

The C11 addition to Litmus

Jacques-Pascal Deplaix - London

Introduction: What is Litmus ?

Litmus is compiler which takes a « litmus test » and produces an executable that tests memory models

A litmus test looks like this:

X86 MP

" "

{ }

```
P0          | P1          ;  
MOV [x],$1 | MOV EAX,[x] ;  
MOV [y],$1 | MOV ECX,[y] ;
```

exists (1:EAX=1 /\ 1:ECX=0)

The out: An example

```
static void *P0(void *_vb) {
    mbar();

    for (int _i = _size_of_test-1 ; _i >= 0 ; _i--) {
        barrier_wait(_th_id,_i,&barrier[_i]);

        asm __volatile__ (
            "\n"
            "#START litmus P0\n"
            "#_litmus_P0_0\n\t"

            "movl $1,%[x]\n"
            "#_litmus_P0_1\n\t"

            "movl $1,%[y]\n"
            "#END litmus\n\t"
            :[x] "=m" (_a->x[_i]),[y] "=m" (_a->y[_i])
            :
            : "cc", "memory"
        );
    }
    mbar();
    return NULL;
}
```

How to run litmus

If you have an adventurous soul, these steps can compile the exemple given before:

```
$ mkdir /tmp/test  
$ litmus MP.litmus -o /tmp/test  
$ cd /tmp/test  
$ make
```

...

```
$ ./MP.exe
```

Test MP Allowed

Histogram (4 states)

```
500020:>1:EAX=0; 1:ECX=0;  
24      *>1:EAX=1; 1:ECX=0;  
5       :>1:EAX=0; 1:ECX=1;  
499951:>1:EAX=1; 1:ECX=1;
```

Ok

Condition exists (1:EAX=1 /\ 1:ECX=0) is validated

Observation MP Sometimes 24 999976

For more informations, see: <http://diy.inria.fr/doc/litmus.html>

Handling C: Motivations

The motivations for the C frontend is the following:

- Handle multiple architectures with the same test
- Can be used to test the C compiler itself
- Be able to test the C model

Future

Herd with C

Questions ?

« C'est pas faux ! »