

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

Source: <http://linux.derkeiler.com/Mailing-Lists/Kernel/2008-10/msg09005.html>

- *From:* Arjan van de Ven <arjan@xxxxxxxxxxxxxxx>
 - *Date:* Wed, 22 Oct 2008 15:30:54 -0700
-

From b62f70d43d3e9a9a70775b40cb4ecba058f07a01 Mon Sep 17 00:00:00 2001

From: Arjan van de Ven <arjan@xxxxxxxxxxxxxxx>
Date: Wed, 22 Oct 2008 15:29:00 -0700
Subject: [PATCH] get rid if __cpuinit and __cpuexit

as discussed briefly at the kernel summit; __cpuinit and __cpuexit are causing a ton more complexity than they're worth (just see how many section warnings an average build has); and the *vast* majority of people using linux have CPU hotplug enabled anyway (since it's required for suspend/resume). The code size saved is also very minimal, since although many files are touched, very few are actually present in a kernel image at a time.

For now, __cpuinit and __cpuexit are left behind as empty skeletons to not break in-flight stuff using it, it should be removed in a release or two entirely.

Signed-off-by Arjan van de Ven <arjan@xxxxxxxxxxxxxxx>

Documentation/cpu-hotplug.txt | 4 +-
arch/alpha/kernel/smp.c | 4 +-
arch/arm/common/gic.c | 2 +-
arch/arm/kernel/smp.c | 12 ++--
arch/arm/mach-realview/localtimer.c | 8 +++-
arch/arm/mach-realview/platsmp.c | 6 +-
arch/cris/arch-v32/kernel/smp.c | 2 +-
arch/frv/kernel/setup.c | 2 +-
arch/ia64/kernel/err_inject.c | 8 ++--
arch/ia64/kernel/mca.c | 12 +++-
arch/ia64/kernel/numa.c | 4 +-
arch/ia64/kernel/palinfo.c | 4 +-
arch/ia64/kernel/salinfo.c | 4 +-
arch/ia64/kernel/setup.c | 10 ++--
arch/ia64/kernel/smpboot.c | 12 ++--
arch/ia64/kernel/topology.c | 18 ++++----
arch/ia64/mm/contig.c | 2 +-
arch/ia64/mm/discontig.c | 2 +-
arch/ia64/sn/kernel/setup.c | 4 +-

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

arch/m32r/kernel/smpboot.c | 2 +-
arch/mips/kernel/cevt-bcm1480.c | 2 +-
arch/mips/kernel/cevt-r4k.c | 2 +-
arch/mips/kernel/cevt-sb1250.c | 2 +-
arch/mips/kernel/cevt-smtc.c | 2 +-
arch/mips/kernel/cpu-bugs64.c | 4 +-
arch/mips/kernel/cpu-probe.c | 10 +++--
arch/mips/kernel/smp-mt.c | 6 +-
arch/mips/kernel/smp-up.c | 6 +-
arch/mips/kernel/smp.c | 6 +-
arch/mips/kernel/smtc.c | 2 +-
arch/mips/kernel/spram.c | 14 +++----
arch/mips/kernel/time.c | 2 +-
arch/mips/kernel/traps.c | 8 +++--
arch/mips/kernel/watch.c | 2 +-
arch/mips/lib/uncached.c | 2 +-
arch/mips/mipssim/sim_smtc.c | 6 +-
arch/mips/mipssim/sim_time.c | 2 +-
arch/mips/mm/c-r3k.c | 8 +++--
arch/mips/mm/c-r4k.c | 34 ++++++-----
arch/mips/mm/c-tx39.c | 2 +-
arch/mips/mm/page.c | 40 ++++++-----
arch/mips/mm/sc-ip22.c | 2 +-
arch/mips/mm/sc-mips.c | 2 +-
arch/mips/mm/sc-r5k.c | 2 +-
arch/mips/mm/sc-rm7k.c | 6 +-
arch/mips/mm/tlb-r3k.c | 2 +-
arch/mips/mm/tlb-r4k.c | 8 +++--
arch/mips/mm/tlb-r8k.c | 4 +-
arch/mips/mm/tlbex.c | 70 ++++++-----
arch/mips/mm/uasm.c | 70 ++++++-----
arch/mips/mm/uasm.h | 24 +++++-----
arch/mips/mti-malta/malta-smtc.c | 6 +-
arch/mips/mti-malta/malta-time.c | 2 +-
arch/mips/pci/pci-ip27.c | 2 +-
arch/mips/pmc-sierra/yosemite/smp.c | 12 +++--
arch/mips/sgi-ip27/ip27-init.c | 4 +-
arch/mips/sgi-ip27/ip27-smp.c | 6 +-
arch/mips/sgi-ip27/ip27-timer.c | 6 +-
arch/mips/sgi-ip27/ip27-xtalk.c | 6 +-
arch/mips/sibyte/bcm1480/smp.c | 8 +++--
arch/mips/sibyte/sb1250/smp.c | 8 +++--
arch/parisc/kernel/hardware.c | 2 +-
arch/parisc/kernel/processor.c | 2 +-
arch/parisc/kernel/smp.c | 4 +-
arch/powerpc/kernel/smp.c | 2 +-
arch/powerpc/kernel/sysfs.c | 12 +++--
arch/powerpc/mm/numa.c | 12 +++--
arch/s390/appldata/appldata_base.c | 8 +++--
arch/s390/kernel/entry.h | 2 +-
arch/s390/kernel/setup.c | 4 +-

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

arch/s390/kernel/smp.c | 10 ++--
arch/sh/kernel/smp.c | 4 +-
arch/sh/kernel/traps_32.c | 2 +-
arch/sparc/kernel/smp.c | 8 ++--
arch/sparc/kernel/sun4d_smp.c | 2 +-
arch/sparc/kernel/sun4m_smp.c | 4 +-
arch/sparc64/kernel/ds.c | 4 +-
arch/sparc64/kernel/entry.h | 2 +-
arch/sparc64/kernel/irq.c | 4 +-
arch/sparc64/kernel/mdesc.c | 2 +-
arch/sparc64/kernel/smp.c | 4 +-
arch/sparc64/kernel/sysfs.c | 4 +-
arch/sparc64/mm/init.c | 2 +-
arch/x86/kernel/acpi/boot.c | 4 +-
arch/x86/kernel/apic_32.c | 12 ++--
arch/x86/kernel/apic_64.c | 18 ++++-----
arch/x86/kernel/cpu/addon_cpuid_features.c | 6 +-
arch/x86/kernel/cpu/amd.c | 22 ++++-----
arch/x86/kernel/cpu/centaur.c | 26 ++++-----
arch/x86/kernel/cpu/centaur_64.c | 6 +-
arch/x86/kernel/cpu/common.c | 58 ++++++-----
arch/x86/kernel/cpu/cpufreq/powernow-k8.c | 4 +-
arch/x86/kernel/cpu/cyrix.c | 40 ++++++-----
arch/x86/kernel/cpu/intel.c | 22 ++++-----
arch/x86/kernel/cpu/intel_cacheinfo.c | 44 ++++++-----
arch/x86/kernel/cpu/mcheck/mce_64.c | 16 +++--
arch/x86/kernel/cpu/mcheck/mce_amd_64.c | 10 ++--
arch/x86/kernel/cpu/mcheck/mce_intel_64.c | 4 +-
arch/x86/kernel/cpu/mcheck/therm_throt.c | 8 ++--
arch/x86/kernel/cpu/transmeta.c | 6 +-
arch/x86/kernel/cpu/umc.c | 2 +-
arch/x86/kernel/cpuid.c | 4 +-
arch/x86/kernel/ds.c | 2 +-
arch/x86/kernel/genx2apic_uv_x.c | 4 +-
arch/x86/kernel/i387.c | 6 +-
arch/x86/kernel/irq_32.c | 2 +-
arch/x86/kernel/microcode_core.c | 2 +-
arch/x86/kernel/mmconf-fam10h_64.c | 14 +++--
arch/x86/kernel/msr.c | 4 +-
arch/x86/kernel/numaq_32.c | 4 +-
arch/x86/kernel/process.c | 8 ++--
arch/x86/kernel/ptrace.c | 2 +-
arch/x86/kernel/setup.c | 2 +-
arch/x86/kernel/setup_percpu.c | 16 +++--
arch/x86/kernel/smpboot.c | 28 ++++-----
arch/x86/kernel/smpcommon.c | 2 +-
arch/x86/kernel/tlb_64.c | 2 +-
arch/x86/kernel/tsc.c | 2 +-
arch/x86/kernel/tsc_sync.c | 18 ++++-----
arch/x86/kernel/vsyscall_64.c | 6 +-
arch/x86/kernel/xsave.c | 2 +-

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

arch/x86/mach-voyager/voyager_smp.c | 4 +-
arch/x86/mm/init_64.c | 4 +-
arch/x86/mm/numa_64.c | 2 +-
arch/x86/mm/pat.c | 2 +-
arch/x86/pci/amd_bus.c | 4 +-
arch/x86/xen/setup.c | 6 +-
arch/x86/xen/smp.c | 8 ++--
arch/x86/xen/spinlock.c | 2 +-
arch/x86/xen/xen-ops.h | 2 +-
arch/xtensa/kernel/time.c | 2 +-
block/blk-softirq.c | 4 +-
drivers/acpi/ec.c | 2 +-
drivers/acpi/numa.c | 2 +-
drivers/acpi/processor_core.c | 4 +-
drivers/acpi/processor_idle.c | 4 +-
drivers/base/cpu.c | 4 +-
drivers/base/topology.c | 8 ++--
drivers/cpufreq/cpufreq.c | 2 +-
drivers/cpufreq/cpufreq_stats.c | 2 +-
drivers/hwmon/coretemp.c | 4 +-
drivers/infiniband/hw/ehca/ehca_irq.c | 6 +-
drivers/s390/sysinfo.c | 2 +-
include/asm-x86/ds.h | 2 +-
include/asm-x86/mmconfig.h | 4 +-
include/asm-x86/mpspec.h | 2 +-
include/asm-x86/numa_64.h | 8 ++--
include/asm-x86/ptrace.h | 2 +-
include/asm-x86/smp.h | 4 +-
include/linux/cpu.h | 2 +-
include/linux/init.h | 9 ++--
init/calibrate.c | 6 +-
kernel/cpu.c | 6 +-
kernel/fork.c | 4 +-
kernel/hrtimer.c | 6 +-
kernel/rcuclassic.c | 6 +-
kernel/rcupreempt.c | 6 +-
kernel/relay.c | 2 +-
kernel/sched.c | 8 ++--
kernel/smp.c | 2 +-
kernel/softirq.c | 8 ++--
kernel/softlockup.c | 4 +-
kernel/timer.c | 10 ++--
lib/percpu_counter.c | 2 +-
mm/page-writeback.c | 4 +-
mm/page_alloc.c | 6 +-
mm/slab.c | 10 ++--
mm/slub.c | 4 +-
mm/vmstat.c | 6 +-
net/iucv/iucv.c | 2 +-
170 files changed, 642 insertions(+), 645 deletions(-)

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
diff --git a/Documentation/cpu-hotplug.txt b/Documentation/cpu-hotplug.txt
index 94bbc27..50f7609 100644
```

```
--- a/Documentation/cpu-hotplug.txt
```

```
+++ b/Documentation/cpu-hotplug.txt
```

```
@@ -236,7 +236,7 @@ Q: If i have some kernel code that needs to be aware of CPU arrival and
A: This is what you would need in your kernel code to receive notifications.
```

```
#include <linux/cpu.h>
```

```
- static int __cpuinit foobar_cpu_callback(struct notifier_block *nfb,
```

```
+ static int foobar_cpu_callback(struct notifier_block *nfb,
```

```
unsigned long action, void *hcpu)
```

```
{
```

```
unsigned int cpu = (unsigned long)hcpu;
```

```
@@ -254,7 +254,7 @@ A: This is what you would need in your kernel code to receive notifications.
```

```
return NOTIFY_OK;
```

```
}
```

```
- static struct notifier_block __cpuinitdata foobar_cpu_notifier =
```

```
+ static struct notifier_block foobar_cpu_notifier =
```

```
{
```

```
.notifier_call = foobar_cpu_callback,
```

```
};
```

```
diff --git a/arch/alpha/kernel/smp.c b/arch/alpha/kernel/smp.c
```

```
index e657c45..b9eef22 100644
```

```
--- a/arch/alpha/kernel/smp.c
```

```
+++ b/arch/alpha/kernel/smp.c
```

```
@@ -360,7 +360,7 @@ secondary_cpu_start(int cpuid, struct task_struct *idle)
```

```
/*
```

```
* Bring one cpu online.
```

```
*/
```

```
-static int __cpuinit
```

```
+static int
```

```
smp_boot_one_cpu(int cpuid)
```

```
{
```

```
struct task_struct *idle;
```

```
@@ -488,7 +488,7 @@ smp_prepare_boot_cpu(void)
```

```
{
```

```
}
```

```
-int __cpuinit
```

```
+int
```

```
__cpu_up(unsigned int cpu)
```

```
{
```

```
smp_boot_one_cpu(cpu);
```

```
diff --git a/arch/arm/common/gic.c b/arch/arm/common/gic.c
```

```
index 7fc9860..fb37983 100644
```

```
--- a/arch/arm/common/gic.c
```

```
+++ b/arch/arm/common/gic.c
```

```
@@ -241,7 +241,7 @@ void __init gic_dist_init(unsigned int gic_nr, void __iomem *base,
```

```
writel(1, base + GIC_DIST_CTRL);
```

```
}
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
-void __cpuinit gic_cpu_init(unsigned int gic_nr, void __iomem *base)
+void gic_cpu_init(unsigned int gic_nr, void __iomem *base)
{
if (gic_nr >= MAX_GIC_NR)
BUG();
diff --git a/arch/arm/kernel/smp.c b/arch/arm/kernel/smp.c
index e42a749..3dd3029 100644
--- a/arch/arm/kernel/smp.c
+++ b/arch/arm/kernel/smp.c
@@ -72,7 +72,7 @@ enum ipi_msg_type {
IPI_CPU_STOP,
};

-int __cpuinit __cpu_up(unsigned int cpu)
+int __cpu_up(unsigned int cpu)
{
struct cpuinfo_arm *ci = &per_cpu(cpu_data, cpu);
struct task_struct *idle = ci->idle;
@@ -157,7 +157,7 @@ int __cpuinit __cpu_up(unsigned int cpu)
/*
* __cpu_disable runs on the processor to be shutdown.
*/
-int __cpuexit __cpu_disable(void)
+int __cpu_disable(void)
{
unsigned int cpu = smp_processor_id();
struct task_struct *p;
@@ -204,7 +204,7 @@ int __cpuexit __cpu_disable(void)
* called on the thread which is asking for a CPU to be shutdown -
* waits until shutdown has completed, or it is timed out.
*/
-void __cpuexit __cpu_die(unsigned int cpu)
+void __cpu_die(unsigned int cpu)
{
if (!platform_cpu_kill(cpu))
printk("CPU%u: unable to kill\n", cpu);
@@ -218,7 +218,7 @@ void __cpuexit __cpu_die(unsigned int cpu)
* of the other hotplug-cpu capable cores, so presumably coming
* out of idle fixes this.
*/
-void __cpuexit cpu_die(void)
+void cpu_die(void)
{
unsigned int cpu = smp_processor_id();

@@ -247,7 +247,7 @@ void __cpuexit cpu_die(void)
* This is the secondary CPU boot entry. We're using this CPUs
* idle thread stack, but a set of temporary page tables.
*/
-asmlinkage void __cpuinit secondary_start_kernel(void)
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
+asmlinkage void secondary_start_kernel(void)
{
struct mm_struct *mm = &init_mm;
unsigned int cpu = smp_processor_id();
@@ -305,7 +305,7 @@ asmlinkage void __cpuinit secondary_start_kernel(void)
* Called by both boot and secondaries to move global data into
* per-processor storage.
*/
-void __cpuinit smp_store_cpu_info(unsigned int cpuid)
+void smp_store_cpu_info(unsigned int cpuid)
{
struct cpuinfo_arm *cpu_info = &per_cpu(cpu_data, cpuid);

diff --git a/arch/arm/mach-realview/localtimer.c b/arch/arm/mach-realview/localtimer.c
index 44d178c..2133507 100644
--- a/arch/arm/mach-realview/localtimer.c
+++ b/arch/arm/mach-realview/localtimer.c
@@ -101,7 +101,7 @@ int local_timer_ack(void)
return 0;
}

-static void __cpuinit twd_calibrate_rate(unsigned int cpu)
+static void twd_calibrate_rate(unsigned int cpu)
{
void __iomem *base = TWD_BASE(cpu);
unsigned long load, count;
@@ -148,7 +148,7 @@ static void __cpuinit twd_calibrate_rate(unsigned int cpu)
/*
* Setup the local clock events for a CPU.
*/
-void __cpuinit local_timer_setup(unsigned int cpu)
+void local_timer_setup(unsigned int cpu)
{
struct clock_event_device *clk = &per_cpu(local_clockevent, cpu);
unsigned long flags;
@@ -178,7 +178,7 @@ void __cpuinit local_timer_setup(unsigned int cpu)
/*
* take a local timer down
*/
-void __cpuexit local_timer_stop(unsigned int cpu)
+void local_timer_stop(unsigned int cpu)
{
__raw_writel(0, TWD_BASE(cpu) + TWD_TIMER_CONTROL);
}
@@ -190,7 +190,7 @@ static void dummy_timer_set_mode(enum clock_event_mode mode,
{
}

-void __cpuinit local_timer_setup(unsigned int cpu)
+void local_timer_setup(unsigned int cpu)
{
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
struct clock_event_device *clk = &per_cpu(local_clockevent, cpu);

diff --git a/arch/arm/mach-realview/platsmp.c b/arch/arm/mach-realview/platsmp.c
index e102aeb..3525a8e 100644
--- a/arch/arm/mach-realview/platsmp.c
+++ b/arch/arm/mach-realview/platsmp.c
@@ -29,7 +29,7 @@ extern void realview_secondary_startup(void);
* control for which core is the next to come out of the secondary
* boot "holding pen"
*/
-volatile int __cpuinitdata pen_release = -1;
+volatile int pen_release = -1;

static unsigned int __init get_core_count(void)
{
@@ -72,7 +72,7 @@ static void scu_enable(void)

static DEFINE_SPINLOCK(boot_lock);

-void __cpuinit platform_secondary_init(unsigned int cpu)
+void platform_secondary_init(unsigned int cpu)
{
trace_hardirqs_off();

@@ -107,7 +107,7 @@ void __cpuinit platform_secondary_init(unsigned int cpu)
spin_unlock(&boot_lock);
}

-int __cpuinit boot_secondary(unsigned int cpu, struct task_struct *idle)
+int boot_secondary(unsigned int cpu, struct task_struct *idle)
{
unsigned long timeout;

diff --git a/arch/cris/arch-v32/kernel/smp.c b/arch/cris/arch-v32/kernel/smp.c
index 52e16c6..b93b3d0 100644
--- a/arch/cris/arch-v32/kernel/smp.c
+++ b/arch/cris/arch-v32/kernel/smp.c
@@ -211,7 +211,7 @@ int setup_profiling_timer(unsigned int multiplier)
*/
unsigned long cache_decay_ticks = 1;

-int __cpuinit __cpu_up(unsigned int cpu)
+int __cpu_up(unsigned int cpu)
{
smp_boot_one_cpu(cpu);
return cpu_online(cpu) ? 0 : -ENOSYS;
diff --git a/arch/frv/kernel/setup.c b/arch/frv/kernel/setup.c
index 0669e13..ad2714a 100644
--- a/arch/frv/kernel/setup.c
+++ b/arch/frv/kernel/setup.c
@@ -708,7 +708,7 @@ static void __init reserve_dma_coherent(void)
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
/*
 * calibrate the delay loop
 */
-void __cpuinit calibrate_delay(void)
+void calibrate_delay(void)
{
loops_per_jiffy = __delay_loops_MHz * (1000000 / HZ);

diff --git a/arch/ia64/kernel/err_inject.c b/arch/ia64/kernel/err_inject.c
index c539c68..aad440 100644
--- a/arch/ia64/kernel/err_inject.c
+++ b/arch/ia64/kernel/err_inject.c
@@ -225,17 +225,17 @@ static struct attribute_group err_inject_attr_group = {
.name = "err_inject"
};
/* Add/Remove err_inject interface for CPU device */
-static int __cpuinit err_inject_add_dev(struct sys_device * sys_dev)
+static int err_inject_add_dev(struct sys_device * sys_dev)
{
return sysfs_create_group(&sys_dev->kobj, &err_inject_attr_group);
}

-static int __cpuinit err_inject_remove_dev(struct sys_device * sys_dev)
+static int err_inject_remove_dev(struct sys_device * sys_dev)
{
sysfs_remove_group(&sys_dev->kobj, &err_inject_attr_group);
return 0;
}
-static int __cpuinit err_inject_cpu_callback(struct notifier_block *nfb,
+static int err_inject_cpu_callback(struct notifier_block *nfb,
unsigned long action, void *hcpu)
{
unsigned int cpu = (unsigned long)hcpu;
@@ -256,7 +256,7 @@ static int __cpuinit err_inject_cpu_callback(struct notifier_block *nfb,
return NOTIFY_OK;
}

-static struct notifier_block __cpuinitdata err_inject_cpu_notifier =
+static struct notifier_block err_inject_cpu_notifier =
{
.notifier_call = err_inject_cpu_callback,
};
diff --git a/arch/ia64/kernel/mca.c b/arch/ia64/kernel/mca.c
index 7dd96c1..c886f95 100644
--- a/arch/ia64/kernel/mca.c
+++ b/arch/ia64/kernel/mca.c
@@ -629,7 +629,7 @@ ia64_mca_register_cpev (int cpev)
 * Outputs
 * None
 */
-void __cpuinit
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
+void
ia64_mca_cmc_vector_setup (void)
{
cmcv_reg_t cmcv;
@@ -1779,7 +1779,7 @@ static struct irqaction mca_cpep_irqaction = {
* format most of the fields.
*/

-static void __cpuinit
+static void
format_mca_init_stack(void *mca_data, unsigned long offset,
const char *type, int cpu)
{
@@ -1809,7 +1809,7 @@ static void * __init_refok mca_bootmem(void)
}

/* Do per-CPU MCA-related initialization. */
-void __cpuinit
+void
ia64_mca_cpu_init(void *cpu_data)
{
void *pal_vaddr;
@@ -1861,7 +1861,7 @@ ia64_mca_cpu_init(void *cpu_data)
PAGE_KERNEL));
}

-static void __cpuinit ia64_mca_cmc_vector_adjust(void *dummy)
+static void ia64_mca_cmc_vector_adjust(void *dummy)
{
unsigned long flags;

@@ -1871,7 +1871,7 @@ static void __cpuinit ia64_mca_cmc_vector_adjust(void *dummy)
local_irq_restore(flags);
}

-static int __cpuinit mca_cpu_callback(struct notifier_block *nfb,
+static int mca_cpu_callback(struct notifier_block *nfb,
unsigned long action,
void *hcpu)
{
@@ -1887,7 +1887,7 @@ static int __cpuinit mca_cpu_callback(struct notifier_block *nfb,
return NOTIFY_OK;
}

-static struct notifier_block mca_cpu_notifier __cpuinitdata = {
+static struct notifier_block mca_cpu_notifier = {
.notifier_call = mca_cpu_callback
};

diff --git a/arch/ia64/kernel/numa.c b/arch/ia64/kernel/numa.c
index c93420c..d288cde 100644
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid of __cpuinit and __cpuexit

```
--- a/arch/ia64/kernel/numa.c
+++ b/arch/ia64/kernel/numa.c
@@ -30,7 +30,7 @@ EXPORT_SYMBOL(cpu_to_node_map);
cpumask_t node_to_cpu_mask[MAX_NUMNODES] __cacheline_aligned;
EXPORT_SYMBOL(node_to_cpu_mask);

-void __cpuinit map_cpu_to_node(int cpu, int nid)
+void map_cpu_to_node(int cpu, int nid)
{
int oldnid;
if (nid < 0) { /* just initialize by zero */
@@ -51,7 +51,7 @@ void __cpuinit map_cpu_to_node(int cpu, int nid)
return;
}

-void __cpuinit unmap_cpu_from_node(int cpu, int nid)
+void unmap_cpu_from_node(int cpu, int nid)
{
WARN_ON(!cpu_isset(cpu, node_to_cpu_mask[nid]));
WARN_ON(cpu_to_node_map[cpu] != nid);
diff --git a/arch/ia64/kernel/palinfo.c b/arch/ia64/kernel/palinfo.c
index e5c57f4..3a8c519 100644
--- a/arch/ia64/kernel/palinfo.c
+++ b/arch/ia64/kernel/palinfo.c
@@ -968,7 +968,7 @@ palinfo_read_entry(char *page, char **start, off_t off, int count, int *eof, voi
return len;
}

-static void __cpuinit
+static void
create_palinfo_proc_entries(unsigned int cpu)
{
#define CPUSTR "cpu%d"
@@ -1029,7 +1029,7 @@ remove_palinfo_proc_entries(unsigned int hcpu)
}
}

-static int __cpuinit palinfo_cpu_callback(struct notifier_block *nfb,
+static int palinfo_cpu_callback(struct notifier_block *nfb,
unsigned long action, void *hcpu)
{
unsigned int hotcpu = (unsigned long)hcpu;
diff --git a/arch/ia64/kernel/salinfo.c b/arch/ia64/kernel/salinfo.c
index ecb9eb7..84eba7a 100644
--- a/arch/ia64/kernel/salinfo.c
+++ b/arch/ia64/kernel/salinfo.c
@@ -574,7 +574,7 @@ static const struct file_operations salinfo_data_fops = {
.write = salinfo_log_write,
};

-static int __cpuinit
```

[PATCH] get rid of __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
+static int
salinfo_cpu_callback(struct notifier_block *nb, unsigned long action, void *hcpu)
{
    unsigned int i, cpu = (unsigned long)hcpu;
    @@ -615,7 +615,7 @@ salinfo_cpu_callback(struct notifier_block *nb, unsigned long action, void *hcpu
    return NOTIFY_OK;
}

-static struct notifier_block salinfo_cpu_notifier __cpuinitdata =
+static struct notifier_block salinfo_cpu_notifier =
{
    .notifier_call = salinfo_cpu_callback,
    .priority = 0,
diff --git a/arch/ia64/kernel/setup.c b/arch/ia64/kernel/setup.c
index de636b2..3699432 100644
--- a/arch/ia64/kernel/setup.c
+++ b/arch/ia64/kernel/setup.c
    @@ -742,7 +742,7 @@ const struct seq_operations cpuinfo_op = {
    #define MAX_BRANDS 8
    static char brandname[MAX_BRANDS][128];

-static char * __cpuinit
+static char *
get_model_name(__u8 family, __u8 model)
{
    static int overflow;
    @@ -772,7 +772,7 @@ get_model_name(__u8 family, __u8 model)
    return "Unknown";
}

-static void __cpuinit
+static void
identify_cpu (struct cpuinfo_ia64 *c)
{
    union {
    @@ -852,7 +852,7 @@ setup_per_cpu_areas (void)
    * In addition, the minimum of the i-cache stride sizes is calculated for
    * "flush_icache_range()".
    */
-static void __cpuinit
+static void
get_max_cacheline_size (void)
{
    unsigned long line_size, max = 1;
    @@ -909,10 +909,10 @@ get_max_cacheline_size (void)
    * cpu_init() initializes state that is per-CPU. This function acts
    * as a 'CPU state barrier', nothing should get across.
    */
-void __cpuinit
+void
cpu_init (void)
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
{
- extern void __cpuinit ia64_mmu_init (void *);
+ extern void ia64_mmu_init (void *);
static unsigned long max_num_phys_stacked = IA64_NUM_PHYS_STACK_REG;
unsigned long num_phys_stacked;
pal_vm_info_2_u_t vmi;
diff --git a/arch/ia64/kernel/smpboot.c b/arch/ia64/kernel/smpboot.c
index 1dcbb85..5fd38c7 100644
--- a/arch/ia64/kernel/smpboot.c
+++ b/arch/ia64/kernel/smpboot.c
@@ -373,7 +373,7 @@ smp_setup_percpu_timer (void)
{
}

-static void __cpuinit
+static void
smp_callin (void)
{
int cpuid, phys_id, itc_master;
@@ -464,7 +464,7 @@ smp_callin (void)
/*
* Activate a secondary processor. head.S calls this.
*/
-int __cpuinit
+int
start_secondary (void *unused)
{
/* Early console may use I/O ports */
@@ -481,7 +481,7 @@ start_secondary (void *unused)
return 0;
}

-struct pt_regs * __cpuinit idle_regs(struct pt_regs *regs)
+struct pt_regs * idle_regs(struct pt_regs *regs)
{
return NULL;
}
@@ -493,7 +493,7 @@ struct create_idle {
int cpu;
};

-void __cpuinit
+void
do_fork_idle(struct work_struct *work)
{
struct create_idle *c_idle =
@@ -503,7 +503,7 @@ do_fork_idle(struct work_struct *work)
complete(&c_idle->done);
}

-static int __cpuinit
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
+static int
do_boot_cpu (int sapicid, int cpu)
{
int timeout;
@@ -807,7 +807,7 @@ set_cpu_sibling_map(int cpu)
}
}

-int __cpuinit
+int
__cpu_up (unsigned int cpu)
{
int ret;
diff --git a/arch/ia64/kernel/topology.c b/arch/ia64/kernel/topology.c
index 26228e2..913e1ad 100644
--- a/arch/ia64/kernel/topology.c
+++ b/arch/ia64/kernel/topology.c
@@ -131,11 +131,11 @@ struct cpu_cache_info {
struct kobject kobj;
};

-static struct cpu_cache_info all_cpu_cache_info[NR_CPUS] __cpuinitdata;
+static struct cpu_cache_info all_cpu_cache_info[NR_CPUS];
#define LEAF_KOBJECT_PTR(x,y) (&all_cpu_cache_info[x].cache_leaves[y])

#ifdef CONFIG_SMP
-static void __cpuinit cache_shared_cpu_map_setup( unsigned int cpu,
+static void cache_shared_cpu_map_setup( unsigned int cpu,
struct cache_info * this_leaf)
{
pal_cache_shared_info_t csi;
@@ -170,7 +170,7 @@ static void __cpuinit cache_shared_cpu_map_setup( unsigned int cpu,
&csi) == PAL_STATUS_SUCCESS);
}
#else
-static void __cpuinit cache_shared_cpu_map_setup(unsigned int cpu,
+static void cache_shared_cpu_map_setup(unsigned int cpu,
struct cache_info * this_leaf)
{
cpu_set(cpu, this_leaf->shared_cpu_map);
@@ -293,7 +293,7 @@ static struct kobj_type cache_ktype_percpu_entry = {
.sysfs_ops = &cache_sysfs_ops,
};

-static void __cpuinit cpu_cache_sysfs_exit(unsigned int cpu)
+static void cpu_cache_sysfs_exit(unsigned int cpu)
{
kfree(all_cpu_cache_info[cpu].cache_leaves);
all_cpu_cache_info[cpu].cache_leaves = NULL;
@@ -302,7 +302,7 @@ static void __cpuinit cpu_cache_sysfs_exit(unsigned int cpu)
return;

```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
}

-static int __cpuinit cpu_cache_sysfs_init(unsigned int cpu)
+static int cpu_cache_sysfs_init(unsigned int cpu)
{
u64 i, levels, unique_caches;
pal_cache_config_info_t cci;
@@ -346,7 +346,7 @@ static int __cpuinit cpu_cache_sysfs_init(unsigned int cpu)
}

/* Add cache interface for CPU device */
-static int __cpuinit cache_add_dev(struct sys_device * sys_dev)
+static int cache_add_dev(struct sys_device * sys_dev)
{
unsigned int cpu = sys_dev->id;
unsigned long i, j;
@@ -392,7 +392,7 @@ static int __cpuinit cache_add_dev(struct sys_device * sys_dev)
}

/* Remove cache interface for CPU device */
-static int __cpuinit cache_remove_dev(struct sys_device * sys_dev)
+static int cache_remove_dev(struct sys_device * sys_dev)
{
unsigned int cpu = sys_dev->id;
unsigned long i;
@@ -416,7 +416,7 @@ static int __cpuinit cache_remove_dev(struct sys_device * sys_dev)
* When a cpu is hot-plugged, do a check and initiate
* cache kobject if necessary
*/
-static int __cpuinit cache_cpu_callback(struct notifier_block *nfb,
+static int cache_cpu_callback(struct notifier_block *nfb,
unsigned long action, void *hcpu)
{
unsigned int cpu = (unsigned long)hcpu;
@@ -436,7 +436,7 @@ static int __cpuinit cache_cpu_callback(struct notifier_block *nfb,
return NOTIFY_OK;
}

-static struct notifier_block __cpuinitdata cache_cpu_notifier =
+static struct notifier_block cache_cpu_notifier =
{
.notifier_call = cache_cpu_callback
};
diff --git a/arch/ia64/mm/contig.c b/arch/ia64/mm/contig.c
index 0ee085e..72cb323 100644
--- a/arch/ia64/mm/contig.c
+++ b/arch/ia64/mm/contig.c
@@ -151,7 +151,7 @@ static void *cpu_data;
*
* Allocate and setup per-cpu data areas.
*/
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
-void * __cpuinit
+void *
per_cpu_init (void)
{
int cpu;
diff --git a/arch/ia64/mm/discontig.c b/arch/ia64/mm/discontig.c
index d8c5fcd..c3bead8 100644
--- a/arch/ia64/mm/discontig.c
+++ b/arch/ia64/mm/discontig.c
@@ -493,7 +493,7 @@ void __init find_memory(void)
* find_ernode_space() does most of this already, we just need to set
* local_per_cpu_offset
*/
-void __cpuinit *per_cpu_init(void)
+void *per_cpu_init(void)
{
int cpu;
static int first_time = 1;
diff --git a/arch/ia64/sn/kernel/setup.c b/arch/ia64/sn/kernel/setup.c
index bb1d249..b248b7d 100644
--- a/arch/ia64/sn/kernel/setup.c
+++ b/arch/ia64/sn/kernel/setup.c
@@ -192,7 +192,7 @@ void __init early_sn_setup(void)
}

extern int platform_intr_list[];
-static int __cpuinitdata shub_1_1_found;
+static int shub_1_1_found;

/*
* sn_check_for_wars
@@ -561,7 +561,7 @@ static void __init sn_init_pdas(char **cmdline_p)
* Also sets up a few fields in the nodepda. Also known as
* platform_cpu_init() by the ia64 machvec code.
*/
-void __cpuinit sn_cpu_init(void)
+void sn_cpu_init(void)
{
int cpuid;
int cpuphyid;
diff --git a/arch/m32r/kernel/smpboot.c b/arch/m32r/kernel/smpboot.c
index fc29948..1bc5a47 100644
--- a/arch/m32r/kernel/smpboot.c
+++ b/arch/m32r/kernel/smpboot.c
@@ -352,7 +352,7 @@ static void __init do_boot_cpu(int phys_id)
}
}

-int __cpuinit __cpu_up(unsigned int cpu_id)
+int __cpu_up(unsigned int cpu_id)
{
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

int timeout;

diff --git a/arch/mips/kernel/cevt-bcm1480.c b/arch/mips/kernel/cevt-bcm1480.c

index 0a57f86..9cf0790 100644

--- a/arch/mips/kernel/cevt-bcm1480.c

+++ b/arch/mips/kernel/cevt-bcm1480.c

@@ -107,7 +107,7 @@ static DEFINE_PER_CPU(struct clock_event_device, sibyte_hpt_clokevent);

static DEFINE_PER_CPU(struct irqaction, sibyte_hpt_irqaction);

static DEFINE_PER_CPU(char [18], sibyte_hpt_name);

-void __cpuinit sb1480_clokevent_init(void)

+void sb1480_clokevent_init(void)

{

unsigned int cpu = smp_processor_id();

unsigned int irq = K_BCM1480_INT_TIMER_0 + cpu;

diff --git a/arch/mips/kernel/cevt-r4k.c b/arch/mips/kernel/cevt-r4k.c

index 4a4c59f..3812578 100644

--- a/arch/mips/kernel/cevt-r4k.c

+++ b/arch/mips/kernel/cevt-r4k.c

@@ -160,7 +160,7 @@ int c0_compare_int_usable(void)

#ifndef CONFIG_MIPS_MT_SMTC

-int __cpuinit mips_clokevent_init(void)

+int mips_clokevent_init(void)

{

uint64_t mips_freq = mips_hpt_frequency;

unsigned int cpu = smp_processor_id();

diff --git a/arch/mips/kernel/cevt-sb1250.c b/arch/mips/kernel/cevt-sb1250.c

index 63ac3ad..fb68565 100644

--- a/arch/mips/kernel/cevt-sb1250.c

+++ b/arch/mips/kernel/cevt-sb1250.c

@@ -105,7 +105,7 @@ static DEFINE_PER_CPU(struct clock_event_device, sibyte_hpt_clokevent);

static DEFINE_PER_CPU(struct irqaction, sibyte_hpt_irqaction);

static DEFINE_PER_CPU(char [18], sibyte_hpt_name);

-void __cpuinit sb1250_clokevent_init(void)

+void sb1250_clokevent_init(void)

{

unsigned int cpu = smp_processor_id();

unsigned int irq = K_INT_TIMER_0 + cpu;

diff --git a/arch/mips/kernel/cevt-smtc.c b/arch/mips/kernel/cevt-smtc.c

index 5162fe4..673d467 100644

--- a/arch/mips/kernel/cevt-smtc.c

+++ b/arch/mips/kernel/cevt-smtc.c

@@ -245,7 +245,7 @@ irqreturn_t c0_compare_interrupt(int irq, void *dev_id)

}

-int __cpuinit mips_clokevent_init(void)

+int mips_clokevent_init(void)

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
{
uint64_t mips_freq = mips_hpt_frequency;
unsigned int cpu = smp_processor_id();
diff --git a/arch/mips/kernel/cpu-bugs64.c b/arch/mips/kernel/cpu-bugs64.c
index 02b7713..a4ce82a 100644
--- a/arch/mips/kernel/cpu-bugs64.c
+++ b/arch/mips/kernel/cpu-bugs64.c
@@ -167,7 +167,7 @@ static inline void check_mult_sh(void)
panic(bug64hit, !R4000_WAR ? r4kwar : nowar);
}

-static volatile int daddi_ov __cpuinitdata = 0;
+static volatile int daddi_ov = 0;

asmlinkage void __init do_daddi_ov(struct pt_regs *regs)
{
@@ -239,7 +239,7 @@ static inline void check_daddi(void)
panic(bug64hit, !DADDI_WAR ? daddiwar : nowar);
}

-int daddiu_bug __cpuinitdata = -1;
+int daddiu_bug = -1;

static inline void check_daddiu(void)
{
diff --git a/arch/mips/kernel/cpu-probe.c b/arch/mips/kernel/cpu-probe.c
index 0cf1545..12dbcd6 100644
--- a/arch/mips/kernel/cpu-probe.c
+++ b/arch/mips/kernel/cpu-probe.c
@@ -544,7 +544,7 @@ static inline void cpu_probe_legacy(struct cpuinfo_mips *c)
}
}

-static char unknown_isa[] __cpuinitdata = KERN_ERR \
+static char unknown_isa[] = KERN_ERR \
"Unsupported ISA type, c0.config0: %d.";

static inline unsigned int decode_config0(struct cpuinfo_mips *c)
@@ -650,7 +650,7 @@ static inline unsigned int decode_config3(struct cpuinfo_mips *c)
return config3 & MIPS_CONF_M;
}

-static void __cpuinit decode_configs(struct cpuinfo_mips *c)
+static void decode_configs(struct cpuinfo_mips *c)
{
/* MIPS32 or MIPS64 compliant CPU. */
c->options = MIPS_CPU_4KEX | MIPS_CPU_4K_CACHE | MIPS_CPU_COUNTER |
@@ -820,7 +820,7 @@ const char *__cpu_name[NR_CPUS];
/*
* Name a CPU
*/
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpunit and __cpuexit

```
-static __cpunit const char *cpu_to_name(struct cpuinfo_mips *c)
+static const char *cpu_to_name(struct cpuinfo_mips *c)
{
const char *name = NULL;

@@ -903,7 +903,7 @@ static __cpunit const char *cpu_to_name(struct cpuinfo_mips *c)
return name;
}

-__cpunit void cpu_probe(void)
+void cpu_probe(void)
{
struct cpuinfo_mips *c = &current_cpu_data;
unsigned int cpu = smp_processor_id();
@@ -966,7 +966,7 @@ __cpunit void cpu_probe(void)
c->srsets = 1;
}

-__cpunit void cpu_report(void)
+void cpu_report(void)
{
struct cpuinfo_mips *c = &current_cpu_data;

diff --git a/arch/mips/kernel/smp-mt.c b/arch/mips/kernel/smp-mt.c
index 87a1816..6a5c3af 100644
--- a/arch/mips/kernel/smp-mt.c
+++ b/arch/mips/kernel/smp-mt.c
@@ -149,7 +149,7 @@ static void vsmp_send_ipi_mask(cpumask_t mask, unsigned int action)
vsmp_send_ipi_single(i, action);
}

-static void __cpunit vsmp_init_secondary(void)
+static void vsmp_init_secondary(void)
{
extern int gic_present;

@@ -162,7 +162,7 @@ static void __cpunit vsmp_init_secondary(void)
STATUSF_IP6 | STATUSF_IP7);
}

-static void __cpunit vsmp_smp_finish(void)
+static void vsmp_smp_finish(void)
{
/* CDFIXME: remove this? */
write_c0_compare(read_c0_count() + (8* mips_hpt_frequency/HZ));
@@ -188,7 +188,7 @@ static void vsmp_cpus_done(void)
* (unsigned long)idle->thread_info the gp
* assumes a 1:1 mapping of TC => VPE
*/
-static void __cpunit vsmp_boot_secondary(int cpu, struct task_struct *idle)
+static void vsmp_boot_secondary(int cpu, struct task_struct *idle)
```

[PATCH] get rid if __cpunit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
{
struct thread_info *gp = task_thread_info(idle);
dvppe();
diff --git a/arch/mips/kernel/smp-up.c b/arch/mips/kernel/smp-up.c
index ead6c30..372f575 100644
--- a/arch/mips/kernel/smp-up.c
+++ b/arch/mips/kernel/smp-up.c
@@ -27,11 +27,11 @@ static inline void up_send_ipi_mask(cpumask_t mask, unsigned int action)
* After we've done initial boot, this function is called to allow the
* board code to clean up state, if needed
*/
-void __cpuinit up_init_secondary(void)
+void up_init_secondary(void)
{
}

-void __cpuinit up_smp_finish(void)
+void up_smp_finish(void)
{
}

@@ -43,7 +43,7 @@ void up_cpus_done(void)
/*
* Firmware CPU startup hook
*/
-void __cpuinit up_boot_secondary(int cpu, struct task_struct *idle)
+void up_boot_secondary(int cpu, struct task_struct *idle)
{
}

diff --git a/arch/mips/kernel/smp.c b/arch/mips/kernel/smp.c
index 7b59cfb..7c392d4 100644
--- a/arch/mips/kernel/smp.c
+++ b/arch/mips/kernel/smp.c
@@ -85,7 +85,7 @@ static inline void set_cpu_sibling_map(int cpu)

struct plat_smp_ops *mp_ops;

-__cpuinit void register_smp_ops(struct plat_smp_ops *ops)
+void register_smp_ops(struct plat_smp_ops *ops)
{
if (mp_ops)
printk(KERN_WARNING "Overriding previously set SMP ops\n");
@@ -97,7 +97,7 @@ __cpuinit void register_smp_ops(struct plat_smp_ops *ops)
* First C code run on the secondary CPUs after being started up by
* the master.
*/
-asm__linkage __cpuinit void start_secondary(void)
+asm__linkage void start_secondary(void)
{
unsigned int cpu;
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
@@ -209,7 +209,7 @@ void __devinit smp_prepare_boot_cpu(void)
* and keep control until "cpu_online(cpu)" is set. Note: cpu is
* physical, not logical.
*/
-int __cpuinit __cpu_up(unsigned int cpu)
+int __cpu_up(unsigned int cpu)
{
struct task_struct *idle;

diff --git a/arch/mips/kernel/smtc.c b/arch/mips/kernel/smtc.c
index 897fb2b..1355da7 100644
--- a/arch/mips/kernel/smtc.c
+++ b/arch/mips/kernel/smtc.c
@@ -578,7 +578,7 @@ void smtc_prepare_cpus(int cpus)
* (unsigned long)idle->thread_info the gp
*
*/
-void __cpuinit smtc_boot_secondary(int cpu, struct task_struct *idle)
+void smtc_boot_secondary(int cpu, struct task_struct *idle)
{
extern u32 kernelsp[NR_CPUS];
unsigned long flags;
diff --git a/arch/mips/kernel/spram.c b/arch/mips/kernel/spram.c
index 6ddb507..f52f588 100644
--- a/arch/mips/kernel/spram.c
+++ b/arch/mips/kernel/spram.c
@@ -39,7 +39,7 @@
/*
* Different semantics to the set_c0_* function built by __BUILD_SET_C0
*/
-static __cpuinit unsigned int bis_c0_errctl(unsigned int set)
+static unsigned int bis_c0_errctl(unsigned int set)
{
unsigned int res;
res = read_c0_errctl();
@@ -47,7 +47,7 @@ static __cpuinit unsigned int bis_c0_errctl(unsigned int set)
return res;
}

-static __cpuinit void ispram_store_tag(unsigned int offset, unsigned int data)
+static void ispram_store_tag(unsigned int offset, unsigned int data)
{
unsigned int errctl;

@@ -66,7 +66,7 @@ static __cpuinit void ispram_store_tag(unsigned int offset, unsigned int data)
}

-static __cpuinit unsigned int ispram_load_tag(unsigned int offset)
+static unsigned int ispram_load_tag(unsigned int offset)
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
{
unsigned int data;
unsigned int errctl;
@@ -84,7 +84,7 @@ static __cpuinit unsigned int dspram_load_tag(unsigned int offset)
return data;
}

-static __cpuinit void dspram_store_tag(unsigned int offset, unsigned int data)
+static void dspram_store_tag(unsigned int offset, unsigned int data)
{
unsigned int errctl;

@@ -100,7 +100,7 @@ static __cpuinit void dspram_store_tag(unsigned int offset, unsigned int data)
}

-static __cpuinit unsigned int dspram_load_tag(unsigned int offset)
+static unsigned int dspram_load_tag(unsigned int offset)
{
unsigned int data;
unsigned int errctl;
@@ -117,7 +117,7 @@ static __cpuinit unsigned int dspram_load_tag(unsigned int offset)
return data;
}

-static __cpuinit void probe_spram(char *type,
+static void probe_spram(char *type,
unsigned int base,
unsigned int (*read)(unsigned int),
void (*write)(unsigned int, unsigned int))
@@ -199,7 +199,7 @@ static __cpuinit void probe_spram(char *type,
}
}

-__cpuinit void spram_config(void)
+void spram_config(void)
{
struct cpuinfo_mips *c = &current_cpu_data;
unsigned int config0;
diff --git a/arch/mips/kernel/time.c b/arch/mips/kernel/time.c
index 1f467d5..edc657d 100644
--- a/arch/mips/kernel/time.c
+++ b/arch/mips/kernel/time.c
@@ -87,7 +87,7 @@ void __init clocksource_set_clock(struct clocksource *cs, unsigned int clock)
cs->mult = (u32) temp;
}

-void __cpuinit clockevent_set_clock(struct clock_event_device *cd,
+void clockevent_set_clock(struct clock_event_device *cd,
unsigned int clock)
{
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
u64 temp;
diff --git a/arch/mips/kernel/traps.c b/arch/mips/kernel/traps.c
index 80b9e07..9546269 100644
--- a/arch/mips/kernel/traps.c
+++ b/arch/mips/kernel/traps.c
@@ -1416,7 +1416,7 @@ int cp0_compare_irq;
int cp0_perfcount_irq;
EXPORT_SYMBOL_GPL(cp0_perfcount_irq);

-static int __cpuinitdata noulri;
+static int noulri;

static int __init ulri_disable(char *s)
{
@@ -1427,7 +1427,7 @@ static int __init ulri_disable(char *s)
}
__setup("noulri", ulri_disable);

-void __cpuinit per_cpu_trap_init(void)
+void per_cpu_trap_init(void)
{
unsigned int cpu = smp_processor_id();
unsigned int status_set = ST0_CU0;
@@ -1542,11 +1542,11 @@ void __init set_handler(unsigned long offset, void *addr, unsigned long size)
local_flush_icache_range(ebase + offset, ebase + offset + size);
}

-static char panic_null_cerr[] __cpuinitdata =
+static char panic_null_cerr[] =
"Trying to set NULL cache error exception handler";

/* Install uncached CPU exception handler */
-void __cpuinit set_uncached_handler(unsigned long offset, void *addr,
+void set_uncached_handler(unsigned long offset, void *addr,
unsigned long size)
{
#ifdef CONFIG_32BIT
diff --git a/arch/mips/kernel/watch.c b/arch/mips/kernel/watch.c
index c154069..63ddb10 100644
--- a/arch/mips/kernel/watch.c
+++ b/arch/mips/kernel/watch.c
@@ -100,7 +100,7 @@ void mips_clear_watch_registers(void)
}
}

-__cpuinit void mips_probe_watch_registers(struct cpuinfo_mips *c)
+void mips_probe_watch_registers(struct cpuinfo_mips *c)
{
unsigned int t;
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
index a6d1c77..12b2ede 100644
--- a/arch/mips/lib/uncached.c
+++ b/arch/mips/lib/uncached.c
@@ -36,7 +36,7 @@
 * values, so we can avoid sharing the same stack area between a cached
 * and the uncached mode.
 */
-unsigned long __cpuinit run_uncached(void *func)
+unsigned long run_uncached(void *func)
{
register long sp __asm__("$sp");
register long ret __asm__("$2");
diff --git a/arch/mips/mipssim/sim_smtc.c b/arch/mips/mipssim/sim_smtc.c
index d6e4f65..d40d38f 100644
--- a/arch/mips/mipssim/sim_smtc.c
+++ b/arch/mips/mipssim/sim_smtc.c
@@ -54,7 +54,7 @@ static inline void ssmtc_send_ipi_mask(cpumask_t mask, unsigned int action)
/*
 * Post-config but pre-boot cleanup entry point
 */
-static void __cpuinit ssmtc_init_secondary(void)
+static void ssmtc_init_secondary(void)
{
void smtc_init_secondary(void);

@@ -64,7 +64,7 @@ static void __cpuinit ssmtc_init_secondary(void)
/*
 * SMP initialization finalization entry point
 */
-static void __cpuinit ssmtc_smp_finish(void)
+static void ssmtc_smp_finish(void)
{
smtc_smp_finish();
}
@@ -79,7 +79,7 @@ static void ssmtc_cpus_done(void)
/*
 * Platform "CPU" startup hook
 */
-static void __cpuinit ssmtc_boot_secondary(int cpu, struct task_struct *idle)
+static void ssmtc_boot_secondary(int cpu, struct task_struct *idle)
{
smtc_boot_secondary(cpu, idle);
}
diff --git a/arch/mips/mipssim/sim_time.c b/arch/mips/mipssim/sim_time.c
index 881ecbc..7cf1061 100644
--- a/arch/mips/mipssim/sim_time.c
+++ b/arch/mips/mipssim/sim_time.c
@@ -83,7 +83,7 @@ static void mips_timer_dispatch(void)
}
```

[PATCH] get rid of __cpuinit and __cpuexit

```
-unsigned __cpuinit get_c0_compare_int(void)
+unsigned get_c0_compare_int(void)
{
#ifdef MSC01E_INT_BASE
if (cpu_has_veic) {
diff --git a/arch/mips/mm/c-r3k.c b/arch/mips/mm/c-r3k.c
index 5500c20..00d499e 100644
--- a/arch/mips/mm/c-r3k.c
+++ b/arch/mips/mm/c-r3k.c
@@ -26,7 +26,7 @@
static unsigned long ical_size, dcache_size; /* Size in bytes */
static unsigned long ical_lsize, dcache_lsize; /* Size in bytes */

-unsigned long __cpuinit r3k_cache_size(unsigned long ca_flags)
+unsigned long r3k_cache_size(unsigned long ca_flags)
{
unsigned long flags, status, dummy, size;
volatile unsigned long *p;
@@ -61,7 +61,7 @@ unsigned long __cpuinit r3k_cache_size(unsigned long ca_flags)
return size * sizeof(*p);
}

-unsigned long __cpuinit r3k_cache_lsize(unsigned long ca_flags)
+unsigned long r3k_cache_lsize(unsigned long ca_flags)
{
unsigned long flags, status, lsize, i;
volatile unsigned long *p;
@@ -90,7 +90,7 @@ unsigned long __cpuinit r3k_cache_lsize(unsigned long ca_flags)
return lsize * sizeof(*p);
}

-static void __cpuinit r3k_probe_cache(void)
+static void r3k_probe_cache(void)
{
dcache_size = r3k_cache_size(ST0_ISC);
if (dcache_size)
@@ -307,7 +307,7 @@ static void r3k_dma_cache_wback_inv(unsigned long start, unsigned long size)
r3k_flush_dcache_range(start, start + size);
}

-void __cpuinit r3k_cache_init(void)
+void r3k_cache_init(void)
{
extern void build_clear_page(void);
extern void build_copy_page(void);
diff --git a/arch/mips/mm/c-r4k.c b/arch/mips/mm/c-r4k.c
index 6e99665..7c14836 100644
--- a/arch/mips/mm/c-r4k.c
+++ b/arch/mips/mm/c-r4k.c
@@ -100,7 +100,7 @@ static inline void r4k_blast_dcache_page_dc32(unsigned long addr)
blast_dcache32_page(addr);
```

[PATCH] get rid of __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

}

-static void __cpuinit r4k_blast_dcachepage_setup(void)

+static void r4k_blast_dcachepage_setup(void)

{

unsigned long dc_lsize = cpu_dcacheline_size();

@@ -114,7 +114,7 @@ static void __cpuinit r4k_blast_dcachepage_setup(void)

static void (* r4k_blast_dcachepage_indexed)(unsigned long addr);

-static void __cpuinit r4k_blast_dcachepage_indexed_setup(void)

+static void r4k_blast_dcachepage_indexed_setup(void)

{

unsigned long dc_lsize = cpu_dcacheline_size();

@@ -128,7 +128,7 @@ static void __cpuinit r4k_blast_dcachepage_indexed_setup(void)

static void (* r4k_blast_dcachepage_indexed)(void);

-static void __cpuinit r4k_blast_dcachepage_setup_size(void)

+static void r4k_blast_dcachepage_setup_size(void)

{

unsigned long dc_lsize = cpu_dcacheline_size();

@@ -213,7 +213,7 @@ static inline void tx49_blast_icache32_page_indexed(unsigned long page)

static void (* r4k_blast_icache32_page_indexed)(unsigned long addr);

-static void __cpuinit r4k_blast_icache32_page_indexed_setup(void)

+static void r4k_blast_icache32_page_indexed_setup(void)

{

unsigned long ic_lsize = cpu_icacheline_size();

@@ -230,7 +230,7 @@ static void __cpuinit r4k_blast_icache32_page_indexed_setup(void)

static void (* r4k_blast_icache32_page_indexed_indexed)(unsigned long addr);

-static void __cpuinit r4k_blast_icache32_page_indexed_indexed_setup(void)

+static void r4k_blast_icache32_page_indexed_indexed_setup(void)

{

unsigned long ic_lsize = cpu_icacheline_size();

@@ -254,7 +254,7 @@ static void __cpuinit r4k_blast_icache32_page_indexed_indexed_setup(void)

static void (* r4k_blast_icache32_page_indexed_indexed)(void);

-static void __cpuinit r4k_blast_icache32_page_indexed_indexed_setup_size(void)

+static void r4k_blast_icache32_page_indexed_indexed_setup_size(void)

{

unsigned long ic_lsize = cpu_icacheline_size();

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
@@ -275,7 +275,7 @@ static void __cpuinit r4k_blast_icache_setup(void)

static void (* r4k_blast_scache_page)(unsigned long addr);

-static void __cpuinit r4k_blast_scache_page_setup(void)
+static void r4k_blast_scache_page_setup(void)
{
unsigned long sc_lsize = cpu_scache_line_size();

@@ -293,7 +293,7 @@ static void __cpuinit r4k_blast_scache_page_setup(void)

static void (* r4k_blast_scache_page_indexed)(unsigned long addr);

-static void __cpuinit r4k_blast_scache_page_indexed_setup(void)
+static void r4k_blast_scache_page_indexed_setup(void)
{
unsigned long sc_lsize = cpu_scache_line_size();

@@ -311,7 +311,7 @@ static void __cpuinit r4k_blast_scache_page_indexed_setup(void)

static void (* r4k_blast_scache)(void);

-static void __cpuinit r4k_blast_scache_setup(void)
+static void r4k_blast_scache_setup(void)
{
unsigned long sc_lsize = cpu_scache_line_size();

@@ -718,11 +718,11 @@ static inline void rm7k_erratum31(void)
}
}

-static char *way_string[] __cpuinitdata = { NULL, "direct mapped", "2-way",
+static char *way_string[] = { NULL, "direct mapped", "2-way",
"3-way", "4-way", "5-way", "6-way", "7-way", "8-way"
};

-static void __cpuinit probe_pcache(void)
+static void probe_pcache(void)
{
struct cpuinfo_mips *c = &current_cpu_data;
unsigned int config = read_c0_config();
@@ -1044,7 +1044,7 @@ static void __cpuinit probe_pcache(void)
* executes in KSEG1 space or else you will crash and burn badly. You have
* been warned.
*/
-static int __cpuinit probe_scache(void)
+static int probe_scache(void)
{
unsigned long flags, addr, begin, end, pow2;
unsigned int config = read_c0_config();
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
@@ -1123,7 +1123,7 @@ extern int r5k_sc_init(void);
extern int rm7k_sc_init(void);
extern int mips_sc_init(void);

-static void __cpuinit setup_scache(void)
+static void setup_scache(void)
{
struct cpuinfo_mips *c = &current_cpu_data;
unsigned int config = read_c0_config();
@@ -1256,7 +1256,7 @@ static void nxp_pr4450_fixup_config(void)
NXP_BARRIER();
}

-static int __cpuinitdata cca = -1;
+static int cca = -1;

static int __init cca_setup(char *str)
{
@@ -1267,7 +1267,7 @@ static int __init cca_setup(char *str)
__setup("cca=", cca_setup);

-static void __cpuinit coherency_setup(void)
+static void coherency_setup(void)
{
if (cca < 0 || cca > 7)
cca = read_c0_config() & CONF_CM_CMASK;
@@ -1310,7 +1310,7 @@ static void __cpuinit coherency_setup(void)

#ifdef CONFIG_DMA_NONCOHERENT

-static int __cpuinitdata coherentio;
+static int coherentio;

static int __init setcoherentio(char *str)
{
@@ -1322,7 +1322,7 @@ static int __init setcoherentio(char *str)
__setup("coherentio", setcoherentio);
#endif

-void __cpuinit r4k_cache_init(void)
+void r4k_cache_init(void)
{
extern void build_clear_page(void);
extern void build_copy_page(void);
diff --git a/arch/mips/mm/c-tx39.c b/arch/mips/mm/c-tx39.c
index f7c8f9c..d602f86 100644
--- a/arch/mips/mm/c-tx39.c
+++ b/arch/mips/mm/c-tx39.c
@@ -339,7 +339,7 @@ static __init void tx39_probe_cache(void)
}
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuint and __cpuexit

```
}

-void __cpuint tx39_cache_init(void)
+void tx39_cache_init(void)
{
extern void build_clear_page(void);
extern void build_copy_page(void);
diff --git a/arch/mips/mm/page.c b/arch/mips/mm/page.c
index 1417c64..c8aad63 100644
--- a/arch/mips/mm/page.c
+++ b/arch/mips/mm/page.c
@@ -65,8 +65,8 @@ UASM_L_LA(_copy_pref_both)
UASM_L_LA(_copy_pref_store)

/* We need one branch and therefore one relocation per target label. */
-static struct uasm_label __cpuintdata labels[5];
-static struct uasm_reloc __cpuintdata relocs[5];
+static struct uasm_label labels[5];
+static struct uasm_reloc relocs[5];

#define cpu_is_r4600_v1_x() ((read_c0_prid() & 0xffffffff) == 0x00002010)
#define cpu_is_r4600_v2_x() ((read_c0_prid() & 0xffffffff) == 0x00002020)
@@ -110,23 +110,23 @@ void copy_page(void *to, void *from) __attribute__((alias("copy_page_array")));
EXPORT_SYMBOL(copy_page);

-static int pref_bias_clear_store __cpuintdata;
-static int pref_bias_copy_load __cpuintdata;
-static int pref_bias_copy_store __cpuintdata;
+static int pref_bias_clear_store;
+static int pref_bias_copy_load;
+static int pref_bias_copy_store;

-static u32 pref_src_mode __cpuintdata;
-static u32 pref_dst_mode __cpuintdata;
+static u32 pref_src_mode;
+static u32 pref_dst_mode;

-static int clear_word_size __cpuintdata;
-static int copy_word_size __cpuintdata;
+static int clear_word_size;
+static int copy_word_size;

-static int half_clear_loop_size __cpuintdata;
-static int half_copy_loop_size __cpuintdata;
+static int half_clear_loop_size;
+static int half_copy_loop_size;

-static int cache_line_size __cpuintdata;
+static int cache_line_size;
#define cache_line_mask() (cache_line_size - 1)
```

[PATCH] get rid if __cpuint and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
-static inline void __cpuinit
+static inline void
pg_addiu(u32 **buf, unsigned int reg1, unsigned int reg2, unsigned int off)
{
if (cpu_has_64bit_gp_regs && DADDI_WAR && r4k_daddiu_bug()) {
@@ -146,7 +146,7 @@ pg_addiu(u32 **buf, unsigned int reg1, unsigned int reg2, unsigned int off)
}
}

-static void __cpuinit set_prefetch_parameters(void)
+static void set_prefetch_parameters(void)
{
if (cpu_has_64bit_gp_regs || cpu_has_64bit_zero_reg)
clear_word_size = 8;
@@ -245,7 +245,7 @@ static void __cpuinit set_prefetch_parameters(void)
4 * copy_word_size));
}

-static void __cpuinit build_clear_store(u32 **buf, int off)
+static void build_clear_store(u32 **buf, int off)
{
if (cpu_has_64bit_gp_regs || cpu_has_64bit_zero_reg) {
uasm_i_sd(buf, ZERO, off, A0);
@@ -254,7 +254,7 @@ static void __cpuinit build_clear_store(u32 **buf, int off)
}
}

-static inline void __cpuinit build_clear_pref(u32 **buf, int off)
+static inline void build_clear_pref(u32 **buf, int off)
{
if (off & cache_line_mask())
return;
@@ -281,7 +281,7 @@ static inline void __cpuinit build_clear_pref(u32 **buf, int off)
}
}

-void __cpuinit build_clear_page(void)
+void build_clear_page(void)
{
int off;
u32 *buf = (u32 *)&clear_page_array;
@@ -369,7 +369,7 @@ void __cpuinit build_clear_page(void)
pr_debug("\t.set pop\n");
}

-static void __cpuinit build_copy_load(u32 **buf, int reg, int off)
+static void build_copy_load(u32 **buf, int reg, int off)
{
if (cpu_has_64bit_gp_regs) {
uasm_i_ld(buf, reg, off, A1);
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuint and __cpuexit

```
@@ -378,7 +378,7 @@ static void __cpuint build_copy_load(u32 **buf, int reg, int off)
}
}
```

```
-static void __cpuint build_copy_store(u32 **buf, int reg, int off)
+static void build_copy_store(u32 **buf, int reg, int off)
{
if (cpu_has_64bit_gp_regs) {
uasm_i_sd(buf, reg, off, A0);
@@ -423,7 +423,7 @@ static inline void build_copy_store_pref(u32 **buf, int off)
}
}
```

```
-void __cpuint build_copy_page(void)
+void build_copy_page(void)
{
int off;
u32 *buf = (u32 *)&copy_page_array;
diff --git a/arch/mips/mm/sc-ip22.c b/arch/mips/mm/sc-ip22.c
index 1f602a1..96ef449 100644
--- a/arch/mips/mm/sc-ip22.c
+++ b/arch/mips/mm/sc-ip22.c
@@ -168,7 +168,7 @@ struct bcache_ops indy_sc_ops = {
.bc_inv = indy_sc_wback_invalidate
};
```

```
-void __cpuint indy_sc_init(void)
+void indy_sc_init(void)
{
if (indy_sc_probe()) {
indy_sc_enable();
diff --git a/arch/mips/mm/sc-mips.c b/arch/mips/mm/sc-mips.c
index b55c2d1..22b361f 100644
--- a/arch/mips/mm/sc-mips.c
+++ b/arch/mips/mm/sc-mips.c
@@ -100,7 +100,7 @@ static inline int __init mips_sc_probe(void)
return 1;
}
```

```
-int __cpuint mips_sc_init(void)
+int mips_sc_init(void)
{
int found = mips_sc_probe();
if (found) {
diff --git a/arch/mips/mm/sc-r5k.c b/arch/mips/mm/sc-r5k.c
index f330d38..c1cdf3 100644
--- a/arch/mips/mm/sc-r5k.c
+++ b/arch/mips/mm/sc-r5k.c
@@ -99,7 +99,7 @@ static struct bcache_ops r5k_sc_ops = {
.bc_inv = r5k_dma_cache_inv_sc
};
```

[PATCH] get rid if __cpuint and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
-void __cpuinit r5k_sc_init(void)
+void r5k_sc_init(void)
{
if (r5k_sc_probe()) {
r5k_sc_enable();
diff --git a/arch/mips/mm/sc-rm7k.c b/arch/mips/mm/sc-rm7k.c
index e3abfb2..91cbc23 100644
--- a/arch/mips/mm/sc-rm7k.c
+++ b/arch/mips/mm/sc-rm7k.c
@@ -86,7 +86,7 @@ static void rm7k_sc_inv(unsigned long addr, unsigned long size)
/*
* This function is executed in uncached address space.
*/
-static __cpuinit void __rm7k_sc_enable(void)
+static void __rm7k_sc_enable(void)
{
int i;

@@ -107,7 +107,7 @@ static __cpuinit void __rm7k_sc_enable(void)
}
}

-static __cpuinit void rm7k_sc_enable(void)
+static void rm7k_sc_enable(void)
{
if (read_c0_config() & RM7K_CONF_SE)
return;
@@ -128,7 +128,7 @@ struct bcache_ops rm7k_sc_ops = {
.bc_inv = rm7k_sc_inv
};

-void __cpuinit rm7k_sc_init(void)
+void rm7k_sc_init(void)
{
struct cpuinfo_mips *c = &current_cpu_data;
unsigned int config = read_c0_config();
diff --git a/arch/mips/mm/tlb-r3k.c b/arch/mips/mm/tlb-r3k.c
index f0cf46a..3766088 100644
--- a/arch/mips/mm/tlb-r3k.c
+++ b/arch/mips/mm/tlb-r3k.c
@@ -277,7 +277,7 @@ void __init add_wired_entry(unsigned long entrylo0, unsigned long entrylo1,
}
}

-void __cpuinit tlb_init(void)
+void tlb_init(void)
{
local_flush_tlb_all();

diff --git a/arch/mips/mm/tlb-r4k.c b/arch/mips/mm/tlb-r4k.c
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpunit and __cpuexit

index 5ce2fa7..e983cb8 100644

--- a/arch/mips/mm/tlb-r4k.c

+++ b/arch/mips/mm/tlb-r4k.c

@@ -388,7 +388,7 @@ void __init add_wired_entry(unsigned long entrylo0, unsigned long entrylo1,
* lifetime of the system

*/

-static int temp_tlb_entry __cpunitdata;

+static int temp_tlb_entry;

__init int add_temporary_entry(unsigned long entrylo0, unsigned long entrylo1,
unsigned long entryhi, unsigned long pagemask)

@@ -427,7 +427,7 @@ out:

return ret;

}

-static void __cpunit probe_tlb(unsigned long config)

+static void probe_tlb(unsigned long config)

{

struct cpuinfo_mips *c = ¤t_cpu_data;

unsigned int reg;

@@ -455,7 +455,7 @@ static void __cpunit probe_tlb(unsigned long config)

c->tlbsize = ((reg >> 25) & 0x3f) + 1;

}

-static int __cpunitdata ntlb = 0;

+static int ntlb = 0;

static int __init set_ntlb(char *str)

{

get_option(&str, &ntlb);

@@ -464,7 +464,7 @@ static int __init set_ntlb(char *str)

__setup("ntlb=", set_ntlb);

-void __cpunit tlb_init(void)

+void tlb_init(void)

{

unsigned int config = read_c0_config();

diff --git a/arch/mips/mm/tlb-r8k.c b/arch/mips/mm/tlb-r8k.c

index 4f01a3b..7b884ba 100644

--- a/arch/mips/mm/tlb-r8k.c

+++ b/arch/mips/mm/tlb-r8k.c

@@ -214,14 +214,14 @@ void __update_tlb(struct vm_area_struct * vma, unsigned long address, pte_t pte)
local_irq_restore(flags);

}

-static void __cpunit probe_tlb(unsigned long config)

+static void probe_tlb(unsigned long config)

{

struct cpuinfo_mips *c = ¤t_cpu_data;

[PATCH] get rid if __cpunit and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
c->tlbsize = 3 * 128; /* 3 sets each 128 entries */
}

-void __cpuinit tlb_init(void)
+void tlb_init(void)
{
unsigned int config = read_c0_config();
unsigned long status;
diff --git a/arch/mips/mm/tlbex.c b/arch/mips/mm/tlbex.c
index 979cf91..d9b00df 100644
--- a/arch/mips/mm/tlbex.c
+++ b/arch/mips/mm/tlbex.c
@@ -60,7 +60,7 @@ static inline int __maybe_unused r10000_llsc_war(void)
* why; it's not an issue caused by the core RTL.
*
*/
-static int __cpuinit m4kc_tlb_war(void)
+static int m4kc_tlb_war(void)
{
return (current_cpu_data.processor_id & 0xffff00) ==
(PRID_COMP_MIPS | PRID_IMP_4KC);
@@ -144,16 +144,16 @@ static inline void dump_handler(const u32 *handler, int count)
* We deliberately chose a buffer size of 128, so we won't scribble
* over anything important on overflow before we panic.
*/
-static u32 tlb_handler[128] __cpuinitdata;
+static u32 tlb_handler[128];

/* simply assume worst case size for labels and relocs */
-static struct uasm_label labels[128] __cpuinitdata;
-static struct uasm_reloc relocs[128] __cpuinitdata;
+static struct uasm_label labels[128];
+static struct uasm_reloc relocs[128];

/*
* The R3000 TLB handler is simple.
*/
-static void __cpuinit build_r3000_tlb_refill_handler(void)
+static void build_r3000_tlb_refill_handler(void)
{
long pgdc = (long)pgd_current;
u32 *p;
@@ -197,7 +197,7 @@ static void __cpuinit build_r3000_tlb_refill_handler(void)
* other one. To keep things simple, we first assume linear space,
* then we relocate it to the final handler layout as needed.
*/
-static u32 final_handler[64] __cpuinitdata;
+static u32 final_handler[64];

/*
```

[PATCH] get rid if __cpuinit and __cpuexit

[PATCH] get rid if __cpuint and __cpuexit

```
* Hazards
@@ -221,7 +221,7 @@ static u32 final_handler[64] __cpuintdata;
*
* As if we MIPS hackers wouldn't know how to nop pipelines happy ...
*/
-static void __cpuint __maybe_unused build_tlb_probe_entry(u32 **p)
+static void __maybe_unused build_tlb_probe_entry(u32 **p)
{
switch (current_cpu_type()) {
/* Found by experiment: R4600 v2.0/R4700 needs this, too. */
@@ -246,7 +246,7 @@ static void __cpuint __maybe_unused build_tlb_probe_entry(u32 **p)
*/
enum tlb_write_entry { tlb_random, tlb_indexed };

-static void __cpuint build_tlb_write_entry(u32 **p, struct uasm_label **l,
+static void build_tlb_write_entry(u32 **p, struct uasm_label **l,
struct uasm_reloc **r,
enum tlb_write_entry wmode)
{
@@ -391,7 +391,7 @@ static void __cpuint build_tlb_write_entry(u32 **p, struct uasm_label **l,
* TMP and PTR are scratch.
* TMP will be clobbered, PTR will hold the pmd entry.
*/
-static void __cpuint
+static void
build_get_pmde64(u32 **p, struct uasm_label **l, struct uasm_reloc **r,
unsigned int tmp, unsigned int ptr)
{
@@ -452,7 +452,7 @@ build_get_pmde64(u32 **p, struct uasm_label **l, struct uasm_reloc **r,
* BVADDR is the faulting address, PTR is scratch.
* PTR will hold the pgd for vmalloc.
*/
-static void __cpuint
+static void
build_get_pgd_vmalloc64(u32 **p, struct uasm_label **l, struct uasm_reloc **r,
unsigned int bvaddr, unsigned int ptr)
{
@@ -524,7 +524,7 @@ build_get_pgd_vmalloc64(u32 **p, struct uasm_label **l, struct uasm_reloc **r,
* TMP and PTR are scratch.
* TMP will be clobbered, PTR will hold the pgd entry.
*/
-static void __cpuint __maybe_unused
+static void __maybe_unused
build_get_pgde32(u32 **p, unsigned int tmp, unsigned int ptr)
{
long pgdc = (long)pgd_current;
@@ -559,7 +559,7 @@ build_get_pgde32(u32 **p, unsigned int tmp, unsigned int ptr)

#endif /* !CONFIG_64BIT */

-static void __cpuint build_adjust_context(u32 **p, unsigned int ctx)
```

[PATCH] get rid if __cpuint and __cpuexit

[PATCH] get rid if __cpuinit and __cpuexit

```
+static void build_adjust_context(u32 **p, unsigned int ctx)
{
unsigned int shift = 4 - (PTE_T_LOG2 + 1) + PAGE_SHIFT - 12;
unsigned int mask = (PTRS_PER_PTE / 2 - 1) << (PTE_T_LOG2 + 1);
@@ -585,7 +585,7 @@ static void __cpuinit build_adjust_context(u32 **p, unsigned int ctx)
uasm_i_andi(p, ctx, ctx, mask);
}

-static void __cpuinit build_get_ptep(u32 **p, unsigned int tmp, unsigned int ptr)
+static void build_get_ptep(u32 **p, unsigned int tmp, unsigned int ptr)
{
/*
* Bug workaround for the Nevada. It seems as if under certain
@@ -610,7 +610,7 @@ static void __cpuinit build_get_ptep(u32 **p, unsigned int tmp, unsigned int ptr)
UASM_i_ADDU(p, ptr, ptr,
```