

4.2.4 [C] lock/unlock Construct

Synopsis

The lock/unlock constructs are equivalent to the lock/unlock statements in Fortran 2008.

Syntax

```
[C] xmp_lock_t lock-object [, lock-object ]...
[C] #pragma xmp lock (lock-object) [ acquired_lock (success) ] [ stat (status) ]
[C] #pragma xmp unlock (lock-object) [ stat (status) ]
```

Note that the variable *lock-object* is a coarray, the variable *success* is an expression of type `bool`, and the variable *status* is an expression of type `int`.

Description

The `lock` construct, in combination with the `unlock` construct, is used to control a *lock-object*. The *lock-object* must be defined as a coarray to control it on a target node. The *lock-object* must be an expression of type `xmp_lock_t` which is an opaque object defined in “xmp.h”.

If the `acquired_lock` clause is not used in the `lock` construct and the *lock-object* is locked, a node stops at the `lock` construct until the *lock-object* is unlocked by a different node. If the `acquired_lock` clause is used in the `lock` construct and the *lock-object* is locked by a different node, a node does not stop at the `lock` construct and the variable *success* is defined with the value `false` and `lock` construct leaves the *lock-object* unchanged. If the `acquired_lock` clause is used in the `lock` construct and the *lock-object* is unlocked, the variable *success* is defined with the value `true`.

The *status* is defined with one of the follow symbolic constants when executing `lock/unlock` construct.

- `XMP_STAT_SUCCESS`
- `XMP_STAT_LOCKED`
- `XMP_STAT_UNLOCKED`
- `XMP_STAT_LOCKED_OTHER_IMAGE`

If an execution of `lock/unlock` construct is success, the *status* is defined with `XMP_STAT_SUCCESS`. A condition where the *status* is defined with `XMP_STAT_LOCKED`, `XMP_STAT_UNLOCKED`, or `XMP_STAT_LOCKED_OTHER_IMAGE` is the same as that where the *status* is defined with `STAT_LOCKED`, `STAT_UNLOCKED`, or `STAT_LOCKED_OTHER_IMAGE` in Fortran 2008. These symbolic constants are defined in “xmp.h”. If any other error condition occurs during execution of these constructs, the *status* is defined with a value which is different from the value of `XMP_STAT_SUCCESS`, `XMP_STAT_LOCKED`, `XMP_STAT_UNLOCKED`, and `XMP_STAT_LOCKED_OTHER_IMAGE`.

Example

```

XcalableMP C
#include "xmp.h"

xmp_lock_t lock_obj:[*];
int A:[*], B;
#pragma xmp nodes p(2)
...
#pragma xmp lock(lock_obj:[2])

```

```
10  if(xmp_node_num() == 1){
    A:[2] = B;
}
#pragma xmp unlock(lock_obj:[2])
```

001
002
003
004
005
006
007
008
009
010
011
012
013
014
015
016
017
018
019
020
021
022
023
024
025
026
027
028
029
030
031
032
033
034
035
036
037
038
039
040
041
042
043
044
045
046
047
048
049
050
051
052
053
054
055
056
057