THE mini• tasker

DECUS
RT-11 SIG NEWSLETTER

FEBRUARY 1980 VOL. 6 NO. 1

Contributions to the newsletter should be sent to:

Ken Demers
MS-48
United Technologies Research Center
East Hartford, Connecticut 06108
203 727-7241

Other communications can be sent to:

John T. Rasted
JTR Associates
58 Rasted Lane
Meriden,Conn. 06450
203 634-1632

or

RT-11 SIG
C/O DECUS
One Iron Way
MR2-3/E55
Marlboro,Mass. 01752
617 481-9511 Ext. 4141

FROM THE EDITOR

The article on TSX (multi-user RT-11) has generated much interest. However, I do not have Nigel Bevan's address in England. As soon as I receive it, I will publish it in the next "Minitasker". Those desiring his address earlier can call me. *

To obtain a copy of the latest Symposium tape, if you live outside the U.S.A., contact:

Howard Schultens
Physiologisches Inst. 2
Humboldtallee 7
D-3400 Goettingen
W. Germany
Telex: 96703
Tel: 0551/395914

Ir. J. Loeve
Head Dept. CRW/ASV
Eramus University, Rotterdam
P.O. Box 1738
3000 Dr.
Rotterdam, Holland

Ian Calhaem
D.S.I.R
P.O. Box 1320
Wellington
New Zealand

*NOTE: I obtained Nigel's address just before printing:

Nigel Bevan
National Physical Laboratory
DNACS
Teddington, Middx, TW11 OLW
ENGLAND

RT-11 STEERING COMMITTEE

JOHN T. RASTED — RT-11 SIG CHAIRMAN
JTR ASSOCIATES
58 RASTED LANE
MERIDEN, CT 06450
(203) 634-1632

KEN DEMERS — NEWSLETTER EDITOR, DECNET CONTACT
MS-48
UNITED TECHNOLOGIES RESEARCH CENTER
SILVER LANE
EAST HARTFORD, CT 06108
(203) 727-7241

MARK BARTELT — RT-11 LUG COORDINATOR, RT/RSX CONTACT, WEST COAST
CALIFORNIA INST. OF TECH.
MS 356-48
1201 CALIFORNIA
PASADENA, CA 91125
(213) ~~795-6811~~ EXT ~~2663~~ 356 - 6811

NICK BOURGEOIS / 1736 — TAPE COPY COORDINATOR, MACRO CONTACT
SANDIA LABORATORIES
P.O.BOX 5800
ALBUQUERQUE, NM 87185
(505) 844-8088

JAMES R. CUTLER — STANDARDS COORDINATOR, RT/RSX CONTACT, MIDWEST
SPACE PHYSICS RESEARCH LAB
2455 HAYWARD
ANN ARBOR, MICH 48105
(313) 763-9940

ART HERMES — RT/RSX CONTACT, EAST COAST
MIT/LNS BATES LINEAR ACCELERATOR
P. O. BOX 95
MIDDLETON, MA 01949
(617) 245-6600

CARL LOWENSTEIN — RT-11 HHK COORDINATOR, SIG HARDWARE CONTACT
UNIVERSITY OF CALIF, SAN DIEGO
MARINE PHYSICAL LABORATORY OF THE
SCRIPPS INSTITUTION OF OCEANOGRAPHY
SAN DIEGO, CA 92152
(714) 294-3678

ERIC MORTON
SANDERS TECH. INC.
P.O.BOX 1226
NASHUA, NH 03061
(603) 882-1000 EXT 253

RT-11 DECUS LIBRARY
COORDINATOR
C CONTACT

JOHN L. PRATHER
PA COLL. OF POD. MED.
8TH _RACE STREETS
PHILADELPHIA, PA 19107
(215) 629-0300 EXT 155

FORTRAN CONTACT

SUSAN S. RASTED
SOFTWARE DYNAMICS INC.
1000 YALE AVE.
WALLINGFORD, CT 06492
(203) 265-2226

FMS-11 CONTACT

MARILYN RUNYON
39 LOCUST POINT RD.
LOCUST, N.J. 07760

WISH LIST

EDWIN J. STEVENS
EMDA INC.
899 EAST GREEN ST.
PASADENA, CA 91106
(213) 795-5991

MU-BASIC CONTACT

RAY STRACKBEIN
CHALFRONT COMMUNICATIONS
73-680 HIGHWAY 111
PALM DESERT, CA 92260
(714) 346-3987

SYMPOSIA COORDINATOR

J. W. TIPPIE
ARGONNE NATIONAL LABS
9700 S. CASS AVE
ARGONNE, ILL 60439

CAMAC CONTACT

GREG WOODBURY
DEPT. OF MICROBIOLOGY
BOX 3020
DUKE UNIV. MED. CENTER
DURHAM, N.C. 27710
(919) 684-5138

BASIC CONTACT

ALAMO PDP-8 & PDP-11 LOCAL USERS GROUP

Chairman: Douglas Threatt
USAF School of Aerospace Medicine
Biometric Division (BRP)
Brooks AFB, TX 78235
(512) 536-3886

ALBUQUERQUE RT-11 LOCAL USERS GROUP (ARTLUG)

Chairman: N.A. Bourgeois, Jr.
Sandia Labs 11736
P.O. Box 5800
Albuquerque, NM 87185
(505) 264-8088

BAY AREA PDP-11 LOCAL USERS GROUP (BUG)

Chairman: Joan Salkin
806 Taurus Drive
Foster City, CA 94404
(415) 349-3173

BAY RT-11 LOCAL USERS GROUP

Chairman: Lyle Ryan
Electronic Engineer
CETUS
600 Bancroft Way
Berkeley, CA 94710
(415) 549-3300

BIOMEDICAL COMPUTING UCLA LOCAL USERS GROUP

Chairman: Dr. R.M. Harper
UCLA
Dept. of Anatomy
Los Angeles, CA 90024

CAROLINA'S LOCAL USERS GROUP

Chairman: John Scogin, III
Lockwood Greene Engineers
P.O. Box 491
Spartanburg, SC 29304
(803) 582-2132

CENTRAL ALABAMA PDP-11 LOCAL USERS GROUP

Chairman: Roy Swatzell
V.A. Hospital
700 S. 19th St.
Birmingham, AL 35233
(205) 933-8101 X239

CENTRAL OKLAHOMA PDP-ELEVEN USERS SOCIETY (COPEUS)

Chairman: Mike O'Dell
University of Oklahoma
202 W. Boyd
Room 107
Norman, OK 73019
(405) 325-2621

CHESAPEAKE AREA RT-11 LOCAL USERS GROUP

Chairman: Henry Dardy
Naval Research Laboratory
Code 8133
Washington, D.C. 20375
(202) 767-3336

CHINA LAKE RT-11 LOCAL USERS GROUP

Chairman: Michael Levine
Naval Weapons Center
Code 3513
China Lake, CA 93555
(714) 939-3575

EASTERN TENNESSEE PDP-11 LOCAL USERS GROUP

Chairman: Kenneth J. Cross
Oak Ridge National Lab
Box X
Bldg. 3500
Oak Ridge, TN 37830
(615) 574-5729

FLORIDA PDP-11 LOCAL USERS GROUP

Chairman: Robert Carson
Rollins College
Physics Dept.
Winter Park, Fla. 37289
(305) 646-2000 X2517

FLORIDA USERS OF RT-11 (FLURT)

Chairman: Patrick Perrott
PELCOR
1607 Forsyth Rd.
Orlando, FL 32801
(301) 275-1132

ITHACA MINICOMPUTER LOCAL USERS GROUP (IMLUG)

Chairman: Ray Helmke
Cornell University
Wilson Laboratory
Ithaca, NY 14853
(607) 256-4882

MADISON WISCONSIN LOCAL USERS GROUP

Chairman: Richard Marleau
University of Wisconsin
Dept. of Elec. & Comp.
Madison, WI 53706
(608) 262-1191

METRO ED LOCAL USERS GROUP

Chairman: Bruce Alcock
Riverdale Country School
West 253rd St. & Fieldstone Rd.
Bronx, NY 10471
(212) 549-8810

MIDLANDS LOCAL USERS GROUP

Chairman: Donal Caviness
Air Force Global Weather Central
Offutt AFB, NE 68113
(402) 294-3073

MINNESOTA LOCAL USERS GROUP (MINNELUG)

Chairman: Richard Newman
University of Minnesota
Biomedical Center
Diehl Hall
Minneapolis, MN 55455
(612) 373-2156

MIT RT-11 LOCAL USERS GROUP (MITLUG)

Chairman: Thomas Provost
MIT/LNS Bates Linear Accelerator
P.O. Box 95
Middleton, MA 01949
(617) 245-6600

NEW ENGLAND USERS OF RT-11 (NEURT)

Chairman: Eric Morton
Sanders Tech Inc.
P.O. Box 1226
Nashua, NH 03061
(603) 882-1000 X253

NORTHWEST LSI EDUCATIONAL LOCAL USERS GROUP

Chairman: Ron Tenison
Catlin Gabel School
8825 SW Barnes Road
Portland, OR 97225
(503) 297-1894

NY METRO LOCAL USERS GROUP

Chairman: John Runyon
FTCC
25 Broad St.
New York, N.Y. 10004
(212) 747-5573

PDP-11 SEABOARD LOCAL USERS GROUP

Chairman: Karen Hill
Princeton Plasma Physics
C-Site
Forrestal Campus
Princeton, N.J. 08540
(609) 452-5092

PENN STATE LOCAL USERS GROUP

Chairman: Rich Hoover
Penn State University
USDA Pasture Res. Lab.
University Park, PA 16802

PHOENIX LOCAL USERS GROUP

Chairman: Ken Bates
Realtime Systems
4621 N. 16th St.
Suite E-509
Phoenix, AZ 85016

PITTSBURGH AREA LOCAL USERS GROUP

Chairman: Richard Pethia
LRDC Bldg.
University of Pittsburgh
3939 O'Hara St.
Pittsburgh, PA 15260
(412) 624-4905

PORTLAND AREA PDP-11 LOCAL USERS GROUP

Chairman: Bill Bagley
Stan Wiley Realtors
200 SW Market
Suite 1800
Portland, OR 97201
(503) 222-5561 X7472

SAN DIEGO PDP-11 LOCAL USERS GROUP

Chairman: Rich Laverty
NARF Bldg. 463, Code 332
North Island
San Diego, CA 92135

SOUTHEASTERN MICHIGAN PDP-11 LOCAL USERS GROUP

Chairman: James Cutler
University of Michigan
Space Physics Res.
2455 Hayward
Ann Arbor, MI 48105
(313) 764-6587

SOUTHERN CALIFORNIA USERS OF RT-11 (SCURT)

Chairman: Ray Strackbein
Chalfront Communications
73-680 Highway 111
Palm Desert, CA 92260
(714) 346-3987

SPOKANE AREA LOCAL USERS GROUP

Chairman: Jerry Clark
American Sign & Indicator
P.O. Drawer 2727
Spokane, WA 99220
(509) 535-4101

ST. LOUIS PDP-11 LOCAL USERS GROUP

Chairman: Richard Aurbach
Monsanto Agricultural
Products Company
800 N. Lindbergh
St. Louis, MO 63166
(314) 694-5453

UPPER VALLEY PDP-11 LOCAL USERS GROUP

Chairman: Steve Ladd
CREARE Inc.
P.O. Box 71
Hanover, NH 03755
(603) 643-3800

WESTERN RESERVE USERS GROUP (WRUG)

Chairman: John Kremer
U.S. Steel-Lorain Works
1807 E. 28th St.
MS 10B
Lorain, OH 44055
(216) 277-2890 or 2403

USER REQUESTS

If anyone has any information on a handler for the LA36 that will run under RT-11 V3B (with Multi-Terminal support), it would be greatly appreciated. I did not include the DL-11 that the LA36 is attached to as one of the Multi-Terminal devices. It is free to be used as a handler controlled device. Please send any information to:

Ron Pipes
U.S. Borax Research Corp.
412 Crescent Way
Anaheim, Calif. 92801

USER RESPONSES

Reply to Randy Lee, Nov. '79 Mini-Tasker --

When using TECO, the quickest way to insert ASCII characters is the <n>I$ command. This inserts the character whose ASCII value is <n> (<n> is interpreted in the current radix). To insert the right set bracket symbol, "}" , the command 125I$ should be given if the current radix is 10, otherwise 175I$ should be given for radix 8. The same can be done for the tilde, "^".

If the character will be used frequently (like the set brackets in C or Pascal) then the character could be stored in a Q-register. Then, when the character is needed, the character would just be retrieved.

This is not as easy as just typing the character, but it will work until RT-11 is changed to allow the characters at the high end of the ASCII set to be entered without converting them to escapes.

ENHANCEMENTS TO VT52.TEC AND VEG.TEC

In the RT-11 V3B distribution kit there are some files that contain TECO macros; two of these are to give VT52 users a powerful screen editor with cursor control. The first macro, VEG.TEC, is just a startup macro. It gets the name of the file to be edited and loads the screen control macro, VT52.TEC, and the file. The macro will also give help if the user requests it (the available commands are shown). The VT52 macro is the controller for the screen editor. With VT52 running, the entire screen becomes a window into the file, with the cursor showing were the pointer is. Commands are entered through the numeric keypad or as control characters (the help part of VEG.TEC tells what all the keys mean). Any characters that are not commands are immediately inserted, and the screen is updated. The editor can be learned in a few minutes, and all editing is much faster and easier than with standard TECO or EDIT.

The first problem with the screen macros is that they do not work under the FB monitor until a SET TT NOCRLF is given. Without it, the screen fills up with garbage. I have a command file to start up the editor called SCOPE.COM as follows:

```
SET TT NOFB
SET TT NOPAGE
SET TT NOCRLF
EDIT/EXECUTE/TECO SY:VEG.TEC
SET TT CRLF
SET TT PAGE
SET TT FB
```

Now for editing, I just enter @SCOPE and everthing is taken care of.

After using the editor for awhile, I thought of some nice enhancements that would suit my needs a little better. The followings changes are the results:

To VEG.TEC --

1. The file size can be entered using the square brackets ([and]).

2. The command summary and keyboard map is printed only if a question mark is entered for the file name. After printing the help, it asks for the filename again.

3. The old file name (which is saved from one edit to the next so that if you are editing a file several times, you don't need to reenter the file name) can be edited. If you want to change just one character instead of erasing the whole name and retyping the new one, backspace the cursor to where you want the change, delete the old charcaters and type the new ones.

4. If the file EDTMAC.TEC is on SY:, it is read in and placed in the Q-register G as a macro. This allows a macro that is large to be saved in a file. It is handy if a certain editing function is needed regularly enough to warrant writing a TECO macro.

5. File name echo is much quicker.

To VT52.TEC --

New Control character commands -

ctrl-G - Executes the macro that VEG.TEC read from EDTMAC.TEC

ctrl-P - Equvalent to <esc>24. Used as a Page command. Example: ctrl-P followed by cursor down will moe the cursor 24 lines down, thus brinsins a completely new page onto th screen.

ctrl-A - Add lines to the R buffer. This puts lines into the secondary save buffer (besides the one controlled by the numeric pad).

ctrl-E - The enters characters into the R buffer. The difference between ctrl-A and ctrl-E is that <esc><n> ctrl-A puts <n> LINES into the buffer, while <esc><n> ctrl-E puts <n> CHARACTERS into the buffer. This is handy for savins only parts of lines.

ctrl-R - This retrieves whatever was in the R buffer and inserts where the cursor is.

ctrl-Z - Does an EX command. If <esc> ctrl-Z is entered, the file is closed, and a @EDTFIN.COM is siven. This file can contain a command to compile the last file edited (COMPILE @SY:EDIT.FIL) or any other monitor commands.

For those who have not used VT52.TEC, these are only the commands I added. There are already commands for cursor movement, strins searchins, insertins text, deletins text (lines or characters), movins blocks of text, plus the ability to enter direct TECO commands. All in all, the screen editor is a bis improvement over the standard hardcopy terminal type of editor.

If you have TECO and would like a copy of the macros for screen editins (the improved ones), send a floppy disk to:

Ron Pipes
U.S. Borax Research Corp.
412 Crescent Way
Anaheim, Calif. 92801

USER INPUT

DATBAS (DECUS 11-294) -- ERROR OF OMISSION

This refers to a set of BASIC-11 V2 extension routines that permit readins and writins at any implemented bus address. A user has reported a problem with the GETB routine. When the variable type is numeric (A% or A) the value of the returned arsument ranses from 0 to 255 rather than from -128 to 127. This problem may be avoided by rescalins the value after its return to BASIC as follows:

```
CALL GETB(A$,B)
IF B=>128 THEN B=B-256
```

When I set around to fixins this, I'll resubmit it to the DECUS library.

Nick Bourseois

NOVA biomedical

Dear Mr. Demers:

Most of our 8080/8085 software is developed on our PDP-11 (RT-11 V03). A cross assembler creates an Intel formated ASCII hex file on an RX01 floppy disk. Some months ago we purchased a Data I/O system 19 PROM programmer, and we needed a program to transmit the hex file from the disk to the Data I/O's RAM.

Enclosed is a listing of the program that has worked well for us. Other DECUS members may find it useful.

We chose to use the current loop serial link, and this choice nesessitated hardware changes. The opto-isolator on the DAta I/O was changed to a faster one. Also, the Data I/O supplies only 2mA to the loop that detects transmissions from the PDP-11. Therefore, we had to install a booster circuit to present the proper current to the PDP 11's DLV-11. Diagrams showing the changes are enclosed.

Very truly yours,

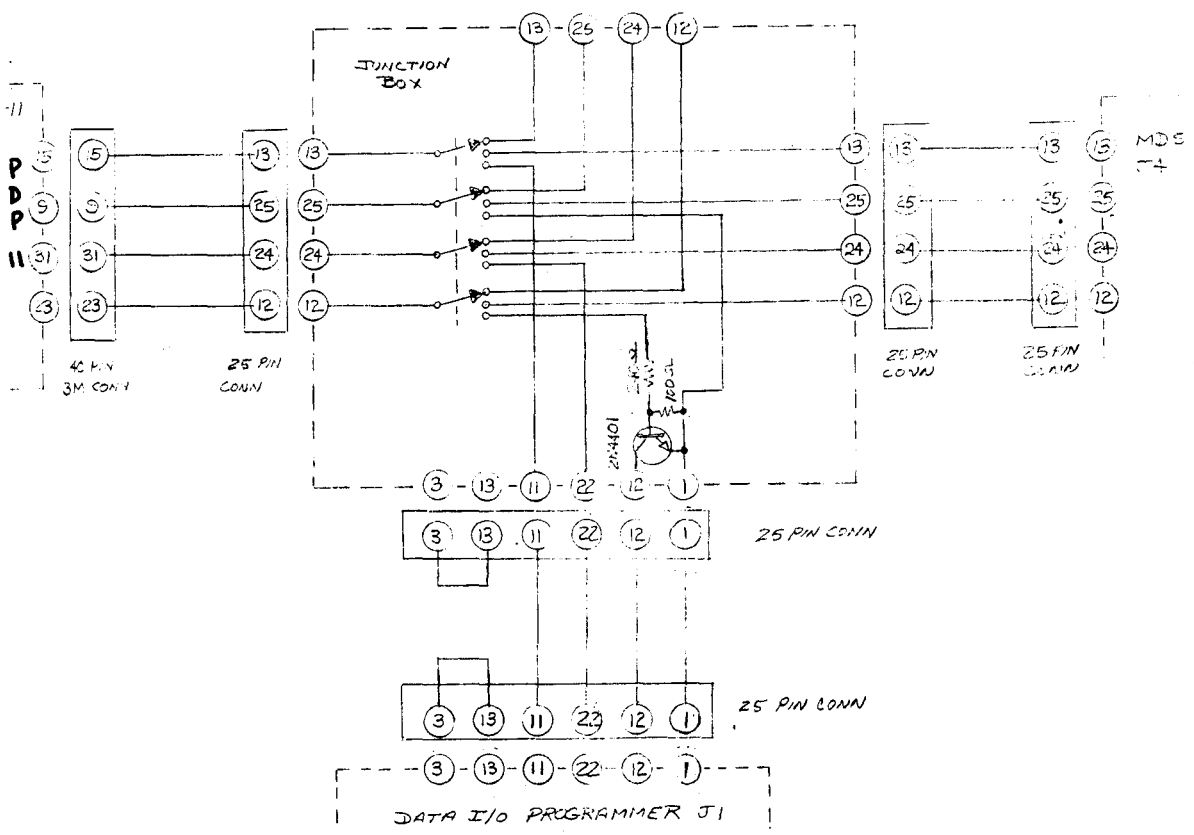
Ted Knowlton

Ted Knowlton
Software Engineer

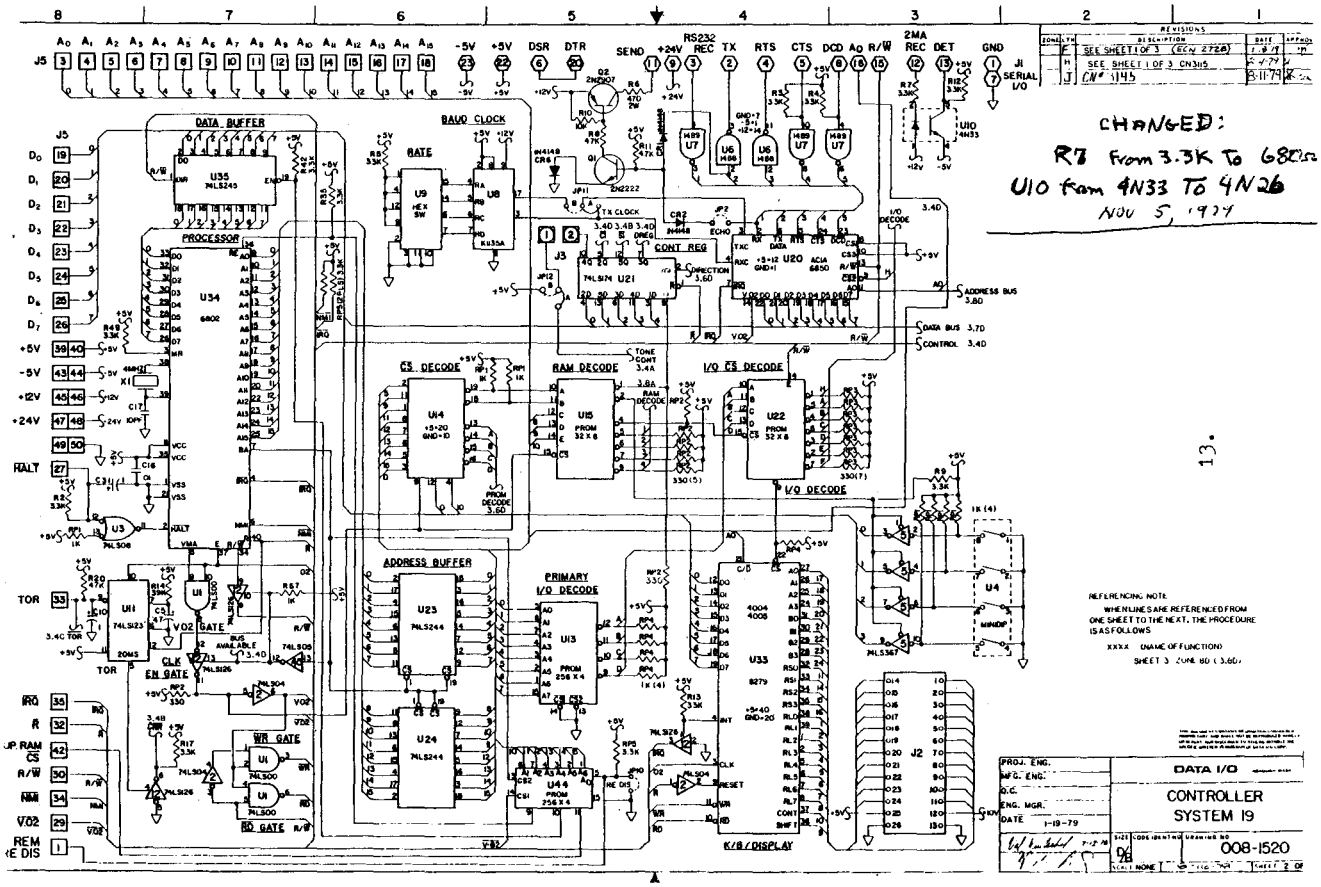
JUNCTION BOX

PDP/MDS/DATA I/O

10/22/79



14.



CHANGED:
R7 From 3.3K To 680Ω
U10 From 4N33 TO 4N26
NOV 5, 1979

REFERENCING NOTE
WHEN LINES ARE REFERENCED FROM ONE SHEET TO THE NEXT, THE PROCEDURE IS AS FOLLOWS
XXXX (NAME OF FUNCTION)
SHEET 3 ZONE 8D (3.6D)

PROJ. ENG.	DATA I/O
MFG. ENG.	CONTROLLER SYSTEM 19
Q.C.	
ENG. MGR.	
DATE 1-19-79	008-1520

13.

```
                        .LIST TTM       ;NARROW LISTING
                        .ENABL LC       ;ENABLE LOWER CASE

                    ; TRANSMIT INTEL HEX FORMAT FILE FROM PDP 11 DISK
                    ; TO DATA I/O SYSTEM 19 PROGRAMMER.

                    ;                   REVISED   27 DEC 79

                    ; This program transmits a 4K (max.) portion of
                    ; an Intel format Hex file from PDP 11 (RT-11 V03)
                    ; to a DATA I/O System 19 Prom Programmer.
                    ; Before calling this program, put desired PROM
                    ; module in Programmer.  Turn on power, press
                    ; 'SELECT', then 'F1' and 'START'.  Programmer is
                    ; then under remote control from RT-11.
                    ; After '*', type one Hex file name.  After file is
                    ; read and prompt displayed, type starting Hex
                    ; address - which must be at 1K boundaries;
                    ; e.g. 0, 400, 800, C00, 1000, 1400 etc.
                    ; After typing 'Return', go to Programmer, wait
                    ; for completed transmission, exit remote control,
                    ; (modify data), program PROM's, reenter remote
                    ; control; continue with another 4K portion, if
                    ; desired.

                    ; NOTE:  The buffer size set for this program is
                    ; 40K bytes.  At 43 characters/record and 16 data
                    ; bytes/record, the buffer will hold what amounts
                    ; to about 15K bytes of object code.  A hex file
                    ; that is too big may be split up.  It will be
                    ; necessary to edit in a ':00' at the split so
                    ; that program will reconnize end of file.

                            .MCALL  .CSIGEN, .READW
                            .MCALL  .GTLIN, .TTYOUT
                            .MCALL  .EXIT, .SETTOP
                            .MCALL  .CLOSE

                    ; DLV11 ADDRESSES

            175610  RCSR=175610     ; RCVR CSR
            175612  RBUF=RCSR+2     ; RCV BUFFER
            175614  XCSR=RCSR+4     ; XMIT CSR
            175616  XBUF=RCSR+6     ; XMIT BUFFER

                    ; MASKS FOR XCSR, RCSR
            000200  RCVDN=200       ; RCVR DONE
            000200  XMRDY=200       ; XMIT READY

                    ; For debugging use:
                    ; LINK/MAP:TT:/DEBUG PROM.OBJ

000000              PROM::  ;DOUBLE COLON FOR GLOBAL SYMBOL (FOR ODT)
000000                      SETTOP #HEXBEE          ;SET HIGHEST ADRS
000006  013704      XM:     MOV @#RBUF, R4
        175612
000012  013704              MOV @#RBUF, R4          ;CLEAR DLV11
        175612
000016  012704              MOV #MSG1, R4
        001402'
000022                      .TTYOUT (R4)+           ;TYPE MSG ON DECSCOPE
000030  020427              CMP R4, #MSGE1
        002604'
000034  001372              BNE XM
000036              XMRST:
000036  012700              MOV #<MSGE2-MSG2>, R0
        000002
000042  012704              MOV #MSG2, R4
        002604'
000046  004567              JSR R5, XMTMSG          ;XMIT 'H' (RET)
        001050
000052  005703              TST R3
000054  001011              BNE XM1         ;BR IF NO ERROR
```



```
000056                      XME:                    ;COMMUNICATION ERROR
000056  012704              MOV #MSG3, R4
        002606'
000062                      XME1:   .TTYOUT (R4)+
000070  020427              CMP R4, #MSGE3
        002640'
000074  001372              BNE XME1
000076                      .EXIT
000100                      XM1:
000100  012700              MOV #<MSGE4-MSG4>, R0
        000004
000104  012704              MOV #MSG4, R4 ; 83A (Intel format)
        002640'
000110  004567              JSR R5, XMTMSG
        001006
000114  005703              TST R3
000116  001757              BEQ XME

000120  012700              MOV #<MSGE6-MSG6>, R0
        000002
000124  012704              MOV #MSG6, R4           ; U (add LF)
        002651'
000130  004567              JSR R5, XMTMSG
        000766
000134  005703              TST R3
000136  001747              BEQ XME

000140                      .CSIGEN @#50,#DEFTY,#0 ;GET FILE NAME, ETC
000154  005005              CLR R5
000156                      .READW #ARGBL,#3,#HEXBUF,#HEXBFL, R5
000214                      .CLOSE #3
000222  000410              BR START

000224                      RESTRT:
000224  012704              MOV #MSG8, R4
        002657'
000230                      RST1:   .TTYOUT (R4)+
000236  020427              CMP R4, #MSGE8
        002706'
000242  001372              BNE RST1

000244                      START:
000244                      .GTLIN #LINBUF, #PROMPT         ;GET STARTING ADRS
000244  012702              MOV #LINBUF, R2
        001366'
000270  005001              CLR R1          ;INITIALIZE R1
000272                      START1:
000272  112200              MOVB (R2)+, R0
000274  001435              BEQ PARAM       ;END OF PARAM INPUT
000276  020227              CMP R2, #LINBUF+5
        001373'
000302  002350              BGE RESTRT      ;TOO MANY CHARS
000304  004567              JSR R5, HEXTST          ;CHECK FOR ASCII HEX CHAR
        000550
000310  005703              TST R3
000312  001744              BEQ RESTRT      ;START OVER IF NOT LEGIT CHAR
000314  122700              CMPB #'0, R0
        000060
000320  001410              BEQ START5      ;CHECK POS'N OF ZERO
000322  005701              TST R1
000324  001404              BEQ START3      ;BR IF NOTHING ENTERED
000326                      START2:
000326  032701              BIT #177400, R1
        177400
000332  001357              BNE START1      ;BR IF 2 CHARS ALREADY ENTERED
000334  000301              SWAB R1
000336                      START3:
000336  060001              ADD R0, R1
000340  000754              BR START1       ;GET ANOTHER CHAR

000342                      START5:
000342  005701              TST R1
```

```
127 000344  001752      BEQ START1      ;BR IF LEADING ZERO
128 000346  000767      BR START2
129
130 000350              PARAM:
131 000350  005701      TST R1
132 000352  001005      BNE PAR1        ;BR IF NOT 0
133 000354  012701      MOV #30060, R1          ;ASCII '00'  (=LO LIMIT)
            030060
134 000360  012702      MOV #30460, R2    ;ASCII '10' (=HI LIMIT)
            030460
135 000364  000454      BR OUTPAR
136 000366              PAR1:
137                     ; 2 ASCII HEX CHARS IN R1
138 000366  020227      CMP R2, #LINBUF+4
            001372'
139 000372  001027      BNE PARE        ;BR IF NOT 3 CHARS TOTAL RCVD
140
141                     ; 3 CHARS TOTAL RCVD.  MUST=04,08,0C ELSE->START
142 000374  122701      CMPB #'0, R1
            000060
143 000400  001311      BNE RESTRT      ;ILLEGAL IF NOT '0'
144 000402  000301      SWAB R1
145 000404  010102      MOV R1, R2
146                     ; PREPARE R2=2K HIGHER THAN R1
147 000406  122702      CMPB #'4, R2
            000064
148 000412  001003      BNE PAR01
149 000414  012702      MOV #30464, R2          ;LOW=400H, HI=1400H
            030464
150 000420  000436      BR OUTPAR
151 000422              PAR01:
152 000422  122702      CMPB #'8, R2
            000070
153 000426  001003      BNE PAR03
154 000430  012702      MOV #30470, R2          ;LO=800H, HI=1800H
            030470
155 000434  000430      BR OUTPAR
156 000436              PAR03:
157 000436  122702      CMPB #'C, R2
            000103
158 000442  001270      BNE RESTRT
159 000444  012702      MOV #30503, R2          ;LO=C00H, HI=1C00H
            030503
160 000450  000422      BR OUTPAR
161
162                     ; 4 CHARS RCVD -- EVEN
163 000452              PARE:
164 000452  020227      CMP R2, #LINBUF+5
            001373'
165 000456  001267      BNE RESTRT      ;MIN 4 CHARS ELSE ERROR
166 000460  122701      CMPB #'0, R1
            000060
167 000464  001411      BEQ PARE1
168 000466  122701      CMPB #'4, R1
            000064
169 000472  001406      BEQ PARE1
170 000474  122701      CMPB #'8, R1
            000070
171 000500  001403      BEQ PARE1
172 000502  122701      CMPB #'C, R1
            000103
173 000506  001246      BNE RESTRT
174 000510              PARE1:
175 000510  010102      MOV R1, R2
176 000512  062702      ADD #400, R2
            000400
177
178                     ; NOTE:  NO HIGH ADDRESS LIMIT IS TESTED.
```



```
; 000516                OUTPAR:
2
3                       ; R1=HIGH ORDER 2 ASCII HEX CHARS SPECIFYING
4                       ; LOW LIMIT.  E.G.  04 => 400
5                       ;                   2C => 2C00
6
7                       ; R2=ADRS IN R1 SPECIFYING HIGH LIMIT+1K
8                       ; E.G.    14 => 1400+400=1800
9                       ;         3C => 3C00+2C00+1000
10 000516  010100       MOV R1, R0
11 000520  004567       JSR R5, XMTMR ;XMIT VIRTUAL MEMORY OFFSET
            000532
12 000524  012700       MOV #CMSGLT-MSG7>, R0
            000004
13 000530  012704       MOV #MSG7, R4           ;'00W CR'
            002653'
14 000534  004567       JSR R5, XMTMSG          ;REST OF VIR MEM CMD
            000362
15 000540  005703       TST R3
16 000542  001002       BNE OP1         ;OK
17 000544  000167       JMP XME
            177306
18 000550              OP1:
19 000550  012700       MOV #44415, R0          ; 'I' CR
            044415
20 000554  004567       JSR R5, XMTNR
            000476
21 000560  005003       CLR R3
22 000562              OP2:
23 000562  005303       DEC R3
24 000564  001376       BNE OP2                 ;DELAY
25
26 000566  012704       MOV #HEXBUF, R4
            002706'
27 000572  005067       CLR NTLF
            000534
28 000576  005067       CLR NTLCA
            000534
29
                        ;SEARCH PARAMETERS
30 000602              SPAR:
31 000602  005767       TST NTLF
            000524
32 000606  001402       BEQ SP1AA
33 000610  016704       MOV LCA, R4             ;LAST COLON ADRS
            000520
34 000614              SP1AA:
35 000614  122427       CMPB (R4)+, #':
            000072
36 000620  001370       BNE SPAR
37 000622  010405       MOV R4, R5
38 000624  005305       DEC R5          ;R5=ADRS WHERE ':' FOUND
39 000626  122427       CMPB (R4)+, #'0
            000060
40 000632  001402       BEQ SP1A
41 000634  005204       INC R4
42 000636  000403       BR SP1B
43 000640              SP1A:
44 000640  122427       CMPB (R4)+, #'0
            000060
45 000644  001466       BEQ SP7         ;BR IF EOF, CHAR CNT=0
46 000646              SP1B:
47 000646  112400       MOVB (R4)+, R0
48 000650  000300       SWAB R0
49 000652  112403       MOVB (R4)+, R3
50 000654  060300       ADD R3, R0      ;R0 NOW=HIGH 2 OF RECORD ADRS
51 000656  020001       CMP R0, R1      ;LOW LIMIT
52 000660  100474       BMI SP1J        ;BR IF ADRS < LOW LIMIT
53 000662  001024       BNE SP1D
54 000664  005767       TST NDF
            000442
55 000670  001403       BEQ SP1C        ;BR IF NOT FLAGGED
56 000672  005067       CLR NTLF
            000434
57 000676  000420       BR SP1F
```

```
                        SP1D:
                        ; EXAMINE L.O. 2 ADRS BYTES (LO2)
                        ; IF LO2=0X, WHERE X <> 0, TRANSMIT PREVIOUS
                        ; RECORD BECAUSE IT CONTAINS SOME DATA WITHIN
                        ; CHOSEN PARAMATERS.  SET FLAG.
000700  122427 000060     CMPB (R4)+, #'0
000704  001015            BNE SP1E
000706  122427 000060     CMPB (R4)+, #'0
000712  001412            BEQ SP1F
                        ; CONDITIONS MET
000714  010567 000414     MOV R5, LCA         ;SAVE THIS COLON'S ADR
000720  052767 000377     BIS #377, NTLF      ;SET FLAG
        000404
000726  016705 000404     MOV NTLCA, R5 ;ADRS OF PREVIOUS RECORD
000732  000402            BR SP1F
                        SP1D:
000734  020027 002321     CMP R0, R2
000740                  SP1E:
                        ; RECORD DATA ADRS IS WITHIN CHOSEN PARAMS.
000740  010504            MOV R5, R4
000742                  SP1:
000742  121427 000012     CMPB (R4), #12      ;LF
000746  001412            BEQ SP5
000750                  SP3:
000750  033727 175614     BIT @#XCSR, #XMRDY
        000200
000756  001774            BEQ SP3
000760  112437 175616     MOVB (R4)+, @#XBUF  ;XMIT CHAR
000764  005705            TST R5
000766  001705            BEQ SPAR        ;BR IF END OF RECORD
000770  010405            MOV R4, R5      ;MAKE R5 <> 0
000772  000763            BR SP1
                        ; END OF RECORD
000774                  SP5:
000774  005005            CLR R5
000776  033727 175610     BIT @#RCSR, #RCVDN
        000200
001004  001761            BEQ SP3         ;XMT & CONTINUE SEARCH
001006  123727 175612     CMPB @#RBUF, #12
        000012
001014  001755            BEQ SP3
001016  000167 177034     JMP XME         ;COMMUN. ERROR


                        ; END OF FILE
001022                  SP2:
001022  012700 000005     MOV #<MSGF5-MSG5>, R0
001026  012704 002644'    MOV #MSG5, R4           ; :00
001032  004567 000064     JSR R5, XMTMSG
001036  005703            TST R3
001040  001002            BNE SP9
001042  000167 177010     JMP XMF
001046                  SP9:
001046  000167 177172     JMP START       ;GET READY FOR ANOTHER INPUT

001052                  SP11:
001052  010567 000260     MOV R5, NTLCA ;SAVE ADRS OF THIS COLON
```



```
001056  000656            BR SP1AA


                        ; SUBROUTINE:
                        ; TEST CONTENTS OF R0 FOR ASCII HEX DIGIT.
                        ; CAUSE BRANCH (ON RETURN) TO "START" IF FAIL.

001060                  HEXTST:
001060  012703 000001     MOV #1, R3
001064  120027 000060     CMPB R0, #'0
001070  103412            BLO HEXTF1
001072  120027 000106     CMPB R0, #'F
001076  101007            BHI HEXTF1
001100  120027 000071     CMPB R0, #'9
001104  003403            BLE HEXTX
001106  120027 000101     CMPB R0, #'A
001112  103401            BLO HEXTF1
001114                  HEXTX:
001114  000205            RTS R5
001116                  HEXTF1:
001116  005303            DEC R3
001120  000205            RTS R5


                        ; TRANSMIT MSG CODE TO DATA I/O PROGRAMMER;
                        ; AWAIT RESPONSE.
                        ; R0=CHAR COUNT OF MSG
                        ; R1,R2 NOT USED NOR ALTERED.
                        ; R3=0 ON RETURN, ONLY IF ERROR
                        ; R4=ADRS OF MESSAGE.
                        ; R0,R3 USED AS TIMER, INTERNALLY

001122                  XMTMSG:
001122  113703 175612     MOVB @#RBUF, R3         ;CLEAR & RESET
001126  113703 175612     MOVB @#RBUF, R3         ;RBUF & RCSR
001132                  XMT1A:
001132  005003            CLR R3
001134                  XMT1:
001134  033727 175614     BIT @#XCSR, #XMRDY
        000200
001142  001003            BNE X1A         ;BR IF READY
001144  005303            DEC R3
001146  001372            BNE XMT1
001150  000417            BR XEX
001152                  X1A:
001152  112437 175616     MOVB (R4)+, @#XBUF
001156  005300            DEC R0
001160  001364            BNE XMT1A
001162                  X1B:
001162  005000            CLR R0
001164  012703 000001     MOV #1, R3
001170                  X1:
001170  033727 175610     BIT @#RCSR, #RCVDN
        000200
001176  001006            BNE X3          ;BR IF CHAR RCVD
001200  005300            DEC R0
001202  001372            BNE X1
001204  005303            DEC R3
001206  001370            BNE X1
001210                  XEX:
001210  005003            CLR R3          ;SIGNAL ERROR
001212  000205            RTS R5
001214                  X3:
001214  113700            MOVB @#RBUF, R0
```

```
                175612
001220          120027       CMPB R0, #12            ;LF
                000012
001224          001756       BEQ X1B
001226          120027       CMPB R0, #'>
                000076
001232          001366       BNE XEX           ;ERROR IF NOT
001234                    X4:
001234          033727       BIT @#RCSR, #RCVDN
                175610
                000200
001242          001774       BEQ X4            ;WAIT TIL CR IS RCVD
001244          113700       MOVB @#RBUF, R0         ;READ & RESET STATUS BIT
                175612
001250          012703       MOV #1, R3
                000001
001254          000205       RTS R5            ;MSG ACKNOWLEDGED OK

                          ; TRANSMIT CONTENTS OF R0, H.O., THEN L.O. BYTES.
                          ;         NO RESPONSE AWAITED
001256                    XMTNR:
001256          012704       MOV #1, R4
                000001
001262                    XN1:
001262          000300       SWAB R0
001264                    XN3:
001264          033727       BIT @#XCSR, #XMRDY
                175614
                000200
001272          001774       BEQ XN3
001274          110037       MOVB R0, @#XBUF
                175616
001300          005304       DEC R4
001302          001767       BEQ XN1
001304          000205       RTS R5

001306          057032    DEFTY:  .RAD50 /OBJOBJOBJOBJ/
001310          057032
001312          057032
001314          057032
001316                    ARGBL:  .BLKW 6
001332                    NILF:   .BLKW 1 ;NEXT-TO-LAST FLAG
001334                    LCA:    .BLKW 1 ;LAST COLON ADRS - STORAGE
001336                    NTLCA:  .BLKW 1 ;NEXT TO LAST COLON ADRS - STORAGE

                             .NLIST BEX     ;DON'T LIST BINARY EXPANSION

001340                    PROMPT:
001340             015       .ASCII <15><12>
001342             123       .ASCII /STARTING ADDRESS: / <200>
                             .EVEN
001366                    LINBUF:
                             .BLKB 12
                             .EVEN

001402                    MSG1:
001402             015       .ASCII <15><12> /This program transmits a 4K/
001437             040       .ASCII / portion of an Intel format/ <15><12>
001474             110       .ASCII /HEX file (max size=80 blocks) to the/
001540             040       .ASCII / DATA I/<57>/O Programmer./ <15><12>
001567             102       .ASCII /Before calling this program, put desired/
001637             040       .ASCII / PROM module in Programmer./ <15><12>
001674             124       .ASCII /Turn on power, press 'SELECT', then 'F1'/
001744             040       .ASCII / and 'START'./ <15><12>
001763             120       .ASCII /Programmer is then under remote control /
002033             146       .ASCII /from RT11./ <15><12>
002047             101       .ASCII /After '*', type one HEX file name.  After
002120             040       .ASCII / file is read and prompt/<15><12>
002157             144       .ASCII /displayed, type starting Hex address whic
002224             040       .ASCII / must be at 1K boundaries:/ <15><12>
002260             145       .ASCII /e.g. 0, 400, 800, C00, 1000, 1400, etc./
002327             015       .ASCII <15><12>/After (Ret), go to Programmer, wa
002374             040       .ASCII / for completed transmission./<15><12>
002432             145       .ASCII /exit remote control, do your thing, /
```



```
002476             162       .ASCII /reenter remote control./<15><12> /continu
002537             040       .ASCII / with another portion, if desired./
002601             015       .ASCII <15><12><12>
002604                    MSGE1:
002604             110    MSG2:   .ASCII /H/<15>
002606                    MSGE2:
002606             015    MSG3:   .ASCII <15><12><12><7><7>
002613             040             .ASCII /  Communication error/
002640                    MSGE3:
002640             070    MSG4:   .ASCII /83A/<15>
002644                    MSGE4:
002644             072    MSG5:   .ASCII /:00/<15><12>
002651                    MSGE5:
002651             125    MSG6:   .ASCII /U/<15>
002653                    MSGE6:
002653             060    MSG7:   .ASCII /00W/<15>
002657                    MSGE7:
002657             015    MSG8:   .ASCII <15><12>/Incorrectly specified/
002706                    MSGE8:

                050000    HEXBFL=20480.     ;WORDS
                             .EVEN
002706                    HEXBUF:
                122706'   HEXBFE=.+<2*HEXBFL>       ;40K BUFFER


                000000'             .END PROM
```

THE UNIVERSITY OF ASTON IN BIRMINGHAM

The Sumpner Building, 19 Coleshill Street, Birmingham B4 7PB
Tel: 021.359 3611 Ex 559/284

The Department of Electrical and Electronic Engineering

Mr Ken Demers
MS-48
United Technologies Research Center
Silver Lane
East Hartford
Conn. 06108
U S A

ELE/MHA/KA
22 November 1979

Dear Mr Demers

The enclosed note on the pseudo-random number generator routines RAN and RANDU provided in the V.01C Fortran system might be of interest to RT-11 SIG members.

At present I am coding a routine based on the algorithm

$$y_{n+1} = 16807\ y_n \pmod{2^{31}-1}$$

which is considered better.

Yours sincerely

Martin Ackroyd

Notes on the PDP-11 Fortran-Callable Pseudo-Random

Number Generator Routines RAN and RANDU

Martin H. Ackroyd November 1979

The PDP-11 Fortran system provides two Fortran callable library subroutines RAN and RANDU for generating pseudo-random floating point numbers in the range 0 - 1. Apart from the information given in the DEC Fortran user manual (reproduced in appendix I), there is no explicit information available about the routines. RAN is used by teletraffic simulation programs used in the Telecommunications Group, so it seemed worthwhile to find out more about it. By running the machine code of each routine through an 'inverse assembler' program to convert it to assembly language, some points can be noted from examination of its code. Appendix II is the assembly language version of RAN, with comments added by the writer.

1. RANDU is identical to RAN except for the means of returning the floating point result. Calls to RAN and RANDU may be intermingled, the same pair of variables being used as the integer arguments.

2. The multiplicative congruential method is used to transform a pseudo random 32-bit double word into a new pseudo-random double word. The algorithm used is

$$y_{n+1} = a\, y_n \pmod m$$

where y_n = 32-bit number held in I1, I2, the integer Fortran arguments.

$a = (1+2+2^{16}) = 65539$

$m = 2^{31} = 2{,}147{,}483{,}648.$

The period of the sequence should be $2^{31}/4 = 536{,}870{,}912$. (R.P. Chambers, "Random number generation on digital computers", IEEE spectrum, vol. 4, pp. 48-56, 1967).

3. Each routine consists of two quite separate parts: one to implement the multiplicative congruential algorithm and one to convert the double word integer in I1, I2 into a floating number.

4. D.G. Childers and A. Durling, *Digital Filtering and Signal Processing*, West, 1975, give a routine from the IBM Scientific Subroutine Package:

```
      SUBROUTINE RANDU (IX, IY, YFL)

      IY = IX* 65539            (multiply by 2**16+3 and reduce modulo
                                 2**32)
      IF(IY) 2,3,3
```



```
    2 IY = IY+2147483647+1      (reduce modulo 2**31)

    3 YFL = IY                  (convert to floating number in
                                 0 to 2**31)

      YFL = YFL*.4656613E-9     (normalise to range 0 - 1)

      RETURN
      END
```

(Although ostensibly written in Fortran this routine is machine and compiler dependent.) The algorithm is evidently the same as the DEC RANDU, so any information about tests that have been done on the IBM routine may apply to the DEC one.

5. If pseudo random integers are required, I1 could be used. I1 lies in the range 0 to 32767. The use of I2 is not recommended; the period of its low-order bits may be too short (see Chambers, IEEE Spectrum). An integer-only routine could be derived by lopping off the part of RAN which effects the conversion to floating point.

Appendix I - see RT-11 Fortran user manual

Appendix II:

```
.ENABL LC
;  Fortran-callable subroutine RAN from the DEC RT-11 Fortran
;  library.  This assembly language listing produced
;  by processing the linked machine code via a disassembler.
;  Comments added by M H Ackroyd, Aston University, November 1979.
;
;  The Fortran call is:
;                X = RAN(I1,I2)
;  where I1,I2 are the state variables.  Initially, I1 and I2 should
;  be set to zero or saved values froma previous
;  run.
R0=%0
R1=%1
R2=%2
R3=%3
R4=%4
R5=%5
SP=%7
PC=%7
.GLOBL RAN
;
```

```
;*****************************************************************
;  Convert the double integer I1,I2 into a new pseudo random integer
;  by multiplicative congruential method.
;*****************************************************************
;
RAN:    CMPB (R5),#2    ;Correct number of arguments (2) in the
                        ; Fortran CALL?
        BNE ERROR       ;No - use error handling trap
        TST (R5)+       ;no. of args +ve? (?should always be satisfied?)
        MOV (R5)+,R2    ;Address of I1 into R2
        MOV (R5)+,R3    ;Address of I2 into R3
        MOV (R2),R0     ;I1 into R0
        MOV (R3),R1     ;I2 into R1
        BEQ INIT        ;Initialise if I2=0
;
;  Multiply double word in R0,R1 by 2
;
        ASL R1
        ROL R0
;
;  add double word in I1,I2 to double word in R0,R1
;
        ADD (R2),R0
        ADD (R3),R1
        ADC R0
;
;  Add I2 into R0 equivalent to multiplying double word
;  in I1,I2 by 2**16 and reducing the result modulo 2**32
;
        ADD (R3),R0
;  (multiplication by 1+2+2**16, with modulo 2**32 reduction
;    has now been effected)
;
;  If the result is positive, do nothing, else change the sign bit
;  (reduce modulo 2**31)
;
        BPL NEXT
        ADD #100000,R0
        BPL NEXT        ;?Seems a mite superfluous?
;
;  Place new double word in R0,R1 in I1,I2
;
NEXT:   MOV R0,(R2)
        MOV R1,(R3)
;
;*****************************************************************
;  Convert the pseudo-random bit pattern in R0,R1 into a uniformly
;  distributed floating-point number.
;*****************************************************************
;
        MOV #201,R2     ;Initial value for exponent into R2
;
;  Repeatedly multiply the double word in R0,R1 by 2 until overflow occurs
;
AGAIN:  ASL R1
        ROL R0
        BCS DONE        ;overflow occurred?
        DEC R2          ;no - reduce exponent and repeat
        BR AGAIN
```



```
;
;  Use the bits in the place where the exponent
;  goes in the floating number as pseudo random bits in the
;  mantissa.
;
DONE:   CLRB R1         ;clear the lower byte of R1 which will have had
                        ; some of its bits cleared in the normalisation
                        ; process above
        BISB R0,R1      ;put the bits from R1, where the exponent will go
                        ; into the cleared space in R0
        SWAB R1         ;swap bytes so that any bits that have been cleared
                        ; (ie when R0,R1 was shifted more than 8 places)
                        ;are at least significant end of word
        CLRB R0         ;clear space for exponent
        BISB R2,R0      ;clear space for exponent
        SWAB R0         ;put exponent in top half of word
;
;  rotate the double resister R0,R1 to set the exponent in the right
;  position with the most significant bit of R0 being the (+) sign bit
;
        ROR R0
        ROR R1          ;Floating answer is now in R0,R1
        BR RETURN       ;finished!

INIT:   MOV #3,R1       ;initialise at start of pseudo random sequence
        INC R0          ;R0,R1 now hold 2**16+3
        BR NEXT

ERROR:  TRAP 000320

RETURN: RTS PC          ;return to calling program
        .END
```

DEC INPUT

Dear Ken,

I'm writing in response to the article "Simple Bootstrap Loaders for RL01, RK05" in the September 1979 issue of the Minitasker, submitted by T. Miles of the University of Victoria. The procedure for the RL01 bootstrap includes patching block 1 of the volume. Customers should be careful about using this patch, since words 0 through 63 of block 1 contain the bad block replacement table for the RL01 (see the RT-11 System User's Guide, Section 8.2.11.2, "Replacing Bad Blocks").

Version 3 and 3B users can safely modify the replacement table if they use .BAD files rather than replacement blocks to handle bad blocks. They must avoid using DUP with the /R option, and the monitor INITIALIZE command with the /REPLACE option.

For Version 4 of RT-11, the following locations in block 1 are used:

000-201
204-251
700-703
722-773
776-777

The remaining locations are reserved for DIGITAL; the INITIALIZE code may use them regardless of whether .BAD files or replacement blocks are selected to handle bad blocks.

Sincerely,

Joyce D. Chernoff

Joyce Desmarais Chernoff
RT-11 Software Documentation

7 TRACK MAGTAPE DOCUMENTATION

The RT-11 V3B documentation contains incomplete and incorrect information regarding the use of 7 track magtape to transfer information to or from 6 bit character oriented systems. The documentation will be corrected in the next release of RT-11.

TERMINAL PORTS ON THE PDT-11

All of the PDT terminal ports require than the EIA data terminal ready signal (connector pin 20) be asserted by the external equipment in order for the ports to function normally. If this pin is left unconnected or is not asserted, the transmitter CSR ready bit corresponding to that port will always remain cleared, and programs will typically hang waiting for the bit to become set (or for the corresponding interrupt), in order to do their output. (The receiver portion is unaffected by this signal; characters received by the port are made available to the software regardless of the condition of this signal.)

In addition, the ports do not produce a data set ready signal (on pin 6). Therefore, if the external equipment or cable does not supply a DTR signal, the usual technique of tying pins 6 and 20 together on the PDT-side EIA connector will be ineffectual in supplying an asserted DTR signal; the signal must somehow be asserted externally.

Although these conditions should not be a problem with DEC terminals or cabling, they may present problems when dealing with non-DEC terminals or non-standard cabling.

VIRTUAL ARRAYS ON THE 11/23 (FORTRAN IV/RT-11 V2.1)

Use of virtual arrays under the SJ and FB monitors on 11/23 based systems requires that the SJ and FB monitors be patched to run in 28K words; this is a customization patch on page 2-36 of the RT-11 System Generation Manual entitled "Running RT-11 in Less Memory Than Is Available".

Also note that FORTRAN IV/RT-11 V2.1 programs cannot be run as virtual jobs.

MAXIMUM THROUGHPUT ON THE LA120

The LA120 will usually run at its maximum 180 character per second speed whenever it is operated at baud rates of 2400 or greater. There are cases, however, that require the maximum possible baud rate to acheive maximum throughput, e.g., when large numbers of blanks are being printed. For this reason, it is recommended that the LA120 be operated at 9600 baud whenever possible. (When host XON/XOFF support is not available, 1200 baud will usually be the maximum "safe" speed.)

PAST SYMPOSIUM INFORMATION

TAPE COPY

We produced 81 tapes at the Fall 79 DECUS symposium in San Diego. They included 350 files the first of which is an annotated directory (TAPE.DIR). I wish to offer my thanks to both those of you who contributed material for the tape and to those of you who worked to produce the copies.

I am willing to continue making additional copies on request. The rules of this game are simple. A tape in reusable packing along with return postage (not cash or check) is required. Any tape arriving without the reusable packing and return postage will be considered as a very welcome gift. We now have four tapes, the first three will each fit on a 600 ft reel, the last on a 1200 ft reel and all four will fit on a 2400 ft reel. The tapes are 1978 Chicago, 1978 San Francisco, 1979 New Orleans and 1979 San Diego.

Starting with the Spring 80 DECUS symposium we will join with several of the other SIGs and let the DECUS library handle all reproduction and distribution of our tape. We will continue to accept submissions (on floppy disk and mag tape) and build our master tape at the symposium. LUGs will recieve a free copy of the tape to distribute as they see fit. Others may order copies from the DECUS library for a small fee ($20 for Fall 79 tapes). After some period of time copies of the tape will be available via the regular DECUS library distribution for a higher fee.

Tape distribution through the DECUS library adds some formal requirements. All contributors MUST sign a release for the material they submit. No proprietary or licensed software in whole or part is allowed. Also, a short written description of the material is required (machine readable form will be a great aid to your master tape builder). An excellent example of a written description appears as the first entry in the annotated directory of the Fall 79 DECUS RT SIG tape (the file DECUS.TXT on the tape).

For the next symposium in Chicago I need some help for building the master tape. I am looking for a local shop, preferably near the symposium hotel (Hyatt Regency O'Hare), that would be willing to permit me to use their equipment for building our RT SIG master tape. I'll need at minimum a system having RX01s or RX02s and 9-track mag tape (2400 ft reels). A larger disk such as RK05, RL01 or RL02 would be very desirable. If necessary I can bring my own bootable RK05 disk cartridge. Actual use of the system would be during the evening hours. If anyone is willing to volunteer their shop and services please contact me.

Nick Bourgeois / 1736
Sandia Laboratories
P. O. Box 5800
Albuquerque, NM 87185
(505) 844-8088

ANNOTATED DIRECTORY OF THE 1979 FALL RT-11 DECUS TAPE
SAN DIEGO, CA.
****

This file describes the contents of the RT-11 development group's submissions to the DECUS swap tape for Fall 1979 DECUS in San Diego.

<2042 BLKS TOTAL>

DECUS.TXT	This file
TECO kit:	
TECO.SAV	The runnable TECO program
CRTASM.COM	Assembly instructions for scope module
TECLNK.COM	Link instructions for TECO
CRTPRE.MAC	Supplied DEC-standard scope module prefix
CRTRUB.MAC	and code
CRTRUB.OBJ	Object modules to recreate TECO
SCREEN.OBJ	
SCRINS.OBJ	
SCROLL.OBJ	
TECO.OBJ	
TECOIO.OBJ	
TIODCD.OBJ	
TIOEIO.OBJ	
TIOENC.OBJ	
TIOFET.OBJ	
TIOIAS.OBJ	
TIOINI.OBJ	
TIORFS.OBJ	
LOCAL.TEC	TECO macros, compressed
SEARCH.TEC	
SQU.TEC	
TYPE.TEC	
VTEDIT.TEC	
TECO.INI	
LOCAL.TES	TECO macros, uncompressed
SEARCH.TES	
SQU.TES	
TECO.TES	
TECOIN.TES	
TYPE.TES	
VTEDIT.TES	
TECO.DOC	The latest TECO manual
TECORN.DOC	Release notes for TECO V35
VTEDIT.DOC	VTEDIT manual

Othello game playing program:

```
OTHELO.COM      Build instructions for OTHELLO
OTHELO.FOR      Sources
VTHELO.FOR
OLOGIC.FOR
OTHELO.SAV      Runnable OTHELLO game
VTHELO.SAV      Virtual version (for XM under RT-11 V4)
```

Master-mind game playing program:

```
MMIND.COM       Build instructions
MMIND.FOR       Sources
VMIND.FOR
MLOGIC.FOR
MTEXT.MAC
EXTEND.OBJ      A binary module
MMIND.SAV       Runnable Master-mind game
VMIND.SAV       Virtual version (for XM under RT-11 V4)
MMIND.REL       Foreground version
```

Plane landing game program:

```
OPLANE.SAV      Ordinary runnable version
FPLANE.REL      Foreground version
VPLANE.SAV      Virtual version (for XM under RT-11 V4)
```

Other userful items:

```
BANNER.SAV      Prints gothic or Roman letter banners
LIFE.SAV        Life program
LIFE.TXT         and explanation
MTYSET.MAC      MTYSET example from RT Support manual:
MTYSET.SAV       auto-baud and terminal type determination
SIGNON.MAC      Useful start-up program to require date/time
SIGNON.SAV
RTMON.MAC       Foreground system status display (V4 only)
TTYSET.SAV      Set terminal characteristics (contians HELP)
VDT.OBJ         Virtual/multi-terminal version of ODT
VM.MAC          VM handler for memory over 28K (V4 only)
SY.MAC          System parameter handler (V4 only)
```

****

JOHN R. BARR, PHD
ASSISTANT PROFESSOR
COMPUTER SCIENCE DEPARTMENT
UNIVERSITY OF MONTANA
MISSOULA
(406) 243-2883

PASCAL
DOCUMENTATION TO BE IN
SYMPOSIUM PROCEEDINGS

```
 10-Dec-79
RUNTME.PAC   112 13-Nov-79      NPASL .OBJ    24 13-Nov-79
CPASL .OBJ     7 13-Nov-79      SELECT.OBJ     6 13-Nov-79
SELECT.SAV    12 13-Nov-79      UTIL  .PAC    18 13-Nov-79
MISC  .PAC    51                NBUGS .RNO    27 22-Oct-79
NEWS  .RNO    37 22-Oct-79      PASGUI.RNO    84 22-Oct-79
 10 Files, 378 Blocks
```



```
 10-Dec-79
PASS1 .PAS   196 02-Dec-79      PASS1 .OBJ   122 02-Dec-79
PASS1 .SAV    74 02-Dec-79      NPASL .OBJ    24 02-Dec-79
CPASL .OBJ     7 13-Nov-79
 5 Files, 423 Blocks
 10-Dec-79
NPASL .OBJ    24 02-Dec-79      CPASL .OBJ     7 13-Nov-79
PASS2 .PAS   204 06-Dec-79      PASS2 .OBJ   119 06-Dec-79
PASS2 .SAV    86 06-Dec-79
 5 Files, 440 Blocks
```

****

NICK BOURGEOIS / 1736
SANDIA LABORATORIES
PO BOX 5800
ALBUQUERQUE, NM 87185
(505) 844-8088

BASIC CROSS REFERENCE PROGRAMS
SELECTED RT-11 V03B PATCHES
FORTRAN V2.1 PATCHES
BASIC V2 PATCHES

```
 10-Dec-79
CROSS .BAS    22 10-Dec-79      CROSS0.BAS     7 10-Dec-79
CROSS1.BAS    13 10-Dec-79      CROSS2.BAS     8 10-Dec-79
CROSSV.BAS    23 10-Dec-79      CROSS .RNO     6 10-Dec-79
CROSS .DOC     7 10-Dec-79      MON016.BAT     3 10-Dec-79
MON015.BAT     7 10-Dec-79      MON017.BAT     7 10-Dec-79
MON018.BAT     5 10-Dec-79      MON019.BAT     2 10-Dec-79
MON020.BAT    .9 10-Dec-79      FXV02 .MAC     1 10-Dec-79
PAT10 .COM     2 10-Dec-79      SRC006.BAT     4 10-Dec-79
UTL013.BAT     4 10-Dec-79      MON021.BAT     3 10-Dec-79
PAT11 .COM     2 10-Dec-79      PAT12 .COM     2 10-Dec-79
PAT06 .MAC     1 10-Dec-79      PAT06 .FOR     1 10-Dec-79
FXV11 .MAC     1 10-Dec-79      PAT11 .MAC     1 10-Dec-79
PAT07 .FOR     1 10-Dec-79      PAT09A.MAC     1 10-Dec-79
PAT11 .FOR     1 10-Dec-79      PAT12 .MAC     1 10-Dec-79
PAT12A.FOR     1 10-Dec-79      PAT12B.FOR     1 10-Dec-79
PAT09B.MAC     1 10-Dec-79      PAT13 .MAC     2 10-Dec-79
PAT01 .MAC     1 10-Dec-79      UTL001.BAT     3 10-Dec-79
UTL002.BAT     2 10-Dec-79      MON006.BAT     4 10-Dec-79
SRC001.BAT     2 10-Dec-79      SRC002.BAT     2 10-Dec-79
BAS001.COM     2 10-Dec-79      BAS001.TEC     1 10-Dec-79
MSC002.MAC     1 10-Dec-79      UTL010.MAC     1 10-Dec-79
MSC002.COM     2 10-Dec-79      UTL010.COM     1 10-Dec-79
MON013.BAT     3 10-Dec-79      MON014.BAT     5 10-Dec-79
UTL007.BAT     2 10-Dec-79      UTL011.BAT     2 10-Dec-79
UTL008.BAT     3 10-Dec-79      UTL012.BAT     3 10-Dec-79
MON001.BAT     2 10-Dec-79      PATCH .COM     2 10-Dec-79
MON005.BAT     4 10-Dec-79      MON007.BAT     2 10-Dec-79
MON004.BAT     6 10-Dec-79      MON003.BAT     5 10-Dec-79
MSC001.BAT     2 10-Dec-79      SRC003.BAT     4 10-Dec-79
MON010.BAT     3 10-Dec-79      UTL003.BAT     2 10-Dec-79
UTL004.BAT     2 10-Dec-79      UTL005.BAT     2 10-Dec-79
UTL006.BAT     2 10-Dec-79      MON011.BAT     8 10-Dec-79
MON012.BAT     2 10-Dec-79      MON009.BAT     3 10-Dec-79
MON002.BAT     7 10-Dec-79      PAT13 .FOR     1 10-Dec-79
FXV08 .MAC     1 10-Dec-79      PAT09 .FOR     2 10-Dec-79
```

```
PAT10 .MAC      1 10-Dec-79       PAT14 .MAC      1 10-Dec-79
PAT14 .FOR      1 10-Dec-79       PAT15A.MAC      1 10-Dec-79
PAT15 .FOR      1 10-Dec-79       PAT16 .MAC      1 10-Dec-79
PAT16 .FOR      1 10-Dec-79       PAT17 .MAC      1 10-Dec-79
FXV17 .MAC      1 10-Dec-79       FXV21 .MAC      1 10-Dec-79
PAT21 .FOR      1 10-Dec-79       PAT22C.MAC      1 10-Dec-79
PAT17 .FOR      2 10-Dec-79       PAT14 .COM      2 10-Dec-79
PAT16 .COM      2 10-Dec-79       MON024.BAT      6 10-Dec-79
UTL015.BAT      2 10-Dec-79       MSC004.BAT      1 10-Dec-79
PAT01 .FOR      1 10-Dec-79       PAT02 .MAC      1 10-Dec-79
PAT02 .FOR      1 10-Dec-79       PAT03 .MAC      1 10-Dec-79
PAT03 .FOR      1 10-Dec-79       PAT05A.MAC      1 10-Dec-79
PAT05B.MAC      1 10-Dec-79       PAT05 .FOR      1 10-Dec-79
PAT08 .MAC      2 10-Dec-79       PAT08A.FOR      1 10-Dec-79
PAT08B.FOR      1 10-Dec-79       FXV01 .MAC      1 10-Dec-79
PAT10 .FOR      2 10-Dec-79       PAT18 .MAC      1 10-Dec-79
PAT18A.FOR      1 10-Dec-79       PAT18B.FOR      1 10-Dec-79
PAT19 .MAC      1 10-Dec-79       PAT20 .MAC      1 10-Dec-79
FXV20 .MAC      1 10-Dec-79       PAT20A.FOR      1 10-Dec-79
PAT20B.FOR      1 10-Dec-79       PAT21 .MAC      1 10-Dec-79
PAT22D.MAC      1 10-Dec-79       FXV22 .MAC      1 10-Dec-79
PAT22 .FOR      1 10-Dec-79       PAT15C.MAC      1 10-Dec-79
PAT22A.MAC      1 10-Dec-79       PAT22B.MAC      1 10-Dec-79
PAT01 .COM      2 10-Dec-79       PAT13 .COM      2 10-Dec-79
PAT08 .COM      2 10-Dec-79       PAT09 .COM      2 10-Dec-79
PAT15 .COM      3 10-Dec-79       PAT18 .COM      2 10-Dec-79
PAT21 .COM      2 10-Dec-79       PAT19 .COM      2 10-Dec-79
PAT22 .COM      3 10-Dec-79       PAT03 .COM      2 10-Dec-79
PAT02 .COM      2 10-Dec-79       FXV03 .MAC      1 10-Dec-79
PAT05 .COM      2 10-Dec-79       PAT17 .COM      2 10-Dec-79
PAT07 .COM      2 10-Dec-79       PAT06 .COM      2 10-Dec-79
MON025.BAT      2 10-Dec-79       PAT07 .MAC      1 10-Dec-79
MSC005.BAT      2 10-Dec-79       PAT04 .COM      2 10-Dec-79
PAT04 .MAC      1 10-Dec-79       PAT04 .FOR      1 10-Dec-79
MSC006.BAT      2 10-Dec-79       MSC007.COM      2 10-Dec-79
MSC007.MAC      1 10-Dec-79       PAT20 .COM      2 10-Dec-79
MON026.BAT      5 10-Dec-79       MON027.BAT      2 10-Dec-79
SRC008.BAT      3 10-Dec-79       UTL016.BAT      2 10-Dec-79
MON028.BAT      2 10-Dec-79       UTL018.BAT      3 10-Dec-79
UTL020.BAT      2 10-Dec-79       PATA1 .MAC      1 10-Dec-79
PATA2 .MAC      1 10-Dec-79       PATB  .COM      2 10-Dec-79
PATA  .COM      2 10-Dec-79       PATB1 .MAC      1 10-Dec-79
PATB2 .MAC      1 10-Dec-79       PATC  .COM      2 10-Dec-79
PATC1 .MAC      1 10-Dec-79       PATC2 .MAC      1 10-Dec-79
PATD  .COM      2 10-Dec-79       PATD1 .MAC      1 10-Dec-79
PATD2 .MAC      1 10-Dec-79       PATE  .COM      2 10-Dec-79
PATE1 .MAC      1 10-Dec-79       PATE2 .MAC      1 10-Dec-79
PATF  .COM      2 10-Dec-79       PATF2 .MAC      1 10-Dec-79
PATF1 .MAC      1 10-Dec-79       PATG  .COM      2 10-Dec-79
PATG1 .MAC      1 10-Dec-79       PATG2 .MAC      1 10-Dec-79
PATH  .COM      2 10-Dec-79       PATH1 .MAC      1 10-Dec-79
PATH2 .MAC      1 10-Dec-79       PATI  .COM      2 10-Dec-79
PATI1 .MAC      1 10-Dec-79       PATI2 .MAC      1 10-Dec-79
PATJ  .COM      2 10-Dec-79       PATJ1 .MAC      1 10-Dec-79
PATJ2 .MAC      1 10-Dec-79       PATJ3 .MAC      1 10-Dec-79
PATJ4 .MAC      1 10-Dec-79       PATL  .COM      2 10-Dec-79
```



```
PATL1 .MAC      1 10-Dec-79       PATL2 .MAC      1 10-Dec-79
PATM  .COM      2 10-Dec-79       PATM1 .MAC      1 10-Dec-79
PATM2 .MAC      1 10-Dec-79       PATN  .COM      2 10-Dec-79
PATN1 .MAC      1 10-Dec-79       PATN2 .MAC      1 10-Dec-79
PATO  .COM      2 10-Dec-79       PATO1 .MAC      1 10-Dec-79
PATO2 .MAC      1 10-Dec-79       DIRPAT.COM      1 10-Dec-79
 194 Files, 436 Blocks
```

****

M. LEVINE / CODE 3513 FORTRAN CROSS REFERENCE PROGRAM
NAVAL WEAPONS CENTER ADVENTURE AND DUNGEON GAMES
CHINA LAKE, CA 93555
(714) 939-2417

```
 10-Dec-79
INDEX .DOC     20 10-Dec-79       INDEX .HLP      3 10-Dec-79
INDEX .COM      2 10-Dec-79       IDXLST.COM      1 10-Dec-79
IDXXFR.COM      3 10-Dec-79       IDX453.COM      1 10-Dec-79
IDX460.COM      1 10-Dec-79       IDX353.BAT      1 10-Dec-79
IDX360.BAT      1 10-Dec-79       INDEX .MAP      7 10-Dec-79
INDEX .SAV     24 10-Dec-79       INDEX .MAC      5 10-Dec-79
OUTPUT.MAC     10 10-Dec-79       STORE .MAC     10 10-Dec-79
RAD50 .MAC      3 10-Dec-79       IOLINE.MAC     12 10-Dec-79
GET   .MAC     14 10-Dec-79       OPNCLO.MAC     13 10-Dec-79
IOCHR .MAC      4 10-Dec-79       EVAL  .MAC     23 10-Dec-79
TABLE .MAC      5 10-Dec-79       LINETP.MAC     10 10-Dec-79
SUPER .MAC     13 10-Dec-79       IMPURE.MAC      4 10-Dec-79
BUFFER.MAC      2 10-Dec-79       DUNGEO.SAV    216 10-Dec-79
DUNGEO.DOC     18 10-Dec-79       DTEXT .DOC      0 10-Dec-79
DTEXT .DAT    383 10-Dec-79       DINDX .DAT     20 10-Dec-79
ADVENT.DOC      6 10-Dec-79       ADVENT.HLP      8 10-Dec-79
TEXT  .TXT    147 10-Dec-79       ADVENT.SAV     92 10-Dec-79
 34 Files, 1082 Blocks
```

****

GREG ADAMS STRUCTURED FORTRAN PREPROCESSOR
CFB OTTAWA (N)
OTTAWA ONTARIO CANADA
N11-03123

```
 10-Dec-79
FLXMIN.SAV     77 05-Oct-79       FLXGEN.COM      2 11-Oct-79
FLXIO .FLX     18 03-Oct-79       FLXANL.FLX     33 03-Oct-79
FLXLST.FLX     60 03-Oct-79       FLECS .FLX     54 03-Oct-79
FLXGOS.FLX     16 03-Oct-79       FLXGO .FLX     16 31-Jul-79
FLXSUP.FLX     14 05-Oct-79       TIMEIN.FLX      7 23-Jul-79
DATEIN.FLX      5 23-Jul-79       DATIME.FLX      4 14-Sep-79
FLXGEN.DOC     10 04-Dec-79
 13 Files, 316 Blocks
```

****

SOURCES FOR THE RT-11 DEVELOPMENT GROUPS SUBMISSION

```
 10-Dec-79
RTMON .MAC     32 06-Nov-79       MTYSET.MAC     17 01-Nov-79
SIGNON.MAC      4 21-Nov-79       OTHELO.FOR     37 20-Nov-79
SPEED .FOR      1 12-Nov-79       POWER .FOR      1 12-Nov-79
ALT   .FOR      1 12-Nov-79       PANEL .FOR      4 12-Nov-79
ATT   .FOR      1 12-Nov-79       VERTCL.FOR      1 12-Nov-79
VERT  .FOR      1 12-Nov-79       COURSE.FOR      1 12-Nov-79
BAR   .FOR      1 12-Nov-79       CLIMB .FOR      1 12-Nov-79
CURMOV.FOR      2 12-Nov-79       LEFT  .FOR      1 13-Nov-79
RIGHT .FOR      1 13-Nov-79       LEVEL .FOR      1 13-Nov-79
PAGE1 .FOR      2 20-Nov-79       PAGE2 .FOR      3 20-Nov-79
MTEST .FOR      9 26-Nov-79       PLANE .FOR     11 03-Dec-79
ERASE .FOR      1 21-Nov-79       VPLANE.COM      1 04-Dec-79
LOGIC .FOR     11 03-Dec-79       OPLANE.COM      1 23-Nov-79
FPLANE.COM      1 23-Nov-79       VPLANE.FOR      1 03-Dec-79
CPLANE.COM      1 04-Dec-79       MTEXT .MAC      2 03-Dec-79
MMIND .FOR     10 04-Dec-79       VMIND .FOR      1 04-Dec-79
MLOGIC.FOR     10 04-Dec-79       VTHELO.FOR      1 04-Dec-79
OLOGIC.FOR     36 04-Dec-79       OTHELO.COM      1 04-Dec-79
MMIND .COM      1 04-Dec-79
 37 Files, 212 Blocks
```

****

CARL D. LOWENSTEIN
MARINE PHYSICAL LAB
BLDG. 106, NOSC
SAN DIEGO, CA 92152

(714) 294-3678

RATIONALIZED FORTRAN PROCESSOR, TAKEN FROM KERNIGHAN & PLAUGER, SOFTWARE TOOLS DISTRIBUTION TAPE. INTERFACED TO RT-11 FILE SYSTEM THROUGH COMMAND STRING INTERPRETER.

```
RATFOR.SAV     61 21-Nov-79       RATFOR.FOR    367 21-Nov-79
```

USER APPLICATION WORKSHOP

Once again the workshop session was a great success. There were four informal presentations followed by lots of interactive exchange from the floor. The informal presentations were:

Greg Adams -- An Air Traffic Controll Simulator.

Randy Kaplan -- Using MiniMINCs for Data Collection.

Ned Rhodes -- A Data Acquisition System.

Mark Servello -- Monitoring Pulmonary-Lung Functions.

Nick Bourgeois

UPCOMING SYMPOSIUM INFORMATION

There will be an RT-11/PDP-11 assembly language tutorial scheduled during the Spring, 1980 Symposium. Art Hermes will discuss addressing modes and Ray Streckbein will discuss program requests. This session is intended for those not experienced in PDP-11 assembly language programming. Those planning to attend should read "Fundamentals of PDP-11 Programming by R. W. Southern. This self-paced book is availible at the DEC Training Center Bookstore in Bedford, Mass.

RT-11 PATCHES

MDUP FAILURE ON RL01 (RT-11 V3B)

After booting a mastape, MDUP(.MT or .MM) will hang upon any attempt to access an RL01. The following patch will correct this problem.

```
.R PATCH

FILE NAME--
*MDUP.MT or MDUP.MM<RETURN>
*160/    0        46464<LINE FEED>
162/     0        340<RETURN>
*E
```

State Electricity Commission of Victoria

Herman Research Laboratory
Howard Street, Richmond, Vic. 3121
Telephone 429 1511
Telex 31153
Cable & Telegraphic Address: Electrocom Melbourne

Our Reference
rc

Dear Ken

The enclosed copy of an SPR response from DEC has been received in response to my SPR which you published in Volume 5, No 4, of the Mini-Tasker.

Since the fault reported was generated by a whole class of programs similar to that given, I believe that the meaning of the response is:

> 'The FORTRAN compiler does not support mixed-mode arithmetic when the in-line code option is used.'

As this is a significant restriction on the usefulness of the compiler, I believe it should be widely publicised.

Yours sincerely

Geoffrey R Grinton
SCIENTIFIC OFFICER

D I G I T A L

SPR RESPONSE

SUBJECT: SPR NUMBER 11-25412

	SYSTEM	SYS VERS	PRODUCT	PROD VERS	COMPONENT
SOFTWARE:	RT-11	3B	FORTRAN-IV	2.1	COMPILER

PROBLEM STATEMENT

Compiler generates incorrect code when inline code option is used.

RESPONSE

This will be fixed in the next release of FORTRAN IV, in the meantime please either use threaded code option or use a real dummy variable or apply FLOAT on the integer variable.

COBOL FOR RT-11

RT-11 and TSX users now have the option of COBOL programming with the announcement by S&H Computer Systems, Inc. of COBOL-PLUS, a full ANS '74 Standard COBOL compiler and run-time system which will run on any PDP-11 or LSI-11.

Designed to run programs much larger than physical memory, the COBOL-PLUS compiler allows automatic virtual memory program segmentation by a "least-recently-used" technique to keep the most active set of segments in memory. COBOL-PLUS run-time routines for OPEN-CLOSE, input-output, and ISAM are also managed in this fashion.

COBOL-PLUS also provides interactive screen-handling capabilities through the use of positional ACCEPT and DISPLAY statements. Powerful input-output facilities are provided by support of sequential, relative, and indexed-sequential files. Indexed access files are dynamically self-reorganizing. Record locking facilities are provided to synchronize access to shared relative and indexed organization files. Disk sorts are performed by the highly efficient and field-proven RTSORT facility.

A great deal of programming flexibility is provided to the COBOL-PLUS programmer by support of a variety of data types, including full support of usage type DISPLAY and COMPUTATIONAL data items. DISPLAY type items may have leading, separate, trailing separate, or trailing non-separate signs and may be up to 18 digits long. COMPUTATIONAL data items may be 1, 2, or 4 words long.

COBOL-PLUS implements most of the ANS '74 Level 2 language features. Included are the COMPUTE statement; the OCCURS DEPENDING ON clause; level 66, 77, and 88 data items; complex conditional expressions; and the PERFORM VARYING and MOVE CORRESPONDING statements.

For more information contact H. Dale Coleman, S&H Computer Systems, Inc., 1027 17th Ave. South, Nashville, Tennessee, (615) 327-3670.

Software Dynamics has a communications package to enable telecommunications with Data Terminal Systems series 500 Electronic Cash Registers. This package uses Binary Synchronous Communications (BSC) for communication between the host and cash register(s). Requires RT-11 F/B monitor, V3B or later and either a DUV11, DU11 or DUP11 interface.

For further information contact:

Software Dynamics, Inc.
1000 Yale Avenue
Wallingford, CT 06492
(203) 269-4998

?&!®

Warner-Eddison Associates, Inc.
186 Alewife Brook Parkway, Cambridge, Massachusetts 02138 (617) 661-8124

Dear Mr. Demers:

Enclosed is information about INMAGIC™ data management software. We think this will be of interest to DECUS Newsletter readers.

Warner-Eddison Associates developed INMAGIC™ for its work in organizing libraries and information centers. There has been such a demand for software with applications like those of INMAGIC™ that we are now selling the software itself.

If you have any questions about INMAGIC™ or about Warner-Eddison's services, please feel free to call.

Sincerely yours,

Alexandra Baker
Development Division

INMAGIC™ SOFTWARE

AVAILABLE FOR DEC PDP-11s

INMAGIC™ data management software from Warner-Eddison Associates is now available for use on DEC PDP-11s under the RT-11, RSX or RSTS/E operating systems.

INMAGIC™ allows users to build, update and search their own databases for information records describing technical reports, books, proposals, vendor catalogs, client and personnel files.

Features include: unlimited number and length of fields within each record and unlimited subfields within each field; searching by Boolean, AND, OR, and NOT logic; sorting letter-by-letter, word-by-word, numerically or chronologically. Library cards, mailing lists and labels, or catalogs are possible print formats. Users may also define their own report styles. INMAGIC™ is written in FORTRAN IV.

The INMAGIC™ User's Manual describes INMAGIC™'s five working environments: SELECT, SORT, MAINTAIN, DEFINE, and AUDIT, and provides system documentation.

DECUS

DIGITAL EQUIPMENT COMPUTER USERS SOCIETY
ONE IRON WAY, MR2-3/E55
MARLBORO, MASSACHUSETTS 01752

MOVING OR REPLACING A DELEGATE?

Please notify us immediately to guarantee continuing receipt of DECUS literature. Allow up to six weeks for change to take effect.

( ) Change of Address
( ) Delegate Replacement

DECUS Membership No.: _____

Name: _____

Company: _____

Address: _____

_____

State/Country: _____

Zip/Postal Code: _____

Mail to: DECUS - ATT: Membership
One Iron Way, MR2-3
Marlboro, Massachusetts 01752 USA

Affix mailing label here. If label is not available, print old address here. Include name of installation, company, university, etc.