```
3/          jmp sbf                 / ignore seq. break
            jmp a40
            jmp a1      / use test word for control, not iot 11 co

/ 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
tvl,  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,    10          / 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 uncertancy
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 4 bits, rotate ccw, rotate cw, (both mean hyperspace)
/    fire rocket, and fire torpedo. Low order 4 bits, same for
/    other ship.  Routine is entered by jsp cwg.

40/

cwr,        jmp mg1     / normally iot 11 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 sn
            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 sn
            dac X
            terminate

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

            define
dispt A,Y,B
            repeat 6  B=B+B
            lio Y
            jda kcb
            xct db1
            term

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

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


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

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

```
define
        varsft
        dzm xys
        dac t1
        idx xys
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 sft
        dap .+
        lac .
        dac .+6
        dac .+6
        xor (10000          / changescr 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

sft,    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 of 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
si1,    add (311040
        sub (62210
        sma
        jmp si2
        add (62210

si3,    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

        lac cos
csx,    jmp .

si2,    cma
        add (62210
        sma
        jmp si3
        add (62210
        spa
        jmp .+3
        sub (62210
        jmp si3

        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
/largest input number = 177777

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

sq3,    isp sq1
        jmp .+3
        lac sq2
sqx,    jmp .

        lac sq2
        sal 1s
        dac sq2
        lac sqt
        rcl 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
        terminate

define
        contab A, B
        plinst A
        jsp ocs
        lac B
        jmp oce
        terminate

ocs,    dap ocz                  /puts in swap
        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 sy1
        plinst (lio sy1
        clf 6
ocj,    setup occ,6
ocg,    lio .                    /outline table
och,    cla
        rcl 3s
        dio oci
        lio (swp                 swp=opr 60
        dispatch
        opr
        jmp oc1
oco,    jmp oc2
ocq,    jmp oc3
ocp,    jmp oc4
ocr,    jmp oc5
        jmp oc6
```

```
        plinst (szf 5           /7 code
        add (4
        dap ocn
        plinst ocn
        plinst (dac sx1
        plinst (dio sy1
        plinst (jmp sc6
        plinst (clf 5
        plinst (lac scm
        plinst (cma
        plinst (dac scm
        plinst (lac ssm
        plinst (cma
        plinst (dac ssm
        plinst (lac csm
        plinst (lio ssd
        plinst (dac ssd
        plinst (dio csm

        plinst (lac sse
        plinst (lio csn
        plinst (dac csn
        plinst (dio sse
        plinst ocm
ocx,    jmp .

ocm,    jmp .
ocn,    jmp .

oc1,    plinst (add sen
        jsp ocs
        lac (sub sen
oce,    dac i oc
        idx oc
        jsp ocs
        plinst (ioh
        lac (xct db2
ocd,    dac i oc
        idx oc
        lio oci
        count occ, och
        idx ocg
        jmp ocj

oc2,    comtab (add sex, (add ssm
oc3,    comtab (add sse, (sub csm
oc4,    comtab (sub sem, (sub ssm
oc5,    comtab (add csn, (sub ssd
oc6,    szf 6
        jmp oc9
        stf 6
        plinst (dac ssa
        lac (dio ssi
        jmp ocd
oc9,    clf 6
        plinst (lac ssa
        lac (lio ssi
        jmp ocd
```

```
/display gravitational stars

define
        starp
        add bx
        swap
        add by
        swap
        ioh
        xct db2
        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 bpr
        random
        sar 9s
        sar 6s
        spa
        cma
        sal 3s
        add (bds
        dap bjm
        clf 6
bjl,    cla cli-opr
        szf 4
        jmp bjc
        sub ny1 1
        swp
        sub nx1 1
        jmp bjm-1
bjc,    sub ny1
        swp
        sub nx1
        xct db2
bjm,    jmp .
bds,    repeat 10, starp
        szf 6
bpx,    jmp .
        stf 6
        lac bx
        cma
        dac bx
```

```
            jmp bgl

/background display - 3/12/68, pms.

            define
dislis J,   0
            dap fx+R
            clf 5
            lac flo+R
            dap fpo+R
fs,         dap fin+R
            dap fyn+R
            idx fyn+R

fin,        lac         /lac Y
            sub fpr     /right margin
            sma
            sub (20000
            add (2000
frr,        spq
fou,        jmp fuu+R
fie,        sub (1000
            sal 8s
fyn,        lio         /lio Y
            jda kcb
            xct dbl
            stf 5
fid,        idx fyn+R
            sad (lio 0+2
            jmp flp+R
            sad fpo+R
            jmp fx+R
            dap fin+R
            idx fyn+R
            jmp fin+R

fuu,        szf 5
fx,         jmp         /return
            idx flo+R
            idx flo+R
            sas (0+2
            jmp fid+R
            law J
            dac flo+R
            jmp fid+R

flp,        lac (lio J
            sad fpo+R
            jmp fx+R
            dap fin+R
            law J+
            dap fyn+R
            jmp fin+R

fpo,        lio
flo,        J
            terminate
```

```
            define
background
            jsp bck
            termin

bck,        dap bcx
            szs 40
            jmp bcx
            jsp 1m
            jsp 2m
            idx bcc
            and (1
            sza
            jsp 3m
bc1,        law 3
            and bcc
            sza i
            jsp 4m
bc3,        isp bkc
            jmp bcy
            law i 10
            dac bkc
            law i 1
            add fpr
            spa
            add (20000
            dac fpr
bcy,        jsp 1m
bcx,        jmp

1m,         dislis 1j, 1q
2m,         dislis 2j, 2q
3m,         dislis 3j, 3q
4m,         dislis 4j, 4q

bcc,        0
bkc,        0
fpr,        10000
```

```
mul=mus
div=dis

start
```

```
/main control routine for spaceships

nob=30                          /total number of colliding objects

ml0,    setup mtc, 5000         /delay for loop
        init ml1, mtb           /loc of calc routines
        add (nob
        dap mx1                 / x
nx1=mtb nob
        add (nob
        dap my1                 / y
ny1=nx1 nob
        add (nob
        dap ma1                 / count for length of explosion or torp
na1=ny1 nob
        add (nob
        dap mb1                 / count of instructions taken by calc routine
nb1=na1 nob
        add (nob
        dac mdx                 / dx
ndx=nb1 nob
        add (nob
        dac mdy                 / dy
ndy=ndx nob
        add (nob
        dap mom                 /angular velocity
nom=ndy nob
        add (2
        dap mth                 / angle
nth=nom 2
        add (2
        dac mfu                 /fuel
nfu=nth 2
        add (2
        dac mtr                 / no torps remaining
ntr=nfu 2
        add (2
        dap mot                 / outline of spaceship
not=ntr 2
        add (2
        dap mco                 / old control word
nco=not 2
        add (2
        dac mh1
nh1=nco 2
        add (2
        dac mh2
nh2=nh1 2
        add (2
        dac mh3
nh3=nh2 2
        add (2
        dac mh4
nh4=nh3 2
nnn=nh4 2
```

```
4       szf 4
        jmp . 4
        law dj6
        stf 4
        jmp . 3
        law dj5
        clf 4
        dap . 1
        lac .
        dac db1
        idx .-2
        xct .-3
        dac db2

        law ss1
        xor mtb
        sza
        jmp mdn
        law ss2
        xor mtb 1
        sza
        jmp mdn
        law 1       / test if both ships out of torps
        add ntr
        spa
        jmp md1
        law 1
        add ntr 1
        spa i
        jmp mdn
md1,    xct tlf     / restart delay is 2X torpedo life
        sal 1s
        dac ntd
        jmp ml1

mdn,    count ntd,ml1
        stf 1
        stf 2
        law ss1
        xor mtb
        sza
        clf 1
        sza i
        idx 1sc
        law ss2
        xor mtb 1
        sza
        clf 2
        sza i
        idx 2sc
        clf 2
        jmp a
db1,    0
db2,    0
dj5,    dpy-1
        dpy-4000
dj6,    dpy-1 400
        dpy-4000 400
```

```
a1,     law mg2                 / test word control
        dac cwg
        jmp a

a40,    law cwr     / here from start at 4
        dac cwg
        jmp a6

1sc,    0           /scores
2sc,    0

a,      lac gct
        sma
        jmp a5
        count gct, a5
        lac 1sc
        sas 2sc
        jmp a4
        lac fiu
        sad (jmp 4
        jmp a4
        law i1
        dac gct
a5,     lat
        and (40
        sza i
        jmp a2
a4,     jmp fi1
        lat
        and (40
        sza
        jmp a2
        dzm 1sc
        dzm 2sc
a6,     lat
        rar 6s
        and (37
        sza
        cma
        dac gct
a2,     clear mtb, nnn-1    / clear out all tables
        law ss1
        dac mtb
        law ss2
        dac mtb 1
        lac (200000
        dac mx1
        dac my1
        cma
        dac mx1 1
        dac my1 1
        lac (144420
        dac mth
```

```
        law nnn    / start of outline program
        dac not
        lio ddd
        spi i
        jmp a3
        jda oc                  / compile outline
        ot1
a3,     dac not 1
        dap fi1
        jda oc
        ot2
        dap fi2
        xct tno
        dac ntr
        dac ntr 1
        lac foo
        dac nfu
        dac nfu+1
        law 2000
        dac nb1
        dac nb1 1
        xct mhs
        dac nh2
        dac nh2 1
        jmp m10
```

```
/ cotrol word get routines

mg1,    dap mg3
        cli
        iot 11
        rir 4s
mg3,    jmp .

mg2,    dap mg4
        lat
        swap
mg4,    jmp .

idl,    idx mth
        idx mfu
        idx mtr
        idx mco
        idx mot
        idx mom
        idx mh1
        idx mh2
        idx mh3
        idx mh4
ids,    idx mx1
        idx my1
        idx ma1
        idx mb1
        idx mdy
        idx mdx
ml1,    lac .                   / 1st control word
        sza i                   / zero if not active
        jmp mc1                 / not active
        dap . 1
        jmp .
mb1,    lac .                   / alter count of number of instructions
        add mtc
        dac mtc
mc1,    idx ml1
        sad (lac mtb 1
        jmp idl
        sas (lac mtb nob
        jmp ids
        background
        jsp blp                 / display massive star
        count mtc, .            / use up rest of time of main loop
        jmp ml0                 / repeat whole works
```

```
col,    dap cox
        law 1
        add ml1
        sad (lac mtb nob
        jmp cox-1
        dap ml2
        law 1
        add mx1
        dap mx2
        law 1
        add my1
        dap my2
        law 1
        add ma1
        dap ma2
        law 1
        add mb1
        dap mb2
ml2,    lac .                   / 2nd control word
        spq         / can it collide?
        jmp mq2                 / no
mx1,    lac .                   / calc if collision
mx2,    sub .                   / delta x
        spa                     / take abs val
        cma
        dac mt1
        sub me1                 / < EPSILON ?
        sma
        jmp mq2                 / no
my1,    lac .
my2,    sub .
        spa
        cma
        sub me1                 / < epsilon ?
        sma
        jmp mq2                 / no
        add mt1
        sub me2
        spa
        jmp cox
mq2,    idx mx2                 / end of comparison loop
        idx my2
        idx ma2
        idx mb2
        index ml2, (lac mtb nob, ml2
        idx cox
cox,    jmp .
```

```
/routine to set explosion

            law 20
            dac i mb1
            dac i mb2
sex,        lac (mex 400000      /EXPLODE
            dac i ml1            / replace calc routine with explosion
            dac i ml2
            lac i mb1            / duration of explosion
mb2,        add .
            cma
            sar 8s
            add (1
ma1,        dac .
ma2,        dac .

/ misc calculation routines

            / explosion

mex,        lac i mdx
            sar 3s
            add i mx1
            dac i mx1
            lac i mdy
            sar 3s
            add i my1
            dac i my1
            law mst
            dap msh
            lac i mb1            / time involved
            cma cli-opr
            sar 3s
            dac mxc
            sub (140
            sma
            idx msh
mz1,        lac ran
            and (777
            ior (sc1
            dac mi1
            random
            scr 9s
            sir 9s
msh,        xct .
mi1,        hlt
            add i my1
            swap
            add i mx1
            jda kcb
            xct db1
            count mxc, mz1
            count i ma , mb1
            dzm i ml1
            jmp mb1

mst,        scr 1s
            scr 3s
```

```
/ torpedo calc routine

tcr,        jsp col
            jmp sex-3
            count i ma1, tc1
            lac (mex 400000
            dac i ml1
            law i 2
            dac i ma1
            law 20
            dac i mb1
            jmp mb1

tc1,        lac i mx1
            sar 9s
            xct the
            add i mdy
            dac i mdy
            sar 3s
            add i my1
            dac i my1
            sar 9s
            xct the
            add i mdx
            dac i mdx
            sar 3s
            add i mx1
            dac i mx1
            dispt i, i my1, 1
            jmp mb1

/ hyperspace routines

/ this routine handles a non-colliding ship invisibly
/ in hyperspace

hp1,        count i ma1, mb1
            law hp3                 / next step
            dac i ml1
            law 7
            dac i mb1
            random
            scr 9s
            sir 9s
            xct hr1
            add i mx1
            dac i mx1
            swap
            add i my1
            dac i my1
            dzm i mdx
            dzm i mdy
            xct hd2
            dac i ma1
            jmp mb1
```

```
/ this routine handles a ship breaking out of
/ hyperspace.

hp3,    jsp col
        jmp sex
        count i ma1,hp6
        law 2000
        dac i mb1
        lac i mh4
        add hur
        dac i mh4
        random
        ior (400000
        add i mh4
        sma
        jmp po1
        lac i mh1
        dac i ml1
        lac ran
        scr 9s
        sir 9s
        xct hr2
        dac i mdy
        dio i mdx
        setup hpt,3
        lac ran
        sar 6s
        dac i mom
        lac ran
        dac i mth
hp4,    lac i mth
        sma
        sub (311040
        spa
        add (311040
        dac i mth
        count hpt,hp4
        count i mh2,hp7
        dzm i mh2
hp7,    xct hd3
        dac i mh3
hp6,    lac i mx1
        dispt i  i my1, 2
        jmp mb1


kcb,    0                       /relocate for center display
        dap kc1
        lac kcb
        szf 4
        jmp . 6
        sub mx1 1
        swap
        sub my1 1
        swap
        jmp kc1
        sub mx1
        swap
        sub my1
        swap
kc1,    jmp .
```

```
/ spaceship calc
ss1,        jsp i cwg
            dio scw
            jmp sr0

ss2,        jsp i cwg
            rir 4s
            dio scw

sr0,
sc1,        clf 6
            jsp col
            jmp sex
            lio scw
            clf 6 cla-opr          /update angle
            spi
            add maa
            ril 1s
            spi
            sub maa
mom,        add .
            dac i mom
            szs 10
            jmp . 3
            dzm i mom
            ral 7s
            ril 1s
            spi
            stf 6
            lio i mfu
            spi i
            clf 6

mth,        add .
            sma
            sub (311040
            spa
            add (311040
            dac i mth
            jda sin
            dac sn
            dzm bx
            dzm by
            szs 60
            jmp bsg
            lac i mx1
            dac t1
            mul t1
            scr 1s
            dac acx
            cla
            scr 2s
            dio iox
            lac i my1
            dac t1
            mul t1
            scr 1s
            dac acy
```

```
        cla
        scr 2s
        swap
        add iox
        swap
        scl 2s
        add acx
        add acy
        sub str
        sma i sza-skp
        jmp pof
        add str
        varsft
        dac t1
        jda sqt
        mul t1
        undosft
        scr 9s
        scr 8s
        sza
        jmp bsg
        scr 1s
        dio t1
        integrate mx1, bx
        integrate my1, by
bsg,    lac i mth
        jda cos
        dac cs
        sar 9s
        xct sac
        szf i 6
        cla
        add by
        diff mdy, my1, (sar 3s
        lac sn
        sar 9s
        xct sac
        cma
        szf i 6
        cla
        add bx
        diff mdx, mx1, (sar 3s
        scale sn, 5s, ssn
        scale cs, 5s, scn
        lac i mx1
        szf 4
        sub nx1
        szf i 4
        sub nx1 1
        sub ssn
        dac sx1
        sub ssn
        dac stx
```

```
            lac i my1
            szf 4
            sub ny1
            szf i 4
            sub ny1 1
            add scn
            dac sy1
            add scn
            dac sty
            scale sn, 9s, ssn
            scale cs, 9s, scn
            dac scm
            lac ssn
            dac ssm
            add scn
            dac ssc
            dac ssd
            lac ssn
            sub scn
            dac csn
            cma
            dac csm
            cla cli-opr
            jda kcb
            xct db2
mot,sp5,    jmp i .
sq6,        ioh
            llo scw
            ril 2s
            spi i                   / not blasting
            jmp sq9                 / no tail
            ranct sar 9s, sar 4s, src
            scale sn, 8s, ssn
            scale cs, 8s, scn
sq7,        count i mfu, st2
            dzm i mfu
            jmp sq9
```

```
st2,    yincr sx1, sy1, sub
        lio sy1
        xct db1
        count src,sq7
sq9,    count i ma1, sr5        / check if torp tube reloaded
        dzm i ma1               / prevent count around
mco,    lac .                   / previous control word
        cma
        szs i 30
        clc
        and scw    / present control word
        ral 3s                  / torpedo bit to bit 0
        sma
        jmp sr5                 / no launch
        count i mtr, st1        / check if torpedos exhausted
        dzm i mtr                       /prevent count around
        jmp sr5
st1,    init sr1, mtb nob-1             /search for unused object
sr1,    lac .
        sza i                   / 0 if unused
        jmp sr2
        law i 1
        add sr1
        dap sr1
        sas (lac mtb-1
        jmp sr1
        jmp sr5                 / no space for new objects

sr2,    lac (tcr                / set up torpedo calc
        dac i sr1
        law nob
        add sr1
        dap ss3
        lio stx
        swp
        szf 4
        add nx1
        szf i 4
        add nx1 1
        swp
ss3,    dio .
        add (nob
        dap ss4
        lio sty
        swp
        szf 4
        add ny1
        szf i 4
        add ny1 1
        swp
ss4,    dio .
```

```
        add (nob
        dap sr6
        add (nob
        dap sr7
        add (nob
        dap sr3
        add (nob
        dap sr4
        lac sn
        xct tvl
        cma
        add i mdx
sr3,    dac .
        lac cs
        xct tvl
        add i mdy
sr4,    dac .
        xct rlt
        dac i ma1               / permit torp tubes to cool
        xct tlf   / life of torpedo
sr6,    dac .
        lac (lac mtb nob-1
        sub sr1
        sal 3s
        add (30
sr7,    dap .                   / length of torp calc.
sr5,    lac scw
        dac i mco
        count i mh3, mb1
        dzm i mh3
        lac i mh2
        sza i
        jmp mb1
        lac scw
        and (600000
        xor (600000
        sza
        jmp mb1
        lac i ml1
        dac i mh1
        lac (hp1 400000
        dac i  ml1
        xct hd1
        dac i ma1
        law 2
        dac i mb1
        jmp mb1
```

```
/ here to handle spaceships dragged into star

/ spaceship in star

pof,        dzm i mdx
            dzm i mdy
            szs 50
            jmp po1
            lac (377777
            dac i mx1
            dac i my1
            lac i mb1
            dac ssn
            count ssn, .
            jmp mb1

po1,        lac (mex 400000      / now go bang
            dac i ml1
            lav i 10
            dac i ma1
            jmp mb1

/ outlines of spaceships

ot1,        111131
            111 11
            111111
            11 163
            311111
            146111
            111114
            700000
. 5/

ot2,        013113
            1131 1
            116313
            131111
            161151
            111633
            365114
            700000
. 5/


            variables
            constants


.-60/
p,          . 100/              / space for patches
```

```
/display score routine

fss,    0           /set size of spaceship
        dap fs1
        lac fss
        dac scm
        dac ssc
        dac ssd
        dac scn
        dac csm
        cma
        dac csn
        dzm ssn
        dzm ssm
fs1,    jmp .

fi1,    law .       /set return of compiled outline
        sub c21
        dac t6
fi2,    law .
        sub c21
        dac t7
fis,    lac c23
        dac t4
        szf 3
        lio 2sc     /get score
        szf i 3
        lio 1sc
        scl 1s
        cla
        div c12
        hlt
        sza
        jmp fx1
fkr,    dic t3
        law 400
        jda fss
        law fys
        dap frt
flt,    idx t3
        cma
        dac t3
        law fus
        dap i t6
        dap i t7
fus,    lac c20
        szf 3
        cma
        dac sy1
        lac t4
        add c20
        dac t4
        dac sx1
```

```
fds,    szf 3       /display spaceship
        law not
        szf i 3
        law not 1
        dap fug
        idx t5
        ral 9s
        cli
        xct db2
        isp t3
fug,    jmp i .
frt,    jmp .

fys,    law 4000
        jda fss
        law fub
        dap i t6
        dap i t7
        lac c26
        dac sx1
        lac c20
        szf i 3
        cma
        add c30
        dac sy1
        law i 2
        dac t3
        jmp fds
fub,    szf 3
        jmp . 3
        stf 3
        jmp . 2
        clf 3
        cli
        iot 11
        dio t1
        law 21
        xor t1
        sza i
        jmp fik
        law 42
        xor t1
        sza
        jmp fkg
        law a4+
        jmp fwt-1
fik,    law 4
        dap fiu
fwt,    add .
        dac t1
        isp t1
        jmp .-1
fiu,    jmp .
```

```
fx1,        dio t1
            dac t3
            law fx2
            dap frt
            law 1100
            jda fss
            jmp flt
fx2,        lio t1
            jmp fkr

fkg,        szf 3
            jmp fis
            szf 4
            jmp . 4
            law dj6
            stf 4
            jmp . 3
            law dj5
            clf 4
            dap . 1
            lac .
            dac db1
            idx .-2
            xct .-3
            dac db2
            jmp fis

c12,        12
c20,        200000
c21,        21
c23,        -200000-30000
c26,        -260000
c30,        30000
t3,         0
t4,         0
t5,         0
t6,         0
t7,         0

mtb,                    / table of objects and their properties


start 4
```

```
spacewar 4.4  5/17/63  ddp  : pt 1 - pass 1
spacewar 4.4  5/21/63  ddp  : pt 2 - pass 1
stars by prs  for s/w 2b - pass 1
foo
nx1=mtb nob
ny1=nx1 nob
start
foo - pass 1
spacewar 4.4  5/17/63  ddp  : pt 1 - pass 2
spacewar 4.4  5/21/63  ddp  : pt 2 - pass 2
usw      1362      a+13      count     i1
stars by prs  for s/w 2b - pass 2
spacewar 4.4 syms   5/23/63   jcm

Constants area, inclusive
from     to
2736     3063
```

1j	6077
1m	706
1q	6117
1sc	1345
2j	6121
2m	764
2q	6141
2sc	1346
3j	6143
3m	1042
3q	6403
4j	6405
4m	1120
4q	7747
a	1347
a1	1337
a2	1405
a3	1436
a4	1370
a40	1342
a5	1364
a6	1377
acx	2727
acy	2731
bc1	667
bc3	673
bcc	1176
bck	656
bcx	705
bcy	704
bds	544
bjl	530
bjm	543
bjq	537
bkc	1177
blp	464
blx	511
bpt	512
bpx	645
bsg	2275
bx	2704
by	2705
c12	3407
c21	3411
c23	3412
c26	3413
c20	3410
c30	3414
col	1534
cos	100
cox	1614
cs	2732
csm	2676
csn	2701

csx	146
cwg	2720
cwr	40
db1	1331
db2	1332
ddd	20
dj5	1333
dj6	1335
fds	3272
fi1	3225
fi2	3230
fid	24
fie	16
fik	3347
fin	7
fis	3233
fiu	3355
fkg	3367
fkr	3247
flo	55
flp	45
flt	3254
foo	12
fou	15
fpo	54
fpr	1200
frr	14
frt	3305
fs	4
fs1	3224
fss	3210
fub	3325
fug	3304
fus	3262
fuu	34
fwt	3351
fx	35
fx1	3356
fx2	3365
fyn	20
fys	3306
get	2721
grv	32
hd1	23
hd2	24
hd3	25
hp1	1744
hp3	1775
hp4	2036
hp6	2053
hp7	2051
hpt	2724
hr1	26
hr2	27

hur	30
idl	1472
ids	1504
iox	2730
kc1	2102
kcb	2060
ma1	1630
ma2	1631
maa	13
mb1	1517
mb2	1624
mco	2451
md1	1303
mdn	1307
mdx	2707
mdy	2710
me1	16
me2	17
mex	1632
mfu	2711
mg1	1460
mg2	1465
mg3	1464
mg4	1471
mh1	2713
mh2	2714
mh3	2715
mh4	2716
mhs	2
mi1	1667
ml1	1512
ml2	1556
ml0	1201
mom	2123
mot	2402
mq1	1522
mq2	1604
msh	1666
mst	1704
mt1	2722
mtb	3422
mtc	2706
mth	2137
mtr	2712
mx1	1561
mx2	1562
mxc	2723
my1	1571
my2	1572
mz1	1653
na1	3532
nb1	3562
nco	3704
ndx	3612

ndy	3642
nfu	3676
nh1	3706
nh2	3710
nh3	3712
nh4	3714
nnn	3716
nob	30
nom	3672
not	3702
ntd	2717
nth	3674
ntr	3700
nx1	3452
ny1	3502
oc	226
oc1	373
oc2	416
oc3	424
oc4	432
oc5	440
oc6	446
oc9	456
occ	2672
ocd	407
oce	400
ocg	250
och	251
oci	2673
ocj	246
ock	237
ocm	371
ocn	372
oco	262
ocp	264
ocq	263
ocr	265
ocs	222
ocx	370
ocz	225
ot1	2630
ot2	2645
p	3110
po1	2623
pof	2607
ran	31
rlt	10
sac	14
sbf	61
sc1	2111
scm	2674
scn	2663
scw	2725
sex	1620

sft	66
si1	112
si2	147
si3	117
sin	106
sn	2726
sp5	2402
sq1	220
sq2	221
sq3	171
sq6	2403
sq7	2430
sq9	2446
sqt	162
sqx	174
sr1	2467
sr2	2500
sr3	2543
sr4	2547
sr5	2561
sr6	2553
sr7	2560
src	2735
sr0	2111
ss1	2103
ss2	2106
ss3	2514
ss4	2526
ssa	2702
ssc	2700
ssd	2677
ssi	2703
ssm	2675
ssn	2662
st1	2465
st2	2434
str	15
stx	2733
sty	2734
sx1	2670
sy1	2671
t1	2665
t2	2666
t3	3415
t4	3416
t5	3417
t6	3420
t7	3421
tc1	1721
tcr	1706
the	21
tlf	11
tno	6
tvl	7

v2	3
xys	2664
xyt	2667