

Re: [PATCH 10/10] atomic.h : Add atomic64 cmpxchg, xchg and add_unless to x86_64

Re: [PATCH 10/10] atomic.h : Add atomic64 cmpxchg, xchg and add_unless to x86_64

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2007-02/msg04068.html>

- *From:* Mathieu Desnoyers <mathieu.desnoyers@xxxxxxxxxx>
 - *Date:* Mon, 12 Feb 2007 11:43:15 -0500
-

* Andi Kleen (ak@xxxxxxxx) wrote:

Mathieu Desnoyers <mathieu.desnoyers@xxxxxxxxxx> writes:

```
/**
 *@@ -402,7 +405,7 @@ static __inline__ long atomic64_sub_return(long i,
 atomic64_t *v)
 */
#define atomic_add_unless(v, a, u) \
({ \
 - int c, old; \
 + __typeof__((v)->counter) c, old; \
```

This change seems useless. Please drop.

In general you should enforce atomic_t vs atomic64_t types on x86, otherwise people will write code that won't compile on other architectures that have truly different implementations.

I would suggest using inlines.

-Andi

Hi Andi,

Yes, I agree this typeof is not needed and more error prone. All the original atomic.h code that uses cmpxchg (which includes the atomic_add_unless) uses defines instead of inline functions, probably to circumvent a circular dependency between system.h and atomic.h on powerpc (which my patch addresses). Therefore, it makes sense to use inline functions that will provide type checking.

Digging into the FRV architecture shows me that it is also affected by such a circular dependency. Here is the diff applying this against the rest of my atomic.h patches.

Re: [PATCH 10/10] atomic.h : Add atomic64 cmpxchg, xchg and add_unless to x86_64

Re: [PATCH 10/10] atomic.h : Add atomic64 cmpxchg, xchg and add_unless to x86_64

Signed-off-by: Mathieu Desnoyers <mathieu.desnoyers@xxxxxxxxxxx>

```
diff --git a/include/asm-alpha/atomic.h b/include/asm-alpha/atomic.h
index 7b4fba8..f5cb7b8 100644
--- a/include/asm-alpha/atomic.h
+++ b/include/asm-alpha/atomic.h
@@ -2,6 +2,7 @@
#define _ALPHA_ATOMIC_H

#include <asm/barrier.h>
+#include <asm/system.h>

/*
 * Atomic operations that C can't guarantee us. Useful for
@@ -190,20 +191,21 @@ static __inline__ long atomic64_sub_return(long i, atomic64_t * v)
 * Atomically adds @a to @v, so long as it was not @u.
 * Returns non-zero if @v was not @u, and zero otherwise.
 */
-#define atomic_add_unless(v, a, u) \
-({ \
- __typeof__((v)->counter) c, old; \
- c = atomic_read(v); \
- for (;;) { \
- if (unlikely(c == (u))) \
- break; \
-
```

Re: [PATCH 10/10] atomic.h : Add atomic64 cmpxchg, xchg and add_unless to x86_64