

WatchDog Module Parameters

This file provides information on the module parameters of many of the Linux watchdog drivers. Watchdog driver parameter specs should be listed here unless the driver has its own driver-specific information file.

See Documentation/admin-guide/kernel-parameters.rst for information on providing kernel parameters for builtin drivers versus loadable modules.

watchdog core:

open_timeout:

Maximum time, in seconds, for which the watchdog framework will take care of pinging a running hardware watchdog until userspace opens the corresponding /dev/watchdogN device. A value of 0 means an infinite timeout. Setting this to a non-zero value can be useful to ensure that either userspace comes up properly, or the board gets reset and allows fallback logic in the bootloader to try something else.

acquirewdt:

wdt_stop:

Acquire WDT 'stop' io port (default 0x43)

wdt_start:

Acquire WDT 'start' io port (default 0x443)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

advantechwdt:

wdt_stop:

Advantech WDT 'stop' io port (default 0x443)

wdt_start:

Advantech WDT 'start' io port (default 0x443)

timeout:

Watchdog timeout in seconds. $1 \leq \text{timeout} \leq 63$, default=60.

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

alim1535_wdt:

timeout:

Watchdog timeout in seconds. ($0 < \text{timeout} < 18000$, default=60)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

alim7101_wdt:

timeout:

Watchdog timeout in seconds. ($1 \leq \text{timeout} \leq 3600$, default=30)

use_gpio:

Use the gpio watchdog (required by old cobalt boards). default=0/off/no

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

ar7_wdt:

margin:

Watchdog margin in seconds (default=60)

nowayout:

Disable watchdog shutdown on close (default=kernel config parameter)

armada_37xx_wdt:

timeout:

Watchdog timeout in seconds. (default=120)

nowayout:
Disable watchdog shutdown on close (default=kernel config parameter)

at91rm9200_wdt:

wdt_time:
Watchdog time in seconds. (default=5)

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

at91sam9_wdt:

heartbeat:
Watchdog heartbeats in seconds. (default = 15)

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

bcm47xx_wdt:

wdt_time:
Watchdog time in seconds. (default=30)

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

coh901327_wdt:

margin:
Watchdog margin in seconds (default 60s)

cpu5wdt:

port:
base address of watchdog card, default is 0x91

verbose:
be verbose, default is 0 (no)

ticks:
count down ticks, default is 10000

cpwd:

wd0_timeout:
Default watchdog0 timeout in 1/10secs

wd1_timeout:
Default watchdog1 timeout in 1/10secs

wd2_timeout:
Default watchdog2 timeout in 1/10secs

da9052wdt:

timeout:
Watchdog timeout in seconds. $2 \leq \text{timeout} \leq 131$, default=2.048s

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

davinci_wdt:

heartbeat:
Watchdog heartbeat period in seconds from 1 to 600, default 60

ebc-c384_wdt:

timeout:
Watchdog timeout in seconds. ($1 \leq \text{timeout} \leq 15300$, default=60)

nowayout:

Watchdog cannot be stopped once started

ep93xx_wdt:

nowayout:

Watchdog cannot be stopped once started

timeout:

Watchdog timeout in seconds. ($1 \leq \text{timeout} \leq 3600$, default=TBD)

eurotechwdt:

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

io:

Eurotech WDT io port (default=0x3f0)

irq:

Eurotech WDT irq (default=10)

ev:

Eurotech WDT event type (default is *int*)

gef_wdt:

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

geodewdt:

timeout:

Watchdog timeout in seconds. $1 \leq \text{timeout} \leq 131$, default=60.

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

i6300esb:

heartbeat:

Watchdog heartbeat in seconds. ($1 < \text{heartbeat} < 2046$, default=30)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

iTCO_wdt:

heartbeat:

Watchdog heartbeat in seconds. ($2 < \text{heartbeat} < 39$ (TCO v1) or 613 (TCO v2), default=30)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

iTCO_vendor_support:

vendorsupport:

iTCO vendor specific support mode, default=0 (none), 1=SuperMicro Pent3, 2=SuperMicro Pent4+, 911=Broken SMI BIOS

ib700wdt:

timeout:

Watchdog timeout in seconds. $0 \leq \text{timeout} \leq 30$, default=30.

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

ibmasr:

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

imx2_wdt:

timeout:

Watchdog timeout in seconds (default 60 s)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

indydog:

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

iop_wdt:

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

it8712f_wdt:

margin:

Watchdog margin in seconds (default 60)

nowayout:

Disable watchdog shutdown on close (default=kernel config parameter)

it87_wdt:

nogameport:

Forbid the activation of game port, default=0

nocir:

Forbid the use of CIR (workaround for some buggy setups); set to 1 if

system resets despite watchdog daemon running, default=0

exclusive:

Watchdog exclusive device open, default=1

timeout:

Watchdog timeout in seconds, default=60

testmode:

Watchdog test mode (1 = no reboot), default=0

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

ixp4xx_wdt:

heartbeat:

Watchdog heartbeat in seconds (default 60s)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

machzwd:

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

action:

after watchdog resets, generate: 0 = RESET(*) 1 = SMI 2 = NMI 3 = SCI

max63xx_wdt:

heartbeat:

Watchdog heartbeat period in seconds from 1 to 60, default 60

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

nodelay:

Force selection of a timeout setting without initial delay (max6373/74 only, default=0)

mixcomwd:

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

mpc8xxx_wdt:

timeout:

Watchdog timeout in ticks. (0<timeout<65536, default=65535)

reset:

Watchdog Interrupt/Reset Mode. 0 = interrupt, 1 = reset

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

mv64x60_wdt:

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

ni903x_wdt:

timeout:

Initial watchdog timeout in seconds (0<timeout<516, default=60)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

nic7018_wdt:

timeout:

Initial watchdog timeout in seconds (0<timeout<464, default=80)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

omap_wdt:

timer_margin:

initial watchdog timeout (in seconds)

early_enable:

Watchdog is started on module insertion (default=0)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

orion_wdt:

heartbeat:

Initial watchdog heartbeat in seconds

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

pc87413_wdt:

io:

pc87413 WDT I/O port (default: io).

timeout:

Watchdog timeout in minutes (default=timeout).

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

pika_wdt:

heartbeat:

Watchdog heartbeats in seconds. (default = 15)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

pnx4008_wdt:

heartbeat:

Watchdog heartbeat period in seconds from 1 to 60, default 19

nowayout:

Set to 1 to keep watchdog running after device release

pnx833x_wdt:

timeout:

Watchdog timeout in Mhz (68Mhz clock), default=2040000000 (30 seconds)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

start_enabled:

Watchdog is started on module insertion (default=1)

rc32434_wdt:

timeout:

Watchdog timeout value, in seconds (default=20)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

riowd:

riowd_timeout:

Watchdog timeout in minutes (default=1)

s3c2410_wdt:

tmr_margin:

Watchdog tmr_margin in seconds. (default=15)

tmr_atboot:

Watchdog is started at boot time if set to 1, default=0

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

soft_noboot:

Watchdog action, set to 1 to ignore reboots, 0 to reboot

debug:

Watchdog debug, set to >1 for debug, (default 0)

sa1100_wdt:

margin:

Watchdog margin in seconds (default 60s)

sb_wdog:

timeout:

Watchdog timeout in microseconds (max/default 8388607 or 8.3ish secs)

sbc60xxwdt:

wdt_stop:

SBC60xx WDT 'stop' io port (default 0x45)

wdt_start:

SBC60xx WDT 'start' io port (default 0x443)

timeout:

Watchdog timeout in seconds. (1<=timeout<=3600, default=30)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

sbc7240_wdt:

timeout:

Watchdog timeout in seconds. (1<=timeout<=255, default=30)

nowayout:

Disable watchdog when closing device file

sbc8360:

timeout:

Index into timeout table (0-63) (default=27 (60s))

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

sbc_epx_c3:

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

sbc_fitpc2_wdt:

margin:

Watchdog margin in seconds (default 60s)

nowayout:

Watchdog cannot be stopped once started

sbsa_gwdt:

timeout:

Watchdog timeout in seconds. (default 10s)

action:

Watchdog action at the first stage timeout, set to 0 to ignore, 1 to panic. (default=0)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

sc1200wdt:

isapnp:

When set to 0 driver ISA PnP support will be disabled (default=1)

io:

io port

timeout:

range is 0-255 minutes, default is 1

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

sc520_wdt:

timeout:

Watchdog timeout in seconds. (1 <= timeout <= 3600, default=30)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

sch311x_wdt:

force_id:

Override the detected device ID

therm_trip:

Should a ThermTrip trigger the reset generator

timeout:

Watchdog timeout in seconds. 1<= timeout <=15300, default=60

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

scx200_wdt:

margin:

Watchdog margin in seconds

nowayout:

Disable watchdog shutdown on close

shwdt:

clock_division_ratio:

Clock division ratio. Valid ranges are from 0x5 (1.31ms) to 0x7 (5.25ms). (default=7)

heartbeat:

Watchdog heartbeat in seconds. (1 <= heartbeat <= 3600, default=30)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

smc37b787_wdt:

timeout:

range is 1-255 units, default is 60

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

sofwdog:

soft_margin:

Watchdog soft_margin in seconds. (0 < soft_margin < 65536, default=60)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

soft_noboot:

Sofdog action, set to 1 to ignore reboots, 0 to reboot (default=0)

stmp3xxx_wdt:

heartbeat:

Watchdog heartbeat period in seconds from 1 to 4194304, default 19

tegra_wdt:

heartbeat:

Watchdog heartbeats in seconds. (default = 120)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

ts72xx_wdt:

timeout:

Watchdog timeout in seconds. (1 <= timeout <= 8, default=8)

nowayout:

Disable watchdog shutdown on close

tw4030_wdt:

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

txx9wdt:

timeout:

Watchdog timeout in seconds. (0<timeout<N, default=60)

nowayout:

Watchdog cannot be stopped once started (default=kernel config parameter)

uniphier_wdt:

timeout:

Watchdog timeout in power of two seconds. (1 <= timeout <= 128, default=64)

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

w83627hf_wdt:

wdt_io:
w83627hf/thf WDT io port (default 0x2E)

timeout:
Watchdog timeout in seconds. 1 <= timeout <= 255, default=60.

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

w83877f_wdt:

timeout:
Watchdog timeout in seconds. (1<=timeout<=3600, default=30)

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

w83977f_wdt:

timeout:
Watchdog timeout in seconds (15..7635), default=45)

testmode:
Watchdog testmode (1 = no reboot), default=0

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

wafer5823wdt:

timeout:
Watchdog timeout in seconds. 1 <= timeout <= 255, default=60.

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

wdt285:

soft_margin:
Watchdog timeout in seconds (default=60)

wdt977:

timeout:
Watchdog timeout in seconds (60..15300, default=60)

testmode:
Watchdog testmode (1 = no reboot), default=0

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

wm831x_wdt:

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)


wm8350_wdt:

nowayout:
Watchdog cannot be stopped once started (default=kernel config parameter)

sun4v_wdt:

timeout_ms:
Watchdog timeout in milliseconds 1..180000, default=60000)

nowayout:



Watchdog cannot be stopped once started

