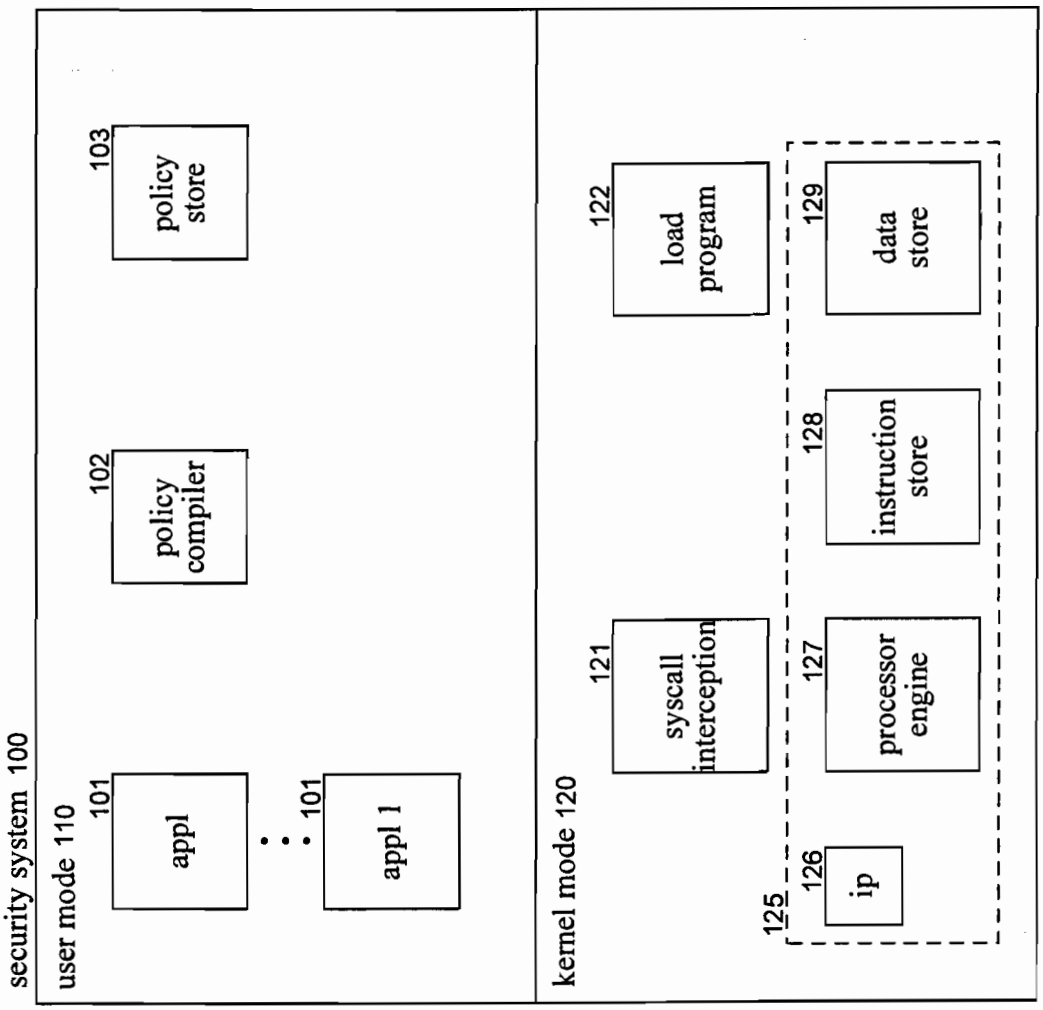

security system 100
user mode 110
101
appl
101
appl 1
102
policy compiler
103
policy store
kernel mode 120
121
syscall interception
122
load program
125
126
ip
127
processor engine
128
instruction store
129
data store

FIG. 1

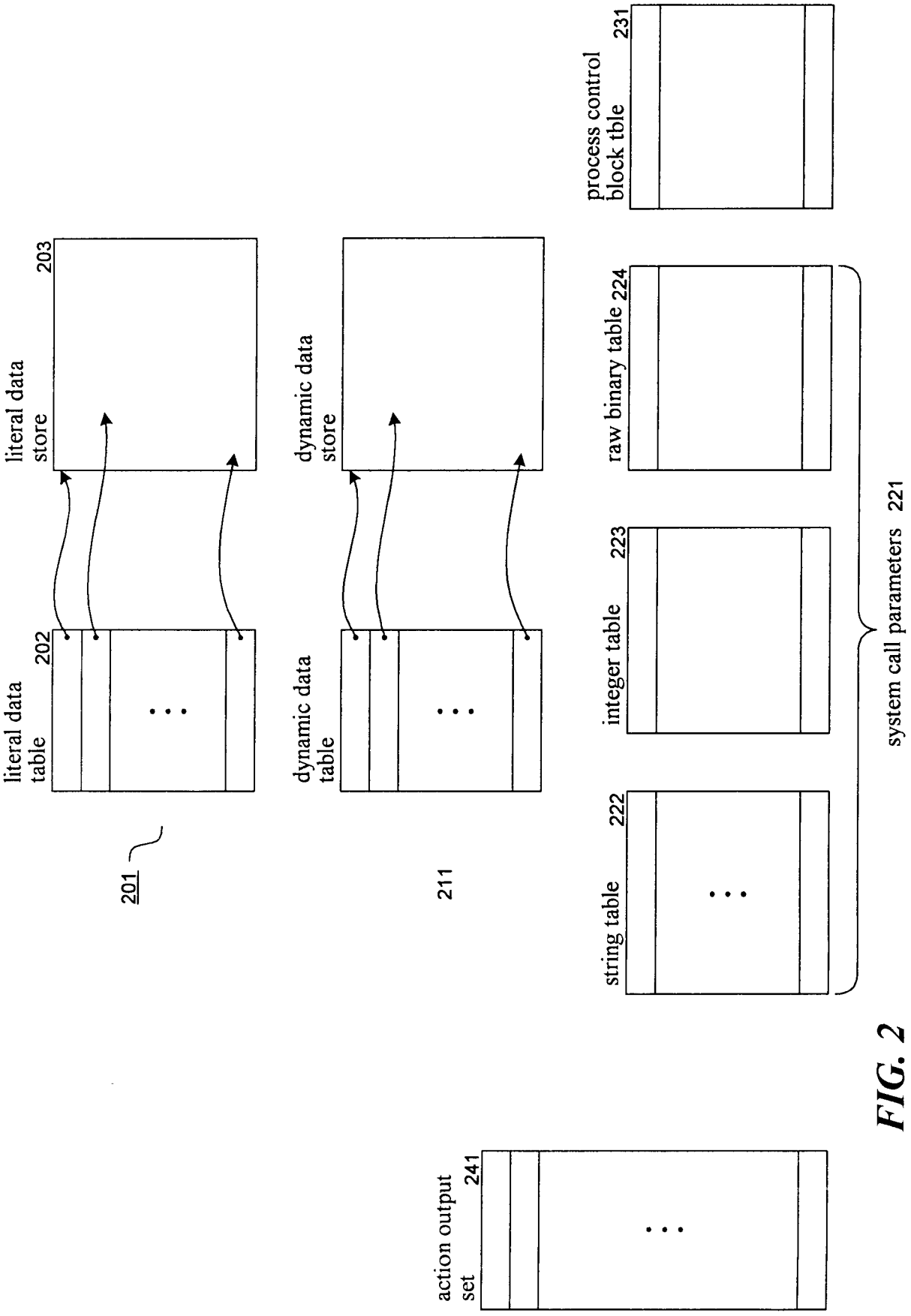
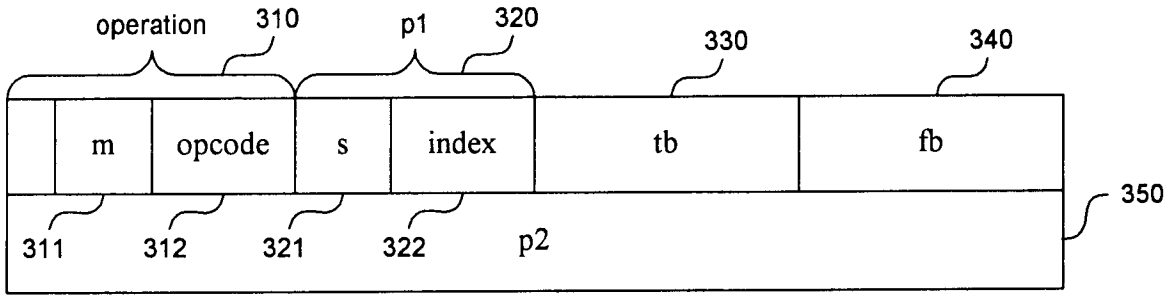

literal data store
203
literal data table
202
201
dynamic data store
dynamic data table
211
process control block tble
231
raw binary table
224
integer table
223
string table
222
system call parameters
221
action output set
241

FIG. 2



m
01 - immediate data
10 - dynamic data
11 - literal data

s
01 - syscall parameter
10 - process control block

index - s = 01
001xxx - string table
010xxx - integer table
100xxx - block table

index - s = 10
0 - application id
1 - session id of thread

FIG. 3

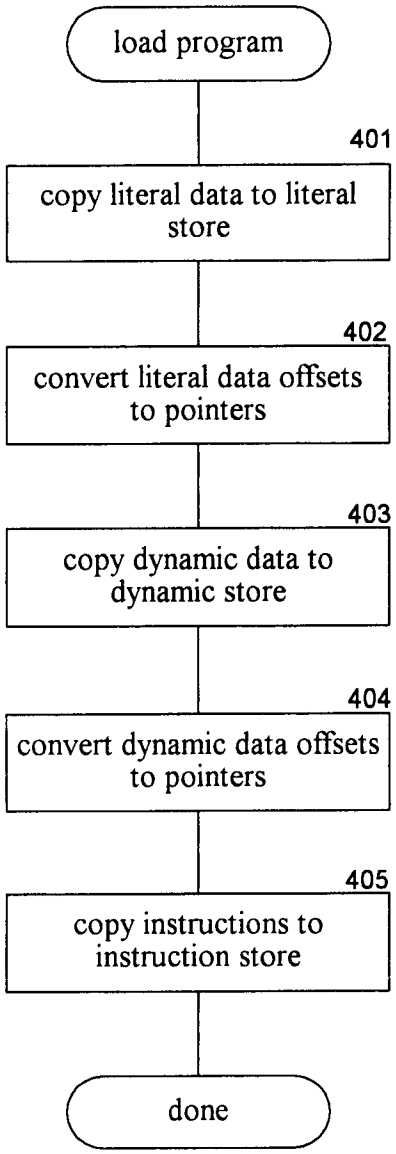

load program
401
copy literal data to literal store
402
convert literal data offsets to pointers
403
copy dynamic data to dynamic store
404
convert dynamic data offsets to pointers
405
copy instructions to instruction store
done

FIG. 4

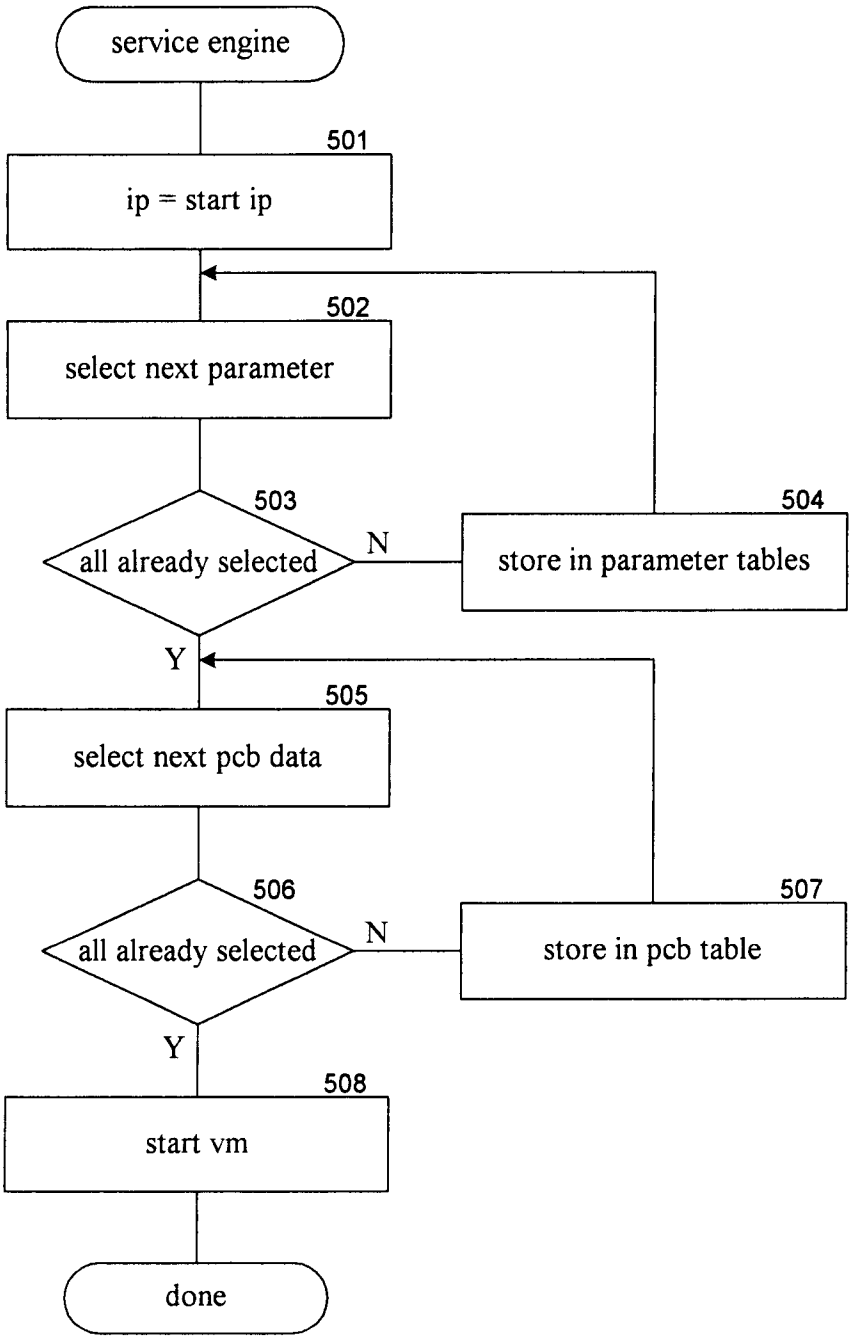

service engine
501
ip = start ip
502
select next parameter
503
all already selected
N
504
store in parameter tables
Y
505
select next pcb data
506
all already selected
N
507
store in pcb table
Y
508
start vm
done

FIG. 5

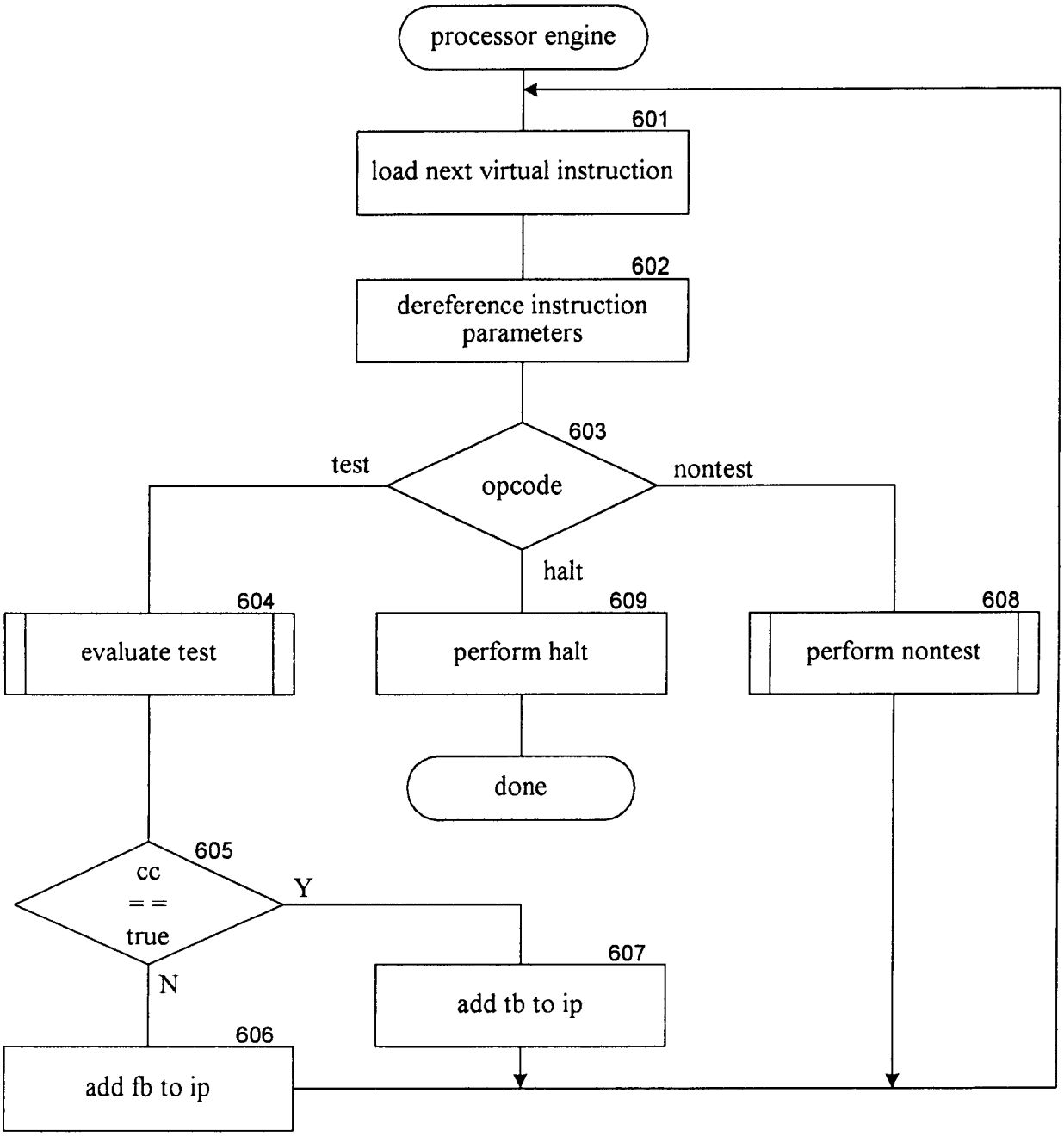

processor engine
601
load next virtual instruction
602
dereference instruction parameters
603
opcode
test
nontest
halt
604
evaluate test
609
perform halt
608
perform nontest
done
605
cc == true
Y
N
607
add tb to ip
606
add fb to ip

FIG. 6

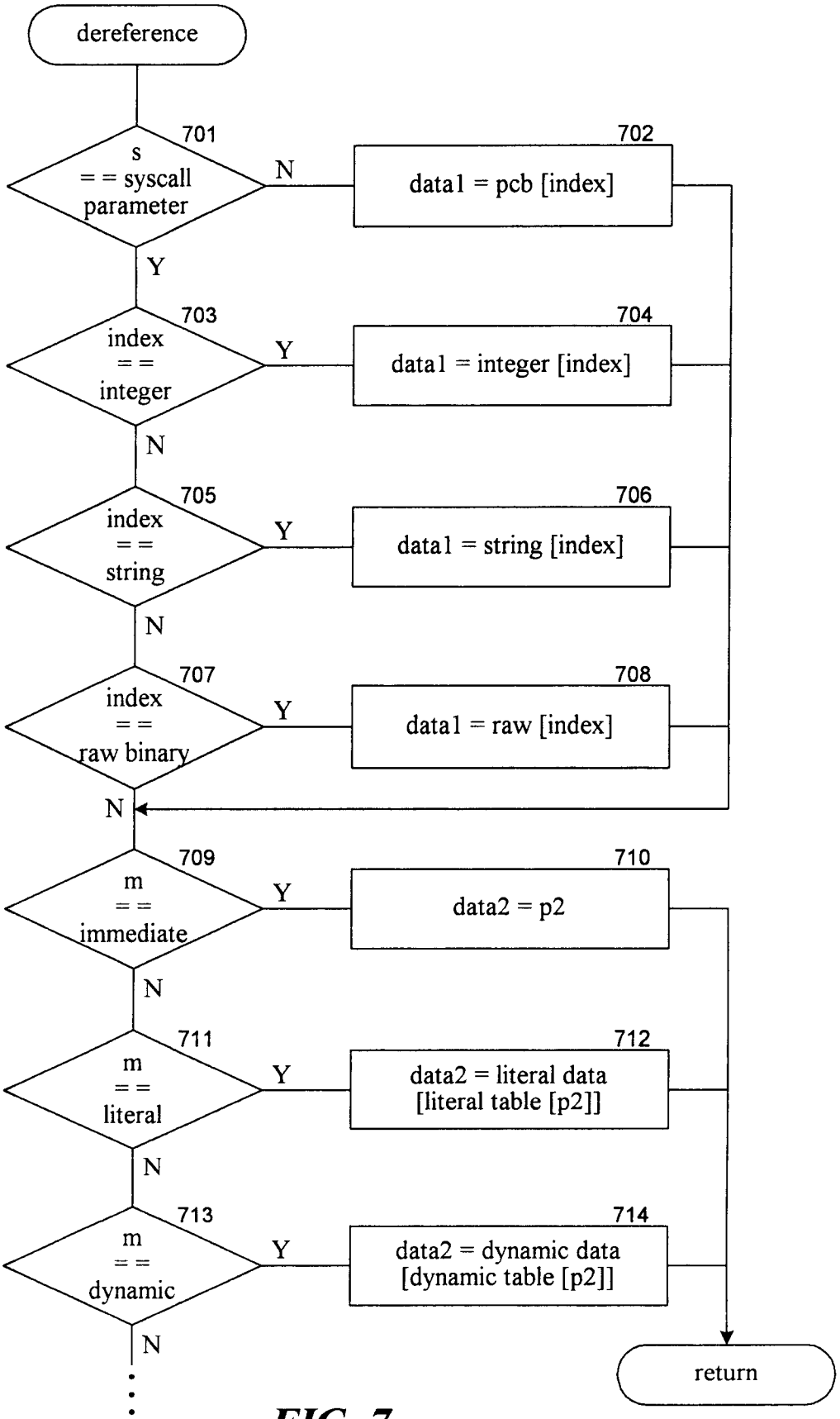

dereference
701
s
= = syscall
parameter
N
702
data1 = pcb [index]
Y
703
index
= =
integer
Y
704
data1 = integer [index]
N
705
index
= =
string
Y
706
data1 = string [index]
N
707
index
= =
raw binary
Y
708
data1 = raw [index]
N
709
m
= =
immediate
Y
710
data2 = p2
N
711
m
= =
literal
Y
712
data2 = literal data
[literal table [p2]]
N
713
m
= =
dynamic
Y
714
data2 = dynamic data
[dynamic table [p2]]
N
return

FIG. 7

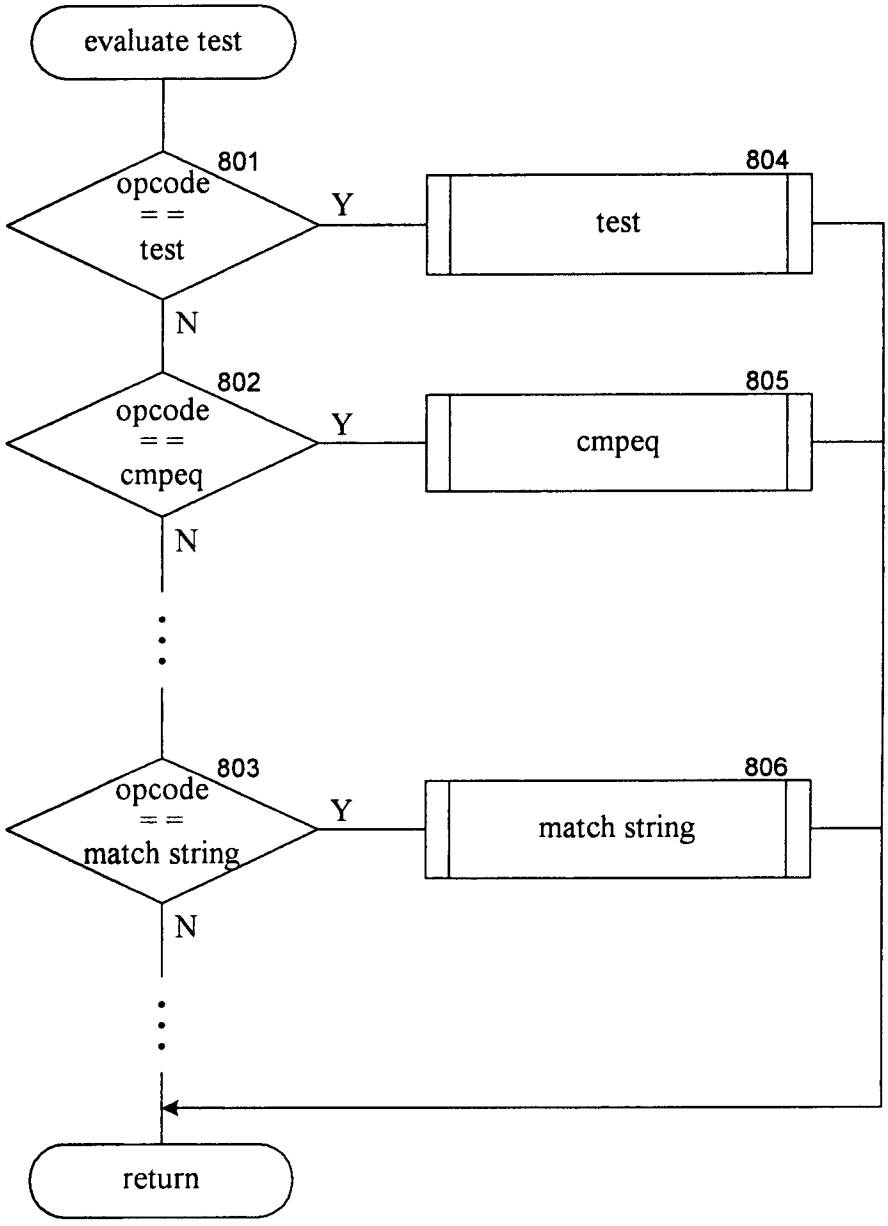

evaluate test
801
opcode
==
test
Y
804
test
N
802
opcode
==
cmpeq
Y
805
cmpeq
N
803
opcode
==
match string
Y
806
match string
N
return

FIG. 8

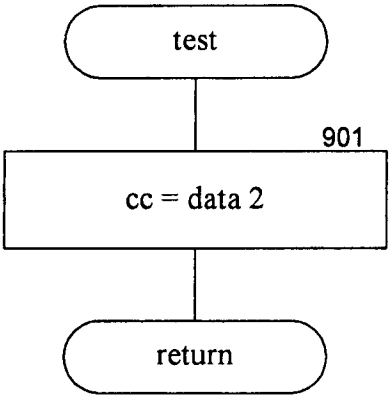


FIG. 9

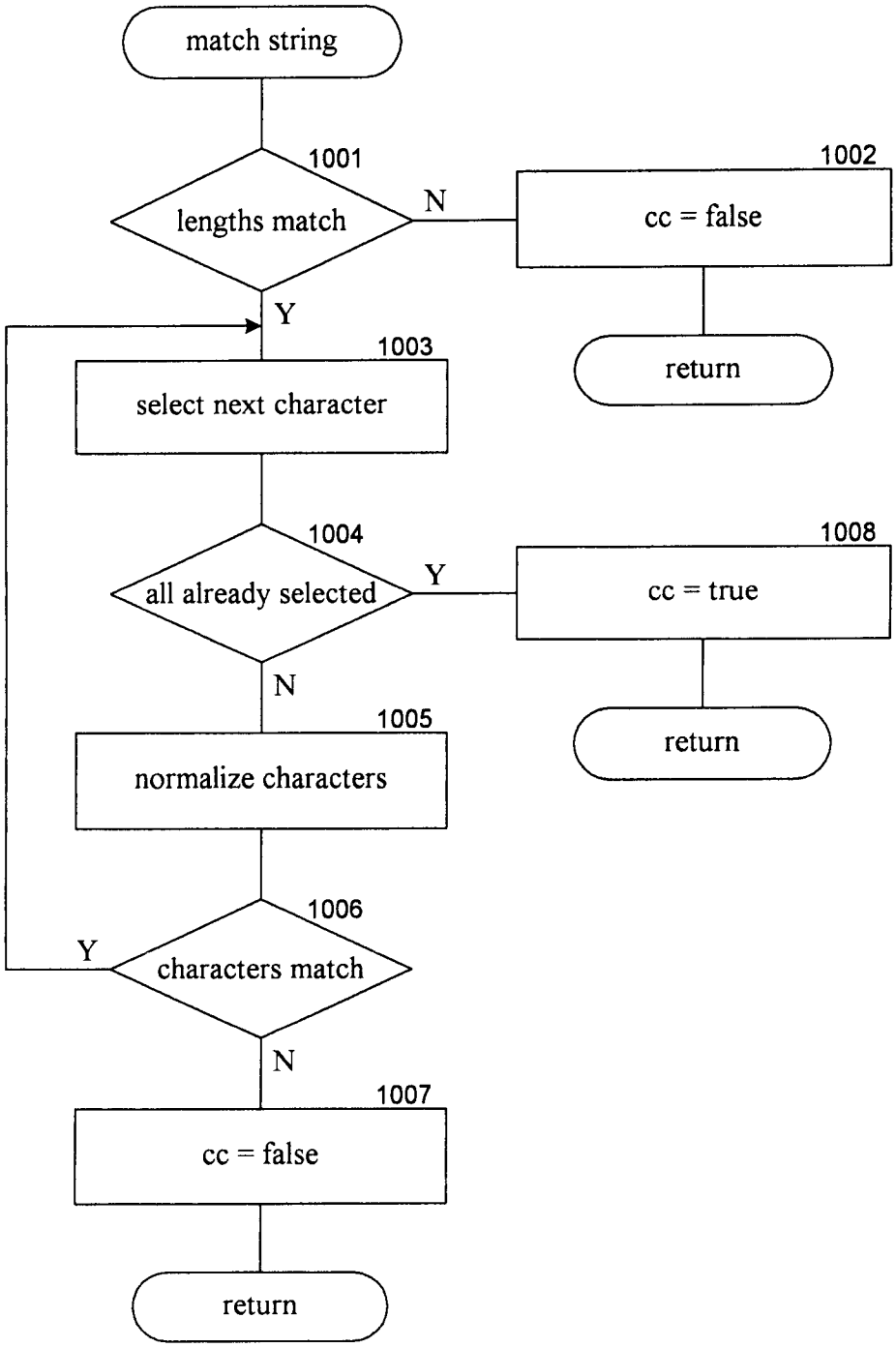

match string
1001
lengths match
N
1002
cc = false
return
Y
1003
select next character
1004
all already selected
Y
1008
cc = true
return
N
1005
normalize characters
1006
characters match
Y
N
1007
cc = false
return

FIG. 10

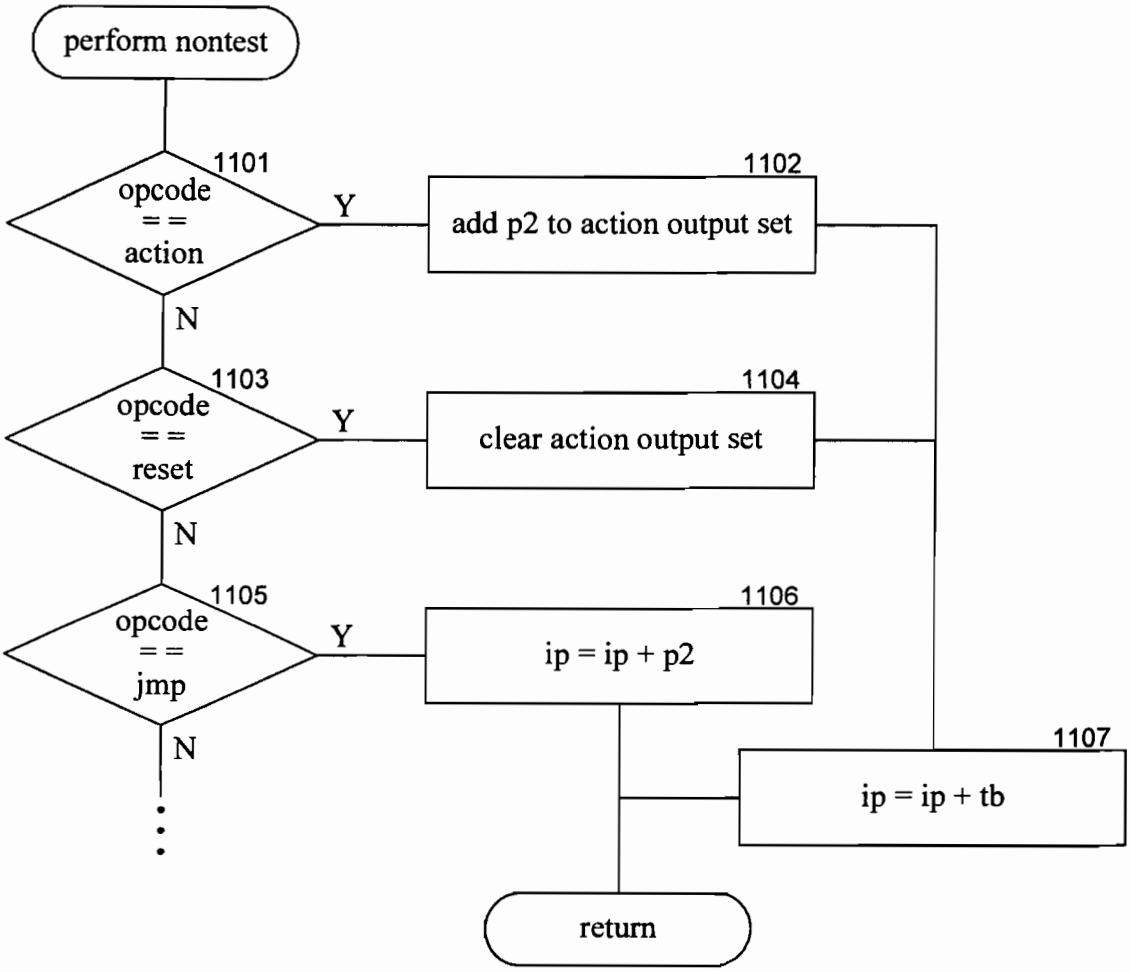

perform nontest
1101
opcode
==
action
Y
1102
add p2 to action output set
N
1103
opcode
==
reset
Y
1104
clear action output set
N
1105
opcode
==
jmp
Y
1106
ip = ip + p2
N
1107
ip = ip + tb
return

FIG. 11