Idle injection and cpu power topology description

Status of development

- Idle injection mechanism merged upstream => v4.19
  - Provides the API to create threads playing idle
  - Based on smpboot threads (uses hotplug generic code)
  - Set idle and run durations

- Became a component of the drivers/powercap
  - Not usable alone
  - Needs another component to use the framework
  - Can be re-used by the ‘intel_powerclamp’ as requested by Rafael J. Wysocki

Usage

- All idle threads are created at boot time on each CPU
  - idle-inject/N

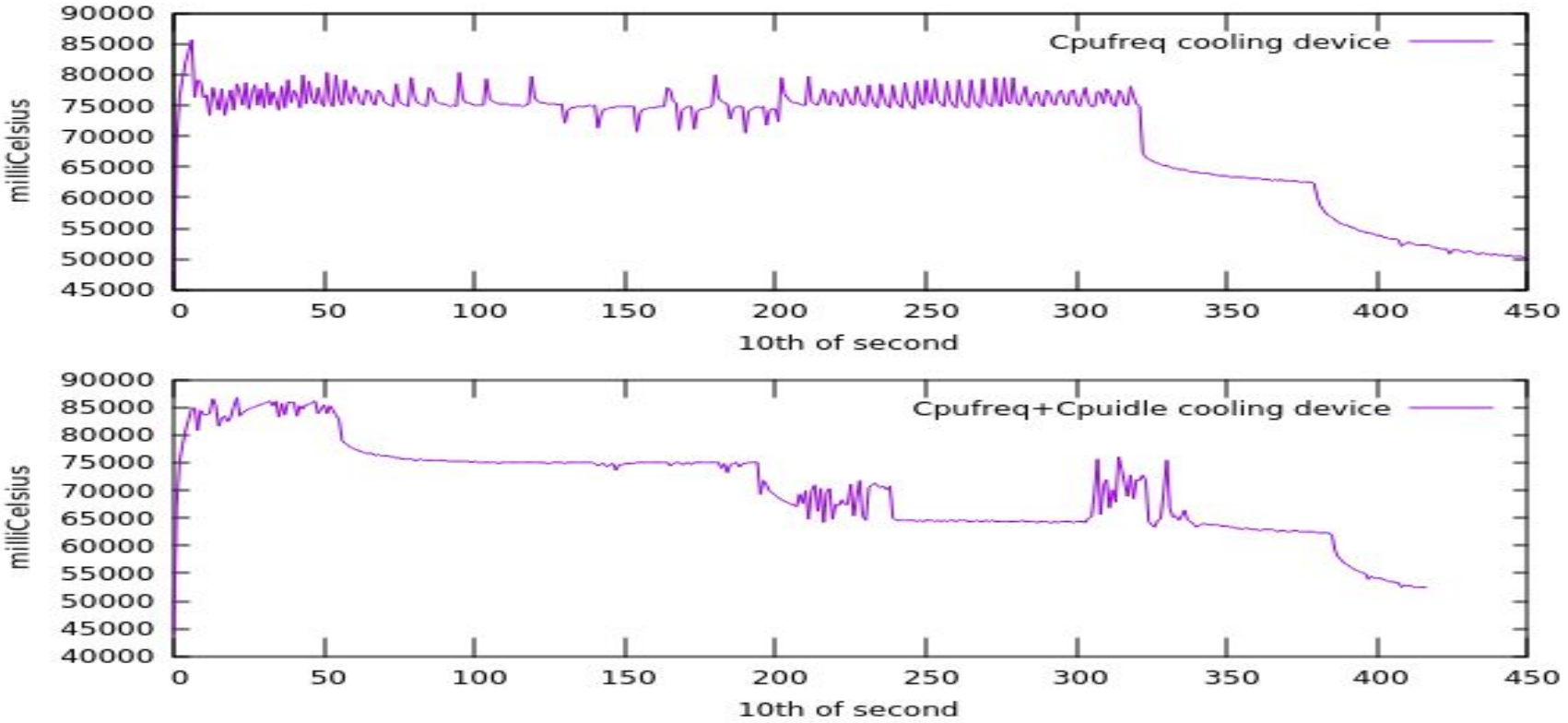
- Simple API
  1. idle_inject_register(struct cpumask *cpumask) => ii_dev
  2. idle_inject_set_duration(ii_dev, run_ms, idle_ms)
  3. idle_inject_start(ii_dev)
  4. idle_inject_stop(ii_dev)

  Up to the caller to manage concurrency

Cooling device

- Different use cases
  - Boards without cpufreq can use idle injection as an alternative
  - When the cpufreq cooling fails to mitigate the idle injection can be an alternative
  - Combining freq changes and idle injection can improve the performances (needs proof)

- Prototype shows:
  - Idle injection cooling device allows to drop the static leakage
  - Mixing idle injection and frequency changes allows to smooth temperature changes
  - Idle injections increase the latency but also the throughput


milliCelsius
90000
85000
80000
75000
70000
65000
60000
55000
50000
45000
0
50
100
150
200
250
300
350
400
450
10th of second
Cpufreq cooling device
milliCelsius
90000
85000
80000
75000
70000
65000
60000
55000
50000
45000
40000
0
50
100
150
200
250
300
350
400
450
10th of second
Cpufreq+Cpuidle cooling device

Problems to solve

- Idle injection, OPP changes:
  - Same cooling device with different policies ?
    - Governor ?
    - Cooling device itself ?
    - What about DT ?
  - Different cooling device ?
    - How to specify which one to use ?

- Description of the cooling device in the DT:
  - Back compatibility ?
  - Switching the cooling strategy ?
  - Latest Viresh's changes allow to specify the CPUs acting as cooling device

Problems to solve

- Based on capacity equivalence we should be able to combine OPP+idle and OPP decreasing

- Do the proof combining idle injection and OPP changes is worth to
  - Deepest idle states are expensive (cache flush, exit latency, timings, ...)
  - Gap between OPPs tend to be smaller and number of OPP are higher
  - Proof can be done mathematically (February - Internship)