Timer HWIP Technical Specification

Overview

This document specifies RISC-V Timer hardware IP functionality. This module conforms to the Comportable guideline for peripheral functionality. See that document for integration overview within the broader top level system.

Features

- 64-bit timer with 12-bit prescaler and 8-bit step register
- Compliant with RISC-V privileged specification v1.11
- Configurable number of timers per hart and number of harts

Description

The timer module provides a configurable number of 64-bit counters where each counter increments by a step value whenever the prescaler times out. Each timer generates an interrupt if the counter reaches (or is above) a programmed value. The timer is intended to be used by the processors to check the current time relative to the reset or the system power-on.

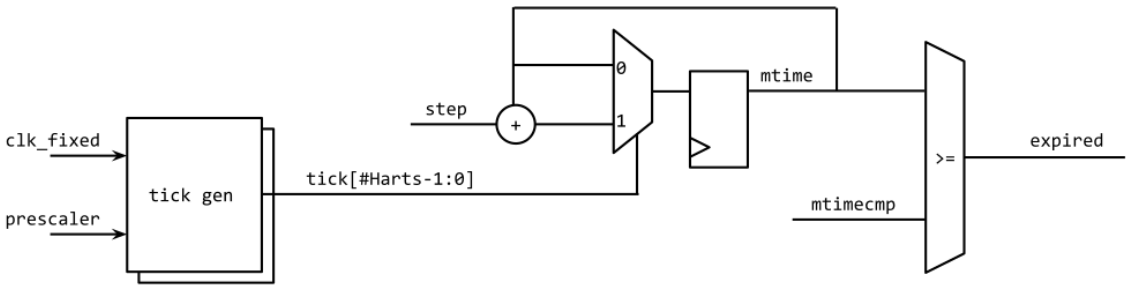
In this version, the timer doesn't consider low-power modes and assumes the clock is neither turned off nor changed during runtime.

Compatibility

The timer IP provides memory-mapped registers equivalent to `mtime` and `mtimecmp` which can be used as the machine-mode timer registers defined in the RISC-V privileged spec. Additional features such as prescaler, step, and a configurable number of timers and harts have been added.

Theory of Operation

Block Diagram



The timer module is composed of tick generators, counters, and comparators. A tick generator creates a tick every time its internal counter hits the `CFG0.prescaler` value. The tick is used to increment `mtime` by the `CFG0.step` value. The 64-bit `mtime` value is compared with the 64-bit `mtimecmp`. If `mtime` is greater than or equal to `mtimecmp`, the timer raises an interrupt.

Design Details

Tick Generator

The tick module inside the timer IP is used to generate a fixed period of pulse signal. This allows creation of a call-clock timer tick such as 1us or 10us regardless of the system clock period. It is useful if the system has more than one clock as a clock source. The firmware just needs to adjust the `CFG0.prescaler` value and the actual timer interrupt handling routine does not need a variable clock period to update `mtimecmp`.

For instance, if a system switches between 48MHz and 200MHz clocks, a prescaler value of **47** for 48MHz and **199** for 200MHz will generate a 1us tick. In this version, the timer only supports a single fixed clock, so the firmware should change `CFG0.prescaler` appropriately.

Configurable number of timers and harts

The timer IP supports more than one HART and/or more than one timer per hart. Each hart has a set of tick generator and counter. It means the timer IP has the same number of prescalers, steps, and `mtime` registers as the number of harts.

Each hart can have multiple sets of `mtimecmp`, comparator logic, and expire interrupt signals. This version of the IP is fixed to have one Hart and one Timer per Hart.