSCO EQU $E812
502B RSOPC EQU $502B
0040 RSOPP EQU $40
0008 RSWRC EQU $08
0010 RSCLS EQU $10
E7C3 PRC EQU $E7C3
002B INTERL EQU $2B
000A LF EQU $0A
000F SI EQU $0F

```

```

A000 ORG $A000

```

```

A000 34 36 SEIKO PSHS D,X,Y
A002 C6 40 LDB #RSOPP ; Ouverture en //
A004 F7 602B STB RSOPC
A007 B0 E812 JSR RSCO
A00A 25 77 BCS SEIKF
A00C B6 E7C3 LDA PRC
A00F 8A 01 ORA #1
A011 B7 E7C3 STA PRC Select. mem. forme
A014 BE 5F40 LDX #ENDAD
A017 86 78 LDA #3*INTERL

```

```

A019 6F 80 SEIK1 CLR ,X+ Raz octets inutil.
A01B 4A DECA
A01C 26 FB BNE SEIK1
A01E C6 08 LDB #RSMRC
A020 F7 602B STB RSOFC Ecrit. sur liaison
A023 8D 69 BSR SEIK10 Ecrit BS
A025 25 5C BCE SEIK9
A027 8E 4000 LDX #STAD Debut de l'ecran

A02A B6 28 SEIK2 LDA #40 40 colonnes
A02C 34 02 PSHS A Compteur de Col.
A02E 34 10 SEIK3 PSHS X Sauv. pointeur
A030 B6 08 LDA #8 8 bits par octet
A032 34 02 PSHS A Compt. gen. decalq

A034 C6 40 SEIK4 LDB #640 Bit dans CY->Stop

A036 A6 E4 SEIK5 LDA ,S Recup compte decal
A038 34 02 PSHS A compt decal 1 oct.
A03A A6 84 LDA ,X Recup octet concer

A03C 44 SEIK6 LSRA Le decaler
A03D 6A E4 DEC ,S Un decal en moins
A03F 26 FB BNE SEIK6 Pas fini?, contin.
A041 32 61 LEAS 1,S Resto de S
A043 30 88 28 LEAX INTERL,X Descendre 1 ligne
A046 56 RORB Recup bit select.
A047 24 ED BCC SEIK5 Pas encore fini
A049 56 RORB Bit de gauche a :
A04A 8D 42 BSR SEIK10 on a fini,on ecrit
A04C 24 04 BCC SEIK7 Caract bien ecrit
A04E 32 64 LEAS 4,S On remet dans S
A050 20 31 BRA SEIK9 Erreur

A052 30 89 FEEB SEIK7 LEAX -7*INTERL,X X au debut
A056 6A E4 DEC ,S Decr compt gen dec
A058 26 DA BNE SEIK4 On continue
A05A 32 61 LEAS 1,S Resto de S
A05C 35 10 PULS X Recup du debut
A05E 30 01 LEAX 1,X Avancer sur octet
A060 6A E4 DEC ,S Decr compt colonne
A062 26 CA BNE SEIK3 Traiter 8 octets
A064 32 61 LEAS 1,S Resto de S
A066 C6 0A LDB #LF
A068 BD 24 BSR SEIK10 Avancer d'1 ligne
A06A 25 17 BCS SEIK9
A06C 30 89 00F0 LEAX 6*INTERL,X

```

```

A070 8C 5F40 CMPX #ENDAD Fini ?
A073 25 B5 BLD SEIK2 Non
A075 CC 070A LDD #7*256+LF Cadrage

A078 8D 14 SEIK8 BSR SEIK10
A07A 25 07 BCS SEIK9
A07C 4A DECA
A07D 26 F9 BNE SEIK8 Pas encore fini
A07F C6 0F LDB #SI Mode caractere
A081 8D 02 BSR SEIK10

A083 34 01 SEIK9 PSHS CC Sauver l'etat CY
A085 C6 10 LDB #RSCLS
A087 F7 602B STB RSOFC
A08A 8D 02 BSR SEIK10 On ferme
A08C 35 B7 PULS CC,D,X,Y,PC Retour

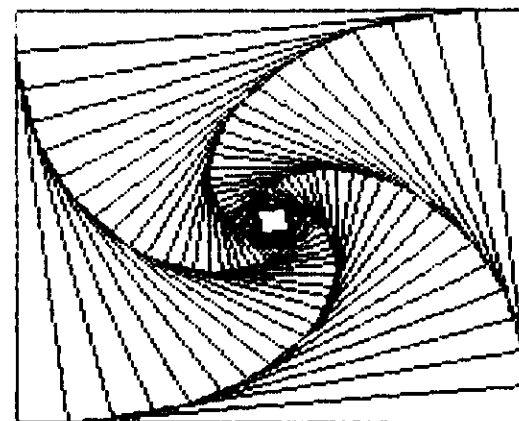
A08E 7E EB12 SEIK10 JMP RSCC

A091 39 RTE

0000 END

00000 Total Errors

```



4.2 L'interface MUSIQUE et JEUX

Cette interface qui utilise un PIA 6821 est logée aux adresses suivantes :

&H E7CC pour le PORTA (et le DDRA)

&H E7CD pour le PORTB (et le DDRB)

&H E7CE pour le CRA

&H E7CF pour le CRB

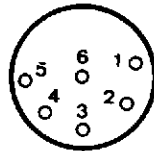
Cette interface a un double rôle :

1. Gérer les deux manettes de jeux (JOYSTICKS) branchées sur les connecteurs 6 broches dont le schéma est donné ci-dessous :

Connexion aux manettes

MANETTE MANETTE

0 1
 PA₀ — AVANT — PA₄
 PA₁ — ARRIÈRE — PA₅
 PA₂ — GAUCHE — PA₆
 PA₃ — DROITE — PA₇
 PB₆ — BOUTON — PB₇
 CA₁ — CA₂



1 — MASSE
 2 — AVANT
 3 — ARRIÈRE
 4 — GAUCHE
 5 — BOUTON
 6 — DROITE

Liaisons au PIA

Brochage Prise

C'est le rôle du PORTA et des bits B₆ et B₇, tous programmés en entrées, que de permettre cette gestion, ainsi que des deux entrées d'interruption CA₁ et CA₂.

2. Synthétiser des sons à l'aide d'un convertisseur numérique/analogique (CNA) réalisé avec les 6 bits restants du PORTB, de B₀ à B₅, programmés en sorties et bufferisés, et d'un circuit de conversion du type R/2R dont la sortie filtrée est reliée à la ligne SON du connecteur standard.

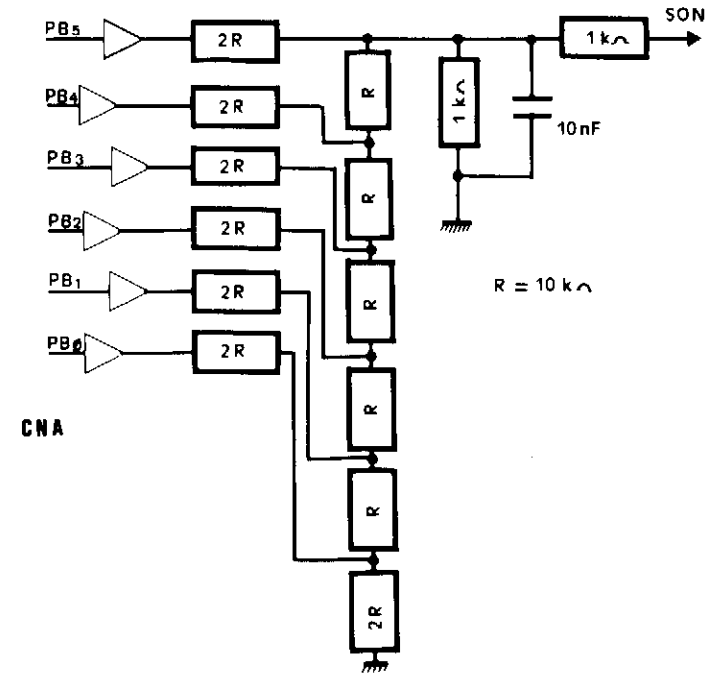
Le niveau de tension maximum sur cette sortie est de 450 mV.

Attention : A la mise sous tension les deux PORTS du PIA sont programmés en ENTREES, il faut donc commencer tout programme par l'initialisation en SORTIE des bits B₀ à B₅ du PORTB.

Soit par exemple en assembleur :

```
CLR $E7CF Mise à 0 du CRB2 → DDRB
LDD # $3F04
STA $E7CD B0 à B5 en SORTIES
STA $E7CD Mise à 1 du CRB2 → PORTB
```

CNA



CNA

1. Principe de fonctionnement des manettes

A l'intérieur de chaque manette sont placés 4 interrupteurs de position, plus 1 interrupteur de commande.

Au repos tous ces interrupteurs sont ouverts et les entrées sont donc au "1" logique.

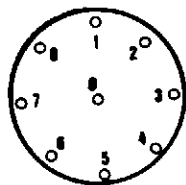
Si un interrupteur est actionné, il se ferme et l'entrée correspondante passe alors au "0" logique, car l'interrupteur est relié à la masse (borne 1).

Les 4 bits d'une manette (par exemple la manette 0) fournissent donc les codes suivants :

BASIC	A ₃	A ₂	A ₁	A ₀	POSITION
0	1	1	1	1	REPOS
1	1	1	1	0	AVANT
5	1	1	0	1	ARRIÈRE
3	0	1	1	1	DROITE
7	1	0	1	1	GAUCHE
2	0	1	1	0	AVANT & DROITE
8	1	0	1	0	AVANT & GAUCHE
4	0	1	0	1	ARRIÈRE & DROITE
6	1	0	0	1	ARRIÈRE & GAUCHE

Pour le BASIC, les positions de la manette sont codées de 0 à 8, 0 correspond à l'état neutre (repos), les valeurs de 1 à 8 correspondent aux positions NORD, NORD-EST, EST, etc. c'est-à-dire à la rotation dans le sens horaire (voir fig. 3).

Code BASIC d'une manette jeu



De la même façon, si l'interrupteur de commande de la manette 0 n'est pas actionné alors l'entrée bit 6 du PORTB est à 1, sinon ce bit passe à 0.

Le programme moniteur JOYS\$ implanté en &HE827 a pour objet de fournir (par l'accumulateur B) le code BASIC de la position de la manette dont le numéro lui aura été passé (par l'accumulateur A), ainsi que l'état du bouton de commande (par le bit de CARRY du CCR).

Le bouton de commande est codé comme suit :

Si $C=0 \rightarrow$ la gachette est au repos.

Si $C=1 \rightarrow$ la gachette est enfoncée.

On peut éventuellement faire fonctionner la gachette en mode interruption puisque B6 et CA1 d'une part et, B7 et CA2, d'autre part sont reliés.

Dans ce cas le logiciel doit être créé par l'utilisateur.

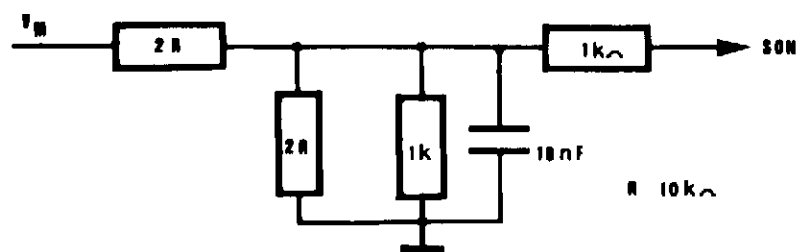
2. Principe de fonctionnement du convertisseur

Calcul de l'influence d'un bit

Supposons que seul le bit B5 soit à 1, et appelons V_M la tension en sortie du buffer correspondant.

Dans ce cas le schéma équivalent du convertisseur est le suivant :

BIT 5="1" et autres BIT à "0"



Dans ce cas la tension sur la sortie SON n'est que de $0,045 \cdot V_M$. On peut calculer de la même façon la tension en sortie si seul le bit B4 est à "1" \rightarrow $SON = 0,023 \cdot V_M$ soit la moitié de la tension créée par B5.

On voit donc que la tension sur la sortie SON est proportionnelle au poids du BIT à "1".

Si plusieurs BIT sont à "1" simultanément, la tension SON sera égale à la somme des tensions propres à chaque bit. Exemple, si B5 et B4 sont seuls à 1, la tension SON vaudra $(0,045 + 0,023) \cdot V_M = 0,068 \cdot V_M$.

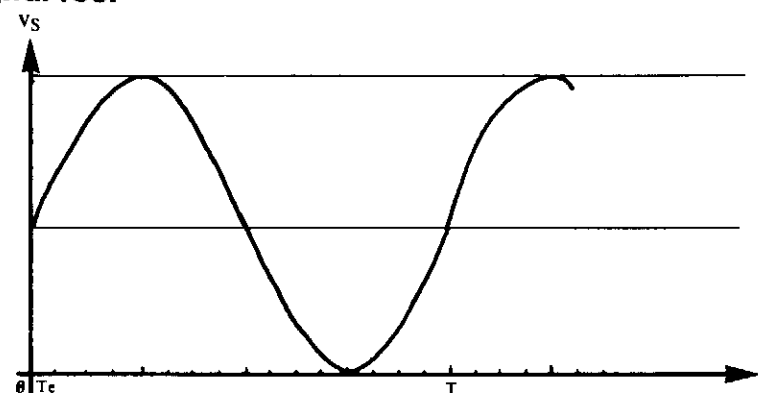
La tension maximum en sortie sera donc d'environ $0,09V_M$ soit 450mV puisque V_M est au maximum égale à 5 Volts.

Méthode de synthèse d'un son

a) Échantillonnage

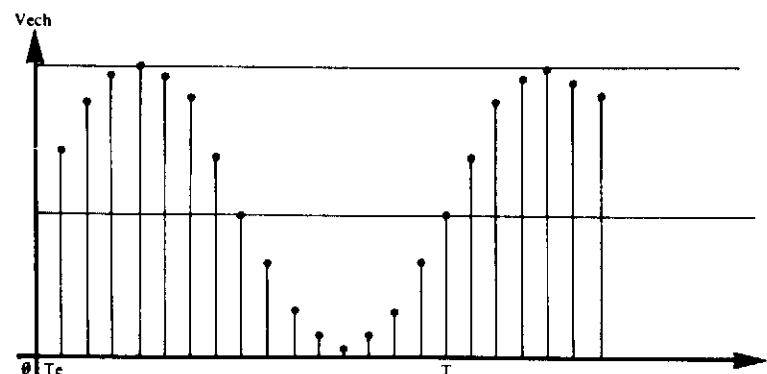
Soit le son pur sinusoïdal ci-dessous :

Signal réel



Échantillonner ce signal consiste à mesurer des "échantillons" de tension à des intervalles de temps égaux (période d'échantillonnage). Si l'on échantillonne avec une période $T_c = T/12$ le signal échantillonné est alors le suivant :

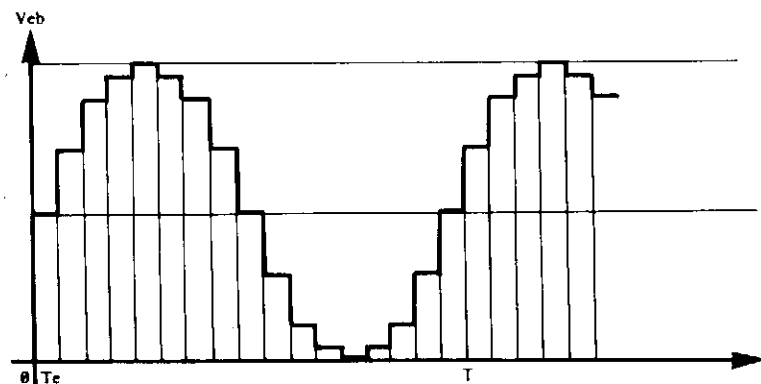
Signal échantillonné



b) Échantillonnage-blocage

Si on maintient la valeur de la tension échantillonnée entre deux prises d'échantillon, on bloque l'échantillon, et le signal correspondant a alors l'allure suivante :

Signal échantillonné bloqué



c) Synthèse

Si l'on veut "fabriquer" un signal approché du signal réel de la figure 5, il suffit de ranger dans une table binaire (6 bits dans le cas présent) des mots dont la valeur est proportionnelle aux échantillons.

Puis avec une fréquence $f_e = 1/T_e$, on envoie ces mots binaires dans le convertisseur N/A qui fabrique alors une tension SON proportionnelle aux échantillons successifs.

Entre deux échantillons, le précédent est maintenu en sortie et même légèrement intégré.

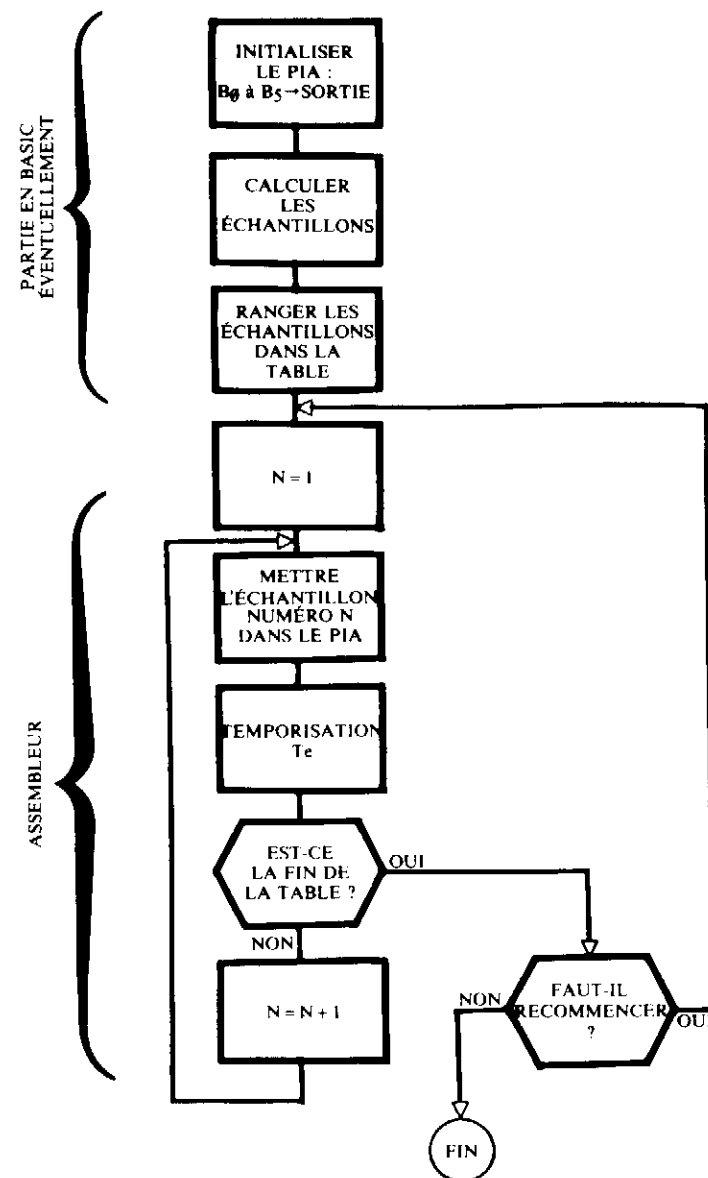
On reconstituera d'autant mieux un son que le nombre d'échantillons sera grand.

On peut faire varier la fréquence d'un son en augmentant ou en diminuant l'intervalle de temps séparant deux échantillons.

Logiciel de synthèse

Le logiciel de synthèse d'un son suivra l'ordinogramme ci-dessous.

SYNTHÈSE D'UN SON



La partie de calcul peut éventuellement être réalisée en BASIC (voir programme ci-joint), mais pour plus de souplesse il est conseillé de réaliser la partie de synthèse proprement dite en assembleur (voir programme ci-joint).

```

10 ' *****
20 ' *
30 ' * PROGRAMME DE SYNTHESE *
40 ' * SONORE *
50 ' *
60 ' *****
70 '
80 '
90 '
100 '-----INITIALISATION-----
110 '
120 SCREEN2,4,4:CONSOLE0,24:CLS
122 LOCATE0,0,0:PRINT"** SYNTHESE D'UN
SON SINUSOIDAL **"
130 CLEAR,&H9FFF
140 DIM ECH(255)
150 '
160 '-----INIT. DU PIA-----
170 '
180 POKE &HE7CF,0 'Acces a DDRB
190 POKE &HE7CD,&H3F 'B0 a B5 en SORTIE
200 POKE &HE7CF,&H04 'Acces au PORTB
210 '
220 '-----ROUTINE ASSEMBLEUR-----
230 '
240 FOR I=0 TO 57
250 READ D
260 POKE &HA000+I,D
270 NEXT I
280 '
290 DATA &H34,&H7E
300 DATA &H1A,&H10
310 DATA &HFC,&HA0,&H52
320 DATA &HF3,&HA0,&H54
330 DATA &HFD,&HA0,&H56
340 DATA &HBE,&HA0,&H52
350 DATA &HF6,&HA0,&H51
360 DATA &H30,&HB5
370 DATA &HA6,&HB4
380 DATA &HB7,&HE7,&HCD
390 DATA &HB6,&HA0,&H50
400 DATA &H4A
410 DATA &H26,&HFD
420 DATA &HBC,&HA0,&H56

```

122

```

430 DATA &H2D,&HEE
440 DATA &H1F,&H10
450 DATA &HB3,&HA0,&H54
460 DATA &H1F,&H01
470 DATA &H7F,&HE7,&HC9
480 DATA &HB6,&HE7,&HCB
490 DATA &HB1,&HFF
500 DATA &H24,&HDA
510 DATA &H1C,&HEF
520 DATA &H35,&HFE
530 '
540 '-----CALCUL DES ECHANTILLONS---
550 '
560 CONSOLE15,24:CLS:LOCATE0,15,0
570 INPUT"NB D'ECHANTILLONS (255 MAX) :
",NE
580 INPUT"TEMPORISATION (de 0 a 255) : "
,TE
590 INPUT "PAS DE PRELEVEMENT : ",PAS
600 PRINT:PRINT:COLOR1:ATTRB1,1:PRINT"SI
LENCE, JE CALCULE":COLOR2:ATTRB0,0
610 POKE &HA050,TE 'Duree de Tempo
620 POKE &HA051,PAS 'Pas d'echantill
630 POKE &HA052,&H0 'Debut de la tabl
640 POKE &HA053,&H00 'd'echantillonage
650 POKE &HA054,NE @ 256 'Nombre
660 POKE &HA055,NE MOD 256 'd'echant
670 CONSOLE2,12:CLS:LOCATE0,2,0
680 LINE(0,16)-(0,100),6
690 LINE(0,100)-(320,100),6
692 LOCATE1,2:PRINT"Vs":LOCATE39,11:PRIN
T"t";
700 W=6.28
710 FOR I=0 TO NE-1
720 ECH(I)=31+31*SIN(W*I/NE)
730 POKE &HB000+I,ECH(I)
740 NEXT I
750 FOR I=0 TO NE-1 STEP PAS
760 FOR J=0 TO PAS-1
770 PSET(I+J,100-ECH(I))
780 NEXT J,I
790 '
800 '-----SYNTHESE D'UN SON-----
810 '
820 EXEC &HA000
830 FORN=1TO100:NEXTN
840 GOTO 560
850 END

```

```

*****
*
* ROUTINE D'ENVOI DES ECHANTILLONS
*
* DANS LE PORTE DU PIA
*
*
* ENTREES :
*
* $A050 = Duree de la temporisation
* $A051 = Pas d'echantillonnage
* $A052 = Debut de la table
* $A054 = Nombre d'echantillons
*
*
* SORTIES :
*
* $A056 = Fin de la table
*
*
* PORTB en $E7CD
*
* CLAVIER en $E7C8 et $E7C9
*
*
*****

```

```

E7CD PORTB2 EQU    $E7CD
E7C8 PORTA1 EQU    $E7C8
E7C9 PORTB1 EQU    $E7C9

```

```

A000                                ORG    $A000

A000 34 7E      PSHS    U,Y,X,DP,B,A
A002 1A 10      DRCC    #$10
A004 FC A052    LDD     $A052
A007 F3 A054    ADDD    $A054
A00A FD A056    STD     $A056
A00D BE A052    LDX     $A052
A010 F6 A051 BOUC0 LDB     $A051

A013 30 B5 BOUC1 LEAX    B,X
A015 A6 B4      LDA     ,X
A017 B7 E7CD    STA     PORTB2
A01A B6 A050    LDA     $A050

```

```

A01D 4A          BOUC2 DECA
A01E 26 FD          BNE    BOUC2
A020 BC A056      CMPX    $A056
A023 2D EE          BLT    BOUC1
A025 1F 10          TFR     X,D

A027 B3 A054      SUBD    $A054
A02A 7F E7C9      CLR     PORTB1
A02D B6 E7C8      LDA     PORTA1 CLAVIER
A030 81 FF          CMPA    #$FF
A032 24 DC          BHS     BOUC0
A034 1C EF          ANDCC   #$EF
A036 35 FE          PULS    A,B,DP,X,Y,U,PC

                                0000      END

```

00000 Total Errors

C'est ce programme assembleur qui est rentré en \$A000 à l'aide de l'instruction POKE &HA000 + I, D et appelé ensuite par EXEC &HA000. Pour générer d'autres types de sons il suffit de changer ligne 720 la formule de calcul des échantillons.

4.3 Le MODEM

L'extension TELETTEL (\$E7E8, \$E7FF) contient un MODEM et un interface de raccordement direct à la ligne conformes aux normes (PTT et V23) appliquées pour le terminal annuaire et le terminal VEL17Y.

Modem pour TO7

Ce modem s'interface d'un côté au bus interne du micro-ordinateur et de l'autre à la ligne du réseau téléphonique commuté.

Il transmet les informations binaires provenant du terminal en mode synchrone et fonctionne en modulation par déplacement de fréquence.

Selon l'application, la transmission s'effectue :

- en 75/1200 bits/s duplex intégral dans le cas d'une liaison avec une banque de données ;
- en 1200/1200 bits/s alternat pour liaison entre ordinateurs.

Ce modem est conforme aux recommandations de l'avis V23 du CCITT.

	Fonctionnement 75/1200	Fonctionnement 1200
Débit d'information	1200 bits/s rec 75 bits/s em	1200 bits/s
Rapidité de modulation	1200 bauds rec/ 75 bauds em	1200 bauds
Mode de transmission	asynchrone	asynchrone
Type de modulation	déplacement de fréquence	déplacement de fréquence
Procédure de transmission	duplex intégral	alternat
Liaison à la voie de transmission :		
• Support de transmission	RC	RC
• Impédance aux accès	600Ω symétrique	600Ω symétrique
• Niveau émission	-2 et -10 dBm	-2 et -10 dBm
• Niveau réception	0 à -43 dBm	0 à -43 dBm
• Régulation du courant continu de ligne	automatique	automatique
Alimentation	-5V ± 12V	+5V / ± 12V
Consommation	< 1,5W	< 1,5W
Température fonctionne- ment	5 °C à + 45 °C	5° C à +45 °C
Température stockage	-40 °C à + 70 °C	-40 °C à + 70 °C
Humidité relative	≤ 80 %	≤ 80 %
Dimensions :		
H × L × P(mm)	70 × 70 × 50	70 × 70 × 50

Elle permet la consultation des banques de données en se branchant simplement sur la prise téléphone, et en utilisant la ROM contenant le logiciel de gestion de l'extension.

Le fonctionnement à 1200 bauds en half duplex permet la communication entre deux TO7 via la ligne PTT commutée ou entre un TO7 et un ordinateur ayant son propre protocole à condition de développer le logiciel de gestion approprié.

Ce modem contient un relais qui permet de basculer la ligne entre le poste et le TO7 ainsi qu'un circuit d'interface asynchrone ACIA 6850 de Motorola.

5. Le moniteur

5.1 Map Mémoire du T9000 et du TO7 Modèle 1

Map Mémoire du TO7 modèle 1

ADRESSE (HEX)

0000-3FFF	CARTOUCHE ROM ENFICHABLE 16K
4000-5FFF	MEMOIRE D'ECRAN 8K
6000-60FF	PAGE 0
6100-7FFF	MEMOIRE UTILISATEUR 8K
8000-BFFF	EXTENSION MEMOIRE 16 K
C000-DFFF	8K LIBRE
E000-E7BF	1,9K POUR FLOPPY
E7C0-E7C7	PIA 6846 SYSTEME
E7C8-E7CB	PIA 6821 SYSTEME
E7CC-E7CF	PIA 6821 EXTENSION JEU
E7D0-E7DF	CONTROLEUR DE MINI-FLOPPY
E7E0-E7E3	PIA 6821 RS-232 ET INTERFACE PARALLELE
E7E4-E7FF	TELETEL
E800-FFFF	MONITEUR SYSTEME 6K

REVISION : 03/01/83

Map mémoire du TO7 modèle 2

ADRESSE (HEX)

0000-3FFF	CARTOUCHE ROM ENFICHABLE 16K
4000-5FFF	MEMOIRE D'ECRAN 8K
6000-60FF	PAGE 0
6100-9FFF	MEMOIRE UTILISATEUR 16K
A000-DFFF	EXTENSION MEMOIRE 16K PAGINABLE
E000-E7BF	1,9K POUR FLOPPY
E7C0-E7C7	PIA 6846 SYSTEME
E7C8-E7CB	PIA 6821 SYSTEME
E7CC-E7CF	PIA 6821 EXTENSION JEU
E7D0-E7DF	CONTROLEUR DE MINI-FLOPPY
E7E0-E7E3	PIA 6821 RS-232 ET INTERFACE PARALLELE
E7E4-E7FF	LOGIC GATE ARRAY
E7E8-E7FF	TELETEL 6850 ACIA
E800-FFFF	MONITEUR SYSTEME 6K

REVISION : 15/07/83

5.2 Page zéro du moniteur système du T9000

Adresse	Identificateur	Commentaires
\$6000	TERMIN,25	Table des terminateurs de lignes
\$6019	STATUS,1	BIT7 = SEMIGRAPH, BIT6 = ROLLUP RAPIDE, BIT5 = INTERRUPT TIMER USER, BIT3 = CU.USER, BIT2 = CURSEUR, BIT1 = INCREMENT CLAVIER, BIT0 = TOUCHE DEJA LUE
\$601A	TABP,1	Forme avec RANG un pointeur courant de 16 bits dans la table des terminateurs de lignes
\$601B	RANG,1	
\$601C	TOPTAB,1	Forme avec TOPRAN un pointeur de 16 bits sur le sommet logique de la table des terminateurs
\$601D	TOPRAN,1	1 ^{re} rangée de la fenêtre
\$601E	BOTTAB,1	Forme avec BOTRAN un pointeur de 16 bits sur la fin logique de la table des terminateurs
\$601F	BOTRAN,1	Dernière rangée de la fenêtre
\$6020	COLN,1	
\$6021	IRQTP,2	Pointeur IRQ utilisateur
\$6023	FIRQTP,2	Pointeur FIRQ utilisateur
\$6025	CCIPT,2	Pointeur INTERRUPT sur CCI
\$6027	TIMEPT,2	Pointeur INTERRUPT TIMER UTILISATEUR
\$6029	K7.OPC,1	Mot de code de l'opération cassette désirée
\$602A	K7.STA,1	Status courant du contrôleur cassette
\$602B	RS.OPC,1	Mot de code pour la liaison RS-232
\$602C	RS.STA,1	Status courant de la liaison RS-232
\$602D	USERAF,2	Pointeur sur le générateur de caractère usager
\$602F	SWI,2	Pointeur SWI
\$6031	TEMPO,2	Tempo général de l'interpréteur musical
\$6033	DUREE,2	Durée
\$6035	TIMBRE,1	Timbre
\$6036	OCTAVE,2	Octave
\$6038	FORME,1	Couleur du PLOT et du DRAW (- 8 à + 7)
\$6039	ATRANG,1	BIT7 = SCRFL, BIT2 = LARGEUR, BIT1 = HAUTEUR
\$603A	ATRSCR,1	BIT7 = FOND, BIT6 = FORME, BIT2 = LARGEUR, BIT1 = HAUTEUR
\$603B	COLOUR,1	X.X.B.V.R.B.V.R
\$603C	TELETL,1	Flag de mode Teletel et PR.STA, BIT7 = MODE PAGE, BIT6 = ECHO
\$603D	PLOTX,2	Dernier plot
\$603F	PLOTY,2	Dernier plot
\$6041	CHDRAW,1	Caractères ASCII utilisés pour PSET, LINE et BOX "CH"
\$6042	CURSFL,1	Flag de mouvement curseur indiquant que l'on ne pose pas de liens de ligne dans la table des terminateurs
\$6043	COPCHR,1	Flag indiquant que BACKSPACE et HORIZONTAL TABULATION copient le caractère courant
\$6044	BAUDS,2	Paramètre de vitesse de la liaison RS232
\$6046	NOMBRE,1	Nombre de bits de transmission RS232 : 1 ou 2
\$6047	GRCODE,1	Code déclenchant le mode graphique de l'imprimante

PRIMITIVES DE DISQUE

\$6048	DK.OPC,1	MOT DE COMMANDE : code instruction utilisé par DKCONT pour appeler une commande paramétrée par les registres qui suivent
\$6049	DK.DRV,1	N° du disque sélectionné
\$604A	DK.TRK,2	N° de piste
\$604C	DK.SEC,1	N° de secteur
\$604D	DK.NUM,1	Nombre de secteurs en accès multiple
\$604E	DK.STA,1	MOT D'ETAT : contient le code d'erreur signalé par C=1 en sortie de DKCONT
\$604F	DK.BUF,2	I/O BUFFER POINTER

REGISTRES COURANTS DE PISTE

\$6051	TRACK0,2	Position courante de la tête sur le drive 0
\$6053	TRACK1,2	Position courante de la tête sur le drive 1
\$6055	TRACK2,2	Position courante de la tête sur le drive 2
\$6057	TRACK3,2	Position courante de la tête sur le drive 3

AUTRES REGISTRES

\$6059	SEQUE,1	0 = NORAML, 2 = ESCAPE, 4 = UNSEP, 6 = DEVCO2, 8 = DEVCO3
\$605A	SCRPT,2	Pointeur d'écran
\$605C	SAVCOL,1	Double de COLOUR
\$605D	ASCII,1	Dernier caractère
\$605E	KEY,1	Touche clavier
\$605F	CMPTKB,1	REPEATS du clavier
\$6060	STADR,2	1 ^{er} octet définissant l'origine de la fenêtre
\$6062	ENDDR,2	Dernier octet définissant la fin de la fenêtre
\$6064	TCSAV,1	Sauvegarde de l'ETAT courant du timer
\$6065	TCTSAV,2	Sauvegarde du COMPTE courant du timer
\$6067	BAUD,1	Paramètre de vitesse de transmission cassette
\$6068	SAVTR,2	Sauvegarde des attributs courants d'écran
\$606A	USI,1	1 ^{er} caractère des séquences UNIT SEPARATOR
\$606B	COMPT,1	Compteur de caractères répétés
\$606C	TEMP,2	Registre temporaire
\$606E	SAVEST,2	Registre de sauvegarde de S
\$6070	ACCENT,1	FLAG ACCENT 2 = affich. accent ; 1 = saut des 2 octets ; 3 = cédille

Adresse	Identificateur	Commentaires
\$6071	SS2GET,1	
\$6072	SS3GET,1	
\$6073	ABCMPT,2	Compteur de la touche STOP pour avorter une lecture cassette
\$6075	EFCMPT,1	Flag de clignotement curseur
\$6076	BLOCZ,2	Bloc de deux 0 pour initialiser les registres 16 bits
\$6078	SCROLLS,1	Flag de "smooth scroll"
\$6079	CHX1,2	Table des choix effectués au "menu" : 1 ^{er} choix
\$607B	CHRX2,2	Table des choix effectués au "menu" : 2 ^e choix
\$607D	CHRX3,2	Table des choix effectués au "menu" : 3 ^e choix
\$607F	RUNFLG,1	Flag indiquant une entrée en mode "RUN"

\$6080	DKFLG,1	Flag indiquant la présence du contrôleur de disque
\$6081	STKEND,80	Profondeur de la pile système
\$60D0	STACK,*-1	Sommet de la pile
\$60D1	APPLIC,1	Checksum de l'application en cours
\$60D2	DECALG,1	Décalage du light-pen
\$60D3	LPBUFF,16	Buffer de saisie du light-pen
\$60E3	NBPST,1	Nombre de pistes par face disque
\$60E4	CPTSCT,1	Compteur de secteurs
\$60E5	DTAB1,3	Zone de travail
\$60E8	FTAB1,1	Fin de zone de travail DTAB1
\$60E9	TAB2,2	ADDRESS MARK CLOCKS \$F5 \$FE
\$60EB	DTAB2,12	Zone de travail
\$60F7	FTAB2,0	Fin de zone de travail DTAB2

5.3 Les adresses d'entrée-sortie

Adresses de l'écran

STAD	EQU	\$4000
ENDAD	EQU	STAD + \$1F40
CURSOR	EQU	STAD + \$118 ORIGINE DE LA PREMIERE RANGEE
TELCUR	EQU	STAD + \$258 ORIGINE DE LA DEUXIEME RANGEE
ORGROL	EQU	STAD + \$140 ORIGINE DU ROLLUP
ORGBL	EQU	STAD + \$280 ORIGINE DU ROLLUP DOUBLE HAUTEUR
DERCUR	EQU	STAD + \$F18 CURSEUR SUR LA DERNIERE LIGNE
DIRECT	EQU	\$60 ADRESSE DE LA PAGE 0
SETDP	DIRECT	
INTERL	EQU	\$28 SAUT INTERLIGNE

Adresses d'entrée/sorties

6846	CSR	EQU	\$E7C0	COMPOSITE STATUS REGISTER
	CRC	EQU	CSR + 1	REGISTRE DE CONTRÔLE DU PORT C
				CC1 (input) est libre (IRQ externe sur T9000)
				CC2 (output) son
				CT0 (output) Ecriture sur la K7
	DDRC	EQU	CSR + 2	REGISTRE DE DIRECTION DU PORT C
	PRC	EQU	CSR + 3	REGISTRE DE DONNEES DU PORT C
				P0 (output) Forme
				P1 (input) Interrupteur crayon optique
				P2 (output) 1/2 teinte du tour : 0 = pastel, 1 = sature. Sur le T9000, c'était la commande de page couleur (desactivée)
				P3 (output) Led clavier
				P4 (output) Rouge tour
				P5 (output) Vert tour
				P6 (output) Bleu tour
				P7 (input) Lecture K7
	TCR	EQU	CSR + 5	TIMER CONTROL REGISTER
	TMSB	EQU	CSR + 6	TIMER M.S.B.
	TLSB	EQU	CSR + 7	TIMER L.S.B.

Modifications TO7 Modèle 1

Adresse	Identificateur	Commentaires
\$6019	STATUS,1	BIT3 = Flag interrupt clavier BIT1 supprimé sur le TO7
\$6025	NMIPT,Z	Pointeur NMI utilisateur
\$6067	LATCLV,1	Latence clavier programmable
\$6073	BUZZ,1	Flag de buzzer : 0 = ON 1 = OFF
\$60D2	DECALG,1	Décalage du light-pen : Il fonctionne en octets sur le T9000 et en points sur le TO7 modèle 1
\$60D3	LPBUFF,24	Buffer de saisie du light-pen : 24 octets sur le TO7 modèle 1 au lieu de 16 sur le T9000
\$60FE	TSTRST,2	Test du Reset

Modifications TO7 Modèle 2 (TO7-70)

Adresse	Identificateur	Commentaires
\$6038	FORME,1	Couleur du PLOT et du DRAW, [-8,7] normal [8,15] pastel
\$603B	COLOUR,1	1/2 teinte fond. 1/2 teinte forme. B.V.R.B.V.R.
\$6081	STKEND,76	Profondeur de la pile système
\$60CD	PTCLAV,2	Pointeur décodage clavier
\$60CF	PTGENE,2	Pointeur générateur de caractère

6821 SYSTEME

PRA	EQU	\$E7C8	REGISTRE DE DONNEES DU PORT A PA0-7 (input) Lecture matrice clavier
PRB	EQU	PRA + 1	REGISTRE DE DONNEES DU PORT B PB0-7 (output) Ecriture matrice clavier
CRA	EQU	PRA + 2	REGISTRE DE CONTROLE DU PORT A CA1 (input) Lecture de INITRAME CA2 (output) Commande moteur K7
CRB	EQU	PRA + 3	REGISTRE DE CONTROLE DU PORT B CB1 (output) Commande de OUTPUT ENABLE sur T9000 et TO7 modele 1, commande d'incrustation sur modele 2 : 0 = mode incruste, 1 = mode normal (positionné à 1 par défaut comme pour OUTPUT ENABLE).

6821 JEUX

PRA1	EQU	\$E7CC	REGISTRE DONNEES PORT A1 PA0-7 (input) Lecture directions joysticks : 0 Nord0, 1 Sud0, 2 West0, 3 Est0, 4 Nord1, 5 Sud1, 6 West1, 7 Est1
PRB1	EQU	PRA1 + 1	REGISTRE DONNEES PORT B1 PB0-5 (input) Convertisseur digital/analogique sur 6 bits PB6 (input) Action joystick 0 PB7 (input) Action joystick 1
CRA1	EQU	PRA1 + 2	REGISTRE DE CONTROLE PORT A1 CA1 (input) Action joystick 0
CRB1	EQU	PRA1 + 3	REGISTRE DE CONTROLE PORT B1 CB1 (input) Action joystick 1

6821 RS-232 ET INTERFACE PARALLELE CENTRONICS

PRA2	EQU	\$E7E0	REGISTRE DE DONNEES PORT A2 PA0 (output) Receive data PA1 (output) Clear to send PA5 (input) Request to send PA6 (input) Data terminal ready PA7 (input) Transmit data
PRB2	EQU	PRA2 + 1	REGISTRE DE DONNEES PORT B2 PB0-7 (output) Données paralleles
TES.SRC			
CRA2	EQU	PRA2 + 2	REGISTRE DE CONTROLE PORT A2 CA1 (input) Request to send
CRB2	EQU	PRA2 + 3	REGISTRE DE CONTROLE PORT B2 CB1 (input) Acknowledge CB2 (output) Strobe

Modifications TO7 Modèle 1

Circuit 6846 :

\$E7C1	CRC	Registre de contrôle du PORT C : CC1 (input) est libre (IRQ sur le T9000)
--------	-----	--

Modifications TO7 Modèle 2 (TO7-70)

Circuit 6821 système

\$E7CB	CRB	Registre de contrôle du PORT B : CB1 (output) : — commande OUTPUT ENABLE sur T9000 et TO7 modèle 1 — commande d'incrustation sur modèle 2 : 0 mode incrusté 1 mode normal positionné à 1 par défaut comme pour OUTPUT ENABLE
\$E7C9	PRB	PB0 PB1 Adressage codé de la matrice clavier PB2 PB3 CS de la banque RAM 16K n° 0 PB4 CS de la banque RAM 16K n° 1 PB5 PB6 CS codé des banques de RAM 16K de l'extension PB7 mémoire 64K.