

Table des matières

PRESENTATION DE LA ROM CONTROLEUR	3
L'IDENTIFICATEUR DE LA ROM CONTROLEUR (IDDISK).....	3
LES FONCTIONS STANDARD DE LA ROM CONTROLEUR	3
GENERALITES DU SYSTEME PHYSIQUE.....	5
LES REGISTRES SYSTEME.....	5
LE RAMDISK.....	6
LES FORMATS SUR SUPPORTS MAGNETIQUES	6
LE QDD	7
Structure d'une piste QDD.....	7
Correspondance système des secteurs QDD	7
LE FLOPPY	8
Structure d'une piste simple densité.....	8
Structure d'une piste double densité.....	9
Entrelacements système pour floppy	9
LES TEMPORISATIONS.....	9
LE CYCLIC REDUNDANCY CHECK.....	10
Nécessité du CRC	10
Principe de calcul du CRC	10
Optimisation du calcul du CRC.....	11
GENERALITES DU SYSTEME LOGIQUE.....	13
STRUCTURE DU SECTEUR DE NOM DE DISQUETTE	13
STRUCTURE DE LA FAT.....	13
STRUCTURE D'UNE ENTREE DE FICHIER DANS LE CATALOGUE.....	13
LE FICHIER SCRATCH.DOS	14
CHARGEMENT DE LA FAT - CAS D'UNE FAT DE 160 OCTETS.....	14
RECHERCHE D'UN BLOC LIBRE DANS LA FAT POUR LE FLOPPY	14
REGISTRES MONITEUR	15
LES ENTREES STANDARD DU SYSTEME D'EXPLOITATION DU CONTROLEUR	16
\$A004/\$E004 : Fonctions standard.....	16
\$A007/\$E007 : Lancement du boot	17
\$A00A/\$E00A : Formatage	17
\$A00D/\$E00D : Chargement de la FAT.....	17
\$A010/\$E010 : Ouverture d'un fichier	17
\$A013/\$E013 : Effacement du fichier courant	17
\$A016/\$E016 : Ecriture du secteur courant.....	18
\$A019/\$E019 : Création d'un fichier.....	18
\$A01C/\$E01C : Recherche d'un bloc libre.....	18
\$A01F/\$E01F : Initialisation d'une opération sur un bloc	18
\$A022/\$E022 : Clôture d'une opération en écriture	18
\$A025/\$E025 : Fonctions standard avec numéro de secteur réel (QDD exclusivement).....	19
PROCEDURES STANDARD DU SYSTEME LOGIQUE	19
Lecture d'un fichier :	19
Sauvegarde d'un fichier sans écrasement:	19
Sauvegarde d'un fichier avec écrasement :	19
Effacement d'un fichier :	19
ROUTINE TYPE DE GESTION DE FICHIER	19
PROGRAMMATION DES DIFFERENTS CONTROLEURS.....	23
LE THMFC1	23
Les registres disque	23
Les opérations automatiques.....	24
La lecture automatique d'un secteur	24
L'écriture automatique d'un secteur	24
La lecture automatique de l'adresse.....	25
LE WD2793 (CONTROLEUR INTERNE DU TO9).....	26
Les registres disque	26

Table des matières

<i>Lecture ou écriture d'un secteur (READ SECTOR - WRITE SECTOR)</i>	26
<i>Lecture d'une adresse (READ ADDRESS)</i>	27
<i>Lecture d'une piste (READ TRACK)</i>	27
<i>Écriture d'une piste (WRITE TRACK)</i>	27
DESASSEMBLAGE DES ROMS CONTROLEURS	28
CONTROLEUR INTERNE DES TO8, TO8D ET TO9+ (THMFC1) - 1ERE BANQUE.....	28
CONTROLEUR INTERNE DES TO8 ET TO8D (THMFC1) - 2EME BANQUE.....	36
CONTROLEUR INTERNE DU TO9+ (THMFC1) - 2EME BANQUE	43
CONTROLEUR INTERNE DU TO9 (WD2793)	51
CONTROLEUR EXTERNE CD90-351 VERSION A (THMFC1) - 1ERE BANQUE	57
CONTROLEUR EXTERNE CD90-351 VERSION A (THMFC1) - 2EME BANQUE	63
CONTROLEUR EXTERNE CD90-351 VERSION A (THMFC1) - 3EME BANQUE	69
CONTROLEUR EXTERNE CD90-351 VERSION A (THMFC1) - 4EME BANQUE	72
CONTROLEUR EXTERNE CD90-351 VERSION B (THMFC1) - 1ERE BANQUE	78
CONTROLEUR EXTERNE CD90-351 VERSION B (THMFC1) - 2EME BANQUE.....	83
CONTROLEUR EXTERNE CD90-351 VERSION B (THMFC1) - 3EME BANQUE.....	89
CONTROLEUR EXTERNE CD90-351 VERSION B (THMFC1) - 4EME BANQUE.....	92
CONTROLEUR EXTERNE CD90-640	98
CONTROLEUR EXTERNE CD90-015	104
CONTROLEUR EXTERNE CQ90-028	110
CONTROLEUR NANORESEAU VERSION III	116
SOURCES	122

PRESENTATION DE LA ROM CONTROLEUR

L'identificateur de la ROM contrôleur (IDDISK)

Les 3 caractères débutant toute ROM moniteur donnent une évaluation des capacités du moniteur à gérer ses périphériques:

Adresse		Caractère	Signification
MO	TO		
\$A000	\$E000	M	Contrôleur Thomson Micro-Informatique
		D	Contrôleur Western Digital
		Q	Contrôleur ODD
		R	Contrôleur de nanoréseau
\$A001	\$E001	T	FAT de 160 octets
		K	FAT de 80 octets
\$A002	\$E002	D	Double densité
		C	Simple densité
\$A003	\$E003	\$XX	Checksum de l'identificateur de ROM

Le caractère en \$A001/\$E001 renseigne sur la taille de la FAT, et donne donc une idée du nombre de pistes du support. Une FAT de 160 octets implique un disque de 80 pistes. Une FAT de 80 octets implique un disque de 40 pistes ou moins.

On peut remarquer que le caractère contenu à l'adresse \$A002/\$E002 est le même que celui qui est renvoyé dans DKSTA (\$204E/\$604E) lors de l'initialisation moniteur d'un contrôleur disque (DKSTA est en fait initialisé par le programme, sans rapport nécessaire avec le registre \$A002/\$E002).

Les exceptions sont à réserver pour les ROM nanoréseau dont l'identificateur n'a qu'une signification restreinte : « REC » pour la version 3, « RES » pour la version 2, etc... A l'initialisation du moniteur, seul le premier caractère ("R") renseigne alors sur la présence d'un contrôleur nanoréseau.

Ces trois premiers caractères permettent à certains logiciels de savoir comment traiter les données du support.

Le caractère en \$A003/\$E003 est le checksum, soit la somme des trois premiers caractères augmentée de \$55. La validité de ce checksum renseigne le système, au reset, sur la présence d'une ROM contrôleur. Il renseigne aussi l'ExtraMoniteur sur le type de support qu'il aura à gérer. Lors de son initialisation, l'ExtraMoniteur reconnaît quatre types de contrôleur d'après le checksum d'IDDISK et met à jour leur caractéristique :

Checksum	Densité	Pistes par face	Secteurs par piste	Octets par secteur
\$27	Simple	40	16	128
\$28	Double	40	16	255
\$30 et \$34	Simple	25	xx	128
Autre	Double	80	16	255

Pour le test de reconnaissance par l'ExtraMoniteur, seuls les bits 0 à 5 du checksum sont significatifs. Si vous concevez un nouveau contrôleur, veillez à ce que votre checksum d'IDDISK corresponde bien aux possibilités du lecteur ou inversement.

Les fonctions standard de la ROM contrôleur

Les appels aux différentes fonctions de la ROM contrôleur ont des entrées normalisées (ici, avec leur nom officiel).

Les 3 premières entrées concernent les opérations physiques de base, dont la structure logicielle interne varie en fonction des contraintes de chaque contrôleur :

\$A004/\$E004	DKCONT	Fonctions standard (équivalent à \$E82A sur TO et au SWI de code \$26 sur MO)
\$A007/\$E007	DKBOOT	Lancement du boot
\$A00A/\$E00A	DKFMT	Formatage

Suivent les 8 entrées relatives au système d'exploitation logique. Cet embryon de système d'exploitation a été utilisé par toute une gamme de logiciels (ColorPaint, Polyphonia,...) lorsque le système n'assurait pas l'exploitation logique des lecteurs de disquette (sur MO, TO7 et TO7-70) et comporte de pratiques raccourcis de fonctions. Les routines concernées ont à peu de choses près toujours la même structure logicielle, car n'ayant pas beaucoup évolué. Les registres d'entrée et de retour sont invariables :

\$A00D/\$E00D	LECFA	Lecture de la FAT
\$A010/\$E010	RECFI	Recherche d'un fichier
\$A013/\$E013	RECUP	Effacement d'un fichier
\$A016/\$E016	ECRSE	Ecriture d'un secteur
\$A019/\$E019	ALLOD	Allocation de départ (allocation d'une entrée de fichier dans le catalogue)
\$A01C/\$E01C	ALLOB	Allocation d'un bloc
\$A01F/\$E01F	MAJCL	Mise à jour cluster
\$A022/\$E022	FINTR	Fin du transfert

Pour les contrôleurs QDD, il existe une entrée de plus :

\$A025/\$E025 Fonctions standard avec numéro de secteur réel

... et éventuellement :

\$A028/\$E028 Fonctions système avec numéro de secteur réel

GENERALITES DU SYSTEME PHYSIQUE

Les registres système

MO	TO	Signification
\$2048	\$6048	Mot de commande pour l'opération Ce registre peut contenir les valeurs suivantes: \$01 Demande d'initialisation du contrôleur \$02 Lecture d'un secteur \$04 Passage du lecteur en simple densité \$08 Ecriture d'un secteur \$10 Passage en double densité \$20 Recherche de la piste 0 \$40 Recherche de la piste dont le numéro se trouve en \$204A/\$604A \$80 Demande de vérification (écriture ou formatage)
\$2049	\$6049	Numéro de lecteur Le numéro de lecteur peut varier de 1 à 3 pour les lecteurs disquettes. Jusqu'à 4 pour les ordinateurs gérant le RamDisk (TO8, TO8D et TO9+)
\$204A	\$604A	Extension du numéro de piste Ce registre est généralement à 0 en guise d'extension à 16 bits du numéro de piste. Dans le cas de la lecture d'un secteur QDD réel, il sera différent de 0 et les registres \$204C-\$204D/\$604C-\$604D contiendront le numéro de secteur appelé (de 1 à 400).
\$204B	\$604B	Numéro de piste Ce registres contient le numéro de piste appelé, de 0 à 79 pour la double densité, de 0 à 39 pour l'???? et de 0 à 24 pour le QDD si l'appel de piste doit être interprété.
\$204C	\$604C	Numéro de secteur Ce registre contient le numéro de secteur appelé de 1 à 16. Idem pour le QDD si l'appel de secteur doit être interprété. Dans le cas où il s'agit du numéro de secteur QDD réel, les registres \$204C-\$204D/\$604C-\$604D contiennent le numéro de secteur (de 1 à 400) et le contenu du registre \$604A doit être différent de 0.
\$204D	\$604D	Facteur d'entrelacement pour formatage.
\$204E	\$604E	Code de l'erreur physique (retour de programme) Ce registres peut contenir les valeurs suivantes: \$01 Disquette protégée en écriture \$02 Erreur de piste \$04 Erreur de secteur \$08 Erreur sur les données \$10 Lecteur non prêt \$20 Erreur de vérification (écriture ou formatage) \$40 Contrôleur non prêt \$80 Disquette non formatée
\$204F-\$2050	\$604F-\$6050	Pointeur sur le tampon mémoire du secteur Lors d'un transfert de secteur, ce registre contient le pointeur sur l'espace mémoire pour le transfert.

Les registres suivants, dans l'intervalle \$2051-\$2058/\$6051-\$6058, sont officiellement répertoriés de la façon suivante:

\$2051-\$2052/\$6051-\$6052	(TRACK0)	Position de la tête du lecteur 0
\$2053-\$2054/\$6053-\$6054	(TRACK1)	Position de la tête du lecteur 1
\$2055-\$2056/\$6055-\$6056	(TRACK2)	Position de la tête du lecteur 2
\$2057-\$2058/\$6057-\$6058	(TRACK3)	Position de la tête du lecteur 3

Ils sont en fait utilisés d'une autre façon:

MO	TO	Signification
\$2051	\$6051	Non utilisé ou extension à 16 bits de la position des têtes pour le contrôleur 0
\$2052	\$6052	Position des têtes pour les lecteurs 0 et 1 Il s'agit en fait du numéro de piste sur laquelle sont positionnées les têtes (inférieures et supérieures simultanément) des lecteurs 0 et 1.
\$2053	\$6053	Non utilisé ou extension à 16 bits de la position des têtes pour le contrôleur 1
\$2054	\$6054	Position des têtes pour les lecteur 2 et 3 Il s'agit en fait du numéro de piste sur laquelle sont positionnées les têtes (inférieures et supérieures simultanément) des lecteurs 2 et 3.
\$2055-\$2056	\$6055-\$6056	Registre temporaire système Sur TO9, TO8, TO8D et TO9+, ce registre sert notamment comme tampon pour le pointeur de programme à exécuter en banque 1 du moniteur.
\$2057	\$6057	Bits d'activation du lecteur
\$2058	\$6058	Flag de densité

MO	TO	Signification
\$20D3-\$20E2	\$60D3-\$60E2	Table des entrelacements de piste pour formatage Cette plage est utilisée pour recevoir les 16 numéros de secteurs organisés selon l'entrelacement demandé. (Voir Le Floppy/Entrelacements système pour le Floppy)

Le RamDisk

Le RamDisk est un disque virtuel, composé exclusivement de mémoire vive, toujours en double densité et qui permet des accès plus rapides aux données que les supports magnétiques. De plus, il est créé en mémoire commutable à partir de la banque la plus éloignée, même si pour cela le système doit déplacer les fichiers CHG de plusieurs banques. Sa capacité peut être fixée au choix à 64, 128, 192, 256 ou 320 kilooctets. Chaque banque du RamDisk est divisée en 4 pistes de 16 secteurs double densité, comportant donc chacun 256 octets. Lors d'une opération RamDisk, la banque concernée est commutée dans l'espace cartouche (\$0000-\$3FFF sur TO); c'est d'ailleurs la raison pour laquelle le système d'exploitation RamDisk ne peut accéder à un secteur utilisateur s'il se trouve dans cette zone.

Les secteurs sont sauvegardés de l'adresse \$0000 vers l'adresse \$3FFF (entrelacement 1) pour les 4 pistes internes à chaque banque RamDisk tandis que les banques sont initialisées de la banque de numéro maximum vers la banque de numéro minimum.

Le nombre total de pistes varie en fonction de la capacité du RamDisk:

Taille du RamDisk	Nombre de pistes	Numéros de pistes
64	16	De 16 à 31
128	32	De 0 à 31
192	48	De 0 à 47
256	64	De 0 à 63
320	80	De 0 à 79

Dans la régularité physique d'un RamDisk, la piste 20 (FAT, catalogue et nom de disquette) fait exception: quelque soit la capacité du RamDisk, cette piste se trouve toujours dans la banque de numéro maximum, de l'adresse \$0000 à \$0FFF; pour un TO8 sans extension mémoire, c'est la banque 15; avec extension, c'est la banque 31. Voici donc les formules pour calculer le numéro de banque et l'offset d'un secteur RamDisk, BanqueMax étant la banque de numéro maximum et tout résultat d'opération restant entier:

Piste	RamDisk	Formule
20	Tous	Adresse = \$0000
	Tous	Banque = BanqueMax
Inférieure à 20	Tous	Adresse = (((Piste + 1) & 4) * 16) + (Secteur - 1) * 256
	64 k	Banque = BanqueMax - ((Piste - 15) / 4)
	> 64 k	Banque = BanqueMax - ((Piste + 1) / 4)
Supérieure à 20	Tous	Adresse = (((Piste & 4) * 16) + (Secteur - 1)) * 256
	64 k	Banque = BanqueMax - ((Piste - 16) / 4)
	> 64 k	Banque = BanqueMax - (Piste / 4)

Au sein du système d'exploitation, un programme ré-étalonne le numéro de piste dans le cas d'un RamDisk de 64 kilooctets. Pour ramener l'intervalle des numéros de pistes de type RamDisk de 16-31 à 0-15, on soustrait 16 au numéro de piste et on comble le passage de la piste 20 en retirant 1 au numéro obtenu si la piste traitée va au-delà de la piste 20. On obtient donc un intervalle de 0 à 14. Le numéro de piste 20 étant conservé en l'état par la routine - parce qu'étant traité comme celui de la piste de FAT et de catalogue par les routines suivantes - l'éventualité de rencontrer le numéro de piste 37 (\$25), altéré, lui, par la routine de conversion, sèmerait la confusion, puisqu'en retirant 16 et 1 à 37, on obtient le numéro de piste... 20! C'est pourquoi il est exclu du traitement.

Les formats sur supports magnétiques

Quand une information disque est transmise, en entrée ou en sortie, elle passe par le tampon prévu à cet effet, le « passe-plat » entre le lecteur et de processeur. L'écriture commencerait alors dès l'instant où la commande d'écriture serait activée. Vous vous réjouiriez d'avoir sauvegardé vos données sur le disque mais - surprise ! - lors de la lecture de vos données, vous vous rendriez compte que vous n'obtenez ce que vous avez sauvegardé qu'une fois de temps en temps, au mieux avec les données de fin de piste en début de lecture, au pire avec un décalage de un ou plusieurs bits.

Lorsqu'une disquette est formatée, le contrôleur commande l'écriture de deux types de mots sur la piste : les mots de données et les mots de synchronisation. Les mots de synchronisation sont des marquages qui permettent de placer la lecture ou l'écriture en bonne position pour la transmission des mots de données, un « bip » de répondeur. Après ce « bip », qui a toujours une structure binaire relative à la densité du formatage, la transmission est étalonnée de telle façon que le premier bit suivant capturé, le mot de synchronisation passé, soit bien toujours le bit de poids fort de la première donnée. Inutile de dire que sans les mots de synchronisation, les données de la disquette risqueraient d'être transmises avec un décalage dont la force varierait en fonction du bon vouloir du lecteur.

Il serait intéressant de savoir ce qui s'inscrit réellement sur la disquette.

En double densité, le mot de synchronisation utilisé est \$A10A mais ce n'est pas ce que le lecteur écrit sur la piste. Pour le contrôleur THMFC1 par exemple, le tampon est divisé en deux registres 8 bits, corrélatifs mais bien distincts. Le premier, auquel on peut accéder en lecture comme en écriture, permet de transférer la partie significative du mot (bits de données). Le deuxième registre, auquel on ne peut accéder qu'en écriture, permet de programmer la partie « synchro » (bits de synchro), qui décidera selon sa nature du type mot transmis, synchronisation ou donnée. La réalité physique est légèrement différente. Le mot \$A10A contient pourtant toutes les informations nécessaires pour dévoiler sa véritable identité. Prenons donc le premier octet de ce mot et intercalons un bit à 0 entre chaque bit. Agissons de même avec le deuxième mais en décalant le résultat d'un bit vers la gauche. Puis ajoutons les deux résultats :

```
$A1 => %10100001 => % 1 0 1 0 0 0 0 1
$0A => %00001010 => %0 0 0 0 1 0 1 0
-----
%0 1 0 0 0 1 0 0 1 0 0 0 1 0 0 1 => $4489
```

\$4489 est le mot physique inscrit sur la piste, mot de synchronisation utilisé en général sur les lecteurs acceptant la norme MFM.

Lorsque la tête magnétique rencontre un bit à l'état 1, il s'opère un changement de la magnétisation. Lorsqu'elle rencontre un bit à l'état 0, elle maintient l'état de magnétisation dans lequel elle se trouve. Les bits à 1 sont donc les seuls à générer des impulsions magnétiques. Mais les bits à 0 ne doivent pas être trop nombreux et provoquer un maintien de la magnétisation trop long si l'on ne veut pas sortir, malgré les irrégularités de fonctionnement d'un lecteur, du processus de synchronisation et donc qu'il soit transmis des données décalées. Or c'est justement la particularité du mot de synchronisation. (etc... [pas fini !])

Le QDD

Une disquette QDD est gérée en simple densité, simple face (paire) et dotée d'une seule piste concentrique. Lorsqu'une opération concernant les données est demandée, le moteur est lancé, la tête du lecteur se déplace sur la piste. Lorsqu'elle se trouve en fin de piste et que l'on demande une nouvelle opération, elle revient automatiquement en début de piste.

Structure d'une piste QDD

Sur la piste d'une disquette QDD sont écrits au formatage une amorce, les 400 secteurs numérotés de 1 à 400 et l'intervalle jusqu'à la fin de la piste:

	Nombre de données	Valeur de données
<i>Amorce de piste</i>		
Espaces	2796	\$16
<i>Secteur (400 fois)</i>		
Marquage identificateur	1	\$A5
N° secteur (Poids fort)	1	\$XX
N° secteur (Poids faible)	1	\$XX
Checksum identificateur *	1	\$XX
Espaces	10	\$16
Marquage secteur	1	\$5A
Données secteur	128	\$XX
Checksum secteur *	1	\$XX
Espaces	17	\$16
<i>Fin de piste</i>		
Espaces	Variable	\$16

* La valeur des checksums QDD est la somme de tout octet précédant ces checksums jusqu'au marquage inclus.

Pour conserver le même système d'exploitation logique pour tous les périphériques permettant la sauvegarde, périphérique cassette exclu, le système simule un découpage en pistes et secteurs de la piste QDD. Ainsi, les secteurs QDD peuvent être sollicités selon la norme d'appel de tout secteur en plaçant un numéro de piste variant entre 0 et 24 en DKTRK (\$604A-4B) et un numéro de secteur variant entre 1 et 16 en DKSEC (\$604C), en même temps que de pouvoir être sollicités en plaçant leur numéro réel variant entre 1 et 400 en DKSEC (\$604C-4D) si le registre \$604A est différent de 0.

Correspondance système des secteurs QDD

Une routine se charge d'interpréter les numéros de piste (\$604B) et secteur (\$604C) en numéro de secteur QDD et est sollicitée par les routines du QDD lorsque le contenu de \$604A est à 0. Afin d'éclaircir cette sombre procédure de calcul, voici le tableau récapitulatif des formules utilisées pour les correspondances QDD:

Piste	Secteur	Correspondance QDD
0	1-4	(Secteur*4)+381
0	5-8	(Secteur*4)+367
0	9-12	(Secteur*4)+350
0	13-16	(Secteur*4)+336

Généralités du système physique

1	Tous	(Secteur*4)+317
2	Tous	(Secteur*4)-2
3-8	Tous	(Secteur*4)+((8-Piste)*64)
9-14	Tous	(Secteur*4)+((14-Piste)*64)-1
15-19	Tous	(Secteur*4)+((20-Piste)*64)-2
20-24	Tous	(Secteur*4)+((Piste-20)*64)-3

On peut remarquer que l'entrelacement système d'une piste QDD est de 4, pour permettre les différentes opérations intermédiaires entre chaque appel et traiter une piste moniteur en une seule passe. Mais le perfectionnement ne s'arrête pas là. Sachant que la piste 20 est la première à être appelée lors d'une opération sur un fichier, cette piste commence donc au secteur 1. Les pistes suivant la piste 20 sont les premières à être sollicitées : l'entrelacement de 4 par rapport au numéro du dernier secteur de la piste 20 est respecté. Viennent ensuite les pistes précédant la piste 20 : l'entrelacement est respecté toutes les 6 pistes, tant que faire se peut. On voit donc bien que l'agencement spécifique d'une piste QDD permet de traiter les données de fichiers avec un minimum de passages.

Ci-après, la table des correspondances entre ces deux modes d'appel, numéros de secteurs horizontalement et de "pistes" verticalement:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	385	389	393	397	387	391	395	399	386	390	394	398	388	392	396	400
1	321	325	329	333	337	341	345	349	353	357	361	365	369	373	377	381
2	2	6	10	14	18	22	26	30	34	38	42	46	50	54	58	62
3	324	328	332	336	340	344	348	352	356	360	364	368	372	376	380	384
4	260	264	268	272	276	280	284	288	292	296	300	304	308	312	316	320
5	196	200	204	208	212	216	220	224	228	232	236	240	244	248	252	256
6	132	136	140	144	148	152	156	160	164	168	172	176	180	184	188	192
7	68	72	76	80	84	88	92	96	100	104	108	112	116	120	124	128
8	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64
9	323	327	331	335	339	343	347	351	355	359	363	367	371	375	379	383
10	259	263	267	271	275	279	283	287	291	295	299	303	307	311	315	319
11	195	199	203	207	211	215	219	223	227	231	235	239	243	247	251	255
12	131	135	139	143	147	151	155	159	163	167	171	175	179	183	187	191
13	67	71	75	79	83	87	91	95	99	103	107	111	115	119	123	127
14	3	7	11	15	19	23	27	31	35	39	43	47	51	55	59	63
15	322	326	330	334	338	342	346	350	354	358	362	366	370	374	378	382
16	258	262	266	270	274	278	282	286	290	294	298	302	306	310	314	318
17	194	198	202	206	210	214	218	222	226	230	234	238	242	246	250	254
18	130	134	138	142	146	150	154	158	162	166	170	174	178	182	186	190
19	66	70	74	78	82	86	90	94	98	102	106	110	114	118	122	126
20	1	5	9	13	17	21	25	29	33	37	41	45	49	53	57	61
21	65	69	73	77	81	85	89	93	97	101	105	109	113	117	121	125
22	129	133	137	141	145	149	153	157	161	165	169	173	177	181	185	189
23	193	197	201	205	209	213	217	221	225	229	233	237	241	245	249	253
24	257	261	265	269	273	277	281	285	289	293	297	301	305	309	313	317

Remarque : Le boot QDD se trouve au secteur QDD numéro 68, c.a.d en piste 7 secteur 1 et non en piste 0 secteur 1 comme pour le floppy.

Le floppy

Les disquettes utilisées dans les lecteurs floppy sont constituées de plusieurs pistes bouclées. Leur nombre peut varier jusqu'à 80 pistes par face. Chaque piste comporte 16 secteurs constitués de 128 ou 256 octets selon la densité.

Structure d'une piste simple densité

	Nombre de données	Valeur de données
<i>Amorce de piste</i>		
Intervalle	16	\$FF
<i>Secteur (16 fois)</i>		
Espaces	6	\$00
Marquage identificateur *	1	\$FEC7
N° piste	1	\$XX
N° de face	1	Toujours \$00
N° secteur	1	\$XX
Taille du secteur	1	\$00
CRC identificateur	2	\$XXXX

Généralités du système physique

Intervalle	12	\$FF
Espaces	6	\$00
Marquage de données *	1	\$FBC7
Données secteur	128	\$XX
CRC secteur	2	\$XXXX
Intervalle	22	\$FF
<i>Fin de piste</i>		
Intervalle	Variable	\$FF

* Les marquages identificateur et de données assurent en même temps la synchronisation

Structure d'une piste double densité

	Nombre de données	Valeur de données
<i>Amorce de piste</i>		
Intervalle	30	\$4E
<i>Secteur (16 fois)</i>		
Espaces	12	\$00
Synchronisation	3	\$A10A
Marquage identificateur	1	\$FEFF
N° piste	1	\$XX
N° de face	1	Toujours \$00
N° secteur	1	\$XX
Taille du secteur	1	\$01
CRC identificateur	2	\$XXXX
Intervalle	22	\$4E
Espaces	12	\$00
Synchronisation	3	\$A10A
Marquage de données	1	\$FBFF
Données secteur	256	\$XX
CRC secteur	2	\$XXXX
Intervalle	44	\$4E
<i>Fin de piste</i>		
Intervalle	Variable	\$4E

Entrelacements système pour floppy

Les entrelacements ont la particularité de permettre la temporisation des accès aux secteurs. Selon le temps d'exécution du traitement des informations entre deux appels disque et si l'entrelacement est judicieusement choisi, il assure un positionnement presque immédiat sur le secteur suivant.

Table des entrelacements système de 1 à 15

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	1	9	2	10	3	11	4	12	5	13	6	14	7	15	8	16
3	1	12	7	2	13	8	3	14	9	4	15	10	5	16	11	6
4	1	5	9	13	2	6	10	14	3	7	11	15	4	8	12	16
5	1	14	11	8	5	2	15	12	9	6	3	16	13	10	7	4
6	1	9	4	12	7	15	2	10	5	13	8	16	3	11	6	14
7	1	8	15	6	13	4	11	2	9	16	7	14	5	12	3	10
8	1	3	5	7	9	11	13	15	2	4	6	8	10	12	14	16
9	1	10	3	12	5	14	7	16	9	2	11	4	13	6	15	8
10	1	9	6	14	3	11	8	16	5	13	2	10	7	15	4	12
11	1	4	7	10	13	16	3	6	9	12	15	2	5	8	11	14
12	1	5	9	13	4	8	12	16	3	7	11	15	2	6	10	14
13	1	6	11	16	5	10	15	4	9	14	3	8	13	2	7	12
14	1	9	8	16	7	15	6	14	5	13	4	12	3	11	2	10
15	1	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2

Les temporisations

Si une routine gérant des opérations disquette est développée, il est impératif de respecter les temporisations imposées par le constructeur, afin que les composants électroniques fragiles ou non protégés ne soient pas détériorés.

Le Cyclic Redundancy Check

Nécessité du CRC

Comment détecter efficacement une erreur dans une zone de données ?

Si l'on utilisait un checksum, l'opération consisterait donc à ajouter les mots de données les uns aux autres, à ajouter éventuellement une valeur fixe et sauvegarder ce résultat en fin de secteur. Mais on sait que les irrégularités mécaniques d'un lecteur peuvent provoquer d'erreur que sur un seul bit, les données en étant du coup ou décalées, ou altérées. Pour peu que deux bits de même position relative à la taille du checksum soient intervertis, le secteur sera déclaré correct, bien que les données qu'il contient aient changé. Dans le cas d'un checksum, une erreur dans une zone de données peut malheureusement être facilement contrebalancée par une autre. Or il est nécessaire d'élaborer un système de détection d'erreur plus sûr.

Le CRC permet la détection des erreurs dans la zone d'identification et la zone de données de chaque secteur. Son intérêt est de tester chaque bit de la zone de données et de détecter près de 100% des erreurs au bit près.

Principe de calcul du CRC

Le principe de calcul est plus simple qu'il n'y paraît.

Le calcul débute à partir d'une valeur 16 bits pour l'initialisation, que nous appellerons CRC, puisque c'est sur lui que le calcul agit et qu'il contiendra la valeur finale. La routine de calcul va tester alors simultanément les bits du CRC et les bits de la zone à traiter, du bit de poids fort vers le bit de poids faible. Si le bit testé du CRC est à 1 et que le bit testé de la zone est à 0, le programme va opérer un ou exclusif entre le résultat et le CRC-CCITT (le générateur polynomial utilisé pour la plupart des calculs de CRC 16 bits soit \$11021, pris à \$1021 puisque le bit de poids fort est hors champ d'un registre 16 bits) à partir de la position courante plus un bit. Si le bit testé du résultat est à 0 et que le bit testé de la zone est à 1, la même opération sera effectuée. Dans les deux autres cas, il ne se produira rien.

Tout au long du calcul, le CRC se construit et change d'état en fonction de son propre état et de celui des données qu'il vérifie, conservant une trace de la différence entre ses propres variations et les données rencontrées. Le bit 0 du CRC-CCITT est fondamental. Il est le seul bit dont on peut être certain qu'il positionnera à 1 le bit 0 du CRC s'il se produit un changement dans la comparaison. Sa seule présence ne garantirait pas néanmoins la détection d'une erreur, puisqu'il suffirait de passer 16 bits pour que l'information soit perdue et que le secteur soit déclaré correct. L'« enroulement » du test activé par les autres bits (bits 12 et 5) assure plus ou moins une récupération de l'erreur détectée au bit 0. En fin de calcul, le résultat a donc conservé une trace réelle ou sous-entendue de l'erreur.

Le principe de calcul est illustré par la routine suivante, bien que non officielle (les données de la zone à traiter sont perdues pour ajouter à la clarté du programme) :

```

CRC      LDD      #$E295  Fixe la valeur de départ du CRC
          LDX      #256    256 octets à traiter
          LDU      #ZONE   Pointeur sur la zone à traiter
* Fixe la taille pour le nombre de bits à tester par octets
CRC0     LDY      #8      8 bits par octet
*-----
*----- Principe de calcul
* Teste le bit de CRC (et positionne le CRC pour le XOR avec le CRC-CCITT)
CRC1     ASLB                      | Décale le CRC
          ROLA                      | d'un bit vers la gauche
          BCS      CRC2            Si bit à 1, teste état du bit de zone
* Si bit de CRC à 0
          ASL      ,U              Teste le bit de zone
          BCS      CRC3            Si bit de zone à 1, XOR avec le CRC-CCITT
          BRA      CRC4            Sinon, bit suivant
* Si bit de CRC à 1
CRC2     ASL      ,U              Teste le bit de zone
          BCS      CRC4            Si bit de zone à 0, XOR avec le CRC-CCITT. Sinon, suivant
* Commute les bits du CRC-CCITT
CRC3     EORA     #$10            Ou exclusif du CRC avec le CRC-CCITT (poids fort)
          EORB     #$21            Ou exclusif du CRC avec le CRC-CCITT (poids faible)
*-----
* Bit suivant
CRC4     LEAY     -1,Y            | Si les 8 bits n'ont pas été traités,
          BNE     CRC1            | boucle sur le bit suivant
* Donnée suivante
          LEAU     1,U            Pointeur sur la donnée suivante
          LEAX     -1,X            | Si toutes les données de la zone n'ont pas été traitées,
          BNE     CRC0            | boucle sur la donnée suivante
          RTS
* Zone de stockage des données à traiter
ZONE     RMB     256
    
```

Généralités du système physique

Il est à savoir que cette procédure de calcul est suivie aussi pour les calculs de CRC sur disquette PC formatée en MFM double densité. Le calcul de CRC des données ne se fait plus alors sur une zone de 256, mais de 512 octets.

Les routines de calcul du CRC dans les différentes ROM contrôleur peuvent varier suivant l'évolution de leur optimisation par les programmeurs au fur et à mesure des développements.

Optimisation du calcul du CRC

En vue d'optimiser la vitesse du programme, plutôt que de traiter bit par bit, la procédure ci-dessus revient à dire que si l'on calcule le masque M adéquat, l'opération de calcul du CRC par groupe de 8 bits revient à faire séquentiellement :

$$\text{CRC} = (\text{CRCLow} * 2^8) \text{ XOR } (\text{M} * 2^{12}) \text{ XOR } (\text{M} * 2^5) \text{ XOR } (\text{M} * 2^1)$$

... puisque le CRC-CCITT a les bits 12, 5 et 0 positionnés à 1. Le masque M devrait tenir compte en même temps des bits entrants et des bits sortants qui, comme nous l'avons déjà vu, sont toujours pris en ligne de compte du fait de l'enroulement. Le calcul du masque M serait donc :

$$\text{M} = \text{CRCHigh XOR donnée8bits XOR (donnée8bits/2^4)}$$

... ce qui donne le résultat suivant :

```
* Calcul du CRC
* D = Valeur de départ du CRC
* Y = Nombre d'octets de données
* U = Pointeur sur les données
MCRC      STD   CRC           Mémorise le CRC
MCRC0     LDB   CRCH         Lit le CRCHigh
          EORB  ,U+         Masque pour les bits sortants
          PSHS  B           |
          LSRB             |
          LSRB             | Masquage des
          LSRB             | bits entrants
          LSRB             |
          EORB  ,S+         |
* Masque pour M*2^1
          CLRA             Pour 16 bits
          STD   MASK1       Mémorise le masque
* Masque pour M*2^5
          ASLB             |
          ROLA             |
          ASLB             |
          ROLA             |
          ASLB             Décalage de 5 bits
          ROLA             vers la gauche
          ASLB             |
          ROLA             |
          ASLB             |
          ROLA             |
          STD   MASK5       Mémorise le masque
* Masque pour M*2^12
          ASLB             |
          ROLA             |
          ASLB             |
          ROLA             |
          ASLB             |
          ROLA             |
          ASLB             Décalage de 7 bits de plus
          ROLA             vers la gauche
          ASLB             |
          ROLA             |
          ASLB             |
          ROLA             |
          STD   MASK12      Mémorise le masque
* Calcul du CRC
          LDA   CRCL         Lit le CRCLow
          EORA  MASK12       |
          EORA  MASK5         Effectue le masquage pour le CRCHigh
          EORA  MASK1         |
          CLRB             Efface le CRCLow
          EORB  MASK12+1     |
          EORB  MASK5+1      Effectue le masquage pour le CRCLow
          EORB  MASK1+1      |
          STD   CRC           Mémorise le CRC
* Calcul suivant
```

Généralités du système physique

	LEAY	-1,Y	Si pas fin des données,
	BNE	MCRC0	calcul de CRC suivant
	RTS		
CRC	EQU	*	Résultat du calcul du CRC
CRCH	RMB	1	Poids fort du CRC
CRCL	RMB	1	Poids faible du CRC
MASK1	RMB	2	Masque pour M*2^1
MASK5	RMB	2	Masque pour M*2^5
MASK12	RMB	2	Masque pour M*2^12

On comprend mieux dès lors le programme de calcul du CRC contenu notamment dans la 1ère banque des contrôleurs internes des T08, T08D et T09+ :

```

* Calcul du CRC
* D = Valeur de départ du CRC
* Y = Nombre d'octets de données
* U = Pointeur sur les données
MCRC      PSHS  B,A      Empile le CRC
MCRC0     LDA   ,S      Lit le CRCHigh
          EORA  ,U+     Masque les bits sortants
          PSHS  A      |
          LSRA          |
          LSRA          |
          LSRA          | Masque les bits entrants
          LSRA          |
          EORA  ,S      |
          STA   ,S      |
          TFR  A,B      Masque dans B
          LSRB          |
          LSRB          | Ajuste masque pour M*2^5
          LSRB          |
          ASLA          |
          ASLA          | Ajuste masque
          ASLA          | pour M*2^12
          ASLA          |
          EORB  2,S     | Ajuste poids fort du
          STB   2,S     | masque M*2^5
          TFR  A,B     Poids fort pour M*2^5
          EORA  2,S     XOR poids fort pour M*2^12
          STA   1,S     Sauve poids fort CRC
          ASLB          Poids faible pour M*2^5
          EORB  ,S     XOR poids faible pour M*2^1
          STB   2,S     Sauve poids faible CRC
          LEAS  1,S     Rétablit la pile
          LEAY -1,Y     | Caractère
          BNE  MCRC0    | suivant
          PULS A,B,PC   Récupère le CRC dans D et retour

```

GENERALITES DU SYSTEME LOGIQUE

Chaque secteur a une capacité d'un maximum de 256 octets.

Structure du secteur de nom de disquette

Ce secteur se trouve au secteur 1 de la piste 20 et est initialisé avec la valeur \$FF.

Les 8 premiers octets du secteur contiennent le nom de la disquette, cadré à gauche et complété par des blancs.

Le reste du secteur est inutilisé.

Structure de la FAT

La table d'allocation des fichiers (FAT) se trouve au secteur 2 de la piste 20.

Les données contenues dans la FAT ont la structure suivante :

- Premier octet à \$00.
- Zone de la FAT inférieure (en deçà de la piste 20). La longueur de cette zone est fixe (40 blocs). La zone exclue, si elle existe, est initialisée à \$FE, la zone libre est initialisée à \$FF.
- Réserve de 2 blocs pour la piste 20 avec la valeur \$FE.
- Zone de la FAT supérieure (au-delà de la piste 20). La longueur de cette zone est variable selon la capacité du disque. La zone libre est initialisée à \$FF.
- Exclusion du reste du secteur avec la valeur \$FE

Lors de la création d'un fichier, le numéro du premier bloc logique libre est sauvé dans à l'offset 13 de l'entrée dans le catalogue. Les 8 secteurs du bloc sont mis à disposition pour la sauvegarde. Si la sauvegarde nécessite un bloc de plus, le numéro du bloc logique suivant est écrit à l'offset + 1 du numéro de bloc logique précédent dans la FAT. Sinon, on écrit à cet offset la valeur \$C0 plus le nombre de secteurs utilisés dans le bloc.

Structure d'une entrée de fichier dans le catalogue

Le catalogue commence toujours à partir du secteur 3 de la piste 20 et se termine au secteur 16. Une entrée de fichier a une structure invariable de 32 octets. En simple densité, chaque secteur contiendra 4 entrées consécutives, en double densité 8.

- Offsets 0-7 : Nom du fichier, cadré à gauche, complété par des blancs.
 - Particularité de l'offset 0 : \$00 Entrée de fichier effacée
Lors de la recherche d'un fichier, cette entrée doit être passée. Lors de la création d'un fichier, elle est disponible.
 - \$FF Fin du catalogue
Lors de la recherche d'un fichier, ce flag indique la fin du catalogue. Lors de la création d'un fichier, la place est disponible.
 - Autre Entrée valide de fichier
- Offsets 8-10 : Extension (suffixe) du fichier sur 3 octets, cadré à gauche, complété par des blancs.
 - BAS Fichier Basic
 - DAT Fichier de donnée
 - ASM Fichier assembleur
 - BIN Fichier binaire
 - MAP Fichier graphique
 - CHG Fichier cartouche
 - CFG Fichier palette
 - BAT Fichier auto-exécutable (AUTO.BAT)
 - POL Fichier Polyphonia
 - etc...
- Offset 11 : Type de fichier :
 - \$00 pour un programme Basic ASCII et binaire
 - \$01 pour des données Basic en ASCII
 - \$02 pour un programme en langage machine (binaire)
 - \$03 pour un fichier assembleur édité en ASCII

Il est à noter que, depuis lors, d'autres types de fichier ont été créés. Par exemple, \$0A pour les fichiers Paragraphe. Il est pourtant préférable d'utiliser les types standard que de créer un type particulier de fichier si cela ne se justifie pas. Les types courants combinés avec le flag de fichier sont les suivants :

 - \$00 \$00 : Fichier Basic standard
 - \$00 \$FF : Fichier Basic sauvé en ASCII
 - \$02 \$00 : Fichier binaire (exécutable, graphique, etc...). La particularité de ce fichier est de ne pouvoir dépasser linéairement les 64 kilooctets, mais bénéficie de la présence de la taille du fichier en tête de fichier et éventuellement de son adresse d'implantation.
 - \$03 \$FF : Fichier Assembleur standard
 - \$01 \$FF : Fichier de données en ASCII. Utilisé pour les fichiers de grande taille.

Généralités du système logique

Il est fortement conseillé de se cantonner à ces combinaisons lors de la création d'un format, les fichiers binaires ayant une structure particulière, et éviter de fait la confusion lors de sa lecture ou de son écriture par un autre logiciel. Préférez personnaliser votre fichier par le suffixe.

- Offset 12 : Flag de fichier
 - \$00 pour du binaire
 - \$FF pour de l'ASCII
- Offset 13 : Numéro du premier bloc logique du fichier.
- Offsets 14-15 : Nombre d'octets utilisés dans le dernier secteur du fichier.
Lors de sa création, un fichier peut contenir exactement dans un ou plusieurs blocs. Ce cas entraîne une aberration notable : un bloc de plus sera alloué, un secteur dans le bloc et le nombre d'octets dans le dernier secteur de fichier sera mis à 0. Il se peut donc que le système d'exploitation assigne des blocs entiers... ne contenant aucune donnée !
- Offsets 16-23 : Commentaire du fichier, cadré à gauche, complété par des blancs. Sinon à \$00.
- Offset 24 : Jour de création du fichier, de 1 à 31. Sinon 0.
- Offset 25 : Mois de création du fichier, de 1 à 12. Sinon 0.
- Offset 26 : Année de création du fichier, de 0 à 99. L'année de création n'est valide que si le jour et le mois de création sont valides.
- Offsets 27-29 : Libre.
- Offset 30 : Mode de lancement du fichier CHG.
 - Bits 0-5 : Nombre de banques nécessaires pour l'application CHG
 - Bit 7 : Exécution à froid automatique de l'application si à 1. ATTENTION ! L'exécution automatique de l'application ne s'exécute pas en JMP mais en JSR. Le programme doit donc en principe protéger l'état de tous les registres utilisés et rendre la main à « Appel de programmes » par un RTS. Une initialisation préalable est donc possible avant le lancement de l'application.
- Offset 31 : Checksum du nom de fichier pour un fichier CHG (somme des 8 octets du nom sur 8 bits).
Ce checksum protège le nom du fichier CHG. Si le fichier CHG est renommé par exemple sous Basic, il ne sera plus traité comme une application valide. Son nom ne s'affichera plus dans la fenêtre d'« appel de programmes », si le checksum n'a pas été mis à jour.

Le fichier SCRATCH.DOS

Le nom de fichier "SCRATCH.DOS" est utilisé pour sauver un fichier dans le mode avec écrasement. Le système logique sauve le fichier sous le nom de "SCRATCH.DOS"; si le fichier a été sauvé sans incidents, l'entrée de la sauvegarde précédente est effacée - si elle existe - et le fichier "SCRATCH.DOS" est renommé avec le nom de fichier courant. Ceci afin d'avoir toujours sur le disque une version valide du fichier si la sauvegarde n'a pu s'effectuer correctement.

Chargement de la FAT - Cas d'une FAT de 160 octets

L'intérêt est d'accéder à tous les octets d'une FAT double densité à l'aide un mode d'adressage indexé signé avec accumulateur 8 bits (dont l'opérande serait de type "A,X").

Pour se faire, on swape les deux moitiés de la FAT et on fait pointer le registre d'adressage sur le milieu de cette FAT arrangée. Ainsi, si la valeur de l'accumulateur 8 bits va de \$00 à \$7F, on accède aux positions 128 à 255 du secteur et donc aux 128 premiers octets de la FAT. Si la valeur de l'accumulateur va de \$80 à \$FF, on accède aux positions 0 à 127 et donc aux 32 derniers octets de la FAT.

Recherche d'un bloc libre dans la FAT pour le floppy

Deux procédures distinctes entrent en jeu.

La première cherche le bloc libre le plus proche du bloc courant en tendant vers le bloc 0 si le bloc courant est dans la première moitié de la FAT et en tendant vers le bloc maximum si le bloc courant est dans la deuxième moitié de la FAT. Cette procédure est utilisée pour l'allocation d'un bloc quelconque du fichier.

Si cette recherche est infructueuse, une recherche sur la totalité de la FAT est engagée en testant simultanément la disponibilité des blocs se trouvant de part et d'autre du bloc du milieu et en tendant vers les extrémités de la FAT. Cette procédure est utilisée pour l'allocation du premier bloc d'un fichier ou en dernier recours pour l'allocation d'un bloc quelconque.

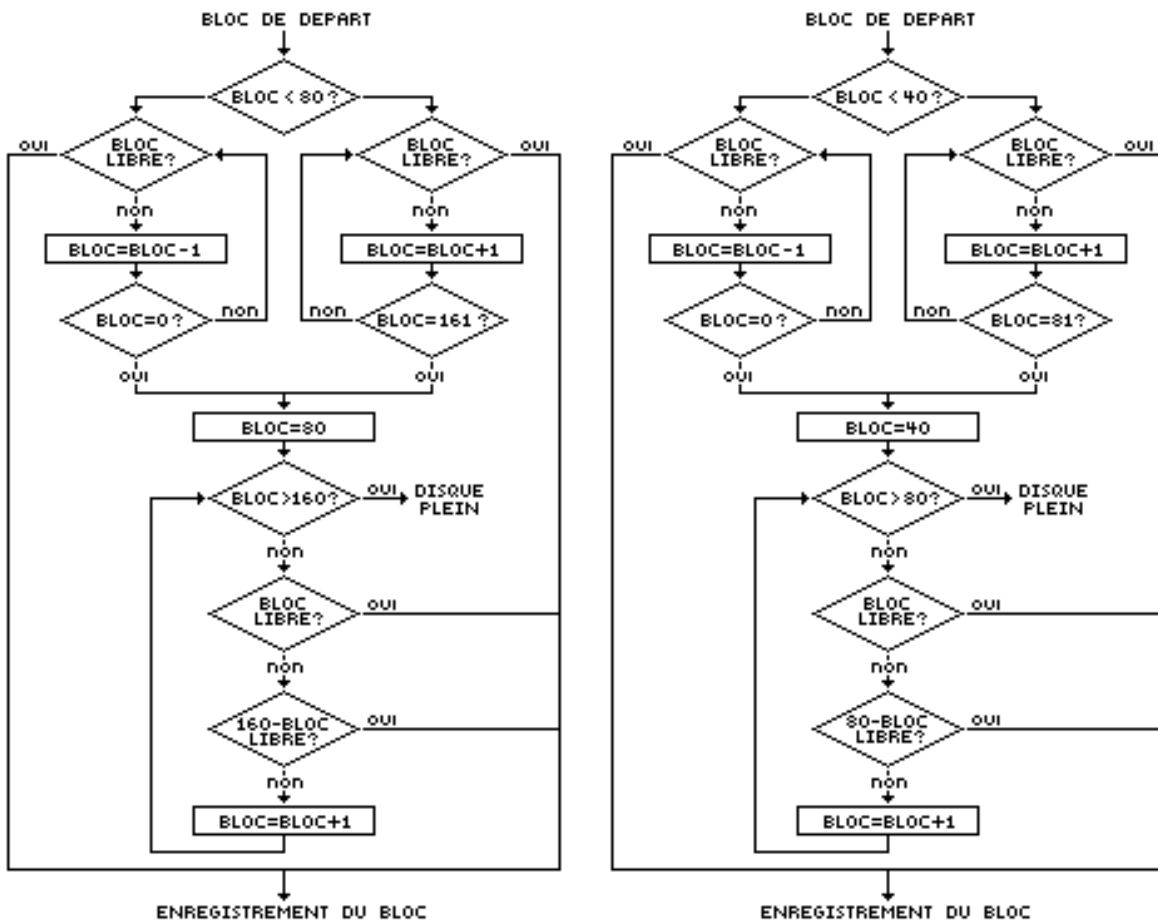
Dans le cas d'une FAT simple densité, il faut bien évidemment veiller à ce que cette FAT soit précédée d'au moins 128 octets de valeur différente de \$FF, sinon le programme risque de retourner un bloc libre alors que la FAT est pleine. Généralement, les mémoires tampons des secteurs de données et de FAT sont consécutifs et initialisés à 0.

Cette routine peut donner une idée de la procédure de recherche d'un bloc libre sur Thomson, mais ignore visiblement la préférence de l'ExtraMoniteur à sélectionner les blocs de numéro pair.

Pour disque 80 pistes

Pour disque 40 pistes

Généralités du système logique



Registres moniteur

La définition de certains de ces registres et valeurs (dont le commentaire est marqué d'un *) est imposée par les routines du moniteur, celle des autres étant en fait laissée au choix du programmeur (sauf pour les registres de \$60F9 à \$60FD, qui ont des usages différents selon les routines utilisées).

Néanmoins, ils ont été utilisés comme suit dans la plupart des routines élaborées par les programmeurs sur Thomson:

MO	TO	Signification
\$20E5	\$60E5	* Codes d'erreur du système logique 0 : Pas d'erreur 1 : Commande inconnue 2 : Disque absent 3 : * Erreur d'entrée-sortie 4 : * Fichier inexistant 5 : * Disque plein Les extensions éventuelles de ces codes sont laissées au bon vouloir des programmeurs.
\$20E6	\$60E6	Code de l'opération logique demandée 0 : - Initialisation du registre - * Clôture du fichier 1 : * Lecture d'un fichier 2 : * Ecriture d'un fichier sans écrasement 3 : * Ecriture d'un fichier avec écrasement 4 : Affichage du répertoire d'un disque 5 : Formatage de la disquette Les extensions éventuelles de ces codes sont laissées au bon vouloir des programmeurs.
\$20E7-\$20E8	\$60E7-\$60E8	Pointeur sur le descripteur de fichier (11 caractères)
\$20E9-\$20EA	\$60E9-\$60EA	* Pointeur sur la mémoire tampon du secteur
\$20EB	\$60EB	* Type de fichier (\$00, \$01, \$02, \$03...)
\$20EC	\$60EC	* Flag de fichier (\$00 ou \$FF)

Généralités du système logique

\$20ED-\$20EE	\$60ED-\$60EE	* Pointeur sur la mémoire tampon de la FAT
\$20EF	\$60EF	Numéro de lecteur
\$20F0	\$60F0	* Code de l'opération logique en court
\$20F1-\$20F2	\$60F1-\$60F2	Pointeur sur la fin des données dans la mémoire tampon de secteur
\$20F3-\$20F4	\$60F3-\$60F4	Pointeur courant sur les données
\$20F5	\$60F5	Compteur de secteur
\$20F6	\$60F6	* Numéro du bloc courant/Checksum cassette
\$20F7-\$20F8	\$60F7-\$60F8	* Nombre d'octets dans le dernier secteur du fichier
\$20F9	\$60F9	Dernier secteur + 1 du bloc courant
\$20FA	\$60FA	Numéro du secteur courant des données
\$20FB-\$20FC	\$60FB-\$60FC	Numéro de la piste courante des données
\$20FD	\$60FD	Flag d'initialisation du contrôleur \$00 : Contrôleur non initialisé \$55 : Contrôleur initialisé

Les entrées standard du système d'exploitation du contrôleur

\$A004/\$E004 : Fonctions standard

Cette entrée donne la possibilité d'exécuter diverses opérations fondamentales sous les modalités suivantes:

Reset du contrôleur (tout lecteur sauf RamDisk)

Registres d'entrée :

- \$2048/\$6048 Code de commande \$01

Registres de retour :

- C de CC à 0, \$604E contient le code de densité du lecteur (« D » si double densité, « C » si simple). Si erreur, C de CC à 0 et code d'erreur dans \$204E/\$604E.

Lecture d'un secteur (tout lecteur)

Registres d'entrée :

- \$2048/\$6048 Code de commande \$02
- \$2049/\$6049 Numéro du lecteur
- \$204A-\$204B/\$604A-\$204B Numéro de piste
- \$204C/\$604C Numéro de secteur (de 1 à 16)
- \$204F-\$2050/\$604F-\$2050 Pointeur sur la mémoire tampon de réception

Registres d'entrée pour QDD avec numéro de secteur réel:

- \$2048/\$6048 Code de commande \$02
- \$2049/\$6049 Numéro du lecteur
- \$204A/\$604A \$FF
- \$204B-\$204C/\$604B-\$604C Numéro de secteur (de 1 à 400)
- \$204F-\$2050/\$604F-\$2050 Pointeur sur la mémoire tampon de réception

Registres de retour :

- C de CC à 0. Si erreur, C de CC à 1 et code d'erreur dans \$204E/\$604E.

Passage en simple densité (floppy exclusivement)

Registres d'entrée :

- \$2048/\$6048 Code de commande \$04

Registres de retour :

- Aucun

Écriture d'un secteur (tout lecteur)

Registres d'entrée :

- \$2048/\$6048 Code de commande \$08 (\$88 pour écriture avec vérification)
- \$2049/\$6049 Numéro du lecteur
- \$204A-\$204B/\$604A-\$204B Numéro de piste
- \$204C/\$604C Numéro de secteur
- \$204F-\$2050/\$604F-\$2050 Pointeur sur la mémoire tampon de réception

Registres d'entrée pour QDD avec numéro de secteur réel:

- \$2048/\$6048 Code de commande \$08 (\$88 pour écriture avec vérification)
- \$2049/\$6049 Numéro du lecteur
- \$204A/\$604A \$FF
- \$204B-\$204C/\$604B-\$604C Numéro de secteur (de 1 à 400)
- \$204F-\$2050/\$604F-\$2050 Pointeur sur la mémoire tampon de réception

Registres de retour :

- C de CC à 0. Si erreur, C de CC à 1 et code d'erreur dans \$204E/\$604E.

Passage en double densité (floppy exclusivement)

Registres d'entrée :

Généralités du système logique

- \$2048/\$6048 Code de commande \$10
- Registres de retour :
- Aucun

Recherche de la piste 0 (floppy exclusivement)

- Registres d'entrée :
- \$2048/\$6048 Code de commande \$20
- Registres de retour :
- C de CC à 0. Si erreur, C de CC à 1 et code d'erreur dans \$204E/\$604E.

Recherche d'une piste quelconque (floppy exclusivement)

- Registres d'entrée :
- \$2048/\$6048 Code de commande \$20
 - \$204A-\$204B/\$604A-\$204B Numéro de piste (sur 16 bits)
- Registres de retour :
- C de CC à 0. Si erreur, C de CC à 1 et code d'erreur dans \$204E/\$604E.

Remarque : dans le manuel, il est inscrit que le passage de densité peut provoquer une erreur mais elle n'est pas délivrée par les programmes eux-mêmes, puisque ceux-ci ne renvoient jamais d'erreur. Une erreur, donc, que seule la logique impose...

\$A007/\$E007 : Lancement du boot

- Registres d'entrées :
- Aucun
- Registres de retour :
- Aucun

\$A00A/\$E00A : Formatage

- Registres d'entrée :
- \$2048/\$6048 Code \$80 si vérification demandée, sinon \$00 (floppy exclusivement)
 - \$204D/\$604D Facteur d'entrelacement de 1 à 15 (floppy exclusivement)
- Registres de retour :
- C de CC à 0. Si erreur, C de CC à 1 et code d'erreur dans \$204E/\$604E.

\$A00D/\$E00D : Chargement de la FAT

- Registres d'entrée :
- \$20ED/\$60ED Pointeur sur la mémoire tampon de FAT (128 ou 256 octets selon la densité)
- Registres de retour :
- C de CC à 1 si erreur de chargement. 0 si pas d'erreur.

\$A010/\$E010 : Ouverture d'un fichier

- Registres d'entrées :
- La FAT a été chargée (entrée \$A00D/\$E00D)
 - \$20E9-\$20EA/\$60E9-\$60EA Pointeur sur la mémoire tampon de secteur
 - \$20E7-\$20E8/\$60E7-\$60E8 Pointeur sur le descripteur de fichier (11 octets)
 - \$20F0/\$60F0 Mode d'ouverture du fichier
 - \$20EB/\$60EB Type du fichier recherché
 - \$20EC/\$60EC Flag du fichier recherché
- Registres de retour :
- C de CC à 1 si erreur de chargement. Le registre \$20E5/\$60E5 contient le code d'erreur du système d'exploitation logique \$03 si I/O Error (aussi dans A). 0 si pas d'erreur
 - \$20F9/\$60F9 Numéro du secteur de l'entrée du fichier dans le catalogue (aussi dans B). Si une sauvegarde avec écrasement est demandée (\$03 dans le registre \$20F0/\$60F0), c'est le fichier nommé « SCRATCH.DOS » qui sera recherché. Si l'entrée du fichier n'a pas été trouvée, ce registre est à 0. ATTENTION! Le registres C de CC n'est pas mis à 1 dans ce dernier cas.
 - \$20F6/\$60F6 Numéro du premier bloc de fichier
 - \$20F5/\$60F5 Compteur de secteur à 0
 - \$20F7-\$20F8/\$60F7-\$60F8 Nombre d'octets dans le dernier secteur du fichier (sur 16 bits)
 - \$20FA-\$20FB/\$60FA-\$60FB Pointeur sur l'entrée du fichier dans le secteur (aussi dans Y)

\$A013/\$E013 : Effacement du fichier courant

- Registres d'entrée :
- La FAT a été chargée (entrée \$A00D/\$E00D)
 - Le fichier à effacer doit avoir été recherché dans le catalogue au préalable (entrée \$A010/\$E010).
- Registres de retour :

Généralités du système logique

- C de CC à 1 si erreur de sauvegarde du secteur de catalogue. Le registre \$20E5/\$60E5 contient le code d'erreur du système d'exploitation logique \$03 si I/O Error (aussi dans A). 0 si pas d'erreur.
- Y contient le pointeur sur la mémoire tampon de FAT.
- L'entrée du fichier a été effacée, le secteur de catalogue a été sauvé, la FAT a été mise à jour mais pas sauvée sur le disque. Pour sauver la FAT, positionner le mode « écriture d'un fichier sans écrasement » (code \$02) et appeler la routine de clôture en \$A022/\$E022.

\$A016/\$E016 : Ecriture du secteur courant

Registres d'entrée :

- \$20E9-\$20EA/\$60E9-\$60EA Pointeur sur la mémoire tampon de secteur

Registres de retour :

- Le secteur courant est sauvé sur le disque
- C de CC à 0. Si erreur, C de CC à 1, code d'erreur dans \$204E/\$604E, code contrôleur \$03 (IO Error) dans \$20E5/\$60E5, la FAT est rétablie.
- On peut remarquer que pour faciliter le travail du programmeur, le registre Y pointe sur le début du secteur de sauvegarde.

\$A019/\$E019 : Création d'un fichier

Registres d'entrées :

- La FAT doit avoir été chargée.
- \$20E9-\$20EA/\$60E9-\$60EA Pointeur sur la mémoire tampon de secteur
- \$20E7-\$20E8/\$60E7-\$60E8 Pointeur sur le descripteur de fichier (11 octets)
- \$20F0/\$60F0 Mode d'ouverture du fichier (\$02 ou \$03)
- \$20EB/\$60EB Type du fichier
- \$20EC/\$60EC Flag du fichier

Registres de retour :

- C de CC à 0. Si erreur, C de CC à 1, code d'erreur dans \$204E/\$604E, code d'erreur contrôleur dans \$20E5/\$60E5.
- \$20F6/\$60F6 Nouveau numéro de bloc
- Le secteur de catalogue à été sauvé sur le disque avec :
 - Le nom de fichier initialisé (SCRATCH.DOS si code \$03, nom du fichier courant si code \$02)
 - Type de fichier initialisé
 - Flag de fichier initialisé
 - Bloc de départ du fichier initialisé

\$A01C/\$E01C : Recherche d'un bloc libre

Registres d'entrée :

- La FAT doit avoir été chargée.
- \$20F6-\$60F6 Numéro du bloc courant

Registres de retour :

- Si FAT pleine, C de CC à 1. Le registre \$20E5/\$60E5 contient le code d'erreur du système d'exploitation logique \$05 si « Disk Full » (aussi dans A). 0 et C de CC à 0 si pas d'erreur.
- \$20F9/\$60F9 Nouveau numéro de bloc. La nouvelle place dans la FAT passe de \$FF à \$00.

\$A01F/\$E01F : Initialisation d'une opération sur un bloc

Registres d'entrée :

- \$20F6/\$60F6 Numéro du bloc courant

Registres de retour :

- \$20F5/\$60F5 Nombre de secteur du dernier bloc (initialisé à 1 pour écriture de bloc)
- \$20FA/\$60FA Numéro du premier secteur du bloc (1 ou 9)
- \$20FB/\$60FB Numéro de piste (sur 16 bits)

\$A022/\$E022 : Clôture d'une opération en écriture

Registres d'entrée :

- La FAT a été chargée (entrée \$A00D/\$E00D)
- Le fichier a été ouvert en mode sauvegarde
- \$20E9-\$20EA/\$60E9-\$60EA Pointeur sur la mémoire tampon de secteur
- \$20E7-\$20E8/\$60E7-\$60E8 Pointeur sur le descripteur de fichier de 11 caractères
- \$20F0/\$60F0 Mode d'ouverture du fichier (\$02 ou \$03)

Registres de retour :

- C de CC à 0. Si erreur, C de CC à 1 et code d'erreur dans \$204E/\$604E.
- Si le mode d'ouverture du fichier est \$02 (sauvegarde sans écrasement), les opérations suivantes sont effectuées :
 - Sauvegarde de la FAT sur le disque
 - Rétablissement de la FAT en mémoire
- Si le mode d'ouverture du fichier est \$03 (sauvegarde avec écrasement), les opérations suivantes sont effectuées :

Généralités du système logique

- Recherche dans le catalogue du fichier de nom courant
- Effacement du fichier de nom courant
- Recherche du fichier SCRATCH.DOS
- Remplacement du nom SCRATCH.DOS avec le nom de fichier courant
- Ecriture du secteur de catalogue
- Sauvegarde de la FAT sur le disque
- Rétablissement de la FAT en mémoire

ATTENTION ! Le nombre de secteurs du dernier bloc dans la FAT ainsi que la taille du dernier secteur dans le catalogue n'ont pas été mis à jour. Prévoyez d'inscrire l'un dans la FAT avant sa sauvegarde, puis recherchez le fichier en mode \$02 dans le catalogue par l'entrée \$A010/\$E010, mettez l'autre à jour et sauvegardez le secteur par l'entrée \$A016/\$E016.

\$A025/\$E025 : Fonctions standard avec numéro de secteur réel (QDD exclusivement)

Equivalente aux fonctions standard en \$A004/\$E004 mais \$204A/\$604A doit être à \$FF et le numéro de secteur de 1 à 400 dans le registre \$204A-\$204B/\$604A-\$604B.

Procédures standard du système logique

Lecture d'un fichier :

- Code \$01 dans \$20F0/\$60F0
- Chargement de la FAT
- Ouverture du fichier
- ...< chargement du fichier >...

Sauvegarde d'un fichier sans écrasement:

- Code \$02 dans \$20F0/\$60F0
- Chargement de la FAT
- Création du fichier
- Si le fichier existe déjà, erreur et sort
- ... < sauvegarde du fichier >...
- Clôture d'opération en écriture
- Ouverture du fichier en lecture
- Mise à jour de la taille du dernier secteur dans le catalogue
- Sauvegarde du secteur de catalogue

Sauvegarde d'un fichier avec écrasement :

- Code \$03 dans \$20F0/\$60F0
- Chargement de la FAT
- Création du fichier (SCRATCH.DOS si code \$03, nom de fichier courant si code \$02)
- ...< sauvegarde du fichier >...
- Clôture d'opération en écriture (Efface fichier courant et renomme « SCRATCH.DOS » avec nom du fichier courant)
- Ouverture du fichier en lecture
- Mise à jour de la taille du dernier secteur dans le catalogue
- Sauvegarde du secteur de catalogue

Effacement d'un fichier :

- Code \$02 dans \$20F0/\$60F0
- Chargement de la FAT
- Recherche du fichier dans le catalogue
- Effacement du fichier courant
- Clôture d'opération en écriture

Routine type de gestion de fichier

Le programme suivant est tiré de ColorPaint. Il n'est pas présent dans sa totalité, le but de l'exposé étant seulement de montrer comment les registres de \$60E5 à \$60FD sont utilisés pour rester compatibles dans l'usage des routines du moniteur.

Le caractère à lire ou à écrire se trouve dans le registre B

```
Programme principal
080C 34 7E          PSHS      U,Y,X,DP,B,A  |
```

Généralités du système logique

080E	86 60	LDA	#\$60	DP sur registres
0810	1F 8B	TFR	A,DP	moniteur
0812	86 55	LDA	#\$55	Si le lecteur a été
0814	91 FD	CMPA	<\$FD	initialisé, exécute
0816	27 15	BEQ	#\$82D	l'opération demandée
0818	97 FD	STA	<\$FD	
081A	86 02	LDA	#\$02	Si le contrôleur est
081C	0D 80	TST	<\$80	absent, sort avec erreur
081E	27 39	BEQ	#\$859	
0820	0F F0	CLR	<\$F0	
0822	86 01	LDA	#\$01	Sinon, initialise
0824	97 48	STA	<\$48	le contrôleur
0826	BD E8 2A	JSR	#\$E82A	
0829	86 02	LDA	#\$02	Si erreur de contrôleur,
082B	25 2C	BCS	#\$859	sort
082D	D6 EF	LDB	<\$EF	Met à jour le numéro de
082F	D7 49	STB	<\$49	lecteur
0831	9E E9	LDX	<\$E9	Met à jour le pointeur
0833	9F 4F	STX	<\$4F	sur le buffer de secteur
0835	86 01	LDA	#\$01	Code Opération Inconnue
0837	D6 E6	LDB	<\$E6	Code 0 ?
0839	10 27 01 2D	LBEQ	#\$096A	- Clôt le fichier
083D	C1 02	CMPB	#\$02	Code 2 ?
083F	10 27 00 B6	LBEQ	#\$08F9	- Ecrit sans écrasement
0843	C1 01	CMPB	#\$01	Code 1 ?
0845	27 1C	BEQ	#\$0863	- Lit un fichier
0847	C1 03	CMPB	#\$03	Code 3 ?
0849	10 27 00 8F	LBEQ	#\$08DC	- Ecrit avec écrasement
084D	C1 04	CMPB	#\$04	Code 4 ?
084F	10 27 01 48	LBEQ	#\$099B	- Affiche le répertoire
0853	C1 05	CMPB	#\$05	Code 5 ?
0855	10 27 02 2A	LBEQ	#\$0A83	- Formate le disque
0859	97 E5	STA	<\$E5	Initialise l'erreur
085B	0F FD	CLR	<\$FD	Contrôleur à réinitialiser
085D	43	COMA		Erreur dans CC
085E	7D	FCB	\$7D >TST \$0FE5	
085F	0F E5	CLR	<\$E5	Sort sans erreur
0861	35 FE	PULS	A,B,DP,X,Y,U,PC	
Lecture d'un fichier				
0863	0D F0	TST	<\$F0	Si le fichier est déjà
0865	26 0C	BNE	#\$0873	ouvert, lit les données
0867	BD 08 EE	JSR	#\$08EE	sinon ouvre le fichier
086A	25 EF	BCS	#\$085B	Si erreur d'ouverture, sort
086C	86 04	LDA	#\$04	Si le fichier n'existe
086E	5D	TSTB		pas, erreur "Fichier
086F	27 E8	BEQ	#\$0859	inexistant", sinon
0871	20 18	BRA	#\$088B	engage la lecture.
0873	9E F3	LDX	<\$F3	Tant qu'on n'est pas à la
0875	9C F1	CMPX	<\$F1	fin du buffer de secteur,
0877	26 5B	BNE	#\$08D4	lit les données du buffer
0879	96 F9	LDA	<\$F9	
087B	91 F5	CMPA	<\$F5	
087D	27 08	BEQ	#\$0887	Une fois que le dernier
087F	22 23	BHI	#\$08A4	secteur du fichier est
0881	0F E6	CLR	<\$E6	atteint, sort sans
0883	0F F0	CLR	<\$F0	erreur.
0885	20 D8	BRA	#\$085F	
0887	81 09	CMPA	#\$09	Si on a atteint le
0889	26 19	BNE	#\$08A4	dernier secteur du bloc,
088B	10 9E ED	LDY	<\$ED	convertit
088E	BD E0 1F	JSR	#\$E01F	le numéro du bloc courant
0891	D6 F6	LDB	<\$F6	
0893	5C	INCB		Enregistre le numéro du
0894	A6 A5	LDA	B,Y	bloc courant dans la FAT
0896	97 F6	STA	<\$F6	
0898	C6 09	LDB	#\$09	Programme d'office la
089A	D7 F9	STB	<\$F9	lecture de 8 secteurs,
089C	81 C0	CMPA	#\$C0	mais si le dernier bloc
089E	25 04	BCS	#\$08A4	est atteint, alors
08A0	84 0F	ANDA	#\$0F	programme le nombre de
08A2	97 F9	STA	<\$F9	secteurs du dernier bloc.
08A4	9E E9	LDX	<\$E9	Pointeur courant sur les
08A6	9F F3	STX	<\$F3	données initialisé.
08A8	CC 43 80	LDD	#\$4380	Calcule l'offset
08AB	B1 E0 02	CMPA	#\$E002	de fin de buffer

Généralités du système logique

08AE	27 02	BEQ	\$08B2	du secteur selon
08B0	CB 7F	ADDB	#\$7F	la densité.
08B2	96 F5	LDA	<\$F5	S'il s'agit du dernier
08B4	91 F9	CMPA	<\$F9	secteur du fichier, cet
08B6	25 02	BCS	\$08BA	offset est le nombre
08B8	DC F7	LDD	<\$F7	d'octets du dernier sect.
08BA	4F	CLRA		En conséquence de quoi,
08BB	30 8B	LEAX	D,X	met à jour le pointeur
08BD	9F F1	STX	<\$F1	sur la fin du buffer
08BF	96 FA	LDA	<\$FA	
08C1	0C FA	INC	<\$FA	
08C3	0C F5	INC	<\$F5	Lit le
08C5	97 4C	STA	<\$4C	secteur
08C7	DC FB	LDD	<\$FB	courant
08C9	DD 4A	STD	<\$4A	
08CB	BD E8 2A	JSR	\$E82A	
08CE	86 03	LDA	#\$03	Si erreur disque, erreur
08D0	25 87	BCS	\$0859	d'entrée-sortie
08D2	9E F3	LDX	<\$F3	
08D4	E6 80	LDB	,X+	Lit un caractère
08D6	9F F3	STX	<\$F3	du buffer de secteur
08D8	E7 61	STB	\$01,S	
08DA	20 A9	BRA	\$0885	et sort sans erreur
Ecriture avec écrasement				
08DC	0D F0	TST	<\$F0	Si fichier déjà ouvert
08DE	26 46	BNE	\$0926	écrit dans buffer et sort
08E0	8D 0C	BSR	\$08EE	Cherche fichier SCRATCH.DOS
08E2	25 86	BCS	\$086A	Si erreur disque, sort
08E4	5D	TSTB		Si le fichier n'existe
08E5	27 21	BEQ	\$0908	pas, l'ouvre en écriture,
08E7	BD E0 13	JSR	\$E013	sinon efface d'abord
08EA	25 E4	BCS	\$08D0	le fichier SCRATCH.DOS
08EC	20 1A	BRA	\$0908	avant de l'ouvrir.
08EE	D7 F0	STB	<\$F0	Mémorise opération courante
08F0	BD E0 0D	JSR	\$E00D	Charge la FAT
08F3	25 03	BCS	\$08F8	Si erreur, sort
08F5	BD E0 10	JSR	\$E010	Cherche l'entrée du fichier
08F8	39	RTS		
Ecriture sans écrasement				
08F9	0D F0	TST	<\$F0	Si le fichier est déjà
08FB	26 29	BNE	\$0926	ouvert, écrit les données
08FD	8D EF	BSR	\$08EE	Cherche l'entrée du fichier
08FF	25 E1	BCS	\$08E2	Si erreur, sort
0901	86 08	LDA	#\$08	Code "Fichier existe déjà"
0903	5D	TSTB		Si le fichier existe,
0904	10 26 FF 51	LBNE	\$0859	sort avec erreur, sinon
--				
0908	BD E0 19	JSR	\$E019	crée l'entrée du fichier
090B	97 E5	STA	<\$E5	Si erreur disque, sort
090D	25 D3	BCS	\$08E2	avec erreur
090F	9E E9	LDX	<\$E9	Initialise le pointeur
0911	9F F3	STX	<\$F3	courant sur les données
0913	30 89 00 80	LEAX	\$0080,X	
0917	C6 43	LDB	#\$43	Calcule le pointeur
0919	F1 E0 02	CMPB	\$E002	sur la fin du buffer
091C	27 03	BEQ	\$0921	de secteur selon la
091E	30 88 7F	LEAX	\$7F,X	densité
0921	9F F1	STX	<\$F1	
0923	BD E0 1F	JSR	\$E01F	Convertit le numéro de bloc
0926	9E F3	LDX	<\$F3	Tant qu'on n'est pas en fin
0928	9C F1	CMPX	<\$F1	de buffer, écrit la
092A	25 28	BCS	\$0954	donnée dans le buffer
092C	8D 31	BSR	\$095F	sinon sauve le secteur
092E	25 BA	BCS	\$08EA	Si erreur disque, sort
0930	9E E9	LDX	<\$E9	Initialise le pointeur
0932	9F F3	STX	<\$F3	courant sur les données
0934	96 FA	LDA	<\$FA	Numéro de secteur
0936	0C FA	INC	<\$FA	suivant
0938	0C F5	INC	<\$F5	Compteur de secteur + 1
093A	81 08	CMPA	#\$08	Tant que le dernier
093C	27 04	BEQ	\$0942	secteur du bloc n'est pas
093E	81 10	CMPA	#\$10	atteint, écrit dans le
0940	26 12	BNE	\$0954	buffer de secteur

Généralités du système logique

```

0942 10 9E ED      LDY      <$ED      | Recherche un bloc libre
0945 BD E0 1C      JSR      $E01C      | dans la FAT
0948 25 C3         BCS      $090D      Si erreur, sort
094A 96 F6         LDA      <$F6
094C D7 F6         STB      <$F6
094E 4C            INCA
094F E7 A6         STB      A,Y
0951 BD E0 1F      JSR      $E01F      Convertit le numéro de bloc
0954 E6 61         LDB      $01,S
0956 9E F3         LDX      <$F3
0958 E7 80         STB      ,X+
095A 9F F3         STX      <$F3
095C 7E 08 5F      JMP      $085F      Sort sans erreur
---- Sauve le secteur courant
095F 96 FA         LDA      <$FA      | Initialise le
0961 97 4C         STA      <$4C      | numéro de secteur
0963 DC FB         LDD      <$FB      | Initialise le
0965 DD 4A         STD      <$4A      | numéro de piste
0967 7E E0 16      JMP      $E016      Sauve le secteur courant

Clôt le fichier
096A 86 01         LDA      #$01      Erreur initialisée
096C D6 F0         LDB      <$F0      | Fichier
096E C5 02         BITB     #$02      | ouvert en lecture ?
0970 10 27 FE E5   LBEQ     $0859      Oui, sort avec erreur
0974 8D E9         BSR      $095F      Sauve le secteur courant
0976 25 B6         BCS      $092E      Si erreur, sort
0978 10 9E ED      LDY      <$ED
097B 96 F6         LDA      <$F6      | Place le nombre
097D D6 F5         LDB      <$F5      | de secteurs du dernier
097F C8 C0         EORB     #$C0      | bloc augmenté de $C0 à
0981 4C            INCA      | sa place dans la FAT
0982 E7 A6         STB      A,Y
0984 BD E0 10      JSR      $E010      Cherche l'entrée du fichier
0987 25 BF         BCS      $0948      Si erreur, sort
0989 DC F3         LDD      <$F3      | Mise à jour
098B 93 E9         SUBD     <$E9      | du nombre d'octets
098D ED 2E         STD      $0E,Y     | du dernier secteur
098F BD E0 16      JSR      $E016      Ecrit le secteur de FAT
0992 25 9A         BCS      $092E      Si erreur, sort
0994 BD E0 22      JSR      $E022      Clôt le fichier
0997 25 95         BCS      $092E      Si erreur, sort
0999 20 C1         BRA      $095C      Sort sans erreur

Affichage du répertoire
099B
[--]
0A80 16 FD DC      LBRA     $085F      Sort sans erreur

Formatage du disque
0A83 86 07         LDA      #$07      | Entrelacement 7
0A85 97 4D         STA      <$4D      | initialisé
0A87 86 03         LDA      #$03      Code erreur d'entrée-sortie
0A89 BD E0 0A      JSR      $E00A      Formatage disque
0A8C 10 25 FF 3F   LBCS     $09CF      Si erreur, sort
0A90 20 EE         BRA      $0A80      Sort sans erreur

```

PROGRAMMATION DES DIFFERENTS CONTROLEURS

Le THMFC1

Les registres disque

\$E7D0 Registre de commande CMD0 (écriture)

Bit 7 : 0
Bit 6 : 0
Bit 5 : 0 Format MFM
 1 Format FM
Bit 4 : Validation de détection du mot de synchronisation
Bit 3 : Désactivation du séparateur de données pour formatage
Bit 2 : 0 Lecture
 1 Ecriture
Bits 1-0 : Code pour programmation d'une opération automatique:
 00 Reset lecteur
 01 Ecriture du secteur
 10 Lecture de l'adresse
 11 Lecture du secteur

\$E7D0 Registre d'état STAT0 (lecture)

Bit 7 : Image de l'horloge caractère (DRQ)
Bit 6 : 0
Bit 5 : 0
Bit 4 : 1 Opération en phase terminale
Bit 3 : 1 Opération terminée
Bit 2 : 1 Erreur CRC
Bit 1 : 1 Bit 7 pour opération automatiques
Bit 0 : 1 Bon résultat de la détection de synchronisation

\$E7D1 Registre de commande CMD1 (écriture)

Bit 7 : Bit de compatibilité
Bits 6-5 : Longueur du secteur
 00 128 caractères par secteur
 01 256 caractères par secteur
 10 512 caractères par secteur
 11 1024 caractères par secteur
Bit 4 : Face du disque
 0 Face 1
 1 Face 0
Bits 3-2-1 : Commande de précompensation
Bit 0 : Inhibition système, lorsque signal ready inactif (bit à 1)

\$E7D1 Registre d'état STAT1 (lecture)

Bit 7 : 0
Bit 6 : 1 - Détection index floppy
 - Présence disquette QDD
Bit 5 : 1 Changement de disquette (non utilisé pour QDD)
Bit 4 : Image inversée commande moteur
Bit 3 : 1 - Détection piste 0 floppy
 - Information détection QDD
Bit 2 : 1 Protection écriture floppy QDD
Bit 1 : 1 Information ready en provenance du floppy ou du QDD
Bit 0 : Non utilisé

\$E7D2 Registre de commande CMD2 (écriture)

Bit 7 : Non utilisé
Bit 6 : 0 Face 0 du floppy
 1 Commande active du moteur QDD
 Commandes inverses
Bit 5 : 0 Commande de direction des têtes vers l'extérieur (floppy)
 1 Commande de direction des têtes vers l'intérieur (floppy)
Bit 4 : 0 Commande de pas inactivée (floppy)
 1 Commande de pas activée (floppy)
Bit 3 : Non utilisé
Bit 2 : 0 Désactivation du moteur (floppy)
 1 Activation du moteur

- Bits 1-0 : Commande de sélection de lecteur (floppy et QDD) active à l'état 1
- \$E7D3 Bus de transfert pour les données RDATA (lecture) et WDATA (écriture)
Bits 7-0 : Tampon
- \$E7D4 Extension du bus de donnée (écriture)
Bits 7-0 : \$FF pour les données
\$0A pour le mot de synchronisation
- \$E7D5 Numéro de secteur WSECT (lecture/écriture)
Bits 7-0 : Numéro de secteur pour opérations automatiques
Ce numéro de secteur sera comparé au numéro de secteur sur le disque lors de la lecture de l'identificateur de données pendant les opérations automatiques.
- \$E7D6 Numéro de piste WTRCK (lecture/écriture)
Bits 7-0 : Numéro de piste pour opérations automatiques
Ce numéro de piste sera comparé au numéro de piste sur le disque lors de la lecture de l'identificateur de données pendant les opérations automatiques.
- \$E7D7 Programmation du séparateur de données WCELL (lecture/écriture)
Bit 7 : 0 Modification caractéristiques du séparateur de données (pour les pistes intérieures de la disquette. Précompensation).
1 Fonctionnement normal du séparateur de données
Bits 6-0 : Valeur du séparateur de données selon le mode de codage utilisé
- \$3F Codage FM
- \$1F Codage MFM
Si le bit 7 de WCELL est à l'état 1, la valeur contenue dans les bits 6 à 0 agissent comme temporisateur de lecture et d'écriture. Pour la double densité, cette valeur est à \$1F, soit moitié moindre que pour la simple densité et on aura dans RDATA un caractère composé d'un bit sur deux du mot physique lu. On peut accélérer à loisir la vitesse d'absorption des données, la limite de cohérence de capture étant bien sûr relative à la rapidité du microprocesseur. La limite supérieure est en fait vite atteinte, arrivant un moment où les données sont transmises plus vite que la routine de capture n'est capable de les traiter. La lecture du mot physique réellement inscrit sur le support magnétique en MFM impliquerait que l'ordinateur tourne au moins à 2MHz pour un séparateur de données à \$0F.

Les opérations automatiques

Les opérations automatiques permettent de positionner la lecture ou l'écriture sur la zone d'identification du secteur ou sur la zone de données du secteur par un programme réduit. Les procédures de détection et de vérification de validité de ces zones s'effectuent automatiquement, à la manière du WD2793.

La lecture automatique d'un secteur

La lecture automatique d'un secteur procède de la façon suivante :

- Recherche de 3 caractères de synchronisation successifs
- Comparaison du caractère suivant avec la valeur \$FE
- Comparaison du caractère suivant avec le contenu de WTRCK
- Comparaison du caractère suivant avec l'état du bit SIDE de CMD1 (bit 4)
- Comparaison du caractère suivant avec le contenu de WSECT
- Comparaison des bits 0 et 1 du caractère suivant avec les bits LG0 et LG1 du registre CMD1
- Contrôle des 8 caractères de la zone d'identification (test de validité du CRC)
- Attente du contrôleur pendant 27 caractères
- Recherche de 1 caractère de synchro dans un délai de 42 caractères lus
- Activation de la sortie DREQ
- Le CPU répond au contrôleur en lisant le registre RDATA permettant la réactivation de DREQ et le transfert des caractères du champ de données

L'écriture automatique d'un secteur

Les 7 premières phases étant les mêmes que pour la lecture, l'écriture automatique d'un secteur procède de la façon suivante :

- Recherche de 3 caractères de synchronisation successifs.
- Comparaison du caractère suivant avec la valeur \$FE.
- Comparaison du caractère suivant avec le contenu de WTRCK.
- Comparaison du caractère suivant avec l'état du bit SIDE de CMD1 (bit 4).
- Comparaison du caractère suivant avec le contenu de WSECT.
- Comparaison des bits 0 et 1 du caractère suivant avec les bits LG0 et LG1 du registre CMD1.
- Contrôle des 8 caractères de la zone d'identification (test de validité du CRC).
- Attente du contrôleur correspondant à la lecture de 22 caractères.
- Activation de la porte d'écriture et écriture par le contrôleur de 12 caractères à \$00 suivis par un octet de synchronisation.
- Activation de la sortie DREQ.
- Le CPU répond au contrôleur en écrivant dans le registre WDATA permettant la réactivation de DREQ et le transfert des caractères du champ de données.

La lecture automatique de l'adresse

La lecture automatique de l'adresse permet, entre autre, de récupérer l'entrelacement de piste. Elle procède de la façon suivante :

- Recherche de 3 caractères de synchronisation successifs
- Comparaison du caractère suivant avec la valeur \$FE
- Activation de la sortie DREQ
- Le CPU répond au contrôleur en lisant le registre RDATA permettant la réactivation de DREQ et le transfert des caractères. Le transfert de données doit être effectué dans un délai de 25µs après l'activation de DREQ. Dans le cas contraire, le contrôleur calcule les 2 octets de contrôle CRC correspondant au N octets de donnée transférés et positionne le bit CRCER (bit 2 de STAT0).

Ci-après, la routine tirée de QBACKUP pour la recherche de l'entrelacement.

```

.....
9952 86 14          LDA    #$14          | Piste 20
9954 97 4B          STA    <$4B          | demandée
9956 1A 50          ORCC   #$50          | Gel des interruptions
9958 BD 9C B4       JSR    $9CB4          | Initialise DP
---- Efface la table des flag de présence secteur
995B CE 99 F7       LDU    #$99F7          | Ptr table des flags secteur
995E 86 0F          LDA    #$0F          | 16 octets à effacer
9960 6F C6          CLR    A,U
9962 4A             DECA
9963 2A FB          BPL    $9960          | Efface la table
---- Initialise la lecture d'adresse
9965 7F 9A 07       CLR    $9A07          | Compteur de secteur à 0
9968 C6 02          LDB   #02            | Pose commande différente
996A D7 48          STB   <$48          | de reset contrôleur
996C BD 9C BE       JSR   $9CBE          | Sélectionne le lecteur
996F 7F 98 0D       CLR   $980D          | Flag de passage à 0
9972 BD 9E 29       JSR   $9E29          | Recherche active de la piste
9975 BD 9D 65       JSR   $9D65          | Initialise la précompensation
9978 C6 1A          LDB   #$1A          | Passe en "lecture d'adresse"
997A BD 9D 7D       JSR   $9D7D          | Initialise registres disque
---- Récupère le numéro de secteur
997D 10 8E 17 70   LDY   #$1770          | Compteur de recherche
9981 86 03          LDA   #03            | 3 caractères à lire
9983 31 3F          LEAY -,$01,Y         | Si fin de décomptage,
9985 27 5A          BEQ   $99E1          | entrelacement 7
9987 E6 84          LDB   ,X             | Attend le passage
9989 C5 02          BITB #02            | de l'identificateur
998B 27 F6          BEQ   $9983          | de secteur
998D E6 03          LDB   $03,X         | Reset status
998F E6 84          LDB   ,X             | Attend fin de
9991 2A FC          BPL   $998F          | transmission
9993 E6 03          LDB   $03,X         | Récupère donnée
9995 E7 E2          STB   ,-S           | Sauve donnée en pile
9997 4A             DECA
9998 26 F5          BNE   $998F          | Pour 3
----- Vérifie la validité du CRC d'identificateur
999A E6 84          LDB   ,X             | Attend la
999C C5 08          BITB #08            | fin de
999E 27 FA          BEQ   $999A          | l'opération
99A0 C5 04          BITB #04            | Si erreur de CRC,
99A2 26 3B          BNE   $99DF          | entrelacement 7
---- Eteint le moteur
99A4 6F 84          CLR   ,X             | Reset lecteur
99A6 BD 9C CE       JSR   $9CCE          | Eteint le moteur
---- Teste si le numéro de secteur est déjà répertorié
99A9 E6 61          LDB   $01,S         | Lit n° de secteur
99AB 32 63          LEAS $03,S         | Rétablit la pile
99AD 5A             DECB
99AE C1 0F          CMPB #0F            | Si hors champ,
99B0 22 34          BHI   $99E6          | entrelacement 7
99B2 CE 99 F7       LDU   #$99F7          | Ptr table flags secteurs
99B5 63 C5          COM   B,U
99B7 27 2D          BEQ   $99E6          | Commute le flag
----- Répertorie le numéro de secteur
99B9 5C             INCB
99BA B6 9A 07       LDA   $9A07          | Réajuste numéro de secteur
99BD CE 9A 08       LDU   #$9A08          | Offset sur table entrelacmts
99C0 E7 C6          STB   A,U           | Ptr sur entrelacements
---- Secteur suivant
99C2 7C 9A 07       INC   $9A07          | Ecrit le numéro de secteur
99C5 81 0F          CMPA #0F            | Compteur de secteur + 1
99C7 25 9F          BLO   $9968          | Si pas dernier secteur,
-----
99DD 35 F7          PULS CC,A,B,X,Y,U,PC | lit adresse suivante
-----
Si erreur, force la table en entrelacement 7 soit consécutivement
la série de numéros 1 8 15 6 13 4 11 2 9 16 7 14 5 12 3 10 et... 0
99DF 32 63          LEAS $03,S         | Rétablit la pile
99E1 6F 84          CLR   ,X             | Reset lecteur
99E3 BD 9C CE       JSR   $9CCE          | Eteint le moteur
99E6 5F             CLRB
99E7 CE 9A 08       LDU   #$9A08          | N° de secteur à 0
99EA 5C             INCB
99EB E7 C0          STB   ,U+           | Ptr sur entrelacements
99ED CB 06          ADDB #06            | Ajuste n° de secteur
99EF C4 0F          ANDB #0F            | Ecrit numéro de secteur
99F1 26 F7          BNE   $99EA          | Offset pour entrelacement 7
-----
99F5 20 D2          BRA   $99C9          | Limite n° de secteur
-----
Sort du programme

```

99F7	RMB	16	Table de validité d'entrelacements
9A07	RMB	1	Compteur pour secteurs
9A08	RMB	16	Table des entrelacements
.....			

Le WD2793 (contrôleur interne du TO9)

Les registres disque

\$E7D0 Registre de commande (écriture)

Le type de la commande est surtout repérable par son premier quartet, qui est pratiquement invariable :

Commande		Bit de commande dans \$E7D0							
		7	6	5	4	3	2	1	0
Type I	Restore	0	0	0	0	h	V	r1	r0
	Seek	0	0	0	1	h	V	r1	r0
	Step	0	0	1	u	h	V	r1	r0
	Step In	0	1	0	u	h	V	r1	r0
	Step Out	0	1	1	u	h	V	r1	r0
Type II	Read Sector	1	0	0	m	S	E	C	0
	Write Sector	1	0	1	m	S	E	C	a0
Type III	Read Address	1	1	0	0	0	E	0	0
	Read Track	1	1	1	0	0	E	0	0
	Write Track	1	1	1	1	0	E	0	0
Type IV	Force Interrupt	1	1	0	1	I3	I2	I1	I0

\$E7D0 Registre d'état (lecture)

	Status des bits dans \$E7D0							
	7	6	5	4	3	2	1	0
Commandes type I	Not Ready	Write protect	Head Loaded	Seek Error	CRC Error	Track 0	Index	Busy
Read Sector	Not Ready	0	Record Type	Rec Not Found	CRC Error	Lost Data	DRQ	Busy
Write Sector	Not Ready	Write protect	Write Fault	Rec Not Found	CRC Error	Lost Data	DRQ	Busy
Read Address	Not Ready	0	0	Rec Not Found	CRC Error	Lost Data	DRQ	Busy
Read Track	Not Ready	0	0	Rec Not Found	0	Lost Data	DRQ	Busy
Write Track	Not Ready	Write protect	Write Fault	Rec Not Found	0	Lost Data	DRQ	Busy

\$E7D1 Registre de piste

Ce registre contient le numéro de piste courant

\$E7D2 Registre de secteur

Ce registre doit contenir le numéro du secteur à rechercher

\$E7D3 Bus de transfert des données

Ce registre permet le transfert des données entre le CPU et le lecteur.

\$E7D8 Numéro de lecteur/densité du secteur

Bit 7 : 1 = 128 octets par secteur

0 = 256 octets par secteur

Bits 1-2 : 01 = Contrôleur 0

10 = Contrôleur 1

Bit 0 : 0 = Face 0

1 = Face 1

Lecture ou écriture d'un secteur (READ SECTOR - WRITE SECTOR)

Les registres de piste et de secteur sont chargés au préalable avec les valeurs recherchées.

- Localisation d'une zone d'identification
- Comparaison du numéro de piste avec le numéro de piste demandée
- Comparaison du numéro de secteur avec le numéro secteur demandé
- Test de validité du CRC d'identification
- Localisation de la zone de données. Cette phase de recherche doit aboutir dans un temps inférieur à 1 seconde soit 5 révolutions maximum du disque. Passé ce délai, une interruption est générée et le bit d'état « Record Not Found » est positionné.
-

Lecture d'une adresse (READ ADDRESS)

A la réception de cette commande, la première zone d'identification rencontrée sur le disque est transférée. Cette zone est constituée des informations suivantes:

- Numéro de piste
- Numéro de face
- Numéro de secteur
- Longueur de secteur
- CRC High
- CRC Low

Lecture d'une piste (READ TRACK)

L'opération de lecture commence sur le front descendant de l'impulsion d'index et continue jusqu'à l'impulsion d'index suivante.

Le flux de données est synchronisé sur chaque mot de synchro.

Cette commande présente des caractéristiques intéressantes pour établir un diagnostic sur le fonctionnement d'un lecteur :

- Le logique de contrôle CRC est inactive, donc le transfert des données existe quelque soit la validité de celles-ci.
- Les caractères espaces entre les différentes zones d'informations sont transmis ainsi que les mots de synchro.

Ce flux ininterrompu de données, de caractères espaces et d'identificateurs permet de mettre en évidence certains dysfonctionnements dans la transmission.

Ecriture d'une piste (WRITE TRACK)

La phase d'écriture commence par le front descendant de l'impulsion d'index et continue jusqu'à l'impulsion d'index suivante. Toutes les données présentes dans le bus sont enregistrées sur le disque exceptés certains caractères que le contrôleur interprète comme des commandes :

Donnée	Interprétation en FM	Interprétation en MFM
de \$00 à \$F4	Ecrit caractère avec horloge \$FF	Ecrit caractère de \$00 à \$F4 en MFM
\$F5	Non implémenté	Ecrit \$A1 en MFM, active calcul de CRC
\$F6	Non implémenté	Ecrit \$C2 en MFM
\$F7	Génère 2 octets CRC	Génère 2 octets CRC
de \$F8 à \$FB	Ecrit caractère avec horloge \$C7, active calcul du CRC	Ecrit caractère de \$F8 à \$FB en MFM
\$FC	Ecrit \$FC avec horloge \$D7	Ecrit caractère \$FC en MFM
\$FD	Ecrit \$FD avec horloge \$FF	Ecrit caractère \$FD en MFM
\$FE	Ecrit \$FE avec horloge \$C7, active calcul du CRC	Ecrit \$FE en MFM
\$FF	Ecrit \$FF avec horloge \$FF	Ecrit \$FF en MFM

DESASSEMBLAGE DES ROMS CONTROLEURS

Contrôleur interne des TO8, TO8D et TO9+ (THMFC1) - 1ère banque

Identificateur de RomDisk (IDDISK)				E098 BD FF A0 JSR \$FFA0 Exécute en 2ème banque
E000 4D FCC "M" Contrôleur Thomson	E099 20 EF BRA \$E08C Sort du programme			
E001 54 FCC "T" FAT 160 octets				
E002 44 FCC "D" Double densité				
E003 3A FCB \$3A Checksum de l'identificateur				
Indirections standard sur les opérations physiques				Initialisation des registres
E004 16 00 6F LBRA \$E076 Fonctions standard	E09D 86 60 LDA #\$60		Fixe	
E007 16 00 1B LBRA \$E025 Lancement du boot	E09F 1F 8B TFR A,DP		le DP	
E00A 16 04 A6 LBRA \$E4B3 Formatage	E0A1 8E E7 D0 LDY #\$E7D0		Ptr registres disque	
---- Indirections standard sur les opérations logiques	E0A4 0F 4E CLR <\$4E		Efface le code d'erreur	
Note: Les indirections suivantes existent sur TO8, TO8D et TO9+ mais envoient à des adresses différentes. La première concerne le contrôleur externe du TO8 et le contrôleur interne du TO8D, la deuxième le contrôleur du TO9+ (consulter la fin du désassemblage).	E0A6 39 RTS			
E00D 17 0C A8 LBSR \$ECB8/\$EC8A Chargement de la FAT	Choix du lecteur			
E010 17 0C A5 LBSR \$ECB8/\$EC8A Ouverture d'un fichier	E0A7 DC 48 LDD <\$48		Lit commande moniteur	
E013 17 0C A2 LBSR \$ECB8/\$EC8A Effacement d'un fichier	E0A9 81 01 CMPA #\$01		Si reset demandé,	
E016 17 0C 9F LBSR \$ECB8/\$EC8A Ecriture d'un secteur	E0AB 26 02 BNE \$E0AF		choisit le	
E019 17 0C 9C LBSR \$ECB8/\$EC8A Création d'un fichier	E0AD C6 01 LDB #\$01		lecteur 1	
E01C 17 0C 99 LBSR \$ECB8/\$EC8A Allocation d'un bloc	E0AF CE E0 BE LDU #\$E0BE		Ptr table bits lecteurs	
E01F 17 0C 96 LBSR \$ECB8/\$EC8A Initialise opération sur bloc	E0B2 A6 C5 LDA B,U		Lit bits de lecteur	
E022 17 0C 93 LBSR \$ECB8/\$EC8A Clôture d'écriture	E0B4 A7 02 STA \$02,X		Init registre disque	
	E0B6 97 57 STA <\$57		Init registre système	
	E0B8 39 RTS			
	Extinction du moteur			
	E0B9 86 40 LDA #\$40		Code pour éteindre moteur	
	E0BB A7 02 STA \$02,X		Eteint le moteur	
	E0BD 39 RTS			
***** SYSTEME D'EXPLOITATION PHYSIQUE FLOPPY *****	Table des bits de lecteurs			
Lancement du boot floppy	E0BE 41 FCB \$41		Lecteur 0	
E025 8D 76 BSR \$E09D Initialise les registres	E0BF 01 FCB \$01		Lecteur 1	
E027 0F 49 CLR <\$49 Lecteur 0 demandé	E0C0 42 FCB \$42		Lecteur 2	
E029 8D 7C BSR \$E0A7 Choix du lecteur	E0C1 02 FCB \$02		Lecteur 3	
E02B 17 00 D0 LBSR \$E0FE Reset du lecteur				
E02E 86 02 LDA #\$02	Module d'exécution des fonctions standard			
E030 34 02 PSHS A	E0C2 86 02 LDA #\$02		2 tentatives	
E032 0F 58 CLR <\$58	E0C4 34 02 PSHS A		maximum	
E034 CC 00 01 LDD #\$0001	E0C6 4F CLR		Offset à 0	
E037 DD 4B STD <\$4B	E0C7 D6 48 LDB <\$48		Lit la commande moniteur	
E039 CE 62 00 LDU #\$6200	E0C9 54 LSRB			
E03C DF 4F STU <\$4F	E0CA 4C INCA			
E03E 17 03 66 LBSR \$E3A7	E0CB 24 FC BCC \$E0C9			
E041 8D 76 BSR \$E0B9	E0CD 4A DECA		Ajuste numéro de fonction	
E043 DE 4F LDU <\$4F	E0CE 48 ASLA		Calcule offset de fonction	
E045 24 08 BCC \$E04F	E0CF 10 8E E0 F0 LDY #\$E0F0		Ptr table des vecteurs	
E047 96 58 LDA <\$58	E0D3 AD B6 JSR [A,Y]		Exécute la fonction	
E049 26 20 BNE \$E06B	E0D5 24 17 BCC \$E0EE		Si pas d'erreur, sort	
E04B 03 58 COM <\$58	E0D7 96 4E LDA <\$4E		Lit code d'erreur moniteur	
E04D 20 E5 BRA \$E034	E0D9 85 51 BITA #\$51		Si erreur contrôleur/	
E04F 10 8E 62 7F LDY #\$627F	E0DB 26 11 BNE \$E0EE		lecteur/protection, sort	
E053 34 20 PSHS Y	E0DD 6A E4 DEC ,S		Si dernière tentative,	
E055 86 55 LDA #\$55	E0DF 27 0C BEQ \$E0ED		sort	
E057 6A C4 DEC ,U	E0E1 0F 4E CLR <\$4E		Efface code d'erreur	
E059 63 C4 COM ,U	E0E3 81 08 CMPA #\$08		Si erreur sur le données,	
E05B AB C0 ADDA ,U+	E0E5 27 DF BEQ \$E0C6		nouvelle tentative	
E05D 11 A3 E4 CMPU ,S	E0E7 8D 4B BSR \$E134		Recherche de la piste 0	
E060 26 F5 BNE \$E057	E0E9 25 F2 BCS \$E0DD		Si erreur, nouvel essai	
E062 35 20 PULS Y	E0EB 20 D9 BRA \$E0C6		Nouvelle tentative	
E064 A1 C4 CMPA ,U	E0ED 53 COMB		Erreur dans CC	
E066 26 08 BNE \$E070	E0EE 35 82 PULS A,PC			
E068 7E 62 00 JMP \$6200	Table de vecteurs des fonctions standard			
---- Sortie si erreur de boot	E0F0 E0 FE FDB \$E0FE		01 Reset du lecteur	
E06B 35 02 PULS A	E0F2 E3 A7 FDB \$E3A7		02 Lecture d'un secteur	
E06D 4A DECA	E0F4 E1 6E FDB \$E16E		04 Passage en simple densité	
E06E 26 C0 BNE \$E030	E0F6 E1 77 FDB \$E177		08 Ecriture d'un secteur	
E070 0F 80 CLR <\$80	E0F8 E1 74 FDB \$E174		10 Passage en double densité	
E072 6E 9F 00 1E JMP [\$001E]	E0FA E1 34 FDB \$E134		20 Recherche de la piste 0	
	E0FC E4 52 FDB \$E452		40 Recherche d'une piste	
Traitement des fonctions standard	Routine de reset du lecteur			
E076 34 7F PSHS U,Y,X,DP,B,A,CC	E0FE 8D 34 BSR \$E134		Recherche de la piste 0	
E078 8D 23 BSR \$E09D	E100 25 31 BCS \$E133		Si erreur, sort	
E07A 96 48 LDA <\$48	---- Définition de la densité système			
E07C 81 01 CMPA #\$01	E102 86 01 LDA #\$01		Piste 1	
E07E 27 06 BEQ \$E086	E104 97 4B STA <\$4B		demandée	
E080 96 49 LDA <\$49	E106 17 03 71 LBSR \$E47A		Déplace les têtes	
E082 81 04 CMPA #\$04	E109 C6 44 LDB #'D		Code double densité	
E084 27 0D BEQ \$E093	E10B A6 01 LDA \$01,X		Lit status lecteur	
E086 8D 1F BSR \$E0A7	E10D 85 08 BITA #\$08		Si pas de détection de	
E088 8D 38 BSR \$E0C2	E10F 27 02 BEQ \$E113		piste 0, double densité	
E08A 8D 2D BSR \$E0B9	E111 C6 43 LDB #'C		Code simple densité	
E08C 35 02 PULS A	E113 D7 4E STB <\$4E		Fixe le code densité	
E08E 1E 8A EXG A,CC	---- Retour des têtes en piste 0			
E090 44 LSR A	E115 8D 1D BSR \$E134		Recherche de la piste 0	
E091 35 FE PULS A,B,DP,X,Y,U,PC	E117 0F 52 CLR <\$52		Toute tête	
Traitement des fonctions standard pour le RamDisk	E119 0F 54 CLR <\$54		en piste 0	
E093 CE E0 04 LDU #\$E004	E11B 6F 84 CLR ,X		Eteint le moteur	
E096 DF 55 STU <\$55	E11D 0F 58 CLR <\$58		Initialise en double densité	
	---- Définition de la densité du lecteur			

Contrôle interne des TO8, TO8D et TO9+ (THMFC1) - 1ère banque

<p>E11F DC E9 LDD <\$E9 E121 93 ED SUBD <\$ED Récupère la valeur E123 2C 05 BGE \$E12A absolue de la différence E125 43 COMA entre le pointeur de FAT E126 53 COMB et le pointeur de buffer E127 C3 00 01 ADDD #\$0001 E12A 4D TSTA Si différence > 256, E12B 26 05 BNE \$E132 double densité E12D 5D TSTB Si différence < 128, E12E 2A 02 BPL \$E132 double densité E130 03 58 COM <\$58 Sinon simple densité E132 4F CLRA Pas d'erreur dans CC E133 39 RTS</p> <p style="text-align: center;">Recherche de la piste 0</p> <p>E134 C6 64 LDB #\$64 100 tentatives maximum E136 A6 01 LDA \$01,X lit status E138 85 08 BITA #\$08 Si piste 0 trouvée, E13A 26 13 BNE \$E14F sort E13C 96 57 LDA <\$57 Lit bits lecteur E13E 8A 10 ORA #\$10 E140 A7 02 STA \$02,X Déplace E142 84 EF ANDA #\$EF la tête E144 A7 02 STA \$02,X E146 8D 1D BSR \$E165 Temporisation E148 5A DECB Si dernière tentative, E149 10 27 03 27 LBQ \$E474 erreur "Lecteur non prêt" E14D 20 E7 BRA \$E136 Nouveau déplacement E14F 8D 0E BSR \$E15F Temporisation E151 17 03 54 LBSR \$E4A8 Ptr position de la tête E154 6F A4 CLR Y Tête en piste 0 E156 6F 06 CLR \$06,X initialise registre piste E158 39 RTS</p> <p style="text-align: center;">Temporisation de 864µs</p> <p>E159 10 8E 00 6C LDY #\$006C Valeur de temporisation E15D 20 0A BRA \$E169 Effectue la boucle ---- Temporisation de 14976µs E15F 10 8E 07 50 LDY #\$0750 Valeur de temporisation E163 20 04 BRA \$E169 Effectue la boucle ---- Temporisation de 5984µs E165 10 8E 02 EC LDY #\$02EC Valeur de temporisation E169 31 3F LEAY -\$01,Y Boucle pour E16B 26 FC BNE \$E169 temporisation E16D 39 RTS</p> <p style="text-align: center;">Routine de passage en simple densité</p> <p>E16E 86 FF LDA #\$FF Fixe flag de simple E170 97 58 STA <\$58 densité E172 4F CLRA Pas d'erreur dans CC E173 39 RTS</p> <p style="text-align: center;">Routine de passage en double densité</p> <p>E174 0F 58 CLR <\$58 Fixe flag de double densité E176 39 RTS</p> <p style="text-align: center;">Routine d'écriture d'un secteur</p> <p>E177 17 02 D8 LBSR \$E452 Recherche la piste E17A 25 FA BCS \$E176 Si erreur, sort E17C A6 01 LDA \$01,X lit le status disque E17E 85 04 BITA #\$04 Si bit de protection E180 27 05 BEQ \$E187 actif, E182 86 01 LDA #\$01 initialise code "Disquette E184 16 02 EF LBRA \$E476 protégée" et sort E187 96 58 LDA <\$58 Si simple densité E189 10 26 00 93 LBNE \$E220 demandée, traite ---- Ecriture d'un secteur double densité E18D 17 01 B1 LBSR \$E341 Fixe la précompensation E190 C6 19 LDB #\$19 Ecriture automatique E192 34 01 PSHS CC Protège le CC E194 17 01 FD LBSR \$E394 Initialise registres disque ---- Recherche du secteur demandé E197 10 8E 75 00 LDY #\$7500 Compteur pour une piste E19B 31 3F LEAY -\$01,Y Si fin du décomptage, E19D 10 27 02 9F LBQ \$E440 erreur de secteur E1A1 E6 84 LDB ,X lit status opération E1A3 C5 02 BITB #\$02 Si secteur pas trouvé, E1A5 27 F4 BEQ \$E19B boucle E1A7 A7 03 STA \$03,X Réinitialise status ---- Ecriture du secteur double densité E1A9 E6 84 LDB ,X Attend fin de E1AB 2A FC BPL \$E1A9 transmission E1AD A7 03 STA \$03,X Ecrit un mot de synchro E1AF E6 84 LDB ,X Attend la fin de E1B1 2A FC BPL \$E1AF la transmission E1B3 C6 FB LDB #\$FB Ecrit le marquage E1B5 E7 03 STB \$03,X de données E1B7 4F CLRA 256 caractères à écrire E1B8 E6 84 LDB ,X Attend la fin de E1BA 2A FC BPL \$E1B8 la transmission E1BC E6 C0 LDB ,U+ lit un caractère E1BE E7 03 STB \$03,X Enregistre le caractère E1C0 4A DECA Pour les 256 E1C1 26 F5 BNE \$E1B8 caractères E1C3 E6 84 LDB ,X lit status opération E1C5 C5 08 BITB #\$08 Attend la fin</p>	<p>E1C7 27 FA BEQ \$E1C3 de l'opération E1C9 8D 8E BSR \$E159 Temporisation E1CB 0D 48 TST <\$48 Si pas de vérification, E1CD 2A 48 BPL \$E217 sort E1CF 17 02 88 LBSR \$E45A Activation du lecteur E1D2 C6 1B LDB #\$1B Lecture automatique E1D4 17 01 BD LBSR \$E394 Initialise registres disque ---- Recherche du secteur demandé E1D7 86 04 LDA \$04 Compteur recherche marquage E1D9 10 8E 75 00 LDY #\$7500 Compteur pour une piste E1DD 31 3F LEAY -\$01,Y Si fin du décomptage, E1DF 10 27 00 F9 LBQ \$E2DC erreur de vérification E1E3 E6 84 LDB ,X lit status opération E1E5 C5 02 BITB #\$02 Si secteur pas trouvé, E1E7 27 F4 BEQ \$E1DD boucle E1E9 E6 03 LDB \$03,X Réinitialise status disque ---- Recherche du marquage de données E1EB 4A DECA Si fin de décomptage, E1EC 10 27 02 58 LBQ \$E448 erreur de piste E1F0 E6 84 LDB ,X Attend fin de E1F2 2A FC BPL \$E1F0 transmission E1F4 E6 03 LDB \$03,X lit le caractère E1F6 C1 FB CMPB #\$FB Si pas marquage de données, E1F8 26 F1 BNE \$E1EB nouveau test ---- Vérification du secteur double densité E1FA 4F CLRA 256 caractères à vérifier E1FB E6 84 LDB ,X Attend fin de E1FD 2A FC BPL \$E1FB transmission E1FF E6 03 LDB \$03,X Charge le caractère E201 E1 C0 CMPB ,U+ Si caractère incorrect, E203 26 03 BNE \$E208 exit E205 4A DECA Pour les 256 E206 26 F3 BNE \$E1FB caractères ---- Test de la validité de la vérification E208 E6 84 LDB ,X lit status opération E20A C5 08 BITB #\$08 Si pas fin d'opération, E20C 27 FA BEQ \$E208 boucle E20E 4D TSTA Si pas secteur entier E20F 26 02 BNE \$E213 vérifié, erreur E211 C5 04 BITB #\$04 Si CRC de données E213 10 26 00 C5 LBNE \$E2DC incorrect, erreur E217 35 01 PULS CC Rétablit CC E219 6F 84 CLR ,X Reset lecteur E21B 39 RTS</p> <p style="text-align: center;">Vérification d'un secteur double densité pour formatage</p> <p>E21C 34 01 PSHS CC Rétablit CC E21E 20 AF BRA \$E1CF Vérifie secteur</p> <p style="text-align: center;">Ecriture d'un secteur simple densité</p> <p>E220 17 01 31 LBSR \$E354 Fixe la précompensation E223 17 01 35 LBSR \$E35B Calcul du CRC pour le secteur E226 34 01 PSHS CC Préserve CC E228 34 06 PSHS B,A Empile le CRC ---- Recherche le secteur E22A 17 00 BD LBSR \$E2EA Recherche le secteur E22D 24 05 BCC \$E234 Si ok, passe E22F 35 06 PULS A,B Rétablit la pile E231 16 02 0C LBRA \$E440 Erreur de secteur ---- Ecriture de 6 blancs E234 C6 FF LDB #\$FF Passage en mode E236 E7 04 STB \$04,X "données" E238 CC 06 24 LDD #\$0624 6 caractères à écrire E23B E7 84 STB ,X Passe en écriture données FM E23D E6 84 LDB ,X Attend fin de E23F 2A FC BPL \$E23D transmission E241 6F 03 CLR \$03,X Ecrit un caractère à 0 E243 4A DECA Pour les 6 E244 26 F7 BNE \$E23D caractères ---- Ecriture du marquage de données E246 E6 84 LDB ,X Attend fin de E248 2A FC BPL \$E246 transmission E24A CC FB C7 LDD #\$FBC7 Passe en mode "synchro" E24D ED 03 STD \$03,X et écrit marquage ---- Ecriture du secteur simple densité E24F C6 FF LDB #\$FF Passe en mode "données" E251 A6 84 LDA ,X Attend fin de E253 2A FC BPL \$E251 transmission E255 A6 C0 LDA ,U+ Lit premier caractère E257 ED 03 STD \$03,X Enregistre premier caractère E259 86 7F LDA \$03,X 127 caractères restants E25B E6 84 LDB ,X Attend fin de E25D 2A FC BPL \$E25B transmission E25F E6 C0 LDB ,U+ lit le caractère E261 E7 03 STB \$03,X Sauve le caractère E263 4A DECA Pour tout E264 26 F5 BNE \$E25B le secteur ---- Ecriture du CRC de données E266 E6 84 LDB ,X Attend fin de E268 2A FC BPL \$E266 transmission E26A 35 04 PULS B Récupère CRCHigh/Low E26C E7 03 STB \$03,X Sauve CRCHigh/Low E26E 43 COMA Si pas deuxième passe, E26F 26 F5 BNE \$E266 boucle ---- Clôture du secteur simple densité E271 E6 84 LDB ,X Attend fin de</p>
---	--

Contrôleur interne des TO8, TO8D et TO9+ (THMFC1) - 1ère banque

E64F 86 0C LDA #50C | 12 mots à écrire
 E651 E6 84 LDB ,X | Attend fin de
 E653 2A FC BPL \$E651 | transmission
 E655 6F 03 CLR \$03,X | Ecrit un mot à 0
 E657 4A DECA | Pour les 12
 E658 26 F7 BNE \$E651 | blancs
 ---- Ecriture des trois mots de synchronisation
 E65A E6 84 LDB ,X | Attend fin de
 E65C 2A FC BPL \$E65A | transmission
 E65E CC A1 0A LDD #A10A | Passe en mode "synchro"
 E661 ED 03 STD \$03,X | et écrit mot de synchro
 E663 E6 84 LDB ,X | Attend fin de
 E665 2A FC BPL \$E663 | transmission
 E667 A7 03 STA \$03,X | Ecrit mot de synchronisation
 E669 E6 84 LDB ,X | Attend fin de
 E66B 2A FC BPL \$E669 | transmission
 E66D A7 03 STA \$03,X | Ecrit mot de synchronisation
 ---- Ecriture du marquage de données
 E66F E6 84 LDB ,X | Attend fin de
 E671 2A FC BPL \$E66F | transmission
 E673 CC FB FF LDD #FBFF | Passe en mode "donnée"
 E676 ED 03 STD \$03,X | et écrit le marquage
 ---- Ecriture du secteur double densité
 E678 4F CLRA | 256 mots à écrire
 E679 E6 84 LDB ,X | Attend fin de
 E67B 2A FC BPL \$E679 | transmission
 E67D C6 E5 LDB #E5 | Ecrit un
 E67F E7 03 STB \$03,X | mot à \$E5
 E681 4A DECA | Pour les 256
 E682 26 F5 BNE \$E679 | mots
 ---- Ecriture du CRC de données
 E684 E6 84 LDB ,X | Attend fin de
 E686 2A FC BPL \$E684 | transmission
 E688 C6 78 LDB #78 | Ecrit le
 E68A E7 03 STB \$03,X | CRCHigh
 E68C E6 84 LDB ,X | Attend fin de
 E68E 2A FC BPL \$E68C | transmission
 E690 C6 27 LDB #27 | Ecrit le
 E692 E7 03 STB \$03,X | CRCLow
 ---- Ecriture de la plage de battement
 E694 86 2C LDA #2C | 12 mots à écrire
 E696 E6 84 LDB ,X | Attend fin de
 E698 2A FC BPL \$E696 | transmission
 E69A C6 4E LDB #4E | Ecrit un mot
 E69C E7 03 STB \$03,X | à \$4E
 E69E 4A DECA | Pour les 12
 E69F 26 F5 BNE \$E696 | mots
 ---- Secteur suivant
 E6A1 6A E4 DEC ,S | Secteur
 E6A3 10 26 FF 65 LBNE \$E60C | suivant
 E6A7 16 00 A1 LBRA \$E74B | Clôt la piste et vérifie

Formatage d'une piste en simple densité
 E6AA 17 FC A7 LBSR \$E354 | Fixe la précompensation
 E6AD 17 FF 24 LBSR \$E5D4 | Attend le passage d'index
 E6B0 CC FF FF LDD #FFFFFF | Passe en mode
 E6B3 ED 03 STD \$03,X | "données"
 E6B5 C6 24 LDB #24 | Passe en
 E6B7 E7 84 STB ,X | formatage
 ---- Ecriture de l'amorce de piste
 E6B9 86 10 LDA #10 | 16 mots à écrire
 E6BB E6 84 LDB ,X | Attend fin de
 E6BD 2A FC BPL \$E6BB | transmission
 E6BF C6 FF LDB #FFF | Ecrit un mot
 E6C1 E7 03 STB \$03,X | à \$FF
 E6C3 4A DECA | Pour les 16
 E6C4 26 F5 BNE \$E6BB | mots
 ---- Ecriture de la plage de 6 blancs
 E6C6 86 06 LDA #06 | 6 mots à écrire
 E6C8 E6 84 LDB ,X | Attend fin de
 E6CA 2A FC BPL \$E6C8 | transmission
 E6CC 6F 03 CLR \$03,X | Ecrit un mot à 0
 E6CE 4A DECA | Pour les 6
 E6CF 26 F7 BNE \$E6C8 | mots
 ---- Ecriture du marquage d'identification
 E6D1 E6 84 LDB ,X | Attend fin de
 E6D3 2A FC BPL \$E6D1 | transmission
 E6D5 CC FE C7 LDD #FEC7 | Ecrit le marquage
 E6D8 ED 03 STD \$03,X | en mode "synchro"
 ---- Ecriture de l'identificateur de secteur
 E6DA E6 84 LDB ,X | Attend fin de
 E6DC 2A FC BPL \$E6DA | transmission
 E6DE C6 FF LDB #FF | Lit premier paramètre
 E6E0 A6 C0 LDA ,U+ | et passe en mode "donnée"
 E6E2 ED 03 STD \$03,X | Sauve le premier paramètre
 E6E4 86 05 LDA #05 | 5 paramètres restants
 E6E6 E6 84 LDB ,X | Attend fin de
 E6E8 2A FC BPL \$E6E6 | transmission
 E6EA E6 C0 LDB ,U+ | Lit le paramètre
 E6EC E7 03 STB \$03,X | Sauve le paramètre
 E6EE 4A DECA | Pour tous les
 E6EF 26 F5 BNE \$E6E6 | paramètres
 ---- Ecriture de la plage de battement
 E6F1 86 0C LDA #0C | 12 mots à écrire
 E6F3 E6 84 LDB ,X | Attend fin de

E6F5 2A FC BPL \$E6F3 | transmission
 E6F7 C6 FF LDB #FF | Ecrit un mot
 E6F9 E7 03 STB \$03,X | à \$FF
 E6FB 4A DECA | Pour les 12
 E6FC 26 F5 BNE \$E6F3 | mots
 ---- Ecriture de 6 blancs
 E6FE 86 06 LDA #06 | 6 mots à écrire
 E700 E6 84 LDB ,X | Attend fin de
 E702 2A FC BPL \$E700 | transmission
 E704 6F 03 CLR \$03,X | Ecrit un mot à 0
 E706 4A DECA | Pour les 6
 E707 26 F7 BNE \$E700 | mots
 ---- Ecriture du marquage de données
 E709 E6 84 LDB ,X | Attend fin de
 E70B 2A FC BPL \$E709 | transmission
 E70D CC FB C7 LDD #FBC7 | Ecrit le marquage
 E710 ED 03 STD \$03,X | en mode "synchro"
 ---- Ecriture du secteur
 E712 E6 84 LDB ,X | Attend fin de
 E714 2A FC BPL \$E712 | transmission
 E716 CC E5 FF LDD #E5FF | Ecrit le premier caractère
 E719 ED 03 STD \$03,X | en mode "donnée"
 E71B 86 7F LDA #7F | 127 caractères restants
 E71D E6 84 LDB ,X | Attend fin de
 E71F 2A FC BPL \$E71D | transmission
 E721 C6 E5 LDB #E5 | Ecrit un caractère
 E723 E7 03 STB \$03,X | à \$E5
 E725 4A DECA | Pour tout
 E726 26 F5 BNE \$E71D | le secteur
 ---- Ecriture du CRC de données
 E728 E6 84 LDB ,X | Attend fin de
 E72A 2A FC BPL \$E728 | transmission
 E72C C6 5D LDB #5D | Ecrit le
 E72E E7 03 STB \$03,X | CRCHigh
 E730 E6 84 LDB ,X | Attend fin de
 E732 2A FC BPL \$E730 | transmission
 E734 C6 30 LDB #30 | Ecrit le
 E736 E7 03 STB \$03,X | CRCLow
 ---- Ecriture de la plage de battement
 E738 86 16 LDA #16 | 22 mots à écrire
 E73A E6 84 LDB ,X | Attend fin de
 E73C 2A FC BPL \$E73A | transmission
 E73E C6 FF LDB #FF | Ecrit un mot
 E740 E7 03 STB \$03,X | à \$FF
 E742 4A DECA | Pour les 22
 E743 26 F5 BNE \$E73A | mots
 ---- Secteur suivant
 E745 6A E4 DEC ,S | Secteur
 E747 10 26 FF 7B LBNE \$E6C6 | suivant
 ---- Ecriture jusqu'à l'index
 E74B A6 01 LDA \$01,X | Lit status disque
 E74D 85 40 BITA #40 | Si index atteint,
 E74F 26 08 BNE \$E759 | exit
 E751 A6 84 LDA ,X | Attend fin de
 E753 2A FC BPL \$E751 | transmission
 E755 E7 03 STB \$03,X | Ecrit mot de remplissage
 E757 20 F2 BRA \$E74B | Mot suivant
 ---- Test de demande de vérification
 E759 35 02 PULS A | Rétablit la pile
 E75B 6F 84 CLR ,X | Reset du lecteur
 E75D 0D 48 TST <\$48 | Si pas de vérification,
 E75F 2A 22 BPL \$E783 | sort du programme
 ---- Vérification de la piste formatée
 E761 86 E5 LDA #E5 | Valeur pour remplissage
 E763 8D 1F BSR \$E784 | Remplit le secteur
 E765 CE 60 D3 LDU #60D3 | Ptr sur table entrelacements
 E768 17 FC EF LBSR \$E45A | Activation du lecteur
 E76B A6 C0 LDA ,U+ | Lit numéro de secteur
 E76D 27 14 BEQ \$E783 | Si fin de table, sort
 E76F 97 4C STA <\$4C | Initialise numéro de secteur
 E771 34 40 PSHS U |
 E773 96 58 LDA <\$58 | Si double densité,
 E775 27 05 BEQ \$E77C | vérifie double densité
 E777 17 FB 67 LBSR \$E2E1 | Vérifie simple densité
 E77A 20 03 BRA \$E77F | Secteur suivant
 E77C 17 FA 9D LBSR \$E21C | Vérifie double densité
 E77F 35 40 PULS U |
 E781 24 E8 BCC \$E76B | Si ok, secteur suivant
 E783 39 RTS |
 Remplissage du secteur
 E784 DE 4F LDU <\$4F | Ptr sur buffer secteur
 E786 D6 58 LDB <\$58 | Lit la densité lecteur
 E788 C4 80 ANDB #80 | Calcule taille du secteur
 E78A A7 C0 STA ,U+ | Remplit le
 E78C 5A DECB | secteur avec
 E78D 26 FB BNE \$E78A | la valeur convenue
 E78F 39 RTS |
 Initialisation de la piste de catalogue
 E790 86 14 LDA #14 | Piste 20
 E792 97 4B STA <\$4B | demandée
 E794 17 FC E3 LBSR \$E47A | Déplace les têtes
 ---- Initialise les 16 secteurs à \$FF
 E797 86 FF LDA #FF | Valeur de remplissage
 E799 8D E9 BSR \$E784 | Remplit buffer de secteur

Contrôleur interne des TO8, TO8D et TO9+ (THMFC1) - 1ère banque

E79B	86 10	LDA	#\$10		16 secteurs
E79D	97 4C	STA	<\$4C		à écrire
E79F	17 FC B8	LBSR	\$E45A		Activation du lecteur
E7A2	17 F9 E2	LBSR	\$E187		Ecrit le secteur
E7A5	25 18	BCS	\$E7BF		Si erreur, sort
E7A7	0A 4C	DEC	<\$4C		Secteur
E7A9	26 F4	BNE	\$E79F		suivant
---- Initialisation de la FAT					
E7AB	DE 4F	LDU	<\$4F		Ptr sur buffer de secteur
E7AD	6F C4	CLR	,U		Premier octet de FAT à 0
E7AF	CC FE FE	LDD	#\$FEFE		Réserve piste
E7B2	ED C8 29	STD	\$29,U		de catalogue
E7B5	86 02	LDA	#\$02		Secteur 2
E7B7	97 4C	STA	<\$4C		demandé
E7B9	17 F9 CB	LBSR	\$E187		Ecrit le secteur
E7BC	25 01	BCS	\$E7BF		Si erreur, sort
E7BE	4F	CLRA			Pas d'erreur dans CC
E7BF	39	RTS			

Les programmes suivants sont cités à titre d'information.
Ils ne font pas partie de la ROM disque.

Note: Le programme suivant existe sur TO8, TO8D et TO9+ dans les 2
banques mais à des adresses différentes selon la machine. La
première concerne les TO8 et TO8D, la deuxième le TO9+.

Exécution d'une table d'indirection dans l'autre page ROM du moniteur					
ECB8/EC8A	34 40	PSHS	U		
ECBA/EC8C	EE 62	LDU	\$02,S		Récupère vecteur d'exécution
ECBC/EC8E	33 5D	LEAU	-\$03,U		Réajuste le vecteur
ECBE/EC90	FF 60 55	STU	\$6055		Sauve vecteur d'exécution
ECC1/EC93	35 40	PULS	U		
ECC3/EC95	32 62	LEAS	\$02,S		Expulse le PC
ECC5/EC97	7E FF A0	JMP	\$FFA0		Exécute dans l'autre page ROM

.....

Note: Le programme suivant existe sur TO8, TO8D et TO9+. Mêmes adresses
pour les trois dans les 2 banques.

Exécution d'un programme dans l'autre page ROM du moniteur					
FFA0	34 03	PSHS	A,CC		Protège le CC
FFA2	1A 50	ORCC	#\$50		Gel des interruptions
FFA4	B6 E7 C3	LDA	\$E7C3		
FFA7	8A 10	ORA	#\$10		Passe dans l'autre page ROM
FFA9	B7 E7 C3	STA	\$E7C3		
FFAC	35 03	PULS	CC,A		Rétablit les registres
FFAE	AD 9F 60 55	JSR	[\$6055]		Exécute le programme
FFB2	34 03	PSHS	A,CC		Protège le CC
FFB4	1A 50	ORCC	#\$50		Gel des interruptions
FFB6	B6 E7 C3	LDA	\$E7C3		
FFB9	8A 10	ORA	#\$10		Revient en ROM initiale
FFBB	B7 E7 C3	STA	\$E7C3		
FFBE	35 83	PULS	CC,A,PC		

Contrôleur interne des TO8 et TO8D (THMFC1) - 2ème banque

```

Identificateur de RomDisk (IDDISK)
E000 4D FCC "M"      Contrôleur Thomson
E001 4B FCC "K"      Fat 80 octets
E002 43 FCC "C"      Simple densité
E003 30 FCC "$30"    Checksum de l'identificateur
  
```

```

Indirections standard sur les opérations physiques
E004 16 00 6F LBRA $E076 Fonctions standard QDD/RAM
E007 16 00 1E LBRA $E028 Lancement du boot QDD
E00A 16 03 3B LBRA $E348 Formatage QDD/RAM
---- Indirections standard sur les opérations logiques
E00D 16 05 1D LBRA $E52D Chargement de la FAT
E010 16 05 9F LBRA $E5B2 Ouverture d'un fichier
E013 16 05 7E LBRA $E594 Effacement d'un fichier
E016 16 05 5A LBRA $E573 Ecriture d'un secteur
E019 16 05 FE LBRA $E61A Création d'un fichier
E01C 16 06 54 LBRA $E673 Allocation d'un bloc
E01F 16 06 9E LBRA $E6B8 Initialise opération sur bloc
E022 16 04 B1 LBRA $E4D6 Clôture d'écriture
---- Indirections standard sur les opérations physiques
E025 16 00 45 LBRA $E06D Fonctions standard QDD/RAM
  
```

***** SYSTEME D'EXPLOITATION PHYSIQUE DU QDD *****

```

Lancement du boot QDD
E028 17 00 90 LBSR $E0BB Initialisation registres
E02B 0F 49 CLR <$49 Lecteur 0 demandé
E02D 17 00 A1 LBSR $E0D1 Choix du lecteur
E030 25 35 BCS $E067 Si erreur, sort
E032 17 00 D2 LBSR $E107 Initialise en simple densité
E035 25 30 BCS $E067 Si erreur, sort.
E037 CC 00 44 LDD #$0044 Secteur 68 (Boot QDD)
E03A DD 4C STD <$4C sélectionné
E03C CE 62 00 LDU #$6200 Ptr buffer secteur
E03F DF 4F STU <$4F sélectionné
E041 17 01 F8 LBSR $E23C lit le secteur QDD
E044 DE 4F LDU <$4F Ptr buffer secteur
E046 25 1F BCS $E067 Si erreur, sort
E048 17 00 A2 LBSR $E0ED Eteint le lecteur
E04B 10 8E 62 7F LDY #$627F Limite du décodage
E04F 34 20 PSDS Y en pile
E051 86 55 LDA #$55 Checksum de départ
E053 6A C4 DEC ,U Décode les
E055 63 C4 COM ,U données
E057 AB C0 ADDA ,U+ Calcule le checksum
E059 11 A3 E4 CMPL ,S Si pas fin du décodage,
E05C 26 F5 BNE $E053 boucle
E05E 35 20 PULS Y Rétablit la pile
E060 A1 C4 CMLA ,U Si checksum incorrect,
E062 26 03 BNE $E067 lance l'application
E064 7E 62 00 JMP $6200 Sinon, exécute le boot
---- Sortie si erreur de boot
E067 0F 80 CLR <$80 Flag "contrôleur absent"
E069 6E 9F 00 1E JMP [$001E] Lance l'application à froid
  
```

```

Opérations standard avec numéro de secteur QDD réel
E06D 34 7F PSHS U,Y,X,DP,B,A,CC
E06F 86 FF LDA #$FF Flag pour opérations
E071 B7 60 4A STA $604A QDD avec secteur réel
E074 20 05 BRA $E07B Lance l'opération
  
```

```

Opérations standard avec numéro de secteur QDD interprété
E076 34 7F PSHS U,Y,X,DP,B,A,CC
E078 7F 60 4A CLR $604A Flag de secteur interprété
E07B 8D 3E BSR $E0BB Initialise registres DP et X
---- Traite le reset du contrôleur
E07D 96 48 LDA <$48 Lit la commande
E07F 81 01 CMLA #$01 Si reset contrôleur
E081 27 21 BEQ $E0A4 demandé, traite
E083 96 49 LDA <$49 Lit le numéro de lecteur
E085 81 04 CMLA #$04 Si pas lecteur RamDisk,
E087 26 1B BNE $E0A4 opérations QDD
---- Traite la lecture du secteur RamDisk
E089 96 48 LDA <$48 Lit la commande
E08B 85 02 BITA #$02 Si pas lecture secteur
E08D 27 05 BEQ $E094 passe
E08F BD E7 03 JSR $E703 Lit le secteur RamDisk
E092 20 20 BRA $E0B4 Sort du programme
---- Traite l'écriture du secteur RamDisk
E094 85 08 BITA #$08 Si pas écriture secteur,
E096 27 05 BEQ $E09D sort avec erreur
E098 BD E6 C8 JSR $E6C8 Ecrit le secteur RamDisk
E09B 20 17 BRA $E0B4 Sort du programme
---- Erreur si contrôleur non prêt
E11F 17 00 DD LBSR $E1FF Teste disponibilité lecteur
E122 25 4C BCS $E170 Si erreur, sort
E124 34 01 PSHS CC Préserve les interruptions
E126 8D 59 BSR $E181 Recherche du secteur demandé
E128 25 47 BCS $E171 Si erreur, sort
E12A C6 FF LDB #$FF Initialise en
E12C E7 04 STB $04,X mode données
  
```

```

E09D 86 40 LDA #$40 "Contrôleur non prêt"
E09F 97 4E STA <$4E dans registre d'erreur
E0A1 43 COMA Erreur dans CC
E0A2 20 10 BRA $E0B4 Sort avec erreur
---- Exécution de l'opération
E0A4 DC 4B LDD <$4B Protection de
E0A6 34 06 PSHS B,A DKTRK et DKSEC
E0A8 8D 27 BSR $E0D1 Test disponibilité lecteur
E0AA 25 02 BCS $E0AE Si erreur, sort
E0AC 8D 48 BSR $E0FE Exécute l'opération
---- Sortie du programme
E0AE 35 06 PULS A,B Restaure DKTRK
E0B0 DD 4B STD <$4B et DKSEC
E0B2 8D 39 BSR $E0ED Eteint le lecteur
E0B4 35 02 PULS A Dépile le CC
E0B6 1E 8A EXG A,CC Restaure le CC
E0B8 44 LSRA Bit d'erreur dans CC
E0B9 35 FE PULS A,B,DP,X,Y,U,PC
  
```

```

Initialisation des registres disque
E0BB 86 60 LDA #$60 Initialise
E0BD 1F 8B TFR A,DP le DP
E0BF 8E E7 D0 LDX #$E7D0 Ptr sur registres disques
E0C2 0F 4E CLR <$4E Efface code d'erreur
E0C4 6F 84 CLR ,X Reset lecteur
E0C6 86 CE LDA #$CE Séparateur de donnée
E0C8 A7 07 STA 7,X avec compteur de $4E
E0CA 86 FF LDA #$FF Programme
E0CC A7 04 STA 4,X le mode "donnée"
E0CE 6F 01 CLR 1,X Pas de précompensation
E0D0 39 RTS
  
```

```

Sélection des bits de lecteur QDD
E0D1 D6 49 LDB <$49 Si face impaire
E0D3 C5 01 BITB #$01 demandée, erreur
E0D5 26 11 BNE $E0E8 "lecteur non prêt"
E0D7 CE E0 F2 LDU #$E0F2 Table des bits de lecteurs
E0DA A6 C5 LDA B,U Sélectionne
E0DC A7 02 STA 2,X le lecteur
E0DE 97 57 STA <$57 adéquats.
E0E0 E6 01 LDB 1,X Si disque
E0E2 C5 40 BITB #$40 QDD absent, erreur
E0E4 26 02 BNE $E0E8 "lecteur non prêt"
E0E6 4F CLRA Pas d'erreur dans CC
E0E7 39 RTS
  
```

```

Fixe l'erreur de lecteur
E0E8 86 10 LDA #$10 Erreur "Lecteur non prêt"
E0EA 16 01 B4 LBRA $E2A1 Fixe l'erreur
  
```

```

Extinction du moteur du QDD
E0ED 86 00 LDA #$00 Extinction du
E0EF A7 02 STA 2,X moteur QDD
E0F1 39 RTS
  
```

```

Table des bits de lecteurs
E0F2 01 FCB $01 Lecteur 0
E0F3 00 FCB $00 Lecteur 1 (inutile)
E0F4 02 FCB $02 Lecteur 2
E0F5 00 FCB $00 Lecteur 3 (inutile)
  
```

```

Traitement de l'opération QDD
E0F6 96 48 LDA <$48 Fonction demandée
E0F8 85 01 BITA #$01 Si reset du contrôleur
E0FA 26 0B BNE $E107 demandé, traite
E0FC 85 08 BITA #$08 Si écriture d'un secteur
E0FE 26 12 BNE $E112 demandée, traite
E100 85 02 BITA #$02 Si lecture d'un secteur
E102 10 26 01 2E LBNE $E234 demandé, traite
E106 39 RTS
  
```

```

Initialisation du contrôleur QDD
E107 86 43 LDA #C "C" pour simple densité
E109 97 4E STA <$4E dans registre d'état
E10B 86 FF LDA #$FF Simple densité
E10D 97 58 STA <$58 initialisée
E10F 6F 84 CLR ,X Reset lecteur
E111 39 RTS
  
```

```

Ecriture d'un secteur QDD
E112 96 4A LDA <$4A Lit flag de secteur réel
E114 26 05 BNE $E11B Si réel, passe
E116 17 01 99 LBSR $E2B2 Recherche correspondance
E119 25 55 BCS $E170 Si erreur, sort
E11B 8D 58 BSR $E175 Teste protection en écriture
E11D 25 51 BCS $E170 Si disque protégé, sort
E12E CC 03 04 LDD #$0304 Initialise en
E131 E7 84 STB ,X écriture MFM
---- Ecriture de 3 espaces
E133 E6 84 LDB ,X Attend fin de
E135 2A FC BPL $E133 transmission
E137 C6 16 LDB #$16 Valeur pour un espace
E139 E7 03 STB $03,X Ecrit le caractère
  
```


Contrôleur interne des TO8 et TO8D (THMFC1) - 2ème banque

```

E685 23 F2      BLS  $E679      | est > à 80, rajoute 2 pour
E687 CB 02     ADDB #02       | établir une incrémentation
E689 C1 A1     CMPB #161      | Au-delà du dernier bloc ?
E68B 20 ED     BRA  $E67A     | Boucle pour test
---- Recherche par symétrie d'un bloc libre dans la FAT (bloc de départ)
E68D C6 50     LDB  #80       | Offset de départ (80)
E68F 86 05     LDA  #05       | Code erreur disque plein
E691 C1 A0     CMPB #160      | Si dernier test,
E693 10 22 FE F8 LBHI $E58F     | sort avec erreur
E697 A6 A5     LDA  B,Y      | Si la place
E699 81 FF     CMPA #0FF     | est libre,
E69B 27 14     BEQ  $E6B1     | sort
E69D 34 04     PSHS B        | Préserve l'offset de bloc
E69F C0 50     SUBB #80      | Calcule la symétrie
E6A1 50       NEGB          | par rapport au
E6A2 CB 50     ADDB #80      | bloc 80
E6A4 A6 A5     LDA  B,Y      | Si la place
E6A6 81 FF     CMPA #0FF     | est libre,
E6A8 27 05     BEQ  $E6AF     | sort
E6AA 35 04     PULS B        | Restitue l'offset de bloc
E6AC 5C       INCB          | Place
E6AD 20 E0     BRA  $E68F     | suivante
E6AF 32 61     LEAS $01,S   | Rétablit la pile
E6B1 6F A5     CLR  B,Y      | Valide la place du bloc
E6B3 5A       DECB          | Ajuste et stocke
E6B4 D7 F9     STB  <$F9    | le numéro de bloc
E6B6 4F       CLRA          | Sort sans erreur
E6B7 39       RTS

```

```

Initialisation pour opération sur un bloc
E6B8 D6 F6     LDB  <$F6    | Numéro de bloc courant
E6BA 4F       CLRA          | Calcule le
E6BB 54       LSRB         | numéro
E6BC DD FB     STD  <$FB    | de piste
E6BE 4C       INCA          | Initialisation du compteur
E6BF 97 F5     STA  <$F5    | de secteur
E6C1 24 02     BCC  $E6C5   | Selon la parité du numéro
E6C3 86 09     LDA  #09     | de bloc, le premier secteur
E6C5 97 FA     STA  <$FA    | est le secteur 1 ou 9
E6C7 39       RTS

```

***** SYSTEME D'EXPLOITATION PHYSIQUE DU RAMDISK *****

```

Ecriture d'un secteur RamDisk
E6C8 8D 18     BSR  $E6E2   | Correspondance pour 64 kilos
E6CA 34 02     PSHS A       | Protège la banque initiale
E6CC 8D 47     BSR  $E715   | Secteur RamDisk trouvé ?
E6CE 25 08     BCS  $E6D8   | Non, sort avec erreur
E6D0 5F       CLRB          |
E6D1 A6 A0     LDA  ,Y+     | Ecrit dans
E6D3 A7 C0     STA  ,U+     | le secteur
E6D5 5A       DECB          | RamDisk
E6D6 26 F9     BNE  $E6D1   |
E6D8 35 02     PULS A       | Rétablit la banque
E6DA B7 E7 E6 STA  $E7E6 | initiale
E6DD DC 55     LDD  <$55    | Rétablit le numéro
E6DF DD 4A     STD  <$4A    | de piste
E6E1 39       RTS

```

```

Correspondance des pistes pour RamDisk de 64 kilooctets
E6E2 DC 4A     LDD  <$4A    | Protection du registre
E6E4 DD 55     STD  <$55    | de piste
E6E6 C1 14     CMPB #014   | Si piste 20, sort
E6E8 27 15     BEQ  $E6FF   | immédiatement
E6EA 96 CD     LDA  <0CD   |
E6EC 84 07     ANDA #07    | Si la capacité
E6EE 81 01     CMPA #01    | du RamDisk
E6F0 26 0D     BNE  $E6FF   | est de 64 kilooctets,
E6F2 C1 25     CMPB #025   | il est
E6F4 27 09     BEQ  $E6FF   | nécessaire
E6F6 C0 10     SUBB #010   | de retirer 16 au numéro
E6F8 C1 03     CMPB #003   | de piste et si la piste
E6FA 23 01     BLS  $E6FD   | est au-delà de la piste
E6FC 5A       DECB          | 20, lui retirer 1.
E6FD D7 4B     STB  <$4B    | Stocke le nouveau numéro
E6FF B6 E7 E6 LDA  $E7E6 | Numéro banque initiale dans A
E702 39       RTS

```

```

Lecture d'un secteur RamDisk
E703 8D DD     BSR  $E6E2   | Correspondance pour 64 kilos
E705 34 02     PSHS A       | Protège la banque initiale
E707 8D 0C     BSR  $E715   | Secteur RamDisk trouvé ?
E709 25 CD     BCS  $E6D8   | Non, sort avec erreur
E70B 5F       CLRB          |
E70C A6 C0     LDA  ,U+     | Lit les 256
E70E A7 A0     STA  ,Y+     | octets du
E710 5A       DECB          | secteur RamDisk
E711 26 F9     BNE  $E70C   |
E713 20 C3     BRA  $E6D8   | Sortie sans erreur

```

```

Positionnement sur secteur
E715 10 9E 4F LDY  <$4F    | Si le buffer de secteur
E718 10 8C 40 00 CMPY #04000 | se trouve en mémoire basse,
E71C 25 27     BLO  $E745   | erreur de lecteur.

```

```

E71E 96 CD     LDA  <0CD   | Si le RamDisk
E720 84 07     ANDA #07    | n'existe pas,
E722 27 1E     BEQ  $E742   | erreur de contrôleur.
E724 81 01     CMPA #01    | S'il s'agit
E726 26 0C     BNE  $E734   | d'un RamDisk de 64 kilos,
E728 96 4B     LDA  <04B   | vérifie que le numéro
E72A 81 14     CMPA #014   | de piste
E72C 27 0D     BEQ  $E73B   | (20 exclu)
E72E 81 0E     CMPA #00E   | ne dépasse pas
E730 23 09     BLS  $E73B   | 14, sinon
E732 20 0E     BRA  $E742   | contrôleur non prêt.
E734 C6 10     LDB  #010   | Si RamDisk > à 64 kilos,
E736 3D       MUL          | vérifie que le numéro de
E737 D1 4B     CMPB <$4B   | piste n'est pas hors champ
E739 23 07     BLS  $E742   | sinon contrôleur non prêt.
E73B 8D 0E     BSR  $E74B   | Calcule pointeur sur secteur
E73D 8D 27     BSR  $E766   | Active la banque requise
E73F 0F 4E     CLR  <$4E   | Sort sans erreur
E741 39       RTS

```

```

Erreurs RamDisk
E742 86 40     LDA  #040   | Erreur "Contrôleur non prêt"
E744 7D       FCB  $7D   | >TST $8610
E745 86 10     LDA  #010   | Erreur "Lecteur non prêt"
E747 97 4E     STA  <$4E   | Fixe l'erreur
E749 43       COMA          | Erreur dans CC
E74A 39       RTS

```

```

Pointeur sur secteur RamDisk
E74B 96 4C     LDA  <04C   |
E74D 5F       CLRB          |
E74E 4A       DECA          | Ajuste l'offset
E74F 1F 03     TFR  D,U     | de secteur
E751 96 4B     LDA  <04B   | S'il s'agit de la piste
E753 81 14     CMPA #014   | 20, sort immédiatement
E755 27 0E     BEQ  $E765   |
E757 22 01     BHI  $E75A   | Si la piste est en deçà de
E759 4C       INCA          | 20, n° de piste + 1
E75A 84 03     ANDA #003   |
E75C 27 07     BEQ  $E765   | Ajuste
E75E 33 C9 10 00 LEAU $1000,U | l'offset
E762 4A       DECA          | de piste
E763 26 F9     BNE  $E75E   |
E765 39       RTS

```

```

Commutation banque RamDisk
E766 D6 4B     LDB  <$4B   | S'il s'agit de la piste 20,
E768 C1 14     CMPB #014   | la banque à sélectionner
E76A 27 06     BEQ  $E772   | est la plus éloignée.
E76C 22 01     BHI  $E76F   | Sinon, numéro plus 1 si
E76E 5C       INCB          | inférieur à 20 puis
E76F 54       LSRB         | divise par 4 pour offset
E770 54       LSRB         | de banque.
E771 21       FCB  $21   | >BRN $E7D2
E772 5F       CLRB          | Offset pour piste 20
E773 D7 6C     STB  <$6C   |
E775 96 CD     LDA  <0CD   | Calcul du
E777 85 10     BITA #010   | numéro de
E779 27 03     BEQ  $E77E   | banque maximum
E77B 86 1F     LDA  #01F   | - 31 ($1F) avec extension,
E77D 7D       FCB  $7D   | >TST $860F
E77E 86 0F     LDA  #00F   | 15 ($0F) sans extension.
E780 90 6C     SUBA <$6C   | Soustrait l'offset de
E782 8A 60     ORA  #060   | banque, autorise écriture
E784 B7 E7 E6 STA  $E7E6 | et commute la banque.
E787 39       RTS

```

```

Formatage du RamDisk
E788 D6 CD     LDB  <0CD   | Lit état périphériques
E78A C4 07     ANDB #007   | Si pas de RamDisk, erreur
E78C 27 B4     BEQ  $E742   | "contrôleur non prêt"
E78E B6 E7 E6 LDA  $E7E6 | Protège la banque
E791 34 02     PSHS A       | courante
E793 96 CD     LDA  <0CD   | Selon la présence ou
E795 85 10     BITA #010   | l'absence de
E797 27 03     BEQ  $E79C   | l'extension mémoire,
E799 86 7F     LDA  #07F   | active en écriture
E79B 7D       FCB  $7D   | >TST $860F
E79C 86 6F     LDA  #06F   | la banque de numéro
E79E B7 E7 E6 STA  $E7E6 | maximum
E7A1 10 9E 76 LDY  <$76    | Pointe sur piste 20 ($0000)
E7A4 8E FF 2D LDX  #0FF2D | Nom de baptême du RamDisk
E7A7 7E F3 41 JMP  $F341   | Initialise la piste 20

```

Les programmes suivants sont cités à titre d'information. Ils ne font pas partie de la ROM disque.

***** SYSTEME D'EXPLOITATION PHYSIQUE DU RAMDISK (SUITE) *****

Formatage du RamDisk (suite)

Contrôleur interne des TO8 et TO8D (THMFC1) - 2ème banque

```

---- Initialisation du secteur de nom de disque
F341 A6 80 LDA ,X+
F343 A7 A0 STA ,Y+
F345 10 8C 00 08 CMPY #$0008 Transfère le nom
F349 26 F6 BNE $F341 du RamDisk
F34B 6F A0 CLR ,Y+
F34D 10 8C 01 00 CMPY #$0100 Efface le reste du secteur
F351 26 F8 BNE $F34B
---- Initialise la piste catalogue
F353 86 FF LDA #$FF Valeur pour initialisation
F355 A7 A0 STA ,Y+
F357 10 8C 10 00 CMPY #$1000 Comble jusqu'à la fin
F35B 26 F8 BNE $F355
F35D 86 10 LDA #$10
F35F 3D MUL Calcule nombre de pistes
---- Initialisation du secteur de FAT
F360 10 8E 01 00 LDY #$0100 Ptr secteur de FAT
F364 6F A0 CLR ,Y+ Efface premier octet
F366 86 FF LDA #$FF Taille reste du secteur
F368 8E FF FF LDX #$FFFF Valeur pour remplissage
F36B AF A1 STX ,Y++ Valide la piste
F36D 80 02 SUBA #$02 Ajourne taille remplissage
F36F 5A DECB Pour toutes les
F370 26 F9 BNE $F36B pistes du RamDisk
F372 C6 FE LDB #$FE Valeur pour remplissage
F374 E7 A0 STB ,Y+
F376 4A DECA Comble le reste de FAT
F377 26 FB BNE $F374
F379 96 CD LDA <$CD Lit état périphériques
F37B 7E FF 02 JMP $FF02 Conclut le formatage

.....

---- Teste si RamDisk de 64 kilooctets demandé
FF02 84 07 ANDA #$07 Isole taille du RamDisk
FF04 81 01 CMPA #$01 Si pas RamDisk 64 kilos,
FF06 26 17 BNE $FF1F passe
---- Initialisation de la FAT si RamDisk de 64 kilooctets demandé
FF08 8E 01 01 LDX #$0101 Ptr sur début de FAT

FF0B 86 FE LDA #$FE Valeur à $FE
FF0D A7 80 STA ,X+
FF0F 8C 02 00 CMPX #$0200 Remplit le secteur de FAT
FF12 26 F9 BNE $FF0D
FF14 4C INCA Valeur à $FF
FF15 8E 01 21 LDX #$0121 Ptr sur bloc 32
FF18 A7 80 STA ,X+
FF1A 8C 01 41 CMPX #$0141 Remplit jusqu'au bloc 64
FF1D 26 F9 BNE $FF18
---- Conclut l'initialisation de la FAT pour tout RamDisk
FF1F 8E FE FE LDX #$FEFE Réserve la piste
FF22 BF 01 29 STX $0129 du catalogue
FF25 35 02 PULS A Rétablit la banque
FF27 B7 E7 E6 STA $E7E6 cartouche
FF2A 0F 4E CLR <$4E Sort sans erreur
FF2C 39 RTS

Nom de baptême du RamDisk
FF2D 52 61 6D 20 44 FCC "Ram Disk"
FF32 69 73 6B

.....

Exécution d'un programme dans l'autre page ROM du moniteur
FFA0 34 03 PSHS A,CC Protège le CC
FFA2 1A 50 ORCC #$50 Gel des interruptions
FFA4 B6 E7 C3 LDA $E7C3
FFA7 8A 10 ORA #$10
FFA9 B7 E7 C3 STA $E7C3
FFAC 35 03 PULS CC,A Rétablit les registres
FFAE AD 9F 60 55 JSR [$6055] Exécute le programme
FFB2 34 03 PSHS A,CC Protège le CC
FFB4 1A 50 ORCC #$50 Gel des interruptions
FFB6 B6 E7 C3 LDA $E7C3
FFB9 8A 10 ORA #$10 Revient en ROM initiale
FFBB B7 E7 C3 STA $E7C3
FFBE 35 83 PULS CC,A,PC

```

Contrôleur interne du TO9+ (THMFC1) - 2ème banque

Identificateur de RomDisk (IDDISK)
 E000 4D FCC "M" Contrôleur Thomson
 E001 4B FCC "K" Fat 80 octets
 E002 43 FCC "C" Simple densité
 E003 30 FCB \$30 Checksum de l'identificateur

Indirections standard sur les opérations physiques
 E004 16 00 6D Fonctions standard QDD/RAM
 E007 16 00 1E LBRSA \$E028 Lancement du boot QDD
 E00A 16 02 AA LBRSA \$E2B7 Formatage QDD/RAM
 ---- Indirections standard sur les opérations logiques
 E00D 16 04 8D LBRSA \$E49D Chargement de la FAT
 E010 16 05 0F LBRSA \$E522 Ouverture d'un fichier
 E013 16 04 EE LBRSA \$E504 Effacement d'un fichier
 E016 16 04 CA LBRSA \$E4E3 Ecriture d'un secteur
 E019 16 05 6E LBRSA \$E58A Création d'un fichier
 E01C 16 05 C4 LBRSA \$E5E3 Allocation d'un bloc
 E01F 16 06 06 LBRSA \$E628 Initialise opération sur bloc
 E022 16 04 21 LBRSA \$E446 Clôture d'écriture
 ---- Indirections standard sur les opérations physiques
 E025 16 00 43 LBRSA \$E06B Fonctions standard QDD/RAM

***** SYSTEME D'EXPLOITATION PHYSIQUE DU QDD *****

Lancement du boot QDD
 E028 17 00 88 LBSR \$E0B3 Initialisation registres
 E02B 0F 49 CLR <\$49 Lecteur 0 demandé
 E02D 17 00 99 LBSR \$E0C9 Sélectionne le lecteur
 E030 17 00 CC LBSR \$E0FF Initialise en simple densité
 E033 25 30 BCS \$E065 Si erreur, sort.
 E035 CC 00 44 LDD #S0044 Secteur 68 (Boot QDD)
 E038 DD 4C STD <\$4C sélectionné
 E03A CE 62 00 LDU #S6200 Ptr buffer secteur
 E03D DF 4F STU <\$4F sélectionné
 E03F 17 01 F7 LBSR \$E239 Lit le secteur QDD
 E042 DE 4F LDU <\$4F Ptr buffer secteur
 E044 25 1F BCS \$E065 Si erreur, sort
 E046 17 00 9C LBSR \$E0E5 Eteint le lecteur
 E049 10 8E 62 7F LDY #S627F Limite du décodage
 E04D 34 20 PSHS Y en pile
 E04F 86 55 LDA #S55 Checksum de départ
 E051 6A C4 DEC ,U Décode les
 E053 63 C4 COM ,U données
 E055 AB C0 ADDA ,U+ Calcule le checksum
 E057 11 A3 E4 CMPU ,S Si pas fin du décodage,
 E05A 26 F5 BNE \$E051 boucle
 E05C 35 20 PULS Y Rétablit la pile
 E05E A1 C4 CMPA ,U Si checksum incorrect,
 E060 26 03 BNE \$E065 lance l'application
 E062 7E 62 00 JMP \$E200 Sinon, exécute le boot
 ---- Sortie si erreur de boot
 E065 0F 80 CLR <\$80 Flag "contrôleur absent"
 E067 6E 9F 00 1E JMP [\$001E] Lance l'application à froid

Opérations standard avec numéro de secteur QDD réel
 E06B 34 7F PSHS U ,Y ,X ,DP ,B ,A ,CC
 E06D 86 FF LDA #FFF Flag pour opérations
 E06F B7 60 4A STA \$604A QDD avec secteur réel
 E072 20 05 BRA \$E079 Lance l'opération

Opérations standard avec numéro de secteur QDD interprété
 E074 34 7F PSHS U ,Y ,X ,DP ,B ,A ,CC
 E076 7F 60 4A CLR \$604A Flag de secteur interprété
 E079 8D 38 BSR \$E0B3 Initialise registres DP et X
 E07B 96 49 LDA <\$49 lit le numéro de lecteur
 E07D 81 04 CMPA #S04 Si pas lecteur RamDisk,
 E07F 26 1B BNE \$E09C opérations QDD
 ---- Traite la lecture du secteur RamDisk
 E081 96 48 LDA <\$48 Lit la commande
 E083 85 02 BITA #S02 Si pas lecture secteur
 E085 27 05 BEQ \$E08C passe
 E087 BD E6 73 JSR \$E673 Lit le secteur RamDisk
 E08A 20 20 BRA \$E0AC Sort du programme
 ---- Traite l'écriture du secteur RamDisk
 E08C 85 08 BITA #S08 Si pas écriture secteur,
 E08E 27 05 BEQ \$E095 sort avec erreur
 E090 BD E6 38 JSR \$E638 Ecrit le secteur RamDisk
 E093 20 17 BRA \$E0AC Sort du programme
 ---- Erreur si contrôleur non prêt
 E095 86 40 LDA #S40 "Contrôleur non prêt"
 E097 97 4E STA <\$4E dans registre d'erreur
 E099 43 COMA Erreur dans CC
 E09A 20 10 BRA \$E0AC Sort avec erreur
 ---- Exécution de l'opération
 E09C DC 4B LDD <\$4B Protection de
 E09E 34 06 PSHS B ,A DKTRK et DKSEC
 E0A0 8D 27 BSR \$E0C9 Test disponibilité lecteur
 E0A2 25 02 BCS \$E0A6 Si erreur, sort
 E0A4 8D 48 BSR \$E0EE Exécute l'opération
 ---- Sortie du programme
 E0A6 35 06 PULS A ,B Restaure DKTRK

E0A8 DD 4B STD <\$4B et DKSEC
 E0AA 8D 39 BSR \$E0E5 Eteint le lecteur
 E0AC 35 02 PULS A Dépile le CC
 E0AE 1E 8A EXG A ,CC Restaure le CC
 E0B0 44 LBSRA Bit d'erreur dans CC
 E0B1 35 FE PULS A ,B ,DP ,X ,Y ,U ,PC

Initialisation des registres disque
 E0B3 86 60 LDA #S60 Initialise
 E0B5 1F 8B TFR A ,DP le DP
 E0B7 8E E7 D0 LDX #S7D0 Ptr sur registres disques
 E0BA 0F 4E CLR <\$4E Efface code d'erreur
 E0BC 6F 84 CLR ,X Reset lecteur
 E0BE 86 CE LDA #SCE Séparateur de donnée
 E0C0 A7 07 STA \$07 ,X avec compteur de \$4E
 E0C2 86 FF LDA #SFF Programme le mode données
 E0C4 A7 04 STA \$04 ,X le mode "donnée"
 E0C6 6F 01 CLR \$01 ,X Pas de précompensation
 E0C8 39 RTS

Sélectionne le lecteur QDD
 E0C9 D6 49 LDB <\$49 Lit numéro de lecteur
 E0CB C5 FD BITB #SFD Si ni drive 0 ni drive 2,
 E0CD 26 11 BNE \$E0E0 erreur "lecteur non prêt"
 E0CF CE E0 EA LDU #S0E0A Ptr table bits lecteurs
 E0D2 A6 C5 LDA B ,U Lit bits de lecteur
 E0D4 A7 02 STA \$02 ,X Initialise registre disque
 E0D6 97 57 STA <\$57 Initialise registre système
 E0D8 86 01 LDB \$01 ,X Lit status
 E0DA C5 40 BITB #S40 Si disquette absente,
 E0DC 26 02 BNE \$E0E0 erreur "lecteur non prêt"
 E0DE 4F CLRA Pas d'erreur dans CC
 E0DF 39 RTS

Fixe l'erreur de lecteur
 E0E0 86 10 LDA #S10 Erreur "Lecteur non prêt"
 E0E2 16 01 BA LBSRA \$E29F Fixe l'erreur

Extinction du moteur du QDD
 E0E5 86 00 LDA #S00 Extinction du
 E0E7 A7 02 STA \$02 ,X moteur QDD
 E0E9 39 RTS

Table des bits de lecteurs
 E0EA 01 FCB \$01 Lecteur 0
 E0EB 00 FCB \$00 Lecteur 1 (inusité)
 E0EC 02 FCB \$02 Lecteur 2
 E0ED 00 FCB \$00 Lecteur 3 (inusité)

Traitement de l'opération QDD
 E0EE 96 48 LDA <\$48 Fonction demandée
 E0F0 85 01 BITA #S01 Si reset du contrôleur
 E0F2 26 08 BNE \$E0FF demandé, traite
 E0F4 85 08 BITA #S08 Si écriture d'un secteur
 E0F6 26 12 BNE \$E10A demandé, traite
 E0F8 85 02 BITA #S02 Si lecture d'un secteur
 E0FA 10 26 01 33 LBNE \$E231 demandée, traite
 E0FE 39 RTS

Initialisation du contrôleur QDD
 E0FF 86 43 LDA #S43 "C" pour simple densité
 E101 97 4E STA <\$4E dans registre d'état
 E103 86 FF LDA #SFF Simple densité
 E105 97 58 STA <\$58 initialisée
 E107 6F 84 CLR ,X Reset lecteur
 E109 39 RTS

Ecriture d'un secteur QDD
 E10A 96 4A LDA <\$4A Lit flag de secteur réel
 E10C 26 05 BNE \$E113 Si réel, passe
 E10E 17 01 A0 LBSR \$E2B1 Recherche correspondance
 E111 25 59 BCS \$E16C Si erreur, sort
 E113 17 00 5B LBSR \$E171 Teste protection en écriture
 E116 25 54 BCS \$E16C Si disque protégé, sort
 E118 17 00 E1 LBSR \$E1FC Teste disponibilité lecteur
 E11B 25 4F BCS \$E16C Si erreur, sort
 E11D 34 01 PSHS CC Préserve les interruptions
 E11F 8D 5C BSR \$E17D Recherche du secteur demandé
 E121 25 4A BCS \$E16D Si erreur, sort
 E123 C6 FF LDB #SFF Initialise en
 E125 E7 04 STB \$04 ,X mode données
 E127 86 03 LDA #S03 3 caractères à écrire
 E129 C6 04 LDB #S04 Initialise en
 E12B E7 84 STB ,X écriture MFM
 ---- Ecriture de 3 espaces
 E12D E6 84 LDB ,X Attend fin de
 E12F 2A FC BPL \$E12D transmission
 E131 C6 16 LDB #S16 Valeur pour un espace
 E133 E7 03 STB \$03 ,X Ecrit le caractère
 E135 4A DECA Pour les 3
 E136 26 F5 BNE \$E12D caractères
 ---- Ecriture de l'identificateur de données
 E138 E6 84 LDB ,X Attend fin de

Contrôleur interne du TO9+ (THMFC1) - 2ème banque

```

E3D7 C6 08 LDB #S08 Exclut la détection de
E3D9 E7 84 STB ,X synchronisation
E3DB 10 9E 4C LDY <$4C Lit numéro de secteur
E3DE 10 8C 00 05 CMPY #S0005 Si secteur de FAT,
E3E2 27 1E BEQ $E402 vérification spéciale
E3E4 E6 84 LDB ,X Attend fin de
E3E6 2A FC BPL $E3E4 transmission
E3E8 E6 03 LDB $03,X Lit le caractère
E3EA C1 FF CMPB #SFF, Si pas $FF,
E3EC 26 0F BNE $E3FD erreur de vérification
E3EE 4A DECA Pour les 128
E3EF 26 F3 BNE $E3E4 caractères
---- Vérification de la validité du checksum
E3F1 E6 84 LDB ,X Attend fin de
E3F3 2A FC BPL $E3F1 transmission
E3F5 E6 03 LDB $03,X Lit le caractère
E3F7 6F 84 CLR ,X Reset lecteur
E3F9 C1 DA CMPB #SDA Si checksum correct,
E3FB 27 1A BEQ $E417 secteur suivant
E3FD 86 20 LDA #S20 Code "erreur de vérification"
E3FF 16 FE 9D LBRA $E29F Sort avec erreur
---- Vérification du secteur de FAT
E402 E6 84 LDB ,X Attend fin de
E404 2A FC BPL $E402 transmission
E406 E6 03 LDB $03,X Si caractère lu <0,
E408 26 F3 BNE $E3FD erreur de vérification
E40A E6 84 LDB ,X Attend fin de
E40C 2A FC BPL $E40A transmission
E40E E6 03 LDB $03,X Lit le caractère
E410 E1 C0 CMPB ,U+ Compare avec secteur de FAT
E412 26 E9 BNE $E3FD Si erreur, sort
E414 4A DECA Pour les 128
E415 26 F3 BNE $E40A caractères
---- Secteur suivant
E417 35 01 PULS CC Rétablit les interruptions
E419 10 9E 4C LDY <$4C Lit numéro de secteur
E41C 10 8C 01 90 CMPY #S0190 Si pas dernier secteur,
E420 26 8B BNE $E3AD secteur suivant
E422 39 RTS
---- Sortie si erreur
E423 35 01 PULS CC Rétablit les interruptions
E425 43 COMA Erreur dans CC
E426 39 RTS
Exécution de toute opération standard
E427 34 42 PSHS U,A Préserve registres utilisés
E429 CE 00 04 LDU #S8004 Pointeur fonctions standard
E42C DF 55 STU <$55 dans le jump de programme
E42E 96 CD LDA <$CD Etat périphériques externes
E430 48 ASLA Teste si floppy
E431 35 42 PULS A,U
E433 24 03 BCC $E438 Si non, reste en banque 1
E435 7E FF A0 JMP $FFA0 Fonctions floppy (bnk 0)
E438 7E E0 04 JMP $E004 Fonctions RamDisk-QDD (bnk 1)
***** SYSTEME D'EXPLOITATION LOGIQUE *****
Nom de fichier temporaire pour la sauvegarde
E43B 53 43 52 41 FCC "SCRATCH.DOS"
E43F 54 43 48 20
E443 44 4F 53
Clôture d'écriture
E446 D6 F0 LDB <$F0 Lit code commande
E448 C1 02 CMPB #S02 Si écrasement demandé,
E44A 27 25 BEQ $E471 écrit FAT seule
E44C 0A F0 DEC <$F0 Passe en sauevg. sans ecr.
E44E BD E5 22 JSR $E522 Recherche fichier courant
E451 25 31 BCS $E484 Si erreur disque, sort
E453 5D TSTB Si l'entrée n'existe pas,
E454 27 05 BEQ $E45B pas d'effacement
E456 17 00 AB LBSR $E504 Effacement du fichier
E459 25 29 BCS $E484 Si erreur, sort
E45B 0C F0 INC <$F0 Retour en sauevg avec écrasmt
E45D BD E5 22 JSR $E522 Cherche fichier "SCRATCH.DOS"
E460 25 22 BCS $E484 Si erreur, sort
E462 C6 0A LDB #S0A Recopie le nom
E464 9E E7 LDY <$E7 du fichier
E466 A6 85 LDA B,X courant à la place
E468 A7 A5 STA B,Y de "SCRATCH.DOS"
E46A 5A DECB
E46B 2C F9 BGE $E466
E46D 8D 74 BSR $E4E3 Ecrit le secteur catalogue
E46F 25 13 BCS $E484 Si erreur de disque, sort
---- Sauvegarde de la FAT
E471 86 02 LDA #S02 Programme la
E473 97 4C STA <$4C sauvegarde
E475 C6 14 LDB #S14 de la FAT
E477 4F CLRA
E478 DD 4A STD <$4A
E47A 8D 4B BSR $E4C7 Rétablissement de FAT swapée
E47C 9F 4F STX <$4F Sauvegarde
E47E 8D 63 BSR $E4E3 de la FAT
E480 25 02 BCS $E484 Si erreur de disque, sort

```

```

E482 0F F0 CLR <$F0 Clôture du fichier effectuée
E484 39 RTS
Swap d'une FAT double densité
E485 34 07 PSHS B,A,CC
E487 9E ED LDY <$ED
E489 30 89 00 80 LEAX $0080,X
E48D A6 1F LDA -$01,X
E48F E6 88 7F LDB $7F,X
E492 A7 88 7F STA $7F,X
E495 E7 82 STB ,X Déplace pointeur à rebours
E497 9C ED CMPX <$ED Fin de FAT ?
E499 26 F2 BNE $E48D Non, échange suivant
E49B 35 87 PULS CC,A,B,PC
Chargement de la FAT
E49D DC ED LDD <$ED Pointeur sur FAT swapée
E49F 0D 25 TST <$25 FAT système initialisée ?
E4A1 26 02 BNE $E4A5 Non, initialise avec
E4A3 DD 25 STD <$25 pointeur sur FAT swapée
E4A5 93 25 SUBD <$25 Si la différence entre les
E4A7 C1 80 CMPB #S80 deux est de 128 octets,
E4A9 27 03 BEQ $E4AE prend le pointeur système
E4AB 9E ED LDY <$ED Récupère le pointeur de
E4AD 7D FCB >TST $9E25 début de secteur de FAT
E4AE 9E 25 LDY <$25 et initialise les
E4B0 9F ED STX <$ED pointeurs pour charger
E4B2 9F 4F STX <$4F le secteur
E4B4 86 02 LDA #S02 Chargement
E4B6 8D 31 BSR $E4E9 de la FAT
E4B8 25 CA BCS $E484 Si erreur disque, sort
E4BA 0D 58 TST <$58 Contrôleur double densité ?
E4BC 26 08 BNE $E4C6 Oui, swap
E4BE 8D C5 BSR $E485 de la FAT
E4C0 30 89 00 80 LEAX $0080,X et pointeur courant
E4C4 9F ED STX <$ED en milieu de FAT
E4C6 39 RTS
Rétablissement de la FAT swapée
E4C7 9E ED LDY <$ED S'il s'agit
E4C9 0D 58 TST <$58 d'une FAT
E4CB 26 F9 BNE $E4C6 double densité,
E4CD 30 88 80 LEAX -$80,X repositionne le pointeur
E4D0 8D F2 BSR $E4C4 de FAT en début de secteur
E4D2 20 B1 BRA $E485 et rétablit la FAT swapée
Chargement du premier secteur de catalogue
E4D4 86 03 LDA #S03 N° du ler secteur catalogue
E4D6 9E E9 LDY <$E9 Pointeur sur buffer
E4D8 9F 4F STX <$4F catalogue initialisé
E4DA 8D 0D BSR $E4E9 Chargement du secteur
E4DC 25 E9 BCS $E4C7 Si erreur, rétablit la FAT
E4DE 39 RTS Sort sans erreur
Chargement d'un secteur
E4DF 8D 0F BSR $E4F0 Charge le secteur
E4E1 20 F9 BRA $E4DC Si erreur, rétablit la FAT
Routine de sauvegarde d'un secteur
E4E3 86 08 LDA #S08 Sauve le
E4E5 8D 0B BSR $E4F2 secteur
E4E7 20 F3 BRA $E4DC Si erreur, rétablit la FAT
Charge un secteur de la piste 20
E4E9 97 4C STA <$4C Sélection du secteur
E4EB C6 14 LDB #S14 Piste 20 demandée
E4ED 4F CLRA
E4EE DD 4A STD <$4A
E4F0 86 02 LDA #S02 Lecture du secteur demandée
E4F2 97 48 STA <$48 Type d'opération programmé
E4F4 10 9E E9 LDY <$E9 Initialise ptr buffer secteur
E4F7 17 FF 2D LBSR $E42 Opération disquette standard?
E4FA 86 03 LDA #S03 Code "erreur d'entrée-sortie"
E4FC 39 RTS
Sortie si erreur
E4FD 8D 08 BSR $E4C7 Rétablit la FAT si swapée,
E4FF 97 E5 STA <$E5 stocke erreur système,
E501 43 COMA erreur dans CC
E502 43 COMA et rétablit code d'erreur
E503 39 RTS
Effacement du fichier courant
E504 A6 2D LDA $0D,Y Premier bloc du fichier
E506 97 F6 STA <$F6 sélectionné
E508 6F A4 CLR ,Y Elimine le nom du fichier
E50A 8D D7 BSR $E4E3 Ecrit le secteur catalogue
E50C 25 F1 BCS $E4FF Si erreur disque, sort
E50E 10 9E ED LDY <$ED
E511 D6 F6 LDB <$F6 Efface
E513 5C INCB (met à $FF)
E514 A6 A5 LDA B,Y tous les blocs
E516 6F A5 CLR B,Y du fichier
E518 6A A5 DEC B,Y courant
E51A 1F 89 TFR A,B répertoriés
E51C 81 C0 CMPA #S0C dans la FAT

```


Contrôleur interne du TO9+ (THMFC1) - 2ème banque

```

E668 C1 03      CMPB  #S03      | de piste et si la piste
E66A 23 01      BLS   $E66D      | est au-delà de la piste
E66C 5A         DECB                | 20, lui retirer 1.
E66D D7 4B      STB   <$4B      | Stocke le nouveau numéro
E66F B6 E7 E6   LDA   $E7E6      | Numéro banque initiale dans A
E672 39        RTS
    
```

Lecture d'un secteur RamDisk

```

E673 8D DD      BSR   $E652      | Correspondance pour 64 kilos
E675 34 02      PSHS  A          | Protège la banque initiale
E677 8D 0C      BSR   $E685      | Secteur RamDisk trouvé ?
E679 25 CD      BCS   $E648      | Non, sort avec erreur
E67B 5F         CLRB                |
E67C A6 C0      LDA   ,U+        | Lit les 256
E67E A7 A0      STA   ,Y+        | octets du
E680 5A         DECB                | secteur RamDisk
E681 26 F9      BNE   $E67C      |
E683 20 C3      BRA   $E648      | Sortie sans erreur
    
```

Positionnement sur secteur

```

E685 10 9E 4F   LDY   <$4F      | Si le buffer de secteur
E688 10 8C 40 00 CMPY  #$4000      | se trouve en mémoire basse,
E68C 25 27      BLO   $E6B5      | erreur de lecteur.
E68E 96 CD      LDA   <$CD      | Si le RamDisk
E690 84 07      ANDA  #S07      | n'existe pas,
E692 27 1E      BEQ   $E6B2      | erreur de contrôleur.
E694 81 01      CMPA  #S01      | S'il s'agit
E696 26 0C      BNE   $E6A4      | d'un RamDisk de 64 kilos,
E698 96 4B      LDA   <$4B      | vérifie que le numéro
E69A 81 14      CMPA  #S14      | de piste
E69C 27 0D      BEQ   $E6AB      | (20 exclu)
E69E 81 0E      CMPA  #S0E      | ne dépasse pas
E6A0 23 09      BLS   $E6AB      | 14, sinon
E6A2 20 0E      BRA   $E6B2      | contrôleur non prêt.
E6A4 C6 10      LDB  #S10      | Si RamDisk > à 64 kilos,
E6A6 3D         MUL                | vérifie que le numéro de
E6A7 D1 4B      CMPB  <$4B      | piste n'est pas hors champ
E6A9 23 07      BLS   $E6B2      | sinon contrôleur non prêt.
E6AB 8D 0E      BSR   $E6BB      | Calcule pointeur sur secteur
E6AD 8D 27      BSR   $E6D6      | Active la banque requise
E6AF 0F 4E      CLR   <$4E      | Sort sans erreur
E6B1 39        RTS
    
```

Erreurs RamDisk

```

E6B2 86 40      LDA   #S40      | Erreur "contrôleur non prêt"
E6B4 7D         FCB   $7D >TST $8610
E6B5 86 10      LDA   #S10      | Erreur "lecteur non prêt"
E6B7 97 4E      STA   <$4E      | Fixe l'erreur
E6B9 43         COMA                | Erreur dans CC
E6BA 39        RTS
    
```

Pointeur sur secteur RamDisk

```

E6BB 96 4C      LDA   <$4C      |
E6BD 5F         CLRB                | Ajuste l'offset
E6BE 4A         DECA                | de secteur
E6BF 1F 03      TFR   D,U        |
E6C1 96 4B      LDA   <$4B      | S'il s'agit de la piste
E6C3 81 14      CMPA  #S14      | 20, sort immédiatement
E6C5 27 0E      BEQ   $E6D5      |
E6C7 22 01      BHI   $E6CA      | Si la piste est en deçà de
E6C9 4C         INCA                | 20, n° de piste + 1
E6CA 84 03      ANDA  #S03      |
E6CC 27 07      BEQ   $E6D5      | Ajuste
E6CE 33 C9 10 00 LEAU  $1000,U  | l'offset
E6D2 4A         DECA                | de piste
E6D3 26 F9      BNE   $E6CE      |
E6D5 39        RTS
    
```

Commutation banque RamDisk

```

E6D6 D6 4B      LDB  <$4B      | S'il s'agit de la piste 20,
E6D8 C1 14      CMPB  #S14      | la banque à sélectionner
E6DA 27 06      BEQ   $E6E2      | est la plus éloignée.
E6DC 22 01      BHI   $E6DF      | Sinon, numéro plus 1 si
E6DE 5C         INCB                | inférieur à 20 puis
E6DF 54         LSRB                | divise par 4 pour offset
E6E0 54         LSRB                | de banque.
E6E1 21         FCB   $21 >BRN $E742
E6E2 5F         CLRB                | Offset pour piste 20
E6E3 D7 6C      STB  <$6C      |
E6E5 96 CD      LDA   <$CD      | Calcul du
E6E7 85 10      BITA  #S10      | numéro de
E6E9 27 03      BEQ   $E6EE      | banque maximum
E6EB 86 1F      LDA   #S1F      | - 31 ($1F) avec extension,
E6ED 7D         FCB   $7D >TST $860F
E6EE 86 0F      LDA   #S0F      | 15 ($0F) sans extension.
E6F0 90 6C      SUBA  <$6C      | Soustrait l'offset de
E6F2 8A 60      ORA   #S60      | banque, autorise écriture
E6F4 B7 E7 E6   STA   $E7E6      | et commute la banque.
E6F7 39        RTS
    
```

Formatage du RamDisk

```

E6F8 D6 CD      LDB  <$CD      | Lit état périphériques
E6FA C4 07      ANDB  #S07      | Si pas de RamDisk, erreur
E6FC 27 B4      BEQ   $E6B2      | "contrôleur non prêt"
E6FE B6 E7 E6   LDA   $E7E6      | Protège la banque
E701 34 02      PSHS  A          | courante
    
```

```

E703 96 CD      LDA   <$CD      | Selon la présence ou
E705 85 10      BITA  #S10      | l'absence de
E707 27 03      BEQ   $E70C      | l'extension mémoire,
E709 86 7F      LDA   #S7F      | active en écriture
E70B 7D         FCB   $7D >TST $866F
E70C 86 6F      LDA   #S6F      | la banque de numéro
E70E B7 E7 E6   STA   $E7E6      | maximum
E711 10 9E 76   LDY   <$76      | Pointe sur piste 20 ($0000)
E714 8E FF 2D   LDX  #SFF2D     | Non de baptême du RamDisk
E717 7E E7 1A   JMP   $E71A     | initialise la piste 20
    
```

Note: Le "JMP \$E71A", qui ne brille que par son inutilité, est un résidu du programme pour TO8D, où l'initialisation du secteur et nom de disque (programme ci-après) avait son point d'entrée hors de la ROM contrôleur disque.

--- Initialisation du secteur de nom de disque

```

E71A A6 80      LDA   ,X+        |
E71C A7 A0      STA   ,Y+        | Transfère le nom
E71E 10 8C 00 08 CMPY  #S0008      | du RamDisk
E722 26 F6      BNE   $E71A      |
E724 6F A0      CLR   ,Y+        |
E726 10 8C 01 00 CMPY  #S0100      | Efface le reste du secteur
E72A 26 F8      BNE   $E724      |
E72C 86 FF      LDA   #SFF      |
E72E A7 A0      STA   ,Y+        | Valeur pour initialisation
E730 10 8C 10 00 CMPY  #S1000      |
E734 26 F8      BNE   $E72E      | Comble jusqu'à la fin
E736 86 10      LDA   #S10      |
E738 3D         MUL                | Calcule nombre de piste
E739 10 8E 01 00 LDY   #S0100      | du RamDisk dans B
E73D 6F A0      CLR   ,Y+        | --- Initialisation du secteur de FAT
E73F 86 FF      LDA   #SFF      | Ptr secteur de FAT
E741 8E FF FF   LDX  #SFFF      | Efface premier octet
E744 AF A1      STX  ,Y++       | Taille du reste du secteur
E746 80 02      SUBA  #S02      | Valeur pour remplissage
E748 5A         DECB                | Valide la piste
E749 26 F9      BNE   $E744      | Ajoute taille de remplissage
E74B C6 FE      LDB  #SFE      | Pour toutes les pistes
E74D E7 A0      STB  ,Y+        | du RamDisk
E74F 4A         DECA                | Valeur pour remplissage
E750 26 FB      BNE   $E74D      | Comble le reste de la FAT
E752 96 CD      LDA   <$CD      |
E754 7E FF 02   JMP   $FF02     | Lit état périphériques
    
```

Les programmes suivants sont cités à titre d'information. Ils ne font pas partie de la ROM disque.

***** SYSTEME D'EXPLOITATION PHYSIQUE DU RAMDISK (SUITE) *****

--- Formatage du RamDisk (suite)

```

E755 84 07      ANDA  #S07      | --- Teste si RamDisk de 64 kilooctets demandé
E757 81 01      CMPA  #S01      | Isole taille du RamDisk
E759 26 17      BNE   $FF1F      | Si pas RamDisk 64 kilos,
E75B 26 17      BNE   $FF1F      | passe
E75D 8E 01 01   LDX  #S0101     | --- Initialisation de la FAT si RamDisk de 64 kilooctets demandé
E75F 86 FE      LDA   #SFE      | Ptr sur début de FAT
E761 A7 80      STA   ,X+        | Valeur à $FE
E763 A7 80      STA   ,X+        |
E765 8C 02 00   CMPX  #S0200     | Remplit le secteur de FAT
E767 26 F9      BNE   $FF0D      |
E769 4C         INCA                | Valeur à $FF
E76B 8E 01 21   LDX  #S0121     | Ptr sur bloc 32
E76D A7 80      STA   ,X+        |
E76F 8C 01 41   CMPX  #S0141     | Remplit jusqu'au bloc 64
E771 26 F9      BNE   $FF18      |
E773 8E FE 01 01 LDY   #SFEFE     | --- Conclut l'initialisation de la FAT pour tout RamDisk
E775 BF 01 29   STX  $0129     | Réserve la piste
E777 35 02      PULS  A          | du catalogue
E779 B7 E7 E6   STA   $E7E6     | Rétablit la banque
E77B 0F 4E      CLR   <$4E      | cartouche
E77D 39        RTS                | Sort sans erreur
    
```

Nom de baptême du RamDisk

```

E77E 52 61 6D 20 44 FCC "Ram Disk"
E780 69 73 6B
    
```

Exécution d'un programme dans l'autre page ROM moniteur

```

E781 34 03      PSHS  A,CC       | Protège le CC
E783 1A 50      ORCC  #S50      | Gel des interruptions
E785 B6 E7 C3   LDA   $E7C3      |
E787 8A 10      ORA   #S10      | Passe dans l'autre page ROM
E789 B7 E7 C3   STA   $E7C3      |
    
```


Contrôleur interne du TO9+ (THMFC1) - 2ème banque

FFAC	35 03	PULS	CC,A	Rétablit les registres
FFAE	AD 9F 60 55	JSR	[\$6055]	Exécute le programme
FFB2	34 03	PSHS	A,CC	Protège le CC
FFB4	1A 50	ORCC	#\$50	Gel des interruptions
FFB6	B6 E7 C3	LDA	\$(7C3	
FFB9	8A 10	ORA	#\$10	Revient en ROM initiale
FFBB	B7 E7 C3	STA	\$(7C3	
FFBE	35 83	PULS	CC,A,PC	

Contrôleur interne du TO9 (WD2793)

```

Identificateur de RomDisk (IDDISK)
E000 44 FCC *D"      Contrôleur Western Digital
E001 54 FCC *T"      Fat 160 octets
E002 44 FCC *D"      Double densité
E003 31 FCB $31      Checksum descripteur ROM

Indirections standard sur les opérations physiques
E010 16 06 85 LBRA $E698 Ouverture d'un fichier
E013 16 06 64 LBRA $E67A Effacement d'un fichier
E016 16 06 40 LBRA $E659 Ecriture d'un secteur
E019 16 06 B5 LBRA $E701 Création d'un fichier
E01C 16 07 3B LBRA $E75A Allocation d'un bloc
E01F 16 07 7D LBRA $E79F Initialise opération sur bloc
E022 16 05 8C LBRA $E5B1 Clôture d'écriture

***** SYSTEME D'EXPLOITATION PHYSIQUE FLOPPY *****
Initialisation des registres
E025 86 60 LDA # $60      Initialise le DP
E027 1F 8B TFR A,DP
E029 8E E7 D0 LDX # $E7D0 Ptr sur registres disque
E02C CE E0 4A LDU # $E04A Liste des bits de lecteurs
E02F 39 RTS

Sélection des bits de lecteurs
E030 96 49 LDA <$49      Récupère numéro de lecteur
E032 91 57 CMPA <$57      Si déjà initialisé,
E034 27 13 BEQ $E049      sort
E036 BD E0 EB JSR $E0EB   Lance l'interruption lecteur
E039 5F CLR B           Pour 16 bits
E03A DD 57 STD <$57      Marque passage d'exécution
E03C A6 08 LDA $08,X     Lit le registre de lecteur
E03E 84 80 ANDA # $80     Préserve la densité
E040 8D EA BSR $E02C     Récupère ptr sur liste
E042 D6 49 LDB <$49      Lit numéro de lecteur
E044 AA C5 ORA B,U       Fixe la densité
E046 A7 08 STA $08,X     Ecrit le registre de lecteur
E048 4F CLRA            Sortie sans erreur
E049 39 RTS

Liste des bits de lecteurs
E04A 02 FCB $02         Lecteur 0
E04B 03 FCB $03         Lecteur 1
E04C 04 FCB $04         Lecteur 2
E04D 05 FCB $05         Lecteur 3

Traitement des fonctions standard
E04E 34 7F PSHS U,Y,X,DP,B,A,CC
E050 8D D3 BSR $E025     Initialise les registres
E052 96 49 LDA <$49      Lit le numéro de lecteur
E054 81 04 CMPA # $04     Si pas RAMDISK,
E056 26 09 BNE $E061     passe

--- Traitement pour RamDisk
E058 96 48 LDA <$48      Lit la commande
E05A 85 0A BITA # $0A     Si lecture ou sauvegarde
E05C 26 23 BNE $E081     secteur, passe
E05E 4F CLRA            Sortie
E05F 20 4F BRA $E0B0     sans erreur

--- Traitement pour support disque
E061 A6 84 LDA ,X        Récupère bit "NotReady"
E063 43 COMA            Sélectionne les bits lecteur
E064 97 6C STA <$6C     Lit la commande
E066 8D C8 BSR $E030     Si reset lecteur demandé,
E068 96 48 LDA <$48     exécute opération
E06A 81 01 CMPA # $01     Si lecteur prêt, passe
E06C 27 10 BEQ $E07E     Si lecteur prêt, passe
E06E D6 6C LDB <$6C     Si lecteur prêt, passe
E070 D4 58 ANDB <$58     Si lecteur prêt, passe
E072 2B 0A BMI $E07E     Si ok, passe
E074 17 02 33 LBSR $E2AA  Init position des têtes
E077 24 05 BCC $E07E     Si ok, passe
E079 17 02 24 LBSR $E2A0  Init position des têtes
E07C 20 32 BRA $E0B0     Sort du programme

--- Programmation du nombre de tentatives
E07E 86 03 LDA # $03     3 tentatives
E080 7D FCB $7D         >TST $8601
E081 86 01 LDA # $01     1 tentative
E083 34 02 PSHS A       Empile compteur tentatives

--- Exécution de la commande
E085 4F CLRA            Offset à 0
E086 D6 48 LDB <$48     Lit la commande
E088 54 LSR B           Convertit le bit
E089 4C INCA            de commande en
E08A 24 FC BCC $E088     offset
E08C 4A DECA            Ajuste
E08D 48 ASLA            l'offset
E08E 10 8E E0 B9 LDY # $E0B9  Ptr sur table des vecteurs
E092 AD B6 JSR [A,Y]     Exécute l'opération
E094 24 18 BCC $E0AE     Si pas d'erreur, sort

--- Traitement si erreur
E096 96 4E LDA <$4E     Récupère code d'erreur
E098 85 51 BITA # $51     Si erreurs contrôleur/
E09A 26 12 BNE $E0AE     lecteur/protection, sort

```

```

E004 16 00 47 LBRA $E04E Fonctions Floppy/RamDisk
E007 16 1F 8F LBRA $FF99 Lancement du boot Floppy
E00A 16 03 4A LBRA $E357 Formatage Floppy/RamDisk
---- Indirections standard sur les opérations logiques
E00D 16 05 F8 LBRA $E608 Chargement de la FAT

E09C 6A E4 DEC ,S       Décompte des tentatives
E09E 27 0D BEQ $E0AD     Si dernière tentative, sort
E0A0 0F 4E CLR <$4E     Efface code d'erreur
E0A2 81 08 CMPA # $08     Si erreur sur les données,
E0A4 27 DF BEQ $E085     nouvelle tentative
E0A6 17 02 96 LBSR $E33F Recherche la piste 0
E0A9 25 F1 BCS $E09C     Si erreur, autre tentative
E0AB 20 D8 BRA $E085     Nouvelle tentative
E0AD 53 COMB           Erreur dans CC
---- Sortie du programme
E0AE 35 02 PULS A       Rétablit la pile
E0B0 7E E3 9B JMP $E39B Sort du programme

Sélection de la simple densité
E0B3 86 80 LDA # $80     Flag simple densité
E0B5 21 FCB $21         >BRN $E106

Sélection de la double densité
E0B6 4F CLRA            Flag double densité
E0B7 20 87 BRA $E040     Fixe la densité

Table des fonctions standard
E0B9 E0 C7 FDB $E0C7     01 Reset du lecteur
E0BB E2 46 FDB $E246     02 Lecture d'un secteur
E0BD E0 B3 FDB $E0B3     04 Passage en simple densité
E0BF E1 C3 FDB $E1C3     08 Ecriture d'un secteur
E0C1 E0 B6 FDB $E0B6     10 Passage en double densité
E0C3 E3 3F FDB $E33F     20 Recherche piste 0
E0C5 E2 86 FDB $E286     40 Recherche piste demandée

Reset du lecteur
E0C7 86 44 LDA # 'D     Initialise DKSTA pour
E0C9 97 4E STA <$4E     double densité
E0CB 17 01 D2 LBSR $E2A0 Init position des têtes
E0CE DC E9 LDD <$E9     Récupère la valeur absolue
E0D0 93 ED SUBD <$ED     de la différence entre le
E0D2 2C 05 BGE $E0D9     pointeur de FAT et le
E0D4 43 COMA           pointeur de buffer secteur
E0D5 53 COMB
E0D6 C3 00 01 ADDD # $0001
E0D9 4D TSTA           Si différence > à 256,
E0DA 26 DA BNE $E0B6     double densité
E0DC 5D TSTB           Si différence < 128,
E0DD 2A D7 BPL $E0B6     double densité
E0DF 20 D2 BRA $E0B3     Sinon, simple densité

Temporisation de 234ms
E0E1 10 8E 30 00 LDY # $3000 Nombre d'itérations
E0E5 3D MUL             Temporisation
E0E6 31 3F LEAY -,$01,Y
E0E8 26 FB BNE $E0E5
E0EA 39 RTS

Lance l'interruption lecteur
E0EB C6 D0 LDB # $D0     Code "Force Interrupt"
---- Lance une opération lecteur
E0ED F7 E7 D0 STB $E7D0 Lance l'opération
E0F0 8D 08 BSR $E0FA     Temporise
E0F2 8D 06 BSR $E0FA     l'opération
E0F4 8D 04 BSR $E0FA     de 40µs
E0F6 8D 02 BSR $E0FA
E0F8 8D 00 BSR $E0FA
E0FA 39 RTS

Temporisation de 19460µs
E0FB 10 8E 04 00 LDY # $0400 Nombre d'itérations
E0FF 20 E4 BRA $E0E5     Temporisation

***** SYSTEME D'EXPLOITATION PHYSIQUE DU RAMDISK *****
---- Teste l'existence du RamDisk
E101 96 74 LDA <$74     Lit registre périphériques
E103 85 02 BITA # $02     Si le RamDisk existe,
E105 26 05 BNE $E10C     passe
E107 32 62 LEAS $02,S   Rétablit la pile
E109 7E E3 05 JMP $E305 Erreur "Lecteur non prêt"

---- Teste le numéro de piste
E10C 96 4B LDA <$4B     Lit le numéro de piste
E10E 81 10 CMPA # $10     Si le numéro de piste se
E110 25 04 BLO $E116     trouve dans le champ
E112 81 20 CMPA # $20     autorisé, passe
E114 23 05 BLS $E11B
E116 32 62 LEAS $02,S   Rétablit la pile
E118 7E E2 7A JMP $E27A Erreur de secteur

```

Contrôleur interne du TO9 (WD2793)

```

---- Calcule les paramètres pour la commutation sur Ram
E11B 84 04 ANDA #S04 Isole bit P2
E11D 34 02 PSHS A pour PRC du PIA
E11F D6 4B LDB <$4B Lit numéro de piste
E121 C4 08 ANDB #S08
E123 58 ASLB Isole et ajuste bit
E124 58 ASLB P6 pour PRC du PIA
E125 58 ASLB
E126 E8 E0 EORB ,S+ Ajoute au bit P2
E128 34 04 PSHS B
E12A F6 E7 C3 LDB $E7C3 Lit valeur PRC du PIA
E12D C4 AF ANDB #SAF Efface bits P6 et P2
E12F E8 E0 EORB ,S+ Ajoute au résultat
E131 B6 E7 C9 LDA $E7C9 Lit valeur DDRB du PIA
E134 8A F8 ORA #SF8 Isole bits de banque
E136 DD 55 STD <$55 Sauve paramètres commutation

---- Calcule le pointeur sur la piste Ram
E138 E8 A0 00 LDY #SA000 Ptr en début de Ram
E13B 96 4B LDA <$4B Lit numéro de piste
E13D 84 03 ANDA #S03 4 pistes par Ram
E13F 48 ASLA
E140 48 ASLA Multiplie
E141 48 ASLA par 16
E142 48 ASLA
E143 30 86 LEAX A,X
E145 C6 FF LDB #SFF Multiplie par 256 et
E147 3D MUL ajoute au pointeur
E148 30 8B LEAX D,X

---- Calcule le pointeur sur le secteur Ram
E14A 96 4C LDA <$4C Lit numéro de secteur
E14C 4A DECA Ajuste numéro de secteur
E14D 30 86 LEAX A,X
E14F C6 FF LDB #SFF Multiplie par 256 et
E151 3D MUL ajoute au pointeur
E152 30 8B LEAX D,X

---- Calcule les paramètres pour retour sur Ram actuelle
E154 B6 E7 C9 LDA $E7C9 Récupère bits
E157 84 07 ANDA #S07 DDRB du PIA
E159 F6 E7 C3 LDB $E7C3 récupère PRC du PIA
E15C DD 6C STD <$6C Sauve paramètres commutation
E15E 39 RTS

Lit un secteur du RamDisk
E15F 8D A0 BSR $E101 Calcule ptr sur secteur ram
E161 34 07 PSHS B,A,CC
E163 10 9E 4F LDY <$4F Ptr sur buffer secteur
E166 34 20 PSHS Y Empile ptr destination
E168 20 12 BRA $E17C Copie le secteur

Ecrit un secteur du RamDisk
E16A 8D 95 BSR $E101 Calcule ptr sur secteur ram
E16C 34 07 PSHS B,A,CC
E16E 34 10 PSHS X Empile ptr destination
E170 DC 55 LDD <$55
E172 9E 6C LDY <$6C Echange paramètres
E174 1E 01 EXG D,X Ram source et
E176 DD 55 STD <$55 destination
E178 9F 6C STX <$6C
E17A 9E 4F LDY <$4F Lit ptr dans DKBUF

Module de copie de secteur Ram
E17C 1A 50 ORCC #S50 Gel des interruptions
E17E 9F 6E STX <$6E Sauve ptr de lecture
E180 86 09 LDA #S09 8 (+1) blocs
E182 34 02 PSHS A de 32 octets
E184 6A E4 DEC ,S Si secteur entièrement
E186 27 2F BEQ $E1B7 écrit, sort

---- Lit le buffer
E188 9E 6E LDY <$6E Récupère ptr de lecture
E18A DC 55 LDD <$55 Lit params Ram source
E18C 8D 2D BSR $E1BB Commute sur Ram source
E18E 86 08 LDA #S08
E190 EE 81 LDU ,X++
E192 10 AE 81 LDY ,X++ Lit et empile
E195 34 60 PSHS U,Y 32 octets
E197 4A DECA
E198 26 F6 BNE $E190
E19A 9F 6E STX <$6E Sauve ptr de lecture

---- Ecrit le buffer
E19C AE E8 21 LDY $21,S Décale le ptr
E19F 30 88 20 LEAX $20,X destination de
E1A2 AF E8 21 STX $21,S 32 octets
E1A5 DC 6C LDD <$6C Lit params Ram destination
E1A7 8D 12 BSR $E1BB Commute sur Ram destination
E1A9 86 08 LDA #S08
E1AB 35 60 PULS Y,U
E1AD 10 AF 83 STY ,--X Dépile et écrit
E1B0 EF 83 STU ,--X 32 octets
E1B2 4A DECA
E1B3 26 F6 BNE $E1AB
E1B5 20 CD BRA $E184 Pour 8 séries de 32 octets

---- Sortie du programme
E1B7 32 63 LEAS $03,S Rétablit la pile
E1B9 35 07 PULS CC,A,B

Commutation de rams

```

```

E1BB B7 E7 C9 STA $E7C9 Initialise DDRB du PIA
E1BE F7 E7 C3 STB $E7C3 Initialise PRC du PIA
E1C1 4F CLRA Pas d'erreur dans CC
E1C2 39 RTS

***** SYSTEME D'EXPLOITATION PHYSIQUE FLOPPY (SUITE) *****

Routine d'écriture d'un secteur
E1C3 96 49 LDA <$49 Lit numéro de lecteur
E1C5 81 04 CMPA #S04 Si RamDisk, écrit
E1C7 27 A1 BEQ $E16A secteur RamDisk
E1C9 8D 54 BSR $E21F Teste protection disquette
E1CB 25 6C BCS $E239 Si erreur, sort
E1CD 17 00 B6 LBSR $E286 Déplace les têtes
E1D0 25 67 BCS $E239 Si erreur, sort
E1D2 C6 A8 LDB #SA8 Code "Write Sector"
E1D4 34 61 PSHS U,Y,CC
E1D6 31 03 LEAY $03,X
E1D8 8D 60 BSR $E23A
E1DA 20 02 BRA $E1DE Effectue l'opération disque
E1DC A7 A4 STA ,Y Lance la sauvegarde
E1DE A6 C0 LDA ,U+ Sauve le caractère
E1E0 E6 84 LDB ,X lit un caractère
E1E2 C5 02 BITB #S02 Lit status disque
E1E4 26 F6 BNE $E1DC Si fin de transmission,
E1E6 C5 01 BITB #S01 sauve le caractère
E1E8 26 F6 BNE $E1E0 Si pas fin du secteur,
E1EA 35 61 PULS CC,Y,U caractère suivant
E1EC BD E2 6C JSR $E26C Teste si erreur
E1EF 25 48 BCS $E239 Si erreur, sort
E1F1 0D 48 TST <$48 Si pas de vérification
E1F3 2A 43 BPL $E238 demandée, sort

---- Vérification d'un secteur
E1F5 C6 88 LDB #S88 Code "Read Sector"
E1F7 34 41 PSHS U,CC
E1F9 8D 3F BSR $E23A Effectue l'opération disque
E1FB 0F 4E CLR <$4E Efface code d'erreur
E1FD E6 84 LDB ,X Lit status disque
E1FF C5 02 BITB #S02 Si pas fin de transmission,
E201 27 0E BEQ $E211 passe
E203 A6 03 LDA $03,X Charge le caractère
E205 A1 C0 CMPA ,U+ Si caractère correct,
E207 27 F4 BEQ $E1FD comparaison suivante
E209 0C 4E INC <$4E Code d'erreur <> de 0
E20B E6 84 LDB ,X Lit status disque
E20D C5 01 BITB #S01 Attend la fin
E20F 26 FA BNE $E20B du secteur
E211 C5 01 BITB #S01 Si pas fin de secteur,
E213 26 E8 BNE $E1FD caractère suivant
E215 35 41 PULS CC,U
E217 0D 4E TST <$4E Si pas d'erreur,
E219 27 51 BEQ $E26C teste et sort

---- Fixe l'erreur de vérification
E21B 86 20 LDA #S20 Code "Erreur de vérification"
E21D 20 63 BRA $E282 Fixe l'erreur et sort

Teste la protection disquette
E21F 34 04 PSHS B
E221 C6 A8 LDB #SA8 Code "Write Sector"
E223 BD E0 ED JSR $E0ED Lance l'opération lecteur
E226 BD E0 F0 JSR $E0F0 Temporalisation
E229 A6 84 LDA ,X Lit le status
E22B BD E0 EB JSR $E0EB Lance l'interruption lecteur
E22E 35 04 PULS B
E230 85 40 BITA #S40 Si disquette non protégée,
E232 27 04 BEQ $E238 sort sans erreur
E234 86 01 LDA #S01 Code "disquette protégée"
E236 20 4A BRA $E282 Fixe l'erreur et sort

---- Sortie si disquette non protégée en écriture
E238 4F CLRA Pas d'erreur dans CC
E239 39 RTS

Effectue l'opération disque
E23A 96 4C LDA <$4C Initialise registre
E23C A7 02 STA $02,X disque pour le secteur
E23E 1A 50 ORCC #S50 Gel des interruptions
E240 BD E0 ED JSR $E0ED Lance l'opération lecteur
E243 DE 4F LDU <$4F Ptr sur buffer de secteur
E245 39 RTS

Lit un secteur
E246 96 49 LDA <$49 Lit numéro de lecteur
E248 81 04 CMPA #S04 Si lecteur RamDisk,
E24A 10 27 FF 11 LBEQ $E15F lit secteur RamDisk
E24E 8D 36 BSR $E286 Positionne tête sur piste
E250 25 E7 BCS $E239 Si erreur, sort
E252 C6 88 LDB #S88 Code "Read Sector"
E254 34 41 PSHS U,CC
E256 8D E2 BSR $E23A Effectue l'opération disque
E258 E6 84 LDB ,X Lit status disque
E25A C5 02 BITB #S02 Si caractère pas arrivé,
E25C 27 06 BEQ $E264 passe
E25E A6 03 LDA $03,X Lit le caractère

```

Contrôleur interne du TO9 (WD2793)

```

E260 A7 C0 STA ,U+ Mémorise le caractère lu
E262 20 F4 BRA $E258 Caractère suivant
E264 C5 01 BITB $S01 Si lecteur non libéré,
E266 26 F0 BNE $E258 boucle
E268 35 41 PULS CC,U Restitue les interruptions
E26A 20 00 BRA $E26C Pourquoi pas...

Teste si erreur d'opération disque
E26C C5 04 BITB $S04 Si LostData à 1,
E26E 26 10 BNE $E280 erreur de piste
E270 C5 08 BITB $S08 Si CRCError à 1,
E272 26 09 BNE $E27D erreur sur les données
E274 C5 10 BITB $S10 Si RecordNotFound à 1,
E276 26 02 BNE $E27A erreur de secteur
E278 4F CLR A Pas d'erreur dans CC
E279 39 RTS

Fixe l'erreur de secteur
E27A 86 04 LDA $S04 Code d'erreur de secteur
E27C 7D FCB $7D >TST $8608
---- Fixe l'erreur sur les données
E27D 86 08 LDA $S08 Code d'erreur de données
E27F 7D FCB $7D >TST $8602
---- Fixe l'erreur de piste
E280 86 02 LDA $S02 Code d'erreur de piste
E282 97 4E STA <$4E Fixe code d'erreur
E284 43 COMA Erreur dans CC
E285 39 RTS

Positionne la tête sur la piste
E286 17 00 A1 LBSR $E32A Ptr sur position de tête
E289 EC A4 LDD ,Y Lit position de tête
E28B 84 80 ANDA $S80 Si tête déjà déplacée,
E28D 27 04 BEQ $E293 passe
E28F 8D 79 BSR $E30A Teste si bonne piste lecteur
E291 25 0D BCS $E2A0 Si erreur, sort
E293 D1 4B CMPB <$4B Si sur piste demandée,
E295 27 03 BEQ $E29A sort
E297 BD E3 17 JSR $E317 Déplace la tête du lecteur
E29A 4F CLR A Sur 16 bits
E29B E7 01 STB $01,X Fixe piste actuelle
E29D ED A4 STD ,Y Fixe position de tête
E29F 39 RTS

Initialise le positionnement des têtes
E2A0 CC 80 00 LDD $S8000 Flag de reset des registres
E2A3 DD 51 STD <$51 Initialise la position
E2A5 DD 53 STD <$53 des têtes
E2A7 DD 57 STD <$57 Initialise reset contrôleur
E2A9 39 RTS

Détection de présence du disque
E2AA A6 01 LDA $01,X Copie position actuelle
E2AC A7 03 STA $03,X dans position demandée
E2AE C6 10 LDB $S10 Code "Seek"
E2B0 BD E0 ED JSR $E0ED Lance l'opération lecteur
E2B3 BD E0 EB JSR $E0EB Lance l'interruption lecteur

---- Détection de présence du lecteur
E2B6 A6 84 LDA ,X Lit le status
E2B8 84 02 ANDA $S02 Mémoire l'état du bit
E2BA 97 58 STA <$58 de détection de l'index
E2BC 10 8E 82 00 LDY $S8200 Compteur pour une piste
E2C0 31 3F LEAY -$01,Y Si fin de décomptage,
E2C2 27 3B BEQ $E2FF erreur "lecteur non prêt"
E2C4 A6 84 LDA ,X Tant que le bit
E2C6 84 02 ANDA $S02 de détection d'index
E2C8 91 58 CMPA <$58 n'a pas changé d'état,
E2CA 27 F4 BEQ $E2C0 boucle sur le décomptage

---- Détection de présence de la disquette
E2CC C6 07 LDB $S07 7 tentatives maximum
E2CE 34 01 PSHS CC Protège bits d'interruptions
E2D0 1A 50 ORCC $S50 Gel des interruptions
E2D2 A6 84 LDA ,X Lit le status
E2D4 84 02 ANDA $S02 Attend le passage
E2D6 27 FA BEQ $E2D2 de l'index
E2D8 10 8E 09 C4 LDY $S09C4
E2DC 31 3F LEAY -$01,Y Temporisation de 20ms
E2DE 26 FC BNE $E2DC
E2E0 A6 84 LDA ,X Lit le status
E2E2 31 21 LEAY $01,Y Augmente le compteur
E2E4 84 02 ANDA $S02 Boucle jusqu'au prochain
E2E6 27 F8 BEQ $E2E0 passage de l'index
E2E8 35 01 PULS CC Rétablit les interruptions
E2EA 10 8C 31 1B CMPY $S311B Si le passage d'index
E2EE 2B 0C BMI $E2FC suivant n'a pas été détecté
E2F0 10 8C 33 57 CMPY $S3357 entre 178 et 186 ms,
E2F4 2A 06 BPL $E2FC tentative suivante
E2F6 03 58 COM <$58 Commute bit "NotReady"
E2F8 4F CLR A Pas d'erreur dans CC
E2F9 16 FD E5 LBRA $E0E1 Temporisation de 234000µs

---- Sortie si erreur
E2FC 5A DECB Si pas dernière tentative,
E2FD 26 CF BNE $E2CE tentative suivante
E2FF A6 08 LDA $08,X
E301 84 80 ANDA $S80 Conserve bit de densité
E303 A7 08 STA $08,X

```

```

---- Fixe l'erreur de lecteur
E305 86 10 LDA $S10 Code "Lecteur non prêt"
E307 16 FF 78 LBRA $E282 Fixe l'erreur et sort

Lecture du numéro de piste sur le disque
E30A C6 C0 LDB $S00 Code "Read Address"
E30C 8D 27 BSR $E335 Lance l'opération
E30E 17 FF 5F LBSR $E270 Teste si erreur
E311 25 03 BCS $E316 Si erreur, sort
E313 E6 02 LDB $02,X Récupère numéro de piste
E315 4F CLR A Pas d'erreur dans CC
E316 39 RTS

Déplace la tête du lecteur
E317 E7 01 STB $01,X Fixe position actuelle
E319 D6 4B LDB <$4B Fixe position
E31B E7 03 STB $03,X de destination
E31D C6 10 LDB $S10 Code "Seek"
E31F 8D 14 BSR $E335 Lance l'opération
E321 34 20 PSHS Y
E323 BD E0 FB JSR $E0FB Temporisation
E326 E6 01 LDB $01,X Lit position actuelle
E328 35 A0 PULS Y,PC

Pointe sur position de tête
E32A 10 8E 60 51 LDY $S6051 Ptr sur positions de têtes
E32E 96 49 LDA <$49 Lit numéro de lecteur
E330 84 02 ANDA $S02 Isole offset
E332 31 A6 LEAY A,Y Positionne sur bon registre
E334 39 RTS

Effectue une opération lecteur
E335 BD E0 ED JSR $E0ED Lance l'opération lecteur
E338 E6 84 LDB ,X Lit bit de status
E33A C5 01 BITB $S01 Tant que lecteur occupé,
E33C 26 FA BNE $E338 attend
E33E 39 RTS

Recherche de la piste 0
E33F 17 FF 68 LBSR $E2AA Détection présence disque
E342 25 12 BCS $E356 Si erreur, sort
E344 5F CLR B Code "Restore"
E346 8D EE BSR $E335 Lance l'opération
E348 C5 04 BITB $S04 Si piste 0 atteinte,
E34A 26 02 BNE $E34D sort sans erreur
E34C 27 B8 BEQ $E305 Sinon "Lecteur non prêt"
E34E 17 FD AB LBSR $E0FB Temporisation
E350 8D D8 BSR $E32A Ptr sur position tête
E352 6F A4 CLR ,Y Registre de tête
E354 6F 21 CLR $01,Y en position $0000
E356 39 RTS

Routine de formatage d'une disquette
E357 34 7F PSHS U,Y,X,DP,B,A,CC
E359 BD E0 25 JSR $E025 Initialise registres lecteur
E35C 96 49 LDA <$49 Lit numéro de lecteur
E35E 81 04 CMPA $S04 Si RamDisk, initialise
E360 10 27 00 32 LBEQ $E396 catalogue et sort
E364 8D 3C BSR $E3A2 Initialise opération
E366 8D 43 BSR $E3AB Fixe densité selon logiciel
E368 17 FC C5 LBSR $E030 Sélectionne les bits Lecteur
E36B 17 FF D1 LBSR $E33F Recherche de la piste 0
E36E 25 2B BCS $E39B Si erreur, sort
E370 17 FE AC LBSR $E21F Teste protection disquette
E373 25 26 BCS $E39B Si erreur, sort
E375 0F 4A CLR <$4A Efface registre
E377 0F 4B CLR <$4B piste DKTRK
E379 86 03 LDA $S03 3 tentatives
E37B 97 6C STA <$6C maximum
E37D 17 00 9E LBSR $E41E Formatage de la piste
E380 17 01 26 LBSR $E4A9 Vérification de la piste
E383 24 06 BCC $E38B Si ok, piste suivante
E385 0A 6C DEC <$6C Si encore une tentative,
E387 26 F4 BNE $E37D nouvel essai
E389 20 10 BRA $E39B Sinon, sort avec erreur
E38B A6 01 LDA $01,X Lit piste actuelle
E38D 81 4F CMPA $79 Si piste 79, initialise
E38F 27 05 BEQ $E396 catalogue et sort
E391 17 01 50 LBSR $E4E4 Sinon, déplace la tête
E394 20 E3 BRA $E379 et piste suivante
E396 17 01 5B LBSR $E4F4 Initialise piste catalogue
E399 0F D3 CLR <$D3
E39B 35 02 PULS A Récupère CC
E39D 1E 8A EXG A,CC Restaure CC
E39F 44 LSR A Active bit d'erreur
E3A0 35 FE PULS A,B,DP,X,Y,U,PC

Initialise opération
E3A2 96 48 LDA <$48 Efface la commande
E3A4 84 80 ANDA $S80 en conservant le
E3A6 97 48 STA <$48 bit fixant la densité
E3A8 0F 4E CLR <$4E Efface le registre d'erreur
E3AA 39 RTS

Fixe la densité de formatage selon le type de logiciel
E3AB 34 10 PSHS X

```

Contrôle interne du TO9 (WD2793)

E3AD CE E3 D5 LDU #E3D5 Ptr sur paramètres logiciels
E3B0 8E 00 03 LDY #E0003 Ptr sur 3ème lettre applic.
E3B3 C6 04 LDB #E04 4 caractères à comparer
E3B5 A6 84 LDA ,X Si les caractères sont
E3B7 A1 C0 CMPA ,U+ identiques, décrémente
E3B9 27 09 BEQ #E3C4 les compteurs
E3BB 5C INCB |
E3BC 33 C5 LEAU B,U |
E3BE A6 C4 LDA ,U |
E3C0 26 EE BNE #E3B0 |
E3C2 20 0F BRA #E3D3 Sinon, sort
E3C4 30 1F LEAX -\$01,X |
E3C6 26 03 BNE #E3CB |
E3C8 30 88 1A LEAX \$1A,X |
E3CB 5A DECB |
E3CC 26 E7 BNE #E3B5 |
E3CE A6 C4 LDA ,U |
E3D0 B7 E7 D8 STA #E7D8 |
E3D3 35 90 PULS X,PC |

Paramètres pour repérage de densité des logiciels
E3D5 52 45 47 FCC "REG" "GEREZ VOS FICHES"
E3D8 6E FCB \$6E Checksum de l'application
E3D9 80 FCC \$80 Simple densité
E3DA 43 49 50 FCC "CIP" "PICTOR"
E3DD 4A FCB \$4A Checksum de l'application
E3DE 80 FCC \$80 Simple densité
E3DF 4C 45 4D FCC "LEM" "MELODIA"
E3E2 74 FCB \$74 Checksum de l'application
E3E3 80 FCB \$80 Simple densité
E3E4 4C 4F 50 FCC "LOP" "POLYPHONIA"
E3E7 B5 FCB \$B5 Checksum de l'application
E3E8 00 FCB \$00 Double densité
E3E9 53 45 47 FCC "SEG" "GESTION PRIVEE"
E3EC B6 FCB \$B6 Checksum de l'application
E3ED 00 FCB \$00 Double densité
E3EE 00 FCB \$00 - Fin de liste -

Prépare la table des entrelacements
E3EF 8E 60 D3 LDY #E60D3 Ptr sur table entrelacements
E3F2 C6 10 LDB #E10 16 octets à effacer
E3F4 6F 85 CLR B,X |
E3F6 5A DECB |
E3F7 26 FB BNE #E3F4 |
E3F9 30 88 10 LEAX \$10,X |
E3FC 34 10 PSHS X |
E3FE 8E 60 D3 LDY #E60D3 Ptr table entrelacements
E401 86 01 LDA #E01 Départ avec secteur 1
E403 D6 4D LDB <\$D Lit facteur d'entrelacement
E405 A7 84 STA ,X Ecrit numéro de secteur
E407 30 85 LEAX B,X Décale ptr selon facteur
E409 4C INCA Numéro de secteur + 1
E40A 81 10 CMPA #E10 Si secteur > à
E40C 22 0E BHI #E41C secteur 16, sort
E40E AC E4 CMPX ,S Si pas hors table,
E410 25 02 BLO #E414 recherche secteur suivant
E412 30 10 LEAX -\$10,X Retour de pointeur
E414 6D 84 TST ,X Si place libre,
E416 27 ED BEQ #E405 enregistre secteur
E418 30 01 LEAX \$01,X Sinon, teste
E41A 20 F2 BRA #E40E place suivante
E41C 35 90 PULS X,PC |

Formatage d'une piste
E41E 34 49 PSHS U,DP,CC |
E420 1A 50 ORCC #E50 Gel des interruptions
E422 17 FF CA LBSR #E3EF Crée table entrelacements

Sélection du type de secteur selon la densité
E425 10 8E E5 62 LDY #E562 Liste double densité
E429 7D E7 D8 TST #E7D8 Si simple densité
E42C 2A 04 BPL #E432 sélectionnée,
E42E 10 8E E5 84 LDY #E584 liste simple densité
E432 4F CLRA Pour 16 bits
E433 DE 4F LDU <\$F Ptr sur table de formatage
E435 E6 A0 LDB ,Y+ Lit valeur itération
E437 27 0F BEQ #E448 Si fin de liste, sort
E439 1F 01 TFR D,X Itération sur 16 bits
E43B E6 A0 LDB ,Y+ Ecrit le mot
E43D E7 C0 STB ,U+ à écrire
E43F AF C1 STX ,U++ Ecrit le nombre d'itérations
E441 8E E4 7B LDY #E47B Adresse de retour
E444 AF C1 STX ,U++ pour le PULS PC
E446 20 ED BRA #E435 Série suivante

E448 DE 4F LDU <\$F Ptr sur table de formatage
E44A 96 4B LDA <\$4B Lit numéro de piste
E44C A7 4F STA \$0F,U Ecrit en place de n° de piste
E44E 8E E4 87 LDY #E487 Pose le vecteur pour
E451 AF C8 2B STX \$2B,U n° de secteur suivant
E454 8E E4 92 LDY #E492 Pose le vecteur pour
E457 AF C8 4E STX \$4E,U dernier secteur
E45A 10 8E 60 D4 LDY #E60D4 Ptr sur table entrelacements
E45E 86 E7 LDA #E7 Fixe le DP
E460 1F 8B TFR A,DP à E7
E462 8E 00 10 LDY #E0010 16 mots à écrire
E465 C6 FF LDB #E0010 Mot pour remplissage piste

E467 0D D8 TST <\$D8 Si formatage en double
E469 2B 05 BMI #E470 densité alors
E46B 8E 00 1E LDY #E001E 30 mots à écrire et \$4E
E46E C6 4E LDB #E4E pour mot de remplissage
E470 86 F8 LDA #E78 Code "Write Track"
E472 97 D0 STA <\$D0 initialisé
E474 17 FC 79 LBSR #E0F0 Temporisation
E477 86 02 LDA #E02 Bit pour test de transmission
E479 D7 D3 STB <\$D3 Reset status
E47B 95 D0 BITA <\$D0 Attente de fin de
E47D 27 FC BEQ #E47B l'opération d'écriture
E47F D7 D3 STB <\$D3 Ecrit le mot sur la piste
E481 30 1F LEAX -\$01,X Pour le nombre
E483 26 F6 BNE #E47B de mots convenu
E485 37 94 PULS B,X,PC Saute au programme suivant
---- Mise à jour du numéro de secteur

E487 34 04 PSHS B |
E489 E6 A0 LDB ,Y+ Lit n° dans liste entrelacem
E48B E7 C8 EC STB -\$14,U Ecrit en place n° de secteur
E48E 35 04 PULS B |
E490 20 E9 BRA #E47B Sauve le caractère
---- Remplissage de l'intervalle de piste si dernier secteur
E492 FE 60 4F LDY #E604F Ptr sur table de formatage
E495 6D 3F TST -\$01,Y Si pas dernier secteur,
E497 26 E2 BNE #E47B reste dans la routine
E499 96 D0 LDA <\$D0 Lit status disque
E49B 85 02 BITA #E02 Si pas fin de transmission,
E49D 27 04 BEQ #E4A3 passe
E49F D7 D3 STB <\$D3 Sauve le caractère
E4A1 20 F6 BRA #E499 Caractère suivant
E4A3 85 01 BITA #E01 Si pas fin de piste,
E4A5 26 F2 BNE #E499 caractère suivant

E4A7 35 C9 PULS CC,DP,U,PC
---- Vérification de la piste formatée
E4A9 8E E7 D0 LDY #E7D0 Initialise registre disque
E4AC 0D 48 TST <\$48 Si pas de vérification
E4AE 2C 1C BGE #E4CC demandée, sort
E4B0 86 E5 LDA #E85 Valeur pour remplissage
E4B2 8D 1A BSR #E4CE Remplit un secteur

E4B4 4F CLRA |
E4B5 4C INCA |
E4B6 97 4C STA <\$4C |
E4B8 C6 03 LDB #E03 3 tentatives
E4BA D7 6C STB <\$6C maximum
E4BC 17 FD 36 LBSR #E1F5 Vérifie le secteur
E4BF 24 05 BCC #E4C6 Si ok, secteur suivant
E4C1 0A 6C DEC <\$6C Tentative
E4C3 26 F7 BNE #E4BC suivante
E4C5 39 RTS |
---- Secteur suivant pour vérification
E4C6 96 4C LDA <\$4C Si pas
E4C8 81 10 CMPA #E10 secteur 16,
E4CA 26 E9 BNE #E4B5 secteur suivant
E4CC 4F CLRA Pas d'erreur dans CC
E4CD 39 RTS |

Remplissage du buffer de secteur
E4CE 10 9E 4F LDY <\$4F Pointeur sur secteur
E4D1 D6 49 LDB <\$49 Lit le numéro de lecteur
E4D3 C1 04 CMPB #E04 Si pas RamDisk, remplit
E4D5 26 02 BNE #E4D9 selon densité
E4D7 20 03 BRA #E4DC Sinon, secteur 256 octets
E4D9 F6 E7 D8 LDB #E7D8 Lit la densité lecteur
E4DC C4 80 ANDB #E80 Isole bit de densité
E4DE 17 A0 STA ,Y+ Remplit le secteur avec la
E4E0 5A DECB valeur convenue selon
E4E1 26 FB BNE #E4DE la densité
E4E3 39 RTS |

E4E4 C6 50 LDB #E50 Déplace la tête d'une piste vers l'intérieur pour le formatage
E4E6 17 FE 4C LBSR #E335 Code "Step-In"
E4E9 BD E0 FB JSR #E0FB Lance l'opération
E4EC 17 FE 3B LBSR #E32A Temporisation
E4EF 6C 21 INC #E01,Y Pointe n° de piste actuelle
E4F1 0C 4B INC <\$4B Piste disque + 1
E4F3 39 RTS Piste moniteur + 1

Initialisation logique de la piste du catalogue
---- Efface tous les secteurs de la piste
E4F4 8E 00 14 LDY #E0014 Piste 20
E4F7 9F 4A STY <\$4A demandée
E4F9 86 FF LDA #E7F Valeur pour remplissage
E4FB 8D D1 BSR #E4CE Remplit le buffer de secteur
E4FD 86 08 LDA #E08 Programme écriture du
E4FF 9A 48 ORA <\$48 secteur avec vérification
E501 97 48 STA <\$48 si demandée
E503 5F CLRB |
E504 5C INCB |
E505 D7 4C STB <\$4C Secteur
E507 17 FB 44 LBSR #E04E 1
E50A 25 4D BCS #E559 demandé
E50C C1 10 CMPB #E10 Sauve le secteur
E50E 26 F4 BNE #E504 Si erreur, sort
---- Initialise la FAT
E510 86 01 LDA #E01 Pour les 16 secteurs
de la piste Positionne sur

Contrôleur interne du TO9 (WD2793)

```

E512 97 4C STA <$4C | secteur 1
E514 9E 4F LDX <$4F | Pointeur sur secteur
E516 6F 84 CLR X | Premier octet de FAT à 0
E518 CC FE FE LDD #SFEFE | Réserve les blocs
E51B ED 88 29 STD $29,X | de la piste catalogue
---- Initialise éventuellement le secteur du nom pour RamDisk
E51E 96 49 LDA <$49 | Lit numéro de lecteur
E520 81 04 CMPA #S04 | Si pas RamDisk, sauve le
E522 26 2D BNE $E551 | secteur et sort
E524 34 10 PSHS X |
E526 8E E5 5A LDX #E55A | Fixe pointeur de secteur
E529 9F 4F STX <$4F | sur le nom "RAM DISK"
E52B BD E0 4E JSR $E04E | Sauve le secteur
---- Réserve les blocs de FAT pour le RamDisk (RamDisk de 128k)
E52E 35 10 PULS X | Réinitialise le
E530 9F 4F STX <$4F | pointeur de FAT
E532 86 10 LDA #S10 | 64 blocs à réserver
E534 34 12 PSHS X,A |
E536 30 01 LEAX $01,X | Positionne au début de FAT
E538 CC FE FE LDD #SFEFE | Valeur pour réservation
E53B ED 88 40 STD $40,X |
E53E ED 88 60 STD $60,X |
E541 ED 89 00 80 STD $0080,X | Réserve les blocs
E545 ED 89 00 A0 STD $00A0,X |
E549 ED 81 STD ,X++ |
E54B 6A E4 DEC ,S | Pour tous
E54D 26 EC BNE $E53B | les blocs
E54F 35 12 PULS A,X |
---- Sauve le secteur de FAT
E551 0C 4C LNC <$4C | Secteur 2 demandé
E553 17 FA F8 LBSR $E04E | Sauve le secteur
E556 25 01 BCS $E559 | Si erreur, sort
E558 4F CLRA | Sort sans erreur
E559 39 RTS |

```

```

Nom officiel du Ram Disk
E55A 52 41 4D 20 44 FCC "RAM DISK"
E55F 49 53 4B

```

```

Table pour la fabrication d'un secteur double densité pour formatage
(1 octet d'itérations / 1 octet pour la valeur)
E562 0C 00 03 F5 FCB $0C,$00,$03,$F5
E566 01 FE 01 00 FCB $01,$FE,$01,$00
E56A 01 00 01 01 FCB $01,$00,$01,$01
E56E 01 01 01 F7 FCB $01,$01,$01,$F7
E572 16 4E 0C 00 FCB $16,$4E,$0C,$00
E576 03 F5 01 FB FCB $03,$F5,$01,$FB
E57A 80 E5 80 E5 FCB $80,$E5,$80,$E5
E57E 01 F7 2C 4E FCB $01,$F7,$2C,$4E
E582 00 00 FCB $00,$00 - Fin de la liste -

```

```

Table pour la fabrication d'un secteur simple densité pour formatage
(1 octet d'itérations / 1 octet pour la valeur)
E584 03 00 03 00 FCB $03,$00,$03,$00
E588 01 FE 01 00 FCB $01,$FE,$01,$00
E58C 01 00 01 01 FCB $01,$00,$01,$01
E590 01 00 01 F7 FCB $01,$00,$01,$F7
E594 0C FF 03 00 FCB $0C,$FF,$03,$00
E598 03 00 01 FB FCB $03,$00,$01,$FB
E59C 40 E5 40 E5 FCB $40,$E5,$40,$E5
E5A0 01 F7 16 FF FCB $01,$F7,$16,$FF
E5A4 00 00 FCB $00,$00 - Fin de la liste -

```

***** SYSTEME D'EXPLOITATION LOGIQUE *****

```

Nom de fichier temporaire pour la sauvegarde
E5A6 53 43 52 41 FCC "SCRATCH DOS"
E5AA 54 43 48 20
E5AE 44 4F 53

```

```

Clôture d'écriture
E5B1 D6 F0 LDB <$F0 | Sauvegarde sans
E5B3 C1 02 CMPB #S02 | écrasement ?
E5B5 27 25 BEQ $E5DC | Oui, écrit FAT seule
E5B7 0A F0 DEC <$F0 | Passe en sauveg. sans escr.
E5B9 BD E6 98 JSR $E698 | Cherche fichier courant
E5BC 25 31 BCS $E5EF | Si erreur disque, sort
E5BE 5D TSTB | Fichier existe déjà?
E5BF 27 05 BEQ $E5C6 | Non, pas d'effacement
E5C1 17 00 B6 LBSR $E67A | Effacement du fichier
E5C4 25 29 BCS $E5EF | Si erreur, sort
E5C6 0C F0 INC <$F0 | Retour en sauveg avec écrasmt
E5C8 BD E6 98 JSR $E698 | Cherche fichier "SCRATCH.DOS"
E5CB 25 22 BCS $E5EF | Si erreur, sort
E5CD C6 0A LDB #S0A |
E5CF 9E E7 LDX <$E7 | Recopie le nom
E5D1 A6 85 LDA B,X | du fichier
E5D3 A7 A5 STA B,Y | courant à la place
E5D5 5A DECB | de "SCRATCH.DOS"
E5D6 2C F9 BGE $E5D1 |
E5D8 8D 7F BSR $E659 | Ecrit le secteur catalogue
E5DA 25 13 BCS $E5EF | Si erreur de disque, sort
----
E5DC 86 02 LDA #S02 |

```

```

E5DE 97 4C STA <$4C | Programme la
E5E0 C6 14 LDB #S14 | sauvegarde
E5E2 4F CLRA | de la FAT
E5E3 DD 4A STD <$4A |
E5E5 8D 56 BSR $E63D | Rétablissement de FAT swapée
E5E7 9F 4F STX <$4F | Sauvegarde
E5E9 8D 6E BSR $E659 | de la FAT
E5EB 25 02 BCS $E5EF | Si erreur de disque, sort
E5ED 0F F0 CLR <$F0 | Clôture du fichier effectuée
E5EF 39 RTS |

```

```

Swap de la FAT
E5F0 34 07 PSHS B,A,CC |
E5F2 9E ED LDX <$ED | Pointeur
E5F4 30 89 00 80 LEAX $0080,X | en milieu de FAT
E5F8 A6 1F LDA -$01,X | Echange l'octet précédent
E5FA E6 88 7F LDB $7F,X | avec le 127ème octet
E5FD A7 88 7F STA $7F,X |
E600 E7 82 STB ,X | Déplace pointeur à rebours
E602 9C ED CMPX <$ED | Fin de FAT ?
E604 26 F2 BNE $E5F8 | Non, échange suivant
E606 35 87 PULS CC,A,B,PC |

```

```

Charge la FAT
E608 DC ED LDD <$ED | Pointeur sur FAT swapée
E60A DD 25 TST <$25 | FAT système initialisée ?
E60C 26 02 BNE $E610 | Non, initialise avec
E60E DD 25 STD <$25 | pointeur sur FAT swapée
E610 93 25 SUBD <$25 | Si la différence entre les
E612 C1 80 CMPB #S80 | deux est de 128 octets,
E614 27 03 BEQ $E619 | prend le pointeur système
E616 9E ED LDX <$ED |
E618 7D FCB $7D >TST $9E25 |
E619 9E 25 LDX <$25 | Initialise les pointeurs
E61B 9F ED STX <$ED |
E61D 9F 4F STX <$4F |
E61F 86 02 LDA #S02 |
E621 8D 3C BSR $E65F | Chargement
E623 25 CA BCS $E5EF | de la FAT
E625 8D 0B BSR $E632 | Si erreur, sort
E627 2B 08 BMI $E631 | Contrôleur
E629 8D C5 BSR $E5F0 | double densité ?
E62B 30 89 00 80 LEAX $0080,X | Oui, swap
E62F 9F ED STX <$ED | de la FAT
E631 39 RTS | et pointeur courant
| en milieu de FAT

```

```

Teste la densité du lecteur
E632 34 05 PSHS B,CC |
E634 BD E0 EB JSR $E0EB | Lance l'interruption lecteur
E637 35 05 PULS CC,B |
E639 7D E7 D8 TST $E7D8 | Teste densité lecteur
E63C 39 RTS |

```

```

Rétablit la FAT swapée
E63D 9E ED LDX <$ED | S'il s'agit
E63F 8D F1 BSR $E632 | d'une FAT
E641 2B EE BMI $E631 | double densité,
E643 30 88 80 LEAX -$80,X | repositionne le pointeur
E646 8D E7 BSR $E62F | de FAT en début de secteur
E648 20 A6 BRA $E5F0 | et rétablit la FAT swapée

```

```

Charge le premier secteur de catalogue
E64A 86 03 LDA #S03 | N° du ler secteur catalogue
E64C 9E E9 LDX <$E9 | Pointeur sur buffer
E64E 9F 4F STX <$4F | catalogue initialisé
E650 8D 0D BSR $E65F | Chargement du secteur
E652 25 E9 BCS $E63D | Si erreur, rétablit la FAT
E654 39 RTS | Sort sans erreur

```

```

Charge un secteur
E655 8D 0F BSR $E666 | Charge le secteur
E657 20 F9 BRA $E652 | Si erreur, rétablit la FAT

```

```

Sauve un secteur
E659 86 08 LDA #S08 | Sauve le
E65B 8D 0B BSR $E668 | secteur
E65D 20 F3 BRA $E652 | Si erreur, rétablit la FAT

```

```

Charge un secteur de la piste 20
E65F 97 4C STA <$4C | Sélection du secteur
E661 C6 14 LDB #S14 | Sélection
E663 4F CLRA | de la piste
E664 DD 4A STD <$4A | 20
E666 86 02 LDA #S02 | Mode lecture sélectionné
E668 97 48 STA <$48 | Type d'opération programmé
E66A 10 9E E9 LDY <$E9 | Ptr buffer catalogue dans Y
E66D 17 F9 94 LBSR $E004 | Opération disquette
E670 86 03 LDA #S03 | Code d'erreur "entrée-sortie"
E672 39 RTS | Sort avec ou sans erreur

```

```

Sortie si erreur
E673 8D C8 BSR $E63D | Rétablit la FAT si swapée,
---- Fixe l'erreur
E675 97 E5 STA <$E5 | stocke erreur système,
E677 43 COMA | erreur dans CC
E678 43 COMA | et rétablit code d'erreur

```


Contrôleur externe CD90-351 version A (THMFC1) - 1ère banque

```

E409 CC 1F 22   LDD   #&1F22   Sinon, séparateur inactif
E40C A7 07     STA   $07,X   Fixe séparateur
E40E E7 01     STB   $01,X   Fixe 256 mots par secteur
E410 39               RTS

    Fixe la précompensation pour simple densité
E411 86 BF     LDA   #&BF   Séparateur actif à $3F
E413 A7 07     STA   $07,X   Fixe séparateur
E415 6F 01     CLR   $01,X   Fixe 128 mots par secteur
E417 39               RTS

    Calcul automatique du CRC identificateur en simple densité
E418 DE 4F     LDU   <$4F   Ptr buffer secteur
E41A 10 8E 00 80 LDY   #&0080  128 caractères à traiter
E41E CC BF 84   LDD   #&BF84  Valeur CRC de départ
---- Calcul d'un CRC
E421 34 06     PSHS  B,A     Empile le CRC
E423 A6 E4     LDA   ,S     Lit le CRCHigh
E425 A8 C0     EORA  ,U+    Masque les bits sortants
E427 34 02     PSHS  A
E429 44         LSR   A
E42A 44         LSR   A
E42B 44         LSR   A
E42C 44         LSR   A
E42D A8 E4     EORA  ,S
E42F A7 E4     STA   ,S
E431 1F 89     TFR   A,B     Masque dans B
E433 54         LSR   B
E434 54         LSR   B
E435 54         LSR   B
E436 48         ASLA  A
E437 48         ASLA  A
E438 48         ASLA  A
E439 48         ASLA  A
E43A E8 62     ORB   $02,S
E43C E7 62     STB   $02,S
E43E 1F 89     TFR   A,B
E440 A8 62     EORA  $02,S
E442 A7 61     STA   $01,S
E444 58         ASLB  B
E445 E8 E4     ORB   ,S
E447 E7 62     STB   $02,S
E449 32 61     LEAS  $01,S
E44B 31 3F     LEAY  -$01,Y
E44D 26 D4     BNE  $E423
E44F 35 06     PULS  A,B
E451 39               RTS

    Routine de lecture d'un secteur
E452 17 00 AF   LBSR  $E504   Recherche la piste
E455 10 25 00 98 LBSCS $E4F1   Si erreur, sort
E459 34 01     PSHS  CC   Protège CC
E45B 96 58     LDA   <$58   Lit densité lecteur
E45D 26 43     BNE  $E4A2   Si simple densité, traite
---- Lecture d'un secteur en double densité
E45F 17 FF 9E   LBSR  $E400   Fixe la précompensation
E462 C6 1B     LDB   $1B     Lecture automatique
E464 17 FF 2F   LBSR  $E396   Initialise registres disque
---- Recherche le secteur
E467 86 04     LDA   #&04   Compteur recherche marquage
E469 10 8E 75 00 LDY   #&7500  Compteur pour une piste
E46D 31 3F     LEAY  -$01,Y   Si fin de décomptage,
E46F 10 27 00 7F LBQEQ $E4F2   erreur de secteur
E473 E6 84     LDB   ,X     Lit status opération
E475 C5 02     BITB  $02     Si secteur non trouvé,
E477 27 F4     BEQ   $E46D  boucle
---- Recherche du marquage de données
E479 E6 03     LDB   $03,X   Reset du status
E47B 4A       DECA  $03,X   Si fin de décomptage,
E47C 10 27 00 7A LBQEQ $E4FA   erreur de piste
E480 E6 84     LDB   ,X     Attend fin de
E482 2A FC     BPL   $E480  transmission
E484 E6 03     LDB   $03,X   Lit le caractère
E486 C1 FB     CMPB  #&FB   Si pas marquage de données,
E488 26 F1     BNE  $E47B  boucle
---- Lecture du secteur double densité
E48A 4F       CLRA  A     256 caractères à lire
E48B E6 84     LDB   ,X     Attend fin de
E48D 2A FC     BPL   $E48B  transmission
E48F E6 03     LDB   $03,X   Charge le caractère
E491 E7 C0     STB   ,U+    Ecrit le caractère
E493 4A       DECA  A     Pour tout
E494 26 F5     BNE  $E48B  le secteur
---- Test de validité de la lecture
E496 E6 84     LDB   ,X     Lit status opération
E498 C5 08     BITB  #&08   Si pas fin opération,
E49A 27 FA     BEQ   $E496  boucle
E49C C5 04     BITB  #&04   Si CRC incorrect,
E49E 26 56     BNE  $E4F6  erreur sur les données
E4A0 20 4B     BRA   $E4ED  Sort du programme

    Lecture d'un secteur en simple densité
E4A2 17 FF 6C   LBSR  $E411   Fixe la précompensation
E4A5 17 FF 01   LBSR  $E3A9   Recherche le secteur
E4A8 25 48     BCS   $E4F2   Si erreur, sort
E4AA 31 23     LEAY  $03,Y   Ajuste compteur tempo

```

```

E44C 17 FD 6E   LBSR  $E21D   Temporisation
---- Recherche du début des données
E44F C6 FB     LDB   #&FB   Fixe code de synchro
E4B1 E7 03     STB   $03,X   à trouver
E4B3 86 80     LDA   #&80   128 caractères à lire
E4B5 C6 38     LDB   #&38   Passe en mode
E4B7 E7 84     STB   ,X     "lecture synchro"
E4B9 10 8E 01 00 LDY   #&0100  Compteur pour recherche
E4BD 31 3F     LEAY  -$01,Y   Si fin du décomptage,
E4BF 27 39     BEQ   $E4FA   erreur de piste
E4C1 E6 84     LDB   ,X     Lit status opération
E4C3 C5 01     BITB  #&01   Si pas de synchro,
E4C5 27 F6     BEQ   $E4BD  boucle
---- Lecture du secteur simple densité
E4C7 E6 03     LDB   $03,X   Reset du status
E4C9 C6 28     LDB   #&28   Passage en mode
E4CB E7 84     STB   ,X     "lecture données"
E4CD E6 84     LDB   ,X     Attend fin de
E4CF 2A FC     BPL   $E4CD  transmission
E4D1 E6 03     LDB   $03,X   Charge le caractère
E4D3 E7 C0     STB   ,U+    Ecrit le caractère
E4D5 4A       DECA  A     Pour tout
E4D6 26 F5     BNE  $E4CD  le secteur
---- Vérification de la validité de la lecture
E4D8 E6 84     LDB   ,X     Attend fin de
E4DA 2A FC     BPL   $E4D8  transmission
E4DC E6 03     LDB   $03,X   Lit le CRCHigh/CRCLow
E4DE 34 04     PSHS  B     Empile le CRCHigh/CRCLow
E4E0 43       COMA  A     Si pas deuxième passe,
E4E1 26 F5     BNE  $E4D8  boucle
E4E3 17 FF 32   LBSR  $E418   Calcule CRC pour les données
E4E6 1E 89     EXG   A,B     Ajuste CRC pour comparaison
E4E8 10 A3 E1   CMPD  ,S++    Si CRC incorrect,
E4EB 26 09     BNE  $E4F6   erreur sur les données
E4ED 35 01     PULS  CC     Rétablit CC
E4EF 6F 84     CLR   ,X     Reset lecteur
E4F1 39               RTS

---- Fixe l'erreur de secteur
E4F2 86 04     LDA   #&04   Code "Erreur de secteur"
E4F4 20 06     BRA   $E4FC   Fixe l'erreur
---- Fixe l'erreur sur les données
E4F6 86 08     LDA   #&08   Code "Erreur sur les données"
E4F8 20 02     BRA   $E4FC   Fixe l'erreur
---- Fixe l'erreur de piste
E4FA 86 02     LDA   #&02   Code "Erreur de piste"
---- Fixe l'erreur
E4FC 97 4E     STA   <$4E   Fixe l'erreur moniteur
E4FE 6F 84     CLR   ,X     Reset du lecteur
E500 35 01     PULS  CC     Rétablit le CC
E502 43       COMA  A     Erreur dans CC
E503 39               RTS

    Routine de recherche d'une piste
E504 8D 06     BSR   $E50C   Activation du lecteur
E506 25 03     BCS   $E50B   Si erreur, sort
E508 8D 22     BSR   $E52C   Déplace les têtes
E50A 4F       CLRA  A     Pas d'erreur dans CC
E50B 39               RTS

    Activation du lecteur
E50C 10 8E 80 00 LDY   #&8000  Compteur de test
E510 D6 57     LDB   <$57   Récupère bits de lecteur
E512 CA 04     ORB   #&04
E514 E7 02     STB   $02,X   Stimulation
E516 C4 FB     ANDB  #&FB   du lecteur
E518 E7 02     STB   $02,X
E51A 31 3F     LEAY  -$01,Y   Si fin de décomptage,
E51C 27 08     BEQ   $E526   "Lecteur non prêt"
E51E E6 01     LDB   $01,X   Lit status disque
E520 C4 02     ANDB  #&02   Si lecteur non disponible,
E522 27 EC     BEQ   $E510  test suivant
E524 4F       CLRA  A     Pas d'erreur dans CC
E525 39               RTS

    Fixe l'erreur de lecteur
E526 86 10     LDA   #&10   Code "Lecteur non prêt"
E528 34 01     PSHS  CC     Ajuste la pile
E52A 20 D0     BRA   $E4FC   Fixe l'erreur

    Déplacement des têtes sur la piste demandée
E52C 17 FC EA   LBSR  $E219   Temporisation
E52F 8D 29     BSR   $E55A   Ptr sur position de tête
E531 E6 A4     LDB   ,Y     Lit position actuelle
E533 96 57     LDA   <$57   Lit bits de lecteur
E535 D0 4B     SUBB  <$4B   Retire position demandée
E537 27 18     BEQ   $E551   Si = 0, sort
E539 2A 03     BPL   $E53E   Si > à 0, recul de la tête et
E53B 8A 20     ORA   #&20   Sinon, avance de la tête et
E53D 50       NEGB  A     valeur absolue intervalle
E53E A7 02     STA   $02,X   Fixe sens de déplacement
E540 8A 14     ORA   #&14
E542 A7 02     STA   $02,X
E544 84 EB     ANDA  #&EB   Stimulation du
E546 A7 02     STA   $02,X   déplacement pas à pas
E548 17 FC CE   LBSR  $E219   Temporisation

```

Contrôleur externe CD90-351 version A (THMFC1) - 1ère banque

E54B	5A	DECB		Pour le nombre de
E54C	26 F2	BNE	\$E540	déplacements convenu
E54E	17 FC C2	LBSR	\$E213	Temporisation
E551	8D 07	BSR	\$E55A	Ptr sur position de tête
E553	D6 4B	LDB	<\$4B	Lit numéro de piste
E555	E7 06	STB	\$06,X	Initialise piste disque
E557	E7 A4	STB	,Y	Initialise position système
E559	39	RTS		
Pointeur sur position de la tête				
E55A	31 8D 7A F4	LEAY	>\$6052,PCR	Ptr positions têtes
E55E	96 49	LDA	<\$49	Lit numéro de lecteur
E560	84 02	ANDA	#\$02	Ajuste offset
E562	31 A6	LEAY	A,Y	Ajuste ptr sur position
E564	39	RTS		

Contrôleur externe CD90-351 version A (THMFC1) - 2ème banque

Identificateur de RomDisk (IDDISK)				Routine de formatage floppy					
E000	4D	FCC	"M"	Contrôleur Thomson	E0A0	34 7F	PSHS	U,Y,X,DP,B,A,CC	
E001	54	FCC	"T"	FAT 160 octets	E0A2	8D 50	BSR	\$E0F4	Initialise registres
E002	44	FCC	"D"	Double densité	E0A4	8D 61	BSR	\$E107	Fixe densité de formatage
E003	3A	FCB	\$3A	Checksum de l'identificateur	E0A6	17 00 A3	LBSR	\$E14C	Choix du lecteur
Indirections standard sur les opérations physiques				---- Formatage de la piste					
E004	16 00 24	LBRA	\$E02B	Fonctions standard	E0A9	25 42	BCS	\$E0ED	Si erreur, sort
E007	16 00 2A	LBRA	\$E034	Lancement du boot	E0AB	17 00 CF	LBSR	\$E17D	Recherche la piste 0
E00A	16 00 93	LBRA	\$E0A0	Formatage	E0AE	25 3B	BCS	\$E0EB	Si erreur, sort
---- Indirections standard sur les opérations logiques				---- Sortie si erreur de formatage					
E00D	16 00 36	LBRA	\$E046	Chargement de la FAT	E0B3	25 36	BCS	\$E0EB	Si erreur, sort
E010	16 00 3C	LBRA	\$E04F	Ouverture d'un fichier	E0B5	34 01	PSHS	CC	Protège flag interruptions
E013	16 00 42	LBRA	\$E058	Effacement d'un fichier	E0B7	1A 50	ORCC	#50	Gel des interruptions
E016	16 00 48	LBRA	\$E061	Ecriture d'un secteur	E0B9	17 01 0A	LBSR	\$E1C6	Prépare table entrelacements
E019	16 00 4E	LBRA	\$E06A	Création d'un fichier	E0BC	86 02	LDA	#502	2 tentatives maximum
E01C	16 00 54	LBRA	\$E073	Allocation d'un bloc	E0BE	34 02	PSHS	A	Crée liste des paramètres
E01F	16 00 5A	LBRA	\$E07C	Initialise opération sur bloc	E0C0	17 01 35	LBSR	\$E1F8	Formate la piste
E022	16 00 60	LBRA	\$E085	Clôture d'écriture	E0C3	17 01 5F	LBSR	\$E225	Si pas d'erreur, suivante
---- Indirections standard sur les opérations physiques				---- Clôt le formatage					
E025	16 FF FD	LBRA	\$E025	- Boucle à vide -	E0C6	24 0B	BCC	\$E0D3	Tentative suivante
E028	16 FF FD	LBRA	\$E028	- Boucle à vide -	E0C8	6A E4	DEC	,S	Rétablit la pile
---- Point d'entrée pour opération standard floppy				---- Piste suivante					
E02B	34 46	PSHS	U,B,A	Point d'entrée traitement	E0CA	26 F4	BNE	\$E0C0	Dégel des interruptions
E02D	33 8C D4	LEAU	<\$E004,PCR	Banque 0 demandée	E0CC	35 02	PULS	A	Rétablit la pile
E030	86 00	LDA	#500	Exécution programme	E0CE	35 01	PULS	CC	Dégel des interruptions
E032	20 5A	BRA	\$E08E		E0D0	43	COMA		Erreur dans CC
---- Point d'entrée pour lancement du boot floppy				---- Clôt le formatage					
E034	34 46	PSHS	U,B,A	Point d'entrée traitement	E0E2	17 03 5E	LBSR	\$E443	Initialise piste catalogue
E036	33 8C CE	LEAU	<\$E007,PCR	Banque 0 demandée	E0E5	25 04	BCS	\$E0EB	Si erreur, sort
E039	86 00	LDA	#500	Exécution programme	E0E7	86 50	LDA	#50	Pose piste 80
E03B	20 51	BRA	\$E08E		E0E9	97 4B	STA	<\$4B	Extinction du moteur
---- Point d'entrée pour formatage floppy				---- Piste suivante					
E03D	34 46	PSHS	U,B,A	Point d'entrée traitement	E0EB	8D 7F	BSR	\$E16C	Récupère CC
E03F	33 8C C8	LEAU	<\$E00A,PCR	Banque 1 demandée	E0ED	35 02	PULS	A	Dégel des interruptions
E042	86 01	LDA	#501	Exécution programme	E0EF	1E 8A	EXG	A,CC	Bit d'erreur dans CC
E044	20 48	BRA	\$E08E		E0F1	44	LSRA		
---- Point d'entrée pour chargement de la FAT floppy				Initialise registres					
E046	34 46	PSHS	U,B,A	Point d'entrée traitement	E0F4	1F 50	TFR	PC,D	PC dans D
E048	33 8C C2	LEAU	<\$E00D,PCR	Banque 2 demandée	E0F6	84 70	ANDA	#70	Fixe le DP
E04B	86 02	LDA	#502	Exécution programme	E0F8	1F 8B	TFR	A,DP	MO=\$20 TO=\$60
E04D	20 3F	BRA	\$E08E		E0FA	96 48	LDA	<\$48	Lit code de commande
---- Point d'entrée pour ouverture d'un fichier floppy				Fixe la densité de formatage selon le type de logiciel					
E04F	34 46	PSHS	U,B,A	Point d'entrée traitement	E0FC	84 80	ANDA	#80	Garde bit de vérification
E051	33 8C BC	LEAU	<\$E010,PCR	Banque 2 demandée	E0FE	97 48	STA	<\$48	Ptr registres disque
E054	86 02	LDA	#502	Exécution programme	E100	30 8D 06 CC	LEAX	>\$E7D0,PCR	Efface code d'erreur
E056	20 36	BRA	\$E08E		E104	0F 4E	CLR	<\$4E	
---- Point d'entrée pour effacement d'un fichier sur floppy				Paramètres pour repérage de densité des logiciels					
E058	34 46	PSHS	U,B,A	Point d'entrée traitement	E132	52 45 47	FCC	"REG"	"GEREZ VOS FICHES"
E05A	33 8C B6	LEAU	<\$E013,PCR	Banque 2 demandée	E135	6E	FCB	\$6E	Checksum de l'application
E05D	86 02	LDA	#502	Exécution programme	E136	FF	FCB	\$FF	Simple densité
E05F	20 2D	BRA	\$E08E		E137	43 49 50	FCC	"CIP"	"PICTOR"
---- Point d'entrée pour écriture d'un secteur sur floppy				Paramètres pour repérage de densité des logiciels					
E061	34 46	PSHS	U,B,A	Point d'entrée traitement	E13A	4A	FCB	\$4A	Checksum de l'application
E063	33 8C B0	LEAU	<\$E016,PCR	Banque 2 demandée	E13B	FF	FCB	\$FF	Simple densité
E066	86 02	LDA	#502	Exécution programme	E13C	4C 45 4D	FCC	"LEM"	"MELODIA"
E068	20 24	BRA	\$E08E		E13F	74	FCB	\$74	Checksum de l'application
---- Point d'entrée pour création d'un fichier sur floppy				Paramètres pour repérage de densité des logiciels					
E06A	34 46	PSHS	U,B,A	Point d'entrée traitement	E140	FF	FCB	\$FF	Simple densité
E06C	33 8C AA	LEAU	<\$E019,PCR	Banque 2 demandée	E146	53 45 47	FCC	"SEG"	"GESTION PRIVÉE3"
E06F	86 02	LDA	#502	Exécution programme	E149	B6	FCB	\$B6	Checksum de l'application
E071	20 1B	BRA	\$E08E		E14A	00	FCB	\$00	Double densité
---- Point d'entrée pour allocation d'un bloc sur floppy				***** SYSTEME D'EXPLOITATION PHYSIQUE FLOPPY *****					
E073	34 46	PSHS	U,B,A	Point d'entrée traitement	E141	4C 4F 50	FCC	"LOP"	"POLYPHONIA"
E075	33 8C A4	LEAU	<\$E01C,PCR	Banque 2 demandée	E144	B5	FCB	\$B5	Checksum de l'application
E078	86 02	LDA	#502	Exécution programme	E145	00	FCB	\$00	Double densité
E07A	20 12	BRA	\$E08E						
---- Point d'entrée pour initialisation d'une opération sur un bloc									
E07C	34 46	PSHS	U,B,A	Point d'entrée traitement					
E07E	33 8C 9E	LEAU	<\$E01F,PCR	Banque 2 demandée					
E081	86 02	LDA	#502	Exécution programme					
E083	20 09	BRA	\$E08E						
---- Point d'entrée pour clôture d'écriture sur floppy									
E085	34 46	PSHS	U,B,A	Point d'entrée traitement					
E087	33 8C 98	LEAU	<\$E022,PCR	Banque 2 demandée					
E08A	86 02	LDA	#502	Exécution programme					
E08C	20 00	BRA	\$E08E						
---- Module de commutation de banque ROM et d'exécution du traitement									
E08E	C6 01	LDB	#501	Banque 1 pour retour					
E090	34 04	PSHS	B	Commute banque ROM					
E092	A7 8D 07 42	STA	>\$E7D8,PCR	Exécute le programme					
E096	AD C4	JSR	,U	Retour en banque 1					
E098	35 04	PULS	B						
E09A	E7 8D 07 3A	STB	>\$E7D8,PCR						
E09E	35 C6	PULS	A,B,U,PC						

Contrôleur externe CD90-351 version A (THMFC1) - 2ème banque

E296 E6 84 LDB ,X	Attend fin de	E33E CC FF FF LDD # \$FFFF	Passé en mode
E298 2A FC BPL \$E296	transmission	E341 ED 03 STD \$03,X	"dommées"
E29A C6 4E LDB # \$4E	Ecrit un mot	E343 C6 24 LDB # \$24	Passé en
E29C E7 03 STB \$03,X	à \$4E	E345 E7 84 STB ,X	formatage
E29E 4A DECA	Pour les 22	---- Ecriture de l'amorce de piste	
E29F 26 F5 BNE \$E296	mots	E347 86 10 LDA # \$10	16 mots à écrire
---- Ecriture de 12 blancs		E349 E6 84 LDB ,X	Attend fin de
E2A1 86 0C LDA # \$0C	12 mots à écrire	E34B 2A FC BPL \$E349	transmission
E2A3 E6 84 LDB ,X	Attend fin de	E34D C6 FF LDB # \$FF	Ecrit un mot
E2A5 2A FC BPL \$E2A3	transmission	E34F E7 03 STB \$03,X	à \$FF
E2A7 6F 03 CLR \$03,X	Ecrit un mot à 0	E351 4A DECA	Pour les 16
E2A9 4A DECA	Pour les 12	E352 26 F5 BNE \$E349	mots
E2AA 26 F7 BNE \$E2A3	blancs	---- Ecriture de la plage de 6 blancs	
---- Ecriture des trois mots de synchronisation		E354 86 06 LDA # \$06	6 mots à écrire
E2AC E6 84 LDB ,X	Attend fin de	E356 E6 84 LDB ,X	Attend fin de
E2AE 2A FC BPL \$E2AC	transmission	E358 2A FC BPL \$E356	transmission
E2B0 CC A1 0A LDD # \$A10A	Passe en mode "synchro"	E35A 6F 03 CLR \$03,X	Ecrit un mot à 0
E2B3 ED 03 STD \$03,X	et écrit mot de synchro	E35C 4A DECA	Pour les 6
E2B5 E6 84 LDB ,X	Attend fin de	E35D 26 F7 BNE \$E356	mots
E2B7 2A FC BPL \$E2B5	transmission	---- Ecriture du marquage d'identification	
E2B9 A7 03 STA \$03,X	Ecrit mot de synchronisation	E35F E6 84 LDB ,X	Attend fin de
E2BB E6 84 LDB ,X	transmission	E361 2A FC BPL \$E35F	transmission
E2BD 2A FC BPL \$E2BB	transmission	E363 CC FE C7 LDD # \$PEC7	Ecrit le marquage
E2BF A7 03 STA \$03,X	Ecrit mot de synchronisation	E366 ED 03 STD \$03,X	en mode "synchro"
---- Ecriture du marquage de données		---- Ecriture de l'identificateur de secteur	
E2C1 E6 84 LDB ,X	Attend fin de	E368 E6 84 LDB ,X	Attend fin de
E2C3 2A FC BPL \$E2C1	transmission	E36A 2A FC BPL \$E368	transmission
E2C5 CC FB FF LDD # \$FBFF	Passe en mode "donnée"	E36C C6 FF LDB # \$FF	Lit premier paramètre
E2C8 ED 03 STD \$03,X	et écrit le marquage	E36E A6 C0 LDA ,U+	et passe en mode "donnée"
---- Ecriture du secteur double densité		E370 ED 03 STD \$03,X	Sauve le premier paramètre
E2CA 4F CLRA	256 mots à écrire	E372 86 05 LDA # \$05	5 paramètres restants
E2CB E6 84 LDB ,X	Attend fin de	E374 E6 84 LDB ,X	Attend fin de
E2CD 2A FC BPL \$E2CB	transmission	E376 2A FC BPL \$E374	transmission
E2CF C6 E5 LDB # \$E5	Ecrit un	E378 E6 C0 LDB ,U+	Lit le paramètre
E2D1 E7 03 STB \$03,X	mot à \$E5	E37A E7 03 STB \$03,X	Sauve le paramètre
E2D3 4A DECA	Pour les 256	E37C 4A DECA	Pour tous les
E2D4 26 F5 BNE \$E2CB	mots	E37D 26 F5 BNE \$E374	paramètres
---- Ecriture du CRC de données		---- Ecriture de la plage de battement	
E2D6 E6 84 LDB ,X	Attend fin de	E37F 86 0C LDA # \$0C	12 mots à écrire
E2D8 2A FC BPL \$E2D6	transmission	E381 E6 84 LDB ,X	Attend fin de
E2DA C6 78 LDB # \$78	Ecrit le	E383 2A FC BPL \$E381	transmission
E2DC E7 03 STB \$03,X	CRCHigh	E385 C6 FF LDB # \$FF	Ecrit un mot
E2DE E6 84 LDB ,X	Attend fin de	E387 E7 03 STB \$03,X	à \$FF
E2E0 2A FC BPL \$E2DE	transmission	E389 4A DECA	Pour les 12
E2E2 C6 27 LDB # \$27	Ecrit le	E38A 26 F5 BNE \$E381	mots
E2E4 E7 03 STB \$03,X	CRCLow	---- Ecriture de 6 blancs	
---- Ecriture de la plage de battement		E38C 86 06 LDA # \$06	6 mots à écrire
E2E6 86 2C LDA # \$2C	44 mots à écrire	E38E E6 84 LDB ,X	Attend fin de
E2E8 E6 84 LDB ,X	Attend fin de	E390 2A FC BPL \$E38E	transmission
E2EA 2A FC BPL \$E2E8	transmission	E392 6F 03 CLR \$03,X	Ecrit un mot à 0
E2EC C6 4E LDB # \$4E	Ecrit un mot	E394 4A DECA	Pour les 6
E2EE E7 03 STB \$03,X	à \$4E	E395 26 F7 BNE \$E38E	mots
E2F0 4A DECA	Pour les 44	---- Ecriture du marquage de données	
E2F1 26 F5 BNE \$E2E8	mots	E397 E6 84 LDB ,X	Attend fin de
---- Secteur suivant		E399 2A FC BPL \$E397	transmission
E2F3 6A E4 DEC ,S	Secteur	E39B CC FB C7 LDD # \$FBC7	Ecrit le marquage
E2F5 10 26 FF 65 LBNE \$E25E	suivant	E39E ED 03 STD \$03,X	en mode "synchro"
---- Ecriture jusqu'à l'index		---- Ecriture du secteur	
E2F9 A6 01 LDA \$01,X	Lit status disque	E3A0 E6 84 LDB ,X	Attend fin de
E2FB 85 40 BITA # \$40	Si index atteint,	E3A2 2A FC BPL \$E3A0	transmission
E2FD 26 08 BNE \$E307	exit	E3A4 CC E5 FF LDD # \$E5FF	Ecrit le premier caractère
E2FF A6 84 LDA ,X	Attend fin de	E3A7 ED 03 STD \$03,X	en mode "donnée"
E301 2A FC BPL \$E2FF	transmission	E3A9 86 7F LDA # \$7F	127 caractères restants
E303 E7 03 STB \$03,X	Ecrit mot de remplissage	E3AB E6 84 LDB ,X	Attend fin de
E305 20 F2 BRA \$E2F9	Mot suivant	E3AD 2A FC BPL \$E3AB	transmission
---- Test de demande de vérification		E3AF C6 E5 LDB # \$E5	Ecrit un caractère
E307 35 02 PULS A	Rétablit la pile	E3B1 E7 03 STB \$03,X	à \$E5
E309 6F 84 CLR ,X	Reset du lecteur	E3B3 4A DECA	Pour tout
E30B 0D 48 TST < \$48	Si pas de vérification,	E3B4 26 F5 BNE \$E3AB	le secteur
E30D 2A 1B BPL \$E32A	sort du programme	---- Ecriture du CRC de données	
---- Vérification de la piste formatée		E3B6 E6 84 LDB ,X	Attend fin de
E30F 86 E5 LDA # \$E5	Valeur pour remplissage	E3B8 2A FC BPL \$E3B6	transmission
E311 17 01 0C LBSR \$E420	Remplit le secteur	E3BA C6 5D LDB # \$5D	Ecrit le
E314 33 8D 7D BB LEAU > \$60D3,PCR	Ptr sur table entrelacements	E3BC E7 03 STB \$03,X	CRCHigh
E318 17 FE 8A LBSR \$E1A5	Activation du lecteur	E3BE E6 84 LDB ,X	Attend fin de
E31B A6 C0 LDA ,U+	Lit numéro de secteur	E3C0 2A FC BPL \$E3BE	transmission
E31D 27 0B BEQ \$E32A	Si fin de table, sort	E3C2 C6 30 LDB # \$30	Ecrit le
E31F 97 4C STA < \$4C	Initialise numéro de secteur	E3C4 E7 03 STB \$03,X	CRCLow
E321 34 40 PSHS U		---- Ecriture de la plage de battement	
E323 17 01 7B LBSR \$E4A1	Vérifie le secteur	E3C6 86 16 LDA # \$16	22 mots à écrire
E326 35 40 PULS U		E3C8 E6 84 LDB ,X	Attend fin de
E328 24 F1 BCC \$E31B	Si ok, secteur suivant	E3CA 2A FC BPL \$E3C8	transmission
E32A 39 RTS		E3CC C6 FF LDB # \$FF	Ecrit un mot
		E3CE E7 03 STB \$03,X	à \$FF
		E3D0 4A DECA	Pour les 22
		E3D1 26 F5 BNE \$E3C8	mots
Formatage d'une piste en simple densité		---- Secteur suivant	
E32B 17 02 6D LBSR \$E59B	Fixe la précompensation	E3D3 6A E4 DEC ,S	Secteur
---- Attend l'index		E3D5 10 26 FF 7B LBNE \$E354	suivant
E32E E6 01 LDB \$01,X		---- Ecriture jusqu'à l'index	
E330 C5 40 BITB # \$40	Attend l'index	E3D9 A6 01 LDA \$01,X	Lit status disque
E332 27 FA BEQ \$E32E		E3DB 85 40 BITA # \$40	Si index atteint,
E334 E6 03 LDB \$03,X	Efface registre d'écriture	E3DD 26 08 BNE \$E3E7	exit
E336 E6 84 LDB ,X		E3DF A6 84 LDA ,X	Attend fin de
E338 2A FA BPL \$E334	Petite temporisation	E3E1 2A FC BPL \$E3DF	transmission
E33A E6 84 LDB ,X	préalable	E3E3 E7 03 STB \$03,X	Ecrit mot de remplissage
E33C 2A FC BPL \$E33A			
---- Initialisation des registres pour formatage			

Contrôleur externe CD90-351 version A (THMFC1) - 2ème banque

```

E3E5 20 F2      BRA    $E3D9      Mot suivant
---- Test de demande de vérification
E3E7 35 02      PULS   A              Rétablit la pile
E3E9 6F 84      CLR    ,X             Reset du lecteur
E3EB 0D 48      TST   <$48           Si pas de vérification,
E3ED 2A 1D      BPL   $E40C          sort du programme
---- Vérification de la piste formatée en simple densité
E3EF 86 E5      LDA   #$E5           Valeur pour remplissage
E3F1 8D 2D      BSR   $E420          Remplit le secteur
E3F3 33 8D 7C DC LEAU  >$60D3,PCR     Ptr sur table entrecrochements
E3F7 17 FD AB    LBSR  $E1A5          Activation du lecteur
E3FA A6 C0      LDA   ,U+            Lit numéro de secteur
E3FC 27 0E      BEQ   $E40C          Si fin de table, sort
E3FE 97 4C      STA  <$4C            Initialise numéro de secteur
E400 34 40      PSHS  U             
E402 CC 5D 30   LDD   #$5D30         CRC de secteur
E405 17 00 E7   LBSR  $E4EF          Vérifie secteur
E408 35 40      PULS  U             
E40A 24 EE      BCC   $E3FA         Si ok, secteur suivant
E40C 39          RTS

      Fixe la précompensation pour double densité
E40D D6 4B      LDB   <$4B           Lit registre piste
E40F C1 39      CMPB  #$39          
E411 2A 05      BPL   $E418          Si au-delà de piste 57,
E413 CC 9F 20   LDD   #$9F20         séparateur actif
E416 20 03      BRA   $E41B         
E418 CC 1F 22   LDD   #$1F22         Sinon, séparateur inactif
E41B A7 07      STA  $07,X           Fixe séparateur
E41D E7 01      STB  $01,X           Fixe 256 mots par secteur
E41F 39          RTS

      Remplissage du secteur
E420 DE 4F      LDU   <$4F           Ptr sur buffer secteur
E422 D6 58      LDB   <$58           Lit la densité lecteur
E424 C4 80      ANDB  #$80           Calcule taille du secteur
E426 A7 C0      STA  ,U+            Remplit le
E428 5A          DECB                    secteur avec
E429 26 FB      BNE   $E426          la valeur convenue
E42B 39          RTS

      Déplacement de tête d'une piste pour le formatage
E42C 8D 6A      BSR   $E498          Temporisation
E42E 96 57      LDA  <$57           Lit bits de lecteur
E430 8A 20      ORA   #$20           Avance de la tête
E432 A7 02      STA  $02,X          demandée
E434 8A 14      ORA   #$14          
E436 A7 02      STA  $02,X          Stimulation du
E438 84 EB      ANDA  #$EB          déplacement pas à pas
E43A A7 02      STA  $02,X         
E43C 0C 4B      INC  <$4B          
E43E 96 4B      LDA  <$4B           Numéro de piste + 1
E440 A7 06      STA  $06,X         
E442 39          RTS

      Initialisation de la piste de catalogue
E443 C6 3B      LDB  #$3B           79-59= piste 20
E445 96 57      LDA  <$57           Lit bits de lecteur
E447 84 DF      ANDA  #$DF          Active le
E449 A7 02      STA  $02,X          lecteur
E44B 8A 10      ORA   #$10          
E44D A7 02      STA  $02,X          Déplace la tête d'une
E44F 84 EF      ANDA  #$EF          piste vers l'extérieur
E451 A7 02      STA  $02,X         
E453 8D 43      BSR  $E498          Temporisation
E455 5A          DECB                    Pour les
E456 26 F3      BNE  $E44B          59 pistes
E458 17 01 95   LBSR  $E5F0         Pointe sur position des têtes
E45B 86 14      LDA  #$14           Piste 20
E45D A7 A4      STA  ,Y              demandée
E45F A7 06      STA  $06,X          Init registre piste disque
E461 97 4B      STA  <$4B          Init registre piste moniteur
---- Initialise les 16 secteurs à $FF
E463 86 FF      LDA  #$FF           Valeur de remplissage
E465 8D B9      BSR  $E420          Remplit buffer de secteur
E467 86 10      LDA  #$10           16 secteurs
E469 97 4C      STA  <$4C           à écrire
E46B 17 FD 37   LBSR  $E1A5          Activation du lecteur
E46E 17 01 8A   LBSR  $E5FB          Ecrit le secteur
E471 25 18      BCS  $E48B          Si erreur, sort
E473 0A 4C      DEC  <$4C           Secteur
E475 26 F4      BNE  $E46B          suivant
---- Initialisation de la FAT
E477 DE 4F      LDU  <$4F           Ptr sur buffer de secteur
E479 6F C4      CLR  ,U              Premier octet de FAT à 0
E47B CC FE FE   LDD  #$FEFE         Réserve piste
E47E ED C8 29   STD  <$29,U         de catalogue
E481 86 02      LDA  #$02           Secteur 2
E483 97 4C      STA  <$4C           demandé
E485 17 01 73   LBSR  $E5FB          Ecrit le secteur
E488 25 01      BCS  $E48B          Si erreur, sort
E48A 4F          CLRA                    Pas d'erreur dans CC
E48B 39          RTS

      Temporisation de 864µs
E48C 10 8E 00 6C LDY   #$006C         Valeur de temporisation

```

```

E490 20 0A      BRA   $E49C          Effectue la boucle
---- Temporisation de 14976µs
E492 10 8E 07 50 LDY   #$0750         Valeur de temporisation
E496 20 04      BRA   $E49C          Effectue la boucle
---- Temporisation de 5984µs
E498 10 8E 02 EC LDY   #$02EC         Valeur de temporisation
E49C 31 3F      LEAY  -$01,Y        Boucle pour
E49E 26 FC      BNE  $E49C          temporisation
E4A0 39          RTS

      Lecture d'un secteur double densité
E4A1 C6 1B      LDB  #$1B           Lecture automatique
E4A3 34 01      PSHS  CC            Protège flag d'interruptions
E4A5 17 00 91   LBSR  $E539          Initialise registres disque
---- Recherche du secteur demandé
E4A8 86 04      LDA  #$04           Compteur recherche marquage
E4AA 10 8E 75 00 LDY   #$7500         Compteur pour une piste
E4AE 31 3F      LEAY  -$01,Y        Si fin du décomptage,
E4B0 10 27 00 80 LBEQ  $E534         erreur de vérification
E4B4 E6 84      LDB  ,X             Lit status opération
E4B6 C5 02      BITB  #$02          Si secteur pas trouvé,
E4B8 27 F4      BEQ  $E4AE          boucle
E4BA E6 03      LDB  $03,X         Réinitialise status disque
---- Recherche du marquage de données
E4BC 4A          DECA                    Si fin de décomptage,
E4BD 10 27 01 2B LBEQ  $E5EC         erreur de piste
E4C1 E6 84      LDB  ,X             Attend fin de
E4C3 2A FC      BPL  $E4C1         transmission
E4C5 E6 03      LDB  $03,X         Lit le caractère
E4C7 C1 FB      CMPB  #$FB          Si pas marquage de données,
E4C9 26 F1      BNE  $E4BC         nouveau test
---- Vérification du secteur double densité
E4CB 4F          CLRA                   
E4CC E6 84      LDB  ,X             256 caractères à vérifier
E4CE 2A FC      BPL  $E4CC         Attend fin de
E4D0 E6 03      LDB  $03,X         transmission
E4D2 E1 C0      CMPB  ,U+          Charge le caractère
E4D4 26 03      BNE  $E4D9         Si caractère incorrect,
E4D6 4A          DECA                    exit
E4D7 26 F3      BNE  $E4CC         Pour les 256
E4D9 E6 84      LDB  ,X             caractères
E4DB C5 08      BITB  #$08          Lit status opération
E4DD 27 FA      BEQ  $E4D9         Si pas fin d'opération,
E4DF 4D          TSTA                    boucle
E4E0 10 26 00 50 LBNE  $E534         Si pas secteur entier
E4E4 C5 04      BITB  #$04          vérifié, erreur
E4E6 10 26 00 4A LBNE  $E534         Si CRC de données
E4EA 35 01      PULS  CC            incorrect, erreur
E4EC 6F 84      CLR  ,X             Rétablit CC
E4EE 39          RTS             Reset lecteur

      Vérification d'un secteur simple densité
E4EF 34 01      PSHS  CC            Protège flag d'interruptions
E4F1 34 06      PSHS  B,A          
E4F3 8D 57      BSR  $E54C          Recherche du secteur
E4F5 25 3B      BCS  $E532          Si erreur, sort
---- Recherche du marquage de données
E4F7 C6 FB      LDB  #$FB           Mot de synchro
E4F9 E7 03      STB  $03,X         à trouver
E4FB E6 80      LDA  #$80           128 caractères à vérifier
E4FD C6 38      LDB  #$38           Passe en
E4FF E7 84      STB  ,X             "lecture synchro"
E501 E6 84      LDB  ,X            
E503 C5 01      BITB  #$01          Si pas de synchro, boucle
E505 27 FA      BEQ  $E501         
E507 E6 03      LDB  $03,X         Reset du status
---- Lecture du secteur simple densité
E509 C6 28      LDB  #$28           Passe en
E50B E7 84      STB  ,X             "lecture donnée"
E50D E6 84      LDB  ,X             Attend fin de
E50F 2A FC      BPL  $E50D         transmission
E511 E6 03      LDB  $03,X         Charge le caractère
E513 E1 C0      CMPB  ,U+          Si caractère incorrect,
E515 26 1B      BNE  $E532         erreur de vérification
E517 4A          DECA                    Pour tout
E518 26 F3      BNE  $E50D         le secteur
---- Lecture du CRC de données
E51A A6 84      LDA  ,X             Attend fin de
E51C 2A FC      BPL  $E51A         transmission
E51E A6 03      LDA  $03,X         Lit le CRCHigh
E520 E6 84      LDB  ,X             Attend fin de
E522 2A FC      BPL  $E520         transmission
E524 E6 03      LDB  $03,X         Lit le CRCLow
E526 10 A3 E4   CMPD  ,S            Si CRC incorrect,
E528 26 07      BNE  $E532         erreur de vérification
E52B 35 06      PULS  A,B           Rétablit la pile
E52D 35 01      PULS  CC            Rétablit le CC
E52F 6F 84      CLR  ,X             Reset du lecteur
E531 39          RTS

      Fixe l'erreur de vérification
E532 35 06      PULS  A,B           Rétablit la pile
E534 E6 20      LDA  #$20           Code "erreur en vérification"
E536 16 00 A3   LBRA  $E5DC         Fixe l'erreur

```


Contrôleur externe CD90-351 version A (THMFC1) - 2ème banque

----	écriture du secteur simple densité								
E673	C6 FF	LDB	#\$FF	Passe en mode "données"	E695	E6 84	LDB	,X	Attend fin de
E675	A6 84	LDA	,X	Attend fin de	E697	2A FC	BPL	,\$E695	transmission
E677	2A FC	BPL	,\$E675	transmission	E699	6F 03	CLR	,\$03,X	Ecrit un mot à 0
E679	A6 C0	LDA	,U+	Lit premier caractère	E69B	E6 84	LDB	,X	Attend fin de
E67B	ED 03	STD	,\$03,X	Enregistre premier caractère	E69D	2A FC	BPL	,\$E69B	transmission
E67D	86 7F	LDA	,\$7F	127 caractères restants	----	Passage du reste du secteur simple densité			
E67F	E6 84	LDB	,X	Attend fin de	E69F	C6 20	LDB	,\$20	Passe en mode
E681	2A FC	BPL	,\$E67F	transmission	E6A1	E7 84	STB	,X	"lecture données"
E683	E6 C0	LDB	,U+	Lit le caractère	E6A3	E6 84	LDB	,X	Attend fin de
E685	E7 03	STB	,\$03,X	Sauve le caractère	E6A5	2A FC	BPL	,\$E6A3	transmission
E687	4A	DECA		Pour tout	E6A7	E6 03	LDB	,\$03,X	Lit le caractère à perte
E688	26 F5	BNE	,\$E67F	le secteur	E6A9	43	COMA		2 fois de
----	écriture du CRC de données				E6AA	26 F7	BNE	,\$E6A3	suite
E68A	E6 84	LDB	,X	Attend fin de	E6AC	EC 7E	LDD	,-\$02,S	Relit le CRC
E68C	2A FC	BPL	,\$E68A	transmission	E6AE	35 01	PULS	CC	Dégel des interruptions
E68E	35 04	PULS	B	Récupère CRCHigh/Low	E6B0	6F 84	CLR	,X	Reset lecteur
E690	E7 03	STB	,\$03,X	Sauve CRCHigh/Low	E6B2	0D 48	TST	<\$48	Si pas vérification
E692	43	COMA		Si pas deuxième passe,	E6B4	10 2B FE 37	LBMI	,\$E4EF	demandée, traite
E693	26 F5	BNE	,\$E68A	boucle	E6B8	39	RTS		
----	Clôture du secteur simple densité								

Contrôleur externe CD90-351 version A (THMFC1) - 3ème banque

<p>Identificateur de RomDisk (IDDISK)</p> <pre> E000 4D FCC "M" Contrôleur Thomson E001 54 FCC "T" FAT 160 octets E002 44 FCC "D" Double densité E003 3A FCB \$3A Checksum de l'identificateur ---- Indirections standard sur les opérations physiques E004 16 00 24 LBRA \$E02B Fonctions standard E007 16 00 2A LBRA \$E034 Lancement du boot E00A 16 00 30 LBRA \$E03D Formatage ---- Indirections standard sur les opérations logiques E00D 16 00 F2 LBRA \$E102 Chargement de la FAT E010 16 01 75 LBRA \$E188 Ouverture d'un fichier E013 16 01 54 LBRA \$E16A Effacement d'un fichier E016 16 01 30 LBRA \$E149 Ecriture d'un secteur E019 16 01 D4 LBRA \$E1F0 Création d'un fichier E01C 16 02 2A LBRA \$E249 Allocation d'un bloc E01F 16 02 6C LBRA \$E28E Initialise opération sur bloc E022 16 00 86 LBRA \$E0AB Clôture d'écriture ---- Indirections standard sur les opérations physiques E025 16 FF FD LBRA \$E025 - Boucle à vide - E028 16 FF FD LBRA \$E028 - Boucle à vide - ---- Point d'entrée pour opération standard floppy E02B 34 46 PSHS U,B,A E02D 33 8C D4 LEAU <\$E004,PCR Point d'entrée traitement E030 86 00 LDA #\$00 Banque 0 demandée E032 20 5A BRA \$E08E Exécution programme ---- Point d'entrée pour lancement du boot floppy E034 34 46 PSHS U,B,A E036 33 8C CE LEAU <\$E007,PCR Point d'entrée traitement E039 86 00 LDA #\$00 Banque 0 demandée E03B 20 51 BRA \$E08E Exécution programme ---- Point d'entrée pour formatage QDD E03D 34 46 PSHS U,B,A E03F 33 8C C8 LEAU <\$E00A,PCR Point d'entrée traitement E042 86 01 LDA #\$01 Banque 1 demandée E044 20 48 BRA \$E08E Exécution programme ---- Point d'entrée pour chargement de la FAT floppy E046 34 46 PSHS U,B,A E048 33 8C C2 LEAU <\$E00D,PCR Point d'entrée traitement E04B 86 02 LDA #\$02 Banque 2 demandée E04D 20 3F BRA \$E08E Exécution programme ---- Point d'entrée pour ouverture d'un fichier sur floppy E04F 34 46 PSHS U,B,A E051 33 8C BC LEAU <\$E010,PCR Point d'entrée traitement E054 86 02 LDA #\$02 Banque 2 demandée E056 20 36 BRA \$E08E Exécution programme ---- Point d'entrée pour effacement d'un fichier sur floppy E058 34 46 PSHS U,B,A E05A 33 8C B6 LEAU <\$E013,PCR Point d'entrée traitement E05D 86 02 LDA #\$02 Banque 2 demandée E05F 20 2D BRA \$E08E Exécution programme ---- Point d'entrée pour écriture d'un secteur sur floppy E061 34 46 PSHS U,B,A E063 33 8C B0 LEAU <\$E016,PCR Point d'entrée traitement E066 86 02 LDA #\$02 Banque 2 demandée E068 20 24 BRA \$E08E Exécution programme ---- Point d'entrée pour création d'un fichier sur floppy E06A 34 46 PSHS U,B,A E06C 33 8C AA LEAU <\$E019,PCR Point d'entrée traitement E06F 86 02 LDA #\$02 Banque 2 demandée E071 20 1B BRA \$E08E Exécution programme ---- Point d'entrée pour allocation d'un bloc sur floppy E073 34 46 PSHS U,B,A E075 33 8C A4 LEAU <\$E01C,PCR Point d'entrée traitement E078 86 02 LDA #\$02 Banque 2 demandée E07A 20 12 BRA \$E08E Exécution programme ---- Point d'entrée pour initialisation d'opération sur un bloc E07C 34 46 PSHS U,B,A E07E 33 8C 9E LEAU <\$E01F,PCR Point d'entrée traitement E081 86 02 LDA #\$02 Banque 2 demandée E083 20 09 BRA \$E08E Exécution programme ---- Point d'entrée pour clôture d'écriture sur floppy E085 34 46 PSHS U,B,A E087 33 8C 98 LEAU <\$E022,PCR Point d'entrée traitement E08A 86 02 LDA #\$02 Banque 2 demandée E08C 20 00 BRA \$E08E Exécution programme ---- Module de commutation de banque ROM et d'exécution du traitement E08E C6 02 LDB #\$02 Banque 2 E090 34 04 PSHS B pour retour E092 A7 8D 07 42 STA >\$E7D8,PCR Commute banque ROM E096 AD C4 JSR ,U Exécute le programme E098 35 04 PULS B Retour en E09A E7 8D 07 3A STB >\$E7D8,PCR banque 2 E09E 35 C6 PULS A,B,U,PC ***** SYSTEME D'EXPLOITATION LOGIQUE ***** Nom de fichier temporaire pour la sauvegarde EOA0 53 43 52 41 FCC "SCRATCH.DOS" </pre>	<pre> EOA4 54 43 48 20 EOA8 44 4F 53 Clôture d'écriture EOAB D6 F0 LDB <\$F0 Lit code commande EOAD C1 02 CMPB #\$02 Si écrasement demandé, EOAF 27 25 BEQ \$E0D6 écrit FAT seule EOB1 0A F0 DEC <\$F0 Passe en sauveg. sans ecr. EOB3 17 00 D2 LBSR \$E188 Recherche fichier courant EOB6 25 31 BCS \$E0E9 Si erreur disque, sort EOB8 5D TSTB Si l'entrée n'existe pas, EOB9 27 05 BEQ \$E0C0 pas d'effacement EOBB 17 00 AC LBSR \$E16A Effacement du fichier EOBE 25 29 BCS \$E0E9 Si erreur, sort EOC0 0C F0 INC <\$F0 Retour en sauveg avec écrasmt EOC2 17 00 C3 LBSR \$E188 Recherche fichier "SCRATCH.DOS" EOC5 25 22 BCS \$E0E9 Si erreur, sort EOC7 C6 0A LDB #\$0A EOC9 9E E7 LDX <\$E7 Recopie le nom EOCB A6 85 LDA B,X du fichier EOCD A7 A5 STA B,Y courant à la place EOCF 5A DECB B,Y de "SCRATCH.DOS" EOD0 2C F9 BGE \$E0CB EOD2 8D 75 BSR \$E149 Ecrit le secteur catalogue EOD4 25 13 BCS \$E0E9 Si erreur de disque, sort ---- Sauvegarde de la FAT EOD6 86 02 LDA #\$02 EOD8 97 4C STA <\$4C Programme la EODA C6 14 LDB #\$14 sauvegarde EODC 4F CLRA de la FAT EODD DD 4A STD <\$4A EODF 8D 4C BSR \$E12D Rétablissement de FAT swapée EOE1 9F 4F STX <\$4F Sauvegarde EOE3 8D 64 BSR \$E149 de la FAT EOE5 25 02 BCS \$E0E9 Si erreur de disque, sort EOE7 0F F0 CLR <\$F0 Clôture du fichier effectuée EOE9 39 RTS Swap d'une FAT double densité EOEA 34 07 PSHS B,A,CC EOEB 9E ED LDX <\$ED Pointeur EOEE 30 89 00 80 LEAX >\$0080,X en milieu de FAT E0F2 A6 1F LDA -\$01,X Echange l'octet précédent E0F4 E6 88 7F LDB <\$7F,X avec le 127ème octet E0F7 A7 88 7F STA <\$7F,X E0FA E7 82 STB ,X Déplace pointeur à rebours E0FC 9E ED CMPX <\$ED Fin de FAT ? E0FE 26 F2 BNE \$E0F2 Non, échange suivant E100 35 87 PULS CC,A,B,PC Chargement de la FAT E102 DC ED LDD <\$ED Pointeur sur FAT swapée E104 0D 25 TST <\$25 FAT système initialisée ? E106 26 02 BNE \$E10A Non, initialise avec E108 DD 25 STD <\$25 pointeur sur FAT swapée E10A 93 25 SUBD <\$25 Si la différence entre les E10C C1 80 CMPB #\$80 deux est de 128 octets, E10E 27 04 BEQ \$E114 prend le pointeur système E110 9E ED LDX <\$ED Récupère le pointeur de E112 20 02 BRA \$E116 début de secteur de FAT E114 9E 25 LDX <\$25 Initialise les E116 9F ED STX <\$ED pointeurs pour charger E118 9F 4F STX <\$4F le secteur E11A 86 02 LDA #\$02 Chargement E11C 8D 31 BSR \$E14F de la FAT E11E 25 C9 BCS \$E0E9 Si erreur disque, sort E120 0D 58 TST <\$58 Contrôleur double densité ? E122 26 08 BNE \$E12C Oui, swap E124 8D C4 BSR \$E0EA de la FAT E126 30 89 00 80 LEAX >\$0080,X et pointeur courant E12A 9F ED STX <\$ED en milieu de FAT E12C 39 RTS Rétablissement de la FAT swapée E12D 9E ED LDX <\$ED E12F 0D 58 TST <\$58 E131 26 F9 BNE \$E12C double densité, E133 30 88 80 LEAX <-\$80,X repositionne le pointeur E136 8D F2 BSR \$E12A de FAT en début de secteur E138 20 B0 BRA \$E0EA et rétablit la FAT swapée Chargement du premier secteur de catalogue E13A 86 03 LDA #\$03 N° du ler secteur catalogue E13C 9E E9 LDX <\$E9 Pointeur sur buffer E13E 9F 4F STX <\$4F catalogue initialisé E140 8D 0D BSR \$E14F Chargement du secteur E142 25 E9 BCS \$E12D Si erreur, rétablit la FAT E144 39 RTS Sort sans erreur Chargement d'un secteur E145 8D 0F BSR \$E156 Charge le secteur E147 20 F9 BRA \$E142 Si erreur, rétablit la FAT </pre>
--	--

Contrôleur externe CD90-351 version A (THMFC1) - 3ème banque

E295	97	F5	STA	<\$F5	de secteur	E29D	39	RTS	
E297	24	02	BHS	\$(E29B	Selon la parité du numéro				
E299	86	09	LDA	#\$09	de bloc, le premier secteur				
E29B	97	FA	STA	<\$FA	est le secteur 1 ou 9				

Contrôleur externe CD90-351 version A (THMFC1) - 4ème banque

```

Identificateur de RomDisk (IDDISK)
E000 4D FCC "M"      Contrôleur Thomson
E001 4B FCC "K"      FAT 160 octets
E002 43 FCC "C"      Double densité
E003 30 FCB $30      Checksum de l'identificateur
  
```

```

----- Indirections standard sur les opérations physiques
E004 16 00 E4 LBRA $E0EB Fonctions standard QDD
E007 16 00 96 LBRA $E0A0 Lancement du boot QDD
E00A 16 03 94 LBRA $E3A1 Formatage QDD
----- Indirections standard sur les opérations logiques
E00D 16 05 47 LBRA $E557 Chargement de la FAT
E010 16 05 71 LBRA $E584 Ouverture d'un fichier
E013 16 06 6B LBRA $E681 Effacement d'un fichier
E016 16 05 5E LBRA $E577 Ecriture d'un secteur
E019 16 05 C7 LBRA $E5E3 Création d'un fichier
E01C 16 06 1E LBRA $E63D Allocation d'un bloc
E01F 16 06 7E LBRA $E6A0 Initialise opération sur bloc
E022 16 04 F5 LBRA $E51A Clôture d'écriture
----- Indirections standard sur les opérations physiques
E025 16 00 BB LBRA $E0E3 Fonctions standard QDD
E028 16 00 C5 LBRA $E0F0 Fonctions système QDD
  
```

```

----- Point d'entrée pour opération standard floppy
E02B 34 46 PSHS U,B,A
E02D 33 8C D4 LEAU <$E004,PCR Point d'entrée traitement
E030 86 00 LDA #$00 Banque 0 demandée
E032 20 5A BRA $E08E Exécution programme
  
```

```

----- Point d'entrée pour lancement du boot floppy
E034 34 46 PSHS U,B,A
E036 33 8C CE LEAU <$E007,PCR Point d'entrée traitement
E039 86 00 LDA #$00 Banque 0 demandée
E03B 20 51 BRA $E08E Exécution programme
  
```

```

----- Point d'entrée pour formatage QDD
E03D 34 46 PSHS U,B,A
E03F 33 8C C8 LEAU <$E00A,PCR Point d'entrée traitement
E042 86 01 LDA #$01 Banque 1 demandée
E044 20 48 BRA $E08E Exécution programme
  
```

```

----- Point d'entrée pour chargement de la FAT floppy
E046 34 46 PSHS U,B,A
E048 33 8C C2 LEAU <$E00D,PCR Point d'entrée traitement
E04B 86 02 LDA #$02 Banque 2 demandée
E04D 20 3F BRA $E08E Exécution programme
  
```

```

----- Point d'entrée pour ouverture d'un fichier floppy
E04F 34 46 PSHS U,B,A
E051 33 8C BC LEAU <$E010,PCR Point d'entrée traitement
E054 86 02 LDA #$02 Banque 2 demandée
E056 20 36 BRA $E08E Exécution programme
  
```

```

----- Point d'entrée pour effacement d'un fichier sur floppy
E058 34 46 PSHS U,B,A
E05A 33 8C B6 LEAU <$E013,PCR Point d'entrée traitement
E05D 86 02 LDA #$02 Banque 2 demandée
E05F 20 2D BRA $E08E Exécution programme
  
```

```

----- Point d'entrée pour écriture d'un secteur sur floppy
E061 34 46 PSHS U,B,A
E063 33 8C B0 LEAU <$E016,PCR Point d'entrée traitement
E066 86 02 LDA #$02 Banque 2 demandée
E068 20 24 BRA $E08E Exécution programme
  
```

```

----- Point d'entrée pour création d'un fichier sur floppy
E06A 34 46 PSHS U,B,A
E06C 33 8C AA LEAU <$E019,PCR Point d'entrée traitement
E06F 86 02 LDA #$02 Banque 2 demandée
E071 20 1B BRA $E08E Exécution programme
  
```

```

----- Point d'entrée pour allocation d'un bloc sur floppy
E073 34 46 PSHS U,B,A
E075 33 8C A4 LEAU <$E01C,PCR Point d'entrée traitement
E078 86 02 LDA #$02 Banque 2 demandée
E07A 20 12 BRA $E08E Exécution programme
  
```

```

----- Point d'entrée pour initialisation d'opération sur un bloc
E07C 34 46 PSHS U,B,A
E07E 33 8C 9E LEAU <$E01F,PCR Point d'entrée traitement
E081 86 02 LDA #$02 Banque 2 demandée
E083 20 09 BRA $E08E Exécution programme
  
```

```

----- Point d'entrée pour clôture d'écriture sur floppy
E085 34 46 PSHS U,B,A
E087 33 8C 98 LEAU <$E022,PCR Point d'entrée traitement
E08A 86 02 LDA #$02 Banque 2 demandée
E08C 20 00 BRA $E08E Exécution programme
  
```

```

----- Module de commutation de banque ROM et d'exécution du traitement
E08E C6 03 LDB #$03 Banque 3
E090 34 04 PSHS B pour retour
E092 A7 8D 07 42 STA >$E7D8,PCR Commute banque ROM
E096 6E C4 JMP ,U Exécute le programme
E098 35 04 PULS B Retour en
E09A E7 8D 07 3A STB >$E7D8,PCR banque 3
E09E 35 C6 PULS A,B,U,PC
  
```

***** SYSTEME D'EXPLOITATION PHYSIQUE DU QDD *****

```

E141 A7 02 STA $02,X | moteur QDD
E143 39 RTS |
  
```

```

Lancement du boot QDD
E0A0 17 00 66 LBSR $E109 Initialisation registres
E0A3 0F 49 CLR <$49 Lecteur 0 demandé
E0A5 17 00 7A LBSR $E122 Choix du lecteur
E0A8 17 00 AE LBSR $E159 Initialise en simple densité
E0AB CC 00 44 LDD #$0044 Secteur 68 (Boot QDD)
E0AE DD 4C STD <$4C sélectionné
E0B0 33 8D 81 4C LEAU >$6200,PCR Ptr buffer secteur
E0B4 DF 4F STU <$4F sélectionné
E0B6 17 01 D8 LBSR $E291 Lit le secteur QDD
E0B9 DE 4F LDU <$4F Ptr buffer secteur
E0BB 25 20 BCS $E0DD Si erreur, sort
E0BD 17 00 7F LBSR $E13F Eteint le lecteur
E0C0 31 8D 81 BB LEAY >$627F,PCR limite du décodage
E0C4 34 20 PSHS Y en pile
E0C6 86 55 LDA #$55 Checksum de départ
E0C8 6A C4 DEC ,U Décode les
E0CA 63 C4 COM ,U données
E0CC AB C0 ADDA ,U+ Calcule le checksum
E0CE 11 A3 E4 CMPLU ,S Si pas fin du décodage,
E0D1 26 F5 BNE $E0C8 boucle
E0D3 35 20 PULS Y Rétablit la pile
E0D5 A1 C4 CMPA ,U Si checksum incorrect,
E0D7 26 04 BNE $E0DD lance l'application
E0D9 6E 8D 81 23 JMP >$6200,PCR Sinon, exécute le boot
----- Sortie si erreur de boot
E0DD 0F 80 CLR <$80 Flag "contrôleur absent"
E0DF 6E 9F 00 1E JMP >[001E] Lance l'application à froid
  
```

```

Opérations standard avec numéro de secteur QDD réel
E0E3 34 7F PSHS U,Y,X,DP,B,A,CC
E0E5 8D 22 BSR $E109 Initialise registres disque
E0E7 86 FF LDA $FFF Flag pour opérations
E0E9 20 05 BRA $E0F0 QDD avec secteur réel
  
```

```

Opérations standard avec numéro de secteur QDD interprété
E0EB 34 7F PSHS U,Y,X,DP,B,A,CC
E0ED 8D 1A BSR $E109 Initialise registres disque
E0EF 4F CLR CLRA Flag de secteur interprété
----- Opération système avec numéro de secteur QDD interprété
E0F0 97 4A STA <$4A Ajourne le flag/n° de piste
E0F2 DC 4B LDD <$4B Protection de
E0F4 34 06 PSHS B,A ,DKTRK et DKSEC
E0F6 8D 2A BSR $E122 Choix du lecteur
E0F8 25 04 BCS $E0FE Si erreur, sort
E0FA 8D 4C BSR $E148 Exécute l'opération
E0FC 8D 41 BSR $E13F Eteint le moteur QDD
  
```

```

----- Sortie du programme
E0FE 35 06 PULS A,B | Restaure DKTRK
E100 DD 4B STD <$4B et DKSEC
E102 35 02 PULS A Dépile le CC
E104 1E 8A EXG A,CC Restaure le CC
E106 44 LSR A Bit d'erreur dans CC
E107 35 FE PULS A,B,DP,X,Y,U,PC
  
```

```

Initialisation des registres disque
E109 1F 50 TFR PC,D PC dans D
E10B 84 70 ANDA #$70 Initialise le DP
E10D 1F 8B TFR A,DP MO=$20 TO=$60
E10F 30 8D 06 BD LEAX >$E7D0,PCR Ptr sur registres disques
E113 0F 4E CLR <$4E Efface code d'erreur
E115 6F 84 CLR ,X Reset lecteur
E117 86 CE LDA $CE Séparateur de donnée
E119 A7 07 STA $07,X avec compteur de $4E
E11B 86 FF LDA $FFF Programme le mode données
E11D A7 04 STA $04,X le mode "domnée"
E11F 6F 01 CLR $01,X Pas de précompensation
E121 39 RTS
  
```

```

Sélection des bits de lecteur QDD
E122 D6 49 LDB <$49 Si face impaire
E124 C5 FD BITB #$FD demandée, erreur
E126 26 12 BNE $E13A "lecteur non prêt"
E128 33 8D 00 18 LEAU >$E144,PCR Table des bits de lecteurs
E12C A6 C5 LDA B,U Sélectionne
E12E A7 02 STA $02,X le lecteur
E130 97 57 STA <$57 adéquat.
E132 E6 01 LDB $01,X Si disque
E134 C5 40 BITB #$40 QDD absent, erreur
E136 26 02 BNE $E13A "lecteur non prêt"
E138 4F CLR CLRA Pas d'erreur dans CC
E139 39 RTS
  
```

```

Fixe l'erreur de lecteur
E13A 86 10 LDA #$10 Erreur "Lecteur non prêt"
E13C 16 01 B9 LBRA $E2F8 Fixe l'erreur
  
```

```

Extinction du moteur du QDD
E13F 86 00 LDA $00 | Extinction du
Table des bits de lecteurs
E144 01 FCB $01 Lecteur 0
E145 00 FCB $00 Lecteur 1 (inutilisé)
  
```


Contrôleur externe CD90-351 version A (THMFC1) - 4ème banque

```

Initialise le buffer de FAT
E3D7 DE 4F LDU <$4F Ptr buffer secteur
E3D9 CC 28 FF LDD #$28FF Initialise lère partie de
E3DC 8D 12 BSR $E3F0 FAT (40 blocs à $FF)
E3DE CC 02 FE LDD #$02FE Exclut la piste 20
E3E1 8D 0D BSR $E3F0 (2 blocs à $FE)
E3E3 CC 08 FF LDD #$08FF Initialise la suite de la
E3E6 8D 08 BSR $E3F0 FAT (8 blocs à $FF)
E3E8 CC 4D FE LDD #$4DFE Exclut le reste de la FAT
E3EB 8D 03 BSR $E3F0 (77 blocs à $FE)
E3ED CC 01 8C LDD #$018C Checksum du secteur de FAT

---- Ecriture du secteur
E3F0 E7 C0 STB ,U+ Pose l'état du bloc de FAT
E3F2 4A DECA $E3F0 Pour le nombre de
E3F3 26 FB BNE $E3F0 blocs convenus
E3F5 39 RTS

Retour en début de piste
E3F6 A6 01 LDA $01,X Lit le status
E3F8 85 02 BITA #$02 Si pas fin de piste,
E3FA 26 FA BNE $E3F6 boucle
E3FC 16 FE 61 LBRA $E260 Retour en début de piste

Formatage de la piste QDD
E3FF 34 01 PSHS CC Préserve les interruptions
E401 1A 50 ORCC #$50 Gel des interruptions
E403 DE 4F LDU <$4F Ptr buffer secteur
E405 E6 03 LDB $03,X Reset status
E407 E6 84 LDB ,X
E409 2A FA BPL $E405 Attente
E40B E6 84 LDB ,X préalable
E40D 2A FC BPL $E40B
E40F C6 04 LDB #$04 Initialise
E411 E7 84 STB ,X l'écriture MFM

---- Formate un secteur
E413 10 8E 0A EC LDY $0AEC Taille de l'amorce
E417 86 16 LDA $S16 Ecrit l'amorce
E419 8D 5E BSR $E479 de piste
E41B 10 8E 00 00 LDY $0000 Initialise numéro de secteur
E41F 31 21 LEAY $01,Y Secteur suivant
E421 10 9F 4C STY <$4C dans secteur courant
E424 86 A5 LDA $SA5 Ecrit le marquage
E426 8D 4D BSR $E475 d'identificateur
E428 96 4C LDA <$4C Ecrit le poids fort
E42A 8D 49 BSR $E475 du numéro de secteur
E42C 96 4D LDA <$4D Ecrit le poids faible
E42E 8D 45 BSR $E475 du numéro de secteur
E430 9B 4C ADDA <$4C Met à jour et
E432 8B A5 ADDA $SA5 écrit le checksum
E434 8D 3F BSR $E475 d'identificateur
E436 10 8E 00 07 LDY $0007 Ecrit
E43A 86 16 LDA $S16 10
E43C 8D 3B BSR $E479 espaces
E43E 86 5A LDA $SA5 Ecrit le marquage
E440 8D 33 BSR $E475 de secteur
E442 10 8E 00 05 LDY $0005 Si le secteur courant est
E446 10 9C 4C CMPY <$4C le secteur de FAT,
E449 27 39 BEQ $E484 écrit le secteur de FAT
E44B 10 8E 00 80 LDY $0080 Sinon, écrit un
E44F 86 FF LDA $SFF secteur
E451 8D 26 BSR $E479 quelconque
E453 86 DA LDA $SDA Ecrit le checksum
E455 8D 1E BSR $E475 de secteur
E457 10 8E 00 14 LDY $0014 Ecrit
E45B 86 16 LDA $S16 17
E45D 8D 1A BSR $E479 espaces

---- Secteur suivant
E45F 10 9E 4C LDY <$4C Lit numéro de secteur
E462 10 8C 01 90 CMPY $S0190 Si pas dernier secteur,
E466 26 B7 BNE $E41F secteur suivant
E468 8D 0B BSR $E475 Ecrit un espace
E46A E6 01 LDB $01,X Lit status
E46C C5 02 BITB $S02 Si pas fin de piste,
E46E 26 F8 BNE $E468 comble
E470 6F 84 CLR ,X
E472 35 01 PULS CC Reset lecteur
E474 39 RTS Rétablit les interruptions

---- Module d'écriture d'un caractère
E475 10 8E 00 01 LDY $0001 - Entrée pour écrire 1 mot
E479 E6 84 LDB ,X - Entrée pour écrire Y mots
E47B 2A FC BPL $E479 Attente signal transmission
E47D A7 03 STA $03,X Ecriture du caractère A
E47F 31 3F LEAY -$01,Y Y
E481 26 F6 BNE $E479 fois
E483 39 RTS

---- Ecriture du secteur de FAT
E484 DE 4F LDU <$4F Ptr buffer secteur
E486 4F CLRA Premier caractère à 0
E487 8D EC BSR $E475 Ecrit le caractère
E489 81 8C CMPA $S8C Si fin de FAT,
E48B 27 CA BEQ $E457 retour au formatage
E48D A6 C0 LDA ,U+ Lit caractère de FAT
E48F 20 F6 BRA $E487 Sauve caractère de FAT

Vérification du formatage QDD
E491 10 8E 00 00 LDY $0000 Initialise numéro de secteur

```

```

E495 31 21 LEAY $01,Y Secteur suivant
E497 10 9F 4C STY <$4C dans secteur courant
E49A DE 4F LDU <$4F Ptr buffer FAT
E49C 17 FD B5 LBSR $E254 Teste disponibilité lecteur
E49F 25 69 BCS $E50A Si erreur, sort

---- Recherche du secteur
E4A1 34 01 PSHS CC Préserve interruptions
E4A3 17 FD 2F LBSR $E1D5 Recherche du secteur
E4A6 25 63 BCS $E50B Si erreur, sort
E4A8 CC 5A 80 LDD #$5A80 Initialisation de la
E4AB ED 03 STD $03,X détection synchronisation
E4AD 86 80 LDA $S80 128 caractères à vérifier
E4AF C6 18 LDB $S18 Initialise la lecture MFM
E4B1 E7 84 STB ,X + synchro + séparateur

---- Recherche de la synchronisation
E4B3 E6 84 LDB ,X Lit status
E4B5 C5 01 BITB $S01 Si pas de synchro,
E4B7 27 FA BEQ $E4B3 boucle
E4B9 E6 03 LDB $03,X Reset status

---- Vérification du secteur
E4BB C6 FF LDB $SFF Initialise en lecture
E4BD E7 04 STB $04,X de données
E4BF C6 08 LDB $S08 Exclut la détection de
E4C1 E7 84 STB ,X synchronisation
E4C3 10 9E 4C LDY <$4C Lit numéro de secteur
E4C6 10 8C 00 05 CMPY $S0005 Si secteur de FAT,
E4CA 27 1E BEQ $E4EA vérification spéciale
E4CC E6 84 LDB ,X Attend fin de
E4CE 2A FC BPL $E4CC transmission
E4D0 E6 03 LDB $03,X Lit le caractère
E4D2 C1 FF CMPB $SFF Si pas $FF,
E4D4 26 0F BNE $E4E5 erreur de vérification
E4D6 4A DECA Pour les 128
E4D7 26 F3 BNE $E4CC caractères

---- Vérification de la validité du checksum
E4D9 E6 84 LDB ,X Attend fin de
E4DB 2A FC BPL $E4D9 transmission
E4DD E6 03 LDB $03,X Lit le caractère
E4DF 6F 84 CLR ,X Reset lecteur
E4E1 C1 DA CMPB $SDA Si checksum correct,
E4E3 27 1A BEQ $E4FF secteur suivant
E4E5 86 20 LDA $S20 Code "erreur de vérification"
E4E7 16 FE 0E LBRA $E2F8 Sort avec erreur

---- Vérification du secteur de FAT
E4EA E6 84 LDB ,X Attend fin de
E4EC 2A FC BPL $E4EA transmission
E4EE E6 03 LDB $03,X Si caractère lu <0>,
E4F0 26 F3 BNE $E4E5 erreur de vérification
E4F2 E6 84 LDB ,X Attend fin de
E4F4 2A FC BPL $E4F2 transmission
E4F6 E6 03 LDB $03,X Lit le caractère
E4F8 E1 C0 CMPB ,U+ Compare avec secteur de FAT
E4FA 26 E9 BNE $E4E5 Si erreur, sort
E4FC 4A DECA Pour les 128
E4FD 26 F3 BNE $E4F2 caractères

---- Secteur suivant
E4FF 35 01 PULS CC Rétablit les interruptions
E501 10 9E 4C LDY <$4C Lit numéro de secteur
E504 10 8C 01 90 CMPY $S0190 Si pas dernier secteur,
E508 26 8B BNE $E495 secteur suivant
E50A 39 RTS

---- Sortie si erreur
E50B 35 01 PULS CC Rétablit les interruptions
E50D 43 COMA Erreur dans CC
E50E 39 RTS

***** SYSTEME D'EXPLOITATION LOGIQUE *****

Nom de fichier temporaire pour la sauvegarde
E50F 53 43 52 41 FCC "SCRATCH DOS"
E513 54 43 48 20
E517 44 4F 53

Clôture d'écriture
E51A D6 F0 LDB <$F0 Sauvegarde sans
E51C C1 02 CMPB $S02 écrasement ?
E51E 27 23 BEQ $E543 Oui, écrit FAT seule
E520 0A F0 DEC <$F0 Passe en sauveg. sans ecr.
E522 8D 60 BSR $E584 Cherche fichier courant
E524 25 30 BCS $E556 Si erreur disque, sort
E526 5D TSTB Fichier existe déjà?
E527 27 05 BEQ $E52E Non, pas d'effacement
E529 17 01 55 LBSR $E681 Effacement du fichier
E52C 25 28 BCS $E556 Si erreur, sort
E52E 0C F0 INC <$F0 Retour en sauveg avec écrasmt
E530 8D 52 BSR $E584 Cherche fichier "SCRATCH.DOS"
E532 25 22 BCS $E556 Si erreur, sort
E534 C6 0A LDB $S0A
E536 9E E7 LDY <$E7 Recopie le nom
E538 A6 85 LDA B,X de fichier
E53A A7 A5 STA B,Y courant à la place
E53C 5A DECB de "SCRATCH.DOS"
E53D 2C F9 BGE $E538
E53F 8D 36 BSR $E577 Ecrit le secteur catalogue

```

Contrôleur externe CD90-351 version A (THMFC1) - 4ème banque

```

E541 25 13      BCS  $E556      Si erreur de disque, sort
--- Sauvegarde de la FAT
E543 86 02      LDA  #$02
E545 97 4C      STA  <$4C
E547 C6 14      LDB  #$14
E549 4F         CLR  CLRA
E54A DD 4A      STD  <$4A
E54C DC ED      LDD  <$ED
E54E DD 4F      STD  <$4F
E550 8D 25      BSR  $E57
E552 25 02      BCS  $E55
E554 0F F0      CLR  <$F0
E556 39         RTS

      Chargement de la FAT
E557 9E ED      LDX  <$ED
E559 9F 4F      STX  <$4F
E55B 86 02      LDA  #$02
E55D 20 0D      BRA  $E56C

      Fixe l'erreur moniteur
E55F 97 E5      STA  <$E5
E561 43         COMA
E562 43         COMA
E563 39         RTS

      Sortie sans erreur
E564 4F         CLR  CLRA
E565 39         RTS

      Charge le premier secteur de catalogue
E566 86 03      LDA  #$03
E568 9E E9      LDX  <$E9
E56A 9F 4F      STX  <$4F
E56C 97 4C      STA  <$4C
E56E C6 14      LDB  #$14
E570 4F         CLR  CLRA
E571 DD 4A      STD  <$4A
E573 86 02      LDA  #$02
E575 20 02      BRA  $E579

      Routine pour sauvegarde d'un secteur
E577 86 08      LDA  #$08
E579 97 48      STA  <$48
E57B 10 9E E9   LDX  <$E9
E57E 17 FA 83   LBSR $E004
E581 86 03      LDA  #$03
E583 39         RTS

      Ouverture d'un fichier
E584 8D E0      BSR  $E566
E586 25 D7      BCS  $E55F
E588 8E 00 04   LDX  #$0004
E58B 10 9E E9   LDX  <$E9
E58E DE E7      LDU  <$E7
E590 D6 F0      LDB  <$F0
E592 C1 03      CMPB #$03
E594 26 04      BNE  $E59A
E596 33 8D FF 75 LEAU >$E50F,PCR
E59A 5F         CLR  CLRB
E59B C1 0B      CMPB #$0B
E59D 24 24      BHS  $E5C3
E59F A6 A5      LDA  B,Y
E5A1 81 FF      CMPA #$FF
E5A3 27 1B      BEQ  $E5C0
E5A5 5C         INCB
E5A6 A1 C0      CMPA ,U+
E5A8 27 F1      BEQ  $E59B
E5AA 31 A8 20   LEAY <$20,Y
E5AD 30 1F      LEAX -$01,X
E5AF 26 DD      BNE  $E58E
E5B1 0C 4C      LNC  <$4C
E5B3 96 4C      LDA  <$4C
E5B5 81 10      CMPA #$10
E5B7 22 07      BHI  $E5C0
E5B9 17 FA 48   LBSR $E004
E5BC 86 03      LDA  #$03
E5BE 20 C6      BRA  $E586
---- Sortie si fichier introuvable
E5C0 5F         CLR  CLRB
E5C1 20 1B      BRA  $E5DE

      Récupération des paramètres du fichier
E5C3 E6 2B      LDB  $0B,Y
E5C5 D1 EB      CMPB <$EB
E5C7 26 E1      BNE  $E5AA
E5C9 E6 2C      LDB  $0C,Y
E5CB D1 EC      CMPB <$EC
E5CD 26 DB      BNE  $E5AA
E5CF D6 4C      LDB  <$4C
E5D1 A6 2D      LDA  $0D,Y
E5D3 97 F6      STA  <$F6
E5D5 0F F5      CLR  <$F5
E5D7 AE 2E      LDX  $0E,Y
E5D9 9F F7      STX  <$F7
E5DB 10 9F FA   STY  <$FA
    
```

```

E5DE D7 F9      STB  <$F9
E5E0 16 FF 81   LBRA  $E564

      Création d'un fichier
E5E3 10 9E ED   LDY  <$ED
E5E6 8D 6F      BSR  $E657
E5E8 25 9C      BCS  $E586
E5EA D7 F6      STB  <$F6
E5EC 17 FF 77   LBSR $E566
E5EF 25 F7      BCS  $E5E8
E5F1 10 9E E9   LDY  <$E9
E5F4 8E 00 04   LDX  #$0004
E5F7 E6 A4      LDB  ,Y
E5F9 27 20      BEQ  $E61B
E5FB 86 05      LDA  #$05
E5FD C1 FF      CMPB #$FF
E5FF 27 1A      BEQ  $E61B
E601 31 A8 20   LEAY <$20,Y
E604 30 1F      LEAX -$01,X
E606 26 EF      BNE  $E5F7
E608 0C 4C      INC  <$4C
E60A 96 4C      LDA  <$4C
E60C 81 10      CMPA #$10
E60E 22 07      BHI  $E617
E610 17 F9 F1   LBSR $E004
E613 86 03      LDA  #$03
E615 20 D8      BRA  $E5EF
---- Sortie si erreur "disque plein"
E617 86 05      LDA  #$05
E619 20 CD      BRA  $E5E8

      Mise à jour du fichier
E61B 9E E7      LDX  <$E7
E61D D6 F0      LDB  <$F0
E61F C1 03      CMPB #$03
E621 26 04      BNE  $E627
E623 30 8D FE E8 LEAX >$E50F,PCR
E627 C6 0A      LDB  #$0A
E629 A6 85      LDA  B,X
E62B A7 A5      STA  B,Y
E62D 5A         DECB
E62E 2C F9      BGE  $E629
E630 96 EB      LDA  <$EB
E632 A7 2B      STA  $0B,Y
E634 96 EC      LDA  <$EC
E636 D6 F6      LDB  <$F6
E638 ED 2C      STD  $0C,Y
E63A 16 FF 3A   LBRA  $E577

      Allocation d'un bloc
E63D D6 F6      LDB  <$F6
---- Recherche chromatique d'un bloc libre dans la FAT (bloc quelconque)
E63F C1 28      CMPB #40
E641 22 0E      BHI  $E651
E643 5D         TSTB
E644 27 11      BEQ  $E657
E646 A6 A5      LDA  B,Y
E648 81 FF      CMPA #$FF
E64A 27 2D      BEQ  $E679
E64C 5A         DECB
E64D C1 28      CMPB #40
E64F 23 F2      BLS  $E643
E651 CB 02      ADDB #$02
E653 C1 51      CMPB #$51
E655 20 ED      BRA  $E644
---- Recherche par symétrie d'un bloc libre dans la FAT (bloc de départ)
E657 5F         CLR  CLRB
E658 31 A8 28   LEAY <40,Y
E65B 86 05      LDA  #$05
E65D C1 28      CMPB #40
E65F 10 22 FE FC LEAY <$22,FE,FC
E663 A6 A5      LDA  B,Y
E665 81 FF      CMPA #$FF
E667 27 0B      BEQ  $E674
E669 50         NEGB
E66A A6 A5      LDA  B,Y
E66C 81 FF      CMPA #$FF
E66E 27 04      BEQ  $E674
E670 50         NEGB
E671 5C         INCB
E672 20 E7      BRA  $E65B
E674 CB 28      ADDB #40
E676 31 A8 D8   LEAY <-40,Y
E679 6F A5      CLR  B,Y
E67B 5A         DECB
E67C D7 F9      STB  <$F9
E67E 16 FE E3   LBRA  $E564

      Effacement du fichier courant
E681 A6 2D      LDA  $0D,Y
E683 97 F6      STA  <$F6
E685 6F A4      CLR  ,Y
E687 17 FE ED   LBSR $E577
E68A 25 8D      BCS  $E619
E68C 10 9E ED   LDY  <$ED
E68F D6 F6      LDB  <$F6
    
```

Contrôleur externe CD90-351 version A (THMFC1) - 4ème banque

E691	5C	INCB		(met à \$FF)	E6A0	D6 F6	LDB	<\$F6	Numéro de bloc courant
E692	A6 A5	LDA	B, Y	tous les blocs	E6A2	4F	CLRA		Calcule le
E694	6F A5	CLR	B, Y	du fichier	E6A3	54	LSRB		numéro
E696	6A A5	DEC	B, Y	courant	E6A4	DD FB	STD	<\$FB	de piste
E698	1F 89	TFR	A, B	répertoriés	E6A6	4C	INCA		Initialisation du compteur
E69A	81 C0	CMPA	#\$C0	dans la FAT	E6A7	97 F5	STA	<\$F5	de secteur
E69C	25 F3	BLO	#\$E691		E6A9	24 02	BHS	#\$E6AD	Selon la parité du numéro
E69E	20 DE	BRA	#\$E67E	Pas d'erreur dans CC	E6AB	86 09	LDA	#\$09	de bloc, le premier secteur
					E6AD	97 FA	STA	<\$FA	est le secteur 1 ou 9
					E6AF	39	RTS		

Initialisation d'une opération sur un bloc

Contrôleur externe CD90-351 version B (THMFC1) - 1ère banque

<p>E157 D6 49 LDB <\$49 E159 C1 04 CMPB #S04 E15B 10 24 03 DD LBHS \$E53C E15F 20 EA BRB \$E14B</p> <p style="text-align: center;">Extinction du moteur</p> <p>E161 86 40 LDA #S40 E163 A7 02 STA \$02,X E165 39 RTS</p> <p style="text-align: center;">Table des bits de lecteurs</p> <p>E166 41 FCB \$41 E167 01 FCB \$01 E168 42 FCB \$42 E169 02 FCB \$02</p> <p style="text-align: center;">Module d'exécution des fonctions standard</p> <p>E16A 86 02 LDA #S02 E16C 34 02 PSHS A E16E 96 48 LDA <\$48 E170 31 8D 00 51 LEAY >\$E1C5,PCR E174 85 01 BITA #S01 E176 26 30 BNE \$E1A8 E178 31 8D 02 EC LEAY >\$E468,PCR E17C 85 02 BITA #S02 E17E 26 28 BNE \$E1A8 E180 31 8D 00 BA LEAY >\$E23E,PCR E184 85 08 BITA #S08 E186 26 20 BNE \$E1A8 E188 31 8D 00 6F LEAY >\$E1FB,PCR E18C 85 20 BITA #S20 E18E 26 18 BNE \$E1A8 E190 31 8D 03 86 LEAY >\$E51A,PCR E194 85 40 BITA #S40 E196 26 10 BNE \$E1A8 E198 31 8D 00 99 LEAY >\$E235,PCR E19C 85 04 BITA #S04 E19E 26 08 BNE \$E1A8 E1A0 31 8D 00 97 LEAY >\$E23B,PCR E1A4 85 10 BITA #S10 E1A6 27 1A BEQ \$E1C2 E1A8 AD A4 JSR ,Y E1AA 24 17 BCC \$E1C3 E1AC 96 4E LDA <\$4E E1AE 85 51 BITA #S51 E1B0 26 11 BNE \$E1C3 E1B2 6A E4 DEC ,S E1B4 27 0C BEQ \$E1C2 E1B6 0F 4E CLR <\$4E E1B8 81 08 CMPA #S08 E1BA 27 B2 BEQ \$E16E E1BC 8D 3D BSR \$E1FB E1BE 25 F2 BCS \$E1B2 E1C0 20 AC BRA \$E16E E1C2 53 COMB E1C3 35 82 PULS A,PC</p> <p style="text-align: center;">Routine de reset du lecteur</p> <p>E1C5 8D 34 BSR \$E1FB E1C7 25 31 BCS \$E1FA</p> <p style="text-align: center;">--- Définition de la densité système</p> <p>E1C9 86 01 LDA #S01 E1CB 97 4B STA <\$4B E1CD 17 03 72 LBSR \$E542 E1D0 C6 44 LDB #D E1D2 A6 01 LDA \$01,X E1D4 85 08 BITA #S08 E1D6 27 02 BEQ \$E1DA E1D8 C6 43 LDB #'C E1DA D7 4E STB <\$4E</p> <p style="text-align: center;">--- Retour des têtes en piste 0</p> <p>E1DC 8D 1D BSR \$E1FB E1DE 0F 52 CLR <\$52 E1E0 0F 54 CLR <\$54 E1E2 6F 84 CLR ,X E1E4 0F 58 CLR <\$58</p> <p style="text-align: center;">--- Définition de la densité du lecteur</p> <p>E1E6 DC E9 LDD <\$E9 E1E8 93 ED SUBD <\$ED E1EA 2C 05 BGE \$E1F1 E1EC 43 COMA E1ED 53 COMB E1EE C3 00 01 ADDD #S0001 E1F1 4D TSTA E1F2 26 05 BNE \$E1F9 E1F4 5D TSTB E1F5 2A 02 BPL \$E1F9 E1F7 03 58 COM <\$58 E1F9 4F CLRA E1FA 39 RTS</p> <p style="text-align: center;">Recherche de la piste 0</p> <p>E1FB C6 64 LDB #S64 E1FD A6 01 LDA \$01,X E1FF 85 08 BITA #S08 E201 26 13 BNE \$E216</p>	<p>Lit numéro de lecteur Si >= à 4, erreur de lecteur Sélectionne bits de lecteur</p> <p>Code pour éteindre moteur Éteint le moteur</p> <p>Lecteur 0 Lecteur 1 Lecteur 2 Lecteur 3</p> <p>2 tentatives maximum Lit code de commande Vecteur "reset contrôleur" Si reset contrôleur demandé, traite Vecteur "lecture secteur" Si lecture secteur demandée, traite Vecteur "écriture secteur" Si écriture secteur demandée, traite Vecteur "recherche piste 0" Si piste 0 demandée, traite Vecteur "recherche piste" Si recherche de piste demandée, traite Vecteur "simple densité" Si simple densité demandée, traite Vecteur "double densité" Si double densité pas demandée, sort avec erreur Exécute l'opération Si pas d'erreur, sort Lit code d'erreur moniteur Si erreur contrôleur/lecteur/protection, sort Si dernière tentative, sort Efface code d'erreur Si erreur sur les données, nouvelle tentative Recherche de la piste 0 Si erreur, nouvel essai Nouvelle tentative Erreur dans CC</p> <p>Recherche de la piste 0 Si erreur, sort</p> <p>Piste 1 demandée Déplace les têtes Code double densité Lit status lecteur Si pas de détection de piste 0, double densité Code simple densité Fixe le code densité</p> <p>Recherche de la piste 0 Toute tête en piste 0 Éteint le moteur Initialise en double densité</p> <p>Récupère la valeur absolue de la différence entre le pointeur de FAT et le pointeur de buffer</p> <p>Si différence > 256, double densité Si différence < 128, double densité Sinon simple densité Pas d'erreur dans CC</p> <p>100 tentatives maximum Lit status Si piste 0 trouvée, sort</p>	<p>E203 96 57 LDA <\$57 E205 8A 10 ORA #S10 E207 A7 02 STA \$02,X E209 84 EF ANDA #SEF E20B A7 02 STA \$02,X E20D 8D 1D BSR \$E22C E20F 5A DECB E210 10 27 03 28 LBEQ \$E53C E214 20 E7 BRA \$E1FD E216 8D 0E BSR \$E22E E218 17 03 55 LBSR \$E570 E21B 6F A4 CLR ,Y E21D 6F 06 CLR \$06,X E21F 39 RTS</p> <p style="text-align: center;">Temporisation de 864µs</p> <p>E220 10 8E 00 6C LDY #S006C E224 20 0A BRA \$E230</p> <p style="text-align: center;">---- Temporisation de 14976µs</p> <p>E226 10 8E 07 50 LDY #S0750 E22A 20 04 BRA \$E230</p> <p style="text-align: center;">---- Temporisation de 5984µs</p> <p>E22C 10 8E 02 EC LDY #S02EC E230 31 3F LEAY -\$01,Y E232 26 FC BNE \$E230 E234 39 RTS</p> <p style="text-align: center;">Routine de passage en simple densité</p> <p>E235 86 FF LDA #SFF E237 97 58 STA <\$58 E239 4F CLRA E23A 39 RTS</p> <p style="text-align: center;">Routine de passage en double densité</p> <p>E23B 0F 58 CLR <\$58 E23D 39 RTS</p> <p style="text-align: center;">Routine d'écriture d'un secteur</p> <p>E23E 17 02 D9 LBSR \$E51A E241 10 25 00 9E LBSC \$E2E3 E245 A6 01 LDA \$01,X E247 85 04 BITA #S04 E249 27 05 BEQ \$E250 E24B 86 01 LDA #S01 E24D 16 02 EE LBRA \$E53E E250 96 58 LDA <\$58 E252 10 26 00 92 LBNE \$E2E8</p> <p style="text-align: center;">---- Ecriture d'un secteur double densité</p> <p>E256 17 01 BD LBSR \$E416 E259 C6 19 LDB #S19 E25B 34 01 PSHS CC E25D 17 01 4C LBSR \$E3AC</p> <p style="text-align: center;">---- Recherche du secteur demandé</p> <p>E260 10 8E 75 00 LDY #S7500 E264 31 3F LEAY -\$01,Y E266 10 27 02 9E LBEQ \$E508 E26A E6 84 LDB ,X E26C C5 02 BITB #S02 E26E 27 F4 BEQ \$E264 E270 A7 03 STA \$03,X</p> <p style="text-align: center;">---- Ecriture du secteur double densité</p> <p>E272 E6 84 LDB ,X E274 2A FC BPL \$E272 E276 A7 03 STA \$03,X E278 E6 84 LDB ,X E27A 2A FC BPL \$E278 E27C C6 FB LDB #SFB E27E E7 03 STB \$03,X E280 4F CLRA256 caractères à écrire E281 E6 84 LDB ,X E283 2A FC BPL \$E281 E285 E6 C0 LDB ,U+ E287 E7 03 STB \$03,X E289 4A DECA Pour les 256 E28A 26 F5 BNE \$E281 E28C E6 84 LDB ,X E28E C5 08 BITB #S08 E290 27 FA BEQ \$E28C E292 17 FF 8B LBSR \$E220 E295 0D 48 TST <\$48 E297 2A 4A BPL \$E2E3 E299 17 02 86 LBSR \$E522 E29C C6 1B LDB #S1B E29E 17 01 0B LBSR \$E3AC</p> <p style="text-align: center;">---- Recherche du secteur demandé</p> <p>E2A1 86 04 LDA #S04 E2A3 10 8E 75 00 LDY #S7500 E2A7 31 3F LEAY -\$01,Y E2A9 10 27 00 FA LBEQ \$E3A7 E2AD E6 84 LDB ,X E2AF C5 02 BITB #S02 E2B1 27 F4 BEQ \$E2A7 E2B3 E6 03 LDB \$03,X</p> <p style="text-align: center;">---- Recherche du marquage de données</p> <p>E2B5 4A DECA E2B6 10 27 02 56 LBEQ \$E510</p>	<p>Lit bits lecteur</p> <p>Déplace la tête</p> <p>Temporisation Si dernière tentative, erreur "Lecteur non prêt"</p> <p>Nouveau déplacement Temporisation Ptr position de la tête Tête en piste 0 Initialise registre piste</p> <p>Valeur de temporisation Effectue la boucle</p> <p>Valeur de temporisation Effectue la boucle</p> <p>Valeur de temporisation Boucle pour temporisation</p> <p>Fixe flag de simple densité Pas d'erreur dans CC</p> <p>Fixe flag de double densité</p> <p>Recherche la piste Si erreur, sort Lit le status disque Si bit de protection actif, initialise code "Disquette protégée" et sort Si simple densité demandée, traite</p> <p>Fixe la précompensation Ecriture automatique Protège le CC Initialise registres disque</p> <p>Compteur pour une piste Si fin du décomptage, erreur de secteur Lit status opération Si secteur pas trouvé, boucle Réinitialise status</p> <p>Attend fin de transmission Ecrit un mot de synchro Attend la fin de la transmission Ecrit le marquage de données</p> <p>Attend la fin de la transmission Lit un caractère Enregistre le caractère</p> <p>caractères lit status opération Attend la fin de l'opération</p> <p>Temporisation Si pas de vérification, sort Activation du lecteur Lecture automatique Initialise registres disque</p> <p>Compteur recherche marquage Compteur pour une piste Si fin du décomptage, erreur de vérification lit status opération Si secteur pas trouvé, boucle Réinitialise status disque</p> <p>Si fin de décomptage, erreur de piste</p>
--	---	--	--

Contrôleur externe CD90-351 version B (THMFC1) - 1ère banque

```
E40D A6 84      LDA      ,X
E40F 39         RTS
---- Recherche du secteur simple densité suivant
E410 4A         DECA      Recherche du
E411 26 BA      BNE      $E3CD  secteur suivant
E413 53         COMB      Erreur dans CC
E414 20 F7      BRA      $E40D  Sort du programme

    Fixe la précompensation pour double densité
E416 C1 39      CMPB     #$39
E418 2A 05      BPL      $E41F  Si au-delà de piste 57,
E41A CC 9F 20   LDD      #$9F20  séparateur actif
E41D 20 03      BRA      $E422
E41F CC 1F 22   LDD      #$1F22  Sinon, séparateur inactif
E422 A7 07      STA      $07,X  Fixe séparateur
E424 E7 01      STB      $01,X  Fixe 256 mots par secteur
E426 39         RTS

    Fixe la précompensation pour simple densité
E427 86 BF      LDA      #$BF  Séparateur actif à $3F
E429 A7 07      STA      $07,X  Fixe séparateur
E42B 6F 01      CLR      $01,X  Fixe 128 mots par secteur
E42D 39         RTS

    Calcul automatique du CRC identificateur en simple densité
E42E DE 4F      LDU      <$4F  Ptr buffer secteur
E430 10 8E 00 80 LDY      #$0080 128 caractères à traiter
E434 CC BF 84   LDD      #$BF84 Valeur CRC de départ
---- Calcul d'un CRC
E437 34 06      PSHS     B,A      Empile le CRC
E439 A6 E4      LDA      ,S      lit le CRCHigh
E43B A8 C0      EORA     ,U+     Masque les bits sortants
E43D 34 02      PSHS     A
E43F 44         LSRB     LSRB
E440 44         LSRB     LSRB
E441 44         LSRB     LSRB  Masque les bits entrants
E442 44         LSRB     LSRB
E443 A8 E4      EORA     ,S
E445 A7 E4      STA      ,S
E447 1F 89      TFR      A,B      Masque dans B
E449 54         LSRB     LSRB
E44A 54         LSRB     LSRB  Ajuste masque pour M*2^5
E44B 54         LSRB     LSRB
E44C 48         ASLA     ASLA
E44D 48         ASLA     ASLA  Ajuste masque
E44E 48         ASLA     ASLA  pour M*2^12
E44F 48         ASLA     ASLA
E450 E8 62      EORB     $02,S  Ajuste poids fort du
E452 E7 62      STB      $02,S  masque pour M*2^5
E454 1F 89      TFR      A,B      Poids fort pour M*2^5
E456 A8 62      EORA     $02,S  XOR poids fort pour M*2^12
E458 A7 61      STA      $01,S  Sauve poids fort CRC
E45A 58         ASLB     ASLB  Poids faible pour M*2^5
E45B E8 E4      EORB     ,S      XOR poids faible pour M*2^1
E45D E7 62      STB      $02,S  Sauve poids faible CRC
E45F 32 61      LEAS     $01,S  Rétablit la pile
E461 31 3F      LEAY     -$01,Y  Caractère
E463 26 D4      BNE      $E439  suivant
E465 35 06      PULS     A,B      Récupère le CRC
E467 39         RTS

    Routine de lecture d'un secteur
E468 17 00 AF   LBSR     $E51A  Recherche la piste
E46B 10 25 00 98 LBSC     $E507  Si erreur, sort
E46F 34 01      PSHS     CC      Protège CC
E471 96 58      LDA      <$58  Lit densité lecteur
E473 26 43      BNE      $E4B8  Si simple densité, traite
---- Lecture d'un secteur en double densité
E475 17 FF 9E   LBSR     $E416  Fixe la précompensation
E478 C6 1B      LDB      #$1B  Lecture automatique
E47A 17 FF 2F   LBSR     $E3AC  Initialise registres disque
---- Recherche le secteur
E47D 86 04      LDA      #$04  Compteur recherche marquage
E47F 10 8E 75 00 LDY      #$7500  Compteur pour une piste
E483 31 3F      LEAY     -$01,Y  Si fin du décomptage,
E485 10 27 00 7F LBQB     $E508  erreur de secteur
E489 E6 84      LDB      ,X      Lit status opération
E48B C5 02      BITB     #$02  Si secteur non trouvé,
E48D 27 F4      BEQ      $E483  boucle
---- Recherche du marquage de données
E48F E6 03      LDB      $03,X  Reset du status
E491 4A         DECA     DECA    Si fin de décomptage,
E492 10 27 00 7A LBQB     $E510  erreur de piste
E496 E6 84      LDB      ,X      Attend fin de
E498 2A FC      BPL      $E496  transmission
E49A E6 03      LDB      $03,X  Lit le caractère
E49C C1 FB      CMPB     #$FB  Si pas marquage de données,
E49E 26 F1      BNE      $E491  boucle
---- Lecture du secteur double densité
E4A0 4F         CLRA     CLRA    256 caractères à lire
E4A1 E6 84      LDB      ,X      Attend fin de
E4A3 2A FC      BPL      $E4A1  transmission
E4A5 E6 03      LDB      $03,X  Charge le caractère
E4A7 E7 C0      STB      ,U+     Ecrit le caractère
E4A9 4A         DECA     DECA    Pour tout
E4AA 26 F5      BNE      $E4A1  le secteur
```

```
---- Test de validité de la lecture
E4AC E6 84      LDB      ,X      Lit status opération
E4AE C5 08      BITB     #$08  Si pas fin opération,
E4B0 27 FA      BEQ      $E4AC  boucle
E4B2 C5 04      BITB     #$04  Si CRC incorrect,
E4B4 26 56      BNE      $E50C  erreur sur les données
E4B6 20 4B      BRA      $E503  Sort du programme

    Lecture d'un secteur en simple densité
E4B8 17 FF 6C   LBSR     $E427  Fixe la précompensation
E4BB 17 FF 01   LBSR     $E3BF  Recherche le secteur
E4BE 25 48      BCS      $E508  Si erreur, sort
E4C0 31 23      LEAY     $03,Y  Ajuste compteur tempo
E4C2 17 FD 6B   LBSR     $E230  Temporisation
---- Recherche du début des données
E4C5 C6 FB      LDB      #$FB  Fixe code de synchro
E4C7 E7 03      STB      $03,X  à trouver
E4C9 86 80      LDA      #$80 128 caractères à lire
E4CB C6 38      LDB      #$38  Passe en mode
E4CD E7 84      STB      ,X      "lecture synchro"
E4CF 10 8E 01 00 LDY      #$0100 Compteur pour recherche
E4D3 31 3F      LEAY     -$01,Y  Si fin du décomptage,
E4D5 27 39      BEQ      $E510  erreur de piste
E4D7 E6 84      LDB      ,X      Lit status opération
E4D9 C5 01      BITB     #$01  Si pas de synchro,
E4DB 27 F6      BEQ      $E4D3  boucle
---- Lecture du secteur simple densité
E4DD E6 03      LDB      $03,X  Reset du status
E4DF C6 28      LDB      #$28  Passage en mode
E4E1 E7 84      STB      ,X      "lecture données"
E4E3 E6 84      LDB      ,X      Attend fin de
E4E5 2A FC      BPL      $E4E3  transmission
E4E7 E6 03      LDB      $03,X  Charge le caractère
E4E9 E7 C0      STB      ,U+     Ecrit le caractère
E4EB 4A         DECA     DECA    Pour tout
E4ED 26 F5      BNE      $E4E3  le secteur
---- Vérification de la validité de la lecture
E4EE E6 84      LDB      ,X      Attend fin de
E4F0 2A FC      BPL      $E4EE  transmission
E4F2 E6 03      LDB      $03,X  Lit le CRCHigh/CRCLow
E4F4 34 04      PSHS     B      Empile le CRCHigh/CRCLow
E4F6 43         COMA     COMA    Si pas deuxième passe,
E4F7 26 F5      BNE      $E4EE  boucle
E4F9 17 FF 32   LBSR     $E42E  Calcule CRC pour les données
E4FB 1E 89      EYG      A,B      Ajuste CRC pour comparaison
E4FE 10 A3 E1   CMPD     ,S++    Si CRC incorrect,
E501 26 09      BNE      $E50C  erreur sur les données
E503 35 01      PULS     CC      Rétablit CC
E505 6F 84      CLR      ,X      Reset lecteur
E507 39         RTS

    Fixe l'erreur de secteur
E508 86 04      LDA      #$04  Code "Erreur de secteur"
E50A 20 06      BRA      $E512  Fixe l'erreur
---- Fixe l'erreur sur les données
E50C 86 08      LDA      $08  Code "Erreur sur les données"
E50E 20 02      BRA      $E512  Fixe l'erreur
---- Fixe l'erreur de piste
E510 86 02      LDA      #$02  Code "Erreur de piste"
---- Fixe l'erreur
E512 97 4E      STA      <$4E  Fixe l'erreur moniteur
E514 6F 84      CLR      ,X      Reset du lecteur
E516 35 01      PULS     CC      Rétablit le CC
E518 43         COMA     COMA    Erreur dans CC
E519 39         RTS

    Routine de recherche d'une piste
E51A 8D 06      BSR      $E522  Activation du lecteur
E51C 25 03      BCS      $E521  Si erreur, sort
E51E 8D 22      BSR      $E542  Déplace les têtes
E520 4F         CLRA     CLRA    Pas d'erreur dans CC
E521 39         RTS

    Activation du lecteur
E522 10 8E 80 00 LDY      #$8000  Compteur de test
E526 D6 57      LDB      <$57  Récupère bits de lecteur
E528 CA 04      ORB      $04
E52A E7 02      STB      $02,X  Stimulation
E52C C4 FB      ANDB     #$FB  du lecteur
E52E E7 02      STB      $02,X
E530 31 3F      LEAY     -$01,Y
E532 27 08      BEQ      $E53C  Si fin de décomptage,
E534 86 01      LDB      $01,X  "Lecteur non prêt"
E536 C4 02      ANDB     $02  Lit status disque
E538 27 EC      BEQ      $E526  Si lecteur non disponible,
E53A 4F         CLRA     CLRA    test suivant
E53B 39         RTS  Pas d'erreur dans CC

    Fixe l'erreur de lecteur
E53C 86 10      LDA      #$10  Code "Lecteur non prêt"
E53E 34 01      PSHS     CC      Ajuste la pile
E540 20 D0      BRA      $E512  Fixe l'erreur

    Déplacement des têtes sur la piste demandée
E542 17 FC E7   LBSR     $E22C  Temporisation
E545 8D 29      BSR      $E570  Ptr sur position de tête
```

Contrôleur externe CD90-351 version B (THMFC1) - 1ère banque

E547	E6	A4	LDB	,Y	Lit position actuelle	E564	17	FC	BF	LBSR	\$E226	Temporisation	
E549	96	57	LDA	<\$57	Lit bits de lecteur	E567	8D	07		BSR	\$E570	Ptr sur position de tête	
E54B	D0	4B	SUBB	<\$4B	Retire position demandée	E569	D6	4B		LDB	<\$4B	Lit numéro de piste	
E54D	27	18	BEQ	\$E567	Si = 0, sort	E56B	E7	06		STB	\$06,X	Initialise piste disque	
E54F	2A	03	BPL	\$E554	Si > à 0, recul de la tête	E56D	E7	A4		STB	,Y	Initialise position système	
E551	8A	20	ORA	#\$20	Sinon, avance de la tête et	E56F	39			RTS			
E553	50		NEGB		valeur absolue intervalle								
E554	A7	02	STA	\$02,X	Fixe sens de déplacement								
E556	8A	14	ORA	#\$14									
E558	A7	02	STA	\$02,X	Stimulation du	E570	31	8D	7A	DE	LEAY	>\$6052,PCR	Ptr positions têtes
E55A	84	EB	ANDA	#\$EB	déplacement pas à pas	E574	96	49		LDA	<\$49		Lit numéro de lecteur
E55C	A7	02	STA	\$02,X		E576	84	02		ANDA	#\$02		Ajuste offset
E55E	17	FC	CB	LBSR	\$E22C	E578	31	A6		LEAY	A,Y		Ajuste ptr sur position
E561	5A		DECB		Temporisation	E57A	39			RTS			
E562	26	F2	BNE	\$E556	Pour le nombre de								
					déplacements convenu								

Contrôleur externe CD90-351 version B (THMFC1) - 2ème banque

```

Identificateur de RomDisk (IDDISK)
E000 4D FCC *M* Contrôleur Thomson
E001 54 FCC *T* FAT 160 octets
E002 44 FCC *D* Double densité
E003 3A FCB $3A Checksum de l'identificateur

Indirections standard sur les opérations physiques
E004 16 00 24 LBRA $E02B Fonctions standard
E007 16 00 2C LBRA $E036 Lancement du boot
E00A 16 00 A2 LBRA $E0AF Formatage

---- Indirections standard sur les opérations logiques
E00D 16 00 3C LBRA $E04C Chargement de la FAT
E010 16 00 43 LBRA $E056 Ouverture d'un fichier
E013 16 00 4A LBRA $E060 Effacement d'un fichier
E016 16 00 51 LBRA $E06A Ecriture d'un secteur
E019 16 00 58 LBRA $E074 Création d'un fichier
E01C 16 00 5F LBRA $E07E Allocation d'un bloc
E01F 16 00 66 LBRA $E088 Initialise opération sur bloc
E022 16 00 6D LBRA $E092 Clôture d'écriture

---- Indirections standard sur les opérations physiques
E025 16 FF FD LBRA $E025 - Boucle à vide -
E028 16 FF FD LBRA $E028 - Boucle à vide -

---- Point d'entrée pour opération standard floppy
E02B 34 46 PSHS U,B,A
E02D 33 8C D4 LEAU <$E004,PCR Point d'entrée traitement
E030 86 00 LDA #$00 Banque 0 demandée
E032 8D 68 BSR $E09C Exécution programme
E034 35 C6 PULS A,B,U,PC

---- Point d'entrée pour lancement du boot floppy
E036 34 46 PSHS U,B,A
E038 33 8C CC LEAU <$E007,PCR Point d'entrée traitement
E03B 86 00 LDA #$00 Banque 0 demandée
E03D 8D 5D BSR $E09C Exécution programme
E03F 35 C6 PULS A,B,U,PC

---- Point d'entrée pour formatage floppy
E041 34 46 PSHS U,B,A
E043 33 8C C4 LEAU <$E00A,PCR Point d'entrée traitement
E046 86 01 LDA #$01 Banque 1 demandée
E048 8D 52 BSR $E09C Exécution programme
E04A 35 C6 PULS A,B,U,PC

---- Point d'entrée pour chargement de la FAT floppy
E04C 33 8C BE LEAU <$E00D,PCR Point d'entrée traitement
E04F 86 02 LDA #$02 Banque 2 demandée
E051 8D 49 BSR $E09C Exécution programme
E053 1F 30 TFR U,D Restaure le registre D
E055 39 RTS

---- Point d'entrée pour ouverture d'un fichier floppy
E056 33 8C B7 LEAU <$E010,PCR Point d'entrée traitement
E059 86 02 LDA #$02 Banque 2 demandée
E05B 8D 3F BSR $E09C Exécution programme
E05D 1F 30 TFR U,D Restaure le registre D
E05F 39 RTS

---- Point d'entrée pour effacement d'un fichier sur floppy
E060 33 8C B0 LEAU <$E013,PCR Point d'entrée traitement
E063 86 02 LDA #$02 Banque 2 demandée
E065 8D 35 BSR $E09C Exécution programme
E067 1F 30 TFR U,D Restaure le registre D
E069 39 RTS

---- Point d'entrée pour écriture d'un secteur sur floppy
E06A 33 8C A9 LEAU <$E016,PCR Point d'entrée traitement
E06D 86 02 LDA #$02 Banque 2 demandée
E06F 8D 2B BSR $E09C Exécution programme
E071 1F 30 TFR U,D Restaure le registre D
E073 39 RTS

---- Point d'entrée pour création d'un fichier sur floppy
E074 33 8C A2 LEAU <$E019,PCR Point d'entrée traitement
E077 86 02 LDA #$02 Banque 2 demandée
E079 8D 21 BSR $E09C Exécution programme
E07B 1F 30 TFR U,D Restaure le registre D
E07D 39 RTS

---- Point d'entrée pour allocation d'un bloc sur floppy
E07E 33 8C 9B LEAU <$E01C,PCR Point d'entrée traitement
E081 86 02 LDA #$02 Banque 2 demandée
E083 8D 17 BSR $E09C Exécution programme
E085 1F 30 TFR U,D Restaure le registre D
E087 39 RTS

---- Point d'entrée pour initialisation d'une opération sur un bloc
E088 33 8C 94 LEAU <$E01F,PCR Point d'entrée traitement
E08B 86 02 LDA #$02 Banque 2 demandée
E08D 8D 0D BSR $E09C Exécution programme
E08F 1F 30 TFR U,D Restaure le registre D
E091 39 RTS

---- Point d'entrée pour clôture d'écriture sur floppy
E092 33 8C 8D LEAU <$E022,PCR Point d'entrée traitement
E095 86 02 LDA #$02 Banque 2 demandée
E097 8D 03 BSR $E09C Exécution programme
E099 1F 30 TFR U,D Restaure le registre D
E09B 39 RTS

---- Module de commutation de banque ROM et d'exécution du traitement
E09C C6 00 LDB #$00 Banque 0
    
```

```

E09E 34 04 PSHS B pour retour
E0A0 A7 8D 07 34 STA >$E7D8,PCR Commute banque ROM
E0A4 AD C4 JSR ,U Exécute le programme
E0A6 1F 03 TFR D,U Protège le registre D
E0A8 35 04 PULS B Retour en
E0AA E7 8D 07 2A STB >$E7D8,PCR banque X
E0AE 39 RTS

***** SYSTEME D'EXPLOITATION PHYSIQUE FLOPPY *****

Routine de formatage floppy
E0AF 34 7F PSHS U,Y,X,DP,B,A,CC
E0B1 8D 50 BSR $E103 Initialise registres
E0B3 8D 61 BSR $E116 Fixe densité de formatage
E0B5 17 00 A3 LBSR $E15B Choix du lecteur
E0B8 25 42 BCS $E0FC Si erreur, sort
E0BA 17 00 C7 LBSR $E184 Recherche la piste 0
E0BD 25 3B BCS $E0FA Si erreur, sort
E0BF 17 00 B6 LBSR $E178 Teste protection disquette
E0C2 25 36 BCS $E0FA Si erreur, sort

---- Formatage de la piste
E0C4 34 01 PSHS CC Protège flag interruptions
E0C6 1A 50 ORCC #$50 Gel des interruptions
E0C8 17 01 02 LBSR $E1CD Prépare table entrelacements
E0CB 86 02 LDA #$02 | 2 tentatives
E0CD 34 02 PSHS A maximum
E0CF 17 01 2D LBSR $E1FF Crée liste des paramètres
E0D2 17 01 57 LBSR $E22C Formate la piste
E0D5 24 0B BCC $E0E2 Si pas d'erreur, suivante

---- Sortie si erreur de formatage
E0D7 6A E4 DEC ,S Tentative
E0D9 26 F4 BNE $E0CF suivante
E0DB 35 02 PULS A Rétablit la pile
E0DD 35 01 PULS CC Dégel des interruptions
E0DF 43 COMA Erreur dans CC
E0E0 20 18 BRA $E0FA Sortie du programme

---- Piste suivante
E0E2 35 02 PULS A Rétablit la pile
E0E4 35 01 PULS CC Dégel des interruptions
E0E6 96 4B LDA <$4B Lit numéro de piste
E0E8 81 4F CMPA #$4F Si piste 79,
E0EA 27 05 BEQ $E0F1 exit
E0EC 17 03 44 LBSR $E433 Déplace la tête d'une piste
E0EF 20 D3 BRA $E0C4 Piste suivante

---- Clôt le formatage
E0F1 17 03 56 LBSR $E44A Initialise piste catalogue
E0F4 25 04 BCS $E0FA Si erreur, sort
E0F6 86 50 LDA #$50 Pose
E0F8 97 4B STA <$4B piste 80
E0FA 8D 77 BSR $E173 Extinction du moteur
E0FC 35 02 PULS A Récupère CC
E0FE 1E 8A EXG A,CC Dégel des interruptions
E100 44 LSR A Bit d'erreur dans CC
E101 35 FE PULS A,B,DP,X,Y,U,PC

Initialise registres
E103 1F 50 TFR PC,D PC dans D
E105 84 70 ANDA $70 Fixe le DP
E107 1F 8B TFR A,DP MO=$20 TO=$60
E109 96 48 LDA <$48 Lit code de commande
E10B 84 80 ANDA $80 Garde bit de
E10D 97 48 STA <$48 vérification
E10F 30 8D 06 BD LEAX >$E7D0,PCR Ptr registres disque
E113 0F 4E CLR <$4E Efface code d'erreur
E115 39 RTS

Fixe la densité de formatage selon le type de logiciel
E116 34 10 PSHS X
E118 CE E9 41 LDU #$E941 Ptr sur paramètres logiciels
E11B 8E 00 03 LDX #$0003 Ptr sur 3ème lettre applic.
E11E C6 04 LDB #$04 4 caractères à comparer
E120 A6 84 LDA ,X Si les caractères sont
E122 A1 C4 CMPA ,U identiques, décrémente
E124 27 09 BEQ $E12F les compteurs
E126 5C INCB Sinon, paramètres
E127 33 C5 LEAU B,U suivants
E129 A6 C4 LDA ,U Si pas fin de liste,
E12B 26 EE BNE $E11B nouvelle comparaison
E12D 20 10 BRA $E13F Sinon, sort
E12F 30 1F LEAX -$01,X Pour lettre
E131 26 03 BNE $E136 suivante
E133 8E 00 1A LDX #$001A Pointe checksum application
E136 33 41 LEAU $01,U Caractère
E138 5A DECB suivant
E139 26 E5 BNE $E120
E13B A6 C4 LDA ,U Fixe la densité
E13D 97 58 STA <$58 pour le formatage
E13F 35 90 PULS X,PC
    
```


Contrôleur externe CD90-351 version B (THMFC1) - 2ème banque

E664	C6 24	LDB	#\$24		Passé en écriture	E693	2A FC	BPL	#\$691		transmission
E666	E7 84	STB	,X		données FM	E695	35 04	PULS	B		Récupère CRCHigh/Low
E668	E6 84	LDB	,X		Attend fin de	E697	E7 03	STB	03,X		Sauve CRCHigh/Low
E66A	2A FC	BPL	#\$668		transmission	E699	43	COMA			Si pas deuxième passe,
E66C	6F 03	CLR	03,X		Ecrit un caractère à 0	E69A	26 F5	BNE	#\$691		boucle
E66E	4A	DECA			Pour les 6	---- Clôture du secteur simple densité					
E66F	26 F7	BNE	#\$668		caractères	E69C	E6 84	LDB	,X		Attend fin de
---- Ecriture du marquage de données											
E671	E6 84	LDB	,X		Attend fin de	E69E	2A FC	BPL	#\$69C		transmission
E673	2A FC	BPL	#\$671		transmission	E6A0	6F 03	CLR	03,X		Ecrit un mot à 0
E675	CC FB C7	LDD	#\$FBC7		Passé en mode "synchro"	E6A2	E6 84	LDB	,X		Attend fin de
E678	ED 03	STD	03,X		et écrit marquage	E6A4	2A FC	BPL	#\$6A2		transmission
---- Ecriture du secteur simple densité											
E67A	C6 FF	LDB	#\$FF		Passé en mode "données"	---- Passage du reste du secteur simple densité					
E67C	A6 84	LDA	,X		Attend fin de	E6A6	C6 20	LDB	#\$20		Passé en mode
E67E	2A FC	BPL	#\$67C		transmission	E6A8	E7 84	STB	,X		"lecture données"
E680	A6 C0	LDA	,U+		Lit premier caractère	E6AA	E6 84	LDB	,X		Attend fin de
E682	ED 03	STD	03,X		Enregistre premier caractère	E6AC	2A FC	BPL	#\$6AA		transmission
E684	86 7F	LDA	#\$7F		127 caractères restants	E6AE	E6 03	LDB	03,X		Lit le caractère à perte
E686	E6 84	LDB	,X		Attend fin de	E6B0	43	COMA			2 fois de
E688	2A FC	BPL	#\$686		transmission	E6B1	26 F7	BNE	#\$6AA		suite
E68A	E6 C0	LDB	,U+		Lit le caractère	E6B3	EC 7E	LDD	-\$02,S		Relit le CRC
E68C	E7 03	STB	03,X		Sauve le caractère	E6B5	35 01	PULS	CC		Dégel des interruptions
E68E	4A	DECA			Pour tout	E6B7	6F 84	CLR	,X		Reset lecteur
E68F	26 F5	BNE	#\$686		le secteur	E6B9	0D 48	TST	<\$48		Si vérification
---- Ecriture du CRC de données											
E691	E6 84	LDB	,X		Attend fin de	E6BB	10 2B FE 37	LBMI	#\$4F6		demandée, traite
						E6BF	39	RTS			

Contrôleur externe CD90-351 version B (THMFC1) - 3ème banque

```
Identificateur de RomDisk (IDDISK)
E000 4D FCC *M* Contrôleur Thomson
E001 54 FCC *T* FAT 160 octets
E002 44 FCC *D* Double densité
E003 3A FCB $3A Checksum de l'identificateur

----- Indirections standard sur les opérations physiques
E004 16 00 24 LBRA $E02B Fonctions standard
E007 16 00 2C LBRA $E036 Lancement du boot
E00A 16 00 34 LBRA $E041 Formatage
---- Indirections standard sur les opérations logiques
E00D 16 01 01 LBRA $E111 Chargement de la FAT
E010 16 01 84 LBRA $E197 Ouverture d'un fichier
E013 16 01 63 LBRA $E179 Effacement d'un fichier
E016 16 01 3F LBRA $E158 Ecriture d'un secteur
E019 16 01 E3 LBRA $E1FF Création d'un fichier
E01C 16 02 39 LBRA $E258 Allocation d'un bloc
E01F 16 02 7B LBRA $E29D Initialise opération sur bloc
E022 16 00 95 LBRA $E0BA Clôture d'écriture
---- Indirections standard sur les opérations physiques
E025 16 FF FD LBRA $E025 - Boucle à vide -
E028 16 FF FD LBRA $E028 - Boucle à vide -

---- Point d'entrée pour opération standard floppy
E02B 34 46 PSHS U,B,A
E02D 33 8C D4 LEAU <$E004,PCR Point d'entrée traitement
E030 86 00 LDA #$00 Banque 0 demandée
E032 8D 68 BSR $E09C Exécution programme
E034 35 C6 PULS A,B,U,PC
---- Point d'entrée pour lancement du boot floppy
E036 34 46 PSHS U,B,A
E038 33 8C CC LEAU <$E007,PCR Point d'entrée traitement
E03B 86 00 LDA #$00 Banque 0 demandée
E03D 8D 5D BSR $E09C Exécution programme
E03F 35 C6 PULS A,B,U,PC
---- Point d'entrée pour formatage QDD
E041 34 46 PSHS U,B,A
E043 33 8C C4 LEAU <$E00A,PCR Point d'entrée traitement
E046 86 01 LDA #$01 Banque 1 demandée
E048 8D 52 BSR $E09C Exécution programme
E04A 35 C6 PULS A,B,U,PC
---- Point d'entrée pour chargement de la FAT floppy
E04C 33 8C BE LEAU <$E00D,PCR Point d'entrée traitement
E04F 86 02 LDA #$02 Banque 2 demandée
E051 8D 49 BSR $E09C Exécution programme
E053 1F 30 TFR U,D Restaure le registre D
E055 39 RTS
---- Point d'entrée pour ouverture d'un fichier floppy
E056 33 8C B7 LEAU <$E010,PCR Point d'entrée traitement
E059 86 02 LDA #$02 Banque 2 demandée
E05B 8D 3F BSR $E09C Exécution programme
E05D 1F 30 TFR U,D Restaure le registre D
E05F 39 RTS
---- Point d'entrée pour effacement d'un fichier sur floppy
E060 33 8C B0 LEAU <$E013,PCR Point d'entrée traitement
E063 86 02 LDA #$02 Banque 2 demandée
E065 8D 35 BSR $E09C Exécution programme
E067 1F 30 TFR U,D Restaure le registre D
E069 39 RTS
---- Point d'entrée pour écriture d'un secteur sur floppy
E06A 33 8C A9 LEAU <$E016,PCR Point d'entrée traitement
E06D 86 02 LDA #$02 Banque 2 demandée
E06F 8D 2B BSR $E09C Exécution programme
E071 1F 30 TFR U,D Restaure le registre D
E073 39 RTS
---- Point d'entrée pour création d'un fichier sur floppy
E074 33 8C A2 LEAU <$E019,PCR Point d'entrée traitement
E077 86 02 LDA #$02 Banque 2 demandée
E079 8D 21 BSR $E09C Exécution programme
E07B 1F 30 TFR U,D Restaure le registre D
E07D 39 RTS
---- Point d'entrée pour allocation d'un bloc sur floppy
E07E 33 8C 9B LEAU <$E01C,PCR Point d'entrée traitement
E081 86 02 LDA #$02 Banque 2 demandée
E083 8D 17 BSR $E09C Exécution programme
E085 1F 30 TFR U,D Restaure le registre D
E087 39 RTS
---- Point d'entrée pour initialisation d'un opération sur un bloc
E088 33 8C 94 LEAU <$E01F,PCR Point d'entrée traitement
E08B 86 02 LDA #$02 Banque 2 demandée
E08D 8D 0D BSR $E09C Exécution programme
E08F 1F 30 TFR U,D Restaure le registre D
E091 39 RTS
---- Point d'entrée pour clôture d'écriture sur floppy
E092 33 8C 8D LEAU <$E022,PCR Point d'entrée traitement
E095 86 02 LDA #$02 Banque 2 demandée
E097 8D 03 BSR $E09C Exécution programme
E099 1F 30 TFR U,D Restaure le registre D
E09B 39 RTS
---- Module de commutation de banque ROM et d'exécution du traitement
E09C C6 02 LDB #$02 Banque 2
E09E 34 04 PSHS B pour retour
E0A0 A7 8D 07 34 STA >$E7D8,PCR Commute banque ROM
```

```
E0A4 AD C4 JSR ,U Exécute le programme
E0A6 1F 03 TFR D,U Protège le registre D
E0A8 35 04 PULS B Retour en
E0AA E7 8D 07 2A STB >$E7D8,PCR banque X
E0AE 39 RTS

***** SYSTEME D'EXPLOITATION LOGIQUE *****

Nom de fichier temporaire pour la sauvegarde
E0AF 53 43 52 41 FCC "SCRATCH.DOS"
E0B3 54 43 48 20
E0B7 44 4F 53

Clôture d'écriture
E0BA D6 F0 LDB <$F0 Lit code commande
E0BC C1 02 CMPB #$02 Si écrasement demandé,
E0BE 27 25 BEQ $E0E5 écrit FAT seule
E0C0 0A F0 DEC <$F0 Passe en sauvg. sans ecr.
E0C2 17 00 D2 LBSR $E197 Recherche fichier courant
E0C5 25 31 BCS $E0F8 Si erreur disque, sort
E0C7 5D TSTB Si l'entrée n'existe pas,
E0C8 27 05 BEQ $E0CF pas d'effacement
E0CA 17 00 AC LBSR $E179 Effacement du fichier
E0CD 25 29 BCS $E0F8 Si erreur, sort
E0CF 0C F0 INC <$F0 Retour en sauvg avec écrasmt
E0D1 17 00 C3 LBSR $E197 Cherche fichier "SCRATCH.DOS"
E0D4 25 22 BCS $E0F8 Si erreur, sort
E0D6 C6 0A LDB #$0A
E0D8 9E E7 LDX <$E7 Recopie le nom
E0DA A6 85 LDA B,X du fichier
E0DC A7 A5 STA B,Y courant à la place
E0DE 5A DECB de "SCRATCH.DOS"
E0DF 2C F9 BGE $E0DA
E0E1 8D 75 BSR $E158 Ecrit le secteur catalogue
E0E3 25 13 BCS $E0F8 Si erreur de disque, sort

---- Sauvegarde de la FAT
E0E5 86 02 LDA #$02
E0E7 97 4C STA <$4C Programme la
E0E9 C6 14 LDB #$14 sauvegarde
E0EB 4F CLRA de la FAT
E0EC DD 4A STD <$4A
E0EE 8D 4C BSR $E13C Rétablissement de FAT swapée
E0F0 9F 4F STX <$4F Sauvegarde
E0F2 8D 64 BSR $E158 de la FAT
E0F4 25 02 BCS $E0F8 Si erreur de disque, sort
E0F6 0F F0 CLR <$F0 Clôture du fichier effectuée
E0F8 39 RTS

Swap d'une FAT double densité
E0F9 34 07 PSHS B,A,CC
E0FB 9E ED LDX <$ED
E0FD 30 89 00 80 LEAX >$0080,X Pointeur
E101 A6 1F LDA -$01,X en milieu de FAT
E103 B6 88 7F LDB <$7F,X Echange l'octet précédent
E106 A7 88 7F STA <$7F,X avec le 127ème octet
E109 E7 82 STB ,X Déplace pointeur à rebours
E10B 9C ED CMPX <$ED Fin de FAT ?
E10D 26 F2 BNE $E101 Non, échange suivant
E10F 35 87 PULS CC,A,B,PC

Chargement de la FAT
E111 DC ED LDD <$ED Pointeur sur FAT swapée
E113 0D 55 TST <$55 FAT système initialisé ?
E115 26 02 BNE $E119 Non, initialise avec
E117 DD 55 STD <$55 pointeur sur FAT swapée
E119 93 55 SUBD <$55 Si la différence entre les
E11B C1 80 CMPB #$80 deux est de 128 octets,
E11D 27 04 BEQ $E123 prend le pointeur système
E11F 9E ED LDX <$ED Récupère le pointeur de
E121 20 02 BRA $E125 début de secteur de FAT
E123 9E 55 LDX <$55 Initialise les
E125 9F ED STX <$ED pointeurs pour charger
E127 9F 4F STX <$4F le secteur
E129 86 02 LDA #$02 Chargement
E12B 8D 31 BSR $E15E de la FAT
E12D 25 C9 BCS $E0F8 Si erreur disque, sort
E12F 0D 58 TST <$58 Contrôleur double densité ?
E131 26 08 BNE $E13B Oui, swap
E133 8D C4 BSR $E0F9 de la FAT
E135 30 89 00 80 LEAX >$0080,X et pointeur courant
E139 9F ED STX <$ED en milieu de FAT
E13B 39 RTS

Rétablissement de la FAT swapée
E13C 9E ED LDX <$ED
E13E 0D 58 TST <$58
E140 26 F9 BNE $E13B S'il s'agit
E142 30 88 80 LEAX <-$80,X d'une FAT
E145 8D F2 BSR $E139 double densité,
E147 20 B0 BRA $E0F9 repositionne le pointeur
de FAT en début de secteur
et rétablit la FAT swapée
```


Contrôleur externe CD90-351 version B (THMFC1) - 3ème banque

E296 6F A5 CLR B,Y Valide la place du bloc
 E298 5A DECB | Ajuste et stocke
 E299 D7 F9 STB <\$F9 | le numéro de bloc
 E29B 4F CLRA | Sort sans erreur
 E29C 39 RTS |

Initialisation d'une opération sur un bloc
 E29D D6 F6 LDB <\$F6 Numéro de bloc courant
 E29F 4F CLRA | Calcule le

E2A0 54 LSRB
 E2A1 DD FB STD <\$FB
 E2A3 4C INCA
 E2A4 97 F5 STA <\$F5
 E2A6 24 02 BCC \$E2AA
 E2A8 86 09 LDA #\$09
 E2AA 97 FA STA <\$FA
 E2AC 39 RTS

| numéro
 | de piste
 | Initialisation du compteur
 | de secteur
 | Selon la parité du numéro
 | de bloc, le premier secteur
 | est le secteur 1 ou 9

Contrôleur externe CD90-351 version B (THMFC1) - 4ème banque

```

Identificateur de RomDisk (IDDISK)
E000 4D FCC "M"      Contrôleur Thomson
E001 4B FCC "K"      FAT 160 octets
E002 43 FCC "C"      Double densité
E003 30 FCB $30      Checksum de l'identificateur

----- Indirections standard sur les opérations physiques
E004 16 00 F3 LBRA $E0FA Fonctions standard QDD
E007 16 00 A5 LBRA $E0AF Lancement du boot QDD
E00A 16 03 A7 LBRA $E3B4 Formatage QDD
---- Indirections standard sur les opérations logiques
E00D 16 05 5A LBRA $E56A Chargement de la FAT
E010 16 05 84 LBRA $E597 Ouverture d'un fichier
E013 16 06 7E LBRA $E694 Effacement d'un fichier
E016 16 05 71 LBRA $E58A Ecriture d'un secteur
E019 16 05 DA LBRA $E5F6 Création d'un fichier
E01C 16 06 31 LBRA $E650 Allocation d'un bloc
E01F 16 06 91 LBRA $E6B3 Initialise opération sur bloc
E022 16 05 08 LBRA $E52D Clôture d'écriture
---- Indirections standard sur les opérations physiques
E025 16 00 CA LBRA $E0F2 Fonctions standard QDD
E028 16 00 D4 LBRA $E0FF Fonctions système QDD

---- Point d'entrée pour opération standard floppy
E02B 34 46 PSHS U,B,A
E02D 33 8C D4 LEAU <$E004,PCR Point d'entrée traitement
E030 86 00 LDA #$00 Banque 0 demandée
E032 8D 68 BSR $E09C Exécution programme
E034 35 C6 PULS A,B,U,PC
---- Point d'entrée pour lancement du boot floppy
E036 34 46 PSHS U,B,A
E038 33 8C CC LEAU <$E007,PCR Point d'entrée traitement
E03B 86 00 LDA #$00 Banque 0 demandée
E03D 8D 5D BSR $E09C Exécution programme
E03F 35 C6 PULS A,B,U,PC
---- Point d'entrée pour formatage QDD
E041 34 46 PSHS U,B,A
E043 33 8C C4 LEAU <$E00A,PCR Point d'entrée traitement
E046 86 01 LDA #$01 Banque 1 demandée
E048 8D 52 BSR $E09C Exécution programme
E04A 35 C6 PULS A,B,U,PC
---- Point d'entrée pour chargement de la FAT floppy
E04C 33 8C BE LEAU <$E00D,PCR Point d'entrée traitement
E04F 86 02 LDA #$02 Banque 2 demandée
E051 8D 49 BSR $E09C Exécution programme
E053 1F 30 TFR U,D Restaure le registre D
E055 39 RTS
---- Point d'entrée pour ouverture d'un fichier floppy
E056 33 8C B7 LEAU <$E010,PCR Point d'entrée traitement
E059 86 02 LDA #$02 Banque 2 demandée
E05B 8D 3F BSR $E09C Exécution programme
E05D 1F 30 TFR U,D Restaure le registre D
E05F 39 RTS
---- Point d'entrée pour effacement d'un fichier sur floppy
E060 33 8C B0 LEAU <$E013,PCR Point d'entrée traitement
E063 86 02 LDA #$02 Banque 2 demandée
E065 8D 35 BSR $E09C Exécution programme
E067 1F 30 TFR U,D Restaure le registre D
E069 39 RTS
---- Point d'entrée pour écriture d'un secteur sur floppy
E06A 33 8C A9 LEAU <$E016,PCR Point d'entrée traitement
E06D 86 02 LDA #$02 Banque 2 demandée
E06F 8D 2B BSR $E09C Exécution programme
E071 1F 30 TFR U,D Restaure le registre D
E073 39 RTS
---- Point d'entrée pour création d'un fichier sur floppy
E074 33 8C A2 LEAU <$E019,PCR Point d'entrée traitement
E077 86 02 LDA #$02 Banque 2 demandée
E079 8D 21 BSR $E09C Exécution programme
E07B 1F 30 TFR U,D Restaure le registre D
E07D 39 RTS
---- Point d'entrée pour allocation d'un bloc sur floppy
E07E 33 8C 9B LEAU <$E01C,PCR Point d'entrée traitement
E081 86 02 LDA #$02 Banque 2 demandée
E083 8D 17 BSR $E09C Exécution programme
E085 1F 30 TFR U,D Restaure le registre D
E087 39 RTS
---- Point d'entrée pour initialisation d'un opération sur bloc
E088 33 8C 94 LEAU <$E01F,PCR Point d'entrée traitement
E08B 86 02 LDA #$02 Banque 2 demandée
E08D 8D 0D BSR $E09C Exécution programme
E08F 1F 30 TFR U,D Restaure le registre D
E091 39 RTS
---- Point d'entrée pour clôture d'écriture sur floppy
E092 33 8C 8D LEAU <$E022,PCR Point d'entrée traitement
E095 86 02 LDA #$02 Banque 2 demandée
E097 8D 03 BSR $E09C Exécution programme
E099 1F 30 TFR U,D Restaure le registre D
E09B 39 RTS
---- Module de commutation de banque ROM et d'exécution du traitement
E09C C6 00 LDB #$00 Banque 0
E09E 34 04 PSHS B pour retour
E0A0 A7 8D 07 34 STA >$E7D8,PCR Commute banque ROM

```

```

E0A4 6E C4 JMP ,U Exécute le programme
E0A6 1F 03 TFR D,U Protège le registre X
E0A8 35 04 PULS B Retour en
E0AA E7 8D 07 2A STB >$E7D8,PCR banque X
E0AE 39 RTS

***** SYSTEME D'EXPLOITATION PHYSIQUE DU QDD *****

Lancement du boot QDD
E0AF 17 00 66 LBSR $E118 Initialisation registres
E0B2 0F 49 CLR <$49 Lecteur 0 demandé
E0B4 17 00 7A LBSR $E131 Choix du lecteur
E0B7 17 00 AE LBSR $E168 Initialise en simple densité
E0BA CC 00 44 LDD #$0044 Secteur 68 (Boot QDD)
E0BD DD 4C STD <$4C sélectionné
E0BF 33 8D 81 3D LEAU >$6200,PCR Ptr buffer secteur
E0C3 DF 4F STU <$4F sélectionné
E0C5 17 01 DC LBSR $E2A4 Lit le secteur QDD
E0C8 DE 4F LDU <$4F Ptr buffer secteur
E0CA 25 20 BCS $E0EC Si erreur, sort
E0CC 17 00 7F LBSR $E14E Eteint le lecteur
E0CF 31 8D 81 AC LEAY >$627F,PCR Limite du décodage
E0D3 34 20 PSHS Y en pile
E0D5 86 55 LDA #$55 Checksum de départ
E0D7 6A C4 DEC ,U Décode les
E0D9 63 C4 COM ,U données
E0DB AB C0 ADDA ,U+ Calcule le checksum
E0DD 11 A3 E4 CMPU ,S Si pas fin du décodage,
E0E0 26 F5 BNE $E0D7 boucle
E0E2 35 20 PULS Y Rétablit la pile
E0E4 A1 C4 CMPA ,U Si checksum incorrect,
E0E6 26 04 BNE $E0EC lance l'application
E0E8 6E 8D 81 14 JMP >$6200,PCR Sinon, exécute le boot
---- Sortie si erreur de boot
E0EC 0F 80 CLR <$80 Flag "contrôleur absent"
E0EE 6E 9F 00 1E JMP >[$001E] lance l'application à froid

Opérations standard avec numéro de secteur QDD réel
E0F2 34 7F PSHS U,Y,X,DP,B,A,CC
E0F4 8D 22 BSR $E118 Initialise registres disque
E0F6 86 FF LDA #$FF Flag pour opérations
E0F8 20 05 BRA $E0FF QDD avec secteur réel

Opération standard avec numéro de secteur QDD interprété
E0FA 34 7F PSHS U,Y,X,DP,B,A,CC
E0FB 8D 1A BSR $E118 Initialise registres disque
E0FC 4F CLRA Flag de secteur interprété
---- Opération système avec numéro de secteur QDD interprété
E0FF 97 4A STA <$4A Ajourne le flag/n° de piste
E101 DC 4B LDD <$4B Protection de
E103 34 06 PSHS B,A DKTRK et DKSEC
E105 8D 2A BSR $E131 Choix du lecteur
E107 25 04 BCS $E10D Si erreur, sort
E109 8D 4C BSR $E157 Exécute l'opération
E10B 8D 41 BSR $E14E Eteint le moteur QDD
---- Sortie du programme
E10D 35 06 PULS A,B Restaure DKTRK
E10F DD 4B STD <$4B et DKSEC
E111 35 02 PULS A Dépile le CC
E113 1E 8A EXG A,CC Restaure le CC
E115 44 LSRA Bit d'erreur dans CC
E116 35 FE PULS A,B,DP,X,Y,U,PC

Initialisation des registres disque
E118 1F 50 TFR PC,D PC dans D
E11A 84 70 ANDA #$70 Initialise le DP
E11C 1F 8B TFR A,DP MO=$20 TO=$60
E11E 30 8D 06 AE LEAX >$E7D0,PCR Ptr sur registres disques
E122 0F 4E CLR <$4E Efface code d'erreur
E124 6F 84 CLR ,X Reset lecteur
E126 86 CE LDA #$CE Séparateur de donnée
E128 A7 07 STA $07,X avec compteur de $4E
E12A 86 FF LDA #$FF Programme le mode données
E12C A7 04 STA $04,X le mode "donnée"
E12E 6F 01 CLR $01,X Pas de précompensation
E130 39 RTS

Sélection des bits de lecteur QDD
E131 D6 49 LDB <$49 Si face impaire
E133 C5 FD BITB #$FD demandée, erreur
E135 26 12 BNE $E149 "lecteur non prêt"
E137 33 8D 00 18 LEAU >$E153,PCR Table des bits de lecteurs
E13B A6 C5 LDA B,U Sélectionne
E13D A7 02 STA $02,X le lecteur
E13F 97 57 STA <$57 adéquat.
E141 E6 01 LDB $01,X Si disque
E143 C5 40 BITB #$40 QDD absent, erreur
E145 26 02 BNE $E149 "lecteur non prêt"
E147 4F CLRA Pas d'erreur dans CC
E148 39 RTS

```


Contrôleur externe CD90-351 version B (THMFC1) - 4ème banque

E687	CB 28	ADDB	#40	Rétablit l'offset	E6A9	6A A5	DEC	B,Y	courant
E688	31 A8 D8	LEAY	<-40,Y	Rétablit le pointeur	E6AB	1F 89	TFR	A,B	répertoriés
E68C	6F A5	CLR	B,Y	Réserve la place du bloc	E6AD	81 C0	CMPA	#SC0	dans la FAT
E68E	5A	DECB		Ajuste et stocke	E6AF	25 F3	BLO	\$E6A4	
E68F	D7 F9	STB	<\$F9	le numéro de bloc	E6B1	20 DE	BRA	\$E691	Pas d'erreur dans CC
E691	16 FE E3	LBRA	\$E577	Sort sans erreur	Initialisation pour opération sur un bloc				
Effacement du fichier courant					E6B3	D6 F6	LDB	<\$F6	Numéro de bloc courant
E694	A6 2D	LDA	\$0D,Y	Premier bloc du fichier	E6B5	4F	CLRA		Calcule le
E696	97 F6	STA	<\$F6	sélectionné	E6B6	54	LSRB		numéro
E698	6F A4	CLR	,Y	Elimine le nom du fichier	E6B7	DD FB	STD	<\$FB	de piste
E69A	17 FE ED	LBSR	\$E58A	Ecrit le secteur catalogue	E6B9	4C	INCA		Initialisation du compteur
E69D	25 8D	BCS	\$E62C	Si erreur disque, sort	E6BA	97 F5	STA	<\$F5	de secteur
E69F	10 9E ED	LDY	<\$ED		E6BC	24 02	BCC	\$E6C0	Selon la parité du numéro
E6A2	D6 F6	LDB	<\$F6	Efface	E6BE	86 09	LDA	#S09	de bloc, le premier secteur
E6A4	5C	INCB		(met à \$FF)	E6C0	97 FA	STA	<\$FA	est le secteur 1 ou 9
E6A5	A6 A5	LDA	B,Y	tous les blocs	E6C2	39	RTS		
E6A7	6F A5	CLR	B,Y	du fichier					

Contrôleur externe CD90-640

```

E408 1A 50      ORCC  #$50      Gel des interruptions
---- Création de la table de formatage
E40A 31 8D 01 08 LEAY  $E516,PCR Table pour double densité
E40E A6 8D 03 C6 LDA   $E7D8,PCR
E412 85 80      BITA  #$80      Si simple densité, table
E414 27 04      BEQ   $E41A     pour simple densité
E416 31 8D 01 1E LEAY  $E538,PCR
E41A 4F         CLRA
E41B DE 4F      LDU   <$4F     Pour 16 bits
E41D E6 A0      LDB   ,Y+      Ptr buffer secteur
E41F 27 10      BEQ   $E431     Lit un paramètre
E421 1F 01      TFR   D,X      Si fin de table, exit
E423 E6 A0      LDB   ,Y+      Nombre d'itérations dans X
E425 E7 C0      STB   ,U+      Ecrit la valeur
E427 AF C1      STX   ,U++     de remplissage
E429 30 8D 00 3D LEAX  $E46A,PCR Ecrit itération
E42D AF C1      STX   ,U++     Programme de remplissage
E42F 20 EC      BRA   $E41D     dans liste formatage
E431 DE 4F      LDU   <$4F     Paramètre suivant
E433 96 4B      LDA   <$4B     Ptr buffer secteur
E435 A7 4F      STA   $0F,U    Lit n° de piste
E437 30 8D 00 3B LEAX  $E476,PCR Ecrit n° de piste
E43B AF C8 2B   STX   $2B,U    Inclut la mise à jour
E43E 30 8D 00 3F LEAX  $E481,PCR du numéro de secteur
E442 AF C8 4E   STX   $4E,U    Inclut le test
                                     de f21(u)-38(t )-27(le)21( )-3( )-38(AF)-27( C8 4)-44(E )-44( S)-44(TX )-30( $)-30(4E)-27(,U )-30( )-27( )-44( | )-43

```

Contrôleur externe CD90-640

E58C	25 13	BCS	5E5A1	Si erreur de disque, sort	E62B	97 F6	STA	<\$F6	du fichier
E58E	86 02	LDA	5\$02		E62D	0F F5	CLR	<\$F5	Compteur de secteur à 0
E590	97 4C	STA	<\$4C	Programme la sauvegarde de la FAT	E62F	AE 2E	LDX	\$0E,Y	Stocke le nombre d'octets dans le dernier secteur
E592	C6 14	LDB	5\$14		E631	9F F7	STX	<\$F7	Stocke pointeur d'entrée
E594	4F	CLRA			E636	D7 F9	STB	<\$F9	Numéro secteur catalogue
E595	DD 4A	STD	<\$4A		E638	16 FF 73	LBRA	5E5AE	Sort sans erreur
E597	DC ED	LDD	<\$ED	Initialise ptr sur FAT					Création d'un fichier
E599	DD 4F	STD	<\$4F	Sauve la FAT	E63B	10 9E ED	LDY	<\$ED	Bloc libre dans la FAT ?
E59B	8D 24	BSR	5E5C1	Si erreur de disque, sort	E63E	8D 6E	BSR	5E6AE	Non, sort avec erreur
E59D	25 02	BCS	5E5A1	Clôture du fichier effectuée	E640	25 8E	BCS	5E5D0	Stocke le numéro de bloc
E59F	0F F0	CLR	<\$F0		E642	D7 F6	STB	<\$F6	Charge ler secteur catalogue
E5A1	39	RTS			E644	17 FF 69	LBSR	5E5B0	Si erreur, sort
					E647	25 F7	BSR	5E640	Pointeur sur buffer catalogue
					E649	10 9E E9	LDY	<\$E9	Nombre d'entrées par secteurs
					E64C	8D BB	BSR	5E609	Si l'entrée est effacée, prend la place.
					E64E	E6 A4	LDB	Y	Code d'erreur Disque plein
					E650	27 20	BEQ	5E672	Si l'entrée est libre, prend la place
					E652	86 05	LDA	5\$05	
					E654	C1 FF	CMPB	5\$FF	
					E656	27 1A	BEQ	5E672	Si toujours pas d'entrée trouvée, X passe à l'entrée suivante
					E658	31 A8 20	LEAY	\$20,Y	Si tous les secteurs de catalogue ont été passés en revue, erreur "disque plein"
					E65B	30 1F	LEAX	-\$01,	
					E65D	26 EF	BNE	5E64E	
					E65F	0C 4C	INC	<\$4C	
					E661	96 4C	LDA	<\$4C	
					E663	81 10	CMPA	5\$10	
					E665	22 07	BHI	5E66E	
					E667	17 F9 9A	LBSR	5E004	
					E66A	86 03	LDA	5\$03	
					E66C	20 D9	BRA	5E647	
									Sortie si erreur "disque plein"
					E66E	86 05	LDA	5\$05	Code "disque plein"
					E670	20 CE	BRA	5E640	Sort avec erreur
									Mise à jour du fichier
					E672	9E E7	LDX	<\$E7	
					E674	D6 F0	LDB	<\$F0	
					E676	C1 03	CMPB	5\$03	
					E678	26 04	BNE	5E67E	
					E67A	30 8D FE DC	LEAX	5E55A,PCR	
					E67E	06 0A	LDB	5\$0A	
					E680	A6 85	LDA	B,X	
					E682	A7 A5	STA	B,Y	
					E684	5A	DECB		
					E685	2C F9	BGE	5E680	
					E687	96 EB	LDA	<\$EB	
					E689	A7 2B	STA	\$0B,Y	
					E68B	96 EC	LDA	<\$EC	
					E68D	D6 F6	LDB	<\$F6	
					E68F	ED 2C	STD	\$0C,Y	
					E691	16 FF 2D	LBRA	5E5C1	
									Allocation d'un bloc
					E694	D6 F6	LDB	<\$F6	
									Recherche chromatique d'un bloc libre dans la FAT (bloc quelconque)
					E696	C1 28	CMPB	5\$40	
					E698	22 0E	BHI	5E6A8	
					E69A	5D	TSTB		
					E69B	27 11	BEQ	5E6AE	
					E69D	A6 A5	LDA	B,Y	
					E69F	81 FF	CMPA	5\$FF	
					E6A1	27 2D	BEQ	5E6D0	
					E6A3	5A	DECB		
					E6A4	C1 28	CMPB	5\$40	
					E6A6	23 F2	BLS	5E69A	
					E6A8	CB 02	ADDB	5\$02	
					E6AA	C1 51	CMPB	5\$81	
					E6AC	20 ED	BRA	5E69B	
									Recherche par symétrie d'un bloc libre dans la FAT (bloc de départ)
					E6AE	5F	CLRB		
					E6AF	31 A8 28	LEAY	40,Y	
					E6B2	86 05	LDA	5\$05	
					E6B4	C1 28	CMPB	5\$40	
					E6B6	10 22 FE F0	LBHI	5E5AA	
					E6BA	A6 A5	LDA	B,Y	
					E6BC	81 FF	CMPA	5\$FF	
					E6BE	27 0B	BEQ	5E6CB	
					E6C0	50	NEGB		
					E6C1	A6 A5	LDA	B,Y	
					E6C3	81 FF	CMPA	5\$FF	
					E6C5	27 04	BEQ	5E6CB	
					E6C7	50	NEGB		
					E6C8	5C	INCB		
					E6C9	20 E7	BRA	5E6B2	
					E6CB	CB 28	ADDB	5\$40	
					E6CD	31 A8 D8	LEAY	-40,Y	
					E6D0	6F A5	CLR	B,Y	
					E6D2	5A	DECB		
					E6D3	D7 F9	STB	<\$F9	
					E6D5	16 FE D6	LBRA	5E5AE	
									Effacement du fichier courant
					E6D8	A6 2D	LDA	\$0D,Y	
					E6DA	97 F6	STA	<\$F6	

Contrôleur externe CD90-640

E6DC	6F A4	CLR	,Y	Elimine le nom du fichier						
E6DE	17 FE E0	LBSR	\$\$E5C1	Ecrit le secteur catalogue		Initialisation d'une opération sur un bloc				
E6E1	25 8D	BCS	\$\$E670	Si erreur disque, sort		E6F7	D6 F6	LDB	<\$\$F6	Numéro de bloc courant
E6E3	10 9E ED	LDY	<\$\$ED			E6F9	4F	CLRA		Calcule le
E6E6	D6 F6	LDB	<\$\$F6	Efface		E6FA	54	LSRB		numéro
E6E8	5C	INCB		(met à \$\$FF)		E6FB	DD FB	STD	<\$\$FB	de piste
E6E9	A6 A5	LDA	B,Y	tous les blocs		E6FD	4C	INCA		Initialisation du compteur
E6EB	6F A5	CLR	B,Y	du fichier		E6FE	97 F5	STA	<\$\$F5	de secteur
E6ED	6A A5	DEC	B,Y	courant		E700	24 02	BCC	\$\$E704	Selon la parité du numéro
E6EF	1F 89	TFR	A,B	répertoriés		E702	86 09	LDA	\$\$S09	de bloc, le premier secteur
E6F1	81 C0	CMPA	\$\$C0	dans la FAT		E704	97 FA	STA	<\$\$FA	est le secteur 1 ou 9
E6F3	25 F3	BLO	\$\$E6E8			E706	39	RTS		
E6F5	20 DE	BRA	\$\$E6D5	Sortie sans erreur						

Contrôleur externe CD90-015

```

Identificateur de RomDisk (IDDSIK)
E000 44 FCC "D"      Contrôleur Western Digital
E001 4B FCC "K"      Fat de 80 octets
E002 43 FCC "C"      Simple densité
E003 27 FCB $27      Checksum de l'identificateur

Indirections standard sur les opérations physiques
E004 16 00 6D LBRA $E074 Fonctions standard
E007 16 00 1B LBRA $E025 Lancement du boot
E00A 16 03 B4 LBRA $E3C1 Formatage

---- Indirections standard sur les opérations logiques
E00D 16 06 44 LBRA $E654 Chargement de la FAT
E010 16 06 6D LBRA $E680 Ouverture d'un fichier
E013 16 07 66 LBRA $E77C Effacement d'un fichier
E016 16 06 5A LBRA $E673 Ecriture d'un secteur
E019 16 06 C2 LBRA $E6DE Création d'un fichier
E01C 16 07 19 LBRA $E738 Allocation d'un bloc
E01F 16 07 79 LBRA $E79B Initialise opération sur bloc
E022 16 05 F2 LBRA $E617 Clôture d'écriture
    
```

***** SYSTEME D'EXPLOITATION PHYSIQUE *****

```

Lancement du boot
E025 8D 40 BSR $E067   Initialise les registres
E027 86 01 LDA #01     Reset du contrôleur
E029 97 48 STA <$48    demandé
E02B 8D 47 BSR $E074   Exécute fonction standard
E02D 25 32 BCS $E061   Si erreur, sort
E02F 0F 4A CLR <$4A    Piste 0
E031 0F 4B CLR <$4B    demandée
E033 0F 49 CLR <$49    Lecteur 0 demandé
E035 97 4C STA <$4C    Secteur 1 demandé
E037 31 8D 82 44 LEAY $627F,PCR Limite de décodage boot
E03B 30 8D 81 C1 LEAX $6200,PCR Début du buffer de boot
E03F 9F 4F STX <$4F    Fixe le ptr buffer secteur
E041 86 02 LDA #02     Lecture d'un secteur
E043 97 48 STA <$48    demandée
E045 8D 2D BSR $E074   Exécute opération standard
E047 25 18 BCS $E061   Si erreur, sort
E049 34 20 PSHS Y      Limite de décodage en pile
E04B 86 55 LDA #055    Valeur de départ du checksum
E04D 6A 84 DEC ,X       Décode une donnée
E04F 63 84 COM ,X       du boot
E051 AB 80 ADDA ,X+     Mise à jour du checksum
E053 AC E4 CMXP ,S     Si pas fin du boot,
E055 26 F6 BNE $E04D   donnée suivante
E057 32 62 LEAS $02,S  Rétablit la pile
E059 A1 84 CMPA ,X     Si checksum incorrect,
E05B 26 04 BNE $E061   reset à froid application
E05D 6E 8D 81 9F JMP $6200,PCR  Sinon, exécute le boot

---- Sortie si erreur de boot
E061 0F 80 CLR <$80    Flag "contrôleur absent"
E063 6E 9F 00 1E JMP [$001E] Lance l'application à froid

Initialise les registres
E067 33 8D 00 5E LEAU $E0C9,PCR Ptr table bits lecteurs
E06B 1F 50 TFR PC,D    Pour TO, DP=$60
E06D 84 70 ANDA #070   Pour MO, DP=$20
E06F 1F 8B TFR A,DP   Initialise registres disque
E071 16 00 97 LBRA $E10B

Traitement des fonctions standard
E074 34 7F PSHS U,Y,X,DP,B,A,CC |
E076 8D EF BSR $E067   Initialise les registres
E078 C6 06 LDB #06     6 tentatives
E07A 34 04 PSHS B      maximum
E07C 96 48 LDA <$48    Lit la commande moniteur
E07E 30 8D 00 4B LEAX $E0CD,PCR  Ptr sur "Reset"
E082 85 01 BITA $01     Si reset demandé,
E084 26 20 BNE $E0A6   exécute le programme
E086 30 8D 01 23 LEAX $E1AD,PCR  Ptr sur "Lecture secteur"
E08A 85 02 BITA #02    Si lecture demandée,
E08C 26 18 BNE $E0A6   exécute le programme
E08E 30 8D 00 98 LEAX $E12A,PCR  Ptr sur "Ecriture"
E092 85 08 BITA #08    Si écriture demandée,
E094 26 10 BNE $E0A6   exécute le programme
E096 30 8D 02 AA LEAX $E344,PCR  Ptr sur "Piste 0"
E09A 85 20 BITA #020   Si piste 0 demandée,
E09C 26 08 BNE $E0A6   exécute le programme
E09E 30 8D 01 8C LEAX $E22E,PCR  Ptr sur "Recherche piste"
E0A2 85 40 BITA #040   Si recherche piste non
E0A4 27 1B BEQ $E0C1   demandée, sort avec erreur
E0A6 AD 84 BSR ,X       Exécute le programme
E0A8 24 18 BCC $E0C2   Si ok, sort
E0AA 96 4E LDA <$4E    Lit code d'erreur
E0AC 85 51 BITA #051   Si erreur contrôleur/
E0AE 26 12 BNE $E0C2   lecteur/protection, sort
E0B0 6A E4 DEC ,S       Si plus de tentatives,
E0B2 27 0D BEQ $E0C1   sort avec erreur
E0B4 0F 4E CLR <$4E    Efface code d'erreur
E0B6 81 08 CMPA #08     Si erreur sur les données,
    
```

```

E0B8 27 C2 BEQ $E07C   nouvel essai
E0BA 17 02 87 LBSR $E344 Retour en piste 0
E0BD 25 F1 BCS $E0B0   Si erreur, nouvel essai
E0BF 20 BB BRA $E07C   Nouvelle tentative

---- Sortie du programme
E0C1 53 COMB          Erreur dans CC
E0C2 35 06 PULS A,B   Récupère CC
E0C4 1E 9A EXG B,CC   Réinitialise CC
E0C6 54 LSRB         Ajuste bit d'erreur
E0C7 35 FE PULS A,B,DP,X,Y,U,PC |

Table des bits de lecteurs
E0C9 01 FCB $01      Lecteur 0
E0CA 02 FCB $02      Lecteur 1
E0CB 04 FCB $04      Lecteur 2
E0CC 08 FCB $08      Lecteur 3

Routine de reset contrôleur
E0CD 86 01 LDA #01     Code lecteur 0
E0CF 30 8D 06 FD LEAX $E7D0,PCR Ptr registres disque
E0D3 A7 03 STA $03,X   Initialise lecteur 0
E0D5 86 43 LDA #1      Activation de la
E0D7 97 4E STA <$4E    simple densité
E0D9 CC 80 00 LDD #08000 Flag pour reset registres
E0DC DD 51 STD <$51    Reset des registres
E0DE DD 53 STD <$53    de pistes
E0E0 DD 55 STD <$55    Reset des registres
E0E2 DD 57 STD <$57    de lecteurs
E0E4 86 0F LDA #0F     Sélectionne tous
E0E6 A7 08 STA $08,X   les lecteurs
E0E8 8D 34 BSR $E11E   Temporisation de 1 seconde
E0EA 6F 08 CLR $08,X   Désélectionne lecteurs
E0EC A6 03 LDA $03,X   Lit status
E0EE 84 80 ANDA #080   Si pas de lecteur
E0F0 27 08 BEQ $E0FA   occupé, passe
E0F2 8D 2A BSR $E11E   Temporisation de 1 seconde
E0F4 A6 03 LDA $03,X   lit status
E0F6 84 80 ANDA #080
E0F8 26 0A BNE $E104
E0FA C6 0A LDB #0A     Déplace les têtes de
E0FC 17 02 BC LBSR $E3BB 10 pistes vers l'extérieur
E0FF 17 02 AF LBSR $E3B1 Déplace de 5 pistes intérieur
E102 20 06 BRA $E10A   Sort du programme
E104 43 COMA          Erreur dans CC
E105 86 40 LDA #040    Code "Contrôleur non prêt"
E107 97 4E STA <$4E    Fixe l'erreur
E109 85 FCB $85       >BITA #04F
E10A 4F CLRA         Pas d'erreur dans CC

---- Initialise les registres disque
E10B 30 8D 06 C1 LEAX $E7D0,PCR Ptr registres disque
E10F 6C 84 INC ,X     ,X
E111 A6 04 LDA $04,X
E113 A6 02 LDA $02,X
E115 86 C0 LDA #0C0
E117 A7 02 STA $02,X
E119 86 00 LDA #000
E11B A7 08 STA $08,X
E11D 39 RTS          Pas de lecteur
                    sélectionné

Temporisation de 1 seconde
E11E 10 8E 5F 46 LDY #05F46 Compteur pour 1 seconde
E122 3D MUL          Effectue la temporisation
E123 3D MUL
E124 3D MUL
E125 31 3F LEAY -$01,Y
E127 26 F9 BNE $E122
E129 39 RTS

Routine d'écriture d'un secteur
E12A 96 48 LDA <$48
E12C 48 ASLA
E12D 27 05 BEQ $E134
E12F 17 00 FC LBSR $E22E Recherche la piste
E132 25 D7 BCS $E10B Si erreur, sort
E134 30 8D 06 98 LEAX $E7D0,PCR  Ptr registre disque
E138 C6 C5 LDB #0C5    Code "read sector"
E13A 34 41 PSHS U,CC  Protège flag interruptions
E13C 17 00 E3 LBSR $E222 Engage l'opération disque
E13F DE 4F LDU <$4F    Ptr buffer de secteur
E141 10 8E 5F 46 LDY #05F46 Compteur pour une piste
E145 31 3F LEAY -$01,Y Si fin de décomptage,
E147 27 0F BEQ $E158 traite l'erreur
E149 E6 03 LDB $03,X
E14B C5 80 BITB #080 Si fin opération, sort
E14D 27 09 BEQ $E158
E14F 54 LSRB         Si pas de donnée,
E150 24 F3 BCC $E145 attend capture
E152 A6 C0 LDA ,U+    Lit la donnée
E154 A7 84 STA ,X     Sauve la donnée
E156 20 ED BRA $E145 Donnée suivante

---- Sortie si pas de disquette
E158 35 41 PULS CC,U   Rétablit les interruptions
E15A 31 A4 LEAY ,Y    Si fin de piste,
    
```


Contrôleur externe CD90-015

```

E2D9 8A 10   ORA   #S10
E2DB A7 03   STA   $03,X
----
E2DD 96 4B   LDA   <$4B   Lit numéro de piste
E2DF A7 07   STA   $07,X
E2E1 A1 01   CMPA  $01,X   Lit numéro de piste courante
E2E3 27 34   BEQ   $E319   Si sur la piste, sort
E2E5 2B 2B   BMI   $E312   Si en deçà, déplace
---- Déplacement des têtes vers l'intérieur
E2E7 E6 01   LDB   $01,X   Demande le déplacement
E2E9 5C       INCB                d'une piste vers
E2EA E7 05   STB   $05,X   l'intérieur
---- Déplacement des têtes
E2EC D1 4B   CMPB  <$4B
E2EE 26 04   BNE   $E2F4
E2F0 C6 B3   LDB   #B3
E2F2 E7 03   STB   $03,X
E2F4 8D 3E   BSR   $E334   Sélectionne le lecteur
E2F6 C6 C3   LDB   #C3       Passe en mode
E2F8 E7 02   STB   $02,X     "seek"
E2FA 34 20   PSHS  Y
E2FC 10 8E 5F 46 LSHY  #S5F46   Compteur pour une piste
E300 31 3F   LEAY  -$01,Y    Si fin de décomptage,
E302 27 0A   BEQ   $E30E    "lecteur non prêt"
E304 E6 03   LDB   $03,X   Lit status
E306 C5 80   BITB  #S80     Si pas fin opération,
E308 26 F6   BNE   $E300    test suivant
E30A 35 20   PULS  Y
E30C 20 D3   BRA   $E2E1   Déplacement suivant
E30E 35 20   PULS  Y
E310 20 AC   BRA   $E2BE   Erreur "lecteur non prêt"
---- Déplacement des têtes vers l'extérieur
E312 E6 01   LDB   $01,X   Demande le déplacement
E314 5A       DECB                d'une piste
E315 E7 05   STB   $05,X   vers l'extérieur
E317 20 D3   BRA   $E2EC   Déplace les têtes
----
E319 96 48   LDA   <$48
E31B 48       ASLA
E31C 27 04   BEQ   $E322
E31E DC 4A   LDD   <$4A     Lit numéro de piste
E320 ED A4   STD   ,Y       Initialise position tête
E322 86 01   LDA   #S01
E324 A7 03   STA   $03,X
E326 4F       CLRA
E327 16 FD E1 LBRA  $E10B   Pas d'erreur dans CC
Sort du programme
----
Pointe sur position de tête
E32A 96 49   LDA   <$49     Lit numéro de lecteur
E32C 31 8D 7D 21 LEAY  $6051,PCR Ptr sur positions de têtes
E330 48       ASLA          Calcule offset
E331 31 A6   LEAY  A,Y      Positionne sur bon registre
E333 39       RTS
----
Initialise registres disque
E334 34 12   PSHS  X,A
E336 30 8D 04 96 LEAX  $E7D0,PCR Ptr registres disque
E33A 6F 08   CLR   $08,X   Reset bits de lecteur
E33C 96 49   LDA   <$49     Lit numéro de secteur
E33B A6 C6   LDA   A,U     Repère bits de lecteur
E340 A7 08   STA   $08,X   Initialise bits de lecteur
E342 35 92   PULS  A,X,PC
----
Routine de recherche de la piste 0
E344 8D 73   BSR   $E3B9   Se place en piste 0
E346 8E 80 00 LDX   #S8000   Valeur pour dévalidation
E349 8D DF   BSR   $E32A   Ptr sur position tête
E34B AF A4   STX   ,Y
E34D 8D 62   BSR   $E3B1   Dévalide le lecteur
E34F 9E 4A   LDX   <$4A     Se place en lère piste
E351 34 10   PSHS  X       Protège le numéro
E353 0F 4B   CLR   <$4B     de piste
E355 17 FE D6 LBSR  $E22E   Piste 0 demandée
E358 0F 4E   CLR   <$4E     Recherche la piste
E35A 17 FE D1 LBSR  $E22E   Efface code d'erreur
E35D 35 10   PULS  X       Recherche la piste
E35F 9F 4A   STX   <$4A     Restitue le numéro
E361 16 FD A7 LBRA  $E10B   de piste
Sort du programme
----
Déplacement d'une piste vers l'intérieur
E364 86 01   LDA   #S01
E366 A7 8D 04 67 STA  $E7D1,PCR Déplace les têtes
E36A 4C       INCA
E36B A7 8D 04 66 STA  $E7D5,PCR
E36F 20 0B   BRA   $E37C
---- Déplacement d'une piste vers l'extérieur
E371 86 01   LDA   #S01
E373 A7 8D 04 5E STA  $E7D5,PCR
E377 4C       INCA
E378 A7 8D 04 55 STA  $E7D1,PCR
---- Déplacement des têtes
E37C 86 B3   LDA   #B3
E37E A7 8D 04 51 STA  $E7D3,PCR
E382 8D B0   BSR   $E334   Sélectionne le lecteur
E384 86 C3   LDA   #C3       Passe en mode
E386 A7 8D 04 48 STA  $E7D2,PCR   "seek"

```

```

E38A 34 10   PSHS  X
E38C 8E 5F 46 LDX   #S5F46   Compteur pour une piste
E38F 30 1F   LEAX  -$01,X   Si fin de décomptage,
E391 27 1A   BEQ   $E3AD   erreur "lecteur non prêt"
E393 A6 8D 04 3C LDA  $E7D3,PCR   Lit status
E397 85 80   BITA  #S80     Si pas fin opération,
E399 26 F4   BNE   $E38F    boucle
E39B 35 10   PULS  X
E39D 1C FE   ANDCC #SFE     Efface bit d'erreur
E39F 85 04   BITA  #S04
E3A1 26 03   BNE   $E3A6
E3A3 17 FE 3A LBSR  $E1E0   Init "lecteur non prêt"
E3A6 86 01   LDA   #S01
E3A8 A7 8D 04 27 STA  $E7D3,PCR
E3AC 39       RTS
---- Sortie si erreur
E3AD 35 10   PULS  X       Rétablit la pile
E3AF 20 F2   BRA   $E3A3   Erreur "lecteur non prêt"
----
Se place en lère piste (piste 5)
E3B1 C6 05   LDB   #S05   Déplacement de 5 pistes
---- Déplace les têtes de plusieurs pistes vers l'intérieur
E3B3 8D AF   BSR   $E364   Déplace d'1 piste vers intér.
E3B5 5A     DECB                Pour le nombre de
E3B6 26 FB   BNE   $E3B3   pistes convenu
E3B8 39       RTS
----
Retour en piste 0
E3B9 C6 2D   LDB   #S2D   Déplacement de 45 pistes
---- Déplace les têtes de plusieurs pistes vers l'extérieur
E3BB 8D B4   BSR   $E371   Déplace d'1 piste vers extér.
E3BD 5A     DECB                Pour le nombre de
E3BE 26 FB   BNE   $E3BB   pistes convenu
E3C0 39       RTS
----
Routine de formatage d'une disquette
E3C1 34 7F   PSHS  U,Y,X,DP,B,A,CC
E3C3 17 FC A1 LBSR  $E067   Initialise les registres
E3C6 96 48   LDA   <$48
E3C8 84 7F   ANDA  #S7F    Garde le bit
E3CA 27 02   BEQ   $E3CE    de vérification
E3CC 0F 48   CLR   <$48
E3CE 17 FC FC LBSR  $E0CD   Reset du contrôleur
E3D1 25 15   BCS   $E3E8   Si erreur, sort
E3D3 0F 4E   CLR   <$4E     Efface code d'erreur
E3D5 17 FF E1 LBSR  $E3B9   Retour en piste 0
E3D8 17 FE 62 LBSR  $E23D   Teste présence disquette
E3DB 25 0B   BCS   $E3E8   Si erreur, sort
E3DD A6 8D 03 F2 LDA  $E7D3,PCR   Lit status
E3E1 85 10   BITA  #S10    Si pas bit "disk protect",
E3E3 27 06   BEQ   $E3EB    passe
E3E5 17 FD 7D LBSR  $E165   Erreur "disque protégé"
E3E8 16 00 E7 LBRA  $E4D2   Sort du programme
---- Fabrication de la table des entrelacements
E3EB 30 8D 7C E4 LEAX  $60D3,PCR Ptr table entrelacements
E3EF C6 0F   LDB   #S0F    16 octets à effacer
E3F1 6F 85   CLR   B,X
E3F3 5A     DECB                Efface table entrelacements
E3F4 26 FB   BNE   $E3F1
E3F6 30 88 10 LEAX  $10,X   Empile ptr sur
E3F9 34 10   PSHS  X       fin de table
E3FB 30 8D 7C D4 LEAX  $60D3,PCR Ptr table entrelacements
E3FF 86 01   LDA   #S01    Commence avec secteur n°1
E401 D6 4D   LDB   <$4D     Lit facteur d'entrelacement
E403 A7 84   STA   ,X      Ecrit numéro de secteur
E405 30 85   LEAX  B,X     Décale pointeur selon facteur
E407 4C       INCA          Numéro de secteur + 1
E408 81 10   CMPA  #S10    Si > à secteur 16,
E40A 22 0E   BHI   $E41A   sort
E40C AC AE   CMPX  ,S      Si pas hors table,
E40E 25 02   BLO   $E412   recherche secteur suivant
E410 30 10   LEAX  -$10,X  Retour de pointeur
E412 6D 84   TST   ,X      Si place libre,
E414 27 ED   BEQ   $E403   enregistre secteur
E416 30 01   LEAX  $01,X   Sinon, teste place
E418 20 F2   BRA   $E40C   suivante
E41A 32 62   LEAS  $02,S   Rétablit la pile
---- Formatage de la piste
E41C 0F 4B   CLR   <$4B
E41E 6F 8D 03 B5 CLR  $E7D7,PCR   Piste 0 demandée
E422 0F F8   CLR   <$F8
E424 86 06   LDA   #S06    6 tentatives
E426 34 02   PSHS  A       maximum
E428 0F F7   CLR   <$F7
E42A 96 D3   LDA   <$D3    Initialise le numéro
E42C 97 FA   STA   <$FA    de lecteur
E42E 17 00 BC LBSR  $E4ED   Formate la piste
E431 33 8D FC 94 LEAU  $E0C9,PCR Table des bits lecteurs
E435 0F F7   CLR   <$F7    Compteur de secteur à 0
E437 17 00 A8 LBSR  $E4E2   Initialise buffer à $E5
E43A D6 F7   LDB   <$F7    Lit le compteur de secteur
E43C C1 10   CMPB  #S10    Si les 16 secteurs ont
E43E 27 14   BEQ   $E454   été traités, exit
E440 30 8D 7C 8F LEAX  $60D3,PCR Ptr sur table entrelacements
E444 A6 85   LDA   B,X     Lit entrelacement courant
E446 97 4C   STA   <$4C    Fixe le numéro de secteur

```

Contrôleur externe CD90-015

E448	17 FC C0	LBSR	\$E10B	Initialise registres
E44B	17 FC DC	LBSR	\$E12A	Sauve le secteur
E44E	25 4E	BCS	\$E49E	Si erreur, tentative suivante
E450	0C F7	INC	<\$F7	Compteur de secteur + 1
E452	20 E6	BRA	\$E43A	Secteur suivant
E454	32 61	LEAS	\$01,S	Rétablit la pile
E456	86 01	LDA	#\$01	
E458	91 4B	CMPA	<\$4B	
E45A	26 24	BNE	\$E480	
E45C	0F 4B	CLR	<\$4B	Piste 0 demandée
E45E	17 FC AA	LBSR	\$E10B	Initialise registres
E461	17 FD 49	LBSR	\$E1AD	Lecture du secteur
E464	24 0D	BCC	\$E473	Si pas d'erreur passe
E466	6C 8D 03 6B	INC	\$E7D5,PCR	Demande piste suivante
E46A	0F 4E	CLR	<\$4E	Efface code d'erreur
E46C	17 FF 0D	LBSR	\$E37C	Déplace les têtes
E46F	25 61	BCS	\$E4D2	Si erreur, sort
E471	20 A9	BRA	\$E41C	Formate la piste
---- Déplacement des têtes				
E473	6C 8D 03 5E	INC	\$E7D5,PCR	Demnde déplacement intérieur
E477	17 FF 02	LBSR	\$E37C	Déplace les têtes
E47A	25 56	BCS	\$E4D2	Si erreur, sort
E47C	0C 4B	INC	<\$4B	Piste + 1
E47E	8D 62	BSR	\$E4E2	Initialise buffer de secteur
E480	0C F8	INC	<\$F8	Piste
E482	0C 4B	INC	<\$4B	suivante
E484	86 28	LDA	#\$28	
E486	91 4B	CMPA	<\$4B	Si piste 40, sort
E488	27 1C	BEQ	\$E4A6	
E48A	17 FE D7	LBSR	\$E364	Déplace d'1 piste vers intér.
E48D	25 43	BCS	\$E4D2	Si erreur, sort
E48F	96 4B	LDA	<\$4B	Lit numéro de piste
E491	A7 8D 03 3C	STA	\$E7D1,PCR	Initialise registres
E495	A7 8D 03 3E	STA	\$E7D7,PCR	de piste courante
E499	17 FC 6F	LBSR	\$E10B	Initialise registres
E49C	20 86	BRA	\$E424	Formatage de la piste
---- Sortie si erreur				
E49E	6A E4	DEC	,S	Si pas dernière tentative,
E4A0	26 86	BNE	\$E428	nouvel essai
E4A2	32 61	LEAS	\$01,S	Rétablit la pile
E4A4	20 2C	BRA	\$E4D2	Sort du programme
---- Initialise tous les secteurs de la piste de catalogue				
E4A6	8E 00 14	LDX	#\$0014	Piste 20
E4A9	9F 4A	STX	<\$4A	demandée
E4AB	CC FF 7F	LDD	#\$FF7F	Fabrique un secteur de
E4AE	8D 35	BSR	\$E4E5	128 octets à \$FF
E4B0	86 10	LDA	#\$10	16 secteurs
E4B2	97 4C	STA	<\$4C	Initialise registre secteur
E4B4	86 08	LDA	#\$08	Ecriture secteur
E4B6	97 48	STA	<\$48	demandée
E4B8	17 FB 49	LBSR	\$E004	Exécute opération standard
E4BB	25 15	BCS	\$E4D2	Si erreur, sort
E4BD	0A 4C	DEC	<\$4C	Pour les
E4BF	26 F7	BNE	\$E4B8	16 secteurs
---- Initialise la FAT				
E4C1	9E 4F	LDX	<\$4F	Ptr sur secteur
E4C3	6F 84	CLR	,X	Efface premier octet
E4C5	CC FE FE	LDD	#\$FEFE	Réserve la piste
E4C8	ED 88 29	STD	\$29,X	du catalogue
E4CB	86 02	LDA	#\$02	Secteur 2
E4CD	97 4C	STA	<\$4C	demandé
E4CF	17 FB 32	LBSR	\$E004	Exécute opération standard
E4D2	25 04	BCS	\$E4D8	Si erreur, sort
E4D4	86 28	LDA	#\$28	Positionne en
E4D6	97 4B	STA	<\$4B	piste 40
E4D8	17 FC 30	LBSR	\$E10B	
E4DB	1F A8	TFR	CC,A	Erreur dans A
E4DD	35 01	PULS	CC	Rétablit CC
E4DF	44	LSRA		Fixe le bit d'erreur
E4E0	35 FE	PULS	A,B,DP,X,Y,U,PC	
Initialise un buffer de secteur pour formatage				
E4E2	CC E5 7F	LDD	#\$E57F	128 octets à \$E5
---- Initialise un buffer de secteur				
E4E5	9E 4F	LDX	<\$4F	Ptr sur buffer de secteur
E4E7	A7 85	STA	B,X	Ecrit le valeur
E4E9	5A	DECB		Autant de fois
E4EA	2A FB	BPL	\$E4E7	que convenu
E4EC	39	RTS		
Formatage de la piste				
E4ED	34 01	PSHS	CC	Protège interruptions
E4EF	1A 50	ORCC	#\$50	Gel des interruptions
E4F1	17 00 D8	LBSR	\$E5CC	Préparation identificateur
E4F4	8D 7B	BSR	\$E571	Calcule CRC identificateur
E4F6	DE 4F	LDU	<\$4F	Ptr buffer secteur
E4F8	6F C8 24	CLR	\$24,U	Clôt zone de paramètres
E4FB	33 8D FB CA	LEAU	\$E0C9,PCR	Ptr table bits lecteurs
E4FF	17 FE 32	LBSR	\$E334	Sélectionne le lecteur
E502	CC 40 FF	LDD	#\$40FF	Registres test et remplissage
E505	30 8D 02 C7	LEAX	\$E7D0,PCR	Ptr registre disque
E509	A5 03	BITA	\$03,X	Attend front ascendant
E50B	26 FC	BNE	\$E509	de l'index
E50D	A5 03	BITA	\$03,X	Attend front descendant
E50F	27 FC	BEQ	\$E50D	de l'index
E511	E7 84	STB	,X	

E513	C6 DB	LDB	#\$DB	Passé en mode
E515	E7 02	STB	\$02,X	formatage
E517	A5 03	BITA	\$03,X	Attend front ascendant
E519	26 FC	BNE	\$E517	de l'index
E51B	A5 03	BITA	\$03,X	Attend front descendant
E51D	27 FC	BEQ	\$E51B	de l'index
---- Ecrit amorce de piste				
E51F	10 8E 00 36	LDY	#\$0036	54 caractères à écrire
---- Ecrit intervalle de secteur				
E523	DE 4F	LDU	<\$4F	Ptr buffer secteur
E525	30 8D 02 A7	LEAX	\$E7D0,PCR	Ptr registre disque
E529	CC 01 FF	LDD	#\$01FF	Caractère à \$FF
E52C	E7 84	STB	,X	Sauve le caractère
E52E	A5 03	BITA	\$03,X	Attend fin de
E530	27 FC	BEQ	\$E52E	transmission
E532	31 3F	LEAY	-\$01,Y	Pour les 54
E534	26 F6	BNE	\$E52C	caractères
---- Ecrit 12 espaces				
E536	31 2C	LEAY	\$0C,Y	12 caractères à écrire
E538	C6 AA	LDB	#\$AA	Caractère à \$AA
E53A	E7 84	STB	,X	Sauve le caractère
E53C	A5 03	BITA	\$03,X	Attend fin de
E53E	27 FC	BEQ	\$E53C	transmission
E540	31 3F	LEAY	-\$01,Y	Pour les 12
E542	26 F6	BNE	\$E53A	caractères
---- Ecrit le secteur				
E544	E6 C0	LDB	,U+	Lit un caractère
E546	E7 84	STB	,X	Sauve le caractère
E548	27 06	BEQ	\$E550	Si fin de table, sort
E54A	A5 03	BITA	\$03,X	Attend fin de
E54C	26 F6	BNE	\$E544	transmission
E54E	20 FA	BRA	\$E54A	Caractère suivant
---- Secteur suivant				
E550	0C F7	INC	<\$F7	Numéro de secteur + 1
E552	D6 F7	LDB	<\$F7	Lit le numéro de secteur
E554	C1 10	CMPB	#\$10	Si secteur 16,
E556	10 27 00 A0	LBEQ	\$E5FA	sort du programme
E55A	4F	CLRA		Pour 16 bits
E55B	1F 01	TFR	D,X	<Résidus>
E55D	30 8D 7B 72	LEAX	\$6D03,PCR	Ptr sur liste entrelacement
E561	30 8B	LEAX	D,X	Ptr sur numéro de secteur
E563	A6 84	LDA	,X	Initialise
E565	97 FA	STA	<\$FA	numéro de secteur
E567	8D 63	BSR	\$E5CC	Prépare identificateur
E569	8D 06	BSR	\$E571	Calcule CRC identificateur
E56B	10 8E 00 BE	LDY	#\$00BE	190 caractères d'intervalle
E56F	20 B2	BRA	\$E523	Ecrit nouveau secteur
Calcule CRC identificateur				
E571	CC F5 7E	LDD	#\$F57E	
E574	9E 4F	LDX	<\$4F	
E576	ED 84	STD	,X	Ecrit marquage identif.
E578	CC FF FF	LDD	#\$FFFF	
E57B	34 06	PSHS	B,A	Initialise
E57D	DE 4F	LDU	<\$4F	le CRC
E57F	86 0B	LDA	#\$0B	Ptr sur paramètres
E581	97 FC	STA	<\$FC	10 octets
---- Calcule le CRC de la zone				
E583	0A FC	DEC	<\$FC	Si fin du calcul,
E585	27 23	BEQ	\$E5AA	exit
E587	C6 04	LDB	#\$04	Calcul pour 1 bit sur 2
E589	37 02	PULU	A	Lit le paramètre
E58B	34 04	PSHS	B	Empile compteur de paramètres
E58D	49	ROLA		Passé un bit
E58E	68 62	ASL	\$02,S	Expulser bit
E590	69 61	ROL	\$01,S	CRC
E592	46	RORA		Teste bit sortant
E593	49	ROLA		caractère
E594	28 2D	BVC	\$E5C3	Si pas différents, passe
E596	E6 62	LDB	\$02,S	
E598	C8 21	EORB	#\$21	
E59A	E7 62	STB	\$02,S	Active le
E59C	E6 61	LDB	\$01,S	CRC-CITIT
E59E	C8 10	EORB	#\$10	
E5A0	E7 61	STB	\$01,S	
E5A2	49	ROLA		Décale registre
E5A3	35 04	PULS	B	
E5A5	5A	DECB		Bit suivant
E5A6	26 E3	BNE	\$E58B	
E5A8	20 D9	BRA	\$E583	Caractère suivant
---- Ecriture du CRC				
E5AA	35 04	PULS	B	Lit poids fort CRC
E5AC	8D 3A	BSR	\$E5E8	Expanse poids fort CRC
E5AE	ED A1	STD	,Y++	Ecrit poids fort CRC
E5B0	35 04	PULS	B	Lit poids faible CRC
E5B2	8D 3A	BSR	\$E5E8	Expanse poids faible CRC
E5B4	ED A1	STD	,Y++	Ecrit poids faible CRC
---- Ajoute l'intervalle d'espaces				
E5B6	8E 00 0B	LDX	#\$000B	
E5B9	CC FF FF	LDD	#\$FFFF	
E5BC	ED A1	STD	,Y++	
E5BE	30 1F	LEAX	-\$01,X	Ajoute intervalle d'espaces
E5C0	26 FA	BNE	\$E5BC	
E5C2	39	RTS		
---- Temporisation				
E5C3	C6 03	LDB	#\$03	

Contrôleur externe CD90-015

E5C5 C6 03	LDB #503	Temporisation ré pondant	E65E 43	COMA	Erreur dans CC
E5C7 5A	DECB	au temps	E65F 39	RTS	
E5C8 26 FD	BNE \$E5C7	d'activation du CRC-CCITT			
E5CA 20 D6	BRA \$E5A2				
Préparation de l'identificateur de secteur					
E5CC 0F F9	CLR <\$F9	Numéro de face à 0	E660 4F	CLRA	Pas d'erreur dans CC
E5CE 0F FB	CLR <\$FB	Densité à 0	E661 39	RTS	
E5D0 86 04	LDA #504	Expansion des	Sortie sans erreur		
E5D2 97 FD	STA <\$FD	4 paramètres	E662 86 03	LDA #503	Secteur 3 demandé
E5D4 33 8D 7B 20	LEAU \$60F8,PCR	Ptr sur paramètres	E664 9E E9	LDX <\$E9	Buffer secteur
E5D8 10 9E 4F	LDY <\$4F	Ptr sur liste paramètres	E666 9F 4F	STX <\$4F	initialisé
E5DB 31 22	LEAY \$02,Y	Positionne sur paramètres	E668 97 4C	STA <\$4C	Initialise secteur
E5DD E6 C0	LDB ,U+	Lit un paramètre	E66A C6 14	LDB #514	
E5DF 8D 07	BSR \$E5E8	Expansion du paramètre	E66C 4F	CLRA	Piste 20 demandée
E5E1 ED A1	STD ,Y++	Sauve expansion	E66D DD 4A	STD <\$4A	
E5E3 0A FD	DEC <\$FD	Pour les	E66F 86 02	LDA #502	Code pour lecture secteur
E5E5 26 F6	BNE \$E5DD	4 paramètres	E671 20 02	BRA \$E675	Charge le secteur
E5E7 39	RTS				
Espace un caractère					
E5E8 34 04	PSHS B	Empile le caractère	E673 86 08	LDA #508	Code pour écriture secteur
E5EA C6 01	LDB #501	Bit d'arrêt	E675 97 48	STA <\$48	Mise à jour de l'opération
E5EC 4F	CLRA	Pour 16 bits	E677 10 9E E9	LDY <\$E9	
E5ED 1A 01	ORCC #501	Bit rentrant à 1	E67A 17 F9 87	LBSR \$E004	Exécute opération standard
E5EF 59	ROLB	Décale avec un bit	E67D 86 03	LDA #503	Code I/O Error initialisé
E5F0 49	ROLA	entrant à 1	E67F 39	RTS	
E5F1 68 E4	ASL ,S	Décale le caractère	Ouverture d'un fichier		
E5F3 59	ROLB	Décale avec le	E680 8D E0	BSR \$E662	Charge ler secteur catalogue
E5F4 49	ROLA	bit sortant du caractère	E682 25 D8	BCS \$E65C	Si erreur, sort
E5F5 24 F6	BCC \$E5ED	Si pas fin, boucle	E684 8E 00 04	LDX #50004	4 entrée par secteur
E5F7 32 61	LEAS \$01,S	Rétablit la pile	E687 10 9E E9	LDY <\$E9	Init. ptr sur les entrées
E5F9 39	RTS		E68A DE E7	LDU <\$E7	et sur nom de fichier
Sortie du programme de formatage					
E5FA CC 80 C0	LDD #580C0		E68C D6 F0	LDB <\$F0	Si la sauvegarde avec
E5FD E7 8D 01 D1	STB \$E7D2,PCR		E68E C1 03	CMPB #503	écrasement est programmée,
E601 A5 8D 01 CE	BITA \$E7D3,PCR	Attend fin de	E690 26 04	BNE \$E696	le fichier recherché
E605 26 FA	BNE \$E601	l'opération	E692 33 8D FF 76	LEAU \$E60C,PCR	s'appelle "SCRATCH.DOS"
E607 35 01	PULS CC	Dégel des interruptions	E696 5F	CLRB	
E609 16 FA FF	LBRA \$E10B	Sort du programme	E697 C1 0B	CMPB #50B	Si le nom est trouvé,
***** SYSTEME D'EXPLOITATION LOGIQUE *****					
Nom de fichier temporaire pour la sauvegarde					
E60C 53 43 52 41	FCC "SCRATCH.DOS"		E699 24 24	BHS \$E6BF	recupère les paramètres.
E610 54 43 48 20			E69B A6 A5	LDA B,Y	Si on se trouve en fin
E614 44 4F 53			E69D 81 FF	CMPA #5FF	de catalogue,
Clôture du fichier					
E617 D6 F0	LDB <\$F0	Sauvegarde sans	E69F 27 1B	BEQ \$E6BC	sortie du programme.
E619 C1 02	CMPB #502	écrasement ?	E6A1 5C	INCB	Sinon, compare le nom de
E61B 27 23	BEQ \$E640	Oui, écrit FAT seule	E6A2 A1 C0	CMPA ,U+	l'entrée avec nom de
E61D 0A F0	DEC <\$F0	Passé en sauvg. sans écr.	E6A4 27 F1	BEQ \$E697	fichier demandé
E61F 8D 5F	BSR \$E680	Ouvre le fichier	E6A6 31 A8 20	LEAY \$20,Y	Passé à
E621 25 30	BCS \$E653	Si erreur disque, sort	E6A9 30 1F	LEAX -\$01,X	l'entrée
E623 5D	TSTB	Fichier existe déjà?	E6AB 26 DD	BNE \$E68A	suivante
E624 27 05	BEQ \$E62B	Non, pas d'effacement	E6AD 0C 4C	INC <\$4C	Si le dernier
E626 17 01 53	LBSR \$E77C	Effacement du fichier	E6AF 96 4C	LDA <\$4C	secteur du catalogue
E629 25 28	BCS \$E653	Si erreur, sort	E6B1 81 10	CMPA #510	est atteint, sort du
E62B 0C F0	INC <\$F0	Retour en sauvg avec écrasmt	E6B3 22 07	BHI \$E6BC	programme
E62D 8D 51	BSR \$E680	Ouvre fichier "SCRATCH.DOS"	E6B5 17 F9 4C	LBSR \$E004	Sinon, secteur suivant
E62F 25 22	BCS \$E653	Si erreur, sort	E6B8 86 03	LDA #503	Code "erreur d'entrée-sortie"
E631 C6 0A	LDB #50A		E6BA 20 C6	BRA \$E682	Boucle avec ou sans erreur
E633 9E E7	LDX <\$E7	Recopie le nom	---- Sortie si fichier introuvable		
E635 A6 85	LDA B,X	du fichier	E6BC 5F	CLRB	Si fichier introuvable,
E637 A7 A5	STA B,Y	courant à la place	E6BD 20 1B	BRA \$E6DA	secteur=0 et sort
E639 5A	DECB	de "SCRATCH.DOS"	Récupère les paramètres de fichier		
E63A 2C F9	BGE \$E635		E6BF E6 2B	LDB \$0B,Y	Si type de fichier non
E63C 8D 35	BSR \$E673	Écrit le secteur catalogue	E6C1 D1 EB	CMPB <\$EB	toléré, poursuit la
E63E 25 13	BCS \$E653	Si erreur de disque, sort	E6C3 26 F7	BNE \$E6BC	recherche dans le catalogue
---- Sauvegarde de la FAT					
E640 86 02	LDA #502		E6C5 E6 2C	LDB \$0C,Y	Si flag de fichier non
E642 97 4C	STA <\$4C	Programme la	E6C7 D1 EC	CMPB <\$EC	toléré, poursuit la
E644 C6 14	LDB #514	sauvegarde	E6C9 26 F1	BNE \$E6BC	recherche dans catalogue
E646 4F	CLRA	de la FAT	E6CB D6 4C	LDB <\$4C	Stocke le
E647 DD 4A	STD <\$4A		E6CD A6 2D	LDA \$0D,Y	premier bloc
E649 DC ED	LDD <\$ED	Ptr sur FAT	E6CF 97 F6	STA <\$F6	du fichier
E64B DD 4F	STD <\$4F	Sauvegarde	E6D1 0F F5	CLR <\$F5	Compteur de secteur à 0
E64D 8D 24	BSR \$E673	de la FAT	E6D3 AE 2E	LDX \$0E,Y	Stocke le nombre d'octets
E64F 25 02	BCS \$E653	Si erreur de disque, sort	E6D5 9F F7	STX <\$F7	dans le dernier secteur
E651 0F F0	CLR <\$F0	Clôture du fichier effectuée	E6D7 10 9F FA	STY <\$FA	Stocke pointeur d'entrée
E653 39	RTS		E6DA D7 F9	STB <\$F9	Numéro secteur catalogue
Chargement de la FAT					
E654 9E ED	LDX <\$ED	Ptr sur buffer de FAT	E6DC 20 82	BRA \$E660	Retour sans erreur
E656 9F 4F	STX <\$4F	initialisé	Création d'un fichier		
E658 86 02	LDA #502	Secteur 2 demandé	E6DE 10 9E ED	LDY <\$ED	Bloc libre
E65A 20 0C	BRA \$E668	Charge la FAT	E6E1 8D 6F	BSR \$E752	dans la FAT ?
Fixe l'erreur moniteur					
E65C 97 E5	STA <\$E5	Écrit code dans DKSTA	E6E3 25 9D	BCS \$E682	Non, sort avec erreur
Retour sans erreur					

Contrôleur externe CD90-015

E709	22 07	BHI	\$E712	erreur Disque Plein	E75A	10 22 FE FE	LBHI	\$E65C		sort avec erreur
E70B	17 F8 F6	LBSR	\$E004	Charge secteur suivant	E75E	A6 A5	LDA	B,Y		Si la place
E70E	86 03	LDA	#S03	Code "erreur d'entrée-sortie"	E760	81 FF	CMPA	#\$FF		est libre,
E710	20 D8	BRA	\$E6EA	Boucle avec ou sans erreur	E762	27 0B	BEQ	\$E76F		sort
----	Sortie si erreur "disque plein"				E764	50	NEGB			Offset pour test symétrique
E712	86 05	LDA	#S05	Code "disque plein"	E765	A6 A5	LDA	B,Y		Si la place
E714	20 CD	BRA	\$E6E3	Sort avec erreur	E767	81 FF	CMPA	#\$FF		est libre,
					E769	27 04	BEQ	\$E76F		sort
					E76B	50	NEGB			Offset pour test symétrique
Mise à jour du fichier					E76C	5C	INCB			Place
E716	9E E7	LDX	<\$E7		E76D	20 E7	BRA	\$E756		suivante
E718	D6 F0	LDB	<\$F0	Si la sauvegarde	E76F	CB 28	ADDB	#40		Rétablit l'offset
E71A	C1 03	CMPB	#S03	avec écrasement	E771	31 A8 D8	LEAY	-40,Y		Rétablit le pointeur
E71C	26 04	BNE	\$E722	est programmée, le fichier	E774	6F A5	CLR	B,Y		Réserve la place du bloc
E71E	30 8D FE EA	LEAX	\$E60C,PCR	est nommé "SCRATCH.DOS"	E776	5A	DECB			Ajuste et stocke
E722	C6 0A	LDB	#\$0A		E777	D7 F9	STB	<\$F9		le numéro de bloc
E724	A6 85	LDA	B,X	Recopie le	E779	16 FE E4	LBRA	\$E660		Sort sans erreur
E726	A7 A5	STA	B,Y	nom de fichier						
E728	5A	DECB		dans le catalogue	Effacement du fichier courant					
E729	2C F9	BGE	\$E724		E77C	A6 2D	LDA	\$0D,Y		Premier bloc du fichier
E72B	96 EB	LDA	<\$EB	Recopie du type de fichier	E77E	97 F6	STA	<\$F6		sélectionné
E72D	A7 2B	STA	\$0B,Y	dans le catalogue	E780	6F A4	CLR	,Y		Élimine le nom du fichier
E72F	96 EC	LDA	<\$EC	Recopie du flag de fichier	E782	17 FE EE	LBSR	\$E673		Écrit le secteur catalogue
E731	D6 F6	LDB	<\$F6	et du numéro de bloc	E785	25 8D	BCS	\$E714		Si erreur disque, sort
E733	ED 2C	STD	\$0C,Y	courant dans le catalogue	E787	10 9E ED	LDY	<\$ED		
E735	16 FF 3B	LBRA	\$E673	Sauve le secteur de catalogue	E78A	D6 F6	LDB	<\$F6		Efface
					E78C	5C	INCB			(met à \$FF)
Allocation d'un bloc					E78D	A6 A5	LDA	B,Y		tous les blocs
E738	D6 F6	LDB	<\$F6	Bloc de départ	E78F	6F A5	CLR	B,Y		du fichier
----	Recherche chromatique d'un bloc libre dans la FAT				E791	6A A5	DEC	B,Y		courant
E73A	C1 28	CMPB	#40	Si bloc > à 40,	E793	1F 89	TFR	A,B		répertoriés
E73C	22 0E	BHI	\$E74C	ajuste l'offset du bloc	E795	81 C0	CMPA	#\$C0		dans la FAT
E73E	5D	TSTB		Si le bloc 0 est atteint,	E797	25 F3	BLO	\$E78C		
E73F	27 11	BEQ	\$E752	recherche par symétrie	E799	20 DE	BRA	\$E779		Sortie sans erreur
E741	A6 A5	LDA	B,Y	Si la place	Initialisation d'un opération sur un bloc					
E743	81 FF	CMPA	#\$FF	est libre,	E79B	D6 F6	LDB	<\$F6		Numéro de bloc courant
E745	27 2D	BEQ	\$E774	sort	E79D	4F	CLRA			Calcule le
E747	5A	DECB		Décrémente le bloc	E79F	DD FB	STD	<\$FB		numéro
E748	C1 28	CMPB	#40	d'office, mais si le bloc	E7A1	4C	INCA			de piste
E74A	23 F2	BLS	\$E73E	est > à 40, rajoute 2 pour	E7A2	97 F5	STA	<\$F5		Initialisation du compteur
E74C	CB 02	ADDB	#S02	établir une incrémentation	E7A4	24 02	BCC	\$E7A8		de secteur
E74E	C1 51	CMPB	#81	Au-delà du dernier bloc ?	E7A6	86 09	LDA	#\$09		Selon la parité du numéro
E750	20 ED	BRA	\$E73F	Boucle pour test	E7A8	97 FA	STA	<\$FA		de bloc, le premier secteur
----	Recherche par symétrie d'un bloc libre dans la FAT				E7AA	39	RTS			est le secteur 1 ou 9
E752	5F	CLRB		Offset pour test à 0						
E753	31 A8 28	LEAY	40,Y	Position en milieu de FAT						
E756	86 05	LDA	#\$05	Code "erreur disque plein"						
E758	C1 28	CMPB	#40	Si dernier test,						

Contrôleur externe CQ90-028

```

Identificateur de RomDisk (IDDSIK)
E000 51 FCC "Q"      Contrôleur QDD
E001 4B FCC "K"      Fat de 40 octets
E002 43 FCC "C"      Simple densité
E003 34 FCB $34      Checksum de l'identificateur
  
```

```

Indirections standard sur les opérations physiques
E004 17 00 36 LBSR $E03D Fonctions standard
E007 17 00 33 LBSR $E03D Lancement du boot
E00A 17 00 30 LBSR $E03D Formatage
---- Indirections standard sur les opérations logiques
E00D 17 00 2D LBSR $E03D Chargement de la FAT
E010 17 00 2A LBSR $E03D Ouverture d'un fichier
E013 17 00 27 LBSR $E03D Effacement d'un fichier
E016 17 00 24 LBSR $E03D Ecriture d'un secteur
E019 17 00 21 LBSR $E03D Création d'un fichier
E01C 17 00 1E LBSR $E03D Allocation d'un bloc
E01F 16 07 8B LBRA $E7AD Initialise opération sur bloc
E022 16 06 01 LBRA $E626 Clôture d'écriture
Indirections standard sur les opérations physiques (suite)
E025 16 01 35 LBRA $E15D Fonction standard (n° réel)
  
```

```

Routine pour calcul du checksum de l'IDDSIK de la banque 1
E028 8D 0F BSR $E039 Commute sur banque 1
E02A EB 0C ADDB ,U+   Calcul partiel du checksum
E02C 20 04 BRA $E032 Rétablit banque 0
  
```

```

Exécute une routine en banque 1 moniteur
E02E 8D 09 BSR $E039 Commute sur banque 1
E030 AD 5D JSR -$03,U Exécution de la routine
E032 86 00 LDA #000 Rétablit banque 0
E034 A7 8D 07 A4 STA $E7DC,PCR de la ROM
E038 39 RTS
  
```

```

Commute sur la banque 1 de la ROM moniteur
E039 86 80 LDA #080 Banque ROM 1 demandée
E03B 20 F7 BRA $E034 Commute banque ROM
  
```

```

Programme d'exécution des 9 premières indirections
E03D 34 4B PSHS U,DP,A,CC
E03F 1A 50 ORCC #050 Gel des interruptions
E041 EE 65 LDU $05,S Récupère le PC en pile
E043 17 01 DA LBSR $E220 Fixe le DP
E046 96 48 LDA <$48 Lit commande
E048 81 01 CMPA #001 Si pas reset contrôleur
E04A 26 13 BNE $E05F demandé, passe
E04C 34 40 PSHS U
E04E 33 8D FF B5 LEAU $E007,PCR
E052 11 A3 E4 CMPU ,S
E055 35 40 PULS U
E057 26 06 BNE $E05F
---- Reset du contrôleur
E059 8D 30 BSR $E08B Commute banque ROM
E05B 8D 79 BSR $E0D6 Reset du contrôleur
E05D 20 20 BRA $E07F Sort du programme
  
```

```

---- Exécution si lecteur 0
E05F 96 49 LDA <$49 Lit numéro de lecteur
E061 26 09 BNE $E06C Si pas 0, passe
E063 AD C9 01 3B JSR $013B,U Exécute l'opération en $E142
E067 17 02 4E LBSR $E2B8 Eteint le moteur
E06A 20 15 BRA $E081 Sort du programme
  
```

```

---- Exécution si pas lecteur 0
E06C 8D 1D BSR $E08B Commute banque ROM
E06E 8D 3F BSR $E0AF Vérifie checksum IDDISK
E070 24 07 BCC $E079 Si ok, passe
E072 86 10 LDA #010 Code "lecteur non prêt"
E074 97 4E STA <$4E Fixe l'erreur
E076 43 COMA Erreur dans CC
E077 20 06 BRA $E07F Sort du programme
E079 0A 49 DEC <$49 Ajuste n° de lecteur
E07B 8D B1 BSR $E02E Exécute en banque 1
E07D 0C 49 INC <$49 Rétablit n° de lecteur
---- Sortie du programme
E07F 8D 1D BSR $E09E Rétablit page initiale
E081 35 02 PULS A Récupère CC
E083 1E 8A EXG A,CC Rétablit CC
E085 44 LSRA Bit d'erreur dans CC
E086 35 4A PULS A,DP,U Rétablit
E088 32 62 LEAS $02,S la pile
E08A 39 RTS
  
```

```

Passe en page de ROM moniteur
E08B 96 81 LDA <$81 Passe en
E08D 8A 10 ORA #010 commutation
E08F A7 8D 07 54 STA $E7E7,PCR de page directe
E093 A6 8D 07 4E LDA $E7E5,PCR
E097 8A 80 ORA #080 Active commutation de ROM
E099 A7 8D 07 48 STA $E7E5,PCR
E09D 39 RTS
  
```

Rétablit page initiale

```

E09E A6 8D 07 43 LDA $E7E5,PCR
E0A2 84 7F ANDA #07F Désactive commutation ROM
E0A4 A7 8D 07 3D STA $E7E5,PCR
E0A8 96 81 LDA <$81 Rétablit commutation
E0AA A7 8D 07 39 STA $E7E7,PCR de page initiale
E0AE 39 RTS
  
```

```

Vérification du checksum de l'IDDISK en banque 1 (présence contrôleur)
E0AF 34 44 PSHS U,B
E0B1 5F CLRB
E0B2 33 8D FF 4D LEAU $E003,PCR Limite de lecture
E0B6 34 40 PSHS U en pile
E0B8 33 5D LEAU -$03,U Ajuste ptr sur IDDISK
E0BA 17 FF 6B LBSR $E028 Ajoute caractère suivant
E0BD 11 A3 E4 CMPU ,S si pas le 3ème caractère
E0C0 26 F8 BNE $E0BA d'identification, suivant
E0C2 32 62 LEAS $02,S rétablit la pile
E0C4 CB 55 ADDB #055 Ajoute checksum de départ
E0C6 34 04 PSHS B
E0C8 5F CLRB
E0C9 17 FF 5C LBSR $E028 Efface le résultat
E0CC E1 E0 CMPB ,S+ Lit le checksum en banque 1
E0CE 35 44 PULS B,U Si checksum correct, sort
E0D0 27 02 BEQ $E0D4
E0D2 43 COMA Erreur dans CC
E0D3 39 RTS
---- Sortie si checksum de l'IDDISK correct
E0D4 4F CLRA Pas d'erreur dans CC
E0D5 39 RTS
  
```

```

Routine de reset du contrôleur
E0D6 34 34 PSHS Y,X,B
E0D8 17 01 41 LBSR $E21C Initialise registres
E0DB 8D 4D BSR $E12A Init registres disque
E0DD 8D D0 BSR $E0AF Vérifie checksum IDDISK
E0DF 25 42 BCS $E123 Si erreur, simple densité
E0E1 5F CLRB Efface le caractère
E0E2 33 8D FF 1A LEAU $E000,PCR Ptr sur 1er caractère IDDISK
E0E6 17 FF 3F LBSR $E028 Lit le caractère en banque 1
E0E9 C1 51 CMPP #0 Si pas caractère Q,
E0EB 26 09 BNE $E0F6 reset lecteur
E0ED 33 8D 00 3C LEAU $E12D,PCR
E0F1 17 FF 3A LBSR $E02E Exécute en banque 1
E0F4 20 2D BRA $E123 Active simple densité
E0F6 33 8D FF 0D LEAU $E004+$0003,PCR Ptr sur fonctions standard
E0FA 86 01 LDA #001 Reset contrôleur
E0FC 97 48 STA <$48 demandé
E0FE 17 FF 2D LBSR $E02E Exécute reset en banque 1
E101 5F CLRB Efface le caractère
E102 33 8D FE FC LEAU $E002,PCR Ptr sur 3ème caractère IDDISK
E106 17 FF 1F LBSR $E028 Lit le caractère en banque 1
E109 C1 44 CMPB #D Si pas caractère D,
E10B 26 16 BNE $E123 active simple densité
E10D 86 04 LDA #004 Force en
E10F 97 48 STA <$48 simple densité
E111 33 8D FE F2 LEAU $E004+$0003,PCR Ptr sur fonctions standard
E115 17 FF 16 LBSR $E02E Exécute en banque 1
E118 86 20 LDA #020 Recherche de la piste 0
E11A 97 48 STA <$48 demandée
E11C 17 FF 0F LBSR $E02E Exécute en banque 1
E11F 86 01 LDA #001 Rétablit mot de commande
E121 97 48 STA <$48 de reset
E123 86 43 LDA #0C Active la
E125 97 4E STA <$4E simple densité
E127 4F CLRA Pas d'erreur dans CC
E128 35 B4 PULS B,X,Y,PC
  
```

```

E12A CC 01 16 LDD #0116
E12D ED 84 STD ,X
E12F CC 00 3A LDD #003A
E132 ED 84 STD ,X
E134 CC 02 30 LDD #0230
E137 ED 84 STD ,X
E139 86 03 LDA #003
E13B A7 84 STA ,X
E13D 86 80 LDA #080
E13F A7 08 STA $08,X
E141 39 RTS
  
```

```

Indirections standard sur les opérations physiques pour lecteur 0
E142 16 00 CF LBRA $E214 Fonctions standard
E145 16 00 5F LBRA $E1A1 Lancement du boot
E148 16 03 21 LBRA $E46C Formatage
---- Indirections standard sur les opérations logiques pour lecteur 0
E14B 16 05 15 LBRA $E663
E14E 16 05 3F LBRA $E690
E151 16 06 3A LBRA $E78E
E154 16 05 2C LBRA $E683
E157 16 05 94 LBRA $E6EE
E15A 16 05 EC LBRA $E749
  
```


Contrôleur externe CQ90-028

```

E404 8D 14 BSR $E41A Charge un caractère
E406 A7 C0 STA ,U+ Ecrit le caractère
E408 9B 4A ADDA <$4A Ajourne le
E40A 97 4A STA <$4A checksum
E40C 5A DECB BNE $E404 Pour les 128
E40D 26 F5 BNE $E404 caractères
E40F 8D 09 BSR $E41A Charge un caractère
E411 91 4A CMPA <$4A Si checksum correct,
E413 27 03 BEQ $E418 sort
E415 C6 08 LDB #$08 Code "erreur sur données"
E417 39 RTS
---- Sortie si pas d'erreur de lecture
E418 5F CLRB Pas d'erreur
E419 39 RTS

Lit un caractère sur la piste
E41A A6 84 LDA ,X | Attend fin de
E41C 2A FC BPL $E41A transmission
E41E A6 01 LDA $01,X | Lit le caractère
E420 39 RTS

Recherche du secteur sur la piste
E421 5F CLRB
E422 0F 4A CLR <$4A Efface valeur checksum
E424 17 FE 6C LBSR $E293
E427 A6 08 LDA $08,X
E429 2B 33 BMI $E45E
E42B A6 84 LDA ,X | Attend fin de
E42D 2A F8 BPL $E427 transmission
E42F A6 01 LDA $01,X | Charge le caractère
E431 81 A5 CMPA #$A5 Si identificateur trouvé,
E433 27 06 BEQ $E43B passe
E435 81 16 CMPA #$16 Si espace trouvé,
E437 27 F2 BEQ $E42B caractère suivant
E439 20 E9 BRA $E424 Nouvelle recherche
E43B 97 4A STA <$4A Sauve départ checksum
E43D 8D DB BSR $E41A Charge un caractère
E43F 91 4C CMPA <$4C Si pas poids for secteur,
E441 26 DF BNE $E422 nouvelle recherche
E443 9B 4A ADDA <$4A Ajourne le
E445 97 4A STA <$4A checksum
E447 8D D1 BSR $E41A Charge un caractère
E449 91 4D CMPA <$4D Si pas poids faible sect.,
E44B 26 D5 BNE $E422 nouvelle recherche
E44D 9B 4A ADDA <$4A Ajourne
E44F 97 4A STA <$4A checksum
E451 8D C7 BSR $E41A Charge un caractère
E453 91 4A CMPA <$4A Si checksum correct,
E455 27 05 BEQ $E45C sort
E457 86 04 LDA #$04 Code "erreur de secteur"
E459 16 FF 89 LBRA $E3E5 Sort avec erreur
---- Sortie si secteur trouvé
E45C 4F CLRA Pas d'erreur dans CC
E45D 39 RTS

E45E C1 01 CMPB #$01
E460 27 F5 BEQ $E457
E462 34 04 PSHS B
E464 17 FE 3A LBSR $E2A1
E467 35 04 PULS B
E469 5C INCB
E46A 20 B6 BRA $E422

Formatage de la disquette QDD
E46C 34 34 PSHS Y,X,B |
E46E 17 FD AB LBSR $E21C Initialise registres
E471 DE 4A LDU <$4A Protège numéro
E473 34 40 PSHS U de piste
E475 DE 4C LDU <$4C Protège numéro
E477 34 40 PSHS U de secteur
E479 96 48 LDA <$48
E47B 84 80 ANDA #$80 Préserve vérification
E47D 97 48 STA <$48
E47F 0F 4E CLR <$4E Efface code d'erreur
E481 17 FD 71 LBSR $E1F5 Teste présence disquette
E484 25 03 BCS $E489 Si erreur, sort
E486 17 FD 79 LBSR $E202 Test protection disquette
E489 10 25 00 EF LBCS $E57C Si erreur, sort
E48D 17 FC 9A LBSR $E12A
E490 8D 5A BSR $E4EC Initialise secteur courant
E492 DE 4F LDU <$4F Ptr buffer de secteur
E494 C6 80 LDB #$80 128 octets
E496 86 FF LDA #$FF Valeur pour remplissage
E498 A7 C0 STA ,U+
E49A 5A DECB BNE $E498 Remplit le secteur avec 5FF
E49B 26 FB LDA BNE $E498
E49D 86 01 LDA #$01 1 tentative
E49F 34 02 PSHS A maximum
E4A1 A6 08 LDA $08,X
E4A3 2A FC BPL $E4A1
E4A5 17 FD F9 LBSR $E2A1
E4A8 17 FD D3 LBSR $E27E
E4AB C6 DC LDB #$DC | Temporisation de
E4AD 17 FD C5 LBSR $E275 217140 µs
---- Ecriture de l'amorce de secteur
E4B0 C6 12 LDB #$12 18 caractères à sauve

```

```

E4B2 33 8D 01 53 LEAU $E609,PCR Ptr sur série espaces
E4B6 A6 C0 LDA ,U+ Lit le caractère
E4B8 17 FE EB LBSR $E3A6 Sauve le caractère
E4BB 5A DECB BNE $E4B6 Pour les 18
E4BC 26 F8 BNE $E4B6 caractères
---- Ecriture de l'identificateur de secteur
E4BE 86 A5 LDA #$A5 Valeur départ checksum
E4C0 97 4A STA <$4A Init valeur départ checksum
E4C2 96 4C LDA <$4C lit poids fort n° secteur
E4C4 9B 4A ADDA <$4A Ajourne le
E4C6 97 4A STA <$4A checksum
E4C8 96 4C LDA <$4C Lit poids fort n° secteur
E4CA 17 FE D9 LBSR $E3A6 Sauve le caractère
E4CD 96 4D LDA <$4D Lit poids faible n° secteur
E4CF 9B 4A ADDA <$4A Ajourne le
E4D1 97 4A STA <$4A checksum
E4D3 96 4D LDA <$4D Lit poids faible n° secteur
E4D5 17 FE CE LBSR $E3A6 Sauve le caractère
E4D8 96 4A LDA <$4A Lit le checksum
E4DA 17 FE C9 LBSR $E3A6 Sauve le caractère
E4DD DC 4C LDD <$4C Lit n° de secteur courant
E4DF 10 83 00 05 CMPD #$0005 Si secteur de FAT,
E4E3 27 0F BEQ $E4F4 traitement spécial
E4E5 DE 4F LDU <$4F Ptr sur buffer de secteur
E4E7 20 0F BRA $E4F8 Sauve le secteur
---- Augmentation du compteur de secteur
E4E9 DC 4C LDD <$4C Numéro secteur courant
E4EB 7D FCB $7D >TST $4F5F
E4EC 4F CLRA Numéro de secteur
E4ED 5F CLRB à 0
E4EE C3 00 01 ADDD #$0001 Numéro de secteur + 1
E4F1 DD 4C STD <$4C Mémorise numéro de secteur
E4F3 39 RTS

---- Sauvegarde du secteur
E4F4 33 8D 00 91 LEAU $E589,PCR Ptr sur image de FAT
E4F8 17 FE 86 LBSR $E381 Sauve le secteur
---- Secteur suivant à formater
E4FB 8D EC BSR $E4E9 Augmente n° de secteur
E4FD 10 83 01 91 CMPD #$0191 Si pas secteur 401,
E501 26 AD BNE $E4B0 secteur suivant
E503 C6 01 LDB #$01 Temporisation
E505 17 FD 6D LBSR $E275 de 987 µs
E508 17 FD 7F LBSR $E28A
E50B A6 08 LDA $08,X
E50D 2A FC BPL $E50B
E50F 17 FD 8F LBSR $E2A1
E512 17 FD 7E LBSR $E293
E515 C6 6E LDB #$6E | Temporisation
E517 17 FD 5B LBSR $E275 de 108570 µs
E51A 8D D0 BSR $E4EC
E51C 17 FF 02 LBSR $E421 Recherche le secteur
E51F 10 25 00 4D LBCS $E570 Si introuvable, nouvel essai
E523 DC 4C LDD <$4C Lit n° du secteur courant
E525 10 83 00 05 CMPD #$0005 Si secteur de
E529 26 06 BNE $E531 FAT, comparaison avec
E52B 33 8D 00 5A LEAU $E589,PCR l'image de FAT
E52F 20 02 BRA $E533
E531 DE 4F LDU <$4F Ptr buffer secteur
E533 17 FD 5D LBSR $E293
E536 17 FE E1 LBSR $E41A Charge un caractère
E539 81 5A CMPA #$5A Si identificateur secteur,
E53B 27 06 BEQ $E543 charge le secteur
E53D 81 16 CMPA #$16 Si espace,
E53F 26 F2 BNE $E533 caractère suivant
E541 20 F3 BRA $E536 Sinon, nouvelle recherche
E543 97 4A STA <$4A Initialise départ checksum
E545 C6 80 LDB #$80 128 caractères à vérifier
E547 17 FE D0 LBSR $E41A Charge un caractère
E54A A1 C0 CMPA ,U+ Si caractère correct,
E54C 27 07 BEQ $E555 passe
E54E 86 04 LDA #$04 Code "erreur de secteur"
E550 97 4E STA <$4E Fixe le code
E552 43 COMA Erreur dans CC
E553 20 1B BRA $E570 Sort du programme
E555 9B 4A ADDA <$4A Ajourne le
E557 97 4A STA <$4A checksum de secteur
E559 5A DECB BNE $E547 Pour les 128
E55A 26 EB BNE $E547 caractères
E55C 17 FE BB LBSR $E41A Charge un caractère
E55F 91 4A CMPA <$4A Si checksum incorrect,
E561 26 EB BNE $E54E erreur de secteur
---- Secteur suivant à vérifier
E563 8D 84 BSR $E4E9 Augmente n° de secteur
E565 10 83 01 91 CMPD #$0191 Si pas secteur 401,
E569 26 B1 BNE $E51C secteur suivant
E56B 4F CLRA Pas d'erreur dans CC
E56C 35 02 PULS A Rétablit la pile
E56E 20 0C BRA $E57C Sort du programme
E570 1E 8A EXG A,CC
E572 6A E4 DEC ,S
E574 10 26 FF 29 LBNB $E4A1
E578 1E 8A EXG A,CC
E57A 35 02 PULS A
E57C 17 FD 1D LBSR $E29C
E57F 35 40 PULS U Rétablit numéro
E581 DF 4C STU <$4C de secteur

```


Contrôleur externe CQ90-028

E717	81 10	CMPA	#\$10	en revue,	E76D	5C	INCB	
E719	22 07	BHI	#\$E722	erreur "disque plein"	E76E	E7 E4	STB	,S
E71B	17 F8 E6	LBSR	#\$E004	Charge secteur suivant	E770	20 E5	BRA	\$E757
E71E	86 03	LDA	#\$03	Code "erreur d'entrée-sortie"	E772	E6 61	LDB	\$01,S
E720	20 D8	BRA	#\$E6FA	Boucle avec ou sans erreur	E774	C1 33	CMPB	#49
----	Sortie si erreur "disque plein"				E776	25 DF	BLO	\$E757
E722	86 05	LDA	#\$05	Code "disque plein"	E778	86 05	LDA	#05
E724	16 FF 44	LBRA	#\$E66B	Sort avec erreur	E77A	32 62	LEAS	\$02,S
	Mise à jour du fichier				E77C	16 FE EC	LBRA	\$E66B
E727	9E E7	LDX	<#\$E7		----	Vérifie si bloc libre		
E729	D6 F0	LDB	<#\$F0	Si la sauvegarde	E77F	63 A5	COM	B,Y
E72B	C1 03	CMPB	#\$03	avec écrasement	E781	27 03	BEQ	\$E786
E72D	26 04	BNE	#\$E733	est programmée, le fichier	E783	63 A5	COM	B,Y
E72F	30 8D FE E8	LEAX	#\$E61B,PCR	est nommé "SCRATCH.DOS"	E785	39	RTS	
E733	C6 0A	LDB	#\$0A		----	Test de validité du numéro de bloc		
E735	A6 85	LDA	B,X	Recopie le	E786	32 64	LEAS	\$04,S
E737	A7 A5	STA	B,Y	nom de fichier	E788	5A	DECB	
E739	5A	DECB		dans le catalogue	E789	D7 F9	STB	<#\$F9
E73A	2C F9	BGE	#\$E735		E78B	16 FE E2	LBRA	\$E670
E73C	96 EB	LDA	<#\$EB	Recopie du type de fichier				
E73E	A7 2B	STA	#\$0B,Y	dans le catalogue	E78E	A6 2D	LDA	\$0D,Y
E740	96 EC	LDA	<#\$EC	Recopie du flag de fichier	E790	97 F6	STA	<#\$F6
E742	D6 F6	LDB	<#\$F6	et du numéro de bloc	E792	6F A4	CLR	,Y
E744	ED 2C	STD	#\$0C,Y	courant dans le catalogue	E794	17 FE EC	LBSR	\$E683
E746	16 FF 3A	LBRA	#\$E683	Sauve le secteur de catalogue	E797	25 8B	BCS	\$E724
	Allocation d'un bloc				E799	10 9E ED	LDY	<#\$ED
E749	D6 F6	LDB	<#\$F6	Bloc de départ	E79C	D6 F6	LDB	<#\$F6
E74B	5C	INCB		Départ au	E79E	5C	INCB	
E74C	5C	INCB		numéro de bloc	E79F	A6 A5	LDA	B,Y
E74D	C4 FE	ANDB	#\$FE	pair suivant	E7A1	6F A5	CLR	B,Y
E74F	7D	FCB	#\$7D >TST \$C62A		E7A3	6A A5	DEC	B,Y
E750	C6 2A	LDB	#\$2A	Départ au bloc 42	E7A5	1F 89	TFR	A,B
E752	34 04	PSHS	B	Empile départ ascendant	E7A7	81 C0	CMPA	#\$C0
E754	5A	DECB		Ajuste et empile	E7A9	25 F3	BLO	\$E79E
E755	34 04	PSHS	B	départ descendant	E7AB	20 DE	BRA	\$E78B
E757	E6 61	LDB	#\$01,S	Lit bloc ascendant				
E759	C1 33	CMPB	#49	Si hors champ, passe	Initialise une opération sur un bloc			
E75B	24 04	BHS	#\$E761	au bloc descendant	E7AD	D6 F6	LDB	<#\$F6
E75D	8D 20	BSR	#\$E77F	Teste validité numéro bloc	E7AF	4F	CLRA	
E75F	6C 61	INC	#\$01,S	Bloc ascendant + 1	E7B0	54	LSRB	
E761	E6 E4	LDB	,S	Lit bloc descendant	E7B1	DD FB	STD	<#\$FB
E763	27 0D	BEQ	#\$E772	Si = 0, teste si fin de FAT	E7B3	4C	INCA	
E765	8D 18	BSR	#\$E77F	Teste validité numéro bloc	E7B4	97 F5	STA	<#\$F5
E767	C5 01	BITB	#\$01	S'intéresse toujours	E7B6	24 02	BCC	\$E7BA
E769	26 02	BNE	#\$E76D	en priorité au	E7B8	86 09	LDA	#\$09
E76B	C0 04	SUBB	#\$04	bloc de numéro pair	E7BA	97 FA	STA	<#\$FA
					E7BC	39	RTS	

Contrôleur nanoréseau version III

E000	52	FCC	"R"		----	E0BB	9D 80	JSR	<\$80
E001	45	FCC	"E"			E0BD	33 C9 40 00	LEAU	>\$4000,U
E002	43	FCC	"C"			E0C1	20 18	BRA	\$E0DB
E003	2F	FCB	\$2F						----
E004	7E E6 EB	JMP	\$E6EB			E0C3	AE 43	LDX	\$03,U
E007	7E E6 0B	JMP	\$E60B	Lancement du boot		E0C5	EE 46	LDU	\$06,U
E00A	7E E6 E3	JMP	\$E6E3			E0C7	B6 5F 64	LDA	\$5F64
E00D	7E E2 0A	JMP	\$E20A	Chargement de la FAT		E0CA	27 0F	BEQ	\$E0DB
E010	7E E2 36	JMP	\$E236			E0CC	4A	DECA	
E013	7E E3 33	JMP	\$E333			E0CD	27 EE	BEQ	\$E0BD
E016	7E E2 29	JMP	\$E229			E0CF	4A	DECA	
E019	7E E2 94	JMP	\$E294			E0D0	27 E9	BEQ	\$E0BB
E01C	7E E2 EF	JMP	\$E2EF			E0D2	4A	DECA	
E01F	7E E3 52	JMP	\$E352			E0D3	26 E0	BNE	\$E0B5
E022	7E E1 CD	JMP	\$E1CD			E0D5	EC 9F 5F FC	LDD	>[\$5FFC]
E025	7E E6 7C	JMP	\$E67C			E0D9	33 CB	LEAU	D,U
E028	7E E4 19	JMP	\$E419			E0DB	1F 10	TFR	X,D
E02B	7E E6 9F	JMP	\$E69F			E0DD	30 84	LEAX	,X
						E0DF	27 0D	BEQ	\$E0EE
E02E	C6 B0	LDB	#\$B0			E0E1	86 F0	LDA	#\$F0
E030	8C	FCB	\$8C	>CMPX #C680		E0E3	54	LSRB	
E031	C6 80	LDA	#\$80			E0E4	24 07	BCC	\$E0ED
E033	8C	FCB	\$8C	>CMPX #C6C0		E0E6	8A 04	ORA	#\$04
E034	C6 C0	LDA	#\$C0			E0E8	30 1F	LEAX	-\$01,X
E036	81	FCB	\$81	>CMPA #5F		E0EA	26 01	BNE	\$E0ED
E037	5F	CLRA				E0EC	4C	INCA	
E038	7E E5 5C	JMP	\$E55C			E0ED	39	RTS	
						E0EE	86 F8	LDA	#\$F8
E03B	7E E6 44	JMP	\$E644			E0F0	39	RTS	
E03E	7E E5 E2	JMP	\$E5E2			E0F1	CC 00 F8	LDD	#\$00F8
						E0F4	34 04	PSHS	B
E041	8E E7 CC	LDX	#\$E7CC			E0F6	20 31	BRA	\$E129
E044	EC 05	LDD	\$05,X						
E046	2A 5F	BPL	\$E0A7			E0F8	E6 22	LDB	\$02,Y
E048	F1 60 52	CMPB	\$6052	Numéro de poste		E0FA	58	ASLB	
E04B	26 56	BNE	\$E0A3			E0FB	58	ASLB	
E04D	4C	INCA				E0FC	C4 3C	ANDB	#\$3C
E04E	27 53	BEQ	\$E0A3			E0FE	E7 5F	STB	-\$01,U
E050	E6 05	LDB	\$05,X			E100	9D 8C	JSR	<\$8C
E052	2A F9	BPL	\$E04D			E102	20 0F	BRA	\$E113
E054	EC 06	LDD	\$06,X						
E056	FD 60 54	STD	\$6054			E104	8D 0B	BSR	\$E111
E059	A6 17	LDA	-\$09,X			E106	25 05	BCS	\$E10D
E05B	84 01	ANDA	#\$01			E108	A1 58	CMPA	-\$08,U
E05D	34 02	PSHS	A			E10A	27 01	BEQ	\$E10D
E05F	BD E7 6C	JSR	\$E76C	Initialise registres		E10C	43	COMA	
E062	6F 58	CLR	-\$08,U			E10D	39	RTS	
E064	BD E1 6A	JSR	\$E16A						
E067	25 32	BCS	\$E09B			E10E	86 F8	LDA	#\$F8
E069	81 F0	CMPA	#\$F0			E110	8C 8D B0	CMPX	#\$8DB0
E06B	2F 3E	BLE	\$E0AB			E113	34 02	PSHS	A
E06D	9D 41	JSR	<\$41			E115	CC C0 66	LDD	#\$C066
E06F	E7 55	STB	-\$0B,U			E118	DD D0	STD	<\$D0
E071	7F 5F F3	CLR	\$5FF3			E11A	E7 21	STB	\$01,Y
E074	6F 5D	CLR	-\$03,U			E11C	48	ASLA	
E076	6F 5A	CLR	-\$06,U			E11D	97 D0	STA	<\$D0
E078	8D 7E	BSR	\$E0F8			E11F	CC 10 10	LDD	#\$1010
E07A	25 16	BCS	\$E092			E122	4A	DECA	
E07C	8D BA	BSR	\$E038			E123	27 59	BEQ	\$E17E
E07E	25 12	BCS	\$E092			E125	D5 D0	BITB	<\$D0
E080	BD E5 25	JSR	\$E525			E127	26 F9	BNE	\$E122
E083	E6 5A	LDB	-\$06,U			E129	C6 01	LDB	#\$01
E085	26 0B	BNE	\$E092			E12B	4C	INCA	
E087	24 07	BCC	\$E090			E12C	27 50	BEQ	\$E17E
E089	C6 0F	LDB	#\$0F			E12E	D5 D1	BITB	<\$D1
E08B	E7 5D	STB	-\$03,U			E130	27 F9	BEQ	\$E12B
E08D	BD E4 19	JSR	\$E419			E132	D6 D2	LDB	<\$D2
E090	8D A2	BSR	\$E034			E134	E1 A4	CMPB	,Y
E092	7D 5F F3	TST	\$5FF3			E136	26 46	BNE	\$E17E
E095	27 04	BEQ	\$E09B			E138	4C	INCA	
E097	AD 9F 5F F1	JSR	>[\$5FF1]			E139	27 43	BEQ	\$E17E
E09B	96 C3	LDA	<\$C3			E13B	D6 D1	LDB	<\$D1
E09D	84 FE	ANDA	#\$FE			E13D	2A F9	BPL	\$E138
E09F	AA E0	ORA	,S+			E13F	DC D2	LDD	<\$D2
E0A1	97 C3	STA	<\$C3			E141	E1 23	CMPB	\$03,Y
E0A3	BD E6 9F	JSR	\$E69F			E143	26 39	BNE	\$E17E
E0A6	3B	RTI				E145	A7 22	STA	\$02,Y
						E147	35 01	PULS	CC
E0A7	BD E6 8A	JSR	\$E68A			E149	2B 1D	BMI	\$E168
E0AA	3B	RTI				E14B	26 06	BNE	\$E153
						E14D	96 D2	LDA	<\$D2
E0AB	88 C0	EORA	#\$C0			E14F	A7 C0	STA	,U+
E0AD	85 F0	BITA	#\$F0			E151	25 15	BCS	\$E168
E0AF	26 EA	BNE	\$E09B			E153	D6 D1	LDB	<\$D1
E0B1	9D 20	JSR	<\$20			E155	2B 09	BMI	\$E160
E0B3	20 E6	BRA	\$E09B			E157	86 06	LDA	#\$06
----						E159	4A	DECA	
E0B5	AD 9F 5F F8	JSR	>[\$5FF8]			E15A	27 24	BEQ	\$E180
E0B9	20 20	BRA	\$E0DB						

Contrôleur nanoréseau version III

```

E15C D6 D1      LDB  <$D1
E15E 2A F9      BPL  $E159
E160 DC D2      LDD  <$D2
E162 ED C1      STD  ,U++
E164 30 1E      LEAX -$02,X
E166 26 EB      BNE  $E153
E168 9D 70      JSR  <$70
E16A 4F          CLRA
E16B CC 08 02   LDD  #$0802
E16E 4A          DECA
E16F 27 0F      BEQ  $E180
E171 D5 D1      BITB <$D1
E173 27 F9      BEQ  $E16E
E175 0F D1      CLR  <$D1
E177 96 D1      LDA  <$D1
E179 2B 05      BMI  $E180
E17B A6 22      LDA  $02,Y
E17D 39          RTS

E17E 35 02      PULS A          Rétablit la pile
E180 43          COMA          Erreur dans CC
E181 0E 70      JMP  <$70

E183 BD E6 82   JSR  $E682
E186 EC 23      LDD  $03,Y
E188 97 D2      STA  <$D2
E18A D7 D2      STB  <$D2
E18C A6 A4      LDA  ,Y
E18E 97 D2      STA  <$D2
E190 35 01      PULS CC
E192 2B 21      BMI  $E1B5
E194 26 15      BNE  $E1AB
E196 37 02      PULU A
E198 97 D2      STA  <$D2
E19A 25 19      BCS  $E1B5
E19C D6 D0      LDB  <$D0
E19E 26 0B      BNE  $E1AB
E1A0 86 10      LDA  #$10
E1A2 D6 D0      LDB  <$D0
E1A4 26 05      BNE  $E1AB
E1A6 4A          DECA
E1A7 26 F9      BNE  $E1A2
E1A9 20 0A      BRA  $E1B5
E1AB 37 06      PULU A,B
E1AD 97 D2      STA  <$D2
E1AF D7 D2      STB  <$D2
E1B1 30 1E      LEAX -$02,X
E1B3 26 E7      BNE  $E19C
E1B5 86 9E      LDA  #$9E
E1B7 97 D1      STA  <$D1
E1B9 4C          INCA
E1BA 27 04      BEQ  $E1C0
E1BC D6 D0      LDB  <$D0
E1BE 27 F9      BEQ  $E1B9
E1C0 0E 70      JMP  <$70

*****

E1C2 53 43 52 41 FCC  "SCRATCH DOS"
E1C6 54 43 48 20
E1C7 44 4F 53

E1CD D6 F0      LDB  <$F0
E1CF C1 02      CMPB #$02
E1D1 27 23      BEQ  $E1F6
E1D3 0A F0      DEC  <$F0
E1D5 8D 5F      BSR  $E236
E1D7 25 30      BCS  $E209
E1D9 5D          TSTB
E1DA 27 05      BEQ  $E1E1
E1DC 17 01 54   LBSR $E333
E1DF 25 28      BCS  $E209
E1E1 0C F0      INC  <$F0
E1E3 8D 51      BSR  $E236
E1E5 25 22      BCS  $E209
E1E7 C6 0A      LDB  #$0A
E1E9 9E E7      LDX  <$E7
E1EB A6 85      LDA  B,X
E1ED A7 A5      STA  B,Y
E1EF 5A          DECB
E1F0 2C F9      BGE  $E1EB
E1F2 8D 35      BSR  $E229
E1F4 25 13      BCS  $E209
E1F6 86 02      LDA  #$02
E1F8 97 4C      STA  <$4C
E1FA C6 14      LDB  #$14
E1FC 4F          CLRA
E1FD DD 4A      STD  <$4A
E1FF DC ED      LDD  <$ED
E201 DD 4F      STD  <$4F
E203 8D 24      BSR  $E229
E205 25 02      BCS  $E209
E207 0F F0      CLR  <$F0
E209 39          RTS

E20A 9E ED      LDX  <$ED

```

```

E20C 9F 4F      STX  <$4F
E20E 86 02      LDA  #$02
E210 20 0C      BRA  $E21E

E212 97 E5      STA  <$E5
E214 43          COMA
E215 39          RTS

E216 4F          CLRA
E217 39          RTS

E218 86 03      LDA  #$03
E21A 9E E9      LDX  <$E9
E21C 9F 4F      STX  <$4F
E21E 97 4C      STA  <$4C
E220 C6 14      LDB  #$14
E222 4F          CLRA
E223 DD 4A      STD  <$4A
E225 86 02      LDA  #$02
E227 20 02      BRA  $E22B
E229 86 08      LDA  #$08
E22B 97 48      STA  <$48
E22D 10 9E E9   LDY  <$E9
E230 17 FD D1   LBSR $E004
E233 86 03      LDA  #$03
E235 39          RTS

E236 8D E0      BSR  $E218
E238 25 D8      BCS  $E212
E23A 8E 00 04   LDX  #$0004
E23D 10 9E E9   LDY  <$E9
E240 DE E7      LDU  <$E7
E242 D6 F0      LDB  <$F0
E244 C1 03      CMPB #$03
E246 26 04      BNE  $E24C
E248 33 8D FF 76 LEAU >$E1C2,PCR
E24C 5F          CLR B
E24D C1 0B      CMPB #$0B
E24F 24 24      BHS  $E275
E251 A6 A5      LDA  B,Y
E253 81 FF      CMPA #$FF
E255 27 1B      BEQ  $E272
E257 5C          INCB
E258 A1 C0      CMPA ,U+
E25A 27 F1      BEQ  $E24D
E25C 31 A8 20   LEAY <$20,Y
E25F 30 1F      LEAX -$01,X
E261 26 DD      BNE  $E240
E263 0C 4C      INC  <$4C
E265 96 4C      LDA  <$4C
E267 81 10      CMPA #$10
E269 22 07      BHI  $E272
E26B 17 FD 96   LBSR $E004
E26E 86 03      LDA  #$03
E270 20 C6      BRA  $E238
E272 5F          CLR B
E273 20 1B      BRA  $E290

E275 E6 2B      LDB  $0B,Y
E277 D1 EB      CMPB <$EB
E279 26 F7      BNE  $E272
E27B E6 2C      LDB  $0C,Y
E27D D1 EC      CMPB <$EC
E27F 26 F1      BNE  $E272
E281 D6 4C      LDB  <$4C
E283 A6 2D      LDA  $0D,Y
E285 97 F6      STA  <$F6
E287 0F F5      CLR  <$F5
E289 AE 2E      LDX  $0E,Y
E28B 9F F7      STX  <$F7
E28D 10 9F FA   STY  <$FA
E290 D7 F9      STB  <$F9
E292 20 82      BRA  $E216

E294 10 9E ED   LDY  <$ED
E297 8D 70      BSR  $E309
E299 25 9D      BCS  $E238
E29B D7 F6      STB  <$F6
E29D 17 FF 78   LBSR $E218
E2A0 25 F7      BCS  $E299
E2A2 10 9E E9   LDY  <$E9
E2A5 8E 00 04   LDX  #$0004
E2A8 E6 A4      LDB  ,Y
E2AA 27 21      BEQ  $E2CD
E2AC 86 05      LDA  #$05
E2AE C1 FF      CMPB #$FF
E2B0 27 1B      BEQ  $E2CD
E2B2 31 A8 20   LEAY <$20,Y
E2B5 30 1F      LEAX -$01,X
E2B7 26 EF      BNE  $E2A8
E2B9 0C 4C      INC  <$4C
E2BB 96 4C      LDA  <$4C
E2BD 81 10      CMPA #$10
E2BF 22 07      BHI  $E2C8
E2C1 17 FD 40   LBSR $E004
E2C4 86 03      LDA  #$03

```

Contrôleur nanoréseau version III

E2C6	20 D8	BRA	\$E2A0	E377	85 08	BITA	#\$08
E2C8	86 05	LDA	#\$05	E379	26 02	BNE	#\$E37D
E2CA	7E A2 12	JMP	SA212	E37B	8A 80	ORA	#\$80
E2CD	9E E7	LDX	<\$E7	E37D	84 87	ANDA	#\$87
E2CF	D6 F0	LDB	<\$F0	E37F	A7 84	STA	,X
E2D1	C1 03	CMPB	#\$03	E381	C4 F0	ANDB	#\$F0
E2D3	26 04	BNE	#\$E2D9	E383	54	LSRB	
E2D5	30 8D FE E9	LEAX	>#\$E1C2,PCR	E384	C8 40	EORB	#\$40
E2D9	C6 0A	LDA	#\$0A	E386	EA 84	ORB	,X
E2DB	A6 85	LDA	B, X	E388	E7 80	STB	,X+
E2DD	A7 A5	STA	B, Y	E38A	8C 5F 40	CMPX	#\$5F40
E2DF	5A	DECB		E38D	26 E2	BNE	#\$E371
E2E0	2C F9	BGE	#\$E2DB	E38F	9D 77	JSR	<\$77
E2E2	96 EB	LDA	<\$EB	E391	A6 42	LDA	\$02, U
E2E4	A7 2B	STA	\$0B, Y	E393	48	ASLA	
E2E6	96 EC	LDA	<\$EC	E394	48	ASLA	
E2E8	D6 F6	LDB	<\$F6	E395	48	ASLA	
E2EA	ED 2C	STD	\$0C, Y	E396	2B 02	BMI	#\$E39A
E2EC	16 FF 3A	LBRA	#\$E229	E398	8A 04	ORA	#\$04
E2EF	D6 F6	LDB	<\$F6	E39A	84 74	ANDA	#\$74
E2F1	C1 28	CMPB	#\$28	E39C	A7 42	STA	\$02, U
E2F3	22 0E	BHI	#\$E303	E39E	C6 8B	LDB	#\$8B
E2F5	5D	TSTB		E3A0	7D 5F F6	TST	#\$5FF6
E2F6	27 11	BEQ	#\$E309	E3A3	26 0D	BNE	#\$E3B2
E2F8	A6 A5	LDA	B, Y	E3A5	84 70	ANDA	#\$70
E2FA	81 FF	CMPA	#\$FF	E3A7	A7 42	STA	\$02, U
E2FC	27 2D	BEQ	#\$E32B	E3A9	C6 8F	LDB	#\$8F
E2FE	5A	DECB		E3AB	D4 C3	ANDB	<\$C3
E2FF	C1 28	CMPB	#\$28	E3AD	EA 42	ORB	\$02, U
E301	23 F2	BLS	#\$E2F5	E3AF	D7 C3	STB	<\$C3
E303	CB 02	ADDB	#\$02	E3B1	39	RTS	
E305	C1 51	CMPB	#\$51	E3B2	6D 48	TST	\$08, U
E307	20 ED	BRA	#\$E2F6	E3B4	2A F5	BPL	#\$E3AB
E309	5F	CLRB		E3B6	8A 04	ORA	#\$04
E30A	31 A8 28	LEAY	<\$28, Y	E3B8	A7 42	STA	\$02, U
E30D	86 05	LDA	#\$05	E3BA	20 EF	BRA	#\$E3AB
E30F	C1 28	CMPB	#\$28	E3BC	63 42	COM	\$02, U
E311	10 22 FE FD	LBHI	#\$E212	E3BE	2A A6	BPL	#\$E366
E315	A6 A5	LDA	B, Y	E3C0	9D 41	JSR	<\$41
E317	81 FF	CMPA	#\$FF	E3C2	C6 70	LDB	#\$70
E319	27 0B	BEQ	#\$E326	E3C4	7D 5F F6	TST	#\$5FF6
E31B	50	NEGB		E3C7	27 02	BEQ	#\$E3CB
E31C	A6 A5	LDA	B, Y	E3C9	C6 74	LDB	#\$74
E31E	81 FF	CMPA	#\$FF	E3CB	D4 C3	ANDB	<\$C3
E320	27 04	BEQ	#\$E326	E3CD	53	COMB	
E322	50	NEGB		E3CE	86 84	LDA	#\$84
E323	5C	INCB		E3D0	ED 41	STD	\$01, U
E324	20 E7	BRA	#\$E30D	E3D2	8D 12	BSR	#\$E3E6
E326	CB 28	ADDB	#\$28	E3D4	25 13	BCS	#\$E3E9
E328	31 A8 D8	LEAY	<-\$28, Y	E3D6	CC 20 18	LDD	#\$2018
E32B	6F A5	CLR	B, Y	E3D9	ED 43	STD	\$03, U
E32D	5A	DECB		E3DB	6F 46	CLR	\$06, U
E32E	D7 F9	STB	<\$F9	E3DD	6F 47	CLR	\$07, U
E330	16 FE E3	LBRA	#\$E216	E3DF	6C 45	INC	\$05, U
E333	A6 2D	LDA	\$0D, Y	E3E1	BD E0 37	JSR	#\$E037
E335	97 F6	STA	<\$F6	E3E4	25 03	BCS	#\$E3E9
E337	6F A4	CLR	, Y	E3E6	7E E0 31	JMP	#\$E031
E339	17 FE ED	LBSR	#\$E229	E3E9	39	RTS	
E33C	25 8C	BCS	#\$E2CA	E3EA	A7 5C	STA	-\$04, U
E33E	10 9E ED	LDY	<\$ED	E3EC	6F 58	CLR	-\$08, U
E341	D6 F6	LDB	<\$F6	E3EE	9D 20	JSR	<\$20
E343	5C	INCB		E3F0	A6 41	LDA	\$01, U
E344	A6 A5	LDA	B, Y	E3F2	88 80	EORA	#\$80
E346	6F A5	CLR	B, Y	E3F4	BD E5 27	JSR	#\$E527
E348	6A A5	DEC	B, Y	E3F7	7D 5F F3	TST	#\$5FF3
E34A	1F 89	TFR	A, B	E3FA	27 04	BEQ	#\$E400
E34C	81 C0	CMPA	#\$C0	E3FC	AD 9F 5F F1	JSR	>[\$5FF1]
E34E	25 F3	BLO	#\$E343	E400	BD E6 9F	JSR	#\$E69F
E350	20 DE	BRA	#\$E330	E403	A6 5A	LDA	-\$06, U
E352	D6 F6	LDB	<\$F6	E405	27 0E	BEQ	#\$E415
E354	4F	CLRA		E407	35 10	PULS	X
E355	54	LSRB		E409	A6 5C	LDA	-\$04, U
E356	DD FB	STD	<\$FB	E40B	81 0F	CMPA	#\$0F
E358	4C	INCA		E40D	27 1F	BEQ	#\$E42E
E359	97 F5	STA	<\$F5	E40F	AA E4	ORA	, S
E35B	24 02	BCC	#\$E35F	E411	A7 E4	STA	, S
E35D	86 09	LDA	#\$09	E413	35 99	PULS	CC, DP, X, PC
E35F	97 FA	STA	<\$FA	E415	6C 5A	INC	-\$06, U
E361	39	RTS		E417	35 90	PULS	X, PC
*****				E419	CC 20 10	LDD	#\$2010
E362	FF FF FF FF	FCB	\$\$\$,\$FF,\$FF,\$FF	E41C	BA E7 C7	ORA	#\$E7C7
E366	A6 42	LDA	\$02, U	E41F	84 3F	ANDA	#\$3F
E368	6A 48	DEC	\$08, U	E421	F5 E7 D0	BITB	#\$E7D0
E36A	26 32	BNE	#\$E39E	E424	27 F3	BEQ	#\$E419
E36C	9D 80	JSR	<\$80	E426	4A	DECA	
E36E	8E 40 00	LDX	#\$4000	E427	26 F8	BNE	#\$E421
E371	A6 84	LDA	, X	E429	1A 50	ORCC	#\$50
E373	E6 84	LDB	, X	E42B	7E E7 4A	JMP	#\$E74A
E375	84 0F	ANDA	#\$0F	E42E	BD E6 9F	JSR	#\$E69F
				E431	35 19	PULS	CC, DP, X
				E433	BD E8 06	JSR	#\$E806

Contrôleur nanoréseau version III

E436	C1 03	CMPB	#\$03		E4EA	EC 46	LDD	\$06,U	
E438	26 04	BNE	#\$43E		E4EC	ED 56	STD	-\$0A,U	
E43A	6E 9F 5F FE	JMP	>[\$5FFE]	Relance l'application	E4EE	E3 43	ADDD	\$03,U	
E43E	FC E7 C6	LDD	#\$7C6		E4F0	ED 46	STD	\$06,U	
E441	4C	INCA			E4F2	E6 58	LDB	-\$08,U	
E442	83 00 01	SUBD	#\$0001		E4F4	E7 55	STB	-\$0B,U	
E445	26 FB	BNE	#\$442		E4F6	5C	INCB		
E447	7A 60 57	DEC	#\$6057		E4F7	C4 07	ANDB	#\$07	
E44A	27 EE	BEQ	#\$43A		E4F9	E7 58	STB	-\$08,U	
E44C	1C F0	ANDCC	#\$F0		E4FB	39	RTS		
E44E	34 19	PSHS	X,DP,CC		E4FC	9D 41	JSR	<\$41	
E450	BD E7 6C	JSR	#\$76C	Initialise registres	E4FE	A6 22	LDA	\$02,Y	
E453	E7 5A	STB	-\$06,U		E500	84 0F	ANDA	#\$0F	
E455	10 AE 62	LDY	\$02,S		E502	8D D1	BSR	#\$E4D5	
E458	A6 A4	LDA	,Y		E504	25 0B	BCS	#\$E511	
E45A	4C	INCA			E506	BD E1 04	JSR	#\$E104	
E45B	33 5F	LEAU	-\$01,U		E509	25 06	BCS	#\$E511	
E45D	9D 61	JSR	<\$61		E50B	8D DD	BSR	#\$E4EA	
E45F	A6 C4	LDA	,U		E50D	0E 20	JMP	<\$20	
E461	C6 F0	LDB	#\$F0		E50F	6F 41	CLR	\$01,U	
E463	ED 23	STD	\$03,Y		E511	39	RTS		
E465	E7 55	STB	-\$0B,U		E512	9D 41	JSR	<\$41	
E467	6F 58	CLR	-\$08,U		E514	BD E0 F8	JSR	#\$E0F8	
E469	8D AE	BSR	#\$419		E517	25 F6	BCS	#\$E50F	
E46B	6D 5A	TST	-\$06,U		E519	8D A9	BSR	#\$E4C4	
E46D	27 BF	BEQ	#\$42E		E51B	25 F2	BCS	#\$E50F	
E46F	BD E5 C0	JSR	#\$E5C0		E51D	8D D3	BSR	#\$E4F2	
E472	2B BA	BMI	#\$E42E		E51F	8D EC	BSR	#\$E50D	
E474	9D 9F	JSR	<\$9F		E521	A6 59	LDA	-\$07,U	
E476	C6 90	LDB	#\$90		E523	26 EC	BNE	#\$E511	
E478	9D B1	JSR	<\$B1		E525	A6 41	LDA	\$01,U	
E47A	25 B2	BCS	#\$42E		E527	2B E8	BMI	#\$E511	
E47C	8D 3A	BSR	#\$E4B8		E529	85 F8	BITA	#\$F8	
E47E	9D 38	JSR	<\$38		E52B	26 16	BNE	#\$E543	
E480	CC 80 10	LDD	#\$8010		E52D	48	ASLA		
E483	97 D0	STA	<\$D0		E52E	8E E5 33	LDX	#\$E533	
E485	96 C1	LDA	<\$C1		E531	6E 96	JMP	[A,X]	
E487	30 01	LEAX	\$01,X		E533	E5 11	FDB	#\$E511	
E489	26 04	BNE	#\$E48F		E535	E4 BF	FDB	#\$E4BF	
E48B	88 08	EORA	#\$08		E537	E5 47	FDB	#\$E547	
E48D	97 C1	STA	<\$C1		E539	E5 F5	FDB	#\$E5F5	
E48F	D5 D0	BITB	<\$D0		E53B	E3 BC	FDB	#\$E3BC	
E491	26 F4	BNE	#\$E487		E53D	E0 31	FDB	#\$E031	
E493	BD E0 F1	JSR	#\$E0F1		E53F	E7 5A	FDB	#\$E57A	
E496	25 E6	BCS	#\$E47E		E541	E5 43	FDB	#\$E543	
E498	1F 89	TFR	A,B		E543	6E 9F 5F FA	JMP	>[\$5FFA]	
E49A	54	LSRB			E547	6E 4A	JMP	\$0A,U	
E49B	54	LSRB			E549	9D 24	JSR	<\$24	
E49C	54	LSRB			E54B	43	COMA		
E49D	C4 0E	ANDB	#\$0E		E54C	35 8C	PULS	B,DP,PC	
E49F	84 0F	ANDA	#\$0F		E54E	8D 8E	BSR	#\$E4DE	
E4A1	8E E4 A8	LDX	#\$E4A8		E550	8C 8D 8D	CMPY	#\$8D8D	
E4A4	AD 95	JSR	[B,X]		E553	6A 5B	DEC	-\$05,U	
E4A6	20 D6	BRA	#\$E47E		E555	27 F2	BEQ	#\$E549	
E4A8	E4 FC	FDB	#\$E4FC		E557	BD E4 19	JSR	#\$E419	
E4AA	E4 B8	FDB	#\$E4B8		E55A	20 0C	BRA	#\$E568	
E4AC	E5 11	FDB	#\$E511		E55C	34 0C	PSHS	DP,B	
E4AE	E4 E4	FDB	#\$E4E4		E55E	BD E7 41	JSR	#\$E741	
E4B0	E3 EA	FDB	#\$E3EA		E561	BD E7 6C	JSR	#\$E76C	Initialise registres
E4B2	E5 12	FDB	#\$E512		E564	86 06	LDA	#\$06	
E4B4	E5 11	FDB	#\$E511		E566	A7 5B	STA	-\$05,U	
E4B6	E5 11	FDB	#\$E511		E568	E6 E4	LDB	,S	
E4B8	CC 01 A1	LDD	#\$01A1		E56A	27 16	BEQ	#\$E582	
E4BB	A7 58	STA	-\$08,U		E56C	2A 45	BPL	#\$E5B3	
E4BD	0E 99	JMP	<\$99		E56E	C1 B0	CMPB	#\$B0	
E4BF	BD E0 34	JSR	#\$E034		E570	27 23	BEQ	#\$E595	
E4C2	20 BA	BRA	#\$E47E		E572	2B 2E	BMI	#\$E5A2	
E4C4	6F 59	CLR	-\$07,U		E574	EA 5D	ORB	-\$03,U	
E4C6	A1 58	CMPA	-\$08,U		E576	6F 58	CLR	-\$08,U	
E4C8	27 08	BEQ	#\$E4D2		E578	9D 99	JSR	<\$99	
E4CA	A1 55	CMPA	-\$0B,U		E57A	1E 89	EXG	A,B	
E4CC	26 05	BNE	#\$E4D3		E57C	9D AF	JSR	<\$AF	
E4CE	A7 58	STA	-\$08,U		E57E	25 D3	BCS	#\$E553	
E4D0	6C 59	INC	-\$07,U		E580	35 8C	PULS	B,DP,PC	
E4D2	39	RTS			E582	C6 D0	LDB	#\$D0	
E4D3	43	COMA			E584	E7 24	STB	\$04,Y	
E4D4	39	RTS			E586	8D 38	BSR	#\$E5C0	
E4D5	8D ED	BSR	#\$E4C4		E588	2B C9	BMI	#\$E553	
E4D7	23 04	BLS	#\$E4DD		E58A	9D 9F	JSR	<\$9F	
E4D9	EC 56	LDD	-\$0A,U		E58C	BD E4 F2	JSR	#\$E4F2	
E4DB	ED 46	STD	\$06,U		E58F	9D AF	JSR	<\$AF	
E4DD	39	RTS			E591	25 BE	BCS	#\$E551	
E4DE	8D F9	BSR	#\$E4D9		E593	35 8C	PULS	B,DP,PC	
E4E0	E6 55	LDB	-\$0B,U		E595	9D 97	JSR	<\$97	
E4E2	20 15	BRA	#\$E4F9		E597	BD E1 04	JSR	#\$E104	
E4E4	8D EF	BSR	#\$E4D5		E59A	25 B7	BCS	#\$E553	
E4E6	25 29	BCS	#\$E511						
E4E8	9D A5	JSR	<\$A5						

Contrôleur nanoréseau version III

```

E59C BD E4 EA JSR $E4EA
E59F 4F CLRA
E5A0 35 8C PULS B,DP,PC

E5A2 9D 97 JSR <$97
E5A4 8D 28 BSR $E5CE
E5A6 2B AB BMI $E553
E5A8 9D A5 JSR <$A5
E5AA BD E4 EA JSR $E4EA
E5AD 9D AF JSR <$AF
E5AF 25 9D BCS $E54E
E5B1 35 8C PULS B,DP,PC

E5B3 C6 90 LDB #$90
E5B5 9D 99 JSR <$99
E5B7 CC 01 A0 LDD #$01A0
E5BA A7 58 STA -$08,U
E5BC 9D B1 JSR <$B1
E5BE 20 BE BRA $E57E

E5C0 E6 5F LDB -$01,U
E5C2 CB 03 ADDB #$03
E5C4 C4 3C ANDB #$3C
E5C6 E7 5F STB -$01,U
E5C8 54 LSRB
E5C9 54 LSRB
E5CA EA 24 ORB $04,Y
E5CC 9D 99 JSR <$99
E5CE 9D 38 JSR <$38
E5D0 C6 0F LDB #$0F
E5D2 5A DECB
E5D3 2B 0C BMI $E5E1
E5D5 95 D0 BITA <$D0
E5D7 26 F9 BNE $E5D2
E5D9 95 D0 BITA <$D0
E5DB 26 F5 BNE $E5D2
E5DD 95 D0 BITA <$D0
E5DF 26 F1 BNE $E5D2
E5E1 39 RTS

E5E2 8D 18 BSR $E5FC Passe en RAMA
E5E4 BD E6 B7 JSR $E6B7 Efface zone paramètres
E5E7 C6 04 LDB #$04
E5E9 ED 41 STD $01,U
E5EB CC 1F 40 LDD #$1F40
E5EE ED 44 STD $04,U
E5F0 6C 46 INC $06,U
E5F2 7E E6 79 JMP $E679

E5F5 AE 46 LDX $06,U
E5F7 20 09 BRA $E602 Affiche le message

Affiche un caractère
E5F9 BD E8 03 JSR $E803 Affiche le caractère
E5FC 7E E7 77 JMP $E777 Passe en RAMA

Affiche un message
E5FF BD E8 03 JSR $E803 Affiche le caractère
E602 8D F8 BSR $E5FC Passe en RAMA
E604 E6 80 LDB ,X+ lit le caractère
E606 C1 04 CMPB #$04 Si pas terminateur $04,
E608 26 F5 BNE $E5FF caractère suivant
E60A 39 RTS

Lancement du boot
E60B 8E 9C 40 LDX #$9C40
E60E BD E7 24 JSR $E724
E611 27 0B BEQ $E61E
E613 30 1F LEAX -$01,X
E615 26 F7 BNE $E60E
E617 7F 60 80 CLR $6080 Flag "contrôleur absent"
E61A 6E 9F 00 1E JMP >[$001E] Lance l'application à froid

5FF6 Flag de machine ($00=T07 $02=T07-70)

5FF8-5FF9
5FFA-5FFB
5FFC-5FFD
5FFE-5FFF Vecteur de lancement de l'application à froid

E61E 8E 5F 50 LDX #$5F50 Ptr zone paramètres
E621 8D D9 BSR $E5FC Passe en RAMA
E623 6F 80 CLR ,X+
E625 8C 5F F8 CMPX #$5FF8 Efface zone paramètres
E628 26 F9 BNE $E623
E62A B6 FF F0 LDA $FFFO Lit numéro de machine
E62D 48 ASLA Ajuste, isole et stocke
E62E 84 02 ANDA #$02 le bit de présence d'un
E630 A7 1E STA -$02,X T07-70
E632 CC E5 E1 LDD #$E5E1 Pointe sur un RTS
E635 ED 81 STD ,X++
E637 8C 5F FE CMPX #$5FFE Initialise table vecteurs
E63A 26 F9 BNE $E635

```

```

E63C CC E6 17 LDD #$E617
E63F ED 84 STD ,X
E641 7F 60 58 CLR $6058
E644 BE E0 41 LDX #$E041
E647 BF 60 21 STX $6021
E64A BD E7 70 JSR $E770 Initialise le vecteur
Initialise registres

---- Affiche message de présentation
E64D 8E E6 C5 LDX #$E6C5
E650 8D B0 BSR $E602 Affiche le message

---- Affiche numéro de poste
E652 8D 2E BSR $E682 Lit numéro de poste
E654 C6 2F LDB #'0-1
E656 5C INCB Récupère la
dizaine
E657 80 0A SUBA #10
E659 2A FB BPL $E656
E65B C1 30 CMPB #'0
E65D 27 02 BEQ $E661 Si dizaine <> 0, affiche
E65F 8D 98 BSR $E5F9
E661 1F 89 TFR A,B
E663 CB 3A ADDB #'0+10
E665 8D 92 BSR $E5F9 Affiche l'unité
E667 8D 99 BSR $E602 Affiche le message (CR+LF)
E669 C6 3C LDB #$3C
E66B 8D 4C BSR $E6B9 Efface zone paramètres
E66D 33 4B LEAU $0B,U
E66F 10 AE C4 LDY ,U
E672 86 10 LDA #$10
E674 BD E7 63 JSR $E763
E677 8D 11 BSR $E68A
E679 8E 5F A0 LDX #$5FA0
E67C 7F 60 57 CLR $6057
E67F 7E E4 47 JMP $E447

---- Récupère le numéro de poste
E682 B6 E7 D8 LDA $E7D8 Lit le numéro
de poste réseau
E685 84 1F ANDA #$1F
E687 A7 A4 STA ,Y Initialise n° de poste
E689 39 RTS

E68A CE E7 D0 LDU #$E7D0
E68D CC C1 00 LDD #$C100
E690 ED C4 STD ,U
E692 86 1E LDA #$1E
E694 A7 43 STA $03,U
E696 BD E7 24 JSR $E724
E699 27 12 BEQ $E6AD
E69B 85 10 BITA #$10
E69D 26 F7 BNE $E696
E69F BD E7 24 JSR $E724
E6A2 CC C1 80 LDD #$C180
E6A5 BD E7 53 JSR $E753
E6A8 4A DECA
E6A9 B7 E7 D0 STA $E7D0
E6AC 39 RTS

E6AD B6 60 58 LDA $6058
E6B0 26 04 BNE $E6B6
E6B2 86 82 LDA #$82
E6B4 A7 C4 STA ,U
E6B6 39 RTS

E6B7 C6 1C LDB #$1C
E6B9 CE 5F A0 LDU #$5FA0
E6BC E7 C4 STB ,U
E6BE 5C INCB
E6BF 6F C5 CLR B,U
E6C1 5A DECB
E6C2 26 FB BNE $E6BF
E6C4 39 RTS

Message pour présentation
E6C5 0B 0D FCB $0B,$0D
E6C7 4E 41 4E 4F 52 FCC "NANORESEAU LD USTL V3 p"
E6CC 45 53 45 41 55
E6D1 20 4C 44 20 55
E6D6 53 54 4C 20 56
E6DB 33 20 20 70
E6DF 04 FCB $04

---- Message pour retour de ligne
E6E0 0A 0D 04 FCB $0A,$0D
E6E2 04 FCB $04

Formatage
E6E3 7F 60 4E CLR $604E Efface le code d'erreur
E6E6 86 0A LDA $60A
E6E8 B7 60 48 STA $6048

E6EB 34 76 PSHS U,Y,X,B,A
E6ED CC E0 41 LDD #$E041
E6F0 B3 60 21 SUBD $6021 Si interruption pas
installée, lance le boot
E6F3 27 03 BEQ $E6F8
E6F5 BD E6 0B JSR $E60B
E6F8 BD E7 70 JSR $E770 Initialise registres
E6FB C6 10 LDB #$10 16 octets à effacer
E6FD 8D BA BSR $E6B9 Efface zone paramètres
E6FF 86 08 LDA #$08
E701 E6 36 LDB -$0A,Y

```


Contrôleur nanoréseau version III

E703	C1 08	CMPB	#\$08			
E705	26 02	BNE	#\$E709			
E707	8A 40	ORA	#\$40			
E709	ED 42	STD	\$02,U			
E70B	EC 3D	LDD	-\$03,Y			
E70D	ED 47	STD	\$07,U			
E70F	CC 03 80	LDD	#\$0380			
E712	E7 45	STB	\$05,U			
E714	31 36	LEAY	-\$0A,Y			
E716	33 4A	LEAU	\$0A,U			
E718	BD E7 63	JSR	#\$E763			
E71B	BD E6 79	JSR	#\$E679			
E71E	35 F6	PULS	A,B,X,Y,U,PC			
E720	C6 E0	LDB	#\$E0			
E722	8D 73	BSR	#\$E797			
E724	CC C0 66	LDD	#\$C066			
E727	FD E7 D0	STD	#\$E7D0			
E72A	F7 60 53	STB	#\$6053			
E72D	FC E7 D0	LDD	#\$E7D0			
E730	BC E7 D2	CMPX	#\$E7D2			
E733	10 83 10 00	CMPD	#\$1000			
E737	39	RTS				
E738	8D EA	BSR	#\$E724			
E73A	26 FC	BNE	#\$E738			
E73C	39	RTS				
E73D	E6 58	LDB	-\$08,U			
E73F	E7 24	STB	\$04,Y			
E741	F6 60 53	LDB	#\$6053			
E744	2B 04	BMI	#\$E74A			
E746	8D DC	BSR	#\$E724			
E748	26 FC	BNE	#\$E746			
E74A	CC C1 00	LDD	#\$C100			
E74D	FD E7 D0	STD	#\$E7D0			
E750	CC 40 E6	LDD	#\$40E6			
E753	FD E7 D0	STD	#\$E7D0			
E756	F7 60 53	STB	#\$6053			
E759	39	RTS				
E75A	31 4A	LEAY	\$0A,U			
E75C	FE 5F F4	LDU	#\$5FF4			
E75F	A6 C0	LDA	,U+			
E761	4C	INCA				
E762	44	LSRA				
E763	AE A1	LDX	,Y++			
E765	AF C1	STX	,U++			
E767	4A	DECA				
E768	26 F9	BNE	#\$E763			
E76A	20 04	BRA	#\$E770			
				Initialise registres		
				Initialise les registres		
E76C	C6 E7	LDB	#\$E7			Fixe
E76E	1F 9B	TFR	B,DP			le DP
E770	CE 5F 5F	LDU	#\$5F5F			
E773	10 8E 60 52	LDY	#\$6052			Ptr sur n° de poste
				---- Passe en RAMA pour lecture des paramètres		
E777	F6 E7 C3	LDB	#\$E7C3			
E77A	CA 01	ORB	#\$01			Passe en RAMA
E77C	F7 E7 C3	STB	#\$E7C3			
E77F	39	RTS				
				---- Passe en RAMB		
E780	F6 E7 C3	LDB	#\$E7C3			
E783	C4 FE	ANDB	#\$FE			Passe en RAMB
E785	20 F5	BRA	#\$E77C			
E787	FC 5F F6	LDD	#\$5FF6			
E78A	ED 48	STD	\$08,U			
E78C	86 F0	LDA	#\$F0			
E78E	AE 5E	LDX	-\$02,U			
E790	33 41	LEAU	\$01,U			
E792	26 02	BNE	#\$E796			
E794	86 F8	LDA	#\$F8			
E796	39	RTS				
E797	EA 58	ORB	-\$08,U			
E799	8D A4	BSR	#\$E73F			
E79B	86 F8	LDA	#\$F8			
E79D	20 0B	BRA	#\$E7AA			
E79F	8D 9C	BSR	#\$E73D			
E7A1	8D E4	BSR	#\$E787			
E7A3	20 05	BRA	#\$E7AA			

E7A5	8D 96	BSR	#\$E73D			
E7A7	BD E0 C3	JSR	#\$E0C3			
E7AA	34 02	PSHS	A			
E7AC	7E E1 83	JMP	#\$E183			
E7AF	C6 E0	LDB	#\$E0			
E7B1	34 04	PSHS	B			
E7B3	BD E1 0E	JSR	#\$E10E			
E7B6	A8 E0	EORA	,S+			
E7B8	25 05	BCS	#\$E7BF			
E7BA	A1 58	CMPA	-\$08,U			
E7BC	27 01	BEQ	#\$E7BF			
E7BE	43	COMA				
E7BF	39	RTS				

SOURCES

Manuel de l'assembleur 6809 du TO7 / TO7-70 par Michel Weissberger (CEDIC NATHAN)
Désassemblage des ROM du TO8D par Dominique Van Den Broeck
Microcomputer Components - Data Catalog 1986/87 (SIEMENS)
SY6591/SY6591A Floppy Disk Controller (FDC) (Synertek)
SY1791-02/SY1793-02 Floppy Disk Controller (FDC) (Synertek)
FD179X-02 floppy disk formatter/controller family (Western Digital Corporation)
WD1691 floppy support logic (F.S.L.) (Western Digital Corporation)
Le contrôleur de floppy WD1770 Western Digital (Fiche n°18 - Micro-Systèmes janvier 1986)
Télé/ordinateur système TO9 (Documentation technique SAVEMA)
Télé/ordinateur système TO7-TO7.70-MO5 (Documentation technique SAVEMA)
Extension MO5 (Documentation technique SAVEMA)
Manuel technique des TO8, TO9 et TO9+ par Jean-Claude Mariaccia et Olivier Savin (CEDIC/NATHAN)
Le livre du lecteur de disquette AMIGA (Micro Application)
Lecteur / enregistreur de disquette DD90-352 (Documentation technique COFADEL)
Micro ordinateur MO5 NR (Documentation technique COFADEL)
Documentation technique du TMS 9900 Floppy Disk Controller
Guide du TO8D par Thomson/COFADEL (CEDIC NATHAN)