```
.TITLE ALDISP - ALIENS DISPLAY FUNCTIONS
        .SBTTL ******************************************
        .SBTTL *                                        *
        .SBTTL *MODULE          ALDISP                  *
        .SBTTL *PROGRAMMER      DAVE THEURER            *
        .SBTTL *FUNCTION        ALIENS DISPLAY FUNCTIONS*
        .SBTTL *                                        *
        .SBTTL ******************************************

        .NLIST
        .INCLUDE ALCOMN
        .PAGE
        .LIST
        .PAGE
        .SBTTL GLOBALS
                                ENTRY POINTS
        .GLOBL VGADD,VGJSRL,VGVCTR,GETDSP,LDRDSP,VGCNTR,SWAPVG,VGDOT
        .GLOBL VGADD2,VGHEXZ,DISPLA,VGHALT,INIDSP,JSRDOT,VGADD3
        .GLOBL DPRSTA,D2GAME
        .GLOBL INFO,RQRDSP,DSPHOL
        .GLOBL VGRTSL,DGOVER,DPLPLA,PICLO,PICHI
        .GLOBL INITEM,VGYABS,VGYAB1,INICOL
                                TABLES
        .GLOBL INVERSE,INVEXP,D07MSK,BOXPRO,LOGPRO,PTTANP,PTTANF,PTSPLF
        .GLOBL VGBRIT,VGLIST,VGSIZE,XCOMP,D70MSK,SCALE,DSPSYS
        .GLOBL VGSCA1,VGVTR1,VGVTR2,VGSCAL,VGSTAT,VGSTA1
        .GLOBL PTSTR1,PTEXP1,PTCURS,PTSPI1,PTTANK,PTESHO,PTSPLA,PTSPAR
        .GLOBL BCINFO,BCCURS,BCSHOT,BCINVA,BCEXPL,BCNYMP,BCINFO,BCENEL
        .GLOBL BUFBSL,BUFBSH,BUFASL,BUFASH,JMPMAL,JMPMAH,BCWELL,BUFSWL,BUFSWH
        .GLOBL JMPALO,JMPAHI,JMPBLO,JMPBHI,BCSTAR,BFASTA,BFBSTA,PTFUSE,PTFUSX
        .GLOBL PPSPXI,CPSPXI,FPSPXI,KILLER
        .GLOBL QCHKS6,QCHKS7,QCHKS8
RATS     0C0
MZCOLO   8                      NEW COLOR STAT BIT MASK
MZBRIT   0                      NEW INTENSITY STAT BIT

        .SBTTL DISPLAY-MAINLINE

 FUNCTION       USING THE DATABASE SET UP BY THE GAME PORTION OF THE PROGRAM,
                BUILD, IN THE AVAILABLE BUFFER, THE VECTOR GENERATOR DISPLAY LIST

 INPUT          DATABASE

SPACG 0                 SUPPRESS SPACE GAME CODE
        .NLIST CND
        .CSECT
DISPLAY
        JSR INIMAT              SET UP MATH BOX
        LDA VECRAM
        CMP JMPMAL+4
        IFEQ                    TRYING TO HALT
        LDA SPARE3              YES.
        IFEQ                    HALT YET
        RTS                     NO. GO AWAY
        ENDIF
        ENDIF
        LDA QDSTATE
        CMP I,CDPLAY
        IFNE                    ANYTHING BUT PLAY STATE
        LDA I,BCINFO            YES. DEFAULT TO INFO BUFFER
```

```
        JSR SBCLOG
        JSR BIGTEX
        IFCC                    SET UP LARGE BUFFER
                                BUFFER AVAILABLE FILL IT
        JSR DSTATE              EXECUTE DISPLAY STATE
ZATVG2
        LDA SECUVY
        IFNE                    ATARI ON SCREEN
        LDY I,27                YES. VERIFY
        LDA I,0E
        SEC
        BEGIN
        SBC NY,SECUVG
        DEY
        MIEND
        TAY
        IFNE
        EOR I,0E5
        ENDIF
        IFNE
        EOR I,029
        ENDIF
        STA QT3
        ENDIF
        ENDIF
        LDA I,BCINFO
        JSR SBCSWI
        LDA JMPMAL+2
        STA VECRAM
        LDA JMPMAH+2
        STA VECRAM+1
        ELSE
        JMP DENORM              PLAY STATE
        ENDIF
        RTS
        .PAGE
       .PAGE
        .SBTTL DISPLAY STATE EXECUTOR
DSTATE
        LDX QDSTATE
        LDA X,DROUTAD+1
        PHA
        LDA X,DROUTAD
        PHA
NOOPR   RTS
DROUTAD
        .WORD DENORM-1          GAME PLAY - TOP OF WELL, DOWN THE TUBE
        .WORD DSPSYS-1          SYSTEM CONFIGURATION
        .WORD DSBOOM-1          GAME PLAY - BOOM
        .WORD GETDSP-1          DATA ENTRY - HI SCORE INITIALS
        .WORD RQRDSP-1          DATA ENTRY - RANKING
        .WORD LDRDSP-1          INFO ONLY - HI SCORE TABLE
        .WORD DGOVER-1                      GAME OVER PLAYER X
        .WORD DPLPLA-1                      PLAY PLAYER X
        .WORD DPRSTA-1           PRESS START
        .WORD BOXPRO-1          LOGO BOX
        .WORD LOGPRO-1          LOGO
        .WORD D2GAME-1          2 GAME MINIMUM
DROUTEN
        .PAGE
```

```
        .SBTTL  DISPLAY-GAME PLAY MAINLINE

DENORM                      DISPLAY CURSOR
        LDA I,BCCURS
        JSR SBCLOG
        JSR DSPCUR
        LDA I,BCCURS
        JSR SBCSWI
                                    DISPLAY CHARGES
        LDA I,BCSHOT
        JSR SBCLOG
        JSR DSPCHG
        LDA I,BCSHOT
        JSR SBCSWI
                                    DISPLAY INVADERS
        LDA I,BCINVA
        JSR SBCLOG
        JSR DSPINV
        LDA I,BCINVA
        JSR SBCSWI
                                    DISPLAY EXPLOSIONS
        LDA I,BCEXPL
        JSR SBCLOG
        JSR DSPEXP
        LDA I,BCEXPL
        JSR SBCSWI
                                    DISPLAY NYMPHS
        LDA I,BCNYMP
        JSR SBCLOG
        JSR DSPNYM
        LDA I,BCNYMP
        JSR SBCSWI
                                    DISPLAY INFORMATION  SCORES, MSGS, ETC.
        LDA I,BCINFO
        JSR SBCLOG
        JSR INFO
ZATVG1
        LDA QSTATUS
        IFPL                        ATTRACT
        LDA I,0F2                           YES. ATARI BETTER BE ON SCREEN
        CLC
        LDY I,39.
        BEGIN
        ADC NY,SECUVG
        DEY
        MIEND
        STA QT6                     SAVE RESSLT  SHOULD BE 0
        ENDIF
        LDA I,BCINFO
        JSR SBCSWI
                                    DISPLAY WELL
        JSR DSPWEL                  DISPLAY WELL
        LDA I,BCENEL                DISPLAY ENEMY LINES
        JSR SBCLOG
        JSR DSPENL
        LDA I,BCENEL
        JSR SBCSWI
        LDA I,BCSTAR                DISPLAY STAR FIELD
        JSR SBCLOG
        JSR DSTARF
```

```
        LDA I,BCSTAR
        JSR SBCSWI
        LDA I,0
        STA ROTDIS

        LDA JMPMAL                  SET MASTER POINTER TO JSRL
        STA VECRAM                  LIST FOR SUBLISTS CREATED ABOVE
        LDA JMPMAH
        STA VECRAM+1

        RTS
        .PAGE
        .SBTTL  BUFFER CONTROL

 INPUT ACC SUB BUFFER GROUP INDEX CODE
 OUTPUT VGLIST 2    VGY SET UP TO VACANT BUFFER
        ACC,X,Y DESTROYED
SBCLOG
        TAX                         SET UP VECTOR RAM POINTS TO
        ASL                         UNUSED BUFFER
        TAY
        LDA X,BUFACT
        IFEQ                        BUFFER A OR B ACTIVE
        LDX Y,BUFBSL                A IS CTIVE. BUILD IN B
        LDA Y,BUFBSH
        ELSE
        LDX Y,BUFASL                B IS ACTIVE. BUILD IN A
        LDA Y,BUFASH
        ENDIF
        STX VGLIST
        STA VGLIST+1
        LDA I,0
        STA VGY
        RTS
                                    OPPOSITE OF SBCLOG-PLACE PTR
                                    TO ACTIVE BUFFER INTO INDYLO
SBCACT  TAX
        ASL
        TAY
        LDA X,BUFACT
        IFEQ
        LDX Y,BUFASL                A IS ACTIVE
        LDA Y,BUFASH
        ELSE
        LDX Y,BUFBSL                B IS ACTIVE
        LDA Y,BUFBSH
        ENDIF
        STX INDYLO
        STA INDYLO+1
        LDA I,0
        STA VGY
        RTS
 INPUT ACC SUBBUFFER GROUP INDEX CODE
 OUTPUT RTS ADDED TO END OF NEWLY BUILT BUFFER
        SWITCH SET TO POINT TO NEW BUFFER
        BUFACT X FLIPPED INDICATING NEW BUFFER ACTIVE
SBCSWI  PHA
        JSR VGRTSL                  INSERT RTSL AT END OF BUFFER
        PLA
        TAX
```

```
        ASL
        TAY
        LDA Y,BUFSWL            SET UP SWITCH LOCATION
        STA INDYLO
        LDA Y,BUFSWH
        STA INDYHI
        LDA X,BUFACT
        EOR I,01
        STA X,BUFACT
        IFEQ                    WHICH IS THE NEW BUFFER TO DISLAY
        LDA Y,JMPALO            BUFFER A
        LDX Y,JMPAHI
        ELSE
        LDA Y,JMPBLO            BUFFER B
        LDX Y,JMPBHI
        ENDIF
        LDY I,0                 UPDATE SWITCH TO PT TO
        STA NY,INDYLO           NEW BUFFER
        TXA
        INY
        STA NY,INDYLO
        RTS

BIGTEX                          ASSIGN LARGE BUFFER FOR TEXT
        LDA JMPMAL+2
        CMP VECRAM
        IFNE                    BEEN HERE BEFORE
        STA VECRAM              NO. SET UP MASTER POINTER FOR TEXT ONLY.
        SEC
        RTS                     EXIT
        ENDIF
        LDA BUFACT              YES. INSERT JMP TO AREA WITH MORE ROOM.
        IFEQ
        LDX I,02                BIG AREA 1  1ST HALF OF VECRAM
        ELSE
        LDX I,08                BIG AREA 2  2ND HALF OF VECRAM
        ENDIF
        LDA X,JMPALO
        LDY I,0
        STY SECUVY
        STA NY,VGLIST
        INY
        LDA X,JMPAHI            INSERT JMPL TO AREA WITH MORE ROOM
        STA NY,VGLIST
        LDA X,BUFASL            POINT VGLIST AT NEW AREA
        STA VGLIST
        LDA X,BUFASH
        STA VGLIST+1
        CLC
        RTS
        .PAGE
        .SBTTL DISPLAY-WELL
CSUSTA   3
CSUINT   1
DSPWEL
        LDA ROTDIS
        IFNE                    REBUILD WELL
        LDA I,BCWELL
        JSR SBCLOG
        JSR BLDWEL              YES
```

```
        LDA I,BCWELL
        JSR SBCSWI
        ENDIF
        LDA I,BCWELL            SET UP PTR TO ACTIVE WELL BUFFER
        JSR SBCACT

        .SBTTL  DISPLAY-SPOKE COLORS

        .SBTTL  DISPLAY-SPOKE PULSE STATUS
        LDA I,0
        LDX I,NLINES-1
        BEGIN                   LOOP FOR EACH SPOKE
        STA X,SPOKST            CLEAR SPOKE PULSE STATUS
        DEX
        MIEND
        LDA CURMOD
        IFPL                    CURSOR AT TOP
        LDX WINVMX              YES.
        BEGIN                   LOOP FOR EACH INVADER
        LDA X,INVAY
        IFNE                    ACTIVE INVADER
        LDY I,0                 YES. DEFAULT
        LDA X,INVAC1
        AND I,INVABI
        CMP I,ZABPUL
        IFEQ                    PULSAR
        INY                     YES. SET PULSAR BIT D0
        STY TEMP0
        LDA X,INVAC1            YES
        AND I,INVMOT
        IFEQ                    FLIPPING
        LDA PULSON              NO.
        IFPL                    PULSARS ON
        LDA X,INVAY             YES.
        CMP PULPOT
        IFCC                    POTENT PULSAR
        INC TEMP0               YES. SET PULSE BIT D1
        INC TEMP0

        ENDIF
        ENDIF
        LDA TEMP0               SET CCW LEG STATUS
        LDY X,INVAL2
        ORA Y,SPOKST
        STA Y,SPOKST
        ENDIF
        LDY X,INVAL1
        LDA TEMP0
        ORA I,80                SET BASE BIT
        ORA Y,SPOKST
        STA Y,SPOKST
        ENDIF
        ENDIF
        DEX
        MIEND
        ENDIF
        LDA I,WELCOL            DEFAULT WELL COLO
        LDY SUZTIM
        IFNE
        IFPL                    SUPERZAPPER ACTIVE
```

```
        LDA QFRAME              YES. SUPERZAPPER IS DEFAULT
        AND I,7
        CMP I,7
        IFEQ
        LDA I,1                 NO BLACK
        ENDIF
        ENDIF
        ENDIF
        STA TEMP0               DEFAULT COLOR
        LDY I,-1
        LDX I,-1
        STX TEMP3               DEFAULT NO BONUS FLASH
        LDA CURSY
        IFNE                    C URSOR ALIVE
        LDA CURSL2
        IFPL
        LDX CURSL1              YES.
        LDY CURSL2
        ENDIF
        ENDIF
        STX TEMP1               SAVE FLASLIGHT SPOKES
        STY TEMP2
        LDA BOFLASH
        IFPL                    BONUS FLASH
        AND I,0E                YES. SET BASE COLOR
        LSR
        STA TEMP3
        DEC BOFLASH
        ENDIF
        LDX I,NLINES-1
        BEGIN                   LOOP FOR EACH SPOKE
        LDY I,WELCOL            DEFAULT WELL COLOR
        LDA X,SPOKST
        IFNE                    PULSE
        AND I,2
        IFNE                    YES. PULSING
        LDA QFRAME
        AND I,1
        TAY
        ENDIF
        ELSE
        CPX TEMP1               NO.
        IFNE
        CPX TEMP2
        ENDIF
        IFEQ                    NO. CURSOR FLASHLIGHT
        LDY I,CURCOL            YES. CURSOR COLOR
        ELSE
        LDA BOFLASH             NO.
        IFPL                    BONUS FLASH
        TXA                     YES. BONUS COLOR
        CLC
        ADC TEMP3               PLUS BASE COLOR
        AND I,7                 MOD 8
        CMP I,7
        IFEQ
        LDA I,3                 NO BLACK
        ENDIF
        TAY
        ELSE
```

```
        LDY TEMPO                   NO. USE DEFALT COLOR
        ENDIF
        ENDIF
        ENDIF
        TYA
        LDY X,STALOC
        STA NY,INDYLO
        DEX
        MIEND
        LDX I,NLINES-1              YES. REDO TOP RUNGS
        BIT WELTYP
        IFMI
        DEX                         PLANAR, SO 1 LESS RUNG
        ENDIF
        BEGIN                       LOOP FOR EACH RUNG
        LDY I,0C0                   DEFAULT ON
        LDA X,SPOKST
        IFMI                        PULSAR
        LDY I,0                     YES. TURN OFF
        ENDIF
        STY PZL
        LDY X,RUNLOC
        LDA NY,RUNGVG
        AND I,1F
        ORA PZL
        STA NY,RUNGVG
        DEX
        MIEND
        RTS
 OFFSETS INTO WELL SUBROUTINE OF COLOR STATS FOR EACH LINE
STALOC  .BYTE 0A8,9C,92,86,7C,70,66,5A,50,44,3A,2E,24,18,0E,2,0B2
 OFFSETS INTO WELL SUBROUTINE  +0FE  OF COLOR STATS FOR EACH TOP RUNG
RUNLOC  .BYTE 3B,37,33,2F,2B,27,23,1F,1B,17,13,0F,0B,07,03,3F
CHKSM6          .BYTE QCHKS6
        .PAGE
        .SBTTL  DISPLAY-NYMPHS
IEYL 4
DSPNYM
        LDY I,NYMCOL
        STY COLOR
        LDA I,MZCOLO
        JSR VGSTAT
        LDX I,XADJL
        JSR VGYAB1                  POSITION BEAM AT VANISH PT.
        LDA I,18.
        STA PXL                     MAX # DISPLAYABLE
        LDX I,NNYMPH-1
        STX INDEX1
        LDY I,0
        BEGIN                       LOOP FOR EACH NYMPH
        LDX INDEX1
        LDA X,NYMPY
        IFEQ                        NYMPH ACTIVE
        JMP NONYM                   NO. SKIP IT
        ENDIF
        CMP I,50                    YES.
        IFCS                        SKIP EVERY OTHER ONE PAST THIS DEPTH
        DEC INDEX1
        ENDIF
        PHA
```

```
        AND I,3F                FAKE PROJECTION  USE NYMPH DEPTH TO GET SCALES
        STA NY,VGLIST           LINEAR SCALE
        PLA
        ROL
        ROL
        ROL
        AND I,3
        CLC
        ADC I,1
        ORA I,70
        INY
        STA NY,VGLIST           BINARY SCALE
        INY
        LDA X,NYMPL             GET NYMPH LINE
        TAX
        LDA X,LIFSZL            VECTOR TO NYMPH
        SEC
        SBC ZADJL
        STA SZL
        STA NY,VGLIST           Z LSB
        INY
        LDA X,LIFSZH
        SBC ZADJL+1
        STA SZH
        AND I,1F
        STA NY,VGLIST           Z MSB
        INY
        LDA X,LIFSXL
        STA SXL
        STA NY,VGLIST           X LSB
        INY
        LDA X,LIFSXH
        STA SXH
        AND I,1F
        STA NY,VGLIST           X MSB
        INY                     DISPLAY A DOT
        LDA I,0
        STA NY,VGLIST           0 Z LSB
        INY
        STA NY,VGLIST           0 Z MSB
        INY
        STA NY,VGLIST           0 X LSB
        LDA I,0A0
        INY
        STA NY,VGLIST           BRIGHTNESS, 0 X MSB
        INY
        LDA SZL                 DRAW VECTOR BACK TO FAKE V.P.
        EOR I,0FF
        CLC
        ADC I,1
        STA NY,VGLIST           Z LSB
        INY
        LDA SZH
        EOR I,0FF
        ADC I,0
        AND I,1F
        STA NY,VGLIST           Z MSB
        INY
        LDA SXL
        EOR I,0FF
```

```
        CLC
        ADC I,1
        STA NY,VGLIST           X LSB
        INY
        LDA SXH
        EOR I,0FF
        ADC I,0
        AND I,1F
        STA NY,VGLIST           X MSB
        INY
        CPY I,0F0
        IFCS                    VGLIST LSB INDEX MAXING OUT
        DEY
        JSR VGADD               YES. UPDATE VGLIST
        LDY I,0                 RESET LSB INDEX
        ENDIF
        DEC PXL                 EXIT EARLY IF MAX
        BMI EXCESS              LIMIT REACHED
NONYM   DEC INDEX1
        MIEND                   EXIT LOOP AFTER LAST NYMPH
EXCESS  TYA
        IFNE
        DEY
        JSR VGADD               UPDATE VGLIST
        ENDIF
ZQATLI          LDA QT1
        IFNE
        LDA WAVEN1
        CMP I,10.
        IFCS
        LDA I,7A
        STA FRTIMR
        ENDIF
        ENDIF
        LDA I,1
        JMP VGSCA1
VGDOT   PHA
        LDY I,0                 DRAW A DOT
        TYA
        STA NY,VGLIST
        INY
        STA NY,VGLIST
        INY
        STA NY,VGLIST
        INY
        PLA
        STA NY,VGLIST
        LDA I,4                 UPDATE DISPLAY POINTER
        CLC
        ADC VGLIST
        STA VGLIST
        IFCS
        INC VGLIST+1
        ENDIF
        RTS
        .PAGE
       .SBTTL  DISPLAY-CURSOR
DSPCUR
        LDA I,CURCOL
        STA COLOR
```

```
        LDA CURSY
        IFNE
        CMP I,ILINDDY
        IFCC                        AT BOTTOM
        STA PYL                     NO. DEPTH
        STA TEMPY
        LDA CURSL2
        CMP I,81
        IFNE                        DON T DISPLAY BLASTED CURSOR
        LDY CURSL1                  CURSOR S WELL LINE #S
        LDA CURSPO                  GET CURSOR POSITION BETWEEN LINES
        LSR
        AND I,07
        CLC
        ADC I,CNCURS                ADD IN BASE PIC #
        JSR ONELIN                  DRAW LINE
        ENDIF
        ENDIF
        ENDIF
        RTS
       .SBTTL  DISPLAY-INVADERS     MAINLINE
DSPINV
        LDA CURMOD
        IFPL                        CURSOR AT TOP
        LDX I,NINVAD-1              YES
        STX INDEX1
        BEGIN                       LOOP FOR EACH INVADER
        LDX INDEX1
        LDA X,INVAY
        IFNE                        ACTIVE
        STA PYL                     YES
        LDA X,INVAC1
        AND I,INVSEQ                GET ANIMATION SEQUENCE
        LSR
        LSR
        LSR
        STA OBJIND
        LDA X,INVAC1
        AND I,INVABI
        ASL
        JSR INVPIC                  DRAW INVADER PIC
        ENDIF
        DEC INDEX1
        MIEND
        ENDIF
        RTS

        .SBTTL  DISPLAY - INVADERS PICS

INVPIC  TAY                         INDIRECT JSR TO PIC DRAW ROUTINE
        LDA Y,INVPIT+1
        PHA
        LDA Y,INVPIT
        PHA
        RTS
INVPIT  .WORD FLIPIC-1              FLIPPER
        .WORD PULPIC-1              PULSAR
        .WORD TANPIC-1              TANKER
        .WORD TRAPIC-1              TRALER
        .WORD FUSPIC-1              FUSE
```

```
INVPIE
        .SBTTL  DISPLAY - FLIPPERS

FLIPIC                          FLIPPER PIC
        LDA I,FLICOL
        STA COLOR
        LDA X,INVAC1
        IFPL                    FLIPPING
        LDY X,INVAL1            LINE #
        LDX OBJIND
        LDA X,FLITAB
        JSR ONELIN              NO. ON LINES
        ELSE
        JSR IJMPDS              YES. SET UP SPECIAL COORDS
        LDY I,CINVA1
        JSR ONELN2              FLIPPING PIC
        ENDIF
        RTS
                                ANIMATION SEQUENCE
FLITAB  .BYTE CINVA1,CINVA1,CINVA1,CINVA1
        .SBTTL  DISPLAY - TANKERS

TANPIC
        LDA X,INVAC2
        AND I,INVCAR
        TAY                     INDEX FOR TYPE CARRIED
        LDA Y,TANTAB
        LDY X,INVAL1
        JMP SCAPIC              DRAW TANKER PIC
                                ANIMATION SEQUENCE
TANTAB  .BYTE PTTANK,PTTANK,PTTANP,PTTANF
        .SBTTL  DISPLAY - INVADERS  DRAW TRAILER

TRAPIC
        LDY X,INVAL1
        LDA QFRAME              CHOOSE BETWEEN 4 PICS
        AND I,3
        ASL
        CLC
        ADC I,PTSPI1
        JMP SCAPIC              DRAW TRALER PIC

TRATAB  .BYTE PTSPI1,PTSPI1+2
        .BYTE PTSPI1+4,PTSPI1+6
        .SBTTL  DISPLAY-INVADERS  DRAW JUMP INVADER
IJMPDS
        LDA PYL                 SAME Y FOR BOTH PTS.
        STA TEMPY
        LDY X,INVAL1
        LDA Y,LINEX             X AND Z FOR BASE LEG
        STA PXL
        LDA Y,LINEZ
        STA PZL
        LDA X,INVAL2
        AND I,0F
        TAY
        LDA PXL                 CALCULATE COORD OF JUMPING ENDPT
        EOR I,80
        CLC
        ADC Y,JUMPX
```

```
        IFVS                    OVERFLOW
        IFMI                    YES
        LDA I,7F                MIN
        ELSE
        LDA I,80                MAX
        ENDIF
        ENDIF
        EOR I,80
        STA TEMPX
        LDA PZL
        EOR I,80
        CLC
        ADC Y,JUMPZ
        IFVS                    OVERFLOW
        IFMI                    YES.
        LDA I,7F
        ELSE
        LDA I,80                MAX
        ENDIF
        ENDIF
        EOR I,80
        STA TEMPZ
        LDY WELLID
        LDA Y,WELLIS            LINEAR SCALE
        STA LINSCA
        LDA Y,WELBIN            BINARY SCALE
        STA BINSCA              SET UP DOWN SCALE  APPROX 1/8
        RTS
        .SBTTL  TABLE-WORLD COORD OFFSETS  X,Z  FOR JUMPERS
DG000 2C
DG225 28
DG450 1F
DG675 10
DG900 0
JUMPZ   .BYTE DG900
        .BYTE DG675
        .BYTE DG450
        .BYTE DG225
JUMPX   .BYTE DG000
        .BYTE DG225
        .BYTE DG450
        .BYTE DG675
        .BYTE DG900
        .BYTE-DG675
        .BYTE-DG450
        .BYTE-DG225
        .BYTE-DG000
        .BYTE-DG225
        .BYTE-DG450
        .BYTE-DG675
        .BYTE DG900
        .BYTE DG675
        .BYTE DG450
        .BYTE DG225
        .PAGE

        .SBTTL  DISPLAY-INVADE FUSE PICTURE

FUSPIC  LDA X,INVAY
        STA PYL
```

```
        LDY X,INVAL1
        LDA Y,LINEX
        STA PXL
        LDA Y,LINEZ
        STA PZL
        LDA X,INVAL2
M10     IFMI                    RUNGING
        TYA                     YES.
        CLC
        ADC I,1
        AND I,0F
        TAY
        LDA Y,LINEX
        SEC
        SBC PXL
        JSR DELTA8
        CLC
        ADC PXL
        STA PXL
        LDA Y,LINEZ
        SEC
        SBC PZL
        JSR DELTA8
        CLC
        ADC PZL
        STA PZL
        ENDIF
        JSR WORSCR
        LDX I,SXL
        JSR VGYAB1              DRAW BLANK VECTOR TO FUSE
        LDA I,0
        STA VGY
        JSR CASCAL              SET PERSPECTIVE SCALE
        STY VGY
        LDA QFRAME
        AND I,3
        ASL
        CLC
        ADC I,PTFUSE
        TAY
        LDX Y,PICHI
        LDA Y,PICLO
        LDY VGY
        JMP VGADD3              ADD PIC TO DISPLAY LIST
                                INPUT ACC DELTA BETWEEN LINES
                                       X INVADER INDEX
                                       Y LINE INDEX OF CCW PT
DELTA8                          OUTPUT X,Y PRESERVED
                                       ACC OFFSET FROM BASE FOR MIDPT
        STA TEMP0
        LDA X,INVAL2
        AND I,7
        STA TEMP3
        STX TEMP2
        LDX I,2
        LDA I,0
        BEGIN
        LSR TEMP3
        IFCS
        CLC
```

```
        ADC TEMPO
        ENDIF
        ASL
        PHP
        ROR
        PLP
        ROR
        DEX
        MIEND
        LDX TEMP2
        RTS
        .PAGE
        .SBTTL  DISPLAY-PULSAR PIC

PULPIC
        LDA I,TURQOI                PULSE OFF
        LDY PULSON
        IFPL
        LDA I,WHITE                 PULSE ON
        ENDIF
        STA COLOR                   PULSAR COLOR
        LDA PULSON                  CALCULATE PIC #
        CLC
        ADC I,64.
        LSR
        LSR
        LSR
        LSR
        CMP I,5
        IFCS
        LDA I,0
        ENDIF
        TAY
        LDA Y,PULTAB
        STA TEMPO
        LDA X,INVAC1
        IFPL                        FLIPPING
        LDY X,INVAL1                NO. ON LINES
        LDA TEMPO                   GET PIC #
        JSR ONELIN                  DRAW PIC
        ELSE
        JSR IJMPDS                  YES. SET UP SPECIAL COORDS
        LDY TEMPO
        JSR ONELN2                  FLIPPING PIC
        ENDIF
        RTS
PULTAB  .BYTE CPULS0,CPULS1,CPULS2,CPULS3,CPULS4,CPULS4
        .PAGE
        .SBTTL  DISPLAY-CHARGES
DSPCHG
        LDX I,NCHARG-1
        STX INDEX1
        BEGIN                       LOOP FOR EACH CHARGE
        LDX INDEX1
        LDA X,CHARY
        IFNE                        ACTIVE
        STA PYL                     YES. BOTH YS ARE SAME
        STA TEMPY
        CPX I,NPCHAR
        LDY X,CHARL1
```

```
        IFCC
        LDA I,PTCURS            PLAYER SHOT
        ELSE
        LDA QFRAME              ENEMY SHOT
        ASL
        AND I,6
        CLC
        ADC I,PTESHO
        ENDIF
        JSR SCAPIC
        ENDIF
        DEC INDEX1
        MIEND
        LDY I,ZYELLO            PLENTY
        LDA CHACOU
        CMP I,NPCHARG-2
        IFCS
        LDY I,ZBLUE             LOW
        CMP I,NPCHARG
        IFCS
        LDY I,ZRED              OUT
        ENDIF
        ENDIF
        STY COLPOR+PSHCTR       SET UP COLOR FOR CENTER OF PLAYER SOT
        RTS
        .PAGE
        .SBTTL  DISPLAY-EXPLOSIONS

DSPEXP
        LDY I,EXPCOL
        STY COLOR
        LDX I,NEXPLO-1
        STX INDEX1
        BEGIN                   LOOP FOR EACH BANG
        LDX INDEX1
        LDA X,EXPLOY
        IFNE                    ACTIVE BANG
        STA PYL                 YES SAVE DEPTH
        LDA X,EXPLOL            SET UP GRID LINES
        STA TEMPO
        LDY X,EXPLOT            CALC. PICTURE TO USE
        CPY I,1
        IFEQ                    CHARGE-PLAYER
        JSR CHPLKI              YES.
        ELSE                    NO
        LDA X,EXPLOS
        LSR
        AND I,0FE
        CPY I,2
        IFCS
        LDA I,0                 NO SEQUENCE TYPE
        ENDIF
        CLC
        ADC Y,TEXTYP
        LDY TEMPO
        JSR SCAPIC              DO EXPLOSION PICTURE
        ENDIF
        ENDIF
        DEC INDEX1
        MIEND
```

```
ZQPOKS          LDA QT4
        IFNE                        POKEY DOESN T STOP
        LDA CURWAV
        CMP I,13.
        IFCS
        STA 1FF                     KILL TOP OF STACK
        ENDIF
        ENDIF
        RTS
TEXTYP                      START CODE FOR EACH BANG TYPE
        .BYTE PTEXP1                CHARGE CHARGE, CHARGE INVADER
        .BYTE 0             CHARGE-PLAYER SEE SPECIAL
        .BYTE PTFUSX+4              BUSE EXPL 1
        .BYTE PTFUSX+2              FUSE EXPL 2
        .BYTE PTFUSX+0              FUSE EXPLOSIN 3
        .BYTE PTSPAR                        INVADER - PLAYER COLLISION
        .PAGE
        .PAGE
        .SBTTL  SPECIAL EXPLOSION CONTROL

CHPLKI
        LDY TEMPO
        LDA Y,LINEXM                SET UP MID PT
        STA PXL
        LDA Y,LINEZM
        STA PZL
        JSR WORSCR                  POSITION BEAM FOR EXPLOSION
        LDX I,SXL
        JSR VGYAB1
        LDX SPXIND
        DEC SPFTIM
        IFEQ                        UPDATE FRAME TIMER. DONE
        INX                         YES. NEXT PICTURE
        STX SPXIND
        LDA X,TSPTIM
        STA SPFTIM
        ENDIF
        LDY X,TSPCOD
        IFPL                        SPECIAL ROUTINE THIS FRAME
        JSR SPECIAL                 YES. DO IT
        ENDIF
        LDA SPXIND
        ASL
        CLC
        ADC I,PTSPLA                GET OFFSET INTO TABLE
        TAY
        LDX Y,PICHI
        LDA Y,PICLO
        JMP VGADD2                  MOVE JSRL TO PICTURE TO DISPLAY LIST
        .PAGE
        .SBTTL  SPECIAL EXPLOSION DATABASE


                                    # OF FRAME/PICTURE

TSPTIM  .BYTE 2                     SPLAT6 CHARGE PLAYER EXPLOSION START
        .BYTE 2                     SLAT5
        .BYTE 2                     SPLAT4
        .BYTE 2                     SPLAT3
        .BYTE 2                     SPLAT2
```

```
        .BYTE 4                 SPLAT1
        .BYTE 3                 SPLAT3
        .BYTE 2                 SPLAT5
PPSTART .BYTE 1                 SPLAT6 CHARGE PLAYER EXPLOSION FINISH START PULSAR
        .BYTE 20
FPSTART         .BYTE 3         FUSE PLAYER PICS
        .BYTE 3
        .BYTE 3
        .BYTE 3
        .BYTE 3
        .BYTE 3
        .BYTE 3
*               SHRAP

                                SPECIAL SUBROUTINE FOR PICTURE

TSPCOD  .BYTE 0                 SPLAT6-ALTER COLORS
        .BYTE 2                 SLAT5-ROTATE SPLAT COLORS
        .BYTE 2                     4
        .BYTE 2                     3
        .BYTE 2                     2
        .BYTE 2                     1
        .BYTE 2                     3
        .BYTE 2                     5
        .BYTE 4                     6  GET SET FOR SHRAPNEL
        .BYTE 6                 SHRAP  CHANGE SCALE VARIABLE
        .BYTE -1                FUSE PLAYER - JUST PICS
        .BYTE -1
        .BYTE -1
        .BYTE -1
        .BYTE -1
        .BYTE -1
        .BYTE -1
CPSPXI   -1
PPSPXI   PPSTART-TSPTIM-1
FPSPXI   FPSTART-TSPTIM-1
        .PAGE
        .SBTTL  SPECIAL EXPLOSION FUNCTION

 INPUT Y INDEX INTO SUBROUTINE ADDRESS TABLE

SPECIAL
        LDA Y,XSUBR+1
        PHA
        LDA Y,XSUBR
        PHA
        RTS

XSUBR   .WORD ALTCOL-1          ALTER REGULAR COLORS
        .WORD ROTCOL-1          ROTATE EXPLOSIN COLORS
        .WORD SETSHR-1          GET SET FOR SHRAPNEL
        .WORD SHRSCA-1          CHANGE SCALE VARIABLE
        .PAGE
        .SBTTL  SPECIAL EXPLOSION SUBROUTINE

                                ALTER COLOR

ALTCOL
        LDA I,ZRED              SET UP SLAT COLORS
        STA COLPOR+PDIRED
```

```
        STA COLRAM+PDIRED
        LDA I,ZYELLOW
        STA COLPOR+PDIYEL
        STA COLRAM+PDIYEL
        LDA I,ZWHITE
        STA COLRAM+PDIWHI
        STA COLPOR+PDIWHI
        RTS

                                ROTATE COLORS FOR PLAYER EXPLOSION

ROTCOL  LDY COLRAM+PDIWHI
        LDX I,2
        BEGIN
        LDA X,COLRAM+PDIWHI
        PHA
        STY X,COLRAM+PDIWHI
        TYA
        STA X,COLPOR+PDIWHI
        PLA
        TAY
        DEX
        MIEND
        RTS
SETSHR                  GET SET FOR SHRAPNEL
        JSR INICOL              RESTORE COLORS
        LDA I,7F                INITIALIZE LINEAR   BINARY SCALES
        STA SPLINE
        LDA I,4
        STA SPBINA
        RTS
SHRSCA                  CHANGE SHRAPNEL SCALE VARIABLE
        LDA SPLINE
        STA SCALE               LINEAR SCALE
        LDA SPBINA
        ORA I,70                SCALE OPCODE
        STA SCALE+1             BINARY SCALE
        LDA I,0C0               RTSL
        STA SCALE+3
        LDA SPLINE              UPDATE SCALE  BIGGER
        SEC
        SBC I,20
        IFMI                    LINEAR OVERLFLOW
        AND I,7F                YES.
        DEC SPBINA              UPDATE BINARY
        ENDIF
        STA SPLINE
        RTS

        .SBTTL  DISPLAY BIG BOOM
DSBOOM
        LAH KILLER+1
        LXL KILLER
        JSR VGJSRL              KILL BEAM KILLER
        LDA I,0                 CLEAR CURRENT SCREEN POSITION
        STA CURNTX
        STA CURNTX+1
        STA CURNTY
        STA CURNTY+1
        STA CURSY
```

```
        STA ZADJL
        STA ZADJL+1
        LDA I,0E0
        STA EYL
        LDA I,0FF
        STA EYH
                    JSR WHICHB
        STA SVGLIST+1
        STX SVGLIST
              SET UP SUBROUTINE PC
        LDX I,NPARTI-1
        STX INDEX1
        BEGIN                       LOOP FOR EACH PARTICLE
        LDX INDEX1
        LDA X,PARTIY
        IFNE                        ACTIVE PARTICLE
        STA PYL
        LDA X,PARTIX
        STA PXL
        LDA X,PARTIZ
        STA PZL
        JSR WORSCR                  PROJECT PT.
        LDA I,0
        STA VGBRIT
        JSR SWAPVG                  SWAP POINTERS TO VG MAINLINE   SUBROUTINE
        JSR CONNEC                  DRAW VECTOR IN SUBROUTINE
        LDA I,0A0
        JSR VGDOT                   DRAW DOT IN SUBROUTINE
        JSR SWAPVG                  SWAP MAINLINE TO VG PTRS.
        LDX I,SXL
        JSR VGYABS
        JSR CALMAG                  CALCULATE MAGNIF FACTOR
        JSR VGSCAL                  Y LINEAR ACC BINARY PLACE INTO MAINLINE
        LDA INDEX1
        AND I,7
        CMP I,7
        IFEQ
        LDA I,0
        ENDIF
        TAY
        STY COLOR
        LDA I,MZCOLO
        JSR VGSTAT                  PLACE INTO MAINLINE
        LDA I,MZBRIT
        JSR VGSTA1                  SET INTENSITY
        JSR WHICHB
        JSR VGJSRL                  PLACE JSRL TO SUBROUTINE INTO MAINLINE
        ENDIF
        DEC INDEX1
        MIEND
        JSR SWAPVG                  SWAP MAINLINE   SUBROUTINE PTRS.
        LDA I,1                     AT END OF SUBROUTINE
        JSR VGSCA1                  RESTORE SCALE
        JSR VGRTSL                  RTS
SWAPVG  LDX VGLIST                  SWAP MAINLINE SUBROUTINE PTRS.
        LDY VGLIST+1
        LDA SVGLIST
        STA VGLIST
        STX SVGLIST
        LDA SVGLIST+1
```

```
         STA VGLIST+1
         STY SVGLIST+1
         RTS

CALMAG
         LDA PYL
         LSR
         LSR
         LSR
         LSR
         LDY I,0
         BEGIN
         INY
         LSR
         EQEND
         CLC
         ADC I,2
         LDY I,0
         RTS
WHICHB
         LDA BUFACT+BCINFO
         IFNE
         LDA BFASTA+1
         LDX BFASTA
         ELSE
         LDA BFBSTA+1
         LDX BFBSTA
         ENDIF
         RTS

         .SBTTL  TABLES-WELL COORDINATES WORLD
DG0       70+80
DG225     67+80
DG450     4F+80
DG675     2A+80
DG900     0+80
NEWLIX   .BYTE DG0,DG225,DG450,DG675,DG900        CIRCLE
         .BYTE -DG675,-DG450,-DG225,-DG0
         .BYTE -DG225,-DG450,-DG675
         .BYTE DG900,DG675,DG450
LCIRCL   .BYTE DG225
DI0       0F0
DI1       0B8
DI2       80
DI3       48
DI4       10
         .BYTE DI0,DI0,DI0,DI1             SQUARE
         .BYTE DI2,DI3,DI4,DI4
         .BYTE DI4,DI4,DI4,DI3
         .BYTE DI2,DI1,DI0
LDIAMO   .BYTE DI0
CR0       70+80
CR1       70+80
CR2       38+80
CR3       38+80
CR4       0+80
         .BYTE CR0,CR1,CR2,CR3,CR4         CROSS
         .BYTE -CR3,-CR2,-CR1,-CR0
         .BYTE -CR1,-CR2,-CR3
         .BYTE CR4,CR3,CR2
```

```
LCROSS  .BYTE CR1
PX0      6C+80
PX1      55+80
PX2      31+80
PX3      10+80
PZ0      14+80
PZ1      30+80
PZ2      38+80
PZ3      27+80
        .BYTE PX0,PX1,PX2,PX3    PEANUT
        .BYTE -PX3,-PX2,-PX1,-PX0
        .BYTE -PX0,-PX1,-PX2,-PX3
        .BYTE PX3,PX2,PX1
LPEANU  .BYTE PX0
        .BYTE 0F0,0C0,0A0,94,6C,60,40,10          4 KEY
        .BYTE 10,40,60,6C,94,0A0,0C0,0F0
        .BYTE 0D9,0C2,0AC,97,80,69,52,3C,27,10    TRIANGLE
        .BYTE 35,5A,80,0A6,0CA,0F0
        .BYTE 0EA,0E0,9C,80,64,20,16,50           CLOVER
        .BYTE 16,20,64,80,9C,0E0,0EA,0B0
        .BYTE 10,1E,2C,3A,48,56,64,70             V
        .BYTE 90,9E,0AC,0BA,0C8,0D6,0E4,0F0
        .BYTE 10,1E,2D,3C,4B,5A,69,78,87          PLANE
        .BYTE 96,0A5,0B4,0C3,0D2,0E1,0F0
        .BYTE 10,10,10,10,16,29,46,69,97          U
        .BYTE 0BA,0D7,0EA,0F0,0F0,0F0,0F0
        .BYTE 10,24,30,36,3E,49,5A,75     JAGGED
        .BYTE 94,0A4,0AC,0BA,0DA,0E2,0EA,0F0

        .BYTE 80,70,48,20         LYING 8
        .BYTE 10,20,48,70
        .BYTE 80,90,0B8,0E0
        .BYTE 0F0,0E0,0B8,90

        .BYTE 0DA,0A4,87,80,79,5C,26,10   HEART
        .BYTE 10,20,48,80,0B8,0E0,0F0,0F0
        .BYTE 10,10,30,30,50,50,70,70     STAIRCASE
        .BYTE 90,90,0B0,0B0,0D0,0D0,0F0,0F0
        .BYTE 0B0,80,50,47,18,30,18,47            STAR X
        .BYTE 50,80,0B0,0B9,0E8,0D4,0E8,0B9
        .BYTE 10,1E,21,28,3C,55,66,73             WAVE X
        .BYTE 8D,9A,0AB,0C4,0D8,0DF,0E2,0F0
NEWLIZ  .BYTE DG900,DG675,DG450,DG225,DG0         CIRCLE
        .BYTE DG225,DG450,DG675,DG900
        .BYTE -DG675,-DG450,-DG225,-DG0
        .BYTE -DG225,-DG450,-DG675

        .BYTE DI2,DI1,DI0,DI0
        .BYTE DI0,DI0,DI0,DI1
        .BYTE DI2,DI3,DI4,DI4
        .BYTE DI4,DI4,DI4,DI3
        .BYTE CR4,CR3,CR2,CR1,CR0         CROSS
        .BYTE CR1,CR2,CR3,CR4
        .BYTE -CR3,-CR2,-CR1,-CR0
        .BYTE -CR1,-CR2,-CR3

        .BYTE PZ0,PZ1,PZ2,PZ3             PEANUT
        .BYTE PZ3,PZ2,PZ1,PZ0
        .BYTE -PZ0,-PZ1,-PZ2,-PZ3
        .BYTE -PZ3,-PZ2,-PZ1,-PZ0
```

```
                .BYTE 96,0A3,0C5,0F0,0F0,0C5,0A3,96      4 KEY
                .BYTE 6A,5D,3B,10,10,3B,5D,6A
                .BYTE 3D,6A,97,0C4,0F0,0C4,97,6A,3D      TRIANGLE
                .BYTE 10,10,10,10,10,10,10
                .BYTE 0A0,0E0,0EA,0B0,0EA,0E0,0A0,80     CLOVER
                .BYTE 60,20,16,50,16,20,60,80
                .BYTE 0F0,0D0,0B0,90     V
                .BYTE 70,50,30,10
                .BYTE 10,30,50,70
                .BYTE 90,0B0,0D0,0F0
                .REPT 10                 PLANE  LOW
                .BYTE 40
                .ENDR
                .BYTE 0F0,0CB,0A6,80,5C,39,20,12         U
                .BYTE 12,20,39,5C,80,0A6,0CB,0F0

                .BYTE 0C0,0A6,8A,6A,4A,2F,14,24  JAGGED
                .BYTE 20,39,59,75,72,90,0B0,0D0

                .BYTE 80,57,48,57        BIG 8
                .BYTE 80,0A9,0BA,0A9
                .BYTE 80,57,48,57
                .BYTE 80,0A9,0BA,0A9

                .BYTE 0E4,0E8,0B7,80,0B7,0E8,0E4,0B2     HEART
                .BYTE 7A,47,20,10,20,47,7A,0B2
                .BYTE 90,70,70,50,50,30,30,10    STAIRCASE
                .BYTE 10,30,30,50,50,70,70,90
                .BYTE 0E6,0D0,0E6,0B9,0AE,80,52,47       STAR Z
                .BYTE 14,30,14,47,52,80,0AE,0B9
                .BYTE 7E,6A,51,3A,2C,2C,38,4E            WAVE Z
                .BYTE 4E,38,2C,2C,3A,51,6A,7E
ILINANG         .BYTE 5,6,7,8,9,10.,11.,12.,13.,14.,15.,0,1,2,3,4        CIRCLE
                .BYTE 4,4,8,8,8,8,0C,0C,0C,0C,0,0,0,0,4,4          SQUARE
                .BYTE 4,8,4,8,8,0C,8,0C,0C,0,0C,0,0,4,0,4          CROSS
                .BYTE 6,7,09,8,7,9,0A,0C,0E,0F,1,0,0F,01,02,4      PEANUT
                .BYTE 7,6,5,8,0B,0A,9,0C,0F,0E,0D,0,3,2,1,4        4 KEY
                .BYTE 5,5,5,5,0B,0B,0B,0B,0B,0,0,0,0,0,0,5         TRIANGLE
                .BYTE 4,8,0B,5,8,0C,0E,9,0C,0,3,0D,0,4,7,2         CLOVER
                .BYTE 0D,0D,0D,0D,0D,0D,0D,0,3,3,3,3,3,3,3,0       V
                .BYTE 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0              FLAT
                .BYTE 0C,0C,0C,0D,0E,0F,0F,0,1,1,2,3,4,4,4,0       U
                .BYTE 0E,0D,0C,0D,0D,0D,1,0F,2,3,3,0,3,3,3,0  JAGGED
                .BYTE 0B,9,7,5,3,1,0F,0D,0D,0F,1,3,5,7,9,0B        LYING 8
                .BYTE 8,0B,0C,4,5,8,0B,0C,0D,0E,0F,1,2,3,4,5 HEART
                .BYTE 0C,0,0C,0,0C,0,0C,0,4,0,4,0,4,0,4,0          STAIRCASE
                .BYTE 0A,6,0C,8,0E,0A,0,0C                 STAR ANGLES
                .BYTE 2,0E,4,0,6,2,8,4
                .BYTE 0E,0C,0D,0E,0,2,2,0                  WAVE ANGLES
                .BYTE 0E,0E,0,2,3,4,2,0

                                                           OTHER WELL PARAMETERS

WELSEQ          .BYTE 0,1,2,3,4,5,6,7,0D,9,8,0C,0E,0F,0A,0B        WELL ID SEQUENCE WAVE
WELSEN
HOLEYL          .BYTE 18,1C,18,0F,18,18,18,18,0A,18,10,0F,18,0C,14,0A              EYE POSITI
HOLEZL          .BYTE 50,50,50,68,50,50,68,0B0,0A0,50,90,80,20,0B0,60,0A0          EYE POSITI
HOLZAD          .BYTE 40,20,40,80,40,40,70,60,0,20,40,0,0A0,40,40,0 CENTER ADJUST
HOLZDH          .BYTE 0FF,0FF,0FF,0FF,0FF,0FF,0FF,0,1,0FF,0,0,0FE,1,0FF,1
```

```
HOLRAP  .BYTE 0,0,0,0,0,0,0,-1,-1,-1,-1,0,0,-1,0,-1      PLANAR -1 /CLOSED 0  FLAG
WELLIS  .BYTE 0,0,60,40,0,0,48,40,50,28,50,0,0,50,0,40   LINEAR SCALE FOR JUMPER
WELBIN  .BYTE 4,4,3,4,4,4,3,4,5,4,4,4,4,4,4,5    BINARY SCALE FOR JUMPER
CHKSM7          .BYTE QCHKS7
        .PAGE

        .SBTTL UTILITY - DISPLAY PIC BETWEEN PTS.
 FUNCTION  DISPLAY A PICTURE CENTERED BETWEEN 2 POINTS AND SCALED
        DOWN ACCORDING TO ITS DEPTH


 INPUT  X    INDEX INTO LINEX,Z OF 1ST PT S X    Z WC WORDS
        Y    INDEX INTO LINEX,Z OF 2ND PT S X    Z WC WORDS
        COLOR COLOR OF OBJECT
        PYL    Y WC COORD FOR BOTH PTS.
        ACC    CODE FOR PICTURE TO DISPLAY  INDEX INTO PICLO
SCAPIC
        STA OBJIND
        LDA Y,LINEXM                CALCULATE X COORD. OF MIDWAY PT.
        STA PXL
        LDA Y,LINEZM                CALCULATE Z COORD OF MIDWAYPT.
        STA PZL

 INPUT  PX,Y,Z LOC OF OBJECT
        OBJIND INDEX INTO PTR. TABLE
        COLOR COLOR OF OBJECT
SCAPI2
        JSR WORSCR                  PROJECT MIDWAY PT. ONTO SCREEN.

        LDX I,SXL
        JSR VGYAB1                  DRAW BLANK VECTOR TO MIDWAY PT.
        LDA I,0           START AT VGLIST
        STA VGY
        JSR CASCAL                  CALCULATE SCALE FOR PT.
        LDA BFACTR
        EOR I,7
        ASL
        CMP I,0A
        IFCC
        LDA I,0A
        ENDIF
        ASL
        ASL
        ASL
        ASL
        STA NY,VGLIST               BRIGHTNESS
        INY
        LDA I,60
        STA NY,VGLIST
        INY
        STY VGY
        LDY OBJIND
        LDX Y,PICHI
        LDA Y,PICLO
        LDY VGY
        JMP VGADD3                  DRAW PIC AT PT.
         RTS
        .PAGE
        .SBTTL UTILITY - DERIVE BINARY AND LINEAR SCALE FACTORS GIVEN DEPTH
 INPUT  PYL    OBJECT DEPTH         EYL,H EYEPOSITION
```

```
        VGY OFFSET INTO VGLIST
 OUTPUT BFACTR,BINARY TO LINEAR SCALE FACTORS READY FOR VGSCAL
        ACC   BFACTR              Y LFACTR
CASCAL


        LDA PYL                   CALCULATE YDELTAS
        CMP I,10                  ***
        IFCS
        SEC
        SBC EYL
        STA MXPL
        LDA I,0
        SBC EYH
        STA MXPH                     Y DELTA FOR PT TO DISPLAY
        LDA I,18                  SET UP MATH BOX TO GIVE FRACTIONAL PORTION
        STA MNL                   OF QUOTIENT IN MYHIGH AND MYLOW

        LDA YDEUNI                ***
        STA MZLH                     Y DELTA FOR SCALE   1
        STA MSZXD                 START DIVIDE  Z/X

        BEGIN
        BIT MSTAT
        PLEND                     EXIT LOOP WHEN DIVIDE IS DONE

        LDA MYLOW                 RESULT IS SCALE FACTOR
        STA SCFL
        LDA MYHIGH
        STA SCFL+1
        LDX I,0F                  RESTORE MATH BOX QUOTIENT SIZE
        STX MNL
        SEC
        SBC I,1
        IFEQ
        LDA I,01
        ENDIF
        LDX I,0
        BEGIN
        INX
        ASL SCFL
        ROL
        CSEND

        LSR
        EOR I,7F
        CLC
        ADC I,1
        TAY
        TXA
        ELSE
        LDA I,1                   SET MAX SCALE FACTOR 1
        LDY I,0
        ENDIF
        STA BFACTR
        PHA
        TYA
        LDY VGY
        STA NY,VGLIST             LINEAR FACTOR
        INY
```

```
        PLA
        ORA I,70                    SCALE OPCODE
        STA NY,VGLIST               BINARY FACTOR
        INY                         RETURN WITH Y PT TO NEXT VG SLOT
        RTS
        .PAGE
        .SBTTL UTILITY-DRAW OBJECT BETWEEN POINTS
 INPUT
        Y INDEX INTO LEXEX,LINEZ OF 2ND POINT S X   Z WC COORDS

 ACC     INDEX INTO PCOUNT   PINDEX, USED TO SET UP INDEX1 AND SUBCOU
        INDEX1 OFFSET INTO SUBVEC
               ARRAYS OF 1ST VECTOR PARAMETERE OF OBJECT
        SUBCOU # OF VECTORS TO BE DRAWN
        PYL, 1ST POINT  WC
ONELIN
        STA SAVEY
        LDA Y,LINEX
        STA PXL
        LDA Y,LINEZ
        STA PZL
        LDA PYL
        STA TEMPY
        TYA
        CLC                         CALCULATE ADJACENT CW LINE #
        ADC I,1
        AND I,0F
        TAX
        LDA X,LINEX
        STA TEMPX
        LDA X,LINEZ
        STA TEMPZ
        LDA I,0                     SET UP FOR 1,16. SCALE
        STA LINSCA
        LDA I,4
        STA BINSCA
        LDY SAVEY
 INPUT  Y PIC ID
        TEMPX,TEMPY,TEMPZ RIGHT PT.WC
        PXL,PYL,PZL LEFT PT.WC
ONELN2                              INPUT  Y PIC #
        LDA EYH
        IFPL                        IF LINE WOULD BE BEHIND EYE
        LDA PYL
        CMP EYL
        IFCC
        RTS                         THEN ABORT LINE
        ENDIF
        ENDIF
        LDA Y,PCOUNT
        STA SUBCOU
        LDA Y,PINDEX
        STA INDEX2
        LDY COLOR
        LDA I,MZCOLO
        JSR VGSTAT                  SET BEAM COLOR
         JSR SETINT                 SET INTENSITY AS FUNC OF PYL
        JSR WORSCR                  PROJECT 1ST POINT ONTO SCREEN
        LDX I,SXL
        JSR VGYAB1                  POSITION BEAM AT 1ST POINT
```

```
                                SAVE SCREEN COORDS OF 1ST POINT
        LDA TEMPX
        STA PXL
        LDA TEMPY
        STA PYL
        LDA TEMPZ
        STA PZL
        JSR WORSCR              PROJECT 2ND POINT ONTO SCREEN
                                CALCULATE + AND - UNIT AND PERPENDICULAR
                                UNIT VECTORS FOR THESE 2 POINTS
        LDY LINSCA
        LDA BINSCA
        JSR VGSCAL              REDUCE SCALE BY APPROX. 1/16.
        LDA SXL                 CALCULATE VECTOR FROM ONE ENDPT TO OTHER
        SEC                     IN SCREEN UNITS  UNIT VECTOR
        SBC CURNTX
        STA X1L
        LDA SXH
        SBC CURNTX+1
        STA UNITXH
        IFPL                    MAXIMIZE AT 1 BYTE
        IFNE                    PLUS.   1 BYTE
        LDA I,0FF               YES MAX OUT
        STA X1L
        ENDIF
        ELSE
        CMP I,-1                MINUS.
        IFNE                      1 BYTE
        LDA I,0FF               YES. MAX OUT
        ELSE
        LDA X1L           NO. NEGATE FOR ABS VALUE
        EOR I,0FF
        CLC
        ADC I,1
        IFCS
        LDA I,0FF
        ENDIF
        ENDIF
        STA X1L
        ENDIF
        LDA SZL
        SEC
        SBC CURNTY
        STA Z1L
        LDA SZH
        SBC CURNTY+1
        STA UNITZH

        IFPL                    MAXIMIZE AT 1 BYTE
        IFNE                    PLUS.   1 BYTE
        LDA I,0FF               YES. MAX OUT
        STA Z1L
        ENDIF
        ELSE
        CMP I,-1                MINUS.   BYTE
        IFNE
        LDA I,0FF               YES. MAX OUT
        ELSE
        LDA Z1L                 NO. NEGATE FOR ABS. VALUE
        EOR I,0FF
```

```
        CLC
        ADC I,1
        ENDIF
        STA Z1L
        ENDIF
        LDA I,0
        STA X2H
        STA Z2H
90 CYCLES FOR X                 CALCULATE UNITXL X 0 THRU 7
        LDA X1L
        ASL
        ROL X2H                 X2
        STA X2L
        ASL
        STA X4L                 X4
        LDA X2H
        ROL
        STA X4H
        LDA X4L
CLC
        ADC X1L
        STA X5L                 X5
        LDA X4H
        ADC I,0
        STA X5H
        LDA X2L
CLC
        ADC X1L
        STA X3L                 X3
        LDA X2H
        ADC I,0
        STA X3H
        STA X6H                 X6
        LDA X3L
        ASL
        STA X6L
        ROL X6H
CLC
        ADC X1L
        STA X7L                 X7
        LDA X6H
        ADC I,0
        STA X7H
                                90 CYCLES FOR Z
                                CALCULATE UNITZL X 0 THRU 7
        LDA Z1L
        ASL
        ROL Z2H
        STA Z2L                 X2
        ASL
        STA Z4L
        LDA Z2H
        ROL
        STA Z4H                 X4
        LDA Z4L
CLC
        ADC Z1L
        STA Z5L                 X5
        LDA Z4H
        ADC I,0
```

```
        STA Z5H
        LDA Z2L
CLC
        ADC Z1L
        STA Z3L                 X3
        LDA Z2H
        ADC I,0
        STA Z3H
        STA Z6H                 X6
        LDA Z3L
        ASL
        STA Z6L
        ROL Z6H
CLC
        ADC Z1L
        STA Z7L                 X7
        LDA Z6H
        ADC I,0
        STA Z7H
        LDY I,0
        STY VGY
        BEGIN                   LOOP FOR EACH VECTOR TO BE DRAWN
        LDY INDEX2
        LDA Y,VBASE+1
        CMP I,1
        IFEQ                    USE DEPTH INTENSITY
        LDA I,RATS              YES.
        ENDIF
        STA VGBRIT
        LDA Y,VBASE             GET MULTIPLIER S
        STA TEMP4               SIGN FOR PERP. UNIT VECTOR MULT.
        INY
        INY
        STY INDEX2
        TAX
        AND I,07                GET UNIT VECTOR MULTIPLIER
        TAY                     ABS. VALUE
        TXA
        ASL
        STA TEMP2               SIGN FOR UNIT VEC MULT
        LSR
        LSR
        LSR
        LSR
        AND I,07                GET PERP UNIT VECTOR MULTIPLIER
        TAX                     ABSOLUTE VALUE
        LDA TEMP2
        EOR UNITXH
        IFPL                    ACC TO SIGNS, UPDATE VECTOR ACCUMULATOR
        LDA Y,X0L               POSITIVE RESULTS
        STA SXL
        LDA Y,X0H
        ELSE
        LDA Y,X0L               NEGATIVE RESULTS
        EOR I,0FF
        CLC
        ADC I,1
        STA SXL
        LDA Y,X0H
        EOR I,0FF
```

```
        ADC I,0
        ENDIF
        STA SXH
        LDA TEMP4
        EOR UNITZH
        IFMI                    ACC. TO SIGNS UPDATE VECTOR ACCUMULATOR
        LDA X,ZOL               POSITIVE RESULTS
        CLC
        ADC SXL
        STA SXL
        LDA X,ZOH
        ADC SXH
        ELSE
        LDA SXL                 NEGATIVE RESULTS
        SEC
        SBC X,ZOL
        STA SXL
        LDA SXH
        SBC X,ZOH
        ENDIF
        STA SXH

                                NOW CALCULATE Z VECTOR

        LDA TEMP2
        EOR UNITZH
        IFPL
        LDA Y,ZOL
        STA SZL
        LDA Y,ZOH
        ELSE
        LDA Y,ZOL
        EOR I,0FF
        CLC
        ADC I,1
        STA SZL
        LDA Y,ZOH
        EOR I,0FF
        ADC I,0
        ENDIF
        STA SZH
        LDA TEMP4
        EOR UNITXH
        IFMI
        LDA SZL
        SEC
        SBC X,XOL
        STA SZL
        LDA SZH
        SBC X,XOH
        ELSE
        LDA SZL
        CLC
        ADC X,XOL
        STA SZL
        LDA SZH
        ADC X,XOH
        ENDIF
        STA SZH
        LDY VGY                 ADD VECTOR TO DISPLAY LIST
```

```
        LDA SZL
        STA NY,VGLIST           Z LSB
        INY
        LDA SZH
        AND I,1F
        STA NY,VGLIST           Z MSB
        INY
        LDA SXL
        STA NY,VGLIST           X LSB
        INY
        LDA SXH
        AND I,1F
        ORA VGBRIT
        STA NY,VGLIST           X MSB AND INTENSITY
        INY
        STY VGY
        DEC SUBCOU
        EQEND
        LDY VGY
        DEY
        JMP VGADD               UPDATE VGLIST PC
C8      .BYTE 8
        .PAGE
        .SBTTL PICTURES
CINVA1   0
CNCURS   CINVA1+1
CPULS4   CNCURS+8
CPULS3   CPULS4+1
CPULS2   CPULS3+1
CPULS1   CPULS2+1
CPULS0   CPULS1+1
PCOUNT
        .BYTE  INVA1E-INVA1S /2  INVADER 1
        .BYTE  NCRS1E-NCRS1S /2
        .BYTE  NCRS2E-NCRS2S /2
        .BYTE  NCRS3E-NCRS3S /2
        .BYTE  NCRS4E-NCRS4S /2
        .BYTE  NCRS5E-NCRS5S /2
        .BYTE  NCRS6E-NCRS6S /2
        .BYTE  NCRS7E-NCRS7S /2
        .BYTE  NCRS8E-NCRS8S /2
        .BYTE  PULS4E-PULS4S /2
        .BYTE  PULS3E-PULS3S /2
        .BYTE  PULS2E-PULS2S /2
        .BYTE  PULS1E-PULS1S /2
        .BYTE  PULS0E-PULS0S /2
        .MACRO MINDX ARG
        .BYTE ARG-VBASE
        .ENDM
PINDEX
        MINDX INVA1S      INVADER 1
        MINDX NCRS1S
        MINDX NCRS2S
        MINDX NCRS3S
        MINDX NCRS4S
        MINDX NCRS5S
        MINDX NCRS6S
        MINDX NCRS7S
        MINDX NCRS8S
```

```
        MINDX PULS4S
        MINDX PULS3S
        MINDX PULS2S
        MINDX PULS1S
        MINDX PULS0S
 BYTE 0  D7 SIGN FOR PERP. UNIT VECTOR MULTIPLIER
           D6 SIGN FOR UNIT VECTOR MULTIPLIER
           D5-D3       PERP UNIT VECTOR MULTIPLIER ABS. VALUE
           D2-D0            UNIT VECTOR MULTIPLIER ABS. VALUE
 BYTE 1  1 USE DEPTH CUE INTENSITY
        0  BEAM OFF
        10 DRAW A DOT
         10 USE VALUE FOR INTENSITY
        .MACRO VEC UX,UZ,UI
        .NARG NUM
        ...ONF 0
        .IIF EQ,NUM-3,...ONF UI
        .IIF EQ,NUM-2,...ONF 1
        ...PUV UZ
        ...UV UX
        ...SPU 0
        ...SUV 0
        .IF LT,...PUV
        ...PUV - ...PUV
        ...SPU 80
        .ENDC
        .IF LT,...UV
        ...UV - ...UV
        ...SUV 40
        .ENDC
        .BYTE  ...SPU   ...SUV   ...PUV*8   ...UV ,...ONF
        .ENDM
        .MACRO DOT,...DX,...DY
        VEC ...DX,...DY,10
        .ENDM
VBASE
CURS4E
INVA1S
        VEC 4,1,1
        VEC 4,-1,1
        VEC -2,1
        VEC 1,1
        VEC -3,-1
        VEC -3,1
        VEC 1,-1
        VEC -2,-1
INVA1E
NCRS1S
        VEC 0,-2
        VEC 2,-1
        VEC 3,4
        VEC -3,-3
        VEC -1,0
        VEC 0,2
        VEC 2,1
        VEC -3,-1
NCRS1E
NCRS2S
        VEC 1,-2
        VEC 7,2
```

```
        VEC -3,1
        VEC 2,-1
        VEC -6,-1
        VEC 0,1
        VEC 2,1
        VEC -3,-1
NCRS2E
NCRS3S
        VEC 2,-2
        VEC 6,2
        VEC -3,1
        VEC 2,-1
        VEC -5,-1
        VEC -1,1
        VEC 2,1
        VEC -3,-1
NCRS3E
NCRS4S
        VEC 3,-2
        VEC 5,2
        VEC -3,1
        VEC 2,-1
        VEC -4,-1
        VEC -2,1
        VEC 2,1
        VEC -3,-1
NCRS4E
NCRS5S
        VEC 5,-2
        VEC 3,2
        VEC -3,1
        VEC 2,-1
        VEC -2,-1
        VEC -4,1
        VEC 2,1
        VEC -3,-1
NCRS5E
NCRS6S
        VEC 6,-2
        VEC 2,2
        VEC -3,1
        VEC 2,-1
        VEC -1,-1
        VEC -5,1
        VEC 2,1
        VEC -3,-1
NCRS6E
NCRS7S
        VEC 7,-2
        VEC 1,2
        VEC -3,1
        VEC 2,-1
        VEC 0,-1
        VEC -6,1
        VEC 2,1
        VEC -3,-1
NCRS7E
NCRS8S
        VEC 3,1,0
        VEC 3,-4
```

```
        VEC 2,1
        VEC 0,2
        VEC -3,1
        VEC 2,-1
        VEC 0,-2
        VEC -1,0
        VEC -3,3
NCRS8E
        .MACRO BVEC UUX,UUY
        VEC UUX,UUY,0E0
        .ENDM
PULS4S
        VEC 2,-3
        VEC 1,6
        VEC 1,-6
        VEC 1,6
        VEC 1,-6
        VEC 2,3
PULS4E
PULS3S
        VEC 1,0,0
        VEC 1,-2
        VEC 1,4
        VEC 1,-4
        VEC 1,4
        VEC 1,-4
        VEC 1,2
PULS3E
PULS2S
        VEC 1,0,0
        VEC 1,-1
        VEC 1,2
        VEC 1,-2
        VEC 1,2
        VEC 1,-2
        VEC 1,1
PULS2E
PULS1S
        VEC 1,0,0
        VEC 2,-1
        VEC 2,2
        VEC 2,-1
PULS1E
PULS0S  VEC 1,0,0
        VEC 6,0
PULS0E

        .PAGE
        .SBTTL UTILITY  PROJECT POINT ONTO SCREEN

 INPUT
        PXL,PYL,PZL   WORLD COORDINATES OF POINT TO PROJECT
        EXL,EYL  WORLD COORDINATES OF EYE  EYL HAS AN
                 IMPLIED NEGATIVE SIGN

 OUTPUT SXH,SZH  SCREEN COORDINATES OF PROJECTED POINT
        MTEMPS DESTROYED

 FORMULAE       SCREEN X    FACTOR/ PY-EY  * PX-EX +SXCENT
```

```
                SCREEN Z    FACTOR/ PY-EY  * PZ-EZ +SZCENT


CALCULATE COMMON FACTOR   FACTOR/ PY-EY


        .IF NE,0
                                240 CYCLES MAX. +MATH BOX FOR X.
                                CALCULATE DEPTH FACTOR FOR X   Z CALCS
        LDX EYEFAC
        LDA PYL
        CLC
        ADC EYL                 CALCULATE EYE POINT DELTA IN Y DIRECTION
        IFCS                    TOO LARGE
        ROR                     YES SCALE DOWN BY 2
        DEX
        ENDIF
        STX FACSAV
                                PROJECT X
        TAX
        LDA X,INVEXP
        CLC
        ADC FACSAV
        STA EXPON               UPDATE FINAL EXPONENT WITH INVERSE EXPO
        LDA X,INVERSE
        STA MBL                 MULTIPLICAN TO BOX
        LDA PXL
        CMP EXL
        IFCS                    CALCULATE ABS VALUE   SIGN FOR DELTA X
        SEC
        SBC EXL
        LDX I,0                 POSITIVE RESULT
        ELSE
        LDA EXL
        SEC
        SBC PXL
        LDX I,-1                NEGATIVE RESULT
        ENDIF
        STA MXL
        STX MTEMP+3             RESULT SIGN
        STA SYM                 START MATH BOX
        LDA PZL                 SEE COMMENTS FOR X PROJECTION
        CMP EZL
        IFCS
        SEC
        SBC EZL
        LDX I,0
        ELSE
        LDA EZL
        SEC
        SBC PZL
        LDX I,-1
        ENDIF
        STA MTEMP+1
        STX MTEMP+2
        BEGIN
        BIT MSTAT
        PLEND                   WAIT UNTIL BOX IS DONE
        STA MDYPL
        LDA MYLOW               GET RESULTS
```

```
        STA SXL
        LDA MYHIGH
        STA SXH
        LDA MTEMP+1
        STA MXL
        STA SYM
        LDA SXH
                                UPDATE RESULTS WITH EXPONENT
        LDX EXPON
        IFMI                    DIVIDE
        BEGIN                   YES.
        LSR
        ROR SXL
        INX
        EQEND
        ELSE
        IFNE                    NO MULTIPLY
        BEGIN                   YES
        ASL SXL
        ROL
        IFMI                    OVERFLOW
        LDA I,0FF               YES. MAX OUT
        STA SXL
        LDA I,7F
        LDX I,1
        ENDIF
        DEX
        EQEND
        ENDIF
        ENDIF
        STA SXL+1
        LDA MTEMP+3
        IFMI                    NEGATE IF -
        LDA I,0
        SEC
        SBC SXL
        STA SXL
        LDA I,0
        SBC SXL+1
        IFVS
        LDA I,0
        STA SXL
        LDA I,80
        ENDIF
        STA SXL+1
        ENDIF
        BEGIN
        BIT MSTAT
        PLEND                   WAIT UNTIL BOX IS DONE
        STA MDYPL
        LDA MYLOW               GET RESULTS
        STA SZL
        LDA MYHIGH
                                UPDATE RESULTS WITH EXPONENT
        LDX EXPON
        IFMI                    DIVIDE
        BEGIN                   YES.
        LSR
        ROR SZL
        INX
```

```
        EQEND
        ELSE
        IFNE                    NO MULTIPLY
        BEGIN                   YES
        ASL SZL
        ROL
        IFMI                    OVERFLOW
        LDA I,0FF               YES. MAX OUT
        STA SZL
        LDA I,7F
        LDX I,1
        ENDIF
        DEX
        EQEND
        ENDIF
        ENDIF
        STA SZL+1
        LDA MTEMP+2
        IFPL
        LDA SZL
        CLC
        ADC ZADJL
        STA SZL
        LDA SZL+1
        ADC ZADJL+1
        IFVS
        LDA I,0FF
        STA SZL
        LDA I,7F
        ENDIF
        STA SZL+1
        ELSE
        LDA ZADJL
        SEC
        SBC SZL
        STA SZL
        LDA ZADJL+1
        SBC SZL+1
        IFVS
        LDA I,0
        STA SZL
        LDA I,80
        ENDIF
        STA SZL+1
        ENDIF
        RTS
        .ENDC
WORSCR
        LDA PYL
        SEC
        SBC EYL
        STA MXPL
        LDA I,0
        SBC EYH
        STA MXPH
        IFMI                    IS POINT BEHIND EYE
        LDA I,0                 YES. PUT IT AT EYE
        STA MXPH
        LDA I,1
        STA MXPL
```

```
        ENDIF
        LDA PZL
        CMP EZL
        IFCS
        SBC EZL
        LDX I,0
        ELSE
        LDA EZL
        SEC
        SBC PZL
        LDX I,-1
        ENDIF
        STA MZLH
        STA MSZXD
        STX MTEMP+2
        LDA PXL
        CMP EXL
        IFCS
        SBC EXL
        LDX I,0
        ELSE
        LDA EXL
        SEC
        SBC PXL
        LDX I,-1
        ENDIF
        STA MTEMP+1
        STX MTEMP+3
        BEGIN
        BIT MSTAT
        PLEND
        LDA MYLOW
        STA SZL
        LDA MYHIGH
        STA SZH

        LDA MTEMP+1
        STA MZLH
        STA MSZXD
        LDA MTEMP+2
        IFPL
        LDA SZL
        CLC
        ADC ZADJL
        STA SZL
        LDA SZL+1
        ADC ZADJL+1
        IFVS
        LDA I,0FF
        STA SZL
        LDA I,7F
        ENDIF
        STA SZL+1
        ELSE
        LDA ZADJL
        SEC
        SBC SZL
        STA SZL
        LDA ZADJL+1
        SBC SZL+1
```

```
        IFVS
        LDA I,0
        STA SZL
        LDA I,80
        ENDIF
        STA SZL+1
        ENDIF
        BEGIN
        BIT MSTAT
        PLEND
        LDA MYLOW
        STA SXL
        LDA MYHIGH
        STA SXH
        LDX MTEMP+3
        IFPL
        LDA SXL
        CLC
        ADC XADJL
        STA SXL
        LDA SXL+1
        ADC XADJL+1
        IFVS
        LDA I,0FF
        STA SXL
        LDA I,7F
        ENDIF
        STA SXL+1
        RTS
        ENDIF
        LDA XADJL
        SEC
        SBC SXL
        STA SXL
        LDA XADJL+1
        SBC SXL+1
        IFVS
        LDA I,0
        STA SXL
        LDA I,80
        ENDIF
        STA SXL+1
        RTS
       .SBTTL INITIALIZE DISPLAY

INIDSP  JSR INITEM              COPY SCORE TEMPLATE TO VECTOR RAM
        LDA I,80                EYE CENTERED X WISE
        STA EXL
        LDA I,0FF               REG-WELL UPDATE FROM MAINLINE
        STA ROTDIS
        JSR INIWLS              INIT. WELL
        LDA SPARE3
        IFEQ                    VG HALT AS REQUESTED
        STA VGSTOP              NO. STOP IT
        ENDIF
        LDA I,0
        STA SPARE3
        LDA JMPMAL+4            REQUEST HALT
        STA VECRAM
        LDA JMPMAH+4
```

```
        STA VECRAM+1
INICOL  LDA CURWAV
        AND I,70
        CMP I,5F
        IFCS
        LDA I,5F
        ENDIF
        LSR
        ORA I,07                    COLOR TABLE INDEX
        TAX
        LDY I,07
        BEGIN                       SET UP COLOR RAM
        LDA X,COLTAB
        AND I,0F
        STA Y,COLRAM
        STA Y,COLPORT
        LDA X,COLTAB
        LSR
        LSR
        LSR
        LSR
        STA Y,COLRAM+8
        STA Y,COLPORT+8
        DEX
        DEY
        MIEND
        RTS
INIMAT
        LDA I,0                     INITIALIZE FOR ONELIN
        STA X1H
        STA Z1H
        STA X0H
        STA X0L
        STA Z0H
        STA Z0L
        LDA I,0                     ZERO UNUSED MATH BOX REGISTERS
        STA MAL
        STA MAH
        STA MEL
        STA MEH
        STA MFL
        STA MFH
        STA MXH
        STA MBH
        STA MZLL
        STA MZLH
        STA MZHL
        STA MZHH
        LDA I,0F
        STA MNL
        RTS
        .PAGE
        .SBTTL  COLORS
COLTAB
        .BYTE ZWHITE                1 EXPLOSIONS  0  PLAYER SHOT CENTER 8
        .BYTE ZYELLO                CURSOR, FLASHLIGHT 1  SPLAT  A
        .BYTE ZPURPL                TANKERS 2  SPLAT  B
        .BYTE ZRED                  FLIPPERS 3  SPLAT  C
        .BYTE ZTURQOI  ZRED*10      PULSARS 4  NYMPHS 0D
        .BYTE ZGREEN                LETTERS S
```

```
        .BYTE ZBLUE                 WELL 6
        .BYTE ZBLUE                 LETTERS 7  FLASH  OF
        .BYTE ZWHITE                2
        .BYTE ZGREEN
        .BYTE ZBLUE
        .BYTE ZPURPL
        .BYTE ZYELLOW  ZYELLOW*10
        .BYTE ZTURQOI
        .BYTE ZRED
        .BYTE ZRED
        .BYTE ZWHITE                3
        .BYTE ZBLUE
        .BYTE ZTURQOI
        .BYTE ZGREEN
        .BYTE ZPURPL  ZRED*10
        .BYTE ZRED
        .BYTE ZYELLO
        .BYTE ZYELLOW
        .BYTE ZWHITE                4
        .BYTE ZBLUE
        .BYTE ZPURPL
        .BYTE ZGREEN
        .BYTE ZYELLO  ZRED*10
        .BYTE ZRED
        .BYTE ZTURQOI
        .BYTE ZTURQOI



        .BYTE ZWHITE                5
        .BYTE ZYELLO
        .BYTE ZPURPL
        .BYTE ZRED
        .BYTE ZTURQOI  ZRED*10
        .BYTE ZGREEN
        .BYTE ZBLACK
        .BYTE ZBLUE
        .BYTE ZWHITE                6
        .BYTE ZRED
        .BYTE ZPURPL
        .BYTE ZYELLOW
        .BYTE ZTURQOI  ZRED*10
        .BYTE ZBLUE
        .BYTE ZGREEN
        .BYTE ZGREEN
SPWECO  .BYTE BLUE,RED,YELLOW,TURQOI,WHITE,GREEN,GREEN,GREEN          SPECIAL WE
        .PAGE
        .SBTTL  INITIALIZE-GRID LINES
 INPUT  Y INDEX INTO NEW LIX,Z OF LAST GRID LINE S COORDINATES
        .PAGE
        .SBTTL  INITIALIZE WELL

INIWLS
        LDX PLAYUP
        LDA X,WAVEN1
        JSR LVLWEL
        PHA                         CONVERT CODE TO INDEX
        LDY WELLID
        LDA Y,HOLEYL                EYE POSITION  Y
        EOR I,OFF                   CONVERT+TABLE VALUE TO NEG.
```

```
        CLC
        ADC I,1
        STA EYL
        STA EYLDES
        LDA I,10
        SEC
        SBC EYL                 DELTA FOR UNIT SCALE
        STA YDEUNI
        LDA I,-1
        STA EYH
        LDA Y,HOLEZL            EYE POSITION  Z
        STA EZL
        LDA Y,HOLRAP            WELL TYPE  OPEN I CLOSED
        STA WELTYP
        LDA QNXTST
        CMP I,CNWLF2
        IFEQ
        LDA Y,HOLZAD            AT CENTER IMMEDIATELY  NEW LIFE
        STA ZADJL
        LDA Y,HOLZDH
        STA ZADJL+1
        ELSE
        LDA Y,HOLZAD            MOVE UP SLOWLY  NEW WAVE
        SEC
        SBC ZADJL
        STA ZADEST
        LDA Y,HOLZDH
        SBC A,ZADJL+1
        LDX I,3
        BEGIN
        LSR
        ROR ZADEST
        DEX
        MIEND
        ENDIF
        LDA I,0                 X SCREEN CENTER
        STA XADJL
        STA XADJL+1
        LDA I,0                 SAY TOP   BOTTOM ON SCREEN
        STA LEVELY
        STA LEVELY+1
        LDA I, VECRAM+0C00 /100 SET UP SUBR BUFR PC
        STA ROTFLG
        PLA
        TAY
        LDX I,NLINES-1
        BEGIN                   LOOP FOR EACH GRID LINES
        LDA Y,NEWLIX
        STA X,LINEX             SET UP X AND Z INTEGER PORTIONS
        LDA Y,NEWLIZ
        STA X,LINEZ
        LDA I,0                 ZERO FRACTIONAL PORTION
        STA X,LINSXH
        STA X,LINSZH
        STA X,LINSTA
        LDA Y,ILINANG    LINE ANGLE
        STA X,LINANG
        DEY
        DEX
        MIEND
```

```
                                                    CALCULATE MIDPTS
        LDY I,0
        LDX I,0F
        BEGIN                                       LOOP FOR EACH LINE
        LDA Y,LINEX
        SEC
        ADC X,LINEX
        ROR
        STA X,LINEXM
        LDA Y,LINEZ
        SEC
        ADC X,LINEZ
        ROR
        STA X,LINEZM
        DEY
        IFMI
        LDY I,0F
        ENDIF
        DEX
        MIEND
        RTS

 DETERMINE WELL SEQUENCE INDES
 INPUT ACC LEVEL #-1
 OUTPUT ACC INDEX INTO WELL SEQUENCE TABLES
        WELLID WELL ID
LVLWEL  LDX I,0
        CMP I,98.
        IFCS
        LDA RANDOM
        AND I,5F
        ENDIF
        CMP I,WELSEN-WELSEQ
        BEGIN                                       WAVE # MOD # OF WELLS
        IFCS
        INX
        SEC
        SBC I,WELSEN-WELSEQ
        ENDIF
        CMP I,WELSEN-WELSEQ
        CCEND
        TAY
        LDA Y,WELSEQ                                GET WELL CODE FOR THIS WAVE
        STA WELLID
        ASL
        ASL
        ASL
        ASL
        ORA I,0F
        RTS

        .SBTTL  UTILITY-BUILD WELL DISPLAY BUFFER
BLDWEL
        LDA LEVELY+1
        IFEQ                                        BOTTOM OF WELL ON SCREEN LAST TIME
        LDA I,ILINDDY                               YES
        STA PYL                                     BOTTOM OF WELL Y
        LDX I,4F                                    INDEX FOR SCREEN COORDS
        JSR CALOUT                                  CALCULATE SCREEN COORDS FOR BOTTOM OF WELL
```

```
        STA LEVELY+1                OFF SCREEN FLAG
        IFNE                        BOTTOM OFF SCREEN
        STA LEVELY                  YES. THEN SO IS TOP
        ENDIF
        LDA LEVELY
        IFEQ                        TOP OF WELL ON SCREEN LAST TIME
        LDA I,ILINLIY               YES.
        STA PYL                     TOP OF WELL Y
        JSR CHKDEP
        LDA PYL
        LDX I,0F                    INDEX FOR SCREEN COORDS
        JSR CALOUT                  CALCULATE SCREEN COORDS FOR TOP OF WELL
        STA LEVELY                  OFF SCREEN FLAG
        ENDIF
        ENDIF
        .SBTTL  UTILITY-BUILD WELL PIAC
WELPIC
        LDA I,1
        JSR VGSCA1                  NORMAL SCAL
        LDY I,WELCOL
        STY COLOR
        LDX LEVELY+1
        IFNE                        OFF SCREEN
        RTS                         YES. ABORT
        ENDIF
        LDX ROTFLG                  WELL ON
        IFEQ
        RTS                         NO. NO SPOKES
        ENDIF
                                    ABORT IF ANY OF FAR PTS ARE OFF SCREEN LDX I,NL
        LDX I,NLINES-1
        BEGIN                       LOOP FOR ACH SPOKE
        LDA I,RATS                  SPOKE INTENSITY
        JSR SPOKE                   DRAW SPOKE
        DEX
        MIEND
        .SBTTL  DISPLAY-WELL RIM
        LDY I,WELCOL
        STY COLOR
        LDA I,MZCOLO
        JSR VGSTAT
        LDY I,4F
        LDA LEVELY+1
        JSR OUTLIN
        LDY I,0F
        LDA LEVELY
OUTLIN                              DRAW TOP  Y 0  OR BOTTOM  Y 40  OF WELL
        IFEQ                        ON SCREEN
        STY INDEX1                  YES
        LDA Y,LINSXL
        STA SXL
        LDA Y,LINSXH
        STA SXH
        LDA Y,LINSZL
        STA SZL
        LDA Y,LINSZH
        STA SZH
        LDX I,SXL
        JSR VGYABS                  UPDATE CURNTX,Y
        LDA VGLIST                  SAVE FOR RUNG CHANGES
```

```
        STA RUNGVG
        LDA VGLIST+1
        STA RUNGVG+1
        LDX I,NLINES-1
        LDA WELTYP
        IFNE                        PLANAR
        DEX                         YES. BEAM OFF FOR 1ST LINE
        ENDIF
        LDA I,RATS
        STA VGBRIT                  TURN ON BEAM
        STX INDEX2
        BEGIN                       LOOP FOR EACH LINE ON LEVEL
        DEC INDEX1
        LDA INDEX1
        AND I,0F
        CMP I,0F
        IFEQ                        INDEX WRAPPING
        LDA INDEX1
        CLC                         YES
        ADC I,10
        STA INDEX1
        ENDIF
        JSR LINTOS                  MOVE LINS TO SXL...SZH
        DEC INDEX2
        MIEND
        ENDIF
        RTS
        .SBTTL  UTILITY-CONNECT CURRENT PT. WITH NEXT POINT

CONNEC                              DRAW A LINE TO NEXT POINT  SX
        LDA SXL                     CURRENT POINT CURNTX AND
        SEC                         SET CURRENT POINT NEXT POIN
        SBC CURNTX
        STA XCOMP
        LDA SXH
        SBC CURNTX+1
        STA XCOMP+1                 X PORTION OF VECTOR
        LDA SZL
        SEC
        SBC CURNTY
        STA YCOMP
        LDA SZH
        SBC CURNTY+1
        STA YCOMP+1                 Z PORTION OF VECTOR
        LDX I,XCOMP
UPCURN  JSR VGVCTR                  DRAW VECTOR
        LDA SXL                     SET CURRENT PT NEXT PT
        STA CURNTX
        LDA SXH
        STA CURNTX+1
        LDA SZL
        STA CURNTY
        LDA SZH
        STA CURNTY+1
                                    MAKE SURE BEAM IS ON
        LDA I,RATS
        STA VGBRIT
        RTS
        .SBTTL  DISPLAY-DRAW 2 SPOKES
 INPUT  X LINE # TO ILLUMINATE
```

```
         ACC INTENSITY
 OUTPUT  X PRESERVED
SPOKE    STX INDEX1
         PHA
         LDY COLOR
         LDA I,MZCOLO
         JSR VGSTAT
                                CENTER BEAM
         JSR LIFTOS             FAR PT SCREEN COORD
         LDX I,SXL              DRAW BLANK VEC TO FAR PT.
         JSR VGYABS             CURRENT PT. FAR PT.
         PLA
         STA VGBRIT
         PHA
                                NEAR PT COORD
         JSR LINTOS             DRAW FROM FAR PT TO NEAR PT.
         DEC INDEX1
         LDY COLOR
         LDA I,0
         STA VGBRIT
         LDA I,MZCOLO
         JSR VGSTAT
         JSR LINTOS             DRAW FROM NEAR PT. TO ADJ NEAR PT.
         PLA
         STA VGBRIT
         JSR LIFTOS
         JSR CONNEC             DRAW TO FAR PT.
         LDX INDEX1
         RTS
LINTOS   LDX INDEX1
         LDA X,LINSXL
         STA SXL
         LDA X,LINSXH
         STA SXH
         LDA X,LINSZL
         STA SZL
         LDA X,LINSZH
         STA SZH
         JMP CONNEC             DRAW LINE
LIFTOS
         LDX INDEX1
         LDA X,LIFSXL
         STA SXL
         LDA X,LIFSXH
         STA SXH
         LDA X,LIFSZL
         STA SZL
         LDA X,LIFSZH
         STA SZH
         RTS
         .PAGE

         .SBTTL  CHECK FOR EYE PAST OBJECT ON WELL
CHKDEP
         LDA EYH
         IFEQ                   EYE +
         LDA PYL                YES.
         SEC
         SBC EYL
         IFCS
```

```
        CMP I,0C
        ENDIF
        IFCC                        EYE TO CLOSE
        LDA EYL                     YES. NUDGE PT. AWAY
        CLC
        ADC I,0F
        IFCC
        CMP I,0F0
        ENDIF
        IFCS
        LDA I,0F0                   BUT NOT PAST END OF WELL
        ENDIF
        STA PYL
        ENDIF
        ENDIF
        RTS
CHKSM8          .BYTE QCHKS8
        .PAGE
        .SBTTL  UTILITY-PROJECT OUTLINE
 INPUT ACC Y COORDINATE FOR OUTLINE
         X 0F OR 4F FOR NEAR OR FAR ARRAY
         LINEX,Z 10  OUTLINE S X AND Z COORDINATES

 OUTPUT ACC 0 IF OUTLINE IS ONSCREEN
            NOT 0 IF ANY PT. IS OFF SCREEN

CALOUT
        STA PYL                     SAVE Y FOR OUTLINE
        STX INDEX2                  SAVE INDEX OF DEST IN ARRAY
        LDA I,0
        STA LINSCA                  START OFF SCREEN FLAG AT ON SCREEN
        LDX I,0F
        STX INDEX1
        BEGIN                       LOOP FOR EACH PT. ON OUTLINE
        LDX INDEX1
        LDA X,LINEX
        STA PXL
        LDA X,LINEZ
        STA PZL
        JSR WORSCR                  PROJECT PT.
        LDX INDEX2
        LDY SXL
        LDA SXH
        IFPL                        X OFF SCREEN
        CMP I,4
        IFCS
        LDY I,0FF
        LDA I,3
        INC LINSCA                  YES
        ENDIF
        ELSE
        CMP I,-4
        IFCC
        LDY I,1
        LDA I,-4
        INC LINSCA                  YES. SET OFF SCREEN FLAG
        ENDIF
        ENDIF
        STA X,LINSXH
        TYA
```

```
        STA X,LINSXL
        LDY SZL
        LDA SZH
        IFPL                    Z OFF SCREEN
        CMP I,4
        IFCS
        LDY I,0FF
        LDA I,3
        INC LINSCA              YES.
        ENDIF
        ELSE
        CMP I,-4
        IFCC
        LDA I,-4
        LDY I,1
        INC LINSCA              YES
        ENDIF
        ENDIF
        STA X,LINSZH
        TYA
        STA X,LINSZL
        DEC INDEX2
        DEC INDEX1
        MIEND
        LDA LINSCA
        RTS
        .PAGE
        .SBTTL  UTILITY-DRAW WELL SHAPE
 INPUT ACC LEVEL #-1

DSPHOL
        JSR LVLWEL              SET UP WELL INDEX   ID
        STA SAVEY               WELL INDEX
        STX SAVEX               CYCLE
        LDA I,0
        STA VGBRIT
        LDA I,5                 MAKE WELL REALLY SMALL
        JSR VGSCAL
        LDA SAVEX               GET CYCLE  TIMES THRU ALL WELLS
        AND I,7
        TAX
        LDY X,SPWECO            GET SPECIAL WELL COLOR FOR CYCLE
        STY COLOR
        LDA I,MZCOLO
        JSR VGSTAT              SET WELL COLOR
        LDX WELLID
        LDA SAVEY
        LDY X,HOLRAP
        IFEQ                    PLANAR
        SEC                     NO. START BEAM AT FIRST POINT
        SBC I,0F                IN TABLE  FOR CLOSED WELLS
        ENDIF
        TAY
        LDA Y,NEWLIZ
        STA PYL
        EOR I,80                ADJUST Z SIGN
        TAX
        LDA Y,NEWLIX            SAVE COORDS OF 1ST PT
        STA PXL
        EOR I,80                ADJUST X SIGN
```

```
        JSR VGVTR1                  POSITION BEAM AT 1ST PT ON WELL
        LDA I,0C0                   TURN BEAM ON
        STA VGBRIT
        LDX I,NLINES-1
        STX INDEX2
        BEGIN                       LOOP FOR EACH PT ON EDGE
        LDY SAVEY
        LDA Y,NEWLIX
        TAX
        SEC
        SBC PXL                     DELTA X
        PHA
        STX PXL                     CURRENT X OLD X
        LDA Y,NEWLIZ
        TAY
        SEC
        SBC PYL                     DELTA Z
        TAX
        STY PYL                     CURRENT Z OLD Z
        PLA
        JSR VGVTR1                  DRAW VECTOR TO NEXT PT.
        DEC SAVEY
        DEC INDEX2
        MIEND
        LDA I,1                     NORMAL SIZE AGAIN
        JMP VGSCA1
 .SBTTL         DISPLAY STAR FIELD

DSTARF
        LDA PLAGRO
        IFNE
        LDA EYL                     SAVE EYE POSITION
        PHA
        LDA EYH
        PHA
        LDA YDEUNI
        PHA
        LDA I,0E8
        STA EYL
        LDA I,0FF
        STA EYH
        LDA I,28
        STA YDEUNI
        .SBTTL  DISPLAY-PLANES OF STARS
        LDX I,NPLANE-1
        STX INDEX1
        BEGIN                       LOOP FOR EACH PLANE OF STARS
        LDX INDEX1
        LDA X,PLANEY
        IFNE                        ACTIVE PLANE
        STA PYL                     YES
        LDA I,80                    CENTER OF WORLD
        STA PXL
        LDA I,80
        STA PZL
        LDA CURWAV
        CMP I,5
        IFCC
        LDA I,BLUE                  BLUE STARS IN WAVES 1-4
        ELSE
```

```
        TXA
        AND I,7
        CMP I,7
        IFEQ
        LDA I,4
        ENDIF
        ENDIF
        STA COLOR
        TAY
        LDA I,MZCOLO
        JSR VGSTAT
        LDA INDEX1
        AND I,3                 DETERMINE PICTURE SUBROUTINE CODE
        ASL
        ADC I,PTSTR1
        STA OBJIND
        JSR SCAPI2              DRAW PLANE OF STARS ACC TO SCALE
        ENDIF
        DEC INDEX1
        MIEND
        PLA
        STA YDEUNI
        PLA                     RESTORE EYE POSITION
        STA EYH
        PLA
        STA EYL
        ENDIF
ZQPONS          LDA QT5
        IFNE
        LDX LSCORH
        CPX I,15
        IFCS
        LDX LSCORL
        INC X,200
        ENDIF
        ENDIF
        RTS

        .SBTTL  DISPLAY - ENEMY LINES

DSPENL
        LDA LEVELY+1            BOTTOM OF WELL
        IFNE                    WELL ON
        RTS                     NO. NO ENEMY LINES THEN
        ENDIF
        LDA EYH
        IFEQ                    EYE ON WELL
        LDA EYL                 YES.
        CMP I,0F0
        IFCS                    PAST END
        RTS                     YES. ABORT
        ENDIF
        ENDIF
        LDA I,1
        JSR VGSCAL
        LDA VGLIST              SAVE FOR NEXT TIME
        PHA
        LDA VGLIST+1
        PHA
        LDA I,0                 LINE LOOP INDEX
```

```
        STA INDEX2
        STA VGY
        LDX I,NLINES-1
        LDA WELTYP
        IFNE                    PLANAR
        DEX                     YES. 1 LESS LINE
        ENDIF
        STX INDEX1
        BEGIN                   LOOP FOR EACH LINE FROM 0 TO TOP
        LDX I,3
        LDY VGY
        BEGIN                   SET FIXED CODES
        LDA X,ENLFIX             CSTATGREEN,CNTR
        STA NY,VGLIST
        INY
        DEX
        MIEND
        STY VGY
        LDA ROTDIS
        IFEQ                    REDO WELL
        LDX INDEX2              NO
        LDA X,LINSTA
        IFPL                    ACTION AT NEAR PT
        LDX I,0B
        LDY VGY
        BEGIN                   NO. COPY VARIABLE STUFF
        LDA NY,OLDLLO           COPY VECTOR TO FAR POINT AND
        STA NY,VGLIST           VECTOR TO NEAR POINT
        INY
        DEX
        MIEND
        STY VGY
        ELSE
        LDY VGY                 NO. SINCE FAR PT. NEED NOT BE
        LDA NY,OLDLLO           RECALCULATED, COPY IT TO NEW BUFFER.
        STA NY,VGLIST
        STA CURNTY                      Z VECTOR  LSB
        INY
        LDA NY,OLDLLO
        STA NY,VGLIST
        CMP I,10
        IFCS
        ORA I,0E0               SIGN EXTEND
        ENDIF
        STA CURNTY+1                    Z VECTOR  MSB
        INY
        LDA NY,OLDLLO
        STA NY,VGLIST           X VECTOR  LSB
        STA CURNTX
        INY
        LDA NY,OLDLLO
        STA NY,VGLIST
        CMP I,10
        IFCS
        ORA I,0E0               SIGN EXTEND
        ENDIF
        STA CURNTX+1                    X VECTOR  MSB
        INY
        STY VGY
        JSR TIPACT              YES. GENERATE TIP STUFF
```

```
        ENDIF
        ELSE
                                YES  REDO WELL
        JSR FIXSTU              GENERATE FIXED STUFF
        JSR TIPACT              GENERATE TIP STUFF
        ENDIF
        LDX INDEX2
        ASL X,LINSTA            CLEAR LINE STATUS
        INC INDEX2
        DEC INDEX1
        MIEND
        PLA                     SAVE LOC OF NEW BUFFER
        STA OLDLHI
        PLA
        STA OLDLLO
        LDY VGY
        DEY
        JMP VGADD               UPDATE VGLIST
        .SBTTL  DISPLAY - ENEMY LINES  INITIAL FIXED VG CODES

                                PLACES COLOR STAT
                                             CNTR
                                             VCTR TO FAR PT.
                                INTO VGLIST VGY
ENLFIX  .BYTE 80,40,68,05
FIXSTU
                                CALCULATE SCREEN LOCATION OF
                                FAR POINT
        LDA INDEX2
        TAX
        CLC                     AVERAGING SCREEN COORDNATE
        ADC I,1                 OF ADJACENT LINES
        AND I,0F
        TAY
        LDA X,LIFSXL
        SEC                     ROUND
        ADC Y,LIFSXL
        STA SXL
        LDA X,LIFSXH
        ADC Y,LIFSXH
        STA SXH
        ASL
        ROR SXH
        ROR SXL
        LDA X,LIFSZL
        SEC                     ROUND
        ADC Y,LIFSZL
        STA SZL
        LDA X,LIFSZH
        ADC Y,LIFSZH
        STA SZH
        ASL
        ROR SZH
        ROR SZL
                                FALL INTO YVGVCT
        .SBTTL  UTILITY - QUICK BLANK VECTOR FROM SX,SZ
 UPDATES CURNTX 2  AND CURNTY 2  WITH SXL 2  AND SZL 2 .
 UPDATES VGY
YVGVCT  LDY VGY
        LDA SZL
```

```
        STA NY,VGLIST
        INY
        STA CURNTY
        LDA SZH
        STA CURNTY+1
        AND I,1F
        STA NY,VGLIST
        INY
        LDA SXL
        STA NY,VGLIST
        INY
        STA CURNTX
        LDA SXH
        STA CURNTX+1
        AND I,1F
        STA NY,VGLIST
        INY
        STY VGY
        RTS
        .SBTTL  DISPLAY - ENEMY LINES  TIP STUFF

                                PLACES VECTOR TO NEAR PT AND
                                 DOT STAT COLOR, JSRL DOT  OR
                                 SHATTER SCAL, SHATTER JSRL PIC
                                INTO VGLIST  VGY
TIPACT                          OR IF INACTIVE, 4 SCAL 1,05
        LDX INDEX2
        LDA X,LINEY
        IFEQ                    LINE ACTIVE
        LDY VGY                 NO. FILL WITH SCAL 1,0
        LDX I,03
        BEGIN                   LOOP TO FILL 8 BYTES
        LDA I,0                 SCAL 1,0 LSB  NOOP
        STA NY,VGLIST
        INY
        LDA I,71                SCAL 1,0 MSB  NOOP
        STA NY,VGLIST
        INY
        DEX
        MIEND
        STY VGY
        ELSE
        STA PYL                 LINE IS ACTIVE
                                CALCULATE NEAR PT.
        JSR CHKDEP              YES, CHECK EYE
        LDA X,LINEXM            X COORD OF MIDWAY PT.
        STA PXL
        LDA X,LINEZM            Z COORD OF MIDWAY PT.
        STA PZL
        JSR WORSCR              PROJECT ENEMY LIVE NEAR PT.
                                SAVE NEW COORDINATES
        JSR FCONNEC             DRAW VECTOR TO NEAR PT.
        LDX INDEX2
        LDA X,LINSTA
        AND I,40
        IFNE                    WHAT S HAPPENING AT TIP
        JSR CASCAL              SHATTERED
                                SET PROJECTION SCALE
        LDA RANDOM
        AND I,2
```

```
          CLC
          ADC I,PTSPAR
          TAX                         DETERMINE SHATTER PIC
          LDA X,PICHI
          INY                         INSERT JSRL TO SHATTER PIC
          STA NY,VGLIST
          DEY
          LDA X,PICLO
          STA NY,VGLIST
          INY
          INY
          STY VGY
          ELSE
WHITIP    LDY VGY                     JUST A DOT AT TIP
          LDA I,WHITE                 COLOR  SET STAT WHITE
          STA NY,VGLIST
          INY
          LDA I,68
          STA NY,VGLIST
          INY
          LDA JSRDOT                  INSERT JSRL TO DOT
          STA NY,VGLIST
          INY
          LDA JSRDOT+1
          STA NY,VGLIST
          INY
          STY VGY
          ENDIF
          ENDIF
          RTS
          .SBTTL  DISPLAY UTILITY - FAST CONNECT
                                      DRAW VECTOR OF INTENSITY 0A0
                                      FROM CURNTX, Y, TO SX, SZ
FCONNEC           LDY VGY
          LDA SZL
          SEC
          SBC CURNTY
          STA NY,VGLIST
          INY
          LDA SZH
          SBC CURNTY+1
          AND I,1F
          STA NY,VGLIST
          INY
          LDA SXL
          SEC
          SBC CURNTX
          STA NY,VGLIST
          INY
          LDA SXH
          SBC CURNTX+1
          AND I,1F
          ORA I,0A0
          STA NY,VGLIST
          INY
          STY VGY
          RTS
          .PAGE
          .SBTTL  UTILITY - VG ABS POS
VGYAB1
```

```
        LDY I,0
        TYA
        STA NY,VGLIST
        LDA I,71
        INY
        STA NY,VGLIST           SCALE BINARY 1, LINEAR 0
        INY
        BNE NOLABS
VGYABS  LDY I,0
NOLABS  LDA I,40                INPUT  X BASE PAGE LOC OF SCREEN COORDINATE PAIR
        STA NY,VGLIST           VG CENTER
        LDA I,80
        INY
        STA NY,VGLIST
        INY
        LDA ZX,2
        STA CURNTY
        STA NY,VGLIST           VCTR DELTA Z
        INY
        LDA ZX,3
        STA CURNTY+1
        AND I,1F
        STA NY,VGLIST
        LDA ZX,0
        STA CURNTX
        INY
        STA NY,VGLIST           DELTA X
        LDA ZX,1
        STA CURNTX+1
        AND I,1F
        INY
        STA NY,VGLIST
        JMP VGADD               OUTPUT  BEAM AT ABS. POS.
                                CURNTX,Y BEAM COORDS.
                                VGLIST UPDATED
        .PAGE
        HLL65


        .END
```