

1	2	3
4	5	6
7	8	9
10	11	12
13	14	15
16	17	18
19	20	21
22	23	24
25	26	27
28	29	30
31	32	33
34	35	36
37	38	39
40	41	42
43	44	45
46	47	48
49	50	51
52	53	54
55	56	57
58	59	60
61	62	63
64	65	66
67	68	69
70	71	72
73	74	75
76	77	78
79	80	81
82	83	84
85	86	87
88	89	90
91	92	93
94	95	96
97	98	99
100	101	102

B1

† iB W
A S:

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54

.TITLE CZLIA - RSX ONLINE NIE
.IDENT /01.01/
.ENABL LC
.REM †

IDENTIFICATION

PRODUCT CODE: AC-FF94A-MC
PRODUCT NAME: CZLIAAO RSX ONLINE NETWORK INTERCONNECT EXERCISOR
PRODUCT DATE: JUNE 1985
MAINTAINER: MERRIMACK DIAGNOSTIC ENGINEERING
AUTHOR: ADAM KOJNOK

THE INFORMATION IN THIS DOCUMENT IS SUBJECT TO CHANGE WITHOUT NOTICE AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT CORPORATION. DIGITAL EQUIPMENT CORPORATION ASSUMES NO RESPONSIBILITY FOR ANY ERRORS THAT MAY APPEAR IN THIS DOCUMENT.

NO RESPONSIBILITY IS ASSUMED FOR THE USE OR RELIABILITY OF SOFTWARE ON EQUIPMENT THAT IS NOT SUPPLIED BY DIGITAL OR ITS AFFILIATED COMPANIES.

COPYRIGHT (C) 1985 BY DIGITAL EQUIPMENT CORPORATION

THE FOLLOWING ARE TRADEMARKS OF DIGITAL EQUIPMENT CORPORATION:

DIGITAL PDP UNIBUS MASSBUS
DEC DECUS DECTAPE

REVISION HISTORY:

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

D1

56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71

RSX ONLINE NETWORK INTERCONNECT EXERCISER

USER'S GUIDE

ISSUED BY: Adam J. Kojnok

AC-FF94A-MC

REV A

73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109

TABLE OF CONTENTS		
1.0	INTRODUCTION	2
2.0	BUILDING THE NIE	2
2.1	NIE BUILD PARAMETERS	3
2.2	PROCEDURE TO BUILD THE NIE	3
2.3	INSTALLING NIE	4
3.0	RSX QIO UNA DRIVER BUILD	4
3.1	RSX QIO UNA DRIVER PARAMETERS	5
3.2	BUILDING XE: THE DRIVER	5
3.3	INSTALLING (LOADING) THE DRIVER	5
4.0	NIE COMMANDS	6
4.1	HELP	6
4.2	EXITING THE NIE	6
4.3	BUILDING A NODE TABLE	6
4.4	BUILD COMMAND	7
4.5	SAVING THE CURRENT NODE TABLE	8
4.6	UNSAVE	8
4.7	SHOW	8
4.8	RUN	10
4.9	BOUNCE	13
4.10	MESSAGE	13
4.11	NODES	13
4.12	PRINT/NOPRINT	14
4.13	NOHOE/HOE	14
4.14	SUMMARY	14
4.15	CLEAR	14
4.16	IDENTIFY	15
4.17	SUMMARY	16
5.0	ERROR MESSAGES	16
6.0	COUNTERS INTERPRETATION	19

APPENDIX A SAMPLE NIE BUILD OUTPUT

111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165

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 without 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.

167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223

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
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281

```
XEDRV.MAC      ; XE: Driver code
XETAB.MAC      ; XE: Driver RSX Database
UNAMC.TSK      ; XE: Driver DEUNA ECO microcode file
LUAMC.TSK      ; XE: Driver DELUA ECO microcode file
```

Building command sequence:

NOTE

```
DEV: for Magtape = MT:
DEV: for DL       = DLx:[3,3]
```

```
>MOU DEV:NIE      ; Mount distributions kit
>PIP /NV=DEV:CZNIA.SEQ      ; Copy documentation file
>PIP /NV=DEV:NIEBLD.CMD    ; Copy NIE build command file from distro kit
>@NIEBLD          ; Do the build
```

Answer questions asked by build command file.

*** NOTE ***

See APPENDIX A for sample build print out!

2.3 INSTALLING NIE

NIEBLD.CMD will do the installation at the end of the build. However, if the system is re-started, the following must be installed:

```
MCR>INS [USER-UIC]NIE
MCR>INS [USER-UIC]NI1
```

NOTE

The XE: driver must be installed (LOADED) if a non-DECnet system is being used.

3.0 RSX QIO UNA DRIVER BUILD

The following command lines will assemble the driver code and the driver RSX database:

Assemble driver code -

```
[11,24]xedrv,[11,34]xedrv/cr/-sp=[1,1]exemc/ml,dev:[user-uic]d,-
deuna/ml,[11,10]rsxmc/pa:1,dev:[user-uic]xedrv
```

Assemble driver RSX data base -

```
[11,24]xetab,[11,34]xetab/cr/-sp=[1,1]exemc/ml,-
```


283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338

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

3.1 RSX QIO UNA DRIVER PARAMETERS

The following parameters 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.COM command file will generate DEV:[USER-UIC]XEDRVBLD.COM command file. This file contains the TRB 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,XEO: ; 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

340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396

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.

398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449

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.

451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507

4.5 SAVING THE CURRENT NODE TABLE

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

NIE>ss[ve]

When a mass storage device 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 to be 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 destroying 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 parameters to be used (either default or operator input).

The format is:

NIE>show <argument>

i.e. -

NIE>sh[ow] n[odes] ; Will display the node table
NIE>sh[ow] m[essage] ; Will display message parameters

509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565

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

```

567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623

```

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...).

625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681

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

ASCIDATA:: ;ASCII alphanumeric data pattern

```
.ASCII \ !"#%&'()*+,-./012345\ -
        \6789:;<=>?@ABCDEFGHIJ\ -
        \KLMNOPQRSTUVWXYZ[abcd\ -
        \efghi jklmnopqrstuvwxyz\ -
        \zi }0 \
```

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
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739

STARTING ASCII PATTERN TEST

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

STARTING ONES PATTERN TEST

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

etc

NIE>run all

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

RUN ALL STARTED

PASS 1
TESTING NODE N1 WITH REMAINING TABLE ENTRIES
TESTING NODES: NIE HOST TO N1 TO N2 TO NIE HOST -- RESPONSE OK
TESTING NODES: NIE HOST TO N2 TO N1 TO NIE HOST -- RESPONSE OK
TESTING NODES: NIE HOST TO N1 TO N3 TO NIE HOST -- RESPONSE OK
TESTING NODES: NIE HOST TO N3 TO N1 TO NIE HOST -- RESPONSE OK
TESTING NODES: NIE HOST TO N1 TO N4 TO NIE HOST -- RESPONSE OK
TESTING NODES: NIE HOST TO N4 TO N1 TO NIE HOST -- RESPONSE OK

PASS 1
TESTING NODE N2 WITH REMAINING TABLE ENTRIES
TESTING NODES: NIE HOST TO N2 TO N3 TO NIE HOST -- RESPONSE OK
TESTING NODES: NIE HOST TO N3 TO N2 TO NIE HOST -- RESPONSE OK
TESTING NODES: NIE HOST TO N2 TO N4 TO NIE HOST -- RESPONSE OK
TESTING NODES: NIE HOST TO N4 TO N2 TO NIE HOST -- RESPONSE OK

PASS 1
TESTING NODE N3 WITH REMAINING TABLE ENTRIES
TESTING NODES: NIE HOST TO N3 TO N4 TO NIE HOST -- RESPONSE OK

741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797

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 default parameters 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 positive 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

799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855

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 until 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:

857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913

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 parameters 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

915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971

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

973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028

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
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086

NOT ENOUGH NODES IN NODE TABLE FOR "LOOPPAIR" OR "RUN ALL" COMMANDS
There must be at least 2 node entries in the nodetable to RUN LOOPPAIR or RUN ALL commands.

NOT ENOUGH RECEIVE BUFFERS AVAILABLE
Internal program error. The NIE has lost track of it's receive buffers.

OPEN ERROR ON NIE SAVE FILE
The user attempted to do A SAVE command but NIE was unable to create a save file.

RESPONSE OK
This message indicates that the response expected by the NIE was received by the NIE and that the test data returned was verified to be correct.

PROGRAM ERROR - NO BUFFERS AVAILABLE
This message should not be seen during normal operation. The NIE does some internal buffer management for send and receive messages. The buffer manager is out of buffers and the NIE is unable to continue the requested operation.

READ COUNTERS REQUEST TO DECNET FAILED
Read counters request to DECnet failed. This is an internal error for SHOW COUNTERS request.

RECEIVE DECNET COUNTERS FAILED
Receiving DECnet counters failed. This is an internal error for SHOW COUNTERS request.

SET CHARACTERISTICS FAILED
This message indicates that the set characteristics QIO has been rejected. This would occur if two NIEs were to run on the same system and each would try to select the same protocol/address pairs.

If only one NIE is being run, then this error is an internal program error.

TEXT NOT DEFINED - COMMAND IGNORED
MESSAGE /TYPE=TEXT was given before defining the text. Do a MESSAGE /TEXT=xyz (xyz is a string of ascii characters) to define the text string before issuing MESSAGE /TYPE=TEXT command.

TIME OUT - REMOTE NODE NOT RESPONDING
This message is displayed when the NIE wait time expires while waiting for a response from a remote node. All the connectivity commands will result in this message if the remote node being tested is not responding.

During a RUN command execution, this message indicates that the node(s) being tested are displayed in the message on the previous line.

TRANSMIT ERROR - COMMAND TERMINATED
This error message only indicates that there has been an error when the

1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143

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 environment 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 occur!

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 occur while doing a SHOW COUNTERS command when running under DECnet. The NIE is trying to create a Network Data Queue and fails. 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.

1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201

Multicast Packets Received -
32 bits for the total number of multicast 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 occurrences.

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.

1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259

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 -

1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277

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

DELUA ONLY COUNTERS

PORT DRIVER ERROR -

16 bits for the total number of times the Port Driver attempts to issue another Port/Ancillary 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.

1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334

APPENDIX A
SAMPLE NIE BUILD OUTPUT

```

>MOU MTO:NIE/DENS=1600 ; Mount mag tape kit
>PIP /NV=MTO:CZNIA.SEQ ; Copy documentation file to you area
>PRINT CZNIA.SEQ ; Print doc file for your information
>PIP /NV=MTO: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 syntax 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 syntax 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]NIEASH
>TKB @DR0:[6,6]NIETKB.CMD
>REM NIE
>REM NI1
>INS DR0:[6,6]NIE
>INS DR0:[6,6]NI1

```

SAMPLE NIE BUILD OUTPUT

```

1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382

```

```

>* Do you want to build the DEUNA/DELUA driver?
>PIP SY:[1,1]/NV=MT0:UNAMC.TSK
>PIP DRO:[6,6]/NV=MT0:UNAMC.TSK
>PIP SY:[1,1]/NV=MT0:LUAMC.TSK
>PIP DRO:[6,6]/NV=MT0:LUAMC.TSK
>PIP DRO:[6,6]/NV=MT0:XEDRV.MAC
>PIP DRO:[6,6]/NV=MT0:XETAB.MAC
>PIP DRO:[6,6]/NV=MT0:UML.MAC
>MAC @DRO:[6,6]XEDRVASM.CMD
>TKB @DRO:[6,6]XEDRVBLD.CMD
>
>; We have completed building the driver and the ECO microcode loader
>; support task.
>;
>; *** NOTE ***
>; If DECnet is running and is using your DEUNA or DELUA device then you
>; must not try to LOAD the XE: (DEUNA/DELUA) driver as the device Vector
>; and RSX Device Control Block are already taken and this will cause an
>; error.
>;
>; However, if DECnet is not running and you wish to use the the RSX QIO
>; DEUNA/DELUA driver (XE: Driver) then you can have this command file
>; load it for you. at this time.
>;
>* Do you want to LOAD the XE: (DEUNA/DELUA) driver? Y
>REM UML...

; RSX-11M

>LOAD XE:/PAR=GEN/HIGH
>INS DRO:[6,6]UML

; RSX-11M-PLUS

>INS $CON
>LOAD XE:/PAR=GEN/HIGH
>CON SET XEA VEC=120 CSR=174510
>CON ONLINE XEA,XEO:
>INS DRO:[6,6]UML

>@ <EOF>
†
.END

```

000001

C3

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

Page 27

. 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