IMP NEAR PTR ASYNC NAME DEMOMAIN E1: GROUP PROG SEGMENT BYTE PUBLIC PGROUP E2: PROG 'PROG' E3: ASSUME CS:PGROUP E4: PUBLIC WORK_1, WORK_2, WORK_3, FLAG, FLAG_2 WORK_4, WORK_5, WORK_6, WORK_7, AGDC_SEG PUBLIC ASYNC:NEAR, PUSH_EXE:NEAR, POP_EXE:NEAR, DEMO_INIT:NEAR DMCLO_C_WAIT:NEAR, DMCL1_C_WAIT:NEAR, DMCLO_C_FREE:NEAR EXTRN EXTRN ; EXTRN MES_CL_WAIT:NEAR RECFILL_DEMO:NEAR, CRLFILL_DEMO:NEAR EXTRN DEMO EXE: RECFILL DEMOINEAR, CRLFILL_DEMOINEAR ELPSFILL_DEMOINEAR, TRIFILL_DEMOINEAR, TRAFILL_DEMOINEAR LINE_DEMOINEAR, CL_DEMOINEAR, ELPS_DEMOINEAR CHR_DRAW_DEMO_L:NEAR, CHR_DRAW_DEMO_HINEAR CHR_DRAW_DEMO_L_90INEAR, CHR_DRAW_DEMO_HINEAR MESSAGEINEAR, LINE_DEMO_HINEAR, ELMOINEAR WAITINEAR, MESSAGE_WAITINEAR, GET_PUT_DEMOINEAR PUT_DATA_TRANINEAR, SHRINK_DEMOINEAR, SHRINK_PUTINEAR EXTRN PUSH AX CALL EXTRN EXTRN MOV AL.1 FXTRN OUT EXTRN IN TEST AL,1 EXTRM EXTRN MOV SHRINK_DATA:NEAR, SHRINK_DUCHO:NEAR, SHRINK_PU:N SHRINK_DATA:NEAR, SHRINK_EXE:NEAR, PAINT_DEMO:NEAR FRCOPY_DEMO:NEAR, RECT_DEMO:NEAR, SCROLL_DEMO:NEAR CLIP_DEMO:NEAR, GLIOMAIN:NEAR JZ EXTRN EXTRN MOV EXTRN DEMO_EXE_1: DEMOMAIN PROC NEAR MOV DS. AX MOV MOV ORG 1008 CALL 1 CALL , ;DATA BUFFER FIELD MOV MOV \$1.0 WORK_1 ;0100H CALL DW 0 0 ;0102H MOV \$1,32 WORK 2 DW WORK_3 DW 0 :0104H CALL. 0 ;0106H FLAG DB ;BITO : REAL TIME/WAIT ;BIT1 : SLANT TEXT/ BIT2 : PIXEL MODE/ MOV FLAG 2 DB 0 :01078 CALL WORK_4 ;0108H MOV DW 0 WORK_5 DW 0 ;010AH CALL WORK 6 DW 0 :010CH MOV WORK_7 DW CALL 0 ;010EH AGDC_SEG DW 0 ;0110H ;ENLARGE JUMP TABLE MOV MOV SI,16 CALL : MOV -1-LINE_DEMO_H CS:BYTE PTR FLAG,1 CALL MOV \$1,13 ;WAIT MODE MOV CALL CALL ENLARGE_DEMO CALL MOV \$1,17 MOV MES_CL_WAIT ;MESSAGE(17) \$1,25 CALL MOV CALL : SHRINK CALL MOV MOV CS:BYTE PTR FLAG,1 WAIT MODE MOV \$1.44 MOV \$1.20 CALL. MES_CL_WAIT ;MESSAGE(20) CALL CALL CALL SHRINK_DEMO MOV MOV CALL \$1,22 MES_CL_WAIT :MESSAGE(22) MOV CALL CALL. ;GET, PUT MOV CALL MOV CS: BYTE PTR FLAG,1 ;WAIT MODE MOV \$1,18 ;SCROLL MES_CL_WAIT GET_PUT_DEMO CALL :MESSAGE(18) CALL MOV MOV \$1,19 MOV MES_CL_WAIT :MESSAGE(19) CALL. CALL CALL FR COPY CALL MOV CS:BYTE PTR FLAG,1 ;WAIT MODE MOV \$1,26 MES_CL_WAIT MOV CALL ;MESSAGE(26) CALL FRCOPY DEMO CALL MOV \$1.27 MOV CALL MES_CL_WAIT ;MESSAGE(27) CALL MOV ; PAINT CALL :CLIPPING MOV CS: BYTE PTR FLAG,1 WALT MODE MOV \$1,23 CALL MES_CL_WAIT :MESSAGE(23) MOV MOV CALL PAINT DEMO \$1.33 MOV \$1,24 CALL CALL MES_CL_WAIT :MEESAGE(24) CALL CALL ;CHARACTER CALL MOV CS:BYTE PTR FLAG,0 ;REAL TIME MODE CALL CHR_DRAW_DEMO MOV CS:BYTE PTR FLAG.2 ;REAL TIME MODE (SLANT) CHR DRAW DEMO MOV CALL CS: BYTE PTR FLAG.0 MOV REAL TIME MODE MOV CALL CHR_DRAW_DEM0_90 CALL CS:BYTE PTR FLAG,1 WAIT MODE MOV

JMP NEAR PTR DEMO EXE JMP NEAR PTR DEMO_INIT ;0115H ;0118H JMP NEAR PTR GLIOMAIN :011BH < DEMONSTRATION > PUSH EXE :ENABLE /CSIR/ OD1H.AL AL, OD1H AX.ODOOOH DEMO_EXE_1 ;PC-XA/XL -- 8000H :PC9800 -- D000H AX,8000H OPEN MEMORY WINDOW CS:WORD PTR AGDC_SEG,AX CS:BYTE PTR FLAG,0 REAL TIME MODE DMCLO C WAIT DEMO_THRU CS:BYTE PTR FLAG,1 WAIT MODE MES_CL_WAIT ;MESSAGE(0) MES_CL_WAIT :MESSAGE(32) ;LINE, RECTANGLE, CIRCLE, ELLIPSE CS:BYTE PTR FLAG,0 ;REAL TIME MODE VECT DEMO CS:BYTE PTR FLAG,1 WAIT MODE VECT DEMO CS:BYTE PTR FLAG,0 REAL TIME MODE VECT_DEMO CS:BYTE PTR FLAG.1 WALT MODE MES CL WAIT :MESSAGE(16) CS:BYTE PTR FLAG.0 REAL TIME MODE -2-MES_CL_WAIT ;MESSAGE(13) CHR_DRAW DEMO WAIT MODE (SLANT) CS:BYTE PTR FLAG,3 MES_CL_WAIT CHR_DRAW.DEMO CS:BYTE PTR FLAG,1 :MESSAGE(25) ;WAIT MODE MES_CL_WAIT ;MESSAGE(44) CHR_DRAW_DEMO_90 CS:BYTE PTR FLAG,0 :REAL TIME MODE CHR_DRAW_DEMO CS:BYTE PTR FLAG,2 CHR DRAW DEMO REAL TIME MODE (SLANT) CS:BYTE PTR FLAG,0 ;REAL TIME MODE CHR DRAW DEMO 90 CS:BYTE PTR FLAG,1 WAIT MODE SI,31 MES_CL_WAIT ;MESSAGE(31) SCROLL DEMO DMCLO C WAIT ;RECFILL, CRLFILL, ELPSFILL, TRIFILL, TRAFILL CS:BYTE PTR FLAG,0 ;REAL TIME MODE FILL DEMO CS:BYTE PTR FLAG,1 WAIT MODE FILL DEMO CS: BYTE PTR FLAG,0 :REAL TIME MODE FILL DEMO CS:BYTE PTR FLAG,1 ;WAIT MODE MES_CL_WAIT ;MESSAGE(33) CLIP DEMO DMCLO_C_WAIT DEMO THRU :DRAWING SPEED COMPARISON UNDER COMMON N88BASIC (PC-XL VS. AGDC) CS:BYTE PTR FLAG,1 ;WAIT MODE SI,42 MES_CL_WAIT ;MESSAGE(42) MOV \$1,43

:0112H

MES_CL_WAIT POP_EXE ;MESSAGE(43) DMCLO_C_WAIT CALL CALL FRCOPY DEMO DMCLO_C_WAIT JMP CALL ; CALL CALL PAINT DEMO VECT DEMO: DMCLO_C_WAIT CS:BYTE PTR FLAG,0 CALL LINE_DEMO Rect_demo ;REAL TIME MODE CALL MOV CHR DRAW_DEMO CALL CALL CALL DMCLO_C_WAIT MOV CS:BYTE PTR FLAG,2 CHR_DRAW_DEMO :REAL TIME MODE (SLANT) CALL CRL DEMO CALL CS:BYTE PTR FLAG,0 CHR_DRAW_DEMO_90 SCROLL_DEMO CALL ELPS_DEMO MOV ;REAL TIME MODE RET CALL CALL CHR DRAW DEMO: CALL DMCLO_C_WAIT CHR_DRAW_DEMO_L FILL_DEMO CLIP_DEMO CALL CALL MOV \$1,14 CALL MES_CL_WAIT CHR_DRAW_DEMO_H CHR_DRAW_DEMO_90_1 :MESSAGE(14) CALL. CALL DMCLO_C_WAIT CALL RET JMP DEMOMALN ENDP CHR_DRAW_DEM0_90: PROG ENDS CALL CHR_DRAW_DEMO_L_90 END SI.14 MOV CALL MES_CL_WAIT ;MESSAGE(14) CALL CHR_DRAW_DEMO_H_90 CHR_DRAW_DEMO_90_1: MOV AX,0 BX.29 MOV CX,1119 MOV DX,-29 DMCLO_C_FREE MOV CALL MOV \$1,15 MES_CL_WAIT :MESSAGE(15) CALL RET FILL DEMO: RECFILL_DEMO CALL CALL CRLFILL DEMO ELPSFILL DEMO TRIFILL_DEMO CALL CALL TRAFILL_DEMO RET DEMO_THRU: MOV CS:BYTE PTR FLAG,0 ;REAL TIME MODE VECT_DEMO LINE DEMO H CALL CALL. ENLARGE_DEMO DMCLO_C_WAIT SHRINK_DEMO CALL CALL. CALL CALL DMCLO_C_WAIT GET_PUT_DEMO CALL -5--6-EQU NAME EQUATE_DEMO XS 48H GROUP PROG SEGMENT BYTE PUBLIC PGROUP GROUP YS XE EQU 4AH 'PROG' EQU 4CH PROG ASSUME CS:PGROUP YE XC EQU 4EH 50H EQU ; PUBLIC EADORGL, EADORGH, dADORG, EADIL, EADIH, dADI PUBLIC EAD2L, EAD2H, dAD2, PDISPSL, PDISPSH, PDISPDL, PDISPDH PUBLIC PMAX, MODIO, PTNPL, PTNPH, STACKL, STACKH, BANK, CTRL PUBLIC STATUS, IRR, X, Y, DXX, DY, XS, YS, XE, YE, XC, YC PUBLIC DHH, DV, PITCHS, PITCHD, STMAX, PLANES, PTNCNT PUBLIC ACLMIN, YCLMIN, XCLMAX, YACHTAX, MAGETC, COM PUBLIC DISP_FLAGS, DISP_PITCH, DADL, DADH_VC PUBLIC GCSRX, GCSRYS, GCSRYE, SYNC PUBLIC WINDOW_SEG_XL, WINDOW_SEG_98 : YC EQU 52H DHH EOU 54H DV PITCHS 56H EQU EQU 58H PITCHD EQU 5AH STMAX EQU 5CH PLANES PTNCNT EQU 5EH EQU 60H XCLMIN EQU 62H YCLMIN FOU 64H XCLMAX EQUATE DEMO PROC NEAR 66H EQU YCLMAX EQU 68H : MAGETC EQU 6CH AGDC ADDRESS TABLE COM EQU DISP FLAGS EQU 6EH EQU 70H DISP_PITCH 72H EQU EADORGL EQU EADORGH EQU 74H 76H 0H DADL EOU DADH_WC EQU 2H GCSRX EQU GCSRYS EQU dADORG EQU 3H 78H 748 4 H EAD11. FOU 6H GCSRYE EQU 7CH EAD1H EQU dAD1 EAD2L EQU 7 H SYNC EQU 7EH 88 EQU EAD2H EQU MISCELLANEOUS EQUATE OAH ; dAD2 EQU PDISPSL EQU OBH WINDOW_SEG_XL EQU 8000H OCH PDISPSH EQU OEH WINDOW_SEG_98 EQU ODOOOH PDISPDI, EOU 10H : PDISPDH EQU 12H PMAX MOD10 EQUATE_DEMO EQU 14H ENDP PROG ENDS EOU 16H PTNPL END EQU 18H PTNPH EOU 1AH STACKL EQU 1CH STACKH EQU 1EH BANK EQU 3CH CTRL EQU 3DH STATUS EQU 3CH IRR FOU 3EH EQU X 40H EQU 42H DXX 44H EQU 46H

DY

EQU

DEMO_INIT NAME SAD_GEN: PGROUP GROUP PROG SEGMENT BYTE PUBLIC PUSH DS AX,54FOH ; PAGE ADDRESS SEGMENT 'PROG' PROG MOV ASSUME CS:PGROUP MOV DS,AX :MESSAGE SEGMENT MOV AX.5500H PUBLIC DEMO INIT MOV ES.AX MOV 51,0 EXTRN PUSH EXE:NEAR, POP EXE:NEAR, DMCL1 C:NEAR MOV D1.0 SAD GEN 2: CTRL:BYTE, AGDC_SEG:WORD BANK:BYTE, DISP_FLAGS:WORD, SYNC:WORD, DISP_PITCH:WORD DADL:WORD, GCSRX:WORD, GCSRYS:WORD, GCSRYE:WORD EXTRN MOV AL,OCH EXTRN SAD_GEN_1: SCASB EXTRN EADORGLIVORD, CADARCHIVORD, GUSARISHUMD, GUSARISHUMD EADORGLIVORD, EADORGHIVORD, DADORGIVORD, DADH WCIVORD PDISPSLIVORD, PDISPSHIVORD, PDISPDLIVORD, PDISPDHIVORD PMAXIVORD, STACKLIVORD, STACKHIVORD, STMAXIVORD CHECK IF "PAGE MARK" EXTRN JNZ SAD_GEN_1 EXTRN INC DI EXTRM MOV [SI],DI STORE PAGE ADDRESS PITCHS:WORD, PITCHD:WORD XCLMIN:WORD, YCLMIN:WORD, XCLMAX:WORD, YCLMAX:WORD MAGETC:WORD, STATUS:WORD EXTRN INC SI SI INC EXTRN MOV AL,25H EXTRN SCASB DEMO_INIT PROC NEAR SAD_GEN_2 ;CHECK IF "END MARK '%" JNZ. POP DS RET < INITIALIZE AGDC, DISPLAY MEMORY, AND WORKING AREA > < INITIALIZE AGDC > ; PUSH AX PUSH_EXE CALL MOV AL,1 INIT_AGDC: ;ENABLE /CSIR/ DS:BYTE PTR CTRL,3 ;CTRL=3 SOFTWARE RESET OD1H.AL OUT MOV AL, OD1H PROCESSOR ABORT IN DS:BYTE PTR BANK,1 DS:WORD PTR DISP_FLAGS,8630H DS:WORD PTR DISP_FLAGS,8632H DS:WORD PTR SYNC,4 ;BANK=1 AL,1 AX,0D000H TEST MOV ;DISP_FLAGS=8630H ;DISP_FLAGS=8632H MOV MOV JZ MOD_9801 ;CHECK IF 98XA/9801 MOV ; HS=4 MOV AX.8000H MOV DS:WORD PTR SYNC,8 ;HBP=8 MOD_9801: MOV ; HH=22H DS:WORD PTR SYNC,22H DS:WORD PTR SYNC,45H MOV DS,AX **COPEN MEMORY WINDOW** MOV ;98XA/XL -- 8000H ;9801XX -- D000H HD=45H MOV DS:WORD PTR SYNC,6 ;HFP=6 MOV CS:WORD PTR AGDC SEG.AX DS:WORD PTR SYNC,5 DS:WORD PTR SYNC,16H MOV MOV : VS=5 ;VBP=16H MOV DS:WORD PTR SYNC,177H DS:WORD PTR SYNC,09H CALL INIT AGDC MOV : LF=177H DMCL1_C TILE_TRAN :VFP=9H CALL MOV DS:WORD PTR DISP_FLAGS,8630H ;DISP_FLAGS=8630H CALL MOV DS:WORD PTR DISP_PITCH, 46H DS:WORD PTR DADL,0 CALL SAD_GEN MOV :DISP PITCH=46H ;DADL=0 POP EXE JMP. MOV DS:WORD PTR DADH_WC,4500H DS:WORD PTR GCSRX,0C020H ;WC=45H,DADH=0 ;GCSRX=C020H MOV MOV < MESSAGE PAGE ADDRESS TABLE GENERATION > DS:WORD PTR GCSRYS, 100H GCSRYS=100H MOV DS:WORD PTR GCSRYE, 110H GCSRVE=110H MOV :EADORGL=OCCCEH • DS:WORD PTR EADORGL. OCCCEH MOV -1--2-DS:WORD PTR EADORGH,O DS:WORD PTR DADORG,O 7EEFH, 34A5H, 1C63H, 0C21H OFFFFH, 6DEFH, 34A5H, 1C63H MOV ·FADORGH=0 DW :TILE(0)-R ;DADORG=0 MOV DW DS:WORD PTR PDISPSL,0 :PDISPSI=0 OC21H,OFFFFH, GDEFH, 34A5H MOV DW DS:WORD PTR PDISPSH.1 1C63H, 0C21H,0FFFFH,0EF7BH 0888H, 0000H, 0000H, 0401H :PDISPSH=1 MOV nu DS:WORD PTR PDISPDL,0 ;PDISPDL=0 :TILE(0)-G MOV DW MOV DS:WORD PTR PDISPDH,1 :PDISPDH=1 DW 20A2H, 0000H, 1080H, 0400H 0020H, 0445H, 0000H, 1400H MOV DS:WORD PTR PMAX.4 :PMAX=4 DW ;STACKL=OFFFFH DS:WORD PTR STACKL, OFFFFH 0020H, 0001H, 2208H, 0410H MOV DW 28AAH, 0401H, 0020H, 0401H 0AAAAH, 0401H, 10A0H, 0401H 0020H, 5555H, 0020H, 1401H DS:WORD PTR STACKH,2 DS:WORD PTR STMAX,7FFH ;STACKH=2 ;STMAX=7FFH :TILE(0)-B MOV DW MOV DW DS:WORD PTR PITCHS,46H ;PITCHS=46H MOV DW :PITCHD=46H MOV DS:WORD PTR PITCHD,46H DW 0020H, 0401H, 0AAAAH, 0411H DS:WORD PTR XCLMIN,0 ;XCLMIN=0 MOV DS:WORD PTR YCLMIN.0 :YCLMIN=0 DW 1084H, 0842H, 0000H, 0000H :TILE(1)-R MOV DS:WORD PTR XCLMAX,1119 :XCLMAX=1119 0000H, 294AH, 1004H, 0842H 0000H, 0000H, 294AH, 1084H MOV DW DS:WORD PTR YCLMAX,749 ;YCLMAX=749 DW MOV :MAGV=15.MAGH=15 0842H, 0000H, 0000H, 294AH 6D6BH, 30A5H, 1C63H, 0C21H MOV DS:WORD PTR MAGETC, 1FFH DW ; NON-CLIP ;TILE(1)-G DW RET DW OFBFFH, 54A5H, 20A1H, 1421H DU OC21H, OFBFFH, 54A5H, 2421H 1021H, 0C21H, 0FBFFH, 14A5H DW 0000H, 0000H, 0000H, 0000H 0000H, 0000H, 0000H, 0000H 0000H, 0000H, 0000H, 0000H 0000H, 0000H, 0000H, 0000H < TILING PATTERN TRANSFER > :TILE(1)-B DW DW DW DW TILE TRAN: DS:WORD PTR STATUS,3 DW OBBBBH, OFFFFH, OEEEEH, OFFFFH ;TILE(2)-R TEST OBBBBH.OFFFFH, OEEEEH, OFFFFH CHECK IF PPBUSY=1/0 JNZ TILE TRAN DW ;CHECK IF DPBUSY=1/0 OBBBBH, OFFFFH, OEEEEH, OFFFFH DW OBBBBH, OFFFFH, OEEEEH, OFFFFH 1111H, 0000H, 4444H, 0000H 1111H, 0000H, 4444H, 0000H PUSH DS DW TILE(2)-G AX.CS MOV DW DW MOV DS,AX MOV SI, OFFSET TILING_DATA AX, WORD PTR AGDC_SEG DW 1111H, 0000H, 4444H, 0000H 1111H, 0000H, 4444H, 0000H MOV DW 4444H, 0000H, 1111H, 0000H ;TILE(2)-B MOV ES,AX DW DI.OFOOOH 4444H, 0000H, 1111H, 0000H 4444H, 0000H, 1111H, 0000H MOV DW CX.300H TILING PATTERN LENGTH DW MOV 4444H, 0000H, 1111H, 0000H MOV AL,2 DW OD1H.AL :ENABLE #CSDM# OUT REPZ MOVSW DW ODDDDH, 5555H, 7777H, 5555H ;TILE(3)-R DW ODDDDH, 57D5H, 77F7H, 57D5H MOV AL,1 OD7DDH, 53D5H, 7077H, 5555H ;ENABLE #CSIR# OD1H.AL OUT DW ODDDDH, 5555H, 7777H, 5555H POP D₩ DS :TILE(3)-G RET DW 2222H, 8888H, 8888H, 2222H DW 2222H, 8808H, 8008H, 2022H DW 2022H, 8008H, 8088H, 2222H DW 2222H, 8888H, 8888H, 2222H OAAAAH, OFFFFH, OAAAAH, OFFFFH ;TILE(3)-B < TILING DATA > DW OAAAAH, OFFFFH, OA7EAH, OF7FFH OA7EAH, OF3FFH, OAOAAH, OFFFFH nu DW TILING_DATA: DW OAAAAH, OFFFFH, OAAAAH, OFFFFH

:			DW	0000H, 0000H, 0000H, 0000H	;
DW DW	4149H, 8622H, 0C22H, 8400H 0CA11H, 556EH, 90B1H, 0802H	;TILE(4)-R	; DW	0000H, 0000H, 0000H, 0000H	;TILE(8)-R
DW	0800H,0A755H, 69E2H, 3C02H	;	DW	0000H, 0000H, 0000H, 0000H	;
D W D W	8442H, 0801H, 5BEBH, 4629H 1C6EH,0AA96H, 5509H, 2C21H	; ;TILE(4)-G	DW DW	0000H, 0000H, 0000H, 0000H 0000H, 0000H, 0000H, 0000H	
DW	6565H, 00CBH,0B481H, 4528H	;	DW	8220H, 559DH,0A228H, 1095H	;TILE(8)-G
DW DW	20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,0E510H		DW DW	0000H, 8431H, 2220H,0D79DH 2208H, 0000H, 8220H, 579DH	
DW	20D8H, 9603H, 0420H, 0400H	;TILE(4)-B	DW	0A228H, 1091H, 0000H, 4210H	;
D W D W	0D28BH, 2C4AH, 90A5H, 1041H 0400H, 2D40H, 28ABH, 9606H		D W D W	598CH, 2242H, 08COH, 0842H 085ADH, 5946H, 1088H, 2842H	;TILE(8)-B
DW	0020H, 0081H, 0A2AH, 4C43H	;	DW	0842H,0B4ADH, 5942H, 2042H	;
; DW	3118H, 1008H, 0700H, 0300H	;TILE(5)-R	DW -	0800H, 0842H,0B5ADH, 294AH	;
D₩	OC30DH,OC003H, 4001H, 4270H	;	Ďw	OFFFFH,OFFFFH,OFFFFH,OFFFFH	;TILE(9)-R
D W D W	1718H, 0380H,0C100H, 4000H 0061H, 84F1H,0C639H, 6310H		D W D W	OFFFFH,OFFFFH,OFFFFH,OFFFFH OFFFFH,OFFFFH,OFFFFH,OFFFFH	1
DW	7FFFH, 1008H,0FFFFH, 0300H	;TILE(5)-G	DW	OFFFFH,OFFFFH,OFFFFH,OFFFFH	;
D W D W	OF7FFH, OCOO3H, OFFF7H, 4270H OFFFFH, 0380H, OFFDFH, 4000H		D W D W	7777H, OAAAAH, ODDDDH, OAAAAH 7777H, OAAAAH, ODDDDH, OAAAAH	;TILE(9)-G ;
DW	OF7FFH, 84F1H, OFFFFH, 6310H		DW	7777H, OAAAAH, ODDDDH, OAAAAH	÷
D W D W	3BBAH, OFFFFH, OAFAAH, OEFFFH OE3AFH, OFFEFH, OEAA3H, OFFFFH	;TILE(5)-B	D W D W	7777H, OAAAAH, ODDDDH, OAAAAH ODDDDH, OOOOH, 5555H, OOOOH	, ;TILE(9)-B
D W D W	OBFBAH, OFFBFH, OEB8AH, OEFFFH OA2EBH, OFFFFH, OEEBBH, OFFFEH	:	D W D W	ODDDDH, 0000H, 5555H, 0000H ODDDDH, 0000H, 5555H, 0000H	2
;		,	DW	ODDDDH, 0000H, 5555H, 0000H	;
D W D W	OAAAAH,OFFFFH,OBBBBH,OFFFFH OAAAAH,OFFFFH,OBBBBH,OFFFFH	;TILE(6)-R	; DW	OF914H, 677CH,0E6F6H, 68EFH	;TILE(A)-R
D₩	OAAAAH, OFFFFH, OBBBBH, OFFFFH		DW	OE5A5H,OF6E3H,OC86EH, 3AADH	;
D W D W	OAAAAH,OFFFFH,OBBBBH,OFFFFH 5555H, 0000H, 4444H, 0000H	; ;TILE(6)-G	D W D W	OB472H, OD2FH, O81EH,OFE29H OA9BBH,OF77EH, 3396H, OD23H	:
D₩	5555H, 0000H, 4444H, 0000H	;	DW	OFFF7H, OE75FH, OF6FEH, OE8E7H	;TILE(A)-G
D W D W	5555H, 0000H, 4444H, 0000H 5555H, 0000H, 4444H, 0000H	;	D W D W	OE5F3H, 7DEBH,OFFEFH, 7FFFH OFB32H, 3D1FH, OE8EH,OFF5BH	;
DW	5555H, 0000H, 5555H, 0000H 5555H, 0000H, 5555H, 0000H	;TILE(6)-B	D W D W	OFDBFH, 75FFH,OFFFEH, 7DEBH OB800H, 4574H,OA6A2H, 40C5H	; ;TILE(A)-B
D W D W	5555H, 0000H, 5555H, 0000H	;	DW DW	OAOAOH, 0F641H, 886AH, 308DH	;
₽₩	5555H, 0000H, 5555H, 0000H	;	D W D W	OB422H, O50DH, O81EH, 5621H OA8AAH,OD756H, 2296H, O501H	:
, DW	0000H, 0000H, 0000H, 0200H	;TILE(7)-R	;		,
D W D W	0600H, 0700H, 0700H, 0784H 06E4H, 07F8H, 03A0H, 01E0H	:	DW DW	4000H, 8008H, 0000H, 0000H 0000H, 0000H, 0000H, 2020H	;TILE(B)-R
D₩	0000H, 0000H, 0000H, 0000H	;	DW	0000H, 0280H, 0000H, 0200H	1
D W D W	112AH, 2240H, 48C8H, 9311H 2222H, 4244H, 8088H, 1A19H	;TILE(7)-G	D W D W	0000H, 0022H, 0000H, 0000H 0EAAAH,0D559H,0AAA8H, 5575H	; TILE(B)-G
DW	2102H, 6084H, 8050H, 9411H		DW	2AAAH, 1575H,0AAA2H, 7535H	į
D W D W	2202H, 0644H, 8CODH, 1012H 0000H, 0000H, 0000H, 0000H	, ;TILE(7)-B	DW DW	OAAA2H, 57C5H,OBE8AH, 5615H OAOBAH, 557FH,OAAB3H, 5546H	;
DW DW	0400H, 0100H, 0500H, 0084H 0444H, 0228H, 0100H, 00A0H	:	D W D W	4000H,0AAA8H, 0000H, 2A88H 0000H, 028AH, 0050H,0AAE2H	;TILE(B)-B
0*	04441, 02201, 01001, 00001	3	0*	00001, 02041, 00001,044621	,
	-5-			-6-	
NU		°.			
DW DW	-5- 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;	DW DW	-6- 0B422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H	į
DW ;	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;	DW ;	OB422H, O50DH, O81EH, 5621H OA8AAH,OD756H, 2296H, O501H	; ; ;Tile(b)-g (A)
DW ; TILING_DATA_ DW	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H	: : : TILE(0)-R (0)	DW ; DW DW	OB422H, O50DH, O81EH, 5621H OA8AAH,OD756H, 2296H, O501H OEAAAH,OD559H,OAAA8H, 5575H 2AAAH, 1575H,OAAA2H, 7535H	: :TILE(B)-G (A)
DW ; TILING_DATA_	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS:		DW ; DW	OB422H, O50DH, O81EH, 5621H OA8AAH,OD756H, 2296H, O501H OEAAAH,OD559H,OAAA8H, 5575H	:TILE(B)-G (A)
DW ; TILING_DATA_ DW DW	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H		DW ; DW DW DW ;	0B422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H	;
DW ; TILING_DATA_ DW DW DW DW ; DW	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,0FFFFH, 6DEFH, 34A5H 1C63H, 0C21H,0FFFFH,0EF7BH 6D6BH, 30A5H, 1C63H, 0C21H		DW ; DW DW DW ; DW DW	0B422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 575H,0E8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H, 028AH, 0050H,0AA22H	TILE(B)-G (A)
DW ; TILING_DATA_ DW DW DW DW ;	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,0FFFFH, 6DEFH, 34A5H 1C63H, 0C21H,0FFFFH,0EF7BH 6D6BH, 30A5H, 1C63H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H	TILE(0)-R (0)	DW ; DW DW DW ; DW DW DW DW	0B422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0050H,0AAE2H 0101H,0AAAAH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH	;
DW ; TILING_DATA DW DW DW ; DW DW DW	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,0FFFFH, 6DEFH, 34A5H 1C63H, 0C21H,0FFFFH,0EF7BH 6D6BH, 30A5H, 1C63H, 0C21H	TILE(0)-R (0)	DW ; DW DW DW ; DW DW	0B422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 575H,0E8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H, 028AH, 0050H,0AA22H	;
DW ; TILING_DATA. DW DW DW ; DW DW DW ; DW DW ; DW	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7EEFH, 34A5H, 1CG3H, 0C21H OFFFFH, 6DEFH, 34A5H, 1CG3H 0C21H,OFFFFH, 6DEFH, 34A5H 1CG3H, 0C21H,OFFFFH,0EF7BH 6D6BH, 30A5H, 1CG3H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,OFBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H	TILE(0)-R (0)	DW ; DW DW DW ; DW DW DW DW	0B422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0050H,0AAE2H 0101H,0AAAAH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH	;
DW ; TILING_DATA DW DW DW ; DW DW DW ; DW DW ; DW DW ; DW ; DW ; DW	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7EEFH, 34A5H, 1C63H, OC21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,OFFFFH, 6DEFH, 34A5H 1C63H, 0C21H,OFFFFH,OEF7BH 6D6BH, 30A5H, 1C63H, OC21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,0FBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, OC22H, 8400H 0CA11H, 556EH, 90B1H, 0802H	;TILE(0)-R (0) ; ; ; ; ; ; ; ; ; ; ;	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW ; TILING_DATA. DW DW DW ; DW DW DW ; DW DW ; DW	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7EEFH, 34A5H, 1CG3H, 0C21H OFFFFH, 6DEFH, 34A5H, 1CG3H 0C21H,OFFFFH, 6DEFH, 34A5H 1CG3H, 0C21H,OFFFFH,0EF7BH 6D6BH, 30A5H, 1CG3H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,OFBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H	;TILE(0)-R (0) ; ; ; ; ; ; ; ; ; ; ;	DW ; DW DW DW ; DW DW DW ; ; DW DV ;	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW ; TILING_DATA DW DW ; DW ; DW DW ; DW DW ; DW ; DW ;	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW : TILING_DATA DW DW DW : DW DW DW DW DW DW DW DW DW DW	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7EEFH, 34A5H, 1CG3H, OC21H OFFFFH, 6DEFH, 34A5H, 1CG3H OC21H,OFFFH, 6DEFH, 34A5H 1CG3H, OC21H,OFFFH,OEF7BH 6D6BH, 30A5H, 1CG3H, OC21H OFBFFH, 54A5H, 20A1H, 1421H 0C21H,OFBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, OC22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,OA755H, 69E2H, 3C02H 84422H, 0801H, 58EBH, 4629H 1C6EH,OAA96H, 5509H, 2C21H 6565H, 00CBH,OB481H, 4528H	;TILE(0)-R (0) ; ; ; ; ; ; ; ; ; ; ;	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW ; TILING_DATA DW DW ; DW DW ; DW DW ; DW ; DW ; DW ;	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAAAH SS: 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW TILING_DATA DW DW DW CW DW DW DW DW DW DW DW DW DW D	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7EEFH, 34A5H, 1CG3H, OC21H OFFFFH, 6DEFH, 34A5H, 1CG3H OC21H,OFFFH, 6DEFH, 34A5H 1CG3H, OC21H,OFFFH,OEF7BH 6D6BH, 30A5H, 1CG3H, OC21H OFBFFH, 54A5H, 20A1H, 1421H 0C21H,OFBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, OC22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,OA755H, 69E2H, 3C02H 8442H, 0801H, 58EBH, 4629H 1C6EH,OAA96H, 5509H, 2C21H 6565H, 00CBH,OB481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,OE510H	:TILE(0)-R (0) ; ; ;TILE(1)-G (1) ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ;	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW TILING_DATA DW DW DW CW DW DW DW DW DW CW DW DW DW DW DW DW DW DW DW DW DW DW DW	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAAAH SS: 7EEFH, 34A5H, 1C63H, OC21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,OFFFFH, 6DEFH, 34A5H 1C63H, 0C21H,OFFFFH,OEF7BH 6D6BH, 30A5H, 1C63H, OC21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,OFBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,OA755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,OAA96H, 5509H, 2C21H 6565H, 00C8H,OB481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,OE510H 3118H, 1008H, 0700H, 0300H 0C300H,OC003H, 4001H, 4270H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW TILING_DATA. DW DW DW CW DW DW DW DW DW CW DW DW CW DW DW CW DW DW CW DW DW CW DW CW DW CW DW CW CW CW CW CW CW CW CW CW C	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7EEFH, 34A5H, 1CG3H, OC21H OFFFFH, 6DEFH, 34A5H, 1CG3H OC21H,OFFFH, 6DEFH, 34A5H 1CG3H, OC21H,OFFFH,OEF7BH 6D6BH, 30A5H, 1CG3H, OC21H OFFFH, 54A5H, 20A1H, 1421H 0C21H,OFBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, OC22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,OA755H, 69E2H, 3CO2H 8442H, 0801H, 5BE9H, 4629H 1C6EH,OAA96H, 5509H, 2C21H 6565K, 00CBH,OB481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,OE510H 3118H, 1008H, 0700H, 0300H 0C30DH,OC03CH,4001H, 4270H	:TILE(0)-R (0) ; ; ;TILE(1)-G (1) ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ;	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW ; TILING_DATA DW DW DW ; DW DW ; DW DW ; DW ; DW DW ; DW ; DW ; ; DW ; ; DW ; ; ; ; ; ; ; ; ; ; ; ; ;	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAAAH SS: 7EEFH, 34A5H, 1C63H, OC21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,OFFFFH, 6DEFH, 34A5H 1C63H, 0C21H,OFFFFH,OEF7BH 6D6BH, 30A5H, 1C63H, OC21H 0FBFFH, 54A5H, 20A1H, 1421H 10C21H,OFBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, OC22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,OA755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,OAA96H, 5509H, 2C21H 6565H, 00CBH,OB481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,OE510H 3118H, 1008H, 0700H, 0300H 0C30DH,OC003H, 4001H, 4270H 1718H, 0380H,OC100H, 4001H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;;;
DW ; TILING_DATA. DW DW ; DW DW ; DW DW ; DW ; DW ; DW ;	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7EEFH, 34A5H, 1CG3H, OC21H OFFFFH, 6DEFH, 34A5H, 1CG3H OC21H,OFFFH, 6DEFH, 34A5H 1CG3H, OC21H,OFFFH,OEF7BH 6D6BH, 30A5H, 1CG3H, OC21H OFBFFH, 54A5H, 20A1H, 1421H 0C21H,OFBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, OC22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,OA755H, 69E2H, 3CO2H 84422H, 0801H, 58EBH, 4629H 1C6EH,OAA96H, 5509H, 2C21H 6565K, 00CEH,OB481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,OE510H 3118H, 1008H, 0700H, 0300H 0C30DH,OC003H, 4001H, 4270H 1718H, 0380H,OC100H, 400H 0061H, 84F1H,OC639H, 6310H	:TILE(0)-R (0) ; ; ;TILE(1)-G (1) ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ;	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW ; TILING_DATA. DW DW DW DW DW DW DW DW DW DW	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,0FFFH, 6DEFH, 34A5H 1C63H, 0C21H,0FFFH,0EF7BH 6D6BH, 30A5H, 1C63H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,0FBFFH, 54A5H, 2421H 1021H, 0C21H,0FBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,0A755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,0AA96H, 5509H, 2C21H 6565H, 00C8H,084B1H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,0E510H 3118H, 1008H, 0700H, 0300H 0C30DH,0C003H, 4001H, 4270H 1718H, 0380H,0C100H, 400H 0061H, 84F1H,0C639H, 6310H 112AH, 2240H, 8058H, 811H 2102H, 6084H, 8050H, 911H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;;;
DW TILING_DATA. DW DW DW CW DW DW DW DW CW DW DW CW DW DW CW DW DW CW DW CW DW CW DW CW DW CW CW DW CW CW CW CW CW CW CW CW CW CW CW CW CW	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7EEFH, 34A5H, 1CG3H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1CG3H 0C21H,OFFFFH, 6DEFH, 34A5H 1CG3H, 0C21H,OFFFFH,OEF7BH 6D6BH, 30A5H, 1CG3H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,OFBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,OA755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,OAA96H, 5509H, 2C21H 6565K, 00CBH,OB481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,OE510H 3118H, 1008H, 0700H, 0300H 0C30DH,OC03H, 4001H, 4270H 1718H, 0380H,OC10OH, 400H 0061H, 84F1H,OCC39H, 6310H 112AH, 2240H, 48C8H, 9311H 2222H, 4244H, 8088H, 1A19H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;;;
DW TILING_DATA. DW DW DW DW DW DW DW DW DW DW	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,0FFFH, 6DEFH, 34A5H 1C63H, 0C21H,0FFFH,0EF7BH 6D6BH, 30A5H, 1C63H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,0FBFFH, 54A5H, 2421H 1021H, 0C21H,0FBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,0A755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,0AA96H, 5509H, 2C21H 6565H, 00C8H,08481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,0E510H 3118H, 1008H, 0700H, 0300H 0C30H,0C003H, 4001H, 4270H 1718H, 0380H,0C100H, 400H 0061H, 84F1H,0C639H, 6310H 112AH, 2240H, 48C8H, 9311H 2222H, 4244H, 8058H, 9411H 2202H, 0644H, 8C0DH, 1012H 8220H, 559DH,0A228H, 1095H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;;;
DW TILING_DATA. DW DW DW DW DW DW DW DW DW DW DW DW DW	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,OFFFH, 6DEFH, 34A5H 1C63H, 0C21H,OFFFH, 6DEFH, 34A5H 1C63H, 0C21H,OFFFH,0EF7BH 6D6BH, 30A5H, 1C63H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,OFBFFH, 54A5H, 2421H 1021H, 0C21H,OFBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 5566H, 90B1H, 0802H 0800H,OA755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,OAA96H, 5509H, 2C21H 6565K, 00CBH,OB481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,0E510H 3118H, 1008H, 0700H, 0300H 0C300H,OC003H, 4001H, 4270H 1718H, 0380H,0C100H, 4001H 0061H, 84F1H,OC639H, 6310H 112AH, 2240H, 48C8H, 9311H 2202H, 0644H, 8050H, 911H 2202H, 0644H, 80C0H, 1012H 8220H, 559DH,OA228H, 1095H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4) TILE(7)-G (5)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;;;
DW ; TILING_DATA DW DW DW ; DW DW ; DW DW ; DW ; DW ; DW ; DW ; DW ; DW ; DW ; DW ; DW ; DW ; DW ; ; ; ; ; ; ; ; ; ; ; ; ;	0101H,OAAAAH, 0000H,OA2OAH 0000H,OAOA2H, 0000H,OAAA1H SS: 7EEFH, 34A5H, 1CG3H, OC21H OFFFFH, 6DEFH, 34A5H, 1CG3H OC21H,OFFFH, 6DEFH, 34A5H 1CG3H, OC21H,OFFFH,OEF7BH 6D6BH, 30A5H, 1CG3H, OC21H OFBFFH, 54A5H, 20A1H, 1421H OC21H,OFBFFH, 54A5H, 2421H 1021H, OC21H,OFBFFH, 14A5H 4149H, 8622H, OC22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,OA755H, 69E2H, 3CO2H 0800H,OA755H, 69E2H, 3CO2H 0800H,OA755H, 509H, 4629H 1C6EH,OAA96H, 5509H, 2C21H 6565K, 00CBH,OB481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,OE510H 3118H, 1008H, 0700H, 0300H 1718H, 0380H,OC100H, 4000H 0C61H, 84F1H,OC639H, 6310H 112AH, 2240H, 48C8H, 9311H 2222H, 6084H, 8050H, 9411H 2022H, 0644H, 8C0DH, 1012H 8220H, 559DH,OA228H, 1095H 0000H, 8431H, 2220H,0D79DH	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4) TILE(7)-G (5)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW TILING_DATA. DW DW DW DW DW DW DW DW DW DW DW DW DW	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,0FFFH, 6DEFH, 34A5H 1C63H, 0C21H,0FFFH,0EF7BH 6D6BH, 30A5H, 1C63H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,0FBFFH, 54A5H, 2421H 1021H, 0C21H,0FBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 5566H, 90B1H, 0802H 0800H,0A755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,0AA96H, 5509H, 2C21H 6565K, 00CBH,00A96H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,0E510H 3118H, 1008H, 0700H, 0300H 0C300H,0C003H, 4001H, 4270H 1718H, 0380,0C100H, 4000H 0061H, 84F1H,0C639H, 6310H 112AH, 2240H, 48C8H, 9311H 2202H, 0644H, 80C0H, 1012H 8220H, 559DH,0A228H, 1095H 0000H, 8431H, 2220H,079DH 2028H, 1091H, 0000H, 4210H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4) TILE(7)-G (5)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW TILING_DATA. DW DW DW DW DW DW DW DW DW CW DW DW CW DW CW CW DW CW CW CW CW CW CW CW CW CW C	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,0FFFH, 6DEFH, 34A5H 1C63H, 0C21H,0FFFH,0EF7BH 6D6BH, 30A5H, 1C63H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,0FBFFH, 54A5H, 2421H 1021H, 0C21H,0FBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,0A755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,0AA96H, 5509H, 2C21H 6565H, 00C8H,08481H, 4528H 20A5H, 74EAH, 2C86H, 1453H 5909H, 2429H, 061CH,0E510H 3118H, 1008H, 0700H, 0300H 0C30H,0C03H, 4001H, 4270H 1718H, 0380H,0C100H, 400H 0061H, 84F1H,0C639H, 6310H 112AH, 2240H, 4805BH, 9311H 2202H, 6084H, 8050H, 9411H 2202H, 0644H, 8C0DH, 1012H 8220H, 559DH,0A228H, 1095H 0000H, 8231H, 2220H, 079DH 0A228H, 1091H, 0000H, 4210H 598CH, 2242H, 08CH, 0842H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4) TILE(7)-G (5) TILE(8)-G (6)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW TILING_DATA. DW DW DW DW DW DW DW DW DW DW	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H,0FFFH, 6DEFH, 34A5H 1C63H, 0C21H,0FFFH,0EF7BH 6D6BH, 30A5H, 1C63H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,0FBFFH, 54A5H, 2421H 1021H, 0C21H,0FBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 5566H, 90B1H, 0802H 0800H,0A755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,0AA96H, 5509H, 2C21H 6565K, 00CBH,00A96H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,0E510H 3118H, 1008H, 0700H, 0300H 0C300H,0C003H, 4001H, 4270H 1718H, 0380,0C100H, 4000H 0061H, 84F1H,0C639H, 6310H 112AH, 2240H, 48C8H, 9311H 2202H, 0644H, 80C0H, 1012H 8220H, 559DH,0A228H, 1095H 0000H, 8431H, 2220H,079DH 2028H, 1091H, 0000H, 4210H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4) TILE(7)-G (5) TILE(8)-G (6)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW ; TILING_DATA. DW DW DW DW DW DW DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW DW ; DW DW DW DW DW DW DW DW DW DW	0101H.0AAAAH. 0000H.0A20AH 0000H.0A0A2H. 0000H.0AAA1H SS: 7EEFH. 34A5H. 1CG3H. 0C21H 0FFFFH. 6DEFH. 34A5H. 1CG3H 0C21H.0FFFH. 6DEFH. 34A5H 1CG3H. 0C21H.0FFFH.0EF7BH 6D6BH. 30A5H. 1CG3H. 0C21H 0FBFFH. 54A5H. 20A1H. 1421H 0C21H.0FBFFH. 54A5H. 2421H 1021H. 0C21H.0FBFFH. 14A5H 4149H. 8622H. 0C22H. 8400H 0CA11H. 556EH. 90B1H. 0802H 8442H. 0801H. 5BEBH. 4629H 1C6EH.0AA96H. 5509H. 2C21H 6565H. 00CBH.0B481H. 4528H 20A5H. 74EAH. 2C86H. 14E3H 5909H. 2429H. 061CH.0E510H 311BH. 1008H. 0700H. 0300H 171BH. 0380H.0C100H. 4200H 171BH. 0380H.0C100H. 4000H 0061H. 84F1H.0C639H. 6310H 112AH. 2240H. 48C8H. 9311H 2222H. 6844H. 800BH. 1012H 8220H. 559DH.0A228H. 1015H 0000H. 8431H. 2220H.0D79DH 2208H. 0000H. 8220H. 579DH 0A228H. 1091H. 0000H. 4210H 598CH. 2242H. 08C0H. 0844H. 084H. 2420H 085ADH. 594GH. 1088H. 2842H 0854DH. 594GH. 1088H. 2842H 085ADH. 594CH. 2042H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4) TILE(7)-G (5) TILE(8)-G (6)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW TILING_DATA. DW DW DW DW DW DW DW DW DW DW	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1CG3H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1CG3H 0C21H,0FFFFH, 6DEFH, 34A5H 1CG3H, 0C21H,0FFFH,0EF7BH 6D6BH, 30A5H, 1CG3H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 1021H, 0C21H,0FBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 556CH, 90B1H, 0802H 0800H,0A755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,0AA96H, 5509H, 2C21H 6565K, 00CH,0B481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,0E510H 3118H, 1008H, 0700H, 0300H 0C30DH,0C003H, 4001H, 4270H 1718H, 0380H,0C100H, 4000H 0061H, 84F1H,0C639H, 6310H 112AH, 2240H, 48C8H, 9311H 2202H, 0644H, 8050H, 911H 2202H, 0644H, 8C0DH, 1012H 8220H, 559DH,0A228H, 1095H 0000K, 8431H,220H,0079DH 2208H, 000H, 8220H, 579DH 0A228H, 1091H, 0000H, 4210H 598CH, 2242H, 08C0H, 0842H 085ADH, 5946H, 1088H, 2842H 0842H,0B5ADH, 294AH 0F914H, 677CH,0E6F6H, 68EFH 0C5ASH,0C88H, 0C86H, 3AADH	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4) TILE(7)-G (5) TILE(8)-G (6) TILE(8)-B (7)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;
DW ; TILING_DATA. DW DW DW DW DW DW DW DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW ; DW DW DW ; DW DW DW ; DW DW DW ; DW DW DW DW DW DW DW DW DW DW	0101H.OAAAAH. 0000H.OA2OAH 0000H.OAOA2H. 0000H.OAAA1H SS: 7EEFH. 34A5H. 1CG3H. 0C21H 0FFFFH. 6DEFH. 34A5H. 1CG3H 0C21H.OFFFH. 6DEFH. 34A5H 1CG3H. 0C21H.OFFFH.OEF7BH 6D6BH. 30A5H. 1CG3H. 0C21H 0FBFFH. 54A5H. 20A1H. 1421H 0C21H.OFBFFH. 54A5H. 2421H 1021H. 0C21H.OFBFFH. 14A5H 4149H. 8622H. 0C22H. 8400H 0CA11H. 556EH. 90B1H. 0802H 8442H. 0801H. 5BEBH. 4629H 1C6EH.OAA96H. 5509H. 2C21H 6565H. 00CBH.OB481H. 4528H 20A5H. 74EAH. 2C86H. 14E3H 5909H. 2429H. 061CH.OE510H 311BH. 1008H. 0700H. 0300H 171BH. 0380H.OC100H. 4270H 171BH. 0380H.OC100H. 4000H 0061H. 84F1H.OC639H. 6310H 112AH. 2240H. 48C8H. 9311H 2222H. 6084H. 8050H. 1012H 8220H. 559DH.OA228H. 1095H 0000H. 8431H. 2220H. 0079DH 2208H. 0000H. 8220H. 579DH 0A228H. 1091H. 0000H. 4210H 598CH. 2242H. 08C0H. 0842H 085ADH. 594GH. 1088H. 2842H 085ADH. 594GH. 1088H. 2842H 0842H.085ADH. 294AH 0842H.085ADH. 294AH	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4) TILE(5)-R (4) TILE(8)-G (6) TILE(8)-B (7)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;;;
DW TILING_DATA. DW DW DW DW DW DW DW DW DW DW	0101H, 0AAAAH, 0000H, 0A20AH 0000H, 0A0A2H, 0000H, 0AAA1H SS: 7EEFH, 34A5H, 1C63H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1C63H 0C21H, 0FFFH, 6DEFH, 34A5H 1C63H, 0C21H, 0FFFH, 0EFFH, 0EF7BH 6D6BH, 30A5H, 1C63H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H, 0FBFFH, 54A5H, 2421H 1021H, 0C21H, 0FBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H, 0A755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH, 0AA96H, 5509H, 2C21H 6565H, 00CBH, 0B481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH, 0E510H 3118H, 1008H, 0700H, 0300H 0C300H, 0C003H, 4001H, 4270H 1718H, 0380H, 0C100H, 4000H 0061H, 84F1H, 0C639H, 6310H 112AH, 2240H, 48C8H, 9311H 2202H, 0644H, 800BH, 1012H 8220H, 559DH, 0A228H, 1095H 0000H, 823H, 2220H, 0079DH 2208H, 0000H, 8220H, 579DH 0A228H, 1091H, 0000H, 4210H 598CH, 2242H, 08C0H, 0842H 085ADH, 5946H, 1088H, 2842H 085ADH, 5946H, 1088H, 2842H 085ADH, 5946H, 1088H, 2842H 085ADH, 594CH, 1088H, 2842H 0842H, 085ADH, 294AH 085ADH, 594CH, 1088H, 2842H 0842H, 085ADH, 294AH 08472H, 0027H, 081EH, 0FE29H 0A98BH, 0F77CH, 081EH, 0FE29H	:TILE(0)-R (0) :TILE(1)-G (1) :TILE(4)-R (2) :TILE(4)-G (3) :TILE(5)-R (4) :TILE(5)-R (4) :TILE(7)-G (5) :TILE(8)-G (6) :TILE(8)-B (7) :TILE(A)-R (8)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;;;
DW TILING_DATA. DW DW DW DW DW DW DW DW DW DW	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H SS: 7EEFH, 34A5H, 1CG3H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1CG3H 0C21H,0FFFH, 6DEFH, 34A5H 1CG3H, 0C21H,0FFFH,0EF7BH 6D6BH, 30A5H, 1CG3H, 0C21H 0FBFFH, 54A5H, 20A1H, 1421H 0C21H,0FBFFH, 54A5H, 2421H 1021H, 0C21H,0FBFFH, 14A5H 4149H, 8622H, 0C22H, 8400H 0CA11H, 556EH, 90B1H, 0802H 0800H,0A755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H 1C6EH,0AA96H, 5509H, 2C21H 6565H, 00CBH,0B481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,0E510H 3118H, 1008H, 0700H, 0300H 0C30DH,0C003H, 4001H, 4270H 1718H, 0380H,0C100H, 400H 0061H, 84F1H,0CG39H, 6310H 112AH, 2240H, 4805AH, 9411H 2202H, 6084H, 8050H, 9411H 2202H, 0644H, 8C0DH, 1012H 8220H, 559DH,0A228H, 1095H 0000H, 8431H, 2220H, 0879DH 0A228H, 1091H, 0000H, 4210H 598CH, 2242H, 08C0H, 0842H 0842H,0B5ADH, 5942H, 2042H 0842H,0B5ADH, 5942H,0B5ADH, 294AH 0F914H, 67CH,0E6F6H, 68EFH 0F5A5H,0F6E3H,0C86EH, 3AADH 0F472H, 002FH, 081EH,0FE29H	TILE(0)-R (0) TILE(1)-G (1) TILE(4)-R (2) TILE(4)-G (3) TILE(5)-R (4) TILE(5)-R (4) TILE(8)-G (6) TILE(8)-B (7)	DW ; DW DW DW ; DW DW DW DW ; ; pEMO_INIT PROG ENDS	08422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H 0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H 4000H,0AAA8H, 0000H, 2A88H 0000H,028AH, 0050H,0AAE2H 0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;

PGROUP PROG	SEGMENT	DMCL PROG BYTE PUBLIC 'PROG'		MOV JMP ;	DS:WORD PTR DY,-749 DMCLO_1	;DY=-749
	; PUBLIC PUBLIC	CS:PGROUP DMCLO_C, DMCLO_B, DMCLI_C, DMC DMCLI_C_WAIT, WAIT, DMCLO_C FR MESSAGE_WAIT, MES_CL_WAIT.FILL	EE, MES_CL_WAIT	DMCL1_C_WAIT: TEST JZ MOV CALL	CS:BYTE PTR FLAG,1 DMCL1_C AX,4 WAIT	;CHECK IF 'WAIT'=1/0
	; EXTRN ;	MESSAGE:NEAR		; DMCL1_C: MOV	DS:WORD PTR PTNCNT,OFFFFH	;PTNCNT=OFFFFH
	EXTRN EXTRN EXTRN EXTRN	FLAG:WORD, PTNCNT:WORD, PMAX:W DY:WORD, Y:WORD, DXX:WORD, X:W STATUS:WORD, COM:WORD, MAGETC: EADIH:WORD, EADIL:WORD, DHH:WO	ORD, MODIO:WORD WORD, dAD1:BYTE	JMP ; dmcl1 b: mov	DMCL1 DS:WORD PTR PTNCNT,0	;PTNCNT=0
DMCL	EXTRN ; PROC	WORK_7:WORD NEAR		DMCL1: MOV MOV	DS:WORD PTR EAD1H,4 DS:WORD PTR EAD1L,0	;EAD1H=4 ;EAD1L=0
;	;			MOV MOV MOV	DS:BYTE PTR dAD1,0 DS:WORD PTR DHH,1119 DS:WORD PTR DV,2250	;dAD1=0 ;DHH=1119 ;DV=2250
; ; [DMCL ; [DMCL	O_C] CLE	AY MEMORY CLEAR > ARS VISIBLE 3 PLANES WITHOUT WO ARS VISIBLE 3 PLANES AND NON-VI:	RKING AREA TO "FFFFH" SIBLE 4 PLANES TO "FFFFH"	MOV MOV MOV	DS:WORD PTR MAGETC,1FFH DS:WORD PTR MOD10,0 DS:WORD PTR COM,8E1EH	;MAGETC=1FFH NON-CLIP ;MOD1=0,MOD0=0 ; <a_recfill_a> ;TL=0,SS=1,WL=1,WR=1</a_recfill_a>
; dmclo_c	; VAIT:			RET		;FAST=1
	TEST JZ TEST JNZ	CS:BYTE PTR FLAG,1 DMCLO C CS:BYTE PTR FLAG,4 DMCLO C END	;CHECK IF 'WAIT'=1/0 ;CHECK IF 'PIXEL'=1/0	DMCLO_1: MOV MOV MOV	DS:WORD PTR Y,749 DS:WORD PTR DXX,1119 DS:WORD PTR X,0	;Y=749 ;DX=1119 ;X=0
DMCLO C	MOV Call ;	AX,4 WAIT		MOV MOV MOV	DS:WORD PTR MAGETC,1FFH DS:WORD PTR MODIO,0 DS:WORD PTR COM,903EH	;MAGETC=1FFH NON-CLIP ;MOD1=0,MOD0=0 ; <r recfill=""> ;TL=0,SS=1,WL=1,WR=1</r>
UMCLU_C	MOV MOV	AX,0 BX,749	;X=0 ;Y=749	RET		;FAST=1
	MOV MOV	CX,1119 DX,-755	;DX=1119 ;DY=-755	; DMCLO_C_FREE:	NCTUARD REP. CEATUR 1	
DMCLO_C	CALL END: RET	DMCLO_C_FREE		TEST JNZ MOV	DS:WORD PTR STATUS,1 DMCLO_C_FREE DS:WORD PTR PTNCNT,OFFFFH	;CHECK IF PPBUSY=1/0 ;PTNCNT=OFFFFH
DMCLO_B	;			MOV	DS:WORD PTR PMAX,4 DS:WORD PTR PLANES,7	;PMAX=4 ;PLANES=7
	TEST JNZ MOV	DS:WORD PTR STATUS,1 DMCLO_B DS:WORD PTR PTNCNT,0	;CHECK IF PPBUSY=1/0 ;PTNCNT=0	MOV MOV MOV	DS:WORD PTR MAGETC,1FFH DS:WORD PTR MOD10,0 DS:WORD PTR X,AX	;MAGETC=1FFH NON-CLIP ;MOD1=0,MOD0=0 ; X=(?)
DMCL0:	MOV MOV	DS:WORD PTR PMAX,4 DS:WORD PTR PLANES,7	;PMAX=4 ;PLANES=7	MOV MOV MOV	DS:WORD PTR Y,BX DS:WORD PTR DXX,CX DS:WORD PTR DXX,CX DS:WORD PTR DY,DX	; Y=(?) ; DX=(?) ;DY=(?)
		-1-			-2-	
	MOV	DS:WORD PTR COM,903EH	; <r_recfill> ;TL=0,SS=1,WL=1,WR=1 ;FAST=1</r_recfill>	END		
	RET ;					
; ; ;	< WAIT ;	>				
WAIT:	TEST JZ MOV	CS:BYTE PTR FLAG,1 WAIT END CX.OFFFFH	;CHECK IF 'WAIT'=1/0			
WAIT_1:	LOOPNZ		;LOOP UNTIL CX=0			
WAIT EN	DEC JNZ D: RET	AX WAIT	;LOOP UNTIL AX=0			
;	;					
MES_CL_	; WAIT FIL MOV JMP	L: CS:WORD PTR WORK_7,10 MES CL WAIT 2				
MES_CL_		CS:WORD PTR WORK_7,45				
MES_CL_	WAIT_2: CALL TEST JNZ	MESSAGE_WAIT CS:WORD PTR FLAG,4 MES_CL_WAIT_1				
MES CL_	CALL RET WAIT_1: CALL RET	DMCL0_C_WAIT				
MESSAGE	; TEST JZ TEST JNZ CALL MOV	CS:BYTE PTR FLAG,1 MESSAGE_WAIT_1 CS:BYTE PTR FLAG,4 MESSAGE_WAIT_1 MESSAGE AX,CS:WORD PTR WORK_7	;CHECK IF "WAIT"=1/0 ;CHECK IF "PIXEL"=1/0 ;MESSAGE(?)			
MESSAGE ;	CALL _WAIT_1: RET ;	WAIT				

; DMCL PROG

ENDP ENDS

	NAME	GLIOMAIN		GCLS_1:			
PGROUP	GROUP	PROG			TEST	DS:WORD PTR STATUS,1	
PROG		BYTE PUBLIC 'PROG'			JNZ	GCLS_1	;CHECK IF PPBUSY=1/0
	ASSUME	CS:PGROUP			MOV	DS:WORD PTR PTNCNT,0	;PTNCNT=0
	;				MOV	DS:WORD PTR PMAX,4	;PMAX=4
	PUBLIC	GLIOMAIN			MOV	DS:WORD PTR PLANES,7	;PLANES=7
	:				MOV	DS:WORD PTR MAGETC, 1FFH	;MAGETC=1FFH NON-CLIP
	EXTRN	STATUS:WORD, PTNCNT:WORD, PM/	AX:WORD, PLANES:WORD		MOV	DS:WORD PTR MOD10,0	;MOD1=0,MOD0=0
	EXTRN	MAGETC:WORD, MODIO:WORD, X:WO			MOV	DS:WORD PTR X,0	; X=0
	EXTRN	DY:WORD, COM:WORD, XE:WORD, Y			MOV	DS:WORD PTR Y, -749	; Y=-749
	EXTRN	XC:WORD, YC:WORD, DHH:WORD, I			MOV	DS:WORD PTR DXX,1119	;DX=1119
	EXTRN	XCLMIN:WORD, YCLMIN:WORD, XCI			MOV	DS:WORD PTR DY,755	;DY=755
	EXTRN	EAD2L:WORD, EAD2H:WORD, dAD2			MOV	DS:WORD PTR COM,903EH	; <r_recfill></r_recfill>
	EXTRN	EADORGL:WORD, EADORGH:WORD, o				bornene inn conforen	;TL=0,SS=1,WL=1,WR=1
	EXTRN	WORK_1:WORD,					;FAST=1
	·	WORK_I.WORD,			JMP	G POP EXE	11851-1
GLIOMAI	N [°]	PROC NEAR					
GETONAT	· ·	I NOC ADAR		LINE:	,		
•	,			5185.	MOV	BX,DS:[BP]	;XS=(?)
2	CRADU	L10/B10 >			MOV	CX,DS:[BP+2]	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
2	S GRATH				XOR	CX,OFFFFH	
:					INC	cx	;YS=(?)
,					MOV	SI,DS:[BP+4]	;XE=(?)
	, PUSH	BX			MOV	DI,DS:[BP+6]	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
	CALL	G_PUSH_EXE			XOR	DI,OFFFFH	
	CALL	INIT_SEG			INC	DI	;YE=(?)
	CMP	AH,5			MOV	AX,DS:[BP+8]	,15-(1)
	JZ	GCLS	;"GCLS"		MOV	DX,ES	
	CMP	AH,7	, 0025		MOV	DS, DX	
	JZ	LINE	;"LINE", "REC", "REC_FILL"		CMP	AH,0	
	CMP	AH,8	, LINE , KEC , KEC_TIEL		JZ	G_LINE_EXE	
	JZ	CRL	;"CRL","ELPS","CRL FILL"		CMP	AH,1	
	12	CKL	;"ELPS_FILL"		JZ		
	CMP	AH,9	3 0010_1100		CMP	G_REC_EXE AH,2	
	JZ	PAINT	;"PAINT"		JZ	G_RECFILL_EXE	
	CMP	AH, ODH	, 18181		JMP	G_POP_EXE	
	JZ	COPY	;"COPY"			0,101,000	
	JMP	G POP EXE	, соп	GLINE			
	Jur	G FOF EXE		U_LINE_	TEST	ACTION DTP CTATUC 1	
CDL					JNZ	DS:WORD PTR STATUS,1	CHECK IF PPBUSY=1/0
CRL:	IMD	CPL 1				G_LINE_EXE	; PL=0
	JMP	CRL_1			MOV	CS:WORD PTR WORK_1,1800H	;IP=0,PXEN=0
DAINT	,				CALL	COLOR CAL	, 1P-0, PAEN-0
PAINT:	IMD	DAINT 1			MOV	COLOR_CAL DS:WORD PTR X,BX	; X=(?)
	JMP	PAINT_1					
00.004	,				MOV	DS:WORD PTR Y,CX	; Y=(?)
COPY:	140	CODY 1			MOV	DS:WORD PTR XE,SI	;XE=(?)
	JMP	COPY_1			MOV	DS:WORD PTR YE,DI	;YE=(?)
CCI C .	,				JMP	GENERAL_CP	
GCLS:	MOV	17 50		C	, 		
	MOV	AX, ES		G_REC_E		CC-UODD DTD UODK 1 40000	
	MOV	DS,AX			MOV	CS:WORD PTR WORK_1,4800H	; <a_rec> PXEN=0</a_rec>
		-1-				-2-	
		1				2	
	JMP	REC_RECFILL_EXE		CRL_2:			
6 85651	;	REC_RECFILL_EXE		CRL_2:	MOV	CS:WORD PTR WORK_1,5000H	; <crl> CF=0, IP=0</crl>
G RECFI	; LL EXE:			CRL_2:			; <crl> CF=0,1P=0 ;PXEN=0</crl>
G RECFI	;	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH	; <a_recfill_c></a_recfill_c>	CRL_2:	TEST	AH,20H	
G RECFI	; LL EXE:		;TL=0,SS=1,WL=1,WR=1	CRL_2:	TEST JZ	AH,20H CRL_CRLFILL_EXE	:PXEN=0
	; LL EXE: MOV	CS:WORD PTR WORK_1,8C3CH			TEST JZ MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH	
	; LL EXE: MOV FILL EXE	CS:WORD PTR WORK_1,8C3CH	;TL=0,SS=1,WL=1,WR=1	CRL_2: CRL_CRL	TEST JZ MOV FILL_E)	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE:	:PXEN=0
	; LL EXE: MOV FILL EXE TEST	CS:WORD PTR WORK_1,8C3CH :: DS:WORD PTR STATUS,1	;TL=0,SS=1,WL=1,WR=1 ;FAST=1		TEST JZ MOV FILL_E) TEST	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1	;PXEN=0 ; <crlfill> TL=0,SS=1</crlfill>
	; MOV FILL EXE TEST JNZ	CS:WORD PTR WORK_1,8C3CH :: DS:WORD PTR STATUS,1 REC_RECFILL_EXE	;TL=0,SS=1,WL=1,WR=1		TEST JZ MOV FILL EX TEST JNZ	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE	:PXEN=0
	; MOV FILL EXE: TEST JNZ CALL	CS:WORD PTR WORK_1,8C3CH :: DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL	;TL=0.SS=1,WL=1,WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0		TEST JZ MOV FILL EX TEST JNZ CALL	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0</crlfill>
	; MOV FILL EXE: TEST JNZ CALL MOV	CS:WORD PTR WORK_1,8C3CH :: DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX	;TL=0.SS=1,WL=1,WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; X=(?)		TEST JZ MOV FILL E) TEST JNZ CALL MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?)</crlfill>
	; MOV FILL EXE: TEST JNZ CALL MOV MOV	CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX	;TL=0,SS=1,WL=1,WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?)		TEST JZ MOV FILL_E) TEST JNZ CALL MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;V=(?)</crlfill>
	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,8X DS:WORD PTR Y,CX DS:WORD PTR X,SI	;TL=0.SS=1,WL=1,WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?)		TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR XC,CX DS:WORD PTR VC,CX DS:WORD PTR DXX,SI	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?)</crlfill>
REC REC	; NOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX	;TL=0,SS=1,WL=1,WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?)		TEST JZ MOV FILL_E) TEST JNZ CALL MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;V=(?)</crlfill>
	; NOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV _CP:	CS:WORD PTR WORK_1,8C3CH :: DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR XS,SI DS:WORD PTR YS,DI	;TL=0.SS=1.WL=1.WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?) ;YS=(?)	CRL_CRL	TEST JZ MOV FILL EX TEST JNZ CALL MOV MOV JMP ;	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR XC,CX DS:WORD PTR VC,CX DS:WORD PTR DXX,SI	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;V=(?)</crlfill>
REC REC	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV CP: MOV	CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,8X DS:WORD PTR Y,CX DS:WORD PTR YS,SI DS:WORD PTR YS,DI DS:WORD PTR PMAX,4	;TL=0,SS=1,WL=1,WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4		TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV JMP ; :	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;V=(?) ;DXX=(?)</crlfill>
REC REC	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR X,SI DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR PTNCAT,0FFFH	;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 : x=(?) ; v=(?) ;xS=(?) ;YS=(?) ;PMAX=4 ;PTNCKT=0FFFFH	CRL_CRL	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV JMP ; ; : MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP]	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;V=(?)</crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV CCP: MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR YS,SI DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR PMCNT,0FFFFH DS:WORD PTR MODIO,1	;TL=0,SS=1,WL=1,WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4	CRL_CRL	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV JMP ; : MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR DXX.SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2]	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;V=(?) ;DXX=(?)</crlfill>
REC REC	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR X,SI DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR PTNCAT,0FFFH	:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : x=(?) : v=(?) :xS=(?) :YS=(?) :PMAX=4 :PTNCKT=0FFFFH	CRL_CRL	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV JMP ; : MOV MOV XOR	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;V=(?) ;DXX=(?)</crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR XS,SI DS:WORD PTR YS,DI DS:WORD PTR PTNCNT,0FFFFH DS:WORD PTR MODIO,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR COM,AX	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCKT=0FFFFH ;MOD1=0.MOD0=1</pre>	CRL_CRL	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV JMP ; ; ; MOV MOV XOR INC	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,0FFFH CX	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;DXX=(?) ;X=(?)</crlfill>
REC REC	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,8X DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR PTNCKT,0FFFH DS:WORD PTR PTNCKT,0FFFH DS:WORD PTR WORL1	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCKT=0FFFFH ;MOD1=0.MOD0=1</pre>	CRL_CRL	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV JMP ; : MOV MOV XOR	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;DXX=(?) ;X=(?) ;X=(?) ;Y=(?)</crlfill></pre>
REC REC	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR XS,SI DS:WORD PTR YS,DI DS:WORD PTR PTNCNT,0FFFFH DS:WORD PTR MODIO,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR COM,AX	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCKT=0FFFFH ;MOD1=0.MOD0=1</pre>	CRL_CRL	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV JMP ; ; : MOV MOV XOR INC MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR DXX.SI GENERAL_CP BX,DS:[BP] CX,0FFFFH CX AX,DS:[BP+4]	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;DXX=(?) ;X=(?) ;X=(?) ;Y=(?)</crlfill>
REC REC	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR XS,SI DS:WORD PTR YS,DI DS:WORD PTR PTNCNT,0FFFFH DS:WORD PTR MODIO,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR COM,AX	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCKT=0FFFFH ;MOD1=0.MOD0=1</pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV S XOR INC MOV XOR INC MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;DXX=(?) ;X=(?) ;X=(?) ;Y=(?)</crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YS,SI DS:WORD PTR YS,DI DS:WORD PTR PTNCNT.0FFFFH DS:WORD PTR PTNCNT.0FFFFH DS:WORD PTR WORK_11 DS:WORD PTR WORK_11 DS:WORD PTR CM,AX G_POP_EXE	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;xS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag></com.flag></pre>	CRL_CRL	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV S XOR INC MOV XOR INC MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES	;PXEN=0 ; <crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;DXX=(?) ;X=(?) ;X=(?) ;Y=(?)</crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV JMP ; MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR MODIO,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP]	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;xS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;xC=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV JMP ; : MOV XOR INC MOV XOR INC MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1 PAINT_2	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;Y=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL ;CHECK IF PPBUSY=1/0</crlfill></pre>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,8X DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR PTNCNT,0FFFH DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR COM,AX G_POP_EXE BX.DS:[BP] CX,DS:[BP]2]	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;xS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag></com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV TEST JNZ CALL MOV MOV JMP ; ; ; MOV XOR INC XOR INC XOR INC XOR INC XOR INC XOR INC XOR INC XOR INC XOR INC XOV XOV XOV XOV XOV XOV XOV XOV XOV XOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR XC,BX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,0FFFFH CX AX,DS:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;DXX=(?) ;X=(?) ;X=(?) ;Y=(?) ;AH>B.COL, AL>D.COL</crlfill></pre>
REC REC	; MOV FILL EXE: MOV TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMOD10,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP] CX,DS:[BP+2] CX,OFFFH	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;xS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;xC=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV JMP ; : MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1 PAINT_2	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;Y=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL ;CHECK IF PPBUSY=1/0</crlfill></pre>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV JMP ; MOV MOV JMP ; INC	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR MODIO.1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP] CX,DS:[BP] CX,OFFFFH CX	<pre>;TL=0.SS=1, WL=1, WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCKT=0FFFFH ;MOD1=0, MOD0=1 ;<com.flag> :XC=(?) ;YC=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV JMP ; : MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1 PAINT_2	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?) :X=(?) :X=(?) :X=(?) :AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1</paint></crlfill>
REC REC	; MOV FILL EXE: MOV TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV SIMP ; MOV MOV XOR INC MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR PTNCAT.OFFFH DS:WORD PTR PTNCAT.OFFFFH DS:WORD PTR MODIO.1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX.DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX SI,DS:[BP+6] AX,DS:[BP+8]	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; X=(?) ;Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 :PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :XC=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV NOV NOV NOV NOV NOV MOV TEST JNZ MOV CALL MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DS:WORD PTR YC,CX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,0FFFH CX AX,DS:[BP+2] CX,0FFFFH CX AX,DS:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR WORK_1,6830H PAINT_COL DS:WORD PTR XCLMIN,0	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :DXX=(?) :X=(?) :X=(?) :AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=0</paint></crlfill>
REC REC	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,8X DS:WORD PTR Y,0X DS:WORD PTR YS,01 DS:WORD PTR YS,01 DS:WORD PTR PTNCNT,0FFFH DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR COM,AX G_POP_EXE BX.DS:[BP] CX,0FFFFH CX S1,DS:[BP+4] D1,DS:[BP+4] DX,ES	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; X=(?) ;Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 :PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :XC=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR CALL	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,0FFFFH CX AX,0S:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR WORK_1,6830H PAINT_COL DS:WORD PTR XCLMIN,0 DS:WORD PTR YCLMIN,-749	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;DXX=(?) ;X=(?) ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;AH>B.COL, SS=1 ;XCLMIN=0 ;VCLMIN=-749</crlfill></pre>
REC REC	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR YNCNT,0FFFH DS:WORD PTR YMOD10,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP+2] CX,0FFFH CX S1,DS:[BP+4] D1,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; X=(?) ;Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 :PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :XC=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV MOV CALL MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR XC.SX GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX.ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,749 DS:WORD PTR XCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,1119	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :X=(?) :X=(?) :X=(?) :AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMIN=749 :XCLMIN=749</paint></crlfill>
REC REC	; MOV FILL EXE: MOV TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV JMP ; MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YOBIO,1 DS:WORD PTR YOBIO,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX.DS:[BP] CX,DS:[BP+2] CX,OFFFH CX SI,DS:[BP+4] DI,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX SI,DI	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; X=(?) ;Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 :PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :XC=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV XOR NOV TEST JNZ MOV CALL MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC.BX DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_COL DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :X=(?) :X=(?) :X=(?) :AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=0 :VCLMIN=-749 :XCLMAX=0</paint></crlfill>
REC REC	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR YNCNT,0FFFH DS:WORD PTR YMOD10,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP+2] CX,0FFFH CX S1,DS:[BP+4] D1,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; X=(?) ;Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 :PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :XC=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV XOR INC MOV MOV CALL MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR XC.SX GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX.ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,749 DS:WORD PTR XCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,1119	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :X=(?) :X=(?) :X=(?) :AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMIN=749 :XCLMIN=749</paint></crlfill>
REC REC	; MOV FILL EXE: MOV TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV JMP ; MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YOBIO,1 DS:WORD PTR YOBIO,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX.DS:[BP] CX,DS:[BP+2] CX,OFFFH CX SI,DS:[BP+4] DI,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX SI,DI	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;xS=(?) ;YS=(?) :PMAx=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :xC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ; ; ;</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV XOR INC MOV MOV XOR INC MOV MOV CALL MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR XC.BX CX.DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX.ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,10119 DS:WORD PTR YCLMAX,10 DS:WORD PTR MAGETC.OFFH DS:WORD PTR MAGETC.OFFH DS:WORD PTR MAGETC.OFFH DS:WORD PTR XCLMAX,10	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?) :X=(?) :X=(?) :AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=-749 :XCLMIN=-749 :XCLMAX=0 :MAGETC=0FFH :X=(?)</paint></crlfill>
REC REC	; MOV FILL EXE: MOV TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV JMP ; MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,8X DS:WORD PTR Y,0X DS:WORD PTR YS,01 DS:WORD PTR YS,01 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR COM,AX G_POP_EXE BX.DS:[BP] CX,0FFFFH CX S1,DS:[BP+4] D1,OS:[BP+8] DX,ES DS,DX S1,D1 CRL2	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK 1F PPBUSY=1/0 ; x=(?) ; y=(?) ;xS=(?) ;YS=(?) :PMAX=4 :PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :XC=(?) ;YC=(?) ;DX=(?) ;DY=(?) ;</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV TEST JNZ CALL MOV MOV JMP ; ; ; MOV XOR INC MOV MOV XOR INC CALL MOV CALL MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP+2] CX,0FFFH CX AX,DS:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR WORK_1,6830H PAINT_COL DS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,749 DS:WORD PTR XCLMAX,1119 DS:WORD PTR XCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR MAGTC,0FFH	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;DXX=(?) ;X=(?) ;AH>B.COL, AL>D.COL ;CHECK IF PPBUSY=1/0 ;CHECK IF P</crlfill></pre>
REC REC	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,8X DS:WORD PTR Y,0X DS:WORD PTR YS,01 DS:WORD PTR YS,01 DS:WORD PTR PTNCNT,0FFFH DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 SI,DS:[BP] CX,0FFFH CX SI,DS:[BP+4] DI,SS:[BP+8] DX,ES DS,DX SI,D1 CRL_2 CS:WORD PTR WORK_1,5C00H AH,20H	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;xS=(?) ;YS=(?) :PMAx=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :xC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ; ; ;</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV XOR INC MOV MOV XOR INC MOV MOV CALL MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR XC.BX CX.DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX.ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,10119 DS:WORD PTR YCLMAX,10 DS:WORD PTR MAGETC.OFFH DS:WORD PTR MAGETC.OFFH DS:WORD PTR MAGETC.OFFH DS:WORD PTR XCLMAX,10	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?) :X=(?) :X=(?) :AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=-749 :XCLMIN=-749 :XCLMAX=0 :MAGETC=0FFH :X=(?)</paint></crlfill>
REC REC	; MOV FILL EXE: MOV TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV JMP ; MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YORA DS:WORD PTR YONA,4 DS:WORD PTR YONA,4 DS:WORD PTR YORA,1 DS:WORD PTR WORK_1 DS:WORD PTR WORK_1 CX,OFFFH CX SI,DS:[BP+2] CX,DS:[BP+4] DI,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX SI,D1 CRL_2 CS:WORD PTR WORK_1,5COOH	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;xS=(?) ;YS=(?) :PMAx=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :xC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ; ; ;</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV S; TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX.DS:[BP+4] DX.ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR WORK_1,6830H PAINT_COL DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,1119 DS:WORD PTR YCLMAX,00 DS:WORD PTR MAGETC.OFFH DS:WORD PTR X,BX DS:WORD PTR X,BX DS:WORD PTR X,BX DS:WORD PTR YCX	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?) :X=(?) :X=(?) :AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=-749 :XCLMIN=-749 :XCLMAX=0 :MAGETC=0FFH :X=(?)</paint></crlfill>
REC_REC General Crl_1:	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YOLD,1 DS:WORD PTR YOLD,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP] CX,DS:[BP+2] CX,OFFFH CX SI,DS:[BP+4] DI,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX SI,D1 CRL_2 CS:WORD PTR WORK_1,5C00H AH.20H ELPS_ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH	<pre>;TL=0.SS=1.WL=1.WR=1 ;FAST=1 :CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;xS=(?) ;YS=(?) :PMAx=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :xC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ; ; ;</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV S; TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX.DS:[BP+4] DX.ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR WORK_1,6830H PAINT_COL DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,1119 DS:WORD PTR YCLMAX,00 DS:WORD PTR MAGETC.OFFH DS:WORD PTR X,BX DS:WORD PTR X,BX DS:WORD PTR X,BX DS:WORD PTR YCX	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?) :X=(?) :X=(?) :AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=-749 :XCLMIN=-749 :XCLMAX=0 :MAGETC=0FFH :X=(?)</paint></crlfill>
REC_REC General Crl_1:	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,8X DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR YNOR,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 S:WORD PTR WORK_1,5C00H AH,2OH ELPS_ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH XE:	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : X=(?) ; v=(?) :XS=(?) :YS=(?) :PMAX=4 :PTNCNT=OFFFFH ;MOD1=0.MOD0=1 :<com.flag> :XC=(?) :DX=(?) :DX=(?) :DY=(?) ; :<elps> CF=0.IP=0 ;PXEN=0</elps></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV S; TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH KE: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR VC.CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR VORK_1.6830H PAINT_COL DS:WORD PTR XCLMIN.0 DS:WORD PTR XCLMIN.749 DS:WORD PTR XCLMAX,1119 DS:WORD PTR XCLMAX,0 DS:WORD PTR XCLMAX,0 DS:WORD PTR XCLMAX,0 DS:WORD PTR X,BX DS:WORD PTR X,BX DS:WORD PTR X,BX DS:WORD PTR X,CX GENERAL_CP	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;V=(?) ;AH>B.COL, AL>D.COL ;CHECK IF PPBUSY=1/0 ;AH>B.COL, AL>D.COL ;CHECK IF PPBUSY=1/0 ;CHAN=0 ;YCLMIN=-749 ;XCLMAX=1119 ;YCLMIN=-749 ;XCLMAX=0 ;YACHTAX=0 ;YACHTAX=0 ;YACHTAX=0 ;YACHTAX=0 ;YACTAX=1 ;YACTA</crlfill></pre>
REC_REC General Crl_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR YORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:EBP+2] CX,DS:[BP+4] D1,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX S1,D1 CRL_2 CS:WORD PTR WORK,1,5COOH AH.20H ELPS_ELPSFILL_EXE CS:WORD PTR WORK,1,5C3CH XE: DS:WORD PTR STATUS,1	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : X=(?) ; v=(?) :XS=(?) :YS=(?) :PMAX=4 :PTNCNT=OFFFFH ;MOD1=0.MOD0=1 :<com.flag> :XC=(?) :DX=(?) :DX=(?) :DY=(?) : : :<elps> CF=0.IP=0 ;PXEN=0 ;<elpsfill> TL=0.SS=1</elpsfill></elps></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV NOV TEST TEST JNZ MOV CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX.ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,0 DS:WORD PTR YCLMAX,101 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR Y,CX GENERAL_CP BX,DS:[BP] BX,2	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;Y=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;<paint> ;PMOD=0, TL=0, SS=1 ;XCLMIN=0 ;YCLMIN=-749 ;YCLMIN=-749 ;YCLMIN=-0 ;YCLMIN=-0 ;YCLMIN=-0 ;YCLMIN=-749 ;YCLMIN=-749 ;YCLMIN=0 ;YCLMIN=-749 ;YCLMIN=0 ;YCLMIN=0 ;YCLMIN=0 ;YCLMIN=0 ;YCLMIN=0 ;YCLMIN=10 ;YCLMIN=0 ;YC</paint></crlfill></pre>
REC_REC General Crl_1:	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR YORL1 DS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR CM,AX G_POP_EXE BX.DS:[BP+2] CX.DS:[BP+2] CX.DS:[BP+4] D1,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX S1,D1 CRL_2 CS:WORD PTR WORK_1,5C00H AH,20H ELPS_ELPSFILL_EXE DS:WORD PTR STATUS,1 ELPS_ELPSFILL_EXE	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : X=(?) ; v=(?) :XS=(?) :YS=(?) :PMAX=4 :PTNCNT=OFFFFH ;MOD1=0.MOD0=1 :<com.flag> :XC=(?) :DX=(?) :DX=(?) :DY=(?) ; :<elps> CF=0.IP=0 ;PXEN=0</elps></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV XOR NOV TEST JNZ MOV CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,EX DS:WORD PTR YC,CX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR WORK_1,6830H PAINT_COL DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:W	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :DXX=(?) :X=(?) :X=(?) :X=(?) :AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :YCLMIN=-749 :YCLMIN=-749 :YCLMIN=0 :YCLMIN=-0 :YCLMIN=-749 :YCLMIN=-749 :YCLMIN=0</paint></crlfill>
REC_REC General Crl_1:	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,6X DS:WORD PTR Y,6X DS:WORD PTR YS,01 DS:WORD PTR YS,01 DS:WORD PTR YS,01 DS:WORD PTR YMOD10.1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX.DS:[BP] CX,DS:[BP+2] CX,DS:[BP+4] D1,DS:[BP+4] D1,DS:[BP+8] DX,ES DS,DX S1,D1 CRL_2 CS:WORD PTR WORK_1,5C3CH AH,2OH ELPS_ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR STATUS,1 ELPS_ELPSFILL_EXE COLOR_CAL	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : X=(?) :Y=(?) :XS=(?) :YS=(?) :PMAX=4 :PTNCNT=0FFFFH :MOD1=0.MOD0=1 :<com.flag> :XC=(?) :VC=(?) :DX=(?) :DX=(?) :DY=(?) : :<elps> CF=0.IP=0 :PXEN=0 :<elpsfill> TL=0.SS=1 :CHECK IF PPBUSY=1/0</elpsfill></elps></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV S; TEST JNZ MOV CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DS:WORD PTR YC,CX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP+2] CX,0FFFH CX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR YCLMIN,0 DS:WORD PTR XCLMAX,1119 DS:WORD PTR XCLMAX,1119 DS:WORD PTR XCLMAX,0 DS:WORD PTR X,BX DS:WORD PTR Y,CX GENERAL_CP	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;V=(?) ;AH>B.COL, AL>D.COL ;CHECK IF PPBUSY=1/0 ;AH>B.COL, AL>D.COL ;CHECK IF PPBUSY=1/0 ;CHAN=0 ;YCLMIN=-749 ;XCLMAX=1119 ;YCLMIN=-749 ;XCLMAX=1119 ;YCLMAX=0 ;Y=(?) ;X=(?)</crlfill></pre>
REC_REC General Crl_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR YORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR YORK,1 DS:UORD PTR YORK,1 DS:EBP+2] CX,DS:[EP+2] CX,DS:[EP+4] D1,DS:[EP+6] AX,DS:[EP+6] AX,DS:[EP+8] DX,ES DS,DX S1,D1 CRL,2 CS:WORD PTR WORK,1,5C00H AH,2OH ELPS_ELPSFILL_EXE CS:WORD PTR YORK,1,5C3CH XE: DS:WORD PTR STATUS,1 ELPS_ELPSFILL_EXE COLOR_CAL DS:WORD PTR XC,BX	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : X=(?) ; v=(?) :XS=(?) :YS=(?) :PMAX=4 :PTNCNT=OFFFFH ;MOD1=0.MOD0=1 :<com.flag> :XC=(?) :DX=(?) :DX=(?) :DY=(?) : : :<elps> CF=0.1P=0 :PXEN=0 :<elpsf1ll> TL=0.SS=1 :CHECK IF PPBUSY=1/0 :XC=(?)</elpsf1ll></elps></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV XOR INC MOV MOV XOR INC MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH.20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR XC.BX DS:WORD PTR XC.SI GENERAL_CP BX.DS:[BP] CX.DS:[BP+2] CX.OFFFFH CX AX.DS:[BP+4] DX.ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR XCLMIN.0 DS:WORD PTR XCLMIN.0 DS:WORD PTR XCLMIN.0 DS:WORD PTR XCLMIN.0 DS:WORD PTR YCLMIN.749 DS:WORD PTR YCLMAX.1019 DS:WORD PTR YCLMAX.1119 DS:WORD PTR YCLMAX.1119 DS:WORD PTR YCLMAX.0 DS:WORD PTR Y.CX GENERAL_CP BX.DS:[BP] BX.2 CX.0S:[BP+2] CX.3 CX.0FFFFH	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;Y=(?) ;BODY>CHARACTER FACE ;BODY>CHARACTER FACE</crlfill></pre>
REC_REC General Crl_1:	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YS,SI DS:WORD PTR YS,SI DS:WORD PTR YODIO,1 AX.CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX.DS:[BP+2] CX.DS:[BP+4] D1,DS:[BP+6] AX.DS:[BP+8] DX.ES DS,DX S1,D1 CRL_2 CS:WORD PTR WORK_1,5C00H AH,20H ELPS ELPSFILL_EXE CS:WORD PTR YORK_1,5C3CH XE: DS:WORD PTR STATUS,1 ELPS_ELPSFILL_EXE COLOR_CAL DS:WORD PTR YORK_2,2C3CH DS:WORD PTR YORK_2,2C3CH DS:WORD PTR YORK_2,2C3CH DS:WORD PTR YORK_2,2C3CH DS:WORD PTR YORK_2,2C3CH DS:WORD PTR YORK_2,2C3CH XE: DS:WORD PTR YC,2X	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : X=(?) :Y=(?) :XS=(?) :YS=(?) :PMAX=4 :PTNCNT=0FFFFH :MOD1=0.MOD0=1 :<com.flag> :XC=(?) :VC=(?) :DX=(?) :DX=(?) :DY=(?) : :<elps> CF=0.IP=0 :PXEN=0 :<elpsfill> TL=0.SS=1 :CHECK IF PPBUSY=1/0</elpsfill></elps></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV XOR INC MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX AX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR WORK_1,6830H PAINT_COL DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCL	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;<paint> ;PMOD=0, TL=0, SS=1 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=0</paint></crlfill></pre>
REC_REC General Crl_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,8X DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR YNORT,0FFFH DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 DS:WORD PTR WORK,1 SI,DS:[BP] CX,DS:[BP+4] DI,DS:[BP+4] DI,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX SI,DI CRL_2 CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR WORK,1,5C3CH XE: DS:WORD PTR YORK,1 ELPS_ELPSFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DHDU_CAL	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : X=(?) :Y=(?) :XS=(?) :YS=(?) :PMAX=4 :PTNCNT=0FFFFH :M0D1=0.M0D0=1 :<com.flag> :XC=(?) :VC=(?) :DX=(?) :DX=(?) :DX=(?) : : :<elps> CF=0.IP=0 :PXEN=0 :<elpsfill> TL=0.SS=1 :CHECK IF PPBUSY=1/0 :XC=(?) ;YC=(?)</elpsfill></elps></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV S: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DS:WORD PTR YC,CX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP+2] CX,0FFFH CX DS:UORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR XCLMIN,0 DS:WORD PTR YCLMIN,-749 DS:WORD PTR YCLMIN,-749 DS:WORD PTR YCLMIN,-749 DS:WORD PTR XCLMAX,1119 DS:WORD PTR XCLMAX,1119 DS:WORD PTR XCLMAX,1119 DS:WORD PTR XCLMAX,0 DS:WORD PTR XCLMAX,0 DS:WORD PTR X,BX DS:WORD PTR Y,CX GENERAL_CP BX,DS:[BP] BX,2 CX,0FFFFH CX CX,0FFFFH CX CX,0FFFFH CX,3 CX,0FFFFH	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;Y=(?) ;BODY>CHARACTER FACE ;BODY>CHARACTER FACE</crlfill></pre>
REC_REC General Crl_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR YORL1 DS:WORD PTR WORL1 DS:WORD PTR WORL1 DS:WORD PTR WORL1 DS:WORD PTR WORL1 DS:WORD PTR WORL1 DS:WORD PTR YORL1 DS:WORD PTR YORL1 DS:WORD PTR YORL1 DS:WORD PTR YORL1 DS:WORD PTR WORL1 DS:WORD PTR WORL1 CX,DS:[BP+2] CX,OFFFH CX SI,DS:[BP+6] AX,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX SI,DI CRL2 CS:WORD PTR WORK_1,5COOH AH.20H ELPS_ELPSFILL_EXE COLOR_CAL DS:WORD PTR YORK_1,5C3CH XE: DS:WORD PTR YC,CX DHUCCAL DS:WORD PTR YC,CX DHUCCAL DS:WORD PTR DHH,BX	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : X=(?) ; v=(?) :xS=(?) :YS=(?) :PMAX=4 :PTNCNT=OFFFFH ;MOD1=0.MOD0=1 :<com.flag> :XC=(?) :DX=(?) :DX=(?) :DX=(?) :DY=(?) : :<elpsf (ll=""> TL=0.SS=1 :CHECK IF PPBUSY=1/0 :XC=(?) :YC=(?) :DHH=(?)</elpsf></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV S; TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH.20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR XC.BX CX,DS:[BP+2] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX:WORD PTR XCLMIN.0 DS:WORD PTR XCLMAX.1119 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCLMAX.0 DS:WORD PTR Y.CX GENERAL_CP BX,DS:[BP] BX,2 CX,0S:[BP+2] CX,3 CX,0FFFFH CX AX,DS:[BP+4] DI,DS:(BP+4] DI,DS:(BP+4] DI,DS:[BD] DS:[BP+4] DI,DS:[BP+4] DI,DS:[BP+4] DD] DS:[BP+4] DI	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;CHECK IF PPBUSY=1/</crlfill></pre>
REC_REC General Crl_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YS,SI DS:WORD PTR YS,SI DS:WORD PTR YOD10,1 AX.CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR CM,AX G_POP_EXE BX.DS:[BP+2] CX.DS:[BP+2] CX.DS:[BP+4] D1,DS:[BP+6] AX.DS:[BP+8] DX.ES DS,DX S1,D1 CRL_2 CS:WORD PTR WORK_1,5C00H AH,20H ELPS ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR STATUS,1 ELPS_ELPSFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DHU,AX	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : x=(?) ; v=(?) :xS=(?) ;YS=(?) :PMAX=4 :PTNCNT=OFFFFH ;MOD1=0.MOD0=1 :<com.flag> :xC=(?) :VC=(?) :DX=(?) :DX=(?) :DY=(?) :VC=(?) :CHECK IF PPBUSY=1/0 :xC=(?) :CHECK IF PPBUSY=1/0 :xC=(?) :VC=(?) :DHH=(?) :DU=(?)</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV NOV TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR STATUS,1 PAINT_2 CX.DS:[BP+4] DX.ES DS.DX DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WOR	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;CHECK IF PPBUSY=1/</crlfill></pre>
REC_REC General Crl_1:	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,6X DS:WORD PTR Y,6X DS:WORD PTR YS,01 DS:WORD PTR YS,01 DS:WORD PTR YS,01 DS:WORD PTR YNOR1,0FFFH DS:WORD PTR MODIO.1 AX,CS:WORD PTR WORK,1 DS:WORD PTR MODIO.1 AX,CS:WORD PTR WORK,1 DS:WORD PTR COM,AX G_POP_EXE BX.DS:[BP] CX,DS:[BP+2] CX,OFFFH CX SI,DS:[BP+4] DI,DS:[BP+4] DI,DS:[BP+8] DX,ES DS,DX SI,D1 CRL_2 CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR WORK,1,5C3CH XE: DS:WORD PTR WORK,1,5C3CH XE: DS:WORD PTR YC,CX DHDV_CAL DS:WORD PTR DW,AX DS:WORD PTR DW,AX DS:WORD PTR DW,AX DS:WORD PTR DW,AX DS:WORD PTR DW,AX DS:WORD PTR DW,AX DS:WORD PTR DW,AX	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : X=(?) ; v=(?) :xS=(?) :YS=(?) :PMAX=4 :PTNCNT=OFFFFH ;MOD1=0.MOD0=1 :<com.flag> :XC=(?) :DX=(?) :DX=(?) :DX=(?) :DY=(?) : :<elpsf (ll=""> TL=0.SS=1 :CHECK IF PPBUSY=1/0 :XC=(?) :YC=(?) :DHH=(?)</elpsf></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV S; TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH.20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR XC.BX CX,DS:[BP+2] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX:WORD PTR XCLMIN.0 DS:WORD PTR XCLMAX.1119 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCLMAX.0 DS:WORD PTR Y.CX GENERAL_CP BX,DS:[BP] BX,2 CX,0S:[BP+2] CX,3 CX,0FFFFH CX AX,DS:[BP+4] DI,DS:(BP+4] DI,DS:(BP+4] DI,DS:[BD] DS:[BP+4] DI,DS:[BP+4] DI,DS:[BP+4] DD] DS:[BP+4] DI	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;<paint> ;PMOD=0, TL=0, SS=1 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=0</paint></crlfill></pre>
REC_REC General Crl_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR CAL DS:WORD PTR X,BX DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YS,SI DS:WORD PTR YS,SI DS:WORD PTR YOD10,1 AX.CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR CM,AX G_POP_EXE BX.DS:[BP+2] CX.DS:[BP+2] CX.DS:[BP+4] D1,DS:[BP+6] AX.DS:[BP+8] DX.ES DS,DX S1,D1 CRL_2 CS:WORD PTR WORK_1,5C00H AH,20H ELPS ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR STATUS,1 ELPS_ELPSFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DHU,AX BS:WORD PTR DH,BX DS:WORD PTR DH,AX	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : x=(?) ; v=(?) :xS=(?) ;YS=(?) :PMAX=4 :PTNCNT=OFFFFH ;MOD1=0.MOD0=1 :<com.flag> :xC=(?) :VC=(?) :DX=(?) :DX=(?) :DY=(?) :VC=(?) :CHECK IF PPBUSY=1/0 :xC=(?) :CHECK IF PPBUSY=1/0 :xC=(?) :VC=(?) :DHH=(?) :DU=(?)</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV NOV TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR STATUS,1 PAINT_2 CX.DS:[BP+4] DX.ES DS.DX DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WOR	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;<paint> ;PMOD=0, TL=0, SS=1 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=0</paint></crlfill></pre>
REC_REC General Crl_1:	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,6X DS:WORD PTR Y,6X DS:WORD PTR YS,01 DS:WORD PTR YS,01 DS:WORD PTR YS,01 DS:WORD PTR YNOR1,0FFFH DS:WORD PTR MODIO.1 AX,CS:WORD PTR WORK,1 DS:WORD PTR MODIO.1 AX,CS:WORD PTR WORK,1 DS:WORD PTR COM,AX G_POP_EXE BX.DS:[BP] CX,DS:[BP+2] CX,OFFFH CX SI,DS:[BP+4] DI,DS:[BP+4] DI,DS:[BP+8] DX,ES DS,DX SI,D1 CRL_2 CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR WORK,1,5C3CH XE: DS:WORD PTR WORK,1,5C3CH XE: DS:WORD PTR YC,CX DHDV_CAL DS:WORD PTR DW,AX DS:WORD PTR DW,AX DS:WORD PTR DW,AX DS:WORD PTR DW,AX DS:WORD PTR DW,AX DS:WORD PTR DW,AX DS:WORD PTR DW,AX	<pre>:TL=0.SS=1.WL=1.WR=1 :FAST=1 :CHECK IF PPBUSY=1/0 : x=(?) ; v=(?) :xS=(?) ;YS=(?) :PMAX=4 :PTNCNT=OFFFFH ;MOD1=0.MOD0=1 :<com.flag> :xC=(?) :VC=(?) :DX=(?) :DX=(?) :DY=(?) :VC=(?) :CHECK IF PPBUSY=1/0 :xC=(?) :CHECK IF PPBUSY=1/0 :xC=(?) :VC=(?) :DHH=(?) :DU=(?)</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL E) TEST JNZ CALL MOV MOV NOV TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH (E: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR YC,EX DS:WORD PTR STATUS,1 PAINT_2 CX.DS:[BP+4] DX.ES DS.DX DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WOR	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;X=(?) ;X=(?) ;X=(?) ;AH>B.COL, AL>D.COL :CHECK IF PPBUSY=1/0 ;<paint> ;PMOD=0, TL=0, SS=1 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YCLMIN=0</paint></crlfill></pre>

	SHL SHL SHL RCL SHL RCL SHL	AL,1 AX,1 AX,1 DX,1 AX,1 DX,1 AX,1 AX,1		MOV MOV MOV POP RET ;	ES:WORD PTR EADORGH.O ES:BYTE PTR dADORG,O BP,100H AX	;EADORGH=O ;dADORG=O
	RCL	DX,1		COLOR_CAL: AND	AX,7	
	SHL	AX,1		MOV	BP,AX	
	RCL	DX,1		MOV	AL, CS: BYTE PTR [COL+BP]	
	;			MOV	DS:WORD PTR PLANES, AX	;PLANES=(?)
COPY_2	TEST	DS:WORD PTR STATUS,1		RET		
	JNZ	COPY_2	;CHECK IF PPBUSY=1/0	; COL:		
	MOV	DS:WORD PTR EAD2L,AX	;EAD2L=(?)	CUL. DB	0, 4, 1, 5, 2, 6, 3, 7	;FIT COLOR BITS
	MOV	DS:WORD PTR EAD2H, DX	;EAD2H=(?)	:	0, 4, 1, 3, 2, 0, 3, 1	,111 00000 0115
	MOV	DS:BYTE PTR dAD2,0	;dAD2=0	PAINT_COL:		
	MOV	AX,DI		MOV	DX,AX	
	CALL MOV	COLOR_CAL DS:WORD PTR PITCHS,2	;PITCHS=2	CALL		
	MOV	DS:WORD PTR DHH,23	; DH=23	MOV	AX, DX	
	MOV	DS:WORD PTR DV,23	; DV=23	MOV	AL,AH AX,7	
	MOV	DS:WORD PTR X, BX	;X=(?)	MOV	BP, AX	
	MOV	DS:WORD PTR Y,CX	;Y=(?)	MOV	AL, CS: BYTE PTR [COL+BP]	
	MOV	DS:WORD PTR PMAX,4	;PMAX=4	MOV	DS:WORD PTR DXX,AX	;DXX=(?)
	MOV	DS:WORD PTR MOD10,2 DS:WORD PTR COM,8008H	;MOD1=0,MODO=2 ; <a_copy_ac> COPY</a_copy_ac>	RET		
	1101	05. WORD FIR CON, 80081	;ESE=0, REV=0, ROT=0, {MD}	; DHDV_CAL:		
			;FAST=0 < !BUG!! >	MOV_CAL:	AX,SI	
	JMP	G_POP_EXE		MUL	AX	;DH=DX * *2
120	;			MOV	BX,AX	
;				MOV	CX,DX	
INIT_S	FC.			;		
1011_3	PUSH	AX		MOV MUL	AX,DI AX	;DV=DY * *2
	MOV	AL,9		DHDV CAL 1:	83	, 0 - 01++2
	OUT	OD1H,AL	;ENABLE /CSIR/	OR	DX,DX	
	IN	AL, OD1H		JZ	EXIT_ROT	
	TEST MOV	AL,1		SHR	DX,1	
	JZ	AX,0D000H M0D_9801	CHECK IF 98XA/9801	RCR	AX,1	
	MOV	AX,8000H	, CHECK IF SOXA79801	SHR	CX,1	
MOD_98				RCR JMP	BX,1 DHDV_CAL_1	
	MOV	ES,AX		:		
	MOV	AL,OFOH		EXIT_ROT_1:		
	0UT	91H, AL		SHR	AX,1	
INIT_S	OUT FC 1:	93H,AL		SHR	CX,1	
	TEST	ES:WORD PTR STATUS,1		RCR	BX,1	
	JNZ	INIT_SEG_1	;CHECK IF PPBUSY=1/0	EXIT_ROT: OR	cx,cx	
	MOV	ES:WORD PTR EADORGL,0	;EADORGL=0	JNZ	EXIT_ROT_1	

-5-

; MOY DX.OFCOOH DHDV_CAL_3: TEST AX.DX JZ DHDV_CAL_2 SHR BX.1 JMP DHDV_CAL_3 ; DHDV_CAL_4: SHR BX.1 DHDV_CAL_4: TEST BX.DX JNZ DHDV_CAL_4 RET ; C POP EXE: POP ES POP DS POP DS POP DS POP DS POP D1 POP SS POP DS POP D1 POP SS POP D1 POP SS POP D1 POP SS POP D2 POP BX MOV AX.0 IRET ; G PUSH EXE: POP BX MOV AX.0 IRET ; G PUSH EXE: PUSH AX PUSH DS PUSH DX PUSH DS PUSH DS PUSH BP PUSH S1 PUSH DS PUSH BR PUSH CX PUSH DS PUSH BR PUSH DS PUSH BR PUSH DS PUSH DS PUSH BR PUSH DS PUSH BR PUSH S1 PUSH DS PUSH BR PUSH S1 PUSH DS PUSH BR PUSH BR PUSH DS PUSH BR PUSH DS PUSH BR PUSH CS PUSH BR PUSH DS PUSH BR PUSH DS PUSH BR PUSH BR PUSH BR PUSH BR PUSH BR PUSH DS PUSH BR PUSH DS PUSH BR P

END

-6-

-8-

-7-

;NORMAL END

NAME LINE PGROUP GROUP PROG SEGMENT BYTE PUBLIC 'PROG PROG ASSUME CS:PGROUP PUBLIC LINE DEMO, LINE DEMO H DMCLO_C_WAIT:NEAR, MES_CL_WAIT:NEAR EXTRN WORK_1:WORD, WORK_2:WORD, WORK_3:WORD, WORK_4:WORD WORK_5:WORD, WORK_6:WORD, STATUS:WORD, PMAX:WORD PTNCNT:WORD, X:WORD, Y:WORD, MODIO:WORD, PLANES:WORD EXTRN EXTRN EXTRN XE:WORD, YE:WORD, COM:WORD, PTNCNT:WORD, FLAG:BYTE EXTRN LINE PROC NEAR < LINE DEMO > LINE DEMO: MOV \$1,10 :MESSAGE(10) CALL MES_CL_WAIT MOV CS:WORD PTR WORK_1,64 CS:WORD PTR WORK_2,48 CS:WORD PTR WORK_3,10 ;REPETITION COUNTS (H) ;REPETITION COUNTS (V) MOV ;DISTANCE (X) MOV CS:WORD PTR WORK_4,10 CS:WORD PTR WORK_5,319 CS:WORD PTR WORK_6,239 MOV ;DISTANCE (Y) ;X=319 MOV :Y=239 MOV CALL LINE_EXE_ALL SI,11 MOV MES_CL_WAIT CALL :MESSAGE(11) CALL LINE DEMO_H MOV \$1,12 :MESSAGE(12) CALL MES_CL_WAIT RET LINE DEMO H: REPETITION COUNTS (H); REPETITION COUNTS (V) CS:WORD PTR WORK_1,56 MOV CS:WORD PTR WORK_2,50 CS:WORD PTR WORK_3,20 CS:WORD PTR WORK_4,15 MOV :DISTANCE (X) MOV ;DISTANCE (Y) MOV CS:WORD PTR WORK_5,559 CS:WORD PTR WORK 6,374 MOV :X=559 ;Y=374 MOV LINE_EXE_ALL: DS:WORD PTR STATUS.1 TEST LINE_EXE_ALL DS:WORD PTR PMAX,4 DS:WORD PTR PTNCNT,0FFFFH ;CHECK IF PPBUSY=1/0 JNZ MOV :PMAX=4 ;PTNCNT=OFFFFH MOV -1-LINE EXE_1: ;PTNCNT=(?) DS:WORD PTR PTNCNT,SI MOV MOV DS:WORD PTR MODIO,44H DS:WORD PTR COM,1814H ;MOD1=4,MOD0=4 ;<A_LINE_M1> PL=0 MOV IP=0, PXEN=1, BPPX=1 ADD SI.1111H :PTNCNT+1111H RET : LINE ENDP PROG ENDS END

AX,CS:WORD PTR WORK_5 MOV DS:WORD PTR X,AX AX,CS:WORD PTR WORK_6 MOV :X=(?) MOV MOV DS:WORD PTR Y,AX :Y=(?) ;MOD1=1,MOD0=0 DS:WORD PTR MODIO.10H MOV ;XE=0 BX.0 MOV DX,0 : VF=0 MOV MOV SI,1 CS:BYTE PTR FLAG,4 :PLANES=1 TEST CHECK IF PIXEL=1/0 JNZ LINE_EXE_ALL_1 ;PTNCNT=1111H MOV SI.1111H LINE_EXE_ALL_1: :REPETITION COUNTS (H) MOV CX.CS:WORD PTR WORK 1 LINE_LOW: CALL LINE_EXE BX,CS:WORD PTR WORK_3 LINE_LOW :XE+(?) --> XE ADD LOOP DX,CS:WORD PTR WORK_4 CX,CS:WORD PTR WORK_2 :YE=(?) ADD REPETITION COUNTS (V) MOV LINE_RIGHT: CALL LINE_EXE DX,CS:WORD PTR WORK_4 ;YE+(?) --> YE ADD 1.00P LINE_RIGHT BX,CS:WORD PTR WORK_3 CX,CS:WORD PTR WORK_1 :XE-(?) --> XE SUB REPETITION COUNTS (H) MOV DEC CX LINE_HIGH: CALL LINE EXE BX,CS:WORD PTR WORK_3 LINE_HIGH ;XE-(?) --> XE SUB LOOP SUB DX,CS:WORD PTR WORK_4 :YE-(?) --> YE CX,CS:WORD PTR WORK 2 REPETITION COUNTS (V) MOV LINE_LEFT: CALL LINE EXE DX.CS:WORD PTR WORK 4 ;YE-(?) --> YE SUR LINE_LEFT LOOP RET LINE_EXE: DS:WORD PTR STATUS.1 TEST ;CHECK IF PPBUSY=1/0 LINE_EXE JNZ DS:WORD PTR XE, BX DS:WORD PTR YE, DX CS:BYTE PTR FLAG, 4 ;XE=(?) ;YE=(?) MOV MOV TEST LINE_EXE_1 DS:WORD PTR PLANES,SI JNZ CHECK IF PIXEL=1/0 ;PLANES=(?) MOV DS:WORD PTR COM, 1800H ;<A_LINE_M1> PL=0 ;IP=0,PXEN=0 MOV ;PLANES+1 INC SI RET -2-NAME RECT PGROUP GROUP PROG SEGMENT BYTE PUBLIC 'PROG' PROG ASSUME CS:PGROUP PUBLIC RECT_DEMO EXTRN MES_CL_WAIT:NEAR STATUS:WORD, PMAX:WORD, PTNCNT:WORD, X:WORD, Y:WORD EXTRN MODIO:WORD, PLANES:WORD, XS:WORD, YS:WORD, COM:WORD EXTRN RECT PROC NEAR < RECTANGLE DEMO > RECT_DEMO: :MESSAGE(28) MOV \$1,28 CALL MES_CL_WAIT CALL. RECT_DEMO_L ;MESSAGE(29) MOV \$1,29 CALL MES CL WAIT CALL RECT DEMO H MOV \$1,30 :MESSAGE(30) MES_CL_WAIT CALL RET RECT_DEMO_L: S1,639 ;XS=639 MOV MOV DI,479 : YS=479 REPETITION COUNTS MOV CX.60 RECT_EXE_ALL JMP RECT_DEMO_H: \$1,1119 :XS=1119 MOV MOV D1,749 :YS=749 CX,93 REPETITION COUNTS MOV RECT_EXE_ALL: :X=0 MOV AX.0 ;Y=0 MOV BX,0 RECT_EXE_ALL_1: TEST DS:WORD PTR STATUS,1 JNZ RECT_EXE_ALL_1 :CHECK IF PPBUSY=1/0 MOV DS:WORD PTR PMAX,4 :PMAX=4 DS:WORD PTR PTNCNT, OFFFFH ;PTNCNT=OFFFFH MOV DS:WORD PTR X,AX DS:WORD PTR Y,BX MOV :X=(?) ;Y=(?) MOV DS:WORD PTR MODIO, 10H ;MOD1=1,MOD0=0 MOV

MOV MOV MOV ADD ADD SUB SUB LOOP RET ; ; RECT ENDP PROG ENDS END	DS:WORD PTR PLANES,CX DS:WORD PTR XS,SI DS:WORD PTR YS,DI DS:WORD PTR COM,4800H AX,5 BX,4 SI,5 DI,4 RECT_EXE_ALL_1	:PLANES=(?) :XS=(?) :VS=(?) : <a.rec> :PXEN=0 :X+5> X :Y+4> Y :XS-5> XS :YS-4> YS</a.rec>	PGROUP PROG CRL_ELP ; ; ;	ASSUME ; PUBLIC ; EXTRN ; EXTRN EXTRN ; S ;	CRL_ELPS PROG BYTE PUBLIC 'PROG' CS:PGROUP CRL_DEMO, ELPS_DEMO MES_CL_WAIT:NEAR STATUS:WORD, PMAX:WORD, PTNC MODIO:WORD, PLANES:WORD, DXX WORK_1:WORD, WORK_2:WORD, DH PROC NEAR E DEMO >	(:WORD, COM:WORD
			, Crl_dem Crl_dem	MOV CALL CALL MOV CALL CALL MOV CALL RET ;	S1,36 MES_CL_WAIT CRL_DEMO_L S1,37 MES_CL_WAIT CRL_DEMO_H S1,38 MES_CL_WAIT S1,319 D1,229 BX,238 CRL_EXE_ALL	:MESSAGE(36) :MESSAGE(37) :MESSAGE(38) :XC=319 :YC=239 :DXX=238
			CRL_DEM CRL_EXE CRL_EXE	MOV MOV MOV _ALL: MOV	S1,559 D1,374 BX,373 AX,1 DS:WORD PTR STATUS,1 CRL_EXE_ALL_1 DS:WORD PTR PTNCNT.OFFFFH DS:WORD PTR PTNCNT.OFFFFH DS:WORD PTR YC.DI DS:WORD PTR YC.DI DS:WORD PTR MODIO,10H	:XC=559 ;VC=374 ;DXX=373 :CHECK IF PPBUSY=1/0 :PMAX=4 ;PTNCNT=0FFFFH :X=(?) ;V=(?) ;M0D1=1,M0D0=0
MOV MOV INC SUB JNC RET ; < ELLI	-2- DS:WORD PTR PLANES,AX DS:WORD PTR DXX,BX DS:WORD PTR COM,5000H AX BX,5 CRL EXE_ALL_1 PSE DEMO >	;PLANES=(?) ;DXX=(?) ; <crl> CF=0, P=0 ;PXEN=0 ;PLANES+1 ;DXX-5</crl>		TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV	-1- DS:WORD PTR STATUS,1 ELPS_EXE_ALL_1 DS:WORD PTR XC,S1 DS:WORD PTR YC,D1 AX,CS:WORD PTR VORK_1 DS:WORD PTR DHH,AX AX,CS:WORD PTR WORK_2 DS:WORD PTR WORK_2 DS:WORD PTR WORA DS:WORD PTR MODIO,10H DS:WORD PTR PLANES,CX DS:WORD PTR PLANES,CX DS:WORD PTR OV,BX	:CHECK IF PPBUSY=1/0 ;XC=(?) ;YC=(?) :DHH=(?) :DV=(?) :MOD1=1,MODD=0 ;PLANES=(?) :DY=(?) :SEC=CEC_10=0
ELPS_DEMO: MOV CALL MOV CALL MOV CALL MOV MOV CALL RET	SI.39 MES_CL_WAIT CS:WORD PTR WORK_1,16 CS:WORD PTR WORK_2,9 ELPS_DEMO_1 CS:WORD PTR WORK_1,9 CS:WORD PTR WORK_2,16 ELPS_DEMO_1	;MESSAGE(30) ;DH=16 ;DV=9 ;DH=9 ;DV=16	: CRL_ELP: PROG	MOV SUB JNC RET ; S ENDS END	DS:WORD PTR COM,5COOH CX BX.5 ELPS_EXE_ALL_1 ENDP	: <elps> CF=0, IP=0 ;PXEN=0 ;PLANES+1 ;DXX-5</elps>
ELPS_DEMO_1: CALL MOV CALL CALL MOV CALL RET ;	ELPS_DEMO_L SI,40 MES_CL_VAIT ELPS_DEMO_H SI,41 MES_CL_VAIT	;MESSAGE(40) ;MESSAGE(41)				
ELPS_DEMO_L: MOV MOV JMP ; ELPS DEMO H:	SI,319 DI,239 BX,238 ELPS_EXE_ALL	;XC=319 ;YC=239 ;DY=238				
ELPS DENG N. MOV MOV ELPS EXE_ALL: JNZ MOV MOV ELPS_EXE_ALL_1	SI,559 DI,374 BX,373 DS:WORD PTR STATUS,1 ELPS EXE ALL DS:WORD PTR PMAX,4 DS:WORD PTR PTNCNT,0FFFFH CX,1 :	:XC=559 ;YC=374 ;DY=373 :CHECK IF PPBUSY=1/0 :PMAX=4 :PTNCNT=0FFFFH				

PGROUP	NAME GROUP	RECFILL PROG	
PROG	SEGMENT	BYTE PUBLIC 'PROG'	
	;	CS:PGROUP	
	PUBLIC ;	RECFILL DEMO	
	EXTRN EXTRN	DMCLO_C_WAIT:NEAR, MES_CL_WAIT: MES CL_WAIT_FILL:NEAR	NEAR
	;		
	EXTRN EXTRN	STATUS:WORD, PDISPSL:WORD, PDISI PTNPH:WORD, PTNPL:WORD, X:WORD,	
	EXTRN EXTRN	DY:WORD, PTNCNT:WORD, PLANES:WOR COM:WORD, AGDC_SEG:WORD	RD, MOD10:WORD
RECFILL	; —	PROC NEAR	
	;	TRUC NEAR	
ļ	< RECTA	NGLE FILLING DEMO >	
;			
*`	; MOV	SI,1	
	CALL	MES_CL_WAIT	;MESSAGE(1)
RECFILL	, DEM0_1:		
	TEST JNZ	DS:WORD PTR STATUS,1 RECFILL DEMO 1	;CHECK IF PPBUSY=1/0
	MOV	DS:WORD PTR PDISPSL, 10H	;PDISPSL=10H
	MOV	DS:WORD PTR PDISPSH,0 DS:WORD PTR PMAX,4	;PDISPSH=0 ;PMAX=4
	MOV MOV	DS:WORD PTR PTNPH,0 BX,903EH	;PTNPH=0 ; <r_recfill></r_recfill>
			;TL=0,SS=1,WL=1,WR=1 ;FAST=1
	CALL	RECFILL EXE	,1431-1
	MOV CALL	SI,2 MES_CL_WAIT_FILL	;MESSAGE(2)
	; MOV	BX,90BCH	; <r recfill=""></r>
			;TL=1,SS=1,WL=1,WR=1
	CALL	RECFILL EXE	;FAST=0
	MOV CALL	SI,3 MES_CL_WAIT_FILL	;MESSAGE(3)
	; MOV	BX,90ACH	; <r_recfill></r_recfill>
			;TL=1,SS=0,WL=1,WR=1 ;FAST=0
	CALL	RECFILL_EXE	,7451-0
	MOV	\$1,4	
		-1-	
		PC AV	
	MOV MOV	ES,AX CX,12	REPETITION COUNTS
	MOV TEST	DX,0F7D0H BL,10H	;TILE 0F800H
	JZ MOV	RECFILL_2 DX.0FA30H	;CHECK IF "SS"=1/0 ;TILE_SS 0FA40H
RECFILL	2:		,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
	TEST JNZ	ES:WORD PTR STATUS,1 RECFILL_2	;CHECK IF PPBUSY=1/0
	TEST JNZ	BL,10H RECFILL_2_1	;CHECK IF "SS"=1/0
RECFILL	ADD 2 1	DX,20H	
KECT TEE	ADD	DX,10H	;PTNPL=(?)
	LODSW	ES:WORD PTR PTNPL,DX	
	MOV LODSW	ES:WORD PTR X,AX	;X=(?)
	MOV LODSW	ES:WORD PTR Y,AX	;Y=(?)
	MOV	ES:WORD PTR DXX,AX	;DX=(?)
	LODSW	ES:WORD PTR DY,AX	;DY=(?)
	TEST	BL,80H	
	JNZ	RECFILL_TL1	;CHECK IF "TL"=1/0
	JNZ MOV	RECFILL_TL1 ES:WORD PTR PTNCNT,0	;CHECK IF "TL"=1/0 ;PTNCNT=0
	JNZ MOV MOV AND	RECFILL_TL1 ES:WORD PTR PTNCNT,0 AX,CX AX,7	;PTNCNT=0
	JNZ MOV MOV AND JNZ MOV	RECFILL_TL1 ES:WORD PTR PTNCNT,0 AX,CX	
RECFILL	JNZ MOV MOV AND JNZ MOV	RECFILL_TL1 ES:WORD PTR PTNCNT,0 AX,CX AX,7 RECFILL_3 AX,3	;PTNCNT=0
RECFILL	JNZ MOV MOV AND JNZ MOV 3: MOV MOV	RECFILL_TL1 ES:WORD PTR PTNCNT,0 AX,CX AX,7 RECFILL_3 AX,3 ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1	;PTNCNT=0 ;CHECK IF "WHITE"
	JNZ MOV AND JNZ MOV 3: MOV JMP ;	RECFILL_TL1 ES:WORD PTR PTNCNT,0 AX,CX AX,7 RECFILL_3 AX,3 ES:WORD PTR PLANES,AX	;PTNCNT=0 ;CHECK IF "WHITE" ;PLANES=(?)
RECFILL. RECFILL.	JNZ MOV AND JNZ MOV 3: MOV JMP ; TL1: MOV	RECFILL_TLI ES:WORD PTR PTNCNT,O AX,CX AX,T RECFILL_3 AX,3 ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 RECFILL_TL ES:WORD PTR PTNCNT,16	;PTNCNT=0 ;CHECK IF "VHITE" ;PLANES=(?) ;MOD1=0,MOD0=1 ;PTNCNT=16
	JNZ MOV MOV AND JNZ 3: MOV JNV JMP ; TL1:	RECFILL_TL1 ES:WORD PTR PTNCNT,0 AX,CX AX,7 RECFILL_3 AX,3 ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 RECFILL_TL	;PTNCNT=0 ;CHECK IF "WHITE" ;PLANES=(?) ;MOD1=0,MOD0=1
	JNZ MOV MOV JNZ MOV 3: MOV JMP ; TL1: MOV MOV MOV TL1:	RECFILL_TLI ES:WORD PTR PTNCNT,O AX,CX AX,7 RECFILL_3 AX,3 ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 RECFILL_TL ES:WORD PTR PTNCNT,16 ES:WORD PTR PTNCNT,16 ES:WORD PTR PLANES,7 ES:WORD PTR MODIO,0	;PTNCNT=0 ;CHECK IF "WHITE" ;PLANES=(?) ;MOD1=0,MOD0=1 ;PTNCNT=16 ;PLANES=7 ;MOD1=0,MOD0=0
RECFILL.	JNZ MOV AND JNZ MOV 3: MOV 3: TL1: TL1: MOV MOV TL1: MOV MOV LOOP	RECFILL_TL1 ES:WORD PTR PTNCNT,O AX,CX AX,7 RECFILL_3 AX,3 ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 RECFILL_TL ES:WORD PTR PTNCNT,16 ES:WORD PTR PTNCNT,16 ES:WORD PTR PTNCNT,16 ES:WORD PTR PTNCNT,16 ES:WORD PTR PTNCNT,16 ES:WORD PTR COM,BX RECFILL_2	;PTNCNT=0 ;CHECK IF "WHITE" ;PLANES=(?) ;MOD1=0,MOD0=1 ;PTNCNT=16 ;PLANES=7
RECFILL.	JNZ MOV MOV AND JNZ 3: MOV 3: MOV JMP ; TL1: MOV MOV MOV MOV TL: MOV	RECFILL_TLI ES:WORD PTR PTNCNT,O AX,CX AX,T RECFILL_3 AX,3 ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 RECFILL_TL ES:WORD PTR PTNCNT,16 ES:WORD PTR PLANES,7 ES:WORD PTR MODIO,0 ES:WORD PTR COM,BX	;PTNCNT=0 ;CHECK IF "WHITE" ;PLANES=(?) ;MOD1=0,MOD0=1 ;PTNCNT=16 ;PLANES=7 ;MOD1=0,MOD0=0
RECFILL. RECFILL.	JNZ MOV MOV JNZ JNZ 3: MOV 3: TLI: MOV JMP ; TLI: MOV MOV TLI: MOV ECP POP RET ;	RECFILL_TL1 ES:WORD PTR PTNCNT,O AX,CX AX,7 RECFILL_3 AX,3 ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 RECFILL_TL ES:WORD PTR PTNCNT,16 ES:WORD PTR PTNCNT,16 ES:WORD PTR PTNCNT,16 ES:WORD PTR PTNCNT,16 ES:WORD PTR PTNCNT,16 ES:WORD PTR COM,BX RECFILL_2	;PTNCNT=0 ;CHECK IF "WHITE" ;PLANES=(?) ;MOD1=0,MOD0=1 ;PTNCNT=16 ;PLANES=7 ;MOD1=0,MOD0=0
RECFILL.	JNZ MOV AND JNZ 3: MOV 3: TLI: TLI: MOV MOV MOV MOV HOV LOOP POP RET ; DATA L: DW	RECFILL_TLI ES:WORD PTR PTNCNT,0 AX,CX AX,CX RECFILL_3 AX,3 ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 RECFILL_TL ES:WORD PTR PTNCNT,16 ES:WORD PTR PLANES,7 ES:WORD PTR PLANES,7 ES:WORD PTR MODIO,0 ES:WORD PTR MODIO,0 ES:WORD PTR COM,BX RECFILL_2 DS	; PTNCNT=0 ; CHECK IF "WHITE" ; PLANES=(?) ; MOD1=0, MOD0=1 ; PTNCNT=16 ; PLANES=7 ; MOD1=0, MOD0=0 ; <com.flags></com.flags>
RECFILL. RECFILL.	JNZ MOV AND JNZ MOV 3: MOV 3: TL1: MOV JMP ; TL1: MOV UOOP POP RET ; DATA L: DW DW	RECFILL_TLI ES:WORD PTR PTNCNT,0 AX,CX AX,C RECFILL_3 AX,3 ES:WORD PTR PLANES,AX ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 RECFILL_TL ES:WORD PTR PTNCNT,16 ES:WORD PTR PLANES,7 ES:WORD PTR MODIO,0 ES:WORD PTR COM.BX RECFILL_2 DS 0032H.01AEH.0186H.0FEDEH 003CH.0104H.017CH.0FEEBH 010EH.019AH.017CH.0FEEBH 010EH.019AH.017CH.0FEEBH 010EH.019AH.017CH.0FEEBH	; PTNCNT=0 ; CHECK IF "WHITE" ; PLANES=(?) ; MOD1=0, MOD0=1 ; PTNCNT=16 ; PLANES=7 ; MOD1=0, MOD0=0 ; <com.flags> ; 1. X,Y,DX,DY ; 2. X,Y,DX,DY ; 3. X,Y,DX,DY</com.flags>
RECFILL. RECFILL.	JNZ MOV MOV JNZ JNZ MOV 3: TLI: MOV JMP ; TLI: MOV MOV TLI: MOV MOV LOOP POP RET ; DATA L: DW	RECFILL_TLI ES:WORD PTR PTNCNT,0 AX,CX AX,CX RECFILL_3 AX,3 ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 RECFILL_TL ES:WORD PTR PTNCNT,16 ES:WORD PTR PLANES,7 ES:WORD PTR PLANES,7 ES:WORD PTR MODIO,0 ES:WORD PTR COM,BX RECFILL_2 DS	; PTNCNT=0 ; CHECK IF "WHITE" ; PLANES=(?) ; MOD1=0, MOD0=1 ; PTNCNT=16 ; PLANES=7 ; MOD1=0, MOD0=0 ; <com.flags></com.flags>

	CALL	MES_CL_WAIT_FILL	;MESSAGE(4)
	, MOV	BX,903EH	<pre>;<r_recfill> ;TL=0,SS=1,WL=1,WR=1 <pre>SFACT=1</pre></r_recfill></pre>
	CALL	RECFILL_EXE_1	;FAST=1
	MOV	SI,5	
	CALL	MES_CL_WAIT_FILL	;MESSAGE(5)
	, MOV	BX,90BCH	; <r_recfill> ;TL=1,SS=1,WL=1,WR=1 ;FAST=0</r_recfill>
	CALL	RECFILL_EXE_1	, i AST V
	MOV	SI,6	
	CALL	MES_CL_WAIT_FILL	;MESSAGE(6)
	, MOV	BX,90ACH	<pre>;<r_recfill> ;TL=1,SS=0,WL=1,WR=1 ;FAST=0</r_recfill></pre>
	CALL	RECFILL EXE_1	
	MOV	S1,7	18566165(7)
	CALL :	MES_CL_WAIT_FILL	;MESSAGE(7)
	RET		
RECFILL	; EXE:		
	MOV	DI,1	;DRAW COUNTS
RECFILL		DECELLI EVE I	
	CALL DEC	RECFILL_EXE_L DI	
	JNZ	RECFILL_EXE1	
	RET		
RECFILL	, EXE_1:		
	MOV	DI,1	;DRAW COUNTS
RECFILL	EXE_11: CALL	RECFILL_EXE_H	
	DEC	DI	
	JNZ	RECFILL_EXE_11	
	RET		
RECFILL	, EXE L:		
	MOV	SI, OFFSET RECFILL_DATA_L	
DECELLI	JMP	RECFILL_1	
RECFILL	MOV	SI, OFFSET RECFILL_DATA_H	
ECFILL	1:		
	PUSH	DS	
	MOV MOV	AX,CS DS,AX	
	MOV	AX, WORD PTR AGDC_SEG	
		-2-	
		-	
	DW	0064H,012CH,0154H,0FF10H	; 6. X,Y,DX,DY
	DW DW	00E6H,0172H,014AH,0FF1AH 00DCH,0140H,0140H,0FF24H	; 7. X,Y,DX,DY ; 8. X,Y,DX,DY
	DW	0082H,015EH,0136H,0FF2EH	; 9. X,Y,DX,DY
	DW	008CH,0154H,012CH,0FF38H	;10. X,Y,DX,DY
	DW	00BEH,014AH,0122H,0FF42H	;11. X,Y,DX,DY
	DW DW	00BEH,014AH,0122H,0FF42H 00B4H,0168H,0118H,0FF4CH	;12. X,Y,DX,DY
	DW DW DW ;	00BEH,014AH,0122H,0FF42H	
	DW DW DW ; DATA_H:	008EH,014AH,0122H,0FF42H 0084H,0188H,0118H,0FF4CH 00AAH,0136H,010EH,0FF56H	;12. X,Y,DX,DY ;13. X,Y,DX,DY
	DW DW DW ; DATA_H: DW	008EH,014AH,0122H,0FF42H 0084H,0188H,0118H,0FF4CH 00AAH,0136H,010EH,0FF56H 001EH,028CH,0280H,0FE36H	 ;12. X,Y,DX,DY ;13. X,Y,DX,DY ; 1. X,Y,DX,DY
	DW DW DW ; DATA_H:	008EH,014AH,0122H,0FF42H 0084H,0188H,0118H,0FF4CH 00AAH,0136H,010EH,0FF56H	<pre>;12. X, Y, DX, DY ;13. X, Y, DX, DY ; 1. X, Y, DX, DY ; 2. X, Y, DX, DY ; 3. X, Y, DX, DY</pre>
	DW DW ; DATA_H: DW DW DW DW DW	008EH,014AH,0122H,0FF42H 0084H,0188H,0118H,0FF4CH 00AAH,0136H,010EH,0FF56H 001EH,028CH,0280H,0FE36H 0020H,0195H,0280H,0FE42H 0188H,029EH,0298H,0FE4EH 01A9H,0183H,028CH,0FE5AH	<pre>;12. X, Y, DX, DY ;13. X, Y, DX, DY ; 1. X, Y, DX, DY ; 2. X, Y, DX, DY ; 3. X, Y, DX, DY ; 4. X, Y, DX, DY</pre>
	DW DW DW ; DATA_H: DW DW DW DW DW	008EH,014AH,0122H,0FF42H 0084H,0188H,0118H,0FF4CH 00AAH,0136H,010EH,0FF5GH 001EH,028CH,0280H,0FE3GH 002DH,0195H,028AH,0FE42H 0188H,029CH,0298H,0FE4EH 01A9H,0183H,028CH,0FE5AH 005AH,0280H,0280H,0FE6GH	<pre>;12. X, Y, DX, DY ;13. X, Y, DX, DY ; 1. X, Y, DX, DY ; 2. X, Y, DX, DY ; 3. X, Y, DX, DY ; 4. X, Y, DX, DY ; 5. X, Y, DX, DY</pre>
	DW DW DW ; DATA_H: DW DW DW DW DW DW DW	008EH,014AH,0122H,0FF42H 0084H,0188H,0118H,0FF4CH 00AAH,0136H,010EH,0FF56H 001EH,028CH,0280H,0FE36H 002DH,0195H,02A4H,0FE42H 0188H,029EH,0298H,0FE4EH 01A9H,0183H,028CH,0FE5AH 005AH,0280H,0280H,0FE66H 0069H,01D1H,0274H,0FE72H	<pre>;12. X, Y, DX, DY ;13. X, Y, DX, DY ; 1. X, Y, DX, DY ; 2. X, Y, DX, DY ; 3. X, Y, DX, DY ; 4. X, Y, DX, DY</pre>
	DW DW DW ; DATA_H: DW DW DW DW DW DW DW DW DW DW DW	008EH,014AH,0122H,0FF42H 008EH,018BH,0118H,0FF4CH 00AAH,0136H,010EH,0FF5GH 001EH,028CH,0280H,0FE3GH 002DH,0195H,024AH,0FE42H 018BH,029CH,0298H,0FE42H 0189H,0183H,028CH,0FE5AH 005AH,0280H,0274H,0FE72H 017CH,0262H,0268H,0FE7EH 017CH,0262H,0268H,0FE7EH	<pre>;12. X, Y, DX, DY ;13. X, Y, DX, DY ; 1. X, Y, DX, DY ; 2. X, Y, DX, DY ; 3. X, Y, DX, DY ; 4. X, Y, DX, DY ; 5. X, Y, DX, DY ; 6. X, Y, DX, DY ; 7. X, Y, DX, DY ; 8. X, Y, DX, DY</pre>
	DW DW DW CATA_H: DW DW DW DW DW DW DW DW DW DW DW DW DW	008EH,014AH,0122H,0FF42H 0084H,0188H,0118H,0FF4CH 00AAH,0136H,010EH,0FF56H 001EH,028CH,0280H,0FE36H 002DH,0135H,02A4H,0FE42H 0188H,028CH,0288H,0FE4EH 01A9H,0183H,028CH,0FE5AH 005AH,0280H,0280H,0FE66H 0069H,018DH,0274H,0FE72H 017CH,0262H,0268H,0FE7EH 016DH,01EFH,025CH,0FE36H	<pre>;12. X, Y, DX, DY ;13. X, Y, DX, DY ;13. X, Y, DX, DY ; 2. X, Y, DX, DY ; 3. X, Y, DX, DY ; 4. X, Y, DX, DY ; 5. X, Y, DX, DY ; 6. X, Y, DX, DY ; 8. X, Y, DX, DY ; 9. X, Y, DX, DY</pre>
	DW DW DW DATA_H: DW DW DW DW DW DW DW DW DW DW DW DW DW	008EH,014AH,0122H,0FF42H 008EH,0168H,0118H,0FF4CH 00AAH,0136H,010EH,0FF56H 001EH,028CH,0280H,0FE36H 0020H,0195H,024AH,0FE42H 0188H,029EH,0298H,0FE4EH 0189H,0183H,0280H,0FE66H 0069H,018H,0280H,0FE56H 0069H,018H,0274H,0FE72H 017CH,0262H,0286H,0FE72H 017CH,0262H,025CH,0FE8AH 0096H,024H,0250H,0FE8AH 0096H,0240H,0250H,0FE8AH	<pre>;12. X, Y, DX, DY ;13. X, Y, DX, DY ;13. X, Y, DX, DY ; 2. X, Y, DX, DY ; 3. X, Y, DX, DY ; 4. X, Y, DX, DY ; 5. X, Y, DX, DY ; 6. X, Y, DX, DY ; 7. X, Y, DX, DY ; 8. X, Y, DX, DY ; 9. X, Y, DX, DY</pre>
	DW DW DW CATA_H: DW DW DW DW DW DW DW DW DW DW DW DW DW	008EH,014AH,0122H,0FF42H 0084H,0188H,0118H,0FF4CH 00AAH,0136H,010EH,0FF56H 001EH,028CH,0280H,0FE36H 002DH,0135H,02A4H,0FE42H 0188H,028CH,0288H,0FE4EH 01A9H,0183H,028CH,0FE5AH 005AH,0280H,0280H,0FE66H 0069H,018DH,0274H,0FE72H 017CH,0262H,0268H,0FE7EH 016DH,01EFH,025CH,0FE36H	<pre>;12. X, Y, DX, DY ;13. X, Y, DX, DY ;13. X, Y, DX, DY ; 2. X, Y, DX, DY ; 3. X, Y, DX, DY ; 4. X, Y, DX, DY ; 5. X, Y, DX, DY ; 6. X, Y, DX, DY ; 7. X, Y, DX, DY ; 8. X, Y, DX, DY ; 10. X, Y, DX, DY ;11. X, Y, DX, DY ;12. X, Y, DX, DY</pre>
; RECFILL.	DW DW CW CP DATA_H: DW DW DW DW DW DW DW DW DW DW DW DW DW	008EH,014AH,0122H,0FF42H 008EH,018BH,0118H,0FF4CH 00AAH,0136H,010EH,0FF5GH 001EH,028CH,0280H,0FE3GH 002DH,0195H,0244H,0FE42H 018BH,029EH,0298H,0FE42H 018BH,028CH,028SH,0FE6GH 005AH,0280H,0280H,0FE6GH 005AH,0280H,0274H,0FE72H 017CH,0262H,0263H,0FE6GH 0065H,01EFH,025CH,0FE8AH 0045H,0226H,0238H,0FEAEH	<pre>;12. X, Y, DX, DY ;13. X, Y, DX, DY ;13. X, Y, DX, DY ; 2. X, Y, DX, DY ; 3. X, Y, DX, DY ; 4. X, Y, DX, DY ; 5. X, Y, DX, DY ; 6. X, Y, DX, DY ; 7. X, Y, DX, DY ; 8. X, Y, DX, DY ; 9. X, Y, DX, DY ; 10. X, Y, DX, DY ; 11. X, Y, DX, DY</pre>

; RECFILL_DEMO PROG ENDS END

ENDP

```
CRLFILL
         NAME
PGROUP
         GROUP PROG
SEGMENT BYTE PUBLIC
                                      'PROG'
PROG
         ASSUME
                  CS:PGROUP
         PUBLIC CRLFILL DEMO, ELPSFILL DEMO
                  MES_CL_WAIT:NEAR, MES_CL_WAIT_FILL:NEAR
         EXTRN
                  STATUS:WORD, PDISPSL:WORD, PDISPSH:WORD, PMAX:WORD
PTNPH:WORD, AGDC_SEG:WORD, PTNPL:WORD, DXX:WORD
XC:WORD, YC:WORD, PTKCNT:WORD, PLANES:WORD, MODIO:WORD
COM:WORD, WORK_1:WORD, WORK_2:WORD, DY:WORD
DHH:WORD, DV:WORD
         EXTRN
         EXTRN
         EXTRN
         EXTRN
         EXTRN
CRLFILL PROC
                   NEAR
         < CIRCLE FILLING DEMO >
CRLFILL DEMO:
         MOV
                   SI,34
MES_CL_WAIT
                                                         ;MESSAGE(34)
         CALL
CRLFILL_DEMO_1:
                   DS:WORD PTR STATUS,1
         TEST
                                                         ;CHECK IF PPBUSY=1/0
                   CRLFILL DEMO_1
DS:WORD PTR PDISPSL,10H
          INZ
                                                         ;PDISPSL=10H
         MOV
                   DS:WORD PTR PDISPSH,0
                                                         ;PDISPSH=0
          MOV
                   DS:WORD PTR PMAX.4
         MOV
                                                         :PMAX=4
                   DS:WORD PTR PTNPH,0
                                                         ;PTNPH=0
         MOV
          MOV
                   BX,503CH
                                                         ;<CRLFILL> TL=0,SS=1
                   CRLFILL EXE
         CALL
          MOV
                   51,2
                   MES_CL_WAIT_FILL
         CALL
                                                         :MESSAGE(2)
          MOV
                   BX,50BCH
                                                         ;<CRLFILL> TL=1,SS=1
         CALL
                   CRLFILL EXE
         MOV
                   $1.3
         CALL
                   MES_CL_WAIT_FILL
                                                         :MESSAGE(3)
          MOV
                   BX,50ACH
                                                         ;<CRLFILL> TL=1,SS=0
         CALL
                   CRLFILL EXE
         MOV
                   $1.4
          CALL
                   MES_CL_WAIT_FILL
                                                         ;MESSAGE(4)
          MOV
                   BX,503CH
                                                         ;<R_RECFILL> TL=0,SS=1
         CALL
                   CRLFILL EXE_1
         MOV
                   $1.5
                                      -1-
          JNZ
                   CRLFILL_2_1
                                                         :CHECK IF "SS"=1/0
         ADD
                   DX.20H
CRLFILL_2_1:
                   DX.10H
          ADD
                   ES:WORD PTR PTNPL,DX
                                                         ;PTNPL=(?)
         MOV
          LODSW
                   ES:WORD PTR DXX,AX
                                                         ;DXX=(?)
         MOV
         LODSW
                   ES:WORD PTR XC.AX
         MOV
                                                         :XC=(?)
         LODSW
                   ES:WORD PTR YC,AX
                                                         :YC=(?)
          MOV
         TEST
                   BL. 80H
                   CRLFILL_TL1
          JNZ
                                                         ;CHECK IF "TL"=1/0
          MOV
                   ES:WORD PTR PTNCNT,0
                                                         :PTNCNT=0
              MOV AX.CX
          AND
                   AX,7
                   CRLFILL_3
                                                         CHECK IF "WHITE"
          JNZ
          MOV
                   AX.3
CRLFILL_3:
                   ES:WORD PTR PLANES.AX
          MOV
                                                         :PLANES=(?)
                   ES:WORD PTR MODIO,1
                                                         ;MOD1=0,MOD0=1
         MOV
          JMP
                   CRLFILL_TL
CRLFILL_TL1:
          MOV
                   ES:WORD PTR PTNCNT.16
                                                         :PTNCNT=16
                   ES:WORD PTR PLANES,7
                                                          PLANES=7
          MOV
                   ES:WORD PTR MODIO,0
                                                         ;MOD1=0,MOD0=0
          MOV
CRLFILL TL:
          MOV
                   ES:WORD PTR COM, BX
                                                         ;<COM.FLAGS>
          LOOP
                   CRLFILL_2
          POP
                   DS
          RET
;, .,
          < ELLIPSE FILLING DEMO >
ELPSFILL DEMO:
          MOV
                   $1,35
         CALL
                   MES_CL_WAIT
                                                         :MESSAGE(35)
          MOV
                   CS:WORD PTR WORK_1,16
                                                         ;DH=16
         MOV
                   CS:WORD PTR WORK 2,9
                                                         ;DV=9
         CALL
                   ELPSFILL_DEM0_1
                   CS:WORD PTR WORK_1,9
CS:WORD PTR WORK_2,16
          MOV
                                                         :DH=9
         MOV
                                                         :DV=16
                   ELPSFILL_DEM0_1
          CALL
         RET
ELPSFILL_DEM0_1:
```

```
:MESSAGE(5)
         CALL
                 MES CL WAIT FILL
                                                     ;<CRLFILL> TL=1,SS=1
         MOV
                  BX.50BCH
                 CRLFILL EXE_1
         CALL
         MOV
                  $1.6
                                                     ;MESSAGE(6)
                 MES CL WAIT FILL
         CALL
         MOV
                  RY SOACH
                                                     :<CRLF11.1> T1=1.5S=0
                 CRLFILL EXE_1
         CALL
                  $1,7
         MOV
                 MES CL WAIT FILL
         CALL
                                                     :MESSAGE(7)
         RET
CRLFILL_EXE:
                                                     DRAW COUNTS
         MOV
                  D1.1
CRIFILL EXEL:
                 CRLFILL_EXE_L
         CALL
         DEC
                 CRLFILL_EXE1
         JNZ
         RET
CRLFILL EXE_1:
                 DI.1
                                                     COUNTS
         MOV
CRLFILL EXE_11:
                  CRLFILL_EXE_H
         CALL
         DEC
                  DI
         JNZ
                  CRLFILL_EXE_11
         RET
CRLFILL EXE L:
                  SI, OFFSET CRLFILL_DATA_L
         MOV
                  CRLFILL_1
         JMP
CRLFILL EXE H:
                  SI, OFFSET CRLFILL_DATA_H
         MOV
CRLFILL_1:
         PUSH
                 DS
                  AX,CS
         MOV
                 DS,AX
AX,WORD PTR AGDC_SEG
         MOV
         MOV
         MOV
                  ES.AX
                                                     REPETITION COUNTS
         MOV
                  CX.12
                                                              .. 0F800H
                  DX,OF7DOH
                                                     ;TILE
         MOV
                  BL,10H
         TEST
                                                     CHECK IF "SS"=1/0
                  CRIFILL 2
         17
                                                     ;TILE_SS .. OFA40H
         MOV
                  DX.OFA30H
CRLFILL 2:
         TEST
                  ES:WORD PTR STATUS,1
                  CRLFILL_2
                                                     CHECK IF PPBUSY=1/0
         JNZ
         TEST
                  BL., 10H
                                    -2-
                  DS:WORD PTR STATUS,1
ELPSFILL DEMO 1
         TEST
                                                     ;CHECK IF PPBUSY=1/0
         INZ
         MOV
                  DS:WORD PTR PDISPSL,10H
DS:WORD PTR PDISPSH,0
                                                     :PDISPSL=10H
                                                     :PDISPSH=0
         MOV
                  DS:WORD PTR PMAX,4
                                                     ;PMAX=4
         MOV
         MOV
                  DS:WORD PTR PTNPH,0
                                                     :PTNPH=0
                                                     ;<ELPSFILL> TL=0,SS=1
         MOV
                  BX.5C3CH
                  ELPSFILL EXE
         CALL
                  SI,2
MES_CL_WAIT_FILL
         MOV
                                                     ;MESSAGE(2)
         CALL
                                                     :<ELPSFILL> TL=1.SS=1
         MOV
                  BX.5CBCH
                  ELPSFILL_EXE
         CALL
         MOV
                  $1.3
                                                     ;MESSAGE(3)
                  MES CL WALT FILL
         CALL
                                                     ;<ELPSFILL> TL=1,SS=0
         MOV
                  BX. 5CACH
                  ELPSFILL EXE
         CALL
         MOV
                  51,4
         CALL
                  MES CL WAIT FILL
                                                     :MESSAGE(4)
         MOV
                  BX.5C3CH
                                                     ;<R_RECFILL> TL=0,SS=1
                  ELPSFILL EXE_1
         CALL
                  $1,5
         MOV
                  MES CL WAIT FILL
                                                     :MESSAGE(5)
         CALL
         MOV
                                                     ;<ELPSFILL> TL=1,SS=1
                  BX,5CBCH
         CALL
                  ELPSFILL_EXE_1
         MOV
                  $1.6
                  MES_CL_WAIT_FILL
                                                     :MESSAGE(6)
         CALL
         MOV
                  BX, 5CACH
                                                     ;<ELPSFILL> TL=1,SS=0
         CALL
                  ELPSFILL_EXE_1
         MOV
                  $1.7
                  MES_CL_WAIT_FILL
                                                     :MESSAGE(7)
         CALL
         RET
ELPSFILL_EXE:
                                                     ;DRAW COUNTS
         MOV
                  D1,1
ELPSFILL EXE1:
                  ELPSFILL EXE L
         CALL
         DEC
                  ELPSFILL_EXE1
         INZ.
         RET
ELPSFILL EXE_1:
MOV
                                                     ;DRAW COUNTS
                  D1,1
ELPSFILL EXE_11:
                  ELPSFILL_EXE_H
         CALL
         DEC
                  DI
```

	JNZ RET	ELPSFILL_EXE_11		ELPSFILL_TLI MOV MOV	1: ES:WORD PTR PTNCNT,16 ES:WORD PTR PLANES,7	;PTNCNT=16 ;PLANES=7
ELPSFILI	L EXE L: MOV JMP	SI, OFFSET CRLFILL DATA L ELPSFILL_1		MOV ELPSFILL_TL: MOV LOOF	ES:WORD PTR MODIO,O : ES:WORD PTR COM,BX	;MOD1=0,MOD0=0 ; <com.flags></com.flags>
ELPSFIL	, EXE_H:	SI, OFFSET CRLFILL DATA H		POP		
ELPSFIL	MOV L_1: PUSH	DS		CRLFILL DATA		
ELPSFILI	MOV MOV MOV MOV MOV TEST JZ MOV	AX.CS DS.AX AX.WORD PTR AGDC_SEG ES.AX CX.12 DX.OF7DOH BL.10H ELPSFILL_2 DX.OFA30H	REPETITION COUNTS TILE OF800H CHECK IF "SS"=1/0 TILE_SS OFA40H	DW DW DW DW DW DW DW DW DW DW DW	150, 200, 160 145, 490, 310 140, 480, 180 135, 230, 290 130, 240, 200 125, 450, 270 120, 440, 220 115, 270, 250 110, 280, 240 105, 410, 230	; 1. DX, XC, YC ; 2. DX, XC, YC ; 3. DX, XC, YC ; 4. DX, XC, YC ; 5. DX, XC, YC ; 6. DX, XC, YC ; 7. DX, XC, YC ; 8. DX, XC, YC ; 9. DX, XC, YC ; 10. DX, XC, YC
	TEST JNZ	ES:WORD PTR STATUS,1 ELPSFILL_2	;CHECK IF PPBUSY=1/0	D W D W	100, 400, 260 95, 310, 210	;11. DX, XC, YC ;12. DX, XC, YC
	TEST JNZ	BL,10H ELPSFILL_2_1	;CHECK IF "SS"=1/0	; CRLFILL_DAT/	A_H:	
ELPSFIL	ADD L_2_1:	DX,20H		D W D W	240, 326, 256 232, 784, 496	; 1. DX, XC, YC ; 2. DX, XC, YC
	ADD MOV	DX,10H ES:WORD PTR PTNPL,DX	;PTNPL=(?)	D W D W	224, 768, 288 216, 364, 464	; 3. DX, XC, YC ; 4. DX, XC, YC
	LODSW Mov	ES:WORD PTR DY,AX	;DY=(?)	D W D W	208, 380, 320 200, 720, 432	; 5. DX, XC, YC ; 6. DX, XC, YC
	LODSW Mov	ES:WORD PTR XC,AX	;XC=(?)	D W D W	192, 704, 352 184, 428, 400	; 7. DX, XC, YC ; 8. DX, XC, YC
	LODSW Mov	ES:WORD PTR YC,AX	;YC=(?)	D W D W	176, 444, 384 168, 656, 368	; 9. DX, XC, YC ;10. DX, XC, YC
	MOV MOV	AX,CS:WORD PTR WORK_1 ES:WORD PTR DHH,AX	;DHH=(?)	D W D W	160, 640, 416 152, 492, 336	;11. DX, XC, YC ;12. DX, XC, YC
	MOV MOV	AX,CS:WORD PTR WORK_2 ES:WORD PTR DV,AX	; DV=(?)	;	_	
	TEST JNZ MOV MOV	BL,80H ELPSFILL_TL1 ES:WORD PTR PTNCNT,0 AX,CX	;CHECK IF "TL"=1/0 ;PTNCNT=0	CRLFILL ENDI PROG END END	S	
ELPSFILI	AND JNZ MOV	AX,7 ELPSFILL_3 AX,3	;CHECK IF "WHITE"			
66131161	MOV MOV JMP ;	ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 ELPSFILL_TL	;PLANES=(?) ;MOD1=0,MODO=1			
		F				
					- C -	
	NAME	-5-		CA11	-6-	
PGROUP PROG	SEGMENT	-3- TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP		CALL MOV CALL Ret	L TRAFILL EXE H	;MESSAGE(7)
	GROUP SEGMENT ASSUME ;	TRAFILL PROG BYTE PUBLIC 'PROG'		MOV CALL RET ; TRAFILL_EXE	L TRAFILL_EXE_H SI,7 L MES_CL_WAIT_FILL	:MESSAGE(7)
	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP	FILL:NEAR	MOV CALL RET ; TRAFILL EXE MOV JMP	L TRAFILL EXE H SI,7 MES_CL_WAIT_FILL L: SI,0FFSET_TRAFILL_DATA_L TRAFILL_1	;MESSAGE(7)
	GROUP SEGMENT ASSUME ; PUBLIC ;	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE_H	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD	MOV CALI RET ; TRAFILL EXE MOV	L TRAFILL EXE H SI.7 L MES_CL_WAIT FILL SI.0FFSET TRAFILL DATA L TRAFILL_1 H: SI.0FFSET TRAFILL_DATA_H D S:WORD PTR STATUS,1	;MESSAGE(7) ;CHECK IF PPBUSY=1/0 ;PDISPSL=10H
	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN ; EXTRN EXTRN EXTRN EXTRN ;	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL DEMO, TRAFILL EXE H MES_CL_WAIT:NEAR, MES_CL_WAIT STATUS:WORD, PDISPSL:WORD, YDI PTNPH:WORD, YTNPL:WORD, X:WORD YS:WORD, YE:WORD, X:WORD, PTN	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD	MOV CALI RET ; TRAFILL EXE MOV TRAFILL EXE MOV TRAFILL 1: TEST JNZ	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL SI,0FFSET TRAFILL_DATA L TRAFILL_1 H: SI,0FFSET TRAFILL_DATA_H T DS:WORD PTR STATUS,1 TRAFILL_1	CHECK IF PPBUSY=1/0
PROG	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ;	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE_H MES_CL_WAIT:NEAR, MES_CL_WAIT STATUS:WORD, PDISPSL:WORD, PDI PTNPH:WORD, PTNPL:WORD, X:WORD YS:WORD, YE:WORD, XE:WORD, PTN MODIO:WORD, COM:WORD, AGDC_SEG	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD	MOV C ALL RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TESI JNZ MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL L: SI,0FFSET TRAFILL_DATA L TRAFILL_1 H: SI,0FFSET TRAFILL_DATA_H T DS:WORD PTR TAFILL_DATA_H T DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,0H DS:WORD PTR PTR PDISPSL,0H DS:WORD PTR	;CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0
PROG	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN ; CTRAPE: ; DEMO:	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL DEMO, TRAFILL EXE H MES_CL_WAIT:NEAR, MES_CL_WAIT STATUS:WORD, PDISPSL:WORD, PDI PTNPH:WORD, PTNPL:WORD, X:WORD PTNPH:WORD, TPNPL:WORD, X:WORD NEAR ZOID FILLING DEMO >	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV PUSI MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL L: SI,0FFSET TRAFILL_DATA L TRAFILL_1 H: SI,0FFSET TRAFILL_DATA_H T DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PTNPH,0 H DS AX,CS DS,AX AX,WORD PTR AGDC_SEG ES,AX	;CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0 ;PMAX=4
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL DEMO, TRAFILL EXE H MES_CL_WAIT:NEAR, MES_CL_WAIT STATUS:WORD, PDISPSL:WORD, PDI PTXPH:WORD, PTNPL:WORD, X:WORD, PTN MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > \$1,9 MES_CL_WAIT	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;MESSAGE(9)	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL TRAFILL_1 H: SI,0FFSET TRAFILL_DATA L TRAFILL_1 DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PDNPH,0 H DS:WORD PTR PNNPH,0 H AX.CS DS,AX AX.WORD PTR AGDC_SEG ES.AX CX.10 DX.0F7D0H	;CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0 ;PMAX=4
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL MOV	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL DEMO, TRAFILL EXE H MES_CL_WAIT:NEAR, MES_CL_WAIT STATUS:WORD, PDISPSL:WORD, PDI PTNPH:WORD, PDISPSL:WORD, X:WORD YS:WORD, VE:WORD, X:WORD, X:WORD MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > \$1,9 MES_CL_WAIT BX,703CH	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TESI JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL L: SI,0FFSET TRAFILL_DATA L TRAFILL_1 H: SI,0FFSET TRAFILL_DATA_H T DS:WORD PTR TAFILL_DATA_H T DS:WORD PTR PDISPSL,10H DS:WORD PTR AGDC_SEG ES.AX CX,10 DX.0FTDOH T BL,10H TRAFILL_2	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMAX=4 ;PTNPH=0 :TILE OF800H :CHECK IF "SS"=1/0
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL MOV CALL MOV	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL DEMO, TRAFILL EXE H MES_CL_WAIT:NEAR, MES_CL_WAIT STATUS:WORD, PDISPSL:WORD, PDI PTNPH:WORD, PDISPSL:WORD, X:WORD YS:WORD, YE:WORD, XE:WORD, X:WORD MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > SI,9 MES_CL_WAIT BX.703CH TRAFILL_EXE L SI,2	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;MESSAGE(9) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1</a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL TRAFILL_1 H: SI,0FFSET TRAFILL_DATA L TRAFILL_1 DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR AGDC_SEG ES,AX AX,WORD PTR AGDC_SEG ES,AX CX,10 DX,0F7D0H T BL,10H TRAFILL_2 DX,0FA30H	;CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0 ;PMAX=4 ;PTNPH=0 ;TILE 0F800H
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL MOV CALL	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE.H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PDI PTXPH:WORD, PTNPL:WORD, X:WORD YS:WORD, VE:WORD, XE:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > SI,9 MES_CL_WAIT BX,703CH TRAFILL_EXE L	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;MESSAGE(9) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1 ;MESSAGE(2) ;<a_trafill></a_trafill></a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL TRAFILL_1 H: SI,0FFSET TRAFILL_DATA L TRAFILL_1 DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,0H DS:WORD PTR PDISPSL,0H DS:WORD PTR PDISPSL,0H DS:WORD PTR AGDC_SEG ES,AX CX,10 DX,0F7D0H T BL,10H TRAFILL_2 DX,0FA30H T ES:WORD PTR STATUS,1 TRAFILL_2	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMAX=4 ;PTNPH=0 :TILE OF800H :CHECK IF "SS"=1/0
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL MOV CALL MOV CALL MOV CALL	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE_H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PDI PTXPH:WORD, PTNPL:WORD, X:WORD YS:WORD, VE:WORD, XE:WORD, PTN MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > SI,9 MES_CL_WAIT BX,703CH TRAFILL_EXE_L SX,70BCH TRAFILL_EXE_L	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;MESSAGE(9) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1 ;MESSAGE(2)</a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL L: SI,0FFSET TRAFILL DATA L TRAFILL_1 H: SI,0FFSET TRAFILL_DATA_H T DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,0H DS:WORD PTR PDISPSL,0H T BL,10H TRAFILL_2 T BL,10H TRAFILL_2,1	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMAX=4 ;PTNPH=0 :TILE OF800H :CHECK IF "SS"=1/0 :TILE_SS OFA40H
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL MOV CALL MOV CALL MOV CALL	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL DEMO, TRAFILL EXE H MES_CL_WAIT:NEAR, MES_CL_WAIT STATUS:WORD, PDISPSL:WORD, PDI PTNPH:WORD, PTNPL:WORD, X:WORD YS:WORD, YE:WORD, XE:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > SI,9 MES_CL_WAIT BX,703CH TRAFILL_EXE L SI,2 MES_CL_WAIT_FILL BX,70BCH TRAFILL_EXE L SI,3 MES_CL_WAIT_FILL SI,3 MES_CL_WAIT_FILL	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;MESSAGE(9) ;<a_trafill> ;TL=0.SS=1.WL=1.WR=1 ;MESSAGE(2) ;<a_trafill> ;TL=1.SS=1.WL=1.WR=1 ;MESSAGE(3)</a_trafill></a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 MES_CL_WAIT FILL TRAFILL_1 H: SI,0FFSET TRAFILL_DATA L TRAFILL_1 DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PTAGDC_SEG ES,AX CX,10 DX,0FTD0H T BL,10H TRAFILL_2 DX,0FA30H T ES:WORD PTR STATUS,1 TRAFILL_2 T BL,10H TRAFILL_2_1 DX,20H	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMAX=4 :PTNPH=0 ;TILE 0F800H :CHECK IF "SS"=1/0 ;TILE_SS 0FA40H ;CHECK IF PPBUSY=1/0
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE.H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, VAIT TRAFILQRD, PTNPL:WORD, X:WORD, PTN MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > SI.9 MES_CL_WAIT BX,703CH TRAFILL_EXE L SI.2 MES_CL_WAIT_FILL BX,70BCH TRAFILL_EXE L SI.3 MES_CL_WAIT_FILL BX,70ACH	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;MESSAGE(9) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1 :MESSAGE(2) ;<a_trafill> ;TL=1,SS=1,WL=1,WR=1</a_trafill></a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL I: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L SI,0FFSET TRAFILL DATA L TRAFILL_1 H: SI,0FFSET TRAFILL_DATA_H T DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 T SI,0 DX,0F7D0H T BL,10H TRAFILL_2 T BL,10H TRAFILL_2,1 DX,20H DX,10H ES:WORD PTR PTNPL,DX	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMAX=4 :PTNPH=0 ;TILE 0F800H :CHECK IF "SS"=1/0 ;TILE_SS 0FA40H ;CHECK IF PPBUSY=1/0
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL MOV CALL MOV CALL MOV CALL	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL DEMO, TRAFILL EXE H MES_CL_WAIT:NEAR, MES_CL_WAIT STATUS:WORD, PDISPSL:WORD, PDI PTNPH:WORD, PTNPL:WORD, X:WORD YS:WORD, YE:WORD, XE:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > SI,9 MES_CL_WAIT BX,703CH TRAFILL_EXE L SI,2 MES_CL_WAIT_FILL BX,70BCH TRAFILL_EXE L SI,3 MES_CL_WAIT_FILL SI,3 MES_CL_WAIT_FILL	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;MESSAGE(9) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1 ;MESSAGE(2) ;<a_trafill> ;TL=1,SS=1,WL=1,WR=1 ;MESSAGE(3) ;<a_trafill></a_trafill></a_trafill></a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL L: SI,OFFSET TRAFILL_DATA L TRAFILL_1 H: SI,OFFSET TRAFILL_DATA_H T DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,0H DS:WORD PTR STATUS,1 TRAFILL_2 DX,0F7D0H TRAFILL_2 DX,0FA30H T ES:WORD PTR STATUS,1 TRAFILL_2 DX,0FA30H T ES:WORD PTR STATUS,1 TRAFILL_2 DX,0FA30H SW ES:WORD PTR PTNPL,DX SW	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMAX=4 ;PTNPH=0 :TILE 0F800H :CHECK IF "SS"=1/0 :TILE_SS 0FA40H :CHECK IF PPBUSY=1/0 :CHECK IF "SS"=1/0
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE_H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PDI PTXPH:WORD, PTNPL:WORD, X:WORD YS:WORD, YE:WORD, XE:WORD, PTN MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > S1,9 MES_CL_WAIT BX,703CH TRAFILL_EXE L S1,2 MES_CL_WAIT_FILL BX,70BCH TRAFILL_EXE L S1,3 MES_CL_WAIT_FILL BX,70ACH TRAFILL_EXE L S1,4	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;MESSAGE(9) ;<a_trafill> :TL=0,SS=1,WL=1,WR=1 :MESSAGE(2) ;<a_trafill> ;TL=1,SS=1,WL=1,WR=1 ;MESSAGE(3) ;<a_trafill> ;TL=1,SS=0,WL=1,WR=1</a_trafill></a_trafill></a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL TRAFILL_1 H: SI,0FFSET TRAFILL_DATA_L TRAFILL_1 DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PTNPL,0X SW ES:WORD PTR STATUS,1 TRAFILL_2 DX,0FA30H T BL,10H TRAFILL_2,1 DX,20H ES:WORD PTR PTNPL,DX SW ES:WORD PTR X,AX SW V ES:WORD PTR Y,AX	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMAX=4 ;PTNPH=0 ;TILE 0F800H :CHECK IF "SS"=1/0 ;TILE_SS 0FA40H ;CHECK IF PPBUSY=1/0 :CHECK IF "SS"=1/0 ;PTNPL=(?)
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL CALL CALL CALL CALL CALL CALL CAL	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL DEMO, TRAFILL EXE H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PDI PTNPH:WORD, PTNPL:WORD, X:WORD MES_CL_WAIT.PILLORD, X:WORD NEAR ZOID FILLING DEMO > SI,9 MES_CL_WAIT_BX,703CH TRAFILL_EXE L SI,3 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI,3 MES_CL_WAIT_FILL BX,70ACH TRAFILL_EXE L SI,4 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI,4 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI,5	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;MESSAGE(9) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1 ;MESSAGE(2) ;<a_trafill> ;TL=1,SS=1,WL=1,WR=1 ;MESSAGE(3) ;<a_trafill> ;TL=1,SS=0,WL=1,WR=1 ;MESSAGE(4) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1</a_trafill></a_trafill></a_trafill></a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL I: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL TRAFILL_1 H: SI,0FFSET TRAFILL_DATA_L TRAFILL_1 DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PTNPH,0 H DS AX.CS DS,AX AX.WORD PTR AGDC_SEG ES,AX CX.10 DX.OFTDOH T BL,10H TRAFILL_2 DX.OFA30H T ES:WORD PTR STATUS,1 TRAFILL_2.1 DX.20H ES:WORD PTR PTNPL,DX SW ES:WORD PTR Y,AX SW	:CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0 ;PMAX=4 ;PTNPH=0 ;TILE 0F800H ;CHECK IF "SS"=1/0 ;TILE_SS 0FA40H ;CHECK IF PPBUSY=1/0 ;CHECK IF "SS"=1/0 ;CHECK IF "SS"=1/0
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE.H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PUINORD, X:UORD YS:WORD, YE:WORD, XE:WORD, PTN MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > SI.9 MES_CL_WAIT BX,703CH TRAFILL_EXE L SI.3 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI.4 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI.4 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI.4 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI.4 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI.4 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI.4 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI.4 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L SI.4 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE L	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;WORD ;MESSAGE(9) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1 ;MESSAGE(2) ;<a_trafill> ;TL=1,SS=1,WL=1,WR=1 ;MESSAGE(3) ;<a_trafill> ;TL=1,SS=0,WL=1,WR=1 ;MESSAGE(4) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1 ;MESSAGE(5) ;<a_trafill></a_trafill></a_trafill></a_trafill></a_trafill></a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL SI,0FFSET TRAFILL_DATA L TRAFILL_1 H: SI,0FFSET TRAFILL_DATA_H T DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PTNPH,0 H DS AX.CS DS,AX AX.WORD PTR AGDC_SEG ES.AX CX,10 DX.OF7D0H T BL,10H TRAFILL_2 DX.OFA30H T ES:WORD PTR STATUS,1 TRAFILL_2 T BL,10H TRAFILL_2 T DL,10H ES:WORD PTR PTNPL,DX SW ES:WORD PTR Y,AX SW ES:WORD PTR Y,AX SW	:CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0 ;PMAX=4 ;PTNPH=0 ;TILE 0F800H ;CHECK IF "SS"=1/0 ;TILE_SS 0FA40H ;CHECK IF PPBUSY=1/0 ;CHECK IF "SS"=1/0 ;CHECK IF "SS"=1/0 ;TILE_?) ; X=(?) ; X=(?) ;Y=(?) ;Y=(?)
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; TRAPE: ; DEMO: MOV CALL	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE_H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PDI PTXPH:WORD, PTNPL:WORD, X:WORD YS:WORD, YE:WORD, XE:WORD, PTN MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > S1.9 MES_CL_WAIT BX.703CH TRAFILL_EXE L S1.2 MES_CL_WAIT_FILL BX.703CH TRAFILL_EXE L S1.4 MES_CL_WAIT_FILL BX.703CH TRAFILL_EXE L S1.4 MES_CL_WAIT_FILL BX.703CH TRAFILL_EXE L S1.5 MES_CL_WAIT_FILL BX.703CH TRAFILL_EXE_H S1.5 MES_CL_WAIT_FILL BX.703CH TRAFILL_EXE_H S1.5 MES_CL_WAIT_FILL BX.703CH TRAFILL_EXE_H S1.5 MES_CL_WAIT_FILL BX.703CH TRAFILL_EXE_H S1.5 MES_CL_WAIT_FILL BX.703CH TRAFILL_EXE_H	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;WORD ;A_TRAFILL> ;TL=0,SS=1,WL=1,WR=1 ;MESSAGE(2) ;<a_trafill> ;TL=1,SS=1,VL=1,WR=1 ;MESSAGE(3) ;<a_trafill> ;TL=1,SS=0,VL=1,WR=1 ;MESSAGE(4) ;<a_trafill> ;TL=0,SS=1,VL=1,WR=1 ;MESSAGE(5)</a_trafill></a_trafill></a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 L MES_CL_WAIT FILL L: SI,OFFSET TRAFILL_DATA L TRAFILL_1 H: SI,OFFSET TRAFILL_DATA_H T DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSH,0 DS:WORD PTR STATUS,1 TRAFILL_2 DX,0FA30H T ES:WORD PTR STATUS,1 TRAFILL_2 DX,10H ES:WORD PTR Y,AX SW ES:WORD PTR Y,AX SW ES:WORD PTR Y,AX SW	<pre>:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMAX=4 ;PTNPH=0 ;TILE 0F800H :CHECK IF "SS"=1/0 ;TILE_SS 0FA40H ;CHECK IF PPBUSY=1/0 ;CHECK IF PPBUSY=1/0 ;CHECK IF "SS"=1/0 ;PTNPL=(?) ; X=(?) ; Y=(?) ;YS=(?) ;YE=(?)</pre>
PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE: ; DEMO: MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL MOV	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL DEMO, TRAFILL EXE H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PDI PTNPH:WORD, PTNPL:WORD, X:WORD VS:WORD, VE:WORD, XE:WORD, PTN MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO > SI,9 MES_CL_WAIT_ BX,703CH TRAFILL_EXE L SI,3 MES_CL_WAIT_FILL BX,70ACH TRAFILL_EXE L SI,3 MES_CL_WAIT_FILL BX,70ACH TRAFILL_EXE L SI,3 MES_CL_WAIT_FILL BX,70ACH TRAFILL_EXE L SI,4 MES_CL_WAIT_FILL BX,70BCH TRAFILL_EXE H SI,5 MES_CL_WAIT_FILL BX,70BCH	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;WORD ;MESSAGE(9) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1 ;MESSAGE(2) ;<a_trafill> ;TL=1,SS=1,WL=1,WR=1 ;MESSAGE(3) ;<a_trafill> ;TL=1,SS=0,WL=1,WR=1 ;MESSAGE(4) ;<a_trafill> ;TL=0,SS=1,WL=1,WR=1 ;MESSAGE(5) ;<a_trafill></a_trafill></a_trafill></a_trafill></a_trafill></a_trafill></pre>	MOV CALI RET ; TRAFILL EXE MOV JMP TRAFILL EXE MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	L TRAFILL EXE H SI,7 MES_CL_WAIT FILL TRAFILL_I H: SI,0FFSET TRAFILL_DATA L TRAFILL_I DS:WORD PTR STATUS,1 TRAFILL_I DS:WORD PTR PDISPSL,10H DS:WORD PTR PTNPH,0 H DS:WORD PTR PTNPH,0 H SS AX.CS DS,AX AX.WORD PTR AGDC_SEG ES.AX CX,10 DX,0F7D0H T BL,10H TRAFILL_2 DX,0F7D0H T BL,10H TRAFILL_2 T BL,10H TRAFILL_2 T BL,10H ES:WORD PTR STATUS,1 TRAFILL_2 DX,0FA30H SW ES:WORD PTR Y,AX SW ES:WORD PTR Y,AX SW ES:WORD PTR Y,AX SW ES:WORD PTR YS,AX SW ES:WORD PTR YE,AX SW	:CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0 ;PMAX=4 ;PTNPH=0 ;TILE 0F800H ;CHECK IF "SS"=1/0 ;TILE_SS 0FA40H ;CHECK IF PPBUSY=1/0 ;CHECK IF "SS"=1/0 ;CHECK IF "SS"=1/0 ;TILE_?) ; X=(?) ; X=(?) ;Y=(?) ;Y=(?)

	MOV MOV	ES:WORD PTR PTNCNT,AX AX,CX	;PTNCNT=0
	AND JNZ MOV	AX,7 TRAFILL_3 AX,3	;CHECK IF "WHITE"
TRAFILL	_3: MOV MOV JMP	ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 TRAFILL_TL	;PLANES=(?) ;MOD1=0,MOD0=1
TRAFILL	_TL1: MOV MOV MOV	ES:WORD PTR PTNCNT,16 ES:WORD PTR PLANES,7 ES:WORD PTR MODIO,0	;PTNCNT=16 ;PLANES=7 ;MOD1=0,MOD0=0
TRAFILL	_TL: MOV DEC JNZ POP RET	ES:WORD PTR COM,BX CX TRAFILL_2 DS	; <com.flags></com.flags>
TRAFILL	, DATA_L: DW DW DW DW DW DW DW DW DW DW	0000H,0096H,012CH,001EH,0000H, 0091H,00AFH,0172H,0064H,0019H,0 0122H,00C8H,01B8H,00AAH,0032H,0 0128H,00E1H,01FEH,00FOH,0048H,0 0244H,00FAH,01FEH,0178H,016H,0 0153H,014AH,027FH,0171H,010FH,0 0158H,0131H,01EEH,012BH,0166H,0 0150H,0118H,0150H,00E5H,01ADH,0 0162H,00FFH,00CCH,009FH,0194H,0 0167H,00E6H,003BH,0059H,017BH,0	154H; 2. X, Y, XS, YS, YE, XE 19AH; 3. X, Y, XS, YS, YE, XE 1E0H; 4. X, Y, XS, YS, YE, XE 26H; 5. X, Y, XS, YS, YE, XE 26H; 6. X, Y, XS, YS, YE, XE 21BH; 7. X, Y, XS, YS, YE, XE 18FH; 9. X, Y, XS, YS, YE, XE
TRAFILL	; _DATA .H: DW DW DW DW DW DW DW DW DW DW ;	0000H,00EAH,020DH,0035H,0000H,0 00FEH,0111H,0288H,00B0H,0027H,0 01FCH,0138H,0303H,012BH,004EH,0 02FAH,015FH,037EH,01AGH,0075H,0 03F8H,0186H,03F9H,0221H,009CH,0 0251H,01EAH,045EH,0286H,02CCH,0 025AH,01B0H,0360H,020CH,02C5H,0 0263H,0196H,0262H,0192H,029EH,0 026CH,016FH,0164H,0118H,0277H,0 0275H,0148H,0066H,009EH,0250H,0	2254H; 2. X, Y, XS, YS, YE, XE 2CFH; 3. X, Y, XS, YS, YE, XE 34AH; 4. X, Y, XS, YS, YE, XE 305H; 5. X, YXS, YS, YE, XE 42AH; 6. X, Y, XS, YS, YE, XE 380H; 7. X, Y, XS, YS, YE, XE 336H; 8. X, Y, XS, YS, YE, XE 20CH; 9. X, Y, XS, YS, YE, XE
; TRAFILL PROG			
	CALL	-3-	
	MOV	SI,4 MES_CL_WAIT_FILL	;MESSAGE(4)
	MOV MOV MOV	CS:WORD PTR WORK_1,560 CS:WORD PTR WORK_2,374 BX,6C3CH	;X=560 ;Y=374 ; <a_trifill> ;TL=0,SS=1,WL=1,WR=1</a_trifill>
	CALL MOV CALL	TRIFILL_EXE_1 SI,5 MES_CL_WAIT_FILL	;MESSAGE(5)
	MOV	BX,6CBCH	; <a_trifill> ;TL=1,SS=1,WL=1,WR=1</a_trifill>
	CALL MOV CALL	TRIFILL_EXE_1 SI,6 MES_CL_WAIT_FILL	;MESSAGE(6)
	; MOV	BX,6CACH	; <a_trifill> ;TL=1,SS=0,WL=1,WR=1</a_trifill>
	CALL MOV CALL	TRIFILL_EXE_1 SI,7 MES_CL_WAIT_FILL	;MESSAGE(7)
	, RET ;		
FRIFILL <u>.</u> Frifill	MOV EXE1:	DI,1	;DRAW COUNTS
	CALL DEC JNZ RET	TRIFILL_EXE_L DI TRIFILL_EXE1	
FRIFILL .	MOV	DI,1	;DRAW COUNTS
TRIFILL .	FYF 11.		
	CALL DEC JNZ	TRIFILL_EXE_H DI TRIFILL_EXE_11	
TRIFILL	CALL DEC JNZ RET ; EXE L: MOV	DI TRIFILL_EXE_11 SI,OFFSET TRIFILL_DATA_L	
TRIFILL_	CALL DEC JNZ RET ; EXE L: MOV JMP EXE_H: MOV	DI TRIFILL_EXE_11	
	CALL DEC JNZ RET ; EXE L: MOV JMP EXE_H: MOV	DI TRIFILL_EXE_11 SI,OFFSET TRIFILL_DATA_L TRIFILL_1	

```
NAME
                   TRIFILL
          GROUP PROG
SEGMENT BYTE PUBLIC
 PGROUP
                                      'PROG'
 PROG
          ASSUME CS:PGROUP
          PUBLIC TRIFILL DEMO
          FXTRN
                   DMCLO_C_WAIT:NEAR, MES_CL_WAIT:NEAR
                   MES_CL_WAIT_FILL:NEAR
          EXTRN
          EXTRN
                   STATUS: WORD, PDISPSL: WORD, PDISPSH: WORD, PMAX: WORD
                   PTNPH:WORD, WORK_I:WORD, WORK_2:WORD, PTNPL:WORD
X:WORD, Y:WORD, XC:WORD, YC:WORD, XS:WORD, YS:WORD
PTNCNT:WORD, PLANES:WORD, MODIO:WORD, COM:WORD
          EXTRN
          EXTRN
          EXTRN
          EXTRN
                   AGDC_SEG:WORD
 TRIFILL PROC
                   NEAR
          < TRIANGLE FILLING DEMO >
   [TRIFILL_DEMO]
 ;;;
 TRIFILL DEMO:
          MOV
                   $1,8
          CALL
                   MES_CL_WAIT
                                                        :MESSAGE(8)
 TRIFILL_DEM0_1:
                   DS:WORD PTR STATUS,1
          TEST
          JNZ
                   TRIFILL DEMO_1
                                                        ;CHECK IF PPBUSY=1/0
                   DS:WORD PTR PDISPSL,10H
DS:WORD PTR PDISPSH,0
                                                        :PDISPSL=10H
          MOV
                                                        ;PDISPSH=0
          MOV
          MOV
                   DS:WORD PTR PMAX,4
DS:WORD PTR PTNPH,0
                                                        ;PMAX=4
                                                        :PTNPH=0
          MOV
                   CS:WORD PTR WORK_1,320
                                                        ;X=320
          MOV
          MOV
                   CS:WORD PTR WORK_2,240
                                                        ;Y=240
                                                        ;<A TRIFILL>
          MOV
                   BX,6C3CH
                                                        ;TL=0,SS=1,WL=1,WR=1
          CALL
                   TRIFILL EXE
          MOV
                   $1,2
          CALL
                   MES_CL_WAIT_FILL
                                                        ;MESSAGE(2)
          MOV
                   BX,6CBCH
                                                        ;<A_TRIFILL>
                                                        ;TL=1,SS=1,WL=1,WR=1
          CALL
                   TRIFILL_EXE
          MOV
                   $1,3
                   MES_CL_WAIT_FILL
                                                        :MESSAGE(3)
          CALL
          MOV
                   BX.6CACH
                                                        ;<A_TRIFILL>
                                                        ;TL=1,SS=0,WL=1,WR=1
                                      -1-
         MOV
                  AX, WORD PTR AGDC_SEG
         MOV
                   ES.AX
         MOV
                   CX,12
                                                       REPETITON COUNTS
                                                       ;TILE .. 0F800H
         MOV
                  DX,OF7DOH
         TEST
                  BL,10H
                                                       ;CHECK IF "SS"=1/0
         JZ
                   TRIFILL_2
MOV
TRIFILL_2:
                                                       ;TILE_SS .. OFA40H
                  DX, OFA30H
                  ES:WORD PTR STATUS,1
                                                       ;CHECK IF PPBUSY=1/0
         INZ
                   TRIFILL 2
         TEST
                   BL,10H
                                                       ;CHECK IF "SS"=1/0
         JNZ
                   TRIFILL_2_1
         ADD
                  DX.20H
TRIFILL_2_1:
                  DX.10H
         ADD
         MOV
                   ES:WORD PTR PTNPL,DX
                                                       ;PTNPL=(?)
                  AX,CS:WORD PTR WORK_1
ES:WORD PTR X,AX
         MOV
                                                       ;X=(?)
         MOV
         MOV
                   AX,CS:WORD PTR WORK_2
                                                       ;Y=(?)
         MOV
                   ES:WORD PTR Y,AX
         LODSW
                   ES:WORD PTR XC,AX
                                                       ;XC=(?)
         MOV
         LODSU
                   ES:WORD PTR YC,AX
                                                       ;YC=(?)
         MOV
         LODSW
                   ES:WORD PTR XS,AX
                                                       ;XS=(?)
         MOV
         LODSW
                   ES:WORD PTR YS.AX
                                                       ;YS=(?)
         MOV
         TEST
                   BL..80H
                                                       ;CHECK IF "TL"=1/0
                   TRIFILL_TL1
         JNZ
         MOV
                   AX.O
         MOV
                   ES:WORD PTR PTNCNT,AX
                                                       ;PTNCNT=0
         MOV
                  AX,CX
         AND
                   AX.7
                   TRIFILL_3
                                                       ;CHECK IF "WHITE"
         JNZ
         MOV
                   AX,3
TRIFILL_3:
         MOV
                   ES:WORD PTR PLANES,AX
ES:WORD PTR MODIO,1
                                                       ;PLANES=(?)
                                                       ;MOD1=0,MOD0=1
         MOV
                   TRIFILL_TL
         JMP
TRIFILL_TL1:
         MOV
                   ES:WORD PTR PTNCNT,16
                                                       ;PTNCNT=16
         MOV
                   ES:WORD PTR PLANES.7
                                                       :PLANES=7
                   ES:WORD PTR MODIO,0
                                                       ;MOD1=0,MOD0=0
         MOV
TRIFILL_TL:
                  ES:WORD PTR COM, BX
                                                       :<COM.FLAGS>
         MOV
         DEC
                   CX
         JNZ
POP
                   TRIFILL_2
                  DS
         RET
```

-2-

; TRIFILL DATA L: DW DW DW DW DW DW DW DW DW DW DW DW DW	01CBH,0140H,0098H,01A3H 0190H,0178H,0056H,0137H 0140H,0190H,0051H,0088H 00F0H,0178H,0080H,0048H 0085H,0140H,00F9H,0066H 00A0H,00F0H,0178H,0001H 00F0H,0065H,022H,0039H 0140H,0050H,022H,0128H 0140H,0050H,01F3H,0188H 0140H,0050H,01F3H,0188H 0126H,0050H,01F3H,018H 0126H,00F0H,0108H,01DFH 0315H,01FAH,011BH,029EH 0285H,0256H,00ACH,01ECH 0230H,027FH,00A6H,011AH 01ACH,0258H,0108H,0061H 01ACH,0258H,018AH,09FF2H 012FH,01F6H,018AH,07FF2H 012FH,0176H,028CH,07FECH 0148H,0091H,038AH,012H 0284H,0091H,038AH,012H 0315H,00F1H,02A6H,02FAH 0339H,0176H,01D4H,0300H	<pre>: 1. X,Y,DX,DY : 2. X,Y,DX,DY : 3. X,Y,DX,DY : 4. X,Y,DX,DY : 5. X,Y,DX,DY : 6. X,Y,DX,DY : 6. X,Y,DX,DY : 7. X,Y,DX,DY : 8. X,Y,DX,DY : 9. X,Y,DX,DY : 10. X,Y,DX,DY : 11. X,Y,DX,DY : 12. X,Y,DX,DY : 12. X,Y,DX,DY : 3. X,Y,DX,DY : 5. X,Y,DX,DY : 5. X,Y,DX,DY : 6. X,Y,DX,DY : 6. X,Y,DX,DY : 7. X,Y,DX,DY : 8. X,Y,DX,DY : 8. X,Y,DX,DY : 9. X,Y,DX,DY : 10. X,Y,DX,DY : 11. X,Y,DX,DY : 12. X,Y,DX,DY : 12. X,Y,DX,DY : 12. X,Y,DX,DY : 12. X,Y,DX,DY : 12. X,Y,DX,DY : 12. X,Y,DX,DY</pre>	PGROUP PROG PAINT ; ; ; ; PAINT_C PAINT_C	ASSUME ; PUBLIC EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; CALL MOV MOV CALL ;	PAINT PROG BYTE PUBLIC 'PROG' CS:PGROUP PAINT_DEMO SHRINK_PUT:NEAR, SHRINK_EXE:NE STATUS:WORD, XCLMIN:WORD, YCLM YCLMAX:WORD, MAGETC:WORD, YCLM YCLMAX:WORD, MAGETC:WORD, YCLM YCLMAX:WORD, MAGETC:WORD, YCLM POISPSH:WORD, PDISPSL:WORD, YCLM PUANES:WORD, PDISPSL:WORD, YCLM NEAR DEMO > SHRINK_PUT SI,OFFSET SHRINK DATA CX,6 SHRINK_EXE DS:WORD PTR STATUS,1 PAINT_DEMO_1 DS:WORD PTR XCLMIN,5	IN:WORD, XCLMAX:WORD :WORD, PITCHS:WORD NCNT:WORD, MODIO:WORD
END	-4-		PAINT_I	MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	DS:WORD PTR YCLMIN,80 DS:WORD PTR YCLMAX,7591 DS:WORD PTR YCLMAX,744 DS:WORD PTR MACREC.0FFH DS:WORD PTR PMAX,4 DS:WORD PTR PDISPSH,1 DS:WORD PTR PDISPSH,1 DS:WORD PTR PDISPSL,0 DS:WORD PTR PDISPSL,0 DS:WORD PTR PTNCNT,0FFFFH DS:WORD PTR MODIO,1 DS:WORD PTR MODIO,1 DS:WORD PTR Y,560 DS:WORD PTR Y,560 DS:WORD PTR Y,560 DS:WORD PTR STATUS,1 PAINT_DEM0_2 -1-	;YCLHNN=80 ;YCLMN=80 ;XCLMAX=591 ;YCLMAX=744 ;PITCHS=46H ;PDISPSL=0 ;PTTCHS=46H ;PDISPSL=0 ;PTNCNT=0FFFFH ;MOD1=0,MOD0=1 ;PLANES=6 ;X=300 ;Y=560 ; <paint> ;PMOD=1, TL=0, SS=1 ;CHECK IF PPBUSY=1/0</paint>
MOV MOV MOV MOV MOV MOV MOV PAINT_DEMO_3: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	DS:WORD PTR XCLMIN,596 DS:WORD PTR YCLMIN,80 DS:WORD PTR YCLMIN,80 DS:WORD PTR YCLMAX,1115 DS:WORD PTR YLANES,3 DS:WORD PTR YLANES,3 DS:WORD PTR Y,560 DS:WORD PTR Y,560 DS:WORD PTR Y,560 DS:WORD PTR Y,560 DS:WORD PTR YCMIN,0 DS:WORD PTR YCLMIN,0 DS:WORD PTR YCLMIN,30 DS:WORD PTR YCLMIN,40 DS:WORD PTR YCLMIN,40 DS:WORD PTR YCLMIN,40 DS:WORD PTR YCLMAX,749 DS:WORD PTR YDISPSH,0 DS:WORD PTR PDISPSH,16 DS:WORD PTR PDISPSH,16 DS:WORD PTR PTNCH,16 DS:WORD PTR PTNCH,16 DS:WORD PTR PTNCH,0F800H DS:WORD PTR YNPL,0F800H DS:WORD PTR YNPL,0F800H DS:WORD PTR Y,40 DS:WORD PTR YCM,68A4H	:XCLMIN=596 ;YCLMIN=80 ;XCLMAX=1115 ;YCLMAX=1115 ;YCLMAX=744 ;PLANES=3 ;X=800 ;Y=560 ; <paint> ;PM0D=1, TL=0, SS=1 ;CHECK IF PPBUSY=1/0 ;XCLMIN=0 ;YCLMIN=30 ;XCLMAX=119 ;YCLMAX=749 ;PDISPSH=0 ;PDISPSH=0 ;PDISPSH=16 ;PTNPH=0 ;PTNCNT=16 ;PTNPH=0 ;PTNCNT=18 ;PLANES=7 ;MOD1-0,MOD0=1 ;PTNPL=0F800H TILE(0) ;X=600 ;Y=40 ;<paint> ;PM0D=1, TL=1, SS=0</paint></paint>	PAINT_D ; PAINT PROG	MOV INC NOV MOV INC INC INC MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AX,CS:[S1] S1 S1 S1 S1 S1 S1 S1 S1 S1 S1	;X=(?) ;PTNCNT=16 ;PTNPL+30H> PTNPL ;PTNPL=(?) ; <com.flags> ;16/16 ;15/16 ;13/16 ;12/16 ;11/16 ;10/16</com.flags>
, MOV MOV CALL MOV CALL PAINT_DEMO_4: TEST JNZ MOV MOV MOV MOV MOV RET ; PAINT_NEC: MOV PAINT_NEC: TEST	SI, OFFSET PAINT_DATA BX, 68A4H CX, 12 PAINT_NEC CX, 9 PAINT_NEC DS: WORD PTR STATUS, 1 PAINT_DEMO_4 DS: WORD PTR YCLMIN, 0 DS: WORD PTR YCLMIN, 0 DS: WORD PTR YCLMIX, 749 DS: WORD PTR YCLMAX, 749 DS: WORD PTR MAGETC, 1FFH DX, OF7DOH DS: WORD PTR STATUS, 1	; <paint> ;PM0D=1, TL=1, SS=0 ;REPETITION COUNTS ;REPETITION COUNTS ;CHECK IF PPBUSY=1/0 ;XCLMIN=0 ;YCLMIN=0 ;YCLMIN=0 ;YCLMAX=749 ;MAGETC=1FFH</paint>				

PGROUP PROG	SEGMENT	SCROLL PROG BYTE PUBLIC 'PROG' CS:PGROUP			CALL CMP JB CALL	SCROLL_P3 CX,166 SCROLL_SKIP SCROLL_P4	
	;	SCROLL_DEMO		SCROLL_		SCROLL_DEM0_2	
	; EXTRN	CHR_DRAW_DEMO_H:NEAR			RET;		
	; EXTRN EXTRN EXTRN EXTRN EXTRN	WORK_1:WORD, WORK_2:WORD, WORK_ STATUS:WORD, PLANES:WORD, PMAX: MODIO:WORD, XS:WORD, YS:WORD, X DHH:WORD, DV:WORD, COM:WORD, PD PDISPSL:WORD	WORD, PITCHS:WORD (:WORD, Y:WORD	SCROLL	TEST JNZ MOV MOV MOV	DS:WORD PTR STATUS,1 SCROLL_P1 DS:WORD PTR XS,AX DS:WORD PTR YS,BX DS:WORD PTR X,16	;CHECK IF PPBUSY=1/0 ;XS=(?) ;YS=(?) ;X=16
SCROLL	; PROC	NEAR			MOV	DS:WORD PTR Y,734 DS:WORD PTR DHH,255	; Y=734 ;DH=255 ;DV=255
;	; < SCROL	L DEMO >			MOV MOV	DS:WORD PTR DV,255 DS:WORD PTR COM,840EH	; <a_copy_cc> COPY ;ESE=0,REV=0,ROT=0 ;{MSD},FAST=1</a_copy_cc>
;	;				ADD RET	AX,1	;XS+1
SCROLL	CALL MOV	CHR_DRAW_DEMO_H AX,0	;END OF BACK-GROUND ;X1=0	SCROLL_	; P2: TEST JNZ	DS:WORD PTR STATUS,1 SCROLL_P2	;CHECK IF PPBUSY=1/0
SCROLL_	MOV MOV MOV MOV MOV MOV MOV	AX,0 BX,260 S1,560 D1,477 CS:WORD PTR WORK_1,0 CS:WORD PTR WORK_2,256 CS:WORD PTR WORK_3,560 CS:WORD PTR WORK_4,477 CX,220	;X1=260 ;X2=560 ;Y2=477 ;X3=0 ;Y3=256 ;X4=560 ;Y4=477 ;REPETITION COUNTS		MOV MOV MOV MOV MOV MOV MOV	DS:VORD PTR XS,SI DS:VORD PTR YS,DI DS:VORD PTR YS,DI DS:VORD PTR Y,288 DS:VORD PTR V,734 DS:VORD PTR DH1,255 DS:VORD PTR DV,255 DS:VORD PTR COM,840EH	;XS=(?) ;YS=(?) ;Y=288 ;Y=734 ;DH=255 ;DV=255 ; <a_copy_cc> COPY ;ESE=0,REV=0,ROT=0 ;(MSD),FAST=1</a_copy_cc>
SCROEL_	TEST JNZ	DS:WORD PTR STATUS,1 SCROLL_DEMO_1	;CHECK IF PPBUSY=1/0		SUB Ret	D1,2	;YS-2
SCROLL_	MOV MOV MOV MOV MOV ; DEMO_2:	DS:WORD PTR PDISPSH,1 DS:WORD PTR PDISPSL,0 DS:WORD PTR PLANES,7 DS:WORD PTR PHAX,4 DS:WORD PTR PITCHS,46H DS:WORD PTR MOD10,11H	;PDISPSH=1 ;PDISPSL=0 ;PLANES=7 ;PMAX=4 ;PITCHS=46H ;MOD10=11H	SCROLL_	; TEST JNZ MOV MOV MOV MOV MOV	DS:WORD PTR STATUS,1 SCROLL_P3 DX.CS:WORD PTR WORK_1 DS:WORD PTR XS,DX DX.CS:WORD PTR WORK_2 DS:WORD PTR YS,DX	;CHECK IF PPBUSY=1/0 ;XS=(?) ;YS=(?)
	CALL CMP JB CALL CMP JB	SCROLL_P1 CX,110 SCROLL_SKIP SCROLL_P2 CX,147 SCROLL_SKIP			MOV MOV MOV MOV MOV	DS:WORD PTR X,560 DS:WORD PTR Y,734 DS:WORD PTR DHH.255 DS:WORD PTR DV,255 DS:WORD PTR COM,840CH	; X=560 ; Y=734 ;DH=255 ;DV=255 ; <a_copy_cc> COPY ;ESE=0,REV=0,ROT=0</a_copy_cc>
		-1-				-2-	
	ADD ADD RET	CS:WORD PTR WORK_1,3 CS:WORD PTR WORK_2,3	:{MSD},FAST=0 <1!BUG!!> :XS+3 :YS+3	PGROUP PROG		FRCOPY PROG BYTE PUBLIC 'PROG' CS:PGROUP	
SCROLL_	; P4: TEST	DS:WORD PTR STATUS,1			PUBLIC	FRCOPY_DEMO	
	JNZ MOV	SCROLL_P4 DX,CS:WORD PTR WORK_3	;CHECK IF PPBUSY=1/0		, Extrn ;	PUT_DATA_TRAN:NEAR	
	MOV MOV MOV MOV MOV	DS:WORD PTR XS,DX DX,CS:WORD PTR WORK_4 DS:WORD PTR YS,DX DS:WORD PTR X,832 DS:WORD PTR X,734	;XS=(?) ;YS=(?) ; X=832 ; Y=734		EXTRN EXTRN EXTRN EXTRN	STATUS:WORD, DHH:WORD, DV:WORD, PLANES:WORD, PMAX:WORD, MODIO:W PITCHS:WORD, XS:WORD, YS:WORD, XE:WORD, YE:WORD, MAGETC:WORD	ORD, COM:WORD
	MOV MOV	DS:WORD PTR DHH,255 DS:WORD PTR DV,255	;DH=255 ;DV=255	FRCOPY	PROC	NEAR	
	MOV	DS:WORD PTR COM,840CH	<pre><</pre>	;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;	, < FR_CO	PY DEMO >	
; Scroll Prog	ENDP ENDS END	CS:WORD PTR WORK_4,4	;YS-4	FRCOPY	; DEMO: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV	DS:WORD PTR STATUS,1 FRCOPY_DEMO DS:WORD PTR DHH,255 DS:WORD PTR DV,255 DS:WORD PTR Y,720 DS:WORD PTR Y,720 DS:WORD PTR PLANES,7 DS:WORD PTR PLANES,7 DS:WORD PTR PMAX,4 DS:WORD PTR MOD10,0 DS:WORD PTR COM,980FH	:CHECK IF PPBUSY=1/0 :DH=255 :X=30 :Y=720 :PLANES=7 :PMAX=4 :M0D1=0,M0D0=0 : <put_c> :R0T=0.{MSD}</put_c>
					PUSH MOV MOV CALL POP	DS AX,6COOH DS,AX CX.3000H PUT_DATA_TRAN DS	;SEGMENT "MAAJAN" ;256 X 256 X 3 PLANES
				FRCOPY	MOV MOV DEMO_1: TEST	SI,OFFSET FRCOPY_DATA CX,16 DS:WORD PTR STATUS,1	;END OF "PUT" ;REPETITION COUNTS

MOV MOV INC INC MOV	DS:WORD PTR DV,255 AX,CS:[SI] SI SI DS:WORD PTR DXX,AX	;DV=255 ;DXX=(?)	PGROUP PROG		MESSAGE PROG BYTE PUBLIC 'PROG' CS:PGROUP	
MOV	AX,CS:[SI] SI	, UAA-(.)		, PUBLIC	MESSAGE	
INC	SI DS:WORD PTR DY,AX	;DY=(?)		ÉXTRN	WAIT:NEAR	
MOV INC INC MOV	AX,CS:[SI] SI SI DS:WORD PTR XE,AX	;xE=(?)		EXTRN EXTRN EXTRN EXTRN	STATUS:WORD, PMAX:WORD, PITCHS DV:WORD, MODIO:WORD, PLANES:WO EAD2H:WORD, dAD2:BYTE, X:WORD, FLAG:BYTE, DXX:WORD	RD, EAD2L:WORD
MOV INC INC	AX,CS:[SI] SI SI		MESSAGE	PROC	NEAR	
MOV MOV INC	DS:WORD PTR YE,AX AX,CS:[SI] SI	;YE=(?)	;	, < DRAW	PREVIOUSLY ASSIGNED MESSAGE >	
INC	SI SI DS:WORD PTR MAGETC,AX	;MAGETC=(?)	, ;PRINT	MESSAGE :	ON PREDETERMINED PAGE (SI; O T	0 (?))
MOV LOOP RET	DS:WORD PTR COM,843EH FRCOPY_DEMO_1	;<4_COPY_CC> FR_COPY ;ROT=0,{MSD}		, TEST JNZ MOV MOV	DS:WORD PTR STATUS,1 MESSAGE DS:WORD PTR PMAX,4 DS:WORD PTR PITCHS,2	;CHECK IF PPBUSY=1/0 ;PMAX=4 ;PITCHS=2
; FRCOPY_DATA: DW DW DW DW DW DW DW DW DW DW DW DW DW	2, 1, 1, -2, 1DDH 2, 2, 2, -2, 1AAH 1, 2, 2, -1, 1DDH -1, 2, 2, 0, 1FFH -1, 2, 2, 1, 1DDH -2, 2, 2, 1, 1DDH -2, 1, 1, 2, 1DDH -2, -1, -1, 2, 1DDH -2, -2, 2, 1AAH -1, -2, -2, 1, 1DDH 0, -2, -2, 0, 1FFH 1, -2, -2, -1, 1DDH 2, -2, -2, 1DDH 2, -2, -2, 1AAH -1, -2, -2, 1DDH 2, -2, -2, 1AAH -1, -2, -2, 1DDH 2, -2, -2, 1FFH 2, 0, 0, -2, 1FFH	:FR(0) ;FR(1) :FR(2) :FR(3) :FR(3) :FR(5) :FR(6) :FR(6) :FR(7) :FR(8) :FR(9) :FR(9) :FR(9) :FR(0) :FR(C) :FR(C) :FR(C) :FR(F)	MES_2:	MOV MOV MOV PUSH MOV MOV MOV MOV MOV ADD MOV MOV MOV MOV MOV MOV	DS:WORD PTR DHH,23 DS:WORD PTR DV,23 DS:WORD PTR DV,23 DS:WORD PTR PLANES,6 DS AX,DS ES,AX BX,2 CX,746 AX,54FOH DS,AX SI,SI SI,[SI] AX,5500H DS,AX AL.[SI] AL,80H	:DH=23 :DV=23 :MOD1=9.MOD0=7 :PLANES=6 :X=2 :Y=746 :PAGE ADDRESS SEGMENT :MESSAGE SEGMENT
;	2, 0, 0, 2, 1777	, , , , , , , , , , , , , , , , , , , ,		J NC CMP	MES_1 AL,ODH	;CHECK IF 2 BYTES CODE
FRCOPY ENDP PROG ENDS				JZ ;	MES_SKIP2	;CHECK IF "CR/LF"
END			;COLOR	ATTRIBUT		
				MOV CMP	DX,1 AL,4DH	
	-2-				-1-	
JZ	-2- Chg_color	;"M"=MAGENTA	S.J.IS .J.IS	5 3:	-1-	
JZ INC CMP JZ		;"M"=MAGENTA ;"C"=CYAN	SJIS_JIS ; ADJUST	SUB ;	-1- AL,1FH 5 TO BOARD HARDWARE	
INC CMP JZ INC CMP	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H	;"C"=CYAN		SUB ; JIS CODE ; MOV	AL,1FH E TO BOARD HARDWARE DX,1	
INC CMP JZ INC CMP JZ INC CMP	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H			SUB ; JIS CODE ; MOV SHL SHL SHL	AL,IFH E TO BOARD HARDWARE DX,I AL,I AX,I AX,I	
INC CMP JZ INC CMP JZ CMP JZ INC CMP	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW		SUB ; JIS CODE ; MOV SHL SHL SHL SHL RCL SHL RCL	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 AX, 1 DX, 1 DX, 1	
INC CMP JZ INC CMP JZ INC CMP JZ INC	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,52H	;"C"=CYAN ;"B"=BLUE ;"Y"=VELLOW ;"G"=GREEN		SUB ; JIS CODE ; MOV SHL SHL SHL SHL SHL SHL SHL	AL, IFH E TO BOARD HARDWARE DX, I AL, I AX, I DX, I DX, I AX, I	
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,52H CHG_COLOR DX AL,52H CHG_COLOR DX	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW		SUB ; JIS CODA ; MOV SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL ;	AL, 1FH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 DX, 1	
INC CMP JZ INC C C CMP JZ INC C C CMP JZ INC C C CMP JZ C C C C C C C C C C C C C C C C C C	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,52H CHG_COLOR	;"C"=CYAN ;"B"=BLUE ;"Y"=VELLOW ;"G"=GREEN		SUB ; JIS CODE ; MOV SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL ; POP MOV MOV MOV	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 SUBJECT FR EAD2L, AX DS: WORD PTR EAD2L, DX DS: BYTE PTR EAD2L, DX DS: WORD PTR X, BX	:EAD2L=(?) ;EAD2H=(?) ;AAD2=0 ;X=(?)
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JC CMP JC CMP JC CMP JC CMP JC CMP JC CMP JC CMP JC CMP	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,4CH CHG_COLOR	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"C"=GREEN ;"R"=RED		SUB ; JIS CODE ; MOV SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL MOV MOV	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 DX, 1 DX, 1 DX, 1 DX, 1 SUBJECT PTR EAD2L, AX DS: BYTE PTR dAD2, 0	;EAD2H=(?) ;AAD2=0 ;X=(?) ;Y=(?) ;X=X+28 ; <a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD}</a_copy_ac>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ CMP INC C CMP INC C C CMP INC C C C C C C C C C C C C C C C C C C	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,4CH CHG_COLOR DS ES:WORD PTR PLANES,DX S1 MES_2	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN :"R"=RED ;"L"=BLACK		SUB ; JIS CODE ; MOV SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL SHL SHL SHL SHL SHL SHL SHL SHL SH	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 DX, 1 AX, 1 DX, 1 DX	;EAD2H=(?) ;dAD2=0 ;x=(?) ;y=(?) ;x=x+28 ; <a_copy_ac> COPY</a_copy_ac>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ CMP JZ CMP INC CMP INC CMP INC CMP SUB SUB SUB SUB SUB SUB SUB SUB	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,42H CHG_COLOR DX SI MES_2 BX,2 CX,30 SI	:"C"=CYAN :"B"=BLUE :"Y"=YELLOW :"G"=GREEN :"R"=RED :"L"=BLACK :PLANES=(?)		SUB ; JIS CODE ; MOV SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL SHL SHL SHL SHL SHL SHL SHL SHL SH	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 DX, 1 SUBJECT FOR CAD2L, AX DS: WORD PTR EAD2L, AX DS: WORD PTR EAD2L, AX DS: WORD PTR EAD2L, AX DS: WORD PTR Y, CAD2, 0 DS: WORD PTR Y, CX BX, 28 AX, 8008H	;EAD2H=(?) ;AdD2=0 ;X=(?) ;Y=(?) ;X=X+28 ;A_COPY_AC> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 <<11BUG11>> ;CHECK IF "SLANT"=1/0 ;A_COPY_AC> COPY ;ESE=0,REV=0,ROT=0,{MD}
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ CMP JZ CMP JZ CMP JZ CMP JZ CMP JZ CMP JZ CMP JZ CMP SUB	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,4CH CHG_COLOR DS ES:WORD PTR PLANES,DX SI MES_2 BX,2 CX,30	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN ;"R"=RED ;"L"=BLACK ;PLANES=(?) ;X=2	; ADJUST	SUB ; JIS CODE ; MOV SHL SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL ; POP MOV MOV MOV MOV MOV TEST JZ	AL. IFH E TO BOARD HARDWARE DX.1 AL.1 AX.1 DX.1 AX.1 DX.1 AX.1 DX.1 AX.1 DX.1 DX.1 DX.1 DX.1 DX.1 DX.1 DX.1 D	;EAD2H=(?) ;AAD2=0 ;X=(?) ;Y=(?) ;X=X+28 ;A_COPY_AC> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 <<1!BUG1!>> ;CHECK IF "SLANT"=1/0 ;A_COPY_AC> COPY
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ CHG_COLOR: MOV SUB SKIP2: MOV SUB INC INC INC INC INC INC INC INC INC INC	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H CHG_COLOR DX AL,50H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,4CH CHG_COLOR DS ES:WORD PTR PLANES,DX SI MES_2 BX,2 CX,30 SI SI	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN ;"R"=RED ;"L"=BLACK ;PLANES=(?) ;X=2 ;Y=Y-30		SUB ; JIS CODE ; MOV SHL SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL ; POP MOV MOV MOV MOV MOV MOV MOV MOV MOV	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 CS: WORD PTR EAD2L, AX DS: WORD PTR AD2L, AX DS: WORD PTR X, BX AX, 8008H CS: FLAG, 2 MES_3 AX, 8009H DS: WORD PTR DXX, -4 DS: WORD PTR DXX, -4 DS: WORD PTR DXX, -4 DS: WORD PTR CMA, AX DS: WORD PTR AD2L, AX DS: WORD P	EAD2H=(?) :(AD2=0 :X=(?) :X=(?) :X=X+28 :A_COPY_AC> COPY :ESE=0,REV=0,ROT=0,{MD} :FAST=0 <<11BUG1!>> :CHECK 1F "SLANT"=1/0 :A_COPY_AC> COPY :ESE=0,REV=0,ROT=0,{MD} :SLANT
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ CHG_COLOR: MOV SUB SKIP2: MOV SUB INC INC INC INC INC INC INC INC INC INC	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,42H CHG_COLOR DX SI MES_2 BX,2 CX,30 SI SI MES_2	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN ;"R"=RED ;"L"=BLACK ;PLANES=(?) ;X=2 ;Y=Y-30	; ADJUST	SUB ; JIS CODE ; MOV SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL ; POP MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 DS S: WORD PTR EAD2L, AX DS: WORD PTR Y, BX, 0 DS: WORD PTR Y, CX BX, 2B AX, 8008H CS: FLAG, 2 MES, 3 AX, 8009H	EAD2H=(?) ;AD2=0 ;X=(?) ;Y=(?) ;X=X+28 ;A_COPY_AC> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 <<1!BUG1!>> ;CHECK IF "SLANT"=1/0 ;A_COPY_AC> COPY ;ESE=0,REV=0,ROT=0,{MD} ;SLANT ;DXX=-4
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP INC CMP IZ INC CMP INC CMP INC CMP IZ INC CMP INC CMP INC CMP INC CMP INC CMP INC CMP INC CMP INC CMP INC CMP INC CMP INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB INC SUB SUB INC SUB SUB INC SUB SUB SUB SUB SUB SUB SUB SUB SUB SUB	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,4CH CHG_COLOR DS ES:WORD PTR PLANES,DX SI MES_2 BX,2 CX,30 SI SI MES_2 FT-JIS CODE TO JIS CODE AX,[SI] AL,AH AH,70H	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN ;"R"=RED ;"L"=BLACK ;PLANES=(?) ;X=2 ;Y=Y-30	; ADJUST	SUB ; JIS CODE ; MOV SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL ; POP MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 DS S: WORD PTR EAD2L, AX DS: WORD PTR Y, EAD2L, AX DS: WORD PTR Y, EAD2L, AX DS: WORD PTR Y, EAD2, 0 DS: WORD PTR Y, EAD2, 0 DS: WORD PTR Y, CX BX, 2 AX, 8009H DS: WORD PTR DXX, -4 DS: WORD PTR DXX, -4 DS: WORD PTR COM, AX DS AX, 5500H DS, AX S1 S1	EAD2H=(?) (AD2=0 (X=(?) (X=(?) (X=(?) (X=X+28) (A_COPY_AC> COPY (ESE=0,REV=0,ROT=0,{MD}) (FAST=0 <<11BUG11>> (CHECK 1F "SLANT"=1/0 (A_COPY_AC> COPY (ESE=0,REV=0,ROT=0,{MD}) (SLANT) (DXX=-4) (<com_flags> (MESSAGE SEGMENT)</com_flags>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ CHG_COLOR: MOV INC SUB INC SUB INC INC INC INC INC INC INC INC INC INC	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H CHG_COLOR DX SI MES_2 BX,2 CX,30 SI MES_2 FT-JIS CODE T0 JIS CODE AX,[SJ] AL,AH AH,70H AL,AL SJIS_JIS_1	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN ;"R"=RED ;"L"=BLACK ;PLANES=(?) ;X=2 ;Y=Y-30	; ADJUST	SUB ; JIS CODE ; MOV SHL SHL SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL SHL SHL SHL SHL SHL SHL SHL SHL SH	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 DX, 1 DX, 1 DX, 1 DX, 1 AX, 1 DX, 1 DX, 1 AX, 1 DX, 1 DX, 1 AX, 1 DX, 1 DX, 1 AX, 1 DX, 1 DX, 1 AX, 1 DX, 1 DX	<pre>;EAD2H=(?) ;AAD2=0 ;X=(?) ;X=(?) ;X=X+28 ;A_COPY_AC> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 <<11BUG1!>> ;CHECK IF "SLANT"=1/0 ;A_COPY_AC> COPY ;ESE=0,REV=0,ROT=0,{MD} ;SLANT ;DXX=-4 ;<com_flags></com_flags></pre>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP INC INC INC INC INC INC INC INC INC INC	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H CHG_COLOR DX SI MES_2 FT-JIS CODE TO JIS CODE AX.[SI] AL,AH AH,70H AL,AL SJIS_JIS_1 AL	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN ;"R"=RED ;"L"=BLACK ;PLANES=(?) ;X=2 ;Y=Y-30	; ADJUST MES_3: ; MESSAGE	SUB ; JIS CODE ; MOV SHL SHL SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL ; POP MOV MOV MOV MOV MOV MOV MOV MOV	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 DS S: WORD PTR EAD2L, AX DS: WORD PTR Y, EAD2L, AX DS: WORD PTR Y, EAD2L, AX DS: WORD PTR Y, EAD2, 0 DS: WORD PTR Y, EAD2, 0 DS: WORD PTR Y, CX BX, 2 AX, 8009H DS: WORD PTR DXX, -4 DS: WORD PTR DXX, -4 DS: WORD PTR COM, AX DS AX, 5500H DS, AX S1 S1	EAD2H=(?) (AD2=0 (X=(?) (X=(?) (X=(?) (X=X+28) (A_COPY_AC> COPY (ESE=0,REV=0,ROT=0,{MD}) (FAST=0 <<11BUG11>> (CHECK 1F "SLANT"=1/0 (A_COPY_AC> COPY (ESE=0,REV=0,ROT=0,{MD}) (SLANT) (DXX=-4) (<com_flags> (MESSAGE SEGMENT)</com_flags>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ CHG_COLOR: MOV SUB INC INC INC INC INC INC INC INC INC INC	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,42H CHG_COLOR DX AL,4CH CHG_COLOR DS ES:WORD PTR PLANES,DX SI MES_2 BX,2 CX,30 SI SI MES_2 FT-JIS CODE TO JIS CODE AX.[SI] AL,AH AH,70H AL,AL SJIS_JIS_I AL,AH AL,9EH	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN ;"R"=RED ;"L"=BLACK ;PLANES=(?) ;X=2 ;Y=Y-30	;ADJUST MES_3: ;	SUB ; JIS CODE ; MOV SHL SHL SHL RCL SHL SHL RCL SHL SHL SHL SHL SHL SHL SHL SH	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 DS S: WORD PTR EAD2L, AX DS: WORD PTR Y, EAD2L, AX DS: WORD PTR Y, EAD2L, AX DS: WORD PTR Y, EAD2, 0 DS: WORD PTR Y, EAD2, 0 DS: WORD PTR Y, CX BX, 2 AX, 8009H DS: WORD PTR DXX, -4 DS: WORD PTR DXX, -4 DS: WORD PTR COM, AX DS AX, 5500H DS, AX S1 S1	EAD2H=(?) (AD2=0 (X=(?) (X=(?) (X=(?) (X=X+28) (A_COPY_AC> COPY (ESE=0,REV=0,ROT=0,{MD}) (FAST=0 <<11BUG11>> (CHECK 1F "SLANT"=1/0 (A_COPY_AC> COPY (ESE=0,REV=0,ROT=0,{MD}) (SLANT) (DXX=-4) (<com_flags> (MESSAGE SEGMENT)</com_flags>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP INC SI FOP RET; CHG_COLOR: MOV INC INC INC INC INC INC INC INC INC INC	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H CHG_COLOR DX AL,59H CHG_COLOR DX AL,47H CHG_COLOR DX AL,47H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H CHG_COLOR DX SI MES_2 BX,2 CX,30 SI SI MES_2 FT-JIS CODE T0 JIS CODE AX,[SJ] AL,AH AH,AH	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN ;"R"=RED ;"L"=BLACK ;PLANES=(?) ;X=2 ;Y=Y-30	; ADJUST MES_3: ; MESSAGE	SUB ;]IS CODE ; MOV SHL SHL SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL JH MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 DS S: WORD PTR EAD2L, AX DS: WORD PTR Y, EAD2L, AX DS: WORD PTR Y, EAD2L, AX DS: WORD PTR Y, EAD2, 0 DS: WORD PTR Y, EAD2, 0 DS: WORD PTR Y, CX BX, 2 AX, 8009H DS: WORD PTR DXX, -4 DS: WORD PTR DXX, -4 DS: WORD PTR COM, AX DS AX, 5500H DS, AX S1 S1	EAD2H=(?) (AD2=0 (X=(?) (X=(?) (X=(?) (X=X+28) (A_COPY_AC> COPY (ESE=0,REV=0,ROT=0,{MD}) (FAST=0 <<11BUG11>> (CHECK 1F "SLANT"=1/0 (A_COPY_AC> COPY (ESE=0,REV=0,ROT=0,{MD}) (SLANT) (DXX=-4) (<com_flags> (MESSAGE SEGMENT)</com_flags>

CHRDRAW NAME GROUP PROG SEGMENT BYTE PUBLIC PGROUP 'PROG PROG 12 CHR_DRAW_2 ASSUME CS:PGROUP MOV BX.8009H CHR_DRAW_DEMO_L, CHR_DRAW_DEMO_H, CLIP_DEMO PUBLIC CHR_DRAW_DEMO_L_90, CHR_DRAW_DEMO_H_90 PUBLIC DS:WORD PTR DXX.-4 MOV CHR_DRAW_2: TRAFILL_EXE_H:NEAR, WAIT:NEAR, DMCLO_C_WAIT:NEAR EXTRN MOV DX.840H CX,CS:WORD PTR WORK 6 MOV VORK_2:VORD, VORK_3:VORD, VORK_4:VORD, VORK_5:VORD VORK_6:VORD, STATUS:VORD, PMAX:VORD, PITCHS:VORD EAD2H:VORD, VORK_1:VORD, EAD2L:VORD, X:VORD, Y:VORD DHH:VORD, DV:VORD, MODIO:VORD, PLANES:VORD, COM:VORD DXX:VORD, FLAG:VORD, MAGETC:VORD XXCLMIN:VORD, YCLMIN:VORD, XCLMAX:VORD, YCLMAX:VORD EXTRN DI,CS:WORD PTR WORK_3 MON EXTRN MOV SI,CS:WORD PTR WORK_2 EXTRN CHR DRAW 1: EXTRN INC CS:WORD PTR WORK_1 EXTRM AX,CS:WORD PTR WORK_1 DS:WORD PTR PLANES,AX MOV EXTRN MOV DS:WORD PTR EAD2L,DX MOV CHRDRAW PROC NEAR MOV DS:WORD PTR X.SI DS:WORD PTR Y,DI MOV MOV DS:WORD PTR DHH,23 DS:WORD PTR DV,23 < CHARACTER FONT EXPANSION DEMO > MOV MOV DS:WORD PTR MOD10,97H MOV DS:WORD PTR COM.BX DX,40H ADD CHR DRAW DEMO L: SI,CS:WORD PTR WORK_4 ADD CS:WORD PTR WORK_2,4 CS:WORD PTR WORK_3,476 MOV :X=4 DEC C1. ;Y=476 MOV CHR_DRAW_1 JNZ CS:WORD PTR WORK_4,32 ;BODY (H)=32 MOV CL,CS:BYTE PTR WORK_6 DI,CS:WORD PTR WORK_5 MOV CS:WORD PTR WORK_5,30 CS:WORD PTR WORK_6,1014H MOV : BODY (V)=30 SUB ;20 CHARS X 16 ROWS MOV MOV SI,CS:WORD PTR WORK_2 CHR_DRAW_EXE_ALL CALL DEC CH RET CHR DRAW 1 JNZ RET CHR_DRAW_DEMO_H: CS:WORD PTR WORK_2,2 MOV :X=2 CS:WORD PTR WORK_3,746 ;Y=746 MOV < 90 DEGREES ROTATED CHARACTER FONT EXPANSION DEMO > ;BODY (H)=28 CS:WORD PTR WORK 4,28 CS:WORD PTR WORK 5,30 MOV ;BODY (V)=30 MOV CS:WORD PTR WORK_6,1928H ;40 CHARS X 25 ROWS MOV CALL CHR DRAW EXE ALL CHR_DRAW_DEMO_L_90: CS:WORD PTR WORK_2,2 CS:WORD PTR WORK_3,3 CS:WORD PTR WORK_4,28 RET MOV MOV CHR DRAW EXE ALL: MOV DS:WORD PTR STATUS,1 TEST MOV CS:WORD PTR WORK 5,30 CS:WORD PTR WORK 6,1510H CHR_DRAW_EXE_ALL DS:WORD PTR PMAX,4 CHECK IF PPBUSY=1/0 IN7 MOV MOV :PMAX=4 CHR_DRAW_EXE_ALL_90 CALL DS:WORD PTR PITCHS,2 DS:WORD PTR EAD2H,16H MOV ;PITCHS=2 RET MOV :EAD2H=16H CS:WORD PTR WORK_1,0 MOV ;PLANES=0 CHR_DRAW_DEMO_H 90: TEST CS:BYTE PTR FLAG.2 CS:WORD PTR WORK_2,2 MOV BX,8008H ;<A_COPY_AC> COPY MOV CS:WORD PTR WORK_3,3 MOV -1-CLIP_DEMO_1 DS:WORD PTR MAGETC,1FFH DS:WORD PTR XCLMIN,200 DS:WORD PTR YCLMIN,180 CS:WORD PTR WORK_4,28 CS:WORD PTR WORK_5,30 CS:WORD PTR WORK_6,2819H MOV :BODY (H)=28 JNZ ;BODY (V)=30 MOV MOV MOV :25 CHARS X 40 ROWS MOV CALL CHR DRAW EXE ALL 90 MOV DS:WORD PTR XCLMAX,900 MOV RET MOV DS:WORD PTR YCLMAX,570 CLIP DEMO EXE CHR DRAW FXE ALL 90: CALL DS:WORD PTR STATUS,1 TEST CHECK IF PPBUSY=1/0 CHR_DRAW_EXE_ALL_90 DS:WORD PTR PMAX,4 CHR DRAW DEMO H JNZ CALL ;PMAX=4 MOV DS:WORD PTR PITCHS,2 PITCHS=2 CLIP_DEM0_2: MOV :EAD2H=16H DS:WORD PTR STATUS,1 MOV DS:WORD PTR FAD2H, 16H TEST ;PLANES=0 ;<A_COPY_AC> 90_COPY ;ESE=0,REV=0,ROT=0,{MD} CS:WORD PTR WORK_1,0 JNZ CLIP_DEM0_2 MOV DS:WORD PTR MAGETC, OFFH BX.8018H MOV MOV CALL. CLIP DEMO EXE ;EAD2L=840H (JIS-3021H) ;CHARACTERS X ROWS MOV DX.840H CHR_DRAW_DEMO_H CX,CS:WORD PTR WORK_6 DI,CS:WORD PTR WORK_3 CALL MOV ;Y=(?) MOV MOV SI,CS:WORD PTR WORK_2 :X=(?) CLIP_DEM0_3: DS:WORD PTR STATUS,1 CHR_DRAW_1_90: TEST ;EAD2L=(?) DS:WORD PTR EAD2L, DX CLIP DEMO_3 MOV JNZ DS:WORD PTR X,SI DS:WORD PTR Y,DI DS:WORD PTR DHH,23 ;X=(?) ;Y=(?) DS:WORD PTR MAGETC, 2FFH MOV MOV MOV ;DH=23 CLIP_DEMO_EXE: MOV BX.70ACH MOV DS:WORD PTR DV,23 :DV=23 MOV DS:WORD PTR MODIO,97H ;MOD1=9,MOD0=7 MOV CS:WORD PTR WORK_1 ;PLANES+1 --> PLANES TRAFILL_EXE_H INC CALL AX.CS:WORD PTR WORK 1 MOV MOV AX,40 DS:WORD PTR PLANES, AX ;PLANES=(?) CALL WAIT MOV CALL DS:WORD PTR COM.BX . < COM FLAGS> DMCLO C WAIT MOV CHARACTER CODE + 40H ADD DX.40H RET DI,CS:WORD PTR WORK_5 ADD ;Y+(?) DEC CL. CHR_DRAW_1_90 CHRDRAW ENDP INZ CL,CS:BYTE PTR WORK_6 MOV PROG ENDS SI,CS:WORD PTR WORK_4 DI,CS:WORD PTR WORK_3 :X+(?) ADD END :Y=(?) MOV DEC JNZ Ret CHR DRAW 1 90 < CLIPPING DEMO > ; CLIP DEMO: CALL CHR DRAW DEMO H FND OF BACK-GROUND CLIP DEMO 1: DS:WORD PTR STATUS,1 TEST

-4-

;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 <<!!BUG!!>> ;CHECK |F "SLANT"=1/0

;ESE=0,REV=0,ROT=0,{MD}

;EAD2L=840H (JIS-3021H) ;CHARACTERS X ROWS

;PLANES+1 --> PLANES

:<A COPY AC> COPY

;SLANT

:DXX=-4

:Y=(?)

:X=(?)

;PLANES=(?)

;EAD2L=(?)

;MOD1=9,MOD0=7

<COM. FLAGS> CHARACTER CODE + 40H

:X=(?)

;Y=(?)

:08=23

;DV=23

:X+(?)

;Y-(?)

:X=(?)

;X=2 :Y=3

:X=2

;Y=3

-2-

;BODY (H)=28

: BODY (V)=30

;16 CHARS X 21 ROWS

;MAGETC=OFFH NON-CLIP

;END OF BACK-GROUND

:MAGETC=OFFH CLIP-OUT

:END OF BACK-GROUND

:MAGETC=OFFH CLIP-IN

;TL=1,SS=0,WL=1,WR=1

:<A TRAFILL>

:XCLMIN=200 ;YCLMIN=180

XCLMAX=900

:YCLMAX=570

GETPUT NAME MOV DX.4 PGROUP GROUP PROG SEGMENT BYTE PUBLIC MOV BX,980FH PROG 'PROG ASSUME CS:PGROUP PUT_DEMO_EXE_1: TEST PUBLIC GET_PUT_DEMO, PUT_DATA_TRAN JNZ. MOV EXTRN WALT:NEAR MOV AX,CS:[SI] MOV EXTRN STATUS:WORD, DHH:WORD, DV:WORD, X:WORD, Y:WORD SI INC PLANES:WORD, PMAX:WORD, MODIO:WORD, COM:WORD PITCHS:WORD, PDISPSH:WORD, PDISPSL:WORD, AGDC_SEG:WORD EXTRN INC SI MOV EXTRN MOV AX,CS:[SI] GETPUT PROC NEAR INC SI INC SI MOV • • • • < GET/PUT DEMO > MOV MOV MOV MOV GET PUT DEMO: PUSH DS INIT_PUT_DATA AX,6COOH CALL MOV MOV DS,AX CX,3000H PUT_DATA_TRAN PUT_GET_DEM0_1 PUT_GET_DEM0_1 CALL MOV CALL CALL CALL PUT_GET_DEMO_1 POP DS BX.20H CALL PUT DEMO EXE ADD DEC RET PUT_DEMO_EXE_1 INZ INIT PUT DATA: RET PUSH DS PUT DATA TRAN: SEGMENT "MAAIAN BUF" AX.6600H MOV SI PUSH MOV DS.AX MOV AX,6COOH ;SEGMENT "MAAJAN" MOV MOV FS.AX MOV ES.AX \$1,0 MOV \$1.0 MOV MOV D1,0 MOV DI,3EH ;256 X 256 X 3 PLANES PUT DATA_TRAN_1: CX.3000H MOV MOVSW MOV AX,[SI] REPZ POP DS MOV ES:[DI],AX RET INC SI INC SI PUT_GET_DEMO 1: 1.00P PUT_DEMO_EXE POP CALL SI CALL GET_DEMO_EXE RET MOV AX.10 CALL WAIT GET_DEMO_EXE: RET TEST GET_DEMO_EXE JNZ PUT_DEMO_EXE: MOV SI. OFFSET GET PUT DATA MOV MOV -1-; X=705 DS:WORD PTR X,705 MOV NAME ENLARGE DS:WORD PTR Y,700 DS:WORD PTR PITCHS,46H ; Y=700 ;PITCHS=46H MOV GROUP PROG SEGMENT BYTE PUBLIC PGROUP MOV PROG DS:WORD PTR PDISPSH,1 PDISPSH=1 MOV ASSUME CS:PGROUP :PDISPSI.=0 MOV DS:WORD PTR PDISPSL,0 ;<GET_C> DS:WORD PTR COM, 9AOEH MOV PUBLIC ENLARGE_DEMO ;{MSD} ;256 X 256 X 3 PLANES MOV CX.3000H EXTRN AX,6COOH SEGMENT "MAAJAN" MOV EXTRN MOV ES,AX EXTRN ;256 X 256 X 3 PLANES CX,3000H MOV GET DATA TRAN: ENLARGE PROC NEAR ASSIGN GET/PUT PORT MOV S1.3EH MOV 01.0 GET_DATA_TRAN_1 MOV AX,[SI] ES:[DI],AX MOV INC DI INC D1 ENLARGE DEMO: GET_DATA_TRAN_1 LOOP TEST RET INZ. MOV GET_PUT_DATA: MOV 150,700, 960,445 960,315, 150, 60 ;1.--> 2. X,Y nu MOV ;3.--> 4. X,Y DW MOV MOV MOV GETPUT ENDP MOV CX,16 PROG ENDS DX,1FFH MOV END BX,849FH MOV ENLARGE EXE: TEST IN7 ENLARGE EXE AX,CS:[SI] MOV INC SI INC SI MOV MOV AX,CS:[SI] INC SI INC MOV AX,CS:[SI] MOV INC SI INC SI MOV MOV AX,CS:[SI] MOV INC S INC SI

;<PUT_C> ;ROT=0,{MSD} DS:WORD PTR STATUS.1 PUT_DEMO_EXE_1 ;CHECK IF PPBUSY=1/0 DS:WORD PTR DHH,255 DS:WORD PTR DV,255 ;DH=255 :DV=255 DS:WORD PTR X,AX ;X=(?) :Y=(?) DS:WORD PTR V.AX DS:WORD PTR PLANES,7 ;PLANES=7 DS:WORD PTR PMAX,4 DS:WORD PTR MODIO,0 :PMAX=4 ;MOD1=0,MOD0=0 DS:WORD PTR COM, BX ;<COM.FLAG> ;SEGMENT "MAAJAN" :256 X 256 X 3 PLANES ;CHANGE ROTATION AX, WORD PTR AGDC_SEG ASSIGN GET/PUT PORT PUT_DATA_TRAN_1 DS:WORD PTR STATUS.1 ;CHECK IF PPBUSY=1/0 DS:WORD PTR DHH,255 DS:WORD PTR DV,255 :DH=255 :DV=255 -2-'PROG' STATUS:WORD, PITCHS:WORD, PMAX:WORD, MODIO:WORD X:WORD, Y:WORD, XS:WORD, YS:WORD, DHH:WORD, DV:WORD MAGETC:WORD, COM:WORD, PDISPSH:WORD, PDISPSL:WORD < ENLARGEMENT COPY DEMO > DS:WORD PTR STATUS,1 CHECK IF PPBUSY=1/0 ENLARGE_DEMO DS:WORD PTR PDISPSH,1 ;PDISPSH=1 DS:WORD PTR PDISPSL,0 DS:WORD PTR PITCHS,46H :PDISPSL=0 ;PITCHS=46H DS:WORD PTR PMAX,4 DS:WORD PTR MOD10,0 ;PMAX=4 ;M0D1=0,M0D0=0 SI, OFFSET ENLARGE_DATA REPETITION COUNTS ;MAGH=15,MAGV=15 ;<A_COPY_CC> ES_COPY ;ROT=0,{MSD} DS:WORD PTR STATUS,1 CHECK IF PPRUSY=1/0 DS:WORD PTR X,AX :X=(?) DS:WORD PTR Y,AX :Y=(?) DS:WORD PTR XS,AX ;XS=(?) DS:WORD PTR YS,374 ;YS=374 MOV DS:WORD PTR DHH,AX ;DHH=(?)

REPETITION COUNTS

-3-

-1-

MOV INC INC MOV MOV SUB LOOP RET ; ENLARGE_DATA: DW DW DW DW DW DW DW DW DW DW DW DW DW	AX,CS:[S1] S1 S1 DS:WORD PTR DV,AX DS:WORD PTR COM,BX DX;WORD PTR COM,BX DX,11H ENLARGE EXE 30,749,389,339,109 30,629,400,318,102 30,509,410,297,95 30,389,421,275,88 30,269,432,254,82 30,149,442,233,75 390,749,453,212,68 390,629,464,190,61 390,509,474,169,54 390,149,465,148,47 750,749,495,127,40 750,629,506,105,33 750,509,517,84,27 750,389,527,63,20 750,269,538,42,13 750,149,549,20,6 ;	:DV=(?) :MAGETC=(?) : <com.flag> :MAGH-1>MAGH :MAGV-1>MAGV :X,Y,XS,DHH,DV 16/16 :X,Y,XS,DHH,DV 16/11 :X,Y,XS,DHH,DV 16/11 :X,Y,XS,DHH,DV 16/11 :X,Y,XS,DHH,DV 16/11 :X,Y,XS,DHH,DV 16/10 :X,Y,XS,DHH,DV 16/9 :X,Y,XS,DHH,DV 16/9 :X,Y,XS,DHH,DV 16/6 :X,Y,XS,DHH,DV 16/5 :X,Y,XS,DHH,DV 16/5 :X,Y,XS,DHH,DV 16/4 :X,Y,XS,DHH,DV 16/2 :X,Y,XS,DHH,DV 16/2 :X,Y,XS,DHH,DV 16/1</com.flag>	NAME SIRIINK PCROG PROG SEGMENT BYTE PUBLIC 'PROG' ASSUME CS:PGROUP : PUBLIC SHRINK_DEMO, SHRINK_PUT, SHRINK_DATA, SHRINK_EXE : EXTRN MESSAGE_WAIT:NEAR, DMCLD_C_WAIT:NEAR, PUT_DATA_TRAN:NEAR : EXTRN MESSAGE_WAIT:NEAR, DMCLD_C_WAIT:NEAR, PUT_DATA_TRAN:NEAR : EXTRN MESSAGE_WAIT:NEAR, DMCLD_C_WAID, Y:UORD, COM:WORD EXTRN DHH:WORD, DY:UORD, MACETC:WORD, MODIO:WORD, COM:WORD EXTRN DHISWORD, PLANES:WORD, MICHING, MODIO:WORD, COM:WORD EXTRN PDISPSH:WORD, EAD2H:WORD, dAD2:BYTE, EAD2L:WORD : SHRINK DEMO: CALL SHRINK_COPY DEMO > : : SHRINK DEMO: CALL SHRINK_PUT :16/16 MOV SI.OFFSET SHRINK_DATA MOV CX.6 :15/16> 10/16 CALL SHRINK_EXE PUSH SI MOV SI.21 CALL SHRINK_PUT :16/16 MOV SI.21 CALL SHRINK_PUT :16/16 MOV CX.9 : 9/16> 1/16 CALL SHRINK_PUT :16/16 MOV CX.10FH :MAGETC=10FH MOV CX.16 :MAGETC=10FH MOV DS:WORD PTR STATUS.1 JNZ SHRINK_MAGH :CHECK IF PPBUSY=1/0 MOV DS:WORD PTR STATUS.1 JNZ SHRINK_MAGH :CHECK IF PPBUSY=1/0 MOV DS:WORD PTR YS.10 :XS=10 MOV DS:WORD PTR YS.10 :XS=10 MOV DS:WORD PTR YS.10 :XS=10 MOV DS:WORD PTR YS.10 :XS=10 MOV DS:WORD PTR Y.10 :X=10 MOV DS:WORD PTR Y.10 :X=10 MOV DS:WORD PTR Y.10 :X=10 MOV DS:WORD PTR Y.10 :X=10 MOV DS:WORD PTR MAGETC.DX :MAGH(:2).MAGU=0FH MOV DS:WORD PTR MAGETC.DX :MAGH(:2).MAGH=0FH MOV DS:WORD PTR MAGETC.DX :MAGH(:2).MAGH=0FH MOV DS:WORD PTR MAGETC.DX :MAGH(:2).MAGH=0FH MOV DS:WORD PTR MAGETC.DX :MAGH(:2).MAGH=0FH MOV DS:WORD PTR MAGETC.DX :MAGH(:2).MAGH] : ADD DX.10H :MAGH1> MAGH
LOOP	-2- Shrink_Magh		-1- JNZ SHRINK EXE 1 ;CHECK IF PPBUSY=1/0
LOOP ; MOV MOV SHRINK_MAGV: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SHRINK_MAGH DX,1FOH CX,16 DS:WORD PTR STATUS,1 SHRINK_MAGV DS:WORD PTR XS,10 DS:WORD PTR YS,739 DS:WORD PTR YS,739 DS:WORD PTR Y,259 DS:WORD PTR Y,259 DS:WORD PTR V,259 DS:WORD PTR DH,575 DS:WORD PTR DV,214 DS:WORD PTR MAGETC.DX DS:WORD PTR COM,840FH DX SHRINK_MAGV	;MAGH=15,MAGV=0 ;MAGV=0> 15 :CHECK IF PPBUSY=1/0 ;XS=10 ;YS=739 : X=10 ; Y=259 ;DH=575 ;DV=214 :MAGH=(?),MAGV=OFH ; <a_copy_cc> ES_COPY ;REV=0,ROT=0,{MSD} ;MAGV+1> MAGV</a_copy_cc>	JNZ SHRINK_EXE_1 ;CHECK IF PPBUSY=1/0 MOV AX,CS:[SI] INC SI HOV DS:WORD PTR X,AX ;X=(?) MOV AX,CS:[SI] INC SI INC SI INC SI MOV DS:WORD PTR Y,AX ;Y=(?) MOV AX,CS:[SI] INC SI INC SI INC SI INC SI INC SI INC SI MOV DS:WORD PTR MAGETC,AX ;MAGETC=(?) MOV DS:WORD PTR EAD2H,1 ;CAD2H=1 (GO) MOV DS:WORD PTR EAD2L,10 ;CAD2L=100 (GO) MOV DS:WORD PTR EAD2L,10 ;CAD2L=100 (GO) MOV DS:WORD PTR EAD2L,700 ;CAD2L=100 (GO) MOV DS:WORD PTR COM,800BH ;AA_COPY_AC> ES_COPY
; SHRINK_PUT: TEST JNZ MOV MOV MOV	DS:WORD PTR STATUS,1 SHRINK_PUT DS:WORD PTR DHH,575 DS:WORD PTR DV,214 DS:WORD PTR X,10	;CHECK IF PPBUSY=1/0 ;DH=575 ;DV=214 ; X=10	LOOP SHRINK_EXE_1 RET ; SHRINK_DATA: DW 10,514,1EEH ; 15/16 X,V,MAGETC

-2-

-3-

	NAME	PUSH_PO	Р	
PGROUP		PROG		
PROG	SEGMENT		BLIC	'PROG'
	ASSUME	CS:PGR0	UP	
	;			
	PUBLIC	PUSH_EX	E, POP	EXE
	;			
PUSH POP		PROC	NEAR	
	;			
POP_EXE				
	POP	ES		
	POP	DS		
	POP	DI		
	POP	SI		
	POP	DX		
	POP	CX		
	POP	BX		
	POP	AX		
	IRET			
	;			
PUSH_EX	E:			
	POP	AX		
	PUSH	BX		
	PUSH	CX		
	PUSH	DX		
	PUSH	SI		
	PUSH	DI		
	PUSH	DS		
	PUSH	ES		
	PUSH	AX		
	RET			
	;			
;				
PUSH_PO	P	ENDP		
PROG	ENDS			
	END			

-1-

I made entire 80286 assembly code above for the purpose of both functional verification and demonstration for customers as well as emulating Graphics BIOS on NEC PC-9801 series replacing the entry points.

The PC-9801 had been most popular 16 bit personal computer in Japan due to implementation of full graphics capability as standard feature but eventually defeated by IBM PC clones that assembled Chips & Technologies' chip sets.

This source code demonstrates how to drive μ PD72120 Advanced Graphics Display Controller LSI from system side exhibiting concrete examples.

To see the details, zoom up.