

XCLSL	1.1	Met le curseur sur la dernière ligne et saute ancien curseur
XCSTO	1.2	// en D0, D1 = x, y //
XCRCL	1.2	Remet l'ancien curseur
XPTET	1.3	Ecrit l'en-tête A0
XCLRT	1.4	Vide ligne de tête et écran
XPLST ₁	1.5	Ecrit A0 sur la dernière ligne // et remet curseur en place
PLMOD1	1.6	Ecrit LM (lignes modifiées) et LT (total)
XSE _D _F	1.7	Met en mode éditeur
XSR _I _F	1.7	Met en mode insertion
PINTA _E	2	Ecrit D0.L non signé, sans blanc en décimal en A0 sur écran
PINT _E _W	3	Ecrit D0 _L // sur D1 octets // D0.L " " " sur écran
SINIT1	4	Vide TK
SINIT2	4	Initialise les pointeurs écriture et bloc SPOINTK 4 met pts source de SEKA
RSEKA1	5.1	Transfert SEKA → TK
XSPNT	5.2	Met le pointeur sur ligne D0
PNLG _P	6 6a	} Met A0 = ad début de la ligne D0 [PNLG utilise les pointeurs connus PNLGO est utilisé pour la initialisation
SINIT4	7	Analyses TK : met nb de lignes et TDHAUT
WSEKAP	8	Transfert TK → SEKA et retour seka VIDELO 8 vide labely ₀ et
XCONVC	9	Conversion (32), 0 → 0
XCONVR	10	Remet les pointeurs d'après le n° des lignes
XCONVL	11	Conversion 0 → (32), 0
RER4 RER RERM PRM	12	} sorties erreurs
XFCHR	13	Cherche car D0 dans ligne A0
XANLGI	14	Analyse ligne A0 : — : — ; — 0
PLGNT ₁ ZBE38 ₁ PLGN ₁	15 15.1 16	} écrit tête de ligne n(?) } écrit la ligne (TRI TPL
XANBK	17	Analyse d'un bloc de ligne pour écriture
PTPRE ₁	19	Déplace A0 sur la ligne précédente A0, D2 → ligne précédente
PTSUIV	20	A0, D2 → ligne suivante

XLISTM	21	FS	Listing up
XLISTD	22		listing à partir de A0, D2
PBKLA	23		Ecrit bloc de lignes
XFORM1	24		Met le format pour n;
XLISTS	25	F6	listing down
R	25	SFS	" répète
P	25	F2	" avant le pointeur
O	25	F1	" depuis le début
F	25	SF1	" jusqu'à fin

WDL5 25.1 (#bloc) liste le bloc (#L) liste à partir de ligne L (# -> 00)

XLISTB	25.1		listing bloc
XLISTL	26		" à partir de ligne L
H	26	F3	" 1/2 down

XLIST7 26a 1 ligne down
 XLISTM4 b n à n-1
 XLISTF4 c liste mémoire

XRPL 27 Remplace la ligne pointée par A6

XINS(A) 27 Insertion de A1 devant le pointeur

XINS 29 " devant A0

XMODP 30 Modif des pointeurs ≥ A0 : +D1 octets
+D2 lignes

XDEL1 31 Efface la ligne pointée

XDEL 32 Efface de A1, D1 à A0, D2

WDXMIN 32.1 Décode commande - bloc - ligne WNEW 32.2 new XNEW -3

XDELBK 33 Efface bloc

WDXR 33.1 Décode commande aR (ditto)

XDIT 34 Répète le bloc

WDXM 34.1 Décode commande aM (move)

XMOVE 35 Déplace le bloc

WDAVR 36 (D) Lit 1er car en A6

WDLT) 36 lit lettre en A6 et conversion

WDAL) 36 lit lettre ou chiffre

WDN 37 décode n en A6

WDN(IN) 38 décode n/ > en A6
 C
 PT

WDNAM	39	décode	non vn	
WDLGS	40	"	lignes $\in \{M, D, A, B, label, \emptyset\}$	
WDLGN	42	"	ligne $\in \{n, ligned\}$	
WDLGD	43	"	ligned $\in \{lignes, lignes(\neq) \pm n\}$	
WDBLK \emptyset	44	"	bloc	
A	48	"	"	
WDBKPT	49	"	bloc (\odot ligne \odot)	WNUMM 49a décode ligne WNUMMH 49b " # ligne
XSBLK \emptyset	49	met	bloc = D5 à D6	
\emptyset	50	"	" = D0 à D2	
WDBK \emptyset A	51	décode	bloc \odot et met bloc	
WDBK \emptyset			bloc	
INSER	52	teste si mode = insertion		← {WPRINT 51a} Sont le bloc sur imprimé
WINSAW	53	Est-il permis d'insérer A6 ?		
XMISF \emptyset	54	Mise en forme de A6 avant insertion		
\emptyset				
WDSEKA	55	Retour	GEMDOS	
ZF636	56	X12		
ZF4FC		X9		
ZF3EG	57	écrit message et attend entrée		
ZF3FA	57	X3		
ZE8E8	58	Lecture de valeurs numériques		
XBKOF	59	Enlève les break pts		
TESCP	59	Teste escape		
HD44	60	Initialise déviations		
ZB9BO	61	Boucle de commande de SEKA		
GET79A	68	mot ETRL + x		
HD43	69	} Commandes		
GET78	71		Fonctions F1 à F10, F1' à F10'	
VINITO	67a	initialise retour éditeur		
ESCAPE	67b	teste touche pour escape (sans attente)		
DECSZAD	72	} commandes seka exécutées directement		
DECEX $\frac{1}{2}$	73			
SEXCP	74	exception les met ou les enlève		
GES10	75	position curseur après n> ou n.		
// 20	76	décode $\square \square \square \square$ (pointeurs M, A, B définis graphiquement)		
KONTR	77	contrôle le source		