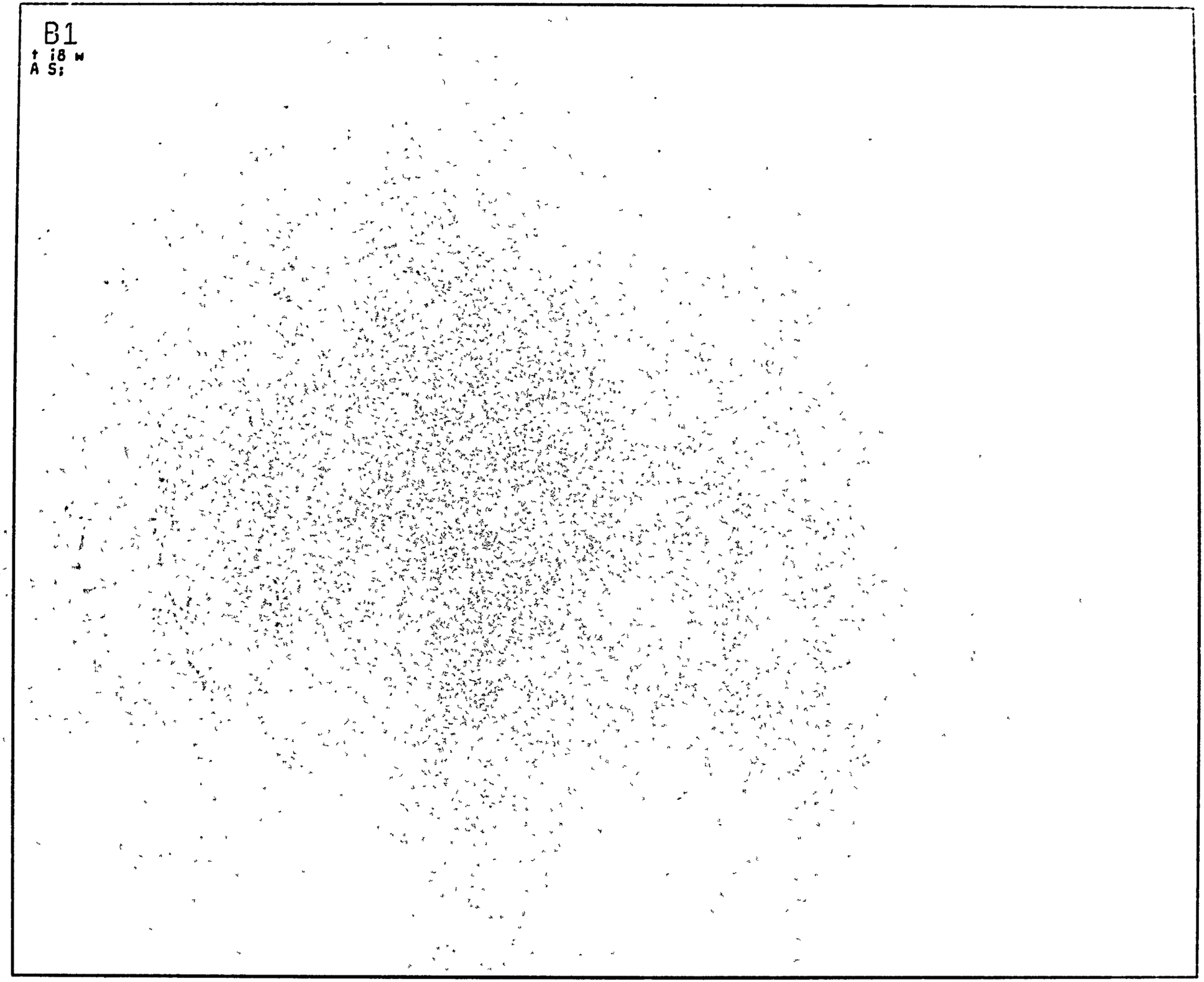
B1
t iB w
A S;

```
                .TITLE  CZNIA  - RSX ONLINE NIE
                .IDENT  /01.01/
                .ENABL  LC
                .REM +

                        IDENTIFICATION
                        ---------- ---

PRODUCT CODE:   AC-FF94A-MC

PRODUCT NAME:   CZNIAA0  RSX ONLINE NETWORK INTERCONNECT EXERCISOR

PRODUCT DATE:   JUNE 1985

MAINTAINER:     MERRIMACK DIAGNOSTIC ENGINEERING

AUTHOR:         ADAM KOJNOK
```


REVISION HISTORY:

REV	DATE	AUTHOR	REASON
A	10-JUN-85	ADAM KOJNOK	ORIGINAL ISSUE, RSX ONLINE NETWORK INTERCONNECT EXERCISOR

RSX ONLINE NETWORK INTERCONNECT EXERCISER

USER'S GUIDE

ISSUED BY: Adam J. Kojnok

AC-FF94A-MC

REV A

TABLE OF CONTENTS

1.0 INTRODUCTION

The Network Interconnect Exerciser (NIE) provides online diagnostic exerciser for Ethernet networks. The NIE determines node ability on the network and provides the operator with error analysis. Node installation, verification, and problem isolation can be performed using the NIE.

The NIE is device independent. The NIE will run with any Ethernet device that can be accessed using the DLX interface mechanism. Also, the NIE may be run on RSX-11M+, 11M and 11S systems.

NOTE:

The DLX (Direct Line Access) interface was designed to enable user programs to use direct, high-level interface to a physical line protocol, bypassing the higher level layers of DECnet. The RSX DEUNA/DELUA QIO Driver uses this interface to communicate with user programs.

The RSX online NIE may be used in the following configurations:

With DECnet
The NIE runs concurrently with DECnet software. The NIE uses two NI protocol types: loopback and remote console. The NIE interfaces to DECnet via the Direct Line Access (DLX) functionality of DECnet. NIE communicates with the DECnet Ethernet driver via the NX: pseudo device driver.

Without DECnet
The RSX online NIE can be run using the RSX stand alone DEUNA driver. This configuration was provided for systems wothout DECnet. DECnet may be present, however, the Ethernet device to be used may not be accessed via DECnet. The NIE was developed using the RSX stand alone DEUNA driver which implements the DLX interface used by DECnet.

Node table on disk or in memory
The node table may be built on disk into a temporary file. Using disk, the node,table may have up to 1024(10) entries. The maximum size of a memory resident node table is around 70(10) node entries.

2.0 BUILDING THE NIE

NOTE

DEV:[USER-UIC] refers to device and UIC where the files were copied to from the distribution kit.

2.1 NIE BUILD PARAMETERS

[1,1]NETLIB.MLB
If the network macro library is not present in account [1,1], the DECnet interfaces are not built into the NIE. The absence of this file will define NONETL=0 in prefix file D.MAC. If NONETL is defined, the network related code will not be included.

M$$NET
If not defined in [11,10]RSXMC.MAC, the DECnet interfaces are not built into the NIE.

R$$11S
If defined in [11,10]RSXMC.MAC, all code concerned with disk storage is conditionalized out.

2.2 PROCEDURE TO BUILD THE NIE

NOTE - The build command file uses the following non-NIE files:

```
        SY:[11,10]RSXMC.MAC   ; Required RSX macro prefix file!!!
              M$$NET is in this file when defined
              R$$11S is in this file when defined
        SY:[1,1]NETLIB.MLB    ; Not required.  If not there no DECnet
                              ; interface code will be generated.
        SY:[1,54]RSX11M.STB   ; Required to build NIE
```

Magnetic media contains the following files:

```
   CZNIA.SEQ                  ; NIE documentation file.

   NIEBLD.CMD                 ; NIE build command file, used to build
                              ;     the NIE.
   DEUNA.MLB                  ; DLX/Ethernet macros (required for both
                              ;     NIE and XE: Driver

;
; The following are NIE macro files
;
   NIESUP.MAC                 ; NIE support task code
   NIESUB.MAC                 ; NIE task subroutines
   NIECMD.MAC                 ; NIE task command processing routines
   NIEPRS.MAC                 ; NIE command parser modules and data
                              ;     structures.

;
; The following are XE: (DEUNA/DELUA) Driver files.
;
```

```
225
226
227
228                     XEDRV.MAC                   ; XE: Driver code
229                     XETAB.MAC                   ; XE: Driver RSX Database
230                     UNAMC.TSK                   ; XE: Driver DEUNA ECO microcode file
231                     LUAMC.TSK                   ; XE: Driver DELUA ECO microcode file
232
233
234                   Building command sequence:
235                   ---------------------- -
236
237                   NOTE
238                   DEV: for Magtape = MT:
239                   DEV: for DL      = DLx:[3,3]
240
241                   >MOU DEV:NIE                  ; Mount distributions kit
242                   >PIP /NV=DEV:CZNIA.SEQ                ; Copy documentation file
243                   >PIP /NV=DEV:NIEBLD.CMD               ; Copy NIE build command file from distro kit
244                   >@NIEBLD                      ; Do the build
245
246                      Answer questions asked by build command file.
247
248                   *** NOTE ***
249                   See APPENDIX A for sample build print out!
250
251
252
253                 2.3  INSTALLING NIE
254
255
256                 NIEBLD.CMD will do the installation at the end of the build.  However,  if
257                 the system is re-started, the folowing must be installed:
258
259                        MCR>INS [USER-UIC]NIE
260                        MCR>INS [USER-UIC]NI1
261
262                 NOTE
263                        The  XE:   driver  must  be  installed  (LOADED)  if   a
264                        non-DECnet system is being used.
265
266
267
268                 3.0  RSX QIO UNA DRIVER BUILD
269
270
271                 The following command lines will assemble the driver code and  the  driver
272                 RSX database:
273
274                 Assemble driver code -
275
276                   [11,24]xedrv,[11,34]xedrv/cr/-sp=[1,1]exemc/ml,dev:[user-uic]d,-
277                   deuna/ml,[11,10]rsxmc/pa:1,dev:[user-uic]xedrv
278
279                 Assemble driver RSX data base -
280
281                   [11,24]xetab,[11,34]xetab/cr/-sp=[1,1]exemc/ml,-
```

deuna/ml,[11,10]rsxmc/pa:1,dev:[user-uic]xetab

3.1 RSX QIO UNA DRIVER PARAMETERS

The following parametes are to be defined in the sy:[11,10]RSXMC.MAC file. These parameters will be defined in the D.MAC prefix file also. The values for these parameters in the D.MAC file will be set to the default. These parameters will be redefined by their definitions in sy:[11,10]RSXMC.MAC.

NOTE

Normally, there should be no reason to define/redefine these parameters for building the XE Driver! The parameters are presented here for documentation purposes.

The following are default definitions:

```
U$$NCT=1                        ; Number of controller on system
U$$NPC=8.                       ; Number of ports per controller
U$$NRS=8.                       ; Number of receive ring entries
U$$NTS=3.                       ; Number of transmit ring entries
```

3.2 BUILDING XE: THE DRIVER

The DEV:[USER-UIC]NIEBLD.CMD command file will generate DEV:[USER-UIC]XEDRVBLD.CMD command file. This file contains the TKB lines to build the XE: driver.

3.3 INSTALLING (LOADING) THE DRIVER

RSX11M-PLUS

```
  MCR>CON SET VEC=vvv CSR=xxxxxx        ; Set vector and CSR
  MCR>CON ONLINE XEA,XE0:               ; Set unit and controller online
  MCR>INS UML                           ; Install DEUNA/DELUA microcode
                                        ; .. loader support task
```

RSX11M

```
  MCR>LOA XE:/PAR=PARTION               ; Load driver
  MCR>INS UML                           ; Install DEUNA/DELUA microcode
                                        ; .. loader support task
```

4.0 NIE COMMANDS

NOTES

Notation in left and right square brackets is optional.

i.e. - M[ESSAGE] /TY[PE]=ASCII<CR>

is equivalent to

M /TY=ASCII<CR>

Also, spelling errors in optional text are ignored.

i.e. - MEZZAGE /TYTE=ASCII<CR>

will be parsed to mean

M /TY=ASCII<CR>

4.1 HELP

A help file showing a summary of NIE commands can be displayed by typing "Help" or "H" in response to the NIE prompt. For example:

```
NIE>he[lp]
or
NIE>?
```

4.2 EXITING THE NIE

The Exit command exits the NIE task. There are no switches or qualifiers for the Exit command. The format is:

```
NIE>e[xit]
```

4.3 BUILDING A NODE TABLE

The current node table is a data structure which the NIE uses to determine which nodes are available for testing. When the node table is saved on a disk file, it is saved in ASCII format to allow the node table data to be examined off line. In addition, the saved node table disk file includes the logical node names.

Data in the node table includes the following:

1. Current node physical address.

2. Default node physical (hardware) address.

3. NIE assigned logical node address. Node data can be accessed via the NIE assigned logical address. When the node table is saved in an ASCII text file the logical node addresses are also saved in ASCII format.

   As nodes are added to the node table the NIE assigns logical addresses in the sequence n1, n2, n3,...,etc. In cases where some nodes have been removed there may be a gap in the logical addresses. In this case when new nodes are added their logical addresses are first assigned from the gap. For example, if the current node table has logical addresses n1, n2, and n5 assigned, new nodes would be assigned logical addresses n3, n4, n6, n7...etc.

4. DECNET address. Phase 4 node addresses have the DECnet address (area.node-number) encoded within the current physical address of the node. The DECnet address is displayed for phase 4 nodes. For non-phase 4 nodes "UNKNOWN" is displayed in place of the DECnet node address.

5. Node device type. (i.e. - DEUNA, DELUA, etc.) In some cases the node device type may not be known.

4.4 BUILD COMMAND

The build command builds the node table by listening to system ID messages sent out by each node in 8 minute intervals.

Command format is:

```
NIE>BU[uild] [/m[in]=xxx]
```

Where the switch /MIN is an optional switch to specify the number of minutes to wait before the build is terminated.

4.5 SAVING THE CURRENT NODE TABLE

The Save command will save the contents of the current node table. The format is:

```
NIE>sa[ve]
```

When a mass storage dev ce is used the node table is saved in [1,2]NIE.TBL file. The file then can be printed or viewed using the editor.

NOTE: With this version of the NIE you cannot specify a file name to be used for the save file. The file name is forced tobe sy:[1,2]NIE.TBL.

When not using a mass storage media, the current node table is copied to a secondary buffer. The node table then can be modified without distroying the original node table.

4.6 UNSAVE

The unsave command will restore the contents of the node table. If a disk is used by the NIE, the node table is restored from [1,2]NIE.TBL file. If no disk is used by the NIE, the secondary buffer is copied over to the primary buffer.

NOTE: With this version of the NIE you cannot specify a file name from which to do the unsave (restore) from.
The command format is:

```
NIE>u[nsave]
```

4.7 SHOW

The Show command will print the physical addresses of nodes selected for testing and the message paramaters to be used (either default or operator input).

The format is:

```
NIE>show <argument>
```

i.e. -

```
NIE>sh[ow] n[odes]          ; Will dispaly the node table

NIE>sh[ow] m[essage]        ; Will display message parameters
```

```
                NIE>sh[ow] c[ounters]        ; Will display counter information

There can be three arguments for Show.  They are:  Nodes, Messages,
Counters.

The Show Nodes command lists all nodes in the Node table.  The list will
include the default physical address, the current physical address and a
logical address assigned to the node by the exerciser.  The node can be
referenced by either of the physical addresses or the logical address.
Logical addresses will be assigned as n1, n2, n3,.....etc.

        NOTE:
          The node table display can be stopped with any character
          from the keyboard.

The Show Messages command will list the message type, message size and the
number of copies to be sent which are currently selected.  Also the
Print/Noprint status will be displayed indicating the error message output
mode.

The Show Counters command will list the counter contents of the host node.


EXAMPLES:


NIE>show message

THE CURRENT MESSAGE PARAMETERS ARE:
TYPE=ASCII , SIZE=292 , COPIES=1, PRINT


NIE>show nodes

   CURRENT ADR        DEFAULT ADR      NAME  DECNET   DEVICE
AA-00-04-00-0A-10  AA-00-03-00-00-01  N0    4.10     DEUNA
AA-00-04-00-1B-10  AA-00-03-00-00-02  N1    4.27     DELUA
AA-00-04-00-0B-10  AA-00-03-00-00-03  N2    4.11     DECNA
AA-00-04-00-9C-10  AA-00-03-00-00-04  N3    4.156    DEQNA


NIE>show counters

ETHERNET COUNTER SUMMARY

SECONDS SINCE LAST ZEROED           : 13257
PACKETS RECEIVED                    : 91847
MULTICAST PACKETS RECEIVED          : 81793
PACKETS RECEIVED IN ERROR (BIT MAP) : 0
DATA BYTES RECEIVED                 : 10491573
MULTICAST BYTES RECEIVED            : 9913460
RCVS LOST - INTERNAL BUF ERROR      : 46
RCVS LOST - LOCAL BUFFER ERROR      : 0
PACKETS TRANSMITTED                 : 12601
```

```
MULTICAST PACKETS TRANSMITTED       : 2675
PKTS XMITTED WITH 3+ COLLISION      : 13
PKTS XMITTED WITH 2+ COLLISION      : 9
PKTS XMITTED BUT DEFERRED           : 169
BYTES TRANSMITTED                   : 1131157
MULTICAST BYTES TRANSMITTED         : 406378
TRANSMIT PACKETS ABORTED (BIT MAP)  : 0
TRANSMIT PACKETS ABORTED COUNTER    : 0
XMIT COLLISION CHECK FAILURE        : 0
```

4.8 RUN

> NOTE
> The RUN command performs the specified test on all entries of the node table. This may cause problems on Extended LANs (multiple LANs connected by Bridges). This means that running a test on all nodes in the node table may not verify that the LAN in question has been properly tested. The node table may not even contain all the nodes on the LAN in question.
>
> Also, on large LANs, running tests against all nodes on the network may be prohibitive.

The run command will cause the execution of the test specified by the argument. The results of the RUN command are used to update the Summary log as well as to output error information. The format is:

```
NIE>r[un] <argument>/pass=nm
```

The argument for Run can be either D[irect], P[attern], L[ooppair] or All.

The qualifier Pass=nn will allow the operator to select the number of passes for the particular test selected. If -1 is specified for the passcount the test will run continuously. If no passcount is specified, the passcount defaults to the passcount set up with the Message command.

The following are standard NIE data patterns:

1. ASCII - The ascii data pattern.
2. ONES - A pattern of all binary 1's.
3. ZEROS - A pattern of all binary 0's.
4. 1ALT - A pattern of alternating binary 1's and 0's starting with 1 (1010...).
5. 0ALT - A pattern of alternating binary 0's and 1's starting with 0 (0101...).

6. CCITT - The ccitt data pattern PDP 11 assembler format.

```
ASCIDATA::                          ;ASCII alphanumer c data pattern

  .ASCII  \ !"#$%&'()*+,-./012345\-
          \6789:;<=>?@ABCDEFGHIJ\-
          \KLMNOPQRSTUVWXYZ[abcd\-
          \efghijklmnopqrstuvwxy\
          \z{|}~ \

CCITTDATA::                           ;CCITT 512 bit test pattern

  .WORD   177603, 157427, 031011, 047321, 163715, 105221
  .WORD   143325, 142304, 040041, 014116, 052606, 172334
  .WORD   105025, 123754, 111337, 111523, 030030, 145064
  .WORD   137642, 143531, 063617, 135075, 066730, 026575
  .WORD   052012, 053627, 070071, 151172, 165044, 031605
  .WORD   166632, 016741
```

EXAMPLES OF RUN COMMAND OUTPUT:

```
NIE>run direct

DIRECT LOOP TESTING STARTED

PASS 1
TESTING NODES:  NIE HOST TO N1    TO NIE HOST -- RESPONSE OK
TESTING NODES:  NIE HOST TO N2    TO NIE HOST -- RESPONSE OK
TESTING NODES:  NIE HOST TO N3    TO NIE HOST -- RESPONSE OK
TESTING NODES:  NIE HOST TO N4    TO NIE HOST -- RESPONSE OK


NIE>run looppair

RUN LOOPPAIR STARTED

PASS 1
TESTING NODES:  NIE HOST TO N1    TO N2    TO N1 TO NIE HOST -  RESPONSE OK
TESTING NODES:  NIE HOST TO N2    TO N3    TO N2 TO NIE HOST -- RESPONSE OK
TESTING NODES:  NIE HOST TO N3    TO N4    TO N3 TO NIE HOST -- RESPONSE OK
TESTING NODES:  NIE HOST TO N4    TO N1    TO N4 TO NIE HOST -- RESPONSE OK


NIE>run pattern
```

```
683
684
685
686
687                         STARTING ASCII PATTERN TEST
688
689                         PASS 1
690                         TESTING NODES:  NIE HOST TO N1    TO NIE HOST -- RESPONSE OK
691                         TESTING NODES:  NIE HOST TO N2    TO NIE HOST -- RESPONSE OK
692                         TESTING NODES:  NIE HOST TO N3    TO NIE HOST -- RESPONSE OK
693                         TESTING NODES:  NIE HOST TO N4    TO NIE HOST -- RESPONSE OK
694
695
696                         STARTING ONES  PATTERN TEST
697
698                         PASS 1
699                         TESTING NODES:  NIE HOST TO N1    TO NIE HOST -- RESPONSE OK
700                         TESTING NODES:  NIE HOST TO N2    TO NIE HOST -- RESPONSE OK
701                         TESTING NODES:  NIE HOST TO N3    TO NIE HOST -- RESPONSE OK
702                         TESTING NODES:  NIE HOST TO N4    TO NIE HOST -- RESPONSE OK
703                                 .
704                                 .
705                               etc
706
707
708
709                         NIE>run all
710
711
712                         PASS 1
713                         TESTING NODES:  NIE HOST TO N1    TO NIE HOST  - RESPONSE OK
714                         TESTING NODES:  NIE HOST TO N2    TO NIE HOST -- RESPONSE OK
715                         TESTING NODES:  NIE HOST TO N3    TO NIE HOST -- RESPONSE OK
716                         TESTING NODES:  NIE HOST TO N4    TO NIE HOST -- RESPONSE OK
717
718
719                         RUN ALL STARTED
720
721                         PASS 1
722                         TESTING NODE N1    WITH REMAINING TABLE ENTRIES
723                         TESTING NODES:  NIE HOST TO N1    TO N2    TO NIE HOST -- RESPONSE OK
724                         TESTING NODES:  NIE HOST TO N2    TO N1    TO NIE HOST -- RESPONSE OK
725                         TESTING NODES:  NIE HOST TO N1    TO N3    TO NIE HOST -- RESPONSE OK
726                         TESTING NODES:  NIE HOST TO N3    TO N1    TO NIE HOST -- RESPONSE OK
727                         TESTING NODES:  NIE HOST TO N1    TO N4    TO NIE HOST -- RESPONSE OK
728                         TESTING NODES:  NIE HOST TO N4    TO N1    TO NIE HOST -- RESPONSE OK
729
730                         PASS 1
731                         TESTING NODE N2    WITH REMAINING TABLE ENTRIES
732                         TESTING NODES:  NIE HOST TO N2    TO N3    TO NIE HOST -- RESPONSE OK
733                         TESTING NODES:  NIE HOST TO N3    TO N2    TO NIE HOST -- RESPONSE OK
734                         TESTING NODES:  NIE HOST TO N2    TO N4    TO NIE HOST    RESPONSE OK
735                         TESTING NODES:  NIE HOST TO N4    TO N2    TO NIE HOST -- RESPONSE OK
736
737                         PASS 1
738                         TESTING NODE N3    WITH REMAINING TABLE ENTRIES
739                         TESTING NODES:  NIE HOST TO N3    TO N4    TO NIE HOST -- RESPONSE OK
```

TESTING NODES: NIE HOST TO N4 TO N3 TO NIE HOST - RESPONSE OK

4.9 BOUNCE

The Bounce command allows the operator to select the path for sending a packet from the NIE host, through the NI then back to the host. The nodes identified in the command will be sequentially placed into the data field of the packet with the forward command. The nodes may be specified by physical or logical addresses. The NIE host node should not be included in the node list for the Bounce command. The results of the BOUNCE command have no effect on the Summary log. The format is:

NIE>bo[unce] addr1,addr2...,addrn

The limit on the number of nodes to which the packet can be forwarded is related to the remaining size of the data field. This command is useful for testing across repeaters or quickly testing endnodes.

4.10 MESSAGE

The Message command allows the operator to change the defa paramaters of message type, message size, and message number. The Format is:

NIE>m[essage] /ty[pe]=ascii/s[ize]=n/c[opies]=n

The message size will be variable, between 46 and 1500 bytes. Message size is defined as the size of the packet data field and excludes the source, destination, packet type and CRC fields.

The message copies is the number of times the message is to be transmitted and is a posative integer. A -1 indicates that packets are to be looped continuously. Default is 1.

4.11 NODES

The Nodes command is used to allow the operator to enter nodes for testing into the current node table. The format is:

NIE>nodes addr1,addr2,...,addrn

The addr argument is the physical address of the node on the NI. The NIE will assign a logical address for each entry in the current node table. Duplicate nodes may be added in this manner although the NIE assigned

logical addresses will always be unique.

4.12 PRINT/NOPRINT

The PRINT command causes all errors to generate error messages which are output to the operator. If the NOPRINT command is issued error reporting stops after the first five errors. In both cases the Summary log will continue to be updated. The PRINT and NOPRINT commands have no arguments and the default is PRINT. The command stays in effect untill changed by the operator. The print status is displayed via the SHOW MESSAGE command.

4.13 NOHOE/HOE

i.e. -

```
NIE>noh[oe]                    ; Do not halt on error

  or

NIE>hoe                        ; Halt on error (DEFAULT)
```

Halt on Error (HOE) causes the current RUN to halt when an error is encountered inspite the pass count. The NO Halt on Error will cause the run to run to completion inspite of errors.

4.14 SUMMARY

The Summary command prints the summary message of conditions and errors as a result of testing. Summary information is obtained by typing Summary when the NIE is running. There are no switches or qualifiers for the summary command.

```
NIE>su[mmary]
```

4.15 CLEAR

The Clear command will have three arguments, node, message and summary.

The format is:

NIE>c[lear] <argument>

The Clear Node/Addr will remove a node from the node table. The node may be specified by either its physical or logical address. OPTIONALLY some implementations may allow a list of physical and logical addresses to be specified.

The "C[lear] N[ode]/A[ll]" will clear the entire node table.

The "C[lear] M[essage]" will reset the message paramaters to the default state.

The "C[lear] Su[mmary]" command will clear the node summary table.

4.16 IDENTIFY

The Identify command will perform a request ID to the physical or logical address included in the command line. The resulting data will then be displayed.

The format is:

NIE>i[dentify] <address>

EXAMPLE:

```
NIE>identify 00-04-00-00-00-bc

NODE CURRENT ADDRESS: 00-04-00-00-00-BC
NODE DEFAULT ADDRESS: 00-00-AB-00-00-0C
RECEIPT NUMBER: A045
MAINTENANCE VERSION: 03
ECO: 00
USER ECO: 00
FUNCTION VALUE 1: 05
FUNCTION VALUE 2: 00
DEVICE: 01
```

OR, using a logical node name -

```
NIE>identify N5

NODE CURRENT ADDRESS: 00-04-00-00-00-BC
NODE DEFAULT ADDRESS: 00-00-AB-00-00-0C
RECEIPT NUMBER: A045
MAINTENANCE VERSION: 03
ECO: 00
USER ECO: 00
FUNCTION VALUE 1: 05
```

FUNCTION VALUE 2: 00
DEVICE: 01

4.17 SUMMARY

A log of events is maintained during RUN command processing. These statistics caN be displayed with the SUMMARY command.

```
NIE>su[mmary]
```

NODE	RCV NOT COMPLETED	RCV COMPLETED	LENGTH ERRORS	COMPARE ERRORS	BYTES COMPARED	NUMBER BYTES TRANSFERED
00-04-00-00-00-10	0	10	1	1	2000	2000
00-04-00-00-00-11	0	10	1	1	2000	2000
00-04-00-00-00-12	0	10	1	1	2000	2000
00-04-00-00-00-13	0	10	1	1	2000	2000

5.0 ERROR MESSAGES

COMMAND OR SWITCH NOT IMPLEMENTED
Command or switch is an optional NIE function. However, this implementation does not support the command.

COMPARE ERROR - BYTE EXPECTED = XXX, BYTE RECEIVED = YYY
This error is displayed when the data sent in a loop message does not match the data received in the response message.

INVALID COMMAND
Invalid or syntactically incorrect command given.

INVALID COUNTER DATA
SHOW COUNTERS error message when running under DECnet. The format of the counter data received from DECnet is incorrect. Internal program error.

INVALID DISK NAME
The has tried to accessed the specified disk. The disk does not exist on the current system. Enter correct device name.

INVALID ETHERNET DEVICE NAME
NX: and XE: are the only Ethernet devices possible with this version of the NIE. NX: requires DECnet present and running. XE: device uses the stand alone RSX QIO DEUNA/DELUA Driver. You must use one of these Ethernet devices.

INVALID HEX CHARACTER
Character specified is not a Hex digit.

INVALID MESSAGE SIZE

Maximum message size on a non DECnet system is 1498(10). Maximum message size on a DECXnet system is the same as that used by DECnet.

INVALID MESSAGE TYPE
Message type specified is not valid.

INVALID NODE ADDRESS
Node address as specified is invalid. Node address specified must be a hex string or a logical node number.
i.e. - AA-00-00-00-00-02 or Nxxx (where, Nxxx is logical node number assigned by NIE)

INVALID OPTION
Option specified is invalid.

INVALID PASS NUMBER
The pass switch specified with one of the RUN command is invalid. Max pass is 100.

INVALID PROGRAM PARAMETER - NOT ENOUGH BUFFERS
The number of receive buffers is a hard wired program parameter. This message is given if this parameter is incorrect.

INVALID PROGRAM PARAMETER - INVALID BUFFER SIZE
If no mass storage media is being used, the minimum buffer size is 514. bytes (disk buffer size plus link word). Physical buffer size is hard wired into the program. This should not normally be seen.

NIE EXITING
Just what it says.

???? NODE ENTRIES ADDED TO NODE TABLE
UNSAVE indicates the number of node entries added to the node table.

NODE NOT FOUND
Node number specified (logical or hex) was not found in the node table. The command requires a node address which is in the node table.

NODE TABLE EMPTY
A command was given that requires that the Node Table be not empty. No further action is taken.

NODE TABLE FULL
Indicates the NIE was unable to add an entry to the node table. i.e. - BUILD, UNSAVE, or adding new entries to the node table (NODE...).

NO NODES SPECIFIED
Command requires a node address to be specified. (i.e. CLEAR NODE/"node")

NUMBER OF COPIES INVALID
Number of copies specified is not valid. Currently Max of copies is set at 10,000.

```
1030
1031
1032
1033                    NOT ENOUGH NODES IN NODE TABLE FOR "LOOPPAIR" OR "RUN ALL" COMMANDS
1034                      There must be at least 2 node entries in the nodetable to  RUN  LOOPPAIR
1035                      or RUN ALL commands.
1036
1037                    NOT ENOUGH RECEIVE BUFFERS AVAILABLE
1038                      Internal program error.  The NIE has lost track of it's recieve buffers.
1039
1040                    OPEN ERROR ON NIE SAVE FILE
1041                      The user attempted to do A SAVE command but NIE was unable to  create  a
1042                      save file.
1043
1044                    RESPONSE OK
1045                      This message indicates  that  the  response  expected  by  the  NIE  was
1046                      recieved  by  the NIE and that the test data returned was verified to be
1047                      correct.
1048
1049                    PROGRAM ERROR - NO BUFFERS AVAILABLE
1050                      This message should not be seen during normal operation.  The  NIE  does
1051                      some  internal  buffer  management  for  send and receive messages.  The
1052                      buffer manager is out of buffers and the NIE is unable to  continue  the
1053                      requested operation.
1054
1055                    READ COUNTERS REQUEST TO DECNET FAILED
1056                      Read counters request to DECnet failed.  This is an internal  error  for
1057                      SHOW COUNTERS request.
1058
1059                    RECEIVE DECNET COUNTERS FAILED
1060                      Receiving DECnet counters failed.  This is an internal  error  for  SHOW
1061                      COUNTERS request.
1062
1063                    SET CHARACTERISTICS FAILED
1064                      This message  indicates  that  the  set  characteristics  QIO  has  been
1065                      rejected.   This would occure if two NIEs were to run on the same system
1066                      and each would try to select the same protocol/address pairs.
1067
1068                      If only one NIE is being run, then this error  is  an  internal  program
1069                      error.
1070
1071                    TEXT NOT DEFINED - COMMAND IGNORED
1072                      MESSAGE /TYPE=TEXT was given before defining the  text.  Do  a  MESSAGE
1073                      /TEXT=xyz  (xyz  is  a  string  of  ascii characters) to define the text
1074                      string before issuing MESSAGE /TYPE=TEXT command.
1075
1076                    TIME OUT - REMOTE NODE NOT RESPONDING
1077                      This message is displayed when the NIE wait time expires  while  waiting
1078                      for  a  response  from a remote node.  All the conectivity commands will
1079                      result  in  this  message  if  the  remote  node  being  tested  is  not
1080                      responding.
1081
1082                      During a RUN command execution, this message indicates that the  node(s)
1083                      being tested are displayed in the message on the previous line.
1084
1085                    TRANSMIT ERROR - COMMAND TERMINATED
1086                      This error message only indicates that there has been an error when  the
```

NIE has attempted to transmit a message. It does not indicate what the error was. Transmission errors are caused by some hardware malfunctioning. When running under DECnet an error log message on the console may indicate what the error was.

UNABLE TO OBTAIN LOCAL ADDRESS FROM ETHERNET DEVICE DRIVER
This message may be received in a non-DECnet evironment only. At initialize time NIE tries to read the current physical address of the Ethernet device. This is needed for source address determination. The source address is used in MOP messages to specify return address of responses. This is a driver/system error. Should not normally occure!

UNABLE TO OPEN ETHERNET PORT
This message indicates that the RSX QIO Driver is not present or that (if running with DECnet) DECnet is not up and running. The error may also indicate that the Ethernet device cannot be initialized.

UNABLE TO OPEN SAVED NODE TABLE FILE
UNSAVE command is unable to open saved node table file.

UNABLE TO READ COUNTERS FROM DEVICE
NIE was unable to read counters from non-DECnet driver. This is an internal program error.

UNABLE TO READ SAVED NODE TABLE FILE
UNSAVE command is unable to read saved node table file.

UNABLE TO READ WORK FILE
If the user specified a disk to be used at initialize time, NIE will create a temporary (work file) file on that disk for it's node table. An error while trying to read this disk will produce the above message.

UNABLE TO TALK TO DECNET
This message can occure while doing a SHOW COUNTERS command when running under DECnet. The NIE is trying to create a Network Data Queue and failes. This may be caused by system resources not available or DECnet is going down.

6.0 COUNTERS INTERPRETATION

DELUA/DEUNA Counter Specification:

The counter values are unsigned integers. Counters latch at their maximum values to indicate overflow.

Seconds Since Last Zeroed -
16 bits for the number of seconds since the counters were last zeroed.

Packets Received -
32 bits for the total number of datagrams received error free.

Multicast Packets Received -
32 bits for the total number of cmulticast datagrams received error free.

Packets Received with Error - Bitmap -

1. Bit <00> - CRC. Block Check Error - A datagram failed the CRC check.

2. Bit <01> - FRAM. Framing Error - A datagram failed the CRC check and did not contain an integral multiple of 8 bits.

3. Bit <02> - MLEN. Message Length Error - A datagram was larger than 1518 bytes.

4. Bits <15:03> = 0.

Packets Received with Error -
16 bits for the total number of datagrams received with one or more errors logged in the bitmap. Includes only datagrams that passed destination address comparison.

Data Bytes Received -
32 bits for the total number of data bytes received error free, exclusive of data link protocol overhead.

Multicast Bytes Received -
32 bits for the total number of multicast data bytes received error free, exclusive of data link protocol overhead.

Receive Packet Lost -
Internal Buffer Error - 16 bits for the total number of times there was a discard of an incoming packet due to lack of internal buffer space. Incoming packets must be error free to be counted.

Receive Packet Lost -
Local Buffer Error - 16 bits for the total number of times there was a problem with a receive ring data buffer. This counter is incremented on one of more of the following occurences.

1. Buffer Unavailable - A datagram was lost because there was no available buffer on the receive ring.

2. Buffer too Small - A datagram was truncated because it was larger than the available buffer space on the receive ring.

Packets Transmitted - 32 bits for the total number of datagrams successfully transmitted, including transmissions in which the collision test signal failed to assert.

Multicast Packets Transmitted -
32 bits for the total number of multicast datagrams successfully transmitted, including transmissions in which the collision test signal failed to assert.

Packets Transmitted -
3+ Attempts - 32 bits for the total number of datagrams successfully transmitted on three or more attempts, including transmissions in which the collision test signal failed to assert.

Packets Transmitted -
2 Attempts - 32 bits for the total number of datagrams successfully transmitted on two attempts, including transmissions in which the collision test signal failed to assert.

Packets Transmitted -
Deferred - 32 bits for the total number of datagrams successfully transmitted on the first attempt after deferring, including transmissions in which the collision test signal failed to assert

Data Bytes Transmitted -
32 bits for the total number of data bytes successfully transmitted, exclusive of data link protocol overhead and not counting data link generated retransmissions, but including transmissions in which the collision test signal failed to assert.

Multicast Bytes Transmitted -
32 bits for the total number of multicast data bytes successfully transmitted, exclusive of data link protocol overhead and not counting DELUA generated retransmissions, but including transmissions in which the collision test signal failed to assert.

Transmit Packets Aborted - Bitmap -

1. Bit <00> - RTRY. Excessive Collisions - Retry error, 16 unsuccessful transmission attempts.

2. Bit <01> - LCAR. Loss of Carrier - Retry error (16 unsuccessful transmission attempts), loss of carrier flag, and non-zero TDR value on last attempt.

3. Bit <02> - 0.

4. Bit <03> - 0.

5. Bit <04> - MLEN. Data Block too Long - The DELUA aborted the transmission process because the datagram exceeded the maximum packet size.

6. Bit <05> - LCOL. Remote Failure to defer - Late collision on the last transmission attempt.

7. Bits <15:06> = 0.

Transmit Packets Aborted -
16 bits for the total number of datagrams that were aborted during transmission for one or more of the bitmapped errors.

Transmit Collision Detect Failure -

16 bits for the total number of times the collision test signal failed to assert following an apparently suscessful transmission.

DELUA ONLY COUNTERS

PORT DRIVER ERROR -
16 bits for the total number of times the Port Driver attempts to issue another Port/Ancilliary command while one is still being processed.

BABBLE COUNTER -
16 bits counter for the total number of times the LANCE reported the babble condition on the channel.

APPENDIX A

SAMPLE NIE BUILD OUTPUT

```
>MOU MT0:NIE/DENS=1600          ; Mount mag tape kit
>PIP /NV=MT0:CZNIA.SEQ          ; Copy documentation file to you area
>PRINT CZNIA.SEQ                ; Print doc file for your information
>PIP /NV=MT0:NIEBLD.CMD         ; Copy NIE build file to your area
>@NIEBLD
>;
>; !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
>; ! NOTE:                                                                !
>; ! This command file must be run from a privileged account as it needs  !
>; ! to copy some files to the SY:[1,1] area.                             !
>; !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
>;
>;
>; The format for the input device and UIC is DEV:[XXX,YYY].  The command
>; file does not do extensive syntex checking.  Therefore, if specified, the
>; format of the device-UIC string must be correct.
>;
>* Input device and UIC of source files [D=DR0:[6,6]] [S]:  mt:
>;
>; The format for the destination device and UIC is DEV:[XXX,YYY].  The command
>; file does not do extensive syntex checking.  Therefore, if specified, the
>; format of the device-UIC string must be correct.
>;
>* Output device and destination UIC [D=DR0:[6,6]]         [S]: <cr>
>PIP DR0:[6,6]/NV=MT:NIESUP.MAC
>PIP DR0:[6,6]/NV=MT:NIESUB.MAC
>PIP DR0:[6,6]/NV=MT:NIECMD.MAC
>PIP DR0:[6,6]/NV=MT:NIEPRS.MAC
>PIP SY:[1,1]/NV=MT:DEUNA.MLB
>PIP /NV=MT:DEUNA.MLB
>INS $MAC
>INS $TKB
>INS $CRF
>MAC @DR0:[6,6]NIEASM
>TKB @DR0:[6,6]NIETKB.CMD
>REM NIE
>REM NI1
>INS DR0:[6,6]NIE
>INS DR0:[6,6]NI1
```

```
1336
1337
1338
1339                            >* Do you want to build the DEUNA/DELUA driver?
1340                            >PIP SY:[1,1]/NV=MTO:UNAMC.TSK
1341                            >PIP DR0:[6,6]/NV=MTO:UNAMC.TSK
1342                            >PIP SY:[1,1]/NV=MTO:LUAMC.TSK
1343                            >PIP DR0:[6,6]/NV=MTO:LUAMC.TSK
1344                            >PIP DR0:[6,6]/NV=MTO:XEDRV.MAC
1345                            >PIP DR0:[6,6]/NV=MTO:XETAB.MAC
1346                            >PIP DR0:[6,6]/NV=MTO:UML.MAC
1347                            >MAC @DR0:[6,6]XEDRVASM.CMD
1348                            >TKB @DR0:[6,6]XEDRVBLD.CMD
1349                            >;
1350                            >; We have completed building the driver and the ECO microcode loader
1351                            >; support task.
1352                            >;
1353                            >; *** NOTE ***
1354                            >; If DECnet is running and is using your DEUNA or DELUA device then you
1355                            >; must not try to LOAD the XE: (DEUNA/DELUA) driver as the device Vector
1356                            >; and RSX Device Control Block are already taken and this will cause an
1357                            >; error.
1358                            >;
1359                            >; However, if DECnet is not running and you wish to use the the RSX QIO
1360                            >; DEUNA/DELUA driver (XE: Driver) then you can have this command file
1361                            >; load it for you. at this time.
1362                            >;
1363                            >* Do you want to LOAD the XE: (DEUNA/DELUA) driver? Y
1364                            >REM UML...
1365
1366                                        ; RSX-11M
1367
1368                            >LOAD XE:/PAR=GEN/HIGH
1369                            >INS DR0:[6,6]UML
1370
1371
1372                                        ; RSX-11M-PLUS
1373
1374                            >INS $CON
1375                            >LOAD XE:/PAR=GEN/HIGH
1376                            >CON SET XEA VEC=120 CSR=174510
1377                            >CON ONLINE XEA,XE0:
1378                            >INS DR0:[6,6]UML
1379
1380                            >@ <EOF>
1381                            ↑
1382        000001                      .END
```

C3

CZNIA RSX ONLINE NIE MACRO M1200 10-JUL-85 14:24 PAGE 26-1
SYMBOL TABLE


```
. ABS.  000000     000
        000000     001
ERRORS DETECTED:  0

VIRTUAL MEMORY USED:  29 WORDS  ( 1 PAGES)
DYNAMIC MEMORY:  20324 WORDS  ( 78 PAGES)
ELAPSED TIME:  00:00:17
.CZNIA.SEQ/-SP=CZNIA
```