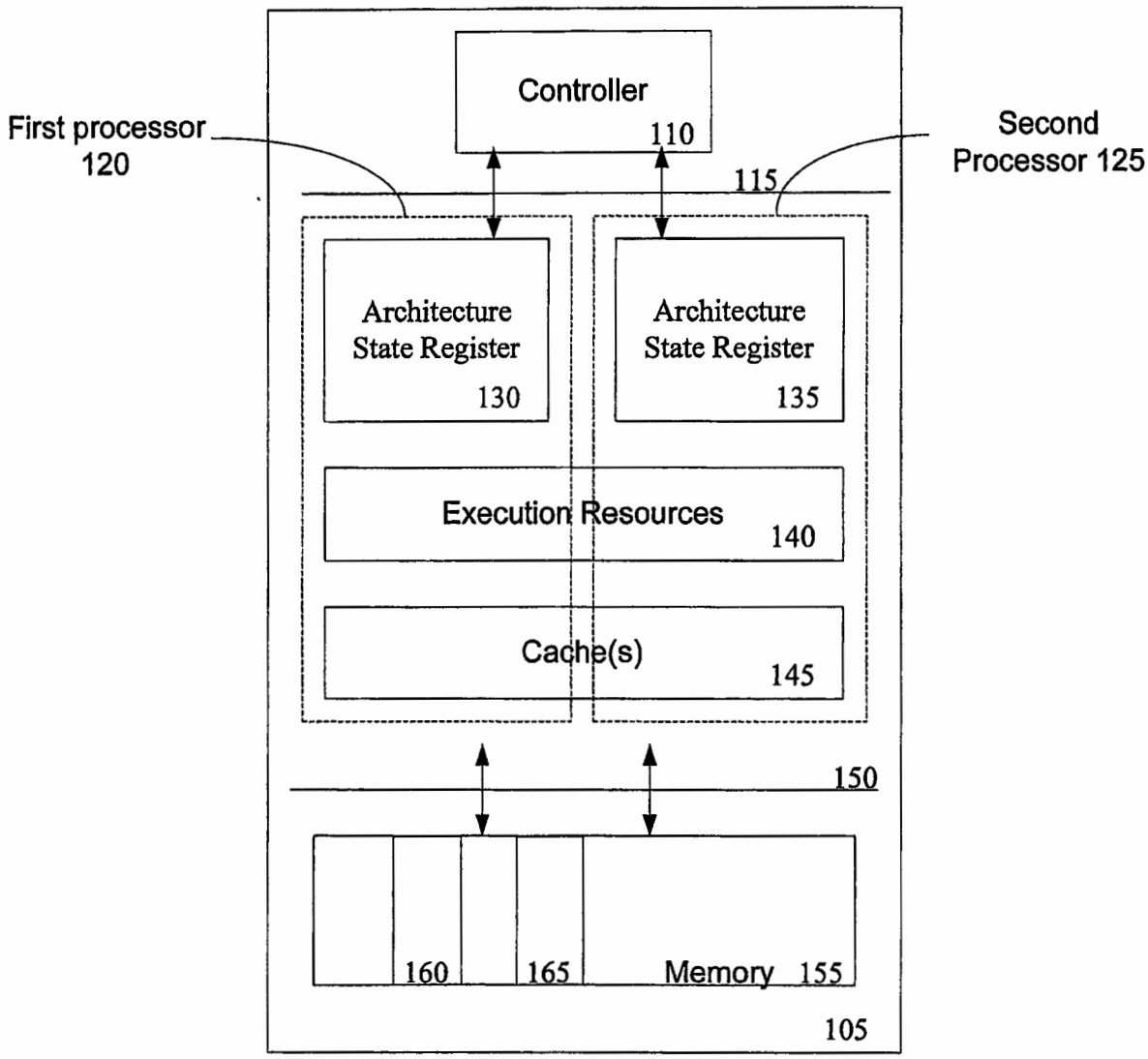

First processor
120
Second
Processor 125
Controller
110
115
Architecture
State Register
130
Architecture
State Register
135
Execution Resources
140
Cache(s)
145
150
160
165
Memory 155
105

FIG. 1

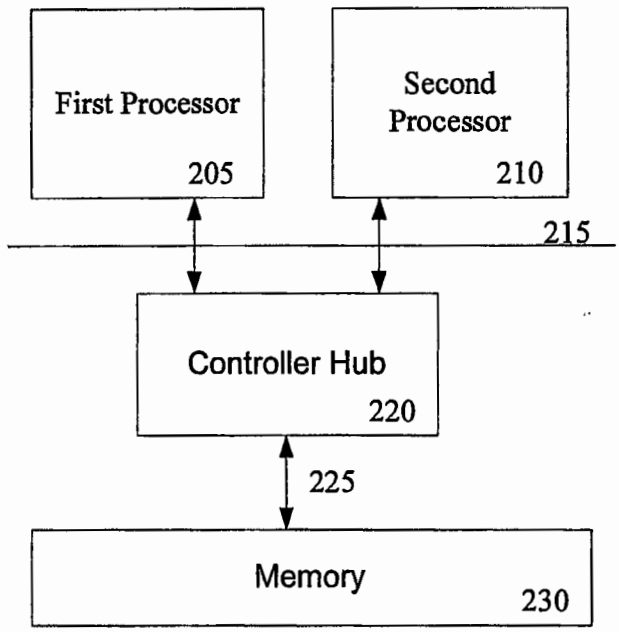

First Processor
205
Second
Processor
210
215
Controller Hub
220
225
Memory
230

FIG. 2

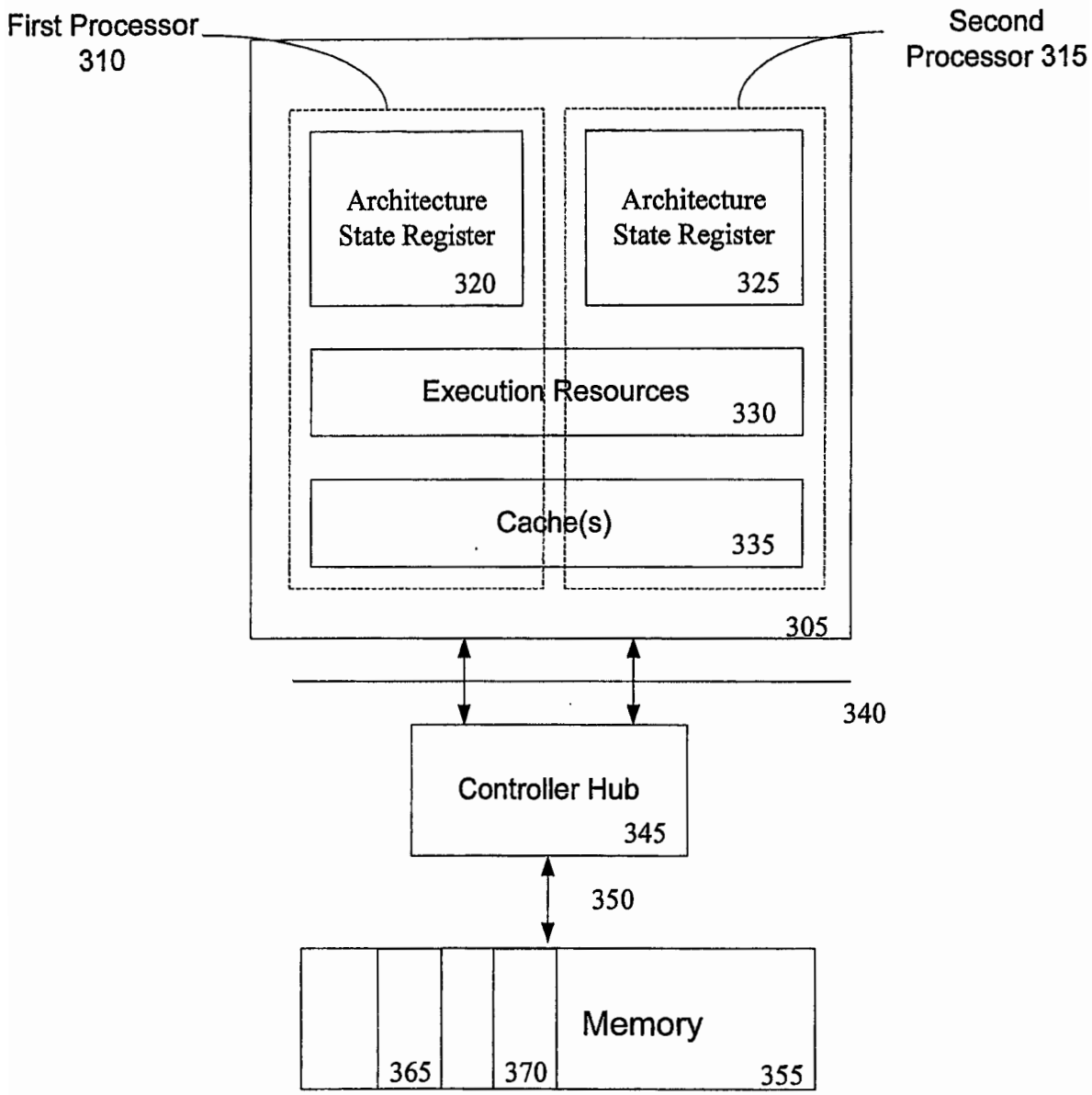

First Processor 310
Second Processor 315
Architecture State Register 320
Architecture State Register 325
Execution Resources 330
Cache(s) 335
305
340
Controller Hub 345
350
Memory 355
365
370

FIG. 3

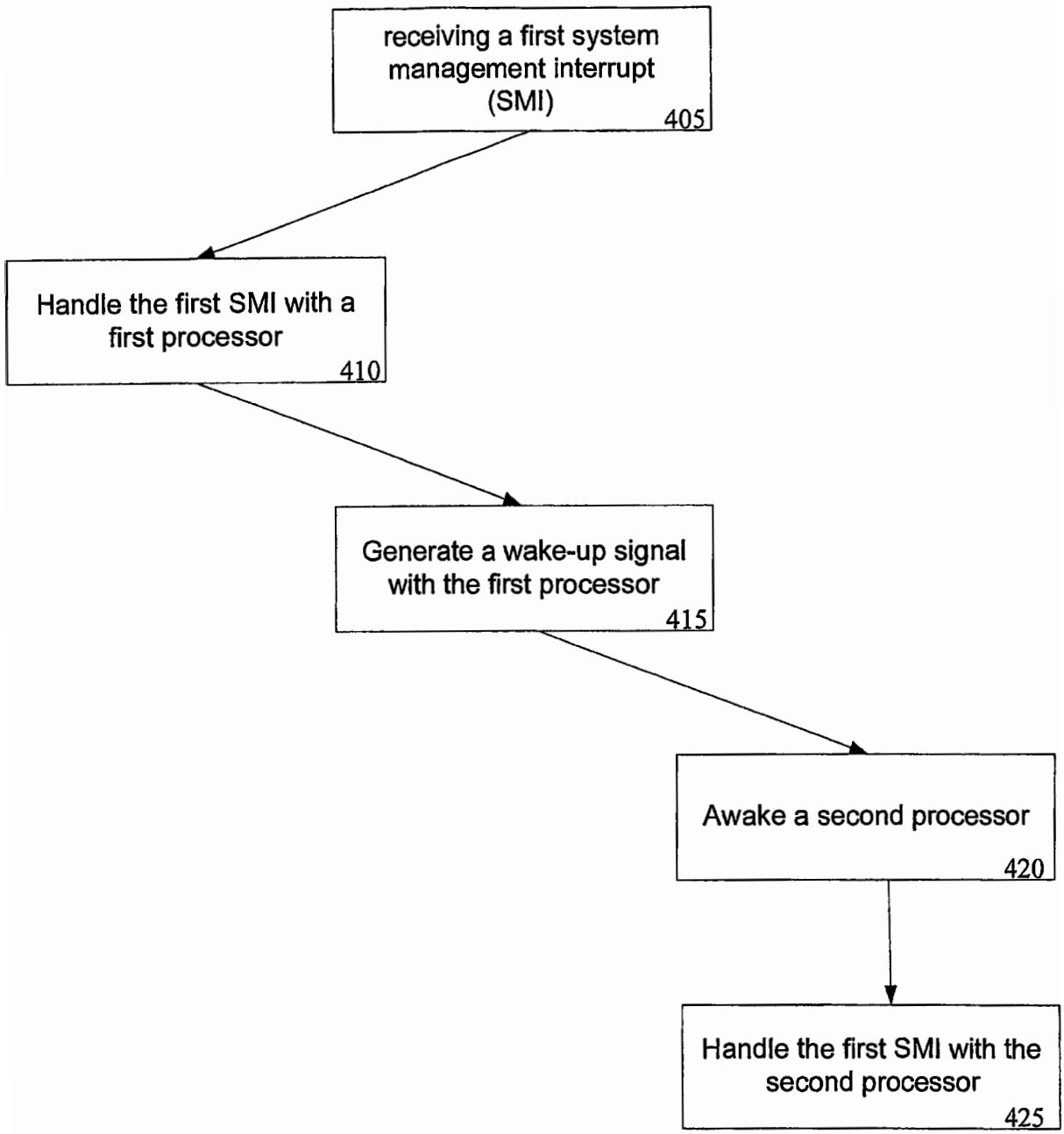

receiving a first system management interrupt (SMI)
405
Handle the first SMI with a first processor
410
Generate a wake-up signal with the first processor
415
Awake a second processor
420
Handle the first SMI with the second processor
425

FIG. 4

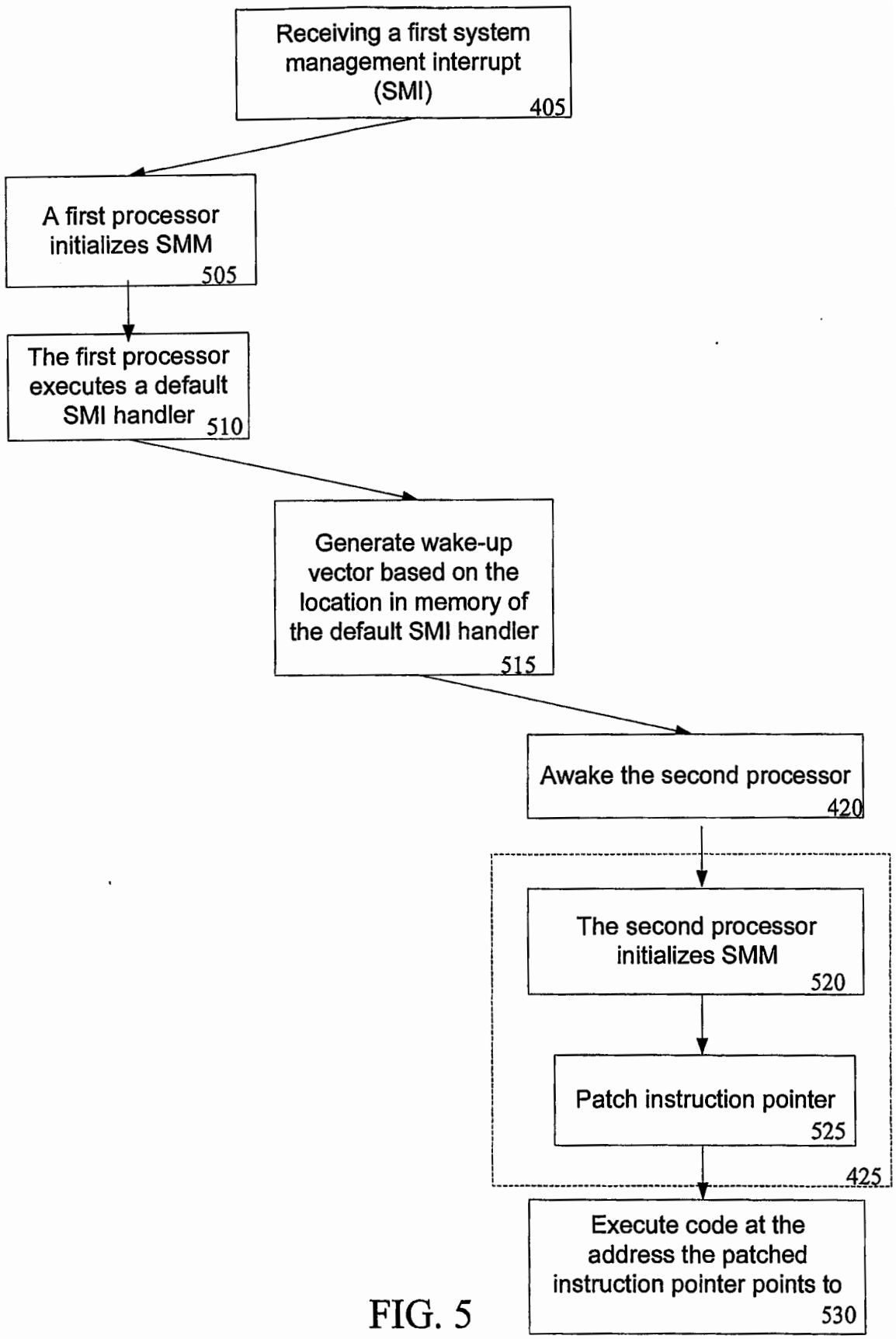

Receiving a first system management interrupt (SMI)
405
A first processor initializes SMM
505
The first processor executes a default SMI handler
510
Generate wake-up vector based on the location in memory of the default SMI handler
515
Awake the second processor
420
The second processor initializes SMM
520
Patch instruction pointer
525
425
Execute code at the address the patched instruction pointer points to
530

FIG. 5

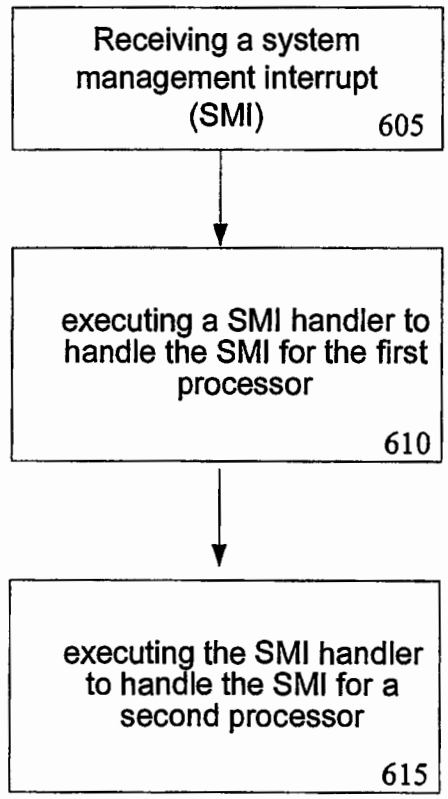

Receiving a system management interrupt (SMI)
605
executing a SMI handler to handle the SMI for the first processor
610
executing the SMI handler to handle the SMI for a second processor
615

FIG. 6

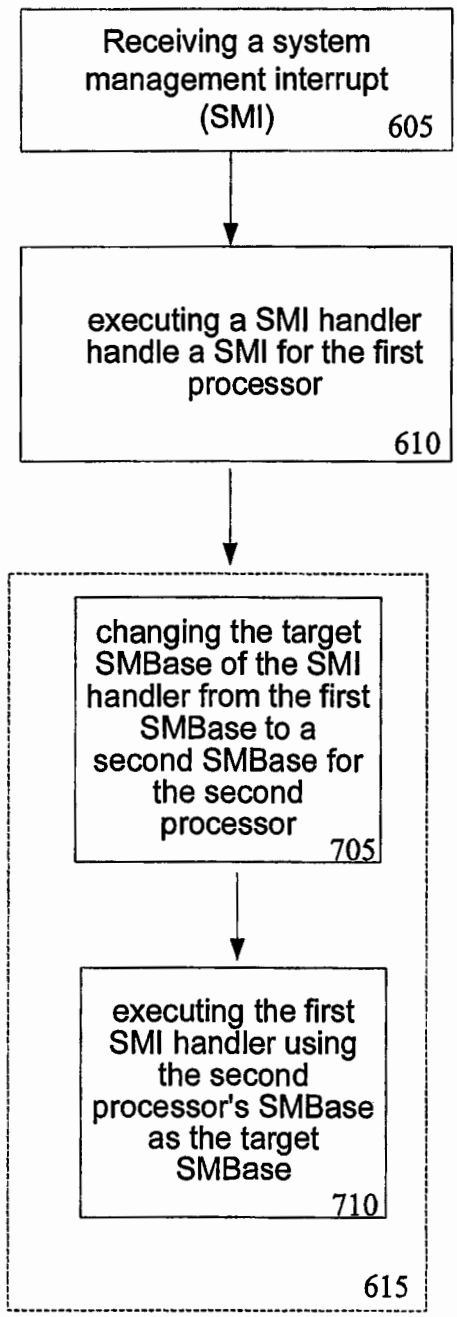

Receiving a system management interrupt (SMI)
605
executing a SMI handler handle a SMI for the first processor
610
changing the target SMBase of the SMI handler from the first SMBase to a second SMBase for the second processor
705
executing the first SMI handler using the second processor's SMBase as the target SMBase
710
615

FIG. 7

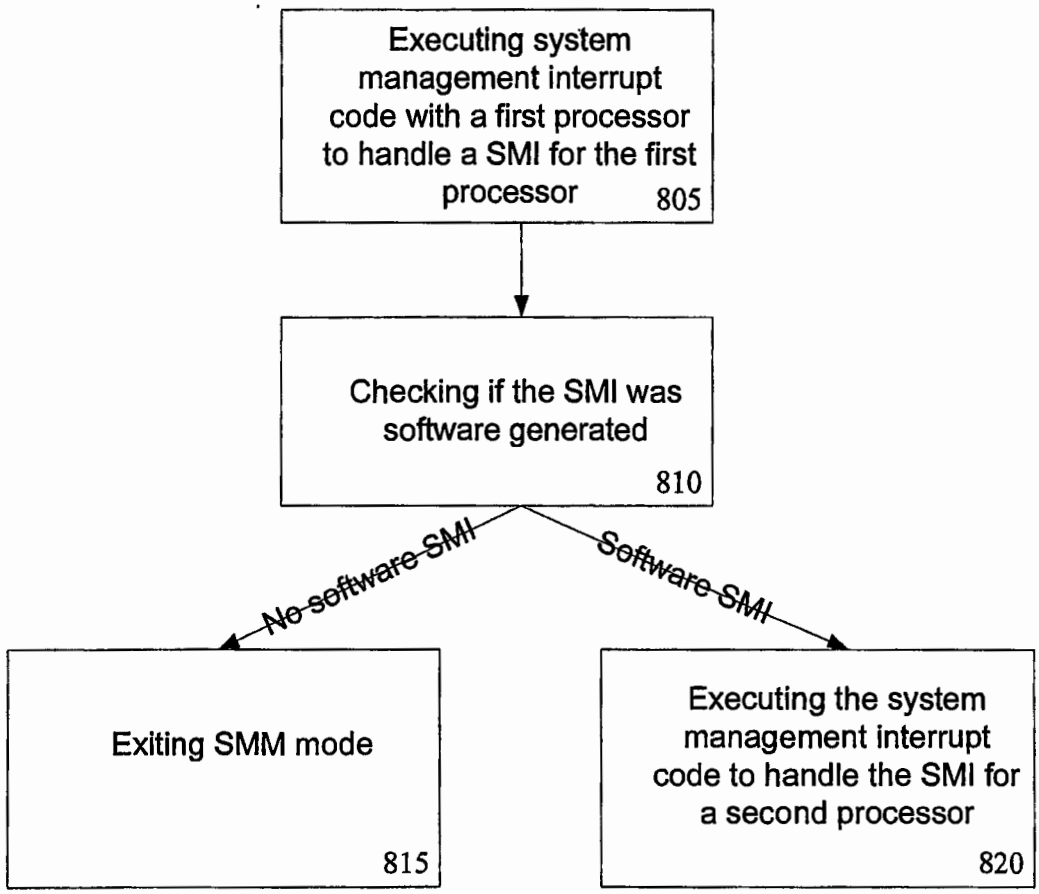

Executing system management interrupt code with a first processor to handle a SMI for the first processor
805
Checking if the SMI was software generated
810
No software SMI
Software SMI
Exiting SMM mode
815
Executing the system management interrupt code to handle the SMI for a second processor
820

FIG. 8

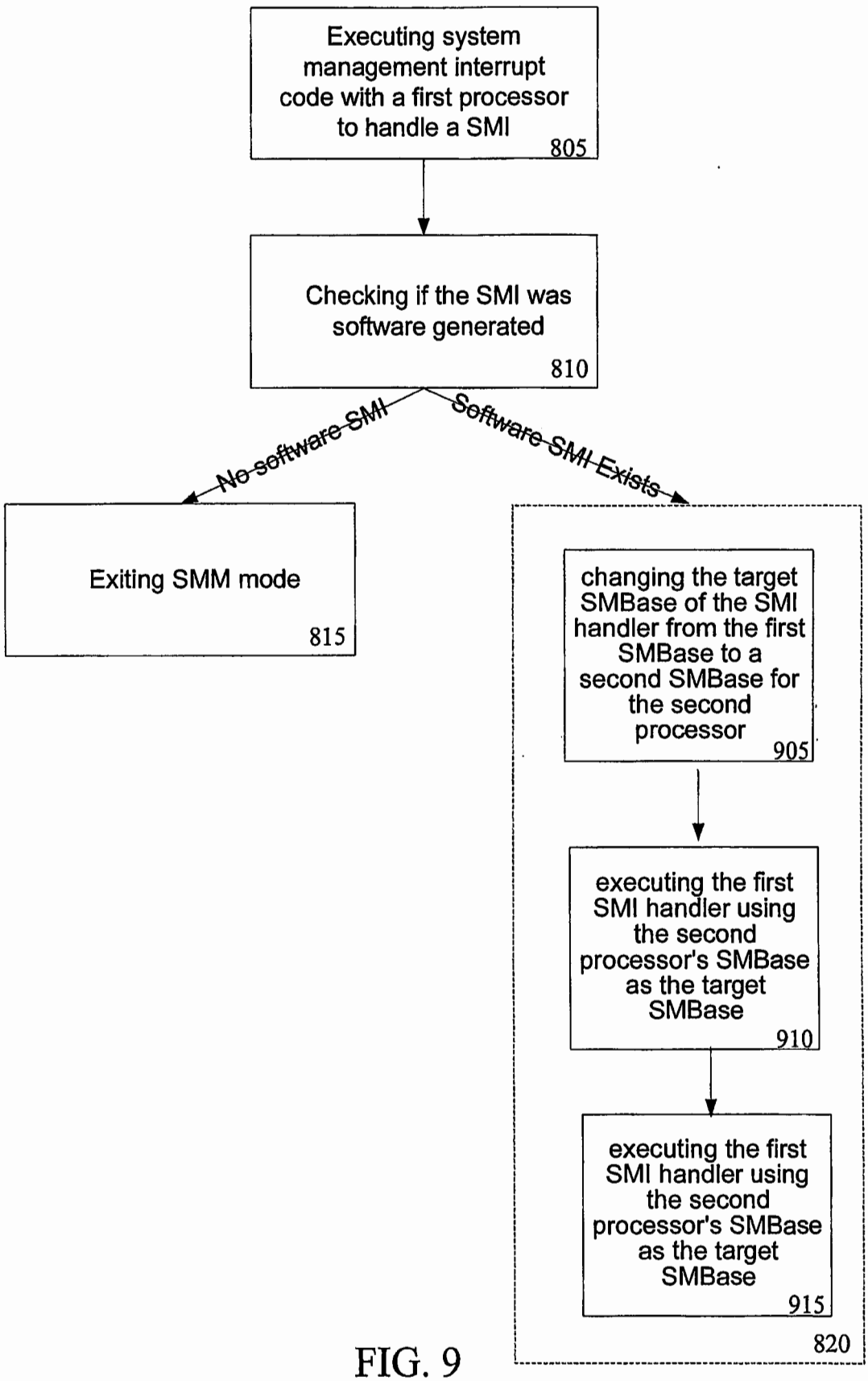

Executing system management interrupt code with a first processor to handle a SMI
805
Checking if the SMI was software generated
810
No software SMI
Software SMI Exists
Exiting SMM mode
815
changing the target SMBase of the SMI handler from the first SMBase to a second SMBase for the second processor
905
executing the first SMI handler using the second processor's SMBase as the target SMBase
910
executing the first SMI handler using the second processor's SMBase as the target SMBase
915
820

FIG. 9