

## Updated Lockup Patches, 2.4.22 – 23 Nforce2, apic timer ack delay, ioapic edge for NMI debug

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2003-12/4673.html>

---

*From:* Ross Dickson ([ross\\_at\\_datcreative.com.au](mailto:ross_at_datcreative.com.au))

*Date:* 12/22/03

To: [linux-kernel@vger.kernel.org](mailto:linux-kernel@vger.kernel.org)

Date: Mon, 22 Dec 2003 14:37:11 +1000

Here are my patches reworked for 2.4.22.  
They should patch on 2.4.23 but differ in line numbers.

I use them on my patched 2.4.23 kern.

Details are in following lkml thread, please refer to it prior to usage.

Updated Lockup Patches, 2.6.0 Nforce2, apic timer ack delay, ioapic edge for NMI debug

If not subscribed it can be found in many archives  
such as

<http://linux.derkeiler.com/Mailing-Lists/Kernel/2003-12/4525.html>

or here

<http://lkml.org/lkml/2003/12/21/7>

Regards,

Ross Dickson

local apic timer ack delay:

---- CUT HERE ----

```
--- linux-2.4.22/arch/i386/kernel/apic.c 2003-06-14 00:51:29.000000000 +1000
+++ linux-2.4.22-rd/arch/i386/kernel/apic.c 2003-12-22 13:18:08.000000000 +1000
@@ -1058,10 +1058,17 @@ inline void smp_local_timer_interrupt(st
     * we can take more than 100K local irqs per second on a 100 MHz P5.
     */
 }

/*
+ * Athlon nforce2 R.D.
+ * preset timer ack mode if desired
+ * e.g. static int apic_timerack = 2;
+ */
+static int apic_timerack;
+
+/*
```

## Linux–Kernel: Updated Lockup Patches, 2.4.22 – 23 Nforce2, apic timer ack delay, ioapic edge for NMI debug

```
* Local APIC timer interrupt. This is the most natural way for doing
* local interrupts, but local timer interrupts can be emulated by
* broadcast interrupts too. [in case the hw doesn't support APIC timers]
*
* [ if a single–CPU system runs an SMP kernel then we call the local
@@ –1077,10 +1084,54 @@ void smp_apic_timer_interrupt(struct pt_
    * the NMI deadlock–detector uses this.
    */
    apic_timer_irqs[cpu]++;

/*
+ * Athlon nforce2 timer ack delay. Ross Dickson.
+ * works around issue of hard lockups in code location
+ * where linux exposes underlying system timing fault?
+ * hopefully manufacturers will fix it soon.
+ * We leave C1 disconnect bit alone as bios/SMM wants?
+ */
+ if(apic_timerack) {
+ if(apic_timerack==1) {
+ /* v1 timer ack delay, inline delay version
+
```