```
executive routine 24 october 1966

psf=iot 0077        bef=iot 0177        bff=iot 0277        rpf=iot 0377
rpn=iot 0477        rcn=iot 0577        lar=iot 0677        psn=iot 1077
ben=iot 1177        bfn=iot 1277        lpf=iot 1377        spn=iot 1477
scn=iot 1577        ad1=iot 1677        rsb=iot 2077        sbr=iot 2577
srw=iot 2677        sei=iot 2777        sps=iot 3077        sbe=iot 3177
sbf=iot 3277        sti=iot 3377        sd1=iot 3477        siw=iot 3577
sxw=iot 3677        rbe=iot 3777        rsn=iot 4077        rsf=iot 4177
srs=iot 4277        lqn=iot 4377        soq=iot 4477        la2=iot 5077
tts=iot 5177        tnr=iot 5277        lbe=iot 5577        usn=iot 5677
uf=iot 5777         ldc=iot 6077        tsn=iot 6177        tsf=iot 6277

ncb=12    /size of typewriter buffer
ewv=5     /restart level

npb=140   /punch buffer size
pwm=30    /restart level

rwm=40    /reader restart level
nuf=20    /number of user fields

3/        add .
          dap . 1
          jmp .

/buffers

b,        tsn         /initial entry
          tts
          lem
          law 5000
sut,      lia
          lar
          scn
          ben
          bff
          psf
          spn
          ben
          bff
          add (xct
          sas (5001
          jmp sut
          rsf
          usf
          cli
          lqn
          lbe
          lpf
          lar
          scn
          szs 10
          jmp par
          dzm sd+22
          lac err
          dac dc1-2
          jmp par
          constants
b 1 ncb×5 1 npb/
eb,

/service io
```

er2

```
srv,    dap sr1
        srw
srr,    skp         /skp i if reader running
        jmp sr0
        rrb
rip,    lac .
        ral 8s
        rcr 8s
        dac i rip
        rpa-i
        idx rip
        sad (lac erb
        lac (lac b
        dac rip
        lio c1
        dio rrs     /buffer not empty
        sub rop
        sza i
        dio srr     /full, shut off reader
        sma
        sub (erb-b
        sad (-rwm
        rsn         /nearly full, restart user
        srw
        xct srr
        jmp sr4
        jmp srr 2

sr0,    rpn
        sni i
        jmp sr5
        rcn
        sni
sr1,    jmp .
        jsp sat
        sps
        jmp sr2
        sti
        jmp sr3
        jsp if0+1
        psf
sr2,    tyi
        jsp itf
        jmp sr4
sr3,    jsp ite
        tyo
        jmp sr4
sr5,    jsp sat
        jsp ite
        ppa
sr4,    idx sr1
        jmp sr1

/set up console or punch buffer index

sat,    dap sa1
        cla
        rcl 4s
        add (bop-1
        dap ie1
        add (bip-bop
        dap if1
        add (bew-bip
        dap ie2
        add (bor-bew
        dap it1
        dap it2
        idx it2
sa1,    jmp .

/index and test if buffer empty

ite,    dap ie7
        lac i ie1
        dap .+2
        law 377
        and .
        lia
ie1,    idx bop
it2,    sad bor+1
it1,    lac bor
        dac i ie1
ie2,    sad bew
        bff
        sad i if1
        ben
ie7,    jmp .

/index and test if buffer full

itf,    dap if7
        lac i if1
        dap . 1
        lac .
        rcr 8s
        ral 8s
        dac i .-3
        bef
if1,    idx bip
        sad i it2
        lac i it1
        dac i if1
        sad i ie1
        bfn
        idx i ie2
        sad i it2
        lac i it1
        dac i ie2
if7,    jmp .

/clear typewriter buffer

if0,    psn
        dap if3
        bff
        lac i if1
        dac i ie1
if3,    jmp .

/buffer pointer table

bop,    z=0
        b+z       z=z+1      /1
        b+z       z=z+ncb    /2
        b+z       z=z+ncb    /3
        b+z       z=z+ncb    /4
        b+z       z=z+ncb    /5
        b+z       z=z+ncb    /6
        b+z       z=z+1      /7
        b+z       z=z+npb    /10 (punch)

bip,    z=0
```

er4

```
         b+z       z=z+1     /1
         b+z       z=z+ncb   /2
         b+z       z=z+ncb   /3
         b+z       z=z+ncb   /4
         b+z       z=z+ncb   /5
         b+z       z=z+ncb   /6
         b+z       z=z+1     /7
         b+z       z=z+npb   /10 (punch)

bew,     z=0
         b+z       z=z+1     /1
         b+z+ncb-ewv+1       zz+ncb     /2
         b+z+ncb-ewv+1       z=z+ncb    /3
         b+z+ncb-ewv+1       z=z+ncb    /4
         b+z+ncb-ewv+1       z=z+ncb    /5
         b+z+ncb-ewv+1       z=z+ncb    /6
         b+z       z=z+1     /7
         b+z+npb-pwm+1       z=z+npb    /10 (punch)

bor,     z=0
         b+z       z=z+1     /1
         b+z       z=z+ncb   /2
         b+z       z=z+ncb   /3
         b+z       z=z+ncb   /4
         b+z       z=z+ncb   /5
         b+z       z=z+ncb   /6
         b+z       z=z+1     /7
         b+z       z=z+npb   /10 (punch)
         b+z

rb,      law rb1   /rpb
         jmp . 2

ra,      law ra1   /rpa
         xct i rr0
         nop
         dap rab
         rsf
rr0,     xt .
         jmp rr8-1
         rpa-i     /set up
         law i 3
         dac r00
         lac c1
         dac i rr0
         lac c2
         dac rrs   /buffer empty
         dac srr   /reader running
         law b
         dap rip
         dap rop
         lio c2

rr8,     dio res
         jsp srv
         nop
         lio c1
         dio rs1   /clear rs1 switch
rrs,     skp i     /skp i if buffer empty
         jmp rop
         xct res   /empty
         jmp rs2   /rs1 entry
         law 20    /normal entry
         siw i
         jmp dms
         lio (jmp rr9
```

ers

```
        dio rs1
        jmp ret

rop,    lac .
        dac t
        lio c2
        idx rop
        sad (lac erb
        lac (lac b
        dac rop
        sub rip
        sza i
        dio rrs   /buffer empty
        sma
        sub (erb-b
        sad (-rwm
        dio srr   /buffer nearly empty
        lio t
rab,    jmp .     /rpa-rpb switch

ra1,    clavswp
        rcl 8s
        dio prb

        usn
res,    0
        jmp rs1 1           /rs1 entry
        siw i     /normal entry
        dio 2
ret,    law 1
        add 1
        dap 1
        spn
        scn
xe1,    xct .
        jmp xe0   /proceed trap
        jmp rm3

rb1,    spi i
        jmp rr8 1
        lac prb
        ril 2s
        rcl 6s
        dac prb
        isp r00
        jmp rr8 1
        law i 3
        dac r00
        lio prb
        jmp res-1

r00,    0         /rpb count

rr9,    lio c1
        xct rrs
        jmp rr8
rs2,    rsf
        jmp rs1 1

prb,    0         /reader buffer

rr,     xt i rr0  /rrb
        nop
        usf
        lio prb
        dio 2
```

jmp ret

```
pb,       law 2      /ppb
          rcl 6s
          jmp pp0

pa,       lac 2      /ppa
pp0,      xct .
          dac t
          spn
          jsp sat
          law 2
          sbf i
          jmp dms
          lio t
          jsp itf
          jmp ret

ti,       xct ra2
          jmp z3
          scn        /tyi
          jsp sat
z19,      law 4
          sps
          sbe i
          jmp dms
          jsp ite
          xt ra2
          jmp z10
z18,      dio 2
          jmp ret

to,       scn        /tyo
          jsp sat
z25,      law to3
          sps
          jmp if0
          law 10
          sbf i
          jmp dms
to3,      lio 2
ra2,      xt .
          jmp z50
z51,      jsp itf
          jmp ret

ix1,      dp . 4     /idx pc
          law 1
          add 1
          dap 1
          jmp .

z3,       jmp ti 2

/selectric translator

z10,      lai
          add z81
          add (z99
          dap . 2
          law 177
          and .
          sas z31
          sad z32
          jmp z11
```

```
         scr 6s
         ril 6s
         sal 6s
         sad z82
         jmp z18
         dap z82
         dio z83
         lio z32
         sza
         lio z31
         law z21
         jmp . 3

z21,     law ti 2
         lio z83
         dap z3
         jmp z18

z11,     cli
         sas z32
         lio (100
         dio z81
         jmp z19
z50,     swp          /tyo translator
         and (77
         sas z31
         sad z32
         jmp z56
z52,     add z82
         add (z99
         dap . 1
         lac .
         rcl 6s
         ral 7s
         and (100
         sad z81
         jmp z51
         dap z81
         lio z65
         sza i
         lio z66
         jsp itf
         jmp z25

z56,     cli
         sas z32
         lio (100
         dio z82
         jmp z52

z31,     74           /upper case fio-dec
z32,     72           /lower

z65,     65           /upper case selectric
z66,     66           /lower

z81,     0

z82,     0

z83,     0
/tables for selectric  (two tables in one)
/tab1=0-6   0-5=6 bit selectric key code   6=case
/tab2=11-17  12-17=6 bit concise code    11=case
/tab2(tab1(x))=x for all legal characters   space if illegal
/concise=tab2(case selectric key)for input
```

```
/selectric=tab1(case concise)for output
/if case bit changes,transmit appropriate extra case shift

z99,    070073    /space,.
        770000    /1,ill
        740071    /2,i
        750070    /3,h
        720077    /4,c.r.
        730077    /5,index=c.r.
        700036    /6,tab
        710000    /7,space
        630066    /8,f
        620067    /9,g
        070064    /ill,d
        070065    /ill,e
        070062    /ill,b
        070063    /ill,c
        070154    /ill,+
        070061    /ill,a

        760057    /0,(
        624000    //=uc9,ill
        540051    /s,r
        550050    /t,q
        520075    /ubackspace
        530000    /v,ill
        500000    /w,ill
        510000    /x,ill
        430046    /y,o
        420047    /z,p
        070044    /ill,m
        400045    /,,n
        470042    /black,k
        440043    /red,l
        060054    /tab,-
        070041    /ill,j

        004033    /center dot=colon,,
        370000    /j,ill
        340031    /k,z
        350030    /l,y
        320035    /m,red
        330000    /n,ill
        300000    /o,ill
        310034    /pblack
        230026    /q,w
        220027    /r,x
        070024    /ill,u
        070025    /ill,v
        360022    /-,s
        560023    /),t
        404055    /semicolon,)
        200173    /(,asterisk=x
        070133    /ill,=
        170000    /a,ill
        140011    /b,9
        150010    /c,8
        120000    /d,ill
        130074    /e,upper case
        100072    /f,lower case
        110000    /g,ill
        030006    /h,6
        020007    /i,7
        660004    /lower case,4
        000005    /.,5
        654002    /upper case,2
```

er 9

```
240003     /backspace,3
070020     /ill,0
040001     /c.r.,1

074040     /space,colon=center dot
754100     /"=uc3,ill
724171     /'=uc4,i
744170     /dollar sign=~,h
764177     /⊃=uc0,c.r.
634177     /uc8=∨,index=c.r.
574136     /∧=uc star,tab
734100     /<,space
704166     />,f
714167     /↑,g
074164     /ill,d
074165     /ill,e
074162     /ill,b
074163     /ill,c
074120     /ill,→

074161     /ill,a
164157     /→,[
604100     /question mark,ill
544151     /s,r
554150     /t,q
524175     /u,backspace
534100     /v,ill
504100     /w,ill
514100     /x,ill
434146     /y,o
424147     /z,p
074144     /ill,m
600145     /,n
474142     /black,k
444143     /red,l
064140     /tab,
074141     /ill,J
364056     /_,overbar=semicolon
374100     /J,ill
344131     /k,z
354130     /l,y
324135     /m,red
334100     /n,ill
304100     /o,ill
314134     /p,black
234126     /q,w
224127     /r,x
074124     /ill,u
074125     /ill,v
160122     /+,s
564123     /],t
774155     /|,]
204106     /⌊,backarrow=∧

074121     /ill,?
174100     /a,ill
144021     /b,/(uc9)=/
154105     /c,downarrow=∨
124100     /d,ill
134074     /e,upper case
104072     /f,lower case
114100     /g,ill
034110     /h,>
024111     /i,↑
660102     /lower case,'=uc4
570107     /×=star,<
```

er10

```
          654103    /upper case,~=dollar sign
          244101    /backspace,"=rev slash(uc3)
          074104    /ill,⊃=uc4=/
          04156     /c.r.|

ar,       lac 0     /arq
          spa
          cma
          dac t2
          law 177
          and i act
          dac t
          cma
          dac t1
          law cod
          dap arl
arl,      lac .
          sza i
          jmp ppq
          sad t2
          jmp arf
          law 2
          adm arl
          jmp arl
arf,      idx arl
          lac i arl
          lio 0
          spi i
          rar 9s
          and (777
          add arn
          dap . 1
          jmp .

ppq,      law flexo  q
          xor 0
          and (777770
          sza
arn,      jmp err
          law 7     /+q1 to q7
          and 0
          sza i
          jmp err
          xt ra2
          jmp no
          add (xrg
          dap ax1
ax1,      lac .
          and t1
          sza
          jmp no
          jsp dxe
          lac t
          dac i ax1
          law 7
          and 0
          rar 6s
          adm i aw1
          jmp yew

define f x,p,m
          flexo  x
[P+ERR]×1000 m-err
terminate

/arq dispatch table
```

```
cod,    f  r,pr,mr
        f  p,pp,mp
        f af,paf,maf
        f sf,pss,maf
        f 1f,p1f,m1f
        f xf,pxf,err
        f tf,ptf,err
        f  f,pff,mff
        f mb,pmb,err
        f ax,pax,err
        f sx,sax,err
        f  x,err,max
        f c1,mm,1m
        f  k,pk,err
        f  b,pt,err
        f  q,err,mq
        f nf,pnf,err
        0

        lac pfn
        rar 6s
        dac 0
yes,    siw
        jsp ix1
        jmp ret

        dac 0
no,     siw i
        jsp ix1
        jmp ret

/arq routines

err,    jmp ill

/+r
pr,     lac t1
        and rw
        sza
        jmp no
        lac t
        dac rw
        lac c2
        dac i rr0
        lac c1
        dac rs1
        jmp yes

/-r
mr,     lac err
        dac i rr0
        lac t1
        and rw
        dac rw
        sza
        jmp no
        lac c1
        dac srr
        dac rs1
        jmp no

rw      0            /reader word

/+p
pp,     lac t1
```

```
        and pw
        sza
        jmp no
        lac t
        dac pw
        lac c1
        dac i pp0
        jmp yes

/-p
mp,     lac err
        dac i pp0
        lac t1
        and pw
        dac pw
        jmp no

pw,     0               /punch word

/+k
pk,     jsp . 4
        rcr 5s

/+b
pt,     jsp . 2
        rcr 9s

        dap . 4
        law 17
        lia
        and 2
        xct .
        dac t1
        dio t
        lac i aw2
        cma
        and t
        and kbw
        sza
        jmp no
        lai
        xor i aw2
        and t1
        xor kbw
        dac kbw
        lai
        xor i aw2
        and t1
        xor i aw2
ptx,    dac i aw2
        lia
        la2
        jmp yes

kbw,    0               /knob and button word

/+nf
pnf,    law sd
        dap pn1
        dzm 0
pn1,    lac .
        and t
        sza
        idx 0
        idx pn1
        sas (lac sd+26
        jmp pn1
        lac 0
        rar 6s
        jmp no-1

/+tf
ptf,    dzm 0
        law 77
        and 2
        jda pfn
        lac i pfp
        jmp no-1
/+mb
pmb,    lio i uf1
        law 1
        sni i
        law 2
        rar 6s
        jmp no-1

/+xf
pxf,    cla
        lio 2
        rcl 6s
        jda pfn
        lio pfp
        dio t
        law 77
        and 2
        jda pfn
        lac i t
        lio i pfp
        dio i t
        dac i pfp
        jmp yes

/+1f
p1f,    jsp gaf
        jmp no
        jsp gpf
        jmp no
        dzm t2
        jmp as3

/-1f
m1f,    law i nuf
        dac t2
        lac i df1
        dac pfp
        law i 1
        adm pfp
        lac i pfp
        sza
        jmp m12
        isp t2
        jmp m1f 4
        jmp no
m12,    jda afn
        jmp dsf

/+af
paf,    cla
        jmp . 2

/+sf
pss,    lac (add
        dac t2
        cla
        jmp . 2

/-af,-sf
maf,    law 600
        dap dss
        law 77
        and 2
        sza i
        jmp af2
        jda pfn
af1,    lac pfn
        dac p
        law i 7777
        and 2
        sza i
        jmp af4
        jda afn
af3,    lac p
        jda pfn
dss,    skp
        jmp asf
        jmp dsf

af4,    law 77
        and 2
        jda pfn
        lac i pfp
        sza
        jmp af3-1
        xct dss
        jsp gaf
        jmp no
        jmp af3

af2,    law i 7777
        and 2
        sza i
        jmp yes
        jda afn
        lac i afp
        and t
        sza i
        jmp af5
        law i nuf
        add i df1
        dac pfn
apr,    lac i pfp
        sad afn
        jmp af6
        idx afp
        sas i df1
        jmp apr
        jmp .      /can't find it
af5,    xct dss
        jsp gpf
        jmp no
        jmp af1
af6,    jsp pfp 1
        jmp af1

/-q
mq,     jsp dxe
        jmp new
```

```
/-f
mff,    jsp daf
        jmp yes

/+f
pff,    law i 7777
        and 2
        sza i
        jmp yes
        ral 6s
        cma
        dac t1
        dac t2
        dac p
        jsp gaf
        jmp no
        law i 1
        adm apx
        isp t1
        jmp gaf 7
        jsp gpf
        jmp no
        law i 1
        adm ppx
        isp p
        jmp gpf 7
ffl,    jsp gaf
        jmp .
        jsp gpf
        jmp .
        law 177
        and i afp
        ior t
        dac i afp
        lac afn
        dip i pfp
        isp t2
        jmp ffl
        jmp yes

/+c1
mm,
uf1,    lac .
        sza
        jmp yes
        lac uc0
        sza i
        jmp mm0    /core 0 empty
        lac uc1
        sza i
        jmp mm1    /core 1 empty
        jsp gaf
        jmp no
        lac (lac
        dac i afp
        lac afn
        dac i uf1
yew,    siw
        jsp ix1
        jmp rew-1

mm0,    lac uf1
        dap uc0
        lac (add uc0
        jmp yew-1
mm1,    lac uf1
```

ev16

```
        dap uc1
        lac (add uc1
        jmp yew-1

/-c1
lm,     lac 1
        and (170000
        sza
        jmp ill
        lac uf1
        jda crl
new,    siw i
        jsp ix1
        jmp rew-1

/+ax
pax,    cla
        jmp . 2

/+sx
sax,    lac (add
        dac t2
        lac t1
        and xrw
        lia
        and t2
        swp
        and (177
        szaVsni
        jmp no
        law 177
        and xrw
        ior t
        ior t2
        dac xrw
        lac (add
        ior i aw2
        jmp ptx

/-x
max,    lac t1
        and xrw
        dac xrw
        lac (-add
        and i aw2
        dac i aw2
        lia
        la2
        jmp no

xrw,    0          /ext. reg. word

dxe,    dap dxx    /dismiss external eq.
        xt ra2
        jmp dxx
        lac i aw1
        ral 6s
        and (7
        add (xrg
        dap . 1
        dzm .
        lac i aw1
        and (707777
        dac i aw1
dxx,    jmp .
```

```
xrg,     repeat 7,0
         20        /level 7

daf,     dap dfx   /dismiss all fields
         law i nuf
         add i df1
         dac pfp
         dzm i pfp
         idx pfp
         sas i df1
         jmp .-3
         law sd
         dac afp
         lac t1
         and i afp
         dac i afp
         idx afp
         sas (sd 26
         jmp .-5
dfx,     jmp .

df,      lac i pfp           /dismiss field
         sas afn
         jmp no
         lac t1
         and i afp
         dac i afp
         dzm i pfp
         jmp yes

asf,     lac t1    /assign field
         and i afp
         lia
         and t2
         swp
         and (177
         szaVsni
         jmp no    /can't get it
         lac i pfp
         sza
         jmp as2
         lac t
         and i afp
         sza
         jmp no
as3,     law 177
         and i afp
         ior t
         ior t2
         dac i afp
         lac afn
         dip i pfp
         jmp yes-3

as2,     sad afn
         jmp as3
         jmp no

afp,     0         /afp→afn
         dap apx
         law i sd-1
         add afp
         rar 6s
         dip afn
apx,     jmp .
```

```
afn,    0          /afn→afp
        dap anx
        lac afn
        ral 6s
        sub (27
        sma
        jmp no
        add (sd 26
        dap afp
anx,    jmp .

pfp,    0          /pfp→pfn
        dap ppx
        lac pfp
        sub i df1
        add (nuf 1
        dac pfn
ppx,    jmp .

pfn,    0          /pfn→pfp
        dap pnx
        law i nuf 1
        add pfn
        sma
        jmp no
        add i df1
        dac pfp
pnx,    jmp .

gaf,    dap apx    /get absolute field
        law sd
        dac afp
        lac i afp
        and (-add
        sza i
        jmp ga2
        idx afp
        sas (sd 26
        jmp gaf 3
        jmp apx
ga2,    idx apx
        jmp afp 2

gpf     dap ppx    /get pseudo field
        law i nuf
        add i df1
        dac pfp
        lac i pfp
        sza i
        jmp gp2
        idx pfp
        sas i df1
        jmp gpf 4
        jmp ppx
gp2,    idx ppx
        jmp pfp 2

di,     dio i di1            /dia
        jmp ret

d,      dio t1     /dcc
        jsp trf
        dip t1
di1,    lio .
        dio t
        jsp trf
```

er19

```
        jmp dc1
        lac i df1 2
        and i act
        and (7777
        sza i
        jmp rad   /jmp ill to flush r.a.
        jmp . 2
dc1,    dip t
        lac 0
        and (170000
        sza i
        jmp dc2
        sas (i
        jmp ill
        law i 7777
        and i rf1
        sza i
        jmp ill

dc2,    dra
        xt . 2
        lai
        sub t
        and (7777
        sub (7652
        and (-77
        sza
        jmp dc3
        lio t
        dia
        lio t1
        lac 0
        dc
c2,     skp i
        jsp ix1
        soq
        jmp ret
        jmp rew

rad,    lac (lac sd 22
        xt ra2
        sas df1 1
        jmp ill
        jmp dc1 1

dc3,    jsp srv
c1,     skp
        jmp dc2

trf,    dap trx
        ril 1s
        cla
        rcl 5s
        sza i
        jmp trx
        rir 6s
        spi
        jmp abs
        sub (nuf 1
        sma
        jmp df2
df1,    add .
        dap . 1
        lac .
        sza i
        jmp ill
```

```
trx,     jmp .

df2,     xct ids
         sas (37-nuf-1
         jmp ill
id0,     lac .     /ID, field 37
         jmp trx-2

abs,     sub (27
         sma
         jmp ret   /selection error
         add (sd 26
         dap df1 2
         idx trx
         jmp trx

sd,      lac
         repeat 17,0
         lac
         lac
         lac       /0 to exclude r.a.
         lac
         lac
         0

db,      soq i     /dba
         jmp wa1
         do i di1
         lac (dba
         jmp da 2

da,      dra       /dra
         lac (dio 2
         dac t2
         dio t
         lai
         add (145
         dap t
         lio t
t2,      0
         jmp ret

start
```