

High Precision Event Timer Driver for Linux

The High Precision Event Timer (HPET) hardware follows a specification by Intel and Microsoft, revision 1.

Each HPET has one fixed-rate counter (at 10+ MHz, hence "High Precision") and up to 32 comparators. Normally three or more comparators are provided, each of which can generate oneshot interrupts and at least one of which has additional hardware to support periodic interrupts. The comparators are also called "timers", which can be misleading since usually timers are independent of each other ... these share a counter, complicating resets.

HPET devices can support two interrupt routing modes. In one mode, the comparators are additional interrupt sources with no particular system role. Many x86 BIOS writers don't route HPET interrupts at all, which prevents use of that mode. They support the other "legacy replacement" mode where the first two comparators block interrupts from 8254 timers and from the RTC.

The driver supports detection of HPET driver allocation and initialization of the HPET before the driver module `_init` routine is called. This enables platform code which uses timer 0 or 1 as the main timer to intercept HPET initialization. An example of this initialization can be found in `arch/x86/kernel/hpet.c`.

The driver provides a userspace API which resembles the API found in the RTC driver framework. An example user space program is provided in [file:samples/timers/hpet_example.c](#)