DEMOMAIN IMP NEAR PTR ASYNC NAME E1: GROUP PROG SEGMENT BYTE PUBLIC JMP NEAR PTR DEMO EXE JMP NEAR PTR DEMO\_INIT PGROUP E2: PROG 'PROG' E3: ASSUME CS:PGROUP E4: JMP NEAR PTR GLIOMAIN PUBLIC WORK\_1, WORK\_2, WORK\_3, FLAG, FLAG\_2 WORK\_4, WORK\_5, WORK\_6, WORK\_7, AGDC\_SEG PUBLIC < DEMONSTRATION > 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 PUSH EXE CALL EXTRN EXTRN MOV AL.1 OD1H.AL FXTRN OUT EXTRN IN AL, OD1H TEST AL,1 EXTRM AX.ODOOOH EXTRN MOV SHRINK\_DATA:NEAR, SHRINK\_EXE:NEAR, PAINT\_DEMO:NEAR FRCOPY\_DEMO:NEAR, RECT\_DEMO:NEAR, SCROLL\_DEMO:NEAR CLIP\_DEMO:NEAR, GLIOMAIN:NEAR JZ DEMO\_EXE\_1 EXTRN EXTRN MOV AX,8000H EXTRN DEMO\_EXE\_1: DEMOMAIN PROC NEAR MOV DS.AX MOV MOV ORG 1008 CALL DMCLO C WAIT 1 CALL DEMO\_THRU , ;DATA BUFFER FIELD MOV MOV \$1.0 WORK\_1 MES\_CL\_WAIT ;0100H CALL DW 0 0 ;0102H MOV \$1,32 WORK 2 DW WORK\_3 DW 0 :0104H CALL MES\_CL\_WAIT 0 ;0106H FLAG DB ;BITO : REAL TIME/WAIT ;BIT1 : SLANT TEXT/ ;LINE, RECTANGLE, CIRCLE, ELLIPSE BIT2 : PIXEL MODE/ MOV FLAG 2 DB 0 :01078 CALL VECT DEMO WORK\_4 ;0108H MOV DW 0 CALL WORK\_5 DW 0 ;010AH VECT DEMO WORK 6 DW 0 :010CH MOV WORK\_7 DW ;010EH CALL VECT\_DEMO 0 AGDC\_SEG DW 0 ;0110H ;ENLARGE JUMP TABLE MOV MOV \$1,16 CALL MES CL WAIT : MOV -1-LINE\_DEMO\_H CS:BYTE PTR FLAG,1 CALL MOV \$1,13 ;WAIT MODE MES\_CL\_WAIT MOV CALL CALL ENLARGE\_DEMO CALL CHR\_DRAW DEMO MOV SI,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 MES CL WAIT MOV \$1.20 CALL. MES\_CL\_WAIT ;MESSAGE(20) CALL CALL CALL SHRINK\_DEMO MOV MOV CALL CHR\_DRAW DEMO \$1,22 MES\_CL\_WAIT :MESSAGE(22) MOV CALL CALL ;GET, PUT MOV CALL CHR DRAW DEMO\_90 CS:BYTE PTR FLAG,1 ;WAIT MODE MOV MOV SI,18 ;SCROLL MES\_CL\_WAIT GET\_PUT\_DEMO CALL :MESSAGE(18) MOV CALL MOV \$1,19 MOV \$1,31 MES CL WALT MES CL WAIT :MESSAGE(19) CALL CALL SCROLL DEMO CALL ;FR\_COPY CALL DMCLO C WAIT MOV ;WAIT MODE ;RECFILL, CRLFILL, ELPSFILL, TRIFILL, TRAFILL CS:BYTE PTR FLAG,1 SI,26 MES\_CL\_WAIT MOV MOV ;MESSAGE(26) CALL CALL FRCOPY DEMO CALL FILL DEMO MOV \$1.27 MOV CALL MES\_CL\_WAIT :MESSAGE(27) CALL FILL DEMO MOV : PAINT CALL FILL DEMO :CLIPPING MOV CS: BYTE PTR FLAG.1 WAIT MODE MOV \$1.23 MES\_CL\_WAIT ;MESSAGE(23) MOV CALL CALL PAINT DEMO MOV \$1.33 MES\_CL\_WAIT MOV \$1,24 CALL CALL MES\_CL\_WAIT :MEESAGE(24) CALL CLIP DEMO DMCLO C WAIT 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 CHR\_DRAW\_DEM0\_90 MOV REAL TIME MODE MOV CALL CALL CS:BYTE PTR FLAG,1 ;WAIT MODE

:011BH :ENABLE /CSIR/ ;PC-XA/XL -- 8000H :PC9800 -- D000H OPEN MEMORY WINDOW CS:WORD PTR AGDC\_SEG,AX CS:BYTE PTR FLAG,0 REAL TIME MODE CS:BYTE PTR FLAG,1 WAIT MODE ;MESSAGE(0) :MESSAGE(32) CS:BYTE PTR FLAG,0 ;REAL TIME MODE CS:BYTE PTR FLAG,1 WAIT MODE CS:BYTE PTR FLAG,0 REAL TIME MODE CS:BYTE PTR FLAG.1 WALT MODE :MESSAGE(16) CS:BYTE PTR FLAG.0 REAL TIME MODE -2-;MESSAGE(13) WAIT MODE (SLANT) CS:BYTE PTR FLAG,3 MES\_CL\_WAIT CHR\_DRAW\_DEMO CS:BYTE PTR FLAG,1 :MESSAGE(25) WAIT MODE ;MESSAGE(44) CHR\_DRAW\_DEMO\_90 CS:BYTE PTR FLAG,0 REAL TIME MODE CS:BYTE PTR FLAG,2 CHR DRAW DEMO REAL TIME MODE (SLANT) CS:BYTE PTR FLAG,0 ;REAL TIME MODE CS:BYTE PTR FLAG,1 WAIT MODE :MESSAGE(31) CS:BYTE PTR FLAG,0 ;REAL TIME MODE CS:BYTE PTR FLAG.1 WAIT MODE CS: BYTE PTR FLAG.0 REAL TIME MODE CS:BYTE PTR FLAG,1 WAIT MODE ;MESSAGE(33)

:0112H

;0115H

;0118H

## 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) \$1,43 MOV

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 DEMOMAIN 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, YCLMAX, 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 56H DV PITCHS EQU EQU 58H PITCHD 5AH EQU STMAX EQU 5CH PLANES PTNCNT EQU 5EH EQU 60H XCLMIN EQU 62H YCLMIN XCLMAX FOU 64H PROC NEAR 66H EQUATE DEMO EQU YCLMAX EQU 68H MAGETC EQU 6CH AGDC ADDRESS TABLE COM EQU DISP FLAGS EQU 6EH EQU 70H DISP\_PITCH 72H EQU 74H 76H EADORGL EQU OH DADL EQU DADH\_WC EQU EADORGH EQU 2H dADORG EQU 3 H GCSRX EQU 78H GCSRX EQU GCSRYS EQU 7AH EAD11. EOU 4 H GCSRYE EQU 7CH EADIH EQU 6H dAD1 EQU 7 H SYNC EQU 7EH EAD21 88 EOU EAD2H EQU MISCELLANEOUS EQUATE OAH ; dAD2 EQU PDISPSL EQU OBH WINDOW\_SEG\_XL EQU 8000H OCH PDISPSH EQU OEH WINDOW\_SEG\_98 EQU ODOOOH PDISPDL EQU 10H PDISPDH EQU 12H PMAX EQU 14H EQUATE\_DEMO ENDP ENDS MOD10 FOU 168 PROG PTNPL END EQU 18H PTNPH EQU 1 A H STACKL EQU 1CH EQU STACKH 1EH EQU 3CH BANK CTRL EQU 3DH STATUS EQU 3CH IRR EQU 3EH X EQU 40H 42H EQU DXX EOU 44H DY EQU 46H

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 MOV CALL 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 DS:WORD PTR GCSRYS, 100H GCSRVS=100H < MESSAGE PAGE ADDRESS TABLE GENERATION > 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 1C63H, 0C21H,0FFFFH,0EF7BH 0888H, 0000H, 0000H, 0401H :PDISPSH=1 MOV DS:WORD PTR PDISPSH,1 DW 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 0020H, 0001H, 2208H, 0410H MOV DS:WORD PTR PMAX.4 :PMAX=4 DW ;STACKL=OFFFFH DS:WORD PTR STACKL, OFFFFH 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 DS:WORD PTR PITCHD,46H DS:WORD PTR XCLMIN,0 MOV :PITCHD=46H DW 0020H, 0401H, 0AAAAH, 0411H ;XCLMIN=0 MOV DS:WORD PTR YCLMIN,0 ;YCLMIN=0 DW 1084H, 0842H, 0000H, 0000H ;TILE(1)-R MOV DS:WORD PTR XCLMAX,1119 DS:WORD PTR YCLMAX,749 :XCLMAX=1119 0000H, 294AH, 1004H, 0842H 0000H, 0000H, 294AH, 1084H MOV DW ;YCLMAX=749 MOV DW ;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 DW OC21H, OFBFFH, 54A5H, 2421H 1021H, OC21H, OFBFFH, 14A5H DW < TILING PATTERN TRANSFER > 0000H, 0000H, 0000H, 0000H TILE(1)-B DW 0000H, 0000H, 0000H, 0001H 0000H, 0000H, 0000H, 0000H 0000H, 0000H, 0000H, 0000H 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 PUSH DS DW 1111H, 0000H, 4444H, 0000H 1111H, 0000H, 4444H, 0000H :TILE(2)-G MOV AX.CS DW DW MOV DS,AX MOV SI, OFFSET TILING\_DATA DW 1111H, 0000H, 4444H, 0000H 1111H, 0000H, 4444H, 0000H AX, WORD PTR AGDC\_SEG MOV DW 4444H, 0000H, 1111H, 0000H ;TILE(2)-B MOV ES,AX DW 4444H, 0000H, 1111H, 0000H 4444H, 0000H, 1111H, 0000H MOV DI, OFOOOH DW ;TILING PATTERN LENGTH CX.300H MOV DW 4444H, 0000H, 1111H, 0000H MOV AL,2 DW OD1H.AL :ENABLE #CSDM# OUT REPZ MOVSW DW ODDDDH, 5555H, 7777H, 5555H :TILE(3)-R ODDDDH, 57D5H, 77F7H, 57D5H MOV AL.1 DW :ENABLE #CSIR# OD7DDH, 53D5H, 7077H. 5555H OUT OD1H.AL DW ODDDDH, 5555H, 7777H, 5555H POP D₩ DS RET DW 2222H, 8888H, 8888H, 2222H ;TILE(3)-G 2222H, 8808H, 8008H, 2022H DW DW 2022H, 8008H, 8088H, 2222H DW 2222H, 8888H, 8888H, 2222H OAAAAH, OFFFFH, OAAAAH, OFFFFH < TILING DATA > DW ;TILE(3)-B OAAAAH, OFFFFH, OA7EAH, OF7FFH OA7EAH, OF3FFH, OA0AAH, OFFFFH DW DW TILING\_DATA: OAAAAH, OFFFFH, OAAAAH, OFFFFH DW

; DW	4149H, 8622H, OC22H, 8400H	;TILE(4)-R	DW	0000H, 0000H, 0000H, 0000H	;
DW DW DW	OCA11H, 556EH, 90B1H, 0802H 0800H,0A755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H	;	DW DW DW	0000H, 0000H, 0000H, 0000H 0000H, 0000H, 0000H, 0000H 0000H, 0000H, 0000H, 0000H	;TILE(8)-R ; ;
DW DW DW	1C6EH,0AA96H, 5509H, 2C21H 6565H, 00CBH,0B481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H	;TILE(4)-G ; ;	D W D W D W	0000H, 0000H, 0000H, 0000H 8220H, 559DH,0A228H, 1095H 0000H, 8431H, 2220H,0D79DH	; ;TILE(8)-G ;
DW DW DW	5909H, 2429H, 061CH,0E510H 20D8H, 9603H, 0420H, 0400H 0D28BH, 2C4AH, 90A5H, 1041H	; TILE(4)-B	D W D W D W	2208H, 0000H, 8220H, 579DH 0A228H, 1091H, 0000H, 4210H 598CH, 2242H, 08C0H, 0842H	; ; ;TILE(8)-B
DW DW	0400H, 2D40H, 28ABH, 9606H 0020H, 0081H, 0A2AH, 4C43H	, , ,	DW DW DW	0B5ADH, 5946H, 1088H, 2842H 0842H,0B4ADH, 5942H, 2042H 0800H, 0842H,0B5ADH, 294AH	
DW DW DW	3118H, 1008H, 0700H, 0300H 0C30DH,0C003H, 4001H, 4270H 1718H, 0380H,0C100H, 4000H	;TILE(5)-R	; DW DW	OFFFFH,OFFFFH,OFFFFH,OFFFFH	;TILE(9)-R :
DW DW DW	0061H, 84F1H,0C639H, 6310H 7FFFH, 1008H,0FFFFH, 0300H 0F7FFH,0C003H,0FFF7H, 4270H	TILE(5)-G	DW DW DW	OFFFFH, OFFFFH, OFFFFH, OFFFFH OFFFFH, OFFFFH, OFFFFH, OFFFFH 7777H, OAAAAH, ODDDDH, OAAAAH	; ; ;TILE(9)-G
D W D W	OFFFFH, 0380H,OFFDFH, 4000H OF7FFH, 84F1H,OFFFFH, 6310H	, , , , , , ,	DW DW	7777H, OAAAAH, ODDDDH, OAAAAH 7777H, OAAAAH, ODDDDH, OAAAAH	
DW DW DW	3BBAH, OFFFFH, OAFAAH, OEFFFH OE3AFH, OFFEFH, OEAA3H, OFFFFH OBFBAH, OFFBFH, OEBBAH, OEFFFH	;TILE(5)-B	DW DW DW	7777H, OAAAAH, ODDDDH, OAAAAH ODDDDH, OOOOH, 5555H, OOOOH ODDDDH, OOOOH, 5555H, OOOOH	; ;TILE(9)-B ;
DW ; DW	OA2EBH,OFFFFH,OEEBBH,OFFFEH OAAAAH,OFFFFH,OBBBBH,OFFFFH	, ;TILE(6)-R	DW DW ;	ODDDDH, 0000H, 5555H, 0000H ODDDDH, 0000H, 5555H, 0000H	;;
DW DW DW	OAAAAH,OFFFFH,OBBBBH,OFFFFH OAAAAH,OFFFFH,OBBBBH,OFFFFH OAAAAH,OFFFFH,OBBBBH,OFFFFH	, , ,	DW DW DW	0F914H, 677CH,0E6F6H, 68EFH 0E5A5H,0F6E3H,0C86EH, 3AADH 0B472H, 0D2FH, 081EH,0FE29H	;TILE(A)-R ; ;
DW DW DW	5555H, 0000H, 4444H, 0000H 5555H, 0000H, 4444H, 0000H 5555H, 0000H, 4444H, 0000H	;TILE(6)-G ; ;	DW DW DW	0A9BBH,0F77EH, 3396H, 0D23H 0FFF7H,0E75FH,0F6FEH,0E8E7H 0E5F3H, 7DEBH,0FFEFH, 7FFFH	; ;TILE(A)-G ;
DW DW DW	5555H, 0000H, 4444H, 0000H 5555H, 0000H, 5555H, 0000H 5555H, 0000H, 5555H, 0000H	; TILE(6)-B	DW DW DW	OFB32H, 3D1FH, OE8EH,OFF5BH OFDBFH, 75FFH,OFFFEH, 7DEBH OB800H, 4574H,OA6A2H, 40C5H	; ; ;TILE(A)-B
DW DW	5555H, 0000H, 5555H, 0000H 5555H, 0000H, 5555H, 0000H	;	DW DW DW	0A0A0H,0F641H, 886AH, 308DH 0B422H, 050DH, 081EH, 5621H 0A8AAH,0D756H, 2296H, 0501H	
D W D W D W	0000H, 0000H, 0000H, 0200H 0600H, 0700H, 0700H, 0784H 06E4H, 07F8H, 03A0H, 01E0H	;TILE(7)-R ; :	; Dw Dw	4000H, 8008H, 0000H, 0000H 0000H, 0000H, 0000H, 2020H	;TILE(B)-R
DW DW DW	0000H, 0000H, 0000H, 0000H 112AH, 2240H, 48C8H, 9311H 2222H, 4244H, 8088H, 1A19H	TILE(7)-G	DW DW DW	0000H, 0280H, 0000H, 0200H 0000H, 0022H, 0000H, 0000H 0EAAAH,0D559H,0AAA8H, 5575H	; ; ;TILE(B)-G
DW DW DW	2102H, 6084H, 8050H, 9411H 2202H, 0644H, 8C0DH, 1012H 0000H, 0000H, 0000H, 0000H	; ; ;TILE(7)-B	DW DW DW	2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H	
D W D W	0400H, 0100H, 0500H, 0084H 0444H, 0228H, 0100H, 00A0H	;	DW DW	4000H,0AAA8H, 0000H, 2A88H 0000H, 028AH, 0050H,0AAE2H	;TILE(B)-B ;
	-5-			-6-	
DW DW ;	0101H,0AAAAH, 0000H,0A20AH 0000H,0A0A2H, 0000H,0AAA1H	;	DW DW	OB422H, O50DH, O81EH, 5621H OA8AAH,OD756H, 2296H, O501H	;
TILING_DATA_SS: DW DW DW DW	7EEFH, 34A5H, 1CG3H, 0C21H 0FFFFH, 6DEFH, 34A5H, 1CG3H 0C21H,0FFFFH, 6DEFH, 34A5H 1CG3H, 0C21H,0FFFFH,0EF7BH	;TILE(0)-R (0) ; ;	DW DW DW :	0EAAAH,0D559H,0AAA8H, 5575H 2AAAH, 1575H,0AAA2H, 7535H 0AAA2H, 57C5H,0BE8AH, 5615H 0A0BAH, 557FH,0AAB3H, 5546H	;TILE(B)-G (A)
; Dw Dw	6D6BH, 30A5H, 1C63H, 0C21H OFBFFH, 54A5H, 20A1H, 1421H	;TILE(1)-G (1) ;	DW DW DW	4000H, 0AAA8H, 0000H, 2A88H 0000H, 028AH, 0050H, 0AAE2H 0101H, 0AAAAH, 0000H, 0A20AH	;TILE(B)-B (B) ; ;
DW DW ;	OC21H,OFBFFH, 54A5H, 2421H 1021H, OC21H,OFBFFH, 14A5H		DW ;	0000H,0A0A2H, 0000H,0AAA1H	;
DW DW DW :	4149H, 8622H, 0C22H, 8400H OCA11H, 556EH, 90B1H, 0802H 0800H,0A755H, 69E2H, 3C02H 8442H, 0801H, 5BEBH, 4629H	;TILE(4)-R (2) ; ;	DEMO_INIT PROG ENDS END	ENDP	
DW DW DW	1C6EH,0AA96H, 5509H, 2C21H 6565H, 00C8H,0B481H, 4528H 20A5H, 74EAH, 2C86H, 14E3H 5909H, 2429H, 061CH,0E510H	:TILE(4)-G (3) ; ;			
DW DW DW DW	3118H, 1008H, 0700H, 0300H 0C30DH,0C003H, 4001H, 4270H 1718H, 0380H,0C100H, 4000H 0061H, 84F1H,0C639H, 6310H	TILE(5)-R (4)			
DW DW DW DW	112AH, 2240H, 48C8H, 9311H 2222H, 4244H, 8088H, 1A19H 2102H, 6084H, 8050H, 9411H 2202H, 0644H, 8C0DH, 1012H	:TILE(7)-G (5) ; ;			
DW DW DW DW	8220H, 559DH,0A228H, 1095H 0000H, 8431H, 2220H,0D79DH 2208H, 0000H, 8220H, 579DH 0A228H, 1091H, 0000H, 4210H	;TILE(8)-G (6) ; ;			
D W D W D W D W	598CH, 2242H, 08C0H, 0842H 085A0H, 5946H, 1088H, 2842H 0842H,084ADH, 5942H, 2042H 0800H, 0842H,085ADH, 294AH	:TILE(8)-B (7) ; ;			
DW DW DW DW	0F914H, 677CH,0E6F6H, 68EFH 0E5A5H,0F6E3H,0C86EH, 3AADH 0B472H, 0D2FH, 081EH,0FE29H 0A9BBH,0F77EH, 3396H, 0D23H	TILE(A)-R (8) ; ;			
DW DW	OB800H, 4574H,0A6A2H, 40C5H OA0A0H,0F641H, 886AH, 308DH	:TILE(A)-B (9) ;			

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_VAIT, MES_CL_VAIT_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: EAD1H:WORD, EAD1L: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		DMCL1: MOV MOV	DS:WORD PTR EADIH,4 DS:WORD PTR EADIL,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
EDMCL EDMCL	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	; _WAIT:			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,IFFH DS:WORD PTR MODIO,O DS:WORD PTR COM,903EH	;MAGETC=1FFH NON-CLIP ;MOD1=0,MOD0=0 ; <r_recfill> ;TL=0,SS=1,WL=1,WR=1</r_recfill>
Directo_c	MOV MOV	AX,0 BX,749	;X=0 ;Y=749	RET		;FAST=1
DMCL0_C	MOV MOV CALL END:	CX,1119 DX,-755 DMCLO_C_FREE	;DX=1119 ;DY=-755	; DMCLO_C_FREE: TEST JNZ	DS:WORD PTR STATUS,1 DMCLO_C_FREE	;CHECK IF PPBUSY=1/0
	RET;			MOV MOV	DS:WORD PTR PTNCNT,OFFFFH DS:WORD PTR PMAX,4	;PTNCNT=OFFFFH ;PMAX=4
DMCLO_B	: TEST JNZ	DS:WORD PTR STATUS,1 DMCLO_B	;CHECK IF PPBUSY=1/0	MOV MOV MOV	DS:WORD PTR PLANES,7 DS:WORD PTR MAGETC,1FFH DS:WORD PTR MOD10,0	;PLANES=7 ;MAGETC=1FFH NON-CLIP ;MOD1=0,MOD0=0
DMCLO:	MOV MOV	DS:WORD PTR PTNCNT,0 DS:WORD PTR PMAX,4	;PTNCNT=0 ;PMAX=4	MOV MOV MOV	DS:WORD PTR X,AX DS:WORD PTR Y,BX DS:WORD PTR DXX,CX	; X=(?) ; Y=(?) ;DX=(?)
	MOV	DS:WORD PTR PLANES,7	;PLANES=7	MOV	DS:WORD PTR DY,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 DEC	WAIT_1 AX	;LOOP UNTIL CX=0			
WAIT EN	JNZ	WAIT	;LOOP UNTIL AX=0			
;	;					
	WAIT FIL MOV JMP	L: CS:WORD PTR WORK_7,10 MES_CL_WAIT_2				
MES_CL_ MES_CL_	MOV	CS:WORD PTR WORK_7,45				
1123_00_	CALL TEST JNZ CALL	MESSAGE_WAIT CS:WORD PTR FLAG,4 MES_CL_WAIT_1 DMCLO_C_WAIT				
MES CL_	RET WAIT_1: CALL RET	DMCL1_C_WAIT				
MESSAGE	TEST JZ TEST JNZ CALL MOV	CS:BVTE PTR FLAG,1 MESSAGE_WAIT_1 CS:BVTE 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
	; 	(a) ( a) ( )			MOV	DS:WORD PTR PMAX,4	;PMAX=4
	PUBLIC	GLIOMAIN			MOV	DS:WORD PTR PLANES,7	;PLANES=7
	, EVTDN	STATUS:WORD, PTNCNT:WORD, PMA	V. UODD DI ANES. UODD		MOV MOV	DS:WORD PTR MAGETC,1FFH DS:WORD PTR MOD10,0	;MAGETC=1FFH NON-CLIP ;MOD1=0,MOD0=0
	EXTRN EXTRN	MAGETC:WORD, MOD10: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, D			MOV	DS:WORD PTR DXX,1119	;DX=1119
	EXTRN	XCLMIN:WORD, YCLMIN:WORD, XCL			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, d	ADORG:BYTE				;TL=0,SS=1,WL=1,WR=1
	EXTRN	WORK_1:WORD,					;FAST=1
	;				JMP	G_POP_EXE	
GLIOMAI	N	PROC NEAR			;		
	;			LINE:	MON	DV DC:[DD]	· VC-(9)
:	C C D A D U	110/010 >			MOV MOV	BX,DS:[BP] CX,DS:[BP+2]	;XS=(?)
:	S GRAFN	L10/B10 >			XOR	CX,05:LBF+2] CX,0FFFFH	
:					INC	cx	;YS=(?)
,	:				MOV	S1,DS:[BP+4]	;XE=(?)
	PUSH	BX			MOV	DI,DS:[BP+6]	
	CALL	G_PUSH_EXE			XOR	D1,OFFFFH	
	CALL	INIT_SEG			INC	DI	;YE=(?)
	CMP	AH,5			MOV	AX,DS:[BP+8]	
	JZ	GCLS	;"GCLS"		MOV	DX,ES	
	CMP	AH,7	***		MOV	DS, DX	
	JZ		;"LINE","REC","REC_FILL"		CMP	AH,O	
	CMP JZ	AH,8 CRL	;"CRL","ELPS","CRL_FILL"		JZ CMP	G_LINE_EXE	
	J 4	ONE.	;"ELPS_FILL"		JZ	AH,1 G_REC_EXE	
	CMP	AH,9	,		CMP	AH,2	
	JZ	PAINT	;"PAINT"		JZ	G_RECFILL_EXE	
	CMP	AH, ODH			JMP	G POP EXE	
	JΖ	COPY	;"COPY"		;	1.22 - 1937 - 201937	
	JMP	G_POP_EXE		G_LINE_			
	;				TEST	DS:WORD PTR STATUS,1	
CRL:					JNZ	G_LINE_EXE	;CHECK IF PPBUSY=1/0
	JMP	CRL_1			MOV	CS:WORD PTR WORK_1,1800H	; <a_line_m1> PL=0</a_line_m1>
0.000	;						;   P=0, PXEN=0
PAINT:	140	DALWE 1			CALL	COLOR_CAL	
	JMP	PAINT_1			MOV MOV	DS:WORD PTR X,BX DS:WORD PTR Y,CX	; X=(?) ; Y=(?)
COPY:	,				MOV	DS:WORD PTR T,CX DS:WORD PTR XE,SI	; Y=(?); XE=(?)
corr.	JMP	COPY_1			MOV	DS:WORD PTR YE,DI	;YE=(?)
	:	001121			JMP	GENERAL_CP	
GCLS:	·				;		
	MOV	AX,ES		G_REC_E	XE:		
	MOV	DS,AX			MOV	CS:WORD PTR WORK_1,4800H	; <a_rec> PXEN=0</a_rec>
						0	
		-1-				-2-	
	100					-2-	
	JMP	-1- REC_RECFILL_EXE		CRL_2:	MOV		CRIN CEED IPED
G RECEI	;			CRL_2:	MOV	-2- CS:WORD PTR WORK_1,5000H	; <crl> CF=0,1P=0 :PXFN=0</crl>
G RECFI	; LL EXE:	REC_RECFILL_EXE	: <a c="" recfill=""></a>	CRL_2:		CS:WORD PTR WORK_1,5000H	; <crl> CF=0,1P=0 ;PXEN=0</crl>
G RECFI	;		; <a_recfill_c> ;TL=0.SS=1.WL=1.WR=1</a_recfill_c>	CRL_2:	MOV TEST JZ		
G RECFI	; LL EXE:	REC_RECFILL_EXE		CRL_2:	TEST	CS:WORD PTR WORK_1,5000H AH,20H	
	; LL EXE: MOV FILL EXE	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH ::	;TL=0,SS=1,WL=1,WR=1	CRL_2: CRL_CRL	TEST JZ MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH S:	;PXEN=0
	; LL EXE: MOV FILL EXE TEST	REC_RECFILL_EXE 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_EXE TEST	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH I DS:WORD PTR STATUS,1	:PXEN=0 : <crlfill> TL=0,SS=1</crlfill>
	; LL EXE: MOV FILL EXE TEST JNZ	REC_RECFILL_EXE 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_EXE TEST JNZ	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH S: DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE	;PXEN=0
	; MOV FILL EXE: TEST JNZ CALL	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH :: DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL	;TL=O,SS=1,WL=1,WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0		TEST JZ MOV FILL_EXE TEST JNZ CALL	CS:WORD PTR WORK_1,5000H AH.20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS: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	REC_RECFILL_EXE 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 EXE TEST JNZ CALL MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH :: 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	REC_RECFILL_EXE 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 EXE TEST JNZ CALL MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH S: 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=(?) ;Y=(?)</crlfill>
	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR X,SX DS:WORD PTR X,SX	;TL=0,SS=1,WL=1,WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?)		TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH CS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,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	REC_RECFILL_EXE 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 EXE TEST JNZ CALL MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH S: 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=(?) :Y=(?)</crlfill>
	; NOV FILL EXE: TEST JNZ CALL MOV MOV MOV	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR X,SX DS:WORD PTR X,SX	;TL=0,SS=1,WL=1,WR=1 ;FAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?)	CRL_CRL	TEST JZ MOV FILL_EXE TEST JNZ CALL MOV MOV JMP ;	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH CS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC,BX DS:WORD PTR YC,CX DS:WORD PTR DXX,SI	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?)</crlfill>
REC REC	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR X,BX DS:WORD PTR X,BX DS:WORD PTR XS,SI DS:WORD PTR YS,DI DS:WORD PTR PTNCNT,0FFFH	;TL=0,SS=1,WL=1,WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH		TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV MOV JMP ; ; : MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS: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]	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?)</crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV CP: MOV MOV MOV MOV MOV	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,CX DS:WORD PTR Y,CX DS:WORD PTR YS,D1 DS:WORD PTR YS,D1 DS:WORD PTR PMAX,4 DS:WORD PTR PMCNT_OFFFFH DS:WORD PTR PMOID,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 EXE TEST JNZ CALL MOV MOV JMP ; : MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH 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]	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?) :DXX=(?)</crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR PTNCNT_OFFFFH DS:WORD PTR PTNCNT_OFFFFH DS:WORD PTR MODIO,1 AX,CS:WORD PTR WORK_1	<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</pre>	CRL_CRL	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; : MOV MOV XOR	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH 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,DS:[BP] CX,DS:[BP]+2] CX,OFFFH	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;DXX=(?) ;X=(?)</crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	REC_RECFILL_EXE 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 PMAX,4 DS:WORD PTR PTNCNT,0FFFH DS:WORD PTR WORK,1 DS:WORD PTR COM,AX	;TL=0,SS=1,WL=1,WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; X=(?) ; Y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH	CRL_CRL	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV MOV JMP ; ; ; ; MOV MOV XOR INC	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS: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] CX,OFFFFH CX	: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	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR PTNCNT_OFFFFH DS:WORD PTR PTNCNT_OFFFFH DS:WORD PTR MODIO,1 AX,CS:WORD PTR WORK_1	<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</pre>	CRL_CRL	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; : MOV MOV XOR NOV XOR NOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH 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] CX.0FFFFH CX AX.DS:[BP+4]	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;DXX=(?) ;X=(?)</crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	REC_RECFILL_EXE 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 PMAX,4 DS:WORD PTR PTNCNT,0FFFH DS: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 ;PTNCNT=0FFFFH ;M0D1=0,M0D0=1</pre>	CRL_CRL	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; MOV XOR INC MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH 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,DS:[BP]2] CX,OFFFH 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	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX DS:WORD PTR Y,CX DS:WORD PTR YS,DI 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 G_POP_EXE	<pre>;TL=0.SS=1.WL=1.WR=1 ;PAST=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 FILL EXE TEST JNZ CALL MOV MOV MOV JMP ; : MOV MOV NOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH 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] 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: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	REC_RECFILL_EXE 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 PMAX,4 DS:WORD PTR PTNCNT,0FFFH DS: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 ;PTNCNT=0FFFFH ;M0D1=0,M0D0=1</pre>	CRL_CRL	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV MOV JMP ; : MOV MOV NOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH 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,DS:[BP]2] CX,OFFFH 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 JMP ; MOV	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL 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 MOID,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP]	<pre>;TL=0.SS=1.WL=1.WR=1 ;PAST=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 FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; ; MOV MOV MOV MOV MOV MOV S:	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH 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	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :DXX=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0</crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV JMP ; MOV MOV XOR INC	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX DS:WORD PTR Y,CX DS:WORD PTR YS,D1 DS:WORD PTR YS,D1 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR MOID,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP] CX,OFFFFH CX	;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=(?) ;YC=(?)</com.flag>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV MOV XOR INC XOR INC XOR INC NOV MOV MOV MOV TEST	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH 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,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint></paint></crlfill>
REC REC	; MOV FILL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC.RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX 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 MORK_1 DS:WORD PTR MORK_1	<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=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; ; NOV MOV XOR INC XOR INC NOV MOV MOV TEST JNZ MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH CRL_CRLFILL_EXE COLOR_CAL 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+2] CX,DS:[BP+2] CX,DS:[BP+2] CX,OFFFH CX AX,DS:[BP+4] DX,ES DS,DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR WORK_1,6830H	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :DXX=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0</crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV _CP: _CP: _CP: _CP: _CP: _CP: _CP: _CP:	REC.RECFILL_EXE 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,BX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI 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,DS:[BP] CX,DS:[BP]+2] CX,OFFFFH CX S1,DS:[BP+4] D1,DS:[BP+6]	;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=(?) ;YC=(?)</com.flag>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE 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	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH CS:WORD PTR WORK_1.503CH CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR XC.BX DS:WORD PTR XC.BX DS:WORD PTR VC.CX DS:WORD PTR DXX.SI GENERAL_CP BX.DS:[BP] CX.DS:[BP+2] CX.OFFFH CX AX.DS:[BP+4] DX.ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR WORK_1.6830H PAINT_COL	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :DXX=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1</paint></crlfill>
REC REC	; MOV FILL EXE: TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV JMP ; MOV MOV XOR INC MOV MOV XOR INC MOV MOV	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX DS:WORD PTR Y,CX DS:WORD PTR YS,D1 DS:WORD PTR YS,D1 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR NOID,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP] CX,OFFFFH CX S1,DS:[BP+4] D1,DS:[BP+8] AX,DS:[BP4]	<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=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; ; MOV MOV XOR NOV XOR NOV SOV TEST JNZ MOV CALL MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CRL_CRLFILL_EXE COLOR_CAL 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,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 XCLMIN,0	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :DXX=(?) :X=(?) :AH&gt;B.COL, AL&gt;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 XOR INC MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC.RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX 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 MORK_1 DS:WORD PTR MORK_1	<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=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; ; MOV XOR INC XOR INC XOR INC XOR INC XOR INC XOR CALL MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS:WORD PTR WORK_1,503CH 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,DS:[BP+2] CX,DS:[BP+2] CX,OFFFH 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 DS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,-749	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;V=(?) ;DXX=(?) :X=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=0 ;VCLMIN=-749</paint></crlfill>
REC REC General	; LL EXE: MOV FILL EXE TEST JNZ CZ CZ CZ CZ CZ MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE 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,BX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR PTNCNT,OFFFFH 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 MODIO,1 AX,CS:WORD PTR MODIO,1 AX,SS:WORD PTR MORK_1 DS:WORD PTR MORK_1 DS:WORD PTR MORK_1 DS:WORD PTR MORK_1 DS:WORD PTR SCHART WORK_1 DS:WORD PTR SCHART WORK_1 SCHART WOR	<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=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV XOR INC MOV XOR INC MOV XOR INC MOV MOV CALL MOV CALL MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH CS:WORD PTR WORK_1.503CH CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR XC.BX 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 WORK_1.6830H PAINT_COL DS:WORD PTR XCLMIN.0 DS:WORD PTR XCLMIN.0 DS:WORD PTR XCLMIN.749 DS:WORD PTR YCLMIN.749 DS:WORD PTR YCLMIN.749 DS:WORD PTR XCLMAX.1119	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :DXX=(?) :X=(?) :AH&gt;B.COL, AL&gt;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 General	; MOV FILL EXE: MOV TEST JNZ CALL MOV MOV MOV MOV MOV MOV JMP ; MOV MOV JMP ; MOV MOV SOR INC MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX DS:WORD PTR Y,CX DS:WORD PTR YS,D1 DS:WORD PTR YS,D1 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR MOID,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP] CX,OS:EBP+2] CX,OFFFFH CX SI,DS:[BP+4] DX,ES DS,DX SI,D1	<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=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV S: INC MOV MOV XOR INC MOV MOV CALL MOV MOV CALL MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS:WORD PTR WORK_1,503CH 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,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR YORK_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,0	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :X=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PM0D=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=749 :XCLMAX=1119 :YCLMAX=0</paint></crlfill>
REC REC General	; LL EXE: MOV FILL EXE TEST JNZ CZ CZ CZ CZ CZ MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC.RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX 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 PMONL,1 DS:WORD PTR MORK_1 DS:WORD PTR MORK_1	<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=(?) ;YC=(?) ;DX=(?)</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV XOR INC MOV XOR INC MOV XOR INC MOV MOV CALL MOV CALL MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR STATUS,1 CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DS:WORD PTR VC,CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP+2] CX,DS:[BP+2] CX,DS:[BP+2] CX,OFFFH 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 YCLMIN,-749 DS:WORD PTR YCLMAX,0 DS:WORD PTR MCGEC,OFFH	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :DXX=(?) :X=(?) :AH&gt;B.COL, AL&gt;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 General	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC_RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC_RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX DS:WORD PTR Y,CX DS:WORD PTR YS,D1 DS:WORD PTR YS,D1 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR MOID,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR COM,AX G_POP_EXE BX,DS:[BP] CX,OS:EBP+2] CX,OFFFFH CX SI,DS:[BP+4] DX,ES DS,DX SI,D1	<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=(?) ;YC=(?) ;DX=(?) ;DY=(?) ;</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; ; ; MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS:WORD PTR WORK_1,503CH 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,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES DS.DX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR YORK_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,0	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;Y=(?) ;AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMAX=119 :YCLMAX=0 :MABTC=0FFH</paint></crlfill>
REC REC	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC.RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX 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 PMONL,1 DS:WORD PTR MORK_1 DS:WORD PTR MORK_1	<pre>;TL=0.SS=1.WL=1.WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;)DY=(?) ;</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV XOR INC MOV XOR INC MOV MOV TEST JNZ CALL MOV MOV CALL MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS:WORD PTR WORK_1,503CH CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR XC.BX DS:WORD PTR VC.CX DS:WORD PTR DXX,SI GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,DS:[BP+2] CX,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 DS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,0 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,1119 DS:WORD PTR MAGETC,0FFH DS:WORD PTR XBX	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?) :X=(?) :X=(?) :AH&gt;B.COL, AL&gt;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	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE 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 YS,DI DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR MOILO,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR MOILO,1 AX,CS:WORD PTR WORK_1 DS:WORD PTR MOILO,1 CX,DS:[BP] CX,DS:[BP] CX,DS:[BP+4] DI,DS:[BP+6] AX,DS:[BP+8] DX,ES DS,DX SI,DI CRL_2 CS:WORD PTR WORK_1,5COOH AH,2OH ELPSFILL_EXE	<pre>;TL=0.SS=1.WL=1.WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;DY=(2) ; ;</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; ; ; MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS:WORD PTR WORK_1,503CH 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 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 YCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCX	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?) :X=(?) :X=(?) :AH&gt;B.COL, AL&gt;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 JMP ; MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	REC_RECFILL_EXE 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,D1 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR MOID,1 AX,CS: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+4] 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 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;)DY=(?) ;</com.flag></pre>	CRL_CRL PAINT_1	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; : TEST JNZ MOV MOV MOV CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH S:WORD PTR WORK_1,503CH 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.OFFFH CX 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 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 YCLMIN.749 DS:WORD PTR YCLMIN.740 DS:WORD PTR YCLMIN.740 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCX GENERAL_CP	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;Y=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=749 :XCLMIN=0 :YCLMIN=749 :XCLMIN=0 :YCL</paint></crlfill>
REC REC GENERAL CRL_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC.RECFILL_EXE COLOR_CAL DS:WORD PTR Y,DX 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 PMOID,1 AX,CS: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:UORD PTR VORK_1 DS:EBP+2] CX,DFFFFH CX SI,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 AH,20H ELPS_ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH	<pre>;TL=0.SS=1.WL=1.WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;DY=(2) ; ;</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; ; MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS:WORD PTR WORK_1,503CH CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR XC.BX DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR VC.CX DS:WORD PTR VC.CX DS:WORD PTR VC.CX DS:WORD PTR VC.CX DS:WORD PTR VC.CX DS:WORD PTR VC.CX DS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2CL 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 YCLMIN,0 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCX GENERAL_CP	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 ;X=(?) ;Y=(?) ;X=(?) ;X=(?) :X=(?) :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMAX=0 :MAGETC=0FFH :X=(?) ;X=(?)</paint></crlfill>
REC REC GENERAL CRL_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE CS:WORD PTR WORK_1,8C3CH DS:WORD PTR STATUS,1 REC.RECFILL_EXE COLOR_CAL DS:WORD PTR Y,BX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PTNCNT,OFFFFH DS:WORD PTR MORK_1 DS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:SIORD PTR WORK_1 DS:EBP-1 CX,DS:[BP+4] DI,DS:EBP+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 STATUS,1	<pre>;TL=0.SS=1.WL=1.WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;DY=(2) ; ;</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV XOR INC MOV MOV MOV MOV CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH CS:WORD PTR WORK_1.503CH 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.DS:[BP+2] CX.DS:[BP+2] CX.DS:[BP+4] DX.ES DS.DX DS:WORD PTR WORK_1.6830H PAINT_COL DS:WORD PTR YCLMIN.0 DS:WORD PTR YCLMIN.0 DS:WORD PTR YCLMIN.749 DS:WORD YCLMIN.749 DS:WORD YCMIN.749 DS:WORD YCMIN.749 DS:WORD YCMIN.749 DS:WORD YCMIN.749 DS:WORD YCMIN.749 DS:WORD YCMIN.749 DS:WORD YCMIN.749 DS:WORD YCMIN.749 DS:WORD YCMIN.749 DS:WORD YCMIN.749 DS:	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;Y=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PM0D=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMIN=0 :YCLMIN=749 :XCLMAX=1119 :YCLMIN=0 :MAGETC=0FFH :X=(?) ;Y=(?)</paint></crlfill>
REC REC GENERAL CRL_1:	; MOV FILL EXE: MOV TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	REC_RECFILL_EXE 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,BX DS:WORD PTR YS,D1 DS:WORD PTR YS,D1 DS:WORD PTR PTNCNT_OFFFFH 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,OFFFFH CX SI,DS:[BP+4] D1,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 STATUS,1 ELPSELPSFILL_EXE	<pre>;TL=0.SS=1.WL=1.WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; x=(?) ; v=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;DY=(2) ; ;</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; ; TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH 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 YCLMIN.0 DS:WORD PTR YCLMIN.0 DS:WORD PTR YCLMIN.749 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCLMAX.0 DS:WORD PTR YCLMAX.0 DS:WORD PTR Y.BX DS:WORD PTR Y.CX GENERAL_CP	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;Y=(?) ;X=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PM0D=0, TL=0, SS=1 :XCLMIN=0 :Y</paint></crlfill>
REC REC GENERAL CRL_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC.RECFILL_EXE COLOR_CAL DS:WORD PTR Y,CX 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 PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR MOIL,1 AX,CS: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,DFFFFH CX SI,DS:[BP+4] DI,DS:[BP+6] AX,DS:[BP+6] AX,DS:[BP+6] AX,DS:[BP+6] AX,DS:[BP+6] AX,DS:[BP+6] CRL_2 CS:WORD PTR WORK_1,5C3CH AH,20H ELPS_ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR VORK_1,5C3CH XE: DS:WORD PTR VORK_1,5C3CH XE: DS:WORD PTR STATUS,1 ELPSFILL_EXE COLOR_CAL	<pre>:TL=0.SS=1.WL=1.WR=1 ;PAST=1 :CHECK IF PPBUSY=1/0 : x=(?) ; y=(?) ;XS=(?) ;YS=(?) :PMAX=4 :PTNCNT=0FFFFH ;MOD1=0.MOD0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;DY=(?) ; ; :<elps> CF=0.1P=0 ;PXEN=0 ;<elpsf1ll> TL=0.SS=1 ;CHECK IF PPBUSY=1/0</elpsf1ll></elps></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; ; MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS:WORD PTR WORK_1,503CH CS: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 VC.CX DS:WORD PTR X51 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR YCLMIN,0 DS:WORD PTR	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 ;X=(?) ;Y=(?) ;X=(?) ;X=(?) :X=(?) :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMAX=0 :MAGETC=0FFH :X=(?) ;X=(?)</paint></crlfill>
REC REC GENERAL CRL_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC RECFILL_EXE 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 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 WORK_1 DS:WORD PTR VORK_1 DS:WORD PTR VORK_1 S:WORD PTR VORK_1,5COOH AH,2OH ELPS_ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR YORK_1,5C3CH XE: DS:WORD PTR STATUS,1 ELPS_ELPSFILL_EXE COLOR_CAL DS:WORD PTR X,BX	<pre>;TL=0,SS=1,WL=1,WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; x=(?) ; y=(?) ;XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0,M0D0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;DY=(?) ; ; ;<elps> CF=0,IP=0 ;PXEN=0 ;<elpsfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;XC=(?)</elpsfill></elps></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV MOV SOR INC MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1.503CH CS: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 VC,CX DS:WORD PTR VC,1,6830H PAINT_C0L DS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,0 DS:WORD PTR XCLMIN,-749 DS:WORD PTR YCLMAX,1119 DS:WORD PTR YCLMAX,0 DS:WORD PTR YC,CX GENERAL_CP BX,DS:[BP] BX,2 CX,DS:[BP] BX,2 CX,DS:[BP] BX,2 CX,DFFFH	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) :Y=(?) :X=(?) :X=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PMOD=0, TL=0, SS=1 :XCLMIN=-749 :XCLMIN=-749 :XCLMAX=119 :YCLMAX=0 :MAGETC=0FFH :X=(?) :BODY&gt;CHARACTRE FACE :BODY&gt;CHARACTRE FACE</paint></crlfill>
REC REC GENERAL CRL_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC.RECFILL_EXE COLOR_CAL DS:WORD PTR Y,CX 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 PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR MOIL,1 AX,CS: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,DFFFFH CX SI,DS:[BP+4] DI,DS:[BP+6] AX,DS:[BP+6] AX,DS:[BP+6] AX,DS:[BP+6] AX,DS:[BP+6] AX,DS:[BP+6] CRL_2 CS:WORD PTR WORK_1,5C3CH AH,20H ELPS_ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR VORK_1,5C3CH XE: DS:WORD PTR VORK_1,5C3CH XE: DS:WORD PTR STATUS,1 ELPSFILL_EXE COLOR_CAL	<pre>:TL=0.SS=1.WL=1.WR=1 ;PAST=1 :CHECK IF PPBUSY=1/0 : x=(?) ; y=(?) ;XS=(?) ;YS=(?) :PMAX=4 :PTNCNT=0FFFFH ;MOD1=0.MOD0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;DY=(?) ; ; :<elps> CF=0.1P=0 ;PXEN=0 ;<elpsf1ll> TL=0.SS=1 ;CHECK IF PPBUSY=1/0</elpsf1ll></elps></com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV JMP ; ; ; ; MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS:WORD PTR WORK_1,503CH CS: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 VC.CX DS:WORD PTR X51 PAINT_2 CS:WORD PTR STATUS,1 PAINT_2 CS:WORD PTR YCLMIN,0 DS:WORD PTR	<pre>;PXEN=0 ;<crlfill> TL=0,SS=1 ;CHECK IF PPBUSY=1/0 ;X=(?) ;Y=(?) ;X=(?) ;AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 ;<paint> ;PM0D=0, TL=0, SS=1 :XCLMIN=0 ;YCLMIN=-749 ;XCLMIN=0 ;YC</paint></crlfill></pre>
REC REC GENERAL CRL_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC RECFILL_EXE 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 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 YORK_1 DS:WORD PTR WORK_1 DS:WORD PTR WORK_1 DS:WORD PTR VORK_1 DS:WORD PTR VORK_1 SS:UBP+4] D1,DS:[BP+4] D1,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 YORK_1,5C3CH XE DS:WORD PTR YC,CX DHUCCAL 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 ;PAST=1 :CHECK IF PPBUSY=1/0 : x=(?) ; y=(?) :SS=(?) ;YS=(?) :PMAX=4 :PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :XC=(?) :VC=(?) :DX=(?) :DX=(?) :DX=(?) :CHECK IF PPBUSY=1/0 ;XC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?)</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV MOV SOR INC MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CS:WORD PTR WORK_1,503CH 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,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,119 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,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,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,119 DS:WORD PTR YCLMAX,0 DS:WORD PTR Y	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;Y=(?) :X=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PM0D=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMAX=1119 :YCLMAX=0 :MAGETC=0FFH :X=(?) :Y=(?) :X=(?) :X=(?) :DDY&gt;CHARACTER FACE :BODY&gt;CHARACTER FACE :Y=(?)</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	REC_RECFILL_EXE 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,D1 DS:WORD PTR YS,D1 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR VORK_1 DS:WORD PTR PMAX,4 DS:WORD PTR VORK_1 DS:WORD PTR VORK_1 DS:WORD PTR COM.AX G_POP_EXE BX,DS:[BP] CX,DS:[BP+2] CX,OFFFH CX SI,DS:[BP+6] AX,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 VORK_1,5C3CH XE: DS:WORD PTR STATUS,1 ELPS_ELPSFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DHUW_CAL DS:WORD PTR DH,BX DS:WORD PTR PTR VAX	<pre>;TL=0.SS=1.WL=1.WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; x=(?) ; V=(?) :XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;DY=(?) ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ;</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV MOV SCALL MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH S:WORD PTR WORK_1,503CH CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,6X DS:WORD PTR YC,6X DS:WORD PTR YC,6X DS:WORD PTR YC,6X DS:WORD PTR DXX,51 GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES 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,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR XCLMAX,1119 DS:WORD PTR XCLMAX,1119 DS:WORD PTR YCLMIN,749 DS:WORD PTR XCLMAX,119 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR Y,6X GENERAL_CP	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;Y=(?) :X=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PM0D=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMAX=1119 :YCLMAX=0 :MAGETC=0FFH :X=(?) :Y=(?) :X=(?) :X=(?) :DDY&gt;CHARACTER FACE :BODY&gt;CHARACTER FACE :Y=(?)</paint></crlfill>
REC REC GENERAL CRL_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC.RECFILL_EXE COLOR_CAL DS:WORD PTR Y,DX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR PMXX,4 DS:WORD PTR PMXX,4 DS:WORD PTR PMXX,4 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:WORD PTR WORK_1 DS:UORD PTR VORK_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_1,5C3CH AH,20H ELPS_ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR VORK_1,5C3CH XE: DS:WORD PTR VORK_1,5C3CH XE: DS:WORD PTR Y,CX DS:WORD PTR Y,CX DD:WORD PTR V,DI DS:WORD PTR DY,AX DS:WORD PTR DY,AX	<pre>:TL=0.SS=1.WL=1.WR=1 ;PAST=1 :CHECK IF PPBUSY=1/0 : x=(?) ; y=(?) :SS=(?) ;YS=(?) :PMAX=4 :PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> :XC=(?) :VC=(?) :DX=(?) :DX=(?) :DX=(?) :CHECK IF PPBUSY=1/0 ;XC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?) ;YC=(?)</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV MOV SOR INC MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC.CX DS:WORD PTR XC.BX DS:WORD PTR XCLMIN.0 DS:WORD PTR YCLMIN.749 DS:WORD PTR YCLMIN.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.0 DS:WORD PTR Y.CX GENERAL_CP	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;Y=(?) :X=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PM0D=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMAX=1119 :YCLMAX=0 :MAGETC=0FFH :X=(?) :V=(?) :X=(?) :DDY&gt;CHARACTER FACE :BODY&gt;CHARACTER FACE :Y=(?)</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	REC_RECFILL_EXE 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,D1 DS:WORD PTR YS,D1 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR VORK_1 DS:WORD PTR PMAX,4 DS:WORD PTR VORK_1 DS:WORD PTR VORK_1 DS:WORD PTR COM.AX G_POP_EXE BX,DS:[BP] CX,DS:[BP+2] CX,OFFFH CX SI,DS:[BP+6] AX,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 VORK_1,5C3CH XE: DS:WORD PTR STATUS,1 ELPS_ELPSFILL_EXE COLOR_CAL DS:WORD PTR YC,CX DHUW_CAL DS:WORD PTR DH,BX DS:WORD PTR PTR VAX	<pre>;TL=0.SS=1.WL=1.WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; x=(?) ; V=(?) :XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;DY=(?) ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ;</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV MOV SCR INC MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR YC.CX DS:WORD PTR XCLBX CX.DS:[BP] CX.DS:[BP] CX.DS:[BP+2] CX.OFFFFH CX AX.DS:[BP+4] DX.ES DS.WORD PTR XCLMIN.0 DS:WORD PTR XCLMIN.0 DS:WORD PTR XCLMIN.0 DS:WORD PTR YCLMIN.749 DS:WORD PTR YCLMIN.749 DS:WORD PTR YCLMIN.749 DS:WORD PTR YCLMAX.0 DS:WORD PTR Y.CX GENERAL_CP BX.DS:[BP] BX.2 CX.0FFFFH CX AX.DS:[BP+4] DI.DS:[BP+4] DI.DS:[BP+4] DX.ES DS.DX	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;Y=(?) :X=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PM0D=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMAX=1119 :YCLMAX=0 :MAGETC=0FFH :X=(?) :V=(?) :X=(?) :DDY&gt;CHARACTER FACE :BODY&gt;CHARACTER FACE :Y=(?)</paint></crlfill>
REC REC GENERAL CRL_1:	; LL EXE: MOV FILL EXE TEST JNZ CALL MOV MOV MOV MOV MOV MOV MOV MOV	REC.RECFILL_EXE CS:WORD PTR WORK_1,8C3CH CS:WORD PTR STATUS,1 REC.RECFILL_EXE COLOR_CAL DS:WORD PTR Y,DX DS:WORD PTR Y,CX DS:WORD PTR YS,DI DS:WORD PTR YS,DI DS:WORD PTR PMXX,4 DS:WORD PTR PMXX,4 DS:WORD PTR PMXX,4 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:WORD PTR WORK_1 DS:UORD PTR VORK_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_1,5C3CH AH,20H ELPS_ELPSFILL_EXE CS:WORD PTR WORK_1,5C3CH XE: DS:WORD PTR VORK_1,5C3CH XE: DS:WORD PTR VORK_1,5C3CH XE: DS:WORD PTR Y,CX DS:WORD PTR Y,CX DD:WORD PTR V,DI DS:WORD PTR DY,AX DS:WORD PTR DY,AX	<pre>;TL=0.SS=1.WL=1.WR=1 ;PAST=1 ;CHECK IF PPBUSY=1/0 ; x=(?) ; V=(?) :XS=(?) ;YS=(?) ;PMAX=4 ;PTNCNT=0FFFFH ;M0D1=0.M0D0=1 ;<com.flag> ;XC=(?) ;YC=(?) ;DX=(?) ;DX=(?) ;DY=(?) ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ;</com.flag></pre>	CRL_CRL PAINT_1 PAINT_2	TEST JZ MOV FILL EXE TEST JNZ CALL MOV MOV MOV SINP ; ; ; TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	CS:WORD PTR WORK_1,5000H AH,20H CRL_CRLFILL_EXE CS:WORD PTR WORK_1,503CH S:WORD PTR WORK_1,503CH CRL_CRLFILL_EXE COLOR_CAL DS:WORD PTR YC,6X DS:WORD PTR YC,6X DS:WORD PTR YC,6X DS:WORD PTR YC,6X DS:WORD PTR DXX,51 GENERAL_CP BX,DS:[BP] CX,DS:[BP+2] CX,OFFFFH CX AX,DS:[BP+4] DX,ES 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,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR XCLMAX,1119 DS:WORD PTR XCLMAX,1119 DS:WORD PTR YCLMIN,749 DS:WORD PTR XCLMAX,119 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMIN,749 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR YCLMAX,0 DS:WORD PTR Y,6X GENERAL_CP	:PXEN=0 : <crlfill> TL=0,SS=1 :CHECK IF PPBUSY=1/0 :X=(?) ;Y=(?) :X=(?) :X=(?) :AH&gt;B.COL, AL&gt;D.COL :CHECK IF PPBUSY=1/0 :<paint> :PM0D=0, TL=0, SS=1 :XCLMIN=0 :YCLMIN=-749 :XCLMAX=1119 :YCLMAX=0 :MAGETC=0FFH :X=(?) :V=(?) :X=(?) :DDY&gt;CHARACTER FACE :BODY&gt;CHARACTER FACE :Y=(?)</paint></crlfill>

	1000 CT 10						
	SHL	- AL,1			MOV	ES:WORD PTR EADORGH,0	;EADORGH=0
	SHL	AX,1			MOV	ES:BYTE PTR dADORG,0	;dADORG=0
	SHL	AX,1			MOV	BP,100H	
	RCL	DX,1			POP	AX	
	SHL	AX,1			RET	nA	
	RCL	DX,1					
	SHL	AX,1		0.000.000	;		
				COLOR_C	CAL:		
	RCL	DX,1			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	:				RET	borworke i na realized fina	,, ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,
	TEST	DS:WORD PTR STATUS,1					
	JNZ	COPY_2	;CHECK IF PPBUSY=1/0		;		
	MOV			COL:			
		DS:WORD PTR EAD2L,AX	;EAD2L=(?)		DB	0, 4, 1, 5, 2, 6, 3, 7	;FIT COLOR BITS
	MOV	DS:WORD PTR EAD2H,DX	;EAD2H=(?)		;		
	MOV	DS:BYTE PTR dAD2,0	;dAD2=0	PAINT_C	:0L:		
	MOV	AX,DI			MOV	DX,AX	
	CALL	COLOR CAL			CALL	COLOR_CAL	
	MOV	DS:WORD PTR PITCHS,2	;PITCHS=2		MOV	AX, DX	
	MOV	DS:WORD PTR DHH,23	:DH=23				
	MOV	DS:WORD PTR DV,23	; DV=23		MOV	AL,AH	
	MOV	DS:WORD PTR X,BX	;X=(?)		AND	AX,7	
					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	;MOD1=0,MOD0=2		RET		
	MOV	DS:WORD PTR COM,8008H	; <a_copy_ac> COPY</a_copy_ac>		;		
			;ESE=0,REV=0,R0T=0,{MD}	DHDV_CA			
			;FAST=0 < !BUG!! >	01101_07	MOV	AX,SI	
	JMP	G_POP_EXE				AX	;DH=DX**2
		41. 41 Jana			MUL		,011-02442
;	,				MOV	BX,AX	
•					MOV	CX,DX	
1117 0					;		
INIT_S		222			MOV	AX,DI	
	PUSH	AX			MUL	AX	;DV=DY**2
	MOV	AL,9		DHDV CA	M. 1:		
	OUT	OD1H,AL	;ENABLE /CSIR/	-	0 R	DX,DX	
	IN	AL, OD1H			JZ	EXIT ROT	
	TEST	AL,1			SHR	DX,1	
	MOV	AX, ODOOOH			RCR		
	JZ	MOD_9801	CHECK IF 98XA/9801			AX,1	
	MOV	AX,8000H	, CHECK IF SOMA/SOUT		SHR	CX,1	
MOD 980		AA,0000n			RCR	BX,1	
100_960		PG 1Y			JMP	DHDV_CAL_1	
	MOV	ES,AX			;		
	MOV	AL,OFOH		EXIT_RC	DT_1:		
	OUT	91H,AL			SHR	AX,1	
	OUT	93H,AL			SHR	CX,1	
INIT_SI	EG_1:				RCR	BX,1	
	TEST	ES:WORD PTR STATUS,1		EXIT_R0		0.71	
	JNZ	INIT_SEG_1	;CHECK IF PPBUSY=1/0	BATT_N		cx.cx	
	MOV	ES:WORD PTR EADORGL.O	;EADORGL=0		OR		
		Solution of the Endowed to	, DADONAL-V		JNZ	EXIT_ROT_1	

-5-

; MOV DX,OFCOOH DHDV\_CAL\_3: TEST AX,DX JZ DHDV\_CAL\_2 SHR AX,1 SHR BX,1 JMP DHDV\_CAL\_3 ; DHDV\_CAL\_4: SHR BX,1 DHDV\_CAL\_4: TEST BX,DX JNZ DHDV\_CAL\_4 RET ; ; G POP EXE: POP ES POP ES POP DS POP DS POP D1 POP S1 POP AX POP CX POP AX POP CX POP AX POP AX POP AX POP BX MOV AX,O IRET ; G PUSH EXE: POP BX MOV AX,O IRET ; G PUSH EXE: POP BX MOV AX,O IRET ; G PUSH EXE: POP BX MOV AX,O IRET ; G PUSH EXE: POP BX MOV AX,O IRET ; G PUSH EXE: POP BX MOV AX,O IRET ; G PUSH EXE: POP BX MOV AX,O IRET ; G PUSH EXE: POP BX MOV AX,O IRET ; G PUSH EXE: POP BX MOV AX,O IRET ; G PUSH EXE: POP BX MOV AX,O IRET ; G PUSH EXE PUSH D1 PUSH D1 PUSH D1 PUSH D3 PUSH BP PUSH BP PUSH BP PUSH BA PUSH BP PUSH BP

;NORMAL END

-7-

-6-

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 SUB 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 DS:WORD PTR COM,1800H JNZ CHECK IF PIXEL=1/0 ;PLANES=(?) MOV ;<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 MOV ;MESSAGE(29) \$1,29 CALL MES\_CL\_WAIT RECT DEMO H CALL \$1,30 ;MESSAGE(30) MOV CALL MES\_CL\_WAIT RET RECT\_DEMO\_L: \$1,639 ;XS=639 MOV MOV D1,479 :YS=479 REPETITION COUNTS MOV CX.60 RECT\_EXE\_ALL JMP RECT\_DEMO\_H: SI,1119 ;XS=1119 MOV MOV D1,749 : VS=749 REPETITION COUNTS MOV CX,93 RECT\_EXE\_ALL: MOV AX.O :X=0 ;Y=0 MOV BX,0 RECT\_EXE\_ALL\_1: DS:WORD PTR STATUS,1 TEST JNZ RECT\_EXE\_ALL\_1 ;CHECK IF PPBUSY=1/0 DS:WORD PTR PMAX,4 DS:WORD PTR PTNCNT,0FFFFH MOV :PMAX=4 ;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 SUB SUB SUB SUB SUB SUB SUB SUB	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=(?) ;YS=(?) ;A.REC> ;PXEN=0 ;X+5> X ;Y+4> Y ;XS-5> XS ;YS-4> YS	PGROUP PROG CRL_ELP ; ;	ASSUME ; PUBLIC ; EXTRN ; EXTRN EXTRN ; S ;	CRL_ELPS PROG BYTE PUBLIC 'PROG' CS:PGROUP CRL_DEMO, ELPS_DEMO MES_CL_VAIT:NEAR STATUS:WORD, PMAX:WORD, PTNCN MODIO:WORD, PLANES:WORD, DXX: WORK_1:WORD, WORK_2:WORD, DHH PROC NEAR E DEMO >	WORD, COM:WORD
			CRL DEM	MOV CALL CALL MOV CALL CALL MOV CALL RET ; 0_L: MOV MOV	SI,36 MES_CL_WAIT CRL_DEMO_L SI,37 MES_CL_WAIT CRL_DEMO_H SI,38 MES_CL_WAIT SI,319 DI,239 BX,238	:MESSAGE(36) :MESSAGE(37) :MESSAGE(38) :XC=319 :YC=239 ;DXX=238
			CRL_DEM CRL_EXE CRL_EXE	MOV MOV MOV _ALL: MOV	CRL_EXE_ALL SI,559 DI,374 BX,373 AX,1 DS:WORD PTR STATUS,1 CRL_EXE_ALL_1 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PTNCNT,0FFFH DS:WORD PTR YC,SI DS:WORD PTR YC,DI DS:WORD PTR MODIO,10H	:XC=559 ;YC=374 ;DXX=373 :CHECK IF PPBUSY=1/0 :PMAX=4 ;PTNCNT=0FFFFH ;X=(?) ;Y=(?) ;MOD1=1,MOD0=0
MOV MOV INC SUB JNC RET ; < ELLIF	-2- DS:WORD PTR PLANES.AX DS:WORD PTR DXX,BX DS:WORD PTR COM.5000H AX BX.5 CRL_EXE_ALL_1 SE DEMO >	:PLANES=(?) ;DXX=(?) ; <crl> CF=0,IP=0 ;PXEN=0 ;PLANES+1 ;DXX-5</crl>		TEST JNZ MOV 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 WORK_1 DS:WORD PTR DHH,AX AX,CS:WORD PTR WORK_2 DS:WORD PTR MODIO.10H DS:WORD PTR MODIO.10H DS:WORD PTR PLANES,CX DS:WORD PTR DY,BX DS:WORD PTR COM,5COOH	:CHECK IF PPBUSY=1/0 ;XC=(?) ;VC=(?) :DV=(?) :DV=(?) :M001=1,M000=0 ;PLANES=(?) :DY=(?) ; <elps> CF=0,IP=0 :PXEN=0</elps>
ELPS_DEMO: MOV CALL MOV CALL MOV CALL MOV CALL RET : ELPS_DEMO_11:	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(39) ;DH=16 ;DV=9 ;DH=9 ;DV=16	; CRL_ELPS PROG	INC SUB JNC RET ; ENDS END	CX BX.5 ELPS_EXE_ALL_1 ENDP	;PLANES+1 ;DXX-5
ELPS_DELL CALL MOV CALL MOV CALL RET ELPS_DEMO_L: MOV	ELPS_DEMO_L SI,40 MES_CL_WAIT ELPS_DEMO_H SI,41 MES_CL_WAIT	;MESSAGE(40) ;MESSAGE(41) ;XC=319				
MOV MOV JMP ELPS DEMO H: MOV MOV ELPS EXE ALL: TEST JNZ JNZ MOV MOV	DI.239 BX.238 ELPS_EXE_ALL 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	:YC=239 :DY=238 :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	
	;		
	;	RECFILL DEMO	
	EXTRN EXTRN	DMCLO_C_WAIT:NEAR, MES_CL_WAIT: MES_CL_WAIT_FILL:NEAR	NEAR
	; EXTRN	STATUS:WORD, PDISPSL:WORD, PDIS	PSH:WORD, PMAX:WORD
	EXTRN	PTNPH:WORD, PTNPL:WORD, X:WORD, DY:WORD, PTNCNT:WORD, PLANES:WO	Y:WORD, DXX:WORD
	EXTRN EXTRN	COM:WORD, AGDC_SEG:WORD	KD, HODIO.WORD
RECFILL	; demo	PROC NEAR	
:	;		
;	< RECTA	NGLE FILLING DEMO >	
;			
	ňov	SI,1	
	CALL ;	MES_CL_WAIT	;MESSAGE(1)
RECFILL	_DEM0_1: TEST	DS:WORD PTR STATUS,1	
	JNZ	RECFILL_DEM0_1	;CHECK IF PPBUSY=1/0
		DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSH,0	;PDISPSL=10H ;PDISPSH=0
	MOV MOV	DS:WORD PTR PMAX,4 DS:WORD PTR PTNPH,0	;PMAX=4 ;PTNPH=0
	MOV	BX,903EH	; <r_recfill> ;TL=0,SS=1,WL=1,WR=1</r_recfill>
			;FAST=1
	CALL MOV	RECFILL_EXE SI,2	
	CALL :	MES_CL_WAIT_FILL	;MESSAGE(2)
	MOV	BX,90BCH	; <r_recfill> ;TL=1,SS=1,WL=1,WR=1</r_recfill>
			;FAST=0
	CALL MOV	RECFILL EXE	
	CALL ;	MES_CL_WAIT_FILL	;MESSAGE(3)
	MOV	BX,90ACH	; <r_recfill> ;TL=1,SS=0,WL=1,WR=1</r_recfill>
	CALL	RECFILL_EXE	;FAST=0
	MOV	SI,4	
		-1-	
	MOV MOV	ES,AX CX,12	REPETITION COUNTS
	MOV	DX,OF7DOH	;TILE OF800H
	TEST JZ	BL,10H RECFILL_2	CHECK IF "SS"=1/0
RECFILL	MOV 2:	DX,OFA30H	;TILE_SS OFA40H
	TEST JNZ	ES:WORD PTR STATUS,1 RECFILL_2	;CHECK IF PPBUSY=1/0
	TEST	BL,10H RECFILL_2_1	;CHECK IF "SS"=1/0
	ADD	DX,20H	, circer 11 55 -170
RECFILL	ADD	DX,10H	
	MOV	ES:WORD PTR PTNPL,DX	;PTNPL=(?)
	MOV LODSW	ES:WORD PTR X,AX	;X=(?)
	MOV	ES:WORD PTR Y,AX	;Y=(?)
	LODSW	ES:WORD PTR DXX,AX	;DX=(?)
	LODSW MOV	ES:WORD PTR DY,AX	;DY=(?)
	TEST JNZ	BL,80H RECFILL_TL1	;CHECK IF "TL"=1/0
	MOV	ES:WORD PTR PTNCNT,0 AX,CX	;PTNCNT=0
	AND	AX,7	AUDON IS NULLES
	JNZ MOV	RECFILL_3 AX,3	;CHECK IF "WHITE"
RECFILL	_3: MOV	ES:WORD PTR PLANES,AX	;PLANES=(?)
	MOV JMP	ES:WORD PTR MODIO,1 RECFILL_TL	;MOD1=0,MOD0=1
DECELLI	;		
RECFILL	MOV	ES:WORD PTR PTNCNT,16	;PTNCNT=16
	MOV MOV	ES:WORD PTR PLANES,7 ES:WORD PTR MOD10,0	;PLANES=7 ;MOD1=0,MOD0=0
RECFILL	TL: MOV	ES:WORD PTR COM, BX	; <com.flags></com.flags>
			, contranto.
	LOOP	RECFILL_2	
		DS	
RECFILL	LOOP POP RET ;		
RECFILL	LOOP POP RET ;	DS 0032H,01AEH,0186H,0FEDEH	; 1. X, Y, DX, DY ; 2. X, Y, DX, DY
RECFILL	LOOP POP RET ; DATA L: DW DW DW	DS 0032H,01AEH,0186H,0FEDEH 003CH,0104H,017CH,0FEE8H 010EH,019AH,0172H,0FEF2H	; 2. X,Y,DX,DY ; 3. X,Y,DX,DY
RECFILL	LOOP POP RET ; DATA L: DW DW	DS 0032H,01AEH,0186H,0FEDEH 003CH,0104H,017CH,0FEE8H	; 2. X,Y,DX,DY

-3-

	CALL	MES_CL_WAIT FILL	;MESSAGE(4)
	; MOV	BX,903EH	; <r_recfill> ;TL=0,SS=1,WL=1,WR=1 ;FAST=1</r_recfill>
	CALL	RECFILL_EXE_1	
	MOV CALL	SI,5 MES_CL_WAIT_FILL	;MESSAGE(5)
	; Mov	BX,90BCH	; <r_recfill> ;TL=1,SS=1,WL=1,WR=1 ;FAST=0</r_recfill>
	CALL MOV	RECFILL_EXE_1 SI,6	;MESSAGE(6)
	CALL ; MOV	MES_CL_WAIT_FILL	
	MUV	BX,90ACH	; <r_recfill> ;TL=1,SS=0,WL=1,WR=1 ;FAST=0</r_recfill>
	CALL	RECFILL_EXE_1	,1,1,51-0
	MOV CALL	SI,7 MES_CL_WAIT_FILL	;MESSAGE(7)
	;	HES_CL_WATT_FILL	,11233402(1)
	RET ;		
RECFILL	EXE: MOV	01-1	;DRAW COUNTS
RECFILL		D1,1	JUNAW COUNTS
	CALL DEC	RECFILL_EXE_L	
	JNZ	RECFILL_EXE1	
	RET :		
RECFILL			
RECELL	MOV EXE_11:	D1,1	;DRAW COUNTS
	CALL	RECFILL_EXE_H	
	DEC .INZ	DI RECFILL_EXE_11	
	RET		
RECFILL			
	MOV JMP	SI, OFFSET RECFILL_DATA_L RECFILL 1	
RECFILL	EXE_H:		
RECFILL	MOV 1:	SI, OFFSET RECFILL_DATA_H	
	PUSH	DS	
	MOV MOV	AX,CS DS,AX	
	MOV	AX, WORD PTR AGDC_SEG	
		-2-	
	DW DW	0064H,012CH,0154H,0FF10H 00E6H,0172H,014AH,0FF1AH	; 6. X,Y,DX,DY ; 7. X,Y,DX,DY
	DW	00DCH,0140H,0140H,0FF24H	; 8. X,Y,DX,DY
	DW DW		; 9. X,Y,DX,DY ;10. X,Y,DX,DY
	DW	00BEH,014AH,0122H,0FF42H	;11. X, Y, DX, DY
;	DW DW	00B4H,0168H,0118H,0FF4CH 00AAH,0136H,010EH,0FF56H	;12. X,Y,DX,DY ;13. X,Y,DX,DY
	;		
RECFILL	DATA_H: DW	001EH,02BCH,02BOH,0FE36H	; 1. X,Y,DX,DY
	DW	002DH,0195H,02A4H,0FE42H	; 2. X,Y,DX,DY
	DW DW	01B8H,029EH,0298H,0FE4EH 01A9H,01B3H,028CH,0FE5AH	; 3. X,Y,DX,DY ; 4. X,Y,DX,DY
	DW	005AH,0280H,0280H,0FE66H	; 5. X,Y,DX,DY
	DW DW	0069H,01D1H,0274H,0FE72H	; 6. X,Y,DX,DY
	DW	017CH,0262H,0268H,0FE7EH 016DH,01EFH,025CH,0FE8AH	; 7. X,Y,DX,DY ; 8. X,Y,DX,DY

		DW	0064H,012CH,0154H,0FF10H	; 6. X,Y,DX,DY
INTS		DW	00E6H,0172H,014AH,0FF1AH	; 7. X,Y,DX,DY
BOOH		DW	OODCH,0140H,0140H,0FF24H	; 8. X,Y,DX,DY
		DW	0082H,015EH,0136H,0FF2EH	; 9. X,Y,DX,DY
:1/0		DW	008CH,0154H,012CH,0FF38H	;10. X,Y,DX,DY
40H		DW	00BEH.014AH.0122H.0FF42H	;11. X,Y,DX,DY
		DW	0084H,0168H,0118H,0FF4CH	;12. X,Y,DX,DY
	:	DW	00AAH,0136H,010EH,0FF56H	;13. X,Y,DX,DY
SY=1/0	,	:	••••••••••••••••	2
	RECELL	L DATA H		
1/0	NEOT 10	DW	001EH,02BCH,02BOH,0FE36H	; 1. X,Y,DX,DY
		DW	002DH,0195H,02A4H,0FE42H	; 2. X,Y,DX,DY
		DW	0188H,029EH,0298H,0FE4EH	; 3. X.Y.DX.DY
		DW	01A9H,01B3H,028CH,0FE5AH	: 4. X.Y.DX.DY
		DW	005AH,0280H,0280H,0FE66H	: 5. X,Y,DX,DY
		DW	0069H,01D1H,0274H,0FE72H	; 6. X,Y,DX,DY
		DW	017CH,0262H,0268H,0FE7EH	; 7. X.Y.DX.DY
		DW	016DH,01EFH,025CH,0FE8AH	; 8. X,Y,DX,DY
		DW	0096H,0244H,0250H,0FE96H	: 9. X.Y.DX.DY
		DW	00A5H.020DH.0244H.0FEA2H	:10. X,Y,DX,DY
		DW	0140H,0226H,0238H,0FEAEH	:11. X.Y.DX.DY
		DW	0131H,022BH,022CH,0FEBAH	:12. X,Y,DX,DY
	;	DW	00D2H,0208H,0220H,0FEC6H	:13. X,Y,DX,DY
	<u>*</u>		000211,020011,022011,012001	, lot all the all the
1/0		,		
	RECFIL	DEMO	ENDP	
	PROG	ENDS	ENDP	
	PROG	END		
· F."		CNU		

```
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, PTNCNT: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
         MOV
                   DS:WORD PTR PMAX.4
                                                         :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
         MOV
                   ES:WORD PTR XC,AX
                                                         :XC=(?)
         LODSW
                   ES:WORD PTR YC,AX
                                                         :YC=(?)
          MOV
         TEST
                   BL . 80H
                   CRLFILL_TL1
                                                         ;CHECK IF "TL"=1/0
          JNZ
          MOV
                   ES:WORD PTR PTNCNT,0
                                                         :PTNCNT=0
              MOV AX.CX
          AND
                   AX,7
                                                         ;CHECK IF "WHITE"
          JNZ
                   CRLFILL_3
          MOV
                   AX.3
CRLFILL_3:
          MOV
                   ES:WORD PTR PLANES, AX
                                                         :PLANES=(?)
                   ES:WORD PTR MODIO,1
         MOV
                                                         ;MOD1=0,MOD0=1
                   CRLFILL_TL
          IMP
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
                   ELPSFILL_DEMO_1
         CALL
                   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
                 DX,OF7DOH
                                                              .. 0F800H
                                                     ;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
                                                     :PDISPSL=10H
         MOV
                 DS:WORD PTR PDISPSL,10H
DS:WORD PTR PDISPSH,0
                                                     :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
         CALL
                  ELPSFILL EXE
                 SI,2
MES_CL_WAIT_FILL
         MOV
                                                     ;MESSAGE(2)
         CALL
                                                     :<ELPSFILL> TL=1.SS=1
         MOV
                  RX.5CRCH
                  ELPSFILL_EXE
         CALL
         MOV
                  $1.3
                                                     :MESSAGE(3)
                  MES CL. WALT FILL
         CALL
                                                     :<ELPSFILL> TL=1,SS=0
         MOV
                  BX.5CACH
                  ELPSFILL_EXE
         CALL
                  51,4
         MOV
         CALL
                  MES_CL_WAIT_FILL
                                                     :MESSAGE(4)
         MOV
                  BX,5C3CH
                                                     ;<R_RECFILL> TL=0,SS=1
                  ELPSFILL EXE_1
         CALL
                  $1,5
         MOV
         CALL
                  MES CL WAIT FILL
                                                     :MESSAGE(5)
         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:
                 D1.1
                                                     ;DRAW COUNTS
         MOV
ELPSFILL EXE_11:
                  ELPSFILL_EXE_H
         CALL
         DEC
                  DI
```

ELPSFILI	JNZ RET ;	ELPSFILL_EXE_11		ELPSFILL_TLI 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
ELFSFIL	MOV JMP	SI,OFFSET CRLFILL_DATA_L ELPSFILL_1		ELPSFILL_TL: MOV LOOP	ES:WORD PTR COM,BX	; <com.flags></com.flags>
ELPSFILI	L EXE_H: MOV	SI, OFFSET CRLFILL DATA H		POP RET	DS	
ELPSFILI		DS		CRLFILL DATA	15	
	MOV	AX,CS DS,AX		DW DW	150, 200, 160 145, 490, 310	; 1. DX, XC, YC ; 2. DX, XC, YC
	MOV	AX,WORD PTR AGDC_SEG ES,AX		D W D W	140, 480, 180 135, 230, 290	; 3. DX, XC, YC ; 4. DX, XC, YC
	MOV MOV	CX,12 DX,0F7D0H	REPETITION COUNTS	DW DW	130, 240, 200 125, 450, 270	; 5. DX, XC, YC ; 6. DX, XC, YC
	TEST	BL,10H ELPSFILL_2	;CHECK IF "SS"=1/0	DW DW	120, 440, 220 115, 270, 250	; 7. DX, XC, YC ; 8. DX, XC, YC
ELPSFILI	MOV	DX,OFA30H ES:WORD PTR STATUS,1	;TILE_SS OFA40H	DW DW DW	110, 280, 240 105, 410, 230 100, 400, 260	; 9. DX, XC, YC ;10. DX, XC, YC ;11. DX, XC, YC
	JNZ TEST	ELPSFILL_2 BL,10H	;CHECK IF PPBUSY=1/0	DW	95, 310, 210	;12. DX, XC, YC
	JNZ ADD	ELPSFILL_2_1 DX,20H	;CHECK IF "SS"=1/0	CRLFILL DATA DW	H: 240, 326, 256	; 1. DX, XC, YC
ELPSFILI		DX,10H		DW DW	232, 784, 496 224, 768, 288	; 2. DX, XC, YC ; 3. DX, XC, YC
	MOV LODSW	ES:WORD PTR PTNPL,DX	;PTNPL=(?)	DW DW	216, 364, 464 208, 380, 320	; 4. DX, XC, YC ; 5. DX, XC, YC
	MOV	ES:WORD PTR DY,AX	;DY=(?)	D W D W	200, 720, 432 192, 704, 352	; 6. DX, XC, YC ; 7. DX, XC, YC
	MOV LODSW	ES:WORD PTR XC,AX	;XC=(?)	DW DW	184, 428, 400 176, 444, 384	; 8. DX, XC, YC ; 9. DX, XC, YC
	MOV	ES:WORD PTR YC,AX AX,CS:WORD PTR WORK_1	;YC=(?)	DW DW	168, 656, 368 160, 640, 416	;10. DX, XC, YC ;11. DX, XC, YC
	MOV MOV	ES:WORD PTR DHH,AX AX,CS:WORD PTR WORK_2	;DHH=(?)	DW	152, 492, 336	;12. DX, XC, YC
	MOV TEST	ES:WORD PTR DV,AX BL,80H	; DV=(?)	; CRLFILL ENDP		
	JNZ MOV MOV	ELPSFILL_TL1 ES:WORD PTR PTNCNT,0 AX,CX	;CHECK IF "TL"=1/0 ;PTNCNT=0	PROG ENDS END		
	AND JNZ MOV	AX,7 ELPSFILL_3 AX,3	;CHECK IF "WHITE"			
ELPSFILI	L_3: MOV MOV JMP ;	ES:WORD PTR PLANES,AX ES:WORD PTR MODIO,1 ELPSFILL_TL	;PLANES=(?) ;MOD1=0,MODO=1			
		-				
		-5-			-6-	
	_					
PGROUP	SEGMENT	TRAFILL PROG BYTE PUBLIC 'PROG' CS:PGROUP		CALL MOV CALL RET	SI,7	;MESSAGE(7)
PGROUP	GROUP SEGMENT ASSUME ;	PROG BYTE PUBLIC 'PROG'		MOV CALL RET ; TRAFILL_EXE_I	SI,7 MES_CL_VAIT_FILL L:	:MESSAGE(7)
PGROUP PROG	GROUP SEGMENT ASSUME ; PUBLIC :	PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL DEMO, TRAFILL EXE H	FILL:NEAR	MOV CALL RET ; TRAFILL EXE I MOV JMP	SI,7 MES_CL_VAIT_FILL L: SI,0FFSET_TRAFILL_DATA_L TRAFILL_1	;MESSAGE(7)
PGROUP PROG	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN ; EXTRN	PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE_H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PDI	SPSH:WORD, PMAX:WORD	MOV CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV	SI,7 MES_CL_VAIT_FILL L: SI,0FFSET_TRAFILL_DATA_L TRAFILL_1	;MESSAGE(7)
PGROUP PROG	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN ; EXTRN EXTRN EXTRN	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, PTNPL:WORD, X:WORD YS:WORD, YE:WORD, X:WORD, PTN	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD	MOV CALL RET TRAFILL EXE I MOV TRAFILL EXE I MOV TRAFILL_TEST	SI,7 MES_CL_WAIT_FILL SI,0FFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,0FFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1	
PGROUP PROG	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ;	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 CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL_1: TEST JNZ MOV	SI,7 MES_CL_WAIT_FILL SI,0FFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,0FFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1 TRAFILL_1 DS:VORD_PTR_PDISPSL,10H	:CHECK IF PPBUSY=1/0 :PDISPSL=10H
PGROUP PROG	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ;	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, PTNPL:WORD, X:WORD YS:WORD, YE:WORD, X:WORD, PTN	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD	MOV CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL_I: TEST JNZ MOV MOV	SI.7 MES_CL_WAIT_FILL SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1 TRAFILL_1 DS:WORD_PTR_PDISPSL,10H DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PMAX,4	;CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0 ;PMAX=4
PGROUP PROG	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ;	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 CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL_I: TEST JNZ MOV MOV MOV MOV PUSH MOV	SI.7 MES_CL_WAIT_FILL SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1 TRAFILL_1 DS:WORD_PTR_PDISPSL,10H DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PMAX,4 DS:WORD_PTR_PMAX,4 DS:WORD_PTR_PMAY,4 DS:WORD_PTR_PM	;CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0
PGROUP PROG TRAFILL ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE ;	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 NEAR	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD	MOV CALL RET : TRAFILL EXE I MOV TRAFILL EXE I MOV TRAFILL_1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL SI.0FFSET_TRAFILL_DATA_L TRAFILL_1 H: SI.0FFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS.1 TRAFILL_1 DS:WORD_PTR_PDISPSL.10H DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PTNPH.0 DS AX.CS DS.AX AX.WORD_PTR_AGDC_SEG	;CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0 ;PMAX=4
PGROUP PROG TRAFILL ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE ; DEMO: MOV	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 >	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD	MOV CALL RET ; TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL_1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1 TRAFILL_1 DS:WORD_PTR_PDISPSL,10H DS:WORD_PTR_PDISPSH,0 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_PMAX,4 DS:WORD_PTR_PMAX,4 DS:WORD_PTR_PMAX,4 DS:WORD_PTR_PMAX,4 DS:WORD_PTR_PMAX,4 DS:X AX,CS DS,AX AX,WORD_PTR_AGDC_SEG ES,AX CX,10	;CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0 ;PMX=4 ;PTNPH=0
PGROUP PROG TRAFILL ; ; ; ; ; ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN ; CTRAPE ; DEMO:	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, YIW MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO >	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;WORD ;MESSAGE(9) ;<a_trafill></a_trafill></pre>	MOV CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL_I: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD PTR FDISPSL,IOH DS:WORD PTR PDISPSL,IOH DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAY,4 DS:WORD PTR PMAY,4 DS:WORD PTR PMAY,4 DS:WORD PTR AGDC_SEG ES,AX CX,10 DX,OFTDOH BL,IOH	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMAX=4 ;PTNPH=0 :TILE 0F800H
PGROUP PROG TRAFILL ; ; ; ; ; ; ; ; ; ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE ; DEMO: MOV CALL MOV CALL	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	SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CKT:WORD, PLANES:WORD :WORD :WORD	MOV CALL RET : TRAFILL EXE I MOV TRAFILL EXE I MOV TRAFILL_I: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI.0FFSET_TRAFILL_DATA_H DS:WORD_PTR_FDISPSL,10H DS:WORD_PTR_PDISPSL,10H DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PTNPH,0 DS:WORD_PTR_PTNPH,0 DS:WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,VORD_PTR_AGDC_SEG ES,AX CX,10 DX,0F7D0H	;CHECK IF PPBUSY=1/0 ;PDISPSL=10H ;PDISPSH=0 ;PMAX=4 ;PTNPH=0
PGROUP PROG TRAFILL ; ; ; ; ; ; ; ;	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE ; DEMO: MOV CALL MOV CALL MOV CALL	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, 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	<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></pre>	MOV CALL RET ; TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL_1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1 TRAFILL_1 DS:WORD_PTR_PDISPSL,10H DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PMAX,4 DS:WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,WORD_PTR_AGDC_SEG ES,AX CX,10 DX,OF7DOH BL,10H TRAFILL_2 DX,OFA30H ES:WORD_PTR_STATUS,1	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMASH=0 :PTNPH=0 :TILE 0F800H :CHECK IF "SS"=1/0 :TILE_SS 0FA40H
PGROUP PROG TRAFILL ; ; ; ; TRAFILL	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE ; DEMO: MOV CALL MOV CALL MOV CALL MOV	PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE_H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PDI PTAPH:WORD, PTNPL:WORD, X:WORD 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,2 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</a_trafill></pre>	MOV CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL I: JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,0H DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAY,4 DS:WORD PTR PMAY,4 DS:WORD PTR AGDC_SEG ES,AX CX,10 DX,OFTDOH BL,10H TRAFILL_2 DX,OFA30H ES:WORD PTR STATUS,1 TRAFILL_2 BL,10H	:CHECK IF PPBUSY=1/0 :PDISPSI=10H :PDISPSH=0 :PMAX=4 :PTNPH=0 :TILE 0F800H :CHECK IF "SS"=1/0 :TILE_SS 0FA40H :CHECK IF PPBUSY=1/0
PGROUP PROG TRAFILL ; ; ; ; TRAFILL	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE ; DEMO: MOV CALL MOV CALL MOV CALL MOV CALL MOV	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, 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.708CH TRAFILL_EXE L S1.3	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CKT: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 CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL EXE I JNZ TRAFILL 1: TEST JNZ TRAFILL 2: MOV MOV MOV MOV MOV MOV MOV MOV TRAFILL 2: TEST JNZ TEST JNZ TEST JNZ	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1 TRAFILL_1 DS:WORD_PTR_PDISPSL,10H DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,VORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,WORD_PTR_AGDC_SEG ES,AX CX,10 DX,OF7DOH BL,10H TRAFILL_2 DX,OFA30H ES:WORD_PTR_STATUS,1 TRAFILL_2	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSH=0 :PMASH=0 :PTNPH=0 :TILE 0F800H :CHECK IF "SS"=1/0 :TILE_SS 0FA40H
PGROUP PROG TRAFILL ; ; ; ; TRAFILL	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE ; DEMO: MOV CALL MOV CALL MOV CALL MOV CALL	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, 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	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;WORD ;(A_TRAFILL&gt; ;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></pre>	MOV CALL RET ; TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1 TRAFILL_1 DS:WORD_PTR_PDISPSL,10H DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PMAX,4 DS:WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,WORD_PTR_AGDC_SEG ES,AX CX,10 DX,OF7DOH BL,10H TRAFILL_2 DX,OFA30H ES:WORD_PTR_STATUS,1 TRAFILL_2L1 DX,20H DX,10H	: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
PGROUP PROG TRAFILL ; ; ; ; TRAFILL	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE ; DEMO: MOV CALL MOV CALL MOV CALL MOV CALL MOV CALL	PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE_H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PDI PTRPH:WORD, PTNPL:WORD, X:WORD YS:WORD, YE:WORD, XE:WORD, YIW WODIO: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 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></a_trafill></pre>	MOV CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL I: JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSL,0H DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAX,4 DS:WORD PTR PMAY,4 DS:WORD PTR PMAY,4 CX,10 DX,OFTDOH BL,10H TRAFILL_2 DX,OFA30H ES:WORD PTR STATUS,1 TRAFILL_2 BL,10H TRAFILL_2 DX,0FA30H ES:WORD PTR STATUS,1 TRAFILL_2 DX,10H DX,10H ES:WORD PTR PTNPL,DX	:CHECK IF PPBUSY=1/0 :PDISPSI=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=(?)
PGROUP PROG TRAFILL ; ; ; ; 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	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, 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 S1,3 MES_CL_WAIT_FILL BX.70ACH TRAFILL_EXE L S1,4 MES_CL_WAIT_FILL BX.70ACH	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;WORD ;A_TRAFILL&gt; ;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></a_trafill></pre>	MOV CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL EXE I MOV TRAFILL I: TEST JNZ TEST JNZ TEST JNZ TEST JNZ TEST JNZ TEST JNZ TEST JNZ TEST JNZ TEST JNZ TEST JNZ TEST JNZ	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1 TRAFILL_1 DS:WORD_PTR_PDISPSL.10H DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_PDISPSH.0 DS:WORD_PTR_STATUS.1 TRAFILL_2 DX.0FA30H ES:WORD_PTR_STATUS.1 TRAFILL_2 DX.20H DX.10H ES:WORD_PTR_PTNPL.DX W ES:WORD_PTR_X.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 :CHECK IF "SS"=1/0 :PTNPL=(?) ; X=(?)
PGROUP PROG TRAFILL ; ; ; ; 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	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, 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 TRAFILL_EXE L SI.4 MES_CL_WAIT_FILL BX,703CH	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;WORD ;A_TRAFILL&gt; ;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></pre>	MOV CALL RET ; TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1 TRAFILL_1 DS:WORD_PTR_PDISPSL,10H DS:WORD_PTR_PDISPSH,0 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,OF7DOH BL,10H TRAFILL_2 DX,OFA30H ES:WORD_PTR_STATUS,1 TRAFILL_2 DX,OFA30H ES:WORD_PTR_STATUS,1 TRAFILL_2 DX,10H ES:WORD_PTR_PTNPL,DX W ES:WORD_PTR_Y,AX W	: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=(?) : X=(?) ; Y=(?)
PGROUP PROG TRAFILL ; ; ; ; TRAFILL	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE ; DEMO: MOV CALL CALL CALL CALL CALL CALL CALL CAL	PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE_H MES_CL_VAIT:NEAR, MES_CL_VAIT. STATUS:VORD, PDISPSL:WORD, PDI PTAPH:VORD, PTNPL:VORD, X:VORD YS:VORD, YE:WORD, XE:WORD, YIW MODIO:WORD, COM:VORD, AGDC_SEG NEAR ZOID FILLING DEMO > SI,9 MES_CL_VAIT BX,703CH TRAFILL_EXE_L SI,2 MES_CL_VAIT_FILL BX,70BCH TRAFILL_EXE_L SI,3 MES_CL_VAIT_FILL BX,70ACH TRAFILL_EXE_L SI,4 MES_CL_VAIT_FILL BX,703CH TRAFILL_EXE_L SI,5	<pre>SPSH:WORD, PMA:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;WORD ;WORD ;A_TRAFILL&gt; ;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></pre>	MOV CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL I: JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H 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 CX.10 DX.OFTDOH BL.10H TRAFILL_2 DX.OFA30H ES:WORD PTR STATUS,1 TRAFILL_21 DX.20H DX.10H ES:WORD PTR PTNPL,DX W ES:WORD PTR Y.AX W ES:WORD PTR Y.AX	:CHECK IF PPBUSY=1/0 :PDISPSI=00 :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 :PTNPL=(?) : X=(?) ; Y=(2) ;XS=(?)
PGROUP PROG TRAFILL ; ; ; ; 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	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.70ACH 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 H	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;WORD ;WORD ;A_TRAFILL&gt; ;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></pre>	MOV CALL RET ; TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL EXE I MOV TRAFILL I: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL,10H DS:WORD PTR PDISPSH,0 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 PMAX,4 DS:WORD PTR AGDC_SEG ES.AX CX,10 DX,OF7D0H BL,10H TRAFILL_2 DX,OFA30H ES:WORD PTR STATUS,1 TRAFILL_2 BL,10H TRAFILL_2 DX,20H DX,20H DX,20H DX,20H DX,10H ES:WORD PTR Y.AX W ES:WORD PTR Y.AX W	: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=(?) : X=(?) :XS=(?) :YS=(?)
PGROUP PROG TRAFILL ; ; ; ; TRAFILL	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; < TRAPE ; DEMO: MOV CALL	<pre>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, PTN MODIO:WORD, COM:WORD, AGDC_SEG NEAR ZOID FILLING DEMO &gt; SI.9 MES_CL_WAIT BX,703CH TRAFILL_EXE L SI.2 MES_CL_WAIT_FILL BX,708CH 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.5 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE H SI.5 MES_CL_WAIT_FILL BX,703CH TRAFILL_EXE_H SI.5 MES_CL_WAIT_FILL BX,708CH TRAFILL_EXE_H SI.5 MES_CL_WAIT_FILL BX,708CH</pre>	<pre>SPSH:WORD, PMAX:WORD , Y:WORD, XS:WORD CNT:WORD, PLANES:WORD :WORD ;WORD ;WORD ;WORD ;(A_TRAFILL&gt; ;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></pre>	MOV CALL RET ; TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL 1: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI,OFFSET_TRAFILL_DATA_L TRAFILL_1 H: SI,OFFSET_TRAFILL_DATA_H DS:WORD_PTR_STATUS,1 TRAFILL_1 DS:WORD_PTR_PDISPSL,10H DS:WORD_PTR_PDISPSH,0 DS:WORD_PTR_PMAX,4 DS:WORD_PTR_PMAX,4 DS:WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,WORD_PTR_PTNPH,0 DS AX,CS DS,AX AX,WORD_PTR_AGDC_SEG ES,AX CX,10 DX,OF7DOH BL,10H TRAFILL_2 DX,OFA30H ES:WORD_PTR_STATUS,1 TRAFILL_2 DX,10H ES:WORD_PTR_STATUS,1 TRAFILL_2_1 DX,20H DX,10H ES:WORD_PTR_Y,AX W ES:WORD_PTR_Y,AX W ES:WORD_PTR_Y,AX W ES:WORD_PTR_YS,AX W ES:WORD_PTR_YS,AX W	:CHECK IF PPBUSY=1/0 :PDISPSL=10H :PDISPSL=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=(?) :X=(?) :XS=(?) :YS=(?) ;YE=(?)
PCROUP PROG TRAFILL ; ; ; TRAFILL	GROUP SEGMENT ASSUME ; PUBLIC ; EXTRN 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	PROG BYTE PUBLIC 'PROG' CS:PGROUP TRAFILL_DEMO, TRAFILL_EXE_H MES_CL_WAIT:NEAR, MES_CL_WAIT. STATUS:WORD, PDISPSL:WORD, PDI PTPH:WORD, PTPL:WORD, X:WORD, YS:WORD, YE:WORD, X:WORD, X:WORD, X: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 TRAFILL_EXE_L SI.4 MES_CL_WAIT_FILL BX.70ACH TRAFILL_EXE_L SI.5 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 ;WORD ;A_TRAFILL&gt; ;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></pre>	MOV CALL RET : TRAFILL EXE I MOV JMP TRAFILL EXE I MOV TRAFILL I: JNZ MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	SI.7 MES_CL_WAIT_FILL L: SI.0FFSET_TRAFILL_DATA_L TRAFILL_1 H: SI.0FFSET_TRAFILL_DATA_H DS:WORD PTR STATUS,1 TRAFILL_1 DS:WORD PTR PDISPSL.10H DS:WORD PTR PDISPSH.0 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 AAGDC_SEG ES.AX CX.10 DX.0F7D0H BL.10H TRAFILL_2 DX.0FA30H ES:WORD PTR STATUS.1 TRAFILL_2 DX.10H ES:WORD PTR YNPL,DX W ES:WORD PTR Y.AX W ES:WORD PTR YS.AX W ES:WORD PTR YS.AX W ES:WORD PTR YS.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 PPBUSY=1/0 :CHECK IF "SS"=1/0 :PTNPL=(?) : X=(?) : X=(?) :XS=(?) :YS=(?)

	MOV MOV	ES:WORD PTR AX,CX	PTNCNT, AX	;PTNCNT=0
	AND JNZ MOV	AX,7 TRAFILL_3 AX,3		;CHECK IF "WHITE"
TRAFILL	_3: MOV	ES:WORD PTR	PLANES, AX	;PLANES=(?)
	MOV JMP	ES:WORD PTR TRAFILL_TL	M0D10,1	;MOD1=0,MOD0=1
TRAFILL	_TL1: MOV MOV MOV	ES:WORD PTR ES:WORD PTR ES:WORD PTR	PLANES,7	;PTNCNT=16 ;PLANES=7 ;MOD1=0,MOD0=0
TRAFILL		ES:WORD PTR		; <com.flags></com.flags>
	DEC JNZ POP RET	CX TRAFILL_2 DS		
TRAFILL	; _DATA_L: DW	00008 00968	01208 00168 00008 01	10EH; 1. X,Y,XS,YS,YE,XE
	DW DW DW DW DW DW DW DW	0091H,00AFH 0122H,00C8H 01B3H,00E1H 0244H,00FAH 0153H,014AH 0155H,0131H 015DH,0118H 0162H,00FFH	0172H,0064H,0019H,0 018BH,00AAH,0032H,0 01FEH,00FOH,004BH,0 0244H,0136H,0064H,0 027FH,0171H,01DFH,0 01EEH,012BH,01C6H,0 015DH,00E5H,01ADH,0 00CCH,009FH,0194H,01	154H; 2. X,Y,XS,VS,VE,XE 19AH; 3. X,Y,XS,VS,VE,XE 1EOH; 4. X,Y,XS,VS,YE,XE 226H; 5. X,Y,XS,VS,YE,XE 226H; 6. X,Y,XS,VS,YE,XE 218H; 7. X,Y,XS,VS,VE,XE 105H; 8. X,Y,XS,VS,VE,XE 149H;10. X,Y,XS,VS,VE,XE
TRAFILL	.DATA .H: DW	00008 00548	02008 00358 00008 01	1D9H; 1. X,Y,XS,YS,YE,XE
	DW DW	00FEH,0111H	,0288H,00B0H,0027H,02	254H; 2. X,Y,XS,YS,YE,XE 2CFH; 3. X,Y,XS,YS,YE,XE
	DW DW	02FAH,015FH	,037EH,01A6H,0075H,03	34AH; 4. X,Y,XS,YS,YE,XE 3C5H; 5. X,Y,XS,YS,YE,XE
	DW DW	025AH,01BDH	,0360H,020CH,02C5H,03	42AH; 6. X,Y,XS,YS,YE,XE 3BOH; 7. X,Y,XS,YS,YE,XE
	DW DW DW	026CH,016FH	,0164H,0118H,0277H,02	336H; 8. X,Y,XS,YS,YE,XE 2BCH; 9. X,Y,XS,YS,YE,XE 242H;10. X,Y,XS,YS,YE,XE
;	;	02100,01100	,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	
TRAFILL Prog	ENDS			
	END			
			-3-	
	CALL	TRIFILL_EXE		
	MOV CALL	SI,4 MES_CL_WAIT_	FILL	;MESSAGE(4)
	; MOV MOV	CS:WORD PTR CS:WORD PTR		;X=560 ;Y=374
	MOV	BX,6C3CH TRIFILL_EXE_	1	; <a_trifill> ;TL=0,SS=1,WL=1,WR=1</a_trifill>
	MOV	SI,5 MES_CL_WAIT_		;MESSAGE(5)
	; MOV	BX,6CBCH		; <a_trifill></a_trifill>
	CALL	TRIFILL EXE		;TL=1,SS=1,WL=1,WR=1
	MOV	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	TRIFILL EXE		,
	CALL	MES_CL_WAIT_	FILL	;MESSAGE(7)
	RET ;			
TRIFILL	MOV	DI,1		;DRAW COUNTS
TRIFILL	CALL	TRIFILL_EXE_	L	
	DEC JNZ RET	TRIFILL_EXE1		
TRIFILL	.EXE_1: MOV	DI,1		;DRAW COUNTS
TRIFILL	EXE_11: CALL	TRIFILL_EXE_	н	
	DEC JNZ	DI TRIFILL_EXE		
	RET			
TRIFILL	MOV		IFILL_DATA_L	
TRIFILL	JMP	TRIFILL_1		
		SL. OFFSET TP	IFILL DATA H	
TRIFILL	MOV	SI,OFFSET TR	IFILL_DATA_H	
TRIFILL.	MOV _1:		IFILL_DATA_H	

PGROUP PROG	NAME GROUP SEGMENT ASSUME	TRIFILL PROG BYTE PUBLIC 'PR CS:PGROUP	OG'	
	; PUBLIC	TRIFILL DEMO		
	; EXTRN	DMCLO_C_WAIT:NEAR,		AR
	EXTRN ;	MES_CL_WAIT_FILL:NE		
	EXTRN EXTRN EXTRN EXTRN EXTRN	STATUS:WORD, PDISPS PTNPH:WORD, WORK_1: X:WORD, Y:WORD, XC: PTNCNT:WORD, PLANES AGDC_SEG:WORD	WORD, WORK_2:W WORD, YC:WORD,	ORD, PTNPL:WORD XS:WORD, YS:WORD
TRIFILL	, PROC ;	NEAR		
į	< TRIAN	GLE FILLING DEMO >		
;	ILL_DEMO	]		
;	;			
TRIFILL	DEMO: MOV CALL	SI,8 MES_CL_WAIT	;	MESSAGE(8)
TRIFILL	; _DEM0_1:			
	TEST JNZ	DS:WORD PTR STATUS, TRIFILL DEMO_1		CHECK IF PPBUSY=1/0
	MOV MOV	DS:WORD PTR PDISPSL DS:WORD PTR PDISPSH		PDISPSL=10H PDISPSH=0
	MOV	DS:WORD PTR PMAX,4	;	PMAX=4
	MOV MOV	DS:WORD PTR PTNPH,O CS:WORD PTR WORK_1,	320 ;	PTNPH=0 X=320
	MOV MOV	CS:WORD PTR WORK_2, BX,6C3CH		Y=240 <a_trifill></a_trifill>
	CALL	TRIFILL EXE		TL=0,SS=1,WL=1,WR=1
	MOV	\$1,2		MESSAGE(2)
	; MOV	MES_CL_WAIT_FILL		
		BX, 6CBCH		<a_trifill> TL=1,SS=1,WL=1,WR=1</a_trifill>
	CALL	TRIFILL_EXE		NEGG1 (2)
	CALL ; MOV	MES_CL_WAIT_FILL		MESSAGE(3)
	NUV	BX,6CACH		<a_trifill> TL=1,SS=0,WL=1,WR=1</a_trifill>
		-1-		
	MOV	AX, WORD PTR AGDC_SEG		
	MOV MOV	ES,AX CX,12		REPETITON COUNTS
	MOV TEST	DX,OF7DOH BL,10H		FILE 0F800H
	JZ MOV	TRIFILL_2 DX,OFA30H		CHECK IF "SS"=1/0 FILE_SS OFA40H
TRIFILL	2: TEST	ES:WORD PTR STATUS,1		
	JNZ TEST	TRIFILL_2 BL,10H	;(	CHECK IF PPBUSY=1/0
	JNZ ADD	TRIFILL_2_1 DX,20H	;0	CHECK IF "SS"=1/0
TRIFILL_		DX,10H		
	MOV	ES:WORD PTR PTNPL,DX AX,CS:WORD PTR WORK_		PTNPL=(?)
	MOV	ES:WORD PTR X,AX	;)	(=(?)
	MOV	AX,CS:WORD PTR WORK_ ES:WORD PTR Y,AX		/=(?)
	LODSW	ES:WORD PTR XC,AX	;)	(C=(?)
	LODSW	ES:WORD PTR YC,AX	;'	/C=(?)
	LODSW MOV	ES:WORD PTR XS,AX	;;	(S=(?)
	LODSW	ES:WORD PTR YS,AX	:'	(S=(?)
	TEST	BL,80H TRIFILL_TL1		CHECK IF "TL"=1/0
	MOV	AX,0		
	MOV	ES:WORD PTR PTNCNT, A AX, CX	х ,1	PTNCNT=0
	AND JNZ	AX,7 TRIFILL_3	;(	CHECK IF "WHITE"
TRIFILL_		AX,3		
	MOV MOV	ES:WORD PTR PLANES, A ES:WORD PTR MODIO, 1		PLANES=(?) 10D1=0,M0D0=1
	JMP ;	TRIFILL_TL		
TRIFILL_	TL1: MOV	ES:WORD PTR PTNCNT,1	6 ;	PTNCNT=16
	MOV MOV	ES:WORD PTR PLANES,7 ES:WORD PTR MOD10,0	;1	PLANES=7 10D1=0,M0D0=0
TRIFILL_		ES:WORD PTR COM, BX		<com.flags></com.flags>
	DEC	CX TRIFILL_2	,	
	POP	DS		
	AL.			

; TRIFILL DATA L: DW DW DW DW DW DW DW DW DW DW DW DW DW	01CBH,0140H,0098H,01A3H 0190H,017BH,0056H,0137H 0140H,0190H,0051H,008BH 00F0H,017BH,008DH,0048H 0085H,0140H,00F9H,0006H 00A0H,00F0H,0178H,0001H 0085H,00A0H,01E8H,003DH 00F0H,0055H,022H,0128H 0140H,0050H,022H,0128H 0140H,0050H,0173H,0198H 01CBH,00A0H,0187H,010AH 01E0H,00F0H,0108H,01DFH	<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 ; 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</pre>	PGROUP PROG	ASSUME ;	PAINT PROG BYTE PUBLIC 'PROG' CS:PGROUP PAINT_DEMO SHRINK_PUT:NEAR, SHRINK_EXE:NE/ STATUS:WORD, XCLMIN:WORD, YCLM YCLMAX:WORD, MAGETC:WORD, PTAX: PDISPSH:WORD, PDISPSL:WORD, PTI PLAMES:WORD, X:WORD, Y:WORD, CO PTNPL:WORD	IN:WORD, XCLMAX:WORD WORD, PITCHS:WORD WCNT:WORD, MODIO:WORD
TRIFILL_DATA_H: DW DW DW DW DW DW DW DW DW DW TRIFILL ENDP PROG ENDS END	0315H.01FAH.011BH.029EH 0285H.025BH.00ACH.01ECH 0230H.027FH.00A6H.011AH 01ACH.025BH.0108H.0061H 014BH.01FBH.01BAH.0FFF2H 0127H.0176H.028CH.0FFECH 014BH.00F2H.0345H.010AH 0280H.036H.038AH.0102H 0280H.036BH.038AH.0102H 0215H.0091H.0358H.02FAH 0315H.0071H.02A6H.02FAH 0339H.0176H.01D4H.0300H	: 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 : 9. X, Y, DX, DY : 10. X, Y, DX, DY : 11. X, Y, DX, DY : 12. X, Y, DX, DY	PAINT .C	CALL MOV MOV CALL ;	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 DS:WORD PTR XCLMIN,5 DS:WORD PTR YCLMAX,744 DS:WORD PTR YCLMAX,744 DS:WORD PTR YCLMAX,744 DS:WORD PTR YCLMAX,744 DS:WORD PTR PITCHS,46H DS:WORD PTR PDISPSH,1 DS:WORD PTR PLANES,6 DS:WORD PTR PLANES,6 DS:WORD PTR Y,560 DS:WORD PTR COM,6834H	<pre>;15/16&gt; 10/16 ;CHECK IF PPBUSY=1/0 ;XCLMIN=5 ;YCLMIN=80 ;XCLMAX=591 ;YCLMAX=744 ;PITCHS=48H ;PDISPSH=1</pre>
			PAINT_[	DEMO_2: TEST JNZ	DS:WORD PTR STATUS,1 PAINT_DEMO_2	;CHECK IF PPBUSY=1/0
MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	-4- DS: WORD PTR XCLMIN,596 DS: WORD PTR YCLMIN,80 DS: WORD PTR YCLMAX,744 DS: WORD PTR YCLMAX,744 DS: WORD PTR YLANES,3 DS: WORD PTR Y.560 DS: WORD PTR Y.560 DS: WORD PTR Y.560 DS: WORD PTR YCMAX,749 DS: WORD PTR YCLMIN,0 DS: WORD PTR YCLMIN,30 DS: WORD PTR YCLMIN,30 DS: WORD PTR YCLMIN,30 DS: WORD PTR YCLMIN,30 DS: WORD PTR YCLMAX,719 DS: WORD PTR YCLMAX,749 DS: WORD PTR YCLMAX,749 DS: WORD PTR YCLMAX,749 DS: WORD PTR PIDSPSL,16 DS: WORD PTR PTNPH,0 DS: WORD PTR YLANES,7 DS: WORD PT	:XCLMIN=596 ;YCLMIN=80 ;XCLMAX=1115 ;YCLMAX=744 ;PLANES=3 ;X=800 ;Y=560 ; <paint> ;PM0D=1, TL=0, SS=1 :CHECK IF PPBUSY=1/0 ;XCLMIN=30 ;XCLMAX=1119 ;YCLMAX=749 ;PDISPSH=0 ;PDISPSH=0 ;PDISPSH=0 ;PTNCNT=16 ;PLANES=7 ;MOD1=0, MOD0=1 ;PTNPH=0 ;YCLM1&gt; ;PM0D=1, TL=1, SS=0 ;<paint> ;PM0D=1, TL=1, SS=0 ;REPETITION COUNTS :CHECK IF PPBUSY=1/0 ;XCLMAX=749 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMAX=740 ;YCLMA</paint></paint>	PAINT_D ; PAINT PROG	MOV INC INC MOV MOV INC INC INC MOV ADD MOV ADD LOOP RET JW DW DW DW DW DW DW DW DW DW DW S;	-1- AX,CS:[S1] S1 S1 S1 S1 S1 S1 S1 S1 S1 S1	;X=(?) ;FTNCNT=16 ;FTNPL+30H> PTNPL ;FTNPL=(?) ; <com.flags> :16/16 :15/16 :14/16 :12/16 :11/16 :10/16</com.flags>

CTRV         10012-000000, X120000, Y120100         Status PT K 1.5.8         CTRV         CTRV         Status PT K 1.5.8         CTRV         CTRV           CTRV         CTRV         CTRV         Status PT K 1.5.8         CTRV         CTR	
intro         SCHL         SCHL <t< td=""><td>CMP CX,166 G' JB SCROLL_SKIP</td></t<>	CMP CX,166 G' JB SCROLL_SKIP
EVEN         CHR, MAR HUMP, FLAMA         PECHLONE         CHR, LINE, MAR, LINE, MAR, LINE, MAR, LINE, LINE, LINE, MAR, LINE,	SCROLL_SKIP:
ETT         LORE 1, TORE, LORE 2, LORE 0, STREED, TORE 0         TTC         TTC         STANDE FT STATUS, 1         TTCS         STANDE FT	
SENDL / PRO       PRO       SENDL DEPO >       SENDL DEP	WORD, WORK_3:WORD, WORK_4:WORD         TEST         DS:WORD PTR STATUS,1           WORD, PMAX:WORD, PITCHS:WORD         JNZ         SCROLL_P1         :CHECK IF PPBUSY=1/0           YS:WORD, X:WORD, Y:WORD         MOV         DS:WORD PTR XS,AX         :XS=(?)           OM:WORD, PDISPSH:WORD         MOV         DS:WORD PTR YS,BX         :YS=(?)           MOV         DS:WORD PTR X,16         :X=16
* SCHOLL DED >         PO         B5:000 FTE COLLAGED         [-4, CPV] [-5,	MOV DS:WORD PTR DHH,255 ;DH=255
SCHUL, S.L.         CH, BAX, SEN, T         ISA 00 F KKX-GROUP         ISA 00 F KKX-GROU	MOV DS:WORD PTR COM,840EH ; <a_copy_cc> COPY ;ESE=0,REV=0,ROT=0 ;{MSD},FAST=1</a_copy_cc>
CALL         CAL STRUME (19) #         The SPECE-EDONS         SCREEL_PE: The SPECE (19) #         SCREEL (19) #         SCR	
MAY         CS:000 PT VORS.3.560         :144300         MVV         B:000 PT RV.355         :00-25           SCRUEL_DEWS_11         :200         :14407         :20071100         :14407           TX         SCRUEL_DEWS_11         :20071100         :1450         :14407           MAY         SCRUEL_DEWS_11         :20071100         :1450         :1450           MAY         SCRUEL_DEWS_11         :20071100         :1500         :1500         :1500           MAY         SCRUEL_DEWS_12         :20181581.0         :20181581.0         :1500         :1500         :1500         :1500         :1500         :20181581.0         :201811.0         :20181581.0         :20181581.0         :20181581.0         :20181581.0         :20181581.0         :20181581.0         :20181581.0         :20181581.0         :20181581.0         <	SEND OF BACK-GROUND         TEST         DS:WORD PTR STATUS,1           ;X1=0         JNZ         SCROLL_P2         ;CHECK IF PPBUSY=1/0           ;Y1=260         MOV         DS:WORD PTR XS,SI         ;XS=(?)           ;X2=560         MOV         DS:WORD PTR YS,DI         ;YS=(?)           ;Y2=477         MOV         DS:WORD PTR X,288         ; X=288
JAZ SCRUL, LORDO, 1 CONCAL (F PPRUSY170 KT 100 KT 100 SCRUP PT STATUS, 1 FOR FOR STATUS, 1 FOR STATU	56         ;Y3=256         MOV         DS:WORD         PTR         DH=255         ;DH=255           60         ;X4=560         MOV         DS:WORD         PTR         DV,255         ;DV=255           77         ;Y4=477         MOV         DS:WORD         PTR         COM,840EH         ; <a_copy_cc>         COPY           ;REPETITION         COUNTS         ;SE=0,REV=0,ROT=0         ;{MSD},FAST=1</a_copy_cc>
NOV         DS:UNDD PTR P015FSL-0         PP015FSL-0         SCOLL_P3:           NOV         DS:UNDD PTR FLASES.7         JOINT PLASES.7         JOINT	;CHECK IF PPBUSY=1/0 RET
JB         SCROLL_SKIP         :08-255         :09-255 <td< td=""><td>0 ;PDISPSL=0 SCROLL_P3: ;PLANES=7 TEST DS:WORD PTR STATUS,1 ;PMAX=4 JNZ SCROLL_P3 ;CHECK IF PPBUSY=1/0 6H ;PITCHS=46H MOV DX;CS:WORD PTR VORK_1 H ;MODIO=11H MOV DS:WORD PTR XS,DX ;XS=(?) MOV DX;CS:WORD PTR VORK_2 MOV DS:WORD PTR VS,DX ;YS=(?) MOV DS:WORD PTR X5,60 ; X=560</td></td<>	0 ;PDISPSL=0 SCROLL_P3: ;PLANES=7 TEST DS:WORD PTR STATUS,1 ;PMAX=4 JNZ SCROLL_P3 ;CHECK IF PPBUSY=1/0 6H ;PITCHS=46H MOV DX;CS:WORD PTR VORK_1 H ;MODIO=11H MOV DS:WORD PTR XS,DX ;XS=(?) MOV DX;CS:WORD PTR VORK_2 MOV DS:WORD PTR VS,DX ;YS=(?) MOV DS:WORD PTR X5,60 ; X=560
ADD         CS:U0RD PTE VORK_1.3 XSF3         :(MSD), FAST=0 <118UG11> XSF3         RET           SCROLL, PI FEST         DS:U0RD PTE VORK_3.3 XSF3         :XSF3         PGROUP GROUP         PROUP GROUP         PROUP CS:U0RD PTE VORK_3.4 XSURE         :VSF3           MOV         DS:U0RD PTE STATUS,1 XSF3         :CRECK IF PPRUSY=1/0         : :ETTEN DS:U0RD PTE VORK,3         :XSF6(7)         : :ETTEN STATUS:00RD, DR:U00RD, DV:U0RD, X:U0RD, X:U0RD, X:U0RD, CON: EXTEN FLAXES,U0RD, PTR:V0RR, 0010:U0RD, CON: EXTEN FLAXES, 100-255         : : : : : : : : : : : : : : : : : : :	MOV         DS:WORD         PTR         DHH,255         ; DH=255           MOV         DS:WORD         PTR         DV,255         ; DV=255
ADD         CS: VORD PTR VORK_1.3         IXS-3         PFR0G         SCMUP         PR0G         SCMUP         PR0G           RCT         IXS-3         IYS-3         IYS-3         IYS-3         IYS-3         IYS-3           SCROLL,P4:         IYS-3         IYS-3         IYS-3         IYS-3         IYS-3           IXZ         SCROLL,P4:         IYS-3         IYS-3         IYS-3         IYS-3           MV         DX:CSURD PTR VORK_2.3         IXS-70         IXS-77         IYS-3         IYS-3           MV         DX:CSURD PTR VORK_3         IXS-77         EXTR         PUT DATA_TRANSLORD, DNE:VORD, DV:VORD, XV:VORD, XV:VORD, VX:VORD, VX:V	-2-
TEST D5:VORD PTR STATUS.1 SCRUL PA NOV D5:VORD PTR VORK.3 MOV D5:VORD PTR VORK.4 MOV D5:VORD PTR VORK.3 MOV D5:VORD PTR VORK.3 KESTER REVOK NOV D5:VORD PTR VORK.3 KESTER REVOK NOV D5:VORD PTR VORK.4 KESTER REVOK KESTER REVOK K	;XS+3 PGROUP GROUP PROG ;YS+3 PROG SEGMENT BYTE PUBLIC 'PROG'
NOV         DS.US.UORD PTR VORK_3         :         :         :         EXTEN \$TATUS:VORD, DHR:VORD, X:VORD, X:VORD, OCH: BXTEN \$TATUS:VORD, PTR VORK_4           MOV         DS:VORD PTR VORK_4         :         EXTEN \$TATUS:VORD, DHR:VORD, MAGETC:VORD         :           MOV         DS:VORD PTR VORK_4         :         EXTEN \$TATUS:VORD, PTR VORD, MAGETC:VORD         :           MOV         DS:VORD PTR VORK_3         :         :         EXTEN \$TATUS:VORD, VE:VORD, MAGETC:VORD         :           MOV         DS:VORD PTR VORK_3         :         :         :         EXTEN \$TATUS:VORD, VE:VORD, MAGETC:VORD           MOV         DS:VORD PTR VORK_3.4         :         :         :         :         :           MOV         DS:VORD PTR VORK_3.4         :         :         :         :         :           SUB         CS:VORD PTR VORK_4.4         :         :         :         :         :         :           SUB         CS:VORD PTR VORK_4.4         :VS-4         :         :         :         :         :           SCROLL         ENDF         :         :         :         :         :         :         :         :           SCROLL         ENDF         :         :         :         :	
MOV         DS:UORD PTR VS.DX         :VS=(7)         EXTEN         PITCHS:UORD, VS:UORD, VS:UORD, DXX:UORD, DXX:UORD, TX, VS32         DXX:UORD, IS:UORD, PTR, V, 734         IS:UORD, PTR, V, 735         ID:UORD, PTR, V, 735         ID:UORD, PTR, V, 735         ID:UORD, PTR, V, 735         ID:UORD, PTR, V, 736         IS:UORD,	3 ; ;XS=(?) EXTRN STATUS:WORD, DHH:WORD, DV:WORD, X:WORD, Y:WORD
MOV         DS:U0RD PTR DBH.255         IDH=255         FRCOPY         PROC         NEAR           MOV         DS:U0RD PTR V07,840CH         1:4.00PV.CC5 COPY         :              FR_COPY DEMO >         :	;YS=(?)EXTRNPITCHS:WORD, XS:WORD, YS:WORD, DX:WORD, DY:WORD; X=832EXTRNXE:WORD, YE:WORD, MAGETC:WORD
iESE-0, REV-0, ROT=0 (MSD), FAST=0 <118UG11>         :         < FR_COPY DEM0 >           SUB         CS:WORD PTR WORK_3.4 SUB         :XS-4         :           SUB         CS:WORD PTR WORK_4.4 SUB         :YS-4         :           RET         :         :         :           SCROLL         EXDP         SCROPY DEM0         :CHECK HI NOV DS:WORD PTR NDH,255         :DH=255           PROG         END         SUBND         S:WORD PTR NDH,255         :DH=255           NOV         DS:WORD PTR NDV.255         :DH=255         :DH=255           NOV         DS:WORD PTR NDV.255         :DH=255           NOV         DS:WORD PTR NV.255         :PH2A54           NOV         DS:WORD PTR NOV.255         :PH2A54           NOV         DS:WORD PTR NOV.255         :PH2A54           NOV         SWORD PTR NOV.255         :PH2A54           NOV         DS:WORD PTR NOV.255         :PH2A54           NOV         SWORD PTR NOV.255         :PH2A54	:DV=255 ;
SUB CS:WORD PTR WORK_4.4 ;YS-4 : RET FRCOPY_DEMO: : SCROLL ENDP PROG ENDS END END END END END END END END	;ESE=0,REV=0,ROT=0 ; < FR_COPY DEM0 > ; {MSD},FAST=0 !BUG!! ;
MOV       AX,6C00H       :SEGMENT         MOV       DS,AX       :256 X 2:         MOV       CX,3000H       :256 X 2:         CALL       PUT_DATA_TRAN       :         POP       DS       :         MOV       CX,16       :         MOV       CX,16       :         FRCOPY_DEM0_1:       :       :         JNZ       FRCOPY_DEM0_1       :         MOV       DS:WORD PTR STATUS,1       :         MOV       DS:WORD PTR STATUS,1       :         MOV       DS:WORD PTR X5,30       :         MOV       DS:WORD PTR VS,720       :       :         MOV       DS:WORD PTR X5,704       :       :	FRCOPY DEMO:           TEST         DS:WORD PTR STATUS,1           JXZ         FRCOPY DEMO         CHECK IF PPBUSY=1/0           MOV         DS:WORD PTR DHH,255         ;DH=255           MOV         DS:WORD PTR DV,255         ;DV=255           MOV         DS:WORD PTR V,30         ;X=30           MOV         DS:WORD PTR Y,720         ;Y=720           MOV         DS:WORD PTR PLANES,7         ;PLANES=7           MOV         DS:WORD PTR PMAX,4         ;PMAX=4           MOV         DS:WORD PTR MOD10,0         ;M0D0=0
MOV         SI,OFFSET         FRCOPY_DATA           MOV         CX,16         ;REPETIT           FRCOPY_DEM0_1:         TEST         DS:WORD         PTR         STATUS,1           JNZ         FRCOPY_DEM0_1         ;CHECK         II           MOV         DS:WORD         PTR         PTICHS,46H         :PITCHS=4           MOV         DS:WORD         PTR         YS.30         :XS=30           MOV         DS:WORD         PTR         YS.720         :YS=720           MOV         DS:WORD         PTR         X,704         :X=704	MOV         AX,6C00H         ;SEGMENT "MAAJAN"           MOV         DS,AX            MOV         CX,3000H         ;256 X 256 X 3 PLANES           CALL         PUT DATA_TRAN            POP         DS
MOV DS:WORD PTR DHH,255 ;DH=255	MOV         SI,OFFSET         FRCOPY_DATA         ;REPETITION COUNTS           MOV         CX,16         ;REPETITION COUNTS           FRCOPY_DEMO_11:         TEST         DS:WORD PTR STATUS,1         ;CHECK IF PPBUSY=1/0           MOV         DS:WORD PTR PITCHS,46H         ;PITCHS=46H           MOV         DS:WORD PTR XS,30         ;XS=30           MOV         DS:WORD PTR YS,720         ;YS=720           MOV         DS:WORD PTR Y,704         ; X=704           MOV         DS:WORD PTR Y,375         ; Y=375

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	ASSUME ;	MESSAGE PROG BYTE PUBLIC 'PROG' CS:PGROUP	
MOV INC	AX,CS:[SI] SI			PUBLIC ;	MESSAGE	
INC MOV	SI DS:WORD PTR DY,AX	;DY=(?)		EXTRN ;	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:WOR EAD2H:WORD, dAD2:BYTE, X:WORD, FLAG:BYTE, DXX:WORD	D, EAD2L:WORD
MOV	AX,CS:[SI] SI		MESSAGE	PROC	NEAR	
INC MOV	SI DS:WORD PTR YE,AX	;YE=(?)	÷	,	PREVIOUSLY ASSIGNED MESSAGE >	
MOV	AX,CS:[SI] SI		, ; ; .		ON PREDETERMINED PAGE (SI; 0 TO	(2))
I NC MOV	SI DS:WORD PTR MAGETC,AX	;MAGETC=(?)	, r k i ni	;		
MOV LOOP Ret	DS:WORD PTR COM,843EH FRCOPY_DEMO_1	; <a_copy_cc> FR_COPY ;ROT=0,{MSD}</a_copy_cc>		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 ;DH=23
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 0, 2, 2, 0, 1FFH -1, 2, 2, 1, 1DDH -2, 2, 2, 2, 1AAH -2, 1, 1, 2, 1DDH -2, 0, 0, 2, 1FFH -2, -2, -2, 1, 1DDH -1, -2, -2, 1, 1DDH -2, -2, -2, 0, 1FFH	;FR(0) ;FR(1) ;FR(2) ;FR(3) ;FR(4) ;FR(5) ;FR(6) ;FR(6) ;FR(7) ;FR(8) ;FR(9) ;FR(4) ;FR(8)		MOV MOV PUSH MOV MOV MOV MOV MOV ADD MOV MOV	DS: WORD PTR DHH,23 DS: WORD PTR DV,23 DS: WORD PTR MD10,97H DS: WORD PTR PLANES,6 DS AX,DS ES,AX BX,2 CX,746 AX,54FOH DS,AX S1,S1 S1,[S1] AX,5500H	:DV=23 :NOD1=9.NOD0=7 :PLANES=6 :X=2 :Y=746 :PAGE ADDRESS SEGMENT :MESSAGE SEGMENT
DW	1, -2, -2, -1, 1DDH	;FR(C)	MEC 01	MOV	DS,AX	, HESSAGE SEGHENT
DW DW DW ;	2, -2, -2, -2, 1AAH 2, -1, -1, -2, 1DDH 2, 0, 0, -2, 1FFH	;FR(D) ;FR(E) ;FR(F)	MES_2:	MOV CMP JNC CMP	AL,[S1] AL,80H MES_1 AL,0DH	;CHECK IF 2 BYTES CODE
FRCOPY ENDP PROG ENDS				JZ ;	MES_SKIP2	;CHECK IF "CR/LF"
END			,CULUK	ATTRIBUT		
				MOV CMP	DX,1 AL,4DH	
	-2-				-1-	
17	-2-	""""=MACENTA	C11C 11C	·	-1-	
JZ INC CMP	CHG_COLOR DX	;"M"=MAGENTA	SJIS_JIS	_3: SUB	-1- AL,1FH	
	CHG_COLOR	;"M"=MAGENTA ;"C"=CYAN		SUB;		
INC CMP JZ INC CMP JZ INC	CHG_COLOR DX AL,43H CHG_COLOR DX AL,42H CHG_COLOR DX		;ADJUST	SUB;	AL,IFH E TO BOARD HARDWARE DX,I AL,I AX,I	
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	;"C"=CYAN	;ADJUST	SUB ; JIS CODE ; MOV SHL SHL SHL RCL SHL SHL	AL,1FH E TO BOARD HARDWARE DX,1 AL,1 AX,1 AX,1 DX,1 AX,1	
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 CHG_COLOR	;"C"=CYAN ;"B"=BLUE	;ADJUST	SUB ; JIS CODE ; MOV SHL SHL SHL RCL SHL RCL SHL SHL	AL, 1FH TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 AX, 1 AX, 1 AX, 1	
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,47H CHG_COLOR DX AL,42H CHG_COLOR CHG_COLOR	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW	;ADJUST	SUB ; JIS CODE ; MOV SHL SHL SHL RCL SHL RCL SHL RCL	AL,1FH E TO BOARD HARDWARE DX,1 AL,1 AX,1 DX,1 DX,1 DX,1 DX,1	
INC CMP JZ INC CMP JZ 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 CHG_COLOR DX AL,47H CHG_COLOR DX AL,52H	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN	;ADJUST	SUB ; JIS CODE ; NOV SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL RCL SHL MOV MOV	AL, 1FH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 DX, 1 AX, 1 DX, 1	;EAD2L=(?) ;EAD2H=(?) ;dAD2=0 ;¥=(?)
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ Z Z Z Z Z Z Z Z Z Z Z Z Z Z Z Z Z Z	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 ;"G"=GREEN ;"R"=RED	;ADJUST	SUB ; JIS CODE ; MOV SHL SHL SHL SHL RCL SHL RCL SHL RCL ; POP MOV MOV	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	;EAD2H=(?) ;dAD2=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,42H CHG_COLOR DX AL,4CH CHG_COLOR DS ES:WORD PTR PLANES,DX SI MES_2	:"C"=CYAN :"B"=BLUE :"Y"=YELLOW :"G"=GREEN :"R"=RED :"L"=BLACK ;PLANES=(?)	;ADJUST	SUB ; JIS CODE ; MOV SHL SHL SHL SHL RCL SHL RCL SHL RCL SHL RCL ; POP MOV MOV MOV MOV MOV MOV MOV TEST	AL.1FH E TO BOARD HARDWARE DX.1 AL.1 AX.1 DX.1 CS:WORD PTR EAD2L,AX DS:WORD PTR Y.CX BX.28 AX.8008H CS:FLAG,2	;EAD2H=(?) ;dAD2=0 ;X=(?) ;Y=(?) ;X=X+28 ; <a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;<!--!BUG!!-->&gt;</a_copy_ac>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP IZ CHG_COLOR: MOV SUB SUB SUB SUB SUB 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,4CH CHG_COLOR DS ES:WORD PTR PLANES,DX S1 MES_2 BX,2 CX,30 S1	;"C"=CYAN ;"B"=BLUE ;"Y"=YELLOW ;"G"=GREEN ;"R"=RED ;"L"=BLACK	;ADJUST	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 MOV MOV MOV MOV MOV MOV	AL.1FH 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 DX.1 DX.1 DX.1 DX.1 DX.1 DX.1 DX.1 DX.1 DX.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 AX.1 DX.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 AX.1 DX.1 AX.1 DX.1 AX.1 DX.1 AX.1 AX.1 DX.1 AX.1 AX.1 DX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.1 AX.2 AX.8008H	<pre>;EAD2H=(?) ;dAD2=0 ;X=(?) ;Y=(?) ;X=X+28 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;&lt;1!BUG1!&gt;&gt; ;CHECK IF "SLANT"=1/0 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD}</a_copy_ac></a_copy_ac></pre>
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,42H CHG_COLOR 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 MOV TEST JZ	AL, 1FH C TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 1 AX, 1 DX, 2 DX, 2	<pre>;EAD2H=(?) ;dAD2=0 ;x=(?) ;v=(?) ;x=x+28 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;&lt;1!BUG1!&gt;&gt; ;CHECK IF "SLANT"=1/0 ;<a_copy_ac> COPY</a_copy_ac></a_copy_ac></pre>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP ZZ INC CMP INC CMP INC CMP INC CMP 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,4CH CHG_COLOR 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	;ADJUST	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 MOV	AL, 1FH TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 DX, 1 CS: FURG PTR EAD2L, AX DS: WORD PTR Y, CABCH, DX AX, 28 AX, 8008H DS: WORD PTR DXX, -4 DS: WORD PTR COM, AX	<pre>;EAD2H=(?) ;dAD2=0 ;X=(?) ;Y=(?) ;X=x+28 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;&lt;11BUG11&gt;&gt; ;CHECK IF "SLANT"=1/0 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;SLANT</a_copy_ac></a_copy_ac></pre>
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 IC INC CMP IC INC CMP IC INC CMP IC INC CMP INC CMP INC CMP IC INC CMP INC CMP INC CMP INC CMP INC CMP INC CMP INC CMP INC CMP INC CMP INC CMP 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,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 SI SI MES_2 FT-JIS CODE TO JIS CODE	:"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 ROV MOV MOV MOV MOV MOV MOV MOV MOV MOV M	AL.1FH E TO BOARD HARDWARE DX.1 AL.1 AX.1 AX.1 DX.1 DS.WORD PTR EAD2L.AX DS.WORD PTR Y.CX BX.28 AX.8008H DS:WORD PTR DXX,-4 DS:WORD PTR CM.AX DS AX.5500H	<pre>;EAD2H=(?) ;dAD2=0 ;x=(?) ;v=(?) ;x=x+28 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;&lt;11BUG11&gt;&gt; ;CHECK IF "SLANT"=1/0 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;SLANT ;DXX=-4</a_copy_ac></a_copy_ac></pre>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ CMG_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,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	:"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 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.1FH 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 AX.1 DX.1 C.1 C.1 DX.1 AX.1 DX.1 DX.1 AX.1 DX.1 DX.1 AX.1 DX.	<pre>;EAD2H=(?) ;dAD2=0 ;X=(?) ;Y=(?) ;X=X+28 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;&lt;11BUG11&gt;&gt; ;CHECK 1F "SLANT"=1/0 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;SLANT ;DXX=-4 ;<com_flags></com_flags></a_copy_ac></a_copy_ac></pre>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP INC SUB INC 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,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 AL,AL	:"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 RCL SHL RCL SHL RCL SHL RCL ; POP MOV MOV MOV MOV MOV MOV MOV MOV MOV MOV	AL, 1FH TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 DX, 1 DX, 1 AX, 1 DX, 2 DX, 2 DX, 2 DX, 2 DX, 2 DX, 2 DX, 2 DX, 2 DX, 4 DX, -4 DX, -4 DX, 4 DX, 5 DOH DX, 4 DX, 4 DX, 5 DOH DX, 4 DX,	<pre>;EAD2H=(?) ;dAD2=0 ;X=(?) ;Y=(?) ;X=X+28 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;&lt;11BUG11&gt;&gt; ;CHECK 1F "SLANT"=1/0 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;SLANT ;DXX=-4 ;<com_flags></com_flags></a_copy_ac></a_copy_ac></pre>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP IC INC CMP IC 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,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 MES_3: ;	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 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, 1 AX, 1 DX, 2 AX, 8008H DS:WORD PTR DXX, -4 DS:WORD PTR CM, AX DS, AX S1 S1	<pre>;EAD2H=(?) ;dAD2=0 ;x=(?) ;v=(?) ;x=x+28 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;&lt;11BUG11&gt;&gt; ;CHECK IF "SLANT"=1/0 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;SLANT ;DXX=-4 ;<com_flags> ;MESSAGE SEGMENT</com_flags></a_copy_ac></a_copy_ac></pre>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP INC CMP INC CMP INC CMP 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,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,AH	:"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 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, 1 AX, 1 DX, 2 AX, 8008H DS:WORD PTR DXX, -4 DS:WORD PTR CM, AX DS, AX S1 S1	<pre>;EAD2H=(?) ;dAD2=0 ;x=(?) ;v=(?) ;x=x+28 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;&lt;11BUG11&gt;&gt; ;CHECK IF "SLANT"=1/0 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;SLANT ;DXX=-4 ;<com_flags> ;MESSAGE SEGMENT</com_flags></a_copy_ac></a_copy_ac></pre>
INC CMP JZ SINC INC SUB INC INC SUB INC SUB SINC INC SUB SINC INC SUB SINC INC SUB SINC INC SUB SINC INC SUB SINC SUB SINC SUB SINC SUB SUB SUB SUB SUB SUB SUB SUB SUB 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,47H CHG_COLOR DX AL,42H CHG_COLOR DX AL,42H 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 TO JIS CODE AX.[SI] AL,AH AH,AH AL,AL SJIS_JIS_1 2	:"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 ROV MOV MOV MOV MOV MOV MOV MOV M	AL, IFH E TO BOARD HARDWARE DX, 1 AL, 1 AX, 1 AX, 1 DX, 1 DX, 1 AX, 1 DX, 1 DX, 1 AX, 1 DX, 2 AX, 8008H DS:WORD PTR DXX, -4 DS:WORD PTR CM, AX DS, AX S1 S1	<pre>;EAD2H=(?) ;dAD2=0 ;x=(?) ;v=(?) ;x=x+28 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;&lt;11BUG11&gt;&gt; ;CHECK IF "SLANT"=1/0 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;SLANT ;DXX=-4 ;<com_flags> ;MESSAGE SEGMENT</com_flags></a_copy_ac></a_copy_ac></pre>
INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP JZ INC CMP IC CMC INC INC INC INC INC INC INC INC INC IN	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,52H CHG_COLOR DX AL,52H 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_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 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, 1 AX, 1 DX, 2 AX, 8008H DS:WORD PTR DXX, -4 DS:WORD PTR CM, AX DS, AX S1 S1	<pre>;EAD2H=(?) ;dAD2=0 ;x=(?) ;v=(?) ;x=x+28 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;FAST=0 &lt;&lt;11BUG11&gt;&gt; ;CHECK IF "SLANT"=1/0 ;<a_copy_ac> COPY ;ESE=0,REV=0,ROT=0,{MD} ;SLANT ;DXX=-4 ;<com_flags> ;MESSAGE SEGMENT</com_flags></a_copy_ac></a_copy_ac></pre>

-2-

-3-

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 :PMAX=4 MOV 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 JNZ MOV :BODY (H)=28 ;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 EXE 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 EAD2H,16H TEST ;PLANES=0 ;<A\_COPY\_AC> 90\_COPY ;ESE=0,REV=0,ROT=0,{MD} CS:WORD PTR WORK\_1,0 JNZ CLIP DEMO\_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 SI,CS:WORD PTR WORK\_2 :X=(?) CLIP\_DEM0\_3: MOV DS:WORD PTR STATUS,1 CHR\_DRAW\_1\_90: TEST DS:WORD PTR EAD2L,DX DS:WORD PTR X,SI ;EAD2L=(?) MOV JNZ CLIP DEMO\_3 ;X=(?);Y=(?) DS:WORD PTR MAGETC, 2FFH MOV MOV DS:WORD PTR Y,DI DS:WORD PTR DHH,23 MOV :DH=23 CLIP\_DEMO\_EXE: MOV DS:WORD PTR DV,23 DS:WORD PTR MOD10,97H MOV :DV=23 MOV BX.70ACH ;MOD1=9,MOD0=7 MOV CS:WORD PTR WORK\_1 ;PLANES+1 --> PLANES TRAFILL\_EXE\_H INC CALL MOV AX.CS:WORD PTR WORK 1 MOV AX,40 DS:WORD PTR PLANES, AX ;PLANES=(?) WAIT MOV CALL DS:WORD PTR COM, BX <COM. FLAGS> DMCLO\_C\_WAIT MOV CALL ;CHARACTER CODE + 40H ADD DX.40H RET DI,CS:WORD PTR WORK\_5 ;Y+(?) ADD 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: CHR\_DRAW\_DEMO\_H CALL FND OF BACK-GROUND CLIP\_DEM0\_1: TEST DS:WORD PTR STATUS,1

;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: DS:WORD PTR STATUS.1 TEST PUBLIC GET\_PUT\_DEMO, PUT\_DATA\_TRAN PUT\_DEMO\_EXE\_1 JNZ. DS:WORD PTR DHH,255 DS:WORD PTR DV,255 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 DS:WORD PTR X,AX MOV EXTRN MOV AX,CS:[SI] GETPUT PROC NEAR INC SI INC SI DS:WORD PTR V.AX MOV • • • • DS:WORD PTR PLANES,7 < GET/PUT DEMO > MOV DS:WORD PTR PMAX,4 DS:WORD PTR MODIO,0 MOV MOV MOV DS:WORD PTR COM, BX GET PUT DEMO: PUSH DS INIT\_PUT\_DATA AX,6COOH CALL MOV MOV DS,AX CX,3000H PUT\_DATA\_TRAN PUT\_GET\_DEMO\_1 PUT\_GET\_DEMO\_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 AX, WORD PTR AGDC\_SEG 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: PUT\_DATA\_TRAN\_1 1.00P PUT\_DEMO\_EXE POP CALL SI CALL GET\_DEMO\_EXE RET MOV AX.10 CALL WAIT GET\_DEMO\_EXE: DS:WORD PTR STATUS.1 RET TEST GET\_DEMO\_EXE JNZ DS:WORD PTR DHH,255 DS:WORD PTR DV,255 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 MOV DS:WORD PTR PDISPSL,0 :PDISPSL=0 ;<GET\_C> DS:WORD PTR COM, 9AOEH MOV PUBLIC ENLARGE\_DEMO ;{MSD} ;256 X 256 X 3 PLANES MOV CX.3000H STATUS:WORD, PITCHS:WORD, PMAX:WORD, MODIO:WORD EXTRN AX,6COOH SEGMENT "MAAJAN" X:VORD, Y:WORD, XS:WORD, YS:WORD, DHH:WORD, DV:WORD MAGETC:WORD, COM:WORD, PDISPSH:WORD, PDISPSL:WORD MOV EXTRN MOV ES,AX EXTRN CX,3000H ;256 X 256 X 3 PLANES MOV GET DATA TRAN: ENLARGE PROC NEAR ASSIGN GET/PUT PORT SI.3EH MOV MOV D1.0 GET\_DATA\_TRAN\_1 < ENLARGEMENT COPY DEMO > MOV AX,[SI] MOV ES:[DI],AX INC DI INC DI ENLARGE DEMO: GET\_DATA\_TRAN\_1 LOOP DS:WORD PTR STATUS,1 TEST RET INZ ENLARGE\_DEMO DS:WORD PTR PDISPSH,1 MOV GET\_PUT\_DATA: DS:WORD PTR PDISPSL,0 MOV 150,700, 960,445 960,315, 150, 60 ;1.--> 2. X,Y ;3.--> 4. X,Y DW DS:WORD PTR PITCHS,46H DS:WORD PTR PMAX,4 MOV DW MOV DS:WORD PTR MODIO.0 MOV SI. OFFSET ENLARGE DATA MOV GETPUT ENDP MOV CX,16 ENDS PROG DX.1FFH MOV END BX.849FH MOV ENLARGE\_EXE: DS:WORD PTR STATUS,1 TEST INZ ENLARGE EXE AX.CS:[SI] MOV INC SI INC SI MOV DS:WORD PTR X,AX MOV AX,CS:[SI] INC SI INC MOV DS:WORD PTR Y,AX MOV AX.CS:[SI] INC SI INC SI MOV DS:WORD PTR XS,AX MOV DS:WORD PTR YS,374 MOV AX.CS:[SI] INC INC S DS:WORD PTR DHH,AX

-3-

-1-

MOV

REPETITION COUNTS

;CHECK IF PPBUSY=1/0

;<PUT\_C> ;ROT=0,{MSD}

;DH=255

:DV=255

;X=(?)

:Y=(?)

;PLANES=7

;<COM.FLAG>

;SEGMENT "MAAJAN"

;CHANGE ROTATION

:256 X 256 X 3 PLANES

ASSIGN GET/PUT PORT

;CHECK IF PPBUSY=1/0

CHECK IF PPBUSY=1/0 ;PDISPSH=1

REPETITION COUNTS

;<A\_COPY\_CC> ES\_COPY

;CHECK IF PPBUSY=1/0

:MAGH=15.MAGV=15

;ROT=0,{MSD}

;X=(?)

;Y=(?)

;XS=(?)

;YS=374

;DHH=(?)

:PDISPSL=0

;PMAX=4 :MOD1=0,MOD0=0

;PITCHS=46H

:DH=255

:DV=255

-2-

'PROG'

:PMAX=4 ;MOD1=0,MOD0=0

NOV INC INC MOV MOV SUB ENLARGE_DATA: DW DW DW DW DW DW DW DW DW DW DW DW DW	Ax, cs:[s1] S1 S1 S1 S1 S1 S2 S1 S2 S2 S2 S2 S2 S2 S2 S2 S2 S2	:DV=(?) :MAGETC=(?) ; <com.flag> :MAGH-1&gt;MAGH :MAGV-1&gt;MAGH :MAGV-1&gt;MAGV :X,Y,XS,DHH,DV 16/16 :X,Y,XS,DHH,DV 16/13 :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 16/10 :X,Y,XS,DHH,DV 16/17 :X,Y,XS,DHH,DV 16/16 :X,Y,XS,DHH,DV 16/15 :X,Y,XS,DHH,DV 16/15 :X,Y,XS,DHH,DV 16/15 :X,Y,XS,DHH,DV 16/15 :X,Y,XS,DHH,DV 16/15 :X,Y,XS,DHH,DV 16/11</com.flag>	PGROUP PROG SHRINK SHRINK	ASSUME ; PUBLIC ; EXTRN EXTRN EXTRN EXTRN EXTRN EXTRN ; PROC ; CALL MOV MOV CALL POP CALL CALL MOV MOV CALL ; MOV MOV	SHRINK PROG BYTE PUBLIC 'PROG' CS:PGROUP SHRINK_DEMO, SHRINK_PUT, SHRIN MESSAGE_VAIT:NEAR, DMCLO_C_VAI STATUS:WORD, XS:WORD, MAGETC:WORD, DH:WORD, DV:WORD, MAGETC:WORD, PMAX:WORD, PLANES:WORD, PITCHS PDISPSH:WORD, EAD2H:WORD, dAD2 NEAR K COPY DEMO > SHRINK_EXE SI SI,21 MESSAGE_VAIT SI,21 MESSAGE_VAIT SI,21 MESSAGE_VAIT SI,21 MESSAGE_VAIT SI,21 MESSAGE_VAIT SI,10FH CX,16 DS:WORD PTR STATUS,1 SHRINK_MAGH DS:WORD PTR XS,10 DS:WORD PTR XS,10 DS:WORD PTR Y,499 DS:WORD PTR Y,499 DS:WORD PTR MAGETC,DX DS:WORD PTR MODIO,0 DS:WORD PTR MODIO,0 DS:WORD PTR COM,840FH DX,10H	T:NEAR, PUT_DATA_TRAN:NEAR X:WORD, Y:WORD , MODIO:WORD, COM:WORD :WORD, PDISPSL:WORD
	-2-				-1-	
LOOP ; MOV MOV SHRINK_MAGV: TEST JNZ MOV MOV MOV MOV MOV MOV MOV MOV	SHRINK_MAGH DX, IFOH CX, I6 DS:WORD PTR STATUS, I SHRINK_MAGV DS:WORD PTR XS, I0 DS:WORD PTR YS, 739 DS:WORD PTR YZ, 739 DS:WORD PTR YZ, 739 DS:WORD PTR YZ, 739 DS:WORD PTR DV, 214 DS:WORD PTR MACTC, DX DS:WORD PTR COM, 840FH DX SHRINK_MAGV DS:WORD PTR STATUS, I SHRINK_PUT DS:WORD PTR MAK, 10 DS:WORD PTR DV, 214 DS:WORD PTR DV, 214 DS:WORD PTR DV, 214 DS:WORD PTR DV, 214 DS:WORD PTR V, 739 DS:WORD PTR V, 739 DS:WORD PTR MAX, 4 DS:WORD PTR COM, 880BH DS AX, 5COOH DS, XX CX, 7740 PUT DATA_TRAN DS SWORD PTR PMAX, 4 DS:WORD PTR PMAX, 4 DS:WORD PTR PMAX, 4 DS:WORD PTR STATUS, 3 SHRINK_EXE DS:WORD PTR PMAX, 4 DS:WORD PTR	<pre>;MAGH=15,MAGV=0 ;MAGV=0&gt; 15 :CHECK IF PPBUSY=1/0 ;XS=10 ;YS=739 ;DH=575 ;DV=214 ;MAGH=(?),MAGV=OFH ;<a_copy_cc> ES_COPY ;REV=0,ROT=0,{MSD} ;MAGV+1&gt; MAGV :CHECK IF PPBUSY=1/0 ;V=739 ;MAGV+1&gt; MAGV :V=739 ;MAX=4 ;PLANES=6 ;MOD1=9,MOD0=7 ;<rut_c> ;REV=0,ROT=0,{MD} ;SEGMENT "NECDEM0.PAT" ;576 X 215 X 1 PLANE :CHECK IF PPBUSY=1/0 ;PMAX=4 ;PLANES=6 ;PLTCNS=48H ;PLISPSH=1 ;PLISPSH=1 ;PLISPSH=1 ;PLISPSH=1</rut_c></a_copy_cc></pre>	; SHRINK SHRINK PROG	DW DW DW DW DW DW DW DW ;	SHRINK_EXE_1 AX,CS:[SI] SI DS:WORD PTR X,AX AX,CS:[SI] SI DS:WORD PTR Y,AX AX,CS:[SI] SI DS:WORD PTR MAGETC,AX DS:WORD PTR MAGETC,AX DS:WORD PTR EAD2L,10 DS:WORD PTR CAD2L,10 DS:WORD PTR CAD2L,10 DS:WORD PTR EAD2L,10 DS:WORD PTR EAD2L,10 DS	<pre>;CHECK IF PPBUSY=1/0 ;X=(?) ;Y=(?) ;MAGETC=(?) ;EAD2H=1 (GO) ;GAD2=10 (GO) ;GAD2=10 (GO) ;DH=575 ;DV=214 ;<a_copy_ac> ES_COPY ;REV=0,ROT=0,{MD}</a_copy_ac></pre> : 15/16 X,Y,MAGETC ;10 /16 X,Y,MAGETC ;0 8/16 X,Y,MAGETC ;3, 2/16 X,Y,MAGETC ;3, 2/16 X,Y,MAGETC ;1/16 X,Y,MAGETC

	NAME	PUSH_PO	P	
PGROUP	GROUP	PROG		
PROG	SEGMENT			'PROG'
	ASSUME	CS:PGR0	UP	
	;			
	PUBLIC	PUSH_EX	E, POP	EXE
	;			
PUSH PO	Р	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-9800 series replacing the entry points.

The PC-9800 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  $\mu$ PD72120 Advanced Graphics Display Controller LSI from system side exhibiting concrete examples.

To see the details, zoom up.