

Dont use num_processors as index to generate logical cpu# in x86_64

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2006-01/msg11452.html>

- *From:* Ashok Raj <ashok.raj@xxxxxxxx>
 - *Date:* Tue, 31 Jan 2006 07:59:23 -0800
-

Hi Andi

Please consider the following patch. I am in process of developing ACPI based cpu hotplug support. Using num_processors as index is not friendly to hotplug, so I turned it to use cpu_present_map instead. The code is also a little bit more cleaner and easy to read.

Cheers,
ashok

--

Cheers,
Ashok Raj
- Open Source Technology Center

Minor cleanup to lend better for physical CPU hotplug.
Earlier way of using num_processors as index doesn't fit if CPUs come and go. This makes the code a little bit better to read, and helps physical hotplug use the same functions as boot.

Reserving CPU0 for BSP is too late to be done in smp_prepare_boot_cpu().
Since logical assignments from MADT is already done via setup_arch()->acpi_boot_init()->parse_lapic

Signed-off-by: Ashok Raj <ashok.raj@xxxxxxxx>

arch/x86_64/kernel/mpparse.c | 19 ++++++++
arch/x86_64/kernel/setup.c | 6 +++++
2 files changed, 14 insertions(+), 11 deletions(-)

Index: linux-2.6.16-rc1-mm2/arch/x86_64/kernel/mpparse.c

=====
--- linux-2.6.16-rc1-mm2.orig/arch/x86_64/kernel/mpparse.c
+++ linux-2.6.16-rc1-mm2/arch/x86_64/kernel/mpparse.c
@@ -106,11 +106,11 @@ static int __init mpf_checksum(unsigned
return sum & 0xFF;
}

Dont use num_processors as index to generate logical cpu# in x86_64

```
-static void __init MP_processor_info (struct mpc_config_processor *m)
+static void __cpuinit MP_processor_info (struct mpc_config_processor *m)
{
int cpu;
unsigned char ver;
- static int found_bsp=0;
+ cpumask_t tmp_map;

if (!(m->mpc_cpuflag & CPU_ENABLED)) {
disabled_cpus++;
@@ -133,8 +133,10 @@ static void __init MP_processor_info (st
return;
}

- cpu = num_processors++;
-
+ num_processors++;
+ cpus_complement(tmp_map, cpu_present_map);
+ cpu = first_cpu(tmp_map);
+
#if MAX_APICS < 255
if ((int)m->mpc_apicid > MAX_APICS) {
printk(KERN_ERR "Processor #%d INVALID. (Max ID: %d).\n",
@@ -160,12 +162,7 @@ static void __init MP_processor_info (st
* entry is BSP, and so on.
*/
cpu = 0;
-
- bios_cpu_apicid[0] = m->mpc_apicid;
- x86_cpu_to_apicid[0] = m->mpc_apicid;
- found_bsp = 1;
- } else
- cpu = num_processors - found_bsp;
+ }
bios_cpu_apicid[cpu] = m->mpc_apicid;
x86_cpu_to_apicid[cpu] = m->mpc_apicid;

@@ -691,7 +688,7 @@ void __init mp_register_lapic_address (
}

-void __init mp_register_lapic (
+void __cpuinit mp_register_lapic (
u8 id,
u8 enabled)
{
Index: linux-2.6.16-rc1-mm2/arch/x86_64/kernel/setup.c
=====
--- linux-2.6.16-rc1-mm2.orig/arch/x86_64/kernel/setup.c
+++ linux-2.6.16-rc1-mm2/arch/x86_64/kernel/setup.c
```

Dont use num_processors as index to generate logical cpu# in x86_64

Dont use num_processors as index to generate logical cpu# in x86_64

```
@@ -704,6 +704,12 @@ void __init setup_arch(char **cmdline_p)
```

```
check_ioapic();
```

```
+ /*
```

```
+ * set this early, so we dont allocate cpu0
```

```
+ * if MADT list doesnt list BSP first
```

```
+ * mpparse.c/MP_processor_info() allocates logical cpu numbers.
```

```
+ */
```

```
+ cpu_set(0, cpu_present_map);
```

```
#ifdef CONFIG_ACPI
```

```
/*
```

```
* Read APIC and some other early information from ACPI tables.
```

```
—
```

To unsubscribe from this list: send the line "unsubscribe linux-kernel" in the body of a message to majordomo@xxxxxxxxxxxxxxxxxxx

More majordomo info at <http://vger.kernel.org/majordomo-info.html>

Please read the FAQ at <http://www.tux.org/lkml/>

-
- Prev by Date: ***Re: 2.6.16-rc1-mm4***
 - Next by Date: ***Re: [RFC] VM: I have a dream...***
 - Previous by thread: ***Re: Fw: [PATCH 2.6.16-rc1-git4] accessfs: a permission managing filesystem***
 - Next by thread: ***[2.6 patch] wrong firmware location in IPW2100 Kconfig entry***
 - Index(es):
 - ◆ ***Date***
 - ◆ ***Thread***