LINUX KERNEL:
STOP OVER-COOLING!

Thara Gopinath, Linaro Inc
Ram Chandrasekar, Qualcomm Innovation Centre
HARDWARE CONSTRAINT / PROBLEM STATEMENT
TIMING CLOSURE

- Any logic circuit will have a delay to propagate an input to output.
- This delay should be less than the time period between synchronizing clocks pulses.
- If the delay is greater than a clock cycle, the circuit will not function as expected.
- This is prevalent across process nodes.
TEMPERATURE AND TIMING CLOSURE

- At lower temperatures, certain voltage and frequency combination **CANNOT** meet timing closure.

- Leads to non-functional system under extremely cold temperatures (< 0 degree Celsius)

- To close timing, the operating voltage must be increased.

- Opposite of what is required to cool down the system.
SOFTWARE / LINUX KERNEL
Linux Thermal Management Framework

- In-kernel framework exists to monitor temperature.

- ENTIRE FRAMEWORK IS BUILT AROUND MITIGATING RISING TEMPERATURE!

- So, what is needed?
  - Thermal framework should handle descending temperature.
  - Thermal framework should support warming devices
Linux Thermal Management Framework

- **Thermal Zone**
  - Area/device in a SoC that has thermal constraint.
  - Driver/Manager to manage the temperature requirements

- **Sensors**
  - Devices with temperature sensing capabilities (I2C ADC converters, bandgaps).
  - Provides temperature data to the framework.

- **Trip Point**
  - Point in the temperature domain upon crossing which a cooling action is initiated

- **Cooling Devices**
  - Devices providing control on power dissipation.
  - Range of cooling states

- **Thermal Governor**
  - Algorithm to manage the thermal zone temperature.
TRIP POINTS

EXISTING FRAMEWORK

HOT TRIP POINTS
Point in temperature domain, upon crossing which system undertakes a specified cooling action (ACTIVE, PASSIVE, HOT, CRITICAL)

COLD TRIP POINTS
Point in temperature domain, upon crossing which system undertakes a specified warming action

EXTENSION NEEDED

STATUS: WORK NOT YET STARTED
COOLING DEVICES AND DRIVERS

**EXISTING FRAMEWORK**

<table>
<thead>
<tr>
<th>ACTIVE COOLING DEVICES AND DRIVERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOFTWARE BASED COOLING MECHANISMS(CPUFREQ COOLING, IDLE INJECTION)</td>
</tr>
</tbody>
</table>

**EXTENSION NEEDED**

| SOFTWARE BASED WARMING MECHANISMS(GENPD BASED WARMING, DISABLING LOWER OPERATING POINTS OF DEVICES) |
| RESOURCE SPECIFIC WARMING MECHANISMS |

**STATUS:**

- **UPSTREAMED:** Qualcomm AOSS based warming devices
- **IN PROGRESS:** Generic Power domain based warming device driver ([https://lkml.org/lkml/2020/6/3/1112](https://lkml.org/lkml/2020/6/3/1112))
- **NOT YET STARTED:** Disabling lower OPP of devices.

Revisit term “Cooling” in the thermal framework.
THERMAL GOVERNOR

EXISTING FRAMEWORK

- STEP-WISE, IPA, BANG_BANG
- SUPPORT FOR MONITORING AND MITIGATING RISING TEMPERATURE (INTERRUPT BASED AND POLLING)

EXTENSION NEEDED

- SUPPORT FOR MONITORING AND MITIGATING FALLING TEMPERATURE (TRIGGERING WARMING ACTION)

STATUS: WORK NOT YET STARTED
Thank you