## NAME

**kvmclock** - Para-virtualized clock driver for x86 KVM guests

## SYNOPSIS

To compile this driver into the kernel, place the following line in your kernel configuration file:

**device kvm_clock**

## DESCRIPTION

This driver reads time-keeping information from the para-virtualized clock device provided by the KVM hypervisor on Linux hosts. The **kvmclock** driver is only implemented on i386 and amd64 platforms. It acts as a timecounters(4) device and is preferred over the Time Stamp Counter (TSC) when available. The driver exports timekeeping information via */dev/pvclock*, enabling the implementation of clock_gettime(2) and related functions without entering the kernel.

The **kvmclock** driver works by accessing a per-vCPU timekeeping structure maintained by the hypervisor. It uses a combination of TSC readings and information from the shared structure to produce a high-resolution timecounter which is invariant under hypervisor events such as vCPU migration and live VM migration.

## SYSCTL VARIABLES

The following variables are available as both sysctl(8) variables and loader(8) tunables:

*dev.kvmclock.0.vdso_enable_without_rdtscp*

> By default, timekeeping information is exported to userspace only when the (virtual) CPU announces support for the "rdtscp" instruction. Setting this sysctl to 1 overrides this behavior, allowing timekeeping information to be exported even in the absence of "rdtscp" support. However, this breaks compatibility with copies of */lib/libc.so.7* released prior to FreeBSD 14.0, and statically linked binaries which embed a copy of the system C library. Thus, this sysctl value should not be changed if the system may execute binaries older than FreeBSD 14.0.

*dev.kvmclock.0.vdso_force_unstable*

> Mark the timecounter as unstable for userspace consumers. This is mostly useful for debugging the driver and userspace timekeeping code, and generally should not be touched.

## SEE ALSO

timecounters(4)

## HISTORY

The **kvmclock** driver first appeared in FreeBSD 13.1.