

https://dspace.org/

An executive program and hardware data sources for high speed traffic simulation using a minicomputer

Yamey, Colin Godfrey

2014-07-25
http://hdl.handle.net/10539/15017

Downloaded from a DSpace repository. https://dspace.org/

```
---
PAGE 18                    ****FASP****
07/04/75
                                        ; DISPATCH TABLE

00764'000765' .DISP:  DISP
00765'027511  DISP:   "/*400+"I
00766'000776'         INIT
00767'027522          "/*400+"R
00770'001007'         RUN
00771'027503          "/*400+"C
00772'001070'         TRAP3
00773'027520          "/*400+"P          ; RE-ENTER PROGRAM
00774'000006$         COMPS
00775'000000          0                  ; END OF TABLE

00776'034002$ INIT:   LDA     3,ISTRT ; START OF INITIAL STORAGE
00777'126400          SUB     1,1
01000'030003$         LDA     2,IEND  ; END OF STORAGE
01001'175400          INC     3,3
01002'172433          SUBZ#   3,2,SNC ; FINISHED ?
01003'002403          JMP     @INITR  ; YES
01004'045777          STA     1,-1,3
01005'000774          JMP     .-4

01006'000001  INITR:  .BLK    1

01007'006024- RUN:    MES
01010'047516          .TXT    *NO OF RUNS = *
01011'047440
01012'020106
01013'052522
01014'051516
01015'036440
01016'000040
01017'006025-         JSR     @.COMD  ; GET REPLY
01020'006005-         GET
01021'040020          STA     0,20      NO OF RUNS REQUIRED
01022'040021          STA     0,21    ; COUNTER
01023'024462          LDA     1,.TR4
01024'044460          STA     1,TR3   ; INITIALISE TRAP ROUTINE
01025'044461          STA     1,TR5
01026'024455          LDA     1,TR2
01027'046046-         STA     1,@PROG
01030'024563          LDA     1,ASTK
01031'044557          STA     1,ADSTK ; INITIALISE INTERRUPT ROUTINE
01032'126400          SUB     1,1
01033'044556          STA     1,CMASK ; CLEAR CURRENT MASK
01034'063511          SKPBZ   TTO     ; WAIT FOR TTO TO FINISH
01035'000777          JMP     .-1
01036'060211          NIOC    TTO     ; CLEAR
01037'024033-         LDA     1,.TBUF ; INITIALISE TTOUT
01040'044044-         STA     1,TOPT
01041'044045-         STA     1,TIPT
01042'126400          SUB     1,1
01043'044041-         STA     1,ACT
01044'062677          IORST           ; RESET THE WORLD
01045'060177          INTEN
01046'002046-         JMP     @PROG   ; JUMP TO PROGRAM

01047'000057  SLASH:  "/
```

```
---
PAGE  19                        ****FASP****
07/04/75
 1                      ; SUBROUTINE TRAP
 2                      ; TRAPS PROGRAM AT BEGINNING OF NEXT RUN
 3                      ; SHOULD BE USED BY INTERRUPT SERVICE ROUTINES
 4                      ; IF NON REENTRANT PSEUDO FUNCTIONS ARE TO BE USED
 5                      ; TRAP DISMISSES THE INTERRUPT
 6
 7                      ; CALLING SEQUENCE:
 8                      ;       TRAP
 9                      ; RETURNS HERE AT BEGINNING OF NEXT RUN
10                      ; WITH INTERRUPTS OFF
11
12                      ; AFTER SERVICING , IR MUST RETURN WITH A
13                      ;       JMP     0 ,
14                      ; TO REENABLE INTERRUPTS
15
16  01050'056434  TRAP1:  STA     3,@TR3  ; SAVE RETURN
17  01051'010433          ISZ     TR3     ; BUMP STACK POINTER
18  01052'020430          LDA     0,TR1   ; PUT IN JUMP
19  01053'042046-         STA     0,@PROG
20  01054'000504          JMP     DISMIS  ; DISMISS INTERRUPT
21
22  01055'020424  TRAP2:  LDA     0,TMSK
23  01056'062077          MSKO    0       ; ALL OUT EXCEPT TTO
24  01057'020424          LDA     0,TR2   ; RESTORE PROG
25  01060'042046-         STA     0,@PROG
26  01061'^20041-         LDA     0,ACT   ; WAIT FOR BUFFER TO EMPTY
27  01062'101004          MOV     0,0,SZR
28  01063'000776          JMP     .-2
29  01064'060277          INTDS
30  01065'036421          LDA     3,@TR5
31  01066'010420          ISZ     TR5
32  01067'005400          JSR     0,3     ; JUMP TO USER ROUTI
33  01070'020414  TRAP3:  LDA     0,TR3
34  01071'024415          LDA     1,TR5
35  01072'122414          SUB#    1,0,SZR ; FURTHER ROUTINES ?
36  01073'000765          JMP     TRAP2+4 ; YES
37  01074'020411          LDA     0,.TR4
38  01075'040407          STA     0,TR3
39  01076'040410          STA     0,TR5
40  01077'060177          INTEN
41  01100'002046-         JMP     @PROG   ; RETURN TO PROG
42
43  01101'177776  TMSK:   177776
44  01102'032032- TR1:    JMP     @.TR2
45  01103'000001  TR2:    .BLK    1
46  01104'001107' TR3:    TR4
47  01105'001107' .TR4:   TR4
48  01106'001107' TR5:    TR4
49  01107'000012  TR4:    .BLK    10.     ; STACK FOR RETURNS
```

```
---
PAGE  20                    ****FASP****
07/04/75
 1.                           ; MASTER INTERRUPT ROUTINE
 2
 3                            ; LAYOUT OF STACK ENTRY
 4
 5
 6          000000   SAC3=0   ; SAVE FOR AC3
 7          000001   SAC0=1   ; SAVE FOR AC0
 8          000002   SAC1=2   ; SAVE FOR AC1
 9          000003   SAC2=3   ; SAVE FOR AC2
10          000004   SCRY=4   ; SAVE FOR CARRY
11          000005   SRTN=5   ; SAVE FOR RETURN
12          000006   SMSK=6   ; SAVE FOR CURRENT MASK
13
14  01121'056467   ISR:     STA     3,@ADSTK ; SAVE AC3
15  01122'034466             LDA     3,ADSTK ; ADDRESS OF STACK
16  01123'041401             STA     0,SAC0,3; SAVE AC0
17  01124'045402             STA     1,SAC1,3
18  01125'051403             STA     2,SAC2,3
19  01126'102560             SUBCL   0,0     ; SAVE CARRY
20  01127'041404             STA     0,SCRY,3
21  01130'020000             LDA     0,0     ; SAVE RETURN
22  01131'041405             STA     0,SRTN,3
23  01132'020457             LDA     0,CMASK ; SAVE CURRENT MASK
24  01133'041406             STA     0,SMSK,3
25  01134'030456             LDA     2,SIZE  ; PUSH STACK
26  01135'157000             ADD     2,3
27  01136'054452             STA     3,ADSTK
28  01137'061477             INTA    0       ; GET DEVICE CODE
29  01140'063777   ISR1:    SKPDZ   CPU     ; IS IT POWER FAIL ?
30  01141'020031             LDA     0,C77   ; YES
31  01142'063714             SKPDZ   RTC     ; IS REAL TIME CLOCK ?
32  01143'020451             LDA     0,C14   ; YES
33  01144'030037-            LDA     2,AMTAB ; ADD OF MASK TABLE -1
34  01145'113000             ADD     0,2     ; ADD OF MASK
35  01146'031000             LDA     2,0,2   ; MASK
36  01147'024443             LDA     1,CMASK ; INCLUSIVE OR NEW
37  01150'124000             COM     1,1     ; MASK WITH OLD
38  01151'133400             AND     1,2
39  01152'132000             ADC     1,2
40  01153'050436             STA     2,CMASK
41  01154'034036-            LDA     3,AJTAB ; ADDOF JUMP TABLE-1
42  01155'117000             ADD     0,3     ; ADD OF SERVICE ROUTINE
43  01156'072177             DOBS    2,CPU   ; MASK AND ENABLE INTERRUPTS
44  01157'007400             JSR     @0,3    ; JUMP TO ROUTINE
45  01160'060277   DISMIS: INTDS
46  01161'034427             LDA     3,ADSTK ; POP STACK
47  01162'030430             LDA     2,SIZE
48  01163'156400             SUB     2,3
49  01164'031406             LDA     2,SMSK,3        ; ISSUE OLD MASK
50  01165'072077             MSKO    2
51  01166'054422             STA     3,ADSTK ; UPDATE POINTER
52  01167'050433             STA     2,CMASK ; UPDATE MASK
53  01170'021404             LDA     0,SCRY,3 ; RESTORE CARRY
54  01171'101210             MOVZR   0,0
55  01172'021405             LDA     0,SRTN,3 ; RESTORE RETURN
56  01173'040000             STA     0,0
57  01174'021401             LDA     0,SAC0,3 ; RESTORE AC0
58  01175'025402             LDA     1,SAC1,3
```

```
---
PAGE  21                    ****FASP****
07/04/75
 1  01176'031403            LDA     2,SAC2,3
 2  01177'036411            LDA     3,@ADSTK
 3  01200'060177            INTEN
 4  01201'002000            JMP     @0          ; RETURN
 5
 6                                   ; ROUTINE TO IGNORE INTERRUPTS
 7  01202'024405  IGNOR:    LDA     1,CLEAR
 8  01203'123000            ADD     1,0         ; MAKE UP CLEAR
 9  01204'040401            STA     0,.+1
10  01205'000000            0
11  01206'001400            JMP     0,3
12  01207'060300  CLEAR:    NIOC    0
13
14  01210'001215' ADSTK:    STACK
15  01211'000001  CMASK:    .BLK    1
16  01212'000007  SIZE:     7
17  01213'001215' ASTK:     STACK
18  01214'000014  C14:      14
19  01215'000106  STACK:    .BLK    10 */       ; RETURN AND SAVE STACK
20  01323'000077  MTAB:     .BLK    77          ; MASK TABLE
21  01422'000077  JTAB:     .BLK    77          ; JUMP TABLE
22
23        001521'           BUS=.
```

```
---
PAGE  22                    ****FASP****
07/04/75
 1                            ; TELETYPE IN INTERRUPT SERVICE
 2                            ; TRANSFERS CONTROL TO ERET ON A
 3                            ; ESCAPE COMMAND
 4                            ; ALL OTHER CHARS ARE IGNORED
 5
 6
 7  01521'000010              10        ; DEVICE CODE
 8  01522'000002              1B14      ; MASK BIT 14
 9  01523'001555'             NEXT1     ; ADDRESS OF NEXT ROUTINE
10
11  01524'060610  TTIN:       DIAC      0,TTI
12  01525'024034              LDA       1,C177  ; STRIP PARITY
13  01526'123400              AND       1,0
14  01527'024425              LDA       1,ESC
15  01530'106414              SUBZ#     0,1,SZR ; IS IT ESCAPE
16  01531'001400              JMP       0,3     ; NO DISMISS
17  01532'006031-             TRAP
18  01533'006023-             CRLF
19  01534'006024-             MES
20  01535'052123              .TXT      *STOP AFTER*
21  01536'052117
22  01537'040440
23  01540'052105
24  01541'051105
25  01542'000000
26  01543'020006              LDA       0,6     ; NO OF RUNS
27  01544'006006-             PUT
28  01545'000005              5                 ; NO OF DIGITS
29  01546'006024-             MES
30  01547'051040              .TXT      * RUNS*
31  01550'047125
32  01551'000133
33  01552'006023-             CRLF
34  01553'002003-             JMP       @ERET
35
36  01554'000033  ESC:        33
37        001555'             NEXT1=.
38
```

```
---
PAGE  23                    ****FASP****
07/04/75
 1                  ; DISK INTERRUPT SERVICE ROUTINE
 2
 3  01555'000020            20                  ; DEVICE CODE
 4  01556'177777            -1                  ; MASK ALL
 5  01557'001564'           NEXTD               ; ADD OF NEXT ROUTINE
 6
 7  01560'034004$ DIN:      LDA     3,DDOS
 8  01561'007414            JSR     @D.INT,3
 9  01562'000401            JMP     .+1
10  01563'002035-           JMP     @INTR
11
12        001564'           NEXTD = .
13
14
15                  ; TELETYPE OUT SERVICE ROUTINE
16
17  01564'000011            11                  ; DEVICE CODE
18  01565'000001            1B15                ; MASK
19  01566'001777'           TEND
20
21  01567'020041- TTOUT:    LDA     0,ACT
22  01570'101004            MOV     0,0,SZR ; ACTIVE ?
23  01571'000421            JMP     .+3     ; YES
24  01572'060211            NIOC    TTO     ; CLEAR
25  01573'001420            JMP     0,3     ; RETURN
26  01574'030044-           LDA     2,TOPT
27  01575'021000            LDA     0,0,2   ; GET CHAR
28  01576'051111            DOAS    0,TTO
29  01577'024045-           LDA     1,TIPT
30  01600'004410            JSR     BUMP    ; EMPTY
31  01601'000401            JMP     .+3     ; YES
32  01602'050044-           STA     2,TOPT  ; BUMP POINTER
33  01603'002035-           JMP     @INTR   ; RETURN
34  01604'050044-           STA     2,TOPT
35  01605'102400            SUB     0,0
36  01606'040041-           STA     0,ACT
37  01607'002035-           JMP     @INTR
38
39
```

```
---
PAGE  24                    ****PASP****
07/04/75
 1                  ; TELETYPE CHARACTER OUTPUT ROUTINE
 2                  ; OUTPUT IS BUFFERED IN A 80 CHARACTER RECIRCULATING BUFFER
 3
 4  01610'020034-  BUMP:   LDA     0,.TEND ; END OF BUFFER
 5  01611'151400           INC     2,2
 6  01612'112415           SUB#    0,2,SNR ; END OF BUFFER ?
 7  01613'030033-          LDA     2,.TBUF ; YES
 8  01614'146414           SUB#    2,1,SZR ; EQUAL ?
 9  01615'001401           JMP     1,3     ; NO
10  01616'001400           JMP     0,3
11
12  01617'0635?7   PUTC1:  SKPBZ   CPU     ; INTERRUPT ON ?
13  01620'000405           JMP     OPUT    ; YES
14  01621'063511           SKPBZ   TTO
15  01622'000777           JMP     .-1
16  01623'061111           DOAS    0,TTO
17  01624'001400           JMP     0,3
18
19  01625'054426   OPUT:   STA     3,TRET
20  01626'040426           STA     0,TRET+1
21  01627'044426           STA     1,TRET+2
22  01630'050426           STA     2,TRET+3
23  01631'024041-          LDA     1,ACT
24  01632'125004           MOV     1,1,SZR ; ACTIVE ?
25  01633'000404           JMP     .+4
26  01634'060277           INTDS
27  01635'063111           DOAS    1,TTO   ; START TTO
28  01636'010041-          ISZ     ACT     ; NO, SET ACTIVE
29  01637'030045-          LDA     2,TIPT
30  01640'041000           STA     0,0,2   ; STORE CHAR
31  01641'060177           INTEN
32  01642'030045-          LDA     2,TIPT
33  01643'024044-          LDA     1,TOPT
34  01644'004744           JSR     BUMP    ; BUFFER FULL ?
35  01645'000775           JMP     .-3     ; YES WAIT FOR SPACE
36  01646'050045-          STA     2,TIPT  ; BUMP POINTER
37  01647'020405           LDA     0,TRET+1
38  01650'024405           LDA     1,TRET+2
39  01651'030405           LDA     2,TRET+3
40  01652'002401           JMP     @TRET   ; RETURN
41
42  01653'000004   TRET:   .BLK    4
43  01657'000120   TBUF:   .BLK    80.
44        001777'          TEND = .
45
46                         .END
```

```
---
PAGE   1
06/16/75
 1                          .TITL   FASPEND
 2                    ; COMPILER FOR FASP
 3                    ; MUST BE LOADED AFTER THE LAST USER ROUTINE
 4                    ; FASPEND PERFORMS ALL INITIALISATION ON FIRST START
 5                    ; AND COMPILES ENTERED PROGRAMS
 6                    ; IT ALSO CONTRAINS THE DISK DRIVERS AND SYSTEM ROUTINES
 7
 8                    ; REVISION HISTORY:
 9                    ;       8/6/75
10
11                    ;                               C. YANEY
12                    ;                               UNIVERSITY OF THE WITWATERSRAND
13                    ;                               JOHANNESBURG
14
15                          .ENT  INIT1,USEND,ISTRT,IEND,PUTP,COMP3,COMP5,DOOS
16                          .ENT  DWRITE
17                          .EXTN IGNORE,RETURN,NONE,INIT,TR2,PUTC
18                          .EXTD AJTAB,AMTAB,C77,.BUS,OPT,IPT,ACT,PROG
19                          .EXTD .RET,ERET,.REC,.COMD,C3,TEMP2,TEMP1,SP
20                          .EXTD DFLG,.READ,C177,.GETC,.PUTC,.WRITE
21                          .EXTN ERROR,NES,CRLF,INITR,GET,MULTIPLY,GETAD
22                          .EXTN GETBR,GETAS
23
24                          .ZREL
25
26  00000-001160' USEND:    INIT1               ; END OF USER ROUTINES
27  00001-000121' .NAME:    NAME
28  00002-000461' .DREAD:   DREAD
29  00003-000001  ISTRT:    .BLK    1           ; START OF INITIAL STORAGE
30  00004-000001  IEND:     .BLK    1           ; END OF INITIAL STORAGE
31  00005-000001  0005:     .BLK    1
32  00006-001607' .PUTP:    PUTP1
33
34        006006-           PUTP = JSR  @.PUTP
35
36                          .NREL
```

```
---
PAGE   2
06/16/75
 1                    ; DISK DRIVERS FOR DOOS 1.6 VER 2
 2
 3                    ; SAVE ROUTINE
 4                    ; CALLING SEQUENCE (IN PROG OR DATA MODE)
 5                    ;       SAVE,FILENAME[,]
 6                    ; IF THE SECOND COMMA IS PRESENT SUBSEQUENT DATA WILL
 7                    ; BE APPENDED TO THE FILE
 8                    ; DATA CANNOT BE WRITTEN TO AN EXISTING FILE
 9                    ; ERROR 11 : FILE EXISTS
10                    ; ERROR 12 : FILE DOES NOT EXIST
11
12  00000'000011'             SAVE            ; CONST ROUTINE
13  00001'000000              0               ; NO STORAGE
14  00002'000000              0
15  00003'000000              0
16  00004'000000              0
17  00005'000542'             NEXT1           ; ADD OF NEXT ROUTINE
18        000001              .TXTM   1
19  00006'051501              .TXT    *SAVE*
20  00007'053105
21  00010'000000
22
23  00011'020001- SAVE:       LDA     0,.NAME ; ADD OF NAME
24  00012'177777              GETAS           ; GET NAME
25  00013'102400              SUB     0,0
26  00014'040504              STA     0,FLAGA ; APPEND FLAG
27  00015'177777              GETBR           ; CHECK BREAK
28  00016'175004              MOV     3,3,SZR ; APPEND ?
29  00017'000403              JMP     .+3     ; NO
30  00020'010500              ISZ     FLAGA
31  00021'177777              GET             ; DUMMY COMMAND
32  00022'030001-             LDA     2,.NAME
33  00023'034005-             LDA     3,DDOS
34  00024'007402              JSR     @D.SRC,3; FILE EXISTS ?
35  00025'000414              JMP     CREATE  ; NO
36  00026'020472              LDA     0,FLAGA
37  00027'101004              MOV     0,0,SZR ; APPEND ?
38  00030'000403              JMP     .+3     ; YES
39  00031'177777              ERROR
40  00032'000013              11.             ; FILE EXISTS
41  00033'024462              LDA     1,.BUFA ; BUFFER ADD
42  00034'102520              SUBZL   0,0     ; LAST SECTOR
43  00035'030001-             LDA     2,.NAME
44  00036'007400              JSR     @D.CYS,3
45  00037'000007              0.OPN
46  00040'000420              JMP     WRITE
47
48  00041'020457  CREATE:     LDA     0,FLAGA
49  00042'101005              MOV     0,0,SNR ; APPEND ?
50  00043'000401              JMP     .+3     ; NO
51  00044'000031'             ERROR
52  00045'000014              12.             ; FILE DOES NOT EXIST
53  00046'102400              SUB     0,0     ; NO PROPS
54  00047'105520              INCZL   0,1     ; TYPE ASCII
55  00050'030001-             LDA     2,.NAME
56  00051'007400              JSR     @D.CYS,3
57  00052'000011              0.CRT           ; CREATE FILE
58  00053'024442              LDA     1,.BUFA
```

```
---
PAGE   3
06/16/75
 1   00054'102400           SUB     0,0
 2   00055'030001-          LDA     2,.NAME
 3   00056'007400           JSR     @D.CYS,3
 4   00057'000007           0.OPN
 5
 6   00060'177777  WRITE:   CRLF
 7   00061'010021$          ISZ     DFLG    ; ENABLE WRITE
 8   00062'177777           RETURN
 9
10   00063'054434  DWRIT:   STA     3,DRET
11   00064'040432           STA     0,DSAV
12   00065'101100           MOVS    0,0
13   00066'030023$          LDA     2,C177
14   00067'143400           AND     2,0
15   00070'030424           LDA     2,SLASH
16   00071'142415           SUB#    2,0,SNR ; IS IT A COMMAND
17   00072'000410           JMP     TERM    ; YES STOP WRITING
18   00073'121000           MOV     1,0     ; NO OF BYTES
19   00074'024005$          LDA     1,OPT   ; ADDRESS OF BYTES
20   00075'030420           LDA     2,.BUFA
21   00076'034005-          LDA     3,DDOS
22   00077'007407           JSR     @D.MOT,3
23   00100'020416           LDA     0,DSAV
24   00101'002416           JMP     @DRET
25
26   00102'034003- TERM:    LDA     3,DDOS
27   00103'030412           LDA     2,.BUFA
28   00104'007400           JSR     @D.CYS,3
29   00105'000013           0.EOF           ; END FILE
30   00106'102400           SUB     0,0
31   00107'040021$          STA     0,DFLG  ; CLEAR FLAG
32   00110'007400           JSR     @D.CYS,3
33   00111'000010           0.CLO
34   00112'020404           LDA     0,DSAV
35   00113'002404           JMP     @DRET
36
37   00114'000057  SLASH:   "/
38   00115'000132' .BUFA:   BUFA
39
40   00116'000001  DSAV:    .BLK    1
41   00117'000001  DRET:    .BLK    1
42   00120'000001  FLAGA:   .BLK    1
43   00121'000011  NAME:    .BLK    11      ; STORAGE FOR NAME
44   00132'000410  BUFA:    .BLK    264.    ; WRITE BUFFER
45         000542'          NEXT1 = .
```

```
---
PAGE   4
06/16/75
 1                      ; GET ROUTINE
 2                      ; CALLING SEQUENCE :
 3                      ; GET FILENAME[,]
 4                      ; IF THE "," IS PRESENT CONSOLE ECHOING WILL BESUPPRESSED
 5                      ; THE FILE IS READ BYTE BY BYTE TILL EOF
 6
 7  00542'000552'               DGET
 8  00543'000000                0
 9  00544'000000                0
10  00545'000000                0
11  00546'000000                0
12  00547'001241'               NEXT2
13  00550'043505                .TXT    *GET*
14  00551'052000
15
16  00552'020001- DGET:         LDA     0,.NAME
17  00553'.00012'               GETAS           ; GET NAME
18  00554'030001-               LDA     2,.NAME
19  00555'034005-               LDA     3,DDOS
20  00556'007403                JSR     @D.SRC,3
21  00557'000402                JMP     .+2     ; FILE DOES NOT EXIST
22  00560'000403                JMP     .+3
23  00561'000044'               ERROR
24  00562'000014                12.
25  00563'000060'               CRLF
26  00564'000015'               GETBR           ; GET BREAK
27  00565'175004                MOV     3,3,SZR ; IS IT "," ?
28  00566'000404                JMP     .+4     ; NO
29  00567'020440                LDA     0,DG1   ; SUPPRESS PRINT
30  00570'040025$               STA     0,.WRITE
31  00571'000021'               GET             ; DUMMY
32  00572'020003-               LDA     0,.DREAD
33  00573'040022$               STA     0,.READ
34  00574'034005-               LDA     3,DDOS
35  00575'024423                LDA     1,.BUFB
36  00576'102400                SUB     0,0
37  00577'030001-               LDA     2,.NAME
38  00600'007400                JSR     @D.CYS,3
39  00601'000007                D.OPN
40  00602'000063'               RETURN
41
42  00603'054423  DREAD:        STA     3,DRET1
43  00604'034005-               LDA     3,DDOS
44  00605'030423                LDA     2,.BUFB
45  00606'007404                JSR     @D.IN,3 ; READ BYTE
46  00607'000404                JMP     .+4     ; END OF FILE
47  00610'125220                MOVZR   1,1     ; 377>177
48  00611'123400                AND     1,0
49  00612'002414                JMP     @DRET1
50  00613'034005-               LDA     3,DDOS
51  00614'007400                JSR     @D.CYS,3
52  00615'000010                D.CLO           ; CLOSE FILE
53  00616'020025$               LDA     0,.PUTC ; RESTORE TELETYPE
54  00617'040026$               STA     0,.WRITE
55  00620'020024$               LDA     0,.GETC
56  00621'040022$               STA     0,.READ
57  00622'000563'               CRLF
58  00623'102400                SUB     0,0
```

```
---
PAGE   6
06/16/75
 1                    ; SUBROUTINES FOR CREATING THE LOOP IN
 2                    ; THE USER AREA
 3                    ; THESE ROUTINES HAVE THE SAME FORMAT AS
 4                    ; USER ROUTINES
 5
 6
 7                    ; INITIAL SUBROUTINE
 8  01301'177777            NONE              ; NO VARIABLES
 9  01302'000000            0
10  01303'000000            0
11  01304'000000            0
12  01305'000000            0
13  01306'001314'           NEXTI             ; NEXT ROUTINE ADDRESS
14        000001            .TXTM   1                 ; PACKING L-R
15  01307'044516            .TXT    *INITIAL*
16  01310'044524
17  01311'044501
18  01312'046000
19  01313'002013$           JMP     @ERET     ; IGNORE COMMAND "INITIAL"
20        001314'           NEXTI=.
21
22                   ; DYNAMIC SUBROUTINE
23                   ; SETS UP START OF LOOP
24  01314'001301'           NONE
25  01315'000000            0
26  01316'000000            0
27  01317'000000            0
28  01320'000000            0
29  01321'001341'           NEXTD
30  01322'042101            .TXT    *DYNAMIC*
31  01323'047101
32  01324'046511
33  01325'041400
34  01326'024020            LDA     1,20.0    ; PROG LINE COUNTER
35  01327'125400            INC     1,1
36  01330'125400            INC     1,1
37  01331'044010$           STA     1,PROG    ; PROG START ADDRESS
38  01332'006006-           PUTP
39  01333'001340'           DN1
40  01334'002013$           JMP     @.REC
41  01335'020020            LDA     0,20      ; NO OF RUNS
42  01336'024021            LDA     1,21      ; COUNTER
43  01337'122400            SUB     1,0
44  01340'040006  DN1:      STA     0,E       ; EPOCH COUNTER
45        001341'           NEXTD=.
46
47                   ; TERMINATION SUBROUTINE
48                   ; SETS UP END OF LOOP
49
50  01341'001314'           NONE
51  01342'000000            0
52  01343'000000            0
53  01344'000000            0
54  01345'000000            0
55  01346'001360'           NEXTT
56  01347'052103            .TXT    *TERMINAL*
57  01350'051115
58  01351'044516
```

```
---
PAGE   7
06/15/75
 1   01352'040514
 2   01353'000000
 3   01354'006006-          PUTP
 4   01355'001357'          T1
 5   01356'014021           DSZ      21
 6   01357'002010$ T1:      JMP      @PROG
 7         001360'          NEXTT=.
```

```
---
PAGE   8
06/16/75
 1
 2  01360'054005- INIT1:  STA     3,DDOS
 3  01361'152400          SUB     2,2     ; SIZE CORE
 4  01362'020430          LDA     0,C4K
 5  01363'113000  S1:     ADD     0,2
 6  01364'035000          LDA     3,0,2   ; SAVE CONTENTS
 7  01365'051000          STA     2,0,2
 8  01366'025000          LDA     1,0,2
 9  01367'055000          STA     3,0,2   ; RESTORE CONTENTS
10  01370'146415          SUB#    2,1,SNR
11  01371'000772          JMP     S1
12  01372'020417          LDA     0,C200  ; SAVE DDOS
13  01373'112400          SUB     0,2
14  01374'050547          STA     2,CMAX  ; HIGHEST USEABLE LOCATION
15
16                                        ; INITIALISE INTERRUPT ROUTINE
17  01375'030002$         LDA     2,AMTAB ; START OF MP"" TABLE
18  01376'176000          ADC     3,3
19  01377'020411          LDA     0,.IGNOR
20  01400'024003$         LDA     1,C77   ; LENGTH OF TABLE
21  01401'147000          ADD     2,1     ; AC1 =END OF TABLE
22  01402'151400          INC     2,2
23  01403'041077          STA     0,77,2  ; STORE ADD OF IGNORE ROUTINE
24                                        ; IN TABLE
25  01404'055000          STA     3,0,2   ; ZERO MASK
26  01405'132414          SUBZ#   1,2,SZR ; LOOP THROUGH TABLE
27  01406'000774          JMP     .-4
28  01407'000404          JMP     INTLP-1
29
30  01410'177777  .IGNOR: IGNOR
31  01411'003720  C200:   2000.
32  01412'004000  C4K:    4000
33
34                                        ; SNAKE THRU USER ROUTINES
35                                        ; AND CREATE INTERRUPT TABLES
36
37  01413'034004$         LDA     3,.BUS  ; START OF USER ROUTINES
38  01414'025400  INTLP:  LDA     1,0,3   ; FIRST WORD OF USER PROG
39  01415'020003$         LDA     0,C77
40  01416'106432          SUBZ#   0,1,SZC ; LESS THAN 77 ?
41  01417'000414          JMP     SBRT    ; NO, MUST BE SUBROUTINE
42  01420'030001$         LDA     2,AJTAB ; INTERRUPT SERVICE  ROUTINE
43  01421'113000          ADD     1,2
44  01422'020015$         LDA     0,C3    ; GET ADDRESS OF ROUTINE
45  01423'163000          ADD     3,0
46  01424'041000          STA     0,0,2   ; STORE IN DISPATCH TABLE
47  01425'030021$         LDA     2,AMTAB ; START OF MASK TABLE
48  01426'133000          ADD     1,2
49  01427'021401          LDA     0,1,3   ; GET M
50  01430'041000          STA     0,0,2   ; STORE MASK
51  01431'013401          LDA     3,2,3   ; ADD OF NEXT ROUTINE
52  01432'000762          JMP     INTLP   ; CONTINUE FOR ALL ROUTINES
53
```

```
---
PAGE   9
06/16/75
 1                                          ;ALLOCATE CONSTANT STORAGE
 2
 3  01433'054004$ SBRT:   STA     3,.BUS  ; FIRST USER SUBROUTINE
 4  01434'020576          LDA     0,.STORC
 5  01435'025401  SB1:    LDA     1,1,3   ; BLOCKS REQUIRED
 6  01436'041401          STA     0,1,3   ; START OF STORAGE
 7  01437'054017$         STA     3,TEMP3 ; SAVE AC3
 8  01440'031402          LDA     2,2,3   ; NO OF LOCS REQUIRED
 9  01441'177777          MULTIPLY
10  01442'034017$         LDA     3,TEMP3
11  01443'030500          LDA     2,CMAX
12  01444'142433          SUBZ#   2,0,SNC ; MEM OVERFLOW ?
13  01445'000403          JMP     .+3     ; NO
14  01446'001261'         ERROR
15  01447'100001          @1              ; FATAL, YOU LOOSE
16  01450'035405          LDA     3,5,3   ; ADD OF NEXT ROUTINE
17  01451'024000-         LDA     1,USEND
18  01452'136433          SUBZ#   1,3,SNC ; FINISHED ?
19  01453'000762          JMP     SB1     ; NO
20
21  01454'040705  SB2:    STA     0,INIT1+1       ; START OF INITIAL STORAGE
22  01455'040003-         STA     0,ISTRT
23  01456'034004$         LDA     3,.BUS  ; START OF ROUTINES
24  01457'025403  SB3:    LDA     1,3,3   ; GET STORAGE REQUIRED
25  01460'041403          STA     0,3,3   ; STORE START OF STORAGE
26  01461'054017$         STA     3 TEMP3
27  01462'031404          LDA     2 4,3   ; LOCATIONS REQUIRED
28  01463'001441'         MULTIPLY
29  01464'034017$         LDA     3,TEMP3
30  01465'030456          LDA     2,CMAX
31  01466'142433          SUBZ#   2,0,SNC ; MEM OVERFLOW ?
32  01467'000403          JMP     .+3     ; NO
33  01470'001445'         ERROR
34  01471'100001          @1              ; FATAL
35  01472'035405          LDA     3,5,3   ; ADD OF NEXT ROUTINE
36  01473'024000-         LDA     1,USEND
37  01474'136433          SUBZ#   1,3,SNC ; FINISHED ?
38  01475'000762          JMP     SB3
39  01476'040004-         STA     0,IEND  ; END OF INITIAL STORAGE
40  01477'040664          STA     0,INIT1+3
41  01500'042442          STA     0,@INS  ; START OF INITIALISATION
42  01501'020476          LDA     0,.COMS ; COMPILE START ADD
43  01502'040004          STA     0,4
44
45  01503'022437  COMPS:  LDA     0,@INS  ; COMPILLER STARTS HERE
46  01504'100400          NEG     0,0
47  01505'100000          COM     0,0
48  01506'040020          STA     0,20,0  ; PROG LINE COUNTER
49  01507'020467          LDA     0,.COMP3
50  01510'040012$         STA     0,ERET  ; ERROR RETURN ADD
51  01511'020470          LDA     0,.PUP1
52  01512'040006-         STA     0,.PUTP ; RESTORE PUTP SUBROUTINE
53
54  01513'000621'         CRLF
55  01514'177777          MES
56        000000          .TXTM 0
57  01515'047105          .TXT    *ENTER PROGRAM*
58  01516'042524
```

```
---
PAGE  10
06/16/75
 1  01517'020122
 2  01520'051120
 3  01521'043517
 4  01522'040522
 5  01523'000115
 6  01524'001513'          CRLF
 7
 8  01525'020020  COMP1:   LDA     0,20     ; SAVE PROG COUNTER
 9  01526'040445           STA     0,CP4
10  01527'020446           LDA     0,INVC
11  01530'177777           PUTC
12  01531'020020$          LDA     0,SP
13  01532'001530'          PUTC
14  01533'006014$          JSR     @COMD    ; GET PROGRAM LINE
15  01534'024405           LDA     1,ER
16  01535'122415           SUBZ#   1,0,SNR ; END OF PROG ?
17  01536'000406           JMP     COMP2    ; YES
18  01537'177777           GETAD            ; SEARCH FOR USER PROG
19  01540'001000           JMP     0,2      ; JMP TO USER ROUTINE
20
21  01541'027505  ER:      "/*400+"E        ; /E
22  01542'177777  INS:     INITR
23  01543'000001  CMAX:    .BLK    1
24
25  01544'020013$ COMP2:   LDA     0,.XEC
26  01545'101400           INC     0,0
27  01546'040013$          STA     0,ERET   ; CHANGE RETURN ADDRESS
28  01547'020414           LDA     2,.CP3   ; PUT RETURN IN USER AREA
29  01550'021000           LDA     0,0,2
30  01551'101005           MOV     0,0,SNR
31  01552'000404           JMP     .+4
32  01553'042020           STA     0,@20,0
33  01554'151400           INC     2,2
34  01555'000773           JMP     .-5
35  01556'020421           LDA     0,.ER10
36  01557'040006-          STA     0,.PUTP  ; ILLEGAL AFTER PROGRAM IS ENTERED
37  01560'022010$          LDA     0,@PROG  ; INITIALISE TRAP
38  01561'042413           STA     0,@CP5
39  01562'002402           JMP     @CP2     ; START AT INITIALISATION
40
41  01563'001565' .CP3:    CP3
42  01564'177777  CP2:     INIT
43  01565'020007$ CP3:     LDA     0,ACT
44  01566'101004           MOV     0,0,SZR
45  01567'000776           JMP     .-2
46  01570'060277           INTDS
47  01571'002011$          JMP     @.XEC
48  01572'000000           0
49  01573'000001  CP4:     .BLK    1        ; SAVE PROG COUNTER
50  01574'177777  CP5:     TR2
51  01575'000041  INVC:    ""
52  01576'001602' .COMP3:  COMP3
53  01577'001503' .COMS:   COMP5
54  01600'001605' .ER10:   ER10
55  01601'001607' .PUP1:   PUTP1
56
57  01602'020771  COMP3:   LDA     0,CP4    ; ERROR RETURN
58  01603'040020           STA     0,20     ; RESTORE PROG COUNTER
```

```
---
PAGE  11
06/16/75
 1  01604'000721           JMP     COMP1
 2
 3  01605'001470' ER10:    ERROR           ; PUTP IS ILLEGAL IN
 4  01605'000012           10.             ; CONSTANT ROUTINES
```

```
---
PAGE  12
06/16/75
 1                    ; SUBROUTINE PUTP
 2                    ; WRITES PROGRAM INTO MEMORY
 3
 4                    ; CALLING SEQUENCE :
 5                    ;     PUTP ( =JSR   @.PUTP )
 6                    ;     ADDRESS OF LAST PROG LINE TO BE WRITTEN
 7
 8  01607'024005$ PUTP1:  LDA     1,OPT
 9  01610'020006$         LDA     0,IPT
10  01611'122405          SUB     1,0,SNR ; END OF LIST ?
11  01612'000403          JMP     .+3
12  01613'001665'         ERROR
13  01614'000010          8.                ; TOO MANY PARMS
14  01615'036726          LDA     2,CMAX
15  01616'021400          LDA     0,0,3   ;GET WORD COUNT
16  01617'175400  PUTP2:   INC     3,3     ; BUMP LINE COUNTER
17  01620'025400          LDA     1,0,3   ; GET PROG WORD
18  01621'046020          STA     1,@20,0 ; STA PROG
19  01622'024020          LDA     1,20,0
20  01623'146433          SUBZ#   2,1,SNC ; MEMORY OVERFLOW ?
21  01624'000403          JMP     .+3     ; NO
22  01625'001613'         ERROR
23  01626'100001          @1                ; FATAL
24  01627'116414          SUB#    0,3,SZR ; FINISHED ?
25  01630'000767          JMP     PUTP2   ; NO.
26  01631'000674          JMP     COMP1
27
28  01632'001633' .STORC: STORC             ; START OF STORAGE
29        001633'         STORC = .
30
31        014000          .BLK    14000   ; STORAGE AREA
32        001360'         .END    INIT1
```

REFERENCES

1. Bruggerman, J.M., et al. Network Flow Simulation for Urban Traffic Control System. NTIS U.S. Department of Commerce, Springfield, 1971.
2. Blum, A.M. 'A General Purpose Digital Traffic Simulator'. Simulation, pp. 9-25, Jan. 1970.
3. Bly, P.H. 'A Computer Simulation of an Intersection with a Bus Lane'.. Transportation and Road Research Laboratory Report, pp. 127-131.
4. Beilby, M.H. 'Traffic Simulation by Digital Computer' Traffic Engineering and Control, pp. 21-27, May 1968.
5. Beilby, M.H. 'Road Traffic Simulation on a Small Computer'. The Computer Journal, pp. 134-137. Vol. 15, No. 2, May 1972.
6. Webster, F.V. 'Traffic Signal Settings', Road Research Technical Paper No. 39, (D.S.I.R.) pp. 26-28.
7. Horner, K.D.B. 'Studies of Traffic Behaviour at Intersections Controlled by Traffic Lights'. National Institute for Road Research, report No. RT/2/63, April, 1963.
8. Grigg, P.J. and Hartley, M.G. 'An Interactive Software Traffic Model'. Universities Transport Study Group Conference, London, Jan., 1973.
9. Grigg, P.J. and Hartley, M.G. 'Simulation of Traffic Flows in a Road Network'. Traffic Engineering and Control, pp. 75-77, Feb., 1975.

10. Woods, J.V. 'The Design and Construction of a Road Traffic Simulator'. M.Sc. Tech. Thesis, University of Manchester Institute of Science and Technology, 1967.

11. Green, D.H. 'Development of a Special Purpose Simulator for Detailed Traffic Studies'. M.Sc. Tech. Thesis. University of Manchester Institute of Science and Technology, 1963.

12. Powner, E.T. 'A Special Purpose Machine for the Study of Queueing Problems in Traffic Networks'. I.E.E. Conference on Computer Science and Technology, pp. 1-8, July, 1969.

13. Green, D.H., Foulk, P.W. and Hartley, M.G. 'On-line Control of Simulated Traffic Intersections'. Second Convention of Advances in Computer Control, UKAC, Institute of Electrical Engineers, 1967.

14. Francis, J.G.F. and Lott, R.S. 'A Simulation Program for Linked Traffic Signals'. Proc. 2nd Int. Symp. on the Theory of Road Traffic Flow, pp. 257-259, London, 1961.

15. Decision Inc. 'DDOS 1.6 Programmers Reference Manual', Decision Inc., California, 1974.

16. Data General 'Paper tape editor'. Data General, 1969.

17. Data General 'Extended Assembler'. Data General, 1969.

18. M. Gordon. 'Basic Call Package Reference Manual', University of South Africa, unpublished.

19. Data General 'Random Number Generator'. Data General 1969.

20. Murray, H.F. 'A General Approach for Generating Natural Random Variables'. I.E.E.E. Transactions on Computers, IC-19: 1210-1213, 1970.

21. Marsaglia, G. 'Expressing a Random Variable in Terms of Uniform Random Variables'. Annals of Mathematical Statistics, 32: 894-898, 1961.

22. Marsaglia, G., MacLaren, M.D., and Bray, T.A. 'A Fast Procedure for Generating Normal Random Variables'. Communications of the A.C.M., 7: 4-10, 1964.

23. Ramberg, J.S., and Schmeiser, B.W. 'An Approximate Method for Generating Symmetric Random Variables'. Communications of the A.C.M., 15: 987-990, 1972.

24. Sobolewski, J.S., and Payne, W.H. 'Pseudo-Noise Arbitary Amplitude Distribution'. I.E.E.E. Transactions of Computers, c-21: 337-345, 1972.

25. Maritsas, D.G., and Hartley, M.G. 'A Case Study of a Versatile Generator of Repeatable Non poissonian Sequences of Pseudo-Random Pulses'. I.E.E.E. Transactions of Computers, c-19: 924-938, 1970.

26. Walker, A.J. 'Studies on Pseudo-Random Number Generators for use in Traffic Simulations. M.Sc. Dissertation, University of the Witwatersrand, 1973.

27. Martin, F.F. 'Computer Modelling and Simulation'. pp. 201-202, New York: John Wiley & Sons, 1968.

28. Adams, W.F. 'Raod Traffic considered as a Random Series' Journal Institute of Civil Engineers, Vol. 4, pp. 121-130, 1936.

29. Miller, I. and Freund, J.E. 'Probability and Statistics', New Jersey: Prentice Hall, pp. 92-93, 1965.

30. Redshaw, S. 'A Repeatable Random Pulse Generator Using Chaincodes'. M.Sc. Tech. Thesis, University of Manchester. Institute of Science and Technology, 1961.

31. Robinson, P.H. 'Two types of Random-pulse Generator with Multiple Independent Outposts'. M.Sc. Tech. Thesis, University of Manchester Institute of Science and Technology, 1963.

32. Hartley, M.G. 'Development, Design and Test Procedures for Random Generators Using Chaincodes'. Proc. I.E.E., Vol. 116, No. 1, Jan., 1969.

33. Pratt, A.R. 'Fast Pseudo-Random Number Generators for Computers'. The Radio and Electronic Engineer, Vol. 40, No. 2, pp, 83-88, Aug., 1970.

34. Watson, E.J. 'Primitive Polynomials (Mod-2)'. Mathematics of Computation, No. 16, pp. 368-369, 1962.

35. Tsao, S.H. 'Generation of delayed Replicas of Maximal Length Linear Binary Sequences'. Proc. I.E.E., Vol. 111, No. 11, Nov., 1964.

36. Kendall, M.G., and Stuart, A. 'Advanced Theory of Statistics', Vol. 1, 2nd Ed. Charles Griffin & Co. London, 1963.

37. Paradine, C.G., and Rivett, B.H.P. 'Statistical Methods for Technologists'. London: The English Universities Press, 1970.

38. Davenport, W.B., Jnr., 'Probability and Random Processes'. New York: McGraw Hill, p. 275, 1970.

39. Saaty, T.L. 'Elements of Queueing Theory'. McGraw Hill, 1961.

40. Data General 'Debug III Users Manual'. Data General, 1972.

41. Gerlough, D.C., and Wagner, F.A. 'Improved Criteria for Traffic Signals at Individual Intersections'. National Cooperative Highway Research Program, Report 32.

42. Green, D.G., Hartley, M.G., Heath, F.G. and Powner, E.T. 'Road Traffic Simulation Employing a hardware Approach: - Pseudo Random Number Generation'. Simulation, U.S.A.

```
---
PAGE   5
05/16/75
 1  00624'002402           JMP     @DRET1
 2
 3  00625'001400  DRET2:   JMP     0,3
 4
 5  00626'000001  DRET1:           .BLK    1
 6  00627'000625' DG1:     DRET2
 7  00630'000631' .BUFB:   BUFB
 8  00631'000410  BUFB:    .BLK    264.
 9        001241'          NEXT2 = .
10                  ; KILL ROUTINE
11                  ; RETURNS A FILE
12                  ; CALLING SEQUENCE:
13                  ;       KILL,FILENAME
14
15  01241'001252'   .       KILL
16  01242'000000            0
17  01243'000000            0
18  01244'000000            0
19  01245'000000            0
20  01246'001267'           NEXT3
21  01247'045511            .TXT    *KILL*
22  01250'046114
23  01251'000000
24
25  01252'020001- KILL:    LDA     0,.NAME
26  01253'000553'          GETAS
27  01254'030001-          LDA     2,.NAME
28  01255'034005-          LDA     3,DDOS
29  01256'007402           JSR     @D.SRC,3
30  01257'000402           JMP     .+2     ; FILE DOES NOT EXIST
31  01260'000403           JMP     .+3
32  01261'000561'          ERROR
33  01262'000014           12.
34  01263'030001-          LDA     2,.NAME
35  01264'007400           JSR     @D.CYS,3
36  01265'000012           0.RET           ; RETURN FILE
37  01266'000000'          RETURN
38        001267'          NEXT3 = .
39
40                  ; BYE ROUTINE
41                  ; TRANSFERS CONTROL TO DDOS
42                  ; CALLING SEQUENCE.
43                  ;       BYE
44
45  01267'001277'          BYE
46  01270'000000           0
47  01271'000000           0
48  01272'000000           0
49  01273'000000           0
50  01274'001301'          DEND            ; END OF DISK
51  01275'041131           .TXT    *BYE*
52  01276'042400
53
54  01277'034005- BYE:     LDA     3,DDOS
55  01300'001444           JMP     D.RST,3
56
57        001301'          DEND = .
```

Author Yamey C G
Name of thesis An executive program and hardware data sources for high speed traffic simulation using a minicomputer 01151

PUBLISHER:
University of the Witwatersrand, Johannesburg
©2013

LEGAL NOTICES:

Copyright Notice: All materials on the University of the Witwatersrand, Johannesburg Library website are protected by South African copyright law and may not be distributed, transmitted, displayed, or otherwise published in any format, without the prior written permission of the copyright owner.

Disclaimer and Terms of Use: Provided that you maintain all copyright and other notices contained therein, you may download material (one machine readable copy and one print copy per page) for your personal and/or educational non-commercial use only.

The University of the Witwatersrand, Johannesburg, is not responsible for any errors or omissions and excludes any and all liability for any errors in or omissions from the information on the Library website.