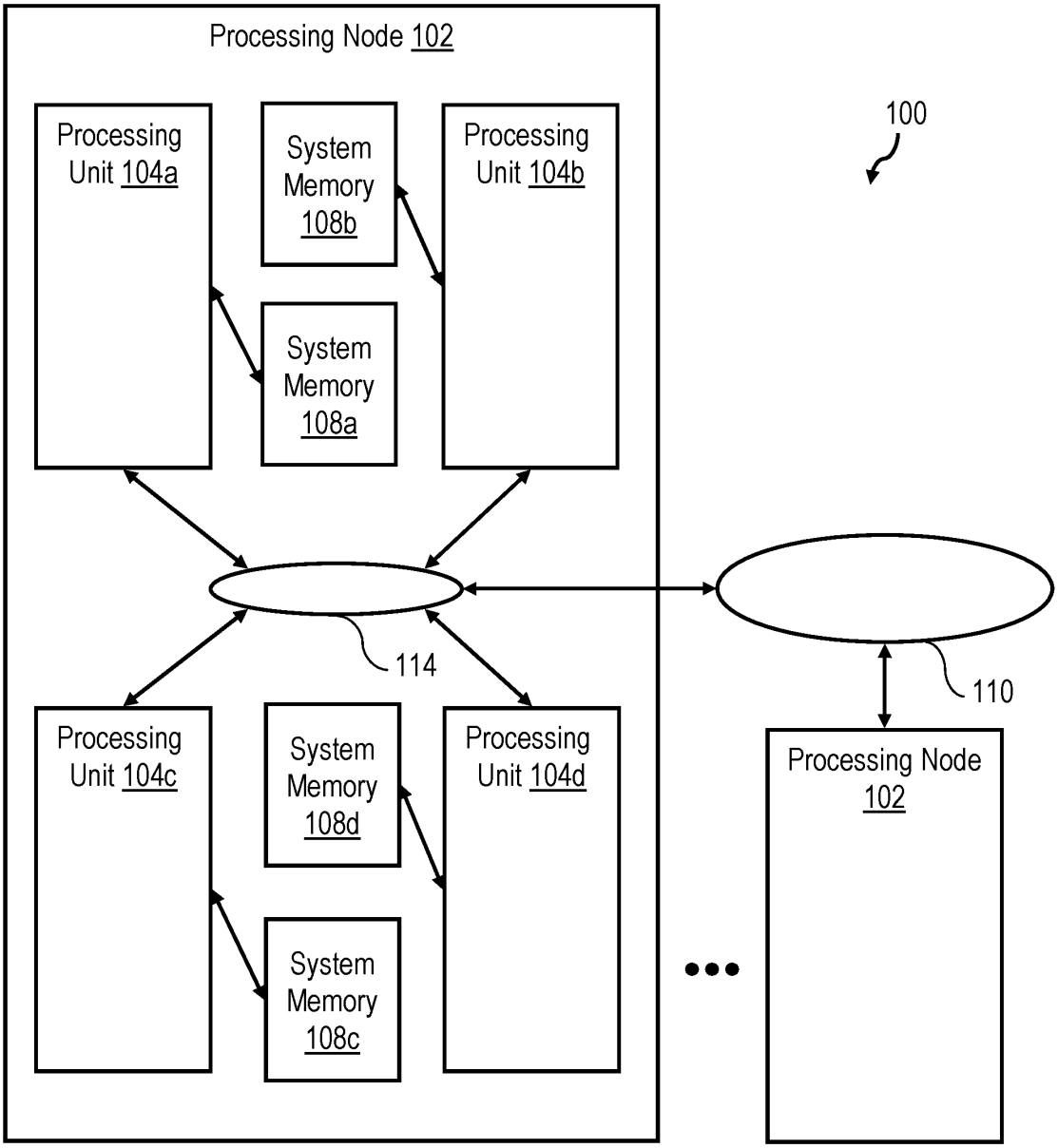
Processing Node 102
Processing Unit 104a
System Memory 108b
Processing Unit 104b
System Memory 108a
100
114
110
Processing Unit 104c
System Memory 108d
Processing Unit 104d
System Memory 108c
Processing Node 102

Figure 1

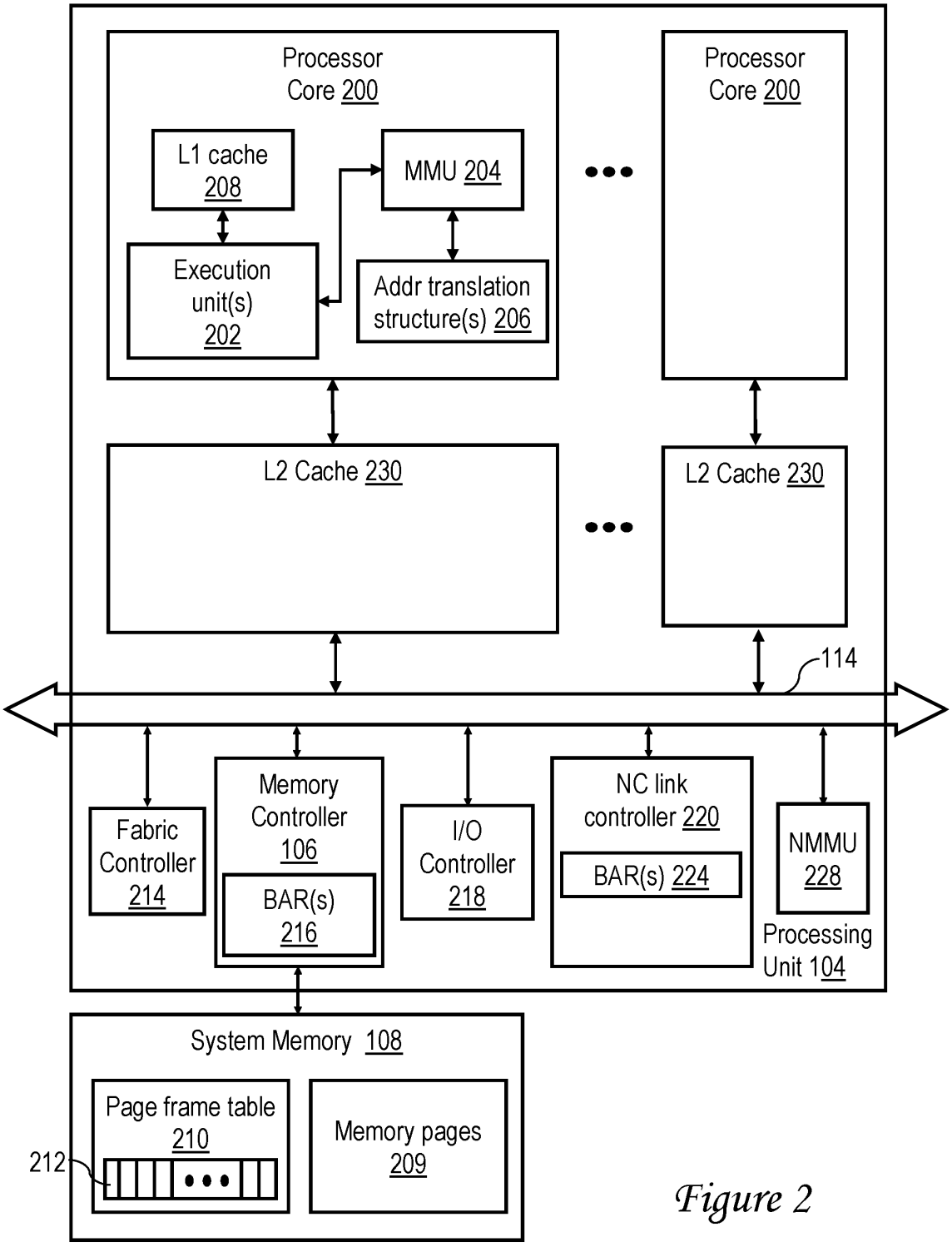
Processor Core 200
L1 cache 208
MMU 204
Execution unit(s) 202
Addr translation structure(s) 206
Processor Core 200
L2 Cache 230
L2 Cache 230
114
Fabric Controller 214
Memory Controller 106
BAR(s) 216
I/O Controller 218
NC link controller 220
BAR(s) 224
NMMU 228
Processing Unit 104
System Memory 108
Page frame table 210
212
Memory pages 209

Figure 2

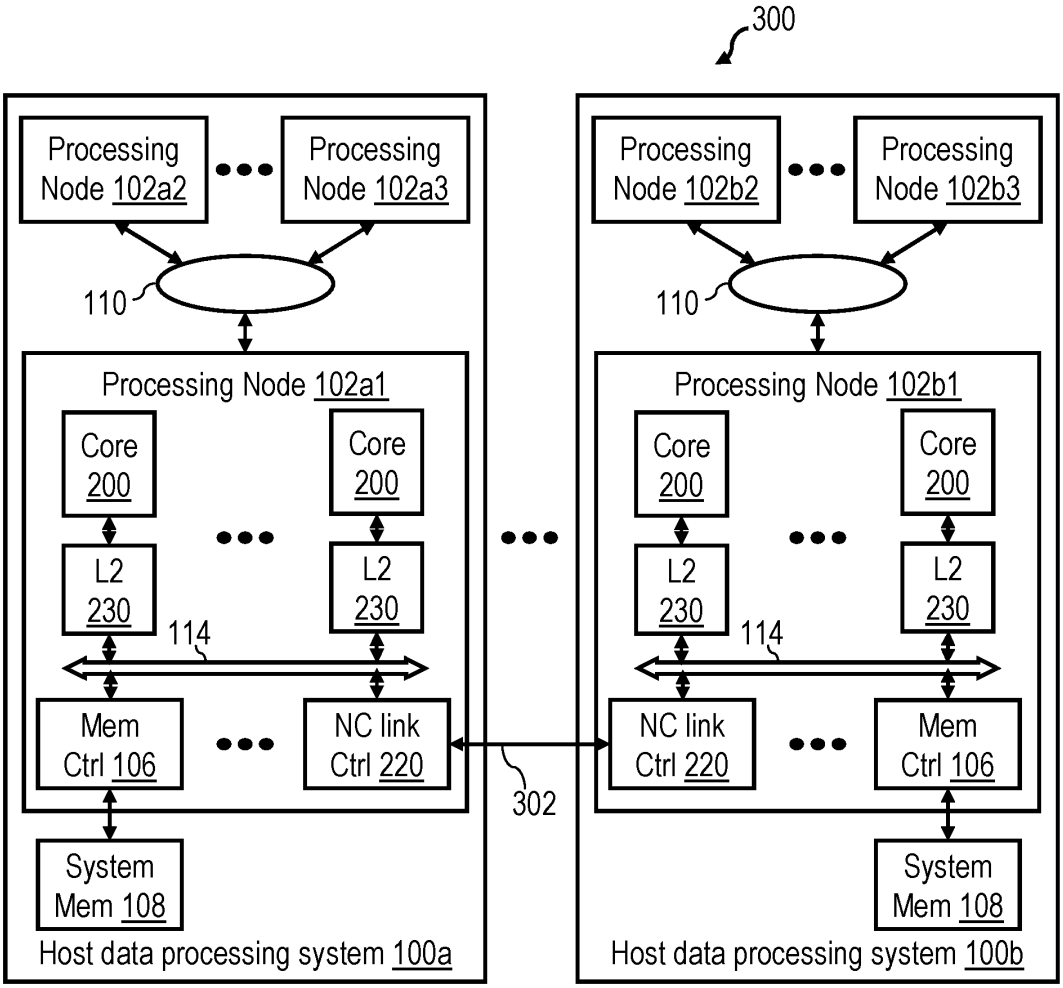
300
Processing Node 102a2
Processing Node 102a3
110
Processing Node 102a1
Core 200
Core 200
L2 230
L2 230
114
Mem Ctrl 106
NC link Ctrl 220
System Mem 108
Host data processing system 100a
302
Processing Node 102b2
Processing Node 102b3
110
Processing Node 102b1
Core 200
Core 200
L2 230
L2 230
114
NC link Ctrl 220
Mem Ctrl 106
System Mem 108
Host data processing system 100b

Figure 3

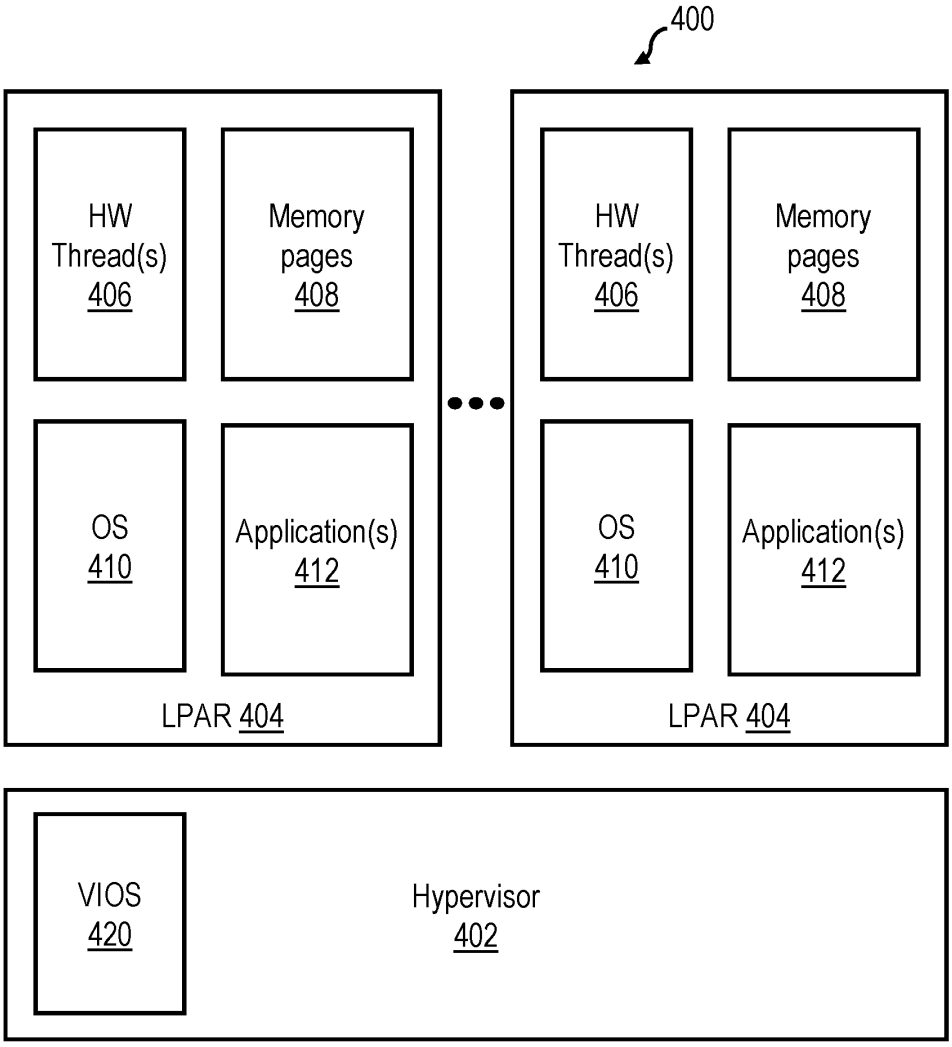

Figure 4

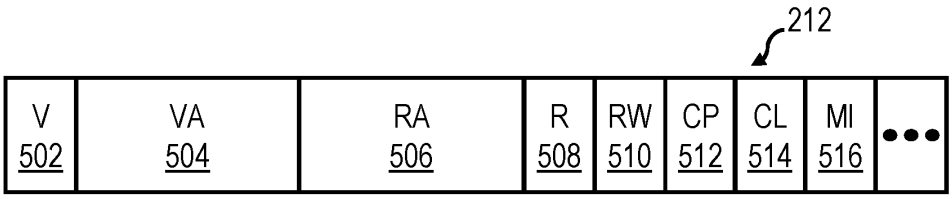

Figure 5

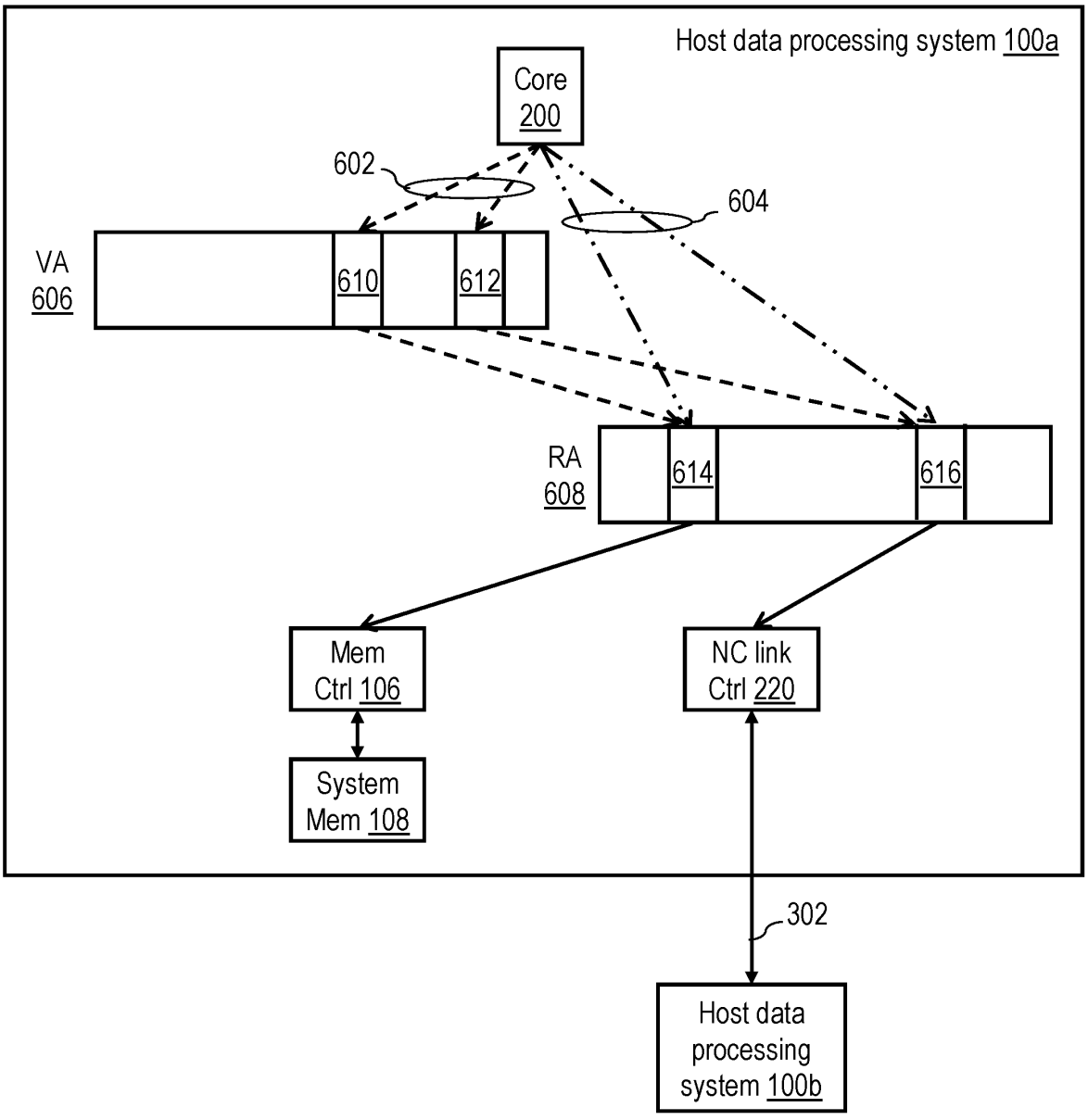
Host data processing system 100a
Core 200
602
604
VA 606
610
612
RA 608
614
616
Mem Ctrl 106
NC link Ctrl 220
System Mem 108
302
Host data processing system 100b

Figure 6



Figure 7

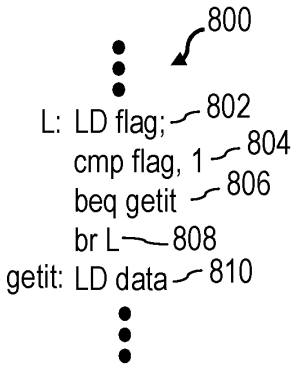

Figure 8

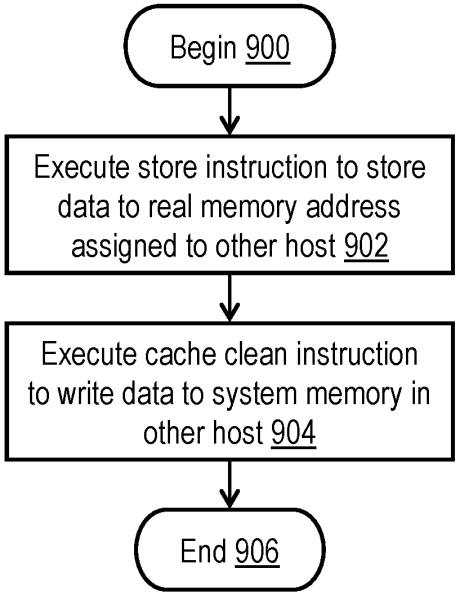

Figure 9

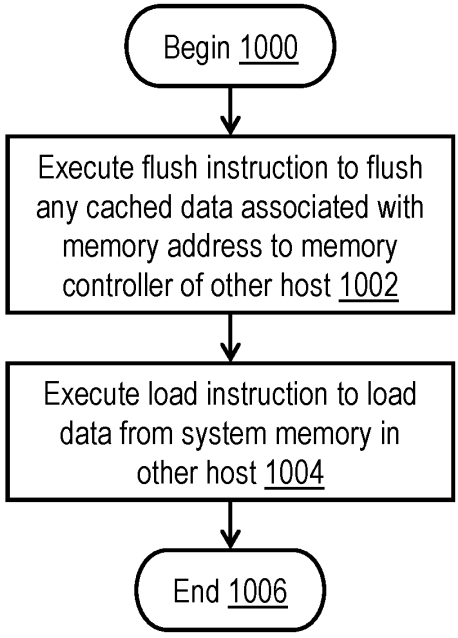

Figure 10

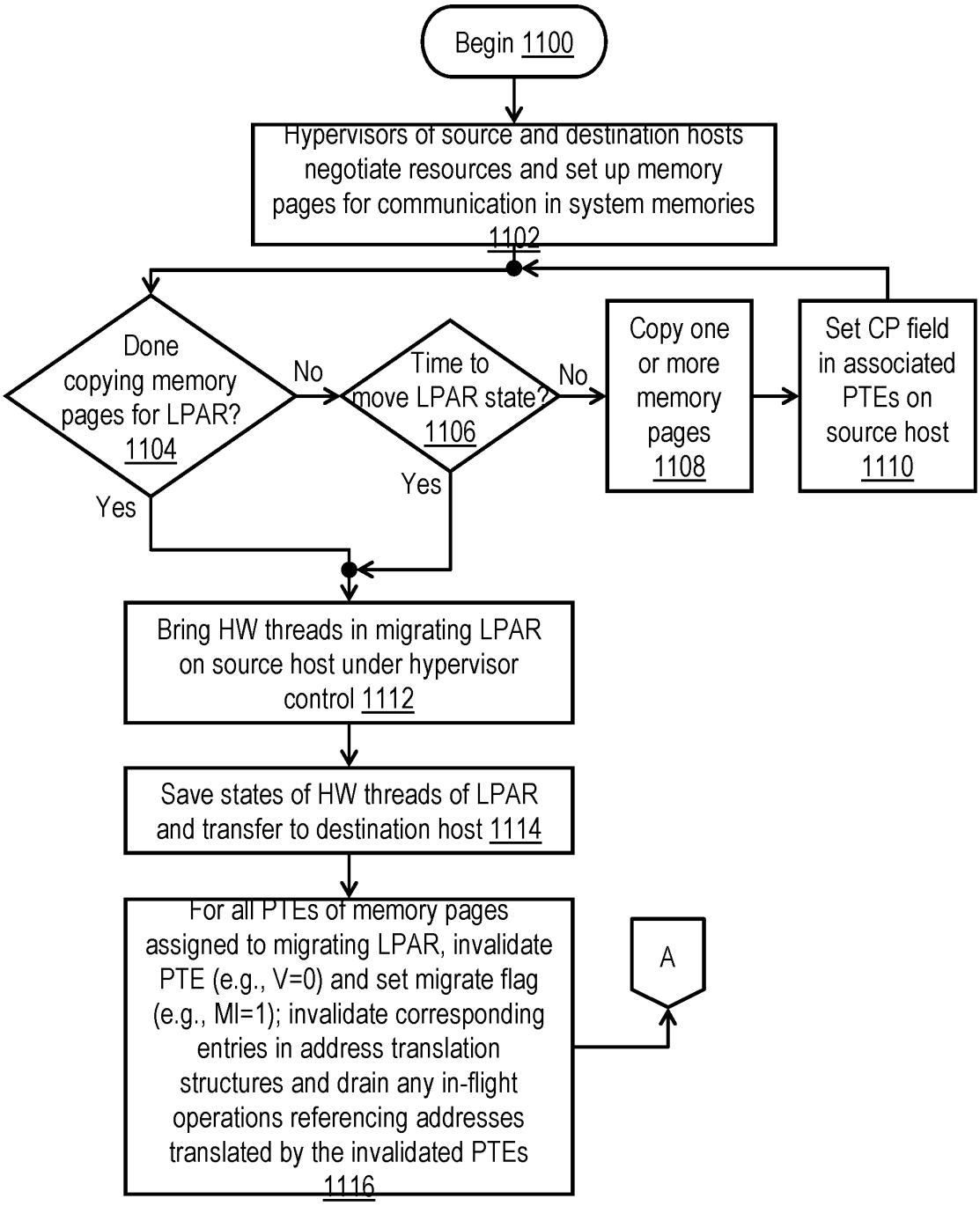
Begin 1100
Hypervisors of source and destination hosts negotiate resources and set up memory pages for communication in system memories 1102
Done copying memory pages for LPAR? 1104
No
Time to move LPAR state? 1106
No
Copy one or more memory pages 1108
Set CP field in associated PTEs on source host 1110
Yes
Yes
Bring HW threads in migrating LPAR on source host under hypervisor control 1112
Save states of HW threads of LPAR and transfer to destination host 1114
For all PTEs of memory pages assigned to migrating LPAR, invalidate PTE (e.g., V=0) and set migrate flag (e.g., MI=1); invalidate corresponding entries in address translation structures and drain any in-flight operations referencing addresses translated by the invalidated PTEs 1116
A

Figure 11A

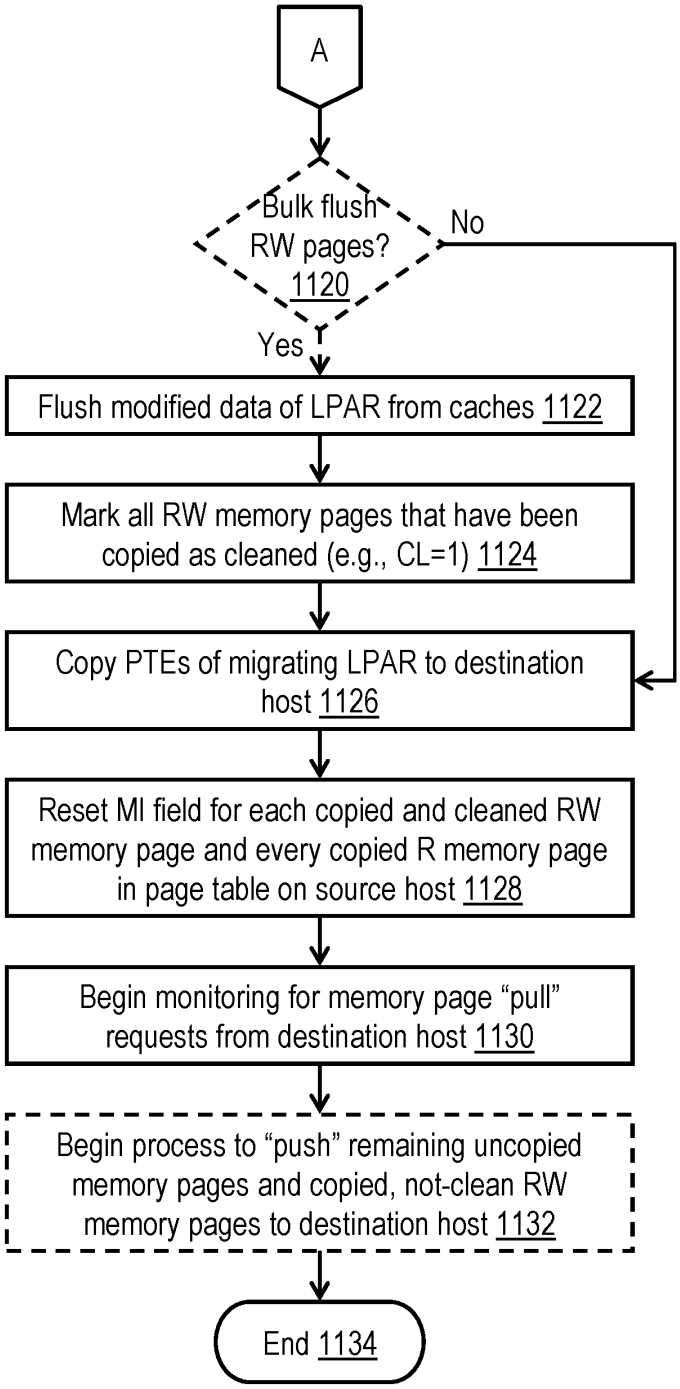
A
Bulk flush RW pages? 1120
No
Yes
Flush modified data of LPAR from caches 1122
Mark all RW memory pages that have been copied as cleaned (e.g., CL=1) 1124
Copy PTEs of migrating LPAR to destination host 1126
Reset MI field for each copied and cleaned RW memory page and every copied R memory page in page table on source host 1128
Begin monitoring for memory page "pull" requests from destination host 1130
Begin process to "push" remaining uncopied memory pages and copied, not-clean RW memory pages to destination host 1132
End 1134

Figure 11B

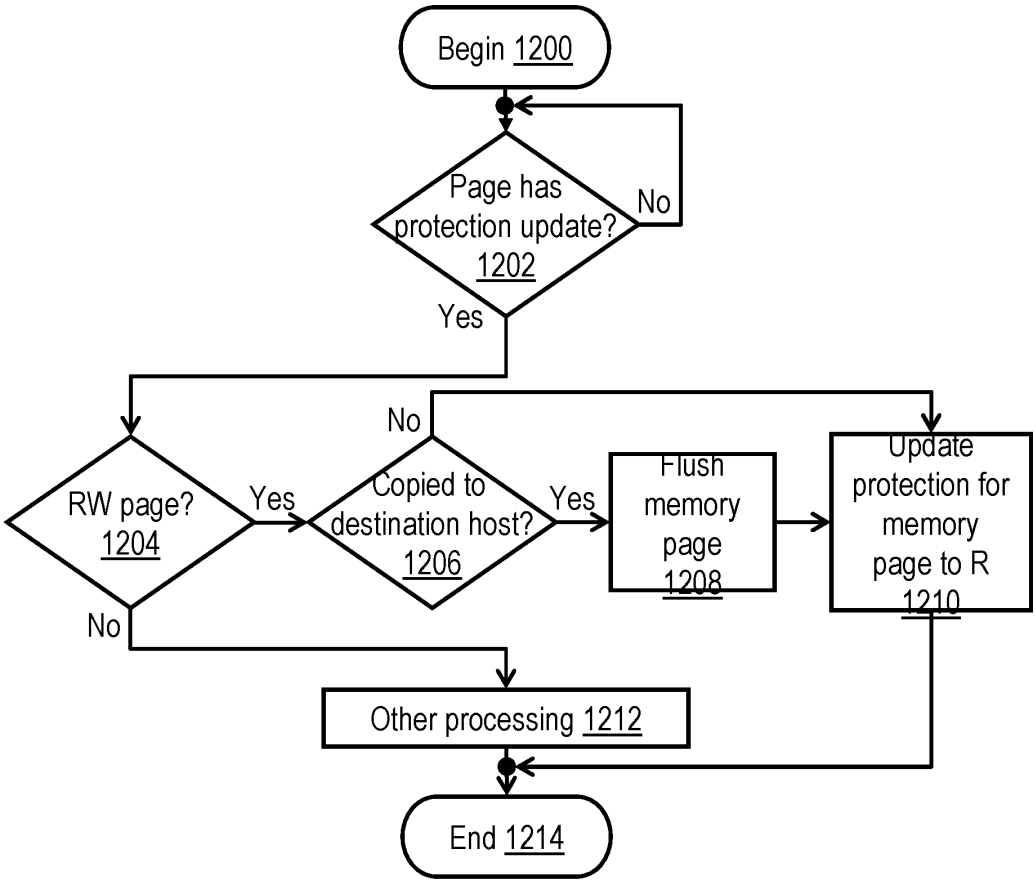
Begin 1200
Page has protection update? 1202
No
Yes
RW page? 1204
Yes
No
Copied to destination host? 1206
Yes
No
Flush memory page 1208
Update protection for memory page to R 1210
Other processing 1212
End 1214

Figure 12

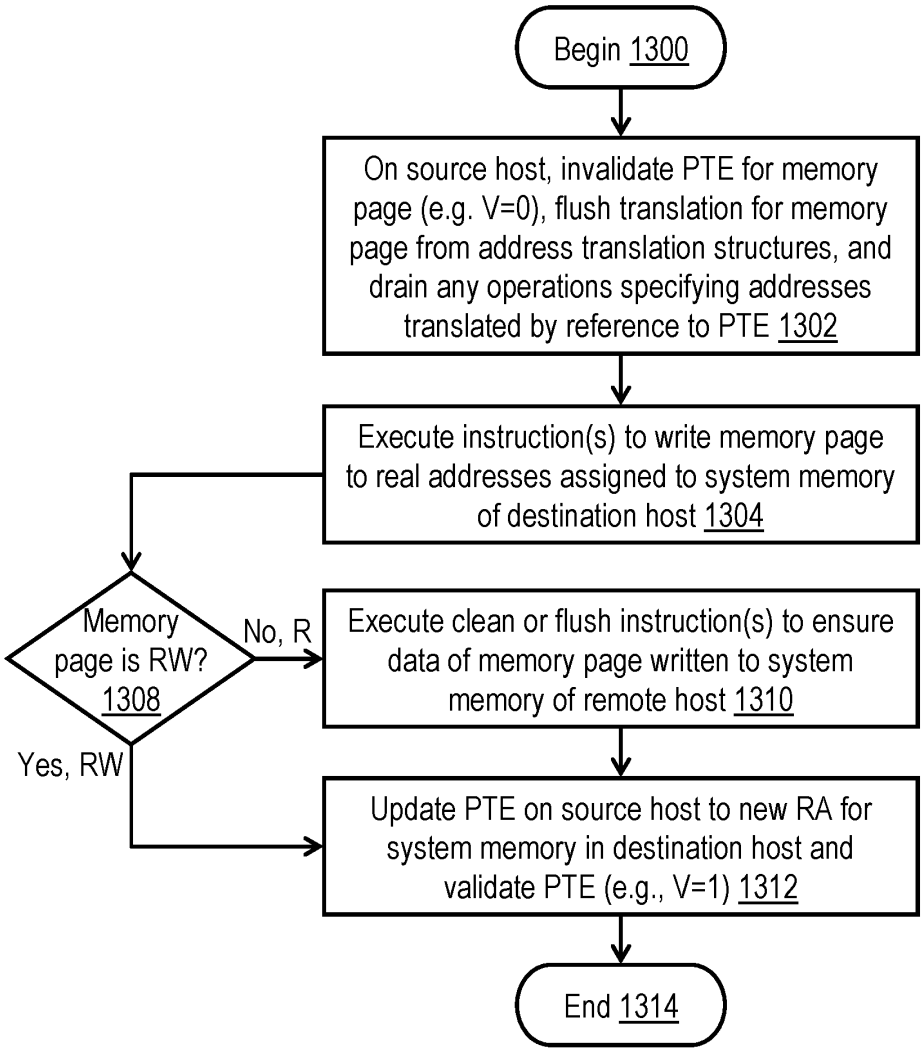
Begin 1300
On source host, invalidate PTE for memory page (e.g. V=0), flush translation for memory page from address translation structures, and drain any operations specifying addresses translated by reference to PTE 1302
Execute instruction(s) to write memory page to real addresses assigned to system memory of destination host 1304
Memory page is RW? 1308
No, R
Execute clean or flush instruction(s) to ensure data of memory page written to system memory of remote host 1310
Yes, RW
Update PTE on source host to new RA for system memory in destination host and validate PTE (e.g., V=1) 1312
End 1314

Figure 13

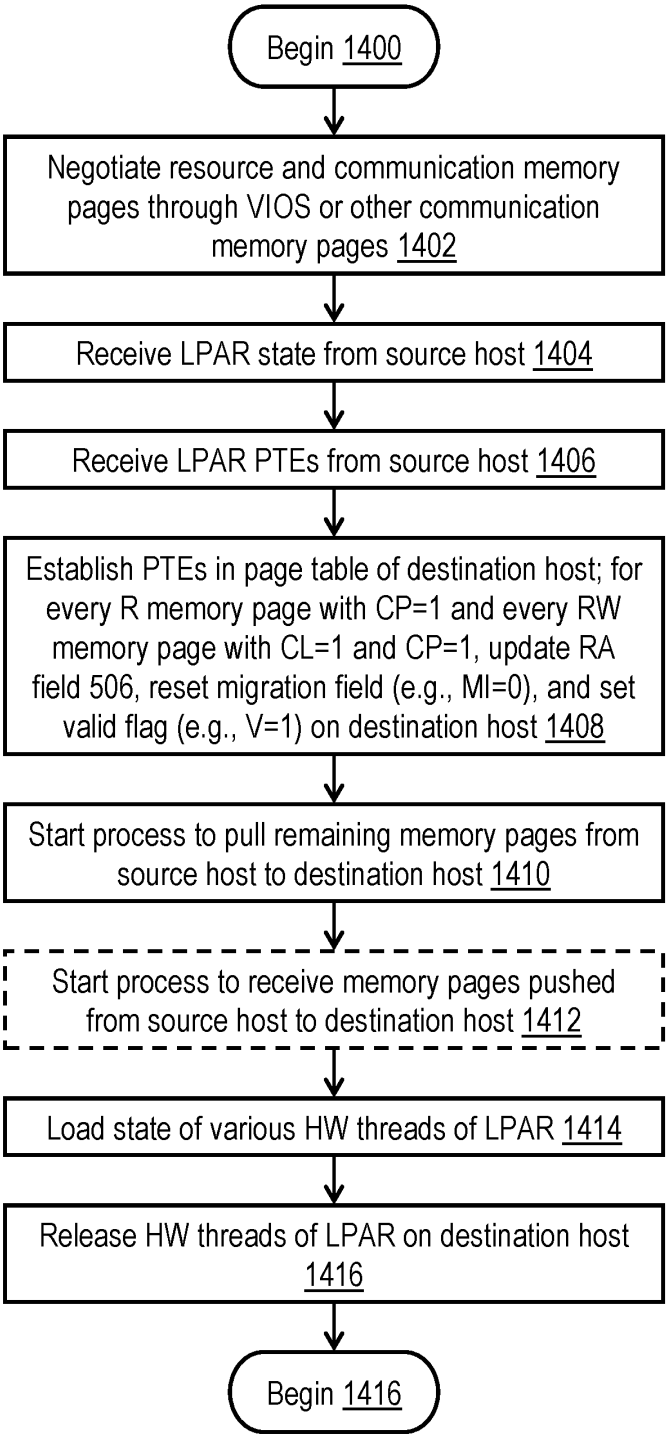
Begin 1400
Negotiate resource and communication memory pages through VIOS or other communication memory pages 1402
Receive LPAR state from source host 1404
Receive LPAR PTEs from source host 1406
Establish PTEs in page table of destination host; for every R memory page with CP=1 and every RW memory page with CL=1 and CP=1, update RA field 506, reset migration field (e.g., MI=0), and set valid flag (e.g., V=1) on destination host 1408
Start process to pull remaining memory pages from source host to destination host 1410
Start process to receive memory pages pushed from source host to destination host 1412
Load state of various HW threads of LPAR 1414
Release HW threads of LPAR on destination host 1416
Begin 1416

Figure 14

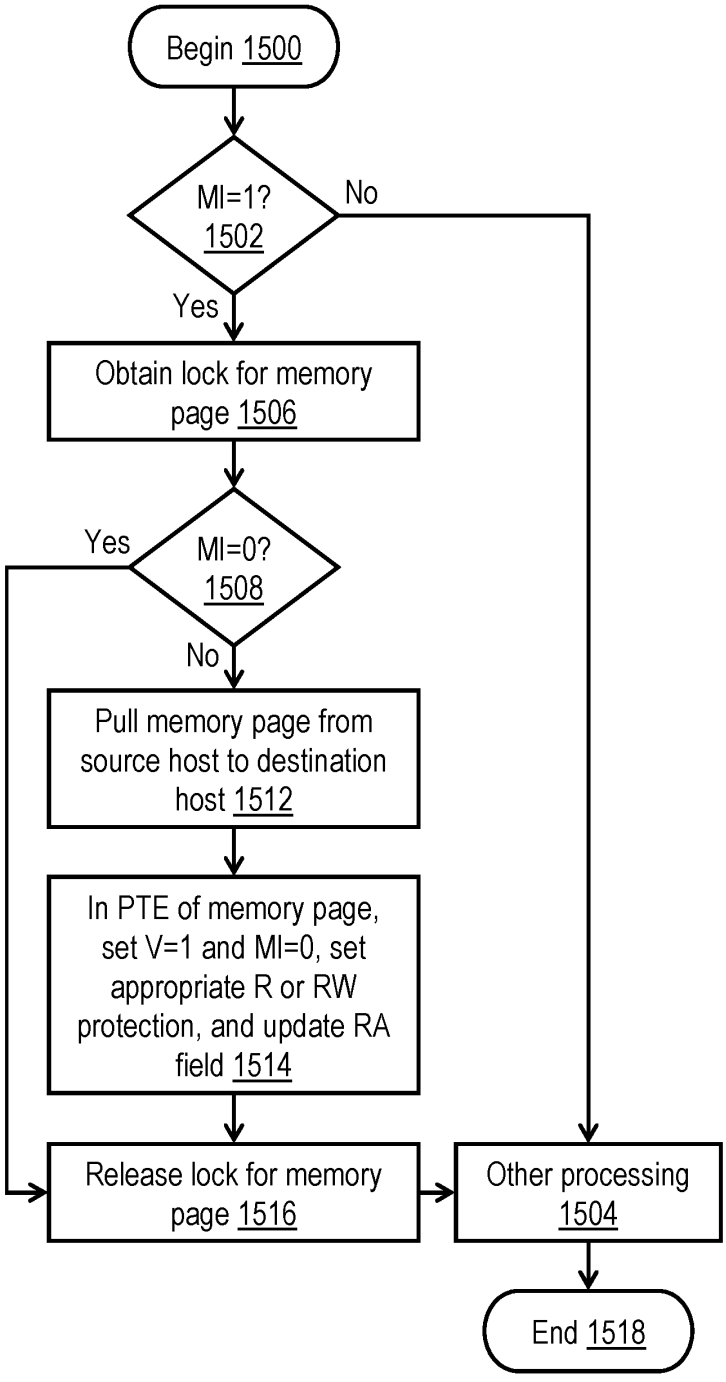
Begin 1500
MI=1? 1502
No
Yes
Obtain lock for memory page 1506
MI=0? 1508
Yes
No
Pull memory page from source host to destination host 1512
In PTE of memory page, set V=1 and MI=0, set appropriate R or RW protection, and update RA field 1514
Release lock for memory page 1516
Other processing 1504
End 1518

Figure 15

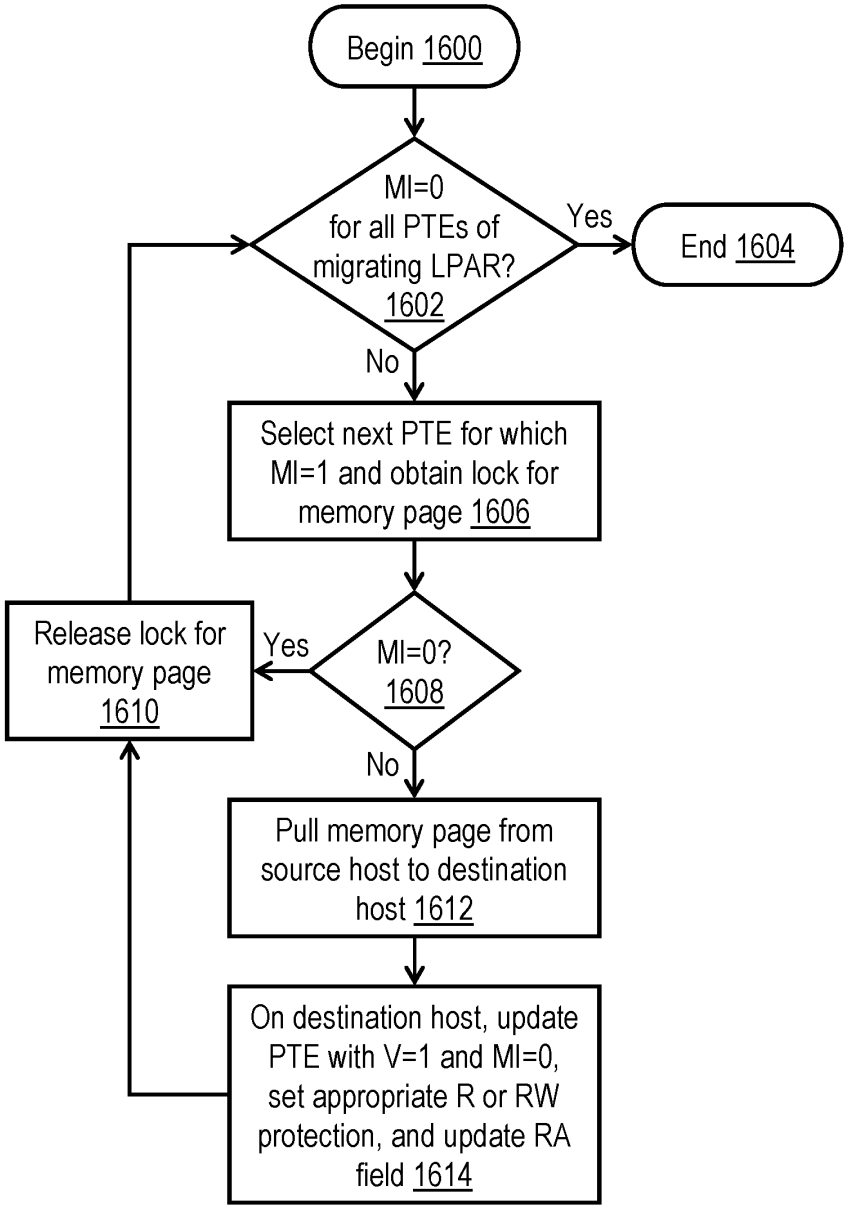
Begin 1600
MI=0 for all PTEs of migrating LPAR? 1602
Yes
End 1604
No
Select next PTE for which MI=1 and obtain lock for memory page 1606
MI=0? 1608
Yes
Release lock for memory page 1610
No
Pull memory page from source host to destination host 1612
On destination host, update PTE with V=1 and MI=0, set appropriate R or RW protection, and update RA field 1614

Figure 16

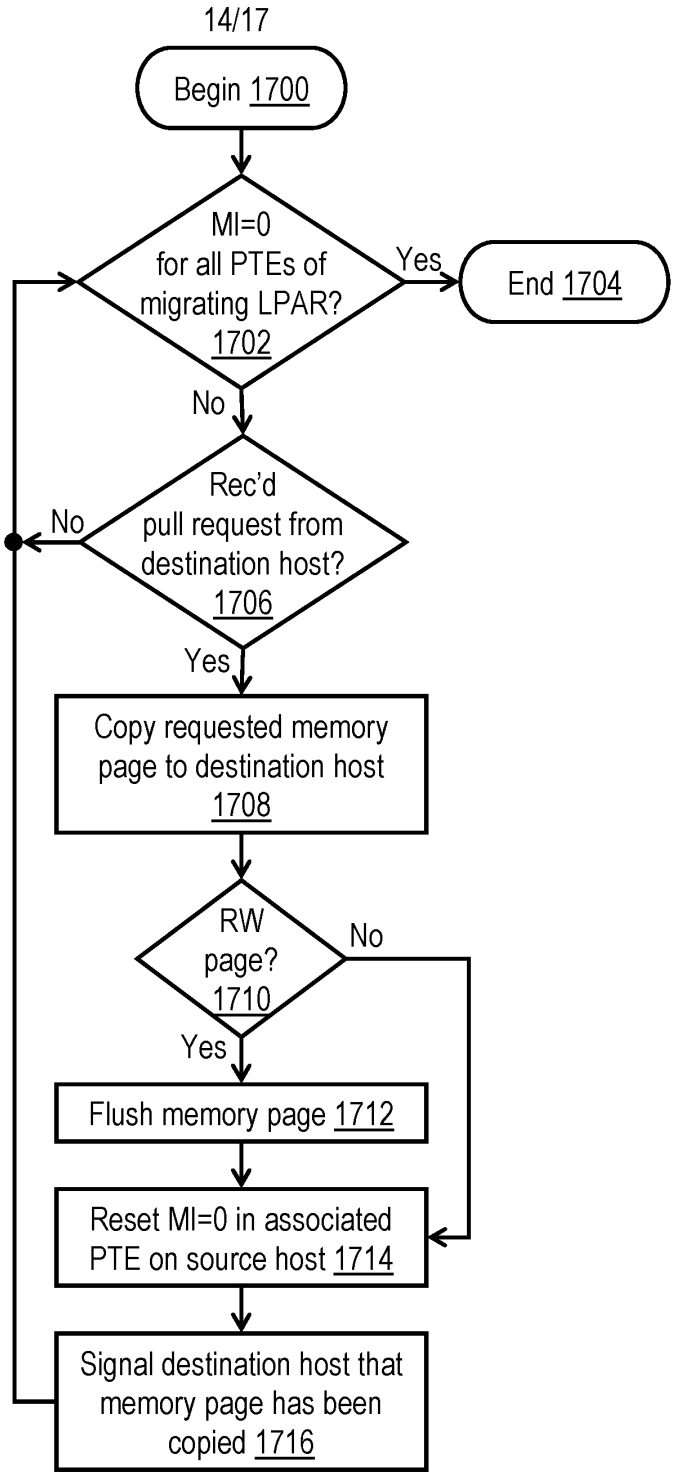
Begin 1700
MI=0 for all PTEs of migrating LPAR? 1702
Yes
End 1704
No
Rec'd pull request from destination host? 1706
No
Yes
Copy requested memory page to destination host 1708
RW page? 1710
No
Yes
Flush memory page 1712
Reset MI=0 in associated PTE on source host 1714
Signal destination host that memory page has been copied 1716

Figure 17

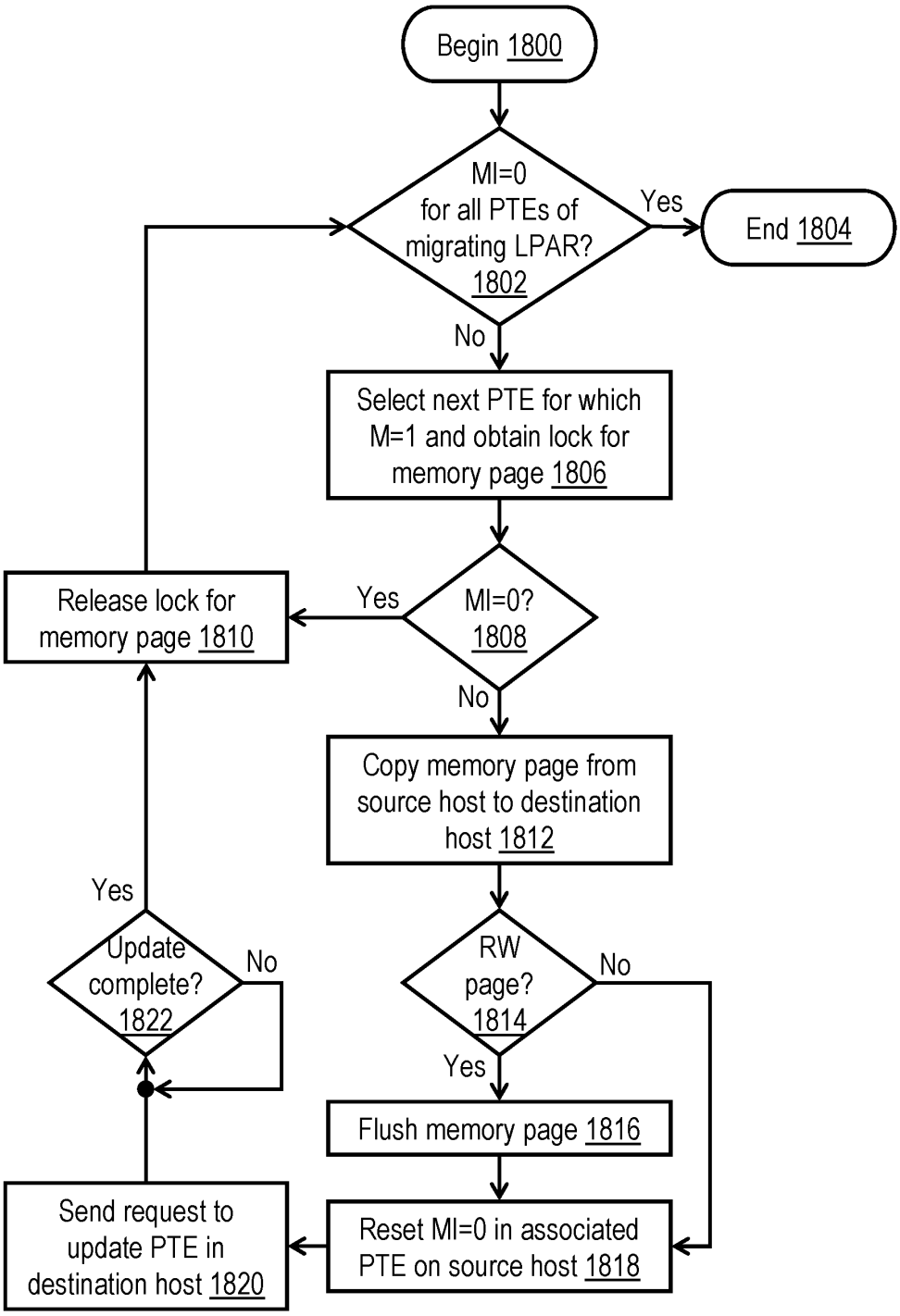
Begin 1800
MI=0 for all PTEs of migrating LPAR? 1802
Yes
End 1804
No
Select next PTE for which M=1 and obtain lock for memory page 1806
MI=0? 1808
Yes
Release lock for memory page 1810
No
Copy memory page from source host to destination host 1812
RW page? 1814
No
Yes
Flush memory page 1816
Reset MI=0 in associated PTE on source host 1818
Send request to update PTE in destination host 1820
Update complete? 1822
No
Yes

Figure 18

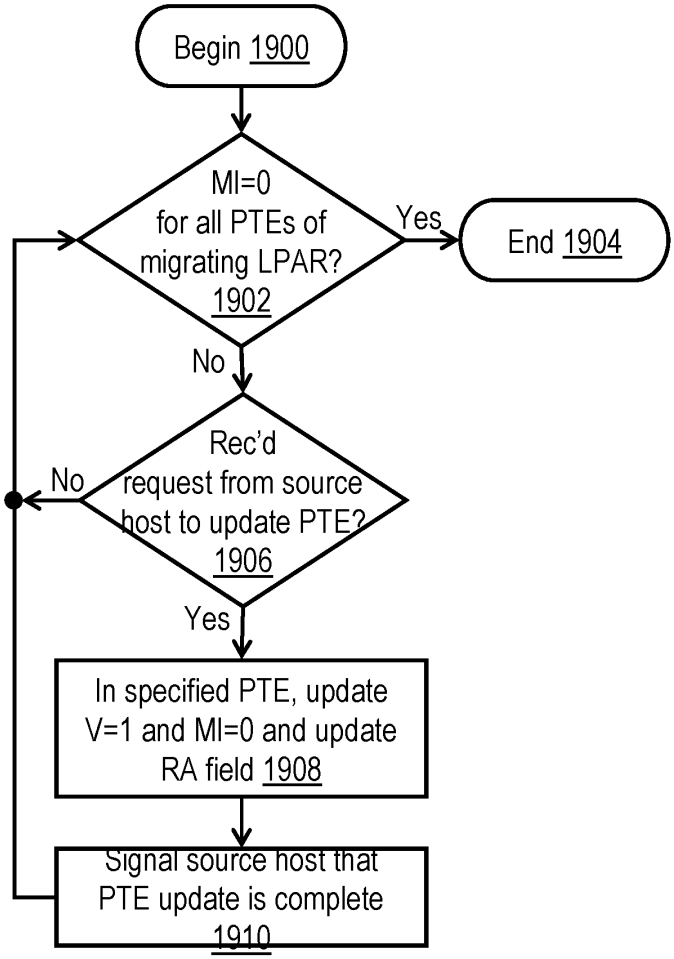
Begin 1900
MI=0 for all PTEs of migrating LPAR? 1902
Yes
End 1904
No
Rec'd request from source host to update PTE? 1906
No
Yes
In specified PTE, update V=1 and MI=0 and update RA field 1908
Signal source host that PTE update is complete 1910

Figure 19

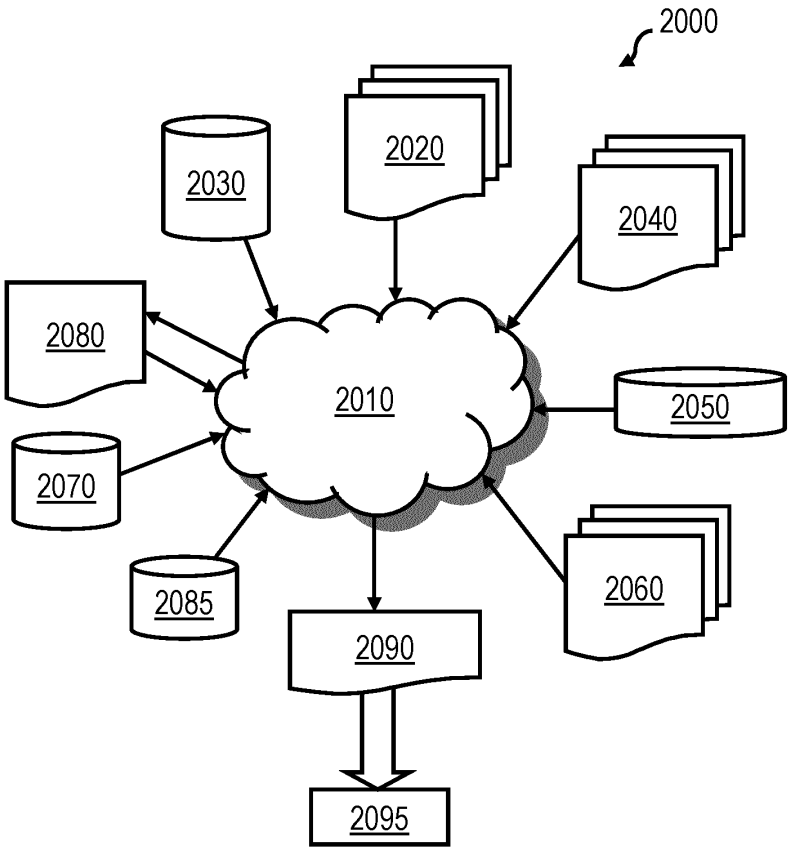

Figure 20