

# Re: [PATCH] Fix Immediate Values x86\_64 support old gcc (v3)

---

*Source:* <http://linux.derkeiler.com/Mailing-Lists/Kernel/2008-05/msg10457.html>

---

- *From:* Mathieu Desnoyers <compudj@xxxxxxxxxxxxxxxxxxxx>
  - *Date:* Wed, 21 May 2008 22:03:09 -0400
- 

\* Mathieu Desnoyers (compudj@xxxxxxxxxxxxxxxxxxxx) wrote:

GCC < 4, on x86\_64, does not accept symbol+offset operands for "i" constraints asm statements. Fallback on generic immediate values if this compiler is detected.

Please replace the previous paragraph in the changelog by :

GCC < 4, on x86\_64, does not accept symbol+offset operands for "i" constraints asm statements. Fallback on a memory read in lieu of immediate value if this compiler is detected.

As Peter pointed out, talking of "generic immediate value" is rather confusing.

Mathieu

Changelog :

- USE\_IMMEDIATE must now be used in lieu of CONFIG\_IMMEDIATE in Makefiles and in C code.
- Every architecture implementing immediate values must declare USE\_IMMEDIATE in their Makefile.
- Tab -> spaces in Makefiles.

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

CC: Sam Ravnborg <sam@xxxxxxxx>

CC: "H. Peter Anvin" <hpa@xxxxxxxx>

CC: Jeremy Fitzhardinge <jeremy@xxxxxx>

CC: Ingo Molnar <mingo@xxxxxx>

CC: David Miller <davem@xxxxxxxx>

CC: Paul Mackerras <paulus@xxxxxx>

---

Makefile | 5 +++++

arch/powerpc/Makefile | 2 ++

Re: [PATCH] Fix Immediate Values x86\_64 support old gcc (v3)

arch/powerpc/kernel/Makefile | 2 +-  
arch/x86/Makefile | 5 +++++  
arch/x86/kernel/Makefile | 2 +-  
arch/x86/kernel/traps\_32.c | 4 +++-  
include/linux/immediate.h | 2 +-  
include/linux/module.h | 4 ++--  
kernel/Makefile | 2 +-  
kernel/module.c | 8 ++++----  
10 files changed, 24 insertions(+), 12 deletions(-)

Index: linux-2.6-sched-devel/arch/x86/Makefile

```
----- linux-2.6-sched-devel.orig/arch/x86/Makefile 2008-05-21 15:16:06.000000000 -0400  
+++ linux-2.6-sched-devel/arch/x86/Makefile 2008-05-21 15:53:00.000000000 -0400  
@@ -43,6 +43,7 @@
```

```
# temporary until string.h is fixed  
KBUILD_CFLAGS += -ffreestanding  
+ export USE_IMMEDIATE := $(CONFIG_IMMEDIATE)  
else  
BITS := 64  
UTS_MACHINE := x86_64  
@@ -78,6 +79,10 @@  
"$(CC)" -fstack-protector-all )
```

```
KBUILD_CFLAGS += $(stackp-y)  
+  
+ # x86_64 gcc 3.x has problems with passing symbol+offset in  
+ # asm "i" constraint.  
+ export USE_IMMEDIATE := $(call cc-ifversion, -ge, 0400, $(CONFIG_IMMEDIATE))  
endif
```

# Stackpointer is addressed different for 32 bit and 64 bit x86

Index: linux-2.6-sched-devel/include/linux/immediate.h

```
----- linux-2.6-sched-devel.orig/include/linux/immediate.h 2008-05-21  
15:16:06.000000000 -0400  
+++ linux-2.6-sched-devel/include/linux/immediate.h 2008-05-21 15:16:59.000000000  
-0400  
@@ -10,7 +10,7 @@  
* See the file COPYING for more details.  
*/
```

```
-#ifdef CONFIG_IMMEDIATE  
+#ifdef USE_IMMEDIATE
```

```
#include <asm/immediate.h>
```

Index: linux-2.6-sched-devel/arch/x86/kernel/Makefile

```
----- linux-2.6-sched-devel.orig/arch/x86/kernel/Makefile 2008-05-21 15:16:51.000000000
```

Re: [PATCH] Fix Immediate Values x86\_64 support old gcc (v3)

```
-0400
+++ linux-2.6-sched-devel/arch/x86/kernel/Makefile 2008-05-21 15:16:59.000000000
-0400
@@ -75,7 +75,7 @@
obj-$(CONFIG_KPROBES) += kprobes.o
obj-$(CONFIG_MODULES) += module_$(BITS).o
obj-$(CONFIG_ACPI_SRAT) += srat_32.o
-obj-$(CONFIG_IMMEDIATE) += immediate.o
+obj-$(USE_IMMEDIATE) += immediate.o
obj-$(CONFIG_EFI) += efi.o efi_$(BITS).o efi_stub_$(BITS).o
obj-$(CONFIG_DOUBLEFAULT) += doublefault_32.o
obj-$(CONFIG_KGDB) += kgdb.o
Index: linux-2.6-sched-devel/kernel/Makefile
```

```
----- linux-2.6-sched-devel.orig/kernel/Makefile 2008-05-21 15:16:52.000000000 -0400
+++ linux-2.6-sched-devel/kernel/Makefile 2008-05-21 15:16:59.000000000 -0400
@@ -75,7 +75,7 @@
obj-$(CONFIG_SYSCTL) += utsname_sysctl.o
obj-$(CONFIG_TASK_DELAY_ACCT) += delayacct.o
obj-$(CONFIG_TASKSTATS) += taskstats.o tsacct.o
-obj-$(CONFIG_IMMEDIATE) += immediate.o
+obj-$(USE_IMMEDIATE) += immediate.o
obj-$(CONFIG_MARKERS) += marker.o
obj-$(CONFIG_LATENCYTOP) += latencytop.o
obj-$(CONFIG_FTRACE) += trace/
Index: linux-2.6-sched-devel/arch/powerpc/Makefile
```

```
----- linux-2.6-sched-devel.orig/arch/powerpc/Makefile 2008-05-21 15:16:06.000000000
-0400
+++ linux-2.6-sched-devel/arch/powerpc/Makefile 2008-05-21 15:16:59.000000000
-0400
@@ -98,6 +98,8 @@
endif
endif
```

```
+export USE_IMMEDIATE := $(CONFIG_IMMEDIATE)
+
ifeq ($(CONFIG_TUNE_CELL),y)
KBUILD_CFLAGS += $(call cc-option,-mtune=cell)
endif
Index: linux-2.6-sched-devel/Makefile
```

```
----- linux-2.6-sched-devel.orig/Makefile 2008-05-21 15:16:51.000000000 -0400
+++ linux-2.6-sched-devel/Makefile 2008-05-21 15:16:59.000000000 -0400
@@ -532,6 +532,11 @@
KBUILD_CFLAGS += -pg
endif
```

```
+# arch Makefile detects if the compiler permits use of immediate values
+ifdef USE_IMMEDIATE
+KBUILD_CFLAGS += -DUSE_IMMEDIATE
```

Re: [PATCH] Fix Immediate Values x86\_64 support old gcc (v3)

```
+endif
+
# We trigger additional mismatches with less inlining
ifdef CONFIG_DEBUG_SECTION_MISMATCH
KBUILD_CFLAGS += $(call cc-option, -fno-inline-functions-called-once)
Index: linux-2.6-sched-devel/arch/powerpc/kernel/Makefile
=====
--- linux-2.6-sched-devel.orig/arch/powerpc/kernel/Makefile 2008-05-21
15:16:06.000000000 -0400
+++ linux-2.6-sched-devel/arch/powerpc/kernel/Makefile 2008-05-21 15:16:59.000000000
-0400
@@ -45,7 +45,7 @@
obj64-$(CONFIG_HIBERNATION) += swsusp_asm64.o
obj-$(CONFIG_MODULES) += module_$(CONFIG_WORD_SIZE).o
obj-$(CONFIG_44x) += cpu_setup_44x.o
-obj-$(CONFIG_IMMEDIATE) += immediate.o
+obj-$(USE_IMMEDIATE) += immediate.o

ifeq ($(CONFIG_PPC_MERGE),y)

Index: linux-2.6-sched-devel/kernel/module.c
=====
--- linux-2.6-sched-devel.orig/kernel/module.c 2008-05-21 15:16:52.000000000 -0400
+++ linux-2.6-sched-devel/kernel/module.c 2008-05-21 15:16:59.000000000 -0400
@@ -2013,7 +2013,7 @@
mod->gpl_future_syms = (void *)sechdrs[gplfutureindex].sh_addr;
if (gplfuturecrcindex)
mod->gpl_future_crcs = (void *)sechdrs[gplfuturecrcindex].sh_addr;
-#ifndef CONFIG_IMMEDIATE
+#ifdef USE_IMMEDIATE
mod->immediate = (void *)sechdrs[immediateindex].sh_addr;
mod->num_immediate =
sechdrs[immediateindex].sh_size / sizeof(*mod->immediate);
@@ -2097,7 +2097,7 @@
marker_update_probe_range(mod->markers,
mod->markers + mod->num_markers);
#endif
-#ifndef CONFIG_IMMEDIATE
+#ifdef USE_IMMEDIATE
/* Immediate values must be updated after markers */
imv_update_range(mod->immediate,
mod->immediate + mod->num_immediate);
@@ -2258,7 +2258,7 @@
/* Drop initial reference. */
module_put(mod);
unwind_remove_table(mod->unwind_info, 1);
-#ifndef CONFIG_IMMEDIATE
+#ifdef USE_IMMEDIATE
imv_unref(mod->immediate, mod->immediate + mod->num_immediate,
mod->module_init, mod->init_size);
#endif
```

Re: [PATCH] Fix Immediate Values x86\_64 support old gcc (v3)

```
@@ -2657,7 +2657,7 @@
}
#endif

#ifndef CONFIG_IMMEDIATE
#ifdef USE_IMMEDIATE
/**
 * _module_imv_update - update all immediate values in the kernel
 *
Index: linux-2.6-sched-devel/include/linux/module.h
=====
--- linux-2.6-sched-devel.orig/include/linux/module.h 2008-05-21 15:16:52.000000000
-0400
+++ linux-2.6-sched-devel/include/linux/module.h 2008-05-21 15:16:59.000000000 -0400
@@ -339,7 +339,7 @@
/* The command line arguments (may be mangled). People like
keeping pointers to this stuff */
char *args;
#ifndef CONFIG_IMMEDIATE
#ifdef USE_IMMEDIATE
struct __imv *immediate;
unsigned int num_immediate;
unsigned long *immediate_cond_end;
@@ -563,7 +563,7 @@

#endif /* CONFIG_MODULES */

#ifndef defined(CONFIG_MODULES) && defined(CONFIG_IMMEDIATE)
#ifdef defined(CONFIG_MODULES) && defined(USE_IMMEDIATE)
extern void _module_imv_update(void);
extern void module_imv_update(void);
extern int is_imv_cond_end_module(unsigned long addr1, unsigned long addr2);
Index: linux-2.6-sched-devel/arch/x86/kernel/traps_32.c
=====
--- linux-2.6-sched-devel.orig/arch/x86/kernel/traps_32.c 2008-05-21
15:16:06.000000000 -0400
+++ linux-2.6-sched-devel/arch/x86/kernel/traps_32.c 2008-05-21 15:16:59.000000000
-0400
@@ -595,7 +595,7 @@
}

DO_VM86_ERROR_INFO(0, SIGFPE, "divide error", divide_error, FPE_INTDIV,
regs->ip)
#ifndef defined(CONFIG_KPROBES) && !defined(CONFIG_IMMEDIATE)
#ifdef !defined(CONFIG_KPROBES) && !defined(USE_IMMEDIATE)
DO_VM86_ERROR(3, SIGTRAP, "int3", int3)
#endif
#endif
DO_VM86_ERROR(4, SIGSEGV, "overflow", overflow)
@@ -860,7 +860,7 @@
acpi_nmi_enable();
}
```

Re: [PATCH] Fix Immediate Values x86\_64 support old gcc (v3)

```
---  
-#if defined(CONFIG_KPROBES) || defined(CONFIG_IMMEDIATE)  
+#if defined(CONFIG_KPROBES) || defined(USE_IMMEDIATE)  
void __kprobes do_int3(struct pt_regs *regs, long error_code)  
{  
    trace_hardirqs_fixup();  
}
```

---

Mathieu Desnoyers

OpenPGP key fingerprint: 8CD5 52C3 8E3C 4140 715F BA06 3F25 A8FE 3BAE 9A68

---

Mathieu Desnoyers

OpenPGP key fingerprint: 8CD5 52C3 8E3C 4140 715F BA06 3F25 A8FE 3BAE 9A68

---

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/>