

spacewar 4.8 7/24/63 dfw .pt 1

```
3/      jmp sbf          / ignore seq. break
        jmp a40
        nop
```

/ interesting and often changed constants

/symb loc usual value (all instructions are executed,
/ and may be replaced by jda or jsp)

```
tno, 6,  law i 41          / number of torps + 1
tv1, 7,  sar 4s          / torpedo velocity
rlt, 10, law i 20          / torpedo reload time
tlf, 11, law i 140         / torpedo life
foo, 12, -20000          / fuel supply
maa, 13, 40              / spaceship angular acceleration
sac, 14, sar 4s          / spaceship acceleration
str, 15, 100             / star capture radius
me1, 16, 6000            / collision "radius"
me2, 17, 3000            / above/2
ddd, 20, -0              / 0 to save space for ddt
the, 21, sar 9s          / amount of torpedo space warpage
mhs, 22, law i 10         / number of hyperspace shots
hd1, 23, law i 40         / time in hyperspace before breakout
hd2, 24, law i 100        / time in hyperspace breakout
hd3, 25, law i 200        / time to recharge hyperfield generators
hr1, 26, scl 9s          / scale on hyperspatial displacement
hr2, 27, scl 4s          / scale on hyperspatially induced velocity
hur, 30, 40000           / hyperspatial uncertainty
ran, 31, 0                / random number
grv, 32, sar 6s          / gravitational constant
```

/ place to build a private control word routine.
/ it should leave the control word in the io as follows:
/ high order 6 bits, high acceleration, normal acceleration,
/ rotate cw, rotate ccw, fire torpedo, and hyperspace.
/ Routine is entered by jsp cwg for ss1 and by jsp dwg
/ for ss2.

40/

```
cwr,      jmp mg1 / normally iot 11 control
dwr,      jmp mg2 / normally iot 111 control
. 20/     / space
```

/ routine to flush sequence breakes, if they occur.

```
sbf,      tyi
          lio 2
          lac 0
          lsm
          jmp i 1
```

```
          define
xincr X,Y,INS
          lac Y
          INS ssn
          dac Y
          lac X
          INS scn
          dac X
          term
```

```
          define
yincr X,Y,INS
          lac Y
          INS scn
          dac Y
          lac X
          -INS+add+sub ssn
          dac X
          terminate
```

```
          define
dispatch
          add (. 3
          dap . 1
          jmp .
          term
```

```
          define
dispt A,Y,B
          repeat 6          B=B+B
          lio Y
          dpy-A+B
          term
```

```
          define
scale A,B,C
          lac A
          sar B
          dac C
          term
```

```
diff V,S,SF      defin
                  add i V
                  dac i V
                  xct SF
                  add i S
                  dac i S
                  term
```

```
random          define
                lac ran
                rar 1s
                xor (355670
                add (355670
                dac ran
                term
```

```
ranct S,SS,C    define
                random
                S
                SS
                sma
                cma
                dac C
                terminate
```

```

define
    varsft
    dzm xys
    dac t1
    v2,
    idx xys
    lac t1
    scr 2s
    dac t1
    sza
    jmp v2+R
    scr 2s
    swap
    terminate

define
    undosft
    dac t1
    dio t2
    lac xys
    add stb
    dap .+1
    lac .
    dac .+6
    dac .+6
    xor (10000) / change scr to scl or scl to scr.
    dac xyt
    lac t1
    dio t2
    scr .
    scr .
    terminate

define
    integrate A,B
    cli
    lac i A
    scr 9s
    scr 1s
    div t1
    hlt
    cma+cli-opr
    xct xyt
    xct grv
    dac B
    terminate

stb,
    lac .-1
    scr 7s
    scr 6s
    scr 5s
    scr 4s
    scr 3s
    scr 2s
    scr 1s
    scr
    scl 1s

```

/sine-cosine subroutine.Adams associates
/calling sequence= number in AC, jda sin or jdacos.
/argument is between +2 pi, with binary point to right of bit 3.
/answer has binary point to right bit 0. Time = 2.35-? ms.
/changed for auto-multiply , ddp 1/19/63

```
cos,      0
          dap csx
          lac (62210
          add cos
          dac sin
          jmp .+4

sin,      0
          dap csx
          lac sin
          spa
s11,     add (311040
          sub (62210
          sma
          jmp s12
          add (62210

s13,     ral 2s
          mul (242763
          dac sin
          mul sin
          dac cos
          mul (756103
          add (121312
          mul cos
          add (532511
          mul cos
          add (144417
          mul sin
          scl 3s
          dac cos
          xor sin
          sma
          jmp csx-1
          lac (377777
          lio sin
          spi
          cma
          jmp csx

csx,     lac cos
          jmp .

s12,     cma
          add (62210
          sma
          jmp s13
          add (62210
          spa
          jmp .+3
          sub (62210
          jmp s13

          sub (62210
```

jmp si1

/integer square root

/input in ac, binary point to right of bit 17, jda sqt

/answer in ac with binary point between bits 8 and 9

/la. st inp t number = 177777

```
sqt,      0
          dap sqx
          law i 23
          dac sq1
          dzm sq2
          lio sqt
          dzm sqt

sq3,      isp sq1
          jmp .+3
          la sq2

sqx,      jmp .

          lac sq2
          sal 1s
          dac sq2
          lac sqt
          rel 2s
          sza i
          jmp sq3
          dac sqt
          lac sq2
          sal 1s
          add (1
          sub sqt
          sma+sza-skp
          jmp sq3
          spa
          cma
          dac sqt
          idx sq2
          jmp sq3

sq1,      0
sq2,      0
```

/outline compiler
/ac=where to compile to, call jda oc
/ot=address of outline table

define

plinst A
lac A
dac i oc
idx oc
termina e

define

comtab A, B
plinst A
jsp ocs
a B
mp oce
terminate

ocs, dap ocz /puts in swap

dio i oc
idx oc
dio i oc
idx oc

ocz, jmp .

oc, 0 /outline compiler proper

dap ocx
lac i ocx
dap ocg
plinst (stf 5
dap ocm
idx ocx

ock, plinst (lac $\overline{sx1}$
plinst (lio $\overline{sy1}$
clf 6

ocj, setup $\overline{occ}, 6$

ocg, lio . /outline ta le

och,

cla
rcl 3s
dio \overline{oci}
lio (rcl 9s
dispatch
opr

oco, jmp oc1

ocq, jmp oc2

ocp, jmp oc3

ocr, jmp oc4

ocr, jmp oc5

jmp oc6

```

    plinst (szf 5      /7 code
    add (4
    dap ocn
    plinst ocn
    plinst (dac  $\overline{sx1}$ 
    plinst (dio  $\overline{sy1}$ 
    plinst (jmp sq6
    plinst (clf 5
    plinst (lac  $\overline{scm}$ 
    plinst (cma
    plinst (dac  $\overline{scm}$ 
    plinst (lac  $\overline{ssm}$ 
    plinst (cma
    plinst (dac  $\overline{ssm}$ 
    plinst (lac  $\overline{csm}$ 
    plinst (lio  $\overline{ssd}$ 
    plinst (dac  $\overline{ssd}$ 
    plinst (dio  $\overline{csm}$ 

    p nst (lac  $\overline{ssc}$ 
    plinst (lio  $\overline{csn}$ 
    plinst (dac  $\overline{csn}$ 
    plinst (dio  $\overline{ssc}$ 
    plinst ocm
ocx,    jmp .

ocm,    jmp .
ocn,    jmp .

oc1,    plinst (add  $\overline{ssn}$ 
        jsp ocs
        lac (sub  $\overline{scn}$ 
oce,    dac i oc
        idx oc
        jsp ocs
        plinst (ioh
        lac (dpy-4000
ocd,    dac i oc
        idx oc
        lio  $\overline{oci}$ 
        count oce, och
        idx ocg
        jmp ocj

oc2,    comtab (add  $\overline{scm}$ , (add  $\overline{ssm}$ 
oc3,    comtab (add  $\overline{ssc}$ , (sub  $\overline{csm}$ 
oc4,    comtab (sub  $\overline{scm}$ , (sub  $\overline{ssm}$ 
oc5,    comtab (add  $\overline{csn}$ , (sub  $\overline{ssd}$ 
oc6,    szf 6
        jmp oc9
        stf 6
        plinst (dac  $\overline{ssa}$ 
        lac (dio  $\overline{ssi}$ 
        jmp ocd
oc9,    clf 6
        plinst (lac  $\overline{ssa}$ 
        lac (lio  $\overline{ssi}$ 
        jmp ocd

```


/display a star

define

starp
add bx
swap
add by
swap
ioh
dpy-4000
terminate

blp, dap blx /star

szs 60
jmp blx
random
rar 9s
and (400700
spa
xor (377777
dac bx
lac ran
ral 4s
and (400700
spa
xor (377777
dac by
jsp bpt
ioh

blx, jmp .

bpt, dap bpx
random
sar 9s
sar 6s
spa
cma
sal 3s
add (bds
dap bjm
cla cli clf 6-opr-opr
dpy-4000

bjm, jmp .
bds, repeat 10, starp

szf 6
jmp .
stf 6
cma
swap
cma
swap
jmp bjm

/background display • 3/13/62, prs.

```
define
dislis J, Q, B
  repeat 6, B=B+B
  clf 5
  lac flo+R
  dap fpo+R
fs,    dap fin+R
       dap fyn+R
       idx fyn+R

fin,   lac      /lac X
       sub fpr  /right margin
       sma
       jmp fgr+R
       add (2000

frr,   spq
fou,   jmp fuu+R
fie,   sub (1000
       sal 8s
fyn,   lio      /lio Y
       dpy-i+B
       stf 5
fid,   idx fyn+R
       sad (lio Q+2
       jmp flp+R
       sad fpo+R
       jmp fx+R
       dap fin+R
       idx fyn+R
       jmp fin+R

fgr,   add (-20000+2000
       jmp frr+R

fuu,   szf 5
fx,    jmp flo+R+1      /return
       idx flo+R
       idx flo+R
       sas (Q+2
       jmp fid+R
       law J
       dac flo+R
       jmp fid+R

flp,   lac (lio J
       sad fpo+R
JP FX+R
       dap fin+R
       law J+1
       dap fyn+R
       jmp fin+R

fpo,   lio
flo,   J
       terminate
```

```

define
background      jsp bck
                  termin

bck,             dap bcx
                  szs 40
                  jmp bcx
                  isp bcc
bcx,             jmp .
                  law i 2
                  dac bcc
                  dislis 1j,1q,3
                  dislis 2j,2q,2
                  dislis 3j,3q,1
                  dislis 4j,4q,0
                  isp bkc
                  jmp bcx
                  law i 20
                  dac bkc
                  law i 1
                  add fpr
                  spa
                  add (20000
                  dac fpr
                  jmp bcx

bcc,             0
bkc,             0
fpr,             10000

```

```

mul=mus
div=dis

```

```

start

```