

```

proctype process1{
    push(555);
    push(777);
    mfence();
    take(tvalue);
    push(999);
}

```

```

proctype process2 (chan ch) {
    steal
}

```

spin -a wsq.pml

gcc -DMEMLIM=1024 -O2 -DCOLLAPSE -DXUSAFE -DSAFETY -w -o pan pan.c

./pan -m10000 -n

Pid: 3121

Depth= 1203 States= 1e+06 Transitions= 2.34e+06 Memory= 168.183 t= 10 R= 1e+05

(Spin Version 6.2.4 -- 8 March 2013)

+ Partial Order Reduction

+ Compression

Full statespace search for:

never claim	- (none specified)
assertion violations	+
cycle checks	- (disabled by -DSAFETY)
invalid end states	+

State-vector 900 byte, depth reached 1234, errors: 0

1239589 states, stored

1675481 states, matched

2915070 transitions (= stored+matched)

13770401 atomic steps

hash conflicts: 27768 (resolved)

Stats on memory usage (in Megabytes):

1097.048 equivalent memory usage for states (stored\*(State-vector + overhead))

49.785 actual memory usage for states (compression: 4.54%)

state-vector as stored = 14 byte + 28 byte overhead

128.000 memory used for hash table (-w24)

0.534 memory used for DFS stack (-m10000)

178.241 total actual memory usage

nr of templates: [ 0:globals 1:chans 2:procs ]

collapse counts: [ 0:3788 2:3593 3:511 4:293 5:12 ]

pan: elapsed time 12.5 seconds

No errors found -- did you verify all claims?

```
proctype process1{
    push(555);
    take(tvalue);
    push(777);
}
```

```
spin -a wsq.pml
gcc -DMEMLIM=1024 -O2 -DCOLLAPSE -DXUSAFE -DSAFETY -w -o pan pan.c
./pan -m10000 -n
Pid: 3443
```

(Spin Version 6.2.4 -- 8 March 2013)  
+ Partial Order Reduction  
+ Compression

Full statespace search for:

never claim	- (none specified)
assertion violations	+
cycle checks	- (disabled by -DSAFETY)
invalid end states	+

State-vector 732 byte, depth reached 712, errors: 0  
971 states, stored  
381 states, matched  
1352 transitions (= stored+matched)  
7223 atomic steps  
hash conflicts: 0 (resolved)

Stats on memory usage (in Megabytes):

0.704	equivalent memory usage for states (stored*(State-vector + overhead))
0.434	actual memory usage for states (compression: 61.60%)
	state-vector as stored = 440 byte + 28 byte overhead
128.000	memory used for hash table (-w24)
0.534	memory used for DFS stack (-m10000)
128.925	total actual memory usage

nr of templates: [ 0:globals 1:chans 2:procs ]  
collapse counts: [ 0:42 2:595 3:94 4:2 5:12 ]

pan: elapsed time 0.01 seconds  
No errors found -- did you verify all claims?

```
BUFF_SIZE 14;      MEM_SIZE 40;      MAX_QUEUE_SIZE 10
```

```
proctype process1{  
    push(555);  
    mfence  
    take(tvalue);  
    mfence;  
    push(777);  
}
```

```
spin -a wsq.pml  
gcc -DMEMLIM=1024 -O2 -DXUSAFE -DSAFETY -DNOCLAIM -w -o pan pan.c  
./pan -m10000 -n  
Pid: 5526
```

(Spin Version 6.2.4 -- 8 March 2013)  
+ Partial Order Reduction

Full statespace search for:

never claim	- (not selected)
assertion violations	+
cycle checks	- (disabled by -DSAFETY)
invalid end states	+

State-vector 676 byte, depth reached 855, errors: 0  
1114 states, stored  
458 states, matched  
1572 transitions (= stored+matched)  
7616 atomic steps  
hash conflicts: 0 (resolved)

Stats on memory usage (in Megabytes):

0.748	equivalent memory usage for states (stored*(State-vector + overhead))
1.020	actual memory usage for states
128.000	memory used for hash table (-w24)
0.534	memory used for DFS stack (-m10000)
129.511	total actual memory usage

pan: elapsed time 0 seconds  
No errors found -- did you verify all claims?

***The buffer must have a minimum size of 14 to enable a push with an extend operation.***

```
proctype process1 {
    push(555);
    take(tvalue1);
    mfence();
    push(777);
}
```

```
proctype process2{
    steal(svalue);
}
```

```
spin -a wsq.pml
gcc -DMEMLIM=1024 -O2 -DXUSAFE -DSAFETY -DNOCLAIM -w -o pan pan.c
./pan -m10000 -n
Pid: 5778
```

(Spin Version 6.2.4 -- 8 March 2013)  
+ Partial Order Reduction

Full statespace search for:

never claim	- (not selected)
assertion violations	+
cycle checks	- (disabled by -DSAFETY)
invalid end states	+

State-vector 784 byte, depth reached 1020, errors: 0  
656069 states, stored  
873730 states, matched  
1529799 transitions (= stored+matched)  
6464115 atomic steps  
hash conflicts: 6553 (resolved)

Stats on memory usage (in Megabytes):

508.049	equivalent memory usage for states (stored*(State-vector + overhead))
481.216	actual memory usage for states (compression: 94.72%)
	state-vector as stored = 741 byte + 28 byte overhead
128.000	memory used for hash table (-w24)
0.534	memory used for DFS stack (-m10000)
2.701	memory lost to fragmentation
607.050	total actual memory usage

pan: elapsed time 5.01 seconds  
No errors found -- did you verify all claims?

```

proctype process1 {
    push(555);
    take(tvalue1);
    mfence();
    push(777);
    push(999);
}

proctype process2 {
    steal(svalue);
}

proctype process3 {
    steal(stealval);
}

```

```

spin -a wsq.pml
gcc -DMEMLIM=1024 -O2 -DCOLLAPSE -DXUSAFE -DSAFETY -DNOCLAIM -w -o pan
pan.c
./pan -m10000 -n
Pid: 5933
Depth= 1221 States= 1e+06 Transitions= 2.28e+06 Memory= 168.476      t= 10.5 R=
1e+05

```

(Spin Version 6.2.4 -- 8 March 2013)  
 + Partial Order Reduction  
 + Compression

Full statespace search for:

never claim	- (not selected)
assertion violations	+
cycle checks	- (disabled by -DSAFETY)
invalid end states	+

State-vector 840 byte, depth reached 1221, errors: 0  
 1168072 states, stored  
 1504580 states, matched  
 2672652 transitions (= stored+matched)  
 12041443 atomic steps  
 hash conflicts: 29515 (resolved)

Stats on memory usage (in Megabytes):

966.918	equivalent memory usage for states (stored*(State-vector + overhead))
47.126	actual memory usage for states (compression: 4.87%)
	state-vector as stored = 14 byte + 28 byte overhead
128.000	memory used for hash table (-w24)
0.534	memory used for DFS stack (-m10000)
175.605	total actual memory usage

nr of templates: [ 0:globals 1:chans 2:procs ]  
 collapse counts: [ 0:3974 2:4049 3:539 4:627 6:12 ]

pan: elapsed time 12.2 seconds  
 No errors found -- did you verify all claims?  
**Only checkable with further compile directives**

```

proctype process1 {
    push(555);
    take(tvalue2);
    mfence();
    push(999);
    take(tvalue1);
}

proctype process2 {
    steal(svalue);
}

proctype process3 {
    steal(stealval);
}

```

```

spin -a wsq.pml
gcc -DMEMLIM=1024 -O2 -DXUSAFE -DSAFETY -DNOCLAIM -w -o pan pan.c
./pan -m10000 -n
Pid: 6021
pan:1: assertion violated (asTop==asBottom) (at depth 880)
pan: wrote wsq.pml.trail

```

***push() and take() get into conflict. Push() operation has finished but the changed bottom value isn't written to the memory yet. So take() which is called by the same process works on wrong variables???***

***Why does take() does not take the value from the buffer?***