



## ***Linux Ubuntu 22.4.5 Manual Pages on command 'IO::Async::LoopTests.3pm'***

***\$ man IO::Async::LoopTests.3pm***

IO::Async::LoopTests(3pm) User Contributed Perl Documentation IO::Async::LoopTests(3pm)

### NAME

"IO::Async::LoopTests" - acceptance testing for IO::Async::Loop subclasses

### SYNOPSIS

```
use IO::Async::LoopTests;

run_tests( 'IO::Async::Loop::Shiney', 'io' );
```

### DESCRIPTION

This module contains a collection of test functions for running acceptance tests on IO::Async::Loop subclasses. It is provided as a facility for authors of such subclasses to ensure that the code conforms to the Loop API required by IO::Async.

### TIMING

Certain tests require the use of timers or timed delays. Normally these are counted in units of seconds. By setting the environment variable "TEST\_QUICK\_TIMERS" to some true value, these timers run 10 times quicker, being measured in units of 0.1 seconds instead. This value may be useful when running the tests interactively, to avoid them taking too long. The slower timers are preferred on automated smoke-testing machines, to help guard against false negatives reported simply because of scheduling delays or high system load while testing.

TEST\_QUICK\_TIMERS=1 ./Build test

## FUNCTIONS

run\_tests

```
run_tests( $class, @tests )
```

Runs a test or collection of tests against the loop subclass given. The class being tested is loaded by this function; the containing script does not need to "require" or "use" it first.

This function runs "Test::More::plan" to output its expected test count; the containing script should not do this.

## TEST SUITES

The following test suite names exist, to be passed as a name in the @tests argument to "run\_tests":

io

Tests the Loop's ability to watch filehandles for IO readiness

timer

Tests the Loop's ability to handle timer events

signal

Tests the Loop's ability to watch POSIX signals

idle

Tests the Loop's support for idle handlers

child

Tests the Loop's support for watching child processes by PID

control

Tests that the "run", "stop", "loop\_once" and "loop\_forever" methods behave correctly

AUTHOR

Paul Evans <leonerdd@leonerdd.org.uk>

perl v5.30.0

2019-11-26

IO::Async::LoopTests(3pm)